

# Açık Kaynak Sürü İHA Benzetim Ortamı

Okan Aşık, Gökçe Uludoğan, H. Kübra Eryılmaz

Bilgisayar Mühendisliği

Boğaziçi Üniversitesi

İstanbul, Türkiye

Email: okan.asik,gokce.uludogan,haticce.eryilmaz@boun.edu.tr

**Özetçe** —İnsansız hava araçları (İHA) düşen maliyetleri sayesinde bir çok alanda kullanılabilmektedir. En önemli uygulama alanlarından birisi de sürü sistemleridir. Sürü sistemlerindeki en büyük güçlüklerden birisi sistem testleridir. Bu testler sürü İHA sistemleri için daha da zor olmaktadır. Bu sebeple sürü İHA sistemlerinin testleri genellikle benzetim ortamında yapılmaktadır. Bu benzetim ortamları genellikle belirli bir araştırma alanı ve araştırma grubu için tasarlanmaktadır. Bu çalışmada genel amaçlı benzetim ortamı olan Gazebo'yu kullanarak açık kaynak bir sürü İHA benzetim ortamı sunuyoruz. ROS ile desteklenen sistemimiz geliştirmeye açık ve modüler bir kıyaslama sistemi sunmaktadır. Genel bir benzetim ortamı kullanılmasıyla bu anlamda ilk çalışmalarındandır. Sistemimizi Teknofest 2018 yarışmasında test ettik. Yarışan takımların kodlarını ve yarışma senaryolarını sürü İHA sistemi ile birlikte açık kaynak olarak sağlamaktayız.<sup>1</sup>

**Anahtar Kelimeler**—Sürü İHA, benzetim ortamı, simülasyon, insansız hava aracı, ROS, Gazebo

## I. GİRİŞ

Bir çok faydası sebebiyle sürü sistemleri aynı işi yapabilen büyük ve karmaşık sistemlere tercih edilmektedir. Bu faydaları fazlalık, kolay değiştirilebilirlik, ucuzluk ve basitlik olarak sıralayabiliriz. Bir sürü sisteminde tasarlanan görevi yapmak için belirlenen alt görevleri yapabilecek birden çok birim bulunur. Arzalanı her hangi birimin yerine yenisi kolaylıkla yerleştirilebilmektedir. Her bir birim küçük ve basit olduğu için üretim maliyeti de oldukça düşük olur. Birimlere verilen alt görevler basit görevlerden oluşur ve bu alt görevlerin toplamı ana görevi oluşturur. Tüm bu faydalarına rağmen sürü sistemlerinin en önemli güçlükleri toplam görevin alt görevlere kolayca bölüştürülebilmesi ve sistemin testinin zorluğudur.

Benzetim ortamları, karmaşık sistemlerin testlerinde en çok kullanılan yöntemlerdendir. Belirli problemlere odaklanan araştırma grupları genellikle kendi benzetim yazılımlarını geliştirmektedir. Buna rağmen belirli problem alanlarında ortak benzetim yazılımları geliştirilmektedir. Bu benzetim ortamlarından birisi de Gazebo [1] benzetim ortamıdır. Gazebo, robotik çalışmalarında kullanılmak üzere geliştirilmiş açık kaynak bir platformdur. Gazebo kullanılarak gerçeğe yakın 3 boyutlu dünyalar oluşturmak ve bu dünyalarda robot algoritmalarını gerçeğe yakın şekilde test etmek oldukça kolay olmaktadır. Her ne kadar Gazebo'nun motor ve fiziksel yüzey etkileşimleri gerçek dünyadan farklı olsa da üst seviye karar ve strateji testleri için çok büyük kolaylıklar sağlamaktadır.

Gazebo son yıllarda insansız hava araçlarını (İHA) benzetmek için de kullanılmaya başlandı [2], [3]. Gazebo için geliştirilen İHA benzetim modülleri PX4 [4] ve Ardupilot [5] gibi açık kaynak İHA yazılım tabanları üzerine inşa edildiğinden genel kullanıma uygun değildir. Bu bilinen yazılım alt yapısını kullanmayan Gazebo'da hazır bulunan İHA modellerini kullanan açık kaynak alt seviye uçuş kontrolcüsü geliştirdik. Uçuş kontrolcüsünün yanı sıra görevleri belirleyen bir senaryo, algılama ve haberleşme modülleri geliştirildi. Ayrıca her görev için belirlenen kriterlere göre algoritmanın performansını ölçen bir puanlama modülü geliştirildi. Tüm bu sistemin yükünü çeşitli makinalara dağıtabilmek ve geliştirilmesini kolaylaştırmak için tüm modüller ROS [6] alt yapısı kullanılarak geliştirildi.

Sürü İHA stratejilerini test edebilmek için bir çok farklı görev seçilebilir ama seçilecek görevden bağımsız olarak çözülmesi hedeflenen sürü İHA problemleri bir çok görevde ortaklık göstermektedir. Bu ortak problemlerden bazıları dağıtık haberleşme, görev dağılımı, görev değişimi ve kaynak yönetimi şeklinde sıralanabilir. Biz geliştirdiğimiz benzetim ortamı için çoklu hedef takibi görevini seçtik. Bu problemi daha somutlaştırmak için görevi askeri çatışma bölgesinde görevlendirilen İHA'ların bölgeyi keşfetmesi ve bulduğu teröristleri takip etmesi olarak belirledik. Benzetim ortamında her bir görev senaryo olarak tasarlanmıştır. Senaryo, keşfedilecek alanın büyüklüğünü, alandaki binaların yerlerini, teröristlerin yerlerini ve hareket zamanlarını, İHA'ların maksimum irtifalarını, İHA'ların haberleşme mesafelerini ve İHA'ların yakıt miktarlarını belirler.

Geliştirdiğimiz sistem her senaryo için bir puan üretmektedir. Bu puan teröristlerin harekete başladıkları binaların tespit edilme süresine, keşfedilen alanın toplam alana oranına ve terörist takibi başarısına bağlıdır. Bu görevler için İHA'lar üzerlerinde bulunan kamera görüntüsünü ve benzetim ortamından sağlanan 3 boyutlu poz bilgisini kullanmaktadır. Kamera, ham görüntüler yerine görüş alanına giren nesnelerin bilgisini İHA kontrolcüsüne sağlamaktadır.

Geliştirdiğimiz benzetim ortamını Teknofest 2018 [7] Sürü İHA yarışmasında test ettik. Yarışma esnasında 4 takım 10 farklı senaryo üzerinde yarıştı. Her senaryoda farklı bir sürü İHA kısıtı sunuldu. Sunulan kısıtlar şu şekildedir; keşif alanının büyüklüğü, görevde kullanılacak İHA sayısı, maksimum haberleşme uzaklığı, terörist hareketliliği ve sayısı, ve İHA'ların yakıt miktarları. Bu şekilde her takımın senaryolardan aldığı puanlar takımlar arasında normalleştirilmiş ve toplam puan hesaplanmıştır. Yarışan takımların kodları da benzetim ortamı ile birlikte açık kaynak olarak paylaşılmaktadır. Bu şekilde yeni geliştirilecek yöntemler için hazır bir

<sup>1</sup><https://github.com/teknofest-suruiha/suruiha>

kıyaslama ortamı sunmayı hedefliyoruz.

## II. İLGİLİ ÇALIŞMALAR

Sürü sistemleri araştırmalarını, sistemin geliştirilmesinde kullanılan yöntem ve sistemin görevi olarak iki genel kategoride toplayabiliriz [8]. Sürü sistemlerinin geliştirilmesinde genellikle davranışsal ve öğrenen yöntemler kullanılmaktadır. Yöntemlerin analizinde küçük ve büyük ölçekli model analizleri kullanılmaktadır. Küçük ölçekli analiz tüm sistemi sürüdeki her birimin davranışı üzerinden değerlendirilirken, büyük ölçekli analiz ortaya çıkan toplam davranış üzerinden değerlendirme yapmaktadır. Küçük ölçekli analiz yaklaşımı her bir birimi detaylı modellediği için bu modellerin benzetim ortamında gerçekleşmesini ve sistemin değerlendirmesini mümkün kılmaktadır. Vaughan tarafından geliştirilen Stage benzetim ortamı mikro ölçekli modellemeye dayanan ve bine yakın tekerlekli robotu gerçek zamanlı çalıştırabilen ilk çalışmalardandır [9].

Benzetim ortamlarının en önemli avantajlarından birisi büyük ölçekli görev analizi yapmayı mümkün kılmasıdır. Tanımlanan göreve göre sürü sisteminin başarısı belirlenen ölçeklerle değerlendirilebilir. Bu sebeple benzetim ortamları sürü sistemlerinin analizde oldukça yaygın olarak kullanılmaktadır [9].

Sürü İHA sistemleri için gerçek dünya testleri oldukça zor olmaktadır. Bunun en büyük sebepleri; havada kalma sürelerinin kısa olması, yüksek gürültülü pozisyon sensörleri, kısıtlı hesaplama kabiliyeti ve haberleşme alt yapılarıdır. Bu sebeplerden dolayı sürü İHA sistemlerinin benzetim ortamı analizleri daha çok önem kazanmaktadır. Yapılan bir çok çalışmada benzetim ortamı analizleri sunulurken bu çalışmalarda hemen hemen her araştırma ekibi kendi araştırma konusu için geliştirdiği benzetim ortamını kullanmaktadır [10]. Bu çalışmada robotik araştırma grupları tarafından yaygın olarak kullanılan genel amaçlı yazılım ve benzetim ortamı ROS/Gazebo üzerinde bir sürü İHA karşılaştırma ortamı sunmaktayız. Bildiğimiz kadarıyla sürü İHA çalışmaları açısından bu sistem bir ilk olmaktadır.

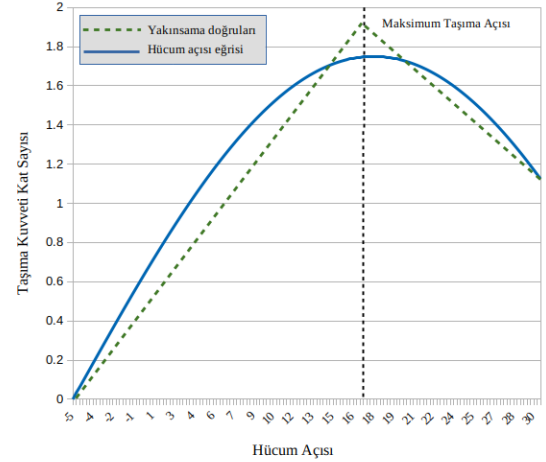
## III. AÇIK KAYNAK BENZETİM ORTAMI

Gazebo [1], robot sistemlerinin yazılımsal ve donanımsal benzetimini oluşturabilen açık kaynaklı bir benzetim ortamıdır. Gazebo, hem kullanıcı hem de programlar için fizik benzetimi, çeşitli sensörler ve arayüzler sunmaktadır. Robotik algoritmalarının test edilmesi ve robot tasarımı Gazebo'nun genel kullanım amaçlarındandır.

ROS (Robot Operating System) [6], robotların yazılım modüllerinin kolayca haberleşmesini sağlayan açık kaynaklı bir yazılımdır. ROS, sağladığı standart haberleşme araç birimi sayesinde geliştirilen yöntemlerin mevcut sistemlere kolayca uygulanabilmesini ve test edilebilmesini sağlamaktadır. ROS, içinde barındırdığı standart kütüphanelerin ve desteklediği robotların yanı sıra, geliştiriciler tarafından yazılmış kütüphanelere ve sürücülere de destek vermektedir. Gazebo, ROS'un sağladığı kolay haberleşme araç birimini kullanarak gerçek robotlar için geliştirilen yazılımların çok az değişiklikle benzetim ortamındaki robotlara uygulanmasını kolaylaştırmaktadır.

## A. İHA Benzetimi

Gazebo'nun aerodinamik benzetimi, akışkanlar mekaniğinin basitçe modellenmesi ile oluşturulmuştur. Akışkanlar mekaniği fazla karmaşık olduğu için, Gazebo bir nesnenin üzerindeki etkiyen kuvvetleri modeller ve kuvvetleri doğrudan nesnenin bağlantılarına uygular. Özellikle, taşıma ve sürüklenme kuvvetleri su altı ve aerodinamik araçlarının hareketinde önemli bir yere sahiptir. Taşıma kuvveti, bir akışkan içinde hareket eden bir cisim üzerinde akım doğrultusuna dik yönde oluşan kuvvetken, sürüklenme kuvveti hareket doğrultusunun tersi yönde etki eden kuvvettir [11]. Aerodinamik modellemeye diğer bir önemli unsur hücum açısıdır. Hücum açısı, nesnenin hareket yönü ile referans düzlem arasındaki açı olarak tanımlanır. Taşıma kuvveti ile hücum açısı arasındaki ilişki Şekil 1'deki kırmızı eğriyle gösterilmiştir. Hücum açısı belirli bir değere ulaştığında taşıma kat sayısı azalmaya başlar. Bu eğri her nesne için deneysel olarak hesaplanmaktadır. Gazebo, bu eğriyi iki doğru ile temsil eder. Bu temsil için kullanılan parametreler tasarlanan İHA'nın parametreleri olarak benzetim ortamına sağlanmaktadır.

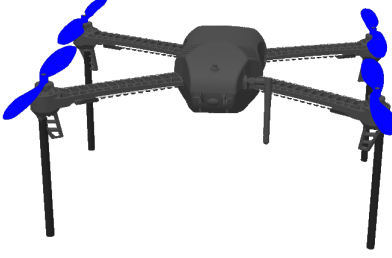


Şekil 1: Taşıma Kuvveti Kat Sayısı ile Hücum Açısı Grafiği. Gazebo Hücum Eğrisini Basitleştirip İki Doğru ile Temsil Etmektedir.

Benzetim ortamında bir tane sabit kanatlı ve bir tane de döner kanatlı olmak üzere iki İHA modeli sunulmaktadır; İris ve Zephyr. Bu modeller İHA'ların basitleştirilmiş aerodinamik modellemesine sahiptir. Bu sebeple ilgili kanatlara verilen dönme hareketine göre İHA'ların taşıma ve sürüklenme kuvvetleri benzetim ortamı tarafında hesaplanmaktadır. İris ve Zephyr Gazebo benzetim ortamında hazır olmasına rağmen alt seviye uçuş kontrolcülerinin bulunmamasıdır. Bu çalışmanın bir parçası olarak bu modeller için kontrolcüler geliştirilmiştir.

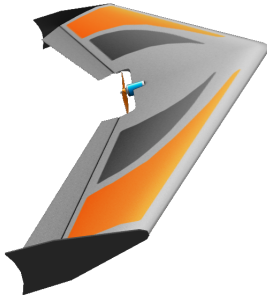
1) *İris Kontrolcüsü:* İris Kontrolcüsü, ROS konusundan gelen ilgili mesajları dinlemektedir. Bu mesajlar, doğrusal ve açısal olmak üzere iki alanda hız komutları içermektedir. Pervanelere verilecek ana hız, doğrusal z eksenini alanından alınır. Belirli bir hız komutuna ulaşan İHA havalandır. İHA'nın hareketini açısal hız alanındaki değerler belirler. İHA'yı x ekseninde hareket ettirmek için açısal y eksenine, y ekseninde hareket ettirmek içinse açısal x eksenine değer verilmelidir.

Kendi ekseninde hareket için ise, açısız z eksenini kullanılmaktadır. X eksenindeki hareket, İHA'yı y ekseninde belirli bir açıda (roll değeri) tutarak sağlanır. Y eksenindeki hareket de x eksenindeki benzer şekilde İHA'yı x ekseninde belirli bir açıda (pitch değeri) tutarak sağlanır. Kendi ekseninde hareket, İHA'nın z eksenindeki pozisyon girilen değere eşit olana kadar kendi etrafında dönmesi şeklindedir. X ve y eksenindeki hareketlerin aksine burada verilen açısal değer (yaw değeri) İHA'nın hızına katkı sağlamaz. Kontrolcü doğrusal hata kontrolcüsü kullanarak dört pervanesinin göreceli hızlarını ayarlamaktadır.



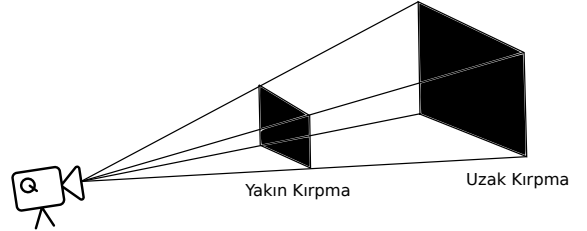
Şekil 2: Gazebo İris Çoklu Döner Kanatlı İHA Modeli

2) *Zephyr Kontrolcüsü*: Zephyr sabit kanatlı İHA için Gazebo benzetiminin aerodinamik prensipleri ile çalışan hazır model kullanılmıştır. Zephyr kontrolcüsü ROS konusundan gelen ilgili mesajlara göre İHA'nın hareketlerini belirlemektedir. Bu mesajlar, tip olarak İris kontrolcüsüne gelen mesajlarla aynıdır; yani, doğrusal ve açısal olmak üzere iki çeşit hız komutu içermektedir. Fakat bu alanlara verilen değerlerin kullanımı farklıdır. Doğrusal x eksenine verilen değer, sabit kanatlı İHA'nın pervanesinin hızını belirler. Açısal x ve y ekseninde verilen değerler kanatların kontrolü için kullanılır. Açısal x eksenindeki değer, yükselme ve alçalma açısını ayarlarken, açısal y değeri sağ veya sola dönüşü sağlar. Yükselme ve alçalma için sağ ve sol kanatların hücum açısı değiştirilir. Sağa ve sola dönüşler için kanatların birbirlerine göre olan hücum açısı değerleri artırılır veya azaltılır. Kontrolcü doğrusal hata düzeltmelerini kullanmaktadır.



Şekil 3: Gazebo Zephyr Sabit Kanatlı İHA Modeli

3) *Sanal Kamera Sensörü*: Her İHA'nın üzerinde yere bakan bir kamera bulunmaktadır. Bu kameranın alanına giren nesnelerin tespit edilip İHA'lara gönderilmesi için sanal bir kamera modülü geliştirilmiştir. Bu sanal kamera görüş alanındaki nesneleri İHA'nın kamera ROS konusuna mesaj olarak iletir. Bu mesaj, nesnelerin tiplerini, isimlerini ve pozisyonlarını içermektedir. Sanal kameranın çalışması için İHA'nın belirli bir yükselti aralığında olması gerekir. Bu aralık senaryoya göre değişmektedir. İHA'nın sanal kameranın çalışma yüksekliğinde olması durumunda kameranın algılayabildiği nesneler hesaplanır. Görüş alanı hesaplanırken kameranın görüş açısı, uzak ve yakın kırpma parametreleri dikkate alınır. Görüş alanı içerisinde kalan modeller hesaplanıp mesaj olarak İHA'ya gönderilir. Örnek kamera görüş alanı Şekil 4'de görülebilir. İHA sensör mesajlarını ilgili ROS konusundan alabilir.



Şekil 4: Örnek Görüş Alanı. Yakın Kırpma ve Uzak Kırpma Alanında Bulunan Nesneler Kamera Tarafından Görülebilir.

4) *Güç Yönetimi*: İHA'ların pille çalıştığı varsayılmıştır. İHA'ların havada kalma süresi sabit olmayıp, pilinin kapasitesine ve pervane hızına göre değişkenlik göstermektedir. İHA'ların güç tüketimi hareketli eklemlere uygulanan dış kuvvetlere göre hesaplanmaktadır. Dolayısıyla, bir veya birden çok ekleme fazla kuvvet uygulanması, pil tüketimini artırır.

Sabit kanatlı Zephyr İHA'sı 3 adet hareketli eklem içermektedir. Bunlardan bir tanesi arkadaki pervanesinin çalıştırılmasında, diğer iki tanesi ise sağ ve solda bulunan kanatların hareket ettirilmesinde kullanılır. Çoklu döner kanatlı İris İHA'sının ise 4 tane hareketli eklemi bulunmaktadır. İHA'nın havada kalabilmesi için tüm bu eklemlere belirli bir kuvvet uygulanması gerekmektedir. Algoritma 1'de gösterildiği gibi her simülasyon adımında hareketli eklemlere uygulanan kuvvetlerin toplanıp o İHA için belirlenen çarpım faktörü ile çarpılmasıyla pil tüketim miktarı belirlenir ve pilin kapasitesinden düşülür. İHA kontrolcüsü, İHA'nın toplam ve kalan pil durumunu ilgili ROS konusunu dinleyerek alabilir. İHA'nın havadayken pilinin bitmesi durumunda, İHA'yı kontrol etmek için gönderilen komutlar işlenmez ve İHA kontrolsüz şekilde yere çakılır. Pili bitmeden piste inebilen İHA'ların pilleri ilk andaki kapasitesine ulaşır ve tekrar kalkış yapabilir.

#### Algoritma 1 Güç Tüketiminin Hesaplanması

```

kuvvet ← 0
for all eklem do
    kuvvet ← kuvvet + |eklem.kuvvet| × katsayı
end for
pil ← pil - kuvvet

```

5) *Haberleşme Modülü*: İHA'ların iletişimini sağlamak için bir haberleşme modülü geliştirilmiştir. Her İHA ayrı bir

ROS modülü tarafından kontrol edilir, bu yüzden merkezi bir planlama algoritması ile kontrol edilemezler. Dağıtık şekilde eş güdümlü olarak çalışabilmeleri için haberleşme modülü üzerinden iletişim kurabilirler. Bu yarışma için İHA'ların haberleşme modülü dışında herhangi bir şekilde haberleşmeleri yasaktır. İHA'ların gönderdiği mesajlar belirli bir mesafedeki tüm İHA'lara haberleşme modülü ile yayın yapılır. Özellikle bir İHA'ya mesaj göndermek mümkün değildir. İHA'ların haberleşme mesafesi senaryoya özgüdür, yani senaryodan senaryoya değişmektedir. Bir İHA mesaj yayınlamak istediğinde ilgili ROS konusuna mesaj gönderir. Bu mesaj, *gönderici* ve *mesaj* alanlarını içermektedir. Gönderici alanı İHA'nın adından, mesaj alanı ise gönderilmek istenen mesajdan oluşmaktadır. Haberleşme modülü, bir mesaj aldığı anda mesajı gönderen İHA'nın mesaj yayınlama alanındaki tüm İHA'ları hesaplar ve ardından bu İHA'ların ilgili ROS konularına bu mesajı iletir. Her İHA'nın mesaj alım ROS konusu farklıdır ve her biri kendi ROS konusunu dinlemektedir. Gönderilecek mesajın uzunluğu ve içeriği ile ilgili herhangi bir kısıtlama bulunmamaktadır.

### B. Senaryo Görevleri ve Değişkenleri

Geliştirdiğimiz sistemde ana görev askeri operasyon alanındaki teröristlerin yerlerinin tespiti ve takibidir. Bu görevin detayları ve zorlukları senaryo parametreleri ile belirlenmektedir. Bu parametreleri şu şekilde sıralayabiliriz:

- Senaryoda kullanılacak İHA sayısı.
- Senaryoda bulunan bina sayısı ve binaların yerleşimi.
- Senaryoda bulunan terörist sayısı ve hareket patikaları.
- İki İHA'nın haberleşebileceği maksimum uzaklık.
- Senaryoda keşfedilmesi gereken toplam alanın boyutu.
- İHA'ların kalkış ve iniş yapacağı pistin alanı ve pozisyonu.
- İHA'ların pil kapasiteleri.
- İHA kamerasının algılamasını sağlayan yükseklik aralığı.

Belirlenen ana görev alt görevlere bölünüp performans ölçümü bu alt görevlerin değerlendirilmesiyle yapılmaktadır. Teröristler senaryoda belirlenen zamanda bir binadan çıkıp harita üzerinde başka bir binaya hareket etmektedir. İHA'lar teröristlerin ilk hareket zamanından itibaren çıkış binasını tespit etmekle görevlidirler. Ayrıca tespit ettikleri teröristlerin hareketlerini mümkün olduğunca takip etmekle görevlidirler. Bu görevi yaparken de mevcut alanı en iyi şekilde keşfetmeleri beklenmektedir. Örnek bir senaryonun ekran görüntüsü Şekil 5'de görülmektedir.

### C. Performans Ölçümü

Yarışmacıların performansları senaryolardan alınan puanların ölçeklendirilmesi ile hesaplanır. Teröristlerin yerinin tespiti ve teröristlerin takibi görevlerinden alınan puanlar 0.4 katsayısıyla, alan taramadan alınan puan ise 0.3 katsayısıyla çarpılıp toplanarak toplam puan 100 üzerinden hesaplanır. Senaryo görevlerinin puanlarının hesaplanması ise şu şekildedir:

1) *Teröristlerin Yerinin Tespiti*: Yarışma senaryolarında benzetimin belirli bir anında bir veya birden çok terörist bir binadan hareket etmeye başlayacaklardır. Bu hareket anında İHA'lar kullandıkları sanal kamera sensörleri ile teröristlerin harekete başladıkları binayı tespit edip yer kontrol sistemine bildirmeleri gerekmektedir. Yanlış bildirimler takımların ceza puanı almasına yol açacaktır. Alınan cezalar takımın aldığı toplam puan ile çarpılıp doğru tespit ile aldığı puanın düşmesine yol açacaktır. Puan hesaplaması teröristler harekete geçtikten sonra başlayacak ve ne kadar erken tespit edilirse alınan puan o kadar yüksek olacaktır. Puan hesaplaması şu şekilde olmaktadır:

$$zf = mz - ihz \quad (1)$$

$$puan = \left\lceil \frac{(tts - zf) \times 90}{tts} \right\rceil + tp \quad (2)$$

$$puan = puan \times [1.0 - (ytp \times yts)] \quad (3)$$

Denklemden geçen kısaltmaların anlamları şu şekildedir:

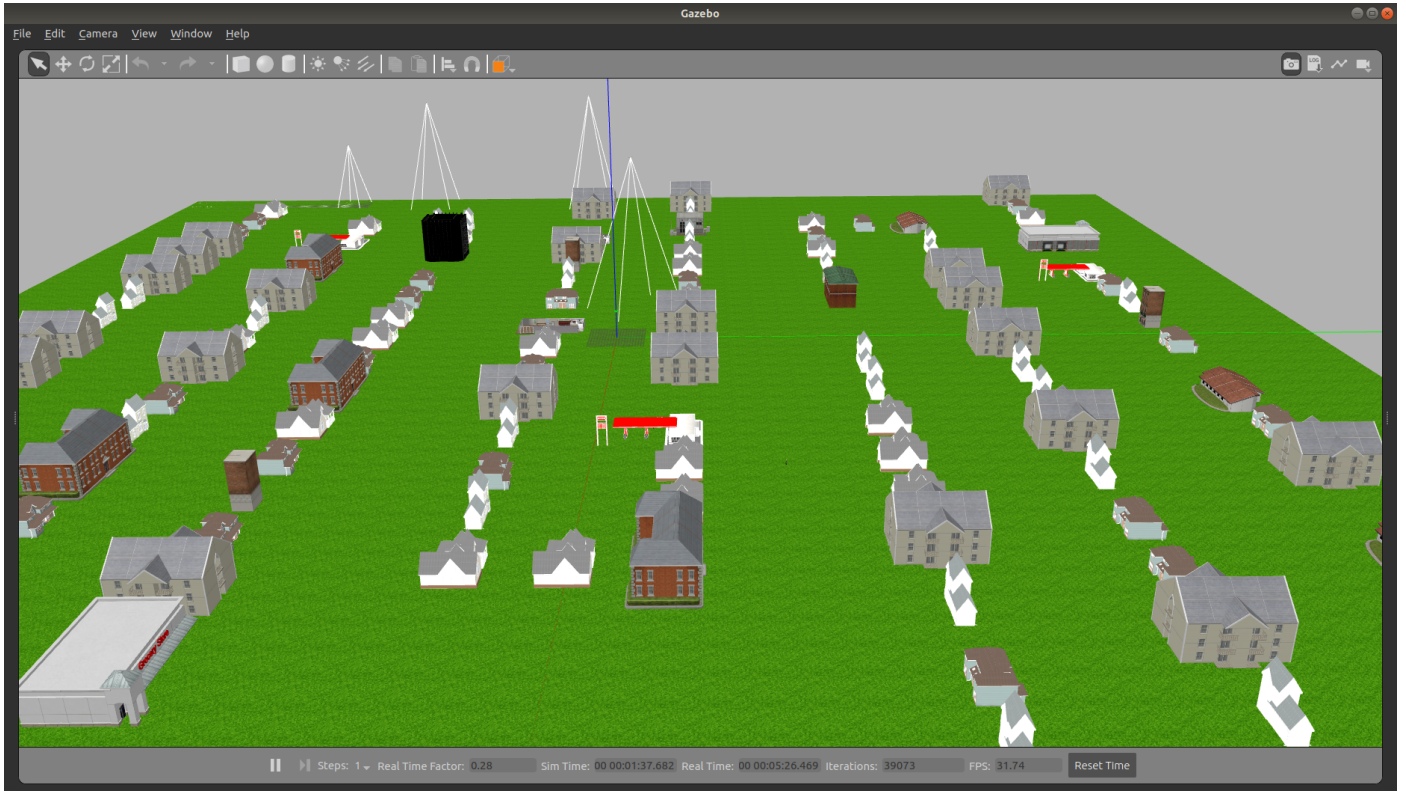
- *mz* teröristlerin görüldüğünü bildiren mesajın geliş zamanını göstermektedir.
- *ihz* teröristlerin ilk hareket zamanını göstermektedir.
- *zf* teröristlerin binadan hareketlerinden itibaren İHA'lar tarafından fark edilene kadar geçen süreyi göstermektedir.
- *tts* teröristlerin tespit edilebileceği toplam süreyi göstermektedir.
- *tp* taban puanı ifade etmektedir.
- *ytp* yanlış tespit puanını göstermektedir.
- *yts* yanlış tespit sayısını göstermektedir.

Eğer İHA sistemi hiç bir şekilde bina tespit bildiriminde bulunmazsa *tp* ile ifade edilen puanı alacaktır. Eğer bildirimde bulunursa yaptığı bildirimin toplam süreye oranı miktarında 0 ile 100 arasında bir puan alacaktır. Eğer yanlış bildirim olursa her bir yanlış bildirim doğru bildirim puanından *ytp* çarpımınca azalmasına sebep olacaktır. Teknofest 2018 yarışmasında *tp* 10 ve *ytp* 0.1 olarak kullanılmıştır.

2) *Teröristlerin Takip Edilmesi*: Teröristler hareket etmeye başladıktan sonra varacakları yere kadar İHA'ların teröristleri takip etmesi beklenmektedir. Takip esnasında sanal kamera tarafından yer üzerinde pozisyonu alınan her teröristin yer kontrol sistemine bildirilmesi beklenir. Alınan mesajlar yer kontrol sistemi tarafından her terörist için ayrı ayrı saklanır ve o ana kadar gelen mesajlardan bir patika oluşturulur. Teröristlerin ilk hareket anından varacakları yere kadar izledikleri patikalar kaydedilir. Daha sonra bu iki patika arasındaki fark dinamik zaman bükmesi algoritmasıyla [12] hesaplanır. Bu fark ne kadar azsa takım o kadar yüksek puan alır. Bu patika farkı hesaplaması senaryoda takip edilmesi gereken tüm teröristler için şu şekilde hesaplanır:

$$puan = \frac{1}{TS} \sum_{terorist_i} \frac{(maxMaliyet_i - maliyet_i) \times 100}{maxMaliyet_i} \quad (4)$$





Şekil 5: Örnek Bir Senaryonun Ekran Görüntüsü

Denklemden geçen kısaltmaların açıklamaları şu şekildedir:

- *TS* toplam terörist sayısını göstermektedir.
- *maxMaliyet* her teröristin simülasyonda takip ettiği gerçek patika vektörünün uzunluklarının toplamına eşittir. Bu aslında teröristin sürekli (0, 0) noktasında bulunduğunu söyleyen bir takımın performansına eşittir.
- *maliyet* her terörist için hesaplanan benzetim ortamında kaydedilen gerçek patika ile İHA'ların bildirdiği patika arasındaki farkı ifade eder. Takip başarısı yüksek bir İHA için bu değer küçük olacak. Başarısız bir İHA takımı için bu değer büyük olacaktır. Bu değer dinamik zaman bükme algoritmasıyla hesaplanmaktadır.

3) *Keşfedilen Alan*: İHA'ların üzerinde bulunan sanal kameranın yerde taradığı alanlar hesaplanır ve bu alanlar simülasyondaki haritadan oluşturulan ayrı bir harita üzerinde işaretlenir. Takımın gözlemlenen alan puanı bu gözlemlenen alanın toplam alana oranı olarak 0 ile 100 arasında bir değer olarak hesaplanır. Bir bölge birden çok İHA tarafından tekrar tekrar gözlemlense dahi puana etkisi olmamaktadır. Puan hesabı için bir bölgenin bir kere herhangi bir İHA tarafından gözlenmesi yeterli olmaktadır. Ayrı harita benzetim ortamındaki alanın çözünürlük parametresine göre bölütlenmesiyle oluşur. Mesela 1000 metreye 1000 metre genişliğindeki bir alanı, çözünürlük parametresi 0.5 olacak şekilde bölütlersek 2000'e 2000 boyutunda bir matris ile (yani 4 milyon hücre ile) gösterebiliriz. Her adımda İHA'ların sanal

sensörlerinin gördüğü alanlar hesaplanıp ayrı haritada ilgili hücreler işaretlenir.

#### IV. TEKNOFEST 2018

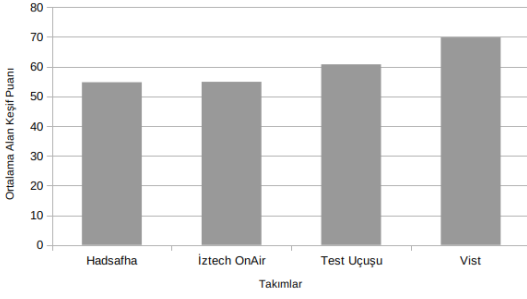
Geliştirdiğimiz benzetim ortamını Teknofest 2018 Sürü İHA yarışmasında test ettik. Yarışmaya 34 başvuru yapıldı. Bu başvurulardan 10'u önelemeyi geçti. Ön elemeyi geçenlerden dört takım yarışmaya katıldı.

##### A. Takımlar

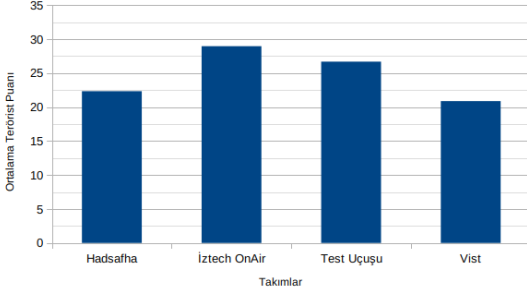
Yarışmaya katılan takımların kodları benzetim ortamı ile birlikte karşılaştırma amaçlı olarak paylaşılmaktadır [13]. Takımlar genellikle yöntemlerini alan taramayı maksimize edecek şekilde seçmişlerdir. Bunda diğer görevlerin daha zor ve ölçmesinin zor olması da rol oynamıştır. Eniyi şekilde teröristleri takip edecek bir algoritmanın, verilen bir senaryo için herhangi bir fırsatçı algoritmadan daha kötü performans gösterme ihtimali vardır. Diğer yandan teröristlerin çıkış binalarını tespit etmeye çalışması yanlış tespitlerden gelebilecek ceza puanları sebebiyle riskli görülmektedir. Tüm bunlara rağmen takımlar temel seviyede verilen görevleri yapmayı başarmıştır.

##### B. Senaryolar

Yarışmada toplamda 10 tane senaryo kullanılmıştır. Her bir senaryoda yarışma kısıtlarını değiştirip geliştirilen sistemlerin performanslarının gerçekçi testinin yapılması hedeflenmiştir. Dört senaryoda haritanın boyutu 1000 metreye 1000 iken altı senaryoda 500 metreye 500 metredir. Sekiz senaryoda



Şekil 6: Takımların Ortalama Alan Keşif Başarısı



Şekil 7: Takımların Ortalama Terörist Takip Başarısı

altı İHA kullanılmasına izin verilirken senaryoların ikisinde 4 İHA kullanılmasına izin verilmiştir. Tüm senaryolar için sabit kanatlı veya döner kanatlı İHA seçimi takımların inisiyatifine bırakılmıştır. Verilen senaryoların ikisinde belirlenen toplam benzetim süresinde tek pil kullanmak İHA'ların görevlerini tamamlamasına yetmeyecek şekilde pil kapasiteleri belirlenmiştir. Senaryoların iki tanesinde haberleşme mesafesi 100 metre olarak belirlenmiş bir tanesinde 50 metre olarak belirlenmiş ve geri kalan 7 tanesinde 10000 metre belirlenerek harita üzerinde kalan İHA'ların her zaman haberleşmesi mümkün kılınmıştır.

### C. Sonuçlar

Teknofest 2018 yarışmasında toplam 10 senaryodan topladığı 84.72 puanla Vist takımı birinci, 82.88 puanla Test Uçuşu takımı ikinci, 76.66 puanla İztech OnAir takımı üçüncü olmuştur. Takımların alan keşfetme ve terörist takibinden aldığı puanlar Şekil 6 ve 7'de görülebilir. Takımların başarısını belirleyen ana unsurun alan keşif başarısı olduğu görülmektedir. İztech OnAir takımı terörist takibinde en başarılı takım olmasına rağmen alan keşfindeki düşük performansı sebebiyle yarışmayı üçüncülükle bitirmiştir. Genel olarak takımların toplam alanın %60'a yakını keşfedebildiğini görüyoruz. Takip başarısı ise ortalama %20-%30 aralığında kalmıştır. Terörist yeri tespitinde sadece İztech OnAir takımı 3 senaryoda tespit mesajı göndermiş bu mesajlardan 2 senaryoda gönderilenler yanlış tespit olduğu için takımın bu iki senaryodan sıfır almasına yol açmış. Bir senaryoda da doğru şekilde tespit yapabilmemiş ama çok geç yaptığı için taban puandan sadece 1 puan fazla alabilmiş. Diğer takımlar tespit etmeye çalışıp puanlarını düşürme riskini göze almamışlardır. Ayrıntılı bilgiye yarışma sonuçlarını yayınladığımız web sayfasından [14] ulaşılabilir.

## V. SONUÇ VE GELECEK ÇALIŞMALAR

Geliştirdiğimiz sistem çok amaçlı bir benzetim ortamı üzerinde çalışan ilk sürü İHA benzetim ortamıdır. ROS ve Gazebo dışında bir İHA yazılım alt yapısına bağımlılığı yoktur. Gazebo'nun yeni geliştirilen aerodinamik benzetim modelini kullanmaktadır. Sistem Teknofest İstanbul 2018 yarışmasında test edilmiş ve kod örneklerinin ve senaryoların bulunduğu bir kıyaslama ortamı olarak sunulmuştur.

Gelecekte sisteme farklı görevleri ekleyerek daha genel bir kıyaslama alt yapısı sunmayı hedefliyoruz. Mevcut senaryolarda yeryüzünün tamamen düz ve kare olduğu varsayılmıştır. Gelecekte, yükseklik haritalarıyla eklemeyi planlıyoruz. Ayrıca mevcut sistemde tüm sensörler gürültüsüz şekilde veri sağlamaktadır. Senaryo parametrelerine gürültü parametresini ekleyip senaryo bazlı sensör gürültüsünü kontrol edebilmeyi hedefliyoruz.

## KAYNAKÇA

- [1] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*(IEEE Cat. No. 04CH37566), vol. 3. IEEE, pp. 2149–2154.
- [2] J. Meyer and S. Kohlbrecher. (2014) Hector quadrotor simulation. [Online]. Available: [http://wiki.ros.org/hector\\_quadrotor](http://wiki.ros.org/hector_quadrotor)
- [3] Dronecode.org. (2019) Px4 gazebo simulation. [Online]. Available: <https://dev.px4.io/en/simulation/gazebo.html>
- [4] Dronecode. (2019) Px4 autopilot. [Online]. Available: <https://px4.io/>
- [5] ArduPilot. (2019) Ardupilot open source autopilot. [Online]. Available: <http://ardupilot.org/>
- [6] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.
- [7] T. Istanbul. (2019) Teknofest istanbul. [Online]. Available: <https://www.teknofestistanbul.org/>
- [8] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.
- [9] R. Vaughan, "Massively multi-robot simulation in stage," *Swarm intelligence*, vol. 2, no. 2-4, pp. 189–208, 2008.
- [10] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, 2018.
- [11] Gazebo. (2019) Gazebo tutorial: Aerodynamics. [Online]. Available: <http://gazebo.org/tutorials?cat=aerodynamics&cat=plugins>
- [12] C. S. Myers and L. R. Rabiner, "A comparative study of several dynamic time-warping algorithms for connected-word recognition," *Bell System Technical Journal*, vol. 60, no. 7, pp. 1389–1409, 1981.
- [13] O. Aşık, G. Uludoğan, and H. K. Eryılmaz. (2019) Teknofest sürü İha benzetim ortamı. [Online]. Available: <https://github.com/teknofest-suriha/suriha>
- [14] ——. (2018) Teknofest istanbul 2018 sürü İha yarışması. [Online]. Available: <https://teknofest-suriha.github.io>