

Kısıt Bilinçli Genetik Algoritma ve A* Yöntemi ile Drone Filosu Optimizasyonu

Düzgün Murat Ertik
Bilişim Sistemleri Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
221307065@kocaeli.edu.tr

Mehmet Emir Salihoğlu
Bilişim Sistemleri Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
221307004@kocaeli.edu.tr

Yusuf Can Demirkol
Bilişim Sistemleri Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
221307061@kocaeli.edu.tr

Özet—This study presents a constraint-aware optimization system for autonomous drone fleets performing last-mile deliveries. The proposed architecture integrates a Genetic Algorithm (GA), A* pathfinding, and Constraint Satisfaction Programming (CSP) to assign delivery tasks to drones while satisfying critical constraints such as limited battery capacity, payload weight limits, active no-fly zones, and delivery time windows.

Each delivery is modeled with spatial and temporal attributes, while drones are defined with operational parameters such as speed, position, battery, and weight capacity. No-fly zones are represented as polygons with defined active time intervals, and any route intersecting an active zone is invalidated. The GA uses a priority-based fitness function to optimize assignments, and A* computes the most cost-efficient routes between drones and delivery locations.

The implementation is developed in Python using modular classes and verified through synthetic test data. Results show that the hybrid model efficiently generates feasible and low-cost delivery plans, making it suitable for real-time applications in smart logistics systems.

Index Terms—Genetic Algorithm, Drone Delivery, Path Planning, Constraint Satisfaction, A* Search, No-Fly Zones, Autonomous Systems

I. GİRİŞ

Son yıllarda insansız hava araçları (İHA'lar), özellikle son kilometre teslimatlarında hızlı ve esnek çözümler sunmalarıyla lojistik sektöründe önemli bir yer edinmiştir. Ancak bu araçların sınırlı batarya kapasiteleri, taşıma ağırlığı kısıtları, zaman pencereleri ve belirli coğrafi alanlarda uçuş yasağı gibi birçok kısıtla karşı karşıya kalmaları, görev atamasının ve rota planlamasının optimize edilmesini zorunlu kılmaktadır.

Bu çalışmada, çoklu drone filosuna sahip bir teslimat sisteminde görev atamalarını optimize etmek amacıyla kısıt bilinçli bir algoritma mimarisi önerilmiştir. Önerilen yapı, genetik algoritma (GA), A* arama algoritması ve kısıt tabanlı programlama (CSP) yaklaşımlarını bütünleştirerek her bir drone'un en uygun teslimatı seçmesini ve geçerli bir rota üzerinden ulaştırmasını sağlamaktadır.

Teslimatlar; ağırlık, öncelik derecesi, zaman aralığı ve konum bilgilerini içerecek şekilde modellenmiş; İHA'lar ise başlangıç konumu, hız, batarya kapasitesi ve taşıma sınırı gibi operasyonel özelliklerle tanımlanmıştır. Ayrıca sistemde uçuşa yasak bölgeler çokgen yapılar halinde modellenmiş ve yalnızca belirli zaman aralıklarında etkin hale gelecek şekilde yapılandırılmıştır.

II. İLGİLİ ÇALIŞMALAR

Drone tabanlı teslimat sistemleri son yıllarda hem akademik dünyada hem de endüstride yoğun ilgi görmektedir. Bu alandaki çalışmalarda genellikle rota planlama, görev ataması ve enerji optimizasyonu gibi alt problemler ele alınmaktadır.

Genetik algoritmalar (GA), bu tür kombinatorial problemlerde çözüme yakın sonuçları hızlıca bulabilmeleri nedeniyle sıklıkla tercih edilmektedir. Literatürde GA'nın drone görev atamalarında başarılı sonuçlar verdiği birçok çalışma bulunmaktadır. Öte yandan, yol bulma ve engellerden kaçınma gibi alt problemler için A* algoritması geniş bir şekilde kullanılmakta olup, özellikle sabit harita üzerinde en kısa yolun hesaplanmasında etkilidir.

Kısıt tabanlı programlama (CSP) ise sistemdeki fiziksel ve operasyonel kısıtların (örneğin taşıma kapasitesi, zaman penceresi veya uçuşa yasak bölgeler) kontrolü için tercih edilen deterministik bir yaklaşımdır.

Bu çalışmada, yukarıda adı geçen üç yöntem bir araya getirilerek hem geçerli hem de optimize görev atamaları yapılması hedeflenmiştir. Literatürde bu üç yaklaşımın bütünleşik olarak kullanıldığı çalışmalar sınırlı sayıda olup, önerilen yöntem bu açıdan yenilikçi bir yapıya sahiptir.

III. SİSTEM TASARIMI VE YÖNTEM

Bu bölümde geliştirilen drone teslimat optimizasyon sisteminin mimarisi ve kullanılan algoritmalar detaylandırılmaktadır. Sistem, görev ataması ve rota planlamasını gerçekleştirmek için üç temel algoritmayı entegre eder: Genetik Algoritma (GA), A* arama algoritması ve Kısıt Tabanlı Programlama (CSP). Uygulama, Python programlama diliyle nesne tabanlı olarak yapılandırılmış ve modüler bir biçimde geliştirilmiştir.

A. Genel Mimari

Sistem, veri girişlerini JSON formatında okuyarak başlar. Her bir *drone*, *teslimat* ve *uçuşa yasak bölge* nesnesi ilgili Python sınıfları ile temsil edilir. Tüm noktalar (drone başlangıçları ve teslimat konumları) bir graf üzerinde düğüm olarak modellenir ve aralarındaki bağlantılar Öklidyen uzaklığa dayalı kenarlarla tanımlanır.

Genetik algoritma, her bir jenerasyonda farklı teslimat-dron kombinasyonlarını değerlendirir. Her çözümün uygunluğu (*fitness*) hesaplanırken, CSP bileşeni teslimatın geçerliliğini kont-

rol eder; A^* ise rota üzerindeki mesafeyi ve olası uçuşa yasak bölge çakışmalarını hesaplar. Böylece yalnızca geçerli ve maliyet açısından verimli eşlemeler değerlendirilir.

Aşağıdaki alt bölümlerde bu bileşenlerin her biri detaylı şekilde açıklanacaktır.

B. Kısıt Tabanlı Programlama (CSP)

Kısıt Tabanlı Programlama (CSP), her bir teslimatın yalnızca geçerli koşulları sağlayan drone'lara atanabilmesini garanti altına almak amacıyla kullanılmıştır. Bu yapıda aşağıdaki kısıtlar dikkate alınmaktadır:

- **Taşıma Kapasitesi:** Teslimatın ağırlığı, drone'un maksimum taşıma kapasitesinden fazla olamaz.
- **Uçuşa Yasak Bölgeler:** Drone ile teslimat noktası arasında çizilen rota, aktif olan bir uçuşa yasak bölge ile kesişmemelidir.
- **Zaman Penceresi:** Teslimatın yapılacağı zaman, teslimat nesnesine tanımlı zaman aralığı içerisinde olmalıdır.
- **Batarya Kapasitesi:** Drone'un kalan bataryası, uçuş boyunca harcanacak enerjiyi karşılayabilecek seviyede olmalıdır. Enerji tüketimi, mesafeye bağlı olarak mAh/metre biriminde hesaplanır.

Her bir eşleşme adımında CSP denetleyicisi, drone ile teslimat arasında geçerli bir ilişki kurulup kurulamayacağını belirler. Geçerli olmayan eşleşmeler, genetik algoritma tarafından değerlendirme dışı bırakılır. Böylece arama alanı anlamlı çözümlerle sınırlandırılmış olur.

C. A^* ile Rota Hesaplama

A^* arama algoritması, her bir drone ile potansiyel teslimat noktası arasında en kısa geçerli rotayı bulmak için kullanılmıştır. Algoritma, başlangıç ve hedef noktalar arasındaki tahmini maliyeti $f(n) = g(n) + h(n)$ Burada:

- $g(n)$: Başlangıçtan mevcut düğüme kadar olan gerçek maliyet (mesafe),
- $h(n)$: Mevcut düğümden hedefe olan öklidyen uzaklık (heuristic),

şeklinde tanımlanır. Tüm konumlar (drone başlangıçları ve teslimat lokasyonları) bir graf üzerinde düğüm olarak temsil edilmiştir. Kenarlar, düğümler arasındaki öklidyen uzaklıklarla ağırlıklandırılmıştır.

A^* algoritması, bu graf üzerinde en düşük maliyetli yolu bulur. Ancak algoritma çalıştırılmadan önce yolun uçuşa yasak bölgelerle çakışıp çakışmadığı CSP bileşeni tarafından kontrol edilir. Eğer rota aktif bir yasak bölge ile kesişiyorsa, söz konusu eşleşme geçersiz sayılır ve algoritma o yöne doğru dallanmaz.

A^* algoritmasının bu projede kullanımı, rota maliyetinin doğrudan enerji tüketimi hesabına da katkı sağlamaktadır. Böylece hem geçerli hem de kısa yollar tercih edilmekte, sistemin toplam enerji verimliliği artırılmaktadır. "

D. Genetik Algoritma ile Görev Ataması

Genetik algoritma (GA), drone'lar ile teslimat noktaları arasındaki uygun eşlemeleri bulmak amacıyla kullanılmaktadır. Bu algoritma, doğadan ilham alan evrimsel bir süreçte

çalışarak çözüm popülasyonları üzerinde seçim, çaprazlama (crossover) ve mutasyon işlemleri uygular.

Başlangıç Popülasyonu: İlk popülasyon, geçerli teslimat-drone eşleşmelerinden rastgele oluşturulur. Her çözüm bireyi, bir drone filosuna ait atamaları temsil eden bir dizi eşleşmeden oluşur.

Uygunluk Fonksiyonu (Fitness): Her bireyin başarımı, aşağıdaki ölçütlere göre değerlendirilir:

- **Toplam Öncelik Skoru:** Her teslimatın öncelik değeri, başarıya katkı sağlar.
- **Enerji Tüketimi:** Drone'un taşıdığı mesafe boyunca harcadığı enerji mAh cinsinden hesaplanır...

Uygunluk, genel formül olarak aşağıdaki şekilde hesaplanır:

$$Fitness = \sum (P_i \times w) - (E_i \times \alpha)$$

Burada w öncelik ağırlığı, α ise ceza katsayısını temsil eder.

Çaprazlama ve Mutasyon: Popülasyondaki en iyi çözümlerden rastgele ikili kombinasyonlar seçilir ve bunların bazı kısımları birleştirilerek yeni bireyler üretilir. Mutasyon işlemi ile belirli oranla yeni eşleşmeler denenerek çeşitlilik korunur.

Elitizm: Her jenerasyonda en başarılı birey doğrudan bir sonraki popülasyona aktarılır. Bu sayede iyi çözümlerin kaybolması engellenir.

Genetik algoritma, belirli bir jenerasyon sayısı boyunca çalıştırılır ve sonunda en yüksek uygunluk değerine sahip birey, sistemin çıktısı olarak seçilir.

IV. DENEYSEL KURULUM

Geliştirilen sistem Python 3.10 programlama dili ile yazılmıştır. Uygulama, nesne tabanlı ve modüler bir yapıdadır. Kullanılan veriler JSON formatında tutulmakta ve program başlangıcında yüklenmektedir. Girdiler, `data/` klasöründe drone, teslimat ve uçuşa yasak bölgeler olmak üzere üç ayrı dosyada saklanmaktadır.

A. Kullanılan Teknolojiler

Uygulama, aşağıdaki açık kaynak Python kütüphaneleri ile desteklenmiştir:

- **matplotlib:** Drone rotalarının 2B olarak görselleştirilmesinde kullanılmıştır.
- **shapely:** Teslimat yollarının uçuşa yasak bölgelerle geometrik kesişimini denetlemek için kullanılmıştır.
- **heapq:** A^* algoritmasındaki öncelikli kuyruk yapısı için kullanılmıştır.
- **datetime, copy, random:** Zaman penceresi kontrolü, veri kopyalama ve popülasyon çeşitliliği için kullanılmış yardımcı Python modülleridir.

B. Test Senaryosu

Test senaryosu, 5 adet drone, 20 adet teslimat ve 3 adet uçuşa yasak bölge ile yapılandırılmıştır. Her bir teslimat, ağırlık ve zaman aralığı gibi kısıtlarla birlikte tanımlanmıştır. Uçuşa yasak bölgeler çokgen olarak tanımlanmakta ve yalnızca belirli saat aralıklarında aktif olmaktadır.

Sistem başlatıldığında aşağıdaki adımları takip eder:

- 1) Tüm veriler dosyalardan yüklenir.

- 2) Konumlara göre düğümler ve kenarlar ile bir grafik oluşturulur.
- 3) Genetik algoritma başlatılarak belirli sayıda jenerasyon boyunca çözüm popülasyonları değerlendirilir.
- 4) A* algoritması her eşleşme için rota maliyetini hesaplar.
- 5) En uygun eşleşme harita üzerinde çizilir ve kullanıcıya sunulur.

Tüm bu işlemler milisaniye düzeyinde gerçekleşmekte ve sistem çıktısı olarak en uygun eşleşme çiftleri terminal üzerinden yazdırılmakta, aynı zamanda görsel olarak da gösterilmektedir.

V. SONUÇLAR VE TARTIŞMA

Geliştirilen sistem, tanımlı bir test senaryosu üzerinde başarıyla çalıştırılmış ve elde edilen çıktılar doğrultusunda değerlendirilmiştir. Genetik algoritma, her jenerasyonda yalnızca geçerli eşleşmeleri dikkate almış; A* algoritması rota maliyetlerini hesaplamış; CSP modülü ise batarya, yük, zaman ve uçuşa yasak bölge kısıtlarını denetlemiştir.

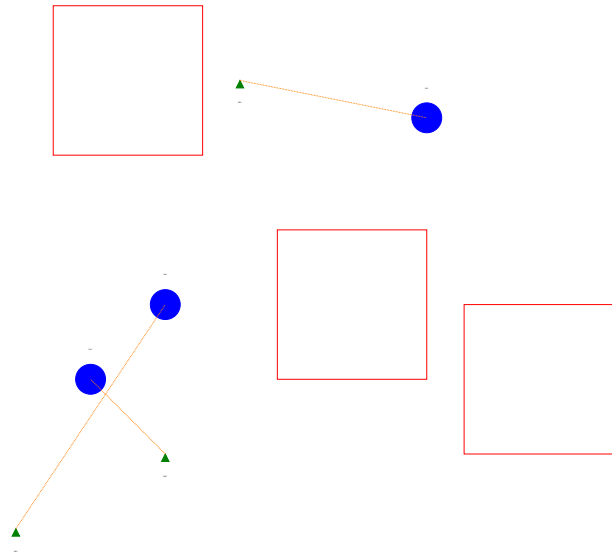
Test senaryosunda başlangıç zamanı sabit olarak 00:40 alınmış, sistem 5 drone, 20 teslimat ve 3 adet uçuşa yasak bölge ile çalıştırılmıştır. Genetik algoritma 30 jenerasyon boyunca çözüm üretmiş ve en iyi eşleşmeler aşağıdaki şekilde elde edilmiştir:

- Drone #1 → Teslimat #16
- Drone #2 → Teslimat #2
- Drone #5 → Teslimat #15

Sistem çıktısında her eşleşme için mesafe, hız ve uçuş süresi gibi bilgiler aşağıda örneklenmiştir:

Drone#1 → Delivery#16 | Distance:
30.00 m | Speed: 8.0 m/s | Time:
3.75 sec

Toplamda genetik algoritmanın çalışma süresi yaklaşık 0.27 saniye olarak ölçülmüş ve bu değer, sistemin gerçek zamanlı uygulamalara uygun olduğunu göstermektedir.



Şekil 1. Drone Teslimat Eşleşmeleri ve Uçuşa Yasak Bölgeler (Senaryo 1)

A. Uçuşa Yasak Bölgelerin Etkisi

Uygulama sırasında aktif olan yasak bölgeler, A* algoritması tarafından rota oluşturulmadan önce CSP modülü ile kontrol edilmiştir. Etkin yasak bölge ile çakışan rotalar geçersiz sayılmış ve algoritma bu eşleşmeleri değerlendirme dışı bırakmıştır. Bu özellik, güvenli uçuş planlaması açısından sistemin önemli bir yetkinliğidir.

VI. SONUÇ VE GELECEK ÇALIŞMALAR

Bu çalışmada, çoklu drone filosu ile teslimat görevlerinin optimize edilmesini amaçlayan bütünlük bir algoritma tasarımı sunulmuştur. Genetik algoritma, A* arama algoritması ve kısıt tabanlı programlama yaklaşımları bir araya getirilerek; batarya sınırı, taşıma kapasitesi, zaman kısıtı ve uçuşa yasak bölgeler gibi gerçek dünya kısıtları altında geçerli ve verimli teslimat atamaları gerçekleştirilmiştir.

Uygulama Python programlama dili ile modüler bir yapıda geliştirilmiş ve test senaryosu üzerinde başarıyla çalıştırılmıştır. Sonuçlar, sistemin milisaniye düzeyinde çalışabildiğini ve oldukça düşük enerji maliyetiyle yüksek öncelikli teslimatları gerçekleştirebildiğini göstermektedir.

Gelecek çalışmalarda aşağıdaki geliştirmelerin yapılması planlanmaktadır:

- Drone'ların batarya tüketimini uçuş süresi, yük ağırlığı ve hız parametreleriyle dinamik olarak modellemek,
- Birden fazla teslimatın aynı drone'a atanabildiği çoklu taşıma senaryolarını desteklemek,
- Gerçek zamanlı trafik veya hava durumu verilerini algoritmaya entegre etmek,
- Henüz tamamlanmamış olan Senaryo 2'yi daha büyük ve yoğun veri kümeleri üzerinde test etmek.

Bu geliştirmeler ile sistemin gerçek dünya koşullarına daha uyumlu ve esnek hale getirilmesi hedeflenmektedir.

KAYNAKLAR

- [1] D. M. Ertik, M. E. Salihoğlu, and Y. C. Demirkol, "Kısıt Bilinçli Genetik Algoritma ve A* Yöntemi ile Drone Filosu Optimizasyonu," Bitirme Projesi, Bilişim Sistemleri Mühendisliği, Kocaeli Üniversitesi, 2025.
- [2] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, 1997, pp. 4104–4108.
- [3] A. M. Pinto, M. F. Santos, and E. P. de Freitas, "Genetic algorithm applied to UAV route planning with battery constraints and no-fly zones," in *Proc. 2021 Int. Conf. on Unmanned Aircraft Systems (ICUAS)*, pp. 1310–1319, 2021.
- [4] S. Koenig and M. Likhachev, "D* Lite," in *Proc. of the National Conference on Artificial Intelligence (AAAI)*, pp. 476–483, 2002.