

Otonom Dronlar için Optimal Görev Atama ve Rota Planlama

Hüseyin Göktaş

**YÜKSEK LİSANS TEZİ**

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Ocak 2022

Optimal Mission Assignment and Route Planning for Autonomous Drones

Hüseyin Göktaş

**MASTER OF SCIENCE THESIS**

Department of Electrical and Electronics Engineering

January 2022

Otonom Dronlar için Optimal Görev Atama ve Rota Planlama

Hüseyin Göktaş

Eskişehir Osmangazi Üniversitesi

Fen Bilimleri Enstitüsü

Lisansüstü Yönetmeliği Uyarınca

Elektrik Elektronik Mühendisliği Anabilim Dalı

Kontrol ve Kumanda Sistemleri Bilim Dalında

YÜKSEK LİSANS TEZİ

Olarak Hazırlanmıştır

Danışman: Dr. Öğr. Üyesi Hasan Serhan Yavuz

Bu tez çalışması, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından EEEAG-116E080 numaralı “İnsansız Hava Araçları ile Nesne Bulma ve Hareketli Nesne Takibi” başlıklı projeye desteklenmiştir.

Ocak 2022

## ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Dr. Öğr. Üyesi Hasan Serhan Yavuz danışmanlığında hazırlamış olduğum “Otonom Dronlar için Optimal Görev Atama ve Rota Planlama” başlıklı YÜKSEK LİSANS tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 13/01/2022

Hüseyin Göktaş

İmza

## ÖZET

İnsansız hava araçları (İHA), özellikle görevin kritik ve tehdidin yoğun olduğu askeri alanlarda, insan kaybı riskinin bulunmaması, farkedilmesinin zor olması ve uçuş maliyetinin ucuz olmasından dolayı insanlı sistemlere göre çok büyük avantajlara sahiptir. Son zamanlarda İHA'lar gözetleme, hedef tespiti ve askeri destek gibi alanların dışında fotoğrafçılık, tarım, paket teslimat gibi özel uygulamalarda da oldukça yaygın kullanılmaktadır. Hedef alanı büyük olan bölgelerde verilen görevi daha etkili bir şekilde gerçekleştirmek için otonom sürü İHA'lara ihtiyaç duyulmaktadır. Sürü İHA'lara görev planlaması yapılırken belirli kriterler göz önüne alınarak İHA'ların toplam maliyetleri en aza indirilmelidir. Bu problemi çözmek için literatürde daha çok sezgisel yöntemli yaklaşımlar kullanılmıştır.

Bu tez çalışmasında, hedef ve İHA sayısının çok olduğu durumlarda görev atama problemini verimli bir şekilde gerçekleştirmek için üç aşamalı hiyerarşik bir yöntem sunulmuştur. İlk aşamada görevlerdeki hedef noktaları  $k$ -ortalamlar kümeleme yöntemi kullanılarak İHA sayısına göre alt kümeler ayrılır. Daha sonra Macar Algoritması kullanılarak optimal drone-alt küme ataması yapılır. Son aşamada ise, ataması yapılan her bir İHA'nın kat edeceği mesafeyi minimize etmek için karınca kolonisi algoritması kullanılarak rota planlaması gerçekleştirilir. Önerdiğimiz yöntem farklı senaryolar altında simüle edilmiş ve metasezgisel optimizasyon yöntemleri ile karşılaştırılarak başarılı sonuçlar elde edilmiştir. Tasarlanan senaryolarda, İHA lokasyonlarının farklı durumları, İHA sayısının kısıtlı olması ya da her bir İHA için görev sayısının kısıtlı olması durumları göz önünde bulundurulmuştur.

**Anahtar Kelimeler:** İnsansız hava aracı (İHA), çoklu görev atama, macar algoritması, karınca kolonisi optimizasyonu,  $k$ -ortalamlar

## SUMMARY

Unmanned aerial vehicles (UAVs) have great advantages over manned systems, especially in mission-critical and threat-intensive military areas. These advantages can be summarized as follows: There is no risk of human loss, it is difficult to be noticed, and the flight cost is cheap. Recently, UAVs have also been widely used in special applications such as photography, agriculture, package delivery, surveillance, target detection and military operations. Autonomous swarm UAVs are needed to more effectively perform the given task in regions with large target areas. While planning a mission by using multi UAVs, the total costs of UAV usage should be minimized by considering certain criteria. In general, heuristic approaches have been used to solve this problem.

In this thesis, we propose a three-stage hierarchical method to efficiently solve the task assignment problem in cases where the number of targets and UAVs is high. In the first stage, the target points in the missions are divided into subsets according to the number of UAVs using the  $k$ -means clustering method. Then, the optimal drone-subsets assignment is made by using the Hungarian Algorithm. In the last stage, route planning is carried out by using the ant colony algorithm to minimize the distance to be covered by each assigned UAV. We simulated the proposed method under different environmental scenarios and obtained successful results when compared with some other metaheuristic optimization methods. In the designed scenarios, different UAV locations, limited number of UAVs or limited number of missions for each UAV were taken into account.

**Keywords:** UAV, multiple task assignment, clustering, Hungarian algorithm, ant colony optimization,  $k$ -means.

## İÇİNDEKİLER

### Sayfa

<b>ÖZET</b> .....	<b>vi</b>
<b>SUMMARY</b> .....	<b>vii</b>
<b>TEŞEKKÜR</b> .....	<b>viii</b>
<b>İÇİNDEKİLER</b> .....	<b>ix</b>
<b>ŞEKİLLER DİZİNİ</b> .....	<b>xi</b>
<b>ÇİZELGELER DİZİNİ</b> .....	<b>xii</b>
<b>SİMGELER VE KISALTMALAR DİZİNİ</b> .....	<b>xiii</b>
<b>1. GİRİŞ VE AMAÇ</b> .....	<b>1</b>
<b>2. LİTERATÜR ARAŞTIRMASI</b> .....	<b>4</b>
<b>3. TEMEL KAVRAMLAR</b> .....	<b>8</b>
3.1 Sürü İHA Görev Planlaması .....	8
3.2. Kümeleme Yöntemleri .....	9
3.2.1 <i>K</i> -ortalamalar Yöntemi .....	9
3.2.2 Spektral Kümeleme Yöntemi .....	10
3.3. Atama Problemi ve Macar Algoritması .....	11
3.4. Rota Planlama .....	12
3.4.1. Kesin Yöntemler .....	13
3.4.2. Metasezgisel Yöntemler .....	15
<u>3.4.2.1 Karınca Kolonisi Algoritması</u> .....	15
<u>3.4.2.2 Genetik Algoritma</u> .....	16
<u>3.4.2.3. Parçacık Sürü Optimizasyonu</u> .....	18
<u>3.4.2.4. Tavlama Benzetimi Algoritması</u> .....	19
<b>4.MATERYAL VE YÖNTEM</b> .....	<b>21</b>
4.1 Hedef Noktalarını Kümelenendirme .....	22
4.2 Alt Küme-İHA Ataması .....	24
4.3 Rota Planlama .....	26
<b>5. BULGULAR VE TARTIŞMA</b> .....	<b>29</b>

5.1. Simülasyon Testleri .....	29
5.1.1 Senaryo-1.....	29
5.1.2 Senaryo-2.....	30
5.1.3 Senaryo-3.....	31
5.2 Test Sonuçları .....	32
5.3. İlave Kısıtlar İçeren Farklı Bir SİHA Senaryosu.....	39
<b>5. SONUÇ VE ÖNERİLER.....</b>	<b>42</b>
<b>KAYNAKLAR DİZİNİ.....</b>	<b>43</b>

## ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
3.1. Örnek bir k-ortalamlar kümeleme yönteminin gösterimi.....	10
3.2. Spektral kümeleme ile k-ortalamlar yönteminin karşılaştırılması.....	11
3.3. Optimal görev atama yöntemi.....	12
3.4. Ağaç veri yapısı.....	15
3.5. Karıncaların gıda kaynağına gidecek en yakın yolu bulma yaklaşımı.....	16
3.6. Genetik algoritmanın şeması.....	17
3.7. Çaprazlama yöntemi.....	18
4.1. Önerilen yöntemin tüm süreçlerini gösteren şema.....	21
4.2. K-ortalamlar yöntemi ile verilerin dört kümeye ayrılmasının gösterimi.....	23
4.3. K-ortalamlar kümeleme yöntemi akış diyagramı.....	24
5.3. Senaryo-1'in gösterimi.....	30
5.3. Senaryo-2'in gösterimi.....	31
5.3. Senaryo-3'in gösterimi.....	31
5.4. Yer istasyonun (550, 450) konumunda olduğu durum için atama sonuçları.....	33
5.5. Görevlendirilen İHA sayısına göre toplam maliyetin gösterimi.....	33
5.6. Yer istasyonun (0, 0) konumunda olduğu durum için atama sonuçları.....	35
5.7. Görevlendirilen İHA sayısına göre toplam maliyetin gösterimi.....	35
5.8. Yer istasyonun farklı konumlarda olduğu durum için atama sonuçları.....	37
5.9. Görevlendirilen İHA sayısına göre toplam maliyetin gösterimi.....	37
5.10. SİHA Senaryosunun gösterimi.....	39
5.11. Görev ataması yapılan SİHA'ların gösterimi.....	40

**ÇİZELGELER DİZİNİ****Çizelge****Sayfa**

- 5.1. Farklı sayıda İHA ve hedef kullanılarak elde edilen toplam maliyet değerleri ..... 38
- 5.2. Farklı sayıda hedefler için hesaplanan ortalama süre verileri ..... 39

**SİMGELER VE KISALTMALAR DİZİNİ****Kısaltmalar**

ARP

ÇDARP

DP

GA

GSP

İHA

KKA

MILP

PSO

SİHA

TB

**Açıklama**

Araç Rotalama Problemi

Çok Depolu Araç Yönlendirme Problemi

Doğrusal programlama

Genetik algoritma

Gezgin Satıcı Problemi

İnsansız Hava Aracı

Karıncı Kolonisi Algoritması

Karışık tamsayılı doğrusal programlama

Parçacık Sürü Optimizasyonu

Silahlı İnsansız Hava Aracı

Tavlama Benzetimi

## 1. GİRİŞ VE AMAÇ

Son yıllarda insansız hava aracı (İHA) teknolojilerinin hızla gelişmesiyle birlikte İHA'lar birçok alanda yaygın olarak kullanılmaya başlanmıştır. İHA'lar, başlangıçta insan için tehlikeli olan askeri amaçlı görevleri yerine getirmek üzere tasarlanmış olmasına rağmen, günümüzde ticari, bilimsel, eğlence, gözetim, denetleme, ürün teslimatları, hava fotoğrafçılığı gibi birçok alanda kullanılmaktadır. Otonom ve uzaktan kumandalı olmak üzere iki gruba ayrılan İHA sistemlerinin en büyük avantajı hayati riski ortadan kaldırmasıdır. Hareket kabiliyetlerinin yüksek olması ve insan müdahalesinin zor ve tehlikeli olduğu alanlarda hareket edebilmesi sayesinde askeri operasyonlarda önemli bir yer almaktadır. Şuanki uygulamaların birçoğu İHA'ların uzaktan kontrolü ile gerçekleştirilirken, bazı görevler için otonom uçuşlar da yapılmaktadır. Gelişen teknoloji ve yüksek olmayan üretim maliyetleri sayesinde elde edilebilirliği kolaylaşan insansız hava araçlarının günlük hayatımızda da kullanımını oldukça artmıştır (Otto, 2018).

Akıllı birimler olarak kullanılabilen sürü İHA sistemleri özellikle savunma sanayisinde son dönemlerde adından sıkça söz ettirmektedir. Sadece bir İHA'nın olduğu görev planlama sürecinde, görevler arasında öncelik sıralaması kolay bir şekilde çözülebilirken, sürü İHA'ların görev planlama süreci oldukça karmaşık bir durum almaktadır. Merkezileştirilmiş ve merkezi olmayan olarak ayrılan çok ajanlı yol planlamasında merkezi yaklaşım, daha büyük sistemlerde yüksek iletişim sorunu yaşar ve herhangi bir noktada arıza meydana gelmesine karşı savunmasızdır. Öte yandan merkezi olmayan yani dağıtık sistemler, merkezi sistemin sınırlamalarını azaltmakta ve görev aralığının genişletilmesine olanak sağlamaktadır. Bir görevi daha verimli bir şekilde yerine getirmek için, otonom araçların varış noktalarına giderken birkaç görevi tamamlaması gerekir. Çok etmenli sistemlerde verimli görev dağılımı konusunda bazı kapsamlı çalışmalar yapılmıştır.

Maliyetleri azaltmak için en kısa yolun bulunmasının önemli olduğu Araç Rotalama Problemi (ARP) ilk olarak 1959 yılında Dantzig ve Ramser tarafından öne sürülmüştür (Dantzig ve Ramser, 1959). NP-Zor problemler sınıfına ait olan bu problem çoklu İHA işbirlikli görev atama probleminde de yaygın bir şekilde kullanılmaktadır. Problemin karmaşıklığı hedef veya İHA sayısına göre değişmektedir. Hedef tespiti, gözetim, bilgi toplama ve benzeri uygulamalarda daha fazla alanı daha kısa sürede taramak için birden fazla İHA kullanılmaktadır. Karmaşık yürütülen görevler sürü İHA sistemleri ile daha hızlı ve daha verimli bir şekilde çözülebilmektedir.

Birden fazla insansız hava aracının olduğu durumlarda, ekibin genel faydasını maksimize etmek için hedef lokasyonlar arasında görev dağılımını ve yol planlamasını verimli bir şekilde yapmak zorlu bir problemdir. Bu tür problemler, tamsayı programlama problemi olarak ele alınabilir. Tamsayı programlama, değişkenlerin bir kısmının veya tümünün tamsayılarla sınırlandırıldığı bir optimizasyon sürecidir. En uygun çözümleri bulmak için literatürde birçok yaklaşım ve yöntem geliştirilmiştir. Bu yaklaşımlardan biri matematiksel bir programlama yöntemi olan karışık tamsayılı doğrusal programlamadır. Drone'ların görev atamasında birden fazla hedef lokasyonun varlığında, problem gezgin satıcı problemi olarak da düşünülebilir. Şehirlerin bir listesi ve her bir çift şehir arasındaki mesafeler verildiğinde, gezgin satıcı İHA'nın her şehri ziyaret edip başlangıç şehre geri döndüğü mümkün olan en kısa rotayı belirlemeyi amaçlamaktadır. Gezgin satıcı problemi, meta-sezgisel yöntemlerden olan genetik algoritma, karınca kolonisi algoritması veya benzer birçok farklı algoritma kullanılarak başarıyla çözülebilmektedir.

Bu tezde, otonom uçuş yapabilen sürü İHA için hedef sayısının İHA sayısından fazla olduğu karmaşık durumlarda görev planlama problemini optimal bir şekilde gerçekleştirmek üzere çalışmalar yapılmıştır. Özellikle son yıllarda sürü İHA'ların görev ataması ile ilgili metotlar literatürde sıkça yer bulmaktadır. Bunlardan bir tanesi hiyerarşik atama yöntemidir. İHA ve hedef sayısı arttıkça problem daha karmaşık hale geldiğinden dolayı, bu problemin tek bir basit yöntem kullanılarak çözülmesi mümkün değildir. Önerdiğimiz yöntemde sürü İHA'lara görev atama problemini çözmek için sırasıyla kümeleme, macar algoritması ve karınca kolonisi optimizasyonunu içeren 3 aşamalı hiyerarşik bir metodoloji sunulmuştur. Bu çalışmada her bir İHA'nın birden fazla hedefe gidip gözetleme görevi yaptıktan sonra tekrar başlangıç noktasına geri döneceği varsayılmıştır.

Bu tez çalışması toplam 5 bölümden oluşmaktadır. **Literatür Araştırması** bölümünde İHA'ların kullanıldığı alanlar ve İHA görev atama ve rota planlamada kullanılan yöntemler incelenmiştir. **Temel Kavramlar** bölümünde görev planlamanın genel tanımını ve yöntemlerinin yapısı ele alınmıştır. **Materyal ve Yöntem**'de ise tez çalışmasında önerdiğimiz üç aşamalı hiyerarşik yöntemin detaylı açıklaması verilmiştir. **Bulgular ve Tartışma** bölümünde yapılan testler verilmiş ve elde edilen sonuçlar tartışılmıştır. Son olarak **Sonuç ve Öneriler** bölümünde ise çalışmada ulaşılan genel sonuçlar açıklanmıştır.

## 2. LİTERATÜR ARAŞTIRMASI

Son yıllarda İHA kullananların sayısında çok büyük bir artış yaşanmaktadır. Aynı zamanda üretici ve geliştirici sayısı da artmaktadır. Bu durum, İHA'ları içeren araştırmalardaki ilgiyi arttırırken teknolojik gelişmelerle birlikte İHA'lardan yararlanılan araştırmaların sayısını da önemli bir şekilde etkilemiştir. İHA'lar önceden sadece askeri uygulamalarda kullanılırken günümüzde ise sivil alanlarda da oldukça fazla sayıda kullanılmaktadır (Cerro vd. 2021). Sivil uygulamalar genellikle; hava haritalama, tarım, paket teslimatı, afet yönetimi, iç güvenlik gibi uygulamalardan oluşmaktadır. Son zamanlarda İHA'lardan yararlanılan çok sayıda askeri uygulama vardır. Glade ve diğerlerine (2000) göre, mevcut ve gelecekteki İHA'ları kullanabilecek olası uygulamalar, ulaşım, sinyal toplama, istihbarat, gözetleme ve keşif, muharebe destek görevlerini içerir. Askeri görevlerin çoğu, gözetleme ve veri toplama gibi görevleri içerir. Mini İHA'ların farkedilmesi zor, dayanıklı ve hızlıdır. Bu nedenlerden dolayı bu tür görevlerde daha çok mini İHA'lar kullanılmaktadır (Coffey ve Montgomery, 2002).

İHA'ların görev planlamasında görev ataması zor bir problemdir. Birden fazla İHA'ya birden fazla görev atanması durumunda, sorun daha da karmaşık hale gelmektedir. Edison ve Shima (2011), merkezi olan yöntemler için çoklu görev çoklu hedef atama problemini bir kombinatoriyal optimizasyon problemi olarak ele almıştır. Problemi çözmek için genetik algoritma önermiştir. Shima vd. (2006), hesaplamayı hızlandırmak için kullandıkları genetik algoritmayı dal ve sınır ve stokastik arama yöntemleriyle karşılaştırdıklarında, genetik algoritma büyük ölçekli problemler için çok daha hızlı sonuçlar vermiştir.

Schumacher vd. (2007), zamanlama ve görev sırası kısıtlamaları ile görev atamasını optimize etmek için karma tamsayılı doğrusal programlamayı kullanmıştır. Shetty vd. (2008) çalışmasında, minimum maliyetli ağ akışı problemi olarak modellenmiş görev atama problemini çözmek için tabu arama sezgiseli geliştirmiştir. Choi vd. (2009), işbirlikçi İHA'ların görev ataması için market tabanlı merkezi olmayan bir görev atama problemi geliştirmiştir. Rasmussen ve Shima (2008), bir grup homojen İHA'ya görevler atamak için

ağaç arama algoritmasını kullanmıştır. Bu yöntemde problem bir karar ağacı olarak temsil edilmiş ve arama algoritması kullanılarak çözülmüştür. Belirsiz koşullar altında görev ataması da araştırmacılar tarafından dikkate alınmıştır. Bertuccelli vd. (2009) çalışmasında, belirsiz ortamlarda heterojen İHA'ların gerçek zamanlı görev atama problemini incelemiştir. Alighanbari ve How (2008), belirsizliğe karşı gerçek zamanlı sağlam bir görev atama yöntemi geliştirmiştir.

Forsmo vd. (2013), çoklu İHA görev atama problem modelinin temel olarak şunlardan oluştuğunu ifade etti: karmaşık tamsayılı doğrusal programlama modeli, dinamik ağ akış optimizasyon modeli ve çoklu akıllı araç tabanlı sistemler. Alighanbari ve How (2005), çoklu İHA görev atama problemlerini dinamik programlama problemi olarak formüle etmektedir. Ayrıca, Alighanbari vd. (2003), Tabu Arama algoritması ile görev tahsis problemini çözmek için karmaşık tamsayılı doğrusal programlama modeli sunmuştur.

Gazali vd. (2016), afet bölgelerinde hayatta kalanların yerlerini belirlemek için döner kanatlı İHA'lar kullanmışlardır. Kurtarma operasyonu için sundukları algoritma ile hayatta kalanların olabileceği alanların fotoğrafları çekilmiş, daha sonra fotoğraflar dörde bölünerek hayatta kalanlar bulunana kadar süreç tekrarlanmıştır. Lee ve Morrison (2015), deniz ortamında arama ve kurtarma problemlerine uygulamak için karma tamsayılı doğrusal programlama önermiştir. Bu tür problemlerde rüzgâr ve akıntılar nedeniyle hayatta kalanları bulma olasılığı değişmektedir. Karışık tamsayılı doğrusal programlama modeli bu tür problemlerde farklı sayısal örnekler üzerinde verimli bir şekilde çalışmıştır.

Schwarzrock vd. (2018), gelecekteki otonom askeri sistemler için işbirlikçi İHA'larda görev atama sorununu ele almıştır. Ajanlar arasında görev atama problemi merkezi bir varlık tarafından oluşturulmuş, ancak görev kararı ajanların kendileri tarafından alınmaktadır. Zhong vd. (2013), dinamik değerlere sahip hedeflere saldırmak için birden fazla İHA'nın görev atamasını, savaş alanında hedeflerin zamanla değiştiğini göz önünde bulundurarak, dinamik programlama tabanlı çok hedefli rota planlaması ve çok alt gruplu karınca kolonisi algoritması birleşiminden oluşan bir çözüm sunmuştur.

Ağaç arama algoritması, karınca kolonisi algoritması ve durum-uzay en iyi ilk arama algoritması merkezi yöntemlerdendir (Rasmussen ve T. Shima, 2008; Zhao vd. 2017). Bu

yöntemlerin çoğu, NP-zor problemlerin çözümünde kullanılan global optimizasyon algoritmalarıdır. Merkezi olmayan algoritmalar ile yapılan çalışmalar, merkezi algoritmalarınkinden önemli oranda daha azdır (Zhao vd. 2016; Qiu vd. 2017). Merkezi olmayan yöntemlerin ana avantajı çoklu İHA görev atama probleminin verimliliğini artırmaktır. Çoklu İHA işbirliği içinde çalışırken, İHA'lerden biri düzgün çalışmasa bile merkezi olmayan algoritmalar sayesinde çoklu İHA sisteminin başarılı bir şekilde çalışabilmesini sağlamak için diğer İHA'lar onun yerine yeniden tahsis edilebilir. Liu vd. (2013), iki seviyeli programlamaya dayalı gerçek zamanlı yol planlaması için bir lider-takipçi hiyerarşik optimizasyon yapısı sunmuştur. Sousa vd. (2004), İHA takımlarının görev planlaması ve yürütülmesi için iki seviyeli bir hiyerarşik yapı tasarlamıştır.

Chen ve Liu (2019), hareketli hedefler için çoklu İHA işbirlikçi görev ataması sorununu ele almıştır. Tanımlanmış kombinatoriyal optimizasyon problemini, bir yönlendirme mekanizmasına dayalı yeni bir parçacık sürüsü optimizasyon algoritması ile çözdüler ve rota planlama ve anlık yeniden planlama metodolojilerini göz önünde bulundurdular. Tang ve Zhu (2016) çalışmasında, parçacık optimizasyon algoritması ile bir görev atama yöntemi tanıtmıştır. Tianhan vd. (2019), bir uçuş rotası planlamak için tehdit ve yol mesafesi maliyetini göz önünde bulundurarak işbirlikçi İHA'lar için saldırı ve rota planlama problemini incelemiştir. Çalışmalarında gelişmiş bir parçacık sürüsü optimizasyon algoritması kullanmışlardır.

Hussein vd. (2012) yaptıkları çalışmada, rota planlama problemi için yörünge ve popülasyon tabanlı metasezgisel yöntemleri karşılaştırmışlardır. Zaman açısından tavlama benzetimi algoritması ön plana çıkarken, en kısa rotayı hesaplama açısından yasaklı arama algoritması en başarılı olmuştur.

Devriye görevlerinde İHA'ların kullanımı ile ilgili de literatürde birçok çalışma bulunmaktadır. Lee vd. (2015) çalışmasında, otonom İHA ile devriye ve gözetleme sistemini önermiştir. Zhou vd. (2019), esas olarak kuşbakışı görüntü birleştirme ve nesne algılamaya dayalı bir İHA devriye sistemine odaklandı. Zaza ve Richards'ın (2014) çalışmalarında, İHA'ların dayanıklılıkları nedeniyle gözetleme görevlerinde giderek daha fazla kullanıldığına dikkat çekilmiştir. Ries ve Ishizaka (2012), deniz gözetimi ortamında İHA rotası için bir karar destek sistemi uygulamayı amaçlamıştır. Ayrıca, gerçek zamanlı bir

yönlendirme sistemi kurmak için matematiksel bir programlama önermiştir. Thakoor vd. (2020), bir bölgenin gözetilerek bilgi toplaması gerektiği durumlar için çoklu İHA rota planlama sorununu ele aldı. Coelho vd. (2017), birden fazla şarj istasyonunu ve operasyonel gereksinimleri göz önünde bulundurarak sınırlı otonom heterojen filo yönlendirme problemini incelemiştir.

Dorling vd. (2017), dronların paket teslimatında maliyet ve süreyi azaltma potansiyeline sahip olduğuna ve birçok büyük kuruluşun drone teslimatına ilgi gösterdiğine dikkat çekmiştir. Örneğin, bu yöntem (Yadav ve Narasimhamurthy 2017)'de dronlar için teslimat planlaması sorununu ele alan bir sezgisel yöntem önerilmiştir. Casbeer ve Holsapple (2011), öncelik kısıtlamaları olan çoklu İHA'larda görevleri atamak için dal ve fiyat optimizasyonu ile sütun oluşturma yöntemini kullanmıştır. Yadav ve Narasimhamurthy (2017) çalışmasında, dronlarda teslimat planlaması sorununu çözmek için sezgisel bir yöntem önerilmiştir.

İHA'nın rota planlaması, Dantzig ve Ramser (1959) tarafından tanıtılan araç yönlendirme probleminin (ARP) bir uzantısı olarak düşünülebilir. Benzin istasyonlarına teslimat sorununu çözmeye çalışmışlardır. Klasik ARP'nin dışında çok sayıda depo ve İHA'nın olduğu problemler de vardır. Gen ve Syarif (2005), Çok Depolu Araç Yönlendirme Problemi (ÇDARP) ile birden fazla depoda bulunan hammaddelerin bir grup fabrikaya taşınması problemini formüle etmiş ve sezgisel bir yaklaşım önermiştir.

### 3. TEMEL KAVRAMLAR

Sürü İHA’larda görev planlama problemi için kullanılan çok farklı yöntemler vardır. Bu bölümün en başında sürü İHA’larda görev planlama problemi açıklanmıştır. Sonrasında literatürde çok yaygın olan kümeleme yöntemlerinden bahsedilmiştir. Daha sonra ise görev atama problemi için tez kapsamında kullanılan macar algoritması hakkında ayrıntılı bilgi verilmiştir. En son olarak önerdiğimiz yöntemin son aşaması olan rota planlama problemini çözen diğer bazı önemli yöntemler detaylandırılarak açıklanmıştır.

#### 3.1 Sürü İHA Görev Planlaması

İHA görev planlaması özellikle askeri operasyonlarda çok önemli bir yere sahiptir. Görev planlamasındaki en büyük amaç toplam yol maliyetini en aza indirmek veya zamandan tasarruf sağlamaktır. Çoklu İHA görev atama problemi bir kombinatoriyal optimizasyon problemidir. Bu problemin yaygın örnekleri, gezgin satıcı problemi, stok kesme problemi, paketleme problemleri ve görev atama problemidir.

Optimizasyon, hedeflenen amaç için verilen kısıtlamaların sağlanarak en uygun çözümün elde edilme işidir. Çoklu İHA görev atama probleminin karmaşıklığı nedeniyle bu problemlerde genellikle sezgisel algoritmalar kullanılır. Görev planlamasını tamamlamak için bir veya birden fazla İHA kullanılabilir. Yer istasyonları, İHA'ların göreve başladığı ve görevi bitirdikleri yerdir. Görev atama planlaması görev başlamadan önce yapılmalıdır. Maliyet ölçümü için; genellikle kat edilen mesafe, geçen süre veya tüketilen yakıt miktarları kullanılır. Bir görev planlamasını başarıyla tamamlamak için atanan İHA’ların tüm hedeflere uğraması ve başlangıç baz istasyonuna tekrar geri dönmeleri gereklidir.

## 3.2. Kümeleme Yöntemleri

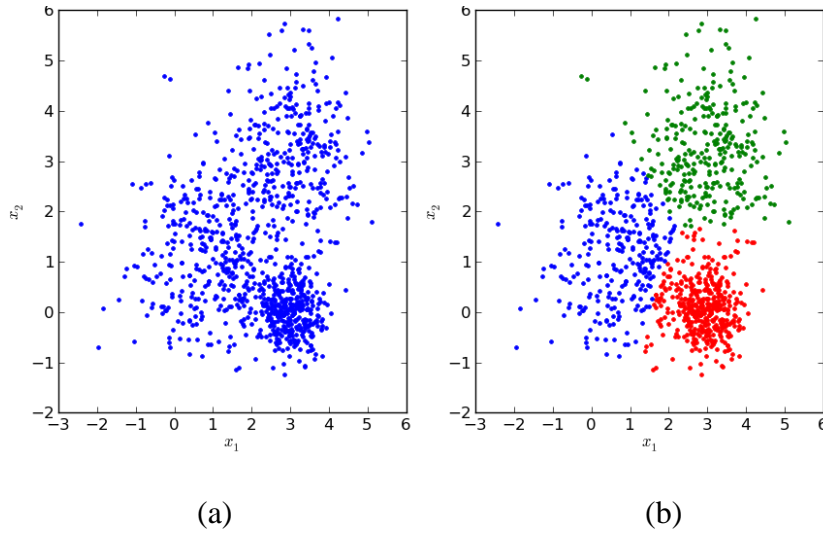
Kümeleme, veri noktaları tarafından oluşan doğal grupların tanımlanmasıdır. Kümeleme yaklaşımları sıklıkla veri madenciliği, makine öğrenimi, örüntü tanıma gibi çok çeşitli alanlarda kullanılmaktadır. Temel anlamda kümelendirme problemi, belirli bir veri setinde homojen veri gruplarını bulma problemi olarak tanımlanır. Bu grupların her birine küme adı verilir. Literatürde birçok yöntem mevcuttur. Bu tez çalışmasında, literatürde en çok ve en yaygın olarak kullanılan küme merkez güdümlü (centroid based) bir algoritma olan  $k$ -ortalamlar ve alternatif olarak bağlantı güdümlü (connectivity based) bir algoritma olan spektral kümeleme yöntemleri kullanılmıştır.

### 3.2.1 $K$ -ortalamlar Yöntemi

$K$ -ortalamlar kümeleme yöntemi, Mac Queen tarafından 1967 yılında geliştirilen ve bir veri setini belirlenen sayıdaki kümelere bölme işlemidir. Literatürde en çok tercih edilen kümeleme algoritmalarından birisi  $k$ -ortalamlar yöntemidir. Bu yöntem ile  $n$  adet noktadan oluşan bir küme  $k$  adet kümeye ayrılır. Öklid uzaklık tabanlı büyük ölçekli problemlerde çok hızlı sonuçlar vermesinden dolayı bu yöntem öncelikli olarak tercih edilmektedir. Veri setindeki nesnelere her biri küme merkezlerine olan uzaklıklarına göre bir kümeye dâhil olurlar. Bir verinin sadece bir kümeye atanması gerekmektedir. Şekil 3.1'de, örnek olarak 2 boyutlu uzayda rastgele konumlarda tanımlanmış olan veri örneklerinin  $k$ -ortalamlar kümeleme yöntemiyle 3 kümeye bölünmesi gösterilmiştir.

Standart bir  $k$ -ortalamlar yöntemi aşağıda belirtilen adımlardan oluşmaktadır:

- i. Belirlenen küme sayısı adedince rastgele merkez noktalar seçilir.
- ii. Verilerin tek tek merkez noktalarına olan uzaklıklarına göre yeniden kümelenebilir yapılar.
- iii. Yeni oluşan kümelere yeni merkezler belirlenir.
- iv. Belirlenen merkezlerin yeri değişmeyene kadar yukarıdaki adımlar tekrarlanır.



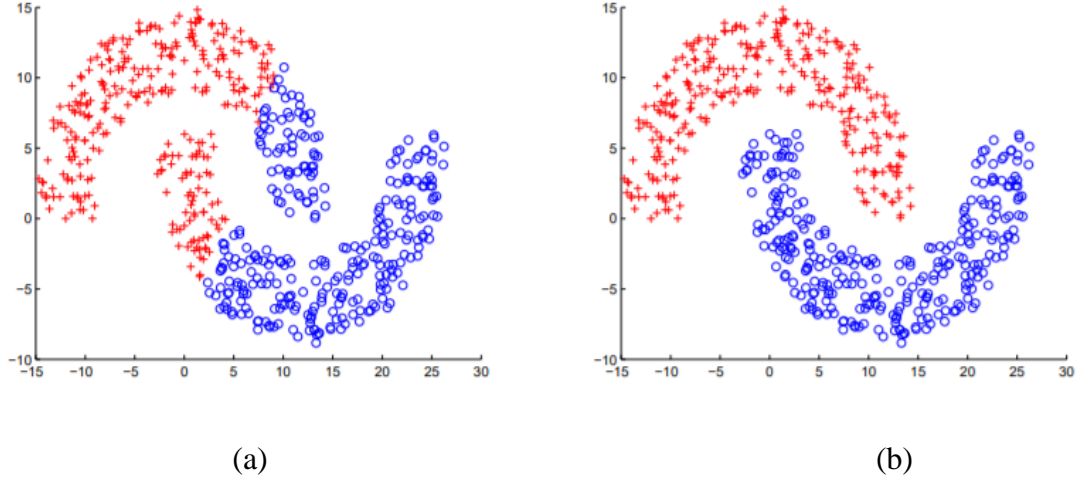
Şekil 3.1: Örnek bir  $k$ -ortalamalar kümeleme yönteminin gösterimi.

(a) Kümeleme öncesi (b) Kümeleme sonrası

### 3.2.2 Spektral Kümeleme Yöntemi

Spektral kümeleme yöntemi belirli verilerin çıkarılmasında en iyi ve en önemli algoritmalarından biri olarak kabul edilir ve büyük miktarda verinin aranmasında iyi sonuçlar vermektedir (Bolla, 2013). Kümeleme, veri madenciliğinin bir dalı olup benzer kümelerden elde edilen bir veri sürecidir. Kümeleme algoritması, veri kümelerini birkaç gruba ayırır; burada bir grup içindeki noktalar arasındaki benzerlikler, iki farklı grup içinde bulunan noktalar arasındaki benzerliklerden daha fazladır. Büyük miktarda veri ile uğraşıldığında, analiz sürecini kolaylaştırmak için veriler birkaç kategoriye ayrılmaktadır. Verileri yalnızca sınıflandırılmak için değil, aynı zamanda geniş çapta toplanan verilere bazı sistem algoritmaları oluşturmak için de kullanılır. Veri kümeleri bulunursa, bu gruplara dayalı bir problem modeli oluşturmak mümkündür.

Kolay uygulama özelliğine sahip olan ve veriler arasındaki bağlantılara dayanan kümeleme sonuçları veren spektral kümeleme yöntemi, bazı parametrelerin seçimine oldukça duyarlıdır ve büyük verilerin hesaplanmasında çok maliyetlidir. Spektral kümeleme,  $k$ -ortalamalar yönteminden farklı olarak düzensiz form gruplarına ait nesnelere gruplayabilir (Şekil 3.2).



Şekil 3.2: Spektral kümeleme ile  $k$ -ortalamalar yönteminin karşılaştırılması. (a)  $K$ -ortalamalar yöntemi, (b) Spektral kümeleme yöntemi

### 3.3. Atama Problemi ve Macar Algoritması

Atama problemi, toplam maliyeti minimize edecek  $n$  adet farklı ajanın yine  $n$  adet farklı göreve atanmasıyla ilgilenen bir kombinatoriyal optimizasyon problemidir. Bu tür problemleri çözmek için birçok yöntem mevcuttur. Macar algoritması görev atama problemlerini çözmeye yaygın olarak kullanılan algoritmalarından biridir. Ajan ve görevler arasında en uygun eşleşmeleri bulup atama görevi yapar (Bkz. Şekil 3.3). Atama probleminde ele alınan tek kriter maliyettir ve ajan sayısının görev sayısından fazla, az ya da eşit olduğu üç farklı durum olabilir. Ajan ve görev sayılarının birbirine eşit olmaması durumunda hayali ajan veya görevler eklenmesi yoluyla atama problemi kolay bir şekilde dengelenir. Her bir görev-ajan atamasının maliyetinin verildiği varsayılarak  $n$  görev ve  $n$  ajanın bulunduğu optimal atama probleminde ikili karar değişkeni  $x_{ij}$  ile ifade edilir.  $x_{ij}$  bir görevin ajana atanması durumunda 1 diğer durumlar için 0 değerini alır.  $c_{ij}$ ,  $i$  ve  $j$  arasındaki maliyeti (uzaklığı) ifade etmektedir. Problemin matematiksel modeli aşağıdaki gibidir.

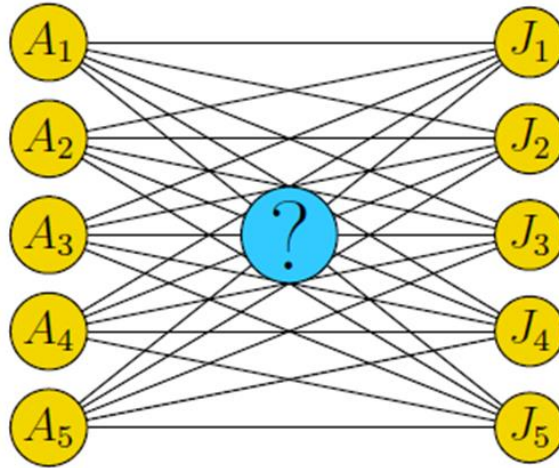
$$Z_{min} = \sum_{i=1}^n \sum_{j=1}^n c_{ij} \cdot x_{ij}, \quad (3.1)$$

$$\sum_{j=1}^n x_{ij}=1; \quad i \in \{1, 2, \dots, n\} \quad (3.2)$$

$$\sum_{i=1}^n x_{ij}=1; \quad j \in \{1, 2, \dots, n\} \quad (3.3)$$

$$x_{ij} \in \{0,1\}, \forall i,j \quad (3.4)$$

Denklem 3.1’de, toplam atama maliyetini en küçükleyen amaç fonksiyonunu ifade etmektedir. Denklem 3.2 her bir ajanın sadece bir hedefe atanmasını sağlar. Benzer şekilde, Denklem 3.3 her bir hedefin sadece bir ajana atanabileceğini ifade eder. Denklem 3.4’de tanımlanan  $x_{ij}$  değişkeni ajan  $i$ ’in hedef  $j$ ’ye atanması durumunda 1, diğer durumda ise 0 değerini alır.



Şekil 3.3: Optimal görev atama yöntemi.

### 3.4. Rota Planlama

Sürü İHA Araç Rotalama Problemi son zamanlarda oldukça ilgi görmektedir. Karmaşık bir kombinatoriyal problem olan araç rotalama problemi (ARP) ve gezgin satıcı problemi (GSP) optimizasyon alanında en önemli problemlerdendir. Belirli kısıtlamalar

altında toplam kat edilen mesafenin minimum olması problemin amaç fonksiyonunu oluşturmaktadır.

Araç rotalama yapılırken aşağıdaki şartlar dikkate alınmalıdır:

- Tüm hedeflere gidilmesi gerekmektedir.
- Her bir hedef noktası tek bir ajan tarafından sadece bir kez ziyaret edilmelidir.
- Rota yer istasyonundan başlayıp tekrar yer istasyonunda sonlanmalıdır.
- Her bir ajan sadece bir rota üzerinde faaliyet göstermelidir.
- Temel amaç ajanların kat edecekleri toplam mesafenin minimize edilmesidir.

Bu bölümde gezgin satıcı problemini etkin şekilde çözebilen kesin çözümlü yöntemler ve metasezgisel yöntemlerden bahsedilmiştir

### 3.4.1. Kesin Yöntemler

Optimum sonuca götüren en önemli kesin yöntemler karmaşık tam sayılı programlama ve doğrusal programlama yöntemleridir. Tamsayılı doğrusal programlama yaklaşımı özellikle iş ve mühendislik hayatında önemli bir yer edinmiştir (Laporte vd., 1986). Tamsayılı programlama, tüm çözüm değişkenlerinin tamsayı değerleriyle sınırlandırıldığı doğrusal programlamanın özel bir versiyonudur. Bazı problemlerde, çözüm değişkenlerinin belirli bir kısmı tamsayı değerinde olma gibi kısıtlamalar vardır. Bu tür problemlerin çözümü için karma tamsayılı doğrusal programlama kullanılır. Örneğin bir sürü İHA görev atama probleminde İHA sayısı kesirli bir sayı olamaz. Bazı durumlarda karışık tamsayılı doğrusal programlama formülasyonları oluşturmak kolay gibi görünse de, belirli formülasyon özellikleri doğrusal programlama tabanlı çözümlerin etkinliğini önemli ölçüde azaltabileceğinden çoğu zaman zor bir probleme dönüşebilmektedir.

Doğrusal programlama (DP), belirli kısıtlamalar altında mümkün olan en iyi sonucu bulmaya çalışan matematiksel bir optimizasyon yöntemidir. Sorunu denklemler olarak ifade eden DP, olası çözümlerin sınırlandırılması ve çözümlerin olası maksimum ve minimum değerlerinin yardımıyla, amaç fonksiyonu bir dışbükey çokyüzlü ile tanımlanır. Çözüm uzayı, bu dışbükey çokyüzlü tarafından kapsanan bölgedir. Optimum çözüm, amaç

fonksiyonunun maksimum veya minimum deęerini aldıęı noktadır. Herhangi bir kısıtlama durumunun olmadığı bir doğrusal programlama probleminde sadece amaç fonksiyonu vardır. Amaç fonksiyonu  $z$ , deęişkenler,  $x_1, x_2, \dots, x_n$  ve sabit katsayılar,  $c_1, c_2, \dots, c_n$  ile gösterilirse amaç denklemi:  $z = c_1x_1 + c_2x_2 + \dots + c_nx_n$  şeklinde ifade edilir. Burada amaç  $z$ 'yi maksimum ya da minimum yapacak  $x_1, x_2, \dots, x_n$  deęerlerinin bulunmasıdır. Amaç fonksiyonu genel olarak (3.5) numaralı denklemdeki gibi ifade edilebilir.

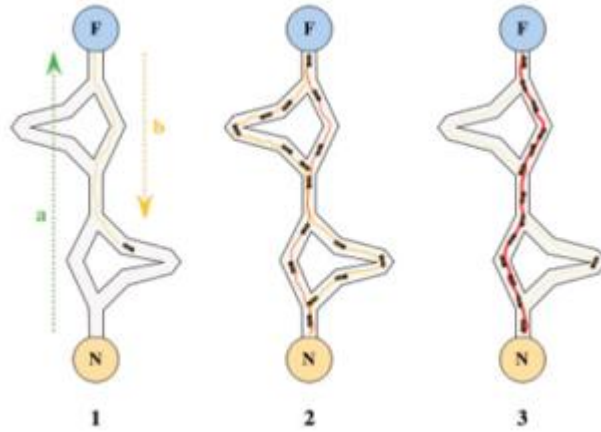
$$z_{\max/\min} = \sum_{j=1}^n c_j x_j, \quad (j=1, 2, \dots, n) \quad (3.5)$$

Doğrusal programlama, yöneylem araştırması alanında yaygın olarak kullanılan bir yöntemdir. Daha çok görev atama, planlama, üretim ve ulaşım gibi problemleri çözmek için kullanılır.

Dal ve Sınır yöntemi, düğümlerinde alt ve üst sınırlar ile çözüm uzayını tarayarak optimal çözümü arayan DP tabanlı ağaç arama yöntemidir. Optimal çözümü ararken üst sınır belirleme, dallandırma ve sınırlama olmak üzere üç farklı işlem yapılır. Çözüm uzayını köklü bir ağaç olarak varsayan dal ve sınır yöntemi, dallanma yaparak en uygun çözümü bulmak için alt problemlere ayrılırlar. Algoritma, bir dala girmeden önce, üst ve alt sınırların optimal bir çözüm olup olmadığını kontrol eder. Dal, mevcut çözümden daha iyi bir çözüm sağlayamazsa, o dal durum uzayından çıkarılır. Eğer düğümden elde edilen deęer üst sınırdan daha iyi ise üst sınır deęeri güncellenir. Bu yöntem özellikle NP-Zor problemler için kullanılırken gerçek zamanlı problemlerde kullanmak için hala çok yavaştır. Şekil 3.4'de görüleceęi üzere ağaç yapısı belirli sayıda düğümleri ve düğümleri birbirine bağlayan dalları olan ve döngü içermeyen bir veri yapısıdır. Ağaç veri yapısının elemanları düğüm, iki düğümü birleştiren bağlantılar da dal olarak ifade edilmektedir. 0 seviyesi kök düğüm olarak adlandırılırken 1'inci seviyede  $n$  tane düğüm vardır.  $(n-1)$ 'inci seviyede her bir düğüm  $(n-2)$  sayıda düğüme dallanır. Süreç her bir düğüm tek dal kalana kadar devam eder.



ise feromen kokusunun daha çok olduđu yolu daha az olduđu yola tercih ederler. Şekil 3.5’de görüldüğü gibi karıncalar yuvalarından yiyeceğe ulaşmak için en başta iki yoldan birini rastgele seçerler. Bir kısmı uzun olan yoldan giderken diğeri bir kısmı kısa olan yoldan ilerlerler. Kısa olan yolda feromen kokusu daha yoğun olacağı için arkandan gelen karıncalar kısa olan yolu seçme eğilimi gösterir. Böylece tüm karıncalar için en kısa yol yakınsanmış olur.



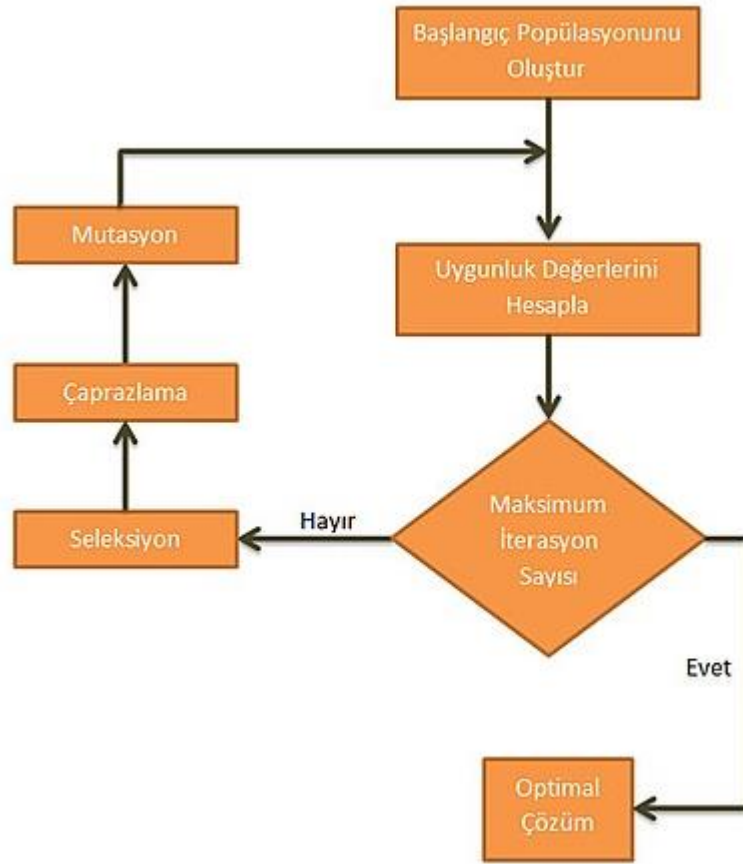
Şekil 3.5: Karıncaların gıda kaynağına gidecek en yakın yolu bulma yaklaşımı.

### **3.4.2.2 Genetik Algoritma**

Genetik algoritma doğal seçim ilkelerine dayanan daha iyi çözümler bulmayı hedefleyen bir optimizasyon yöntemi olarak tanımlanabilir. Karmaşık, çok boyutlu arama uzayında en iyinin hayatta kalması ilkesine göre küresel çözümü arar. GA yaklaşımı uzun çalışmaların neticesinde ilk defa John Holland tarafından ortaya atılmıştır ve öğrencisi David Goldberg tezinde; gaz boru hattının kontrolünü içeren bir problemin çözümünü GA ile gerçekleştirmiştir (Whitley, 1994). Genetik algoritma, problemlere tek bir çözüm üretmek yerine farklı çözümlerden oluşan bir çözüm kümesi üretir.

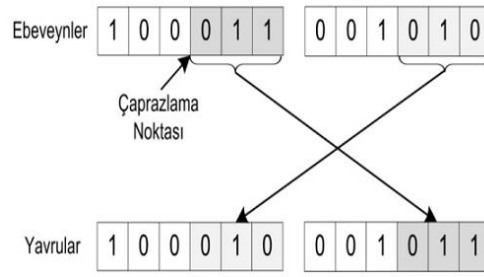
Algoritma, başlangıç popülasyonu adı verilen olası çözümlerin kodlandığı bir çözüm grubu elde edilerek başlatılır. Daha sonra problemin parametreleri uygun şekilde kodlanarak kromozomlar oluşturulur. Populasyonu oluşturan birey sayısı problemin türüne göre değişiklik gösterebilir.

Popülasyonki her bir kromozom, problemin çözümünde bir aday anlamına gelmektedir. Uygunluk fonksiyonu kullanılarak “çaprazlama” ve “mutasyon” genetik operatörleri ile daha iyi kromozomlar bulunmaya çalışılır. Ardı ardına gerçekleştirilen iterasyonlar ile yeni bireyler oluşturulur ve bu bireylerin uygunluk değerleri hesaplanır. Eğer bu bireylerin uygunluk değerleri popülasyondakilere göre daha iyi ise popülasyona dahil edilirler. Kuşak sayısı boyunca popülasyonda yer alan kromozomlar kendilerini daha iyi kromozomlarla yenilerler. İterasyon yeni birey sayısı belirli bir sayıya ulaştınca sona erer (Bkz. Şekil 3.6).



Şekil 3.6: Genetik algoritmanın şeması.

Popülasyonun tüm adaylarının uygunluk fonksiyonu hesaplanarak en kötü adaylar belirlenir ve diğer adaya çaprazlama uygulanır. Böylelikle uygunluk değeri en düşük adaylar elimine edilerek uygunluk değeri daha yüksek olan çocuk kromozomlar oluşturulur. Her çocuk kromozom, ebeveynlerden kromozomların karşıt kısımlarını alır. (Bkz. Şekil 3.7).



Şekil 3.7: Çaprazlama yöntemi

Genetik algoritmalarda mutasyon işlemcisi ile bir geni 1'den 0'a veya 0'dan 1'e çevirerek kromozomdaki bir genin değiştirilmesi sağlanır. Amaç popülasyondaki genetik çeşitliliği sağlamaktır. Belli bir süreden sonra nesildeki kromozomlar birbirini tekrarlama durumuna gelebilir ve yeni bir kromozom üretimi durabilir. Bu yüzden kromozomlardan bazıları mutasyona uğratılır.

### **3.4.2.3. Parçacık Sürü Optimizasyonu**

Sürü halinde hareket eden balık ve kuş sürülerinin sosyal etkileşimden esinlenerek ilk olarak Kennedy ve Eberhart tarafından geliştirilmiş olan PSO, popülasyon tabanlı bir optimizasyon yöntemidir (Marini, 2015). Her bireye parçacık, parçacıklardan oluşan popülasyona da sürü adı verilir. Her bir parçacık kendi bulunmuş olduğu pozisyonu, bir önceki tecrübesinden yararlanarak sürüdeki en iyi pozisyona sahip olan bireye göre ayarlar. PSO, temel olarak sürüde bulunan bireylerin pozisyonunun, sürünün en iyi pozisyona sahip olan bireyine yaklaştırılmasına dayanır. Bu yaklaşma hızı rasgele gelişen durumdur ve çoğu zaman sürü içinde bulunan bireyler yeni hareketlerinde bir önceki konumdan daha iyi konuma gelirler ve bu iterasyon hedefe ulaşıncaya kadar devam eder.

Ana parametreleri pozisyon ve hız vektörü olan PSO, diğer sürü zekâsı algoritmalarına göre daha az parametre içermektedir. Bu bilgiler her iterasyonda güncellenmektedir. Algoritma temel olarak aşağıdaki basamaklardan oluşur;

- i. Rasgele üretilen başlangıç pozisyonları ve hızları ile başlangıç sürüsü oluşturulur.

- ii. Sürü içerisindeki tüm parçacıkların uygunluk değerleri hesaplanır.
- iii. Her bir parçacık için mevcut jenerasyondan yerel en iyi (*pbest*) bulunur. Sürü içerisinde en iyilerin sayısı parçacık sayısı kadardır.
- iv. Mevcut jenerasyondaki yerel eniyiler içerisinde küresel en iyi (*gbest*) seçilir.
- v. Hız ve konum vektörleri takip eden denklemde olduğu gibi yenilenir.

$$v_{id}(t + 1) = w \cdot v_{id}(t) + c_1 \cdot rand_1 \cdot (pbest_{id} - x_{id}) + c_2 \cdot rand_2 \cdot (gbest_{id} - x_{id}) \quad (3.6)$$

$$x_{id}(t + 1) = x_{id}(t) + v_{id}(t + 1) \quad (3.7)$$

Denklem 3.6'da yeni hız değerlerini verirken,  $rand_1$  ve  $rand_2$  [0, 1] değerleri arasında rasgele üretilmiş sayılardır.  $W$  parçacığın hareketsizliğini önlemek için parçacıklara ivme kazandıran atalet ağırlık değerini ve  $c_1, c_2$  hızlanma sabitlerini göstermektedir. Denklem 3.7 kullanılarak yeni konum bilgisi elde edilir.

- vi. Maksimum iterasyon sayısına ulaşıncaya kadar 2, 3, 4 ve 5 adımlar tekrar edilir.

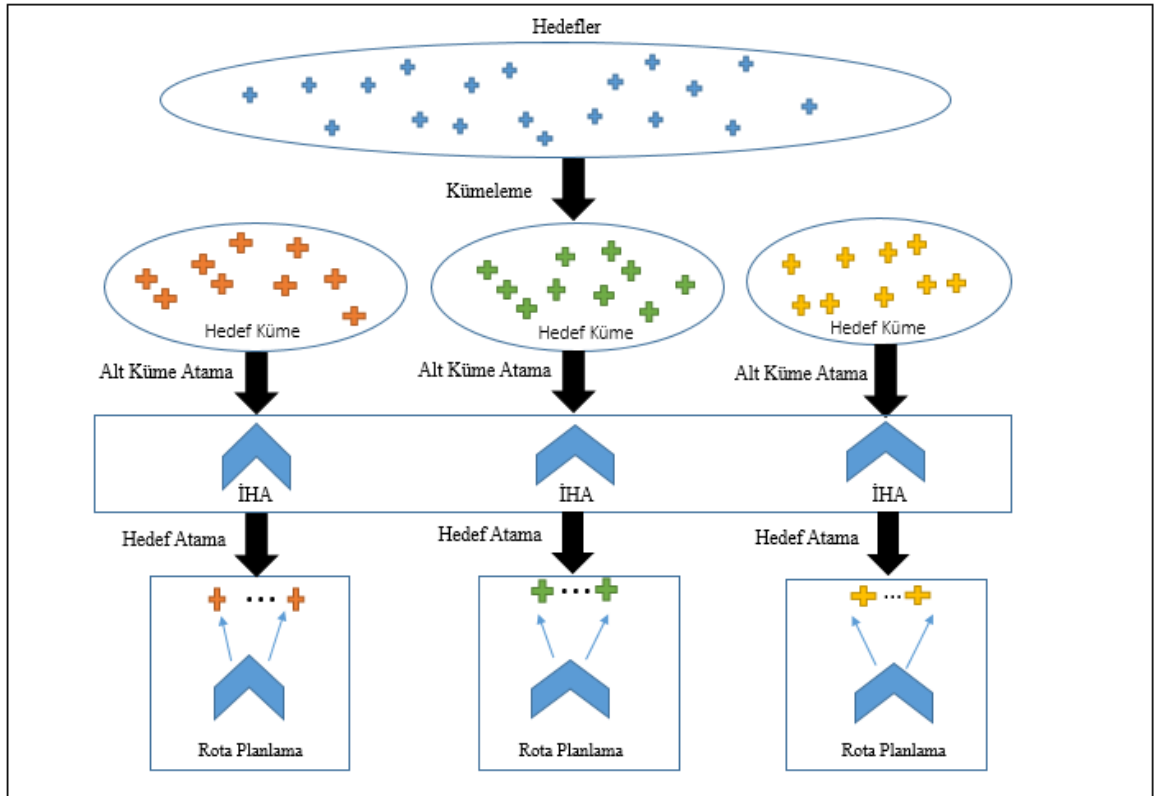
#### **3.4.2.4. Tavlama Benzetimi Algoritması**

TB, kombinatoriyal eniyileme problemlerini çözmek için ilk olarak 1953 yılında Metropolis tarafından önerilmiş ve daha sonra Kirkpatrick tarafından geliştirilmiş stokastik bir arama yöntemidir (Kirkpatrick vd., 1983). Uygulama kolaylığı sayesinde büyük boyutlu problemlerin çözümünde yaygın bir şekilde kullanılmaktadır. Özellikle gezgin satıcı problemlerinde kullanılan TB algoritması iteratif olarak ilerler ve optimum değeri bulmak için çeşitli hareket mekanizmaları kullanılır. TB'de amaç optimal sonuca yaklaşmak için yerel en iyilerden kurtularak genel en iyiye ulaşmaktır. Kabul olasılığı kavramsal bir sıcaklığa bağlı olarak  $e^{-\Delta E/T}$  şeklinde ifade edilir.  $\Delta E$  mevcut çözüm ile üretilen komşu çözümün amaç fonksiyonları arasındaki değişimi,  $T$  ise kontrol parametresi olan sıcaklığı ifade eder.  $\Delta E$  'nin büyük değerleri için kötü çözümün kabul edilme olasılığı, küçük

değerlerinkinden daha düşüktür. Ayrıca, yüksek sıcaklık değerlerinde üretilen yeni çözümlerin çoğu kabul edilecektir. Sıfıra yakın sıcaklıklarda üretilen yeni çözümlerin kabul edilme olasılığı düşükken yüksek sıcaklık değerlerinde üretilen yeni çözümlerin kabul edilme olasılığı oldukça yüksektir. Bu yüzden TB'de, başlangıç sıcaklığı genel olarak yüksek bir değer olarak alınır. TB'nin daha fazla bilgisayar zamanına ihtiyaç duyması ve uygun parametre seçimi için çok deneme gerektirmesi dezavantajlarıdır.

#### 4. MATERYAL VE YÖNTEM

Bu çalışmada otonom insansız hava araçlarının, belirlenen hedef lokasyonlarına giderek önceden tanımlanmış olan keşif, gözetleme gibi görevleri başarılı bir şekilde gerçekleştirmesi için bir görev atama problemi sunulmuştur. Problemden birden fazla İHA ve çok sayıda hedef tanımlanmıştır. Tüm İHA'lar homojen özelliklere sahiptir. Görev planlamasında hedefler ve her bir İHA'nın yer istasyonu girişleri, İHA'lara yapılan rota planlaması ise çıkışları oluşturmaktadır. Otonom uçuş yapan İHA'ların vazifelerini tamamladıktan sonra tekrar başlangıç noktalarına dönecekleri varsayılmıştır. Önerilen yöntemin kümeleme, İHA-alt-küme atama ve rota planlama'dan oluşan aşamaları Şekil 4.1'de özetlenmiştir.

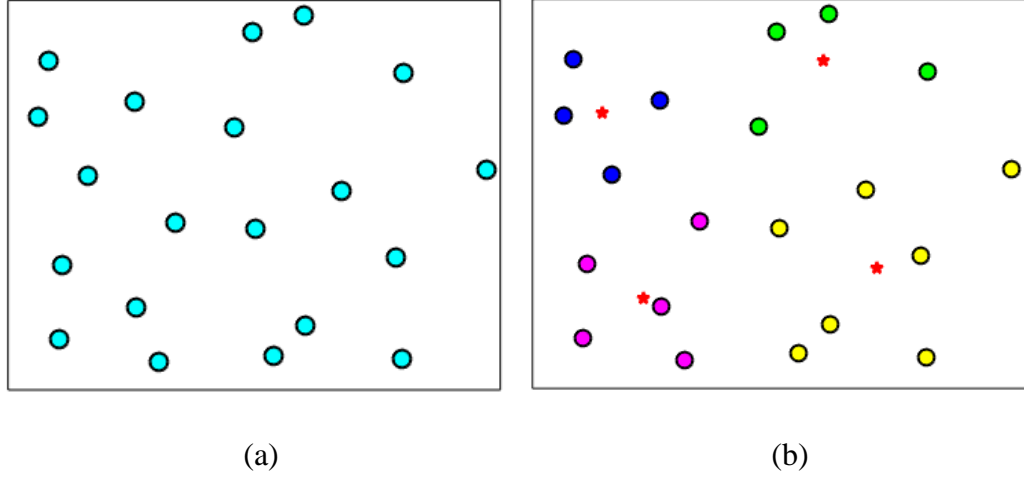


Şekil 4.1: Önerilen yöntemin tüm süreçlerini gösteren şema.

Bu tezde önerilen görev atama probleminde üç farklı kısıtlama getirilmiştir; tanımlanan tüm hedef noktalarına varılması, her bir hedefe sadece bir İHA'nın atanması ve İHA'ların toplam kat ettikleri mesafenin minimum olması. Bu kısıtlar altında görev atama problemini gerçekleştirebilmek için kümeleme, görev atama ve rota planlamadan oluşan üç aşamalı bir yöntem önerilmiştir. Birinci aşama, hedef noktalarının İHA sayısı kadar kümeye bölünmesi işlemidir. Bu aşamada, oldukça yaygın olarak kullanılan  $k$ -ortalamlar kümeleme algoritması ve alternatif olarak graf tabanlı bir yöntem olan spektral kümeleme algoritması kullanılmıştır. İkinci aşama kümeleme sonrasında her bir alt kümenin merkezine göre  $N$  İHA -  $N$  hedef küme arasında görev atama problemidir. Bu problem, literatürde kısaca Macar algoritması olarak bilinen Kuhn-Munkres algoritması ile çözülmüştür. Üçüncü ve son aşama ise, her bir İHA'nın kendisine atanan alt kümedeki hedef noktalarını minimum mesafe ile kat etmesidir. Bu problemin çözümü için ilgili alanda en etkili yöntemlerden biri olan karınca kolonisi optimizasyonu kullanılmıştır.

#### 4.1 Hedef Noktalarını Kümelenendirme

Önerdiğimiz yöntemde belirlenen hedef sayısının görev planlamasında görev alacak İHA sayısından fazla olduğu varsayılmıştır. Bu aşamada kümeleme algoritmaları kullanılarak belirlenen hedef noktaları İHA sayısına eşit olacak şekilde kümelere ayrılmıştır. Küme sayısı görev alacak İHA sayısına göre belirlenmiştir. Şekil 4.2'de 20 adet hedef noktadan oluşan örnek bir veri setinin dört kümeye ayrıldığı gösterilmiştir. Aynı renkle gösterilen hedef noktaları aynı kümeye ayrılan örnekleri ifade etmektedir. Yuvarlak noktalar hedef noktalarını, kırmızı yıldızlar ise küme merkezlerini temsil etmektedir.



Şekil 4.2:  $K$ -ortalamalar kümeleme yöntemi ile verilerin dört kümeye ayrılmasının gösterimi. (a) Kümelemeden önceki veriler, (b) Kümelere ayrılan örnekler ve merkez noktaları.

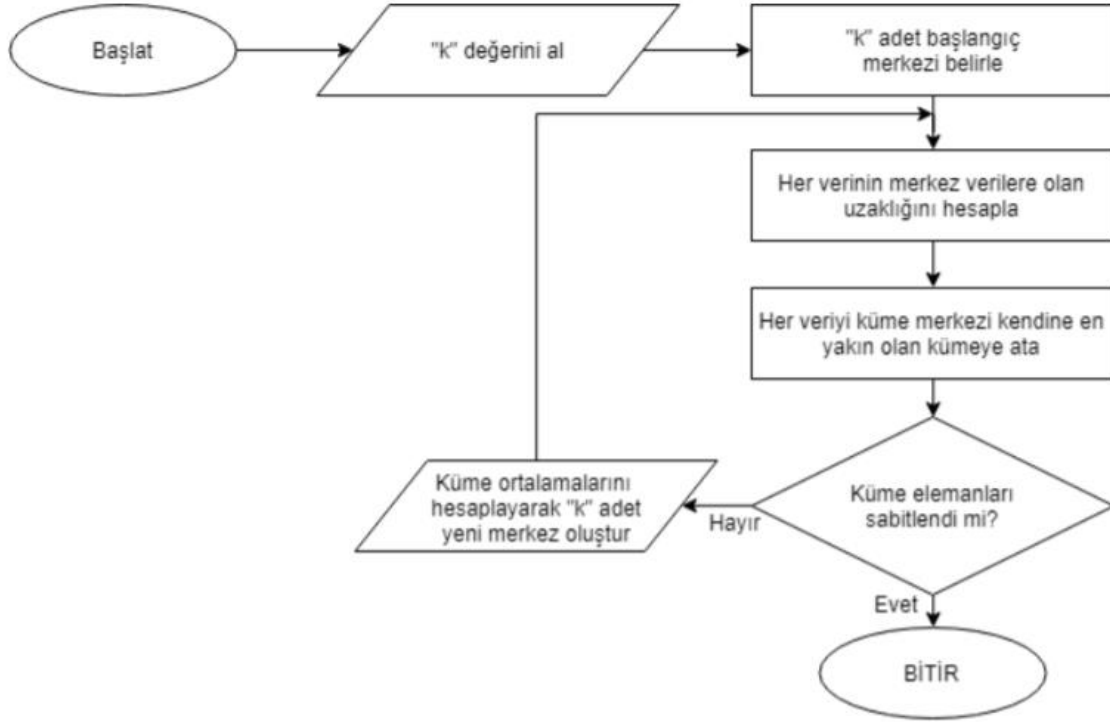
$M$  elemanlı veri seti  $X = \{x_1, \dots, x_M\}$ ,  $k$  küme sayısı ve yeni oluşan kümeler  $S = \{S_1, \dots, S_k\}$  olmak üzere algoritma aşağıdaki basamaklardan oluşur:

- i.  $X = \{x_1, \dots, x_2\}$  veri setinden rastgele  $k$  adet küme merkezi seçilir.
- ii. Tüm veri noktalarının küme merkezlerine olan uzaklıkları hesaplanır.
- iii. Herbir nokta kendisine en yakın olan kümeye atanır.
- iv. Denklem 4.1'de verilen formül ile her bir kümedeki verilerin, küme merkezine ( $\mu_i$ ,  $i=1, \dots, k$ ) olan uzaklıkları minimize edilir.

$$\arg \min_S \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\| \quad (4.1)$$

v. Denklem 4.1'deki değerin değişimi yeterince küçük oluncaya veya kümesi değişen örnek bulunmayana kadar bu işlemler devam eder.

$K$ -ortalamlar kümeleme yönteminin uygulama akış diyagramı ise Şekil 4.3'de gösterilmiştir.



Şekil 4.3:  $K$ -ortalamlar kümeleme yöntemi akış diyagramı

#### 4.2 Alt Küme-İHA Ataması

Atama problemi en çok bilinen kombinatoriyal optimizasyon problemlerinden biridir. Amaç,  $N$  İHA'nın -  $N$  hedef kümeye en düşük maliyetle eşleştirilmesinin sağlanmasıdır. İHA-alt küme eşleştirmesi yaparken her bir İHA'nın alt kümelerin merkezlerine olan uzaklıkları dikkate alınarak atama yapılır. Bu tez çalışmasında, bu atama problemini çözmek için 1955 yılında Kuhn tarafından geliştirilen polinom zamanlı Macar algoritması kullanılmıştır. Macar Algoritması ilgili problemin çözümünde kesin çözüm veren yöntemlerin başında gelmektedir. Atama problemi Macar Algoritmasıyla çözümlenirken satırlar ajanları sütunlar ise hedefleri (ya da tam tersi) ifade eden ve her bir  $(i, j)$  elemanı  $i$ 'inci ajanın  $j$ 'inci hedefe atanması durumunda ortaya çıkacak maliyeti gösteren bir matris şeklinde ifade edilir. Örneğin, 4 İHA ( $M_1, M_2, M_3, M_4$ ) ve 4 alt-küme ( $N_1, N_2, N_3, N_4$ ) bulunan bir atama

probleminde her İHA'nın yalnız bir alt kümeye atanması gerekmektedir. Bu problemi minimum maliyetle çözmek için her bir İHA'nın alt küme merkezlerine olan uzaklıklarına göre maliyet matrisi oluşturulur. Şekil 4.4'de her İHA için atanan bir alt kümenin maliyeti gösterilmektedir.

	N1	N2	N3	N4
M1	704	1030	677	268
M2	438	853	1047	825
M3	981	699	391	839
M4	811	396	889	1146

Şekil 4.4: Maliyet matrisi

Şekil 4.5'de optimum İHA ve hedef alt küme atamaları gösterilmiştir. Sonuç matrisine göre M1 İHA N4 kümesine, M2 İHA N1 kümesine, M3 İHA N3 kümesine atanırken M4 İHA ise N2 kümesine atanmıştır.

	N1	N2	N3	N4
M1	0	0	0	1
M2	1	0	0	0
M3	0	0	1	0
M4	0	1	0	0

Şekil 4.5: Optimum atama matrisi

Eğer atama problemi  $N \times N$  boyutlarında bir matris ile ifade edilecek olursa Macar Algoritmasının adımları aşağıdaki gibi özetlenebilir:

*Adım-1:* Atama matrisinin her bir satırı için, satırdaki en küçük değere sahip olan eleman tespit edilip satırdaki tüm elemanlardan çıkartılarak yeni bir matris elde oluşturulur.

*Adım-2:* Adım-1'deki işlemten sonra aynı işlem sütunlar için de yapılır. Bu işlemin ardından sonra elde edilen yeni matris "indirgenmiş matris" olarak tanımlanmaktadır.

*Adım-3:* İndirgenmiş matriste bulunan bütün sıfır elemanlarını kapatmak için en az sayıda çizgi kullanılarak gerekli satır ve sütunların üzerine çizgi çekilir. Kullanılan çizgi

sayısı maliyet matrisinin boyutuna eşit ise optimal çözüm bulunur ve atama belirlenir. Çizgi sayısı satır sayısından küçükse Adım-4'e gidilir.

*Adım-4:* Sıfır elemanlarını kapatmak için satır ve sütunlar üzerine çizilmiş çizgilerin kapatmadığı matris elemanlarından en küçük değere sahip olanını tespit edilir. Bu değer üzerinden çizgi geçmeyen tüm elemanlardan çıkarılıp üzerinden iki çizgi geçen elemanlara eklendikten sonra Adım-3'e gidilir.

### 4.3 Rota Planlama

Araç Rotalama Problemi (ARP) çözüm bölgesi  $G(V,E)$  grafının İHA'ların tüm kalkış noktalarını ve tüm hedef konumlarını içerdği bir uçuş ağını gösteren kombinasyonel bir problemdir. Düğüm kümesi  $V = \{0,1,\dots,n\}$ , yay kümesi  $E=\{i, j): i, j \in V, i \neq j\}$  ile temsil edilirken, 0 (sıfır) düğümü başlangıç noktasını ve diğer  $n$  sayıdaki düğümler hedef noktalarını ifade etmektedir.  $d_{ij}$   $i$  ve  $j$  ( $i, j \in V$ ) noktaları arasındaki mesafeyi göstermektedir. Uçuş ağında  $k$  numaralı İHA  $i$  noktasından  $j$  noktasına hareket ederse 1, aksi takdirde 0 (sıfır) değerini alacak olan ikili karar değişkenini  $x_{ijk}$  olarak varsayalım. İHA sayısını  $M$  adet, hedef sayısını da  $N$  adet olarak düşünürsek oluşturulan modelin amaç fonksiyonu ve kısıtlarını, aşağıda verilen tamsayılı doğrusal programlama problemi ile formüle edebiliriz.

Amaç fonksiyonu:

$$\text{Min } Z = \sum_{i=0}^N \sum_{j=0, j \neq i}^N \sum_{k=1}^M d_{ij} \cdot x_{ijk} \quad (4.2)$$

Kısıtlar:

$$\sum_{k=1}^M \sum_{j=1}^N x_{ijk} = M, \quad t = 0 \text{ ise} \quad (4.3)$$

$$\sum_{k=1}^M \sum_{j=1, j \neq i}^N x_{ijk} = 1, \quad \forall i \in \{1, \dots, N\} \quad (4.4)$$

$$\sum_{k=1}^M \sum_{i=1, i \neq j}^N x_{ijk} = 1, \quad \forall j \in \{1, \dots, N\} \quad (4.5)$$

$$\sum_{i=1}^N x_{i0k} \leq 1, \quad \forall k \in \{1, \dots, M\} \quad (4.6)$$

$$\sum_{i=1}^N q_i \sum_{j=0, j \neq i}^N x_{ijk} \leq C, \quad \forall k \in \{1, \dots, M\} \quad (4.7)$$

(4.2) numaralı denklemdeki amaç fonksiyonu toplam kat edilecek mesafenin yani maliyetin en aza indirgenmesinin hedeflendiğini göstermektedir. Denklem 4.3’de, yer istasyonundan  $M$  adet İHA’nın çıkacağını, Denklem 4.4 ve 4.5’teki kısıtlar her bir hedef noktasının yalnızca tek bir İHA tarafından ziyaret edilmesi gerektiğini ve hedef noktalarına ulaşılmasını sağlayan yollardan sadece bir tanesinin kullanabileceğini ifade etmektedir. (4.6) numaralı denklemdeki kısıt her bir İHA’nın yer istasyonundan sadece bir kere çıkacağını dolayısıyla rotalamada bir defa kullanılacağını ve son olarak Denklem 4.7’deki numaralı kısıt ise İHA’lara yapılan yüklemelerin araç kapasite değeri  $C$ ’yi geçmemesi gerektiğini belirtmektedir.

Yukarıda bahsedilen ARP’nin çözümü için farklı yaklaşımlar ve birçok algoritma önerilmiştir. Üzerinde çok sayıda optimizasyon geliştirilen ARP kesin ve sezgisel olmak üzere iki gruba ayrılabilir. Problem çözümünde kesin algoritmalar düşük performans göstermesine rağmen birçok sezgisel yaklaşımlı algoritma kısa zaman içinde iyi sonuçlar ortaya koyarlar. Bilinen bu tekniklerden birisi de yapay zekâ sisteminin sürü zekâsı içerisinde yer alan karınca kolonisi algoritmasıdır (KKA). Bu tez çalışmasında kullandığımız KKA gezgin satıcı probleminin çözümünde oldukça iyi sonuçlar vermektedir. GSP’de ilk

önce  $i$  ve  $j$  hedefleri arasındaki mesafeyi bulmak için takip eden denklemdeki Öklit uzaklığı kullanılır.

$$d_{ij} = \left[ (x_i - x_j)^2 + (y_i - y_j)^2 \right]^{1/2} \quad (4.8)$$

Karıncalar  $n$  iterasyonun ardından feromon güncellemesi gerçekleştirirler. Bu güncellenmenin formülü aşağıdaki denklemde gösterildiği gibi ifade edilebilir.

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij} + \sum_{k=1}^N \Delta\tau_{ij}^k \quad (4.9)$$

Buharlaştırma miktarını gösteren  $\rho \in (0,1]$  değer alır ve  $\Delta\tau_{ij}$   $k$ 'inci karıncanın  $t$  ve  $(t+n)$  zamanda  $(i,j)$  yoluna bıraktığı feromon miktarını gösterir.

$$\Delta\tau_{ij}^k = \begin{cases} Q/L_k & , \text{ eğer karınca } k \text{ kenar}(i,j) \text{ turunda kullandıysa,} \\ 0 & , \text{ diğer türlü} \end{cases} \quad (4.10)$$

Her bir karıncanın belirlenen  $\alpha$  ve  $\beta$  değerlerine göre bir düğümden bir sonraki düğüme geçiş olasılığının matematiksel formülü Denklem 4.11'de gösterilmiştir.

$$p_k(i,j) = \begin{cases} \frac{[\tau(i,j)]^\alpha \times [n(i,j)]^\beta}{\sum_{u \in j_k(i)} [\tau(i,u)]^\alpha \times [n(i,j)]^\beta} & \text{eğer } j \in j_k(i) \\ 0 & \text{Diğer durumlarda} \end{cases} \quad (4.11)$$

## 5. BULGULAR VE TARTIŞMA

Bu tez çalışmasında sürü İHA görev planlama probleminin çözümü için hiyerarşik bir algoritma önerilmiştir. Bu bölümde, önerilen yöntem diğer yöntemlerle karşılaştırılarak performans testi yapılmıştır. Bunun için 4 farklı senaryo kurgulanmıştır. Simülasyon ortamında gerçekleştirilen testler, MATLAB kullanılarak Windows 10 İşletim sistemi, intel core i5-3230M CPU 2.60 GHz işlemci, 4 GB RAM'e sahip bir dizüstü bilgisayar kullanılarak gerçekleştirilmiştir. Test edilen veriler 20, 40 ve 60 hedeften oluşmaktadır. Problemlerin türü yer istasyonlarının konumu ve sayısına göre değişiklik göstermiştir. Herbir durum 10 kez test edilerek ortalama sonuçlar alınmıştır. Hedef sayısını  $N$ , İHA sayısını da  $M$  ile ifade edecek olursak önerdiğimiz yöntemde hedef sayısı İHA sayısından fazla olacak ( $N > M$ ) şekilde çalışılmıştır.

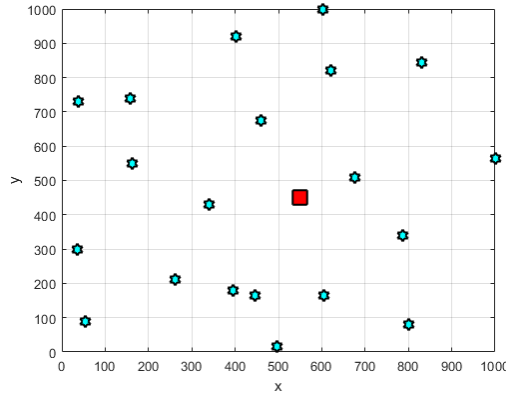
### 5.1. Simülasyon Testleri

Bu tez çalışmasında farklı veriler kullanılarak 4 farklı senaryo hazırlanmıştır. Yer istasyonun konumu ve sayısına göre değişen senaryolarda konum bilgisi olarak  $1000 \times 1000$  birim aralığında kartezyen konumları kullanılmıştır. İHA'lar yer istasyonundan harekete geçmeden önce belirlenen hedeflerin ilgili İHA'lara ataması yapılmıştır. Otonom uçuş yapan İHA'ların kendilerine atanan tüm hedefleri en kısa yol üzerinden gezdikten sonra tekrar kalkış yerlerine dönecekleri varsayılmıştır. İlk üç senaryoda İHA'ların gözetleme, son senaryoda ise kısıtlı sayıda mühimmat taşıdığı varsayılan SİHA'lar ele alınarak bombalama görevi yaptığı farzedilmiştir. Farklı sayıda İHA (ya da SİHA) ve farklı sayıda sabit hedef noktaları kullanılarak önerilen yöntem test edilmiştir.

#### 5.1.1 Senaryo-1

Hazırlanan birinci senaryoda sadece bir tane yer istasyonu vardır ve (450,550) konumunda olduğu varsayılmıştır. Hedefler ise istasyonun etrafında rastgele dağıtılmıştır.

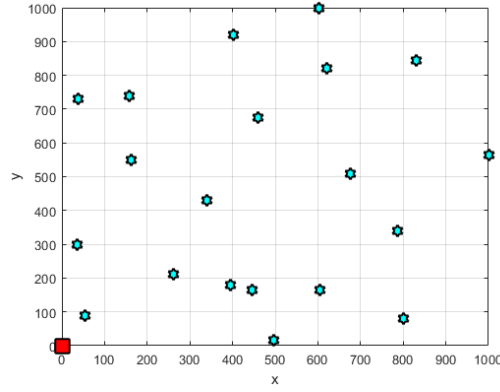
Herbir İHA homejen özelliklere sahip olup verilen görevi gerçekleştirmek üzere başlangıç noktası olan yer istasyonundan kalkış yapıp tüm hedefleri gezerek tekrar başlangıç noktasına geri dönmek zorundadır. Her hedef yalnız bir İHA'ya atanabilir ve herbir İHA her hedefe sadece bir kez uğramalıdır. Şekil 5.1'de görselleştirilen senaryoda kırmızı-kare yer istasyonunu, turkuvaz-yıldızlar ise ataması yapılacak hedefleri ifade etmektedir.



Şekil 5.1: Senaryo-1'in gösterimi.

### 5.1.2 Senaryo-2

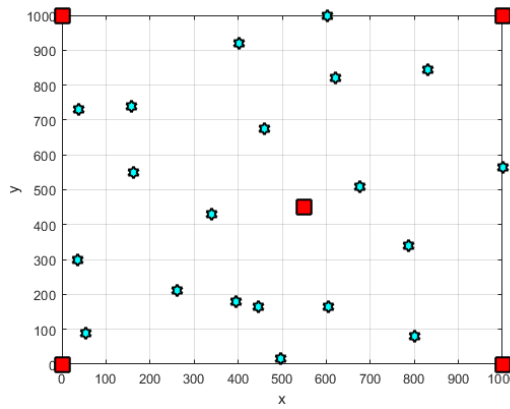
İkinci senaryoda da bir tane yer istasyonu kullanılmış fakat bu sefer konumu merkezde (0,0) olacak şekilde tasarlanmıştır. Homejen özelliklere sahip olan herbir İHA ilk senaryodakine benzer şekilde kendisine atanan tüm hedefleri gezip tekrar başlangıç noktası olan yer istasyonuna geri döneceği varsayılmıştır. Şekil 5.2'de görselleştirilen senaryoda kırmızı-kare yer istasyonunu, turkuvaz-yıldızlar ise ataması yapılacak hedefleri ifade etmektedir.



Şekil 5.2: Senaryo-2'nin gösterimi.

### 5.1.3 Senaryo-3

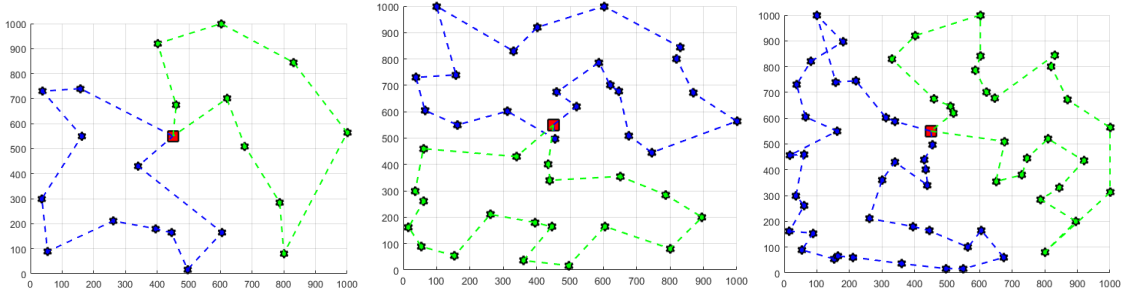
Üçüncü senaryoda diğerlerinin aksine dört veya dörtten fazla yer istasyonu kullanılarak bunların dört tanesi kartezyen koordinatlarının farklı köşelerine geriye kalanlar ise farklı konumlara yerleştirilmiştir. İHA'lar homejen özelliklere sahip olup ilgili hedeflerde gözetleme görevi yaptıktan sonra kalkış yaptıkları yer istasyonlarına geri dönecekleri varsayılmıştır. Şekil 5.3'de görselleştirilen senaryoda kırmızı-kareler yer istasyonlarını, turkuvaz-yıldızlar ise ataması yapılacak hedefleri ifade etmektedir.



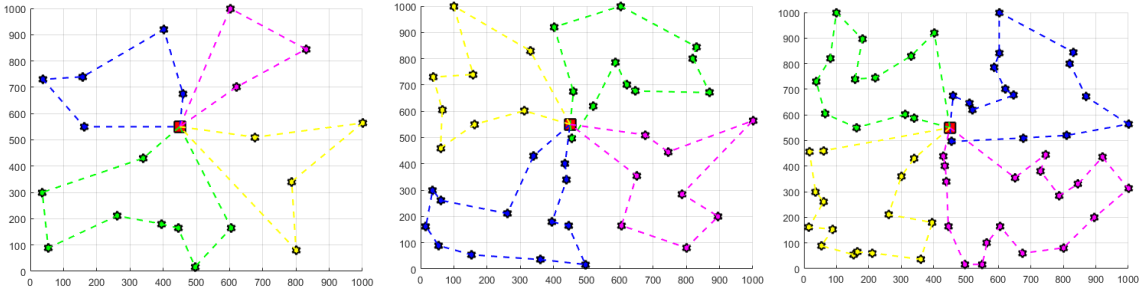
Şekil 5.3: Senaryo-3'ün gösterimi.

## 5.2 Test Sonuçları

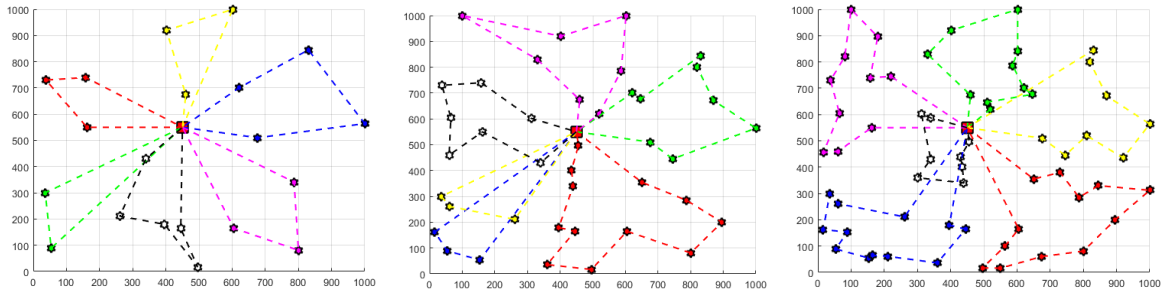
Tüm senaryolardaki testler 2 boyutlu çalışma ortamlarında yapılmıştır. Yer istasyonunun (550,450) konumunda olduğu ilk senaryoda 20, 40 ve 60 adet hedefler için 2, 4, 6 ve 8 adet İHA görevlendirilerek elde edilen test sonuçları Şekil 5.4’de gösterilmiştir.



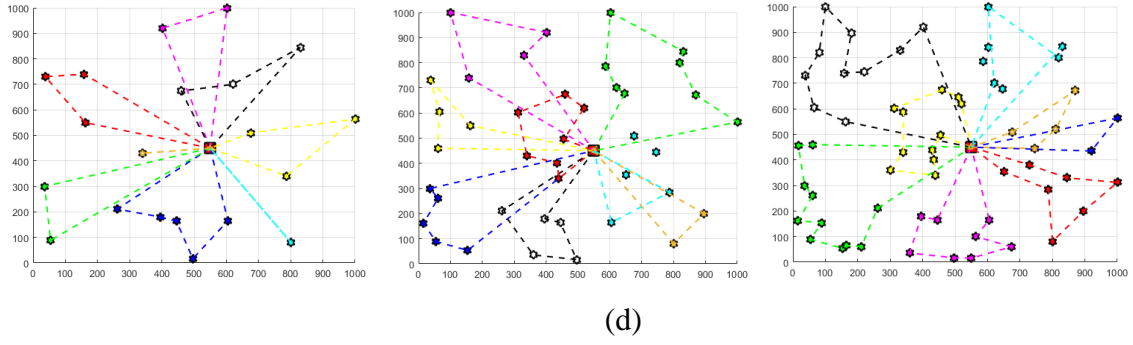
(a)



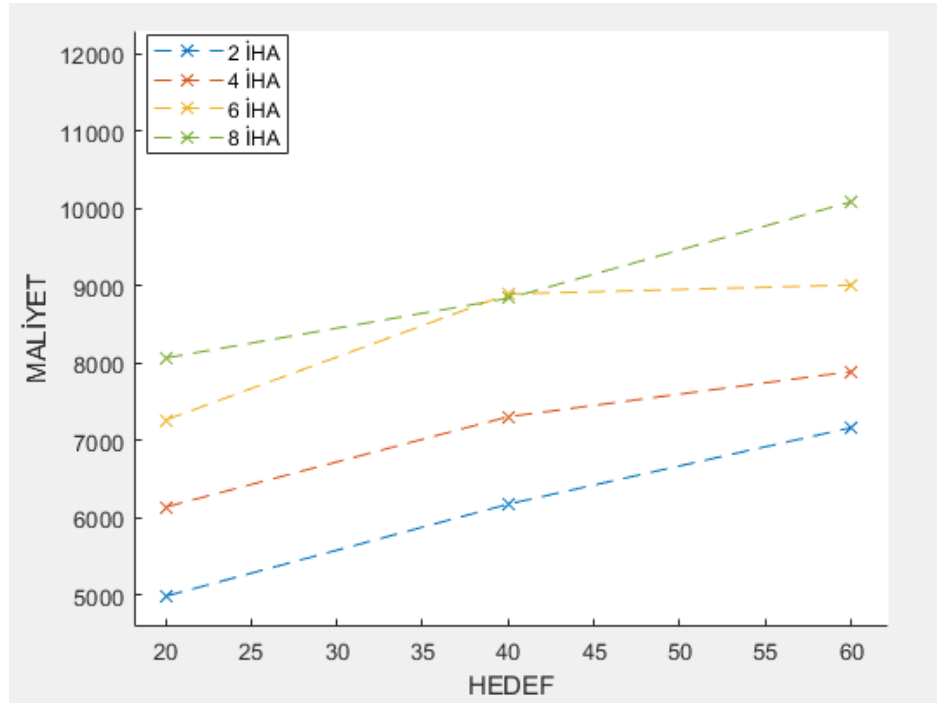
(b)



(c)



Şekil 5.4: Yer istasyonunun (550,450) konumunda olduğu durum için hedef atama sonuçları.  
 (a) D=2, H=20, 40, 60 (b) D=4, H=20, 40, 60 (c) D=6, H=20, 40, 60 (d) D=8, H=20, 40, 60

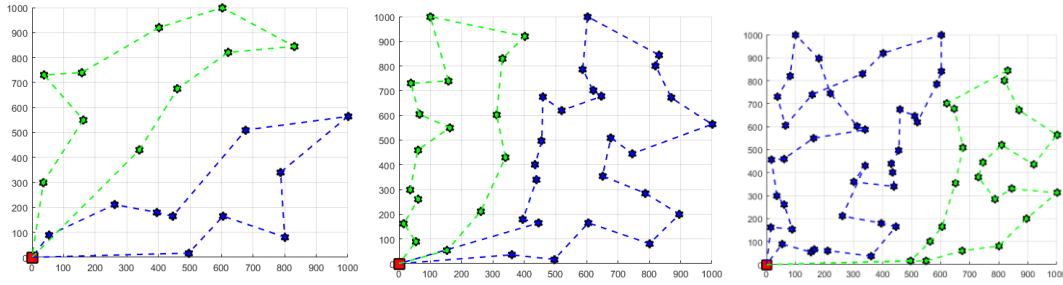


Şekil 5.5: Görevlendirilen İHA sayısına göre toplam maliyetin gösterimi.

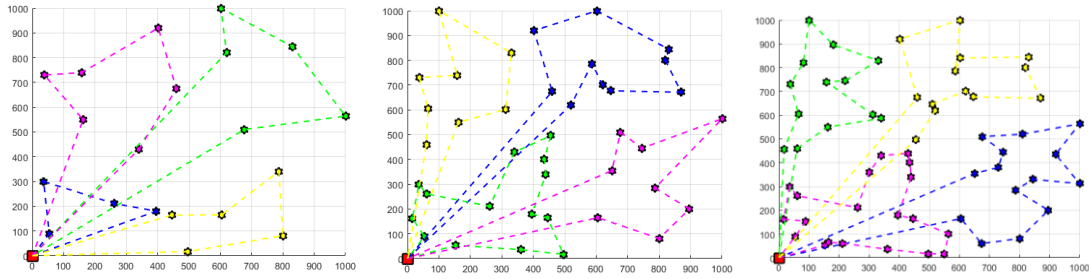
Her bir işlem 10 kez arka arkaya gerçekleştirilerek elde edilen sonuçların ortalaması alınarak grafiğe aktarılmıştır. Şekil 5.5'de görüldüğü üzere aynı sabit hedeflerde görevlendirilen İHA sayısı ile toplam maliyet oranı doğru orantılı olarak artmaktadır. Fakat hedef sayısının 40 olduğu senaryo için görevlendirilen İHA sayısının 6 olduğu durumda

farklı bir durum gözlenmiştir. 6 tane İHA'nın kullanımında hesaplanan toplam maliyet 8 İHA kullanımına göre daha yüksek maliyetli olduğu tespit edilmiştir. Diğer durumlarda ise aynı hedef sayısında atanan İHA sayısı arttığı zaman maliyet de artmıştır.

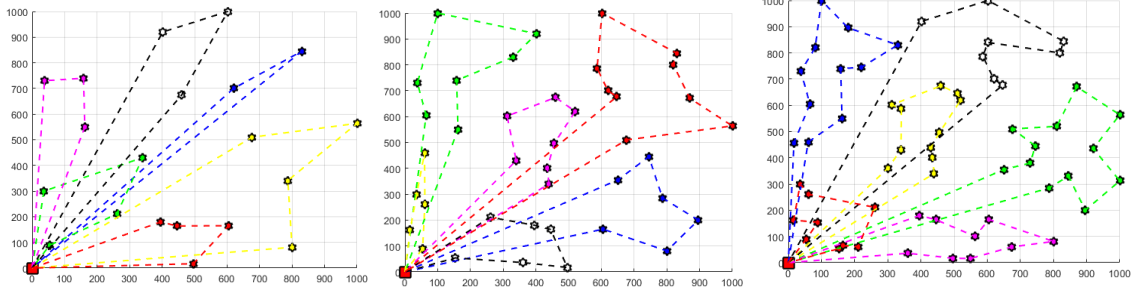
Yer istasyonunun merkez(0,0) konumunda olduğu varsayılan ikinci senaryoda aynı yerden hareket eden tüm İHA'lar gözetleme görevlerini yaptıktan sonra tekrar başlangıç noktasına geldikleri varsayılmıştır. Farklı sayılarda İHA ve hedefler kullanılarak önerilen yöntem test edilmiştir. Örnek sonuçlar Şekil 5.6'da gösterilmektedir.



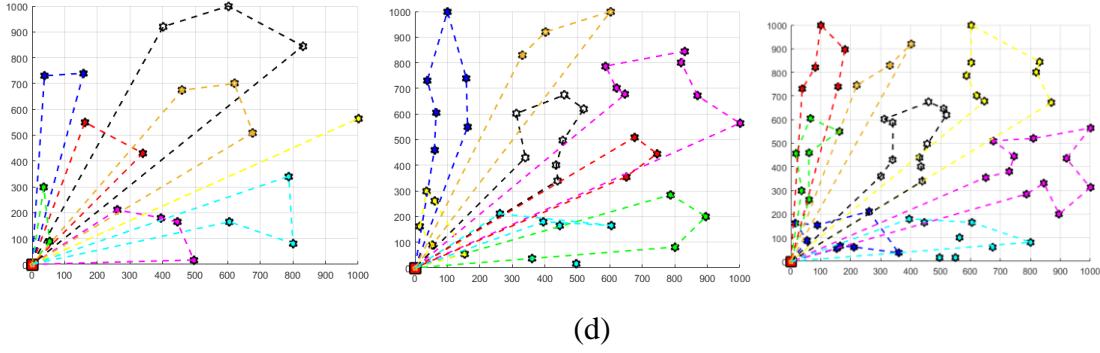
(a)



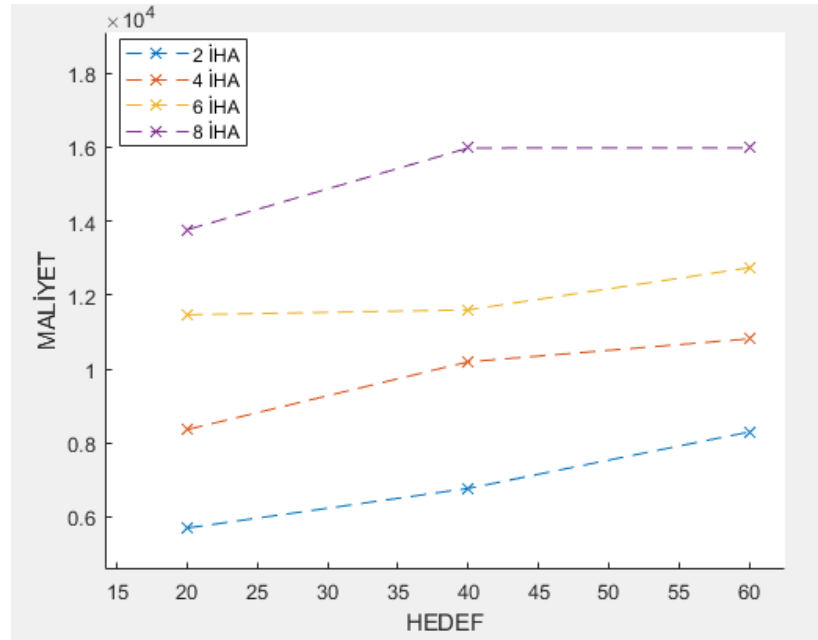
(b)



(c)



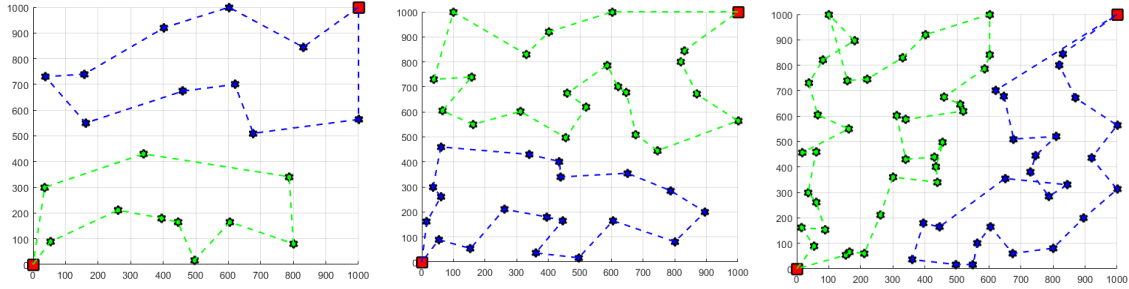
Şekil 5.6: Yer istasyonunun (0, 0) konumunda olduğu durum için atama sonuçları. (a) D=2, H=20, 40, 60 (b) D=4, H=20, 40, 60 (c) D=6, H=20, 40, 60 (d) D=8, H=20, 40, 60



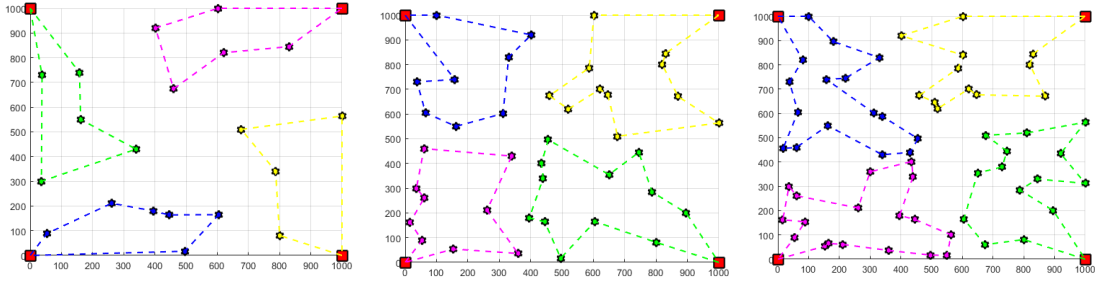
Şekil 5.7: Görevlendirilen İHA sayısına göre toplam maliyetin gösterimi.

10 kez arka arkaya gerçekleştirilen her bir işlemde elde edilen sonuçların ortalaması alınarak grafiğe aktarılmıştır. Şekil 5.7’de görüldüğü gibi her bir hedef sayısı için İHA sayısı arttıkça toplam maliyet miktarı da artmaktadır. Hedef sayısının 40 olduğu senaryoda 4 ile 6 İHA’nın görevlendirildiğinde hesaplanan toplam maliyetleri yaklaşık olarak aynıdır. Bu ikinci senaryoda görevlendirilen İHA sayısının artması her zaman maliyeti artırmıştır.

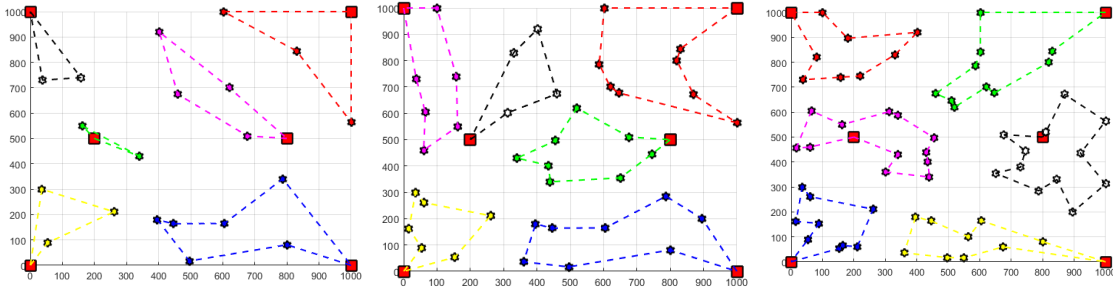
Üçüncü senaryoda ise farklı konumlarda olmak üzere birden fazla yer istasyonu kullanılmıştır. Yer istasyonlarından kalkış yapan İHA'lar kendilerine atanan hedeflerde gözetleme görevini yaptıktan sonra tekrar eski yerlerine dönecekleri varsayılmıştır. Herbir hedef yalnız bir İHA tarafından ziyaret edilecektir. Farklı sayılarda İHA ve hedef sayıları kullanılarak yapılan test sonuçları Şekil 5.8'de gösterilmiştir.



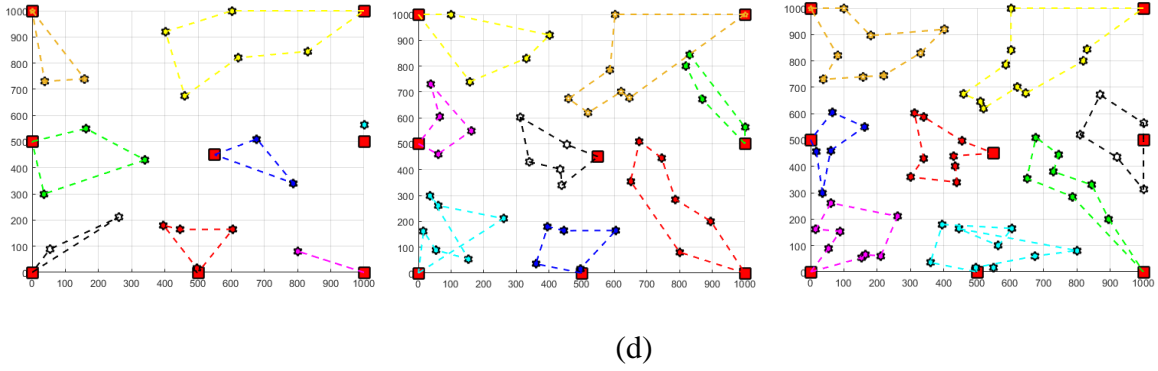
(a)



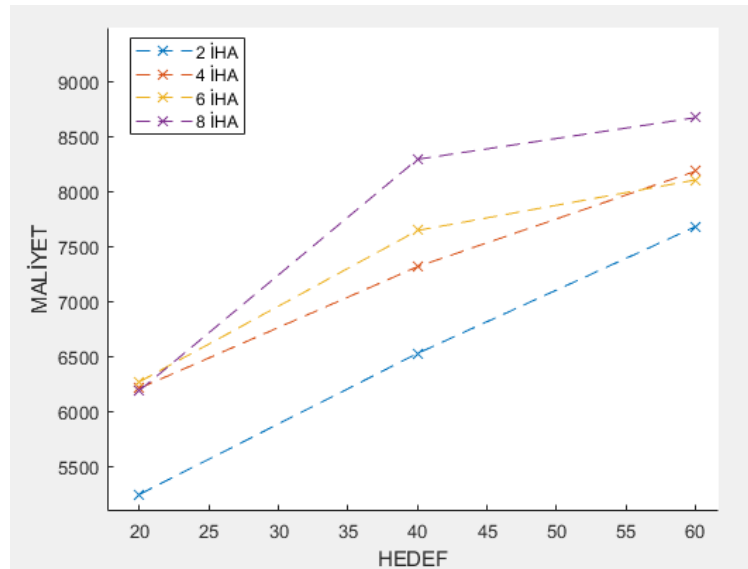
(b)



(c)



Şekil 5.8: Yer istasyonunun farklı konumlarda olduğu durum için atama sonuçları. (a) D=2, H=20, 40, 60 (b) D=4, H=20, 40, 60 (c) D=6, H=20, 40, 60 (d) D=8, H=20, 40, 60



Şekil 5.9: Görevlendirilen İHA sayısına göre toplam maliyetin gösterimi.

Her bir işlem 10 kez tekrar edilerek elde edilen sonuçların ortalaması alınmış ve grafiğe aktarılmıştır. Şekil 5.9'da görüleceği üzere toplam maliyetler incelendiğinde İHA sayısı arttıkça toplam maliyet miktarı da genel olarak artmıştır. 20 hedef sayısı için 8 İHA görevlendirmek 4 veya 6 İHA görevlendirmeye göre daha az maliyetli olduğu görülmektedir. Aynı senaryo için 2 İHA'nın görevlendirilmesinde hesaplanan toplam maliyette ise diğerlerine göre oldukça düşüktür. 40 ve 60 hedefin olduğu diğer senaryolarda da 2 İHA kullanmak maliyetten büyük oranda tasarruf sağlayacaktır. 60 hedefin olduğu

senaryo için ise 4 İHA yerine 6 İHA görevlendirildiğinde daha verimli bir atama görevi yapılmış olacaktır.

Çizelge 5.1. Farklı sayıda İHA ve hedef kullanılarak elde edilen toplam maliyet değerleri

Hedef	İHA	Genetik Algoritma	Tavlama Benzetimi Algoritması	Önerilen Hiyerarşik Yöntem
20	2	<b>4759</b>	5115	4989
	4	<b>6056</b>	6245	6141
	6	<b>7179</b>	7410	7265
	8	8154	8226	<b>8067</b>
40	2	6332	6421	<b>6179</b>
	4	7455	7564	<b>7459</b>
	6	8815	8925	<b>8678</b>
	8	8963	10052	<b>8858</b>
60	2	7296	7350	<b>7167</b>
	4	11072	13256	<b>8196</b>
	6	9213	9867	<b>8152</b>
	8	10960	11985	<b>10085</b>

20, 40, ve 60 adet sabit hedefler için 2, 4, 6 ve 8 adet İHA'nın izleyeceği rotalarda toplam maliyet verileri Çizelge 5.1'de gösterilmiştir. Bu verilere bakılarak 20 adet hedeften oluşan çözüm uzayında 2, 4 ve 6 adet İHA'nın görev aldığı durumda en başarılı sonuçları genetik algoritma vermiştir. Hedef sayısının 40, İHA sayısının 4 adet olduğu senaryoda genetik algoritma önerilen yönteme göre çok az bir farkla daha başarılı bir sonuç göstermiştir. Hedef ve İHA sayısının çok fazla olduğu diğer senaryolarda ise en iyi sonuçlar önerdiğimiz yöntem ile elde edilmiştir. Ayrıca 60 adet hedeften oluşan senaryoda önerilen yöntem ile 4 adet İHA yerine 6 adet İHA görevlendirmenin daha düşük maliyetli bir atama

işlemi yapılacağını göstermiştir. Tavlama benzetimi algoritması ise hesaplanan toplam maliyete göre diğer yöntemlerin çok daha gerisinde kalmıştır. Sonuç olarak İHA ve hedef sayısının çok fazla sayıda olduğu karmaşık problemlerin çözümünde önerdiğimiz üç aşamalı hiyerarşik yöntem daha iyi sonuçlar verirken, daha küçük problemlerde genetik algoritma daha iyi sonuçlar vermiştir.

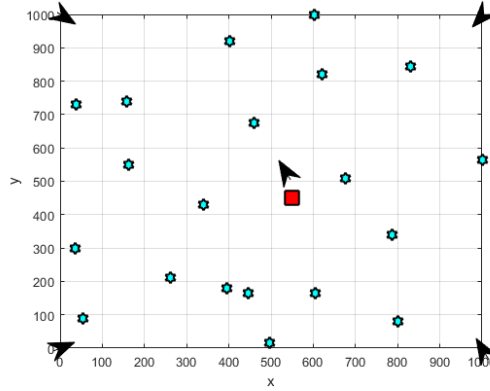
Çizelge 5.2. Farklı sayıda hedefler için hesaplanan ortalama süre verileri

Hedef	Süre (sn)		
	GA	TBA	Önerilen Yöntem
20	2.95	13.43	<b>2.35</b>
40	6.53	23.65	<b>3.24</b>
60	10.49	27.52	<b>4.70</b>

4 adet İHA'nın görevlendirildiği 20, 40 ve 60 adet hedeften oluşan çözüm uzaylarında hesaplanan ortalama sürelerin verileri Çizelge 5.2'de gösterilmiştir. Bu verilere göre önerilen yöntem tüm senaryolarda daha hızlı sonuçlar vermiştir. 20 adet hedeften oluşan senaryo için önerilen yöntem ve genetik algoritmadaki hesaplanan sürelerin birbirine oldukça yakın olduğu görülmüştür. Tavlama benzetimi algoritmasının diğer iki yöntemden daha yavaş çalıştığı tespit edilmiştir.

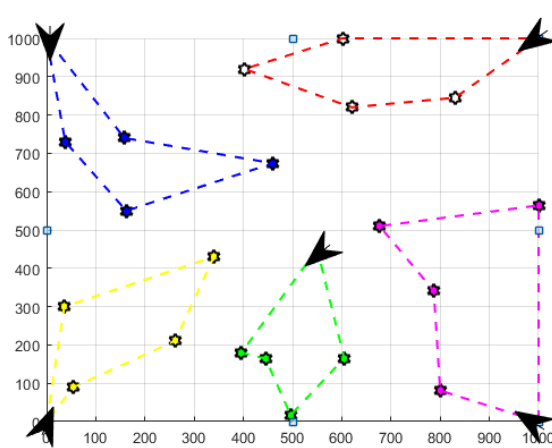
### 5.3. İlave Kısıtlar İçeren Farklı Bir SİHA Senaryosu

Burada SİHA'ların kısıtlı kapasite mühimmat taşıma durumu incelenmiştir. Gerçek hayattan esinlenilerek düşünüldüğü homojen özelliklere sahip olan SİHA'lar en fazla 4 mühimmat taşıyabilmektedir. Bu senaryoda herbir SİHA hedef noktalarını mühimmat kapasitesi dahilinde bombaladıktan sonra tekrar başlangıç noktalarına dönecekleri varsayılmıştır. Şekil 5.10'da görselleştirilen senaryoda kırmızı-kare yer istasyonunu, siyah-oklar SİHA'ları, turkuvaz-yıldızlar ise ataması yapılacak hedefleri ifade etmektedir.

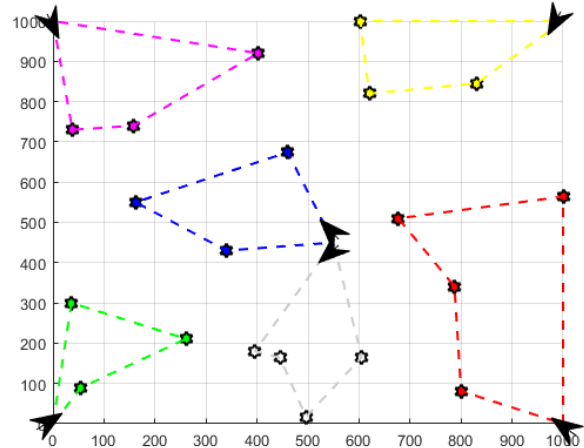


Şekil 5.10: SİHA Senaryosunun gösterimi.

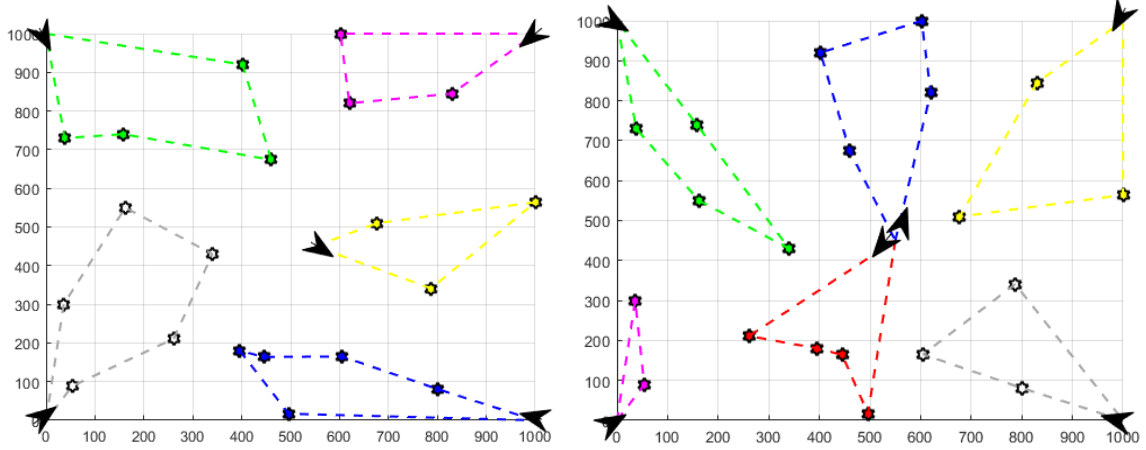
Bu senaryoda rotalama problemi, en fazla 4 hedefi imha edebilecek olan SİHA'ların en az 3 hedefi imha etmesi için bir kısıtlama getirilerek belirlenmiştir. Toplam 20 adet hedef noktası için 5 ve 6 adet SİHA görevlendirilerek simülasyon testleri yapılmıştır. Homojen özelliklere sahip olan SİHA'ların herbiri en fazla 4 hedef noktasını imha edebilecektir. Aynı senaryo yine aynı sayıda SİHA'lar kullanılarak mühimmat kısıtlamasının olmadığı durumla karşılaştırılmıştır. Atanan hedefleri bombaladıktan sonra tekrar harekete başladığı noktaya döndüğü varsayılan SİHA'ların test sonuçları şekil 5.11'de verilmiştir.



(a) Toplam Maliyet=6493



(b) Toplam Maliyet=6566



(c) Toplam Maliyet=6131

(d) Toplam Maliyet=6856

Şekil 5.11: Görev ataması yapılan SİHA'ların gösterimi. (a) ve (b) mühimmat kısıtlamasının olduğu, (c) ve (d) mühimmat kısıtlamasının olmadığı durumlar.

Hedef sayısının 20 olduğu ve 5 adet SİHA'nın görevlendirildiği senaryoda mühimmat kısıtlamasından dolayı her bir SİHA'ya eşit sayıda hedef atanmıştır. Görevlendirilen SİHA sayısı 6 olduğunda ise SİHA'lardan dört tanesine 3 adet, iki tanesine 4 adet olmak üzere farklı sayılarda hedef noktaları atanmıştır. Ayrıca 6 SİHA'nın görevlendirilmesi daha yüksek maliyetli olduğu görülmüştür. Aynı senaryo bu sefer mühimmat kısıtlaması koyulmadan aynı sayıda SİHA'lar ile test edilerek karşılaştırma yapılmıştır. 5 SİHA'nın görevlendirildiği durumda kısıtlamalı yöntemde hesaplanan maliyetin daha yüksek olduğu gözlenirken, 6 adet SİHA'nın kullanıldığı durumda kısıtlamanın olması daha iyi sonuçlar vererek kısıtlama olmayan yöntemle göre daha düşük maliyetli bir sonuç elde edilmiştir. Daha farklı hedef sayıları ve kısıtlar için görevlendirilecek en uygun SİHA sayısı değişiklik gösterebilir.

## 5. SONUÇ VE ÖNERİLER

Bu tezde sürü İHA'ların çok sayıda hedef noktasına uğrayacak biçimde uygun rotalama problemini çözmek için hiyerarşik görev atama yöntemi sunulmuştur. Makine öğrenmesi ve optimizasyon problemlerinin çözümünde oldukça etkili olan birkaç metodolojinin ardışıl olarak kullanıldığı bu yöntemde ilk önce  $k$ -ortalamalar kümeleme yöntemi kullanılarak hedefler İHA sayısına eşit olacak şekilde alt kümelere bölünmüştür. Daha sonra bu alt küme merkezlerinin İHA'lara olan uzaklıkları dikkate alınarak Macar algoritması ile İHA-alt küme ataması yapılmıştır. Son aşamada ise gezgin satıcı problemlerinden biri olan karınca kolonisi algoritması kullanılarak alt küme içindeki tüm hedeflerin o alt kümeye atanan İHA'ya optimal rota planlaması yapılmıştır. Farklı senaryolarda simüle edilen bu yöntemde İHA'lar için gözetleme, SİHA'lar için ise bombalama görevi tanımlandığı varsayılmış, bu yüzden de SİHA'ların kullanıldığı senaryoya ek olarak mühimmat kapasitesi kısıtlaması getirilmiştir. Bilgisayar ortamında yapılan testlerde aynı sabit hedef noktaları için farklı sayılarda İHA kullanılması durumunda maliyetler hesaplanmış ve genetik algoritma tabanlı metasezgisel yaklaşımla sonuçlarıyla karşılaştırılmıştır. Önerilen yöntemin, genel olarak daha fazla sayıda hedef noktası içeren karmaşık görev atama problemlerinin çözümünde daha başarılı sonuçlar verdiği gözlenmiştir.

## KAYNAKLAR DİZİNİ

- Alighanbari, M., How, J. P., 2008, A robust approach to the UAV task assignment problem. *International Journal of Robust and Nonlinear Control*, 18(2), 118 – 134.
- Alighanbari, M., How, J. P., 2005, Cooperative task assignment of unmanned aerial vehicles in adversarial environments, in *Proceedings American Control Conference, Portland*, vol. 7, 4661–4666.
- Alighanbari, M., Kuwata, Y., How, J. P., 2003, Coordination and control of multiple UAVs with timing constraints and loitering, *American Control Conference*, 5311–5316.
- Bertuocelli, L. F., Choi, H., Cho, P., vd., 2009, Real-time multi-UAV task assignment in dynamic and uncertain environments. *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 10 – 13.
- Bolla, M., 2013, *Spectral clustering and biclustering : learning large graphs and contingency tables*, Wiley, Chichester, West Sussex, United Kingdom.
- Casbeer, D. W., Holsapple, R. W., 2011, Column generation for a UAV assignment problem with precedence constraints. *International Journal of Robust and Nonlinear Control*, 21(12), 1421 – 1433.
- Cerro, J.d., Cruz Ulloa, C., Barrientos, A., de León Rivas, J., 2021, Unmanned Aerial Vehicles in Agriculture, *A Survey Agronomy*, 11, 203.
- Chen, X., Liu, Y., 2019, Cooperative task assignment for multi-UAV attack mobile targets, in: *2019 Chinese Automation Congress (CAC)*, IEEE 2151–2156. doi:10.1109/cac48633.2019.8996383.
- Choi, H., Brunet, L., How, J. P., 2009, Consensus-based decentralized auctions for robust task allocation. *IEEE Transactions on Robotics*, 25(4), 912 – 926.
- Coelho, B. N., Coelho, V. N., Coelho, I. M., Ochi, L. S., Haghazari, K. R., vd., 2017, A multi-objective green UAV routing problem, *Computers & Operations Research*, 306–315.
- Coffey, T., Montgomery, J. A., 2002, The emergence of mini uavs for military applications. *Defense Horizons*.
- Dantzig G. B., Ramser, J. H., 1959, The truck dispatching problem, *Management Science*, 80–91.

**KAYNAKLAR DİZİNİ (devam)**

- Edison, E., Shima, T., 2011, Integrated task assignment and path optimization for cooperating uninhabited aerial vehicles using genetic algorithms. *Computers & Operations Research*, 38 (1), 340-356.
- Forsmo, E. J., Grøtli, E. I., Fossen, T. I., Johansen T. A., 2013, Optimal search mission with unmanned aerial vehicles using mixed integer linear programming, *International Conference on Unmanned Aircraft Systems*, 253–259.
- Gao, W., 2020, New ant colony optimization algorithm for the traveling salesman problem, *International Journal of Computational Intelligence Systems* 13(1), 44.
- Gen M., Syarif, A., 2005, Hybrid genetic algorithm for multi-time period production/distribution planning, *Computers & Industrial Engineering*, 799–809.
- Ghazali, S. N. A. M., Anuar, H. A., Zakaria, S. N. A. S., Yusoff, Z., 2016, Determining position of target subjects in maritime search and rescue (MSAR) operations using rotary wing unmanned aerial vehicles (UAVs), in *Proceedings International Conference Information Communication Technology Convergence*, 1–4.
- Glade, D., 2000, *Unmanned aerial vehicles: Implications for military operations*, Technical report, air university.
- Hussein, A., Mostafa, H., Badrel-Din, B., Sultan, O., Khamis, A., 2012, Metaheuristic optimization approach to mobile robot path planning, in *Proceedings International Conference Engineering Technologies*, 1–6.
- Kirkpatrick, S., Gelatt, C. D., Veechi, M. P., 1983, Optimization by simulated annealing, *Science*, 220, 671-679.
- Laporte, G., Nobert, Y., Arpin, D., 1986, An exact algorithm for solving a capacitated location-routing problem. *Annals of Operations Research* 6, 293–310.
- Lee, K.S, Ovinis, M., Nagarajan, T., Seulin, R., Morel, O., 2015, Autonomous patrol and surveillance system using unmanned aerial vehicles, in *Proceedings IEEE 15th International Conference on Environment and Electrical Engineering*, 1291–1297.
- Lee, S., Morrison, J. R., 2015, Decision support scheduling for maritime search and rescue planning with a system of uavs and fuel service stations, in *Proceedings of the IEEE*.
- Liu, W., Zheng, Z., Cai, K.Y., 2013, Bi-level programming based real-time path planning for unmanned aerial vehicles, *Knowledge Management System*, 44, 34–47.
- Marini, F., Walczak, B., 2015, *Particle Swarm Optimization (PSO). A Tutorial*, Chemometrics and Intelligent Laboratory Systems.

### KAYNAKLAR DİZİNİ (devam)

- Otto, A., Agatz, N., Campbell, J., Golden, B., Pesch, E., 2018, Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: a survey, *Networks* 72 (4),411–458.
- Qiu, T., Liu, X., Han, M., Li, M., Zhang, Y., 2017, A self-recoverable time synchronization for sensor networks of healthcare IoT, *Computer Networks*, 481–492.
- Rasmussen S. J., Shima, T., 2008, Tree search algorithm for assigning cooperating UAVs to multiple tasks, *International Journal of Robust and Nonlinear Control*, 18(2), 135–153.
- Ries, J., Ishizaka, A., 2012, A multi-criteria support system for dynamic aerial vehicle routing problems, *Christian Camp and Conference Association*. 1–4.
- Schumacher, C., Chandler, P., Pachter, M., vd., 2007, Optimization of air vehicles operations using mixed-integer linear programming. *Journal of the Operational Research Society*, 58(4), 516 – 527.
- Schwarzrock, J., Zacarias, I., Bazzan, A. L., Araujo Fernandes, R. Q., Moreira, L. H., vd., 2018, Solving task allocation problem in multi 540 unmanned aerial vehicles systems using swarm intelligence, *Engineering Applications of Artificial Intelligence*.
- Shetty, V. K., Sudit, M., Nagi. R., 2008, Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles. *Computers & Operations Research*, 35(6), 1813 – 1828.
- Shima, T., Rasmussen, S. J., Sparks, A. G., vd., 2006, Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms. *Computers & Operations Research*, 33(11), 3252 – 3269.
- Sousa, J., Simsek, T., Varaiya, P., 2004, Task planning and execution for UAV teams. *Proceedings of the IEEE Conference on Decision and Control*, 3804 – 3810.
- Tang, B.W., Zhu, Z.X., Luo, J.J., 2016, Hybridizing Particle Swarm Optimization and Differential Evolution for the Mobile Robot Global Path Planning, *International Journal of Advanced Robotic Systems*, 13, 1–17.
- Thakoor, O., Garg, J., Nagi, R., 2020, Multiagent UAV routing: A game theory analysis with tight price of anarchy bounds,” *IEEE Transactions on Automation Science and Engineering*, 100–116.
- Tianhan, T., Kun, W., Qishuang, W., Bo, F., Gang, W., vd., 2019, UAV cooperative attack and route planning based on DPSO algorithm, *Journal of Physics Conference Series* 742-6596.

**KAYNAKLAR DİZİNİ (devam)**

- Whitley, D., 1994, A genetic algorithm tutorial, *Statistics and Computing*, 4, 65–85.
- Yadav V., Narasimhamurthy, A., 2017, A heuristics based approach for optimizing delivery schedule of an unmanned aerial vehicle (drone) based delivery system, *International Conference on Pattern Recognition (ICAPR)*, 1–6.
- Zaza, T., Richards, A., 2014, Ant colony optimization for routing and tasking problems for teams of UAVs, *International Conference on Control*. 652–655.
- Zhao, T., Pan, X., He, Q., 2017, Application of dynamic ant colony algorithm in route planning for UAV, *International Science and Technology Conference*, 433–437.
- Zhao, W., Meng, Q., Chung, P. W. H., 2016, A heuristic distributed task allocation method for multivehicle multitask problems and its application to search and rescue scenario, *IEEE Transactions on Cybernetics*, 902–915.
- Zhong, L., Luo, Q., Wen, D., Dong, S., Mai Shi, J., vd., 2013, A task assignment algorithm for multiple aerial vehicles to attack targets with dynamic values, *IEEE Transactions on Intelligent Transportation Systems*.
- Zhou, Y., Rui, T. Li, Y., Zuo, X., 2019, A UAV patrol system using panoramic stitching and object detection, *Computers & Electrical Engineering*.