



The joint order batching and vehicle routing problem with time window

Esra Boz^{1*}, Nil Aras²

¹Industrial Engineering, Faculty of Engineering, and Natura Science, KTO Karatay University, 42020, Konya, Türkiye

²Industrial Engineering, Faculty of Engineering, Eskişehir Technical University, 26555, Eskişehir, Türkiye

Highlights:

- Order batching problem and vehicle routing problem is considered jointly
- A mixed integer programming formulation is developed for the problem
- Introduce the test data for the problem

Keywords:

- Genetic algorithm
- Order batching
- Time window vehicle routing problem
- Order picking problem

Article Info:

Research Article
Received: 15.07.2022
Accepted: 21.10.2023

DOI:

10.17341/gazimmfd.1144003

Acknowledgement:

This research is supported by the Eskişehir Technical University Scientific Research Project Grant No 20DRP060.

Correspondence:

Author: Esra Boz
e-mail:
esrayasaarr@gmail.com
phone: +90 221 444 1 251 / 7677

Graphical/Tabular Abstract

This study investigates the OB VRP TW, or Joint Order Batching and Vehicle Routing Problem with Time Window. A novel mixed integer nonlinear mathematical model is developed, and in some circumstances the model linearizes. Since there is no test data linked to this problem in the literature, this study integrated real data and test data linked to the vehicle routing problem with a time window. To tackle the issue, the Genetic Algorithm (GA) is suggested, and the algorithm with the First Come First Served (FCFS) rule is compared with the GA. The objective function according to the number of customers is shown in Figure A. The GA provides a better solution than the algorithm that uses the FCFS rule to tackle the problem.

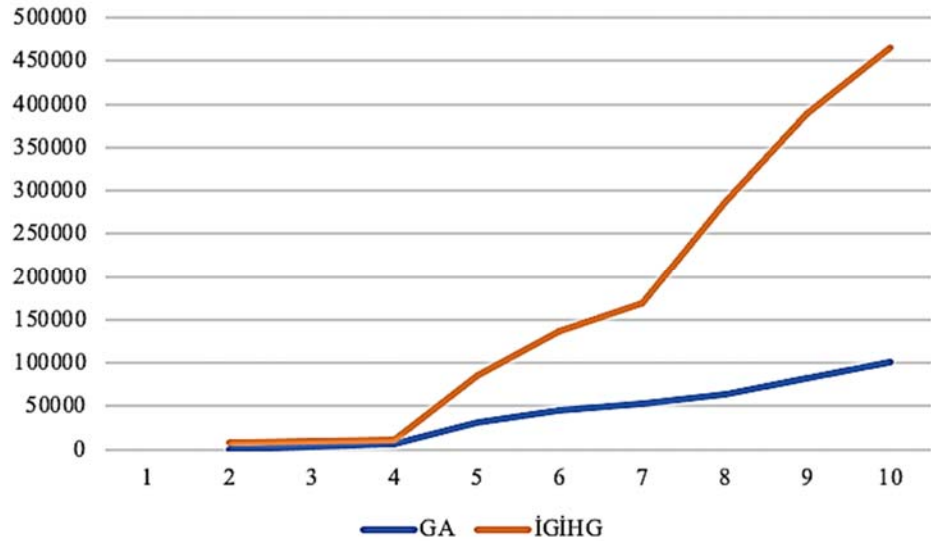


Figure A. The value of the objective function

Purpose:

This study aims to introduce the OB VRP TW test data, develop the new mixed integer linear programming model, and propose a solution algorithm.

Theory and Methods:

To handle the problem, a new mixed integer non-linear programming model is developed. The model is then linearized and solved using GA and FCFS.

Results:

In this study, test data are generated, and experimental design is conducted following the data. The results demonstrated that the GA provides a better solution than the algorithm that uses the FCFS rule to solve it.

Conclusion:

Because OB_VRP_TW is NP-Hard, it cannot be solved using the exact solution method in a reasonable time. As a result, the GA is proposed, and the method is suitable for the OB VRP TW. Using the test data from this study, other metaheuristic algorithms can be developed for future research to compare the algorithm.



Bütünleşik sipariş gruplama ve zaman pencereci araç rotalama problemi

Esra Boz^{1*}, Nil Aras²

¹KTO Karatay Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Endüstri Mühendisliği Bölümü, 42020, Konya, Türkiye

²Eskişehir Teknik Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü, 26555, Eskişehir, Türkiye

Ö N E Ç İ K A N L A R

- Sipariş gruplama ve zaman pencereci araç rotalama problemleri bütünleşik olarak ele alınmıştır
- Bütünleşik sipariş gruplama ve zaman pencereci araç rotalama problemi için yeni bir matematiksel model geliştirilmiştir
- Oluşturulan doğrusal olmayan matematiksel model doğrusallaştırılmıştır

Makale Bilgileri

Araştırma Makalesi

Geliş: 15.07.2022

Kabul: 21.10.2023

DOI:

10.17341/gazimmfd.1144003

Anahtar Kelimeler:

Genetik algoritma,
sipariş gruplama,
zaman pencereci araç
rotalama problemi,
sipariş toplama problemi

Ö Z

Günümüzde şirketler için en önemli unsurlardan biri, siparişleri müşterilere talep edilen doğrultuda ulaştırmaktır. Müşterilerin sipariş etmiş olduğu ürünler müşterilere zamanında ve doğru kalitede ulaştırılmadığı takdirde hem müşteri kaybı hem de maliyete neden olmaktadır. Bu yüzden, bu zorlu sürecin verimli bir şekilde yönetilmesi oldukça önemlidir ve doğru bir yönetim için depo içi ve depo dışı süreçlerin eş zamanlı olarak yürütülmesi gerekmektedir. Bu süreçler depo içinde sipariş toplama sürecinin bir alt dalı olan sipariş gruplama, depo dışında ise bir dağıtım problemi olan araç rotalama süreci olmaktadır. Bu iki süreç entegre olarak düşünülürse, bütünleşik Sipariş Gruplama ve Araç Rotalama Problemi oluşmaktadır. İlgili probleme zaman faktörü eklenmesi ile problem bütünleşik Sipariş Gruplama ve Zaman Pencereci Araç Rotalama Problemi (SG_ZP_ARP)'ne dönüşmektedir. Bu çalışmada, SG_ZP_ARP için literatürdeki çalışmalar sınıflandırılarak incelenmiş ve yeni bir karma tamsayılı doğrusal olmayan programlama modeli geliştirilmiştir. Geliştirilen bu model, belirli koşullar altında doğrusallaştırılmış ve paket çözücü programlar ile yalnızca küçük boyutlu veri setinin bir kısmı çözülebildiğinden dolayı, problemin çözümü için bir metasezgisel arama algoritması olan Genetik Algoritma (GA) yaklaşımı önerilmiştir. İlgili alanda mevcut problem için test verileri olmadığından dolayı alt problemlerin test verileri araştırılmış ve ilgili literatürdeki test verileri ile problemleri problemin uygulaması yapılan bir firmanın bazı verileri entegre edilerek problem için test verileri literatüre eklenmiştir.

The joint order batching and vehicle routing problem with time window

H I G H L I G H T S

- Order batching and time window vehicle routing problems are addressed in an integrated manner
- A new mathematical model has been developed for the integrated order batching and time window vehicle routing problem
- The developed nonlinear mathematical model has been linearized

Article Info

Research Article

Received: 15.07.2022

Accepted: 21.10.2023

DOI:

10.17341/gazimmfd.1144003

Keywords:

Genetic algorithm,
order batching,
time window vehicle routing
problem,
order picking problem

ABSTRACT

During the Covid-19 pandemic, online shopping spread widely, and its quality is later improved and made permanent. Before it is enabled, online shopping is a very expensive and challenging process for businesses to manage, especially since customers can set a time window for their products. Customers may lose business and incur expenditures if the products they requested are not timely and of the appropriate quality supplied to them. Due to this, it is crucial to handle this difficult process well, and for this to happen, both in- and out-of-warehouse procedures must be carried out simultaneously. Order batching, a branch of the order picking process inside the warehouse, and vehicle routing, a distribution problem outside the warehouse, are these processes. The Combined Order Batching and Vehicle Routing Problem arises if these two procedures are considered as being integrated. The linked problem becomes the Integrated Order Batching and Vehicle Routing Problem with Time Window (OB_VRP_TW) when the time component is added. This study categorized and analyzed papers in the literature for OB_VRP_TW and created a novel mixed integer nonlinear programming model. Since just a part of the small data set can be handled by packet encoding programs and this developed model is linearized under specific circumstances, a metaheuristic search algorithm called the genetic algorithm is offered to solve that problem. The test data of the sub-problems were searched for, and the test data for the problem were added to the literature by combining the test data in the literature and some data of a company whose problems were applied, as there were no test data for the current problem in the relevant area.

1. Giriş (Introduction)

Günümüzde müşteriler, sipariş edilen ürünlere mümkün olan en kısa sürede ve en düşük maliyetle ulaşmak istemektedir. Özellikle Covid-19 pandemisi ve sonrasında müşteriler, online olarak sipariş vermekte ve siparişe ulaşmak istedikleri zamanı kendileri seçmektedir. Bu nedenle, müşteri kaybetmemek için firmaların, siparişleri zamanında ve uygun maliyetle teslim etmeleri gerekmektedir. Bu durumda firmaların hem depo içinde hem de depo dışında çözmesi gereken iki sorun ortaya çıkmaktadır. Birincisi müşterilerden gelen siparişlerin hazırlanması ile ilgili Sipariş Toplama Problemi (STP), ikincisi ise siparişlerin müşterilere dağıtılması ile ilgili Araç Rotalama Problemidir (ARP) [1]. Süreçler ardışık olduğu ve birbirini etkilediği için araç rotalama sürecinde bir faktör olan zaman penceresinin, sipariş toplama sürecinde de düşünülmesi gerekmektedir. Bu nedenle her iki sorunu bir bütün olarak değerlendirmek verimliliğin artırılması için önem arz etmektedir.

Depo fonksiyonlarından biri olan sipariş toplama faaliyeti, depoda müşteri istekleri doğrultusunda siparişin hazırlanması ile ilgilidir [2]. STP ise belirli amaçlar doğrultusunda, sipariş toplama faaliyeti içerisindeki verimliliği arttırmayı hedefleyen bir problemdir. STP, belirli durumlarda kullanılabilen Sipariş Gruplama Problemi (SGP) ve Toplayıcı Rotalama Problemlerinin birlikte kullanılması ile oluşmaktadır. Sipariş boyutunun düşük ve sipariş sıklığının yüksek olduğu durumlarda siparişleri gruplayarak toplamak, sipariş toplayıcının depo içerisinde daha az mesafe ve daha az süre ile siparişleri toplamasını sağlamaktadır. Böylelikle SGP, verimliliği en üst düzeye çıkarmak için müşterilerden gelen siparişleri gruplayan bir problemdir. Bu problemde siparişlerin içerikleri farklı olabilmektedir, bu yüzden siparişler farklı miktar ve hacime sahiptir. Sipariş toplayıcılar, depodaki depolama yerlerinden siparişleri toplayan kapasiteli araçlardır. SGP, bu araçlar ile toplanacak olan ürünlerin hangilerinin gruplanarak birlikte toplanması gerektiğine karar vermeye çalışmaktadır. SGP, literatürde kesin yöntemler, sezgisel yöntemler, veri madenciliği ve simülasyon yöntemleri ile çözülmektedir (detaylı bilgi için:[3, 4]). Gruplama, Vinod [5] ile literatüre tanıtılan “Sipariş Gruplama Problemi” ise Gademan vd. [6] ile literatüre eklenmiştir. Gademann ve Velde [7] herhangi bir grubun maksimum teslim süresini en aza indirmeyi amaçlayan SGP için dal ve sınır algoritması geliştirmiştir. Bu çalışma ile birlikte SGP çalışmaları literatürde hızlanmıştır. Hsu vd. [8], toplam mesafeyi en aza indirmeyi amaçlayan SGP’yi çözmek için bir GA yaklaşımı önermiştir. Chen [9], siparişler arası yakınlığı artırmayı amaçlayan SGP için bir veri madenciliği yaklaşımı geliştirmiştir. Henn vd. [10], tüm turların toplam mesafesini en aza indirmeyi amaçlayan SGP’nin çözümü için yinelemeli yerel arama ve karınca kolonileri optimizasyon algoritmalarını sunmuştur. Hsien vd. [11], amacı toplanan grup sayısını azaltmak ve işgücünü dengelemek olan SGP’nin çözümü için bir GA yaklaşımı önermiştir. Nicolas vd. [12], SGP’nin çözümü için, amacı siparişlerin toplanmasında gerekli olan süreyi enküçükleme olan bir tavlama benzetimi yaklaşımı geliştirmiş ve iki farklı firmanın verileri üzerinde problemi, önerilen algoritma aracılığı ile çözümlenmiştir. Pei vd. [13], SGP’nin çözümü için değişken komşuluk arama algoritması geliştirmiştir. Gil-Borras vd.

[14], çevrimiçi SGP için toplama süresini ve toplayıcılar arasındaki iş yükü dengesindeki eşitsizlikleri en aza indiren değişken komşuluk arama algoritması önermiştir.

Bir diğer problem, Araç Rotalama Problemine zaman penceresi eklenmesi ile oluşan Zaman Pencere Araç Rotalama Problemidir (ZP_ARP). Bu problemin amacı, siparişleri müşterilerin istediği zamanda ve en düşük maliyetle müşterilere teslim etmektir. Özellikle tam zamanında üretim kullanılan şirketler için zaman penceresi, genellikle oldukça dar ve kısadır. Bu şirketler, örneğin, bir saatten daha kısa sürede teslimat talep edebilmektedir. Bu yüzden bu problemin uygun bir şekilde çözülmesi önem arz etmektedir. Literatürde ZP_ARP için oldukça fazla çalışma bulunmaktadır. Bunlardan bazıları izleyen şekildedir: Yassen vd. [15], ZP_ARP’yi çözmek için arama algoritmasını; tepe tırmanma, benzetilmiş tavlama, kayıttan kayda, reaktif tabu arama ve büyük tufan algoritmaları ile birleştiren yeni bir hibrit metasezgisel algoritma tanıtmıştır. Song vd. [16], her müşterinin bir zaman penceresine sahip olduğu ve problemin çeşitli araç tiplerini ve enerji tüketimini içerdiği ZP_ARP için bir yapay balık sürüsü algoritması sunmuştur. Marinakas vd. [17], ZP_ARP için yeni çok-uyarlamalı parçacık sürüsü optimizasyon algoritması geliştirmiştir. Liberatore vd. [18], ilgili problem için sütun üretme yöntemi önermiştir.

Bütünleşik problem, SGP ve ZP_ARP’nin entegre olarak düşünülmesi ile oluşturulmaktadır. Bu problem günümüzde güncel ve canlı bir konudur. Bütünleşik problemi literatürde ilk kez ele alan çalışma, Tablo 1’de görüldüğü üzere Schmid vd. [19] tarafından yapılmıştır. Bu çalışmada, SG_ZP_ARP için bir yeni tamsayı doğrusal matematiksel model geliştirilmiştir. Yazarlar, bir depo faaliyeti olan SGP’yi bir ARP olarak düşünmüş, ayrıca bu problemi ARP ile bütünleştirmiştir. Bu çalışmada önerilen matematiksel model, maliyetleri en aza indirmeyi amaçlamaktadır. Literatürdeki ikinci çalışma ise Kuhn vd. [20] tarafından yapılmıştır. Bu çalışmada, problemin çözümü için, bir tamsayı doğrusal matematiksel model geliştirilmiş ve uyarlanabilir geniş komşuluk arama algoritması ile çözümlenmiştir.

Bu çalışma izleyen araştırma soruları üzerine oluşturulmuştur: Bir şirkete ulaşan siparişler zaman penceresine sahipse ve siparişler küçük boyutlu ise, aynı zamanda siparişlerin geliş sıklığı yüksek ise; bu şirket siparişleri zamanında ve minimum maliyetle teslim etmeyi nasıl başarabilir? Bu sorunun cevabı literatürde bazı çalışmalarda mevcuttur [19, 20]. Bu çalışmanın amacı da diğer çalışmalar gibi bu soruya cevap bulmaktır, ayrıca diğer çalışmalardan bazı farkları vardır. Birincisi, bu çalışmada siparişleri oluşturan parçalar arasındaki mesafeler önceden bilinmemektedir ve çözüm sırasında hesaplanmaktadır. Sonuç olarak, uzaklık bilgisi hakkında bir varsayım yoktur. Bu durum, ürünlerin yerlerinin değiştirilmesi noktasında problemin uygulanabilirliğini artırmaktadır. İkincisi, sipariş toplayıcı rotalama sezgisellerinden olan geri dönüş sezgisel yöntemi, çözüm süreci boyunca problem içindeki mesafeleri belirlemek için kullanılmaktadır. Üçüncüsü, bu çalışmanın verileri, literatürde test verileri olarak kullanılabilir düzeydedir. Bu durum, sonraki araştırmacıların çalışmalarını bu veriler üzerinden

Tablo 1 İlgili Çalışmalar (Related literature)

| | Çözüm yöntemi | Mesafelerin bilinirliği varsayımı | Grupların önceden oluşturulması | Parçalar üzerinden işlem | Gerçek veri |
|------------|--|-----------------------------------|---------------------------------|--------------------------|-------------|
| [19] | - | √ | √ | - | - |
| [20] | Uyarlanabilir geniş komşuluk arama algoritması | √ | √ | - | - |
| This study | GA | - | - | √ | √ |

oluşturabileceği ve sonuçları da bu çalışma ile kıyaslayabilecekleri anlamına gelmektedir. Bu çalışma, SG_ZP_ARP için çözüm esnasında grup oluşturma, parçalar üzerinde işlemler gerçekleştirme ve depo içi mesafeyi hesaplama prosedürlerini birlikte işleyen ilk çalışmadır. Aynı zamanda bu çalışma, bütünlük SG_ZP_ARP'yi çözmek için GA kullanan ilk çalışmadır. Dahası, bu çalışmanın sonuçları, literatürde yaygın kullanılan diğer sipariş toplayıcı rotalama sezgiselleri ile kıyaslanmış, çalışmada uygulaması yapılan depo tipine göre ve bu çalışmada verilen veriler ışığında en iyi rotalama sezgiseli belirlenmiştir.

Bu çalışma izleyen şekilde devam etmektedir: İkinci bölümde SG_ZP_ARP ile ilgili bilgiler verilmiş, bu problem için oluşturulan matematiksel model gösterilmiş ve bu modelin doğrulasştırılmasına yer verilmiştir. Üçüncü bölümde, problemin çözümü için önerilen GA ifade edilmiştir. Dördüncü bölüm olan sayısal analiz kısmında, çalışmada uygulaması yapılan depo altyapısı, kullanılan veriler, rotalama sezgiseli, analiz sonuçları ve duyarlılık analizi aktarılmıştır. Beşinci bölümde, yönetimsel çıkarımlar ve sınırlılıklar ifade edilmiştir. Son bölüm olan altıncı bölümde ise çalışmanın sonuçları ve önerilere yer verilmiştir.

2. Problem Tanımı ve Matematiksel Model (Description of the Problem and Mathematical Model)

Bu çalışmada incelenen problem bir gerçek hayat problemidir. Uygulanan yapıldığı şirketin bazı müşterileri küçük ebatlı siparişler vermek istemektedir. Ancak firma siparişleri gruplayarak toplamadığı ve dağıtmadığı için küçük boyutlu siparişler almak istememektedir. Ancak müşteri kaybetmemek için bu süreci başlatması ve verimli bir şekilde yönetmesi gerekmektedir. İlgili süreç başladığı takdirde, müşteriler ürünleri teslim alacakları zamanı belirleyecek ve bir paletten daha az ürün sipariş edeceklerdir. Bu yüzden, şirketin müşteri memnuniyetini sağlarken zaman ve maliyeti dengelemesi gerekmektedir. Böylelikle izleyen soru ortaya çıkmaktadır: Müşterileri siparişleri, müşterilere istenilen zamanda ve en düşük maliyetle nasıl ulaştırılabilir?

Müşteriler, şirkete siparişlerinin teslim edilmesini istedikleri belirli bir zaman aralığı vermektedir. Öyleyse; zaman penceresi, müşterilere ürün sunarken dikkate alınması gereken önemli bir faktördür. Zaman penceresi müşteriler tarafından oluşturulmaktadır. Ürünlerin müşterilere teslimatı gecikirse, yani müşteri tarafından belirlenen zaman aralığından sonra müşteri ziyaret edilirse, belirli bir gecikme maliyeti ödenmelidir. Sonuç olarak, müşteriye ulaşırken rotalama maliyetlerini en aza indirirken zaman verimliliğini en üst düzeye çıkarmak önemlidir. Öte yandan siparişlerinin toplanmasında dikkat edilmesi gereken husus, ürünlerin depolandıkları yerlerden en kısa sürede teslim alınmasıdır. Bu toplama süresi ve toplama mesafesinin en aza indirilmesi, siparişlerin müşterilere zamanında teslim edilmesini ve sipariş toplayıcıların daha verimli çalışmasını sağlamaktadır. Böylece hem gecikme maliyeti azalmakta hem de sipariş toplayıcı aynı sürede daha fazla siparişi toplayabilmektedir. Firmaya gelen siparişlerin miktarları az ve sipariş sıklıkları oldukça fazladır. Sonuç olarak, siparişlerin ayrı ayrı toplanması depo içerisindeki sipariş toplayıcının hem yürüme mesafesini hem de toplama süresini artırmaktadır. Bu yüzden, siparişler gruplanarak toplanmalı ve böylece ilgili maliyetler azaltılmalıdır.

Şirketin bu iki problemi (SGP ve ZP_ARP) aynı anda oluşturması ve ele alması dolayısı ile bir bütün olarak bakması gerekmektedir. Depoda harcanan süre, depo dışı sürece eklendiğinden dolayı, depo içi süreç depo dışı süreci etkilemektedir. Eğer ilgili problemler birlikte incelenir ve çözümlerse, maliyetler daha etkin bir şekilde düşürülebilmektedir.

Firmanın sipariş sistemlerinden siparişler depoya ulaşır ulaşmaz ilgili süreç başlamaktadır. Bütünlük süreçte siparişler gruplanarak toplanmakta, sevkiyat noktasında araçlara yüklenmekte ve müşterilere teslim edilmektedir. Bütünlük problemin çözümü ile izleyen sorulara cevap bulunması amaçlanmaktadır:

- Hangi siparişler hangi gruplara atanacak?
- Grupların toplanma sırası nedir?
- Toplanan siparişler hangi araçlara yerleştirilecek?
- Siparişlerin müşterilere teslim edilme sırası nedir?
- Araçlar müşterilere ne zaman teslim edilecek ve bekleme süresi ne kadar olacak?

Çalışma içerisinde, belirlenen varsayımlar altında yeni bir karma tamsayı doğrusal olmayan programlama modeli geliştirilmiştir. Bu varsayımlar genel anlamda izleyen şekilde ifade edilmiştir:

- Her müşteri yalnızca bir sipariş verebilmektedir.
- Siparişlerin içerisindeki ürünler ayrılamaz; her ürünün aynı grupta olması gerekmektedir.
- Bir siparişte birden fazla ürün olabilmektedir.
- Ürünlerin stoklandığı yerler sabittir ve önceden bilinmektedir.
- Sipariş toplayıcının hızı sabittir ve önceden bilinmektedir.
- Dağıtım aracının hızı sabittir ve önceden bilinmektedir.
- Sipariş toplayıcı, siparişleri her iki yönlü olarak toplayabilmektedir.
- Trafik sıkışıklığı gibi faktörler göz ardı edilmiş, aracın yolda geçirdiği süre, mesafe ve hıza göre hesaplanmıştır.

Ayrıca bu model [19–21] tarafından önerilen modellere eklemeler ve değişiklikler yapılarak geliştirilmiştir. Matematiksel modelin notasyonu Tablo 2'de belirtildiği üzere oluşturulmuştur.

$$\sum_j \sum_v x_{ijv} = 1 \quad \forall i \quad (1)$$

$$\sum_i x_{ijv} - \sum_j x_{jiv} = 0 \quad \forall i, j, v \quad (2)$$

$$\sum_i x_{0iv} = 1 \quad \forall v \quad (3)$$

$$\sum_i x_{i(N+1)v} = 1 \quad \forall v \quad (4)$$

$$\sum_j \sum_i w_i x_{ijv} \leq Q \quad \forall v \quad (5)$$

$$Y_j \geq Y_i + 1 - N(-\sum_v x_{ijv}) \quad \forall i, j \quad (6)$$

$$Y_j \geq 0 \quad \forall j \quad (7)$$

$$a_i \geq d_v + t_{0i} - M(-u_{iv}) \quad \forall i, v \quad (8)$$

$$a_j \geq a_i + t_{ij} + s_i - M(1 - x_{ijv}) \quad \forall i, j, v \quad (9)$$

$$\sum_j x_{ijv} = u_{iv} \quad \forall i, v \quad (10)$$

$$t_i \geq (a_i - \beta_i) \quad \forall i \quad (11)$$

$$a_i \geq \alpha_i \quad \forall i \quad (12)$$

$$\sum_i w_i z_{ib} \leq q \quad \forall b \quad (13)$$

$$\sum_b z_{ib} = 1 \quad \forall i \quad (14)$$

$$u_{iv} = x_{bv} z_{ib} \quad \forall i, v, b \quad (15)$$

$$t_b = (2h_b + \sum_v 2s_{bv})/v_{top} + \sum_h y_{bh} s_h \quad \forall b \quad (16)$$

Tablo 2. Parametrelere ait notasyon (Notation of the parameters)

| Parametre | Açıklama |
|------------------------|--|
| d_{ij} | i müşterisi ile j müşterisi arasındaki mesafe büyüklüğü (km) |
| Q | Dağıtım aracının kapasitesi (kg) |
| q | Sipariş toplayıcı kapasitesi (kg) |
| w_i | i siparişinin ağırlığı (kg) |
| t_{ij} | i müşterisi ile j müşterisi arasındaki mesafe süresi (sa) |
| s_i | i. müşteri siparişlerinin araçtan indirilme süresi (sa) |
| $(\alpha_i - \beta_i)$ | i. müşterinin zaman penceresi. α_i erken gitme süresi, β_i geç gitme süresi |
| s_h | Toplayıcının h ürününü raftan alma süresi (sa) |
| t_{load} | Depoda toplayıcılardan alınan ürünlerin araca yerleştirilme süresi (sa) |
| r_{ik} | i. sipariş için k koridorunda alınan en yüksek yatay mesafe büyüklüğü (km) |
| w | Koridorlar arasındaki yatay mesafe (km) |
| v_{top} | Sipariş toplayıcı hızı (km/sa) |
| c_{ih} | i siparişinin içerisinde h ürünü bulunuyorsa 1, dd 0 |
| v | Araç numarası $v=1,2,\dots,V$ |
| h,m | Ürün numarası $h,m=1,2,\dots,H$ |
| i,j | Müşteri (sipariş) numarası $i,j=1,2,\dots,I$ |
| b | Grup numarası $b=1,2,\dots,B$ |
| k | Koridor numarası $k=1,2,\dots,K$ |
| c_1 | Birinci bileşene ait amaç fonksiyonu maliyet katsayısı |
| c_2 | İkinci bileşene ait amaç fonksiyonu maliyet katsayısı |
| c_3 | Üçüncü bileşene ait amaç fonksiyonu maliyet katsayısı |
| Karar değişkenleri | Açıklama |
| h_b | b. sipariş grubunun toplanması sırasında alınan yatay mesafe büyüklüğü (km) |
| s_{bk} | b. sipariş grubu toplanırken, k koridor boyunca alınan en büyük mesafe büyüklüğü (km) |
| y_{bk} | b grubunun toplanması için k koridoru ziyaret ediliyorsa 1, dd 0 |
| x_{ijv} | v aracı ile i müşterisinden sonra j müşterisine gidiliyorsa 1, dd 0 |
| t_i | i müşteri siparişinin gecikme süresi (tardiness) |
| u_{iv} | i. müşteri siparişi v aracına atanırsa 1, dd 0 |
| y_{hb} | h ürünü b grubuna atanırsa 1, dd 0 |
| a_i | i. müşteri siparişinin teslim edilme süresi |
| z_{ib} | i siparişi b grubuna atanırsa 1, dd 0 |
| d_v | v. dağıtım aracının depodan kalkış zamanı |
| y_i | i. siparişe ait alt tur engelleme için kullanılan karar değişkeni |

$$d_v = \sum_b (t_b x_{bv}) + t_{load} \quad \forall v \quad (17)$$

$$y_{bk} \leq \sum_i r_{ik} x_{ib} \leq M y_{bk} \quad \forall b, k \quad (18)$$

$$k w y_{bk} \leq h_b \quad \forall b, k \quad (19)$$

$$r_{ik} x_{ib} \leq s_{bk} \quad \forall i, b, k \quad (20)$$

$$\sum_v x_{bv} = 1 \quad \forall b \quad (21)$$

$$y_{hb} \geq z_{ib} c_{hi} \quad \forall h, i, b \quad (22)$$

$$t_i \geq 0 \quad \forall i \quad (23)$$

$$y_{hb}, z_{ib}, y_{bk}, x_{ijv}, u_{iv} \in \{0, 1\} \quad \forall i, j, k, h, b \quad (24)$$

kısıtları altında

En küçük Z=

$$c_1 (\sum_b 2h_b + \sum_b \sum_k 2s_{bk}) + c_2 \sum_i \sum_j \sum_v d_{ij} x_{ijv} + c_3 \sum_i t_i \quad (25)$$

(Eş. 25) numaralı denklem, amaç fonksiyonunu temsil etmektedir. Bu fonksiyon içerisinde üç bileşen vardır. Birinci bileşen, depo içerisindeki sipariş toplayıcının siparişleri toplarken almış olduğu toplam mesafeyi temsil etmektedir. Alt problem düşünülecek olursa; bu mesafe sipariş toplama problemi içerisinde hesaplanmaktadır. İkinci bileşen, depo dışında ürünlerin müşterilere ulaştırılması için

alınan toplam mesafe hesaplamasının göstermektedir. Bu bileşen için de alt problem düşünülürse, müşteriler arasındaki toplam mesafe araç rotalama problemi içerisinde hesaplanmaktadır. Üçüncü bileşen ise müşterilerin toplam gecikmesini ifade etmektedir. Bu bileşen araç rotalama ve sipariş toplama problemlerinin kesişim noktası olarak ifade edilebilmektedir.

(Eş. 1) numaralı eşitlik her müşteriye yalnızca bir araç uğramasını sağlamaktadır. Böylelikle her müşteri siparişinin teslimatı için müşterilerin tamamına gidilmiş olmaktadır. (Eş. 2) numaralı kısıt araç rotalama problemi literatüründe akışı sağlayan kısıt olarak geçmektedir. Bir aracın bir müşteriye başka bir müşteriden (depo dahil) gelmesini ve bir aracın bir müşteriden başka bir müşteriye (depo dahil ya da toplanma noktası) gitmesini sağlamaktadır. (Eş. 3) numaralı kısıt her aracın rotasına depodan başlamak zorunda olduğunu ifade etmektedir. (Eş. 4) numaralı kısıt ise her aracın rotasının toplanma noktasında sonlanması gerekliliğini belirtmektedir.

(Eş. 5) numaralı kısıt, araç kapasite kısıtıdır. Her aracın belirli bir kapasitesi vardır ve müşterileri ziyaret eden araçlar bu kapasiteyi geçmemektedir. Dolayısı ile müşterilere teslim edilecek olan siparişlerin ağırlıkları toplamı aracın kapasitesini geçmemektedir. (Eş. 6- Eş. 7) numaralı kısıtlar, araç rotalama problemi literatüründe alt tur eleme kısıtları olarak geçen kısıtlardır. (8) numaralı kısıtta, her araç için depodan ayrıldıktan sonraki ilk müşteriye gidiş süresine ulaşılmaya çalışılmaktadır. (Eş. 9) numaralı kısıt, bir önceki kısıt gibi müşterilere ulaşılma süresini hesaplayan bir formülasyonu ifade

etmektedir. Bu iki kısıtın tek farkı, depodan çıkan aracın servis süresi olmayacağı için birinci müşteriden sonrasına ulaşılmaya çalışılmaktadır. (Eş. 10) numaralı kısıt ile iki karar değişkeni birbirine bağlanmış olmaktadır.

(Eş. 11- Eş. 12) numaralı eşitsizlikler ile gecikme süresi, müşteriye ulaşılan zaman ile müşterinin istediği zaman penceresi arasındaki fark ile hesaplanmaktadır. (Eş. 13) numaralı kısıt sipariş toplayıcı kapasite kısıtıdır. Her bir sipariş toplayıcının belirli bir kapasitesi vardır ve toplayıcılara bu kapasiteden fazla ürün ya da sipariş eklenememektedir. (Eş. 14) numaralı kısıt müşteri siparişlerinin ürünlerinin her birinin yalnızca bir gruba atanmasını sağlar. (Eş. 15) numaralı kısıt, bir müşteri siparişindeki bütün ürünlerin birlikte toplanması sağlanmaktadır. (Eş. 14) numaralı kısıt ile bir bütün halindedir ve böylelikle, iki kısıt birlikte bir siparişteki ürünlerin birlikte toplanmasını sağlar. (Eş. 16) numaralı kısıt ile bir grubun toplanma süresi hesaplanmaktadır.

(Eş. 17) numaralı kısıt, aracın yükleme noktasında geçirdiği süre hesaplamasını ifade etmektedir. Aynı zamanda, bu kısıt sipariş gruplama problemi ve araç rotalama problemini birbirine bağlayan kısıttır. (Eş. 18) numaralı kısıt ile, bir siparişte herhangi bir koridorda en az bir ürün olduğunda, o siparişin atandığı grubun toplanma esnasında o koridora girilmesi sağlanmaktadır. (Eş. 19) numaralı kısıt ile, bir grup için yatay mesafede alınan yol hesaplanmaktadır. (Eş. 20) numaralı kısıt ile, bir grubun toplanması için en fazla gerekli olan dikey mesafe gösterilmektedir. (21) numaralı kısıt, her grubun yalnızca bir araca atanmasını sağlamaktadır. (Eş. 22) numaralı kısıt ile karar değişkenleri eşleştirilmektedir. (Eş. 23-Eş. 24) numaralı kısıtlar işaret kısıtlarıdır.

Doğrusallaştırma:

Verilen matematiksel modelde, (Eş. 15, Eş. 17) numaralı kısıtlar doğrusal olmayan kısıtlar olduğu için, model doğrusal olmayan matematiksel model olmaktadır. Ancak bu kısıtlar doğrusallaştırılabilir düzeydedir ve matematiksel modelin doğrusal olabilmesi için önem taşımaktadırlar.

(Eş. 15) numaralı kısıtta, 0-1 tamsayılı karar değişkenlerinin çarpımından dolayı kısıt doğrusal olmayan bir hale dönüşmektedir. Ancak doğrusallaştırma şartları kapsamında aşağıda ifade edilen maddenin uygulanması ile doğrusallaştırılabilmektedir [22].

$z = x \times y$ ve hem x hem de y tamsayılı karar değişkeni ise, bu ifade aşağıda (Eş. 26- Eş. 28) numaralı denklemler aracılığı ile doğrusallaştırılabilmektedir. Verilen ilk iki eşitsizlik ile x veya y sıfır ise z sıfır olmaktadır. Son eşitsizlikte ise hem x hem de y 1 ise z değeri de 1 olmaktadır.

$$z \leq x \quad (26)$$

$$z \leq y \quad (27)$$

$$z \leq x + y - 1 \quad (28)$$

Böylelikle (Eş. 16) numaralı kısıt (Eş. 29- Eş. 31) numaralı kısıtlara dönüşmektedir:

$$u_{iv} \leq x_{bv} \quad \forall i, v, b \quad (29)$$

$$u_{iv} \leq z_{ib} \quad \forall i, v, b \quad (30)$$

$$u_{iv} \geq x_{bv} + z_{ib} - 1 \quad \forall i, v, b \quad (31)$$

(Eş. 17) numaralı kısıtta; bir değişken 0-1 tamsayılı karar değişkeni, diğer değişken ise sürekli karar değişkeni olduğu için, bu karar değişkenlerin çarpımından dolayı kısıt doğrusal olmayan bir hale dönüşmektedir. Ancak doğrusallaştırma şartları kapsamında aşağıda ifade edilen maddenin uygulanması ile doğrusallaştırılabilmektedir [22].

$z = x \times y$ ve x 0-1 tamsayı karar değişkeni, y sürekli karar değişkeni ise bu ifade (Eş. 32- Eş. 35) numaralı denklemler aracılığı ile doğrusallaştırılabilmektedir. Verilen ilk ve son eşitsizliğe göre; x sıfırsa, z sıfır olmalıdır. Aynı zamanda ikinci ve üçüncü eşitsizliğe göre x 1 ise, z , y 'ye eşit olmalıdır.

$$z \leq H * y \quad (32)$$

$$z \leq x \quad (33)$$

$$z \geq x - (1 - y) * H \quad (34)$$

$$z \geq 0 \quad (35)$$

Böylelikle (Eş. 18) numaralı kısıt (Eş. 36- Eş. 39) numaralı kısıtlara dönüşmektedir:

$$d_v - t_{load} \leq M \sum_b x_{bv} \quad \forall v \quad (36)$$

$$d_v - t_{load} \leq \sum_b t_b \quad \forall v \quad (37)$$

$$d_v - t_{load} \geq \sum_b t_b - (1 - \sum_b x_{bv}) \quad \forall v \quad (38)$$

$$d_v - t_{load} \geq 0 \quad \forall v \quad (39)$$

3. Genetik Algoritma (Genetic Algorithm)

NP-Zor (Non Polynomial-Hard) problemlerin kesin çözüm yöntemleri ile makul bir zamanda optimum çözüme ulaşması çok zor olduğu için metasezgisel yöntemler geliştirilmiştir [23]. Ele alınan SG_ZP_ARP, NP-Zor bir yapıya sahip olduğundan dolayı ve paket çözümler ile yalnızca küçük boyutlu problemlerin bir kısmı çözülebildiğinden dolayı, bu çalışmada problemin çözümü için GA önerilmiştir. Literatüre Holland [24] tarafından tanımlanan GA, evrim gibi biyolojik organizma prosedürüne atıfta bulunan bir arama algoritmasıdır. GA, global bir optimal çözüm üretme yeteneğine sahip olduğu için bu çalışmada kullanılmıştır [25]. Bu algoritma literatürde birçok kez NP-Zor problemleri çözmek için uygulanmıştır (detay için bakınız: [26]). GA; kod gösterimi, başlangıç popülasyonu, uygunluk fonksiyonu, seçim yöntemi olmak üzere farklı bileşenlere sahiptir. Her bileşen problem için çok önemlidir ve bu bölümde bu bileşenler ifade edilmektedir.

3.1. Kod gösterimi (Code Representation)

GA'da kod gösterimi, problemin çözümü için önem arz etmektedir [27]. GA içerisinde öncelikle kod gösterimi belirlenmekte ve ardından diğer adımlar buna göre tamamlanmaktadır.

Literatürde farklı kod gösterimleri önerilmiştir. Bu çalışmada ise Şekil 1'de verilen permütasyon ve rastgele anahtar kodlamasının bir kombinasyonu kullanılmıştır. SG_ZP_ARP, iki aşamalı bir problemdir ve kodlama gösterimi bu aşamaların her ikisini de sağlamak zorundadır. Bu gösterim şeklinde, her sütun bir müşteriyi temsil etmektedir. Araç numarası ikinci satırda yer almaktadır. Müşteri sırası 0 ile 1 arasındaki sayılardan oluşan son satırda belirtilmiştir. Bu sayılar, müşterilerin atandıkları araca göre küçükten büyüğe sıralanarak müşteri ziyaret sırası belirlenmiştir. Şekil 1'de

gösterildiği üzere, birinci araca atanan müşterilerin ziyaret sıralaması 3-1-4-8-9 olarak oluşturulmuştur.

3.2. Başlangıç popülasyonu (Starting Population)

Başlangıç popülasyonunun, algoritmanın en iyi sonuca ne kadar çabuk ulaştığı üzerinde önemli bir etkisi vardır. Algoritma ne kadar hızlı tamamlanırsa, başlangıç popülasyonu o kadar doğru ve hassas bir şekilde oluşturulmaktadır. Bu çalışmada başlangıç popülasyonu birden fazla denemeler yapılacağı için rassal olarak oluşturulmuştur.

3.3. Uygunluk fonksiyonu (Fitness Function)

Bu algoritmanın uygunluk fonksiyonu, SG_ZP_ARP için geliştirilen matematiksel modelin amaç fonksiyonu olarak oluşturulmuştur. Bu fonksiyon, üç bileşenden oluşmakta ve birinci bileşen depo içerisindeki ürünlerin toplanma mesafe maliyeti, ikinci bileşen ürünlerin müşterilere dağıtım mesafe maliyeti, üçüncü bileşen ise ürünler müşterilere ulaştırıldığında ortaya çıkan gecikme maliyetini ifade etmektedir.

3.4. Seçim yöntemi (Selection Method)

Ebeveyn kromozomlarını seçmek için bu adımda eşleşme havuzu oluşturulmaktadır. Kromozomların üretilmesi eşleşme havuzunda gerçekleştiği için bu adım oldukça önemlidir. Bu konudaki literatür, rulet çarkı ve turnuva seçim yöntemleri gibi çeşitli seçim yöntemlerini içermektedir. Bu çalışmada, GA içerisinde yaygın olarak kullanılan bir yöntem olması ve algoritmaya stokastik aramayı dahil etmesi nedeniyle rulet çemberi seçim yöntemi kullanılmıştır [28]. Bu yöntemin ilk aşamasında, kromozomlar uygunluk fonksiyonuna göre sıralanmakta, ikinci adımda, her bir kromozomun seçim olasılığı hesaplanmaktadır. Üçüncü adımda, kümülatif olasılık değerleri bulunmakta, ardından son adım olan dördüncü adımda, rastgele sayılar üretilmekte ve bu sayıların hangi kromozomla eşleştiği bulunmaktadır.

3.5. Yeniden üretim (Regenerate)

Yeniden üretim sırasında eski bir kromozomdan yeni bir kromozom üretilmektedir. Bu çalışmada, yeniden üretim için çaprazlama ve mutasyon işlemleri kullanılmıştır. Çaprazlama, ebeveynlerden yeni bireyler üretmek için kullanılabilen ve GA'nın başansı üzerinde önemli bir etkiye sahip olabilen bir GA üreme operatörüdür [29]. Bu operatörde, eşleştirme havuzundan bir çift kromozom çaprazlama için rastgele seçilmekte ve çaprazlama operatörü bu iki kromozomdan iki yeni oğul birey oluşturmak için kullanılmaktadır. Çaprazlama yönteminin literatürde farklı çeşitleri mevcuttur ancak bu çalışmada yalnızca tek nokta çaprazlama yöntemi kullanılmıştır. Tek nokta çaprazlamada, rastgele bir nokta belirlenmekte ve ebeveyn kromozomlar bu noktadan bölünerek oğul bireyleri oluşturulmaktadır.

Mutasyon, yavru bireylerde bazı genlerin yerini değiştirmek için kullanılmaktadır. Bu işlem, GA'yı çaprazlama ile elde edilmesi mümkün olmayan çözümlere yönlendirdiği için önemli bir faktördür. Literatürde mutasyon operatörünün farklı çeşitleri bulunmaktadır ancak bu çalışmada yalnızca ikili yer değiştirme yöntemi kullanılmıştır. İkili yer değiştirme mutasyon yönteminde kromozom üzerinde bulunan iki gen tanımlanmakta ve bu genler kendi aralarında değiştirilerek yeni bir kromozomun oluşması sağlanmaktadır.

3.6. Ceza Fonksiyonu (Penalty Function)

Uygun olmayan şekilde oluşan kromozomlar için ceza fonksiyonu kullanılarak bu kromozomların uygunluk fonksiyonunun artması sağlanarak, problem bir enküçükleme problemi olduğundan dolayı, sonraki nesillere aktarılması engellenmiş olmaktadır. Bu uygunluk durum kontrolü ise popülasyon oluştuğunda, çaprazlama sonrasında ve mutasyon sonrasında yapılmaktadır. Buna göre, bir kromozom içerisinde hem sipariş toplayıcı kapasite aşımı hem de dağıtım aracı kapasite aşımının olması durumunda, aynı zamanda depo içerisinde herhangi bir siparişin toplanmama ve dağıtım esnasında herhangi bir müşterinin ziyaret edilmemesi durumunda ceza fonksiyonu devreye girmektedir. Bu durumlar süreç içerisinde farklı kısıtları ifade ettiği için kısıtların ceza katsayıları farklı oluşturulmuştur. Böylelikle hesaplanan uygunluk fonksiyonuna ceza fonksiyonu da eklenerek uygunluk fonksiyonu değeri yüksek bir değere ulaşmaktadır. Herhangi bir uygunsuzluğun oluşmadığı durumda ise ceza fonksiyonu sıfır olarak oluşmaktadır.

4. Sayısal Analiz (Numerical Analysis)

Bu bölümde izleyen başlıklarda, SG_ZP_ARP için oluşturulan veriler ve bu verileri kullanarak yapılan deneyler, deneylerin sonuçları ve duyarlılık analizi ifade edilmektedir.

4.1. Test Problemleri (Test Problems)

Bu çalışmada SG_ZP_ARP ele alınmış, ilgili problemin çözümü için yeni karma tamsayılı doğrusal model geliştirilmiştir. Alt problemler (SGP ve ARP) NP-Zor bir yapıya sahip olduğu için SG_ZP_ARP de NP-Zor bir yapıya sahiptir [7,30]. Bu yüzden bu problemin makul bir zamanda en iyi çözüme ulaşması oldukça zordur [23]. Literatürdeki bazı çalışmalar, Sipariş Toplama ve Araç Rotalama Problemlerinin bütünlük olarak düşünülmesi durumunda CPLEX çözücüsü kullanılarak yalnızca küçük boyutlu örneklerin çözülebildiğini göstermişlerdir. Bu çalışmada da aynı şekilde CPLEX çözücüsü ile küçük boyutlu örneklerin bir kısmı çözülebilmıştır.

Veri yapısının oluşturulması için literatürdeki benzer çalışmalar incelenmiş ancak her çalışmada veri yapılarının farklı oluşturulduğu görülmüştür [10,20,31-34]. Bu yüzden bu çalışmada, literatürdeki benzer çalışmalar baz alınarak Tablo 3'deki gibi bir veri yapısı oluşturulmuştur. Buna göre toplayıcı kapasitesi ve araç kapasitesi

| | | | | | | | | | |
|------|------|------|------|-----|------|------|------|------|-----|
| 1 | 2 | 1 | 3 | 4 | 1 | 2 | 3 | 1 | 4 |
| 1 | 2 | 1 | 1 | 2 | 1 | 2 | 1 | 1 | 2 |
| 0,16 | 0,92 | 0,03 | 0,49 | 0,5 | 0,76 | 0,32 | 0,57 | 0,64 | 0,8 |

Şekil 1. GA kod gösterimi (Representation of GA)

Tablo 3. Algoritma parametreleri (The parameters)

| | Sipariş sayısı | Ürün sayısı | Toplayıcı kapasitesi (kg) | Araç kapasitesi (kg) |
|---------------|----------------|-------------|---------------------------|----------------------|
| Küçük boyutlu | 5,10,15 | 30 | 500 | 2500 |
| Orta boyutlu | 50,60,70 | 90 | 500 | 2500 |
| Büyük boyutlu | 100,150,200 | 250 | 500 | 2500 |

sabit olarak alınmakta, sipariş sayısı ve ürün sayısı ise veri boyutuna göre değişkenlik göstermektedir. Veri seti; küçük boyutlu, orta boyutlu ve büyük boyutlu olmak üzere 3 sınıfa ayrılmaktadır.

SG_ZP_ARP, 3. bölümde ifade edildiği üzere bazı parametrelere sahiptir. Problemin çözülebilmesi için belirlenmesi gereken bu parametrelerin bir kısmı SGP'ye ait iken bir kısmı da ZP_ARP'ye ait olmaktadır. İlgili literatürde SG_ZP_ARP için herhangi bir test verisi bulunmamaktadır. Bu yüzden alt problemlere inilerek detaylı bir şekilde arama yapılmıştır. SGP için literatürde bu problem için uygun bir test verisi bulunmazken, ZP_ARP için ilgili verilere Solomon [35] çalışmasında ulaşılmıştır (Bu çalışmada R101 ve R102 veri yapısı kullanılmıştır). SGP ile ilgili olan parametreler ise uygulama yapılan firmadan alınmıştır. Böylelikle hem test verileri hem de gerçek hayat verileri birleştirilerek bütünlük problem için ilgili test verileri oluşturulmuştur.

Problemin uygulandığı deponun parametreleri Tablo 4'de verildiği üzere belirlenmiştir. Toplamda 15 toplama koridoru bulunmaktadır ve toplayıcı, bu toplama koridorlarının hepsini kullanabilmektedir. Ayrıca problemin çözümü için kullanılan diğer veriler izleyen şekildedir:

- Ürünlerin servis süresi ortalama 60 saniyedir.
- Ürünlerin araca yerleştirilme süresi (aracın yüklenme süresi) ortalama 45 dakikadır.
- Sipariş toplayıcının depodaki hızı ortalama 7 km/saat'dir.
- Aracın depodan çıkış hızı ortalama 70 km/saat'dir.
- Müşterinin hizmet süresi ortalama 5 saattir.

Tablo 4. Depo parametreleri (Parameters of the warehouse)

| Parametre | Değer |
|-----------------------|-----------|
| Koridor sayısı | 15 |
| Koridor genişliği | 0,32 (km) |
| Depolama alanı sayısı | 100 |

GA kendi içerisinde; çaprazlama oranı, mutasyon oranı, nesil sayısı, popülasyon büyüklüğü başta olmak üzere bazı parametrelere sahiptir. Bu parametreler, algoritma özelinde değişkenlik gösterebilmekte ve algoritma sonucunu etkileyebilen özelliklere sahip olmaktadır. Bu yüzden; bu parametre değerlerinin seçim yöntemi, deneylerin sonuçlarını önemli ölçüde değiştirebilmektedir. Bu çalışmada, GA parametrelerinin belirlenebilmesi için Taguchi metodu kullanılmıştır. Algoritma parametreleri tasarımında, 5 faktör ve bu faktörlere ait 3 seviye bulunmaktadır. Taguchi metodu; küçük, orta ve büyük boyutlu örnekler için ayrı ayrı oluşturulmuştur. Buna göre; her veri seti için sipariş sayısı faktörü değişkenlik göstermektedir. Bu yüzden, 5 faktör ve 3 seviye bulunduğu için L27 Ortogonal dizisi kullanılmıştır. Böylelikle her veri seti için 27 deney yapılmıştır. Tablo 5'de faktörler ve seviye parametreleri ifade edilmektedir.

Tablo 5. Faktörler ve seviye parametreleri (The factors and the level parameters)

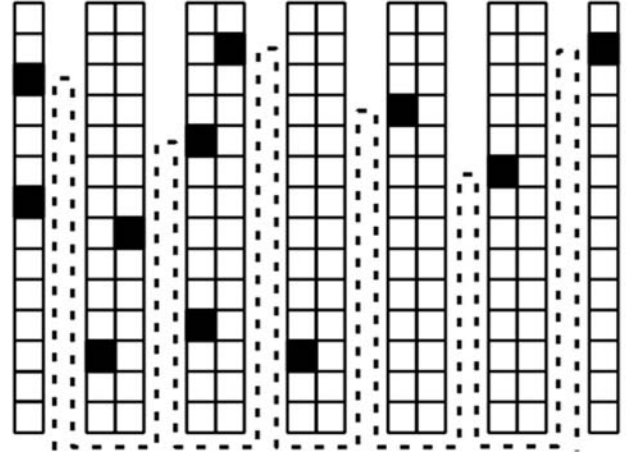
| Faktörler | Seviyeleri | | |
|------------------------------|------------|------|------|
| Çaprazlama oranı | 0,5 | 0,7 | 1 |
| Mutasyon oranı | 0,1 | 0,3 | 0,5 |
| Nesil sayısı | 500 | 1000 | 1500 |
| Popülasyon büyüklüğü | 100 | 500 | 1000 |
| Sipariş sayısı (Küçük boyut) | 5 | 10 | 15 |
| Sipariş sayısı (Orta boyut) | 50 | 60 | 70 |
| Sipariş sayısı (Büyük boyut) | 100 | 150 | 200 |

4.2. Depo Altyapısı (Warehouse Layout)

Bu çalışmada uygulaması yapılan depo paralel açık uçlu, toplama koridorlarına sahip tek bloklu bir depodur. Sipariş toplamak için toplama koridorları vardır ve aynı anda hem sağ hem de sol raflardan ürünler alınabilmektedir. Ürünlerin yerleri ABC analiz sistemine göre belirlenmiştir. Ürünlerin önem dereceleri taleplerine göre bu sistemde tanımlanmakta ve buna göre sınıflandırılmaktadır [36]. Dolayısıyla depoda en çok ticareti yapılan ürünler A sınıfında, daha az ticareti yapılan ürünler B sınıfında, geri kalanlar ise C sınıfındadır. A sınıfı deponun önünde, B sınıfı A sınıfının yanında ve C sınıfı ise kalan depolama alanında yer almaktadır.

4.3. Geri Dönüş Rotalama Sezgiseli (Return Routing Heuristic)

Geri dönüş rotalama sezgiseli, uygulaması oldukça basit bir yöntemdir. Sipariş toplayıcı, ilgili koridora girmekte ve sipariş toplama işlemini bitirdikten sonra, aynı koridordan çıkmaktadır. Şekil 2'de gösterildiği üzere; sipariş toplayıcı, en uzaktaki ürüne kadar giderek ürünü almakta ve sonrasında aynı koridordan geri dönmektedir [37].



Şekil 2. Geri dönüş sezgiseli gösterimi (The representation of the return heuristic algorithm) [38]

4.4. Deneysel Sonuçları (Experimental Results)

Bu çalışmada kullanılan GA, PYTHON programlama dilinde kodlanmış ve Intel Core i5, 2GHz, 16 GB RAM'e sahip bir bilgisayarda ilgili deneyler yapılmıştır. Deneyler sonrasında, sonuç ve hesaplama süreleri kaydedilmiş, aynı zamanda karma tamsayı doğrusal matematiksel model için çözücü olarak CPLEX kullanılmıştır.

GA'ya ek olarak, SG_ZP_ARP'nin çözüm algoritması, firmalarda yaygın bir şekilde kullanımından dolayı, İlk Gelen İlk Hizmet Görür (İĞİHG) kuralı uygulanarak da PYTHON programlama dilinde kodlanmış ve deney sonuçları aynı şekilde kaydedilmiştir. Algoritmalar 30 kez çalıştırılmış, aynı zamanda sonuç ve hesaplama süresi her çalıştırmanın ortalaması alınarak hesaplanmıştır. Burada GA çalıştırılırken kullanılan GA parametreleri, bu parametrelerin analiz edilmesi durumu olmadığı için, Tablo 5'te ifade edilen değerler gözletilmeksizin literatürde en yaygın kullanılan değerler alınmıştır. Buna göre; çaprazlama oranı 0,8, mutasyon oranı 0,2, nesil sayısı 500, popülasyon büyüklüğü 100 olarak alınmıştır. GA sonuçları diğer yöntemlerle kıyaslandığı için GA parametreleri çeşitlendirilmemiş, küçük boyutlu problemler için belirtilen şekilde parametreler oluşturularak değerlendirilmiştir. Sonuçlar Tablo 6'da gösterildiği

üzere oluşturulmuştur. Buna göre, küçük boyutlu veri yapısına sahip olan örnek içerisinde 5 ve 10 müşteri için CPLEX programı sonuç vermekte, ancak 15 müşteri için sonuç verememektedir (CPLEX çözücüsü 10800 saniye ile kısıtlanmıştır). 5 müşteri için en iyi sonucu CPLEX ve GA verirken, 10 müşteri için en iyi sonucu CPLEX çözücüsü vermektedir.

Tablo 6. Küçük, orta ve büyük boyutlu veri setleri için GA, İĞİHG ve CPLEX sonuçları

(GA, İĞİHG and CPLEX results for small, medium and large data sets)

| Müşteri Sayısı | GA | İĞİHG | CPLEX |
|----------------|-----------|-----------|---------|
| 5 | 379,89 | 7516,48 | 379,89 |
| 10 | 3876,36 | 9524,86 | 3788,45 |
| 15 | 6902,31 | 10579,97 | - |
| 50 | 31067,54 | 85346,72 | - |
| 60 | 44402,22 | 135846,46 | - |
| 70 | 53340,49 | 169753,84 | - |
| 100 | 63658,28 | 286456,70 | - |
| 150 | 82058,66 | 389451,08 | - |
| 200 | 100414,41 | 465786,94 | - |

Küçük, orta ve büyük veri setlerinin GA ve İĞİHG kuralı uygulanarak çözülen deneylerin sonuçlarına göre, en iyi sonuçlar GA ile alınmakta olup, İĞİHG kuralı ile oluşturulan algoritmanın sonuçları ise GA

sonuçlarından kötü durumdadır. Müşteri sayısı arttıkça İĞİHG kuralı uygulanarak çözülen deneyler daha kötü sonuçlar vermektedir. Böylelikle bu grafikten GA'nın tüm veri setleri için en iyi sonuçları verdiği anlaşılmaktadır.

GA ile çözülen sonuçların en iyi olduğuna karar verildikten sonra önemli olan nokta GA parametrelerinin hangisinin en iyi olduğuna karar vermektir. Taguchi tekniği ile oluşturulan deney parametreleri GA ile çözülen sonuçları küçük, orta ve büyük boyutlu veri setleri için sırasıyla Tablo 7, Tablo 8 ve Tablo 9'da gösterilmektedir.

Bu bulgular ışığında, sonuç olarak; küçük, orta ve büyük boyutlu veri setleri için en iyi GA parametrelerinin; nesil sayısı ve popülasyon büyüklüğü 1000, çaprazlama oranı 0,75 ve mutasyon oranının ise 0,10 olduğu ifade edilebilmektedir.

Veri setleri arasındaki değişkenliği karşılaştırmak için değişim katsayıları kullanılmıştır ve yapılan analizler sonucunda ulaşılan değerlerin değişim katsayıları Tablo 10'da verilmiştir. Buna göre değişim katsayısı her veri seti için farklılık göstermektedir ve genel anlamda algoritma sonuçları büyük oranda bir değişim katsayısına sahip değildir.

Bir algoritmanın başarısı, çözüme ne kadar hızlı yakınsandığı ile ölçülmektedir [39]. Bu yüzden problemin çözümünde çözüme hangi iterasyonda yakınsandığı önemlidir. Aşağıda verilen Şekil 3, Şekil 4 ve Şekil 5'de x eksenini iterasyon sayısını, y eksenini ise amaç fonksiyonu

Tablo 7. Küçük boyutlu veri seti GA sonuçları (Small size dataset GA results)

| Sipariş sayısı | Çaprazlama oranı | Mutasyon oranı | Nesil sayısı | Popülasyon büyüklüğü | Sonuç | Çözüm süresi (saniye) | En iyi sonuç |
|----------------|------------------|----------------|--------------|----------------------|---------|-----------------------|--------------|
| 5 | 0,50 | 0,10 | 500 | 100 | 379,82 | 2,80 | 379,82 |
| 5 | 0,50 | 0,30 | 1000 | 500 | 379,82 | 10,04 | 379,82 |
| 5 | 0,50 | 0,50 | 1500 | 1000 | 379,82 | 13,22 | 379,82 |
| 5 | 0,75 | 0,10 | 1000 | 1000 | 379,82 | 12,45 | 379,82 |
| 5 | 0,75 | 0,30 | 1500 | 100 | 379,82 | 4,09 | 379,82 |
| 5 | 0,75 | 0,50 | 500 | 500 | 379,82 | 5,86 | 379,82 |
| 5 | 1,00 | 0,10 | 1500 | 500 | 379,82 | 7,70 | 379,82 |
| 5 | 1,00 | 0,30 | 500 | 1000 | 379,82 | 9,31 | 379,82 |
| 5 | 1,00 | 0,50 | 1000 | 100 | 379,82 | 3,82 | 379,82 |
| 10 | 0,50 | 0,10 | 500 | 100 | 4238,69 | 3,55 | 4072,67 |
| 10 | 0,50 | 0,30 | 1000 | 500 | 3946,24 | 20,44 | 3876,19 |
| 10 | 0,50 | 0,50 | 1500 | 1000 | 3906,98 | 22,99 | 3797,46 |
| 10 | 0,75 | 0,10 | 1000 | 1000 | 3884,32 | 20,56 | 3789,35 |
| 10 | 0,75 | 0,30 | 1500 | 100 | 4376,71 | 16,90 | 4025,67 |
| 10 | 0,75 | 0,50 | 500 | 500 | 4244,18 | 8,48 | 4065,29 |
| 10 | 1,00 | 0,10 | 1500 | 500 | 3876,36 | 19,45 | 3799,47 |
| 10 | 1,00 | 0,30 | 500 | 1000 | 3884,62 | 17,63 | 3806,26 |
| 10 | 1,00 | 0,50 | 1000 | 100 | 4251,74 | 15,20 | 4184,00 |
| 15 | 0,50 | 0,10 | 500 | 100 | 7221,67 | 13,18 | 7045,32 |
| 15 | 0,50 | 0,30 | 1000 | 500 | 7011,13 | 19,55 | 6925,95 |
| 15 | 0,50 | 0,50 | 1500 | 1000 | 7745,78 | 23,44 | 6820,62 |
| 15 | 0,75 | 0,10 | 1000 | 1000 | 6902,73 | 20,22 | 6803,24 |
| 15 | 0,75 | 0,30 | 1500 | 100 | 7576,92 | 18,99 | 7139,25 |
| 15 | 0,75 | 0,50 | 500 | 500 | 7268,96 | 15,44 | 6975,51 |
| 15 | 1,00 | 0,10 | 1500 | 500 | 7387,76 | 19,63 | 7194,36 |
| 15 | 1,00 | 0,30 | 500 | 1000 | 6902,31 | 18,00 | 6820,46 |
| 15 | 1,00 | 0,50 | 1000 | 100 | 7698,13 | 20,59 | 7141,42 |

Tablo 8. Orta boyutlu veri seti GA sonuçları (Medium size dataset GA results)

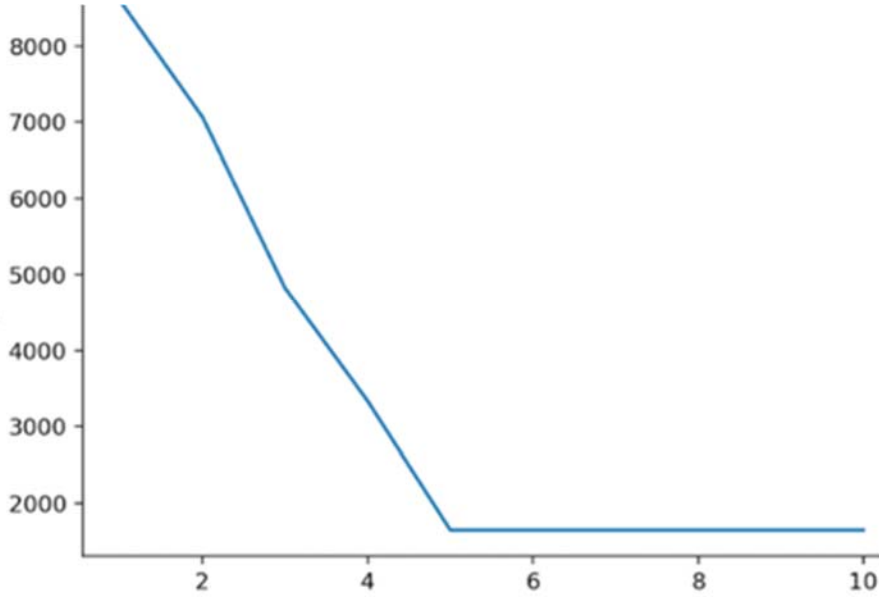
| Sipariş sayısı | Çaprazlama oranı | Mutasyon oranı | Nesil sayısı | Popülasyon büyüklüğü | Sonuç | Çözüm süresi (saniye) | En iyi sonuç |
|----------------|------------------|----------------|--------------|----------------------|----------|-----------------------|--------------|
| 50 | 0,50 | 0,10 | 500 | 100 | 33717,51 | 47,55 | 31034,70 |
| 50 | 0,50 | 0,30 | 1000 | 500 | 31538,19 | 96,88 | 28987,54 |
| 50 | 0,50 | 0,50 | 1500 | 1000 | 31738,78 | 107,41 | 28835,00 |
| 50 | 0,75 | 0,10 | 1000 | 1000 | 31067,54 | 99,64 | 28699,51 |
| 50 | 0,75 | 0,30 | 1500 | 100 | 31858,59 | 86,77 | 28987,54 |
| 50 | 0,75 | 0,50 | 500 | 500 | 32111,83 | 56,33 | 29586,09 |
| 50 | 1,00 | 0,10 | 1500 | 500 | 32069,97 | 91,88 | 30250,99 |
| 50 | 1,00 | 0,30 | 500 | 1000 | 31169,36 | 86,55 | 31098,65 |
| 50 | 1,00 | 0,50 | 1000 | 100 | 32785,25 | 91,47 | 29586,09 |
| 60 | 0,50 | 0,10 | 500 | 100 | 46567,51 | 60,78 | 43792,64 |
| 60 | 0,50 | 0,30 | 1000 | 500 | 45591,93 | 142,45 | 42380,79 |
| 60 | 0,50 | 0,50 | 1500 | 1000 | 44753,21 | 165,70 | 41947,87 |
| 60 | 0,75 | 0,10 | 1000 | 1000 | 44416,18 | 160,59 | 42119,05 |
| 60 | 0,75 | 0,30 | 1500 | 100 | 45174,86 | 99,58 | 42365,51 |
| 60 | 0,75 | 0,50 | 500 | 500 | 47455,49 | 75,00 | 43528,78 |
| 60 | 1,00 | 0,10 | 1500 | 500 | 44402,22 | 105,69 | 42120,64 |
| 60 | 1,00 | 0,30 | 500 | 1000 | 45196,82 | 105,90 | 42524,47 |
| 60 | 1,00 | 0,50 | 1000 | 100 | 47179,33 | 95,84 | 44768,12 |
| 70 | 0,50 | 0,10 | 500 | 100 | 58661,46 | 82,45 | 54850,97 |
| 70 | 0,50 | 0,30 | 1000 | 500 | 54812,19 | 140,57 | 52322,43 |
| 70 | 0,50 | 0,50 | 1500 | 1000 | 53340,49 | 165,43 | 51424,66 |
| 70 | 0,75 | 0,10 | 1000 | 1000 | 54457,84 | 150,38 | 51273,94 |
| 70 | 0,75 | 0,30 | 1500 | 100 | 57678,68 | 105,47 | 53807,92 |
| 70 | 0,75 | 0,50 | 500 | 500 | 57809,37 | 98,74 | 55573,58 |
| 70 | 1,00 | 0,10 | 1500 | 500 | 55003,85 | 110,33 | 51289,64 |
| 70 | 1,00 | 0,30 | 500 | 1000 | 55394,42 | 129,46 | 51866,24 |
| 70 | 1,00 | 0,50 | 1000 | 100 | 57348,36 | 101,89 | 51295,48 |

Tablo 9. Büyük boyutlu veri seti GA sonuçları (Large size dataset GA results)

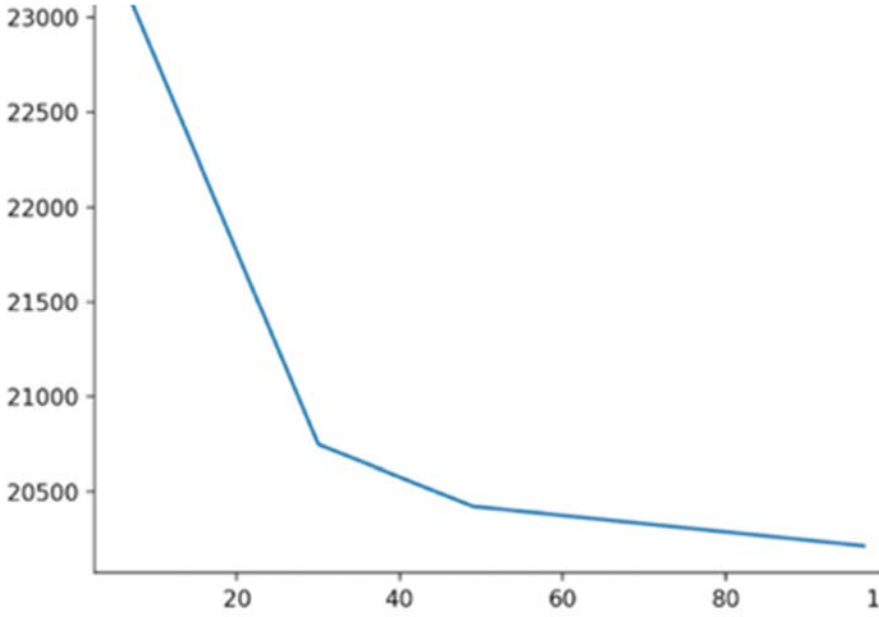
| Sipariş sayısı | Çaprazlama oranı | Mutasyon oranı | Nesil sayısı | Popülasyon büyüklüğü | Sonuç | Çözüm süresi (saniye) | En iyi sonuç |
|----------------|------------------|----------------|--------------|----------------------|-----------|-----------------------|--------------|
| 100 | 0,50 | 0,10 | 500 | 100 | 66492,62 | 92,09 | 64509,91 |
| 100 | 0,50 | 0,30 | 1000 | 500 | 64351,14 | 282,45 | 63764,76 |
| 100 | 0,50 | 0,50 | 1500 | 1000 | 64326,85 | 304,15 | 63281,42 |
| 100 | 0,75 | 0,10 | 1000 | 1000 | 63658,28 | 299,45 | 62812,71 |
| 100 | 0,75 | 0,30 | 1500 | 100 | 66584,35 | 314,59 | 64958,72 |
| 100 | 0,75 | 0,50 | 500 | 500 | 64679,01 | 102,73 | 63597,70 |
| 100 | 1,00 | 0,10 | 1500 | 500 | 64584,34 | 358,46 | 63726,75 |
| 100 | 1,00 | 0,30 | 500 | 1000 | 64173,92 | 420,71 | 63827,89 |
| 100 | 1,00 | 0,50 | 1000 | 100 | 65906,97 | 275,14 | 64608,66 |
| 150 | 0,50 | 0,10 | 500 | 100 | 89698,81 | 218,15 | 83969,90 |
| 150 | 0,50 | 0,30 | 1000 | 500 | 84739,42 | 300,54 | 81596,42 |
| 150 | 0,50 | 0,50 | 1500 | 1000 | 82461,36 | 366,45 | 79406,43 |
| 150 | 0,75 | 0,10 | 1000 | 1000 | 82058,66 | 325,60 | 77469,64 |
| 150 | 0,75 | 0,30 | 1500 | 100 | 86185,51 | 299,58 | 82646,32 |
| 150 | 0,75 | 0,50 | 500 | 500 | 84221,88 | 190,66 | 80952,22 |
| 150 | 1,00 | 0,10 | 1500 | 500 | 83702,72 | 304,44 | 81037,65 |
| 150 | 1,00 | 0,30 | 500 | 1000 | 83031,28 | 250,54 | 79801,55 |
| 150 | 1,00 | 0,50 | 1000 | 100 | 87062,99 | 345,96 | 82249,50 |
| 200 | 0,50 | 0,10 | 500 | 100 | 124882,42 | 543,88 | 120703,15 |
| 200 | 0,50 | 0,30 | 1000 | 500 | 124274,72 | 665,47 | 110093,98 |
| 200 | 0,50 | 0,50 | 1500 | 1000 | 102253,51 | 953,85 | 101043,97 |
| 200 | 0,75 | 0,10 | 1000 | 1000 | 100414,41 | 985,08 | 99750,87 |
| 200 | 0,75 | 0,30 | 1500 | 100 | 115970,72 | 755,14 | 107434,15 |
| 200 | 0,75 | 0,50 | 500 | 500 | 150128,11 | 850,12 | 134364,54 |
| 200 | 1,00 | 0,10 | 1500 | 500 | 115263,11 | 1008,40 | 112381,98 |
| 200 | 1,00 | 0,30 | 500 | 1000 | 115544,62 | 842,07 | 105761,44 |
| 200 | 1,00 | 0,50 | 1000 | 100 | 124195,94 | 590,14 | 123810,27 |

değerini göstermektedir. Bu grafikler sırasıyla; küçük, orta ve büyük boyutlu örneklerin yakınsama davranışlarını ifade etmektedir. Grafiklerden de görüldüğü üzere, küçük boyutlu örnekler 5 iterasyon

sayısı civarında, orta boyutlu örnekler 100 iterasyon sayısı civarında ve büyük boyutlu örnekler ise 1300 iterasyon sayısı civarında iken yakınsama gerçekleştirmektedir.



Şekil 3. Küçük boyutlu örnekler için yakınsama grafiği (Convergence plot for small size samples)



Şekil 4. Orta boyutlu örnekler için yakınsama grafiği (Convergence plot for medium size samples)

Tablo 10. Sonuçların değişim katsayıları
(Coefficients of variation of results)

| Sipariş Sayısı | Değişim katsayısı % |
|----------------|---------------------|
| 5 | 0 |
| 10 | 5,02 |
| 15 | 4,45 |
| 50 | 2,57 |
| 60 | 2,54 |
| 70 | 3,28 |
| 100 | 1,64 |
| 150 | 2,9 |
| 200 | 12,2 |

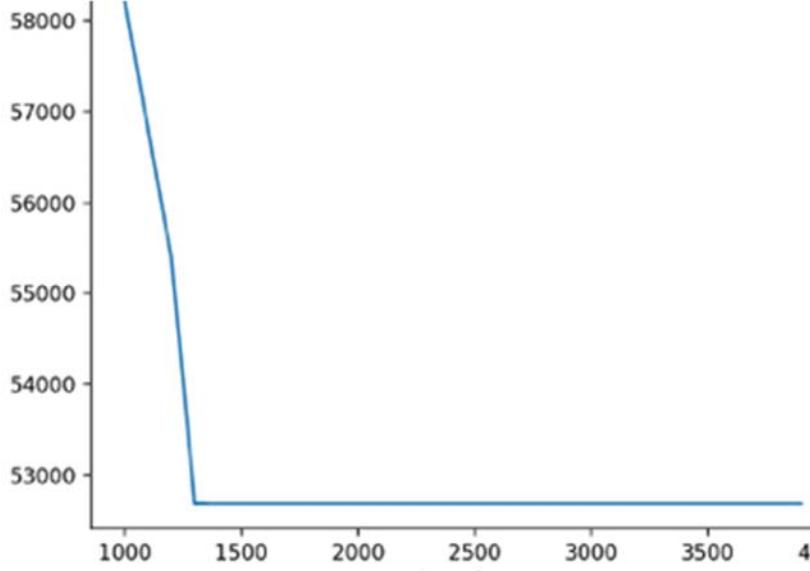
Bu çalışmada ele alınan sipariş toplama ve araç rotalama süreci, içerisinde depoda bulunan siparişlerin ilgili raflardan toplanması

faaliyetini barındırmaktadır. Bu yüzden, sipariş toplayıcının bu siparişleri hangi toplayıcı rotalama sezgiseline göre toplayacağı amaç fonksiyonu değeri için önem arz etmektedir. Çünkü, sipariş toplayıcının siparişleri topladığı sezgisel algoritmaya göre, depo içerisinde geçirilen süre ve mesafe değişiklik göstermektedir. Bu çalışmada oluşturulan karma tamsayılı doğrusal matematiksel model, geri dönüş rotalama sezgiseline göre oluşturulmuş ve problem bu sezgisel ile çözülmüştür. Ancak; diğer rotalama sezgiselleri olan S-şekil, orta nokta ve en büyük boşluk rotalama sezgiselleri ile problemin çözüm sonuçları değişkenlik gösterebileceği için, mevcut problem bu sezgiseller ile de çözülmekte ve sonuçlar Tablo 11’de ifade edilmektedir. Buna göre; en iyi sonuçlar orta nokta rotalama sezgiseli ile alınırken en kötü sonuçlar ise S-şekilli rotalama sezgiseli ile elde edilmektedir. Orta nokta sezgiselinden sonra bu çalışmada uygulanan geri dönüş sezgiseli de ikinci en iyi sonuçları vermektedir.

4.5. Duyarlılık Analizi (Sensitivity Analysis)

Model parametrelerindeki değişimlerin optimum sonucu etkileyip etkilemediği ve eğer bu parametreler sonucu etkilerse ne yönde değiştirdiği üzerine yapılan çalışma, duyarlılık analizi olarak tanımlanmaktadır.

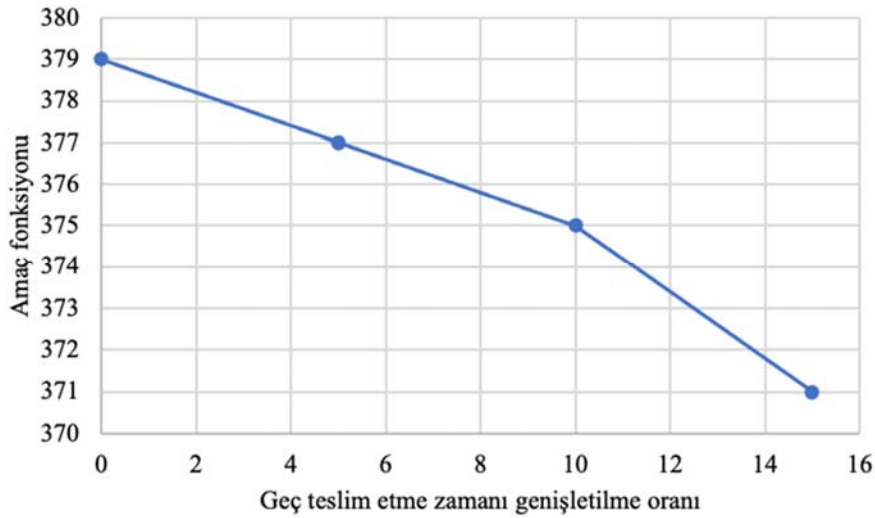
Bu çalışmada duyarlılık analizi; amaç fonksiyonu üzerindeki maliyet katsayıları, müşterilerin zaman pencereleri, sipariş toplayıcı kapasitesi ve dağıtım aracı kapasitesi üzerinden yapılmıştır. Şekil 6'da görülen grafik, müşterilerin ürünleri ile ilgili belirledikleri geç teslim etme süresi üzerine yapılan analizi ifade etmektedir. Buna göre, müşteriler, talep ettikleri geç teslim süresini ne kadar uzatabilirlerse



Şekil 5. Büyük boyutlu örnekler için yakınsama grafiği (Convergence plot for large size samples)

Tablo 11. Rotalama sezgisellerine göre problem sonuçları (Problem results according to routing heuristics)

| Veri seti | Müşteri sayısı | S-şekli | Orta nokta | En büyük boşluk | Geri dönüş |
|---------------|----------------|-----------|------------|-----------------|------------|
| Küçük boyutlu | 5 | 405,86 | 378,50 | 401,40 | 379,82 |
| | 10 | 4652,72 | 3700,58 | 4356,46 | 3789,35 |
| | 15 | 7054,61 | 6798,48 | 7068,32 | 6803,24 |
| Orta boyutlu | 50 | 31584,29 | 27461,67 | 30957,58 | 28699,51 |
| | 60 | 45357,96 | 40729,47 | 43967,15 | 41947,87 |
| | 70 | 57869,17 | 50981,62 | 55681,00 | 51273,94 |
| Büyük boyutlu | 100 | 70286,48 | 60987,25 | 67529,59 | 62812,71 |
| | 150 | 85137,59 | 75927,47 | 83467,67 | 77469,64 |
| | 200 | 109867,90 | 97158,23 | 106527,14 | 99750,87 |



Şekil 6. Geç teslim etme zamanı duyarlılık analizi sonuçları (Late delivery time according to the sensitivity analysis results)

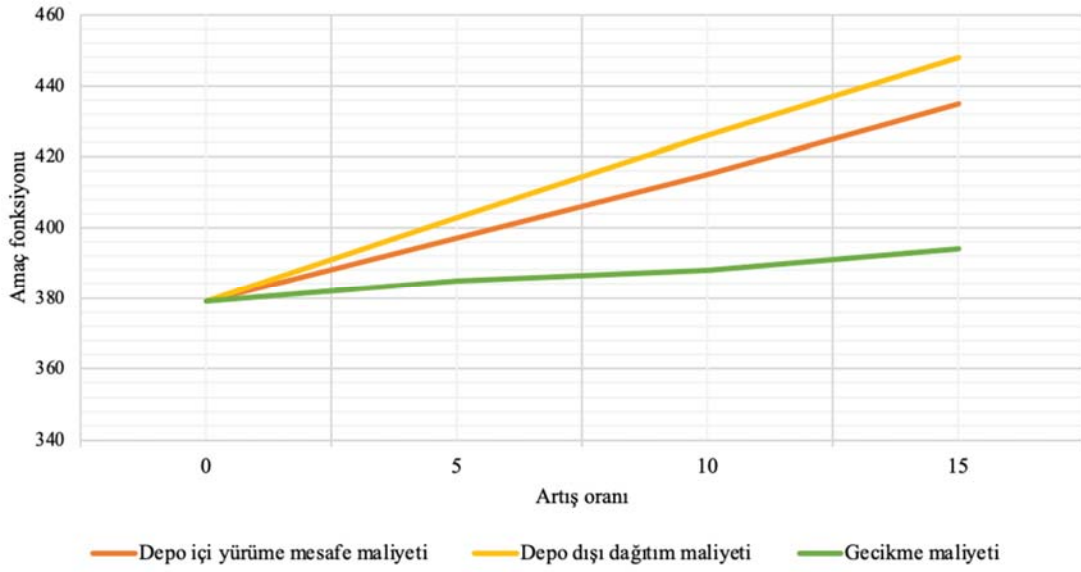
amaç fonksiyonu değeri de buna bağlı olarak düşmektedir. Amaç fonksiyonu içerisinde gecikme maliyeti bileşeni olduğu için ve gecikme yaşayan müşterilerin durumları birbirinden farklı olduğu için, bu analiz sonucuna göre yalnızca geç teslim etme zamanı arttıkça amaç fonksiyonu maliyetinin azaltıldığı ifade edilebilmektedir.

Amaç fonksiyonu parametreleri olan c_1 , c_2 ve c_3 , sırasıyla, depo içi yürüme mesafesi maliyet katsayısı, dağıtım aracı mesafe maliyet katsayısı ve gecikme maliyet katsayısı oranları amaç fonksiyonunda parametre değerler olarak kullanıldığından dolayı duyarlılık analizine dahil edilmiştir. İlgili sonuçlar Şekil 7'de belirtilmektedir ve bu analiz kapsamında üç parametre de belirli oranlarda artırılarak amaç fonksiyonu değeri gözlemlenmiştir. Buna göre, amaç fonksiyonu değerini en çok etkileyen amaç fonksiyonu sabiti depo dışı dağıtım araçlarının mesafe maliyet oranı olmaktadır. Ardından bu sırayı depo içi yürüme mesafe maliyet oranı takip etmektedir ve son olarak amaç

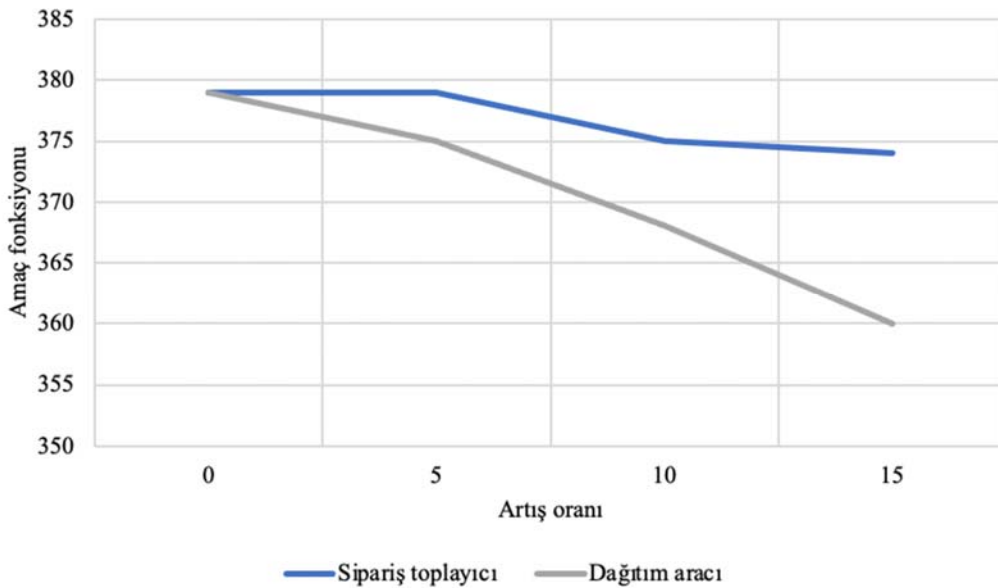
fonksiyonunu en az etkileyen amaç fonksiyonu sabiti ise gecikme maliyeti katsayısı olarak belirtilebilmektedir. Matematiksel model içerisinde ilgili kısıtlarda sağ taraf sabitleri olarak geçen sipariş toplayıcı kapasitesi ve dağıtım aracı kapasitelerinin değerleri belirli oranlarda artırılarak amaç fonksiyonu değeri gözlemlenmiştir. Buna göre Şekil 8'de belirtildiği üzere, dağıtım aracı kapasitesinin artışı amaç fonksiyonu değerini ortalama olarak %1,69 oranında azaltırken, sipariş toplayıcı kapasitesinin artışı ise amaç fonksiyonu değerini ortalama olarak %0,44 oranında azaltmaktadır.

5. Yönetimsel Çıkarımlar ve Sınırlılıklar (Managerial Insights and Limitations)

Şirketler günümüzde daha yaygın bir şekilde müşteriyi kaybetmeme isteği duymaktadır. Müşteri ne kadar ürün sipariş ederse ve bu ürünlerin hangi zaman aralığında teslim edilmesini isterse şirketler bu



Şekil 7. Amaç fonksiyonu sabitleri duyarlılık analizi sonuçları (Objective function constants for sensitivity analysis results)



Şekil 8. Kısıtlar ile ilgili duyarlılık analizi sonuçları (Sensitivity analysis results on constraints)

isteği karşılamak durumundadır. Bu yüzden bu süreci etkili bir şekilde yönetmeleri şirketlerin karlılıklarını artıracaktır. Bu çalışma kapsamında bu süreç araştırılmış, sipariş gruplama problemi ve zaman pencereli araç rotalama problemi bütünlük bir şekilde ele alınmış ve yeni bir karma tamsayı doğrusal matematiksel model geliştirilmiştir. Önerilen matematiksel model, yalnızca küçük boyutlu örneklerin bir kısmı için CPLEX çözücüsü kullanılarak çözülebilmektedir. Bu yüzden; problemin küçük, orta ve büyük boyutlu örnekler için de çözülebilmesi için bir metasezgisel arama algoritması olan GA önerilmiştir. Önerilen bu algoritmanın çözüm sonuçlarının değerlendirilebilmesi için aynı zamanda genelde firmalarda yaygın olarak kullanıldığı için İGİHG kuralı ile de problem sonuçlarına ulaşılmıştır. Bu çalışma ile SG_ZP_ARP için literatüre test verileri eklenmiş olup bu problemin çözümü için GA yaklaşımı önerilmiştir. Ayrıca, sipariş toplayıcı rotalama sezgiselleri ile aynı problem çözümler olarak sonuçları kıyaslanmıştır. İlave olarak sürecin temel etkin faktörlerinin belirlenebilmesi için optimum çözümler üzerinden duyarlılık analizi yapılmıştır.

Bu çalışmada önerilen çözüm yöntemi bazı yönetsel etkilere sahiptir: Bir palet yükünden daha az ürün teslimatı yapacak olan ve müşterilere bu ürünleri belirli zaman aralıklarında teslim etmek isteyen şirketler, bu çalışmada önerilen çözüm yöntemini uygulayarak depo içi ve depo dışı süreçler için bir planlama sonucuna ulaşabilir. Ayrıca çalışmadaki yöntemi uygulayan şirketler aynı zamanda duyarlılık analizi sonuçları ile bazı parametrelerdeki değişikliklerin sonuçlarını izleyebilir.

Bu çalışma, bazı konularda sınırlılıklara sahiptir. Bunlardan birincisi; çalışmada kullanılan depo tipinin yalnızca tek çeşit olmasıdır. Çalışma içerisinde, bir çeşit depo tipi için deneyler gerçekleştirilmekte olup farklı depo tipleri için uygulama yapılmamaktadır. İkinci olarak, stok yerleşimi ABC analizine göre yapıldığı için, ürünler arasındaki mesafeler bu yerleşime göre hesaplanmaktadır. Farklı stok yerleşim türleri ile ürünler arasındaki mesafe dolayısı ile ürünler arasındaki gezinme süresi de değişkenlik gösterebilmektedir. Üçüncü olarak, gecikme maliyeti cezası, müşteri tarafından hesaplanan ve süreye ilişkin bir ceza olarak hesaplanmaktadır. Bu ceza, müşteri kaybı ile ilgili herhangi bir faktörü içermemektedir.

6. Sonuçlar ve Tartışmalar (Results and Discussions)

Bu çalışmada, siparişleri depoda gruplayarak toplayan ve bu siparişleri depodan müşteriye dağıtan bütünlük sipariş gruplama problemi ve zaman pencereli araç rotalama problemi incelenmiştir. Literatür araştırmasına göre, bu problemi araştıran ve çözüm yöntemini öneren çok az sayıda makale bulunmaktadır.

Bu çalışma hem teorik hem de uygulama alanına katkılar sağlamaktadır. Literatüre katkıları izleyen şekilde sıralanabilir:

- Literatüre SG_ZP_ARP için yeni bir karma tamsayı doğrusal matematiksel model eklenmiştir. Bu modelin önemi, siparişler üzerinden değil siparişlerin içerisindeki ürünler üzerinden işlem yapması ve ürünlerin arasındaki uzaklık hakkında mesafeyi çözüm esnasında hesaplayan ilk matematiksel model olmasıdır.
- Mevut literatürde bütünlük problem ile ilgili herhangi bir test verisi yoktur. Ancak bu çalışma ile literatüre ilgili test verileri eklenmiş ve bu test verileri, bundan sonraki süreçte araştırmacıların çalışmalarında bu verileri kullanabileceği ve sonuçlarını ise bu çalışmadaki sonuçlar ile kıyaslayabileceği bir niteliktedir.
- Bu çalışmada bütünlük problemin çözümü için ilk kez GA önerilmiş ve GA parametrelerinin etkinliği üzerine araştırma yapılmıştır.

- Bu çalışma ile SG_ZP_ARP için ve depo tipine göre toplayıcı rotalama sezgisellerinin en yaygın kullanılanları ile ayrı ayrı sonuçlara ulaşılmış ve hangi sezgiselin daha iyi sonuç verdiği ifade edilmiştir.

Bu çalışmanın uygulama alanına katkısı ise bu çalışmadaki yöntemin çevrimiçi ticaret yapan firmalar ve müşterilerine / perakendecilerine bir paletten az ürün ile hizmet veren tüm şirketlerin depoları için uygulanabilir durumda olmasıdır. Bu çalışmanın, bu problemi uygulayan veya uygulayacak olan firmalara ve bu problemi araştıran araştırmacılara yol göstermesi hedeflenmiştir.

Bu çalışma bazı noktalardan genişletilebilir: Bu programlama modeli daha az varsayım ile kesin çözüm yöntemi kullanılarak çözülebilir. Böylece problemin optimum çözümü bulunabilir. Sürdürülebilirlik için probleme çevresel faktörler eklenebilir. Çünkü bu faktörleri göz önünde bulundurmamak hem araç hem de geleceğimiz için önemli bir olgudur. Dahası, bu problem depoda farklı sipariş toplama stratejileri (s-şekilli yönlendirme stratejisi gibi) ile çözülebilir. Heterojen araç ve heterojen sipariş toplayıcı kullanılabilir. Ayrıca, bu sorun farklı depo yerleşimlerinde çözülebilir ve sonuç diğer yerleşim düzeni ile karşılaştırılabilir. Böylece şirket için en uygun depo düzeni bulunabilir.

Teşekkür (Acknowledgement)

Bu çalışma, Esra Boz tarafından yazılan doktora tezinin bir ürünüdür.

Kaynaklar (References)

1. Boz E., Optimizasyon algoritmaları ile depolarda sipariş toplama süreçleri etkinliğinin artırılması, Doktora Tezi, Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Eskişehir, 2022.
2. Zhang J., Wang X., Huang K., Integrated on-line scheduling of order batching and delivery under B2C e-commerce, *Comput Ind Eng*, 94,280-289, 2016.
3. Cergibozaç Ç., Tasan AS., Order batching operations: an overview of classification, solution techniques, and future research, *Journal of Intelligent Manufacturing*, 30, 335-49, 2019.
4. Boz E., Aras N., The order batching problem: A state-of-the-art review, *Sigma Journal of Engineering and Natural Sciences*,40 (2), 402-20, 2022.
5. Vinod HD., Integer Programming and the Theory of Grouping, *J Am Stat Assoc*,64 (326),506-519, 1996.
6. Gademann A., Berg V., H., An order batching algorithm for wave picking in a parallel-aisle warehouse, *IIE transactions*. 33 (5),385-398, 2001.
7. Gademann N., van de Velde S., Order batching to minimize total travel time in a parallel-aisle warehouse, *IIE Transactions*, 37 (1),63-75, 2005.
8. Hsu CM., Chen KY., Chen MC., Batching orders in warehouses by minimizing travel distance with genetic algorithms, *Comput Ind*,56 (2),169-178, 2005.
9. Chen MC., Wu HP., An association-based clustering approach to order batching considering customer demand patterns, *Omega*, 33 (4),333-343, 2005.
10. Henn S., Koch S., Doerner KF., Strauss C., Wäscher G., Metaheuristics for the order batching problem in manual order picking systems, *Business Research*, 3, 82-105 2010.
11. Pan J., Shih P., Wu M., Order batching in a pick-and-pass warehousing system with group genetic algorithm, *Omega*, 57, 238-248, 2015.
12. Nicolas L., Yannick F., Ramzi H., Order batching in an automated warehouse with several vertical lift modules: Optimization and experiments with real data, *Eur J Oper Res*, 267 (3),958-976,2018.
13. Pei Z., Wang Z., Yang Y., Research of Order Batching Variable Neighborhood Search Algorithm based on Saving Mileage. 2019.
14. Gil-Borras S., Pardo EG., Alonso-Ayuso A., Duarte A., A heuristic approach for the online order batching problem with multiple pickers, *Comput Ind Eng*,160, 107517, 2021.

15. Yassen ET., Ayob M., Nazri MZA., Sabar NR., An adaptive hybrid algorithm for vehicle routing problems with time windows. *Comput Ind Eng*, 1 (113),382–391,2017.
16. Song, M. X., Li, J. Q., Han, Y. Q., Han, Y. Y., Liu, L. L., Sun, Q., Metaheuristics for solving the vehicle routing problem with the time windows and energy consumption in cold chain logistics, *Appl Soft Comput*, 95, 106561, 2020.
17. Marinakis Y., Marinaki M., Migdalas A., A Multi-Adaptive Particle Swarm Optimization for the Vehicle Routing Problem with Time Windows, *Inf Sci*, 481 (1),311–329,2019.
18. Liberatore F., Righini G., Salani M., A column generation algorithm for the vehicle routing problem with soft time Windows, *4OR*, 9 (1),49–82, 2011.
19. Schmid V., Doerner KF., Laporte G., Rich routing problems arising in supply chain management. *Eur J Oper Res*, 224 (3), 435–448, 2013.
20. Kuhn H., Schubert D., Holzzapfel A., Integrated Order Batching and Vehicle Routing Operations in Grocery Retail -A General Adaptive Large Neighborhood Search Algorithm, *Eur J Oper Res*, 294 (3),1003–1021, 2020.
21. Öncan T., MILP formulations and an Iterated Local Search Algorithm with Tabu Thresholding for the Order Batching Problem, *Eur J Oper Res*, 243 (1), 142–55,2015.
22. Yağmur E., Kesen S.E, Multi-trip heterogeneous vehicle routing problem coordinated with production scheduling: Memetic algorithm and simulated annealing approaches, *Comput Ind Eng*,161,107649,2021.
23. Talbi EG, *Metaheuristics: from design to implementation*, John Wiley, 2009.
24. Holland JC., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975.
25. Li H., Love PED., Ogunlana S., Genetic algorithm compared to nonlinear optimization for labour and equipment assignment, *Building Research and Information*, 26 (6),322–329,1998.
26. Katoch S., Chauhan SS., Kumar V., A review on genetic algorithm: past, present, and future. *Multimed Tools Appl*, 80 (5), 8091–8126,2021.
27. Altıparmak F., Gen M., Lin L., Karaoglan I. A steady-state genetic algorithm for multi-product supply chain network design, *Comput Ind Eng*, 56 (2),521–537, 2009.
28. Lipowski A., Lipowska D., Roulette-wheel selection via stochastic acceptance, *Physica A: Statistical Mechanics and its Applications*, 391 (6), 2193–2196, 2012.
29. Gen M., Cheng R., *Genetic algorithms and engineering optimization*, John Wiley Sons, 1999.
30. Toth P., Vigo D., A heuristic algorithm for the symmetric and asymmetric vehicle routing problems with backhauls, *Eur J Oper Res*, 113 (3), 528–543, 1999.
31. Cergibozan Ç., Tasan AS., Genetic algorithm based approaches to solve the order batching problem and a case study in a distribution center, *J Intell Manuf*, 33 (1), 137–149, 2022.
32. Pinto ARF., Nagano MS., An approach for the solution to order batching and sequencing in picking systems. *Production Engineering*, 13 (3–4), 325–341,2 019.
33. Xiang X., Liu C., Miao L., Storage assignment and order batching problem in Kiva mobile fulfilment system. *Engineering Optimization*, 50 (11), 1941–62, 2018.
34. Tsai CY., Liou JJH., Huang TM., Using a multiple-GA method to solve the batch picking problem: Considering travel distance and order due time, *Int J Prod Res*, 46 (22), 6533–6555, 2008.
35. Solomon MM., Algorithms for the vehicle routing and scheduling problems with time window constraints *Oper Res*, 35 (2), 254–265, 1987.
36. Görçün ÖF., *Depo ve envanter yönetimi*, Beta Basım Yayım, 2013.
37. Goetschalckx M., Donald Ratliff H., Order picking in an aisle, *IIE transactions*, 20, 53-62, 1988.
38. Roodbergen KJ., *Layout and routing methods for warehouses*, ERIM, Erasmus Research Institute of Management, 2001.
39. Yağmur E., Kesen SE., A memetic algorithm for joint production and distribution scheduling with due dates, *Comput Ind Eng*. 142, 106342, 2020.

