

## ETSİS TABANLI SİMÜLASYON KAVRAMSAL MODELİ VE SENARYOLARININ DOĞRULAMASI

**M. Fatih Hocaođlu<sup>(a,b)</sup>, Halil Kolsuz<sup>(c)</sup>, Erdem Reşber<sup>(b)</sup>**

<sup>(a)</sup> İstanbul Medeniyet Üniversitesi, Mühendislik ve Mimarlık Fakültesi, 34700  
Göztepe/İstanbul [mfatih.hocaoglu@medeniyet.edu.tr](mailto:mfatih.hocaoglu@medeniyet.edu.tr)

<sup>(b)</sup> Ađena Bilişim ve Savunma Teknolojileri, Teknopark İstanbul 34912  
Pendik/İstanbul, [hocaoglu@agenabst.com](mailto:hocaoglu@agenabst.com)

<sup>(c)</sup> Aselsan, 34844 Macunköy/Ankara, [hkolsuz@aselsan.com.tr](mailto:hkolsuz@aselsan.com.tr)

### ÖZ

Etmen tabanlı simülasyon sistemi EtSiS kavramsal modelin ve senaryoların doğrulanabilmesine imkan tanıyan bir kavramsal model dili ve standart senaryo gösterim yapısı sunmaktadır. Gösterim bir betik ile tanımlandığı için, doğrulama süreci kavramsal model geliştirme ve senaryo hazırlanma aşamalarının her ikisini de kuşatır. Çalışmada, uygulama sahasından bağımsız, kavramsal model ve senaryo doğrulama kurallarının tanımlanıp işletilmesine imkân veren, EtSiS tabanlı genel amaçlı bir kavramsal model ve senaryo doğrulama çerçevesi geliştirilmiştir. Uygulamanın ana amacı, varlıklar arasındaki etkileşimleri, atanan görevleri, model davranışlarını ve kavramsal dünyaya ilişkin betimlemeleri doğrulamak ve benzer şekilde doğrulanmış olan kavramsal modeli baz alarak oluşturulan senaryoların doğruluđunu, kapsayıcılıđını kontrol etmektir.

**Anahtar Kelimeler:** Etmen GÜdümlü Simülasyon, EtSiS, Hava Savunma Simülasyonu, Kavramsal Model Doğrulama, Model tabanlı doğrulama, Senaryo doğrulama

## SIMULATION CONCEPTUAL MODEL AND SCENARIO VERIFICATION BASED ON AdSiF

### Abstract

Agent driven simulation framework AdSiF provides a language that allows verifying conceptual models and also it provides a standardized data structure for scenario representation. Since the representation is defined by a script, verification process covers both development of conceptual model and scenario. In this study, an AdSiF based application domain independent conceptual model and scenario verification process, which defines and executes generic script rules for conceptual model

verification and scenario verification, is developed. The main purpose is to verify consistencies for interactions between entities, assigned missions, model behaviors, and the declarations related with the conceptual world and similarly, to check verification of the scenarios created based on the verified conceptual model and to validate how well scenarios cover the conceptual model.

**Keywords:** AdSiF, Agent driven simulation, Air Defense Simulation, Conceptual model verification, Model based verification, Scenario verification

## 1. GİRİŞ

Kavramsal model hazırlama aşaması simülasyon dünyasına ait tüm varlıkların, kavramların, kuralların, ilişkilerin, algoritmik çözünürlüğün, görev ve davranışların tasarlandığı kritik ve tasarıma önemli girdiler sağlayan bir safhadır. Kavram düzeyinde gelişen bir çalışma olması, sınırlarının çizilmesinde ve nesnelleştirilmesinde bazı zorlukları beraberinde getirmektedir. EtSiS (Etmen tabanlı Simülasyon Sistemi-AdSiF (Agent driven Simulation Framework) kavramsal model bileşenlerinin her birini somut bileşenler olarak, bir betik dil ile tanımlar ve işletimini sağlar. Simülasyon senaryoları kavramsal modelde yer alan varlıklara ait parametre değerlerinin uygulama özelinde belirlendiği, ilişkilerin tanımlandığı, görev seçiminin yapılarak, ilişkili parametrik değerlerin belirlendiği kavramsal modelin işletilebilir bir oluşumu olarak ele alınabilir.

Çalışmada kavramsal model ve senaryo tasarımlarının doğruluğunu kontrol eden kuralların tanımlanmasına, işletimine ve doğrulama sonucunda elde edilen sonuca göre eylem seçimine ilişkin, uygulama sahasından bağımsız, bir yöntem önerilmiştir. Kural tanımlama için EtSiS tabanlı bir betik kural gösterimi geliştirilmiştir.

### 1. Etmen Tabanlı Simülasyon ve Kavramsal Model

Kavramsal model geliştirme yaklaşımları, farklı bakış açıları ve paradigmlar ile geliştirilen kavramsal model çerçeveleri (framework), yazılım mühendisliği ile olan ilişkisi, sahaya özel kavramsal model yaklaşımları ve kesikli olay simülasyon sistemleri için çalışmalar literatürde oldukça geniş bir yer tutmaktadır [1], [2]. Kavramsal model gerçek veya tasarlanan sistemin modelleyici algısını gösterir ve modelleyici algısını güçlendirmek, kullanıcı ile geliştirici arasındaki iletişimi sağlamak, tasarıma temel oluşturmak ve sistem isterlerini dokümanete etmek gibi amaçları sağlar [3]. Kavramsal model geliştirme ile başlayan ve model geliştirme, işletim, analiz adımlarını içeren kuşatıcı bir süreç Robinson [4] tarafından sunulmuştur.

Sistem tanımlamada belirli paradigmalara göre gösterimler ve modellenmeler gerçekleştirilir. Bu çalışmada birden fazla paradigmanın tümleştirildiği bir paradigma olarak Durum tabanlı Programlama Paradigması (DtPP) modelleme yaklaşımının esasını oluşturmaktadır. EtSiS kavramsal model bileşenlerini somut nesnel bileşenler olarak sunar. Her birinin bir etmen olarak tanımlandığı simülasyon modellerini, onlara

ait dünya algılarını (kavramlarını) ve karar kuramlarını temsil eden mantık yapılarını, birbirleri ile etkileşimlerini ve iş/görev isteklerini belirten olay tanımlarını, ilişkili olarak davranışlarını ve görev yapılarını içeren bir kavramsal model tanımı için bir dil grameri sunar [5], [6]. Etmen tabanlı bir sistemde kavramsal model ve kavramsal model doğrulama, karar yapıları ve etmen modeli iç dünya tasviri bilgi yapılarının doğrulanmasını da içerir. Dünya tasvirine ilişkin bilgi yapıları, etmenin çevresinden edindiği bilgileri zaman etiketli olarak, bilgi tabanında tutması ile gerçekleştirilir. İlgili karar algoritmaları kurallar (predicate) olarak yine bilgi tabanında tutulurlar [6]. Kavramsal modellemede kural setleri, algıladığı bilgilerle genişlettiği bilgitabanını kullanarak karar vermesi ve davranış yönetimi (işletme, sonlandırma, askıya alma vs..) yapması amacıyla kullanılır. Bu ilişkilendirme kavramsal dünya tasvirinin ve ontoloji tanımının bir boyutudur [7], [8].

## 2. EtSiS: Etmen tabanlı Simülasyon Sistemi

Geliştirilen kavramsal model ve senaryo doğrulama kontrolleri EtSiS dil yapısı esas alınarak gerçekleştirilmiştir. EtSiS genel amaçlı, multi-paradigma bildirimsel betik dildir [5]. Dilin tekrar kullanılabilirlik ve karşılıklı işletilebilirlik konusundaki ana vurgusu, multi-paradigma desteği ve modellemecilere sağladığı etmen programlama yaklaşımıdır. EtSiS varlıkbilimsel (ontolojik) betimlemesinde nesne yönelimli programlama paradigması (NYP), mantık programlama, etmen-tabanlı programlama paradigması [10], [11] ve cephe tabanlı (aspect oriented) programlama paradigmasını tümleştirir. Tümleştirmeyi durum tabanlı programlama olarak isimlendirdiği ve dünya betimlemesini, gramerini oluşturduğu bir paradigmada gerçekleştirir [5], [8].

EtSiS’de bir model  $M = \langle e_i, S, A, Aa \rangle$  olarak tanımlanır. Burada;

$e_i$ : Modelin diğer modellerden, çevreden veya kendi iç dinamiği ile ürettiği ve yine kendisine yönlendirerek aldığı olaylar setini gösterir.

S: Modelin yürütmekte olduğu davranışlarda, aktif olarak işletmekte olduğu aktif durumlar kümesini, bir diğer ifade ile, modelin içerisinde bulunduğu durumları gösterir.

A: Sahip olduğu davranışlar (B) kümesini, bir diğer ifade ile, davranış cephelerini (aspects) gösterir.

Aa: Aktif olarak işletebileceği cepheleri gösterir.

Bir davranış  $B = \langle G, e_i, S_b, e_o, \delta_{phases}, T_{phases}, D_{phases} \rangle$  şeklinde tanımlanır ve burada;

G: Davranış giriş kısıtını,

S<sub>b</sub>: Davranışı oluşturan ardışık durumlar kümesini,

e<sub>o</sub>: Davranışın ürettiği çıktı olaylarını temsil eder.

$\delta_{phases}$ ; Davranışın {Aktif, İptal edilmiş, Askıya alınmış, Tamamlanmış, Tekrar aktive edilmiş} olarak belirlenmiş fazlarına ilişkilendirilmiş fonksiyonları,

Tphases; Davranış fazlarına ilişkilendirilmiş davranış yönetmlerini (ilişkilendirildiği fazda ilişkilendirilen davranışları aktive eder, askıya alır, iptal eder, tekrar aktive eder)

Dphases; Davranış faz geçiş şartları tanımını gösterirler.

Davranışa ait bir durum;  $S_b = \langle G, \delta_{int}, \delta_{out}, \delta_{ext}, e_o, tmp \rangle$  olarak gösterilir ve G durum giriş kısıtını,  $\delta_{int}$  durum iç geçiş fonksiyonunu,  $\delta_{ext}$  harici geçiş fonksiyonunu ve  $\delta_{out}$  çıkış fonksiyonunu,  $e_o$ , durum çıktı olaylarını, tmp ise diğer davranışlar ile kurulan temporal ilişkileri ifade eder.

### 3. Kavramsal Model ve Senaryo Doğrulama

Kavramsal model ve senaryo doğrulama, model tabanlı bir doğrulama süreci olarak ele alınır [12]. Kavramsal model ve senaryo doğrulamasının her ikisi de geliştirme aşamasında çevrim içi olarak yürütülür. Her biri bir önerme (predicate) yapısında tanımlanan doğrulama kuralları, kullanıcılar tarafından tanımlanır, senaryo geliştirme ve kavramsal model geliştirme araçlarına bir yazılım bağımlılığı kurulmadan işletilirler. Kuralların ihlal edilmeleri durumunda “uyarı” veya “hata” tipinde çıktılar üretilir. Üretilen çıktıların mesajları ve fonksiyonel karşılıkları bir betik yapı ile tanımlanır.

Senaryo doğrulamanın literatürde kaplama analizi tabanlı olarak yürütüldüğünü [13] ve eğilimin kavramsal modelin kapsanmasını sağlayıcı test senaryolarının, UML tabanlı olarak otomatik üretiminin sağlanması yönünde geliştiği görmekteyiz [14]. Bu çalışmada, ek olarak senaryo doğrulamada hazırlanan senaryonun problem sahasına ilişkin analiz amaçlarını sağlama derecesinin ölçülmesi hedeflenmiştir.

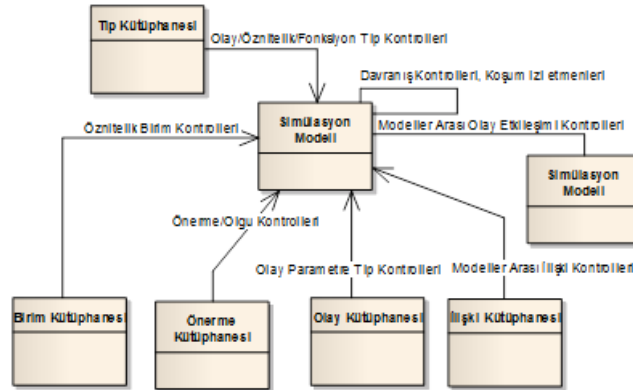
#### 3.1.Kavramsal Model Doğrulama

Kavramsal model, tanımlanan modelleri, davranışları, görevleri, model etkileşimleri ve çevrelerinin tanımlanması çerçevesinde gelişir. Dil teknolojilerinden bağımsız olan kavramsal model tasarımı [13] için, burada EtSiS (AdSiF) betik yapısı ile gösterim sağlanmıştır. Kavramsal model doğrulamada amaç, kurgulanan simülasyon dünyasının gerçek dünya ile olan uyumunu ve tasarımcının kavramsal tanımlamasını ne derece karşıladığını denetlemektir [15], [16], [17]. Kavramsal model doğrulaması aynı zamanda, kullanılan yazılım mühendisliği tekniklerinin geçerliliğini de doğrular. Bu çalışmanın temel yaklaşımını oluşturan model tabanlı senaryo ve kavramsal model doğrulama ve modellerin bir modelleme dili ile gösterimi, özellikle son yıllarda, yaygınlık kazanmıştır [18], [19]. Yapılan çalışmalarda, model tasarımlarında isterlerin sağlanma derecesinin doğrulanmasına dönük testlerin tasarlanması ve otomatikleştirilmesi hedeflenmektedir [20].

Modeller kavramsal dünyamızdaki aktörleri temsil ederler. Tümleşik modellerin ilişkiler yardımıyla birleştirilmesi ile de tümleşik modeller elde edilirler. Kavramsal modelin doğrulanmasında modeller merkezi bir noktada dururlar. Yapılan doğrulama

kontrollerinin bilgi kaynakları Şekil 1’de görülmektedir ve statik doğrulama yöntemi [15] olarak uygulanmaktadır. Yapılan kontroller aşağıda maddeler halinde listelenmiştir.

- Tip kontrolleri; Olay ve özneliklere değer atama tanımlamaları için tip uyumu kontrolleri gerçekleştirilir. Doğrulama modeller arası birim tutarlılığını, aynı ölçü sisteminin kullanılıp kullanılmadığını, kullanılan birimlerin birim kütüphanesinde var olup olmadığının kontrol edilmesini içerir. Farklı birim sistemlerinin kullanılması durumunda kullanıcı dönüşümler konusunda uyarılır. Birim kütüphanesi bu dönüşümlere ilişkin hesaplama tanımlarını içerir.
- Olay davranış kaplama; üretilen fakat bir davranış tarafından tetikleyici olarak kullanılmayan veya tetikleyici olarak kullanılmış fakat herhangi bir davranış tarafından üretilmemiş olayların kaplama kontrolleri gerçekleştirilir. Bir modelin diğer modellere ilettiği olayların iletiği modeller tarafından işletildiğine dair kontroller, model davranış tanımlamaları üzerinden yürütülürken, iletilen olayların parametre tip uyumlulukları olay kütüphanesinden belirtileri (declaration) üzerinden yürütülür. İletilen olayın parametre değer atamalarının doğru yapılp yapılmadığı veya varsayılan değer kullanılıp kullanılmadığı doğrulama testi kapsamında ele alınır.



**Şekil 1: Kavramsal Model Doğrulama Bilgi Kaynakları**

- Durum/Fonksiyon Kaplama; Durumlar tarafından kullanılmayan fonksiyonların, davranışlar tarafından kullanılmayan durumların kaplama kontrolleri yürütülür.
- Temporal ilişki kontrolleri; Temporal ilişkili olarak birbiri ile ilişkilendirilmiş davranışların varlığının kontrolleri yürütülür. İlişkili davranışların aktif bir davranış listesi içerisinde olup olmadıklarının kontrolü ile sağlanır.
- Yapay zeka kontrolleri (fact ve önerme kontrolleri); Önerme kütüphanesinde tanımlanan fakat durumlar tarafından kullanılmayan olgular (facts) ve davranışlar tarafından kullanılmayan önermeler (predicates) kontrol edilir.

- Davranış Listeleri (Aspect) kontrolleri; Tanımlanan fakat herhangi bir davranış listesi içerisinde yeralmayan davranışların var olup olmadığı kontrolü gerçekleştirilir.
- İşletim (koşum) İzi Etmeni Kontrolleri; Belirtimi yapılan işletim izi etmenlerinin durum davranış belirtileri içerisinde kullanılıp kullanılmadığının belirlenmesi amaçlanır. Önceki kontrol amacına benzer olarak, işletim izi etmeni belirtileri ile durumlar içerisindeki kullanımlarının kapsama analizleri amaçlanır.
- İlişki Kontrolleri; Modeller arasında kullanımı tanımlanmış olan ilişkilerin, olay iletimi ve/veya davranış yönetiminde kullanılıp kullanılmadığının kontrolleri sağlanır.
- Abonelik Kontrolleri; Bir model tarafından yayınlanan (published) özniteliğe, bir başka modelin abone olup olmadığı kontrol edilir.
- Mantıksal ifade kontrolleri; Tanımlanmış mantıksal ifadelerin durum kısıtı, davranış kısıtı, güdüm şartı veya temporal ilişki şartı olarak kullanılıp kullanılmadığı kontrol edilir.

Doğrulama boyunca yürütülen kaplama analizleri; fonksiyonel, parametrik ve kod kaplamalarını içerir.

### **3.2.Senaryo Doğrulama**

Senaryolar kavramsal modelden türetilmiş oluşumlar kümesi olarak değerlendirilebilir. Bir senaryoda varlık seçimleri, varlıkların görev seçimleri, görev parametreleri değerleri, model parametreleri değerleri ve bağlantılı tanımlamalar gerçekleştirilir. Bu kapsamda, bir senaryonun doğrulanması da senaryoyu oluşturan ana bileşenlerin doğrulanması ile sağlanır. Doğrulamada aşağıdaki başlıklar ele alınır;

- Gerekli Modellerin Kontrolü; Servis alınan modellerin, senaryoya servis alan modeller eklendiğinde, eklenip eklenmediğinin kontrolü yapılır (örnek; senaryoya radar eklendiğinde görüş hattı hesaplama sunucusunun varlığının kontrolü).
- İşletim Kayıtları; İşletim kayıt seçeneğinin seçili olması durumunda modellerin işletim etmenlerinin tanımlı olup olmadığı kontrolü yürütülür.
- Öznitelik Kontrolleri; Senaryo varlıkları arasında bağlantılı modellerin öznitelik değerlerinin doğru tanımlanıp tanımlanmadığı kontrolü yapılır (örnek; aynı birime ait telsiz frekans değerleri uyumu).
- İlişki Kontrolleri; Modeller arası gerekli ilişkilendirmelerin yapılıp yapılmadığı kontrol edilir (örnek; silah sistemi ile mühimmatların ilişkilendirilmesi).

Senaryoya eklenen bir model eğer bir başka modelden bir hizmet alacak ise bu durumda gerekli modelin senaryoya eklenmesi otomatik olarak gerçekleşir. Gerekli modelin senaryodan silinmesi durumunda kayıt aşamasında kural kontrolleri işlenir ve gerekli

uyarı kullanıcıya gösterilir. İlgili kural gösterimi Şekil 2’de gösterilmiştir. Örnek kuralda senaryoya bir radarın eklenmesi durumunda görüş hattı hesaplama sonucu modelinin senaryoya eklenmesi sağlanmaktadır.

**Şekil 2: Model Bağımlılıkları kontrolleri**

İşletim kayıtları her model için işletim izi belirtilerinin gerçekleştirilip gerçekleştirilmediği ve ilgili işletim izi etmenlerinin davranış yapıları içerisinde kullanılıp kullanılmadığının kontrollerini içerir. İşletim izi etmeni olarak belirtimi yapılmış olmasına rağmen herhangi bir durum belirtiminde ve ilişkili olarak davranış içerisinde kullanılmayan bir iz etmeninin varlığı kontrol edilir [6], [21].

**Şekil 3: Görev Doğrulama**

**Şekil 4: Öznitelik Atama Doğrulaması**

Senaryoya eklenen bir varlığa görev atamasının yapıp yapılmadığının kontrolü Şekil 3’de görülen betik kural gösterimi ile kontrol edilerek doğrulanır. Gösterimde senaryoya dahil edilen bir savaş uçağının görev rotası tanımı olması gerektiği kontrol edilmektedir. Benzer kural tanımları ile görev ve ilgili görev parametre kontrolleri yürütülür.

Senaryoya eklenen modellerin birbirleri ile olan öznitelik değer uyumlulukları kontrol edilerek gerekli uyarılar yapılır. Örnek olarak Şekil 4’de, aynı kuvvete ait telsiz frekanslarının farklı olamayacağını gösteren kural tanımlanmıştır. Senaryoya dahil

edilmiş olan bir modelin tanımlı ilişkilerinin kontrolü ile yapılan doğrulama adımı, ilişki doğrulaması olarak tanımlanır. Şekil 5’de görülen örnekte bir komutan modelinin ilişkileri kontrol edilmektedir. Herhangi bir iletişim modeli ile ilişkilendirilmemiş veya herhangi bir silah sistemi, komuta kontrol merkezi ilişkisi bulunmayan bir komutan modelinin, senaryoda bir işlev yürütmesinin mümkün olamayacağı temel fikri esaslı olarak doğrulama işlemi yürütülür.

**Şekil 5: İlişki Kuralları Tanımlama**

Geliştirilen kurallar için üretilecek mesaj ve seçilecek eylem tanımlaması gerçekleştirilir. Eylem tanımında **Commander\_MustHasRels** isimli kuralın sağlanmaması durumunda verilecek mesaj Şekil 6’de görülmektedir.

**Şekil 6: Kural Mesajı**

## 4. Örnek Uygulamalar

Çalışmada, EtSiS tabanlı olarak geliştirilen Hava Savunma Simülasyonu kavramsal modeli ve senaryo tanımlama uygulama örneği ile çözümün genel uygulanabilirliği gösterilmiştir.

### 4.1.Kavramsal Model Doğrulama

Hava savunma model kütüphanesi tanımlandıktan sonra yürütülen doğrulama işlemi ile Şekil 1’de görülen bilgi kaynakları uyumluluk kontrolleri gerçekleştirilir. Bölüm 3.1’de belirtilen kontroller gerçekleştirilir ve Şekil 7’de görüldüğü şekilde kavramsal model doğrulama sonuçları rapor olarak elde edilir.

Şekilde hatalı olay tanımlamaları, durum kaplama analizleri ve olgu (fact) kontrolleri görülmektedir.

### 4.2.Senaryo Doğrulama

Yer konuşlu aktif hava savunma simülasyonu için geliştirilmiş basit bir senaryo örnek olarak ele alınmıştır. Senaryo varlıkları ve aralarındaki ilişkiler aşağıda listelenmiştir.





ilişkilendirildiğinde uyarı kaldırılır. Füzelerin başlık sensörleri ve savaş başlıkları için uyarılar yapılır. Bu süreç senaryo tamamlanıncaya kadar devam eder. Eğer tanımlı kural bir hata mesajı olarak tanımlanmış ise, senaryonun işletimi esnasında kullanıcı uyarılır ve işletilmesi için ek bir onay istenir. Senaryo hazırlama esnasında yayınlanan hata ve uyarı mesajları Şekil 9'de görülmektedir.



Şekil 8: Senaryo Yerleşimi

Model	Description
F16-1	Hareketli platform için rota atanmamış. $\{(MovingPlatform : Atomic[BAF\_HasNotRoute] = true)\} \Rightarrow MovingPlatform\_MustHasRoute$
Komutan-1	Kural tanımındaki ilişki gereksinimi eksik veya geçersiz. $\{(Commander \rightarrow Relation[Bilgi Alir Min=1 Max=5 Dir=RRD\_LEFT]) \Rightarrow Commander\_MustHasRels$
Telsiz-1	Bir kablosuz haberleşme cihazı herhangi bir tarafta tek başına yer alamaz. $\{(WirelessDevice \rightarrow Count[Min=2 Max=2147483647 SideSensitive=true]) \Rightarrow WD\_CannotB...$

Şekil 9: Kural İşletim Sonuçları

Tablo 1: Senaryo Kuralları

Kural	Kontrol Edilen Modeller	Mesaj
Öznitelik Değerleri	Mavi tarafta komutan ve Radara ilişkilendirilen telsizlerin haberleşebilmesi için aynı frekansta olması gerekir.	Hata
	F16 için bir rota ataması yapılmış olması gerekir.	Uyarı
İlişki Kontrolleri	Komutan bir haberleşme cihazı ile veya bir sensör ile ilişkilendirilmiş olmalıdır.	Hata
	Bir sensör (radar) bir haberleşme cihazı veya bir komutan ile ilişkilendirilmiş olmalıdır.	Hata
	Laçner Füzeye sahip olmalıdır.	Uyarı
	Füze başlık sensörüne sahip olmalıdır.	Uyarı
	Füze savaş başlığına sahip olmalıdır.	Uyarı
	Senaryoda bir tek haberleşme cihazı yer alamaz (haberleşme sağlayacağı bir diğer haberleşme modeli vardır).	Hata

## 5. SONUÇ

Kavramsal model üzerinde yapılan bir doğrulama ile tasarlanan simülasyon dünyasının tutarlılığı ve kaplaması doğrulanırken, senaryo doğrulamasında kavramsal modelde tanımlanan amaçların kurgulanan senaryo ile sağlanıp sağlanamayacağının doğrulanması gerçekleştirilir. Kavramsal modelin kaplanması, varlıklar arasında iletilen tüm olayların, hedef modelde bir karşılığının olup olmadığının kontrolü, davranış durumları veya davranış faz fonksiyonları tarafından çağrılan atomik fonksiyonların karşılıklarının mevcudiyetinin kontrolü, varlık ilişkileri, tip ve birim kontrolleri gibi geniş bir kuşakta gerçekleştirilir.

Kural geliştirme betiği ve arayüzü, herhangi bir uygulama alanı ile sınırlandırılmamıştır. Bu yönü ile süreç hem kavramsal model doğrulması hem de senaryo doğrulaması için jenerik bir tanımlamadır ve kullanıcılar tarafından tanımlanan kurallar ile genişletilmeleri, farklı uygulama alanları için kullanımları mümkündür.

Sunulan çözüm ile modellemecilerin kavramsal modelleme süreci boyunca her aşamada doğrulama ile tutarlı bir kavramsal model geliştirmeleri, analistlerin ise doğru analizleri yapabilecekleri senaryoları oluşturmaları sağlanabilecektir.

## 6. KAYNAKÇA

- [1] S. Robinson, "Conceptual modelling for simulation Part I: definition and requirements," *J. Oper. Res. Soc.*, vol. 59, no. 3, pp. 278–290, 2008.
- [2] D.-J. V. D. Z. (Editor) Stewart Robinson (Editor), Roger Brooks (Editor), Kathy Kotiadis (Editor), *Conceptual Modeling for Discrete-Event Simulation*. CRC Press, 2010.
- [3] D. J. Van Der Zee and J. G. a J. Van Der Vorst, "Guiding principles for conceptual model creation in manufacturing simulation," *Proc. - Winter Simul. Conf.*, no. Shannon 1975, pp. 776–784, 2007.
- [4] S. Robinson, *Simulation: The Practice of Model Development and Use*, Second Edi. Palgrave Macmillan, 2014.
- [5] M. F. Hocaoglu, "AdSiF: Agent Driven Simulation Framework," *Hunstv. Simul. Conf. - HSC2005*, 2005.
- [6] Mehmet F. HOCAOGLU, "Agent based Simulation: Lecturer Notes," AUZEF, Istanbul University, Istanbul, 2014.
- [7] M. F. Hocaoglu, "Conceptual Model and Perdurantist Modeling with Reasoning," *ASIM 2014, ASIM Symp. Simul. Tech. (ASIM SST)*, 2014.
- [8] M. F. Hocaoglu, "Conceptual Model and Perdurantist Modeling with Reasoning," *Simulation Notes Europe*, vol. 24, no. 2, pp. 95–104, 2014.
- [9] Wooldridge, M., *An introduction to MultiAgent Systems*. John Wiley & Sons, Ltd, 2002.
- [10] Y. Shoham, "Agent-Oriented Programming." Technical Report STAN-CS-90-1335- Stanford University, 1990.
- [11] Y. Shoham, "Agent-Oriented Programming, Artificial Intelligence," *Artif. Intell.*, pp. 51–92, 1993.
- [12] M. F. Hocaoglu, İ. Genç, and A. A. Ergün, "AdSiF: Agent Driven Simulation Framework and Model Based Validation and Verification," in *2. Yazılım Kalitesi ve Yazılım Geliştirme Araçları Sempozyumu 2010*, 2010.
- [13] N. A. Karagöz, "Framework For Developing Conceptual Models Of The Mission Space For Simulation Systems," A PhD Thesis Submitted To The Graduate School Of Informatics Of The Middle East Technical University, Ankara, 2008.
- [14] D. P. M. Swagatika Dalai Arup Abhinna Acharya, "Test Case Generation For Concurrent Object-Oriented Systems Using Combinational Uml Models," *Int. J. Adv. Comput. Sci. Appl.*, vol. 3, no. 5, pp. 97–102, 2012.
- [15] D. A. Cook and J. M. Skinner, "How to perform credible verification, validation, and accreditation for modeling and simulation," in *CrossTalk*, 2005, no. 5, pp. 20–23.

## USMOS 2015 ODTÜ, ANKARA

- [16] D. K. Pace, “Modeling and simulation verification and validation challenges,” Johns Hopkins APL Tech. Dig., pp. 163–172, 2004.
- [17] “Defense Modeling and Simulation Office: Online M&S Glossary,” DoD 5000.59M. Washington: Department of Defense.
- [18] P. Fritzson, Principles of Object-Oriented Modeling and Simulation with Modelica 2.1, Wiley-IEEE. Wiley-IEEE Press., 2004.
- [19] E. Cellier, F. E., Kofman, Continuous System Simulation, Springer. Springer., 2006.
- [20] W. Schamai, “Model-Based Verification of Dynamic System Behavior against Requirements,” Linköping Studies in Science and Technology, Dissertation No. 1547, 2013.
- [21] M. F. Hocaoğlu, “AdSiF: Developer Guide” Agena Information & Defense System Ltd. [www.agenabst.com](http://www.agenabst.com), Istanbul, 2013.