

Floyd-Warshall ve Karınca Kolonisi Optimizasyonu Algoritmaları ile Depo Rota Planlaması

Furkan Ulu¹ , Gülveren Tabansız Göç² , Fatih Çavdur¹ 

ÖZET

Amaç: Bu çalışmada, depo rota planlama problemleri için tam ve bütünleşik bir çözüm yaklaşımı önerilmektedir.

Yöntem: Önerilen çözüm yaklaşımına göre, öncelikle depoda ziyaret edilmesi gereken noktaların bazılarını temsil eden bir ağ gösterimi oluşturulmaktadır. Daha sonra, her turda bir gezgin satıcı problemi oluşturmak ve söz konusu ağ yapısının ilgili düğümler arasındaki en kısa yolların belirlenmesi için Floyd-Warshall algoritması kullanılmaktadır. Son olarak, aracın toplam seyahat mesafesini minimize ederek ilgili gezgin satıcı probleminin çözümünü sağlamak için karınca kolonisi optimizasyonu algoritması kullanılmaktadır.

Bulgular: Önerilen çözüm yaklaşımının performansını göstermek için çeşitli veri kümeleri ile hesaplamalı deneyler gerçekleştirilmiştir. Sonuçların oldukça tatmin edici olduğu ve önerilen çözüm metodolojisinin depo rota planlamasında önemli faydalar sağlayabileceği gözlemlenmiştir.

Özgünlük: Çalışmanın en önemli katkısını, depo rota planlama problemi için Floyd-Warshall algoritması ile karınca kolonisi optimizasyonu algoritmasını bir arada kullanan esnek bir çözüm yaklaşımı çerçevesi sunulması oluşturmaktadır.

Anahtar Kelimeler: En Kısa Yol Problemi, Gezgin Satıcı Problemi (GSP), Floyd-Warshall Algoritması, Karınca Kolonisi Optimizasyonu (KKO) Algoritması, Rota Planlama.

JEL Kodları: C61, R41, C63.

Warehouse Route Planning via Floyd-Warshall and Ant Colony Optimization Algorithms

ABSTRACT

Purpose: In this study, we propose a complete and integrated solution approach for warehouse route planning problems.

Methodology: According to the proposed solution approach, a graph representation of the warehouse is first constructed where some vertices correspond to points to be visited. The Floyd-Warshall algorithm is employed to calculate the shortest paths between the corresponding points of the graph to formulate a traveling salesman problem for each tour. Finally, the ant colony optimization algorithm is utilized to solve the corresponding traveling salesman problem by minimizing the total travel distance.

Findings: Computational experiments are conducted with different datasets to demonstrate the performance of the proposed solution approach. It is observed that the results are quite satisfactory and the proposed solution methodology might provide significant benefits in warehouse route planning.

Originality: The main important contribution of our study is the proposal of a flexible solution framework that combines the Floyd-Warshall algorithm and the ant colony optimization algorithm for the general route planning problem.

Keywords: Shortest Path Problem, Traveling Salesman Problem (TSP), Floyd-Warshall Algorithm, Ant Colony Optimization (ACO) Algorithm, Route Planning.

JEL Codes: C61, R41, C63.

¹ Bursa Uludağ Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü, Bursa, Türkiye

² Mudanya Üniversitesi, Mühendislik, Mimarlık ve Tasarım Fakültesi, Endüstri Mühendisliği Bölümü, Bursa, Türkiye

Corresponding Author: Gülveren Tabansız Göç, gulveren.tabansizgoc@mudanya.edu.tr

DOI: 10.51551/verimlilik.1539618

Research Article | Submitted: 27.08.2024 | Accepted: 10.02.2025

Atıf/Cite: Ulu, F. Tabansız Göç, G. ve Çavdur, F. (2025). "Floyd-Warshall ve Karınca Kolonisi Optimizasyonu Algoritmaları ile Depo Rota Planlaması", *Verimlilik Dergisi*, 59(2), 337-354.

EXTENDED ABSTRACT

Warehouses are fundamental components of a supply chain, and they must be effectively managed in line with changing market conditions and increasing demand. Warehouse management, which encompasses receiving, storage, internal transport, order picking, and shipping processes, aims to meet customer demands on time while minimizing costs. Order picking is a critical warehouse operation which might involve gathering, packaging, and delivering items considering customer demands. Since order picking constitutes a significant portion of total warehouse costs, it plays a vital role in reducing operational expenses and optimizing resource usage. A large part of the time spent by order pickers in the warehouse is dedicated to travel, which makes it important to consider the shortest distance-routes. However, traditional routing methods may fall short in providing optimal solutions in complex warehouse structures and may cause delays in order picking, leading to increased logistics costs and decreased operational efficiency. Therefore, warehouse routing processes need to be supported by more efficient algorithms. Advanced optimization techniques plan order picking routes by minimizing the total distance, and hence, saving time and reducing labor costs.

In this study, a complete and integrated solution approach for warehouse route planning problems is developed. The study assumes that there is only one vehicle in the warehouse, starting at a designated point (the depot), collecting items from storage locations, and returning to the starting point at end of its tour, which is modeled as the traveling salesman problem (TSP). According to the proposed solution approach, an initial network representation of the locations to be visited in a particular tour is constructed. We then determine the shortest paths for all nodes in the network using the Floyd-Warshall algorithm, and formulate the corresponding TSP. Finally, the ant colony optimization (ACO) algorithm is used to minimize the total travel distance and solve the TSP.

The main contribution of our study is the proposal of a flexible solution framework that combines the Floyd-Warshall algorithm and the ACO algorithm for rectangular-shaped warehouse route planning problems to provide an efficient and integrated solution adaptable to different scenarios. Additionally, various experiments are conducted with different data sets to demonstrate the performance of the proposed approach.

The solution performance is analyzed based on the changes in terms of the number of pickup points and some of the parameters of the ACO algorithm. It is observed that as the number of pickup points increased, both the route length and the solution time increase in general. An increase in the number of ants is associated with a longer solution time while an increase in the evaporation coefficient resulted in longer route lengths. When the pheromone intensity increases, the average route length slightly decreases, although no significant effect is observed on the solution time. The solution times, however, might be considered as reasonable.

There are some additional concepts to consider in future studies. One of these is about the assumption of having only one vehicle in the system at any given time, which might be restrictive in real-life warehouse environments. In such cases, the methodology would need to be updated to accommodate a multi-vehicle setup. Future research could also explore the use of alternative solution approaches instead of the Floyd-Warshall and ACO algorithms.

1. GİRİŞ

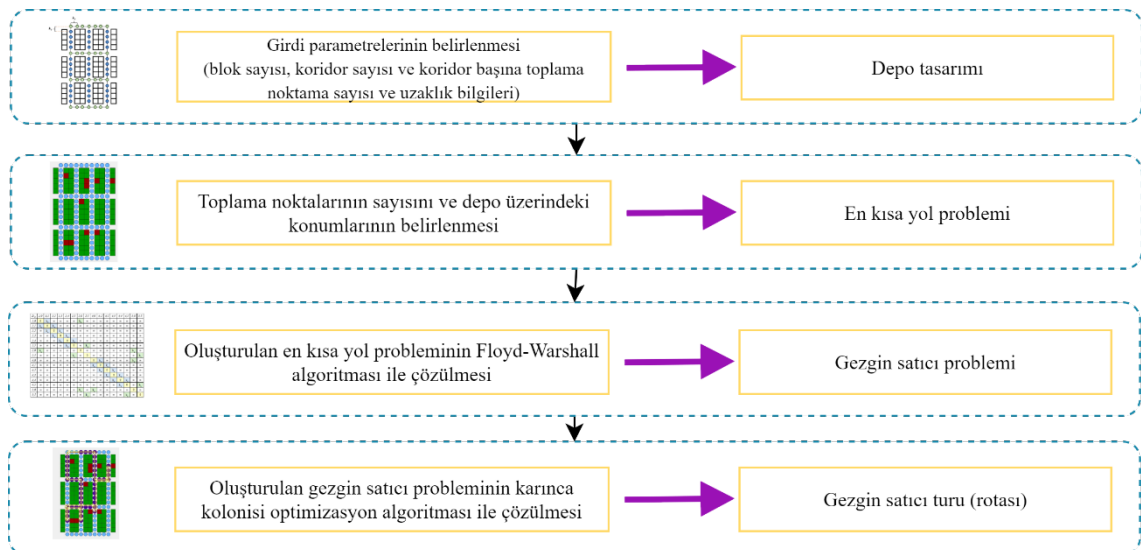
Sipariş toplama, depolardaki stokların müşteri taleplerine uygun şekilde toplanması, paketlenmesi ve doğru zamanda teslim edilmesini içeren kritik bir süreçtir. Bu operasyon, müşteri memnuniyeti için büyük önem taşımaktadır ve etkin yürütüldüğünde ürünlerin hızlı ve doğru bir şekilde teslim edilmesini sağlamaktadır. Sipariş toplama operasyonunda, personelin toplama noktalarını dikkate alarak optimize edilmiş bir rota oluşturması önemlidir. Rotalama, müşteri siparişlerine göre malzemelerin bulunduğu bölgelerin doğru bir planlama ile taranmasını sağlamaktadır. Optimize edilmiş bir rota ile depodaki malzemelerin daha az zaman ve emek harcanarak etkin bir şekilde toplanması hedeflenmektedir. Bu rota, malzemelerin düzenli stoklanması ve toplayıcıların doğru yönlendirilmesiyle belirlenmektedir. Doğru rotalama, sipariş toplama sürecini hızlandırarak depolama maliyetlerini düşürmektedir ve operasyonel verimliliği artırmaktadır.

Depo operasyonlarında, diğer depo operasyonlarına kıyasla en yüksek maliyetli operasyon sipariş toplama olarak kabul edilmektedir (Park, 2012: 3; Zhang ve diğerleri, 2019). Sipariş toplama işlemlerinin toplam depo işletme maliyetinin %55'ten fazlasını oluşturabileceği iddia edilmektedir (Bottani ve diğerleri, 2015: 645) ve bu nedenle lojistikte belirtilen yüksek düzeyde verimliliği elde etmek için bu işlemlere odaklanmak gerekmektedir. Sipariş toplama süreci, seçilen ürünlerin sevkiyatına kadar olan tüm işlemleri içermektedir ve bu nedenle seçilen ürünlerin bir sıralama/birleştirme süreci ile taşınması gibi işlemler de bu süreçte dahil edilmektedir (Rouwenhorst ve diğerleri, 2000). Ancak bu çalışmanın odak noktası, bir dizi öğenin seçilmesini ve seçilen öğelerin depolama yerlerinden alınması kısmını oluşturmaktadır. Problemde, depoda sadece tek bir araç olduğu varsayılmakta ve araç belirli bir başlangıç noktasından (depodan) başlayarak her turda bazı önceden belirlenmiş konumlardan ürünleri alıp aynı başlangıç noktasına geri dönmektedir.

Bu çalışmanın en önemli katkısı, Floyd-Warshall ve karınca kolonisi optimizasyonu (KKO) algoritmaları kullanılarak depo rota planlama problemi için bütünlük bir çözüm metodolojisi geliştirilmesidir. Ayrıca, geliştirilen metodolojinin çözüm performansını göstermek amacıyla çeşitli veri kümeleriyle hesaplamalı deneyler gerçekleştirilmiştir.

Çalışma kapsamında, önerilen çözüm yaklaşımının akış şeması Şekil 1'de gösterilmiştir. Şekil 1 incelendiğinde, öncelikle depo tasarımı belirlemek amacıyla gerekli girdi parametreleri (blok sayısı, koridor sayısı, her koridor başına toplama noktası sayısı ve uzaklık bilgileri) tanımlanır. İkinci adımda, toplama noktalarının sayısı ve depo içindeki konumları belirlenerek, en kısa yol problemi oluşturulur. Üçüncü adımda, bu en kısa yol problemi Floyd-Warshall algoritması ile çözülür ve her bir tur için gezgin satıcı problemi elde edilir. Son olarak, dördüncü adımda, gezgin satıcı problemi (GSP), KKO algoritması ile çözülerek, gezgin satıcının izlediği en verimli rota (gezgin satıcı turu) belirlenir.

Makalenin düzeni şu şekildedir. Bir sonraki bölümde, kısa bir literatür taraması sunulmaktadır. İzleyen bölüm, kullanılan çözüm yaklaşımının ayrıntılarını içermekte; ilgili alt bölümlerde Floyd-Warshall algoritmasıyla en kısa yol hesaplamaları ve KKO algoritmasıyla rotalama konuları bulunmaktadır. Daha sonra, önerilen yaklaşım için geliştirilen bir sayısal örnek ile metodolojinin gösterimi sunulmuş olup, sonraki bölümde çözüm performansı incelenmiştir. Son bölüm, çalışma hakkındaki genel yorumları içermektedir.



Şekil 1. Önerilen çözüm yaklaşımının çerçevesi

2. LİTERATÜR TARAMASI

Depolar, tedarik zincirlerinin temel unsurlarındandır ve piyasa dalgalanmaları, ürün çeşitliliği artışı ve hızlanan tedarik talepleri gibi faktörler, bu sistemlerin işlevlerini etkilemektedir (Rushton ve diğerleri, 2017: 255: 286). Depoların, tedarik zincirinin özel ihtiyaçlarına uygun şekilde planlanması ve yönetilmesi gerekmektedir. Depolar, tesisin kurulumu, işletilmesi, personel ve ekipman kullanımı gibi nedenlerle tedarik zincirinin maliyetli bileşenlerinden biridir ve bu yüzden depo tasarımı ve yönetimi, ürünlerin rekabet gücünü etkileyen kritik bir faktördür (Gu ve diğerleri, 2007).

Firmalar depo yönetiminde iki temel faktörü dikkate almaktadır: birincisi, müşteri taleplerini doğru ve zamanında karşılamak, ikincisi ise bu süreci en düşük maliyetle sağlamaktır. Depolarda teslim alma, depolama, iç taşıma, toplama ve sevkiyat gibi çeşitli operasyonlar yer almaktadır. Teslim alma operasyonu, depo süreçlerinin ilk aşamasıdır ve ürün tedarikçiden geldiğinde bu operasyonda indirme, tanımlama, kontrol ve kabul işlemleri yapılmaktadır. Daha sonrasında kabul edilen ürünler depoya yerleştirilmektedir ve ardından müşteri siparişlerine göre toplama, paketleme ve sevkiyat işlemleri gerçekleştirilmektedir.

Depo operasyonlarından sipariş toplama işlemi ise müşteri siparişlerine cevap olarak talep edilen öğelerin bir depodaki depolama yerlerinden alınması süreci olarak tanımlanabilir (Masae ve diğerleri, 2020). Sipariş toplama operasyonu, toplam depo operasyon maliyetlerinin %55'ten fazlasını oluşturabileceği için, bu operasyonlara odaklanmak lojistikte yüksek verimliliğe ulaşmak için önemli bir strateji olarak değerlendirilmektedir (Bottani ve diğerleri, 2015: 645; Park, 2012: 3; Zhang ve diğerleri, 2019). Ayrıca, sipariş toplayıcıların depoda seyahat etmek için harcadıkları süre, toplam toplama süresinin neredeyse yarısını oluşturmaktadır (Çelik ve Süral, 2019; Calzavara ve diğerleri, 2019). Bunlar dikkate alındığında dağıtım merkezlerinin performansı ve genel tedarik zinciri operasyonlarının verimliliği, sipariş toplama operasyonlarından doğrudan etkilenmektedir (Teck ve Dewil, 2022). Sipariş toplama işleminin önemli olması nedeniyle geçmiş çalışmalarda, sipariş toplama verimliliğinin iyileştirilmesi için, sipariş toplama sistemlerinin Van Gils ve diğerleri (2019b), toplama stratejilerinin Resmi ve diğerleri (2022) ve düzenlerinin Tutam ve White (2019) geliştirilmesi gibi çeşitli yöntemler önerilmiştir.

Sipariş toplama operasyonu, toplanan ürünlerin sevkiyat öncesindeki tüm işlemleri kapsamaktadır. Bu yüzden ilgili literatürde toplanan ürünlerin belirli bir düzene göre sıralanması veya gruplandırılması, bu sürecin bir parçası olarak dikkate alınmaktadır (Rouwenhorst ve diğerleri, 2000). Sipariş birleştirme işleminde, siparişlerin bölünmesi yaklaşımına dayalı olarak farklı sipariş toplama stratejileri (sipariş bölme ve sipariş gruplandırma) kullanılmaktadır. Sipariş bölme yönteminde, siparişteki her öğe ayrı bir toplama birimi olarak kabul edilir, sipariş gruplandırma yönteminde ise toplama personeli öncelikle öğeleri seçer ve ardından bunları organize edip sınıflandırır (Zhu ve diğerleri, 2021). Sipariş birleştirme problemleriyle ilgili olarak literatürde Resmi ve diğerleri (2022), Cergibozan ve Tasan (2022) ile Gil-Borras ve diğerleri (2021) tarafından yapılan çalışmalar incelenebilir. Ayrıca, literatürde, sipariş birleştirme ve sipariş toplama süreçlerinin birbirinden bağımsız olmadığına dair çalışmalar da bulunmaktadır (Aerts ve diğerleri, 2021; Kübler ve diğerleri, 2020; Ardjmand ve diğerleri, 2019; Van Gils ve diğerleri, 2019a). Bu çalışmalar, siparişlerin sınıflandırılmasının verimliliğini artırmaya yönelik, gruplama ve toplama kararlarını eşzamanlı olarak optimize eden yüksek kaliteli çözümler üretmeyi hedeflemektedir (Yousefi Nejad Attari ve diğerleri, 2021). Ancak bu çalışmada, sadece siparişlerin toplanması ve yönlendirilmesi problemi dikkate alınmıştır.

Depo operasyonlarının temel sorunlarından bir diğeri ise ürünlerin depolama yerlerinin belirlenmesiyle ilgilidir. De Santis ve diğerleri (2018), literatürdeki çalışmaların büyük bir kısmının, bu çalışmayla benzer şekilde ürünlerin rastgele bir şekilde depolandığını varsaydığını (rastgele atama politikası) ve sınıf tabanlı ya da özel depolama politikalarını analiz eden çalışmaların oldukça sınırlı olduğunu ifade etmektedir. Bu kapsamda, farklı depo atama politikalarını ele alan çalışmalara örnek olarak Petersen (1999), Dukic ve Oluic (2007) ile Rao ve Adil (2013) gösterilebilmektedir. Depo atama politikası, genel depo operasyonlarının başlıca zorluklarından biri olarak kabul edilmektedir. Bu konuda kapsamlı bir derleme sunan Gu ve diğerleri (2007), depo atama problemini üç farklı kategori altında sınıflandırarak incelemektedir.

Sipariş toplama operasyonunda, operasyonel verimliliği artırarak zaman ve maliyet tasarrufu sağlayan önemli problemlerden biri de rota planlamasıdır. Bazı durumlarda, rota planlaması, süreçler arasındaki senkronizasyonun önemli olduğu üretim programlarını içermektedir. Bu durumlarda, rota planlaması, genel üretim planlama sürecinin bir alt problemi haline gelmektedir. Bu tür durumlarda, aracın her turunda toplama noktaları belirlenmektedir ve araç ürünleri ilgili depolama yerlerinden almaktadır. Araç sayısı birden fazla olduğunda ise araç sayısının belirlenmesi, hizmet yerlerinin seçilmesi ve turların başlangıç zamanlarının belirlenmesi gibi unsurlar da dikkate alınmalıdır. Bu tür bir durumda, Emde ve Boysen (2012a) tarafından önerildiği gibi, iki hiyerarşik seviyeden oluşan bir karar verme süreci kullanılabilir. Ayrıca, rota planlamasının diğer malzeme taşıma operasyonlarıyla (örneğin, teslimat) birleştirildiği çalışmalar (de Souza ve diğerleri, 2008; Golz ve diğerleri, 2012; Alnahhal ve Noche, 2015; Vaidyanathan ve diğerleri, 1999) veya tesis düzeniyle birlikte ele alındığı çalışmalar (Emde ve Boysen, 2012b) da literatürde mevcuttur.

Depo toplama operasyonlarında yalnızca rota planlamasına odaklanıldığında, çeşitli yaklaşımların kullanıldığı gözlemlenmektedir. Literatürde, depo rota planlaması için S-şekli ve en büyük boşluk sezgiselleri gibi basit ve yaygın olarak bilinen sezgisel yöntemler bulunmaktadır. Bu sezgiseller, koridor başına düşen toplama noktalarının sayısına bağlı olarak daha kısa rotalar üretmektedir; ancak, optimal rotaların elde edilmesi, ilgili prosedürlerin bir karışımının kullanılmasıyla mümkün olabilmektedir. Bu tür prosedürler her zaman en iyi rotayı sağlayamasa da basitlikleri nedeniyle uygulamada cazip hale gelmektedir. Örneğin, Masae ve diğerleri (2021), yaprak düzenine sahip bir depoyu en büyük boşluk, S-şekli, geri dönüş ve orta nokta algoritmalarıyla çözmektedir. Silva ve diğerleri (2020) ise sipariş toplama ve depolama tahsis kararlarını optimal, geri dönüş, S-şekli, orta nokta ve en büyük boşluk stratejileriyle birleştirerek, değişken komşuluk araması için bir metasezgisel formülasyon sunmaktadır. Bu tür prosedürler hakkında daha detaylı bilgiler için Caron ve diğerleri (1998; 2000), Petersen (1997; 1999), Koster ve Poort (1998) ile Petersen ve Aase (2004) çalışmaları incelenebilir.

Öte yandan, dikdörtgen biçimli (paralel koridorlu) depolarda rota planlama problemleri bu çalışmada olduğu gibi GSP kurgusu altında ele alınabilmektedir ve bu tür problemler literatürde depo özel GSP olarak adlandırılmaktadır (Lu ve diğerleri, 2016; Scholz ve diğerleri, 2016). GSP, tüm gerçek hayat depo kısıtlamalarını tam olarak temsil etmese de aracın kapasite kısıtlamaları ve ziyaretleri için zaman pencereleri gibi unsurların eklenmesiyle problemin ele alınış şeklini daha gerçekçi hale getirebilmektedir. Ancak, NP-zor bir problem olan GSP'nin optimal çözümüne ulaşmak, hesaplama açısından oldukça zorlayıcı olabilmektedir.

De Santis ve diğerleri (2018), Ratliff ve Rosenthal (1983) ile Koster ve Poort (1998) gibi çalışmalar, depo rota planlama probleminin bazı varyantlarını optimal olarak çözmek için kesin algoritmalar geliştirmişlerdir. Ancak, bu algoritmalar yalnızca belirli ve basit depo konfigürasyonları için geçerli olmaktadır ve iki bloktan daha fazla bloğa sahip depolar için çözüm sağlayamamaktadır. Bu nedenle, bu çalışmada da olduğu gibi bazı çalışmalarda da (örneğin, Roodbergen ve de Koster, 2001; Hall, 1993; Vaughan, 1999) depo rota planlama problemini çözmek için sezgisel yaklaşımlar tercih edilmiştir.

Sezgisel algoritmalar için nispeten daha güncel literatür incelendiğinde, son yıllarda depo verimliliğini artırmak amacıyla geliştirilen birçok farklı yaklaşım öne çıkmaktadır. Literatürde en çok kullanılan metasezgisel algoritmalarından biri de parçacık sürü optimizasyonu algoritmasıdır. Cheng ve diğerleri (2015), sipariş toplama ve rotalama problemini çözmek için parçacık sürüsü optimizasyonu ve KKO tabanlı hibrit bir yöntem önermiştir. Parçacık sürü optimizasyonu, seyahat mesafelerini minimize ederken, bu çalışma ile benzer şekilde KKO algoritması her parti için en uygun toplama rotasını belirlemektedir. Kübler (2020), dinamik depolama konumu ataması ve sipariş toplama rotalarının yönlendirilmesi sorununu çözmek için iteratif bir yaklaşım geliştirmiş ve parçacık sürü optimizasyonu yöntemini kullanarak sipariş toplama ve rotalama problemini çözmüştür. Cergibozan ve Taşan (2022), genetik algoritma ve parçacık sürü optimizasyonu yöntemlerini birleştirerek, mevcut yönteme kıyasla seyahat mesafelerinde %40 iyileşme sağladıklarını göstermiştir. Yousefi Nejad Attari ve diğerleri (2021), sipariş toplama ve rotalama problemini karışık tamsayılı programlama modeli olarak ele almış ve genetik algoritma, parçacık sürü optimizasyonu ve yapay arı kolonisi algoritmaları ile çözmüştür. Lin ve diğerleri (2024) ise parçacık sürü optimizasyonunu kullanarak sipariş gruplama ve rotalama problemini çözmüş ve seyahat mesafesinde %52'lik bir azalma sağlamıştır.

Literatürde en çok kullanılan metasezgisel algoritmalarından bir diğeri ise genetik algoritmadır. Ardjmand ve diğerleri (2019), benzetilmiş tavlama ve genetik algoritma yöntemlerini kullanarak toplama sırası ve rotalarını optimize etmiş, genetik algoritmanın büyük ölçekli problemlerde benzetilmiş tavlama göre daha verimli olduğunu göstermiştir. Cano ve diğerleri (2019) ise genetik algoritmayı, S-Şekil sezgisel algoritmasıyla karşılaştırıldığında %13,9 mesafe tasarrufu sağladıklarını belirtmiştir. Ayrıca, Ardjmand ve diğerleri (2020), manuel sipariş toplama operasyonları için genetik algoritma ve yapay sinir ağı kombinasyonunu sunmuş ve karışık tamsayılı programlama modelleriyle karşılaştırıldığında daha iyi sonuçlar elde etmiştir. Cano ve diğerleri (2024), genetik algoritmanın, çok bloklu yüksek seviyeli depolama sistemlerinde ortak sipariş toplama sorunları çözmek için KKO algoritmasından daha verimli performans gösterdiğini belirtmiştir.

Ayrıca literatürde farklı sezgisel algoritmalar da kullanılmıştır. Örneğin, Xie ve diğerleri (2023), otonom yönlendirilmiş araç destekli toplama sistemleri için değişken komşuluk arama algoritması kullanarak sipariş gruplama ve rota kararlarını entegre etmiştir. Guo ve diğerleri (2023), büyük ölçekli e-ticaret depoları için merkezi olmayan depolama ve bölgesel bant parti seçimini eş zamanlı olarak ele almış ve mal toplamının verimliliğini artırmaya yönelik bir koridor tabanlı yapılandırıcı gruplama algoritması geliştirmiştir. Wu ve diğerleri (2024), sipariş toplama ve rotalama problemi için bulanık tabanlı popülasyon kontrolüne sahip bir memetik algoritma sunarak mevcut yöntemlere kıyasla üstün çözüm kalitesi elde etmiştir. Zhao ve diğerleri (2024), küçük ölçekli problemler için karışık tamsayılı programlama modeli, büyük ölçekli problemler için ise uyarlanmış büyük komşuluk arama tabanlı tabu arama algoritması önermiş ve bu algoritmanın karışık

tamsayı programlama, uyarlanmış büyük komşuluk arama ve tabu arama yöntemlerinden daha üstün olduğunu göstermiştir. DHaen ve diğerleri (2024), dinamik sipariş girişleri ve kapasite kısıtlarını dikkate alan bir çevrimiçi bütünleşmiş gruplama, toplayıcı yönlendirme ve grup zamanlama algoritması geliştirmiş ve bu algoritmanın toplayıcı sayısını %12,5 oranında azalttığını belirtmiştir. Son olarak, Pugliese ve diğerleri (2024), bir perakende dağıtım şirketinin sipariş toplama sürecini optimize etmek için uyarlanmış büyük komşuluk arama algoritmasını kullanmış ve mevcut süreçten daha iyi bir sonuç aldıklarını göstermişlerdir. Bu çalışmada, rota planlama problemi için KKO algoritması kullanılmıştır.

KKO algoritması, olasılıklı bir yaklaşım ile ağlar üzerinde optimal yollar bulmayı hedefleyen bir hesaplama yöntemidir. Marco Dorigo'nun doktora tezinde sunulan ilk versiyonundan bu yana, karıncaların yiyecek kaynağı ile kolonileri arasında bir yol bulma davranışını temel alarak geliştirilmiştir. KKO metasezgiseli, temel olarak karınca çözümlerinin oluşturulması ve feromon güncelleme adımları ile iki ana prosedürden oluşmakta ve çeşitli problemleri çözmek için genel bir yöntem olarak uygulanmaktadır. KKO, farklı alanlardaki çeşitli problemleri çözmek için birçok varyant ile zenginleştirilmiştir. Bu çalışmada, GSP çözümü için KKO algoritması kullanılmakta ve ilgili prosedürler probleme uygun olarak uygulanmaktadır.

KKO algoritmasının detayları için Dorigo ve diğerleri (1999) çalışması incelenebilir. Daha kapsamlı bir inceleme için ise Dorigo ve Stutzle'nin (2004) çalışması referans alınabilir. Ayrıca, KKO uygulamaları hakkında daha fazla bilgiye Stutzle ve Hoos (2000), Dorigo ve diğerleri (2004), Dorigo ve Gambardella (1997), Dorigo ve diğerleri (1996) çalışmalarından ulaşılabilir.

KKO algoritmasının GSP'yi çözmek için uygun olduğu fikriyle uyumlu olarak, De Santis ve diğerleri (2018), satıcıların (araçların) seyahat mesafesini minimize etmek amacıyla Floyd-Warshall algoritmasıyla entegre bir KKO tabanlı yaklaşım önermektedir. Yazarlar, bu yaklaşımın farklı depo düzenlerine uyarlanabileceğini ve hesaplama zorlukları bulunan problemleri makul sürelerde çözmede esnek ve etkili olduğunu vurgulamaktadır. Ayrıca, KKO algoritmasını en kısa yol hesaplamalarıyla entegre etmek için Floyd-Warshall algoritmasını kullanmanın depo rota planlama problemini çözmede etkili bir yöntem olduğunu belirtmektedirler.

Dikdörtgen biçimli bir depodaki rota planlama problemi, bu çalışmada olduğu gibi GSP olarak ele alındığında belirli adımlar izlenmektedir. İlk adımda, depoyu temsil eden bir ağ yapısı oluşturulmaktadır. Bu ağda düğümler, depolama alanlarını; bağlantılar ise bu alanlar arasındaki mesafeleri ifade etmektedir. Ağdaki bazı düğümler, aracın her turda ziyaret etmesi gereken alım noktalarını temsil etmektedir. Başka bir deyişle, her araç turu (muhtemelen) yeni bir GSP çözümü gerektirebilmektedir. Bu süreçte, ağdaki ilgili düğümler arasındaki mesafelerin hesaplanması önem taşımaktadır. Bu hesaplama zorlukları, yalnızca turun oluşturulmasıyla (yani, GSP'nin çözülmesi) değil, aynı zamanda düğümler arasındaki en kısa yolların belirlenmesiyle de ilgilidir.

En kısa yol problemini çözmek için A*, Bellman-Ford, Dijkstra'nın, Floyd-Warshall ve Johnson'ın algoritmaları gibi çeşitli algoritmalar literatürde bulunmaktadır. Bu algoritmaların bazıları tek kaynaklı en kısa yol problemi çözerken (örneğin, Dijkstra (1959)), diğerleri ise tüm çiftlerin en kısa yol problemi çözmektedir (örneğin, Floyd (1962) ve Warshall (1962)). Ayrıca, mesafe ve zamanın birlikte olduğu iki hedefli en kısa yol problemi gibi çok amaçlı varyantları da bulunmaktadır. İki hedefli problem çözüm yaklaşımlarının karşılaştırması için Raith ve Ehrgott (2009) tarafından yapılan çalışma incelenebilir. Bu çalışma kapsamında, GSP için ağın ilgili düğümleri arasındaki en kısa yolları belirlemek için Floyd-Warshall algoritması kullanılmıştır.

Firmalarda genellikle depo yerleşim tasarımı için bilgisayar destekli tasarım kullanılmaktadır. Sadece depo tasarımcılarının küçük bir kısmı, özel bir depo tasarım yazılımı kullanmaktadır (Baker ve Canessa, 2009). Pek çok firma, depolarını deneyimlerine ve pratik kurallarına göre tasarlamaktadır ve daha sonra bilgisayar destekli tasarım yazılım paketlerini kullanarak yerleşim planlarını çizim haline getirmektedir. Literatürde sınırlı sayıda özel depo tasarımı yazılımı oluşturan çalışma bulunmaktadır. Bunlardan biri olan, Roodbergen ve diğerleri (2008) oluşturdukları yazılımlarında, paralel koridorlardan oluşan bir veya birden fazla blok içeren uygun bir yerleşim yapısı belirleyerek toplama alanındaki seyahat mesafelerini minimize eden bir model sunmuştur. Çalışmada, S-şekli, en büyük boşluk sezgisel yöntemleriyle optimal rota bulunmuştur. Depo tasarım yazılımı için bir diğer çalışmada Ozden ve diğerleri (2021) tarafından gerçekleştirilmiştir. Oluşturulan yazılımda, depo tasarımı için göz önünde bulundurulacak parametreler arasında dikdörtgen deponun en-boy oranı, çapraz koridorların sayısı ve konumu, toplama koridorlarının sayısı ve konumu yer almaktadır. Geliştirilen arayüzün temel bileşenleri; toplama listesi verilerinin içe aktarılması, depo yerleşim planının bir ağ olarak oluşturulması, ürünlerin depo içindeki yerleşiminin belirlenmesi, toplayıcıların yönlendirilmesi ve metasezgisel bir algoritma kullanılarak tasarım optimizasyonun sağlanmasıdır. Çalışmada, Lin-Kernighan sezgiseli ile optimal rota bulunmuştur. Bu çalışmada oluşturulan yazılımda benzer şekilde dikdörtgen deponun en-boy bilgileri, dikey ve yatay koridor sayısı, dikey ve yatay koridorların arasındaki mesafe, toplama noktalarının sayısı yer almaktadır. Geliştirilen arayüzün temel bileşenleri; depo

yapısını oluşturmak için gerekli parametrelerin girilebildiği alanlar, oluşturulan depo yapısının kaydedilmesi ve daha önce kaydedilmiş depo tasarımlarının açılmasını sağlayan seçenekler ve depo krokisinin görüntülenmesi için ayrılmış bölümlerden oluşmaktadır. Ayrıca, çalışma kapsamında KKO algoritmasıyla rota oluşturulması için algoritma parametrelerinin girilebildiği alanlar ve rotanın görsel olarak sunulmasını sağlayan butonlar arayüzde bulunmaktadır.

Çalışmanın en önemli katkısını, depo rota planlama problemi için Floyd-Warshall algoritması ile karınca kolonisi optimizasyonu algoritmasını bir arada kullanan esnek bir çözüm yaklaşımı çerçevesi sunulması oluşturmaktadır. Bu çerçeve, farklı senaryolara uyum sağlayarak etkin ve bütünlük bir çözüm sunmayı amaçlamaktadır. Ayrıca, önerilen yaklaşımın performansını göstermek amacıyla çeşitli veri kümeleriyle deneyler gerçekleştirilmiştir.

3. METODOLOJİ

Bu bölüm önerilen çözüm yaklaşımının ayrıntılarını (en kısa yol hesaplamalarını ve KKO algoritmasını) içermektedir. Önerilen metodolojiyi detaylandırmadan önce, çalışmada yapılan varsayımlar aşağıda verilmiştir.

- Depo, paralel uzun koridorlara sahip dikdörtgen bir şekle sahiptir.
- Önerilen yaklaşım sadece depolamanın ilk seviyesinde çalışır.
- Sistemde belirli bir zamanda yalnızca tek bir araç bulunmaktadır.
- Araç, bir koridorda her iki yönde de seyahat edebilir.
- Araç, bir koridorda her iki taraftan da ürün alabilir.
- Araç, bir koridorda yön değiştirebilir.
- Araç tura depoda başlar ve depoda tamamlar.

3.1. En Kısa Yol Problemi ve Floyd-Warshall Algoritması

En kısa yol problemi, bir ağdaki iki düğüm arasındaki yolun toplam maliyetini (veya mesafesini) en az yapacak şekilde bulma problemidir. Ağ, bir dizi düğümden ve bu düğümleri birbirine bağlayan bağlantılardan oluşmaktadır. En kısa yol problemi genellikle her bir bağlantının bir maliyetinin veya uzunluğunun olduğu bir ağırlıklandırılmış ağ üzerinde ele alınmaktadır.

Ağda, düğüm kümesi V ve bağlantı kümesi A olarak tanımlanmaktadır. Her i ve j düğümleri arasındaki maliyet (mesafe) c_{ij} olarak ifade edilmektedir. En kısa yol problemi formülasyonunda değişken tanımlamaları i ve j arasındaki bağlantı, en kısa yol üzerinde ise $x_{ij} = 1$ ve aksi halde $x_{ij} = 0$ şeklinde yapılmaktadır ve ilgili problem aşağıdaki şekilde sunulmaktadır.

$$\min z = \sum_{i=1}^N \sum_{j=1}^N c_{ij} x_{ij} \quad (1)$$

Amaç fonksiyonu (Eşitlik 1) aşağıdaki kısıtlar dikkate alınarak minimize edilmektedir. Burada i başlangıç ve varış noktası ise sırasıyla $b_i = 1$ ve $b_i = -1$ değerlerini almakta olup aksi halde 0 değerini almaktadır.

$$\sum_{j=1}^N x_{ij} - \sum_{j=1}^N x_{ji} = b_i, \quad i = 1, 2, \dots, N \quad (2)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i, j \quad (3)$$

Modelde amaç fonksiyonu (Eşitlik 1) yolun toplam mesafesinin minimizasyonunu amaçlarken Eşitlik 2 başlangıç ve varış noktasına giden yolun oluşturulmasını sağlamaktadır. Eşitlik 3 ise işaret tanımlarını göstermektedir.

Çalışma kapsamında en kısa yol problemini çözmek için kullanılan Floyd-Warshall algoritması, bir ağ yapısındaki tüm düğümler arasındaki minimum mesafeyi bulmayı amaçlamaktadır. Algoritma, n düğüm sayısı için $O(n^3)$ karmaşıklığa sahiptir. Algoritma adımları, öncelikle bir uzaklık matrisi oluşturularak başlamaktadır. Bu matrisin başlangıç değerleri öncelikle sonsuz olarak atanmaktadır, ardından her bağlantı için matris güncellenmektedir ve köşegen sıfıra ayarlanmaktadır. Her iki düğüm çifti için diğer düğümler kontrol edilmektedir ve daha kısa bir yolun varlığı döngülerle araştırılmaktadır. Buna göre güncellenen uzaklık ve öncelik matrisleri kullanılarak ağ üzerindeki en kısa yollar belirlenmektedir. Floyd Warshall algoritmasının sözde kodu Şekil 2'de gösterilmiştir.

```

dist matrisi  $|V| \times |V|$  boyutunda minimum uzaklıkları gösteren
başlangıçta tüm değerleri  $\infty$  olan bir matris olsun.

for each edge (u, v) do // her kenar için
    dist[u][v]  $\leftarrow$  w(u, v) // (u, v) kenarının ağırlığı

for each vertex v do // her düğüm için
    dist[v][v]  $\leftarrow$  0

for k from 1 to |V|
    for i from 1 to |V|
        for j from 1 to |V|
            if dist[i][j] > dist[i][k] + dist[k][j]
                dist[i][j]  $\leftarrow$  dist[i][k] + dist[k][j]
            end if

```

Şekil 2. Floyd-Warshall algoritması sözde kodu (Floyd, 1962)

3.2. GSP ve KKO Algoritması

Çalışma kapsamında depodaki aracın rotalaması bir GSP olarak kurgulanmıştır. Buna göre, önceki alt bölümdekine benzer şekilde tanımlanmış bir ağ üzerinde bir satıcı (araç), i . şehirden (depo noktası düğümünden) j . şehre (depo noktası düğümüne) giderse $x_{ij} = 1$ aksi durumda $x_{ij} = 0$ olarak tanımlanır, aşağıdaki amaç fonksiyonu (Eşitlik 4) turun toplam maliyetini diğer bir deyişle mesafeyi en aza indirmek için kullanılır. Burada c_{ij} , i ile j düğümleri arasındaki mesafeyi göstermektedir.

$$\min z = \sum_{i=1}^N \sum_{j=1}^N c_{ij} x_{ij} \quad (4)$$

$$\sum_{i=1}^N x_{ij} = 1, \quad j = 1, 2, \dots, N \quad (5)$$

$$\sum_{j=1}^N x_{ij} = 1, \quad i = 1, 2, \dots, N \quad (6)$$

$$u_i - u_j + N x_{ij} \leq N - 1, \quad \forall i, j \quad i \neq j \quad (7)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i, j \quad (8)$$

$$u_j \geq 0, \quad \forall j \quad (9)$$

Modelde, amaç fonksiyonu toplam mesafeyi en aza indirmekte olup Eşitlik 4 ile gösterilmiştir. Eşitlik 5 ve Eşitlik 6, satıcının turunu tamamlamak için her bir düğümü ziyaret etmesini, Eşitlik 7 ise alt tur oluşumunun engellenmesini sağlanmaktadır. Eşitlik 8 ve Eşitlik 9 işaret kısıtlarını göstermektedir.

Çalışma kapsamında GSP probleminin çözümü için bir karınca sistemi olarak da adlandırılan KKO algoritması kullanılmıştır. Karınca sistemi iki ana aşamaya ayrılmaktadır. Başlangıç feromon izleri, $\tau_{ij} = m/C^0$ ile belirlenerek elde edilmektedir. Burada m karınca sayısını ve C^0 ise uygun bir prosedürle oluşturulan bir turun uzunluğunu göstermektedir.

Karınca sisteminin çözüm oluşturma aşamasında, m karınca eş zamanlı olarak bir GSP turu oluşturmaktadır. Başlangıçta, her karınca rastgele seçilmiş bir düğümde bulunmaktadır ve her adımda, karınca k , bir sonraki ziyaret edeceği düğümü belirlemek için olasılıksal bir seçim kuralı uygulamaktadır. Burada p_{ij}^k , Eşitlik 10 ile hesaplanan k numaralı karıncanın i düğümünden j düğümüne gitme olasılığı olarak tanımlanmaktadır.

$$p_{ij}^k = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{l \in \mathcal{N}_i^k} (\tau_{il})^\alpha (\eta_{il})^\beta}, \quad j \in \mathcal{N}_i^k \quad (10)$$

Burada $\eta_{ij} = 1/d_{ij}$ olarak tanımlanmaktadır. $\mathcal{N}_{ik}$, karınca k , i düğümüne geldiğinde izin verilen komşuluk (yani, izin verilen düğüm kümesi) olarak tanımlanmaktadır. α ile β sırasıyla τ_{ij} ve η_{ij} değerlerinin göreceli etkisini belirleyen iki parametre olarak tanımlanmaktadır. İki sınır durumun etkilerini burada belirtmek mümkündür. $\alpha = 0$ ise, klasik bir stokastik aç gözlü algoritma ile benzer şekilde, en yakın şehirlerin seçilme olasılığı daha yüksektir. $\beta = 0$ seçildiği bir durumda ise sadece feromon genellikle kötü sonuçlara yol açacak şekilde kullanılmaktadır ve özellikle $\alpha > 1$ olduğunda, tüm karıncaların aynı yolu takip ettiği ve bu nedenle aynı alt-optimal turu oluşturduğu bir durum hızlı bir şekilde ortaya çıkmaktadır (Dorigo ve diğerleri, 1996). Doğru parametre değerlerinin seçimi için Dorigo ve Stutzle (2004) çalışması incelenebilir.

Tüm karıncalar turlarını oluşturduktan sonra, ikinci aşamasında feromon izleri güncellenmektedir. Feromon izleri güncellemesi, öncelikle tüm kollar üzerindeki feromon değerinin sabit bir faktörle azaltılması ile başlamakta ve ardından karıncaların turlarında geçtikleri kollara feromon eklenerek Eşitlik 11 tamamlanmaktadır.

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k, \quad \forall (i, j) \in L \quad (11)$$

Burada $0 < \rho \leq 1$ olan ρ , feromon izlerinin sınırsız birikimini önlemek ve önceki kötü kararların etkisini ortadan kaldırmak için kullanılan feromon buharlaşma oranıdır. $\Delta\tau_{ij}^k$, k karıncasının ziyaret ettiği kollar üzerine bıraktığı feromon miktarını ifade etmekte ve Eşitlik 12 ile hesaplanmaktadır.

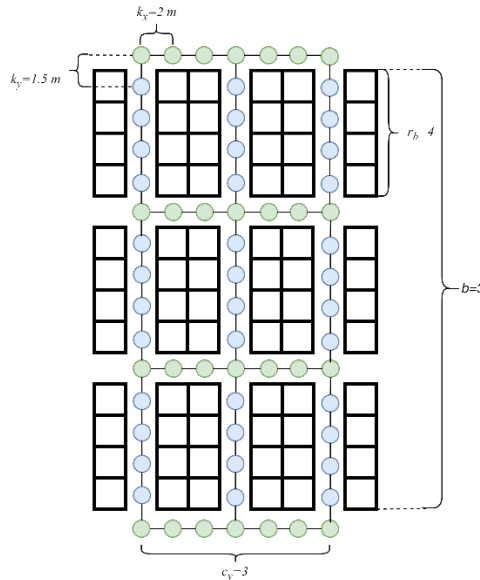
$$\Delta\tau_{ij}^k = \begin{cases} 1/C^k, & (i, j) \in T^k \\ 0 & \text{a. d.} \end{cases} \quad (12)$$

C_k , karınca k tarafından oluşturulan T_k turunun uzunluğunu göstermektedir. Kullanılan kollar arasında daha fazla karınca tarafından kullanılanlar ve daha kısa turların parçası olanlar, daha fazla feromon almaktadır. Bu nedenle gelecekteki iterasyonlarda karıncalar tarafından seçilme olasılığı daha yüksek olmaktadır.

KKO uygulaması için standart parametrelerin yanı sıra (α, β, ρ vb.), GSP olarak karşılık gelen rota planlama problemi bağlamında turun başlangıç noktası ve turu tamamlamak için ziyaret edilecek diğer düğümlerle ilgili verilere ihtiyaç duyulmaktadır. Bir depo yapısındaki başlangıç noktası genellikle aracın turuna başladığı nokta olmakla birlikte, önerilen yaklaşım deponun yanı sıra herhangi bir noktadan tura başlama olanağı sağlayan esnek bir uygulama sağlamayı amaçlamaktadır. Bu tür esneklik, aracın belirli bir zamanda depo dışında bir noktada bulunması ve ardından depo içinde bir noktaya yeniden yönlendirilmesi gerektiği veya tercih edildiği dinamik bir depo ortamında faydalı olabilmektedir. Bu çalışmada, başlangıç noktasını ağıın herhangi bir noktası olarak tanımlamaya izin veren esnek bir çerçeve sunulması amaçlanmıştır.

4. SAYISAL ÖRNEK

Oluşturulan sayısal örnekte, öncelikle dikdörtgen bir depoda, depolama alanlarına bağlı koridor düğümlerinin tanımlanması gerekmektedir. Şekil 3'te, dört bitişik depolama alanıyla ($r_b = 4$), iki yatay ($b = 3$) ve üç dikey ($c_v = 3$) koridora sahip basit bir dikdörtgen depo yapısı sunulmaktadır. Oluşturulan sayısal örnekte, yatay koridorlar arasındaki iki düğüm arası mesafe 2 m ($k_x = 2$ m) ve dikey koridor düğümleri arası mesafe 1,5 m ($k_y = 1,5$ m) olarak belirlenmiştir.



Şekil 3. Sayısal örnek için depo tasarımı

Python programlama dili kullanılarak geliştirilen bir arayüz aracılığıyla, öncelikle kullanıcıdan alınan depo parametreleri bilgilerine dayanarak dinamik bir depo modeli ve daha sonrasında, ilgili problem için bir uzaklık matrisi oluşturulmaktadır. Oluşturulan depo yapısı kullanıcıya görselleştirilerek sunulmakta ve GSP için KKO algoritması ile elde edilen minimum maliyetli rota çizimi simüle edilmektedir.

Depodaki mesafe hesaplamalarını daha sistemli hale getirmek için, Şekil 3'teki ağ gösterimine sahip olan depo, Şekil 4'te gösterildiği gibi iki boyutlu bir matrise dönüştürülmüştür. Bu matris, düğümler arasındaki

mesafelerin kolayca hesaplanmasını sağlamaktadır ve koridor düğümlerinin bağlantılı olduğu depolama yerlerini daha rahat bulmaya olanak tanımaktadır.

-	0	1	2	3	4	5	6	7	8
0		K	K	K	K	K	K	K	
1	D	K	D	D	K	D	D	K	D
2	D	K	D	D	K	D	D	K	D
3	D	K	D	D	K	D	D	K	D
4	D	K	D	D	K	D	D	K	D
5		K	K	K	K	K	K	K	
6	D	K	D	D	K	D	D	K	D
7	D	K	D	D	K	D	D	K	D
8	D	K	D	D	K	D	D	K	D
9	D	K	D	D	K	D	D	K	D
10		K	K	K	K	K	K	K	
11	D	K	D	D	K	D	D	K	D
12	D	K	D	D	K	D	D	K	D
13	D	K	D	D	K	D	D	K	D
14	D	K	D	D	K	D	D	K	D
15		K	K	K	K	K	K	K	

(a)

-	0	1	2	3	4	5	6	7	8
0		1;0	2;0	3;0	4;0	5;0	6;0	7;0	
1	0;1	1;1	2;1	3;1	4;1	5;1	6;1	7;1	8;1
2	0;2	1;2	2;2	3;2	4;2	5;2	6;2	7;2	8;2
3	0;3	1;3	2;3	3;3	4;3	5;3	6;3	7;3	8;3
4	0;4	1;4	2;4	3;4	4;4	5;4	6;4	7;4	8;4
5		1;5	2;5	3;5	4;5	5;5	6;5	7;5	
6	0;6	1;6	2;6	3;6	4;6	5;6	6;6	7;6	8;6
7	0;7	1;7	2;7	3;7	4;7	5;7	6;7	7;7	8;7
8	0;8	1;8	2;8	3;8	4;8	5;8	6;8	7;8	8;8
9	0;9	1;9	2;9	3;9	4;9	5;9	6;9	7;9	8;9
10		1;10	2;10	3;10	4;10	5;10	6;10	7;10	
11	0;11	1;11	2;11	3;11	4;11	5;11	6;11	7;11	8;11
12	0;12	1;12	2;12	3;12	4;12	5;12	6;12	7;12	8;12
13	0;13	1;13	2;13	3;13	4;13	5;13	6;13	7;13	8;13
14	0;14	1;14	2;14	3;14	4;14	5;14	6;14	7;14	8;14
15		1;15	2;15	3;15	4;15	5;15	6;15	7;15	

(b)

Şekil 4. Deponun matris gösterimi

Depoların tanımlanmasından sonra, en kısa yol hesaplamalarının yapılması için bir prosedür oluşturulması gerekmektedir. Bu prosedürde ilk adım öncelikle, düğümler arasındaki uzaklık (k_x , k_y) parametrelerinin tanımlanması ve başlangıç uzaklık matrisinin oluşturulmasıdır. Daha sonra, en kısa yol mesafelerinin belirlenebilmesi için Floyd-Warshall algoritması ile uzaklık matrisi oluşturulmaktadır. Koridor düğümleri arasındaki mesafelerin belirlenmesi için k_x , k_y parametreleri kullanılarak ve komşu koridor düğümleri arası uzaklıklar matrise aktararak D_0 uzaklık matrisi oluşturulmaktadır. Daha sonra, Floyd-Warshall algoritması uzaklık matrisine uygulanarak düğümler arasındaki en kısa mesafe belirlenmektedir. Sayısal örnek olarak verilen depodaki tüm koridor düğümleri için düğümler arası başlangıç uzaklık matrisi D_0 , Şekil 5'te verilmiştir. Floyd-Warshall algoritması iterasyonları D_0 başlangıç matrisi üzerinde uygulanır ve böylece tüm düğümler arasındaki mesafe matrisi oluşturulur.

D_0	1;0	1;1	1;2	1;3	1;4	1;5	2;0	2;5	4;0	4;1	4;2	4;3	4;4	4;5	3;0	3;5
1;0	0	k_y	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
1;1	k_y	0	k_y	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
1;2	∞	k_y	0	k_y	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
1;3	∞	∞	k_y	0	k_y	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
1;4	∞	∞	∞	k_y	0	k_y	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
1;5	∞	∞	∞	∞	k_y	0	∞	k_x	∞	∞	∞	∞	∞	∞	∞	∞
2;0	k_x	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	∞	k_x	∞
2;5	∞	∞	∞	∞	∞	k_x	0	∞	∞	∞	∞	∞	∞	∞	∞	k_x
4;0	∞	∞	∞	∞	∞	∞	∞	0	k_y	∞	∞	∞	∞	∞	k_x	∞
4;1	∞	∞	∞	∞	∞	∞	∞	∞	k_y	0	k_y	∞	∞	∞	∞	∞
4;2	∞	∞	∞	∞	∞	∞	∞	∞	∞	k_y	0	k_y	∞	∞	∞	∞
4;3	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	k_y	0	k_y	∞	∞	∞
4;4	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	k_y	0	k_y	∞	∞
4;5	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	k_y	0	∞	k_x
3;0	∞	∞	∞	∞	∞	∞	k_x	∞	k_x	∞	∞	∞	∞	∞	0	∞
3;5	∞	∞	∞	∞	∞	∞	∞	k_x	∞	∞	∞	∞	∞	∞	k_x	0

Şekil 5. Floyd-Warshall başlangıç matrisi

Floyd-Warshall algoritmasının adımları D_0 matrisi için gerçekleştirildiğinde, Şekil 6-(a)'daki D matrisi elde edilmektedir. Düğümler arasındaki en kısa yol ve bu en kısa yolun rotası D matrisi kullanılarak bulunmaktadır.

Şekil 6-(a)'daki matrise ek olarak en kısa yolun hangi rota üzerinden ulaşılacağı bilgisinin tutulması gerekmektedir. Şekil 6-(b)'deki P matrisinde, her bir düğümden diğer düğümlere olan en kısa yollara, hangi düğüm üzerinden ulaşılacağı görülmektedir. Örneğin, "1;1" düğümü ve "4;4" düğümü için D matrisi incelendiğinde, en kısa yol 13,5 m olarak belirlenir ve P matrisi incelendiğinde ise 1;1, 1;0, 2;0, 3;0, 4;0, 4;1, 4;2, 4;3 ve 4;4 rotası izlenir.

D	1:0	1:1	1:2	1:3	1:4	1:5	2:0	2:5	4:0	4:1	4:2	4:3	4:4	4:5	3:0	3:5
1:0	0,0	1,5	3,0	4,5	6,0	7,5	2,0	9,5	6,0	7,5	9,0	10,5	12,0	13,5	4,0	11,5
1:1	1,5	0,0	1,5	3,0	4,5	6,0	3,5	8,0	7,5	9,0	10,5	12,0	13,5	12,0	5,5	10,0
1:2	3,0	1,5	0,0	1,5	3,0	4,5	5,0	6,5	9,0	10,5	12,0	13,5	12,0	10,5	7,0	8,5
1:3	4,5	3,0	1,5	0,0	1,5	3,0	6,5	5,0	10,5	12,0	13,5	12,0	10,5	9,0	8,5	7,0
1:4	6,0	4,5	3,0	1,5	0,0	1,5	8,0	3,5	12,0	13,5	12,0	10,5	9,0	7,5	10,0	5,5
1:5	7,5	6,0	4,5	3,0	1,5	0,0	9,5	2,0	13,5	12,0	10,5	9,0	7,5	6,0	11,5	4,0
2:0	2,0	3,5	5,0	6,5	8,0	9,5	0,0	11,5	4,0	5,5	7,0	8,5	10,0	11,5	2,0	11,5
2:5	9,5	8,0	6,5	5,0	3,5	2,0	11,5	0,0	11,5	10,0	8,5	7,0	5,5	4,0	13,5	2,0
4:0	6,0	7,5	9,0	10,5	12,0	13,5	4,0	11,5	0,0	1,5	3,0	4,5	6,0	7,5	2,0	9,5
4:1	7,5	9,0	10,5	12,0	13,5	12,0	5,5	10,0	1,5	0,0	1,5	3,0	4,5	6,0	3,5	8,0
4:2	9,0	10,5	12,0	13,5	12,0	10,5	7,0	8,5	3,0	1,5	0,0	1,5	3,0	4,5	5,0	6,5
4:3	10,5	12,0	13,5	12,0	10,5	9,0	8,5	7,0	4,5	3,0	1,5	0,0	1,5	3,0	6,5	5,0
4:4	12,0	13,5	12,0	10,5	9,0	7,5	10,0	5,5	6,0	4,5	3,0	1,5	0,0	1,5	8,0	3,5
4:5	13,5	12,0	10,5	9,0	7,5	6,0	11,5	4,0	7,5	6,0	4,5	3,0	1,5	0,0	9,5	2,0
3:0	4,0	5,5	7,0	8,5	10,0	11,5	2,0	13,5	2,0	3,5	5,0	6,5	8,0	9,5	0,0	11,5
3:5	11,5	10,0	8,5	7,0	5,5	4,0	11,5	2,0	9,5	8,0	6,5	5,0	3,5	2,0	11,5	0,0

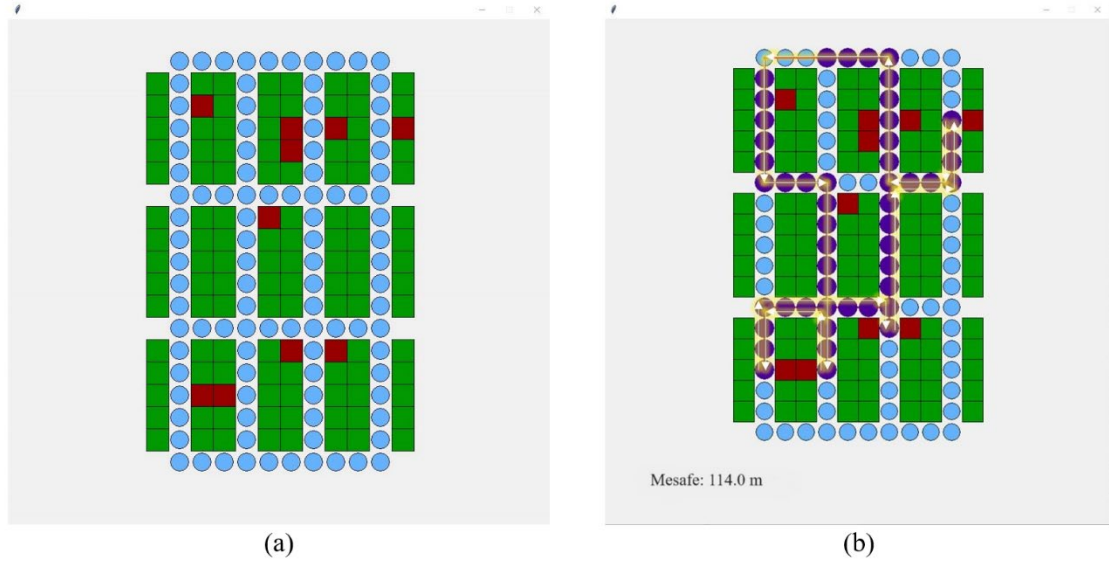
(a)

P	1:0	1:1	1:2	1:3	1:4	1:5	2:0	2:5	4:0	4:1	4:2	4:3	4:4	4:5	3:0	3:5
1:0	-	1:0	1:1	1:2	1:3	1:4	1:0	1:5	3:0	4:0	4:1	4:2	4:3	4:4	2:0	2:5
1:1	1:1	-	1:1	1:2	1:3	1:4	1:0	1:5	3:0	4:0	4:1	4:2	4:3	3:5	2:0	2:5
1:2	1:1	1:2	-	1:2	1:3	1:4	1:0	1:5	3:0	4:0	4:1	4:4	4:5	3:5	2:0	2:5
1:3	1:1	1:2	1:3	-	1:3	1:4	1:0	1:5	3:0	4:0	4:1	4:4	4:5	3:5	2:0	2:5
1:4	1:1	1:2	1:3	1:4	-	1:4	1:0	1:5	3:0	4:2	4:3	4:4	4:5	3:5	2:0	2:5
1:5	1:1	1:2	1:3	1:4	1:5	-	1:0	1:5	3:0	4:2	4:3	4:4	4:5	3:5	2:0	2:5
2:0	2:0	1:0	1:1	1:2	1:3	1:4	-	1:5	3:0	4:0	4:1	4:2	4:3	4:4	2:0	2:5
2:5	1:1	1:2	1:3	1:4	1:5	2:5	1:0	-	4:1	4:2	4:3	4:4	4:5	3:5	2:0	4:5
4:0	2:0	1:0	1:1	1:2	1:3	1:4	3:0	3:5	-	4:0	4:1	4:2	4:3	4:4	4:0	4:5
4:1	2:0	1:0	1:1	1:2	1:3	2:5	3:0	3:5	4:1	-	4:1	4:2	4:3	4:4	4:0	4:5
4:2	2:0	1:0	1:1	1:2	1:5	2:5	3:0	3:5	4:1	4:2	-	4:2	4:3	4:4	4:0	4:5
4:3	2:0	1:0	1:1	1:4	1:5	2:5	3:0	3:5	4:1	4:2	4:3	-	4:3	4:4	4:0	4:5
4:4	2:0	1:0	1:3	1:4	1:5	2:5	3:0	3:5	4:1	4:2	4:3	4:4	-	4:4	4:0	4:5
4:5	2:0	3:5	1:3	1:4	1:5	2:5	3:0	3:5	4:1	4:2	4:3	4:4	4:5	-	4:0	4:5
3:0	2:0	1:0	1:1	1:2	1:3	1:4	3:0	3:5	3:0	4:0	4:1	4:2	4:3	4:4	-	4:5
3:5	1:1	1:2	1:3	1:4	1:5	2:5	3:0	3:5	4:1	4:2	4:3	4:4	4:5	3:5	4:0	-

(b)

Şekil 6. Floyd-Warshall (a) uzaklık matrisi ve (b) öncelik matrisi

Sayısal örnek için görselleştirilen depo modeli, Şekil 7-(a)'da gösterilmektedir. Şekil 7-(a) incelendiğinde, mavi daireler koridor düğümlerini, yeşil kareler depolama yerlerini, aralarındaki kırmızı kareler ise rassal olarak oluşturulan depolama yerlerini temsil eder. Şekil 7-(b)'de, KKO algoritması kullanılarak tamamlanan rota gösterilmektedir.



Şekil 7. (a) Örnek deponun görselleştirilmesi ve (b) rotalaması

5. BULGULAR ve TARTIŞMA

Bu bölümde, depo boyutları ve toplama noktası miktarları değiştirilerek farklı uygulama problemlerinin çözümleri sağlanmaktadır. Hesaplamalar, Windows 10 işletim sistemine sahip, Intel Core i7-7700HQ 2.80GHz işlemci ve 16 GB RAM konfigürasyonuna sahip bir bilgisayarda Python programlama dili kullanılarak gerçekleştirilmiştir.

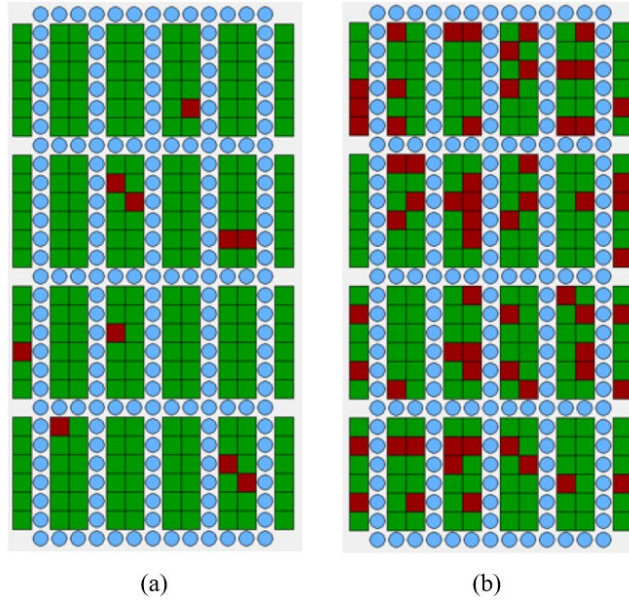
Çalışma kapsamında, çözüm performansı toplama yeri sayısındaki değişime ve KKO algoritması parametrelerinin değişimine göre incelenmiştir. Toplama yeri sayısının, 1 ile 80 arası değiştiği on farklı veri seti oluşturulmuş ve her bir veri setinde depolama yeri 240 olarak sabit tutulmuştur. Toplama yeri sayısındaki değişim ile birlikte rota uzunluğundaki ve çözüm süresindeki değişimler Tablo 1'de gösterilmiştir.

Tablo 1. Toplama yeri sayısındaki değişimin rota uzunluğuna ve çözüm süresine etkisi

Veri Seti	VS-1	VS-2	VS-3	VS-4	VS-5	VS-6	VS-7	VS-8	VS-9	VS-10
Toplama Yeri Sayısı	1	5	10	20	30	40	50	60	70	80
Rota Uzunluğu (m)	24	117	189	195	258	297	351	375	360	414
Çözüm Süresi (s)	0,01	0,03	0,11	0,26	0,36	0,64	0,76	0,66	0,81	1,30

Tablo 1 incelendiğinde, veri setlerinde toplama yeri sayısının arttığı ve KKO algoritmasının parametrelerinin sabit olduğu durumda rota uzunluğunun genel olarak artış gösterdiği gözlemlenmektedir. Benzer şekilde,

çözüm süresi açısından incelendiğinde de toplama yeri sayısı arttıkça çözüm süresinin de genel olarak arttığı görülmektedir. Şekil 8-(a)'da örnek olarak VS-3 için depo tasarımı ve Şekil 8-(b)'de örnek olarak VS-9 için depo tasarımı gösterilmiştir.



Şekil 8. (a) VS-3 ve (b) VS-9 için depo tasarımı

Çalışma kapsamında ayrıca KKO algoritması parametrelerinin değiştirilmesinin etkisi de incelenmiştir. Oluşturulan veri setinde 240 depolama ve 20 toplama yeri (VS-4) kullanılarak KKO parametrelerinin değişimi analiz edilmiştir. KKO algoritmasının parametreleri belirlenirken, $Q = 0,1$, $\alpha = 0,8$, $\beta = 0,7$, $T = 100$ olarak sabit tutulmuş ve karınca sayısı (m), buharlaşma katsayısı (ρ) ve feromon yoğunluğunun (τ) değişimlerinin çözüm performansı üzerindeki etkisi incelenmiştir. Çözüm performansı olarak metre cinsinden rota uzunluğu (R.U.) ve saniye cinsinden çözüm süresi (Ç.S.) değerlendirilmiştir. Çözüm sonuçları Tablo 2'de gösterilmiştir. Tabloda R_i , algoritmanın çalıştırılması sırasını (i), S sabit tutulan parametreleri, D değiştirilen parametreyi göstermektedir. Çözüm sonuçları Tablo 2'de gösterilmiştir.

Tablo 2. KKO algoritmasının parametre değişimlerinin rota uzunluğuna ve çözüm süresine etkisi

S	D	Değer	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	Min.	Max.	Ort.
$\rho=0,25$ $\tau=0,25$	5	R.U.	249	213	240	252	228	231	213	234	219	252	213	252	233
		Ç.S.	0,16	0,32	0,18	0,28	0,18	0,33	0,32	0,20	0,19	0,22	0,16	0,33	0,24
	10	R.U.	219	198	204	207	192	219	204	204	216	201	192	219	206
		Ç.S.	0,34	0,65	0,66	0,47	0,33	0,35	0,31	0,39	0,30	0,30	0,30	0,66	0,41
	15	R.U.	219	201	195	210	228	198	219	216	225	222	195	228	213
		Ç.S.	0,41	0,52	0,48	0,43	0,47	0,45	0,43	0,41	0,49	0,46	0,41	0,52	0,46
$m=10$ $\tau=0,25$	0,25	R.U.	228	222	231	225	207	216	219	213	228	222	207	231	221
		Ç.S.	0,28	0,25	0,27	0,28	0,27	0,28	0,27	0,33	0,27	0,24	0,24	0,33	0,27
	0,50	R.U.	219	243	225	267	246	237	225	240	225	210	210	267	234
		Ç.S.	0,17	0,16	0,19	0,18	0,16	0,16	0,15	0,17	0,15	0,19	0,15	0,19	0,17
	0,75	R.U.	297	267	270	279	261	237	303	294	300	246	237	303	275
		Ç.S.	0,65	0,14	0,66	0,13	0,67	0,15	0,65	0,14	0,66	0,15	0,13	0,67	0,40
$m=10$ $\rho=0,25$	0,25	R.U.	219	225	204	234	222	237	240	237	243	216	204	243	228
		Ç.S.	0,64	0,38	0,64	0,37	0,33	0,33	0,38	0,35	0,46	0,40	0,33	0,64	0,43
	0,50	R.U.	231	225	225	234	228	219	225	222	243	210	210	243	226
		Ç.S.	0,71	0,64	0,28	0,31	0,34	0,32	0,66	0,29	0,34	0,37	0,28	0,71	0,43
	0,75	R.U.	234	234	210	222	228	213	207	201	234	210	201	234	219
		Ç.S.	0,27	0,30	0,35	0,33	0,26	0,30	0,39	0,66	0,66	0,33	0,26	0,66	0,39

Tablo 2'de ilk satırda, $\rho = 0,25$ ve $\tau = 0,25$ değerleri sabit tutularak karınca sayısındaki (m) değişiminin rota uzunluğu ve çözüm süresi üzerindeki etkileri incelenmiştir. Çözüm sonuçları değerlendirildiğinde, üç parametre arasındaki en iyi sonucun genel olarak 10 karınca sayısı ile elde edildiği gözlemlenmiştir. Ayrıca,

karınca sayısındaki artışın çözüm süresindeki artışla doğru orantılı olduğu gözlemlenmiştir. Tablo 2’de ikinci satırda, $m = 10$ ve $\tau = 0,25$ değerleri sabit tutularak buharlaşma katsayısı (ρ) değişiminin rota uzunluğuna ve çözüm süresine etkileri incelenmiştir. Çözüm sonuçları değerlendirildiğinde, buharlaşma katsayısının arttığında, rota uzunluklarının da genellikle arttığı gözlemlenmiştir. Son olarak, Tablo 2’de üçüncü satırda, $m = 10$ ve $\rho = 0,25$ değerleri sabit tutularak feromon yoğunluğu (τ) değişiminin etkisi incelenmiştir. İnceleme sonucunda, feromon yoğunluğu arttığında, ortalama rota uzunluğunun düzenli olarak olmasa da bir miktar azaldığı gözlemlenmiştir. Çözüm süresi açısından incelendiğinde ise buharlaşma katsayısı ve feromon yoğunluğundaki değişimin çözüm süresinde direk olarak etkili olmadığı gözlemlenmiştir.

6. SONUÇ

Tedarik zincirlerinde depolama ve malzeme taşıma operasyonları, lojistik faaliyetler arasında önemli bir yer tutmakta ve yüksek maliyet oluşturmaktadır. Özellikle iç lojistik faaliyetler kapsamında, sipariş toplama süreci, malzeme taşıma operasyonlarının kritik bir bileşeni olarak öne çıkmaktadır. Bu süreç, geleneksel depolardan süpermarket tipi depolara ve insansız depolara kadar farklı depo kurgularında karşılık bulabilmektedir. Geleneksel depolarda, taşıyıcı bir aracın belirli bir başlangıç noktasından hareket ederek talep edilen malzemeleri toplaması, ardından başlangıç noktasına geri dönerek bu malzemeleri boşaltması ile gerçekleşen bir rotalama süreci mevcuttur.

Sipariş toplamada optimum rotanın belirlenmesi, stoklanan malzemelerin en verimli şekilde taranmasını ve kaynakların etkin kullanımını sağlamayı hedeflemektedir. Optimum rota bulunarak zaman ve emek kayıplarının yanı sıra maliyetlerin de azaltılması amaçlanmaktadır. Bu süreçte, optimum ya da optimuma en yakın rotalar; sezgisel yöntemler, matematiksel modeller ve yapay zekâ teknikleri gibi çeşitli bilimsel yaklaşımlar kullanılarak elde edilebilmektedir.

Bu çalışmada, dikdörtgen şekilli depolardaki araç rota planlama problemi için esnek bir çözüm yaklaşımı sunulmuştur. Çalışmada, depoda sadece tek bir araç olduğu varsayılmaktadır ve araç belirli bir başlangıç noktasından, diğer bir deyişle depodan başlayarak, her turda önceden belirlenmiş konumlardan ürünleri alıp aynı başlangıç noktasına geri dönmektedir. Çalışma kapsamında ilgili problem bir gezgin satıcı problemi olarak modellenmiştir. Önerilen çözüm yaklaşımına göre, öncelikle depoda ziyaret edilmesi gereken noktaların bazılarını temsil eden bir ağ gösterimi oluşturulmaktadır. Daha sonra, her turda bir gezgin satıcı problemi oluşturmak için ve ağ yapısının ilgili düğümler arasındaki en kısa yolların belirlenmesi için Floyd-Warshall algoritması kullanılmaktadır. Son olarak, aracın toplam seyahat mesafesini minimize ederek ilgili gezgin satıcı probleminin çözümünü sağlamak için karınca kolonisi optimizasyonu algoritması kullanılmaktadır. Çalışmanın en önemli katkısı, depo rota planlama problemini çözmek için Floyd-Warshall algoritması ile karınca kolonisi optimizasyonu algoritmasını bir arada kullanan esnek bir çözüm yaklaşımı sunulmasıdır. Ayrıca, önerilen yaklaşımın performansını göstermek amacıyla çeşitli veri kümeleriyle deneyler gerçekleştirilmiştir.

Bütünleşik bir çözüm metodolojisi geliştirmemize rağmen, gelecekteki çalışmalarda dikkate alınabilecek olan bazı unsurlardan bahsedilebilir. Bunlardan biri, gerçek hayat depo ortamları için kısıtlayıcı olabilecek olan belirli bir zamanda sistemde yalnızca bir aracın olması varsayımının kaldırılması olabilir. Bu durumda, metodolojimizin, çoklu araç kurgusuna uyacak şekilde güncellenmesi gerekmektedir. Gelecekteki çalışmalarda, Floyd-Warshall ve KKO algoritmaları yerine alternatif çözüm yaklaşımları kullanılması da dikkate alınabilir.

Bilgilendirme /Acknowledgements

Bu çalışmada süresince, Gülveren Tabansız Göç’e sağlanan destekler için TÜBİTAK’a teşekkür ederiz. *We would like to thank TUBITAK for the supports provided to Gülveren Tabansız Göç during this study.*

Yazar Katkıları /Author Contributions

Furkan Ulu: Literatür taraması, Kavramsallaştırma, Metodoloji, Veri Derleme, Analiz *Gülveren Tabansız Göç:* Literatür taraması, Makale Yazımı-inceleme ve düzenleme *Fatih Çavdur:* Literatür taraması, Kavramsallaştırma, Metodoloji, Makale Yazımı-inceleme ve düzenleme *Furkan Ulu:* Literature review, Conceptualization, Methodology, Data Curation, Analysis *Gülveren Tabansız Göç:* Literature review, Writing-review and editing *Fatih Çavdur:* Literature review, Conceptualization, Methodology, Writing-review and editing.

Çatışma Beyanı /Conflict of Interest

Yazarlar tarafından herhangi bir potansiyel çıkar çatışması beyan edilmemiştir. *No potential conflict of interest was declared by the authors.*

Fon Desteği / Funding

Bu çalışmada herhangi bir resmi, ticari ya da kâr amacı gütmeyen organizasyondan fon desteği alınmamıştır.

Any specific grant has not been received from funding agencies in the public, commercial, or not-for-profit sectors.

Etik Standartlara Uygunluk / Compliance with Ethical Standards

Yazarlar tarafından, çalışmada kullanılan araç ve yöntemlerin Etik Kurul izni gerektirmediği beyan edilmiştir.

It was declared by the authors that the tools and methods used in the study do not require the permission of the Ethics Committee.

Etik Beyanı / Ethical Statement

Yazarlar tarafından bu çalışmada bilimsel ve etik ilkelere uyulduğu ve yararlanılan tüm çalışmaların kaynakçada belirtildiği beyan edilmiştir.

It was declared by the authors that scientific and ethical principles have been followed in this study and all the sources used have been properly cited.



Yazarlar, Verimlilik Dergisi'nde yayımlanan çalışmalarının telif hakkına sahiptirler ve çalışmaları CC BY-NC 4.0 lisansı altında yayımlanmaktadır.

The authors own the copyright of their works published in Journal of Productivity and their works are published under the CC BY-NC 4.0 license.

KAYNAKÇA

- Aerts, B., Cornelissens, T. ve Sörensen, K. (2021). "The Joint Order Batching and Picker Routing Problem: Modelled and Solved as a Clustered Vehicle Routing Problem", *Computers & Operations Research*, 129, 105168. <https://doi.org/10.1016/j.cor.2020.105168>
- Alnahhal, M. ve Noche, B. (2015). "Dynamic Material Flow Control in Mixed Model Assembly Lines", *Computers & Industrial Engineering*, 85, 110-119. <https://doi.org/10.1016/j.cie.2015.03.009>
- Ardjmand, E., Bajgirani, O.S. ve Youssef, E. (2019). "Using List-Based Simulated Annealing and Genetic Algorithm for Order Batching and Picker Routing in Put Wall Based Picking Systems", *Applied Soft Computing*, 75, 106-119. <https://doi.org/10.1016/j.asoc.2018.11.019>
- Ardjmand, E., Ghalekhondabi, I., Young II, W.A., Sadeghi, A., Weckman, G.R. ve Shakeri, H. (2020). "A Hybrid Artificial Neural Network, Genetic Algorithm and Column Generation Heuristic for Minimizing Makespan in Manual Order Picking Operations", *Expert Systems with Applications*, 159, 113566. <https://doi.org/10.1016/j.eswa.2020.113566>
- Baker, P. ve Canessa, M. (2009). "Warehouse Design: A Structured Approach", *European Journal of Operational Research*, 193(2), 425-436. <https://doi.org/10.1016/j.ejor.2007.11.045>
- Bottani, E., Montanari, R., Rinaldi, M. ve Vignali, G. (2015). "Intelligent Algorithms for Warehouse Management", *Intelligent Techniques in Engineering Management, Intelligent Systems Reference Library*, 87, (Editör: C. Kahraman, S. Çevik Onar), Springer, Cham, 645-667. https://doi.org/10.1007/978-3-319-17906-3_25
- Calzavara, M., Sgarbossa, F. ve Persona, A. (2019). "Vertical Lift Modules for Small Items Order Picking: An Economic Evaluation", *International Journal of Production Economics*, 210, 199-210. <https://doi.org/10.1016/j.ijpe.2019.01.012>
- Cano, J.A., Correa-Espinal, A.A., Gómez-Montoya, R.A. ve Cortes, P. (2019). "Genetic Algorithms for the Picker Routing Problem in Multi-Block Warehouses", *International Conference on Business Information Systems*, 313-322. https://doi.org/10.1007/978-3-030-20485-3_24
- Cano, J.A., Cortes, P., Munuzuri, J. ve Correa-Espinal, A. (2024). "Solving the Picker Routing Problem in Multi-Block High-Level Storage Systems Using Metaheuristics", *Flexible Services and Manufacturing Journal*, 35(2), 376-415. <https://doi.org/10.1007/s10696-022-09445-y>
- Caron, F., Marchet, G. ve Perego, A. (1998). "Routing Policies and COI-Based Storage Policies in Picker-to-Part Systems", *International Journal of Production Research*, 36(3), 713-732. <https://doi.org/10.1080/002075498193651>
- Caron, F., Marchet, G. ve Perego, A. (2000). "Optimal Layout in Low-Level Picker-to-Part Systems", *International Journal of Production Research*, 38(1), 101-117. <https://doi.org/10.1080/002075400189608>
- Cergibozan, Ç. ve Tasan, A.S. (2022). "Genetic Algorithm Based Approaches to Solve the Order Batching Problem and A Case Study In A Distribution Center", *Journal of Intelligent Manufacturing*, 33(1), 137-149. <https://doi.org/10.1007/s10845-020-01653-3>
- Cheng, C.Y., Chen, Y.Y., Chen, T.L. ve Yoo, J.J.W. (2015). "Using A Hybrid Approach Based on the Particle Swarm Optimization and Ant Colony Optimization to Solve A Joint Order Batching and Picker Routing Problem", *International Journal of Production Economics*, 170, 805-814. <https://doi.org/10.1016/j.ijpe.2015.03.021>
- Çelik, M. ve Süral, H. (2019). "Order Picking in Parallel-Aisle Warehouses with Multiple Blocks: Complexity and A Graph Theory-Based Heuristic", *International Journal of Production Research*, 57(3), 888-906. <https://doi.org/10.1080/00207543.2018.1489154>
- De Santis, R., Montanari, R., Vignali, G. ve Bottani, E. (2018). "An Adapted Ant Colony Optimization Algorithm for the Minimization of the Travel Distance of Pickers in Manual Warehouses", *European Journal of Operational Research*, 267(1), 120-137. <https://doi.org/10.1016/j.ejor.2017.11.017>
- de Souza, M.C., de Carvalho, C.R. ve Brizon, W.B. (2008). "Packing Items to Feed Assembly Lines", *European Journal of Operational Research*, 184(2), 480-489. <https://doi.org/10.1016/j.ejor.2006.09.091>
- DHaen, R., Ramaekers, K., Moons, S. ve Braekers, K. (2024). "A Real-Life Study on the Value of Integrated Optimization in Order Picking Operations under Dynamic Order Arrivals", *Networks*, 84(3), 345-362. <https://doi.org/10.1002/net.22237>
- Dijkstra, E.W. (1959). "A Note on Two Problems in Connexion with Graphs", *Numerische Mathematik*, (1), 269-271. <https://doi.org/10.1007/BF01386390>
- Dorigo, M. ve Gambardella, L.M. (1997). "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem", *IEEE Transactions on Evolutionary Computation*, 1(1), 53-66. <https://doi.org/10.1109/4235.585892>
- Dorigo, M. ve Stutzle, T. (2004). *Ant Colony Optimization*, Cambridge, MIT Press, MA. <https://doi.org/10.7551/mitpress/1290.001.0001>

- Dorigo, M., Di Caro, G. ve Gambardella L.M. (1999). "Ant Algorithms for Discrete Optimization", *Artificial Life*, 5(2), 137-72. <https://doi.org/10.1162/106454699568728>
- Dorigo, M., Maniezzo, V. ve Colomi, A. (1996). "Ant System: Optimization by A Colony of Cooperating Agents", *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 26(1), 29-41. <https://doi.org/10.1109/3477.484436>
- Dukic, G. ve Oluic, C. (2007). "Order-Picking Methods: Improving Order-Picking Efficiency", *International Journal of Logistics Systems and Management*, 3(4), 451-460. <https://doi.org/10.1504/IJLSM.2007.013214>
- Emde, S. ve Boysen, N. (2012a). "Optimally Locating In-House Logistics Areas to Facilitate JIT-Supply of Mixed-Model Assembly Lines", *International Journal of Production Economics*, 135(1), 393-402. <https://doi.org/10.1016/j.ijpe.2011.07.022>
- Emde, S. ve Boysen, N. (2012b). "Optimally Routing and Scheduling Tow Trains for JIT-Supply of mixed-model Assembly Lines", *European Journal of Operational Research*, 217(2), 287-299. <https://doi.org/10.1016/j.ejor.2011.09.013>
- Floyd, R.W. (1962). "Algorithm 97: Shortest Path", *Communications of the ACM*, 5(6), 345. <https://doi.org/10.1145/367766.368168>
- Gil-Borras, S., Pardo, E.G., Alonso-Ayuso, A. ve Duarte, A. (2021) "A Heuristic Approach for the Online Order Batching Problem with Multiple Pickers", *Computers & Industrial Engineering*, 160, 107517. <https://doi.org/10.1016/j.cie.2021.107517>
- Golz, J., Gujjula, R., Gunther, H.O., Rinderer, S. ve Ziegler, M. (2012). "Part Feeding at High-Variant Mixed-Model Assembly Lines", *Flexible Services and Manufacturing Journal*, 24(2), 119-141. <https://doi.org/10.1007/s10696-011-9116-1>
- Gu, J., Goetschalckx, M. ve McGinnis, L.F. (2007). "Research on Warehouse Operation: A Comprehensive Review", *European Journal of Operational Research*, 177(1), 1-21. <https://doi.org/10.1016/j.ejor.2006.02.025>
- Guo, S., Singh, M. ve Goodarzi, S. (2023). "Enhance Picking Viability in E-commerce Warehouses under Pandemic", *International Journal of Production Research*, 61(15), 5302-5321. <https://doi.org/10.1080/00207543.2022.2101400>
- Hall, R.W. (1993). "Distance Approximation for Routing Manual Pickers in A Warehouse", *IIE Transactions*, 25(4), 76-87. <https://doi.org/10.1080/07408179308964306>
- Koster, R.D. ve Poort, E.V.D. (1998). "Routing Orderpickers in A Warehouse: A Comparison between Optimal and Heuristic Solutions", *IIE Transactions*, 30, 469-480. <https://doi.org/10.1023/A:1007599307171>
- Kübler, P., Glock, C.H. ve Bauernhansl, T. (2020). "A New Iterative Method for Solving the Joint Dynamic Storage Location Assignment, Order Batching and Picker Routing Problem in Manual Picker-To-Parts Warehouses", *Computers & Industrial Engineering*, 147, 106645. <https://doi.org/10.1016/j.cie.2020.106645>
- Lin, C.C., Peng, Y.C. ve Kang, J.R. (2024). "Joint Green Dynamic Order Batching and Picker Routing Problem Using PSO with Global Worst Experience", *Applied Soft Computing*, 154, 111336. <https://doi.org/10.1016/j.asoc.2024.111336>
- Lu, W., McFarlane, D., Giannikas, V. ve Zhang, Q. (2016). "An Algorithm for Dynamic Order-Picking in Warehouse Operations", *European Journal of Operational Research*, 248(1), 107-122. <https://doi.org/10.1016/j.ejor.2015.06.074>
- Masae, M., Glock, C.H. ve Vichitkunakorn, P. (2020). "Optimal Order Picker Routing in A Conventional Warehouse with Two Blocks and Arbitrary Starting and Ending Points of a Tour", *International Journal of Production Research*, 58(17), 5337-5358. <https://doi.org/10.1080/00207543.2020.1724342>
- Masae, M., Glock, C.H. ve Vichitkunakorn, P. (2021). "A Method for Efficiently Routing Order Pickers in the Leaf Warehouse", *International Journal of Production Economics*, 234, 108069. <https://doi.org/10.1016/j.ijpe.2021.108069>
- Ozden, S.G., Smith, A.E. ve Gue, K.R. (2021). "A Computational Software System to Design Order Picking Warehouses", *Computers & Operations Research*, 132, 105311. <https://doi.org/10.1016/j.cor.2021.105311>
- Park, B.C. (2012). "Order Picking: Issues, Systems and Models", *Warehousing in the Global Supply Chain*, Editör: Manzini, R. Springer, London, 1-30. https://doi.org/10.1007/978-1-4471-2274-6_1
- Petersen, C. (1997). "An Evaluation of Order Picking Routing Policies", *International Journal of Operations & Production Management*, 17(11), 1098-1111. <https://doi.org/10.1108/01443579710177860>
- Petersen, C. (1999). "The Impact of Routing and Storage Policies on Warehouse Efficiency", *International Journal of Operations & Production Management*, 19(10), 1053-1064. <https://doi.org/10.1108/01443579910287073>
- Petersen, C. ve Aase, G. (2004). "A Comparison of Picking, Storage, and Routing Policies in Manual Order Picking", *International Journal of Production Economics*, 92(1), 11-19. <https://doi.org/10.1016/j.ijpe.2003.09.006>

- Pugliese, L.D.P., Guerriero, F., Macrina, G., Matteucci, M. ve Mosca, V. (2024). "An Adaptive Large Neighborhood Search for the Order Picking Process: The Case of a Retail Distribution Company in Italy", *Procedia Computer Science*, 232, 2606-2615. <https://doi.org/10.1016/j.procs.2024.02.079>
- Raith, A. ve Ehr Gott, M. (2009). "A Comparison of Solution Strategies for Biobjective Shortest Path Problems", *Computers & Operations Research*, 36(4), 1299-1331. <https://doi.org/10.1016/j.cor.2008.02.002>
- Rao, S.S. ve Adil, G.K. (2013). "Optimal class boundaries, number of aisles, and pick list size for low-level order picking systems", *IIE Transactions*, 45, 1309-1321. <https://doi.org/10.1080/0740817X.2013.772691>
- Rasmi, S.A.B., Wang, Y. ve Charkhgard, H. (2022). "Wave Order Picking under The Mixed-Shelves Storage Strategy: A Solution Method and Advantages", *Computers & Operations Research*, 137, 105556. <https://doi.org/10.1016/j.cor.2021.105556>
- Ratliff, H. ve Rosenthal, A. (1983). "Order-Picking in A Rectangular Warehouse: A Solvable Case of the Traveling Salesman Problem", *Operations Research*, 31(3), 507-521. <https://doi.org/10.1287/opre.31.3.507>
- Roodbergen, K. ve de Koster, R. (2001). "Routing Methods for Warehouses with Multiple Cross Aisles", *International Journal of Production Research*, 39(9), 1865-1883. <https://doi.org/10.1080/00207540110028128>
- Roodbergen, K.J., Sharp, G.P. ve Vis, I.F. (2008). "Designing the Layout Structure of Manual Order Picking Areas in Warehouses", *IIE Transactions*, 40(11), 1032-1045. <https://doi.org/10.1080/07408170802167639>
- Rouwenhorst, B., Reuter, B., Stockrahm, V., van Houtum, G.J., Mantel, R.J. ve Zijm, W.H. (2000). "Warehouse Design and Control: Framework and Literature Review", *European Journal of Operational Research*, 122 (3), 515-533. [https://doi.org/10.1016/S0377-2217\(99\)00020-X](https://doi.org/10.1016/S0377-2217(99)00020-X)
- Rushton A., Croucher P. ve Baker P. (2017). "The Handbook of Logistics and Distribution Management: Understanding the Supply Chain", London, UK, 255-265.
- Scholz, A., Henn, S., Stuhlmann, M., Wascher, G. (2016). "A New Mathematical Programming Formulation for the Single-Picker Routing Problem", *European Journal of Operational Research*, 253, 68-84, 2016. <https://doi.org/10.1016/j.ejor.2016.02.018>
- Silva, A., Coelho, L.C., Darvish, M. ve Renaud, J. (2020). "Integrating Storage Location and Order Picking Problems in Warehouse Planning", *Transportation Research Part E: Logistics and Transportation Review*, 140, 102003. <https://doi.org/10.1016/j.tre.2020.102003>
- Stutzle T. ve Hoos H.H. (2000). "Max-Min Ant System", *Future Generation Computer Systems*, 16(8), 889-914. [https://doi.org/10.1016/S0167-739X\(00\)00043-1](https://doi.org/10.1016/S0167-739X(00)00043-1)
- Teck, S. ve Dewil, R. (2022). "A Bi-Level Memetic Algorithm for the Integrated Order and Vehicle Scheduling in A RMFS", *Applied Soft Computing*, 121, 108770. <https://doi.org/10.1016/j.asoc.2022.108770>
- Tutam, M. ve White, J.A. (2019). "Multi-Dock Unit-Load Warehouse Designs with A Cross-Aisle", *Transportation Research Part E: Logistics and Transportation Review*, 129, 247-262. <https://doi.org/10.1016/j.tre.2019.07.011>
- Vaidyanathan, B.S., Matson, J.O., Miller, D.M. ve Matson, J.E. (1999). "A Capacitated Vehicle Routing Problem for Just-in-Time Delivery", *IIE Transactions*, 31(11), 1083-1092. <https://doi.org/10.1080/07408179908969909>
- Van Gils, T., Caris, A., Ramaekers, K. ve Braekers, K. (2019a). "Formulating and Solving the Integrated Batching, Routing, and Picker Scheduling Problem in A Real-Life Spare Parts Warehouse", *European Journal of Operational Research*, 277(3), 814-830. <https://doi.org/10.1016/j.ejor.2019.03.012>
- Van Gils, T., Caris, A., Ramaekers, K., Braekers, K. ve de Koster, R.B. (2019b). "Designing Efficient Order Picking Systems: The Effect of Real-Life Features on the Relationship among planning problems", *Transportation Research Part E: Logistics and Transportation Review*, 125, 47-73, 2019. <https://doi.org/10.1016/j.tre.2019.02.010>
- Vaughan, T. (1999). "The Effect of Warehouse Cross Aisles on Order Picking Efficiency", *International Journal of Production Research*, 37, 881-897. <https://doi.org/10.1080/002075499191580>
- Warshall, S. (1962). "A Theorem on Boolean Matrices", *Journal of the ACM (JACM)*, 9(1), 11-12. <https://doi.org/10.1145/321105.321107>
- Wu, R., He, J., Li, X. ve Chen, Z. (2024). "A Memetic Algorithm with Fuzzy-Based Population Control for the Joint Order Batching and Picker Routing Problem", *Information Sciences*, 656, 119913. <https://doi.org/10.1016/j.ins.2023.119913>
- Xie, L., Li, H. ve Luttmann, L. (2023). "Formulating and solving integrated order batching and routing in multi-depot AGV-assisted mixed-shelves warehouses", *European Journal of Operational Research*, 307(2), 713-730. <https://doi.org/10.1016/j.ejor.2022.08.047>
- Yousefi Nejad Attari, M., Ebadi Torkayesh, A., Malmir, B. ve Neyshabouri Jami, E. (2021). "Robust Possibilistic Programming for Joint Order Batching and Picker Routing Problem in Warehouse Management", *International Journal of Production Research*, 59(14), 4434-4452. <https://doi.org/10.1080/00207543.2020.1766712>

- Zhang, J., Liu, F., Tang, J. ve Li, Y. (2019). "The Online Integrated Order Picking and Delivery Considering Pickers' Learning Effects for an O2O Community Supermarket", *Transportation Research Part E: Logistics and Transportation Review*, 123, 180-199. <https://doi.org/10.1016/j.tre.2019.01.013>
- Zhao, Z., Cheng, J., Liang, J., Liu, S., Zhou, M. ve Al-Turki, Y. (2024). "Order Picking Optimization in Smart Warehouses with Human-Robot Collaboration", *IEEE Internet of Things Journal*, 11, 16314-16324. <https://doi.org/10.1109/JIOT.2024.3352658/mm1>
- Zhu, S., Hu, X., Huang, K. ve Yuan, Y. (2021). "Optimization of Product Category Allocation in Multiple Warehouses to Minimize Splitting of Online Supermarket Customer Orders", *European Journal of Operational Research*, 290(2), 556-571. <https://doi.org/10.1016/j.ejor.2020.08.024>