

# İnsansız Hava Araçları İçin Otonom İniş Sistemi Simülatörü Tasarımı

Oğuzhan BAYRAKTAR, Faruk ÖZDEMİR, Ömer ÇETİN, Güray YILMAZ

Bilgisayar Mühendisliği Bölümü, Hava Harp Okulu, İstanbul, Türkiye  
[2409bayraktar@harbiyeli.hho.edu.tr](mailto:2409bayraktar@harbiyeli.hho.edu.tr), [2421ozdemir@harbiyeli.hho.edu.tr](mailto:2421ozdemir@harbiyeli.hho.edu.tr),  
[o.cetin@hho.edu.tr](mailto:o.cetin@hho.edu.tr), [g.yilmaz@hho.edu.tr](mailto:g.yilmaz@hho.edu.tr)  
 (Geliş/Received: 13.11.2011; Kabul/Accepted: 24.09.2012)

**Özet-** Bu çalışmada, bulanık mantık tabanlı otonom iniş gerçekleştirebilen bir İHA sistemi için simülatör dizaynına değinilmiştir. İHA sisteminin otonom iniş görevini yerine getirebilmesi için gerekli görev bilgisayarı çalışma kapsamında modellenmiştir. Temel uçuş kontrolünü sağlamak amacıyla faydalanılan uçuş bilgisayarı Matlab-Simulink ortamında yer alan Aerosim Blokset bileşeni dahilindeki altıncı dereceden serbesiyet denklemlerine dayalı Aerosonde İHA modelinden yaralanılarak uyarlanmıştır. Çalışmanın amacı otonom iniş esnasında karşılaşılabilecek hata durumlarında sistem operatörünün davranışlarının geliştirilmesi ve simülasyon ortamında başarısının ölçülebilmesi için bir simülatör ortamı sağlamaktır.

**Anahtar Kelimeler-** İnsansız hava aracı simülatörü, bulanık mantık, otonom iniş sistemi

## Autonomous Landing System Simulator Design for UAV

**Abstract-** In this study, fuzzy logic based autonomously landing system simulator for Unmanned Air Vehicle (UAV) has been mentioned. Mission computer has been modeled in the scope of work to concern autonomous landing mission of UAV system. In order to control basic flight attributes, flight computer has been adapted to Aerosonde UAV model which is based on sixth degree of freedom equations in Aerosim block set involved in Matlab-Simulink. The purpose of this work is to develop the operators behavior against the unexpected situations while UAV platform on landing and, provide a simulator environment for measuring success in simulation environment.

**Keywords-** Unmanned aerial vehicles, fuzzy logic, autonomous landing system

### 1. GİRİŞ

Bir İnsansız Hava Aracı Sistemi (İHA), bir operatör tarafından yerden kontrol ve idare edilen, üzerinde göreve ilişkin bir takım faydalı yükleri ve algılayıcıları taşıyabilen, bir takım görevleri otonom olarak yerine getirme yeteneğine sahip, uçan bir platform ve diğer yer donanımları olarak tanımlanabilir [1]. İHA sistemini diğer insanlı hava araçlarından ayıran en önemli iki nokta; platform üzerinde bir personel olmaması ve uçuşun genelinde faydalanılabilecek otonom yapılar ile desteklenmesi olarak sayılabilir.

Her ne kadar İHA sistemleri “insansız” olarak değerlendirilen yapılar olsa dahi operatör desteğine uçuşun çeşitli evrelerinde ve özellikle beklenmeyen acil durumların çözülmesinde ihtiyaç duymaktadırlar. Bu nedenle İHA operatörlerinin eğitimi ve bu beklenmeyen durumlara hazırlıklı olmaları gerekir. Bu amaçla ve bunun dışında çeşitli aviyonik testlerin gerçekleştirilebilmesi, otonom mimarilerin sınanması, geliştirilmesi gibi nedenlerden dolayı literatürde farklı teknikler ile

geliştirilmiş İHA simülatörü tasarım ve de uygulamalarına yönelik çalışmalar görülmektedir [2, 3, 4].

Çalışma kapsamında otonom yetenekleri iniş esnasında kısmen yerine getirebilen bir platform için personel eğitimi ve donanım ya da yazılım geliştirilmesi için bir simülatör tasarımına değinilecek ve iniş esnasında platformun otonom yeteneklerini yerine getirmesini kısmen modelleyebilecek bulanık mantık tabanlı bir yaklaşım geliştirilecektir. Amaç, en temel anlamda iniş esnasında otonom hareket eden hava aracının olası beklenmedik durumlar karşısında operatörün yapması gereken işlemleri benzetebileceği bir ortam geliştirmektir.

İHA üzerinde otonom yapıyı gerçekleştirmek için iki temel bilgisayar mimarisinden bahsedilebilir: Platformun temel emniyetli uçuş görevini aerodinamik prensipler doğrultusunda dinamik durum denklemlerini çözerek yerine getirmesine imkan sağlayan uçuş bilgisayarı ve göreve istinaden anlık olarak platformun davranışlarını değiştirecek komutları üreten ve uçuş bilgisayarına gönderen görev bilgisayarıdır. Çalışmada Matlab Aerosim

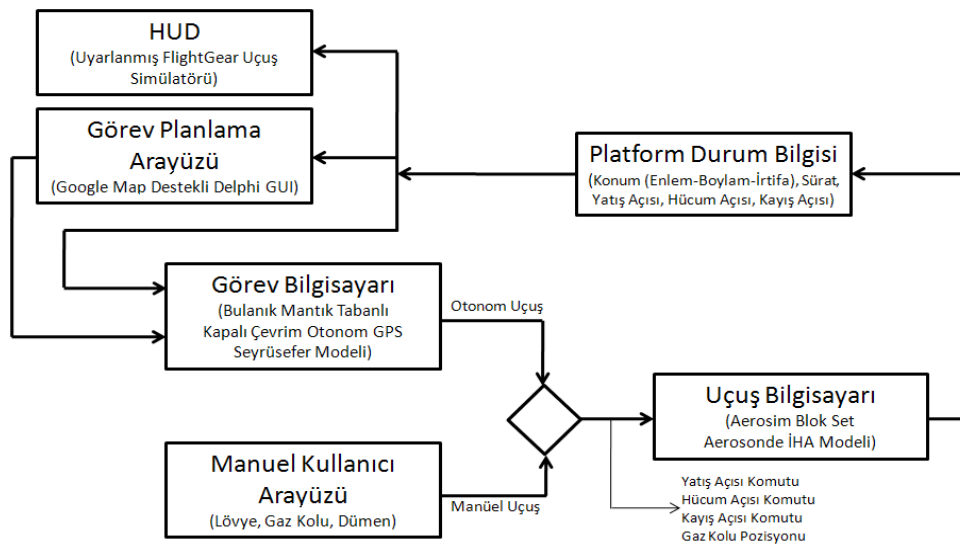
Blok Seti dahilinde yer alan ve altıncı dereceden dinamik serbesiyet denklemlerine dayalı olarak Aerosonde İHA platformu için modellenmiş yapıdan uçuş bilgisayarı olarak faydalanılmaktadır. Ayrıca bir uçuş bilgisayarı ve dinamik uçuş modeli geliştirilmeyecektir. Bu model dinamik olarak kontrol yüzeylerindeki değişimleri aerodinamik kurallar çerçevesinde modelleyebilmekte ve verilen komutlara istinaden platformun dinamik değişimlerini geri besleyebilmektedir. Görev bilgisayarı ise noktadan noktaya GPS tabanlı üç boyutlu seyrüsefer imkanı sağlayabilecek şekilde bulanık mantık denetleyicileri geliştirilmiş ve istenilen hedef noktaya erişilebilmesi için gerekli komutları üretebilecek otonom yetenekte bir sistem olarak teşkil edilmiştir. Genel simülasyon mimarisi Şekil 1’de gösterilmektedir.

Bir uçağın havada tutunabilmesi için yeterince süratli olması ya da kanat yüzeyinin mümkün olduğunca geniş olması gereklidir. Böylece platform üzerinde kaldırma kuvvetinin artmasına ve manevra yeteneğinin gelişmesine neden olacaktır. Havacılıkta uçuş esnasında karşılaşılan olağan dışı bir durumun ya da uygulanan hatalı bir manevranın emniyetli biçimde çözümü için mümkün olduğunca fazla irtifa gerekmektedir. Ancak hata yapılma olasılığı fazla olan kritik iniş sürecinde, hem sürat hem de irtifa düşük olduğundan, hatanın giderilme şansı ise oldukça düşüktür. Bu nedenle iniş, tüm uçuşun en riskli evresi olarak kabul edilebilir. Amerikan Hava Kuvvetleri Araştırma Laboratuvarının yaptığı açıklamalara göre Predator insansız hava aracının yaptığı kazaların %60,2 si operatör hatası faktöründen kaynaklanmakta ve bunların da % 60 ı iniş esnasında meydana gelmektedir [5]. Bu kazaların önlenmesi ya da azaltılması için yapılması gerekenlerden ilki operatör eğitimidir. Simülasyon eğitimleri ile olası benzer durumlar yaşanması ihtimaline karşı ucuz maliyetli tecrübe kazanan operatörün, hata yapma olasılığı azalacak ve bilgi seviyesi artacaktır. İkinci çözüm ise iniş gibi hatanın fazla olduğu, yüksek risk taşıyan durumlarda insan faktörünün minimuma indirilmesi ve platformun inişinin otonom olarak

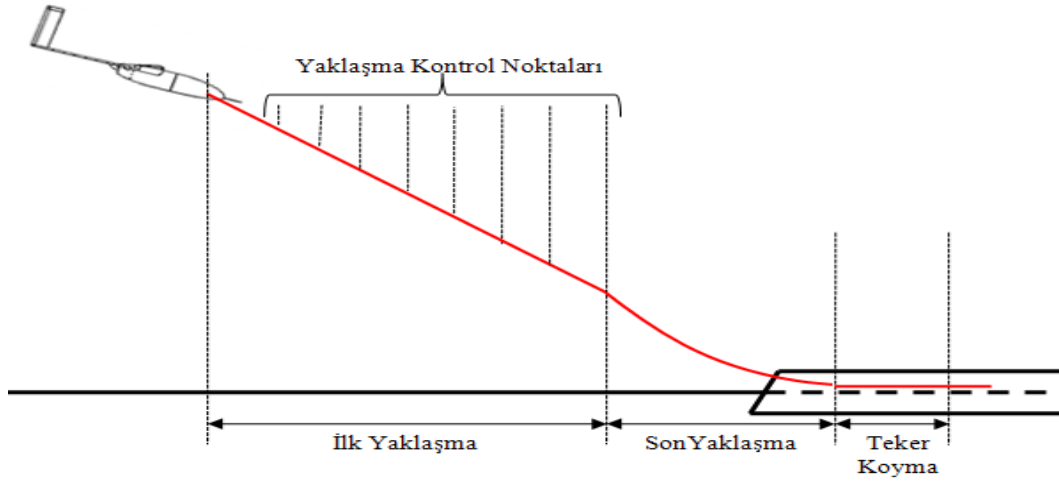
gerçekleştirebilmesidir [6]. Bu durum gelişmiş otonom sistemlerin üretilmesi ile mümkün olacaktır. Her ne kadar otonom yapılar uçuşun geneli için geliştirilmiş olsalar dahi, uçuş süresince insan faktörü mutlaka uçuş emniyetini etkileyecek şekilde yer alacak ve simülasyonlar ile insan eğitimi ihtiyacı doğacaktır. Bu ihtiyaç ise otonom iniş yapabilen ve aynı zamanda da personel eğitimi için kullanılacak olan simülasyon sisteminin tasarlanması ile çözümlenecektir.

Herhangi bir insan etkisi olmaksızın önceden belirlenmiş iniş paterninin insansız hava aracı tarafından uçulmasına otonom iniş denir. Otonom inişte İHA önceden belirlenmiş kontrol noktalarına göre bir seyrüsefer yardımcısını referans alarak (ILS, TACAN, D-GPS...vb.) ilk yaklaşma periyodu boyunca önceden belirlenmiş belirli sayıda yol kontrol noktasına insan müdahalesi olmaksızın sırasıyla Şekil 2’de gösterildiği biçimde ulaşır. Her kontrol noktasında anlık konum, irtifa ve sürat bilgilerinin istenen değerlerde olması arzu edilir. Başarılı bir iniş yapmak için kontrol edilmesi gereken üç ana değişken bulunur. Bunlar; piste göre platformun yatay konumu, düşey konumu ve süratidir. Son aşamada amaçlanan davranış pistin tam orta noktasına, istenilen sürati sabit tutarak teker koyabilmektedir.

Tasarlanan simülasyon sistemi İHA’nın otonom iniş sürecini olabildiğince gerçeğe yakın benzetmeyi amaçlamaktadır. Ancak bir iniş yardımcısı olmadan ve seyrüsefer yardımcısı olarak sadece GPS tabanlı yaklaşımın bulunduğu bu modelde otonom iniş belirli aralıklarla belirlenmiş yaklaşma kontrol noktalarına sırasıyla ulaşarak gerçekleştirilmiştir. Dolayısıyla otonom inişin sadece ilk yaklaşma evresi çalışma kapsamında otonom olarak icra edilmektedir. İlk yaklaşma evresinde platform mümkün olduğunca irtifa kaybedecek ve de süratini iniş için düşürmeye çalışacaktır. Hiç şüphesiz bu evre gerçekleşecek olağan dışı bir durumda platform için en riskli uçuş evresi olacaktır.



Şekil 1. Simülasyon mimarisi



Şekil 2. Otonom iniş pateni planlaması

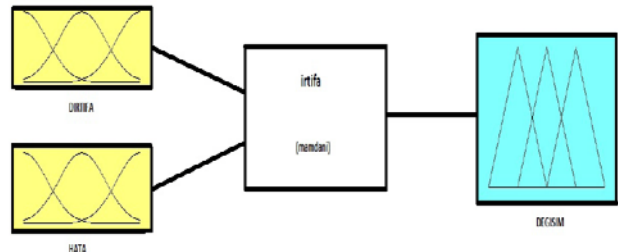
## 2. OTONOM İNİŞ SİSTEMİ TASARIMI

Personel eğitimi ve iniş sürecinde karşılaşılabilecek risklerin gösterilmesi amacıyla bir İHA otonom iniş sistemi simülâtörünün geliştirilebilmesi için kullanıcıya sunulacak İHA yer kontrol istasyonunu benzeten grafik ara yüzleri ile otonom iniş yapısının benzetilmesi gerekmektedir.

Çalışmanın bu bölümünde simülâtörün otonom iniş sistemi benzetimi tasarımına değinilecektir. Simülâtör kapsamında uçuş bilgisayarı MATLAB Simulink ortamında yer alan Aerosim Blok Set kapsamında bulunan Aerosonde dinamik uçuş modeli vasıtasıyla gerçekleştirilmiştir [7]. Otonom yapıyı tamamlayabilmek için bulanık mantık tabanlı kontrolörlerden faydalanılarak görev bilgisayarı yapısı oluşturulmuştur.

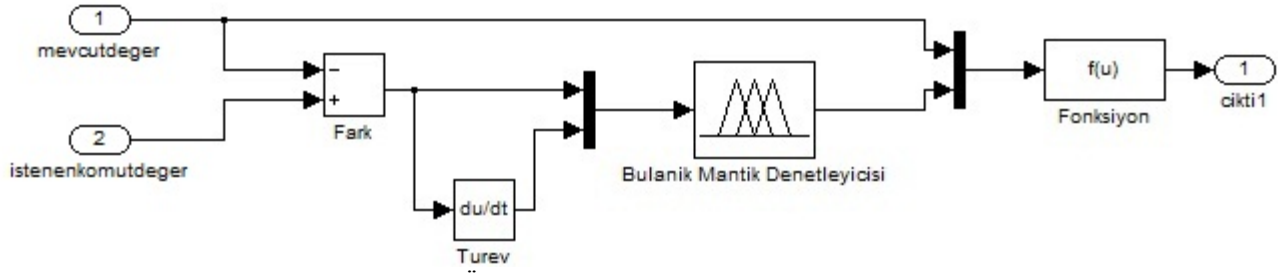
Geliştirilen simülâtör kapsamında otonom iniş sisteminin otonom olarak gerçekleştirilebilmesi için noktadan noktaya seyrüsefer yapısında bir takım değişiklikler yapılarak, simülâtör kullanıcıya otonom iniş gerçekleştirme imkanı sunulacaktır. Bu nedenle geliştirilen görev bilgisayarı kapsamında üç adet bulanık mantık denetleyicisinden (Fuzzy Logic Controller) faydalanılmıştır [8]. Bu kontrolörler İHA'nın enlem, boylam, irtifa ve sürat değerlerini kontrol ederek platformun verilen komutlar neticesinde istenilen pozisyon ve davranışı otonom biçimde sergileyerek uçuşmasını sağlamaktadırlar. Buradaki ilk iki denetleyici İHA'nın kontrol yüzeylerini (irtifa dümeni, istikamet dümeni ve kanatçıklar) hareket ettirmekte, diğer denetleyici ise gaz kolunu kontrol ederek istenilen sürat değerini sağlamaktadır. Bu denetleyicilerin amacı daha önceden belirlenen yol kontrol noktalarına sırasıyla uğrayarak, iniş için belirlenen piste otonom yaklaşmayı gerçekleştirmektir. Öncelikle hedefteki ilk noktaya gitmek

için hedef koordinatlar ve sürat bilgileri ara yüz ile sisteme alınır ve uçağın istenilen koordinata ulaşması için ilgili kumandalara gerekli komutlar verilir. Uçağın ilerleyişinde dış etkenlerden dolayı oluşacak (rüzgar, yer etkisi gibi) sapmaların denetimi ve istenilen pozisyonu sağlaması için yapacağı hareketler ile o anki koordinat bilgileri ve istenilen koordinat bilgileri görev bilgisayarıncı aradaki hata hesaplanarak kapalı bir çevrim halinde sürekli kontrol ünitesine yollanır. Kontrol ünitesinde bu hataları düzeltmek için gerekli komutlar üretilir ve uçağın ilgili kumandalarına kaç derecelik bir açı yapmaları gerektiği bildirilir.



Şekil 3. Örnek bulanık mantık denetleyicisi gösterimi (irtifa için)

Bu çalışma kapsamında otonom inişin modellenmesi amacıyla geliştirilen bulanık mantık denetleyicileri Şekil 3'de gösterildiği biçimde iki giriş ve bir çıkış mimarisinde "Mamdani" tipinde tasarlanmıştır. Temel bulanık mantık tabanlı kontrolörler ise Şekil 4'de gösterildiği şekilde komut değer ile mevcut değer karşılaştırılmasından elde edilen hata ve hatanın değişimi şeklinde oluşan girdilerden ve kontrol yüzeyinin davranışını belirleyen çıktıdan oluşmaktadır. Denetleyicilerin kural tabloları deneysel yaklaşımlar ile en temel şekilde oluşturulmuştur.



Şekil 4. Örnek otonom kontrol sistemi mimarisi

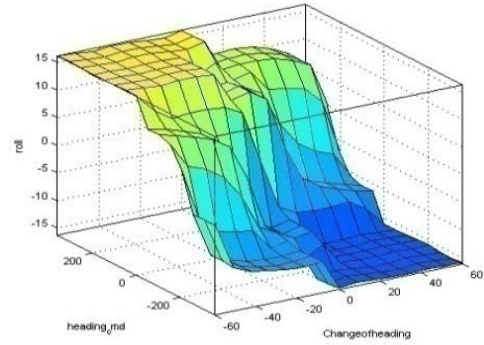
Uçuş başı kontrolü platformun anlık olarak bulunduğu uçuş başı ile gidilmek istenen komut uçuş başı değerlerini karşılaştırarak anlık ara değerler üretir. Bu değerler istenilen uçuş başı elde edilinceye kadar sürekli değişmekte ve istenilen uçuş başına giderek yaklaşılmaktadır. Uçuş başı kontrolünü gerçekleştiren bulanık mantık denetleyicisinin giriş değer aralığı -360 ile 360 arasında değer alabilmektedir. Çünkü hedeflenen uçuş başı ile anlık bulunulan uçuş başı arasında en fazla 359 derece hata oluşabilecektir.

İrtifa kontrolü anlık olarak bulunulan irtifa ile komut irtifa değeri arasında ara değerler üretir. Mevcut irtifadan, komut irtifa değeri çıkartılır ve hesaplanan hata değeri denetleyiciye giriş olarak gönderilir. Bulanık mantık denetleyicisinin ürettiği çıkış değeri uçuş bilgisayarına komut olarak gönderilir. Denetleyicinin girişi Şekil 6'da gösterildiği şekilde; hata miktarı olarak maksimum 3000 feet ya da minimum -3000 feet olarak tasarlanmıştır. Çünkü uçuş kumandaları her ne kadar bu manevraları platformun yapısal limitleri dahilinde gerçekleştirebilecek olsa dahi, iniş safahatında istenilen irtifa değerinin 3000 feet dışında bir noktada yer alması iniş sürecinin iptal edilmesini gerektiren bir durum olarak kabul edilmiştir. İrtifa kontrolüne ait çıktı parametresi değeri aralığı ise Şekil 6'da gösterildiği gibi -30 ile 30 değerleri arasında belirlenmiştir. Bu değer istenilen irtifa değişikliğini gerçekleştirebilmesi için İHA'nın hücum açısını belirtmektedir. Perdövites (stall) ya da yüksek hız (over speed) durumunu engellemek için bu şekilde sınırlandırılmıştır.

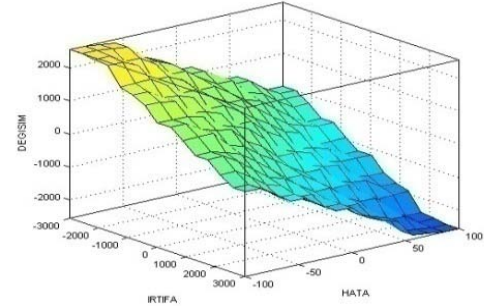
Geliştirilen simülasyon kapsamında sürat kontrolünü yapan bulanık mantık denetleyicisi platformun anlık olarak sahip olduğu sürat ile komut olarak verilen sürat değeri arasında ara değerler üretmektedir. Platformun o anki mevcut süratinden istenilen sürat değeri çıkartılır ve bulanık mantık denetleyicisine gönderilir.

Bulanık mantık denetleyicisinin ürettiği değerle istenilen değer toplanarak uçuş bilgisayarına komut olarak gönderilir. Sürat kontrolünü gerçekleştiren bulanık mantık denetleyicisinin giriş değer aralığı -100 ile 100 m/sn arasında farklı değerler alabilmektedir. Buradaki hata farkı için girilen aralık ise -10 ile 10 m/sn arasında yer alabilmektedir. Denetleyiciye ait çıktı parametresi değeri aralığı ise Şekil 6'da gösterildiği gibi -100 ile 100 m/sn değerleri arasında belirlenmiştir. Görev bilgisayarı kapsamında otonom inişin gerçekleştirilebilmesi için

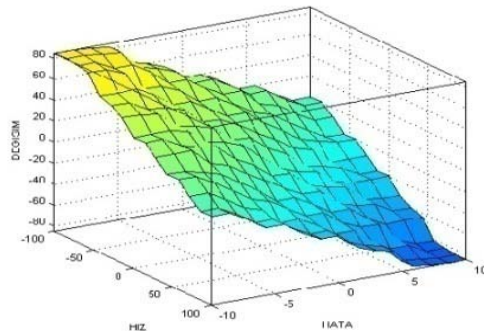
geliştirilen bulanık mantık denetleyicilerine ait yüzey diyagramları Şekil 5'de gösterilmektedir.



(a)



(b)

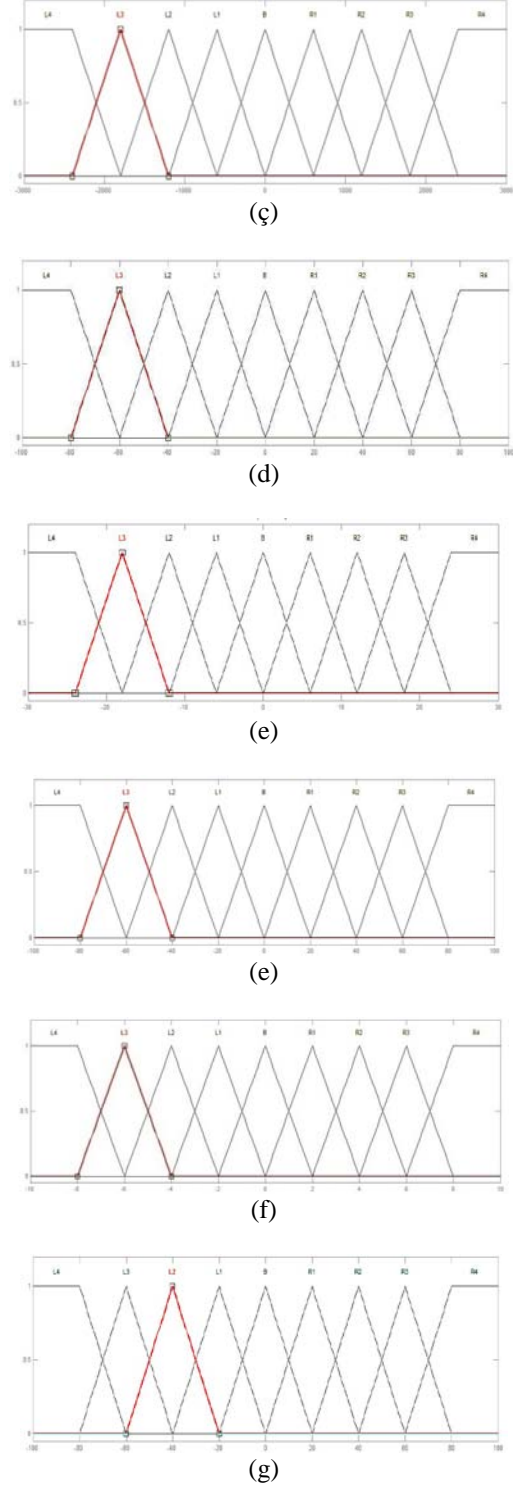
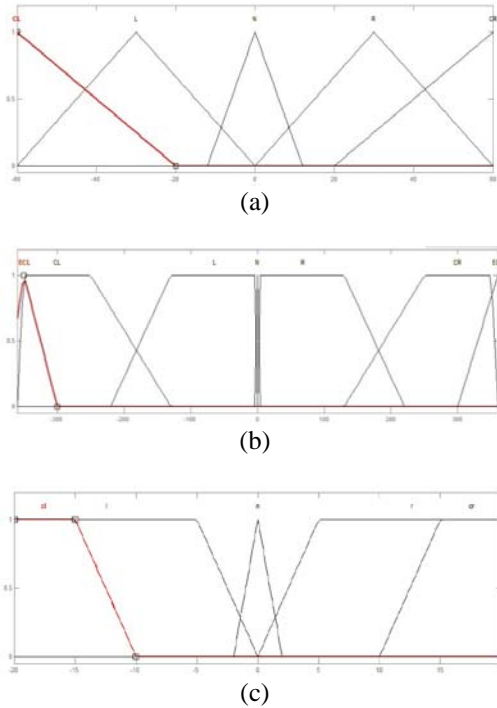


(c)

Şekil 5. Kontrol yüzey diyagramları (a) Uçuş Başı (b) İrtifa (c) Sürat

### 3. SİMÜLATÖR TASARIMI

Geliştirilen otonom iniş sistemini benzeten simülâtör yapısı daha önceki çalışmalarımızda geliştirilmiş olan noktadan noktaya seyrüsefer imkanı sağlayan [9] simülâtör sistemine entegre edilerek kullanılmaktadır. Geliştirilen otonom iniş modülü görev bilgisayarının bileşeni olarak tasarlanmış ve istenildiğinde devreye alınabilecek şekilde gerçekleştirilmiştir. Mümkün olduğunca mevcut İHA yer istasyonlarındaki mimariye yakın şekilde bir ara yüz tasarlanmıştır. Otonom inişin planlanması aşamasında, GPS tabanlı konum bilgilerinin kullanıcıdan alınması için “Delphi” programlama ortamında “Google Maps” içerikli görev planlama ara yüzü geliştirilmiştir. Matlab ortamında modellenmiş olan İHA platformunun görev bilgisayarına bu komut parametreleri TCP/IP protokolü ile oluşturulan bir bağlantı vasıtasıyla iletilmektedir. Aerosonde İHA platformu için Aerosim Blok Seti kapsamında geliştirilmiş ve bu çalışma kapsamında uçuş bilgisayarı olarak uyarlanan birimden alınan platformun davranışına ilişkin aerodinamik veriler (uçuş başı, yatış açısı, irtifa, sürat gibi) UDP protokolü ile açık kaynak kodlu FlightGear görsel uçuş simülâtörüne aktarılır [10]. Böylece FlightGear açık kaynak kodlu uçuş simülâtörü yazılımında yapılan uyarlamalar sayesinde İHA'nın yaptığı manevralar anlık olarak Head-Up Display (HUD) üzerinden görsel şekilde takip edilebilmektedir. Simülâtör sistemi otonom inişin gerçekleştirileceği meydan ve iniş ile ilgili yol noktalarının tanımlanacağı bir görev ara yüzünden, Şekil 7'de gösterilen İHA sisteminin HUD görüntüsünün yer aldığı bir ara yüzden, platformun dıştan görüntüsünün izlenebileceği bir dış görüntü ekranından ve Şekil 7'de yer alan platformun davranışlarının izlenebileceği ve çeşitli komutların girilebileceği bir komut ekranından oluşmaktadır.



Şekil 6. (a,b) Uçuş başı kontrolü girdileri (c) Uçuş başı kontrolü çıktısı (ç,d) İrtifa kontrolü girdileri (d) İrtifa kontrolü çıktısı (e,f) Sürat kontrolü girdileri (g) Sürat kontrolü çıktısı

Kullanıcı otonom inişin planlamasını gerçekleştirir ve inişin gerçekleştirilmesini HUD ve kontrol ara yüzü üzerinden takip eder. Platform istenilen parametre değerlerine, daha önceden sisteme tanımlanmış senaryolar kapsamında farklı nedenlerden (limit dışı rüzgar gibi

meteorolojik şartlar ya da motor arızası gibi platform arızaları gibi) dolayı ulaşamıyor ise uçuş kontrol listesi (Flight Checklist) kapsamında ilgili durumla ilgili verilen talimatları sırasıyla yerine getirmeye çalışır. Bu komutlar kontrol ara yüzü vasıtasıyla bazı sistemleri devreye almak ya da çıkarmak veya manüel kontrole geçerek talimatları uygulamak gibi farklı görevlerden oluşabilir. Senaryo dahilinde icra edilen tüm uçuş, iniş planlama aşaması dahil, zaman bilgisi ile birlikte icra edilen eylemler ve verilen komutlarla birlikte kayıt edilir. Simülasyon uçuşunun sonunda inişin başarılı ya da başarısız olarak sonuçlandırılması ile tüm uçuş bu kayıttan izlenerek operatörün eksik ya da hatalı komutları belirlenerek açıklanır.



(a)



(b)

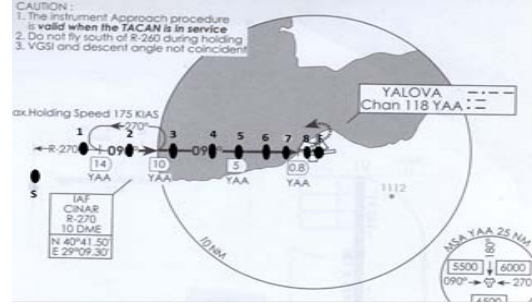
Şekil 7. (a) Delphi ortamında geliştirilen kullanıcı ara yüzü (b) FlightGear ortamında geliştirilen HUD tasarımı

Çalışma kapsamında İHA simülatörünü kullanan operatörün normal ve acil durum usullerini kavrayabilmesi ve herhangi bir acil durumda kullanabilmesi için bir de uçuş kontrol listesi (Flight Checklist) geliştirilmiştir. Amaç herhangi bir acil durumda, şayet ihtiyaç var ise, İHA'nın otonom uçuştan çıkarılıp manüel olarak kontrol edilmesine karar verilmesi ve uçuş bilgisayarına doğrudan entegre edilmiş olan konsollar yardımıyla manüel uçuşun sağlanmasıdır. Kontrol listesi düz uçuş, iniş gibi temel usulleri içermekte bunun yanında da acil durumlar için motor arızası, iniş takımı arızası, iletişim bağlantısının kopması gibi temel acil durum usullerini kapsamaktadır.

#### 4. ÖRNEK UYGULAMA PLANLAMASI

Çalışma kapsamında geliştirilen İHA iniş sistemi simülatörünün denenmesi için, Yalova Meydanı TACAN alçalma planı 08-26 pistine 270 radyal'den yaklaşarak Şekil 8'de gösterildiği biçimde gerçeğe yakın bir iniş planı oluşturulmuştur. Patern toplam on yol noktasından oluşmaktadır. Iniş paterni "S" ile ifade edilen noktadan başlar ve "F" noktasında son bulur. İHA toplam dokuz yol noktasına sırasıyla ulaşmayı amaçlayacaktır. Kullanıcı

"Google Maps"ten faydalanarak çalışma kapsamında geliştirilen GUI yardımıyla Yalova Meydanı'nın alçalma paternini oluşturmuş ve bu noktaların koordinatlarını otonom iniş sistemine girdi olarak göndermiştir.



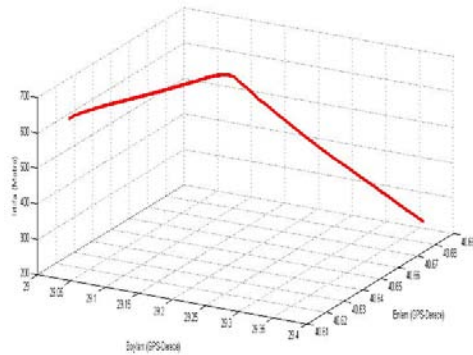
Şekil 8. Yalova TACAN alçalma kontrol noktaları

Şekil 8'de gördüğümüz gibi iniş paterni birbirini takip eden Tablo 1'de bilgileri yer alan dokuz kontrol noktasından oluşturulmuştur. Birinci nokta 2100 feet irtifada başlar ve dokuzuncu noktada 200 feet irtifada patern sonlanır. Bu kontrol noktalarına bakıldığında piste yaklaştıkça noktalar arasındaki sıklık artmakta ve irtifa değişikliği azalmaktadır.

Tablo 1. Örnek otonom iniş sistemi uygulama paterni yol kontrol noktaları

Nokta	Enlem	Boylam	İrtifa
S	40.613105°	29.041000°	2100
1	40.683431°	29.067579°	1800
2	40.683431°	29.130667°	1600
3	40.683431°	29.178893°	1300
4	40.683431°	29.221856°	1100
5	40.683431°	29.264052°	900
6	40.683431°	29.299377°	700
7	40.683431°	29.337762°	500
8	40.683431°	29.364080°	350
F	40.683431°	29.374133°	200

Tablo 1'de yer alan yol kontrol noktalarından oluşan iniş paterni, daha önceden tanımlanan kontrollerden meydana getirilmiş otonom iniş sistemi simülatörü vasıtasıyla oluşturulduğunda Şekil 9'da gösterilen patern elde edilmiştir.



Şekil 9. Örnek otonom iniş sistemi uygulama paterni sonucu

## 5. SONUÇ

Bu çalışma kapsamında genel bir İHA sistemine ait simülasyon uygulamasının bir bileşeni olan otonom iniş sistemi ve manüel iniş sistemine ait simülasyon modülü geliştirilmiştir. Çalışma kapsamında geliştirilen simülasyonun amacı otonom veya manüel iniş gerçekleştiren bir İHA platform operatörünün beklenmeyen durumlar karşısında yapması gereken işlemleri daha iyi kavraması ve tecrübe kazanmasıdır. Bu eğitimin gerçek bir platform ile icra edilmesi hem maliyet olarak yüksek hem de risk açısından oldukça tehlikelidir. Bu nedenle iniş esnasında karşılaşılabilecek olası beklenmedik durumlara karşı verilecek eğitimin gerçekleştirilebileceği en iyi ortam simülatördür. Otonom yeteneklerin simülasyon ortamında elde edilebilmesi için bulanık mantık tabanlı denetleyicilerden faydalanılmıştır. Geliştirilen simülasyon farklı tipte İHA sistemlerini modelleyebilecek nitelikte geliştirilmiştir.

Çalışma kapsamında bulanık mantık tabanlı denetleyicilerden, kararlı ve de hızlı yanıtlar üretebilecek şekilde otonom modüllerin benzetilmesi için faydalanılmıştır. Tasarım oldukça başarılı şekilde çalışma kapsamında uygulanmış ve de İHA sistemlerinin otonom iniş esnasındaki davranışları etkin şekilde benzetilmiştir.

Simülasyonun görsel çıktıları açık kaynak kodlu simülasyon yazılımları ile etkin şekilde desteklenmiş ve de yapılan arayüz tasarımı ile etkin şekilde kullanılabilmesine imkan tanınmıştır. Gerçekleştirilen arayüz çalışmaları benzetilmek istenen farklı platformlara ait simülasyonlar için uyarlanabilir şekilde tasarlanmıştır.

Bu çalışma özellikle, İHA sistemlerinin kullanıcı eğitimine yönelik, uçuşun farklı otonom olarak icra edilebilen evrelerinin benzetilebilmesi için platformdan bağımsız olarak geliştirilmiş olması açısından literatüre katkı sağlamaktadır. Çalışmanın bir sonraki evresinde İHA simülasyonunun benzer mimarilerden faydalanılarak otonom kalkış modülünün geliştirilmesi ve uçuşun tüm evrelerini kapsayacak şekilde yeteneklerin geliştirilmesi amaçlanmaktadır.

## KAYNAKLAR

- [1] T. Altunok, "Türkiye'nin İHA Serüveni", *Bilim ve Teknik Dergisi*, 517(38), 2010.
- [2] P. Kaňovský, L. Smrček, C. Goodchild, "Simulation of UAV Systems", *Acta Polytechnica*, 45(4), 109-113, 2005.
- [3] B. De Beer ve M. Lewis, "Lightweight UAV Simulation for use in Multi-Agent Human-in-the-Loop experiments", **Proceedings of the European Concurrent Engineering Conference**, EUROESIS, 51-56, 2007.
- [4] S. Ates, I. Bayezit, G. Inalhan, "Design and Hardware-in-the-Loop Integration of a UAV Microavionics System in a Manned-Unmanned Joint Airspace Flight Network Simulator", *Journal of Intelligent and Robotic Systems*, V.54, Issue 1-3, 359-386, 2009.
- [5] USAF Safety Center, **Predator Mishap Report**, 2004.
- [6] L. Doitsidis, K. P. Valavanis, N. C. Tsourveloudis, M. Kontitsis, "A Framework for Fuzzy Logic Based UAV Navigation and Control", **IEEE International Conference on Robotics & Automation**, New Orleans, LA, 2004.

- [7] M. Baldonado, C.-C.K. Chang, L. Gravano, A. Paepcke, "Aerosim, Aeronautical Simulation Block Set v1.1, Users Guide", [www.u-dynamics.com](http://www.u-dynamics.com), The Stanford Digital Library Metadata Architecture. *Int. J. Digit. Libr.* Page(s):108-121, 1997.
- [8] S. Kurnaz, O.Cetin, O.Kaynak, "Fuzzy Logic Based Approach to Design of Flight Control and Navigation Tasks for Autonomous Unmanned Aerial Vehicles", *Journal of Intelligent and Robotic Systems*, Volume 54, Numbers 1-3, 2009.
- [9] O.Cetin, M.Ç.Kaplan, A Aydın "Bulanık Mantık Tabanlı Yaklaşım ile Otonom İnsansız Hava Araçları için Yer Kontrol Sistemi Simülasyon Dizaynı", **USMOS 2009, 3. Ulusal Savunma Uygulamaları Modelleme ve Simülasyon Konferansı**, ODTÜ-Kültür Kongre Merkezi-Ankara, 17-18 Haziran, 2009.
- [10] Internet: FlightGear Open-source Flight Simulator, [www.flightgear.org](http://www.flightgear.org), Last Reached 12.Feb.2011.