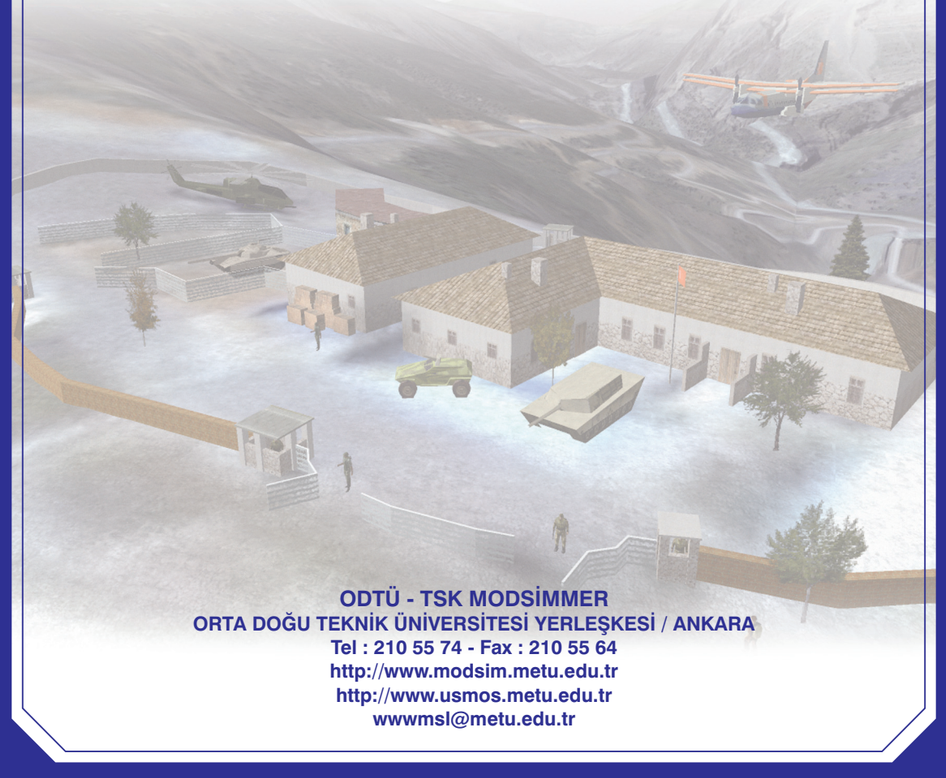


ODTÜ - TSK MODSİMMER
ORTA DOĞU TEKNİK ÜNİVERSİTESİ YERLEŞKESİ / ANKARA
Tel : 210 55 74 - Fax : 210 55 64
http://www.modsim.metu.edu.tr
http://www.usmos.metu.edu.tr
wwwmsl@metu.edu.tr

USMOS 2007
18 - 19 Nisan
ODTÜ ANKARA

USMOS 2007

ODTÜ - TSK
MODSİMMER

İKİNCİ ULUSAL SAVUNMA UYGULAMALARI MODELLEME ve SİMÜLASYON KONFERANSI

BİLDİRİ KİTABI

18 - 19 Nisan 2007
ODTÜ ANKARA

USMOS 2007

İKİNCİ ULUSAL SAVUNMA UYGULAMALARI MODELLEME VE SİMÜLASYON KONFERANSI

BİLDİRİ KİTABI

Genelkurmay Başkanlığı
Milli Savunma Bakanlığı
Savunma Sanayii Müsteşarlığı
ODTÜ
ODTÜ-TSK MODSİMMER
Kara Harp Okulu, Savunma Bilimleri Enstitüsü
Deniz Harp Okulu, Deniz Bilimleri ve Mühendisliği Enstitüsü
Hava Harp Okulu, Havacılık ve Uzay Teknolojileri Enstitüsü

18-19 NİSAN 2007
ODTÜ KÜLTÜR VE KONGRE MERKEZİ
ANKARA

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

Basım Yeri

Altan Matbaacılık, Ankara

0.312 230 57 41

ÖNSÖZ

Milli Savunma Bakanlığı, Türk Silahlı Kuvvetleri, üniversiteler, kamu araştırma kurumları ve firmalar tarafından yürütülen Modelleme ve Simülasyon (MODSİM) alanındaki temel ve uygulamalı araştırma ile teknoloji ve sistem geliştirme çalışmalarının paylaşılması, MODSİM sistem ve teknolojilerindeki gelişmelerin değerlendirilmesi, yeni MODSİM strateji ve hedeflerinin tartışılması amacıyla iki yılda bir "Ulusal Savunma Uygulamaları Modelleme Simülasyon Konferansı (USMOS)" düzenlenmektedir.

Birinci USMOS konferansı ODTÜ-TSK Modelleme ve Simülasyon Araştırma ve Uygulama Merkezi (MODSİMMER) ile Savunma Sanayii Müsteşarlığı koordinatörlüğünde, Genelkurmay Başkanlığı'nın desteği ile 2-3 Haziran 2005 tarihlerinde düzenlenmiştir. Yaklaşık 500 kişinin katıldığı USMOS 2005 konferansında, 28 bildiri sunulmuş, 18 firma ürünlerini ve çalışmalarını tanıtmıştır.

İkinci USMOS konferansı (USMOS 2007) 18-19 Nisan 2007 tarihlerinde, ODTÜ-TSK MODSİMMER tarafından, Milli Savunma Bakanlığı, Genelkurmay Başkanlığı, Savunma Sanayii Müsteşarlığı desteği, Kara Harp Okulu Savunma Bilimleri Enstitüsü, Deniz Harp Okulu Deniz Bilimleri ve Mühendisliği Enstitüsü ve Hava Harp Okulu Havacılık ve Uzay Teknolojileri Enstitüsü işbirliğiyle düzenlenmiştir. Ana teması "MODSİM Sistemlerinin Savunma Amaçlı Karar Destek Süreçlerine Entegrasyonu ve Kullanımının Yaygınlaştırılması" olan USMOS 2007'de, 12 oturumda 47 adet bildiri sunulmuştur. Ayrıca konferans programı kapsamında "Modelleme ve Simülasyon Alanında ArGe Projeleri" konulu bir panel düzenlenmiştir.

NATO ülkelerinin ve savunma sanayi firmalarının stratejik öncelikleri arasında yer alan "MODSİM Yeteneğine Sahip Olmak" hedefi ile ilgili olarak, Türkiye'de de MODSİM faaliyetleri, ürünleri, bildirileri ve projelerinde nitel ve nicel olarak artış görülmektedir. Bu artış ve konferansa gösterilen yüksek ilgi ülkemizde MODSİM alanında önemli bilgi birikimi ve deneyiminin bulunduğunun göstergesidir. Diğer taraftan biliyoruz ki savunma ihtiyacına yönelik oldukça karmaşık bir çok MODSİM projesi yerli olanaklar kullanarak Türk Savunma Sanayi tarafından başarıyla tamamlanmış veya yürütülmektedir. Üstelik son yıllarda bilişim sektörünün en karmaşık ürünlerinden olan MODSİM sistemlerinin Türk Savunma Sanayi tarafından Güney Kore, Hollanda ve Bangladeş gibi ülkelere ihracatı dahi gerçekleştirilmiştir. Bu alanda ithal ikamesi önemli ölçüde sağlanmaktadır. Artık Türk Savunma Sanayii'ndeki küçük veya büyük ölçekli bir çok yerli firma savunma alanındaki karmaşık MODSİM ihtiyaçlarına yönelik mal ve hizmetler üretebilmektedir.

Ülkemizdeki Modelleme ve Simülasyon alanında kazanılmış bu bilgi birikiminin rastlantısal olmadığı bilinmelidir. Bu alandaki sistematik çalışmalar 1997 yılında düzenlenen TSK MODSİM Semineri ile başlatılmıştır. Gnkur.Bşk.lığı BİLKARDEM

Bşk.lığı genel koordinatörlüğünde hazırlanan "Yapılabilirlik Etüdü" ışığında Harp Ak.K.lığı "Atatürk Kültür ve Harp Oyunu Merkezi" 26 Mayıs 1998 tarihinde hizmete girmiştir. 19 Kasım 1998 tarihinde Genelkurmay Başkanlığı, Savunma Sanayii Müsteşarlığı ve Orta Doğu Teknik Üniversitesi arasında imzalanan bir protokol ile MODSİM Laboratuarı (MODSİMMER) 24 Haziran 1999 tarihinde, İTÜ-TSK MODSİM Laboratuarı ise 23 Mart 2000 tarihinde açılmış, ve ihtiyaç makamının tanımlamaları doğrultusunda ilk MODSİM projeleri başlatılmıştır. 1999 yılında da ODTÜ Enformatik Enstitüsü'ne bağlı olarak Modelleme ve Simülasyon Anabilim Dalı kurulmuş ve bu alanda Yüksek Lisans Programı açılmıştır. ODTÜ-TSK MODSİMMER, 24 Mayıs 2001 tarihinde resmi gazetede yayımlanan yönetmelik gereği "ODTÜ-TSK MODSİM Araştırma ve Uygulama Merkezi"ne dönüştürülmüştür.

Kısa bir geçmişi verilen MODSİM araştırma, faaliyet ve projelerinin çıktılarını MODSİM camiası ilk kez 2005 yılında düzenlenen USMOS 2005'de görme fırsatı yakalamıştır. Bugün araştırma ve teknoloji geliştirme çalışmalarının sabırlı ve sistematik bir çaba gerektirdiğini; ancak alınan sonuçların alınan bu riske ve çabaya değdiğini görüyoruz. Ülkemizde uzun çabalar sonucunda oluşturulan bu kritik altyapı ve kabiliyetin bilincini bu konferans vesilesiyle sizlerle paylaştığımızı düşünüyor ve güçlü olan bu yönümüzü daha fazla güçlendirip bu alanda dünya pazarında marka ve lider oyunculardan biri olabileceğimize inanıyoruz.

Bu konferansın düzenlenmesine katkıda bulunan Genelkurmay Başkanlığı'na, Milli Savunma Bakanlığı'na, Savunma Sanayii Müsteşarlığı'na, Kara Harp Okulu Savunma Bilimleri Enstitüsü'ne, Deniz Harp Okulu Deniz Bilimleri ve Mühendisliği Enstitüsü'ne Hava Harp Okulu Havacılık ve Uzay Teknolojileri Enstitüsü'ne, ODTÜ Rektörlüğü'ne ve Savunma Sanayii İmalatçılar Derneği'ne; konferansın ana konuşmacısı Boğaziçi Üniversitesi Endüstri Mühendisliği Bölüm Başkanı Prof. Dr. Gülay Barbarosoğlu'na; bilgi ve birikimlerini bizimle paylaşan değerli panel konuşmacılarına, oturum başkanlarına, yazarlara, bildirileri değerlendiren hakemlere, konferansın kurullarında yer alan üyelere, tüm katılımcılara, sergi alanında yer alan ve konferans sponsoru olan üniversitelere. araştırma kurumlarına, savunma sanayi kuruluşlarına teşekkür ederiz.

Doç. Dr. Veysi İşler
ODTÜ-TSK MODSİMMER Başkanı
Koordinasyon ve Yürütme Kurulları Adına

DÜZENLEYEN KURULUŞLAR

KONFERANS KURULLARI

Koordinasyon Kurulu

Doç. Dr. Veysi İŞLER (Konferans Başkanı, ODTÜ-TSK MODSİMMER)
Prof. Dr. Semih BİLGEN (Program Kurulu Başkanı, ODTÜ)
Y. Müh. Alb. Ziya İPEKKAN (Gnkur. BİLKARDEM Bşk.lığı)
Dr. Y. Müh. Alb. Erdal TORUN (MSB ARGE ve Tekno. D. Bşk.lığı)
Doç. Dr. Elife ÜNAL (MSB SSM.lığı)
Prof. Dr. Y. Müh. Alb. Taner ALTINOK (KHO, Sav. Bil. Ens.)
Dz. Yb. Erhan ERKAHVECİ (DHO, Dz. Bil. ve Müh. Ens.)
Yrd. Doç. Dr. Hv. Müh. Alb. Sefer KURNAZ (HHO, Hvc. ve Uzy. Tekno. Ens.)
Dr. Faruk A. YARMAN (SaSaD)

Yürütme Kurulu

Doç. Dr. Veysi İŞLER (Konferans Başkanı, ODTÜ-TSK MODSİMMER)
Prof. Dr. Semih BİLGEN (Program Kurulu Başkanı, ODTÜ)
Y. Müh. Alb. Bekir KAZANDIR (KHO, Sav. Bil. Ens.)
Yrd. Doç. Dr. Öğ. Alb. Türkay KISA (KHO, Sav. Bil. Ens.)
Hv. Mu. Ütğm. Ömer ÇETİN (HHO, Hvc. ve Uzy. Tekno. Ens.)
Hv. Müh. Alb. Ali ŞİMŞEK (MSB ARGE ve Tekno. D. Bşk.lığı)
Mu. (OBİ) Yb. Umut AKIN (Gnkur. BİLKARDEM Bşk.lığı)
Mu. (OBİ) Bnb. Hüseyin SAYIN (Gnkur. BİLKARDEM Bşk.lığı)
Uzm. C. Cumhur KIYKIM (MSB SSM.lığı)
Uzm. Serpil ÖZCAN (MSB SSM.lığı)
Prof. Dr. Faruk POLAT (ODTÜ)
Prof. Dr. Levent KANDİLLER (Çankaya Ü.)
Dr. Aysu Betin CAN (ODTÜ)
Y.Müh. Utkan ERYILMAZ (ODTÜ-TSK MODSİMMER)
Y.Müh. Vedat KARAARSLAN (Konferans Genel Sekreteri, ODTÜ-TSK MODSİMMER)

Program Komitesi

M.Ali AKÇAYOL (Gazi Ü.)
Sevgi AKGÜN (TUBİTAK MAM)
Turgut AKYÜREK (MSB Tek. Hiz. D.)
Nafiz ALEMDAROĞLU (ODTÜ)
Ferda Nur ALPASLAN (ODTÜ)
Taner ALTINOK (KHO Sav. Bil. Ens.)
Fulya ALTIPARMAK (Gazi Ü.)
Ömer ANLAĞAN (TÜBİTAK)
A.Turan ARAL (Atılım Ü.)
Ahmet ARSLAN (Selçuk Ü.)
Harun ARTUNER (Hacettepe Ü.)
Birce BAKIRLI (Hv.K.K.lığı)
Fatih Fikret BAL (KHO)
Selim Saffet BALCISOY (Sabancı Ü.)
Nejat BASIM (KHO Sav. Bil. Ens.)
Hakan BAŞARAN (ARMERKOM)
Semih BİLGEN (ODTÜ)
Müslim BOZYİĞİT (ODTÜ)
Aysu Betin CAN (ODTÜ)
Tolga CAN (ODTÜ)
Kürşat ÇAĞILTAY (ODTÜ)
Tolga ÇAPIN (Bilkent Ü.)
Çiğdem DEMİR (Bilkent Ü.)
Onur DEMİRÖRS (ODTÜ)
Berna DENGİZ (Başkent Ü.)
Oğuz DİKENELLİ (EGE Ü.)
Ali DOĞRU (ODTÜ)
Şebnem DÜZGÜN (ODTÜ)
Çetin ELMAS (Gazi Ü.)
Serpil EROL (Gazi Ü.)
Filiz ERSÖZ (KHO Sav. Bil. Ens.)
Akif ESENDEMİR (ODTÜ)
Mustafa İlhan GÖKLER (ODTÜ)
Muhittin GÖKMEN (İTÜ) MODSİMLAB)
Uğur GÜDÜKBAY (Bilkent Ü.)
Kadir GÜNDOĞAN (KHO Sav. Bil. Ens.)
Nusret HATIRNAZ (KHO Sav. Bil. Ens.)
Altunkan HIZAL (ODTÜ)
Kayhan İMRE (Hacettepe Ü.)
İlker İPEKÇİ (Gnkur. BİLKARDEM Bşklığı)
Veysi İŞLER (ODTÜ-TSK MODSİMMER)
Bilgin KAFTANOĞLU (ODTÜ)
Levent KANDİLLER (Çankaya Ü.)
T.Yaşar KATIRCIOĞLU (MSB SATEM)
Bekir KAZANDIR (KHO Sav. Bil. Ens)
Ömer KIRCA (ODTÜ)
Türkay KISA (KHO Sav. Bil. Ens.)
Sinan KAYALIGİL (ODTÜ)
Nevzat KILINÇ (MSB Tek.Hiz. D.)
Sencer KOÇ (ODTÜ)
Yaşar KÖSE (KHO)
Kemal LEBLEBİCİOĞLU (ODTÜ)
Halit OĞUZTÜZÜN (ODTÜ)
Bülent ÖRENCİK (TUBİTAK MAM)
Nur Evin ÖZDEMİREL (ODTÜ)
Altan ÖZKİL (Gnkur. BİLKARDEM Bşklığı)
Ercan ÖZTEMEL (Marmara Ü.)
M.Orhan ÖZYALÇIN
(Gnkur. BİLKARDEM Bşklığı)
Faruk POLAT (ODTÜ)
İhsan SABUNCUOĞLU (Bilkent Ü.)
Yekta SOYLU (KHO)
Fatih Erdoğan SEVİLGEN (GYTE)
Cevat ŞENER (ODTÜ)
Gökhun TANYER
(TUBİTAK UEKAE İLTAREN)
Cengiz TAVUKÇUOĞLU
(KHO Sav. Bil. Ens.)
Önder Haluk TEKBAŞ (KHO)
Tuğba TEMİZEL (ODTÜ)
Mehmet TOLUN (Çankaya Ü.)
İ. Hakkı TOROSLU (ODTÜ)
Erdal TORUN (MSB ARGE ve Tekno. D.)
Ayhan Özgür TOY (Dz.K.K.lığı)
Göktürk ÜÇOLUK (ODTÜ)
Elife ÜNAL (SSM)
Samim ÜNLÜSOY (ODTÜ)
Kadir VAROĞLU (Başkent Ü.)
Fatoş Yarman VURAL (ODTÜ)
Yasemin YARDIMCI (ODTÜ)
Faruk A.YARMAN (SaSaD)
Adnan YAZICI (ODTÜ)
Ali YAZICI (TOBB ETÜ)
Orhan YILDIRIM (ODTÜ)

ANA SPONSOR

HAVELSAN

SPONSORLAR

aselsan

MEKSAN SİSTEM

MilSOFT

OYTEK

STM

TAI

DESTEK VEREN KURULUŞLAR

SaSaD

SAVUNMA SANAYİİ İMALATÇILAR DERNEĞİ
DEFENSE INDUSTRY MANUFACTURERS ASSOCIATION

EMEĞİ GEÇENLERE TEŞEKKÜR

ODTÜ-TSK MODSİMMER çalışanları

Uğur KESKİN

Elmas BAYBÖRÜ

Selma GÖMDENİZ

Fırat BELGE

ve stajyerler

Sebahattin DÜZEL

Mustafa Mahir KAPLANCI

Konferansın organizasyonunda gösterdiğiniz özverili çalışmalarınız için teşekkür ederim.

Doç.Dr. Veysi İŞLER
ODTÜ-TSK MODSİMMER Başkanı
USMOS 2007 Konferans Başkanı

İÇİNDEKİLER

ÖNSÖZ ..... **III**

DÜZENLEYEN KURULUŞLAR ..... **V**

KONFERANS KURULLARI ..... **VII**

ANA SPONSOR ..... **XI**

SPONSORLAR ..... **XI**

DESTEK VEREN KURULUŞLAR ..... **XI**

EMEĞİ GEÇENLERE TEŞEKKÜR ..... **XII**

İÇİNDEKİLER ..... **XIII**

BİLDİRİLER **1**

DAĞITIK SİMÜLASYON UYGULAMALARI **3**

SAGES: SANAL GERÇEKLİK ORTAMINDA HLA UYUMLU VE STEREO GÖRÜNTÜ, FİZİK VE ÇEVRESEL SES DESTEKLİ SAVAŞMA YETENEĞİ EĞİTİM SİMÜLATÖRÜ ... 5

SU ÜSTÜ PLATFORMLARININ MODELLENMESİ VE TİCARİ BİR SENTETİK HEDEF SİMÜLATÖRÜ İLE ENTEGRELİ SİMÜLASYONU ... 15

ÇEVİK ENGEL MÜFREZE SAVUNMASI VE TANK TAKIMI TAARRUZUNUN DAĞITIK ETKİLEŞİMLİ SİMÜLASYONU ... 27

BÜYÜK ÖLÇEKLİ AĞLARIN DEVS TABANLI MODELLENMESİ VE SİMÜLASYONU 37

FİZİK TABANLI MODSİM-I **49**

3-BOYUTLU PALETLİ ARAÇ DİNAMİĞİ SİMÜLASYONU ... 51

BALİSTİK BİR FÜZENİN MODELLENMESİ VE OTOPİLOT TASARIMI ... 61

FST: HİDROFONUN YER DEĞİŞTİRMESİNİN HESAPLANMASI İÇİN BENZETİM TABANLI SONOBOY MEKANİK TASARIM ARACI ... 71

UÇUŞ KONTROL SİSTEMİ TASARIMI VE GELİŞTİRİLMESİ İÇİN SİMÜLASYON TABANLI TEST PLATFORMU YAZILIMI F-SIM ... 82

BİRLİKTE ÇALIŞABİLİRLİK **89**

C4I İLE M&S SİSTEMLERİNİN BİRLİKTE ÇALIŞABİLİRLİĞİNE MUHAREBE YÖNETİM DİLİ YAKLAŞIMI ... 91

DÖNGÜDE DONANIMSAL BENZETİM TEKNİĞİ VE BİLEŞENLERİ ... 101

MİLSOFT DDS ARAKATMANI VE DDS'İN SAVAŞ YÖNETİM SİSTEMLERİNDE SİMÜLASYON AMAÇLI KULLANIMI ... 109

FARKLI SİMÜLASYON SİSTEMLERİNİN ENTEGRASYONU İÇİN KONTROL MİMARİSİ ... 119

FİZİK TABANLI MODSİM-II **129**

SU ÜSTÜ GEMİLERİNİN AKUSTİK İZLERİNİN ZAMAN UZAYINDA MODELLENMESİ VE SİMÜLASYONU ... 131

NORMAL MOD YÖNTEMİ KULLANILARAK SU ALTI PASİF AKUSTİK DALGA YAYILIMININ MODELLENMESİ VE SİMÜLASYONU ... 138

RADAR ARAYICI BAŞLIKLAR VE RADAR GÜDÜMÜNÜN MODELLENMESİ VE SİMULASYONU ... 145

RADAR KESİT ALANI (RKA) MODELLEME VE SİMÜLASYONU ... 156

TEMSİLİ ORTAMLAR VE GÖRSELLEŞTİRME ... 167

MÜHATEM: MÜŞTEREK HAREKAT ALANI SİMÜLASYON SİSTEMİ: TEMSİLİ ORTAM STANDART VE UYGULAMALARI ... 169

MÜHATEM YAZILIM PROGRAMLAMA KÜTÜPHANESİ ... 180

YÜKSEK ÇÖZÜNÜRLÜKLÜ ARAZİ VERİLERİNİN GRAFİK İŞLEMCİSİ KULLANILARAK GÖRSELLEŞTİRİLMESİ ... 191

SAVUNMA SİSTEMLERİ TASARIMINDA İNSAN BİLGİSAYAR ETKİLEŞİMİ ÇALIŞMALARININ ROLÜ ... 200

EĞİTİM SİMÜLATÖRLERİ VE MODSİM SİSTEMLERİ TEDARİĞİ ... 209

MODELLEME, SİMÜLASYON (MODSIM), VE ARGE TABANLI SİSTEM TASARIMI, TEDARİĞİ VE KURULUMU ... 211

EĞİTİM AMAÇLI SİMÜLATÖRLERDE MALİYET-ETKİNLİK ANALİZİ: HAVAN EĞİTİM SİMÜLATÖRÜ ÖRNEĞİ VE BİR MODEL ÖNERİSİ ... 221

SANAL BAKIM EĞİTİM SİSTEMLERİ ... 231

MODEL TABANLI TASARIM SÜREÇLERİNDE SİMÜLASYON VE ÜRÜN GELİŞTİRMEDE ALT YÜKLENİCİLİK SİSTEMİNİN UYGULAMASI ... 241

HLA ALTYAPILARI-I ... 251

HLA TABANLI DAĞITIK SİMÜLASYONLARDA FEDERE YÖNETİM ALTYAPISI ... 253

FEDERASYONLAR İÇERİSİNDE DİNAMİK FEDERE KÜMELENMELERİ VE ZAMAN YÖNETİMİ ... 261

SİMULASYON MODELLEME ALTYAPISI (SİMA) : DEVS TABANLI HİYERARŞİK VE MODÜLER BİR MODELLEME VE KOŞUM ALTYAPISI ... 271

ÖLÇEKLENDİRİLEBİLİR HLA UYUMLU SİMÜLASYONLAR İÇİN UYGULAMA ÇATISI: SİMALT ... 283

KARAR DESTEK AMAÇLI MODSİM-I ... 293

ASKERİ BİRLİKLERİN İNTİKAL PLANLAMASI İÇİN BİR SİMÜLASYON MODELİ ... 295

MUHAREBEDE MÜHİMMAT DAĞITIM SİSTEMİNİN MODELLENMESİ VE OPTİMİZASYONU ... 306

ARAÇ ROTALAMA PROBLEMLERİNİN ÇÖZÜMÜNE BİR YAKLAŞIM ... 316

TAKTİKSEL EN KISA YOL PROBLEMLERİNİN GENETİK ALGORİTMA YAKLAŞIMI İLE ÇÖZÜMÜ ... 326

HLA ALTYAPILARI-II ... 337

KABAN : SİMULASYON VERİLERİ GEÇERLEME VE OTOMATİK KULLANICI ARABİRİMİ OLUŞTURMA ALTYAPISI ... 339

FEDERASYON BAŞARIM TEST ORTAMI ... 350

DAĞITIK BENZETİM SİSTEMLERİ İÇİN PERFORMANS ANALİZİ YÖNTEMİNİN GELİŞTİRİLMESİ ... 360

DAĞITIK BENZETİM UYGULAMALARINDA YENİDEN KULLANILABİLİRLİK, VERİ DAĞITIMI VE ZAMAN YÖNETİMİ ... 370

KARAR DESTEK AMAÇLI MODSİM-II 381

İSTANBUL BOĞAZI DENİZ GEMİ GEÇİŞ TRAFİĞİNİN BENZETİM MODELLEME VE SENARYO ANALİZİ İLE İNCELENMESİ 383

SİSTEM DİNAMİKLERİ YAKLAŞIMLI TERFİ SİSTEMİ ANALİZ MODELİ (TESAMOS) 394

NÜKLEER, BİYOLOJİK VE KİMYASAL (NBC) TAARRUZ SİMÜLASYON MODELİ .. 406

BENZETİM PROJELERİNDE BİLİMSEL VE TEKNİK YORDAM GELİŞTİRME PRATİĞİNE BİR YAKLAŞIM 417

KAVRAMSAL MODELLEME VE SİMÜLASYON ALTYAPILARI 427

SİMÜLASYON GELİŞTİRME SÜRECİ İÇİN ÇOK KULLANICILI KAVRAMSAL MODELLEME ARACI 429

TOPÇU BATARYASI İÇİN KAVRAMSAL MODELLEME VE UYGULAMALARI 437

DAĞITIK SİMULASYON PROJELERİNİN KAVRAMSAL MODELİNİ GELİŞTİRME ÇALIŞMALARINDA BOM KULLANIMI 447

CİDDİ OYUN VE SİMÜLASYON ALTYAPISI 457

KARAR DESTEK AMAÇLI MODSİM-III 467

TEHDİT DEĞERLENDİRME VE SİLAH TAHSİSİ – BİR UYGULAMA 469

DEĞİŞKEN VURMA OLASILIKLI HAVA SAVUNMA SİLAHLARININ GRUP HALİNDE HAVA SAVUNMA STRATEJİSİNİN PLANLANMASI VE OPTİMİZASYONU 476

ATIŞ KONTROL SİSTEMİ HATA ANALİZİ İLE TEK ATIŞTA VURUŞ İHTİMALİNİN HESAPLANMASI 488

YAZAR DİZİNİ 501

BİLDİRİLER

A1 OTURUMU:

DAĞITIK SİMÜLASYON UYGULAMALARI

SAGES: SANAL GERÇEKLİK ORTAMINDA HLA UYUMLU VE STEREO GÖRÜNTÜ, FİZİK VE ÇEVRESEL SES DESTEKLİ SAVAŞMA YETENEĞİ EĞİTİM SİMÜLATÖRÜ
Selim TEMİZER, Haluk ELİŞ, Namık Kemal TEMİZER

SU ÜSTÜ PLATFORMLARININ MODELLENMESİ VE VR FORCES BACK END ENTEGRELİ SİMÜLASYONU
Mehmet HAKLIDIR, Deniz ALDOĞAN, İsa TAŞDELEN

ÇEVİK ENGEL MÜFREZE SAVUNMASI VE TANK TAKIMI TAARRUZUNUN DAĞITIK ETKİLEŞİMLİ SİMÜLASYONU
Baki GÜNDÜZ, İsmet ÇUHADAR, Halit OĞUZTÜZÜN

BÜYÜK ÖLÇEKLİ AĞLARIN DEVS TABANLI MODELLENMESİ VE SİMÜLASYONU
Ahmet ZENGİN, Hüseyin EKİZ

SAGES: SANAL GERÇEKLİK ORTAMINDA HLA UYUMLU VE STEREO GÖRÜNTÜ, FİZİK VE ÇEVRESEL SES DESTEKLİ SAVAŞMA YETENEĞİ EĞİTİM SİMÜLATÖRÜ[1]

Selim TEMİZER [(a)], Haluk ELİŞ [(b)], Namık Kemal TEMİZER [(c)]

[(a)] Massachusetts Institute of Technology, Cambridge, MA, temizer@alum.mit.edu

[(b)] Genelkurmay BİLKARDEM Başkanlığı, Ankara, halukelis@gmail.com

[(c)] North Carolina State University, Raleigh, NC, nktemize@ncsu.edu

ÖZ

Bu bildirimizde, geliştirmiş olduğumuz Sanal Gerçeklik Ortamında Savaşma Yeteneği Eğitim Simülatörü (SAGES) hakkında bilgi verip, SAGES sisteminin atıcılık, nişancılık, silah ve çatışma eğitimi simülasyonları alanında tamamen yeni olan özelliklerini ortaya koyacağız. SAGES sistemi iki temel bileşenden oluşmaktadır: Bunlardan ilki, Modelleme ve Simülasyon Sistemleri (M&S) alanındaki askeri gereksinimler ve standartların, sanal gerçeklik ve çoklu ortam teknolojilerinin ve ihtiyaç duyulan donanım bileşenlerinin çok kapsamlı bir eğitim simülatörü oluşturmak amacıyla bir araya getirilmesi, öğrenim amaçlı teknoloji testlerinin yapılarak gerekli bilgi birikiminin sağlanması ve bu doğrultuda bir yazılım ve donanım altyapısı oluşturulmasıdır. İkinci temel bileşen ise, oluşturulmuş bu altyapı üzerinde, kazanmış olduğumuz bilgi ve yeteneklerin ortaya konulması ve demonstrasyonu amacıyla, prototip olarak hazırlanmış olan ve bir sanal gerçeklik ortamı oluşturarak bu ortamda G-3 piyade tüfeği ile atış eğitimi sunan bir simülasyon sistemidir. Bu bildirimizde SAGES sisteminde kullanmış olduğumuz teknolojilere ek olarak gelecekte SAGES'e eklemeyi planladığımız yetenekleri, SAGES'in mevcut sistemlere göre üstünlüklerini, sağlayabileceği farklı kullanım alanlarını ve böyle bir simülasyon sistemi ile eğitim gören kullanıcıların elde edeceği avantajları da sunacağız.

Anahtar Kelimeler: Atış Eğitimi, Çatışma Simülasyonu, Eğitim Simülasyonu, HLA, Keskin Nişancı Eğitimi, Nişancı Eğitimi, Personel Seçimi, Sanal Gerçeklik.

[1] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

SAGES: HLA-COMPLIANT COMBAT CAPABILITY TRAINING SIMULATOR IN VIRTUAL REALITY ENVIRONMENT WITH STEREO VISION, RIGID BODY DYNAMICS AND ENVIRONMENTAL AUDIO SUPPORT

ABSTRACT

In this document, we present a simulation system, The Combat Capability Training Simulator In Virtual Reality Environment (SAGES), that we have developed, and we discuss its unique features in the domain of shooting, marksmanship, gun and combat training simulations. The SAGES System is comprised of two major components: The first component entails gathering together Modeling and Simulation (M&S) related military requirements and standards, virtual reality and multimedia technologies, and required hardware components in order to create a comprehensive training simulator, developing know-how and a knowledge base through technology tests and forming a software and hardware infrastructure. The second component is a virtual reality G-3 assault rifle shooter training simulator that is developed as a prototype to serve as a proof-of-concept and to demonstrate the knowledge and skills acquired. We also present the future add-ons and capabilities planned for the SAGES system, superior qualities of the SAGES over existing simulators, new usage areas that SAGES can serve, and the advantages that SAGES offers to its trainees.

Keywords: Combat/Skirmish Simulation, HLA, Marksman Training, Personnel Selection, Sharp Shooter Training, Shooting Training, Training Simulation, Virtual Reality.

1. GİRİŞ

Değişen tehdit algılamaları ve ülkemizin içinde bulunduğu coğrafya dikkate alındığında askeri bakımından her zamankinden daha güçlü ve caydırıcı olmamız gerekmektedir. Kaynakların kısıtlı olması, makro ekonomik göstergeler ve politikalar gereği savunma harcamalarına ayrılan bütçede kısıtlamalara gidilmesi, tüm kamu kurumlarını olduğu gibi silahlı kuvvetlerimizi de kaynaklarını daha verimli kullanmaya itmektedir.

Ülke güvenliğinde görev alan personelimizin sahip oldukları silah sistemlerini daha iyi kullanması, örneğin temel silahlarımızdan olan G-3 piyade tüfeği nişancılığında daha iyi yetişmesi, personelde kendine güven duygusunun artmasına ve sonuç olarak daha etkin vuruş gücüne sahip birliklerin oluşmasına katkıda bulunur.

Bir erin alması gereken atış ve muharebe eğitimi, askerliği süresince belirli bir sayıda mermi atmasını öngörmektedir. Mermi sayısının artırılması kuşkusuz askerimizin eğitimine olumlu yönde katkıda bulunacaktır. Ancak, artan mermi sayısının eğitim maliyetini büyük ölçüde artıracağı da bir gerçektir. Bir G-3 piyade tüfeği mermisinin maliyetinin 2004 yılı itibari ile 0,71 Amerikan Doları (yaklaşık 1 YTL, kaynak: Makine Kimya Enstitüsü) olduğu düşünüldüğünde ve bir er için 10 mermilik bir artırıma gidileceği varsayılırsa, bu durum 10.000 eri olan bir askeri birlik için 100.000 YTL'lik

ek bir maliyet getirecektir. 10 mermi düşük bir sayı olmasına rağmen 100.000 YTL'nin alternatif maliyeti daha farklı bir silah sistemine yatırım olarak kullanılabilir.

Bu bildiride, güvenlikten sorumlu personelin eğitimine katkıda bulunmak ve daha az maliyetle daha iyi eğitimli personel yetiştirebilmek amacıyla geliştirdiğimiz ***Sanal Gerçeklik Ortamında HLA Uyumlu ve Stereo Görüntü, Fizik ve Çevresel Ses Destekli Savaşma Yeteneği Eğitim Simülatörü (SAGES)*** sisteminin ve sistemin yeteneklerini sergilemek amacıyla bir prototip olarak sistem üstüne inşa ettiğimiz G-3 piyade tüfeği atış simülatörünün eğitim simülatörleri alanında tamamen yeni olan özelliklerini ortaya koyup, başta Türk Silahlı Kuvvetleri olmak üzere Emniyet Teşkilatı ve diğer güvenlik kurumları için kullanım alanlarını ve avantajlarını belirteceğiz.

SAGES'in ilk testlerinden olumlu sonuçlar alınmış olup, müteakiben ***güvenilirlik*** ve ***geçerlilik*** testlerine tâbi tutulması düşünülmektedir [1]. Güvenilirlik, SAGES'i kullanarak atış yapan bir personelin atış becerisinde herhangi bir değişiklik olmadan farklı zamanlarda yapacağı atışlardan yakın sonuçlar alması, geçerlilik ise SAGES ile iyi atış yapan bir personelin gerçek hava ve arazi şartlarında gerçek silah sistemiyle yapacağı atışlarda da simülatörde gösterdiğiye başarıya yakın başarı göstermesidir.

Simülatörlerin kullanım amacı, simülatörlerde eğitilen kişilerin çalışma ortamlarında karşılaşacakları gerçek şart ve durumların bilgisayar ve elektronik ortama gerçeğe yakın seviyede aktarılmasıdır [2]. Sanal gerçeklik destekli bir simülatör olan SAGES ise her türlü durumu sanal ortamda yaratabilmektedir. SAGES'te eğitim gören personel riskli bir hareket tarzını zarar görmeden deneme şansına sahip olacaktır.

Yazılım, donanım ve sistem mühendisliği alanlarında çok geniş bir bilgi birikimi ve tecrübenin sonucu olarak ortaya çıkmış olan SAGES sisteminin yeteneklerini somut biçimde ortaya koymak amacıyla hazırladığımız prototip simülatör, G-3 piyade tüfeğinin yatarak destekli 25, 50, 100 ve 200 metre atışlarını simüle etmek için tasarlanmıştır. Ancak SAGES ile çok çeşitli silah sistemlerini ihtiyaç duyulan mesafelerde ve istenilen şekillerde (ayakta, yatarak, vs.) kullanmak mümkündür. Ayrıca SAGES, sanal ortamda her türlü çatışma ve savaş durumunu yaratabilmektedir. Bu sebeple SAGES sistemini sadece bir atıcılık simülasyonu, nişancılık simülasyonu, çatışma simülasyonu veya bir silah eğitim simülasyonu olarak nitelemek yerine ***savaşma yeteneği eğitim simülatörü*** olarak isimlendirmekteyiz.

2. MEVCUT ÇATIŞMA SİMÜLATÖRLERİNİN ÖZELLİKLERİ

Günümüzde gerek ülkemizde gerekse diğer ülkelerde yaygın olarak kullanılan askeri amaçlı geniş çaplı çatışma simülatörleri temel olarak göze zarar vermeyen lazerli atış yapma ve yapılan atışı algılama prensibine dayanan sistemler kullanmaktadırlar. Bu sistemlerin en yaygın olanlarından birisi Çoklu Entegre Lazer Nişan Alma Sistemi (Multiple Integrated Laser Engagement System, MILES) adı verilen sistemdir [3]. MILES Sistemi kullanılarak icra edilen bir çatışma simülasyonunda gerçek silahlar, gerçek araçlar ve gerçek personel kullanılır. Çatışma öncesi her silahın üzerine birer lazer gönderici takılarak sıfırlaması yapılır. Ayrıca bütün personel ve araçların da üzerlerine lazer algılayıcılar bağlanır. Daha sonra çatışma eğitimi, seçilen gerçek bir arazi parçası üzerinde canlı olarak icra edilir, fakat gerçek mermiler yerine MILES sisteminin sağladığı ***lazer mermiler*** kullanılır. Lazer algılayıcılar, üzerlerine herhangi

bir silahtan atılan lazer mermiler isabet etmişse çeşitli sesler çıkarmak suretiyle hedefin vurulduğunu rapor ederler.

Zaman içinde MILES sistemi, artan ihtiyaçlara cevap verememeye başlayınca, ABD ordusu tarafından MILES 2000 adlı daha modern bir sistem geliştirilmiştir [4]. MILES 2000 sistemindeki yeni özellikler arasında Eğitim Sonrası Değerlendirme (After Action Review) yapabilmek ve eğitilen personele detaylı geribildirim sunabilmek amacıyla çatışma verilerini kaydedebilme ve raporlama yeteneği de bulunmaktadır.

Çatışma simülasyonlarında olduğu gibi tabanca/tüfek atışı ve keskin nişancılık eğitimi için kullanılan simülasyon sistemlerinde de tercih edilen temel unsur, güvenilir ve kolay bir çözüm sunması sebebiyle lazer gönderici ve algılayıcı teknolojisidir.

Mermilerin doğrusal bir yol takip etmediği (havan topu benzeri) silah sistemlerinin çok sınırlı bir kısmı simülasyonlarda yer alabilmektedir. Örneğin düşük enerjili radyo frekans (radio frequency, RF) sinyalleri yayan cihazlar, el bombası ve kara mayını tarzı mühimmatın patlama yarıçapını gerçeğe yakın bir şekilde simüle edebilmektedir.

3. MEVCUT MODELLEME VE SİMÜLASYON TEKNOLOJİLERİ

Modelleme ve Simülasyon (M&S) Sistemlerinin günümüzde ulaştığı nokta ve gelecekte M&S alanında kullanılması muhtemel olan teknolojiler hakkında çok kapsamlı bir araştırmanın sonuçlarını yakın bir geçmişte yayınlamış bulunuyoruz [5]. Bu bölümde bir atış/çatışma eğitim simülatörü oluşturmak amacıyla kullanılabilecek yazılım ve donanım teknolojilerinden ve gerek ticari gerekse ücretsiz (açık kaynak, open source) yazılım paketlerinden kısaca bahsedeceğiz.

İşletim sistemi: Eğer geliştirilecek olan simülasyon sisteminde çok özel gerçek-zamanlı performans gereksinimleri bulunuyorsa, INTEGRITY® (Green Hills® Software), VxWorks (Wind River) veya Real-Time Linux gibi bir Gerçek-Zamanlı İşletim Sistemi (Real-Time Operating System, RTOS) kullanılabilir.

Yazılım altyapısı: Matematiksel hesaplamalar için Template Numerical Toolkit (TNT), grafik motorunun sunduğu görüntüleri gerçek zaman ile senkronize etmek için Open Producer ve paylaşılan kaynaklara ulaşımı düzenlemek ve multi-thread desteği sağlamak için Open Threads benzeri kütüphaneler kullanılabilir.

Grafik, animasyon ve ses: Grafik kütüphanesi olarak OpenGL, DirectX, Irrlicht, OpenSceneGraph veya CrystalSpace, animasyon kütüphanesi olarak CAL3D veya Pixar RenderMan® yazılımı, animasyon destekli model formatları olarak OpenFlight veya ReplicantBody ve ses kütüphanesi olarak DirectSound veya OpenAL benzeri kütüphaneler kullanılabilir.

Fiziksel hesaplamalar: Katı-cisim dinamiği (rigid-body dynamics) ve çarpışma hesaplamaları için Havok Physics™, Open Dynamics Engine (ODE), OPCODE, ColDet kütüphaneleri ve/veya bir donanım (PCI kartı) olarak geliştirilmiş olan AGEIA PhysX™ işlemcisi kullanılabilir.

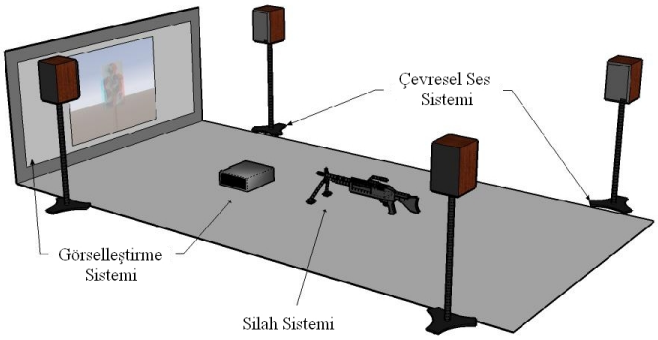
Ağ üzerinden bilgi paylaşımı: Askeri yazılımlarda özellikle aranılan İleri Seviye Mimari (High Level Architecture, HLA) uyumluluğu veya bunun dışındaki Game Networking Engine, HawkNL, Adaptive Communication Environment (ACE) ve BSD soketleri gibi yazılımlar kullanılabilir.

Donanım: Gerçek mermilerin yerine MILES™ benzeri bir lazer sistemleri ve SAGES simülatöründe olduğu gibi silah sistemlerinin pozisyon ve yönelim bilgilerini tespit edip sayısal ortama aktarabilmek için açı ve hız kodlayıcıları (rotary encoder) kullanılabilir.

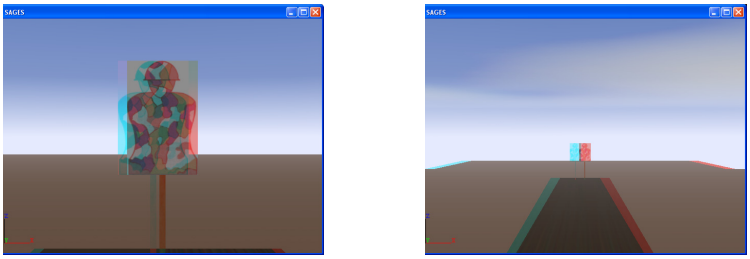
4. SAGES SİSTEMİ

Bu bölümde geliştirdiğimiz prototip simülatör sisteminin nasıl çalıştığı hakkında bilgi verip SAGES sisteminin yazılım ve donanım bileşenlerini hem oluşturduğumuz bilgi birikimi açısından hem de geliştirdiğimiz prototipin özellikleri açısından inceleyeceğiz.

SAGES eğitim simülatörünü kullanan her bir personelin kendi platformu bulunmaktadır ve örnek bir platform Şekil 1'de görülmektedir. Her platformda bir görselleştirme ve ses sistemi ve bir de silah sistemi yer almaktadır. Şekil 2'de ise SAGES simülatörünü kullanan bir personelin karşısına çıkacak olan stereo görüntülerden (anaglyph) örnekler yer almaktadır. Bu resimlere (renkli bir baskıda) kırmızı-mavi anaglyph gözlüklerle bakıldığında sahnenin derinliği algılanabilmektedir.



Şekil 1. Tek Bir Personelin Kullanacağı SAGES Platformu



Şekil 2. Anaglyph Sahne Örnekleri

SAGES'te görselleştirme sistemi olarak bir bilgisayar monitörü, projektörle duvara veya perdeye yansıtmalı bir sistem veya başa takılan (ve 3 boyutlu stereo görüntü sunan) bir görselleştirme sistemi (head-mounted display, HMD) kullanılabilir. Ses sistemi bir kulaklık veya 3 boyutlu çevresel ses sağlayan hoparlör gruplarından oluşabilir. Görüntüleme ve ses sistemi eğitilen personele 3 boyutlu bir sanal gerçeklik ortamı sağlar. Başa takılan görselleştirme sistemlerinde sanal gerçeklik ortamı otomatik olarak oluşur. Diğer sistemlerde ise eğitim gören personel kırmızı-mavi (anaglyph) , polarize (polarizing glass) veya LCD shutter glass gözlükler takarak sanal gerçeklik ortamınındaki görsel unsurları (derinlik, mesafe ve cisimlerin boyutlarını) algılar.

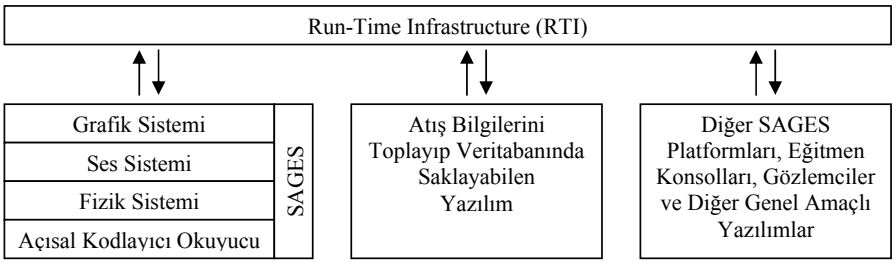
Silah sistemi ise personelin kullanacağı silahı ve silahın ağırlığı, geri tepmesi, vb. özelliklerini temsil eden fiziksel bir model ve silahın pozisyonunu, yönelim bilgilerini ve (silahı ateşlemeye hazır hale getirme, tetiğe basma, vb.) kullanım bilgilerini tespit edip sayısal ortama aktarabilen bir mekanizmadan oluşur.

Eğitim gören personel kendi platformunu bireysel bir çatışma veya nişan alma eğitimi görmek amacıyla kullanabilmektedir. Ayrıca yan yana veya farklı fiziksel mekanlarda bulunan platformlar, ***ağ bağlantısı (RTI)*** yardımıyla, eğitim gören birden fazla personeli ve denetleyici eğitmenleri aynı sanal gerçeklik ortamında da buluşturabilmektedir.

4.1. Yazılım

SAGES sisteminde sanal gerçeklik ortamı oluşturabilmek için 3 boyutlu stereo görüntü sunan grafik yazılımı, fiziksel olayların simülasyon ortamında canlandırılabilmesi için fizik yazılımı ve çatışma ortamındaki sesleri gerçekçi biçimde kullanıcıya sunabilmek için de 3 boyutlu çevresel ses yazılımı mevcuttur. Grafik yazılımı, oluşturduğu stereo görüntüleri çeşitli görselleştirme cihazlarına uygun formatlarda sunabilmektedir: Örneğin başa takılan ve her göz için ayrı ekrana sahip olan sistemlere sağ ve sol göz görüntüsü olarak ayrı ayrı iletebilmekte, monitör veya projektörlü bir sistem için sağ ve sol göz görüntülerini (daha sonradan shutter glass veya anaglyph gözlükler kullanılarak çözülmek üzere) üst üste bindirebilmekte veya ayna benzeri bir düzenek sayesinde (ya da gözleri çaprazlama yoluyla) sağ ve sol göze ayrı ayrı iletilmesi gereken resim çiftleri olarak oluşturabilmektedir. Fizik yazılımı, katı cisim dinamiklerini temsili ortamda hesaplayarak yerçekimi kuvvetini, mermi ağırlığını, ve sanal ortamdaki diğer cisimlerin kütlelerini, eylemsizlik momentlerini ve üzerlerine etkiyen kuvvetleri simüle edebilmektedir. Ses yazılımı ise sanal ortamdaki ses kaynaklarından çıkan sesleri sanal ortamdaki pozisyonlarına, yönelimlerine ve hızlarına uygun olarak eğitim gören personele sunabilmektedir. Böylece sanal ortam birebir gerçekmiş gibi algılanmaktadır. Birçok modülden oluşan yazılımın mimarisi için Temizer Sistemi [6] kullanılmaktadır.

Sanal gerçeklik ortamını oluşturmaya yarayan bu temel yazılım kütüphanelerinin yanısıra sistem HLA uyumlu olup ağ üzerinden sanal gerçeklik ortamıyla ilgili verileri diğer yazılımlarla paylaşabilmektedir. Oluşturduğumuz prototip simülatörümüz bu özellikten faydalanarak her atış sonrası sanal gerçeklik ortamında merminin isabet ettiği noktanın koordinatlarını Run-Time Infrastructure (RTI) vasıtasıyla bu bilgiyle ilgilenen diğer yazılımlara göndermektedir. Yine prototip sistem için hazırladığımız bir başka HLA uyumlu program ise bu bilgileri toplayıp, isteğe göre bir veritabanında biriktirebilmektedir. SAGES sisteminin federasyon yapısı Şekil 3'te görülmektedir.



Şekil 3. SAGES Sistemi Federasyon Yapısı

4.2. Donanım

SAGES simülatörünün merkezi olan açısal kodlayıcı mekanizmasını kurmak için günümüzde kablolu ve kablosuz olmak üzere birçok teknoloji mevcuttur: Örnek olarak özel kızılötesi kameralar aracılığıyla üzerlerine çeşitli yansıtıcı parçalar takılmış cisimlerin konumları bilgisayarlar tarafından takip edilebilmekte (infrared tracking systems) ve optoelektronik gönderici ve alıcı sistemler sayesinde yine benzer şekilde gerçek ortamdaki cisimlerin konum ve yönelim bilgileri sayısal ortama aktarılabilmektedir. Kablosuz teknolojilerin yanısıra bir eksen etrafında dönebilen açı ve hız kodlayıcı donanımlar (rotary encoder) da bir eksene bağlı bulundurulan bir silah sistemi hakkında yönelim bilgilerini tespit edebilmektedir.

Geliştirdiğimiz SAGES prototip simülatörde kullandığımız donanım, kablolu bir sistem olup açısal kodlayıcı (rotary encoder) kullanarak çalışmaktadır. Piyasada fiyatları 2 ile 250 Amerikan Doları arasında değişen ve devir başına 3 ile 10.000 arasında sinyal (pulses per revolution, ppr) gönderebilme kapasitesine sahip çeşitli açısal kodlayıcılar mevcuttur. Hatta fotoğrafçılık gibi özel kullanım alanları için 100.000 ve daha yüksek ppr hassasiyetine sahip açısal kodlayıcılar da bulunmaktadır. Geliştirmiş olduğumuz prototipte maliyeti 5 Amerikan Dolarının altında olan analog bir PC Joystick içinde bulunan 2 adet potansiyometreyi (200KΩ) ve diğer devre elemanlarını, geliştirdiğimiz bir yazılım ile birlikte çalıştırmak suretiyle 10.000 ile 50.000 ppr arasındaki hassasiyetlerde kullanabilmekteyiz. Bu hassaslığı elde edebilmek için potansiyometrelerden gelen sinyal üzerinde çoklu örnekleme ve gürültü azaltma (noise reduction) teknikleri kullanılmıştır. Çizelge 1'de, çeşitli ppr değerlerine sahip açısal kodlayıcıların derece cinsinden hassasiyetleri ve bu hassasiyetteki bir ölçüm cihazı kullanılarak yönelimi hesaplanmış bir silahın sırası ile 10, 25, 50 ve 100 metre uzaklıktaki hedeflere atışları sırasında oluşacak sapma (hata) miktarları görülmektedir.

Çizelge 1. Açısal Kodlayıcıların Hassasiyet Değerleri ve Atışlardaki Sapma Miktarları

PPR	Hassaslık	10 metre	25 metre	50 metre	100 metre
100	3,6 °	62,91 cm	157,29 cm	314,57 cm	629,15 cm
360	1 °	17,46 cm	43,64 cm	87,28 cm	174,55 cm
1000	0,36 °	6,28 cm	15,71 cm	31,42 cm	62,83 cm
5000	0,072 °	1,26 cm	3,14 cm	6,28 cm	12,57 cm
10.000	0,036 °	0,63 cm	1,57 cm	3,14 cm	6,28 cm
50.000	0,0072 °	0,13 cm	0,31 cm	0,63 cm	1,26 cm
100.000	0,0036 °	0,06 cm	0,16 cm	0,31 cm	0,63 cm

5. SAGES SİSTEMİNE EKLENMESİ PLANLANAN YETENEKLER

SAGES sistemi eksiksiz bir sanal gerçeklik ortamı oluşturacak yeteneklerle donatılmıştır. Ayrıca eğitim gören personelin sadece atışlar sırasında değil tüm eğitim boyunca silahını nasıl kullandığı (örneğin nişan alma sırasında tüfeği sabit tutmayı başarıp başaramadığı) da dahil olmak üzere çok detaylı veri toplayabilme kapasitesine

sahiptir. SAGES bu verileri gereğince işleyip maksimum düzeyde faydalı bilgi üretebilecek modüllere henüz sahip değildir. Kısa bir zamanda, sistemimize eğitim sonrası değerlendirme amacıyla toplanan verileri analiz edebilme ve sonuçları detaylı olarak raporlayabilme özelliklerinin katılmasını planlamaktayız.

Donanım olarak mevcut açısal kodlayıcı hassasiyetimizi daha da artırmak için yüksek çözünürlüklü açısal kodlayıcılar üzerinde araştırmalarımıza devam edip, bu kodlayıcıları yazılım desteği ile çok daha hassas hale getirmeyi planlamaktayız.

Daha ileri safhalarda silah sisteminin pozisyon ve yönelim bilgilerini tamamen kablosuz bir şekilde optoelektronik veya kızılötesi takip sistemleri aracılığıyla sayısal ortama aktarmayı planlamaktayız. Silah sisteminin özgürce hareket edebilmesi sistemimize çok daha heyecan verici açılımlar kazandıracaktır. Örneğin kablosuz bir silah sistemi ve görselleştirme için de HMD kullanıldığı takdirde eğitim gören personel fiziksel olarak tamamen boş ve geniş bir odada veya açık bir alanda yürüyor veya koşuyor olmasına rağmen gözlüklerine ve kulaklıklarına gönderilen sanal gerçeklik bilgileri sayesinde kendisini bilgisayar ortamında yaratılmış herhangi bir mekanda hissedebilecek ve bu sanal gerçeklik ortamında eğitim görebilecektir. SAGES sistemine hava şartları ve sanal ortamda yapılan çatışmada alınacak darbelerin personel tarafından hissedilebilmesi amacıyla klima ve sanal gerçeklik yelekleri de entegre edilebilecektir.

6. SAGES SİSTEMİNİN MEVCUT ÇATIŞMA SİMÜLATÖRLERİ İLE MUKAYESESİ

Yerli ve özgün bir sistem olan SAGES'i, incelediğimiz diğer atış ve/veya çatışma simülatörlerinden ayıran ve onlara üstün kılan birkaç temel özelliği bulunmaktadır:

Bu özelliklerden ilki, silah sistemi tarafından fırlatılan mermi veya roketin diğer simülatörlerde ancak bir sensör veya algılayıcı tarafından tespit edildikten sonra sayısal sisteme girmesine karşılık SAGES simülatöründe silah sisteminin konumu ve mermiler tamamen sayısal sistemin bir parçasıdır. Yani eğitim gören personel tetiği çektiği anda sanal ortamda oluşturulan mermi yine sanal ortamdaki hedeflerine doğru ilerlemeye başlamaktadır. Özetle SAGES, diğer simülatörlerden farklı olarak ***silah sistemini ve mermileri de sayısal ortam içine dahil etmektedir.*** Bu özellik SAGES'e çok büyük avantajlar sağlamaktadır. Örnek olarak MILES benzeri lazerli atış simülatörlerinde 25 metreden nişan alma eğitimi yapmak isteyen bir personelin, lazer algılayıcı mekanizmayı fiziksel olarak kendisinden 25 metre uzağa koyması ve karşısına geçip üzerine lazer gönderici takılı tüfeği ile ateş etmesi gerekmektedir. Oysa SAGES simülatöründe silahın konumu ve mermiler sanal gerçeklik ortamının bir parçası olduğu için personel kendisinden fiziksel olarak 50 cm uzaklıkta bulunan bir bilgisayar monitöründe nişan alma eğitimini gerçekleştirebilmektedir. Monitör aracılığı ile kullanıcıya sunulan sanal gerçeklik ortamında hedefler rahatlıkla personele 25 metre (veya istenilen herhangi bir mesafe) uzaklıktaymış gibi gösterilebilmektedir. Tüfek ve mermiler de bu sanal gerçeklik ortamının birer parçası oldukları için atıcı tamamen gerçekçi, geçerli ve güvenilir bir eğitim almış olmaktadır. Bu özelliğin sağladığı avantajlara başka örnekler de verebiliriz: SAGES'te eğitimler kaydedilebilir ve sonradan analiz ve geribildirim sunmak amacıyla tekrar oynatılırken SAGES nişan alma sırasında oluşan nişan hattını bir çizgiyle işaretleyebilir (ve personelin nişan alırken titretme yapıp yapmadığı yavaş çekimlerde rahatlıkla gözlemlenebilir).

SAGES'i diğer simülatörlerden ayıran bir başka özellik ise yukarıda bahsettiğimiz özelliğin bir açılımıdır. Diğer simülatörlerde silah, mermi ve hedefi temsil eden algılayıcılar fiziksel dünyada bulunmak zorunda oldukları için çatışma eğitimlerinde can ve mal güvenliğini tehdit etmeksizin gerçekçi patlamalar, kara araçlarının üzerlerine ateş açılarak devrilmeleri veya havaya uçurulmaları, helikopter ve uçak gibi araçlara ateş açılarak yere düşürülmeleri, vb. senaryolar oluşturmak ve icra etmek mümkün değildir. Buna karşılık SAGES, sanal gerçeklik ortamını genişleterek silah ve mermileri de sayısal ortama dahil ettiği için her türlü patlayabilen/deforme olabilen cansız hedefleri, araçları, binaları ve vurulduğunda etkisiz hale gelebilen bilgisayar kontrollü sanal düşman kuvvetlerini de eğitim amacıyla personelin karşısına çıkartabilmektedir. Örneğin kullanıcı, karşısındaki monitördeki sanal gerçeklik ortamında kendisine doğru yaklaşmakta olan bir helikopter gördüğü zaman silahı ile ateş açarak bu helikoptere hasar verebilmekte ve eğer yeterli seviyede hasar vermeyi başarabilirse helikopteri düşürebilmektedir. Oysa lazerli simülatörler ile içinde insanlara, araçlara veya binalara hasar vermeyi barındıran herhangi bir eğitim senaryosunu uygulamak mümkün değildir.

Ayrıca lazerli atış sistemleri, havan topu ve top gibi eğik atış yapmak zorunda olan silahları ve atıldıktan sonra kendi itme kuvveti ve hedefe yönelme sistemleri sayesinde çizgisel olarak ilerlemek zorunda olmayan roketleri (örneğin güdümlü silahları) simüle etmek için kullanılamamaktadırlar. Buna karşılık SAGES simülatöründe bir havan topundan atılan mermiye sanal gerçeklik ortamında yerçekimi uygulanabilmekte ve düşmesi gereken yere düşürülebilmektedir. Veya bir güdümlü roketin hedefe yönelme sistemi, programlama yoluyla sanal gerçeklik ortamında temsil edilebilmektedir. Yine lazerli sistemlerde, aslında isabet ettiğinde geniş bir alanda tahribat/etki yaratması gereken roketler ve top mermileri sadece noktasal olarak algılanabilmektedir. Oysa sanal gerçeklik ortamında fırlatılan bir roket, isabet ettiği yerde yaratması gereken alan tahribatı etkisini rahatlıkla yaratabilmektedir.

Lazer göndericiler silahlara her takıldıklarında, nişan alma sisteminin kullanıcıya göre sıfırlamasının yapılması gerekmektedir. Bu da lazerli eğitim teçhizatının eğitime hazırlanma süresini uzatmaktadır. Oysa SAGES sisteminde bilgisayar tarafından otomatik olarak sıfırlama yapılabilmektedir. Ayrıca lazer ışığı eğitim alanındaki çalılar ve ağaçların yapraklarından etkilendiği için bu tür engellerin arkasındaki hedeflere nişan alınması durumunda sağlıklı sonuçlar alınamamaktadır. Sanal gerçeklik ortamında ise bu tür problemler çeşitli programlama teknikleri kullanılarak aşılabilmektedir.

7. SONUÇ

Diğer simülatörlerden farklı olarak gece, gündüz, alacakaranlık, güneşin konumu, mevsimler, yağmur, bulut miktarı, sis, rüzgar, vb. her türlü durumu görselleştirebilme ve 3 boyutlu çevresel ses destekli ***gelişmiş bir sanal gerçeklik ortamı*** sunması ve ***silah sistemini ve mermileri*** de ***sanal gerçeklik ortamının bir parçası*** olarak işlemesi dolayısıyla SAGES, başta Türk Silahlı Kuvvetleri ve Emniyet Teşkilatı mensupları olmak üzere tüm güvenlik personeline atış, çatışma ve konsantrasyon yeteneklerini muhafaza etme ve geliştirebilme imkanı sunacak şekilde tasarlanmıştır. SAGES sistemi, planlanan bir operasyon öncesi güvenlik personelini her türlü ihtimale ve yüksek stres koşullarına hazırlayabilmek için bölgenin 3 boyutlu haritasının SAGES'e yüklenerek

çeşitli taarruz ve savunma senaryoları üzerinde eğitim verilmesi, eğitim sonrası performansın analiz edilebilmesi, vb. faaliyetler için de destek sağlamaktadır.

SAGES'in alternatif kullanım alanları içinde ***güvenlik personeli seçimi*** ve ***performans değerlendirmesi*** de bulunmaktadır. Mevcut sistemde personel seçim kriteri sayısal, sözel ve zihinsel yeteneklerinin sınanması şeklindedir. SAGES ile personelin stres altında performansı ve dayanma kapasitesi de ölçülebilecek ve işe daha uygun personelin istihdam edilmesi sağlanabilecektir.

KAYNAKÇA

[1] Muchinsky, P. M., (1997), "Psychology Applied to Work", Brooks/Cole Publishing Company, 5. Baskı, sf. 92-94.

[2] Blanchard, P. N., Thacker, J. W., (1999), "Effective Training: Systems, Strategies and Practices", Prentice Hall, sf. 293.

[3] MILES Sistemi hakkında daha detaylı bilgi için kaynak: http://www.fas.org/man/dod-101/sys/land/miles.htm

[4] MILES 2000 Sistemi hakkında daha detaylı bilgi için kaynak: http://www.peostri.army.mil/PRODUCTS/MILES

[5] Temizer, S., (Ocak 2007), "The State of the Art and The Future of Modeling and Simulation Systems", Journal of Aeronautics and Space Technologies (Hava Harp Okulu – Havacılık ve Uzay Teknolojileri Enstitüsü, Havacılık ve Uzay Teknolojileri Dergisi), Cilt 3, Sayı 1, sf. 41-50.

[6] Temizer, S., (Eylül 2005), "Yazılım Yapılandırma Teknikleri: Temizer Sistemi", II. Ulusal Yazılım Mühendisliği Sempozyumu, UYMS'05, sf. 305-313.

SU ÜSTÜ PLATFORMLARININ MODELLENMESİ VE TİCARİ BİR SENTETİK HEDEF SİMÜLATÖRÜ İLE ENTEGRELİ SİMÜLASYONU

Mehmet HAKLIDIR, Deniz ALDOĞAN, İsa TAŞDELEN

TÜBİTAK MAM, Bilişim Teknolojileri Enstitüsü, Gebze-Kocaeli,

{mehmet.haklidir, deniz.aldogan, isa.tasdelen}@bte.mam.gov.tr

ÖZ

VR-Forces, askeri simülasyon amaçlı platformlar üreten esnek yapılı bir yazılımdır. Bu yazılımda benzetimi yapılan platformlar, basit hareket denklemlerine sahiptir. Ayrıca bu platformların hareket denklemlerinde kullanılan ortam parametreleri yüksek sadakate sahip hareket modelleri oluşturacak düzeyde değildir. Var olan hareket modelleri, gerçek şartlara yakın platformlar ve ortam modelleri kullanılarak taktik eğitim sağlamayı hedefleyen askeri simülasyonların ihtiyaçlarını karşılamakta yetersiz kalmaktadır.

Bu çalışmada, VR-Forces kullanılarak yüksek sadakate sahip askeri eğitim simülatörünün gerçeklenmesi için bir uygulama geliştirilmiştir. Bu kapsamda, sanal bir askeri eğitim sahasında yardımcı taktik unsur olarak yer alan su üstü platformlarının modellenmesi ve VR-Forces yazılımı ile entegreli simülasyonu gerçekleştirilmiştir. Ayrıca dalga, akıntı, rüzgar, mevsim, gece gündüz farkı gibi taktik ortamın fiziksel şartları da modellenmiş ve platformların bu şartlar ve hidrodinamik kuvvet-momentlerin etkisindeki dinamik davranışlarının gözlemlenmesi hedeflenmiştir. C++ programlama diliyle gerçeklenen hareket modeli yazılım modülü, VR-Forces arka uygulamasında (back-end) bulunan bileşen mimarisine bir bileşik nesne olarak entegre edilmiştir.

Anahtar Kelimeler: Dinamik Modelleme, Dağıtık Simülasyon, HLA, Su Üstü Gemileri, Ortam Benzetimi

MODELING AND INTEGRATION OF SURFACE PLATFORMS WITH A COTS SYNTHETICAL PLATFORM SIMULATOR

ABSTRACT

VR-Forces is flexible software that generates platforms for military simulation purposes. These platforms have simple motion equations with environment parameters of inadequate fidelity. These motion models are insufficient for meeting the needs of training-critical military simulations that use platforms and environment models with high-fidelity.

In this paper, we present the modeling and VR-Forces-integrated simulation of surface platforms that appear as secondary tactical elements within a high-fidelity virtual military training simulation. We aim to observe the dynamic behaviors of these platforms under the effect of hydrodynamic force-moments and physical conditions such as current, wind, season, day and night difference that belong to the tactical environment. The software module for the proposed motion model has been implemented in C++ programming language and has been integrated to the component architecture of VR-Forces back-end as a composite object.

Keywords: Dynamical Modeling, Distributed Simulation, HLA, Surface Ships, Environmental Simulation

1. GİRİŞ

Dağıtık simülasyon sistemleri, farklı ortamlarda birbirlerinden bağımsız koşan, ortak bir hafızası olmayan uygulamalardır. Bu simülasyonların birlikte çalışabilmeleri ve tekrar kullanılabilmelerine destek vermek amacıyla, ABD Savunma Bakanlığı bünyesinde HLA (High Level Architecture) adı verilen bir standart geliştirilmiştir.[1]

VR-Forces, HLA uyumlu olup, askeri, güvenlik ya da acil müdahale konularında bilgisayar tarafından üretilen kuvvetlerle (Computer Generated Forces, CGF) simülasyon gerçekleştirmeyi sağlayan esnek yapılı bir yazılımdır. Birbirleriyle haberleşmek için RTI (Run-Time Infrastructure) kullanan VR-Forces uygulamaları, arayüz (front-end) ve arka (back-end) uygulamalar olmak üzere ikiye ayrılırlar.[2]

VR-Forces tabanlı grafik kullanıcı arayüzü uygulaması sayesinde, seçilen bir savaş sahası üzerinde, kullanıcı komutları ile senaryo hazırlanabilir veya hazırlanan bir senaryonun çalışma süreci gözlemlenebilir. Senaryo hazırlanması, seçilen savaş sahasına, platformların eklenmesi veya silinmesi, bu platformlara ait özelliklerin değiştirilmesi ve çeşitli görevler atanması gibi işlemleri içerir.

Simülasyon motoruna sahip VR-Forces arka uygulaması ise, kullanıcı arayüzünden gelen mesajları işleyerek platformların benzetimini ve RTI üzerinden haberleşen diğer simülasyon bileşenleri (federeler) ile iletişimini (platform değişkenlerinin yayınlanması, vb.) gerçekleştirir. VR-Forces uygulamalarında benzetimi yapılan platformlar, (su-üstü, hava, kara ve denizaltı) basit hareket denklemlerine sahiptir. Ayrıca bu platformların hareket denklemlerinde kullanılan ortam parametreleri gerçekçi hareket modelleri oluşturmaya uygun sadakat seviyesinde değildir. Bu hareket modelleri, sadakat seviyesi yüksek, eğitim amaçlı kullanılacak askeri simülasyonların ihtiyaçlarını karşılamakta yetersiz kalmaktadır.

Bu çalışmada, ortam şartlarını (dalga, akıntı, rüzgar, mevsim, gece gündüz farkı) ve hidrodinamik kuvvetleri hesaba katan, 4 serbestlik dereceli su-üstü platformları için bir hareket modeli geliştirilmiştir. Bu hareket modeli için gerçeklenen yazılım modülü, VR-Forces 3.9.1 sürümünün arka uygulamasında bulunan bileşen mimarisine bir bileşik nesne (composite object) olarak entegre edilmiştir. Buradaki amaç, bağımsız bir bileşen olarak geliştirilen yazılım modülünün VR-Forces benzeri başka uygulamalarla da kolaylıkla çalıştırılabilmesidir.

Çalışmanın ilk kısmında su üstü platformlarının hareket denklemlerinin ve manevra modelinin formüle edilme esasları yer alacak, son kısmında ise simülasyon mimarisi anlatılarak, örnek bir çözümle simülasyon sonuçları tartışılacaktır.

2. SU ÜSTÜ PLATFORMUNUN MATEMATİKSEL MODELİ

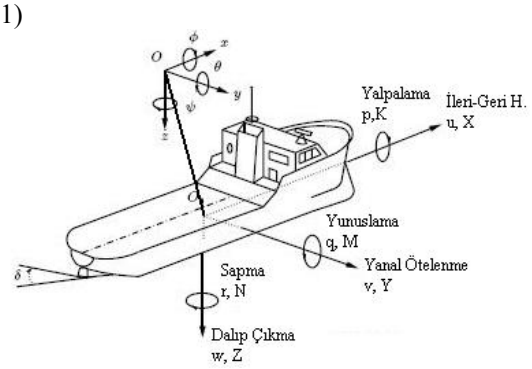
2.1 Altı Serbestlik Dereceli Genel Hareket Denklemleri

Su üstü platformlarının hareketleri 6 serbestlik derecelidir. Her bir serbestlik derecesinin tanımı ve notasyonu Çizelge 1'de gösterilmiştir [3]. Çizelgede tanımlı ilk üç değer ve bu değerlerin zamana bağlı türevleri, platformun x-, y-, z- eksenindeki pozisyonunu ve hareketini, diğer üç değer ise platformun oryantasyonunu ve dönme hareketlerini tanımlamaktadır.

Çizelge 1. Serbestlik derecesi tanımları ve notasyonlar

Serbestlik Derecesi	Tanım	Eksen	Kuvvet ve Momentler	Doğrusal ve Açısal Hız	Konum ve Euler Açıları
1	Boyuna Öteleme	x	X	u	x
2	Yanal Öteleme	y	Y	v	y
3	Dalıp Çıkma	z	Z	w	z
4	Yalpa	x	K	p	ϕ
5	Yunuslama	y	M	q	θ
6	Sapma	z	N	r	ψ

Su üstü platformun 6 serbestlik dereceli hareketi incelendiği zaman, gerçekleştirilebilmesi için iki eksen takımının tanımlanmasına ihtiyaç duyulduğu gözlemlenir. (Şekil 1)



Şekil 1. Koordinat Sistemi

Hareket eksen takımı $X_0\ Y_0\ Z_0$ platforma sabitleştirilmiştir ve gövdeye sabitlendirilmiş eksen takımı olarak adlandırılır. Bu eksen takımının orijini olan O noktası her zaman geminin ağırlık merkezi olarak seçilir.

xz ekseni, simetri ekseni ve ağırlık merkezi G'nin bu eksen üzerinde x_G, y_G and z_G noktaları ile tanımlandığı kabul edilirse, su üstü platformunun temel hareket denklemleri aşağıdaki gibidir[4]:

Boyuna Öteleme: $$X = m[\dot{u} + qw - rv + x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(rp + \dot{q})] \qquad (1)$$

Yanal Öteleme: $$Y = m[\dot{v} + ru - pw - y_G(r^2 + p^2) + z_G(qr - \dot{p}) + x_G(qp + \dot{r})] \quad (2)$$

Dalıp-Çıkma: $$Z = m[\dot{w} + pv - qu - z_G(p^2 + q^2) + x_G(rp - \dot{q}) + y_G(rq + \dot{p})] \quad (3)$$

Yalpa: $$K = I_X\dot{p} + (I_Z - I_y)qr + m[y_G(\dot{w} + pv - qu) - z_G(\dot{u} + ru - pw)] \quad (4)$$

Yunuslama: $$M = I_y\dot{q} + (I_x - I_Z)rp + m[z_G(\dot{u} + qw - rv) - x_G(\dot{w} + pv - qu)] \quad (5)$$

Sapma: $$N = I_Z\dot{r} + (I_y - I_x)pq + m[x_G(\dot{v} + ru - pw) - y_G(\dot{u} + qw - rv)] \quad (6)$$

2.2 Basitleştirme Kabulleri

Bu çalışma kapsamında, hedeflenen sadakat seviyesinde bir model elde edilebilmek için aşağıda yer alan kabuller yapılmıştır.

- Y ekseninde yunuslama momenti etkisiyle tanımlanan açısal hız ve ivmenin (q, $\dot{q}$) sıfır olduğu ve bu momentin manevra modelini etkilemediği
- Z ekseninde platformun dalıp çıkmasına neden olan Z kuvvetinin sıfır olduğu (w, $\dot{w} = 0$) ve bu kuvvetin manevra modelini etkilemediği
- Gövdeye bağlı koordinat sisteminin orijinin, platformun ağırlık merkezi olduğu ve ilgili hareketlerin bu noktayı merkez kabul ederek gerçekleştirildiği
- Platformun uzunlamasına ekseni boyunca simetrik olduğu (y_G=0)
- Simülasyon kapsamındaki su üstü platformlarının türetilecek tek bir jenerik hareket modelini kullanacağı, platformlar arasında farklılaşma ve sınıflandırmanın (özelleştirme) parametrik olarak gerçekleştirileceği

kabulleri yapılmıştır.

2.3 Basitleştirilmiş Hareket Denklemleri

İlgili kabullerle su üstü platformlarının hareketi 4 serbestlik derecesine indirgenmiş ve aşağıdaki hareket denklemleri türetilmiştir

Boyuna Öteleme: $$X = m[\dot{u} - rv - x_G r^2 + z_G rp] \quad (8)$$

Yanal Öteleme: $$Y = m[\dot{v} + ru - z_G\dot{p} + x_G\dot{r}] \quad (9)$$

Yalpa: $$K = I_X\dot{p} - mz_G(\dot{u} + ru)] \quad (10)$$

Sapma: $$N = I_Z\dot{r} + mx_G(\dot{v} + ru) \quad (11)$$

2.4 Kuvvet ve Momentler

Temelde su üstü platformuna etkiyen kuvvet ve momentler, hidrodinamik kuvvet-momentler ve çevresel dış yükler olarak ikiye ayrılabilir.

2.4.1 Hidrodinamik Kuvvet ve Momentler

Su basıncının yüzen su üstü platformun ıslak alanı boyunca entegrasyonu, su üstü platformun bünyesinde hidrodinamik kuvvet ve momentlerin oluşumuna neden olur. Bu kuvvet ve momentler Abkowitz metodu kullanılarak geminin pozisyon, hız ve ivme bileşenleri ile doğrusal olmayan bir denklem takımı şeklinde tanımlanabilir. Manevra modellerinin geliştirilmesi için en önemli adım Taylor'un serisindeki kuvvet ve momentlerin genişletilmesidir. Bu şekilde doğrusal olmayan terimler bağımsız

değişkenler olarak davranırlar ve polinomal bir denklem oluştururlar. Bu genişlemeyi kullanmak için bu fonksiyon ve türevlerinin bölgedeki değişkenlerin değerleri gözden geçirildiğinde, sürekli ve sınırlı olması gerekir. Modelin kesinliği, serinin kesildiği noktaya bağlı olarak değişir. Aşağıda taylor serisi üçüncü kuvvete kadar genişletilerek elde edilmiş kuvvet ve momentler yer almaktadır.[5]

$$X_{hid} = X_{\dot{u}}(\dot{u}) + X_{vr}vr + X_{uu}v^2 \tag{12}$$

$$Y_{hid} = Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} + Y_{\dot{p}}\dot{p} + Y_{\phi|uv|}\phi|uv| + Y_{\phi|ur|}\phi|ur| + +Y_{\phi|uu|}\phi|uu| + Y_{|u|v}|u|v + Y_{ur}ur + Y_{v|v|}v|v| + Y_{v|r|}v|r| + Y_{r|v|}r|v| \tag{13}$$

$$K_{hid} = K_{\dot{v}}\dot{v} + K_{\dot{p}}\dot{p} + K_{|u|v}|u|v + K_{ur}ur + K_{v|v|}v|v| + K_{v|r|}v|r| + K_{r|v|}r|v| + K_{|uv|\phi}|uv|\phi + K_{|ur|\phi}|ur|\phi + K_{uu\phi}uu\phi + K_{|u|p}|u|p + K_{|p|p}|p|p + K_{p}p + K_{\phi\phi\phi}\phi\phi\phi - \Delta G_z(\phi) \tag{14}$$

$$N_{hid} = N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} + N_{|u|v}|u|v + N_{|u|r}|u|r + N_{|r|r}|r|r + N_{|v|r}|v|r + N_{|uv|\phi}|uv|\phi + N_{|ur|\phi}|ur|\phi + N_{|u|u\phi}|u|\phi u \tag{15}$$

2.4.2 Çevresel Dış Yükler

Su üstü platforma etkiyen çevresel dış yükler sinüzoidal ve doğrusal bir karakterde olan dalga yükleri ve zamana göre değişken akıntı ve rüzgar kuvvetleri olarak iki gruba ayrılabilir.

2.4.2.1 Dalga Kuvveti

Karışık denizlere ait gerçek verinin mevcut olmadığı durumlarda denizcilik hesaplarında genellikle idealleştirilmiş matematiksel spektrum fonksiyonlarından yararlanılır. Bunlardan en basiti ve en çok kullanılanı Pierson–Moskowitz spektrumu tam oluşmuş denizler için sabit hızda ve sonsuz bir periyot için sonsuz genişlikte bir saha üzerinden esen rüzgarlar için bir dalga spektrum bağıntısı verilmektedir.[6]

Bu spektrum dalga frekansı ve rüzgâr hızına bağlı olarak aşağıdaki gibi ifade edilir.

$$S_{\xi} = \frac{0.0081g^2}{\omega^5}\exp\left[-0.74\left(\frac{g}{V\omega}\right)^4\right] \tag{16}$$

Burada
ω : Dalga frekansı [rad/s]
V: Deniz yüzeyinden 19,5 metre yukarıda ölçülen rüzgâr hızı [m/s]

Dalga kuvveti ve momentlerinin birinci mertebeden bileşenleri su üstü platformunun hareketini doğrudan etkileyecektir.[7]

$$F_{wi}(t) = \sum_{n=1}^{N}\left[\frac{F_{wa}}{\varsigma_a}(\omega)\varsigma_{an}.\cos\left(\omega_n t + \varepsilon_{\varsigma n} + \varepsilon_{F\varsigma\varsigma}\right)\right]; i = 1,2,...,6 \tag{17}$$

2.4.2.2 Akıntı ve Rüzgar Kuvveti

Rüzgar ve akıntı yüklerinin her ikisi de hız ve bu hızın yön vektörleriyle ifade edilirler. Vektör için ayrıca bir işaret tanımlaması gerekecektir ki, bunun için t=0 anında, geminin kıçından gelen rüzgar veya akıntı sıfır pozisyonunda ve gemiyi sancağa döndüren söz konusu etkilerin ise pozitif yönde olduğu varsayımı yürütülür. Çalışmada her iki etkinin

yalnız geminin ilerleme, yan öteleme ve savrulma hareketleri üzerinde geçerli olduğu varsayılmıştır.[7]

Rüzgar hızı V_R ve rüzgarın açısı γ_R olmak üzere su üstü platforma etkiyen rüzgar kuvvet ve momentleri aşağıdaki gibidir.

$$X_{wr} = \frac{1}{2} C_X(\gamma_R) \rho_w V_R^2 A_T \tag{18}$$

$$Y_{wr} = \frac{1}{2} C_Y(\gamma_R) \rho_w V_R^2 A_L \tag{19}$$

$$N_{wr} = \frac{1}{2} C_N(\gamma_R) \rho_w V_R^2 A_L L \tag{20}$$

Burada C_X, C_Y and C_N direnç katsayıları, ρ_w havanın özgül ağırlığı, A_T ve A_L, geminin enine ve boyuna projeksiyon alanları ve L geminin boyudur. İlgili direnç katsayıları C_i, rüzgarın geliş açısına bağlı olarak farklı değerler almaktadır. Benzetimde, bu katsayılar Isherwood metodu kullanılarak [8], interpolasyonla değişken olarak hesaplanmıştır. Akıntı hızının su üstü platformuna etkisini gösteren kuvvet ve moment denklemleri rüzgarın etkisini gösteren denklemlerle benzer olacaktır.

3. SU ÜSTÜ PLATFORMUN MANEVRA MODELİ

Bir önceki bölümde elde edilen 4 serbestlik dereceli basitleştirilmiş hareket modeli, hidrodinamik kuvvetler ve çevresel dış yüklerle ilişkilendirilirse doğrusal olmayan bir manevra modeli elde edilir. Hidrodinamik türevler, bağımsız değişkenler olarak davranıp polinomal bir hareket denkleminde yer alan katsayılar olabilmeleri için gemi geometrisinden yararlanılan ayrı bir yazılımdan türetilirler. Aynı zamanda gemi hareket denklemlerinin bileşenleri, gemi hızına bağlı olarak boyutsuzlaştırılarak normalize edilir. Bu denklem setinin daha kullanışlı olmasını sağlamaktadır.[9]

$$X' = X'(u') + (1-t)T'(J) + X'_{vr}v'r' + X'_{vv}v'^2 + X'_{rr}r'^2 + X'_{\phi\phi}\phi'^2 + c_{RX}F'_N \sin\delta' \tag{21}$$

$$\begin{aligned} Y' = {} & Y'_v v' + Y'_r r' + Y'_p p' + Y'_\phi \phi' + Y'_{vvv}v'^3 + Y'_{rrr}r'^3 + Y'_{vvr}v'^2 r' + Y'_{vrr}v'r'^2 \\ & + Y'_{vv\phi}v'^2\phi' + Y'_{v\phi\phi}v'\phi'^2 + Y'_{rr\phi}r'^2\phi' + Y'_{r\phi\phi}r'\phi'^2 + (1+a_H)F'_N \cos\delta' \end{aligned} \tag{22}$$

$$\begin{aligned} K' = {} & K'_v v' + K'_r r' + K'_p p' + K'_\phi \phi' + K'_{vvv}v'^3 + K'_{rrr}r'^3 + K'_{vvr}v'^2 r' + K'_{vrr}v'r'^2 \\ & + K'_{vv\phi}v'^2\phi' + K'_{v\phi\phi}v'\phi'^2 + K'_{rr\phi}r'^2\phi' + K'_{r\phi\phi}r'\phi'^2 - (1+a_H)z'_R F'_N \cos\delta' \end{aligned} \tag{23}$$

$$\begin{aligned} N' = {} & N'_v v' + N'_r r' + N'_p p' + N'_\phi \phi' + N'_{vvv}v'^3 + N'_{rrr}r'^3 + N'_{vvr}v'^2 r' + N'_{vrr}v'r'^2 \\ & + N'_{vv\phi}v'^2\phi' + N'_{v\phi\phi}v'\phi'^2 + N'_{rr\phi}r'^2\phi' + N'_{r\phi\phi}r'\phi'^2 + (x'_R + a_H x'_H)F'_N \cos\delta' \end{aligned} \tag{24}$$

Bütün denklemlerde boyutsuz bileşenler (′) sembolü ile gösterilir.

Gemi kütlesi ve eklenmiş kütlenin yer aldığı atalet matrisi aşağıdaki şekilde gösterilebilir.

$$M = \begin{bmatrix} m' + m'_x & 0 & 0 & 0 \\ 0 & m' + m'_y & -m'_y l'_y & m'_y \alpha'_y \\ 0 & -m'_y l'_y & I'_x + J'_x & 0 \\ 0 & m'_y \alpha'_y & 0 & I'_z + J'_z \end{bmatrix} \tag{25}$$

M tekil olmamak üzere,

$$\begin{bmatrix} \dot{u}' \\ \dot{v}' \\ \dot{p}' \\ \dot{r}' \end{bmatrix} = M^{-1} \begin{bmatrix} X' \\ Y' \\ K' \\ N' \end{bmatrix} \tag{26}$$

M matrisinin basitleştirilmiş gösterimi aşağıdaki şekilde olabilir.

$$M = \begin{bmatrix} m_{11} & 0 & 0 & 0 \\ 0 & m_{22} & m_{32} & m_{42} \\ 0 & m_{32} & m_{33} & 0 \\ 0 & m_{42} & 0 & m_{44} \end{bmatrix} \tag{27}$$

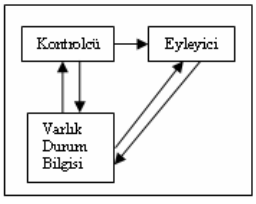
Atalet matrisinin tersi alınıp (26)'da yerine konursa,

$$\begin{bmatrix} \dot{u}' \\ \dot{v}' \\ \dot{p}' \\ \dot{r}' \end{bmatrix} = \begin{bmatrix} (1/m_{11})X' \\ ((m_{33}m_{44}Y' - m_{32}m_{44}K') - m_{42}m_{33}N')/(m_{22}m_{33}m_{44} - m_{32}^2 m_{44} - m_{42}^2 m_{33}) \\ (-m_{32}m_{44}Y' + m_{22}m_{44}K' - m_{42}^2 K' + m_{32}m_{42}N')/(m_{22}m_{33}m_{44} - m_{32}^2 m_{44} - m_{42}^2 m_{33}) \\ (-m_{42}m_{33}Y' + m_{32}m_{42}K' + m_{22}m_{33}N' - m_{32}^2 N')/(m_{22}m_{33}m_{44} - m_{32}^2 m_{44} - m_{42}^2 m_{33}) \end{bmatrix} \tag{28}$$

durum türevleri elde edilir.

4. SU ÜSTÜ PLATFORMUNUN SİMÜLASYONU

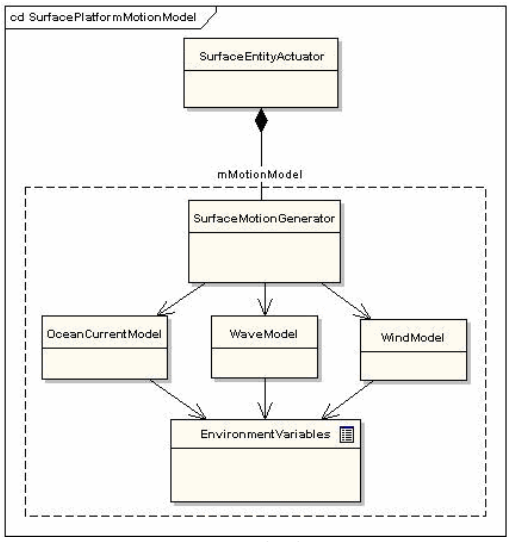
Yazılım modülünün, VR-Forces arka uygulamasına entegrasyonu, VR-Forces bileşen mimarisi kullanılarak gerçekleştirilmiştir. VR-Forces arka uygulamasında senaryo kapsamında yaratılan her varlığa (entity) ait bir bileşen yöneticisi (Component Manager) bulunur. Bileşen yöneticisi, varlığın davranışını gerçeklemek için kontrolcü (controller) ve eyleyici (actuator) olmak üzere iki tip sınıf kullanır. Kontrolcü sınıflar, kullanıcı arayüzünden gelen varlığa ait özellikleri değiştirme veya görev ataması gibi komutları içeren mesajları işleme koyarlar. Eyleyici sınıflar ise, senaryo koşumu sırasında, kontrolcü sınıflardan görevin gerçekleştirilmesi ile ilgili bilgileri alarak, her zaman diliminde (frame time), varlığın konum, hız, vb. gibi durum bilgilerini güncellerler. Kontrolcü, eyleyici sınıflar ve varlık durum bilgisi arasındaki iletişim Şekil 2'de gösterilmiştir.



Şekil 2. Kontrolcü, Eyleyici Sınıflar ve Varlık Durum Bilgisi

Su-üstü platformların hareket modeli için geliştirilen yazılımın VR-Forces uygulamasına entegrasyonu kapsamında, VR-Forces sınıflarından su-üstü platformlarının hareketine özgü kontrolcü ve eyleyici sınıflar türetilmiştir. Su-üstü platformları hareket modeli yazılımı, SurfaceEntityActuator adlı eyleyici sınıfa ait bir üye değişken olarak tanımlanmıştır (Şekil 3). Eyleyici sınıf, su-üstü platformunun konum, hız, vb. gibi durum bilgilerini yazılım modülünü kullanarak günceller.

Koşum başlamış ve su-üstü platformuna görev atanmış ise, yazılım modülü çalışması başlatılır. Bunun için, hareket denklemleri içerisinde yer alan başlangıç değerleri, su-üstü platformuna ait karakteristik parametreler ve hidrodinamik türevler modüle gönderilir. Ayrıca, su-üstü platformunun ilerleme hızı, vs. gibi kullanıcı etkileşim değerleri de eyleyici sınıf tarafından modüle aktarılmaktadır.



Şekil 3. Sınıf Diyagramı

Yazılım modülü, Şekil 3'te gösterilen OceanCurrentModel, WaveModel ve WindModel sınıflarını kullanarak rüzgar hızı, akıntı hızı, vs. gibi ortam değişkenlerinin akıntı, rüzgar ve dalga modellerinde işlenmesini ve bu modellerin sonuçlarının hareket denklemlerine aktarılmasını sağlar.

Modül tarafından hesaplanan zamana bağlı hız, ivme ve yer değiştirme değerlerinden, Euler entegrasyonu yardımıyla su-üstü platformunun Δt süre sonundaki yeni bilgileri (konum, ivme, hız, vs.) elde edilir. Bu bilgileri alan eyleyici sınıf, su-üstü platformunun durum bilgilerini günceller. Bu güncellemeler, su-üstü platformu için yazılan VR-Forces yerel ağ arayüzü sınıfı sayesinde, RTI üzerinden federasyondaki diğer federelere yayınlanır.

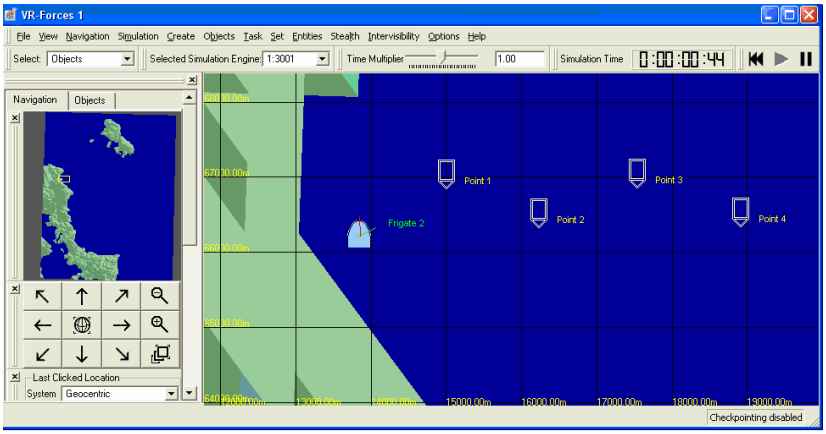
4.1 Örnek Uygulama

Örnek uygulama için seçilen su üstü platformun önemli parametreleri Çizelge 2'de verilmiştir.

Çizelge 2. Su üstü platformun parametreleri

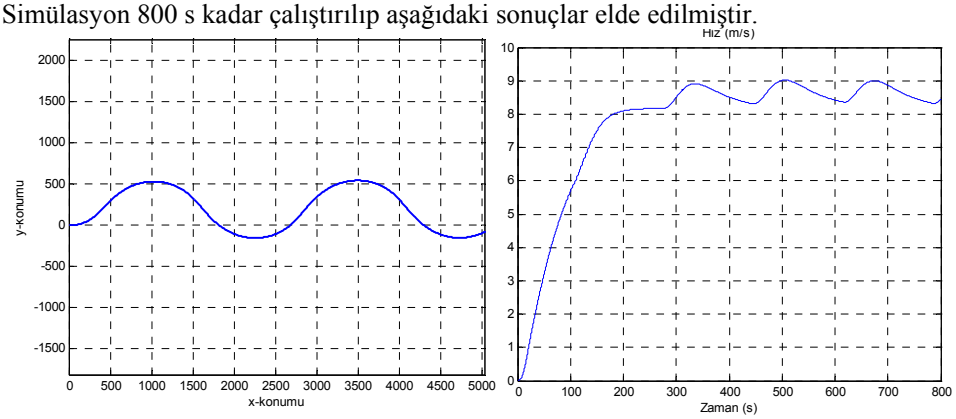
Tanım		Değer	Tanım	Değer
Toplam Uzunluk		175 m	Yükseklik	10.39 m
Toplam Genişlik		25.4 m	Blok Katsayısı	0.559
Draft	ön	8 m	Dümen Alanı	33.0376 m^2
	arka	9 m	En Boy Oranı	1.8219
	orta	8.5 m	Pervane Çapı	6.533 m
Deplasman Hacmi		21222 m^3	Gemin Kütlesi	792 kg

Örnek uygulamada, su üstü platformun 8 m/s'lik hızla, 45 derecelik bacaklar (zig) yapması hedeflenmiş, bununla ilgili rota bilgisi VR Forces'ın kullanıcı arayüzü kullanılarak girilmiştir.(Şekil 4)



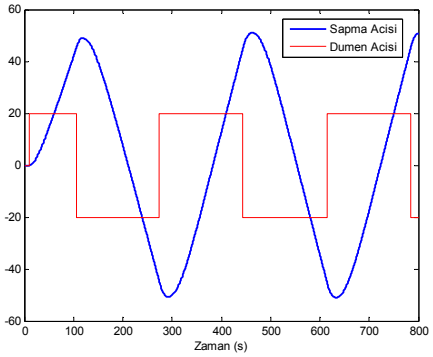
Şekil 4. Platformun Bacak (Zig) Rotası

Simülasyon 800 s kadar çalıştırılıp aşağıdaki sonuçlar elde edilmiştir.



(a) **(b)**

Şekil 5. Bacak Yapan Platformun Zamana Bağlı **(a)** Konum Değişimi **(b)** Hız Değişimi



Şekil 6. Platformun Sapma ve Dümen Açısı Değişimleri

Literatürde Kempf'in zig zag testi[10] olarak da bilinen bu uygulamada, test edilen geminin ilk hızı 0 verilip, 8 m/s hıza çıkması ve aynı zamanda belirtilen noktalara gitmesi emri verilmiştir. Platformun ilk noktaya varması yaklaşık olarak 96 s sürmüştür. İlk döngüyü de yaklaşık olarak 295 s'de tamamlamıştır. Sonuçlar, büyük bir geminin gerçekleştirmesi gereken hareket davranışlarına uygun ve simülasyon açısından da tatmin edicidir.

5. SONUÇLAR

Bu çalışmada, ortam şartlarını ve hidrodinamik kuvvetleri hesaba katan, 4 serbestlik dereceli su-üstü platformları için genel bir hareket modeli geliştirilmiştir ve bu hareket modeli için gerçeklenen yazılım modülü, VR-Forces arka uygulamasında bulunan bileşen mimarisine bir bileşik nesne olarak entegre edilmiştir. Bu entegrasyonun sonunda, VR Forces uygulamasının su üstü platformlarının hareketlerinin, eğitim amaçlı kullanılacak sadakat seviyesi yüksek, askeri simülasyonların ihtiyaçlarını karşılayabilecek seviyeye çıkarıldığı görülmüştür. Aynı zamanda ortam şartlarının simülasyonu ve etkilerinin ilgili platformlara yansıtılması simülasyon mimarisinin zenginleşmesini ve eğitim amaçlı kullanılabilirliğinin artırılmasını sağlamıştır.

KAYNAKÇA

[1] Duman, I., Savaşan, H., Şahin, I., Bir Modelleme ve Simülasyon Standardı olarak HLA ve Simülasyon Uygulamalarında Kullanımı,

[2] VR-Forces Developer's Guide. Mak Technologies.

[3] SNAME(1950), The Society of Naval Architects and Marine Engineers. Nomenclature for treating the motion of submerged body through a fluid, Technical Research Bulletin No. 1-5

[4] Fossen, T. I., Guidance and Control of Ocean Vehicles, John Wiley & Sons, 1994

[5] Abkowitz, M. A., Stability and Motion Control of Ocean Vehicles, M.I.T. Press, Cambridge, Massachusetts, 1969.

[6] Berteaux, H. O., Buoy Engineering, Wiley and Sons, New York, 1976.

[7] Neşer, G., Yanaşan Bir Geminin Hareketlerinin Zamanın Bağlısı Olarak Modellenmesi, DEÜ Mühendislik Fakültesi Fen ve Mühendislik Dergisi, Cilt:8 S:1, 2006.

[8] Isherwood, R. M., Wind Resistance of Merchant Ships, RINA Trans., Vol. 115, pp. 327-338, 1972.

[9] Son, K. H. and Nomoto, K., On the Coupled Motion of Steering and Rolling of a High Speed Container Ship, Naval Architect of Ocean Engineering, Vol. 20, pp. 73-83, 1982.

[10] Kempf, G., Measurements of the Propulsive and Structural Characteristics of Ships, Transactions of SNAME, Vol. 40, pp. 42-57, 1932.

[11] Fossen, T. I., Nonlinear Modeling and Control of Underwater Vehicles, Dr. Ing. thesis, Dept. of Engineering Cybernetics, The Norwegian Institute of Technology, Trondheim, 1991.

ÇEVİK ENGEL MÜFREZE SAVUNMASI VE TANK TAKIMI TAARRUZUNUN DAĞITIK ETKİLEŞİMLİ SİMÜLASYONU[2]

Baki GÜNDÜZ[(a)], İsmet ÇUHADAR[(b)], Halit OĞUZTÜZÜN[(c)]

[(a)]K.K.K.lığı Dağ, Komd. Okl. ve Eğt. Mrk. K.lığı, ISPARTA,

bakigunduz@yahoo.com.

[(b)]K.K.K.lığı 3 ncü Or. İs. Svş. Tb. K.lığı, ERZURUM,

ismetcuhadar@yahoo.com.

[(c)]ODTÜ, Bilgisayar Mühendisliği Bölümü, ANKARA,

oguztuzn@ceng.metu.edu.tr.

ÖZ

Çevik Engel Müfreze Savunması ve Tank Takımı Taarruzu Simülasyonu; çevik engel müfrezesi personelini, tank ve tank takım komutanlarını, tanksavar silah nişancısını, ileri gözetleyicileri ve ateş idare merkezi personelini, temsili ortamda birlikte eğitmek maksadıyla hazırlanmış bir dağıtık etkileşimli simülasyon sistemidir. Sistem HLA uyumlu olup, kişisel bilgisayarlardan oluşan bir yerel ağ üzerinde koşturulmaktadır. Sistem, yedi federeden oluşan bir federasyon olarak gerçekleştirilmiştir. Bunlar İleri Gözetleyici, Ateş İdare Merkezi, Tank, Tank Takımı, Piyade (tanksavar), Çevik Engel Müfrezesi ve Eğitmen federeleridir. Federasyon senaryosu, taarruz eden tank takımına karşı hazırlıksız savunmayı içermektedir.

Anahtar Kelimeler : Ateş İdare Merkezi, Çevik Engel Müfrezesi, Dağıtık Etkileşimli Simülasyon, İleri Gözetleyici, Tank Takımı.

DISTRIBUTED INTERACTIVE SIMULATION OF MOBILE OBSTACLE DETACHMENT DEFENSE AND TANK PLATOON OFFENSIVE OPERATIONS

ABSTRACT

Mobile Obstacle Detachment Defense and Tank Platoon Offensive Operations Simulation, is a distributed interactive simulation system aimed to train Tank, Anti-tank Infantry, Mobile Obstacle Detachment, Forward Observing Post and Fire Control Center personnel on a computer based virtual environment. The system is HLA (High

[2] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

Level Architecture) based and works on LAN combining seven PC's. The federation comprises seven federates, namely, Forward Observing Post Federate, Fire Control Center Federate, Tank Federate, Infantry Federate, Mobile Obstacle Detachment Federate, Tank Platoon Federate and Trainer Federate.

Keywords : Distributed Interactive Simulation, Fire Control Center, Forward Observing Post, Mobile Obstacle Detachment, Tank Platoon.

1. GİRİŞ

Eğitim, başarının temel şartı olmakla birlikte, etkili eğitim; gerçek muharebe sahasına benzer arazi, tecrübeli personel ve bol miktarda mühimmat ve yakıta ihtiyaç göstermektedir. Azalan eğitim sahaları, sık personel değişimi, yüksek mühimmat ve akaryakıt fiyatları farklı askeri sınıflara ait personelin kendi görevleri ve müşterek yapılacak görevlerle ilgili konularda etkili bir eğitim almasını güçleştirmektedir. Harekat esnasında birlikte görev yapan personelin birlikte eğitiminin temsili bir ortam üzerinde, etkileşimli ve görsel içerikli bir simülasyon aracıyla desteklenmesinin maliyet-etkin bir çözüm olacağı yolundaki değerlendirme, Çevik Engel Müfreze Savunması ve Tank Takımı Taarruzu Simülasyonu için temel motivasyon olmuştur. Bu simülasyon, diğer simülasyonlarda olduğu gibi, gerçek eğitimin ikamesi kabul edilemez ve gerçek eğitimi kopya etmez. Yalnızca gerçek eğitime hazırlık sağlar ve onu zenginleştirir.

Sistem yedi federeden oluşan Yüksek Seviyeli Mimari (High Level Architecture-HLA) tabanlı ve dağıtık etkileşimli bir simülasyon uygulamasıdır. Federasyon senaryosu, tank takımı taarruzuna karşı çevik engel müfrezesi ve tanksavar timinin topçu ateş desteği ile yaptığı hazırlıksız savunmayı canlandırmaktadır. Sistem geliştirilirken, Topçu Yüzbaşı Hakan Aydemir tarafından geliştirilen Sahra Topçusu Atış Timi Simülasyonu [1] kapsamındaki İleri Gözetleyici ve AİM federeleri değişiklik yapılmadan yeniden kullanılmıştır.

Federasyonun koşturulması için aynı yerel ağ üzerine konuşlandırılmış yedi adet PC'ye ihtiyaç vardır. PC'lerden bir tanesi RTI servislerini sağlayan RTIExec programını çalıstıracak ve federeler sıra gözetmeksizin federasyona dahil olacaklardır [2]. 128 MB'lık bir ekran kartı ile yapılan çalısmada istenen gerçek zaman performansı yakalanmaktadır. Programlama aracı olarak Microsoft Visual C++ 6.0 ve koşturma ortamı olarak RTI NG 1.3 kullanılmıştır.

2. SİMÜLASYONUN KAPSADIĞI HAREKAT SAHASININ ANA BİLEŞENLERİ

Simülasyon, Mavi kuvvet olarak Topçu, Tank, Piyade ve İstihkam birliklerini, Kırmızı kuvvet olarak ise tank takımını kapsamaktadır. Birliklerin unsurları aşağıda açıklanmaktadır.

2.1. Topçu Birlikleri : Topçu Atış Sistemi, İleri Gözetleyici, Ateş İdare Merkezi ve Atış Bataryasından oluşmaktadır. Bu konuda detaylı bilgi [1] numaralı referansta verildiğinden burada tekrarlanmamıştır.

2.2. Tank Birlikleri : Simülasyonumuz karşı kuvvet olarak bir tank takım federesi ve dost kuvvetler içinde kullanılan bir tank federesini kapsamaktadır.

2.2.1. Tank Takımı : Simülasyonda karşıt kuvvet olarak, dört adet M48A5T1 muharebe alanı ana tankından oluşan bir tank takımı görev yapmaktadır. Bu tanklar taktik duruma göre belirli muharebe düzenlerini ve ilerleme tekniklerini uygulayarak muharebeye destek verirler [3].

2.2.2. Tank Mürettebatı : Tank mürettebatı dört kişiden oluşmaktadır: tank komutanı, şoför, doldurucu ve nişancı. Yalnızca tank komutanı ve nişancı interaktif kullanıcı (oyuncu) olarak simülasyona katılmaktadır.

2.3. Piyade Birlikleri : Piyade birliği olarak sadece tek er seviyesinde tanksavar silah nişancısı sisteme dahil edilmiştir. Personel RPG-7 ve LAW silahlarının nişancılığı konusunda eğitilirken, ileri gözetleyici gibi ateş isteğinde bulunma ve hedef tarifi konusunda da pratik kazanmaktadır [4].

2.4. İstihkam Birlikleri : Çevik Engel Müfrezesi (ÇEM), simülasyonda yer alan istihkam birliğidir. Müfreze, bir kamyon aracılığı ile sahip olduğu mayın, tel üstüvane ve çelik kirpi engellerini süratle araziye döşeyerek bir engel sistemi oluşturur. Böylelikle, hem şoförün, hem de engelleri döşeyen personelin eğitimi yapılmaktadır [5]. Şoför ve istihkam eri interaktif kullanıcılardır.

3. KAPSAM

Bir simülasyonun kalitesi onun gerçeğe ne kadar yaklaştığı, diğer bir deyişle sadakat seviyesi ile ölçülebilir [6]. Bu noktada özellikle liderlik, sevk ve idare gibi çok karmaşık yüzlerce bileşenden oluşan muharebe alanını simüle etmek oldukça zordur. Bu sebeple, simülasyonumuzda, federasyon senaryosu bakımından, nispeten az önemli olduğunu düşündüğümüz bazı faktörleri ya yok saydık ya da sabit kabul ettik.

Tank takım federesi özellikle tank takım komutanlarının, tank komutanlarının ve tank nişancılarının eğitimine yardımcı olacaktır. Çevik engel müfreze federesi küçük çaplı istihkam birliklerinin eğitimine katkı sağlayacaktır. Piyade federesi gerçek tanksavar silahları ile yapılacak atışlardan önce eğiticiler için iyi bir araç olacaktır. İleri gözetleyici ve ateş idare merkezi federesi topçu mürettebatının eğitimini destekleyecektir.

4. FEDERASYON YAPISI

Federasyon; İleri Gözetleyici Federesi [1], Ateş İdare Merkezi Federesi [1], Tank Federesi, Piyade Federesi, Çevik Engel Müfereze Federesi [7], Tank Takımı Federesinden [8] oluşmaktadır.

Eğitmen Federesi, eğitimin uygulayıcısı olan eğitim subayının oyuna müdahale etmesini sağlamaktadır. Bu federe senaryonun oluşturulmasından sorumlu federedir.

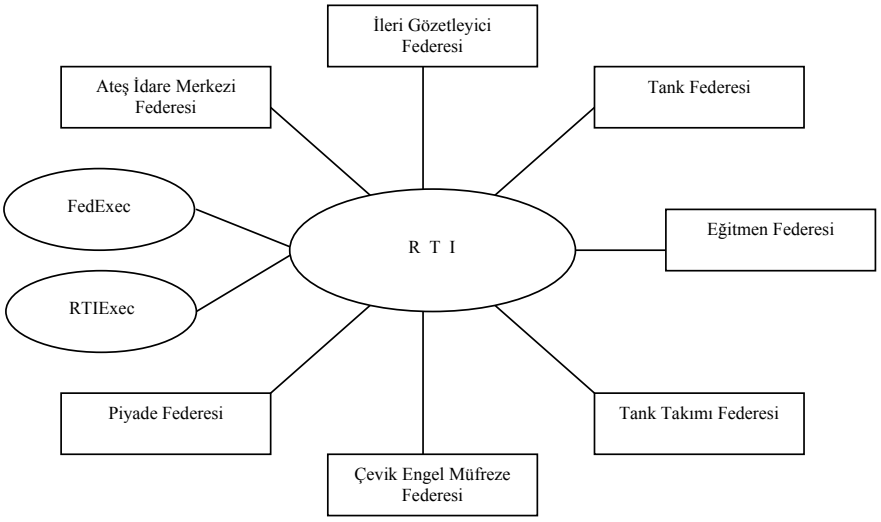
İleri Gözetleyici Federesi eğitilecek olan ileri gözetleyici tarafından kullanılacak ve hedefleri tespit ve atışları tanzim edecektir. Ateş İdare Federesi, İleri Gözetleyiciden aldığı bilgileri işleyerek atış değerleri elde edecek ve ayrıca bir Batarya Federesi olusturulmadığından atış işlemini gerçekleştirecektir.

Tank Takım Federesi, düşmanın görerek ve görmeyerek ateşleri altında ilerleyip mavi kuvvetler tarafından oluşturulan engel sistemi içerisinden geçerek hedefe ulaşmaya çalışacaktır. Tank Federesi, düşmanı ateş altına alarak zayiat verdirmeye, mümkünse imha etmeye çalışacaktır.

Piyade Federesi düşman tanklarını tanksavar silahıyla ateş altına alacak, yetersiz kaldığı zaman topçu birliğinden ateş desteği isteyecektir.

Çevik Engel Müfreze Federesi muharebe alanının emredilen kısımlarına sahip olduğu imkan ve kabiliyete göre engelleri döşeyecektir.

Federeler arasındaki veri akışı kablolu veya kablosuz yerel ağ üzerinde çalısan RTI aracılığı ile sağlanmaktadır. Federeler ihtiyaç duydukları özel verileri kendilerine ait veri tabanlarından alırlar ve paylaşılan verileri RTI aracılığıyla birbirlerine gönderirler. Federasyonun genel yapısı Şekil 1'de gösterildiği gibidir.



Şekil 1. Federasyon yapısına genel bakış.

Federeler tarafından kullanılan nesne sınıfları Batarya, Mermi, İleri Gözetleyici, Ateş İsteği, Atış Düzeltmesi, Tank, Tank Takımı, Piyade, İstihkam, Mayın, Çelik Kirpi, Tel Engel, Ortam ve Arazi olarak adlandırılmıştır. Nesne sınıflarının tanımları için [7,8] kaynaklarına başvurulmalıdır.

5. EĞİTMEN FEDERESİ

Eğitici, senaryonun oynanacağı arazi kesimini ve birliklerin başlangıç yerlerini, çevik engel müfrezesinin kullanabileceği engel sayısını, piyadenin atış yapacağı mühimmat miktarını belirlemekten ve kırmızı tank takımına hedefini göstermekten sorumludur. Eğitici engel sisteminde bulunan engellerin kırmızı tanklara yapacağı hasar ve zayiat oranlarını değiştirme olanağına sahiptir. Ayrıca eğitici, hedeflerin, ileri gözetleyicinin

ve topların yerini federasyon koşumu esnasında değiştirme yetkisine sahiptir. Eğitmen Federesi hava ve atmosfer (hava sıcaklığı, nem durumu, kar, yağmur vb.) (Şekil 2) şartlarını belirleyerek mermi yolu hesabında kullanılan değerleri ve kullanıcıların görüş mesafesini etkileyebilir. Eğitici, basit bir araç yardımıyla yükseklik haritası düzenleyerek arazinin oluşturulması imkanına da sahiptir.

a. Yağmur b. Kar c. Sis

Şekil 2. Hava ve atmosfer koşulları.

Eğitici, 3 boyutlu arazi verileri kullanılarak hazırlanmış olan iki boyutlu arazi haritası üzerinde, simülasyondaki birlikleri ifade eden askeri işaret ve semboller ile oyunun gelişimini ve safhalarını takip edebilir.

6. İLERİ GÖZETLEYİCİ VE AİM FEDERELERİ

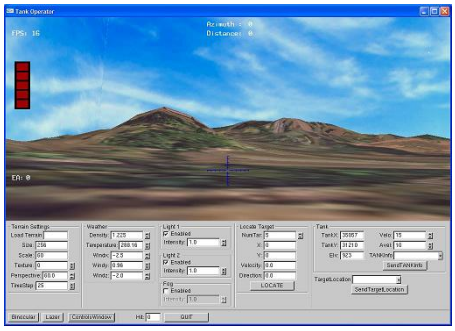
İleri Gözetleyici Federesinin iki görevi vardır. Biri hedef tespiti, diğeri atışlara uygulanacak düzeltmelerin hesaplanması ve bildirilmesidir. AİM Federesi kullanıcısı, İleri gözetleyicinin usulüne uygun olarak gönderdiği hedef bilgilerini, ileri gözetleyicilerin yerini, ve bataryanın yerini yatay plan kağıdına işleyerek hesaplamalarını yapmaktadır. Ayrıntılı bilgi için bakınız [1].

7. TANK FEDERESİ

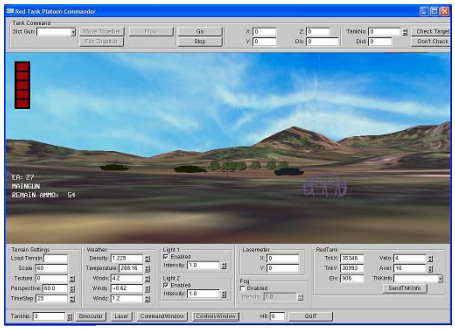
Bu simülasyonda, tek tanktan oluşan bir federe oluşturularak kırmızı tank takımı ile çarpışabilecek ve piyadeye ateş desteği sağlayabilecek bir ortam hazırlanmıştır. Şekil 3'te tank federesinin kullanıcı arayüzü görülmektedir.

8. TANK TAKIM FEDERESİ

Bu federe simülasyonda karşıt kuvvet görevini üslenmiştir. Amacı kendisine eğitici tarafından gösterilen hedefi ele geçirmektir. Şekil 4'de tank takım federesinin kullanıcı arayüzü görülmektedir. Bu federede takımı oluşturan dört tank tek kullanıcı tarafından sevk ve idare edilebilmektedir.



Şekil 3. Tank federesi kullanıcı arayüzü.



Şekil 4. Tank takım federesi kullanıcı arayüzü.

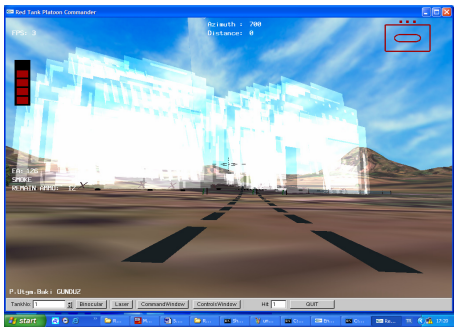
8.1 Hareket kabiliyeti: Tankların hareketini kontrol etmek maksadıyla üç değişik seçenek belirlenmiştir. Oyuncu bu seçeneklerden istediği herhangi birini veya birkaçını kullanabilir. Tank takım komutanı, 3 boyutlu arazi değerleri kullanılarak oluşturulan eğim haritası üzerinde takımına uygun ilerleme mihveri seçerek takımını hedefe yöneltebilir. Bu yöntem dışında tank takım komutanı istenilen ilerleme düzenini seçerek diğer tankların kendi tankına göre hareket etmelerini sağlayabilir. Üçüncü yöntem, her bir tankı tek tek ilerletmektir.

8.2 Tank atışı: Kullanıcı, ilk olarak tank ile hedef arasındaki mesafeyi ölçer. Sonra "Yükseliş Açısını Hesapla" fonksiyonu bulunan mesafe değerini alır ve önceden seçilmiş mühimmat cinsine göre, tank ile hedef arasındaki yüksekliği de hesaba katarak, bu mesafede tank topuna verilmesi gereken yükseliş açısını bulur. Bu fonksiyon tanklardaki "Balistik Hesaplayıcı" gibi çalışır ve balistik hesaplayıcılarda olduğu gibi ölçülen son mesafe yenisi girilene kadar hafızada kalır. Ateşlemeden sonra mermi yolu, balistik modele göre belirlenir.

8.3 Sorumluluk Sahaları: Hedeflerin daha iyi gözetlenmesi maksadıyla takım komutanı tanklara sorumluluk sahaları verebilir. Sorumluluk sahası, kullanıcı tarafından merkezi ve çapı belirlenen dairesel bir alan olarak tanımlanmaktadır. Eğer herhangi bir tanka verilmiş sorumluluk sahası sınırları içerisinden bir düşman tankı veya tanksavar

silahı ateş ederse, o sahadan sorumlu tank otomatik olarak hedef tespiti yapar ve hedefe ateş açar.

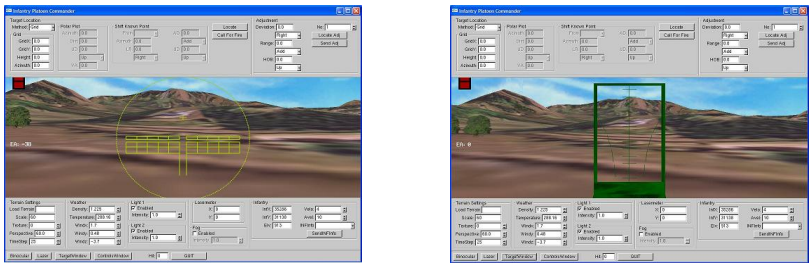
8.4 Engellerden Geçit Açma: Engellerden geçit açması maksadıyla takıma bir adet mayın pulluğu verilmiştir. Bu pulluk 3 numaralı tanka takılıdır. Takım, düşmanı körlemek maksadıyla sislerini kullanabilir. Geçit açma tankı geçit açarken geçtiği bölgeyi işaretler ve geçit açıldıktan sonra diğer tanklar bu işaretleri takip ederek ilerlerler (Şekil 5).



Şekil 5. Geçit Açma

9. PİYADE TAKIMI FEDERESİ

Bu çalışmada yalnızca piyadenin tanksavar silahı kullanımı ele alınmıştır. Kullanıcı RPG-7 ya da LAW silahlarından birisini seçebilmektedir. Ayrıca, bu federeyi kullanan piyade ileri gözetleyici gibi ateş isteği de yapabilmektedir. Şekil 6'da piyade federesinin kullanıcı arayüzleri görülmektedir.



RPG-7 LAW

Şekil 6. Piyade federesi kullanıcı arayüzleri.

10. ÇEVİK ENGEL MÜFREZE FEDERESİ

10.1. Arayüz :

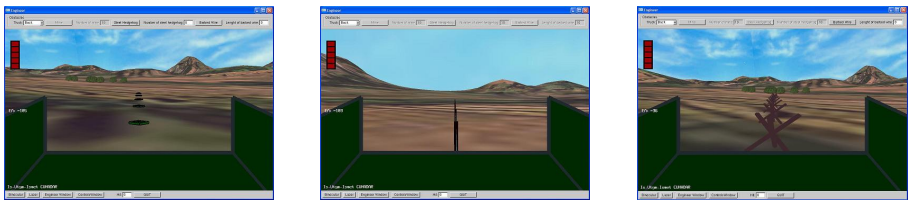
ÇEM iki arayüz kullanmaktadır: şoför için kamyonun şoför mahalli ve engel döşeyenler için kamyonun kasası (Şekil 7).

a. İç kısım b. Arka kısım

Şekil 7. ÇEM kullanıcı arayüzleri.

10.2. Engel İnşası:

Kamyon arka kısmının arayüzü aktifse engel döşenebilir. Bir seferde yalnızca bir tip engel döşenebilir (Standart telüstüvane, çelik kirpi ve M15 infilak etkili antitank mayını). O tipteki tüm engeller döşenince diğer bir tip engel araca yüklenip tekrar engel inşasına geçilebilir (Şekil 8). Mayına belli bir mesafe uzaklıktan geçen tank mayına bastı kabul edilmekte ve imha olmaktadır. Tel engel ve kirpi engellerine çarpan tankın her seferinde hızı düşmekte ve tecrübi bir çarpışma sayısına ulaşınca imha olduğu kabul edilmekte ve bu da smodele aktarılmaktadır.



(a) (b) (c)

Şekil 8. Engel inşası (a. mayın, b. tel engel, c. çelik kirpi).

11. FEDERASYON SENARYOSU

Simülasyonun temel senaryosu, hazırlıksız savunma yapan mavi kuvvetlere karşı, kırmızı tank takımının taarruzunun oynanmasını içermektedir. Kırmızı tank takımı simülasyonda kullanıcının başlangıçta oluşturduğu arazinin herhangi bir bölgesinde, mavi kuvvetleri temsil eden takviyeli bir piyade takımına taarruz etmektedir. Mavi takımında tanksavar piyade, çevik engel müfrezesi ve bir adet tank vardır. Ayrıca mavi kuvvetin genel desteğinde bir topçu bataryası bulunmaktadır.

Taarruz çıkış hattını geçerek başlayan kırmızı taarruzları, ilk olarak tank ve topçu birliğinden açılan görerek ve görmeyerek atışlarla yavaşlatılmaya çalışılmaktadır. Bu safhada ileri gözetleyici ve piyade topçudan ateş isteğinde bulunmaktadır. Bir sonraki aşamada kırmızı kuvvetler taarruz doktrinine uygun ilerleme düzen (kol, hat, kama, geniş kama, kademeli) ve tekniklerini (ilerleme, himayeli ilerleme, himayeli sıçrama) kullanarak hedefe doğru ilerlemektedir. Bu ilerleme, mavi kuvvetlere ait çevik engel müfrezesi tarafından döşenmiş engel sistemi ile karşılaşıncaya kadar devam etmektedir. Mavi kuvvetler, engel sistemi önünde yavaşlayan düşmanı imha etmek maksadıyla bu safhada, piyadenin elinde bulunan kısa menzilli tanksavar silahları dahil, sahip olduğu tüm ateş vasıtalarını kullanmaktadır (Şekil 9).

Şekil 9. Kırmızının Engel Sisteminde Durdurulması.

Engellerin önünde, mavinin yoğun ateşi altında kalan kırmızı tank takımı araziye mevzilenmektedir. Bütün tanklar mayın pulluğu takılı olan tankın engellerden geçit açmasına yardım etmek için, üzerilerinde bulunan sis tertibatını kullanarak engelleri koruyan mavi kuvvetleri körlemekte ve kırmızı tank takım komutanının, tank komutanlarına verdiği sorumluluk sahalarındaki düşmanla ateş muharebesine devam etmektedir.

Bu son safhada amaç, mavi kuvvetler için, önceden hazırladığı engel ve ateş planı ile kırmızı tank takımını durdurmak ve imha etmek, kırmızı kuvvetler için, zayiat vermeden engel sistemini aşarak hedefe ulaşmaktır.

12. TARTIŞMA VE SONUÇ

Geliştirilmiş olan dağıtık etkileşimli simülasyon sisteminin, gereken genişletmeler ve iyileştirmeler yapıldığı takdirde, kısıtlı bilgisayar olanaklarına sahip askeri birliklerde eğitim maksatlı olarak kullanılabileceği düşünülmektedir.

Mevcut sahra topçusu atış timine ait federelerin, daha geniş bir federasyon bünyesinde aynen kullanılmış olması, standartlara uyumlu ve modüler yaklaşımın yararını göstermektedir.

Harekat sahasının karmaşıklığı dikkate alındığında, mevcut sistemde pek çok yenilik ve ilerletme düşünülebilir. Bunlardan bazılarına aşağıda değinilmiştir.

- Hali hazırda arazinin üç boyutlu dijital şekli, bir ortam editörü kullanılarak bilgisayarda oluşturulmaktadır. Daha gerçekçi bir görsellik için, dijital arazi modeli veya dijital yükseklik modeli gibi gerçek arazi modelleri kullanılabilir.
- Tankın dinamik ve kinematik özellikleri simülasyona dahil edilebilir.
- Helikopter, uçak, zırhlı personel taşıyıcı gibi muharebe alanında olabilecek diğer unsurlar da simülasyona katılabilir.
- Daha detaylı bir kontrol ve planlama için tank takım federesindeki tanklar ayrı ayrı birer federe olarak gerçekleştirilebilir.
- Mühimmat çeşitleri ve zırh türleri üzerinde yapılacak çalışmalar ile hedef balistiği simülasyona dahil edilebilir.
- Daha gerçekçi bir hasar modeli için, vurma ve muharebe dışı kalma algoritmaları geliştirilebilir.

KAYNAKÇA

[1] Aydemir, H., Oğuztüzün, H. Topçu Atış Timi Simülasyonu. USMOS 2005, Haziran 2-3, 2005, Ankara.

[2] Kuhl, F., Weatherly, R., Dahmann, J. 'Creating Computer Simulation Systems: An Introduction to High Level Architecture', Prentice Hall, 2000.

[3] FM 17-15 Tank Platoon. Erişilen tarih 22 Mart 2007, www.fprado.com/armorsite/US-Field-Manuals/FM-17-15-Tank-Platoon.pdf

[4] FM 7-8 The Infantry Rifle Platoon and Squad, Erişilen tarih 22 Mart, 2007, www.globalsecurity.org/military/library/policy/army/fm/7-8/

[5] FM 5-100 Engineer Operations, Erişilen tarih 22 Mart 2007, www.globalsecurity.org/military/library/policy/army/fm/5-100/

[6] Durgun, G. (2004). Path Planning For a Tank Platoon. ODTÜ Enformatik Enstitüsü MODSİM Programı Bitirme Projesi Raporu, Ankara.

[7] Çuhadar, İ. (Haziran 2006). Mobile Obstacle Detachment Training Simulation. ODTÜ Enformatik Enstitüsü MODSİM Programı Bitirme Projesi Raporu, Ankara.

[8] Gündüz, B. (Haziran 2006). Tank Platoon Offensive Operations Training Simulation. ODTÜ Enformatik Enstitüsü MODSİM Programı Bitirme Projesi Raporu, Ankara.

BÜYÜK ÖLÇEKLİ AĞLARIN DEVS TABANLI MODELLENMESİ VE SİMÜLASYONU

Ahmet ZENGİN, Hüseyin EKİZ

Sakarya Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Eğitimi Bölümü

54187 Serdivan-SAKARYA

{azengin, ekiz}@sakarya.edu.tr

ÖZET

Dağıtık sistemler, çeşitli algoritmalar ve teknolojiler kullanarak birbirleriyle iletişim yapan birimlerden oluşur. İletişim içerisinde bulunan sistemlerin uyarlanabilirlik, ölçeklenebilirlik, güvenilirlik (sürdürülebilirlik) gibi bir takım niteliklere sahip olması gerekmektedir. Sürekli yeni servis türlerinin ve heterojen ağların bir bütün olarak dahil edilmesiyle, ağlar karmaşık bir hal almaktadır. Ağ sistemlerinin büyüyerek daha karmaşık bir hal alması karşısında yeni ve daha gelişmiş servisleri sunması beklenmektedir. Artan karmaşıklık ve boyut nedeniyle ortaya çıkan sorunların üstesinden gelmek amacıyla geliştirilen çeşitli yöntemler bilgisayar ağlarının ihtiyaçlarına göre kullanılmaktadır. Günümüzdeki ağlarda, hesaplama işlevleri hız ve işlem yapma gücündeki ihtiyaçlara cevap verebilmek amacıyla merkezi bir yapıdan dağıtık bir yapıya doğru kaymaktadır. Performans / maliyet oranının göz önünde tutulması zorunluluğu bu değişimi yeni işlem yapma kapasitelerinin tasarlanması üzerine daha fazla yöneltmektedir. Bu çalışmada, dağıtık sistemlerin karmaşıklık, ölçeklenebilirlik, yönetim vb. sorunlarının incelenmesi, tasarım alternatiflerinin araştırılması ve farklı çözüm yaklaşımlarının incelenmesi amacıyla modelleme ve simülasyon araçlarının kullanıldığı bir çalışma / uygulama gerçekleştirilmiştir.

Anahtar Kelimeler: Modelleme ve Simülasyon (M&S), DEVS, Ağ Yönetimi, Ekoloji, İnternet

DEVS BASED MODELLING AND SIMULATION OF LARGE SCALE NETWORKS

ABSTRACT

Network systems must communicate with one another using a variety of algorithms and technologies. Many systems supporting interconnectivity are required to exhibit essential traits such as adaptability, scalability, and reliability (survivability). At the same time, these networked systems are expected to offer new and more sophisticated services in the face of increasing system heterogeneity. Consequently, to cope with the management of such networks in the presence of ever increasing complexity, various decentralized and centralized approaches are being used to address the needs of private and public organizations. In this study, in order to examine some problems of distributed systems such as complexity and scalability, search for design alternatives

and inspect various solution strategies, an application is developed in which modeling and simulation methodologies and tools are used.

Keywords: Modeling & Simulation, DEVS, Network Management, Ecology, Internet

1. GİRİŞ

Bilgisayar ağları, yeni servis türlerinin ve heterojen ağların bir bütün olarak mevcut ağlara dahil edilmesiyle sürekli büyümekte ve gittikçe daha karmaşık bir hal almaktadır [1]. Ağ hızında ve işlem yapma gücündeki hızlı artış gereksinimi, ağ içerisinde gerçekleştirilen yönlendirme ve yönetim işlemlerinin, merkezi bir yapıdan dağıtık bir yapıya doğru kaymasını zorunlu kılmaktadır. Performans / maliyet oranının göz önünde tutulması zorunluluğu, araştırmacıları ağ üzerinde yeni işlem kapasitelerinin eklenmesi ve yeni yöntemlerin tasarlanması konusuna daha fazla yöneltmektedir. Bu yönelim, nesneye yönelik teknolojilerle birlikte yazılım endüstrisinde yoğunlaşmakta ve yazılım endüstrisindeki yoğunlaşma sonucunda nesneye yönelik eğilim birbirleri ile etkileşen yazılım bileşenlerinin modellenmesi ve tasarlanması popüler konular olarak ortaya çıkmaktadır.

Milyarlarca insanın global ağ içerisinde çalışan uygulamalara günlük yaşamlarının bir parçası olarak düzenli olarak erişecek olması, yakın gelecekte gerçekleşecek olaylardan birisi olacağı tahminleri yapılmaktadır [2].

İnternetin yakın gelecekte 1 milyar düğüme erişeceği düşünülürse, ölçeklenebilirlik kavramının ağların yönetiminde, modellenmesinde ve simülasyonunda yeni boyutlar kazanacağı açıktır. Statik topoloji üreten simülatörler (COMNET, NS2, OPNET, vb.) küçük ağları çalışmak için ideal platformlarken, günümüzde üstel olarak artan ağ sistemlerini modellemede ve değişken yapılı ağ sistemlerinin performansını test etmede yetersiz kalmaktadırlar. Ayrıca, bu simülatörlerin mimarilerinin bir çoğu soyutlama ve hiyerarşiden yoksun olmaları yanında çok büyük hesaplama maliyeti oluşturmaktadırlar [6][7][8]. Yapılan çalışmada, belirtilen kısıtlamaları / sakıncaları ortadan kaldırmaya yönelik olarak DEVS(Discrete Event System Specification) metodolojisi kullanılarak bir ağ simülatörü geliştirilmiştir.

Tasarlanan ağ sisteminin modellenmesi; ağ bileşenlerinin tanımlanmasını, bu bileşenlerde çalışacak yazılım nesnelerinin, etkileşimlerinin ve bu varlıkların işlem yapan düğümlere dağıtılmalarını, ağ topolojilerinin ve iletişim protokollerinin tanımlanmasını içermektedir. Düğümler ve linkler temel ağ bileşenleri olarak tanımlandıktan sonra, DEVS birleşik model tanımı kullanılarak temel bileşenler birbirine bağlanıp birleşik ağ modelleri oluşturulmuştur. Geliştirilen ağ ortamı farklı yönlendirme algoritmalarını (en kısa yol, uzaklık vektörü, oğul zekası, vb.) modelleyebilme yeteneğine sahiptir.

Geliştirilen simülatörün üstünlüklerini ve performansını göstermek amacıyla binlerce düğümden oluşan ağlar modellenmiştir. Modellenen ağlar farklı trafik yükleri altında çalıştırılarak, çalışma sırasında geliştirilen yönlendirme algoritmasının performansı incelenmiştir. Gerçekleştirilen uygulamalardan, geliştirilen simülatörün son derece paralel, esnek ve hızlı çalıştığını, değişik teknolojileri barındıran uygulamaları geliştirebilme yeteneğine sahip olduğu gözlemlenmiştir.

2. AYRIK OLAYLI SİSTEM (DEVS)

DEVS modelleme yaklaşımı, ayrık olaylı modelleme ve simülasyonda temel bir yaklaşım olarak Zeigler tarafından ortaya atılmıştır [3] [4]. DEVS yaklaşımının sadece ayrık olaylı modeller için değil, ayrık zamanlı ve diferansiyel denklemli sistemlerle ifade edilen davranışları uyarlamak için bir hesaplama temeli meydana getirmesi nedeni ile, modelleme ve simülasyon uygulamalarında DEVS teorisinin gücünü görmek önemlidir. Ayrık olaylı yaklaşımlar sınıfı içinde modeller, zamanın sürekli olduğu ancak sınırlı bir zaman periyodunda sonlu sayıda olayın meydana geldiği bir soyutlama seviyesinde tanımlanır [3]. Bu olaylar sistemin durum değiştirmesine neden olabilir. Olaylar arasında sistemin durumu kesinlikle değişmez. Bu nedenler Ayrık sistemler, sistemin durumunun zaman içerisinde sürekli değiştiği sürekli modellerden bu noktada farklıdır.

DEVS bir sistemin davranışını iki farklı seviyede ele alır: atomik ve birleşik. Atomik DEVS; en düşük seviyede, sıralı durumlar arasındaki geçişler gibi ayrık olaylı sistemin otonom davranışını, harici bir girişe (olaylar) nasıl tepki verdiğini ve çıkışı (olaylar) nasıl hesapladığını tanımlar. Birleşik DEVS; daha yüksek bir düzeyde, bir sistemi bileşenler ağı olarak tarif eder. Bileşenler, atomik DEVS modelleri ve birleşik DEVS modelleri olabilir. Bağlantılar, bileşenlerin birbirini nasıl etkilediğini gösterir.

Paralel bir atomik DEVS modeli aşağıdaki yapıdadır:

$$M=<X, S, Y, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta>$$

Burada;

X, giriş değerleri kümesi,
S, durumlar kümesi,
Y, çıkış değerleri kümesi,
δ_{int} : $S \rightarrow S$ *dahili geçiş* fonksiyonu,
δ_{ext} : Q x $X \rightarrow S$ *harici geçiş* fonksiyonu,
burada; $Q = \{(s,e) | s \in S, 0 \leq e \leq ta(s)$ *toplam durum* kümesi,
e, en son olan geçişten bu yana *geçen* süredir.
δ_{ext} : Q x $X \rightarrow S$ *çakışma (confluent) geçiş* fonksiyonu,
λ: $S \rightarrow Y$ çıkış fonksiyonu,
ta: $S \rightarrow R^{+}_{0,\infty}$ zaman ilerleme (time advance) fonksiyonu, 0 ve ∞ arasındaki pozitif reel sayılar kümesidir.

3. SWARMNET MODELLEME VE SİMÜLASYON ÇERÇEVESİ

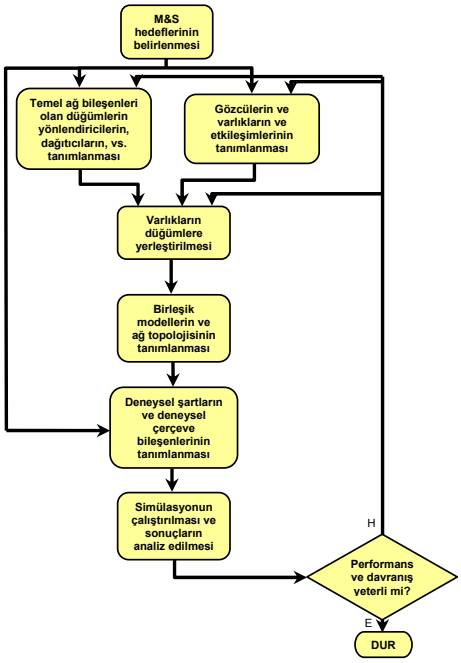
Bu bölümde, dağıtık bir ağ sistemini oluşturan bileşenler ayrık olaylı sistem tanımlama yaklaşımı - DEVS) kullanılarak tanımlanacak ve tanımlanan bileşenlerin davranışları detaylandırılacaktır.

3.1 Ağ Modelleme Süreci

Dağıtık ağ sistemlerinin tasarım ve analizini sistematik bir şekilde gerçekleştirmek için bu sistemlerin modellenmesi ve simülasyon işlemleri bir takım aşamalara bölünmelidir. Bu aşamalardan ilki modelleme ve simülasyon hedeflerinin belirlenmesidir (Şekil 1).

Modelleme hedefleri, modelin sistem tasarımında, yönetiminde ve kontrolünde tanımlanan rolü ile ilişkilidir. Hedefler ifadesi, model tasarlama işleminin belirli problemler üzerine odaklanması vazifesini görür. Önceden hedefler bilinirse, bu hedeflere uygun deneysel çerçeveler geliştirilebilir [3].

İkinci aşamada simülasyon bileşenlerinin tanımlanması işlemi gerçekleştirilir. Üçüncü aşamada, geliştirilen varlıklar ve düğümler birbirleri ile birleştirilir. Bu bileşenler DEVSJAVA modelleme ve simülasyon ortamında birbiriyle bağlanarak değişik topolojiler ve ağ konfigürasyonları meydana getirilebilir. Dördüncü aşamada, modelleme ve simülasyon hedeflerine ve geliştirilen modellere uygun bir deneysel çerçeve tanımlanır. Bu deneysel çerçeve modelin bir takım şartlar altında test edilmesine ve gözlem yapılmasına olanak tanır. Son aşamada, simülasyon çalıştırılır ve sonuçlar gözlemlenir. Eğer sonuçlar kabul edilebilir bir aralıkta ise simülasyon işlemi sonlandırılır. Aksi halde, tekrar başa dönülür ve geliştirilen modellerin parametreleri ayarlanır. Böyle bir süreçte ileri geri hareket edilerek en uygun model geliştirilmiş olur.



Şekil 1. Dağıtık bir ağın modelleme ve simülasyon süreci

3.2. Ağ Modelleme Yaklaşımı ve Ağı Oluşturan Bileşenlerin Tasarımı

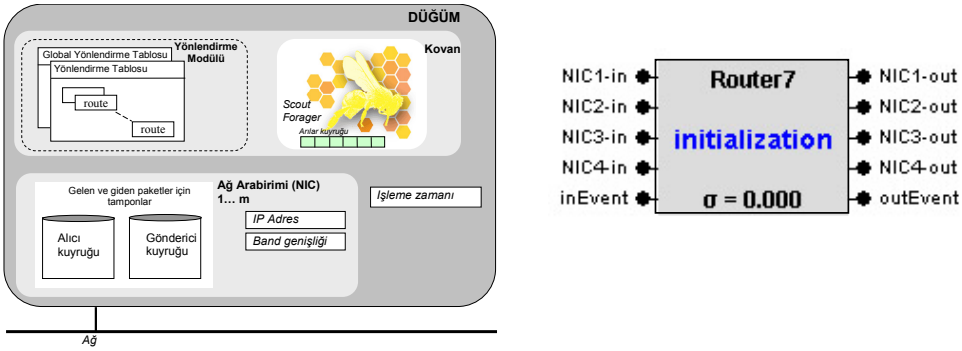
Dağıtık bir ağ sistemini modelleme amacına yönelik oluşturulan linkler, bu linkler aracılığıyla birbirleriyle haberleşen düğümler ve haberleşme aracı olan IP paketleri, 'temel ağ bileşenleri' olarak tanımlanmıştır. Diğer ağ bileşenleri (yönlendirme tabloları ve birleşik modeller) Java dilinde tasarlanmış ve gerçekleştirilmiştir.

3.2.1. Düğüm Atomik Modeli

Ağdaki her bir düğüm, aşağıda yapısı detaylı bir şekilde tanımlanacak olan paketleri işleyebilme ve paketleri uygun hedeflere yönlendirebilme yeteneğine sahip bir anahtarlama birimi olarak modellenmiştir. Modellenen düğümler iki veya daha fazla ağ bağlantısıyla birbirine bağlanan DEVS atomik modelleridir.

Şekil 2, üç adet modülden oluşan bir düğüm mimarisini ve ekran çıktısını göstermektedir. Bu modüller yönlendirme modülü, ağ arabirimi (Nerwork Interface Card - NIC) ve kontrol amacıyla bir takım varlıklar olarak gözcüleri ağ üzerine yayan bir kovandır.

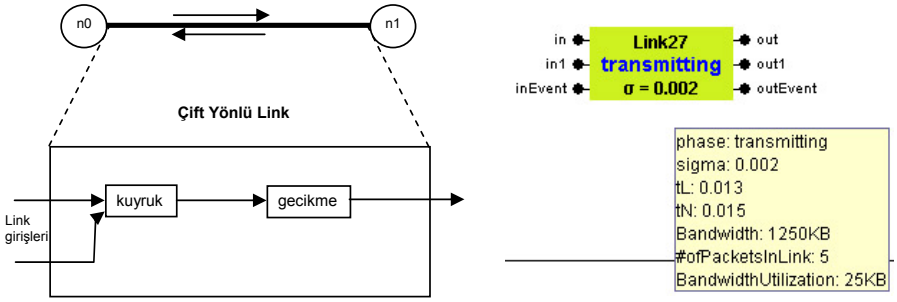
Yönlendirme modülü, bir başka değişle IP işleyici (IP Handler), yönlendirmenin yapıldığı birimdir. Her düğümde, paketler hedef düğümlerine yönlendirme modülü kullanılarak iletilir. Ağ arabirimi(Network Interface-NIC) ağlar arası temel iletişim hizmetleri sağlarken, arabirim ağın topolojisinin yönetiminden sorumludur. Ağ içinde her düğüm IP adresi olarak isimlendirilen bir tanımlayıcı etikete/kimliğe sahiptir. Düğümü oluşturan son modül olarak bir kovan sistemi, gözcüler olarak adlandırılan yapay görevli ve varlıkları üretme işlevine sahip olacak şekilde tasarlanmıştır ve sosyal böcek davranışının ve optimizasyon düzenlerinin esin kaynağı olduğu oğul zekası (swarm intelligence) kullanan yönlendirme algoritmalarının uygulanabilmesine, analizine, test edilmesine ve karşılaştırılmasına olanak tanımaktadır.



Şekil 2. Geliştirilen ağ düğümünün iç yapısı ve dört adet arabirime sahip yönlendirici modunda çalışan düğüm atomik modelinin ekran çıktısı.

3.2.2 Link atomik modeli

Linkler bir ağın temel bileşenlerinden birisidir. Belirli bir düğüm genellikle coğrafik olarak komşusu olan düğümlere linkler aracılığıyla bağlanır. Bütün linkler haberleşme kanalları ve dolayısıyla bant genişliği (bits/s) ve yayılım gecikmesi (s) ile karakterize edilen bit transfer hatları olarak görülebilir.

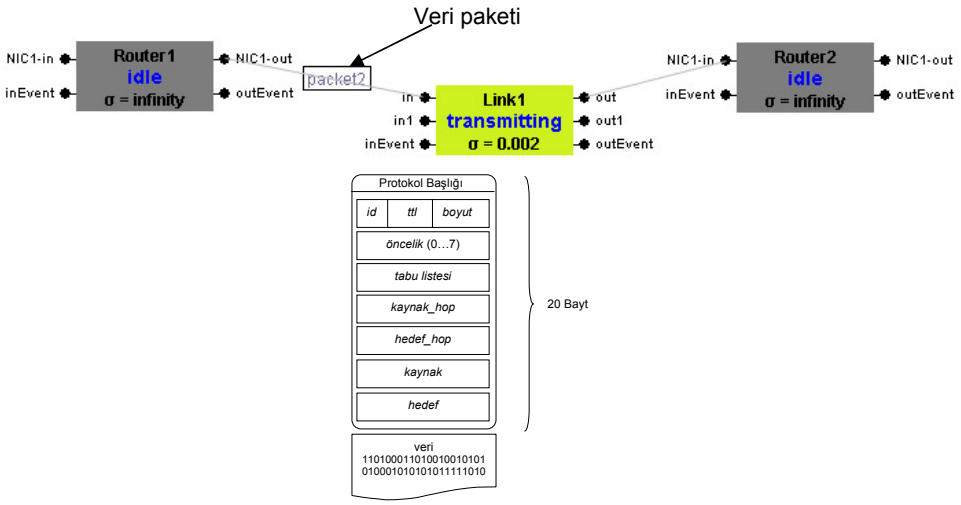


Şekil 3. Çift yönlü (dubleks) bir link modeli ve ekran çıktısı

Linkler her bir düğümün arabirimleriyle birleştirilerek farklı topolojide ağlar meydana getirilebilir (Şekil 3). Her link sınırlı kapasiteye sahip bir tampona sahiptir. Gelen paketler tamponda bekler ve ilk gelen ilk çıkar (FIFO) stratejisi veya diğer kuyruk yönetim yapıları kullanılarak bir sonraki düğüme iletilir. Linkler çift yönlü olarak tasarlanmaları nedeni ile aynı anda her iki yönde trafik akışı mümkündür. Linkler toplam kapasitelerine kadar belirli bir bant genişliğinde trafiği taşıyabilir.

3.2.3. Ağ paketleri

Ağı oluşturan bileşenler arasında DEVS mesajları şeklinde karşılıklı değişilen paketler, *veri* ve *kontrol* paketleri olarak ikiye ayrılabilir. Veri paketleri, bilgi taşıyan basit IP paketleri olarak düşünülebilir. Kontrol paketleri ise, ağ yönetim protokollerinin temelini oluşturan ağ izleme ve kontrol işlevini yerine getirme vazifesi görürler. Bütün uygulamalarda veri ve kontrol paketleri temel olarak aynı yapıya sahip olmalarına ve aynı bilgiyi taşımalarına karşılık, farklılıkları tanımlanan davranışlarındadır.



Şekil 4. Bir veri paketinin düğümler arasında hareketi ve IP paket modeli.

Şekil 4, bir paketin düğümler arası hareketini ve ağdaki bütün tipteki paketler için genel bir paket modelini göstermektedir. Paket modeli bir takım değişken alanlara sahiptir: *id*,

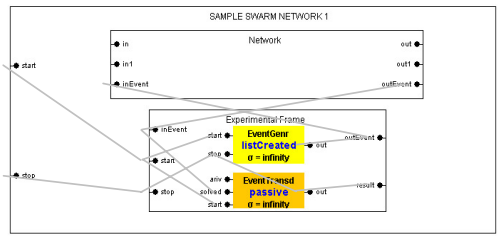
ttl, *paket boyutu*, *öncelik*, *tabu listesi, kaynak adresi*, *hedef adresi*, *kaynak_hop adresi, hedef_hop adresi ve veriler.* Veri hariç diğer bütün değişkenler *Protokol Başlığını (Protokol Header)* oluşturur ve uygulamada bütün paket türleri için 20 Bayt'lık standart bir boyuta sahiptir. Paketin toplam boyutu paket verisinin boyutuna göre değişebilir ve paket boyutu değişkeninde saklanır.

3.2.4. Yönlendirme tabloları

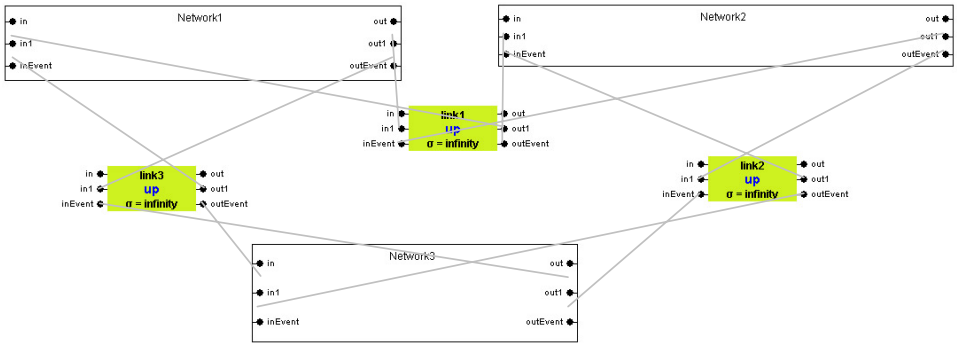
Her düğüm, ağdaki her bir hedef düğümün temsil edildiği bir yönlendirme tablosuna sahiptir. Bu yönlendirme tablosu komşu düğümlerin sütunları, hedef düğümlerin ise satırları temsil ettiği bir matris olarak düşünülebilir (yani satırlar hedeflere, sütunlarda komşulara karşılık düşer). Her bir düğümü bir yönlendirme tablosuyla donatarak veri paketleri sistematik olarak ağ boyunca yönlendirilebilir. Bir düğüm bir paketi belirli bir hedefe doğru iletmeye ihtiyaç duyduğunda, hangi gidiş linkinin (veya DEVS atomik model çıkış portu) kullanılacağı kararı yönlendirme tablosunda belirtilen bilgiye dayanılarak verilir.

3.3. SwarmNet Deneysel Çerçevesi ve Ağ Trafik Modeli

Yapılan çalışmada, ağın performans kriterlerini ölçmek için DEVS deneysel çerçeve bileşenleri *SwarmNet* çerçevesine göre tanımlandı. Deneysel çerçeve, en genel biçiminde sisteme muhtemel girişleri tanımlayan bir *üreteç*, çıkış işlemini tarif eden bir dönüştürücüden (örneğin; performans ölçütlerini hesaplama gibi) ve sistemin eşleştiği durumları (mantıksal ifadeleri) tanımlayan bir onaylayıcıdan oluşur. Yapılan çalışmada kullanılan deneysel çerçeve uyarlaması, bir *olay üreteci (event generator)* ve *olay dönüştürücüden (event transducer)* oluşur (Şekil 5).



Şekil 5. Bir ağın deneysel çerçeveyle bağlantısı.



Şekil 6. Kümeleme yöntemine göre bağlanmış 3 adet ağ.

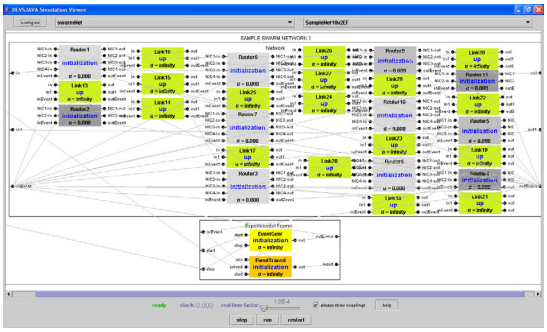
3.4. Ölçekleme

Büyük ölçekli ağların modellenmesinde karşılaşılan en büyük problem ölçeklenebilirliktir. Doğru bir ölçeklenebilirlik için bir kümeleme sistemi uygulanabilir (Şekil 6). Kümeleme bir alt ağı daha üst seviye bir ağ içerisinde bir düğüme indirgeyerek yönetilebilir ağ boyutları sağlar. Bu durumda, tüm düğümlerin toplamında bir ağ hiyerarşisi meydana gelir. Kümelere ayrılmış bir ağda düğümlerin adreslenmesi çoğu kez ağların hiyerarşisini yansıtır.

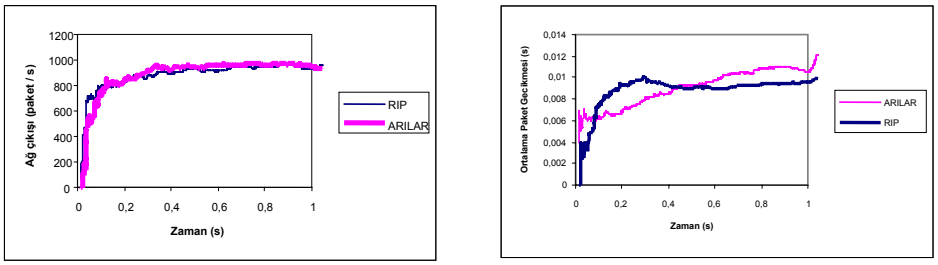
DEVS birleşim altında kapalılık özelliği birleşik bir modeli atomik bir model olarak ele aldığı için, DEVS birleşik modeli kümeleme yöntemiyle benzerlikler taşır. DEVS modelleme yönteminin hiyerarşik ve modüler yapısı, kümeleme yönteminin uygulanmasını kolaylaştırır. Yapılan uygulamada kümeleme düğümlerin adresleme seviyesinde halledilir ve her bir birleşik model bir takım '*sınır düğümlere'* sahiptir. Bu sınır düğümleri diğer birleşik modellere bağlanmak için kullanılır. Sınır düğümleri içinde küme isimlerinin bulunduğu ilave bir yönlendirme tablosuna sahiptir. Bu yöntem ağ düğümleri içinde kayıtlı bulunan yönlendirme veritabanının boyutunu önemli ölçüde düşürür.

4. SWARMNET MODELLEME VE SİMÜLASYON ORTAMINDA YÖNLENDİRME ALGORİTMALARININ UYGULANMASI

Yapılan çalışmada, geliştirilen '*SwarmNet'* modelleme ve simülasyon ortamının çalışmasını ve yönlendirme algoritmalarının bu ortamda uygulanabilirliğini göstermek amacıyla örnek çalışmalar gerçekleştirilmiştir. Yapılan çalışmanın ilk aşamasında; en yaygın kullanılan klasik yönlendirme algoritmalarından biri olan *'yönlendirme bilgi protokolü'* (Routing Information Protocol - *RIP*) algoritmasının '*SwarmNet'* içinde modellenmesi ve simülasyonu gerçekleştirildi. Daha sonra oğul zekası tabanlı bir algoritma geliştirilerek uygulanmıştır. Şekil 7'de deneylerde kullanılan ağın modeli deneysel çerçeve ile birlikte görülmektedir.



Şekil 7. Basit bir ağın deneysel çerçeve ile birlikte DEVSJAVA ekran görünümü.



Şekil 8. RIP ve Arılarla gerçekleştirilen algoritmalarının ağ çıkış değerlerinin karşılaştırılması.

Yapay zeka alanı, farklı mühendislik, yönetim, kontrol ve hesaplama problemlerini çözmek için biyolojik bilgi ve tekniklerden faydalanır [5]. Doğal sistemler, basit yapılı organizmaların, dinamik bir şekilde birbirleriyle etkileşerek son derece karmaşık görevleri yerine getirebilen sistemlerin oluşturabileceğini göstermektedir. Örnek olarak; grup halinde yaşayan canlıların etkileşiminden ortaya çıkan oğul zekası, bilgisayar ağlarının yönetimi gibi karmaşık optimizasyon problemlerinin çözümünde esin kaynağı olmuştur. Oğul zekası kullanan yönlendirme algoritmaları, ortak bir hedefi gerçekleştirmek üzere etkileşen varlıkların işbirliği sonucu dağıtık ve dinamik bir sistem olan ağın çeşitli problemlerini çözebilmektedir. Geliştirilen modelleme ve simülasyon çerçevesinde bal arılarının davranışlarından üretilen basit kurallar ağ yönlendirme işlemine uygulanmıştır. Elde edilen sonuçlar daha önceki uygulamaların sonuçlarıyla karşılaştırılmaktadır (Şekil 8). Deneyler sonucunda elde edilen grafiklerden, ekoloji tabanlı algoritmanın RIP algoritmasına göre daha iyi performans gösterdiği görülmektedir

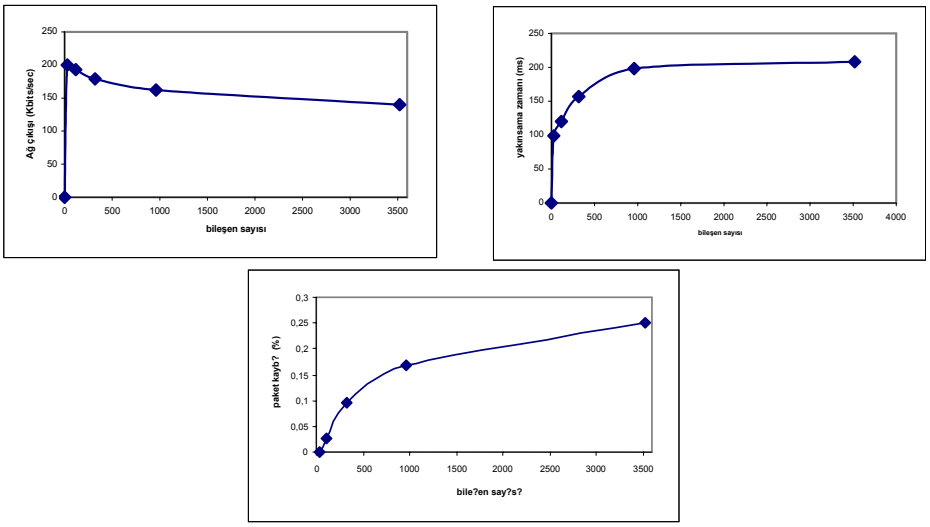
5. BÜYÜK ÖLÇEKLİ AĞLARIN İNCELENMESİ

Bölüm 3.4'tede ifade edildiği gibi büyük ölçekli ağların modellenmesinde karşılaşılan en büyük problem ölçeklenebilirliktir. Düğümlerde yönlendirme işleminde kullanılmak üzere bulunan veritabanı ağın boyutuyla artacağından, büyük bir ağın yönlendirme veritabanı bir düğümün işleyebileceğinden çok daha fazla olur. Bu durumda yönlendirme veri tabanının uygun bir basitlikte yeniden yapılandırılması gerekmektedir. Sınır düğümleri içinde küme isimlerinin bulunduğu ilave bir yönlendirme tablosuna

sahiptir. Bu yöntem ağ düğümleri içinde kayıtlı bulunan yönlendirme veritabanının boyutunu önemli ölçüde düşürür.

Tablo 1. Büyük ölçekli ağ modelleri

Ağ adı	Bileşen sayısı	Koloni sayısı
NET 1	116	4
NET 2	319	11
NET 3	960	87
NET 4	3520	125



Şekil 9. Ağ performans kriterlerinin bileşen sayısıyla değişimi.

Tablo 1'de verilen ağ modellerinin simülasyon sonuçları Şekil 9'daki grafiklerde gösterilmektedir. Grafiklerden, bileşen sayısı artarken ağ performans kriterlerinin kademeli olarak düştüğü görülmektedir. Ağ çıkışının düşüş göstermesinin temel sebebi; büyük ölçekli ağlarda oluşan trafik yüküne paralel olarak paket kaybının daha fazla olmasıdır. Ancak büyüyen ağ boyutlarında oluşan kayıplar ve performans düşümleri kabul edilebilir sınırlardadır [9].

6. SONUÇLAR

Bu çalışmada dinamik ağ sistemlerine ve ağ sistemlerinin yönetimine odaklanan bir modelleme ve simülasyon uygulaması gerçekleştirilmiştir. Dağıtık sistemler, DEVS modelleme ve simülasyon metodolojisi kullanılarak yazılım ve donanım parçalarıyla birlikte modellenmiş, daha sonra özel uygulamalar gerçekleştirilerek geliştirilen modelin geçerlenmesi ve doğrulanması sağlanmıştır. Geliştirilen modeller Java programlama dilinde kodlanmış ve ortaya çıkan uygulama programı *SwarmNet* olarak adlandırılmıştır. Geliştirilen uygulamalar ve modeller zeki bileşenler içeren dağıtık sistemler gibi karmaşık yapıları ve sistemleri kapsamaktadır. Geliştirdiğimiz modelleme ve simülasyon ortamı, ölçeklenebilir, adaptif ve sağlam ağ uygulamalarının

modellenmesi ve tasarımı için örnek bir çerçeve olmuştur. Teorik modelleme ve simülasyon bilgisinin pratiğe dökülerek uygulanması, ülkemizde son derece zayıf olan modelleme ve simülasyon biliminin uygulamalı olarak kavranmasını ve öğrenilmesini sağlamıştır. Bu çalışma, DEVS metodolojisinin geniş bir uygulama alanına uygulanabilirliğini desteklemiştir.

KAYNAKÇA

[1] A. S. Tanenbaum, *Computer Networks*, Prentice-Hall Inc., 1996.

[2] Zeigler, B.P., Mittal, S., Modeling and Simulation of Ultra-large Networks: A Framework for New Research Directions, supported by NSF Grant ANI-0135530, ULN Workshop, July 2002 (addendum to the ULN Workshop 2001) http://www.acims.arizona.edu/EVENTS/ULN/ULN_doc2.pdf

[3] Bernard P. Zeigler. Theory of Modelling and Simulation. Robert E. Krieger, Malabar, Florida, 1984.

[4] Bernard P. Zeigler. Multifacetted Modelling and Discrete Event Simulation. Academic Press, London, 1984.

[5] Panta Lucic and Dušan Teodorovic Transportation Modeling: An Artificial Life Approach Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02) 1082-3409/02 $17.00 © 2002 IEEE

[6] Zengin, A., Ekiz, H., Firat, C., The High Level Architecture (HLA) For Distributed Simulations, 3rd International Symposium on Intelligent Manufacturing Systems, August 30 . 31, 2001, Sakarya

[7] Zengin, A., Ekiz, H., Fırat, C., Köker, R., Paralel ve Dağıtık Ayrık Olaylı Simülasyonlar ve Karşılıklı Çalışabilirlik, ELECO 2002, 2002, Bursa

[8] Zengin, A., Ekiz, H., Dağıtık Simülasyon Uygulamaları ve Yüksek Seviyeli Yapı (HLA), Elektrik - Elektronik Bilgisayar Mühendisliği 9.Ulusal Kongre ve Sergisi, 19 - 23 Eylül 2001, Kocaeli

[9] Zengin A., Sarjoughian H., Ekiz H., "Biologically Inspired Discrete Event Network Modeling", ESS2004, Budapest / Hungary, 2004

.

B1 OTURUMU:

FİZİK TABANLI MODSİM-I

3-BOYUTLU PALETLİ ARAÇ DİNAMİĞİ SİMÜLASYONU
Kemal ÇALIŞKAN, Samim ÜNLÜSOY

BALİSTİK BİR FÜZENİN MODELLENMESİ VE OTOPİLOT TASARIMI
Zekiye Bilinç TATAS, Yakup ÖZKAZANÇ

FST: HİDROFONUN YER DEĞİŞTİRMESİNİN HESAPLANMASI İÇİN BENZETİM TABANLI SONOBOY MEKANİK TASARIM ARACI
Mehmet HAKLIDIR

UÇUŞ KONTROL SİSTEMİ TASARIMI VE GELİŞTİRİLMESİ İÇİN SİMÜLASYON TABANLI TEST PLATFORMU YAZILIMI F-SIM
Ebru ARSLAN, Erkan ARIKAN, İlkay YAVRUCUK

3-BOYUTLU PALETLİ ARAÇ DİNAMİĞİ SİMÜLASYONU

Kemal ÇALIŞKAN [(a)], Y. Samim ÜNLÜSOY [(b)]

[(a)] ODTÜ Makina Müh. Böl., ckemal@metu.edu.tr

[(b)] ODTÜ Makina Müh. Böl., unlusoy@metu.edu.tr

ÖZ

Bu çalışmada, üç boyutlu paletli araç dinamiğinin değişik zemin tipi ve yol yüzeylerinde analizi için geliştirilen bir simülasyon modeli sunulmaktadır. Performans ve seyir konforu analizlerinde; araç gövdesi, süspansiyon kolları ve tekerleklerden oluşan çok gövdeli bir model ile güç aktarım elemanlarının modellerinden oluşan bütünleşik bir yapı kullanılmıştır. Paletler elastik bant olarak modellenmiş, paletleri tekerleklere sardırmak için ayrıntılı bir algoritma geliştirilmiştir. Zemin ve palet tarafından tekerleklere uygulanan kuvvetlerin hesaplanmasında, tekerlek-palet sisteminin direngenliği, zemin özellikleri ve değişken palet gerginliği dikkate alınmıştır. Araca uygulanan tahrik kuvveti, güç aktarım sisteminden gelen kuvvet ile palet-zemin arasında oluşan kuvvetin karşılaştırılması ile elde edilmiştir. Dönme hareketinin simülasyonu için, sağ-sol palet hızlarını girdi olarak alan bir model hazırlanmıştır. Hazırlanan modeller kullanımı kolay bir simülator arayüzü içerisine yerleştirilmiş, simülasyon sonuçları iki ve üç boyutlu animasyonlarla görselleştirilmiştir.

Anahtar Kelimeler: Paletli Araç, Performans, Seyir Konforu, Yönlendirme, Elastik Bant Palet Modeli

3-D TRACKED VEHICLE DYNAMICS SIMULATION

ABSTRACT

In this paper, a simulation model developed for 3-D tracked vehicle dynamics analysis is presented. The performance and ride analysis are conducted using a structure composed of a multibody model of the hull, roadarms and roadwheels, and models of power train components. The tracks on both sides of the vehicle are treated as flexible bands and a detailed algorithm is developed for wrapping the tracks around roadwheels, idler and sprocket. Forces exerted on the wheels by terrain and track are computed considering stiffness of the wheel-track system, terrain properties and dynamic track tension. The tractive force utilized for traction of the tracked vehicle is computed from comparison of the traction from the power train components and available traction from the terrain. For simulation of steering, a model that takes left and right track speeds as inputs is developed. The simulation models are embedded in a user friendly simulator interface and simulation results are enhanced with 2-D and 3-D animations.

Keywords: Tracked Vehicle, Performance, Ride, Steering, Flexible Track Model

1. GİRİŞ

Paletli araçların değişik yol yüzeyleri üzerinde hareketlerinin analiz edilmesi için yapılan çalışmaların uzun bir geçmişi olmasına rağmen, özellikle araç-palet-yol etkileşimlerinin karmaşıklığı ve gelişen bilgisayar donanım ve yazılımlarının da etkisiyle bu alandaki çalışmalar daha gerçekçi modeller elde etmek yolunda hızla devam etmektedir. Paletli araç dinamiği simülasyonun gerçeğe yakın şekilde gerçekleştirilmesi;

- tasarım parametrelerinin araç dinamiği üzerine etkilerinin anlaşılması ve tasarım parametrelerinin optimizasyonu ile,
- gerçekçi operasyon planları yapılması ve belirli bir görev için en doğru aracın seçilmesi konularında büyük yarar sağlayacaktır.

Paletli araç dinamiğinin analizine yönelik çalışmalar genellikle araç dinamiğinin belirli bir alanına (performans, seyir konforu ya da yönlendirme) yöneliktir. Fakat, aracın herhangi bir yol yüzeyi üzerinde karmaşık bir hareket esnasında göstereceği davranışın belirlenmesi için bütünleşik bir modele ihtiyaç vardır.

Bu çalışmada, burulma kollu süspansiyona sahip paletli araçlar için, araç dinamiğinin tüm alanlarını birlikte ele alan, çeşitli yumuşak veya sert zeminlerde paletli araç hareketini gerçekçi biçimde canlandırabilen bir model geliştirilmesi amaçlanmıştır. Bu doğrultuda, performans ve seyir konforu analizlerinde kullanılacak tek bir model, ve yönlendirme analizlerinde kullanılacak ayrı bir model oluşturulmuştur.

2. PERFORMANS VE SEYİR KONFORU MODELİ

Şekil-1'de, model şeması verilen paletli araç performans ve seyir konforu modeli üç ana modülden oluşmaktadır:

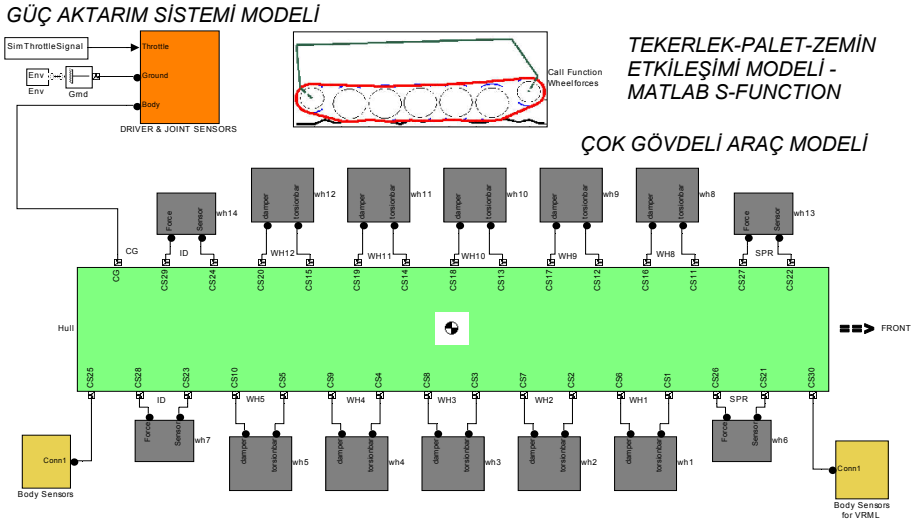
i. Gövde, süspansiyon burulma kolları, sönümleyiciler, tekerlekler, tahrik dişlileri, ve avara tekerlekten oluşan çok gövdeli model.

ii. Dizel motor, tork dönüştürücü, otomatik vites kutusu ve diferansiyel modellerinden oluşan güç aktarım sistemi.

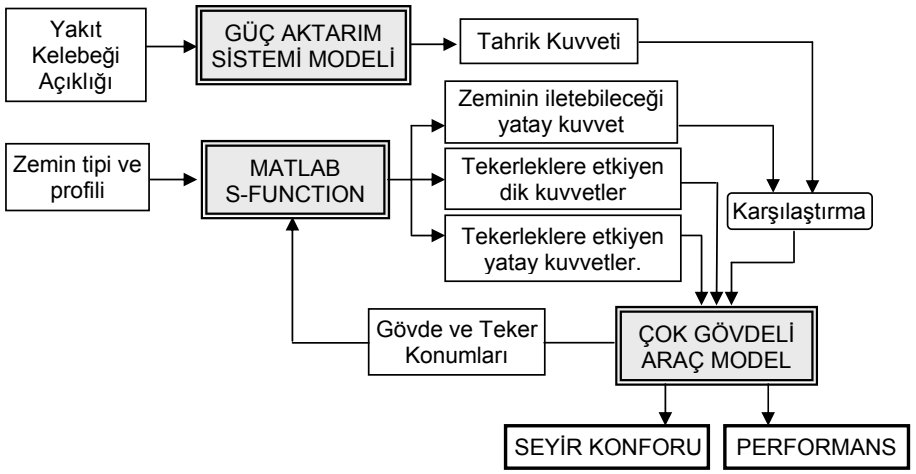
iii. Tekerlek-palet-zemin etkileşimini modelleyen bir algoritma.

Paletli bir aracı hareket ettiren kuvvet; güç aktarım sisteminin tahrik dişlisine ilettiği tahrik torkuna, palet-zemin temas yüzeyinde iletilebilecek yatay kuvvete ve direnç kuvvetlerine bağlıdır. Araca iletilebilecek tahrik kuvveti zemin-palet arasında oluşabilecek yatay kuvvetle sınırlıdır ve palet-zemin arasındaki kaymaya bağlıdır.

Şekil-2'de performans ve seyir konforu modelinin veri akış şeması gösterilmiştir. Analiz yüzey tipi ve geometrisi ile yakıt kelebeği açıklığı girdileriyle başlamaktadır. Motordan tahrik dişlisine iletilen tahrik torku güç aktarım hattı modeliyle hesaplanır. Zeminin taşıyabileceği kayma yükü, tekerleklere uygulanan dikey ve yatay kuvvetlerle birlikte, tekerlek-palet-zemin etkileşimini çözmek için geliştirilen algoritma (Matlab S-function [1]) ile hesaplanır.



Şekil 1. Performans ve Seyir Konforu Model Şeması

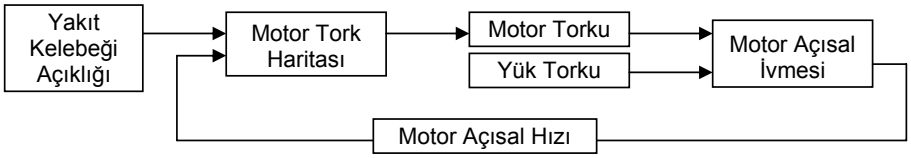


Şekil 2. Performans ve Seyir Konforu Modeli Akış Şeması

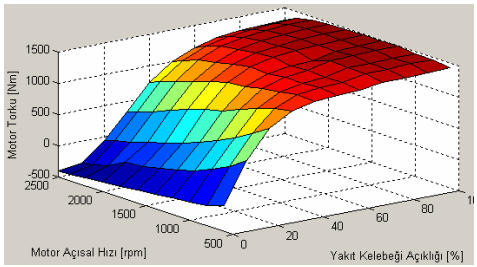
3. GÜÇ AKTARIM SİSTEMİNİN MODELLENMESİ

Güç aktarım sistemi dizel motordan başlayarak sırasıyla tork dönüştürücü, vites kutusu, diferansiyel ve palet tahrik dişlilerinden oluşmaktadır. Dizel motor modeli, yakıt kelebeği açıklığını girdi olarak alır ve motor ivmesi ile motorun ürettiği torku hesaplar (Şekil-3). Motor ivmesinin integrali alınarak motor hızı hesaplanır. Yük torku, tork dönüştürücüden gelen tork ile motor aksesuarlarına harcanan yükün toplamıdır.

Motorun belirli bir yakıt kelebeği açıklığı ve motor hızında verebileceği tork miktarı hız-yakıt kelebeği açıklığı-tork haritasından (Şekil-4) aradeğerleme yapılarak bulunur.



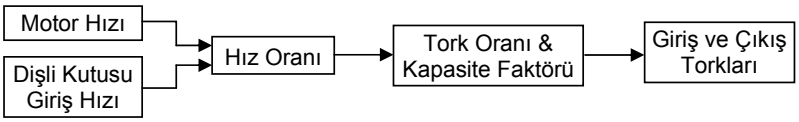
Şekil 3. Motor Modeli Veri Akış Şeması



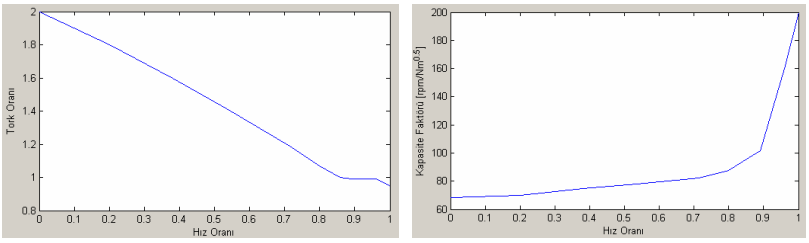
Şekil 4. Örnek Motor Tork Haritası

Tork dönüştürücü matematiksel modeli 1, 2 ve 3 nolu denklemler kullanılarak oluşturulmuştur. Denklemlerde SR, TR ve CF sırasıyla hız oranı, tork oranı ve kapasite faktörünü, *i* and o alt simgeleri ise giriş ve çıkışı temsil etmektedir. Modelin veri akış şeması şekil-5'te verilmiştir. Tork dönüştürücünün giriş ve çıkışındaki hızlar bilindiğinde, tork oranı ve kapasite faktörü karakteristik eğrilerden (şekil-6) bulunur.

$$SR = \frac{\omega_o}{\omega_i} \qquad TR = \frac{T_o}{T_i} \qquad T_i = \left(\frac{\omega_i}{CF_i}\right)^2 \qquad (1), (2), (3)$$

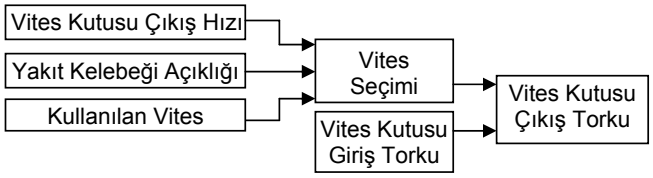


Şekil 5. Tork Dönüştürücü Modeli Veri Akışı

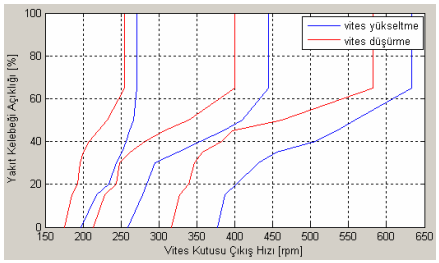


Şekil 6. Örnek Tork Dönüştürücü Karakteristikleri

Otomatik vites kutusunun giriş şaftı tork dönüştürücüye, çıkış saftı ise diferansiyale bağlıdır. Dişli kutusu modelinde veri akışı şekil-7'da gösterilmiştir. Vites seçimi vites kutusu çıkış hızı, yakıt kelebeği açıklığı ve aracın bulunduğu vitesi göz önüne alarak şekil-8'de örneği gösterilen şema yardımıyla yapılır. Vites kutusundan çıkan torkun diferansiyel tarafından iki palet tahrik dişlisine eşit olarak paylaştırıldığı varsayılmıştır.



Şekil 7. Otomotik Vites Kutusu Modeli Veri Akış Şeması



Şekil 8. Örnek Vites Değiştirme Şeması

4. TEKERLEK-PALET-ZEMİN ETKİLEŞİMİNİN MODELLENMESİ

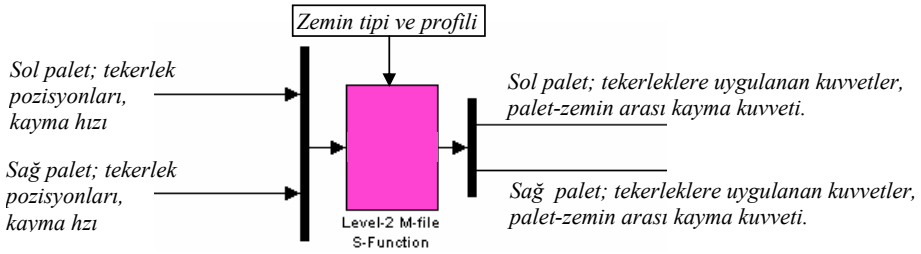
Tekerlek-palet-zemin etkileşimini analiz etmek için oluşturulan Matlab S-function;

i. yol profilini dikkate alarak paletin tekerlekler üzerine sarılması,

ii. tekerleklere zemin ve palet tarafından uygulanan yatay ve dikey kuvvetlerin hesaplanması,

iii. palet-zemin temas yüzeyinde oluşan kayma gerilmesi dağılımının bulunması

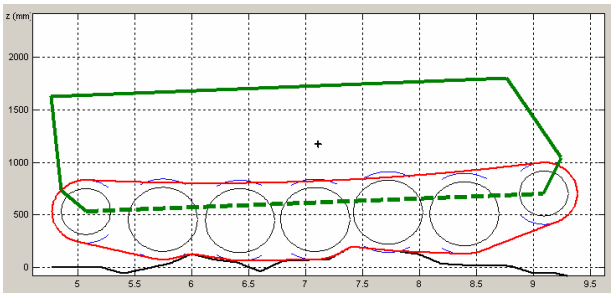
için kullanılmaktadır. S-Function veri akış şeması şekil-9'da verilmiştir.



Şekil 9. Tekerlek-palet-zemin etkileşimi için veri akış şeması

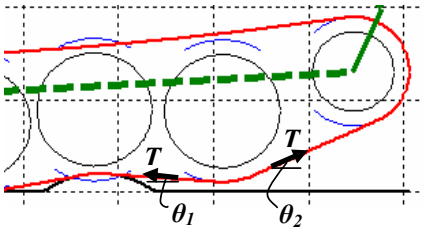
4.1. Paletin Tekerleklere Sarılması ve Tekerleklere Uygulanan Palet Kuvvetleri

Paletin tekerlekler çevresine sarılması ve palet tarafından tekerleklere uygulanacak kuvvetlerin bulunması, paletli araç dinamiğinin en zor konularından biridir. Bu çalışmada paleti modellemek için elastik bant modeli kullanılmıştır [2]. Palet belirli sayıda parçaya bölünerek palet şekli paletin tahrik dişlisine temas ettiği en uç noktadan başlayarak bir sonraki tekerlek konumunu ve aradaki zeminin profilini dikkate alarak nokta nokta belirlenmiştir. Sekil-10'da paletin bozuk yol üzerinde aldığı şekil görülmektedir.



Şekil 10. Bozuk ve sert bir zeminde palet şekli

Palet şekli tespit edildikten sonra, paletler tarafından tekerleklere uygulanan kuvvetler (şekil-11) bulunur.



Şekil 11. Paletin tekerleklere uyguladığı kuvvetler

Palet gerginliğinin paletin aldığı şekle göre değişimi göz önüne alınmıştır. Dinamik palet gerginliğini hesaplamak için palet sarkması ikinci dereceden polinom olarak modellenmiş, sarkma miktarı-palet gerginliği ilişkisi simülasyondan önce belirlenmiştir. Simülasyon sırasında bulunan sarkma miktarından palet gerginliği hesaplanmıştır.

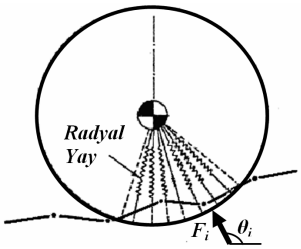
4.2. Zemin Tarafından Tekerleklere Uygulanan Kuvvetler

Sert zeminlerde yalnızca tekerlek-palet sisteminin esnediği varsayılmıştır. Tekerleklere uygulanan kuvvetleri bulmak için, tekerlek temas yüzeyi boyunca radyal parçalara bölünmüş (şekil-12) ve her bir radyal parça üzerine düşen kuvvet parçanın deformasyonu ile eşdeğer elastikiyet katsayısının çarpımı ile bulunmuştur [3].

Yumuşak zeminde tekerlek-palet sistemi rijit kabul edilmiş ve yalnızca zeminin deforme olduğu varsayılmıştır. Zemin deformasyonu-dik basınç bağıntısı zemin tipine göre farklılık göstermektedir. Örnek olarak, toprak için bu bağıntı denklem-4'teki

gibidir. Denklemde p dik basıncı, z deformasyonu, k ve n zemin sabitlerini temsil etmektedir. Uygulanan basıncın palete dik olduğu varsayılmıştır.

$$p = k \cdot z^n \tag{4}$$



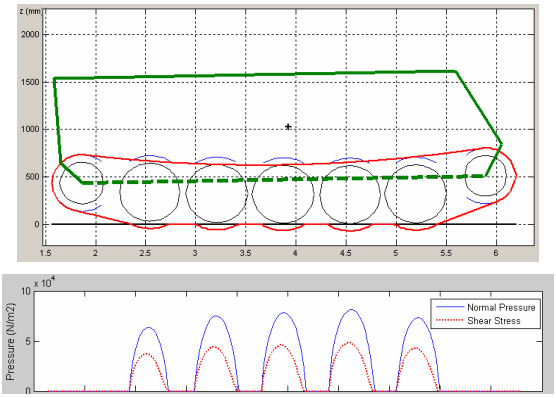
Şekil 12. Tekerlek-Palet Deformasyonu İçin Radyal Yay Modeli

4.3 Palet-Zemin Temas Yüzeyinde Oluşan Kayma Kuvveti

Yumuşak zeminlerde paletle zemin arasında oluşan kayma kuvvetinin belirlenmesi için Bekker [4] tarafından ortaya konmuş olan kayma yerdeğiştirmesi-kayma gerilmesi bağıntısı kullanılmıştır. Toprak zemin için bu bağıntı denklem-5'te verildiği gibidir [5].

$$s = (c + p\tan\phi)\left[1 - \exp(-\frac{j}{K})\right] \tag{5}$$

Denklemde s kayma gerilmesi, p dik basıncı, j kayma yerdeğiştirmesini, ve c, φ, K zemine özel katsayıları ifade etmektedir. Şekil 13'te yumuşak bir zeminde palet altında oluşan dik ve kayma gerilmesi dağılımları görülmektedir.



Şekil 13. Palet Altında Oluşan Dik ve Kayma Gerilmelerinin Dağılımı

Kayma gerilmesinin palet boyunca elde edilmesinin ardından, palet-zemin yüzeyinde oluşan toplam kayma kuvveti, palet genişliğinin b olarak ifade edildiği denklem-6 ile bulunur.

$$F_s = b\int_0^L s(x) \tag{6}$$

Palet altındaki bir noktanın kayma yerdeğiştirmesi, aracın denklem-7 ile verilen kayma oranına bağlıdır. Denklemde ω ve R tahrik dişlisi dönme hızı ve yarıçapını, V ise araç hızını temsil etmektedir.

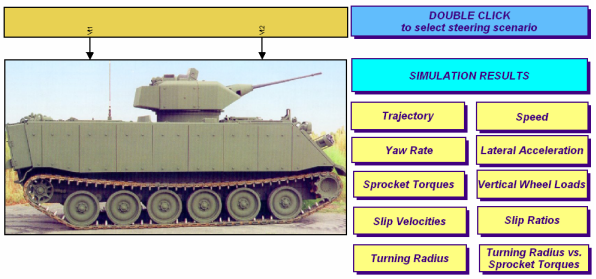
$$\text{S}=\textit{kayma oranı}=\frac{|\omega R - V|}{\omega R} \tag{7}$$

Sert zeminde kayma kuvveti palet ile zemin arasındaki kayma oranına bağlıdır. Bu durumda kayma kuvveti denklem-8 ile bulunur. Denklemde μ_0 ve K zemine özel katsayıları, P dik basıncı, S ise kayma oranını temsil etmektedir [6].

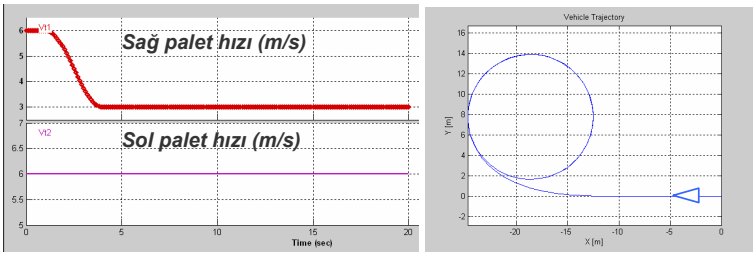
$$F_s = \mu_0 P(1 - e^{KS}) \tag{8}$$

5. YÖNLENDİRME ANALİZİ

Yönlendirme analizinde, sağ ve sol paletlere verilen farklı hızlar sonucunda aracın dinamik davranışı incelenmiştir. Analiz sonucunda aracın izlediği yol, yanal ivme, sapma açısı, tekerleklere uygulanan dik kuvvetler ve dönme hareketi için gerekli olan tahrik dişlisi torkları hesaplanmıştır. Yönlendirme analizinde paletle zemin arasında oluşan kayma kuvvetinin hesaplanması için literatürde bulunan farklı sürtünme modelleri kullanılmıştır [6, 7]. Şekil 14'de yönlendirme analizi modelinin arayüzü görünmektedir. Örnek analizler olarak, şekil 15'te her iki palet 6m/s hızla giderken sağ palet hızının 3m/s düşürüldüğü durumda aracın izleyeceği yol görülmektedir.



Şekil 14. Yönlendirme analizi model arayüzü



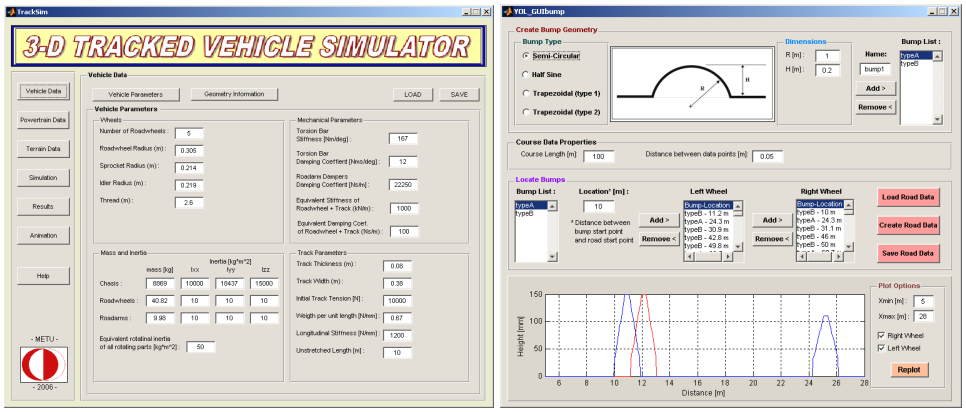
Şekil 15. Örnek yönlendirme simülasyonu

6. SİMÜLATOR ARAYÜZÜ

Önceki bölümlerde anlatılan paletli araç modelinin simülasyonu için kullanımı kolay bir arayüz oluşturulmuştur (şekil-16i). Arayüz, tüm araç ve güç aktarım sistemi

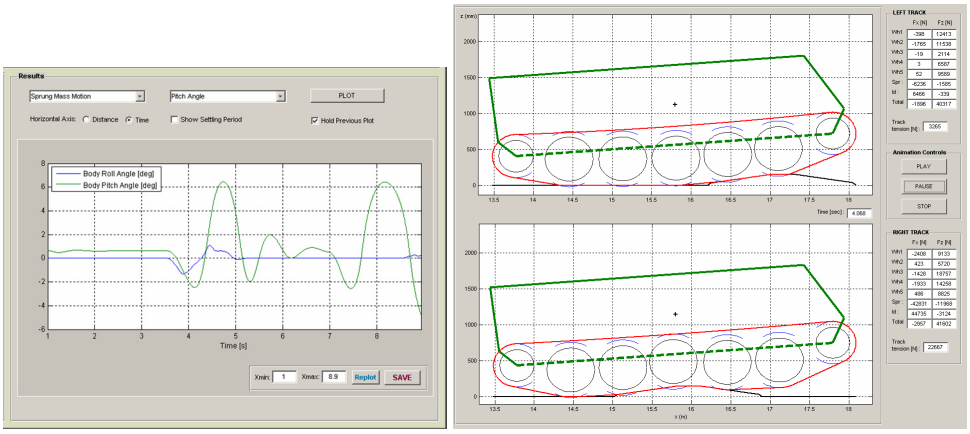
parametrelerinin tanımlanması, yol profili ve zemin tipi seçimi, simülasyon parametrelerinin seçimi ve simülasyon kontrolu, sonuçların grafiklerle görüntülenmesi ile 2 ve 3 boyutlu animasyonların hazırlanması aşamalarında büyük kolaylık sağlamaktadır. Hazırlanan simülasyon modeli bir çok araç tasarım parametresinin araç performansı üzerine etkisini incelemekte kullanılabilir. Bu parametrelerden bazıları; araç ağırlığı ve ataleti, tekerlek sayısı ve çapları, tekerleklerin konumları, süspansiyon direngenliği ve sönümleme katsayısı, başlangıç palet gerginliği, palet boylamsal direngenliği, palet genişliği, palet-tekerlek sisteminin direngenliği ve güç aktarım sistemi elemanlarının karakteristikleri olarak sıralanabilir.

Simülator arayüzünü kullanarak üç tip genel amaçlı yol profili yaratılabilir; kasisli yol, sinüs dalgası profilli yol ve gelişi güzel profilli yol.

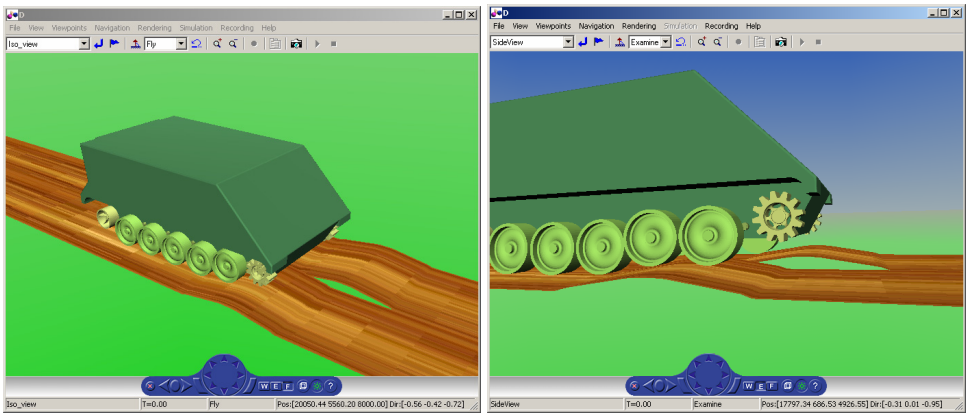


Şekil 16. i. Simülator Arayüzü ii. Kasisli Yol

Sonuçlar hem grafiklerle, hem de 2 ve 3 boyutlu animasyonlarla sunulmuştur (şekil 17 ve 18). Grafik sonuçlarda araç gövdesi ve süspansiyonların her türlü hareketi ve üzerlerinde oluşan kuvvetler ile güç aktarım sistemi elemanlarının hız ve tork bilgilerine ulaşılabilir.



Şekil 17. i. Grafik Sonuçlar ii. İki boyutlu animasyon



Şekil 18. Üç Boyutlu Animasyon

6. SONUÇLAR

Sunulan simülasyon modeli ve simülator arayüzü, burulma kollu süspansiyona sahip paletli araçların üç boyutlu dinamik analizler yoluyla tasarım ve performans kestirimlerinin yapılabilmesi amacına yönelik olarak geliştirilmiştir. Simülatör programı bir çok tasarım parametresinin araç davranışı üzerine etkisinin incelenmesinde, bu parametrelerin eniyilenmesinde ve mevcut araçların değişik zemin tiplerinde ve yüzey profillerinde kabiliyetlerinin sınanmasında kullanılacaktır.

TEŞEKKÜR

Yazarlar çalışmanın ilk aşamasında ki destekleri için FNSS Savunma Sistemleri A.Ş. ye teşekkür ederler.

KAYNAKÇA

[1] MATLAB Ver. 7.1, The Mathworks Inc., 2005.

[2] Sandu, C. and Freeman, J. S., "Connectivity Algorithm for an Extended Rubber-Band Track Model", Heavy Vehicle Systems, v. 9, n. 4, pp. 333-353, 2002.

[3] Dhir, A. and Sankar, S., "Analytical Track Models For Ride Dynamic Simulation of Tracked Vehicles", Journal of Terramechanics, v. 31, n. 2, pp. m Dynamics, v. 37, n. 6, pp. 401-421, 2002.

[4] Bekker, M.G., (1960), "Off-the-Road Locomotion", The University of Michigan Press, Ann Arbor.

[5] Janosi, Z. and Hanamoto, B., (1961), "Analytical Determination of Drawbar Pull as a Function of Slip for Tracked Vehicles in Deformable Soils", Proceedings of the 1st International Conference on Terrain-Vehicle Systems, Turin.

[6] Kitano, M., and Kuma, M., "An Analysis of Horizontal Plane Motion of Tracked Vehicles", Journal of Terramechanics, v. 14, n. 4, pp.211-225,1977.

[7] Wong, J.Y., Chiang, C.F., (2001), "A General Theory for Skid Steering of Tracked Vehicles on Firm Ground", Proc. Instn. Mech. Engrs., v. 215, n.3.

BALİSTİK BİR FÜZENİN MODELLENMESİ VE OTOPİLOT TASARIMI

Zekiye Bilinç TATAS [(a)], Yakup ÖZKAZANÇ[(b)]

(a) HAVELSAN A.Ş, Eskişehir Yolu 7.Km, 06520, ANKARA

btatas@havelsan.com.tr

(b) Hacettepe Üniversitesi Elektrik ve Elektronik Mühendisliği Bölümü

06800, Beytepe, ANKARA

yakup@ee.hacettepe.edu.tr

ÖZ

Balistik füzeler itme vektörü kontrolü ile kontrol edilen, yüksek irtifa ve hız değerlerine ulaşabilen füzelerdir. Bu bildiride balistik füze kontrolü için belirlenen yöntemler ve bu yöntemler kullanılarak elde edilen sonuçlar özetlenecektir. Kontrol yöntemleri sunulmadan önce, SCUD tipi bir balistik füzeye ait hareket denklemleri ve balistik füzeyi oluşturan füze dinamik ve kinematik modelinden kısaca bahsedilmektedir. Doğrusal kontrol teknikleri kullanılarak tasarlanan füze otopilotu ve oluşturulan modeller tek bir simülasyonda birleştirilerek, füze uçuşu incelenmiş, farklı senaryolar kullanılarak, uygulanan kontrol tekniğinin, balistik füze kontrolü için uygun yapıda olup olmadığı sınanmıştır.

Anahtar Kelimeler: Balistik Füze, Balistik Füze Otopilotu, Missile DATCOM, SCUD Füzeleri, Uçuş Kontrol Sistemleri

BALLISTIC MISSILE MODELLING AND AUTOPILOT DESIGN

ABSTRACT

Ballistic missiles usually controlled by thrust vector control and can reach high altitudes and high velocities. Methods determined for ballistic missile control and the results taken by using these methods will be summarized in this paper. Before examining the control methods, the equations of motion for a SCUD type ballistic missile will be derived and missile dynamics and kinematics model will be summarized briefly. Missile autopilot model which is designed by using linear control techniques and derived models will be integrated into a simulation. Model missile flight will be examined and by using different scenarios functionality of the applied control techniques for ballistic missile will be examined.

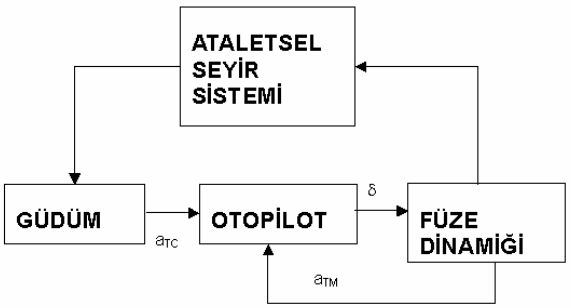
Keywords: Ballistic Missile, Ballistic Missile Autopilot, Flight Control Systems, Missile DATCOM, SCUD Missiles

1. GİRİŞ

Balistik füzeler genellikle itme vektörü kontrolü (thrust vectoring) ile yönlendirilen, yüksek irtifa ve hız değerlerine ulaşabilen güdümlü füzelerdir. Bu bildirinin amacı balistik füze kontrolü için belirlenen yöntemleri etüd etmektir. Bu amaca yönelik örnek balistik füze olarak SCUD-B füzesi seçilmiş, bu füzeye ait özellikler kullanılarak füze aerodinamik katsayıları Missile DATCOM programı vasıtası ile hesaplanmıştır. Füze dinamiği ve kinematik denklemleri incelenmiş, bu denklemler kullanılarak füze yunuslama ve dönme otopilotları için gerekli denklemler türetilmiştir. Doğrusal kontrol yöntemlerinden yararlanılarak, balistik füze için yunuslama ve dönme otopilotları tasarlanmıştır. Tasarlanan otopilot modelleri, füze dinamiği ve kinematik blokları, doğrusal olmayan 6-DOF bir uçuş simülasyonunda bir araya getirilerek, füze uçuşu ve kullanılan uçuş kontrol tekniklerinin başarımı incelenmiştir.

2. BALİSTİK FÜZE KONTROLÜ

Balistik füzeler, çok yüksek hızlara ulaşabilen, dünya üzerinde yerçekimi etkisi ile eliptik yörünge izleyerek çok uzun mesafelere savaş yüklerini büyük hassasiyetle ulaştıran, güdümlü füzelerdir [3, 7]. Balistik füze kontrolü füze güdümü ve otopilotu tarafından gerçekleştirilir. Ataletsel seyir sisteminden gelen füze bilgileri (füze hızı, konumu gibi) füze güdümüne gider. Füzenin belirlenen uçuş yolu boyunca uçmasını sağlayacak güdüm komutları, güdüm bloğunda oluşturularak, füze otopilotuna gönderilir. Füze güdümü ve kontrol yapısı Şekil 1. de gösterilmiştir.



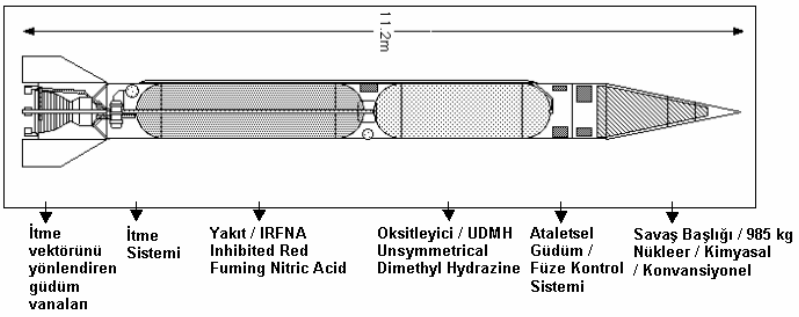
Şekil 1. Füze Güdümü ve Kontrolü Yapısı

Şekil 1. de, a_{TC} güdümden gelen ulaşılması istenen ivme, a_{TM} füzenin sahip olduğu ivme ve δ füze itme vektörü yönlendirme açısını göstermektedir.

Füze otopilotu, güdüm ve füze dinamiğinden gelen bilgileri kullanarak, eksenlere (dönme ve yunuslama eksenleri) göre füzenin kontrol edilmesini ve sistemin kararlılığını sağlar. Balistik füzelerde, kontrol yöntemi olarak, itme vektörü kontrolü kullanılır. Bu yöntemde; otopilotta hesaplanan itme vektörü yönlendirme açıları, füze dinamiğine verilerek (füze egzosunda (plume) bulunan kapakçıklar sayesinde itme yönlendirilir) füzenin dönme, yunuslama ve yalpa açılarındaki sapmaları engellenir ve sistemin kararlılığı temin edilir.

3. SCUD-B FÜZESİ

Bu bölümde, [3, 7, 8, 9] kaynaklarından yararlanılarak, SCUD-B füzesi ile ilgili genel bilgiler verilmiştir. SCUD-B, kısa menzilli, sıvı yakıtlı, tek savaş başlığına sahip balistik bir füzedir. SCUB-B füzesi yapısı Şekil 2.de görülmektedir.



Şekil 2. SCUD-B Füze Yapısı

SCUD-B 11.25 m uzunluğunda, 0.88 m çapında ve 5900 kg ağırlığındadır. Yakıt olarak UDMH (Unsymmetrical Dimethyl Hydrazine) ve IRFNA (Inhibited Red Fuming Nitric Acid) kullanmaktadır. Bu yakıtlar yanma döngüsüne, yakıt pompaları aracılığı ile gönderilir ve kuvvetli bir itme gücü sağlar. Füzenin fırlatma esnasındaki toplam itici ağırlığı 3771 kg' dır. Füze motoru deniz seviyesinde 130 kN' luk itme sağlar. Füzenin yapısal ağırlığı (savaş başlığı hariç) 1102 kg'dır. Füze güdümü, 3 tane dönüölçer kullanan basit bir ataletsel güdüm tekniği ile yapılmaktadır. Dönüölçerlerden kontrol sistemine gelen bilgi, kontrol sisteminden, motor egzosunda bulunan vanalara iletilir ve bu vanalar sayesinde füzenin uçuş yolu ayarlanır. Kontrol vanaları sadece motor yanma süresi boyunca (ilk 65 saniye) çalışır. SCUD-B için nükleer, kimyasal, yüksek patlayıcılı olmak üzere farklı savaş başlıkları geliştirilmiştir. Savaş başlığı füzenin burnunu oluşturan 2.87 m uzunluğundaki bölümde bulunmaktadır ve 985 kg ağırlığına sahiptir.

SCUD-B füzesine ait minimum menzil 50 km maksimum menzil ise 300 km'dir. Füze minimum ve maksimum menzilleri için sırası ile 24 km ve 86 km maksimum yüksekliğe çıkmakta ve uçuşunu 165 saniye ve 313 saniyede tamamlamaktadır.

1962'de kullanıma giren SCUD-B Avrupa ve Orta Doğuda bir çok ülke tarafından kabul görmüş, 1973'de Mısır tarafından İsraile karşı kullanılmıştır.1980–1988 yılları arasındaki 8 yıl savaşları boyunca İran ve Irak birbirlerine karşı, 600'den fazla SCUD-B füzesi kullanmıştır. Raporlar Mısır, İran, Kuzey Kore ve Suriye gibi ülkelerde SCUD-B füzesi geliştirme çalışmalarının devam ettiğini göstermektedir [3].

4. FÜZE DİNAMİĞİ VE KİNEMATİK MODELİ

Füze dinamiği füzenin hareketini ve füzenin kontrol sistemine verdiği yanıtı belirler. Füze dinamiği ve kinematik modelinde, 6-DOF doğrusal olmayan bir model temel

alınmıştır. İtme kontrollü füzeler için oluşturulan füze dinamik ve kinematik modeli için ivme ve moment denklemleri türetilmiştir. Bu denklemler oluşturulurken kullanılan hareket denklemlerinin türetilmesinde [2] numaralı kaynaktan yararlanılmıştır.

Füze üzerindeki kuvvet ve momentler; itme sistemi kuvvetinden kaynaklanan ivme ve momentler, aerodinamik kuvvetlerden kaynaklanan ivme ve momentler ve yerçekimi ivmeleridir. İtme kontrollü füzelerde itme kuvveti yunuslama (δ_{pitch} açısı ile) ve dönme (δ_{yaw} açısı ile) eksenlerinde yönlendirilebilir (yalpa kontrolü yapılmamaktadır). İtme sistemi kuvvetinden kaynaklanan ivme ve momentler kaynak [6]'da verildiği gibi alınmıştır. Yerçekiminden kaynaklanan ivmeler, kaynak [2]'de kullanılan ivme denklemleridir. Aerodinamik kuvvet ve moment katsayıları Missile DATCOM yazılımı [1] kullanılarak SCUD-B tipi bir füze için farklı yükseklik ve mach (hız) değerlerinde hesaplanmıştır.

Füze ekseni üzerindeki toplam ivmeler, sırasıyla denge ekseni ve rüzgar ekseni ivmelerine çevrilmiştir. Bu çevirimde kaynak [2] den faydalanılmıştır. Rüzgar ekseni üzerindeki ivme denklemleri kullanılarak füzeye ait füze toplam hızı, hücum açısı ve yanakayma açısı türetilmiştir. Füze toplam hızı, hücum açısı ve yanakayma değerleri kullanılarak füze ekseni hızları hesaplanmıştır.

Denge eksenindeki momentlerden faydanılarak, füze açısal hızları ve Euler Açıları türetilmiştir. Euler Açıları füze ekseni üzerindeki hızların, Kuzey, Doğu, Aşağı (North-East-Down) hızlarına çevrilmesinde kullanılmaktadır. Füze kuzey, doğu, ve aşağı hızlarından, füzenin kuzey, doğu, ve aşağıda aldığı yol hesaplanabilir. Böylece füze üzerindeki ivme ve moment denklemlerinden füzenin gittiği yörünge bulunmuş olur.

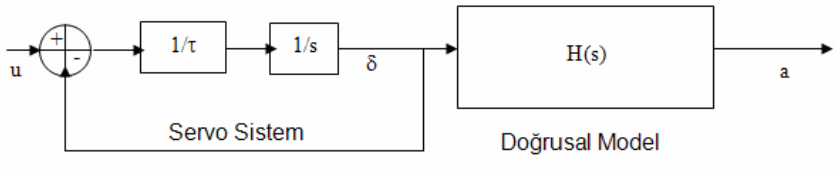
SCUD-B füzesi için geliştirdiğimiz 6-DOF, doğrusal olmayan model yer kısıtlaması nedeniyle burada verilmemiştir. İlgili model için kaynak [5]' e bakılabilir.

5. OTOPİLOT TASARIMI

Balistik füzelerde itme vektörü kontrolü uygulanmaktadır. Otopilot güdümden gelen ivme komutunu, füzeyi yönlendirecek olan itme vektörü açılarına (δ_{yaw} ve δ_{pitch}) çevirmektedir. Otopilotun, ivme komutunu itme vektörü açılarına çevirebilmesi için, ivmeyi itme vektörüne çeviren, kararlı kontrol döngülerine ihtiyaç vardır. Bu bölümde, yunuslama ve dönme eksenlerinde türetilen modeller ve döngülerin kararlı hale gelmesi için izlenen tasarım aşamaları kısaca anlatılmıştır.

4. bölümde oluşturulan füze dinamik ve kinematik modeli kullanılarak, füzeye ait doğrusallaştırılmış yunuslama ve dönme denklemleri türetilmiştir. Bu denklemler kullanılarak füze doğrusal modeli oluşturulmuştur.

Füzeye gelen itme vektörü komutu u bir servo sistem vasıtası ile itme vektörü açısı δ'ya çevrilir, servo sistem dinamiği, τ gecikmesine sahip, birinci derece bir sistem olarak modellenmiştir. δ açısı füze doğrusal modeli kulllanılarak füze ivmesine çevrilir.



Şekil 3. İtme Vektörü Yönlendirme Servosu ve Füze Doğrusal Modeli

Şekil 3. de verilen servo sistem ve füze doğrusal modeli göz önünde bulundurularak, sisteme ait çıkış füze ivmesi a ve giriş itme vektörü komutu olan u olarak alınırsa yunuslama ve dönme eksenlerindeki durum uzayı denklemleri oluşturulur. Durum uzayı denklemleri durum değişkenlerini (δ_{pitch} ve δ_{yaw} yunuslama ve dönme eksenlerindeki itme vektörü yönlendirme açıları, α füze hücum açısı, β füze yanakayma açısı, Q ve R yunuslama hızı ve dönme hızı) kullanarak, "itme vektörü yönlendirme servosu ve füze doğrusal modelinin" giriş ve çıkışı arasındaki ilişkiyi göstermektedir. Durum uzayı denklemlerinde kullanılan durum uzayı matrisleri, füzenin sahip olduğu fiziksel özellikler, itme kuvveti ve ortam koşulları göz önünde tutularak hesaplanan katsayılardan oluşmaktadır. Durum uzayı denklemleri aşağıdaki gibi ifade edilir:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{Q} \\ \dot{\delta}_{pitch} \end{bmatrix} = \begin{bmatrix} Z_\alpha / V_T & 1 & Z_\delta / V_T \\ M_\alpha & M_q & M_\delta \\ 0 & 0 & -1/\tau \end{bmatrix} \begin{bmatrix} \alpha \\ Q \\ \delta_{pitch} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/\tau \end{bmatrix} u \qquad (1)$$

$$a_Z = \begin{bmatrix} Z_\alpha & 0 & Z_\delta \end{bmatrix} \begin{bmatrix} \alpha \\ Q \\ \delta_{pitch} \end{bmatrix} \qquad (2)$$

$Z_\alpha = -\frac{qSC_{L\alpha}}{m}, Z_\delta = -\frac{T}{m}, M_\alpha = \frac{qScc_{m\alpha}}{I_y}, M_q = \frac{qSc^2 c_{mq}}{2V_T I_y}, M_\delta = -\frac{l_T T}{I_y}$ olmak üzere

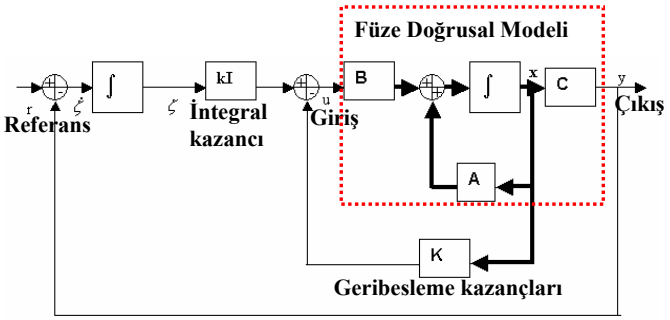
$$\begin{bmatrix} \dot{\beta} \\ \dot{R} \\ \dot{\delta}_{yaw} \end{bmatrix} = \begin{bmatrix} Y_\beta / V_T & (\frac{Y_r}{V_T} - 1) & Y_\delta / V_T \\ N_\beta & N_r & N_\delta \\ 0 & 0 & -1/\tau \end{bmatrix} \begin{bmatrix} \beta \\ R \\ \delta_{yaw} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/\tau \end{bmatrix} u \qquad (3)$$

$$a_Y = \begin{bmatrix} Y_\beta & Y_r & Y_\delta \end{bmatrix} \begin{bmatrix} \beta \\ R \\ \delta_{yaw} \end{bmatrix} \qquad (4)$$

$Y_\beta = \frac{qSC_{y\beta}}{m}, Y_r = \frac{qSbC_{yr}}{2mV_T}, Y_\delta = -\frac{T}{m}, N_\beta = \frac{qSbc_{n\beta}}{I_z}, N_r = \frac{qSb^2c_{nr}}{2V_TI_z}, N_\delta = -\frac{l_TT}{I_z}$ olmak üzere

$C_{L\alpha}, c_{m\alpha}, c_{mq}, C_{y\beta}, C_{yr}, c_{n\beta}, c_{nr}$ katsayıları Missile DATCOM ile hesaplanan füze aerodinamik model katsayılarıdır.V_T toplam füze hızıdır. l_T ağırlık merkezi ve itme kuvvetinin uygulandığı nokta arasındaki uzaklıktır. I_y ve I_z füze eylemsizlik momentleridir. T itme sisteminden verilen eksensel itme kuvveti, q dinamik basınç, S füze referans alanı ve m füze ağırlığıdır. b ve c uzaklıkları ise moment noktalarına olan uzaklıklardır. τ birinci derece servo gecikmesidir.

Yukarıda verilen doğrusallaştırılmış durum uzayı denklemleri ve [4] numaralı kaynakta verilen "Servo Sistem Tasarımı"ndan faydanılarak, füze için yunuslama ve dönme otopilotu tasarlanmıştır. Buna göre sistem ileri besleme yolu üzerine hata karşılaştırma sinyali (referans sinyali) ve sistem arasına bir tümleyici eklenmiş, durum değişkenlerinin her biri durum geri besleme sinyali olarak kullanılmıştır. Tasarlanan otopilot modelinin yapısı Şekil 4. de görülmektedir.

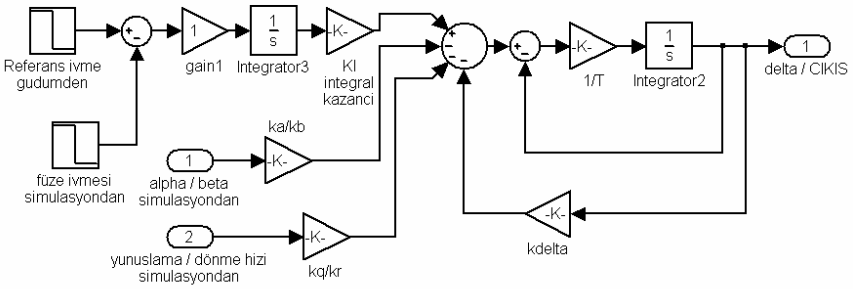


Şekil 4. Otopilot Tasarım Modeli

Sisteme ait kapalı döngü kutupları füze tepki zamanı her iki eksende de 1 sn olacak şekilde seçilmiş, kutup yerleştirme tekniği ile, sisteme ait integral kazancı ve durum geribesleme kazançları hesaplanmıştır. Füze simülasyonunun her koşulda (farklı yükseklik ve mach sayısı koşulları) kararlı çalışması için, sisteme ait integral kazancı ve durum geribesleme kazançlarının her uçuş koşulunda hesaplanması gerekmektedir. Füze yüksekliği için 23; füze hızı için 20 farklı koşul belirlenmiş ve her bir kazanç parametresi için 460 farklı değer hesaplanmıştır. Bu değerler füze simülasyonunda kullanılmak üzere tablolarda tutulmuştur.

Füze simülasyonunda, otopilot modelinde bulunan füze doğrusal modeli bloğu iptal edilecek, α, Q, β, R, füzenin Y yönündeki ivme değeri olan a_y ve füzenin Z yönündeki ivme değeri olan a_z simülasyondan alınacaktır. Referans değerleri olan a_{NC} ivmeleri güdüm komutu olarak güdüm bloğundan gelecektir. Otopilot döngüsünde

hesaplanan δ_{pitch} ve δ_{yaw} değerleri ise çıkış olarak simülasyona gönderilecektir. Simülasyonda kullanılan yunuslama ve dönme otopilotu giriş-çıkışları Şekil 5. de görülmektedir.



Şekil 5. Simülasyonda Kullanılan Yunuslama ve Dönme Otopilotu Giriş-Çıkışları

6. FÜZE UÇUŞ SİMÜLASYONU

Bu bölümde daha önceki bölümlerde oluşturulan, füze dinamiği ve hareket denklemleri modeli ve füze otopilot modeli birleştirilerek füze uçuş simülasyonu modeli oluşturulmuş, füze uçuş simülasyonu modeli kullanılarak, Visual C++ ortamında, hazırlanan simülasyon yazılımı, SCUD-B füzesine ait parametrelerle çalıştırılmıştır. İlk önce kontrolsüz füze uçuşu incelenmiştir.

Balistik füzelerin 90°'ye yakın açılarla, dik olarak, bir lançerden fırlatıldıkları bilinmesine rağmen, SCUD-B için bu açının kaç derece mertebesinde olduğu simülasyonlarla belirlenmiştir. Füze fırlatma açısı 85° alındığında, füzenin 320 saniyede 324 km yol aldığı ve 89 km yükseğe çıktığı gözlenmiştir. Bu değerler, SCUD-B füzesi için verilen maksimum menzil, ve maksimum havada kalma süresi ve maksimum menzil için verilen yükseklik değerleri ile yaklaşık olarak aynıdır.

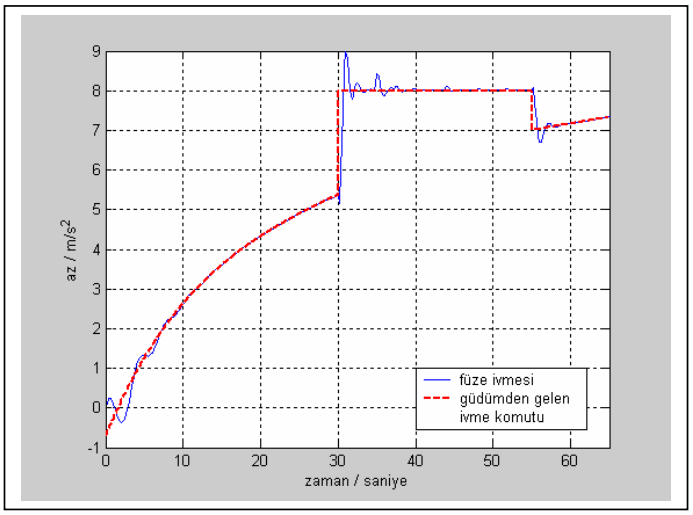
Simülasyonda, füze fırlatma açısı 80° alındığında, füzenin, 100 km yol aldığı, 16 km yüksekliğe çıktığı ve 160 saniye havada kaldığı gözlenmiştir. Bu değerler SCUD-B füzesi için verilen, minimum menzil, minimum havada kalma süresi ve minimum menzil için yükseklik (165 saniye, 50 km menzil ve 24 km yükseklik) değerlerine yakındır.

Yukardaki sonuçlardan görüldüğü gibi SCUD-B için verilen optimum fırlatma açısı maksimum 85°, minimum 80° derece olmalıdır. SCUD-B simülasyonu için elde edilen bu sonuçlar, literatürde verilen değerlerle örtüşmektedir (maksimum ve minimum menzil için). Bu sonuç, füze dinamiği ve kinematiği bloklarının doğru modellendiğinin ve uygun parametreler verildiğinde gerçeğe yakın çalıştığının bir belirtisi olarak yorumlanmıştır.

Senaryo 1

Bu senaryoda, güdümden gelen ivme komutu olarak kontrolsüz füze uçuşundaki füze a_z ivmesi, otopilota gönderilmiştir. Fakat 30-55 saniyeler arasında güdüm ivmesi 8 m/s^2 olarak verilmiş, füzenin bu ivmeyi takip edip edemediği incelenmiştir.

Füze fırlatma açısı 85° alınmıştır. Bu simülasyonun en önemli sonucu hiç şüphesiz, füze ivmesinin, güdümden gelen ivme komutunu takip ettiğini görmek olacaktır. Bu sonuç, otopilot tarafından hesaplanıp, füze dinamik ve kinematik bloğuna giden itme vektörü yönlendirme açılarının, füzenin ivmesinin doğru şekilde hesaplanmasını sağladığını göstermektedir.



Şekil 6. Kontrollü Uçuş Simülasyonu, Zamana Göre a_z İvmesi Değişimi

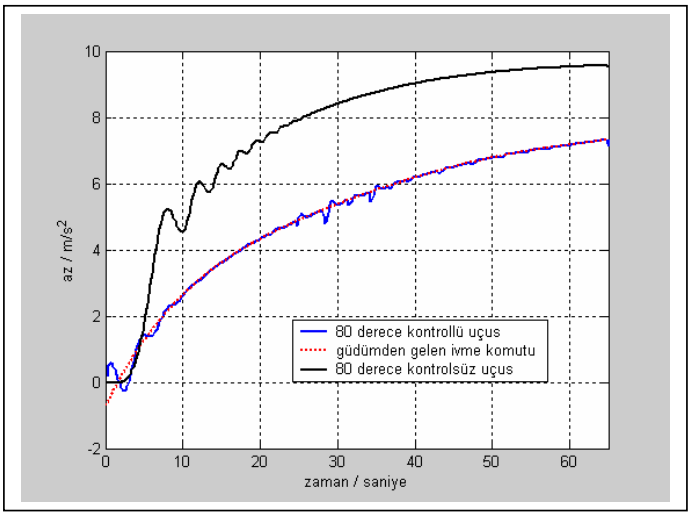
Şekil 6. da, füze ivmesinin güdümden gelen ivmeyi hatasız ve kararlı şekilde takip ettiği gözlenmiştir. Bu sonuç otopilotun doğru modellendiğinin ve bu modelin tüm 6-DOF füze simülasyonu ile uyumlu çalıştığının göstergesidir.

Senaryo 2

SCUD-B füzesi için optimum atış açısı 80-85° aralığında bulunmuştur. Bu senaryoda SCUD-B füzesi 80° ile atılacak fakat otopilota güdümden gelen ivme komutu olarak, füzenin 85° ile kontrolsüz atıldığında sahip olduğu ivme verilecektir.

Buradan beklenen sonuç, 80° ile atılan füzenin, 85°ile atılmış füze ivme komutlarını hatasız takip etmesi ve 85°ile atılmış olan füze yörüngesini izlemesidir.

80° ile atılan füzeye güdüm ivmesi olarak füze 85° ile kontrolsüz atıldığı zaman sahip olduğu ivme verildiğinde füze maksimum yüksekliği 16 km den 60 km ye, füzenin toplam havada kalma süresi 160 saniyeden 275 saniyeye ulaşmıştır. Bu değerler, 85 derece ile atılan füze maksimum yükseklik ve havada kalma sürelerine (98 km, 320 saniye) yakındır.



Şekil 7. Uçuş Simülasyonu, Zamana Göre a_z İvmesi Değişimi

Bu simülasyondan elde edilen sonuca göre, 80° ile atılan füze, 85° ile atılmış füze ivme komutlarını hatasız takip etmiş ve 85° ile atılmış olan füze yörüngesine yakın bir yörünge izlemiştir. Bu yaklaşım, basit bir açık döngü güdüm yönteminin de temellerini oluşturmaktadır. Şekil 7. de 80° ile kontrolsüz atılan füze ivmesi, 80° ile kontrollü atılan füze ivmesi ve kontrollü füzeye verilen ivme komutları gösterilmiştir.

7. SONUÇLAR

Balistik füzeler için türetilen hareket ve moment denklemlerinden yunuslama ve dönme otopilot denklemleri elde edilmiştir. Balistik füze otopilotunu kararlı hale getirmek ve füze tepki zamanını ayarlamak için, otopilot kapalı döngü bir sistem haline getirilmiş ve otopilot kazançları buna göre ayarlanmıştır.

Balistik füze için oluşturulan tüm modeller bir araya getirilip 6-DOF doğrusal olmayan bir füze uçuş simülasyonu modeli oluşturulmuştur. Bu simülasyona, internet kaynaklarında elde edilen parametreler verildiğinde (füze ağırlığı, yarıçapı, uzunluğu, itme kuvveti, yakıt ağırlığı, motor yanma süresi v.b), füze simülasyonunden elde edilen, füze uçuş yolu, füze maksimum yüksekliği, füze havada kalma süresi gibi değerlerin, kaynaklarda verilen değerlerle örtüştüğü görülmüştür. Bu sonuç, füze ağırlık merkezinin değişimi, aerodinamik katsayılar, füze dinamiği ve kinematiği bloklarının doğru modellendiğini ve uygun parametreler ile gerçeğe yakın çalıştığını göstermektedir.

Kısa menzilli balistik füzelerin, lançerden 90°'ye yakın açılarda ve düşük ilk hız değerinde atıldığı bilinmektedir. Simülasyon kullanarak yapılan, kontrolsüz, serbest füze uçuşu için elde edilen sonuca göre, SCUD-B için maksimum menzilde optimum atış açısının maksimum 85°, minimum 80° olması gerekmektedir.

Simülasyonda, kontrol bloğu çalıştırıldığında güdümden gelen ivme komutlarının füze tarafından hatasız takip edildiği gözlenmiştir.

85° ile kontrolsüz atılan füzeye ait ivme değerleri, 80° ile kontrollü atılan füzeye, güdüm komutları olarak verildiğinde, füzenin bu güdüm komutlarını hatasız takip ettiği, ve 80° ile atılan füzenin, yaklaşık olarak, 85° ile atılan füze uçuş yoluna sahip olduğu gözlenmiştir. Bu sonuç füzeye doğru güdüm komutları verildiğinde, uygulanan kontrol sayesinde, füzenin doğru yönlendirildiğini göstermektedir.

Bu çalışma ile balistik füze otopilotlarının doğrusal uçuş modelleri temel alınarak tasarlanabileceği ve bütün otopilot yapılarının kolaylıkla balistik güdüm birimleri ile sinerji içinde çalışabileceği gösterilmiştir. Bu bildirinin temel aldığı çalışma [5] numaralı kaynakta çok daha detaylı olarak raporlanmıştır.

KAYNAKLAR

[1] Blake, William, (1998), "Missile DATCOM User's Manual-1997 FORTRAN 90 REVISION"

[2] Blakelock, John H., (1991), "Automatic Control of Aircraft and Missiles", John Wiley & Sons, pp.610.,pp.7-61.,pp.112-143

[3] "Jane's Strategic Weapon Systems", (Issue Thirty-nine July 2003), Duncan Lennox

[4] Ogata, Katsuhiko, (2002), "Modern Control Engineering", Prentice Hall, pp.847-855

[5] Tatas, Zekiye Bilinç, (2006),"SCUD Tipi Bir Balistik Füzenin Modellenmesi ve Otopilot Tasarımı",Hacettepe Üniversitesi Yüksek Lisans Tezi

[6] Wise, Kevin A., Broy, David J.,(1998),Agile Missile Dynamics and Control, Journal of Guidance, Conrol and Dynamics, Vol21: No:3

[7] www.fas.org

[8] www.globalsecurity.org

[9] www.janes.com

FST: HİDROFONUN YER DEĞİŞTİRMESİNİN HESAPLANMASI İÇİN BENZETİM TABANLI SONOBOY MEKANİK TASARIM ARACI

Mehmet HAKLIDIR

TÜBİTAK MAM, Bilişim Teknolojileri Enstitüsü, Gebze-Kocaeli,
mehmet.haklidir@bte.mam.gov.tr

ÖZ

Sonoboylar temel olarak; gemiler, deniz karakol uçakları ve denizaltı savunma harbi helikopterleri tarafından denize atılan ve yapı itibariyle suda yüzen şamandıralara benzeyen cihazlardır. Sonoboylar aktif, pasif ve özel amaçlı olmak üzere temelde üç kategoriye ayrılırlar. Bu çalışmada pasif sonoboy temel alınmıştır. Pasif sonoboylarin süspansiyon sistemi bir yüzey şamandırası, süspansiyon kablosu, bu kabloya bağlı olan ve hidrofonun yer aldığı şaseden oluşmaktadır.

Sonoboyun pasif dinleme yapabilmesini sağlayan parçası olan hidrofonun, verimli çalışabilmesi için istenen dinleme derinliğinde mümkün oldukça hareketsiz kalması sağlanmalıdır. Sonoboy, bunu gerçekleştirecek bir süspansiyon sistemine ihtiyaç duyar. Bu çalışmada, sonoboyun süspansiyon sistemi incelenmiş, dinamik modellenmesinin yapılabilmesi için matematik modeli oluşturulmuştur. Sonoboyun dinamik davranışlarının analiz edilmesi için MATLAB programı kullanılarak FST (Fiziksel Sonoboy Tasarımı) adlı bir araç geliştirilmiştir. Süspansiyon sisteminin geometrisinin tanımlanabildiği bu araç, aynı zamanda sonoboyun dalga ve akıntı kuvvetlerinin etkisindeki hareketinin, hidrofona yansıyan etkisinin gözlemlenebileceği bir simülasyondur. Simülasyon sonuçlarına ve yorumlarına çalışmanın son kısmında yer verilmiştir.

Anahtar Kelimeler: Sonoboy, Dinamik Modelleme, Denizaltı Savunma Harbi, Ortam Benzetimi, Simülasyon.

FST: SIMULATION BASED SONOBOUY MECHANICAL DESIGN TOOL TO CALCULATE THE REPLACEMENT OF HYDROPHONE

ABSTRACT

Principally, sonobuoys, like free floating buoys, are dropped from the ships, maritime patrol aircrafts and anti submarine warfare helicopters. Sonobuoys are grouped into three categories: passive, active and special purpose. In this study, Passive Sonobuoy is main object. Passive sonobuoy suspension system has a surface float, a suspension cable having a bridle with a hydrophone.

Hydrophone, used to listening passively, has to be hold on motionless to be used efficiently. Thus, sonobuoy needs a suspension system. In this study, sonobuoy suspesion system has been scrutinized, mathematical model has been modeled to model dynamically. A tool, called FST, has been developed by MATLAB to analyze dynamic behavior of the sonobuoy. This tool, geometry of the suspension system can defined by, is a simulation to observe the response of the sonobuoy line hydrophone suspension to surface wave-induced and to flow-induced motions. the results of simulation has been discussed.

Keywords: Sonobuoy, Dynamical Modelling, Anti Warfare Submarine, Environmental Simulation, Simulation.

1. GİRİŞ

I. Dünya savaşının başlangıcıyla denizaltıların, deniz harbinde önemli ve güçlü bir silah haline gelmesi, sualtı akustiği ile ilgili araştırmaların önem kazanmasına neden olmuştur. Daha önce sadece seyir yardımcısı olarak kullanılabileceği düşünülen sonar sistemlerinin gelişimi hız kazanmış, sualtındaki ve/veya suüstündeki gürültü kaynaklarından yayılan ses dalgalarının ve bu dalgaları yayan hedeflerin yerlerinin saptanmasını amaçlar hale gelmiştir. II. Dünya savaşıyla birlikte sonar teknolojisi, sualtı ve su üstü savaş unsurları olan denizaltıların ve gemilerin aktif gönderme ya da pasif dinleme ile birbirlerinin yerlerini, uzaklıklarını saptayabilir hale gelmesini sağlamış ve aktif olarak kullanılmaya başlanmıştır. Özellikle Almanların, İngiliz Deniz Kuvvetlerine gerçekleştirdikleri 'Wolf Pack' adı verilen saldırılar sonucunda geliştirilip, platformdan bağımsız olarak ekonomik, kolay yerleştirilebilir ve kullanılabilir hale dönüştürülmesine ihtiyaç duyulmuştur. Bu aynı zamanda hava kuvvetlerinin denizaltı savunma harbine katılmasını sağlayacak olan sonoboyların temellerini atılmasını neden olmuştur. [1]

Askeri amaçlı olarak denizaltı saptamasında kullanılan sonoboyların dinamikleri hakkında bilinen ilk çalışma Misra'nın 1967 yılında "Denizaltı saptamasında kullanılan yapıların dinamikleri" adlı doktora çalışmasıdır. Wang (1977), FORTRAN'da yazdığı bir bilgisayar programıyla sonoboyların iki eksendeki dinamik hareketlerinin zaman alanlı çözümlemelerini gerçekleştirdi.[2] Hsiung ve Lu (1987), sonoboyun esnek parçalarının maruz kaldığı kuvvetlerin hesaplanması için bir sonlu elemanlar modeli oluşturdu.[3] 1990'ların başından itibaren doğrudan sonoboyla ilgili olmasa da benzer dinamik yapılarla ilgili birçok çalışma yapılmıştır

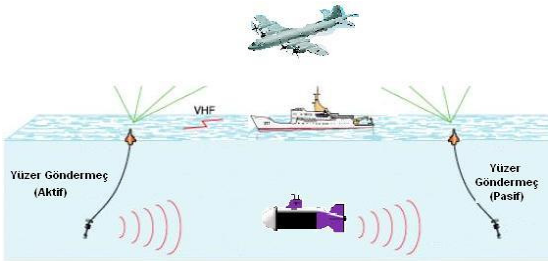
Bu çalışmada, sonoboyun dinamik modellemesinin yapılabilmesi için matematik modeli oluşturulmuş, bu matematik modelden yararlanılarak hidrofonun yer değiştirmesinin hesaplanabilmesi için MATLAB aracılığıyla sonoboy tasarımı yapılabilecek FST adlı bir araç ve çevre şartlarına karşı vereceği cevabı gözlemleyebilmek için bir simülasyon sistemi gerçeklenmiştir.

2. SONOBOYLAR

Donanmaya bağlı Denizaltı Savunma Harbi (DSH) birimlerinin birincil görevleri, denizaltıları saptamak, yerlerini belirlemek ve tanımlamaktır. Bu görevleri gerçekleştirebilmelerini sağlayan en önemli aygıtlardan biri de sonoboylardır.

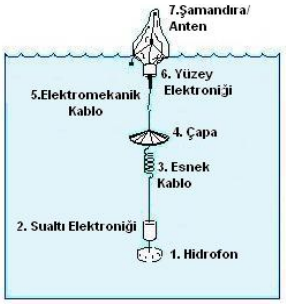
Sonoboy, denizaltıların yerinin, rotasının ve hızının genel olarak veya hassas bir şekilde tespit edilmesi amacıyla su altına yayılan pervane ve makine kaynaklı sesleri ya da denizaltının yüzeyinden yansıyan sonar sinyalini kullanan bir aygıttır. [4]

Atılmadan önce bir kapsülün içinde yer alan sonoboy, denizaltının olduğu düşünülen bölgeye askeri bir hava platformu tarafından atılır. Bir paraşüt aracılığıyla hedeflenen bölgeye düşen sonoboy, denize girdikten hemen sonra etkinleşerek kapsülden kurtulur ve ilgili bölgeye konuşlanarak dinleme yapmaya başlar (Şekil1). Sonoboy, dinleme yaptığı hidrofonu aracılığıyla su altı mikrofonu gibi çalışmaktadır. Hidrofon aracılığıyla topladığı akustik verileri, osilatörü aracılığıyla dönüştürdükten sonra su üstü platformlara iletmekte, böylelikle denizaltının tespitinin kolay, tehlikesiz yapılmasını sağlamaktadır.



Şekil 1. Sonoboyun kullanımı

Sonoboyun alt parçaları Şekil 2'te gösterilmiştir.



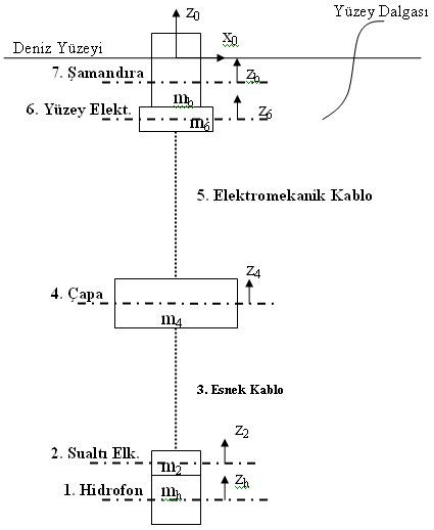
Şekil 2. Yüzer göndermecin alt parçaları

Bu parçaların tamamının amacı, hidrofonun istenen dinleme derinliğinde pozisyonunu koruyabilmektir. Hidrofon, verimli çalışabilmek için bu derinlikte mümkün olduğunca hareketsiz kalmalıdır. Hidrofonun istenmeyen bir gürültü ile sonuçlanacak her türlü hareketi verimin düşmesine neden olacaktır. Bu istenmeyen gürültüleri yaratan iki temel hareket vardır: Yatay ve düşey hareketler. Düşey hareket, yüzey şamandırasına etki eden dalga kuvvetlerinin hidrofon kütlesine etkileşiminden kaynaklanmaktadır. Yatay harekete ise denizin farklı derinliklerindeki değişken akıntı hızları neden olur. Yüzer göndermecin su üstünde yüzen kısmı, su altında kalan parçaları ve hidrofona etki eden akıntı hızı birbirlerinden farklıdır. Bu fark hidrofonun bağıl bir hıza sahip olmasına ve yatayda bir harekete neden olur. Bu hareket ve hareketten doğacak gürültüyü azaltmanın en uygun yolu olarak çapanın yüzey alanın arttırılması ön görülmektedir. Düşey hareketin kararlılığının sağlanabilmesi için de esnek kablonun sönümleme katsayısı göz önünde bulundurularak uygun seçilmesi gerekmektedir.

3. SONOBOYUN MATEMATİKSEL MODELİ

3.1 Fiziksel Model

Sonoboyun fiziksel modeli Şekil 3'de verilmiştir.



Şekil 3 Sistemin Fiziksel Modeli

Bu modelde, gerçek yapının kullanım amaçları ve şekli göz önünde bulundurularak hidrofon ile sualtı elektroniği ve şamandıra ile yüzey elektroniğini birleştiren küçük esnek yapılar göz ardı edilmiş ve bunların sistemde birleşik kütleler gibi davrandığı kabulü yapılmıştır. Sistem, temel olarak kütleler ve onları birleştiren esnek yapılar olmak üzere iki grupta toplanmaktadır. Matematiksel model oluşturulurken de bu ayrımı göz önünde bulunduran bir yaklaşım izlenmiştir.

3.2 Sistemin Dinamiği

Bu çalışmada frekans alanlı ve zaman alanlı çözümlemelerin her ikisi de gerçekleştirilmiştir. Frekans alanlı çözümlemeler, sistemin temel davranışlarını hızlı gözlemlemek ve tasarım çalışmalarına bunu yansıtmak için avantajlıdır. Örneğin, dalga boyunun yüksek olduğu deniz durumu 5 ve 6 için yapılacak bir frekans alanlı çözümleme sonucunda, sonoboyun esnek kabloları üzerindeki gerilim dağılımı kolaylıkla gözlemlenebilir. Bu da tasarım için gerekli olan azami gerilim değerini kolaylıkla hesaplamayı sağlar.

Zaman alanlı çözümleme ise frekans alanlı çözümlemeler sonucunda elde edilemeyecek doğrusal olmayan davranışları belirlemek için kullanılırlar.

3.1.1 Frekans Alanlı Çözümlemeler

Frekans alanlı çözümlemelerde sistemde, hidrofonun dinamik denge denklemi;

$$M_1 \frac{d^2 U_1}{dt^2} + B_1 \frac{dU_1}{dt} = EA_1 \frac{\partial u_1}{\partial z_1} \Big|_{z_1=0} \qquad (1)$$

Sualtı elektroniğinin ve çapanın dinamik denge denklemleri;

$$M_j \frac{d^2U_j}{dt^2} + B_j \frac{dU_j}{dt} = EA_j \frac{\partial u_j}{\partial z_j} \Big|_{z_j=0} - EA_{j-1} \frac{\partial u_{j-1}}{\partial z_j - 1} \Big|_{z_{j-j}=L_{j-1}}) \quad (2)$$

Yüzen şamandıranın dinamik denge denklemi ise;

$$M_B \frac{d^2U_B}{dt^2} + B_B \frac{dU_B}{dt} + \rho g A_{wp} U_B = -EA_n \frac{\partial u_n}{\partial z_n} \Big|_{z_n=L_n} + F_B(t) \quad (3)$$

Bu denklemde $F_B(t)$ dalga kuvvetini, A_{wp} ise şamandıranın su yüzeyi alanını ifade etmektedir.

j parça numarası olmak üzere her bir elastik segmentin (elektromekanik kablo, esnek kablo) denge denklemlerini aşağıdaki şekilde gösterebiliriz.

$$m_j \frac{\partial^2 u_j}{dt^2} + \left(b_j + r_j\right) \frac{\partial u_j}{\partial t} = EA_j \frac{\partial^2 u_j}{\partial z^2_j} \quad (4)$$

Esnek segment boyunca yer değiştirme ve kütlenin yer değiştirmesi aşağıdaki denklemlerle hesaplanabilir.

$$u_j(z_j, t) = e^{iwt}\left[C_{2j-1} \cos k_j z_j + C_{2j} \sin k_j z_j\right] \quad (5)$$

$$U_j(t) = \hat{U}_j e^{iwt} \quad (6)$$

Burada C_J ve U_J kompleks sabitlerdir. k_J ise aşağıdaki ifadeyle hesaplanır.

$$k_j = \sqrt{\frac{m_j\omega^2 - i\omega(b_j + r_j)}{EA_j}} \quad (7)$$

(5) ve (6) numaralı denklemler, (1)-(4) numaralı dinamik denge denklemlerinde yerine konursa denklemlerin zamana bağımlılığının ortadan kalktığı görülür.[5]

Sistemin frekans cevabına göre sönümleme katsayıları;

$$b_{eq} = 2\sigma_v \sqrt{\frac{2}{\pi}} \left(\frac{1}{2}\rho C_f \pi d\right) \qquad B_{eq} = 2\sigma_v \sqrt{\frac{2}{\pi}} \left(\frac{1}{2}\rho C_D \pi d\right) \quad (8)$$

C_D: Kütlenin sürükleme katsayısı
C_f: Kütlenin sürtünme katsayısı
S: Yüzey kesit alanı
d: Kablonun çapı
b_J : Eşdeğer sönümleme katsayısı
B_J : Kütleye bağlı eşdeğer sönümleme katsayısı
EA_J : Eksenel sertlik
r_J : Yapısal sönümleme katsayısı
L_J : Uzunluk
m_J : Kütle
n : Kütle bağlantısı için kullanılan elastik element sayısı
z_J : j sayılı parçanın z koordinatı

3.1.2 Zaman Alanlı Çözümlemeler

Zaman alanlı modelde, elastik segmentlerle kütlelerin birleşim noktaları temel alınarak çözümleme gerçekleştirilmiştir. Aynı zamanda modelde, her elastik segmentin, sıralı bir boyutlu idealleştirilmiş kütle yay sistemlerinden oluşmuş gibi davrandığı kabulü yapılmıştır. Sistemin hareket denklemleri, uzay ve zaman içinde sonlu farklar metodu kullanılarak oluşturulmuştur. Modelde, Δz'in değişken olmasına izin verilerek farklı şekilde bölümlemelere imkân tanınmış ancak benzetim boyunca Δt zamanı sabit kabul edilmiştir. Sistemin sönümleme katsayıları aşağıdaki verilmektedir;

$$B_s = \frac{1}{2}\rho(C_D)_s S_s \tag{9}$$

$$b_s = \frac{1}{2}\rho(C_f)_s \pi d_s \tag{10}$$

s hangi parçadaki düğümün çalıştığını göstermekte kullanılır. M_s ve B_s daima s segmentinin altındaki masif sistem bileşenin özelliklerini açıklamakta kullanılır.[6]

3.1.3 Deniz Durumu ve Dalga Kuvveti

Karışık denizlere ait gerçek verinin mevcut olmadığı durumlarda denizcilik hesaplarında genellikle idealleştirilmiş matematiksel spektrum fonksiyonlarından yararlanılır. Bunlardan en basiti ve en çok kullanılanı Pierson – Moskowitz spektrumu tam oluşmuş denizler için sabit hızda ve sonsuz bir periyot için sonsuz genişlikte bir saha üzerinden esen rüzgarlar için bir dalga spektrum formülasyonu verilmektedir. Bu çalışmada da dalga modeli oluşturulurken bu spektrumdan yararlanılmıştır. Bu spektrum, dalga frekansı ve rüzgâr hızına bağlı olarak aşağıdaki gibi ifade edilir.

$$S_\xi = \frac{0.0081g^2}{\omega^5}\exp\left[-0.74\left(\frac{g}{V\omega}\right)^4\right] \tag{11}$$

Burada
ω : Dalga frekansı [rad/s]
V: Deniz yüzeyinden 19,5 metre yukarıda ölçülen rüzgâr hızı [m/s]

4. TASARIM

FST, sonoboyların benzetim tabanlı mekanik tasarımını yapmak amacıyla MATLAB programı kullanılarak geliştirilmiştir. Sahip olduğu arayüz sayesinde aracı kullanan kişi giriş parametrelerini (kablo tipi, boyu, hidrofonun boyutları, kütlesi …) değiştirerek yeni bir sonoboy tasarlayabilir, bunu kaydedebilir ve elde ettiği sonuçları farklı tasarım sonuçlarıyla karşılaştırabilir. Aynı zamanda bu arayüz aracılığıyla ortam şartları belirlenerek, sonoboyun farklı ortam şartlarındaki davranışları izlenebilmektedir.

(a) **(b)** **(c)**

Şekil 4. **(a)** Sonoboy Tasarım Ekranı, **(b)** Deniz Durumu **(c)** Çevre Şartları Ekranı

Benzetim, zamana bağlı olarak akıntı, deniz durumu ve rüzgar etkisindeki sonoboyun dinamiklerinin tasarlanıp geliştirilebilmesi için kuvvet denge denklemleri setini çözerek, her bir alt parçasının konumunu belirlemeyi amaçlar. Bunun içinde sonoboyun her bir parçasının konumunu belirlemek için iteratif bir yaklaşımla, parçanın konumu bir noktaya yakınsanana kadar çözümleme yapmaktadır. Benzetim sırasında ilk iterasyon şamandıradan başlayıp aşağı doğru sürmektedir. Her yeni çözümde parçalara etkileyen kuvvetler ve buna bağlı konum bilgileri güncellenmektedir.

Parçaların konumları dikeyde 1 cmlik değişiklikler ile hesaplanmaktadır. Her bir iterasyon adımında gerçekleşen bu hesap hassasiyeti, çözümün toplam hassasiyetini belirleyicidir. Sonoboyların bulunması gereken konum santimetreler seviyesinde hassas olmadığı için seçilmiş olan hesap hassasiyeti benzetim sonuçları açısından yeterli olabilmektedir. Eğer iterasyon adımlarının birinde şamandıra dikeyde 1 cmden daha küçük bir yer değişikliğine uğradıysa benzetim çözümün yakınsadığını kabulünü yaparak iterasyonu durdurur ve sonoboyun alt parçalarının o anki pozisyonlarını kayıt eder.

Sistemde, çapa için 3 denklem ve 6 bilinmeyen; üstten gerilim, alttan gerilim, gerilimlerin z eksenindeki açısı (Ψ), gerilimlerin x-y düzlemi ile açısı (θ); hidrofon ve şamandıra için ise tek bir gerilimin etkisi altında oldukları için 3 denklem ve 3 bilinmeyen bulunmaktadır. Bu nedenle çözüm şamandıranın kuvvet denge denklemlerinden yararlanılarak, şamandıradan başlayarak aşağı doğru sürmektedir.

4.1 Örnek Çözüm

Bu çalışmada, benzetimi gerçekleştirecek pasif sonoboyun parametreleri A sınıfı sonoboyların sahip olması gereken özellikler göz önünde bulundurularak ortalama değerler olarak alınmıştır.[7]

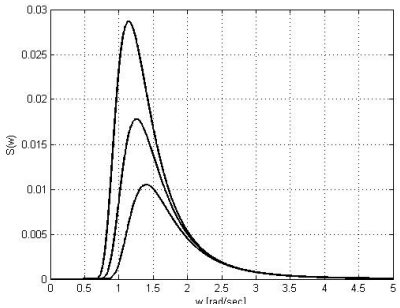
Çizelge 1. Benzetim parametreleri

	Şamandıra	**Yüzey Elek.**	**Çapa**	**Sualtı Elek.**	**Hidrofon**		**EM Kablo**	**Esnek Kablo**
Kütle [kg]	10	20	1,2	5	2	Çap [m]	0,004	0,006
x-kesit alanı [m^2]	0,0256	0,033	0,4	0,033	0,018	Kütle [kg]	0,0015	0,0092
z- kesit alanı [m^2]	0,037	0,041	0,08	0,041	0,024	Boy [m]	100	100
C_D	0.45	0,6	0,6	0,6	0,6	C_D	1,4-0,02	1,4–0,02

Benzetimde dalga modeli oluşturulurken Pierson-Moskowitz Spektrumundan yararlanılmış ve ortam şartları olarak deniz durumu 4'e ait aşağıdaki parametreler kullanılmıştır.

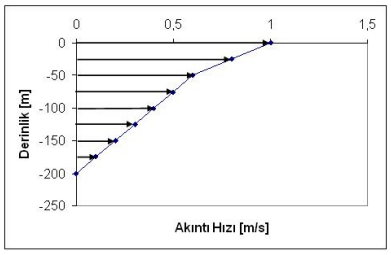
Çizelge 2. Seçilen deniz durumu ve ilgili parametreleri

Deniz Durumu	**Rüzgârın Hızı [m/s]**	**Maks. Dalga Yüksekliği [m]**	**Model Frekansı [Hz]**
4	8-11	2,5	0,12



Şekil 5. Deniz durumu 4 için Pierson-Moskowitz Tayfı

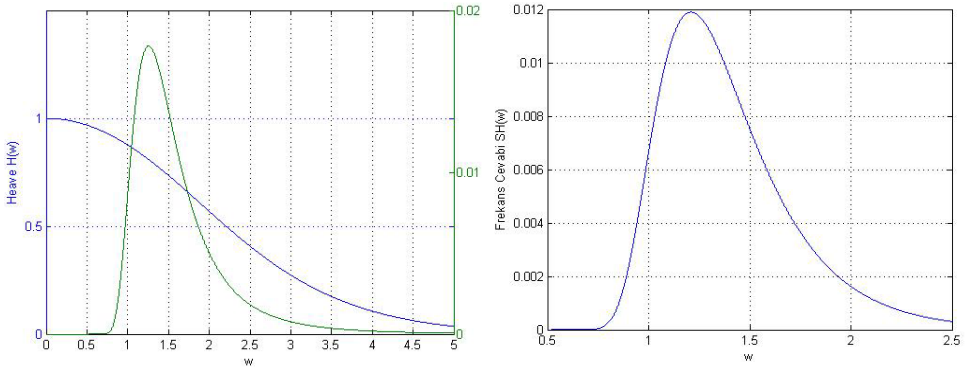
Denizde, derinliğe bağlı akıntı hızı profili için Eckman (1968) örnek alınmıştır.



Şekil 6. Akıntı Hızı Profili

Sistemin oluşturulan matematik modelinden yararlanılarak, MATLAB programı aracılığıyla benzetimi gerçekleştirilmiş ve benzetim sonucunda elde edilen sonuçlar aşağıda verilmiştir.

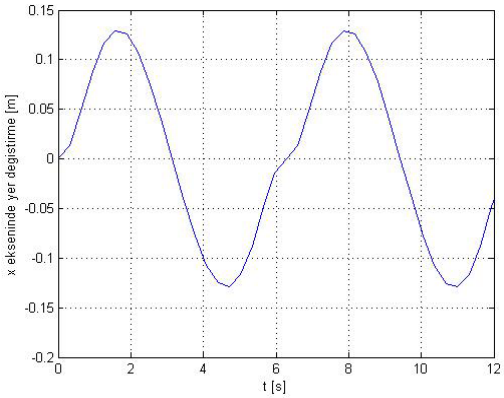
Şamandıra



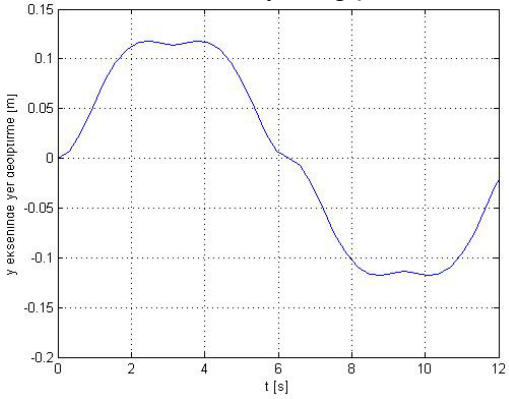
(a) **(b)**

Şekil 7. (**a)** Batıp çıkma (Boyuna ötelenme) – Dalga Frekansı Grafiği, **(b)** Batıp çıkma (Boyuna ötelenme – Heave) Frekans Cevabı

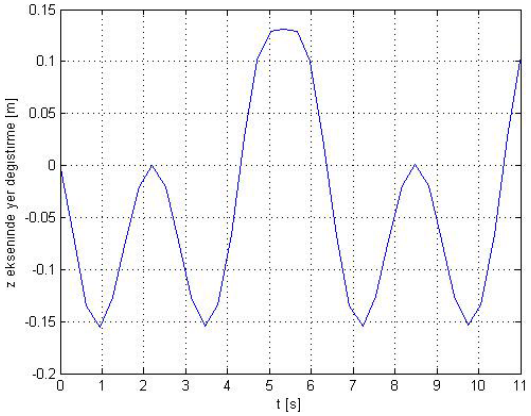
Hidrofon



Şekil 9. Hidrofonun x ekseninde yer değiştirmesi – Zaman grafiği



Şekil 10. Hidrofonun y ekseninde yer değiştirmesi – Zaman grafiği



Şekil 11. Hidrofonun z ekseninde yer değiştirmesi – Zaman grafiği

5. SONUÇLAR

Bu çalışmada, sonoboyun süspansiyon sistemi konum kestirimi yapabilmek için incelenmiş, dinamik modellenmesi gerçekleştirilip, simülasyon ortamında örnek bir çözüm irdelenmiştir.

Çalışmanın sonunda elde edilen temel sonuçlar aşağıdaki gibidir.

- Elde edilen sonuç, mevcut sonoboylarda hidrofonun düşey yer değiştirmesinin azaltılabilmesi için süspansiyon sistemleri geliştirmeye açık olduğu yönündedir.
- Düşey hareketin kararlılığının sağlanabilmesi için esnek kablonun sönümleme katsayısı göz önünde bulundurularak uygun seçilmesi gerekmektedir. Bu çalışma sonucunda ortaya çıkan benzetim, tasarım sürecinde doğru sönümleme katsayısını belirleme konusunda yararlı olacaktır.
- Yatay hareketin minimize edilebilmesi için çapanın çapının optimumunun bulunması gerekir. Bu çalışma konu olan araç, bu amaçla kullanıma çok açıktır.

KAYNAKÇA

[1] Naval Sea Systems Command, "Background of Air ASW Sensors", *Crane*, IN, http://sonobuoy.crane.navy.mil/sensors.htm

[2] Wang, H.T., A Fortran IV computer program for the time domain analysis of the two-dimensional dynamic motions of general buoy-cable-body system, Report 77-0046, David Taylor Naval Ship Research and Development Center, Bethesda, MD, 1977.

[3] Hsiung, C., Lu, Q. and Petolas, B., Dynamic Analysis of Buoy-Cable System, Proceedings Oceans '87, 19, 254-259, 1987.

[4] Navy Aviation Manuals, "Aviation Electronics Technician 1 (Organizational)", http://www.tpub.com/content/aviation/index.htm

[5] J. I. Gobat, "Reducing Mechanical and Flow-Induced Noise in the Surface Suspended Acoustic Receiver", MSc Thesis, MIT, Department of Ocean Engineering, Massachusetts, 1997.

[6] Driscoll, R. Nahon, M., Mathematical Modeling and Simulation of a Moored Buoy System, *Proceedings Oceans '96*, Broward County Convention Center Fort Lauderdale, Florida, USA,1996.

[7] Approved Navy Training System Plan for the Navy Consolidated N88-NTSP-A-50-8910B/A, September 1998.

[8] W. J. Palm III, *Modeling, Analysis, and Control of Dynamic Systems,* Wiley, New York, 2000

UÇUŞ KONTROL SİSTEMİ TASARIMI VE GELİŞTİRİLMESİ İÇİN SİMÜLASYON TABANLI TEST PLATFORMU YAZILIMI F-SIM

Yük. Müh. Ebru ARSLAN [(a)], Erkan Arıkan[(a)] , Dr. İlkay YAVRUCUK [(b)]

[(a)] Meteksan Savunma Sanayii A.Ş, Beytepe Köyü Yolu No:3 06800 Bilkent-ANKARA, {ebrua,earikan}@meteksan.com.tr

[(b)] ODTÜ Uzay ve Havacılık Mühendisliği Bölümü, ODTÜ Havacılık ve Uzay Mühendisliği Bölümü 06531-ANKARA,

yavrucuk@ae.metu.edu.tr

ÖZ

Uçuş sistemlerinin uçuş kontrol tasarımlarına yönelik geliştirilen yazılımlar, donanıma yüklenildiğinde karşılaşılan problemler ve test sürecinin uzunluğu, iş gücü ve zaman yönünden mali açıdan kayıplara sebep olmaktadır. Bu tür yazılımların hazırlanıp test edilmesi en zor adımlarından birisidir. Genellikle yazılımlar kaba hallerinde yazılır ve uçuş testlerinde ince ayarları yapılır. İyi hazırlanmış bir simülasyon ortamında bu kontrol algoritmalarının hazırlanması hem sistem güvenilirliğini artırır, uçuş testi sayısını azaltır, geliştirme hızını artırır dolayısıyla maliyeti düşürür. Bu makalede farklı uçuş platformları için hazırlanmış kullanımı oldukca esnek, modüler tabanlı böylesi bir simülasyon ortamı, F-SIM, tanıtılmaktadır. Bu yazılım, bir uçuş kontrol sisteminin geliştirme sürecinde önemli olan "döngüde yazılım simülasyonu" ve "döngüde donanım simülasyonları" ile sistematik bir süreç izleme imkanı tanımaktadır. F-SIM kütüphanesi, aynı zamanda modüler yapısı sayesinde, pilotlu uçuş platformunun oluşturulması, bir mühendisin geliştirdiği kodun donanıma simüle edilerek yüklenmesini ve yeri geldiğinde eğitim simulatörü olarak kullanılmasına da imkan sağlamaktadır.

Anahtar Kelimeler: Uçuş simülasyonu**,** Döngüde Yazılım Simülasyonu, Döngüde Donanım Simülasyonu, Modelleme, Simülasyon, Test

A SIMULATION TEST PLATFORM FOR İHA CONTROL SYSTEM DESIGN AND DEVELOPMENT SOFTWARE F-SIM

ABSTRACT

Problems encountered during the implemention of control software for flight, is a time consuming and increase cost. Commonly, control software is written in with approximate system values and is fine tuned during flight tests. However, preparing control algorithms in a well-designed simulation environment increases system confidence, reduced the number of flight tests, shortens development time, and hence reduces development cost. In this paper we present such a new software environment, modular enough to handle different flight platforms and different levels of autonomy. This simulation algorithm allows one to do the crucial "software-in-the-loop" and "hardware-in-the-loop" simulations in the development. The modular environment also allows designing simulations for pilot training of existing manned aircraft.

Keywords: Hardware in the loop simulation, Software in the loop simulation , Modeling, Simulation

1. GİRİŞ

Kontrol sistemlerinin geliştirilmesi ve testleri için en hızlı ve ucuz yol sistemin her bir yapıtaşının bilgisayar ortamında modellenip simulasyonlar ile geliştirilmesidir. Bu amaç için hazırlanan simülasyon ortamının son derece detaylı olması gerekir. Kontrol mimarisinin belirlenmesiyle başlayan geliştirme süreci uçuş testine kadar "döngüde yazılım simülasyonu" ve "döngüde donanım simülasyonları" ile sistematik bir süreç izlenmektedir. Bu aşamada sadece kontrol algoritmalarının sistem dinamiğine uygunluğu değil, sonucunda uçacak sistemin gerçek platform üzerinde denenmesi hedeflenir.

Uçuş platformu değiştikce simülasyon modeli ve dolayısıyla kontrol algoritmalarının tasarımı da değişecektir. Bu durumda her yeni uçan platform sistemi (F-4, F-16, IHA vb) için yeni bir simülasyon yazılımı yaratmak gerekir. Bu her yeni sisteme yeni bir iş yükü getirir. Bu makalede önerilen bu sistemin hem komuta kontrol sisteminin geliştirilmesinde, hem test edilmesinde kullanılabilecek ortak bir yazılım platformunun yaratılmasıdır.

Geliştirilen yazılım modüler yapıtaşları üzerine bir uçan platformun kontrol birimlerin her bir alt biriminin modellenmesi ile bir yazılım kütüphanesi oluşturuyor. Bu modüler yapı tanımlanan uçuş platformu değiştikce başta sistemin dinamik modeli olmak üzere minimum düzeyde değişiklik gerektireceği için geliştirme hızını artırır. Bu platformun bir özelliği de bilgisayar ortmanındaki simülasyonun "döngüde donanım simülasyonları" için aviyonik donanımı ile entegrasyonunun sözkonusu olmasıdır.

Bu tür bir platfom hem mevcut projelerin komuta kontrol sistemlerinin geliştirilmesini hızlandıracağı gibi, platformların sinyal seviyesinde aviyonik ve kontrol parametrelerinin simülasyonları nı yapılabilmektedir.

2. KONTROL SİSTEMİ TEST SİMÜLASYONLARI

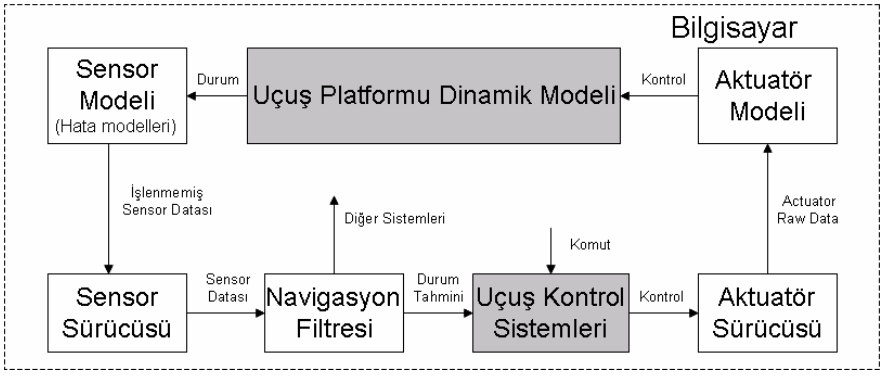
Bir uçan platformun kontrol sisteminin geliştirme sürecini hızlandıran önemli araçlardan biri sistemin simülasyonunu hazırlamaktır. Bu simülasyon sadece sistemin dinamik modelini değil her bir birimin sinyaller düzeyinde modellenmesini gerektirmektedir. Örneğin bir IHA simülasyonunda her bir sensör modellendikten sonra bu sensörden çıkan sinyal orjinal formatında oluşturulmalı, daha sonra sinyal İHA üzerinde olduğu gibi tekrar işlenmeli, anlaşılır bir formata çevrilip İHA işlemci simulatör birimine gönderilmektedir. Bu düzeyde oluşturulan bir simülasyonda hem oluşturulan yazılımların güvenilirlik düzeyi artar, hem de sistemde oluşabilecek hatalar çok önceden simülasyon ortamında görülebilmektedir. Sistemi bu detayda modellemenin avantajı, aksi halde görülemeyecek özelliklerin de incelenmesine olanak sağlamaktadır. Örneğin, sinyal akışının yavaşlaması veya sensör ölçüm hatalarının

sisteme etkisi incelenebilir, kontrol algoritmalarının bu durumlara davranışı incelenebilmektedir.

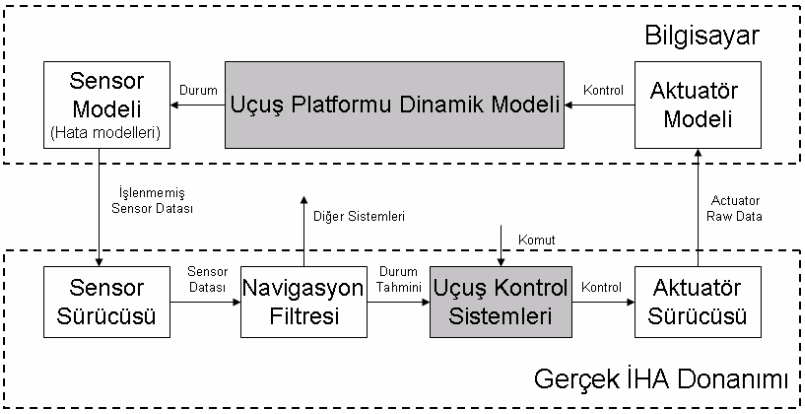
Böylesi bir test platformunun geliştirilmesinin 3 temel ayağı vardır: 1- Yazılım, 2- Elektronik Donanım, 3- Uçuş Platfomu. Bunlardan 1. ve 2. maddeler kontrol sisteminin parçalarıdır ve geliştirilmesi ve test edilmesi uzun ve zahmetli olan kısmıdır. Bu süreci kısaltmanın yolu ise yukarda sözü edildiği şekli ile bir simülasyon ortamında bütün parçaların simülasyonunu yapmaktır. Nitekim, bütün sistem parçalarının bilgisayar ortamında sinyaller düzeyinde modellenmesine "döngüde yazılım simülasyonu" olarak tanımlanmaktadır. (Şekil 1)

Bir sonraki adımda ise simülasyonun dinamik sensör ve aktuatör modelleri döngüde tutulurken gerçek sistemin donanımı olduğu gibi simülasyona dahil edilmektedir. Buna da "döngüde yazılım simülasyonu" olarak adlandırılmaktadır. (Şekil 2)

Şekil 1 ve 2'de gösterilen uçuş kontrol bloğu çok farklı düzeylerde kontrol yazılımı içerebilir; sistemi stabilize eden basit algoritmalardan sistemi tam otonom kılan kompleks yazılımlara kadar geniş bir spektrumda yazılım test etme imkanı olabilmektedir.



Şekil 1. Döngüde Yazılım Simülasyonu

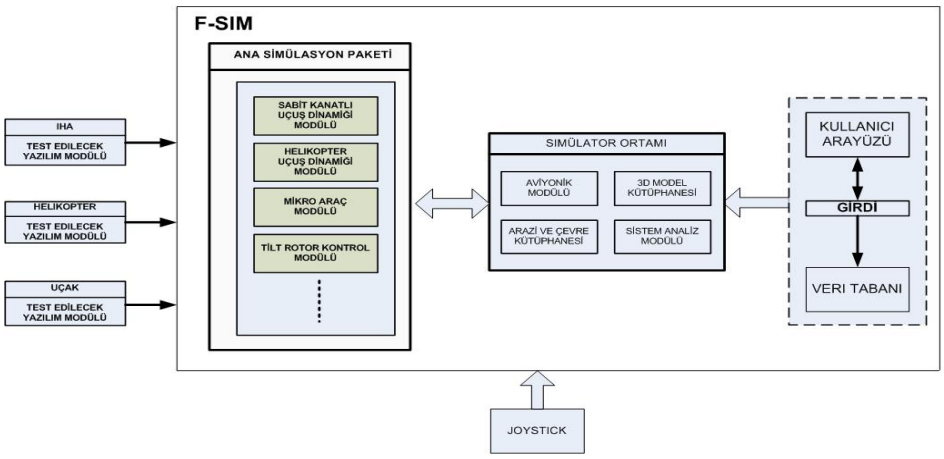


Şekil 2. Döngüde Donanım Simülasyonu

3. F-SIM SİMÜLASYON YAZILIMI

F-SIM yazılımı; sabit kanatlı, helikopter, gibi sistemlerin her türlü simülasyon ihtiyacını hem test, araştırma-geliştirme hem de eğitim amaçlı kullanılabilecek, yukarda anlatılan amaçlara hizmet eden yeni geliştirilmiş ve her geçen gün kabiliyetleri geliştirilen bir simülasyon yazılımıdır.

Bu yazılım, gerçek zamanlı işletim sistemi üzerinde çalışmakta olup modüler bir yazılım mimarisine sahiptir. Bu modüler yapı, böyle bir simülasyonda olması gereken parçaların bir yazılım kütüphanesini içermektedir. Örneğin; çeşitli uçuş platform dinamik modelleri, çeşitli sensor modelleri, sinyal çeşitleri,...vs. Kullanıcı bu kütüphaneden simülasyonun parçalarını seçerek dilediği simülasyon konfigürasyonunu oluşturabilmektedir. Bu bir İHA simülasyonu için olabileceği gibi bir insanlı eğitim simülatörü için de olabilmektedir. F-SIM modüler yazılım mimarisi Şekil 3 de tanımlandığı gibidir.



Şekil 3. F-SIM Yazılım Mimarisi

F-SIM yazılımı, Kullanıcı Arayüzü, Ana Simülasyon Paketi olmak üzere iki ana parçadan oluşmaktadır. Bu yazılımda, test ortamı kullanıcı tarafından oluşturulmaktadır. Kullanıcı, arayüzünde tanımladığı parametreler ile simülasyon ortamını, testini koşturacağı modüle göre özelleştirilmektedir. Kullanıcının girdiği parametre değerleri, XML formatında veri tabanında tutulmaktadır. Kullanıcı parametre değerlerini değiştirerek veya yeni parametre tanımlayarak, sistemde varsayılan XML dosyalarını güncelleştirilebilir. Böylece, kullanıcı istediği test ortamını parametre bazında yapılandırılmaktadır.

F-SIM yazılımının Ana Simülasyon Paketi bölümünde, uçuş dinamik modülleri (helikopter dinamik modülü, uçak dinamik modülü vb) bulunmaktadır. Kullanıcı, istediği dinamik modülünü simulasyona ekleyebilir. Bu dinamik modüller kullanıcı arayüzünden gelen parametrelerle özelleştirilmektedir. Örneğin; testini yapacağı yazılım veya donanım modülü F-16 platformunda test edilmesi gerekiyorsa, bununla ilgili olan parametre değerlerini arayüzden girerek, ana simülasyonda bulunan uçak dinamik modülünü F-16 platformu şeklinde özelleştirmektedir. Yazılımın bu özelliği, bilgisayar ortamında helikopter, sabit kanatlı, İHA vb. platformları hızlı ve kolay simüle edebilmeyi sağlamaktadır.

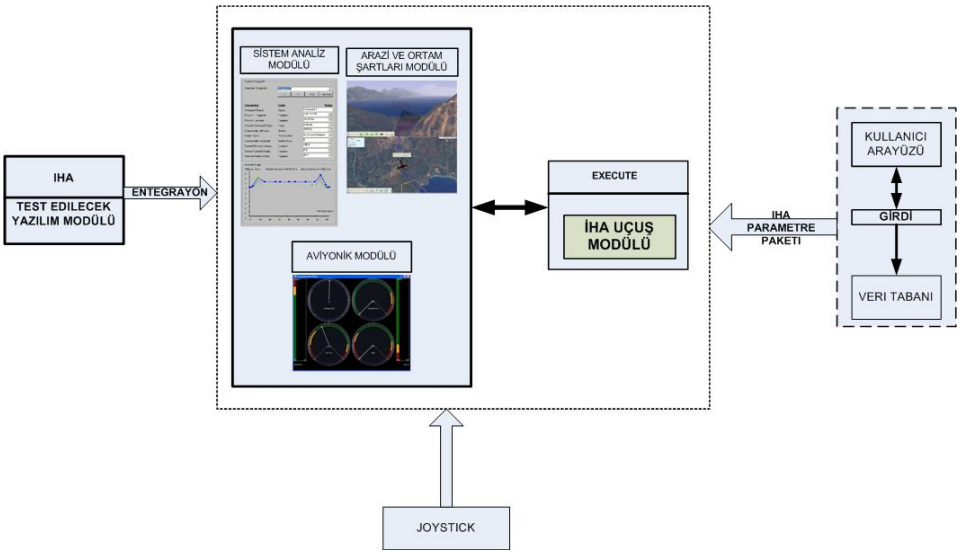
Simülatör ortamında aynı zamanda Sistem Analiz, Arazi-Ortam Şartları ve Aviyonik modülleri gibi modüller vardır. Simülasyon ortamında, kullanıcı arayüzünden gelen parametreler ile tanımlanan platform şekillenmektedir. Amaç, test edilecek olan yazılım modülünün, kullanıcı tarafından analizini kolaylaştırmaktır. Simülatör ortamı oluşturulduktan sonraki test edilecek modül, simülasyon ortamına entegre edilmektedir.

Sistem Analiz modülü, yazılım modülünün testi sırasında oluşan sayısal değerleri, grafik ve tablolar ile kullanıcıya görsel olarak sağlamaktır. Arazi-Ortam Şartları modülü, tanımlanan uçan platformun arazi üzerindeki hareketini ve değişen ortam şartlarına göre verdiği tepkileri simüle etmektedir. Yüklenen arazi modülü 2 boyutlu ve

3 boyutlu olarak kullanıcının karşısına gelmektedir. Kullanıcı simülasyonun şekline bağlı olarak simülasyon ile etkileşir. Bu bir IHA için bir laptop istasyonu olabilir veya bir eğitim simülatörü için bir joystick. Böylece test edilmek istenilen yazılım modülünün görsel anlamda platforma olan etkilerinin gözlemlenme imkanı olmaktadır. Aviyonik modül, oluşturulan uçuş platformuna bağlı olarak kullanıcı tarafından seçilebilmektedir.

Aviyonik penceresinde, seçilen aviyonik objeleri aviypnik kütüphanesinden seçilir, test penceresine konulur ve fare yardımıyla biribirlerine bağlanarak uçuş aviyonik sistemi oluşturulabilmektedir. Böylece kullanıcı, simüle etmek istediği platformun aviyoniklerini istediği kombinasyonda oluşturabilmektedir. Simülasyon çalıştırıldığında, test yapılan yazılım modülünün, aviyonikler ile etkileşimi gözlemlenmiş olur. Parametrelerde oluşan değişikliklere ise ana penceredeki Parametre İzleme butonu kullanılarak erişilimektedir.

Bir IHA'ya eklenmesi gereken bir yazılım olduğunda F-SIM simülasyonu ortamında F-SIM kütüphanesi kullanılarak önce bir IHA platformu oluşturulmaktadır. Buna göre arayüzden gelen parametrelerle ana simülasyon yazılımında IHA uçuş dinamik modülü şekillenmektedir. Bundan sonra yine yazılım kütüphanesinden sensör dosyaları, aktuatör dosyaları çağırılıp İHA' nın hem dinamik modeli hem de sensörden servolara kadar uzanan zincirin her bir halkası modellenmektedir Böylece test edilmesi planlanan yazılım modülü, gerçek sistemdeki işleyişe göre çalışmaktadır. (Şekil 4)



Şekil 4: İHA Yazılım Modül Entegrasyonu.

İHA'da koşturulması planlanan yazılım modülünün, gerçek İHA'ya entegrasyonunda yaşanılan donanımsal ve yazılımsal hataları, F-SIM yazılımında tespiti edilebilir ve yazılım modülünde gerekli düzeltmeler yapılabilmektedir. F-SIM simülasyon yazılımında hazırlanan modül, IHA 'da çalışabilecek formata çevrilir ve gerçek ortama yüklenmektedir.

Yazılım mimarisi, IEEE1516 HLA-RTI dağıtık simülasyon mimarisi ile uyumlu olup, farklı ortamlarda donanımsal olarak oluşturulan uçuş test platformları için emülatör şeklinde de kullanılabilmektedir.

3.1 F-SIM Simülasyon İşletim Sistemi ve Donanım Konfigürasyonu

F-SIM yazılımı platformdan bağımsız ortamda geçek zamanlı işletim sistemlerinde çalışmaktadır. UNIX tabanlı işletim sistemlerinde çalışmakta (LINUX, QNX, SUN SOLARIS vb.) olup Window tabanlı işletim sistemleri için Cygwin kurunumu gerekmektedir. Minimum donanım konfigürasyonları; Pentium 1.4 Ghz işlemci, 512 MB RAM ve GPU programlama kabiliyetine sahip grafik kartı olmalıdır. Sistemin oluşturduğu çalıştırılabilir modüller, arayüzü sayesinde PC104 gibi Tek Kartta Bilgisayarlara (SBC-Single Board Computer) yüklenebilir ve çalıştırılabilir.

4. SONUÇ

Kontrol tasarımlarının geliiştirilmesi ve test edilebilmesi için ortak bir simulasyon platformu önerilmektedir. Bu platform ile hem dinamik modül kütüphanesiden uçan platformun dinamiği oluşturulur hem de, bir aviyonik paketinin sinyal seviyesinde modellenmesine izin verilir. Bu aviyonik paketinin simulasynu F-sim içersinde yapılabileceği gibi, başka dilierde yazılmış halleri de F-Sim simulasyonuna entegre edilebilir. F-Sim simulasyonu hala geliştirilmekte olup, bazı kısıtlı modülleri ile kullanıma hazırdır.

KAYNAKÇA

[1] Eric N. Johnson and Daniel P. Schrage. (January 2004), "System Integration and Operation of a Research Unmanned Aerial Vehicle", AIAA Journal of Aerospace Computing, Information and Communication, 1(1):pp- 5-18.

[2] Suresh Kannan, J. V. R. Prasad, Daniel P. Schrage, Ilkay Yavrucuk, Linda Wills, and Carlos Restrepo (1999) "Simulation and Flight Control Integration using the Open Control Platform for Unmanned Aerial Vehicles", In AIAA Digital Avionics Conference.

A2 OTURUMU:

BİRLİKTE ÇALIŞABİLİRLİK

C4I İLE M&S SİSTEMLERİNİN BİRLİKTE ÇALIŞABİLİRLİĞİNE MUHAREBE YÖNETİM DİLİ YAKLAŞIMI
Ahmet KANDAKOĞLU, İlker AKGÜN, Erkan ERTUĞRUL

DÖNGÜDE DONANIMSAL BENZETİM TEKNİĞİ VE BİLEŞENLERİ
Erkan DUMAN, Hayrettin CAN, Erhan AKIN

MilSOFT DDS ARAKATMANI VE DDS'İN SAVAŞ YÖNETİM SİSTEMLERİNDE SİMÜLASYON AMAÇLI KULLANIMI
Hüseyin KUTLUCA, İzzet Emre ÇETİN, Ertan DENİZ, Barış BAL

FARKLI SİMÜLASYON SİSTEMLERİNİN ENTEGRASYONU İÇİN KONTROL MİMARİSİ
Deniz ALDOĞAN, Seniha KÖKSAL, Cemil AKDEMİR, İsa TAŞDELEN, Oğuz DİKENELLİ

C4I İLE M&S SİSTEMLERİNİN BİRLİKTE ÇALIŞABİLİRLİĞİNE MUHAREBE YÖNETİM DİLİ YAKLAŞIMI[3]

Ahmet KANDAKOĞLU [(a)], İlker AKGÜN [(a)], Erkan ERTUĞRUL[(a)]

[(a)] Harp Akademileri Komutanlığı, HOSİM Başkanlığı, 34334, İstanbul,

{kandak, iakgun, eertugrul}@harpak.tsk.mil.tr

ÖZ

Günümüzün modern Silahlı Kuvvetleri bünyesinde değişik amaçlar doğrultusunda farklı yazılım mimarileriyle birbirinden bağımsız olarak geliştirilen bir çok bilgi sistemi bulunmaktadır. Bu sistemlerin Ağ Merkezli Harp (Network Centric Warfare - NCW), karargâh eğitimi, görev provası ve plan analizi gibi ihtiyaçlardan dolayı barış, tatbikat, kriz ve harp ortamlarında birlikte çalışabilirliğinin önemi daha da artmaktadır. Fakat, anılan sistemlerde veri modeli ve bilgi değişim standartlarına uyumun arzu edilen seviyede dikkate alınmamasından dolayı, birlikte çalışabilirlik güncel ve kritik bir sorun olarak karşımıza çıkmaktadır. İşte, Muharebe Yönetim Dili (Battle Management Language – BML) bu soruna çözüm getirmek amacıyla halen geliştirilmekte olan uluslararası bir standarttır.

BML, askeri harekât icra eden birlik ve sistemlerin komuta kontrolü ile durumsal farkındalığı ve paylaşılan ortak taktik resmi elde etmek için kullanılan bir dildir. BML bir bakıma, gerçek ve simüle edilmiş birlikler ile gelecekteki robot sistemlerin kullanabileceği emir, istek ve raporların standart bir gösterimi olarak da düşünülmektedir. Bununla birlikte BML; Komuta, Kontrol, Muhabere, Bilgisayar ve İstihbarat (C4I), Modelleme Simülasyon (M&S) ve robot sistemlerin ortak bir veri modeli doğrultusunda geliştirilmesini ve böylelikle birlikte çalışabilirlik yeteneğinin daha yazılım mimarisinin tasarımı aşamasında kazanılmasını sağlamaktadır.

Bu çalışmada, BML yaklaşımı ve buna yönelik literatür incelemesi sunulmaktadır. Ayrıca; günümüz modern Silahlı Kuvvetlerin ihtiyaçları gözden geçirilerek önemli hususlara dikkat çekilmekte ve bu ihtiyaçları karşılamak için yapılması gerekenler vurgulanmaktadır.

Anahtar Kelimeler: Birlikte Çalışabilirlik, C4I, Modelleme & Simülasyon (M&S), Muharebe Yönetim Dili, Robot Sistemler.

[3] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup Harp Akademileri Komutanlığı'nı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

BATTLE MANAGEMENT LANGUAGE APPROACH FOR C4I AND M&S SYSTEMS INTEROPERABILITY

ABSTRACT

Today's modern Armed Forces have various information systems developed independently with different software architectures for different purposes. Due to Network Centric Warfare (NCW), training, mission rehearsal and planning analysis requirements, interoperability of these information systems gets more important in peace, exercise, crisis and battle environments. However, because of paying insufficient attention to data models and data exchange standards compatibility in these systems, interoperability remains as a crucial and critical problem. Thus, Battle Management Language (BML) is a universal standard that is being developed as a solution to this problem.

BML is the unambiguous language used to command and control forces and equipment conducting military operations and to provide for situational awareness and a shared, common operational picture. In other words, BML can be defined as a standard representation of commander's orders, intents and reports to be understood by both live forces and automated systems, for simulated and real-world operations. In addition to this, BML provide development of Command, Control, Communication, Computer and Intelligence (C4I), Modeling Simulation (M&S) and robotic systems through a common data model. In this way, BML yields the achievement of interoperability capability in the system design phase.

In this study, the BML approach and its literature review are presented in detail. Furthermore, important points are highlighted following the examination of the requirements of Armed Forces and the actions to meet these requirements are emphasized.

Keywords: Interoperability, C4I, Modeling & Simulation (M&S), Battle Management Language (BML), Robotic Systems.

1. GİRİŞ

Silahlı Kuvvetler, sivil organizasyonlar gibi farklı amaçlar doğrultusunda, farklı yazılım mimarileriyle geliştirilmiş birçok bilgi sistemini bünyesinde barındıran büyük bir organizasyon olarak düşünülebilir. Bu bilgi sistemlerinin en önemlileri arasında Komuta, Kontrol, Muhabere, Bilgisayar ve İstihbarat sistemleri (C4I) ve eğitim maksatlı kullanılan Modelleme Simülasyon (M&S) sistemleri gelmektedir. C4I; günümüz silahlı kuvvetlerinin barış, tatbikat, kriz ve harp şartlarında; harekâtın planlanması, sevk ve idaresi, bilgi aktarımı ve mesajlaşma fonksiyonlarını sağlayan bir bilgi sistemi olarak ifade edilebilir. M&S sistemleri ise; günümüzde ülkelerin gündeminde bulunan kuvvet indirimi, savunma bütçesindeki azalma ve çevre kısıtlamaları dikkate alındığında, gerçek koşullarda icra edilen tatbikatlara oranla daha ucuz ve gerçeğe yakın bir eğitim ortamı sağlayan bilgi sistemleridir.

Askerî karargâhlar Bilgisayar Destekli Tatbikatlarda (BDT) M&S sistemleri kullanılarak eğitilirler. Fakat, bir tatbikatta sadece M&S sistemlerinin kullanılması

karargâhların eğitim hedeflerine ulaşmasında yeterli olamamaktadır. Son yıllarda "savaşır gibi eğitim yap!" prensibi daha da önem kazanmıştır. Karargâhlar, BDT'lerde M&S sistemlerinin sağladığı arayüzler yerine gerçek hayatta sahip oldukları C4I sistemlerini kullanmak istemektedirler. BDT'lerde bir C4I sistemini başarılı bir şekilde kullanmak, M&S ve C4I sistemleri arasında bilgi değişimini ön plana çıkarmakta ve böylelikle bu sistemlerin birlikte çalışabilirliği bir gereksinim haline gelmektedir.

Birlikte çalışabilirlik; "bir sistemin ya da sürecin, ortak standartlar çerçevesinde diğer bir sistemin ya da sürecin bilgisini ve/veya işlevlerini kullanabilme yeteneği" olarak tanımlanabilir [1,2]. Birlikte çalışabilirliğe yönelik problem sahaları olarak; anlamsal (semantik) farklılıklar, yazılım ve donanım sistemleri arasındaki farklılıklar, sınırlı ve sınırsız yazılım sistemleri arasındaki farklılıklar ve güvenlik ihtiyaçları olarak gruplanabilir [3] (Sınırlı sistemler, sistemden beklentilerin ve tüm sistem bileşenlerine ait işlevselliğin bilindiği ve bileşenlerin yapısal olarak düzgün şekilde gruplandığı sistemlerdir. Sınırsız sistemler ise farklı sistemlerle etkileşim içinde olan ve eksik bilgi varlığında bile belirli işleri yapmaları beklenen sistemlerdir).

Bilgi teknolojilerinin hızlı gelişimi, günümüz Silahlı Kuvvetleri için Ağ Merkezli Harp (Network Centric Warfare - NCW) kavramını ortaya çıkarmıştır. NCW'nin vizyonu; muharebe gücünü büyük ölçüde artırmak ve iyi bilgilendirilmiş ve coğrafi olarak geniş bir alana yayılmış durumdaki kuvvetleri oluşturmak maksadıyla; sensörleri, C4I sistemlerini ve vurucu platformları birleştirmek suretiyle, bilgi üstünlüğünü yaratmak ve bu üstünlükten faydalanmaktır [4]. NCW'de; sensör, silah sistemi, C4I sistemi, karar verici gibi unsurların bütünleşik bir sistem gibi birlikte çalışabilirliği söz konusu olmaktadır. Bunun sonucunda; komutanın emir ve niyetinin daha iyi anlaşılabilmesi, etkin müşterek planlama, yatay ve düşey senkronizasyon, paylaşılmış durumsal farkındalık, eş zamanlı etki ve operasyonel çeviklik elde edilmiş olacaktır.

Ayrıca, gelişmekte olan teknolojiler ışığında, günümüzde prototip aşamasında olan robot sistemlerin kullanımına temel olacak yazılım mimarilerinin hazırlanması gerekmektedir.

İşte, söz konusu ihtiyaçlar doğrultusunda, birlikte çalışabilirliği elde etmek üzere uluslararası bir dil olarak Muharebe Yönetim Dili (Battle Management Language – BML) geliştirilmektedir. BML; komutan ve karargâhının hazırlamış olduğu harekât planlarının, alt ve üst birlikler arasında açık, net ve hızlı bir biçimde alınıp verilmesine ve müşterek bir ortamda farklı kuvvetlerin (kara, deniz ve hava) birleşik bir ortamda (koalisyon ortamı) ise farklı ülkelerin harekât plânlarını birbirleriyle paylaşmasına imkân sağlayacaktır. Bu yüzden, BML'in; kuvvet, müşterek ve koalisyon boyutlarını kapsaması gerekmektedir. Kullanılan bilgi sistemleri farklı ülkelere ait olabileceği gibi, belirli bir kuvvete özgü de olabilmektedir.

Bu çalışmada; C4I, M&S ve robot sistemlerin ortak bir veri modeli ve veri standardı doğrultusunda geliştirilmesi ve böylelikle birlikte çalışabilirlik yeteneğinin daha yazılım mimarisinin tasarım aşamasında kazanılmasına yardımcı olabilecek, halen uluslararası bir çalışma grubu tarafından geliştirilmekte olan BML yaklaşımı sunulmaktadır. Bu çalışma bir bakıma BML konusunda bir literatür araştırması olarak da görülebilir.

Çalışma bütününde izlenen metodoloji şu şekildedir. Bir sonraki bölümde; BML'in tanımı, amacı, kapsamı, genel prensipleri, rolü ve katmanları detaylı olarak

incelenmiştir. Ayrıca, bu bölümde BML ile ilgili literatürde yer alan çalışmalar özet olarak sunulmuştur. Son bölümde ise çalışmanın genel bir değerlendirilmesi yapılmış ve müteakip çalışmalara yönelik önerilerde bulunulmuştur.

2. MUHAREBE YÖNETİM DİLİ (BML)

BML, askeri görevleri icra eden birlik ve teçhizatların komuta kontrolü ile durumsal farkındalığı ve paylaşılmış müşterek taktik resmi elde etmek için kullanılan açık ve net bir dil olarak tanımlanmaktadır [5].

Bu tanım doğrultusunda, BML aşağıdaki dört ana koşulu sağlamalıdır:

1. Açık ve net bir yapıda olmalıdır,
2. Komutanın niyetini tam olarak ifade edebilmesini engellememelidir,
3. Mümkün olduğunca C4I veri gösterimlerini kullanmalıdır,
4. Durumsal farkındalığı ve paylaşılmış müşterek ortak taktik resmi elde etmek için bütün unsurların kendileriyle, görevleriyle ve ortamlarıyla ilgili verilerinin iletişimine olanak sağlamalıdır.

Bu prensipler veya diğer bir ifade ile koşullar, gelecek BML çalışmaları için önemli bir yol göstericidir. Mevcut BML konsepti; askerî görevleri icra eden birlik ve sistemlerin komuta kontrolünü içermekte, simüle edilmiş veya gerçek kuvvetlere herhangi bir kısıtlama getirmemektedir. Diğer bir ifade ile; BML, gerçek ve simüle edilmiş kuvvetler arasında "savaşır gibi eğitim yap" prensibine uygun olarak herhangi bir farklılık içermemektedir. Böylelikle; komuta heyeti ast birlikleriyle, simülasyonla veya robot sistemlerle aynı BML aracılığıyla iletişim kurmaktadır.

2.1. BML Boyutları

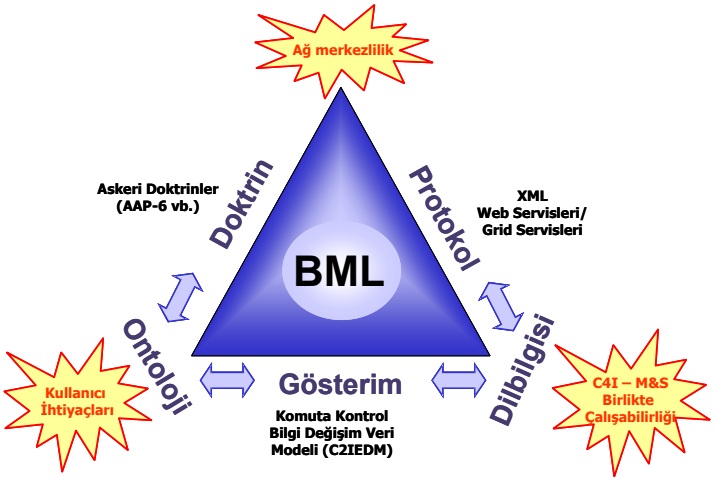
Başlangıçta; BML'in, kusursuz bir şekilde birlikte çalışabilirliği sağlamak maksadıyla üç farklı boyutunun (doktrin, gösterim ve protokol) olması gerektiği tespit edilmiştir [6]. Bu üç boyut, BML'in prensipleri olarak ifade edilen dört ihtiyacı karşılamanın yanısıra birbirlerini de destekler durumdadır. Genel olarak, birbirini destekleyen bu üç boyut BML üçgeni olarak tanıtılmıştır. Daha sonra, söz konusu üç boyutun BML'in amacına ulaşması için yeterli olmadığı vurgulanarak, 4 ve 5'inci boyutları (dilbilgisi ve ontoloji) ile genişletilmiş BML üçgeni tanımlanmıştır (Şekil 1) [7].

2.1.1. Doktrin

BML içerisinde kullanılan her bir terim açık ve net olarak ifade edilmeli ve askeri doktrinlerde yer almalıdır [5]. Diğer bir ifade ile, doktrin; BML'de yer alan bütün terim ve bu terimlerin açık ve net açıklamasından oluşan bir sözlüktür. Bu sözlük; tek bir kuvvetin doktrini ile kısıtlı kalmamalı, diğer ülke ve kuruluşların doktrinlerinin de tanımlanabilmesini desteklemelidir.

2.1.2. Gösterim

Gösterim boyutu; doktrinlerde yer alan terimleri, icra edilecek vazife ve görevlerin tanımlamalarında yer alacak şekilde düzenlemekte ve aralarındaki bağlantıları kurmaktadır. Bir vazife, planlı bir şekilde icra edilmesi gereken bir dizi görev ile ifade edilmektedir. Hâlihazırda; NATO çalışma grupları tarafından geliştirilmiş olan C2IEDM (Command & Control Information Exchange Data Model)'in genişletilmiş



Şekil 1. Genişletilmiş BML üçgeni

sürümünün bu maksatla kullanılması plânlanmakla birlikte, diğer bazı alternatiflerin de kullanımı değerlendirilmektedir [8].

2.1.3. Protokol

Plan, emir ve raporların BML gösterimlerinin iletilebilmesi için protokollere ihtiyaç duyulmaktadır. Protokol boyutu, icra edilecek vazife ve görev olarak verilen askeri emirlerin BML tanımlamalarının kaynak sistemden hedef sisteme aktarılması yöntemini standartlaştırmaktadır. Burada, hedef sistem başka bir C4I sistemi olabileceği gibi bir simülasyon sistemi veya bir robot da olabilmektedir. Web standartları ve grid standartları şu an itibariyle en çok gelecek vadeden adaylardır. Özellikle, veri alışveriş ihtiyaçlarını tanımlamak için XML kullanımı temel ilke olarak düşünülmektedir.

2.1.4. Dilbilgisi

BML dilbilgisi; BML gösteriminde yer alan harekat ihtiyaçları doğrultusunda, bilgi değişimine olanak tanıyabilecek BML protokolünün genişletilmiş bir biçimini elde etmek için kullanılmaktadır. Ayrıca; BML gösterimi, aynı anlama gelen farklı yapıların veya değişen protokollerin kullanımını da sağlamaktadır. Dilbilgisinin BML'i nasıl desteklediği ile ilgili geniş bilgi [9]'da sunulmaktadır. İlgili çalışmada; BML Dilbilgisi, "veri sözlüğünde yer alan bir dizi kelime ve bu kelimeler kullanılarak uygun cümleler oluşturmak için bir dizi kural" olarak tanımlanmaktadır [10].

2.1.5. Ontoloji

Doktrinlerden elde edilen BML Ontolojisi, "Muharebe Yönetim alanının kavramsallaştırmasının biçimsel bir tanımlaması" olarak ifade edilmektedir [10]. BML Ontolojisi; veri sözlüğünde tanımlanan her bir terimi, bunların temelinde yer alan kavramları ve ilişkileri vb. içermektedir. Veri modelleri ile ontoloji arasındaki önemli bir fark; veri modelleri sembolik bilgiyi (veri yapısı ve söz dizimi) gösterirken, ontolojinin, içeriksel bilgi (verinin anlamı) ve veri elemanının kavramsal ve gösterimsel seviyesi ile ilgilenmesidir.

2.2. BML Katmanları ve LCIM Seviyeleri

Genişletilmiş BML üçgeni; artık doktrin, gösterim, protokol, ontoloji ve dilbilgisi kenarlarından oluşan bir beşgen olarak düşünülmektedir [7]. Fakat, bu haliyle bu beş boyutu birbirinden kesin sınırlarla ayırmak mümkün olamamaktadır. Bu sebeple, BML boyutları üçgen yerine katmansal bir yapı ile ifade edilmektedir [7]. LCIM modelinin seviyeleri ile BML katmanlarının karşılaştırılması Şekil 2'de gösterilmektedir. Çalışma kapsamında tanıtılan BML, LCIM modelinin son seviyesi olan kavramsal düzeyde birlikte çalışabilirlik seviyesini hedeflemektedir.

LCIM, Seviye 0 ile başlayıp Seviye 6 ile sonlanan yedi seviyede açıklanmaktadır [7]. Seviye 0'dan seviye 6'ya doğru ilerledikçe birlikte çalışabilirlik yeteneği artmaktadır. Şekil 2'te LCIM seviyeleri ile bu seviyelerin entegre edilebilirlik, birlikte çalışabilirlik ve bütünleşebilirlik gibi diğer kavramlar ile bağlantısı gösterilmektedir.

2.3. BML'in Rolü

BML'in sağlayacağı fayda ve katkılar genel olarak aşağıdaki özetlenebilir [11]:

- Karargâh personelinin sistemler arasında veri tercümanlığı ile uğraşmak yerine eğitim faaliyeti üzerine yoğunlaşabilmesi.
- C4I ve simülasyon sistemleri arasında yer alan entegrasyon amaçlı yazılım arayüzlerinin azaltılması veya ortadan kaldırılması.
- Eğitim ve harekâtı başarı ile gerçekleştirmek maksadıyla, birlik ve silah sistem bilgilerinin sisteme sadece bir defa girilmesi için ihtiyaç duyulan zaman ve iş gücünün azaltılması.
- Farklı ülkelere ait farklı birlikler arasında müşterek bir anlayışın sağlanması.
- Doktrin ile bağlantılı çok iyi biçimlendirilmiş mesajların kullanımı ile birlikler arası iletişimin daha kısa, özlü, doğru ve kusursuz olmasının sağlanması.
- Çok iyi yapısallaştırılmış, veri odaklı mesajlaşma sistemi sayesinde bilginin rahatlıkla işlenebilir ve uygulamaya sokulabilir olarak sunulması ile hem simülasyon hem de robot sistemlerde "Zeki Etmen" (Intelligent Agent) yazılımları kullanımının sağlanması.

Şekil 2. LCIM (solda) ve BML (sağda) seviyeleri [7]

2.4. Literatür İncelemesi

Bu çalışma kapsamında, BML ile ilgili literatürdeki araştırmalar detaylı olarak incelenmiş ve özetleri Çizelge 1'de verilmiştir.

3. SONUÇ VE DEĞERLENDİRME

Bilgi sistemlerinin hayatî öneme sahip olduğu günümüz Silahlı Kuvvetleri için; Ağ Merkezli Harp, karargâh eğitimi, görev provası ve plan analizi gibi ihtiyaçlardan dolayı söz konusu sistemlerin birlikte çalışabilirliği bir zorunluluk haline gelmiştir. Güncel ve kritik bir konu olan birlikte çalışabilirlik ile ilgili olarak literatürde yer alan araştırma ve geliştirme çabalarının ortaya koyduğu farklı yöntem ve yaklaşımlar bulunmaktadır. Son yıllarda önemi giderek artan BML, bu yaklaşımların başında gelmektedir.

BML; M&S, C4I ve robot sistemlerin hem ortak bir veri modeli doğrultusunda geliştirilmesini hem de ortak bir dil ve protokol aracılığıyla veri değişimini gerçekleştirmesini öngören bir yaklaşım olarak karşımıza çıkmaktadır. Diğer bir ifade ile; BML, birlikte çalışabilirlik yeteneğinin, sistemlere sonradan bir bileşen olarak değil, daha yazılım mimarisinin tasarımı aşamasında bir temel unsur olarak kazandırılmasını sağlamaktadır. Bu açıdan değerlendirildiğinde, diğer birlikte çalışabilirlik yöntem ve yaklaşımlarından tamamen farklı bir boyutta ele alınması gerekmektedir. Ayrıca, LCIM'in en üst seviyesi olan kavramsal birlikte çalışabilirliği hedefleyen bir yaklaşım olması ve uluslararası çalışma grupları (SISO ve NMSG-048 grupları) tarafından geliştiriliyor olması, BML'i diğer benzer çalışmalardan ayıran önemli hususlardır.

Başta A.B.D. olmak üzere bir çok ülkede çeşitli ihtiyaçlardan dolayı bir zorunluluk haline gelen birlikte çalışabilirliği sağlama gayretleri; entegrasyon ve uyuşum problemleri, bakım/idame güçlükleri ve yazılım tekrarları nedenleriyle büyük miktarda kaynak israfına sebep olmaktadır. Bu durumdan ders alan ülkeler, kaynak israflarını en aza indirmek ve birlikte çalışabilirliği en üst düzeyde sağlamak amacıyla BML çalışmalarını yakından takip etmektedirler. BML için test ortamlarını oluşturan bu ülkeler, kendi ihtiyaçları doğrultusunda prototip çalışmalar gerçekleştirmekte, elde edilen sonuçlar ve sonradan oluşan ihtiyaçlar çerçevesinde BML çalışmalarına yön vermekte ve yakın gelecekteki M&S, C4I ve robot sistemlerin, BML'in ihtiyaç duyduğu yazılım mimarisi doğrultusunda geliştirilmesi gerektiğini vurgulamaktadırlar.
Prototip çalışmaların sonuçları incelendiğinde, BML'in uygulamaya geçirilmesi esnasında dikkat edilmesi gereken hususlar aşağıdaki gibi ifade edilebilmektedir:

- BML sayesinde insanlar ile sistemler arasındaki etkileşim azalacaktır. Bu durumun komuta kontrol esnekliğinde değişikliğe yol açmaması gerekmektedir.
- Gerçek hayata uygulanabilirliği ve maliyet etkinliği iyice araştırılmalıdır. Çünkü, BML'in özümsenmesi için ihtiyaç duyulan zaman, sistemlerin geliştirilme sürelerini doğal olarak uzatmaktadır.
- C2IEDM veritabanına ve BML dilbilgisine bağımlılığın söz konusu olduğu ve uluslararası standartlarda yapılacak değişikliklerin mevcut bütün sistemlere yansıtılması gerektiği unutulmamalıdır.
- C4I ile M&S sistemler arasındaki veri uyumluluğu kritik bir husustur. Çünkü, uluslararası bir veri modeli olan C2IEDM, ulusal ihtiyaçlar açısından eksik kalabilmekte ve ulusal doktrin ve yönergeleri tam olarak yansıtamamaktadır.

Çizelge 1. BML ile ilgili literatürde yer alan çalışmalar

No	Ref.	Yıl	Çalışmanın özeti
1	[12]	1998	Komutan niyetinin yapısal bir formatta ifade edilmesine önem çeken bir çalışmadır. Çalışma ABD Ordusunun mevcut BML çalışmalarına temel teşkil etmektedir.
2	[5]	2001	Bu makale BML geliştirmenin önemli kavram ve prensiplerini ortaya koymaktadır. Emir ve planları açık ve net hale getirmek için geliştirilen KKBS standart veritabanlarının kullanılması fikri bu makalede ortaya atılmaktadır.
3	[13]	2002	BML kavramının Birleşik ve Müşterek Harekat için genişletilmesini içermektedir.
4	[8]	2003	Simülasyon ile KKBS arasında komuta kontrol bilgilerinin karşılıklı değişimi için C2IEDM kullanımı üzerine bir çalışmadır.
5	[14]	2003	Gelecekteki komuta kontrol mimarisinde, hem komuta kontrol hem de simülasyon sistemleri için BML ortak bir model olarak önerilmektedir.
6	[15]	2003	Senaryo üretimi için BML'in uygulanabilirliği gösterilmektedir.
7	[16]	2004	ABD ordusunun BML prensiplerine yönelik denemelerin nasıl standartlaştırılıp (C2IEDM kullanılarak) genişletildiğini (XMSF protokolü ile) açıklayan bir makaledir.
8	[17]	2004	C-BML uygulamaları için C2IEDM'in standartlaştırılmasını ve Hal Tarzı Analizi (Course of Action Analysis - COAA) sistemleri için BML kullanımını önermektedir.
9	[18]	2004	5N'ye 'Hangi' ve 'Neden' ekleyerek BML'i geniş kapsamlı hale getiren ABD-Almanya İşbirliği Programı SINCE'i tanıtmaktadır.
10	[19]	2004	ABD Tugay/Tabur/Bölük Harekat emirlerinin gösterimi üzerine yoğunlaşarak ABD ordusu BML prensipleri denemelerini incelemektedir.
11	[20]	2004	BML'in koalisyon faaliyetleri için geliştirilmesini önermektedir.
12	[21]	2004	XBML prototipini tanıtmakta ve koalisyon faaliyetleri için BML'in geliştirilmesini değerlendirmektedir.
13	[11]	2004	İngiltere'nin BML diline olan ihtiyacı ortaya konmaktadır.
14	[22]	2004	BML gösterimi için C2IEDM veri modelinin kullanımı üzerine teknik bir rapordur.
15	[23]	2004	C-BML teknik alanını oluşturmak için NATO'ya yapılan önerileri içermektedir.
16	[24]	2005	BML'in C2IEDM'de 5N kullanılarak yapılandırılması çalışmasıdır.
17	[25]	2005	Fransızların BML kullanımının tanıtılmasını ve BML ile COAA sistemlerini desteklemeye yönelik önerileri içermektedir.
18	[26]	2005	XBML prototipinin 5N ve C2IEDM kullanılarak Kara Kuvvetlerinden farklı olarak Hava Kuvvetlerine uygulanmasını gösteren bir makaledir.
19	[27]	2005	BML'in simülasyon davranışlarını nasıl etkilediği incelenmektedir.
20	[28]	2005	SISO içerisinde C-BML geliştirme çalışmalarına yönelik öneriler ifade edilmektedir.
21	[29]	2005	XBML çalışmasının C2IEDM tabanlı WEB servisleri ile nasıl standartlaştırılabileceğini anlatmaktadır.
22	[30]	2005	"Global Information Grid (GIG) üzerinde COAA" kullanımında BML'in önemli bir bileşen olarak kullanımını içermektedir.
23	[6]	2005	BML çalışma grubunun faaliyet ve önerilerini içeren sonuç raporunun özetidir.
24	[7]	2006	BML'in; Doktrin, Gösterim ve Protokol boyutlarına ek olarak 4'üncü ve 5'inci boyutu oluşturacak olan Dilbilgisi ve Ontoloji boyutlarını tanıtmaktadır.
25	[10]	2006	BML geliştirmesi safhalarında her bir boyut için uygulanabilir çeşitli yaklaşımlar belirlenmekte ve özetlenmektedir.
26	[9]	2006	Askerî emirlerin oluşturulması için uygun BML cümleciklerini oluşturacak başlangıç seviyesinde BML dilbilgisi önerilmektedir.
27	[31]	2006	BML kavramının arazi ve hava durumu bilgilerini de içermesini önermektedir.
28	[32]	2006	Karmaşık harekâtın komuta kontrolü için biçimsel bir dilbilgisi sunmaktadır.

- Diğer ülke ve çalışma grupları ile BML geliştirme faaliyetleri konusunda işbirliği yapılmalıdır.

BML'in uygulamaya geçirilmesi sürecinde; halihazırda kullanılan sistemlerden tamamen vazgeçilip, yerlerine yenilerinin, henüz gelişimini tamamlamamış bu yazılım mimarisi göz önüne alınarak sıfırdan geliştirilmesinin zaman, maliyet, uygulanabilirlik ve güvenilirlik açısından mümkün olamayacağı gayet açıktır.

Sonuç olarak;

- M&S ve C4I sistemlerinin birlikte çalışabilirlik ihtiyaçlarının belirlenmesinin,
- Geliştirilen ve/veya planlanan M&S ve C4I projelerinin BML dikkate alınarak yeniden gözden geçirilmesinin,
- BML'in uygulamaya geçirilmesi maksadıyla;
  - Henüz geliştirilmeye başlanmamış M&S ve C4I projelerinin BML tabanlı olarak geliştirilmesinin,
  - Mevcut sistemlerin hâlihazırdaki yapısının korunmasının,
  - BML tabanlı olarak geliştirilecek yeni sistemler ile mevcut sistemler arasında birlikte çalışabilirliği sağlamak maksadıyla BML arayüzlerin geliştirilmesinin ve,
- Diğer ülkelerdeki BML geliştirme faaliyetlerinin yakından takip edilerek, elde edilecek bilgi birikiminden istifade edilmesinin uygun olacağı değerlendirilmektedir.

KAYNAKÇA

[1] IEEE, "IEEE Standard Computer Dictionary", New York, 1990.
[2] Tolk, A., Muguira, J., "The Levels of Conceptual Interoperability Model",03F-SIW-007, Fall Simulation Interoperability Workshop (SIW), 2003.
[3] "Some Current Approaches to Interoperability", Technical Note, CMU/SEI, 2005.
[4] The Joint Staff, J6, Observations on the Emergence of Network Centric Warfare, http:// www.dtic.mil/jcs/j6/education/warfare.html
[5] Carey, S., Kleiner, M., et al., "Standardizing Battle Management Language – A Vital Move Towards the Army Transformation",01F- SIW-067, Fall SIW, 2001.
[6] Blais, C., Galvin, K., and Hieb, M. R., "Coalition Battle Management Language (C-BML) Study Group Report", 05F-SIW-041, Fall SIW, 2005.
[7] Turnitsa, C.D. and Tolk. A., "Battle Management Language: A triangle with Five Sides",06S-SIW-016, Spring SIW, 2006.
[8] Khimeche, L., and de Champs, P., "Courses of Action Analysis and C4I-Simulation Interoperability", 03F-SIW-028, Fall SIW, 2003.
[9] Schade U. and Hieb, M. R., "Formalizing Battle Management Language: A Grammar for Specifying Orders",06S-SIW-068, Spring SIW, 2006.
[10] Tolk. A., Diallo S., et al., "Merging Protocols, Grammar, Representation, and Ontological Approaches in Support of C-BML",06F-SIW-008, Fall SIW, 2006.
[11] Galvin, K., and Cox, A., "Does the United Kingdom Need a Battlespace Management Language",04F- SIW-051, Fall SIW, 2004.
[12] Kleiner, M. S., Carey, S. A., and Beach, J., "Communicating Mission-Type Orders to Virtual Commanders", Proc. of Winter Simulation Conference, 1998.

[13] Carey, S., Kleiner, M., et al., "Standardizing Battle Management Language – Facilitating Coalition Interoperability",02E-SIW-005, European SIW, 2002.
[14] Tolk, A. and Pullen, M., "Ideas for a Common Framework for Military M&S and C3I Systems", 03E-SIW-032, Euro SIW, 2003.
[15] Sprinkle, R. B., Heystek, D. and Lovelady, S. D., "Common Scenario Generation for Army M&S and C4ISR Systems", 03S-SIW-103, Spring SIW, 2003.
[16] Hieb, M. R., Sudnikovich, W., Tolk, A., and Pullen, J. M., "Developing Battle Management Language into a Web Service",04S-SIW-113, Spring SIW, 2004.
[17] Khimeche, L., and de Champs, P., "M&S in decision support for Courses of Action Analysis, APLET",04F-SIW-006, Fall SIW, 2004.
[18] Mayk, I. and Klose, D., "Experimenting with C2 Applications and Federated Infrastructures for Integrated Full-Spectrum Operational Environments in Support of Collaborative Planning and Interoperable Execution", Proc. of C2 Res. and Tech. Symposium, 2004.
[19] Sudnikovich, W., Hieb, M. R., et al., "Developing the Army's Battle Management Language Prototype Environment",04S-SIW-115, Spring SIW, 2004.
[20] Tolk, A., et al., "Coalition Battle Management Language", Fall SIW, 2004.
[21] Hieb, M.R., Sudnikovich, W.P., et al., "Developing Extensible BML to Enable Coalition Interoperability", 04E-SIW-064, European SIW, 2004.
[22] Turnitsa, C., Kovurri, S., et al., W., "Lessons Learned from C2IEDM Mappings Within XBML",04F-SIW-111, Fall SIW, 2004.
[23] Tolk, A. et al., "Merging National Battle Management Language Initiatives for NATO Projects", Proc. of RTA/MSG Conf. "M&S to address NATO's new and existing Military Requirements", RTO-MP-123, Germany, Oct. 2004.
[24] DeMasi, L., Dobbs, et al., "Implementing Battle Management Language: A Case Study Using the Command and Control Information Exchange Data Model and C4I-M&S Reference Object Model",05S-SIW-068, Spring SIW, 2005.
[25] Khimeche, L., and de Champs, P., "APLET's Courses of Action Modeling: A Contribution to CBML",05S-SIW-018, Spring SIW, 2005.
[26] Perme, D., Tolk, A., et al., "Integrating Air and Ground Operations within a Common Battle Management Language",05S-SIW-154, Spring SIW, 2005.
[27] Roberts, J. D. et al., "Achieving Higher Levels of Interoperability Between M&S and C2 systems Through Application of BML to the SINCE Program", Spring SIW, 2005.
[28] Tolk, A., and Blais, C., "Taxonomies, Ontologies, and Battle Management Languages - Recommendations for the Coalition BML Study Group", Spring SIW, 2005.
[29] Tolk, A., et al., "Web Services based on the C2IEDM – Data Mediation and Data Storage",05S-SIW-019, Spring SIW, 2005.
[30] Winters, L., and Tolk, A., "The Integration of Modeling and Simulation with Joint Command and Control on the Global Information Grid", Spring SIW, 2005.
[31] Hieb, M. R., Pullen, et al., "A Geospatial Battle Management Language (geoBML) for Terrain Reasoning", Proc. of 11th C2 Res. and Tech. Sym., 2006.
[32] Schade, U., Hieb, M. R., "Development of Formal Grammars to Support Coalition Command and Control: A Battle Management Language for Orders, Requests and Reports", Proc. of 11th C2 Res. and Tech. Symp., 2006.

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

DÖNGÜDE DONANIMSAL BENZETİM TEKNİĞİ VE BİLEŞENLERİ

Erkan DUMAN [(a)], Hayrettin CAN [(a)], Erhan AKIN [(a)]

[(a)] Fırat Üniversitesi,Bilgisayar Mühendisliği 23119-ELAZIĞ,

{erkanduman, hcan, eakin}@firat.edu.tr

ÖZ

Bu çalışmada gömülü kontrol sistemlerinin test aşamasında kullanılan Döngüde Donanımsal Benzetim(DDB) tekniği ve bu tekniğin bileşenleri incelenmiştir. DDB tekniği, gerçek gömülü kontrol sistemlerinin, gerçek hayatta çalışacakları sistemler yerine benzetilmiş sistemler üzerinde gerçek zamanlı olarak test edilmesi için geliştirilmiş bir yaklaşımdır. Bu yaklaşımın kendisi ve içerdiği alt bileşenleri maddeler halinde detaylı olarak ele alınmaktadır. Ayrıca bu yaklaşımın performansını etkileyen faktörler ve parametreler vurgulanmaktadır.

Anahtar Kelimeler: Döngüde Donanımsal Benzetim, Gerçek Zamanlı Benzetim, Gömülü Kontrol Sistemlerinin Geliştirme ve Test Süreçleri.

HARDWARE-IN-THE-LOOP SIMULATION TECHNIQUE AND ARCHITECTURE

ABSTRACT

In this paper, we studied a real-time simulation technique called hardware-in-the-loop(HIL) simulation and its architecture that it is used in the cycle of an embedded control system development. This technique allows to test the real embedded control systems with simulation environment in real-time instead of real-world systems. We used the basic examples which had been published in the related studies in literature in order to explain this subject. In addition, we pointed out some factors and parameters which influence the performance of this approach.

Keywords: hardware-in-the-loop simulation, real-time simulation, development and test of embedded control systems.

1. GİRİŞ

Gömülü kontrol sistemleri kritik ve tehlikeli uygulamalarda kullanıldıkları zaman çok titiz ve metodolojik bir gerçekleştirmeye ihtiyaç duymaktadırlar çünkü bu tarz kontrol sistemlerinde oluşabilecek hataların piyasaya sunulduktan sonra düzeltilmesi çok zordur. Ayrıca bu sistemlerde oluşabilecek hatalar insan hayatı da dahil olmak üzere telafisi mümkün olmayan sonuçlara neden olabilmektedirler. Dolayısıyla bu ürünlerin geliştirme sürecinde test aşaması büyük bir önem kazanmaktadır ve itinalı bir mühendislik çalışması gerektirmektedir. Ürün geliştirme sürecinde test aşamasının olması, tasarımın daha pahalı, daha uzun süreli ve daha kompleks hale gelmesine neden olmaktadır ancak bu aşamanın hayati önem taşıması kaçınılmaz bir adım olmasına neden olmuştur. Gömülü kontrol sistemlerinin çoğunun kritik şartlar altında çalışmasından dolayı bir çok araştırmacı, bu işin tasarım, geliştirme, gerçekleştirme ve test aşamaları üzerine çalışmalar yapmaktadırlar[1,2].

DDB tekniği gömülü kontrol sistemlerinin geliştirilmesinde ve test edilmesinde kullanılan güncel ve yaygınlığı gittikçe artan gerçek zamanlı bir benzetim tekniğidir. DDB tekniği gerçek gömülü kontrol sistemlerinin gerçek hayatta çalışacakları sistemler yerine benzetilmiş sistemler üzerinde çoğunlukla gerçek zamanlı olarak test edilmesi için geliştirilmiş bir yaklaşımdır. Bu sayede DDB ile bir gömülü kontrol sisteminin değişik koşullarda ve çalışma şartları altında test edilmesi mümkündür[3]. M.A.Sanvido'nun "Döngüde Donanımsal Benzetim Çatısı" başlıklı tezi DDB tekniğinin literatürdeki ilk sunumlarından biridir[4]. M.A.Sanvido, prototip bir helikopterin ve prototip bir hidroelektrik santralinin kontrolünde kullanılmak istenen gömülü kontrol sistemlerini, DDB tekniği ile geliştirmiş ve bu geliştirdiği denetleyicileri gerçek prototipler üzerinde test ettiğinde benzetim ortamındaki sonuçlar ile örtüşen başarılı sonuçlar elde etmiştir.

Diğer benzetim teknikleri, gerçek gömülü kontrol sistemlerinin bir bütün olarak test edilmesine çoğunlukla uygun değildirler sadece denetleyici algoritmasının veya diğer bileşenlerin bağımsız olarak test edilebilmesine izin vermektedirler. Gömülü kontrol sistemlerinde denetleyici algoritması bütün sistemin %20-30'unu oluşturmaktadır. Ayrıca diğer benzetim yöntemlerinde bir operatör tarafından elle üretilen hata ve test koşulları DDB tekniğinde yer alan hata üreteci bileşeni ile otomatik olarak üretilebilmektedir.

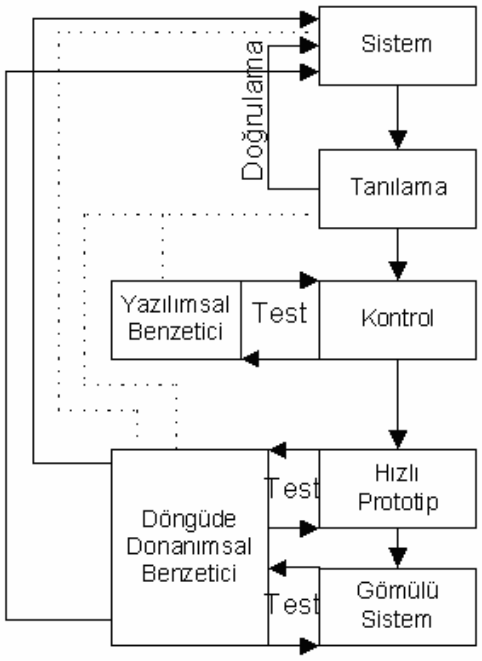
DDB tekniği bu makalede sırasıyla şu şekilde ele alınmıştır: Öncelikle gömülü kontrol sistemleri tanımlanmış ve tasarım süreçleri incelenmiştir. Daha sonra DDB tekniğinin ne olduğu ve bileşenlerinin neler olduğu anlatılmıştır. Örnek bir uygulama ve uygulamaya ait giriş/çıkış değişkenleri gösterilmiştir. Sonuç olarak ise bir DDB örneğinde hangi parametreler dikkat edilmesi gerektiği vurgulanmıştır. Ayrıca günümüzde DDB örneklerinin kazandığı yeni özellikler ve geliştirmeler üzerinde durulmuştur.

2. GÖMÜLÜ KONTROL SİSTEMLERİ

Gömülü kontrol sistemleri, otomobil, ev ve ofis araçları, uzay aracı gibi insansız çalışan bir makine vb. bir çok sistemin kontrolünde kullanılan özel amaçlı bilgisayarlardır. Dijital işlemci teknolojisindeki hızlı gelişmelerden dolayı, gömülü kontrol sistemleri

günlük hayatımızdaki bir çok cihazı kontrol eder hale gelmiştir. Bu sistemler genellikle küçük boyutludurlar ve fonksiyonellikleri sabittir. İşletim sistemleri ve uygulama fonksiyonları aynı program içerisinde birleştirilmiştir. Gömülü kontrol sistemlerinin geliştirilmesi, klasik ürün ve yazılım geliştirme süreçlerine göre çok daha fazla güvenlik ve sağlamlık gerektirmektedir. Çünkü bu sistemlerin büyük bir çoğunluğu hataların giderilme şansının olmadığı kritik ve tehlikeli uygulamalarda kullanılmaktadırlar[5].

Şekil 1.'de bir gömülü kontrol sisteminin tasarım süreci bir akış diyagramı şeklinde gösterilmektedir. Öncelikle sistemin kontrol edilebilmesi için gerekli olan eyleyicilerin ve algılayıcıların belirlenmesi gerekmektedir. Eyleyiciler, denetleyicinin kontrol edeceği sisteme uygulayacağı girişi değişkeni için kullanılan arayüz elemanlarıdır. Örneğin gerilim ile kontrol edilecek elektriksel bir makinede denetleyicinin makineye istediği aralıkta gerilim vermesini sağlayan mekanizmalardır. Benzer şekilde sistemin durumunu makineye bildirmek için kullanılan mekanizmalar ise algılayıcılardır.



Şekil 1: Gömülü Kontrol Sistemi Tasarımı

İkinci aşama sistemin dinamikliğini anlamaktır yani sistemin matematiksel veya fiziksel tanımını yapmak gerekmektedir(Sistem Tanılama Aşaması). Bu aşamada türetilen model ile gerçek sistem karşılaştırılarak doğrulama yapılır. Bu doğrulama işleminde belli bir hata toleransı elde edilinceye kadar model üzerinde iyileştirmeler yapılmaktadır. Üçüncü aşama ise denetleyicinin tasarlanmasıdır. Öncelikle önerilen denetleyicinin istenilen performansı sağlayıp sağlayamayacağı, yazılımsal benzeticiler kullanılarak test edilir. Bu aşamada uygulanan benzetim tekniği tamamen sanal bir benzetim tekniğidir.

Dördüncü aşama, denetleyicinin gerçek platformda test edilmesidir. Bu aşamada sanal bir kontrol sistemi, gerçek veya döngüde donanımsal bir benzetici ile test edilir. Bu aşamada, hızlı prototip benzetim tekniği kullanılmaktadır. Eğer prototipin gerçek sistem üzerinde test edilmesi maliyetli veya tehlikeli ise döngüde donanımsal benzetici kullanılabilir. Sanal benzetimden bu aşamaya geçiş yapılırken genellikle otomatik kod üretme kullanılır[6].Örneğin MATLAB/Simulink kontrol tasarım paketi ile sanal denetleyicinin kodu otomatik üretilebilmektedir.

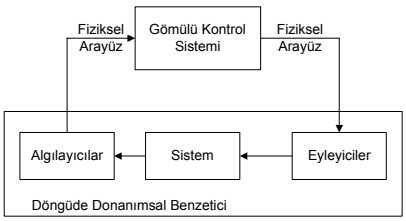
Son aşama ise sahada çalışacak olan gerçek gömülü kontrol sisteminin üretilmesi ve test edilmesidir. Sanal denetleyici kodundan gerçek denetleyici koduna çevrim oldukça zordur. Ayrıca gömülü kontrol sistemindeki yazılım, tek başına denetleyici algoritmasından çok daha zor ve karmaşıktır. Çünkü gömülü kontrol sistemleri, denetleyici algoritmasından başka görevleri de gerçek zamanda yerine getirmektedirler. Denetleyici algoritması, gömülü kontrol sistemindeki yazılımın sadece %20-30 gibi bir kısmını oluşturmaktadır. DDB tekniği ile geliştirilen benzetici bu safhaya gelmiş olan gömülü kontrol sisteminin tehlikesiz ve daha az maliyetle test edilmesine olanak kılmaktadır.

3. Döngüde Donanımsal Benzetici

DDB tekniği Isermann[7] şu şekilde tanımlamaktadır :

- Benzetilmiş sistemler, gerçek donanımsal denetleyiciler ile birlikte çalışabilirler.
- Kontrol edilecek gerçek sistemler, kısmen veya tamamen benzetilmiş sistemler ile yer değiştirebilirler (Algılayıcılar, Fiziksel Süreçler ve Eyleyiciler).

Yukarıdaki iki tanımdan da anlaşılacağı üzere DDB, gerçek donanımsal denetleyicilerin benzetilmiş sistemler üzerinde test edilmesi tekniğidir. Bu benzetim ortamı, çoğunlukla gerçekleştirmesi masraflı veya tehlikeli uygulamalarda kullanılmaktadır[8]. Bir tarafta gerçek donanımsal denetleyici varken diğer tarafta benzetilmiş bir sistem bu denetleyici ile uyumlu bir şekilde çalışmaktadır. Benzetilmiş sistem piyasada var olan özel amaçlı benzeticiler olabileceği gibi kişisel bir bilgisayarda koşan bir benzetim programından da ibaret olabilmektedir(MATLAB/Simulink vb.)[9].



Şekil 2. Bir Döngüde Donanımsal Benzeticinin Blok Diyagramı.

Şekil 2'de görüldüğü gibi gömülü kontrol sisteminin çıkışı, benzeticinin giriş değişkenleridir ve benzeticinin hesapladığı çıkış ise kontrol sisteminin giriş değişkenleridir. Bir döngüde donanımsal benzeticide giriş/çıkış değerleri aynı zaman

aralığında olmalıdır. Benzetici o andaki giriş değişkenlerini alıp sistem denklemlerini hesapladıktan sonra belirlenen zaman aralığı sona ermeden sistemin davranışını belirten çıkışı üretmelidir. Gömülü kontrol sisteminin çıkışı sayısal değilse veya direk bilgisayar tarafından işlenemiyor ise fiziksel bir ara yüz gerekebilir. Sistemin en başta hesaplanan bir matematiksel modeli bulunmaktadır. Bu model üzerinde eyleyicilerin çıkış değerlerine göre hesaplama yapılır ve benzetilmiş algılayıcılara iletilir. Benzetilmiş algılayıcılar, gömülü kontrol sisteminin girişini beslemek üzere çıkış üretirler. Eğer algılayıcıların çıkışı sayısal değil ise veya denetleyici tarafından direk işlenemiyorsa bu iletim sırasında da fiziksel bir ara yüz gerekir.

Ancak dikkat edilmesi gereken önemli bir nokta: benzeticinin yukarıda anlatılan akış diyagramındaki işlemleri gerçek zamanda, denetleyici ile senkronize bir şekilde yapması gerektiğidir. Denetleyicinin ilk başta verilen bir örnekleme zamanı vardır ve benzetici bu zaman aralığı içerisinde girişi almalıdır, sistemin davranışını hesaplamalıdır ve çıkışı üretmelidir. Dolayısıyla böyle bir benzeticinin geliştirilmesinde hız için her türlü fedakarlığa gitmekten kaçınılmamalıdır. Ancak böyle bir benzetici gerçekleştirildikten sonra, gömülü kontrol sistemi gerçek zamanda, değişik yük ve çalışma koşulları altında test edilebilmektedir.

Bir döngüde donanımsal benzetim tekniğinde istenilen esnekliği yakalamak ve etkili sonuçlar alabilmek için aşağıdaki hususlara dikkat edilmesi gerekir.

- Benzeticinin giriş ve çıkış sinyallerinin üretilmesi.
- Benzeticinin gerçek zamanda çalıştırılabilmesi.
- Benzetilmiş sistemin olabildiğince basit matematiksel formüllere indirgenebilmesi ve gerçek zamanda bu formüllerin hesaplanabilmesi. Ancak bu indirgemeler sistemin gerçekçiliğini yitirmediğini garanti etmelidir.
- Test sürecinin verimli olabilmesi için sistemin kendi kendine hata üretebilmesi ve buna karşılık denetleyicinin çıkışının gözlemlenebilmesi.

4. DDB TEKNİĞİNİN TEMEL BİLEŞENLERİ

Bu noktada, “Bütün DDB uygulamaları için ortak olan minimum gereksinim nedir?” sorusuna cevap aramak gerekir. Aslında bu sorunun cevabı iskeletin temel bileşenleridir. Bunun için birçok farklı uygulama analiz edilebilir. Elde edilecek minimum kümesi: {Numerik Benzetim, Donanımsal Arayüz ve Gerçek Zamanda Planlama} olacaktır. Ayrıca bu kümeye “Hata Üretme” gibi çok kullanışlı bir özellik de eklenebilir. Bu makalede benzetim ile uğraşan herkesin aşina olduğu numerik benzetim teknikleri üzerinde durulmayacaktır.

4.1. Gerçek Zamanda Planlama

Gerçek zamanlı sistemler ve gerçek zamanlı işletim sistemleri, çoğunlukla büyük bir fonksiyonel özellik kümesine sahiptirler ve bunları gerçekleştirmek oldukça komplekstir[9]. Bu fonksiyonel özellikler, bir mikrodalga fırınından tutun da Mars’ta çalışacak bir bilgi toplama robotuna kadar bir çok farklı uygulamayı gerçekleştirecek

ölçekte geniş olmalıdır. Gerçek zamanlı bu fonksiyonlar; planlama, görev önceliği atama ve kaynak yönetimi şeklinde kategorize edilebilir.

Planlama, her bir görev ve program parçacığının çalışmasını kontrol eder. O an için en uygun olan görevleri seçer ve bunların sistemde bulunan işlemci veya işlemciler üzerinde koşmasını sağlar. Bir görevin önceliği statik veya dinamik olarak belirlenebilmektedir. Dinamik atamada göz önüne alınan kriterler sırası ile şunlardır: görevin çalışması için ihtiyaç duyulan en kötü zaman, görevin bitmesi gereken zaman aralığı, görevin başlangıç zamanı ve görevin çalışması için ihtiyaç duyduğu kaynaklar.

Gerçek zamanlı işletim sistemlerinde giriş/çıkış birimleri ve hafıza gibi ortak kullanılan kaynakların, farklı görevler tarafından erişimi sırasında çakışma olmayacağını garanti eden mekanizmalar mevcuttur. Çok yaygın olan bu mekanizmalar çok kompleks olabilmektedir ve bazen ölü-kilit problemine neden olabilmektedirler.

4.2. Donanımsal Arayüz

Döngüde donanımsal benzeticinin bir çok farklı harici eyleyici ve algılayıcılar ile haberleşebilmesi gerekmektedir. Bunu gerçekleştirebilmek için farklı yaklaşımlar kullanılabilir. Bunlar arasında dijital sinyal arayüzü için hız avantajından dolayı genellikle *Yeniden Programlanabilir Kapı Dizileri* (Field Programmable Gate Array) tercih edilir. Sinyal üretme ve sinyal toplama aşamasında farklı tiplerdeki arayüzlerin kolayca gerçekleştirilebilmesi ve yazılımsal sürücülerin kolayca yazılabilmesi sorunların çözülmüş olması lazımdır. Bu gereksinimleri sağlayabilmek için aşağıdaki teknikler kullanılmaktadır:

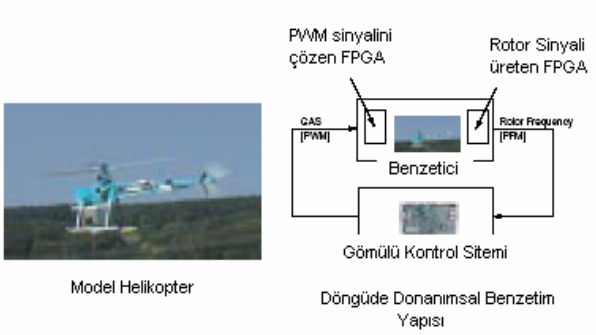
- Donanımsal Tanımlama Dili (HDL) yeniden programlanabilir donanımların ayarlanması
- FPGA kullanılarak sayısal arayüzlerin gerçekleştirimesi
- DAC / ADC kartları kullanılarak analog arayüzlerin gerçekleştirilmesi.

4.2. Hata Üretme ve Test Etme

DDB iskelet yapısının alt bileşeni olan hata üretme ve test etme bileşeni, isimlendirmeden de anlaşılacağı üzere sistemde hata üretmek, hataların ne zaman ortaya çıkacağını planlamak ve gömülü kontrol sisteminin oluşan hatalara karşı verdiği cevapların doğruluğunu kontrol etmek için kullanılır. Denetleyici sistemin sırası ile test edilen durum sayısı arttıkça sağlamlığı da artacaktır. Bir önceki test koşulunda öğrenilen bilgilere, bir sonraki test koşulunda yenileri eklenecektir.

5. DDB ÖRNEĞİ

Şekil 3'de görüldüğü gibi M.Sanvido prototip bir helikopterin kontrolü için kullanılacak olan ve kısaca OLGA diye isimlendirilen gömülü kontrol sistemini, döngüde donanımsal benzetim tekniği ile geliştirmiş ve gerçek uçuş verileri ile örtüşen başarılı sonuçlar elde etmiştir[10].



Şekil 3 : Döngüde Donanımsal Benzetim Örneği.

Yukarıdaki şekilde görüldüğü gibi bir tarafta donanımsal bir kontrol sistemi var iken diğer tarafta bir benzetici bulunmaktadır. Benzeticinin girişleri, kontrol sisteminin motorların hızlarını belirlemek için ürettiği PWM sinyalleridir ki; bunların hızlı bir şekilde çözülebilmesi için PWM-encoder lar FPGA teknolojisi ile gerçeklenmiştir. Benzetici bu giriş bilgileri ve bulunduğu durum bilgisini kullanarak yeni bir duruma geçer. Bunun için prototip helikopterin matematiksel modeli kullanılır. Yeni durumdaki motor bilgilerini kontrol sistemine aktarmak için benzer şekilde rotor frekans sinyali üreten FPGA mevcuttur.

6. SONUÇ

Bu çalışmada, öncelikle gömülü kontrol sistemlerinin geliştirilmesi için kullanılan mevcut yaklaşımlar ve araçlar özetlenmiştir. Daha sonra bu sistemlerin geliştirilmelerinin final aşamasında kullanılan DDB tekniği üzerinde durulmuştur. Bu iki önemli kavram açıklandıktan sonra DDB tekniğinin iskelet yapısı ve bu yapının bileşenleri bölümler halinde açıklanmıştır.

DDB kavramı, literatürde çok yeni olan bir kavram değildir ancak yeni eklentiler ve geliştirmeler yapılarak çok daha kolay ve kullanışlı hale getirilmektedir[11,12]. Örneğin otomatik test için FAUSEL diye isimlendirilen bir hata üretme dili kullanmak, bahse değer bir gelişimdir. Ayrıca MATLAB gibi bilgisayar destekli tasarım yazılımlarındaki yeni paketler ile bu tekniğin nasıl kolay uygulanabilir bir hale geldiği de ilgi çekicidir. Sadece blok diyagramları ve akış şemaları kullanılarak hazır kütüphaneler üzerinden döngüde donanımsal benzetim gerçekleştirilebilmektedir.

KAYNAKÇA

[1] J. Ledin, M. Diclens, J. Sharp (2003), "Single Modeling Environment For Constructing High-Fidelity Plant and Controller Models", American Institute of Aeronautics and Astronautics.

[2] Marco A. A. Sanvido, W. Schaufelberger (2001), "Design of A Framework for Hardware-in-the-loop Simulations and Its Application to A Model Helicopter", Automatic Control Laboratory, ETH Zurich.

[3] Marco A. A. Sanvido, V. Cechticky, W. Schaufelberger (2002), "Testing Embedded Control Systems Using Hardware-in-the-loop Simulation and Temporal Logic", Automatic Control Laboratory, Swiss Federal Institute of Technology, Zurich

[4] Marco A. A. Sanvido (2002), "Hardware-in-the-loop Simulation Framework", Diss. ETH No:14535, ETH Zurich

[5] E. Gill, B. Naasz, T. Ebinsma (2003), "First Results From A Hardware-in-the-loop Demonstration of Closed Loop Autonomous Formation Flying", 26th Annual AAS Guidance and Control Conference.

[6] "LabView FPGA in Hardware-in-the-loop Simulation Applications", National Instruments

[7] R. Isermann, J. Schaffnit, S. Sinsel (1999), "Hardware-in-the-loop Simulation for the Design and Testing of Engine-Control Systems, Control Engineering Practice, 7:643-653.

[8] J. Ledin (2002), "Simulation Takes off with Hardware", CMP LLC, NY.

[9] H. Hanselmann (1996), "Hardware-in-the-loop Simulation Testing and its Integration Into a CACSD Toolset", The IEEE International Symposium on CACSD, Michigan USA.

[10] Marco A. A. Sanvido (1999), "A Computer System for Model Helicopter Flight Control", Technical Memo Nr:3, Institute for Computer Systems, ETH Zurich.

[11] S. Abourida, C. Dufour, J. Belanger, V. Lapointe (2003), "Real-time PC based Simulator of Electric Systems and Drives", International Conference on Power Systems Transients, in New Orleans USA.

[12] S. Ayasun, S. Vallieu, R. Fischl, T. Chmielewski (2003), "Electric Machinery Diagnostic/Testing system and Power Hardware-in-the-loop Studies", SDEMPED 2003, Atlanta USA.

MilSOFT DDS ARAKATMANI ve DDS'in SAVAŞ YÖNETİM SİSTEMLERİNDE SİMÜLASYON AMAÇLI KULLANIMI

Hüseyin KUTLUCA[(a)], İzzet Emre ÇETİN[(b)], Ertan DENİZ[(c)] Barış BAL[(d)]

[(a)] MilSOFT Yazılım Teknolojileri A.Ş, Adres, ODTU Teknokent Ankara, hkutluca@milsoft.com.tr

[(b)] MilSOFT Yazılım Teknolojileri A.Ş, Adres, ODTU Teknokent Ankara, icetin@milsoft.com.tr

[(c)] MilSOFT Yazılım Teknolojileri A.Ş, Adres, ODTU Teknokent Ankara, edeniz@milsoft.com.tr

[(d)] MilSOFT Yazılım Teknolojileri A.Ş, Adres, ODTU Teknokent Ankara, bbal@milsoft.com.tr

ÖZ

DDS Arakatmanı temel olarak veri merkezli (data centric) ve yayımla-abone ol (publish-subscribe) mimarisine dayalıdır. MilSOFT, 2004 yılında başlattığı GEMKOMSİS ARGE projesi kapsamında DDS arakatmanını geliştirmiştir. MilSOFT, DDS arakatmanını Savaş Yönetim Sistemi yazılımının geliştirilmesinde gerçek zamanlı dağıtım arakatmanı ve simülasyon amaçlarına yönelik olarak kullanmaktadır. HLA ve DDS ortak kullanımının değişik uygulamalarda mümkün olabileceği öngörülmektedir. Bu ortak kullanım için her iki arakatmanın özellikleri de dikkate alınarak etkin bir tasarım yapılması sağlanabilir.

Anahtar Kelimeler: Arakatman, DDS, Dağıtık Sistemler, Gerçek Zamanlı Sistemler, OMG, Savaş Yönetim Sistemi, Servis Kalitesi, Yayımla-Abone Ol

MilSOFT DDS MIDDLEWARE and USING DDS FOR SIMULATION IN COMBAT MANAGEMENT SYSTEMS

ABSTRACT

DDS (Data Distribution Service) is a data-centric middleware based on publish-subscribe paradigm. MilSOFT has developed DDS middleware within GEMKOMSIS R&D project that was initiated in 2004. MilSOFT utilized DDS middleware in Combat Management System development as a real-time distribution middleware and for simulation purposes. It is foreseen to use DDS and HLA (High Level Architecture) together in various applications. For this common use, feasible system architectures can be driven by taking both middleware's features into consideration.

Keywords: Combat Management System, DDS, Distributed Systems, Middleware, OMG, Publish-Subscribe, Real-Time Systems, Quality of Service, HLA

1. GİRİŞ

Dağıtık ortamlarda verinin iletilmesi için, dağıtık ortamın ihtiyaç duyduğu kalite özelliklerini sağlayabilen arakatman yazılımları kullanılmaktadır. Simülasyon sistemlerinin entegrasyonu için de, IEEE tarafından standardı yayımlanmış olan High Level Architecture (HLA) ve onun arakatmanı olan RTI yaygın olarak kullanılmaktadır. Bunun yanında Object Management Group (OMG) tarafından gerçek zamanlı sistemler için hazırlanmış olan Data Distribution Service (DDS) arakatman yazılımı standardı da Haziran 2004'te yayımlanmıştır[4].

DDS Arakatmanı temel olarak veri merkezli (data centric) ve yayımla-abone ol (publish-subscribe) mimarisine dayalı bir arakatmandır. Gerçek zamanlı sistemlerin dağıtık ortamdaki ihtiyaçlarını karşılamak üzere zengin servis kalitesi özelliklerini (Quality of Service, QoS) içermektedir.

DDS arakatmanı dağıtık ortamda çalışan gerçek zamanlı sistemlerin ihtiyaçlarını karşılamak üzere geliştirilmiştir[6,7]. DDS arakatmanının kullanıldığı alanlar komuta kontrol sistemleri, endüstriyel otomasyon, dağıtık kontrol ve simülasyon, telekomünikasyon, sensör ağları ve ağ temelli sistemlerdir. Bu alanlardaki uygulamalar hızlı, etkin ve tahmin edilebilir bir veri iletişimine ihtiyaç duyarlar. Ayrıca, üretilen verinin birden çok uygulamaya iletilmesi ve bu özelliğin sistem performansını olumsuz etkilememesi ana beklentilerdir. Bellek, işlemci ve veri yolu gibi kısıtlı kaynakların ihtiyaçlara göre etkin kullanılabilmesi için arakatman parametrelerinin ayarlanabilmesi temel bir ihtiyaçtır. DDS arakatmanı buna yönelik olarak zengin servis kalitesi özellikleri sağlamakta ve bu servis kalitesi özelliklerinin her bir DDS nesnesi seviyesinde ayarlanabilmesine olanak vermektedir. Bu özellikleri ile DDS arakatmanı dağıtık ve gerçek zamanlı sistemler için önemli bir arakatman standardı olmuştur.

MilSOFT 2004 yılında başlattığı GEMKOMSİS ARGE projesi kapsamında DDS arakatmanını geliştirmiştir [3]. GEMKOMSIS projesi, su üstü platformları için modern bir Savaş Yönetim Sistemi (SYS) yazılımının, tümü ile milli ve özgün bir tasarımla, COTS ürünler de kullanılarak, ARGE yaklaşımı ile oluşturulmasına yöneliktir. Bu proje TÜBİTAK Teknoloji ve Yenilik Destek Programları Başkanlığı (TEYDEB) tarafından da desteklenmiş ve 2006 Aralık ayında tamamlanmıştır. GEMKOMSIS projesi kapsamında geliştirilen DDS arakatmanı, diğer komuta kontrol altyapı yazılımları, gemi ortak servisleri ve gemi ortak fonksiyonları 2007 yılında başlanan Sahil Güvenlik Komutanlığı Arama Kurtarma Gemileri için geliştirilecek SYS yazılımının temelini oluşturmuştur. Bunun yanı sıra MilSOFT DDS, şirket içindeki diğer müşterili projelerde de kullanım alanı bulmuştur. MilSOFT DDS arakatmanı ve onunla ilgili geliştirilmiş olan DDS Kod Üreteci ve DDS SPY uygulamaları da tamamlanmış bulunmaktadır.

DDS arakatmanı, mimarisi ve zengin servis kaliteleri ile geniş bir alanda kullanım fırsatları sunmaktadır. DDS arakatmanının simülasyon sistemleri için kullanımı da söz konusudur [1, 2, 9]. RTI arakatmanı ise simülatörlerin birbirine entegrasyonu için uygundur. Ancak, özellikle simülatörlerin, gerçek sistemlerle birlikte / Donanım Çevrimde (Hardware In The Loop) kullanıldığı durumda ortaya çıkan, yüksek sıklıkta ve zaman kritik verilerinin iletimi ihtiyacı için DDS'in HLA'ya göre daha uygun olduğu değerlendirilmektedir[9]. HLA, simülasyon sistemlerinin birlikte ve yeniden kullanılabilirliğini (interoperability & reusability) sağlamak maksadıyla geliştirilmiş

olup, DDS'te bulunan gerçek zamanlı veri iletimi, hataya dayanıklılık, veri filtreleme gibi servis kalitesi özelliklerini içermemektedir.

Savaş Yönetim Sistemi (SYS) gibi gerçek zamanlı sistemler için DDS kullanılırken, bu tür sistemlerin simülasyonu için HLA kullanımı uygun olmaktadır. Bununla birlikte, DDS arakatmanı bu sistemlerin testi ve eğitimi için geliştirilecek simülatörlerde arakatman olarak da kullanılabilmektedir. DDS'in SYS yazılımının testi için geliştirilen simülatörlerde kullanımı bu makalenin kapsamında verilmektedir

Bu makale kapsamında; Bölüm 2'de DDS mimarisi anlatılmaktadır. Bölüm 3'te MilSOFT DDS arakatman yazılımının geliştirilmesi süreci ve MilSOFT DDS arakatmanının DDS'in özellikleri sunulmuştur. DDS arakatmanının simülasyon için kullanım alanları Bölüm 4'te ele alınmıştır. Bölüm 5'te ise DDS arakatmanının SYS yazılımının testi için geliştirilen simülatörlerde kullanımı hususunda bilgi verilmektedir.. Bölüm 6'da varılan sonuçlar ve yapılan değerlendirmeler ifade edilmektedir.

2. DDS ARAKATMANI

OMG, arakatman teknolojileri ve nesneye dayalı yazılım geliştirme teknolojilerinin tanımlanması ve standartlaştırılmasını hedefleyen, kar amacı olmayan uluslararası bir kuruluştur. OMG daha önce UML ve CORBA standardını geliştirmiş ve kullanımının yaygınlaşmasını sağlamıştır.

OMG tarafından gerçek zamanlı sistemler için hazırlanmış DDS arakatman yazılımı standardı Haziran 2004'te yayımlanmıştır.

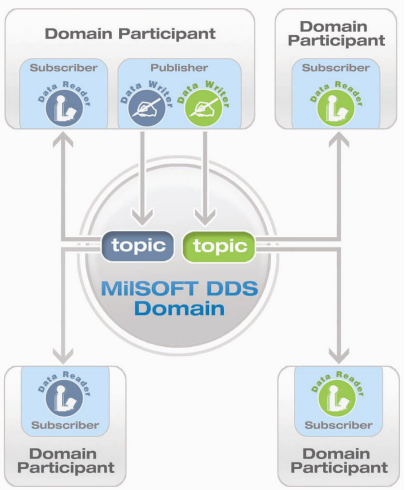
DDS, temel olarak veri merkezli (data centric) ve yayımla-abone ol (publish-subscribe) mimarisine dayalı bir arakatmandır. Sistem, içindeki veriyi üretenden (publisher) kullanana (subscriber) sağlamayı temel aldığından veri merkezlidir. Bunun alternatifi ise, CORBA'da olduğu gibi, farklı bir noktadaki bir servisten hizmet almayı sağlayan servis merkezli arakatmandır.

Yayımla-abone ol mimarisi, veriyi üreten ile veriyi kullananın birbirinden bağımsız olmasını sağlamaktadır. Bunun sonucunda, sisteme yeni modüller kolaylıkla entegre olabilmekte ve sistemde bir değişiklik yapmadan mevcut verilere abone olunup gerektiğinde yeni veriler üretilebilmektedir.

DDS arakatmanı, gerçek zamanlı sistemlerin gerçek zaman ve hataya dayanıklılık ihtiyaçlarını karşılamak üzere geniş bir servis kalitesi desteği sunmaktadır. Verinin güvenli olarak gönderileceğini belirleyen Güvenilirlik (reliability) servis kalitesi buna bir örnektir. Komutlar gibi teslimatı mutlaka gerçekleşmesi gereken veriler güvenli servis kalitesinde gönderilirken (reliable), periyodik olarak güncellenen verilerin bu servis kalitesinde gönderilmesine ihtiyaç yoktur (best effort). Kalıcılık (durability) servis kalitesi; verinin uçucu (volatile), bellekte kalıcı (transient) ya da diskte kalıcı (persistent) olacağını belirler. Bir verinin uçucu olması sadece o anda dinleyen modüllere ulaştırılacağını, bellekte kalıcı ise sonradan sisteme dahil olan modüllere de iletileceğini ve diskte kalıcı ise sistem kapanıp açıldığında bile erişilebileceğini belirler. Bunların dışında sunum sırası, teslim tarihi, sahiplik, yaşam durumu, zaman filtresi

iletişim önceliği, öncesi bilgisi, kaynak limiti gibi servis kalitesi özellikleri de mevcuttur.

DDS arakatmanı, ilklendirme ve iletişim için nesneye dayalı bir arayüz sunar. Bu arayüzdeki nesnelerden temel olarak "DomainParticipant", "Type", "Topic", "Publisher", "Subscriber", "DataWriter" ve "DataReader" örnek olarak verilebilir. Bu nesnelerin birbiri ile ilişkisi Şekil 1'de verilmiştir.



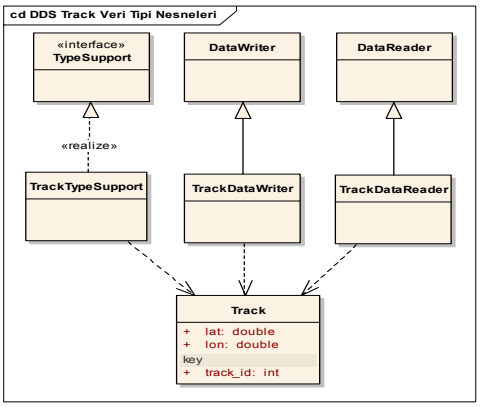
Şekil 1. DDS arakatmanı nesneleri

"Domain" birbiri ile iletişimde bulunan uygulamaları bağlayan dağıtık bir yapıdır. Dolayısı ile aynı "domain"de bulunan uygulamalar birbirleri ile veri alışverişinde bulunabilirler. "**DomainParticipant"** nesnesi uygulamaların bir domain'e lokal üyeliğini temsil eder [4]. İletişimde kullanılacak veriler "**Type"** ile tanımlanır. "Type" ile verinin içeriği ve anahtar alanlar belirlenir. "**Topic"** nesnesi bir "Type" üzerinden iletişim kurmak için tanımlı isim ve servis kalitelerinden oluşur. Uygulamalar "Topic"lere abone olabilirler veya "Topic" yayımlayabilirler.

Veri yayımlamak için öncelikle "**DomainParticipant"** nesnesi üzerinden "**Publisher"** nesnesi oluşturulur. "Publisher" nesnesi bütün yayınlar için bir arayüz oluşturur ve yayın için gerekli "**DataWriter"** nesnesini yaratmaktan sorumludur. Bir "**Topic"** için veri yayımlamak üzere bir "**DataWriter"** yaratılır. Bu "DataWriter" nesnesi aracılığı ile veriler arakatmana yazılır. Benzer şekilde verilere abone olmak için "**Subscriber"** nesnesi yaratılır. Bir "Topic"e abone olmak için "**Subscriber"** nesnesi üzerinden bir "**DataReader"** yaratılır. Uygulamalar, "DataReader"lar üzerinden arakatman aracılığı ile dağıtılan ve kendi abone oldukları "Topic"lere ait verilere erişebilirler.

DDS arakatmanı veri tipine dayalı bir programlama arayüzü sağlamak amacı ile o tipte iletişimi sağlayan özelleşmiş "DataWriter" ve "DataReader" nesnelerini otomatik üretmeyi ve kullanmayı öngörmüştür. Tanımlanan veri tipi üzerinden kod üreteci

kullanılarak veri tipleri için gerekli olan özelleşmiş "TypeSupport", "DataWriter" ve "DataReader" nesneleri yaratılır (Şekil 2).



Şekil 2. DDS Kod Üretecinin Track Veri Tipi için Ürettiği Nesneler

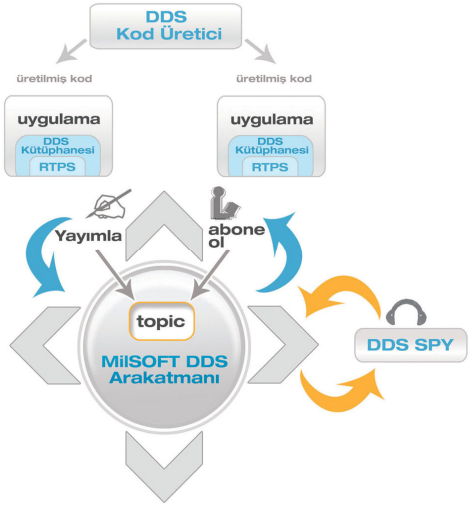
3. MilSOFT DDS ARAKATMANI

MilSOFT, Savaş Yönetim Sistemi ve onun altyapısını geliştirmek üzere başlattığı GEMKOMSİS ARGE projesi kapsamında MilSOFT DDS ürününü geliştirmiştir. Geliştirme çalışmaları 2004 yılı başında DDS standardının taslak sürümü üzerinden başlamış ve bu sayede standardın resmi olarak 1.0 versiyonu olarak yayımlandığı Haziran 2004 tarihinden çok daha önce MilSOFT DDS arakatmanının tasarımı oluşturulabilmiştir.

Geliştirilen MilSOFT DDS arakatmanı GEMKOMSİS Savaş Yönetim Sistemi uygulamalarının veri alışverişini sağlamak üzere entegre edilmiştir. Bunu takip eden süreçte arakatman yazılımı geliştirilmesi çalışmaları devam etmiş, yayımlanan yeni sürümler aracılığı ile uygulama yazılımları sürekli olarak entegre ve test edilmiş; tespit edilen sorunların giderilmesi ve mevcut yeteneklerin iyileştirmesine yönelik çalışmalar yapılmıştır. Bu çalışmalara paralel olarak DDS'te yapılması gerekli iyileştirmeleri planlayabilmek için, verinin iletim süresi ve toplam kapasite gibi performans parametreleri de devamlı olarak izlenmiştir. 2007 yılı ocak ayı itibarı ile DDS standardının 1.2 versiyonuna uyumlu MilSOFT DDS arakatmanı geliştirme faaliyeti tamamlanmıştır. Geliştirilen DDS arakatmanı OMG standardına göre "Minimum Profile" seviyesini sağlamaktadır [4]. Buna ek olarak Savaş Yönetim Sistemi için gerekli olan ve durum (status) verisini kapanıp/çöken ve yeniden başlayan uygulamalara ileten "Durability Profile" ve alınacak verinin sorgu ile filtrelenmesini sağlayan "Content Subscription Profile"özelliği de MilSOFT DDS'e kazandırılmıştır.

MilSOFT DDS arakatmanı gerçek zamanlı sistemlerin dağıtık veri uygulamalarına yönelik ihtiyaçlarını karşılamak üzere C++ programlama dili ile geliştirilmiş olup altyapısında başka bir dağıtım arakatmanını kullanmamaktadır. Ayrıca, gerçek zamanlı sistemlerde gerekli olan uygulamaların sistem başlama zamanından sonra hiç bellek

almaması (no dynamic memory allocation after initialization) özelliğini de barındırmaktadır.



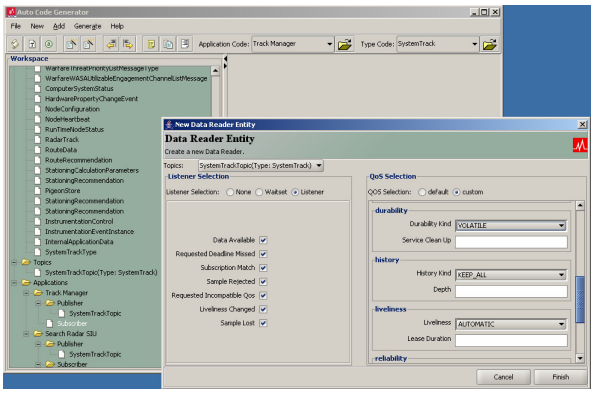
Şekil 3. MilSOFT DDS Arakatmanı

MilSOFT, DDS'in diğer DDS arakatmanları ile protokol seviyesinde uyumlu olması ve olgun bir iletişim protokolüne sahip olması için halihazırda taslak versiyonu yayınlanmış olan RTPS (Real-Time Publish Subscribe) protokolünü de geliştirmiş ve entegre etmiştir[5]. RTPS temelinde UDP/IP üzerinde çalışan bir protokoldür, ancak tasarımı veri iletimi için başka protokolleri de entegre etmeyi mümkün kılmaktadır. RTPS protokolü ile ileride MilSOFT DDS arakatmanına yeni protokoller kolayca entegre edilebilecektir. Ayrıca, RTPS uyumlu diğer DDS ürünleri ile veri alış verişi mümkün olacaktır.

MilSOFT, bir taraftan DDS arakatmanını geliştirirken bir yandan da DDS'in dağıtık bir yapıda etkin ve hatasız kullanımını sağlayacak yardımcı araçları da geliştirmiştir. Bu araçlar sırasıyla DDS Otomatik Kod Üreteci ve DDS Spy'dır.

3.1. DDS Otomatik Kod Üreteci (DDS Code Generator – DDS CG)

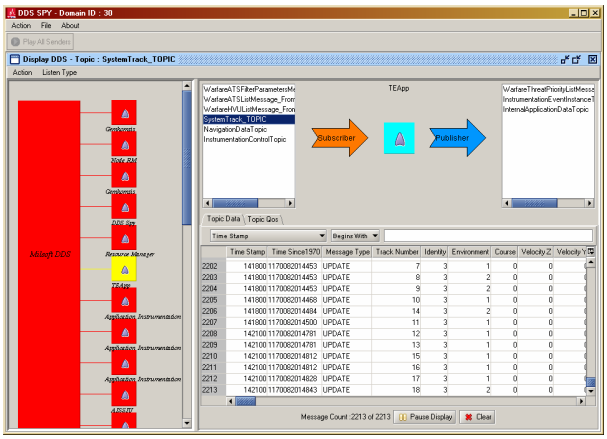
DDS otomatik kod üreteci, uygulama yazılımlarının DDS ile arayüzü için gerekli kodu üretir. Öncelikle veri tipine dayalı bir programlama arayüzü sağlamak için o tipte iletişimi sağlayan özelleşmiş DataWriter ve DataReader nesnelerinin otomatik olarak üretmektedir. Böylece, uygulama yazılımcılarının arakatman ayrıntılarıyla uğraşma zorluğunu ortadan kaldırmaktadır . Bu yaklaşım "Model Driven Development with Generative Programming" yaklaşımına uygundur. DDS CG, sistem içindeki arayüzlerin tek bir araç tarafından geliştirilmesini ve yönetilmesini sağladığından, modüller arası entegrasyonu kolaylaştırmakta ve sistem arayüzlerinin tek bir noktadan kontrolü imkanını vermektedir.



Şekil 4. MilSOFT DDS Otomatik Kod Üreteci

3.2. DDS-SPY Aracı

Diğer bir yardımcı araç olan DDS-Spy ise sistemde bulunan modülleri ve onların yayımladığı ve/veya abone olunan verileri izlemek için kullanılmaktadır. Ayrıca, test amaçlı olarak ihtiyaç duyulan verilerin kopyala-yapıştır yöntemi ile tekrar sisteme verilebilmesini sağlamaktadır. Bu yönleri ile DDS-SPY, uygulama yazılımcılarına, test mühendislerine ve sistem bakım grubuna büyük kolaylıklar sağlamaktadır.



Şekil 5. MilSOFT DDS- Spy aracı

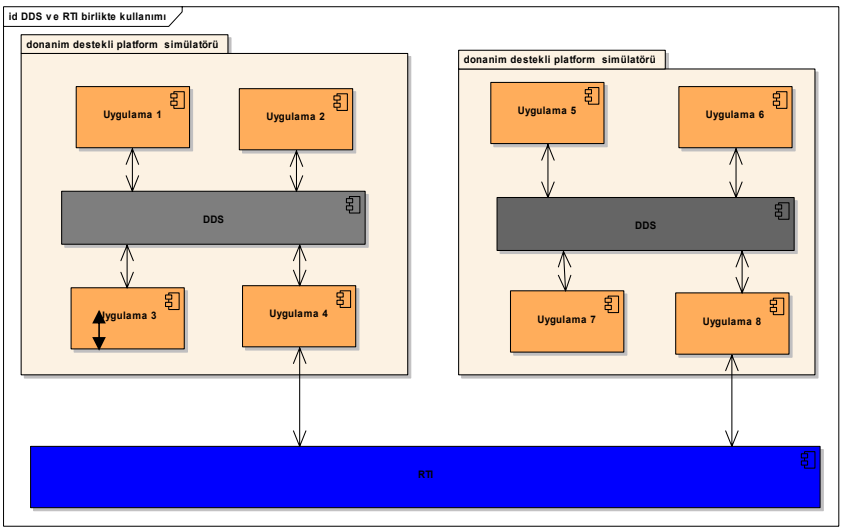
4. DDS'İN SİMULASYON UYGULAMALARINDAKİ YERİ

HLA ve DDS veri merkezli ve "yayımla/abone ol" esasına dayalı benzer standartlardır[1, 2]. Ayrıca, kullanıcı tarafından tanımlı veri yapıları kullanmaları ve içeriğe göre filtreleme özellikleri gibi benzerlikler taşımaktadırlar. HLA standardı zaman yönetimi, hiyerarşik nesne yönetimi, aynı nesnenin faklı alanlarının farklı uygulamalar tarafından güncellenebilmesi gibi simülasyon dünyası için geliştirilmiş özellikler barındırmaktadır. Öte yandan DDS ise zaman filtresi, sahiplik, çalışırlık durumu (liveliness) gibi zengin servis kalitesi özellikleri ve gerçek zamanlı veri

iletimine imkan veren veri modeli ile yüksek performans gerektiren simülatörler için uygun bir arakatman çözümü olmaktadır.

RTI ve DDS arakatmanlarının sistem gereksinimlerine göre birlikte kullanımı mümkündür [1,8]. Örneğin, donanım destekli platform simülasyonu için DDS arakatmanı kullanımı gerçek zaman ihtiyaçları ve DDS servis kaliteleri sebebi ile uygundur. Bu sistemlerde periyodik veri yoğun olarak yayımlanmakta ve birden fazla sistem (Hareket kontrol, görüntü üretme, gösterge simülasyonu vs.) tarafından kullanılmaktadır. Bu veri iletişiminin yayımla-abone ol mimarisi ile etkin bir şekilde gerçekleştirilmesi mümkündür. Ayrıca, yüksek frekansta yayımlanan bir veriyi daha az güncelleme yapan bir sistemin zaman filtresi servis kalitesi sayesinde istediği frekansta alması sağlanabilir. DDS arakatmanı bu sistemin içinde kullanılıyorken bu sistemlerin birleştirilerek HLA uyumlu ortak bir muharebe simülasyonunda RTI üzerinden bağlanması mümkündür.

Benzer şekilde bir Savaş Yönetim Sistemi DDS üzerinden iletişim sağlarken bu sistemlerin HLA uyumlu simülasyon sistemlerine bağlanması amacı ile RTI kullanılabileceği değerlendirilmektedir. Bu tür sistemlerde HLA ile DDS arakatmanları arasında köprü kurularak veri iletişiminin kolaylıkla yapılabileceği öngörülmektedir.



Şekil 6. RTI ve DDS Arakatmanlarının Birlikte Kullanımı

DDS arakatmanının RTI yerine kullanılması da bir uygulama alternatifi olabilir. Bu konuda detaylı araştırma yapılması ve uygulamalar ile denenmesine gerek duyulduğu değerlendirilmektedir.

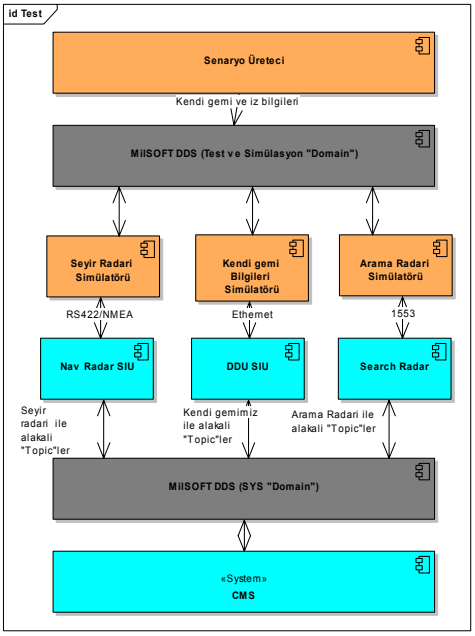
5. DDS'in SAVAŞ YÖNETİM SİSTEMLERİNDE SİMÜLASYON AMAÇLI KULLANIMI

DDS arakatmanı geliştirilmesi ve mevcut bir SYS içerisinde kullanımı GEMKOMSİS projesi kapsamında gerçekleştirilmiştir. MilSOFT DDS arakatmanı, diğer komuta kontrol altyapı yazılımları, gemi ortak servisleri ve gemi ortak fonksiyonları 2007

başında başlanan Sahil Güvenlik Komutanlığı Arama Kurtarma Gemileri için geliştirilecek SYS yazılımının temelini oluşturmaktadır.

Yeni sistemler geliştirilirken eğitim amaçlı simülasyon özellikleri de öngörülmektedir. Bu amaçla yapılan simülasyon için DDS arakatmanını öngörmekteyiz. Böyle bir sistem için gerçek zamanlı sistem için bir domain kullanılır iken simülasyon için verilerin ayrı bir domain üzerinden iletilmesi mümkündür.

SARSHIP kapsamında geliştirilen SYS yazılımını test etmek ve sistemi eğitim modunda kullanabilmek üzere arayüz protokolü seviyesi simülatörler, senaryo editörü ve oynatıcısı geliştirilmektedir (Şekil 7). Bu kapsamda senaryo oynatıcısından gelen veriler alt sistem simülatörleri tarafından işlenmektedir. Alt sistemlerin arayüz protokolü seviyesi simülasyonu ile, alt sistemin mesaj alışverişi simüle edilmektedir. Oluşturulan bu simülasyon konfigürasyonunda veri iletişimi için DDS tercih edilmiştir. Bunun temel gerekçesi mevcut proje kapsamında geliştirilmiş DDS arakatmanının bu uygulama için uygun bir altyapı sağlamasıdır. DDS arakatmanın veri merkezli yayımla/abone ol mimarisinin simülasyon yapısına uyumlu olması bu teknik çözümün seçilmesinde önemli bir etken olmuştur.



Şekil 7. DDS Arakatmanının Simülasyon-SYS Entegrasyonunda kullanımı

Örneğin radar simülasyonu için, Senaryo editörü ve oynatıcısında üretilip DDS arakatmanı üzerinden yayımlanan nesnelere ait bilgiler NavRadarSIM tarafından seyir radarı arayüz simülasyonu yapılarak RS422 üzerinden NMEA-0183 formatında NavRadarSIU bilgisayarına aktarılır. NavRadarSIU ise aldığı veriyi Savaş Yönetim Sistemi veri modeli yapısına uygun formata çevirerek iz seviyesinde DDS arakatmanı aracılığı ile sisteme sağlar.

6. SONUÇ

DDS ve HLA, veri merkezli ve "Yayımla-Abone ol" esasına dayalı standartlardır. Gerçek zamanlı sistemler için tasarlanmış olan DDS arakatmanı simülasyon dünyasında hem kendi başına hem de HLA uyumlu simülasyon sistemleri ile birlikte kullanım alanına sahiptir.

MilSOFT, OMG'nin DDS standardına uygun bir DDS arakatmanı geliştirmiştir. MilSOFT, DDS arakatmanını Savaş Yönetim Sistemi yazılımında gerek gerçek zamanlı sistem ihtiyacının karşılanmasında gerekse simülasyon uygulamalarında etkin olarak kullanmaktadır. Bu açıdan DDS arakatmanı HLA arakatmanı olan RTI yerine kullanılabilmektedir.

Ayrıca, RTI-DDS arakatmanlarının birlikte kullanımının değişik uygulamalarda mümkün olabileceği görülmektedir. Gerçek sistemlerin dağıtık uygulamaları için DDS mimarisi ve arakatmanı kullanılırken, bu sistemlerin HLA uyumlu simülasyon sistemleri ile birlikte kullanımında / entegrasyonunda RTI kullanılması değerlendirilebilir. Bu kullanım biçimi için her iki arakatmanın ve mimari standardın özellikleri dikkate alınarak sistemlerin uygun şekilde tasarlanması mümkündür.

DDS arakatmanının RTI yerine kullanılması konusunda da detaylı araştırma yapılması ve uygulamalar ile denenmesine gerek duyulduğu değerlendirilmektedir.

KAYNAKÇA

[1] Calvo B., Seisdedos I. , (2006), "SimWare Kernel – Real Time Communication System for Simulation", Real-time and Embedded Systems Workshop 2006.

[2] Joshi R, Castellote G. P.,"A Comparison and Mapping of Data Distribution Service and High-Level Architecture", www.rti.com.

[3] MilSOFT, (2006), "MilSOFT DDS Middleware", http://dds.milsoft.com.tr

[4] OMG, (2006): "Data Distribution Service for Real-Time Systems Ver 1.2", http://www.omg.org.

[5] OMG, (2006), The Real-Time Publish-Subscribe Wire Protocol OMG Revised Submission, http://www.omg.org.

[6] Angelo Corsaro, DDS in Air Traffic Control Applications, DDS Information Day, December 2006, Washington, DC USA.

[7] Piet Griffioen , TACTICOS CMS: exploiting the full DDS potential, DDS Information Day, December 2006, Washington, DC USA.

[8] RTI, National Advanced Driving Simulators & Simulation Center, Customer success story, www.rti.com

[9] Whytock. P, (2006), "An Open Standards Approach To Simulator Design", http://europe.elecdesing.com, Article ID: 12025

FARKLI SİMÜLASYON SİSTEMLERİNİN ENTEGRASYONU İÇİN KONTROL MİMARİSİ

Deniz ALDOĞAN[(a)] , Seniha KÖKSAL [(a)], Cemil AKDEMİR [(a)]

İsa TAŞDELEN[(a)], Oğuz DİKENELLİ[(a)]

(a) TÜBİTAK MAM BTE, TÜBITAK Gebze Yerleskesi MAM BTE P.K.:21
41470 Gebze / KOCAELI
{deniz.aldoğan, seniha.koksal, cemil.akdemir, isa.tasdelen,
oguz.dikenelli}@bte.mam.gov.tr

ÖZ

Askeri eğitim amaçlı simülasyon uygulamaları, senaryo hazırlama, bilgisayar tarafından üretilmiş olan kuvvetlerin (Computer Generated Forces-CGF) benzetimi veya algoritmik davranışların benzetimi gibi işlevlerin gerçekleştirilmesi için kullanılır. Taktik simülasyonlarda gerekli olan senaryo hazırlama ve kuvvetlerin benzetimi gibi hizmetler için çeşitli ticari uygulama çatıları bulunmaktadır. Algoritmik davranışların benzetimi için gerekli ihtiyaçların bu ticari çözümler tarafından karşılanması her zaman mümkün olmamaktadır. Bu tür ihtiyaçlara yönelik olarak geliştirilen benzetim araçlarının ticari uygulama çatısı kullanan kontrol arayüzleri ve CGF uygulamalarıyla entegrasyonu önemli bir problemdir. Aynı uygulamada kullanılan farklı altyapılar için senaryo yönetimi, koşum kontrolü gibi kritik kontrol davranışlarında koordinasyonun sağlanması gerekir. Bunu gerçekleştirebilmek için, ticari kontrol mimarisinin özelleştirilerek kullanılması veya özgün bir kontrol mimarisinin geliştirilmesi iki farklı yöntem olarak önerilebilir.

Bu çalışmada, kontrol mimarisi için yukarıda önerilen yöntemler kıyaslanmış, ikinci yöntem ticari uygulamadan yalıtım, modülerlik ve tekrar kullanılabilirlik sağladığı için avantajlı bulunmuştur. Bu yöntem ile HLA (High Level Architecture) ve RPR-FOM (Real-Time Platform-Level Reference Federation Object Model) uyumlu bir simülasyon kontrol mimarisi geliştirilmiştir. Bu mimari, VR-Forces tabanlı kontrol arayüzü ve CGF uygulamaları ile algoritmik davranışların benzetiminin yapıldığı federeler kullanılarak test edilmiştir.

Anahtar Kelimeler: HLA, VR-Forces, Dağıtık Simülasyon, VR-Link, Simülasyon Kontrol Mimarisi

CONTROL ARCHITECTUE FOR INTEGRATION OF DIFFERENT SIMULATION SYSTEMS

ABSTRACT

Simulation applications for military training purposes provide features for building scenarios, simulation of computer generated forces (CGF) or simulation of algorithmic behaviors. There are various commercial application frameworks to support services necessary for tactical simulations such as building scenarios and simulation of forces. However, those frameworks are not always able to meet the needs of algorithmic behaviour simulation. The integration of simulation tools developed for the requirements of algorithmic behavior simulation with control interfaces and CGF applications based on commercial frameworks appears as a crucial problem. Coordination among different infrastructures throughout the same application must be maintained for critical control behaviours like scenario management and run control. To realize these aims, using a customized commercial control architecture or developing an original architecture are two significant methods.

In this study, the two methods proposed above have been compared. The second method is claimed to be more advantageous since it provides modularity, reusability and isolation from the commercial application. This method has been used in the simulation application to develop an HLA (High-Level Architecture) and RPR-FOM (Real-Time Platform-Level Reference Federation Object Model) compliant simulation control architecture. This architecture has been tested by using VR-Forces based control interface and CGF applications along with federates where algorithmic behaviours are simulated.

Keywords: HLA, VR-Forces, Distributed Simulation, VR-Link, Simulation Control Architecture

1. GİRİŞ

HLA(High-Level Architecture), dağıtık modelleme ve simülasyon için ortak bir mimari sağlamak amacıyla geliştirilen bir standarttır. HLA'nın temel bileşeni olan RTI(Run-Time Infrastructure), bir federasyonda, içerisinde simülasyon bileşenleri bulunduran federelerin temel işlemlerinin ve diğer federelerle olan iletişimlerinin koordinasyonunu desteklemek için gereken servisleri sağlar. Bu servisler, aşağıda sıralanmıştır[1]:

- Federasyon yönetimi
- Bildirim yönetimi
- Nesne yönetimi
- Aidiyet yönetimi
- Zaman yönetimi
- Veri Dağıtım Yönetimi

Taktik bir simülasyon uygulaması geliştirirken, RTI tarafından sağlanan bu temel servisler yanı sıra, senaryo hazırlama ve CGF uygulamalarına da ihtiyaç duyulmaktadır. Maliyet ve zaman kısıtları yüzünden, bu uygulamaları geliştirmek yerine ticari bir uygulama çatısı kullanmak tercih edilebilir.

Ticari uygulama çatıları, genel olarak, senaryo hazırlama ve bilgisayar tarafından üretilmiş olan kuvvetlerin (CGF) benzetimi için araçlar sağlar. Senaryo hazırlama aracı, simülasyon koşumu için gerekli taktik ortam, nesne, görev ve plan bilgilerinin hazırlanıp kaydedildiği uygulamadır. Bu arayüz kullanılarak koşum esnasında taktik resmin görüntülenmesi ve koşuma müdahale imkanları da sağlanır. CGF uygulama araçları ise, benzetimi yapılan nesnelerin oluşturulup, silinmesi, bu nesnelere ait görev, plan ve modellerin işletilmesi işlemlerini gerçekleştirmektedir.

Algoritmik davranışların benzetiminin yapıldığı uygulamalarda, yukardaki ticari çözümleri kullanmak yerine ihtiyaçlara yönelik olarak özgün benzetim araçlarının geliştirilmesi gerek maliyet gerekse tekrar kullanılabilirlik açısından daha uygun olabilmektedir. Ancak, özgün benzetim araçları ve ticari uygulama çatılarının birlikte kullanıldığı uygulamalarda entegrasyon önemli bir problem olarak karşımıza çıkmaktadır. Karşılaşılan bu problem, ticari uygulamada mevcut kontrol mimarisinin özelleştirilmesiyle veya tüm sistem için özgün bir kontrol mimarisinin geliştirilmesiyle çözülebilir.

Bu çalışmada yukarıdaki çözüm önerilerinden ikincisi benimsenmiştir. Bu yöntemin seçilmesiyle, kontrol mimarisi ticari uygulamalardan yalıtılarak modülerlik kazanmıştır. Ayrıca, HLA ve RPR-FOM[2] uyumlu geliştirildiği için bu standartlara uygun başka uygulama çatıları ile de kullanılabilir. Bu kontrol mimarisi, uygulama kapsamında VR-Forces tabanlı kontrol arayüzü ve CGF uygulamaları ile algoritmik davranışların benzetiminin yapıldığı federelerin entegrasyonunda test edilmiştir.

Bildirinin ilk kısmında genel olarak ticari uygulama çatıları ve örnek uygulama olarak seçilen VR-Forces mimarisi anlatılmıştır. İlerleyen bölümde geliştirilen federelerin entegrasyonu ve entegrasyon amacıyla VR-Forces uygulamasının özelleştirilmesi açıklanmıştır. Son olarak bu sistemin yönetilebilmesi amacıyla kullanılabilecek iki yöntem tartışılmış ve çalışmada anlatılan simülasyon uygulaması için seçilen yöntem detaylandırılmıştır.

2. Ticari Uygulama Çatılarının Genel Özellikleri

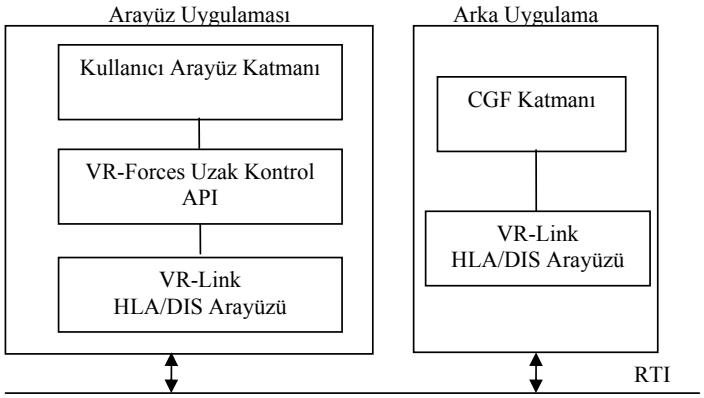
Ticari uygulama çatıları, temelde, senaryo hazırlama imkanı sağlayan kullanıcı arayüzleri ve benzetimi yapılan nesnelerin AI (Artificial Intelligence) motorları ile yönetildiği CGF uygulamaları gibi araçları sunar.

Bu uygulamalar, birbirinden bağımsız çalışan ayrı federeler veya hem arayüz hem de CGF uygulaması halinde tek bir federe olarak gerçekleştirilmiş olabilirler. Arayüz uygulaması simülasyonda kullanılacak senaryonun hazırlanması, nesnelerin oluşturulması, koşumun başlatılıp/durdurulması gibi işlemler için kontrol mesajları üretirken, CGF benzetiminden sorumlu arka uygulama ise bu mesajlara karşı gelen işlemlerin yapıldığı ve simülasyonun gerçekte işletildiği uygulama olarak tanımlanabilir.

İlerleyen bölümde, bu çalışmada seçilen ticari uygulama çatısı olan VR-Forces genel olarak anlatılacaktır.

2.1 VR-FORCES Mimarisi

VR-Forces, CGF benzetimi yapmak amacıyla yazılmış, kullanıcı arayüzü ve arka uygulama olarak ayrılabilen araçlara sahip ticari bir uygulama çatısıdır. DIS ve HLA katmanları üzerinde bir yalıtım sağlayan VR-Link uygulaması olarak gerçeklendiği için bu iki mimari üzerinde de çalışması mümkündür. Şekil 1'de VR-Forces mimarisinin genel bir görünümü verilmiştir.



Şekil 1. VR-Forces Mimarisi

VR-Link katmanı, bu altyapıyı kullanarak geliştirilen uygulamaların DIS veya HLA uyumlu simülasyonlarda kullanılabilmesi amacıyla yazılmış bir yalıtım katmanıdır. Bu katman, iki mimari tarafından sunulan hizmetleri uygulama kullanıcısına tek bir arayüz üzerinden sunarak, kullanıcıyı mimari detaylardan uzak tutmaktadır.

Uzak kontrol mimarisi, VR-Link katmanı üzerinde geliştirilmiş ve arka uygulamaların uzaktan kontrol edilebilmesi amacıyla yazılmış sınıflardan oluşmaktadır. Kontrol edilen VR-Forces arka uygulamalarıyla olan mesajlaşmalar alt seviyede tutulmakla birlikte, geliştirilen uygulama içerisinden bu mesajlara erişim sağlanır.

Uzak kontrol mimarisi üzerinde geliştirilen uygulamalar, senaryo yükleme/kaydetme, koşum başlatma/durdurma, nesne oluşturma/silme, oluşturulan nesnelere görev ve plan atama gibi kontrol işlevlerinin tamamını bu katman vasıtasıyla yerine getirebilmektedirler.

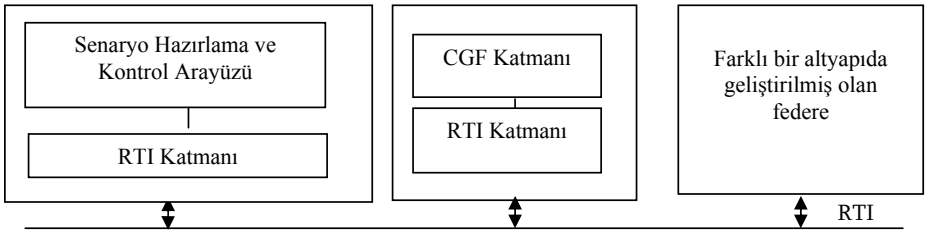
VR-Forces kullanıcı arayüz katmanı, uzak kontrol mimarisi üzerine geliştirilen, VR-Forces arka uygulamalarının görüntülenmesi ve kontrol edilebilmesini sağlayan bir katmandır. Bu katman, arayüz sınıfları Qt çatısı ile geliştirildiği için platform bağımsız bir arayüz uygulaması olarak değerlendirilebilir. Aynı zamanda, ağ arayüz sınıfları VR-Link mimarisi kullanılarak geliştirildiği için dağıtık simülasyon altyapılarından bağımsız bir kontrol uygulamasıdır.

CGF katmanına sahip VR-Forces arka uygulaması, kullanıcı arayüzünden gelen mesajları işleyerek nesnelerin benzetimini ve federasyondaki diğer federelerle iletişimi gerçekleştirir. Senaryo kapsamındaki yerel nesneler için, nesnelerin yaratılması, silinmesi, bu nesnelere atanan görevlerin gerçekleştirilmesi gibi işlevlerin yanı sıra,

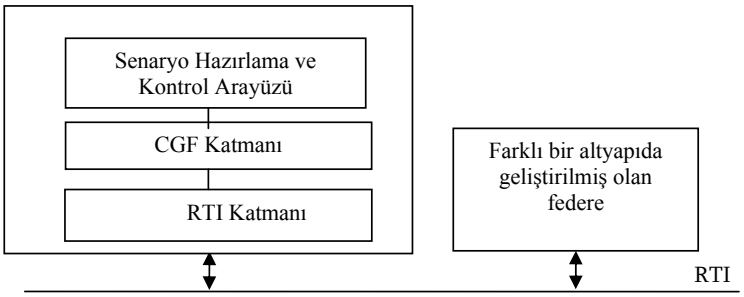
uzak nesneler için diğer federeler tarafından yayınlanan bilgilerin alınıp, depolanmasını sağlar[3].

3. HİBRİT ALTYAPILI UYGULAMALAR

Maliyet, performans ve sunulan hizmetler göz önünde bulundurulduğunda bir uygulamada birden fazla altyapının kullanılması tercih edilebilir. Birden fazla altyapı kullanılarak geliştirilmiş olan federelerin aynı federasyona katılıp haberleştikleri bu uygulamalar hibrit altyapılı olarak nitelendirilebilir. Hibrit altyapılı uygulamalarda federeler arasındaki haberleşme, federasyon nesne modeli (FOM) dosyasında tanımlanmış nesne ve iletişim sınıfları üzerinden yapılmaktadır. Şekil 2'de hibrit altyapılı bir federasyona ait iki ayrı mimari örnek olarak verilmiştir. Bu örneklerden ilkinde, senaryo hazırlanması ve simülasyon kontrolünden sorumlu uygulama ile kuvvet benzetimi yapan CGF uygulaması ayrı federeler olarak gerçeklenmiştir. İkinci örnekte ise tüm bu özellikler tek bir federe içerisinde gerçekleştirilmiştir.



a.CGF ve kontrolün ayrı federelerde gerçekleştirildiği hibrit altyapı mimarisi



b. CGF ve kontrolün tek federede gerçekleştirildiği hibrit altyapı mimarisi

Şekil 2. Örnek Uygulama Mimarileri

Hibrit altyapıların kullanılmasında karşılaşılan temel problem, her altyapıda federeler arasındaki mesajlaşmanın farklı şekilde gerçekleştirilmesidir. Bazı altyapılar için geliştirilmiş olan mesajlaşma sisteminin diğer altyapılar üzerindeki federeler tarafından anlaşılması ya da yönetim mesajlarının federasyon için ortak bir yapıda tanımlanması tüm federelerin aynı şekilde yönetilmesi açısından önem taşımaktadır.

3.1 VR-Forces Kullanılarak Geliştirilen Hibrit Altyapılı Uygulama

Bu çalışmada VR-Forces ve farklı bir altyapı kullanılarak meydana getirilmiş federelerden oluşan bir uygulama geliştirilmiştir. Tamamı VR-Forces altyapısı

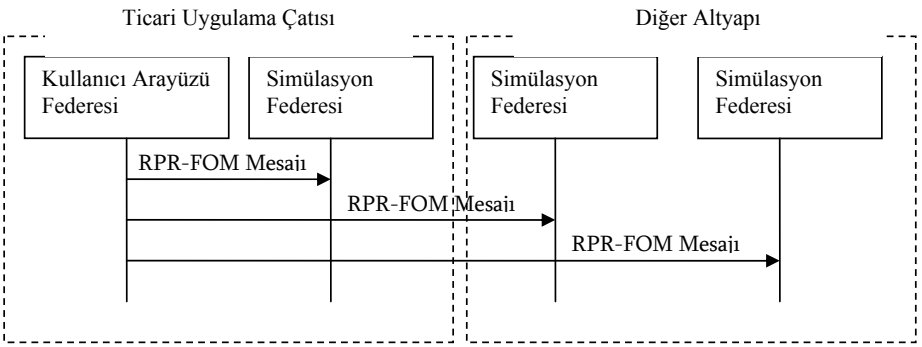
kullanılarak geliştirilen federasyonlarda, VR-Forces arayüz uygulaması tüm federasyonun yönetilmesi amacıyla kullanılabilmektedir. Hibrit altyapılı uygulamada, kontrolün yine kullanıcı tarafından VR-Forces arayüzü ile sağlanması beklenmektedir. Ancak standart VR-Forces kontrol mesajlarının tüm federasyonun kontrolü için kullanılması mümkün değildir. Bunun nedeni VR-Forces altyapısında tüm senaryo ve koşum kontrol mesajlarının RPR-FOM'da tanımlı aynı etkileşim sınıfı(*Data*) içerisinde gönderilmesidir. Bu etkileşim sınıfı alıcı tarafta çözümlenerek hangi VR-Forces kontrol mesajına karşı geldiği belirlenmektedir. VR-Forces arayüzü ve arka uygulamaları arasındaki tüm kontrol mesajları bu yapıya uygun olarak gönderilip alınmaktadır

Bu mesajlaşma yapısının VR-Forces altyapısı kullanılmadan geliştirilen federeler tarafından da kullanılabilmesi için bu çözümleme katmanının ilgili federe tarafından gerçekleştirilmiş olması gerekmektedir. Bu katmanın gerçeklenmesinden sonra tüm federelerin kontrol mesajlarının gönderildiği *Data* etkileşim sınıfına abone olmaları gerekir. Ancak böyle bir durumda tüm kontrol mesajları aynı etkileşim sınıfı üzerinden gönderildiği için, sadece belirli bir federeye gönderilmiş kontrol mesajları bile tüm federeler tarafından alınacak, bu durum ise ağ üzerindeki mesajlaşma yükünün artmasına sebep olacaktır. Bu sebeple çalışmada özgün altyapı kullanılarak geliştirilen federelerde bu katmanın gerçeklenmesi yapılmamış, tüm koşum kontrolünün standart RPR-FOM etkileşim sınıflarıyla gerçekleştirilmesi sağlanmıştır. VR-Forces altyapısı, bu standart yönetim mesajlarına uygun olarak geliştirildiği için, bu altyapı üzerine yazılmış federelerin de RPR-FOM'daki bu mesajlarla yönetilmesi mümkündür. Diğer altyapının kullanıldığı federeler tarafından da bu mesajlar yorumlanabildiği takdirde, tüm federasyonun yönetimi bu etkileşim sınıfları ile sağlanabilir.

3.1.1 Federasyon Kontrol Yöntemleri

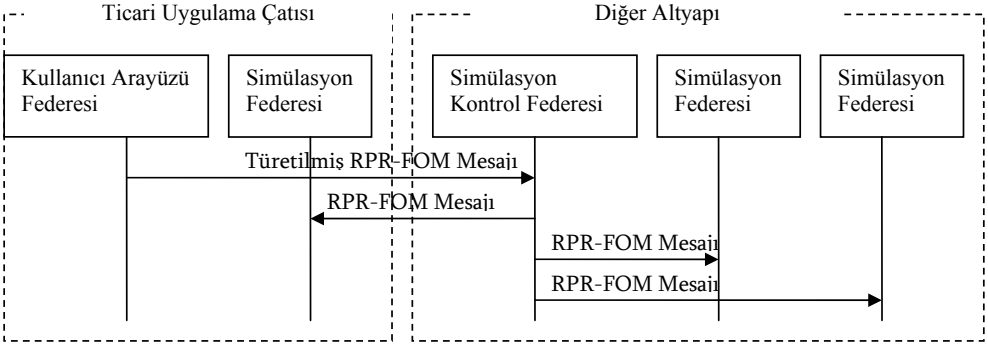
Tüm federelerin aynı tip yönetim mesajları ile kontrolü sağlandıktan sonra bu mesajların bütün federelere eş zamanlı gönderilmesi gerekmektedir. Bu işlem, iki farklı yolla gerçeklenebilir. Bunlardan ilki, VR-Forces altyapısında kontrol mekanizmasını içeren arayüz uygulamasına bu özelliğin kazandırılması, diğeri ise yönetim işlemlerinin arayüzden soyutlanarak farklı bir federe içinde yapılmasıdır.

VR-Forces kullanıcı arayüzünün özelleştirilmesiyle istenen noktalarda yukarıda bahsedilen etkileşim sınıfları gönderilerek tüm federasyonun kontrolünün sağlanması mümkündür. Bu amaçla, standart VR-Forces kontrol mesajlarının gönderildiği fonksiyonlar yeniden tanımlanarak bu fonksiyonlarda aynı kullanıcı etkileşimine karşılık RPR-FOM yönetim mesajlarının gönderilmesi sağlanabilir. Bu etkileşim Şekil 3'de gösterilmektedir.



Şekil 3. Özelleştirilmiş VR-Forces Mesaj Akışı

Federasyon kontrolünün gerçekleştirilmesinde kullanılacak diğer yaklaşımda ise, kullanılacak olan RPR-FOM yönetim mesajlarının federasyondaki bir kontrol federesi tarafından gönderilmesi sağlanmasıdır. Bu yöntemde, VR-Forces kullanıcı arayüzü ve kontrol federesi arasında, kullanıcı yönetim isteklerinin gönderilmesi için, ayrı etkileşim sınıfları tanımlanmalıdır. Simülasyon kontrol federesi arayüzden alacağı bu mesajlara karşılık olarak tüm federasyonun yönetilmesini sağlayan RPR-FOM mesajlarını gönderir. Bu mesaj akışı Şekil 4'de gösterilmektedir.



Şekil 4. Özelleştirilmiş VR-Forces ile Simülasyon Kontrol Federesi Mesaj Akışı

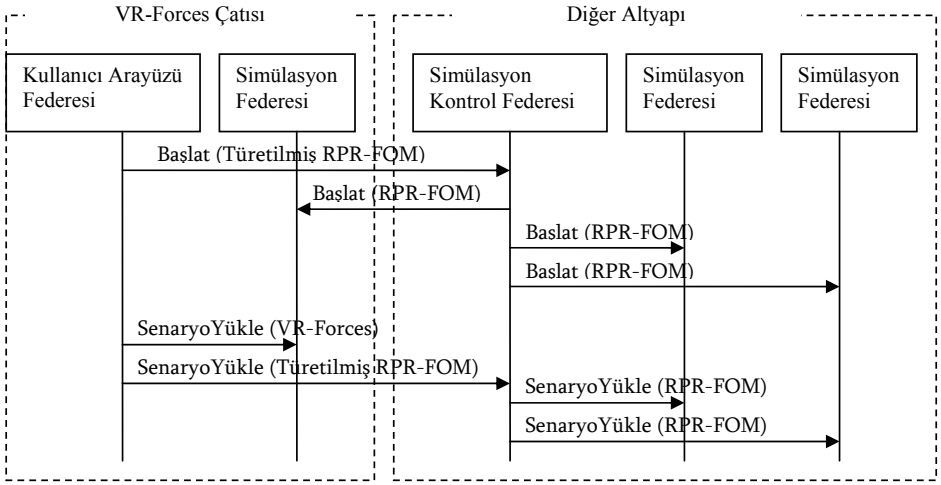
İlk yöntemde kullanıcı işlemlerine karşılık gelen kontrol mesajlarının dağıtımı arayüz uygulaması tarafından yapıldığı için bu yöntem ağ trafiğinin azaltılması bakımından avantajlıdır. Ancak, bu tip bir tasarımın kullanılmasıyla arayüz uygulamasının gerçekleştirilmesi daha karmaşık bir hale gelmektedir. Aynı zamanda, VR-Forces altyapısı tarafından zaman yönetim hizmetlerinin henüz desteklenmeyişi yüzünden, federelerin zaman yönetiminin uygulama tarafından yapılabilmesi sağlanamamaktadır[4].

İkinci yöntemde ise, arayüz ve kontrol federeleri arasında tanımlanan etkileşimlerden dolayı ağ trafiğinin artmasına rağmen, kontrol işlevlerinin arayüzden soyutlanması daha

modüler bir yaklaşım ortaya çıkarmıştır. Aynı zamanda, çalışmanın ilerleyen bölümlerinde zaman senkronizasyon hizmetlerinin simülasyon kontrol federesi tarafından sağlanabilmesi açısından uygulamada ikinci yaklaşımın kullanılması tercih edilmiştir.

3.1.2 Federasyon Kontrol Federesi İşleyişi

Simülasyon kontrol federesi ile geliştirilen mimaride kontrol işleyişi iki biçimde yürütülmektedir. İlk olarak, koşum kontrol işlemleri federasyonun tamamı için tek bir federe tarafından yürütülmektedir. Koşum kontrolü, simülasyonun başlatılması, durdurulması, koşum hızının belirlenmesi gibi işlemleri içermektedir. Bu işlemlerin tüm federeler tarafından senkron olarak gerçekleştirilmesi önemlidir. Bu tip kontrol işlemlerinde, VR-Forces arayüz ve arka uygulamaları arasındaki mesajlaşma engellenerek, bu mesajların arka uygulamaya simülasyon kontrol federesi tarafından gönderilmesi sağlanmıştır. Bu tip bir mesajlaşma akışı olan "Başlat" mesajının akışı Şekil 5'de gösterilmektedir. Kullanıcı arayüzünden başlatma isteği geldiğinde VR-Forces'ın kendi mesajlaşma akışı kesilmiş ve mesajın doğrudan simülasyon kontrol federesine gitmesi sağlanmıştır.



Şekil 5. Özelleştirilmiş VR-Forces ile Simülasyon Yönetim Federesi Mesaj Akışı Örnekleri

İkinci tip kontrol akışında ise varolan VR-Forces davranışının yanında uygulamada ihtiyaç duyulan kontrol akışı ayrı olarak sağlanmıştır. Bu durumda, VR-Forces arayüz ve arka uygulamaları arasındaki mesajlaşmaya paralel olarak simülasyon kontrol federesi tarafından diğer altyapı kullanılarak geliştirilmiş federelere kontrol mesajları gönderilmektedir. Senaryo yüklenmesi, senaryonun kaydedilmesi gibi işlemler anlatılan ikinci kontrol akışı içinde yeralmaktadır. Şekil 5'de gösterilen "SenaryoYükle" mesajı bu tip mesajlaşmaya örnektir. Kullanıcı arayüzünden senaryo yükleme isteği geldiğinde VR-Forces kendi senaryo işlemlerini yaptıktan sonra, simülasyon kontrol federesine bu

isteği bildirir. Simülasyon kontrol federesi senaryo yükleme işleminin kontrolünü yalnızca VR-Forces olmayan federeler üzerinde sağlamaktan sorumludur.

4. SONUÇ

Taktik benzetim uygulamalarının geliştirilmesinde seçilen ticari çatıların uygulama kapsamında hangi ihtiyaçlara yönelik olarak kullanılacağının belirlenmesi önem taşımaktadır. Bu çatılar tarafından sunulan hizmetlerin uygulama geliştirme sürecine ne ölçüde katkı sağladığının değerlendirilmesinin yanı sıra ticari uygulamanın getireceği maliyet de göz önünde bulundurulmalıdır.

Yapılan bu çalışmada, senaryo hazırlanması ve hazırlanan senaryo kapsamında nesne benzetiminin kritik olduğu federeler için ticari çatı kullanılması, hesaplamaya dayalı işlemlerin yoğunlukta olduğu diğer federelerin ise geliştirilen özgün bir altyapı üzerinde gerçeklenmesi tercih edilmiştir. Böylelikle, VR-Forces kullanımının getirdiği maliyet en aza indirilmiştir. Tüm sistemin kontrol edilebilmesi ise yine geliştirilen bir kontrol mimarisi üzerinden gerçekleştirilmiştir.

Geliştirilen kontrol mimarisinin test edilmesi amacıyla, bu çalışmada ticari ürün olarak VR-Forces kullanılmıştır. Ancak bu çatıyı kullanan federelerin kontrolünün geliştirilen kontrol mimarisi tarafından yapılabilmesi için VR-Forces uygulamaları özelleştirilmiştir. Böylelikle, tüm federasyonun entegrasyonu ve senkron bir şekilde kontrol edilmesi sağlanmıştır. Kontrol mimarisinin kullanılan ticari uygulama çatısından bağımsız ve RPR-FOM standardına uygun olarak geliştirilmesi, bu mimarinin ileride başka uygulamalar içerisinde veya başka ticari çatılarla birlikte kullanılabilmesini sağlayacaktır.

5. KAYNAKÇA

[1] IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) – Federate Interface Specification

[2] Shanks G., (2003), ”Real-time Platform Reference Federation Object Model (RPR FOM)”

[3] VR-Forces Developer’s Guide, MAK Technologies

[4] Horney T., Brännström M., Tyskeng M., Mårtensson J., Ng G., Gossage M., Ong M., Ang H., How K., (2004), ”Simulation Framework for Collaborative Fusion Research”, The 7th International Conference on Information Fusion, Stockholm

B2 OTURUMU:

FİZİK TABANLI MODSİM-II

SU ÜSTÜ GEMİLERİNİN AKUSTİK İZLERİNİN ZAMAN UZAYINDA MODELLENMESİ VE SİMÜLASYONU
Serkan AKSOY, Erkul BAŞARAN, Deniz BÖLÜKBAŞ, Sevgi AKGÜN

NORMAL MOD YÖNTEMİ KULLANILARAK SU ALTI PASİF AKUSTİK DALGA YAYILIMININ MODELLENMESİ VE SİMÜLASYONU
Serkan AKSOY, Hüseyin. Avni SERİM, Deniz BÖLÜKBAŞ, Sevgi AKGÜN

RADAR ARAYICI BAŞLIKLAR ve RADAR GÜDÜMÜNÜN MODELLENMESİ VE SİMÜLASYONU
Hale ÜSTÜNER, Yakup ÖZKAZANÇ

RADAR KESİT ALANI (RKA) MODELLEME VE SİMÜLASYONU
Gonca ÇAKIR, Mustafa ÇAKIR, Levent SEVGİ

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

SU ÜSTÜ GEMİLERİNİN AKUSTİK İZLERİNİN ZAMAN UZAYINDA MODELLENMESİ VE SİMÜLASYONU

Serkan AKSOY[a,b], Erkul BAŞARAN[a,b], Deniz BÖLÜKBAŞ[b], Sevgi AKGÜN[b]

[a] Gebze Yüksek Teknoloji Enstitüsü, Elektronik Mühendisliği Bölümü, Gebze, Kocaeli

{saksoy, erkul}@gyte.edu.tr

[b] TÜBİTAK-MAM, Bilişim Teknolojileri Enstitüsü, Gebze, Kocaeli

{deniz.bolukbas, sevgi.akgun}@bte.mam.gov.tr

ÖZ

Su üstü gemilerde mevcut çeşitli gürültü kaynakları nedeni ile oluşan akustik dalgaların gemi gövdesi ile etkileşimi sonucu deniz ortamına yayılımı ile gemilerin akustik izleri oluşmaktadır. Akustik iz bilgisi bir geminin tanınarak, sınıflandırılması bakımından son derece önemli olup, ulusal bilgi karakterindedirler. Bu çalışmada hareket etmeyen bir su üstü gemisinde tek frekanslı olarak modellenen motor ve pervane gürültüsü nedeni ile oluşan akustik dalgaların, gemi gövdesi ile etkileşimi sonucu deniz bölgesinde gemiden belli uzaklıkta tanımlanan bir yarım daire geometrisi boyunca meydana getirdiği akustik izi Zaman Uzayında Sonlu Farklar (*Finite Difference Time Domain*) yöntemi kullanılarak elde edilmiştir. Fiziksel model zaman uzayında akustik dalga denkleminin iki boyutta konuma ve zamana göre ayrıklaştırılıp, zamanda tekrarlamalı çözümü ile elde edilmiştir. Deniz yüzeyi ve geminin deniz üzerindeki yüzeyi basınçtan-arındırılmış (*Pressure-Release*) sınır koşulu ile modellenmiştir. Analitik bakımdan yan ve dibe doğru sonsuz uzun kabul edilen problem uzayının, nümerik bakımdan sonlandırılması için MUR türü sınır koşulu kullanılmıştır. Gemi gövdesi çelikten kabul edilmiş ve yoğunluk farkı ile modellenmiştir. Akustik alan dağılımı ve akustik izler tek başına motor veya tek başına pervane gürültüsünün var olması durumunda elde edilmiştir.

Anahtar Kelimeler: Akustik iz, su üstü gemisi, motor ve pervane gürültüsü, sonlu farklar, zaman uzayı.

TIME DOMAIN MODELING AND SIMULATION OF ACOUSTIC TRACE FOR SURFACE VESSELS

ABSTRACT

The acoustic traces of the surface vessels are calculated due to the interaction between the noise sources and the the physical structure of the vessel and it radiates throughout the underwater region. The acoustic traces of the surface vessel are really important for detection and classification of the targets for the national security demands. In this

work, the acoustical traces relating to the machinery and propeller noise interacting with the structure of the stationary surface vessel are calculated along the half circle from the vessel geometric center. Finite Difference Time Domain (*FDTD*) method is used for this aim as a numerical method. First of all, time domain wave equation is discretized along spatial and time variables. Then, the time domain solution is obtained, iteratively. In order to terminate acoustical problem space, MUR type absorbing boundary condition is applied. The physical structure of the vessel is assumed to be constructed by steel material. The simulation results are given independently for machinery and propeller noise.

Keywords: Acoustic trace, surface ship, machinery and propeller noise, finite difference, time domain.

1. GİRİŞ

Su altı ve su üstü gemilerinin akustik izlerinin bilinmesi, gemilerin tanınması ve sınıflandırılması bakımından ulusal güvenlik anlamında son derece önemlidir. Pratikte akustik iz genelde sadece frekansa bağlı olarak, gürültü spektrumunun eldesi ve çoğu kez de zamana göre izlenmesi anlamına gelse de, gerçekte akustik izler gemiden her yöne doğru her bir frekans için ayrı ayrı akustik dalga yayılım karakterinin çıkartılmasını gerektirir [1]. Bu analiz daha sessiz gemilerin tasarlanması ve sonarın kendi gürültüsü (*Self-Noise*) bakımından optimize edilmesine yardımcı olacaktır.

2. FİZİKSEL MODEL VE ÇÖZÜM YÖNTEMİ

Fiziksel model su altında akustik dalgaların yayılımı karakterini tanımlayan diferansiyel denklemlerin zaman uzayında uygun başlangıç ve sınır koşulları uygulanarak çözümü üzerine kurulmuştur. Bu amaçla matematiksel bakımdan temelde iki farklı yol izlemek mümkündür. Birinci yol birinci mertebeden iki ayrı akustik diferansiyel denklemin çözülmesi, ikinci yol ise ikinci mertebeden diferansiyel denklemin çözülmesi üzerine kuruludur. Bu çalışmada ikinci yol izlenmiş ve iki boyutlu olarak çözüm sağlanmıştır. Buna göre $u(\vec{r},t)$ konuma ve zamana bağlı akustik basınç olmak üzere

$$\Delta u(\vec{r},t) - \frac{1}{v^2}\frac{\partial^2 u(\vec{r},t)}{\partial t^2} = 0 \qquad (0.1)$$

denklemi çözülmelidir. Burada v deniz ortamında ses hızını, Δ Laplace operatörünü gösterir. Bu çalışma Kartezyen koordinatlar sisteminde ve iki boyutlu olarak ele alındığından, Laplace operatörü $\partial^2/\partial x^2 + \partial^2/\partial y^2$ halini alır. (0.1) denklemi zaman uzayında dalga denklemi olarak bilinir ve çözümünün sağlanabilmesi için, uygun başlangıç ve sınır koşullarına ihtiyaç vardır. Deniz yüzeyi ve geminin deniz üstündeki yüzeyi Dirichlet sınır koşulu olarak ta bilinen, basınçtan arındırılmış sınır koşulu ile

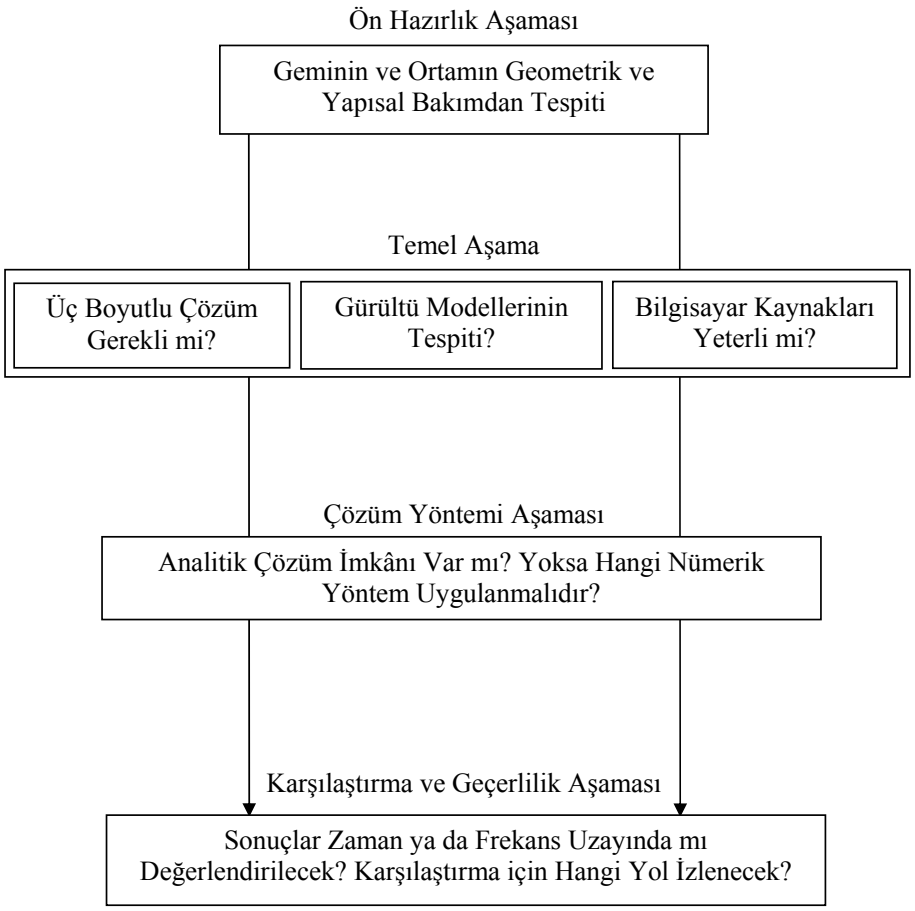
modellenmiştir. Bu durum hava olarak modellenen denizin üst bölgesinde akustik dalga yayılımına izin verilmediği anlamına gelir. (0.1) denkleminin çözümü için Zaman Uzayında Sonlu Farklar (ZUSF) yöntemi kullanılmıştır. Bu yöntem zaman uzayında dalga denkleminin konumda ve zamanda ayrıklaştırma yapılarak, zamanda tekrarlamalı çözüme dayalıdır. Buna göre (0.1) denkleminin zamanda ve konumda ayrıklaştırılması sonucu

$$u^{n+1}(i,j)=\left[1-\left(\frac{v\Delta t}{\Delta x}\right)^2-\left(\frac{v\Delta t}{\Delta y}\right)^2\right]+2u^n(i,j)-u^{n-1}(i,j)$$
$$+\left(\frac{v\Delta t}{\Delta x}\right)^2\left[\left(u^n(i+1,j)+u^n(i-1,j)\right)\right]$$
$$+\left(\frac{v\Delta t}{\Delta y}\right)^2\left[\left(u^n(i,j+1)+u^n(i,j-1)\right)\right].$$
(0.2)

bulunur. Burada Δx, Δy ve Δt konuma ve zamana göre ayrıklaştırma adımlarını, n zamanda hesap tekrar sayısını gösterir. Problemi uzayının sonlandırılması için MUR sınır koşulu kullanılmıştır [2]. Deniz bölgesinde ve su üstü gemisinin geometrik merkezinden belli bir uzaklıkta yarım bir yarım daire boyunca kaydedilen zaman uzayındaki akustik işaretlerden Hızlı Fourier dönüşümü ile frekans uzayında akustik dalganın genlikleri elde edilmiştir. Zaman Uzayında Sonlu Farklar yöntemi ile çözümde kararlılık analizleri kapsamında $\Delta x=\Delta y=\lambda/20$ olmak üzere, dalga boyunun yirmide biri olacak şekilde seçilmiştir. Bu durumda problemin gerçek fiziksel boyutu *164 m x 82 m* ve dalga boyu uzunluğu bakımından boyutu $27\,\lambda \times 54\,\lambda$ olmaktadır. Toplam hücre sayısı *654 x 327*'dir. Buna göre Courant kararlılık kriterine göre zaman adımı $\Delta t = 5{,}657\times10^{-5}$ *sn* olup, zamanda toplam tekrarlama sayısı *n=4000*'dir. Yukarıda bahsedilen verilen kapsamında problemin her bir senaryosu *P4-3.2 GHz* işlemcisine ve *4 GB* hafıza sahip bir kişisel bilgisayarda *370 sn*'de çözülebilmektedir.

3. MODELLEME TEMEL AŞAMALARI

Su üstü gemilerinin akustik izlerinin çıkartılması amacı ile modelleme ve simülasyonunun yapılabilmesi için, temel modelleme adımlarının belirlenmesi gereklidir. Şekil 1'de böyle bir sistematik üzerine kurulu olarak söz konusu temel modelleme adımları gösterilmiştir.

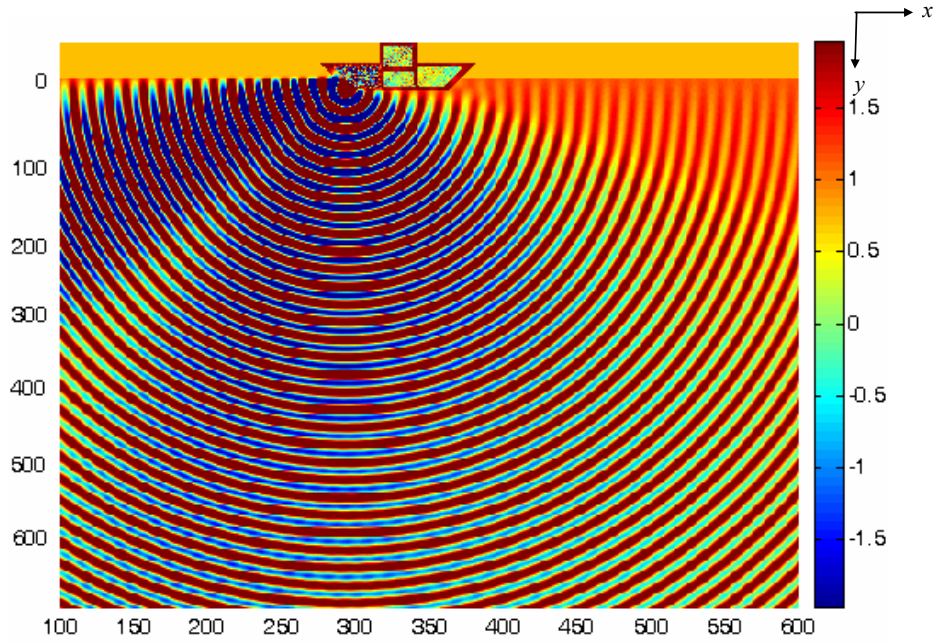


Şekil 1. Su üstü gemilerinin akustik izlerinin çıkartılması için modelleme ve simülasyon temel adımları.

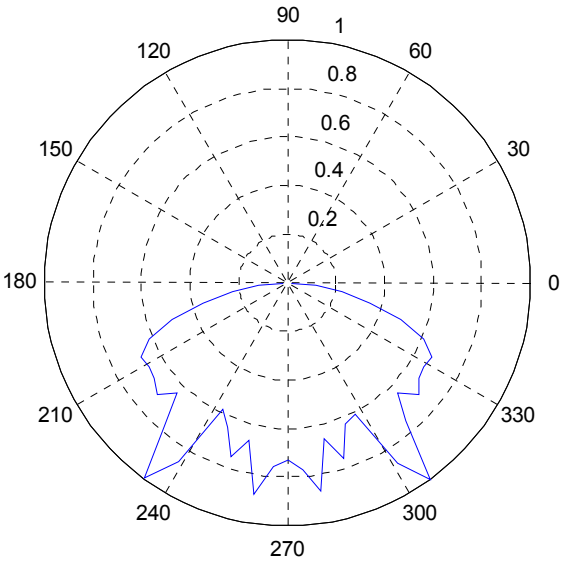
4. SONUÇLAR

Örnek bir problem olarak *25 m* uzunluğa ve *8,25 m* yüksekliğe sahip ve hemen hemen yarısına kadar denize batmış, hareket etmeyen model bir geminin akustik izleri çıkartılmıştır. Sudaki ses hızı *1500 m/sn*, gemi gövdesindeki ses hızı *2500 m/sn*, havadaki ses hızı *340 m/sn* alınmıştır. Gemi gövdesi ile etkileşen pervane gürültüsü nedeni ile oluşan akustik dalgaların dağılımı Şekil 2'de gösterilmiştir. Gemi gövdesi yoğunluk farkları ile modellendiğinden, Şekil 2'de görülebileceği gibi gemi içerisine akustik dalga nüfuzu da dikkate alınarak, çözüm yapılmıştır. Bu durumda oluşabilecek iç rezonanslar v.b. tüm etkiler direkt olarak çözüme dâhil edilmektedir. Deniz yüzeyinde kalan kısmı hariç, gemi gövdesi çelikten kabul edilmiş ve geminin iç bölümlerinde ki

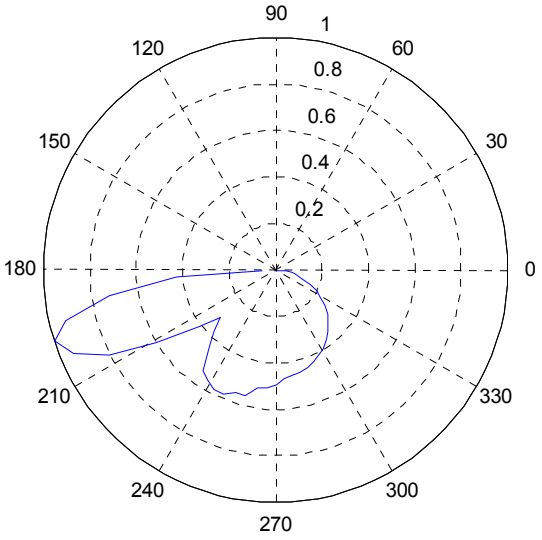
boşluklar hava olarak modellenmiştir. Bu durum gemi gövdesinin malzeme ve geometrik özelliklerinin geminin akustik izine etkisinin incelenmesine imkân vermektedir ve gerçekçi akustik izlerin elde edilmesi bakımından son derece önemlidir. Motor ve pervane gürültüsü noktasal kaynak olarak modellenmiş olup, *80 dB* genlikli ve *500 Hz*'de tek frekanslı sinüzoidal işaretler olarak uygulanmıştır. Deniz bölgesinde ve su üstü gemisinin geometrik merkezinden *16 m* yarıçaplı bir uzaklıkta yarım bir daire boyunca kaydedilen zaman uzayındaki akustik işaretlerin Hızlı Fourier dönüşümü ile frekans uzayına geçirilmesi sonucu akustik dalganın genlikleri elde edilmiştir. Bu aşamada zaman uzayında ki işarette geçici (*transient*) kısımlar çıkartılarak, işaretin sadece sinüzoidal sürekli haldeki (*steady state*) kısmı kullanılmıştır.



Şekil 2. *25 m* uzunluklu ve *8,25 m* yükseklikli model gemi ve pervane gürültüsünün gemi gövdesi ile etkileşimi sonucu oluşan akustik alan dağılımı.



Şekil 3. Motor gürültüsü nedeni ile oluşan akustik iz paterni.



Şekil 4. Pervane gürültüsü nedeni ile oluşan akustik iz paterni.

Simülasyon boyunca gürültü kaynakları ile gemi gövdesinin akustik bakımdan etkileşiminin doğru modellenmesi için yeterli simülasyon zamanının geçmesine izin verilmiştir. Buna göre sadece motor gürültüsünün var olması durumunda akustik iz paterni Şekil 3'de verilmiştir. Yine sadece pervane gürültüsünün var olması durumunda akustik iz paterni Şekil 4'de verilmiştir.

4. YORUM ve GELECEK ÇALIŞMALAR

İncelenen gemi geometrisi ve uygulanan frekans kapsamında, pervane gürültüsünün daha çok geminin kıç bölümüne doğru etkin olduğu, motor gürültüsünün ise motorun simetrik yerleşimi nedeni ile geminin alt yüzünden itibaren simetrik olarak etkili olduğu görülmektedir. Her iki gürültünün aynı anda var olması durumunda ise gürültülerin toplanması gereği, geminin kıç bölümünde gürültünün daha baskın olduğu gözlenmiştir. Çalışma gemide kullanılacak sonar'ın gemi ile ilgili kendi gürültüsünün optimize edilmesi ve gemilerin türüne göre sınıflandırılabilmesi bakımından ulusal güvenlik uygulamalarında önem taşımaktadır. Bu çalışmada sadece iki farklı fakat aynı frekanslı ve dar bandlı gürültü kaynağı modellenmiştir. Gelecekteki çalışmalarda gerçekçi ve geniş bandlı ikiden fazla akustik kaynak nedeni ile empedans türü sınır koşulu ile modellenen gemi gövdesi ile etkileşerek oluşan gürültü diyagramları incelenecektir. Yine hâlihazırda iki boyutlu olarak çözülen problemin, yüksek performanslı ve paralel bilgisayarlar ile üç boyutlu olarak gerçek gürültü ve gemi modelleri kullanılarak çözümü gelecek çalışma planları arasında yer almaktadır.

KAYNAKÇA

[1] (1965), Introduction to Sonar Technology, Bureau of Ships, Department of the Navy, Navships, 0967-129-3010.

[2] A. Taflove and S. C. Hagness, (2005), Computational Electrodynamics: The Finite-Difference Time-Domain Method, 3rd ed. Norwood, MA: Artech House.

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

NORMAL MOD YÖNTEMİ KULLANILARAK SU ALTI PASİF AKUSTİK DALGA YAYILIMININ MODELLENMESİ VE SİMÜLASYONU

Serkan AKSOY[a,b], Hüseyin Avni SERİM [a,b], Deniz BÖLÜKBAŞ[b], Sevgi AKGÜN[b]

[a] Gebze Yüksek Teknoloji Enstitüsü, Elektronik Mühendisliği Bölümü, Gebze, Kocaeli
{saksoy, haserim}@gyte.edu.tr
[b] TÜBİTAK-MAM, Bilişim Teknolojileri Enstitüsü, Gebze, Kocaeli
{deniz.bolukbas, sevgi.akgun}@bte.mam.gov.tr

ÖZ

Su altı akustiğinde temel problem aktif ve pasif akustik dalga aktarımı olarak iki ana kısımda ele alınabilir. Pasif aktarım özellikle sonar sistemlerinin kullanılarak su altı ve su üstü hedeflerin tespitinde son derece önemlidir. Bu çalışmada Normal Mod yöntemi kullanılarak sonar performans modellerinin tahmini ve analizi için pasif su altı akustik dalga yayılımı probleminde fiziksel modelin kurulmasındaki adımlar ve fiziksel tabanlı matematiksel model kapsamında simülasyon için temel gereksinimlerden bahsedilmiştir. Bu kapsamda öncelikle model çözüm yönteminden bahsedilmiş, daha sonra modelleme için temel aşamalar anlatılmıştır. Örnek problem olarak su altı akustiğinde PEKERIS dalga kılavuzu adı ile bilinen iki katmanlı deniz ortamında pasif akustik dalga yayılımı temel modelleme kapsamında bahsedilen aşamalar bakımından değerlendirilmiştir. Normal Mod yöntemi kullanılarak hesaplanan akustik basıncın genliği kullanılarak menzile göre yayılım kaybının değişim karakteri yorumlanmaya çalışılmıştır. Böylece sonar denklemi kapsamında söz konusu yayılım kayıpları ve ilgili gürültü modellerinin pasif sonar performansı üzerine etkisinin araştırılması mümkün olabilecektir.

Anahtar Kelimeler: Pasif aktarım, Sonar, Normal Mod Yöntemi, Yayılım Kaybı, Sonar performansı.

MODELING AND SIMULATION OF PASSIVE UNDERWATER ACOUSTIC WAVE PROPAGATION BY NORMAL MODE METHOD

ABSTRACT

The main problem in underwater acoustic can be considered as active and passive acoustic wave propagation. The passive acoustic wave propagation is especially important for the detection and the classification of ships and underwater targets. In this work, it is mentioned about the main steps for the construction of the physical model in the frame of Normal Mode method which is used in order to estimate the performance modeling of the sonar systems. The fundamental steps are considered for the simulation of the underwater acoustic wave propagation problem in the sense of the mathematical model based on the physical structure of the underwater acoustic problem. In that manner, first of all, the solution method for the considered model is analysed and the fundamental steps are considered for the simulation. As an example, PEKERIS problem is evaluated in the sense of above-mentioned steps. The transmission loss is calculated from the acoustic pressure which is extracted form the solution of the problem by Normal Mode method. Then, the character of the transmission loss depending on the range is evaluated. Thus, the effect of the transmission loss and the noise models on the passive sonar performance can be investigated. The sonar equation is also considered for this purpose.

Keywords: Passive transmission, Sonar, Normal Mode Method, Transmission Loss, Sonar performance.

1. GİRİŞ

Ulusal savunma gereksinimleri kapsamında üzerinde durulması gereken konulardan birisi de su altı ve su üstü hedeflerin tespiti ve sınıflandırılmasıdır. Bu amaç için su altı akustik dalga yayılımından aktif ve pasif aktarım kapsamında faydalanılabilinir [1]. Aktif akustik aktarımda kendi gemimizin tespit edilme ihtimalini yüksek olduğundan, daha çok pasif aktarım ile hedef gemilerin tespiti önem taşımaktadır. Bu kapsamda su altı ve su üstü hedeflerinin tespitinin daha başarılı olması için ve özellikle de askeri harekât öncesi hedefin yerinin doğru olarak tahmini bakımından en önemli aşamalardan birisi, harekâtın yapıldığı su altı ortamında pasif akustik dalga yayılımının matematiksel modelinin kurulup, çözülerek gerçekçi yayılım kayıplarının hesaplanabilmesidir. Pasif aktarımda bu amaç için kullanılan çeşitli yöntemler mevcut olmakla beraber, en yaygın olanlarından birisi Normal Mod yöntemidir [2]. Bu çalışmada pasif akustik aktarım ile hedeflerin tespiti ve kendi gemimizin hedeflerden sakınımının başarılı biçimde sağlanması için kullanılan fiziksel model kapsamında oluşturulan matematiksel model çözümlerine dayalı bir simülasyonda, gerçekçiliği ve güvenilirliği artırmak için izlenmesi gereken temel adımlar ele alınmıştır.

2. FİZİKSEL MODEL VE ÇÖZÜM YÖNTEMİ

Normal Mod yöntemi Helmholtz denkleminin çözümü prensibine dayanır. Helmholtz denkleminin tam çözümleri dalga sayısı üzerinden integral alınarak elde edilebilir. Bu

integral basit deniz ortamlarında reel eksen üzerinde değerlendirilirken, kompleks ve kayıplı durumlar için, reel ve imajiner düzlemde değerlendirilmelidir. Kompleks düzlemde integral konturunun zorlanması (*Deformation*) ile tam normal mod çözümlerine ulaşılır. Bu durum kompleks uzayda kutupları ve bu kutuplarda rezidüleri ortaya çıkarır. Söz konusu rezidüler de ayrık modları tanımlar. Tam akustik çözüm söz ayrık modların toplamı olarak ortaya çıkar. Bu durumda monokromatik noktasal kaynakların oluşturduğu akustik dalgalar için (pratikte işaretler çoğu kez dar-bandlı varsayılabileceğinden) frekans uzayı dalga denklemi aşağıdaki gibi verilir.

$$\mathrm{D}\,u(\mathbf{r}) + k^2 u(\mathbf{r}) = f(\mathbf{r}) \tag{1.1}$$

Helmholtz denklemi olarak bilinen bu denklemde D Laplace operatörünü, k dalga sayısını, f akustik kaynakları, u akustik dalga basıncını gösterir ve tam çözümleri

$$u(\mathbf{r}) = \int_{-\infty}^{\infty} u(\mathbf{r},k)dk + \text{Kesim çizgisi katkısı} \tag{1.2}$$

integrali ile verilir. Bu çözümde integral ayrık (*Discrete*) modları, kesim çizgisi katkısı ise sürekli (*Continuous*) v.b. diğer modları tanımlar [2]. Normal mod yöntemi ayrık modların hesabı bakımından silindirik koordinatlarda ve iki boyutlu olarak (1.1) denkleminin değişkenlerin değişimi yöntemi ile değerlendirilmesi sonucu, akustik basıncın

$$u(r,z) = \frac{ip}{r_o}\sum_{n=1}^{\infty} U_n^*(z_o)\,U_n(z)\,H_0^1(k_{rn}r) \tag{1.3}$$

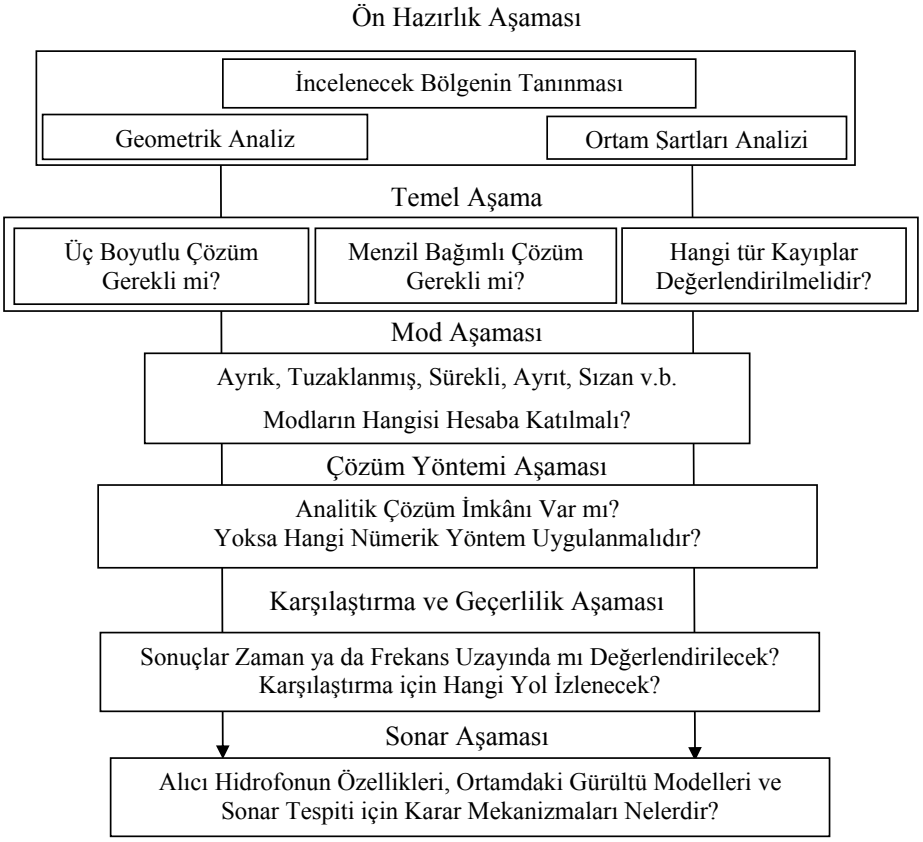
olarak bulunmasına imkan tanır. Burada U_n ve U_n^* derinlikle ilgili baz fonksiyonlarını, H_0^1 sıfırıncı mertebeden ve birinci türden Hankel fonksiyonunu gösterir. Bu kapsamda pasif su altı akustik dalga yayılımı modelleme ve simülasyonu bakımından, temel problem (1.1) denklemi kapsamında (1.3) çözümünün doğru anlaşılarak, değerlendirilmesi meselesidir. Koordinat sistemleri bakımından ayrıklaştırılabilir geometrilere sahip olan deniz bölgelerinde (1.1) denklemi analitik olarak çözülebilmesine rağmen, gerçekçi geometrik ve deniz dibi bölgesine sahip karmaşık deniz ortamlarında, (1.3) denkleminin analitik olarak değerlendirilmesi son derece zor ve hatta mümkün olmamaktadır. Bu durumda nümerik olarak çözüm sağlanması gerekmekte, bu durum ise hangi durumda hangi nümerik yöntemin kullanılmasının en etkin ve doğru sonuç vereceği v.b. sorunlara neden olmaktadır. (1.3) denkleminin değerlendirilmesi sonucu, karmaşık sayı olarak elde edilen akustik basıncın genliği kullanılarak, yayılım kayıpları

$$Yayılım\ kayıpları = 20\log|u(r,z)| \tag{1.4}$$

formülü ile hesaplanıp, sonar denkleminde yerine konularak pasif aktarım için sonar performans tahmin ve analizlerinin yapılabilmesine olanak sağlar.

3. MODELLEME TEMEL AŞAMALARI

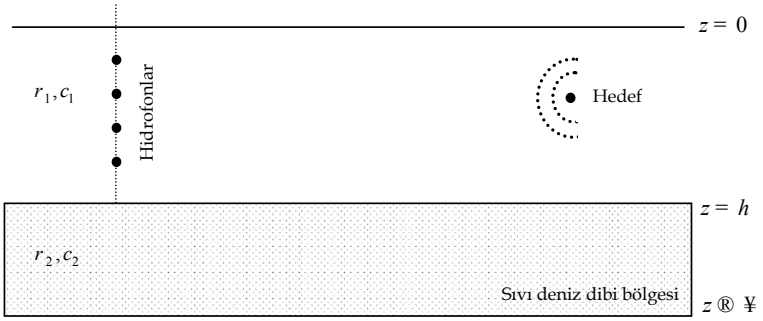
Pasif su altı akustik dalga yayılımının Normal Mod yöntemi kullanılarak modelleme ve simülasyonunun yapılabilmesi için, temel modelleme adımlarının belirlenmesi gereklidir. Şekil 1'de böyle bir sistematik üzerine kurulu olarak temel modelleme adımları gösterilmiştir.



Şekil 1. Normal Mod yönteminin pasif su altı akustik aktarımı simülasyonunda kullanımı için temel modelleme adımları.

4. ÖRNEK PROBLEM (*PEKERIS dalga kılavuzu*)

Modelleme temel aşamaları sistematiği çerçevesinde, su altı akustiğinde yaygın olarak kullanılan ve PEKERIS dalga kılavuzu [3] olarak bilinen iki katmanlı *h* derinlikli ve düz deniz ortamında pasif su altı akustik dalga yayılımı problemi incelenmiştir. Şekil 2'de PEKERIS dalga kılavuzunun geometrik yapısı gösterilmiştir.



Şekil 2. İki katmanlı sıvı–sıvı su altı akustik bölgesi (PEKERIS dalga kılavuzu).

Yukarıdaki şekil kapsamında her bir bölge için ρ yoğunluğu, c ses hızını, z_0 ve z_g sırası ile kaynak ve gözlem noktalarının koordinatlarını gösterir. Modelleme temel aşamaları sistematiği izlenerek PEKERIS problemi aşağıdaki gibi değerlendirilmiştir.

Ön Hazırlık Aşaması:

Geometrik Analiz: Bölge deniz yüzeyi ve dip geometrisi bakımdan düz olup, sıvı-sıvı olarak iki katmanlı yapıya sahiptir. Deniz dibi bölgesi yarı-uzay (*Half-Space*) olarak modellenmiştir. Bölge geometrisi kartezyen ya da silindirik koordinatlar için uygundur.

Ortam Şartları Analizi: Bölge deniz ve deniz dibi bakımından kayıpsızdır. Deniz ve deniz dibi bölgesinde ayrı ayrı ses hızı ve yoğunluk bilgisine ihtiyaç vardır. Ses hızı ve yoğunluk profili her iki bölgede de sabittir.

Temel Aşama:

Bölgede yanal olarak deniz boyunca geometrik ve yapısal değişim olmadığı kabul edildiğinden, üç boyutlu çözüme gerek yoktur. Ses hızı profilleri deniz ve deniz dibi bölgesinde sabit kabul edildiğinden, menzil bağımlı çözüme gerek yoktur. Tüm bölgede kayıplar hesaba katılmadığından, dikey (*Compressional*) veya yatay (*Shear*) dalga titreşim kayıpları v.b. kayıpların değerlendirilmesine gerek yoktur. Silindirik yayılım ve geometrik dispersiyon kayıpları zaten normal mod çözümlerinde değerlendirilmektedir.

Mod Aşaması:

Bölgenin geometrik ve yapısal özellikleri nedeni ile ayrık ve sürekli modlarla beraber, sürekli spektrumlu modların da hesaba katılması gerekmektedir. Aksi takdirde deniz ve

deniz dibi sınırında akustik dalga geçişlerinin doğru olarak modellenmesi mümkün olmayacağından, yayılım kayıpları hesaplarının güvenirliği azalacaktır.

Çözüm Yöntemi Aşaması:

Bölge geometrik bakımdan koordinat sistemler dâhilinde ayrıklaştırılabilir olduğundan, analitik çözüm kartezyen veya silindirik koordinat sisteminde mümkündür. (1.1) denkleminin değişkenlerin değişimi yöntemi kapsamında değerlendirilmesi ile, uygun sınır, süreklilik ve radyasyon koşulları uygulanarak Normal Mod yöntemi ile analitik çözüm sağlanabilir.

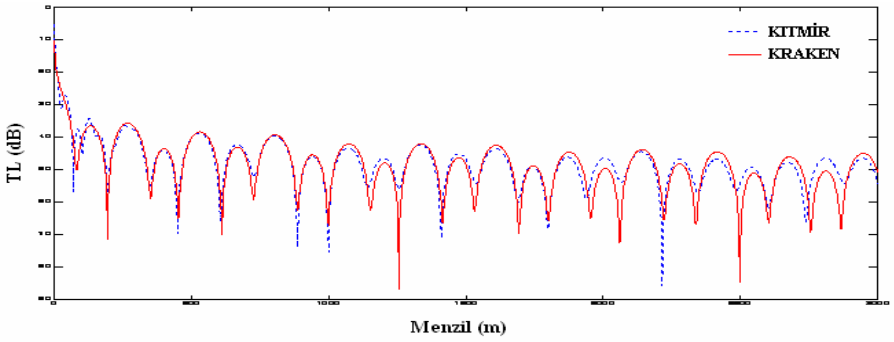
Karşılaştırma ve Geçerlilik Aşaması:

Analitik çözümün sağlanması ile birlikte (1.4) denklemi kullanılarak hesaplanan yayılım kaybı değerleri, menzile ve frekansa bağlı olarak, mevcut bir başka ve kabul görmüş hazır kodlarla (KRAKEN v.b.) veya ölçüm verileri ile karşılaştırılabilir. Karşılaştırma aşamasında kullanılacak hazır programların çoğunun nümerik temelli olduğu unutulmamalıdır. Nümerik hesap tekniklerinin getirdiği çeşitli dezavantajlar nedeni ile bazı kritik durumlar için karşılaştırma sonuçları yeteri kadar iyi olmayabilir. Simülasyon sonucu elde veriler gerek hazır kodlar, gerek ölçüm sonuçları ile karşılaştırılmadan önce, mümkünse hazır kodların ve ölçüm sonuçlarının standart karşılaştırma testlerinden (*Benchmarking*) geçmiş olmasına özen gösterilmelidir [4].

Sonar Aşaması:

Pasif akustik aktarımında sonar bakımından önemli bir durumda, sonarın dizi halinde alıcı hidrofon sistemine sahip olmasıdır. Bu nedenle sonar performansı bakımından Normal Mod çözümlerinin değerlendirilmesi için, birden fazla alıcı dizi halinde dikkate alınarak sonuçlar değerlendirilmelidir. Diğer bir kritik noktada ortamdaki gürültünün yayılım yaptırılarak mı yoksa sadece model olarak mı hesaba katılacağıdır.

Bahsedilen çalışmalar kapsamında $f=60\ Hz$, $h=75\ m$, $z_0=25\ m$, $z_g=50\ m$, $c_1=1400\ m/s$, $c_2=2000\ m/s$, $\rho_1=1\ gr/cm^3$, $\rho_2=2\ gr/cm^3$ parametrelerine sahip PEKERIS dalga kılavuzunda KRAKEN [5] ve geliştirdiğimiz kod (KITMİR) arasında Şekil 3'de yayılım kayıpları bakımından karşılaştırma yapılmış ve iyi sonuçlar elde edilmiştir.



Şekil 3. KRAKEN ve KITMİR arasında yayılım kayıpları bakımından karşılaştırma.

5. YORUM ve GELECEK ÇALIŞMALAR

Bu çalışmada pasif su altı akustik aktarımı problemlerinin çözümünde yoğun olarak kullanılan Normal Mod yönteminin, sonar performans tahmini ve analizleri için modelleme ve simülasyon amaçlı kullanılabilmesi ve özellikle gerçekçi simülasyonların yapılması için fiziksel modelin tanımlanması ile ilgili matematiksel işlemler kapsamında gerekli temel aşamalar anlatılmıştır. Örnek bir problem olarak PEKERİS dalga kılavuzu incelenmiştir. Pürüzlü, eğimli, çukurlu, tepeli, kayıplı v.b. karmaşık daha gerçekçi deniz ve deniz dibi ortamları için verilen sistematik kapsamında gelecekte fiziksel temelli modelleme ve simülasyonlar yapmak mümkündür. Gerçekçi deniz ortamınlarında matematiksel modelin ve mod yapısının daha karmaşık olması beklenmektedir. Yine su altı akustiğinde gerçekçi simülasyonların karmaşık ve ağır matematiksel işlemler gerektirmesinden dolayı, akustik dalga yayılımının fiziksel modelinin kurulmasında etkin derinlik (*Effective Depth*) v.b. yardımcı ve etkin teknikler mevcuttur. Gelecekte daha basit ve etkin olarak bu tür yöntemlerin modelleme ve aşamalar bölümünde anlatılan sistematik kapsamında değerlendirilmesi hedeflenmektedir. Üç boyutlu Normal Mod çözümlerinin zaman uzayında elde edilmesi de planlanan diğer bir çalışmadır.

KAYNAKÇA

[1] Urick R.J., (1983), Principles of Underwater Sound, Peninsula Publishing.

[2] Paul C. Etter, (2003), Underwater Acoustics Modelling and Simulation: Principles, Techniques and Applications, 3 Ed., Taylor & Francis.

[3] Pekeris C. L., (1948), "Theory of propagation of explosive sound in shallow water", Geol. Soc. Am. Mem., 27, pp. 1 -117.

[4] Tindle C. T. & Zhang Z. Y., (1997), "An adiabatic normal mode solution for the benchmark wedge", Journal of the Acoustic Society of America, 101-1, pp. 606-609.

[5] Porter M. B., (2001), The KRAKEN Normal Mode program, SACLANT Undersea Research Centre.

RADAR ARAYICI BAŞLIKLAR ve RADAR GÜDÜMÜNÜN MODELLENMESİ ve SİMULASYONU

Hale ÜSTÜNER [(a)], Yakup ÖZKAZANÇ [(b)]

[(a)] Havelsan A.Ş., Simülasyon ve Eğitim Sistemleri Grup Başkanlığı, 06520 Ankara, hustuner@havelsan.com.tr

[(b)] Hacettepe Üniversitesi, Elektrik ve Elektronik Müh. Bölümü, 06800 Beytepe Ankara, yakup@ee.hacettepe.edu.tr

ÖZ

Bu bildiride, radar arayıcı başlıkları ve radar füze güdümünün yapısı incelenmiş ve yarı-aktif füze arayıcılar için geliştirilmiş bir simulasyon modeli anlatılmıştır. Yarı-aktif füze arayıcı sisteminde, hedefi tespit edebilmek için hedeften yansıyan sinyallerin Doppler frekans kayması aranmıştır. Hedef sinyalinin frekansı takip edilmiş ve açı izleme anten modelleri kullanılarak elde edilen sinyallere göre güdüm komutları hesaplanmıştır. Açı izleme anteni olarak, evre karşılaştırmalı monopulse anten yapıları öngörülmüştür. Yarı-aktif füze sisteminin modellenmesinde oransal seyir güdüm mantığı kullanılmıştır.

Anahtar Kelimeler: Radar Arayıcı Başlıklar, Radar Füze Güdümü, Yarı-Aktif Füze Sistemleri

ABSTRACT

In this paper, generic structure of radar seeker heads and radar missile guidance have been investigated and a simulation model for semiactive missile seeker has been presented. For target acquisition in semiactive missile seeker system, Doppler frequency shift of the target reflected signals are searched. Target is tracked in Doppler and guidance commands are calculated according to the target reflected signals which are obtained by angle tracking antennas. Phase comparison monopulse antenna models are used as angle tracking antennas. For semi-active missile guidance model, proportional navigation guidance is used.

Keywords: Radar Missile Guidance, Radar Seeker Heads, Semiactive Missile Systems

1. GİRİŞ

Elektronik harp sistemlerinde kullanılan güdümlü füzeler, hedef noktasına varıncaya kadar uçuş süresince yönlendirilebilen füzelerdir. Evleme güdüm (homing guidance) sistemleri hedeflerin yerini ve hızlı görüş hattı değişimlerini uçuş boyunca izleyebildiği için manevralı hedeflere karşı en etkili güdüm sistemidir. Çarpışma noktasına yaklaştıkça daha kesin hedef bilgilerinin elde edilmesi evleme güdüm sistemine ait bir özelliktir.

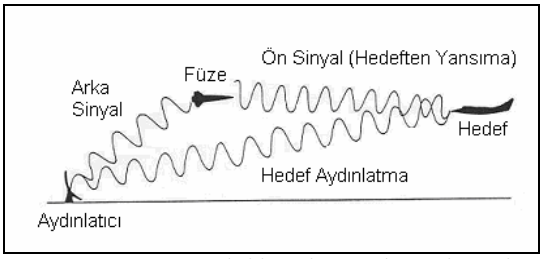
Evleme güdüm sistemlerinde, füzeye yerleştirilmiş olan sensörler yardımıyla hedef takip edilir. Bu sistemler, sensörlerin takip ettiği enerji kaynağına göre üçe ayrılır: aktif, pasif ve yarı-aktif evleme güdüm sistemleri. Yarı-aktif güdüm sistemlerinde füze dışında bir platformda bulunan verici hedefi aydınlatır. Füze arayıcı başlığı, hedeften ve aydınlatıcıdan saçılan sinyalleri kullanır [1].

Bildiride özetlenen çalışmada yarı-aktif radar güdüm sistemleri ve bu sistemlerde kullanılan füze arayıcı başlığının simulasyon modeli Matlab-Simulink yazılımı kullanılarak oluşturulmuştur. Bu simulasyon modeli oluşturulurken yarı-aktif radar güdümü temel alınmıştır. Yarı-aktif radar güdüm modelinde güdüm yöntemi olarak oransal seyir güdümü (proportional navigation guidance) kullanılmıştır. Oransal seyir güdüm yönteminde gerekli komutları hesaplayabilmek için ihtiyaç duyulan yaklaşma hızı ve görüş hattı açısının türevi, izleme radarının doğal çıktıları olduğu için, yarı-aktif radar güdüm sisteminde ayrıca hesaplanmalarına gerek kalmamaktadır.

Tasarlanan yarı-aktif radar güdüm simulasyon modelini doğrulamak için ideal bir oransal seyir güdüm modeli oluşturulmuş ve her iki modelin simulasyon sonuçları karşılaştırılmıştır.

2. YARI-AKTİF RADAR ARAYICILARININ TEMELLERİ

CW yarı-aktif sistemlerin genel yapısı Şekil 1'de gösterilmiştir.



Şekil 1. CW Yarı-Aktif Radar Güdüm Sistemi

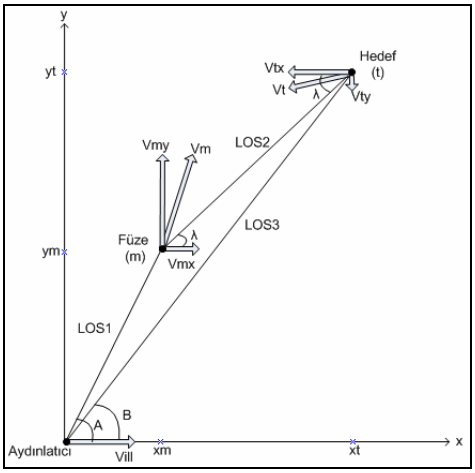
CW yarı-aktif sistemlerde yerde bulunan CW aydınlatıcı, radar huzmesini uçuş boyunca hedef üzerinde tutar. Füze, ön anteniyle hedeften saçılan CW dalgasını, arka anteniyle ise CW aydınlatıcının yan loblarından ışıyan dalgaları alır.

CW füze arayıcısının en temel uygulaması; ön ve arka alıcı, sinyal işleme devresi (speedgate) ve ön anteni kontrol eden bir izleme döngüsünden oluşur. CW füze ayrıca, füzenin güdümlenmesini ve dengede tutulmasını sağlayan otopilota, savaş başlığını optimum zamanda infilak ettirecek tapaya ve elektriksel/hidrolik güç kaynağına sahiptir [2], [3], [4], [5].

Füze, ön ve arka antenleriyle aldığı CW sinyalleri lokal osilatör kullanarak ara frekans (Intermediate Frequency-IF) katına çeker. Ara frekans katındaki ön ve arka anten sinyallerini karıştırıcıdan geçirerek Doppler sinyalini elde eder. Doppler sinyalinin istenen hedefe ait olup olmadığına Doppler frekans arama döngüsünde karar verilir. Doppler sinyalinin aranılan hedefe ait olduğuna karar verilirse füze hedef takibine başlar. Eğer sinyal hedefe ait değilse açı izleme antenleri olası bir hedefi aramaya devam eder [2], [3].

3.YARI AKTİF RADAR GÜDÜMÜ SİMULASYON MODELİ

Yarı-aktif füze arayıcılarında kullanılan ve füze-hedef-aydınlatıcı arasındaki Doppler frekans ilişkisini gösteren geometri Şekil 2'de gösterilmiştir.



Şekil 2. Yarı-Aktif Radar Güdüm Sistemi Geometrisi

Burada, füze konum vektörü (x_m, y_m), hedef konum vektörü (x_t, y_t), füze hız vektörü (V_{mx}, V_{my}), hedef hız vektörü (V_{tx}, V_{ty}), aydınlatıcı hız vektörü V_{ill}, füze-hedef arasındaki görüş hattı (LOS2) açısı λ, aydınlatıcı-füze arasındaki görüş hattı (LOS1) açısı A, aydınlatıcı-hedef arasındaki görüş hattı (LOS3) açısı B olarak gösterilmiştir.

CW aydınlatıcı f_0 frekansında sinyal yaymaktadır. Aydınlatıcının ana veya yan loblarından füzenin arka antenine ulaşan CW dalganın frekansı f_0 'dan f_{rear} kadar sapar. CW aydınlatıcının hedefe gönderdiği f_0 frekansındaki CW dalga ise, f_0 'dan f_{front} kadar saparak füzenin ön antenine yansır [1], [5].

Füzenin arka (referans) antenine gelen sinyalin Doppler frekans kayması (f_{rear}), füze ile aydınlatıcı yaklaşma hızının bir fonksiyonudur. f_{rear} ;

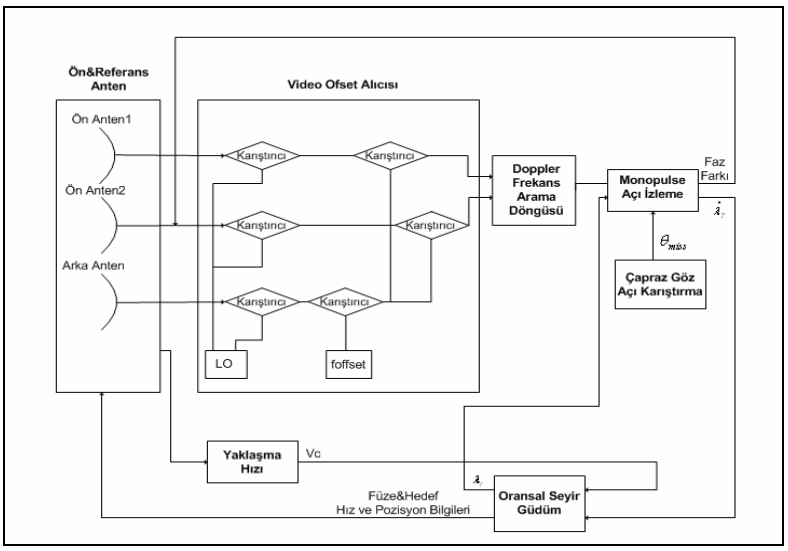
$$f_{rear} = \frac{f_0}{c}\left(v_{ill}\cos\left(\tan^{-1}\frac{y_m}{x_m}\right) - v_{mx}\cos\left(\tan^{-1}\frac{y_m}{x_m}\right) - v_{my}\sin\left(\tan^{-1}\frac{y_m}{x_m}\right)\right) \quad (1)$$

olarak elde edilir.

Füzenin ön anteni ile alınan sinyalin Doppler frekans kayması (f_{front}), füzenin hedef ile aydınlatıcı yaklaşma hızının bir fonksiyonudur. f_{front} ;

$$f_{front} = \frac{f_0}{c}\left(v_{ill}\cos B + v_{tx}\cos B + v_{ty}\sin B + v_{tx}\cos\lambda + v_{ty}\sin\lambda + v_{mx}\cos\lambda + v_{my}\sin\lambda\right) \quad (2)$$

olarak hesaplanır.
Yarı-aktif radar güdüm simulasyon modeli Şekil 3'te gösterildiği gibi tasarlanmıştır.



Şekil 3. Yarı-Aktif Radar Güdümü Blok Diyagramı

Yarı-aktif radar güdüm simulasyon modeli altı alt modelden oluşmaktadır: ön ve referans kanal modeli, video ofset almaç modeli, doppler frekans arama döngüsü modeli, monopulse açı izleme modeli, yaklaşma hızı modeli ve oransal seyir güdüm modeli [5].

Ön ve referans kanal alt modelinde yarı-aktif radar güdümlü füze arayıcısının takip ettiği hedef Doppler frekansı hesaplanır. Yarı-aktif radar güdümlü füzenin ön anteniyle aldığı hedef eko sinyali ve arka anteniyle aldığı aydınlatıcı sinyali elde edilir.

Ön anten olarak monopulse faz karşılaştırmalı antenler temel alınmış ve bu antenlerin çıktıları $Front_Ant1$ ve $Front_Ant2$ olarak tanımlanmıştır. Bu sinyallerin genlikleri ve frekansları aynı, fazları ise farklıdır.

$Front_Ant1$ ve $Front_Ant2$ sinyalleri:

$$\begin{aligned} Front_Ant1 &= \cos(2\pi(f_0 + f_{front})t + \phi) \\ Front_Ant2 &= \cos(2\pi(f_0 + f_{front})t) \end{aligned} \tag{3}$$

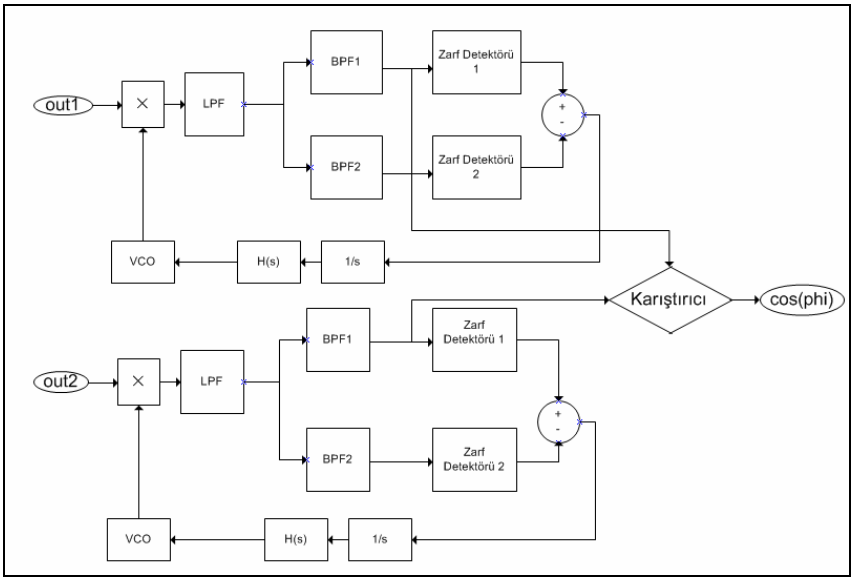
Füzenin arka anteniyle alınan sinyali $Rear_Ant$ ise:

$$Rear_Ant = \cos(2\pi(f_0 + f_{rear})t) \tag{4}$$

yapısındadır.

Ofset video almaç alt modelinde, aralarında monopulse faz farkı olan iki ofsetlenmiş Doppler sinyali elde edilir. Ön ve referans kanal alt modelinde elde edilen ön anten ve referans anten sinyalleri ara frekans (IF) katına indirilir ve referans anten sinyali ofsetlenir.

Doppler frekans arama döngüsü alt modelinde, ofsetlenmiş Doppler sinyalleri (out_1 ve out_2) Doppler frekansı arama filtrelerinde taranır. Bu sinyaller, Doppler frekansı arama filtrelerinde tespit edildiğinde aralarındaki monopulse faz farkı ölçülür. Doppler frekans arama döngüsünün blok diyagramı Şekil 4'te gösterilmiştir [1].



Şekil 4. Doppler Frekansı Arama Döngüsü Blok Diyagramı

Ofsetlenmiş Doppler sinyalleri out_1 ve out_2 sinyalleri ikiye ayrılarak dar bantlı Doppler filtrelerinden (BPF1 ve BPF2) geçirilir. BPF1 filtresinin frekans aralığı $f_{offset} < f < f_s + f_{doppler}$, BPF2 filtresinin frekans aralığı ise $f_{offset} - f_{doppler} < f < f_{offset}$'dir.

BPF1 ve BPF2 çıktıları zarf detektörlerinden geçirilir ve farkları alınır. Fark sinyalinin integrali alınıp, H(s) transfer fonksiyonundan geçirilir ve hata sinyali bulunur. VCO, bu hata sinyaline göre değişen ve tüm Doppler frekans aralığını tarayan bir frekans üretir. Eğer gelen sinyal hedefe aitse, BPF1 veya BPF2'de tespit edilebilir.

Yarı-aktif radar güdüm simulasyon modelinde açı izleme sistemi olarak faz kıyaslamalı monopulse açı izleme kullanılmıştır. Monopulse açı izleme alt modelinde, hedef eko sinyallerinin monopulse anten referans eksenine geliş açısı kullanılarak eko sinyalleri arasındaki faz farkı hesaplanır [5].

Oransal seyir güdüm alt modelinde füzeye uygulanan yanal ivme, füze ve hedefe ait hız ve konum bilgileri hesaplanmıştır. Hesaplamalarda kullanılan yaklaşma hızı ve görüş hattı açısının değişim hızı, yarı-aktif radar güdümlü arayıcı alt modellerinden alınmıştır.

İdeal oransal seyir güdüm modelinde ise ideal bir arayıcıya sahip füzenin güdümü simule edilmiştir. Güdüm komutları doğrudan füze ve hedef kinematiğinden elde edilmiştir.

Simulasyonlar

Bu bölümde üç farklı senaryo için ideal oransal seyir güdüm modelinin ve yarı-aktif radar güdümü modelinin simulasyon sonuçları incelenmiştir.

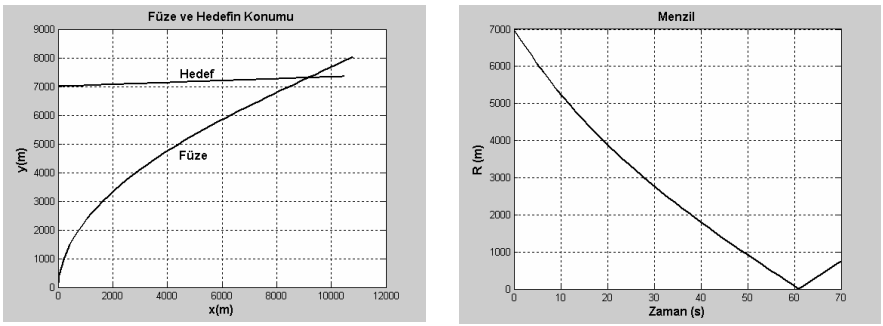
İdeal oransal seyir güdüm modelinde kullanılan füze ve hedef bilgileri, yarı-aktif radar güdümü modeline de uygulanmıştır. İdeal oransal seyir güdüm modelinin ve yarı-aktif radar güdümü modelinin sonuçları karşılaştırılarak, yarı-aktif radar güdümlü füze arayıcı modelinin başarısı değerlendirilmiştir [5].

Senaryo 1

Füze Parametreleri: $x_m(0) = 1m$, $y_m(0) = 1m$, $V_{mx}(0) = 5m/s$, $V_{my}(0) = 200m/s$

Hedef Parametreleri: $x_t(0) = 2m$, $y_t(0) = 7000m$, $V_{tx}(0) = 150m/s$, $V_{ty}(0) = 5m/s$, $a_{tx}(0) = 0m^2/s$, $a_{ty}(0) = 0m^2/s$

İdeal oransal seyir güdüm modelinin simulasyon sonuçları Şekil 5'de gösterilmiştir.

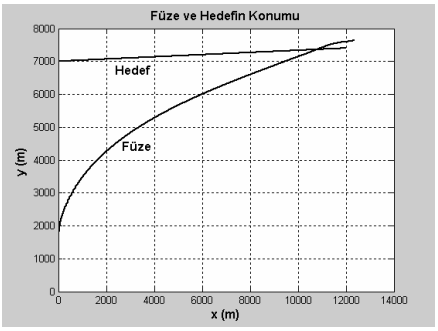


(a) (b)

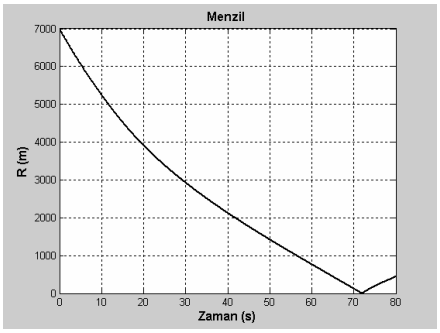
Şekil 5. Senaryo 1: İdeal Oransal Seyir Güdüm Modeli Sonuçları

Şekil 5-a'da füzenin ve hedefin yatay ve dikey eksenlerdeki konumu gösterilmiştir. Şekil 5-b'de gösterilen çizim, füze ile hedef arasındaki menzilin zamana göre değişimidir. Menzil, t=61'de minimum değerine (1.95m) ulaşmaktadır.

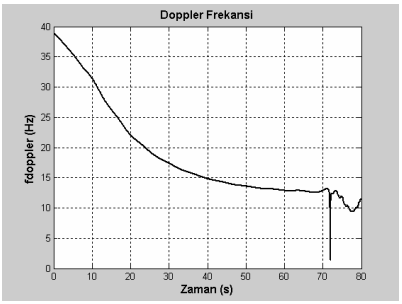
Yarı-aktif radar güdüm modelinin simulasyon sonuçları Şekil 6'da gösterilmiştir.



(a)



(b)



(c)

Şekil 6. Senaryo 1: Yarı-Aktif Radar Güdüm Modeli Sonuçları

Şekil 6-(b)'de gösterilen füze ile hedef arasındaki menzil (R), t=72s'de minimum değerine (6.2m) ulaşmaktadır. Hedef ve füze eğrilerinin kesişmesi ve ıskalama mesafesinin çok küçük olması füzenin hedefi vurduğu anlamına gelmektedir.

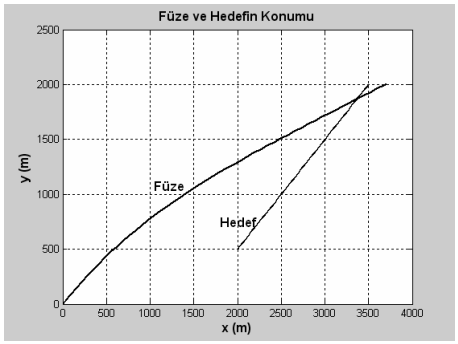
Şekil 6-(c)'de gösterilen Doppler frekansı t=0 ve t=72s aralığında 39Hz'den 13Hz'e düşmüştür. Bu hızlı değişim, Doppler frekans arama döngüsü alt modelinin, hedef Doppler frekansını yaklaşık 10s sonra tespit edebilmesine neden olmuştur.

Senaryo 2

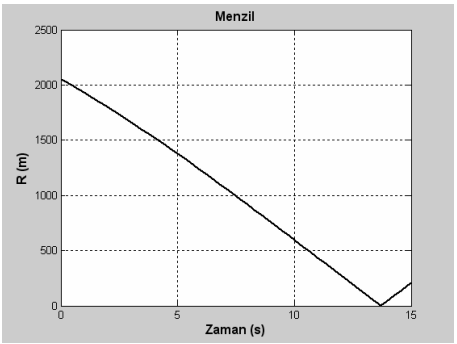
Füze Parametreleri: $x_m(0) = 5m$, $y_m(0) = 5m$, $V_{mx}(0) = 200m/s$, $V_{my}(0) = 200m/s$

Hedef Parametreleri: $x_t(0) = 2000m$, $y_t(0) = 500m$, $V_{tx}(0) = 100m/s$, $V_{ty}(0) = 100m/s$, $a_{tx}(0) = 0m^2/s$, $a_{ty}(0) = 0m^2/s$

İdeal oransal seyir güdüm modelinin simulasyon sonuçları Şekil 7'de gösterilmiştir.



(a)

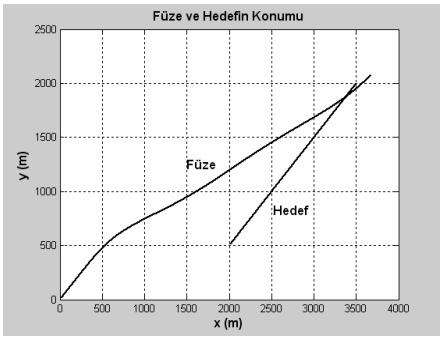


(b)

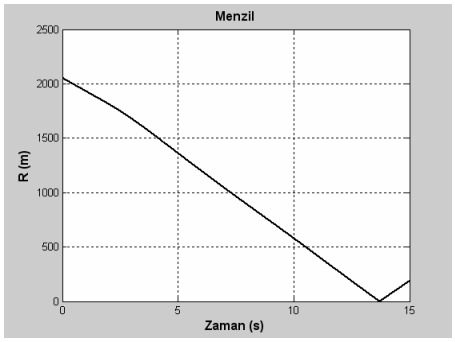
Şekil 7. Senaryo 2: İdeal Oransal Seyir Güdüm Modeli Sonuçları

Şekil 7-b'de gösterilen füze ile hedef arasındaki menzil, minimum değerine (0.2258m) t=13.7s'de ulaşmaktadır. Füze hedefi vurmuştur.

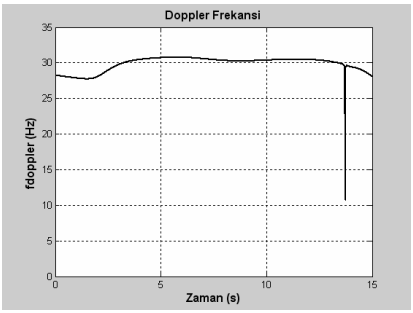
Yarı-aktif radar güdüm modelinin simulasyon sonuçları Şekil 8'de gösterilmiştir.



(a)



(b)



(c)

Şekil 8. Senaryo 2: Yarı-Aktif Radar Güdüm Modeli Sonuçları

Şekil 8-(b)'de gösterilen füze ile hedef arasındaki menzil, t=13.72s'de minimum değerine (0.892m) ulaşmaktadır. Füze hedefi vurmuştur.

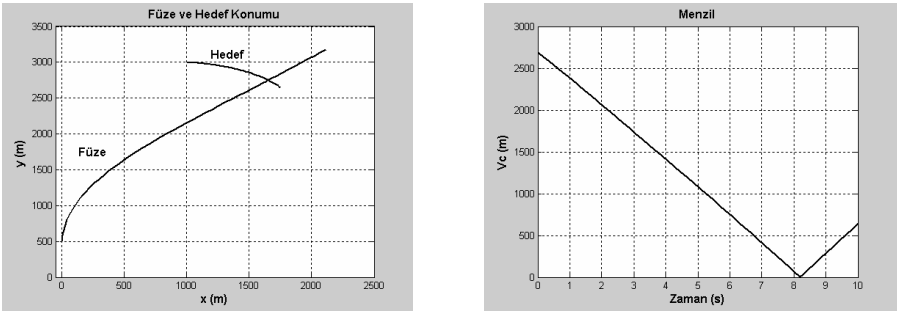
Şekil 8-(c)'de gösterilen Doppler frekansı t=0 ve t=5s aralığında 28.3Hz'den 31Hz'e artmıştır ve çarpışma anına kadar 31Hz'de kalmıştır. Doppler frekans değişimi çok az olduğu için, Doppler frekansı arama döngüsü hedef Doppler sinyalini 0.25s sonunda tespit etmiştir.

Senaryo 3

Füze Parametreleri: $x_m(0) = 5m$, $y_m(0) = 500m$, $V_{mx}(0) = 1m/s$, $V_{my}(0) = 350m/s$

Hedef Parametreleri: $x_t(0) = 1000m$, $y_t(0) = 3000m$, $V_{tx}(0) = 100m/s$, $V_{ty}(0) = -10m/s$, $a_{tx}(0) = -5m^2/s$, $a_{ty}(0) = -5m^2/s$

İdeal oransal seyir güdüm modelinin simulasyon sonuçları Şekil 9'da gösterilmiştir.

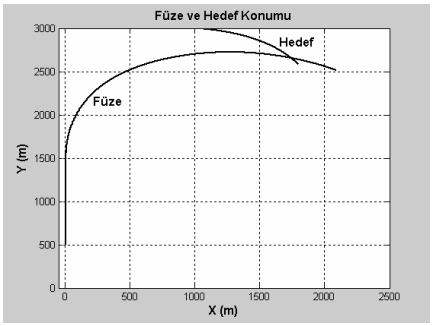


(a) (b)

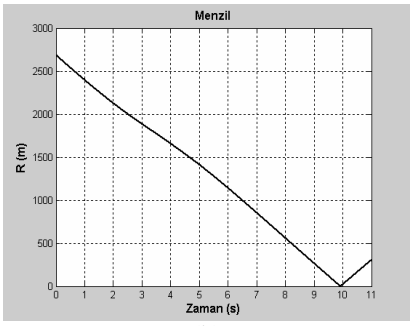
Şekil 9. Senaryo 3: İdeal Oransal Seyir Güdüm Modeli Sonuçları

Şekil 9-b'de gösterilen menzil, t=8.2s'de minimum değerine (0.09m) ulaşmaktadır.

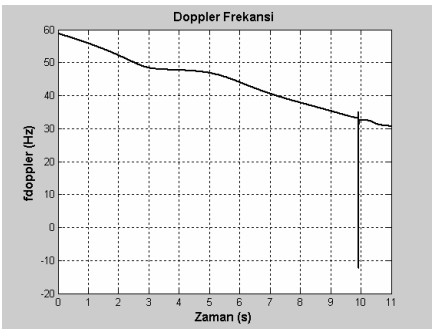
Yarı-aktif radar güdüm modelinin simulasyon sonuçları Şekil 10'da gösterilmiştir.



(a)



(b)



(c)

Şekil 10. Senaryo 3: Yarı-Aktif Radar Güdüm Modeli Sonuçları

Şekil 10-b'de gösterilen füze ile hedef arasındaki menzil, t=9.93s'de minimum değerine (0.24m) ulaşmaktadır. Şekil 10-c'de gösterilen Doppler frekansı çarpışma zamanına kadar 58.8Hz'den 33Hz'e azalmıştır. Doppler frekansı uçuş süresince hızla değiştiği için, Doppler frekans arama döngüsü hedef Doppler sinyalini yakalamakta gecikmiştir.

Üç senaryonun simulasyonu sonucunda ideal oransal seyir güdüm modelinin ve yarı-aktif radar güdümü modelinin hedefi vurduğu görülmektedir. Yarı-aktif radar güdümü modelinin ve ideal oransal seyir güdüm modelinin "Füze ve Hedefin Konumu" grafiklerinde; füze-hedef çarpışma zamanı, Doppler sinyalinin tespit edilme hızına bağlı olarak farklılık göstermiştir.

Gerçekleştirilen senaryolar içinde en hızlı Doppler frekans değişimi birinci senaryoda meydana gelmiştir. Bu senaryoda ilk 10s yarı-aktif radar güdüm modeli, Doppler frekansını tespit edememiş ve füzeyi hedefe yönlendirecek güdüm komutları hesaplanamamıştır. Füze, 10. saniyeden sonra hedefe yönelebilmiştir.

İkinci senaryoda Doppler frekansı çok az değiştiği için arama döngüsü hedef Doppler sinyalini 0.25s içinde tespit edebilmiştir.

Manevralı hedefin kullanıldığı üçüncü senaryoda yarı-aktif radar güdüm modeli Doppler frekansını geç tespit ettiği için, füzenin hedefi yakalayacak manevrayı yapmasını sağlayacak güdüm komutlarının hesaplanması da gecikmiştir. Doppler frekansı tespit edildikten sonra, füzenin hedefi yakalayabilmek için yaptığı manevralar ideal seyir güdüm modelinin manevralarına göre daha keskin ve değişkendir.

4. SONUÇ

Bu çalışmada yarı-aktif radar güdümü ve ideal oransal seyir güdümü modellenmiş ve modellere üç farklı senaryo uygulanmıştır. Simulasyonların sonucunda yarı-aktif radar güdümlü füzenin hedefi vurduğu gözlenmiştir.

Doppler frekansındaki değişimlerin eşzamanlı olarak takip edilebilmesi yarı-aktif radar güdümü modelinin en önemli unsurlarından biridir. Bu nedenle alt modeller arasında en hızlı çalışan blok Doppler frekans arama döngüsü olmalıdır.

Manevrasız hedeflerin kullanıldığı senaryolarda hedef Doppler frekansının geç tespit edilmesi sadece çarpışma zamanın uzamasına neden olmuştur. Doppler frekansı tespit edildikten sonra, füzenin hedefe yönelebilmek için yaptığı manevralar ile ideal seyir güdüm modelinin manevraları benzerdir. Füzenin dönme hızı, hedefe yaklaştıkça azalmıştır.

Manevralı hedefin kullanıldığı üçüncü senaryoda Doppler frekansı uçuş süresince hızlı değişim gösterdiği için yarı-aktif radar güdüm modeli hedef Doppler sinyalini yakalamakta gecikmiştir.

İleriye yönelik olarak Doppler frekansı arama döngüsünü hızlandıracak modeller geliştirilebilir. Açı izleme sistemi için genlik karşılaştırmalı monopulse açı izleme veya konik taramalı açı izleme modellenebilir.

KAYNAKÇA

[1] Özkazanç, Y.,2003, ELE781 Seyir, Güdüm ve Kontrol Ders Notları, Hacettepe Üniversitesi

[2] Ivanov, A., 1983, Semi-active radar guidance, Microwave Journal, 26, 105-120

[3] Ivanov, A., 1990, Radar Guidance Of Missiles, Radar Handbook, McGraw Hill

[4] Neri, F., 2001, Introduction to Electronic Defense Systems, Artech House

[5] Üstüner, H., 2006, Radar Arayıcı Başlıklar ve Radar Güdümünün Modellenmesi ve Simulasyonu, Yüksek Lisans Tezi, Hacettepe Üniversitesi

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

RADAR KESİT ALANI (RKA) MODELLEME VE SİMÜLASYONU

Gonca ÇAKIR [(a)], Mustafa ÇAKIR [(b)], Levent SEVGİ [(c)]

[(a)] Kocaeli Üniversitesi, Mühendislik Fakültesi, gonca@kou.edu.tr

[(b)] Kocaeli Üniversitesi, Sivil Havacılık Yüksek Okulu, mcakir@kou.edu.tr

[(c)] Doğuş Üniversitesi, Mühendislik Fakültesi, Kadıköy-İstanbul, lsevgi@dogus.edu.tr

ÖZ

Ulusal savunma 24-saat, kesintisiz, tümleşik gözetlemeyle başlar. Hava sahanlığımızın, kıyılarımızın, ve sınır boylarının kesintisiz gözetlenmesi her an bir saldırı olasılığına karşı uyanık ve tetikte durmanın ön koşuludur. Sağlıklı ve güvenilir gözetleme amacıyla mevcut her türlü algılayıcının tümleşik kullanılması çok algılayıcılı tümleşik gözetleme sistemlerini ortaya çıkarmıştır. Özellikle radar olarak isimlendirilen elektromanyetik algılayıcıların performansları elektromanyetik dalgaların – radar hedefleriyle etkileşiminin fiziksel olarak çok iyi anlaşılmasına, matematiksel olarak modellenmesine, ve bilgisayar simülasyonlarıyla da incelenmesine bağlıdır. Radar kesit alanı (RKA) olarak isimlendirilen bu etkileşim etkili algılayıcı tasarımı, güvenilir tümleşik gözetleme sistemlerinin kurulumu, hatta uçak, gemi, tank gibi hedeflerin yapımına varıncaya kadar bir çok hususu etkiler. Bu çalışmada RKA modelleme ve simülasyon çalışmaları karakteristik örneklerle özetlenecektir.

Anahtar Kelimeler: FDTD, Modelleme, MoM, Radar Kesit Alanı, RKA, Simülasyon.

RADAR CROSS-SECTION MODELLING AND SIMULATION

ABSTRACT

National defense starts with 24-hour, continuous integrated surveillance. Continuous surveillance of land, sea, and air is the first step of being ready and alert. Reliable surveillance brings the concept of multi-sensor integrated surveillance systems. Radars, as electromagnetic sensors, use the information extracted from *electromagnetic wave – target interaction*. Radar performance, initially, depends on physical understanding of this interaction, mathematical modeling and computer simulations. This interaction, called as radar cross-section (RCS) is important not only in the design of integrated surveillance systems, but also for low-RCS targets, such as aircrafts, ships, and land vehicles. RCS modeling and simulation studies will be discussed in this presentation.

Keywords: FDTD, Modeling, MoM, Radar Cross-Section, RCS, Simulation.

1. GİRİŞ

Günümüzde ulusal savunma dendiğinde akla öncelikle sınırlarımızın denizden, karadan, havadan, ve denizaltından 24-saat kesintisiz gözetlenmesi ve kontrol edilmesi gelmektedir. Çok algılayıcılı tümleşik gözetleme, komuta/kontrol ve atış destek sistemleri bu kavramın birer parçalarıdır. Söz konusu sistemler ARGE'ye dayalı tedarik ve sistem entegrasyonu temelinde yürütülmesi zorunlu olan sistemlerdir.

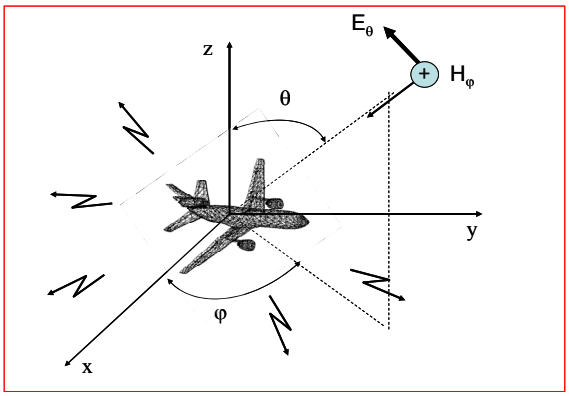
Elektronik gözetleme sistemlerinin temel taşı amaca uygun seçilecek olan radarlardır. Özellikle bilgisayar teknolojilerindeki gelişmelere paralel olarak güçlü sayısal işaret işleme tekniklerinin uygulanabilir olmasıyla yüksek frekans (YF) ve Mikrodalga (MD) radarları öne çıkmaktadır. Hangi tip ve frekans bölgesinde olursa olsun bir radarın performansı elektromanyetik dalgalarla nesnelerin etkileşiminin fiziksel olarak çok iyi anlaşılmasına, matematiksel olarak modellenebilmesine bağlıdır. Günümüzde bunlara ek olarak güçlü bilgisayar simülasyonlarına da gerek duyulmaktadır. Bilgisayar simülasyonlarının kullanıldığı alanların başında da Radar Kesit Alanı (RKA) modelleme gelmektedir.

Elektromanyetik dalgalarla hedeflerin etkileşiminin bir göstergesi olan RKA

$$\sigma = \lim_{R \to \infty} \left\{ 4\pi R^2 \frac{|E_s|^2}{|E_g|^2} \right\} ; \quad \left(E_{g,s} = E_\theta \quad veya \quad E_\phi \right) \tag{1}$$

şeklinde tanımlanmaktadır. Burada R uzaklık, E_g ve E_s ise gelen ve saçılan alanları göstermektedir. Metre-kare boyutunda olan ve uzak alan etkilerini içeren RKA matematiksel hesaplama, açık alan ya da yansımasız oda ölçmeleri ya da bilgisayar simülasyonlarıyla elde edilir. Bir hedef için RKA gelen elektromanyetik dalgaların polarizasyonuna, radarın mono- ya da bi-statik çalışmasına, geliş ve saçılma açılarına, frekansa, ve hedef geometrisine bağlıdır.

Tipik bir RKA ölçme ya da simülasyon senaryosu Şekil 1'de resmedilmiştir.



Şekil 1. RKA modelleme senaryosu

RKA bir uzak alan etkisi olduğundan gelen ve saçılan alanlara ait yönler θ ve φ açılarıyla belirlenir. Kartezyen koordinatlarda orijinde bulunan vir hedef için θ düşey eksenle yapılan açıyı, φ ise yatay düzlemde x-ekseninden ölçülen açıyı gösterir. Ancak, bu açılar RKA terminolojisine göre tanımlanmıştır; yani gelen ve saçılan alanlar için farklıdır. Gelen açı, geliş doğrultusunun açısıdır (propagasyon vektörünün açısı değil), saçılan alanlara ait açılar ise propagasyon vektörünün gösterdiği açılardır. Bu durumda gelen ve saçılan alanlara ait aynı açılar verildiğinde gelen dalgaların aynı doğrultuda geri saçılmasından söz ediliyor demektir. Şekil 1'deki gelen dalga θ-polarizasyonludur, bu nedenle E_θ ve H_φ bileşenleri vardır.

Elektromanyetik dalga – hedef etkileşimi radar işaret frekansına ve hedef geometrisine bağlıdır. Hedefin en uzun boyutuna *l*, işaret dalga boyuna da λ denirse, RKA için üç tipik ve birbirinden farklı bölge söz konusudur: [2-4];

- Düşük frekanslar (*Rayleigh RKA bölgesi*): Bu bölgede hedef boyutları (l) işaret dalga boyuna göre çok küçüktür ($l << \lambda$) ve hedef bir nokta kaynak gibi görülür. Bu bölgede RKA katkısı frekansın dördüncü kuvvetiyle orantılı olmasına karşın ($\sigma \approx f^4$), hedef noktasal kaynak gibi davrandığından RKA değeri ihmal edilebilecek düzeydedir.
- Orta frekanslar (*rezonans RKA bölgesi*): Bu bölgede hedef boyutları ile işaret dalga boyu aynı mertebededir ($l \approx \lambda$) ve hedef bir bütün olarak RKA değerine katkı yapar. Bu nedenle, matematiksel modellerin uygulanabilmesi hemen hemen olanaksızdır. Ancak, bu rezonans RKA bölgesinde işe yarayan güçlü sayısal modeller bulunmaktadır. YF yüzey dalga radarlarının hedeflerinin RKA davranışları bu rezonans bölgesindedir.
- Yüksek frekanslar (*optik RKA bölgesi*): Bu bölgede ise hedef boyutları işaret dalga boyuna göre çok büyüktür ($l >> \lambda$), bu yüzden hedefin her bölgesi ayrı hedeflermiş gibi ele alınabilir. Yüksek frekans yaklaşıklıkları dediğimiz ve geometrik optik (GO), fizik optik (PO), geometrik kırınım kuramı (GTD), birbiçim kırınım kuramı

(UTD) ve fiziksel kırınım kuramı (PTD) olarak isimlendirilen matematiksel yöntemlerle hedefin RKA davranışları modellenebilmektedir. Mikrodalga radarlarına ait hedeflerin RKA davranışları optik RKA bölgesine denk gelmektedir.

Bu sunumda, RKA modelleme ve simülasyon çalışmaları ele alınacaktır. Rezonans RKA bölgesinde kullanılabilecek olan FDTD (*Finite-Difference Time-Domain*) [5-7] ve MoM (*Method of Moments*) [8] yöntemleri ve simülatörleri yanında optik RKA bölgesinde kullanılan yüksek frekans yaklaşıklıkları özetlenecektir. Özellikle FDTD ve MoM gibi yöntemlerin en önemli zorluğu istenen hedefe ait ayrık bilgisayar modellerinin elde edilebilmesidir. Bu sunumda gerçekçi yer, hava, ve su-üstü hedeflerinin üç boyutlu (3D) bilgisayar grafiklerinden ayrık modellerinin otomatik olarak oluşturulması gösterilecektir. Bu modeller kullanılarak kendi geliştirdiğimiz 3D FDTD simülatörü [5] ile sayısal RKA hesaplarına örnekler sunulacaktır. Bu RKA sonuçları MoM yöntemini kullanan ve dolaşımı serbest olan açık-kaynak NEC (*Numerical Electromagnetic Code*) paketiyle karşılaştırılacaktır.

2. FDTD ve MoM RKA SİMÜLATÖRLERİ

Kendi geliştirdiğimiz [5] ve kullanmakta olduğumuz FDTD-RKA simülatörü üç ana modüle sahiptir. Zaman düzleminde Maxwell denklemlerinin ayrılaştırılmasıyla elde edilen iteratif denklemleri kullanarak ayrık hedef civarında yakın elektromanyetik dalgaların simülasyonunu yapan üç boyutlu FDTD, ve eşdeğerlilik prensibini kullanarak uzak alanları hesaplayan YAKUZ ilk iki ana modülü oluşturur. Üçüncü modül ise zaman simülasyonları bitiminde kaydedilen veriden ayrık Fourier dönüşümü (AFD) ile (1) denkleminden RKA davranışını hesaplayan modüldür. Bu sunumda FDTD ve MoM yöntemlerinin ayrıntılarına girilmeyecektir; ilgi duyanlar kaynaklara bakabilir. Onun yerine, doğrudan simülasyonların uygulanması ve karakteristik örnekler ele alınacaktır.

FDTD-RKA simülatörü ile RKA modelleme ve hesaplama şu şekilde gerçeklenir:

1. Ayrık hesap uzayının ortasına yerleştirilen hedef seçilen yönden bir düzlem dalga ile aydınlatılır. Gönderilen düzlem dalga zamanda bir Gauss darbesidir. Bu dalganın hedefe çarpması sonucu hedef üzerinde oluşan yüzey akımlarının yarattığı yakın saçılan alanlar FDTD denklemleriyle elde edilir.
2. Ayrık her zaman anında hedeften saçılan ve hedef civarında geçici süre var olan bu yakın alanların zaman davranışları kullanılarak küresel koordinatlarda uzak alanlar (E_θ ve E_ϕ) elde edilir ve kaydedilir. Bu kaydetme işlemi hedef etrafında tek bir yönde olabileceği gibi, yatay ya da düşey bir düzlemde hedefin etrafında belli açısal aralıklarla ve tümüyle de yapılabilir. Böylece hedefin seçilen düzlemde açısal RKA davranışına ait zaman bilgisi kaydedilir.
3. Kaydedilen uzak alanlar kullanılarak verilen bir frekans için DFT (*Discrete Fourier Transformation*) modülü yardımıyla ve (1) denklemi kullanılarak açısal RKA değişimleri elde edilir. FDTD yöntemi darbesel işaretlerin davranışlarının simülasyonuna izin verdiğinden bir simülasyon ile oldukça geniş bantlı RKA davranışları elde edilir.

Herhangi bir polarizasyon için bu simülasyonlar yapılabilir olsa da, pratikte, sadece düşey ya da yatay polarizasyon için RKA bilgisi istenir. Bu nedenle $\sigma_{\theta\theta}$ ve $\sigma_{\varphi\varphi}$ diye adlandırılan eş-polarizasyon ya da $\sigma_{\theta\varphi}$ ve $\sigma_{\varphi\theta}$ diye adlandırılan çapraz-polarizasyon durumları söz konusudur.

FDTD yönteminde bir hedefin milyonlarca (ve dalga boyuna göre) küçük kübik hücreden oluştuğu varsayılır. Yani, hedefin kübik hücresel modeli kullanılır. Oysa, MoM yönteminde ise hedefin dalga boyuna göre küçük tel parçalarından oluşmuş ızgara modeli kullanılır [8]. MoM yönteminin bir uygulaması NEC'tir [9]. NEC, açık-kaynak denen ve herkesin kullanabileceği, kodlarını geliştirebileceği bir pakettir. Kaliforniya'da bulunan Lawrence Livermore Laboratuarında ve Naval Ocean Systems Merkezinin sponsorluğunda geliştirilmiştir. Kullanışlı bir versiyonu NEC2 [10]'dir. NEC2 incelenen hedefe ait ızgara modelinin her bir telinin GW (*Generate Wire*) komutuyla satır satır oluşturulmasını ister. NEC2 hedef dosyası ASCII formatında ve GW komutlarıyla her telin (x_1,y_1,z_1) ve (x_2,y_2,z_2) şeklinde iki ucunun koordinatlarını ve operasyonel komutları içerir.

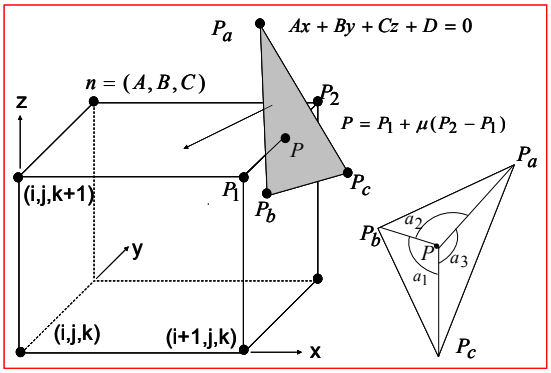
3. AYRIK FDTD VE NEC MODELLERİNİN OLUŞTURULMASI

FDTD ve MoM benzeri yöntemlerin önemli girdisi ayrık hedef modelleridir. Gerçekçi hedefler için böyle ayrık modellerin oluşturulması oldukça zor ve zahmetlidir. Bu amaçla pahalı yazılımlar hatta satın alınabilecek pahalı ayrık modeller bulunmaktadır. Bu sunumda belli bir formatta saklanan grafik dosyasından ayrık FDTD ve NEC modellerinin oluşturulmasına ait bir algoritma anlatılacaktır. Herhangi bir grafik formatı olabilmesine karşın burada 3DS formatı seçilmiştir. Bir 3DS formatlı dosyada üçgenleme yada dörtgenleme bilgisi (köşe noktaları), *kaplama* ya da *materyal* bilgisi, *gölgelendirme* verisi, *kamera* bakış açıları, *ızgara renkleri*, vb., bulunmaktadır. Aşağıda gösterileceği gibi, sadece *köşe noktaları* kullanılarak bir hedefin grafik dosyasından ayrık FDTD ve NEC modellerini oluşturmak olasıdır. Bu çalışmada hedeflerin mükemmel iletken olduğu varsayılmıştır. Ancak, *kaplama* bilgisi kullanılarak hedefin kayıplı malzemelerden oluşmuş modelini çıkarmak da olasıdır.

Grafik dosyasından ayrık FDTD modelinin oluşturulması algoritması şu şekilde çalışır:

- Boyutları L_x, L_y, ve L_z olan dikdörtgensel bir hedef bölgesi ele alınır ve hücre boyutu (Δ) belirlenir.
- Bu hedef bölgesi eşit boyutlu hicrelere bölünür ($N_x=L_x/\Delta$, $N_y=L_y/\Delta$, $N_z=L_z/\Delta$) ve her bir hücre (i,j,k) etiketiyle anılır.
- Her FDTD hücresinin 12-köşesinin $((x_i,y_i,z_i)$, i=1,..12) şeklindeki koordinatları hesaplanır.

- Grafik (3DS) dosyasından bir köşe nokta takımı alınır. Varsayalım ki bu alan bir üçgen olsun. Bu üçgenin (P_a, P_b, ve P_c) şeklinde etiketlenen köşeleri ile ele alınan FDTD hücresi karşılaştırılır (bakınız, Şekil 2).

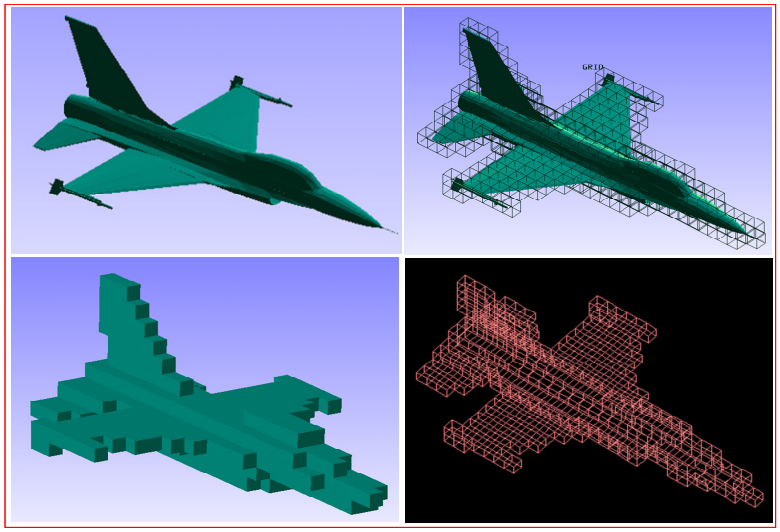


Şekil 2. 3DS verisinden ayrık FDTD hücrelerinin saptanması

Eğer üçgen FDTD hücresinin içinde ise ya da kesişiyorsa bu hücre hedef hücresi olarak etiketlenir ve malzemesi mükemmel iletken yapılır, aksi halde aşağıdaki diğer hususlar kontrol edilir.

- Söz konusu üçgenin (X_{min} - X_{maks}, Y_{min} - Y_{maks}, and Z_{min} - Z_{maks}) işe belirlenen hedef bölgesinin dışında olup olmadığı kontrol edilir, dışında ise bir sonraki kontole geçilir.
- FDTD hücresinin 8-köşesinden 12-kenarına ait doğrular belirlenir ve bu kenarlardan herhangi birinin üçgenin bulunduğu sonsuz düzlem ile kesiştiği nokta bulunur. Bu noktanın üçgenin içinde olup olmadığına bakılır (kesişme noktasında üçgenin a_1, a_2, a_3 açılarının toplamı 360º olmalıdır). Eğer bu sağlanıyorsa hücre yine hedef hücresi olarak etiketlenir.
- Bu işlem bütün FDTD hücreleri ve 3DS dosyasındaki bütün veri için tekrarlanır.

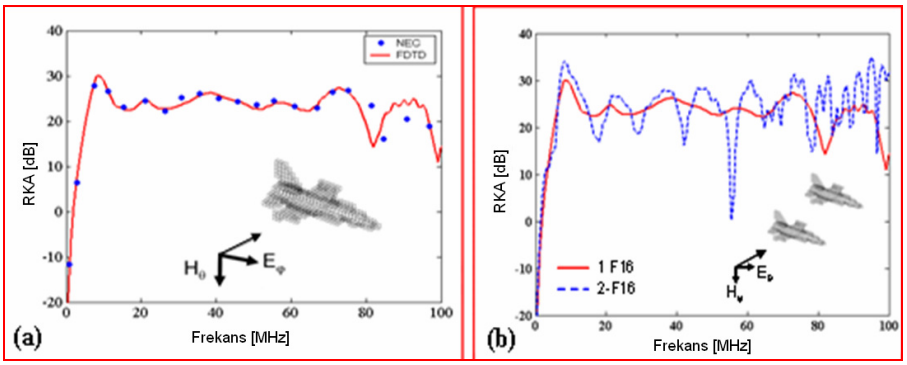
Bir kez FDTD ayrık modeli oluşturulduktan sonra ayrık NEC2 ızgara modelinin oluşturulması oldukça kolaylaşır. Bu amaçla izlene yol şöyle özetlenebilir: Hedefe ait her FDTD hücresinin 8-köşe koordinatları kullanılarak her hücreden GW komutuyla 12-tel üretilir. Bu işlem tamamlandığında geriye yinelenen (ortak) tellerin bulunup yok edilmesi kalır. Ortak teller kaldırıldıktan sonra iç yüzeyler yok edilir. Böylece, hedefin sadece dış yüzeylerinin ızgara bilgisi elde edilmiş olur. Bu işlem sonucu ayrık FDTD ve NEC modelleri aynı yapıya ve boyutlara gelmiş olur. Şekil 3'te bu işlem gösterilmektedir.



Şekil 3. 3DS grafiği, algoritmaya göre ayrıklaştırma ve F16 FDTD-NEC modelleri

4. GERÇEKÇİ HEDEFLERE AİT UYGULAMA ÖRNEKLERİ

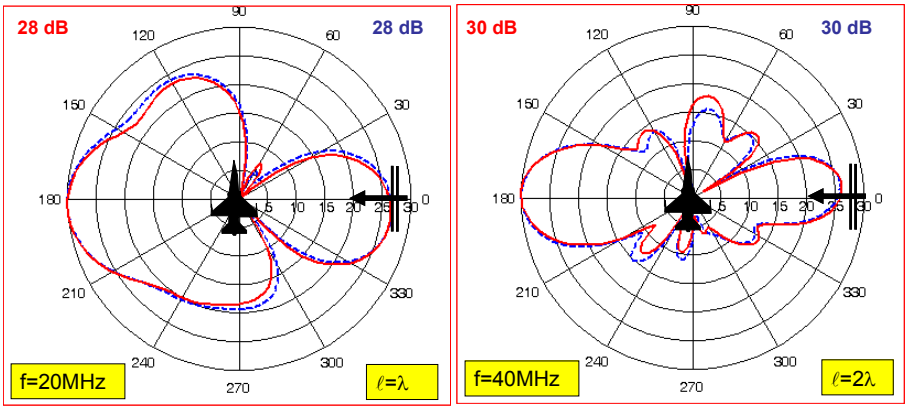
Ayrık model oluşturma ve FDTD ve MoM yaklaşımlarıyla RKA simülasyonlarının yapılmasına ait bir kaç örnek bu bölümde verilmiştir. İlk örnek Şekil 3'te gösterilen ve 15m- uzunluklu F16 savaş uçağına aittir. Ayrık FDTD modeli 488 FDTD yüzey hücresinden oluşmaktadır. FDTD hesap uayının boyutları 49×70×77 hücredir, hücre ve tel uzunlukları Δ=0.5 m seçilmiştir. NEC modelinde buna karşı gelen 1919 tel vardır. Mono-statik durumda bu ayrık F16 hava hedefi için frekansla RKA değişimi Şekil 4'te gösterilmektedir. Şekil 4a, tek bir F16 savaş uçağı için FDTD ve NEC karşılaştırmasına aittir. Biri zaman diğeri frekans domeni yöntemi olan FDTD ve NEC sonuçlarının bu kadar uyumlu çıkması model oluşturma algortimasının ve RKA simülatörlerinin gücünü ortaya koymaktadır. FDTD'de hicre boyutları, NEC'te tel uzunluklarının genelde dalga boyunun onda birinden küçük olması durumunda *güvenilir* sonuç verdiği şekilden açıkça görülmektedir. Hücre boyu (tel uzunluğu) 0.5 m olduğundan bu kriterin sağlandığı en yüksek frekans 60 MHz'dir. Şekilden 60-70 MHz ötesinde iki yöntem sonuçları birbirini tutmamaya başladığı gibi doğrulukları da şüphelidir. Şekil 4b'de ise iki adet 15 m- uzunluklu F16 savaş uçağının 12 m aralıklı olarak yan yana uçması durumunda RKA değişimlerini göstermektedir. Sürekli çizgi tek F16, kesikli çizgi ise iki F16 durumunda FDTD sonuçlarını göstermektedir. Bu örnek, grup halinde uçan savaş uçaklarının belli frekans ve açılarda RCS değerlerinin düşebileceğini belli koşullarda ise artabileceğini söylemektedir. Bunun nedeni hedeflerin yakın alan etkileşimlerinin belirleyici olabilmesidir.



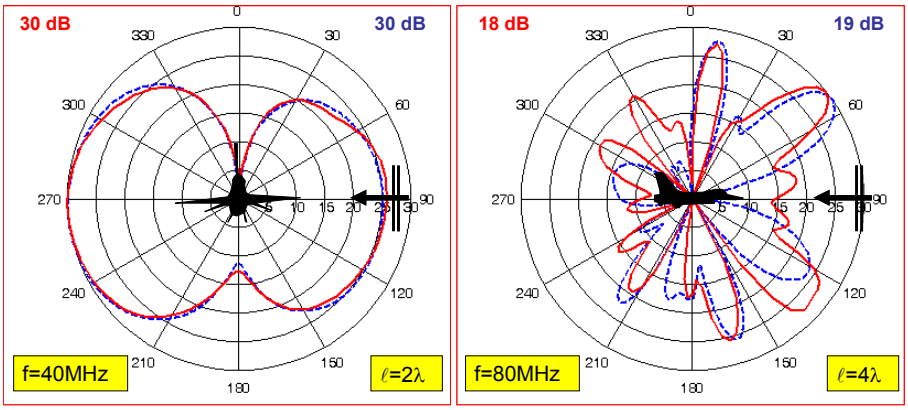
Şekil 4. (a) Tek, (b) İki F16 savaş uçağı için belli bir açıda frekansla RCS değişimi

Şekil 5'te aynı F16 savaş uçağının iki farklı frekansta (20 MHz and 40 MHz) yatay düzlemde açısal bi-statik RKA değişimi gösterilmiştir. Eğriler 30 dB'ye normalize edilmiştir ve gerçek maksimum değerler üstte verilmiştir (FDTD: sol üstte, NEC: sağ üstte; örneğin 20 MHz'deki 28 dB aslında 38 dB'ye karşı düşmektedir). FDTD ve NEC sonuçlarının karşılaştırmalı verildiği bu eğrilerden de görüleceği üzere her iki simülatörün sonuçları birbiriyle uyum içerisindedir. Ayrıca RKA incelemesinde baskın bileşenin her zaman ileriye doğru saçılan alana ait olduğu gözlenmektedir. Unutulmasın ki, RKA tanımı saçılan alanlarla ilgilidir, bu nedenle hedefin ardında saçılan alanın baskın olması gölge bölge denen bu bölgede baskın elektromanyetik alan olacağı anlamına gelmez. Bu baskın saçılan bileşen gelen alanlar zıt fazlı olduğundan hedef ardında saçılan ve gelen alanların toplamı gölge bölgede alanın olmayacağı ya da çok az olacağı sonucunu getirir. Bu da, fiziksel olarak beklenen durumdur.

Şekil 6 ise aynı F16 hedefi için düşey düzlemde ve iki farklı gelen alan doğrultusu için bi-statik açısal RKA değişimlerini göstermektedir ($\sigma_{\varphi\varphi}$ polarizasyon durumu). Açısal değişimler için seçilen parametreler f=40 MHz, θ_i=90°, φ_i=0°, φ_s=0°, 0°≤θ_s≤360°, açısal çözünürlük $\Delta\theta$=3° şeklindedir.

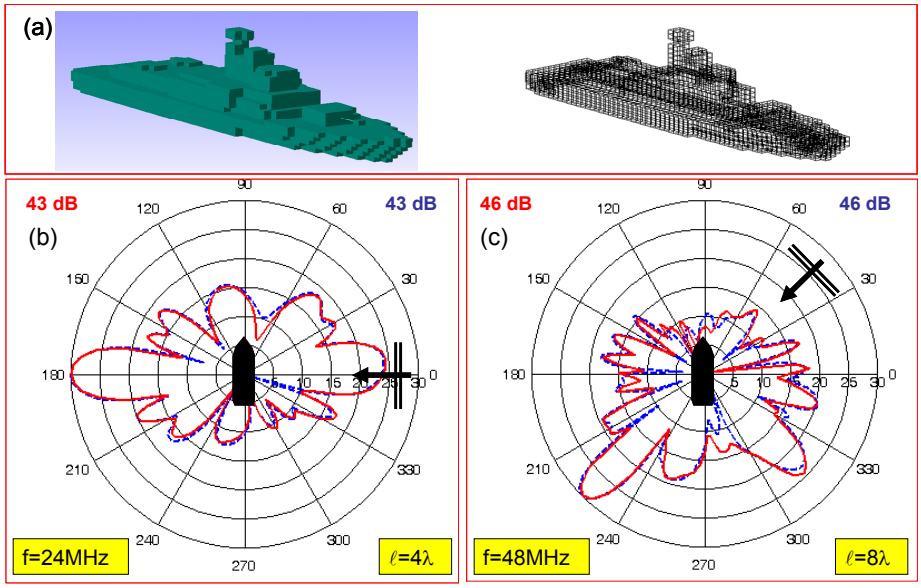


Şekil 5. F16 savaş uçağının yatayda ve $\sigma_{\varphi\varphi}$ durumu için bi-statik RKA değişimi



Şekil 6. F16 savaş uçağının düşeyde ve $\sigma_{\varphi\varphi}$ durumu için bi-statik RKA değişimi

Son örnek bir su-üstü hedefine aittir. Tarantual III isimli ve 50 m-boyundaki geminin ayrık FDTD ve NEC modelleri Şekil 7a'da gösterilmiştir. Toplam 2179 yüzey hücresinden oluşan bu ayrık FDTD modeli 42×93×49 hücrelik bir alan kaplamaktadır. FDTD hesap uzayının 72×123×77 hücre olarak alınması yeterli olmuştur. Hücre boyu ve tel uzunluğu 0.8 m seçilmiştir. Ayrık NEC modelindeki toplam tel sayısı 8059'dur. Şekil 7b'de FDTD ve NEC ile simülasyonları gerçeklenen yatay düzlemde açısal RKA değişimi resmedilmiştir. Burada 24 MHz ve 48 MHz'deki değişimler l=4λ, ve l=8λ durumlarına karşı gelmektedir. İki yöntem sonucunu uyumu yöntemlerin başarısı olduğu kadar oluşturulan ayrık modellerin de birbirine çok yakın olmasıdır. Şekil 7b geminin yandan aydınlatılmasına (θ=90°, φ=90°) aittir. Şekil 7c'de ise gelen alan için açılar θ=90° ve φ=45°'dir.



Şekil 7. Tarantual III için yatayda ve $\sigma_{\varphi\varphi}$ durumu için bi-statik RKA değişimi

5. SONUÇLAR

Ulusal savunmada uçak, gemi yapımında olduğu kadar elektromanyetik algılayıcı tasarımında ve performans testlerinde de nesnelerin RKA davranışlarının bilinmesi kaçınılmazdır. Bu ise elektromanyetik dalgalarla nesnelerin etkileşiminin fiziksel olarak çok iyi anlaşılmasına, matematiksel ve sayısal olarak da gerçekçi biçimde modellenebilmesine bağlıdır. Bu sunumda güçlü RKA modelleme teknikleri üzerinde durulacak ve geliştirilen algoritmalar [11,12] tanıtılacaktır.

KAYNAKÇA

[1] G. Çakır, M. Çakır, L. Sevgi, "A Discrete Model Creation Algorithm and FDTD/MOM-Based RCS Analysis of Realistic Targets", *IEEE Transactions on Antennas and Propagation*, Feb 2007

[2] U. F. Knott, *Radar Cross Section Measurements,* Van Nostrand Reinhold, New York, 1993

[3] J. F. Shaeffer, M. T. Tuley, E. F. Knot, *Radar Cross Section*, Artech House, Norwood, Mass, 1985

[4] C. Jenn, *Radar and Laser Cross Section Engineering*, Chapter 4.,AIAA Education Series, 1995

[5] L. Sevgi, *Complex Electromagnetic Problems and Numerical Simulation Approaches*, IEEE Press – John Wiley & Sons, Piscataway, New Jersey, 2003.

[6] K.S. Yee, "Numerical Solution of Initial Boundary Value Problems Involving Maxwell' s Equations", *IEEE Trans. AP*, V-14, No.3, pp.302-307, May 1966

[7] K.S. Kunz, R.J. Luebbers. *The Finite-Difference Time-Domain Method for Electromagnetics*, CRC Press, Boca Raton, FL, 1993

[8] R. F. Harrington, *Field Computation by Moment Methods*, New York, The Macmillan Co., 1968.

[9] G. J. Burke, A. J. Poggio, "Numerical Electromagnetic Code-Method of Moments, Part I: Program Description, Theory", Technical Document, 116, Naval Electronics System Command (ELEX 3041), July 1977.

[10] Visit http://www.nittany-scientific.com for NEC-WIN Pro.

[11] L. Sevgi, A. M. Ponsford, H.C. Chan, "An Integrated Maritime Surveillance System Based on Surface Wave HF Radars, Part I – Theoretical Background and Numerical Simulations", *IEEE Antennas and Propagation Magazine*, V.43, N.4, pp.28-43, Aug. 2001.

[12] L. Sevgi, "Target Reflectivity and RCS Interaction in Integrated Maritime Surveillance Systems Based on Surface Wave HF Radar Radars", *IEEE Antennas and Propagation Magazine*, pp. 36-51, Feb. 2001

A3 OTURUMU:

TEMSİLİ ORTAMLAR ve GÖRSELLEŞTİRME

MÜHATEM: MÜŞTEREK HAREKAT ALANI SİMÜLASYON SİSTEMİ: TEMSİLİ ORTAM STANDART VE UYGULAMALARI
Çağatay ÜNDEĞER, Altan YAVAŞ, Mustafa OCAK, Umut AKIN, Ziya İPEKKAN

MÜHATEM YAZILIM PROGRAMLAMA KÜTÜPHANESİ
Çağatay ÜNDEĞER, Altan YAVAŞ, Mustafa OCAK, Umut AKIN, Ziya İPEKKAN

YÜKSEK ÇÖZÜNÜRLÜKLÜ ARAZİ VERİLERİNİN GRAFİK İŞLEMCİSİ KULLANILARAK GÖRSELLEŞTİRİLMESİ
H. Billur ENGİN, Emre KOÇ, Selçuk SÜMENGEN, Selim BALCISOY

SAVUNMA SİSTEMLERİ TASARIMINDA İNSAN BİLGİSAYAR ETKİLEŞİMİ ÇALIŞMALARININ ROLÜ
Gökçe YILDIRIM, İsmail BIKMAZ, Kürşat ÇAĞILTAY, Veysi İŞLER

MÜHATEM: MÜŞTEREK HAREKAT ALANI SİMÜLASYON SİSTEMİ: TEMSİLİ ORTAM STANDART VE UYGULAMALARI[4]

Çağatay ÜNDEĞER [(a)], Altan YAVAŞ [(b)],
Mustafa OCAK [(b)], Umut AKIN [(b)] , Ziya İPEKKAN [(b)]

[(a)] STM A.Ş., Kafkas Sk. No:56, Beştepe, Ankara, cundeger@stm.com.tr

[(b)] Genelkurmay BİLKARDEM Bşk.lığı, Bakanlıklar, Ankara,

{ayavas, mocak, uakin, zipekkan}@tsk.mil.tr

ÖZ

Müşterek Harekat Alanı Simülasyon Sistemi: Temsili Ortam Standart ve Uygulamaları (MÜHATEM) Projesi, ODTÜ TSK Modelleme ve Simülasyon Araştırma ve Uygulama Merkezi (MODSİMMER) Çalışma Planında yer alan hedeflerden biri olan *Müşterek Harekat Alanı Simülasyon Sistemi (MÜHASİS)*'nin geliştirilmesi kapsamında, bir alt yapı projesi olarak bir yılı aşkın bir ön çalışmanın ardından Kasım 2002 yılında başlatılmıştır. Bu bildiride, MÜHATEM Projesi özetle tanıtılmaktadır.

Anahtar Kelimeler: Müşterek Harekat Alanı Simülasyon Sistemi, Temsili Harekat Ortamı, Temsili ortam standartları, Temsili ortam veritabanı, Temsili ortam yazılım kütüphanesi, Temsili ortam modelleme.

MÜHATEM: COMMON MISSION SPACE SIMULATION SYSTEM: SYNTHETIC ENVIRONMENT STANDARDS AND APPLICATIONS

ABSTRACT

Following a one-year of preliminary study, the project, *Common Mission Space Simulation System: Synthetic Environment Standards and Applications (MÜHATEM)*, was started in November 2002 as an infrastructure project in the scope of development of *Common Mission Space Simulation System (MÜHASİS)*, which is one of the objectives of METU TUF Modeling and Simulation Research and Development Center (MODSİMMER) Working Plan. In this paper, MÜHATEM Project is briefly introduced.

Keywords: Common Mission Space Simulation System, Synthetic theather of war, Synthetic environment standards, Synthetic environment database, Synthetic environment software library, Synthetic environment modeling.

[4] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

1. GİRİŞ

Bilgisayar benzetimi (simülasyonu), gerçek veya teorik bir fiziksel sistem modelinin bilgisayar ortamında tasarlanması, çalıştırılması ve çalıştırma sonuçlarının analiz edilmesi disiplinidir. Benzetim, gerçek hayatta oluşturulması zor veya imkansız durumları, sınırları çizilmiş bir düzeyde doğru ve gerçekçi şekilde bilgisayar ortamında yaratmayı sağlar ve denemeye dayalı öğrenme prensibini kullanır. Benzetimlerin, sistemin gerçeğinin yerini tutması hiç bir zaman beklenemez çünkü bir şeyin yüzde yüz benzeri ancak o şeyin kendisi olabilir. Askeri alanda, temsili ortam modelleme, benzetimi yapılacak bir harekatın gerçekleştirileceği sentetik ortamın bilgisayarda sanal olarak oluşturulmasıdır.

MÜHATEM Projesi, TSK'nin tüm modelleme ve simülasyon projelerinde ortak bir ihtiyaç olarak öne çıkan temsili ortam modellemesinin, tekrarların önüne geçilerek sağlanması amacıyla, ODTÜ-TSK MODSİMMER tarafından hazırlanan Modelleme ve Simülasyon Çalışma Planında yer alan hedeflerden biri olan MÜHASİS'nin geliştirilmesi kapsamında, bir alt yapı projesi olarak Kasım 2002'de başlatılmıştır. MÜHATEM Projesi ile;

1. MÜHASİS'in ihtiyaç duyacağı harekat alanı unsurlarının, benzetimi yapılmış harekat ortamında en gerçekçi şekilde bir araya getirilmesine yönelik alt yapının tesis edilmesi,
2. Temsili ortam veri standartlarının belirlenmesi,
3. Temsili ortam verilerinin toplanması, derlenmesi ve bir veritabanı oluşturularak kullanılabilir hale getirilmesi,
4. Tekrarların önlenmesi ile kaynak tasarrufu sağlanması ve
5. Benzetim sistemlerinin birlikte çalışabilirliğinin kolaylaştırılması hedeflenmiştir.

Şekil 1. MÜHATEM Kapsamı

Proje kapsamında (bknz. Şekil 1), tüm harekat ortamının (çevre, varlıklar, insan davranış ve görevleri) temsili için gerekli veriler toplanmakta, derlenmekte ve bir temsili ortam veritabanı hazırlanmaktadır. Ayrıca söz konusu veritabanının benzetim sistemlerince kullanılabilmesi için gerekli yazılım kütüphaneleri (API) ve temsili ortam veritabanının idamesi için gerekli sistem uygulama yazılımları geliştirilmektedir.

MÜHATEM kapsamında ele alınan temsili ortam kaynak verileri Bölüm 2'de ve oluşturulan temsili ortam veritabanı Bölüm 3'te anlatılmaktadır. Geliştirilen temsili ortam yazılım programlama kütüphaneleri ve sistem uygulama yazılımları sırasıyla Bölüm 4 ve 5'te sunulmuştur. Bölüm 6'da, sonuç ve ileriye yönelik çalışmalar yer almaktadır.

2. TEMSİLİ ORTAM KAYNAK VERİLERİ

Temsili ortamın askeri benzetimleri destekleyebilmesi için doğruluğu yüksek gerçek dünya verilerine ihtiyaç bulunmaktadır. Proje kapsamında, söz konusu veriler; çevre, harekat alanı varlıkları ve insan davranışları olarak üç kategoride ele alınmaktadır.

2.1. Çevre

Çevre verileri; kara alanları, su alanları, atmosfer ve astronomi bilgilerine ilişkin dört temel kategoride incelenmektedir. Kara alanlarına ilişkin bilgiler, arazi topoğrafyasını belirleyen sayısal yükseklik verileri (DTED [1], DEM [2], vb.), kağıt haritalardan taranmış raster ekran görüntüleri, uydu görüntüleri, hava fotoğrafları, kızılötesi görüntüler, bitki örtüsü haritaları, eş yükselti eğrileri ve vektör haritalar (Esri Shape [3], Vector Product Format [4], vb.) olarak örneklenebilir.

Su alanları da (denizler, göller, nehirler vb.) kara alanlarına benzer şekilde dip topoğrafyasına, dip örtüsüne ve vektör haritalara (S-57 [5], vb.) ilişkin bilgiler içerir. Ancak bunlara ek olarak çok daha dinamik olan su akıntıları, tuzluluk, sıcaklık gibi bilgilere de ihtiyaç vardır.

Atmosfer verileri, su alanları gibi çok değişkendir ve gece gündüz değişimleri ile hızlı farklılaşma gözlemlenir. Atmosfere ilişkin parametreler, meteorolojik ölçüm istasyonları, uydu görüntüleri ve meteoroloji radarları gibi kaynaklardan elde edilir. MÜHATEM kapsamında ele alınan söz konusu parametrelerin bazıları, sıcaklık (°C), mahalli basınç (Mb), buhar basıncı (Mb), nispi nem (%), yatay görüş mesafesi (m), rüzgar hızı (m/sn), rüzgar yönü (derece), bulut yüksekliği (m), bulut yoğunluğu (0..1), yağış türü (yağmur, kar, dolu) ve yağış yoğunluğudur (mm). Sis yüzdesi, parametre olarak kullanılmamaktadır çünkü sis yatay görüş mesafesi parametresi içine dolaylı olarak ithal edilmiş durumdadır. Yatay görüş mesafesi parametresinin yağışsız bir durumu kapsadığı, yağışın görüşe etkisini içermediği varsayılmaktadır.

Son olarak astronomi bilgileri, güneş, ay ve yıldızların durmu ve dünya üzerine etkilerini barındırır. Güneş ve ayın konum bilgileri, referans alınan bir coğrafi noktaya ve tarih/saate göre gökyüzündeki konumlarını belirten pusula ve yükseliş açıları olarak hesaplanır. Aynı saatte faklı coğrafi noktalarda güneş ve ay farklı konumlarda gözlemlenir. Dolayısıyla geniş alanda çalışan benzetim sistemlerinde, güneş ve ayın konumunu her yerde aynı varsaymak hatalı sonuçlara neden olacaktır. Ancak çok küçük alanlarda bu durum göz ardı edilebilir.

2.2 Harekat Alanı Varlıkları

Harekat alanı varlıkları, çevre içinde yer alan her türlü doğal yada yapay detayı içermektedir. Varlıklar genelde statik ve dinamik olarak iki kategoride ele alınır. Evler, elektrik direkleri, köprüler, yollar gibi statik varlıklar genelde durumlarını hasar alma gibi nadir oluşan olaylara bağlı olarak değiştirirler, bu yüzden çoğunlukla statik kabul

edilirler. İnsanlar, hayvanlar, araçlar, silahlar, mühimmatlar, algı cihazları, muhabere sistemleri, elektronik harp sistemleri ve ek teçhizat gibi varlıklar ise çoğunlukla dinamiktir, dolayısıyla değişken durumlara ve parametrelere sahiptirler, aktif olarak çevre ile etkileşir ve benzetim sistemlerinde olayların akışını tetiklerler. Bu varlıklar benzetim sistemlerinin en hayati aktörleridir ve davranışlarının (fizik kuralları ve zeki varlıkların karar mekanizmaları) doğru modellenmesi sistem sonuçlarının doğruluğuna kritik düzeyde etki eder.

2.3. İnsan Davranışları

Bu noktaya kadar çevre ve varlıkları fiziksel yapıları ile tanımladık. Ancak, bir benzetim sisteminde olayları tetikleyen en temel yönlendirici insandır. Bir harekat alanı, insanların (dost ve düşman askerlerin) davranışları ile şekillenir. Dolayısıyla askerlerin, karar mekanizmasına etki eden moral, cesaret, disiplin, kişisel beceriler, eğitimler, askeri doktrinler, görevler, komuta hiyerarşisi gibi etkenlerin belirli bir sadakat düzeyinde modellenmesine ihtiyaç bulunmaktadır. Ancak birey ve grup davranışları, henüz yeterince anlaşılabilmiş ve bilgisayar ortamında tatmin edici düzeyde modellenebilmiş değildir. Bu konu üzerine, algılama, durum farkındalığı sağlama, veri/bilgi çıkarımı/füzyonu yapma, planlama, tepkisel veya çıkarımsal karar verme, öğrenme, koordinasyon, psikolojik etki gibi bir çok alanda yoğun çalışmalar sürdürülmektedir. MÜHATEM kapsamında, öncelikli olarak, anılan çalışmalara asker davranışlarının modellenmesi açısından ışık tutacak görev ve doktrinlerin, formal yöntemlerle belirlenmesine ve tanımlanmasına yoğunlaşılması planlanmaktadır.

3. TEMSİLİ ORTAM VERİTABANI

Önceki bölümde kısaca özetlenen bilgiler, MÜHATEM kapsamında toplanmakta, derlenmekte, düzenlenmekte ve bir temsili ortam veritabanı haline getirilmektedir. Bu güne kadar, benzetim sistemlerinin en temel girdileri olan çevre ve varlıkların temsil edilmesine yönelik çalışmalar yoğunlukla icra edilmiştir. İnsan davranışlarına ilişkin bilgilerin elde edilmesi kapsamında askeri görevlerin tespit edilmesi çalışması halen sürdürülmektedir. Tamamlanan ve geliştirilmesi devam eden bilgi bankalarına ilişkin özet bilgi takip eden bölümlerde anlatılmaktadır.

3.1. Raster Tabanlı Harita Veritabanı

Raster tabanlı harita veritabanı, raster (imaj benzeri) yapıda ifade edilen harita verilerini tüm dünyayı temsil edebilecek bütünleşik bir veritabanında saklayabilmek için geliştirilmiştir. Raster harita verilerine örnek olarak sayısal yükseklik verileri, uydu görüntüleri gibi iki boyutlu matris şekilinde ifade edilebilen veriler verilebilir. Bu veriler, çevre kategorisinde ele alınır ve sadece arazi yüzeyi veya deniz dibi/yüzeyine ilişkin kısıtlı çeşitlilikte özniteliği (örneğin yükseklik, renk, bitki örtüsü kategorisi) barındırır. Satıhta yer alan ev, bina, tank, gemi gibi varlıklar bu bilgi kapsamında yer almaz. Tüm dünyanın tek bir koordinatlandırma tekniği ile kolayca temsil edilebilmesi için MÜHATEM'de coğrafi koordinat sistemi ve WGS84 datum/elipsoidi standart olarak seçilmiş ve tüm yazılım kütüphanelerinde bu standart kullanılmıştır.

MÜHATEM raster tabanlı harita veritabanı, mevcut durumda bir dosya indeksleme mekanizması ile (dosya ve klasörler şeklinde) saklanmakta olup, çoklu çözünürlük ve geniş alan yönetimi yeteneklerine sahiptir. İlk tasarımda, performans ihtiyaçları

dolayısıyla standart bir veritabanı (SQL-Server, Oracle, MS.Access vb.) kullanılmamıştır. Ancak performansı yüksek bir veri saklama yapısının tasarlanabilmesi durumunda standart bir veritabanı kullanımına geçilmesi planlanmaktadır. Raster tabanlı veriler, veritabanında bloklar şeklinde ve birden fazla çözünürlükte saklanmaktadır. Her blok diğer bloklardan bağımsız olarak hafızaya istenilen çözünürlükte yüklenebilmekte veya hafızadan atılabilmektedir. Bu sayede veri yapısı geniş alan yönetim sistemi desteği sağlamaktadır.

3.2. Vektör Tabanlı Harita Veritabanı

Vektör tabanlı harita veritabanı, vektör (nokta, çizgi, alan, ağ vb.) yapıda ifade edilen harita verilerini tüm dünyayı temsil edebilecek bütünleşik bir veritabanında saklayabilmek için geliştirilmektedir. Bu çalışma henüz tamamlanmamıştır. Vektör harita verilerine örnek olarak ülke/il/ilçe sınırları ve merkezlerinin konumları, nehirler, göller, demiryolları, orman alanları, binalar, köprüler, iletişim hatları gibi nokta, çizgi ve alanlarla tanımlanabilen veriler verilebilir. Bu veriler raster haritalardan farklı olarak, satıhta yer alan varlıklara ilişkin konum ve öznitelik bilgilerini tutabilmektedir. Raster haritalar (örneğin kağıt haritalardan taranmış raster ekran görüntüleri) üzerinde sorgulama yapmak mümkün olmazken, vektör haritalar üzerinde çok çeşitli sorgulamalar ve analizler yapılabilmektedir.

3.3. Zamana Dayalı Parametre Veritabanı

Zamana dayalı parametre veritabanı, zaman ve konum ile ilişkili olarak değişimler gösteren parametrelerin değerlerinin saklanması ve enterpolasyon teknikleri ile, tanımlanmamış konum ve zamanlar için ara değerlerin hesaplanabilmesini sağlamak için geliştirilmiştir. Bu tür bilgilere örnek olarak rüzgar hızı/yönü, sıcaklık, nem, bulutluluk gibi atmosfer verileri ve akıntı hızı/yönü, tuzluluk, sıcaklık gibi deniz/göl/nehir suyu bilgileri verilebilir.

MÜHATEM zamana dayalı parametre veritabanı, standart bir veritabanında saklanmakta olup geniş alan yönetimi desteğine sahiptir. Bir veri katmanı için istenilen sayıda parametre tanımlanarak her parametre için bir hesaplama tekniği belirlenebilir. Parametre hesaplamaları, sabit bir değerden, zaman cetveli ile değişimi tanımlanmış bir değer dizisinden veya coğrafya üzerinde bir yada daha fazla noktada tanımlanmış parametre değerlerinden enterpolasyon ile ara değerler bulunarak yapılabilmektedir. Eğer sabit değer seçilmişse, konum ve zaman ne olursa olsun parametrenin değeri sabit kalacaktır. Zaman cetveli yöntemi seçildiğinde ise simülasyon başından mevcut simülasyon zamanına kadar geçen süreye bağlı olarak zaman cetvelinden enterpolasyon ile uygun değer hesaplanır. Sonuncu yöntem de ise, coğrafya üzerinde belirli koordinatlar ve irtifalar örnekleme için seçilerek, bu noktalarda parametrenin farklı zamanlarda aldığı değerlerin belirtilmesi gereklidir. Yeterli sayıda örnekleme yapıldığında önceden belirtilmemiş koordinat, irtifa ve zamanlar için ara değerler enterpolasyon teknikleri ile gerçekci olarak tahmin edilebilir. Bir örnekleme noktası için, örneklemenin yapılacağı koordinat, bu koordinat üzerinde hangi irtifalarda örnek değerlerin tanımlanacağı ve noktadaki parametrenin değerinin nasıl hesaplanacağı belirtilmektedir. Parametre hesaplamaları, sabit bir değerden, zaman cetveli ile değişimi tanımlanmış bir değer dizisinden veya geçmiş ölçümlere dayalı istatistiklerden

yapılabilmektedir. İstatistiklerin sağlıklı şekilde oluşturulabilmesi için noktanın farklı saat, gün, ay ve yıllarda ölçülmüş çok sayıda değerine sahip olunması gereklidir.

Zamana dayalı parametre veritabanının ilk uygulaması, Devlet Meteoroloji İşleri Müdürlüğü ile koordineli çalışmalar sonucunda, tüm Türkiye'nin 2000-2004 yılları arasındaki ölçümlerine bağlı atmosfer parametrelerinin istatistikleri çıkarılarak oluşturulmuştur. Bu sayede atmosfer verilerinin uzun dönem ortalama ve standart sapma bilgileri elde edilmiş ve enterpolasyon teknikleri kullanılarak farklı zaman ve konumlardaki parametre değerleri gerçekci bir şekilde tahmin edilebilmiştir.

3.4. Astronomi Bilgileri

MÜHATEM kapsamında, astronomi bilgilerinin hesaplanması teknikleri üzerine çalışmalar sürdürülmektedir. Bu çalışmalar bir veritabanı oluşturulmasından ziyade astronomik hesaplama teknikleriyle güneş, ay ve yıldızların durumu ve gökyüzü aydınlanmasına etkilerinin belirlenmesine dayanmaktadır. Şu ana kadar güneşin gündüz etkilerine ilişkin fizik tabanlı algoritmalar geliştirilmiş olup gökyüzünün renk tayfının ve aydınlanma miktarının hesaplanması sağlanmıştır. Ay ve yıldızlara ilişkin bilgiler ve gece modellemesi üzerine çalışmalar halen sürdürülmektedir.

3.5. Harekat Alanı Varlık Bilgileri

MÜHATEM'de varlıklar iki farklı yolla tanımlanabilmektedir. Bunlardan ilki, *harekat alanı kavramsal modeli katmanı* ile tanımlama, ikincisi ise *varlık erişim katmanı* ile tanımlamadır. Kavramsal model katmanı kullanıldığında, varlıkların öznitelikleri ve diğer varlıklarla olan ilişkileri kavramsal olarak ifade edilir. İki yada daha çok varlık arasında cümleler şeklinde ilişkiler tanımlanması ve bu ilişkilerin detaylı olarak sorgulanabilmesi mümkündür. Varlık erişim katmanı ise temsili ortamda yer alan varlıkların çalışabilir modellerinin oluşturulabilmesi için geliştirilmiştir. Varlık erişim katmanlarının en önemli özelliği eklenti (plug-in) mekanizması ile sınırsız genişleyebilme kabiliyetidir. Bu kabiliyet sayesinde, hem MÜHATEM geliştirici ekibinin hem de diğer kurum/kuruluş ve firmaların, MÜHATEM'e yeni ve özelleşmiş varlık tipleri ekleyebilmesi veya eskiden tanımlanmış varlıkları değiştirebilmesi mümkün olmaktadır.

Kavramsal model ve varlık erişim katmanları için özel bir veritabanı yapısı oluşturulmamıştır. Uygulamaların ihtiyaçlarına göre katman verileri, hafıza, dosya veya veritabanı alanlarına ikili (binary) veri şeklinde kayıt edilebilir ve geri yüklenebilir. Varlıklar, senaryoların aktif kullanılan elemanlarıdır ve yoğun olarak uygulamaya özel ihtiyaçlar barındırırlar. Dolayısıyla bir veritabanı ile geliştiricileri kısıtlamak sınırlayıcı olacak, esnekliği kısıtlayacaktır.

4. TEMSİLİ ORTAM YAZILIM KÜTÜPHANELERİ

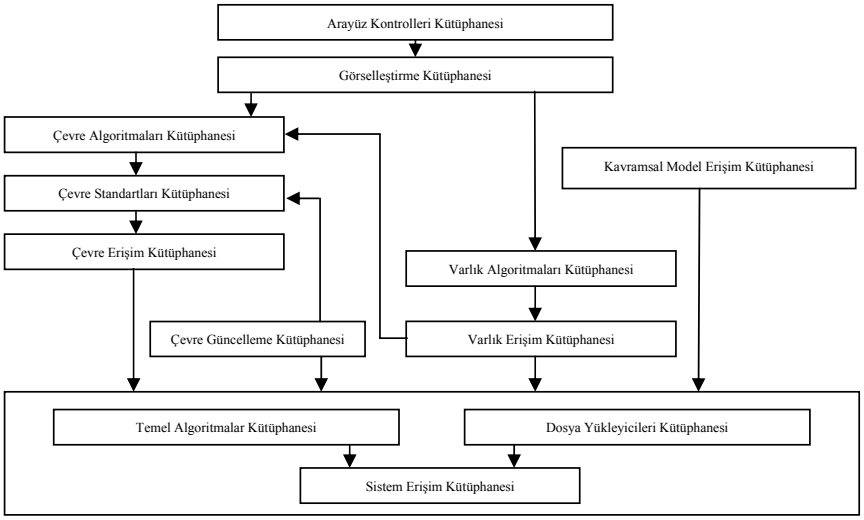
MÜHATEM kapsamında, temsili ortam veritabanının oluşturulması ve kullanılması için C++ tabanlı bir yazılım kütüphanesi (API) geliştirilmektedir. Şu an için API, Windows işletim sisteminde Borland C++ Builder ve Microsoft Visual C++ ve Linux tabanlı işletim sistemlerinde Borland Kylix'te çalışabilmektedir. Bu kütüphane, veritabanı erişimine ek olarak benzetim sistemlerinde sıklıkla kullanılan temel ve özelleşmiş algoritmaları da sağlamaktadır. Söz konusu algoritmalara örnek olarak, koordinat ve

vektör dönüşümleri, çokgen işlemleri, dosya yükleme, yapay zeka ve görselleştirme algoritmaları verilebilir. MÜHATEM yazılım kütüphaneleri Şekil 2'de gösterilen katmanlı yapıya sahiptir.

5. TEMSİLİ ORTAM SİSTEM UYGULAMA YAZILIMLARI

Temsili ortam veritabanının oluşturulması ve idame edilmesi için MÜHATEM yazılım kütüphaneleri ve bu yazılım kütüphaneleri kullanılarak Borland C++ Builder uygulama geliştirme ortamı ile altı tane sistem uygulama yazılımı geliştirilmiştir. Bu uygulamalar aşağıda kısaca özetlenmiştir.

Raster Tabanlı Harita Veritabanı Güncelleme Editörü, çeşitli kaynaklardan elde edilen raster tabanlı harita dosyalarının yüklenmesi, düzenlenmesi ve MÜHATEM raster tabanlı harita veritabanına aktarılmasını sağlayan sistem uygulama yazılımıdır.



Şekil 2. MÜHATEM Yazılım Kütüphaneleri

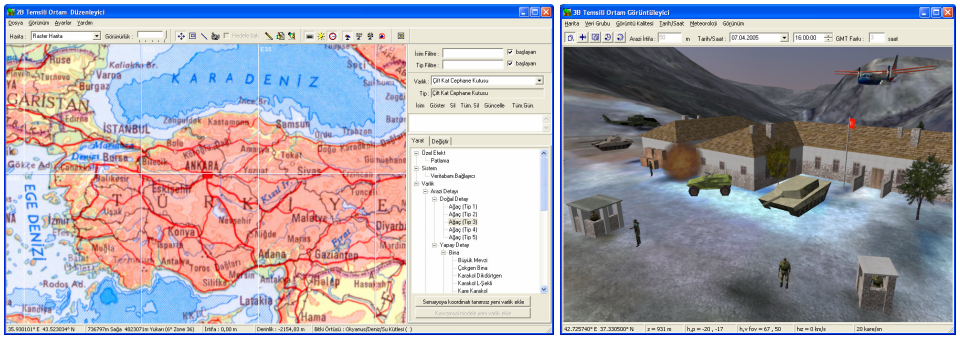
Zamana Dayalı Parametre Veritabanı Güncelleme Editörü, konum ve zaman ile değerleri değişim gösteren parametrelere ilişkin çeşitli kaynaklardan elde edilen ölçümlerinin (örneklemelerin) yüklenmesi, düzenlenmesi ve MÜHATEM zamana dayalı parametre veritabanına aktarılmasını sağlayan sistem uygulama yazılımıdır. Tanımlanan bir dizi parametrenin, geçmiş ölçüm değerleri kullanılarak uzun dönem istatistikleri çıkartılabilmekte ve belirtilen bir tarih ile konumdaki uzun yıllara dayalı ortalama değerleri kolayca hesaplanabilmektedir.

Harekat Alanı Kavramsal Modeli Editörü, harekat sahası varlıklarını (tank, top, asker, uçak, sembol vb.), anılan varlıkların öznitelikleri (hız, irtifa, vb.) ve başka varlıklarla ilişkilerini (helikopter tanka ateş edebilir, vb.) kavramsal olarak tanımlamak için geliştirilmiş bir sistem uygulama yazılımıdır.

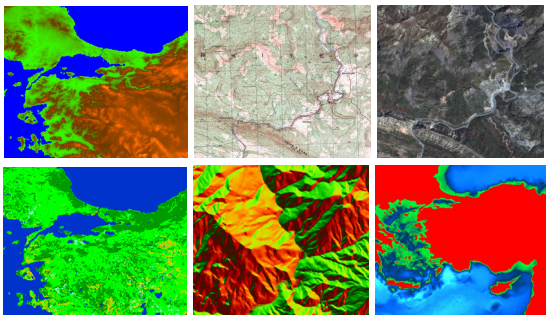
Temsili Ortam Veritabanı Düzenleyici, belirlenen bir bölgenin, raster tabanlı haritaları, meteroloji ve astronomi bilgilerini görüntülemeye ve bu bölge üzerinde yer

alan varlıkları, eklentiler (plug-in) ile tanımlamaya imkan veren bir sistem yazılımıdır. Şekil 3'te bu uygulamaya yönelik örnek bir ekran görüntüsü verilmektedir.

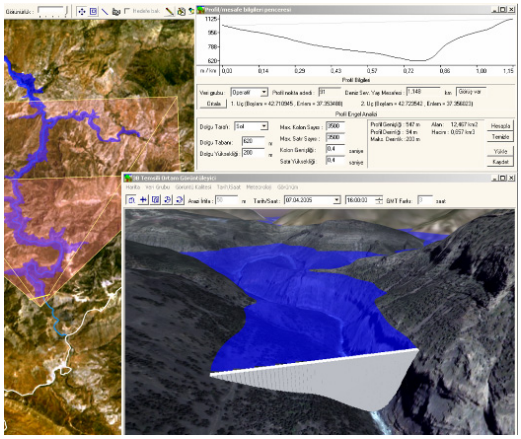
Anılan editör, alan sınırlaması olmaksızın seçilen bir coğrafi bölgenin çeşitli tipde raster haritalarını (bknz. Şekil 4) OpenGL destekli şekilde görüntüleme, görüntülenen coğrafi alanın boyutuna göre otomatik olarak raster harita ölçeğini belirleme, haritaları kabartmalı veya şeffaf şekilde gösterme, arazi üzerinde kesit/bent analizleri yapma (bknz. Şekil 5) ve koordinat dönüşümleri sağlama kabiliyetlerine sahiptir. Ayrıca harita üzerine çizim (nokta, çizgi, alan), varlık (araç, insan, bina, tel çit, engel vb.) ve semboller koyulması, ortamın iki ve üç boyutlu olarak görüntülenmesi, sis etkilerinin hesaplanması, güneş ve ayın belirtilen bir tarih ve konumdaki durumunun ve gökyüzü ile varlıkların aydınlanmasına etkilerinin belirlenmesi sağlanmaktadır (bknz. Şekil 6).



Şekil 3. Temsili Ortam Veritabanı Düzenleyici



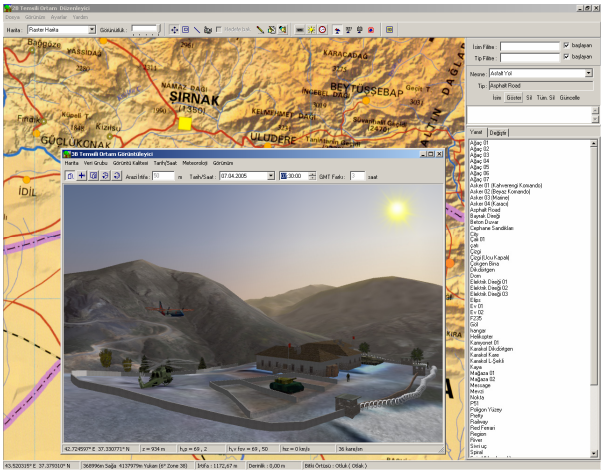
Şekil 4. Sayısal yükseklik haritası, raster ekran görüntüsü, uydu görüntüsü/hava fotoğrafı (üst sıra), yüzey/bitki örtüsü haritası, eğim haritası ve derinlik haritası (alt sıra)

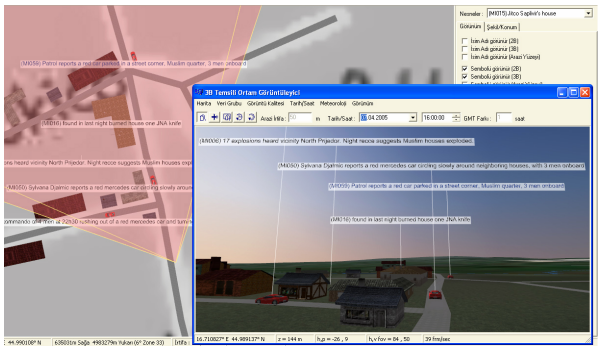


Şekil 5. Kesit/bent analizi

Sinir Ağları Düzenleme Editörü, yapay zeka uygulamalarında sıkça kullanılan sinir ağlarının (neural networks) bir arayüz vasıtasıyla tanımlanması, eğitilmesi ve kullanılmasını sağlayan bir uygulama yazılımıdır. Oluşturulan sinir ağı başka yazılımların kullanımı için bir dosyaya kayıt edilebilir.

Karar Ağaçları Düzenleme Editörü, yapay zeka uygulamalarında sıkça kullanılan karar ağaçlarının (decision trees) bir arayüz vasıtasıyla tanımlanması, eğitilmesi ve kullanılmasını sağlayan bir uygulama yazılımıdır. Oluşturulan karar ağacı başka yazılımların kullanımı için bir dosyaya kayıt edilebilir.





Şekil 6. İki ve üç boyutlu gösterim

6. SONUÇ

Bu bildiride, ODTÜ TSK MODSİMMER tarafından 2002 yılından beri geliştirilmesine devam edilen MÜHATEM Projesi kısaca tanıtılmıştır. Bu projenin temel amacı, tüm modelleme ve simülasyon projelerinde ortak ihtiyaç olan temsili ortam modelinin, tek bir alt yapı tarafından sunulmasıdır. Böylece hem kaynak tasarrufu sağlanması hem de modelleme ve simülasyon sistemlerinin bir arada çalışabilirliğinin kolaylaştırılması hedeflenmiştir. Ayrıca her projede ortam modelleme çalışmalarının baştan yapılmasının yeterli teknolojik olgunluğa erişimde zaman kaybına neden olduğu, uzun vadeli tek bir projenin teknolojiyi hızlı bir şekilde takip etmede avantaj sağlayacağı değerlendirilmiştir. MÜHATEM'in sunduğu alt yapının bundan sonra başlatılacak tüm modelleme ve simülasyon projelerinde bir standart olarak kullanılması ve kurum/kuruluş ile firmalar tarafından gerektiğinde geliştirime katkı sağlanması planlanmaktadır.

MÜHATEM'in geliştirilmesine halen devam edilmekte olup ara sürümleri, "Küçük Ölçekli Harekatın Modellenmesi ve Simülasyonu (BASKIN)" ile "Sınır Gözetleme ve Kontrol Sistemi C4ISR Faaliyetlerinin Modellenmesi ve Simülasyonu (SGKS C4ISR-MOS)" projelerinde kullanılmış, "Müşterek Görev Kuvvetinin Modellenmesi ve Simülasyonu (MGKMOS)" ile "Satıhta Konuşlu Aktif Hava Savunma Harekatının Modellenmesi (SAMMOS)" projelerinde kullanılmakta ve "Deniz Harp Oyunu Simülasyon Sistemi (DEHOS)" projesinde kullanımı planlanmaktadır. Böylece MÜHASİS'in kara, hava ve deniz ayağının önemli bir parçasını oluşturması planlanan MGKMOS, SAMMOS ve DEHOS projelerinin bir arada çalışabilirliğinde HLA ve MÜHATEM kullanımının önemli bir fayda sağlayacağı değerlendirilmektedir.

MÜHATEM projesinde bu güne kadar, raster tabanlı haritalar, meteoroloji, astronomi ve harekat alanı varlıklarına ilişkin bilgilerin toplanması, derlenmesi, kullanılabilir hale getirilmesi ve görselleştirilmesine yönelik çalışmalar yoğunlukla icra edilmiş olup, bundan sonraki aşamada eski çalışmaların iyileştirilmesi; vektör tabanlı harita alt yapısının oluşturulması; parçacık, fizik ve ses motorlarının geliştirilmesi; HLA entegrasyonunun sağlanması ve ileri seviye algoritmaların (görüş/çarpışma testleri yapma, kapsama haritası çıkarma, hendek/çukur oluşturma, ışık/gölge hesaplama, algı ve iletişim hesapları yapma, vb.) geliştirilmesi planlanmaktadır.

KAYNAKÇA

[1] Defense Mapping Agency (DMA), (1996), "Performance Specification Digital Terrain Elevation Model (DTED)" , MIL-PRF-89020.

[2] United States Geological Survey (USGS), (2004), "Standards for Digital Elevation Model (DEM)".

[3] Environmental Systems Research Institute, Inc. (ESRI), (1998), "ESRI Shapefile Technical Description", White Paper.

[4] Defense Mapping Agency (DMA), (1995), "Performance Specification Vector Smart Map (VMap) Level 0", MIL-PRF-89039.

[5] International Hydrographic Organization (IHO), (2000), "IHO Transfer Standard For Digital Hydrographic Data", S-57 Edition 3.1.

MÜHATEM YAZILIM PROGRAMLAMA KÜTÜPHANESİ[5]

Çağatay ÜNDEĞER [(a)], Altan YAVAŞ [(b)],
Mustafa OCAK [(b)], Umut AKIN [(b)] , Ziya İPEKKAN [(b)]

[(a)] STM A.Ş., Kafkas Sk. No:56, Beştepe, Ankara, cundeger@stm.com.tr
[(b)] Genelkurmay BİLKARDEM Bşk.lığı, Bakanlıklar, Ankara,
{ayavas, mocak, uakin, zipekkan}@tsk.mil.tr

ÖZ

Müşterek Harekat Alanı Simülasyon Sistemi: Temsili Ortam Standart ve Uygulamaları (MÜHATEM) Projesi, ODTÜ-TSK Modelleme ve Simülasyon Araştırma ve Uygulama Merkezi (MODSİMMER) Çalışma Planında yer alan hedeflerden biri olan *Müşterek Harekat Alanı Simülasyon Sistemi (MÜHASİS)*'nin geliştirilmesi kapsamında, bir alt yapı projesi olarak bir yılı aşkın bir ön çalışmanın ardından Kasım 2002 yılında başlatılmıştır. Bu bildiride, MÜHATEM Projesi kapsamında geliştirilen yazılım programlama kütüphanesi özetle tanıtılmaktadır.

Anahtar Kelimeler: Müşterek Harekat Alanı Simülasyon Sistemi, Temsili harekat ortamı, Temsili ortam standartları, Temsili ortam veritabanı, Temsili ortam yazılım kütüphanesi, Temsili ortam modelleme.

MÜHATEM SOFTWARE PROGRAMMING LIBRARY

ABSTRACT

Following a one-year of preliminary study, the project, *Common Mission Space Simulation System: Synthetic Environment Standards and Applications (MÜHATEM)*, was started in November 2002 as an infrastructure project in the scope of development of *Common Mission Space Simulation System (MÜHASİS)*, which is one of the objectives of METU TUF Modeling and Simulation Research and Development Center (MODSİMMER) Working Plan. In this paper, the application programing library developed in MÜHATEM Project is briefly introduced.

Keywords: Common Mission Space Simulation System, Synthetic theather of war, Synthetic environment standards, Synthetic environment database, Synthetic environment software library, Synthetic environment modeling.

1. GİRİŞ

[5] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

Bilgisayar benzetimi (simülasyonu), gerçek veya teorik bir fiziksel sistem modelinin bilgisayar ortamında tasarlanması, çalıştırılması ve çalıştırma sonuçlarının analiz edilmesi disiplinidir. Benzetim, gerçek hayatta oluşturulması zor veya imkansız durumları, sınırları çizilmiş bir düzeyde doğru ve gerçekçi şekilde bilgisayar ortamında yaratmayı sağlar ve denemeye dayalı öğrenme prensibini kullanır. Benzetimlerin, sistemin gerçeğinin yerini tutması hiç bir zaman beklenemez çünkü bir şeyin yüzde yüz benzeri ancak o şeyin kendisi olabilir. Askeri alanda, temsili ortam modelleme, benzetimi yapılacak bir harekatın gerçekleştirileceği sentetik ortamın bilgisayarda sanal olarak oluşturulmasıdır.

MÜHATEM Projesi, TSK'nin tüm modelleme ve simülasyon projelerinde ortak bir ihtiyaç olarak öne çıkan temsili ortam modellemesinin, tekrarların önüne geçilerek sağlanması amacıyla, ODTÜ TSK MODSİMMER tarafından hazırlanan Modelleme ve Simülasyon Çalışma Planında yer alan hedeflerden biri olan MÜHASİS'nin geliştirilmesi kapsamında, bir alt yapı projesi olarak Kasım 2002'de başlatılmıştır. MÜHATEM Projesi ile;

6. MÜHASİS'in ihtiyaç duyacağı harekat alanı unsurlarının, benzetimi yapılmış harekat ortamında en gerçekçi şekilde bir araya getirilmesine yönelik alt yapının tesis edilmesi,
7. Temsili ortam veri standartlarının belirlenmesi,
8. Temsili ortam verilerinin toplanması, derlenmesi ve bir veritabanı oluşturularak kullanılabilir hale getirilmesi,
9. Tekrarların önlenmesi ile kaynak tasarrufu sağlanması ve
10. Benzetim sistemlerinin birlikte çalışabilirliğinin kolaylaştırılması hedeflenmiştir.

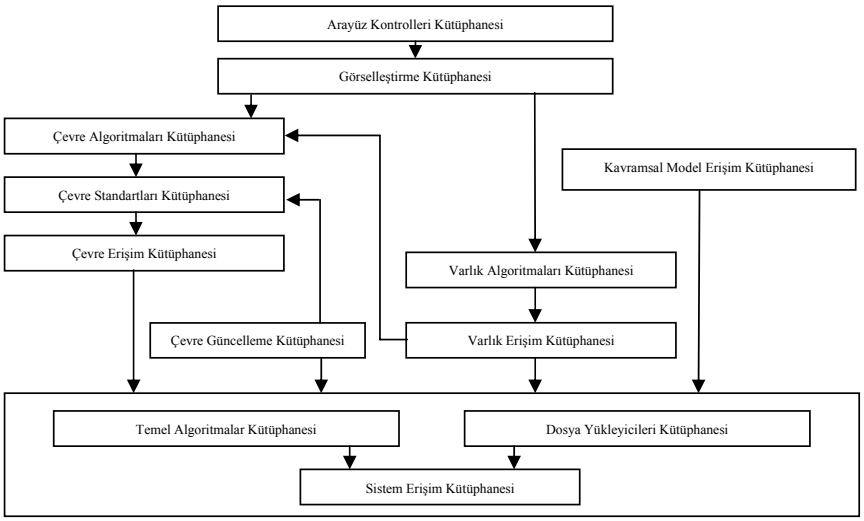
Şekil 1. MÜHATEM Kapsamı

Proje kapsamında (bknz. Şekil 1), tüm harekat ortamının (çevre, varlıklar, insan davranış ve görevleri) temsili için gerekli veriler toplanmakta, derlenmekte ve bir temsili ortam veritabanı hazırlanmaktadır. Ayrıca söz konusu veritabanının benzetim sistemlerince kullanılabilmesi için gerekli yazılım kütüphaneleri (API) ve temsili ortam veritabanının idamesi için gerekli sistem uygulama yazılımları geliştirilmektedir. Bu bildiride, geliştirilen yazılım programlama kütüphaneleri kısaca anlatılmakta olup detaylı bilgi için [1]'ye bakılması önerilir.

MÜHATEM kapsamında geliştirilen kütüphanelerin genel yapısı Bölüm 2'de, alt yapı kütüphaneleri Bölüm 3'te, çevre ilişkili kütüphaneler Bölüm 4'te, harekat alanı varlıklarına ilişkin kütüphaneler Bölüm 5'te ve görselleştirme ile arayüz kontrollerine yönelik kütüphaneler Bölüm 6'da anlatılmaktadır. Bölüm 7'de, sonuç ve ileriye yönelik çalışmalar yer almaktadır.

2. MÜHATEM YAZILIM PROGRAMLAMA KÜTÜPHANESİ

MÜHATEM yazılım programlama kütüphaneleri C++ tabanlı olarak geliştirilmektedir. Şu an için API, Windows işletim sisteminde Borland C++ Builder ile Microsoft Visual C++ ve Linux tabanlı işletim sistemlerinde Borland Kylix'te çalışabilmektedir. Bununların dışında bir JNI ara katmanı yazılarak hem Windows'da hem de Linux'te JAVA tarafından kullanılabilirliği test edilmiştir.



Şekil 2. MÜHATEM Yazılım Kütüphaneleri

Hız ihtiyacı çok yoğun olan benzetim sistemlerinde, C++ kullanımı JAVA ve C# gibi yorumlayıcılara oranla çok daha fazladır. JAVA gibi dillerin birden fazla işletim sistemine kolay taşınabilirliği bu dillerin kullanımını cezbetse bile bu dillerde hazırlanan benzetim sistemlerinin performansları beklenen düzeyde olmamaktadır. C++ ile hazırlanan yazılımlarının standard C/C++ kütüphaneleri dışındaki komutları yoğun olarak kullanmamaları farklı işletim sistemlerine taşınabilirliklerini kolaylaştırmakta ve genelde sadece kullanıcı arayüzleri açısından geçişler zaman almaktadır. Hatta, platform bağımsız kullanıcı arayüz tasarım kütüphaneleri (TCL/TK [2], vb.) kullanan uygulamalarda bu geçiş de zaman almamaktadır. MÜHATEM'in geliştirilmesinde işletim sistemi, derleyici ortamı ve kullanıcı arayüzlerine özel komutların kullanımından azami düzeyde kaçınılmış, kullanımının gerekli olduğu durumlarda ileride anlatılacak olan *Sistem Erişim Kütüphanesi* üzerinden bir ara katman aracılığıyla erişim sağlanmıştır. MÜHATEM yazılım kütüphaneleri Şekil 2'de gösterilen katmanlı yapıya sahiptir. Bu katmanlar, müteakip bölümlerde anlatılan *alt yapı*, *çevre*, *varlıklar* ve

görselleştirme/arayüz kontrolleri kütüphaneleri olarak dört temel kategoride ele alınmaktadır.

3. ALT YAPI KÜTÜPHANELERİ

Alt yapı kütüphaneleri, benzetim sistemleri tarafından yoğunlukla ihtiyaç duyulan genel fonksiyon ve algoritmaları içerir. Bu bölümde genelde MÜHATEM yapısına özelleşmiş kısımlar yer almaz. İlerleyen alt bölümlerde alt yapı kütüphaneleri özetlenmiştir.

3.1 Sistem Erişim Kütüphanesi

Bu kütüphane, işletim sistemi kaynaklarına (metin, tarih, görüntü, ses, disk, hafıza, veritabanı, vb. işlemler) erişimi sağlayan fonksiyon ve nesne sınıflarını içerir ve derleyici/işletim sistemi özelinde kodları barındıran tek kütüphanedir. Diğer yazılım kütüphaneleri bu kütüphane üzerinden sistem kaynaklarına erişir. Böylece kodların derleyici ve platform bağımsız olması sadece bu kütüphanenin güncellenmesi ile sağlanır.

3.2 Temel Algoritmalar Kütüphanesi

Bu kütüphane, genel maksatlı algoritmaları ihtiva eder. Koordinat dönüşümleri (Coğrafi/UTM/kartezyen koordinat sistemi ve WGS-84/ED-50 datum dönüşümleri, vb.), vektör (çevirme, açı hesaplama, çizgi kesişim testleri, matris işlemleri, vb.) ve çokgen işlemleri (delaunay üçgenleştirmesi, çokgen üçgenleştirmesi, kesişim testleri, vb.), yapay sinir ağları, karar ağaçları, koşum zamanı değişkenleri ve koşum zamanı yorumlayıcıları (run-time interpreter - script programlama) bunlara örnek olarak verilebilir. Ayrıca bu kütüphane içinde bütün üst katmanlardaki nesne sınıflarının kalıtım yolu ile türetildiği *TObjectBaseClass* adlı temel nesne sınıfı bulunmaktadır. Bu nesne sınıfı, derleme (compile-time) ve koşum zamanı (run-time) değişkenleri ile metodlarına kullanıcı arayüzleri ve script programlama ile erişimi sağlayan alt yapı metodlarına sahiptir. Bu nesneler üzerinde, C++ benzeri bir dilde script metodlar tanımlamak ve çalıştırmak mümkündür.

3.3 Dosya Yükleyicileri Kütüphanesi

Bu kütüphane, çeşitli formatdaki dosyaların yüklenmesini sağlayan nesne sınıflarını içerir. İmaj yükleme (Bitmap-*.bmp, Truevision TARGA-*.tga, Audio Video Interleave-*.avi, vb.), üç boyutlu model yükleme (3D Studio-*.3ds, Quake 2 Model File-*.md2, vb.), raster coğrafi veri yükleme (Digital Elevation Data-*.dt?, Digital Elevation Model-*.dem, Tagged Image File-*.tif, vb.), vektör coğrafi veri yükleme (ESRI Shape-*.shp, S-57, vb.), sembol yükleme (vektör ve imaj semboller) ve XML dosya yükleme olarak altı ana kategoride geliştirilmiştir. İleride ses dosyalarını yüklemeye ilişkin bir kısmın da eklenmesi planlanmaktadır.

4. ÇEVRE KÜTÜPHANELERİ

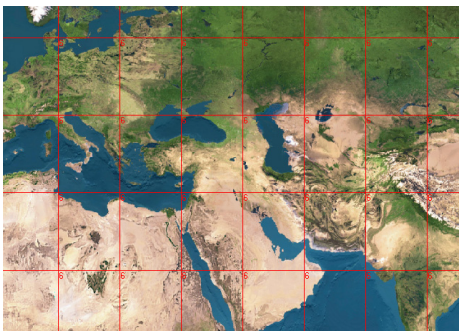
Bu kütüphane, kara, su alanları, atmosfer ve astronomiye ilişkin MÜHATEM veritabanlarının yaratılması ve kullanılmasını sağlayan nesne sınıflarını içerir. Çevre erişim, çevre standartları, çevre algoritmaları ve çevre güncelleme olmak üzere dört alt kütüphaneden oluşur.

4.1 Çevre Erişim Kütüphanesi

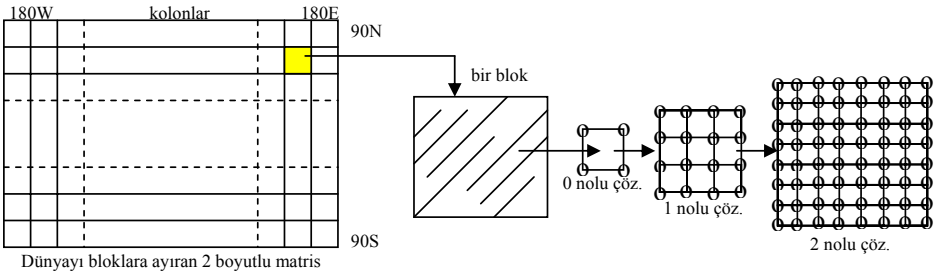
Çevre erişim kütüphanesi, çevre verilerini oluşturmak, bütünleştirmek, saklamak ve geri yükleyerek kullanabilmek amacıyla hazırlanmış jenerik bir yazılım kütüphanesidir. Raster tabanlı harita verileri, vektör tabanlı harita verileri ve nokta tabanlı zaman ilişkili coğrafi veriler olarak üç alt kütüphaneye bölünür.

4.1.1. Raster Tabanlı Harita Verileri

Raster tabanlı harita verileri alt kütüphanesi, tüm dünyayı coğrafi koordinatlar cinsinden eşit boyutlu bloklara bölerek 2 boyutlu bir matrise (bknz. Şekil 3) dönüştürür. Matrisin her elemanı (bloğu) kendi içinde çok çözünürlüklü olarak tutulan 2 boyutlu bir dizi matris daha barındırır. Bir blok 0 ile 9 arasında bir çözünürlükte yaratılabilir, kayıt edilebilir ve geri yüklenebilir. Çözünürlük, bir bloğun verilerini tutan 2 boyut matrisin boyutlarını ifade eder. Çözünürlüğü *n* olan bir bloğun verilerini tutan matrisinin boyutları $2^{(n+1)}$x$2^{(n+1)}$'dir. Dolayısıyla 9 nolu çözünürlüğe sahip bir matrisin boyutları 1024x1024'dür. Matrisin her elemanı bir coğrafi noktaya ilişkin istenilen tiplerde, bir yada daha fazla alanda veri tutar (yükseklik, derinlik, renk (RGB), vb.). Bir blok matrisinin en kuzey ve doğu kenarlarındaki elemanlar kuzey, doğu ve kuzey-doğudaki komşu blokların elemanlarıyla çakışır. Dolayısıyla *n* çözünürlüklü bir matris, bir bloğu $(2^{(n+1)}-1)$x$(2^{(n+1)}-1)$ parçaya böler (bknz. Şekil 4).



Şekil 3. Dünyanın bloklara ayrılması



Şekil 4. Blokların çözünürlükleri içermesi

Bir bloğun maksimum 10 çözünürlük ile sınırlandırılmasının nedeni, bir blok için gerekli hafıza ihtiyacının çok artmamasının ve verileri yükleme ile işleme

performansının çok fazla düşmemesinin garanti edilmesidir. Ancak bu sınırlama ile çok yüksek çözünürlüklü verilerin (örneğin 1 metre) tutulabilmesi için dünyanın çok fazla sayıda bloğa bölünmesi gerekir. Çok fazla blok kullanımı, yüksek çözünürlüklü ama dar alanda çalışan benzetim sistemleri (örneğin kara aracı simülatörleri) için idealdir ancak çok düşük çözünürlüklü ve geniş alanda çalışan bir benzetim sisteminde (örneğin JTLS) küçük küçük bir çok bloğun işlenmesinden dolayı performans çok fazla düşmektedir. Dolayısıyla MÜHATEM'de dünya *stratejik*, *operatif* ve *taktik* olarak üç farklı detay grubunda bloklamaya tabi tutulmuştur. Yazılım geliştiriciler, ihtiyaçlarına göre bu detay gruplarının bir yada daha fazlasını aynı anda kullanabilmektedir.

Stratejik detay grubu, dünyayı 32x16 adet bloğa böler ve her blok en fazla 39,5 saniye (yaklaşık 1 km) çözünürlüklü veri tutabilir. Operatif detay grubu, dünyayı 2048x1024 adet bloğa böler ve her blok en fazla 0,6 saniye (yaklaşık 17 m) çözünürlüklü veri tutabilir. Son olarak taktik detay grubu, dünyayı 131072x65536 adet bloğa böler ve her blok en fazla 0,009 saniye (yaklaşık 27 cm) çözünürlüklü veri tutabilir. İleride daha yüksek çözünürlüklü verilere ihtiyaç olduğu durumda, mühendislik detay grubu eklenerek 4 mm düzeyine inmek mümkündür.

Anılan bloklar, kullanıcı açısından 2 boyutlu bir matris üzerinden erişiliyor gibi gözükse de aslında hafızada yer kaplamamaları açısından ağaç yapısında tutulmaktadırlar. Her blok bir birinden bağımsız olarak yaratılabilmekte, veritabanından yüklenebilmekte ve hafızadan silinebilmektedir. Bloklar istenilen çözünürlükte (veri mevcutsa) yüklenebilmektedir. Bir çözünürlük yüklendiğinde o çözünürlükten küçük olan tüm çözünürlüklerde hafızaya yüklenmektedir. Örneğin 3 nolu çözünürlüğün yüklenmesi aynı zamanda 0, 1 ve 2'nin de yüklenmiş olması anlamına gelir. Bu kütüphane kişiselleştirilebilir blok verileri tutmaya yönelik olup, yüklenen verilerin yorumlanmasını "Çevre Standartları Kütüphanesi"ne bırakır.

4.1.2. Vektör Tabanlı Harita Verileri

Vektör tabanlı harita verileri alt kütüphanesinin, tüm dünyayı adresleyebilecek şekilde vektör verileri tutabilmesi planlanmaktadır. Ancak bu kütüphanenin tasarımı henüz tamamlanmadığı için bu bildiride anlatılmayacaktır.

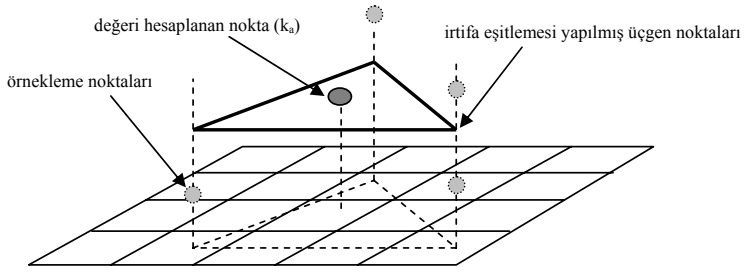
4.1.3. Nokta Tabanlı ve Zaman ilişkili Coğrafi Veriler

Nokta tabanlı ve zaman ilişkili coğrafi veriler kütüphanesi, tanımlanan bir dizi parametrenin zamana ve konuma bağlı olarak değişimini enterpolasyon teknikleri ile hesaplayabilen kişiselleştirilebilir bir veri yapısı sunar. Tanımlanan parametreler (örneğin nem, sıcaklık, bulut yüksekliği, akıntı yönü vb.), sabit bir değere sahip olabilir, benzetim sistemi zamanına bağlı olarak önceden belirlenmiş bir plan (değişim çizelgesi) dahilinde değişebilir veya zaman ile coğrafyaya bağlımlı olarak değerleri hesaplanabilir. Coğrafyaya bağımlı olarak değer tespiti için arazide belirli noktalar ile irtifalarda örneklenmiş ve ölçüm zamanları belirtilmiş parametre değerleri dizisi kullanılır. Bu değerler dizisi sayesinde, ölçümleri olmayan ara noktalar ve zamanların değerleri enterpolasyon teknikleri ile hesaplanır.

Ara değerlerin hesaplanması için, öncelikle farklı koordinatlardaki ölçüm noktaları Delaunay üçgenleştirmesi ile bir üçgen ağına dönüştürülür. Ardından değeri hesaplanacak koordinatın (k_a) denk düştüğü üçgen bulunur. Sonra, üçgenin

köşelerindeki noktaların, k_a noktası ile aynı irtifadaki değerleri enterpolasyon teknikleri ile hesaplanır. İrtifa eşitlemesi yapıldıktan sonra, k_a noktasının değeri, irtifası eşitlenmiş nokta değerleri ile yapılan ikinci bir enterpolasyon ile belirlenir (bknz. Şekil 5).

Bu veri kütüphanesi, katmanlardaki parametrelerin neler olacağı ve hangi verileri tutacağı konusunda her hangi bir kısıt getirmediği için bu parametrelerin belirli bir standarda oturtulması "Çevre Standartları Kütüphanesi" aracılığıyla sağlanır.



Şekil 5. İrtifa eşitlemesi ve üçgen enterpolasyonu ile ara değerlerin hesaplanması

4.2 Çevre Standartları Kütüphanesi

Bu kütüphane, çevre erişim kütüphanesinin sunduğu kişiselleştirilebilir veri yapılarının kullanımını standartlara bağlar (sayısal yükseklik verilerinin "Elevation" adlı bir alanda 4 byte float olarak tutulması, uydu fotoğrafı verilerinin "R", "G", "B" adlı alanlarda 1'er byte tam sayı olarak tutulması, nem parametresinin isminin "Nem", biriminin "%" olması, vb.). Çevre standartları kütüphanesi, yüzey, meteoroloji ve astronomi verileri olarak üç grupta ele alınmaktadır.

4.2.1. Yüzey Verileri

Kara ve su alanlarının yüzey/dip bilgilerine ilişkin raster harita standartları yüzey verileri kategorisinde ele alınır. MÜHATEM'in raster harita standartları; *sayısal yükseklik haritası*, *raster harita görüntüsü* (taranmış kağıt haritalar), *görünür ışık bandı görüntüsü* (uydu görüntüsü, vb.), *kızılötesi görüntü*, *raster yüzey örtüsü haritası* (bitki örtüsü), *yüzey normalleri haritası* (yüzey bakış yönü) ve *su derinlik haritası* olarak belirlenmiştir. Vektör haritalara ilişkin standartlar ise henüz netleşmemiştir.

4.2.2. Meteoroloji Verileri

MÜHATEM'in meteoroloji verilerine ilişkin standartların belirlenmesinde Devlet Meteoroloji İşleri Müdürlüğü ile koordineli çalışılmıştır. Meteorolojik bilgiler, Zamana dayalı parametre veritabanı kullanılarak saklanmış ve Türkiye'nin 2000-2004 yılları arasındaki ölçümlerine dayalı istatistikler çıkarılmıştır. Bu sayede atmosfer verilerinin uzun dönem ortalama ve standart sapma bilgileri elde edilmiş ve benzetim sistemlerinde kullanılmak üzere, enterpolasyon teknikleri ile farklı zaman ve konumlardaki parametre değerleri gerçekci bir şekilde tahmin edilebilmiştir.

MÜHATEM'in standart meteoroloji parametre ve birimleri; *sıcaklık* (°C), *mahalli basınç* (Mb), *buhar basıncı* (Mb), *nispi nem* (%), *görüş mesafesi* (m), *rüzgar hızı* (m/saat), *rüzgar yönü* (derece, kuzey 0 ve (+) açı saat yönünde), *bulut yoğunluğu* (kapalılık oranı, 0..1), *bulut yüksekliği* (m, deniz seviyesinden), *yağış türü* (0 = yağmur,

1 = kar, 2 = dolu) ve *yağış yoğunluğu* (mm) olarak belirlenmiştir. Sis yüzdesi, parametre olarak kullanılmamaktadır çünkü sis yatay görüş mesafesi parametresi içine dolaylı olarak ithal edilmiş durumdadır. Yatay görüş mesafesi parametresinin yağışsız bir durumu kapsadığı, yağışın görüşe etkisini içermediği varsayılmaktadır.

4.2.3. Astronomi Verileri

Astronomi verileri ay, güneş ve yıldızlarını durumunu içermektedir. MÜHATEM'de, güneş ve ayın konumu, durumu ve gökyüzü aydınlanmasına etkileri, verilen bir konum ve zamana (tarih, saat ve GTM) göre belirlenmektedir. Güneşin ve ayın gökyüzündeki konumu, *pusula açısı* (0 derece kuzey ve (+) açı saat yönünde artar) ve *yükseliş açısı* (0 derece ufuğa, 90 derece gökyüzüne ve -90 derece yer yüzüne doğru), ayın dumuru ise *aydınlık yüzey oranı* (0 ile 1 arası) ile ifade edilmektedir. Aydınlatma etkileri ise *genel aydınlanma* (ambient) ve güneş ile ay için *doğrudan aydınlatma* (diffuse) olarak üç değer ile belirlenir. Genel aydınlanma, çevrenin ortalama aydınlanma miktarıdır ve güneş ile ay ışığının doğrudan vurmadığı noktalarda bile yansımalar dolayısıyla en az genel aydınlanma kadar bir ışığın olacağı varsayılır. Doğrudan aydınlatma ise ışık kaynağından gelen ışığın yüzeye dik olarak bir engele raslamadan çarptığı durumda meydana gelen aydınlanma miktarıdır. Toplam aydınlanma değerinin 1 olduğu noktalarda, yüzey rengi en parlak halinde (örneğin uydu görüntüsünün orjinal renkler tonlarında) görülür.

4.3 Çevre Algoritmaları Kütüphanesi

Bu kütüphane, Çevre Standartları Kütüphanesi tarafından sunulan veri yapıları üzerinde çalışan bir dizi algoritma sunar. Algoritmalar, nokta tabanlı, çizgi tabalı, alan tabanlı ve diğer olmak üzere dört kategoride ele alınır. Nokta tabanlı algoritmalar, belli bir noktadaki yükseklik, derinlik, bitki örtüsü gibi verileri okuyan fonksiyonları, çizgi tabanlı algoritmalar, iki nokta arasındaki görüş hattı testi, profil çıkarma gibi işlemleri yapan fonksiyonları ve alan tabanlı algoritmalar, pencereleme mekanizmaları kullanımı ile hızlı alan yükleme/kaydırma, bentler ile bölgeyi su doldurma, kapsama alanı hesaplama gibi çeşitli işlevleri sunmaktadır.

4.4 Çevre Güncelleme Kütüphanesi

Bu kütüphane, Çevre Standartları Kütüphanesinde tanımlanan veri standartlarına uyumlu çevre verilerinin kaynak veri dosyalarından (DEM, DTED, Geo-Tif, vb.) yüklenerek veritabanının güncellenmesini sağlamaktadır.

5. VARLIK KÜTÜPHANELERİ

Varlık kütüphaneleri, varlıkları tanımlamak ve üç boyutlu modellerini oluşturmak için gerekli her türlü işlevi içermektedir. Varlıklar iki farklı yolla tanımlanabilmektedir. Bunlardan ilki, *harekat alanı kavramsal modeli katmanı* ile tanımlama, ikincisi ise *varlık erişim katmanı* ile tanımlamadır.

Kavramsal model katmanı kullanıldığında, varlıkların öznitelikleri ve diğer varlıklarla olan ilişkileri kavramsal olarak ifade edilir. İki yada daha çok varlık arasında cümleler şeklinde ilişkiler tanımlanması ve bu ilişkilerin detaylı olarak sorgulanabilmesi mümkündür. Bu yetenek fazla kullanılmamaktadır.

Varlık erişim katmanı ise temsili ortamda yer alan varlıkların çalışabilir modellerinin oluşturulabilmesi için geliştirilmiştir. Varlık erişim katmanlarının en önemli özelliği eklenti (plug-in) mekanizması ile sınırsız genişleyebilme kabiliyetidir. Bu kabiliyet sayesinde, hem MÜHATEM geliştirici ekibinin hem de diğer kurum/kuruluş ve firmaların, MÜHATEM'e yeni ve özelleşmiş varlık tipleri ekleyebilmesi veya eskiden tanımlanmış varlıkları değiştirebilmesi mümkün olmaktadır. Hali hazırda, model, çizim, sembol, resim, ışık, ses, parçacık efektleri gibi özellikleri sunacak eklentilerin geliştirilmesine devam edilmektedir. Varlık erişim kütüphanesi ile eklenti alt yapısı;

- *TMhtmObjectLayerContainer*,
- *TMhtmObjectLayer*,
- *TMhtmObject* ve
- *TMhtmObjectResource* adında dört temel nesne sınıfından oluşmaktadır.

Söz konusu nesne sınıflarından son ikisi olan *TMhtmObject* ve *TMhtmObjectResource*'dan kalıtım yolu ile yeni nesneler türetilmesi ve DLL (Dynamic Link Library)'ler yolu ile varlık erişim katmanına (*TMhtmObjectLayer*'a) tanıtılması mümkündür.

5.1 TMhtmObjectLayerContainer

TMhtmObjectLayerContainer, varlık katmanlarının listesini barındıran ve bu katmanları ekleme, silme, arama, yükleme, kayıt etme gibi işlemler yapabilen nesne sınıfıdır. Görevi sadece bir dizi katmanı bir arada tutabilecek bir depolama ortamı sunmak ve katmanlara biricik (unique) tanımlama numaraları (ID'ler) atamaktır.

5.2 TMhtmObjectLayer

TMhtmObjectLayer, bir dizi varlığı (tank, uçak, duvar, bina, ışık, algı cihazı, silah, sembol, yazı, vb.) barındırabilen katmanların yaratıldığı nesne sınıfıdır. Katmanlara varlıklar eklenebilir, varlıkların birbirine kalıtım ve parça-bütün ilişkileriyle bağlanması sağlanabilir ve katmana atanan simülasyon koşumu başlangıç ve anlık tarih/saati ile tanımlanmış varlık davranışlarının zamana dayalı olarak ilerletilmesi sağlanabilir. Katmanlar başka katmanlara bağlantı kurarak bu katmanlardaki varlıklar ile kendi içlerindeki varlıklar arasında kalıtım ilişkileri kurabilir. Örneğin "Classes" adlı bir katman ile benzetim sisteminde kullanılacak varlık tiplerini (sınıflarını) tanımlamak ve "Instances" adlı ikinci bir katman ile "Classes" katmanına bağlanarak buradaki varlık tiplerinin kopyalarını "Instances" katmanına aktarmak, böylece bir senaryo oluşturmak mümkündür. Başka bir varlıktan kopya çıkarılırken, kopyası çıkarılan varlığa ek olarak kopyalanan varlık üzerinde yer alan diğer varlıkların da (örneğin bir uçağın üzerindeki pervaneler) toplu olarak kopyalanması mümkün olmakta, benzer şekilde bir varlık silindiğinde ona bağlı tüm alt varlıkların silinmesi de sağlanabilmektedir.

5.3 TMhtmObject

TMhtmObject, varlıkların türetildiği temel eklenti sınıfıdır. Bu sınıf, işlevleri devir alınarak değiştirilebilir bir dizi *virtual* metod sunmakta, böylece farklı işlevlere sahip eklentilerin oluşturulması mümkün olmaktadır. Katmanlara eklenecek varlıklar,

TMhtmObject sınıfından türetilmiş eklentiler ile yaratılabilmekte, varlıkların isim, açıklama ve coğrafi konumları tanımlanabilmekte ve varlıklar başka varlıklarla kalıtım ve parça-bütün ilişkileri kurabilmektedir. Varlıkların coğrafi konumları (koordinat ve doğrultuları), nokta, çoklu nokta, çoklu çizgi, kapalı uçlu çoklu çizgi veya çokgen olarak tanımlanmakta ve bir varlık başka bir varlığa bağlanarak onunla birlikte hareket edebilmektedir. Varlıklar, kendilerinin eklenti olmasına ek olarak, varlık özelliklerini genişletmeyi amaçlayan bir dizi alt eklenti de (kaynak eklenti - resource plug-in) içerebilmektedir. Kaynak eklentiler, müteakip bölümde anlatılmaktadır.

5.4 TMhtmObjectResource

TMhtmObjectResource, varlıkların şekil, sembol, ışık, algılama, yorumlama, haberleşme gibi özelliklerini belirleyen eklentilerin (kaynak eklentilerin) türetildiği nesne sınıfıdır. Bu eklentiler genişleyebilirliğin sağlanmasında varlık eklentilerine oranla çok daha fazla öneme sahiptir çünkü varlık çeşitliliği ancak varlık eklentilerinin sayısı kadar olabilir ve bir varlık aynı anda birden fazla eklenti sınıfından türetilemez. Bununla birlikte tek bir sınıftan türetilmiş varlık, farklı kombinasyonlardaki kaynak eklentileri ile çok farklı özellikte varlığa dönüşebilir.

6. GÖRSELLEŞTİRME VE ARAYÜZ KONTROLLERİ KÜTÜPHANESİ

Görselleştirme Kütüphanesi, raster/vektör haritalar, arazi, atmosfer, varlıklar, karar ağaçları gibi çeşitli verilerin görselleştirilmesine yönelik sınıflar sunmaktadır. Arayüz Kontrolleri Kütüphanesi ise görselleştirme kütüphanesini kullanarak pratik 2/3 boyutlu ortam görşelleştirmesi yapabilen ve etkileşimli şekilde pencere görünüm bilgilerini (harita konumu ile yakınlığı, koordinat çizgileri vb.) değiştirebilen nesne sınıflarını içermektedir. Bu nesne sınıfları, programlama platformu bağımlı olacak şekilde doğrudan kullanıcı arayüzlerine erişim yapmamakta ancak arayüzlerinden çeşitli metodlar ile bilgiyi alıp kullanıcı arayüzlerine callback fonksiyonları ile geri bildirimler sağlamaktadır.

7. SONUÇ

Bu bildiride, ODTÜ-TSK MODSİMMER tarafından 2002 yılından beri geliştirilmesine devam edilen MÜHATEM Projesi kapsamında oluşturulmuş yazılım programlama kütüphanesi kısaca tanıtılmıştır. Bu projenin temel amacı, tüm modelleme ve simülasyon projelerinde ortak ihtiyaç olan temsili ortam modelinin, tek bir alt yapı tarafından sunulmasıdır. Kütüphanenin ara sürümleri, "Küçük Ölçekli Harekatın Modellenmesi ve Simülasyonu (BASKIN)" ile "Sınır Gözetleme ve Kontrol Sistemi C4ISR Faaliyetlerinin Modellenmesi ve Simülasyonu (SGKS C4ISR-MOS)" projelerinde kullanılmış, "Müşterek Görev Kuvvetinin Modellenmesi ve Simülasyonu (MGKMOS)" ile "Satıhta Konuşlu Aktif Hava Savunma Harekatının Modellenmesi (SAMMOS)" projelerinde kullanılmakta ve "Deniz Harp Oyunu Simülasyon Sistemi (DEHOS)" projesinde kullanımı planlanmaktadır.

KAYNAKÇA

[1] ODTÜ-TSK MODSİMMER, (2006), "MÜHATEM Yazılım Geliştirici Kılavuzu", ODTÜ TSK Modelleme ve Simülasyon Araştırma ve Uygulama Merkezi.

[2] Brent Welch, Ken Jones, and Jeff Hobbs, (2003), "Practical Programming in Tcl and Tk", 4th Edition ISBN: 0-13-038560-3.

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

YÜKSEK ÇÖZÜNÜRLÜKLÜ ARAZİ VERİLERİNİN GRAFİK İŞLEMCİSİ KULLANILARAK GÖRSELLEŞTİRİLMESİ

H. Billur ENGİN [(a)], Emre KOÇ [(a)], Selçuk SÜMENGEN [(a)] , Selim BALCISOY [(a)]

[(a)]Sabancı Üniversitesi,

Mühendislik ve Doğa Bilimleri Fakültesi Bilgisayar Grafikleri Laboratuvarı,

{billurengin, emrekoc, selcuk}@su.sabanciuniv.edu,

balcisoy@sabanciuniv.edu

ÖZ

Grafik İşlemcilerinde (Gİ) görülen hızlı ilerleme, geniş ve detaylı arazi verilerinin gerçek zamanlı benzetim sistemlerinde etkin olarak kullanılmasına olanak sağlamaktadır.

Bu makalede, geliştirdiğimiz üç boyutlu arazi görselleştirme motorunu ve İstanbul ili coğrafi yükseklik verilerini kullanarak elde ettiğimiz sonuçları sunuyoruz. Araştırmalarımız sırasında gerçek zamanlı arazi görselleştirilmesi probleminde karşılaşılan sorunlara mevcut yöntemlerin geliştirilmesiyle çözümler bulunmuştur. İşlem yükünün Gİ'ye kaydırılması sonucunda Merkezi İşlem Birimi'nin (MİB) yükü düşürülmüş ve benzetime eklenebilecek farklı işlevlere imkan yaratılmıştır.

Anahtar Kelimeler: Arazi Görselleştirme, Detay Seviyesi, Grafik İşleme Ünitesi, Kesik Geometrik Haritalar, Üçgenleme Metodu.

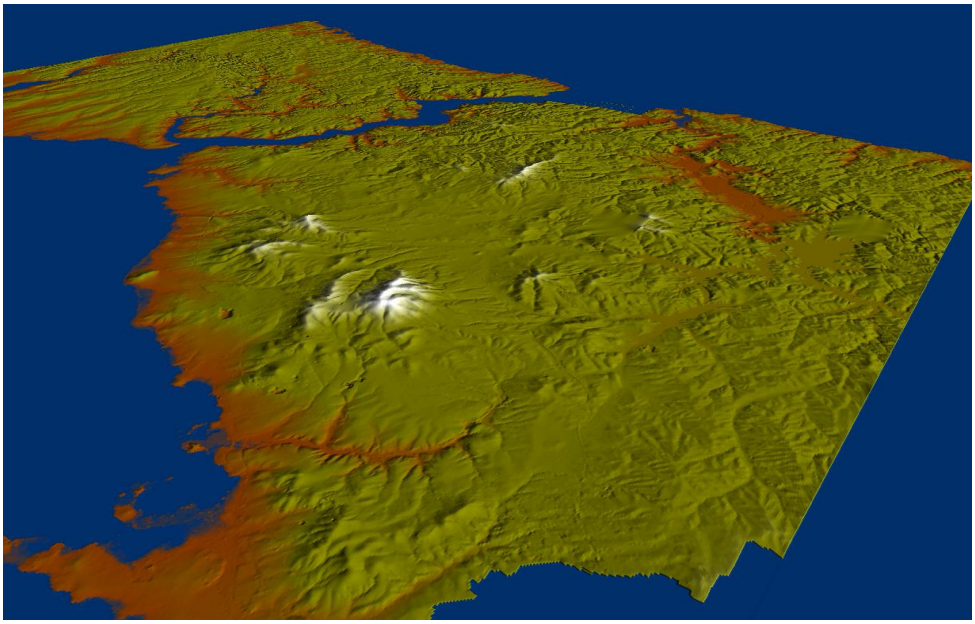
ABSTRACT

Rapid progress in graphics hardware enables integration of large scale and detailed terrain data into real-time simulation systems. In this paper we present the 3D terrain engine and our test results of İstanbul's elevation data. We propose several solutions to rendering of large terrain data by improving existing approaches. Our technique involves shifting computationally complex operations to Graphics Processing Unit (GPU) and saving Central Processing Unit (CPU) resources for computationally intense simulation tasks other than terrain rendering.

Keywords: Clipmaps, GPU, Level of Detail, Terrain Rendering, Triangulation.

1. GİRİŞ

Geniş coğrafi alanların büyük ölçekte görselleştirilmesi söz konusu olduğunda kalabalık veri kümelerinin ele alınması gerekmektedir. Sadece İstanbul ilinin 25 metre aralıklarla alınmış yükseklik verisi 6.153.992 noktadan oluşur (Şekil 1). Bu sayı bütün Marmara Bölgesi' nin veya Türkiye' nin görselleştirilmesi söz konusu olduğunda çok daha büyük değerlere ulaşır. Mevcut donanımlar yüksek çözünürlükteki geometrik yapıları görselleştirebilme kapasitesine sahiptirler, ancak milyonlarca yükseklik ölçümünden oluşan veri kümelerinin işlenmesiyle oluşturulacak benzetimde, gerçek zamanı yakalayabilmek için görselleştirilen araziye ait geometrinin farklı detay seviyelerinde çizilebilmesi gerekmektedir. Çalışmamız sırasında uyguladığımız bu yaygın yöntem için farklı yaklaşımlar mevcut ise de ortak amaç, gerçek zamanlı görselleştirme sırasında mümkün olduğu kadar geometrik detayı korumak ve sayısal olarak örneklenmiş arazi yapısını en doğru şekli ile üç boyutlu ortama taşıyabilmektir. Bu çalışmanın sonucu olarak sunduğumuz yöntem, son zamanlarda büyük ilerleme kaydeden Grafik İşlemcileri' nin arazi görselleştirme sırasında en etkin şekilde kullanımına yöneliktir.



Şekil 1. İstanbul' un 25 metrelik arazi verisinin gerçek zamanda üç boyutlu görselleştirilmesi.

Not: İstanbul Boğazı' nın kuzey doğusunda ve Adalar' da veri kümesinden kaynaklanan bir eksiklik vardır.

2. ARAZİ GÖRSELLEŞTİRME

2.1. İlgili Çalışmalar

Arazi görselleştirme teknikleri genel olarak ele alındığında, arazinin ekranda çizilebilmesi için, öncelikle sabit aralıklarla alınmış yükseklik verilerinden yüzey bilgilerinin elde edilmesi gerekir. Yüzey oluşturmak için genellikle üçgenleme yöntemi kullanılır. Ekranda görünen bütün bölgeleri aynı detay seviyesinde üçgenlemek benzetimi yavaşlatır, bu yüzden araziyi farklı çözünürlük bölgelerine ayırarak çokgen sayısını azaltmak gerekir. Ayrıca arazinin tamamını aynı anda çizmek yerine, sadece kameranın görüş alanı içerisindeki arazi parçalarını çizim hattına göndermek sisteme hız kazandıracaktır. Bu işlemin kritik olan kısmı hiç tartışmasız kameranın görüş alanı değiştikçe, uygun parçaların en hızlı şekilde seçilerek, sınırları belli olmayacak bir şekilde birbirlerine harmanlanmasıdır.

Arazi görselleştirme konusunda şu zamana kadar geliştirilmiş pek çok algoritma vardır. Bunları şu başlıklar altında toparlayabiliriz:

Düzensiz Ağlar: Üçgenlere ayrılmış düzensiz ağlar olarak da bilinirler (TIN)[1]. Üçgen oluşturmak üzere Delaunay üçgenleri yaygın olarak kullanılmaktadır[2].

İkili Ağaç Hiyerarşileri: İkizkenar dik üçgenlerin kenar ortaylarından ikiye ayrılarak bölünmesi esasına dayanır[3].

İkili Ağaç Alanları: Ağaç yapısının yardımıyla geniş alanlı arazilerin geometrik bilgi olarak saklanması üzerine kuruludur.Parça alanlar önceden üçgenlere ayrılarak tampon bölgede tutulur. Parça aralarında yumuşak geçişi sağlamak için aradeğerleme teknikleri kullanılır. İkili ağaç hiyerarşilerinden daha az detaylıdır[4].

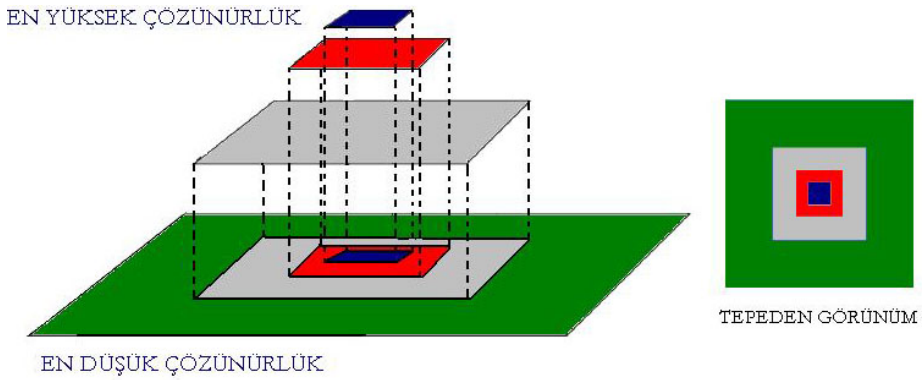
Kesik Geometrik Alanlar: Bu teknik, blok döşeme yöntemleriyle benzerlik göstermektedir[5]. Bloklar halinde saklanan yükseklik verisi, gözlemci odaklı bir ağaç üzerine yerleştirilmiştir. Kaplamaların farklı odak noktalarıyla ve çoklu detay bilgisiyle çizilebilmesinde kullanılan kesik harita yönteminin, yükseklik bilgileri kullanılarak arazi görselleştirilmesi konusuna adapte edilmiş halidir[6].

Üçgen başına düşen hata oranı her kategoride biraz daha artmaktadır[7]. Düzensiz ağlar gerçeğe en yakın sonucu verir ancak işlemsel yükü çok fazladır. Kesik geometrik harita yönteminde hesaplama yükünün Gİ'ye taşınmasıyla, hızlı ve MİB'nin hesap yükünü önemli ölçüde azaltan yöntemler geliştirilmiştir. [8,9,10,11]

2.2 Kesik Geometrik Haritalar

Kesik geometrik haritalar yöntemi, yükseklik bilgilerinin hangi detay seviyesinde üçgenleneceğini, üçgenlerin kameraya olan mesafesine bağlı bir fonksiyon haline getirir. Kameraya yakın olan bölgeler çok detayla, uzak olan bölgelerse daha az detayla çizilir.

Bu yöntem araziyi iki boyutlu yükseklik verisi olarak ele alır. Ekrandaki görüntünün farklı çözünürlük seviyelerinde çizileceği kabul edilirse, kamerayı merkeze oturttuktan sonra, en yüksek çözünürlüklü bölge kare şeklinde, daha sonra gelen (düşük çözünürlüklü) bölgeler ise bu karenin etrafını çevreleyen şeritler şeklinde çizilirler (Şekil 2).



Şekil 2. Kesik Geometrik Harita

2.3. Grafik İşlemcisi ve Merkezi İşlem Birimi Karşılaştırması

Son yıllarda ivme kazanan Gİ teknolojisinin gelişimi, görselleştirme ve benzetim alanında çalışan araştırmacıların geleneksel yöntemler ve algoritmaları gözden geçirmelerine neden olmuş, çok yüksek veri işleme kapasitesine sahip Gİ'nin mimarisi üzerinde çalışabilecek yeni yöntemler geliştirmelerini sağlamıştır. Gİ, aynı anda çok sayıda veriyi işleyebilecek paralel bir mimariye sahiptir. Bu tasarım birbirinden bağımsız olarak çoklu veriyi işlemek için oldukça elverişli iken, özyinelemeli (rekürsif) algoritmalarda verimli değildir. Bu nedenle Gİ kullanılarak yapılan arazi görselleştirmelerinde, dörtlü ağaç yapıları gibi birbirine bağımlı ve özyinelemeli algoritmalar tercih edilmez. Bu yapılar kullanılarak oluşturulan düzensiz üçgenlemeler ile detay seviyesini düşürmek yerine doğrudan kamera uzaklığına bağlı olarak detaylandırılmış düzenli çizgi dizilimleri (grid) geometrisi kullanılır. Burada amaç üçgen sayısını olabildiğince azaltmaktan çok, belirli bir seviyenin altında muhafaza etmek ve Gİ'ye doğru gerçekleşen veri transferini en iyi düzeyde tutmaktır[8].

3. ARAZİ GÖRSELLEŞTİRME YAZILIMI

Laboratuvarımızda geniş coğrafi bölgeleri görüntüleyebilmek için, yapılan çalışmalar sonucunda bir arazi görselleştirme yazılımı geliştirilmiştir. Sistemin çalışma mekanizması ve bazı kritik sorunlara bulduğumuz çözümler şöyledir:

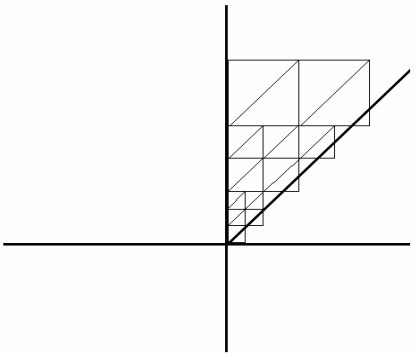
3.1 Yükseklik Bilgisinin Grafik Kartına Gönderilmesi

Sabit aralıklarla alınmış yükseklik örneklerini MİB'de tek boyutlu diziler halinde saklamak yerine, bu örnekler iki boyutlu kaplama dizilerinde saklanarak Gİ'ye gönderilir[8]. Böylelikle x ve y koordinatları bilinen noktaların yükseklik bilgilerine rahatça ulaşılabilir. Kaplama olarak tanımlanan bilgi üzerinde Gİ'nin direk erişimi olduğu için, hesaplamalar daha kolay ve hızlı yapılır. Gİ'nin yeni özelliklerinden biri olan "GL_TEXTURE_RECTANGLE_NV" komutu ile istenilen en ve boyda kaplamalar oluşturulabilir. Yani dikdörtgen şeklindeki araziler kare şeklindeki dizinlere sığdırılmak zorunda kalmadan işleme sokulabilirler.

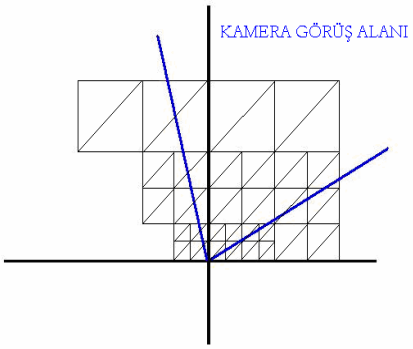
Başlangıçta MİB'den Gİ'ye ekrana çizilen maskenin iki boyutlu dizin koordinatları gönderilir. Kameranın üzerinde bulunduğu koordinat bilgisi sürekli yenilenir. Gİ' de bulunan kamera ve dizin koordinatları kullanılarak iki boyutlu kaplama bilgisinden her bir noktaya denk gelen yükseklik bilgisine rahatlıkla ulaşılabilir. Bu sebeple MİB'den Gİ' ye kaplama bilgisi dışında yükseklik bilgisi gönderilmesine gerek yoktur. Çizilen maskeye ait iki boyutlu koordinat bilgisinin yanısıra Gİ'ye gönderilen z değeri yükseklik yerine, o noktanın hangi detay seviyesinde çizileceğini ve aşağıda anlatılan "detay seviyeleri arasindaki yumuşak geçiş"in kaçıncı adımında olduğunun belirlenmesini sağlar. Bu sayının tam kısmı detay seviyesi için, küsüratlı kısmıysa yapılacak olan aradeğerleme için bilgi vermektedir.

3.2 Kesik Geometrik Haritaların Yerleştirilmesi

45 derecelik bir kesik geometrik harita şablonu MİB'de hazırlanır (Şekil3). Daha sonra kameranın bakış yönü sağa ve sola doğru değiştikçe önceden hazırlanmış şablon kameranın etrafında döndürülerek görüntünün bütünlüğü sağlanır. Harita parçaları arasındaki değişimin ekranda görülmemesi için kameranın 50 derecelik görüş alanı içerisindeki bölüm 45 derecelik harita parçalarıyla doldurulur. Bu doldurma işlemi en çok 135 derece (üç harita parçası), en az 90 derecedir (iki harita parçası) (Şekil4).



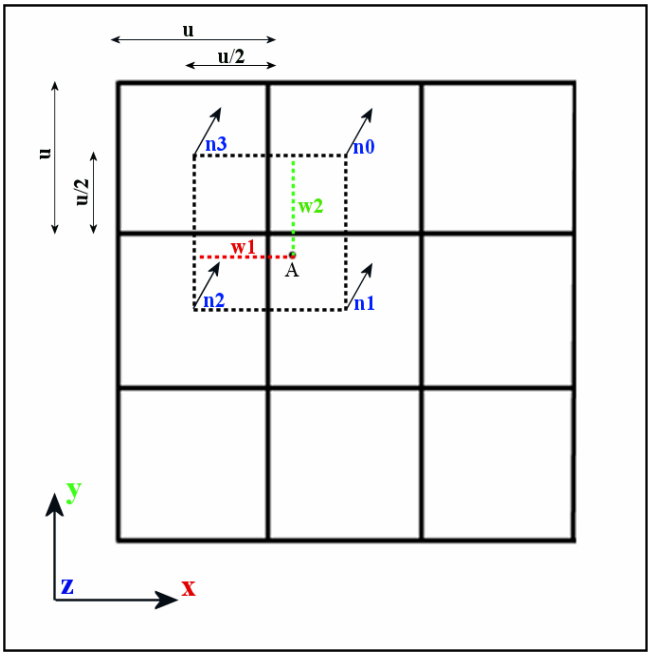
Şekil 3. 45 Derecelik Kesik Geometrik Harita Şablonu



Şekil 4. 135 Derecelik Kesik Geometrik Harita Şablonu

3.3 Normal Hesaplama

Gİ'de yüzey normalleri hesabı her bir piksel için tekrarlanır ve piksel normalleri bulunur, bu sebeple elde edilen görüntüler MİB'de yapılan normal hesaplarından daha detaylı sonuçlar verir. Ekrana çizdirilen bölgenin detay seviyesi ne olursa olsun normal hesabı hep en yüksek çözünürlükten bulunmuş olur.



Şekil 5. Normal Hesaplanması

İlk önce normali hesaplanacak pikselin coğrafi koordinatları (Şekil 5: A noktası) üzerinden iki boyutlu yükseklik verileri içerisinde karşılık geldiği en yakın çizgi dizilimi noktası bulunur. Ardından Newell yöntemiyle [12] bu noktanın etrafında yer alan dört dörtgen yüzeyden herbirinin normalleri (Şekil5: n1, n2, n3, n4) bulunduktan sonra, bu noktaya uzaklıklarına göre ağırlıklı ortalamaları (lineer interpolasyon) alınarak A noktasının normali bulunur (Denklem 1).

$$\begin{aligned} \vec{A}_{n3n0} &= (w_1/u) * \vec{n}_0 + (1 - w_1/u) * \vec{n}_3 \\ \vec{A}_{n2n1} &= (w_1/u) * \vec{n}_1 + (1 - w_1/u) * \vec{n}_2 \\ \vec{A}_n &= (w_2/u) * \vec{A}_{n2n1} + (1 - w_2/u) * \vec{A}_{n3n0} \end{aligned} \qquad (1)$$

$$0 \le w_1,\ w_2 \le u$$

u = birim düzenli çizgi dizilimleri uzunluğu

$\vec{n}_0,\ \vec{n}_1,\ \vec{n}_2,\ \vec{n}_3$ = normal değerleri

3.4 Yükseklik Değerlerine Göre Renk Atama

Program, yükseklik verilerini okuma aşamasındayken değerlerin ortalamasını ve standart sapmasını da hesaplar. Bu değerler Gİ' ye gönderilerek en küçük ve en büyük

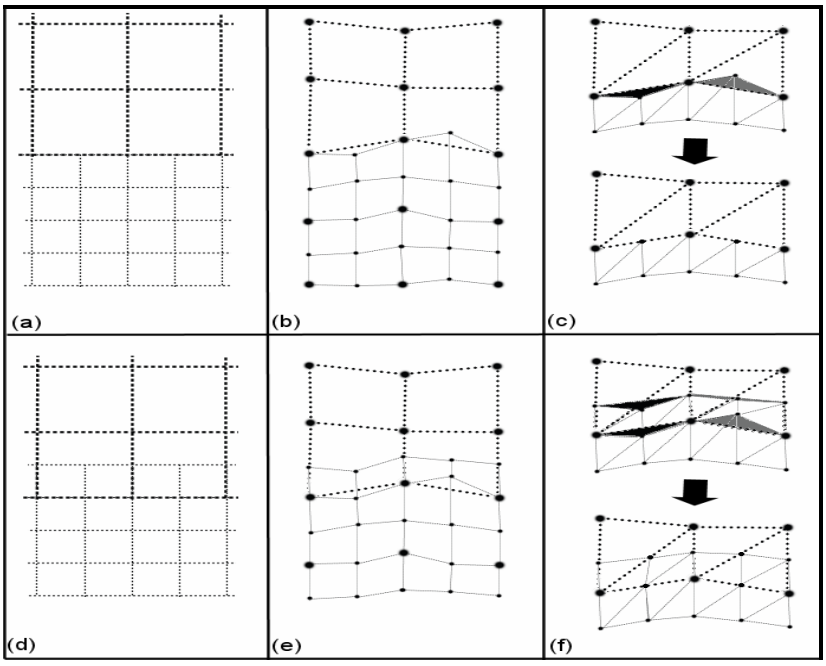
yükseklik değerleri arasında değişen bir harita lejantı oluşturulur. Gİ' nin piksel işlemcisi her bir piksele bu lejanta uygun rengi atar.

3.5 Arazi Üzerinde Hareket Etme

Arazi üzerinde harekete başlandığında kameranın yerini değiştirmek yerine, kesik haritaları oluşturan noktaların yükseklikleri değiştirilir ancak bu değişim tek bir adımda yapıldığında istenmeyen bir görüntü meydana gelir (pop up). Bu sebeple yükseklik değişimleri gözlemcinin her adımında piksel başına aradeğerleme yöntemiyle hesaplanır. Böylelikle yavaş ve yumuşak bir geçiş elde edilmiş olur.

Gözlemci iki üçgen arası mesafeyi kat edene kadar attığı adımlar boyunca kesik haritalar geriye doğru çekilir, gözlemci diğer üçgene ulaştığı zaman kesik harita eski yerine geri döndürülür ve kaplamadan yeni yükseklik bilgileri alınarak kesik haritanin x ve y değerleriyle eşleştirilir.

3.6 Seviye Geçiş Bölgelerinde Arazinin Bütünlüğünün Korunması



Şekil 6. Seviye geçişleri

T birleşim noktaları (T-junction), farkli detay seviyesine sahip kesit haritaları arasındaki geçiş sırasında oluşur. Detay seviyesi yüksek olan kesit haritası her x birimde bir örnekleniyorsa, detay seviyesi düşük kesit haritası her 2x birimde bir örneklenmiştir. Daha önce anlatılan arazi üzerinde hareket etme mekanizması dolayısıyla, kamera hareketi detaylı kesit haritası x birim ilerleme sonrasında kaydırılırken, daha az detaylı kesit haritası ise 2x birimde bir kez kaydırılacaktır. Bu kaydırma sonucunda iki ayrı durum ortaya çıkar

Detaylı kesit haritası sınırındaki her iki örnekten birine, daha az detaylı kesit haritasında karşılık gelen bir yükseklik verisi yoktur (Şekil6a). Bu durumda eğer karşılık bulamayan noktaların yüksekliği komşu noktaların ortalamasından farklı ise üçgenleme sırasında o bölgede bir açıklık ortaya çıkar (Şekil6b). Bu açıklığın giderilmesi için daha az detaylı kesit haritasında karşılık bulamayan noktanın yüksekliği, komşu noktaların ortalaması alınarak tekrar belirlenir(Şekil6c).

İkinci koşulda ise detaylı kesit haritasının x birim kenar uzunluğundaki alanı daha az detaylı kesit haritasının içinde kalır(Şekil6d). Bu durumda yükseklikleri tekrar hesaplanması gereken noktalar daha fazladır(Şekil6e). Bu noktaların yükseklikleri, kendilerine karşılık gelen daha az detaylı kesit haritasının komşu noktaları alınarak tekrar hesaplanır (Şekil 6f). Bütün bu hesaplamalar grafik kartı üzerinde yapılmaktadır.

4. SONUÇLAR ve DEĞERLENDİRME

Çizim işlemleri Gİ'de yapıldığı için, yazılım MİB'nin tüm kapasitesini kullanmak yerine sadece %20'sini kullanarak çalıştırmaktadır. MİB' nin hesaplama yükü oldukça azaltıldığı için benzetimde yapılabilecek olası diğer işlemlere de (yapay zeka, fizik motoru) olanak sağlanmaktadır.

Hesaplamaların Gİ'de yapılmasının sağladığı diğer bir avantaj ise gerçek zamanlı bir benzetim elde edilmiş olmasıdır. Arazi görselleştirme yazılımımız etkileşimli hızlarda çalışmaktadır (Çizelge 1).

Kaplamaları oluştururken "GL_TEXTURE_RECTANGLE_NV" özelliğini kullandığımız için yazılımın çalıştırılacağı bilgisayarın "NVIDIA GeForce" 6, 7 veya 8 serisi grafik kartlarından biri tarafından destekleniyor olması gerekir. Gİ'ye göndereceğimiz verinin üst sınırı kaplama başına 4096 x 4096 noktadır. Bu değerden daha fazla nokta içeren yükseklik verilerinin Gİ'ye parçalara bölünerek yüklenmesi gerekmektedir.

Şu an için geliştirmiş olduğumuz sistem, bizi gerçek zamanlı görselleştirme konusunda hedefimize ulaştırmış bulunmaktadır. Gelecekte birden fazla parçayla çok daha geniş coğrafi alanların büyük ölçekle görselleştirilmesi ana hedefimizdir. Bunun yanında arazinin üzerinin yüksek çözünürlüklü uydu resimleri ile kaplanması, binaların ve yolların coğrafi bilgi sistemlerinden edilinecek veriler ile çizilmesi konularında çalışmalarımız sürmektedir.

Arazi Boyutları (km^2)	**Gösterim için Kullanılan Normal Vektörü Sayısı**	**Çizilen Üçgen Sayısı**	**Görüş Alanı**	**Çerçeve / saniye (fps)**
1110,88	444.352	6.260	150°	12.82
894,72	357.888	5.052	120°	15.87
665,12	266.048	3.748	90°	21.27
458,96	183.584	2.608	60°	32.25
347,52	139.008	1.980	45°	41.99

Çizelge 1. Görselleştirme Motoru Başarım Analizi Tablosu.

KAYNAKÇA

[1] Evans, W., D. Kirkpatrick, and G. Townsend., "Right Triangular Irregular Networks", Algorithmica, Springer- Verlag New York, LLC, Special Issue on Algorithms for Geographical Information ,2001.

[2] Daniel Cohen-Or and Yishay Levanoni, "Temporal continuity of levels of detail in delaunay triangulated terrain," in IEEE Visualization '96, Roni Yagel and Gregory M. Nielson, Eds., San Francisco, California, Oct. 1996,pp. 37–42, IEEE.

[3] Renato, P., M. Antonijuan and R. Lario., "QuadTIN: Quadtree based Triangulated Irregular Networks".

[4] Levenberg, J. "Fast view-dependent level-of-detail rendering using cached geometry.", In Proceedings of the 13th IEEE Visualization 2002 Conference, Boston, 2002.

[5] Losasso, F. and Hoppe, H. "Geometry Clipmaps: terrain rendering using nested regular grids", ACM Transactions on Graphics, ACM Press, Volume 23, 2004-10-13.

[6] C. C. Tanner, C. J. Migdal, and M. T. Jones, "The clipmap: a virtual mipmap," *Proceedings of the 25th annual conference on Computer graphics and interactive techniques,* pp. 151-158, 1998.

[7] M. Clasen and H. C. Hege, "Terrain Rendering using Spherical Clipmaps" , EuroVis 2006 – Proc. Eurographics / IEEE VGTC Symposium on Visualization, pp. 91-98, 2006.

[8] A. Asirvatham and H. Hoppe, "Terrain rendering using GPU-based geometry clipmaps," *GPU Gems,* vol. 2, pp. 109–122, 2005.

[9] A. Brodersen, "Real-time visualization of large textured terrains," in *Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia* Dunedin, New Zealand: ACM Press, 2005.

[10] J. Schneider and R. Westermann, "GPU-Friendly High-Quality Terrain Rendering," *Journal of WSCG,* vol. 14, pp. 49-56, 2006.

[11] Y. Livny, N. Sokolovsky, T. Grinshpoun, J. El-Sana, and I. Beer-Sheva, "Persistent Grid Mapping: A GPU-based Framework for Interactive Terrain Rendering."

[12] F.S. Hill, Computer Graphics using OpenGL , pp. 347- 350, 2001.

SAVUNMA SİSTEMLERİ TASARIMINDA İNSAN BİLGİSAYAR ETKİLEŞİMİ ÇALIŞMALARININ ROLÜ

Gökçe YILDIRIM [(a)], İsmail BIKMAZ [(a)], Kürşat ÇAĞILTAY [(b)], Veysi İŞLER [(c)]

[(a)] Simsoft, ODTÜ KOSGEB-TEKMER,

{gokce, İsmail}@simsoft.com.tr

[(b)] ODTÜ Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü, kursat@metu.edu.tr

[(c)] ODTÜ, Bilgisayar Mühendisliği Bölümü, isler@metu.edu.tr

ÖZ

Askeri sistemler kritik ve hayati görevlerde kullanılmaları nedeni ile bunları kullanan personelin en kısa zamanda, en etkili kararı, en az hata ile vermesi gerekmektedir. Ancak bu sürec gerek fiziksel ve gerekse bilişsel unsurlardan ciddi şekilde etkilenmektedir. Eğer askeri sistemler bu unsurlar gözönüne alınmadan tasarlanırsa can kaybına kadar ulaşan problemler ortaya çıkabilmektedir. Bu bildiride özellikle İnsan Bilgisayar Etkileşimi ve bunun Askeri sistemlerin tasarımı ve geliştirilmesindeki önemi sunulacak, bazı durum çalışmalarından örnekler verilecektir.

Anahtar Kelimeler: Algı, İnsan Bilgisayar Etkileşimi, İnsan Faktörü, Kullanılabilirlik, Savunma Sistemleri,

CRITICAL ROLE OF HUMAN COMPUTER INTERACTION IN THE DESIGN OF DEFENSE SYSTEMS

ABSTRACT

Since military systems have life critical mission, the personnel must use them with minimum error, in the shortest time and they have to give the most effective decision. However, these processes are affected by several physical and cognitive factors. If defense systems are designed without considering those factors, the results could be very catastrophic. In this paper we will present basics of Human Computer Interaction and its importance in the design and development of defense systems. Moreover, we will also present some real life case studies.

Keywords: Defense Systems, Human Computer Interaction, Human Factors, Perception, Usability

1. GİRİŞ

Askeri sistemlerin kullanılabilirliği konusundaki ciddi çalışmaların başlangıcı 2. Dünya savaşı esnasında özellikle savaş uçaklarının kabinlerinin daha basit hale getirilmesi çalışmalarına dayandırılmaktadır. Yapılan çalışmalar ve düzenlemeler sonucunda B-17 ve P-51 Mustang gibi uçakları yeni eğitilen pilotlar daha kolay öğrenmeye ve kullanmaya başlamıştır. Bilişim sistemlerinin özellikle 1970'lerin sonlarından sonra daha da artan seviyelerde yaşamın tüm diğer alanları gibi savunma sistemlerine de girmeye başlaması ile ilgili personelin en az hata ve en yüksek başarım ile bu sistemleri nasıl kullanacakları konusu kritik sorulardan birisi olmuştur.

Bu konuda özellikle İnsan Bilgisayar Etkileşimi alanında ciddi çalışmalar uzun yıllardan beri sürdürülmektedir. Örneğin NATO altında Human Factors başlığı altında çeşitli araştırma ve uygulamalar bulunmaktadır. Anthropometry, bilgi sistemleri arayüzlerinin ve kontrollerin tasarımı, bilişsel mühendislik gibi başlıklar bunlardan sadece bir kaçıdır.

Bu bildiride özellikle İnsan Bilgisayar Etkileşimi ve bunun Askeri sistemlerin tasarımı ve geliştirilmesindeki önemi sunulacak, bazı durum çalışmalarından örnekler verilecektir.

2. İNSAN BİLGİSAYAR ETKİLEŞİMİ NEDİR, NEYLE İLGİLENİR

İnsan Bilgisayar Etkileşimi, etkileşimli (interactive) teknolojilerin tasarımı, değerlendirmesi ve uygulaması ile ilgilenen disiplinler arası bir çalışma alanıdır. İnsan Bilgisayar Etkileşimi çalışma alanı, insan ve bilgisayar arasındaki etkileşimi konu edinmesinden dolayı insan davranışı, psikoloji, bilişsel bilimler, bilgisayar teknolojileri ve yazılım mühendisliği yanında ergonomi, grafik ve endüstriyel tasarım, sosyoloji, antropoloji ve eğitim bilimleri gibi alanlarla da ilişkili bir çalışma alanıdır [1, 2]

İnsan Bilgisayar Etkileşimi (İBE) sisteminin dört ana bileşeni vardır: kullanıcı, görev (task), araç/arayüz (tool), bağlam (context). İBE çalışmalarında kullanıcıların kullandıkları araçlarla belirli işleri yaparken elde edilen gözlem sonuçları, alışkanlık vb. etkilerle birlikte değerlendirilir ve bu veriler etkileşimli sistemlerin geliştirilmesi sürecinde kullanılır.

1945'te Vannevar Bush'un teorik analog bilgisayar fikrini (MEMEX: memory extender) ortaya atmasından sonra Shackel İBE alanında görülen ilk çalışmaları gerçekleştirmiştir [3]. "International Journal of Man-Machine Studies" adlı süreli yayın 1969 yılında kurulmuş, 1970'li yıllarda "kullanıcı dostu" (user-friendliness) kavramı ortaya atılmış, 1976 yılında NATO 'man-computer interaction' konulu bir çalıştaya sponsor olmuştur. Weinberg ve Shneiderman bu yıllarda konu ile ilgili ilk kitapları yayınlamışlardır [4, 5]. 1982 yılında ACM (Association for Computing Machinery) SIGCHI (Special Interest Group on Computer-Human Interaction) ve CHI (International Conference for Human-Computer Interaction) konferanslarını başlatmıştır. 1980'lerin sonlarında İnsan Bilgisayar Etkileşimi alanında dört büyük yayın hayatını sürdürmektedir. Bu dönemde

ilk endüstriyel İnsan Bilgisayar Etkileşimi Laboratuvarları kurulmaya başlanmış, 1990'lı yıllarda web teknolojilerinin gelişmesiyle İBE çalışmaları önemini artırmış ve teknik/tasarımcı merkezli tasarımlar yerine kullanıcı merkezli tasarımlar öne çıkmaya başlamıştır. 90'ların sonlarına doğru üniversitelerde ilk İBE akademik programları başlatılmıştır (örnek: Carnegie Mellon University, Indiana University).

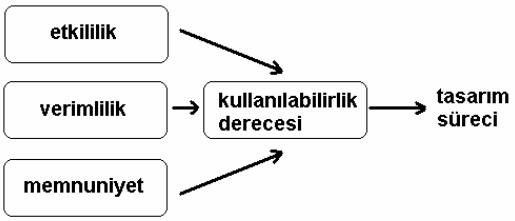
Booth, İnsan Bilgisayar Etkileşimi çalışmalarının çerçevesini aşağıdaki başlıklarla çizer [6].

- İnsanların teknoloji kullanımını etkileyen özellikleri nelerdir?
- Teknolojinin insanların teknolojiyi kullanımına etki eden yönleri nelerdir?
- İnsanlar etkileşimli yeteneklerini nasıl edinir ve kavramlaştırır?
- İnsanların ihtiyaçlarını teknik olanaklarla nasıl eşleştiririz?
- Kullanılabilir (usable) teknolojiler nasıl tasarlanabilir?
- Teknoloji organizasyonları nasıl etkiler?

Günümüzde İnsan Bilgisayar Etkileşimi çalışmaları bilgisayar kullanan nüfusun artması, servislerini etkileşimli teknolojileri kullanarak sağlayan kurumların güvenilirliğinin korunması, kritik teknolojik uygulamaların yaygınlaşmaya başlaması ve akademik anlamda insan davranışı ve zihinsel süreçlerin anlaşılması açısından önem kazanmakta; kabul edilmiş varsayımların yanlışlığını göstererek tasarımcıları ve IT uzmanlarını değerlendirmelerin bilimsel yöntemlerle yapıldığı platformlara çekmeye çalışmaktadır.

3. KULLANILABİLİRLİK NEDİR?

Kullanılabilirlik, bir uygulamada belirlenen işlerin hedef kitle olarak belirlenen kullanıcılar tarafından, gerekli eğitimin ve teknik desteğin verilmesinin ardından, uygun çevre koşullarında kolaylıkla ve etkili biçimde kullanılabilmesi olarak tanımlanabilmektedir. Uygulamada kullanılabilirlik, hedef kitledeki kullanıcıların, verilen işleri yaparken gösterdikleri verimlilik (efficiency), etkililik (effectivity) ve memnuniyetlerinin (satisfaction) ölçüsü cinsinden ifade edilebilir. Etkililik, kullanıcıların uygulamayı kullanarak yapması beklenen işleri ne kadar başarabildiğini ifade eder, bu anlamda etkililik işi yapabilme yüzdesi cinsinden ölçülebilir. Örneğin kullanıcıdan beklenen iş bir web sayfasındaki bilginin yerini bulmaksa etkililik, kullanıcının doğru bilgiyi bulmada gösterdiği başarı ile ifade edilebilir. Kullanılabilirliğin değerlendirilmesinde etkililiğin ölçümü tek başına yeterli olmaz. Belirlenen işi yapmak için kullanılan diğer kaynaklar (zaman, maliyet vb.) verimlilik ölçümü ile değerlendirilir. Web sayfası örneğinde verimlilik, kullanıcının belirlenen işi ne kadar sürede yaptığı ya da hangi yolları izlediği,işi kaç adımda tamamladığı vb. tespit edilerek belirlenebilir. Memnuniyet, kullanıcının uygulamayı kullanırken oluşan fikirlerinin (beğenilenler, beğenilmeyenler vb.) ölçüsünü ifade eder. Memnuniyet, kullanılabilirlik üzerinde etkililik ve verimlilik ile aynı derecede kritik önceliğe sahip değildir, etkililik ve verimlilik tarafından doğrudan etkilenir. Memnuniyet, bu amaçla düzenlenen memnuniyet anketleri ile tespit edilebilir. (Şekil-1)



Şekil 1. Kullanılabilirliği Etkileyen Temel Faktörler

Özetle, kullanılabilirlik derecesi etkililik, verimlilik ve memnuniyetin bir arada değerlendirilmesi ile oluşturulur ve bu değerlendirme tasarım süreci üzerinde belirleyici rol oynar.

Kullanılabilirlik araştırmaları kullanılabilirlik mühendisliği (usability engineering) kapsamında değerlendirilebilir. Kullanılabilirlik mühendisliğinin güçlü yanları doğrudan doğruya uygulamada belirlenen işlere bağlı olması, tek bir arayüz biçimine bağlı olmaması, ve arayüzlerin yinelemeli (iterative) biçimde geliştirilmesine olanak vermesidir. Bununla birlikte kullanılabilirlilik için belirlenen kriterlerin dinamik olduğu ve kullanılabilirliğin çevre faktörlerine bağımlı olduğu unutulmamalıdır.

Kullanılabilirlik, kullanıcı türlerine, kullanıcıların yaptığı işlere ve çevre koşullarına bağlıdır. Bu kapsamda etkileşim, etkileşimli arayüzler, kullanılabilirlik testleri, kullanıcı psikolojisi, sosyo-teknolojik sistem tasarımı, yeni teknolojilere karşı gösterilen direnç gibi konular incelenir. Etkileşimli arayüzler fare, klavye, monitor gibi fiziksel olabileceği gibi masaüstü, dosya/dizinler gibi yazılım tabanlı da olabilir.

Kullanılabilirlik kriterleri uygulamayı satın alacak müşteriler tarafından sözleşmeler ile, uygulamayı geliştiren tasarımcılar tarafından uygulama için belirlenecek hedefler ile, kullanılabilirlik değerlendirmesi yapan kişiler tarafından ya da uygulamayı kullanacak kullanıcılar tarafından belirlenebilir. Kullanılabilirlik kriterlerinin hedef kullanıcı kitlesine, uygulamada belirlenen işlere ve çevre faktörlerine bağlı olarak belirlenmesi gerekir. Örneğin eskisine alternatif olarak geliştirilen bir ürünün kullanılabilirliği şu örnek kriter ile belirlenebilir: "Kullanıcıların %70'i belirlenen işi yeni bir eğitime tabi tutulmadan %95 başarı ile (etkililik) ve eski uygulamaya kıyasla %25 daha hızlı (verimlilik) ve eşit düzeyde memnuniyet değerleri ile gerçekleştirmelidir" ya da "Kullanıcıların %80'i, iki günlük bir eğitimin ardından rutin işlerin %90'ını %90'ın üzerinde bir başarı oranı ile ve eski uygulamayla aynı hızda fakat daha yüksek memnuniyet değerleri ile gerçekleştirmelidir".

4. ARAYÜZ TASARIMINDA YÖNTEMLER

İnsan Bilgisayar Etkileşimi uzmanları iyi bir arayüz tasarımı için üç yol önermektedir.

Birincisi, stil kılavuzlarını (style guidelines) kullanmaktır. Bu kılavuzlar genellikle arayüz geliştiren büyük firmalar tarafından firma içinde geliştirilen tasarımlarda

uygulanmak üzere hazırlanır (Apple Macintosh Guidelines Java Programming Style Guidelines gibi). Bu kılavuzların temel amacı hazırlanan tasarımlarda tutarlılığı sağlamaktır, ancak uyumluluğun kullanılabilirlik ile aynı anlama gelmediğini ihmal etmemek gerekmektedir. Stil kılavuzlarını tasarlanması planlanan arayüzlerde kullanmak her zaman mümkün olmayabilir. Bunun yanında tasarımcıların kendi tasarımlarının kullanılabilirliği hakkında sağlıklı görüşler geliştiremediği gözönüne alınırsa [7] stil kılavuzlarının yorum ve uygulamayı tasarımcıya bırakması kullanılabilirlik açısından sakıncalar yaratabilmektedir.

İkinci yol, iyi tasarımın genel prensiplerini veren tasarım kılavuzlarının (design guidelines) kullanılmasıdır. Tasarım kılavuzları platform bağımsızdır ve kullanılabilirliği artırmaya yöneliktir. 1970'li yıllardan günümüze çok sayıda tasarım kılavuzu önerilmiştir. Günümüzde kullanıcı arayüz tasarımı için kullanılan en popüler tasarım kılavuzlarından birisi Nielsen tarafından verilmektedir (Nielsen's Ten Usability Heuristics). Nielsen'a göre kullanıcı arayüzlerinde aşağıdaki hususlara dikkat edilmelidir [8]:

- Basit ve doğal diyalog kullanımı
- Kullanıcının diliyle konuşma
- Hafıza yükünün en aza indirilmesi
- Tutarlılık
- Geribildirim sağlanması
- Açıkça gösterilmiş çıkış ve işlem sonlandırma
- Kısayolların önerilmesi
- Uygun hata mesajlarının tasarlanması
- Hataların engellenmesi
- Yardım ve belgeleme sağlanması

Tasarım kılavuzları, arayüzlerin geliştirilmesinde yararlı rol oynayabilmekle birlikte çok genel olmaları yoruma açık hale gelmelerine neden olmaktadır. Örneğin tutarlılığın nasıl sağlanacağı, geribildirimin nasıl verileceği, iyi bir hata mesajinin ne gibi özellikleri olması gerektiği konusunda konusunda yeterli bilgi içermez.

Tullis ve Kodimer'e göre stil ve tasarım kılavuzları, sezgi ve alışkanlık doğrultusunda yapılan arayüz tasarımları her zaman en etkili tasarımların oluşturulması için yeterli olmayabilmektedir [9].

Üçüncü yol kullanılabilirlik testlerinden elde edilen deneysel verileri kullanarak tasarımı geliştirmektir. Kullanılabilirlik testleri, kullanılabilirlik kriterlerinin belirlenmesinin ardından hedef kullanıcı kitlesine uygulanır ve sonuçlar, tasarımın yeniden değerlendirilmesi için girdi olarak kullanılır [10].

Kullanılabilirlik testleri için farklı yöntemler kullanılabilir. Kullanıcılar uygulamayı kullanırken sergilediği davranışlar -esneme, gerinme vb.- değerlendirilebileceği gibi kullanıcıdan, uygulama sırasında nasıl bir yol izlediğini sesli olarak aktarması istenerek

(verbal protokol) alınan görüntü ve ses kayıtları test sonrasında davranışlarla birlikte değerlendirilebilir. Göz-izleme (eye-tracking) cihazları kullanılarak elde edilen sonuçlar kullanılabilirlik açısından yorumlanabilir.

5. SAVUNMA SİSTEMLERİ VE İNSAN BİLGİSAYAR ETKİLEŞİMİ

Daha önce de belirtildiği gibi savunma sistemlerinin tasarımında İnsan-Bilgisayar-Etkileşimi konusunun büyük önemi bulunmaktadır. Bu konuda literatüre girmiş en önemli örnek ABD'nin USS Vincennes gemisinin yarattığı trajik problemdir. 3 Temmuz 1988 tarihinde körfezde normal uçuş koridorunu takip eden bir İran yolcu uçağı USS Vincennes gemisinden atılan bir füze nedeni ile düşmüştü. Uçaktaki 290 kişinin ölümü ile sonuçlanan bu trajik olayın nedenleri araştırıldığında büyük oranda İnsan Bilgisayar Etkileşimi kaynaklı olduğu anlaşılmıştı. İran uçağı düşürülmeden az önce bölgede ufak çaplı bir çatışmaya giren gemi personeli stres altındaydılar. Radarda beliren uçağı mürettebat bir F-14 olarak algılamıştı. Daha da vahimi uçak aslında 12500 feet de bulunmasına rağmen bilgiler 9000 feet seviyesine indiği şeklinde yorumlanmıştı. Bunu bir saldırı olarak algılayan kaptan uçağın vurulma emrini verdi. Aynı esnada USS Vincennes'in yakınında bulunan diğer ABD gemileri ise bunun bir saldırı olmadığını biliyorlardı ancak bu kısa sürede müdahale etme şansları olmadı. Peki neden bu trajik olay meydana geldi? İlgili gemiyi inceleyen uzmanlar USS Vincennes'e diğer gemilerden farklı olarak kontrol odasına son teknoloji olan sistemlerin kurulmuş olduğunu raporladılar. Ancak bu son teknoloji sistemde yaklaşan uçak ile ilgili kritik bilgiler farklı ekranlarda gösterilmekteydi. Farklı ekranlardan alınan bilgiler kısa süre içinde ve yüksek stres altında değerlendirildiğinde yanlış yorumlanmıştı. Ekranın birisi yükseklik bilgisini verirken diğer ekran uçağın pozisyonunu ve ne yöne gittiği bilgisini vermekteydi. Bu iki bilgi, bu kadar kısa sürede ve yüksek stres altında hatalı yorumlanmış ve sonuçta yanlış bir karar verilmişti. Diğer gemilerin sistemleri daha eski olmakla beraber USS Vincennes'deki gibi bir hataya yol açacak şekilde tasarlanmamıştı. Daha sonra NATO tarafından oluşturulan NATO STANAG 4586 "Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability" standardının ortaya çıkışında da bu kazadan alınan dersler rol oynamıştır [11].

NATO tarafından çeşitli standartlar ilgili araştırma ve çalışma gruplarınca oluşturulmaktadır. Örneğin "NATO Guidelines on Human Engineering Testing and Evaluation" raporunda sistemlerin daha kullanılabilir olması için nasıl test yapılması gerektiği açıklanmaktadır [12]. Yine NATO tarafından düzenlenen çalıştaylarda daha iyi sistem tasarımlarında insan faktörü konusunun nasıl göz önüne alınacağı konusunda araştırma çalışmaları yapılmaktadır. [13] İnsan merkezli tasarımın özellikle savunma ürünleri tasarım ve geliştirilmesinde nasıl hayata geçirileceği yine askeri araştırma laboratuvarlarının kritik konularından birisidir [14]

Savunma sistemlerinde İnsan Bilgisayar Etkileşimi çalışmaları kapsamında kullanılabilirlik çalışmaları büyük yer tutmaktadır. Ancak uygun test metodolojilerinin ne olacağı ve nasıl uygulanacağı da ayrı bir çalışma konusudur. Örneğin Ben Tristem

havacılık sistemleri testlerinde kullanılmak üzere sezgiseller (heuristics) geliştirmiştir. [15]. Yine bu sistemlerin testlerinde onaylama ve geçerleme konuları da üstünde durulan konular arasındadır [16].

Konu ile ilgili yapılan diğer bazı çalışmalardan bahsedecek olursak, Tristem çalışmasında Modern havacılıkta güvenlik kritik İnsan Bilgisayar Etkileşimi sistemlerinden bahsetmektedir [17.]. Yine buna paralel Johnson çalışmasında havacılık kazalarındaki komplex sistemleri İnsan Bilgisayar Etkileşimi açısından incelemektedir [18].

Sterling ve arkadaşları hedef belirleme sistemlerinin analizini yapmıştır [19]. Monika ve arkadaşları görsel ve işitsel ipuçlarının hedef bulmada askerleri nasıl etkilediğine bakmaktadır [20]. Jesse ve arkadaşları ise daha ileri teknolojiye yoğunlaşıp İnsan-Robot arayüzü konusunda çalışmaktadır [21]

Perrin, Ford ve Steinberg çalışmalarında savunma füze sistemlerindeki sistem durum göstergelerinin kullanılabilirliğini test etmektedirler. Füze sistemlerini kullanan personelin son derece kritik bir görevi gerçekleştirmeleri nedeni ile sıfır hata ile bu sistemleri kullanması gerekmektedir. Bu çalışmalarında "Theater High Altitude Area Defense (THAAD)" çalışanı uzmanlar en uygun gösterge tasarımının ne olduğu konusunda deneysel çalışmalar yapıp en uygun tasarımı bulmaya çalışmışlardır. [22]

Sterling ve Jacobson tarafından ABD Ordu Araştırma Laboratuvarı için hazırlanan raporda hedef tanımlama sistemini (aided target recognition (AiTR) system) etkileyen insan faktörleri ortaya çıkartılmaya çalışılmaktadır. Askeri bu sistemi kullanırken etkileyen bilişsel yük, stres ve performans gibi faktörler bulunmaya çalışılmaktadır. Bu faktörler ile arayüz tasarımı arasındaki ilişki araştırılmakta ve en uygun taktik, teknik ve prosedürler oluşturulmaya çalışılmaktadır. [23]

ABD ordusunun Director of Information Systems for Command, Control, Communications, and Computers (DISC4) biriminin özellikle savunma sistemlerine has kullanılabilecek İnsan Bilgisayar Etkileşimi tasarım kılavuzu eksikliğini gözlemlemesi sonucu 196 yılında bu amaçla bir kılavuz geliştirme çalışması başlatılmıştır. Çeşitli düzeltmelerle bu kılavuzun 1999 yılında 3. sürümü ortaya çıkarılmıştır. [23]

5. SONUÇ

Yukarıda verilen bilgilerden de görüldüğü gibi İnsan Bilgisayar Etkileşimi konusu özellikle tüm savunma sistemleri içine bilişim sistemlerinin girmesi nedeni ile çok kritik bir önem kazanmıştır. Askeri sistemleri çok geniş bir yelpazeden, faklı eğitim seviyelerindeki kullanıcıların kullandığı düşünülürse mümkün olan en düşük efor ve en yüksek başarımla bu sistemlerin kullanılması hayati bir öenme sahiptir. Yaşamın diğer alanlarında kullanılan bilişim sistemlerindeki kullanılabilirlik sorunları kullanıcıların yapmak istedikleri işleri gereğinden fazla zaman ve efor sarfederek gerçekleştirmelerine neden olmaktadır. Ancak savunma sistemlerinde kullanılabilirlik nedeni ile bir askerin karar verme sürecinin bir saniyelik gecikmesi bile sonuçta katastrofik olaylara, ciddi can ve mal kayıplarına yol açabilir. Bu tür sorunların önüne geçilmesi için savunma sanayi için üretim yapan firmalarımızın ilgili teorik ve pratik bilgi ile donatılmış yeterli pratik

deneyime de sahip kullanılabilirlik mühendislerini istihdam etmesi, bünyeleri içinde İnsan Bilgisayar Etkileşimi birimlerini kurmaları gerekmektedir.

Diğer ülkelerin savunma ile ilgili yetkili birimleri bu konuda ulusal seviyede de çalışmalar yürütmektedir. Ülkemizde de bu konuda üst düzey çalışmalar başlatılmalı ve örneğin öncelikle savunma sistemlerine fokuslanmış İnsan Bilgisayar Etkileşimi tasarım kılavuzu oluşturulmalı, ayrıca test standartları belirlenmelidir.

Ulusal Savunma Sistemlerimizin daha güçlü şekilde ortaya çıkmakta olduğu günümüzde mevcut çalışmalar paralel İnsan Bilgisayar Etkileşimi konusunda da ciddi çalışmaların yapılmaya başlaması konusunda nispeten geç kalınmış olmakla beraber bu fark kapatılmayacak kadar büyük değildir. Gerek üniversitelerde ve gerek bazı bu konuyu kritik çalışma alanı seçen Türk şirketlerinde oluşan birikim kullanılarak bu sorunun kolayca üstesinden gelinebilir.

6. KAYNAKÇA

[1] Çağıltay, K. (2006), E-dönüşümü Kullanabilmek? İnsan Bilgisayar Etkileşimi, Kullanılabilirlik ve e-Devlet Projeleri. Bilişim, 91, 2005, pp 16-17.

[2] Shneiderman, B. (1998), Designing the User Interface: Strategies for Effective Human-Computer Interactions, 3rd edition, Reading, 1998, MA: Addison-Wesley

[3] Shackel, B. Ergonomics for a computer. Design, 120, 1959, pp 36-39.

[4] Shneiderman, B. Software Psychology: Human Factors in Computer and Information Systems. 1980, Cambridge, MA: Winthrop.

[5] Weinberg, G. M. The psychology of computer programming, 1971: Van Nostrand Reinhold.

[6] Booth, P. An Introduction to Human-Computer Interaction. 1989, Hove, UK: Lawrence Erlbaum Associates.

[7] Bailey, S. Iterative methodology and Designer Training in Human Computer Interface design. Proceedings of ACM INTERCHI'93 Conference on Human Factors in Computing Systems, 1993, pp 198-205.

[8] Nielsen, J. Ten Usability Heuristics, http://www.useit.com/papers/heuristic/heuristic_list.html. Erişim tarihi 05.01.2006.

[9] Tullis, T. S., and Kodimer, M. L. A Comparison of Direct-Manipulation, Selection, and Data-Entry Techniques for Reordering Fields in a Table. Proceedings of the Human Factors Society Annual Meeting, Atlanta, GA, Oct. 1992.

[10] Nielsen, J. Usability Engineering. 1993, Boston, MA: AP Professional.

[11] Chairman, "STANDARDISATION AGREEMENT (STANAG) 4586: Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability," vol. March, N. S. A. (NSA), Ed., 2 ed: NATO, 2005

[12] Geddie, J. C., Boer, L. C.,Edwards, R. J.,Enderwick, T. P. Ve Graff, N. (2001). NATO Guidelines on Human Engineering Testing and Evaluation" http://stinet.dtic.mil/oai/oai?&verb=getRecord&metadataPrefix=html&identifier=ADA392142

[13] NATO, RTO Meeting Proceedings. (2001). The Human Factor in System Reliability –Is Human Performance Predictable? (RTO MP-032),

http://ftp.rta.nato.int/public//PubFulltext/RTO/MP/RTO-MP-032/MP-032-$$TOC.pdf

[14] Corona, B.M. (1997). Army Research Efforts in Human-Centered Design. http://www.ifp.uiuc.edu/nsfhcs/talks/corona.html

[15] Tristem, B. (1997). Usability Heuristics Applied to Modern Aviation, http://www.doc.ic.ac.uk/~nd/surprise_97/journal/vol2/bvct/

[16] Wise, J.A., Hopkin, V.D. ve Stager, P. (1993). Verification and Validation of Complex Systems: Human Factors Issues (NATO a S I Series Series III, Computer and Systems Sciences). Springer.

[17] Tristem, B. (1997). Safety Critical HCI Systems in Modern Aviation. http://www.doc.ic.ac.uk/~nd/surprise_97/journal/vol1/bvct/

[18] Johnson C.W., (2004). Looking Beyond the Cockpit: Human Computer Interaction in the Causal Complexes of Aviation Accidents

[19] Bruce S. Sterling; Catherine N. Jacobson (2006). A Human Factors Analysis of Aided Target Recognition Technology, October 2006

[20] Monica M. Glumm, Kathy L. Kehring, ve Timothy L. White (2006). Effects of Tactile, Visual, and Auditory Cues About Threat Location on Target Acquisition and Attention to Visual and Auditory Communications August 2006 http://www.arl.army.mil/arlreports/2006/ARL-TR-3863.pdf

[21] Jessie Y.C. Chen, Ellen C. Haas, Krishna Pillalamarri, ve Catherine N. Jacobson (2006). Human-Robot Interface: Issues in Operator Performance, Interface Design, and Technologies July 2006 http://www.arl.army.mil/arlreports/2006/ARL-TR-3834.pdf

[22] Perrin, L., Ford, B. ve Steinberg, D. (1997). Usability Testing of System Status Displays for Army Missile Defense. ACM CHI 97.

[23] Sterling, B.C. ve Jacobson, C.N. (2006). A Human Factors Analysis of Aided Target Recognition Technology, ARL-TR-3959, http://www.arl.army.mil/arlreports/2006/ARL-TR-3959.pdf

[24] Department of the Army Weapon Systems (2007). HCI Style Guide WSHCI, http://www.pnl.gov/wshciweb/wsdownload.htm

B3 OTURUMU:

EĞİTİM SİMÜLATÖRLERİ ve MODSİM SİSTEMLERİ TEDARİĞİ

MODELLEME, SİMÜLASYON (MODSIM), VE ARGE TABANLI SİSTEM TASARIMI, TEDARİĞİ VE KURULUMU
Levent SEVGİ, Gonca ÇAKIR

EĞİTİM AMAÇLI SİMÜLATÖRLERDE MALİYET-ETKİNLİK ANALİZİ: HAVAN EĞİTİM SİMÜLATÖRÜ ÖRNEĞİ VE BİR MODEL ÖNERİSİ
Taner ALTINOK, Bekir KAZANDIR, Bülent SOYKAN, Filiz ERSÖZ

SANAL BAKIM EĞİTİM SİSTEMLERİ
İsmail Hakkı ATEŞ

MODEL TABANLI TASARIM SÜREÇLERİNDE SİMÜLASYON VE ÜRÜN GELİŞTİRMEDE ALT YÜKLENİCİLİK SİSTEMİNİN UYGULAMASI
Özgür EMEKLI, Cağlar UCLER, Özalp VAYVAY

MODELLEME, SİMÜLASYON (MODSIM), VE ARGE TABANLI SİSTEM TASARIMI, TEDARİĞİ VE KURULUMU

Levent SEVGİ [(a)], Gonca ÇAKIR [(b)]

[(a)] Doğuş Üniversitesi, Mühendislik Fakültesi, Kadıköy-İstanbul, lsevgi@dogus.edu.tr

[(b)] Kocaeli Üniversitesi, Mühendislik Fakültesi, gonca@kocaeli.edu.tr

ÖZ

Tümleşik gözetleme, kontrol ve kumanda sistemleri modelleme ve simülasyon (MODSİM) tabanlı ARGE çalışmaları gerektiren sistemlere tipik örneklerdir. Üç tarafı denizlerle çevrili Türkiye'nin bu tip sistemlere olan ilgili on yıllar öncesine dayanmaktadır. Ege denizinin elektronik algılayıcılarla 24-saat kesintisiz gözetlenmesi Türk- Yunan barışının da güvencesi olacaktır. Benzer şekilde Kıbrıs civarında karşılıklı petrol arama çalışmalarının hareketlendiği günümüzde de Akdeniz'in kesintisiz gözetlenmesi zorunludur. Bu sunumda Akdeniz için bir gözetleme sistemi önerilecek ve bu sistem için gerekli MODSİM çalışmalarından örnekler sunulacaktır.

Anahtar Kelimeler: MODSİM, Tümleşik gözetleme, Radar, Dalga yayılımı, RKA.

SISTEM DESIGN, PRECUREMENT AND INSTALLATION BASED ON MODELING AND SIMULATION RESEARCH

ABSTRACT

Integrated surveillance, control and command systems are typical which necessitate research and development (R&D) studies based on modeling and simulation (MODSIM). Turkey, surrounded by Mediterranean, Agean and Black seas from three sides, has long interested in such systems. Continuous surveillance of Agean waters with all possible electronic sensors is also a key for the peace between Turkey and Greece. Similarly, continuous monitoring of activities on the Mediterranean waters is important, nowadays, parallel to the oil drilling conflicts around Cyprus. Novel integrated surveillance system alternatives for the surveillance of East Mediterranean will be introduced and necessary MODSIM based R&D studies will be discussed in this presentation.

Keywords: MODSIM, Integrated surveillance, Radar, Radiowave propagation, RCS

1. GİRİŞ

Bugünlerde sıcak haberlerin başında Güney Kıbrıs Rum yönetiminin Kıbrıs adası etrafında petrol arama yönünde attığı adımların ardından, Türkiye Petrolleri Anonim Ortaklığının (TPAO), Akdeniz'de petrol ve doğalgaz arama çalışmalarına başlamak için harekete geçmesi gelmekte. 2007 yılında petrol ve doğalgaz aramasını Akdeniz'e kaydıracak olan TPAO, bu çerçevede Akdeniz'in sığ sularında 12 millik bir alanı kapsayacak şekilde Antalya-İskenderun ve Mersin Körfezi'nde arama yapacak. İkinci aşamada aramalar derin suları, yani Rum kesiminin de arama yapacağı bölge olan Kıbrıs'ın batısıyla Mısır arasında kalan bölgeyi de kapsayacak şekilde genişletilecek.

Bu haberler, kıyılarımızda her türlü su-üstü ve hava hareketlerinin bir merkezden 24-saat kesintisiz gözetlenmesinin önemini tekrar gündeme getirmiştir. Bu aynı zamanda, ulusal savunmanın da olmazsa olmaz koşullarındandır. Kıyılarımızın denizden, karadan, havadan, ve denizaltından 24-saat kesintisiz gözetlenmesi ve kontrol edilmesi çok algılayıcılı tümleşik gözetleme, komuta/kontrol ve atış destek sistemleriyle olasıdır. Söz konusu sistemlerin tasarımı, satın alınması, kurulması, hatta işletilmesi dahi modelleme ve simülasyon (MODSİM) tabanlı araştırma ve geliştirme (ARGE) çalışmalarını zorunlu kılar.

Yüzey dalgalı yüksek frekans (YDYF) radarlarına Türkiye'de duyulan ilgi Deniz Kuvvetleri için tümleşik deniz gözetleme sistemleri çalışmalarıyla başlamıştır [1]. Profil bilgisi ve ters yapay açıklık prensibinden yararlanarak hedef sınıflama yetenekleriyle donatılmış kara ve hava platformlarında konuşlanmış mikrodalga radarlarının temel algılayıcılar olarak kullanıldığı böyle bir sistem Deniz Kuvvetleri için İTÜV-SAM [2] tarafından 1992-1997 yılları arasında tasarlanmıştır. Bu tarihten sonra sistem kurulum çalışmaları Deniz Kuvvetleri ARMERKOM (Araştırma Merkez Komutanlığı) tarafından devralınarak sürdürülmektedir.

Yüzey dalgalı YF radarları Başbakanlık Denizcilik Müsteşarlığı'nca Türk Boğazlarında kurulan ve 2004'te işletime alınan Gemi Trafik Yönetim Bilgi Sistemi (GTYBS) kapsamında da ele alınmıştır [2]. Her iki sistem kapsamında da YDYF radarlarına ait çalışmalar uluslararası literatürde geniş biçimde yer almıştır. Bunlar arasında sistem geliştirme çalışmaları [3-5], propagasyon etütleri ve modelleri [6-12], girişim ölçü ve modelleri [13,14], radar saçılma yüzeyi (RSY) analizleri [15-17], anten sistemlerinin tasarımları [18,19], radarlar ve işaret işleme tekniklerine ait araştırma ve uygulamalar, ve çevre gürültüsü ve spektrum doluluk ölçmeleri [20-23] sayılabilir.

Bu sunumda, Türkiye için stratejik önemi olan Doğu Akdeniz'de kesintisiz gözetleme sağlayacak, ekonomik ve güvenilir bir sistem ele alınacaktır. Uygun konuşlandırılmış iki adet YDYF radarı ile Kıbrıs adasının arkası dahil tüm Doğu Akdeniz'in stratejik gözetlenmesi söz konusu olabilecektir. İkinci bölümde kısaca YDYF radarları anlatılmıştır. Önerilen sisteme ait ayrıntılar Üçüncü Bölüm'de sunulmuştur. Dördüncü bölümde, böyle bir sistem için özellikle fizik tabanlı MODSİM zorlukları tartışılmıştır. Günümüzde MODSİM eğitimden, sistem entegrasyonuna, performans analizlerinden, ARGE'ye dayalı tedarik çalışmalarına, karar destek uygulamalarından harekat

tatbikatlarına kadar geniş bir alanda yaygın olarak kullanılmaktadır. Sonuçlar Beşinci Bölümde özetlenmiştir.

2. YÜZEY DALGALI YÜKSEK FREKANS RADARLARI

Günümüz gözetleme, kontrol, erken uyarı gibi elektronik sistemlerinin temel taşı amaca uygun seçilecek olan radarlardır. Kullandığı işaret (dalga) cinsi, frekansı yada diğer parametreleri ne olursa olsun her radar değişik fiziksel temellere sahiptir. Örneğin, toprak altında yüzeyden birkaç cm derinliğe gömülen nesneleri algılamak için kullanılan sistem ile, yerin onlarca metre altında gömülü taktik-balistik füzeleri, nükleer silahları yada binaları saptamak, tünelleri bulmak için kullanılacak sistem çok farklı olacaktır. Benzer şekilde, iki vadi arasındaki sınır boyunu gündüz yada gece sürekli gözetlemek için düşünülen sistem ile karasularımızı ve 200 deniz miline dek uzanan uluslararası sulardaki haklarımızı gözetmek için kullanılacak sistem birbirine hiç benzemeyebilecektir.

1970'li yıllara gelindiğinde YF (3-30 MHz) bandını kullanan elektronik sistemlere "eski teknolojiler" ve mikrodalga frekanslarını (yaklaşık 1 GHz ve üstü) kullanan sistemlere "yeni teknolojiler" denmekteydi. Mikrodalga haberleşme sistemleri, mikrodalga radarları, uydu verici ve terminallerinin yaygınlaşmaya başlaması teknolojinin buna ayak uydurmasını zorlamaktaydı. Aradan geçen otuz yıla yakın sürede mikrodalga teknolojisi gelişti, oturdu ve elektronik sistemlerin vazgeçilmezleri arasında yerini aldı. Son on yılda, özellikle radar sistemleri açısından "eski ve yeni teknolojiler" deyimleri yer değiştirir gibi oldu. Bunun en önemli nedeni yüksek kapasite ve hızlı bilgisayarların kullanımı ve radar işaret işleme tekniklerindeki gelişmeler sonucu YF radarlarının önem kazanmaya başlamaları oldu.

Mikrodalga radarlarının kullanımı ufuk hattıyla sınırlıdır. Bu nedenle, örneğin zeminden 40-50 m yüksek bir platformda bile görüş alanı 40 km'yi geçmez. Daha geniş bölgeleri kapsayabilmek için mikrodalga radarları ya hava platformlarına ya da çok yüksek tepelere konuşlandırılır. YF radarları uzak mesafeleri ve geniş bölgeleri kapsayabilen algılayıcılardır. Özellikle YF frekanslı dalgalar *ufuk hattı*nın ötesine ulaşabildiklerinden deniz gözetleme sistemlerinde yaygın olarak kullanılmaya başlanmışlardır. Yer dalgalarıyla (3-10 MHz arasında) 500 km'ye gök dalgalarıyla ise (15-30 MHz arasında) binlerce km'ye ulaşabilmeleri olasıdır.

Alt HF frekanslarını kullanan YDYF radarları değişik amaçlarla yaygın olarak kullanılmaya başlanmıştır. Gök dalgalı YF radarlarına göre hem kurulumu hem de işletimi son derece ucuz olan YDYF radarları ile hem gözetleme, hem kontrol hem de oşinografik bilgi elde etmek olasıdır. YDYF radarları ile tüm kıyılarımızın, üstelik Birleşmiş Milletler UNCLOS (United Nations Convention on the Law of the Sea) anlaşmasında belirtilen 200 deniz miline dek kapsanabilmesi olasıdır.

Örneğin, Kanada Donanması Kanada hükümeti ve Raytheon Kanada firmasının yaklaşık on yıldır ortaklaşa sürdürdükleri çalışmalar sonucu geliştirilen ve denenen iki radarlı tümleşik gözetleme sistemini yaklaşık 10 yıldır ulusal sisteminde kullanmaktadır [4,5]. Avrupa Birliği üye ülkelerinden Almanya, Norveç, İngiltere ve İspanya'nın

birlikte yürüttüğü önemli bir çalışma olan EuroROSE (European Radar Ocean Sensing) projesinde ise yine YDYF radarlarından küçük ve orta ölçekte kapsamalar için kullanılan Codar ve Wera sistemleri [24] başarıyla denenmektedir. Bu sistemlerle, örneğin Kanada hükümeti Atlas okyanusunda zengin petrol yataklarının bulunduğu Terra Nova bölgesindeki dev petrol platformları için tehdit oluşturan kuzey buzullarını gözetlemekte hem de savunmasını güçlendirmektedir. Norveç hükümeti benzer amaçlarla batı kıyılarında bir YDYF radarı kullanmaktadır. İspanya oşinografik amaçlı bir YDYF radarını yine EuroROSE projesi çerçevesinde denemektedir. Almanya, deniz ticaretinin önemli bir yükünü taşıyan Hamburg limanı çevresinde hem oşinografik hem de deniz trafiğinin kontrolü amaçlarıyla YDYF radarları kullanmaktadır. ABD'de Teksas kıyıları ve körfezi, fırtına, dalga yüksekliği, akıntı yönleri, vb. açılarından orta düzeyde mobil bir YDYF radarlarıyla sürekli denetlenmektedir. Avustralya, Yeni Zelanda, Hong Kong ve Singapur'da da benzer çalışmalar bulunmaktadır.

YDYF radar sistemlerinde iyi tasarlanmış verici ve alıcı anten dizileri gekerlidir. Ortalama verici güçleri 10-20 kW, kapsama alanı yaklaşık 100-120 derecelik bir alandan 500-600 km'ye çıkabilmektedir. Sahilde 400-1200 m arasında düzgünleştirilmiş bir alanda kurulan alıcı anten dizileriyle elektronik demet oluşturma ve tarama yapılabilmektedir. Yüksek atmosfer ve çevre gürültüsü ortamında çalışmak zorunda olan bu radarlar akıllı deteksiyon, izleme ve sınıflandırma yazılımlarına gerek duyar. Etkili yazılımların hazırlanması ise fizik temelli MODSİM çalışmalarına bağlıdır.

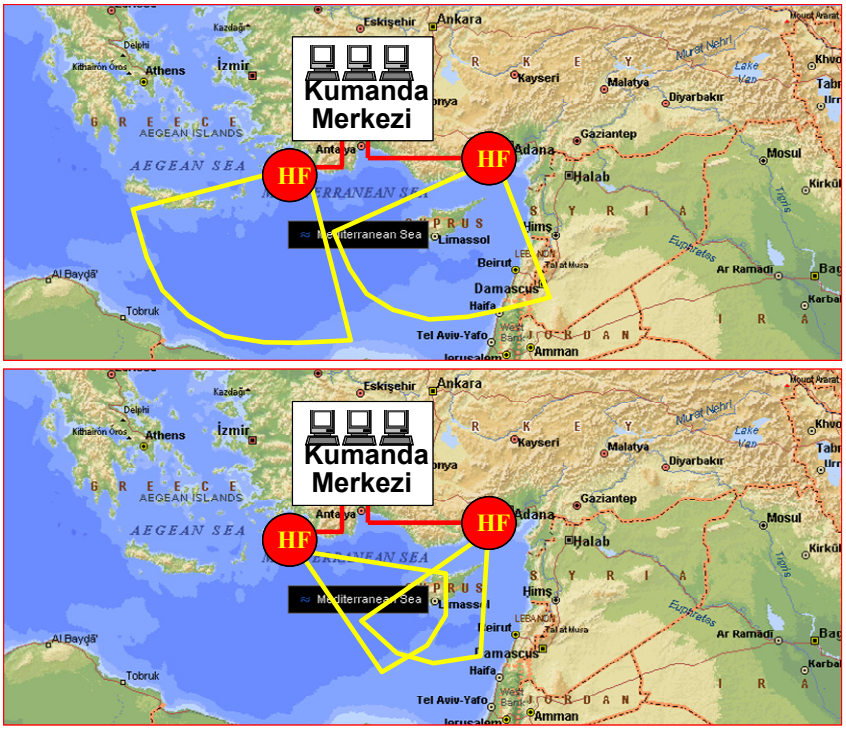
Aralık 2004'te yaşanan yıkıcı 9 büyüklüğündeki Endonezya depremi "*Elektronik sensörlerle tsunamiler önceden saptanabilir mi*?" sorusunu gündeme getirmiştir. YDYF radarlarının tsunami saptama potansiyeli vardır.

3. DOĞU AKDENİZ İÇİN BİR GÖZETLEME SİSTEMİ

Doğu Akdeniz için tipik bir tümleşik gözetleme sistemi Şekil 1'de gösterilmiştir. Şekilde iki farklı yerleşim ve kapsama için tipik olarak çizilmiş iki adet YDYF radarı kullanan bir sistem resmedilmiştir. Türkiye açısından stratejik önemi olan Doğu Akdeniz bölgesinin, Kıbrıs adasının güneyi dahil, sürekli gözetlenmesinin alternatif yolları bulunmaktadır. Örneğin, biri Muğla bölgesinde, diğeri Antalya – Hatay hattında kurulacak iki YDYF radarı ile bölgenin kapsanabilmesi söz konusudur. Elbette, radarların uygun yerleri, bölgede yapılacak gürültü ve iletkenlik ölçmelerine, propagasyon etüdlerine ve kapsama alanları analizlerine göre saptanmak zorundadır. Yine, radarların bakış açıları bu analizlerin sonunda optimize edilmek durumundadır. Ancak, kapsamanın güvenilir ve kesintisiz olabilmesi bölgeye Batı Akdeniz yönünden ve Girit adası civarından girecek hedeflerin daha girer girmez sistemde görülebilmelerine sıkı sıkıya bağlıdır. Giriş – çıkışlar ne denli yüksek performans ile izlenebilirse sistemin başarısı da o denli artacaktır. Her iki radar, karada ya da komuta gemisinde kurulacak bir operasyon kontrol merkezinden kontrol edilecektir.

YDYF radarları özellikleriyle üç tarafı denizlerle çevrili Türkiye'nin, özellikle su-üstü ve alçak irtifa hedefleri için hem ekonomik hem de kullanışlı sistemlerdir. Ancak, bu tip radarların kullanımı sadece bunlarla sınırlı değildir. YF bölgesinde değişik frekanslar kullanılarak daha küçük sistemler, örneğin Çanakkale ve İstanbul Boğazlarının

girişlerini kontrol etmede etkin biçimde kullanılabilir. Deniz taşımacılığında yükleme/boşaltma hareketlerinin yoğun olduğu İzmir, İskenderun gibi limanların kontrolünde görev verilebilir. Türk Boğazları için kurulan GTYBS ile bütünleştirilerek kullanılacak kısa menzilli ve dar kapsamalı YDYF radarları ikincil algılayıcılar olarak sisteme büyük fayda sağlayabilir.



Şekil 1. İki HF radarı ile Doğu Akdeniz'in kesintisiz gözetlenmesi

Benzer şekilde Ege denizinde irili ufaklı adalar nedeniyle performansı oldukça tartışılacak olan YDYF radarlarına dayalı sistem biri Girit bölgesinden girişi, diğeri Çanakkale'den çıkışı kontrol edecek iki radarla yer değiştirebilir. YDYF radarları Ege denizinde, hatta Akdeniz açıklarında su-üstü platformlarında kullanılabilir. Bu sayede operasyon bölgesi istendiğinde kolayca değiştirilebilir.

4. MODSİM TABANLI ARGE ÇALIŞMALARI

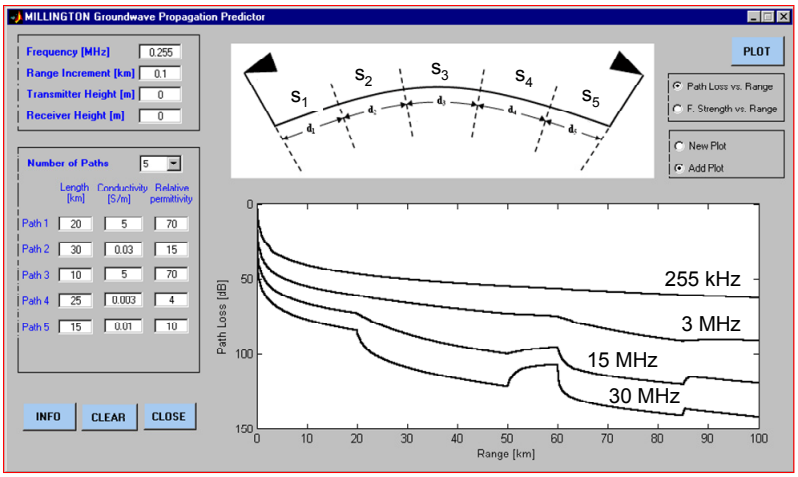
Yeni tip YDYF radarları mevcut ticari teknoloji ve düzenlerle kurulabilecek sistemlerdir. Buna karşın sistemler etkin ve akıllı yazılımlara gereksinir. Çevre gürültüsünün fazla olduğu, yoğun kullanımı olan YF bölgesinde radarlarda işaret işlemenin tüm incelikleri kullanılmak zorundadır. Bu sistemlerde halen çözüm bekleyen önemli sorunların başında iyonosferden gelen istenmeyen işaretler ve etkiler, işaretten genelde 20-50 dB daha yüksek seviyelerde olan deniz kargaşası, çevre gürültüsü,

propagasyon etüdleri, radar kesit alanı (RKA) hesapları, stokastik modelleme ve sistem entegrasyonu sayılabilir. Her biri, doğrusal ve homojen olmayan, zamanla değişen ve stokastik özellikler içeren bu problemlerin basit işaret işleme algoritmalarına dayalı yazılımlarla çözülmesini beklemek hayalden öte bir şey değildir. Henüz bu işaretlerin davranışları gerçeğe yakın biçimde modellenememektedir. Örneğin, deniz kargaşasının yüksek deniz durumlarında kullanılabilecek modeli henüz bulunamamıştır. Yine iyonosfer tabakasının yeryüzü konumuna göre davranışları, kısa, orta ve uzun dönemli hareketlilikleri kestirilememektedir. Bu nedenle, bu ortamlardan gelen işaretlerin etkilerinden kurtulabilmek ve bu işaretleri süzebilmek için sürdürülen çabalar henüz akademik düzeyde kalmakta ve uygulanamamaktadır. Yine de bu radarlar tümleşik gözetleme sistemlerinin önemli unsurları arasına girmiştir.

Bu bölümde iki tipik MODSİM çalışmasına ait örnek ele alınmıştır.

4.1. Propagasyon Etüdleri

YDYF radarlarının başarımı için bölgede propagasyon etüdlerinin yapılması zorunludur. Bu frekans bölgesinde yer dalga yayılımının modellenmesi güçlü propagasyon simülatörlerini gerektirir. Tipik bir problem, karma-yol (Millington) etkisi denen ve deniz-kara geçişlerinin modellenmesi problemidir. Bu konuda Uluslararası Telekomünikasyon Kurulu (ITU) belgelerinde [25] belli koşullarda uygulanabilecek yöntemler bulunmaktadır. Modelleme çalışmaları yaklaşık 20 yıldır grubumuzda da sürdürülmektedir [7-12]. Bugün bu simülatörler uluslararası değişik merkezlerce kullanılmaktadır. Simülatörlerden tipik bir örnek Şekil 2'de gösterilmiştir [8].



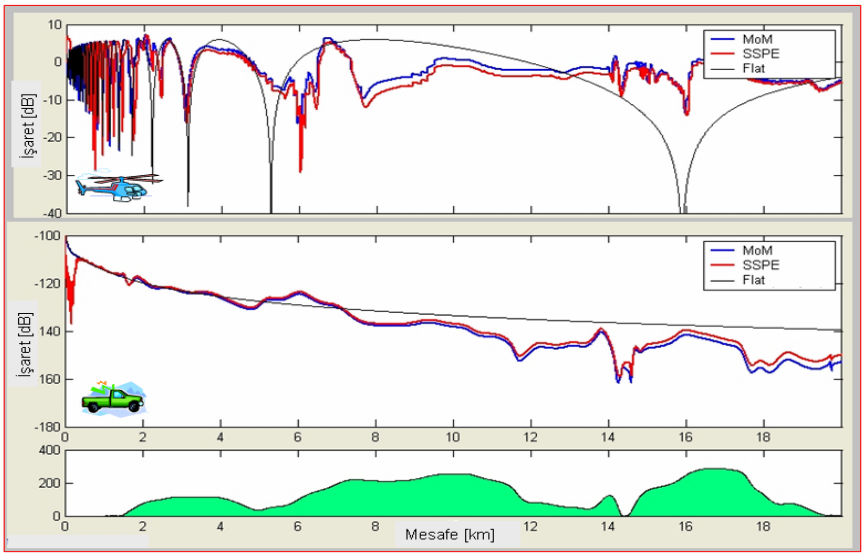
Şekil 2. Propagasyon etüdleri için bir örnek

MILLINGTON simülatörü ITU belgelerinde verilen yöntemlerin otomasyonunu da sağlamaktadır. Kullanıcı, işaret frekansını, kaç km'de bir hesap yapılacağını, kaç parçalı propagasyon yolu olduğunu, bu yolların (deniz ya da kara) uzunluklarını ve elektriksel parametrelerini girdiğinde simülatör otomatik olarak yol kaybını hesaplamaktadır.

Şekilde, 5 parçalı karma yol için 255 kHz, 3 Mhz, 15 MHz, ve 30 MHz frekansları için yol kaybı eğrileri gösterilmiştir. Millington etkisi denen ve ada üzerinde işaret kaybının artması, ancak ada ardında denize geçildiğinde işaretin bir miktar kendini toparlaması açıkça görülmektedir. Millington etkisinin bir kaç yüz kHz'e kadar etkisinin olmadığı ancak MHz'lerden itibaren frekans arttıkça etkinin arttığı gözlenmektedir.

MILLINGTON simülatörü yeryüzü arazi profillerini göz önüne almayan analitik bir yönteme dayanır [8]. Oysa, bir çok gerçekçi propagasyon etüdü arazi profilinin etkisinin de işin içine katılmasını zorunlu kılar. Grubumuzca geliştirilen değişik simülatörlerden ikinci bir örnek [7] Şekil 3'te gösterilmiştir.

Burada kullanıcı istediği arazi profilini verme olanağına sahiptir. Verilen arazi profili üzerinde simülatör hem kuş uçuşu hem de araziyi izleyerek seçilen patikalar üzerinde yol kaybını, üstelik Parabolik Denklem (PE) ve Moment (MoM) gibi iki güçlü yöntem sonuçlarını karşılaştırarak vermektedir.



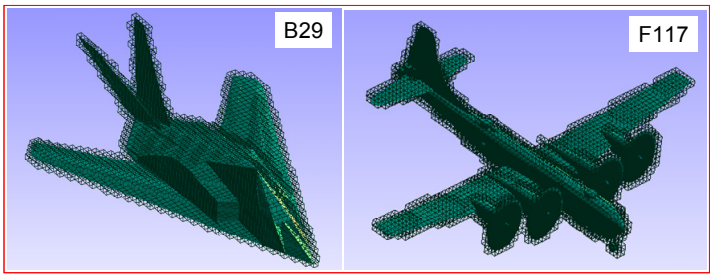
Şekil 3. SSPE ve MOM simülatörleri ile yol kaybı simülasyonlarına bir örnek

Değişik tipte radar kapsama alanları analizleri yapan araştırıcıların bu örneklerde verilen simülatörleri kullanmasının MODSIM ve ARGE'ye dayalı sistem tasarımı ve tedariği açılarından son derece yararlı olacağı açıktır.

4.2. Radar Kesit Alanı (RKA) Modelleme

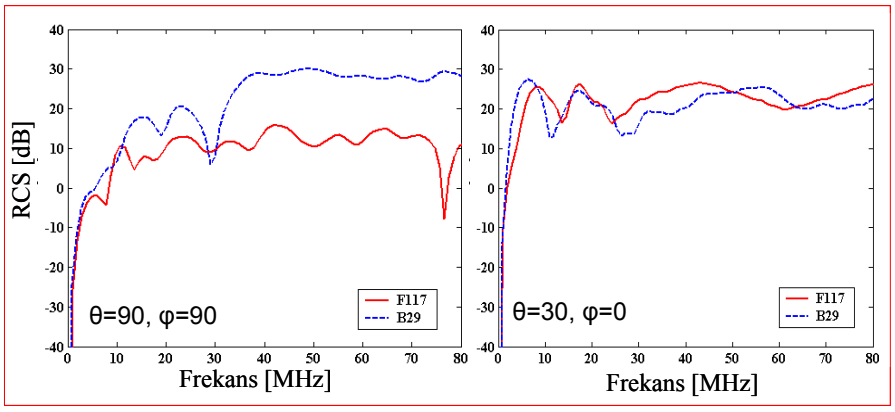
YDHF radarlarının başarımı hedeflerin elektromanyetik saçıcılık özelliklerinin çok iyi anlaşılabilmesine bağlıdır [15-17]. Bu ise diğer bir önemli MODSİM konusudur. Ayrıca, YDYF radarlarının bir önemli özelliği de hayalet uçak (stealth) denen ve RKA değerleri çok azaltılmış hedefleri yakalayabilmesidir. Diğer Mikrodalga radarlarının

yakalayamadığı hayalet uçakların YDYF radarlarınca yakalanabilmesinin bilgisayar simülasyonlarıyla gösterilmesine tipik bir örnek Şekil 4 ve 5'te gösterilmiştir. Şekil 4'te B29 hayalet ve F117 normal hava hedeflerinin ayrık modelleri gösterilmiştir.



Şekil 4. Hayalet uçak B29 ve F1117 için ayrık bilgisayar modelleri

Uzunlukları aynı (15 m) olan bu hava hedeflerinin iki farklı bakış (radar) açısından frekansla RKA değişimleri Şekil 5'te resmedilmiştir. Birinci resim hedeflerin radar yönünde yaklaşmalarını temsil etmektedir. B29'un RKA değerinin F117'ye göre yaklaşık 20dB (yüz kat) daha az olduğu görülmektedir.



Şekil 5. B29 ve F117 uçakları için frekansla RKA değişimi

Oysa, aşağıdan ya da yukarıdan bakan bir YDHF radarına ait senaryo ikinci resimde gösterilmiştir. Burada ise RKA değerleri açısından B29 ile F117 arasında pek fark olmadığı gözlenmektedir.

5. SONUÇLAR

Türkiye, bugün için yeni tipte YDYF radarlarını üretebilecek güçtedir. Bu amaçla gerekli birikim mevcuttur. Yeni tip radarların olumlu olumsuz tüm yönleriyle doğru değerlendirilmesi ve kullanım alanları oldukça yaygın olan bu radarların üretilmesi bu birikimlerini ne derece doğru kullandığına bağlıdır.

Günümüzün ulusal savunma sistemleri ileri teknoloji gerektirir; pahalı ve karmaşıktır. Örneğin sadece belli bir bölgeyi gözetlemek için kurulacak orta düzeyde bir sistemin maliyeti onlarca milyon ABD Dolarından başlayabilmekte ve milyar Dolarlara çıkabilmektedir. Böyle bir sistemin satın alınması bile başlı başına bir ARGE projesidir ve MODSİM ve ARGE tabanlı tasarım, tedarik, kurulum ve işletilmesi problemi olarak ele alınmak zorundadır.

KAYNAKÇA

[1] N. Ince, E. Topuz, E. Panayirci, C. Isik. *Principles of Integrated Maritime Surveillance Systems*. Kluwer Academic, Boston, 2000

[2] L. Sevgi, *Complex Electromagnetic Problems & Numerical Simulation Approaches*, IEEE Press, ISBN No: 0-471-43062-5, April 2003

[3] Uzun Ufuk ve TURBO Proje Ofisi, İTÜV-SAM, İstanbul Teknik Üniversitesi Vakfı, Savunma Araştırmaları Merkezi, Dz.H.H., Tuzla

[4] L. Sevgi, A. M. Ponsford, H.C. Chan, "An Integrated Maritime Surveillance System Based on Surface Wave HF Radars, Part I – Theoretical Background and Numerical Simulations", *IEEE Antennas and Propagation Magazine*, V.43, N.4, pp.28-43, Aug. 2001

[5] A. M. Ponsford, L. Sevgi, H.C. Chan, "An Integrated Maritime Surveillance System Based on Surface Wave HF Radars, Part II – Operational Status and System Performance", *IEEE Antennas and Propagation Magazine*, V.43., N.5, pp.52-63, Oct. 2001

[6] L. Sevgi, "Modeling and Simulation Strategies for Electromagnetic Wave Propagation in Complex Environments: Groundwave Path Loss Prediction Virtual Tools", (Special issue on Electromagnetic Wave Propagation in Complex Environments: A Tribute to Leopold Benno Felsen) *IEEE Transactions on Antennas and Propagation*, (scheduled for) April 2007

[7] F. Akleman, L. Sevgi, "A Novel MoM- and SSPE-based Groundwave Propagation Field Strength Prediction Simulator", *IEEE Antennas and Propagation Magazine*, (scheduled for Oct issue) 2007

[8] L. Sevgi, "A Mixed-Path Groundwave Field Strength Prediction Virtual Tool for Digital Radio Broadcast Systems in Medium and Short Wave Bands", *IEEE Antennas and Propagation Magazine*, Vol. 48, No 4, pp.19-27, Aug 2006

[9] L. Sevgi, Ç. Uluışık, F. Akleman, "A Matlab-based Two-dimensional Parabolic Equation Radiowave Propagation Package", *IEEE Antennas and Propagation Magazine*, Vol. 47, No. 4, pp.184-195, Aug. 2005

[10] M. O. Özyalçın, F. Akleman, L. Sevgi, "A Novel TLM Based Time Domain Wave Propagator", *IEEE Transactions on Antennas and Propagation*, Vol. 51, No. 7, pp. 1679-1683, July 2003

[11] L. Sevgi, F. Akleman, L. B. Felsen, "Ground Wave Propagation Modeling: Problem-matched Analytical Formulations and Direct Numerical Techniques", *IEEE Antennas and Propagation Magazine*, Vol. 44, No.1, pp.55-75, Feb. 2002

[12] F. Akleman, L. Sevgi, "A Novel Finite Difference Time Domain Wave Propagator", *IEEE Antennas and Propagat.*, Vol. 48, No 5, pp.839-841, May 2000

[13] F. Akleman, L. Sevgi, "Stochastic Modeling of Tropospheric Wave Propagation", *URSI, XXVIth General Assembly*, Aug.13-21, 1999, Toronto, Canada

[14] L. Sevgi, A. M. Ponsford, "Propagation and Interference Characteristics of Surface Wave HF Radars", *URSI, XXVIth General Assembly*, Aug.13-21, 1999, Toronto, Canada

[15] L. Sevgi, "Target Reflectivity and RCS Interaction in Integrated Maritime Surveillance Systems Based on Surface Wave HF Radar Radars", *IEEE Antennas and Propagation Magazine*, V.43, N.1, pp. 36-51, Feb. 2001

[16] F. Akleman, M. O. Özyalçın, L. Sevgi "Radar Cross Section Modeling and Stealth Target Design", *1999 Symposium on Aeronautics and Astronautics Defense Technologies*, Turkish Airforce Academy, 34807, April 1999, Yesilyurt / Istanbul Turkey

[17] L. Sevgi, S. Paker, "FDTD Based RCS Calculations and Antenna Simulations", *AEU, International J. of Electronics and Commun*., Vol.52, No.2, pp.65-75, March 1998

[18] L. Sevgi, "Numerical Simulation Approaches for Phased Array Design", *ACES Journal of Applied Computational Electromagnetic Society*, Special issue on Phased Array Design, (invited tutorial) Vol. 21, No.3, pp.206-217, Nov 2006

[19] L. Sevgi, "HF Wire Antenna Array Design via FDTD and MoM Techniques", *Proc. of EEE Canadian Conference on Electrical and Computer Engineering*, pp. 7-12, (ISBN: 0-7803-5579-2)i May 9-12, 1999, Alberta, Canada

[20] L. Sevgi, "Stochastic Modeling of Target Detection and Tracking in Surface Wave High Frequency Radars", *Int. J. of Numerical Modeling*, Vol. 11, No 3, pp.167-181, May 1998

[21] L. Sevgi, A. S. Sanal "Surface Wave HF Radar Simulator", *Proc. of IEE Radar 97* Conference, 14 - 16 October 1997, Edinburgh, UK

[22] F. Akleman, L. Sevgi, "Algılayıcılar ve işaret işleme: Propagasyon Modelleme ve işaret kestirimi", *SIU2000*, Sinyal işleme ve uygulamaları kurultayı, sf.209-214, 12-14 Haziran 2000, Antalya

[23] C. Işık, L. Sevgi, "Electromagetic Interference Measurements and Spectral Occupancy Evaluations in HF Frequency Band", *SIU97,* Sinyal işleme ve uygulamaları kurultayı, sf. 234-237, 1-3 Mayıs 1997, Kusadasi

[24] H. Gunther, et all, "The EuroROSE Project", Proceedings of the 16th International Conference of the American Meteorological Society on "Interactive Information and Processng Systems (IIPS) for Metereology, Oceanography and Hydrology", California, USA, pp 214...217, 9.-14. January 2000

[25] ITU-R, Recommendations, P-368-7, "Groundwave Propagation Curves for frequencies Between 10 kHz and 30 MHz", *International Telecommunications Union*, Mar 1992

EĞİTİM AMAÇLI SİMÜLATÖRLERDE MALİYET-ETKİNLİK ANALİZİ: HAVAN EĞİTİM SİMÜLATÖRÜ ÖRNEĞİ VE BİR MODEL ÖNERİSİ[6]

Taner ALTINOK [(a)], Bekir KAZANDIR[(a)], Bülent SOYKAN [(a)], Filiz ERSÖZ [(a)]

[(a)] Kara Harp Okulu, Savunma Bilimleri Enstitüsü, 06654, Bakanlıklar/ANKARA,
{taltinok, bkazandir, fersoz}@kho.edu.tr, soykanb@gmail.com

ÖZ

Günümüzde maliyet-etkin, en son teknoloji ürünü simülatörlerin eğitimde kullanımı hayati öneme haizdir. Eğitim alanlarının yerleşim bölgelerinin içerisinde kalması, çevreci yaklaşım ve tasarruf konuları bu görüşü desteklemektedir. Bu çalışmada, dünyada geliştirilen havan eğitim simülatörlerinin ortak özellikleri incelenmiş ve eğitim amaçlı simülatörlerin maliyet-etkinlik analizi kapsamlı olarak değerlendirilmiştir. Ayrıca kullanıcı ihtiyaçlarını karşılayabilecek bir havan eğitim simülatör modeli önerilmiş ve maliyet-etkinlik analizi yapılmıştır.

Anahtar Kelimeler: Havan Eğitimi, Maliyet-Etkinlik, Simülasyon, Simülatör, Teknoloji.

COST-EFFECTIVENESS IN TRAINING SIMULATORS: EXAMPLE OF MORTAR TRAINING SIMULATOR AND A MODEL PROPOSAL

ABSTRACT

Nowadays, using simulators, which is cost-effective, and product of latest technology in training has vital importance. This idea has being supported by the residential areas that covers the training zones, environmentalist approaches and savings branches. In this study, the commonalities of mortar training simulators that have been developed by worldwide have analyzed and cost-effectiveness analysis of the training simulators comprehensively evaluated. Besides, a mortar training simulator model, which is capable of meeting the user requirements, has proposed and cost-effectiveness of the model has analyzed.

Keywords: Cost-Effectiveness, Mortar Training, Simulation, Simulator, Technology.

[6] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

1. GİRİŞ

Simülasyon, var olan veya var olması muhtemel sistemlerin dinamikleri üzerinde çalışma yapmak amacıyla o sistemlere ait model geliştirilmesi ve bu modelin canlandırılması işlemidir [1]. Simülatörler ise bir sistemin temel özelliklerini, dinamik davranışlarını taklit eden donanım ve yazılımdan oluşan cihazlardır. Eğitim amacıyla kullanılan simülatörler, gerekli bilgi, beceri ve yetenek seviyesini kazandırmak ve pratik yapılmasını sağlamak için, eğitilen personele gerçek sistem ile etkileşiyormuş hissini verirler [2].

Bilgisayar ve bilgi teknolojilerindeki gelişmeler askeri sistemlere uygulandıkça eğitim amacıyla kullanılan simülatörler yaygınlaşmakta ve ülkelerin azalan savunma bütçeleri ve silah sistemlerinin karmaşıklaşması simülasyon uygulamalarını vazgeçilemez bir araç haline getirmektedir. Günümüzde artık pek çok muharebe ortamı bileşeni sentetik ortamda ayrıntılı olarak simüle edilebilmekte ve gerçekçi bir eğitim ortamı oluşturulabilmektedir.

Eğitim simülatörleri aşağıda sunulan avantajlardan dolayı daha fazla tercih edilir duruma gelmektedir:

- Maliyet-etkin ve zaman-etkin bir çözüm sunması, (Örneğin havan atış eğitimi amacıyla kullanılan gerçek mühimmatın maliyeti oldukça yüksektir. Ancak, bu eğitim simülatör ile verilirse maliyet çok daha düşük bir seviyede kalmaktadır.)
- Gereksiz risk ve tehlikeleri arındırması,
- Çok tekrar ve geri besleme ile eğitimin etkinliğini artırması,
- Çevreye zarar vermemesidir.

Havan eğitim simülatörleri de eğitim simülatörlerine bir örnek olarak verilebilir. Bu çalışmada, eğitim amaçlı simülatörlerin maliyet-etkinlik analizi kapsamlı olarak değerlendirilmiş ve dünyada geliştirilen havan eğitim simülatörlerinin ortak özellikleri incelenmiştir. Ayrıca kullanıcı ihtiyaçlarını karşılayabilecek maliyet-etkin bir havan eğitim simülatör modeli önerilmiştir.

2. HAVAN EĞİTİMİNDE SİMÜLASYON UYGULAMALARI

Askeri sistemlerde meydana gelen teknolojik gelişmeler, havanların klasik ve bilinen türde kullanımını ortadan kaldırmaktadır. Yeni nesil havanlar, beka ve hareket kabiliyetlerinin artırılması maksadıyla, zırhlı araçlar üzerine monte edilerek kullanılmaya başlanmış, isabet oranları ve atış süratlerinin artırılması maksadıyla bilgisayar destekli atış kontrol ve otomatik mermi yükleme sistemleri eklenmiştir. Böylece, havanların ve personelinin bekası ve ateş desteğinin devamlılığı sağlanabilmiştir.

Ayrıca havanlara karşı gelişmiş hedef tespit cihazları ve havan tespit radarlarının yaygın olarak kullanılması, artık bir mevzide üç veya dört atıştan fazla kalınamayacağını göstermiştir. Bu kısa sürede yapılacak atışların da doğru ve süratli olması zorunludur. Tüm bu gelişmeler nitelikli personel ihtiyacını ve dolayısıyla havan eğitiminin önemini daha fazla arttırmaktadır.

2.1. Havan Eğitim Simülasyonlarının Yapısına Göre Sınıflandırılması

Havan eğitiminde kullanılan simülasyon uygulamaları yapısına göre aşağıdaki gibi sınıflandırılabilir:

- Yapısal (Constructive) Havan Eğitim Simülasyonları: Ateş destek unsurlarının lider ve karargâh personelinin eğitimi için tasarlanan bilgisayar destekli simülasyonlardır. Ateş Destek Simülatörü (Fire Support Simulation Trainer - FSST) vb.
- Sanal (Virtual) Havan Eğitim Simülasyonları: Havan Atış Sisteminin bütün unsurlarının veya bir kısmının canlandırıldığı simülasyonlardır. Havan Eğitim Simülatörleri vb.
- Canlı (Live) Havan Eğitim Simülasyonları: Havan mühimmatı hariç tüm gerçek Havan Atış Sistemi elemanlarının ve gerçek personelin kullanıldığı simülasyonlardır. Namlucuk Havan Eğitim Sistemi, Çoklu Uyumlu Lazer Çatışma Simülatörü (Multiple Integrated Laser Engagement System - MILES) vb.

2.2. Havan Eğitim Simülatörleri

Havan eğitim simülatörlerinde eğitilecek tüm personel gerçek zamanlı olarak simülatör ile etkileşerek simüle edilen harekât ortamında havan atışı icra edebilmektedir. Böylece eğitici tarafından hazırlanan senaryo içindeki durumlar ve beklenmedik hedeflere karşı personelin eğitimi sağlanabilmektedir.

Muharebe alanının tümleşik ve karmaşık olması sebebiyle personel eğitiminin yanı sıra personelin senaryo üzerinde eğitilmesini zorunlu kılar. Bu tür senaryoların sentetik muharebe ortamında simüle edilmesi personelin gerçek havan atışını öğrenebilmesi için oldukça etkili bir çözümdür.

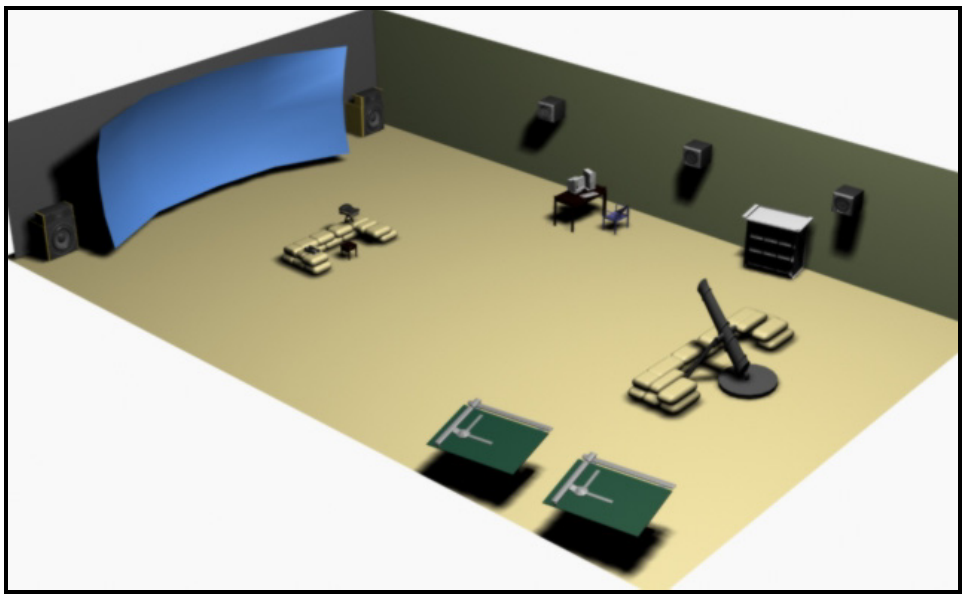
Dünyada mevcut havan eğitim simülatörlerinin genel karakteristikleri özet olarak aşağıda sunulmuştur:

- Ortak altyapı mimarilerinin kullanımı ile tekrar ve karşılıklı kullanılabilirlik ve yeni ihtiyaçlara göre modifikasyon kolaylaşmıştır.
- Atış efekt ve sesi ile 3 boyutlu muharebe ortamı yaratılmakta, havan silahı ve düşman hedefleri gerçekçi simüle edilebilmekte ve gündüz/gece, kar/yağmur, sis vb. çevre koşulları oluşturabilmektedir.
- Havan eğitim simülatörlerinin yeni kullanılmaya başladığı dönemlerde havan eğitim simülatörlerinde bulunması istenen en önemli özellikler arasında sentetik muharebe ortamını görsel ve fiziksel olarak mümkün olduğunca gerçeğe yakın yaratmak sayılmaktaydı. Ancak son zamanlarda önem kazanan özellik bir eğitim senaryosu hazırlayıp, simülasyon aracılığı ile eğitim alacak kişileri bu senaryo çerçevesinde test etmek ve bu sayede personelin eğitilmesini sağlamaktır [3].

3. MODEL ÖNERİSİ

Önerilen havan eğitim simülatörü modelinin temel özellikleri aşağıda özetlenmiş olup 3 boyutlu gösterimi Şekil 1'de sunulmuştur.

- Tekrar ve karşılıklı kullanılabilirlik için Yüksek Seviye Mimari / Simülasyon Ara Katman Yazılımı (HLA/RTI) üzerine kurulmuş ve sınırları iyi belirlenmiş bir simülasyon çerçevesi mimarisi,
- Dünya nüfusunun artması ve ekonomik yaşamdaki değişime bağlı olarak kentlerin nüfusları da giderek artmakta olup Birleşmiş Milletler (BM), 2030'da dünya nüfusunun ortalama yüzde 60'ının kentlerde yaşayacağını öngörmektedir. Askeri açıdan bu durum, gelecekteki muharebelerde yerleşim alanlarının kilit bir rol oynayacağını göstermektedir. Havanların yerleşim alanlarında kullanımı ile ilgili meskûn mahallere özgü senaryolar, arazi görüntüleri ve hedef görüntüleri,
- Hedefin sesinden yaklaşık olarak istikametini ve mesafesini tahmin edebilmek için yüksek kalitede ses çıkışı sağlayan özel olarak tasarlanmış hoparlör ses sistemi,
- Değişik coğrafi özelliklere sahip arazi görüntüleri üzerinde, 300'den fazla sabit ve hareketli hedef ile senaryolar oluşturabilme,
- Esnek, modüler ve diğer simülatör sistemleri ile yüksek etkileşimli, ilave teçhizat gerektirmeden mobil olarak taşınabilme ve 1 günde kurulabilme,
- İleri gözetleyici ile ateş idare merkezi arasında telsiz irtibatı ve veri iletişimi,
- Yüksek kalitede ses çıkışı sağlayan özel olarak tasarlanmış hoparlörlerden oluşan ses sistemi,



Şekil 1. Havan Eğitim Simülatör Modeli 3 Boyutlu Gösterimi

- Gerçek arazi koşullarına yakın, kaygan olmayan zemin,
- 210 derece yatay, 60 derece dikey gözetleme imkânı ve gece görüş cihazından görülen arazi görüntüleri,
- Eğitim sonrası değerlendirme yapabilme ve eğitim değerlendirmesi için eğitimi otomatik olarak kayıt edebilme,
- Projeksiyon ekranına yansıtılan atış alanı senaryosu üzerinde modifiye edilmiş lazer mesafe ölçme cihazı ile istikamet açısı ve mesafe ölçebilme,

4. EĞİTİM AMAÇLI SİMÜLATÖRLERDE MALİYET-ETKİNLİK ANALİZİ

Son yıllarda eğitim amaçlı simülatörlerin ekonomik yönünün değerlendirildiği çok sayıda araştırma yapılmaktadır. Söz konusu araştırmalarda kullanılan ekonomik analizler "Maliyet", "Maliyet-Yarar" ve "Maliyet-Etkinlik" olmak üzere 3 grupta sınıflandırılabilir. Günümüzde en yaygın kullanılan ekonomik analiz tipi maliyet-etkinlik analizidir.

Eğitim amaçlı simülatörler için yapılan maliyet-etkinlik analizlerinde kullanılan en genel yaklaşım, gerçek sistem kullanılarak yapılan eğitimin maliyetinin bulunması ve simülatör kullanımı ile oluşan maliyet ile karşılaştırılmasıdır. Simülatörün kendisini ne kadar zamanda amorti ettiği ise bulunan net tasarruf ile envantere alma maliyeti karşılaştırılarak bulunmaktadır. Bu süre 6 ay ile 3 yıl arasında ise maliyet-etkin olduğu varsayılmaktadır. Ancak hem maliyet hem de etkinlik boyutu birlikte değerlendirilmelidir.

4.1. Maliyet Hesaplama Modeli

Maliyet-etkinlik analizinde en kritik husus, simülatör ve gerçek sistem ile eğitim yapmanın toplam maliyetlerinin doğru ve tam olarak hesaplanmasıdır. Sadece envantere alma maliyetine göre karar vermek yeterli değildir, çünkü sistemin gerçek maliyeti envantere alma maliyeti ile sınırlı değildir. Sistemin gerçek maliyetleri tedarikinden envanterden çıkıncaya kadar olan işletme, bakım, lojistik destek vb. maliyetleri de kapsar. Bu nedenle maliyet hesaplama modeli olarak ömür devri maliyet modeli kullanılmalıdır.

Ömür devri maliyet modeli bir sistemin ömür devri boyunca getireceği maliyetleri ortaya koyan bir hesaplama sistemidir [4]. Temel olarak sistemin toplam maliyeti; Ar-Ge maliyetleri (C_S), yatırım maliyetleri (C_I), işletme ve bakım maliyetleri (C_O) ve elden çıkarma maliyetlerinin (C_D) toplamından ibarettir (1).

$$C = C_S + C_I + C_O + C_D \quad (1)$$

Eğitim amaçlı simülatörlerde Ar-Ge maliyetleri (C_S) ve elden çıkarma maliyetleri (C_D) toplam maliyetler içerisinde çok az bir yer tuttuğu için hesaplamaya dâhil edilmeyebilir.

Ömür devri maliyet modelinde eğitim simülatörü ve gerçek sistem ile yapılan eğitimlerin maliyet hesaplamaları yapılırken her iki eğitim yöntemi için aynı olan personel maliyetleri gibi maliyetler hesaplamaya katılmamalıdır. Yıllık bazda yapılabilecek örnek maliyet hesaplamaları Çizelge 1 ve Çizelge 2'de sunulmuştur.

Çizelge 1. Eğitim Simülatörü ile Yapılan Eğitimin Maliyet Hesaplaması

MALİYET KATEGORİSİ	HESAPLAMA YÖNTEMİ	AÇIKLAMALAR
Yatırım Maliyeti (C_{SI})	Envantere Alma Maliyeti + İnşaat ve Lojistik Destek Maliyeti	Hesaplanan yatırım maliyeti değeri simülatör sisteminin kullanımda olması planlanan yıla bölünerek yıllık yatırım maliyeti hesaplanır.
İşletme Maliyeti (C_{SO})	Operatör Maliyeti + Operatör Eğitim Maliyeti + Tesis İşletme Maliyeti	Tüm işletme maliyetleri yıllık olarak hesaplanır.
Bakım Maliyeti (C_{SM})	Bakım Personeli Maliyeti + Yedek Parça ve Onarım Maliyeti + Tesis Bakım Maliyeti + Bakım Eğitimi Maliyeti	Tüm bakım maliyetleri yıllık olarak hesaplanır.
Toplam Maliyet (C_S)	$C_S = C_{SI} + C_{SO} + C_{SM}$	Bulunan maliyet yıllık toplam maliyettir.

Çizelge 2. Gerçek Sistem ile Yapılan Eğitimin Maliyet Hesaplaması

MALİYET KATEGORİSİ	HESAPLAMA YÖNTEMİ	AÇIKLAMALAR
Amortisman Maliyeti (C_{GA})	Gerçek Sistemin Amortisman Maliyeti + Araç ve Lojistik Destek Malzemelerindeki Amortisman Maliyeti	Tüm amortisman maliyetleri yıllık olarak hesaplanır.
İşletme Maliyeti (C_{GO})	Gerçek Sistem İşletme Maliyeti + Akaryakıt Maliyeti	Tüm işletme maliyetleri yıllık olarak hesaplanır.
Bakım Maliyeti (C_{GM})	Bakım Personeli Maliyeti + Yedek Parça ve Onarım Maliyeti + Bakım Eğitimi Maliyeti	Tüm bakım maliyetleri yıllık olarak hesaplanır.
Toplam Maliyet (C_G)	$C_G = C_{GA} + C_{GO} + C_{GM}$	Bulunan maliyet yıllık toplam maliyettir.

4.2. Etkinlik Ölçme Modeli

Maliyet-etkinlik analizinde en zor husus, her iki eğitim yöntemi ile eğitim yapmanın etkinliğinin ölçülmesidir. Eğitimin etkinliğinin ölçülmesinde literatürde en çok kullanılan etkinlik ölçme modeli "Dört Aşamalı Kirkpatrick Modeli" olup sonuca yönelik ve deneysel bir analiz yöntemidir. Söz konusu model tepki, öğrenme, davranış ve sonuçlar olmak üzere dört değerlendirme aşamasından oluşmaktadır (Çizelge 3) [5].

Çizelge 3. Dört Aşamalı Kirkpatrick Modeli ile Eğitim Etkinliğinin Ölçülmesi

SAFHALAR		ÖLÇÜLEN NİTELİK	ÖLÇME YÖNTEMİ
1	TEPKİ	Eğitilen personelin eğitim yöntemi ile ilgili tepkisinin ölçülmesi	Eğitim öncesi ve sonrası anket, araştırma ve görüşme
2	ÖĞRENME	Eğitimin hedeflediği konuların öğrenilme seviyesinin ölçülmesi	Eğitim öncesi ve sonrası uygulamalı sınav, anket, araştırma ve görüşme
3	DAVRANIŞ	Eğitilen personelin gerçek ortamdaki davranışlarında meydana gelen değişikliğin ölçülmesi	Eğitim öncesi ve sonrası uygulanan performans ölçüm testi
4	SONUÇLAR	Sonuçların değerlendirilmesi	Performans kayıtlarının analizi

Etkinlik ölçme sürecinde ilk üç safha için elde edilecek 100 üzerinden puanlara önemine göre ağırlık faktörü verilmelidir. Bu faktörler (x , y , z) eğitim yöntemlerinden beklenen sonuçlara göre belirlenmelidir (2).

$$E = (x \times E_{Tepki}) + (y \times E_{Öğrenme}) + (z \times E_{Davranış}) \qquad (x + y + z = 1) \qquad (2)$$

4.3. Maliyet-Etkinlik Analizi

Maliyet-etkinlik analizinde, gerçek sistem ve simülatör seçenekleri iki yöntem kullanılarak karşılaştırılabilir [6];

- Maliyet-etkinlik oranlarının karşılaştırılması (C_G/E_G ile C_S/E_S),
- Maliyetteki farkın etkinlikteki farka oranlanması (C_G-C_S/E_G-E_S).

Yukarıdaki oranların karşılaştırılması sonucu eğitim simülatörünün maliyet-etkinlik oranı gerçek sisteminkinden düşük ise simülatör maliyet-etkin olarak değerlendirilebilir. Ayrıca, zaman veya risk de maliyet ve etkinlik gibi üçüncü boyut olarak hesaplamalara katılabilir.

5. ÖNERİLEN MODEL İÇİN MALİYET-ETKİNLİK ANALİZİ

5.1. Maliyet Hesaplaması

Önerilen havan eğitim simülatör modelinin yatırım, işletme ve bakım maliyetleri mevcut simülatörlerin maliyetleri dikkate alınarak derlenmiş ve analizde 20 yıllık ömür devri maliyet modeli kullanılmıştır (Çizelge 4, Çizelge 5).

(Analizde kullanılan maliyet değerleri tahminidir ve bulunabilen en doğru bilgiler derlenmeye çalışılmıştır.)

Çizelge 4. Havan Eğitim Simülatörü ile Yapılan Eğitimin Maliyet Hesaplaması

MALİYET KATEGORİSİ	YILLIK MALİYET TUTARI (YTL)		AÇIKLAMALAR
Yatırım Maliyeti (C_{SI})	Envantere Alma Maliyeti	50 000	Yaklaşık Envantere Alma Maliyeti 1 000 000 YTL ve İnşaat ve Lojistik Destek Maliyeti 500 000 YTL alınmış, planlanan kullanım süresi olan 20 yıla bölünerek yıllık yatırım maliyeti hesaplanmıştır.
	İnşaat ve Lojistik Destek Maliyeti	25 000	
İşletme Maliyeti (C_{SO})	Operatör Maliyeti (2 Kişi)	24 000	Tüm işletme maliyetleri yıllık olarak hesaplanmıştır.
	Operatör Eğitim Maliyeti	5 000	
	Tesis İşletme Maliyeti	12 000	
Bakım Maliyeti (C_{SM})	Bakım Personeli Maliyeti	24 000	Tesis Bakım Maliyeti olarak İnşaat ve Lojistik Destek Maliyetinin %2,5'i alınmıştır.
	Yedek Parça ve Onarım Maliyeti	30 000	
	Tesis Bakım Maliyeti	12 500	
	Bakım Eğitimi Maliyeti	3 000	
Toplam Maliyet (C_S)	$C_S = C_{SI} + C_{SO} + C_{SM}$	**185 500**	Bulunan maliyet yıllık toplam maliyettir.

Çizelge 5. Gerçek Havan Atışı ile Yapılan Eğitimin Maliyet Hesaplaması

MALİYET KATEGORİSİ	YILLIK MALİYET TUTARI (YTL)		AÇIKLAMALAR
Amortisman Maliyeti (C_{GA})	Gerçek Havan Atışının Namlu Ömrü Maliyeti	5 000	Yıllık bir havan namlusunun ömrü yaklaşık 1/5'lik yıpranmaktadır.
	Araç ve Lojistik Destek Malzemelerindeki Amortisman Maliyeti	6 000	
İşletme Maliyeti (C_{GO})	Mühimmat Maliyeti	850 000	Yıllık 180 kişinin eğitildiği varsayılmıştır.
	Akaryakıt Maliyeti	20 000	
Bakım Maliyeti (C_{GM})	Bakım Personeli Maliyeti	30 000	Bakım Maliyetlerine havanların bakımı da eklenmiştir.
	Yedek Parça ve Onarım Maliyeti	40 000	
	Bakım Eğitimi Maliyeti	5 000	
Toplam Maliyet (C_G)	$C_G = C_{GA} + C_{GO} + C_{GM}$	**956 000**	Bulunan maliyet yıllık toplam maliyettir.

5.2. Etkinlik Ölçümü

Havan eğitim simülatörünün etkinlik ölçümünde amaç, "Gerçek havan atışı ile yapılan eğitimin etkinlik değeri 100 kabul edilirse simülatör ile yapılan eğitimin etkinlik değeri ne kadardır?" sorusunun cevabını bulmaktır.

Önerilen model mevcut olmadığı için etkinlik değerlendirmesi "Karşılaştırma Tabanlı Tahmin" yöntemi ile yapılmıştır. Gary Klein tarafından geliştirilen yöntem, eğitimin etkinliğini mevcut olan benzer sistemlerin etkinliklerinden yola çıkarak tahmin edilmesini sağlamaktadır [7]. Obüs simülatörü için Karşılaştırma Tabanlı Tahmin yöntemi ile yapılan örnek bir çalışmada tahmini etkinlik %73 ± 2 bulunmuştur [8]. Bu değer havan eğitim simülatörü için de tahmini etkinlik değeri olarak kabul edilebilir.

Mevcut bir havan eğitim simülatörünün etkinlik ölçümü Dört Aşamalı Kirkpatrick Modeli ile Çizelge 6'da gösterildiği gibi yapılabilir ve ilk üç bölümde elde edilen etkinlik ölçütlerinin sırası ile %10, %30 ve %60 ağırlıkları ile çarpılmasıyla elde edilebilir (3).

$$E = (0{,}10 \times E_{Tepki}) + (0{,}30 \times E_{Öğrenme}) + (0{,}60 \times E_{Davranış}) \quad (3)$$

Çizelge 6. Kirkpatrick Modeli ile Havan Eğitiminin Etkinliğinin Ölçülmesi

SAFHALAR		ÖLÇÜLEN NİTELİK	ÖLÇME YÖNTEMİ
1	TEPKİ	Eğitilen personelin havan atışı yapabileceklerine olan kişisel görüşleri	Eğitim öncesi ve sonrası görüşme
2	ÖĞRENME	Eğitilen personelin havan eğitim konularını ne derece öğrendiği	Eğitim öncesi ve sonrası yetenek sınavı
3	DAVRANIŞ	Eğitilen personelin havan atış ortamındaki davranışlarında meydana gelen değişiklik	Eğitim öncesi ve sonrası uygulanan performans ölçüm testi
4	SONUÇLAR	Sonuçların değerlendirilmesi	Performans kayıtlarının analizi

5.3. Maliyet-Etkinlik Analizi

Önerilen havan eğitim simülatörü ile yapılan eğitimin maliyeti 185 500 YTL ve gerçek havan atışı ile yapılan eğitimin maliyeti ise 956 000 YTL'dir. Maliyet-etkinlik oranları karşılaştırılacak olursa;

$$C_S/E_S = 185\,500 / 73 = 2\,541 \; < \; C_G/E_G = 956\,000 / 100 = 9\,560$$

Sonuç olarak; önerilen model maliyet-etin bir çözümdür ve simülatör kendisini 1 yıldan çok daha kısa bir sürede amorti etmektedir. Kullanımda olan bir havan eğitim simülatörünün genel etkinliğinin değerlendirilmesi ile ilgili yapılan bir çalışmada simülatörün kendisini 1 yıldan daha az bir zamanda amorti ettiği görülmüştür [9].

Eğitim simülatörünün kendisini amorti etme süresi simülatörde eğitilen personel sayısı ile ters orantılıdır. Önerilen model modüler olup, ilave teçhizat gerektirmeden mobil olarak taşınabilme ve kısa sürede kurulabilme özelliğine sahiptir. Böylece daha fazla personel eğitilebilecek ve maliyet-etkinliği artıracaktır.

5.4. Gerçek Sistem/Simülatör Optimum Kullanım Oranları

Simülatörlerin gerçek sistem ile yapılan eğitimlerin yerini tamamen tutması imkânsızdır. Fakat eğitim süreci içerisinde önemli bir halkadır. Eğitim amaçlı simülatörlerde en yüksek etkinlik derecesine en düşük maliyet ve en kısa zamanda ulaştıran çözüm, simülatör ve gerçek sistemin beraber kullanıldığı yöntemdir. Örneğin; havan atış eğitiminde, gerçek mermi atarak havan atışını öğrenme yerine, simülatörde

havan atışını öğrendikten sonra gerçek mermi atma mantığı esas alınmalıdır. Simülatör eğitiminden sonra mutlaka gerçek atış yapılmalı, simülatör belli bir oranda gerçek havan atışı ile birlikte kullanılmalıdır.

Helikopter pilotlarının uçuş ve harbe hazırlık eğitimlerinde kullanılan uçuş simülatörleri için gerçek hava aracında uçuş ile simülatör uçuşu arasında optimum bir kullanım oranı belirlenmesine yönelik bir tez çalışması sonucunda eğitimlerin % 49,18'inin gerçek hava aracında ve % 50,82'sinin simülatörde yapılması sonucuna ulaşılmıştır [10].

6. SONUÇ VE DEĞERLENDİRME

Sonuç olarak, maliyet-etkinlik çalışmaları eğitim amaçlı simülatörlerin geliştirilmesinde ve tedarikinde yol gösterici olabilir. Ülkemizde eğitim simülatörlerinin maliyet boyutunun değerlendirildiği araştırmaların az olması önemli bir eksikliktir. Ekonomik kaynaklarımızın ne kadar kıt olduğu düşünülürse, yapılacak maliyet-etkinlik çalışmaları ile en yüksek etkinlik derecesine en düşük maliyet ve en kısa zamanda nasıl ulaşılabileceği araştırılmalıdır.

KAYNAKÇA

[1] Chung, C.A., (2004), Simulation Modeling Handbook - A Practical Approach, CRC Press, pp.247-289.

[2] The NATO Training Group Working Group on Individual Training And Education Developments, (1998), Simulation in Training, pp.5.

[3] İşler V. ve A. Okutanoğlu, (2005), Simülasyon Sistemlerinde Esnek Senaryo Altyapıları, USMOS 2005 Bildiriler Kitabı, Ankara, ODTÜ, pp.115.

[4] Barringer H.P., (2003), A Life Cycle Cost Summary, Gulf Publishing Company, Texas, pp. 5.

[5] Kirkpatrick, D., (1996), Revisiting Kirkpatrick's Four-Level-Model, Training & Development, January, pp. 54-57.

[6] Fox P., (1965), A Theory of Cost-Effectiveness for Military Systems Analysis, Operations Research, Vol. 13, No. 2, pp. 191-201.

[7] Klein Gary A., Johns P., Perez,R. ve A. Mirabella, (1985), Comparison-Based Prediction of Cost and Effectiveness of Training Devices: A Guidebook (ARI Research Product. 85-29) Alexandria, VA, U.S. Army. Research Institute for the Behavioral and Social Sciences.

[8] Morrison, J. E. ve C. Hammon, (2000), On Measuring the Effectiveness of Large-Scale Training Simulations, IDA paper P-3570, Alexandria VA, Institute for Defense Analysis, pp. IV28 - 30.

[9] Soykan B., (2007), Havan Eğitim Simülatör Sisteminin Genel Etkinliğinin Değerlendirilmesi (Yayınlanmamış Yüksek Lisans Tezi), Ankara, KHO Savunma Bilimleri Enstitüsü, pp.115.

[10] Lafcı A. ve T. Altınok, (2005), Kara Havacılığındaki Pilot Eğitiminde Gerçek Hava Aracı ve Tam Uçuş Simülatörleri Arasında Uçuş Eğitim Oranlarının Bulunması, USMOS 2005 Bildiriler Kitabı, Ankara, ODTÜ, pp.269.

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

SANAL BAKIM EĞİTİM SİSTEMLERİ

İsmail Hakkı ATEŞ

Kaletron A.Ş., Teknokent Silikon Bloklar Kat:1 No:19 ODTÜ, Ankara
ismail.ates@kaletron.com

ÖZ

Sabit ve döner kanatlı hava araçlarının bakım, onarım ve idamesi, gelişmekte olan teknolojiye paralel olarak, giderek daha karmaşık ve yüksek maliyetli bir süreç haline gelmektedir. Hava araçlarının ortalama servis ömründeki artış da denkleme katıldığında bu durum maliyet-etkin çözümleri zarurî kılmaktadır.

Teknolojinin gelişiminin havacılık sistemlerini daha karmaşık hale getirmesinin doğrudan bir etkisi olarak, bu sistemleri idame edecek bakım personelinin de teknik bilgi ve beceri seviyesinin yüksek olması gerekmektedir. Bu gereksinim, eğitim süreçlerini daha uzun, maliyetli ve komplike hale getirmektedir. Bütçe ve verim kıstasları değerlendirilirse, paradoksal bir biçimde teknoloji bu sorunun çözümünde etkin bir aktör olarak rol oynamaktadır.

Modelleme ve simülasyon teknolojilerindeki yüksek ivmeli gelişim, maliyet ve kalite parametreleri arasında optimum bir dengeye sahip eğitim imkânı sağlar. Eğitmen ve öğrenciler arasında iletişim ile görsel ve işitsel etkileşimi maksimize etmesi hasebiyle bakım personelinin niteliğini yükselterek servis ömrü sürecine yansıyacak bir etkiyi doğrurur.

Bu çalışmada, modelleme ve simülasyon teknolojilerinin havacılık bakım ve onarım personelinin eğitim süreçlerindeki yeri analiz edilerek, Sanal Bakım Eğitim Sistemleri tanıtılacak ve örnek çözümler incelenecektir.

Anahtar Kelimeler: bakım, eğitmen, modelleme, öğrenci, platform, simülasyon

VIRTUAL MAINTENANCE TRAINERS

ABSTRACT

The repair, maintenance and supply of fixed and rotary wing aircraft is becoming more complex and expensive process in parallel to the evolving technologies. Considering the increase in the average service life of the aircraft, it becomes mandatory to find out the cost effective solutions.

As a direct consequence of the increased complexity of the aerospace systems caused by the technological developments, it is necessary to have highly skilled and technically trained maintenance personnel to service those systems. This requirement leads the training process to be longer, more complex and more costly. Evaluating the budgetary

and effectivenes criteria, technology shows up as the main actor for the solution of this problem, in a paradoxal way.

Highly accelerated developments on modelling and simulation technologies generates training opportunities with an optimum balance between cost and quality parameters. Maximizing the audio-visual interaction and communicaton between the instructor and the trainees, creates a positive effect on the service life by increasing the quality of the maintenance personnel.

This study examines the effects of modelling and simulation technologies on aerospace maintenance personnel training process, introduces Virtual Maintenance Trainers and provides sample solutions.

Keywords: instructor, maintenance, modelling, platform, simulation, student

1. GİRİŞ

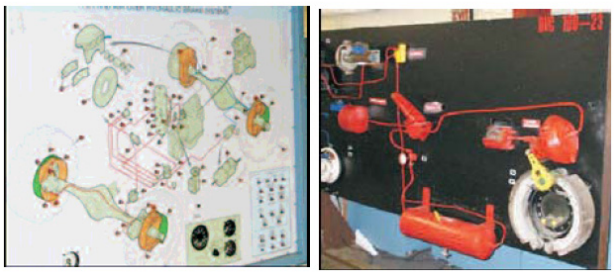
Günümüzde hava platformları; gerek gelişen teknolojilerin yarattığı yeni, tümleşik ve karmaşık sistemleri ihtiva etmesi, gerek özellikle askeri hava platformları dikkate alındığında sınırlar ötesi ve müşterek harekat ihtiyaçları, gerekse giderek kısıtlanan eğitim bütçeleri sebebiyle teknik eğitimlerinde zorluklar yaşanan sistemler özelliğine bürünmüşlerdir. Özellikle bakım ve onarımlardan sorumlu teknik personelin eğitiminde yaşanan zorlukları değerlendirdiğimizde ise; giderek karmaşıklaşan sistemlerin güvenilirlik ve hazırda bulunabilirlik ihtiyaçlarının yükselmesine karşın, bahse konu personelin uzmanlaşması için gereken sürelerin kısalması, sistemlerin azami ölçüde göreve hazır olma gereklilikleri ve eğitim maliyetlerinin sürekli aşağılara çekilmesi talepleri tespit edilmektedir.

Bütün bu göz korkutan zorlukların üstesinden gelebilmenin en etkin yolu ise eğitim programlarını ileri teknolojilerin azami ölçüde kullanılacağı bir şekilde, özellikle de sanal ekipmanların bilgisayar tabanlı üç boyutlu (3D) simülasyonlarını, Sanal Bakım Eğitim Sistemlerini, kullanacak şekilde güncellemektir.

2. BAKIM PERSONELİ EĞİTİMİNDE İZLENEN GELENEKSEL METODOLOJİLER[1]

Hava platformlarına yönelik bakım personeli eğitim metodolojisi incelendiğinde genellikle karşımıza üç aşamalı bir sistem çıkar. Bu sistemde eğitimin ilk aşaması, öğrencilerin sınıflarda eğitmenler tarafından teorik ve başlangıç eğitimine tabi tutulduğu ve genel özellikleriyle hava platformunun, alt sistemlerinin ve parçalarının, işlevleriyle ve birbirleri ile olan ilişkileri ile beraber çalışma ilkeleriyle birlikte tanıtıldığı bir aşamadır. Bu aşamada eğitim araçları olarak bakım kitaplarından ve bilgisayar destekli sunum ve yansılardan faydalanılmaktadır.

İkinci aşama genellikle donanımsal araçların da kullanıldığı ve platform ve ekipmanlara ait mokuplar ile sistem şematik tahtalarından faydalanılan ve komponentlerin de incelenbildiği bir aşamadır (Bknz. Şekil 1).



Şekil 1. Sistem mokupları

Eğitim, öğrencilerin gerçek platform ve ekipmanlarla beraber eğitildiği, yani operasyonel olarak kullanılabilecek bir uçak veya helikopterin bakım personelinin eğitimde kullanıldığı aşama ile sona erer (Bknz. Şekil 2).

Şekil 2. Operasyonel sistem üzerinde eğitim

3. SANAL BAKIM EĞİTİM SİSTEMLERİ VE GELENEKSEL METODOLOJİLERDEN SANAL BAKIM EĞİTİMİNE DÖNÜŞÜM

Günümüzde eğitimde simülasyon uygulamaları sadece pilotlar tarafından kullanılmamaktadır. Simülasyon teknolojileri, eğitimde yeni nesil trendleri temsil eden bir ayrıcalıklı ürün grubu haline dönüşmüştür. Bu teknolojilerin bakım personeli eğitiminde kullanılmasına yönelik yapılan çalışmalar neticelerini vermeye başlamış ve Sanal Bakım Eğitim Sistemleri ortaya çıkmıştır.

Sanal bakım eğitim sistemleri; öğrencilerin bakım prosedürleri, sistem faaliyetleri, servis görevleri ile onarım ve arıza bulma tekniklerini simülasyon aracılığıyla öğrendikleri sanal bir ortamda yaratılış platform, sistem, ekipman ve aletlere verilen isimdir. Sanal bakım eğitim sistemlerinde, üç boyutlu (3D) ortamda saratılan sanal platform, sistem, ekipman ve aletlerle öğrenciler yalnızca üzerinde çalıştıkları nesneleri her boyutta tanıma şansına sahip olmayıp, aynı zamanda dahili yapılarına ve işleyişlerine de etkileşimli bir şekilde vakıf olabilmektedir.

Sanal Bakım eğitim sistemleri ile geleneksel bakım eğitim sistemlerini temel hatlarıyla karşılaştırdığımızda ortaya çıkan sonuç Çizelge 1'de özetlenmektedir:

Çizelge 1. Geleneksel ve Sanal Bakım Eğitim Sistemleri Karşılaştırması

Geleneksel Bakım Eğitim Sistemleri	**Sanal Bakım Eğitim Sistemleri**
Birbirinden bağımsız üç aşamadan oluşur.	Her üç aşama da tümleşik yapıdadır.
Uygulama eğitimleri donanım tabanlıdır.	Uygulama eğitimleri ağırlıklı olarak yazılım tabanlıdır.
Uygulama eğitimleri yalnızca laboratuvar/sınıf ortamında yapılabilir.	Uygulama eğitimleri (platform üzeri eğitim hariç) için mekansal kısıtlamalar yoktur.
İdame ve işletmesi zahmetli ve maliyetlidir.	İdame ve işletmesi kolay ve düşük maliyetlidir.
Emniyet riskleri yüksektir.	Emniyet riskleri düşüktür.

Sanal Bakım Eğitim Sistemleri, bahsedilen geleneksel metodojilerin özellikle ikinci aşamasını sanal ortama aktararak etkinliğini arttırmayı ve üçüncü aşamada gerçekleştirilen platform üzerindeki eğitimin süresini azaltmayı amaçlamaktadır.

Geleneksel bakım eğitimi metdolojisinde ilk aşamada yer alan teorik ve başlangıç eğitiminin tamamiyle bilgisayar destekli eğitim ortamına dönüştürülmesi ve sanal bakım eğitiminin kalbi olarak tanımlanabilecek ikinci aşamaya entegrasyonu, sistemde ve metodolojide yaşanacak olan dönüşümün temelini oluşturmaktadır. Bu sayede eğitime tabi tutulan bakım personeli, platform üzerinde bakım eğitimi aşamasına geçmeden önce, gerek teorik gerekse uyuglamalı olarak tamamiyle etkileşimli bir eğitim ortamında yeterli bilgi ve beceri seviyesine erişmiş olacaklardır.

4. SANAL BAKIM EĞİTİM SİSTEMLERİNİN AVANTAJLARI

İçerik ve alıştırmalar, bilgiyi uygulama ve sürdürebilmede elzemdir. Buradan hareketle, sanal bakım eğitim sistemlerinin her ikisi için de son derce uygun olduğunu söylemek mümkündür. Zira, sanal bakım eğitim sistemleri;

- Öğrencilerin, hatalarından emniyetli bir şekilde ders çıkarmalarına olanak tanır,
- Operasyonel bir platformu, sistemi veya ekipmanı görevinden alıkoymadan eğitime imkan sağlar,
- Daha etkin bir eğitim ortamı yaratarak eğitim süresini kısaltır,
- Daha fazla eğitimin sınıf dışında yapılmasına destek olur,
- Eğitime daha geniş ve standart bir erişim olanağı yaratır.

Gelenkesel bir şekilde kullanılan donanıma dayanan eğitim sistemlerinin aksine sanal bakım eğitim sistemleri; platform, sistem ve ekipmanların üç boyutlu modellerinin kullanıldığı yazılım tabanlı bilgisayar destekli eğitim sistemleridir. Bu sayede öğrenciler eğitim malzemelerine her zaman ve her yerde erişim şansına sahip olarak “deneyerek

öğrenme" ilkesine uygun bir şekilde, sınırsız sayıda alıştırmalar yapabilmekte, normal işleyişleri takip edebilmekte ve gerektiğinde yaptıkları hatalarda anında uyarılar ve ipuçları alabilecek şekilde bakım ve onarım prosedürlerini tam ve eksiksiz olarak yapana dek tekrar edebilmektedirler. Özellikle "deneyerek öğrenme" ilkesinde; yüksek maliyetli ve sınırlı sayıda bulunan donanım tabanlı eğitim sistemlerine öğrencilerin sınırlı süre ile erişim imkanı ile kıyaslandığında, sanal bakım eğitim sistemlerinin avantajları bir kere daha öne çıkmaktadır.

Bütün bunların yanısıra, donanıma dayanan eğitim sistemlerin inşası ve idamesi yüksek maliyetli olup, motor gibi karmaşık sistemlerde bu maliyetler milyonlarca doları bulabilmektedir. Sanal bakım eğitim sistemlerinde ise gerek geliştirme maliyeti gerekse idame maliyeti, donanıma dayanan eğitim sistemlerine göre çok daha makul ölçülerdedir.

Sanal bakım eğitim sistemleri kullanımında yatırımın geri dönüşünü irdelediğimizde ortaya çıkan tablo avantajları tüm çıplaklığıyla göz önüne sermektedir. Sanal bakım eğitim sistemlerinde öğrenciler bakım prosedürleri ve onarım görevleri ile ilgili yetkinliklere çok daha kısa sürede daha maliyet etkin bir şekilde ulaşırken, donanıma dayanan sistemlerde kullanılacak olan işlevsel platform, sistem ve ekipmanların da görev amaçlı kullanılmasıyla söz konusu avantajlar giderek artmaktadır. Yapılan araştırmalar göstermektedir ki, sanal bakım eğitim sistemlerine yatırım yapan kurumların yatırım geri dönüşü %49 ile % 212 arasında olmaktadır. Üç boyutlu sanal bakım eğitim sistemleri sayesinde (Bknz. Şekil 3) eğitim sürelerini yarısına indiği, etkin eğitim hızının iki misline çıktığı ve sınıf ortamında yetkinliğe kolaylıkla ulaşılabildiği tespit edilmiştir.

Şekil 3. Üç boyutlu (3D) Sanal Bakım Eğitim Sistemi

5. SANAL BAKIM EĞİTİM SİSTEMLERİ TEKNOLOJİSİ VE VM*tron*™ [2]

Sanal bakım eğitim sistemleri teknolojilerini incelerken, Kaletron tarafından geliştirilen VM*tron*™ Sanal Bakım Eğitim Sistemi üzerinde örneklemeler ele alınacaktır.

VM*tron*™, geleneksel bakım eğitimi metodolojilerinde ikinci aşamada yer alan donanım tabanlı eğitim metodolojisini tamamiyle yazılım tabanlı ve sanal ortama aktaran bir sistem olarak geliştirilmekler beraber, yapısal anlamda birinci aşamada yer alan bilgisayar destekli teorik eğitimin de entegre edilmesiyle tümleşik bir platform

üzeri bakım eğitimi aşamasının öncesini oluşturmaktadır. Verilen örneklemeler her ne kadar özellikle hava platformlarına yönelikse de, VM*tron*™ kara, hava ve deniz platformları ile bu platformların ana sistemlerine yönelik bakım eğitimleri için hazırlanabilmektedir.

VM*tron*™ ürününün spesifik olarak bir platform için geliştirlmesinde ihtiyaç duyulan en önemli veriler, bahse konu platforma yönelik yapısal ve sistem tasarımları ile işletme ve bakım el kitaplarıdır. Platform ve sistem üreticilerinini yanı sıra, özellikle kullanıcı seviyesinde önemli bir eğitim sistemi olarak değerlendirilen VM*tron*™, kullanıcının ve üretcinin temin edeceği verilerin Kaletron tarafından modellenmesi ve VM*tron*™ bileşenlerine tanımlanması neticesinde, her bir platform ve sistem için özgün olarak yaratılan bir ürüne dönüşmektedir.

VM*tron*™ üç ana bileşenden oluşmaktadır, Öğrenci Bileşeni, Eğitmen Bileşeni ve bu iki bileşeni birbirine bağlayan, haberleşmelerini sağlayan ve sistem simülasyonunu sağlayan Simülasyon Bileşeni.

5.1 Öğrenci Bileşeni

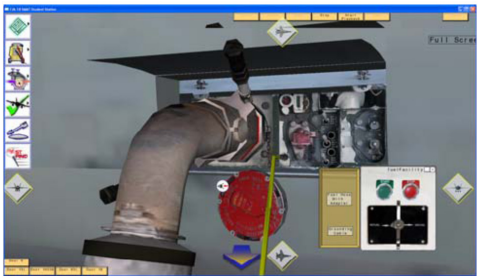
VM*tron*™'da PC tabanlı olarak sunulan öğrenci bileşenlerinde, öğrencilerin etkileşimli olarak kullanabilecekleri bir şekilde tasarlanmış, bakım eğitimi yapılacak olan platform, sistem veya ekipmanın üç boyutlu, ölçekli ve fotografik gerçeklikte modellenmiş halleri vardır.

Etkileşim, doğal ve sezgisel bir arayüzle sağlanmakta olup, masaüstü bilgisayar, kiosk tipi kurulumlar veya web tabanlı kurulumları destekler. İki veya üç dokunmatik ekrandan ve bir PDA tipi el bilgisayarından oluşan öğrenci bileşeni modeli, eğitim gereksinimlerine en uygun cevabı verebilecek konfigürasyonu oluşturur. Öğrenciler etkileşimli sistem sayesinde aşağıda örneklenen birçok uygulamaya erişim şansına sahiptirler:

- Platform, Sistem veya ekipman üzerinde üç boyutlu gezinim (3D Navigsyon)
- Kapak ve kapıları açma/kapatma
- Kabloları sökme/takma
- Konnektörleri bağlama
- Komponentleri sökme/takma/değiştirme
- Test cihazları bağlama
- Test cihazları kullanma
- Takım ve aparatları ayarlama
- Normal sistem operasyonu

Tüm bu etkileşimler, öğrenciler tarafından dokunmatik ekranlar kullanılarak gerçekleştirilir. Öğrenci bileşeni, bir görevi asgari sayıda dokunuşla gerçekleşitrebilecek bir ortamı sağlayacak ve ortamlar arasında rahat ve sorunsuz geçişler sağlayacak şekilde tasarlanmıştır. Üç boyutlu ortam sayesinde öğrenci mekansal olarak bakımını yaptığı

sistem veya ekipmana göre konumunun farkındalığında olmakla beraber; ilgili donanımların da yerleşim ve ulaşılabiliriliğinin fiziksel doğası hakkında sezgisel algılamalara sahip olmaktadır. Bu durum eğitim kalitesini yükseltirken öğrencilerin de çevrelerini çok daha gerçekçi bir ortamda keşfetmelerine yardımcı olmaktadır (Bknz. Şekil 4).



Şekil 4. Öğrenci Bileşeni Örnek Ekran Görüntüsü

5.2 Eğitmen Bileşeni

"Serbest Oynatım (free-play)" yeteneği; ortamın yalnızca belirli bir alıştırma kümesinde yer alan prosedürleri tatbikle sınırlı kalmadığının, aksine bakımı yapılacak sistem veya ekipmanın görsel ve dahili olarak tam bir simülasyonunun da var olduğunun en önemli kanıtlarından birisidir. Eğitmenler sisteme diledikleri kombinsyonlarda arızaları ekleyebilmekte ve öğrencilerin bu problemleri analizlerini gözleyebilmektedirler. Sistemler arasında yer alan karmaşık ilişkilerin varlığıyla serbest onatım özelliği; eğitmenlere problemlerin çözümüne yönelik sabit talimatları izlemenin ötesinde sezgisel anlayışlar da katmaktadır. Eğitmen bileşeninin bazı özellikleri aşağıda örneklenmektedir:

- Arıza ekleme,
- Mevcut yüzlerce alıştırma arasından seçim yapabilme,
- İlave alıştırmalar ekleyebilme,
- Öğrencilerin çalışmalarını izleme ve kaydedebilme
- Ortamı dondurabilme,
- Ortamı işaretleyebilme ve devam ettirebilme,
- Emniyet uyarı ve ikazlarını takip edebilme,
- Sistemlerin ve destek ekipmanlarının işleyişlerini serbest oynatabilme

5.3 Simülasyon Bileşeni

VM*tron*™ bünyesinde iki ayrı tipte simülasyon katmanı barındırmaktadır: Görsel Simülasyon ve Sistem Simülasyonu. Görsel Simülasyon katmanı GL Studio™ kullanılarak gerçekleştirilmektedir ve bu sayede görsel elemanların sistem

simülasyonuna muhtelif usullerde entegrasyonu sağlanmaktadır. Simülason bileşeni bakımı yapılan sistem veya ekipmanın üç boyutlu modellerini, test ve ölçü aletlerinin görsel modellerini ve öğrencilerin kullanıcı arayüzlerini kapsamaktadır (Bknz. Şekil 5).

Şekil 5. Modellenmiş sistem ve test/ölçü aletleri

Sistem simülasyonu katmanında, bakımı yapılan platformun kablo seviyesinde elektriksel, hidrolik, yakıt ve mekanik sistemlerinin simülasyonu ile test ve ölçü aletlerinin simülasyonu yer almaktadır. Örnek vermek gerekirse, bu simülasyonun kablaj bölümü, detaylı kablolama, hidrolik ve yakıt diyagramlarının bir veritabanına girilmesi ile yaratılmaktadır. Daha sonrasında da, tüm bağlantı bilgisinin simülasyonunun uygun yollardan akışı için çalıştırma modunda bir arabağlantı modelinden yararlanılmaktadır. Bu yolla her bir bireysel eleman modellenmekte ve diğer elemanlara arabağlantı modeli ile bağlanmaktadır.

Her iki simülasyon katmanı birbirlerine TCP/IP bağlantısı üzerinden çalışan sağlam ve esnek bir arayüz ile bağlanmakta olup; modülaritesi yüksek tasarım sayesinde diğer simülasyonlarla, veri kaynaklaryla ve veri ekranlarıyla entegrasyonu imkanlar dahilindedir. Çok çeşitli ve farklı uygulamalarla entegrasyon GL Studio™ ile gerçekleştirilmekte olup, yeni arayüzlerin yaratılması saatler değilse bile günler mertebesinde sürelerde mümkün olabilmektedir.

5.4 Donanım

VM*tron*™ masaüstü ve gömülü sistemler de dahil olmak üzere birçok farklı platformda çalışabilmektedir. Sistem, Windows, Linux, Solaris ve IRIX platformlarında çalışabilmekte olup; ölçeklenebilir ve tamamen taşınabilir yapısı sayesinde esnek kullanımlara ve farklı donanım ve işletim sistemi çözümlerine uygulanabilmektedir.

5.5 Geliştirme Süreci

Bakımı yapılacak sistem veya ekipmanın ortama eklenmesi sürecinde, geliştirilen ve kullanılan ileri teknolojiler sayesinde, kısa bir geliştirme süresi yeterli olmaktadır. Bakımı yapılacak sistem veya ekipmana ait üç boyutlu model, varsa mevcut üç boyutlu tasarımlardan, olmadığı zamanlarda da hedef ekipmanın fotoğraflarından yaratılmaktadır. İlave bilgilerin de model bünyesine işlenmesi ile herbir modelin tipi ile fiziksel özellikleri tanımlanabilmektedir. Örnek model tipleri aşağıda sıralanmıştır:

- Kablolar

- Kapak ve kapılar
- Konnektörler
- WRA (Weapon Replaceable Assembly)'lar
- LRU (Line Replaceable Unit)'lar
- Işıklar
- Anahtarlar
- Devre Kesiciler
- Çevirmeli Düğmeler (knob)
- Bastırmalı Düğmeler (pushbutton)
- Ekranlar
- Vidalar
- Civatalar
- Mafsallı elemanlar

Yaratılan üç boyutlu modeller, daha sonra GL Studio™ ortamına aktarılır. Herbir farklı tipteki nesne için otomatik olarak davranışlarını tanımlayan kodların yaratılması mümkün olmakla beraber, bu yaratılışta gereken detay ve sofistike özellikler de ilave olarak tanımlanabilmektedir. Örneğin, kapakların açılabilir olması, anahtarların işlevleri veya "LRU"ların sökülebilirliği gibi. Daha sonrasında nesnelerin tipi değişirse, üç boyutlu modellerde bunun doğrultusunda güncellenmekte ve yeniden aktarılmaktadır.

5.6 Mevcut Durum

Dünya genelinde VM*tron*™ ve benzeri Sanal Bakım Eğitim Sistemlerinin envanterde hangi derecede yer aldığı incelendiğinde görülmektedir ki, henüz bu tür sistemler ağırlıklı olarak askeri platformlara yönelik kullanılmakta olup, mevcut sistemlerin büyük bir çoğunluğu yalnızca birinci aşamanın etkileşimli bir hale getirildiği bilgisayar tabanlı eğitim sistemlerinin ötesine geçmemektedir. Sanal Bakım Eğitim Sistemlerinin en etkin bir şekilde kullanılacağı ilk platform, halen geliştirme aşaması devam eden "F35 Lightning II" uçağı olacak olup, VM*tron*™ ürünü ile Kaletron da bu sistemin adaylarından birisidir.

6. SONUÇ

Etkileşimli üç boyutlu sanal bakım eğitim sistemleri kurumlara ekiplerini platform, sistem ve ekipman düzeyinde istenilen yeterlilikte eğitmek hususunda zaman, verimlilik ve maliyet etkinlik açılarından ciddi avantajlar sağlamaktadır. Sonuçta ortaya çıkan en önemli kâr ise; sadece eğitim ve işletme maliyetlerinde yapılacak olan tasarruf değil, havacılık gibi hatanın ve ihmalin hiçbir zaman affının olamayacağı bir alanda daha yüksek ve daha sorumlu bir seviyede hazırlıklı olmak neticesinde kurtarılacak hayatlardır.

KAYNAKÇA

[1] Sabharwal, A. & Kaas, E., (2005), "Virtual maintenance trainers – replacing hard trainers with interactive 3D simulations", SimTecT 2005 Conference

[2] Kaletron A.Ş. (2006), "VM*tron*™ Virtual Maintenance Trainers"

MODEL TABANLI TASARIM SÜREÇLERİNDE SİMÜLASYON VE ÜRÜN GELİŞTİRMEDE ALT YÜKLENİCİLİK SİSTEMİNİN UYGULAMASI

Özgür EMEKLI [(a)], Çağlar ÜÇLER [(a)], Özalp VAYVAY [(b)]

[(a)] FIGES A.Ş., Celal Bayar Cad. No 6, 16220 Bursa,

{ozgur.emekli, caglar.ucler}@figes.com.tr

[(b)] Marmara Üniversitesi, Göztepe Kampüsü, Mühendislik Fakültesi, Istanbul,
ovayvay@eng.marmara.edu.tr

ÖZ

Günümüzün küreselleşmeye dayalı rekabetçi koşulları, tek başına hayatta kalmaya çalışan ufak ve orta büyüklükteki işletmeleri yavaş yavaş yok ederken, büyük firmaları da karlılık, piyasaya çıkış hızı ve uluslararası tedarik ve hizmet zinciri açısından zorlamaktadır. Özellikle savunma sanayi ve havacılık gibi büyük çaplı projelere hizmet veren ya da vermek isteyen firmalar izole kaldıkları takdirde piyasa şartlarındaki yapabilirlikleri tehlikeye girdiği için orta ve uzun vadede yok olmaya mahkûm gözükmektedirler. Ülkemizdeki sanayinin yapılanması incelendiği ve gelişmiş ülkeler ile karşılaştırıldığı takdirde göreceli olarak sayısı son derece az olan birkaç tane büyük firmanın ve birçok KOBİ'nin hizmet verdiği gözlemlenebilmektedir. Gerek masrafların kısılabilmesi, gerekse üretim ve teknolojinin tabana yayılarak daha sağlıklı bir ekonominin yaratılabilmesi için büyük sanayicileri hem kendi aralarında, ancak hem de küçük ve orta büyüklükteki işletmeler ile ortak çalıştırabilecek modellere geçilmesi son derece önemlidir. Bu çalışma kapsamında sadece taşeron olarak üretimin ufak kısımlarının alt yüklenicilere yaptırılmasından ziyade, çok daha önemli bir dinamik olan bilginin geliştirilmesi için genellikle savunma, havacılık ve otomotiv alanlarında kullanılmakta olan sistem mühendisliği kavramlarından birisi olan Model Tabanlı Tasarım Süreçleri ve bunların kullanılarak efektif işleyebilen Alt Yüklenicilik Sisteminin oluşturulması incelenmektedir.

Anahtar Kelimeler: Alt yüklenicilik, benzetim, model tabanlı tasarım, sistem mühendisliği, ürün geliştirme.

SIMULATON IN MODEL BASED DESIGN AND APPLICATION OF A SUBCONTRACTOR MODEL IN PRODUCT DEVELOPMENT

ABSTRACT

Globalization and today's competitive market circumstances are eliminating many small and medium sized enterprises (SME's), which are operating as stand-alone. Moreover bigger enterprises are forced to enhance their profitability, cycle times and international supply- and service-chains. Especially isolated companies trying to survive in defense and aerospace segment are going to disappear with time, when they cannot cooperate with bigger partners. The industry of Turkey consists of a lot of SME's and a small number of big sized enterprises. Therefore it is from vital importance that bigger enterprises start applying a model for cooperation between themselves by integrating also SME's to reach a healthy and reliable economic structure, where they can compete against global players. This work focuses not on outsourcing of basic production steps, but it concentrates on a much more important dynamic: a system for the subcontracting of the development of knowledge, using systems engineering principles and model based design like in defense, aerospace and automotive engineering.

Key Words: Model based design, subcontractor, product development, simulation, systems engineering.

1. GİRİŞ

Yoğun bürokrasi, yatırım ortamındaki belirsizlikler, yetersiz Ar-Ge harcamaları, tasarım ve marka yaratmadaki yetersizlikler, finansman kaynaklarına erişimde engeller, etkin olmayan pazarlama hizmetleri, düşük ölçekli kapasiteler, düşük verimlilik ve kalite problemleri (bkz. [1]) ile yaşamak zorunda olan Türk şirketleri uluslar arası arenada başarılı olabilmek için çeşitli önlemler almaya başlamışlardır. Firmalar kalite yönetim sistemlerinin standartlaştırılması ve seri üretimi mekanize hale getirme yönünde çalışıp, tasarım ve dolayısı ile piyasaya arz sürelerinin kısaltılması için bilgisayar destekli ERP, PLM, CAD ve CAE teknolojilerinin olabildiğince kullanılması için yatırımlar yapmaktadır.

Tüm bu çalışmaların neticesinde katma değeri yüksek, mühendislik çalışması gerektiren ürünleri piyasaya sunabilir hale gelen Türk Sanayisi bir dönüm noktasına yaklaşmaktadır: Artık şirketler çalışma düzenlerini rayına oturttuktan sonra yenilikçi ürünlere yönlenerek yeni kazanımlar gerçekleştirmeyi istemektedirler. Ancak tüm yenilikçi ürünler ve farklı ürün gamları beraberinde yeni yatırımları da gerekli hale getirmekte olduğundan, firma içinde ne kadar yatırım gerçekleştirilmesi gerektiği ve alt yüklenicilerin hangi derecede kullanılarak teknoloji ve üretim açığı eksiğinin giderilmesi gerektiği sorusu gündeme gelmektedir.

Esasında bu düşünce dünyada çok yeni bir olgu değildir. Şirketlerin yatırım yaparak geliştirdikleri kabiliyetlerinin sınırları, ilk defa 1937 yılında Coase'in araştırmaları ile önem kazanmıştır. Dolayısı ile yakın geçmişimizde teorik ve deneysel yaklaşımlarda "yap ya da al" sorusu her zaman için bir odak noktası niteliğinde olmuştur [2]. Özellikle

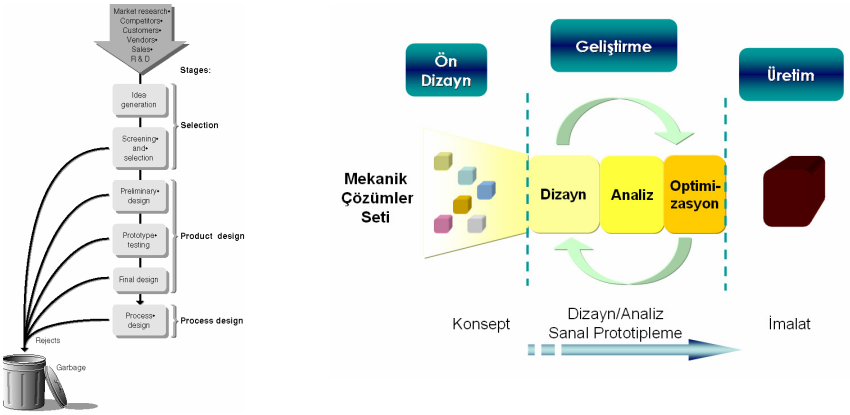
elektronik haberleşme teknolojilerindeki gelişmeler ile çok uluslu şirketlerin dikey örgütlenmeleri kapsamındaki alt yüklenici sistemleri hakkında literatürde birçok araştırmaya rastlamak mümkündür. Yapılan araştırmaların yakınsadığı nokta, genellikle şirketlerdeki iletişim ağlarının ve şirketler arası çalışmanın verimlilik derecesinin birinci dereceden önemli olduğu yönünde olmuştur [2]. Dolayısı ile hangi alanda çalışılırsa çalışılsın, ya da firmalar hangi büyüklükte olursa olsun, alt yüklenicilik -ya da firmaların eşdeğerde olması halinde ortaklıklar- kaçınılmaz olarak beraberinde iletişim problemlerini getirmektedir. Bu çalışmanın ana konusu, iletişimde standart araçlar kullanılarak bu problemlerin daha oluşmadan önlenmesidir.

2. ALT YÜKLENİCİLİK VE ÜRÜN GELİŞTİRME

Dünyada alt yüklenicilik ve ortaklık açısından en etkin çalışmalara havacılık ve savunma sanayinde rastlamak mümkündür. Bu sektör gerek yapılan işlerin karmaşıklığı, gerekse üretimin yüksek kapasite gerektirmesi sebebi ile dünya devlerinin birleştiği ve ortak projelere imza attığı bir sektör olmuştur. Bunun başlıca sebepleri arasında NATO ülkelerinin savunma bütçelerini kısmaları ve ihracat serbestisinin politik ve askeri sebeplerle kısıtlı olmasına dayalı olarak satış adetlerinin düşmesi ve karlılık ve verimliliğin ön plana çıkması yer almaktadır. Dolayısıyla ile denilebilir ki şu anda ülkemizde karşılaştığımız ekonomik sebeplere dayalı bariyerlerin benzerleri yurtdışı savunma sanayisi için de vardır ve orada kullanılmakta olan alt yüklenicilik sisteminin bir benzeri ülkemizde de uygulanabilir ([3]).

Ülkemizde şu anda ne yazık ki ArGe çalışmaları birkaç yeni girişimin dışında alt yüklenicilere verilmemektedir. Belki bir yapının hesabı, tasarımı ya da tasarım doğrulaması yani ürün geliştirme (ÜrGe) iş paketleri olarak dışarı verilmektedir ancak ArGe'nin paylaşımı henüz emekleme safhasındadır. Buna sebep olarak tesis güvenliği vb. etkenler gösterilmektedir. Oysa en önemli sıkıntı karşılıklı güven ve esasında firmaların bilgi alışverişini bütünleşik olarak gerçekleştirmelerini sağlayacak bir model eksikliğidir. Yapılan araştırmalar gerek entegratör firmanın, gerekse alt yüklenici konumundaki firmanın ortak olarak ArGe için alt yapı yatırımı yapması durumunda daha başarılı olunduğunu göstermektedir (bkz. [4]). IBM'in gerçekleştirmiş olduğu bir araştırma bu amaçla alt yüklenici ve ana firmaların bütünleşik bilgisayar teknolojilerine yaptıkları yatırımlar ile birçok iyileştirmelere ulaşabildiğini göstermiştir (bkz. [5]).

Dolayısı ile her şeyden önce bilişim teknolojilerinin doğru bir şekilde firma bünyesine emdirilmesi gerekmektedir. Bunun da iki yönü olduğu düşünülebilir. Her şeyden önce klasik anlamdaki doğrusal ürün geliştirme sürecinden (bkz. Şekil 1 ve [6]) kurtulup analiz sistemlerinin kullanıldığı çözümlere yönlenilmesi gerekmektedir. Bunun için bilgisayar destekli çizim (CAD), sonlu elemanlar, akışkanlar dinamiği, sistem modelleme gibi teknolojileri içeren, prototip maliyetlerini düşüren bilgisayar destekli mühendislik (CAE) kavramı gerek ana entegratör firmada oturtulmalı, gerekse alt yüklenicilerde kullanılmalıdır.



Şekil 1. Klasik Ürün Projelendirme Süreci ve Çevrimsel Ürün Geliştirme Süreci

İkinci önemli nokta ise tüm ürün geliştirme sürecine yukarıdan bakabilmeyi mümkün hale getiren benzetim araçlarının düşünülmesidir. Günümüzde gerek ana entegratör firmaların, gerekse orta seviyedeki alt yüklenicilerin giderek daha fazla sistem bazında hazır yarı mamülleri (motorlar, uçak iskeleti, kanatlar, radar, navigasyon sistemi vb.) aldıkları yani mühendislik çalışmalarını ve dolayısı ile ÜrGe ve ArGe'yi giderek daha alt seviyelere alt sistem entegratörlerine ittelediği tespit edilebilir (bkz. [7]). Dolayısı ile tüm konulara hakim olunabilen, alt sistemlerin ihtiyaçlarının doğru bir şekilde iletilebildiği, alt sistemlerin birbiri ile uyumunun benzetilebildiği bir ürün geliştirme platformu son derece gereklidir. Bu amaçla bir sonraki bölümde bilgisayar destekli mühendislik alanındaki ihtiyaçlar ve Model Tabanlı Tasarıma dayalı Ürün Geliştirme Yaklaşımı incelenecektir.

3. ARGE VE ÜRGE'DEKİ BİLGİSAYARLI MÜHENDİSLİK SÜREÇLERİNDE İHTİYAÇLAR VE ÇÖZÜMLERİ

Savunma sanayi projeleri, genelde farklı disiplinlerden grupların birlikte çalışmasını gerektirmektedir. Bu nedenle firma içi ve firmalar arası proje gruplarının iletişimi önem kazanır. Bu kişilerin terminolojileri, anlayışları ve kullandıkları yazılımlar farklı olduğu için, gruplar arası ortak bir iletişim yöntemine ihtiyaç vardır. (bkz. [8]).

Ayrıca sürekli bir gelişim için, önceki projelerden deneyimlerin ileriye taşınabilirliği çok önemlidir. Tasarlanan bileşenlerin, sonraki projelerde de aynen veya ufak değişikliklerle kullanılabilmesi mümkündür. Kullanılan yazılımların projeye özel değil, konuya özel olması ve yazılımların özelleştirilebilmesi, proje kalitesi ve güvenliği için bir gereksinimdir.

Matlab & Simulink ortamı literatürde yer alan IDEF, RID, CPN ve Flow Chart yöntemlerinin [9] bir bütünü izlenimini yaratmaktadır. Tüm bu modellerin ana prensibi, alt sistemlerin bir kapalı kutu şeklinde olarak tek başına tasarlanabilmesi, böylece girdi ve çıktıların net olarak tanımlanabilmesidir [10]. Bu sayede alt sistemlerin tasarımında giriş çıkış verileri sabitlendiği için, kullanılan özel yazılımların formatları ne olursa

olsun, bu özel yazılımlar ile tespit edilen alt sistemlerin karakteristik özelliklerinin tanımlandığı sistem modellerinin birbirleri ile konuşmaları sağlanabilmektedir.

3.1 Model gelişimi ve benzetimi

Oluşturulan model gerçeği ne kadar yansıtıyor sorusuna yanıt almak için, hem gereksinim yönetimine, hem de sürekli bilgisayar benzetimine ihtiyaç duyulur. Modelin gerçeğe biraz daha yaklaşması için, yapılan her benzetim, model üzerinde bir değişikliği gerektirir. Dolayısıyla kullanılacak yazılımların buna olanak vermesi gereklidir. Burada iki çeşit yazılım kullanıldığının bir kez daha altının çizilmesinde fayda olabilir:

- Genel sistem modeli için yukarıdan bakışı sağlayan sistem bütünleştirme platformu olan yazılımlar (matematiksel model ve sistem platformu)
- Alt sistemlerin ya da bileşenlerin özelliklerinin tespiti için kullanılarak fiziksel özellikleri ve tepki eşitliklerini belirlemeye yarayan bilgisayar destekli mühendislik modelleri (FEA, CFD vb.).

Bu her iki yazılım çeşidi birbirleri ile konuşturularak sistem modelinde gerekli olan tüm ilgili özelliklerin tespiti ve alt modellere yerleştirilmesi mümkün olabilmektedir.

3.2 Test ve Doğrulama Aşaması

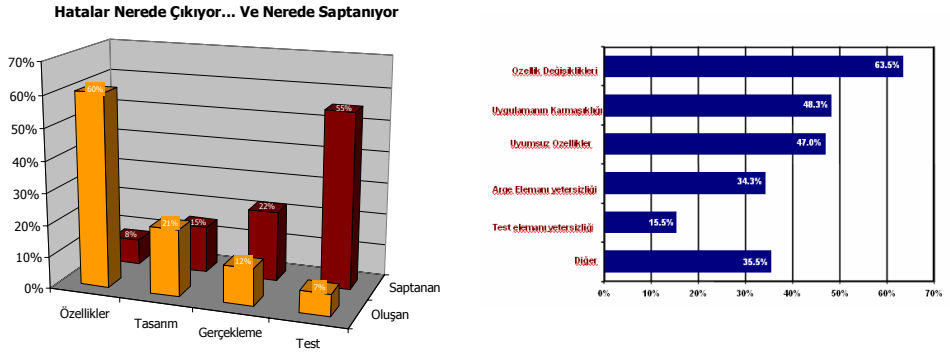
Sürekli benzetimlerle gelişen model, son aşamada gereksinimlere uygunluk açısından test edilmeli ve doğrulanmalıdır. Bir savunma sanayi projesinde, yapılması gerekecek test sayısı, projelerin hassasiyeti ve çapı dolayısıyla yüksek rakamlara ulaşmaktadır. Bu süreç, her bir benzetimin test senaryosuna uygun şekilde yapılıp, sonuçların raporlanmasını içerir ve bu sürecin tamamen otomatikleştirilmesi doğrulama süresi açısından büyük önem taşır.

Bazı durumlarda ortaya çıkartılmak istenen ürünün bir kısmı ya da hepsi yazılım olabilir. Dolayısı ile yazılımların testleri ve doğrulamaları, ya da dahil edildikleri ve kontrol ettikleri sistemler içerisinde denenmeleri ayrı bir disiplin olarak düşünülebilir. Bu alan bu çalışma kapsamında ayrıca irdelenmiştir.

3.3 ArGe'de yazılım mühendisliği aşamaları

Günümüzde neredeyse tüm projeler elektronik bileşenler içermektedir. İşte bu aşamada yazılım mühendisliği kavramı devreye girmektedir. Yazılım mühendisleri, sistemi (algoritmayı) tasarlayan mühendislerin belirttiği nitelikleri sağlayacak kodlar yazmak durumundadırlar. Yazılım mühendisleri, özleri bakımından, sistemleri tasarlayan disiplinlerden farklılık gösterirler. Dolayısı ile yazılım mühendislerinin, sistem için belirtilen tasarım özelliklerini eksiksiz ve hatasız anlayabilecekleri bir yönteme ihtiyaç vardır. Sistemin ne yapacağını doğru anlayamamış bir yazılım mühendisinin yazacağı kodların da istenileni yansıtması beklenemez.

Yazılan kodların, tasarlanan sisteme uygun olarak çalıştığının doğrulanması gerekir.
Bu süreçte, oluşan hataların çoğu ne yazık ki özelliklerin tam olarak tanımlanamamasından kaynaklanmaktadır ve maalesef bu hatalar genelde sürecin son adımlarında ancak tespit edilebilmektedir (bkz Şekil 2).



Şekil 2. Yazılım Mühendisliğinde Hata Kaynakları [9]

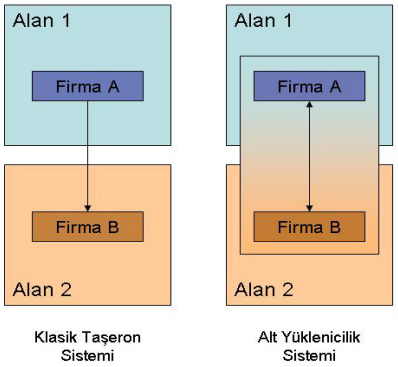
4. UYDU ŞİRKETLER YAPILANMASI İÇİN STRATEJİK ÇÖZÜM ÖNERİSİ; TAVSİYE EDİLEN ALT YÜKLENİCİ MODELİ

Günümüzde süregelen büyük projele bakıldığı zaman büyük firmaların alt yüklenicili ürün geliştirme platformları oluşturduğunu gözlemlemek mümkündür. Örnek olarak Eurofighter projesinde oluşturulan "Sanal Şirket" gösterilebilir. Nihayi ürünün sadece %37,5'i BAE Systems tarafından geliştirilirken geri kalan alt sistemlerin gerek üretimi, gerekse geliştirmesi ortak ve alt yükleniciler tarafından gerçekleştirilmektedir. Aynı şekilde bu ortakların da alt yüklenicileri en son bileşen bazına inilene kadar devam etmektedir (bkz. [10]).

Ayrıca teknolojik açıdan bakıldığı zaman Amerika'daki bütünleşik sanayi modeli ile karşılaştırıldığında optimum olmaktan uzak Avrupa Sanayisi de özellikle ArGe alanında mühendislik havuzları üretebilmek ve sivil ve askeri sanayiden bir görevdeşlik oluşturabilmek için yoğun bir çaba harcamaktadır [3].

Benzer şekilde, Türkiye'de de alt yüklenicilik modeli vasıtası ile bir görevdeşliğe ulaşılmalı ve orta çaplı firmalar ile büyük firmaların entegre olabilecekleri bir ortam yaratılabilmelidir. Bunun gerçek olabilmesi için sadece basit bir işin gönderildiği bir sistemden ziyade, alt sistem geliştirmesinin iş olarak verilmesi ve tüm sistemle bütünleşmenin sağlanması gerekmektedir. Dolayısı ile ortak bir platform en önemli ihtiyaçlardan birisi olarak öne çıkmaktadır (Şekil 3).

Üretim firmaları incelendiği zaman içerisinde yer alan çeşitli süreçlerin izole edildiği çalışma grupları kendi içlerinde farklı firmalar gibi etkileşim göstermektedir (bkz. [12]). Dolayısı ile denilebilir ki eğer entegrasyon platformu olarak kullanılan yazılımlar doğru uygulanırsa, şirketler farklı olmalarına rağmen sanki aynı şirketin bağımsız çalışma grupları gibi bir ilişki içine girebilir.



Şekil 3. Organizasyon Formları

Bu çalışmalar kapsamında, bizim önerimiz:

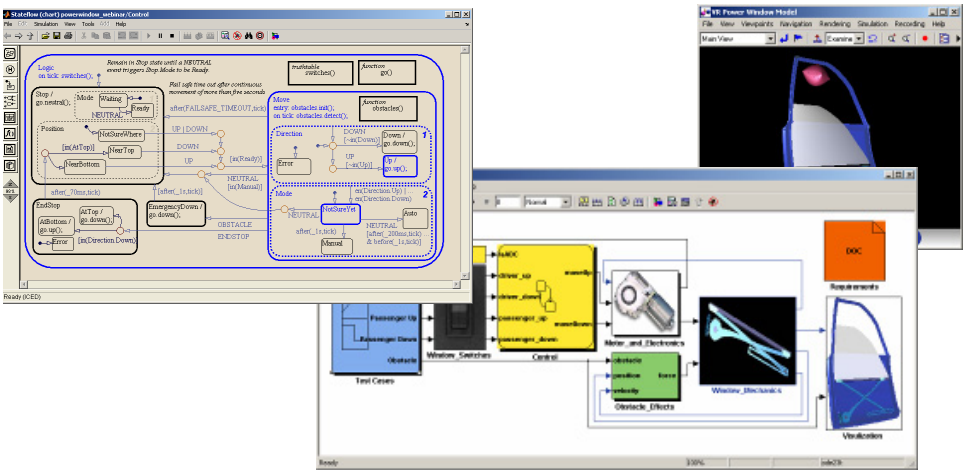
1. Sistem Mühendisliği Prensiplerinin benimsenmesi,
2. Alt yüklenicilerin sanki firmanın bir grubu gibi süreçlere dahil edilmesi,
3. Model Tabanlı Tasarımın uygulanması
4. Bu amaca uygun bir platform olarak Matlab Simulink'in kullanılmasıdır.

Matlab&Simulink kullanımına bir örnek Şekil 4'de görülebilir (bkz. [11]).

Burada kullanılan model, ihtiyaçların belirlenmesi aşamasında sistemin idealleştirilmiş bir gösterimidir. Model, sabit noktalı ve zamanlama davranışı da dâhil, yazılım ve donanım uygulamalarının belirlenmesi için gerekli bilgileri içeren bir "çalıştırılabilir özellik" haline gelene kadar ayrıntılar tasarım geliştirmeleri olarak eklenir. Model ayrıca özelliklerin taşındığı biricik iletişim aracı olduğu için (klasik yöntemde belgelerin karşılığı) olduğundan tasarımlar, hatalar tanıtılmaksızın ve elle yapılan en az çabayla masaüstünden pilot eğitim benzetimleri haline getirilebilir.

Model çalıştırılabilir nitelik olduğu için mühendisler gerçek zamanlı prototipleme ve hedef sisteme gömme için modelden otomatik kod üretebilirler. Modelde olduğu gibi kod da herhangi bir noktada test edilebilir ve doğrulanabilir. Model uyarlanarak ve kod yeniden oluşturulup model ile kod arasındaki teknik özellik uyumluluğu sürdürülerek problemler kolaylıkla düzeltilebilir.

Yeni nesil bir grafiksel araç olan Simulink, etkileşimli grafiksel editörler, grafiksel hata ayıklayıcılar ve model analizi ve kontrolü için araçlar içeren bütünleştirilmiş modelleme ortamı sağladığı ve uyarlanabilir blok kütüphaneleri sunduğu için tercih edilebilir. Böylece mühendisler hassas tasarım, benzetim, uygulama çalışmaları yapıp ve kontrol, işaret işleme, haberleşme ve diğer zamana bağlı sistemleri test edebilirler. Bu yaklaşımda tasarımcı, modelleri tasarım bileşenleri şeklinde hiyerarşiler halinde parçalara ayırabilir ve onları rasgele boyut ve yapıdaki fonksiyonel elementler haline getirebilir. Aslına bağlı kalarak basitçe bir model elemanının yerine diğerini koyarak çok farklı seviyelerde model elde edilebilir. Bu yaklaşım çeşitli tasarım gruplarının alt sistem tasarımlarının düzeltilmesi ve optimize edilmesi için paralel çalışmalarına izin vererek büyük ölçekli sistem geliştirilmesinde verimliliği arttırır.



Şekil 4. Pencere açma sisteminin model tabanlı sistem tasarımı

Önerdiğimiz Model Tabanlı Tasarım'ı benimsemiş olan organizasyonlar görülmeye değer sonuçlar raporlamışlardır. Örnek olarak kusursuz işleyen otomatik oluşturulmuş binlerce satırdan oluşan uçuş kontrol kodu için, verimliliğin %500 oranında iyileştirilmesiyle, tasarım zamanı yarıya düşmüştür (Bkz. [12]).

4.1 Projede çalışan takımların iletişiminin kuvvetlendirilmesi

Klasik tasarım sürecinde, ihtiyaçlar, özellikler, test senaryoları, veri tanımları ve diğer proje detayları kağıt veya bilgisayar belgeleri üzerinde taşınır. Burada, kişiden kişiye göre değişebilecek anlamlar dolayısıyla belirsizlik, bir anlatımın birden çok şekilde anlaşılabilmesi, yanlış yorumlanması gibi aksaklıklar kaçınılmazdır. Bu aksaklıklar, fark edildiği yerde, tekrar geriye dönülerek yok edilebilir. Ancak bu dönüş, aksaklığın tespit aşamasına göre proje için taşınamaz maliyet ve süre artışına neden olabilir. Özellikle yanlış anlaşılmalardan kaynaklanan hatalar, üretim aşamasına kadar taşınıp, ancak orada tespit edilebildiğinde proje için geri dönüşü olmayan sonuçlar ortaya çıkabilir (bkz. Şekil 2). Model Tabanlı Tasarım kavramında iletişim belgeler üzerinden değil, doğrudan çalıştırılabilir grafiksel bir model üzerinden yapılır. "Model Tabanlı" sözü buradan kaynaklanmaktadır. Yazılar yerine, çalıştırılabilir grafiksel diyagramların kullanılması takımlar içindeki ve arasındaki iletişimi dramatik derecede artırmaktadır.

4.2 Tasarımın Sürecinin Sürekli Benzetim Yapılarak İlerlemesi

Model Tabanlı Tasarım'da, en basitten başlanarak grafiksel model oluşturulur. Daha sonra proje ihtiyaçlarına göre adım adım yeni bloklar model üzerine eklenir. Model, bilgisayar ortamında çalıştırılabilir olduğu için, her adımda benzetim yapmak ve karşılaşılan eksiklikleri, hataları kolayca model üzerinde değişiklik yaparak düzeltmek mümkün olur. Bu ise, tasarımda çıkacak hataların ürün geliştirme sürecinin erken fazlarında tespit edilmesini ve düzeltilmesini sağlar.

Tasarımın gelişiminde, proje gereksinimleri belirleyicidir. Model Tabanlı Tasarım'da, modele eklenen her bir blok veya blok grubu için, gereksinim belgeleri

tanımlanabilmekte, hatta bu belge üstünde hangi kısımların ilişkili olduğu dahi belirtilebilmektedir. Böylece tasarım gelişiminde atılan her adımın geçmişi, gereksinim belgelerinden takip edilebilmektedir.

4.3 Sürekli Test ve Doğrulama

Model Tabanlı Tasarım, her aşamada çalıştırılan model üzerinde geliştiği için, aslında tasarımın testi sürekli yapılmaktadır. Ancak son kısımda, mutlaka iyi/kötü hal senaryoları yaratılarak, tüm tasarımın doğru çalıştığının test edilmesi ve tasarımın doğrulanarak, onaylanması kaçınılmazdır. Burada, belirlenen test senaryoları tamamen otomatik olarak model üzerine uygulanır. Sonuçlar yine otomatik olarak rapor şeklinde üretilir.

4.4 Modelden Otomatik Olarak Kod Üretilmesi

Gömülü yazılım içeren tasarımlar için kod çok önemlidir. Model Tabanlı Tasarım'da can alıcı nokta, sürekli benzetimlerle adım adım doğrulanarak geliştirilmiş modelden, tamamen otomatik olarak kod üretilebilmesidir. Burada önemli olan, kodun modele ilişkin tüm özellikleri yüzde yüz doğru olarak yansıtmakta olacağıdır. Bu ise, kodlama sürecinde yapılabilecek hataları önler.

5. SONUÇ

Piyasa dinamikleri ve ülkemizin de içinde bulunduğu karlılık baskısı firmaları giderek daha az maliyet ile çalışabilecekleri ortamlara itmektedir. Maliyetleri kısarak ArGe yapmanın bir yolu, alt yüklenicilik kavramını uygulamak, böylece konusunda uzman kuruluşların bilgi birikimlerini kullanarak bir görevdeşlik oluşturmaktır. Buna ek olarak ilerleyen teknolojinin olanaklarını kullanmak, özellikle benzetim alanında çalışmalar yaparak, bu sayede prototip masraflarını düşürmektir.

Bu çalışma kapsamında tüm bu etkenlerin ışığı altında büyük ve orta çaplı firmaların bir arada çalışabilecekleri bir alt yüklenicilik modeli önerilmiştir. Model tabanlı tasarım kavramları ve sistem modelleme araçları arasından Matlab&Simulink'in kullanımı ile firmalar arası bütünleşmenin olanağı ve bu sayede birden fazla noktada aynı anda ürün geliştirme ve araştırma projelerini gerçekleştirme potansiyeli gösterilmiştir.

Buna ek olarak araştırmacılar özellikle ülkeler arası işbirlikleri şart koşan alt yüklenicilik modellerinin arttığına dikkat çekmektedirler (bkz. [13]). Dolayısı ile dünya üzerinde hâlihazırda kullanılan model tabanlı tasarım süreçleri ve sistem mühendisliği yaklaşımlarını uygulamak, yeni yurt dışı pazarlarına açılmak adına da firmalara katma değer sağlayabilecektir.

KAYNAKÇA

[1] DPT, (2003), Türkiye Sanayi Politikası Raporu, (AB Üyeliğine Doğru)

[2] OKAMOTO Y., (2006), Integration versus Outsourcing in Stable Industry Equilibrium with Communication Networks, DISCUSSION PAPER No. 54, © 2006 by Institute of Developing Economies, JETRO

[3] SCHMITT B.(2007), From Cooperation to Integration: Defence and Aerospace Industries in Europe, EU-ISS Chaillot Paper 40, July 2000, http://aei.pitt.edu/500/01/chai40e.pdf

[4] ROSSİNİ G.,LAMBERTİNİ L., (2003), Endogeneous outsourcing and vertical integration with process R&D, Dipartimento Scienze Economiche, Università di Bologna, in its series Working Papers, number 487, http://ideas.repec.org/p/bol/bodewp/487.html

[5] IBM Business Consulting Services (2007), Aerospace and Defence, Transformation, White Paper, ©Copyright IBM Corporation 2002, www.ibm.com

[6] Meredith and Shafer, Operations Management for MBAs, 2nd Edition

[7] McALEESE J., (1997), Aerospace Industry Consolidation, News from the Front Lines, 1997 OEM/Supplier Symposium

[8] Aberg R.(2004) ,S. Gage, Strategy for Successful Enterprise-Wide Modeling and Simulation with COTS Software, AIAA

[9] Source: Software Development Times,(2002) Venture Development Corp. data

[10] Mason H.(2006),PLM for the Eurofighter Typhoon, NDIA proceedings.ndia.org/3AF6/Howard_Mason_2.pdf

[11] Barnard A.P.(2005), Software Development Principles Applied to Graphical Model Development, AIAA, Paper Nr #5888

[12] Yanik P., (2004), Migration from Simulation to Verification with ModelSim®, EDA Tech Forum, Newton M

[13] ITOH H.(2007), The Theories of International Outsourcing and Integration: A Theoretical Overview from the Perspective of Organizational Economics, Conference Paper, Japan Center for Economic Research, Tokyo

A4 OTURUMU:

HLA ALTYAPILARI-I

HLA TABANLI DAĞITIK SİMÜLASYONLARDA FEDERE YÖNETİM ALTYAPISI
Tolga BAŞTÜRK, Yasir Kurtuluş AVCIBAŞI, Onur DESTANOĞLU, Ömer EROĞLU, Hürcan Gökçe SOLTER, F. Erdoğan SEVİLGEN, İsa TAŞDELEN, Oğuz DİKENELLİ

FEDERASYONLAR İÇERİSİNDE DİNAMİK FEDERE KÜMELENMELERİ VE ZAMAN YÖNETİMİ
Aydın OKUTANOĞLU, Müslim BOZYİĞİT

SİMÜLASYON MODELLEME ALTYAPISI (SİMA) : DEVS TABANLI HİYERARŞİK VE MODÜLER BİR MODELLEME VE KOŞUM ALTYAPISI
Ahmet KARA, Doruk BOZAĞAÇ, M. Nedim ALPDEMİR

ÖLÇEKLENDİRİLEBİLİR HLA UYUMLU SİMÜLASYONLAR İÇİN UYGULAMA ÇATISI: SİMALT
Göktuğ F. GÖKDOĞAN, Günay ÖZKAN, Turgay ÇELİK

HLA TABANLI DAĞITIK SİMÜLASYONLARDA FEDERE YÖNETİM ALTYAPISI

Tolga BAŞTÜRK, Yasir Kurtuluş AVCIBAŞI, Onur DESTANOĞLU, Ömer EROĞLU, Hürcan Gökçe SOLTER, F. Erdoğan SEVİLGEN, İsa TAŞDELEN, Oğuz DİKENELLİ

TÜBİTAK MAM, Bilişim Teknolojileri Enstitüsü, 41470, KOCAELİ

{yasir.avcibasi, tolga.basturk, onur.destanoglu, omer.eroglu, hurcan.solter, erdogan.sevilgen, isa.tasdelen, oguz.dikenelli}@bte.mam.gov.tr

ÖZ

Bu bildiride TÜBİTAK Marmara Araştırma Merkezi Bilişim Teknolojileri Enstitüsü tarafından hazırlanan HLA[1] tabanlı dağıtık benzetim geliştirme ve entegre etme sürecinde kullanılabilecek bir altyapı olan Federe Yönetim Altyapısını (FYA) tanıtılmaktadır. Bu katmanın amacı etki alanı (domain) kullanıcılarını, kullanılacak dağıtık benzetim mimarisinden soyutlayarak kendi mimarilerini oluşturmalarına imkân sağlamak ve dağıtık benzetim ortamından kaynaklanan programlama yükünü en aza indirerek etki alanı kullanıcısının sadece kendi görev uzayına odaklanmasına imkân vermektir.

Anahtar Kelimeler: Arakatman yazılımı, HLA, Federe Yönetim Altyapısı, Dağıtık Benzetim, FOM

FEDERATE MANAGEMENT INFRASTRUCTURE IN HLA BASED DISTRIBUTED SIMULATIONS

ABSTRACT

This paper addresses Federate Management Infrastructure(FYA), which can be used in HLA based distributed simulation development and integration processes. The purposes of this layer are to allow domain users to design their own architectures by being abstracted from distributed simulation architecture and to allow domain users to focus their attention only on their workspaces by minimizing programming burden originating from distributed simulation environment. In this work, a national solution prepared by TUBITAK Marmara Research Center Information Technologies Institute, will be discussed, and the modules it consists of will be examined.

Keywords: Middleware, HLA, Federate Management Infrastructure, Distributed Simulation, FOM

1. GİRİŞ

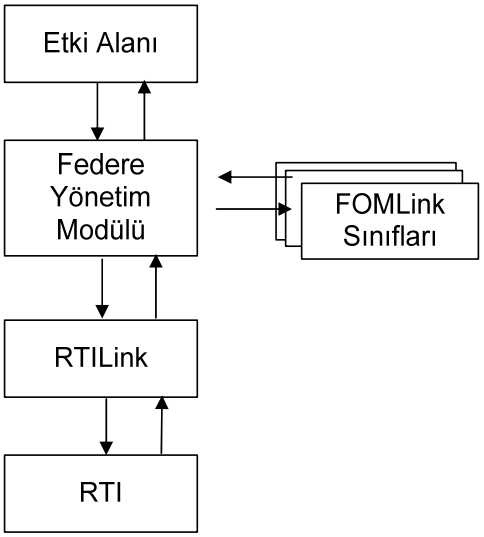
Teknolojideki hızlı ilerlemeler, güçlü bilgisayarların geliştirilmesi, benzetim sistemlerinden beklenenleri arttırmıştır. Bununla birlikte aynı ortamda koşan benzetimlerin birlikte çalışılabilirliğinin sağlanması da son yıllarda üzerinde yoğun çalışmalar yapılan alanlardan biri olmuştur. HLA (High Level Architecture), IEEE Standard 1516 [1] 'ya göre hazırlanmış ve dağıtık benzetimlerin bir arada çalışabilmesi için sunulmuş bir mimaridir. Günümüzde sıkça karşılaşılan bir yanılgı HLA'in kullanıldığı dağıtık benzetimlerde, simülasyonun mimarisinin HLA mimarisi ile şekillenmesi gerektiğidir. Oysa HLA standardının gerçeklemesi olan RTI(Real Time Infrastructure) aslında bir haberleşme katmanıdır ve kendi üzerinde tanımlanan görev uzayının mimari tasarımına etkisi yoktur. Günümüzde HLA tabanlı pek çok benzetimde RTI kodlarının görev uzayı kodlarına girdiği ve mimariyi gereksiz ve yanlış bir şekilde etkilediği görülmektedir. Bu bildiride, TÜBİTAK BTE tarafından geliştirilen ve etki alanı katmanını dağıtık benzetim mimarisinden bağımsız hale getirerek kendi mimarisini geliştirmesine izin veren, ulusal bir benzetim altyapı mimarisi olan FYA (Federe Yönetim Altyapısı) anlatılacaktır.

Bildirinin devamında FYA mimarisinin modülleri olan FYM ve RTILink ve FOMLink'in genel yapısı verilecek, modüller ayrıntılı bir şekilde incelenecek ve gelecekte eklenilmesi beklenen yeni özelliklerden bahsedilecektir.

2. GENEL MİMARİ

FYA; Federe Yönetim Modülü (FYM), RTILink ve FOMLink modüllerinden oluşmaktadır. FYM, bileşen desteği ile bileşen-bileşen, bileşen-nesne ve federe-federe mesajlaşmasına izin veren, oluşturulan RTI nesneleri ile gerçek sınıfları eşleyerek nesne yönetimi sağlayan, zaman yönetimi hizmeti veren ve senaryo yükleme işlemlerinden sorumlu olan modüldür. RTILink modülü ise altyapının HLA arayüzü olarak düşünülebilir. Bu katman RTI üzerinden mesajlaşma yönetimini gerçekleştirir. Bir postane gibi çalışarak federe içine giren mesajları ilgililere iletir ve gönderilecek mesajları RTI'a yönlendirir. Federenin FOM'a (Federate Object Model) dayanarak oluşturacağı nesnelerin kayıt işlerini ve üye olunacak nesnelerin üyelik işlemlerini, otomatik olarak, federenin federasyona girişi sırasında gerçekleştirir. FYM katmanı ile bütünleşik biçimde nesne yönetimi, zaman yönetimi, mesajlaşma gibi temel RTI servislerini sağlar.

FOMLink modülü ise HLA FOM'da tanımlanan her bir nesne için C++ sınıfları üreten modüldür. Etki alanı, diğer yardımcı modüller ile birlikte FYM ve RTILink'in genel mimarisi Şekil 1'de anlatılmaktadır. Etki alanı kullanıcısı FYM'yi kullanarak ihtiyaç duyduğu hizmeti almaktadır. FYM genel olarak sistemin nasıl kullanılacağını bilen katmandır. Kullanıcıdan gelen istekleri diğer modülleri kullanarak karşılar. Dağıtık benzetimin etki alanından soyutlandığı katman burasıdır.



Şekil 1. FYA Genel Mimarisi

RTILink modülü RTI ile konuşan ve sistemin dış dünyaya bakan kesimidir. Bu modül RTI arayüzü görevini üstlenmektedir. RTILink, dışarıdan gelen mesajların ve nesnelerin nasıl yorumlanacağını bilen ve dış dünyaya, yani federeden federasyona, gönderilecek tüm bilgileri RTI'ın anlayacağı şekilde paketleyen modüldür. FOMLink modülü FOM'da tanımlanan nesnelerin gerçek sınıflarını oluşturan ve bu nesnelerin her biri için kodlayıcı ve yorumlayıcı sınıflar sunan yapıdır.

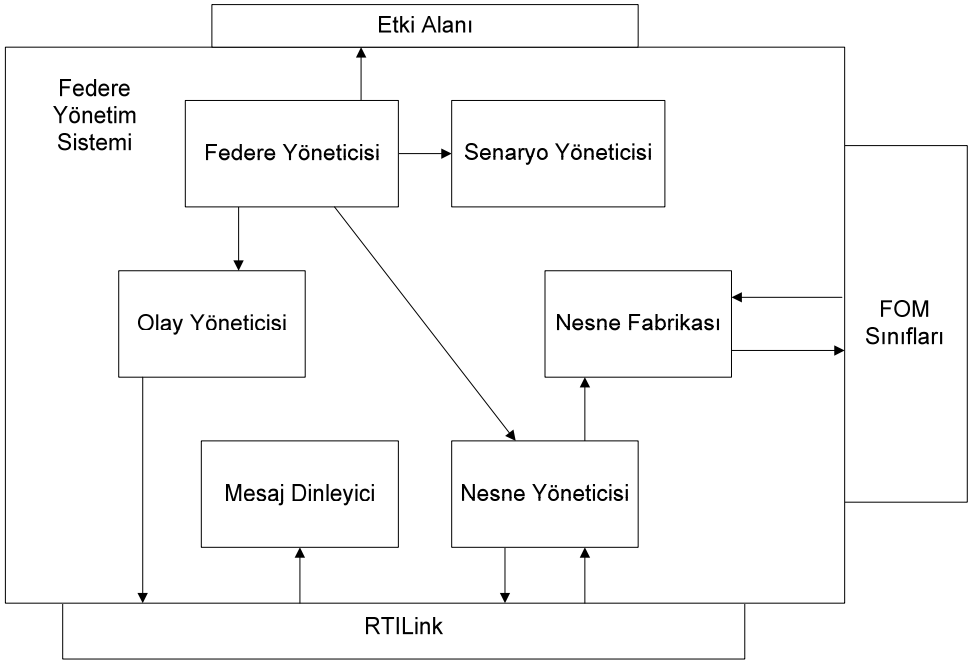
Örneğin nesne güncellemede, kullanıcı FOMLink tarafından üretilmiş C++ sınıfı üzerinden güncellenecek parametreleri girer. FYM modülündeki Federe Yöneticisi sınıfını kullanarak güncellenecek nesneyi belirtir. İlgili nesne için uygun kodlayıcı sınıf bulunur ve C++ nesnesinden RTILink modülündeki nesneye parametreler geçilir. FYM modülündeki gerekli güncellemeden sonra RTILink modülü devreye girerek RTI üzerinden nesnenin güncellenmesi sağlanır.

3. FYM

Federe Yönetim Modülü etki alanı kullanıcısını dağıtık benzetim mimarisinden soyutlayan, kendi mimarisini geliştirmesine imkan tanıyan modüldür.

Genel yazılım mimarisi Şekil 2'de gösterilen Federe Yönetim Modülünün gerçekleştiği temel işlemler aşağıda sıralanmıştır.

- Federe-federasyon işlemleri
- Koşum kontrol
- Nesne yönetimi
- Mesajlaşma
- Zaman çizelgeleme ve zaman yönetimi



Şekil 2. FYM Genel Mimarisi

Federenin federasyona katılması ile yayınlayacağı ve üye olacağı nesnelerin tanımlanması işlemi tamamen otomatikleştirilmiştir. Kullanıcı, yapılandırma dosyasında federesinin federasyonda kullanacağı tekil ismi belirtir ve etki alanı katmanında "FedereYoneticisi" sınıfını başlatarak federesini oluşturur. FYM'nin sahip olduğu işlevsellik bu sınıf üzerinden kullanılabilir. Böylece kullanıcı tek bir sınıf kullanarak dağıtık benzetim ortamında bir federe olarak federasyona dahil olabilir ve diğer federelerle haberleşebilir.

Nesne yönetimi hem FYM hemde RTILink katmanlarında yapılmaktadır. Nesneler, RTILink modülünde ortak bir tipte (Nesne sınıfı) tutulur. FYM'de ise her "Nesne" sınıfı, FOM' da belirtilen ve FOMLink vasıtası ile üretilen gerçek nesne sınıfı ile eşleştirilerek saklanır. Bu eşleştirme işleminden sorumlu olan sınıf "Nesne Yöneticisi" sınıfıdır. FOM'da tanımlı nesnelerin oluşturulma işlemini "Nesne Fabrikası" sınıfı ile beraber gerçekleştirir. Bu sınıf daha önceden FOMLink modülü ile oluşturulmuş FOM sınıflarından uygun olanı seçip başlatan sınıftır.

"Olay Yöneticisi" sınıfı ise mesajlaşma trafiğini yöneten sınıftır. FYM bileşenlerin nesnelere, diğer bileşenlere ve diğer federelere mesaj gönderip almasına olanak tanır. Burada bileşen ve nesneyi benzetim mimarisi açısından incelemek faydalı olacaktır. Nesneler benzetim dünyasında tanımlanmış, özellikleri belirtilmiş birimlerdir. HLA tabanlı benzetimlerde nesneler FOM denen XML tabanlı dosyada tanımlıdırlar. Burada nesnelerin bütün özellikleri belirtilir. Bileşenler ise etki alanı seviyesinde tanımlanırlar. Kullanıcının yapacağı işe ve tasarımına göre her federede bir ya da daha çok bileşen olabilir ve bu bileşenler bir ya da daha çok nesneyi kontrol edebilirler. Kullanıcı, "Bilesen" sınıfından türeterek kendi bileşen sınıflarını oluşturabilir. Kullanıcı istediği

mesaj tipine istediği nesne veya bileşeni eşleyebilir. Bu eşleme sonucunda bir nesnenin attığı mesaj RTI üzerine çıkmadan birbaşka nesneye iletilebilir. FYM, hem iç hemde dış mesajlaşmayı desteklemektedir. RTI'a gidecek mesajlar RTILink modülüne iletilir. RTI'dan gelen mesajlarda yine bu sınıf içinde yorumlanıp federe içinde ilgili nesne veya bileşenlere gönderilebilir. FYM modülü "callback" yapısında mesaj alımını da desteklemektedir. Bu yapı ile gelen mesajlar kullanıcının kaydettirdiği fonksiyonuna geçilir. Olay yönetimi "Mesaj Dinleyici" sınıfı ile asenkron hale getirilmiştir. Tek "thread"'li RTI mekanizmasında; RTI'dan gelen mesajlar bu "thread"'ile gelir ve tüm "callback" fonksiyonları tamamlanıncaya kadar "thread" beklemede kalır. FYM modülünde ise gelen mesajın geçileceği kullanıcı fonksiyonları için "thread" açılır ve RTI "thread"'inin serbest kalması sağlanır. Bu şekilde RTI'ın bekleme zamanı düşürülür ve performans attırılır.

"Senaryo yöneticisi" sınıfı ise benzetim içinde kullanıcı tarafından hazırlanan senaryoyu okuyan, işleyen ve kullanıcının hizmetine sunan sınıftır. Senaryo yükleme ve kaydetme işlemleri bu sınıf tarafından gerçekleştirilir.

Dağıtık benzetimlerde zaman yönetimi oldukça karmaşıktır ve gerçeklenmesi zordur. Federe yönetim altyapısı zaman pullu (time stamped) zaman yönetimini desteklemektedir. Bu yapıda federenin belirli zaman aralıkları ile çalışması esas alınır. Örneğin zaman aralığının 0.01 saniye olduğu durumlarda, FYM bu değerini aşmayacak şekilde bileşenlere işlerini yapmaları için belli bir süre verir ve her seferinde zamanı 0.01 saniye arttırarak ilerletir.

Diğer bir yöntem ise bileşenlere ne kadar zamana ihtiyaç duyduklarını sormaktır[2]. Bu sorgu sonucunda dönen zaman değeri RTI'a bildirilir. RTI bu zaman aralıklarından en küçüğünü seçer ve tüm federelerin bu zaman aralığı kadar zamanlarını ilerletmelerini ister. FYM zaman adımlarının bu değerlere uygun gitmesinden sorumludur. Böylece tüm benzetim herhangi bir mesajı kaçırmadan en küçük ortak adımı atarak koşumuna devam eder.

Zaman çizelgeleyici, ileriye dönük ya da tekrarlanan mesaj gönderimleri için kullanılan bir yapıdır. Federe farklı zaman aralıkları oluşturarak mesaj gönderimi gerçekleştirebilir. Örneğin her 2 saniyede bir atılacak mesaj zaman çizelgeleyici ile zamanlanabilir.

4. RTILink

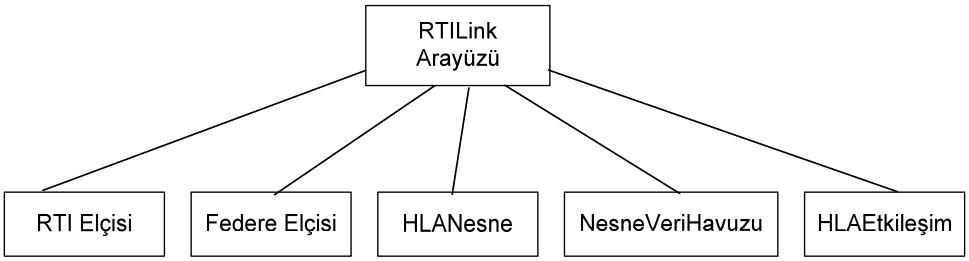
FYM katmanı ile RTI arasında çalıştırılmak üzere tasarlanan RTILink modülü, FYM' nin benzetim altyapısında kullanılan mimariden bağımsız olarak çalışabilmesini sağlamaktadır.

Mevcut yapısı ile HLA benzetim alt yapısını kullanan benzetimlerde RTI-federe etkileşimini sağlayan RTILink modülünün başlıca görevleri aşağıda sıralanmıştır.

- HLA üzerinden mesajlaşma yönetimini sağlamak
- HLA üzerinden benzetim nesnesi yönetimini, nesne özelliklerinin güncellenmesini ve yaratılan nesnelerin algılanmasını sağlamak
- Nesne ve mesaj iletimi sırasında güncellenen özelliklerin kodlanması ve kodlanmış olarak alınan özelliklerin çözülmesini işlemini gerçekleştirmek

- FYM katmanı ile etkileşimli olarak zaman yönetimini gerçekleştirmek

Genel yazılım mimarisi Şekil- 3 de gösterilen RTILink, FYM ile etkileşim arayüzü olarak en üst seviyede tek bir sınıf içermektedir. Bu sınıf yardımı ile her türlü federasyon işlemi gerçekleştirilebilmektedir. "RTIElçisi" ve "FederateElçisi" sınıfları, sırasıyla "RTIAmbassador" ve "NullFederateAmbassador" sınıflarından türetilmiş ve RTILink isterleri için ihtiyacı karşılayacak şekilde düzenlenmiştir. Bu sınıflar aracılığı ile RTI etkileşimi sağlanmaktadır. "HLANesne" sınıfı yayınlanan yada üye olunan ve FOM'da tanımlanmış nesne tiplerinin bilgilerinin tutulduğu sınıftır. "NesneVeriHavuzu" sınıfı ise bu nesnelerin sahip oldukları parametrelerinin değerlerini tutan sınıftır. Bu sınıfların oluşturulması ve RTI ile etkileşimi RTILink katmanında otomatikleştirilmiştir. "HLAEtkileşim" sınıfı ise FOM'da tanımlanan "interaction" sınıflarını tanımlamak için kullanılmıştır.Bu sınıf kullanılarak federasyonda yer alan federeler arası mesajlaşma gerçekleştirilir.

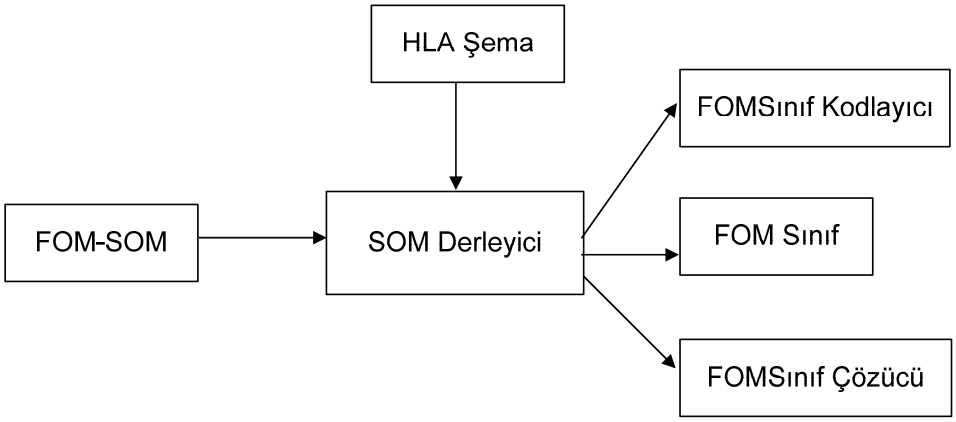


Şekil 3. RTI-Link Genel Mimarisi

5. FOMLİNK

RTI, temel olarak HLA tabanlı federeler arasındaki objelerin yaratılıp, silinmesi ve objelerin özelliklerinin güncellenmesi işlemlerini gerçekleştiren, federeler arasında mesajlaşmayı sağlayan bir iletişim mekanizmasıdır. Benzetim üzerindeki diğer federelerdeki benzetim nesnelerinde meydana gelen değişiklikler, geliştirilen ara katman API (Application Programming Interface)'leri tarafından kontrol edilmelidir, RTI bu kontrol mekanizmasını gerçekleştirmez[3].

Geliştirilen FOMLink sayesinde; benzetim nesnelerinin C++ sınıflarına eşlemesi sağlanır. FOM dosyasında tanımlı nesneler ve tipleri için erişim atama metotları tanımlı sınıflar oluşturulur.



Şekil 4. FOM-Link Genel Yapısı

FOMLink'in temel yapısı Şekil 4'de anlatılmıştır. Buna göre ilk olarak, FOM'un HLA şema ile uymluluğu kontrol edilir. Her bir FOM sınıfı için eşlenik C++ sınıfı üretilip, sınıf öznitelikleri erişim ve atama fonksiyonları yaratılır. Ayrıca, özniteliklerin RTI ya gönderilmeden önce kodlanması ve alınan özniteliklerin çözülmesi işini gerçekleyen "FOMSınıf Kodlayıcı" ve "FOMSınıf Çözücü" sınıfları yaratılır.

FOM değişikliği yapıldığında yukardaki işlemler terkrarlanarak API'lerin tekrar üretilmesi ve bu API'leri kullanan diğer üst katman uygulamalarının da tekrar derlenip yapılan değişikliklerden haberdar olması sağlanır. Burada dikkat edilmesi gereken nokta ise, FOM'daki değişiklik sözdizilimsel veya anlamsal bir değişlik olabilir. Dolayısıyla üst katman yazılımlarının, anlamsal değişiklikleri anlayabilecek yetenekte geliştirilmesi gerekir.

6. FYA PERFORMANS ANALİZİ

Asenkron mesaj iletimi FYA'nın en önemli özelliklerinden biridir. Bu yapıda HLA ile etkileşim tek "thread" üzerinden çalışan RTILink katmanı aracılığı ile gerçekleştirilmektedir. Ancak bunun haricinde uygulama katmanına hizmet veren yapı çoklu "thread" yapısında işlem görmektedir. Uygulama işlemleri için bir "thread" havuzu oluşturan FYM, RTILink katmanından gelen her etkileşim isteğini (callback metod çağrımı) boş durumdaki bir "thread" e yönlendirerek federe elçisinin kullandığı zaman dilimini en aza indirmektedir. Böylelikle tüm sistemin tepki süresi hızlanmakta ve genel çalışma süresi azalmaktadır..

Genel olarak FYM' nin çoklu "thread" işlemlerinde mevcut diğer ürünlerden iyi performans sergilemesi, FYM'nin HLA etkileşimi için RTILink katmanını kullanmasından kaynaklanmaktadır. Bu ara katman sayesinde HLA erişimi, uygulama katmanından bağlanılan "thread" sayısından bağımsız olarak, RTILink katmanında tek "thread" üzerinden gerçekleştirilmektedir. RTI üzerinden tek kanaldan aktarılan veri, FYM katmanında çoklu kanaldan işlenirken, çoklu kanaldan gelen veri ise sıralanıp RTI üzerinden federasyona tek kanaldan iletilmektedir.

Piyasada, mevcut RPR_FOM sürümlerini destekleyen çeşitli ürünler bulunmaktadır. Ancak RPR_FOM'dan türetilen FOM ların ya da tamamı ile yeni oluşturulan FOM yapılarının kullanılması durumunda, FOM içerisinde bulunan her nesne ve etkileşim için ayrı sınıfların kodlanmasına ihtiyaç duyulmaktadır. FYA da ise bu işlem, bünyesinde bulundurduğu FOMLink ile federe kodunun derleme zamanından önce, FOM dosyasını kullanarak, tüm HLA etkileşim ve HLA nesne sınıflarının otomatik olarak yaratılması ile sağlanmaktadır. Basit ve sabit boyutlu veri yapılarının kodlama ve kod çözücü metodlarını otomatik olarak üreten FOMLink, uygulama yazılımcısına sadece dinamik boyutlu veri yapılarının kodlama ve kod çözücü metodlarının kodlanmasını bırakmaktadır.

Diğer ürünler, benzetime başlamak ve federasyona katılmak için tek bir sınıf kullanırlar, ayrıca nesne ve etkileşim sınıflarının yayımlanması ve üye olunması işlemleri için, ilgili sınıflardan birer nesne türetilmesini ve ilgili etkileşim sınıfına "callback" metodu kayıt ettirilmesini şart koşmaktadırlar. Bunun yanında HLA' e özgü bir yayımlama ya da abonelik işlemi gerektiğinde ise kullanıcının RTI metodlarına müdahale etmesi gerekmektedir. Bu yapı karmaşık ve hata yapılması kolay bir işlem gerektirir.

FYA yapısında ise uygulama yazılımcısının üye olacağı ve yayımlayacağı sınıfları çalışma zamanından önce yapılandırma dosyasında belirtmiş olması yeterlidir. FYM, tek sınıf üzerinden benzetimi başlatıp, federasyona katılırken otomatik olarak yayımlama ve üye olma işlemlerini gerçekleştirecektir. Böylelikle kullanıcı kaynaklı hatalar en aza indirgenirken, HLA tabanlı benzetimlerin zor adımlarından biri olan federasyona, federeyi tanıtma işlemi otomatik ve kolay bir şekilde gerçekleştirilmektedir.

7. SONUÇ

Bu bildiride HLA tabanlı dağıtık benzetim geliştirme ve bütünleştirme sürecinde kullanılabilecek bir altyapı olan FYA tanıtılmıştır. TÜBİTAK MAM BTE bünyesinde geliştirilen bu uygulama etki alanı kullanıcılarını, kullanılacak dağıtık benzetim mimarisinden soyutlayarak, kendi mimarisine yoğunlaşmasına izin verir. Ayrıca asenkron mesajlaşma desteği vererek etkin bir federe tasarlamaya imkân verir. Kullanıcının dağıtık benzetimin doğasından kaynaklanan programlama yükünü en aza indirir. Ulusal bir uygulama olması sayesinde, yeni ihtiyaçlara göre istenilen değişikler hızlı bir şekilde yapılabilecektir.

KAYNAKÇA

[1] IEEE 1516, Standard for Modeling & Simulation—High Level Architecture, 2000.

[2] Fujimoto Richard, 2000. "Paralel and Distributed Simulation Systems." John Wiley & Sons, Inc.

[3] L. Granowetter, MÄK Technologies, Inc.(1998)"Solving the FOM-Independence Problem".

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

FEDERASYONLAR İÇERİSİNDE DİNAMİK FEDERE KÜMELENMELERİ VE ZAMAN YÖNETİMİ

Aydın OKUTANOĞLU [(a)], Müslim BOZYİĞİT [(b)]

[(a)] Meteksan Sistem A.Ş., ODTÜ Teknokent Silikon Blok Kat:1 No:7 Ankara, aokutanoglu@meteksan.com.tr

[(b)] ODTÜ Bilgisayar Müh., ODTÜ Bilgisayar Mühendisliği Bölümü Ankara, bozyigit@ceng.metu.edu.tr

ÖZ

Dağıtık simülasyonların geliştirilmesinde kullanılan federasyonlar içerisinde birden fazla federe bulunmaktadır. Bu federelerin bazıları federasyonun çalışması sırasında birbirleri ile diğerlerine oranla daha fazla ilişkili olabilirler ve bu yakın ilişkilerinin sonucu olarak birbirleri ile çok daha fazla etkileşim içerisinde olacaklardır. Bu çalışmada bu birbirleri ile yakın ilişki içerisinde olan federelere ait federe kümelerin dinamik olarak belirlenmesi ve bunlar arasında etkili zaman yönetiminin geliştirilmesi ele alınacaktır.

Federe kümelerinin birbirleri ile ilişkili olma durumu, Veri Dağıtım Yönetimi (Data Distribution Management – DDM) servisleri kullanılarak tanımlanmış olan dağıtım uzaylarının (routing spaces – RS) boyutlarına bağlı bir fonksiyon olarak alınmaktadır. Bir dağıtım uzayı içerisinde federeler arasındaki mesafe bu federelerin yakınlık/uzaklık durumlarını belirlemektedir. Buna göre, federe kümeleri göreceli olarak birbirlerine yakın olan federelerden oluşmaktadır.

Dağıtık simülasyonun çalışması sırasında federelerin birbirlerine yakınlıkları zamanla değişebilecektir. Bu nedenle, federe kümeleri zamanla küçülebilmekte veya büyüyebilmektedir. Federe kümelerinin birbirlerinde oldukça uzak olduğu durumlarda bu federeleri zaman yönetimi açısından birbirleri ile senkronize etmeye gerek yoktur. Bu federeler bu zaman dilimi içerisinde birbirleri ile ilişkili değillerdir ve birbirleri ile herhangi bir etkileşime girmemeleri önemlidir. Ancak, bu kümeler tekrar birbirlerine yaklaşmaya başladıklarında mantıksal zamanlarının tekrar senkronize edilmesi gerekmektedir. Bu çalışma tam olarak bunu yapmaktadır.

Anahtar Kelimeler: Dağıtık Simülasyon, HLA, Federe Kümeleri, Federe Senkronizasyonu, Veri Dağıtımı Yönetimi, Zaman Yönetimi.

DYNAMIC FEDERATE CLUSTERS AND TIME MANAGEMENT IN FEDERATIONS

ABSTRACT

There are more than one federates in the federations that are used to develop distributed simulation applications. Some of these federates will be more related compared to others during simulation execution. There will be much more frequent interactions between these related federates. In this work, the formation of the clusters of the related federates and the effective time management between them are investigated.

The relevance of federates is taken to be a function of a degree of proximity along dimensions of routing spaces which are defined using Data Distribution Management services. The distances along a a dimension of interest between federates in a routing space is taken to determine the proximity of federates. Thus, a federate cluster is composed of relatively close federates.

Proximity of federates could change during the simulation execution. For this reason, federate clusters dynamically expand or shrink in runtime. When federate clusters are distant to each other, there is no need to synchronize them because they are not related with each other. During this period, logical times of clusters are not synchronized. If these federate clusters get close to each other, however, we will need to resynchronize their logical times, which is the heart of this study.

Keywords: Clustering, Data Distribution Management, Distributed Simulation, HLA, Federate Synchronization, Time Management.

1. GİRİŞ

Dağıtık simülasyonları geliştirilmesinde kullanılan en güncel yöntem olan HLA (High Level Architecture) içerisinde mantıksal nesnelerin yönetiminden sorumlu olan federe grubu federasyon olarak adlandırılmaktadır [1, 2]. Federeler federasyon içerisinde tanımladıkları nesnelerin özelliklerinin değiştirilmesi ve tanımlanmış etkileşimlerin kullanılması sayesinde birbirleri ile etkileşim içerisinde bulunmaktadırlar.

Simülasyon çalışma zamanı sırasında bazı federe grupları birbirleri ile diğerlerine oranla daha fazla ilişkili durumu gelebilmektedirler. Bu çalışma içerisinde birbirleri ile yakın ilişki içerisinde bulunan federelerin oluşturdukları mantıksal gruplara federe kümeleri [8] denmektedir. Federe kümelenmesine örnek olarak içerisinde birden fazla alt çatışma senaryosunun olduğu bir dağıtık savaş simülasyonu gösterilebilir. Eğer alt çatışmalar birbirlerinden yeterince uzakta iseler, aynı çatışma içerisindeki tank, uçak gibi federeler birbirleri ile çok yakın olarak ilişkilidirler, ancak diğer çatışmalardaki federeler ile yakın bir bağları yoktur.

Bu çalışmanın ana amaçları, federasyon içerisindeki mantıksal kümelenmeleri kullanarak zaman senkronizasyonu için gereken mesajların sayısını azaltmak, dolayısı ile senkronizasyon performansını yükseltmek, ve zaman yönetimi içerisindeki gereksiz bloklanmaları azaltarak federeler arasındaki paralelliği arttırmaktır.

Mantıksal olarak birbirinden bağımsız olan ve aralarında herhangi bir etkileşimin olmadığı federe kümelerinin kendi aralarında zaman senkronizasyonunun olması bir zorunluluk değildir. Yukarıda belirtilen amaçlara ulaşmak için federe kümeleri arasındaki bu muhtemel mantıksal zaman uyuşmazlığı bir avantaj olarak kullanılacaktır. Federelerin birbirleri ile ilişkili olmadığı zaman aralıklarını bulabilmek için DDM [3, 4, 5] servisleri kullanılmaktadır.

Federasyon kümeleri arasındaki mantıksal bölünmeler simülasyonun çalışması sırasında zamanla değişebilmektedirler. Dağıtık savaş simülasyonu örneğine tekrar dönersek, birbirinden uzak farklı noktalarda geçen alt çatışmalar düşünülmektedir. Bu alt çatışmalardan birinde bulunan bir uçak grubu farklı bir çatışmaya doğru ilerlediğinde daha önceki federe kümesiyle arasındaki mantıksal bağlantı kopmakta yeni ulaştığı yerdeki küme ile yakın ilişki kurulmaktadır. Bu durumda bu iki kümenin formasyonunun değişmesi gerekmektedir.

İkinci bölümde dinamik kümelenme algoritmasının detayları bulunmaktadır. Üçüncü bölümde federe kümelenmeleri için iyileştirilmiş zaman yönetimi mekanizması açıklanmıştır. Sonuç ve planlanan çalışmalar dördüncü bölümde verilmiştir.

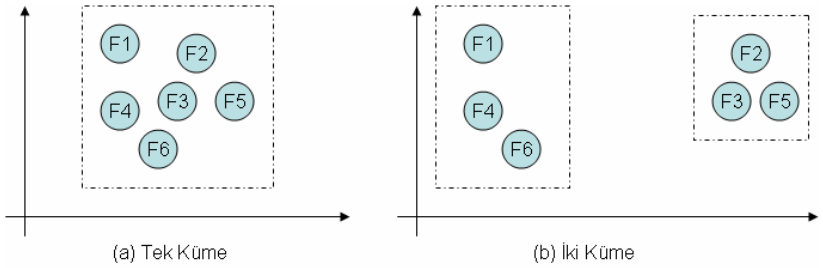
2. DİNAMİK FEDERE KÜMELENMELERİ

Trafik simülasyonu gibi fiziksel ortam simülasyonları içerisinde bilindiği üzere araba, yaya gibi simüle edilen elamanların aralarındaki mesafe belirli bir uzaklığı geçtiğinde birbirleri arasında herhangi bir etkileşim olmayacaktır. Öte yandan birbirleri ile yakın olan federeler arasında yoğun etkileşimler görülecektir.

Önerilen dinamik kümelenme algoritmasında DDM içerisindeki üyelik alanları (subscribe region) ve güncelleme alanlarına (update region) bağlı olarak hesaplanan mesafeler kullanılarak federasyon içerisindeki federeler birbirlerine olan uzaklıklarına göre alt gruplara ayrılmaktadırlar. Federe kümelerinin birbirlerine olan uzaklıkların hesaplanabilmesi için her bir federenin simülasyonun sanal uzayı içerisindeki yerinin hesaplanması gerekmektedir. Bu hesaplama için federelerin DDM servislerini kullanarak tanımladıkları üyelik ve güncelleme alanları kullanılmaktadır. Federelerin tanımladıkları tüm alanların bileşkesi o federenin dağıtık uzayındaki (routing space) yerini belirtmektedir.

2.1 Yeni Federe Kümesi Oluşumu

Bu işlem içerisinde, var olan federe kümelerinin iki veya daha fazla farklı kümeye bölünüp bölünmemesi gerektiğine karar verilmektedir. Şekil 1'de ek bir federe kümesinin simülasyon çalışması sırasında iki ayrı kümeye bölünmesi gösterilmiştir.



Şekil 1. Küme bölünmesi

Federe kümeleri içerisindeki olası bölünmelerin belirlenmesi için Sweep-and-Prune [6, 7] algoritmasındaki sıralı listeler kullanılmaktadır. Federasyon içerisinde tanımlanmış olan her bir dağıtım uzayının tüm eksenleri için bir sıralı liste tutulmaktadır. Küme içerisinde bir bölünme olabilmesi için bu sıralı listelerden en az birinde bir boşluk oluşması gerekmektedir.

Tablo 1. İki eksen için sıralı listeler

X Eksenindeki sıralı liste						
Federe	F4	F1	F6	F3	F2	F5
Son nokta değeri	10	10	12	25	27	28

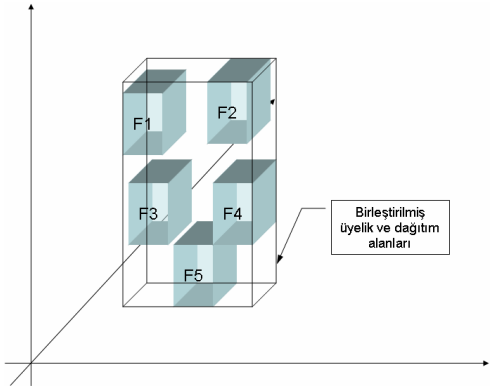
Y Eksenindeki sıralı liste						
Federe	F6	F4	F3	F5	F2	F1
Son nokta değeri	12	13	14	14	16	17

Tablo 1'de Şekil 1 (b)'de verilmiş olan federeler için hesaplanmış sıralı listeler verilmiştir. Asıl algoritmada herbir federe için hesaplanmış bileşke alanın o eksendeki başlangıç ve bitiş değerleri tutulmaktadır, ancak tablonun gösterimde basitleşmesi amacıyla herbir federe için bir değer gösterilmiştir. Tabloda görülebileceği gibi, Y ekseninde herhangi bir boşluk oluşmamıştır, ancak X ekseni için oluşturulan sıralı listede F6 ile F3 federeleri arasında bir boşluk bulunmaktadır. Bu boşluk değerinin daha önceden belirlenmiş olan eşik değerini aşması durumunda örnekte olduğu gibi federe kümesi iki ayrı kümeye bölünmektedir.

2.2 İki federe kümesinin birleştirilmesi

Algoritmanın bu adımında iki farklı federe kümesinin birbiri ile birleştirilmesinin gerekli olup olmadığı incelenmektedir. Böyle bir durumla iki federe kümesi içerisindeki federelerin birbirlerine yaklaşması durumunda karşılaşılmaktadır. Federe kümelerinin birbirlerine yaklaşmalarının hesaplanabilmesi için federe kümelerin arasındaki mesafenin düzenli olarak hesaplanması gerekmektedir.

Federe kümesinin yerinin dağıtım uzayında belirlenebilmesi için küme içerisinde yer alan federelere ait alanların bileşkesinin alınması gerekmektedir. Şekil 2'de 3 boyutlu bir dağıtım uzayında tanımlanmış olan federe kümesi için hesaplanan bileşke alan gözükmektedir.



Şekil 2. 3 boyutlu uzayda alan birleştirme

Aşağıdaki formülde bir federe kümesi (FC_i) için bileşke alanın ($U_{FC_i}^{RS_k}$) ne şekilde hesaplanacağı gösterilmektedir. Bu formül içerisindeki $SR^{RS_k}(F_j)$, F_j federesine ait RS_k dağıtım uzayındaki üyelik alanını göstermektedir. Aynı şekilde $UR^{RS_k}(F_j)$ bu uzaydaki güncelleme alanını göstermektedir.

$$U_{FC_i}^{RS_k} = \bigcup \left\{ SR^{RS_k}(F_j) \cup UR^{RS_k}(F_j) \right\} \quad \forall F_j \in FC_i \tag{1}$$

RS_i dağıtım alanında FC_a ve FC_b arasındaki mesafe ise $d^{RS_i}(FC_a, FC_b)$ ifadesi ile gösterilmektedir. Bu mesafenin hesaplanması aşamasında sıralı listeler algoritmasındakine benzer şekilde her bir eksendeki mesafelerin en büyüğü alınmaktadır. Bu şekilde hesaplanan mesafe tam gerçek mesafeden küçük veya eşit olacaktır ve bu algoritma içerisinde gerçek mesafeden ziyade karşılaştırma için kullanılacak bir yaklaşık değere ihtiyaç duyulmaktadır.

İki federe kümesi arasındaki mesafe daha önceden belirlenmiş eşik değerinden küçük hale gelmesi durumunda bu iki küme birleştirilmektedir. Bu koşul aşağıdaki formülde verilmektedir.

$$d^{RS_i}\left(FC_a, FC_b\right) \leq \omega^{RS_i} \quad \forall RS_i \tag{2}$$

3. FEDERE KÜMELERİ ARASINDA ZAMAN YÖNETİMİ

Daha öncede belirtildiği gibi her bir federe kümesi kendi içinde birbirleri ile yakından ilişkili federelerden oluşmaktadır. Ancak, iki ayrı kümede bulunan federeler arasında ise tanım gereği etkileşim bulunmamaktadır. Küme ayrışma eşik değerinin yeterince büyük seçilmesi durumda belirli bir süre boyunca da aralarında herhangi bir etkileşim bulunmayacaktır. Bu durum göz önüne alınarak farklı federe kümeleri arasında mantıksal zaman bakımından senkronizasyon gereksiz duruma gelmektedir.

Önerilen zaman yönetimi mekanizmasında birbirlerinden uzakta olan federe kümeleri arasında mantıksal zaman bakımından bağımlılık kırılmaktadır. Bunun sonucu olarak farklı kümelerdeki federeler birbirlerinden bağımsız olarak zamanlarını ilerletebilmekte ve herbir federe kümesi birbirinden bağımsız farklı zamanlarda ilerleme durumuna gelmektedir. Burada kritik olan konu, bu kümelerin birbirlerinden uzakta kaldıkları süre sonunda tekrar birbirlerine yakınlaşabilecekleri ve tekrar etkileşmeye başlayabilecekleridir. Bu durumda bu federe kümelerinin zamanları tekrar birbirleri ile senkronize hale getirilmelidir.

Federe kümelerinin zamanlarının birbirinden bağımsız olduğu periyoda lokal senkronizasyon periyodu, kümelerin zamanlarını birbirlerine uyumlu hale getirilmeye çalışıldığı periyoda tekrar senkronizasyon periyodu ve kümelerin birbirleri ile tam senkronize olarak çalıştığı periyoda normal senkronizasyon periyodu denmektedir.

3.1. Lokal senkronizasyon periyodu

İki federe kümesi birbirinden yeterince uzak oldukları durumda aralarındaki zaman bağımlılığı kırılmaktadır ve bu zaman dilimine lokal senkronizasyon periyodu denmektedir. Federelerin DDM servislerini tam olarak kullandıkları varsayıldığından dolayı birbirleri arasında herhangi bir etkileşim olmayacağı garanti edilmektedir ve bu nedenle birbirleriyle uyumsuz mantıksal zamanlara sahip olmaları herhangi bir mantıksal soruna yol açmayacaktır.

3.2 Tekrar senkronizasyon periyodu

Tekrar senkronizasyon periyodu içerisinde hızlı olan (mantıksal zamanı diğerine göre daha büyük olan) federe kümesi birbirleriyle etkileşime geçmeden belirli bir süre önce bloklanmaktadır. Bu sayede yavaş olan federelere hızlı olanları yakalayabilmesi için fırsat tanınmaktadır. Belirli bir süre sonunda yavaş olan federe kümesi hızlı olanı yakalayacak ve normal senkronizasyon başlayacaktır.

Tekrar senkronizasyon periyodu yavaş olan federenin hızlı olanı yakalayabilmesine yetecek kadar uzun olması gerekmektedir. Bu sürenin yeterli olması için tekrar senkronizasyona uygun anda başlanması gerekmektedir. Bu başlama zamanının belirlenmesi için dağıtım uzayı içerisinde federe kümelerinin birbirlerine yaklaşma hızları (AS – approaching speed) hesaplanmaktadır.

Simülasyonun çalışması sırasında tüm federe kümesi çiftleri için yaklaşma hızları düzenli olarak hesaplanmaktadır. RS_i dağıtım alanında FC_a ve FC_b federe kümelerinin yaklaşma hızları aşağıdaki formül ile hesaplanmaktadır.

$$AS^{RS_i}(FC_a, FC_b) = -\frac{\nabla d^{RS_i}(FC_a, FC_b)}{\nabla LT_{FC_a} + \nabla LT_{FC_b}} \qquad (3)$$

Bu formülde ∇LT_{FC_a} FC_a federe kümesinin mantıksal zaman farkı, ∇LT_{FC_b} FC_b federe kümesinin mantıksal zaman farkı ve $\nabla d^{RS_i}(FC_a, FC_b)$ ise bu federe kümeleri arasındaki mesafe değişim miktarıdır. Bu formül ile hesaplanan yaklaşma hızı bu iki federe kümesinin bir birim mantıksal zaman içerisinde hangi miktarda birbirlerine yaklaştıklarını vermektedir. Bölme sonucunun negatifinin alınması sonucunda yaklaşma hızının kümeler birbirlerine yaklaşırken pozitif, diğer durumda ise negatif olması sağlanmaktadır.

Tekrar senkronizasyonun ne zaman başlanacağına karar verilebilmesi için gereken asıl parametre ise kritik mesafedir (CD – Critical Distance). İki federe kümesi arasındaki mesafe kritik mesafeden küçük olduğu anda tekrar senkronizasyon başlatılmaktadır. İki federe için kritik mesafe aşağıdaki formül ile hesaplanmaktadır.

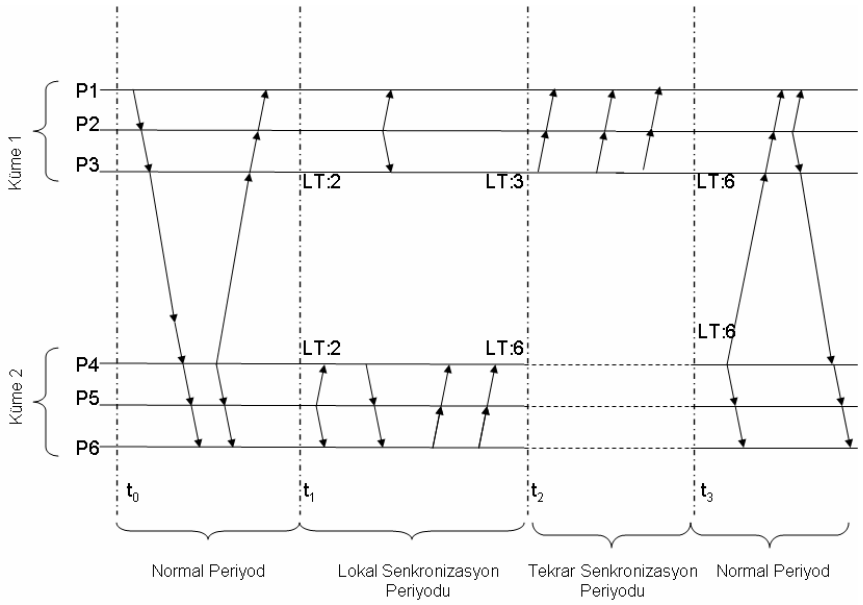
$$CD^{RS_i}(FC_a, FC_b) = \tau + AS^{RS_i}(FC_a, FC_b).\left(Abs(LT_{FC_a} - LT_{FC_b})\right) \qquad (2)$$

CD değeri federe kümelerinin bölünmeleri için kullanılan eşik değerinden büyük olmalıdır. Formüldeki ikinci bileşen ise yavaş olan federe kümesinin aradaki mantıksal zaman farkının kapanabilmesi sırasında katedeceği mesafedir.

İki federe kümesi arasıdaki herhangi bir dağıtım uzayındaki mesafe kritik mesafeden daha küçük olmaya başladığı durumda bu iki küme tekrar senkronizasyon evresine girmektedir. Bu şart aşağıda formüle edilmiştir;

$$d^{RS_i}(FC_a, FC_b) \le CD^{RS_i}(FC_a, FC_b) \qquad \exists RS_i \qquad (3)$$

Önerilen sistemin daha iyi anlaşılabilmesi için aşağıdaki örnek çalışma durumunun incelenmesi faydalı olacaktır.



Şekil 3. Zaman senkronizasyon periyodları

Şekil 3'de altı federeden oluşan bir federasyon içerisinde simülasyonun çalışması sırasında oluşmuş bir kümelenme yapısı gösterilmektedir. Burada Küme1 ve Küme2 olarak adlandırılan iki farklı federe kümesi bulunmaktadır. Burada t_0 ve t_1 aralığında federeler birbirleri ile etkileşim içerisindedirler ve mantıksal zamanları birbirleri ile senkron bir şekilde ilerlemektedir. t_1 zamanında iki federe grubu birbirinden uzaklaşmış durumdadırlar ve 2 ayrı federe kümesi oluşturulmuştur. Bu zaman dilimi içerisinde federe kümeleri arasında herhangi bir etkileşim gerçekleşmemiştir. Bu federe kümelerinin mantıksal zamanları birbirinden farklı hızlarda ilerlemektedir ve t_2 zamanına ulaşıldığında Küme1'in mantıksal zamanı 3, Küme2'nin mantıksal zamanı 6 olarak hesaplanmıştır. t_2 zamanında bu federe kümeleri arasındaki mesaje kritik mesafeye ulaşmış ve tekrar senkronizasyon periyodu başlamıştır. t_2 ile t_3 zamanı arasında devam eden tekrar senkronizasyon periyodu sırasında hızlı olan Küme2 durulmuş ve Küme1'e onu yakalama fırsatı sunulmuştur. t_3 zamanında Küme1'in mantıksal zamanı Küme2'nin mantıksal zamanına ulaşmış ve tekrar senkronizasyon periyodu sona ermiştir.

Önerilen zaman yönetimi mekanizmasının uygun olarak çalışabilmesi için aşağıda belirtilen bazı varsayımlar yapılmıştır.

- İlk varsayım federe hızından bağımsız zaman yönetiminin kullanılmasıdır. Simülasyon içerisindeki federelerin zaman-adımlı (time-stepped) yöntemini kullanması bu yöntem sınıfına girer. Her bir federenın kendi zamanını sürekli olarak belirli aralıklarla arttırması anlamına gelen bu yöntem sayesinde federelerin yerleri belirli bir süreklilik içerisinde hesaplanabilecektir. Olay tabanlı (Event-based) zaman ilerletmesi yöneteminin kullanılması da bir başka

yöntemdir. Bu durumda federelerin bir yerden başka bir yere zıplaması mümkün olabilmekte ve bahsedilen kritik mesafe hesaplamasında ani değişiklikler olmaktadır. Bu belirtilen kısıtlamaya genel olarak uyan uygulama grubu olarak fiziksel simülasyonlar verilebilir.

- Dağıtık savaş simülasyonu veya trafik simülasyonu gibi fiziksel simülasyonlar içerisinde zamanın ilerletilmesi için genellikle zaman-adımlı mekanizma kullanılmaktadır.
- Sistemin çalışabilmesi için yapılan ikinci varsayım ise federelerin çalışması sırasında belirli bir hızı koruyarak ilerlemelerinin sağlanmasıdır. Bu varsayımın yapılmasındaki amaç kritik mesafe hesaplamasında kullanılan yaklaşma hızlarının dramatik miktarlarda değişmesini engelleyerek federelerin birbirleri ile iletişime geçebilecekleri bir mesafeye yaklaştıklarında zamanlarının birbirleriyle uyumsuz olmasını önlemektir. Bu duruma yönelik olarak kritik mesafe içerisine bir miktar ekstra değer eklenerek yaklaşma hızlarındaki bir miktar değişikliğin tolere edilmesi sağlanmıştır. Ancak yüksek miktarlardaki hız değişiklerine çözüm için farklı yöntemlerin kullanılması gerekmektedir. Bu kısıtlamanın gevşetilebilmesi için geliştirilebilecek çözümlerden bir tanes federelere kendilerinin ileride çıkabilecekleri hızları sisteme önceden bildirebilecekleri bir imkan sağlanmasıdır. Önerilebilecek bir başka yöntem ise federe kümeleri içerisinde korunumlu (conservative) zaman yönetimi kullanılırken, federe kümeleri arasında optimistik zaman yönetiminin kullanılması olacaktır. Bu sayede federelerin hızlarını aşırı ölçüde değişmesi sonucunda federelerin birbirlerine yaklaştıklarında zaman uyumsuzluklarının ortaya çıkması durumunda bu iki federe kümesi geri alınarak (rollback) son hesaplanan yaklaşma hızları bilinerek bu periyod tekrarlanabilinecektir.

4. SONUÇ VE GELECEKTE YAPILACAK ÇALIŞMALAR

Bu çalışma içerisinde yakınlık ilişkisi üzerine kurulmuş bir federe kümelendirme algoritması sunulmuştur. Aynı zamanda bu federe kümelenmeleri üzerinde kullanılabilecek yeni bir zaman yönetimi mekanizması sunulmuştur. Bu zaman yönetimi mekanizması federelerin birbirlerinden yeterince uzak olmaları durumunda birbirleri ile etkileşime geçmeyecekleri fikrine dayanmaktadır. Bu sayede bu uzak federelerin birbirleriyle mantıksal zaman bakımından senkronize olmalarına gerek kalmayacaktır.

Bu yöntemin iki ana faydası bulunmaktadır. Bunlardan ilki federelerin birbirlerinden uzak olmaları durumunda dağıtık zaman yönetimi içerisindeki bağımlılık kümeleri küçüldüğünden dolayı zaman yönetiminde kullanılan mesajların miktarında önemli bir düşüş sağlanmasıdır. İkinci avantaj ise birbirinden uzak federelerin birbirlerine gereksiz olarak bağımlılıklarını önlediği için federeler arasında paralelliği arttırmasıdır.

Çalışmanın ana dezavantajı ise kritik mesafe hesaplamasında geçmiş verilerin kullanılmasıdır. Düzensiz federeler içerisinde yaklaşma hızının hesaplanmış değerlerden farklı olma ihtimali her zaman bulunmaktadır. Bu tip düzensiz federeler için kritik mesafe hesaplamasına girdi sağlayabileceği bir geri besleme mekanizması sağlanarak bu problem aşılabileceği düşünülmektedir.

Bu çalışmada yapılması planlanan ilk geliştirme, algoritmanın aynı anda ikiden fazla federe kümesi ile çalışmasının sağlanmasıdır. Bu durumda algoritmadaki muhtemel kilitlenmeler (deadlock) ve performans konularının daha da önem kazanacağı aşıkardır.

Sistemin çalışması için gereken değişikliklerin bir ara katman içerisine yerleştirilerek var olan RTI sistemlerinde herhangi bir değişiklik yapılmadan gerçekleştirilmesi çok önemlidir. Bu ara katman içerisinde değiştirilmiş olan RTIAmbassador ve FederateAmbassador sınıflarının federeler tarafından kullanılması sayesinde gerekli bilgiler elde edilebilmektedir.

KAYNAKÇA

[1] Chandy, K. M. and J. Misra. “Distributed Simulation: A case study in design and verification of distributed programs”. IEEE Transactions on Software Engineering. SE-5(5), 440-452, 1979.

[2] Cheung, S. and M. Loper. “Synchronizing simulations in distributed interactive simulations”. In 1994 Winter Simulation Conference Proceedings, pp. 1316-1323. 1994.

[3] I. Tacic and R. Fujimoto. "Synchronized Data Distribution Management in Distributed Simulations." Proceedings of the 12th Workshop on Parallel and Distributed Simulation (PADS '98), pp. 108-115. May 1998.

[4] D. Van Hook, S. Rak, J. Calvin. “Approaches To RTI Implementation of HLA Data Distribution Management Services”. 15th Distributed Interactive Simulation Workshop, paper 96-14-084, Fall 1996.

[5] Mark Hyett, Roger Wuerfel. “Implementation of the Data Distribution Management Services in the RTI-NG”. Fall Simulation Interoperability Workshop 2002, Paper no. 02S-SIW-044.

[6] J. D. Cohen, M. C. Lin, D. Manocha, and M. K. Ponamgi. “I-COLLIDE: An Interactive and Exact Collusion Detection System for Large-Scale Environments”. In Symposium on Interactive 3D Graphics, pages: 189-196, 218, 1995.

[7] Elvis S. Liu, Milo K. Yip, Gino Yu. “Scalable interest management for multidimensional routing space”. Proceedings of the ACM symposium on Virtual reality software and technology VRST '05, pages: 82 – 85, 2005.

[8] Okutanoğlu, A. and Bozyiğit, M. “Proximity-Aware Synchronization within Federation Communities”. In Proceedings of the Tenth IEEE International Symposium on Distributed Simulation and Real-Time Applications (Ds-Rt'06) - Volume 00 (October 02 - 04, 2006). DS-RT. IEEE Computer Society, Washington, DC, 185-192.

SİMULASYON MODELLEME ALTYAPISI (SİMA) : DEVS TABANLI HİYERARŞİK VE MODÜLER BİR MODELLEME VE KOŞUM ALTYAPISI

Ahmet KARA , Doruk BOZAĞAÇ , M. Nedim ALPDEMİR

TÜBİTAK UEKAE / İLTAREN, Tunalı Hilmi Cad. Binnaz Sok. Kat:2 No:3, Kavaklıdere ANKARA, {ahmetk, dorukd, nedima}@iltaren.tubitak.gov.tr

ÖZ

Karmaşık sistemlerin simulasyon modellerinin tanımlanması ve koşturulması çoğu zaman zorlu bir süreci içerir. Bu tür karmaşık sistemlerin oluşturulmasında kullanılması planlanan simulasyon yazılım altyapılarında aranan özelliklerden anahtar önemde olanlardan bir tanesi de, karmaşık bir sistemin hiyerarşik ve modüler bir tarzda yeniden kullanılabilir alt-modellerin montajı ile oluşturulmasına destek sağlamasıdır. Ayrıca, tanımlanan modellerin koşturulma ilkelerini düzenleyen protokolün de formal bir temeli olan sağlam ve güvenilir bir mekanizmaya dayalı olması gerekmektedir. Bu makalede, ayıredici özellikleri aşağıda özetlenen Simulasyon Modelleme Altyapısı (SiMA)'nı tanıtmayı amaçlıyoruz: SiMA karmaşık sistem modelleri oluşturmada sistem teorisine dayalı formal bir temel sağlayan DEVS yaklaşımı esas almaktadır; SiMA simulasyon motoru, model koşum metodolojisi olarak paralel DEVS protokolünü uygulamaktadır; SiMA DEVS yaklaşımına özel bir port tipi ve geçiş fonksiyonu ekleyerek formal tanımını genişletir; SiMA, model tanımlama dökümanından otomatik olarak model hiyerarşisi oluşturan bir model oluşturma katmanı, hızlı ve ayarlanabilir bir simulasyon çıktı oluşturma altyapısı, veri iletişim altyapısını dönüştürme fonksiyonları ile ve veri tipi tutarlılığı ile destekleyen KOD Oluşturucusu (KODO) gibi ek araçları da içermektedir. Özet olarak SiMA, hiyerarşik model montajı, yeniden kullanılabilirlik ve birlikte çalışabilirlik desteği sunarak, karmaşık sistemlerin simulasyonuna dayalı uygulamalarda geliştirme zamanını önemli ölçüde azaltmayı hedeflemektedir.

Anahtar Kelimeler: Modelleme, Simulasyon, Yazılım Altyapısı, DEVS, yeniden kullanılabilirlik.

SIMULATION MODELING ARCHITECTURE (SIMA): DEVS BASED HIERARCHICAL AND MODULAR MODELING AND EXECUTION INFRASTRUCTURE

ABSTRACT

Defining and executing simulation models of complex systems is often a non-trivial task. One of the key requirements of simulation software frameworks for such complex systems is that the model construction should be supported by hierarchical and modular composition of re-usable sub models in a uniform and scalable way. Further, the accompanying simulation execution protocol should provide a robust and reliable mechanism for executing the models according to a well-defined modeling formalism.

In this paper we introduce Simulation Modeling Architecture (SiMA) which has the following distinct charateristics: SiMA is based on DEVS approach which provides a solid formal basis for complex model construction; SiMA Simulation Execution Engine implements the parallel DEVS protocol which provides a well-defined and robust mechanism for model execution; SiMA extends the DEVS formalism with a specific port type and transition function; SiMA comes with a variety of tools such as an automated model construcion pipeline from a model hierarchy specification document, an efficient and configurable trace generation framework, a Code Generation Tool (KODO) that supports message processing infrastructure through automated marshalling/unmarshalling accross the simulation layers to maintain type safety and consistency. As such, SiMA aims to significantly reduce the cost of simulation application construction for complex systems in the long run, by supporting hierarchical composability, re-usability and interoperability.

Keywords: Modelling, Simulation,Software Framework, DEVS, reusability, model composition.

1. GİRİŞ

Bileşen temelli (component-based) ya da modüler simulasyon tanımlama yaklaşımları simulasyon camiasında uzun süredir uygulanan ve araştırılan konulardır. Sürekli Simulasyon (continuous simulation) camiasında Modelica [4] modelleme dili, Dsblock [2] ve Simulink gibi sistemler; Kesikli Olay (Discrete Event) Simülasyon camiasında da DEVS [5], COST [11], SIMKIT [12] yaklaşımları bu konuda öne çıkanlardan bazıları olarak sayılabilir. Bütün bu yaklaşımların ortak noktası model ve simulatör yazılımları tasarımında belli bir disiplinin yerleşmesine ve yeniden kullanılabilirlik isterlerinin karşılanmasına yönelik arayışlarıdır. Bunların içinden DEVS yaklaşımının kayda değer bir özelliği, sürekli (continuous) sistemlerin quantizasyonuna dayanan bir yaklaşımla, içinde hem sürekli hem de kesikli karakterdeki bileşenlerin bulunduğu hibrid sistemler için bütünleştirici bir teorik çerçeve olarak kabul görmeye başlamasıdır [1]. Quantizasyon yaklaşımı, sürekli sistemlerin kesikleştirme (discretization) işlemini zaman boyutu (time base) yerine durum değerleri boyutunda (state base) gerçekleştirme fikrine dayanmakta olup simulasyon camiasında zaman kesikleştirmeye (time discretization) nispeten yeni olan bir yaklaşımdır [9]. Bu özelliğiyle DEVS yaklaşımı, farklı karakterdeki modeller için, hem *model tanımlama* hem de *koşum zamanı simulasyon bileşenleri* düzeyinde, hiyerarşik model inşa disiplini, yeniden kullanılabilirlik ve birlikte çalışabilirlik ilkelerini destekleyebilme potansiyeli taşımaktadır.

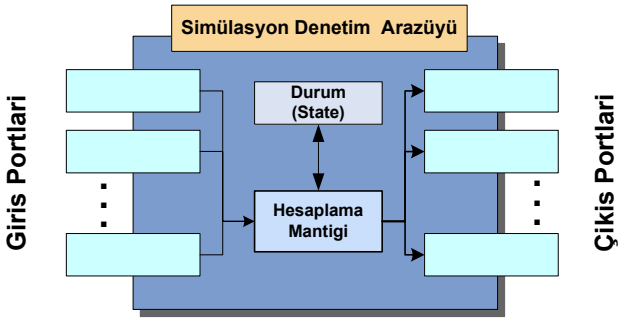
Bu bildiride tanıtılan Simülasyon Modelleme Altyapısı (SiMA), karmaşık bir sistemin simulasyon modelinin, modüler ve hiyerarşik bir mimari ile oluşturulup koşturulmasına olanak sağlayan bir altyapı yazılımıdır. SiMA, karmaşık sistem modelleri oluşturmada sistem teorisine dayalı formal bir temel sağlayan DEVS yaklaşımını esas almakta, simulasyon motoru da model koşum metodolojisi olarak paralel DEVS protokolünü [5] uygulamaktadır. DEVS yaklaşımını temel alan PowerDEVS [9], DEVSim++ [7], DEVS-C++ [6] ve DEVSJava [8] gibi simulasyon ortamları da bulunmaktadır. Bunlardan DEVSJava ve DEVSim++ kullanılarak, diğerleri ise dökümantasyonları üzerinden incelenmiş olup, genellikle araştırma amaçlı olarak üniversitelerde

geliştirilmiş olmaları nedeniyle yüksek performans, kararlılık ve güvenilirlik gerektiren üretim ortamları için yeterli olgunluk düzeyinde olmadıkları izlenimi edinilmiştir. Buna binaen hem kurumumuzda yürütülen modelleme ve simulasyon projelerinde kullanılacak sağlam bir yazılım altyapısı oluşturmak, hem de Bölüm 0'de belirtilen eklentilerin desteklenmesi gibi sebeplerle SiMA geliştirilmiştir. SiMA, modellenecek olan sistemden bağımsız bir yazılım katmanı sunarak simulasyon uygulamalarında gereksiz kodlama tekrarlarını önlemeyi, yeniden kullanılabilirlik ve birlikte çalışabilirlik ilkelerini desteklemeyi, böylece yazılım geliştirme zamanı ve işgücünü en aza indirmeyi hedeflemektedir.

Bildirinin bundan sonraki kısmı aşağıdaki bölümleri içermektedir: 2. bölümde SıMA'nın ana hatlarıyla yapısı, çalışma ilkeleri ve teorik arkaplanı anlatılmakta, 3. bölümde SiMA'nın performans özelliklerini anlatmak amacıyla yapılan test sonuçları tartışılmakta, 4. bölümde ise sonuçlar sıralanmaktadır.

2. SİMA'NIN ANATOMİSİ

SiMA'da tanımlanabilecek simülasyon modelleri *atomik* ve *bağlaşık (coupled)* olmak üzere iki ayrı tipte olabilmektedir. Atomik modeller modellenen sistemin fonksiyonel alt bileşenlerini oluşturur. Bağlaşık modeller ise doğrudan bir model tanımı içermeyip başka atomik veya bağlaşık modellerin hiyerarşik bir yapıda birbirine bağlanmasıyla tanımlanırlar.

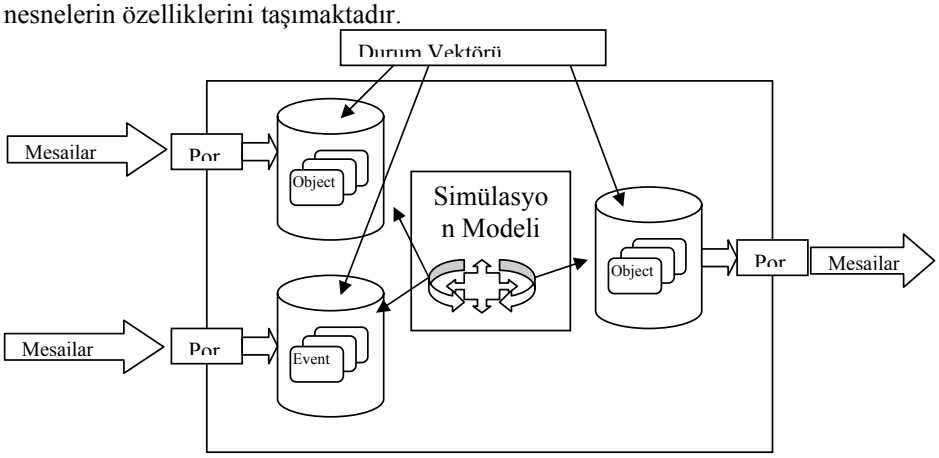


Şekil 1. Ana Hatlarıyla Simülasyon Modellerinin Yapısı

Atomik ve bağlaşık modeller, üzerlerinde tanımlı portlar aracılığıyla simülasyon içerisinde yürüyen bilgi akışına katılırlar. Her bir model simülasyona katacağı her bir nesne/anlık veri (event) tipi için birer çıkış portu açmak zorundadır. Aynı şekilde modeller ilgilendikleri ve kendi iç işlemlerinde kullanma ihtiyacı duydukları veri tiplerinin her biri için birer giriş portu açarlar.

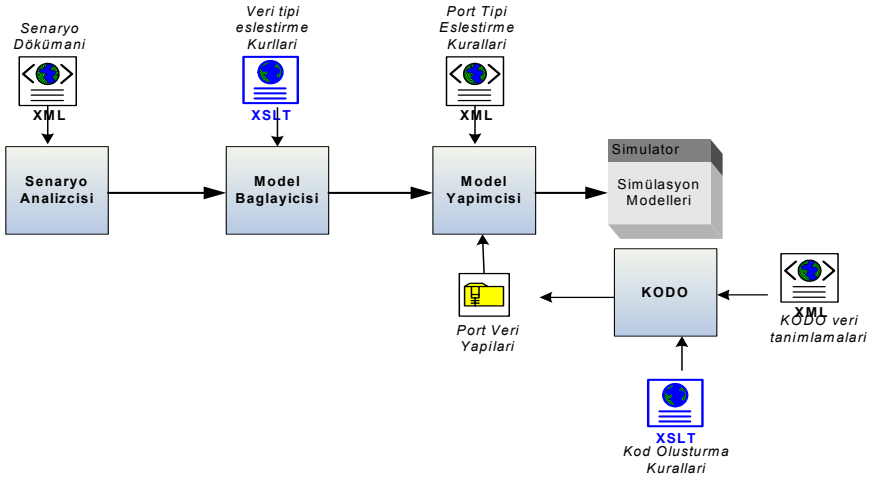
İki çeşit port bulunur: *ReceiverPort* ve *DispatcherPort*. Bunlar bağlaşık ve atomik modeller içerisinde farklı şekillerde kullanılırlar. *ReceiverPort* atomik modeller tarafından giriş portu olarak kullanılır ve sadece aldığı veriyi saklamakla yükümlüdür. DispatcherPort ise atomik modellerde sadece çıkış portu, bağlaşık modellerde ise hem giriş hem de çıkış portu olarak kullanılır ve aldığı veriyi kayıtlı olan alıcılarına

dağıtmakla yükümlüdür. SiMA atomik modelleri birbirleriyle bağlantı portlarından aktardıkları mesajlar ile haberleşerek verilerini paylaşırlar. Aktarılan bu mesajların yapısı port tiplerine özel olmakta ve simülasyon içerisinde modellemesi yapılan nesnelerin özelliklerini taşımaktadır.



Şekil 2. Akıllı Atomik modelin iç yapısı

SiMA sisteminde kullanıcının model geliştirme sürecini kolaylaştırmak amacıyla temel modelleme birimi olarak Şekil 2'de gösterilen *akıllı atomik model* geliştirilmiştir. Akıllı atomik model, port ve durum verilerinin yönetiminde sistematik bir mekanizma içermekte, model geliştiricilerine veri yönetimi için elverişli bir soyutlama katmanı sunmaktadır.



Şekil 3. Simulasyon modelleri oluşturma süreci

SiMA altyapısı KODOluşturucusu(KODO), Senaryo Analizcisi, Model bağlayıcı ve yapımcısı gibi yardımcı araçlarla desteklenerek, senaryo tanım dökümanından (XML dökümanı) otomatik olarak SiMA modellerinin ve SiMA protokolünü uygulayacak simülatörün oluşturularak koşturulmaya hazır bir modül haline getirildiği model oluşturma süreci otomasyonu gerçekleştirilmiştir. Bu otomasyon süreci Şekil 3'teki diyagramda görsel olarak ifade edilmiştir.

2.1 Formal Temeller

Bu bölümde SiMA nın temel aldığı formal yaklaşım anlatılmıştır. SiMA'da temel modelleme birimi olan bir atomik modelin simülasyon sırasında nasıl davranacağı, zamana bağımlılığın nasıl yürütüleceği iyi tanımlanmış bir formal yapı tarafından (DEVS formalizmi) belirlenmiştir. DEVS yaklaşımına göre formal olarak bir simülasyon modeli aşağıdaki yapı ile gösterilir:

$$M = < X, S, Y, \delta_{\text{int}}, \delta_{ext}, \lambda, ta > \quad (1)$$

Öyle ki:

X: girdi değerleri kümesi
S: durum değerleri kümesi (set of states)
Y: çıktı değerleri kümesi
δ_{int} :$S \rightarrow S$ dahili değişim fonksiyonu (internal transition function)
δ_{ext} : $QxX \rightarrow S$ harici değişim fonksiyonu (external transition function). Öyle ki:
$Q = \{(s,e) \mid s \in S, 0 \le e \le ta(s)\}$ durum kümesi ve
e en son değişimden *şu ana kadar geçen zaman.*
λ : $S \rightarrow Y$ *çıktı fonksiyonu*
ta : $S \rightarrow R^{+}_{0,\infty}$ zaman ilerletme fonksiyonudur *(0 ve* $\propto$ *dahil pozitif reel sayılar)*

Bu formal model uyarınca, verili bir zamanda sistemin s durumunda olduğunu varsayarsak, eğer giriş portlarından girdi değeri varmıyorsa sistem s durumunda e=ta(s) kadar kalacak sonra $\lambda(s)$fonksiyonunu kullanarak y değerlerini üretecek, hemen ardından da δ_{int} *(s)* fonksiyonunu (yani dahili değişim fonksiyonunu) uygulayarak s' durumuna geçiş yapacaktır. Eğer e=ta(s) zamanı geçmeden önce (yani e ≤ ta(s) ise) bir girdi değeri gelirse o zaman sistem δ_{ext} *(s,e,x)* (yani s, e ve x değerlerine bağlı harici değişim fonksiyonu) fonksiyonunu kullanarak s' durumuna geçiş yapacaktır. ta(s) tamamlandığında ise (yani simulatorün kendi dahili zaman aralığı tamamlamdığında) $\lambda(s)$fonksiyonu kullanılarak y değerleri üretilecek, hemen ardından da δ_{int} *(s)* fonksiyonu uygulanacaktır. Her iki durumda da sistem artık yeni bir durum seti (S') ve yeni bir zaman aralığına (ta(s')) sahip olacak ve çevrim böylece sürüp gidecektir.

2.2 DEVS tanımına SiMA eklentileri

SiMA altyapısının geliştirilme sürecinde ortaya çıkan çeşitli ihtiyaçların ve iyileştirme arayışlarının bir sonucu olarak DEVS modeline iki noktada katkı sağlanması söz konusu olmuştur. Bunlardan birincisi model portlarını *veri tipi* bazlı sınıflandırmaya imkan sağlayacak "port tipi" kavramının tanımlanmasıdır. İkincisi ise modellerin birbirlerinin durum değerlerine aynı simulasyon adımı içinde erişmesini gerekli kılan durumların

genelleştirilmesi suretiyle DEVS formal modelinin ek bir fonksiyonla genişletilmesidir. *DirectFeedThrough (DFT) Transition Function* adı verilen bu ek fonksiyonunun amacı DFT tipindeki giriş portlarında gerçekleşen veri değişimlerine anında tepki vererek (ve gerektiğinde bu verilere basit transformasyonlar uygulayarak) çıkış portlarına aktarmak, böylece aynı zaman adımı içinde modeller arası veri iletişimini hızlı bir biçimde sağlamaktır. Başka bir deyişle bu eklemeyle modellerin simülasyon zamanını ilerletmeden mevcut durum verilerini başka modellerle paylaşmaları, ya da başka modellerden gönderilen verilere basit ekleme/dönüştürme işlemleri uygulayarak diğer modellere iletmeleri mümkün olmaktadır. Bu mekanizma HLA standardındaki *Episodik Simülation* ve *zero look-ahead* [13] yaklaşımlarıyla benzer bir mantığı uygulamaktadır. Bu eklemeyle elde edilen DEVS formal modeli aşağıdaki şekilde gösterilebilir:

$$M =< X, S, Y, \delta_{\text{int}}, \delta_{ext}, \lambda, ta, \delta_{df} > \qquad (2)$$

Öyle ki:

X: P_{in} girdi portlarından modele ulaşan değerler kümesi,

Y: P_{out} çıktı portlarından model dışına gönderilen değerler kümesi,

P_{in} : Giriş portları kümesi, öyleki $P_{in} = \{(\tau, I_x) \mid \Gamma \mapsto \tau \wedge I_x \subseteq X\} \wedge$ for $\forall x \in I_x$, $\tau \rightarrow x$ (giriş portları bir (τ, I_x) ikilisidir ki τ , Γ veri tipi sistemine göre geçerli bir tip *(port tipi)*, ve I_x ilgili giriş portundan modele ulaşan verilerin kümesi olup X kümesinin alt kümesidir),

P_{out} : Çıkış portları kümesi, öyleki $P_{out} = \{(\tau, O_y) \mid \Gamma \mapsto \tau \wedge O_y \subseteq Y\}$ (çıkış portları bir (τ, O_y) ikilisidir ki τ , Γ veri tipi sistemine göre geçerli bir tip *(port tipi)*, ve O_y ,ilgili çıkış portundan dışarı gönderilen verilerin kümesi olup Y kümesinin alt kümesidir),

$\delta_{df} : PDFT_{in} \in P_{in} \rightarrow \{PDFT_{out_1}, ..., PDFT_{out_n}\} \subseteq P_{out}$: *direct feed through transition* fonksiyonudur ki bu fonksiyonun tanım kümesi giriş portlarının bir elemanı (tek bir port) değer kümesi ise çıkış portlarının bir alt kümesidir.

Burada dikkat çekilmesi gereken nokta, model portlarının belli bir veri tipi sistemine göre geçerli olan veri tipleri ile belirleniyor olmasıdır ($\forall x \in I_x$, $\tau \mapsto x$). Her portun tanımlı bir tipinin olması veri akış ve portlar arasında tip bazlı eşleştirmelere zemin hazırlamakta, eşleşen portların otomatik olarak bağlantılantırılmasına (automatic port mapping) olanak sağlamakta, simulasyonun bilgi uzayının güvenli ve sağlam bir altyapı ile oluşturulması sağlanmaktadır.

Belirtmek gerekir ki *Direct feed through transition* fonksiyonu potansiyel olarak modeller arasında döngülere (algebraic loops) yol açabilecek bir yapı olduğundan, teorik açıdan model meşruiyeti (DEVS model legitimacy [5]) ihlallerine engel olmak amacıyla SiMA'nın port verileri işleme altyapısı bu tür döngülerin oluşmamasını garanti altına alan bir emniyet mekanizması içermektedir. Bu emniyet mekanizması DFT portlarına erişimi denetleyerek model meşruiyetini ihlal edecek döngüleri program içinde tanımlı özel bir kilit aracılığıyla engellemektedir.

2.3 Koşum Protokolü

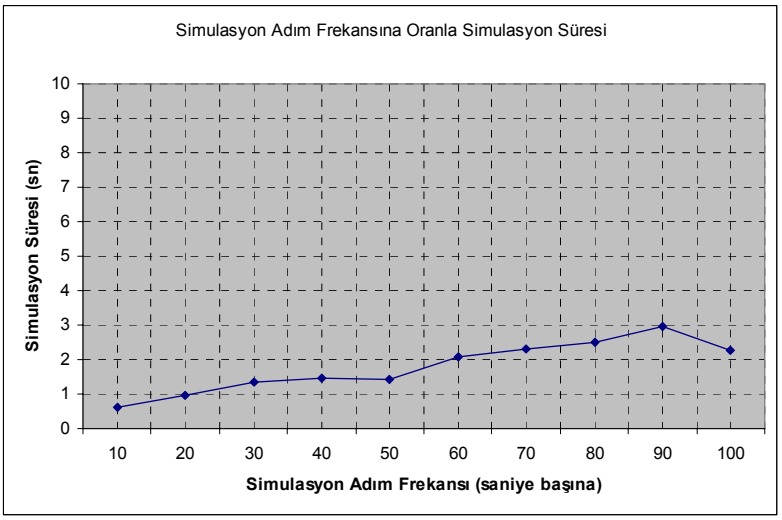
SiMA simülasyon protokolunu simülator ve koordinator bileşenleri aracılığıyla uygular. Bağlaşık model ve atomik model bileşenleri de simülator ve koordinatorler ile harmonik bir ilişki içindedirler. Bütün simülasyonu yöneten bir ana simülator bulunmaktadır. Bu simülator, hiyerarşinin en tepesindeki bağlaşık modelin simülasyon denetim arayüzünü kullanarak protokolü uygular. Her Bağlaşık model bir Koordinator ile eşleşir ve koordinatörün simülasyon denetim arayüzünü kullanarak protokolü uygular. Her koordinator bağlı olduğu bağlaşık modelin içinde bulunan modeller ile eşleşir. Koordinator bu alt modellerin simülasyon denetim arayüzünü kullanarak protokolü uygular. Böylece simülasyon protokolü tepeden en alttaki atomic modellere kadar düzenli ve istikrarli bir biçimde uygulanır.

3. PERFORMANS ÖZELLİKLERİ

Bu bölümde SiMA kullanılarak oluşturulmuş olan simulasyon uygulamalarının performans özelliklerinin incelenmesi amaçlanmıştır. Performans özelliklerinin iki ana noktadan incelenmesi planlanmıştır:

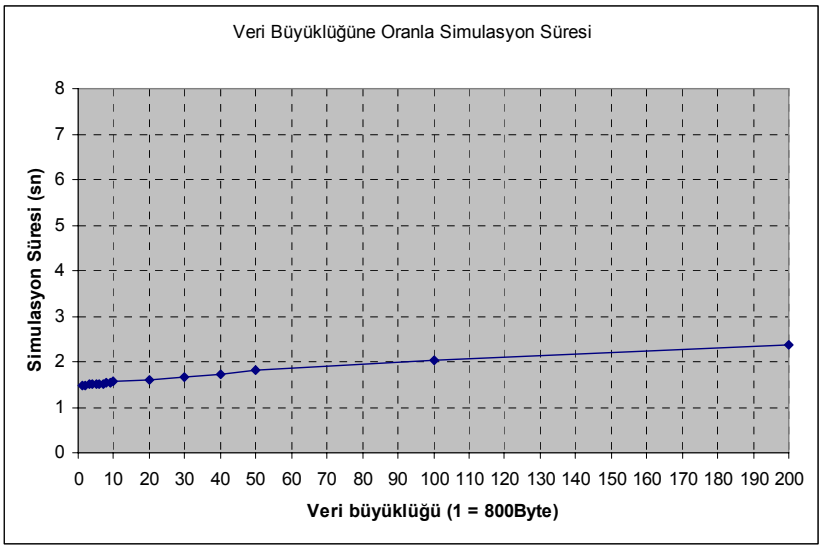
1. Simulasyon adım aralığının simulasyon performansına etkisi. Burada yapılan simulasyon modellerinin çalışma sıklığını artırarak (simulasyon adım aralığını azaltarak) bunun simulasyon motoru ve portlar arası veri aktarım altyapısının işleyiş performansını nasıl etkilediğini gözlemlemektir.
2. Portlar arasında akan veri büyüklüğünün (message size) artmasının simulasyon performansına etkisi. Burada amaçlanan, simulasyon adım aralığı sabit tutulduğunda modellerin ürettiği veri miktarındaki artışa mesaj iletim altyapısının nasıl cevap verdiğini gözlemlemektir.

Yapılan testlerde AMD 4200+ X2 Dual işlemcili, 2GB RAM Belleği olan bir bilgisayar üzerinde Windows XP Professional 64 bit kurulu olarak kullanılmıştır. Simulasyon modelleri iki savaş sistemi modeli (ikisi de SiMA bağlaşık (coupled) model) ve bir ortam modeli (SiMA atomik model) olacak şekilde toplam 3 SiMA modelinden oluşmaktadır.



Şekil 4. SiMA ile yapılan testlerde simulasyon adım aralığına bağlı performans değerleri grafiği

Birinci sınıfta SİMA ile yapılan testlerde simülasyon adım frekansı 10 Hz ile 100 Hz (100ms ile 10 ms lik simulasyon adımları) arasında 10 Hz lik artış aralığı ile değiştirilmiştir. Böylece, özellikle simulasyon hassasiyet isterlerinin küçük simulasyon adımlarını dayattığı durumlarda altyapının buna nasil tepki vereceğine ilişkin sonuçlara ulaşmak hedeflemiştir. Şekil 4'te de görüldüğü gibi altyapının bu artışa üstel veya üstele yakın değil de düşük eğimli, doğrusal bir artışla cevap vermesinin, altyapının sağlıklı işlediğinin bir göstergesi olduğu değerlendirilmektedir.



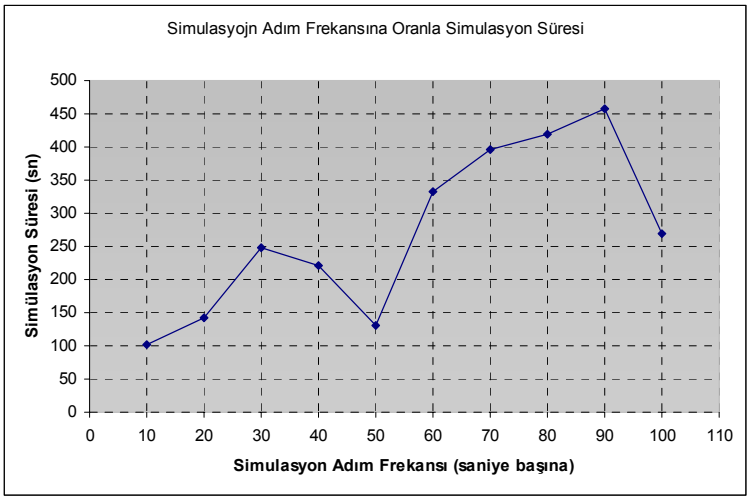
Şekil 5. SiMA ile yapılan testlerde veri büyüklüğüne bağlı performans değerleri

İkinci sınıfta yer alan testler sırasında da portlar arasında iletilen verilerin kademeli olarak artırılması ile simulasyon modülleri arasında dolaşan veri miktarının artması sağlanmıştır. Bu testlerde simulasyon model portları arasında akan verilerin miktari yaklaşık olarak 800 Byte ile 160 KB arasında değiştirilmiştir. Sadece veri artışının altyapı üzerindeki etkisi test edilmek istendiğinden, bu testlerde iletilen veriler modeller tarafından işlenmemektedir. Şekil 5'te verilen grafikte görüldüğü üzere veri büyüklüğü artışının (160KB'ta kadar olan dilim içerisinde) simulasyon performansına olan etkisi düzenli (uniform) ve logaritmiktir (200 katlık bir veri artışı simulasyon süresinde iki katın altında bir artışa yol açmıştır). Bu verilere bakıldığında SiMA'nın portlar arası mesaj iletim altyapısının sağlıklı işlediği sonucuna varılabileceği değerlendirilmektedir. Tekrar vurgulamak gerekir ki burada sadece mesaj iletim altyapısının darboğaz oluşturup oluşturmadığının anlaşılması hedeflendiği için, simulatörler iletilen verileri işleyen akıllı kodlar içermemektedirler. Bu nedenle artışın doğrusal yerine logaritmik olması yadırganmamalıdır.

3.1 SiMA HLA Adaptörü ile yapılan RTI merkezli simulasyon testleri

SiMA ile yapılan testlerin aynısı HLA-tabanlı bir simulasyon ortamında da tekrarlanarak sınırlı bir karşılaştıma imkanı oluşturulmak istenmiştir. Burada SiMA ile RTI bazlı ortamların öncelikleri ve yapıları itibariyle tümüyle eşdeğer sistemler olmadıkları kabul edilmekle beraber, yaygın kullanıma konu olan ve bir simulasyon birlikte çalışabilirlik standardı olan HLA-tabanlı sistemlerle yapılan karşılaştırmanın SiMA'nın davranış paterni konusunda daha sağlıklı bir fikir verebileceği değerlendirilmiştir. RTI testleri için kullanılan donanım altyapısı, RTI bilgisayarı olarak 8*AMD 4400+ X2 Opteron işlemcili, 16GB bellek (RAM) kapasiteli, üzerinde Windows 2003 Enterprise 64 bit kurulu bir sunucu, Federelerin kurulduğu bilgisayar

olarakta AMD 4200+ X2 Dual işlemcili, 2GB bellek (RAM) kapasiteli, üzerinde Windows XP Professional 64 bit kurulu bir bilgisayar kullanılmıştır. Ağ hızı 100Mbit/s dir. Simulasyon modelleri SiMA testlerinde kullanılan modellerle tamamen aynıdır. İki savaş sistemi federesi ve bir ortam federesi olacak şekilde toplam üç federe kullanılmıştır. Federelerin iç implementasyonu SiMA modelleri ile aynı kodlarla gerçekleştirilmiştir.



Şekil 6. HLA ile yapılan testlerde simulasyon adım aralığına oranla simulasyon süresi grafiği

Değerlere bakıldığında performans açısından, SiMA'nın frekans artırım testleri baz alındığında ortalama 150 kat hızlı çalıştığı gözlenmektedir. HLA test verilerini gösteren Şekil 6'daki grafikte göze çarpan diğer nokta da dalgalanmalardır.Şekil 6'daki adım aralığı testlerinde genel bir artış eğilimi olmakla beraber göze çarpan bir dalgalanma eğilimi de gözlenmektedir. Testler gece vakti, ağ kullanımının olmadığı zaman diliminde yapılmış olmasına karşın nispeten istikrarsız sonuçlar alınması, özellikle ağ ortamının simulasyonun işleyişini deterministik olmayan bir tarzda etkileyebileceği anlamına geldiği değerlendirilmektedir.

4. SONUÇ

Bu makalede SiMA'nın ana hatlarıyla yapısı ve çalışma prensipleri ile üzerine oturtulduğu teorik temeller tanıtılmış, ayrıca 3 bileşenli (federeli) bir simulasyon kullanılarak yapılan testlerin sonuçları verilmiştir. Yapılan testlerde SiMA altyapısının sergilediği davranış motiflerinin, modeller arası mesaj iletim miktarı, ve modellerin çalıştırılma sıklığının (simulasyon adım aralığının) etkileri söz konusu olduğunda, sağlıklı bir eğilim sergilediği gözlemlenmiştir. Ayrıca aynı simulasyonun HLA tabanlı bir ortamda tekrarlanması ile elde edilen sonuçlarla yapılan karşılaştırmalarda elde edilen yaklaşık değerlere bakıldığında SiMA'nın yaklaşık 150 kat daha hızlı çalıştığı gözlenmiştir. Bunda başlıca etkenin SiMA'nın model koşturumunu ve dolayısıyla da

modeller arası iletişmi ağ ortamına çıkmadan, aynı (ve bir tek) process içinde yürütmesi gerçeği olduğu söylenebilir.

KAYNAKÇA

[1] Kofman, E and Junco, S (2001), "Quantized State Systems. A DEVS Approach for Continuos System Simulation" Transactions of SCS, Vol 18 (3), pp 123-132

[2] Otter, M. and Elmqvist, H. (1995), "The Dsblock model inteface for exchanging model components",proc. EUROSIM'95, pp 505-510, 1995.

[3] Praehofer, H., (1991), "System Theoretic Foundations for combined Discrete – Continuous System Simulation", PhD Thesis, J Keper University of Linz, 1991.

[4] Mattson, S.E. and Elmqvist, H. "An overview of the modelling language Modelica", Proc. Eurosim Simulation Congress (EUROSIM'98), pp 182-186, Espoo, Finland, 1998

[5] Zeigler, B.P., Praehofer, H. and Kim, T.G., "Theory of Modeling and Simulation", Academic Press, Second Edition, 2000

[6] Cho, H and Cho, Y. DEVS–C++ Reference Guide. The University of Arizona, 1997

[7] Kim, T.G., DEVSim++ User's Manual. C++ Based Simulation withHierarchical Modular DEVS Models. Korea Advance Institute of Science and Technology, 1994.

[8] Zeigler, B. and Sarjoughian, H. Introduction to DEVS Modeling and Simulation with JAVA: A Simplified Approach to HLA-Compliant Distributed Simulations. Arizona Center for Integrative Modeling and Simulation, 2000.

[9] Kofman, E., Lapadula, M.and Pagliero, E. PowerDEVS: A DEVS–Based Environment for Hybrid System Modeling and Simulation, Technical Report LSD0306, Universidad Nacional de Rosario, 2003

[10] Cellier,F. and Kofman E, "Continuous System Simulation", Springer, 2006

[11] Chen G. and Szymanski B.K., "COST: A Component-Oriented Discrete Event Simulator", Proc. of the 2002 Winter Simulation Conference, pp. 776-782, 2002

[12] Buss A. "Component-Based Simulation Modellig with SIMKIT", Proc. of the 2002 Winter Simulation Conference, pp. 243-249, 2002

[13] Kuhl, F., Weatherley, R. and Dahmann J."Creating Computer Simulation Systems: An Introduction to the High Level Architecture", Prentice Hall PTR, 2000

ÖLÇEKLENDİRİLEBİLİR HLA UYUMLU SİMÜLASYONLAR İÇİN UYGULAMA ÇATISI: SİMALT

Göktuğ F. GÖKDOĞAN [(a)], Günay ÖZKAN [(a)], Turgay ÇELİK [(a)]

(a) MilSOFT Yazılım Teknolojileri, ODTU Teknokent / ANKARA,
{gokdogan, gunayozkan, turgaycelik}@gmail.com

ÖZ

HLA Federasyon geliştirimi yavaş, hataya açık ve maliyetli bir süreçtir. Federasyon Anlaşması aracılığıyla çözülen, simülasyon kontrolü, federasyon yönetimi, senaryo dağıtımı ve yönetimi gibi tekrarlayan problemler HLA standardı ile çözüme kavuşturulmamıştır. Bunların yanı sıra, olgu veya tür tabanlı federe mimarisinin seçilmesi gibi mimari bir kararda ölçeklendirilebilirlik bir kısıt olabilir. Örneğin 100 tank olgusunun olduğu bir simülasyon uygulaması için olgu tabanlı federeler uygun değildir. Bir başka problem, tüm geliştiricilerin HLA bilme zorunluluğu olması ve modülerliği artırmak için modelleme mantığı ile HLA kavramlarının birbirinden ayrılması gereğidir. Bu makalede, söz konusu problemleri aşmak için, birlikte çalışabilirlik ve ölçeklendirilebilirliği artıracak, modelleme mantığı ile HLA kavramlarını tam olarak birbirinden soyutlayıp geliştirme ve bakım maliyetlerini düşürecek bir simülasyon geliştirme uygulama çatısı anlatılacaktır.

Anahtar Kelimeler: Eğitim Simülasyonu, FOM / SOM, Gerçek Zamanlı Simülasyon, İleri Seviye Mimarı, Koşum Zamanı Altyapısı (KOZA), Nesne Yönelimli Altyapı, Simülasyon Ölçeklendirilebilirliği, Zaman adımlı Simülasyon

A FRAMEWORK FOR RAPID DEVELOPMENT OF SCALABLE HLA-COMPLIANT SIMULATIONS: SIMALT

ABSTRACT

Developing HLA federation is a slow, error-prone and costly process. There are recurring problems like simulation control, federation management, scenario distribution and management which usually solved through Federation Agreements but not through HLA standard itself. Also in some cases scalability is a hard constraint in architectural choices like developing instance or type based federates. (i.e. can not go with instance based federates in a simulation with 100 tanks in the field). Another problem is a need on separation of concerns between modeling and HLA issues; not every developer/modeler needs to be HLA literate.

In order to overcome these problems we propose a simulation development framework for increasing interoperability and scalability, providing full separation of modeling and HLA concerns with decreased development and maintenance costs.

Keywords: Object Oriented Framework, High Level Architecture (HLA), FOM / SOM, Real-time Simulations, Runtime Infrastructure (RTI), Simulation Scalability, Time-stepped Simulations, Training Simulations

1. GİRİŞ

HLA uyumlu yazılım geliştirme süreci zor ve maliyetlidir. Ayrıca, HLA uyumlu yazılımlar, dağıtık doğaları gereği karmaşıktır, uygulamaların anlaşılması zordur. Uygulama, kendi işlevselliğinin yanında diğer yazılım birimleriyle haberleşmek için kullanılan HLA arayüzlerine çağrılar ve RTI geri çağrımlarını karşılayacak kesimler içerir. Bu durum uygulamanın bakım maliyetini artırır.

Bu makalede bu zorluklar göz önünde bulundurularak geliştirilen simülasyon altyapısı ve dağıtık nesne katmanı ele alınacaktır. Bu kapsamda, ikinci kesimde klasik yöntemle HLA Uyumlu Yazılım Geliştirme sürecinde karşılaşılması muhtemel problemler örnek bir uygulama ile ortaya konacak; üçüncü kesimde, SIMALT simülasyon altyapısı tanıtılacak, haberleşmeyi sağlayan dağıtık nesne katmanı, uygulama mantığını içeren Simülasyon Modülü, simülasyon koşumunu kontrol eden Simülasyon Motoru anlatılacak; dördüncü kesimde SIMALT'ın ölçeklendirilebilirliği incelenecektir ve örnek bir genişleme uygulaması sunulacaktır.

2. KLASİK YÖNTEMLE HLA UYUMLU YAZILIM GELİŞTİRME

Bu kesimde örnek bir senaryo tanımlanarak geliştirme sürecinde yaşanması olası problemler ortaya konmaktadır.

Örnek olarak hazırlanan simülasyon geliştirme senaryosunda: komuta kontrol, muhabere, keşif, gözetleme ve istihbarat amaçlı sistemler ile tank, top, uçak, füze vb. savunma sistemlerini içeren bir muharebe ortamının eğitim amaçlı olarak simülasyonunun geliştirilmesi amaçlanmaktadır.

Bu problemin klasik bir çözümü şu şekilde olabilir: Modelleme ihtiyaçlarından yola çıkarak, her bir sistem modeli ayrı bir federe olarak tasarlanır. Bu durumda modeli geliştirilecek sistem sayısı kadar federe olur ve her bir federe birden fazla olguyu idame ettirir.

İlerleyen kesimlerde çözülmesi gereken alt problemler irdelenecektir.

2.1 Federe İçi Veri İletişimi

Nesne tabanlı düşünüldüğünde, bir nesnenin HLA KOZA (RTI) üzerinden dinlediği diğer federelerdeki nesnelerin yanı sıra, kendisiyle aynı federede olan diğer nesneleri de dinlemesi gerekecektir. Örneğin arazide hareket eden bir tank nesnesi, ancak hem kendi federesindeki hem de diğer federelerdeki unsurları dinleyerek çarpışma kontrollerini yapabilir. Bu durumda en kolay çözüm, tasarımı bir federe aynı anda tek bir olguyu modelleyecek şekilde değiştirmek olacaktır. Ancak senaryo alanında olgu sayısının yüksek olması yada değişken olması bunu bir alternatif olmaktan

çıkarmaktadır. Böyle bir durumda mevcut mimari tasarıma sadık kalınırsa, aynı ya da farklı federelerde modellenen olguların tek bir mekanizma ile dinlenmesi uygun bir yaklaşım olacaktır. Bu noktada her federe için iç ve dış olgular arasında standart bir iletişim sağlanması ile ilgili bir gereksinim ortaya çıkmış olur.

2.2 Ölçeklendirilebilirlik

Her bir federenin ayrı birer görev olarak çalıştığı varsayılarak, görevlerin mevcut sunuculara en iyi yük dağılımını sağlayacak şekilde dağıtılması gerekir. Bu noktada karşılaşılabilecek ilk sorun, her bir federenin yükünün senaryo bazlı olarak farklılık gösterecek olmasıdır. Bu da farklı senaryolar için farklı görev dağılımları oluşturma ihtiyacını doğurur ve iyi bir yük dağılımını uygulamada imkansız hale getirir.

Bir başka sorun ise, aynı anda farklı iki senaryoyu koşturma ihtiyacını karşılama durumunda, çalışması gereken görev sayısının iki katına çıkmasıdır. Bu durum aynı anda birçok katılımcının bağımsız olarak senaryo koşturma ihtiyacı ile daha da karmaşık hale gelir. N sayıda katılımcının ve M adet modelin olduğu varsayıldığında toplam federe sayısı:

Her bir senaryo için:

(1 Katılımcı Federesi + M Model Federesi) = 1+M federe

N adet senaryo için:

N x (1+M) = N + NxM federe olacaktır.

Katılımcı sayısı 20, model sayısı 10 kabul edilirse, 220 federe/görev olacaktır. Bu noktada altyapı problemlerinin yanı sıra, ticari HLA KOZA'ların federe adedine göre lisanslanıyor olması, ciddi bir maliyet artışına neden olacaktır.

2.3 Tekrar tekrar karşılaşılan problemler

Federasyon Anlaşması (*Federation Agreement*), HLA aracılığı ile tanımlanamayan, ancak aynı federasyondaki bütün federelerin uyması gereken kuralları tanımlar. Bunlardan federasyon yönetimi, simülasyon kontrolü, senaryo dağıtımı gibi hususların her federasyon için tekrar tanımlanması gerekir ve bu tanımlar federasyonlar arası paralellik gösterir.

Örneğin birçok simülasyon Açılma, Planlama, Koşma ve Bitiş olarak 4 safhada düşünülebilir. İlk etapta federasyonu oluşturacak bir federeye ihtiyaç duyulur. Ayrıca safhalar arası geçişleri düzenlemekten, ilgili senkronizasyon noktalarının duyurmaktan sorumlu bir federenin olması gerekir. Bu noktada sorumlu federeler kadar, kullanılan yöntemler de önemlidir. Bir önceki örnekte safhaların birbirinden ayrımının senkronizasyon noktaları ile yapılacağı her federe tarafından önceden kararlaştırılmış bir yöntemdir. Benzer şekilde senaryonun federasyonda nasıl dağıtılacağı da kararlaştırılması gereken başka bir konudur. Bu ve benzeri birçok husus, HLA ile bir standarda kavuşturulmamış, ancak her federasyon için tekrar çözümlenip, karara bağlanması ve alınan kararlara göre gerçekleştirim yapılması gereken konulardır. Bu noktada üretilmiş farklı çözümler aynı zamanda federelerin ilerde birbirleri ile uyumlu çalışmalarını da güçleştirecektir.

2.4 HLA ile Modelleme Kodunun Ayrılması

HLA'da tüm federeler iki kesimden oluşur. Birinci kesim uygulama mantığını içerirken, ikinci kesim diğer federelerle haberleşmeyi sağlayan HLA çağrı ve geri dönüşlerini (*callback*) içerir. Bu iki kesim bir arada geliştirildiğinde uygulama mantığı ile haberleşme mantığı iç içe girmiş olacaktır. Bu durumda tüm federe geliştiricilerinin hem sorumlu oldukları federenin iş mantığını hem de karmaşık HLA haberleşme mantığını bilmeleri gerekir. Bu da zaman alıcı ve maliyeti yüksek olan HLA uyumlu yazılım geliştirme süreci eğitimlerinin tüm geliştiricilere sağlanmasını gerektirir. Tüm geliştiricilerin HLA uyumlu yazılım geliştirme sürecine aşinalığı sağlansa bile, her federede uygulama mantığı ile iç içe olan HLA kodları, bakım maliyetlerini artıracak, ilgili federe farklı bir geliştiriciye verildiğinde adaptasyon sürecini uzatacaktır.

3. SIMALT'IN YAPISI VE ALT BİLEŞENLERİ

Yukarıda anlatılan problemlerin kolaylıkla üstesinden gelinebilmesi ve özellikle de gerçek zamanlı zaman adımlı simülasyonların hızlı bir şekilde geliştirilebilmesi için SİMALT adında bir simülasyon altyapısı öneriyoruz.

SİMALT'ı 3 başlık altında inceleyebiliriz:

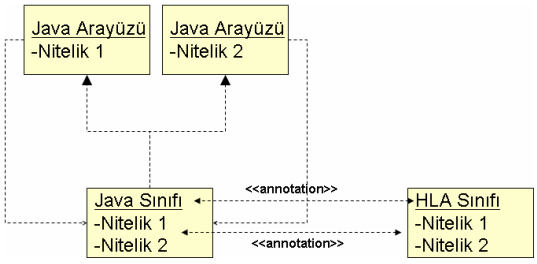
3.1 Dağıtık Nesne Katmanı

Dağıtık Nesne Katmanı (DNK) nesne tabanlı bir HLA soyutlaması sağlar. Bu soyutlama için HLA'daki etkileşim ve nesne sınıflarına karşılık POJO (Plain Old Java Object) sınıflarını kullanır. Bu sınıflara genel olarak Paylaşılan Veri adı verilir.

3.1.1 Paylaşılan Veri

Paylaşılan Veri'yi FOM ve SOM'larımızın Java'daki nesne karşılıkları olarak özetleyebiliriz. FOM'da geçen her sınıfa karşılık Paylaşılan Veri'de somut (*concrete*) bir Java sınıfı vardır. Bunlara Gerçekleştirim Sınıfı adı verilir. Gerçekleştirim Sınıfları FOM'daki sınıflar arası hiyerarşiyi aynen yansıtır. Her Gerçekleştirim Sınıfı ilgili olduğu HLA sınıfını bir *annotation* ile işaret eder. Gerçekleştirim Sınıfının her bir niteliği de ilgili oldukları HLA niteliğini yine bir *annotation* ile işaretler. Gerçekleştirim Sınıfları ayrıca her bir nitelik için setter/getter metotları sağlar.

SOM'daki bir sınıf FOM'daki bir sınıfın niteliklerinin bir altkümesini temsil ediyor olabilir. Bu durum bir FOM sınıfının farklı formlara girebilmesi anlamına gelir ve HLA nesnelerini programlama dillerindeki nesnelere eşlerken zorluk çıkarır. Bu sıkıntının üstesinden gelmek amacı ile veritabanı sistemlerindeki *View* kavramına benzer bir kavram geliştirdik. FOM'da geçen bir sınıfın her bir formu bir *View*'e karşılık gelir. Bu *View*, Paylaşılan Veri'de bir Java arayüz sınıfı olarak gerçeklenir. Arayüzde tanımlı metotlar, *View* olduğu HLA sınıfına karşılık gelen Gerçekleştirim Sınıfının *setter/getter* metotlarının bir alt kümesini oluşturur. Gerçekleştirim sınıfı bu arayüzlerin hepsini gerçekler. Böylelikle bütün FOM/SOM veri modelini Paylaşılan Veri modeli ile karşılayabiliriz. *View*, Gerçekleştirim sınıfı ve HLA Sınıfı arasındaki ilişki Şekil 1'de gösterilmiştir.



Şekil 1. DNK ile SOM ve FOM HLA Sınıflarının Java Sınıflarına Eşleştirilmesi

Görüleceği üzere Paylaşılan Veri otomatik olarak üretilmeye çok uygun bir yapısı vardır. Bu konu ayrı bir makalede ele alınacaktır.

3.1.2 Dağıtık Nesne Katmanı Bağlantı Arayüzü

DNK'nın Paylaşılan Veri yardımıyla sağladığı nesne tabanlı HLA arayüzüdür. Bu arayüz kullanılarak

- Paylaşılan Veri *View*'ları ile etkileşim ve nitelik bazlı nesne yayınlama, yayın dinleme bildirileri yapılabilir.

```
dolconnection.publishObjectClass(PhysicalEntityView.class);
dolconnection.subscribeInteractionClass(Detonation.class);
```

- Etkileşim gönderilir, nesne tescilleme, güncelleme ve silme yapılır. Nesne güncellemesi sırasında DNK sadece değişen nitelikleri tespit eder ve sadece onlar için nitelik güncelleme çağrısı yapar.

```
PhysicalEntityView pe = dolconnection
        .registerObjectInstace(PhysicalEntityView.class);
pe.setName("T1");
pe.setID(1);
dolconnection.update(pe);

Detonation detonation = new Detonation();
detonation.setTarget(2);
dolconnection.sendInteraction(detonation);
```

- Etkileşimler ve güncellenen, silinen ve yeni tescillenen nesneler sorgulanabilir.

```
Iterator<PhysicalEntityView> iter = dolconnection
        .queryObject(PhysicalEntityView.class, UPDATED);
while (iter.hasNext()) {
    PhysicalEntityView pe = iter.next();
    // ...
}
```

- DNK ayrıca sahibi olunmayan nesnelere güncelleme isteği göndermek için mekanizma sunar. DNK sorgu sonucunda dönen nesne ile değiştirilen nesne arasında değişen nitelikleri tespit eder ve o nitelikleri nesne sahibinin değiştirmesini istemek için özel bir etkileşim gönderir.

```
Iterator<PhysicalEntityView> iter = dolconnection
        .queryObject(PhysicalEntityView.class, UPDATED);
PhysicalEntityView orginal = iter.next();
PhysicalEntityView edited = orginal.clone();
edited.setName("T11");
dolconnection.requestUnownedUpdated(orginal, edited);
```

Bir DNK birden fazla DNK Bağlantısı idame edebilir. Bu bağlantılar birbirlerinden bağımsızdırlar ve birbirlerini etkileyemezler. DNK Bağlantılarının en önemli özelliği, hangi federe üzerinden yapılmış olursa olsun yapılan çağrılara aynı sonuçları döndürmeleridir. Bunu sağlamak için DNK zaman bağımlı ve zaman düzenleyici bir federe üzerinde çalışmalıdır.

DNK sayesinde modelleme sırasında HLA kodu yazmaya gerek kalmaz. Bu sayede HLA ile model kodu otomatik olarak ayrılmış olur. Aynı zamanda kullanımı kolay nesne tabanlı bir arayüz sayesinde geliştirme hızı artar. DNK Bağlantı arayüzündeki sorgulama kabiliyeti Federasyondaki bütün nesneleri ve etkileşimleri döndürecek şekilde çalıştığı için Federe içi veri iletişimi de sağlanmış olur.

3.2 Simülasyon Modülü

Kendine ait DNK Bağlantısına sahip her bir model Simülasyon Modülü (SM) olarak adlandırılır. Her bir SM ayrı bir federe olarak düşünülebilir. SM'ler için geçerli olan tek kısıtlama DNK Bağlantıları dışında kendi aralarında veri iletişimi yapmamalarıdır. Bu kısıtlamayı 3 nolu HLA kuralına benzetebiliriz. Bunun dışında bir SM iç yapısı konusunda tamamen özgürdür.

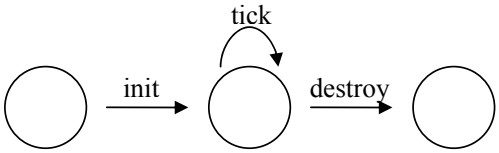
Tüm SM'ler, *init, tick, destroy* olmak üzere 3 metodun gerçekleştirimini sağlarlar.

init sırasında, SM'ye konfigürasyon parametreleri ve DNK Bağlantısı geçilir ve SM bu sırada HLA bildirilerini yapar.

tick sırasında, SM'ler modellemelerini yaparlar. Her bir *tick* bir zaman adımına karşılık gelir ve bütün simülasyon koşumu sırasında art arda çağrılır.

destroy ile SM'ler simülasyonun bitimi sırasında yapmaları gereken işleri yaparlar.

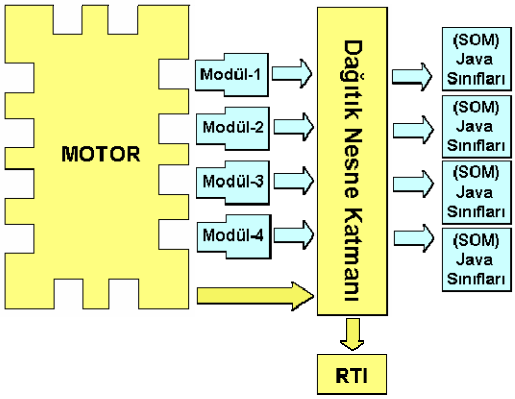
Bir SM'nin bütün yaşam döngüsü Şekil 2'deki gibi özetlenebilir.



Şekil 2. SM yaşam döngüsü

3.3 Simülasyon Motoru

Dağıtık olarak simülasyon koşumunu yönetir. Her bir federenin kendine ait bir Simülasyon Motor'u vardır. Simülasyon koşumunu yönetmek için her bir Simülasyon Motoru o federedeki DNK'yı ve SM'leri diğer Simülasyon Motorları ile koordineli olarak yönetir.



Şekil 3. SIMALT'ın Genel Mimarisi

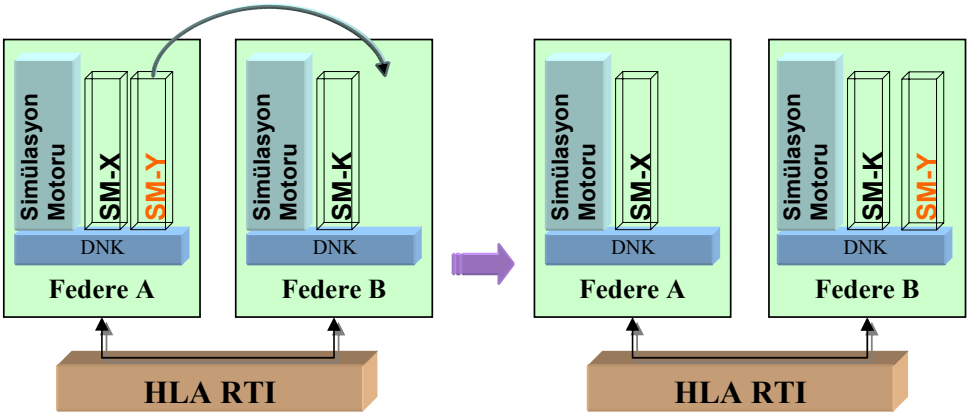
Simülasyon motoru federasyondaki senkronizasyon noktalarını ve bunun sayesinde simülasyon safhalarını idame eder. SM'leri simülasyon motoru oluşturur ve arayüzündeki çağrıları yapar. Simülasyom Motoru bir federedeki zaman adımı içinde bütün SM'leri sıralı çalıştırırken, federeler arasındaki zaman adımlarının uyumlamasını HLA'deki zaman yönetimi (*Time Management*) servisini kullanarak sağlar.

Motorlardan biri yönetici motor olarak atanır. Yönetici Motor diğer motorlardan farklı olarak federasyonun kurulmasından ve senaryonun simülasyonda dağıtılmasından sorumludur. Simülasyon hızını idame eder, dolayısı ile gerçek zaman ile uyumluluğu da sağlamaktan sorumludur.

Motor içinde varsayılan bir senaryo gerçekleştirimi bulunmaktadır. Bu gerçekleştirim hangi SM'lerin hangi federelerde hangi parametrelerle yükleneceği bilgisini taşır. Ancak istenirse bu gerçekleştirim federasyonun ihtiyacına göre kolaylıkla dışarıdan değiştirilebilir.

4. SIMALT'IN ÖLÇEKLENDİRİLEBİLİRLİĞİ VE GENİŞLETİLEBİLİRLİĞİ

Önceden belirttiğimiz gibi DNK Bağlantısı hangi federe üzerinden yapılmış olursa olsun yapılan çağrılara aynı sonuçları döndüreceğini DNK garantiler. Bu da federeler üzerindeki farklı SM dağılımlarının simülasyon sonucunu değiştirmemesini sağlar. Bu noktada seçilen senaryoya göre farklı SM dağılımlarına çalışma zamanında bile karar verme esnekliği kazandırır. Yani basit bir konfigürasyon ayarı ile istersek bütün simülasyonu tek bir federede/görevde çalıştırır, istersek her bir olguyu farklı bir federede çalıştırırız. Bu kabiliyet bize güçlü bir yük-denge mekanizması sunar ve ölçeklendirilebilirlik konusunda çok büyük bir kazanç sağlar.



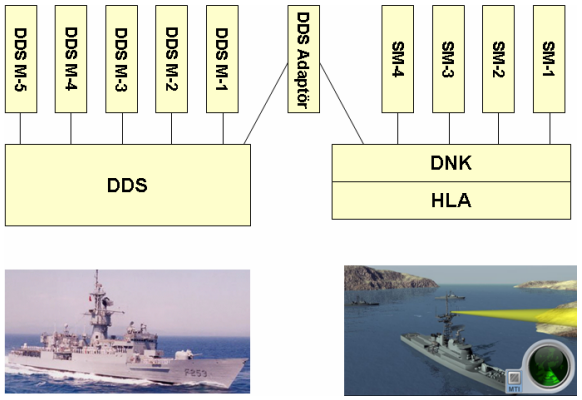
Şekil 4. SM'lerin Dağıtılabilirliği

SIMALT, modüler yapısı sayesinde genişlemeye açıktır. Bu bağlamda günümüzde önemli bir konu olan HLA'nın diğer altyapı standartlarıyla uyumlu hale getirilmesine SIMALT kullanılarak bir örnek verilecektir.

4.1 HLA - DDS Entegrasyonu

Simülasyon sistemlerinin gerçek sistemler ile entegre edilmesi, simülasyonun özellikle eğitim maksatlı olarak kullanımı göz önüne alındığında, kaçınılmazdır. DDS (*Data Distribution Service*) gerçek zamanlı sistemlerde (ve diğer bir çok alanda) çok sık kullanılan bir ara katman standardıdır. İki farklı dünyanın problemlerine çözüm getirmeyi hedefleyen HLA ve DDS standartlarının yayımlama ve üye olma üzerine kurulu benzer yapıları bu iki dünyanın entegrasyonu için de uygun bir ortam oluşturmaktadır. Simülasyon alt yapısı bu entegrasyonu daha da uygulanabilir kılmaktadır.

DDS ile entegrasyon ihtiyacını gidermek için Şekil 5'de gösterilen bağlantı simülasyon alt yapısı üzerinden kolaylıkla kurulabilmektedir. DDS adaptörü uygulamanın ihtiyaçlarına yönelik olarak el ile yazılabilmektedir. Ancak DDS veri modeli ve HLA nesne modelleri arasında ilişkiyi görsel olarak tanımlayan bir araç ile DDS Adaptör'ün yapılandırılması ve aradaki dönüşüm ve uyumlamanın otomatik yapılması uygulanabilir bir çözümdür.



Şekil 5. DDS - HLA entegrasyonu

5. SONUÇ

Bu makalede HLA Uyumlu Yazılım Geliştirme sürecini iyileştirecek bir altyapı uygulaması olan SIMALT'ın adreslediği problemler, alt bileşenleri, ölçeklendirilebilirliği ve genişleyebileceği örnek bir alan anlatılmıştır.

SIMALT, ihtiyaçlara göre sürekli gelişen bir altyapıdır. Mevcut hali ticari bir projede geniş bir geliştirme ekibi tarafından kullanılmış ve başarısı görülmüştür. Gelecekte HLA standardının daha da yayılması ile birçok simülasyon uygulamasına altyapı olacağı düşünülmektedir.

KAYNAKÇA

[1] IEEE High Level Architecture, 1516-2000

[2] POJO, http://en.wikipedia.org/wiki/Plain_Old_Java_Object

[3] Annotations, http://java.sun.com/j2se/1.5.0/docs/guide/language/annotations.html

B4 OTURUMU:

KARAR DESTEK AMAÇLI MODSİM-I

ASKERİ BİRLİKLERİN İNTİKAL PLANLAMASI İÇİN BİR SİMÜLASYON MODELİ
Uğur Ziya YILDIRIM, İhsan SABUNCUOĞLU, Barbaros TANSEL

MUHAREBEDE MÜHİMMAT DAĞITIM SİSTEMİNİN MODELLENMESİ VE OPTİMİZASYONU
Hünkâr TOYOĞLU, Oya Ekin KARAŞAN, Bahar Yetiş KARA

ARAÇ ROTALAMA PROBLEMLERİNİN ÇÖZÜMÜNE BİR YAKLAŞIM
Esin ELDELEKLİ, Kemal LEBLEBİCİOĞLU

TAKTİKSEL EN KISA YOL PROBLEMLERİNİN GENETİK ALGORİTMA YAKLAŞIMI İLE ÇÖZÜMÜ
Ferhat UÇAN, Gülay Büyükaksoy KAPLAN, Reha ÇAPUTÇU, Mehmet HAKLIDIR, Ömer Faruk ARAR

ASKERİ BİRLİKLERİN İNTİKAL PLANLAMASI İÇİN BİR SİMÜLASYON MODELİ[7]

Uğur Ziya YILDIRIM [(a)], İhsan SABUNCUOĞLU [(b)], Barbaros TANSEL [(b)]

[(a)] Genelkurmay Başkanlığı ve Bilkent Üniversitesi, Ankara, uziya@bilkent.edu.tr

[(b)] Bilkent Üniversitesi, Ankara, {sabun, barbaros}@bilkent.edu.tr

ÖZ

Bu makalede askeri birliklerin İntikal Planlama Problemi'ni (İPP) analiz etmek için geliştirdiğimiz bir simülasyon modelini ve bu konuda devam etmekte olan çalışmamızı sunmayı hedefliyoruz. İPP genel olarak askeri birliklerin farklı coğrafi mevkilerdeki kışlalarından çıkıp değişik ulaştırma vasıtaları ile değişik ulaştırma ağlarını (karayolu, havayolu, denizyolu, demiryolu ve iç su yolları) kullanarak ve önceden hazırlanan zaman fasılalı bir intikal planının kısıtları dahilinde hedef bölgelerine konuşlanması olarak ifade edilebilir. Bizim ana katkımız şu anda çoğunlukla manuel olarak hazırlanan intikal planlarının gerçekten planlandığı şekilde icra edilip edilemeyeceği ve muhtemel problem sahalarının ne olacağı konusunda fikir edinmemizi sağlayacak bir simülasyon modeli geliştirmektir. Bu simülasyon modeli, halen Silahlı Kuvvetler tarafından da kullanılan tamamen milli ve Java tabanlı bir coğrafi bilgi sistemi yazılımı olan GeoKIT'in belli modüllerinden istifade eden, nesne-tabanlı, modüler ve esnek bir yapıya sahip, orta detay seviyesi içeren bir kesikli olay simülasyonudur.

Anahtar Kelimeler: Askeri İntikal Planlaması, Coğrafi Bilgi Sistemi (CBS), GeoKIT, Java, Kesikli Olay Simülasyonu, Nesne-Tabanlı Simülasyon.

[7] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

A SIMULATION MODEL FOR DEPLOYMENT PLANNING OF MILITARY UNITS

ABSTRACT

In this article, we present our ongoing work in developing a simulation model to analyze the Deployment Planning Problem (DPP) for military units. DPP may in general be defined as the problem of planning the movement of geographically dispersed military units from their home bases to their final destinations using different transportation assets and a multimodal transportation network (land, air, rail, sea and inland waterways) while obeying the constraints of a time-phased force deployment data prepared in advance. Our main contribution is to develop a simulation model to test the feasibility of currently manually-planned deployment scenarios and help provide insights into potential problem areas. This object-oriented, moderate-resolution and modular discrete event simulation model utilizes some parts of the Application Programming Interface (API) of GeoKIT, a completely national, pure-Java Geographical Information System that is currently used by the Armed Forces.

Keywords: Discrete Event Simulation, Geographical Information Sytem, GeoKIT, Java, Military Deployment Planning, Object-Oriented Simulation.

1. GİRİŞ

Günümüzde artan asimetrik tehditler ve dünya çapındaki terörist faaliyetler barış ve istikrar ortamını tehdit etmeye devam etmektedir. Ülkemizde yaşanma ihtimali hiç de uzak olmayan deprem ve sel baskını gibi doğal afetler neticesinde yardıma ihtiyaç duyan bölgelere yardımın zamanında ulaştırılması Silahlı Kuvvetleri de ilgilendirmektedir. Bu olaylar tüm dünyada olduğu gibi ülkemizde de askeri lojistik destek sistemlerinin ve yapılarının sorgulanmasına yol açmıştır. Eskiden olduğu gibi tehdit beklenen bir bölgeye çok sayıda büyük birliğin yığınak yapması stratejisi artık yerini değişik coğrafi mevkilerde bulunan küçük fakat seyyar ve ateş gücü yüksek birliklerin değişik ulaştırma vasıtalarını kullanarak kısa zamanda görev bölgelerine intikaline bırakmıştır. Barış zamanında muhtemel tehdit ve doğal afet bölgelerine gereken miktarda birliğin gereken zamanda konuşlanması için intikal planları yapılmaktadır. Muhtemel bir kriz veya doğal afet vuku bulduğunda mevcut bu planların aynen veya değiştirilerek kullanılması veya kısa sürede yeni planların yapılması gerekecektir. Ancak bildiğimiz kadarıyla şu anda neredeyse tamamı manuel olarak yapılan bu planların gerçek hayatta istenildiği şekilde yürüyüp yürümeyeceğini, muhtemel problem sahalarının ve bunların çözümlerinin ne olabileceğini öngörecek yeterli detay seviyesinde, ulaştırma altyapısı bilgilerini içeren ve nispeten kısa sürede sonuç üretecek bir simülasyon modeli bulunmamaktadır. Bizi bu çalışmaya yönlendiren ana sebepler bunlardır. Geliştirdiğimiz simülasyon modeli Akgün ve Tansel [1] tarafından İPP için geliştirilen optimizasyon modelinden elde edilen neticelerin rassal şartlar altında denenmesine de olanak sağlayacaktır. Bu bildirinin 2nci bölümünde literatürde bulunan benzer uygulamalar sunulmaktadır. 3ncü bölümde kısaca İPP ortaya konmakta ve ilgili detaylar açıklanmaktadır. 4ncü bölümde İPP için geliştirilen

simülasyon modeli açıklanmakta, kısaca modelin doğrulama ve geçerlemesiyle ilgili devam etmekte olan çalışmalardan bahsedilmektedir. 5nci bölümde sonuçlar ve devam eden çalışmalar sunulmaktadır.

2. LİTERATÜR TARAMASI

Literatürde askeri birliklerin intikallerinin planlaması maksadıyla geliştirilen değişik detay seviyelerinde bilgisayar modelleri ve simülasyonlar bulunmaktadır. Kaynak [2] ve [3] çalışmamızla ilgili olan askeri intikal planlama sistemleri ve stratejik ulaştırma modelleri hakkında detaylı bilgiler sunmaktadır. Kullanılan detay seviyesi ve kullanılma maksatlarına göre bu modelleri iki grup altında toplamak mümkündür. Birincisi farklı harekat alanları arasında (mesela Türkiye'den Afganistan'a) veya bir harekat alanında (mesela Türkiye içinde) birlik intikallerinin planlanmasında kullanılabilecek ve nispeten daha az detaylı modellerdir. Farklı harekat alanları arasında hava ve denizyolu kullanılarak yapılan intikallere stratejik ulaştırma denmektedir. Stratejik ulaştırma planlaması ile ilgili en çok kullanılan iki model MKİS [4] *(Müttefik Konuşlanma ve İntikal Sistemi)* ve BAAS [5] *(Birleşik Akış ve Analiz Sistemi)*'dır. Bir harekat alanı içerisinde birlik intikallerinin planlanmasında kullanılan modellere bir örnek ise HAİLDA [6] *(Harekat Alanı İçi Lojistik Destek Aracı)*'dır. İkinci grupta ise detay seviyesi daha yüksek olan ve birinci gruptaki modellere girdi sağlamak ve veri üretmek gibi maksatlarla da kullanılan modeller bulunmaktadır. Bunlara örnek olarak TLDS [7] *(Taktik Lojistik Dağıtım Sitemi)*, ETMD [8] *(Entegre Tüketim Maddesi Desteği)*, KKMAİD [9] *(Kara Kuvvetleri Muharebe Alanı İhtiyaç Değerlendirmesi)*, LİMSİM [10] *(Liman Simülasyonu)*, ULKAB [11] *(Ulaştırma Kabiliyeti)* ve *(Kolordu Topçusu Mühimmat İkmal Sisteminin Simülasyon ile Değerlendirilmesi)* [12] sayılabilir. Birinci ve ikinci gruptaki intikal planlama ile ilgili model ve sistemlerin kesikli olay simülasyonu olması, coğrafi bilgi sistemi desteği, herhangi bir bilgisayar sisteminde çalışabilme yeteneği ve modellediği ulaştırma ağlarına göre kıyaslamalı gösterimi Çizelge 1'de sunulmuştur.

Simulasyon/ Model	**Havayolu (H), Demiryolu (D), Suyolu (S), Karayolu (K)**	**Platform Bağımsız (Evet / Hayır)**	**CBS Desteği**	**Kesikli Olay Simülasyonu (Evet / Hayır)**	**Diğer Bilgiler**
MKİS [4]	H,S	Hayır	Sınırlı	Hayır	Türkiye dahil birçok ülke tarafından stratejik ulaştırma planlamasında kullanılan bir NATO modelidir.
BAAS [5]	H,S	Hayır	Sınırlı	Hayır	Stratejik ulaştırma planlamasında kullanılan gizli bir ABD modelidir.
HAİLDA [6]	H,D,S,K	Hayır	Sınırlı	Evet	Bir harekat alanı içerisindeki intikallerin planlaması için kullanılan bir ABD modelidir.
TLDS [7]	H,D,S,K	Hayır	Sınırlı	Evet	EXTEND™ and SDI Industry Pro kullanılarak yapılmış ve lojistik dağıtım sistemini modelleyen bir simülasyon.
ETMD [8]	-	Hayır	Yok	Hayır	Onaydan geçmiş harekat planlarına yeterli ikmal desteği sağlanıp sağlanamayacağını test için kullanılan bir ABD modelidir.
KKMAİD [9]	K	Hayır	Yok	Hayır	Zaman fasılalı bir intikal planının hazırlanmasında kullanılan bir editör.
LİMSİM [10]	K	Hayır	Sınırlı	Hayır	Limanlarda yapılan yükleme ve boşaltma ile limandan çıkan birlik miktarlarıyla ilgili bir simülasyon.
ULKAB [11]	K	Hayır	Yok	Evet	Kışlalarda intikal öncesi yapılan yükleme ve hazırlıklar ile ilgili bir simülasyon.
Kor. Top. Müh. İkmal Sis. Sim.ile Değ. [12]	K	Hayır	Yok	Evet	ARENA Simülasyon sistemi kullanılarak yapılmış ve kolordu topçu mühimmatı ikmali sistemini değerlendiren bir simülasyon modeli.

Çizelge 1. Askeri birliklerin intikal planlaması ve lojistik desteği maksadıyla kullanılan bilgisayar modelleri ve simülasyonlar.

Çizelge 1'deki modellerden MKİS NATO planlama faaliyetlerinde kullanılmaktadır. MKİS sadece stratejik ulaştırmanın planlanması maksadıyla kullanılabilmektedir ve karayolu ile demiryolu ulaştırması planlaması yapamamaktadır. Kara ve demiryolu intikalleri mevcut durumda manuel olarak planlanmaktadır. Gerek stratejik ulaştırma ve gerekse bir harekat alanı için yapılan ulaştırma planlarının gerçek hayatta istenildiği şekilde yürüyüp yürümeyeceğini test edecek, muhtemel problem sahalarının ve bunların çözümlerinin ne olabileceğini öngörecek, orta detay seviyesinde ve gerçek ulaştırma

altyapısı bilgilerini içeren simülasyon modelimizin askeri birlik intikallerinin daha iyi planlanmasında gereken bir analitik karar desteği sağlayacağı kanısındayız.

3. PROBLEM VE SİSTEMİN DETAYLARI

İPP genel olarak askeri birliklerin farklı coğrafi mevkilerdeki kışlalarından çıkıp değişik ulaştırma vasıtaları ile değişik ulaştırma ağlarını (karayolu, havayolu, denizyolu, demiryolu ve iç su yolları) kullanarak ve önceden hazırlanan zaman fasılalı bir intikal planının kısıtları dahilinde hedef bölgelerine konuşlanması olarak ifade edilebilir. İntikal bir ülke sınırları dahilinde olabileceği gibi ülkeler arasında da olabilir. Bu durumda stratejik ulaştırma adını alır. İntikal eden unsurlar kışlalarından çıktıktan sonra doğrudan hedef bölgelerine karayolu ile intikal edebilecekleri gibi, belli transfer noktalarında (limanlar, havaalanları ve tren istasyonları) farklı ulaştırma vasıtalarına aktarılarak başka ulaştırma ağlarını da kullanabilirler. Bu ulaştırma vasıtaları ve ağları ihtiyaç duyulduğu miktarda ve birbiri ardına kullanılabilir. Ancak bir intikal esnasında yapılan aktarma sayısının mümkün olduğu kadar az olmasının intikali kolaylaştıracağı unutulmamalıdır. Yüklemenin yapıldığı transfer noktalarına Yükleme Noktası (YN) ve indirmenin yapıldığı transfer noktalarına ise İndirme Noktası (İN) adını verebiliriz. Yükleme Noktalarını Hava Yükleme Noktası (HYN), Deniz Yükleme Noktası (DYN) ve Tren Yükleme Noktası (TYN) diye detaylandırabiliriz. Benzer şekilde İndirme Noktalarını da Hava İndirme Noktası (HİN), Deniz İndirme Noktası (DİN) ve Tren İndirme Noktası (TİN) diye detaylandırabiliriz. Birlikler YN ve İN içerisinde sınırlı kapasiteye sahip bekleme veya park etme bölgelerinden istifade ederler.

Kışlalarından çıkan birliklerin doğrudan hedef bölgelerine veya transfer noktalarına giderken konvoylar halinde intikal etmektedir. Konvoyun büyüklüğü harekatla ilgili hususlar ve taktik mülahazalara göre değişmektedir. Birlik intikalleri genelde öncü (konakçı) grubu, personel grubu ve kargo grubu olacak şekilde üç grup halinde yapılmaktadır. Bu farklı grupların kışlalarından çıkış ve hedef bölgelerine varışlarının senkronizasyonu mevcut taşıma unsurları ile kapasiteleri (ağırlık, hacim, koltuk sayısı, uzunluk), ulaştırma ağının durumu, harekatla ilgili kısıtlar ve tehdit durumu gibi faktörlere bağlıdır [1]. Bir birlik ihtiyaca göre kendi ulaştırma vasıtalarına ilave olarak sivil sektörden veya başka ülkelerden kiralanan ulaştırma vasıtalarından da istifade edebilir. Kriz zamanlarında bir an önce hedef bölgelerine ulaşmak önemliyken barış zamanında maliyet önemli bir unsurdur. Barış zamanında muhtemel tehdit ve doğal afet bölgelerine gereken miktarda birliğin gereken zamanda konuşlanması için intikal planları yapılmaktadır. Muhtemel bir kriz veya doğal afet vuku bulduğunda mevcut bu planların aynen veya değiştirilerek kullanılması veya kısa sürede yeni planların yapılması gerekecektir. Barış veya kriz zamanında olsun her intikal önceden hazırlanan zaman fasılalı bir intikal planının kısıtları dahilinde yapılmaktadır. Bu kısıtlar en azından birliklerin hangi unsurlarının hangi ulaştırma vasıtalarını ve ağlarını kullanacağını, en erken ve en geç hedef bölgesinde bulunma zamanlarını içermektedir.

4. SİMÜLASYON MODELİ

İPP'nin simülasyon modelini oluşturmak için ilk defa Schruben [13] tarafından ortaya konan ve Sargent [14] ile Buss [15] tarafından geliştirilen Olay Grafikleri (OG, *Event Graphs*) metodolojisini kullandık. OG metodolojisinde düğümler simülasyondaki olayları ve bunlar arasındaki yönlü ayrıtlar bir olay tarafından simülasyonun gelecek

olay listesine başka bir olayın yerleştirilmesini temsil etmekte kullanılır. Ayrıtın kesikli olması ise bu ayrıtın çıktığı düğümün temsil ettiği olayın ayrıtın diğer ucundaki olayı simülasyonun gelecek olay listesinden çıkartmasını temsil eder. OG simülasyon modelimizin kavramsal ve mantıksal modelini oluşturmakta kullanılmıştır.

Simülasyon modelimizin parçalarını modüler ve esnek bir yapı ile birbirleriyle ilişkilendirmek için Dinleyen Olay Grafiği Objeleri (DOGO, *Listener Event Graph Objects*) [16] altyapısını kullandık. Bu yapıda simülasyondaki objeler ilgilendikleri diğer objelere kendilerini *dinleyici* olarak kaydederler ve sadece ilgi duydukları durum değişkenlerinde meydana gelen değişikliklere istinaden belirledikleri uygun faaliyetleri yaparlar. *Dinlenen* obje ise bundan etkilenmez. Simülasyon modülleri arasındaki bu esnek *dinleyen* ve *dinlenen* ilişkisi bir doktorun steteskopuna benzer bir şekille ifade edilir. Steteskopun üçgene benzer ucundaki obje *dinlenen* objedir (Şekil 1).

OG ve DOGO altyapısı ile geliştirilen kesikli olay simülasyonları bir Java kütüphanesi (API) olan *Simkit* kullanılarak programlanabilir. *Simkit* ABD Deniz Kuvvetleri Yüksek Lisans Okulu'nda geliştirilmiş, 35'den fazla yüksek lisans tez çalışmasında kullanılmış ve ABD Kara Kuvvetlerinin gelecek nesil kara muharebe simülasyonu olan COMBAT XXI'in simülasyon motoru olarak seçilmiştir. *Simkit* http://diana.nps.edu/Simkit/ adresinden indirilebilir. DOGO'nun *Simkit*'teki karşılığı Simülasyon Olay Dinleme (SOD, SimEventListener pattern) altyapısıdır.

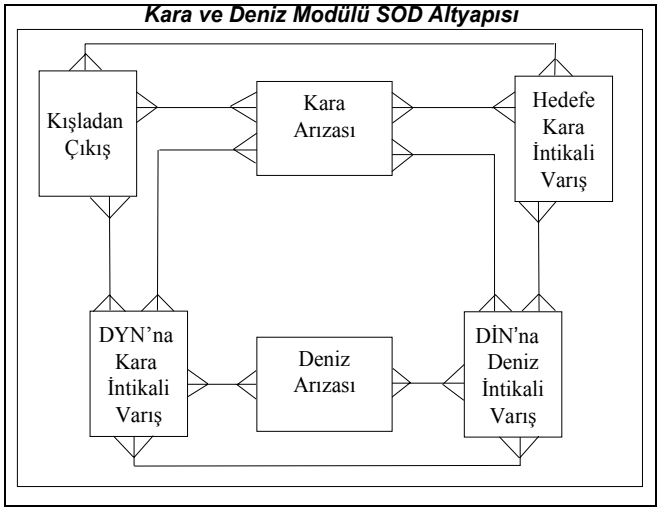
Gerçek ulaştırma altyapısına ait bilgileri (kara ve demiryolları, şehirler, köprüler ve su yolları v.b.) kullanmak ve animasyon desteği sağlamak maksadıyla tamamen milli, lisanslı, Kara Kuvvetleri Komutanlığı ve Genelkurmay Başkanlığı tarafından da kullanılan ve tamamen Java ile yazılmış bir CBS olan GeoKIT'in modüllerinden istifade ettik.

Yukarıda kısaca açıkladığımız araçları kullanarak geliştirdiğimiz simülasyon modeli üç ana unsurdan oluşmaktadır; *Grafik ara yüzü*, *şebeke modülü* ve *ana simülasyon modülü*. Grafik ara yüzü gerçek ulaştırma ağına ait coğrafi veriyi ve simülasyondaki varlıkları ekranda göstermede, ve fareyi kullanarak ekranda bir ulaştırma planına ait formları doldurarak bir senaryoyu yaratmak için kullanılmaktadır. Tüm CBS verisi WGS84'ten UTM formatına dönüştürülebilmekte ve senaryolar XML formatında kaydedilip tekrar yüklenebilmektedir. *Şebeke modülü* senaryo tanımlarken ve intikal güzergahlarını belirlerken karayolu veya demiryolu şebekeleri üzerinde mesafece en kısa yolları ve en kısa yollardan kullanıcı tarafından girilen belli bir faktör kadar uzun olan yolları listeleyerek senaryo yaratıcısına kolaylık sağlamaktadır. *Ana simülasyon modülü* kara, deniz, hava ve demiryolu intikalleri için hazırlanan 4 alt modülden oluşmaktadır. Bunlar da 3'er alt modülden oluşmaktadır. Şu anda simülasyonun tamamı için her biri ortalama 150 satır uzunlukta olan 84 adet Java sınıfı yazılmış bulunmaktadır. Simülasyona ilave yetenekler eklendikçe bu miktarın artacağı değerlendirilmektedir.

*Ana simülasyon modülü*ndeki alt unsurların (kara, deniz, hava ve demiryolu) birbiriyle irtibatı DOGO ve SOD altyapısını kullanarak gerçekleştirilmiştir.

4.1. Kara ve Deniz Modülleri İçin Simülasyon Olay Dinleme Altyapısı

Kara ve deniz modüllerinin ve onların alt modüllerinin birbirinden haberdar olması Şekil 1'de gösterilen *Kara ve Deniz Modülü SOD Altyapısı* ile sağlanmaktadır.



Şekil 1. Kara ve Deniz Modülü SOD Altyapısı

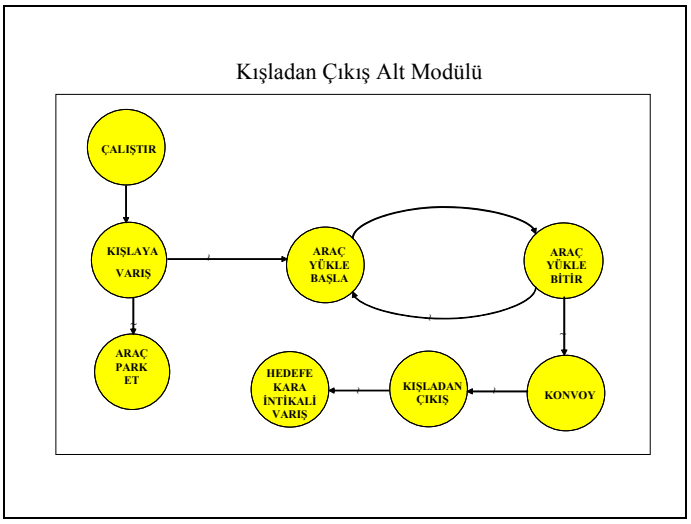
Kışlasından çıkan bir birlik karayoluyla doğrudan hedef bölgesine intikal edebilir veya DYN'a giderek unsurlarını gemiye yükler ve denizyoluyla DİN'a gidip unsurlarını indirdikten sonra karayoluyla hedef bölgesine intikal edebilir. Kara intikali her kullanıldığında intikal vasıtalarında arızalar meydana gelebilir veya unsurlar bir trafik kazasına karışabilir. Bu sebeplerden meydana gelebilecek gecikmeler *Kara Arızası* alt modülünde modellenmektedir. Kışladan Çıkış, DYN'dan Kışlaya ve DİN'dan hedef bölgesine yapılan tüm kara intikalleri *Kara Arızası* alt modülü tarafından dinlenmektedir. Bir arıza veya trafik kazası neticesinde meydana gelebilecek gecikmeler hesaplanarak birliğin gideceği yere varış zamanı yeniden hesaplanmaktadır. Benzer durum *Deniz Arızası* alt modülü ile *DYN'na Kara İntikali Varış* ve *DİN'na Deniz İntikali Varış* alt modülleri için de geçerlidir. Her alt modülün birbirini dinleyebilmesi için ikisinde de aynı isimde ve aynı parametrelere sahip olaylar bulunmalıdır. Kara ve Hava Modülü SOD Altyapısı ile Kara ve Demiryolu Modülü SOD Altyapısı da aynı mantıkla simülasyonun alt modüllerinin birbiriyle irtibatını sağlar.

Makalede sayfa kısıtlaması nedeniyle sadece Kara modülüne ait 3 en basit alt modülün OG aşağıda açıklanacaktır. Burada sunulan OG basitleştirilmiştir. Ayrıtlar üzerindeki zaman gecikmeleri, eğer şartları ve modüle ait durum değişkenleri ile parametreler sadelik sağlamak maksadıyla gösterilmemiştir.

4.2. Kışladan Çıkış Alt Modülü

Kışladan Çıkış alt modülü birliklerin gerekli yüklemelerini yaparak doğrudan karayolu ile hedef bölgelerine veya başka bir ulaştırma vasıtasına bindirme yapmak maksadıyla bir transfer noktasına konvoy halinde intikalini modeller. Ulaştırma vasıtaları birliği bir defa da taşıyamadığında gidilen yerden tekrar dönmek ve malzeme ile personel taşımak zorunluluğu da modellenmiştir. Şekil 2'deki Olay Grafiğine transfer noktalarına

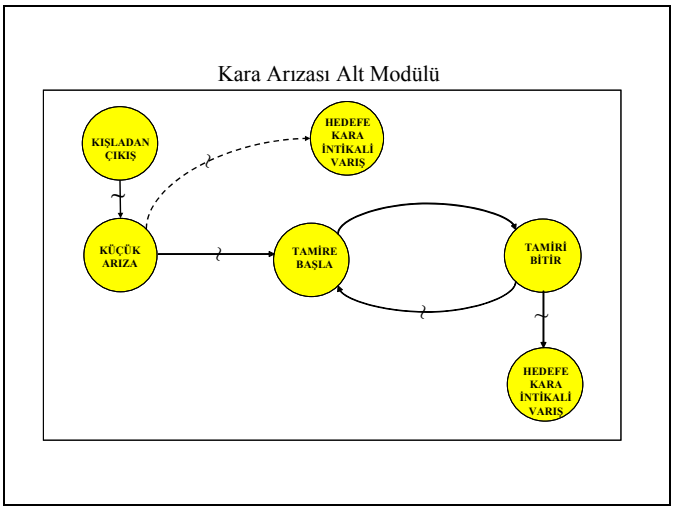
intikalle ilgili olaylar sadelik maksadıyla dahil edilmemiştir. Çalıştır olayı simülasyona ait durum değişkenlerinin sıfırlanması ve birliklerin ilk konuş noktalarına atanmasında kullanılmaktadır.



Şekil 2. Kışladan Çıkış Alt Modülü Olay Grafiği

4.3. Kara Arızası Alt Modülü

Kara Arızası alt modülü Şekil 3'dedir.

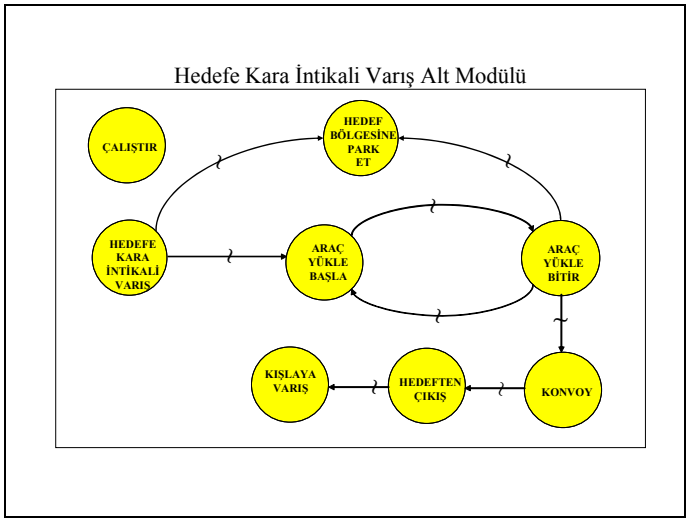


Şekil 3. Kara Arızası Alt Modülü Olay Grafiği

Karayolu ile intikal eden her araçta küçük, orta ve büyük çapta arızalar meydana gelebilir. Şekil 3 sadelik maksadıyla sadece kışlasından çıkıp doğrudan hedef bölgesine giden kara vasıtalarında meydana gelebilecek küçük çaplı arızaların yol açabileceği gecikmeyi modellemektedir. Şekil 3 de belirtilen hesaplamaları yapmak için ihtiyaç duyulan her aracın bozulma olasılık dağılımları ile tamir sürelerinin dağılımlarına ilave olarak, Türkiye'nin karayollarında trafik kazalarının sık meydana geldiği "kara noktalar" Kara Yolları Genel Müdürlüğü web sayfasından alınarak, http://www.kgm.gov.tr/asps/trafik/karanokta.htm, CBS verisine dahil edilmiştir ve kullanılmaktadır.

4.4. Hedefe Kara İntikali Varış Alt Modülü

Hedefe Kara İntikali Varış alt modülü Şekil 4'dedir.



Şekil 4. Hedefe Kara İntikali Varış Alt Modülü Olay Grafiği

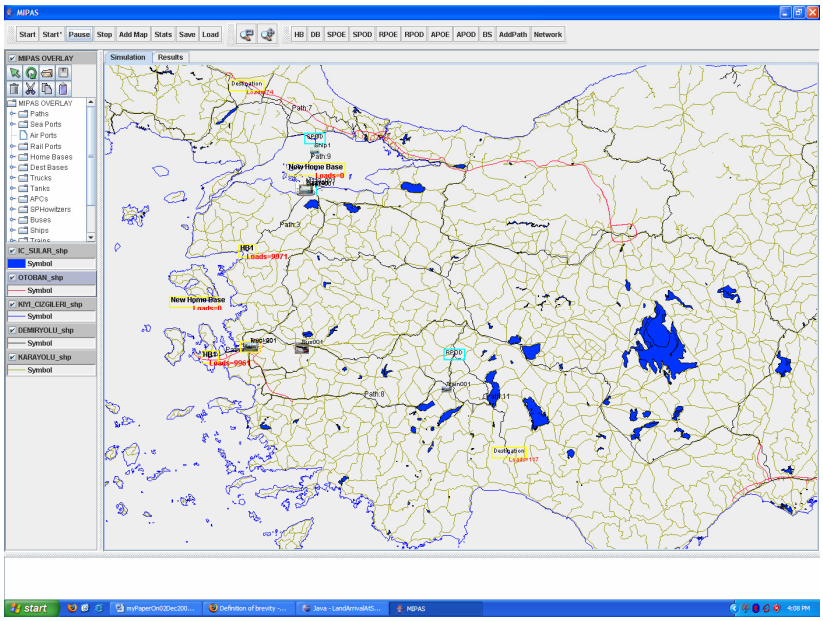
Bu alt modül konvoyun gitmesi gerek hedef bölgesine varışını, varsa yük ve personelini boşaltmasını, eğer ilave yük veya personel taşınması gerekiyorsa geldiği transfer noktası veya kışlaya konvoy halinde dönmesini, gerekmiyorsa da hedef bölgesinde park ederek beklemesini modeller. Şekil 3'de sadelik maksadıyla sadece kışlaya geri dönüş modellenip transfer noktalarına muhtemel dönüşler dahil edilmemiştir.

Daha karmaşık olan Deniz, Hava ve Demiryolu modüllerinin alt modüllerine ait OG sayfa kısıtlaması sebebiyle burada sunulamamıştır.

4.5. Animasyon

Simülasyona ait animasyonu sağlamak maksadıyla *PingThread* sınıfı tarafından deterministik zaman aralıklarıyla simülasyonun gelecek olay listesi içerisine bir *Ping* olayı dahil edilmektedir. *Simulation* sınıfı bu olayı dinlemekte ve duyduğunda varlıkların mevkilerini güncellemektedir. GUIMain sınıfı ise simülasyon varlıklarına ait şekilleri ekranda tekrar boyayarak güncellemektedir. Kesikli olay simülasyonlarında

basit hareket ve animasyon hakkında daha detaylı bilgi için [16]'ya bakınız. Bu uygulamanın basit bir görüntüsü Şekil 5'dedir. Burada Türkiye haritası üzerindeki karayolları, demiryolu ve otobanlar ile küçük bir senaryoya ait bilgiler gösterilmektedir.



Şekil 5. Küçük Bir Senaryo ile Karayolu, Demiryolu ve Otobanlar

4.5. Doğrulama ve Geçerleme

Simülasyon modelinin doğrulama ve geçerleme faaliyetleri [17]'de belirtilen bazı yöntemler dahilinde devam etmektedir. [1]'deki İPP'nin optimizasyon modeline ait 6 determistik senaryonun çözümleri ile simülasyon deterministik değerlerle çalıştırılarak elde edilen değerler doğrulama maksadıyla karşılaştırılmaktadır. Geçerleme maksadıyla Genelkurmay Bilimsel Karar Destek Merkezi'nden temin edilen geçmiş intikallere ait verilerle simülasyondan elde edilen stokastik sonuçlar mukayese edilecektir.

5. SONUÇLAR VE DEVAM EDEN ÇALIŞMALAR

Bu bildiride, hazırlanan intikal planlarının gerçekten planlandığı şekilde icra edilip edilemeyeceği ve muhtemel problem sahalarının ne olacağı konusunda fikir edinmemizi sağlayacak, karar verici makamlara uygun alternatifler sunmaya yardımcı olacak bir simülasyon modeli geliştirme çalışmalarımızdan bahsettik. Bu modelin orta detay seviyesinde kısa sürede analiz yapmaya imkan sağlayarak bu konuda büyük bir eksikliği kapatacağına inanıyoruz. Tugay seviyesinde bir birliğin hazır olan intikal planı simülasyon modelimize girilmiştir. Doğrulama ve geçerleme faaliyetleri güven verici bir seviyeye ulaştıktan sonra ilk etapta bu intikal planının analizi yapılacaktır. Simülasyon deneylerinde girdi ve çıktı değişkeni sayılarının fazlalığı bizi klasik deney tasarımlarından ziyade Latin Hiperküp Örneklemesi yöntemini kullanmaya yönlendirmiştir. Bu konuda detaylı bilgi ve ilave kaynaklar [18]'de bulunmaktadır.

KAYNAKÇA

[1] Akgün, İbrahim and Tansel, Barbaros., "Optimization of Transportation Requirements in the Deployment of Military Units". *Computers and Operations Research*, Volume 34, Issue 4, Nisan 2007, pages 1158 – 1176.

[2] Boukhtouta, A., Bedrouni, A., Berger, J., Bouak, F., and Guitouni, A. "A Survey of Military Planning Systems". Aşağıdaki adreste http://www.dodccrp.org/events/9th_ICCRTS/CD/papers/096.pdf ve 15 Aralık 2006'da ulaşılmıştır.

[3] McKinzie, Kaye and Barnes, Wesley J. "A Review of Strategic Mobility Models Supporting the Defense Transportation System". Aşağıdaki adreste www.me.utexas.edu/~barnes/research/files/update_06_2003/RSMMSDTS.pdf ve 15 Aralık 2006'da ulaşılmıştır.

[4] Heal, G.N., Garnett, I.K., "Allied Deployment and Movement System (ADAMS) Version 3.0 Tutorial", Technical Note 669, NATO Consultation, Command and Control Agency, The Hague, Eylül 2001.

[5] JFAST 8.1 Handbook. "Joint Flow and Analysis System for Transportation". http://www.jfast.org. Kasım 15, 2000.

[6] Groningen, C.N.V., Blachowicz, D., Braun, M.D., Love, R.J., Simunich, K.L. and Widing, M.A. "Modeling Military Deployment in Theaters of Operations – Balancing Deployment Alternatives". Aşağıdaki adreste http://www.dis.anl.gov/temps/LogisticsSpectrum.ELIST.MergedFigures.8Aug05.pdf ve 15 Aralık 2006'da ulaşılmıştır.

[7] Krause, L.C. and Parsons, David J. "Tactical Logistics and Distribution System (TLOADS) Simulation". Proceedings of the 1999 Winter Simulation Conference.

[8] Gunshenan, J., Kratkiewicz, G., and Cashon, R. "Integrated Consumable Item Support (ICIS) Model". Bu adreste http://ms.ie.org/SIW_LOG/97S/97S-SIW-170.doc ve 15 Aralık 2006'da ulaşılmıştır.

[9] SABRE: Logistics Federates Homepage: http://ms.ie.org/SIW_LOG/Log_Federates/FederateTable.shtml. En son 15 Aralık 2006'da ulaşılmıştır.

[10] PORTSIM *(The Port Simulation)*. Aşağıdaki adreste http://chl.erdc.usace.army.mil/%5CMedia/5/9%5CMTMC%20TEA%20Models.ppt ve 15 Aralık 2006'da ulaşılmıştır.

[11] Burke, J. F., Love, R.J., Macal, C.M., Howard, D.L. and Jackson, J. "Modeling Force Deployment from Army Installations Using the Transportation System Capability (TRANSCAP) Model". Aşağıdaki adreste http://www.osti.gov/bridge/servlets/purl/757530-BfenI/webviewable/757530.pdf ve 15 Aralık 2006'da ulaşılmıştır.

MUHAREBEDE MÜHİMMAT DAĞITIM SİSTEMİNİN MODELLENMESİ VE OPTİMİZASYONU[8]

Hünkâr TOYOĞLU [(a,b)], Oya Ekin KARAŞAN [(b)], Bahar Yetiş KARA [(b)]

[(a)] Kara Kuvvetleri Komutanlığı Karar Destek Şubesi, Ankara 06100, hunkar@bilkent.edu.tr

[(b)] Bilkent Üniversitesi Endüstri Mühendisliği Bölümü, Ankara 06800, {karasan,kara}@bilkent.edu.tr

ÖZ

Bu çalışmanın esas amacı Kara Kuvvetleri Komutanlığı (K.K.K.lığı)'nın muharebe alanında mühimmat dağıtımı planlamasına yardımcı olacak bir karar destek aracı geliştirmektir. Geniş bir literatür taramasından sonra mevcut modellerin ihtiyacımız olan modelin özelliklerini tam karşılamadıkları ve amacımıza yönelik kullanılamayacakları gösterilmiştir. Bu sebeple, ürün akışına dayanan üç kademeli yeni bir model önerilmiştir. Bu model (1) birden fazla ürün dağıtmakta, (2) esnek olmayan zaman pencerelerini dikkate almakta, (3) ihtiyaç noktalarına birden fazla araç veya depo ile hizmet edilmesine izin vermekte ve (4) yer seçiminin, iki ayrı kademede yapılabilmesine imkân sağlamaktadır. Farklı geçerli eşitsizlikler geliştirilmiş ve çözüm süresi kısaltılmıştır. Son olarak, gerçek bir senaryoda modelin performansı gösterilmiş ve umut verici sonuçlar elde edilmiştir.

Anahtar Kelimeler: Ağ Tasarımı, Lojistik, Dağıtım, Yer Seçimi-Yol Atama Problemi, Askeri Uygulama.

MODELING AND OPTIMIZATION OF AMMUNITION DISTRIBUTION SYSTEM IN BATTLE

ABSTRACT

Main objective of this study is to develop a decision support tool to help Land Forces Command in planning the distribution of ammunition on the battle field. We demonstrate after extensive literature review that the existing models can not be used for our purposes for they do not consider the specifics of the problem in this study. Thus, we propose a novel three-layer commodity-flow location routing formulation that (1) distributes multiple products, (2) respects hard time windows, (3) allows demand points to be supplied by more than one vehicle/depot and (4) decides the location of facilities at two different layers. We derive several valid inequalities and speed up the solution time. Finally, we illustrate the performance of the model in a real life scenario and encouraging results are obtained.

[8] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

Keywords: Network Design, Logistics, Distribution, Location Routing Problem, Military Application.

1. GİRİŞ VE MOTİVASYON

K.K.K.lığı soğuk savaş döneminde tümen teşkilatına dayanan ve çok sayıda ağır silahlara sahip büyük ordular halinde teşkilatlanmıştır. Soğuk savaşın sona ermesi ile birlikte K.K.K.lığı daha küçük, çevik ve hareketli bir kuvvet yapısını hedeflemiştir. Bu kuvvet yapısı genellikle daha küçük ve hafif olan tugaylardan oluşmaktadır. Böyle bir kuvvet sadece ateş gücüne dayanmayıp aynı zamanda hareket kabiliyetine de dayanmaktadır. Bu şekilde hızlı hareket eden bir kuvveti desteklemek için hareket kabiliyeti yüksek ve etkin bir mühimmat dağıtım sistemine ihtiyaç duyulmaktadır.

K.K.K.lığına etkin ve esnek bir ikmal sistemi sağlamak maksadıyla, bu çalışmada Mobil Mühimmat İkmal Sistemi (MMİS) önerilmektedir. Önerilen sistemde esas amaç mühimmatı birliklerin mümkün olduğunca yakınına götürmektir ve bu maksatla Sabit Transfer Nokta (STN)'ları ve Mobil Transfer Nokta (MTN)'ları kullanılmaktadır. STN'ları demiryolunun bittiği veya demiryolu üzerinde mühimmatın en ileriye kadar götürülebildiği noktalardır. Mühimmat bu noktalarda trenlerden tırlara aktarılmaktadır. MTN'ları genellikle mühimmat araçlarının veya stoklarının daha ileride bulunan birliklere intikal ettirilmeden önce kısa süreli olarak bekletildikleri ileri ikmal noktalarıdır. Mühimmat bu noktalarda tırlardan mühimmat araçlarına aktarılmaktadır.

2. PROBLEMİN TANIMI VE LİTERATÜR TARAMASI

2.1. Problemin Tanımı

MMİS sürekli bir bütünleme ve itme sistemidir. Üretilen veya tedarik edilen mühimmat öncelikle birinci seviye depolar tarafından teslim alınmaktadır. Mühimmat daha sonra demiryolu ile mümkün olduğunca ileriye intikal ettirilmektedir. Mühimmatın demiryolu ile intikal ettirilebileceği noktalar, muhtemel STN'ları olarak tanımlanmıştır.

Bu çalışmada birinci seviye depolar ile STN'ları arasındaki mühimmat akışı analiz edilmemiştir. İhtiyaç duyulan miktarda mühimmatın STN'larına demiryolu vasıtası ile zamanında taşınabileceği farz ve kabul edilmiştir. Mühimmat STN'larından MTN'larına tır veya kamyonlar ile karayollarından taşınmaktadır. Müteakiben MTN'larından mühimmat araçları ile birliklere dağıtılmaktadır.

STN'larının, MTN'larının ve birliklerin bulunduğu bir muharebe sahası tahayyül edilerek, MMİS'ni modellemek maksadıyla birliklerin bulunduğu konumlar bir anlığına dondurulup muharebe sahasının resmi çekilmiştir. Bu durumda çözülmesi gereken problem, birlikleri doğru nitelik ve miktarda mühimmatla istenilen yer ve zamanda ikmal etmektir. Bunu gerçekleştirmek maksadıyla şu planlama faktörleri dikkate alınmalıdır; (a) açılacak STN'larının ve MTN'larının sayısı ve yerleri, (b) STN'larından MTN'ları vasıtası ile birliklere kadar mühimmat taşıyacak araçların güzergâhları ve zaman çizelgeleri.

Söz konusu bu iki problemi ayrı ayrı çözmek alt-eniyi (sub-optimal) çözümlere imkân verebilmektedir [1]. Dolayısı ile bu kararlar aynı anda alınmak zorundadır. Bu sebeple yerleşim, yol atama ve çizelgeleme problemlerini ihtiva eden MMİS tasarım problemi bir yer seçimi-yol atama problemidir (Location Routing Problem, LRP).

2.2. Literatür Taraması

LRP stratejik bir karar olan yerleşim ve operasyonel bir karar olan yol atama/çizelgeleme konularını ele almakta ve birden fazla ihtiyaç noktasına hizmet eden tesislerin en iyi miktar, kapasite ve yerleşimlerinin bulunması ile araçların en iyi yol atama ve çizelgeleme problemini çözmektedir [2]. MMİS kademeler halinde temsil edilecek olursa [3], STN'ları, MTN'ları ve birlikler sırası ile birinci, ikinci ve üçüncü kademelerde bulunmaktadırlar. Ayrıca bir araç turu, o araç ile ikmal yapılan birden çok STN'sı, MTN'sı ve birliği ihtiva etmektedir.

LRP literatürünü tarayan bazı çalışmalar bulunmaktadır [3,4]. 1998 yılında yapılan son çalışmada 33 adet makale önce 11 adet problem kriterine göre ve müteakiben çözüm metotlarına göre (kesin veya yaklaşık çözümler) sınıflandırılmıştır [2]. Bu çalışmada söz konusu 11 kriter revize edilmiş, 6 ekleme yapılmış ve 52 adet makale 17 kritere göre sınıflandırılmıştır. Yapılan sınıflandırmanın özeti Çizelge 1'de görülmektedir.

Çizelge 1. Literatürün özeti

1. Hiyerarşik seviye		7. Tesis kademesi		13. Yer seçim kararı	
a. Teslim etme veya alma	50	a. İki	41	a. Bir kademede	50
b. Teslim etme ve alma	2	b. Üç/Dört	11	b. İki kademede	2
2. İhtiyaç		8. Planlama periyodu		14. Ürün sayısı	
a. Rassal değil	40	a. Tek	48	a. Tek	49
b. Rassal	12	b. Çok	5	b. Çok	3
3. Tesis sayısı		9. Zaman sınırlaması		15. İkmal	
a. Tek	11	a. Yok	43	a. Tekli	52
b. Çok	41	b. Gevsek	9	b. Çoklu	0
4. Araç filosu		c. Kati	0	16. Stok	
a. Tek	8	10. Amaç fonksiyonu		a. Var	2
b. Homojen	36	a. Tekli	46	b. Yok	50
c. Heterojen	8	b. Çoklu	7	17. Çözüm yöntemi	
5. Araç kapasitesi		11. Veri		a. Kesin	20
a. Var	31	Gerçek	16	b. Yaklaşık	34
b. Yok	21	Gerçek değil	36		
6. Tesis kapasitesi		12. Çözüm uzayı			
a. Var	23	a. Sürekli	2		
b. Yok	29	b. Ayrık	50		

Literatürde kapasite tanımlanmamış homojen araç filosu kullanılırken bu çalışmada kapasite tanımlanmış heterojen araç filosu kullanılmaktadır. Literatürde mevcut modellerin çoğunluğunda tesislere kapasite tanımlanmamıştır. Ancak bu çalışmada tesislerin kapasiteleri sınırlıdır. Literatürde yer alan LRP modelleri genelde iki kademeli iken MMİS'de üç kademe bulunmaktadır. Mevcut çalışmalardan sadece dokuzu gevşek zaman kısıtlaması içermekte, hiçbir çalışmada iki taraflı zaman penceresi bulunmamaktadır. Bu çalışmada iki taraflı zaman pencereleri bulunmaktadır. Makalelerin çoğunda sadece bir kademede yerleşim kararı verilmektedir. MMİS'de iki farklı kademede tesis yerleştirilmektedir. Sadece üç makale çoklu ürün dağıtmakta diğer tüm çalışmalar tek ürün içermektedir. Son olarak, büyük ikmal sistemlerinde daha iyi çözümlere ulaştırabilecek çoklu ikmal sistemini hiçbir çalışma içermemekte, tüm modeller tekli ikmal sistemini zorlamaktadır.

Yapılan literatür araştırması, bu çalışmada önerilen modelin bazı özelliklerinin literatürde yeteri kadar göz önüne alınmadığını göstermektedir. Bu çalışma, doğrudan söz konusu özelliklerin tek bir modelde toplanmasına yöneliktir. Yazarların bilgisine göre bu makale böylesine kapsamlı bir model geliştirmek için yapılan ilk çalışmadır.

3. MODELİN GELİŞTİRİLMESİ

3.1. Notasyon

Muharebe sahası üç çeşit düğümün (node) (muhtemel STN'ları / MTN'ları yerleri ve birliklerin bilinen yerleri) bulunduğu bir ağ şeklinde modellenmiştir. Bu benzetimle birlikte MMİS yönlü ve bağlı bir ağ olarak kabul edilmiştir. Bu ağ $G=(N,E)$ şeklinde temsil edilmektedir ve N düğümleri, E düğümler arasındaki yolları temsil etmektedir. N ayrışık üç adet altkümeye ayrılmıştır, $N=N_F \cup N_M \cup N_C$. N_F muhtemel STN'sı, N_M muhtemel MTN'sı ve N_C ise birliklerin yerlerini göstermektedir. Ayrıca $N_{FM}=N_F \cup N_M$ ve $N_{MC}=N_M \cup N_C$. E iki ayrı yol ağından oluşmaktadır, $E=E_1 \cup E_2$. E_1, üzerinde tır/kamyonların gidebileceği çift yönlü karayollarıdır. Bu yollar üzerinde tır/kamyonlar (a) STN'ları ve MTN'ları arasında ve (b) sadece MTN'ları arasında hareket edebilmektedir. E_2, mühimmat araçlarının gidebileceği arazi üzerindeki yollardır ve bu yollar üzerinde mühimmat araçları (a) MTN'ları ve birlikler arasında ve (b) sadece birlikler arasında hareket edebilmektedir. V iki alt kümeden oluşan araç setidir, $V=V_F \cup V_M$. V_F tır/kamyon alt kümesi ve V_M mühimmat aracı alt kümesidir. P mühimmat çeşidi setidir. $D=[D_{ij}]$ uzaklık matrisidir ve D_{ij} düğüm i ve j arasındaki uzaklıktır. Bu uzaklık düğüm i ile j arasındaki en kısa yoldur ve bu yollar bilinmektedir. Ayrıca G üzerindeki tüm ulaşımın en kısa yollar üzerinde ve sabit hızla (*speed*) yapıldığı farz ve kabul edilmiştir. $T=[T_{ij}]$ ulaşım zamanı matrisidir ve $T_{ij} = D_{ij}$ /*speed* düğüm i ile j arasındaki ulaşım zamanıdır.

Bu makalede MMİS tasarım problemi için dört indisli bir karışık tamsayılı programlama modeli sunulmuş ve takip eden parametreler kullanılmıştır. Q^p_i birlik i'nin mühimmat p için olan ihtiyacıdır. İnsan gücü, arazi, tehdit, yangın, vb. hususlardan dolayı her transfer noktası i'nin her bir mühimmat çeşidi p için bir kapasitesi bulunmaktadır ve CD^p_i olarak temsil edilmektedir. CV^p_v araç v'nin mühimmat p için kapasitesidir ve CT_v ise aracın toplam kapasitesidir. Yukarıda açıklandığı şekilde her bir mühimmat çeşidi için ve her bir birlik için en erken ve en geç ikmal zamanı vardır ve bunlar TE^p_i ve TL^p_i olarak temsil edilmişlerdir. TM^p mühimmat p'nin birlikler arasındaki maksimum en geç ikmal zamanıdır, $TM^p=\max_{i \in Nc}\{TL^p_i\}$. TC^p_v mühimmat p'nin bir biriminin araç v ile bir saat taşınma maliyetidir. VC_v araç v'nin tedarik maliyetidir. FC_v transfer noktası i'nin kurulma maliyetidir.

MMİS tasarım problemi şu şekilde özetlenebilir. Muharebe sahasında sonlu sayıda birlik bulunmaktadır. Ayrıca yine sonlu sayıda muhtemel STN'sı ve MTN'sı yerleşimleri bulunmaktadır. Mühimmat STN'larından MTN'larına tırlarla, MTN'larından birliklere mühimmat araçları ile taşınmaktadır. Alınması gereken kararlar şunlardır: (1) açılacak STN'ları ve MTN'larının sayısı ve yerleri ve (2) tır/kamyon ve mühimmat araçlarının sayısı, merkezleri ve güzergâhları. Bu kararlar verilirken aynı zamanda transfer noktası açma maliyetleri, araç satın alma maliyetleri ve mühimmat dağıtım maliyetleri en aza indirilmelidir. Ayrıca şu hususlar sağlanmalıdır; birliklerin tüm ihtiyaçları verilen zaman tahditleri içinde sağlanacaktır, transfer noktası

ve araç kapasitelerine uyulacaktır, her araç kendi transfer noktasından çıkacak ve turunu tamamladıktan sonra aynı transfer noktasına geri dönecektir, her araç bir planlama periyodunda sadece bir kez dağıtım yapacaktır.

3.2. Kısıtlar

3.2.1. Mühimmat Akış Dengesi Kısıtları

Modelde negatif olmayan karar değişkeni olan ve araç v ile düğüm i'den j'ye taşınan mühimmat p'nin miktarını temsil eden f_{ijv}^{p} kullanılmıştır.

$$\sum_{v\in V_F}\left(\sum_{\substack{j\in N_{FM}\\ j\neq i}} f_{jiv}^{p}-\sum_{\substack{j\in N_{M}\\ j\neq i}} f_{ijv}^{p}\right)=\sum_{v\in V_M}\sum_{j\in N_C} f_{ijv}^{p}\quad \forall i\in N_M, p\in P \tag{1a}$$

$$\sum_{v\in V_M}\left(\sum_{\substack{j\in N_{MC}\\ j\neq i}} f_{jiv}^{p}-\sum_{\substack{j\in N_{C}\\ j\neq i}} f_{ijv}^{p}\right)=Q_i^{p}\quad \forall i\in N_C, p\in P \tag{1b}$$

$$\sum_{\substack{j\in N_{FM}\\ j\neq i}} f_{jiv}^{p}\geq \sum_{\substack{j\in N_{FM}\\ j\neq i}} f_{ijv}^{p}\quad \forall i\in N_M, v\in V_F, p\in P \tag{2a}$$

$$\sum_{\substack{j\in N_{MC}\\ j\neq i}} f_{jiv}^{p}\geq \sum_{\substack{j\in N_{MC}\\ j\neq i}} f_{ijv}^{p}\quad \forall i\in N_C, v\in V_M, p\in P \tag{2b}$$

Mühimmat sisteme STN'larından girmekte ve birlikler tarafından tüketilmektedir. Kısıtlar (1), MTN'ları veya birliklere giren mühimmat miktarının, çıkan miktar ile ihtiyacın toplamına eşit olmasını zorlamaktadır. Kısıtlar (2), araçların yolları üzerindeki ara düğümlerden mühimmat almalarına engel olmaktadır.

3.2.2. Araç Akış Dengesi Kısıtları

Modelde ikili karar değişkeni olan x_{ijv} kullanılmıştır. Eğer araç v düğüm i'den j'ye giderse x_{ijv}=1 ve gitmezse x_{ijv}=0.

$$\sum_{i\in N_F}\sum_{j\in N_M} x_{ijv}\leq 1\quad \forall v\in V_F \tag{3a}$$

$$\sum_{i\in N_M}\sum_{j\in N_C} x_{ijv}\leq 1\quad \forall v\in V_M \tag{3b}$$

$$\sum_{j\in N_M} x_{jiv}=\sum_{j\in N_M} x_{ijv}\quad \forall i\in N_F, v\in V_F \tag{4a}$$

$$\sum_{j\in N_C} x_{jiv}=\sum_{j\in N_C} x_{ijv}\quad \forall i\in N_M, v\in V_M \tag{4b}$$

$$\sum_{\substack{j\in N_{FM}\\ j\neq i}} x_{jiv}=\sum_{\substack{j\in N_{FM}\\ j\neq i}} x_{ijv}\quad \forall i\in N_M, v\in V_F \tag{5a}$$

$$\sum_{\substack{j\in N_{MC}\\ j\neq i}} x_{jiv}=\sum_{\substack{j\in N_{MC}\\ j\neq i}} x_{ijv}\quad \forall i\in N_C, v\in V_M \tag{5b}$$

Tır/kamyonlar sadece STN'larında ve mühimmat araçları sadece MTN'larında bulunabilirler. Kısıtlar (3), araçların birden fazla transfer noktasında bulunamayacağını belirtmektedir. Kısıtlar (4), araçların turları sonunda ilk çıkış yaptıkları transfer noktasına dönmelerini sağlamaktadır. Kısıtlar (5), her bir aracın giriş yaptığı düğümden çıkış yapmasını sağlamaktadır. Kısıtlar (3)-(5), ayrıca her bir araç güzergâhında tek bir transfer noktası olmasını sağlamakta ve araçların aynı anda bir düğümden iki farklı düğüme gitmelerine engel olmaktadır.

3.2.3. Kapasite Kısıtları

Modelde ikili karar değişkeni olan y_i kullanılmıştır. Eğer transfer noktası i kullanılıyorsa y_i=1 ve kullanılmıyorsa y_i=0.

$$\sum_{v\in V_F}\sum_{j\in N_M} f_{ijv}^{p} \le CD_i^p \cdot y_i \quad \forall i \in N_F, p \in P \tag{6a}$$

$$\sum_{v\in V_F}\sum_{\substack{j\in N_{FM}\\ j\neq i}} f_{jiv}^{p} \le CD_i^p \cdot y_i \quad \forall i \in N_M, p \in P \tag{6b}$$

$$\begin{aligned} f_{ijv}^{p} \le CV_v^p \cdot x_{ijv} \quad & \forall i \in N_F, j \in N_M, v \in V_F, p \in P; \forall i, j \in N_M, i \neq j, v \in V_F, p \in P \\ & \forall i \in N_M, j \in N_C, v \in V_M, p \in P; \forall i, j \in N_C, i \neq j, v \in V_M, p \in P \end{aligned} \tag{7}$$

$$\begin{aligned} \sum_{p\in P} f_{ijv}^{p} \le CT_v \cdot x_{ijv} \quad & \forall i \in N_F, j \in N_M, v \in V_F; \forall i, j \in N_M, i \neq j, v \in V_F \\ & \forall i \in N_M, j \in N_C, v \in V_M; \forall i, j \in N_C, i \neq j, v \in V_M \end{aligned} \tag{8}$$

Kısıtlar (6), transfer noktalarının kapasitelerinden fazla mühimmat göndermelerini engellemektedir. Ayrıca kullanılmayan transfer noktalarının mühimmat göndermesini engellemektedir. Kısıtlar (7), araç kapasitelerine uyulmasını sağlamakta ve kullanılmayan araçların mühimmat taşımalarına engel olmaktadır. Kısıtlar (8), toplam araç kapasitelerine uyulmasını sağlamaktadır.

3.2.4. İlişki Kısıtları

Modelde karar değişkeni olan w_{ijp} kullanılmıştır. Eğer mühimmat p, i düğümünden j düğümüne gidiyorsa w_{ijp}=1 ve gitmiyorsa w_{ijp}=0.

$$\begin{aligned} \sum_{p\in P} f_{ijv}^{p} \ge x_{ijv} \quad & \forall i \in N_F, j \in N_M, v \in V_F; \forall i, j \in N_M, i \neq j, v \in V_F \\ & \forall i \in N_M, j \in N_C, v \in V_M; \forall i, j \in N_C, i \neq j, v \in V_M \end{aligned} \tag{9}$$

$$\begin{aligned} \left(\sum_{l\in N_C} Q_l^p\right) \cdot w_{ijp} \ge f_{ijv}^{p} \quad & \forall i \in N_F, j \in N_M, v \in V_F, p \in P; \forall i, j \in N_M, i \neq j, v \in V_F, p \in P \\ & \forall i \in N_M, j \in N_C, v \in V_M, p \in P; \forall i, j \in N_C, i \neq j, v \in V_M, p \in P \end{aligned} \tag{10}$$

$$\sum_{v\in V_F} f_{ijv}^{p} \ge w_{ijp} \quad \forall i \in N_F, j \in N_M, p \in P; \forall i, j \in N_M, i \neq j, p \in P \tag{11a}$$

$$\sum_{v\in V_M} f_{ijv}^{p} \ge w_{ijp} \quad \forall i \in N_M, j \in N_C, p \in P; \forall i, j \in N_C, i \neq j, p \in P \tag{11b}$$

Kısıtlar (9), bir araç eğer kullanılıyorsa herhangi bir çeşit mühimmat taşımasını zorlamakta ve mühimmat taşımayan bir aracın yolculuk etmesine izin vermemektedir. Kısıtlar (10) ve (11), karar değişkenleri f ve w arasında doğru mantıksal ilişkileri sağlamaktadır. Diğer bir ifade ile eğer mühimmat p, i'den j'ye gitmiyorsa bu düğümler arasında mühimmat p akışı olmamalıdır ve eğer mühimmat p, i'den j'ye gidiyorsa bu düğümler arasında pozitif bir mühimmat p akışı olmalıdır.

3.2.5. Zamanla İlişkili Kısıtlar

Modelde negatif olmayan karar değişkeni olan ve mühimmat p'nin i düğümüne varış zamanını temsil eden t^p_i kullanılmıştır.

$$TE_i^p \le t_i^p \le TL_i^p \quad \forall i \in N_C, p \in P \tag{12}$$

$$t_i^p = 0 \quad \forall i \in N_F, p \in P \tag{13}$$

$$\begin{aligned} t_i^p + T_{ij} \cdot w_{ijp} - \left(1 - w_{ijp}\right) \cdot TM^p \le t_j^p \quad & \forall i \in N_F, j \in N_M, p \in P; \forall i, j \in N_M, i \ne j, p \in P \\ & \forall i \in N_M, j \in N_C, p \in P; \forall i, j \in N_C, i \ne j, p \in P \end{aligned} \tag{14}$$

Kısıtlar (12), tüm mühimmat çeşitleri için birliklerin zaman pencerelerini ortaya koymaktadır. Kısıtlar (13), mühimmatların STN'larına varış zamanını sıfırlamaktadır. Kısıtlar (14), mühimmatların düğümlere varış zamanını doğru olarak hesaplamaktadır.

Karar değişkeni w_{ijp} araçlar hakkında hiçbir bilgi taşımamaktadır ve bu sebeple kısıtlar (12)-(14) araç alt-turlarını engelleyememektedir. Buna engel olmak için alt-tur eleme kısıtları (15) kullanılmıştır [5]. Karar değişkeni tt^v_i araç v'nin i düğümüne varış zamanını temsil etmektedir.

$$\begin{aligned} tt_i^v + T_{ij} \cdot x_{ijv} - \left(1 - x_{ijv}\right) \cdot TM^p \le tt_j^v \quad & \forall i \in N_F, j \in N_M, v \in V_F; \forall i, j \in N_M, i \ne j, v \in V_F \\ & \forall i \in N_M, j \in N_C, v \in V_M; \forall i, j \in N_C, i \ne j, v \in V_M \end{aligned} \tag{15}$$

3.2.6. Amaç Fonksiyonu

Amaç fonksiyonu üç değişik maliyet biriminden oluşmaktadır. (16) mühimmatın toplam dağıtım maliyeti, (17) araçların toplam tedarik maliyeti ve (18) transfer noktalarının tesis maliyetidir.

$$\sum_{i \in N} \sum_{j \in N} \sum_{v \in V} \sum_{p \in P} f_{ijv}^p \cdot TC_v^p \cdot T_{ij} \tag{16}$$

$$\sum_{i \in N_F} \sum_{j \in N_M} \sum_{v \in V_F} x_{ijv} \cdot VC_v + \sum_{i \in N_M} \sum_{j \in N_C} \sum_{v \in V_M} x_{ijv} \cdot VC_v \tag{17}$$

$$\sum_{i \in N_{FM}} y_i \cdot FC_i \tag{18}$$

3.2.7. Model

Yukarıdaki açıklamaların ışığında MMİS tasarım problemi aşağıda gösterilmiştir.

min (16)+(17)+(18)

s.t. (1) – (15)

$$f_{ijv}^{p} \geq 0 \quad \forall i,j \in N, i \neq j, v \in V, p \in P$$

$$t_{i}^{p} \geq 0 \quad \forall i \in N, p \in P$$

$$tt_{i}^{v} \geq 0 \quad \forall i \in N, v \in V$$

$$x_{ijv} \in \{0,1\} \quad \forall i,j \in N, i \neq j, v \in V$$

$$w_{ijp} \in \{0,1\} \quad \forall i,j \in N, i \neq j, p \in P$$

$$y_{i} \in \{0,1\} \quad \forall i \in N_{FM}$$

3.3. Geçerli Eşitsizlikler (Valid Inequalities)

Daha önceki bölümde MMİS tasarım problemi karışık tamsayılı programlama olarak modellenmiştir. Müteakiben çözüm zamanını kısaltmak maksadıyla bazı geçerli eşitsizlik belirlenmiş ve bunlar 4'üncü bölümde anlatıldığı şekilde test edilmiştir. 4'üncü bölümde sunulacak olan sonuçlara göre çözüm zamanı açısından modele en olumlu katkıyı sağlayan eşitsizlik aşağıda sunulmuştur.

$$\sum_{\substack{j \in N_{FM} \\ j \neq i}} x_{ijv} \leq 1 \quad \forall i \in N_{M}, v \in V_{F} \qquad \text{(GE–1a)}$$

$$\sum_{\substack{j \in N_{MC} \\ j \neq i}} x_{ijv} \leq 1 \quad \forall i \in N_{C}, v \in V_{M} \qquad \text{(GE–1b)}$$

(GE–1), bir aracın bir planlama periyodunda her hangi bir düğümden iki veya daha fazla düğüme gitmesini engellemektedir.

4. HESAPLAMA DENEYLERİ

Bu bölümde öncelikle geçerli eşitsizliklerin performansları bazı test problemleri üzerinde değerlendirilmiştir. Müteakiben seçilen geçerli eşitsizlikler esas modele eklenerek oluşturulan modelin performansı gerçek senaryolar üzerinde denenmiştir. Tüm hesaplamalar 1.83 GHz CPU, 1 GB RAM ve Windows XP işletim sistemi olan bir diz üstü bilgisayarda yapılmıştır. Karışık tamsayılı programlama çözücüsü olarak ILOG CPLEX 10.1 ve kodlama dili olarak GAMS 22.3 kullanılmıştır.

4.1. Test problemleri ile yapılan deneyler

Geçerli eşitsizlikler hakkında fikir sahibi olabilmek için altı adet test problemi (TP) oluşturulmuştur. Test problemlerinde kullanılan veri şu şekildedir; $|N_F|$=3, $|N_M|$=8, $|N_C|$=10, $|V_F|$=5, $|V_M|$=10 ve $|P|$=3. Geçerli eşitsizlikleri mukayese ederken, CPLEX tarafından verilen eniyilik aralık $(|z_c - z_b|)/(|z_c|)$ değerleri kullanılmıştır. z_b olabilecek en iyi çözümün amaç fonksiyon değeri ve z_c şu anki çözümün amaç fonksiyonu değeridir. Hesaplamalar 3600 saniye sonra sonlandırılmıştır. Çizelge 2'de en iyi geçerli eşitsizliğin performansı sunulmuştur.

Çizelge 2. Geçerli eşitsizlik mukayesesi

	Bir saat sonraki eniyilik aralığı (%)								
	TP 1	TP 2	TP 3	TP 4	TP 5	TP 6	En az	Ortalama	En fazla
Esas Model	16.36	-	15.76	4.35	16.80	-	4,35	-	-
GE–1	14.44	8.45	11.16	0.61	16.16	14.11	0,61	10,82	16,16

4.2. Gerçek hayat problemleri ile yapılan deneyler

Bu bölümde (GE–1) eklendikten sonra modelin büyük ölçekli bir muharebe senaryosuna uygulaması ve performansı gösterilmektedir. Senaryoda bir kolordu taarruz etmektedir. Kolordunun dört tugayı ve her tugayın beş taburu bulunmakta ve mühimmat bu taburlara ikmal edilmektedir. Kolordu bölgesinde STN'sı için üç ve MTN'sı için sekiz ayrı muhtemel yerleşim belirlenmiştir. Tugaylar sıra ile ikmal edilmektedir. Diğer bir ifade ile bir tugayın tüm taburları aynı ikmal zaman penceresine sahiptir ve tugayların zaman pencereleri birbirleri ile çakışmamaktadır. Karmaşıklığı azaltmak için mühimmat çeşitleri günlük kullanım oranlarına göre üç gruba ayrılmıştır. Kolordu da sekiz tır/kamyon ve on altı adet mühimmat aracı bulunmaktadır. İnsan gücü ve malzeme eksikliklerinden dolayı, kolordu bir STN'sı ve her tugay birer MTN'sı açacaktır. Taburlar konum ve ihtiyaçlarını kendi tugaylarına bildirdiklerinden dolayı, her tabur kendi tugayının MTN'sı tarafından ikmal edilecektir. Senaryo çözüm sonuçları Çizelge 3'de özetlenmiştir.

Çizelge 3. Senaryo çözüm özeti

	En iyilik aralığı (%)		
	5 saat	15 saat	24 saat
Esas Model	-	-	-
GE–1	2.22	2.18	2.00

Çizelgeden görüleceği üzere esas model 24 saatte bir çözüm bulmazken, (GE-1) eklenmiş model senaryoyu beş saat içinde % 2.22 en iyilik aralığına kadar çözebilmektedir.

5. ÖZET VE SONUÇLAR

Soğuk Savaş döneminin sona ermesi ile birlikte K.K.K.lığı ateş gücünün yanı sıra çevikliğine de güvenen daha ufak bir kuvvet yapısına yönelmiştir. Hızlı hareket kabiliyetine sahip yeni kuvvetleri desteklemek, muharebe alanında esnek ve etkin bir dağıtım sistemi sağlayabilmek maksadıyla bu çalışmada Mobil Mühimmat İkmal Sistemi (MMİS) teklif edilmiştir.

Çalışmada, öncelikle MMİS tasarım probleminin bir yer seçimi-yol atama problemi olduğu gösterilmiştir. Müteakiben daha önce yayımlanmış uygun bir model bulmak maksadıyla literatür taranmıştır. Literatür araştırılırken 17 kriterden oluşan bir sınıflandırma oluşturulmuş, mevcut 52 çalışma sınıflandırılmış ve daha önce hiç veya az dikkate alınmış kıstaslar belirlenmiştir. En önemli saptamalar şöyledir; hiç bir çalışmada iki taraflı zaman pencereleri bulunmamaktadır, üç kademeli çalışmalardan sadece iki adedi iki değişik kademede yer seçimi yapmaktadır, üç çalışma haricinde diğerleri tek ürün dağıtmaktadır ve hiç bir çalışma çoklu ikmal yapmamaktadır. Ancak MMİS

tasarım problemi bu dört unsuru da içermektedir. Bu çalışmada ele alınan problem nispeten yenidir ve literatürde benzer bir model bulunmamaktadır. Bu sebeple ürün akışı yaklaşımı ile yukarıda açıklanan tüm unsurları içeren bir karışık tamsayılı model geliştirilmiştir. Müteakiben çözüm zamanını kısaltmak maksadıyla birçok geçerli eşitsizlikler bulunmuş ve performansları test edilmiştir. Son olarak seçilen geçerli eşitsizlik modele eklenerek gerçek bir senaryo çözülmüştür. Sonuçlar bu senaryonun makul bir zaman içinde ve tatmin edici bir seviyede çözülebildiğini göstermektedir.

KAYNAKÇA

[1] Salhi, Said, Graham K. Rand, (1989), "The efect of ignoring routes when locating depots", European Journal of Operational Research, 39, 150-156.

[2] MTN, Hokey, Vaidyanathan Jayaraman, Rajesh Srivastava, (1998), "Combined location-routing problems: A synthesis and future research directions", European Journal of Operational Research, 108, 1-15.

[3] Laporte, Gilbert, (1988), "Location-routing problems", Bruce L. Golden, A.A. Asad, eds., Vehicle Routing: Methods and Studies, North-Holland Publishing, Amsterdam,Holland, 163-198.

[4] List, George F., Pitu B. Mirchandani, Mark A. Turnquist, Konstantinos G. Zografos, (1991), "Modeling and analysis for hazardous materials transportation: Risk analysis, routing/scheduling and facility location", Transportation Science 25(2), 100-114.

[5] C.E.Miller, A.W.Tucker, R.A.Zemlin, (1960), "Integer programMTNg formulation of traveling salesman problems", J.Assoc.Comput.Mach., 7, 326-329.

ARAÇ ROTALAMA PROBLEMLERİNİN ÇÖZÜMÜNE BİR YAKLAŞIM[9]

Esin ELDELEKLİ [(b)], Prof. Dr. Kemal LEBLEBİCİOĞLU [(a)]

[(a)] Orta Doğu Teknik Üniversitesi, Elektrik-Elektronik Müh. Böl., 06531, Ankara,
kleb@metu.edu.tr

[(b)] Esin ELDELEKLİ, Kara Harp Okulu, 06654, Ankara,
esineldelekli@gmail.com

ÖZET

Bu çalışmada belli kısıtlar altındaki bazı araç rotalama problemleri ele alınmıştır. Temel problem, farklı bölgelerde bulunan geri dönüşüm atıklarının toplandığı noktaların T (T>1) günlük bir periyot içerisinde belli kısıtlar sağlanarak boşaltılması problemidir. Amaçlanan bu atık noktalarının boşaltılması işleminde minimum mesafeyi veren rotaların bulunmasıdır. Depo noktası ve kapasite kısıtı kriterlerine bağlı olarak üç ana varyasyon ele alınmıştır. Bu varyasyonlara ait çözümleri bulmak için kullanılan arama tekniği genetik algoritmadır.

Anahtar Kelimeler: Araç Rotalama Problemi, Genetik Algoritma

AN APPROACH TO VEHICLE ROUTING PROBLEMS ABSTRACT

In this study, special cases of vehicle routing problem are considered. The problem is recycling waste collection during T (T>1) day period that recycle waste containers are in the different parts of an unnamed area. When solving the problem, aim of the study is minimizing cost of collection that is the total distance of the routes. Three modifications depending on the depot point and capacity constraints of the vehicle routing problem are considered. In all these problems genetic algorithm is used as the solution technique.

Keywords: Genetic Algorithm, Vehicle Routing Problem

1. GİRİŞ

Artan nüfus ve büyüyen şehirlerde ulaşım, dağıtım veya toplama işlemleri beraberinde büyük sorunlar getirmektedir. Bu işlerin belediyeler veya şirketler tarafından etkin bir şekilde gerçekleştirilebilmeleri için maliyetlerin azaltılması veya karın arttırılması gerekmektedir. Karın arttırılması veya maliyetlerin azaltılması bu tip problemlerde

[9] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

başarılı bir rotalama işlemi ile gerçekleşmektedir. Dolayısıyla en etkili güzergahlar maliyetlerin azaltıldığı ve/veya karın arttırıldığı güzergahlar olacaktır [1]. Maliyetin azaltılması içinde en önemli nokta seçilen rotada kullanılan yolların en kısa mesafeyi vermesidir.

Araç rotalama probleminden türetilen problemlerden bazıları şunlardır: zaman aralıklı araç rotalama problemleri, kapasite sınırlı araç rotalama problemleri, periyodik araç rotalama problemleri, geri dönüşlü araç rotalama problemleri, dinamik araç rotalama problemleridir [2]. Yaptığımız çalışmada ele alınan problemler periyodik araç rotalama problemine benzemekle birlikte belirli bir problem tipine uymamaktadır.

2. PROBLEMLERİN TANIMLARI

Ele aldığımız problemler bir bölgede farklı yerlerde bulunan atık noktalarının T (T>1) günlük bir periyot içerisinde belli kısıtlar sağlanarak ve tek bir araç kullanılarak boşaltılmasıdır. Çalışmada amaç aracın farklı noktalarda bulunan atık noktalarını belirlenen tur sayısı içerisinde kısıtları sağlayacak şekilde ziyaret etmesi üzerine kurulmuştur. Problemlerde yer alan kısıtlardan biri süre kısıtıdır. Araç her turu 4 saati aşmayacak şekilde gerçekleştirmelidir. Tur süresi olarak aracın tur içerisinde ziyaret ettiği atık noktalarının arasında geçen süre alınmış, atık noktalarının boşaltılması ve doğabilecek problemler için gereken süreler göz ardı edilmiştir. Probleme ait diğer bir kısıt ise bir atık noktasının aynı tur içerisinde birden fazla kez ziyaret edilmemesidir. Aynı zamanda bir atık noktasının hiç ziyaret edilmemesi durumuna engel olabilmek için atık noktasının bütün turlar içerisinde en az bir kez ziyaret edilmesi kısıtı konulmuştur. Yukarıda verilen tüm kısıtlar ele aldığımız üç problem türü içinde geçerlidir. Ayrıca problemler için problemlerin yapısı itibariyle bazı ek kısıtlar söz konusudur.

2.1. Depo Noktası ve Kapasite Kısıtı Olmayan Araç Rotalama Problemi

Araç rotalama probleminin tanımında yer alan 'müşteri' kavramını bu problemde atık noktaları karşılamaktadır. Her gün bir tur gerçekleştirilmekte ve araç, ilk turda belirlenen ilk noktadan başlayarak turu tamamladıktan sonra turda yer alan en son noktada yani bulunduğu noktada diğer tur (gün) için beklemektedir. Atık noktaları günlük dolum oranına sahiptir. Atık noktaları boşaltılmadıkları her gün kendilerine ait günlük dolum oranı kadar dolmaktadır. Hangi atık noktasının kaç kez ziyaret edileceği turun akışına ve günlük dolum oranlarına göre belirlenir. Bu probleme ait matematiksel formülasyon aşağıda verilmiştir.

$N = \{1,...,20\}$, atık noktalarının sayısı,

$K = \{1,...,10\}$, gün sayısı,

d_{ij} = i. atık noktasından j. atık noktasına olan mesafe,

s_{ij} = i. atık noktasından j. atık noktasına gitmek için gereken süre,

$S = 4$, maksimum tur süresi,

$$x_{it} = \begin{cases} 1 & \text{,i. atik noktasi t gününde ziyaret ediliyorsa} \\ 0 & \text{,ö.d.} \end{cases}$$

$$u_{ijt} = \begin{cases} 1 & \text{,i. atik noktasından j. atik noktasına t gününde gidiliyorsa} \\ 0 & \text{,ö.d.} \end{cases}$$

$$\text{Min} \quad \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{K} d_{ij}u_{ijt} \qquad (1)$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N} s_{ij}u_{ijt} \le S, \quad \forall t \in K, \qquad (2)$$

$$x_{it} \le 1, \quad \forall i \in N, t \in K \qquad (3)$$

$$\sum_{t=1}^{K} x_{it} \ge 1, \quad \forall i \in N, \qquad (4)$$

$$x_{it} \in \{0,1\}, \quad \forall i \in N, t \in K, \qquad (5)$$

$$u_{ijt} \in \{0,1\}, \quad \forall i, j \in N, t \in K, \qquad (6)$$

Matematiksel formülasyonda denklem (1) problemin amaç fonksiyonudur. Aracın kat ettiği mesafe minimize edilmektedir. Eşitsizlik (2) de süre kısıtını ortaya koymaktadır. Aracın her bir t gününde (turunda) süreyi aşmaması sağlanmaktadır. Eşitsizlik (3) herhangi bir atık noktasının bir tur içinde tekrar ziyaret edilmesini engellemektedir fakat formülasyonda yer alan değişkenler 0-1 tamsayılı değişkenler oldukları için bu kısıt kendiliğinden sağlanmaktadır. Eşitsizlik (4) de ise atık noktalarının tümünün mutlaka verilen periyot içerisinde yer alması sağlanmaktadır. Periyot içerisinde atık noktalarının her biri en az bir kez yer almalıdır. (5) ve (6), çözümü oluşturan değişkenlerin 0-1 tamsayılı değişkenler olduğunu göstermektedir.

2.2. Depo Noktası Olan ve Kapasite Kısıtı Olmayan Araç Rotalama Problemi

Bu problem depo noktası ve kapasite kısıtı olmayan araç rotalama probleminin bütün kısıtlarını taşımaktadır. Bu problemin depo noktası ve kapasite kısıtı olmayan araç rotalama probleminden farkı aracın tur sonunda gideceği, ayrıca tura bu noktadan başlayacağı bir depo noktasının bulunmasıdır. Problemde araç, her bir turu tamamladıktan sonra belirlenen depoya dönmektedir. Burada depo noktasının getirilmesi ile amaçlanan aracın tur içinde edindiği yükü boşaltabilmesidir. Bu probleme ait matematiksel formülasyon aşağıda verilmiştir.

$$\text{Min} \quad \sum_{i=0}^{N}\sum_{j=0}^{N}\sum_{t=1}^{K} d_{ij}u_{ijt} \qquad (1)$$

$$\sum_{i=0}^{n}\sum_{j=0}^{n} s_{ij}u_{ijt} \le S, \quad \forall t \in K, (i \ne j) \qquad (2)$$

$$x_{it} \le 1, \quad \forall i \in N, t \in K \qquad (3)$$

$$\sum_{t=1}^{K} x_{it} \ge 1, \quad \forall i \in N, \qquad (4)$$

$$x_{it} \in \{0,1\}, \quad \forall i \in N, t \in K, \qquad (5)$$

$$u_{ijt} \in \{0,1\}, \quad \forall i, j \in N, t \in K, \qquad (6)$$

$$\sum_{i=0}^{N} u_{ipt} = \sum_{j=0}^{N} u_{pjt}, \quad \forall p \in N, t \in K \qquad (7)$$

Yukarıda verilen formülasyonda i veya j indisinin 0 değerini alması aracın depodan çıktığını veya depoya gittiğini göstermektedir. Denklem (7) ile aracın bir atık noktasını ziyaret ettiğinde o atık noktasında beklememesi, terk etmesi sağlanmaktadır. Aynı zamanda denklem aracın aynı tur içersinde depodan bir atık noktasına, herhangi bir atık noktasından tekrar depoya dönmesini sağlamaktadır.

2.3. Depo Noktası ve Kapasite Kısıtı Olan Araç Rotalama Problemi

Bu problem depo noktası olan ve kapasite kısıtı olmayan araç rotalama probleminin kapasite kısıtı getirilmiş durumudur. Dolayısıyla diğer problemlerde yer alan bütün kısıtlar bu problem içinde geçerlidir. Bu problemde araca kapasite kısıtı getirilmiştir. Turların gerçekleştirilmesi, dolayısıyla atık noktalarının boşaltılması işlemi aracın belirli bir kısıta sahip olduğu düşünülerek gerçekleştirilmektedir. Kapasite yetersizliği ile gerçek hayatta ele alınan problemlerde sıkça karşılaşılmaktadır. Bu durum problemimizin gerçek hayatta karşılaşılan problemlere daha uygun hale gelmesini sağlamaktadır. Bu problemin matematiksel formülasyonu depo noktası olan ve kapasite kısıtı olmayan araç rotalama problemine ait matematiksel formülasyondaki tüm eşitlik ve eşitsizlikleri içermektedir. Ayrıca kapasite kısıtı ile yeni iki denklem matematiksel formülasyonda yer almaktadır.

M = aracın kapasitesi,

m_{it} = t.günde i.atık noktasının doluluk oranı

ma_{it} = t.günde i.atık noktasından alınan atık miktarı

$$\sum_{i=1}^{N} ma_{it} x_{it} = M, \quad \forall t \in K \qquad (7)$$

$$ma_{it} \le m_{it}, \quad \forall i \in N, t \in K \qquad (8)$$

Denklem (7) bu problemde araca getirilen kapasite kısıtını göstermektedir. Herhangi bir turda boşaltılan atık noktalarının doluluk oranları toplamı aracın kapasitesine eşit alınmaktadır. Başka bir deyişle araç son atık noktasındaki atıkların hepsini boşaltamasa da kapasitesinin izin verdiği ölçüde o atık noktasını boşaltmaktadır. Dolayısıyla tam olarak boşaltılmış olmasa da bu atık noktası da tura dahil edilir.

3. GENETİK ALGORİTMA

Problemin çözümü için evrimsel teoriye dayanan [3], genetik algoritma kullanılmıştır. Genetik algoritma bir problem için mümkün ve uygun çözümlerden belirli sayıda popülasyon yaratarak, bu popülasyonun çeşitli işlemlerden (doğal seçme, üreme gibi)

geçerek her nesilde (generation) art arda geliştirilmesidir [4]. Kuşağın en uygun yani en kaliteli bireyi problem için en uygun çözümü verir.

Çözüm uzayında bir noktanın uygunluk değeri (kalitesi) "uygunluk fonksiyonu" ile belirlenir [5]. Bu çalışmada ele alınan probleme ait uygunluk fonksiyonu amaç fonksiyonundan etkilendiği gibi atık noktalarının günlük dolum oranlarından da etkilenmektedir. Aşağıda uygunluk fonksiyonu verilmiştir.

$$\text{Max} \quad 50000 - (10 * \sum_{i=0}^{N}\sum_{j=0}^{N}\sum_{t=1}^{K} d_{ij} u_{ijt} + \sum_{i=1}^{N} \max(0, tdo_i - 100) * 0{,}2 + 50 * v(tdo_i - 100)$$

tdo_i = i. atık noktasına ait toplam doluluk oranı

$$v(x) = \begin{cases} 1, & x \le 0 \\ 0, & x > 0 \end{cases}$$,birim fonksiyondur.

Depo noktası olan problemler için i ve j değişkenleri sıfırdan başlamaktadır. Problemlerimizde amaç, amaç fonksiyonunu minimize etmek olduğundan ve genetik algoritma maksimizasyona dayalı bir algoritma olduğundan uygunluk fonksiyonu yukarıda verildiği gibi maksimizasyona dönüştürülmüştür. Uygunluk fonksiyonunda eğer atık noktalarının bazılarının veya hepsinin toplam doluluk oranı %100'ü aşıyorsa %100'ü aşan kısmın 0,2'si alınarak 50 ceza puanı ile toplam gidilen mesafeye eklenmektedir. Bu şekilde, %100 doluluk oranını aşan atık noktası dizilimini içeren kromozomların algoritma tarafından tercih edilmemesi amaçlanmaktadır. Genetik algoritmaya ait çözüm aşamaları açıklanırken problemler arasında farklılık olmadığı durumlarda anlatım tek bir problem üzerinden olacaktır.

3.1. Çözümlerin Kodlanması

Çözümlerin genetik algoritmada kromozom olarak ifade edilebilmeleri ve kodlanmış çözüm uzayında yer almaları için kodlanmaları gerekmektedir. Problemimiz için en uygun olanı gerçek kodlama tekniğidir. Şekil-1'de depo noktası olmayan problem için bir çözüm örnek olarak verilmiştir.

1. tur: 2 4 5	6. tur: 13 9
2. tur: 18	7. tur: 4 14 18
3. tur: 20 1 2 6	8. tur: 3 10 15
4. tur: 11 19 7 8 12	9. tur: 10
5. tur: 17	10.tur: 1 6

Şekil-1: Probleme Ait Bir Çözümün Gösterimi

Şekil-2'de ise yukarıda verilen çözümün gerçek kodlama ile kodlanmış hali verilmiştir.

Kromozom:
2 4 5 18 20 1 2 6 11 19 7 8 12 17 13 9 4 14 18 3 10 15 10 1 6

Şekil-2: Probleme Ait Çözümün Kromozom Olarak İfade Edilmesi

3.2. Doğal Seçim İşleminin Uygulanması

Doğal seçim yöntemlerinden rulet tekerliği seçim yöntemi [4] çaprazlama ve mutasyon aşamalarında ana kromozomların seçimi için kullanılmaktadır. Ayrıca ana kromozom çiftlerinden elde edilen yeni kromozomların popülasyona eklenmesinde de sıralama seçim yöntemi [4] kullanılmaktadır. Yeni popülasyonu elde etmek için yapılan işlemler: yeni kromozomlar ve popülasyondaki diğer kromozomlar bir araya getirilir, uygunluk değerlerine göre büyükten küçüğe sıralanır ve popülasyon büyüklüğüne eşit sayıda kromozom en büyükten başlayarak sırayla alınır.

3.3. Çaprazlama İşleminin Uygulanması

Çaprazlama operatörü temel olarak popülasyondan seçilen iki ana çözümün (bireyin) arasında bilgi değişimi yapılması ile yeni çözümlerin üretilmesidir [6]. Çaprazlanacak çiftlerin yarısı çok noktalı çaprazlama yöntemi [6] ile diğer yarısı tek noktalı çaprazlama yöntemi [6] ile çaprazlanır. Algoritmada kullanılan çok noktalı çaprazlama yöntemi iki ana kromozomdan seçilen ikişer turun karşılıklı olarak kromozomlar arasında yer değiştirmesi şeklindedir. Algoritmada kullanılan tek noktalı çaprazlama yöntemi iki ana kromozomdan seçilen ikişer tur arasında klasik tek noktalı çaprazlama yönteminin uygulanması şeklindedir.

3.4. Mutasyon İşleminin Uygulanması

Mutasyon operatörü ile bir kromozomda mutasyon oranına bağlı ve rassal olarak belirlenen noktalardaki değerler yenileriyle değiştirilerek yeni kromozomlar elde edilir [7]. Problem için oluşturulan genetik algoritmada gerçek kodlama ile kodlanmış kromozomların mutasyonu için iki mutasyon yöntemi kullanılmaktadır. Mutasyona girecek belirlenen sayıda kromozoma basit mutasyon yöntemi [7] ve yine belirlenen bir sayıda ki kromozoma karşılıklı değişim mutasyon yöntemi [7] uygulanmaktadır. Basit mutasyon yönteminde popülasyondan seçilen ana kromozomda rasgele seçilen atık noktalarıyla tüm atık noktaları içinden rastsal olarak seçilen atık noktaları yer değiştirmektedir. Karşılıklı değişim mutasyon yöntemindeyse seçilen ana kromozomda belirlenen iki noktadaki atık noktaları karşılıklı olarak yer değiştirerek yeni kromozom elde edilmektedir.

3.5. Kontrol Operatörünün Kullanılması

Kontrol operatörü genetik algoritmanın bir parçası olmamakla birlikte problemlerimize getirilen kısıtlar dolayısıyla kullanılmaktadır. Genetik algoritmanın çaprazlama ve mutasyon adımlarında elde edilen yeni kromozomların istenilen kısıtları sağlamadıkları durumlar ortaya çıkmaktadır. Bu nedenle çaprazlama ve mutasyon işlemlerinden elde edilen yeni çocuk çözümler kontrol operatöründen geçmektedir.

Problemler için ortaya konan kısıtlardan ilki, aracın herhangi bir tur içerisinde bir atık noktasını en fazla bir kez ziyaret edebilmesidir. Kontrol operatöründe ilk adımda bu kısıtın sağlanıp sağlanmadığı kontrol edilmektedir. Çaprazlama veya mutasyon

işleminden elde edilen yeni çocuk çözüm kontrol operatörüne girdiğinde bir tur içerisinde birden fazla sayıda yer alan atık noktaları belirlenir. Daha sonra belirlenen atık noktalarının ziyaret sayısı bir olacak şekilde çözüm yeniden düzenlenir ve operatörde yer alan bir sonraki adıma geçilir.

Problemler için ortaya koyulan kısıtlardan ikincisi ise gerçekleştirilecek turların süresinin 4 saati aşmamasıdır. Kontrol operatöründe ilk adımdan elde edilen yeni çözümde yer alan turların süreleri hesaplanır. Çözüme ait hesaplanan tur sürelerinden 4 saatlik süreyi aşan bir tur var ise o turun süresi 4 saatin altında veya eşit oluncaya kadar o turdan bir atık noktası çıkarılır. Tur sürelerinin hepsi istenen kısıtı sağladığı durumda yeni çözüm elde edilmiş olur ve operatörün üçüncü adımına geçilir.

Problemler için ortaya konan son kısıt aracın tüm atık noktalarını periyot içerisinde en az bir kez ziyaret etmesi zorunluluğudur. Çözümde yer almayan atık noktaları belirlenir. Belirlenen bir atık noktasının hangi tura gireceği tur sürelerine göre belirlenir. Atık noktası tur içerisinde yer alabileceği bütün noktalara tek tek konarak yeni çözümler oluşturulur. Bu çözümlerden uygunluk değeri en iyi olan çözüm alınır. Kontrol operatörü üçüncü adımı da tamamladıktan sonra tekrar başa dönmektedir. Tüm kısıtlar sağlandığı zaman kontrol operatöründen çıkılır.

3.6. Elitizm İşleminin Uygulanması

Çaprazlama ve mutasyon operatörleri kullanıldığında o popülasyondaki en iyi çözümün kaybedilme ihtimali vardır. Bu yüzden bazı çalışmalarda elitizm operatörü kullanılmaktadır [8]. Elitizm operatörü ile en iyi çözüm her zaman popülasyonda tutulur [8]. Ele alınan problem için kullanılan algoritmada eğer bir popülasyondaki en iyi çözüm çaprazlama ve mutasyon işlemlerinden sonra değişiyorsa bu çözüm bahsedilen işlemlerden sonra elde edilen yeni popülasyona eklenmektedir. Tüm bu işlemlerin sona ermesiyle eğer gereken şart sağlanmıyorsa, genetik algoritma adımlarının ilkine dönülmektedir. Başka bir deyişle, bir sonraki nesile geçilmektedir.

4. DENEYSEL ÇALIŞMALAR

Problemlerin çözümünde kullanılan genetik algoritma, MATLAB 7.0 programı ile kodlanmış ve 1.9 GHz 512 MB Pentium 4 işlemciye sahip bir bilgisayarda çalıştırılmıştır. Program, verilerin ve bazı parametre değerlerinin kullanıcı tarafından girilebileceği veya rassal olarak üretilebileceği şekilde tasarlanmıştır.

Program çalıştırılırken kullanıcıdan verilerin girilip girilmeyeceğinin bilgisi istenmektedir. Kullanıcının verdiği bilgiye dayanarak ya veriler rassal olarak elde edilmekte ya da kullanıcı daha önce oluşturduğu bir dosyadan verileri girmektedir. Bu veriler probleme ait çözümlerden oluşan başlangıç popülasyonu, atık noktalarına ait günlük dolum oranları, uzaklık matrisi ve süre matrisidir. Ele alınan problemlerde günlük dolum oranları sabittir. Tüm problemler için periyodun başında tüm atık noktalarının boş olduğu varsayılmaktadır. Her iki durumda da kullanıcıdan parametre değerlerinin mutlaka girilmesi istenmektedir. Bu parametreler atık noktası sayısı, tur

sayısı, maksimum kromozom uzunluğu, popülasyon büyüklüğü, maksimum nesil sayısıdır.

Bu çalışmada ele alınan üç tip problem için farklı düzeyde girdiler alınarak farklı problemler ortaya konmuş ve girdilerin problemlerde nasıl bir etki yarattığı gözlemlenmiştir. Tüm problemlerde atık noktası sayısı 20 olarak alınmış ve tur sayısı 10 olarak belirlenmiştir. Başlangıç popülasyonu rassal olarak üretilmiş ve her problemde aynı popülasyon başlangıç popülasyonu olarak kullanılmıştır. Uzaklık matrisi de 5 ve 150 km. arasında değişen değerler alınarak rassal olarak elde edilmiş ve gerçek hayatta kullanılan aynı rotaların (tek yönlü yollar, trafik kuralları vb. nedeniyle) tekrar kullanılamayacağı düşünülerek asimetrik olarak alınmıştır. Süre matrisi ise aracın hızının 40 km. olduğu düşünülerek uzaklık matrisinden elde edilmiştir.

Depo noktası ve kapasite kısıtı olmayan araç rotalama problemi için üç örnek problem oluşturulmuştur. Bu problemlerde girdilerden günlük dolum oranı üç farklı düzeyde alınmıştır. Örnek problemlerden ilkinde günlük dolum oranları her bir depoya %10 ile %25 arasında atanmıştır. İkincisinde günlük dolum oranları her bir depoya %30 ile %50 arasında, üçüncü örnek problemde %50 ile %70 arasında atanmıştır.

Depo noktası olan ve kapasite kısıtı olmayan problem için de, depo noktası olmayan problemlerde olduğu gibi üç farklı düzeyde günlük dolum oranları alınarak üç farklı örnek problem çözülmüştür.

Depo noktası ve kapasite kısıtı olan problemden ise yine üç farklı düzeyde girdi alınarak üç tane örnek problem türetilmiştir. Fakat diğer problem tiplerinden farklı olarak burada atık noktalarına ait günlük dolum oranları bu üç örnek problemde de %10 ile %25 arasında alınmıştır. Bu problemlerde farklı düzeylerde alınan girdi araca ait kapasite düzeyidir. Birinci örnek problemde aracın kapasitesi 2 tam dolu atık kutusunu, ikincide 3 tam dolu atık kutusunu ve üçüncü problemde 4 tam dolu atık kutusunu alacak şekilde düzenlemiştir.

Problem tiplerinden üretilen örnek toplam 9 problem için genetik algoritma uygulanmış ve en iyi rotalar elde edilmiştir. Depo noktası olmayan problemler için elde edilen en iyi rotalara bakıldığında günlük dolum oranlarının artırılmasının periyodun sonlarında yer alan turlarda yığılmalara yol açtığı görülmektedir. Başka bir deyişle, ilk turlarda az sayıda atık noktası, son turlarda ise çok sayıda atık noktası aynı tur içerisinde ziyaret edilmektedir. Ayrıca turlar arasında uğranılan bir depo bulunmadığından araç bir tur boyunca bir atık noktasında bekleme eğilimi göstermektedir. Şekil-3'de verilen çözümde araç 3. atık noktasında bekleme eğilimi göstermektedir.

1. tur: 20	6. tur: 16
2. tur: 18 9 1	7. tur: 9 7 10 13
3. tur: 4	8. tur: 17 14 11 6 15 12
4. tur: 3	9. tur: 8 2 5
5. tur: 3	10.tur: 19

Şekil-3: Depo Noktası ve Kapasite Kısıtı Olmayan %30 ile %50 Günlük Doluluk Oranına Sahip Atık Noktalarının Boşaltılması İçin Elde Edilen En İyi Rotalar

Depo noktası olan ve kapasite kısıtı olmayan problemler için elde edilen en iyi rotalara bakıldığında depo noktası olmayan problemlerde olduğu gibi günlük dolum oranlarının artırılmasının periyodun sonlarında yer alan turlarda yığılmalara yol açtığı görülmektedir. Şekil-4'de son turlardaki yığılmalar görülmektedir.

1. tur: 0 6 0	6. tur: 0 13 17 15 0
2. tur: 0 14 0	7. tur: 0 11 19 0
3. tur: 0 3 0	8. tur: 0 20 18 16 9 1 12 8 2 0
4. tur: 0 6 0	9. tur: 0 5 4 7 10 0
5. tur: 0 20 0	10.tur: 0 6 0

Şekil-4: Depo Noktası Olan ve Kapasite Kısıtı Olmayan %30 ile %50 Günlük Doluluk Oranına Sahip Atık Noktalarının Boşaltılması İçin Elde Edilen En İyi Rotalar

Depo noktası ve kapasite kısıtı olan problemler için elde edilen en iyi rotalara bakıldığında araca ait kapasite 2 tam dolu atık depoya eşit olduğunda atık noktalarının çoğunlukla ilk iki turda ziyaret edildikleri görülmektedir.

1. tur: 0 13 17 18 16 6 15 0	6. tur: 0 8 3 0
2. tur: 0 1 4 19 0	7. tur: 0 14 0
3. tur: 0 9 0	8. tur: 0 12 0
4. tur: 0 7 10 0	9. tur: 0 20 0
5. tur: 0 11 2 0	10.tur: 0 5 0

Şekil-5: Araca ait Kapasitenin 2 Tam Dolu Atık Kutusu Olduğu Depo Noktası ve Kapasite Kısıtı Olan Problem İçin Elde Edilen En İyi Rotalar

Araç diğer turlarda çoğunlukla birer atık noktasını ziyaret etmektedir. Araca ait kapasite 3 tam dolu atık depo çöpünü alacak duruma getirildiğinde atık noktaları diğer turlarda da yer almaktadır. Başka bir deyişle, yığılma durumu biraz kaybolmaktadır, fakat yine de yeterli değildir. Yığılmaların karşılaşılmadığı durum aracın kapasitesinin 4 tam dolu atık deposuna yükseltildiği durumdur.

1. tur: 0 14 0	6. tur: 0 9 1 4 7 0
2. tur: 0 3 16 0	7. tur: 0 8 2 0
3. tur: 0 13 17 5 0	8. tur: 0 6 18 12 0
4. tur: 0 11 10 0	9. tur: 0 6 15 0
5. tur: 0 8 19 0	10.tur: 0 20 0

Şekil-6: Araca ait Kapasitenin 4 Tam Dolu Atık Kutusu Olduğu Depo Noktası ve Kapasite Kısıtı Olan Problem İçin Elde Edilen En İyi Rotalar

5. SONUÇ

Bu çalışmada araç rotalama problemlerinin çözümü için genetik algoritma kullanılmıştır. Problemin çözümünde elde edilen sonuçlara göre genetik algoritmanın bu problemler için optimale yakın sonuçlar verdiği söylenebilir.

Gerçek hayatta bu çalışmada ele alınan problemlerin sayısız versiyonu bulunmaktadır. Problemlere ait girdiler değiştirilerek veya ek kısıtlar getirilerek karşılaşılan problemlerin çözümünde ele aldığımız problemlerin örnek oluşturabileceği düşünülmektedir.

Gelecekte yapılacak bir çalışmada, atık noktalarına verilen günlük dolum oranları stokastik olarak tahmin edilebilir. Günlük dolum oranları dinamik bir yapıya sahip olabilir. Bu durumun atık noktalarının boşaltılması problemine daha gerçekçi bir yaklaşım olduğu açıktır. Bizim ele aldığımız depo noktası ve kapasite kısıtı olmayan araç rotalama problemi atık noktalarının boşaltılması problemidir. Ayrıca bu problem, aracın periyot başında ve sonunda uğradığı bir depo bulunacak şekilde düzenlenebilir. Bu probleme kapasitenin aşılması ihtimali gibi bir sorunun olmadığı ve toplanan ve/veya dağıtılan ürünlerin uzun ömürlü olduğu bir problem örnek olarak verilebilir.

KAYNAKÇA

[1] Assad A. A., (1988), "Modeling and Implementation Issues In Vehicle Routing", Vehicle Routing: Methods and Studies, North Holland, pp.7-45.

[2] Toth P. ve D.Vigo, (2002),The Vehicle Routing Problem, Siam Pub., pp.27-186.

[3] Emel, G. G. ve Ç. Taşkın , (2002), "Genetik Algoritmalar ve Uygulama Alanları", Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 21(1), pp.129-152.

[4] Karaboğa, D., (2004), Yapay Zeka Optimizasyon Algoritmaları, Atlas Yay.

[5] Jones, T. ve G.J.E. Rawlins, (1993), "Reverse Hillclimbing, Genetic Algorithms and The Busy Beare Problem", Proceedings of 5th. International Conference on Genetic Algorithms, Morgan Kaufmann pub., pp.121-140.

[6] Kuo, T. ve S. Hwang, (1993), "A Genetic Algorithm With Disruptive Solution", Proceedings of 5th. International Conference on Genetic Algorithms, Morgan Kaufman pub., pp.154 – 168.

[7] Michalewicz, Z., (1992), Genetic Algorithms + Data Structures = Evolution Programs, Springer - Verlag.

[8] Man, K.F., K.S. Tang ve S. Kwong., (1999), Genetic Algorithms Concepts and Designs, Springer.

TAKTİKSEL EN KISA YOL PROBLEMLERİNİN GENETİK ALGORİTMA YAKLAŞIMI İLE ÇÖZÜMÜ

Ferhat UÇAN[(a)], Gülay Büyükaksoy KAPLAN[(a)], Reha ÇAPUTÇU[(a)], Mehmet HAKLIDIR[(a)], Ömer Faruk ARAR [(a)]

(a) TÜBİTAK MARMARA ARAŞTIRMA MERKEZİ, Bilişim Teknolojileri Enstitüsü, 41470,Gebze, Kocaeli

{ferhat.ucan, gulay.kaplan, mehmet.haklıdır, reha.caputcu, omer.arar}@bte.mam.gov.tr

ÖZ

Askeri birimlerin savaş sırasında intikallerini en kısa sürede ve en az zararla tamamlamaları hayati öneme sahiptir. Bu çalışmada, iki senaryo kapsamında tanımlanan problemlerin genetik algoritma ile çözümü ele alınmıştır. Birinci senaryoda birimlerin izleyeceği en kısa yolun bulunmasında, genetik algoritmanın değişik parametrelerinin çözüm üzerine etkisi incelenmiştir. İkinci senaryoda ise çatışma olasılığının bulunduğu bir ortamda, birimlerin ilerlemesi sırasında çatışma haberlerinin alınması durumunda, en kısa yolun dinamik olarak genetik algoritma ile bulunması ele alınmıştır.

Anahtar Kelimeler: Askeri Simülasyon, En Kısa Yol Problemi, Genetik Algoritmalar, Komuta Kontrol, Taktiksel Rota Planlama.

THE SOLUTION OF TACTICAL SHORTEST PATH PROBLEMS USING GENETIC ALGORITHM APPROACH

ABSTRACT

During combat, transfers of military units in the shortest period and with least damage are vitally important. In this study, shortest path problems, described in two scenarios, have been solved using genetic algorithm approach. In the first scenario, the impact of the different parameters of genetic algorithm on the solution has been investigated for finding the shortest path the units follows. In the second scenario, in an environment with possibility of combat, the aim is to find the shortest path by employing genetic algorithm approach dynamically, in case any combat information arrives during the transfer of units.

Keywords: Command Control, Genetic Algorithms, Military Simulation, Shortest Path Problem, Tactical Route Planning

1. GİRİŞ

Çeşitli konumlarında engeller bulunan bir ortamda, bir başlangıç noktasından bir hedef noktasına minimum mesafeli hareketin yapılması şeklinde tanımlanan en kısa yol problemi, birçok alanda değişik şekillerde karşılaşılan bir problemdir. Askeri alanda birliklerin en kısa zamanda ve güvenli bir şekilde istenilen yere intikali de bu değişik şekilllerden biridir. Bu çalışmada, genetik algoritma yaklaşımı kullanılarak en kısa yol problemleri çözülmeye çalışılmıştır. Bildirinin ikinci ve üçüncü bölümünde sırasıyla en kısa yol problemi ve genetik algoritmalar kısaca tanıtılmıştır. Dördüncü bölümde genetik algoritmanın taktiksel en kısa yol problemine uygulanması, beşinci bölümde gerçekleştirilen simülasyonlar anlatılmış, altıncı bölümde sonuçlar değerlendirilmiştir.

2. EN KISA YOL PROBLEMİ

Harp simülasyonlarında bilinçli olarak hareket etme kabiliyetine sahip sanal kuvvetler gittikçe daha yaygın bir şekilde kullanılmaktadır. Bu sanal kuvvetlerin gerçeğe en yakın şekilde modellenmesi için çeşitli alanlarda çalışmalar sürdürülmektedir. Bu çalışma alanlarından bir tanesi de sanal ortamların oluşturulması ve sanal kuvvetlerin bu ortamlarda yollarını bulma problemidir. Yol bulma problemi sanal ortamda iki nokta arasındaki intikali temsil etmektedir ve uygulandığı alana göre çeşitli kısıtlar içerebilmektedir. İki nokta arasındaki intikalin; en az eğimli yolu kullanarak, en az yakıtı sarf ederek, en az ekipmanla veya en kısa mesafeli yolu kullanarak yapılması istenebilir. Uygulamalarda çoğunlukla mesafe kısıtı ile karşılaşılmaktadır. En kısa mesafeye sahip yolu bulabilmek en kısa yol problemi olarak isimlendirilmektedir.

En kısa yol problemi, çeşitli konumlarında engeller bulunan bir ortam üzerinde, belli bir başlangıç noktasından, belli bir hedef noktasına minimum mesafeli hareketin yapılması şeklinde tanımlanmaktadır. Bu problem ile farklı alanlarda farklı şekillerde karşılaşılmaktadır. Bilgisayar ağlarında, özellikle enerji kısıtı olan kablosuz sensör ağlarında, enerji korunumunu sağlamak amacıyla, veri paketlerinin en az sensör kullanılarak hedef noktaya taşınması gerekmektedir. Bu da en kısa mesafeli rotanın bulunması ile mümkündür. Robotların verilen göreve bağlı olarak en kısa yol ile hedefe ulaşması en kısa yol probleminin çözümü ile mümkün olmaktadır. Harp simülasyonlarında özellikle sanal kuvvetlerin üç boyutlu ortamda belli bir hedef noktaya en kısa yolla ulaşımı, en kısa yol probleminin çözümünü gerektirmektedir. Strateji oyunları, sanal karakterlerin hareketlerinde ve belli hedef noktalara ulaşmalarında en kısa yol probleminin çözümüne ihtiyaç duymaktadır.

3. GENETİK ALGORİTMALAR

Genetik algoritma evrimsel hesaplama tekniğinin bir parçası, evrimsel hesaplama ise yapay zekânın bir dalıdır. Herhangi bir problemin genetik algoritma ile çözümü, mümkün olan çözümlemeleri sanal olarak evrimden geçirerek yapılmaktadır [1].

3.1 Genetik Algoritmanın Aşamaları

Başlangıçta, n adet kromozom içeren popülasyon oluşturulur. Her x kromozomunun çözüme yakınlığı f(x) değerlendirilir. Yeni popülasyon oluşuncaya kadar izleyen adımlar tekrar edilir: İki ebeveyn kromozomlarının uyumluluğuna göre seçilir; Yeni bireyler elde etmek amacıyla, ebeveynler bir çaprazlama olasılığına göre çaprazlanır;

Yeni bireylerin genlerinin mutasyon olasılığına göre kromozom içindeki konumu değiştirilir; Yeni bireyler popülasyona eklenir. Eğer yeni popülasyonda sonuç tatmin edici ise algoritma sona erdirilir ve son popülasyon çözüm olarak sunulur. Genetik Algoritmanın standart bir sonlanma kriteri bulunmamaktadır. Sonucun yeterince iyi olması ya da yakınsamanın elde edilmesi algoritmanın durması için kriter olarak kabul edilebilir [1].

Genetik algoritmaların en kısa yol problemine uygulandığı bazı alanlar:

- Ağ dolaşımı [2] : Ağ üzerinden iletilen veri paketlerinin en az gecikme ile hedeflerine varması amaçlanmıştır.
- Taşımacılık problemleri [3] : Taşımacılık alanında maliyeti azaltmak probleminin çözümü araştırılmıştır.
- Mobil robot yol planlaması[4] : Mobil bir robotun içinde hareket edeceği alan için, bir çarpışmadan sakınma modeli oluşturulmuştur
- WDM (Wavelength Division Multiplexing-Dalgaboyu Bölmeli Çoğullama) [5] : WDM optik ağlarında Multicast (aynı anda pek çok hedefe) verinin ağda en kısa yolu izlemesi amaçlanmıştır.
- Hava Araçlarına Ait Sensörlerin Görev Planlaması ve Dinamik Kaynak Ayırımı [6] : Havacılıkta kullanılan sensörlerin hazırlık ve birbirlerine uzaklık durumlarına göre en verimli şekilde kullanılması için çalışma yapılmıştır.
- Fabrika İnşaatlarında Hammadde/ Ürün İstasyonların Yerlerinin Belirlenmesi [7] : Hammadde işleme maliyetlerinin azaltılması amaçlanmıştır.

4. GENETİK ALGORİTMANIN TAKTİKSEL EN KISA YOL PROBLEMİNE UYGULANMASI

Bu çalışmada genetik algoritmanın en kısa yol probleminin çözümünü bulma amacıyla kullanılmasında, her birey başlangıç noktasından bitiş noktasına olan bir rotayı göstermektedir. Kromozomun ilk geni başlangıç noktası, son geni de bitiş noktasını göstermektedir. Kromozomların gen sayısı değişkendir, fakat graftaki toplam düğüm sayısından fazla olması mümkün değildir. Her kromozom, rotada üzerinden geçilen düğümlerin numaralarından oluşan bir tamsayı dizisi şeklindedir. Bu çalışma kapsamında kullanılan genetik algoritma operatörleri, basit genetik algoritma operatörlerinden farklıdır [8, 9].

Yukarıda anlatılan özellikteki bireylere sahip ilk popülasyon, rastlantısal olarak oluşturulur. Popülasyonun birey sayısı nesiller boyunca sabit kalan ve genetik algoritmanın çalışmasına etki eden önemli bir faktördür. İlk popülasyondaki bireyler rastlantısal olarak üretildiğinde herhangi bir veya birden fazla düğüm üzerinde çevrimler oluşabilir. Kullanılan genetik algoritma yaklaşımında bireylerde oluşan çevrimler temizlenmektedir, çünkü bu çevrimler oluştuğu anda kaldırılmazlarsa, çaprazlama sonucu bozuk bireylerin oluşmasına neden olabilirler [8]. Popülasyondaki birey sayısı, genetik algoritmanın çalışma hızına ve bulduğu sonucun gerçek çözüme yakınlığına etki edecek çok önemli bir parametredir. Simülasyon aşamasında bu parametrenin değeri değiştirilerek genetik algoritmanın başarımı üzerindeki etkisi

belirlenmiştir. İlk popülasyon oluşturulduktan sonra bireylerin sonuca ne derece yakın olduklarını belirlemek için uygunluk (fitness) fonksiyonu kullanılır. Böylece potansiyel çözüm adayları değerlendirilerek iyilik derecesi bulunmuş olur. Yol bulma probleminde kullanılan değerlendirme fonksiyonu yolun toplam uzunluğu ile ters orantılıdır. Bir bireyin ifade ettiği rota ne kadar kısa ise uygunluk değeri o kadar yüksek olacaktır.

Nesil üretme döngüsünde ebeveynleri ikililer halinde grupladıktan sonra çaprazlama ve mutasyon ile çocuk bireyler oluşturulur. Uygunluk değeri yüksek olan bireyler yaşatılırken, diğer bireyler yok edilir. Böylece genetik algoritma nesiller ilerledikçe daha iyi bireyler üretmeye ve sonuca yakınsamaya başlar. Ebeveyn çiftleri oluşturulurken art arda olan iki birey aynı gruba atanmıştır. Her ikili grup için çaprazlama yapılarak iki yeni çocuk birey elde edilir. Bu çalışmada kullanılan çaprazlama yöntemi, klasik çaprazlama yöntemlerinden farklıdır. Bu yöntem iki kromozom arasında rotanın bir bölümünün takas edilmesi esasına dayanır. Bunun için çaprazlama yapılacak iki bireye ilişkin rotaların ortak noktalarından herhangi biri seçilir ve ilk çocuk için, seçilen noktaya kadar olan rota ilk ebeveynden, seçilen noktadan sonraki rota ikinci ebeveynden alınır. İkinci çocuk birey için ise ortak noktaya kadar olan rota ikinci ebeveynden, ortak noktadan sonraki rota ilk ebeveynden alınır. Böylece iki yeni rotayı temsil eden iki çocuk birey elde edilmiş olur. Ortak noktası bulunmayan iki rotaya ilişkin kromozomlar çaprazlanamaz. Başlangıç populasyonunda rotalarının bir bölümü ortak olan bireyler üretilir, her düğümün en az iki bireyde bulunması garanti edildiğinden bireyler çaprazlama operatörünün uygulanmasına uygun olarak seçilmiştir. Çaprazlama sonucu oluşan çocuk bireyler bazı çevrimler içerebilir, bu çevrimler düzeltme fonksiyonu ile kaldırılır [8]. Bu çevrimler temizlendikten sonra seçilen mutasyon oranına göre mutasyon operatörü çocuk bireylere uygulanarak, gen değişimi ve popülasyonda çeşitlilik sağlanır. Kullanılan mutasyon yöntemine göre rastgele seçilen bir noktadan sonraki rota değiştirilerek alternatif bir rota ile hedefe gidilmektedir. Popülasyonda tercih edilen mutasyon oranı da genetik algoritmanın değiştirilebilen bir parametresidir [8]. Mutasyon oranı [0.1..0.5] aralığında farklı değerler ile denenmiştir. Mutasyon oranı sıfıra çok yakın seçilirse, yeni bireyler çaprazlama veya kopyalama sonrasında olduğu gibi kalır. Eğer mutasyon olur ise yeni bireyin bir kısmı değiştirilmiş olur. Eğer bu oran %100 olursa kuşak içindeki bireyler tamamen değişir, %0 olursa hiç değişmeden kalır. Oluşan yeni çözümlerin önceki çözümü kopyalamasını önlemek ve sonuca daha hızlı ulaşmak amacıyla bu problemin çözümünde mutasyon oranı 0.5 civarında seçilmiştir. Fakat [0.1..0.5] aralığında daha düşük mutasyon oranlarında da genetik algoritma doğru sonuca yakınsamaktadır.

İki ebeveynden çaprazlama ve mutasyon operatörleri ile farklı gen sırasına sahip iki çocuk birey elde edilir. Genetik algoritmanın yaşam döngüsünde hangi bireylerin yaşayacağı ve gen havuzuna seçileceğini belirleyebilmek için, çocuk bireylerin uygunluk değeri hesap edilir. Daha sonra turnuva seçim yöntemi uygulanarak çocukların uygunluk değerleri ile ebeveynlerin uygunluk değerleri karşılaştırılır. Çocuklar ebeveynlerinden daha yüksek uygunluk değerlerine sahip iseler yaşatılır ve gen havuzuna seçilerek genetik algoritmanın sonraki nesillerinde kullanılırlar, aksi halde çocuklar gen havuzuna alınmaz ve önceki ebeveynler genetik algoritmanın sonraki neslinde kullanılırlar. Genetik algoritma, bu döngüdeki işlemleri (çaprazlama, mutasyon, iyi bireylerin seçimi) sonlanma kriteri sağlanana kadar devam ettirir.

Sonlanma kriteri, genetik algoritmanın sonuca ulaşma süresini ve sonuca yakınsanmasını etkileyen önemli bir parametredir. Bu kriter, belli bir nesil üretme sayısı olabileceği gibi bireylerin belli bir oranda birbirine benzeşmesi veya neslin en başarılı bireyinin bir önceki neslin en başarılı bireyine eşit olması, yani uygunluk değeri açısından iyileşme kaydedilememesi gibi bir koşul da olabilir. Simülasyon aşamasında farklı sonlanma kriterleri kullanılarak, sonlanma kriterinin genetik algoritmanın çalışması üzerindeki etkisi belirlenmiştir. Sonuçta, sonlanma kriteri sağlanıncaya kadar ortaya çıkan en yüksek uygunluk değerli bireyin ifade ettiği rota çözümü verir.

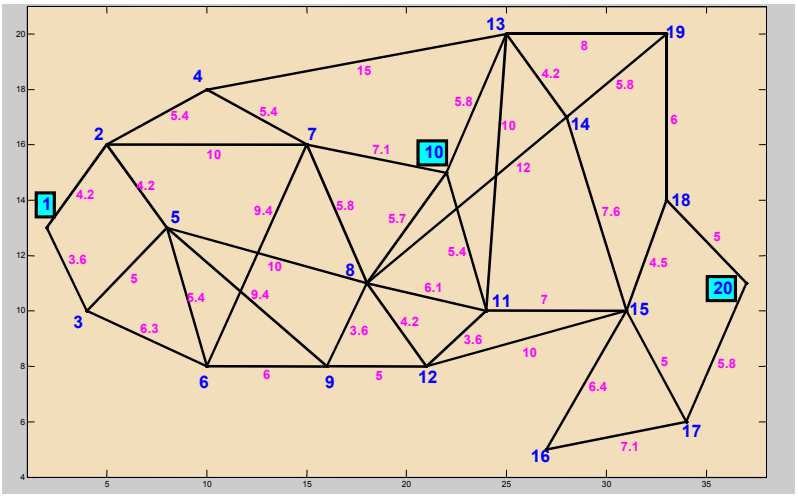
Genetik algoritmanın en önemli üç parametresi çaprazlama oranı, mutasyon oranı ve populasyondaki birey sayısıdır. Çaprazlama ve mutasyon oranı farklı değerlerle denemeler yapılarak en uygun değerleri bulunmultur. Populasyondaki birey sayısı da genetik algoritmanın önemli bir parametresidir. Eğer kromozom sayısı az olursa GA çözüm aranan uzayın ancak bir kısmını gezebilir ve çaprazlama için fazla bir seçeneği yoktur. Kromozom sayısı çok fazla olursa GA çok yavaş çalışır. Kromozom sayısı ağdaki düğüm sayısına göre belirlenebilmektedir.

5. SİMÜLASYON

Bu çalışmada bir önceki bölümde çalışma biçimi anlatılan genetik algoritma yaklaşımı, iki askeri senaryo dahilindeki en kısa ve en güvenli kısa yol problemlerinin çözülmesinde kullanılmıştır.

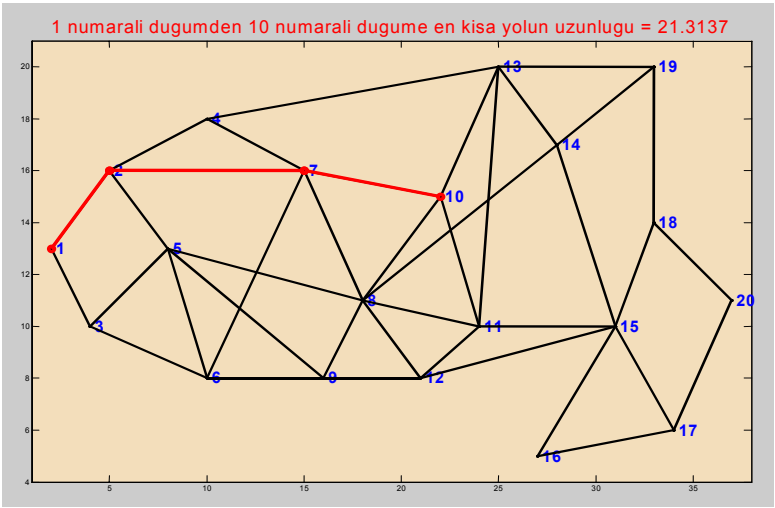
5.1. Senaryo-1

Harekat öncesi topçu taburunun ilk toplanma alanından, taktik toplanma alanına en kısa yolu kullanarak intikal etmesi planlanmıştır. (Tüm taburlar harekat öncesi taktik toplanma alanında bir araya gelerek intikallerini gerçekleştirirler.) İntikalle ilgili yol planı Şekil 1'de verilmektedir. Burada 1 no'lu düğüm başlangıç noktasını, 10 no'lu düğüm taktik toplanma noktasını göstermektedir. Düğümler arasındaki uzaklıklar, görsel karmaşa yaratılmaması için yalnızca Şekil 1'de gösterilmiştir. Diğer şekiller için de aynı uzaklıklar geçerlidir. Simülasyon kapsamında, komuta kontrol birimi bu en kısa yolu belirleyebilmek için ilgili bir algoritmaya ihtiyaç duymaktadır.



Şekil 1 Senaryo-1'e ilişkin intikal yol planı

İlk olarak Senaryo-1 göz önüne alınmış ve genetik algoritma ile en kısa yolun bulunmasında birey sayısının seçiminin sonuca etkisi analiz edilmiştir. Literatürde yaygın olarak kullanılmakta olan Dijkstra ve Floyd algoritmaları [10] ile bulunan ilk toplanma alanı ile taktik toplanma alanı arasındaki en kısa yol Şekil 2'de kırmızı renkli segmentlerle gösterilmiştir.



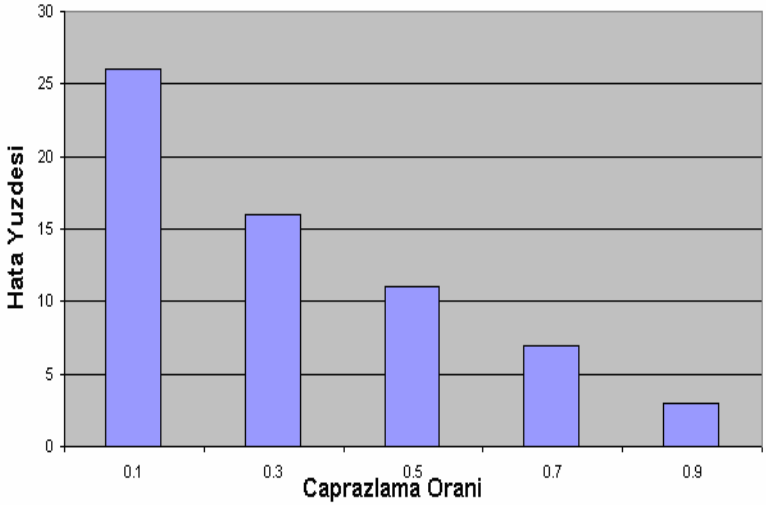
Şekil 2 İntikale ilişkin en kısa yol

Gen havuzundaki birey sayısının genetik algoritmanın çalışmasına etkileri analiz edilmiştir. Genetik algoritmanın başarımı birey sayısıyla orantılı olarak artmaktadır. Birey sayısı fazla seçildiğinde hem gen çeşitliliği fazla olmakta, hem de sonuca ulaşma adım sayısı (nesil sayısı) azalmaktadır. Birey sayısı 10 olarak seçildiği durum için genetik algoritma ile bulunan yol uzunluğu: 25,19; birey sayısı 20 olarak seçildiği

durum için: 22,084; birey sayısı 30 olarak seçildiği durum için ise 21,31dir. Gen havuzundaki birey sayısı 30 olarak seçildiğinde genetik algoritma her çalışmasında diğer algoritmalarla (Floyd, Dijkstra) aynı sonucu bulabilmiştir.

Genetik algoritmanın sonlanma kriterinin belirlenmesi de çalışmasını etkileyen önemli bir parametredir. Genetik algoritma üç nesil çalıştırıldığında başlangıç noktasından taktik toplanma alanı arasındaki yolun uzunluğu 24,34 olarak bulunmuştur. Genetik algoritmanın sonlanma kriteri olarak bireylerin benzerliği kullanıldığı zaman elde edilen sonuç daha iyi olmaktadır. Çünkü genetik algoritma bireyler birbirine benzeştiğinde sonuca yakınlaşmış demektir, ama bunun kaç nesilde yakalanabileceğini önceden kestirmek güçtür, ayrıca bu ilk popülasyondaki bireylerin çözüme yakınlığına, problemin karmaşıklığına ve düğüm sayısına da bağlıdır. Bu problem için yapılan tüm simülasyonlarda bireylerin benzerliği sonlanma kriteri olarak kullanıldığında tam sonuca ulaşılmıştır.

Çaprazlama operatörünün genetik algoritmanın çalışmasına etkileri de simülasyonlarla analiz edilmiştir. Genetik algoritmanın ebeveynler arası çaprazlama oranı yükseldikçe başarım artmaktadır. Çünkü çaprazlama ile gen çeşitliliğine sahip değişik bireyler bulunmakta ve sonuca ulaşma adım sayısı azalmaktadır. Çaprazlama oranının değişimine ilişkin grafik Şekil 3'te gösterildiği gibidir.



Şekil 3. Çaprazlama Operatörü Etkisi

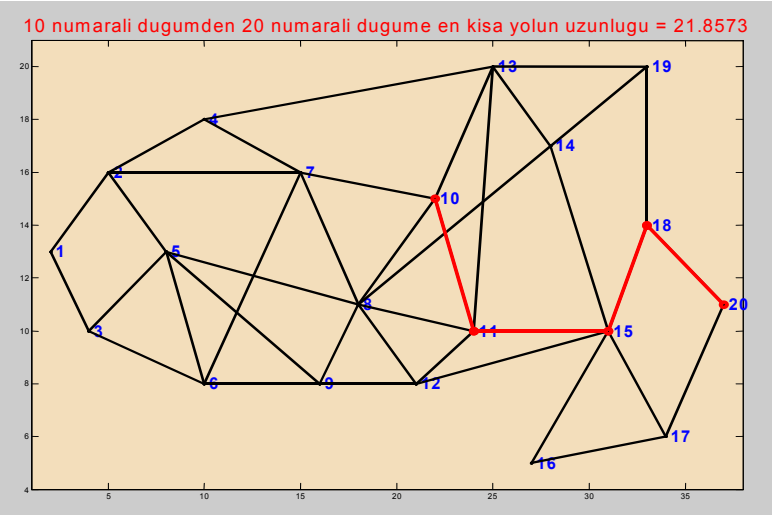
5.2. Senaryo-2

Topçu taburu taktik toplanma alanına intikal ettikten sonra, çatışmanın başladığı bilgisini alır ve Tugay'a destek vermek için en kısa zamanda çatışma alanına intikal etmesi gerekmektedir. Çatışma alanı Şekil 1'de 20 no'lu düğüm ile gösterilmiştir. Topçu taburunun intikalini en kısa zamanda ve mümkün olan en az kayıpla tamamlayabilmesi için en güvenilir güzergahı kullanması gerekmektedir. Çatışmanın gidişatının süratle değişmesi, hangi güzergahların açık ya da kapalı olduğunu, çatışmanın nerelerde hızlanıp yavaşladığını ve güzergah üzerindeki köprü, geçit gibi

intikali sağlayan yapıların durumlarını önceden tahmin edilmesini engellemektedir. Kurulan telsiz bağlantıları ile bu durum bilgileri sürekli güncellenmekte, bu bilgileri kullanan tabur, intikal için kullanacağı güzergahı bu bilgi akışı doğrultusunda değiştirmektedir. Simülasyon kapsamında, komuta kontrol birimi bu bilgileri kullanarak güvenilir taktiksel rota planını belirleyebilmek için genetik algoritma yaklaşımını kullanmıştır.

Taktik toplanma bölgesinden (10 no'lu düğüm) çatışma noktasına (20 no'lu düğüm) intikal için Floyd ve Dijkstra algoritmaları ile bulunan en kısa yol Şekil 4'teki gibidir. Segment kopma ve güvenlik değeri değişimi gibi durumlarda, Floyd ve Dijkstra gibi literatürde iyi bilinen statik en kısa yol algoritmaları graf topolojisindeki en ufak bir değişimde çözüme ve matris hesabına en baştan başladığından kullanışlı değildir. Floyd ve Dijkstra sabit ağ yapısında etkin olarak çalışan algoritmalardır. Ancak en kısa yol problemi gerçek zaman kısıtları altında çalıştığında ve segmentler dinamik olarak değiştiğinde Floyd ve Dijkstra algoritmaları etkin olarak kullanılamaz [11].

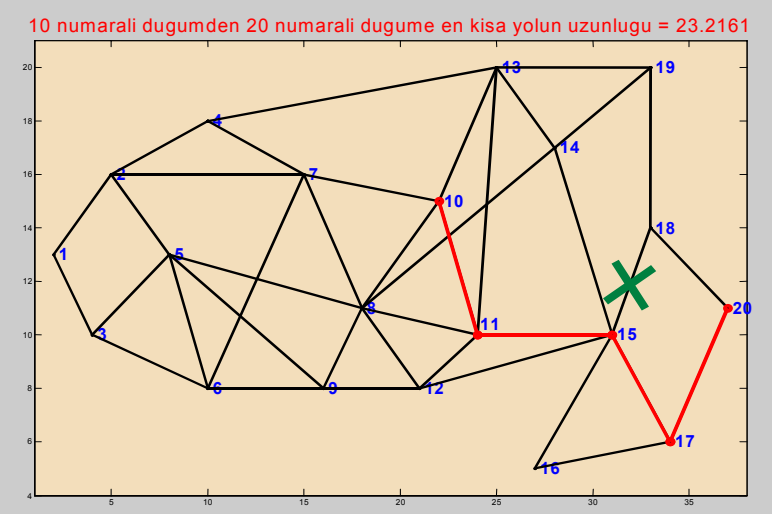
Segmentlerdeki çatışmalar zaman içerisinde değişmektedir, bazı güzergahlarzamanla kullanılamaz hale gelebilmektedir. 15. ve 18. düğümler arası segment kullanılamaz hale geldiği duruma ilişkin genetik algoritma çözümü Şekil 5'te gösterilmiştir. Genetik algoritma başlangıçta tüm yollar açık olduğu durumda çalıştırılmaya başlanmış 10. düğümden 11. düğüme ulaştığında, 15. ve 18. düğümler arası bağlantı kopmuştur (Şekil 5'te çarpı işareti ile belirtilmiştir). Bu durumda genetik algoritma alternatif bir yol üreterek, çatışmanın şiddetinin az olduğu segmentlerden en kısa şekilde nasıl hedefe ulaşılabileceğini bulmuştur.



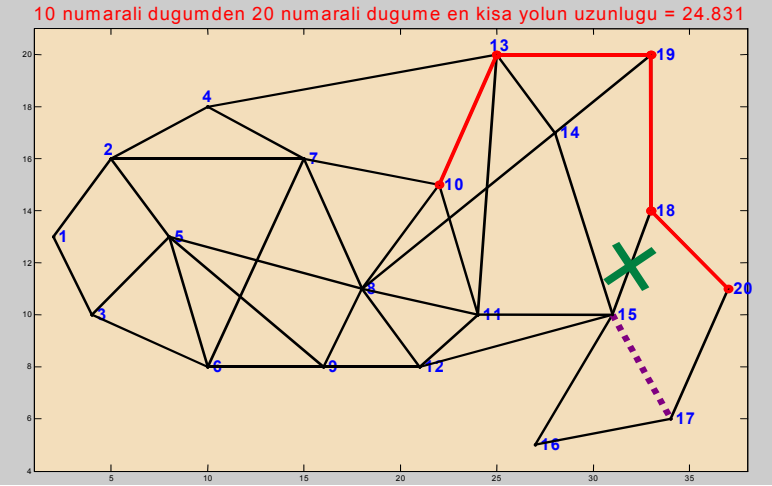
Şekil 4. Taktik Toplanma Noktasından Çatışma Noktasına İntikal (Floyd ve Dijkstra Çözümleri)

Genetik algoritmanın, problemin çözümüne getirdiği yenilik sadece segmentlerin koptuğu durum için geçerli değildir. Segmentler tamamen kopmasa bile üzerlerindeki çatışmanın şiddeti arttıkça genetik algoritma daha güvenli segmentlerden geçerek en efektif yolu bulmayı amaçlar. Genetik algoritmanın uygunluk fonksiyonu, güzergahların

hem güvenlik değerine, hem de uzunluğuna bağlıdır. 15. ve 18. düğümler arası bağlantı kopukken, 15. ve 17. düğümler arası çatışma şiddeti artar (Şekil 5'de kesikli çizgi ile belirtilmiştir), 18. ve 19. segmentler arası çatışma şiddeti azalır ise genetik algoritma Şekil 6'da görüldüğü gibi dinamik davranış değişikliğine cevap verecek şekilde yeni bir yol üretir. Genetik algoritma çatışma şiddeti artan yolları tercih etmeme eğiliminde, alternatif çözümler üretebilmektedir.



Şekil 5. Genetik Algoritma Çözümü (Segment Kopma Durumu)



Şekil 6. Genetik Algoritma Çözümü (Segment Güvenlikleri Değişme Durumu)

6. SONUÇLAR

Bu çalışmada en kısa yol probleminin askeri alandaki iki özel hali, genetik algoritma kullanılarak çözülmüştür. Birinci problemde, tanımlanmış bir alanda topçu taburunun en kısa sürede taktik toplanma noktasına ulaşma problemi genetik algoritmanın farklı parametreleri kullanılarak çözülmüş ve parametrelerin çözüm üzerindeki etkisi incelenmiştir. Gerçekleştirilen simülasyonlarda başlangıç popülasyonundaki birey sayısındaki artışın, sonucun bulunmasına olumlu katkıda bulunduğu gözlenmiştir. Simülasyonlarda etkisi incelenen bir diğer kriter de sonlanma kriteridir. Sonlanma kriteri olarak bireylerin benzerliğinin kullanılması nesil sayısının kullanılmasına göre daha iyi sonuç vermiştir. Ebeveynler arası çaprazlama oranının artmasının başarımı arttıran diğer bir faktör olduğu simülasyon sonuçlarından gözlenmiştir. İkinci problem literatürde dinamik en kısa yol problemi olarak tanımlanan probleme karşı düşmektedir. Burada topçu taburunun, taktik toplanma alanından çatışma alanına giderken izlediği güzergah üzerinde çatışma durumlarını göz önüne alması gerekmektedir. Ayrıca çatışma sonucu yolun bazı kısımlarının kullanılamaz hale gelmesi durumunda dinamik olarak izleyeceği güzergahı güncellemesi söz konusudur. Bu problemin çözümünde genetik algoritmanın kullandığı uygunluk fonksiyonu, hem yollardaki çatışma durumlarını hem de güzergah uzunluğunu hesaba katmaktadır.

KAYNAKÇA

[1] Kurt M., Semetay C., "Genetik Algoritma Ve Uygulama Alanları" , Mühendis ve Makina Dergisi, Ekim 2001

[2] Munemoto M., Takai Y., Sato Y., "An Intelligent Network Routing Algorithm by a Genetic Algorithm", Proceedings of the Fourth International Conference on Neural Information Processing, pp.547-550, 1997

[3] http://portal.acm.org/citation.cfm?id=1143184.1143201

[4] Xin D., Hua-hua C., Wei-kang G.," Neural Network and Genetic Algorithm based global path planning in a static environment", Journal of Zhejiang University SCIENCEISSN 1009-3095, Oct. 7, 2004

[5] Siregar J.H., Zhang Y., Takagi H.," Optimal Multicast Routing Using Genetic Algorithm for WDM Optical Networks", IEICE Tran. Commun, Vol.E88–B, No.1, 2005

[6] Soliday S.W., "A Genetic Algorithm Model for Mission Planning and Dynamic Resource Allocation of Airborne Sensors", Raytheon Systems Company Sensors and Electronic Systems, 10.03.1999

[7] Arapoglu R.A., Norman B.A., Smith A.E., "A Genetic Algorithm Approach to Input/Output Station Location in Facilities Design", Proceedings of the Second Japan-Australia Joint Workshop on Intelligent and Evolutionary Systems, November 1998, 155-170.

[8] Ahn CH, Ramakhrisna R.S, "A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations", IEEE Trans Evolutionary Computation, Vol.6, No.6, 2002, 566-579.

[9] Wu W, Ruan Q, "AGene-Constrained Genetic Algorithm for Solving Shortest Path", IEEE ICSP'04 Proceddings, 2510-2513.

[10] Skiena SS, The Algorithm Design Model, 1998, Springer Verlag.

[11] Sudip Misra, B. John Oommen, "Dynamic algorithms for the shortest path routing problem: learning automata-based solutions". IEEE Transactions on Systems Man, and Cybernetics, Part B 35(6): 1179-1192 (2005)

A5 OTURUMU:

HLA ALTYAPILARI-II

KABAN: SİMULASYON VERİLERİ GEÇERLEME VE OTOMATİK KULLANICI ARABİRİMİ OLUŞTURMA ALTYAPISI (69)
İbrahim DEMİR, Zahir TEZCAN, M. Nedim ALPDEMİR

FEDERASYON BAŞARIM TEST ORTAMI (41)
Cüneyd FIRAT, Murat AÇIKGÖZ

DAĞITIK BENZETİM SİSTEMLERİ İÇİN PERFORMANS ANALİZİ YÖNTEMİNİN GELİŞTİRİLMESİ (15)
Ramazan CENGİZ, F. Erdoğan SEVİLGEN

DAĞITIK BENZETİM UYGULAMALARINDA YENİDEN KULLANILABİLİRLİK, VERİ DAĞITIMI VE ZAMAN YÖNETİMİ (55)
Ersin ÜNSAL

KABAN : SİMULASYON VERİLERİ GEÇERLEME VE OTOMATİK KULLANICI ARABİRİMİ OLUŞTURMA ALTYAPISI

İbrahim DEMİR, Zahir TEZCAN, M. Nedim ALPDEMİR

TÜBİTAK UEKAE / İLTAREN, Tunalı Hilmi Cad. Binnaz Sok. Kat:2 No:3, Kavaklıdere ANKARA,

{ibrahimd, zahirt, nedima}@iltaren.tubitak.gov.tr

ÖZ

Veri kalitesi ve tutarlılığını hedefleyen uygulamalarda, işlenecek verilerin sistematik ve etkin bir geçerleme sürecine tabi tutulması vazgeçilmez bir önkoşuldur. Bu gerçek simulasyon uygulamaları için daha da geçerlidir, çünkü veri ve model arabirim analiz ve geçerleme işlemi, simulasyon geçerleme, doğrulama ve test (GD&T) sürecinin önemli bir bileşenini oluşturmaktadır. Basit veri yapılarının, aralık değer kontrolu, format kontrolu gibi basit geçerlemelerini gerçekleştirmek nispeten basit bir işlem olsa da, birden fazla model arabirimini kapsayan, ve farklı veri parçacıkları arasında semantik bağımlılıklar içeren karmaşık ve hiyerarşik veri yapılarının geçerleme işlemi zaman ve efor gerektiren bir sürece dönüşmektedir. Bu nedenle, karmaşık geçerleme işlemlerini otomatik olarak gerçekleştirmeyi destekleyen veri geçerleme altyapıları, simulasyon uygulamaları geliştiricileri tarafından ihtiyaç duyulan ve istenen bir özelliktir. Bu makalede bir veri geçerleme ve otomatik Kullanıcı Arabirimi (KAB) oluşturma altyapısı olan ve karmaşık veri geçerleme kurallarının dinamik olarak KAB'inde uygulanmasını destekleyen KABAN'ı tanıtmayı amaçlıyoruz. KABAN karmaşık geçerleme kurallarının ve bunların uygulanacağı veri yapılarının XML dökümanlarında tanımlanmasını ve koşum zamanında kural ihlallerinin kullanıcı-dostu bir görsel yaklaşımla KAB'inde ilgili veri parçacıklarına uygulanmasını sağlar. KABAN'ın ayırdedici üç özelliği; kural tanımlama şemasının ifade gücü, olay-tabanlı dinamik kural uygulama mekanizması ve kullanıcı arabirimleri ile sağladığı etkin ve kullanıcı-dostu entegrasyon yaklaşımı olarak sıralanabilir.

Anahtar Kelimeler: Veri Geçerleme, Otomatik Kullanıcı Arabirimi Oluşturma, Simulasyon, Yazılım Altyapısı.

KABAN : SIMULATION DATA VALIDATION AND AUTOMATIC USER INTERFACE GENERATION INFRASTRUCTURE

ABSTRACT

Effective and systematic validation of input data is an essential prerequisite for applications aiming at ensuring the quality and homogeneity of information to be processed. This is even more strikingly true in the simulation domain as data and model interface analysis and validation is an important part of simulation model validation, verification and testing (VV&T) procedures. Although simple validation of primitive data types, such as range checking, or format checking, can be performed relatively easily, validation of complex data types containing semantic dependency relationships between different parts of complex data hierarchies spanning multiple simulation model interfaces can be a time consuming effort. Supporting such complex validation tasks via validation frameworks in an automated way is therefore a desirable feature for simulation application developers. In this paper we introduce KABAN; a data validation and UI generation framework that supports automatic and just-in-time imposition of sophisticated validation rules. KABAN allows for declarative specification of complex validation rules in an XML document, which are applied to a UI data entry form where rule violations are dynamically presented to the user through an intutitive visual mapping onto the set of invalid data items. The strength of KABAN lies along three aspects: the expressive power of its rule definition schema, the event based dynamic rule imposition mechanism, and its effective and user-oriented integration with user interfaces.

Keywords: Data Validation, Automatic UI Generation, Modelling, Simulation, Software Framework.

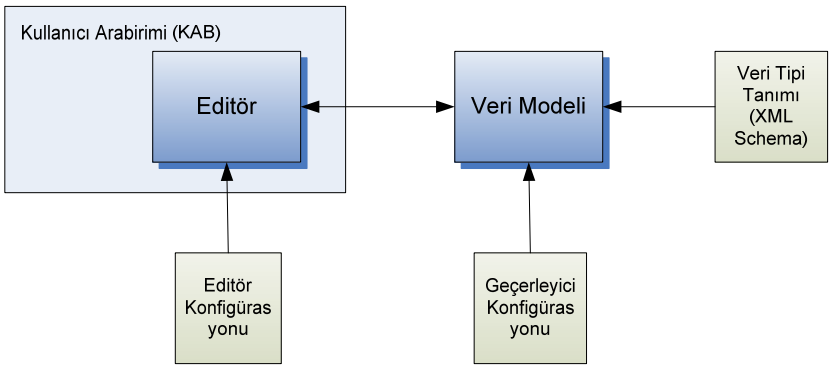
1. GİRİŞ

Veri kalitesi ve tutarlılığını hedefleyen uygulamalarda, işlenecek verilerin sistematik ve etkin bir geçerleme sürecine tabi tutulmasının gerekli olduğu bilinen bir gerçektir. Veri Geçerleme ihtiyacı bazı uygulama alanlarında daha da büyük önem kazanabilmektedir. Bu durum simulasyon uygulamaları için daha da geçerlidir, çünkü veri analiz ve geçerleme işlemi, simulasyon geçerleme, doğrulama ve test (GD&T) sürecinin önemli bir bileşenini oluşturmaktadır [4]. Özellikle kullanıcıdan elde edilen verilerin geçerlemesinin, veri işleme sürecinin mümkün olduğunca erken bir aşamasında yapılabilmesi, böylece kullanıcıya veri tutarsızlıkları konusundaki geri beslemelerin mümkün olduğunca erken iletilebilmesi için, kullanıcı ara birimlerindeki veri giriş formlarının geçerleme yeteneğiyle donatılmış olması ayrı bir önem taşımaktadır. Veri yapılarının, aralık değer kontrolu, format kontrolu gibi basit geçerlemelerini gerçekleştirmek nispeten basit bir işlem olsa da, birden fazla model arabirimini kapsayan, ve farklı veri parçacıkları arasında semantik bağımlılıklar içeren karmaşık ve hiyerarşik veri yapılarının geçerleme işlemi zaman ve efor gerektiren bir sürece dönüşmektedir. Bu nedenle, karmaşık geçerleme işlemlerini otomatik olarak gerçekleştirmeyi destekleyen veri geçerleme altyapıları, simulasyon uygulamaları geliştiricileri tarafından ihtiyaç duyulan ve istenen bir özelliktir.

Yukarıda sözü edilen ihtiyaçları karşılamaya yönelik olarak program geliştiricilerin kullanımı amacıyla değişik özelliklerdeki veri geçerleme kütüphaneleri geliştirilmiş bulunmaktadır. Örneğin Commons Validator [1], DataSift [2] ve iScreen [3] bunlardan birkaçı olarak sıralanabilir. Bu tür veri geçerleme kütüphanelerinin ortak yönleri olmakla beraber, her birinin öne çıkardığı farklı bir özellik de olabilmektedir. Örneğin iScreeen, Kullanıcı arabiriminden bağımsızlığı ve basitliği, DataSift genişletilebilirlik ve esnekliği ayırdedici özellik olarak öne çıkarmaktadırlar. Bu makalede tanıttığımız KABAN'ın ayırdedici üç özelliği ise; kural tanımlama şemasının ifade gücü, olay-tabanlı dinamik kural uygulama mekanizması ve kullanıcı arabirimleri ile sağladığı etkin ve kullanıcı-dostu entegrasyon yaklaşımı olarak sıralanabilir. Kural tanımlama şemasının ifade yetenekleri ve hiyerarşik yapısı sayesinde basit geçerleme işlemlerinin yanında farklı veri alanları arasında şartlı bağımlılıklar kurmak, daha da ötesinde bu bağımlılık ifadelerine karmaşık aritmetik ve mantıksal birimler yerleştirmek mümkün olmaktadır. Dolayısıyla geçerleme işlemi sentaktik veri tipi kontrollerinin yanında semantik bir boyut kazanmaktadır.

Bildirinin geri kalanı aşağıdaki gibi düzenlenmiştir: Üçüncü bölümde KABAN'ın ana bileşenleri ve yapısı tanıtılmaktadır. Dördüncü bölümde kural tanımlama şemasının yapısı anlatılmakta, beşinci bölümde geçerleme yeteneklerinden örnekler verilmekte ve bu kuralların kullanıcı açısından görsel olarak nasıl bir etkiye yol açtıkları gösterilmektedir. Altıncı bölümde ise alınan dersler ve sonuçlar yer almaktadır.

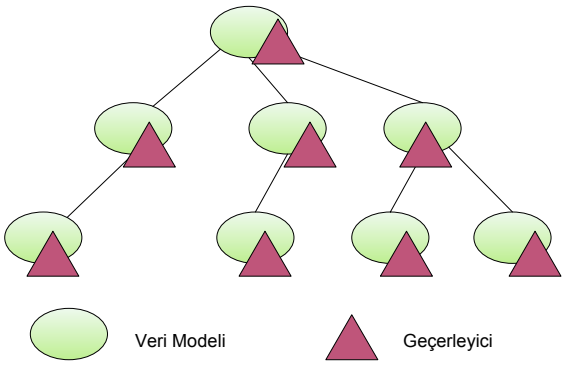
2. KABAN'IN ANA BİLEŞENLERİ



Şekil 1. KABAN'ın bileşenleri

Kaban sisteminde XML Schema' da tanımlı veri tiplerinden türetilen veri yapıları `Veri Modeli` tipindeki nesnelerin içerisinde saklanır. Bir Veri Modeli bir Editörle eşleşir. Her Editör bir Editör konfigürasyonu, her Veri Modeli ise bir Geçerleyici Konfigürasyonu ile eşleşir. Geçerleyici Konfigürasyonu dosyasında hangi veri tipi için nasıl bir geçerleme kuralının uygulanacağı bilgisi yer almaktadır. Bir nesneye Veri Modeli bağlandığında o veri tipi ve her alt veri alanı için bir Veri Modeli nesnesi

oluşturulur. Başka bir deyişle hiyerarşik bir tarzada Veri Modeli ağacı oluşur. Bir editöre Veri Modeli bağlandığı zaman, o editörün alt-editörlerine, her editör tipine uygun bir alt Veri Modeli bağlanarak Şekil 2'de ifade edilen yapı ortaya çıkar. Veri Modeli ağacındaki her Veri Modeli için geçerleyici belirtilmesi zorunlu değildir. Geçerleyici tanımlanmamış veri modelleri otomatik olarak toplam geçerleme işleminin dışında bırakılırlar.



Şekil 2. Veri Modeli ve Geçerleyici hiyerarşisi

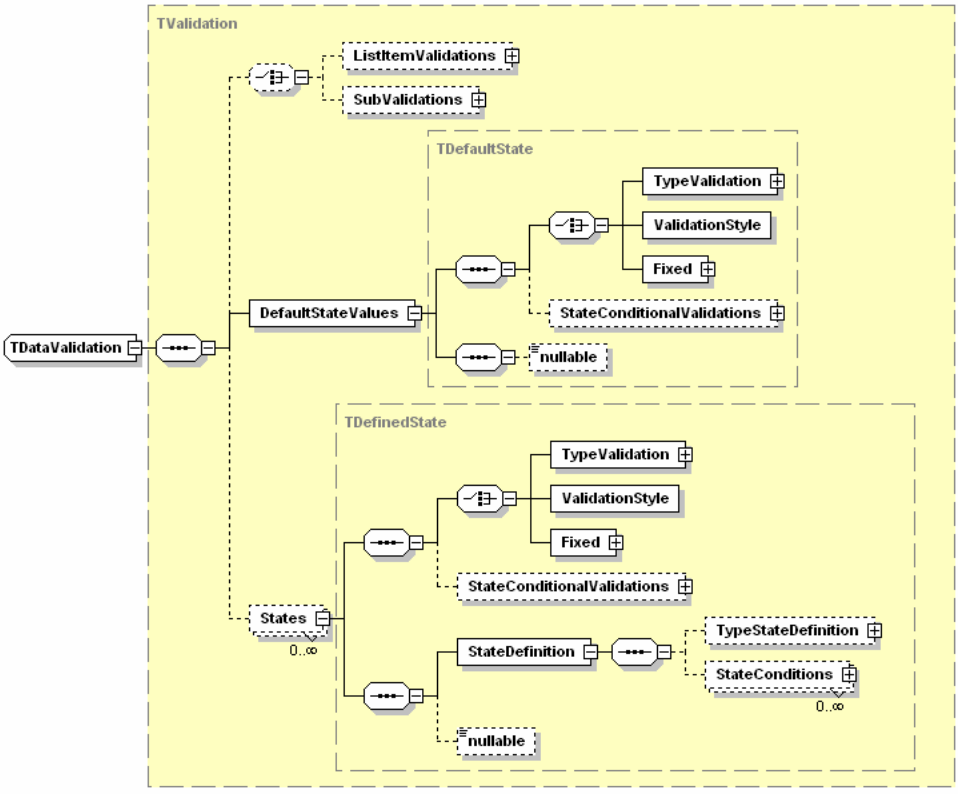
2.1 Geçerleme Kurallarının Uygulanma Mantığı

Veri Modeli muhafaza ettiği verilere bir değişiklik yapıldığında ilgili güncelleme (update) olaylarını oluşturur. Bu olaylara kaydolan (subscription) editörler bunları tespit ederler ve kendilerini güncellerler. Veri Modeline bir geçerleme kuralı bağlıysa veri alanlarındaki değer değişikliği ile beraber geçerleme işlemi başlatılır. Geçerleme işlemi başka güncellemelere yol açarak ilgili güncelleme olaylarını da tetikleyebilir. Editorler Veri Modeli üzerinden bu güncelleme mesajlarına ulaşıp hata mesajlarını gösterirler. Kaban geçerleme kuralları tanımlama şeması birbirine bağımlı geçerleme tanımlamalarına olanak tanır. Kaban sistemi bu bağımlıkları kendi içerisinde muhafaza eder ve bağımlılığı olan Veri Modelinin değeri değiştirildiğinde bağımlı Veri Modeli'in geçerleme kuraları da tetiklenir. Bütün bağımlı işlemlerin bitmesi beklenir ve güncelleme olayları bu işlemler sonlandıktan sonra oluşturulur.

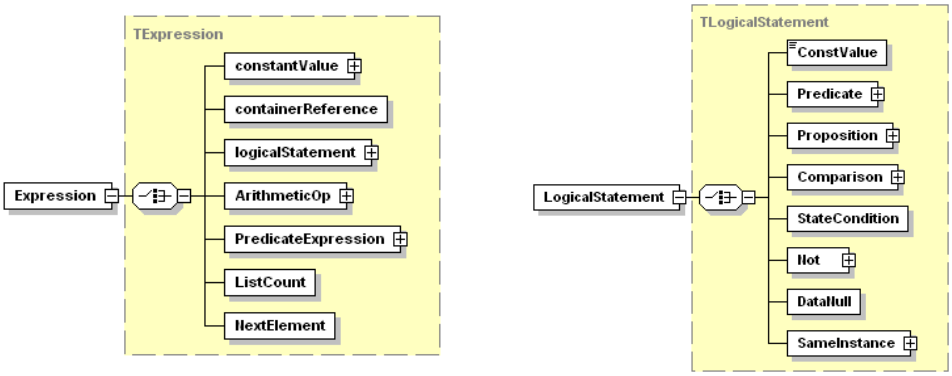
3. KABAN KURAL TANIMLAMA ŞEMASI

KABAN altyapısında kurallar, mantıksal ve/veya cebirsel ifadelerden oluşan önermelerin de yer alabildiği bir yapı içinde tanımlanır. İfadeler içinde yer alan birimlerin bazıları kendi içinde tekrar ifade üniteleri kullanabilir. Mesela bir aritmetik işlemin her bir ögesi başka bir ifade (expression) olabilir. Başka bir deyişle özyineli (recursive) bir ifade gramerine destek vardır. Diğer önemli bir bileşen de Veri Modellerinin durumu (state) ile ilgili tanımlamalara olanak veren yapıdır. Bu yapı sayesinde bir Veri Modeli için farklı koşullarda geçerli olabilecek farklı durum değerleri tanımlanabilmekte böylece o veri modelinin kendisi için belli şartlarda geçerli kabul ettiği durumların listesi tutulabilmektedir. Durum tanımlamaları geçerleme motorunun

işleyişi açısından önem arzetmekte, her veri modelinin verili bir durum için geçerli olup olmadığı, ya da belli bir anda durumunun ne olduğu bu yapılar sayesinde motor tarafından yönetilebilmektedir. KABAN'ın kural tanımlama şeması Şekil 3'te basitleştirilmiş olarak verilmiştir. Şekilde görülen grafik gösterim, XML Spy adlı bir XML aracı tarafından oluşturulmuştur. Diktörtgen kutucuklar *XML element* yapılarını göstermekte, ağaç benzeri gösterim *element*'ler arası hiyerarşik ilişkileri ortaya koymaktadır.



Şekil 3. KABAN Kural tanımlama XML Şemasının basitleştirilmiş hali



Şekil 4. İfade Sentaksını tanımlayan XML Şema yapıları

3.1 İfade Çeşitleri

Kaban geçerleme ifadeleri bir XML dökümanında *XML element* olarak tanımlanırlar. Başka bir deyişle kural grameri üst düzeyde *XML Schema grameri* [6] ile kısıtlıdır. İfade tanımlama dili daha alt düzeyde, ifadelerin sentaktik yapısını tanımlayan XML Şema tanımlamalarıyla kısıtlıdır. Bu XML Şema tanımlamaları Şekil 4'te verilmiş olup Şekil 3'te verilen ana şemanın bir parçasıdırlar.

Bu XML şemasında tanımlanabilecek kurallarda yer alan ifade tipleri aşağıda ve listelenmiştir.

- **Sabit Değer(Constant Value):** Belli bir veri alanı için sabit değer belirtmek için kullanılır.
- **Barındırıcı Referans(Container Reference):** VeriModeli ağacı üzerinde başka bir Veri Modeli belirtmek için kullanılır. Belirtilen Veri Modeli üzerinde bağıl olarak belirtilir. Dosya path sistemi kullanılır. (örnek: "konum/x" gibi)

- **Mantıksal Tümce (Logical Statement):** true veya false dönen hesaplanmış ifade. LogicalStatement aşağıda daha detaylı açıklanacaktır.

- **Cebirsel İşlem (Arithmetic Operation):** bölme, çarpma, toplama, çıkarma, mod, vb.. işlemleri yapılabilir.
- **İstatistiksel İfade (Statistical Expression):** Liste Veri Modeli üzerinde gezinmeye olanak tanır. *Sum, Min, Max* gibi operasyon tipleri vardır. *Sum* Liste üzerinde gezinip her iterasyonda çıkan değeri toplayıp ifade olarak toplamı döner. Min liste içersinde gezinip her iterasyonda yapılacak tanımlanmış expression ifadelerinden minimum olanını döner.
- **Liste Eleman Sayısı (List Count):** Bir listenin eleman sayısını döner.

- **Önerme (Proposition):** *AND, OR, All* (tümü) ve *Any (herhangi biri)* gibi mantıksal işlemlerin mantıksal alt ifadeler üzerinde uygulanmasına olanak tanır. Bunlardan All ve Any işlemleri Liste üzerinde iterasyon yapar ve işlem tipine göre değer döner. Mesela Any seçilmişse iterasyon yaparken herhangi bir "true" dönen expression hesapladığında işlem "true" döner. AND ve OR işlemleri ise en az iki mantıksal alt ifadeye uygulanmak üzere tanımlanabilirler.
- **Karşılaştırma (Comparison):** Karşılaştırma işlemlerine olanak tanır. (<, >, <=, ==, v.b.)
- **Durum Sorgulama (State Inquiry):** Herhangi bir Veri Modelinin durumunu sorgular.
- **Değil(Not): I**nverse operasyonu tanımlar.
- **Boş Veri(Null Data):** Belitilen Veri Modelinin veri alanının boş olup olmadığını kontrol eder.

4. GEÇERLEME ÖRNEKLERİ

KABAN sisteminde her veri modeli tipi için (int, double, string vs.) bir çok geçerleme çeşidi vardır. KABAN altyapısının daha iyi anlaşılması için bir kaç örnek aşağıda anlatılmıştır.

```
<Expression>
      <ArithmeticOp ArithmeticOperationType="subtract">
         <item>
            <containerReference containerReference="..\SinifKapasitesi"/>
         </item>
         <item>
            <ListCount containerReference="..\OgrenciListesi"/>
         </item>
      </ArithmeticOp>
</Expression>
```

Yukarıdaki XML verisinde KABAN matematiksel ifade örneği gösterilmiştir. İfade bir çıkarma işleminden oluşmaktadır. Sınıf kapasitesi değerinden öğrenci listesinin uzunluğu çıkarılarak sınıftaki boş yer sayısı hesaplanmaktadır. XML verisi içindeki *containerReference* özelliği (attribute) ile gösterilmiş olan "`..\SinifKapasitesi`" değeri ile sınıf kapasitesi değerine ulaşılmaktadır. Bu değere ulaşılırken KABAN geçerleme motoru bu ifadenin geçtiği yerin bir yukarısındaki veri modeline ulaşıp bulunduğu veri modelindeki *SinifKapasitesi* adlı alt parametreye ulaşır. Bu yöntem bilgisayarınızdaki dosya konumlarını belirtmek için kullanılan yöntem ile aynıdır. KABAN göreli (relative) dosya ulaşım yöntemi kullanılmasının kullanıcının aşina olduğu bir yöntem olmasının yanında bir faydası da hazırlanan geçerleme XML verisinin başka veri modelleri içinde kullanılabilmesine olanak sağlamasıdır. *ListCount* elementi ile belirtilmiş olan ifadenin değeri *ListCount* elementi içersinde *containerReference* ile belirtilmiş olan veri modelinin eleman sayısıdır.

```
<Number>
   <NumberRange>
      <Max inclusive="true">
         <containerReference containerReference="..\Ucret"/>
      </Max>
   </NumberRange>
</Number>
```

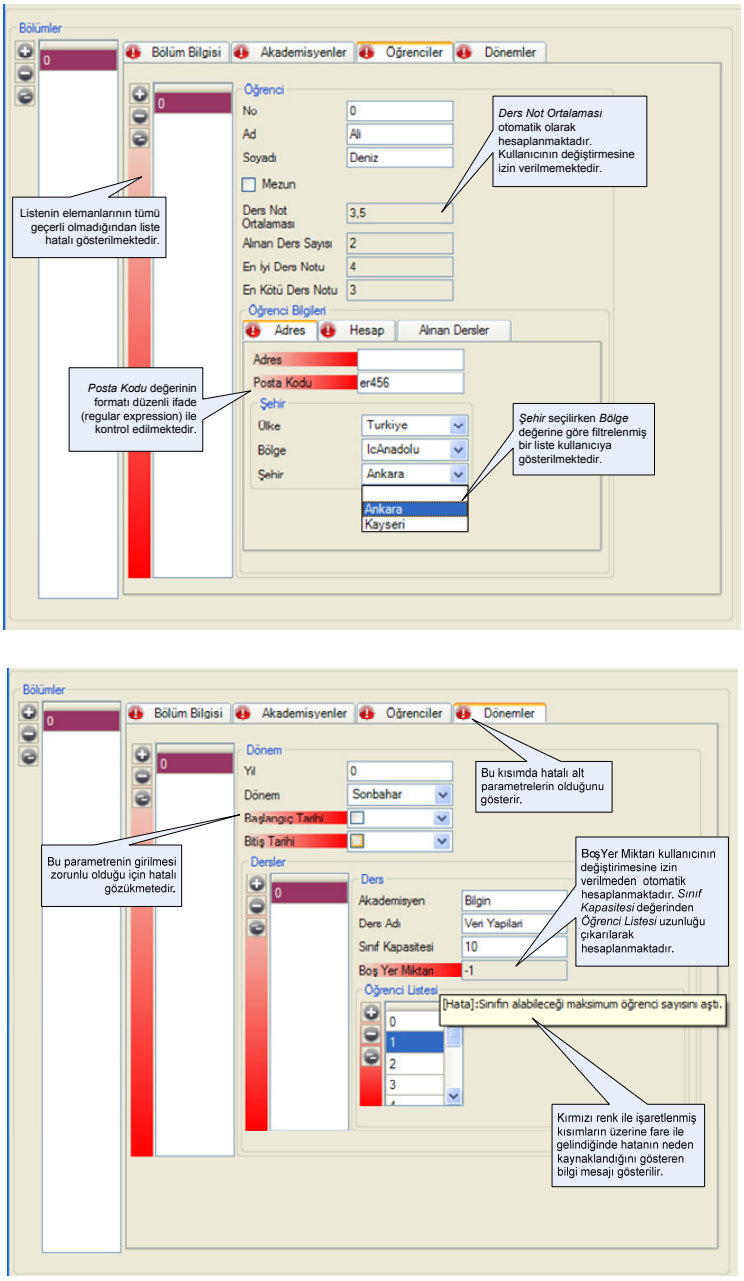
Yukarıdaki ifade örneğinde ise sayısal bir alana üst sınır koyulmaktadır. Üst sınır olarak sabit bir değer koyulabildiği gibi bu örnekte görüldüğü üzere üst sınır olarak başka bir veri modelinin değeri de verilebilir. Bu örnekteki XML bir öğrencinin o ana kadar ödemiş olduğu para miktarını belirten parametrenin geçerleme kodundan alnmıştır. Ödenen miktarın ücretten fazla olmaması için üst sınır koyulmuştur.

```
<Proposition BinaryLogicalType="OR">
      <item>
         <Comparison operationType="eq">
            <lhs>
              <ListCount containerReference="..\StudentInfo\CoursesTaken"/>
            </lhs>
            <rhs>
               <constantValue>
                   <double>0</double>
               </constantValue>
            </rhs>
         </Comparison>
      </item>
      <item>
         <DataNull containerReference="..\StudentInfo\CoursesTaken"/>
      </item>
</Proposition>
```

Yukarıdaki ifadede bir önerme gösterilmiştir. Öğrencinin aldığı ders sayısı 0'a eşitse veya öğrencinin aldığı dersleri muhafaza eden veri modeli boş ise ifade doğrudur. Bu örnekte mantıksal ifadeler iç içe kullanılmışltır. *OR* (veya) önermesinin birinci elemanı bir karşılaştırma ifadesidir. KABAN motoru bu karşılaştırmayı hesaplayıp cevabını *OR* önermesinin birinci girdisi yapar. *OR* önermesinin ikinci girdisi de veri modeline has bir ifadedir. Veri modeli boş ise *doğru* olan bir ifadedir.

```
<String>
   <RegEx>
      <regex>[0-9]{6,6}</regex>
   </RegEx>
</String>
```

Yukarıdaki ifadede *Posta Kodu* parametresinin formatı belirtilmiştir. Formatı düzenli ifadeler (regular expression) kullanılarak belirtilmiştir. Belirtilen format posta kodunun 6 adet rakamdan oluşmasını sağlamaktadır. Yukarıda sıralan örneklerin görsel olarak kullanıcıya nasıl yansıdığına ilişkin arayüz örnekleri Şekil 5'te verilmiştir. Arayüzlerde geçerleme kurallarının uygulanması neticesinde renklendirilmiş olan bölgeler ve bu bölgelere ilişkin açıklamalar görülebilmektedir.



Şekil 5. Arayüz Örnekleri

5. SONUÇ

Bu bildiride her türlü uygulamada kullanılabilecek olan bir veri geçerleme ve KAB oluşturma altyapısı olan KABAN tanıtılmıştır. KABAN altyapısı kurumumuz bünyesinde yürütülmekte olan büyük bir simulasyon projesinde kullanıcı arayüzlerindeki veri tiplerinin tanımlanması, veri giriş formlarının otomatik olarak oluşturulması ve girilen verilerin geçerlenmesi amacıyla başarıyla kullanılmıştır. Özellikle, Şartlı (conditional) ve karmaşık bağımlılık ilişkilerinin tanımlanmasına verdiği destek, istendiğinde KAB'inden bağımsız bir geçerleme altyapısı sunması ve istenildiğinde de KAB ile kolayca entegre olabilmesi gibi özellikleriyle öne çıkmaktadır..

Deneyimler sırasında gözlenen bir gerçeği ifade etmek gerekirse, geçerleme kurallarının XML formatında tanımlanması dolayısıyla, kuralların karmaşıklık düzeyi arttıkça kural tanımlayıcının yükü doğal olarak önemli ölçüde artmaktadır. Bütün kural tanımlama ve uygulama sistemlerinde belli düzeylerde geçerli olan bu zorluğun, geçerleme kurallarının uygulama koduna ve KAB koduna bağımlı olmayan bir formda sadece bir kez tanımlanması dolayısıyla kazanılan kod idame avantajıyla bir ölçüde dengelendiği söylenebilir.

KAYNAKÇA

[1] Commons Validator, http://jakarta.apacheçorg/commons/validator, Apache Software Foundation, 2006

[2] DataSift: Data Validation and Transformation Framework, http://www.datasift.org, The Auel Project, 2004

[3] Shallman, D., i-screeen: The Java Object Validation Framework, http://i-screen.org/, 2006

[4] Balcı, O.,"Verification, Validation, and Testing",in Handbook of Simulation, Eds. Jerry Banks, John Wiley & Sons, 1998

[5] Petrakos, G.A. and Farmakis, G.E., "A Declarative Approach To Data Validation Of Statistical Data Sets, Based On Metadata", http://citeseer.ist.psu.edu/ petrakos00declarative.html, 2000

[6] XML Schema Part 0: Primer, W3C Recommendation, Second Edition, http://www.w3.org/XML/Schema, 28 October 2004

FEDERASYON BAŞARIM TEST ORTAMI

Dr. Cüneyd FIRAT, Murat AÇIKGÖZ

C2Tech

TÜBİTAK TEKSEB No: C-210 Gebze-KOCAELİ

[cuneyd.firat;murat.acikgoz]@ctech.com.tr

ÖZ

Kapsamlı ve yüksek çözünürlüklü modellerin söz konusu olduğu HLA federasyonlarında veri ve zaman başarımı sistem mimarisi açısından kritiklik arz etmektedir. Bu tür simülasyon sistemlerinin tasarımı aşamasında veri yoğunlukları ile modeller bazında zaman performans isterlerinin analiz edilmesi ve federasyon mimarisinin bu analiz sonuçlarına göre belirlenmesi doğru olacaktır. Böylece hem sistem başarımı hem de maliyet açısından kazanımlar elde edilecektir. Bu bildiride federasyon mimarilerinin veri ve zaman başarımı açısından test edilebilmesine olanak sağlayan RTI başarım test ortamı anlatılmaktadır. Söz konusu ortam IEEE 1516 uyumlu farklı RTI ürünlerinin başarım karşılaştırması amacıyla da kullanılabilecektir. Bu ortam farklı federasyon mimarileri ve veri yapılarını tanımlama ve elde edilen sonuçları kaydetme işlevlerine sahiptir. Bu kapsamda geliştirilen ortam tanıtılmakta ve pRTI™ kullanılarak yapılan örnek analiz çalışmasının sonuçları da verilmektedir.

Anahtar Kelimeler: Başarım, Federasyon Mimarisi, HLA, RTI, Test

FEDERATION PERFORMANCE TEST ENVIRONMENT

ABSTRACT

The data and time performance of HLA federations containing high-resolution and high amount of models is critical in system architecture point of view. The design time performance testing based federation configuration will provide advantages for both satisfying the required data throughput and timing constraints of the simulation and decrease the cost risks. In this paper such an environment for RTI and federation architecture performance testing is introduced. This environment is capable of running any IEEE 1516 compliant RTI product and provides necessary functions allowing federation and data configuration management. It has also the performance recording functions. Example results generated by using pRTI™ are presented in the paper.

Keywords: Federation Architecture, HLA, Performance, RTI, Test.

1. GİRİŞ

Simülasyon sistemleri için dağıtık yapı vazgeçilmez isterlerden bir tanesi olmuştur. Geçen süre içinde dağıtık simülasyon yapılarını destekleyen süreç ve standartlar (ALSP, DIS, CORBA, HLA) yeni gereksinimleri de göz önüne alarak oldukça standart ve esnek bir yapıya kavuşmuştur. Ancak gelinen noktada dağıtık simülasyon mimarisine ilişkin özellikle veri ve zaman başarımı açısından bir çok soru tasarımcılar açısından cevapsız kalabilmektedir. Uygulamaların çeşitliliği ile donanım ve yazılım teknolojilerindeki gelişmelere paralel olarak simülasyon uygulamalarının kapsamı ve kullanılan modellerin gerçekliği dramatik derecelerde artmıştır. Örneğin bazı çalışmalarda bir ülkedeki tüm internet topolojisi ve trafiğinin modellenmesi kadar geniş kapsamlı modelleme ve simülasyon çalışmaları yapılmaktadır [1]. Öte yandan özellikle mühendislik analizi kapsamında geliştirilen (örn. dalga yayılım modelleri) modellerin gerçeklik düzeyi ve algoritmik karmaşıklıkları bu modellerin çok hızlı ve yoğun şekilde veri paylaşımı yapmasına yol açmaktadır. Bu gibi veri paylaşımının yoğun olduğu ve özellikle gerçek zaman ya da gerçek zamandan hızlı koşması istenen simülasyon uygulamalarında federasyon yapısı ve dağıtık yapı mimarisinin optimizasyonu gerekmektedir. Bu optimizasyon için önemli unsurlardan bir tanesi de başarım test ve analizlerinin yapılabilmesidir. Aksi takdirde beklenmedik sonuçlar ile karşılaşmak olasılığı mevcuttur [2].

DMSO tarafından geliştirilen ilk RTI sürümünden bu yana performans analizleri için bir çok çalışma yapılmıştır [3][4][5][6]. Bu analizlerin bir kısmı yeni tekniklerin denenmesi diğer bölümü ise var olan RTI sürümlerinin farklı şartlar altında test edilmesi amacıyla yapılmıştır. Ancak yapılan çalışmalar standart bir test yaklaşımını ve ortamını kullanmadığı için ortaya çıkan sonuçlar her çalışmanın kendi varsayımları ile sınırlı kalmıştır.

Geçtiğimiz dönemde HLA federasyon tanımlamaları ve koşum işlevleri IEEE tarafından ele alınarak IEEE 1516 serisi altında standart bir yapıya kavuşturulmuştur [7] . Ancak standart performans özelliklerini içermemekte ve özellikle RTI işlevleri ve arayüzleri açısından çeşitliliğe de elvermektedir. Aynı standarda uygun mevcut RTI ürünleri arasında hem arayüzler hem de gerçekleme yaklaşımları açısından farklılıklar bulunmaktadır. Bu nedenle aynı federasyon mimarisini farklı RTI ürünleri ile uyumlulaştırmak için ek bir düzenlemeye gereksinim olmaktadır. Bununla birlikte farklı RTI ürünlerinin doğal olarak farklı başarımları da olmaktadır.

C2Tech'in yaptığı simülasyon geliştirme projelerinde tasarlanan federasyon yapılarının farklı veri yükü ve çeşitli koşullar altında başarım analizlerini yapabilmek için bir test ortamı geliştirilmiştir. Bu test ortamı IEEE 1516 uyumlu olan farklı RTI ürünlerini test edebilmeye olanak sağlamaktadır. Elde edilen sonuçlar bir takım olası darboğazları ortaya çıkarmakta ve hem federasyon yapısı hem de konuşlandırma (deployment) kararlarının optimize edilmesine katkıda bulunmaktadır. Öte yandan böyle bir analiz lisanslamayı federasyondaki federe sayısına göre yapan RTI ürünleri açısından maliyet optimizasyonunu da sağlamaktadır.

İkinci bölümde geliştirilen test ortamının motivasyonu ve isterleri ile daha önce bu konuda yapılmış olan çalışmalar ile karşılaştırması ortaya konulmuştur. Üçüncü

bölümde bu ortamın mimarisi anlatılmaktadır. Dördüncü bölümde pRTI™ [10] kullanılarak yapılan örnek testlerin sonuçları sunulmuştur. Sonuç bölümünde ise yorum ve öneriler yer almaktadır.

2. MOTİVASYON VE GEREKSİNİMLER

Federasyon tasarımının başlıca adımları;

- Nesne modellerinin tanımlanması,
- Federe sayısı ve yapılarının ortaya konulması

olarak karşımıza çıkmaktadır. Nesne modelleri tasarımının kaynağı modellenen görev uzayında yer alan varlık ve işlevlerin çözümü ile aralarındaki ilişkilerin niteliği olmaktadır. Nesne modelleri arasındaki etkileşim ve veri paylaşım gereksinimleri de model yapısının oluşturulmasını etkileyen önemli faktörlerdendir.

Federe sayısı ve bunların yapıları ise yazılım tasarımı, konuşlanma mimarisi ve kullanılacak olan RTI ürününün lisanslama yöntemi ile ilişkili olarak belirlenmektedir. Federe sayısı ya da kullanılan işlemci sayısı ile sınırlandırılan lisanslama yaklaşımları mimari tasarımı etkilemektedir.

Gerek nesne model yapısının gerekse de federe sayı ve yapılarının belirlenmesinde alınacak kararların performansa olan etkileri mimarinin gereksinimleri sağlayabilmesi açısından değerlendirilmelidir. Kullanılacak olan RTI ürününün ortaya koyulan model ve federe yapıları ile göstereceği başarım tüm sistemin başarımını sınırlandıran darboğazlardan bir tanesi olabilir. Federasyon tasarımının veri yapıları, nesne model yapısı ve federe sayıları itibariyle prototiplerinin oluşturulması ve çeşitli yükler altında testlerinin yapılması söz konusu tasarımın başarımını ortaya koyabilecektir. Test sonuçları RTI'ın kapasitesi ve zaman maliyetlerinin belirlenmesine olanak sağlayacaktır. Bu veriler ışığında federasyon tasarımı optimize edilebilecektir.

Bu çalışma kapsamında ortaya koyulan federasyon test ortamı federasyon prototiplerinin kolaylıkla oluşturulup koşturulabilmesine ve sonuçlarının gözlemlenerek analiz edilmesine olanak sağlamak üzere geliştirilmiştir. Test ortamının geliştirilmesinde dayanak olan başlıca isterler olarak;

- Herhangi bir kısıt olmadan istenilen sayıda federenin sisteme eklenebilmesine olanak sağlaması,
- Federasyon nesne modelindeki tüm nesne ve etkileşim sınıflarının listelenmesi ve seçilen sınıfın yayınlanmak ya da abone olunmak için federeye eklenmesine izin vermesi,
- Nesne sınıf örneklerine isim atanması ve federasyona atanmasına olanak sağlaması,
- Federe ve nesne modelleri zamanlama tiplerinin (time-constrained, time regulating) atanmasına olanak sağlaması,
- Gönderilen/alınan verilerin zamanlamalarının raporlanması,

[10] Pitch firması ürünü pRTI™

- Adımsal ilerleme veya belirli bir zaman aralığında sürekli koşum şeklinde denemeler yapılabilmesi,
- Ölçümlerin seri (single thread) veya paralel (asynchronous) yapılabilmesine olanak sağlaması,
- Reliable ve best-effort parametrelerinin seçilebilmesine olanak sağlaması,
- Öznitelik (attribute) ve parametrelere atanmak üzere standart veri yapıları oluşturulmasına ve kullanılması,
- Denemelerin tek veya birden fazla makine ile yapılabilmesi,
- Yapılacak testlerin oluşturulup koşturulabilmesi,
- IEEE 1516 uyumlu RTI ürünlerinin kolaylıkla entegre edilebilmesi,

sayılabilir. Yapılacak testler hem iki federe arasındaki başarımı hem de çoklu federe yapısı ve/veya sürekli koşum ile elde edilecek olan başarımı ortaya koymalıdır. Bu tür testlerde ölçülecek ve raporlanacak tipik ölçütler olarak;

- Abone olan Federe sayısı vs. verinin ortalama ulaşım hızı,
- Veri boyutu vs. iki federe arasındaki ulaşım hızı,
- Çeşitli servislerin kullanıldığı durumlarda (Time Management., DDM Management.) ölçümlerin yapılabilmesi,

sayılabilir.

Bu bildiride verilen örnek sonuçlar ise pRTI™ ile yapılan testlere aittir.

2.1 Diğer Çalışmalar

RTI performans analizi konusunda bilinen ve standartlaşmış bir değerlendirme yöntemi bulunmamaktadır. Ancak daha önce bu amaçla yapılan bazı çalışmalarda;

a. RID dosyasındaki ağ denetleme parametrelerinin en iyi başarımı verecek şekilde ayarlanması ve testi [8],
b. Belirli RTI ürünlerinin (örn. pRTI™) performans testleri için geliştirilen ve bu ürünlerin çeşitli federasyon koşulları ve ağ konfigürasyonu altında hem veri hem de zaman başarımını analizi [9] [10],
c. Yüksek hassasiyet ve kritiklik içeren gerçek zamanlı dağıtık simülasyon uygulamalarının performans analizi [6],
d. Farklı RTI ürünlerini çeşitli koşullar ve RTI servisleri altında testi ve analizine olanak sağlayan esnek test düzenekleri [11],
e. RTI ile diğer simülasyon arakatmanlarının karşılaştırılmasına dönük analizler [3],
f. Yüksek performanslı ağ alt-yapısında RTI performans analizi [4],

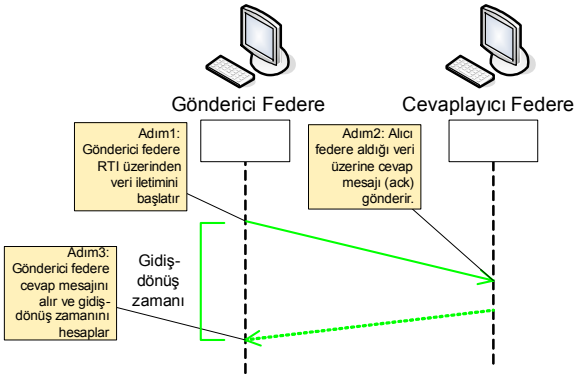
gerçekleştirilmiştir.

Geliştirilen test sistemi ise d maddesindekine benzer bir amacı hedeflemektedir. IEEE 1516 uyumlu farklı RTI ürünlerini değişken federe sayısı, değişken veri boyutları,

değişken ağ yapısı ve farklı RTI servisleri ile denemeye ve sonuçları kaydetmeye olanak sağlamaktadır. Bu esnek yapısıyla düzenek C2Tech tarafından gerçekleştirilen simülasyon projelerinde ve kullanılacak farklı RTI ürünleri ile başarımlarını tasarım aşamasında tespit edip maliyet-etkin bir federasyon yapısının oluşturulmasına imkan sağlamaktadır.

3. FEDERASYON TEST DÜZENEĞİ

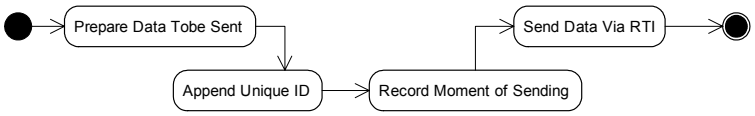
Düzenlenen test ortamında RTI üzerinden iletilen verinin hedef veya hedeflere ulaşım süresi temel ölçümdür. Yapılan ölçümlerde en tipik ölçüm federeler arasında gönderilen verinin gönderilme ile bu gönderiye verilen cevap mesajının alınması arasında geçen süredir. Test düzeneği de bu ölçümü yapacak şekilde düzenlenmiştir. Bu ölçümün elde edilebilmesi için 3 adımın gerçekleşmesi gerekmektedir. Bu adımlar Şekil 1'de gösterilmiştir.



Şekil 1. Test düzeneği

Adım1:

Test döngüsünün ilk adımıdır. Gönderici federe test amaçlı gönderilecek olan veriyi hazırlar. Bu verinin test döngüsünde geçerlilik kazanabilmesi için veriye test başlıkları eklenir. Tekil test numarası bu başlığın bir parçasıdır. Gönderim anı kayıt edildikten sonra veri RTI üzerinden gönderilir. RTI üzerinden gönderim 2 yolla olabilir. Bunlardan birisi nesne (Object Class Instance) oluşturup güncellemek, diğeri ise parametreli mesaj (Interaction Class) göndermektir. Bu adımın ayrıntılı akışı Şekil 2'de gösterilmiştir.



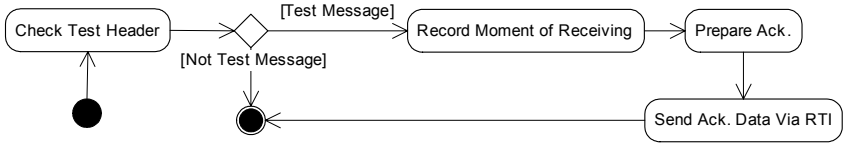
Şekil 2. Adım1 ayrıntılı akış diyagramı

Adım2:

Bu adım cevaplayıcı federenin test mesajı almasıyla başlar. Alınan mesajın test mesajı olması test başlığı taşımasına bakılarak kontrol edilir. Cevaplayıcı federe cevap mesajı (Ack.) hazırlar. Hazırlanan cevap verisi HLA mesajı (interaction class) olarak zaman

ekseniyle ilişkilendirilmeden (asynchronous delivery) gönderilir. Bu adımın ayrıntılı akışı Şekil 3'te gösterilmiştir. Cevap olarak hazırlanan mesaj verisinin içeriği şöyledir:

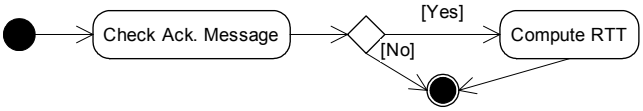
- Tekil test numarası
- Cevaplayan federe ismi
- Gelen test verisinin miktarı
- Gelen test verisinin zaman yönetimi açısından niteliği (RO veya TS)
- Eğer gelen test başlığında belirtilmişse test verisinin kopyası. Bu durumda gelen veri miktarı büyüklüğünde cevap verisi gönderilmiş olur.



Şekil 3. Adım2 ayrıntılı akış diyagramı

Adım3:

Bu adım gönderici federenin cevap mesajı (ack.) alması üzerine başlar. Alınan cevap mesajının başlığına bakılarak bu federeden gönderilmiş bir test mesajına cevap olarak gönderildiği kontrol edilir. Eğer öyle ise test mesajı gönderme ve cevap alma süresi hesaplanır. Böylece bir test ölçümü elde edilerek adımlar tamamlanır. Bu adımını akışı Şekil 4'te verilmiştir.



Şekil 4. Adım3 ayrıntılı akış diyagramı

Test sisteminde kullanılan ölçüm metodunun bazı özellikleri performans ölçüm sonuçlarında etkili olmuştur. Bu özelliklerin federasyon başarımına olan etkilerinde;

a. Başlık Verisi: Ölçümün yapılabilmesi için gönderilen veriye başlık bilgisi eklenmektedir. Gönderilen veri miktarı değerlendirmesinde başlık miktarı da göz önünde bulundurulmalı.

b. Gidiş-Dönüş Ölçümü: Test düzeneğinde verinin tek yönlü iletim süresi değil çift yönlü (gidiş-dönüş) iletim süresi hesaplanmıştır. Test sonuçları değerlendirilirken iletim süresinin bu niteliği de hesaba katılmalıdır.

Mevcut ölçümlerde sistemin federasyon başarımına olan etkisi için ayrıntılı bir ölçüm ve değerlendirme yapılmamıştır. Ancak bundan sonraki aşamalarda bu etkilerin de analiz edilmesi hedeflenmektedir.

4. ÖRNEK TESTLER

Şimdiye kadar yapılan test çalışmalarında performansı üzerinde etken olabilecek parametreler değiştirilerek ölçümler alınmıştır. Yapılan her bir test ölçümünde

değiştirilen parametreye göre başlıklar listelenmiştir. Zaman ölçümleri milisaniye olarak gösterilmiştir.

4.1. Veri Miktarı

Yapılan ardıl ölçümlerde gönderilen veri miktarı değiştirilerek gidiş-dönüş miktarı hesaplanmıştır. Her bir ölçümde gönderilen veri miktarı 50 byte artırılmıştır. Yapılan ölçümlerin grafiksel gösterimi Şekil 5'de verilmektedir. Çizelge 1'de ise ölçüm serisinden örnekler yer almaktadır.

Çizelge 1. Veri Miktarı ve Gidiş-Dönüş Zamanı

Veri Miktarı (byte)	64	864	1714	2614	3764	4864	8214
Gidiş-Dönüş Zamanı (ms)	7.94	8.12	8.94	9.47	9.91	10.72	15.52

4.2. RTI Zaman Yönetimi Servisi

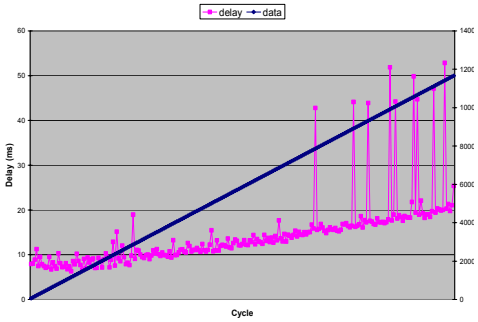
Farklı denemelerde sadece RTI Zaman Yönetimi (Time Management) devre dışı bırakılarak bu servisin iletim zamanına olan etkisi gözlemlenmiştir. Zaman yönetimi kullanılan ve kullanılmayan her iki durum için 285 ölçüm alınıştır. Zaman yönetimi kullanılmayan durumda ölçüm ortalaması 4,77 ms, zaman yönetimi olan durumda ise ölçüm ortalaması 5,05 ms olmuştur. Aradaki fark çok belirgin olmamakla birlikte zaman yönetimi kullanılması durumunda iletim zamanında daha büyük dalgalanmalar görülmektedir. Çizelge 2'de ise ölçüm serisine ilişkin veriler yer almaktadır. Şekil 8'de ölçümlerin grafiği verilmiştir.

Çizelge 2. Zaman Yönetim Servisi Değerleri

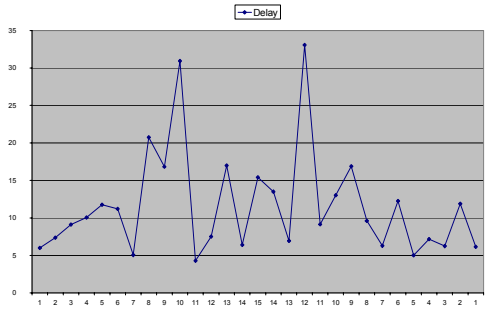
	RO	TS
Ortalama	4.77 ms	5.05 ms
En Küçük	3.49 ms	4.28 ms
En Büyük	17.70 ms	18.13 ms
Varyans	1.20 ms	1.79 ms

4.3. Nesne (Object Class Instance) Miktarı

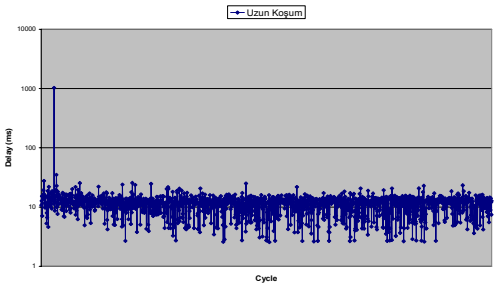
Bu denemede federasyonda bulunan ve veri akış trafiğine katılan nesnelerin sayısı değiştirilerek performans ölçümleri yapılmıştır. İlk 15 ölçümde bütün nesneler güncellenmiş ve nesne sayısı 1 artırılmıştır. Dolayısıyla 15. ölçüm yapıldığı esnada ortamda 15 nesne bulunmaktadır. 15. ölçümden itibaren ise bütün nesneler güncellenmekle birlikte 1 nesne silinmiştir. Dolayısıyla son ölçümde yine 1 nesne kalmıştır. Yapılan nesne güncellemelerinde 1 cevaplayıcı federe vardır, güncelleme veri miktarı 64 byte ve zaman etiketlidir. Şekil 6'da bütün ölçümlerde (toplam 29) bulunan nesnenin güncelleme ve cevaplanma zaman aralığı anlatılmıştır. Yapılan testte nesne sayısı veri akış performansını etkilememiştir.



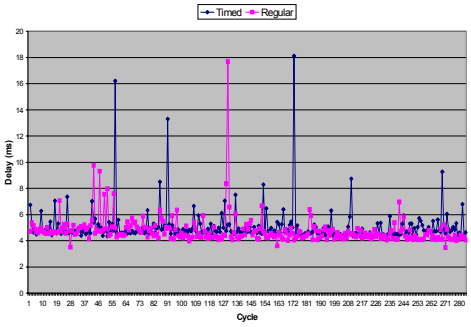
Şekil 5. Artan veri miktarına karşın iletim zamanı değişimi



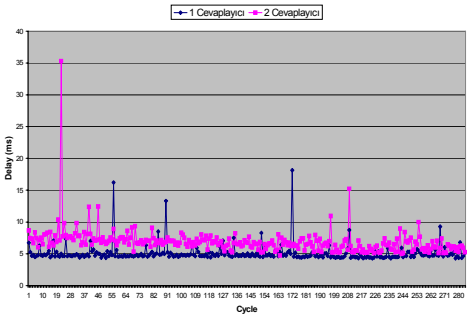
Şekil 6. Nesne sayısının performansa etkisi



Şekil 7. Uzun süreli koşum



Şekil 8. Zaman Yönetimi kullanımının etkisi



Şekil 9. Federe sayısının performansa olan etkisi

4.4. Abone Federe Sayısı

Bu denemede cevaplayıcı federe sayısı değiştirilerek performansa etkisi gözlenmiştir. İlk denemede 1 gönderici federe ve 1 cevaplayıcı federe kullanılarak 284 ölçüm yapılmıştır. Diğer denemede ise 1 gönderici ve 2 cevaplayıcı federe kullanılmıştır. 1 cevaplayıcı durumunda ortalama gidiş dönüş miktarı 5,06 ms, 2 cevaplayıcı durumunda ise 6,89 ms olmuştur. İlgili grafik Şekil 9'da gösterilmiştir. Çizelge 3'de ise ölçüm serisine ilişkin veriler yer almaktadır.

Çizelge 3. Abone Federe Sayısı Değerleri

	2 Cevaplayıcı	1 Cevaplayıcı
Ortalama	6.89 ms	5.06 ms
En Küçük	4.73 ms	4.28 ms
En Büyük	35.31 ms	18.13 ms
Varyans	4.37 ms	1.79 ms

4.5. RTI Sunucusu Devamlılığı

Bu denemede görülmek istenen uzun süreli bir HLA federasyon koşumunda RTI sunucusunun devamlılığı ve istikrarıdır. Yapılan denemede RTI sunucusu olarak pRTI 1516 kullanılmıştır. 1500 test ölçümü birbiri ardına yapılmıştır, her test ölçümünde 64 byte veri gönderilmiştir ve 9 cevaplayıcı federeden cevap alınmıştır. Veri gönderilmesinde zaman yönetimi servisleri kullanılmıştır. Şekil 7'de verilen grafik zaman ekseninde bu ölçümlerin değer değişimlerini göstermektedir. Ölçümlerin gösterdiği gibi RTI sunucusunun uzun süreli koşması performans üzerinde belirgin bir etki yapmamıştır. Ancak bazı ara değerlerde gözlemlenen farklılıklar nedeni ile ölçüm serisinin varyansı yüksek çıkmıştır. Çizelge 4'de ise ölçüm serisine ilişkin veriler yer almaktadır.

Çizelge 4. RTI Sunucusu Devamlılığı Değerleri

Ortalama	12.001 ms
En Küçük	2.56 ms
En Büyük	1023.42 ms
Varyans	669.89 ms

5. SONUÇ

Geliştirilen test ortamı ile testleri tekrarlayarak ve uzun süreli testler yaparak kurulan federasyon mimarisinin RTI performansı analiz edilebilmektedir. Bu bildiride verilen testler örnek olarak yapılmış ve çok sayıda tekrara dayanmamaktadır. Elde edilen sonuçlar incelendiğinde bir çok testte elde edilen zaman başarım ortalamalarının 20 ms'nin altında kaldığı gözlemlenmiştir. Ancak grafiklerde de görüldüğü gibi bazı anlarda ortalama değerden çok farklı performans olabilmektedir. Bu tür sapmaların sistem performansı ve nedenselliğin korunabilmesi tasarım aşamasında göz önüne alınması gerekmektedir. Bu analizler mimari tasarım aşamasında geliştiriciler için bir öngörü sağlamaktadır. Ancak test düzeneğinin gecikme ölçümlerinin hassasiyeti

federasyondaki makineler arasındaki zaman senkronizasyon hassasiyeti ile sınırlıdır. Düzeneğin zaman doğruluğunu arttırtmak için zaman ölçümünde bilgisayar saatlerinin eşgüdümü ve ortak saat kullanımının (network time) sağlanması planlanmaktadır. Ayrıca elde edilen sonuçların analizi için işlevselliğin arttırılması hedeflenmektedir. Bu sistemin MaK RTI ve diğer 1516 uyumlu RTI ürünleri testler yapılarak karşılaştırmaların elde edilmesi de amaçlanmaktadır.

KAYNAKÇA

[1] G. Riley, R. Fujimoto, M. Ammar, K. Permula, and D. Xu. (2001) "Distributed network simulations using the dynamic simulation backplane", In Proceedings of International Conference of Distributed Computing Systems.

[2] Imbrogno, M., Robbins, W., Pieris, G., (2004), "Selecting a HLA Run-Time Infrastructure: Overview of Critical Issues Affecting the Decision Process for War-in-a-Box", Technical Memorandum, DRDC, pp.3-5.

[3] Guha, R, Lee, J., Kachirski, O., (2005), "Evaluating Performance Of Distributed Computing Technologies – HLA And Tspaces On A Cluster Computer", Proceedings 19th European Conference on Modelling and Simulation.

[4] Fujimoto, R, Hoare, P., (1998), "HLA RTI Performance in High Speed LAN Environments", Proceedings of the Fall Simulation Interoperability Workshop.

[5] Ferenci, L.S., Fujimoto, R, (1999), "RTI Performance on Shared Memory and Message Passing Architectures", Proceedings of the Fall Simulation Interoperability Workshop.

[6] Lauderdale, G. D., Crues, E. Z., Snyder, D. V., Hasan, D. A., (2003), "Further Studies on the Feasibility of a Distributed ISS and HTV Simulation", Proceedings of the Fall Simulation Interoperability Workshop.

[7] IEEE (2001) "IEEE 1516, High Level Architecture (HLA)", IEEE, www.ieee.org.

[8] Möller, B., Karlsson, M, (2005), "Making RTI Tuning Easy With Performance Profiles", Proceedings of the European Simulation Interoperability Workshop,.

[9] Pitch Technologies: "RTIperf 1516 a HLA 1516 Performance Test Tool", www.pitch.se/rtiperf.

[10] Richardson, R., Wuerfel, R, (1998), "RTI Perfromance Testing", Proceedings of the Fall Simulation Interoperability Workshop.

[11] Tufarolo, J., et.al., (1999), "Automated Distributed System Testing: Designing an RTI Verification System", Proceedings of the Winter Simulation Conference.

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

DAĞITIK BENZETİM SİSTEMLERİ İÇİN PERFORMANS ANALİZİ YÖNTEMİNİN GELİŞTİRİLMESİ

Ramazan CENGİZ [(a)], Yrd. Doç. Dr. F. Erdoğan SEVİLGEN[(b)]

[(a)] TÜBİTAK MAM BTE, Gebze/Kocaeli, ramazan.cengiz@bte.mam.gov.tr

[(b)] GYTE, Gebze/Kocaeli, sevilgen@bilmuh.gyte.edu.tr

ÖZ

Sistem performansının izlenmesi ve analizlerinin yapılabilmesi, dağıtık benzetim sistemleri için çok önemli ve kritik bir problemdir. Bu tür sistemlerde problemlerin oluşma nedenleri ve performans kayıplarına yol açabilecek unsurların farklılığı; sistem analizlerinin karmaşıklaşmasına ve hata takibinin çok zor yapılmasına sebep olmaktadır. Performans alanında yapılacak incelemeler sonucu ortaya koyulacak kıstaslar ve oluşturulacak test ortamları sayesinde, üzerinde çalışılan sistemin performans sınırları ve kısıtları belirlenebilecek, yeni geliştirilen sistemlerin performans kontrolleri yapılarak daha etkin ürünler elde edilmesi sağlanacak ve kullanılacak modelleme benzetim alanına göre uygun performanslı ara katmanların seçilmesi için karşılaştırmalar yapılabilecektir. Bu kapsamda yaptığımız çalışmada, dağıtık benzetim sistemleri geliştirilmesi için üretilen iki HLA(High Level Architecture) yazılımı olan DMSO RTI 1.3v6, pRTI1516 v2.1 ara katmanlarının yapıları incelenerek iki sistemde performans takibinde temel alınabilecek ortak noktalar tanımlanmıştır. Bu sistemlerin ortak özellikleri kullanılarak yeni bir benzetim ara katmanı geliştirmiş ve kendi geliştirdiğimiz bu benzetim ara katmanı ile diğer iki HLA tabanlı yazılımın performans analizlerini yapmak için bir performans analiz sistemi tasarlanmıştır. Her üç sistemin aynı test ortamlarında, aynı test koşullarında denenmelerini sağlayarak bu sistemlerin karşılaştırılabilmesi için metrikler ortaya koyulmuştur. Ortaya çıkan analiz sistemi her türlü benzetim sistemine kolayca uyum sağlayabilecek şekilde geliştirilmiştir.

Anahtar Kelimeler: dağıtık benzetim, dağıtık sistemler, performans analizi, sistem performansı, sistem yükü, sistem gecikmesi.

A PERFORMANCE ANALYSİS METHODOLOGY FOR DISTRIBUTTED SIMULATION SYSTEMS

ABSTRACT

Analyzing and monitoring the system performance of distributed systems is a very important and critical problem. These systems have very different parts which can cause problems and there are many different kinds of elements that can leak performance of system. The criteria that can be generated with performance analysis results and with

the testbeds that are situated for simulations tests can make a good effect on distributed simulations projects. Such systems can be used for controlling and analyzing the performance while generating new high performance systems, defining the performance lacks and fails about the system working on and also comparing simulation middleware by performance for a specific chosen simulation model and simulation system. In this situation, we make our research on two different standard HLA implementations DMSO RTI 1.3v5, pitchRTI1516 and another one that we designed and implemented by examining these systems, to find out a performance methodology that can be used for all. By doing such a system we could test all these simulation subsystems in same environment and conditions and generated some information that can be used for comparing such systems. As a result of the research, it is generated a generic simulation subsystem performance analyzer utility for distributed systems.

Keywords: distributed simulation, distributed systems, performance analysis, system performance, system throughput

1. GİRİŞ

Dağıtık benzetim sistemlerinin performansı büyük oranda geliştirme için kullanılacak altyapıya bağlıdır. Bundan dolayı sistem tasarımcıları ve program geliştiriciler, üzerinde çalıştıkları sisteme en uygun altyapıyı bulabilmek için kullanılabilecek sistemleri karşılaştırabilecekleri bir test ortamına ihtiyaç duyarlar. Yine sistem yazılımı devam ettiği sırada geliştiriciler sistemin performans eksiklerini görebilmek ve takip edebilmek için bir ortama ihtiyaç duyarlar ya da geliştirilmesi biten bir ürünün hatalarını takip edebilmek için sistem bakımcısı benzetim sistemini ve üzerinde çalıştıkları alt yapıyı sürekli kontrol eden bir sisteme ihtiyaç duyar. Burada verilen örnekler daha da çoğaltılabilir. Bütün bu isteklerde bulunan ortak nokta benzetim alt yapılarının performans kıstaslarının ortaya koyulabilmesi ve bunların takip edilebilmesi probleminin olmasıdır, bundan dolayı bu konuda yapılacak incelemeler yukarıda bahsettiğimizle alakalı birçok sorunun çözülmesine katkıda bulunacaktır. Dağıtık sistemlerin farklı makineler üzerinde, farklı işletim sistemleri üzerinde, farklı yoğunlukta sistemlerde çalışabilmeleri bu sistemlerin performans analizlerini karmaşıklaştırmaktadır[6]. Bu açıdan da yapılacak incelemeler büyük önem arz etmektedir.

Dağıtık benzetim sistem altyapıları üzerinde çalışmalar, DMSO(ABD Savunma Modelleme ve Benzetim Ofisi)'nin askeri amaçlı yapılan modelleme çalışmaları, birbiriyle birlikte çalışabilen benzetimler tasarlayabilmek amacıyla desteklediği HLA çalışmalarıyla, hız kazanmıştır[1]. HLA 1998 yılında bir standart haline gelmiş ve bütün dünyada kabul görmüştür. Bir standart haline gelmesinden günümüze standardın hem kendi gelişimini sağlayabilmesi için hem mevcut altyapılarla kıyaslamalarının yapılabilmesi için de yine birçok performans analizi yapılmıştır. R. Fujimoto ve P. Hoare HLA standardında geliştirilen RTI çalışmalarında, RTI'ın performansını karakterize edebilmek için yüksek hızlı ağlarda denemeler yapmışlardır[2]. Rack ve arkadaşları CORBA ve DIS kullanarak geliştirdikleri SEEDS [3] ortamında sistem performansını takip edebilmek için testler yapmışlardır.

Bu makale kapsamında, üzerinde çalıştığımız üç benzetim sistemi altyapısının, bütün benzetim altyapılarında da kullanılabilecek bir performans inceleme yöntemi geliştirerek değerlendirilmesi için hazırlanan çalışmadan bahsedilecektir. Üzerinde çalışacağımız benzetim altyapıları; Pitch tarafından geliştirilen pRTI1516 ara katmanı, DMSO tarafından geliştirilen RTI 1.3v6 ara katmanı ve kendi geliştirdiğimiz genel amaçlı bir dağıtık benzetim ara katmanı olarak sıralanmaktadır. Bu ara katmanlar hakkında geniş bilgi 2. Bölümde verilecektir.

Ara katmanların test edilebilmesi için geliştirilen yapılar hakkında ayrıntılı bilgi 3. bölümde ifade edilecektir. Ara katmanların ortak noktaları ve performans test hedefleri göz önünde bulundurularak bir benzetim tasarımı yapılmıştır. Bu benzetime ait tasarım ve deneyler hakkında bilgiler 4. Bölümde verilmiştir.

2. PITCH RTI, DMSO RTI ve RAM DSA

Bu bölümde deneylerde kullanılacak ara katmanlar hakkında açıklayıcı bilgiler verilecek ve son olarak bu sistemlerde ortak karakter olarak ortaya çıkan özelliklerden bahsedilecektir.

2.1. PRTI 1516 v2.1

PRTI 1516 yazılımı, (Nisan 2003) IEEE 1516 2. sürümlü standardında belirlenen bütün servisleri sağlayan bir yazılımdır. IEEE 1516, HLA 1,3 (1998) standardının uluslararası boyutlara yaygınlaşmasıyla oluşturulmuş bir standarttır[8]. İlk sürümü 2000 yılında kabul edilmiş ve ondan sonra birçok firmanın yoğun ürün geliştirmesi döneminin sonucu olarak, 2003 yılında 2. sürümü kabul edilmiştir. Günümüze kadar geçen süre içinde yine birçok açık kaynak kodlu ve özel çıkartılan ürünler bu standardı sağlamaya yönelik araştırmalar yapmıştır. Günümüzde HLA standardının bir bakıma geliştirilme nedeni olan DMSO(ABD Savunma Modelleme Benzetim Ofisi)'nin gerçek standartlarda ürün olarak kendini ispatlayan yazılımlar listesinde[4] sadece 36 farklı sürümlü ürün bulunmaktadır. Bu listelenen ürünlerden 11 tanesi PRTI yazılımının değişik sürümleri olup, Pitch Teknoloji firması tarafından geliştirilmiştir.

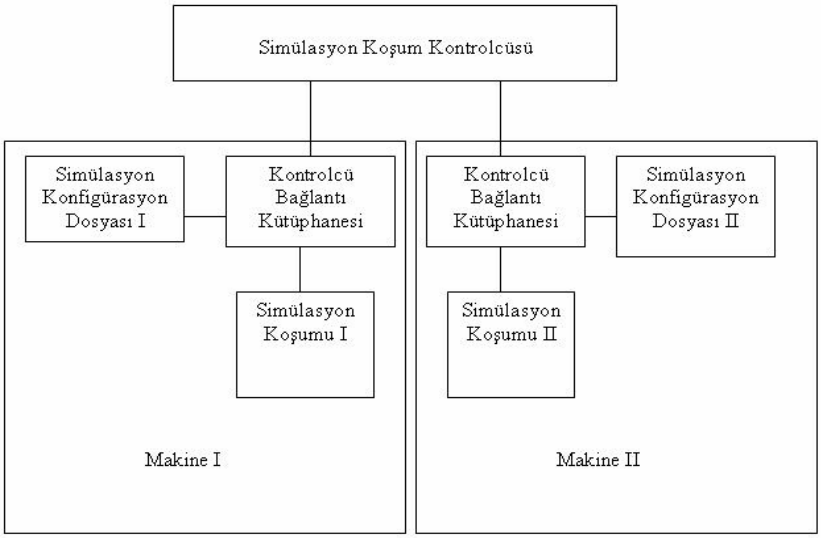
2.2. DMSO RTI 1.3v6 NG

DMSO tarafından geliştirilmesi desteklenen ve HLA standardının oluşmasına temel teşkil eden araştırmalar neticesinde ortaya çıkan ilk yazılımlardandır. HLA modelleme ve benzetim tasarımları için nesne tabanlı bir yapı sunmaktadır[9]. Her matematiksel model HLA ile bir "obje" olarak tanımlanabilmektedir. Bu da gerçek dünya içinde sanal objelerden oluşan bir benzetim oluşturulabilmesi için ortam oluşturmaktadır.

2.3. RAM Dağıtık Benzetim Altyapısı

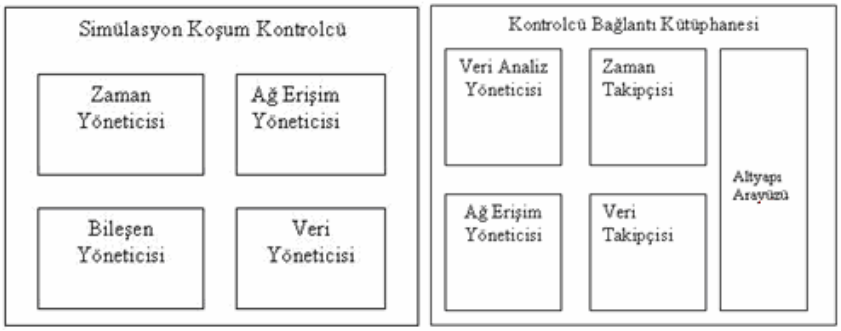
pRTI1516 ve DMSO RTI 1.3 NG üzerinde çalışmalar yaparken ortaya çıkan genel benzetim karakteristikleri göz önünde bulundurularak yazılmış bir benzetim ara katmanıdır. Geliştirilmesindeki amaç HLA standardına uygun olarak geliştirilmiş iki altyapı ile birlikte, dağıtık benzetim ara katmanı kavramına uygun başka bir yazılımın da, test edilebilirliğinin gözlemlenmesini sağlamak ve bu sistemi test ortamında kullanmaktır. Geliştirmelerde Microsoft Visual Studio 2005, ACE-TAO[10] kütüphaneleri kullanılmıştır. Geliştirilen sistem sağladığı genel amaçlı sistem tanımlama yapısıyla her türlü benzetimin geliştirilmesine olanak sağlayacak şekilde tasarlanmıştır.

Geliştirilen sistem benzetim zaman yönetimini, veri iletim yönetimini, sisteme yeni bileşenlerin katılıp ayrılmasını sağlayan bileşen kontrol yönetimini başarıyla sağlayabilmektedir. Geliştirilen sistemin genel şeması Şekil 1'de görüldüğü gibidir.



Şekil 1. RAM Benzetim altyapısı genel bağlantı şeması

Şekilde 1'de görüldüğü gibi benzetim koordinasyon işlevi, benzetim koşum koordinatörü ve benzetimle koordinatör arasındaki bağlantı işlerini yapmak üzere hazırlanmış bağlantı kütüphanesiyle yerine getirilir.



Şekil 2. RAM Benzetim Altyapısı Bağlantı Koordinatörü ve Bağlantı Kütüphanesi Şeması

Benzetim koşum kontrolcü, birlikte çalışmak isteyen benzetim parçalarının kendilerini kayıt ettirebilecekleri ve gurubun paylaşımına sunacakları nesneleri işlettirebilecekleri bir mekanizmaya sahiptir. Zaman yönetimi de yine bu koordinatör aracılığıyla yapılır. Şekil 2'de görüldüğü gibi koordinatörün çalışması için hayati önem oluşturan 4 modül geliştirilmiştir. Bu modüller dağıtık simülasyon koşumlarındaki zaman yetkilendirmesini yürüten zaman yöneticisi, ağ erişim yöneticisi, federasyona yeni

federelerin katılması ve ayrılmasını yürüten bileşen yöneticisi ve bileşenler arasındaki haberleşme mekanizmasını oluşturan veri yöneticisidir.

Benzetim koordinasyonun diğer tamamlayıcı unsuru benzetimlerin koordinasyon çevrimine katılabilme ve veri paylaşabilme işlerini benzetimler için yürüten bağlantı kütüphanesidir. Bağlantı kütüphanesi, altyapının uygulanacağı benzetim sistemine ait birlikte koşum için gerekli yapılandırma bilgilerini okuyarak benzetim adına koordinasyon işlerini yürütür. Bağlantı kütüphanesi kapsamında geliştirilen modüller şu şekilde sıralanabilir:

Altyapı arayüzü: Altyapının ağ bağlantıları hariç yürüttüğü bütün işler bu arayüz aracılığıyla yapılmaktadır. Benzetim sistemi bu arayüzle altyapıdan kendi işletimini soyutlayarak tasarımını basitleştirmektedir.

Konfigürasyon Dosyası ve Veri Analiz Yöneticisi: Benzetim sisteminin sağlıklı bir şekilde çalışabilmesi, dağıtık benzetime katılabilmesi için gerekli yapılandırma bilgileri xml tabanlı tutulan bir yapılandırma dosyası aracılığıyla altyapı tarafından veri analiz yöneticisi ile okunur ve işletilir. Bu yapılandırma bilgileri çalışacak benzetimin bir benzetim grubuna katılması için gerekli koşacak benzetime ait federe adı, katılacak benzetim gurubu (federasyon) kimlik bilgileri, dağıtık benzetimde haber alınması istenilen nesne ve güncelleme bilgileri yer alır.

Zaman ve Veri Takipçileri: Benzetim sistemine ait güncelleme bilgilerinin alınması, benzetime iletilmesi ve benzetimin yaptığı güncelleme isteklerinin koordinatöre iletilmesi işlevlerini yürütürler.

2.4. Sistemlerin Ortak Özellikler

Yapısal olarak incelenen ve testlerde kullanılan her üç dağıtık simülasyon ara katmanı, yapısal ve sistem geliştirme bakımında faklılıkları bulunmasıyla birlikte, işlevsellikleri itibariyle ortak özellikler taşımaktadırlar. Bu özellikler şu şekilde ifade edilebilir:

Her dağıtık benzetim ara katmanı ağ ortamında çalışabilecek şekilde tasarlanmıştır.

Her ara katman benzetim zaman yönetimini sağlamalıdır.

Ara katmanlar benzetimler arasında paylaşılan verilerin de benzetim zamanı olarak eşzamanlı olarak diğer benzetimlere iletilmesini sağlamalıdır. Verilerin benzetimler arasında kaybolmadan istenilen sırada ve zamanını geciktirmeden iletilmesi birlikte çalışabilirlik açısından çok önemlidir.

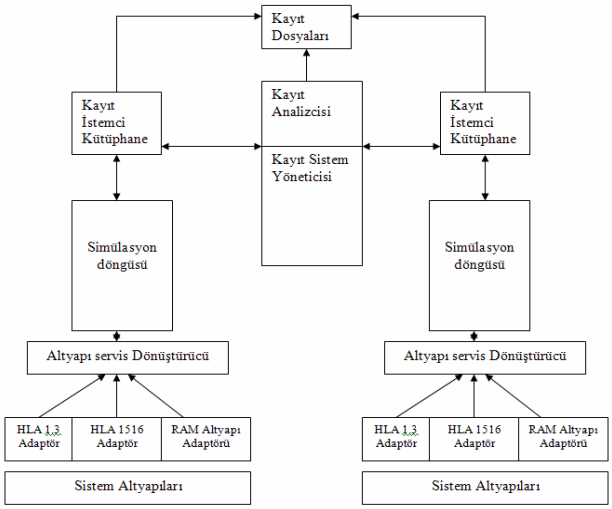
Altyapılar, benzetimlerin birlikte çalışabilmesi için bir arayüz sunmalı ve çalışma anında yeni bir benzetimin mevcut benzetim ağına dahil olup işlemi bitinceye kadar altyapı bağlantıları bu arayüz aracılığıyla sağlayabilmelidir.

Yukarıda ifadesi geçen ortak noktalar Şekil 3'te görüldüğü gibi benzetimlerle ağ arasında bir arayüz oluşturur. Bu sayede benzetim sadece kendi döngüsüyle ilgili işleri yürütür.

Şekil 3. Benzetim altyapılarında görülen ortak noktalar

3. PERFORMANS ÖLÇÜMLERİ İÇİN GELİŞTİRİLEN YAPILAR

Performans analiz ölçümlerinin yapılabilmesi için çok parçalı bir yapı geliştirilmiştir. Yapı temelde her türlü benzetim altyapısıyla çalışılmaya imkân sağlayan bir dönüştürücü, her türlü performans kayıtlarının farklı makinelerde ve ortamlarda eş zamanlı olarak tutulabilmesi için geliştirilen eş zamanlı kayıt sistemi ile kayıt analiz ve gözlemleme biriminden oluşmaktadır. Sistem genel görünüm şeması Şekil 4' de verildiği gibidir.



Şekil 4. Performans Analiz Sistemi Genel Şema

3.1. Genel Amaçlı Dağıtık Altyapı Dönüştürücü

Performans analizlerinin alt yapıdan bağımsız çalışabilmesi için benzetimle kullanılacak altyapı arasında veri akışını sağlayan sistemdir. Bu sistem sayesinde benzetimin geliştirilmesi sırasında ara katmanı kullanmak için ekstra iş tanımlanması ve benzetimin her altyapı için yeniden düzenlenmesi iş yükleri azaltılmış olmaktadır. Performans analizi yapılacak her bir benzetim ara katmanı geliştirdiğimiz sisteme uygun olarak geliştirilecek bir altyapı adaptörü aracılığıyla test edilebilmektedir. Alt yapı servis dönüştürücü yapısı analiz sistemi için ekstra bir veri ve zaman kaybı getirmemektedir.

Bu da yapılan analizlerde kullanılan kayıt verilerinin sadece benzetim ara katmanına ait olmasını sağlamaktadır. Elde edilen ürünle; yeni gelişen her yeni ara katman teknolojisi, belirlenen kıstaslar içinde performans analizine tabii tutulabilecektir. Geliştirilen dönüştürücünün işlevselliğinin test edilebilmesi için HLA 1.3 standardına örnek DMSO RTI 1.3v6 versiyonlu RTI ara katmanı, HLA 1516 standardına örnek olarak pRTI1516 ara katmanı ve RAM Dağıtık Benzetim Altyapısı için adaptör yazılımları hazırlanmıştır. Bu adaptörleler her bir ara katman üzerinde federasyon ve federelere ait tanımlamaların kolaylıkla yapılması sağlamaktadır. Bütün federasyon tanımlamalarının her adaptör için tek tip bir konfigürasyon dosyası aracılığıyla yapılması ara katmanlar arası geçişleri kolaylaştırmıştır.

3.2. Çoklu Bilgisayarda Eşzamanlı Kütük Kayıt Sistemi

Dağıtık sistemlerin ağ ortamında farklı makinelerde çalışmalarının getirdiği bir zorluk her makinenin sistem saatinin farklı olmasıdır. Bu saatlerin fiziksel olarak milisaniyeler mertebesinde eş zamanlı çalışmalarını sağlamak çok zor bir problemdir. Performans analizleri sırasında kontrol edilecek kayıtların ortak zaman değerlerini paylaşmaları için bir zaman yönetim sistemi kurulması ihtiyacı duyulmuştur. Ağ ortamındaki makineler için zaman senkronizasyonu konusunda değişik çalışma gurupları tarafından araştırmalar yapılmaktadır. Özellikle SNTP[11] (ağ zaman protokolü)'nün daha etkin hale getirilmesi için çalışmalar vardır. SNTP sistem düzeyinde zaman senkronizasyonunu 15 mili saniye –50 mili saniye arasında sağlayabilmektedir. Geliştirdiğimiz sistemin ağ ortamında zaman kayıklığı ~10 milisaniyedir.

Kayıt sistemi farklı makinelerde elde edilen kayıt bilgilerinin makinelerin yerel disklerinde dosyalar halinde tutulmasını böylece benzetim koşumu sırasında ağ yoğunluğunun artmasını engelleyecek şekilde geliştirilmiştir. Kayıt işlemi bittikten sonra birkaç federenin bir araya gelmesiyle oluşan bir federasyona ait performans analizinin yapılabilmesi için her federe için hazırlanmış kütük kayıt dosyaları, kayıt analiz aracına yüklenerek federasyona ait çok yönlü performans analizleri gerçekleştirilebilmektedir.

3.2.1. Kütük Kayıt Sistemi Yönetici Birim

Kayıt sistemi yönetici birimi performans analizine katılacak her yeni benzetim koşumu için kendisine bağlantı isteği gönderen hayıt sistemi istemci birimlerinin bağlantılarını kaydeden ve farklı makinelerden bağlanan istemcilerin zaman senkronizasyonunu için tetikleme mesajlarını gönderen birimdir. Zaman sinyali ağ veri akışını engellemeyecek şekilde 4 byte uzunluğunda zaman işaretiyle istemcilere gönderilir.

3.2.2. Kütük Kayıt Sistemi İstemci Birim

Kayıt istemci birimi zaman yönetimini yönetici biriminden aldığı güncellemeler sayesinde kendi saat değerini güncellemek şeklinde yapar. Kayıt istemcisi benzetim sırasında performans için gerekli kayıt bilgilerini yerel diskte analiz işlemlerinde kullanılmak üzere kaydeder.

3.2.3. Kütük Kayıtları Analiz Birimi

Performans analizlerinin yapıldığı sistem birimidir. Farklı makinelerdeki benzetimler sırasında tutulan kayıt bilgileri bu birime yüklenerek federasyona ait bütün zaman ve

nesne akış bilgileri birleştirilebilmektedir. Birleştirme işlemlerinden sonra sistem gecikmeleri, zaman benzetim gecikmeleri, sistem veri taşıma kapasitesi gibi analizler yapılabilir, benzetim saatine göre akış grafikleri elde edilebilir. Federasyona ait genel analiz işlemleri bütün koşum bittikten sonra yapılabileceği gibi koşumlar devam ederken de yerel disklerdeki kayıt dosyalarının mevcut hallerini kullanmak suretiyle yapılabilmektedir. Ayrıca her federeye ait olan kayıt bilgileri koşum anında analiz aracı kullanılarak görselleştirilebilmektedir[7].

4. DENEYLER VE SONUÇLARIN DEĞERLENDİRİLMESİ

Bu bölümde performans sisteminin geliştirildiği deneylerde kullanılacak benzetim ve koşum denemelerinin yapılacağı ağ ortamı hakkında açıklayıcı bilgiler verilecektir. Testlerde göz önünde bulundurulacak performans metrikleri hakkında bilgiler verilecektir.

Testlerde aranacak sonuç dağıtık benzetim sistemlerin ara katmanlarına ait performans değerlerini oluşturabilmektir. Dağıtık sistem altyapıları yapısal olarak farklılık gösterebilirler, burada bizim hedeflediğimiz nokta aşağıdaki ifade edilecek noktalarda cevaplar bulabilmek ve bu şekilde dağıtık benzetim sistemlerinde performans değerlendirmesini sağlayabilecek kıstaslar ortaya koyabilmektir.

- Benzetim sisteminin zaman yönetimi performansı, sisteme her yeni benzetim eklendiğinde, sistemde paylaşılan toplam veri boyutu arttığında, sistemde paylaşılan toplam değişken sayısı arttığında nasıl etkilenir.
- Benzetim sisteminin veri yönetimi performansı, sisteme her yeni benzetim eklendiğinde, sistemde paylaşılan toplam veri boyutu arttığında, sistemde paylaşılan toplam değişken sayısı arttığında nasıl etkilenir.

Elbette ki dağıtık sistemler için performans kısıtı oluşturabilecek farklı noktalar da bulunabilir, biz çalışmamızda bütün benzetimlerde kullanılabilecek bir analiz metodu olarak bu iki nokta üzerinde yoğunlaştık ve test edilecek altyapıları bu iki noktada ifade edilen değerlerin bir ifadesi olarak karakterize etmeye çalıştık.

4.1. Deneylerin Yürütülebilmesi İçin Geliştirilen Benzetim Sistemi

Benzetim altyapılarının test edilebilmesi için, veri akışının yoğun olduğu ve zaman senkronizasyonun çok önemli olduğu zaman tabanlı modellenen bir hareketli molekül benzetimi tasarlandı. Benzetimin zaman tabanlı çalışması zaman yönetiminin yoğun olarak devrede olmasını gerektirmektedir. Veri akşının yoğunluğu veri yönetiminin dengeli kontrol edilebilmesini gerektirmektedir. Geliştirilen benzetim değişik zamanlarda, değişik büyüklüklerde moleküllerin benzetime dahil edilmesini sağlamakta, böylece ara katmanlara düşen görev yoğunluğu kontrol edilebilmektedir.

4.2. Deneylerin Yürütüldüğü Ortam

Örnek deney çalışmaları gigabit ethernet ağlıyla birebirine bağlı bilgisayarların bulunduğu bir laboratuar ortamında yapılmıştır. Bu sayede ağ yapısının sistem bütününde yapacağı farklılıkları azaltmak mümkün olmuştur. Bu sayede ağda meydana gelen gecikmeler mümkün olduğunca azaltılmış ve elde edilen değerler daha çok sistem

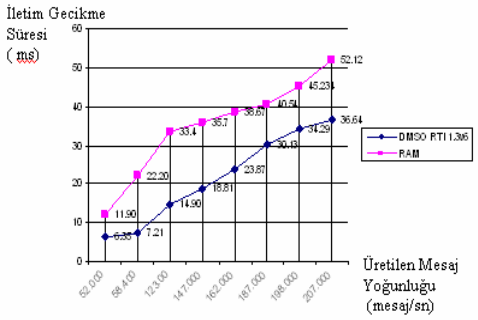
benzetim yazılımlarından ve ara katmandan kaynaklanmış olacaktır. Testler sırasında ağda meydana gelen gecikme de ölçülmüş ve sistem değerlerine etkisi yansıtılmıştır.

Sistem değerlendirmelerinde esas alınan veriler, sistem koşumları sırasında kayıt altına alınan değerlerin ortalamaları alınarak elde edilen değerlerdir. Analizler 1 saniyelik aralıklarla örneklenerek değerlendirme grafikleri oluşturulacaktır. Deneyler için yukarıda bilgisi verilen analiz sistemi kullanılmıştır.

4.3. Performans Metrikleri

Deneylerde performans incelemeleri için benzetim sisteminde yapılan ortalama veri transferi ve sistem birimlerinde meydana gelen gecikmeler birincil metrikler olarak kullanılmıştır[5]. Ortalama veri iletimi, saniyede sistemde akan ortalama veri sekizli cinsinden belirtilmiş, sistem birimlerinde meydana gelen veri iletim ve zaman istekleri gecikmeleri milisaniyeler mertebesinde ölçümlenmiştir. Gecikmeler performans sisteminde tutulan ortak saat bilgisi üzerinden zaman damgası vurulan mesajların yaşam döngülerinin çıkarılması ve ara katmana giriş-çıkış sürelerinin farklarının alınmasıyla hesaplanmaktadır.

Yapılan örnek bir deneyde 2 Makinede koşan 2 Federenin mesaj üretme yoğunluğu artırıldığında DMSO ve RAM ara katmanlarının performansının gözlenmesi amaçlandı. Sistemde iletilen ortalama mesaj boyutu 152 sekizlidir. Değişik yoğunluklardan alınan örnekler neticesinde iletimde meydana gelen gecikmeler Şekil 5'te görüldüğü gibi elde edilmiştir.



Şekil 5. Ara Katmanlar Gecikme Grafiği

5. SONUÇ

Geliştirilen performans analizi ortamıyla ve tanımlanan analiz yöntemiyle her türlü benzetim ara katmanının kontrolünü yapmak mümkün olmaktadır. Performans kontrolünün yapılması için zaman ve nesne gecikmeleri ile nesne iletim yükü hesaplanmış ve sistem performansının bu hesaplanan değer aracılığıyla, diğer ara katmanlarla karşılaştırılması sağlanmıştır. Önceden belirlenmiş hedeflerle elde edilen deney sonuçları karşılaştırılarak sistem tutarlılıkları ortaya koyulmuştur.

KAYNAKÇA

[1] Buss A., Jackson L., Distributed Simulation Modeling: A Comparison Of HLA, CORBA, and RMI, Proceedings of Winter Simulation Conference, pp.819-825, 1998.

[2] Fujimoto R., Hoare P., HLA RTI Performance in High Speed LAN Environments, Spring Simulation Interoperability Workshop, 2000.

[3] Rackl G., Stefani F., Héran F., Pasquarelli A., Ludwig T.,Airport Simulation using CORBA and DIS, pp. 465–472., 2000.

[4] http://www.dmso.mil/public/transition/hla/statusboard

[5] Gokhale S., Gokhale A., Gray J., A Model-Driven Performance Analysis Framework for Distributed, Performance-Sensitive Software, Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, (IPDPS 2005).

[6] Kounev S., Performance Modeling and Evaluation Distributed Component-Based Systems Using Queueing Petri Nets, IEEE Transactions On Software Engineering, 2006.

[7] Harmon T., Klefstad R., Automatic Performance Visualization of Distributed Real-Time Systems, Proceedings of the Ninth IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing, 2006.

[8] US Department of Defense,High Level Architecture Integace Speczjication, Version 1.3 of IEEE P1516.1, M&S HLA - Federate I/F Spec, DRAFT 1 of 20 April, 1998.

[9] Xiaoxia Shi., Qiuhai Z., The Introduction on High Level Architecture (HLA) and Run-Time Infrastructure (RTI), SICE Annual Conference in Fukui,2003.

[10] OMG (Object Management Group), (1998), The Common Object Request Broker: Architecture and Specification—Revision 2.2, Technical Report.

[11] D. Mills., Simple Network Time Protocol (SNTP).RFC 1769. March 1995

DAĞITIK BENZETİM UYGULAMALARINDA YENİDEN KULLANILABİLİRLİK, VERİ DAĞITIMI VE ZAMAN YÖNETİMİ[11]

Ersin ÜNSAL [(a)]

[(a)] Savunma Bilimleri Enstitüsü, Kara Harp Okulu, Ankara, ersin.unsal@gmail.com

ÖZ

Benzetim uygulamaları, silahlı kuvvetlerin daha etkin bir şekilde eğitilmelerini ve göreve hazırlanmalarını sağlamaktadır. Daha kapsamlı ve daha karmaşık benzetim projelerinin, zaman-etkin ve maliyet-etkin yöntemlerle gerçekleştirilmesi silahlı kuvvetler açısından önemli bir ihtiyaçtır. Bu ihtiyaca cevap verebilmek amacıyla, günümüzde dağıtık benzetim yaklaşımına başvurulmaktadır. Dağıtık benzetim uygulamalarında önem verilmesi gereken konular arasında yeniden kullanılabilirlik, zaman yönetimi ve veri yönetimi konuları başı çekmektedir. Bu çalışmada, DEB (DIS), YSM (HLA) ve TENA standartları tanıtılacak, bu standartların yeniden kullanılabilirlik, zaman yönetimi ve veri yönetimi yaklaşımları karşılaştırmalı olarak tartışılacaktır.

Anahtar Kelimeler: DEB, YSM, TENA, Dağıtık Benzetim, Yeniden Kullanılabilirlik, Veri Dağıtımı, Zaman Yönetimi

ABSTRACT

Simulation software helps the military forces to train and rehearse for missions. It is a national security concern to develop more complex and comprehensive simulations in both time effective and cost effective ways. Distributed simulation approach is widely used to answer these needs. Reusability, time management and data distribution are of the most important areas in distributed simulations. This paper will introduce DIS, HLA and TENA standards with a discussion on reusability, time management and data distribution approaches of these standarts comparatively.

Keywords: DIS, HLA, TENA, Distributed Simulation, Reusability, Data Distribution, Time Management

1. GİRİŞ

Benzetim sistemleri silahlı kuvvetler için oldukça büyük öneme haizdir. Askeri sistemler geliştikçe, daha kapsamlı ve daha karmaşık benzetim sistemlerinin maliyet-etkin ve zaman-etkin yöntemlerle geliştirilmesi ihtiyacı ortaya çıkmaktadır. Ayrıca daha

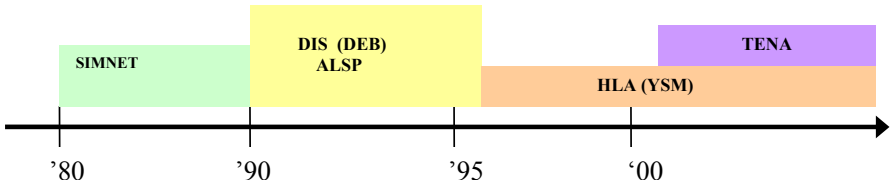
[11] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

gerçekçi test ve eğitim ortamlarının oluşturulabilmesi için benzetim bileşenlerinin ve gerçek varlıkların etkileşim içinde olacağı, coğrafi sınırlardan bağımsız olarak koşabilecek benzetim projelerinin geliştirilmesine ihtiyaç vardır.

Günümüzde bu ihtiyaçlara cevap vermek amacıyla dağıtık benzetim mimarileri kullanılmaktadır. Bu mimariler, benzetim projelerin zaman-etkin ve maliyet-etkin bir şekilde geliştirilmelerini sağlarken, farklı coğrafi konumlardaki benzetim bileşenleri ve gerçek varlıklar arasındaki etkileşimini de sağlamaktadırlar. Bu çalışmada, üç önemli dağıtık benzetim mimarisi tanıtılacak; bu mimariler yeniden kullanılabilirlik, veri dağıtımı ve zaman yönetimi yaklaşımları bakımından karşılaştırılmalı olarak incelenecektir. Bu çalışmanın amacı, dağıtık benzetim mimarilerinin faydalarını ve farklılıklarını ortaya çıkarmaktır.

2. DAĞITIK BENZETİM MİMARİLERİNİN TARİHSEL SÜRECİ

DEB – **D**ağıtık **E**tkileşimli **B**enzetim (DIS – Distributed Interactive Simulation) ve YSM – **Y**üksek **S**eviyeli **M**imari (HLA – High Level Architecture) savunma sanayiinde yaygın olarak kullanılan, standartları oluşmuş dağıtık benzetim mimarileridir [1, 2]. TENA – **T**est ve **E**ğitim Ola**NA**klı Mimari (Test and Training Enabled Architecture) ise henüz standartlaşma sürecini tamamlamamış, geliştirme ve yaygınlaştırma süreci devam eden bir mimaridir [3]. Bu mimarilerin her üçü de Amerika Savunma Bakanlığı'nın değişik birimlerinin dağıtık benzetim alanında yaptığı araştırma ve geliştirme çalışmaları sonucu ortaya çıkmıştır. Şekil 1.'de dağıtık benzetim mimarilerinin tarihsel gelişimi görülmektedir.



Şekil 1. Dağıtık Benzetim Standartlarının Tarihsel Gelişimi

SIMNET projesi, tank eğitimlerini bir ağ üzerinden yapmak amacıyla 1980'lerin başlarında başlatılmış bir programdır. DEB, DARPA (US Defence Advanced Research Project) tarafından desteklenen SIMNET projesi sonrasında ortaya çıkmış bir protokoldür. ALSP, DARPA desteğiyle MITRE şirketi tarafından geliştirilmiş diğer bir dağıtık benzetim protokoldür. YSM ise DEB ve ALSP ile yapılmış çalışmalardan elde edilen kazanımlar ve yeni ihtiyaçlar çerçevesinde, DMSO (Defense Modeling and Simulation Office) öncülüğünde tasarlanmış ve geliştirilmiş bir mimaridir. TENA ise Amerika Savunma Bakanlığı'nın DOT&T (Director, Operational Test and Evaluation) birimi tarafından yürütülen Foundation Initative 2010 projesinin temel ürünü olarak ortaya çıkmıştır.

3. DAĞITIK ETKİLEŞİMLİ BENZETİM - DEB

Dağtık Etkileşimli Benzetim'in temel hedefi, farklı konumlardaki değişik benzetimlerin etkileşimli olarak çalışabilmesi için gerekli olan altyapının tanımlanmasıdır. Karmaşık ve gerçeğe yakın sanal sistemler in oluşturulması hedeflenmiştir. Bir DEB benzetiminde sanal kullanıcılar ve gerçek nesneler (tank, uçak) bir benzetime dahil olabilmektedir [4].

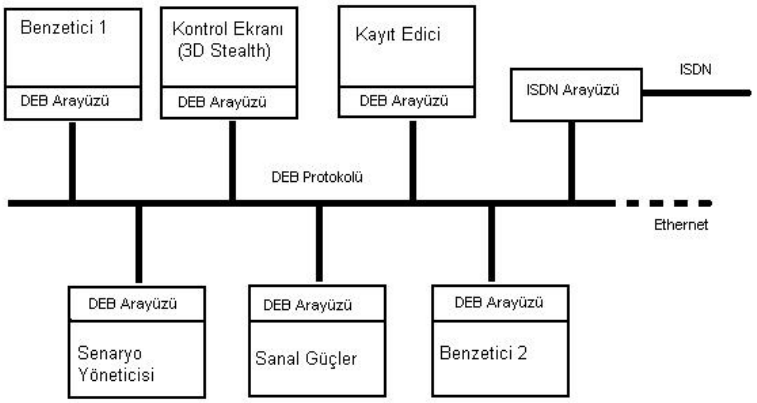
DEB yaklaşımın temel ilkeleri şunlardır [5, 6, 7, 8]:

Nesne/Olay Mimarisi: Yapay ortamdaki tüm dinamik varlıklar, durumlarını ve eylemlerini, PVÜ - Protokol Veri Ünitesi (PDU – Protocol Data Unit) isimli standart bilgi paketleri aracılığıyla tüm diğer varlıklara bildirirler.

Benzetim Düğümlerinin Özerkliği: Bireysel bir benzetici tarafından üretilen tüm olaylar, diğer tüm benzeticilerin ulaşabileceği şekilde yayımlanırlar. Üretilen bu olayların diğer üreticiler üzerinde yaratacağı etki, diğer üreticilerin kendi sorumluluklarındadır.

Yalnızca Durum Değişikliği Bilgisinin İletilmesi: Benzeticiler, sadece benzetimini gerçekleştirdikleri varlığa ait durum değişikliği bilgisini iletirler. Devamlı hareketlilikler azaltılmış bir güncelleme sıklığı ile iletilirler. Her benzetici, yeni bir durum güncelleme bilgisi ulaşana kadar, son gelen durum bilgisini kullanarak, benzetimin durumu ile ilgili yerel bir tahmin yapar. Buna gözü kapalı tahmin (Dead-Reckoning) ismi verilir.

Temelde, DEB bir protokoldür ve benzetimler için herhangi bir yazılım mimarisini adreslemez. DEB protokolüne uygun olarak iletim yapabilen tüm benzeticiler, herhangi bir DEB tabanlı benzetime dahil olabilirler. Şekil 2'de örnek bir DEB benzetimi yapısı görülmektedir.



Şekil 2. Örnek DEB Benzetimi Yapısı

4. YÜKSEK SEVİYELİ MİMARİ

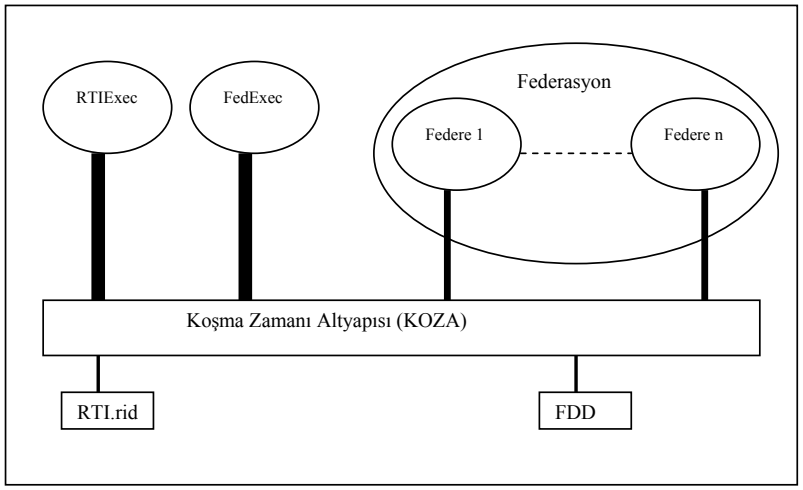
YSM, yeniden kullanılabilirlik ve birlikte işlerlik prensipleri doğrultusunda geliştirilmiş bir mimaridir. YSM'deki temel kavram ve ilkeleri şu şekildedir [8, 9, 10, 11] :

- Benzetimler, *federe* ismi verilen ve gerçek dünyada varolan nesneleri modelleyen benzetim parçalarından oluşurlar. Federeler bir araya gelerek *federasyon*ları oluştururlar. Federe ve federasyonların uyması gereken kurallar, YSM standardının "YSM Kuralları" bölümünde belirtilmiştir.

- Federelerin birbirleriyle etkileşimini sağlayan, üzerinde koştukları yazılıma Koşma Zamanı Altyapısı – KOZA (Run-time Infastructure – RTI) ismi verilir. YSM standardının "Arayüz Tanımı" kısmı, KOZA'nın sahip olması gereken arayüzleri ve sağlaması gereken servisleri tanımlamaktadır. Standarda uygun olarak farklı kişi ve kuruluşlar tarafından KOZA yazılımları geliştirilebilir.

- Birlikte çalışabilirliğin ve yeniden kullanılabilirliğin gereği olarak tüm nesne ve etkileşimlerin detaylı olarak belirlenmiş ortak bir biçime uygun şekilde tanımlanmaları gerekmektedir. YSM standardı bu amaçla Federasyon Nesne Modeli, Benzetim Nesne Modeli ve Yönetim Nesne Modeli'ni tanımlar.

KOZA, YSM uyumlu benzetimler için birçok farklı servis sunar. Bu servisler federasyon yönetimi, bildirim yönetimi, nesne yönetimi, sahiplik yönetimi, veri dağıtımı yönetimi ve zaman yönetimi ana başlıkları altında yer alırlar [9]. Şekil 3.'de örnek bir YSM uyumlu benzetim yapısı görülmektedir.



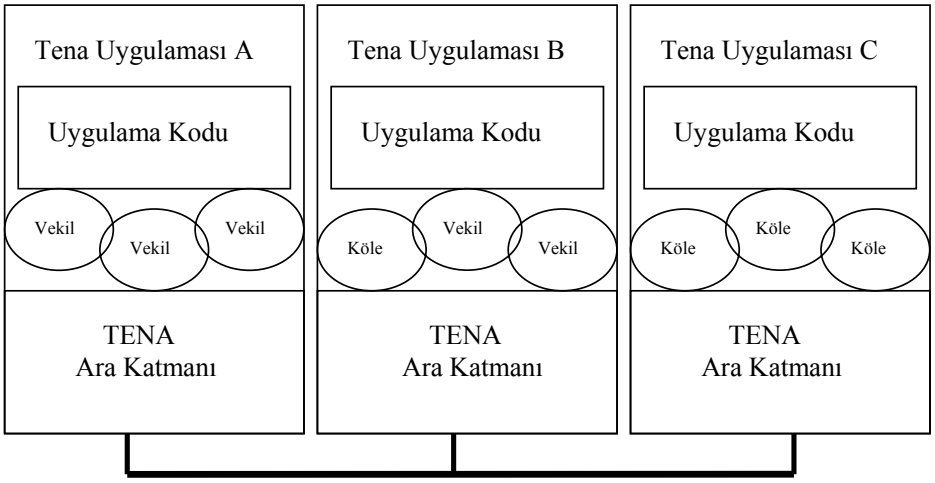
Şekil 3. YSM Uyumlu Benzetimi Yapısı

5. TEST VE EĞİTİM OLANAKLI MİMARİ - TENA

TENA, silahlı kuvvetlerin test ve eğitim saha operasyonlarına hizmet etmek amacıyla geliştirilmiş bir mimaridir. TENA'nın tasarlanmasında da YSM'de olduğu gibi yeniden kullanılabilirlik ve birlikte işlerlik gereksinimlerine öncelik verilmiştir. TENA için önem arz eden diğer bir gereksinim ise birleştirilebilirlik (composability) gereksinimidir [3, 12]. Birleştirilebilirlik, yeniden kullanılabilirlik ve birlikte işlerlik özelliklerine haiz bileşenlerinin biraraya getirilerek kolay ve hızlı bir şekilde yeni bir sistem oluşturulmasını ifade eder.

TENA, YSM ve DII COE (Defense Information Infrastructute Common Operating Environment) mimarileri de dahil olmak üzere birçok mimari üzerinde yapılan çalışmalar ve edinilen tecrübeler sonucu tasarlanmıştır. TENA'nın ana yapısı, DARPA'nın büyük ölçekli ve birlikte çalışabilir sistem mimarileri araştırmalarını temel alan "Extended C4ISR Architecture Framework" çerçevesi ile de uyumludur [3].

TENA, mantıksal saha (logical range) vizyonunu gerçek kılmayı amaçlar. Mantıksal saha, gerçek askeri varlıkların birbirleriyle ve benzetilmiş askeri güçlerle etkileşime girebildiği bir ortamdır. Gerçek askeri varlıkların, benzetilmiş güçlerin ve silahların gerçek dünyadaki koordinatları, bu etkileşim için bir sorun oluşturmaz. TENA, mantıksal ağlar için eşler arası ağ yapısı (peer-to-peer network) adresler. Uygulamalar sunumcu (server) olursa köle (servant) isimli TENA nesneleri kullanılır. İstemci (client) uygulamalar, diğer uygulamaların kölelerini temsil eden vekilleri (proxy) kullanırlar. Şekil 4'de örnek bir mantıksal sahası görülmektedir.



Şekil 4. Örnek TENA Mantıksal Sahası

6. YENİDEN KULLANILABİLİRLİK

Yeniden kullanılabilirlik, bir yazılım parçasının veya diğer bir iş çıktısının birden fazla programda veya yazılım sisteminde kullanılabilmesidir [13]. Yeniden kullanılabilirlik kavramı ve getirdiği avantajlar yıllardır tartışılmaktadır. Yeniden kullanılabilirlik, yazılım geliştirme çevrim sürelerini kısaltırken, üretkenliği ve kaliteyi arttırmaktadır. (McIlroy Kanunu, D.M. McIlroy, 1969). DEB ve YSM'nin yeniden kullanılabilirlik konusundaki başarıları, benzetici ve federelerin yeniden kullanılabilirliği karşılaştırılarak değerlendirilebilir.

Daha önce de belirtildiği üzere, DEB temelde bir protokoldür ve yazılım mimarisi anlamda bir standardı yoktur. Şekil 2.'ye bakıldığında DEB uyumlu benzeticilerin kolay bir şekilde benzetime dahil olabileceği düşüncesi oluşsa da, birlikte çalışabilirlik açısından sorunlarla karşılaşılmaktadır. Benzeticilerin DEB uyumlu olması, birlikte çalışabilir olduğu anlamına gelmemektedir [6].

Gerek sadece bir protokol olması gerekse birlikte çalışabilirlik sorunları nedeniyle DEB'in yeniden kullanılabilirlik anlamında diğer mimariler kadar başarılı olduğu söylenemez.

YSM bir mimari tanımdır ve bu tanım yeniden kullanılabilirlik ve birlikte çalışabilirlik üzerine kurulmuştur. Değişik amaçlarla geliştirilmiş federeler, farklı projelerde tekrar tekrar başarıyla kullanılabilir. Ayrıca federeler, KOZA'nın sağladığı arayüz tanımlarına ve servislere uygun olarak geliştirildikleri için federelerin geliştirilmesi sürecinde de yeniden kullanılabilirliğin avantajlarından yararlanılabilir.

TENA, yeniden kullanılabilik, birlikte işlerlik ve birleştirilebilirlik gereksinimleri temel alınarak tasarlanmış bir mimaridir. Ayrıca TENA, sahip olduğu havuz (repository) sayesinde benzetim uygulamalarının üretkenliğini ve kalitesini artırabilecektir. TENA havuzunda yardımcı araçlar, TENA ara katmanını da içeren yazılım kütüphaneleri, dökümantasyon, TENA nesne modeli ve bu modellerin standart gerçeklemeleri yer alır [3]. TENA'nın üretkenliği ve kaliteyi artırmak konusundaki bir diğer avantajı da TENA ara katmanının sağladığı otomatik kod üretebilme yeteneğidir.

7. VERİ DAĞITIMI

Dağıtık sistemlerde düğümlerin birbirleriyle etkin bir şekilde haberleşebilmesi oldukça önemli bir ihtiyaçtır. Veri dağıtımı yönetiminin amacı düğümlerin işleyeceği veri miktarının (ölçeklenebilirlik) ve ağ üzerinde taşınan verinin (performans) azaltılmasıdır [11]. Ayrıca verilerin güvenli bir şekilde dağıtımı da önemli bir konudur. Burada güvenirlikten kasıt, gönderilen verilerin karşı tarafa doğru ve eksiksiz olarak ulaştırılabilmesidir.

DEB, UDP/IP protokolü kullandığı için veri paketlerinin kaybolması ve/veya sıralarının karışması muhtemeledir. Dolayısıyla veri iletiminin güvenilirliği ile ilgili problemlerle karşılaşılmaktadır [8].

DEB'de veri gönderimi Protokol Veri Ünitesi adı verilen standart yapılar aracılığıyla yapılır. İhtiyaçlara göre hazırlanmış sınırlı sayıda PVÜ vardır ve yeni ihtiyaç olması halinde çeşitli onay mekanizmaları vasıtasıyla yeni PVÜ'ler tanımlanır. Veri gönderiminin standart PVÜ'lere göre yapılması ise iki farklı açıdan sıkıntı yaratmaktadır. İlki, amaca uygun bir PVÜ'nün bulunmaması durumunda, oldukça uzun bir süreç olan yeni bir PVÜ tanımlanması sürecine ihtiyaç duyulabilir. Diğer bir sorun ise, PVÜ'ler genel amaçlı oldukları için fazladan alanlar içerebilmektedirler. Bu da verinin iletildiği ağın verimsiz kullanılmasına sebep olmaktadır. Bu durum ölçeklenebilirlik amacına ters düşmektedir [6].

YSM'de federeler, KOZA'nın sağladığı Veri Dağıtımı Yönetimi Servisi'ni (VDYS) kullanabilirler. Bu servisin kullanımı ihtiyaca göre belirlenir ve bu servis zaman yönetiminden bağımsız olarak kullanılabilir. VDYS, nesne yönetimi ve bildirim yönetimi servisleriyle uyumlu bir şekilde çalışan, ölçeklenebilir, işlemci gücünü ve bant genişliği etkin bir şekilde kullanabilen bir servistir [9]. YSM, sadece ihtiyaç duyulan ve/veya değişen bilgilerin iletimini gerçekleştirir. Böylece gereksiz bilgilerin ağ üzerinde taşınmasına izin vermemektedir. Ancak, YSM veri paketleri üzerindeki kılıflar, DEB'dekilerine oranla daha büyüktür ve bunun performans üzerinde olumsuz etkileri olmaktadır [6]. YSM'nin etkin bir veri dağıtımı yapabilmesi, VDYS'i gerekli şekilde kullanabilecek bir tasarım yapılmasına da bağlıdır [14]. VDYS'de eşleştirme ve kayıt olan grupların yönetimi anlamında birtakım problemlerin olduğunu belirten ve bu konularda yeni yaklaşımlar öneren çalışmalara da literatürde rastlanmaktadır [15].

Dağıtık ortam için yazılım geliştirilmesi ve dağıtık ortamda verinin etkin bir şekilde paylaşılabilmesi, sadece benzetim alanına özgü bir ihtiyaç değildir. Birçok askeri ve sivil projelerde dağıtık mimariler kullanılmaktadır. Dağıtık mimarilerde en çok tercih edilen ara katmanlardan bir tanesi CORBA mimarisidir [16]. KOZA yazılımının CORBA mimarisi kullanılılarak geliştirilmesi konusunda çeşitli çalışmalar bulunmaktadır [17]. CORBA standardını da geliştirmiş olan OMG [18], Veri Dağıtımı Servisi – VDS (Data Distribution Services – DDS) standardı üzerinde çalışmaktadır. VDS'nin YSM ile karşılaştırıldığı ve YSM'nin VDS'nin özelliklerinden faydalanabilmesi konusunda da çalışmalar vardır [19].

TENA ara katmanı CORBA mimarisi esas alınarak geliştirilmiştir. TENA'da uygulamalar, sunulan farklı veri alışverişi yöntemlerinden birini seçebilirler. Bu yöntemler :

- Publish/Subscribe Yöntemi : YSM ve DEB'de var olan yöntemdir.
- RMI (Remote Method Invocation) Yöntemi : CORBA ve Java'da var olan yöntemdir.
- Mesaj Yöntemi :Uygulamalar arasında bireysel mesajlar gönderilebilmesini sağlayan bir yöntemdir.
- Veri Akım (Data Stream) Yöntemi : Ses, görüntü ve veri hattı iletimi için destek veren bir yöntemdir.

TENA ara katmanı, CORBA'da var olan dağıtık nesne modeli ve YSM'de var olan Yayınla/Kayıt Ol modellerini birleştirerek tasarlanan SDO (Stateful Distributed Object) soyut modelini esas alır. Bu sayede her iki yaklaşımında güçlü taraflarını TENA'da görmek mümkündür. Bu yaklaşım daha basit, daha kullanışlı ve daha etkin arayüzlerin tasarlanmasını sağlamaktadır. [20].

TENA ile yapılmış ve yaklaşık 180.000 varlığın işlendiği örnek bir gerçekleştirimde farketme süresi, güncelleme süresi, hafıza ve işlemci kullanımı ile ilgili herhangi bir problem görülmediği gözlenmiştir [21].

8. ZAMAN YÖNETİMİ

Dağıtık benzetim sistemlerindeki çözülmesi gereken problemlerden bir tanesi de zaman yönetimidir. Farklı işlemci hızlarına sahip bilgisayarlar üzerinde koşan dağıtık benzetim bileşenlerinin ortak bir zaman kavramı oluşturması ve iletişim amaçlı gönderilen mesajların ağ üzerindeki olası yoğunluk ve problemlerden bağımsız olarak istenen sırada gönderilebilmesi gerekmektedir.

DEB benzetimleri sadece gerçek zamanda koşarlar. DEB benzetimlerinde, dağıtık benzetim bileşenleri kendi yerel zamanlarını, diğer benzetimlerden bağımsız olarak yönetirler. DEB'de alternatif zaman yönetim yaklaşımlarını da destekleyen bir zaman yönetim sistemi yoktur [7, 8].

KOZA'nın Zaman Yönetimi Servisi ise YSM uyumlu benzetimler için oldukça kapsamlı bir zaman yönetim hizmeti sağlamaktadır. YSM'de benzetimler gerçek zamanda koşabililirler ancak gerçek zamanda koşma zorunluluğu bulunmamaktadır.

YSM uyumlu benzetimler için geçerli olan zaman yönetiminin özellikleri şunlardır [9].

- Her federenin bir *mantıksal zamanı* vardır. Federasyon içinse böyle bir zaman tanımı yoktur. Bu zamanın bir birimi yoktur ancak bir başlangıç değeri vardır ve düzenli aralıklarla ilerler. Federelerin mantıksal zamanlarının ilerlemesinin koordinasyonu Zaman Yönetimi Servisi tarafından yapılmaktadır.

- Federeler zaman yönetimine farklı biçimde katılabilirler. Federelerin iki konuda tercih yapmaları gerekir. İlk konu federelerin *zamana-bağımlı* olup olmayacağıdır. Diğer konu ise federelerin *zamanı-yöneten* olup olmayacağıdır. Hem zamanı-yöneten hem de zamana-bağlı federeler olabileceği gibi (tam senkronize federeler), hem zaman yönetimine katılmayan hem de zamandan bağımsız çalışan federeler olabilir (varsayılan durum).

- Zaman yönetiminin etkisi gönderilen ve ulaşan olaylarda görülür. Ulaşan olaylar ya Ulaşma Sıralı (Receiving Order - RO) ya da Zaman Damgası Sıralı (Time Stamp Order - TSO) olabilirler.

- KOZA, zaman damgası sıralı olayları düzgün sırayla dağıtacağını garanti eder.

TENA, gerçek zamanlı benzetimleri gerçeklemeyi amaçlayan ve çalışma zamanı performansına önem veren bir mimaridir. Dolayısıyla YSM'de olduğu gibi farklı zaman yönetimlerini destekleyen bir mekanizması bulunmamaktadır.

9. SONUÇ VE TARTIŞMA

1980'lerde SIMNET ile başlayan dağıtık benzetim mimarisi çalışmaları, günümüzde kullanılan YSM ve 2010 vizyonuyla hazırlanan TENA ile devam etmektedir. Bu süreçte özellikle yeniden kullanılabilirlik, birlikte işlerlik ve TENA ile birlikte birleştirilebilirlik konularına önem verilmiştir. Bu sayede benzetim projelerinin zaman-etkin ve maliyet-etkin geliştirilebilmesi hedeflenmiştir. Dağıtık benzetim projelerinin performansını ve ölçeklenebilirliğini artırmak amacıyla veri dağıtımı konusunda yeni yaklaşımlara başvurulmuştur. Zaman yönetimi konusunda ise genelde gerçek zamanlı yaklaşıma ağırlık verilmiş, ancak özellikle YSM farklı isteklere cevap verilebilmesi amacıyla alternatif zaman yönetim mekanizmalarını destekleyecek şekilde tasarlanmıştır. Bu bildiride tartışılan dağıtık benzetim mimarileri ve bu mimarilerin temel özellikleri Çizelge 1'de karşılaştırmalı olarak gösterilmiştir.

Çizelge 1. Dağıtık Benzetim Sistemlerinin Karşılaştırılması

	Yeniden Kullanılabilirlik	Veri Dağıtımı	Zaman Yönetimi
DEB	Belirli bir mimariyi adreslemeyen, birlikte çalışabilirlik sorunları olan bir mimari	Güvenilirliği düşük ve esnek olmayan bir veri dağıtımı sistemi	Gerçek zamanlı yaklaşımı benimseyen bir mimari
YSM	Yeniden kullanılabilirliği hedefleyen bir mimari	Ölçeklenebilirliği ve performansı hedefleyen esnek bir veri dağıtım sistemi	Zaman yönetimi servisi üzerinden farklı zaman yönetimi alternatifleri sunan bir mimari
TENA	Yeniden kullanılabilirliği ve birleştirilebilirliği hedefleyen bir mimari	Önceki yaklaşımların güçlü taraflarını birleştiren etkin bir veri dağıtım sistemi	Gerçek zamanlı yaklaşımı benimseyen bir mimari

Ülkemizde dağıtık benzetim yaklaşımı yaygın olarak kullanılmaya başlamıştır. Güncel benzetim projelerin bir çoğu YSM uyumlu olarak geliştirilmektedir. Ayrıca dağıtık benzetim alanında yardımcı yazılım geliştirme çalışmaları da devam etmektedir.

Dağıtık benzetim mimarileri konusunda bilgi birikimin ve deneyimin artması ile birlikte daha karmaşık, insanın ve gerçek nesnelerin (tank, top, gemi) dahil olduğu daha gerçekçi ve büyük çaplı benzetim projelerin başarıyla geliştirilmesi mümkün olacaktır. Bu tür projeler, silahlı kuvvetlerimizin daha etkin bir şekilde eğitilmesine hizmet edecek, eğitimlerin kapsamı ve gerçekçiliği giderek artacaktır.

KAYNAKÇA

[1] 1516-2000 IEEE Standard for Modeling and Simulation High Level Architecture

[2] 1278-1993 IEEE Standard for Distributed Interactive Simulation

[3] TENA Architecture Reference Document Version 2002, Review Edition

[4] Fujimoto, R.M., Weatherly, 1996, R.M., "HLA Time Management and DIS", PADS

[5] Zalcman, L., "What Distributed Interactive Simulation (DIS) Protocol Data Units (PDU) Should My Australian Defence Force Simulator Have?", Australian Goverment, Department of Defence, Defence Science and Technology Organization

[6] Ryan, P., Zalcman, L., 1999, "The DIS vs HLA Debate: What's in it for Australia", SIMTECT

[7] Jense, H., Kuijpers, N., 1996, "DIS – The Next Generation: A perspective for European Space Programmes", ESA

[8] Jense, G. J., Kuijpers, N.H.L., Dumay, A.C.M., 1997, "DIS AND HLA: Connecting People, Simulations and Simulators in the Military, Space, Civil Domains", IAF

[9] Kuhl, F., Weatherly, R., Dahmann, J., 1999, "Creating High Level Computer Simulation Systems", Prentice Hall PTR

[10] RTI 1.3-Next Generation Programmer's Guide Version 3.2, 2000, DMSO

[11] Ünsal, E., Sevilgen, F.E., 2006, "Yüksek Seviyeli Mimari İçin Modelleme ve Uygulama Geliştirme Aracı", SAVTEK

[12] Noseworthy, J. Russell, 2003, The TENA Middleware, Supporting Real-Time Application Development for the Dod Range Community

[13] IEEE Standart Computer Dictionary: A Compilation of IEEE Standart Computer Glossaries, 1999, New York, NY

[14] Kujipers, N., Lukkien J., Huijbrechts, B., 1999 , "Applying Data Distribution Management and Ownership Management Services of the HLA Specification", SIW

[15] Raczy, C., Jun, Y., Tan, G., Chuan, T. S., 2002, "Adaptive Data Distribution Management for HLA RTI", ESIW

[16] http://www.corba.org

[17] Au, T. A., "Performance Issues of HLA Run Time Infastructure based on CORBA", http://www.siaa.asn.au

[18] http://www.omg.org

[19] Joshi, R., Castellote, G. P., "A Comparision and Mapping of Data Distribution Service and High-Level Architecture", http://www.rti.com

[20] Noseworthy, J. Russell, 2003, The TENA Middleware, Supporting Real-Time Application Development for the Dod Range Community

[21] Hudgins, Gene, TENA Offers Range Interoperability and Resource Reuse Solutions for Tests and Trainings, NDIA Test&Evaluation Conference

B5 OTURUMU:

KARAR DESTEK AMAÇLI MODSİM-II

İSTANBUL BOĞAZI DENİZ GEMİ GEÇİŞ TRAFİĞİNİN BENZETİM MODELLEME VE SENARYO ANALİZİ İLE İNCELENMESİ
Birnur ÖZBAŞ, İlhan OR, Alper Ö. ALMAZ, Erhan GÖNÜLTAŞ, Tayfur ALTIOK, Özgecan S. ULUSCU

SİSTEM DİNAMİKLERİ YAKLAŞIMLI TERFİ SİSTEMİ ANALİZ MODELİ (TESAMOS)
Umut AKIN, Altan ÖZKİL

NÜKLEER, BİYOLOJİK ve KİMYASAL (NBC) TAARRUZ SİMÜLASYON MODELİ
Ebru ARSLAN, Güher BOZDOĞAN

BENZETİM PROJELERİNDE BİLİMSEL VE TEKNİK YORDAM GELİŞTİRME PRATİĞİNE BİR YAKLAŞIM
Yıldırım BAHADIRLAR

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

İSTANBUL BOĞAZI DENİZ GEMİ GEÇİŞ TRAFİĞİNİN BENZETİM MODELLEME VE SENARYO ANALİZİ İLE İNCELENMESİ

Birnur ÖZBAŞ [(a)], İlhan OR [(b)], Alper Ö. ALMAZ [(c)], Erhan GÖNÜLTAŞ [(d)], Tayfur ALTIOK [(e)] , Özgecan S. ULUSCU [(f)]

[(a)] Boğaziçi Üniversitesi, Endüstri Mühendisliği Bölümü 34342 Bebek, İstanbul, birnur@ozbas.com.tr

[(b)] Boğaziçi Üniversitesi, Endüstri Mühendisliği Bölümü, or@boun.edu.tr

[(c)] Boğaziçi Üniversitesi, Endüstri Mühendisliği Bölümü, alperalmaz@hotmail.com

[(d)] Boğaziçi Üniversitesi, Endüstri Mühendisliği Bölümü, erhan_gonultas@yahoo.com

[(e)] Rutgers University, Department of Industrial & Systems Engineering, Piscataway, NJ 08854, USA, altiok@rci.rutgers.edu

(f) Rutgers University, Department of Industrial & Systems Engineering, ozgecanu@eden.rutgers.edu

ÖZ

Bu çalışma İstanbul Boğazı'ndaki transit deniz trafiği için bir benzetim modeli geliştirmeyi ve geçiş talebinin tipi ve sıklığının yanı sıra, çeşitli doğal etkenlerin, kaynakların, karar politikalarının sistem üzerindeki etkilerini incelemek için senaryo analizi gerçekleştirmeyi amaçlamaktadır. Bu çerçevede modelde Boğaz trafik kuralları ve düzenlemeleri, gemi gelişleri/özellikleri, kılavuz kaptan/römorkör hizmetleri, meteorolojik, coğrafi ve mevsimsel koşullar göz önünde tutulmuştur. Takip edilen model çıktıları ise geçen gemi sayısı ve tipi, gemi geçiş süreleri ve bekleme zamanları, kuyruklardaki gemi sayıları, Boğaz boyunca gemi yoğunlukları, kılavuz kaptan ve römorkör kullanım oranlarıdır. Bu bakış ile Boğaz trafik benzetim modeli, ayrıntılı transit gemi geliş süreci alt-modeli, rassal görüş mesafesi ve akıntı alt-modelleri, takip mesafesi ve diğer yönetimsel kontroller ile donatılmıştır. Güncel Boğaz trafik kuralları ve düzenlemeleri, mevcut kılavuz kaptan/römorkör hizmetleri, trafik şeritleri ve sollama koşulları da modele yüklenmiştir. Çalışmanın sonuçları, muhtelif senaryolar çerçevesinde, sunulan etkenlerin izlenmekte olan çıktılar üstündeki ayrı ayrı ve bütünleşik etkilerini açığa çıkarmaya yöneliktir. Bu doğrultuda, gemi geliş sıklığı, gemi özellikleri ve görüş mesafesi gibi dışsal faktörler ile kılavuz kaptan/römorkör imkanları ve Boğazdaki takip mesafesi gibi kontrol araçlarının önemi irdelenmektedir. Bunun yanı sıra kurulacak modelin kapsamlı bir risk analizi çalışması yapmaya yönelik bir platform oluşturması beklenmektedir.

Anahtar Kelimeler: Arena®, Benzetim, Deniz Trafiği, İstanbul Boğazı, Risk Analizi

INVESTIGATION OF TRANSIT MARITIME TRAFFIC IN THE STRAIT OF ISTANBUL THROUGH SIMULATION MODELING AND SCENARIO ANALYSIS

ABSTRACT

This study aims to develop a functional simulation model for the transit traffic in the Strait of Istanbul and to perform scenario analysis to investigate the effects of type and frequency of transit demand, as well as various natural factors, resources and decision / policy rules, on the system. In this regard, the Strait traffic rules and regulations, transit vessel profiles, pilotage and tugboat services, meteorological and geographical conditions are considered in the model to provide a platform to analyze the effectiveness of various policies and decisions related to the transit traffic in the Strait. The statistics tracked and collected in the analysis are number and types of vessels passed, transit times and waiting times of vessels, number of vessels in the queues, vessel densities throughout the Strait, pilot and tugboat utilizations. In this regard, the Strait traffic simulation model is equipped with comprehensive arrival process sub-model, stochastic visibility and current sub-models, parameterization of pursuit distances and other administrative controls (regarding vessels in transit in the Strait). Current Turkish Straits' traffic rules and regulations, the available pilotage and tugboat services, traffic lanes and overtaking conditions are all imposed to the model as well. Results of the study display the individual or aggregate effects of the input factors on the tracked model outputs, in the context of various scenarios. Within this framework, the important effects of external factors (such as vessel profiles and arrival rates), and control tools (such as pilotage and tugboat availabilities and pursuit distance in the Strait) are highlighted. Moreover, it is expected that this model will provide a platform for a comprehensive risk management and analysis study of the Strait of Istanbul.

Keywords: Arena®, Maritime traffic, Risk Analysis, Simulation, Strait of Istanbul.

1. GİRİŞ

Dar ve kıvrımlı yapıları nedeniyle nehri andıran Türk Boğazları (İstanbul ve Çanakkale Boğazları), dünyadaki en stratejik suyolu sistemlerindendir. Gemilerin seyrini zorlaştıran coğrafi yapıları, meteorolojik şartlar, yoğun ve artan transit/yerel trafik, gemi/kargo özellikleri, kıtalararası köprüler ve enerji nakil hatları gibi fiziksel engeller Boğazlar'ın trafik şartlarını karmaşık ve riskli olmasına neden olmaktadır.

İstanbul Boğazı, coğrafî olarak, dünyanın en dar suyollarından birisidir. Uzunluğu 31 kilometre, ortalama derinliği 45 metredir [1]. Ortalama genişliği 1,5 km iken, bu genişlik en dar noktasında 700 metreye inmektedir. Ayrıca, yoğun sis, kuvvetli akıntı ve rüzgâr gibi sık yaşanan olumsuz hava koşulları, Boğaz'daki seyri daha karmaşık hâle getirmektedir.

Ayrıca, İstanbul Boğazı'ndaki trafiğin tehlikesini arttıran doğal olmayan etkenler de vardır. Bunlardan biri şehir hatları gemilerinin, balıkçı teknelerinin, özel teknelerin neden olduğu yoğun yerel trafiktir [2]. Bir diğeri ise transit gemilerin sıklığı ve kargo özellikleridir. 2005 yılında 54.500'den fazla (10.300'ü tehlikeli madde taşıyan olmak üzere) gemi İstanbul Boğazı'ndan geçiş yapmıştır.

Bu olumsuz ortamda, deniz kazası riskini kontrol altında tutup, azaltırken, seyir güvenliğini artırmak için, Türkiye Cumhuriyeti gelişmiş bir gemi izleme ve kontrol sistemi (GTH) kurmuş, ayrıca kapsamlı bir Deniz Trafik Kuralları ve Düzenlemeleri Yönetmeliğini (K&D) yürürlüğe almıştır. GTH, Boğaz'ın yanında, Marmara Denizi ve Karadeniz'e doğru 20 millik bir alanda yakından takip imkânı sağlarken, gerek GTH'nin operasyonu, gerekse de K&D'nin uygulama yetki ve sorumluluğu Gemi Trafik Hizmetleri Merkezi'ne (GTHM) verilmiştir. İstanbul Boğazı'nı geçecek olan tüm gemiler, Boğaz seyirlerini GTHM'nin K&D'ye dayanan talimatlarına göre yaparlar [2].

Bu çalışmanın amacı, İstanbul Boğazı'ndaki transit gemi trafiği için, K&D'ye dayanan bir benzetim modeli oluşturarak; geçmiş transit gemi ve çevresel veriler temelinde, politika, kaynak mevcudiyeti, transit gemi profili ve çevresel şart değişiklerinin etkilerini hızla analiz edip değerlendirecek, gerçekçi ve pratik bir ortam sağlamaktır. Başka bir deyişle, amaç İstanbul Boğazı'ndaki deniz trafiğini etkileyen tüm kuralları, prosedürleri, doğal ve insan yapısı şartları inceleyen bir platform elde etmektir. Ayrıca bu platform, GTHM'nin gerçek zamanlı karar verme mekanizmasına destek sağlayacak bir araç konumunda olabilmelidir. Uzun vadede, bu çalışma İstanbul Boğazı'nın kapsamlı bir risk yönetimi ve analizi çalışmasına da platform oluşturacaktır.

2. MODEL KURULUMU VE HEDEFLER

Kurulan deniz trafiği modeli aşağıdaki ana unsurlardan oluşturmaktadır:

- Boğazın kuzey ve güney girişlerinde rassallaştırılmış gemi gelişleri;
- Rassallaştırılmış gemi tipleri ve özellikleri;
- Rassallaştırılmış görüş ve akıntı koşulları;
- Mevcut Boğaz Trafik Kuralları ve Düzenlemeleri (K&D);
- Kılavuzluk ve römorkör hizmetleri;
- İki ana trafik şeridi, bir geçiş şeridi ve bütünleşik geçiş kuralları.

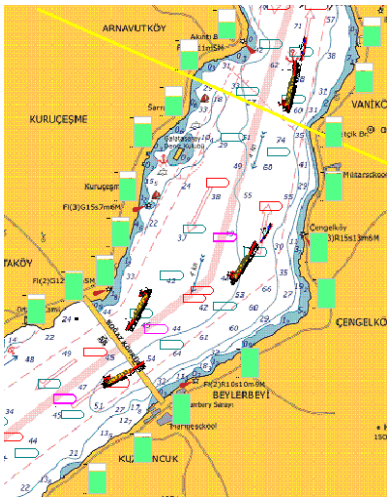
Çalışma amaçları doğrultusunda, benzetim modeli Boğaz'daki transit gemilerin davranışlarını etkileyen şartları gerçekçi olarak yansıtmakta ve böylece trafik akışını doğru bir biçimde taklit etmektedir. Çeşitli girdi faktörlerinin seviyeleri 'senaryolar' halinde birlikte düzenlenmiştir ve böylece değişik senaryoların sonuçlarını inceleme ve karşılaştırma olanağı doğmuştur. Senaryo analizi ile;

- Artan transit trafik talebinin, olumsuz meteorolojik koşulların, hizmet seviyesi (römorkör ve kılavuz hizmetleri gibi) ve gemi sınıf önceliklerindeki değişikliklerin etkileri daha iyi tahmin edilebilir;
- K&D'nin uygulamaya yönelik farklı yorumlarının etkileri incelenebilir;
- Değişik gemi öncelikleri sınanarak, bekleyen gemilerin tehlikeli yığılmaları, bekleme zamanları ve istenmeyen transit gemi yoğunlukları azaltılmaya çalışılabilir;
- Kılavuz kaptan ve römorkör sayı hedefleri ve bu seviyelerdeki beklenen kullanım oranları, transit trafik yoğunluğunun bir fonksiyonu olarak belirlenebilir.

Benzetim modeli bu konuları ele alacak şekilde özel olarak tasarlanmıştır.

3. MODEL YAPISI

Sistemin girdileri, transit gemilerin tipleri, uzunlukları, hızları, demirleme süreleri, kılavuz kaptan ve römorkör talepleri, duraklı veya duraksız geçiş durumları, giriş yönü, görüş ve akıntı durumu olarak sıralanabilir. Model mantığını daha anlaşılır kılmak ve senaryo analizini kolaylaştırmak için kontrol parametreleri üç sınıfta toplanmıştır (dışsal ve içsel parametreler ile K&D) [3]. Dışsal parametreler (GTHM tarafından kontrol edilemeyen parametreler) meteorolojik koşullar, transit gemi profili, geliş sıklığı, hız ve kılavuz/römorkör talepleri olarak özetlenebilir. Kılavuz kaptan ve römorkör sayısı, transit gemi öncelikleri ve gemilerin birbirini geçiş kuralları modelin içsel parametreleridir (GTHM tarafından kontrol edilebilen parametreler). Üçüncü sınıfı oluşturan Kurallar ve Düzenlemeler bu çalışma boyunca değişmez olarak alınmıştır.



Şekil 1. Benzetim modeli

4. MODEL MANTIĞI VE AKIŞI

İstanbul Boğazı transit trafiği benzetim modeli Kurallar ve Düzenlemelere dayanır. Her gemi için, Boğaz'a yaklaşmasından bölgeden ayrılmasına kadar, demirleme, bekleme (gece, görüş, kötü akıntı koşulları, kaptan/römorkör durumu ve kural kısıtlamaları nedenleriyle), birbirini geçiş fırsatları, akıntı etkileri ve takip mesafesi gibi, geminin başına gelebilecek olayların hepsi dikkate alınmış ve benzetim modeline eklenmiştir.

Dikkate alınan gemi tipleri Yolcu gemileri, LNG-LPG gemileri, Tehlikeli Madde (HazMat) gemileri, Tankerler ve Dökme/Kuru Yük gemileridir. Ayrıca, K&D tarafından (gemilerin hassasiyetlerini, potansiyel riskleri ve özel ihtiyaçlarını yansıtacak şekilde) tip ve uzunluğa göre tanımlanan 11 *muamele sınıfı* da modelde uygulanmıştır.

Gemiler Boğaz'da sekiz farklı bölgeden geçer [4]. Her bölge, birbirinden 2 gomina (0,2 deniz mili ≈ 370 m.) mesafedeki bir dizi 'istasyon'a ayrılmıştır. Bu suretle, Boğaz 16,8 deniz millik mesafeyi kapsayan 84 istasyona ayrılmıştır. Burada amaç, K&D uygulamasını doğru olarak modellemek ve Boğaz'daki iki transit gemi arasındaki takip mesafesiyle ilgili bir esneklik kazanmaktır. Bu mesafe GTHM'nin kontrol parametreleri arasındadır ve modelde bu şekilde yer almıştır. K&D'nin katı bir şekilde uygulamasında ve temel ayarlarda, takip mesafesi 8 gomina olarak alınmıştır.

Benzetim modelinde transit gemiler için üç trafik şeridi vardır (kuzey yönlü, güney yönlü ve geçiş şeridi) [3]. K&D Boğaz'ın en dar yerinde (Vaniköy – Kanlıca arası) gemilerin birbirine geçmesine izin verilmediği için, geçiş şeridi bu kısımda çıkarılmıştır.

Fiili uygulama paralelinde, hızlı bir geminin önündeki yavaş bir gemiyi geçmesi aşağıdaki şartlar sağlandığı takdirde mümkün olmaktadır:

i) Geçişin beklenen tamamlanma süresi içinde, geçiş şeridinde karşından gelen başka bir geminin bulunmaması;

ii) Geçiş şeridindeki aynı yönde ilerleyen en yakın geminin en az takip mesafesi kadar uzakta olması;

iii) Geminin Kandilli Bölgesi'ne gelmeden geçişini tamamlayacak kadar hızlı olması.

Transit gemilerin kılavuz kaptan ve römorkör ihtiyaçları/şartları da dikkate alınmıştır.

Modelde, günlük görüş mesafeleri dışardan girdi olarak alınabildiği gibi geçmiş verilere dayalı olasılık dağılımları ile model içinde de üretilebilir. Geçmiş yılların analizi görüş mesafesi realizasyonlarında güçlü bir mevsimsel özelliğe işaret etmiştir (düşük görüş mesafeleri kış mevsiminde çok daha sık görülmektedir). Bu nedenle yıl, potansiyel görüş mesafesi gerçekleşmelerine göre üç mevsime ayrılmıştır: yaz, kış ve geçiş mevsimleri. Yaz (Haziran-Eylül) ve geçiş mevsimlerindeki (Mayıs ve Ekim) görüş mesafeleri ampirik dağılımlar kullanılarak modellenmiştir. Kış mevsimi görüş mesafesi modellenmesinde ise, bir açık/kapalı prosese dayalı, faz tipi dağılım kullanılmıştır [5]. Bu yaklaşımda, iyi görüş koşulları açık durumuyla temsil edilirken, kötü görüş koşulları kapalı durumla gösterilmiştir. Açık/kapalı dönemlerinin uzunlukları genelleştirilmiş Erlang dağılımlarının karışımıyla yakınsanmıştır [6].

Boğaz'da kuzeyden güneyden doğru bir yüzey akıntısı vardır. Bu akıntı, Boğaz'ın en dar yerinde en yüksek değerine ulaşırken, girişlerde bu değerin %90'ı mertebesine inmektedir. En yüksek akıntı değeri yaklaşık 2,7 knot ortalama ile, 0 ila 6 arasında rassal olarak değişir [7]. Doğal olarak, bu akıntı transit gemilerin gerçek hızlarını etkiler (güney yönlü gemileri hızlandırırken kuzey yönlü gemileri yavaşlatır). Ayrıca, yüksek akıntı seviyelerinde seyir sorunları yaşayabilecek gemiler ya römorkör alarak geçiş yapar, ya da şartlar iyileşene kadar bekletilir[2]. Bu faktörler modelde, akıntıya bağlı hız değişimleri, römorkör atamaları ve/veya gemi giriş engellemeleri olarak yer almaktadır.

Boğaz boyunca günlük akıntı değerleri, en yüksek değerin rassal olarak belirlenmesi ve bu değerin her istasyona, önceden belirlenen oranlarda yansıtılmasıyla elde edilir. En yüksek değerin olasılık dağılımı, parametreleri geçmiş akıntı verilerine dayanan, özbağlanımlı tahmin modeli esaslıdır; söz konusu oranlar da, geçmiş akıntı verilerine dayanır [6].

5. DOĞRULAMA VE GEÇERLEME

Doğrulamaya yönelik olarak, modelleme adımlar hâlinde gerçekleştirilmiş ve her adımda hata ayıklama ve test işlemleri yapılmıştır. Bunun yanında, benzetim modeli çerçevesinde geliştirilen animasyon, gemilerin giriş, çıkış ve transit geçişi boyunca görsel olarak takip edilmesine ve doğrulama sürecinin desteklenmesine olanak sağlamıştır.

Geçerleme sürecinin ilk aşamasında uç koşul geçerlemesi yapılmıştır. Bu amaçla, seçilen parametrelere uç değerler atanmış ve modelin ürettiği davranış, gerçek sistemin aynı koşullar altında göstermesi beklenen davranışla karşılaştırılmıştır.

Bir benzetim modelinin geçerliliğini ölçecek en kesin test, modelin çıktı değerlerinin gerçek sistemden beklenen değerlere gösterdiği yakınlıktır. Bu nedenle, benzetim modeli (Ocak – Temmuz 2005) dönem için 10 yineleme ile çalıştırılmış ve sonuçların

ortalaması bu dönemin gerçek istatistikleriyle karşılaştırılmıştır. Ayrıca, sistemin stabil duruma gelmesi amacıyla, bu dönemdeki istatistiklerin kullanılmadığı bir aylık (Aralık) bir ısınma periyodu kullanılmıştır. Karşılaştırılan temel başarı ölçütleri geçen gemi sayısı, ortalama transit geçiş süresi ve gemilerin ortalama bekleme zamanlarıdır. Boğaz'daki gemi yoğunluğu, kuyruklarda bekleyen gemi sayısı, kılavuz kaptan ve römorkör kullanım oranları da toplanmış ancak uygun geçmiş istatistik bulunmadığı için karşılaştırmada kullanılamamıştır. Örnek olarak, 2005 yılının ilk yedi ayında geçen toplam gemi sayısı ile 10 yinelemeli benzetim koşumlarının sonucunda ortaya çıkan, geçiş yapan toplam gemi sayısı ortalaması Tablo 1'de gösterilmektedir. Gerçek verideki ve benzetim sonucundaki toplam, güney yönlü ve kuzey yönlü geçiş yapan gemi sayıları oldukça yakındır.

Tablo 1. 2005 verisinde ve benzetimde yedi ayda geçiş yapan gemi sayısı

	2005 Verisi	Benzetim Modeli			
Yön	*Toplam*	*Ortalama*	*Std. Sapma*	*Alt Güven Sınırı*	*Üst Güven Sınırı*
Toplam	31411	31468.2	22.34	31445.81	31490.59
KG	15712	15737.6	21.05	15716.51	15758.69
GK	15699	15730.6	6.67	15723.92	15737.28

6. MODEL ÇIKTILARI

Modelde, her benzetim koşumu Kuzey ve Güney girişlere, toplam gemi girişlerine, tüm gemi tiplerine göre ve her gemi tipi için ayrı ayrı hesaplanmış aylık çıktı dosyaları oluşturur. Çıktı dosyalarındaki istatistiksel değerler aşağıda sözü edilen tepki değişkenlerinin maksimum, minimum, ortalama değerleri, standart sapma ve yüzde 95lik güven aralıklarından oluşmaktadırlar;

- Her ayın sonunda (transit geçiş izni için) kuyruklarda bekleyen gemilerin sayısı,
- Geçişlerini tamamlamış olan gemilerin sayısı,
- Boğaz geçişini tamamlamış gemilerin bekleme süresi (geçiş yönlerine göre, tüm gemiler için ve her gemi tipi için ayrı ayrı),
- Boğaz boyunca her bölgedeki ve tüm Boğaz'daki gemi yoğunluğu (deniz mili başına düşen transit gemi sayısı, tüm gemiler için ve her gemi tipi için ayrı ayrı),
- Boğaz transit geçişini tamamlamış gemilerin geçiş süreleri (geçiş yönlerine göre, tüm gemiler için ve her gemi tipi için ayrı ayrı),
- Kılavuz kaptan ve römorkör kullanım oranları (toplam kullanım süresinin kullanılabilirlik süresine oranı);

7. SENARYO ANALİZLERİ VE SONUÇLARI

Deneysel benzetim koşumları 7 faktör ve onların çıktı istatistikleri üzerindeki etkilerine odaklanmışlardır. Senaryolar bahsi geçen faktörlerin, Tablo 2'de gösterilen, seviyeleri göz önüne alınarak oluşturulmuşlardır.

Tablo 2. Senaryo analizlerindeki faktörler ve seviyeleri

Faktör	Adı	Düşük	Yüksek
A	Geliş Sıklığı	Normal	Yüksek
B	Gemi Profili	Normal	HazMat Yüksek
C	Kılavuz Kaptan/Römorkör	20/9	15/6
D	Akıntı Profili	Normal	Yüksek
E	Kandilli Kuralı	Normal	Tutucu
F	Takip Mesafesi	4 gomina	8 gomina
G	Mevsim	Yaz	Kış

Gemi geliş sıklığı, her transit gemi tipinin geliş sıklık dağılımının ortalaması anlamına gelmektedir. Gemi profili ise gemilerin sıklık dağılımlarını tiplerine göre belirlemektedir. Kılavuz kaptan/römorkör faktörü mevcut kılavuz kaptan/römorkör sayılarıdır. Akıntı profili, günlük en yüksek akıntı seviyesini oluşturmak için kullanılan olasılık dağılımının ortalama değerini yansıtmaktadır. Kandilli kuralı K&D'nin önemli maddelerinden birisidir. Bu kurala göre, riski azaltmak adına, belli tipteki gemiler Kandilli bölgesinde (Boğazın en dar noktası) karşılaşmamalıdırlar. Fakat gemilerin Boğaz'daki seyirleri sırasındaki kesin hızlarını tahmin etmek güç olduğundan, bu kural belli bir güven faktörü ile uygulanmalıdır. Güven faktörünün "Normal" seviyesinde, gemilerin Boğaz'a girişleri (beklenen geçiş hızları temel alınarak) Kandilli bölgesinin iki yönünde iki istasyon içinde istenmeyen karşılaşmaların olması engellenecek şekilde, düzenlenmiştir. "Tutucu" seviyede ise, istenmeyen karşılaşmaların Kandilli'nin iki yönünde dört istasyon içinde olması engellenmişlerdir. Öte yandan, daha önce sözü edildiği gibi, "takip mesafesi", modelde 2–10 gomina arasında değişebilen bir tasarım parametresi olarak belirlenmiştir. Senaryo analizlerinde, düşük ve yüksek seviyeler için 4 ve 8 gomina kullanılmıştır. Mevsimler, sis realizasyonlarındaki mevsimsel farklar ve gece/gündüz uzunluklarındaki farklardan dolayı Boğaz transit geçişlerini etkiler. Senaryolarda mevsim yaz ve kış olarak iki farklı seviyede değerlendirilmiştir.

Belirlenen 64 senaryonun 10 yinelemesi altı aylık dönemler için koşturulmuştur. Her senaryonun çıktıları toparlanmış ve Design Expert 7.0 yazılımı kullanılarak belirlenen faktörlerin seçilmiş tepki değişkenleri üzerindeki etkileri analiz edilmiştir.

Önemli faktörlerin tepki değişkenleri üzerindeki etkilerinin oranları Tablo 3'te verilmiştir. Bu tabloda pozitif ektiler (+) işaretle, negatif etkiler (-) işaretle gösterilmektedir. Örneğin, geçen gemi sayısındaki değişimin % 98.72' sine gemi geliş sıklığı sebep olmaktadır. Görüldüğü üzere, genelde tepki değişkenlerini etkileyen en önemli faktörler geliş sıklığı, kılavuz kaptan/römorkör sayıları ve takip mesafesidir.

Gemi geliş sıklığı, geçen gemi sayısını ve Boğazdaki gemi yoğunluğunu önemli ölçüde etkilemektedir. Bunun yanı sıra, kuyruklardaki gemi sayısını ve ortalama-maksimum bekleme sürelerini etkileyen en önemli faktördür. Ayrıca, kılavuz/römorkör sayıları kuyruklardaki gemi sayısını ve ortalama-maksimum bekleme sürelerini etkileyen en önemli ikinci faktördür ve tabiî ki kılavuz ve römorkör kulanım oranlarını da önemli ölçüde etkilemektedir. Takip mesafesinin gemilerin birbirini geçiş koşulları üzerindeki önemi sebebiyle transit sürelerine etkisi büyüktür. Bu faktör ayrıca gemi yoğunlukları ve ortalama bekleme sürelerini de büyük ölçüde etkilemektedir. Diğer daha az önemli faktörlerden mevsim en çok bekleme sürelerini (görüş koşulları üzerindeki etkisi sebebiyle), gemi profili ise kuyruklardaki gemi sayısını ve bekleme sürelerini etkiler.

Tablo 3. Önemli faktörlerin tepki değişkenlerinin varyansı üzerindeki yüzdelik katkıları

	A	B	C	D	E	F	G
Tepki değişkenleri / Faktörler	Geliş Sıklığı	Gemi Profili	K. Kaptan Römorkör	Akıntı Profili	Kandilli Kuralı	Takip Mesafesi	Mevsim
Geçen gemi sayısı	(+) 98.72		(-) 0.08				(-) 0.02
Gemilerin ort. geçiş süresi	(+) 8.02	(+) 0.10	(-) 1.67	(+) 9.76		(+) 77.26	
Gemilerin ort. bekleme süresi	(+) 41.00	(+) 7.79	(+) 30.02	(+) 0.62	(+) 0.14	(+) 2.32	(+) 9.03
Gemilerin maks. bekleme süresi	(+) 37.23	(+) 4.07	(+) 28.60	(+) 2.50		(+) 0.49	(+) 8.34
Kuyruklardaki gemi sayısı	(+) 48.19	(+) 7.17	(+) 30.79	(+) 0.29	(+) 0.14	(+) 2.68	(+) 3.84
Boğazdaki gemi yoğunluğu	(+) 91.68		(-) 0.39	(+) 0.70		(+) 5.73	
K.kaptan kullanım oranı	(+) 26.6		(+) 70.40	(+) 0.24		(+) 1.91	
Römorkör kullanım oranı	(+) 15.1		(+) 83.15	(+) 0.17		(+) 0.91	

Akıntı profili geçiş zamanını ve maksimum bekleme süresini etkilerken, Kandilli kuralının en az etkili faktör olduğu gözlemlenmiştir.

Tablo 4 bazı senaryoların temel senaryo (yani, mevsimin kış, Kandilli kuralının normal seviyesinde ve diğer faktörlerin Tablo 4'ün ilk satırındaki değerlerle tanımlı olduğu senaryo) ile karşılaştırmalarını, çıktıların nispi değerleri üstünden, göstermektedir. Bu karşılaştırma faktörlerin çıktılar üzerindeki tek tek ve birlikte etkilerini algılamayı güçlendirerek, model sonuçlarının ilginç yorumlarına yol açmaktadır.

Tablo 4. Seçilmiş senaryoların temel senaryo ile karşılaştırması

Gemi Profili	Geliş Sıklığı	K.Kaptan Römorkör	Akıntı Profili	Takip Mesafesi	Geçen Gemi	Ort. Geç. Sür.	Ort. Bekleme	Max. Bekleme	Kuyruk. Gemi	Boğaz Yoğ.	K.Kaptan Kullanımı	Römorkör Kullanımı
Normal	Normal	15/6	Normal	8 gomina	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
HazMat	Yüksek	15/6	Normal	8 gomina	1.15932	1.01566	4.02211	7.78457	5.32738	1.17711	1.16683	1.14922
Normal	Yüksek	20/9	Normal	8 gomina	1.17328	1.02348	1.51278	1.46301	1.51786	1.20057	0.90285	0.80743
HazMat	Normal	20/9	Normal	8 gomina	1.00014	1.00587	0.85602	0.71088	0.84524	1.00567	0.75578	0.67559
Normal	Yüksek	15/6	Yüksek	8 gomina	1.16240	1.03229	3.47543	5.63412	4.18155	1.20009	1.19531	1.18218
HazMat	Normal	15/6	Yüksek	8 gomina	0.99634	1.01468	2.10074	2.76693	1.91071	1.01118	1.00838	1.00679
Normal	Normal	20/9	Yüksek	8 gomina	0.99976	1.01957	0.60835	0.72902	0.61310	1.01936	0.76717	0.68817
HazMat	Yüksek	20/9	Yüksek	8 gomina	1.16990	1.04012	2.82703	2.89334	2.74702	1.21725	0.91323	0.81802
Normal	Yüksek	15/6	Normal	4 gomina	1.16674	0.97162	2.64717	4.47892	2.93452	1.13303	1.12395	1.12165
HazMat	Normal	15/6	Normal	4 gomina	0.99843	0.96380	1.48821	1.82794	1.34524	0.96206	0.95846	0.96165
Normal	Normal	20/9	Normal	4 gomina	0.99991	0.96575	0.45233	0.44538	0.47917	0.96584	0.72462	0.64942
HazMat	Yüksek	20/9	Normal	4 gomina	1.17214	0.97652	1.98133	2.08159	1.88393	1.14452	0.85662	0.76608
Normal	Normal	15/6	Yüksek	4 gomina	0.99980	0.97652	0.97371	1.13732	0.91964	0.97591	0.97454	0.98122
HazMat	Yüksek	15/6	Yüksek	4 gomina	1.16205	0.98630	3.61081	7.15573	4.49405	1.14594	1.13333	1.12745
Normal	Yüksek	20/9	Yüksek	4 gomina	1.17402	0.99119	1.23784	1.27718	1.20238	1.16341	0.87270	0.78366
HazMat	Normal	20/9	Yüksek	4 gomina	1.00024	0.98141	0.79459	0.86520	0.72619	0.98142	0.73702	0.66141

Bu çalışmadan elde edilen bir başka sonuç da, belli tipteki uğraklı geçiş yapan gemilere (UG) verilebilinecek önceliklerin genel etkisi üzerinedir. Uğraksız geçen gemi (USG) seyir planında Türk Boğazları bölgesinde bir liman bulunmayan gemidir. UG ise seyir planında, Türk Boğazları bölgesinde bir liman bulunan gemi olarak tanımlanabilir [2]. UG'nin stratejik önemi olan servisler sağlama potansiyeli sebebi ile (belirtilen bölgedeki, küçük lojistik gecikmelerin bile büyük sorunlara yol açabileceği, büyük rafinerilere petrol taşımak gibi), GTHM bu statüde bulunan gemilere öncelik tanımak isteyebilir. Öte yandan, bu tür öncelikler USG için olumsuzluk olarak görülebilir. Bu tür bir öncelik kararının etkilerini gözlemlemek amacıyla, "destek katsayısı (DK)" adında bir seri parametre tanımlanmış ve her UG'nin bekleme süresi ilgili DK ile çarpılarak bu gemiler için "uyarlanmış bekleme süreleri" hesaplanmıştır. Gelen gemiler kuyruklarda bekleme sürelerine bakılarak sıralandığı için, bu şekilde, yükseltilmiş DK değeri ile bahsi geçen UG'ye bir öncelik sağlanmaktadır. DK'nın genel ve bireysel etkisini analiz etmek üzere temel senaryo üzerinde bir seri benzetim koşumu gerçekleştirilmiştir. Bu koşumların farklı DK'ların bekleme süreleri üzerindeki etkilerini açıklayan ilk sonuçlarının, hiçbir önceliğin sağlanmadığı temel senaryoyla karşılaştırması Tablo 5'te özetlenmiştir. Sonuçlar, DK'lar dikkatli seçildiği takdirde, UG'lere sağlanan önceliğin USG'lerin maruz kaldığı olumsuzluktan yüksek olacağını göstermektedir.

Tablo 5. Farklı senaryoların DK=1.0 vakasından sapmaları

	Tüm gemiler			**Tüm USG'ler**			**Tüm UG'ler**		
	DK=1.50	*DK=1.75*	*DK=2.0*	*DK=1.50*	*DK=1.75*	*DK=2.0*	*DK=1.50*	*DK=1.75*	*DK=2.0*
Ort. Bekleme Süresi	%1,68	%3,80	%4,42	%12,15	%18,26	%21,88	%-15,75	%-20,32	%-24,68
Maks. Bekleme Süresi	%10,79	%19,78	%23,18	%11,05	%20,06	%23,47	%-24,48	%-28,96	%-35,78

8. SONUÇ

Bu çalışmada İstanbul Boğazı deniz transit trafiğinin bir benzetim modeli ve senaryo analizleri ile bu modelden elde edilen sonuçlar tartışılmaktadır. Sonuçlar geliş sıklığının, kılavuz kaptan/römorkör sayılarının, takip mesafesi ve gemi önceliklerinin önemine dikkat çekmektedir. Ayrıca, akıntı ve görüş mesafesi gibi, meteorolojik şartların transit trafik üzerindeki etkilerinin altı çizilmiştir.

Elde edilen sonuçlar, zor meteorolojik koşulların transit trafikteki artış ve gemi profillerindeki değişiklikler ile birleşmesi sonucu, istenmeyen trafik yoğunluklarının oluşabileceğini ve bunun da Boğazdaki risk seviyesini önemli ölçüde arttırabileceğini göstermektedir. Öte yandan, GTHM'nin sahip olduğu yönetim ve kontrol araçları bahsi geçen yoğunluğu ve riski kontrol altına almakta çok etkili olabilmektedir.

9. GELECEK ÇALIŞMALAR VE RİSK ÇALIŞMASI METODOLOJİSİ

Şu anda, sözü edilen benzetim çalışması, muhtelif genişletmeler, K&D'in fiili uygulamasında yapılan değişikliklerin eklenmesi, gemi geliş sıklıkları ve profillerinin dağılımlarının güncellemesi çalışmaları ile devam etmektedir. Transit trafik sistemini

etkileyen tüm faktörlerin kapsamlı incelenmesi, genişletilmiş senaryo analizleri ile entegre modelde hayata geçirilecektir.

Bunun yanı sıra, kurulacak modelin İstanbul Boğazı için kapsamlı bir risk analizi çalışması yapmaya yönelik bir platform oluşturması beklenmektedir. Proje hedefinde vurgulanan risk analizi için, riskin iki temel öğesinin iyi tanımlanması gerekir; deniz kazalarının gerçekleşme olasılıkları ve kazaların gerçekleşmeleri halinde meydana gelebilecek zararlar. Bu iki öğe sağlıklı belirlenebilirse, risk bu öğelerin çarpımı olarak görülebilir. Birçok uzmanla yapılan görüşmeler ve literatür taramalar, muhtelif deniz kazalarının gerçekleşme olasılıklarının birbirlerinden, hatta geçmişte gerçekleşip istatistiklere yansımış olanlardan, çok farklı olabileceğine işaret etmektedir. Bu sebepten kazaların oluşumunu etkileyen faktörler ayrı ayrı belirlenmeli ve bu faktörlerin belli konumlarda olması halinde "şartlı kaza olasılıklarının" ne olacağı tahmin edilmelidir. Ancak, bu yaklaşım kaza olasılıklarının tahmininde geçmiş istatistiklerin doğrudan kullanılmasını zorlaştırmaktadır. Bu durumda, sözü edilen "şartlı kaza olasılıklarının" tahmini için, gene literatürdeki uygulamaların paralelinde, basit ve sade sorulardan oluşan anketlerle uzmanlardan sübjektif uzman görüşlerine başvurulması öngörülmüştür. Gerçekleşen bir kazanın yol açacağı zararın ne olabileceği (insan hayatına, yapıya ve çevreye) belirlenirken de benzer bir anket çalışması kullanılabilir. Sonuç olarak, tüm bu değerleri kurulan benzetim modeline entegre ederek, İstanbul Boğazı'nı geçmekte olan bir geminin, doğal ve insan yapımı çevre şartları (gemi tipi, boyu, kılavuz kaptan/römorkör durumu, akıntı ve sis durumu, lokasyonu, civardaki diğer gemilerin konumları v.s.) göz önünde tutularak, bulunduğu lokasyon etrafındaki kaza riski bu çarpımlarla ölçümlenmeye çalışılacaktır.

TEŞEKKÜR

Yazarlar bu araştırmaya destek veren TÜBİTAK'a (Araştırma Projesi No: 104Y207 aracılığı ile) ve Boğaziçi Üniversitesi Bilimsel Araştırma Projeleri Fonu'na (Araştırma Projesi No: 07M104 aracılığı ile) katkılarından dolayı teşekkür ederler.

KAYNAKÇA

[1] Öztürk, B., (1995), "Turkish Straits. New Problems, New Solutions", Chapter: The Istanbul Strait, A Closing Biological Corridor, ISIS, pp. 145-154.

[2] VTS Users Guide, (2004), "Turkish Straits Vessel Traffic Service", General Management of Coastal Safety and Salvage Administrations, 3rd edition, May 2004, Istanbul.

[3] Özbaş, B., (2005), "Simulation of Maritime Transit Traffic in the İstanbul Strait", M.S. Thesis, Department of Industrial Engineering, Boğaziçi University.

[4] Or, İ. and Kahraman, İ., (2001), "A Simulation Study of the Accident Risk in the Istanbul Strait", Int. J. of Emergency Management, Vol. 1, No. 2, pp. 110-124.

[5] Altıok, T., (1996), "Tools of Probability", Performance Analysis of Manufacturing Systems, pp. 15-65, Springer, New York.

[6] Almaz, A. Ö., (2006), "Investigation of Transit Maritime Traffic in the Strait of, Istanbul through Simulation Modeling and Scenario Analysis", M.S. Thesis, Department of Industrial Engineering, Boğaziçi University.

[7] Almaz, A.Ö., Or, İ, Özbaş, B. (2006), "Investigation of the Maritime Transit Traffic in the Istanbul Channel through Simulation Modeling and Scenario Analysis", International Journal of Simulation Systems, Science and Technology, Special Issue on: Simulation in Industry, Business, and Services, Volume 7, Number 7, p.p. 1– 9.

SİSTEM DİNAMİKLERİ YAKLAŞIMLI TERFİ SİSTEMİ ANALİZ MODELİ (TESAMOS)[12]

Mu.Yb.H.Umut AKIN[(a)], Dr.Y.Müh.Yb.Altan ÖZKİL[(a)]

[(a)]Genelkurmay BİLKARDEM Bşk.lığı, 06650 Bakanlıklar ANKARA,
uakin@tsk.mil.tr , aozkil@tsk.mil.tr

ÖZ

Silahlı Kuvvetler terfi sistemine ilişkin hali hazırda çeşitli çalışmalar yapılmaktadır. Bu kapsamda, çeşitli öngörülerde bulunulmakta ve değişik terfi sistemleri ve politikalar önerilmektedir. Önerilen bu modellerin; adil olma ve eşitliği sağlama, arzu edilen birikimi yaratma ve sistemi arzu edilen bir sürede dengeye ulaştırma gibi hedefleri sağlayıp sağlamadığının uygulamaya geçilmeden önce analiz edilmesi önem kazanmaktadır. Bu maksatla çalışmada Sistem Dinamikleri yaklaşımı kullanılarak önerilen terfi sistemleri modellenmiş ve çeşitli simülasyonlarla geleceğe yönelik politikaların geçerliliği incelenmiştir.

Anahtar Kelimeler : Sistem Dinamikleri, Modelleme ve Simülasyon, Terfi Sistemleri

PROMOTION SYSTEM ANALYSIS MODEL WITH SYSTEM DYNAMICS APPROACH

ABSTRACT

There are a variety of studies being conducted concerning promotions which is a subsystem of Armed forces manpower management system. In this context, various promotion systems and policies are being proposed. The most important part of the process of determining over such proposals is to analyze them considering some criteria such that the proposed system is able to sustain equality, build necessary accessions in the future and bring the system to a steady state in the short term before it is put in effect. To meet these requirements, the System Dynamics approach has used in the study to model the proposed promotion systems and the validity of various policies has determined.

Key Words : System Dynamics, Modeling and Simulation, Promotion Systems

1. GİRİŞ

Günümüzde İKY'deki gelişmeler, Silahlı Kuvvetlerde objektif terfi sistemlerinin sübjektif olanlardan daha fazla değer görmesine sebep olmaktadır. Bu yüzden, Silahlı Kuvvetler en uygun terfi sistem ve kriterlerini kendi hiyerarşik yapısına katma çabasındadır. Böylelikle mevcut ile ihtiyaç duyulan personel miktarı arasındaki fark en aza indirgenmiş olacaktır.

[12] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

Silahlı Kuvvetler terfi sistemine ilişkin hali hazırda çeşitli çalışmalar yapılmaktadır. Bu kapsamda, çeşitli öngörülerde bulunulmakta ve değişik terfi sistemleri ve politikalar önerilmektedir. Önerilen bu modellerin; adil olma ve eşitliği sağlama, arzu edilen birikimi (mahrut yapısı) yaratma ve sistemin arzu edilen bir sürede dengeye ulaşma gibi hedefleri sağlayıp sağlamadığının uygulamaya geçilmeden önce analiz edilmesi önem kazanmaktadır. Bu maksatla çalışmada Sistem Dinamikleri (SD) yaklaşımı kullanılarak önerilen terfi sistemleri modellenmiş ve çeşitli simülasyonlarla geleceğe yönelik politikaların geçerliliği incelenmiştir.

2. TERFİ SİSTEMLERİ

Terfi, en genel anlamıyla; personelin organizasyonun hiyerarşik yapısı içerisinde daha fazla yetki ve sorumluluğa sahip olarak üst seviyeye yükselmesidir. Terfinin sözlükteki anlamı ise; "bir kişiyi belli bekleme süreleri veya değerlendirme testleri sonrasında daha fazla imtiyaz ve ücretlerle daha yüksek veya daha iyi bir pozisyona yükseltmek" olarak tanımlanmaktadır. [1]

Bir çok ücret artışları terfi ile ilişkilendirilmekle beraber yükselmedeki en belirgin özellik yetki ve sorumluluk artışıdır.[2][3]

Terfi sisteminin öncelikli iki temel niteliği adil ve eşitlikçi olmasıdır ve sistem sübjektif ve önyargılı değerlendirmenin etkisini minimum seviyeye indirecek şekilde dizayn edilmelidir. [4]

Terfi sistemleri dört grupta toplanabilir:

- Mutlak ve göreceli değerlere dayalı sistemler
- Yukarı veya dışarı sistemler
- Kıdeme dayalı sistemler
- Rassal sistemler

3. SİSTEM DİNAMİKLERİ YAKLAŞIMI

SD, sıvı akışı simülasyonunun bilgisayar ortamına uyarlanması şeklinde düşünülebilir. Model depolar (stoklar olarak da adlandırılır) ve sıvı akışından (veya basitçe akış) oluşmaktadır. Depolara gelen veya depolardan çıkan akışlar, karar vericiler tarafından belirlenen kurallara göre açılıp kapatılan vanalarla kontrol edilmektedir. Stoklar ve akışların seviyeleri hakkındaki eğilim bilgileri karar vericiler tarafından takip edilebilir ve bu bilgilere dayanarak vanaların açılıp kapatılmasına karar verilebilir.

Sıvı akışı modellemesi ile Silahlı Kuvvetlerdeki personel akışı arasında paralellik mevcuttur.[5] Depolar aday personeli, görevdekileri veya Silahlı Kuvvetlerdeki tüm personeli temsil edebilir. Akışlar; eğitim sistemleri, yada mesleki gelişimi ifade eden rütbeler arasında olabilir.

SD yaklaşımının bu tür kuvvet yapısı problemlerinde kullanılmasının bir başka ve daha da önemli özelliği de geri besleme kavramıdır.[6] Geri beslemeli sistemlerin davranışını anlamak SD yaklaşımının hedefidir.[7]

4. SİLAHLI KUVVETLER TERFİ SİSTEMİ ve MODELLEMESİ

Silahlı Kuvvetler yapısında terfi, askeri personelin daha fazla ehliyet isteyen bir rütbeye yükseltilmesi, yahut eski rütbe veya derecesinin üstünde sorumluluklarının

artırılmasıdır.[8] Terfi sisteminin ana öğeleri; görevleri yerine getirmek üzere tesis edilen kadro yapısı, bu kadroların rütbelere dağılımını gösteren mahrut yapısı, birikimi tesis edecek ve farklı başarı/tecrübe seviyelerindeki personelin farklı zamanlarda terfisine olanak sağlayacak her bir rütbe için uygulanacak rütbe bekleme süreleri, aday konumda bulunanlar arasından terfi edeceklerin seçimini sağlayacak liyakate dayalı değerlendirme sistemi ve hizmette bulunan tüm personelden mesleğin tüm aşamalarında en etkin şekilde yararlanmayı mümkün hale getirecek motivasyon sistemi olmalıdır.

4.1 Yeni Terfi Sistemi Prensipleri

Yeni terfi sistemi personelin bulunduğu rütbedeki toplam performansı ve potansiyelini dikkate alacaktır. Bu nedenle de değerlendirme sistemi objektif değerlere dayanacaktır.
Yeni sistemde ortalama toplam hizmet süresi 31 yıldır. Sisteme her yıl 100 personel Teğmen olarak katılmaktadır. Teğmen rütbesinden Üsteğmen rütbesine terfiler 3ncü yıl sonunda gerçekleşmekte ve tüm personel terfi etmektedir. Üsteğmen rütbesinden Yüzbaşı rütbesine ise yine tüm personel fakat farklı zamanlarda terfi etmektedirler.

4.2. Mahrut Yapısı Hesaplamaları

Modellemede 3'lü kontrol alanı yaklaşımı kabul edilmiş olup; sevk ve idare seviyesi olarak Albay rütbesi bir seviye, Yarbay-Binbaşı rütbeleri bir seviye ve Yüzbaşı-Üsteğmen-Teğmen rütbeleri bir seviye olarak kabul edilmiştir. Aşağıya doğru her bir seviyede kendisinden üst seviyenin 3 katı personel bulunması öngörülmektedir. Bu yapı içerisinde her seviyedeki personel sayısı hesabı aşağıda gösterilmektedir.

$$\textit{Albay Sayısı} = x \quad (1)$$

$$\textit{Yarbay-Binbaşı Sayısı} = 3x \quad (2)$$

$$\textit{Yzb.-Ütğm.-Tğm. Sayısı} = 9x \quad (3)$$

Yüzde olarak Albay sayısı ise:

$$x + 3x + 9x = 100 \quad (4)$$

$$x = \% 7{,}70 \quad (5)$$

Hesaplamalar her bir seviye için yapıldığında Alb. : %7.70, Yb.-Bnb. : %23.10, Yzb.-Ütğm.-Tğm. : %69.20 olarak bulunur.
Hesaplamaların, aynı seviyedeki rütbelere dağılımı ise o rütbedeki ortalama bekleme süresinin yukarıda bulunan yüzdelere dağıtılması ile hesaplanabilir. Örneğin Yarbay rütbesinde asgari bekleme süresi 3 azami bekleme süresi 6 yıldır. Ortalama bekleme süresi ise;

$$\textit{Yb.Ort.Bek.Süresi} = \textit{Asgari Süre} + (\textit{Azami Süre} - \textit{Asgari Süre} + 1) / 2 \quad (6)$$
$$= 3 + (6 - 3 + 1) / 2$$
$$= 5 \textit{ yıl}$$

olarak bulunur.
Bnb. Rütbesi için de ortalama bekleme süresi 5 yıl olarak bulunur. Yb. ve Bnb. Rütbeleri için ise ortalama toplam bekleme süresi 10 yıldır. Dolayısıyla Yarbay rütbesindeki personel yüzdesi;

%Yb. = (% Yb-Bnb x Yb.Ort.Bek.Süresi) / Yb.-Bnb.Ort.Top.Bek.Süresi **(7)**

= (23,10 x 5) / 10

= 11,55

hesaplaması ile bulunabilir. Tüm rütbeler için hesaplamalar yapılırsa Çizelge 1'deki sonuçlar elde edilir.

Çizelge 1. Rütbelerdeki personel yüzdeleri

Rütbe	Asgari	Azami	Ortalama	%
Albay	9	9	9	7,70
Yarbay	3	6	5	11,55
Binbaşı	3	6	5	11,55
Yüzbaşı	3	9	6	27,68
Üsteğmen	3	9	6	27,68
Teğmen	3	3	3	13,84

Sistemdeki toplam personel sayısı ise; sisteme her yıl 100 kişi girdiği dikkate alındığında ve Teğmen ve Üsteğmen rütbelerinde personelin tamamı terfi ettiğinden bu rütbelerdeki ortalama birikim;

Ort.Birikim = (Ütğm.Ort.Bek.sür. x 100)+(Tğm.Ort.Bek.sür. x 100) (8)

= (6 x 100) + (3 x 100)

= 900

olarak hesaplanır. Üsteğmen ve Teğmen toplamının genel toplamdaki yüzdesi;

% Ütğm.+Tğm. = 27,68 + 13, 84

≈ 41,50

Buradan da genel personel Mevcudu;

Genel Mevcut = (900 x 100) / 41,50

≈ 2168

Benzer şekilde diğer rütbeler için hesaplanan personel miktarları da Çizelge 2'de sunulmaktadır.

Çizelge 2. Rütbelere göre personel mevcutları

Rütbe	Hesaplama	Mevcut
Albay	(7,70 x 2168) / 100	168
Yarbay	(11,55 x 2168) / 100	250
Binbaşı	(11,55 x 2168) / 100	250
Yüzbaşı	(27,68 x 2168) / 100	600
Üsteğmen	(27,68 x 2168) / 100	600
Teğmen	(13,84 x 2168) / 100	300
TOPLAM		2168

Personelin rütbelere dağılımı o rütbedeki ortalama bekleme süresine bölündüğünde, anılan rütbedeki birikimi tesis etmek üzere her yıl, o rütbeye terfi etmesi gereken miktar hesaplanabilir. Örneğin Albay olacak Yarbay sayısı;

Terfi Edecek Miktar = Albay Sayısı / Alb.Ort.Bek.Süresi **(9)**

= 168 / 9

≈ 19

olarak bulunur. Aynı şekilde diğer rütbeler için terfi ve Teğmenliğe giriş miktarları Çizelge 3'teki gibi hesaplanır.

Çizelge 3. Terfi miktarları hesaplaması

Rütbe	Hesaplama	Terfi
Albay	168 / 9	19
Yarbay	250 / 5	50
Binbaşı	250 / 5	50
Yüzbaşı	600 / 6	100
Üsteğmen	600 / 6	100
Teğmen	300 / 3	100

4.3. Terfi Modelleri

Bu çalışmada, SD yaklaşımı ile yapılan modellemenin seviyesi yüksek ve çözünürlüğü düşük olduğundan modellemede tek tek bireyler değil her bir rütbedeki toplam mevcutlar dikkate alınmaktadır.

Bu kapsamda, personel sicil yeterlilikleri ve deneyim süreleri dikkate alınarak dört model önerilmektedir. Modeller; terfiye aday olunan yıllardaki yeterlilik oranlarının artan, azalan ve normal dağılım olabileceği veya rassal bir özellik gösterebileceği ya da daha yüksek çözünürlüklü modellerle elde edilebileceğine dayanmaktadır.

4.3.1. Model-1 : Eşit olasılıklı terfi modeli

Eşit olasılıklı terfi modeli; terfi etmeye hak kazanan, diğer bir ifade ile rütbede asgari bekleme süresini tamamlayan personel ile aynı rütbede azami rütbe bekleme süresini tamamlamış tüm personeli tek bir havuzda değerlendiren ve müteakiben de anılan personeli personel yeterlilik derecelerini dikkate alarak terfi ettiren bir modeldir.

4.3.2. Model-2 : Rütbede geçirilen yılları dikkate alan terfi modeli

Rütbede geçirilen yılları dikkate alan terfi modeli; asgari rütbe bekleme ile azami rütbe bekleme süreleri içerisinde yer alan personelin rütbede geçirmiş olduğu yılları, diğer bir ifade ile deneyimlerini dikkate alan bir modeldir. Bu modelde temel esas rütbedeki bekleme yıllarından idare tarafından belirlenecek oranlarda terfiye müsaade edilmesidir.

Örneğin modelde, 3ncü yılını bitiren personel grubundan terfi edecek personelin %5'lik kısmına karşılık gelenlerin terfi etmesi sağlanmaktadır.

4.3.3. Model-3 : İlk ve son yılları belli oranlarda garantili diğer yılları eşit olasılıklı terfi modeli.

İlk ve son yılları belli oranlarda garantili diğer yılları eşit olasılıklı terfi modelinde temel amaç; asgari rütbe bekleme süresini tamamlamış, diğer bir ifade ile erken terfiye aday personel ile bulunduğu rütbede azami bekleme süresini tamamlamış, diğer bir ifade ile terfi için son hakkı olan personelin idare tarafından motive edilerek en iyilerin terfisinin garanti altına alınmasıdır. İlk ve son yıllarında bulunanlar haricindeki personel için Model-1'de kabul edilen eşit olasılıklı yaklaşım kullanılmaktadır.

Söz konusu modelin uygulanmasında, idare tarafından terfiye aday olunan rütbedeki ilk yıl ve son yıl terfi oranlarının %5 olması öngörülmüştür. Tespit edilen bu oran, terfiye

aday olunan ilk ve son yılında bulunanlardan yeterliliği olsa bile bazı personelin terfisini engelleyecektir.

4.3.4. Model-4 : Kademeli eşit olasılıklı terfi modeli.

Kademeli eşit olasılıklı terfi modeli, tüm personeli eşit olasılıklarla kademeli havuzlarda değerlendirmek suretiyle terfi ettirmeyi esas almaktadır. Bunun amacı, personeli aynı potada değerlendirmekle birlikte rütbelerdeki bekleme yıllarındaki deneyimleri de kademelerin oranlarını belirleyerek kontrol etmektir.

Bu modelde x1, x2, x3, ... , xn değişkenleri oluşturulan n kademede terfi için kabul edilecek oranlardır. Söz konusu kademelere ilişkin belirlenen oranların toplamı 1 olmalıdır. Örneğin; x1= 0,10, x1 = 0,20 ise modelin uygulanmasında ilk olarak terfiye hak kazanan tüm personel bir havuzda toplanarak terfi edecek 63 personelin %10'luk kısmına karşılık gelen 6 personel bu havuzdan sicil yeterliliklerine bakılmak suretiyle seçilecektir. Müteakiben rütbedeki 3ncü yılını dolduranlar havuzdan çıkarılarak diğer kalanlar terfi edecek personelin %20'sini oluşturan 12 personelin seçilmesi için yeni bir havuz oluşturulmaktadır. Modelde personel seçimi, terfiye aday son iki yılındaki personelden oluşan kademedeki havuzdan terfi edeceklerin belirlenmesi ile o rütbe için sonlanmaktadır.

4.4. Model etkinlik ölçütleri

Önerilen dört model etkinlik ölçütleri dikkate alınarak incelenecektir.Bunlar;

- Kontrol alanı oranları (KAO)
- Rütbe bekleme süreleri (RBS)
- Ortalama hizmet Süreleri (OHS)
- Rütbe-yıl terfi olasılıkları (TO)
- En erken terfi olasılığı (ETO)
- En geç terfi olasılığı (GTO)

4.5. Modelin Vensim® Uygulaması

Yukarıda açıklanan modeller Vensim® ortamına sistemin de anlaşılmasına imkan verecek şekilde kolaylıkla aktarılmıştır.
Buna göre 4 alternatif için tek bir model kurulmuştur. Model parametrelerinden mahrut yapısı hesaplamalarının yapılabilmesi, model tipinin seçilebilmesi, rütbe bekleme süreleri ile ilgili parametrelerin takip ve kontrolünün kolay olması için anılan parametreler bir MS Excel® tablosunda bulunmaktadır.

4.5.1. Model-1: Eşit olasılıklı terfi modeli

Terfi modellerinde sistem bitişli bir sistem olarak kabul edilmiştir. Bu kapsamda Model-1 100 yıllık bir dönem için çalıştırılmıştır. Sistemin etkinlik ölçütleri bazında elde edilen sonuçlar kullanılarak dengeye ulaşma zamanları incelenmiştir. Etkinlik ölçütlerinden en uzun sürede dengeye ulaşan ölçüte göre hareketli ortalamalar metodu kullanılarak denge noktası tespit edilmiştir. Bu kapsamda sistemin bu andan sonraki 40 yıl süreli davranışı dikkate alınmıştır.

Isınma periyodu tespitinden sonra geliştirilen model kullanılarak yapılacak tekrar sayısının belirlenebilmesi için 10 tekrar yapılmıştır. Tekrar sayısının belirlenmesi işlemi

de etkinlik ölçütleri bazında gerçekleştirilmiştir. Etkinlik ölçütleri bazında hesaplanan tekrar sayılarından en büyük olanı model için gerekli tekrar sayısı olarak belirlenmiştir. Burada; $\alpha = 0{,}05$ ve β = etkinlik ölçütü$_{ort}$ * 0,005 kabul edilerek

$$i \geq s^2_{(n)} (z_{1-\alpha/2} / \beta) \qquad (10)$$

eşitsizliği kullanılarak tekrar sayısı hesaplanmıştır. Buna göre ETO etkinlik ölçütü için;

$s^2_{(n)} = 1{,}12952E\text{-}06,$
$z_{1-\alpha/2} = z_{0{,}975} = 1{,}96,$
$\beta = ETO_{ort} * 0{,}005 = 0{,}0055 * 0{,}005 = 0{,}000275$
$i \geq 220$ ve buradan da $i = 221$

Model-1 ile yapılması gereken tekrar sayısı 221 olarak bulunmuştur. Eşit olasılıklı terfi modeline ilişkin ilk 10 tekrara ilave olarak 211 tekrar daha yapılmış ve etkinlik ölçütlerine ait %95 seviyesinde güven aralıkları oluşturularak Çizelge 4'te sunulmuştur.

4.5.2. Model-2 : Rütbede geçirilen yılları dikkate alan terfi modeli

Rütbede geçirilen yılları dikkate alan terfi modelinde rütbedeki her bekleme yılına karşılık gelen terfi oranlarının tespit edilmesi gerekmektedir. Anılan oranların tespitinde, geçmiş yıllara ait sicil paternleri kullanılarak değişik sicillere ilişkin analizler yapılmıştır. Bu analizler sonucunda elde edilen paternlerin beta dağılımına uygun olduğu gözlemlenmiş, bu maksatla değişik α ve β değerleri kullanılarak farklı sicil eğrileri yaratılmıştır. Bu maksatla $\alpha = 0{,}6$, 1,2 ve 1,8 ve $\beta = 0{,}8$, 1,0 ve 1,2 değerleri için bunların kombinasyonu olan 9 değişik sicil senaryosu kullanılarak Model-2'nin etkinlik ölçütleri belirlenmiştir.

Çizelge 4. Model-1 etkinlik ölçütleri için güven aralıkları.

	Xort	StdDev	h	*l*	*u*
Alb/Bnb-Yb	2.742	0.031	0.004	2.738	2.746
Bnb-Yb/Tgm-Utgm-Yzb	3.443	0.042	0.006	3.437	3.448
Alb -Top	171	0	0	171	171
Bnb-Yb Top	469	5	1	468	470
Tgm-Utgm-Yzb Top	1614	4	1	1613	1614
Utgm RBS	6.031	0.020	0.003	6.028	6.034
Yzb RBS	5.999	0.071	0.009	5.989	6.008
Bnb RBS	3.156	0.083	0.011	3.145	3.167
Yb RBS	3.998	0.048	0.006	3.991	4.004
Alb RBS	5.000	0.000	0.000	5.000	5.000
Utgm OHS	9.031	0.020	0.003	9.028	9.034
Yzb OHS	15.030	0.072	0.010	15.020	15.039
Bnb OHS	18.185	0.104	0.014	18.171	18.199
Yb OHS	22.183	0.117	0.015	22.168	22.199
Alb OHS	27.183	0.117	0.015	27.168	27.199
Utgm Yzb TO	0.248	0.000	0.000	0.248	0.248
Yzb Bnb TO	0.123	0.001	0.000	0.123	0.124
Bnb Yb TO	0.869	0.063	0.008	0.861	0.877
Yb Alb TO	0.112	0.001	0.000	0.112	0.112
En Erken Terfi Olasılığı	0.006	0.001	0.000	0.005	0.006
En Geç Terfi Olasılığı	0.000	0.000	0.000	0.000	0.000
n=	**221**				
t220,975=	**1.9719**				

Değişik α ve β değerlerini içeren 9 tekrar için sistemin dengeye ulaştığı yıldan itibaren müteakip 40 yılın ortalamaları Çizelge 5'tedir. Bu sonuçlara göre etkinlik ölçütlerinin beklenen değerlerine göre en uygun modelin Model 2G olduğu görülmektedir.

Çizelge 5. Model-2 etkinlik ölçütleri ortalamaları

	2A α=0.6 β=0.8	2B α=0.6 β=1.0	2C α=0.6 β=1.2	2D α=1.2 β=0.8	2E α=1.2 β=1.0	2F α=1.2 β=1.2	2G α=1.8 β=0.8	2H α=1.8 β=1.0	2I α=1.8 β=1.2
Alb/Bnb-Yb	2.71	2.68	2.66	2.91	2.88	2.85	3.00	2.98	2.96
Bnb-Yb/Tgm-Utgm-Yzb	3.43	3.41	3.40	3.33	3.32	3.32	3.31	3.29	3.27
Alb -Top	171	171	171	171	171	171	171	171	171
Bnb-Yb Top	463	458	455	497	492	487	513	510	507
Tgm-Utgm-Yzb Top	1586	1562	1547	1656	1634	1617	1696	1676	1659
Utgm RBS	6.03	6.03	6.03	6.03	6.03	6.03	6.03	6.03	6.03
Yzb RBS	5.56	5.17	4.94	6.67	6.32	6.05	7.30	6.98	6.71
Bnb RBS	3.13	3.08	3.03	3.56	3.49	3.43	3.73	3.70	3.67
Yb RBS	3.79	3.68	3.68	4.16	4.11	4.05	4.42	4.37	4.32
Alb RBS	5.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00
Utgm OHS	9.03	9.03	9.03	9.03	9.03	9.03	9.03	9.03	9.03
Yzb OHS	14.59	14.21	13.97	15.70	15.35	15.08	16.33	16.01	15.74
Bnb OHS	17.71	17.29	17.00	19.25	18.84	18.51	20.06	19.71	19.41
Yb OHS	21.50	20.97	20.68	23.41	22.94	22.56	24.48	24.08	23.73
Alb OHS	26.50	25.97	25.68	28.41	27.94	27.56	29.48	29.08	28.73
Utgm Yzb TO	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25
Yzb Bnb TO	0.13	0.14	0.14	0.11	0.12	0.12	0.11	0.11	0.11
Bnb Yb TO	0.89	0.93	0.97	0.64	0.67	0.70	0.58	0.59	0.60
Yb Alb TO	0.11	0.12	0.12	0.11	0.11	0.11	0.11	0.11	0.11
En Erken Terfi Olasılığı	0.02	0.04	0.04	0.00	0.00	0.00	0.00	0.00	0.00
En Geç Terfi Olasılığı	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

4.5.3. Model-3 : İlk ve son yılları belli oranlarda garantili diğer yılları eşit olasılıklı terfi modeli.

Modelde, terfi oranları bulunulan rütbenin terfiye aday ilk ve son yıllarında bulunanlar için %5 olarak belirlenmiştir. Bunun haricindeki personel için ise Model-1'deki gibi rassal terfi oranları kullanılmıştır.

Eşit olasılıklı terfi modeline ilişkin ilk 10 tekrara ilave olarak 48 tekrar daha yapılmış ve etkinlik ölçütlerine ait %95 seviyesinde Model-1'deki gibi güven aralıkları oluşturulmuştur.

4.5.4. Model-4 : Kademeli eşit olasılıklı terfi modeli.

Kademeli eşit olasılıklı terfi modeli koşturumunda her kademede oluşturulan havuzların oranları *beta* dağılımı ile artan ve azalan olarak tespit edilmiş, havuzlar içerisindeki terfi yıllarına ilişkin oranlar ise Model-1'deki gibi rassal olarak kabul edilmiştir. Rütbelere göre her kademede oluşturulan havuzlara ilişkin artan terfi oranları Çizelge 6'da sunulmuştur. Etkinlik ölçütlerine ait %95 seviyesinde Model-1'deki gibi güven aralıkları oluşturulmuştur.

Çizelge 6. Rütbelere göre havuz kademelerinin oranları.

Havuzlar	Utgm	Yzb	Bnb	Yb
Kademe-1	0.06	0.04	0.29	0.23
Kademe-2	0.14	0.1	0.71	0.77
Kademe-3	0.21	0.15		
Kademe-4	0.27	0.19		
Kademe-5	0.33	0.24		
Kademe-6		0.28		

4.6. Model eniyilemesi sonuçları.

Modellerde dikkate alınan etkinlik ölçülerinin her bir model bazında aldıkları ortalama değerler incelendiğinde rütbelerdeki personel sayısı, ortalama rütbe bekleme süreleri, ortalama hizmet süreleri ve terfi olasılıklarının genelde benzerlik gösterdiği

belirlenmiştir. Yapılan çalışmanın temel amacının kontrol alanları üzerinde yoğunlaşmış olması nedeniyle modellerin arzu edilen 1/3 kontrol alanı oranını sağlaması için Vensim eniyileme modülü çalıştırılmıştır.

Modeller incelendiğinde; Bnb-Yb ile Alb. oranının genelde 2,5 civarında, Tgm-Utgm-Yzb ile Bnb-Yb oranının ise 3,5 civarında olduğu görülmektedir. Anılan oranların 3,0'e yaklaşması için Bnb-Yb sayısının arttırılmasının uygun olabileceği değerlendirilmiştir. Bunu sağlamak için ise yüzbaşı rütbesinden binbaşı rütbesine geçişte 0,626 olarak kullanılmakta olan terfi oranını Vensim eniyileme modülü kullanılarak 0,60 ile 0,65 arasında değiştirmek suretiyle kontrol alanı oranının modellerdeki davranışının nasıl değiştiği gözlemlenmiştir. Seçilen 4 model için uygulanan eniyileme sonuçları Çizelge 7'de sunulmaktadır.

Buna göre yüzbaşılıktan binbaşı rütbesine geçişteki 0,646 terfi oranı daha iyi sonuçlar vermektedir. Modelde kullanılan diğer parametreler için de benzer çalışmalar yapılarak model kontrol alanı için eniyilenebilir.

Çizelge 7. Terfi modelleri için eniyileme sonuçları.

Bnb-Yb/Alb	Önceki Değer	Sonraki Değer	Tgm-Utgm-Yzb / Bnb-Yb	Önceki Değer	Sonraki Değer
MODEL-1	2.742	2.948	MODEL-1	3.443	3.188
MODEL-2G	3.000	3.292	MODEL-2G	3.31	2.975
MODEL-3	2.766	3.240	MODEL-3	3.412	2.907
MODEL-4 Artan	2.920	3.101	MODEL-4 Artan	3.920	3.675

6. SONUÇ ve DEĞERLENDİRMELER

6.1. Genel Sonuçlar

Geliştirilen dört terfi sistemine ilişkin modeller çalıştırılarak elde edilen etkinlik ölçütleri bazındaki sonuçlar Çizelge 8'de sunulmaktadır.

Sonuçlar incelendiğinde; kontrol alanı etkinlik ölçütü açısından Alb/Bnb-Yb oranında Model-2 ve Model-4'ün diğer modellere göre arzu edilen oranı sağladığı, Bnb-Yb / Tgm-Utgm-Yzb oranında ise Model-2'nin diğer modellere göre hedeflenen oranı daha iyi yakaladığı, belirlenmiştir.

Ancak toplam personel sayısına bakıldığında; Model-1 ve Model-3'deki subay toplamının başlangıç hedefi olan 2168'e diğer modellere göre daha yakın oldukları tespit edilmiştir.

Çizelge 8. Etkinlik ölçütlerinin modeller bazında aldıkları değerler.

	Beklenen Değer	MODEL-1	MODEL-2	MODEL-3	MODEL-4
Alb/Bnb-Yb	3.000	2.742	3.00	2.766	4.013
Bnb-Yb/Tgm-Utgm-Yzb	3.000	3.443	3.31	3.412	3.649
Alb -Top	168	171	171	171	128
Bnb-Yb Top	500	469	513	473	513
Tgm-Utgm-Yzb Top	1500	1614	1696	1614	1871
Subay Toplam	2168	2254	2380	2258	2513
Utgm RBS	6.000	6.031	6.03	6.032	7.953
Yzb RBS	6.000	5.999	7.30	5.998	7.221
Bnb RBS	4.000	3.156	3.73	3.222	4.489
Yb RBS	4.000	3.998	4.42	4.000	4.491
Alb RBS	5.000	5.000	5.00	5.000	4.998
Utgm OHS	9.000	9.031	9.03	9.032	10.953
Yzb OHS	15.000	15.030	16.33	15.031	18.174
Bnb OHS	19.000	18.185	20.06	18.253	22.663
Yb OHS	23.000	22.183	24.48	22.253	27.154
Alb OHS	29.000	27.183	29.48	27.253	32.153
Utgm Yzb TO		0.248	0.25	0.248	0.169
Yzb Bnb TO		0.123	0.11	0.123	0.109
Bnb Yb TO		0.869	0.58	0.818	0.395
Yb Alb TO		0.112	0.11	0.112	0.147
En Erken Terfi Olasılığı		0.006	0.00	0.000	0.000
En Geç Terfi Olasılığı		0.000	0.00	0.000	0.000

Terfi sistemlerinden beklenen ortalama rütbe bekleme süreleri; Üsteğmen rütbesi için 6, Yüzbaşı rütbesi için 6, Binbaşı rütbesi için 5, Yarbay rütbesi için 5 ve Albay rütbesi için 6 yıl olarak belirlenmiştir. Bu esaslar dikkate alınarak her model için gerçekleşen ortalama rütbe bekleme süreleri kullanılarak *Ki Kare* istatistikleri hesaplanmıştır.

α = 0,05 güven seviyesi için *Ki Kare* istatistikleri 4 serbestlik derecesi (v =5-1) için tablodan 9,49 olarak elde edilmiştir. Bu durum tüm modellerin RBS açısından arzu edilen seviyeyi sağladığını göstermektedir.

Ortalama hizmet süreleri açısından terfi sistemlerinde Üsteğmen rütbesi için 9, Yüzbaşı rütbesi için 15, Binbaşı rütbesi için 20, Yarbay rütbesi için 25 ve Albay rütbesi için 31 yıl OHS öngörülmüştür. Bu esaslar dikkate alınarak her model için gerçekleşen OHS'leri kullanılarak *Ki Kare* istatistikleri hesaplanmıştır.

α = 0,05 güven seviyesi için *Ki Kare* istatistikleri 4 serbestlik derecesi (v =5-1) için tablodan 9,49 olarak elde edilmiştir. Bu durum tüm modellerin OHS açısından arzu edilen seviyeyi sağladığını göstermektedir.

Terfi olasılıkları açısından modeller incelendiğinde; Binbaşı rütbesinden Yarbay rütbesine terfi olasılığı hariç tüm modellerin benzer sonuçlara sahip olduğu ve olasılıklar arası fazla fark olmadığı görülmektedir. Binbaşı rütbesinden Yarbay rütbesine terfide seçiciliğin artması açısından 0,561 terfi olasılığı ile Model-4'ün daha iyi bir sonuç ürettiği görülmektedir.

Modeller arasında sadece Model-1'in hem en erken RBS değerleri hem de yüksek erken terfi olasılığı değerine sahip olması nedeniyle uygulanabilir düzeyde olduğu belirlenmiştir.

Tüm etkinlik ölçütleri dikkate alındığında, Model-1, Model-2 ve Model-3'ün özellikle kontrol alanı oranlarını sağlamaları nedeniyle benzer özellikler gösterdiği, ancak erken terfi olasılığının yüksek olması ve her zaman en iyi personeli seçme imkanı tanıması nedeniyle Model-1'in diğer modellere göre uygun olacağı değerlendirilmektedir.

6.2. Değerlendirmeler

6.2.1. Model-1: Eşit olasılıklı terfi modeli

Eşit Olasılıklı Terfi Modeli; terfiye aday personelin, değerlendirme notları ve aynı zamanda o rütbedeki hizmet yıllarına bağlı olarak kazandıkları deneyim seviyeleriyle birlikte değerlendirilmesine fırsat vermektedir. Model, terfiye aday personelin tamamı arasından, terfi liyakat seviyesi en yüksek olanların seçilmesine imkan sağlamaktadır.

Bu avantajlara ilave olarak model personelin terfi etme şansı ve motivasyonunun terfi adaylığı süresince devam etmesini de sağlamaktadır.

Eşit Olasılıklı Terfi Modeli'nin sürekli en iyi personeli seçmesi özelliği rütbelerde farklı bekleme yıllarında yer alacak personel miktarlarında dalgalanmalara neden olabilmekte bu husus ise personel planlamalarını olumsuz etkilemektedir.

6.2.2. Model-2: Rütbede geçirilen yılları dikkate alan terfi modeli

Rütbede Geçirilen Yılları Dikkate Alan Terfi Modeli; personele terfiye aday olduğu ilk yılında dahi belirli bir oranda terfi şansı verilmesi, gençleştirmeye imkan verirken; ilk yılında terfi edemeyenlerin, müteakip yıllarda artan oranlarla terfi şansına sahip olmaları nedeniyle, rekabetin ve motivasyonun canlı kalmasına imkan sağlamaktadır.

Model, terfi dönemindeki tüm personelden, en başarılı olanların değil, gruplardaki en başarılı olanların terfisine imkan vermektedir. Bu nedenle, değerlendirme notu düşük dahi olsa, üst rütbeye terfi edenler olabilmektedir.

6.2.3. Model-3: İlk ve son yılları belli oranlarda garantili diğer yılları eşit olasılıklı terfi modeli

Model, terfi adaylığının ilk yılında bulunan gruba tanınan terfi oranının gençleştirmeyi hissedilir derecede artırması, terfi adaylığının son yıla kadar devam etmesi nedeni ile motivasyonun canlı kalmasının sağlanması ve ara dönemde bulunan personel için bir havuz içerisinde daha fazla personel ile beraber değerlendirileceğinden, rekabet ve kalitenin artması açılarından diğer modellere göre bir üstünlük sağlamaktadır.

Terfi dönemindeki tüm personelden, en başarılı olanların değil, gruplardaki en başarılı olanlar bu modelde terfi edebilmektedirler. Bu nedenle, bazı durumlarda değerlendirme notu düşük dahi olsa, üst rütbeye terfi edenler olabilir.

6.2.4. Model-4: Kademeli eşit olasılıklı terfi modeli

Anılan model, personele terfiye aday olduğu ilk yılında dahi belirli bir oranda terfi şansı vererek, gençleştirmeye imkan verirken; ilk yılında terfi edemeyenlerin, müteakip yıllarda belirli oranlarla terfi şansına sahip olmaları nedeniyle, rekabetin ve motivasyonun canlı kalmasına imkan sağlamaktadır.

Modelin mahzuru, terfi dönemindeki her bir gruba belirli bir terfi oranı verilmesi, o rütbedeki en başarılı olanların değil, gruplardaki en başarılı olanların terfisine imkan vermesidir. Bu nedenle, değerlendirme notu düşük dahi olsa, üst rütbeye terfi edenlerin olması mümkündür.

KAYNAKÇA

[1] Macmillan Contemporary Dictionary, *ABC Publishing Co.*, 801 (1998)

[2] Tortop, N., "Personel Yönetimi (Personnel Management)", *TODAİE Yayınları, DİE Matbaası*, Ankara, 121 (1992).

[3] Ivancevich, J.M., Donnelly, J.H., and Gibson, J.L., "Managing for Performance, Revised Edition", *Business Publications Inc.*, Plano, Texas, 227-228 (1983).

[4] LTC Kenen,T.A., "The Problem of Promotion In The United States Army: Some Considerations Involved", *The Ohio State University Press*, 13 (1947).

[5] Akın, H.U., "The Potential os System Dynamics Methods in Human Resources Modeling", *Department of National Defence Publication*, Ottawa, 2 (2000).

[6] Coyle, R. G., "Management Systems Dynamics", *John Wiley & Sons*, New York, 182 (1977).

[7] Forrester, J.W., "Industrial Dynamics", *Pegasus Communications*, Waltham, MA, (1961).

[8] Askeri Terimler Sözlüğü (1977)

NÜKLEER, BİYOLOJİK ve KİMYASAL (NBC) TAARRUZ SİMÜLASYON MODELİ

Ebru ARSLAN [(a)], Güher BOZDOĞAN [(a)]

[(a)] METEKSAN SAVUNMA A.Ş., Beytepe Köyü Yolu No: 3 Bilkent-ANKARA,

{ebrua, gbozdogan}@meteksan.com.tr

ÖZ

Gelişen teknoloji ile savaşlarda kullanılan; nükleer, biyolojik ve kimyasal (NBC) silahların kolay geliştirilebilir ve üretilebilir olması, bu konudaki tehditleri ve buna bağlı olarak taktiksel anlamda eğitimin önemini arttırmaktadır (Silah, savaş teknolojilerde kullanılmaz Teknolojiler silah yapımında kullanılır.) Bu amaçla geliştirilen Taktik Nükleer, Biyolojik, ve Kimyasal Taarruz NE Simülasyon Yazılımı (TNBCAS), muhtemel bir NBC taarruzu senaryosundaki, nükleer, biyolojik ve kimyasal kirlenmeyi ve yayılımı; atmosferik şartlara ve coğrafi özelliklere bağlı olarak modellemeyi amaçlamaktadır. TNBCAS yazılımı, Diffüzyon modeli, Nükleer/Kimyasal/Biyolojik Hasar Kıymetlendirme Modeli, Arazi Modeli yapılarını bünyesinde barındırmaktadır. TNBCAS yazılımı, bu modelleri kullanılarak, NBC taarruzuna maruz kalmış bölge üzerindeki hasar analizlerini gerçekleştirerek, grafiksel ve sayısal olarak gösterime olanak sağlayacaktır. TNBCAS yazılım mimarisi, IEEE1516 HLA-RTI dağıtık simülasyon mimarisi ile uyumlu olup, diğer harp oyunu simülasyonları ile entegre olabilmektedir.

Anahtar Kelimeler: Nükleer, Biyolojik ve Kimyasal (NBC) Taaaruz, Simülasyon, Modelleme, Kavramsal Modelleme,HLA-RTI, Dağıtık Simülasyon Modeli

NUCLEAR, BIOLOGICAL AND CHEMICAL (NBC) ATTACK SIMULATION MODEL

ABSTRACT

The easiness of the development and the production of the Nuclear, Biological and Chemical (NBC) weapons to be used in today's wars, has increased the threats from NBC and therefore has scaled up the importance of tactical education on this matter. Tactic Nuclear Biological Chemical Attack Simulation (TNBCAS) software, which was developed on these purposes, is expected to model the nuclear, biological and chemical contamination and propagation within a typical NBC attack scenario with respect to the atmospheric conditions and geographical characteristics. TNBCAS software accommodates diffusion model, Nuclear/Chemical/Biological detriment estimation model and terrain model in structure. With these models, TNBCAS software would provide visual and numerical analyses via implementing the detriment analyses for areas exposed to NBC attack. TNBCAS software architecture would be compliant to IEEE1516 HLA-RTI distributed simulation architecture standard and therefore would be able to integrate with other war game simulations.

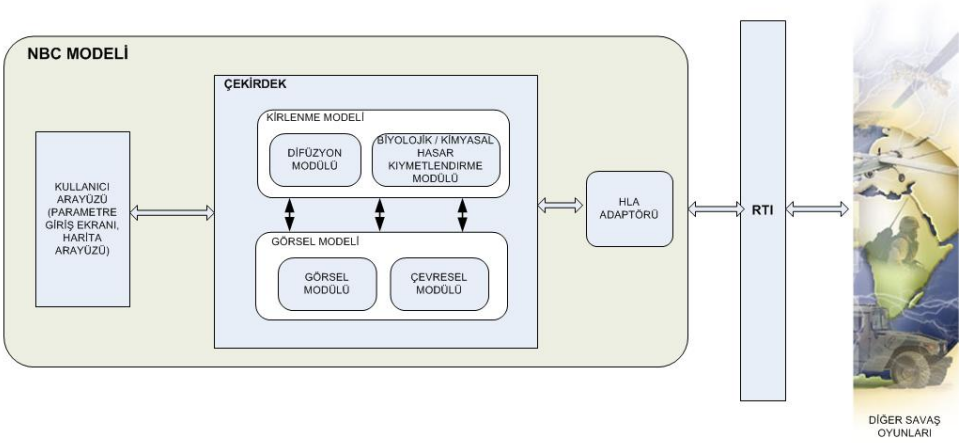
Keywords: Nuclear, Biological and Chemical (NBC) Attack, Simulation, Modelling, Conceptual Modelling, HLA-RTI Distributed Simulation Model.

1. GİRİŞ

NBC saldırısının modellenmesi, zararlı parçacık bulutunun yayılımı ile etkilerinin simüle edilmesidir. Parçacıklar, coğrafi özelliklere bağlı olarak çeşitli katmanlardan (hava, su, toprak) oluşan bir ortamda hareket ederler. Parçacıkların hareketinde tüm bu katmanların farklı etkileri ile beraber, parçacıkların çökmesi, rüzgar, sıcaklık ve nemin etkisi, arazinin topoğrafik özellikleri, arazi yüzeyinin etkisi (bitki örtüsü) modellenmektedir. Parçacıkların birim alandaki yoğunluğuna bağlı olarak taarruzun yaratmış olduğu hasar modellenmektedir. Nükleer etkilerin modellenirken, ısı ve blast etkisinden bahsedip, sadece ikincil radyasyon etkilerin dikkate alınmaktadır.

2. MODELLER

Bu yazılım, dağıtık simülasyon mimarisine sahip olarak, harp oyunları için gerekli NBC taarruzunu modellemektedir.Yazılımın mimarisi şekil 1'de tasarlandığı gibidir.



Şekil 1: Yazılım mimarisi

TNBCAS (Taktik Nükleer, Biyolojik, ve Kimyasal Taarruz Simülasyon Yazılımı) aşağıdaki matematiksel modelleri içermektedir.

2.1 Kirlenme Modeli

Kirlenme modeli; nükleer, biyolojik ve kimyasal başlığa sahip olabilecek silahların hedefi vurması durumunda atmosferik ve coğrafi şartları da göz önüne alarak ortaya çıkacak olan yayılımın zamansal modelidir. Bu simülasyonda silahlar önlenememiş ya

da kısmi tahrip edilmiş olabilecek ve farklı yoğunlukta nükleer, biyolojik ve kimyasal etkiye sahip olabilecektir. Ortaya çıkan yayılım ve taşınımlar ile etkiler görsel olarak kullanıcıya sunulmaktadır.
Hesaplama, silahın hedef isabeti anında kontrol merkezinde isabet alan bölge hakkında kayıtlı olan atmosferik, ortam koşulları, patlama yüksekliği, mühimmat tipi ve etki çapı esas alınarak yapılmaktadır

2.1.1 Nükleer Kimyasal/Biyolojik Hasar Kıymetlendirme Modeli

Kimyasal ve biyolojik silahlar, nükleer silahlarla karşılaştırıldığında, tasarım ve teknolojilerinin kolay, maliyetlerinin de düşük olması nedeniyle, çeşitli ülke ve terörist grupların kitle imha silahı olarak kullanmaktadır.
Biyolojik silahlar; canlılar üzerinde hastalığa yol açan mikroorganizmalar bulaştırmak suretiyle, karşı tarafın direnme gücünü kırmak amacıyla geliştirilmişlerdir. Bu silahlar hedef kitleye, top mermileri, uçak bombaları, füze ve roketlerce taşınan harp başlıkları aracılığı ile ulaştırılırlar.
Kimyasal silahların kullanımının daha oturmuş olması ve son derece ucuza mal edilebilmeleri nedeni ile biyolojik silahlara göre daha önemli bir tehdittir. Kimyasal silahların yayılımı, sıvı haldeki zehirli maddelerle hedef bölgesi üzerinde bir gaz bulutu ve aeresol oluşturulması ilkesine dayanır. Oluşan gaz bulutunun ömrü, seçilen zehirli maddenin türü ve hava şartlarına bağlı olarak 30 dakikadan birkaç haftaya kadar değişiklik göstermektedir.
Biyolojik ve Kimyasal (BK) Harekât Alanı Hasar Kıymetlendirmesi (Battle Damage Assessment-BDA)'nin modellenmesi ve simülasyonu kapsamında geliştirilecek olan BC model federesinin benzeri olan sistemler aşağıda sıralanmıştır:

1. HPAC (Hazard Prediction and Assessment Capability).
2. ANBACIS (Automated Nuclear, Biological, and Chemical Information System)
3. MATHEW/ADPIC :The Mass Consistent Wind Field (MATHEW) / Atmospheric Diffusion Particle in Cell (ADPIC) model.
4. NUSSE (The Non-Uniform Simple Surface Evaporation)
5. PEGEM (Post-Engagement Ground Effects Model)
6. VLSTRACK (Vapour Liquid Solid Tracking)

Düşman Biyolojik/Kimyasal (BK) başlıklı füze saldırısının önlenememesi veya kısmen önlenmesi halinde BK ajanlara maruz kalınması durumunda etkilerinin en aza indirilmesi ve pasif savunma tedbirlerinin uygulanabilmesi için, bu ajanların yayılışlarının önceden tahmin edilip tedbirler alınması hayati önem taşımaktadır. Kimyasal ve biyolojik ajanların atmosferik taşınımlarının yön ve etki alanlarını önceden tahmin etmenin en güvenilir yolu, meteorolojik verilerin ve arazi özelliklerinin birlikte kullanılması ile mümkün olmaktadır. Doğa, Meteorolojik veriler ve arazi özellikleri kullanılarak oluşturulan hasar tahmin ve değerlendirme programının temeli atmosferik taşınım ve yayılım modellemesine dayanır.

BK Harekât Alanı Hasar Kıymetlendirme modelleme ve simülasyon federesi aşağıdaki davranışları içermektedir:

1. Biyolojik ve Kimyasal kirlenme etkilerinin, atma vasıtalarının kısmi tahribi veya önlenememesi durumlarında atmosferik şartlara bağlı yayınım ve taşınım (konveksiyon) matematiksel hasar modelinin zaman bağımlı olarak geliştirilmektedir.
2. Biyolojik kirlenme etkilerinin, atma vasıtalarının kısmi tahribi veya önlenememesi durumlarında atmosferik şartlara bağlı kimyasal yayınım (difüzyon) ve taşınım (konveksiyon) matematiksel hasar modelinin zaman bağımlı olarak geliştirilmekte, iki faaliyet sonucunda atmosferdeki belirli bir noktadan, verili başlangıç koşullarıyla (boundary conditions) oluşan bir kimyasal ve biyolojik yayınımın, zaman içinde yeryüzündeki yoğunluk (konsantrasyon) ve etkinlik dağılımı hesaplanacak ve görselleştirilebilmektedir. Böylece, bir bölgenin zaman içinde hangi seviyelerde kimyasal ve biyolojik kirlenmeye maruz kalabileceği bilinerek, dost unsurların hasar kıymetlendirilmesi yapılabilmektedir.
3. Matematiksel modellerden yararlanılarak, Biyolojik/Kimyasal (BK) kirlenme ve Biyolojik/Kimyasal (BK) taarruzuna karşı pasif savunma tedbirleri, zaman bağımlı olarak, Biyolojik/Kimyasal (BK) federesi (Biological/Chemical federate) haline getirilebilmekte ve simülasyon sistemine entegre edilebilmektedir.
4. Biyolojik/Kimyasal (BK) başlıklı bir füzenin savunma sistemleri tarafından engellenememesi durumunda, simülasyon tarafından füzenin tipi, taşıdığı ajan tipi, çevre ve meteorolojik etkiler göz önüne alınarak ajanın yayılımı modellenebilmekte ve oluşturacağı etkiler hasar tespit birimi ile belirlenebilmektedir.

Simülasyon ortamında, görsel modeli tarafından üretilen meteorolojik etkiler (rüzgar, yağmur, kar, vb.) sayısal harita modelleri ile kullanılır partikül taşınım analizleri yapılır, atmosfere bırakılan tehlikeli mühimmatın sivil-askeri hedefler üzerinde oluşturacağı etkiler hesaplanarak etki altında kalması muhtemel alanlar yayılım bölgesi ile birlikte harita üzerinde gösterilebilmektedir. Doğru hesaplama yapabilmek için hasar değerlendirme aşamasında özellikle güncel ve anlık yerel atmosferik değerler kullanılmaktadır. Yerel ve güncel atmosferik veriler özellikle bölgesel (dar alanlardaki) saldırılar durumunda önem kazanmakta olup, atmosferik verilerin alınacağı bölge çözünürlüğü kullanıcı isterleri doğrultusunda belirlenmekte ve yüksek, orta ve düşük çözünürlük imkanları kullanıcıya sunulmaktadır.
Hasar alanının ve BK ajanların yoğun olarak taşındığı alanlar ile daha zayıf taşınabileceği alanlar değişik renk ve yoğunluklarda harita üzerinde gösterilmekte ve kullanıcı istediği anda fare yardımı ile harita üzerinde istediği noktadaki yoğunlukları gözlemleyebilmektedir. Aynı zamanda bu harita üzerinde ana kümenin yeri, etki alanı, ajan türü, etki süresi ve zamanı ve belirsizlik bilgileri aynı ekranda görülebilmektedir.

Görsel model şu katmanlardan oluşacaktır:

- Sayısal Harita Yükseklik Modeli
- Arazi Örtüsü

Model Girdileri

- Atmosferik Veriler
  - Rüzgar hızı, Rüzgar Yönü, Yağmur, Kar, Basınç, Nem, Sıcaklık, vb.
- Kimyasal/Biyoljik Silah Özellikleri
  - Taşıdığı B/K ajan miktarı, tipi,
- Sayısal Harita ve CBS
  - Uygun çözünürlükte DTED veya istenilen formattaki sayısal haritalar, Haritalar üzerinde konuşlu Şehir, Tesis, Yollar, vb. bilgileri.

Model Çıktıları

- Kitle üzerine etkisi
- Toksit seviyeleri
- Kirlenmiş alanlar
- Maruz kalan objeler
- Hasar alanları
- Muhtemel ölüm ve yaralanmalar

Model, meteorolojik etkilerin zaman bağımlı olarak atmosferik yayılımını ve yüzey üzerindeki durumunu olasılık tabanlı kestirebilmektedir. Rüzgar alanındaki ve atmosferik olaylar üzerinde oluşacak belirsiz yapılar (büyük boyutlu bilinmeyen değişimler veya dar bölgeli türbülans etkileri) belirli bir ortalama değer ve standart sapma ile rasgele olarak üretilmektedir.

2.1.2 Difüzyon Modeli

Taktik Nükleer, Biyolojik, ve Kimyasal Taarruz Simülasyon Yazılımı (TNBCAS) sisteminde difüzyon, konveksiyon , reaksiyon, çökme/tortu (deposition), yüzeylerden yansıma durumlarını barındırarak modellenmektedir. Atmosferde parçacık yayılımında, rüzgar konveksiyon, parçacıkların kimyasal özellikleri, türbülans etkileri ve çökme/tortu nedeniyle de reaksiyon modellenmesi gerekmektedir. Sistem aşağıda tanımlanmış olan girdileri almaktadır:

- Yüksek, orta ya da düşük çözünürlükteki atmosforik veriler: rüzgar hızı, rüzgar yönü, sıcaklık ve nem miktarı
- Arazi yükseltileri ve arazi yüzey özellikleri (ağaçlık alan, boş arazi, kuru arazi, neme doymuş arazi)
- Yayılımın tipi (çizgisel, noktasal, dikdörtgensel, dairesel) ve yeri
- Yayılımın ilk konsantrasyonu ve dozajı
- Parçacığın fiziksel ve kimyasal özellikleri (ağırlığı, bozulum miktarı)

TNBCAS sisteminde, difüzyon,

$$\frac{\partial N}{\partial t} + u_0 \frac{\partial N}{\partial x} = D \frac{\partial^2 N}{\partial x^2} + F(N)$$

dinamik denklemine uygun olarak modellenmektedir.

N : madde miktarı

$\frac{\partial N}{\partial t}$: birim zamandaki madde değişimi

u_0 : rüzgar hızı

$u_0 \frac{\partial N}{\partial x}$: rüzgar nedeniyle olan, birim zamandaki madde değişimi (konveksiyon)

D : difüzyon katsayısı (türbülans ve farklı katmanlar nedeniyle dinamik olarak değişebilecektir.)

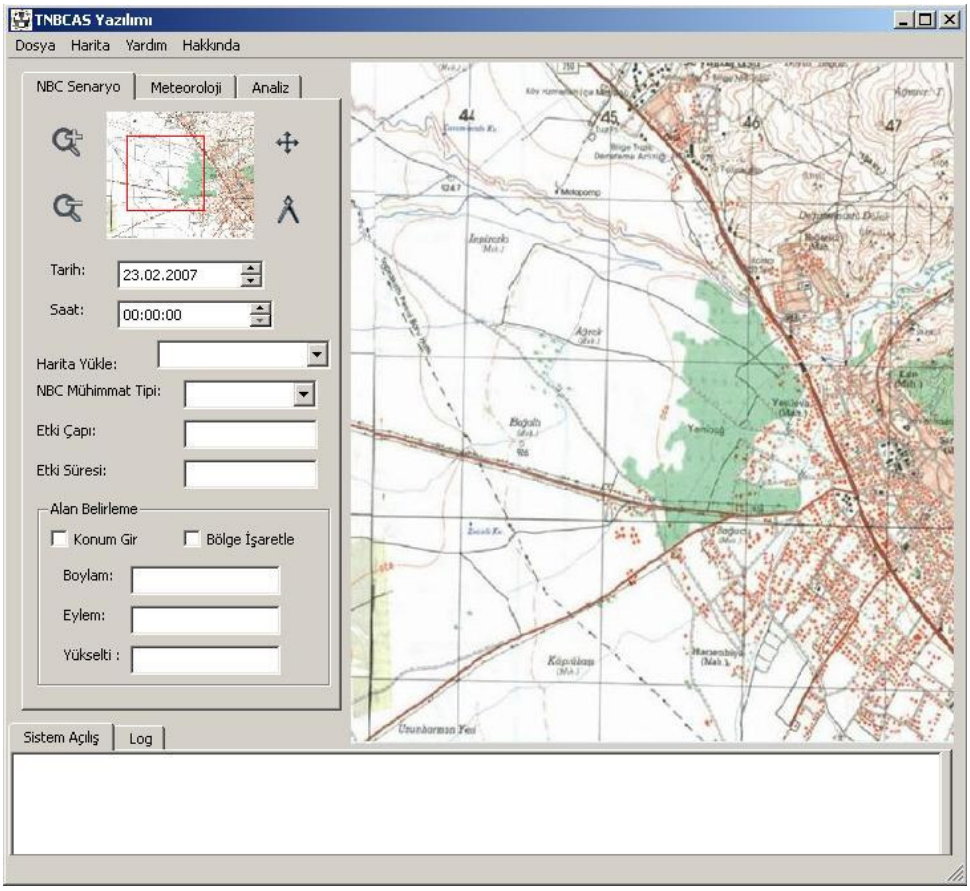
$D \frac{\partial^2 N}{\partial x^2}$: difüzyona bağlı birim zamandaki madde değişimi

$F(N)$: bozulum, çöküntü/tortu ya da hava dışındaki maddelerce (toprak, bitkiler) emilim olayları nedeniyle olan madde miktarı azalımı hızı

3. Yazılım Arayüzleri

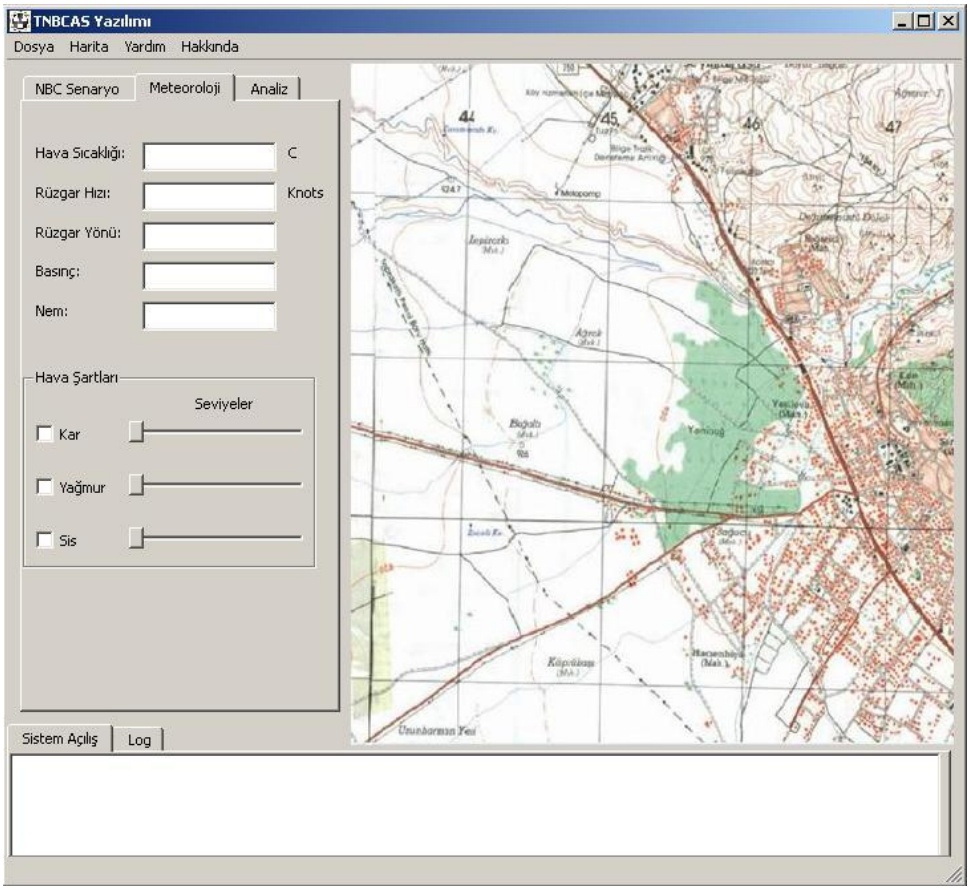
3.1 TNBCAS Yazılım Arayüzü

TNBCAS yazılım arayüzü NBC Senaryo ekranı, meteoroloji arayüzü ve analiz arayüzü olmak üzere üç ana bölümden oluşmaktadır. NBC senaryo ekranı, planlanan NBC Taarruzunun alanı ve türü hakkında ilgili verilerin girildiği kullanıcı arayüzüdür. (Şekil 2)



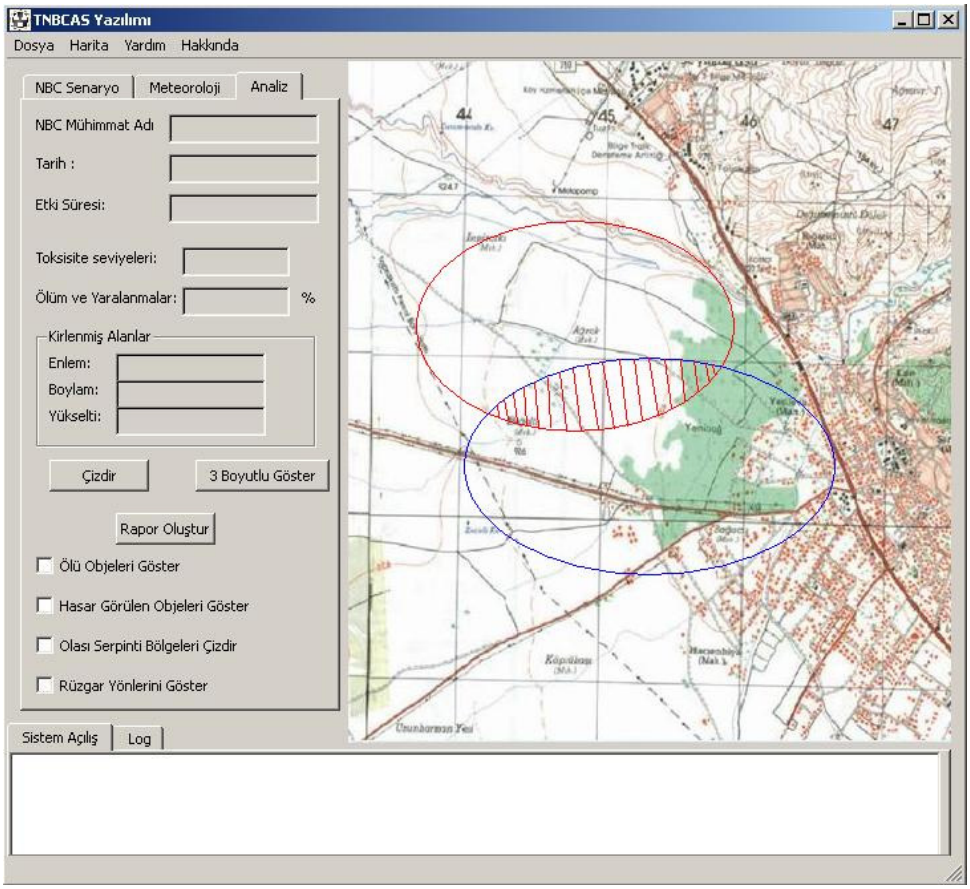
Şekil 2. NBC Taarruzu senaryo hazırlama ekranı

Meteoroloji arayüzü, NBC simülasyonun yapılacağı bölgenin sıcaklık , basınç, rüzgar hızı ve yönüne ilişkin verilerin girildiği kullanıcı arayüzüdür. (Şekil 3)



Şekil 3. Meteoroloji raporu hazırlama arayüzü

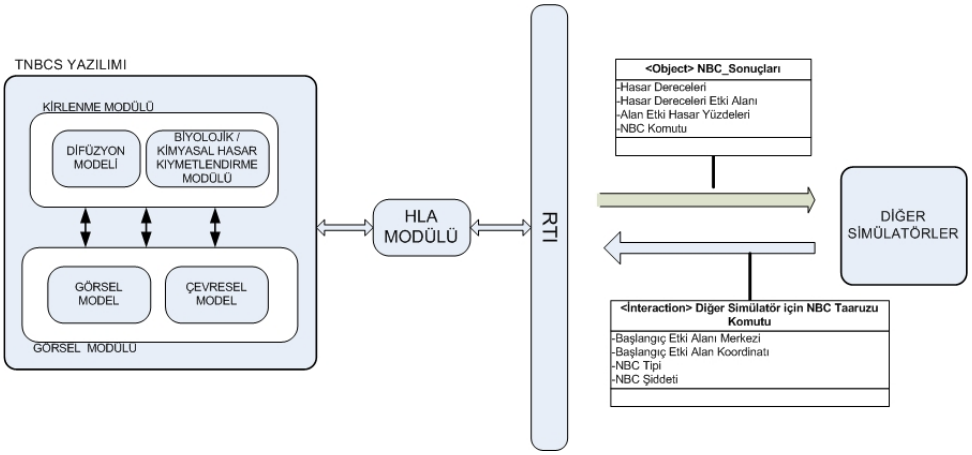
Analiz arayüzü kullanılarak NBC simülasyonun etkilerine yönelik değerlendirmeler ve bu bilgileri içeren rapor çıktıları üretilmektedir. (Şekil 4)



Şekil 4. NBC Taarruzu analiz arayüzü

3.2 HLA-RTI modeli

Sistem tek federasyondan oluşmaktadır. Federasyon, diğer simülatörlere, kendi alanı üzerinde NBC taarruzu gerçekleştirme imkanına sahiptir. NBC taarruzu kabiliyetine sahip simülatörlere ise, NBC taarruzu komutu gönderebilmektedir. Bu federe, RTI üzerinden, dışarıdaki bağlı simülasyonlara, Taarruzun hasar derecelerine göre sınıflanmış etki alanını ve bu sınıflardaki hasar yüzdelerini, Taarruzun ilk etki alanı ve merkez koordinatını, taarruzun tipini ve şiddetini gönderebilmektedir. Sistem bağlı simülatörlerden ise, taarruzun ilk etki alanı ve merkez koordinatını, tipini ve şiddeti, ve meteoroloji bilgilerini almaktadır . Modelin kavramsal modeli şekil 5'deki gibidir.



Şekil 5. HLA-RTI kavramsal modeli

4. SONUÇ

Taktik Nükleer, Biyolojik, ve Kimyasal Taarruz Simülasyon (TNBCAS) yazılımı kullanılarak, harp oyunlarında NBC Taarruzunun simüle edilmesine kolaylık sağlanması hedeflenmiştir. Dağıtık simülasyon mimarisine sahip olması, diğer simülasyon modelleri ile kolaylıkla iletişim kurulabilmesine olanak sağlayacaktır. Yazılım mimarisinin modüler yapıda olması, bir sonraki aşamada hedeflediğimiz; dost ve düşman modunda hesaplama ve tahmin yapabilen, karar destek imkanı sağlayan, yer, zaman, meteorolojik şartlar, hedef nitelik ve niceliği gibi hususları dikkate alan, ayrıntılı analiz yapabilecek modüller geliştirmektir. Böylece, eğitim tabanlı simülasyon yazılım kavramından sıyrılarak, karar destek yardımcısı gerçek bir "Muharebe Komuta Kontrol" yazılımına dönüştürülmesi hedeflenmektedir.

KAYNAKÇA

[1] Banks, B. Robert, (1994), "Growth and Diffusion Phenomena: mathematical frameworks and applications.", New York: Springer-Verlag, 368 p.

BENZETİM PROJELERİNDE BİLİMSEL VE TEKNİK YORDAM GELİŞTİRME PRATİĞİNE BİR YAKLAŞIM

Yıldırım BAHADIRLAR

TÜBİTAK MAM Bilişim Teknolojileri Enstitüsü,

P.K. 21, 41470 Gebze-Kocaeli, yildirim.bahadirlar@bte.mam.gov.tr

ÖZ

Bu çalışma özellikle benzetim projelerinde bilimsel ve teknik içeriği yoğun olabilen yordam geliştirme pratiğini ana hatları ile tanımlamakta ve bu altsürecin yazılım ayrıntılı tasarım geliştirme sürecindeki yerini belirtmektedir. İsterler Analizi, Mimari ve Öntasarım çalışmaları sonucunda istenen yüksek sadakatli çözümün ancak bilimsel ve teknik temele dayanan bir yaklaşımla çözülebileceği ve bu yaklaşımın doğrulama çalışmalarına da bir altyapı sağlayacağı belirtilmektedir. Önerilen pratik 7 ana adımda tanımlanmaktadır. Bunlar, 1.Bilgi Kaynaklarının, Bilgi Birikiminin Araştırılması ve Derlenmesi, 2.Öntasarımdaki Kullanım Durumları ve İşlevlere Yönelik Yordamların Belirlenmesi, 3.Yordam İçeriklerinin Modeller Yönünden Sınıflandırılması, 4.Matematiksel Tanıtlara Sahip Kuramlar ve Nümerik Çözümler ile Modelleme, 5.Yordamların Karma Modellerle Karşılanması, 6.Model Öğelerin Elde Edilmesi ve Mikro-Testler, 7.Yordamlara İlişkin Modellerin Tamamlanması ve Altbirim Testleridir. Ayrıca, bir radar benzetimi uygulamasında yeralan yüksek duyarlılıklı yordam geliştirme çalışmalarından, önerilen pratiğe örnek olabilecek ayrıntı düzeyinde sonuçlar verilmektedir. Önerilen yaklaşımın bileşen temelli yazılım geliştirme ve yeniden kullanılabilirlik çalışmalarına sağlayacağı katkıya da ayrıca değinilmektedir.

Anahtar Kelimeler: Analitik Benzetim, Bileşen Bazlı Yazılım Geliştirme, Mühendislik Düzeyi Benzetim, Yüksek Duyarlıklı Benzetim, Yordam Geliştirme.

AN APPROACH TO SCIENTIFIC AND TECHNICAL ALGORITHM DEVELOPING PRACTICE IN SIMULATION PROJECTS

ABSTRACT

In this study an algorithm developing practice, that may have intensive scientific and technologic contents, mostly in simulation projects is expressed by describing its major steps, and is positioned in detailed design phase of the software development process. It is stated that the high fidelity solution revealed at the end of user requirement analysis and preliminary design phase can be achieved using such an approach that principally bases on the scientific and a technical core, and this approach may also provide a vital income for the software validation studies. The practice is defined in 7 major steps. These are; 1.Investigation and Compilation of the Information Resources and Information Aggregation, 2.Definition of Conceptual Algorithms Implied by Use Cases and Functions in Preliminary Design, 3.Classification of Model Types in Algorithms,

4.Modeling Based on Theories with Mathematical Proofs and by Numerical Methods, 5.Description of Algorithms with Hybrid Models, 6.Implementations of Models and Micro-Tests, 7.Finishing Models of Algorithms and Subunit Tests. In addition, several illustrative results that may be skillful examples for the practice are given from a detail of high fidelity radar simulation study. A contribution of the practice to the component based software development and to the reusability studies is also stated.

Keywords: Algorithm Development, Analytical Simulation, Component Based Software Development, Engineering Level Simulation, High Fidelity Simulation.

1. GİRİŞ

Benzetim projeleri geliştirme süreci başlangıcındaki İsterler Analizi, Genel Sistem Mimari Tasarımı ve Sistem Öntasarım süreçleri geliştirilecek yazılım ürününün müşteri aracılığı ve katılımı ile genel bir tanımını ortaya koymaktadır. Sistem İsterlerinin ve müşteri senaryolarının, izlenebilirlikleri de sağlanarak öntasarıma aktarılması aşaması yazılım ürününü betimlemeye doğru atılmış bir ilk adım olarak tanımlanabilir. Sistem mühendisleri ve Geliştiriciler, bu aşamada müşteri isterleri ve müşterinin sahip olduğu ilgili kaynakları çözümlemiş durumdadır. Bunlar da kullanılarak yazılım ürününün genel işlevleri, kapsamı Sistem Mimarisi ve Öntasarıma yansıtılmış durumda olmalıdır.

Sistem Ayrıntılı Tasarımı aşamasında öntasarım sırasında tanımlanan kullanım durumları, aktivite diyagramları ve arayüzler kavramsal gösterimler olmanın ötesine taşınarak, genel deyişle, ayrıntılı akış, durum diyagramları ve sınıf diyagramları düzeyinde hazırlanmaktadır. Bu bildirinin konusu olan ayrıntılı yordamların hazırlanması da bu aşamada gerçekleştirilebilir. Sistem Ayrıntılı Tasarımı aşamasında hazırlanacak yordamlar müşteriden doğrudan elde edilmiş ve/veya Geliştirici tarafından tariflenmiş işlem adımları olabileceği gibi, Geliştiricinin bilimsel ve teknik temele dayalı olarak ve matematiksel (analitik) yolları izleyerek tasarlayacağı işlemler bütünü olabilir. Burada sözkonusu olan, ikinci yolla hazırlanacak yordamların geliştirilme pratiği, projenin konu ve kapsamına bağlı olarak Sistem Ayrıntılı Tasarım aşaması başlangıcında önemli bir altsüreç olarak yerini alabilmektedir. Önerilen bu altsürecin bir bilimsel kuram geliştirme yaklaşımı ile benzerlik içinde bir yordam geliştirme pratiği olduğu söylenebilir. Bildiri yazarının bilgisi dahilinde benzer bir pratik önerisi teknik yazında bulunmamaktadır. Bu bildiride belirtilen çıkış noktasındaki sorunu mühendislik düzeyi ya da analitik benzetim yaklaşımlarına öykünerek çözmeye yönelmek burada verilen akışa benzer bir akış ile sonuçlanabilir [1-2].

Sistem İsterleri aşamasında müşteri ile birlikte tanımlanan yazılım ürününü oluşturmak için gerekli olabilecek yordamlar hemen hemen tümüyle, bir matematiksel çıkarım altında toplanabilecek ve kapalı bir denklem ya da bir nümerik çözümle üstesinden gelinebilecek türden değildir. Sistem İsterlerinin çeşitliliği ve kapsamı özellikle benzetim sistemi yazılım projelerinde bunu açıkça göstermektedir. Müşteri, bir analitik benzetim yazılımından çok, insan etkileşiminin de içinde olduğu (human-in-the-loop) bir yazılım ürünü beklemesine rağmen Sistem İsterleri yüksek duyarlıklı (high fidelity) bir benzetimi gerektirebilmektedir [1-2]. Bu durum önemli oranda uygulama alanını niteliğinden ve gerektirdiği sadakat düzeyinden kaynaklanabilmektedir. İnsan etkileşimi doğal olarak benzetim çevriminin gerçek zamanlı olmasını da gerektirmektedir.

Matematiksel, buluşsal (heuristic) ve davranışsal (behavioral) modeller, bunların karmaları ve ardarda bağlanmaları ile Sistem İsterlerini yerine getirecek yordamları oluşturmak olanaklı görünmektedir. Bu anlamda, bir benzetim projesinde yordam geliştirme, bunları bir yazılım ürünü olarak gerçekleme ve test süreci bir bilimsel kuram geliştirme sürecine benzetilebilir. Müşteri isterleri ve öntasarım Geliştirici Grubunu bir "kuram geliştirme" işi ile karşı karşıya bırakmış görünebilir. Benzetim ürününün doğrulanması (validation) sürecinde de bu bakış açısının önemi ortaya çıkmaktadır.

Bölüm-4'de yüksek duyarlıklı bir benzetim uygulamasından önerilen yordam geliştirme pratiğine örnek olabilen ayrıntılar üzerinde durulmaktadır. Örnek uygulama geliştirildiği dönemde tüm adımları ile önerilen pratiği karşılamamaktadır, ancak pratik içinde yeralan karma ve yüksek duyarlıklı model uygulamalarına iyi bir örnek oluşturmaktadır.

2. SÜREÇ ADIMLARI

Bilimsel ve teknik temele ağırlıklı biçimde dayalı yordam geliştirme pratiği yazılım ayrıntılı tasarım süreci başlangıcında yeralabilen bir altsüreç olarak önerilebilir. Şekil 1'de bu ana adımlar ve adımlar arası ilişkiler öbek çizim biçiminde gösterilmiştir.

2.1. Bilgi Kaynaklarının, Birikiminin Araştırılması ve Derlenmesi

Bilimsel ve teknik temelli yordam geliştirme pratiği bulunduğu nokta nedeniyle proje konusunun içerdiği yordamların gerektirdiği belirli uzmanlık alanlarına sahip Geliştiricilerin çalışmada yer almasını önşart koşabilir; grubun bazı üyelerinin akademik personel olmasının tercih edilmesinde yarar vardır. Aynı zamanda Geliştiricilerin birbirleri ile sürekli ve yakın çalışarak bir grup oluşturmasından da yarar umulmalıdır.

Bu süreç adımında proje yordamlarının gerektirdiği konular Geliştiricilerin kendi uzmanlık alanlarına göre pay edilir. Geliştiriciler ilgili konularda hem teknik yazında (bilimsel ve teknik kitaplar, makaleler, bildiriler, teknik raporlar, v.b.) hem de müşterinin sahip olduğu altyapıda (el-kitapları, eğitim notları, sistem özellikleri dokümanları, gerçek sistemler üzeri incelemeler, v.b.) kaynak taraması ve bilgi derlemesi yaparlar. Sistemli olarak elde edilen kaynaklardan ve dokümanterden (film, fotoğraf, ekran çıktıları, v.b.) proje bünyesinde uygulama alanlarına özgü bir derleme hazırlanarak özel bir kütüphane oluşturulabilir.

Geliştirici Grubu ve lideri ortak çalışma toplantıları düzenleyerek kaynakların irdelenmesini ve özel konulara odaklanılmasını sağlar. Bu yolla bilgi birikiminin projeye yönelik, hali hazır düzeyi yeterli oranda ortaya çıkarılır. Geliştirici Gurubu bu adımda kaynaklarda yer alan bilgi düzeyine projede yeterli olacak oranda ulaşmalıdır. Özel kütüphanenin kapsamlılığı ve araştırıcıların bilgi düzeyinin yeterlilik oranları bu adıma ilişkin bir ölçüt olarak atanabilir, ölçümü kolay olmayabilir.

2.2. Öntasarımdaki Kullanım Durumları ve İşlevlere Yönelik Yordamların Belirlenmesi

Sistem İsterleri Analizi ve Öntasarım Sürecinde yer alan aynı Geliştirici Grubunun bu adımda görevine devam ettiği varsayılmalıdır. Öntasarım sürecinin çıktısı olan kullanım durumlarında yer alan ve ana işlevlere yönelik yordam gruplamaları ve gerektirdikleri ana çözümler bu adımda belirlenir. Geliştirici Grubu üyelerinin teknik toplantıları ve eşli çalışmaları (pair-working) ile çözümlere ilişkin ana yordamlar ve içerikleri

kategorilere ayrılarak bu adımda belirlenir. Burada matematiksel, buluşsal ve davranışsal model kategorileri belirlenmeye çalışılmalıdır. Matematiksel, buluşsal ve davranışsal modelleme gereksiniminin ve düzeyinin belirlenmesi bu basamağın konusudur. Öntasarım süreci ile betimlenen benzetim ürününün nihai kabulü ve doğrulanması aşamasında doğrulamaya esas olacak bilimsel ve teknik düzey bu adımda önemli oranda belirlenecektir. Ayrıntılı Tasarım Toplantısında müşteri katılımı ile yenilenmesi yine olanaklı tutulmalıdır. Toplantıda, majör değişiklik isteği ortaya çıkabilmeli ve değerlendirilebilmelidir. Kullanım durumlarına bağlı olarak izlenebilir yordam grupları ve model tiplerinin dökümü bu basamağın bir ölçütü olmalıdır.

2.3. Yordam İçeriklerinin Modeller Yönünden Sınıflanması

Ana yordamların belirlenmesi basamağından sonra bu adımda ana yordamların çözümlemesi yapılarak yordam yapıtaşları belirlenir. Yapıtaşları matematiksel, davranışsal, buluşsal özelliklere sahip olabilir ve bu aşamadan sonra tek başlarına birer işlev gören ve testlere sokulabilen modeller olarak algılanmalıdır. Bu basamağın çıktısı ana yordamlar altına toplanmış, işlevsel modeller kümesidir. Ana yordamlar altında farklı kategorilerden (matematiksel, davranışsal ve buluşsal) modeller yer alabilir. Bu basamağın bir ölçütü yordamların somut modellerinin belirlenebilmiş olmasıdır.

2.4. Matematiksel Tanıtlara Sahip Kuramlar ve Nümerik Çözümler ile Modelleme

Bir önceki adımda belirlenen ana yordamlar altındaki modeller bu adımda tek tek ele alınarak Geliştirici Grubu uzman personeli tarafından incelenir ve kuramsal boyutta ayrıntılarına girilir. Özellikle, matematiksel ve davranışsal modeller açısından Adım 2.1'de oluşturulan altyapı ve bilgi birikimi kullanılır; böylelikle model içerikleri ortaya konur. Bu adımda modellere ilişkin matematik, nümerik ve davranışsal yöntemler ortaya çıkarılmalı, alternatif yöntemler belirlenmelidir.

Bilimsel ve teknik yazına dayalı izlenebilirlik bu aşamada ortaya konulmalı, gerekli biçimde dokümantasyonu yapılabilmelidir. Teknik raporlar bu kapsamda tanımlanmış dokümanlar olabilir ve ileriki adımlarda da geliştirilmek üzere bu adımda hazırlanmaya başlanabilir. Buluşsal modeller ana özellikleri, davranışları ve akışları yönlerinden bu aşamada tasarlanmalı, işlem yüklerinin ve zamanlarının kestirimi yapılmalıdır.

Modellerin birer yapıtaşı olarak proje isterlerini karşılamadaki teknik özellikleri bu aşamada kesinleştirilebilmelidir. İşlem yükleri ve işlem zamanlarının kestirilmesi bu aşamada yapılmaya başlanmalıdır. Modeller yönünden tasarım alışverişleri ve alternatifleri bu aşamada değerlendirilmelidir. Hazır modellerin incelenmesi, değerlendirmesi ve projede kullanımının belirlenmesi bu adımda yapılmalıdır. Bu paragrafta belirtilen tüm çıktıların denetlenmesi basamağın bir ölçütü olmalıdır.

2.5. Yordamların Karma Modellerle Karşılanması

Giriş bölümünde işaret edildiği gibi, Müşteri İsterleri Analizi ve Öntasarım süreci sonucunda betimlenen yazılım ürününün altişlevlerini tek başlarına karşılayabilen bilimsel ve teknik modeller hemen hemen hiç ya da doğrudan hali hazırda bulunmamaktadır. Bu noktadan hareketle Adım 2.2'de belirlenen ana yordamların Adım 2.3 ve 2.4'de ayrıntılandırılan modeller ile karma biçimler oluşturularak karşılanması tasarlanmalıdır. Bu basamağın başlıca görevi ana yordamların işlevini yerine getirecek gerekli karma model yapısının tanımlanmasıdır. Adım 2.4'de belirlenen modellerin

kendi aralarında birlikte çalışabilirlik ve geçerli sonuçları üretmesi gibi saptamalar bu adımda sonuca ulaştırılmalıdır. Modeller arasında geçişleri sağlayan, doğrulaması yapılabilecek ve buluşsal olabilen iç arabirimlerin tanımlanması bu adımda yapılmalıdır.

Karma bir modelin ana yordamın işlevini yerine getirebilme özelliği bu basamağın çıktısı niteliğinde ortaya konulmalıdır. Aynı zamanda karma modelin üreteceği çıktıların geçerliliğinin bilimsel ve teknik açıdan nasıl gösterileceği bu aşamada belirlenmelidir. Referans çözümler ve/veya ölçümler ile karşılaştırma, istatistiksel analiz yolu ile gösterme gibi yöntemlerin uygulama planları bu adımda belirtilmelidir. Karma model yapılarının tanımlanabilmiş olması bu basamağın bir ölçütü olmalıdır.

2.6. Model Öğelerin Elde Edilmesi ve Mikro-Testler

Model öğeleri Adım 2.4'de belirlenen matematiksel, davranışsal ve buluşsal modeller ve alternatifleridir. Adım 2.4'de belirlenen yöntemler ile bir yardımcı yazılım geliştirme kapsamı içinde bu modellerin denemeleri ve sonuçlarının irdelenmesi bu basamağın ana işidir. Projede önerilen programlama dili dışında daha yüksek seviyeden bir programlama dili de kullanılarak bu basamağın konusu olan modellerin geliştirilmesi sağlanabilir. Burada yazılım geliştirme yöntemi olarak test-merkezli bir yöntem uygulanabilir. Model geliştirme aşamaları ve bu sırada hazırlanacak yöntemlerin testler ile aşamalı olarak sınanması planlanmalıdır.

Mikro-test kavramı Adım 2.4'de belirlenen modellerin geliştirilmesi tamamlandığında uygulanacak testleri tanımlamak için kullanılmaktadır. Bu testler aracılığı ile modellerin ana yordamlar içindeki işlevini yerine getirip getiremeyeceği sınanacaktır. Bu basamağın çıktısı olarak ana yordamların karma yapısında kullanılan modeller Ayrıntılı Tasarıma girdi olacak biçimde metotları ve ön denemeleri düzeyinde hazırlanmış olacaktır. Mikro-testlerin dokümantasyonu uygun biçimde gerçekleştirilmelidir. Mikro testler ve başarımları bu basamağın bir ölçütü olmalıdır.

2.7. Yordamlara İlişkin Modellerinin Tamamlanması ve Altbirim Testleri

Adım 2.5'de tanımlanan karma modeller ile ana yordamın sağlanması işlemi bu aşamada gerçeklenecek ve testleri yapılacaktır. Bu adımda Adım 2.6'da elde edilen modellerin matematiksel, buluşsal olabilecek iç arabirimler gerçekleştirilerek karma model yapısına ulaşılması gerçekleştirilecektir. Bu adımda ana yordamın sınanmasını hedefleyen altbirim testleri tanımlanacak ve gerçeklenecektir. Testlerin içeriğinde tasarlanan yordam ile üretilen verilerin analizi, referans çözümler ve/veya gerçek ölçüm değerleri ile karşılaştırma aşamaları da bulunmalıdır.

Bu adım sonucunda ana yordamlara ilişkin karma model yapısı, metotları, ayrıntılı yordam arayüzleri, hesap yükleri ile beraber ortaya çıkarılacaktır. Yazılım ürünü Ayrıntılı Tasarımına girdi olacak ana yordamlara ilişkin tüm bilgiler, taslaklar ve altbirim test sonuçları üretilmiş olacaktır. Benzetimin doğrulaması aşamasında referans olabilecek altbirim testlerinin dokümantasyonun yapılması önemli görülmelidir. Altbirim testlerinin ayrıntılı dökümü ve test başarımları bu basamağın ölçütü olmalıdır.

3. GÜNCELLEME VE AYRINTILI TASARIMA GEÇİŞ

Model öğelerin elde edilmesi ve ana yordamların tamamlanması (Adım 2.6 ve 2.7) aşamasında öntasarıma ilişkin ana yordamların farklı biçimde tasarlanması gerektiği

durumlar ortaya çıkabilir. Bu durumlarda Öntasarım Belgelerini güncelleyerek yeni tasarımın yansıtılması gereklidir. Söz konusu adımlarda ortaya çıkan yeni kısıtlar altında öntasarım güncelleme işi yapılmalıdır. Bazı Müşteri İsterlerin eski biçimi ile karşılanamayacağı durumları görüşmek üzere öntasarım güncelleme (revizyon) için müşteri görüşünün alınması sürece dahil edilmelidir. Adım 2.6 ve 2.7'de öntasarıma ilişkin ana yordamların farklı biçimde tasarlanması gerektiği durumlar ortaya çıktığında bunlar Genel Sistem Mimarisinde güncelleme gerektirecek ölçekte olabilir. Sistem mimarisinde yapılacak güncelleme getiri ve götürüleri ile müşteriye aktarılmalı ve müşteri görüşü sürece dahil edilmelidir. Bunların sonucunda Genel Sistem Mimari Belgelerini güncelleyerek yeni tasarım belgelere yansıtılmalıdır.

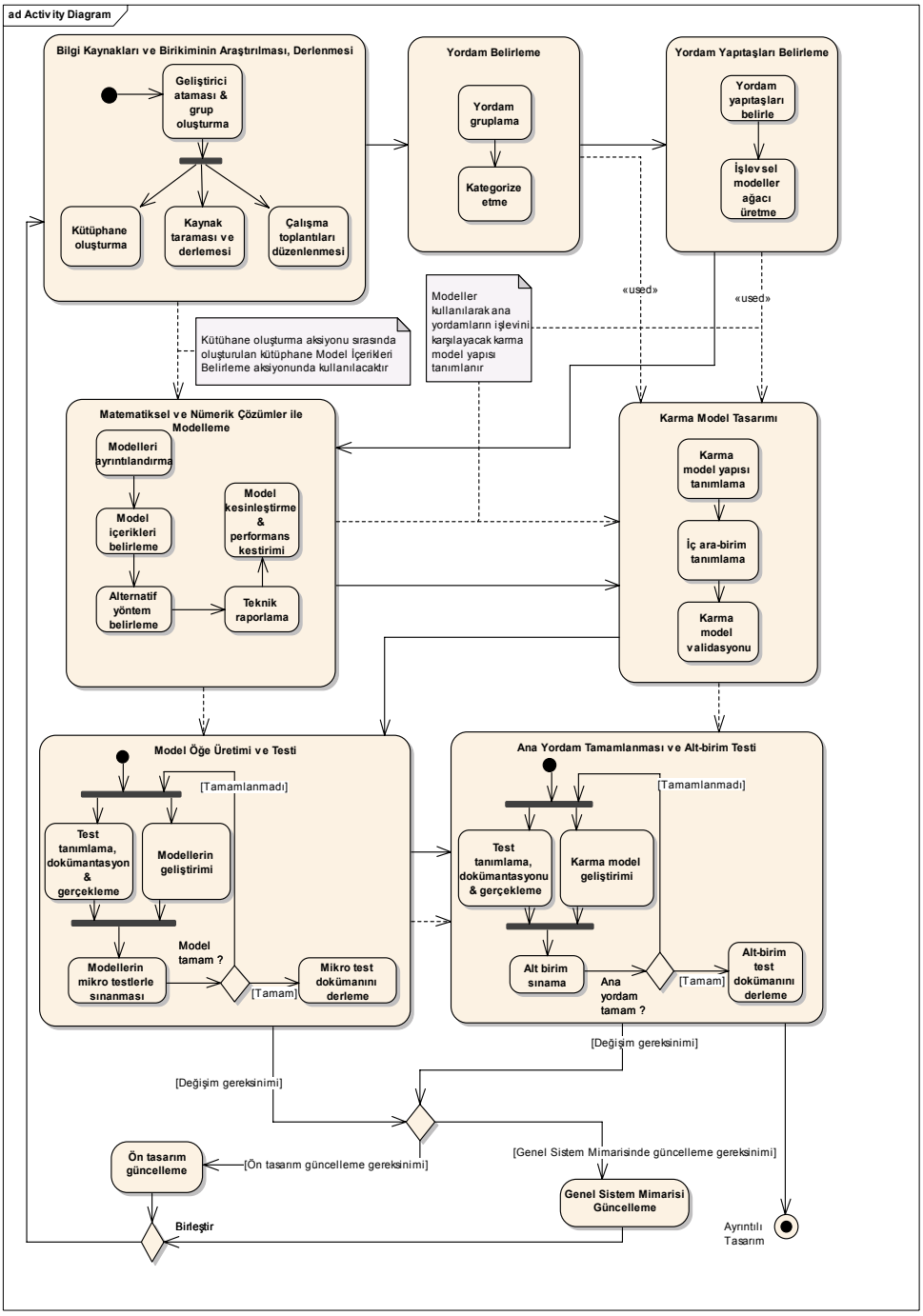
Adım 2.7 sonucunda yukarıda değinildiği gibi ana yordamlara ilişkin karma model yapısı, metotları, ayrıntılı yordam arayüzleri, hesap yükleri ile beraber ortaya çıkarılacaktır. Ana yordamlar açısında Yazılım Ürünü Ayrıntılı Tasarımına girdi olacak tüm arayüz bilgileri, parametreler, metotlar, taslak diyagramlar ve akışlar bu bölümde tanımlanan süreç sonunda elde edilmiş olmalıdır, Ayrıntılı Tasarıma kaynak teşkil edecek bilimsel ve teknik temelli kapsamlı çalışma gerçekleştirilmiş olacaktır.

4. ÖRNEK BİR BENZETİM UYGULAMASI

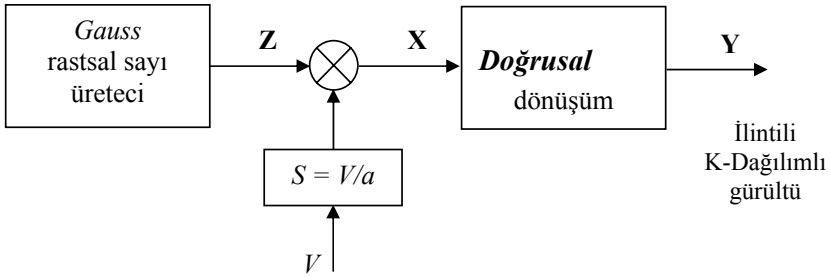
Bir kıyı gözetim radarının benzetimi çalışmaları kapsamında yüksek duyarlılıklı benzetim isteri konmuş ve bu kapsamda radar sinyalinin temel bandda sentezine dayalı bir yaklaşım gerçekleştirilmiştir. Radar benzetim ekranında kara yankıları, deniz yankısı ve hedeflerden gelen yankı gerçeğe yakın biçimde oluşturulmaktadır. Algılayıcı ve hedef takip yordamları sentezlenen sinyali girdi alarak ardarda çalıştırılmaktadır. Gerçekçi gürültü, bayılma (fading) ve kırpışma (jitter) etkileri de benzetimde yeralmakta ve sistem gerçek zamanlı çalışmaktadır. Bu açıdan bakıldığında radar benzetimi bir modeller silsilesi ya da karma bir model bütünüdür. Deniz kargaşası yüksek duyarlıklı biçimde, algılayıcı ve hedef izleme birimlerine gerçek etkiyi oluşturması üzere karma bir ana yordam olarak geliştirilmiştir. Bu çalışma yukarıda tanımlanan yordam geliştirme adımlarını genel anlamda hemen hemen tümüyle içermiştir, ancak çalışma burada önerilen pratiğin tüm adımlarını gerçekleştirildiği dönemde ayrıştırılmış olarak gösterecek biçimde bölümlendirilmemiştir. İzleyen bölümlerde kargaşa benzetiminin bazı sonuçları ve aynı yaklaşımla sistemde yer alan kırpışma etkisinin sonuçlarından örnekler verilmektedir [3].

4.1. K-Dağılımlı Deniz Kargaşası Yordamı

K-Dağılımlı deniz kargaşası yordamının temel öğesi Şekil 2'de verilen gürültü modelidir. Burada, Z *Gauss* Olasılık Yoğunluk İşlevine (OYİ) sahip vektördür ve kargaşanın kırpışma bileşenini temsil eder. *S*, gama dağılımına sahip modüle edici, yavaş değişen ve topaklanmayı sağlayan bileşendir. Doğrusal dönüşüm ile kargaşa sinyaline zamanda özilinti (autocorrelation) özelliği kazandırılmaktadır [4]. Bu model çıkışında elde edilen kargaşa sinyali deniz yüzeyinin radar kesit alanını frekansa bağlı olarak verebilen ampirik bir model çıktısı ile ölçeklendirilmektedir. K-Dağılımı biçim parametresi de benzer biçimde deneye dayalı bir model aracılığı ile sıyırma (grazing) açısının menzile göre değişimine bağlı belirlenmekte ve modele girdi oluşturulmaktadır. Böylelikle bir deniz kargaşası ana yordamı elde edilmiştir [3]. Adım 2.6'da değinilen mikro-testler kapsamında değerlendirilebilecek çalışmalarda kargaşa modelinin



Şekil 1. Bilimsel ve teknik temele dayalı yordam geliştirme adımları.



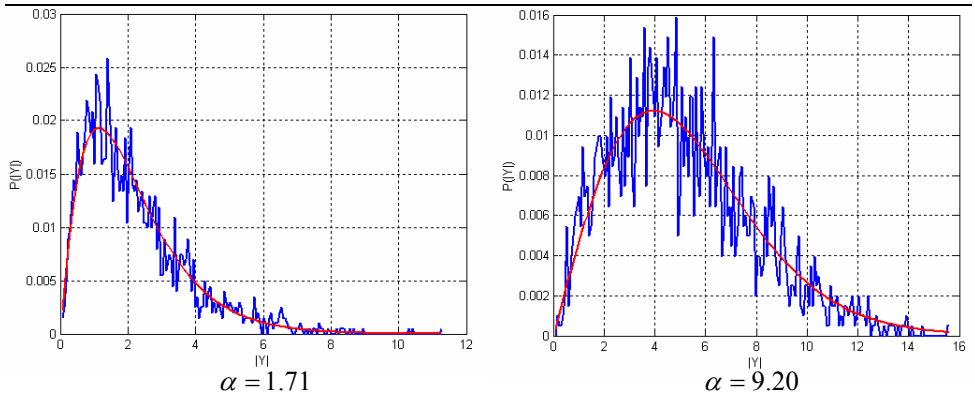
Şekil 2. K-dağılımlı kargaşa sinyali üretim modeli.

çıktısının istatistiksel analizi yapılmış ve çıkış sinyaline katılan özilinti özelliği hesaplanmıştır. Şekil 3'de bu test çalışmasından örnekler verilmiştir.

K-Dağılımlı kargaşa yordamının algılayıcı katındaki ve bir anlamda da radar ekranındaki görsel etkisini kestirmek amacıyla temel algılayıcı ünitesini (sabit eşiklemeyi) esas alan bir karşılaştırma yapılmıştır. K-Dağılımlı kargaşanın sıçramalı doğasını açığa çıkaran ve onu *Rayleigh* Dağılımdan ayıran durumu ortaya koymak için bir *Monte-Carlo* benzetimi yapılmıştır. Bu çalışmanın sonuçları Çizelge 1'de verilmektedir. Çizelgeden de görüldüğü gibi, *Rayleigh* Dağılımını esas alan bir algılayıcıda Yanlış Alarm Olasılığı $P_{FA}=10^{-5}$ olacak biçimde eşik değeri ayarlandığında K-Dağılımlı kargaşa ile karşılaşılması durumunda yanlış alarm oranı yaklaşık 33 kata kadar artabilmektedir [3].

Çizelge 1. Yanlış Alarm Olasılığı P_{FA}'nın K-dağılımı ve Özelliklerine Bağlı Değişimi.

OYİ		***Rayleigh* dağılımı**	**K-dağılımı**			
			$\alpha = 2$		$\alpha = 5$	
			İlintili	İlintisiz	İlintili	İlintisiz
P_{FA}	**Örnek Adedi**	$\hat{P}_{FA}$	$\hat{P}_{FA}$	$\hat{P}_{FA}$	$\hat{P}_{FA}$	$\hat{P}_{FA}$
10^{-3}	10^{5}	$1.13\text{x}10^{-3}$	$2.75\text{x}10^{-3}$	$2.80\text{x}10^{-3}$	$2.50\text{x}10^{-3}$	$2.20\text{x}10^{-3}$
10^{-5}	10^{6}	**$1.10\text{x}10^{-5}$**	**$36.3\text{x}10^{-5}$**	**$29.4\text{x}10^{-5}$**	$14.9\text{x}10^{-5}$	$15.5\text{x}10^{-5}$

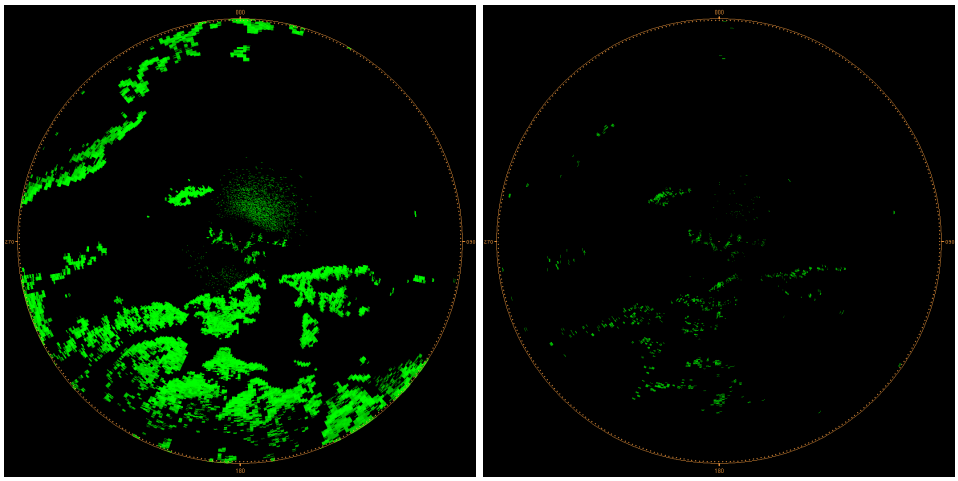


$\alpha = 1.71$ $\alpha = 9.20$

Şekil 3. Örnek OYİ'leri ve biçim parametresine (α) bağlı analitik K-dağılımı OYİ'leri.

4.2. Kırpışma ve Özilinti Niteliğinin Etkisi

Radar ekranında kara yankılarının her 360 derecelik taramada aynı kalmadığı gözlenen bir durumdur. Bu etkinin bir bölümü elektromanyetik dalganın çok yollu geri yansımasından, atmosferin oluşturduğu etkilerden kaynaklanırken önemli bir bölümü de radarın verici katlarındaki kırpışma (jitter) bileşeninden kaynaklanabilmektedir. Bu etkilerin tümü sentezlenen sinyalin faz bileşenine yavaş değişen bir gürültü bileşeni olarak eklenmiştir. Sonuçta radar ekranında gerçekçi bir bayılma/değişim etkisi elde edilmiştir. Ayrıca, sentezlenen sinyal MTI (Moving Target Indication) süzgecinden geçirildiğinde taramadan taramaya değişen ve gerçekçi biçimde kırpışma nedeniyle tümüyle bastırılamayan, ancak oldukça bastırılmış kara yankıları elde edilmektedir. Şekil 4-a'da karma yordamlar ile elde edilen bir "yüksek duyarlıklı" radar ekranı benzetimi, Şekil 4-b'de aynı ekranın MTI süzgecinden geçirilmiş biçimi görülmektedir. Şekil 4-b'de tamamen bastırılamayan kara yankılarının etkileri görülmektedir.



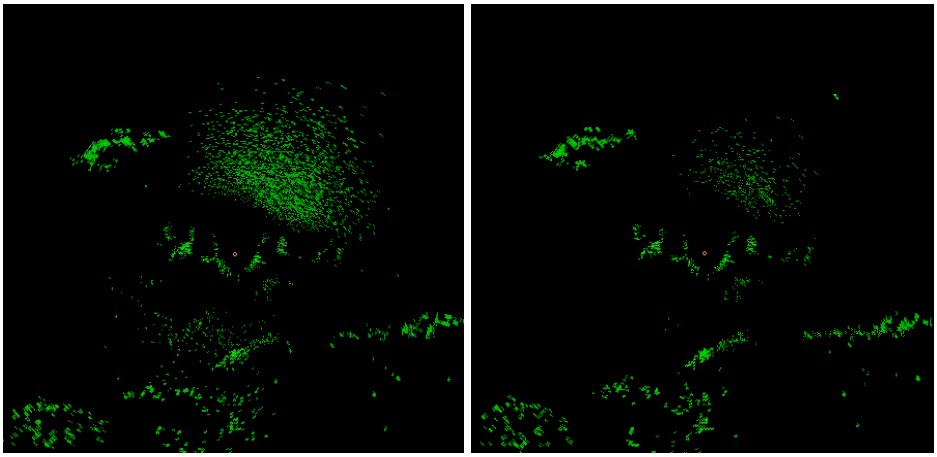
(a) (b)

Şekil 4. "Yüksek duyarlıklı" radar ekranı benzetimi ve MTI süzgeci çıkışı.

K-Dağılımının özilinti niteliğinin karma model yapısında bulunması ve özilintinin yüksek ve düşük olarak nitelenebilecek seçeneklere sahip olması durumunda MTI süzgeci çıkışında bu etkinin ortaya çıktığı görülebilmektedir. Özilinti niteliği yüksek olan durumda beklendiği gibi kargaşa sinyali daha fazla bastırılabilmektedir. Şekil 5-a'da "düşük" özilintili kargaşa sinyalinin bastırılma durumu, Şekil5-b'de işe "yüksek" özilintili kargaşa sinyalinin bastırılma durumu görülmektedir.

5. TARTIŞMA VE SONUÇ

İsterler Analizi, Genel Sistem Mimari Tasarımı ve Sistem Öntasarım altsüreçleri sonucunda genellikle kavramsal düzeyde sistem çözümü ortaya konmuş olmaktadır. Bu aşamada, özellikle yüksek duyarlıklı ya da mühendislik seviyesi benzetimlerde sistemin detay tasarımına geçişi sağlayan ana yordamların elde olması gereklidir. Bu durum hesaplama yüklerinin, detaylı arayüzlerin ve çalışma sonucunda ulaşılacak sadakat derecesinin belirlenmesi için de önem taşımaktadır. Benzetim uygulamaları geliştirme



(a) (b)

Şekil 5. "Düşük" ve "yüksek" özilintili kargaşanın MTI süzgeci ile bastırılması.

sürecinin bu aşaması bir sistematik altında toplanarak bilimsel ve teknik temelli bir yordam geliştirme pratiği niteliğinde önerilmiştir. Doğrulama çalışmalarına katkı açısından da önerilen pratiğin ayrı bir öneminin olduğu vurgulanmalıdır. Bu pratik altında geliştirilen model öğeler, karma modeller ve ana yordamlar kurumsal olarak da bir birikim oluşturacak, yeniden kullanılabilirlik ve bileşen temelli yazılım geliştirme çalışmalarına önemli girdi sağlayacaktır [5].

6. TEŞEKKÜR

Bildiri yazarı çalışmaya katkılarında dolayı Onur Destanoğlu, Baki Batı, Selçuk Giray Özdamar'a ve örnek benzetim uygulamasındaki katkılarından dolayı Doç. Dr. A. Arif Ergin, Bülent Özge, İsa Taşdelen ve Faruk Sarı'ya teşekkürü bir borç bilir.

KAYNAKÇA

[1] Fujimoto, R.M., (2000), Parallel and Distributed Simulation Systems, John Wiley & Sons Inc., NY, Weinheim, Singapore, pp.3-25.

[2] Etter, P.C., (2003), Underwater Acoustic Modeling and Simulation, Spon Press, London & NY, pp.1-18.

[3] Bahadırlar, Y., (2003), Mikrodalga Radarda Kargaşa ve Gürültü Modelleme Teknik Raporu: Sürüm-1, Sürüm-2, TÜBİTAK MAM BTE arşivi, Sürüm-1: 41 sayfa, Sürüm-2: 52 sayfa.

[4] M. Rangaswamy, D.D. Weiner, A. Öztürk, (1995) "Computer Generation of Correlated Non-Gaussian Radar Clutter," IEEE Trans on Aerospace and Electronic Systems, Vol. 31, No. 1, pp 106-116.

[5] Sommerville, I., (2004), Software Engineering, 7th edition, Addison Wesley, Chapter-4.

A6 OTURUMU:

KAVRAMSAL MODELLEME VE SİMÜLASYON ALTYAPILARI

SİMÜLASYON GELİŞTİRME SÜRECİ İÇİN ÇOK KULLANICILI KAVRAMSAL MODELLEME ARACI
Mert ŞİMŞİR, Burcu AKMAN, Özgür AYDOĞDU, Aydın OKUTANOĞLU, Mithat KALE

TOPÇU BATARYASI İÇİN KAVRAMSAL MODELLEME VE UYGULAMALARI
Gürkan ÖZHAN, Halit OĞUZTÜZÜN

DAĞITIK SİMULASYON PROJELERİNİN KAVRAMSAL MODELİNİ GELİŞTİRME ÇALIŞMALARINDA BOM KULLANIMI
Yasemin TİMAR, İsmail BIKMAZ, İsa TAŞDELEN, Sevgi AKGÜN, Oğuz DİKENELLİ

CİDDİ OYUN VE SİMÜLASYON ALTYAPISI
İsmail BIKMAZ, Gökçe YILDIRIM, Veysi İŞLER, Kürşat ÇAĞILTAY, Kayhan İMRE

SİMÜLASYON GELİŞTİRME SÜRECİ İÇİN ÇOK KULLANICILI KAVRAMSAL MODELLEME ARACI

Mert ŞİMŞİR [(a)], Burcu AKMAN [(b)], Özgür AYDOĞDU [(a)], Aydın OKUTANOĞLU [(a)], Mithat KALE [(a)]

[(a)] Meteksan Sistem A.Ş., ODTÜ Teknokent MODSİMMER A binası Ankara,

{msimsir, oaydogdu, aokutanoglu, mkale} @meteksan.com.tr

[(b)] ODTÜ Enformatik Enstitüsü., Merkez Mühendislik Binası 4.Kat Ankara,
akkan@ii.metu.edu.tr

ÖZ

Kavramsal model simülasyon hedeflerini sağlamak için simüle edilmesi gereken varlıkları, temel etkinlikleri ve etkileşimleri tanımlamaya yarar. Kavramsal modeller, simülasyon uygulamaları geliştirme yaşam döngüsü içinde simülasyon gereksinimlerinin belirlenmesi ile simülasyon uygulamasının tasarımı arasında yer alır. Kavramsal Modelleme Aracı (KAMA), simülasyon sistemleri geliştirme yaşam döngüsünde kavramsal modelleme faaliyetine katkıda bulunmak üzere geliştirilmektedir. Bu çalışmada, KAMA'nın temel işlevleri ve bu işlevlerin kavramsal modelleme sürecine olan katkıları anlatılmaktadır.

KAMA'da kavramsal modellerin görsel yöntemlerle tanımlanması sağlanmıştır. Model bileşenlerinin tanımı için bilgi giriş işlevleri bulunmaktadır. Modellerin ve ilişkilerin geçerlenmesi için bir ortak söz dizimi ve anlam bilim kural tabanı (OSAT) oluşturulmuştur. KAMA'da geliştirilen model ve bileşenlerinin saklanması ve tekrar kullanıma hazır tutulması için kullanılacak bir ortak veri ambarı (OVA) düşünülmüştür. Bu ortak veri ambarında yer alacak model ve bileşenler için de bir onaylama mekanizması geliştirilmiştir.

KAMA, kavramsal modellerin kolay ve düzenli bir şekilde tanımlanmasını sağlamak üzere geliştirilmektedir. Bunun sonucunda da taktisyen ile simülasyon geliştiricisi arasında bir köprü vazifesi görecektir. KAMA, askeri simülasyon uygulamalarının gerçekleştirilebilmesi için gereken modelleme ihtiyacını karşılamak üzere bir çalışma ortamı sunacaktır.

Anahtar Kelimeler: Doğrulama, görselleştirme, kavramsal model, kavramsal modelleme aracı, modelleme, simülasyon.

MULTIUSER CONCEPTUAL MODELLING TOOL FOR SIMULATION DEVELOPMENT PROCESS

ABSTRACT

Conceptual model is used for defining entities, basic activities and interactions that are required to be simulated in order to achieve simulation goals. In the life cycle of development of simulation applications, conceptual models lie between specifying simulation requirements and designing simulation applications. Conceptual Modeling Tool (KAMA) contributes to the conceptual modeling activity in the life cycle of simulation application development. In this work, main functionalities of KAMA and contributions of these functionalities to conceptual modeling process are described.

In KAMA, the conceptual models are provided to be defined using visual methods. In order to define model components, there exists information input activities. A common syntax and semantic rule base (OSAT) is formed for validation of generated models and relations. There is a common data warehouse (OVA) which is used to keep models and their components and to enable reusability of these models. An approving mechanism has been developed for the models and components that will be stored in this common data warehouse.

KAMA is being developed to define conceptual models in an easy and systematic way. Therefore, it will construct a bridge between the tactician and simulation developer. KAMA will provide a working space to meet modeling requirements of military simulation applications.

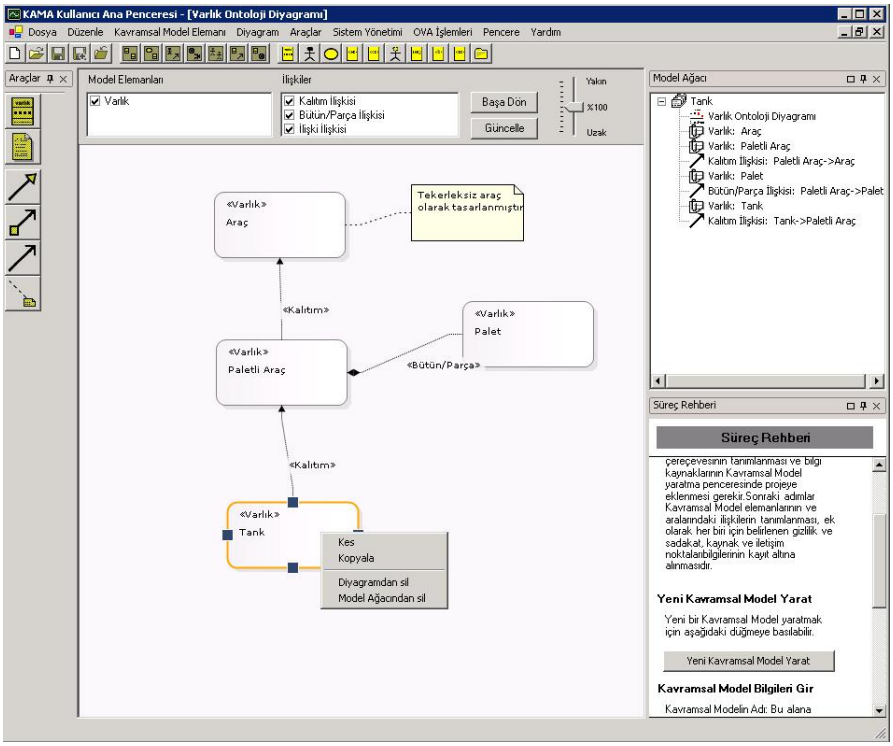
Keywords: Conceptual model, conceptual modeling tool, modeling, simulation, validation, visualization.

1. GİRİŞ

Kavramsal model simülasyon hedeflerini sağlamak için simule edilmesi gereken varlıkları, temel etkinlikleri ve etkileşimleri tanımlamaya yarar [1]. Kavramsal model kullanımı; hem gereksinim belirleme çalışmaları sırasında gözden kaçan eksik ve tekrarların elenerek gereksinimler listesinin geçerlenmesine, hem de geliştiricilerin ortak bir yaklaşım ve dil kullanmalarına yardımcı olur.

KAMA ortamında dört temel bileşen bulunmaktadır. Bunlar; model, diyagram, ilişki ve model elemanıdır. Model, kavramsal modeli temsil etmektedir. Diyagramlar, bir kavramsal modelin değişik açılardan ifade edilebilmesine olanak verir. Varlık Ontoloji Diyagramı, Görev Uzayı Diyagramı ve İş Akışı Diyagramı diyagram tiplerine örnek olarak verilebilir. İlişkiler, model elemanları arasında çizilen ve model elemanları arasındaki ilişkiyi ifade eder. İlişki tiplerine örnek olarak Kalıtım İlişkisi, Bütün-Parça İlişkisi, Kontrol Akışı İlişkisi verilebilir. Model elemanı, kavramsal modeli oluşturan kavramsal model elemanı bileşeni temsil eder. Varlık, Görev, İş model elemanlarına örnek olarak verilebilir. Farklı model elemanı tipleri bünyesinde farklı özellikleri barındırır. Özellikler, Kalıtım İlişkisi yolu ile birbirine aktarılabilir.

KAMA kavramsal modelleme yapacak olan kullanıcının çalışmasını kolaylaştıracak şekilde tasarlanmıştır. Diyagramların oluşturması için bir grafiksel tasarım editörü bulunmaktadır. KAMA'da modelin oluşturulmasına yardımcı olmak amacıyla kullanıcıyı yönlendiren ve Süreç Rehberi adı verilen bir özellik de yer almaktadır. Şekil 1'de örnek olarak oluşturulmuş bir varlık ontojoji diyagramı gösterilmektedir.



Şekil 1. KAMA kullanıcı arayüzü

KAMA'da sistem, model oluşturma sırasında ve sonrasında Ortak Sözdizimi ve Anlambilimi Kural Tabanı (OSAT) kuralları doğrultusunda doğrulama yapmaktadır. Örnek olarak, belli tipteki model elemanları arasında çizilebilecek ilişki tipleri OSAT ile belirlenmiştir. Bu konu Ortak Sözdizimi ve Anlambilimi Kural Tabanı (OSAT) bölümünde daha ayrıntılı olarak anlatılmaktadır.

KAMA'da oluşturulan model ve model bileşenlerinin tekrar kullanımını sağlamak amacıyla Ortak Veri Ambarı (OVA) yapısı oluşturulmuştur. Bu konuya Ortak Veri Ambarı (OVA) bölümünde ayrıntılı bir biçimde değinilmektedir.

KAMA'da meydana getirilen kavramsal modelin yapısına farklı görüş açılarından bakabilmek için hiperbolik ve konik ağaç gösterim tekniklerini baz alan görselleştirme yöntemleri kullanılmıştır. Bu konu Görselleştirme bölümünde daha detaylı olarak anlatılmaktadır.

KAMA bünyesinde ayrıca XML tabanlı veri gönderme/alma, raporlama, veri yedekleme, kullanıcı ve grup yönetimi, grup tabanlı yetkilendirme ve modellere ait gizlilik seviyeleri belirleme gibi yardımcı unsurlar da yer almaktadır.

2. ORTAK SÖZDİZİMİ VE ANLAMBİLİMİ KURAL TABANI (OSAT)

KAMA bir görev uzayının kavramsal modelini değişik açılardan ve değişik seviyelerde modelleyebilmek amacıyla tasarlanmıştır. Bu sebeple, KAMA ile modellenen bir kavramsal model, içerisinde farklı bakış açılarını yansıtan değişik tipte diyagramlar, model elemanları ve ilişkiler barındırır.

Model elemanlarının hangi diyagramlarda bulunması gerektiğini ve hangi model elemanlarının arasında hangi ilişkilerin bulunabileceği gibi kuralların tanımlanması ve bu kurallar doğrultusunda dinamik geçerlenme sağlanması amacı ile KAMA projesinde bir Ortak Sözdizimi ve Anlambilimi Kural Tabanı (OSAT) oluşturulmuştur.

OSAT'ta temel olarak aşağıda sıralanan kurallar yer almaktadır:

1. Model elemanı ile ilgili olan kurallar:
    - *İlişki tipleri ve ilişkili model elemanları:* Bir model elemanının hangi tipteki model elemanlarına, hangi tipteki ilişkiler ile bağlanabileğini gösteren kurallar kümesidir.
    - *Alt elemanları:* Bir model elemanının hangi tipteki model elemanlarının alt elemanı olabileceğini gösteren kurallar kümesidir.
    - *Türeyebileceği model elemanları:* Bir model elemanının hangi tipteki model elemanlardan türeyebileceğini gösteren kurallar kümesidir.
2. Diyagramlar ile ilgili olan kurallar:
    - *Model elemanları:* Bir diyagramda hangi model elemanlarının bulunabileceğini gösteren kurallardır.
    - *İlişkiler:* Bir diyagramda hangi ilişkilerin bulunabileceğini gösteren kurallar kümesidir.
    - *Gösterim:* Diyagrama özel kuralların tanımlanabildiği kural tabanıdır. Burada bir diyagramda mutlak bulunması gereken elemanlar, model elemanlarının ve ilişkilerin dizilimi ile ilgili kurallar oluşturulabilir.
3. Tüm elemanlar ile ilgili olan kurallar:
    - *Zorunlu alanlar:* Bir eleman tanımlanırken zorunlu olarak girilmesi gereken alanları tanımlayan kurallardır.

OSAT, KAMA ile görsel modelleme yapılırken dinamik olarak kullanılabildiği gibi, modellerin ve bileşenlerinin doğrulanması için de kullanılır. OVA'da yer alan elemanlar diğer modellerde de kullanılacağından, hepsinin OSAT'a uygunluğunun kontrol edilmesi gerekmektedir. Doğrulama işlemi bu amaçla kullanılmaktadır.

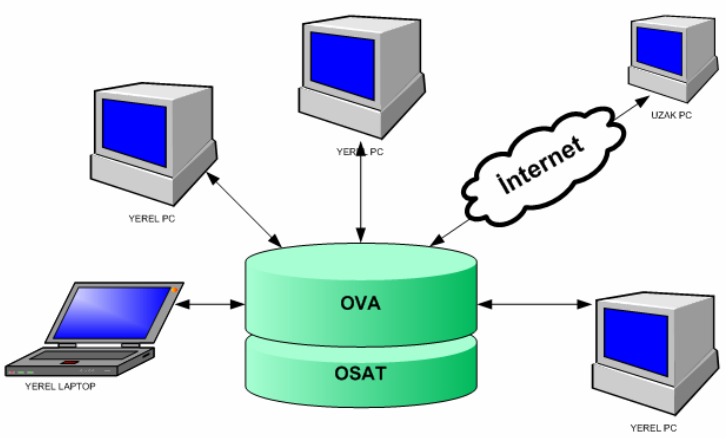
3. ORTAK VERİ AMBARI (OVA)

Kavramsal Modelleme Aracı kullanılarak tanımlanan model ve model bileşenleri tekrar kullanılmak üzere paylaşılan bir veri ambarında saklanmaktadır. Buna Ortak Veri Ambarı (OVA) adını vermekteyiz.

OVA'nın temel amacı yeni bir model oluşturulurken daha önceden yaratılmış olan model ve model bileşenlerinden azami olarak faydalanmaktır. Tabii ki, tekrar kullanımın gerçekleşmesi için öncelikle OVA'nın kısmen doldurulması gerekmektedir. Bir kavramsal model üzerinde çalışan KAMA kullanıcısı ihtiyaçları doğrultusunda OVA'ya bakarak gerekli bileşenleri bulabilecek ve kullanabilecektir. Buradaki bileşenlerin yetersiz olması durumunda güncelleme akışı kullanılarak OVA'nın güncellenmesi sağlanacaktır.

OVA'nın verimli bir şekilde kullanılabilmesi için güvenilir ve güncel olması önemlidir. Bunun için onaylama ve versiyonlama mekanizmaları geliştirilmiştir. Onaylama mekanizması, KAMA kullanılarak oluşturulan model ve model bileşenlerinin yayınlanması ve yetkili bir kullanıcının bunları kontrol ederek onaylaması prensibine dayanmaktadır. Böylece OVA üzerinde bulunan model ve model bileşenlerinin geçerli ve güvenilir olması sağlanacaktır. Versiyonlama mekanizması ise, model ve model bileşenlerinin versiyon bilgisinin bulunması ve yeni versiyonlarının üretilebilmesi prensibine dayanmaktadır. Versiyonlama, OVA'da bulunan model ve model bileşenlerinin olgunlaşmasına ve güncel kalmasına yardım eder. Versiyon numarasının artması ilerlemenin göstergesi olarak görülür. Örneğin, bir model elemanın ihtiyaca göre ekleme ve güncellemeler yapılarak yeni bir versiyonu oluşturulur ve bunun, model elemanının eski versiyonuna göre daha olgun ve gelişmiş olduğu varsayılır.

OVA merkezi bir sunucuda yer almaktadır (Şekil 2). KAMA Kullanıcıları ağ üzerinden OVA'ya bağlanarak OVA ile ilgili işlemleri gerçekleştirirler. OVA işlemleri web servis tabanlı olarak tasarlanmıştır. Bunun için ağ üzerinde bir web sunucu da yer almaktadır. Web sunucu, OVA'nın yer aldığı veritabanı sunucusuyla aynı makina olabileceği gibi farklı bir makina da olabilir. Lokal olarak çalışan model geliştirme arayüzünün OVA ile iletişimi XML tabanlı Web Servis altyapısı üzerinden gerçekleştirilmiştir. Bu web servisleri kullanıcı adı ve şifre ile korunmakta ve yetkisiz kişilerin sistemi kullanmaları engellenmektedir. Aynı zamanda WebServislerinin XML tabanlı açık bir mimari olması nedeniyle uygun yetkilere sahip farklı uygulamaların OVA bilgilerine kendi arayüzü kullanılarak erişebilmesi imkanı bulunmaktadır.



Şekil 2. OVA'nın ağ üzerinden paylaşımı

KAMA ayrıca OVA işlemlerini kullanmaksızın, ağ bağlantısız olarak yerel veritabanı üzerinden çalışabilmektedir. Kullanıcı herhangi bir ağa bağlı olmadan çalışabilmekte ve daha sonra ağa bağlanarak çalışmalarını OVA'ya gönderebilmektedir.

Temel OVA işlemleri arama, yayınlama, onaylama ve silme olarak özetlenmiştir.

3.1. Arama

OVA'daki model ve model bileşenlerinin aranmasını sağlar. Kullanıcı bir model üzerinde çalışırken ihtiyacına göre, farklı kriterler girerek arama yapar ve kullanmak istediği bileşeni kendi çalışma ortamına ekler.

3.2. Yayınlama

Model ve model bileşenlerinin OVA için yayınlanmasını sağlar. Kullanıcı kendi yerel çalışma ortamında geliştirdiği bileşenleri belli bir seviyeye getirdikten sonra onaylanmak üzere yayınlar. Yayınlama sırasında önce sistem tarafından OSAT kurallarına göre doğrulama yapılmaktadır.

3.3. Onaylama

Yayınlanmış model ve model bileşenlerinin onaylanmasını sağlar. Onaylama seçeneği sadece yetkili bir kullanıcı için aktif olmaktadır. Yayınlanmış olan bileşeni kontrol eden yetkili bu bileşeni onaylayarak kullanıcıların kullanımına açmış olmaktadır. Onaylanan bileşen yeni bir versiyon alır.

3.4. Silme

Yayınlanmış model/model bileşenlerinden uygun görülmeyenlerin silinmesini sağlar. Silme seçeneği sadece yetkili bir kullanıcı için aktif olmaktadır. Kontrol sonrasında geçersiz sayılan bileşen bu seçenek kullanılarak silinebilmektedir.

4. GÖRSELLEŞTİRME

Bir kavramsal modeli değişik açılardan ve değişik seviyelerde modelleyebilmek amacıyla tasarlanan KAMA'da, iki boyutlu diyagramlara ek olarak, üç boyutlu görselleştirme seçeneği de bulunmaktadır. Bu yöntem sayesinde diyagramları ve uygulama menülerini kullanarak geliştirilen bir kavramsal model üç boyutlu olarak incelenebilmektedir. KAMA'da görselleştirme, model üzerinde değişiklik yapmak amacıyla değil, mevcut yapıyı farklı açılardan incelemek amacıyla kullanılmaktadır.

Sistemde diyagram türlerine göre farklılık gösteren iki çeşit görselleştirme tekniği bulunmaktadır. Bunlardan birincisi olan Hiperbolik Gösterim, değişken yapılı diyagramların gösteriminde, ikincisi olan Koni Ağacı Görselleştirmesi ise, sabit yapılı diyagramların gösteriminde kullanılmaktadır.

1. Hiperbolik Görselleştirme:

   Görev Uzayı, Varlık İlişkileri ve Varlık Durum diyagramları gibi değişken yapı gösteren diyagramların görselleştirilmesinde kullanılır. Üç boyutlu bir kürenin içine model elemanları ve model elemanları arasındaki ilişkiler görüntülenir. Kürenin içinde gezilebilmekte ve istenilen model elemanları ön plana getirilebilmektedir. Balık gözü algoritması sayesinde ön planda olan model elemanına göre diğer model elemanları önem sırasına göre küçülerek gösterilirler. KAMA'da Hiperbolik Görselleştirme için 'H3Viewer' [2] kütüphanesinden yararlanılmıştır.

2. Koni Ağacı şeklinde Görselleştirme:

   Varlık Ontoloji, Komuta Hiyerarşisi ve Teşkilat Yapısı diyagramları gibi sabit yapı gösteren diyagramların görselleştirilmesinde kullanılır. Üç boyutlu uzay içinde model elemanları kutucuklar halinde, aralarındaki ilişkiler ise çizgilerle gösterilir. Model elemanları koni şeklinde, kök model elemanından aşağıya doğru ilişki bazlı olarak ağaç yapısında gösterilir. KAMA'da koni ağacı görselleştirmesi için 'Visual Toolkit (VTK)' grafik kütüphanesine [3] eklenti olarak geliştirilen 'Graph Visualization' kütüphanesinden [4] yararlanılmıştır.

KAMA'da her bir diyagram ilgili görselleştirme yöntemi ile görselleştirilmektedir. Buna ek olarak bir diyagram türüne ait tüm diyagramlardaki model elemanları da bu yöntemle görselleştirilerek, geniş bir görüş açısı sağlamaktadır.

Kavramsal modeli görselleştirme başlığı altında kavramsal modelde bulunan tüm diyagramlardaki tüm model elemanları ve ilişkileri Hiperbolik gösterim ile görselleştirilmektedir.

5. SONUÇ

Bu çalışmada simülasyonların geliştirilmesi sırasında kavramsal modellenmenin yapılması için oluşturulmuş olan sistemin yapısı anlatılmaktadır. Geliştirilmiş olan sistem standart UML modelleme aracından farklı olarak ek diyagram tipleri ve model elemanları ile farklı ihtiyaçlara cevap vermekte ve özellikle askeri simülasyonların kavramsal modellerinin oluşturulmasında kolaylıklar sağlamaktadır.

Geliştirilmiş olan ortak veri ambarı sayesinde kavramsal modellerin veya altbileşenlerinin tekrar kullanılabilirliği yeni bir kavram olarak uygulanmıştır. Bu sayede farklı kavramsal modellerde oluşturulan bilgilerin farklı projelerde tekrar ve kolaylıkla kullanımı sağlanmaktadır. Geliştirilmiş olan OSAT yapısı, kavramsal modelleme aracının sadece bir görsel model oluşturmanın yanında akıllı bir şekilde oluşabilecek hataları engellemektedir. Bu sayede her seviyedeki kullanıcı için daha rahat kullanımlı bir modelleme yöntemi oluşturulmuştur.

Geliştirilen sisteme ileriki aşamalarda senkronizasyon özelliklerinin de eklenmesi planlanmaktadır. Bu sayede farklı kişiler tarafından geliştirilen modelin otomatik olarak birleştirilmesi sağlanabilecektir. Ayrıca OVA üzerinde belirli model elemanlarının çoklu geliştirilmesini kolaylaştırma amaçlı olarak kilit (lock) sistemi gibi bazı yetenekler de sisteme eklenebilecektir.

KAYNAKÇA

[1] Karagöz, N. A., Demirörs, O., Gencel Ç., Ündeğer, Ç., (2006), "Simülasyon Sistemlerinde Görev Uzayı Kavramsal Modeli Geliştirme: Bir Süreç Tanımı", SAVTEK, SAVUNMA TEKNOLOJİLERİ KONGRESİ

[2] Munzner, T. , (1998), "Drawing Large Graphs with H3Viewer and Site Manager", Proceedings of Graph Drawing '98, Lecture Notes in Computer Science 1547, pp. 384-393, Springer-Verlag.

[3] Visualization Tool Kit (http://www.vtk.org/).

[4] Graph Visualization Library (http://www.comp.leeds.ac.uk/djd/graphs/).

TOPÇU BATARYASI İÇİN KAVRAMSAL MODELLEME VE UYGULAMALARI

Gürkan ÖZHAN [(a)], Halit OĞUZTÜZÜN[(b)]

[(a)] Bilgisayar Mühendisliği Bölümü, Orta Doğu Teknik Üniversitesi, 06531, Ankara, Türkiye, gozhan@ceng.metu.edu.tr

[(b)] Bilgisayar Mühendisliği Bölümü, Orta Doğu Teknik Üniversitesi, 06531, Ankara, Türkiye, oguztuzn@ceng.metu.edu.tr

ÖZ

Son zamanlarda modelleme ve simülasyon camiasında model tabanlı geliştirme yaklaşımına dair bir ilgi uyanmıştır. Alana özel metamodeller ve model dönüşümleri bu yaklaşımın köşe taşlarını teşkil etmektedir. Bu bildiri Sahra Topçuluğu (ST) alanının (indirekt) ateş tekniklerine dair, taktikten ziyade teknik hususları odak noktası alan bir kavramsal model sunmaktadır. Kavramsal model, GME aracıyla, metaGME altında bir metamodel olarak oluşturulmuştur. Metamodel, alan varlıklarını ve mesaj iletişimi şeklinde varlık etkileşimlerini içerir. OCL ile tanımlanmış kısıtlar model elemanlarına bağlanmıştır. Metamodeldeki bütün varlıklar NATO'nun C2IEDM elemanlarından türetilmiştir. Bunların yanı sıra, ST alan modelinden Yüksek Seviye Mimarisi (YSM) Nesne Model Şablonu (NMŞ) modeline, Model Sürümlü Mimari (MSM) felsefesine uygun dönüşümler geliştirilmektedir. Basit olarak, alan varlıkları federelere, mesaj iletişimleri de nesne ve etkileşim sınıflarına dönüştürülürler. Dahası, davranışsal yönlerin metamodel ve model dönüşüm sürecine dahil edilmesi planlanmaktadır. Bu itibarla, ST alan modelinde birtakım görev tipleri tanımlanmıştır.

Anahtar Kelimeler: Metamodel, Model, Model Dönüşümü, Sahra Topçusu, NATO C2IEDM, Görev, Yüksek Seviye Mimarisi, YSM, Nesne Model Şablonu, Model Sürümlü Mimari, OCL

CONCEPTUAL MODELLING FOR FIELD ARTILLERY BATTERY AND ITS APPLICATIONS

ABSTRACT

There has been a recent interest in the model-based development approach in the modeling and simulation community. Domain-specific metamodels and model transformations constitute the cornerstones of this approach. This paper presents a conceptual model for (indirect) fire techniques of the Field Artillery (FA) domain that focus on technical, rather than tactical issues. The conceptual model is built with the GME tool as a metamodel under metaGME. The metamodel includes domain entities and entity interactions as message communications. Constraints defined in OCL are

attached to model elements. All of the entities in the metamodel are derived from the NATO's C2IEDM elements. Besides these, transformations from the FA domain model into the High Level Architecture (HLA) Object Model Template (OMT) model are being developed, honoring the Model Driven Architecture (MDA) philosophy. Simply, domain entities are transformed into federates, and the communicated messages are transformed into object and interaction classes. Incorporation of behavioral aspects into the metamodel and model transformation process is further being planned. In this regard, some mission types are defined in the FA metamodel.

Keywords: Metamodel, Model, Model Transformation, Field Artillery, NATO C2IEDM, Mission, HLA, Object Model Template, Model Driven Architecture, OCL

1. GİRİŞ

Önceki birtakım konferanslarda Model Sürümlü Mimari (MSM)'yi Yüksek Seviye Mimari (YSM) tabanlı dağıtık simülasyona uygulamak yönünde çağrılar yapılmıştı [1-4]. MSM, günümüzün çoğunlukla kod merkezli pratiğinin aksine, yazılım geliştirmeye model merkezli bir yaklaşımı öne çıkarır. Platformdan Bağımsız Model (PBM) adı verilen üst düzey bir kavramsal modelin, Platforma Özel Model (PÖM) adı verilen, özel bir platforma has modele çevrildiği dönüşümler, MSM'nın kalbi varsayılmıştır [5].

Generic Modeling Environment (GME), alana özel modelleme ve program sentezleme ortamları oluşturmaya yarayan, konfigüre edilebilir bir araç takımıdır [6]. Konfigürasyon, uygulama alanının modelleme paradigmasını (dilini) belirleyen metamodellerle sağlanır. Modelleme paradigması alana ait bütün sentaktik, semantik ve sunum bilgisini içerir. Soyut sentaks grafiksel bir UML [7] sınıf diyagramı editörü ile, statik semantik ise Object Constraint Language (OCL) [8] ifadeleri ile belirtilir.

Graph Rewriting and Transformation (GReAT) [9-10], modelden modele dönüşüm yapmaya yarayan çizge tabanlı bir dönüştürme aracıdır. GReAT motoru girdi çizgesini (yani modelini) alır, üzerine dönüşümleri uygular, ve çıktı çizgesini (yani modelini) üretir. GReAT motoru bu süreçte hepsi de GME ortamında geliştirilmiş olan kaynak ve hedef metamodellerini, dönüşüm tanımlarını, ve kaynak modelini kullanır. Dönüşüm kuralları önceden tespit edilmiş bir düzen içinde yürütülür ve çıktı modeli bu kuralların dikte ettiği eylemler ışığında üretilir.

Bu bildiride Sahra Topçuluğu (ST) alanının (indirekt) ateş tekniklerine dair, taktikten ziyade teknik hususları odak noktası alan bir kavramsal model sunmaktayız. Kavramsal model, GME aracıyla, metaGME altında bir metamodel olarak oluşturulmuştur. Metamodel, alan varlıklarını ve mesaj iletişimi şeklinde varlık etkileşimlerini içerir. OCL ile tanımlanmış kısıtlar model elemanlarına bağlanmıştır. Metamodeldeki bütün varlıklar NATO'nun C2IEDM elemanlarından türetilmiştir. Bunların yanı sıra, ST alan modelinden YSM Nesne Model Şablonu (NMŞ) modeline, Model Sürümlü Mimari (MSM) felsefesine uygun dönüşümler geliştirilmiştir. Basit olarak alan varlıkları federelere, iletişim mesajları da nesne ve etkileşim sınıflarına dönüştürülürler. Davranışsal yönlerin metamodelleme ve model dönüşüm sürecine dahil edilmesi hedefinin bir parçası olarak da, ST alan modelinde birtakım görev tipleri tanımlanmıştır.

2. SAHRA TOPÇUSU ALAN MODELİ

ST'nin genel misyonu düşman unsurları top, roket ve füze ateşiyle imha, tesirsiz hale getirme veya baskı altına alma ve tüm ateş destek mevcudatının müşterek sınıflar harekatına katılmasına yardımcı olmaktır [12]. ST silahlarının genellikle mestur bölgelerde mevzilenmesi, ST atış tekniğini bir indirekt atış problemi haline getirmektedir. İndirekt ST atış tekniği probleminine çözüm getiren görerek atış, İleri Gözetleyici (İG), Ateş İdare Merkezi (AİM) ve atış kısımlarının bir araya gelmesinden oluşan Sahra Topçu Takımı (STT)'nin koordineli gayretleri ile gerçekleştirilmektedir.

Modelleme bağlamımızda İG bir ateş destek timinin topyekün gördüğü vazifeleri isterler çerçevesinde üstlenmiştir. Genel olarak İG uygun hedefleri tespit edip yerlerini belirler ve bu bilgileri AİM'e indirekt atış talebiyle iletir. İG açılan ateşlere ait tetkik veilerini AİM'e sağlayarak ateşin hedefler üzerine tanzim edilmesine gayret eder. İG modelimizde merkezi, merkezi olmayan ve önceden tespit edilmiş seçeneklerinden [13] sonuncusu ile; yani, belli bir bataryanın emrine atanmış vaziyette kontol edilir.

Ateş idaresi, ateş gücünün tatbik edilmesi olup, amacı her koşulda sürekli, doğru ve hızlı ateş desteği sağlamaktır [14]. Modelimizde topçu bataryasının ön planda olup hem taktik, hem de teknik ateş idaresini üstlendiği otonom kip esas alınmıştır [14]. Batarya AİM ateş isteklerinin mercii ve ateş emri, ateş komutu ve İG'ye haber gibi İG ve ateş kısımlarını yöneten komutların üreticisidir. Öte yandan, tabur AİM telsiz çevrimini takip eder ve gerektiğinde görevi iptal edip yeni bir ateş emri yayımlayabilir.Batarya ateş kısmı AİM tarafından verilen emirler doğrultusunda namluyu hedefe tevcih eder, mermiyi yükler ve ateş açar.

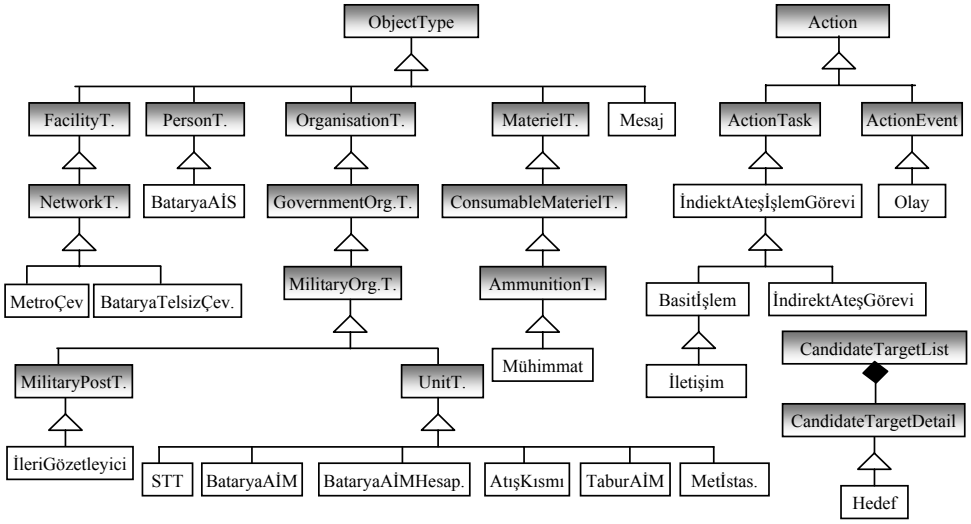
Modellemedeki varlıklar, tek tek topçuluk dünyasındaki gerçek varlıklar ve onların askeri hiyerarşi içerisindeki konumlarına odaklanmak yerine, fonksiyonel bir bakış açısıyla, görerek atış tekniği çerçevesinde temel sorumluluklar ve ana iş akışları dikkate alınarak tespit edilmiştir. Bu bağlamda STT üçlüsü İG, batarya AİM ve atış kısmından teşekkül edilmiştir. Batarya AİM ise kendi içinde batarya Ateş İdare Subayı (AİS) ve batarya AİM hesapçısından oluşmaktadır. Bu düzende batarya AİS ateş emri ve devamlı harekat usulü oluşturulmasından, ve batarya AİM hesapçısı da, aldığı ateş emirleri doğrultusunda, ateş komutlarının üretilmesinden sorumludurlar. Bu emir ve komutlar ST görevlerinde temel ve merkezi bir rol taşıdıklarından, üretici ve tüketicileri açıkça modellenmiştir. Dış varlıklarla olan ilişkilerde muhatap, modüler geliştirme prensibine uygun olarak, sadece batarya AİM olarak kabul edilmiştir.

Yukarıda bahsedilen esas varlıkların yanı sıra, teknik ateş idaresine önemli bir girdi sağlayan metro raporlarını üreten meteoroloji istasyonu, bir ateş görevinin merkezinde bulununan hedef, ateş kısımları tarafından hedef üzerine gönderilen mühimmat, tüm varlıkların içinde yaşadıkları doğal ortam ve insan yapımı ortamlar da birer varlık olarak modellenmiştir. Ayrıca, telsiz ve metro çevrimi gibi muharebe alanı çevrimleri de modellenmiştir. Bir çevrime dahil olan varlıklar, çevrimdeki tüm iletişime vakıf olmaktadırlar. Batarya çevrimine batarya AİM, İG ve tabur AİM, metro çevrimine ise meteoroloji istasyonu, batarya AİM ve tabur AİM dahildir.

Bir görev, alan varlıklarının belli kurallar ve kısıtlar çerçevesinde koordineli etkileşimleriyle icra edilir. Bu bağlamda, çeşitli STT üyeleri tarafından veriler toplanır,

emirler hazırlanır, ilan ve icra edilir. Bu süreçler modelde varlıklar arasındaki mesaj iletişimleriyle ifa edilir. Temelde bir iletişim kaynak, mesaj gövdesi ve hedef(ler)den oluşur. Ayrıca, eğer iletişim bir çevrim dahilinde cereyan ediyorsa, o çevrim de iletişimde belirtilir. Bir mesaj birtakım yapısal alanların bileşimiyle oluşan, bir görevin icrasına yönelik anlaşılır bir bilgi parçasıdır.

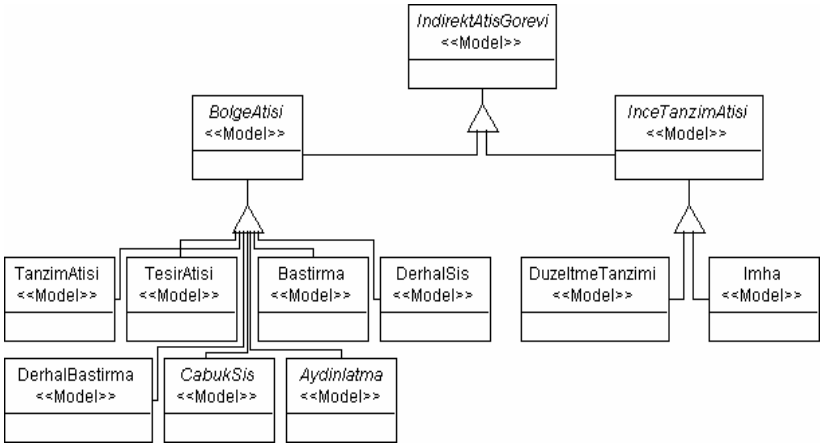
Modeldeki tüm üst düzey varlıklar, NATO Referans Modeli'nin çekirdeği ve ayrıca NATO Corporate Data Model'in bir görünüm modeli olan, C2 Information Exchange Data Model (C2IEDM) [15] elemanlarından türetilmiştir. Böylelikle model iyi bilinen, yaygın standart bir yapıya dayandırılarak, geçerlilik ve kabul ile kullanım ve entegrasyon kolaylığı elde edilmeye çalışılmıştır. Şekil 1'de üst düzey ST alan varlıkları (gölgesiz kutular) ve C2IEDM'deki uzatma noktaları (gölgeli kutular) gösterilmiştir.



Şekil 1. C2IEDM'e Bağlanmış Halde ST Alan Varlıkları

2.1. ST Görevleri

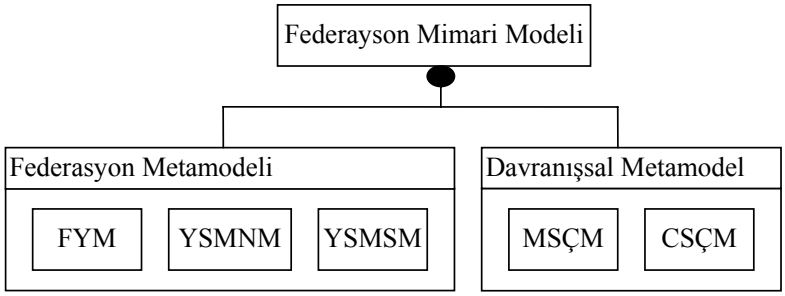
Bu çalışma, ST alanına ait statik bir model geliştirimi ve dönüşümünün yanı sıra, *görev* kavramını ortaya atarak dinamik davranışları da kapsamayı hedeflemektedir. Bu bağlamda bir ST görev hiyerarşisi oluşturulup her görev tipi için bir senaryo gerek sözlü, gerekse Canlı Seri Çizelgesi (CSÇ) diyagramı [19] olarak hazırlanmıştır. ST görevleri, [13]'deki tasnif takip edilerek, uygulanan tanzim tipine göre Bölge Atışı ve İnce Tanzim Atışı olarak ikiye ayrılırlar. Bölge Atışı da kendi içinde Tanzim Atışı, Tesir Atışı, Bastırma, Derhal Bastırma, Derhal Sis, Aydınlatma ve Çabuk Sis şeklinde tasnif edilmiştir. Genellikle tanzim atışından sonra tesir atışı icra edildiğinden, Tanzim/Tesir en temel ve yaygın görev senaryosu olup, diğerleri küçük farklılıklarla beraber benzer adımları içerir. İnce Tanzim Atışı ise Düzeltme Tanzimi ve İmha görevi şeklinde iki çeşittir. Bunların dışında kalan görev tipleri ve/veya atış teknikleri bu çalışmanın kapsamına dahil edilmemişlerdir. Alan modelinin ST görevlerine ait bölümü Şekil 2'de gösterilmiştir. ST alan modeline davranışa ait yönlerin, görev tanımları ve icra senaryoları şeklinde eklenmesine dair detaylı bilgiler için [21]'ye başvurulabilir.



Şekil 2. ST Görev Tipleri

3. FEDERASYON MİMARİ MODELİ

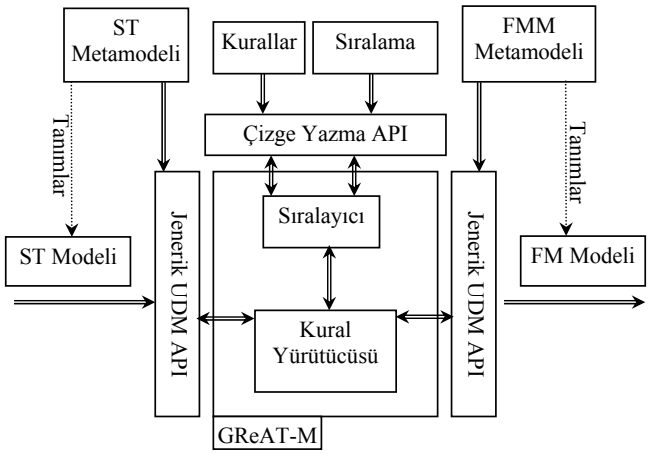
Federasyon Mimari Modeli (FMM) tüm detaylarıyla [22]'de tanımlanmıştır. Burada genel biri fikir oluşması açısından kısaca bahsedilecektir. Şekil 3'te gösterildiği üzere, FMM statik ve yapısal bilgileri tanımlayan Federasyon Metamodeli (FM) ile dinamik, gözlemlenebilir davranışları tanımlayan Davranışsal Metamodel (DM) şeklinde iki mantıki parçadan oluşur. FM kendi içerisinde Federasyon Yapısı Metamodeli (FYM), YSM Nesne Metamodeli (YSMNM) ve YSM Servisleri Metamodeli (YSMSM) kısımlarından oluşur. DM ise Mesaj Seri Çizelgesi (MSÇ) Metamodeli (MSÇM) ve CSÇ Metamodeli (CSÇM) kısımlarından oluşur.



Şekil 3. Sadeleştirilmis Federasyon Mimari Modeli ([22]'den Adapte)

4. ST – FMM DÖNÜŞÜMÜ

Dönüşüm tanımları GME'de, GReAT dilinin metamodeli olan UMLModelTransformer paradigması kullanılarak geliştirilmiştir. Model dönüşümleri GReAT Motoru (GReAT-M) ile yürütülür. Motor, ST ve FMM metamodellerini, dönüşüm tanımlarını ve bir ST modelini girdi olarak alıp, verilen ST modeline karşılık gelen bir FMM üretir. Motor, jenerik API'ler [17] kullandığından dolayı her türlü model dönüşümü yürütmeye uygundur. GReAT-M ile model dönüşüm süreci Şekil 4'te [18]'den adapte edilip sadeleştirilmiş bir taslak çizim olarak gösterilmiştir.



Şekil 4. GReAT Yürütme Motoru. ([18]'dan Adapte)

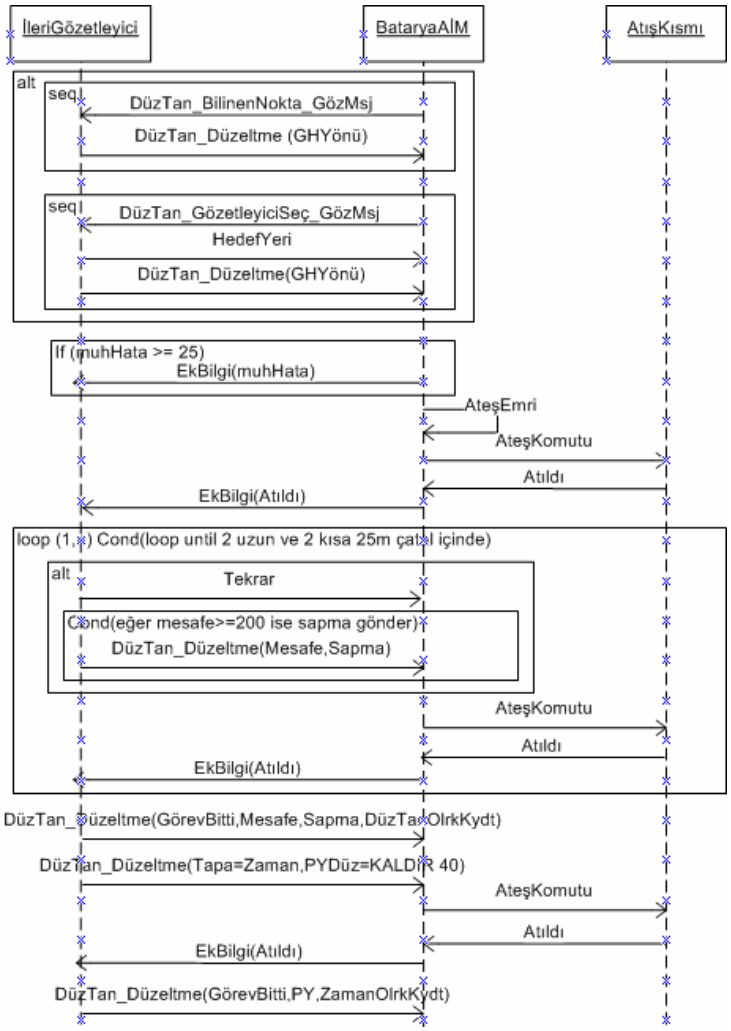
ST alan modelinden FMM'ne olan dönüşümlerde statik yapısal dönüşümlere ait yaklaşımlar, tespit edilen problemler, tercih edilen çözüm yöntemleri ve geniş bir örnek dönüşüm kümesi [20]'de anlatılmıştır. Görev icra senaryolarına ait sözel anlatımlar hazırlanıp CSÇ diyagramları çizilmiştir. Bu kısımda düzeltme tanzimi görevi örnek olay incelemesi kısaca takdim edilmiştir (detaylar için bkz. [21]). Dinamik, davranışsal model dönüşümlerinin gerçekleştirimi ise halihazırda devam etmektedir.

4.1. Örnek Olay İncelemesi: Düzeltme Tanzimi Görevi

Muharebe sahasında hava durumu, konum ve malzemeye ait standart koşulların aynı anda sağlanması hemen hemen imkansız olduğundan, bütün standart dışı koşullardan kaynaklanan birikmiş hata etkilerini ortadan kaldırmak amacıyla düzeltme tanzimi görevi icra edilir.

Düzeltme tanzimi yapılmasına karar verildikten sonra, batarya AİS, devamlı harekat usulünde bulunmayan veya oradakinden farklı olan, görev tipi, bilinen nokta (eğer İG tarafından verilmeyecekse), ateş edecek birlik, tanzimi yürütecek parça, kafile, barut hakkı ve diğer gerekli bilgileri belirten bir ateş emri hazırlayıp batarya AİM hesapçısına gönderir. Ayrıca, tanzim tipi, bilinen nokta ve tapa gibi ateş emrinin İG'yi ilgilendiren kısımları da bir Gözetleyiciye Mesaj ile gönderilir. Akabinde, ikaz emri, takip edecek ve ateş edecek parçalar, özel talimatlar, ateş etme metodu ve yan ve yükseliş açılarını içeren bir ateş komutu batarya AİM hesapçısı tarafından hazırlanıp atış kısımlarına yollanır. En sonunda da, İG, kendisinin artık tanzimi sevk etmeye hazır olduğunu belli eden yön bilgisini rapor eder.

Taraflar hazır olduktan sonra vuruş fazı başlar. Düzeltme tanziminin gayesi, aynı veya 25 metre mesafeli (hata ihtimali 25 metreden büyükse 50 metre) atış verisiyle, Gözcü Hedef (GH) hattı boyunca dört tane (iki ileri ve iki kısa) atım gözlemlemektir. Hedefe isabet veya mesafece doğru bir atım gözlemi hem ileri hem de kısa yerine geçer. Düzeltme tanziminin zaman kısmında amaç, aynı atış verisiyle yapılan dört tane atımın ortamala Paralanma Yüksekliğini (PY) düzeltme tanzim noktasının 20 metre üzerine getirmektir. Düzeltme tanzimi görevi için sadeleştirilmiş bir CSÇ diyagramı Şekil 5'te gösterilmiştir.

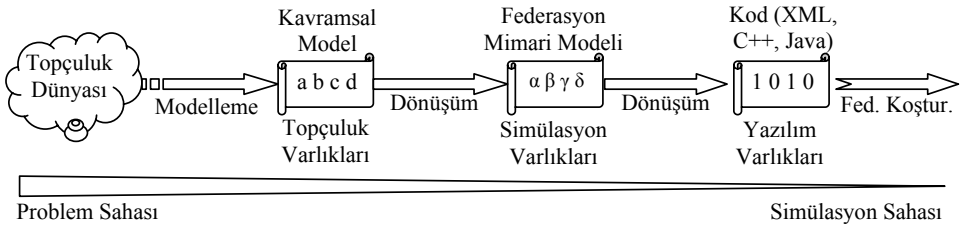


Şekil 5. Sadeleştirilmiş Düzeltme Tanzimi CSÇ Diyagramı

5. UYGULAMALAR

Genel olarak model tabanlı yazılım geliştirme ve model dönüşümü, ve özel olarak da, sahra topçuluğu kavramsal modelinden YSM tabanlı dağıtık simülasyon modeline otomatik (yarı otomatik) dönüşümün, eğitimden prototiplemeye, model geçerlemeden otomatik kod üretimine kadar geniş bir potansiyel kullanım alanı vardır. Bu kısımda, bahsi geçen kullanım alanlarına öncü bir örnek teşkil etmesi beklenen çalışmalara kısaca değinilmektedir. Bu çalışmalar, kavramsal alan modelinden başlanıp model dönüşümleri vasıtasıyla koşturulabilir kod üretimine kadar uzanan ve bu bağlamda MSM yaklaşımının bir gerçekleştirimi olduğu iddia edilebilecek bir sürecin parçalarını teşkil etmektedir. Şekil 6'de resmedilen sürecin ST kavramsal modellemesi ve ST modelinden FMM'e dönüşüm bu bildiride ele alınmıştır. Federasyon mimarisi'ne ait

metamodelleme [22]'de, FMM'den (belli bir ölçüye kadar) koşturulabilir kod üretimi ise [23]'te anlatılmıştır.



Şekil 6. Kavramsal Modelden Koşturulabilir Koda Dönüşüm Süreci

YSM uyumlu bir federasyonun mimarisini tanımlayan bir metamodel [22]'de önerilmiştir. Statik olan Nesne Modeli (NM) ve Federe Arayüz Şartnamesi (FAŞ) modellemesinin yanısıra, dinamik davranışları betimlemede kullanılmak üzere CSÇ ve MSÇ modellemesi de yapılmıştır. Gerçekleştirimi GME'de yapılan metamodelin CSÇ kısmı bu bildiride bahsi geçen ST görevlerinin tanımlanmasında kullanılmaktadır.

FMM'den federasyona ait davranışsal ve varsayılan temel Java/AspectJ kodu üreten bir kod üreticisi ise [23]'te anlatılmıştır. İlgi Yönelimli Programlama (İYP) yaklaşımı benimsenerek gözlemlenebilir davranışa ait taban kodu ve federelere özel ilgi (aspect) kodu ayrı ayrı üretilmektedir. Üretilen kod ihtiyaca ve hedef koşturma platformuna göre uzatılmalı ve/veya ilgilerle örgülenmelidir. Üretilen koda ilgili kaynak model elemanlarına işaret eden yorumlar eklenerek modelden koda takip edilebilirlik sağlanmıştır. Halihazırda sadece pitch-RTI desteklenmektedir. Bununla beraber, RTI gerçekleştirimine özel kısımlar ayrı bir XML dosyasında tutularak, farklı RTI sağlayıcılarının desteklenmesinde kolaylık elde edilmiştir.

6. SONUÇ VE TARTIŞMALAR

Bu bildiride ST indirekt ateş tekniğini içeren, teknik seviyeye odaklı, alan varlıklarının, etkileşimlerinin ve çeşitli ateş görevlerinin tanımlanabildiği bir metamodel, ve bu metamodel kullanılarak geliştirilmiş bir modelin FMM'ye otomatik dönüştürülme süreci anlatılmıştır. FMM metamodeli ve bu dönüşümün doğal devamı olarak da, FMM'den temel simülasyon kodu üretimi [22] ve [23]'te anlatılmıştır. Üretilen temel kodun İYP ile örgülenip belli bir hedef simülasyon platformu üzerinde koşturulması [24]'deki çalışmada anlatılmıştır. Böylelikle, bir gerçek hayat probleminin kavramsal alan modeli oluşturulduktan sonra, model dönüşümleri ile koşturulabilir bir YSM tabanlı dağıtık simülasyon modeline aktarılabileceği gösterilmiştir. Bu ise yazılım geliştirme dünyasında oldukça güncel ve sıcak bir konu olan MSM yaklaşımının erken bir manifestosu olarak ilan edilebilir.

Bununla beraber model döşümlerinde ciddi problemler mevcuttur. Öncelikle dönüşüm kuralları kaynak ve hedef metamodellere sıkı sıkıya bağlı olduğundan, herhangibir metamodeldeki değişiklik dönüşüm kurallarını da doğrudan etkilemektedir. Diğer bir nokta ise, kaynak ve hedef modellerde bilgi içeriği açısından farklılık olduğu durumlarda, dönüşüm kurallarına dışarıdan bilgi sağlamak ve/veya birtakım sezgisel yöntemler uygulamak gerekliliğidir. Dışarıdan bilgi sağlamanın süreci etkileşimli

kılacağı, dolayısıyla da dönüşümlerde tam otomasyonu sekteye uğratacağı bellidir. Ayrıca, dönüşüm yazarları doğal olarak her iki metamodelde de uzmanlığı gerektiren, ince ve oldukça sıkıcı olabilen bir süreçten geçmek durumundadırlar. Bu hususta kolaylık sağlama adına, daha üst seviyede ve dekleratif bir sözdizimi ile, yüksek perdeden dönüşüm tanımlama, yani dönüşüm kuralları için dönüşüm kuralları yazma yöntemleri üzerinde çalışma yapmayı hedeflemekteyiz.

MSM yaklaşımında varılması beklenen nihai nokta, hem kaynak, hem de hedef alan modellerini iyi bilen ve birincisinden ikincisine model dönüşüm kurallarını yazabilen kalifiye mühendislerin model dönüşümünü bir kere tanımlamalarından sonra, alan uzmanlarının artık sadece uzamanı oldukları alana ait modeller üretip, bunlardan muhtelif platformlara yönelik, otomatik bir şekilde doğru, koşturulabilir kod üretmeyi başarmaktır. Kamuoyu tarafından MSM'nin oldukça iddialı ve ütopik bulunduğu yaygın bir kanı olmakla beraber, makina dilinden assembly programcılığına, ve assembly'den yüksek seviye programcılığına geçilirken de benzeri tarihsel süreçlerin yaşandığı ve bu bildiride anlatılan öncü çalışmalar dikkate alınırsa, MSM'nin bir süre sonra yazılım dünyasının bir gerçeği haline geleceğini düşünmekteyiz.

TEŞEKKÜR

ST kavramsal modelinin oluşturulmasındaki katkılarından dolayı Pınar Evrensel ve Serdar Kutlu'ya, ve bu çalışmayı teşvik eden STM A.Ş.'ye teşekkür ederiz.

KAYNAKÇA

[1] A. Tolk, (2002), "Avoiding Another Green Elephant –A Proposal for the Next generation HLA based on the Model Driven Architecture", Proceedings of 2002 Fall Simulation Interoperability Workshop, Orlando, FL.

[2] S. Parr and R. Keith-Magee, (2003), "Making the Case for MDA", Proceedings of 2003 Fall Simulation Interoperability Workshop, Orlando, FL.

[3] J.S. Dahmann and J.W. Wilson, (2003), "MDA and HLA: Applying Standards to Development, Integration and Test of the Single Integrated Air Picture Integrated Architecture Behavior Model", Proceedings of 2003 Fall Simulation Interoperability Workshop, Orlando, FL.

[4] T. Peugeot and P. de Champs, (2005), "HLA and MDA at EADS SDC Simulation. Assessment of Current and Future Practice", Proceedings of 2005 European Simulation Interoperability Workshop, Toulouse, France.

[5] (2000), "The Model-Driven Architecture", http://www.omg.org/mda.

[6] A. Ledeczi, M. Maroti, A. Bakay, G. Karsai, J. Garrett, C. Thomason, G. Nordstrom, J. Sprinkle and P. Volgyesi, (2001), "The Generic Modeling Environment", Proceedings of WISP'2001, Budapest, Hungary.

[7] J. Rumbaugh, I. Jacobson, and G. Booch, (1998), "The Unified Modeling Language Reference Manual", Addison-Wesley.

[8] Object Management Group, (2001), "Object Constraint Language Specification", OMG Document formal/01-9-77.

[9] A. Agrawal, G. Karsai and A. Ledeczi., (2003), "An End-to-End Domain-Driven Software Development Framework", 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA), Anaheim, California.

[10] G. Karsai, A. Agrawal, F. Shi, J. Sprinkle, (2003), "On the Use of Graph Transformations for the Formal Specification of Model Interpreters", JUCS.

[11] D. Çetinkaya and H. Oğuztüzün, (2006), "A Metamodel for the HLA Object Model", 20th European Conference on Modeling and Simulation (Track on Modeling and Simulation Methodologies), Bonn, Germany.

[12] FM 6-50, (1990), Tactics, Techniques, and Procedures for The Field Artillery Cannon Battery, http://www.globalsecurity.org/military/library/policy/army/fm/6-50/index.html

[13] FM 6-30, (1991), "Tactics, Techniques, and Procedures for Observed Fire", http://www.globalsecurity.org/military/library/policy/army/fm/6-30/index.html

[14] FM 6-40, (1999), Tactics, Techniques, and Procedures for Field Artillery Manual Cannon Gunnery, http://www.globalsecurity.org/military/library/policy/army/fm/6-40/index.html

[15] (2003), "The C2 Information Exchange DataModel", C2IEDM Main-US-DMWG Edition 6.1, Greding, Germany

[16] U.S. Department of Defense, (1998), "Draft Standard for Modeling and Simulation High Level Architecture – Object Model Template", IEEE P1516.2.

[17] A. Bakay, "The UDM Fwk", http://www.isis.vanderbilt.edu/Projects/ mobies.

[18] A. Agrawal, Z. Kalmar, G. Karsai, F. Shi, (2003), A. Vizhanyo, "GReAT User Manual", Institute for Software-Integrated Systems Vanderbilt University.

[19] Brill, M., Damm, W., Klose, J., Westphal, B., Wittke, H., (2004), "Live Sequence Charts:An introduction to lines, arrows and strange boxes in the context of formal verification", SoftSpez Final Report, pp 374-399

[20] Özhan, G., Oğuztüzün, H., (2006), "Model-Integrated Development of HLA-Based Field Artillery Simulation", Proceedings of 2006 European Simulation Interoperability Workshop (EuroSIW), Stockholm.

[21] Özhan, G., Oğuztüzün, H., (2007), "A Metamodel for Battery-Based Field Artillery", Manüskript.

[22] Topçu, O., Adak, M., Oğuztüzün, H., (2007), "A Metamodel for Federation Architectures", Manüskript.

[23] Adak, M., Topçu, O., Oğuztüzün, H., (2007), "Model-Based Code Generation for HLA Federates", Manüskript.

[24] Efe, O., Adak, M., Oğuztüzün, H., (2007), "Davranış Belirtimlerinin Kod Üretimi Yoluyla Canlandırılması ve Bir Uygulama", Manüskript.

DAĞITIK SİMULASYON PROJELERİNİN KAVRAMSAL MODELİNİ GELİŞTİRME ÇALIŞMALARINDA BOM KULLANIMI

Yasemin TİMAR [(a)], İsmail BIKMAZ [(b)], İsa TAŞDELEN [(c)], Sevgi AKGÜN [(d)], Oğuz DİKENELLİ [(e)]

(a) (c) (d) (e) TÜBİTAK MAM Bilişim Teknolojileri Enstitüsü, 41470 Gebze/Kocaeli, yasemin.timar@bte.mam.gov.tr, isa.tasdelen@bte.mam.gov.tr, , sevgi.akgun@bte.mam.gov.tr, oguz.dikenelli@bte.mam.gov.tr ,

(b) SIMSOFT ,ODTÜ Teknokent Bölgesi, KOSGEB Teknoloji Geliştirme Merkezi NO:113 , 06531 ODTÜ ANKARA, ismail@simsoft.com.tr

ÖZ

Kavramsal modelleme dağıtık simülasyon sistemlerinin geliştirilmesinde oluşturulan modelin geliştirmenin tüm süreçlerine girdi sağlaması sebebiyle kritik bir süreçtir. Kavramsal modellemede kullanılan iyi bir meta model simülasyon bileşenlerinin ayrıştırılmasına, anlaşılmasına ve bileşenlerin başka sistemlerde kullanılabilmesine (tekrar kullanılabilirlik) olanak sağlamaktadır. High Level Architeture (HLA) ve Federation Object Model (FOM) altyapısına sahip dağıtık simülasyon projelerinin geliştirme sürecinde kavramsal modelin en önemli hedefi birlikte çalışabilirlik (interoperability), tekrar kullanılabilirlik (reusability), bileşen tabanlı geliştirme (component oriented development) gibi önemli özellikleri simülasyon sisteminin taşımasını sağlamaktır. Base Object Model (BOM)[1] simülasyon sistemlerini yönetilebilir ve tekrar kullanılabilir küçük parçalardan oluştuğunu kabul eden modelleme şablonudur. BOM'a dayalı kavramsal model çalışması bu hedeflerin gerçekleştirilmesine olanak sağlar. Bu çalışmada, BOM kullanılarak dağıtık simülasyon sisteminin kavramsal modelinin oluşturulması sürecinin ana adımları, bu modelden Realtime Platform Reference FOM (RPR FOM) uyumlu bir FOM üretilmesi süreçlerindeki deneyimler ve kullanılan UML notasyonları anlatılmaktadır.

Anahtar Kelimeler: BOM, FOM, HLA, Kavramsal Model, RPR FOM

USING *BOM* IN CONCEPTUAL MODELLING OF DISTRIBUTED SIMULATON PROJECTS

ABSTRACT

Conceptual modeling is a critical process in the development of distributed simulation systems because of the fact that output model becomes an input to the all phases of the development process. Good meta model used in the conceptual modeling ensures to understand and decompose the simulation entities and to reuse in other simulation systems. Main purpose of the conceptual modeling of distributed simulation projects which has High Level Architecture (HLA) and Federation Object Model (FOM) infrastructure is to accomplish that the simulation system has the main properties such

as interoperability, reusability, component oriented development. Base Object Model (BOM)[1] is a model template that assumes the simulation systems are composed of manageable and reusable piece-parts. Conceptual modeling using BOM ensures that the simulation system will hold those main properties. In this work, the experiences in the process of using BOM to form the conceptual model of the distributed simulation systems, main steps of the process and forming a RPR FOM compliant FOM from the conceptual model with the UML notations used will be shared.

Keywords: BOM, Conceptual Model, FOM, HLA, RPR FOM

1. GİRİŞ

HLA (High Level Architecture**)** dağıtık simülasyon sistemlerinin geliştirilebileceği esnek bir simülasyon altyapısı sunar. Böylece simülasyonların birlikte çalışabilirliği ve simülasyon bileşenlerinin tekrar kullanılabilirliği sağlanmış olur. Bu yapıda simülasyon bileşenlerini oluşturan federeler RTI (Real Time Infrastructure) üzerinden haberleşirler. HLA uyumlu simülasyonlarda federasyonun yapılandırılması ve veri akışının istenilen biçimde sağlanabilmesi için tüm simülatörlerin nesne modellerini bildirdikleri Simülasyon Nesne Model (Simulation Object Model-SOM) dosyaları ve kurulan federasyona özel ortak ve paylaşılacak nesne bilgilerinin tutulduğu Federasyon Nesne Model (Federation Object Model-FOM) dosyası hazırlanmalıdır. Simülatör geliştirmek için temel adımlar FEDEP (Federasyon Geliştirme ve İşletme Süreci – Federation Development and Execution Process)'de tanımlanmıştır. Bu adımlar arasında, kavramsal modeli oluşturma aşaması simülasyonun doğru tanımlanması için en önemli adımdır.

Kavramsal Model simülasyonu yapılacak gerçek sistemin anlaşılmasını kolaylaştırmak, işin kapsamını belirlemek amacıyla gerçekleştirilir. Özellikle karmaşık askeri simülasyonlarda geliştiricilerin görev uzayını anlamasını ve kullanıcı ile aynı dili kullanarak sistemi daha genel ve soyut düzeyde tanımlamasını sağlar. Bu sayede simülasyona getirilecek kısıtlar, alternatif akışlar belirlenebilir. Kavramsal model tasarımdan teste kadar uzanan simülasyon geliştirme süreci boyunca yol gösterici bir tanımlamadır.

FEDEP süreci kullanıcıların simülasyon sistemlerindeki ihtiyaçlarının belirlenmesinden, simülasyon sistemini test, kabul muayene işlemlerine kadar geçen aktiviteleri kapsayan ve bu aktiviteler arasındaki iletişim ve koordinasyon sorunlarını en az indirmeye çalışan bir süreçtir. FEDEP sürecinde FOM ve SOM'un oluşturulması Federasyon Tasarımı adımının etkinliklerinde önemli ve kritik işlemlerden biridir. Ancak bu kritik etkinlik öncesinde analiz aşamalarında yapılması gereken kavramsal modelin ve bu modele bağlı tekrar kullanılabilir bileşenlerin belirlenmesidir. Tekrar kullanılabilir bileşenlerin belirlenmesi FOM ve SOM'ların oluşturulmasında doğrudan kullanılacak bir çıktı sağlamaktadır.

FEDEP sürecinde, federelerin tekrar kullanılabilirliği, birlikte çalışabilirliği ve makul maliyetlerde gerçekleşmesi düşünülerek, bileşenlerin belirlenmesi ve bu bileşenleri bütünün parçalarını oluşturmasının sağlanması, kavramsal model aşamasının bir parçasıdır. Base Object Model (BOM), FEDEP süreciyle bütünleşerek simülasyon

sistemleri için çok kritik olabilecek kavramsal modellerin tanımlanmasında kullanılabilmektedir. "BOM kavramı simülasyon ve federasyon parçalarının çıkartılabilen ve tekrar kullanılabilen bloklar veya bileşenler şeklinde modellenmesini temel almaktadır" [1]. Bloklar ve bileşenler kullanılarak simülasyon sistemlerini gerçeklemek mümkün olabilmektedir.

BOM ile geliştiricilerin logo tasarımı benzeri bir tasarım yöntemi kullanarak simülasyon sisteminin geliştirmesi öngörülmektedir. Yani her bir bileşen düşünülürken bu bileşenin daha sonra başka sistemlerin oluşmasında bir bileşen olarak kullanılabileceği göz önünde bulundurulmalıdır. BOM her bir bileşenin girdisi, çıktısı ve beklenen sonuçlar ile ilgilenilirken, bileşenin iç gerçeklenmesi saklı tutulmaktadır. BOM'da her bir bileşenin nesneleri, olayları, değişkenleri hakkında detaylı bilgiler bulunmaktadır. BOM'da elde edilen veriler daha sonra FOM ve SOM'un oluşturulmasına girdi olacaktır. BOM'da her bir bileşenin diğer federelere sağladığı veya diğer federelerden aldığı veriler kavramsal olarak ele alınmaktadır. Ancak, BOM çıktıları, FEDEP sürecinde sadece FOM ve SOM oluşturulmasında değil, doğrulama, geçerleme ve test aşamalarında dahi kullanılabilmektedir.

Bu çalışmada, birinci bölümde BOM 'un temel yapıtaşları olan kavramlar, meta modelin elemanları açıklanmakta ve kavramsal modellemede kullanımları hakkında bilgi verilmektedir. İkinci bölüm örnek bir kavramsal model çalışmasında BOM kullanımını açıklamaktadır. Üçüncü bölüm kavramsal model çalışmasından FOM oluşturulmasını anlatırken, çalışmalarımızdan elde ettiğimiz deneyimler sonuçlar bölümünde yer almaktadır.

1. BOM ile KAVRAMSAL MODELLEME

BOM sağladığı temel fonksiyon ile kavramsal modellemenin gerçeklenmesine katkıda bulunarak, detay gerçeklemesi ile gerçeklemenin arayüzünü birbirinden ayırmaktır. BOM ile oluşturulan bloklar, federeler içerisindeki nesneleri, federelerin birbirleri ile ilişkilerini, federelerin kendi içlerindeki sorumluluklarını içermektedir. Bunu yaparken tüm BOM sürecinin etkinlikleri ve bileşenleri ilgili standartta belirtilmiştir. Bu bileşenler aşağıdakiler gibidir [1].

Model Tanımlanması

Kavramsal Model Bileşenleri

- Etkileşim Şablonları (Pattern of Interplay)
- Durum Makinesi (State Machine)
- Varlık Tipleri (Entity Type)
- Olay Tipleri (Event Type)

Model Eşlemesi

- Varlık Tipleri Eşlemesi
- Olay Tipleri Eşlemesi

Nesne Model Tanımlaması

- HLA Sınıfları
- HLA Mesajları (Interaction)

- HLA Üye Değişkenleri (Attribute)
- HLA Parametreleri
- HLA Veri tipleri

BOM ile tüm bu modellerin çıkartılması ve kavramsal modelden en son FOM ve SOM'un oluşturulması sağlanabilmektedir. Nesne Model Tanımlaması işlemleri esnasında RealTime Platform Reference (RPR) [4] standardının göz önüne alınarak FOM ve SOM oluşturulması birlikte çalışabilirlik açısından son derece önemlidir. RPR standardı SISO tarafından desteklenen, ve gerçek zamanlı simülasyon sistemlerinde temel olacak, ve en azından simülasyon sistemlerinin temel olan RPR FOM üzerinden konuşmasını sağlayabilecek bir nesne şablon modelidir.

Yukarıda listelenen model bileşenleri, çeşitli çevrimler kullanarak geliştirilen simülasyon sisteminin daha gerçekçi detaylanması açısından önemlidir. Model tanımlaması hazırlanan BOM modelleri ile ilgili detaylı bilgi verilmesini sağlamaktadır. Kavramsal Model bileşenleri, sistemin tüm katılımcıları açısından anlaşılması ve gerçeklemeye doğrudan girdi olarak kullanılması açısından gereklidir. Etkileşim şablonları (modelleri) bileşenler arasındaki etkileşimin belirlenmesini amaçlamaktadır. Etkileşim esnasındaki tüm olayların çıkartılmasını, varyasyon ve istisnaların belirlenmesini ve varlıkların ortaya çıkmasını sağlamaktadır. Her bir olayın tetikleyicilerinin ve olay ile ilgililerin belirlemesi için etkileşim şablonları temel modeldir. Durum Makinesi (State Machine) bir veya birden fazla etkileşim modellerinde kullanılan varlıkların durumlarının ve olaylar karşısındaki davranışlarının detaylandırılmasında kullanılmaktadır. Varlık tipleri kavramsal modelde etkileşimleri sağlayan, olayları başlatan veya olaylardan etkilenen nesnelerdir. Varlık tipleri etkileşim modellerinden ve durum makinesinden çıkartılabilmektedir. Olay tipleri ise kavram olarak modellerde karşılaşılan varlıkların durum değiştirmesine sebep olan, olaylar, varyasyonlar ve istisnalara karşılık gelmektedir. Olay tiplerini iki grup altında toplamak mümkündür; BOM tetikleyicileri ve BOM mesajları. Kavramsal modeldeki varlıkların bir durum karşısında olay tetiklemesi mümkün olabilmektedir. Bu durumda tetikleme durumunun da modelde tespit edilmesi gerekmektedir. Tetiklenme olayı HLA'de nesne üye değişkenlerinde güncelleme meydana gelmesi durumunda veya mesaj gelmesi durumunda mümkün olabilmektedir. BOM mesajları ise iki varlık arasında etkileşimin olması durumunda meydana gelmektedir. BOM mesajları da HLA'de nesne üye değişkenlerin güncelleme meydana gelmesi durumunda veya mesaj gönderilmesi durumuna karşılık gelebilmektedir.

Model eşlenmesi ile oluşturulan kavramsal model ile HLA nesne modellerinin eşlenmesi sağlamaktadır. Varlık tipi ve olay tipi eşlemeleri ile kavramsal modeldeki nesnelerde HLA yapıları arasında eşlemeler gerçekleştirilmektedir. Bu eşlemeler ile HLA sınıfları, mesajlar, üye değişkenler ve parametreleri oluşturulmaktadır. Bu eşlemelerde çeşitli tablolar kullanılmaktadır. Burada yapılan çift taraflı izlenebilirlik tabloları oluşturularak kavramsal model ile nesne modelleri arasında bağlantı sağlamaktır.

2. BOM STANDARDINA DAYANAN KAVRAMSAL MODEL ÇALIŞMASI

Bu bölümde, BOM standardına dayanan kavramsal model çalışmasına örnek olmak üzere dağıtık simülasyon ortamında benzetimi yapılmak istenen gemi seyrine ilişkin senaryo tanımlama, etkileşimlerin belirlenmesi ve olay tablolarının oluşturulması anlatılacaktır. İlerleyen bölümlerde bu örnek üzerinden BOM-FOM tanımlama çalışması yapılacaktır.

BOM çalışması, gerçek dünyada karşılığı olan ve simülasyon ortamında benzetimi yapılması öngörülen simülasyon bileşenlerinin belirlenmesinde, bileşenlerin birbirleriyle olan ilişkilerinin, veri alışverişinin ortaya çıkarılmasında yardımcı olduğu kadar, simülasyonun yapıldığı HLA federasyonuna dahil olacak federelerin hangi bileşenleri içereceğini, dolaysıyla federeler arasındaki iletişim içeriğini belirlemede etkili bir yöntemdir.

HLA Federasyonunda bulunan tüm nesne ve mesajlar FOM (Federation Object Model)'da, Federasyona dâhil olan tek bir federenin kullandığı tüm nesne ve mesajları SOM (Simulation Object Model)'da belirtilir. BOM çalışmasıyla birlikte federeler belirlenirken federelerin SOM'ları ortaya çıkar. Federasyonda yer alacak tüm federelere ilişkin SOM'ların birleşimi FOM'u oluşturmaktadır. Diğer yandan referans olarak kullanılan bir FOM varsa; yani federasyonun özellikle de öngörülen hazır simülasyon sistemleri ile birlikte çalışabilirliği planlanıyorsa ve RPR FOM gibi standart kabul edilen bir FOM referans olarak kabul edilmiş ise referans alınan FOM yapısında tanımlı olmayan SOM nesne ve mesajları, referans FOM'un bütünlüğünü bozmayacak şekilde eklenmelidir.

Kavramsal model çalışmasının ana hedeflerinden biri geliştirilecek olan sistemin işlevselliğini ifade etmek ve bu işlevleri gerçekleştirecek bileşenlerin tanımlamaktır.

Proje isterleri temel alınarak, BOM da tanımlanan karşılıklı etkileşim modelleri (pattern of interplay) [1] tespit edilir. Etkileşim modeli, etkileşimi bir senaryo olarak ifade eden olaylar grubunu, bu olayların alıcısı ve vericisi durumundaki simülasyon varlıklarını ve olayların içeriğini tanımlayan olay tablosundan oluşmaktadır.

Örnek olarak benzetimi yapılacak olan bir geminin hareketinin "Gemi belirtilen bir noktaya doğru seyir yapacak." cümlesiyle açıklanan bir etkileşim modelini oluşturmak için aşağıdaki işlemler gerçekleştirilir.

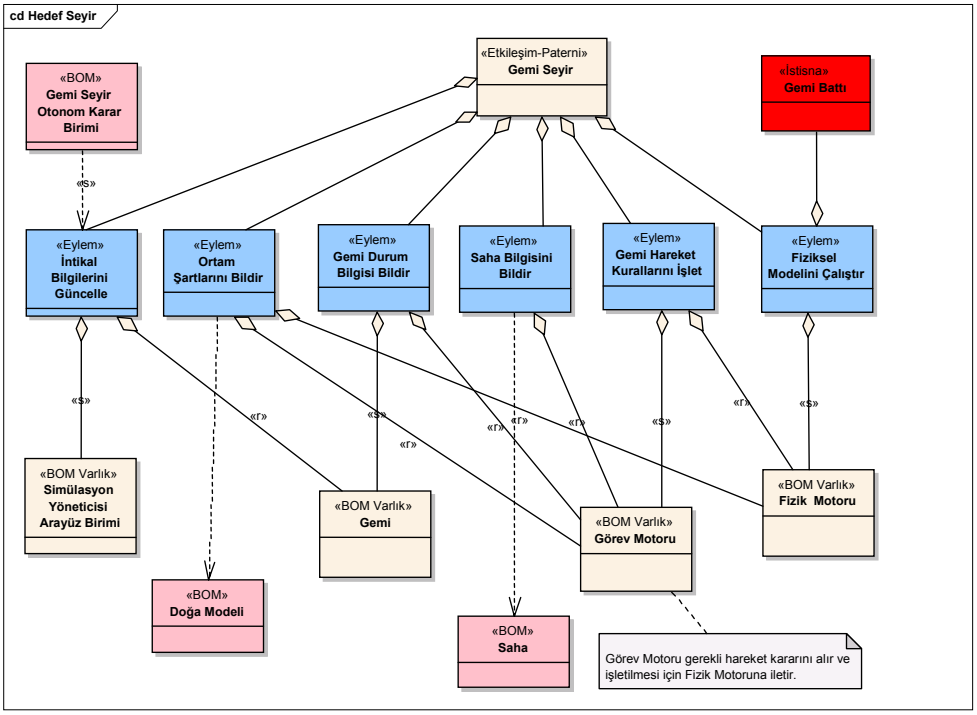
2.1. Etkileşim Senaryosunun Yazılması

BOM senaryosu olarak ifade ettiğimiz etkileşim senaryosu, senaryoya ilişkin açıklama, başlangıç girdileri, önkoşullar, adımlar, alternatif akışlar, istisna durumlar ve sonlandırma bölümlerinden oluşmaktadır. Etkileşimde bulunan simülasyon varlıklarını arasında gerçekleşen olayları adım olarak ifade eder.

2.2. Senaryo Tablosunun Oluşturulması

Etkileşimde bulunan ve benzetimi yapılacak olan simülasyon varlıkları tespit edilir ve etkileşim senaryosunun her adımını karşılayan olay alıcı ve vericisiyle birlikte UML kullanılarak BOM standardında yer alan etkileşim diyagramları olarak ifade edilir.

Etkileşim diyagramları, UML sınıf diyagramlarında tip bilgisi ile belirtilen BOM meta modelinde yer alan olay, varlık vb. ifade edilmesiyle oluşturulur. Gönderici ve alıcı özeliği "r" ve "s" ile gösterilir.



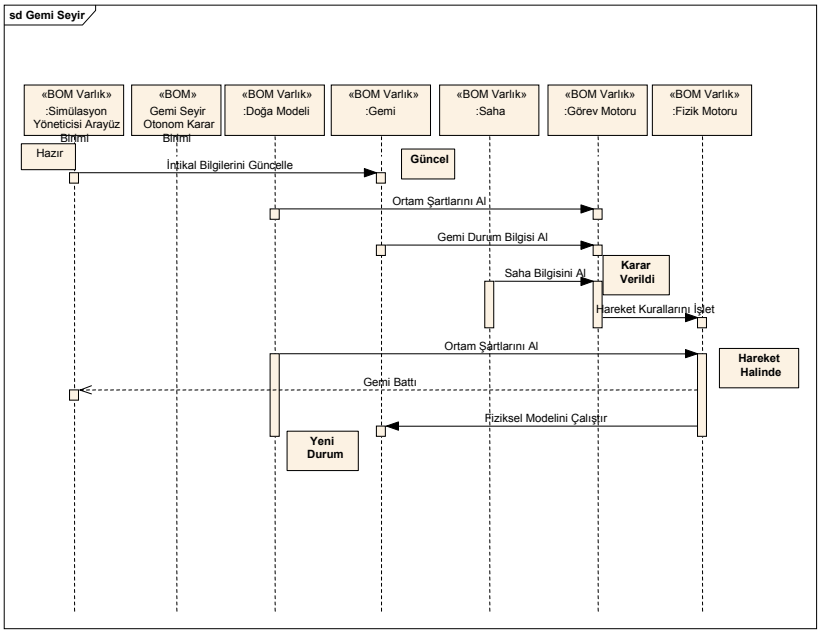
Şekil 1. Gemi Seyri Etkileşim Diyagramı

Görsel olarak ifade edilen etkileşim senaryosu için bir tablo oluşturulur ve her olay varsa bağlı olduğu koşulla/tetikle birlikte tabloya eklenir. Örnekte Gemi Seyri etkileşiminin tablosunun bir kısmı gösterilmektedir.

Sıra	Davranış	Eylem İsmi	Gönderen	Alan	Olay	BOM	Koşul
1	Eylem	İntikal Bilgilerini Güncelle	Simülasyon Yöneticisi Arayüz Birimi	Gemi	IntikalBilgiMesaj	Gemi Seyir Otonom Karar B. Arayüz Birimi.	
2	Eylem	Ortam Şartlarını Bildir	Doğa Modeli	Görev Motoru	OrtamSartlarMesaj	Doğa Modeli	
3	Eylem	GemiDurum Bilgisi Bildir	Gemi	Görev Motoru, Simülasyon Yöneticisi Arayüz Birimi	GemiDurumMesaj		
4	Eylem	Saha Bilgisini Bildir	Saha	Görev Motoru	SahaBilgisiMesaj	Saha	
5	Eylem	GemiHareket Kurallarını İşlet	Görev Motoru	Fizik Motoru	HareketKuralMesaj		

Tablo 1. Gemi Seyri Etkileşim Tablosu

Ayrıca etkileşim senaryosu aşağıdaki şekilde görüldüğü gibi sekans diyagramı olarak ifade edilirken simülasyon varlıklarının durum geçişleri belirtilir.



Şekil 2. Gemi Seyri Sekans Diyagramı

2.3. Olay Tablosunun Oluşturulması

Her bir olayla ilişkili mesajların içeriklerini tanımlayan bir tablonun oluşturulması özellikle mesaj içeriklerinin detaylandırılması amacıyla yapılır. Bu içerik ileride

yapılacak BOM-FOM çalışmasında nesnelerin üye değişkenlerini ve/veya mesajların parametre içeriklerini oluşturmak için kullanılacaktır.

İsim	**Gönderici Karakteristikleri**	**Alıcı Karakteristikleri**	**İçerik Karakteristikleri**	**Tetik Koşulu**
IntikalBilgiMesaj	Senaryo Giriş Arayüzü	Gemi	Hız	
			Rota	
			3BKonum	
			Süre	
OrtamSartlarMesaj	Doğa Modeli	Görev Motoru, Fizik Motoru	Hava Durumu	Periyodik
			Yoğunluk	
			Deniz Durumu	
GemiDurumMesaj	Gemi	Görev Motoru, SİMKON Arayüz Birimi	3BKonum	Periyodik
			Hız	
			Silah Listesi	
			Sensör Listesi	
			Kerteriz	
SahaBilgisiMesaj	Saha	Görev Motoru	DTED Bilgisi	YeniSahaMi = TRUE
HareketKuralMesaj	Görev Motoru	Fizik Motoru	Dümen Açısı	Peryodik
			Fren faktör	
			Hız faktör	

Tablo 2. Gemi Seyri Olay Tablosu

3. BOM-FOM ÇALIŞMASI

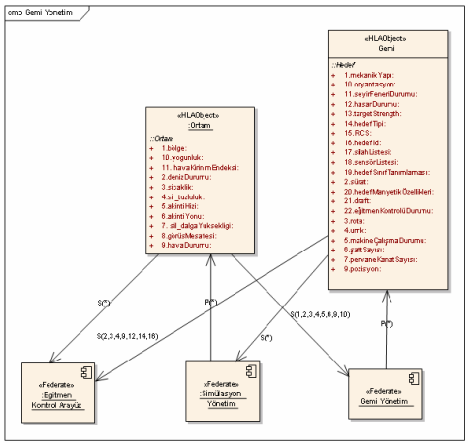
BOM çalışmasıyla ortaya çıkan simülasyon varlıkları ve etkileşimleri simülasyon altyapısına uygun bir veri formatında tanımlanmalıdır. Dağıtık benzetim ortamı olan HLA de, federeler ve federeler arasındaki nesne/mesaj alışverişiyle gerçekleşen iletişimi ifade etmek gerekir. Bu ifade tam olarak etkileşim senaryosunun nasıl gerçekleneceğini açıklayan bir ifadedir.

Simülasyon varlıklarının belirli kısıtlar göz önüne alınarak bir federeye karşılık gelmesi ya da birbiri ile ilişkili birden fazla varlığın federe oluşturmasına karar verilir. HLA federasyonunda bulunacak federelerin belirlenmesi sırasında göz önünde tutulan ana noktalar aşağıda sıralanmıştır;

- Proje isterleri
- Federasyonun mimari kısıtları
  - Simülasyonun donanım altyapısı
  - Yoğun veri akışı gereksinimi
  - Veri iletiminde gecikmenin az olması
  - Fazla sayıda simülasyon varlığı gerekliliği
  - Belirli bir zaman aralığında bir işlem biriminin veri işleme yeteneği
- Yeniden kullanılabilirlik.

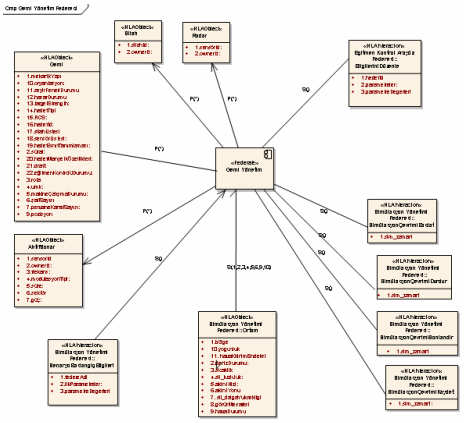
Yukarıdaki kriterler göz önünde tutularak bileşenlerin arasındaki iletişim, veri akışları ve bileşenlerin durumları etkileşim senaryoları ile ortaya çıkarılır. Hazırlanan tüm etkileşim senaryoları incelenir ve olayların niteliğinden ve içeriğinden yararlanılarak

federasyonda bulunması gereken federelerin karakteristiği, sorumluluğu ve hangi bileşenleri içereceğine karar verilir. Etkileşim senaryolarındaki olaylar federeler arasındaki nesne veya mesajların oluşturulmasına katkıda bulunur. Olayların içerikleri nesnelerin üye değişkenleri, mesajların ise parametrelerini oluşturur. Anlık bilgilerin mesaj olarak, simülasyon boyunca yaşayan bileşenlerin nesne olarak ifade edilir. Bu çalışma sonucunda nesneler ve üye değişkenlerin federeler arasındaki "yayınlama" (publish) "kayıt olma" (subscribe) ilişkisi gösteren bir şekil ve mesajların federelerle ilişkisini gösteren ikinci bir şekil çizilir. Bu ifade biçimi BOM veya herhangi bir modelleme standardının bir parçası değildir. Özgün bir yaklaşımla ortaya çıkarılmıştır.



Şekil 3. BOM-FOM Diyagramı

Gelinen noktada geliştirilen simülasyon sistemini oluşturan federasyon dahilindeki tüm federelerin hangi simülasyon bileşenlerini/ varlıklarını içerdikleri, sorumlulukları ve birbirleriyle etkileşimi tanımlanmış durumdadır. Diğer bir ifadeyle bir federenin SOM 'una kaynaklık edecek kavramsal federe arayüzü oluşturulmuş olur. Kavramsal federe arayüzü ifade biçimi de herhangi bir modelleme standardının parçası olmayıp özgün bir yaklaşımla oluşturulmuştur.



Şekil 4. BOM-SOM Diyagramı

SOM lar birleştirilerek FOM oluşturulur. Birlikte çalışabilirlik ve tekrar kullanılabilirlik söz konusu olduğunda ise referans alınacak FOM'a bu SOM'lar bütünlüğü bozmayacak ve etkileşimler tekrar ifade etmeyecek biçimde eklenir. Böylece federeler arasındaki nesne ve mesaj iletişimini ifade eden FOM oluşturulmuş olur.

4. SONUÇLAR

Kullanıcı isterlerinin tam olarak karşılanabilmesi ve tasarımın gerçek sistem işlevlerini yansıtması için test ve doğrulama çalışmalarına da yol gösterecek bir kavramsal model çalışması yapılması gerekmektedir. Bu süreçte görev uzayını yakından bilen uzmanlarla birlikte yapılması zorunludur. Yapılan çalışma sonucunda elde edilen etkileşim senaryoları federelerin entegrasyonu safhasında doğru etkileşimin gerçeklendiğinin belirlenmesinde anahtar rol oynarlar. Federelerin işlevselliğini ve birbirleriyle etkileşimini ifade eden etkileşim senaryoları test edilebilir senaryolar olarak ortaya çıkar.

Simülasyon sistemini oluşturan bileşenlerin aynı seviyede ayrıştırılamaması kavramsal model çalışmasında karşılaşılan güçlüklerden biridir. Sistemin ana işlevleri göz önünde bulundurularak bileşen ayrıştırma seviyesini kontrol altında tutmak gereklidir. Özgün yaklaşımlarla ortay çıkan ifadeler ile ayrıştırma seviyesi Federeler ve içerdikleri simulasyon bileşenleri olarak BOM-FOM diyagramlarıyla kolayca ortaya konulabilir.

BOM kullanılarak yapılan modelleme çalışmaları sonucunda, teknik karmaşıklığı yüksek, çok bileşenli dağıtık simülasyon sistemlerinin, işlevsel ve davranışsal özelliklerinin, isterlerinin, simülasyon bileşenlerinin, ilişkilerinin, etkileşimlerinin vb. belirlendiği kavramsal model çalışması ve ürünleri, geliştirme sürecini hızlandırmaktadır. FEDEP in tüm adımları uzun zaman almaktadır. BOM kullanılarak, her adım büyük ölçüde kısaltılabilir.

KAYNAKÇA

[1] Simulation Interoperability Standards Organization (SISO) Base Object Model Product Development Group (31 March 2006), "Base Object Model (BOM) Template Specification", SISO-STD-003-2006

[2] IEEE Std 1516.3™-2003, IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP).

[3] CODE-Net XML, SOAP and Simulation Development Paul Gustavson, Larry Root, Steve Goss, Jane Bachman Synetics.

[4] Simulation Interoperability Standards Organization (SISO), (2003) "Guidance, Rationale, and Interoperability Manual for the Real-timePlatform reference Federation Object Model (RPR FOM).

USMOS 2007, 18-19 Nisan 2007, ODTÜ, Ankara

CİDDİ OYUN VE SİMÜLASYON ALTYAPISI

İsmail BIKMAZ [(a)], Gökçe YILDIRIM [(b)], Veysi İŞLER [(c)],
Kürşat ÇAĞILTAY [(d)], Kayhan İMRE [(e)]

[(a)] Simsoft, ODTÜ KOSGEB-TEKMER, ismail@simsoft.com.tr

[(b)] Simsoft, ODTÜ KOSGEB-TEKMER, gokce@simsoft.com.tr

[(c)] ODTÜ, Bilgisayar Mühendisliği Bölümü, isler@ceng.metu.edu.tr

[(d)] ODTÜ, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü, kursat@metu.edu.tr

[(e)] Hacettepe Üniv., Bilgisayar Mühendisliği Bölümü, ki@hacettepe.edu.tr

ÖZ

Bilgisayar teknolojilerinde meydana gelen hızlı gelişmeler, geniş ağ kullanımı, hızlı işlemciler, güçlü ekran kartları simülasyon ve oyun teknolojilerinde önemli katkılar sağlamaktadır. Ciddi oyun teknolojileri günümüzde çok popüler olan geliştirme konularından biridir. Bununla birlikte simülasyon ve ciddi oyun teknolojileri arasında ciddi bir bağ ve yakınlık oluşmuştur. Ciddi oyun ve simülasyon altyapısında bulunması gereken birçok bileşen vardır. Hem oyun hem de simülasyon teknolojilerinde kullanılması gereken bileşenleri 2B/3B grafik çizim bileşeni, "scene graph" bileşeni, fizik bileşeni, animasyon bileşeni, yapay zeka bileşeni, ses bileşeni ve betik bileşeni olarak saymak mümkündür. Simülasyon ve oyun altyapısının çözmesi gereken önemli bir bileşen de ağ bileşenidir. Hem simülasyon teknolojilerinde hem de oyun teknolojilerinde istemci/sunucu veya dağıtık mimariler uygulanabilmektedir. Dolayısıyla ağ modülü ciddi oyun ve simülasyon altyapısı için önemli bir bileşendir. Bu bileşenlerin bir arada uyumlu çalışmasını sağlamak ve mimari açıdan esnek çözüm önerilerini içermek platformun önemini artırmaktadır. Tüm bu temel fonksiyonlar sağlandığında yapılması gereken, tüm altyapıya önemli bir katma değer sağlayacak senaryo geliştirme araçlarının eklenmesidir. Bu çalışmada ciddi oyun ve simülasyon altyapısı geliştirmede gerekli olan bileşenlerin detayları, problemler ve çözümleri, ve altyapının genel mimarisi anlatılmaktadır.

Anahtar Kelimeler: Ciddi Oyun ve Simülasyon Sistemleri, 2B/3B Grafik Çizim, "Scene Graph", Yapay Zeka, Ağ, HLA

SERIOUS GAME AND SIMULATION FRAMEWORK

ABSTRACT

Advances in computer technologies, broad network usage, high speed processors and advanced graphics cards have provided high contributions in simulations and games technologies. Serious games are one of currently very popular development topics. However, there has become a strong link between simulations and serious games. There are many components that should exist in serious game and simulation framework. Major components that should be found in both serious game and simulation technologies are 2D/3D renderer, scene graph, physics, animation, artificial intelligence, audio and script. Network is a vital component that should be solved in a game and simulation framework. Distributed architectures are used in both simulation and game technologies. Therefore, network component is very important for a serious game and simulation framework. Interoperability of all these components and the flexible architecture increase the importance of this serious game and simulation platform. When all these functionalities are satisfied, scenario development tools should be added to the platform, which increases the usability of the platform. In this study, the details of components that are necessary in developing a serious game and simulation framework, potential problems and solutions, and architecture of this platform are explained.

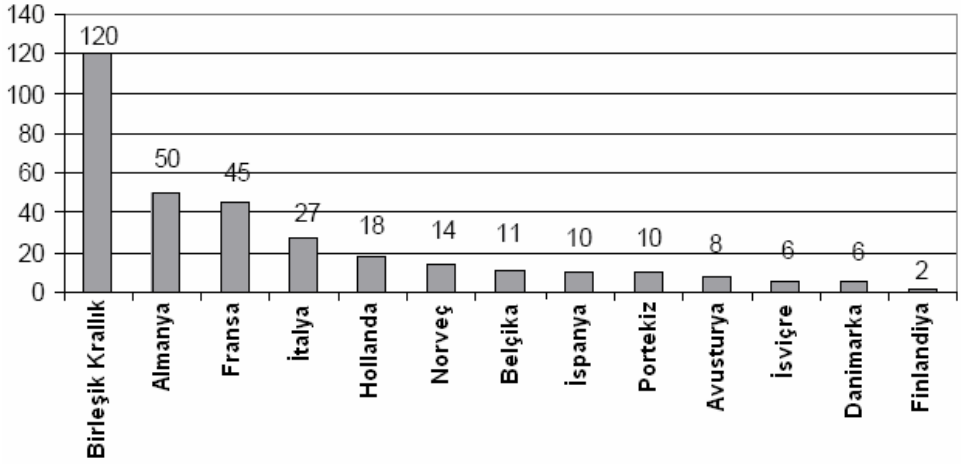
Keywords: Serious Games and Simulation Systems, 2D/3D Renderer, Scene Graph, Artificial Intelligence, Network, HLA

1. GİRİŞ

Eğlendirmenin yanısıra eğitim ve öğretim amaçları güden ve dünya olaylarının bir benzetimini sağlayan ciddi oyunlar yalnızca belirli bir yaş grubunu değil tüm yaş gruplarına hitap etmektedir. Oyunlar üzerine yapılan çalışmalar sonucunda, bilgisayar oyunlarının özellikle çocukların kişisel gelişimleri, öğrenme durumları, eleştirel düşünmeleri, yaratıcılıkları, kavramsal öğrenmeleri gibi bir çok konuda olumlu etkiler oluşturduğunun yanısıra yetişkin eğitiminde de motivasyonu arttırıcı ve oyunsal öğrenmeyi sağlayıcı yönde olumlu etkileri olduğu saptanmıştır [1, 2, 3, 4, 5, 6]. Ciddi oyunlar eğitim, askeri, devlet, sağlık, kurumsal ve reklamcılık gibi pek çok sektörden farklı yaş grubu insanlara ulaşabilmektedir. Ciddi oyunlar dünyanın en büyük oyun konferansları arasında olan GDC ve E3'te zirve konusu olurken en önemli simülasyon ve eğitim konferanslarından biri olan I/ITSEC'te de ayrı bir oturum olarak yer almaya başlamıştır. Amerikan Ordusu oyununun ABD'de askere kabul testi olarak kullanılması ciddi oyunların önemini ortaya koyan örneklerinden sadece birisidir.

Oyun sektörü tüm dünya toplamı düşünüldüğü zaman inanılmaz rakamlara ulaşmıştır. 2002 yılında donanım ve yazılım birlikte alındığında sektörün toplamı 19 milyar $

olarak tahmin edilmektedir. Oyun sektörü 1997 yılından beri film ve müzik endüstrilerinin önünde yer almaya başlamıştır [14, 15]. Dünyadaki 1.1 milyar cep telefonunun büyük bir çoğunluğu üzerlerinde hazır oyunlarla satışa sunulmaktadır. Böylesi inanılmaz boyutlara ulaşan oyun sektörü pek çok ülkenin ve şirketin dikkatini çekmektedir. Bazı ülkeler (örn. Kore) elektronik oyunları stratejik öneme sahip bir sektör olarak ilan etmektedir. Ayrıca, bilgisayar oyunları sadece eğlence sektöründe kullanılan bir araç olmaktan çıkmış, bunun yanında iş eğitimi, askeri simülasyonlar ve sağlık sektörü gibi bir çok alanda aktif olarak kullanılmaktadır [7]. Bütün bu gelişmelere karşın maalesef özellikle ciddi oyunlar konusunda ülkemizde yapılan çalışmalar çok cılız kalmaktadır. En son yapılan araştırmalarda Türkiye'nin oyun geliştirme çalışmalarında Avrupa'dan çok geri durumda olduğu görülmektedir. Avrupa elektronik oyun çalışmalarının 2005'teki durumunun sunulduğu OECD raporunda ülkeler sıralamasında Türkiye'nin henüz listeye dahi giremediği görülmektedir (Şekil 1). Elektronik oyunlar konusunda Türkiye'nin geri kalma nedenlerinden birisi de bu konuda gerek teorik ve gerekse pratik akademik çalışmaların da çok kısıtlı olmasıdır.



Şekil 1. Avrupa ülkelerindeki oyun geliştirme stüdyo sayısı ([25])

Türkiye'de oyun sektöründen kazanılan gelirin donanım ve yazılım göz önüne alınırsa muhtemelen bir milyon dolar civarında olduğu tahmin edilmektedir ki bu rakam bilişim sektörünün tüm büyüklüğü göz önüne alındığı takdirde göz ardı edilebilir. Teorik olarak bir ilçenin tüm sene patates ekerek ülkeye kazandırdığı rakamı 15-20 kişinin çalıştığı bir ofisten beyin gücüne dayanarak geliştirilmiş bir çevrim içi oyun ile kazanmak mümkün iken pratikte bu yönde adımlar atılmaması ülkemiz adına büyük bir eksikliktir.

Ülkemizdeki bu eksikliğin yanı sıra dünyada da bu alanda doldurulması gereken boşluklar bulunmaktadır. ABD'nin Los Angeles kentinde düzenlenen 2006 E3 Dünya Video Oyunu Fuarına katılan bu sektörün uzmanları elektronik oyun teknolojisinin geleceğe yönelik projeleri arasında, oyun konsolu, bilgisayar ve cep telefonu kullanıcılarının aynı anda internet üzerinden oyun oynayabilmelerinin bulunduğunu

belirtmişlerdir. Elektronik oyunlarda çoklu platform düşüncesinin sektörün geleceği açısından önemli olduğunu açıklamışlardır. Microsoft'un ileri teknolojiler grubu Başkanı Laura Flyer, özellikle ekonomik nedenlerden ötürü bu yönde bir gelişme bulunduğunu kaydederek, oyunların pahalı hale geldiğini, çoklu platformun, oyun üreticilerinin gelecekte ayakta kalmaları için tek çare olacağını belirtmiştir.

Türkiye ve diğer ülkelerdeki bilgi ve iletişim teknolojilerinin eğitimdeki uygulamalarına baktığımızda çok başarılı örnekler bulmakta zorlanılmaktadır. Genelde uygulamalar "drill and practice" türünün dışına çıkamamakta olup küçük Flash tabanlı oyunlar ile kısıtlı kalmaktadır. Kritik düşünme yetisini geliştiren türde uygulamalar nadir bulunmaktadır. Age of Empires ya da SimEarth gibi örnekler ise eğlence öncelikli oldukları için eğitimde kullanımları kısıtlı kalmaktadır. Bu konudaki açığı kapatmak için son yıllarda özellikle ABD'de, yoğun gerçekli oyunların eğitimde kullanımına yönelik bir eğilim belirmiştir. Harvard üniversitesinin dünyaca tanınan profesörü Chris Dede'in RiverCity projesi veya Massachusetts Institute of Technology ile Microsoft işbirliğinde sürdürülen GamesToTeach projeleri bunlar arasında en bilinenlerdir. Ancak Learning & Leading with Technology dergisinin Nisan 2006 sayısında Alix Peshette ve David Thornburg'un da belirttiği gibi öğrencileri test çözmeye hazırlayan pek çok oyun yazılımı bulunurken onları yüksek seviyeli düşünmeye hazırlayan örnekler yeterli değildir.

Bilgi ve iletişim teknolojilerinin özellikle ciddi oyunlar ve simülasyonlar oluşturulmasında etkin kullanımıyla ilgili olarak yaşanan eksiklikler, çok platformlu, yoğun gerçekli ve yığınsal çok kullanıcılı altyapıyı zorunlu kılmaktadır. Bu bildiride çok platformlu, yoğun gerçekli ve yığınsal çok kullanıcılı ciddi oyun ve simülasyonların üretilmesini sağlayan bir altyapı çalışması sunulmaktadır.

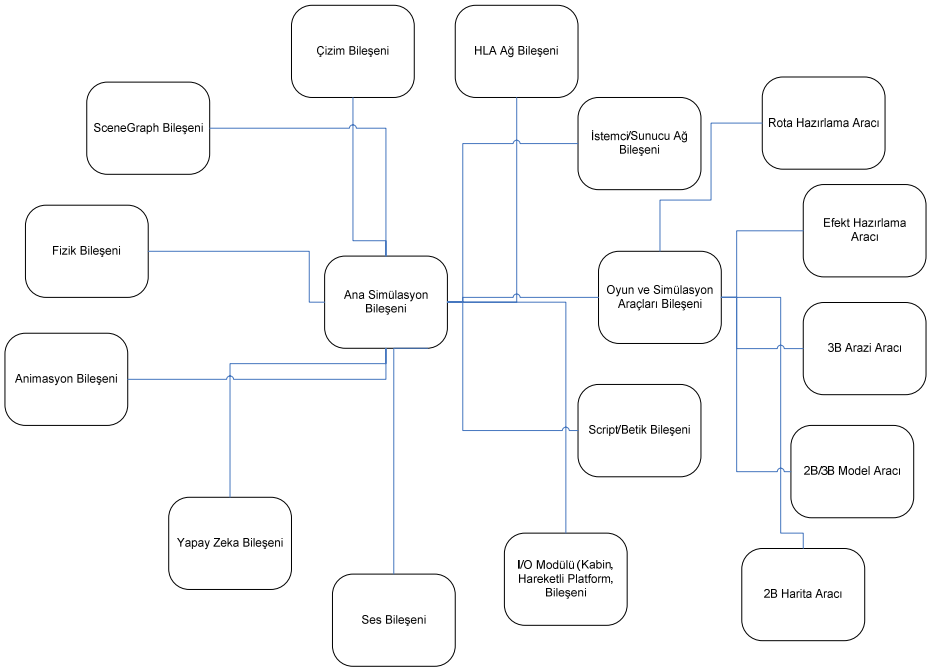
2. CİDDİ OYUN VE SİMÜLASYON ALTYAPI MİMARİSİ

Çok kullanıcılı, yoğun gerçekli ve çok platformlu ciddi oyun ve simülasyonların oluşturulmasını sağlayacak olan altyapı mimarisi, hem oyunların hem de simülasyonların ihtiyacı olan yeteneklere sahip olmalıdır. Bu çalışmada sunulan altyapı mimarisi Şekil 2'de yer almaktadır.

Oyun ve simülasyon motorları gerçek-zamanlı grafik sunan interaktif oyun ve simülasyonların temel yazılım bileşenini oluşturmaktadır [9, 10, 11, 12, 13, 16, 24]. Oyun ve simülasyonların değişik platformlar üzerinde koşmasını sağlamak da bu motorların sağladıkları önemli özelliklerden biridir. Mevcut oyun ve simülasyon motorları tarafından sunulan fonksiyonaliteler bu altyapıda yer alan ana özellikler kapsamında ele alınmaktadır. Bu amaç doğrultusunda oluşturulacak bileşenler, ana simülasyon bileşenine ek olarak oyun ve simülasyon motorlarının sahip oldukları ana fonksiyonalitelerden 2B/3B grafik çizim, "scene graph", fizik, yapay zeka, ses ve animasyon bileşenlerini içermektedir. Bu bileşenler yeniden kullanılabilir birimler olarak karşımıza çıkacaktır.

3B uygulamalarda özellikle büyük ve karmaşık dünyaları hiyerarşik olarak tanımlayan "scene graph"lar sahnedeki tüm objeleri nodlar olarak ifade eder. Bu şekilde düğüm bazlı hiyerarşik bir anlayış içeren "scene graph"lar obje erişimi ve birbirleriyle

ilişkilerinin tanımlanmasından çıkabilecek sorunları minimize eder. Ancak bu avantajların yanısıra, hafıza gereksinimleri ve erişim hızı, "scene graph" tasarımı sırasında gözönünde bulundurulması gereken en önemli nokta olacaktır. Hafıza maliyetlerini aza indirmek ve hiyerarşik erişim hızını azaltmak amacıyla değişik en uygun veri yapıları ve paternler kullanılacaktır.



Şekil 2. Altyapı mimarisi

2B/3B grafik çizim bileşeni dahilinde oyun ve simülasyon ortamındaki tüm sahnenin görsel gerçekliğe en yakın ve gerçek zamanlı olarak çizilmesi sağlanacaktır. Görsel gerçekliğe ulaşılması amacıyla temel çizim tekniklerinin yanısıra "ray tracing" ve "radiosity" gibi görsel gerçekliği çok büyük ölçüde artıran ileri seviye çizim yöntemlerinden de yararlanılacaktır [8]. Ancak görsel gerçekliğin artırılması sırasında yaşanabilecek performans kaybını minimuma indirmek için çeşitli optimizasyon yöntemleri kullanılacaktır. Ayrıca, son yıllarda donanımda yaşanan gelişmeler sayesinde, özellikle yoğun hesaplama içeren çizim ve hesaplamalarda programlanabilir grafik donanımı üzerinde çalışan gölgelendiricilerden yararlanılacaktır.

Oyun ve simülasyon uygulamalarının gerçek dünyadaki gibi olabilmesini sağlayan diğer önemli özellik de sahnedeki tüm nesnelerin birbirleriyle ve tek başına davranışlarının fizik kurallarına uygun olmasıdır. Altyapının fizik bileşeni kapsamında fiziksel kuralların mümkün olduğunca gerçeğe uygun ve gerçek zamanlılığı azaltmayacak ölçüde modellenmesi gerçeklenecektir. Kinematik, katı nesne dinamiği, kısıtlar ve

eklemler, çarpışma algılama, araç dinamiği, doğal faktörlerin dinamikleri ve deformasyonlar bu kapsamda ele alınacaktır. Fizik bileşeni içinde kullanılacak türlü teknikler performansı azaltıcı olduklarından bu teknikler konusunda çeşitli optimizasyon yöntemleri kullanılacaktır.

Sanal oyun ve simülasyon ortamındaki gerçekliliği sağlayan diğer bir etken de sestir. Ses bileşeni kapsamında ortamdaki seslerin değişik özelliklerde seslerin üç boyutlu olarak çalınması sağlanacaktır.

Animasyonlar oyun ve simülasyon uygulamaları için gerçeğe uygunluk açısından vazgeçilmez bir öğe olarak karşımıza çıkmaktadır. Özellikle oyunlarda karakterlere ait animasyonların doğallığı oyunların başarısında büyük rol oynamaktadır. Animasyon bileşeni kapsamında karakterlerin hareket animasyonları çok gerçekçi olacak şekilde gerçeklenecektir. Bu nedenle gerçekçiliği artıran ve gerçek hayatı karakter hareketleri açısından en iyi şekilde benzeten ileri seviye animasyon tekniklerinden yararlanılacaktır [21]. Animasyondaki gerçekçiliği yakalamak amacıyla geliştirilecek olan bu bileşen bu altyapıyı diğer oyun ve simülasyon motorlarından ayıran en önemli bileşen olarak karşımıza çıkacaktır.

Yapay zeka bileşeni özellikle ciddi oyun ve simülasyonlarda varlıkları akıllı kılan, gerçek yaşamdaki karar verme ve taktik geliştirme yeteneklerinin benzetilmesini sağlayan bir bileşen olacaktır. Ciddi oyun ve simülasyon altyapısının en önemli bileşenlerinden biri de yapay zeka olacaktır. Yapay zeka bileşeni kapsamında ele alınacak konular başlıca aşağıdaki gibidir:

- Hedef Takibi
- Rota Takibi
- Rota Bulma
- Hedef Seçimi
- Engellerden Sakınma
- Toplu/Grup Hareket
- Yapay Aptallık

Yapay zeka bileşeni ile sunulan taktik geliştirme yöntemleriyle plan seçme işlemi kural tabanlı olarak yapılacaktır. Oyun ve simülasyonlarda otomatik hareket edecek olan ajanların başarı/başarısızlıklarını tecrübe ettikçe öğrenmeleri sağlanabilecek ve bu yeteneğin gerçeklenmesinde "neural networks" gibi yöntemlerden yararlanılabilecektir [22, 23]. Ayrıca, ajanların başarı/başarısızlıklarını sonraki nesillere aktarabilmesi mümkün kılınabilecek ve bunun için genetik algoritmalar kullanılacaktır. Mevcut oyun ve simülasyon motorlarının sahip oldukları yetenekler arasında yapay zeka çoğu zaman eksik ve yetersiz kalmaktadır. Gelişmiş yapay zeka yeteneklerini de içecerecek olan ciddi oyun ve simülasyon altyapısı yapay zeka bileşeni yönüyle de mevcut oyun ve simülasyon motorlarından ayrılmaktadır.

Simülasyon ve oyun motorunun önemli fonksiyonlarından biri de yerel ağ ve internet üzerinden çok kullanıcılı ortama hizmet vermesidir. Simülasyon ve oyun motorlarında ağ motorları kullanımlarına göre karakteristikler gösterebilmektedir. Ağ motoru iki ana alt modüle ayrılarak gerçekleştirilecektir. Bu alt modüllerden biri simülatörlerin beraber çalışmalarını sağlayacak olan "High Level Architecture (HLA)" mimarisine uygun HLA

ağ bileşenidir. HLA mimarisi kullanılarak simülasyonun birçok bilgisayara dağıtılmasının ve kullanıcıların etkileşimlerinin düzenlenmesini sağlayacaktır. Standart olan ve simülasyon mimarilerinde çok tercih edilen bir yöntemdir [17, 18, 19, 20]. Diğer bileşen ise çok kullanıcılı oyun mimarisine uygun istemci/sunucu ağ bileşenidir.

Çok kullanıcılı oyun ve simülasyon altyapısı ağ mimarisi açısından çok ciddi zorluklar ve kısıtlar içermektedir. Binlerce oyuncunun katılımında farklı ağlar arasında senkronizasyonu sağlamak, ağ zaman protokollerinin kullanılmadığı durumlarda zamanlama mekanizmasını çalıştırmak karşılaşılacak problemlerden biridir. Çok kullanıcılı oyun motorlarında katılımın çoğalmasıyla kullanılacak sunucu sistemlerinin sayısının ihtiyaca göre artması gerekmektedir. Her bir sunucu oyunun bir kısmını veya bir kopyasını tutacak ve istemcilere gerekli servisleri sağlayacaktır. Katılımcıların artmasıyla gerekli olan hesaplama gücü de dinamik olarak değişecektir. Sunucu sistemleri arasında eşit yük dağılımını sağlamak, sunucuların karşılayabileceği limitleri belirlemek ve sunucular arasında yük dağıtım geçişlerini kullanıcıya fark ettirmeden veya minimum etkiyle gerçekleştirmek gerekmektedir. Çok kullanıcılı oyunlarda düşük bant genişliği, standart olmayan bağlantılar, yüksek paket kayıpları, yüksek gecikmeler karşılaşılacak ve çözülmesi gereken olası problemlerdendir. Ağ bileşenleri ile tüm bu sorunlara optimum çözüm sağlanması planlanmaktadır.

Çok kullanıcılı oyun motorlarında kullanıcıların ilgi alanlarının yönetiminin gerçekleştirilmesi gerekmektedir. Bu amaçla kullanıcıya tüm veriler değil sadece oyuncunun ihtiyacı olan ve ilgi alanındaki bilginin gönderilmesi gerekmektedir. Böylece gerekli olan bant genişliği minimize edilmiş olacaktır. Ayrıca, ağ performansı nedeniyle yaşanabilecek sıkıntılar için çözümler üretilmesi planlanmaktadır. Duruma göre oyundaki bir sonraki olayları tahmin ederek yeni ve güncel veriler gelmeden kullanıcıya tepkilerin çok hızlı bir şekilde verilmesini sağlayan "Dead Reckoning" mekanizmaları kullanılabilecektir.

Oyun ve simülasyon motorlarının önemli fonksiyonlarından biri de programcılara sağladıkları avantaj ve kolaylıklardır. Alt kütüphanelerin kullanımı ne kadar kolay ve etkin olursa ortaya çıkabilecek oyun tasarımında başarıyı yakalamak o kadar mümkün olacaktır. Çok kullanıcılı oyun ve simülasyon altyapısı programcıların çok kolay kullanabilecekleri kolay program edilebilirlik ve test edilebilirlik açısından önemli avantaj sağlayacak "script" dili ve "script" bileşenine sahip olacaktır. "Script" bileşeni oyun ve simülasyondaki tüm nesnelere erişimleri, nesneler arasındaki etkileşimleri, nesnelerin olaylarını, olayların tetiklenmesini, sebep sonuç ilişkilerini içerek şekilde gerçekleştirilecektir.

Çok hızlı senaryo ve oyun geliştirilmesi "script" kullanarak mümkün olmakla beraber, senaryo geliştiricilerinin çok hızlı ve görsel olarak senaryo geliştirmeleri altyapıya özgünlük ve önem katacaktır. Senaryo hazırlama aracının aşağıdaki özellikleri içermesi planlanmaktadır:

- Görsel "Script" Aracı
- 3B Arazi Aracı
- 2B Harita Aracı
- 3B Model Aracı

- 2B Model Aracı
- Rota Aracı
- Efekt Aracı

Oyun ve simülasyon altyapısının önemli fonksiyonlarından biri de Giriş/Çıkış (I/O) işlemlerini gerçekleştirecek olmasıdır. "Head-Mounted Display (HMD)" sistemleri ve hareketli platformlar senaryolara gerçekçilik hissi vermesi açısından askeri ve sivil simülasyonlarda tercih edilmektedir. I/O bileşeni sayesinde kabinler, HMD sistemleri ve hareketli platformlar ile etkileşimler gerçekleştirilecektir. Diğer oyun ve simülasyon motorlarında I/O işlemleri genellikle çeşitlendirilmemiş bulunmaktadır. I/O bileşeni bu yönüyle ciddi oyun ve simülasyon altyapısını diğer motorlardan ayıran önemli bir bileşen olarak karşımıza çıkmaktadır.

Bu çalışmanın en önemli özelliklerinden birisi de mobil erişimlerin sağlanacak olmasıdır. Altyapının çok platformlu olmasında role sahip olacak olan mobil sunucu ve istemciler bulundurulacaktır. Bu sayede mobil erişimlerle oyun ve simülasyona kısıtlı yeteneklerle de olsa katılınabilmesi sağlanabilecektir.

3. SONUÇ

Yukarıda verilen bilgilerden de görüldüğü gibi, ciddi oyunlar dünyada pek çok farklı sektörde popüler olmaya başlamıştır. Bunun temel sebeplerinden biri de ciddi oyunların değişik yaş gruplarına hitap etmesi, farklı amaçlar doğrultusunda motivasyon artırıcı ve eğitim içerikli etkiye sahip olmalarıdır. Çok önemli oyun geliştirme ve simülasyon eğitimi fuar ve konferanslarında bu konuya ağırlık verilmesi ciddi oyunların önemini ortaya çıkaran faktörlerdendir.

Bilgisayar teknolojilerinin ciddi oyunlar ve simülasyonlarda kullanılmaya başlaması çok uzun bir geçmişe sahip değildir. Donanım ve yazılım dünyasında yaşanan teknolojik gelişmeler, üst seviye çalışma ortamına ihtiyaç duyan simülasyon ve ciddi oyunların da yaygınlaşmasına ve gelişmesine ön ayak olmuştur. Simülasyon ve oyunların oluşturulması için farklı yazılım motorları geliştirilmektedir. Ancak bu motorlar özellikle yoğun kullanıcılı, yoğun gerçekli ve çok platformlu ciddi oyun ve simülasyonların çalışması sözkonusu olduğunda temel simülasyon bileşenleri dışında yapay zeka, animasyon ve ağ gibi önemli bileşenlere de ileri seviyede ihtiyaç duymaktadırlar. Bu çalışmanın konusu olan altyapı mimarisi tüm bu bileşenleri de dikkate alarak sunduğu üst seviyede çözümlerle mevcut motorlardan ayrılmaktadır. Ayrıca, yaşanacak olası problemler tespit edilmiş olup bunların çözülebilmesi için alternatif optimizasyon teknikleri verilmiştir.

Ciddi oyunlar ve simülasyonlar açısından dünyada yaşanan gelişmelerin ülkemizde de böyle bir altyapı çalışmasıyla somutlaşabilmesi ulusal açıdan büyük bir öneme sahip olacaktır. Bu altyapı sayesinde askeriye ve devlet gibi pek çok sektör için kolayca kısa zamanda geliştirilebilecek olan simülasyon ve ciddi oyunların sayısı büyük ölçüde artacaktır. Oyun ve simülasyon geliştirmede yaşanan sıkıntıların böyle bir altyapı sayesinde minimuma indirilmesi uluslararası düzeyde de önemli adımlar atılmasını sağlayacaktır.

KAYNAKÇA

[1] Brouwer-Janse, M.D., Fulton Suri, J., Coleman, R., Edwards, S., Fozard, J.L., Keyson, D.V., (1995), "From our past to our future: User interfaces over the lifespan", Companion Proceedings of CHI '95, pp.187-188.

[2] Hymel, S., Zinck, B., Ditner, E., (1993), "Cooperation versus competition in the classroom", Exceptionality Education Canada, 3, 1&2, pp.103-128.

[3] Inkpen, K., (1997), "Three Important Research Agendas for Educational Multimedia: Learning, Children, and Gender", Proceedings of Educational Multimedia'07, Calgary AB, pp.521-526.

[4] Johnson, D.W., Maruyana, G., Johnson, R., T., Nelson, D., Skon, L., (1981), "Effects of cooperative, competitive and individualistic goal structures on achievement: A meta- Analysis", Psychology Bulletin, 89(1), pp.47-62.

[5] Lucas, K., Sherry, J. L., (2004), "Sex Differences in Video Game Play: A Communication-Based Explanation", Communication Research 31(5), pp.499-523.

[6] Yilmaz, E., Cagiltay, K., (2005), "History of Digital Games in Turkey", Proceedings of DiGRA 2005 Conference.

[7] Kirriemuir, J., (2002), "Video Gaming and Digital Learning Technologies: Relevance and Opportunities", Dlib Magazine.

[8] F. S. Hill, Jr., (2000), Computer Graphics Using OpenGL, Prentice Hall, New Jersey, USA.

[9] K. Pallister, (2005), Game Programming Gems 5, Charles River Media, Rockland, Massachusetts, USA.

[10] Gamedev.net Web Sitesi, http://www.gamedev.net, Son erişim tarihi: Şubat 2007.

[11] Gamasutra – The Art & Business of Making Games Web Sitesi, http://www.gamasutra.com, Son erişim tarihi: Şubat 2007.

[12] Delta 3D Open Source Gaming & Simulation Engine Backed By the U.S. Military Web Sitesi, http://www.delta3d.org, Son erişim tarihi: Şubat 2007.

[13] Torque Game Engine Web Sitesi, http://www.garagegames.com, Son erişim tarihi: Şubat 2007.

[14] SecondLife Web Sitesi, http://secondlife.com, Son erişim tarihi: Şubat 2007.

[15] ActiveWorlds Web Sitesi, http://www.activeworlds.com, Son erişim tarihi: Şubat 2007.

[16] DevMaster's Game and Graphics Engines Database Web Sitesi, http://www.devmaster.net/engines, Son erişim tarihi: Şubat 2007.

[17] Department of Defense, (1999), "High Level Architecture RTING 1.3v2, Programmer's Guide".

[18] Simulation Interoperability Standards Organization, http://www.sisostds.org, Son erişim tarihi: Şubat 2007.

[19] HLA, Federation Development and Execution Process (FEDEP) Model, Version 1.3, (1998).

[20] IEEE 1516, High Level Architecture (HLA), IEEE, www.ieee.org, Son erişim tarihi: Şubat 2007.

[21] R. Parent, Morgan & Kaufmann, (2001), Computer Animation: Algorithms and Techniques.

[22] AI-implant Web Sitesi, http://www.biographictech.com, Son erişim tarihi: Şubat 2007.

[23] S. Russell and P. Norvig, (2002), Artificial Intelligence: A Modern Approach (2nd Edition), Prentice Hall.

[24] T. Alexander, (2003), Massively Multiplayer Game Development, Charles River Media.

[25] Interactive Software Federation of Europe (ISFE) (2004a), ISFE Yearbook 2003-2004.

B6 OTURUMU:

KARAR DESTEK AMAÇLI MODSİM-III

TEHDİT DEĞERLENDİRME VE SİLAH TAHSİSİ – BİR UYGULAMA
Taner GÜLEZ, İnci YÜKSEL, Meysun AVCI ÖZGÜN, Dilek ARSLAN, Seçil ARSLAN, Ömer KIRCA, Sedef MERAL, Sakine BATUN, Zeynep KİRKİZOĞLU, Faruk SOYKAN, Levent ACUN, Ümit KIYAK, Cumhur SOMER

DEĞİŞKEN VURMA OLASILIKLI HAVA SAVUNMA SİLAHLARININ GRUP HALİNDE HAVA SAVUNMA STRATEJİSİNİN PLANLANMASI VE OPTİMİZASYONU
Taner GÜLEZ, Levent KANDİLLER, Nur Evin ÖZDEMİREL

ATIŞ KONTROL SİSTEMİ HATA ANALİZİ İLE TEK ATIŞTA VURUŞ İHTİMALİNİN HESAPLANMASI
İnci YÜKSEL, Nur Evin ÖZDEMİREL, Levent KANDİLLER

TEHDİT DEĞERLENDİRME VE SİLAH TAHSİSİ – BİR UYGULAMA[13]

Taner GÜLEZ[13(a)], İnci YÜKSEL[(a)], Meysun AVCI ÖZGÜN[(a)], Dilek ARSLAN [(a)], Seçil ARSLAN[(a)], Ömer KIRCA[(b)], Sedef MERAL[(b)], Sakine BATUN[(b)], Zeynep KİRKİZOĞLU[(b)], Hv. Svn. Yb. Faruk SOYKAN[(c)], Hv. Svn. Yb. Levent ACUN[(c)], Hv. Svn. Yb. Ümit KIYAK[(c)], Hv. Svn. Bnb. Cumhur SOMER[(c)]

[(a)] ASELSAN A.Ş. Macunköy/Ankara,
[(b)] O.D.T.Ü., Endüstri Mühendisliği
[(c)] K.K.K. Hava Savunma Okulu ve Eğitim Merkez Komutanlığı

ÖZ

Bu bildiride hava savunması yapan bir grup silahın, taarruz için yaklaşan hedeflere en etkin şekilde eşlenmesi için geliştirilen "Tehdit Değerlendirme ve Silah Tahsisi (TDST)" algoritması tanıtılmıştır. Değişik nitelikte sensörler kullanılarak oluşturulan bir hava resminde bulunan tehditlerin, anlık tehdit seviyeleri ve silahların etkinlik değerlerini oluşturan faktörler ve bu faktörlerin ağırlıkları analitik hiyerarşi süreci (analytical hierarchy process-AHP) metodu ile belirlenmiştir. Bu metodun çıktısı, silah tahsisi ile ortaya çıkacak etkinlerin hesaplanmasında kullanılmıştır. Toplam etkinliği maksimum yapmayı amaçlayan bir matematik model oluşturulmuştur ve anlık hava resimleri için geçerli sonuçlar ürettiği görülmüştür. Anlık çözümler üreten bu model *statik model* olarak adlandırılmıştır. Modelin belirli kurallara göre işleyen bir algoritmayla birlikte kullanılması ile modele zamanla yer değiştiren hedefleri de hesaba katma ve böylece sürekli olarak çalışabilme kabiliyeti kazandırılmıştır ve bu model de *dinamik model* olarak adlandırılmıştır. Yapılan testler ile TDST algoritması doğrulanmıştır.

Anahtar Kelimeler: AHP, silah-tehdit eşlemesi, tamsayılı programlama, tehdit değerlendirme

THREAT EVALUATION AND WEAPON ASSIGNMENT – AN APPLICATION

ABSTRACT

We propose a "Threat Evaluation and Weapon Assignment (TEWA)" algorithm which is developed for allocation of defense weapons to attacking threats in a most efficient way. A snapshot of the situation is constituted by the data gathered via different types of sensors and instantaneous threat level of the threats and efficiency of the weapons on this snapshot is evaluated through a method based on analytical hierarchy process

[13] Bu bildiride ifade edilen fikirler tümüyle yazarlara ait olup TSK'yı ve bildiride adı geçen diğer kurum ve kuruluşları hiç bir surette bağlamamaktadır.

(AHP). The output of this method is used to compute the effectiveness parameters for weapon allocations. A mathematical model is formed in order to maximize the total effectiveness. It has been observed that this model (i.e. *static model*) produces valid results for the environmental picture, The ability of concerning the moving threats in time and operate continuously is obtained by using the static model in an algorithm with several rules (i.e. *dynamic model*). The algorithm has been verified by the scenario tests applied.

Keywords: AHP, integer programming, threat evaluation, weapon-target allocation

1. GİRİŞ

Bu makalede "Tehdit Değerlendirme ve (Bu Tehditlere) Silah Tahsisi (TDST)"nin bir uygulaması anlatılmaktadır. TDST problemi, kaynak tahsisi problemlerinin alt başlığı olarak literatürde yer alır ve şu şekilde ifade edilebilir: Bir muharebe ortamında, dost güçlere bağlı, daha önceden konuşlandırılmış, yer değiştirebilen fakat anlık olarak konum bilgileri mevcut olan, mühimmat seviyeleri takip edilen ve kabiliyetleri bilinen, merkezi bir komuta kontrol sistemi tarafından yönetilen silah sistemleri bulunmaktadır. Dost sensörlerden alınan bilgiler Hava Resmi Oluşturma Algoritması ile işlenmektedir. Bu algoritma ile oluşturulan hava resminde, yaklaşan tehdidin nitelikleri ve konumu elde edilmektedir. TDST uygulamasındaki amaç, bu hava resmi ve silahların oluşturduğu muharebe ortamında silah-tehdit eşleşmesinden en çok faydayı sağlamaktır.

Silah-Tehdit Eşleme literatürü kapsamında düşmana en çok zararın verilmesi, ([1], [2], [3], [4], [5], [6], [7], [8]), dost birliklere verilecek zararın en aza indirilebilmesi ([9], [10], [11], [12], [13]), en az mühimmat ile en çok faydanın sağlanabilmesi ([14]) ve tehdit-silah eşleşmesi ile ortaya çıkacak etkinliğinençoklanması ([15]) gibi örnekler bulunmaktadır.

TDST Algoritmaları, tehdit ve silah sayılarının fazla olduğu operasyonel durumlarda, karar verici kişilere bir destek mekanizması olarak yardımcı olur. Örneğin, değişik nitelikte on adet tehdit ile farklı kabiliyetlere sahip on adet silahın bulunduğu bir muharebe ortamında değerlendirilmesi gereken (10x10) 100 adet olası silah-tehdit eşleri bulunmaktadır. Özellikle, komuta kontrol sistemlerinde haberleşme (emir gönderimi ve emir kabul) ile silah hazırlanma süreleri düşünüldüğünde, saniyeler bazında silah tahsislerine karar vermek büyük önem kazanmaktadır. TDST algoritmaları ortamdaki farklı tehdit seviyelerini ve farklı kabiliyetleri düşünerek toplam etkinliği artırıcı çözümler sunduğundan, bu algoritmaların kullanımı da aynı ölçüde önem kazanmaktadır.

Geliştirilen TDST algoritması isminden de anlaşılabileceği gibi iki ana bölümden oluşmaktadır. Bunlar:

1. Tehdit Değerlendirme: Tehditlerin nitelikleri, silahların durumları ve tehditlere etki seviyeleri ve silah-tehdit eşleşmeleri de gözetilerek, olası tüm eşleşmeler için etkinlik değerlerinin bulunması
2. Silah Tahsisi: Silah-tehdit eşleşmelerinin toplam etkinliğiençoklayacak şekilde gerçekleştirilmesi

2. MATEMATİKSEL MODEL – SİLAH TAHSİSİ

Bir problem ortamında bulunan kaynakların, sahip oldukları özelliklere göre (iş yapış süresi, işi yapma kabiliyeti, maliyeti vb.) yapılması gerekli işlere atanması Kaynak Tahsisi problemlerinin temelini oluşturur. Bu kapsamda, tamsayılı matematik modeller silah tahsisi probleminde atamaların gerçekleştirilmesi için kullanılabilecek araç olmaktadır. Geliştirilen tamsayılı matematik modelde şu özellikler bulunmaktadır:

- Ana karar değişkeni 0 veya 1 tamsayı değerlerini alır. Bu, bir silahın tahsis edildiği tehdidi belirler. Bu nedenle oluşturulan model bir 0–1 tamsayılı matematik modeldir.
- Kısıtlar:
  - Her silah en çok bir tehdide tahsis edilebilir
  - Her tehdide en az bir silah tahsis edilmelidir
  - Her tehdide bir tip silahtan en çok belirli bir sayıda tahsis yapılabilir
  - Modele eklenen özel durumlara bağlı diğer kısıtlar
- Amaç, tüm silah-tehdit tahsisleri ile ortaya çıkan toplam etkinliğin ençoklanmasıdır.

Model matematiksel olarak ifade edilirse,

Karar değişkenleri:

$$x_{is} = \begin{cases} 1 & \textit{eğer s silahı i tehdidine atanmış ise} \\ 0 & \textit{diğer durumda} \end{cases} :$$

Parametreler:

e_{is} : i tehdidi ile s silahının eşlenmesi ile doğacak etkinlik

N_i : i tehdidine atanabilecek en çok silah sayısı

K_{ik} : i tehdidine atanabilecek en çok k tipi silah sayısı

$$\textit{maks} \quad \sum_{i=1}^{N} \sum_{s=1}^{M} e_{is} x_{is}$$

öyle ki

$$\sum_{i=1}^{N} x_{is} \le 1 \qquad \forall s \qquad (1)$$

$$1 \le \sum_{s=1}^{M} x_{is} \le N_i \qquad \forall i \qquad (2)$$

$$\sum_{s \in A_k} x_{is} \le K_{ik} \qquad \forall i, k \qquad (3)$$

$$x_{is} \in \{0,1\} \tag{4}$$

3. ETKİNLİK DEĞERLERİNİN OLUŞTURULMASI (TEHDİT DEĞERLENDİRME)

Bu çalışmada geliştirilen modele girdi olan herhangi bir silah tahsisinden doğacak etkinlik değerinin belirlenmesi gerekir. Literatürde yer alan silah tahsis model ve uygulamalarında tahsis etkinlik parametrelerinin bilindiği varsayılmaktadır. Bu parametrelerin belirlenmesi için çok sayıda değişik faktörün değerlendirilmesi gerekmektedir. Ancak, düşmanın taarruz stratejisi, bilinmeyen tehditlerin nitelikleri ya da nitelikleri bilinen tehditlerin ortama göre değişen vurma olasılıkları gibi önceden öngörülemeyen durumlar da tahsis modellerinin kısıtlı kaldığı durumlardır. Bu nedenle, düşman tehdit seviyesinin belirlenmesinde tehditlerin temel özellikleri ve elimizdeki savunma araçları ile olası eşleşmesi halinde elde edilecek faydanın üzerinde durulması yararlı olacaktır. Tehditlerin temel özelliklerinin oluşturduğu tehdit miktarları konunun uzmanlarınca değerlendirilerek, başarı olasılığı yaklaşık olarak hesaplanabilir.

AHP (Analytical Hierarchy Process) bir konu üzerinde farklı boyutlarda uzman bilgilerinin ortaya çıkarılması için kullanışlı bir metottur. AHP yöntemi Saaty tarafından önerilmiş ve yaygın olarak kullanılmaktadır [16]. Tahsis etkinlik parametrelerinin belirlenmesi için esas olarak

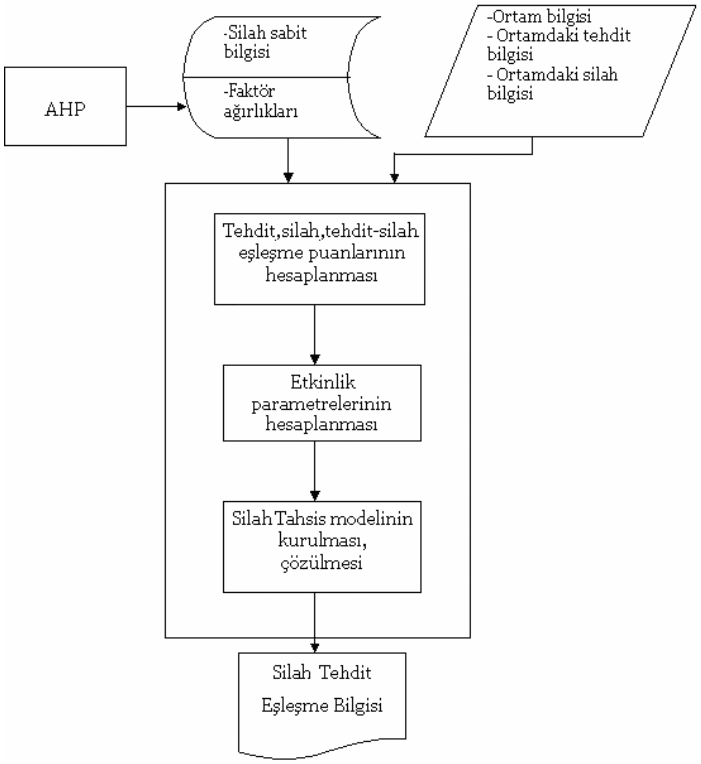
- Tehdide
- Silaha
- Silah - Tehdit eşleşmesine

bağlı faktörler kullanılmıştır.

Bu faktörler ve ağırlıklarının belirlenmesi için AHP yöntemi uygulanmıştır. Uygulanan AHP yöntemi ile hava savunma uzmanlarının bilgileri değerlendirilerek, faktörlerin ağırlık değerleri belirlenmiştir. Herhangi bir anda kullanılacak etkinlik değerlerinin belirlenmesi için faktör puanlarının hesaplandığı bir yöntem geliştirilmiştir.

4. UYGULAMA (SİLAH TAHSİSİ)

Geliştirilen tahsis modelinin ve AHP yöntemi ile hesaplanan etkinlik parametrelerinin kullanılması ile toplam etkinliğin ençoklanması süreci akış diyagramı olarak Şekil 1'de verilmiştir.



Şekil 1. TDST algoritması süreci

Zaman boyutu olarak genelde silah tahsisi problemi, anlıktır. Başka bir deyişle, bir anlık hava resmi üzerinden en uygun eşleşmeleri yapmaya çalışır. Tek bir tehdit-silah kümesi üzerinde bir anlık çözüm üreten modeller *statik model* olarak adlandırılır. Mevcut mühimmatı ve taarruz senaryolarını iki bölümde inceleyen çalışmalar da vardır. Bu çalışmalarda zaman boyutu ayrıca ele alınıp bir çizelgeleme yapılmaktadır. Zamana bağlı çözüm üretebilen tehdit-silah eşleme modelleri dinamik model olarak adlandırılır. Önerilen TDST algoritması, geliştirilen statik bir silah-tehdit eşleme modelini

- Tehdidin zaman içindeki konumunu tahmin etmek ve bu tahmine göre etkinlik değerlerini hesaplamak
- Savunulan bölgeye yaklaşan tehdidi hızlı bir şekilde saptamak ve silah tahsisi önerisinde bulunmak
- Silah tahsislerinin sık değişimlerini önlemek
- Gerektiğinde daha tehlikeli tehditler üzerinde ateş yoğunluğu oluşturmak

gibi belirli kurallar dâhilinde bir algoritma içinde kullanılarak dinamik ortamda çözüm üretmektedir. Diğer bir deyişle, bir anlık hava resminden diğer bir hava resmine geçişlerin muharebe durumunun önemi ve bu durumun belirlediği kurallar çerçevesinde yapılması ile algoritmaya dinamik çözüm kabiliyeti kazandırılmıştır. Bu çalışma kapsamında geliştirilen TDST modeli denenmiş ve geçerliliği kanıtlanmıştır.

5. SONUÇ

TDST algoritmalarının kara, hava ve deniz platformlarında uygulamaları bulunmaktadır. Genel işleyiş olarak TDST algoritmaları birinci basamağında, ortamdaki sensörlerden alınan bilgiler doğrultusunda oluşturulan ortam resminde, tehdit tipi, hızı, yönü, uzaklığı, büyüklüğü gibi parametreleri kullanarak düşmanın tehdit seviyesini belirler. Bu basamak, algoritmanın "Tehdit Değerlendirme" basamağıdır. İkinci basamak olan silah tahsisi ise, belirlenen etkinlik değerleri kullanılarak en etkin silah tahsisini gerçekleştirmektir.

Kullanılacak model seçiminde, modelin hızı ve çözüm kalitesi önemli kriterlerdir. Kullanılan modelin çıktılarının, taarruz karakteristiklerine karşı yürütülen taktiğe göre uygun sonuçlar vermesi için kullanılan puanlandırma kriterlerinin ve algoritma kurallarının iyi seçilmesi gerekmektedir. Puanlandırmalar sırasında anket sorularının bir grup içinde farklı bakış açıları ile değerlendirilebileceği ve bu nedenle cevapların grup için detaylı değerlendirilmesinin faydalı olacağı öğrenilen dersler arasındadır. Statik model için en iyi olan bir çözüm dinamik model için her zaman en iyi olmayabilir. Böyle durumlarda model, anlık ortam resmini en iyi şekilde çözmekten ödün vermiş gibi görünebilir ancak birkaç ortam resmi arka arkaya düşünüldüğünde tüm taarruz için doğru sonuçlar üretilmiş olduğu görülebilir.

6. TEŞEKKÜR

Bu makalenin hazırlanmasında emeği geçen K.K.K. Hava Savunma Okulu ve Eğitim Merkez Komutanlığı'na çok teşekkür ederiz.

KAYNAKÇA

[1] Lloyd, S. P. and Witsenhausen, H. S.,(1986), "*Weapons Allocation is NP-Complete*", Proceedings of the Summer Computer Simulation Conference, Reno, Nevada,

[2] Hosein P.A., Walton J.T., Athans M., (1988), "*Dynamic Weapon-Target Assignment Problems with Vulnerable C2 Nodes*", Laboratory for Information and Decision Systems, Report LIDS - P .1786

[3] R.K. Ahuja, A. Kumar, J. Krishna, J.B. Orlin, (2003), "*Exact and Heuristic Methods for the Weapon Target Assignment Problem*", MIT Sloan School of Management Working Paper 4464-03

[4] Manne A.S. (1958), "*A Target Assignment Problem*", Operations Research, Vol. 6, No.3. , pp.346 - 351

[5] Hosein P. A., Athans M., (1990), "*Some Analytical Results for the Dynamic Weapon-Target Allocation Problem*", Laboratory for Information and Decision Systems, Report LIDS-P-1944

[6] Day R.H., (1966), "*Allocating Weapons to Target Complexes By Means Of Nonlinear Programming*", Operations Research, Vol. 14, No. 6., pp. 992 – 1013

[7] Passy U., (1971), "*Nonlinear Assignment Problems Treated by Geometric Programming*", Operations Research, Vol. 19, No. 7, pp. 1675-1690.

[8] Den Broeder. G. G., Ellison, R. E., and Emerling, L., (1959), On Optimum Target Assignments, Operations Research, Vol. 7, pp. 322-326

[9] Hosein P. A., Athans M., (1990), "*An Asymptotic Result for the Multi-Stage Weapon-Target Allocation Problem, Part I*" Proceedings of the 29th Conference on Decision and Control, Honolulu, HW

[12] Hosein P. A., Athans M., (1990), "*Preferential Defense Strategies. Part II: The Dynamic Case*", MIT Laboratory for Information and Decision Systems, Report LIDS-P-2003,

[11] Lee J.Z., Lee C.Y., Su S.F., (2002), "*An Immunity-Based Ant Colony Optimization Algorithm for Solving Weapon–Target Assignment Problem*", Applied Soft Computing, Vol. 2, pp 39-47

[12] Lemus, F., David, K. H., (1963), "*An Optimum Allocation of Different Weapons to A Target Complex*", Operations Research, Vol. 11, No. 5. pp. 787 - 794

[13] W. P. Malcolm, (2004), "*On the Character and Complexity of Certain Defensive Resource Allocation Problems*", Weapons Systems Division Systems Sciences Laboratory, DSTO-TR-1570 [Public Release Permitted Book Section]

[14] Kwon et al. (1999), "*Lagrangian Relaxation Approach to the Targeting Problem*", Naval Research Logistics, Vol. 46, pp. 640–653,

[15] Rosenberger J. M., Yücel A., (2005), "*The Generalized Weapon Target Assignment Problem*", 10th International Command and Control Research and Technology Symposium, the Future of C2, McLean, VA

[16] Saaty, T. L., (1980), "*The Analytic Hierarchy Process*", McGraw-Hill, New York

DEĞİŞKEN VURMA OLASILIKLI HAVA SAVUNMA SİLAHLARININ GRUP HALİNDE HAVA SAVUNMA STRATEJİSİNİN PLANLANMASI VE OPTİMİZASYONU

Taner GÜLEZ [(a)], Levent KANDİLLER [(b)], Nur Evin ÖZDEMİREL [(c)],

[(a)] ASELSAN A.Ş. Macunköy/Ankara, tgulez@aselsan.com.tr
[(b)] O.D.T.Ü., Endüstri Mühendisliği, kandil@cankaya.edu.tr
[(c)] O.D.T.Ü., Endüstri Mühendisliği, nurevin@ie.metu.edu.tr

ÖZ

Bu bildiride bir ya da birden fazla değerli merkezi yapıyı savunan, değişken vurma olasılıklarına sahip bir grup hava savunma silahının, yaklaşan hava ataklarına karşı savunma stratejisi incelenmiştir. Silahların hedefleri vurma olasılıkları atış kontrol sistemlerinin hata analizi ile elde edilmiş tek atışta vurma ihtimalleri (TAVİ) olarak düşünülmüştür. Bu olasılıklar kullanılarak silahların atış zamanlarını çizelgeleyen bir karışık tamsayılı doğrusal olmayan matematiksel model oluşturulmuştur. Bu modelin içerdiği doğrusal olmayan üç farklı ifadeye doğrusal yaklaşımlar geliştirilmiştir. Bu yaklaşımlar eşliğinde oluşturulan yeni matematiksel model ile doğrusal olmayan ifadeler içeren model çözülmüştür. Sonuç olarak doğrusallaştırılan modelin zaman ve çözüm performansları doğrulanmıştır.

Anahtar Kelimeler: Değişken Vurma Olasılığı, Doğrusallaştırma, Hava Savunma, Optimizasyon, Silah – Tehdit Eşlemesi

PLANNING AND OPTIMIZATION OF AIR DEFENSE STRATEGY FOR A GROUP OF AIR DEFENSE WEAPONS WITH VARYING HIT PROBABILITIES

ABSTRACT

In this paper, we investigate defense strategy of a group of air defense weapons with varying probabilities against air threats aiming one or more high-valued assets. Hit probabilities of the weapons are taken as Single Shot Hit Probabilities (SSHP) of weapons derived from Error Analysis of Fire Control Systems. A mixed nonlinear integer mathematical model has been constructed by using these probabilities. Linearization techniques are developed for three nonlinearities present in this mathematical model. The final model formulation is verified in terms of both solution and time performance.

Keywords: Air Defense, Linearization, Optimization, Varying Hit Probabilities, Weapon Target Assignment

1. GİRİŞ

Silah – Tehdit Eşleme algoritmaları kaynak tahsisi problemlerinin alt başlığı olarak literatürde yer alır. Problem, silahların ve tehditlerin başarı olasılıkları ile ifade edilir ve bu nedenle bir silah-tehdit eşleme problemi için kurulan matematiksel model genelde doğrusal değildir. Silah-Tehdit Eşleme literatürü kapsamında düşmana verilebilecek zararınençoklanması, alınan zararın enazlanması, kullanılacak mühimmatın enazlanması ve tehdit-silah eşleşmesi ile ortaya çıkacak etkinliğinençoklanması gibi örnekler, bir anlık ve birden fazla safha içeren şekillerde incelenmiştir. Bu bildiride muharebe ortamında, daha önceden konuşlandırılmış merkezi bir komuta kontrol sistemi tarafından yönetilen ve yüksek öneme sahip merkezleri koruyan bir ya da birden fazla silah sisteminin saldırı amacı ile yaklaşan düşmana eşlenmesi problemi incelenmiştir. Bildiride, doğrusal olmayan bir eşleme modelinin doğrusallaştırılmasını ve doğrusal modelin çözümler ile geçerliliğinin gösterilmesi anlatılmaktadır.

2. PROBLEM TANIMI

Komuta kontrol merkezinden yönetilen ve mobil olmayan bir grup hava savunma silahı, stratejik olarak yüksek öneme sahip merkezlere saldırı amaçlı yaklaşan tehditlere karşı savunma yapmaktadırlar. Her silahın kendine özgü sabit bir atışa hazır hale gelme süresi ve kullandığı mühimmat tipi bulunmaktadır. Silahların hedefi vurma olasılıkları, atış kontrol sistemlerinin hata analizi ile elde edilmiş tek atışta vurma ihtimali (TAVİ) olarak kabul edilebilir. Bu olasılık değeri, düşmanın büyüklüğü, yönü, hızı, kullanılan mühimmat tipi, ortam sıcaklığı gibi birçok faktör kullanılarak hesaplandığından sabit bir parametre değildir, değişken bir yapıdadır.

Modelleme süreci boyunca yapılan varsayımlar aşağıda belirtilmiştir.

- Her merkezin kendine özel bir değerliliği vardır.
- Tehditlerin saldırı yönleri ve hızları sabittir.
- Bir saldırı senaryosunda bir tehdit birden fazla merkeze saldırabilir.
- Tehditler genel olarak uçaklardır ancak bu tehditlerden merkezlere fırlatılan bir mühimmat da gerekli olduğu takdirde bir tehdit olarak modele katılabilir.
- Tehditlerin merkezleri vurma olasılık saldırı boyunca sabittir.
- Silahların belirli maksimum ve minimum saldırı menzilleri vardır.
- Silahlar herhangi bir merkezi özellikle korumamaktadırlar. Diğer bir deyişle, alan savunması yapılmaktadır.
- Gerektiği takdirde silahlarda bir merkez olarak tanımlanıp, modele katılabilir.
- Dost radarlardan gelen tehditlerle ilgili bilgiler tamamen doğrudur.
- Savunma taktiği olarak At-Bak-At (Shoot-Look-Shoot) kullanılmıştır. Bu taktikte bir tehdit üzerine yapılan iki angajman üst üste gelmemeli, bir angajman sonrasında tehdide gönderilen mühimmatın tehdidi vurmadığı gözlenirse başka bir mühimmat angaje edilebilir.

Geliştirilen yaklaşımın amacı saldırı sonrasında önemli merkezlerin en yüksek olasılıkla hayatta kalmalarını sağlamaktır. Modellemenin çıktısı olarak eşleşmesi gereken silah-tehdit çiftleri ve silahların atış zamanları beklenmektedir.

3. FORMÜLASYON

Ortamda J tane, $j = 1,\ldots,J$ şeklinde endekslenmiş silah bulunmaktadır. Ayrıca, $i = 1,\ldots,I$ tehditleri; $k = 1,\ldots K$ ise saldırıya geçilen merkezleri işaret etmektedir.

w_k : k merkezinin önemi

q_{ik} : i tehdidinin k merkezini vurma olasılığı

I_k : k merkezini hedefleyen tehditler kümesi

s_i : i tehdidine, angajman yapılabilecek süre içerisinde atılabilecek maksimum mühimmat sayısı. Atılan her mühimmat r endeksi ile belirtilmiştir ($r = 1, 2, ..., s_i$).

d_j : j silahında bulunan maksimum mühimmat sayısı

p_{jir} : j silahından i tehdidine atılan r'ıncı mühimmatın vurma olasılığı

$\left[Q_{ji}, R_{ji}\right]$: i tehdidinin j silahına eşlenebileceği zaman periyodu

Δ_j : j silahının atış emrini kabul ettikten sonra atışa hazır hale gelme süresi

M : j silahının atış emrini kabul ettikten sonra atışa hazır hale gelme süresi

Karar Değişkenleri:

$$x_{jir} = \begin{cases} 1, & \text{eğer } j \text{ silahı } i \text{ tehditine } r.\text{atış için angaje edilmiş ise} \\ 0, & \text{diğer durumda} \end{cases}$$

t_{ir} : At-Bak-At taktiğine göre i tehdidinin üzerindeki $r.$ atışın başlama zamanı.

Δ_{jir} : j silahından i tehdidi üzerine atılan $r.$ mühimmatın, tehdide varmasına kadar geçen süre. Bu endekslemeler ışığında, problemin matematiksel gösterimi şu şekilde oluşur.

$$\text{maks} \sum_{k=1}^{K} w_k \prod_{i \in I_k} \left(1 - q_{ik} \left[\prod_{j=1}^{J} \prod_{r=1}^{R} \left(1 - p_{jir}\right)^{x_{jir}} \right] \right) \qquad (1.1)$$

öyle ki

$$\sum_{i=1}^{I} \sum_{r=1}^{s_i} x_{jir} \le d_j \qquad \forall j \qquad (1.2)$$

$$\sum_{i=1}^{J} x_{ji1} \le 1 \qquad \forall i \qquad (1.3)$$

$$\sum_{j=1}^{J} x_{jir+1} \le \sum_{j=1}^{J} x_{jir} \qquad \forall i, r = 1, ..., s_i - 1 \qquad (1.4)$$

$$M\left(1-\sum_{j=1}^{J} x_{ji,r+1}\right)+t_{i,r+1}\geq t_{ir}+\sum_{j=1}^{J} x_{jir}\Delta_{jir} \quad \forall i,\ r=1,...,s_i-1 \tag{1.5}$$

$$\sum_{j=1}^{J} x_{jir}Q_{ji}\leq t_{ir}\leq \sum_{j=1}^{J} x_{jir}\left(R_{ji}-\Delta_{jir}\right) \qquad \forall i,\ r=1,...,s_i \tag{1.6}$$

$$t_{ir}\geq 0 \qquad \forall i,\forall r=1,...,s_i \tag{1.7}$$

$$x_{jir}\in\{0,1\} \qquad \forall j,\forall i,\forall r=1,...,s_i \tag{1.8}$$

Kısıt (1.1), saldırı sonunda değerli merkezlerin en yüksek olasılık ile hayatta kalmasını amaçlamaktadır. Bu amacı gerçekleştirirken merkezleri, göreli değerlerine göre ağırlandırmaktadır. Kısıt (1.2) herhangi bir i tehdidine karşı yapılabilecek maksimum angajman sayısını gösterir. Kısıt (1.3), bir tehdit üzerine ilk önce birinci sıradaki angajmanın yapılması gerektiğini, kısıt (1.4) ise bir tehdit üzerindeki angajmanların sıra ile yapılması gerektiğini belirtir. Eğer bir tehdit üzerine r. angajman yapılmış ise, $(r+1)$. angajmanın yapılabilmesi için en az Δ_{jir} kadar bir sürenin geçmesi gerektiği kısıt (1.5) ile modellenmiştir. En son olarak ise, kısıt (1.6) herhangi bir atışın, angajman yapılabilir aralıklarda başlamasını ve bitmesini belirtir. Matematik modelde yer alan kısıtlar, Orhan Karasakal'ın doktora tezinden alınmıştır[1]. Bu modelde yer alan üç farklı doğrusal olmayan ifade bulunmaktadır. Bu noktalar:

1. Δ_{jir} ve x_{jir} terimlerinin çarpımı
2. Amaç fonksiyonunda bulunan değişken çarpımları
3. Değişken TAVİ değerleri

olarak tanımlanabilir. Bu kısımların doğrusallaştırılması ve modele katılması ilerleyen bölümlerde sunulmuştur.

3.1. Δ_{jir} ve x_{jir} Terimlerinin Çarpımı

(1.5) ve (1.6) numaralı kısıtlarda Δ_{jir} ve x_{jir} değişkenlerinin çarpımı modelin doğrusal olarak ifade edilmesini engellemektedir. Bu ifadelerin doğrusallaştırılabilmesi için ek tanımlamalar yapılabilir. D_{ji}, i tehdidinin ilk belirlendiği anda j silahına olan uzaklığı, v_i, i tehdidinin hızını ve v_j, j silahından atılan mühimmatın hızını belirtsin. Tehditlerin atak yönü doğrusal ve hızları sabit olduğundan dolayı, Δ_{jir} terimi Δ_j, D_{ji}, v_i ve v_j terimleri cinsinden ifade edilebilir.

$$\Delta_{jir}=\frac{D_{ji}-v_i\left(\Delta_j+t_{ir}\right)}{v_j+v_i}+\Delta_j$$

Böylece Δ_{jir} terimini $\sum_{j=1}^{J} x_{jir}\Delta_{jir}$ ifadesinde yerine koyarsak:

$$\sum_{j=1}^{J} x_{jir}\Delta_{jir} = \sum_{j=1}^{J} x_{jir}\left(\frac{D_{ji} - v_i\left(\Delta_j + t_{ir}\right)}{v_j + v_i} + \Delta_j\right)$$

$$= \sum_{j=1}^{J}\left(\frac{D_{ji} - v_i\Delta_j}{v_j + v_i}\right)x_{jir} - \sum_{j=1}^{J}\left(\frac{v_i}{v_j + v_i}\right)x_{jir}t_{ir} + \sum_{j=1}^{J}\Delta_j x_{jir} \quad \forall i, \forall r = 1,\ldots,s_i$$

Burada dikkat edilmesi gereken önemli nokta, Δ_{jir} hesaplaması, gerçek durumun oldukça basitleştirilmiş bir halidir. Hesaplamada havada kafa kafaya çarpışma varsayılmıştır. Gerçek durumda her zaman doğru olmayan bu varsayım için açılar ile ilave hesaplamalar yapılabilir.

Verilen yeni ifadede x_{jir} ve t_{ir} değişkenlerinin çarpımı nedeni ile doğrusallık yakalanamaz. Bu sorunun çözümü için $y_{jir} = x_{jir}t_{ir}$ şeklinde yeni bir değişken tanımlanabilir. Eğer bu değişkenin $x_{jir} = 0$ olduğunda $y_{jir} = 0$, $x_{jir} = 1$ durumunda da $y_{jir} = t_{ir}$ olması sağlanırsa, doğrusallaştırma yapılabilir. Kısaca

$$\beta_{ji} = \frac{D_{ji} - v_i\Delta_j}{v_j + v_i} \text{ ve } \gamma_{ji} = \frac{v_i}{v_j + v_i}$$

olarak tanımlarsak kısıt (1.5) ve kısıt (1.6)

$$M\left(1 - \sum_{j=1}^{J} x_{ji,r+1}\right) + t_{i,r+1} \geq t_{ir} + \sum_{j=1}^{J}\beta_{ji}x_{jir} - \sum_{j=1}^{J}\gamma_{ji}y_{jir} + \sum_{j=1}^{J}\Delta_j x_{jir} \quad \forall i, r = 1,\ldots,s_i - 1 \quad (1.9)$$

$$\sum_{j=1}^{J} x_{jir}Q_{ji} \leq t_{ir} \leq \sum_{j=1}^{J} x_{jir}R_{ji} - \sum_{j=1}^{J}\beta_{ji}x_{jir} + \sum_{j=1}^{J}\gamma_{ji}y_{jir} - \sum_{j=1}^{J}\Delta_j x_{jir} \quad \forall i, r = 1,\ldots,s_i - 1 \quad (1.10)$$

Bir sonraki işlem, y_{jir} değişkenin uygun bir şekilde matematik modele eklenmesidir. Bu sebeple, ilave kısıtlar modele eklenmelidir.

$$y_{jir} \leq Mx_{jir} \quad \forall j, \forall i, r = 1,\ldots,s_i \quad (1.11)$$

$$y_{jir} \leq t_{ir} \quad \forall j, \forall i, r = 1,\ldots,s_i \quad (1.12)$$

$$y_{jir} \geq 0 \quad \forall j, \forall i, r = 1,\ldots,s_i \quad (1.13)$$

y_{jir} değişkeninin etkin kullanımı, bu değişkenin $x_{jir} = 1$ olduğu durumlarda t_{jir} değişkenine eşit olmasına bağlıdır. Kısıt (1.12), bir küçük veya eşit kısıtıdır ve bu nedenle y_{jir} değişkeni her zaman t_{jir} değişkenine eşit olmayabilir. Bu eşitliğin garanti edilmesi amacı ile amaç fonksiyonunda y_{jir} değişkenine sonucu etkilemeyecek kadar küçük bir ağırlık verilmesi gerekir. Bu ağırlığa ε diyebiliriz. Böylece amaç fonksiyonu aşağıda belirtilen ifadeye dönüşür:

$$maks \sum_{k=1}^{K} w_k \prod_{i \in I_k} \left(1 - q_{ik} \left[\prod_{j=1}^{J} \prod_{r=1}^{s_i} \left(1 - p_{jir} \right)^{x_{jir}} \right] \right) + \sum_{j=1}^{J} \sum_{i=1}^{I} \sum_{r=1}^{s_i} \varepsilon y_{jir}$$

3.2. Amaç Fonksiyonunda Bulunan Değişken Çarpımları

Amaç fonksiyonundaki terimlerin çarpımı sonucunda da doğrusallığın kaybolması söz konusudur. Benzer bir amaç fonksiyonu Metler ve Preston tarafından da ele alınmıştır [2]. Aşağıda kısıt (1.15)'de görüldüğü gibi bir a_{ik} değişkeni modele eklenirse, amaç fonksiyonu da (1.14) gibi ifade edilebilir.

$$\max \sum_{k=1}^{K} w_k \prod_{i \in I_k} a_{ik} \tag{1.14}$$

öyle ki

$$a_{ik} \le 1 - q_{ik} \left[\prod_{j=1}^{J} \prod_{r=1}^{s_i} \left(1 - p_{jir} \right)^{x_{jir}} \right] \qquad \forall k, i \in I_k \tag{1.15}$$

Tek başına bu değişiklik modelin doğrusallaşmasını sağlamaz, kısıtlar ve amaç fonksiyonu hala doğrusal değildir. Bu noktada modelin doğrusallaştırılması için amaç fonksiyonunun anlamını etkileyebilecek bir değişiklik yapılması gerekir. a_{ik} değişkenlerini ortak bir a_k değeri ile değiştirirsek, $a_k = \min_{i \in I_k} \{ a_{ik} \}$, (1.14) ve (1.15) aşağıda verilen hale gelir.

$$maks \sum_{k=1}^{K} w_k \prod_{i \in I_k} a_k = maks \sum_{k=1}^{K} w_k a_k^{|I_k|} \tag{1.16}$$

öyle ki

$$a_k \le a_{ik} \le 1 - q_{ik} \left[\prod_{j=1}^{J} \prod_{r=1}^{s_i} \left(1 - p_{jir} \right)^{x_{jir}} \right] \quad \forall k, i \in I_k \tag{1.17}$$

Dikkat çekilmesi gereken nokta, a_k alt sınırını kullanarak amaç fonksiyonun anlamının "*ağırlıklandırılmış merkezlerin en az hayatta kalma olasılığınınençoklanmasıdır*". Asıl amaç fonksiyonu ile burada sunulan amaç fonksiyonları anlam bakımından aynı değildir. Metler ve Preston [1], araştırmalarında tehdidin merkezleri vurma olasılığını $q_{ik} = 1$ kabul etmiştir. Bu nedenle eğer bir tehdit savunma katmanlarından kaçabilirse, kesin olarak hedefini vuracaktır. Sonuç olarak $J \ge I$ olduğu durumlarda, kaçan tehdit olmaması olasılığı sıfırdan büyük olması için her tehdide en az bir silah atanması gerekmektedir. Asıl amaç fonksiyonumuzda tehditlerin merkezleri vurma olasılığı

$0 < q_{ik} \le 1$ olduğundan her tehdide en az bir silah verilmediği durumlarda da kaçan tehdit olmaması olasılığı sıfırdan büyük olabilir.

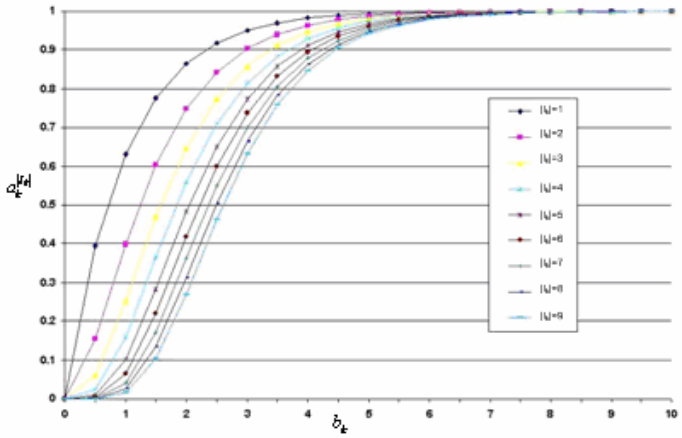
Amaç fonksiyonunun anlamının değişmesine rağmen problemin çözümü için uygulamaya devam etmek, problemin çözüm şekli için bazı fikirler verebilir. (1.17)'de her iki tarafın da doğal logaritması alınırsa kısıt (1.18) haline gelir.

$$\ln\left(1-a_k\right) \ge \ln\left(1-a_{ik}\right) \ge \ln\left(q_{ik}\right) + \sum_{j=1}^{J}\sum_{r=1}^{s_i} x_{jir} \ln\left(1-p_{jir}\right) \qquad \forall k,i \in I_k \quad (1.18)$$

Bu noktada $\ln\left(1-a_k\right) = -b_k$ tanımlamasını yapabiliriz ve $a_k^{|I_k|}$ şu şekilde ifade edilebilir.

$$a_k^{|I_k|} = \left(1-e^{-b_k}\right)^{|I_k|}$$

Verilen b_k terimi bir değişken değildir, (1.16)'ya yapılacak yaklaşım için kullanılacaktır. $|I_k|$ parametresinin farklı değerleri için çizilen $a_k^{|I_k|}$ grafiği Şekil 1'de verilmiştir. Bu parametrenin herhangi bir değeri için Şekil 1'de görülen eğrilere parçalı doğrusal yaklaşım yapılabilir. $|I_k| = 5$ için bir yaklaşım örneği Şekil 2'de verilmiştir.



Şekil 1. Farklı $|I_k|$ değerleri için $a_k^{|I_k|}$ grafikleri

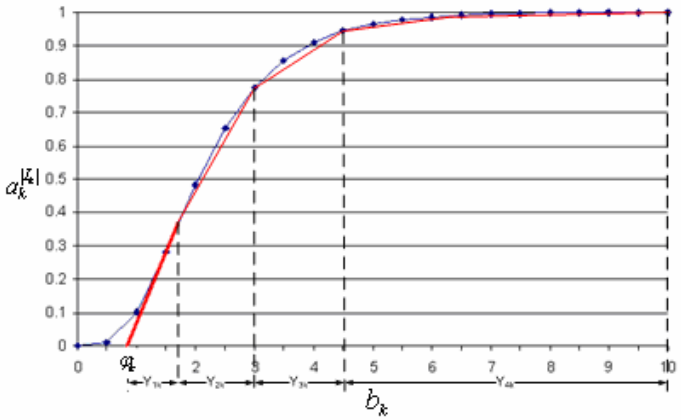
Şekil 2'de doğru parçalarının b_k eksenindeki büyüklüğü için $Y_{\alpha k}$ değişkeni (α: segment endeksi), segmentlerin eğimi için $c_{\alpha k}$ parametresi tanımlanmıştır. Böylece matematik modele (1.20) eklenerek amaç fonksiyonu (1.19) gibi yazılabilir.

$$maks \sum_{k=1}^{K} w_k \sum_{\alpha=1}^{\mathrm{A}} C_{\alpha k} Y_{\alpha k} + \sum_{j=1}^{J}\sum_{i=1}^{I}\sum_{r=1}^{s_i} \varepsilon y_{jir} \qquad (1.19)$$

öyle ki

$$\sum_{j=1}^{J}\sum_{r=1}^{s_i} x_{jir}\ln\left(1-p_{jir}\right)+\ln\left(q_{ik}\right)+\sum_{\alpha=1}^{A} Y_{\alpha k}+\varphi_k \le 0 \qquad \forall k,\forall i \qquad (1.20)$$

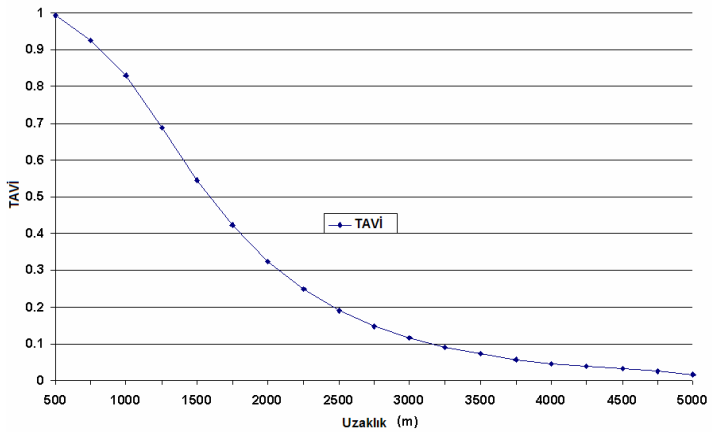
Kısıt (1.20)'ye eklenen φ_k ifadesi, Şekil 2'de de görülebileceği gibi, parçalı doğrusal yaklaşımın başladığı noktayı göstermektedir. Sonuç olarak amaç fonksiyonunun doğrusallaştırılması, anlamından fedakârlık edilerek gerçekleştirilebilir.



Şekil 2. $|I_k|=5$ için $a_k^{|I_k|}$ grafiğine yapılmış parçalı doğrusal yaklaşım

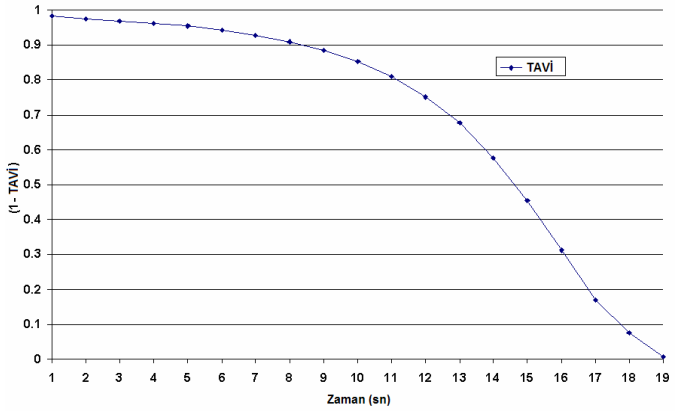
3.3.Değişken TAVİ Değerleri

Doğrusallığın kaybolmasına neden olan son nokta ise değişken TAVİ değerleridir. TAVİ değerleri atış kontrol sistemlerinde hata analizi ile hesaplanmıştır. Her bir silahın yaklaşan hedeflere karşı farklı TAVİ değerleri olacaktır. Sabit bir silahtan, silaha doğru gelen hareketli bir hedefe atılan mühimmatın, hedefin uzaklığına göre değişen vurma olasılığının değişimi örnek olarak Şekil 3'de görülmektedir.



Şekil 3. Uzaklığa bağlı TAVİ değerlerinin değişimi

(1.20)'de verildiği gibi TAVİ değerleri doğal logaritma içinde bulunmaktadır ve hesaba katılması gereken ifade $(1-p_{jir})$ değeridir. Tehdidin hızının ve yönünün değişmediği varsayıldığında, 250 m/s hızı ile yaklaşan bir tehdit için zamana bağlı $(1-p_{jir})$ grafiği Şekil 4'te bulunmaktadır.



Şekil 4. Zamana bağlı değişen $(1-p_{jir})$ grafiği

Örnek olarak verilen $(1-p_{jir})$ grafiği üzerine eğri uydurma çalışmaları yapılmış ancak kapalı formları en iyi sonuç veren eğrilerin kısıt (1.20)'i doğrusallaştıramadığı görülmüştür. Orta noktalarda hatası büyük olan bir yaklaşım olsa da, $[(1-p_{ji})/p_{ji}^{\psi_{ji}t_{jir}}]$ ifadesi uyarlanabilecek parametre değerleri ile $(1-p_{jir})$ için doğrusallaştırmaya uygun

bir kapalı formdur. p_{ji} ve ψ_{ji} parametreleri $(1-p_{jir})$ eğrisinin uç noktalarına göre seçilebilecek bir uygun değerleri belirtmektedir. Bu ifadeyi (1.20)'de kullanırsak yeni doğrusallaştırılmış kısıtımız şu hale gelir:

$$\ln(q_{ik})+\sum_{j=1}^{J}\sum_{r=1}^{s_i}x_{jir}\ln(1-p_{ji})-\sum_{j=1}^{J}\sum_{r=1}^{s_i}\psi_{ji}\ln(p_{ji})y_{jir}+\sum_{\alpha=1}^{\mathrm{A}}Y_{\alpha k}+\varphi_k\le 0 \quad \forall k,\ \forall i$$

Son olarak doğrusallaştırılmış matematik modelimizi yazarsak

$$maks\sum_{k=1}^{K}w_k\sum_{\alpha=1}^{\mathrm{A}}C_{\alpha k}Y_{\alpha k}+\sum_{j=1}^{J}\sum_{i=1}^{I}\sum_{r=1}^{s_i}\varepsilon y_{jir}$$

öyle ki

$$\ln(q_{ik})+\sum_{j=1}^{J}\sum_{r=1}^{s_i}x_{jir}\ln(1-p_{ji})-\sum_{j=1}^{J}\sum_{r=1}^{s_i}\psi_{ji}\ln(p_{ji})y_{jir}+\sum_{\alpha=1}^{\mathrm{A}}Y_{\alpha k}+\varphi_k\le 0 \qquad \forall k,\ \forall i$$

$$\sum_{i=1}^{I}\sum_{r=1}^{s_i}x_{jir}\le d_j \qquad \forall j$$

$$\sum_{j=1}^{J}x_{ji1}\le 1 \qquad \forall i$$

$$\sum_{j=1}^{J}x_{jir+1}\le\sum_{j=1}^{J}x_{jir} \qquad \forall i, r=1,\ldots,s_i-1$$

$$M\left(1-\sum_{j=1}^{J}x_{ji,r+1}\right)+\sum_{j=1}^{J}t_{ji,r+1}\ge\sum_{j=1}^{J}t_{jir}+\sum_{j=1}^{J}\beta_{ji}x_{jir}-\sum_{j=1}^{J}\gamma_{ji}y_{jir}+\sum_{j=1}^{J}\Delta_j x_{jir} \quad \forall i, r=1,\ldots,s_i-1$$

$$\sum_{j=1}^{J}x_{jir}Q_{ji}\le\sum_{j=1}^{J}t_{jir}\le\sum_{j=1}^{J}x_{jir}R_{ji}-\sum_{j=1}^{J}\beta_{ji}x_{jir}+\sum_{j=1}^{J}\gamma_{ji}y_{jir}-\sum_{j=1}^{J}\Delta_j x_{jir} \qquad \forall i, r=1,\ldots,s_i-1$$

$$y_{jir}\le Mx_{jir}, \quad y_{jir}\le t_{ir}, \qquad y_{jir}\ge 0 \qquad \forall i,\forall j, r=1,\ldots,s_i$$

$$t_{ir}\ge 0 \qquad \forall i, r=1,\ldots,s_i$$

$$x_{jir}\in\{0,1\} \qquad \forall i,\forall j, r=1,\ldots,s_i$$

$$0\le Y_{\alpha k}\le U_{\alpha k} \quad \forall k,\alpha=1,2,\ldots,\mathrm{A}$$

4. MODELİN DOĞRULAMASI

Formülasyon bölümünde verilen asıl matematik model ile doğrusallaştırılan matematik model, iki tehdit, iki silah ve bir merkez için sekiz farklı senaryoda çözülmüştür. Senaryolar Tablo 1'de görülebilir. Asıl matematik modelde bulunan zamana bağlı olasılık değerleri için eğri uydurma algoritmaları ile beşinci dereceden bir polinom bulunmuştur. Asıl model GAMS optimizasyon paketi içinde bulunan BARON çözücüsü ile çözülmüştür. Asıl modelin çözüm zamanı senaryo başına 8-12 saniye arasında değişiklik göstermektedir. Doğrusallaştırılan modelin çözüm zamanı ise 0,1 saniye civarındadır. İki modelin çözüm sonuçları Tablo 2'de sunulmuştur.

Tablo 1. Modelleri test etmek amacıyla oluşturulan senaryolarda olasılık değerleri

	TAVİ Değerleri	q_{ik} Değerleri
Senaryo 1	$p_{11} = p_{12} = p_{21} = p_{22}$	$q_{11} = q_{21}$
Senaryo 2	$p_{11} > p_{12} = p_{21} = p_{22}$	$q_{11} = q_{21}$
Senaryo 3	$p_{11} > p_{12} > p_{21} = p_{22}$	$q_{11} = q_{21}$
Senaryo 4	$p_{11} > p_{12} > p_{21} > p_{22}$	$q_{11} = q_{21}$
Senaryo 5	$p_{11} = p_{12} = p_{21} = p_{22}$	$q_{11} > q_{21}$
Senaryo 6	$p_{11} > p_{12} = p_{21} = p_{22}$	$q_{11} > q_{21}$
Senaryo 7	$p_{11} > p_{12} > p_{21} = p_{22}$	$q_{11} > q_{21}$
Senaryo 8	$p_{11} > p_{12} > p_{21} > p_{22}$	$q_{11} > q_{21}$

Tablo 2. İki modelin çözümlerinin karşılaştırılması

	Doğrusallaştırılmış Model	Asıl Model	Fark	
	Toplam Olasılık	Toplam Olasılık	Büyüklük	Yüzde
Senaryo 1	0.9422	0.9422	0.0000	0.00%
Senaryo 2	0.9599	0.9616	0.0017	0.18%
Senaryo 3	0.9583	0.9619	0.0036	0.38%
Senaryo 4	0.9474	0.9480	0.0006	0.07%
Senaryo 5	0.9251	0.9251	0.0000	0.00%
Senaryo 6	0.9427	0.9562	0.0135	1.41%
Senaryo 7	0.9525	0.9562	0.0037	0.39%
Senaryo 8	0.9234	0.9484	0.0250	2.64%

5. SONUÇ

Sonuç olarak, amaç, taktik ve olasılıklar nedeni ile doğrusal ifade edilemeyen silah - tehdit eşleme problemine, doğrusallaştırma teknikleri uygulayarak yeni model oluşturulmuştur. Bu modelin, asıl model ile aynı senaryolar üzerinde çözülmesi ile çözüm kalitesi ve çözüm süresi bakımından geçerliliği kanıtlanmıştır.

TEŞEKKÜR

Bu bildirinin yazarları bildirinin hazırlanmasına katkılarından dolayı Orhan Karasakal ve İnci Yüksel'e teşekkür ederler.

KAYNAKÇA

[1] Karasakal O. (2004), "Optimal Air Defense Strategies For A Naval Task Group", Doktora Tezi, ODTÜ, Ankara

[2] Metler W.A., Preston F.L. , (1989), "Solutions to A Probabilistic Resource Allocation Problem", Proceedings of the 28th Conference on Decision and Control Tampa, FL, pp.1606–1611

ATIŞ KONTROL SİSTEMİ HATA ANALİZİ İLE TEK ATIŞTA VURUŞ İHTİMALİNİN HESAPLANMASI

İnci YÜKSEL [(a)], Nur Evin ÖZDEMİREL [(b)], Levent KANDİLLER [(c)],

[(a)] ASELSAN A.Ş. Macunköy/Ankara, iyuksel@aselsan.com.tr

[(b)] O.D.T.Ü., Endüstri Mühendisliği, nurevin@ie.metu.edu.tr

[(c)] O.D.T.Ü., Endüstri Mühendisliği, kandiller@cankaya.edu.tr

ÖZ

Bildiride bir atış kontrol sisteminin yerden havaya tek atışta vuruş ihtimalinin hata analizi yolu ile hesaplanması metodu verilmiştir. Bu metot geliştirilirken yerden yere atış için kullanılan iki boyutlu yöntemler temel alınmış ve bunların üç boyutlu sistemler için uyarlaması yapılmıştır. Önerilen yöntemle mevcut atış kontrol sisteminin herhangi bir hedef için hesaplanabilen tek atışta vuruş ihtimali, silah tehdit eşleme problemlerinin karar destek amaçlı optimizasyonunda ve simülasyonunda kullanılabilir. Oluşturulan model ve önerilen metot atışlı testlerden alınan veriler ile test edilmiş ve sonuçları atışlı testlerin sonuçları ile karşılaştırılarak doğrulanmıştır.

Anahtar Kelimeler: atış kontrol sistemi, etkin değer hatası, hata analizi, iki değişkenli gauss tipi olasılık dağılımı, tek atışta vuruş ihtimali

CALCULATION OF SINGLE SHOT HIT PROBABILITY OF A FIRE CONTROL SYSTEM BY ERROR ANALYSIS

ABSTRACT

A method for calculating single shot hit probability of a fire control system, which is firing from surface to air, by error analysis is proposed that can be used in the design process to measure the performance of the design and be proactive. While this method is being developed, two dimensional methods for firing from surface to air are taken as basis and they are adapted to the three dimensional models. In addition, single shot hit probability of a fire control system, which is calculated through this method, can be used in optimization and simulation of weapon target allocation problems for decision support. The model and the method proposed are verified by means of the data of real fire tests.

Keywords: bi-variate normal probability distribution, error analysis, fire control system, single shot hit probability, root mean square error

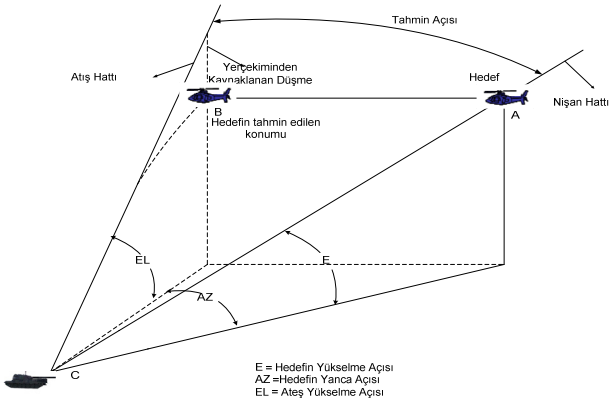
1. GİRİŞ

Bu çalışmada atış kontrol problemi için kullanılan genel geometri Şekil 1'de verilmiştir. Şekil 1'e göre, silah C noktasında konumlanmıştır ve hedef A noktasında

görülmüştür. Atış kontrol sistemi B noktasında verilen, hedefin gelecek konumunu belirler. Atış kontrol probleminin çözümü, ilk olarak A noktasında görülen hedefi vurmak için ne şekilde atış yapmalıyım sorusunun yanıtıdır. Dolayısı ile çözüm, hedefin ilk konumu ve irtifası kullanılarak "Balistik Yükseliş (EL)" ve "Balistik Yan (AZ)" açılarının bulunması ile elde edilir.

2. HATA ANALİZİ

Nişan alınan noktanın hedefin orta noktası olduğu düşünülür, dolayısı ile sistemde herhangi bir hata olmadığı durumda Uçuş Süresi (Time Of Flight-TOF) kadar zaman sonra gerçekleşen çarpışmanın nişan alınan noktada olması gerekmektedir. Fakat atış kontrol sistemindeki herhangi bir hata nişan noktasının yerinin değişmesine sebep olmakta, bu ise bir hata analizi yapılması gereğini ortaya koymaktadır.



Şekil 1. Atış Kontrol Probleminin Genel Geometrisi

2.1 Hataların Sınıflandırılması

Bu analizde hataların sınıflandırılması, ABD Savunma Bakanlığı kaynakları [1] referans alınarak yapılmıştır. Bir atış kontrol sisteminin fiziksel hataları sistematik hata veya tesadüfî hata olarak ikiye ayrılmaktadır. Sistematik hatalar tek yönde oluşan hatalar (bias) olup varlığı nişan alınan noktanın olasılık dağılımının orta noktasının nişan alınan noktadan hata miktarı kadar ötelenmesine neden olur. Tesadüfî hataların var olduğu durumda ise çarpışma noktasının, merkezi nişan alınan nokta olan dairesel normal dağılıma sahip olduğu varsayılır. Genel olarak atış kontrol hataları dört sınıfa ayrılmıştır [1]:

a. Sabit Sistematik Hatalar (Fixed Biases, μ_1): Yerçekimi ivmesi, dönü-dengeli mermiler için tabii yan, nişan Hattı/Silah Paralaksı, zarar görmüş ya da ayarsız herhangi bir atış kontrol ekipmanı nedeniyle oluşan diğer hata kaynakları. Bu hata nişan alınan noktayı tek tönde μ_1 kadar saptırır.

b. Durumdan duruma değişen sistematik hatalar (Occasion-to-occasion biases, μ_2): Araç eğiminin yol açtığı hatalar, hava yoğunluğundaki değişim, hava sıcaklığındaki değişim, vb. Bu hata nedeni ile nişan noktasında meydana gelen

sapma (μ_2), (a)'dan gelen μ_1 ortalaması ile (b)'deki varyansın yarattığı dağılımdan çekilir.

c. Patlamadan patlamaya değişen sistematik hatalar (Burst-to-burst biases, μ_3): Bu hata nedeni ile nişan noktasında meydana gelen sapma (μ_3), (b)'den gelen μ_2 ortalaması ile (c)'deki varyansın yarattığı dağılımdan çekilir.

d. Mermiden mermiye değişen hatalar (Round-to-round errors, μ_4),: Mühimmat dağılımı, vb. Bu hata nedeni ile nişan noktasında meydana gelen sapma (μ_4), (c)'den gelen μ_3 ortalaması ile (d)'deki varyansın yarattığı dağılımdan çekilir.

Tek atışta vuruş ihtimali (TAVİ) hesaplamalarında, bu hatalardan bir olasılık dağılımı elde edilmektedir. Sabit sistematik hatalardan (a) gelen μ_1'e ek olarak patlamadan patlamaya, durumdan duruma ve mermiden mermiye değişen hatalardan (b-c-d) gelen varyanslar göz önüne alınarak TAVİ hesabı yapılır [1].

2.2 Hedefi vurma Performansını Etkileyebilecek Hata Kaynakları

2.2.1 Hata Kaynaklarının Sınıfandırılması

Bu bölümde, söz konusu hatalara sebep olabilecek hata kaynakları Bölüm 0'de verilen sınıflandırmaya uygun olarak sistematik hata veya tesadüfî hata olarak sınıflandırılmıştır. Bu hata kaynakları sistemin girdi veya çıktılarından kaynaklanıyor olabilir. İncelenen atış kontrol sisteminin hata kaynakları ve sonuçları Çizelge 1'de listelenmiştir. Bu atış kontrol sisteminde göreli hedef hızı hatası dışındaki hata kaynaklarının yol açtığı sistematik hataların tamamı kompanse edilmiş olduğu için bu hata kaynakları sistematik hataya sebep olmamaktadır.

2.2.2 Hata Kaynaklarının Yol Açtığı Hatanın Ölçülmesi

Atış kontrol sisteminde her bir hata kaynağının sistemin çıktısında yol açtığı hatayı ölçmek için her bir hata kaynağının yol açtığı hatanın sistem içindeki yayılımının atış kontrol sisteminde kullanılan denklemler ile belirlenmesi gerekmektedir. Hataların bir atış kontrol sistemi içerisindeki akışı, o hata ile ilgili sinyallerin atış kontrol sistemi içerisindeki akışıdır. Sistem çıktısındaki hata, atış kontrol sistemindeki bütün hataların sonucudur. Sonuç olarak elde edilen toplam sistem hatası, ortalama sistematik hata ve standart sapma olarak ifade edilir. Her bir hata kaynağının sistemin toplam hatasına olan katkısının bulunması için öncelikle sistem girdilerinin ve çıktılarının ilişkisinin gösterilmesi gerekmektedir. Sistemdeki girdi-çıktı ilişkilerinin bulunması amacı ile her olası hata kaynağı incelenmelidir.

Tahmin Açısındaki Hata: Tahmin açısının hatalı hesaplanması hedefe ateş ederken namluya verilecek balistik yükseliş ve yan açılarının yanlış hesaplanması ile ilgilidir. Atış kontrol problemi yükselişçe ve yanca olmak üzere iki boyutta çözülür ve namluya verilecek önleme ve ilave yükseliş açıları balistik hesaplamalarla bulunur. Önleme ve ilave yükseliş açıları hesaplanırken, oluşabilecek bazı hatalar kompanse edilmeye çalışılır. Kompanse edilemeyen hatalar, sistemde tek yönlü hataya sebep olurken kompanse edilmiş hatalar ölçüm hataları sebebi ile sistemde tesadüfî hataya sebep olur. Bu hata kaynaklarının ne kadar hataya yol açtığının hesaplanması balistik denklemlerden türetilen hata yayılım fonksiyonları kullanılarak yapılır. Her bir hata kaynağının çıktıda yol açtığı hatayı bulmak amacı ile her hata kaynağının "etkin değer

hatası (root mean square error-RMSE)" kullanılacaktır. Bazı hata kaynakları için mevcut olmayan RMSE değerleri için algılayıcıların ±Δ şeklindeki ölçüm hassaslığı aralığında düzenli olarak dağıldıkları varsayılmıştır.

Çizelge 1. Hata Kaynakları

Hata Kaynakları		Sistematik Hata	Tesadüfî Hata
1. Silah yönlenme hatası			√
2. Elle takip/nişan alma hatası			√
3. Nişan hattı stabilizasyonu hatası			√
4. Göreli hedef hızı hatası		√	√
5. Hedef mesafesi hatası			√
6. Balistik hesaplamalar			√
7. Silah stabilizasyonu hatası			√
8. Mermiden mermiye değişen mühimmat dağılımı hatası			√
9. Yanlılık (bias) faktörleri ve yanlılık parametrelerinin ölçümünde veya hesaplanmasındaki hatalar	9.1. Kule yan eğimi		√
	9.2. Namlu ucu sapması ve bozucu etkiler		√
	9.3. Yan rüzgâr		√
	9.4. Mesafe rüzgârı		√
	9.5. Mermi ilk hızı hesaplama hatası		√
	9.6. Hava sıcaklığı ölçümü		√
	9.7. Hava basıncı ölçümü		√
	9.8. Nişan ayarı hatası		√
	9.9. Atışlı sıfırlama		√

Tahmin açısında hataya yol açan her bir hata kaynağının çıktı hatası, balistik açıların hesaplanmasında kullanılan denklemler yardımıyla elde edilir:

Her hata kaynağı için:

1. Hiçbir hatanın olmadığını varsayarak balistik yan ve yükseliş açılarının hesaplanır. $f(y, x_1,x_2,\ldots x_n) = O_0$

   $f(y, x_1,x_2,\ldots x_n)$: Balistik yan ve yükseliş açılarını bulmak için kullanılan balistik denklemlerin oluşturduğu denklem sistemi

   y : Hata kaynağı ile ilgili parametre, y=TRR_A,TRR_E, TR, θ, MD, MT, W, WD, T_g, T, P, X_R, Y_R, R_{Bor}, A_{Z0}, E_{Z0}, $R_{gerçek}$ (Kısaltmaların hangi hata kaynaklarına ait oldukları aşağıda hata kaynakları açıklanırken verilmiştir.)

   $x_1,x_2,\ldots x_n$: Balistik yan ve yükseliş açılarını elde etmek için kullanılan diğer parametreler

   O_0 : Hata olmadığı durumda elde edilen çıktı

2. Balistik yan ve yükseliş açılarını, hata kaynağının hatası ile genişletilen orijinal hata parametresini balistik denklemlerde kullanarak bul:

   2.1. Hata kaynağı sistematik hataya yol açıyorsa:

   $$\alpha_y = \left| f(\sigma_y + y, x_1,x_2,\ldots x_n) - O_0 \right|$$

α_y : Hata kaynağının çıktıda meydana getirdiği hata

σ_y : Hata kaynağının RMS hatası

2.2. Hata kaynağı tesadüfî hataya yol açıyorsa:

$$\alpha_y = 0.5\left(\left|f(y+\sigma_y, x_1, x_2, \ldots x_n) - O_0\right| + \left|f(y-\sigma_y, x_1, x_2, \ldots x_n) - O_0\right|\right)$$

Sonuç olarak, tahmin açısındaki hatanın hesaplanması için bu hataya sebep olan hata kaynaklarının tesadüfî ve sistematik hataya sebep olan hata kaynakları olarak iki gruba ayrılması gerekmektedir.Tahmin açısında tesadüfi hataya sebep olan hata kaynakları girdilerin ve balistik denklemlerin doğruluğu ile ilgilidir. Balistik denklemlerin girdileri atış kontrol sisteminde bulunan algılayıcılardan alınmaktadır. Girdilerin bir kısmı kompanse edilmiş yanlılık faktörleri (bias factors) ile ilgilidir. Ayrıca, bu faktörler ile ilgili ölçüm hataları balistik açıların yanlış hesaplanmasına neden olabilmektedir. Bu hata kaynakları aşağıda listelenmiştir:

a) Hedef mesafesi ölçüm hatası (TR)
b) Balistik hesaplamalardan kaynaklanan hata (BC)
c) Elle takip/nişan alma hatası (MT)
d) Mermiden mermiye değişen mühimmat dağılımı hatası (AD)
e) Yanlılık faktörleri ve yanlılık parametrelerinin ölçümünde veya hesaplanmasındaki hatalar
    i. Kule yan eğimi ölçüm hatası (θ)
    ii. Namlu ucu sapması ve bozucu etkiler (MD)
    iii. Rüzgâr hızı ölçüm hatası (Yan Rüzgâr – Mesafe Rüzgârı) (W)
    iv. Rüzgâr yönü ölçüm hatası (Yan Rüzgâr – Mesafe Rüzgârı) (WD)
    v. Cephane Sıcaklığı ölçüm hatası (Mermi ilk hızı hesaplama hatası) (T_g)
    vi. Hava sıcaklığı ölçüm hatası (T)
    vii. Hava basıncı ölçüm hatası (P)
    viii. Ayna ve namlu arası uzaklığın ölçüm hatası (Boresight Hatası) (X_R, Y_R)
    ix. Boresight Mesafesi ölçüm hatası (Boresight Hatası) (R_{Bor})
    x. Atışlı sıfırlama açıları (A_{Z0}, E_{Z0})
    xi. Göreli hedef yan hızı (TRR_A)
    xii. Göreli hedef yükseliş hızı (TRR_E)

Tahmin açısında sistematik hataya yol açan iki tip hata kaynağı vardır. Bunlar hedef mesafesi ölçümü ile ateş etme arasında geçen zaman ve göreli hedef yaklaşma hızıdır. Hedef mesafesi ölçümü ile ateş etme arasında geçen zaman diliminde hedefin göreli hareketi yok sayılmaktadır. Bunun yanında hedefin silaha yaklaşma yönündeki göreli hızı sistem tarafından kompanse edilmemiş olması nedeni ile hedefin TOF boyunca

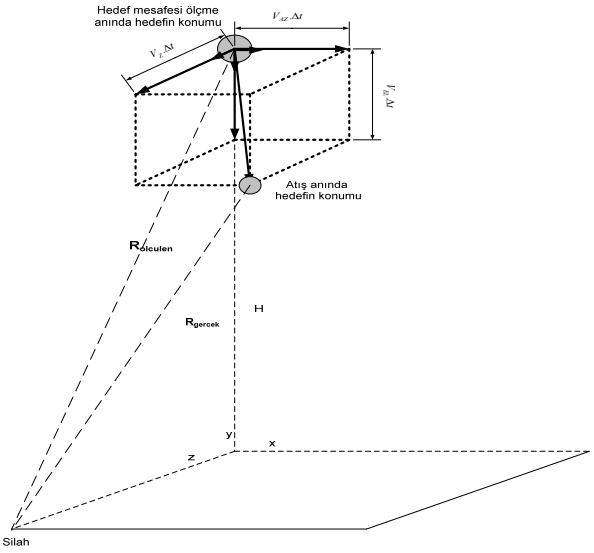
silah yönündeki göreli hareketi de yok sayılmaktadır. Bu iki hata kaynağı sebebi ile sistem balistik yükseliş ve balistik yan açılarını yanlış hedef mesafesi kullanarak hesaplamaktadır, dolayısı ile çarpışma noktası hedefin merkezinden ötelenmiş olmaktadır. Hedefin göreli hareketi nedeniyle oluşan bu hataların genel geometrisi Şekil 2Şekil ’de verilmiştir. Gerçek hedef mesafesi ile hesaplanan balistik yan ve yükseliş açısı ile ölçülen hedef mesafesi ile hesaplananlar arasındaki fark çarpışma noktasındaki yanca ve yükselişçe yer değiştirmeyi verir. Yanca yer değiştirme, yancadaki tek yönlü tahmin açısı hatasını (μ_A); yükselişçe yer değiştirme, yükselişçedeki tek yönlü tahmin açısı hatasını (μ_E) vermektedir. Dolayısıyla sistematik hata hesaplaması, hata hesaplama sürecine hatasız değer olarak gerçek hedef mesafesi, hatalı değer olarak ölçülen hedef mesafesi girilerek hesaplanır. Gerçek hedef mesafesi şu şekilde hesaplanır:

$$R_{\text{gercek}} = \sqrt{(\Delta t.V_{AZ})^2 + (\sqrt{R_{olculen}^2 - H^2} - (\Delta t + TOF).Vz)^2 + (H - \Delta t.V_{EL})^2} \qquad (1)$$

Δt : Hedef mesafesi ölçme ile ateş etme arasında geçen süre, sn

R_{gercek} : Atış etme esnasındaki hedef mesafesi, m

$R_{olculen}$: Ölçülen hedef mesafesi, m



Şekil 2. Hedef Mesafesi Ölçüm Hatasının Genel Geometrisi

Uçuş Süresindeki Hata: Bu hata, birbirlerine göre hareketli hedef ve silahlar için geçerlidir. Uçuş süresinin yanlış hesaplanması, hedefin gelecekteki konumunun tahmininin yanlış olmasına sebep olmaktadır. Uçuş süresi, aynı zamanda nişan alınan noktanın gerçek nişan alınan noktadan sapmasına da sebep olabilen birçok hata kaynağı nedeni ile hatalı hesaplanabilmektedir. Uçuş süresi yanında atış hattında sapmaya da neden olan bu hata kaynaklarının diğer hata kaynaklarından daha farklı bir şekilde hesaplanması gerekmektedir, çünkü uçuş süresindeki hata ve atış hattının

yönlenmesindeki hata birbirinden bağımsız hatalar değillerdir. Uçuş süresi hatası aşağıdaki gibi hesaplanmaktadır:

$$\alpha_{X_TOF_A} = \arctan\left(\frac{\sigma_{X_TOF} V_{AZ}}{\sqrt{R^2 - H^2} - \sigma_{X_TOF} V_z}\right) \quad (2)$$

$$\alpha_{X_TOF_EL} = \arctan\left(\frac{H - \sigma_{X_TOF} V_{EL}}{\sqrt{R^2 - H^2} - \sigma_{X_TOF} V_z}\right) - \arctan\left(\frac{H}{\sqrt{R^2 - H^2}}\right) \quad (3)$$

σ_{TOF} : Uçuş süresindeki hata (s)

$\alpha_{X_TOF_AZ}$: Yancada uçuş süresi hatası (milyem)

$\alpha_{X_TOF_EL}$: Yükselişçede uçuş süresi hatası (milyem)

V_{AZ}: Göreli hedef yan hızı (m/s)

V_{EL}: Göreli hedef yükseliş hızı (m/s)

V_Z: Göreli hedef yaklaşma hızı (m/s)

H : Hedefin irtifası (metre)

R : Hedef mesafesi (metre)

Uçuş süresindeki hata, σ_{TOF}, ölçüm hatalarının uçuş süresi hesaplamalarında dikkate alındığı balistik denklemler ile hesaplanmaktadır. Aşağıdaki hata kaynakları hem tahmin açısı hatasına hem de TOF hatasına sebep olmaktadır:

a) Hedef mesafesi hatası (TR)
b) Balistik hesaplamalardan kaynaklanan hata (BC)
c) Yanlılık parametrelerinin ölçümünde veya hesaplanmasındaki hatalar
    i. Rüzgâr hızı ölçüm hatası (Yan Rüzgâr – Mesafe Rüzgârı) (W)
    ii. Rüzgâr yönü ölçüm hatası (Yan Rüzgâr – Mesafe Rüzgârı) (WD)
    iii. Cephane Sıcaklığı ölçüm hatası (Mermi ilk hızı hesaplama hatası) (T_g)
    iv. Hava sıcaklığı ölçüm hatası (T)
    v. Hava basıncı ölçüm hatası (P)

σ_X'in yukarıdaki hata kaynaklarından herhangi birinin RMSE değeri olduğunu varsayalım. Genel olarak, bu hata kaynağı tahmin açısında, yancada $\alpha_{X_A_AZ}$ milyem, yükselişçede ise $\alpha_{X_A_EL}$ milyemlık hataya sebep olur. Bunun yanında σ_X RMSE, TOF'ta σ_{X_TOF} saniyelik bir hataya sebep olmaktadır. Bu üç değer, "Tahmin Açısındaki Hata" bölümünde verilen balistik denklemlerin hatalı ve hatasız inputlar ile çözülüp sonuçların farklarının alındığı süreç kullanılarak bulunur. σ_{X_TOF} saniyelik TOF hatası,

göreli yan, yükseliş ve yaklaşma hızı sırası ile V_{AZ}, V_{EL} ve V_Z olan hareketli hedeflerin yancada, yükselişçede ve yaklaşma yönünde sırası ile $V_{AZ}.\sigma_{X_TOF}$, $V_{EL}.\sigma_{X_TOF}$ ve $V_Z.\sigma_{X_TOF}$ metre yer değiştirmelerine sebep olur. Bu yer değiştirmeler, nişan hattının yancada $\alpha_{X_TOF_AZ}$ milyem, yükselişçede ise $\alpha_{X_TOF_EL}$ milyem sapmasına neden olur. Bu hatanın sistem hatasına toplam katkısı, hata kaynağının yol açtığı tahmin açısı ve TOF hatasının karelerinin toplamının karekökü (Root sum of squares-RSS) ile bulunur [2]. Dolayısı ile hata kaynağının sebep olduğu toplam hata yancada $\sqrt{\alpha_{X_A_AZ}^{\ 2} + \alpha_{X_TOF_AZ}^{\ 2}}$ milyem, yükselişçede $\sqrt{\alpha_{X_A_EL}^{\ 2} + \alpha_{X_TOF_EL}^{\ 2}}$ milyem olur.

Silah Yönlenme Hatası: Silah stabilizasyonu hatası, namlunun silah hareketi sebebi ile nişan alınan noktadan sapması sonucu oluşan hatadır. Bu tip hata, namlunun gerçek nişan noktasını yancada (yükselişçede) σ_{WSX} (σ_{WSY}) kadar bir hata ile göstermesi, atış hattının yönlenmesi gereken doğrultudan yancada (yükselişçede) α_{WSX} (α_{WSY}) kadar sapmasına sebep olmaktadır. Bir diğer hata kaynağı ise silahın hareketinden etkilenen nişan hattı stabilizasyonu hatasıdır. Bu tip hatada namlunun gerçek nişan noktasını yancada (yükselişçede) σ_{LSX} (σ_{LSY}) kadar bir hata ile göstermesi, atış hattının yönlenmesi gereken doğrultudan yancada (yükselişçede) α_{LSX} (α_{LSY}) kadar sapmasına sebep olmaktadır. Silah yönlenme hatası şu şekilde hesaplanmaktadır:

$$\alpha_{WOX} = \sqrt{\sigma_{WSX}^{\ 2} + \sigma_{LSX}^{\ 2}} \quad (4)$$

$$\alpha_{WOY} = \sqrt{\sigma_{WSY}^{\ 2} + \sigma_{LSY}^{\ 2}} \quad (5)$$

α_{WOX} : Yancada silah yönlenme hatası, (milyem)

α_{WOY} : Yükselişçede silah yönlenme hatası, (milyem)

σ_{WSX} : Yancada silah stabilizasyonu hatası, (milyem)

σ_{WSY} : Yükselişçede silah stabilizasyonu hatası, (milyem)

σ_{LSX} : Yancada nişan hattı stabilizasyonu hatası, (milyem)

σ_{LSY} : Yükselişçede nişan hattı stabilizasyonu hatası, (milyem)

Hata analizi sonucunda açı cinsinden hesaplanıp hedef mesafesine göre hedefi kaçırma uzaklığına çevrilen sistemin toplam sistematik ve tesadüfî hatası bulunur. Bunlar, tek atışta vuruş ihtimali hesaplamak için gerekli hata dağılımlarının parametreleridir. Sistemde bulunan bütün sistematik hatalar birbirinden bağımsızdır. Dolayısı ile sistemin toplam sistematik hatası, sistemde bulunan bütün sistematik hataların kareleri-toplamının-karekökü olarak bulunur. Sistemde farklı hata kaynaklarından kaynaklanan bütün tesadüfî hatalar birbirinden bağımsızdır. Bu nedenle sistemin toplam tesadüfî hatası, bütün tesadüfî hataların kareleri-toplamının-karekökü'dür.

3. TEK ATIŞTA VURUŞ İHTİMALİNİN (TAVİ) HESAPLANMASI

Tek atışta vuruş ihtimali (TAVİ), atılan tek bir merminin hedefi vurma ihtimali olarak tanımlanır. TAVİ, yapılan hata analizi sonucu elde edilen atış kontrol sisteminin

sistematik ve tesadüfî hatalarından çıkarılan olasılık dağılımı kullanılarak bulunur. Bunun yanısıra, hedefin de modellenmesi gerekmektedir.

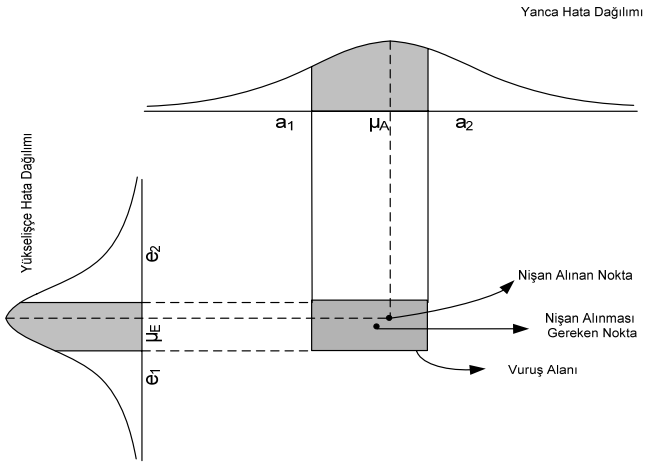
Hata analizi, atış hattındaki sapmanın iki boyutlu bir düzlem üzerinde olduğunu varsaymaktadır. Fakat hedef üç boyutlu bir hacme sahiptir. TAVİ'nin hesaplanması için hedefin merminin hız vektörüne dik olan düzlem üzerindeki izdüşümü alınır. Gerçek hedefler için izdüşümün bulunması zor bir süreç olduğu için hedef ya da hedef üzerindeki hassas bölge daire, kare ya da dikdörtgen olarak modellenir [2]. Bu hata dağılımları ortalamaları μ_A ve μ_E, standart sapmaları σ_A and σ_E olan Gauss tipi (Normal) olasılık dağılımına sahiptir. Olasılık dağılımlarının ortalama ve standart sapmaları aşağıdaki gibi bulunur:

μ_A / μ_E = Yancadaki/Yükselişçedeki sistematik hata kare-toplamının-karekökü

σ_A / σ_E = Yancadaki/Yükselişçedeki tesadüfî hata kare-toplamının-karekökü

TAVİ hesaplamasına geçmeden önce yapılan varsayımları anlatmak önemlidir:

1. Yanca ve yükselişçedeki hata dağılımları Gauss tipi hata dağılımları olup birbirlerinden bağımsızdırlar.
2. Atış yapılan hedeflerin çarpışma anında mermi yoluna dik olan düzlem üzerindeki izdüşümlerinin her zaman kenarları koordinat eksenleri ile çakışan bir dikdörtgen olduğu kabul edilmiştir ve bu izdüşüm alanı "vuruş bölgesi" olarak adlandırılmıştır.
3. Atış kontrol sisteminin iyi tasarlanmış olduğu ve sistematik hatalarının bir çoğunun balistik hesaplamalar ile kompanse edilmiş olduğu varsayılmıştır. Fakat hedefin silaha yaklaşma yönündeki göreli hızı ve ateş etme ile mesafe ölçme arasında geçen zamanın kompanse edilmemiş olması nedeni ile yanca ve yükselişçedeki hata dağılımlarının ortalamaları sıfırdan farklıdır: $\mu_A \neq 0,\ \mu_E \neq 0$.
4. Nişan alınan nokta hedefin ağırlık merkezinden μ_A ve μ_E kadar ötelenmiştir.
5. Hatalar hesaplandıktan sonra bulunan çarpışma noktası vurma alanı içinde ise hedefin vurulduğu varsayılmıştır.



Şekil 3. İki değişkenli gauss tipi olasılık dağılımı varlığında TAVİ hesaplama metodu

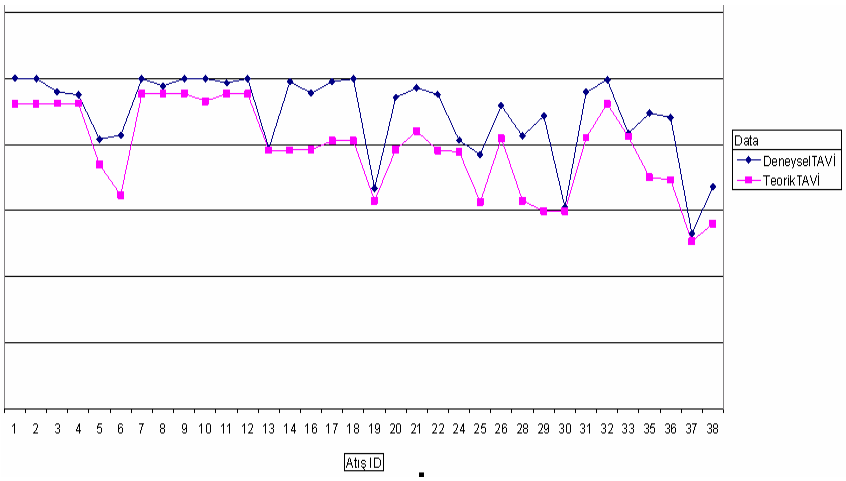
TAVİ hesaplamaları için kullanılan metot Şekil 3'te verilmiştir. Yanca hata dağılımı ve yükselişçe hata dağılımları, R mesafesindeki bir hedef için çarpışma noktasının yer aldığı bir düzlem oluşturmaktadır. Başarılı bir vuruş yapabilmek için nişan alınan noktanın hedefin ağırlık merkezi olduğu durumda, yancadaki hata a_1 ve a_2 arasında, yükselişçedeki hata da e_1 ve e_2 arasında bir değer almalıdır. Hedef modeli ile ilgili varsayım vuruş alanının kenarları koordinat eksenleri olan *a* ve *e* ile çakışık bir dikdörtgen olduğunu garanti etmektedir. Dolayısı ile yanca ve yükselişçe hata dağılımlarının birbirinden bağımsız olduğu durumda yancadaki hatanın a_1 ve a_2 arasında olması olasılığı ile yükselişçedeki hatanın e_1 ve e_2 arasında olması olasılığının çarpımı çarpışma noktasının vuruş alanı içinde olma olasılığını verir. Sonuç olarak TAVİ değeri birbirinden bağımsız iki Gauss tipi olasılık dağılımı ile hesaplanır. TAVİ hesaplamalarında *A* ve *E* rastlantısal değişkenleri ile gösterilen yanca ve yükselişçedeki hata dağılımları arasında korelâsyon yoktur, dolayısı ile TAVİ hesaplamalarında kullanılacak kümülâtif olasılık fonksiyonunda kullanılan A_R, hedefin hassas bölgesinin mermi hızına dik olan düzlem üzerindeki izdüşümünün alanını göstermektedir:

$$F(a,e) = \iint_{A_R} \frac{1}{2\pi\sigma_A\sigma_E} \exp\left\{\frac{-1}{2}\left[\frac{(a-\mu_A)}{\sigma_A^{\,2}} - \frac{(e-\mu_E)}{\sigma_E^{\,2}}\right]\right\} dade \qquad (6)$$

TAVİ değerlerini bulmak için kullanılan hata dağılımları, hata kaynaklarının RMSE değerleri kullanılarak bulunmuştur. Dolayısı ile bu hata dağılımları bir standart sapmalık beklenen hata değişimini gösterir. Bu da atışlar sonucu elde edilecek TAVİ değerlerindeki hatanın atışların üçte ikisi için bulunan hatadan daha az olması anlamına gelmektedir [3]. Sonuç olarak, bulunan TAVİ değerleri, atışlarda bulunan değerlerden yapılan atışların üçte ikisi için daha düşük olmalıdır.

Tank atış kontrol sisteminin TAVİ değerini hata analizi ile hesaplama yöntemi bir atış kontrol sistemine TAVİ değerlerini teorik olarak bulmak amacı ile uygulanmıştır. Modeli geçerli kılmak amacı ile atışlı testlerden alınan girdiler ile model denenmiş ve

elde edilen teorik sonuçlar atışlı testlerden elde edilen deneysel sonuçlar ile teorik sonuçların istatistik anlamları göz önünde bulundurularak karşılaştırılmıştır. Karşılaştırma sonuçları Şekil 4'de verilmiştir. Gizlilik nedeni ile bu testlerin ayrıntıları hakkında bilgi verilememektedir.



Şekil 4. Teorik TAVİ ile Deneysel TAVİ Değerlerinin Karşılaştırılması

4. SONUÇ ve ÖNERİLER

Bu çalışma, hata analizi yolu ile TAVİ hesaplanmasına ilişkin metodun bir atış kontrol sistemi için uygulanabilir olduğunu ve metot kullanılarak elde edilen TAVİ sonuçlarının gerçeğe uygunluğunu göstermiştir. Önerilen metot bir atış kontrol sisteminin tasarımı sırasında alt sistemlerin seçiminin yapılması için kullanılabilecektir. Bu metot ile bulunan TAVİ değerleri silah tehdit eşleme problemlerinin karar destek amaçlı optimizasyonunda ve benzetiminde kullanılabilecektir. Ayrıca, bu metot kullanılarak yapılan bir atış kontrol sistemi tasarımının TAVİ ile ilgili sistem gereklerini karşılayıp karşılamadığı tasarım aşamasından öngörülebilecek, karşılamadığı durumda gerekli önlemler yine bu metot kullanılarak önceden alınabilecektir. Metot, farklı atış kontrol sistemleri için de kullanılabilir.

TEŞEKKÜR

Yazarlar, hata değerlerinin belirlenmesinde ve modeldeki katkılarından dolayı Emre Ayazoğlu, Ramazan Arık ve Alper Erdener'e, model hakkındaki değerlendirmeleri için Hüseyin Yavuz, Faruk Mengüç ve Bülent Mete'ye teşekkür ederler.

KAYNAKÇA

[1] *U.S. Department of Defense Handbook: Fire Control Systems – General* (MIL-HDBK-799 AR). April,1996.

[2] Macfadzean, R., *Surface Based Air Defense System Analysis*, Boston, London:Artech House, 1992. pp.87-133

[3] Metz, D. , Raymond Von Wahle, *Sniper Weapon Fire Control Error Budget Analysis*, Aberdeen. Army Research Laboratories, 1999.

YAZAR DİZİNİ

ACUN Levent 469
AÇIKGÖZ Murat 350
AKDEMİR Cemil 119
AKGÜN İlker 91
AKGÜN Sevgi 131, 138, 447
AKIN Erhan 101
AKIN Umut 169, 180, 394
AKMAN Burcu 429
AKSOY Serkan 131, 138
ALDOĞAN Deniz 15, 119
ALMAZ Alper Ö. 383
ALPDEMİR M. Nedim 271, 339
ALTINOK Taner 221
ALTIOK Tayfur 383
ARAR Ömer Faruk 326
ARIKAN Erkan 82
ARSLAN Dilek 469
ARSLAN Ebru 82, 406
ARSLAN Seçil 469
ATEŞ İsmail Hakkı 231
AVCIBAŞI Yasir Kurutuluş 253
AYDOĞDU Özgür 429
BAHADIRLAR Yıldırım 417
BAL Barış 109
BALCISOY Semih 191
BAŞARAN Erkul 131
BAŞTÜRK 253
BATUN Sakine 469
BIKMAZ İsmail 200, 447, 457
BOZAĞAÇ Doruk 271
BOZDOĞAN Güher 406
BOZYİĞİT Müslim 261
BÖLÜKBAŞ Deniz 131, 138
CAN Hayrettin 101
CENGİZ Ramazan 360
ÇAĞILTAY Kürşat 200, 457
ÇAKIR Gonca 156, 211
ÇAKIR Mustafa 156
ÇALIŞKAN Kemal 51
ÇAPUTÇU Reha 326
ÇELİK Turgay 283
ÇETİN İsmet Emre 109
ÇUHADAR İsmet 27
DEMİR İbrahim 339
DENİZ Ertan 109
DESTANOĞLU Onur 253
DİKENELLİ Oğuz 119, 253, 447
DUMAN Erkan 101
EKİZ Mehmet 37
ELDELEKLİ Ersin 316
ELİŞ Haluk 5
EMEKLİ Özgür 241
ENGİN H. Billur 191
EROĞLU Ömer 253
ERSÖZ Filiz 221
ERTUĞRUL Erkan 91
FIRAT Cüneyd 350
GÖKDOĞAN Göktuğ 283
GÖNÜLTAŞ Erhan 383
GÜLEZ Taner 469, 476

GÜNDÜZ Baki 27
HAKLIDIR Mehmet 15, 71, 326
İMRE Kayhan 457
İPEKKAN Ziya 169, 180
İŞLER Veysi 200, 457
KALE Mithat 429
KANDAKOĞLU Ahmet 91
KANDİLLER Levent 476, 488
KAPLAN Gülay Büyükaksoy 326
KARA Ahmet 271
KARA Bahar Yetiş 306
KARAŞAN Oya Ekin 306
KAZANDIR Bekir 221
KIRCA Ömer 469
KIYAK Ümit 469
KİRKİZOĞLU Zeynep 469
KOÇ Emre 191
KÖKSAL Seniha 119
KUTLUCA Hüseyin 109
MERAL Sedef 469
LEBLEBİCİOĞLU Kemal 316
OCAK Mustafa 169, 180
OR İlhan 383
OĞUZTÜZÜN Halit 27, 437
OKUTANOĞLU Aydın 261, 429
ÖZBAŞ Binnur 383
ÖZDEMİREL Nur Evin 476, 488
ÖZGÜN Meysun Avcı 469
ÖZHAN Gürkan 437
ÖZKAN Günay 283
ÖZKAZANÇ Yakup 61, 145
ÖZKİL Altan 394
SABUNCUOĞLU İhsan 295
SERİM Hüseyin Avni 138
SEVGİ Levent 156, 211
SEVİLGEN F. Erdoğan 253, 360
SOLTER Hürcan Gökçe 253
SOMER Cumhur 469
SOYKAN Bülent 221
SOYKAN Faruk 469
SÜMENGEN Selçuk 191
ŞİMŞİR Mert 429
TANSEL Barbaros 295
TAŞDELEN İsa 15, 119, 253, 447
TATAS Zekiye Bilinç 61
TEMİZER Namık Kemal 5
TEMİZER Selim 5
TEZCAN Zahir 339
TİMAR Yasemin 447
TOYOĞLU Hünkar 306
UÇAN Ferhat 326
ULUSCU Özgecan S. 383
ÜÇLER Çağlar 241
ÜNDEĞER Çağatay 169, 180
ÜNLÜSOY Samim 51
ÜNSAL Ersin 370
ÜSTÜNER Hale 145
VAYVAY Özalp 241
YAVAŞ Altan 169, 180
YAVRUCUK İlkay 82
YÜKSEL İnci 469, 488
YILDIRIM Gökçe 200, 457
YILDIRIM Uğur Ziya 295
ZENGİN Ahmet 37