

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ZAMAN PENCERELİ ARAÇ ROTALAMA PROBLEMİ İÇİN
KÜMELEME DESTEKLİ METASEZGİSEL ÇÖZÜM ÖNERİSİ**

YÜKSEK LİSANS TEZİ

Tuğba Gül YANTUR

Endüstri Mühendisliği Anabilim Dalı

Endüstri Mühendisliği Bilim Dalı

MAYIS 2023

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ZAMAN PENCERELİ ARAÇ ROTALAMA PROBLEMİ İÇİN
KÜMELEME DESTEKLİ METASEZGİSEL ÇÖZÜM ÖNERİSİ**

YÜKSEK LİSANS TEZİ

Tuğba Gül YANTUR

Endüstri Mühendisliği Anabilim Dalı

Endüstri Mühendisliği Bilim Dalı

Tez Danışmanı: Doç. Dr. Özer UYGUN

MAYIS 2023

Tuğba Gül YANTUR tarafından hazırlanan “Zaman Pencerele Araç Rotalama Problemi için Kümeleme Destekli Metasezgisel Çözüm Önerisi” adlı tez çalışması 15.05.2023 tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı Endüstri Mühendisliği Bilim Dalı’nda Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı : **Dr.Öğr.Üyesi Abdullah Hulusi KÖKÇAM**
Sakarya Üniversitesi

Jüri Üyesi : **Doç.Dr. Fuat ŞİMŞİR**
Karabük Üniversitesi

Jüri Üyesi : **Doç.Dr. Özer UYGUN** **(Danışman)**
Sakarya Üniversitesi

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum “Zaman Pencereyi Araç Rotalama Problemi için Kümeleme Destekli Metasezgisel Çözüm Önerisi” başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi’nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

15.05.2023

Tuğba Gül YANTUR

TEŐEKKÜR

Çalıőmamda bana yön gösteren, deęerli bilgi ve tecrübeleriyle desteklerini esirgemeyen, kendisiyle çalıőmaktan büyük onur ve gurur duyduğum tez danışmanım sayın Doç. Dr. Özer UYGUN'a sonsuz teşekkürlerimi sunarım.

Birlikte çalıőmaktan gurur duyduğum ve deęerli fikirleriyle bizlere vizyon katan sayın yöneticim Oęuzhan YILDIZ'a, bu süreçte manevi desteęini esirgemeyen ve çalıőmaktan gurur duyduğum deęerli yöneticim Sevcan ER ÇELİK'e ve deęerli çalıőma arkadaşlarıma teşekkürü bir borç bilirim.

Çalıőmamın tamamlanmasında emeęi geçen ve çalıőmamızı deęerli fikirleriyle aydınlatan sayın Arő. Gör. Serap ERCAN CÖMERT'e teşekkürlerimi sunarım.

Son olarak tüm hayatım boyunca manevi destekleriyle yanımda olan en deęerli ve sevgili aileme teşekkürlerimi sunarım.

Tuęba Gül YANTUR

İÇİNDEKİLER

Sayfa

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	vii
TEŞEKKÜR	ix
İÇİNDEKİLER	xi
KISALTMALAR	xiii
TABLO LİSTESİ	xv
ŞEKİL LİSTESİ	xvii
ÖZET	xix
SUMMARY	xxi
1. GİRİŞ	1
1.1. Tezin Amacı	2
1.2. Literatür Araştırması	3
2. ARAÇ ROTALAMA PROBLEMLERİ	9
2.1. Kapasite Kısıtlı Araç Rotalama Problemleri	9
2.2. Topla-Dağıt Araç Rotalama Problemleri	9
2.3. Çok Depolu Araç Rotalama Problemleri.....	10
2.4. Bölünmüş Talepli Araç Rotalama Problemleri	10
2.5. Stokastik Araç Rotalama Problemleri	10
2.6. Zaman Pencere Araç Rotalama Problemleri	10
2.6.1. Zaman pencere Araç rotalama problemi matematiksel modellenmesi ...	11
3. YÖNTEM	15
3.1. Kümeleme Algoritmaları.....	15
3.1.1. Hiyerarşik kümeler.....	16
3.1.2. Hiyerarşik olmayan kümeler	17
3.1.2.1. K-Ortalamlar (K-Means) yöntemi	17
3.1.2.2. K-Medoids yöntemi	18
3.1.2.3. Yoğunluk tabanlı kümeleme algoritması (DBSCAN)	18
3.2. Karınca Kolonisi Optimizasyonu Algoritması	21
3.2.1. Karınca kolonisi algoritması yapısı.....	22
3.2.2. Karınca turunun matematiksel modellenmesi.....	23
3.2.3. Karınca kolonisi algoritması adımları	24
3.3. Yanıt Yüzey Yöntemi Ve Deney Tasarımı	24
3.3.1. Deneyin analizi	26
3.3.2. Deneyin optimizasyonu.....	27
4. UYGULAMA	29
4.1. Teslimat Noktalarını Kümeleme Aşaması	32
4.1.1. Araç içi kapasite ve yerleşim kısıtlarının algoritmaya dahil edilmesi	32
4.1.2. Uygun parametrelerin bulunması	37
4.1.3. Kümeleme sonuçları	44
4.2. Zaman Pencere Araç Rotalama Uygulaması	45

4.2.1. Algoritma parametrelerinin belirlenmesi	45
4.2.2. Karınca kolonisi algoritmasının uygulanması.....	51
5. SONUÇ VE ÖNERİLER	59
KAYNAKLAR.....	61
EKLER	65
ÖZGEÇMİŞ.....	68

KISALTMALAR

GSP	: Gezgin Satıcı Problemi
ARP	: Araç Rotalama Problemleri
KKARP	: Kapasite Kısıtlı Araç Rotalama Problemleri
ZPKARP	: Zaman Pencereleli Kapasiteli Araç Rotalama Problemleri
EZPARP	: Esnek Zaman Pencereleli Araç Rotalama Problemleri
DBSCAN	: Yoğunluk Tabanlı Kümeleme
RSM	: Yanıt Yüzeyi Yöntemi
GA	: Genetik Algoritma
SNN	: Paylaşılan En Yakın Komşu
ANOVA	: Varyans Analizi
km	: kilometre
m^2	: metrekare
ϵ	: Epsilon

TABLO LİSTESİ

Sayfa

Tablo 4.1. Teslimat Başlangıç ve Bitiş Zamanları	30
Tablo 4.2. Sipariş Gruplandırması.....	31
Tablo 4.3. Kümeleme Algoritması	33
Tablo 4.4. Kümeleme Algoritması Deney Tasarımı	38
Tablo 4.5. Response A için Faktörlerin Analizi	39
Tablo 4.6. Response A için Model Özeti	39
Tablo 4.7. Response B için Faktörlerin Analizi	40
Tablo 4.8. Response B için Model Özeti.....	40
Tablo 4.9. Response C için Faktörlerin Analizi	42
Tablo 4.10. Response C için Model Özeti.....	42
Tablo 4.11. Kümeleme Sonuçları.....	44
Tablo 4.12. Karınca Kolonisi Algoritması Deney Tasarımı.....	47
Tablo 4.13. Response A - Ceza Maliyeti Çıktıları	48
Tablo 4.14. Response A - Ceza Maliyeti Model Özeti	48
Tablo 4.15. Response B – Alınan Yol Çıktıları.....	49
Tablo 4.16. Response B – Alınan Yol Model Özeti.....	50
Tablo 4.17. Karınca Kolonisi Algoritması adımları.....	51
Tablo 4.18. Çalışma Sonucu Elde Edilen Çıktılar.....	53
Tablo 4.19. Çalışma Öncesi Durum	57

ŞEKİL LİSTESİ

Sayfa

Şekil 3.1. Kümeleme Algoritmalarının Sınıflandırılması	15
Şekil 3.2. Epsilon Değerleri Arasındaki Komşuluklar.....	19
Şekil 3.3. Üç tip nokta ve kümelemesi (Rezai,2016).....	21
Şekil 3.4. Y, X1 ve X2 değişkenlerine ait Yanıt Yüzeyi	25
Şekil 4.1. Teslimat Adresleri ve Ana Depo Konumu	29
Şekil 4.2. Araç İçi Alt ve Üst Taban Alanı Gösterimi	31
Şekil 4.3. Araç İçi Yerleşim Algoritma Akış Diyagramı.....	36
Şekil 4.4. Teslimat Adresleri Ortalama Uzaklıklar.....	37
Şekil 4.5. Response A değerinin Pareto Grafiği	40
Şekil 4.6. Response B değerinin Pareto Grafiği	41
Şekil 4.7. Response C değerinin Pareto Grafiği	43
Şekil 4.8. Response Optimizer Sonucu.....	43
Şekil 4.9. Kümeleme Sonuçlarının Harita Üzerinde Gösterimi.....	44
Şekil 4.10. Optimizer Sonuçları.....	50
Şekil 4.11. Oluşturulan Tüm Rotalar	53
Şekil 4.12. Küme 1 Rotası	54
Şekil 4.13. Küme 2 Rotası	54
Şekil 4.14. Küme 3 Rotası	55
Şekil 4.15. Küme 4 Rotası	55
Şekil 4.16. Küme 5 Rotası	56
Şekil 4.17. Küme 6 Rotası	56
Şekil 4.18. Küme 7 Rotası	56

ZAMAN PENCERELİ ARAÇ ROTALAMA PROBLEMİ İÇİN KÜMELEME DESTEKLİ METASEZGİSEL ÇÖZÜM ÖNERİSİ

ÖZET

Araç rotalama problemleri, başta lojistik ve dağıtım sektörleri olmak üzere literatürde üzerine oldukça çalışılmış ve çeşitli kısıt ve parametrelerden oluşan gerçek hayat problemlerinden biridir. Ana amacı maliyet enküçüklemesi olan araç rotalama problemleri içeriği kapsamında sezgisel ya da metasezgisel yöntemlerle çözümlenmektedir.

Bu çalışmada, bir lojistik firmasına ait beyaz eşya ana dağıtım deposundan Ankara ve çevre illerdeki ev ya da bayilere yapılacak olan teslimatların, teslimat saat aralığı kısıtı altında rotasının planlanmasıdır.

Firma Ankara’da yer alan ana dağıtım deposundan Ankara ve çevre illerdeki 75 adet bayiye 20 dakika servis süresi ile teslimat yapmaktadır. Her bayinin kendine ait teslimat zaman aralıkları bulunmaktadır. Müşteri gereksinimi olarak bayiler için belirlenen saat aralığında teslimat yapılması en önemli kısıttır. Zaman penceresinden önce ya da sonra teslimat yapılabilir ancak geç kalınması durumunda ceza maliyeti olarak maliyeti artıracaktır, incelenen problem bu nedenle “Esnek zaman pencereci araç rotalama” problemleri sınıfında yer almaktadır.

Problem mevcut depoda gerçek zamanlı bir problem olmakla birlikte, sevkiyatlar planlama personelinin deneyim ve görüşlerine dayanarak planlanmakta, sistemsel bir rotalama çalışması bulunmamaktadır. Hedeflenen, sevkiyatlarda maliyetlerin azaltılması, minimum sefer sayısı ile maksimum teslimat yapılmasıdır. Sevkiyat ve hizmet kalitesinin artırılması ve planlamada oluşan hataların minimuma indirilmesi hedeflenmektedir.

Sevkiyat planlama personelinin yapmış olduğu planlama sistemi incelenerek sistematik hale getirilecek çıkarımlar analiz edilmiştir. Elde edilen verilere göre bayi konumları, teslimat saat aralıkları, sipariş hacimleri sınıflandırılmıştır, müşteriye ve sevkiyata ait kısıtlar analiz edilmiştir.

Önce kümele sonra rotala yaklaşımına dayanan iki aşamalı yöntem önerilmiştir. Teslimat adresleri kapasite kısıtı altında MATLAB R2022a programı kullanılarak kümelendirilmiştir. Siparişlerin düzenli olarak Ankara bölgesinde yoğunluk göstermesi nedeniyle yoğunluk bazlı bir kümeleme gerektirmesi, küme sayısının başlangıçta bilinmemesi ve gürültü noktalarına karşı toleranslı bir kümeleme algoritması olması nedeniyle DBSCAN algoritmasının uygun bir yöntem olduğuna ulaşılmıştır.

Kümelenen her bir rota için MATLAB R2022a programı kullanılarak metasezgisel yöntemlerden biri olan Karınca Kolonisi Algoritması yaklaşımı ile zaman pencereler kısıtı altında araç rotalaması yapılmıştır. Her bayi için zaman pencereleri dışında geç ziyaret etme durumlarında ceza maliyeti eklenmiştir.

Analiz sonucu elde edilen veriler ile daha önce yapılmış olan planlama verileri mali ve operasyonel olarak karşılaştırılmıştır.

A CLUSTERING APPROACH FOR THE METAHEURISTIC SOLUTION OF VEHICLE ROUTING PROBLEM WITH TIME WINDOW

SUMMARY

In the last few years, one of the most important factors in the increasing competition in the global and local markets has been the delivery of the service or product to the end customer as soon as possible. In order to take part in this competition, companies focus on logistics processes in line with the expectations of their customers.

In addition to delivering the product or service to the end customer as soon as possible, companies aim to keep their costs constant or reduce their costs with various strategies. Completing internal and external processes quickly and without sacrificing quality causes high costs. Fleet management and planning constitute one of the most financially significant processes within the realm of logistics operations. It is necessary to achieve optimum results by performing accurate analysis of multi-criteria decision-making processes.

In the literature, various methods have been developed to solve vehicle routing problems depending on each criterion. Constraints such as capacity, distance, time, pick-and-drop and demand can be one of these constraints.

Vehicle routing problems, the main purpose of which is cost minimization, are solved with heuristic or metaheuristic methods within the scope of their content.

During the shipment process, which vehicles will leave the warehouse at which capacity, which route they will follow, and in which order they will reach the delivery points can be planned as optimum and close to the optimum with the help of various techniques.

In this study, the problem is to plan the routes for delivering white goods from a main warehouse to homes or dealers in Ankara and surrounding cities, considering the delivery time window constraint. The main warehouse, where the delivery orders are located, is in Kahramankazan, Ankara. The provinces within the scope of the surrounding provinces are Kırıkkale, Çankırı, and Kırşehir.

The most important constraint provided by the customer is the time windows, and services can be provided outside of these time windows, but in such cases, a penalty cost is applied. The penalty cost for unit lateness is set to ten times the duration of being late. Due to allowing late or early visits, the problem type is categorized in the literature as "Flexible Time Window Vehicle Routing Problem" under the heading of "Vehicle Routing Problems."

The service time that the dealers will offer to the delivery points is considered equal and 20 minutes. Time Windows are fixed and an appointment is made for one of the following time windows from sales dealers to customers for each delivery. The main focus in solving the problem is to minimize cost and deliver within the time window.

A two-stage method based on “cluster first and then route” approach has been proposed.

The order lists have been examined and it has been observed that the number of orders, number of dealers, population, the delivery addresses etc. were concentrated in the Ankara region.

For this reason, it has been concluded that the most suitable algorithm for the problem is the density-based clustering algorithm (DBSCAN).

The number of vehicles, defined as the number of clusters, is also unknown at the beginning of the problem. The DBSCAN algorithm, which is a suitable method for density-based clustering, has been applied with the inclusion of noise points.

Delivery load and vehicle capacity are calculated based on floor area. The sum of the floor areas of the deliveries will not exceed the vehicle floor area.

The ground and top area of the vehicle is constant and $30 m^2$, throughout the problem. In cases where the order total exceeds $30 m^2$, the remaining orders must be loaded onto another vehicle.

The upper floor area of the vehicle is an imaginary area designed for product groups that are placed on the floor but not too high and can be placed on top of each other.

The orders referred to as A-Group orders are refrigerator orders that cover both the ground floor area and the top floor area of the vehicle.

B-group products include washing machine, dryer and dishwasher product categories. Different from the A group products, the ones in this product group can be placed on top of each other in a maximum of 2 pieces in the vehicle.

While the floor area of A group refrigerator orders is taken as $1.3 m^2$, the floor area of B group products is assumed to be $0.36 m^2$. Although the base sizes vary according to the model types, the values obtained from the average of the real values are assumed as fixed values as the problem constraint.

Response Surface Method (RSM) were used to optimize the *Minpts* and ϵ values in the clustering phase. While the factors were chosen as *Minpts* and ϵ values in the experimental design, the number of clusters, occupancy rate and noise numbers were taken as the outputs.

If the number of clusters is minimized, the occupancy rates are maximized and the noise points number is minimized, the Response Optimizer detects the optimum value as 95.68 km for the ϵ parameter specified as Factor A and as 3 points for the *Minpts* parameter specified with Factor B.

The delivery points are clustered with the density-based DBSCAN algorithm with the optimum parameters.

During clustering, in-vehicle placement was integrated so as not to exceed the capacity of the vehicle. If there is capacity in the vehicle, the DBSCAN algorithm structure was reshaped by including the next point in the cluster.

Thence the vehicle occupancy rate was increased by placing the product groups that meet the stackability criterion on top of each other.

Each cluster represents a vehicle. After the vehicles are clustered on the basis of their capacity, location constraints and their proximity to each other, the problem arose in which order the delivery will be made according to the time windows.

As a result of clustering with optimum parameter values, 7 clusters were formed.

The average occupancy rates of the first 6 clusters are clustered as 94.89%, and since the last cluster is included in the other provinces category, it is a known and accepted constraint that the orders are always low.

In the second stage, clusters with a high occupancy rate and a minimized number of vehicles will be routed to make their deliveries under time window constraints with the Ant Colony Algorithm approach, which is one of the metaheuristic methods, using the MATLAB R2022a program. The constraint, which can wait if it arrives at the delivery point earlier than the time start, but adds penalty cost if it arrives late, is integrated into the algorithm.

Response Surface Method (RSM) has been also used to optimize the parameters. While the factors were selected as *MaxIT*, *Nant*, *alpha*, *beta* parameters in the experimental design, the penalty cost and the distance travelled (*km*) value obtained as a result of the analysis were taken as the outputs.

When the results are examined, Factor A is 100 for *MaxIT* parameter; Factor B is 162 for parameter *Nant*; Factor C was determined as 5 for *alpha* parameter and 0 for *beta* parameter as optimum values.

Unlike the classical Ant Colony algorithm, delivery constraints within time windows are included in the algorithm. Each vehicle will be routed taking into account time window constraints.

As a result of this routing, an approach that provides both capacity, placement constraints and time window constraints is proposed. The data obtained as a result of the analysis and the previous planning data were compared financially and operationally. The proposed approach has achieved a 30% improvement in the number of vehicles. The vehicle occupancy rates have been increased to an average of 94.89%.

This study offers a new solution proposal approach for vehicle routing problems with large datasets and time windows. In future studies, studies on the conditions assumed or accepted in this study are planned.

1. GİRİŞ

Günümüzde global ve yerel pazarda artan rekabette en önemli unsurlardan birisi son kullanıcıya hizmet ya da ürünün en kısa sürede sunulması olmuştur. Bu rekabette yer alabilmek için firmalar, müşterilerinin beklentileri doğrultusunda lojistik süreçlerine yoğunlaşmaktadır.

Son müşteriye ürün ya da hizmetin en kısa sürede ulaştırılmasının yanında firmalar kendi maliyetlerini çeşitli stratejilerle sabit tutmak ya da azaltmak hedefindedir. İç ve dış süreçlerin hızlı ve kaliteden ödün vermeden tamamlanması yüksek maliyetlere neden olmaktadır. Lojistik süreçlerinin tamamında en maliyetli süreçlerden biri filo yönetimi ve planlanmasıdır. Çok kriterli karar verme süreçlerinden doğru analizleri gerçekleştirerek optimum sonuçlara ulaşmak gerekmektedir. Filo araçlarından minimum maliyetle maksimum verimi alabilmek için çeşitli optimizasyon teknikleri rota planlama ve çizelgeleme süreçlerinde uygulanmaktadır.

Her bir kritere bağlı olarak araç rotalama problemlerini çözmek için literatürde çeşitli yöntemler geliştirilmiştir. Kapasite, mesafe, zaman, topla-bırak ve talep gibi kısıtlar bu kısıtlardan biri olabilir. Araç rotalama problemleri NP-zor problemler sınıfında yer almakta olup, sezgisel ve meta-sezgisel çözüm metotları ile çözümlenebilir.

Süreç dinamiklerine göre farklı unsurlar rota planlanmasında kriter olarak ele alınabilmektedir. Müşteri isteğine göre talep önceden bilinebilir ya da değişkendir. Değişen şartlara ayak uydurabilen bir planlama sağlamak çok daha zor olabilir. Bazen de müşterinin en önemli kriteri kalite olabilir, bu durumda kalite süreçlerinde yapılacak olan süreç iyileştirme adımları müşterinin beklentilerini karşılayacaktır.

Dış lojistik hizmeti sağlayıcıları süreçlerinde müşteri kontratlarına bağlı olarak, sipariş tamamlanma süresi de en önemli kriterlerden biri olabilir. Özellikle e-ticaret sektöründe yer alan firmalar geç teslimatları önlemek için dış lojistik hizmeti sağlayıcılar ile stratejik bir ilişki kurarlar. Müşterinin düşük maliyet ve yüksek performans beklentilerini karşılamak amacıyla depolama, paketleme, etiketleme, sevkiyat süreçleri üçüncü parti lojistik firmaları tarafından sağlanır. Müşteri beklentilerine uygun olarak, bayi ve son kullanıcı teslimatlarında “Zamanında

Teslimat” kriterine yoğunlaşarak lojistik süreçlerinde iyileştirmelerin gerçekleştirilmesi hedeflenmektedir.

Zamanında teslimat kriteri, lojistik süreçlerinde depo mal kabulünden başlayan ve sevkiyat sonuna kadar incelenmesi gereken bir süreci kapsar. Bu kriterin karşılanabilmesi için mal kabulden sevkiyata kadar geçen sürenin doğru planlanması büyük önem taşımaktadır. Sevkiyat sürecinde hangi araçların hangi kapasitede depodan çıkış yapacağı, hangi rotayı izleyeceği ve hangi sıralama ile teslimat noktalarına ulaşacağı çeşitli teknikler yardımıyla optimum ve optimuma yakın olarak planlanabilir.

1.1. Tezin Amacı

Bu çalışmada dış lojistik hizmeti sağlayıcısına ait beyaz eşya deposundan bayilere yapılan teslimatların planlanması problemi incelenmiştir. Ana depodan Ankara ve çevre illerdeki ev ve bayilere teslimat işlemi gerçekleştirilir.

Müşterinin en önemli 2 kriteri Kalite ve Zamanında Teslimat kriterleridir. Dış lojistik hizmeti sağlayıcı firma kalite problemlerini çözümlmek için sevkiyat süreçlerinde kaizenler uygulayarak kalite hedeflerinin üzerinde bir performans sergilemişlerdir. Zamanında teslimat hedeflerini tutturmak için araç rotalama stratejilerini gözden geçirecektir.

Amaç minimum araç sayısı ile maliyet azaltımı ile bir zaman pencereleri içerisinde en son kullanıcı olan müşteriye hizmet sağlamaktır. Rota için her yeni eklenen araç, yakıt ve personel başta olmak üzere maliyeti artıran en büyük unsurdur. Bir diğer maliyet arttırıcı etmen ise zamanında teslim edilmeyen teslimatlar nedeniyle oluşan ceza maliyetleridir. Bu durum hem müşteri-dış lojistik hizmeti sağlayıcı firma arasında müşteri ilişkilerini zayıflatarak ileriki dönemlerde kontrat yenilenme sürecini etkileyecektir, hem de en son kullanıcı tarafından alınan olumsuz yorumlar nedeniyle müşteri memnuniyetinden kayba neden olacaktır.

Tezin problemi çözümlemesindeki ana odağı, maliyet ve zaman penceresi içerisinde teslimatların gerçekleştirilmesidir.

Bir sonraki bölümde Literatür Araştırması, uygulanan yöntemler ve uygulama adımları anlatılacaktır. Son bölümde ise uygulamanın çıktıları yorumlanarak çalışmanın gelecek planları ele alınmıştır.

1.2. Literatür Araştırması

1959 yılında ilk kez Dantzig ve Ramser tarafından önerilen Araç Rotalama Problemleri için literatürde, çeşitli kısıtlar ile ele alınmış ve bu problemlere çeşitli çözüm yöntemleri önerilmiş pek çok çalışma yer almaktadır.

Araç rotalama problemi türlerinin çeşitliliği, kısıt ve çözüm önerilerinin çeşitliliği nedeniyle bu bölümde "Önce Kümele Sonra Rotala" yaklaşımını ele alan yakın zamanda yapılmış bazı çalışmaları bu başlık altında incelenmiştir.

Araç Rotalama Problemlerinin Önce Kümele Sonra Rotalama Yaklaşımları ile çözümlenen bazı çalışmalar şunlardır;

Zhang (2017) çalışmasında, Kapasite Kısıtlı Araç Rotalama Problemi (KKARP) için yoğunluk tabanlı kümeleme (DBSCAN) metodu ve Karınca Kolonisi Algoritması ile rotalama yaklaşımını önermiştir. Tüm müşteri noktaları yoğunluklarına göre kümelenene kadar kümeleme işlemi tekrarlanır. Büyük ölçekli kapasite kısıtlı araç rotalama problemleri için kısa sürede optimum çözümü elde etme konusunda başarılı bir yaklaşım sunulmuştur.

Cömert ve ark. (2019), tek depodan müşterilere haftalık olarak dağıtım hizmeti sağlarken aynı zamanda toplama hizmeti de gerçekleştiren bir Eş Zamanlı Topla Dağıt araç rotalama problemini ele almışlardır. İlk aşamada kümeleme yöntemlerinden K-Ortalamlar ve K-Medoids kullanılarak noktalar kümelendi. İkinci adımda ise tam sayılı doğrusal programlama ile kümeler rotalanarak maliyet fonksiyonunu minimize etmek hedeflenmiştir. Çalışma sonunda ANOVA testi ile elde edilen sonuçlar ve maliyet fonksiyonları sonucunda iki yöntemin de benzer sonuçlar verdiğine ulaşılmıştır.

Akdaş ve ark. (2021) çalışmalarında, Maltepe Belediyesi tarafından sağlanan katı atık toplama sürecine ait araç rotalama problemine, kümeleme ve Karınca kolonisi algoritmalarını uygulamış ve %13 daha kısa rota uzaklığı elde etmişlerdir.

Villalba ve Rotta (2020), temsilci müşteri seçimi için yoğunluk tabanlı kümeleme (DBSCAN) ve K-Ortalamlar kümeleme algoritmaları uygulanmış ve sonuçları incelenmiştir. Kümeleme kalitesi, hesaplama süresi, küme sayısı vb. etkenler değerlendirilerek her iki yöntemin de güçlü ve zayıf olduğu yönleri karşılaştırmıştır.

Choudhari ve ark. (2022), kapasite kısıtlı araç rotalama problemini DBSCAN algoritması kullanarak yoğunluk bazlı kümelemiş ve tam sayılı lineer doğrusal programlama yöntemini kullanılarak tek depolu kapasite kısıtlı araç rotalama problemini incelemiştir. Çözüm gerçek hayat durumlarında kullanılabilir olup rotalamada Christofide algoritması kullanılmıştır.

Bu bölümün devamında Araç Rotalama Problemlerinin "Önce Kümele Sonra Rotala" yaklaşımlarıyla çözümlendiği "zaman penceresi" kısıtı altında ele alındığı bazı çalışmaları inceleyeceğiz.

Özdağ ve ark. (2012), çalışmalarında üniversite güz dönemi öğretim planını oluşturmak üzere ders saati ve günü taleplerini ağırlıklı graf modeli oluşturmuş ve Karınca Kolonisi algoritması kullanılarak optimum dağılımını gerçekleştirmiştir. Python programlama dili kullanılarak veri girişi yapılmakta ve MySQL veritabanı ile veriler kaydedilmekte ve sorgulanabilmektedir.

Ünsal ve Yiğit (2018), çalışmalarında okul servisinin rotalanması problemini zaman ve maliyet kısıtları altında ele aldılar. İlk olarak kümeleme fazında K-ortalamlar algoritması ile noktaları kümeledikten sonra en uygun rotanın belirlenmesinde yapay zeka tekniklerinden biri olan Genetik Algoritma (GA) yöntemini kullandılar. Elde edilen sonuçlar incelendiğinde hem zaman hem de kilometre cinsinden yol kazancı sağlanmıştır.

Bujel ve ark. (2018), zaman pencereli kapasiteli araç rotalama probleminin (ZPKARP) performansını iyileştirmek amacıyla Bölümlendirilmiş Yoğunluk Tabanlı kümeleme (Recursive-DBSCAN) yöntemini uygulayarak Google OR-Tools ve klasik yoğunluk tabanlı kümeleme (DBSCAN) yöntemini kıyaslamışlardır. Sonucunda teslimat noktalarını yinelemeli olarak kümeleyerek, Google Optimizasyon araçlarına kıyasla rota sürelerini %61 oranında azaltmıştır.

Bozdemir ve ark. (2019), çalışmalarında şirket çalışanlarının sabah ya da akşam çalışma durumlarına göre en yakın noktadan servis rotasının oluşturulması problemini ele aldılar. Durak noktaları için belirlenen noktalar K-Ortalamlar, K-medoids ve K-modes olmak üzere 3 farklı kümeleme algoritması uygulanarak sonuçları incelendi. Yöntemler birbirleri arasında kıyaslanmıştır.

Göçken ve Yaktubay (2019), çalışmalarında zaman pencereli ve kapasite kısıtlı araç rotalama probleminde, öncelikle K-Ortalamlar, Merkez Tabanlı sezgisel kümeleme, yoğunluk tabanlı kümeleme ve Paylaşılan En Yakın Komşu (SNN) kümeleme metotları ile kümelendi. Rotalama aşamasının ardından Genetik algoritma yöntemini kullanarak rotanın optimizasyon işlemi tamamlanmış ve Solomon kıyaslama veri kümesi ile test edilmiştir. Seyahat süresi, bekleme süresi ve araç sayısı gibi kriterler göz önüne alındığında daha iyi sonucu veren K-Ortalamlar yöntemi ile kümelendi olan rotalama çözümü olmuştur. Çeşitli kriterler üzerinden kümeleme algoritmalarının performansları yorumlanmıştır.

Cömert ve ark. (2020) çalışmalarında, Esnek Zaman Pencereli Araç Rotalama (EZPARP) problemine uygun bir süpermarket problemi örneğini, önce kümele sonra rotala yöntemine dayalı karışık tam sayılı doğrusal programlama modeli ile çözümlenmiştir. Müşteri talepleri belirlenen zaman pencereleri içerisinde karşılanmaması durumunda ceza uygulanması göz önüne alınarak K-Ortalamlar (K-Means) ve K-Medoids kümeleme yöntemleri ile kümelendi. Rotalama aşamasında tam sayılı doğrusal programlama kullanılmıştır. Elde edilen sonuçlar firmanın gerçek maliyetleri ile karşılaştırılmış ve önerilen modelin daha iyi bir sonuç verdiği gözlemlenmiştir. Her iki yöntem karşılaştırıldığında ise K-Medoids K-Ortalamlar algoritmasından daha iyi sonuçlar vermiştir.

Villalba ve Rotta (2022), zaman pencereli araç rotalama probleminde, Python yazılımını kullanarak K-Ortalamlar ve Optics kümeleme yöntemleri ile kümeleme adımını gerçekleştirmiştir. En Yakın Komşu yöntemini kullanarak rotaları oluşturmuş ve rota optimizasyonunu gerçekleştirerek önce kümele sonra rotala temelli bir yaklaşım önermiştir. Ayrıca, Solomon örnekleri temel alınarak bir karşılaştırma yapılmış, sonucunda toplam mesafede %2.09'lük bir azalma ve gereken araç sayısında %10.87'lik bir azalma gözlemlenmiştir.

Sanchez ve ark. (2022), zaman pencereli araç rotalama problemlerinde hava ve gürültü kirliliğine neden olmayan, yüksek enerji verimliliğine sahip elektrikli araçların rotalamasını incelemişlerdir. Bu çalışma, zaman pencerelerini göz önünde bulunduran batarya ünitesi konumlandırma problemi için karışık tam sayılı lineer programlama modeli önermektedir. Seyahat süresi ve pil seviyeleri gibi kriterler göz önüne alınmıştır. K-Ortalamlar ile noktalar kümelenmiş ve en uygun batarya ünitesi noktaları hesaplanmıştır. Elektrikli araçların sayısının genellikle yük talebi ve müşteri zaman pencereleri tarafından belirlendiğini göstermektedir. K-Ortalamlar kümeleme yaklaşımının kullanılması, arama alanını etkili bir şekilde azaltarak hesaplama sürelerini düşürmekte önemli bir rol oynadığı gözlemlenmiştir.

Le ve ark. (2022), COVID-19 pandemisi yaşandığı dönemde belirli zaman aralıklı iş merkezlerinin ve marketlerinin açılması nedeniyle bozulabilir gıda dağıtımında yaşanan rotalama problemi olan, aynı zamanda bir zaman pencereli araç rotalama probleminin çözülmesi üzerine çalışmışlardır. Ürünler farklı boyalarda paketlenmiş olup aracın kapasite kısıtı da çalışmaya dahil edilmiştir. Kapasiteli K-Ortalamlı Kümeleme Algoritması ve Dal-Sınır Algoritmasının uygulanmasıyla mevcut durumdan daha iyi sonuçlar elde edildiği gözlemlenmiştir.

Wang ve ark. (2023), çalışmalarında çok depolu zaman kısıtlı araç rotalama problemlerini müşterilerin kümelenirken alt kümelerin de oluşması ve ardından Genetik algoritma ile rotanın planlanması olarak çözümlenmişlerdir. Çalışma klasik kümeleme yöntemlerinden çok daha iyi sonuçlar elde etmiştir.

Bu çalışmada yer alan problem, Zaman pencereli araç rotalama problemi sınıfında olup, kapasite kısıtları dikkate alınarak kümeleme işlemiyle araçların maksimum dolulukla teslimat yapmasını sağlayan bir çalışmadır. Maksimum doluluk ile beraber azalan araç sayısı, dolaylı yollardan lojistik maliyetlerini de minimize edecektir. Kullanılan DBSCAN yöntemine ürün grubu bazlı yerleştirme kısıtı eklenmiştir. Araç içinde kapasite olduğu takdirde bir sonraki noktayı kümeye dahil ederek DBSCAN algoritması yapısı yeniden şekillenmiştir.

İkinci aşamada doluluk oranı yüksek, minimize edilen araç sayısına sahip kümeler, zaman pencereleri kısıtı altında teslimatlarını gerçekleştirecek şekilde rotalanacaktır.

Burada da kullanılan ve metasezgisel yöntemlerden biri olan Karınca kolonisi algoritmasına, zaman pencereleri kısıtı entegre edilerek, eğer zaman pencereleri dışında bir geç teslimat gerçekleştiriliyorsa ceza maliyetinin oluşmasını ve her karınca turunda bu maliyetler üzerinden sonuçlar elde etmesi sağlanmıştır. Zaman kısıtı dahilinde uygulanmış olan Karınca kolonisi algoritması yaklaşımı ile teslimatlar müşteri talebine uygun rotalanmış olacaktır.

Bu çalışma, veri seti büyük olan ve zaman pencereleri bulunan araç rotalama problemleri için yeni bir çözüm önerisi yaklaşımı sunmaktadır.

2. ARAÇ ROTALAMA PROBLEMLERİ

Lojistik maliyetlerin azaltılması hedefi dolayısıyla firmalar araç rotalama problemlerini etkili yöntemlerle çözümlenmeye çalışmışlardır. Araç rotalama algoritmaları sayesinde gerçek durumdaki belirli kısıtlar sağlanırken dağıtım maliyetleri en küçüklenmeye çalışılır. Araç ortalama problemlerinde kısıtlar talep, kapasite, süre ve zaman pencereleri gibi kısıtları içerebilir, depo sayısı, araç sayısı, zaman penceresinin esnek ya da sıkı olması gibi durumlar problemlerin türlerini belirlemektedir.

Kesin çözüm algoritmaları küçük ölçekli problemleri başarıyla çözebilirken, büyük ölçekli problemlerin çözümü için sezgisel/metasezgisel yöntemlere başvurmak gerekmektedir (Cordeau vd., 2005; Crevier vd., 2007).

Her bir araç, rotasını bir depodan başlatmakta ve tüm müşterileri ziyaret ettikten sonra tekrar aynı depoya geri dönmektedir. Filodaki her aracın belirli bir yük taşıma kapasitesi bulunmaktadır ve tüm araçların kapasiteleri birbirine eşittir. Ayrıca, araçlar benzer maliyetlere (sigorta, bakım vb.) sahiptir, yani araçlar arasında farklılık göstermezler. Araç kapasiteleri ve müşteri talepleri önceden bilinmektedir ve bu bilgiler deterministik olarak sağlanmaktadır (Kumar ve Panneerselvam, 2012).

2.1. Kapasite Kısıtlı Araç Rotalama Problemleri

Bu problem türünde, n adet aracın C sabit kabul edilen kapasite kısıtı altında rotalarının planlanması problemidir. Rotada planlanan müşterilerin siparişleri bölünemez ve sipariş toplamları araç kapasitesini geçmeyecek şekilde rotalanmalıdır.

Her müşteri bir kez ziyaret edilir ve rotanın başlangıç ve bitiş noktası aynı noktadır.

2.2. Topla-Dağıt Araç Rotalama Problemleri

Müşterilere sadece teslimat yapmak değil, aynı zamanda onlardan da teslim alınacak olan ürünlerin teslimatının yapılması problemidir, literatürde daha çok dolu ürünlerin

birakılıp boş kutuların alınması şeklinde malzeme planlanması sürecindeki çalışmalarda uygulanmaktadır.

Dağıtım ve toplama işlemleri eş zamanlı yapılabileceği gibi tek veya iki araçla farklı zamanlarda da yapılabilir (Hoff, Gribkovskaia, Laporte ve Lokketangen, 2009)

2.3. Çok Depolu Araç Rotalama Problemleri

Birden fazla deponun bulunduğu gerçek hayat problemlerine uygun araç rotalama problemleridir. Her araç başladığı depo noktasına tur sonunda dönmektedir.

2.4. Bölünmüş Talepli Araç Rotalama Problemleri

Gerçek hayat problemlerine uygun olabilecek şekilde, maliyet enküçülenmesi hedefi ile bir müşteriye birden fazla araç teslimat gerçekleştirerek müşteri siparişinin bölünebildiği problem türüdür.

2.5. Stokastik Araç Rotalama Problemleri

Stokastik müşteriler, her i müşterisinin varlığı için p_i olasılığı vardır, $1 - p_i$ olasılığıyla müşteri yoktur.

Stokastik talepler, her müşterinin talebi q_i , rassal bir değişkendir.

Stokastik zamanlar ise, servis zamanları s_i ve dolaşım zamanları t_{ij} rassal değişkenlerdir (Yılmaz, 2008).

2.6. Zaman Pencere Araç Rotalama Problemleri

Zaman pencere araç rotalama problemlerinde, hedef noktaların belirlenmiş ziyaret etme süreleri bulunmaktadır. Verilen başlangıç ve bitiş süreleri arasında ziyaret edilebilir, problem türüne göre erken ya da geç ziyaret etme durumlarında ceza fonksiyonu uygulanabilir.

Esnek zaman pencere araç rotalama problemlerinde, hedef noktaya belirli bir zaman penceresi dışında da hizmet verilebilir ancak bu durumda bir ceza maliyeti uygulanacaktır.

Sıkı zaman pencereleli araç rotalama probleminde ise araç hedef noktaya söz konusu zaman penceresinin başlangıcından önce ulaşmış ise beklemek zorundadır, eğer zaman penceresinin bitişinden itibaren bir sürede ulaşmış ise hizmet verememektedir.

2.6.1. Zaman pencereleli araç rotalama problemi matematiksel modellenmesi

Zaman pencereleli araç rotalama problemi (ZPARP) türleri için probleme ait kısıtlara göre yeni matematiksel ifadeler modele eklenebilir ya da modelden çıkarılabilir.

ZPARP'nin matematiksel formülasyonu aşağıda gösterilmiştir (Özkök,2021).

Kümeler;

$C\{1,2,3,\dots,n\}$ Müşteri Kümesi

$N\{0,1,2,3,\dots,n\}$ Düğüm Kümesi – Düğüm 0 depo noktasıdır

$V\{1,2,3,\dots,v\}$ Araç Kümesi

Karar değişkenleri;

$$X_{ijk} = \begin{cases} 1, & k \text{ nolu araç } i \text{ müşterisinden } j \text{ müşterisine gidiyorsa} \\ 0, & \text{aksi takdirde} \end{cases}$$

t_i i . müşteri varış zamanı (i . düğüm zamanı)

w_i i . bekleme süresi (i . müşteriye erken ulaşma durumunda bekleme süresi ya da i . düğümün servise hazır olması için geçen süre)

Parametreler;

d_{ij} i . ve j . müşterileri arasındaki uzaklık

a_i i . müşteride servise en erken başlama anı

b_i i . müşteride servise en geç başlama anı

c_{ij} i . ve j . müşterileri arasındaki maliyet

t_{ij} i . ve j . müşterileri arasındaki seyahat süresi

q_i i . düğümünün talep miktarı

e_j j . müşterisinin zaman aralığının başlangıç zamanı

s_i i . müşterisindeki hizmet süresi ($t_i + w_i$)

- l_j j . müşterisindeki son hizmet alma zamanı
 l_0 depo zaman aralığı son tarihi (bir aracın maksimum seyahat süresi)
 Q özdeş araçların maksimum kapasitesi
 k toplam araç sayısı
 n toplam müşteri sayısı
 c_i i . müşteri düğümü
 c_0 merkez depo düğümü

Amaç fonksiyonları;

Minimum

$$\sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^v d_{ij} * X_{ij} \quad (3.1)$$

$$\sum_{k=1}^v \sum_{j=1}^n X_{0jk} \quad (3.2)$$

Şu kısıtlara göre;

$$\sum_{k=1}^v \sum_{j=1}^n X_{0jk} \leq 1, \forall k \in V \quad (3.3)$$

$$\sum_{j=1}^n X_{0jk} = \sum_{j=1}^n X_{0jk} \leq 1, \forall k \in V \quad (3.4)$$

$$\sum_{k=1}^v \sum_{j=0, j \neq i}^n X_{ijk} = 1, \forall i \in C \quad (3.5)$$

$$\sum_{k=1}^v \sum_{i=0, i \neq j}^n X_{ijk} = 1, \forall j \in C \quad (3.6)$$

$$\sum_{k=1}^n (m_i * \sum_{j=0, i \neq j}^n X_{ijk}) \leq 0, \forall k \in V \quad (3.7)$$

$$\sum_{i=0}^n \sum_{j=0, j \neq i}^n X_{ijk} (d_{ij} + s_i + w_i) \leq l_0, \forall k \in V \quad (3.8)$$

$$t_0 = w_0 = s_0 = 0 \quad (3.9)$$

$$t_i + w_i + s_i + d_{ij} = t_j, \quad \forall (i, j) \in N, \quad i \neq j, \quad \text{if } X_{ijk} = 1 \quad (3.10)$$

$$w_i = \max\{e_i - t_i, 0\}, \forall i \in C \quad (3.11)$$

$$e_j \leq (t_j + w_j) \leq l_j, \quad \forall j \in C \quad (3.12)$$

$$X_{ijk} \in \{0,1\}, \forall k \in V, \forall (i, j) \in N \quad (3.13)$$

$$t_i \geq 0, \forall i \in C \quad (3.14)$$

$$w_i \geq 0, \forall i \in C \quad (3.15)$$

Denklem (3.1) toplam alınan yolu minimum yapmayı hedefleyen amaç fonksiyonudur.

Denklem (3.2) minimum araç sayısını hedefleyen kısıttır.

Denklem (3.3) depodan rotasına başlayan araç sayısının en fazla v olmasını sağlar, araç kullanımını kontrol etmek ve gereksiz araç kullanımını önlemek için bir yöntem sağlar.

Denklem (3.4) her bir rotanın aynı depodan başlayıp aynı depoda bitmesini sağlar.

Denklem (3.5) ve (3.6) ile her bir müşteriye tek bir aracın hizmet edebileceği şartını sağlar. Bu denklemler, her bir müşteriye yalnızca bir aracın atanmasını sağlar ve çakışan atamaları önler.

Denklem (3.7) bir rotadaki müşterilerin toplam talebi araç kapasitesini aşmamalı kısıtını sağlar.

Denklem (3.8)-(3.12) hizmet süresini de dahil ederek belirlenen zaman pencerelerinde hizmet verme kısıtlarıdır.

Denklem (3.8) maksimum seyahat süresi kısıtıdır. Her bir aracın toplam seyahat süresinin belirli bir değeri aşmamasını sağlar.

Denklem (3.9) i . düğümünün 0 olması yani rotanın ilk adresi olan depo için kullanılacak karar değişkenlerini tanımlar.

Denklem (3.10) ise eğer k nolu araç i . müşterinden j . müşteriye gidiyorsa, i . müşterisinden j . müşterisine giden süreyi gösteren kısıttır.

Denklem (3.11) aracın müşteriye zaman penceresinin başlangıç değerinden erken gelmesi durumunda bekleme süresinin hesaplanması kısıtıdır.

Denklem (3.12) her müşteriye ait hizmetin müşteriye ait zaman aralığında verilmesini sağlayan kısıttır. Denklem (3.13)-(3.15) karar değişkenlerinin alabileceği değer aralıklarını tanımlar.

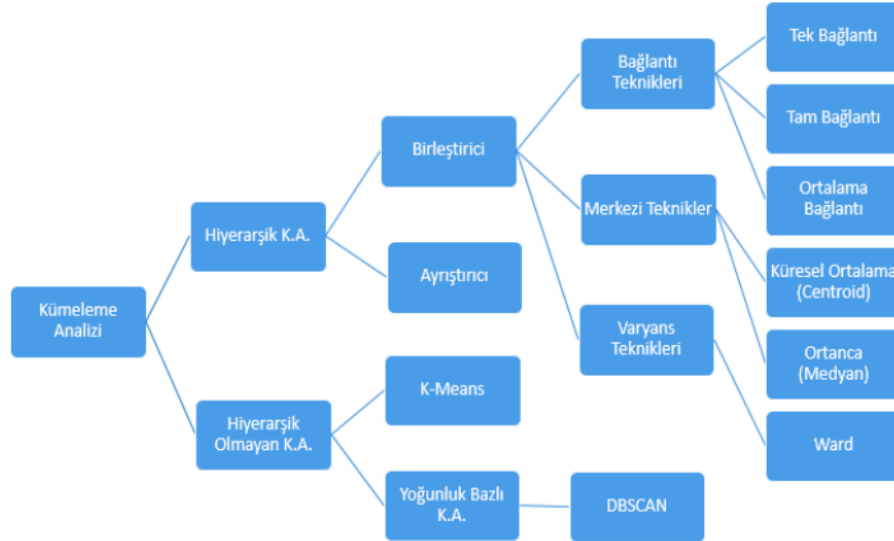
3. YÖNTEM

3.1. Kümeleme Algoritmaları

Kümeleme analizi, veriler içerisinde benzerlik oranları en yüksek olan verileri gruplayarak veri kümeleri oluşturan bir analiz yöntemidir. Analizde amaçlanan durum her kümenin birbirinden farklı, küme elemanlarının ise benzer ya da eşit olmasıdır.

Literatürde kümeleme algoritmalarının çok fazla sayıda olmasının nedeni, veri setlerinin yapısına (düzenli ya da gürültülü), veri setlerinin büyüklüğüne, algoritmanın kümeyi oluşturma yapısına (birimden bütüne, bütünden birime) ve daha birçok etmene bağlı olmasıdır. Bu nedenle keskin hatlarıyla belirlenmiş bir kümeleme algoritması gruplaması yerine kümeleme yöntemlerini en genel haliyle Hiyerarşik ve Hiyerarşik olmayan kümeleme metotları olarak iki başlıkta gruplayabiliriz.

Kümeleme analizinin birçok çeşidi olmasıyla beraber bu çalışmada değinilecek olan yöntemleri Şekil 3.1'deki gibi göstermek mümkündür (Aydın, 2022).



Şekil 3.1. Kümeleme Algoritmalarının Sınıflandırılması

3.1.1. Hiyerarşik kümeler

Hiyerarşik kümeleme, elemanlarını hiyerarşik bir yapıda gruplandıran bir kümeleme yöntemidir. Birleştirici ya da ayrıştırıcı olmak üzere iki temel yaklaşıma sahiptir. Birleştirici yöntemlerde her eleman başlangıçta benzerliklerine göre gruplanmış bağımsız birer kümedir. Her bir adımda benzer kümeler bir araya gelerek kümeleri oluşturur.

Ayrıştırıcı yöntemlerde ise tüm elemanlar bir kümede başlar ve benzer olmayan elemanlar küme içinden ayrıştırılarak yeni kümelemeler oluşturur. Ek bağlantı (single linkage), tam bağlantı (complete linkage), ortalama bağlantı (average linkage) gibi kriterler alt başlıkları olarak incelenebilir. Bu kriterler kümelerin birleşim aşamasında önem taşıyan ve sonucu etkileyen faktörlerdir.

Hiyerarşik kümeleme yöntemlerinde küme sayısının önceden bilinmesi gerekmektedir ve veri setlerinin küçük olduğu durumlar için daha uygun yöntemlerdir.

Tek Bağlantı (Single linkage) kümeleme tekniğinde elemanlar arasında benzerlik ya da uzaklık matrisi ele alınarak benzer noktalar kümelendir. Tam Bağlantı (Complete linkage) kümeleme tekniğinde ise benzerlik ya da uzaklık matrisinde en uzak elemanlar dikkate alınarak kümeleme yapılır. Ortalama bağlantı (Average linkage) kümeleme tekniği ise, elemanların uzaklıklarının ortalamasına göre kümelendiğini gerçekleştirir, bu nedenle gürültü noktalarından diğer iki tekniğe göre daha toleranslı bir teknik olacaktır.

Küresel Ortalama (Centroid) kümeleme tekniği ise, kümeyi oluşturacak elemanların küme merkezlerinin hesaplandığı ve bu merkezler ele alınarak benzerlik ya da uzaklık matrisi kullanarak kümelerin oluşturulduğu bir yöntemdir. Bu yöntemde, her eleman başlangıçta bir kümedir. Eleman sayısı kadar küme vardır ve her küme bir merkez (Centroid) ile gösterilir.

Kümeler birleştikçe yeni merkez nokta, dahil edilen kümelerin merkez noktalarının ağırlıklı ortalaması ile bulunur. Tek bir küme oluşuncaya kadar işlem tekrarlanır. Yoğun dağılıma sahip ve merkez noktaların net olarak bilindiği veri setleri için uygun bir yöntemdir.

Ortanca (Medyan) kümeleme tekniği ise, Küresel Ortalama kümeleme tekniğinden farklı olarak yeni kümenin merkezinin ortalamaya göre değil, orta noktaya (medyan) göre bulunmasıdır.

Küresel Ortalama kümeleme tekniğinde yeni küme merkezi ağırlıklandırılarak bulunduğu için kümelemek istenen veri setine uygun olmayabilir, eşit ağırlıkta orta noktalarının baz alınarak kümelenebilmesi için geliştirilen bir tekniktir.

Ward tekniğinde ise uzaklık ya da benzerlik matrisi kullanılarak birleştirilen kümelerin homojenliğini ölçmek için varyans tabanlı bir kriter kullanılır. Birleştirme sonucunda küme içi varyansın artışı minimize edilir, kümeler arası heterojenlik maksimize edilir.

3.1.2. Hiyerarşik olmayan kümeler

Hiyerarşik olmayan kümeleme algoritmaları, kümeleme işlemini hiyerarşi olmadan, kümeye dahil olacak elemanları benzerlik ya da uzaklık matrislerine göre kümelemektedir. Kümeye dahil olacak “n” adet eleman sayısından daha az olacak şekilde “k” küme adedi belirtilmelidir.

3.1.2.1. K-Ortalamlar (K-Means) yöntemi

Veri noktalarının uzaklık ölçütüne göre en yakın kümeye dahil olmasını sağlayan yöntemdir. Küme merkezleri oluşturarak veri noktalarını seçilen merkezler çerçevesinde kümeler. Her iterasyonda küme merkezi güncellenir, veri noktalarını yeniden kümeleyerek nihai kümelere ulaşır.

K-Ortalamlar algoritmasının amaç fonksiyonu:

$$J = \sum_{j=1}^k \sum_{i=1}^n d(x_i, \mu_j) = \sum_{j=1}^k \sum_{i=1}^n \|x_i - \mu_j\|^2$$

Değişkenler:

n = küme elemanı sayısı, $i = (1,2,3,\dots, n)$

k = küme sayısı, $j = (1,2,3,\dots, k)$

$\mu_j = j$. kümenin merkezi

K-Ortalamlar algoritma adımları:

Adım 1: Küme sayısı olan k belirlenir.

Adım 2: k adet nokta seçilerek her biri küme oluşturur.

Adım 3: Geriye kalan $n - k$ adet noktanın küme merkez noktalarına uzaklıkları hesaplanır.

$$d(x_i, \mu_j) = \|x_i - \mu_j\|^2$$

Adım 4: $n - k$ adet noktanın her biri kendine en yakın noktaya atanır.

$$c_j = i: d(x_i, \mu_j) \leq d(x_i, \mu_p), p \neq j, \quad i = 1, 2, \dots, n$$

Adım 5: Küme atama işlemleri için yeniden kalan noktalar ve merkez noktaya uzaklıkları hesaplanır.

$$\mu_j = \frac{1}{c_j} \sum_{i \in c_j} x_i, \forall j$$

Adım 6: Tüm küme noktaları atanana kadar devam eder.

K-Ortalamlar algoritması gürültü değerlerine karşı toleranslı değildir, eğer veri setinde anormal veya aykırı değerler varsa, bu değerlerin ortalamayı etkileme potansiyelinden dolayı küme merkezlerini yanlış konumlandırabilir.

3.1.2.2. K-Medoids yöntemi

K-Ortalamlar algoritmasından farkı, küme merkezlerini veri elemanlarından seçmesidir. Her iterasyonda yeni küme merkezi bir küme elemanı olacak şekilde kümeler güncellenir ve nihai kümelere ulaşılır.

K-Medoids gürültülü veri setlerine K-Ortalamlar algoritmasından daha çok toleranslıdır, K-Medoids gerçek veri noktalarını kullanarak merkezleri seçer ve bu sayede gürültüye karşı daha dirençli olur.

3.1.2.3. Yoğunluk tabanlı kümeleme algoritması (DBSCAN)

Yoğunluk tabanlı kümeleme algoritması (DBSCAN - Density-Based Spatial Clustering of Applications with Noise), veri noktalarının yoğunluk tabanlı olarak gruplandırıldığı bir kümeleme algoritmasıdır. DBSCAN, bir veri noktasının

etrafındaki yoğunluk tabanlı bir bölgeye dayanarak veri noktalarını kümelere atar. Yoğun bölgeleri birleştirirken düşük yoğunluğa sahip bölgeleri gürültü olarak tanımlar.

DBSCAN bazı noktalarda yoğunluk içeren ve gürültü noktaları bulunan veri setleri için kullanılır. Diğer yöntemlerden farklı olarak, uzaklık ya da benzerlik matrisinden çok yoğunluk baz alındığı için gürültü noktaları tespit edebilir.

Küme sayısının başlangıçta belirlenmesine gerekli değildir. Ancak DBSCAN parametreleri küme sayısının seçilmesinde çok büyük bir etmen olduğu için parametrelerin veri setine uygun ve doğru girilmesi önem taşımaktadır. Epsilon ve Minpts parametrelerini inceleyecek olursak;

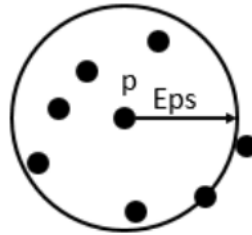
Epsilon (ϵ), bir veri noktasının komşuluk yarıçapını belirleyen değerdir. Bir veri noktası için epsilon değeri, o noktanın çevresindeki diğer noktaları kapsayan bir alanı ifade eder.

Eğer bir veri noktası için, ϵ uzaklığı yarıçapı içerisinde en az MinPts sayısında diğer noktaları içeriyorsa, o nokta çekirdek nokta (core point) olarak kabul edilir. Çekirdek noktalar alanları içerisindeki noktaları da dahil ederek küme oluştururlar.

MinPts ise, bir veri noktasının çekirdek nokta olarak kabul edilebilmesi için en az kaç adet komşu noktası olması gerektiğini belirleyen bir parametredir. Eğer epsilon uzaklığı içerisinde en az Minpts kadar komşu nokta bulunmuyorsa, o veri noktası gürültü nokta (noise point) kabul edilir.

Bir $p \in D$ noktasının $N_\epsilon(p)$ olan epsilon değerleri arasındaki komşulukları Şekil 3.2'deki gibi gösterilir (Aydın,2022):

$$N_\epsilon(p) = \{q \in D \mid \text{dist}(p, q) \leq \epsilon\}$$



Şekil 3.2. Epsilon Değerleri Arasındaki Komşuluklar

Algoritma adımları inceleyelim.

Adım 1: Rastgele daha önce işlem yapılmamış bir nokta seçilir.

Adım 2: Epsilon (ϵ) uzaklığında Minpts değerine eşit ya da büyük sayıda nokta olup olmadığı kontrol edilir. Koşulu sağlıyorsa, seçilen nokta çekirdek noktadır ve “0” etiketini alır. Eğer koşulu sağlamıyor ise, gürültü nokta olarak tanımlanır ve “-1” etiketini alır. Nokta ziyaret edildi olarak işaretlenir.

Adım 3: Eğer bu nokta çekirdek nokta ise, komşu noktalarıyla birlikte bir küme oluşturur.

Adım 4: Rastgele ziyaret edilmemiş bir nokta seçilir. Eğer bu nokta, bir kümeye dahil edilmiş ise, epsilon yarıçapında Minpts adet ya da daha fazla adette nokta olup olmadığına bakılır.

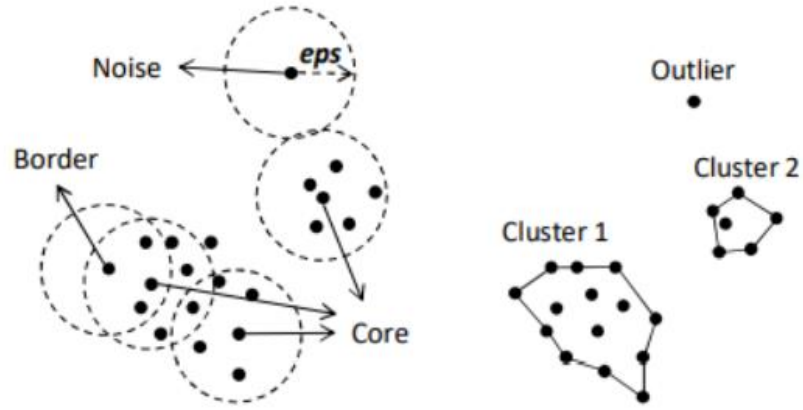
Minpts değerine eşit veya daha fazla nokta var ise, komşu noktaları seçilen noktanın kümesine dahil edilir.

Eğer Minpts değerinden az nokta var ise, bu nokta “Sınır Nokta” olarak adlandırılır, komşu noktaları kümeye dahil edilmez.

Rastgele seçilen nokta “ziyaret edildi” olarak etiketlenir.

Adım 5: Tüm noktalar “gürültü” ve “ziyaret edildi” olana kadar işlem devam ettirilir.

Bu örnek gösteriminde ϵ değeri 1, MinPts değeri 5 olarak alınmış olup, gürültü/aykırı veriden etkilenmeyerek farklı büyüklük ve şekillerde iki küme elde edilmiştir (Rezai,2016). Şekil 3.3’de verilmiştir.



Şekil 3.3. Üç tip nokta ve kümelemesi (Rezai,2016)

DBSCAN algoritmasında küme sayısını, eleman sayılarını etkileyen en önemli adım, parametrelere verilen değerlerdir. Bu nedenle, bu değerlerin veri setine uygun olacak şekilde doğru değerlerde verilmesi gerekmektedir.

Parametrelere değer verme noktasında veriyi iyi analiz edip görselleştirmek yöntemlerden biri olabilir, bu yöntemin dışında deneme-yanılma ile, elde edilen sonuçların analiz edilmesi de uygulanabilecek yöntemlerden biridir. Daha az test verisiyle sonuçların gözlemlendiği istatistiksel analizlerle de tespit yöntemlerinden biri olabilir.

3.2. Karınca Kolonisi Optimizasyonu Algoritması

NP-zor optimizasyon problemlerinin çözümünde kullanılan Karınca kolonisi algoritmaları ilk olarak Marco Dorigo tarafından ortaya atılan, karınca kolonilerinin feromon sağlayarak besin kaynağına en yakın yoldan ulaşmalarından ilham alınarak oluşturulmuş bir algoritmadır.

Feromon, karıncaların yön bulmaları gerektiğinde birbirleri arasında bilgi iletmek amacıyla kullandıkları vücut kimyasallarıdır. İzledikleri rotada feromon izleri karıncalar tarafından güncellenir, bu nedenle feromonun yoğun olması yolun kalitesini gösterecektir, tercih edilme ihtimalini artırır.

Feromon kuvvetlendirme oranı (*alpha*) düğümler arası feromon miktarlarının önem derecesini belirler. Sezgisellik Kuvvetlendirme Oranı (*beta*) düğümler arasındaki mesafenin önem derecesini belirleyen parametredir.

3.2.1. Karınca kolonisi algoritması yapısı

Karınca kolonisi algoritması, problemlerin optimizasyonunda kullanılan Karınca kolonisi algoritmasında kullanılan parametreler;

Karınca Sayısı, karınca kolonisi algoritmasında kullanılan karınca sayısıdır. Karınca sayısı problemin karmaşıklığına bağlı olarak seçilir. Karınca sayısı arttıkça, daha geniş arama yeteneğine sahip olacaktır.

Karınca sayısı, karınca kolonisinin keşif yetenekleri ve optimizasyon potansiyelini etkiler. Fazla karınca miktarı, hesaplama yükünü artırdığı için optimum parametre değerine deney tasarımları ya da deneme-yanılma yöntemleri ile ulaşılabilir.

Feromon miktarı, karıncaların izlediği yollar üzerinde bıraktıkları izler olarak tanımlanabilir.

Feromon güncelleme katsayısı, *tau*, her rotada buharlaşan feromonun hangi oranda yenileneceğini gösterir, bu değer artması algoritma sonuçlarında yakınsama gözlemlenmesine neden olur.

Feromon buharlaşma katsayısı ise, *rho*, her yeni kolonide feromon buharlaşacaktır, bu buharlaşma oranı için verilen değerdir. Düşük bir parametre değeri, karınca turlarının tekrarlanmasına ve arama yeteneklerinin azalmasına neden olacaktır.

Feromon buharlaşma katsayısı, [0,1] aralığında değer almaktadır (Özdağ ve ark, 2012).

alpha parametresi, karıncaların feromon miktarını önemseme derecesidir. Yüksek bir *alpha* değeri, karıncaların feromon izlerine daha fazla odaklanmasını sağlar ve daha fazla feromon biriktiren yolları tercih etmelerini sağlar. *alpha* değeri, feromon miktarının karar sürecindeki ağırlığını belirler. Feromon miktarının önemini gösteren pozitif bir tamsayı (Dereli, Daş, 2010) olarak değer alır.

beta parametresi, karıncaların heuristik bilgiyi (örneğin, uzaklık veya maliyet gibi) ne kadar önemseydiğini belirler. Parametre değeri yükseldikçe, karıncaların heuristik bilgiyi daha fazla dikkate almalarını sağlar ve kısa veya ucuz yolları tercih etmelerini sağlar. Beta değeri, heuristik bilginin karar sürecindeki ağırlığını belirler. β seçilebilirlik parametresinin önemini gösteren bir alt-parametre olarak 1 ile 10 arasında (Dereli, Daş, 2010) değer almaktadır.

3.2.2. Karınca turunun matematiksel modellenmesi

Karıncanın algoritmasında ilk adımda karınca sayısı belirlenir. Ardından her bir karınca rastgele bir düğüme yerleştirilir ve tüm düğümleri tek tek ziyaret ederek turunu tamamlar. Her bir karıncanın mevcut düğümden bir sonraki düğüme gitme olasılığını hesaplamak için matematiksel formül Denklem 1'de ifade edilir (Özdemir, 2008; Serin, 2009).

$$P_{i,j}^l = \frac{[\tau_{ij}]^\alpha [n_{ij}]^\beta}{\sum_{l \in N_i^l} [\tau_{il}]^\alpha [n_{il}]^\beta}, \text{ eğer } j \in N_i^l$$

Burada,

$P_{i,j}^l$:	l karıncasının i düğümünden j düğüme geçme olasılığı
τ_{ij}	:	i ve j düğümleri arasındaki feromon değeri
n_{ij}	:	i ve j düğümleri arasındaki sezgisel değeri
<i>alpha</i>	:	feromon katsayısı
<i>beta</i>	:	sezgisel katsayısı
N	:	düğümler kümesi

Bu formül ile oluşturulan feromon izleri ve heuristik bilgi birleştirilerek bir sonraki düğümün olasılığı belirlenir. Heuristik bilgi, düğümler arasındaki uzaklık, maliyet veya diğer faktörleri temsil ederken feromon karıncalar tarafından oluşturulan ve düğümler arası iletişimi sağlayan kimyasal izlerdir. Feromonun belirli buharlaşma ya da yenilenme miktarlarıyla izinin ve heuristik bilginin ağırlıklı kombinasyonu hesaplanır ve yüksek değere sahip düğümlere karıncalar daha fazla eğilim gösterir.

Böylece, her bir karınca sıradaki düğümünü hesaplayarak ve rastgele olarak düğümleri ziyaret ederek yolunu tamamlar.

3.2.3. Karınca kolonisi algoritması adımları

Adım 1: Karınca sayısı belirlenerek düğümlere rastgele olarak yerleştirilir.

Adım 2: Her bir karınca belirtilen *alpha* ve *beta* değerlerine göre düğüm rotalarını oluştururlar.

Adım 3: Her bir karıncanın alacağı rota mesafesi hesaplanır.

Adım 4: Karıncaların rota mesafelerine göre yollardaki feromon değerleri artırılır.

Adım 5 : Feromon buharlaşma değeri kadar yollardaki feromon değeri azaltılır.

Adım 6: Belirtilen iterasyon sayısına ulaşılan kadar veya hedeflenen değere ulaşıncaya kadar Adım 2 tekrarlanır.

3.3. Yanıt Yüzey Yöntemi Ve Deney Tasarımı

“Denemelerin Optimum Koşullara Ulaşması” ismi ile 1951 yılında Box and Wilson tarafından geliştirilen Yanıt Yüzeyi Yöntemi, çıktı değişkeninin girdi parametreleri tarafından nasıl etkilendiği ve çıktının optimizasyonunu gerektiren uygulamalarda modelleme ve analizleme için kullanılan matematiksel ve istatistiksel bir tekniktir.

Deneme setinin hazırlanması, modelin oluşturulması (regresyon analizi), varyans analizi (ANOVA) ve optimizasyon Yanıt Yüzeyi metodunun başlıca kısımlarını oluşturmaktadır.

İlk aşaması, girdi ve çıktı parametrelerinin tanımlanması ve her bir girdiye ait seviyelerin belirlenmesidir. Ardından girdi değerlerinin çıktı üzerindeki etkisini gözlemlemek ve aralarındaki ilişkiyi oluşturacak olan bağıntıyı oluşturmaktır.

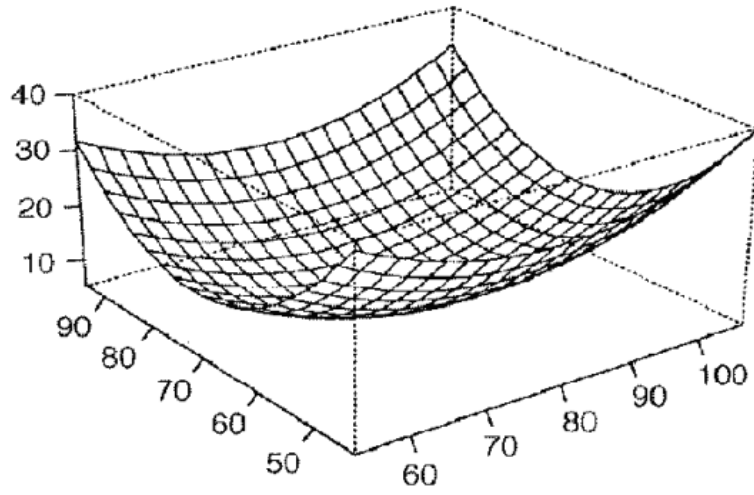
Optimum noktaya yaklaşıldıkça, yanıt yüzeyindeki eğim artacaktır, eğriliğin tahminlenebilmesi için bazı analiz metotlarıyla entegre olarak optimum noktaya ulaşılır.

Örneğin; bir kimyasal sürecin sonucu olan *Y* çıktısına ait sıcaklık (X_1) ve basınç (X_2) gibi iki bağımsız değişken olması durumunda formülizasyon,

$$Y = f (X_1, X_2)$$

ile ifade edilebilir. Bu bağıntı, sıcaklık ve basınç değerlerinin sonuç Y'yi nasıl etkilediğini ifade eder.

Bu yanıt yüzeyinde, her X_1 ve X_2 değerine bir Y değeri karşılık gelmektedir ve X_1 apsiste , X_2 ordinatta olmak üzere Y , bu düzleme dik noktalanarak üç boyutlu olarak çizilmektedir. Oluşturulan grafik “yanıt yüzeyi” olarak adlandırılır. Şekil 3.4'te yanıt değişkeni Y ile bağımsız değişkenler X_1 ve X_2 arasındaki ilişki grafiksel olarak gösterilmektedir (Çiftçi, 2011).



Şekil 3.4. Y, X1 ve X2 değişkenlerine ait Yanıt Yüzeyi

Etken sayısı çok olduğunda yanıt yüzeyi modelinin karmaşıklaşması, tasarım seçimi ve veri toplama sürecini zorlaştırır. Bu durumda, çok sayıda etkenin aynı anda değerlendirilmesi ve etkileşimlerinin anlaşılması önem kazanır. Bu gibi durumlarda, MBT (Merkezi Bileşik Tersinir Yüzey Tepki Optimizasyonu) ve Box-Behnken Tasarımı gibi yöntemler kullanılabilir (Çiftçi,2011).

Box-Behnken Tasarımı yanıt yüzeyi analizinde sık kullanılan bir deney tasarımı yöntemidir. Bu tasarımda, faktörlerin seviyeleri belirlenir ve deney noktaları bu seviyelerde olacak şekilde düzenlenir. Bu tasarım, deney sayısını minimize ederken etkileşimleri de hesaba katarak yanıt yüzeyi modellerinin oluşturulmasını sağlar.

Merkezi Bileşik Tasarımlar (Central Compozite Design), çok faktörlü yanıt yüzeyi modelleri geliştirmek için bir yöntemdir. Bu yöntemde deneyler, faktörleri seçmek, etkileşimleri belirlemek ve en uygun yanıt koşullarını bulmak için tasarlanır ve analiz edilir. MBT, faktörler arasındaki etkileşimleri dikkate alarak yanıt yüzeyinin daha doğru bir modelini oluşturur.

Cevap yüzey yönteminin daha az deney gerektirirken daha fazla bilgi çıktısı sağlar. Değişkenlerin birbirleri arasındaki etkileşimleri inceleme fırsatı verir. Sisteme ait matematiksel bir model tanımlayarak bağımlı ve bağımsız değişkenler arasında ilişkiyi açıklar.

Cevap yüzey yöntemi doğrusal olmayan sistemleri modellemede başarılı bir yöntem olmayabilir. Simetrik olmayan fonksiyonlar (hiperbolik ya da çan eğrisi) polinomlar ile modellenememektedir (Kasap, 2017).

3.3.1. Deneyin analizi

Yanıt Yüzeyi analizinde elde edilen ve yorumlanacak oldukça fazla sayıda parametre vardır. Değişkenlerin doğasına ve analiz yapılacak sistemin özelliklerine bağlı olarak değişkenler “sürekli” ya da “kesikli” olarak tanımlanır.

Eğer değişkenler belirli değerler arasında kesikli adımlarla değişiyorsa, yanıt yüzeyi analizi için “kesikli” değişkenler kullanılır.

Eğer değişkenler sürekli bir aralıkta değer alabiliyorsa (mesela sıcaklık, basınç gibi), bu değişkenler yüzeyi analizi için “sürekli” değişkenler kullanılır ve deney tasarımı sürekli aralıklardaki noktaları içerecek şekilde yapılmalıdır.

Bu kısımda çalışmada ele alınan kapsam kısıtında deney analizinin yorumlanması anlatılacaktır.

R^2 , varyans analizi sonucunda elde edilen bu değer, veri noktalarının modele ne kadar iyi uyduğunu ve varyasyonun ne kadarının model tarafından açıklandığını temsil eder. 0-1 aralığında değerler almaktadır.

R^2 , değeri arttıkça, oluşturulan modelin veri ile uyumlu olduğunu ve varyasyonun büyük kısmının oluşturulan deney tasarımı ile açıklandığını gösterir. Aynı zamanda tahminleme gücünün yüksekliği ve modelin kullanılabilirliğini de açıklar.

Deney sonucunun güvenilirliđi için güven derecesi ve P deđeri kullanılabilir. Güven derecesi, istatistiksel analizlerde sonuçların güvenilirliđini deđerlendirmek ve dođru kararlar vermek için önemlidir. Yüksek bir güven derecesi, sonuçların daha güvenilir olduđunu gösterirken, düşük bir güven derecesi, belirsizlik içerdikini ifade eder. Eđer güven derecesi %95 olarak sečilmiř ise, hata olasılıđı genellikle $\alpha = 0,05$ olarak belirlenir. Bu, yanlışlıkla yanlış sonuçlara ulaşma olasılıđının %5 olduđu anlamına gelir.

F , bir regresyon modelinin anlamlılıđını deđerlendirir, modelin açıkladıđı varyansın hata varyansına oranı olarak tanımlanabilir.

F , bir hipotez testi sonucunda hesaplanır, H_0 hipotezi ve H_1 alternatif hipotez tanımlanır. Sıfır hipotezi, bađımsız deđişkenlerin modelde anlamlı katkısının olmadığı anlamına gelir. Alternatif hipotez ise en az bir bađımsız deđişkenin modelde anlamlı bir şekilde katkı sağladığını ifade eder. F deđeri yükseldikçe, sıfır hipotezi reddedilir ve modelin anlamlı olduđu kabul edilir.

Yapılan deney tasarımlarında F deđerini modelin anlamlılıđını deđerlendirmek için kullanılırken, güven derecesini %95 olarak hipotez testi sonucunu yorumlamak ve karar vermek için kullanacađız.

3.3.2. Deneyin optimizasyonu

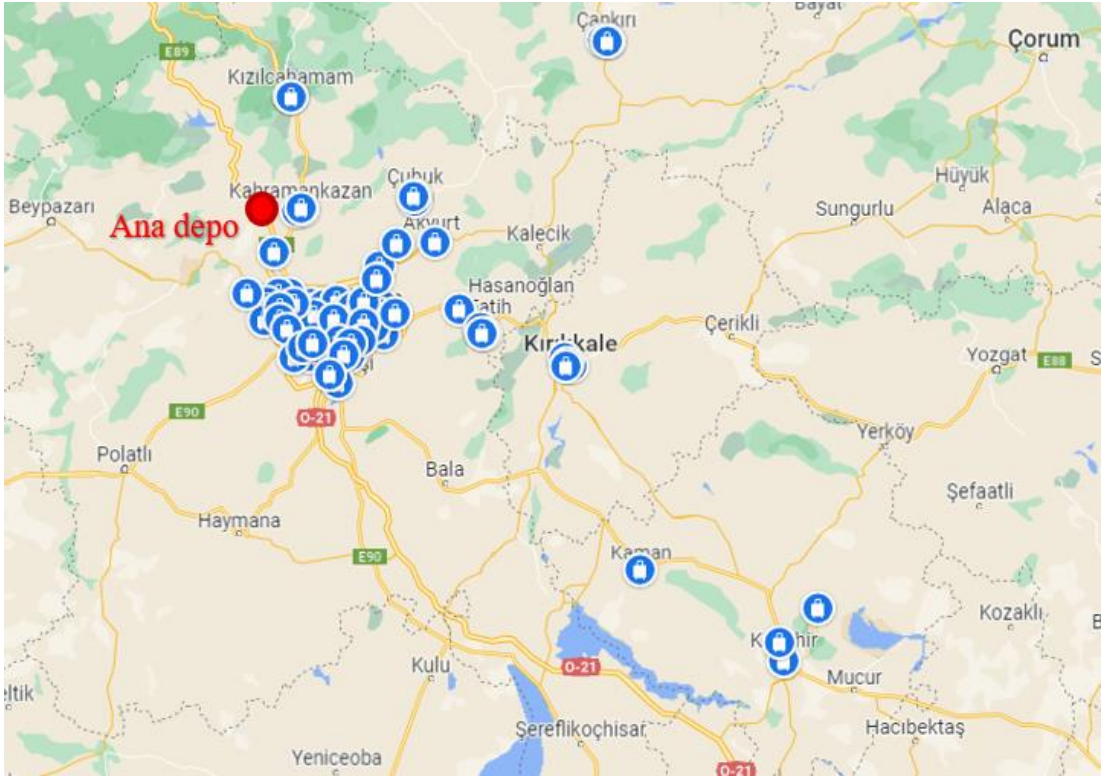
Çıktı parametrelerinin minimum, maksimum ya da belirli bir hedefte istenmesi durumunda, girdi parametrelerinin etkisi incelenir. Çekicilik (Desirability) fonksiyonu, bu aşamada, tüm çıktı sonuçlarını analiz eden 0-1 aralıđında deđer alan bir metottür. Çekicilik fonksiyonunun deđeri istenilen çıktı kriterlerine yaklaşma oranını verecektir, deđer 1'e yaklařıkça belirlenen kriterlerin sağlandıđını gösterir.

4. UYGULAMA

Çalışma, bir lojistik firmasına ait beyaz eşya dağıtım deposundan Ankara ve çevre iller olmak üzere ev ya da bayilere yapılacak teslimatların, randevu saati kısıtı altında dağıtım rotasının planlanmasıdır.

Çevre iller kapsamındaki iller Kırıkkale, Çankırı ve Kırşehir'dir.

Veri seti olarak bir günlük sipariş ve teslimat adresleri baz alınmıştır, teslimat adresleri ve ana depo konumu Şekil 4.1'de verilmiştir.



Şekil 4.1. Teslimat Adresleri ve Ana Depo Konumu

Problem mevcut depoda gerçek zamanlı bir problem olmakla birlikte, sevkiyatlar planlama personelinin deneyim ve görüşlerine dayanarak planlamakta, sistemsal bir rotalama çalışması bulunmamaktadır.

Problemimizde, Ankara ve çevre illerde 75 adet ev veya bayi adresi için teslimat siparişleri bulunmaktadır. Teslimat siparişlerinin yer aldığı ana depo Ankara Kahramankazan'da yer almaktadır.

Bayiler ve 76 numaralı ana depo için oluşturulmuş uzaklık matrisi ekte verilmiştir. Uzaklık matrisinde uzaklıklar km birimi olarak verilmiştir.

Müşteri tarafından verilen en önemli kısıt zaman pencereleri olmakla beraber zaman pencereleri dışında hizmet verilebilir ancak bu durumda ceza maliyeti uygulanmaktadır. Birim geç kalma ceza maliyeti, erken gidilen sürenin ya da geç kalınması durumunda geç kalınan sürenin on katı birim olarak belirlenmiştir.

Geç ya da erken ziyaret durumuna izin verilmesi nedeniyle, problem türü literatürdeki "Araç Rotalama Problemleri" başlığı altında "Esnek Zaman Pencereli Araç Rotalama Problemi" olarak yer almaktadır.

Bayilerin teslimat noktalarına sunacağı servis süresi eşit ve 20 dakika olarak ele alınmıştır. Her ev ya da bayi adresine ait teslimat zaman pencereleri bulunmaktadır.

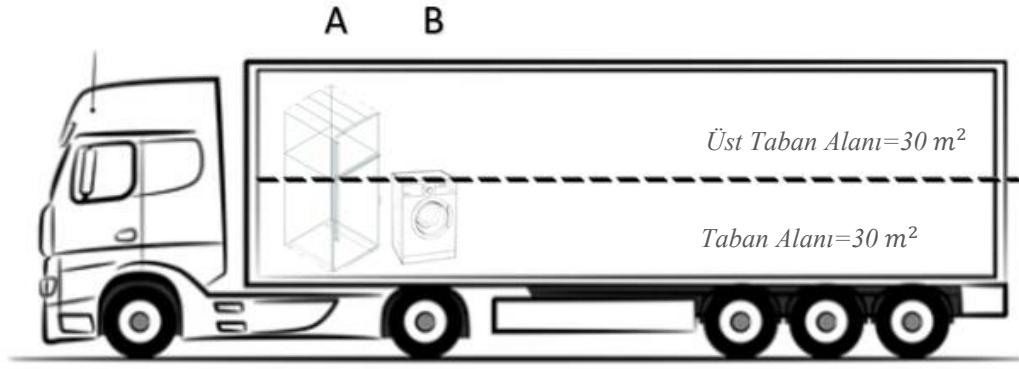
Zaman Pencereleri sabittir ve her teslimat için satış bayilerinden müşterilere aşağıdaki zaman pencerelerinden biri için randevu oluşturulmaktadır. Zaman pencereleri Tablo 4.1'de görülmektedir.

Tablo 4.1. Teslimat Başlangıç ve Bitiş Zamanları

Teslimat Başlangıç	Teslimat Bitiş
10:00	12:00
10:00	14:00
12:00	14:00
12:00	16:00
14:00	16:00
14:00	18:00
16:00	18:00

Teslimat yükü ve araç kapasitesi taban alanına göre hesaplanmaktadır. Sevkiyatta yer alan ürünlerin taban alanları toplamı, araç taban alanını geçmeyecektir.

Araç için alt ve üst taban tasarımı Şekil 4.2'de yer almıştır.



Şekil 4.2. Araç İçi Alt ve Üst Taban Alanı Gösterimi

Aracın taban ve üst taban alanı problem boyunca sabit ve $30 m^2$ 'dir, sipariş toplamı $30 m^2$ 'yi geçtiği durumlarda kalan siparişlerin bir diğer araca yükleme yapılması gerekmektedir.

Aracın üst taban alanı, tabana yerleştirilen ancak yüksekliği fazla olmayan ve üst üste de yerleştirilebilen ürün grupları için hayali olarak tasarlanmış alandır.

Teslimatı yapılacak sipariş listesindeki ürün grupları Tablo 4.2'deki gibi gösterilmiştir.

Buzdolabı grubu siparişler A-Grubu siparişler olarak adlandırılmıştır. A-Grubu ürünler aracın hem taban hem de üst taban alanını kaplayan ürün gruplarıdır.

Tablo 4.2. Sipariş Gruplandırması

	A-Grubu Sipariş	B-Grubu Sipariş
Ürün Kategorileri	Buzdolabı	Bulaşık Makinası Çamaşır Makinası Kurutma Makinası
Taban Alanı (m^2)	$1.3 m^2$	$0.36 m^2$
Üst taban Alanı (m^2)	$1.3 m^2$	$0.36 m^2$

B-grubu ürünler ise çamaşır makinası, kurutma makinası ve bulaşık makinası ürün kategorilerini içerir. Bu ürün grubundakiler A grubu ürünlerinden farklı olarak araç içerisinde maksimum 2 adet olacak şekilde üst üste yerleştirilebilirler.

A grubu buzdolabı siparişlerinin taban alanı $1.3 m^2$ alınırken, B grubu ürünlerin siparişlerinde taban alanları $0.36 m^2$ olacak şekilde varsayılmıştır. Model çeşitlerine göre taban ölçüleri değişmesine rağmen problem kısıtı olarak gerçek değerlerin ortalamasından elde edilen değerler sabit değerler olarak varsayılmıştır.

Problem çözümünde “Önce Kümele Sonra Rotala” yaklaşımı kullanılmıştır. Kümeleme aşamasında problem yapısı ve veri türüne uygun olarak seçilen DBSCAN kümeleme algoritmasının uygulanması anlatılacaktır.

İkinci aşamasında ise elde edilen kümelerin her biri için teslimat noktalarının “zaman kısıtları” da ele alınarak Karınca Kolonisi Algoritması yöntemi ile rotalanması yer alacaktır.

4.1. Teslimat Noktalarını Kümeleme Aşaması

Probleme ait veri seti incelendiğinde, teslimat adreslerinin bir bölgede yoğunluk gösterdiğini ancak uç noktalarda da teslimat yapılması gereken noktalar bulunduğunu Şekil 4.1’de görmekteyiz.

Ana depo’nun Ankara bölgesinde yer almasıyla birlikte, sipariş listeleri incelediğinde teslimat adreslerinin sipariş sayısı, bayi sayısı, nüfus vb. etmenlerin de etkisiyle her zaman Ankara bölgesinde yoğunluk oluşturduğu gözlemlenmiştir.

Bu nedenle probleme en uygun algoritmanın yoğunluk tabanlı kümeleme algoritması (DBSCAN) olduğu sonucuna ulaşılmıştır.

Küme sayısı problem başlangıcında bilinmemektedir. Gürültü noktaların da dahil edilmesiyle yoğunluk tabanlı kümelemeye uygun bir yöntem olan DBSCAN algoritması uygulanmıştır.

4.1.1. Araç içi kapasite ve yerleşim kısıtlarının algoritmaya dahil edilmesi

Problem rotası depodan başlayıp depoda bitecek olan aynı zamanda bir “Gezgin Satıcı Problemi”dir.

Bu nedenle kümeleme sonunda elde edeceğimiz her küme ana depodan başlayacak ve ana depoda rotasını tamamlayacak bir araç olarak tasarlanmıştır. Bu nedenle k değişkeni küme sayısını ifade ediyorken aynı zamanda araç sayısını da ifade etmektedir.

Oluşacak her bir kümenin bir aracı temsil etmesi, uygulanacak kümeleme algoritmasının yanında kümeleme işlemine bir kapasite kısıtının da dahil edilmesini gerektirmektedir. Algoritma adımları Tablo 4.3'te verilmiştir.

Tablo 4.3. Kümeleme Algoritması

Oluşturulan Algoritma

BAŞLA

Uzaklık matrisi değerlerini Excel'den oku

$taban_alani=0$; $ust_taban_alani=0$; $max_weight_per_cluster=30$;
 $ust_cap=0$;

Sipariş adetlerini Excel'den oku.

ϵ ve MinPts değeri tanımla

Tüm noktaların etiketlerini-1 olarak tanımla

for $i=1$ 'den teslimat noktası sayısına kadar

ϵ değerine eşit ya da daha kısa mesafede bulunan noktaları bul

if minimum MinPts kadar nokta bulunduran nokta varsa

noktaların etiketlerini 0 olarak ata.

end if

end for

$k = 0$, küme tanımla

for $i=1$ 'den teslimat noktası sayısına kadar

Tablo 4.3. (Devamı) Kümleme Algoritması

```
if i noktasının etiketi 0 ise
    küme sayısını 1 arttır
        ExpandCluster fonksiyonunu çalıştır
    end if
end for
for i=1'den teslimat noktası sayısına kadar
    if i noktasının etiketi -1 ise
        en yakın kümenin etiketini ata
    end if
end for

function ExpandCluster
```

BİTİR

Algoritmada ilk olarak ekte yer alan ana depo ve teslimat noktalarının uzaklık matrisi tanımlanır.

Ardından aracın taban ve üst taban alanı başlangıçta 0 kabul edilerek, maksimum kapasite, $\max_weight_per_cluster = 30$ olacak şekilde tanımlanır.

B grubu ürünlerinden siparişte tek bir adet olması durumunda, araca yerleştirilmiş o ürünün üst kısmında bir B-grubu ürünün yerleşebileceği kadar bir adet kapasite oluşmuş olur. Sıradaki sipariş eğer B grubu ürünü ise tabana yerleştirilmek yerine daha önce yerleştirilmiş olan ürünün üzerine yerleştirilecektir. Bunun için üst kapasite, ust_cap , başlangıçta 0 olacak şekilde tanımlanır.

A grubu ürünler *buzdolabi_yuk*, B grubu ürünler *bgrubu_yuk* ve siparişteki B grubu ürün sayısı *Bsip_adedi* olarak tanımlanmıştır. *Bsip_adedi*, üst kapasite oluşma durumunu kontrol etmek için tanımlanmış bir değişkendir.

ϵ (Epsilon değeri) ve *Minpts* değişkenleri tanımlanır, değeri atanır.

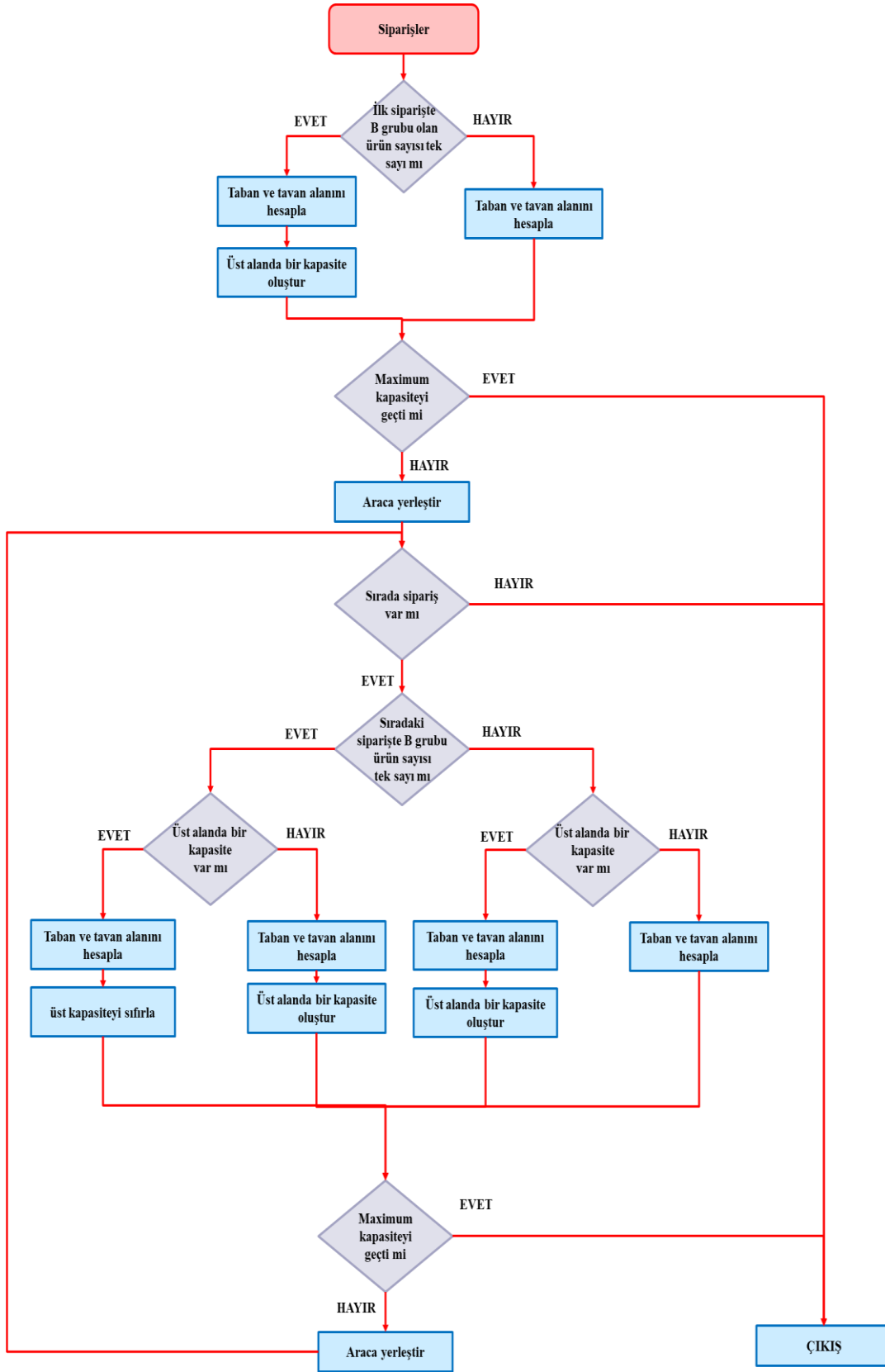
Algoritma, başlanıçta, tüm teslimat noktalarının etiketini “-1” (gürültü noktası) olarak atama yapar, ardından *i* değişkeni 1’den başlayarak *n* adet teslimat noktasına kadar her değer için ϵ uzaklığında en az *Minpts* adet nokta bulundurup bulundurmadığının kontrolünü sağlar.

Eğer ϵ uzaklığında en az *Minpts* adet nokta bulunduruyor ise o noktanın etiketi “0” olarak güncellenir, yani çekirdek nota olarak tanımlanır.

Bir sonraki adımda, *i* değişkeni 1’den başlayarak *n* teslimat noktasına kadar her noktanın etiketi kontrol edilir, eğer etiketi “0” olan bir çekirdek nokta var ise *k* küme sayısı artırılarak bir küme oluşturulur ve çekirdek nokta dahil edilir.

Ardından, *ExpandCluster* fonksiyonu ile çekirdek noktanın çevresindeki komşu noktalar kapasite kısıtını aşacak noktaya gelinceye kadar kümelenir. Bu fonksiyon içerisinde aracın yerleştirilmesi de sağlanmış olur.

ExpandCluster fonksiyonunun, araç kapasitesi ele alınarak algoritma akışı Şekil 4.3’te verilmiştir.



Şekil 4.3. Araç İçi Yerleşim Algoritma Akış Diyagramı

4.1.2. Uygun parametrelerin bulunması

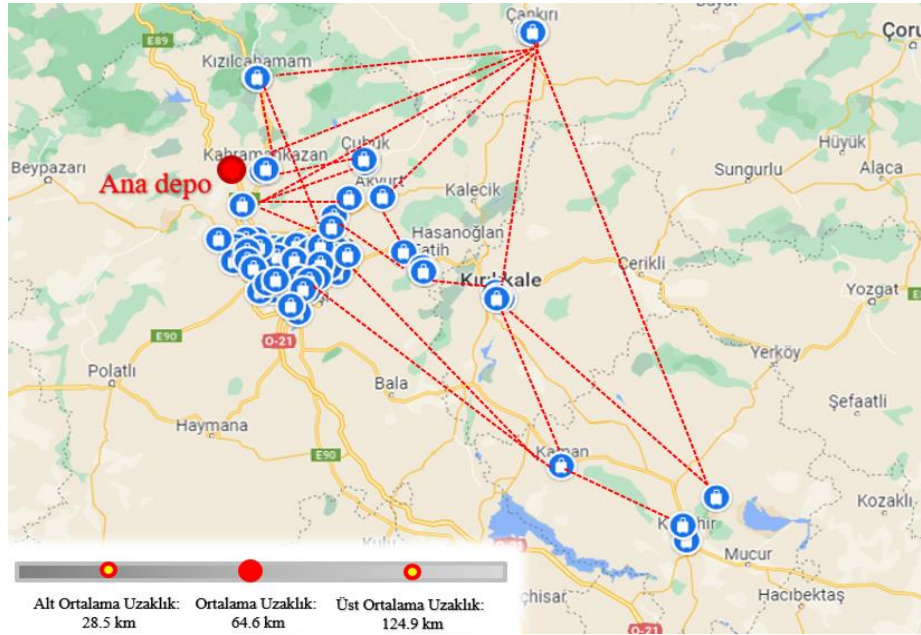
Deneysel çalışmalar da optimum bir sonuç elde edebilmek için deney tasarımının doğru yapılması ve parametrelerin doğru belirlenmesi gerekmektedir.

DBSCAN algoritmasında kullanılan Epsilon ve Minpts parametreleri küme sayısı, küme rotası, toplam alınan yol miktarı, toplam tüketilen yakıt, maliyet ve daha birçok çıktıyı etkileyen en önemli etmenlerdir.

Kümeleme aşamasında MinPts ve Epsilon değerlerini optimize etmek için Yanıt Yüzeyi Yöntemi (Response Surface Metodu - RSM) ile Minitab 21 kullanılmıştır.

Deney tasarımında faktörler Minpts ve Epsilon değerleri olarak seçilirken, kümeleme analizi sonucunda elde edilen küme sayısı, küme dolulukları ve gürültü sayısı çıktıları deney çıktıları olarak alınmıştır.

Epsilon değerine ait faktör seviyeleri belirlenirken, teslimat noktaları ve depo arasında oluşturulan uzaklık matrisinin verilerinden faydalanılmıştır. Tüm matristeki genel uzaklık ortalaması kümenin orta seviye, ortalamanın altında kalan uzaklıklara sahip noktaların uzaklık ortalaması düşük seviye ve ortalama üstünde uzaklıklara sahip noktaların ortalaması ise yüksek seviye verileri olarak alınmıştır. Şekil 4.4'te gösterilmiştir.



Şekil 4.4. Teslimat Adresleri Ortalama Uzaklıklar

Bu durumda uzaklık matrisine göre ortalama deęer 64.6 km, düşük seviye deęer 28.5 km ve yüksek seviye deęer 124.9 olarak belirlenmiřtir.

Minpts noktası faktörü için ise harita üzerinde nokta sayıları gözlemlenmiř olup 1 ve 6 arası faktör seviyeleri tanımlanmıřtır. Bu deęerler literatürde kullanılan 1-5 deęerlerine ek olarak veri setinin gözlem ile analizi nedeniyle 1-6 arası deęerler uygulanmıřtır.

2 Faktör seviyesi ve 2'den fazla faktör seviyeleri olması durumunda tercih edilen Yanıt Yüzeyi Metodu ve Central Compozite Design bařlıęı tercih edilmiřtir. Deęerler "sürekli" deęiřken olarak tanımlanmıřtır. Epsilon deęeri ise uzaklık belirttięi için sürekli bir deęerdir. Minpts girdisi ise, bir veri kümesindeki yoğunluęun düşük olduęu durumlarda daha yüksek "minPts" deęerinin tercih edilebilmesi, yoğunluęun yüksek olduęu durumlarda daha düşük "minPts" deęerinin kullanılabilmesi için sürekli deęiřken olarak tanımlanmıřtır.

Farklı kombinasyonlarda hazırlanan 14 deney ile Yanıt Yüzeyi metodu uygulanmıřtır. Deney tasarımları Tablo 4.4'te verilmiřtir.

Tablo 4.4. Kümeleme Algoritması Deney Tasarımı

Deney No	GİRDİLER		ÇIKTILAR		
	Epsilon	MinPts	Küme Sayısı	Küme dolulukları	Gürültü
	Factor A	Factor B	Response A	Response B	Response C
1	28.5	1	10	0.59	0
2	28.5	2	9	0.65	0
3	28.5	4	8	0.63	3
4	28.5	5	7	0.69	7
5	28.5	6	7	0.69	7
6	64.6	1	7	0.84	0
7	64.6	2	7	0.84	0
8	64.6	3	7	0.84	0
9	64.6	4	7	0.84	0
10	64.6	6	5	0.97	7
11	124.9	1	7	0.84	0
12	124.9	2	7	0.84	0
13	124.9	4	7	0.84	0
14	124.9	6	7	0.84	0

14 adet deney tasarımının sonucunda elde edilen analizler Tablo 4.5'te verilmiřtir.

Tablo 4.5. Response A için Faktörlerin Analizi

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	6,375	0,222	28,67	0,000	
Factor A	-0,624	0,129	-4,84	0,001	1,02
Factor B	-0,764	0,141	-5,41	0,001	1,04
Factor A* Factor A	1,417	0,230	6,16	0,000	1,03
Factor B* Factor B	-0,279	0,248	-1,12	0,294	1,01
Factor A*Factor B	0,776	0,169	4,60	0,002	1,03

Model özeti Tablo 4.6’da yer almaktadır.

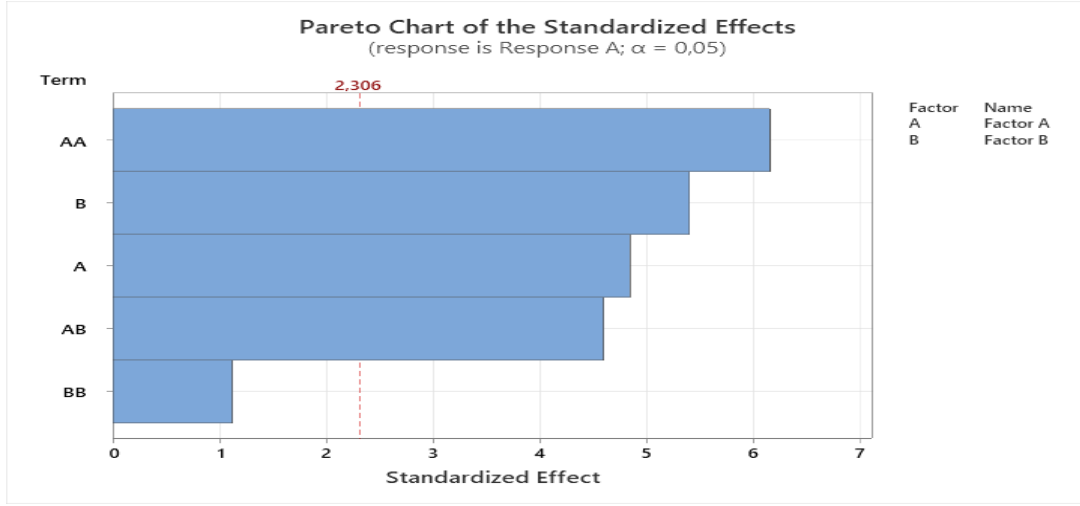
Tablo 4.6. Response A için Model Özeti

S	R²	R²(adj)	R²(pred)
0,381650	93,09%	88,77%	77,21%

Response A, küme sayıları için sonuçları yorumladığımızda, modelin geçerliliği ve anlamlılığını ifade eden $R - sqr$ değeri %93.09 olarak bulunmuştur.

Modelde güven seviyesi %95 seçildiği için P-Value değeri 0.05 değerinin altında kalan değerlerin cevap üzerinde etkisinin olduğunu, bu nedenle her iki parametrenin de küme sayısı değerleri üzerinde etki ettiğini göstermektedir, katsayılar istatistiksel olarak anlamlıdır ve sıfırdan farklı bir değere sahiptir.

Her iki parametrenin birer ve birlikte etkisini gösteren grafik Şekil 4.5’de gösterilmiştir. Bu çıktıya göre, hem Epsilon hem de Minpts değerleri sonuç üzerinde etkili olup ikili kombinasyonlarının da sonuç üzerinde etkilidir.



Şekil 4.5. Response A değerinin Pareto Grafiği

Response B, araç doluluk oranları için analiz sonuçları Tablo 4.7’de verilmiştir.

Tablo 4.7. Response B için Faktörlerin Analizi

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	0,8920	0,0169	52,70	0,000	
Factor A	0,09590	0,00980	9,79	0,000	1,02
Factor B	0,0323	0,0108	3,00	0,017	1,04
Factor A* Factor A	-0,1630	0,0175	-9,31	0,000	1,03
Factor B* Factor B	0,0265	0,0189	1,40	0,198	1,01
Factor A*Factor B	-0,0243	0,0128	-1,89	0,095	1,03

Model Özeti Tablo 4.8’de yer almaktadır.

Tablo 4.8. Response B için Model Özeti

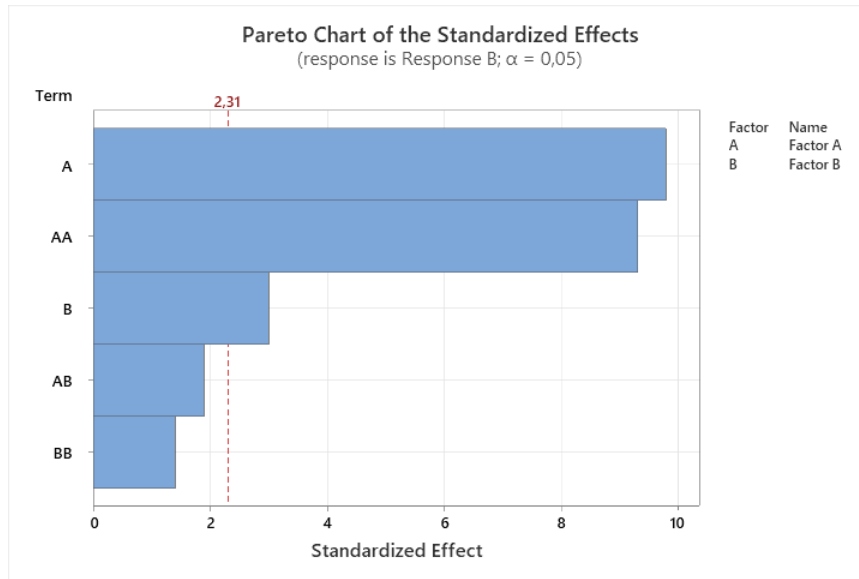
S	R^2	$R^2(\text{adj})$	$R^2(\text{pred})$
0,0290513	95,69%	92,99%	82,56%

Response B, araç doluluk oranları için sonuçları yorumladığımızda, modelin geçerliliği ve anlamlılığını ifade eden $R - sqr$ değeri %95.69 olarak bulunmuştur.

Modelde güven seviyesi %95 seçildiği için P-Value değeri 0.05 değerinin altında kalan değerlerin cevap üzerinde etkisinin olduğunu, bu nedenle her iki parametrenin de araç doluluk oranları üzerinde etki ettiğini göstermektedir, katsayılar istatistiksel olarak anlamlıdır ve sıfırdan farklı bir değere sahiptir.

P-Value değeri 0 çıkan bir faktör olan *epsilon* parametresinin, analizde önemli bir etkiye sahip olduğunu gösterir.

Her iki parametrenin birer ve birlikte etkisini gösteren grafik Şekil 4.6'da gösterilmiştir. Bu çıktıya göre, hem Epsilon hem de Minpts değerleri sonuç üzerinde etkilidir.



Şekil 4.6. Response B değerinin Pareto Grafiği

Response C, gürültü noktaları için analiz sonuçları Tablo 4.9'da verilmiştir.

Tablo 4.9. Response C için Faktörlerin Analizi

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	0,473	0,699	0,68	0,517	
Factor A	-1,646	0,405	-4,07	0,004	1,02
Factor B	2,256	0,444	5,08	0,001	1,04
Factor A* Factor A	0,159	0,723	0,22	0,832	1,03
Factor B* Factor B	1,716	0,780	2,20	0,059	1,01
Factor A*Factor B	-2,102	0,530	-3,97	0,004	1,03

Model Özeti Tablo 4.10’da yer almaktadır.

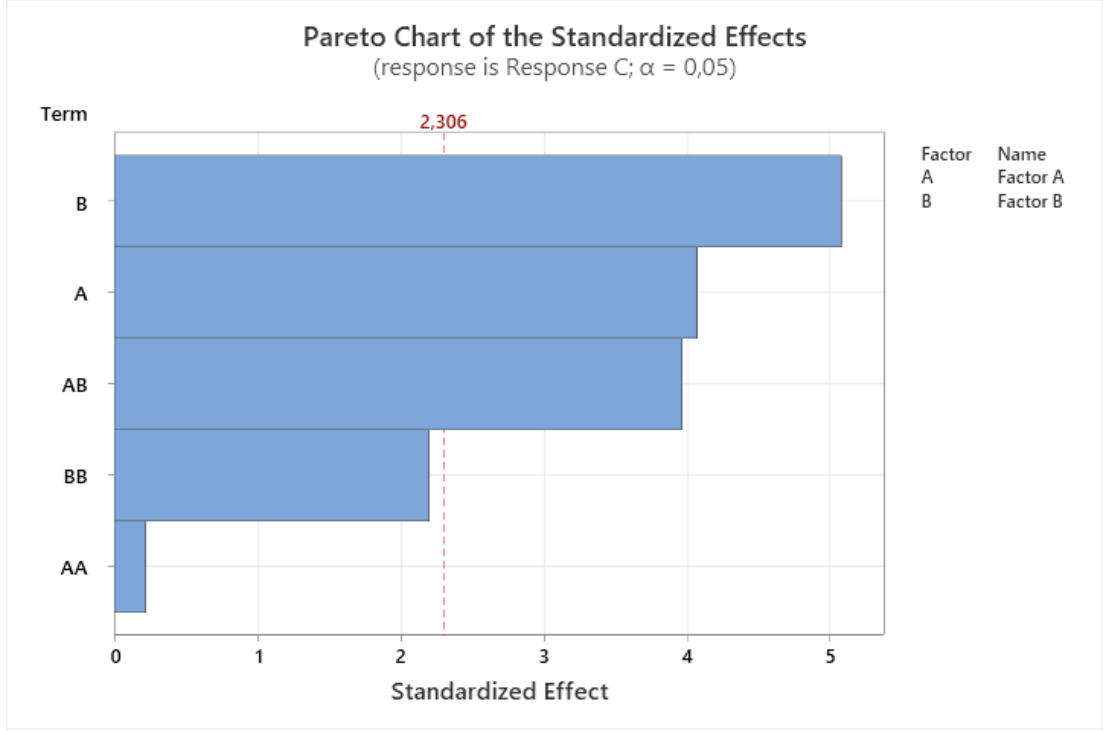
Tablo 4.10. Response C için Model Özeti

S	R²	R²(adj)	R²(pred)
1,20012	89,97%	83,7%	57,47%

Response C, gürültü noktaları için sonuçları yorumladığımızda, modelin geçerliliği ve anlamlılığını ifade eden $R - sqr$ değeri %89.97 olarak bulunmuştur.

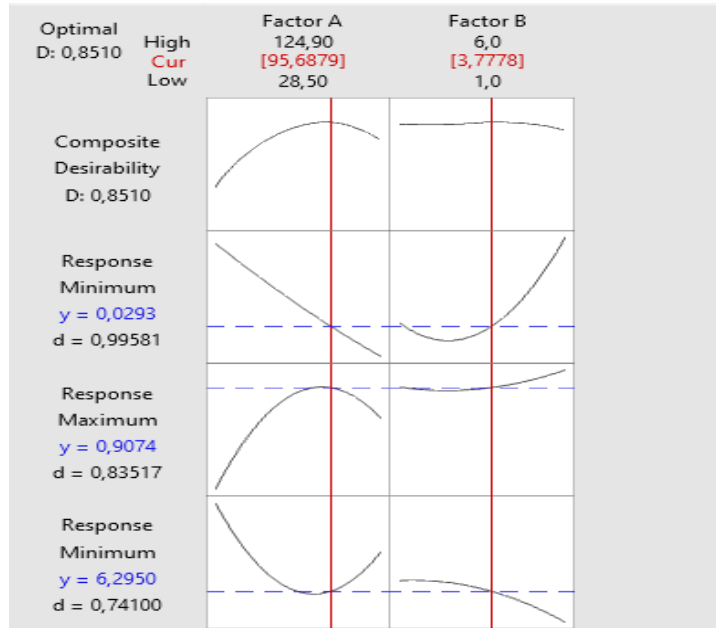
Modelde güven seviyesi %95 seçildiği için P-Value değeri 0.05 değerinin altında kalan değerlerin cevap üzerinde etkisinin olduğunu, bu nedenle her iki parametrenin de gürültü değerleri üzerinde etki ettiğini göstermektedir, katsayılar istatistiksel olarak anlamlıdır ve sıfırdan farklı bir değere sahiptir.

Her iki parametrenin birer ve birlikte etkisini gösteren grafik Şekil 4.7’de gösterilmiştir. Bu çıktıya göre, hem Epsilon hem de Minpts değerleri sonuç üzerinde etkili olup ikili kombinasyonlarının da sonuç üzerinde etkilidir.



Şekil 4.7. Response C değerinin Pareto Grafiği

Eğer küme sayısını minimize, küme doluluk oranlarını maksimize ve gürültü sayısını minimize edecek doğrultuda Response Optimizer çalıştırılmıştır. Her küme bir araca eşit olduğu için küme sayısını minimize etmek maliyeti önemli ölçüde etkileyecektir.



Şekil 4.8. Response Optimizer Sonucu

Şekil 4.8’de de görüldüğü üzere, Factor A olarak belirtilen Epsilon parametresi için optimum değer 95.68 km iken, Factor B ile belirtilen Minpts parametresi için optimum değer ise 3, minimum 3 noktayı kapsayacak şekilde alınmıştır.

4.1.3. Kümeleme sonuçları

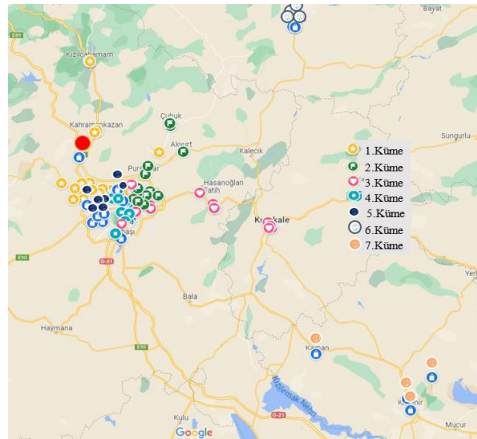
Optimum parametre değerleri ile yapılan kümeleme sonucunda $k = 7$ olmak üzere 7 ayrı küme oluşturulmuştur.

Her bir kümeye ait sonuçlar Tablo 4.11’de gösterilmektedir.

Tablo 4.11. Kümeleme Sonuçları

Küme	Küme Elemanları															Küme Yük Toplamı	Doluluk Oranı							
	1	2	3	4	5	6	9	29	30	31	32	33	34	35	36			37	38	39	51	53	55	
1	1	2	3	4	5	6	9	29	30	31	32	33	34	35	36	37	38	39	51	53	55	59.32	98.87%	
2	7	8	10	46	47	54	60	61	62	67	68	69	70	72	75								59.32	98.87%
3	11	12	13	14	15	16	17	18	19	41	43	44	45	73									59.24	98.73%
4	20	21	22	23	40	42	48	50	71	74													58.68	97.80%
5	24	25	26	27	28	49	52																55.24	92.07%
6	56	57	58	59																			49.8	83.00%
7	63	64	65	66																			11.48	19.13%

İlk 6 kümenin ortalama doluluk oranları %94.89 olarak kümelendiği, son kümenin ise diğer iller kategorisinde yer alması sebebiyle siparişlerin daima az yoğunlukta olduğu bilinen ve kabul edilen bir kısıttır. Teslimat adreslerinin Kümeleme sonucunda harita üzerinde gösterimi Şekil 4.9’da verilmiştir.



Şekil 4.9. Kümeleme Sonuçlarının Harita Üzerinde Gösterimi

Bu aşamaya kadar teslimat noktaları, maksimum araç doluluk oranı ile yoğunluk bazlı kümelenmiştir. DBSCAN kümeleme algoritmasına kapasite kısıtının uygulanması sonucunda, minimum araç – minimum maliyet hedefine yaklaşılmıştır. Bir sonraki adımda, her bir küme için teslimat noktalarına ait teslimat zaman pencereleri aralığında hizmet verilmesi için teslimat adresleri rotalanacaktır.

4.2. Zaman Pencereci Araç Rotalama Uygulaması

Kümeleme aşamasında elde edilen 7 adet kümenin rotalanması için Metsaezgisel yöntemlerden biri olan Karınca Kolonisi Algoritması önerilmiştir.

Problemin boyutları, dönemsel artan ve azalan bir sipariş durumuna bağlı olması nedeniyle teslimat adreslerinin sayısı değişkenlik göstermektedir. Buna bağlı olarak araç sayısı, problem veri seti Sezgisel Yöntemler ile çözülemeyecek veri boyutlarına ulaşabilir.

Problemin büyük ve karmaşık olmasından dolayı doğrusal programlamanın bu durumlarda yeterli kalamayabileceği düşünülmüştür. Karınca Kolonisi Algoritması, araç rotalama problemleri gibi kombinasyonlar içeren optimizasyon problemleri ile başa çıkabilme yetisine sahiptir. Problemin dinamik yapısı nedeni ile metasezgisel yöntemlerden biri olan Karınca Kolonisi Algoritması ile çözümlenmiştir.

Ancak geleneksel Karınca Kolonisi algoritmasından farklı olarak, probleme ait zaman pencereleri bu aşamada probleme dahil edilmiş, daha öncesinde kapasite kısıtını sağlayan kümeler bu adımda öncelik ve zaman pencereleri kısıtları altında değerlendirilmiştir.

4.2.1. Algoritma parametrelerinin belirlenmesi

Karınca Kolonisi Algoritmasına ait global parametreler olan *MaxIT*, *Nant*, *alpha*, *beta* parametreleri için bir deney tasarımı yapılmıştır.

alpha parametresi için, karıncaların feromon miktarına verdiği önemin bir ağırlığı olan tanımlanabilir. Bu değer genellikle 1-5 aralığında bir değer aldığı için, çalışmada da bu değerler aralığında bir deney tasarımı yapılmıştır.

beta parametresi ise, heuristik bilgileri önemseme ağırlığıdır, bu değer genellikle 1-5 aralığında bir değer aldığı için, çalışmada da bu değerler aralığında bir deney tasarımı yapılmıştır.

Nant ve *MaxIT* parametreleri ise, literatür çalışmalarının izlenmesi ve deneme-yenilme yöntemlerinden elde edilen veriler sonucunda, 100-200 aralığında değerler almışlardır.

rho ve *tau* parametreleri ise, karıncaların rotalarını etkileyen buharlaşma ve feromon yenilenme oranlarıdır. Deney tasarımına dahil etmemekle beraber bu değerler ortalama olan 0.5 olarak seçmiştir. Bunun nedeni ise, feromon yenilenme ya da buharlaşma oranlarının ortalama değerlerde tutulup, uç noktalarda değer olarak feromonun hiç yenilenmemesi ya da feromonun hiç buharlaşmaması durumunun ortadan kaldırılmak istenmesidir.

Kümeleme aşamasında MinPts ve Epsilon değerlerini optimize etmek için Yanıt Yüzeyi Yöntemi (Response Surface Metodu - RSM) kullanılmıştır, Minitab 21 kullanılmıştır. Problemin doğasına ve gereksinimlerine bağlı olarak değerler “sürekli” değişken olarak tanımlanmıştır.

Deney tasarımında faktörler *MaxIT*, *Nant*, *alpha*, *beta* parametreleri olarak seçilirken, analiz sonucunda elde edilen ceza maliyeti ve alınan yol (km) değeri çıktıları deney çıktıları olarak alınmıştır. Deney tasarımı Tablo 4.12’de verilmiştir.

Tablo 4.12. Karınca Kolonisi Algoritması Deney Tasarımı

Deney No	DENEY TASARIMI					
	GİRDİLER				ÇIKTILAR	
	<i>MaxIT</i>	<i>Nant</i>	<i>alpha</i>	<i>beta</i>	<i>ceza</i>	<i>km</i>
	Factor A	Factor B	Factor C	Factor D	Response A	Response B
1	100	100	0	0	38700	39387.5
2	125	100	1	0	34900	35617.3
3	150	100	2	0	43090	43816.55
4	175	100	3	0	36560	37240.05
5	200	100	4	1	57480	57873.65
6	100	200	0	1	53290	53830.33
7	125	125	0	1	52570	53078.25
8	150	125	1	1	54700	55172.8
9	175	125	2	2	67070	67514.43
10	200	125	3	2	69980	70376
11	100	150	4	2	66630	66959.4
12	125	200	0	2	62090	62419
13	150	150	0	3	63490	63857.25
14	175	150	1	3	72610	72947
15	200	150	2	3	70960	71310
16	100	175	3	3	72180	72517.9
17	125	175	4	4	73760	74129.78
18	150	200	0	4	71930	72277.08
19	175	175	0	4	68780	69103.65
20	200	175	1	4	73030	73331.13
21	200	175	5	5	79290	79637.98
22	200	175	4	5	71540	71852.25
23	200	175	3	5	70840	71158.48
24	200	175	2	5	74730	75014.25
25	200	175	1	5	76490	76772.68
26	100	175	1	1	53640	54076.65
27	125	150	2	2	63440	63860.88
28	150	200	1	4	76260	76571.08
29	200	100	3	3	77560	77859.08
30	125	100	5	2	64280	64594.65

Analiz sonucunda, Response A, ceza maliyeti çıktısı için elde edilen analizler Tablo 4.13'de verilmiştir.

Tablo 4.43. Response A - Ceza Maliyeti Çıktıları

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	67661	2519	26,86	0,000	
Factor A	1571	2287	0,69	0,503	9,45
Factor B	873	4804	0,18	0,858	35,64
Factor C	2843	3376	0,84	0,413	14,35
Factor D	14829	5121	2,90	0,011	36,35
Factor A* Factor A	-1882	3089	-0,61	0,551	5,50
Factor B* Factor B	-2910	9095	-0,32	0,753	42,86
Factor C* Factor C	472	2846	0,17	0,871	3,87
Factor D* Factor D	-28222	9703	-2,91	0,011	46,58
Factor A*Factor B	-13311	9082	-1,47	0,163	65,80
Factor A*Factor C	-4395	4262	-1,03	0,319	12,46
Factor A*Factor D	16736	9847	1,70	0,110	65,33
Factor B*Factor C	-1708	10718	-0,16	0,876	88,06
Factor B*Factor D	19444	17739	1,10	0,290	156,82
Factor C*Factor D	5077	11400	0,45	0,662	75,40

Model Özeti Tablo 4.14'te verilmiştir.

Tablo 4.14. Response A - Ceza Maliyeti Model Özeti

S	R²	R²(adj)	R²(pred)
3064,47	%96,94	%94,09	%81,15

Response A, ceza maliyeti çıktısı için sonuçları yorumladığımızda, modelin geçerliliği ve anlamlılığını ifade eden R^2 değeri %96,94 olarak bulunmuştur. R^2 değeri yükseldikçe deney setini iyi ifade ettiği sonucuna ulaşırız.

Modelde güven seviyesi %95 seçildiği için P değeri 0.05 değerinin altında kalan değerlerin cevap üzerinde etkisinin olduğunu, bu nedenle β parametresinin ceza maliyeti üzerinde etki ettiğini göstermektedir.

Analiz sonucunda, Response B, alınan yol (km) çıktısı için elde edilen analizler Tablo 4.15'te verilmiştir.

Tablo 4.15. Response B – Alınan Yol Çıktıları

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	68034	2515	27,05	0,000	
Factor A	1593	2283	0,70	0,496	9,45
Factor B	926	4797	0,19	0,849	35,64
Factor C	2836	3371	0,84	0,413	14,35
Factor D	14648	5113	2,86	0,012	36,35
Factor A* Factor A	-1904	3084	-0,62	0,546	5,50
Factor B* Factor B	-2783	9081	-0,31	0,763	42,86
Factor C* Factor C	467	2842	0,16	0,872	3,87
Factor D* Factor D	-27966	9689	-2,89	0,011	46,58
Factor A*Factor B	-13215	9068	-1,46	0,116	65,80
Factor A*Factor C	-4389	4256	-1,03	0,319	46,58
Factor A*Factor D	16601	9832	1,69	0,112	65,33
Factor B*Factor C	-1556	10702	-0,15	0,886	88,06
Factor B*Factor D	19233	17712	1,09	0,295	156,82
Factor C*Factor D	5018	11383	0,44	0,666	75,40

Model Özeti Tablo 4.16'da yer almaktadır.

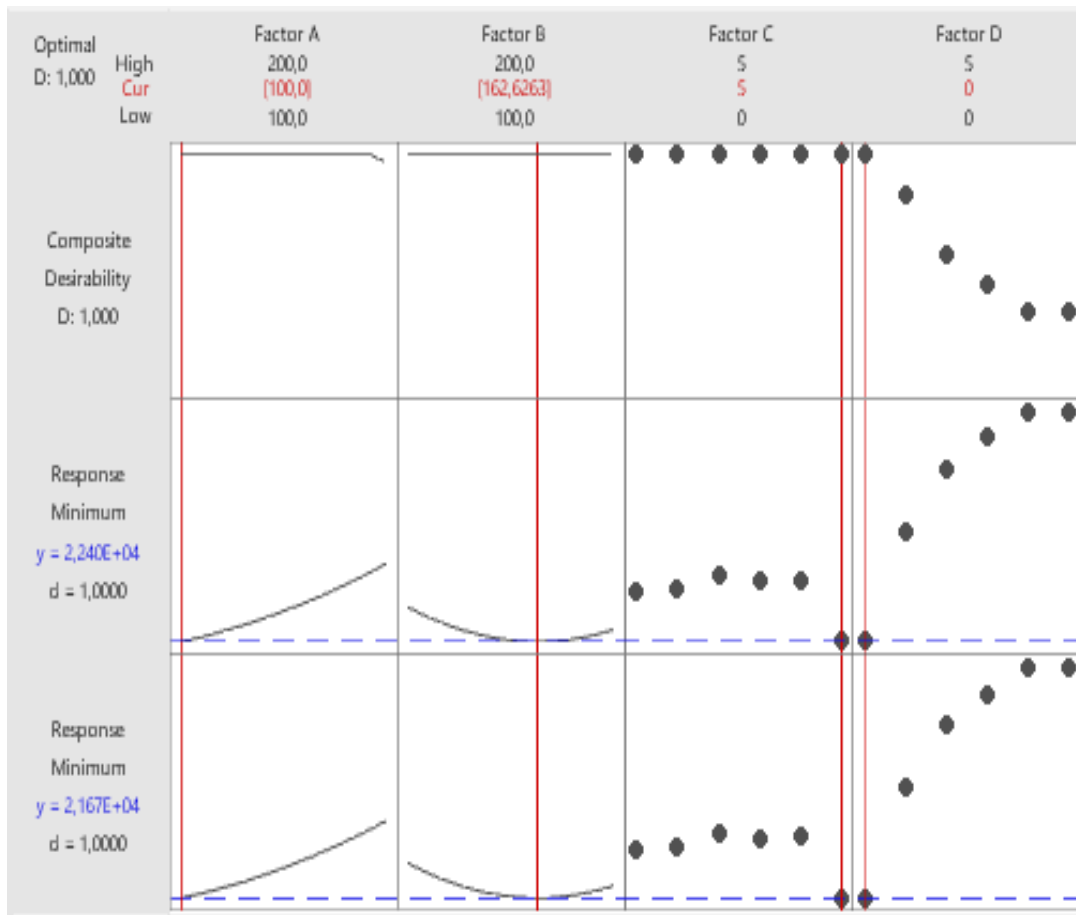
Tablo 4.16. Response B – Alınan Yol Model Özeti

S	R^2	$R^2(\text{adj})$	$R^2(\text{pred})$
3059,83	%96,89	%93,99	%80,75

Response B, alınan yol (km) çıktısı için sonuçları yorumladığımızda, modelin geçerliliği ve anlamlılığını ifade eden R^2 değeri %96,89 olarak bulunmuştur. R^2 değeri yükseldikçe deney setini iyi ifade ettiği sonucuna ulaşırız.

Modelde güven seviyesi %95 seçildiği için P değeri 0.05 değerinin altında kalan değerlerin cevap üzerinde etkisinin olduğunu, bu nedenle β parametresinin alınan yol (km) üzerinde etki ettiğini göstermektedir.

Ceza maliyeti ve alınan yol (km) çıktılarını minimize edecek doğrultuda Response Optimizer çalıştırılmıştır. Elde edilen sonuçlar Şekil 4.10’da gösterilmiştir.



Şekil 4.10. Optimizer Sonuçları

Sonuçlar incelendiğinde, Faktor A, *MaxIT* parametresi için 100; Faktor B, *Nant* parametresi için 162; Faktor C, *alpha* parametresi için 5 ve *beta* parametresi için ise 0 değeri optimum değerler olarak belirlenmiştir.

4.2.2. Karınca kolonisi algoritmasının uygulanması

Kümeleme algoritması sonucunda elde edilen her 7 küme için, her 7 araç için rotalama aşamasına geçilmiştir.

Uygulanacak olan Karınca Kolonisi algoritması için MATLAB R2022a programı kullanılmıştır.

Bu çalışmada klasik Karınca Kolonisi algoritmasından farklı olarak zaman pencereleri içerisinde teslimat kısıtı algoritmaya dahil edilmiştir. Her araç zaman kısıtlarını dikkate alarak rotalanacaktır.

Zaman penceresi kısıtı dahilinde uygulanan Karınca Kolonisi algoritması Tablo 4.17'de verilmiştir.

Tablo 4.17. Karınca Kolonisi Algoritması adımları

Oluşturulan Algoritma

BAŞLA

Teslimat adreslerine ait x, y koordinatlarını oku.

Zaman matrisini ve süre matrisini oku.

Zaman pencerelerini oku.

Teslimat adresi sayısını bul.

MaxIT, *Nant*, *alpha*, *beta*, *rho* ve *tau* parametrelerini tanımla.

Karınca kolonisi için boş bir matris tanımla.

En iyi karıncanın maliyetini sonsuz değer al.

for i'den *MaxIT* kadar

 for k 1'den *Nant* kadar

Tablo 4.17. (Devamı) Karınca Kolonisi Algoritması adımları

turun başlangıç noktasını ana depo seç.

karıncanın toplam zamanını 0'a eşitle.

cezayı 0 olarak tanımla.

for tur adımını 2'den *NVar* adres sayısı kadar

 başlangıç noktasını turun sonuna bitiş adresi olarak ekle

alpha ve *beta* parametrelerinden sıradaki adres için seçilme ihtimali bul.

 RouletteWheelSelection fonksiyonu ile sıradaki rotayı seç

 Seçilen adresi tura ekle

 Gidiş zamanını karıncanın aldığı süreye ekle

 if karınca zaman penceresinden önce geldi ise

 zaman penceresinin başlangıcını bekler ve 20 dk hizmet süresi

eklenir

 elseif karınca zaman penceresi aralığında geldi ise

 hizmet süresi 20 dk eklenir

 elseif karınca zaman penceresinden daha geç geldi ise

 ceza maliyeti geç kalınan sürenin 10 birim katı olarak eklenir

 hizmet süresi 20 dk eklendi

 end if

 end for

Karıncanın maliyetine ceza maliyetini ekle.

 if karıncanın maliyeti en iyi karıncanın maliyetinden küçükse

 karınca

 end if

end for

Tablo 4.17. (Devamı) Karınca Kolonisi Algoritması adımları

feromon güncelleme fonksiyonu

BİTİŞ

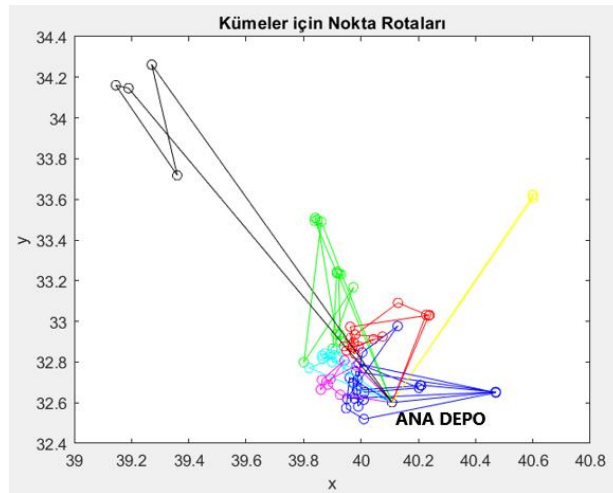
Uygulama sonucunda elde edilen rotalar ve maliyetleri Tablo 4.18’de verilmiştir.

Maliyet hesabında aracın yakıt olarak 100 kilometrede 20 litre yakıt yaktığı varsayılarak ceza maliyeti de eklenerek maliyet hesaplanmıştır.

Tablo 4.18. Çalışma Sonucu Elde Edilen Çıktılar

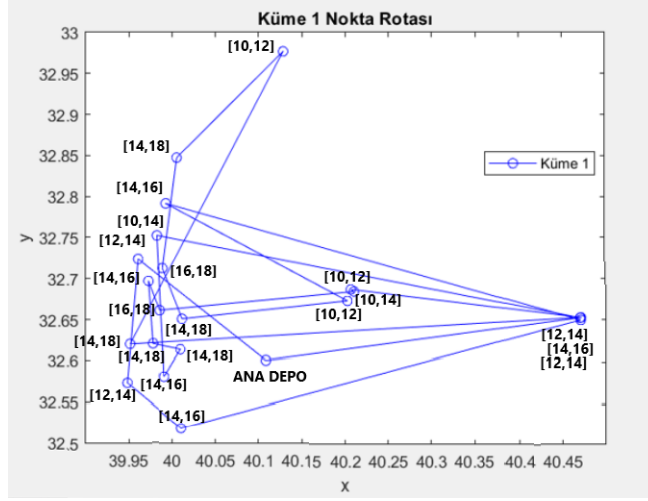
ROTALAR	Alınan Yol (km)	Ceza Maliyeti (TL)	Toplam Maliyet
1 76 36 30 31 6 53 1 38 39 55 9 29 5 2 3 35 37 33 34 32 51 4 76	35,436.4	₺ 34,780	₺ 176,525.40
2 76 8 10 60 75 68 47 62 67 46 54 70 69 61 7 72 76	4,367.8	₺ 4,100	₺ 21,571.20
3 76 14 19 73 43 15 16 18 13 11 12 44 45 41 17 76	9,982.2	₺ 9,510	₺ 49,438.80
4 76 71 21 20 42 50 23 40 22 48 74 76	159.7	0	₺ 638.80
5 76 28 26 49 24 27 25 52 76	124.8	0	₺ 499.20
6 76 59 56 57 58 76	321.4	0	₺ 1,285.60
7 76 65 66 63 64 76	473.7	0	₺ 1,894.80
TOPLAM			₺ 251,853.80

Tüm rotalar Şekil 4.11’de verilmiştir.

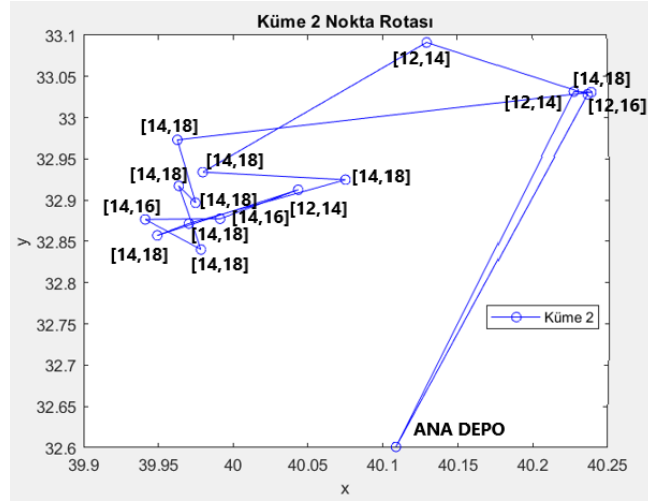


Şekil 4.11. Oluşturulan Tüm Rotalar

Birinci kümeye ait rota gösterim grafiği Şekil 4.12’de, ikinci kümeye ait rota gösterim grafiği Şekil 4.13’te verilmiştir.

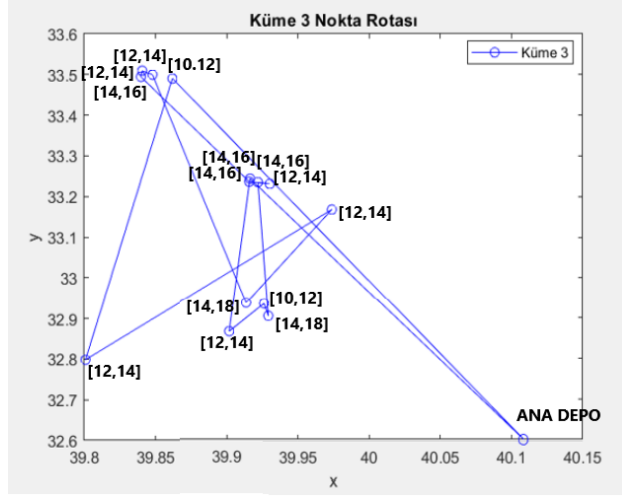


Şekil 4.12. Küme 1 Rotası

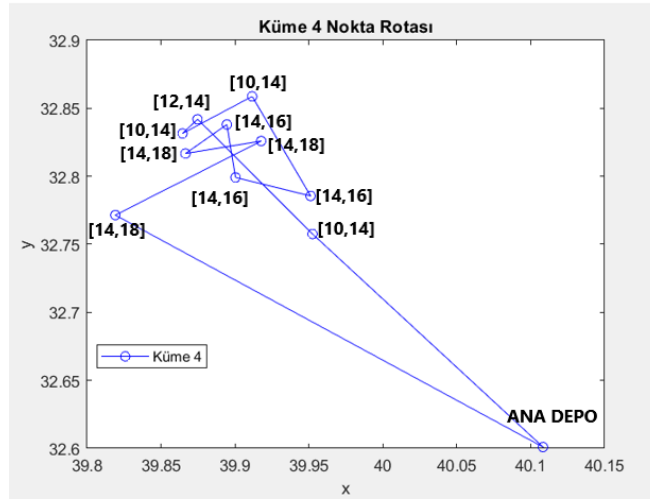


Şekil 4.13. Küme 2 Rotası

Üçüncü kümeye ait rota gösterim grafiği Şekil 4.14'te, dördüncü kümeye ait rota gösterim grafiği Şekil 4.15'de verilmiştir.

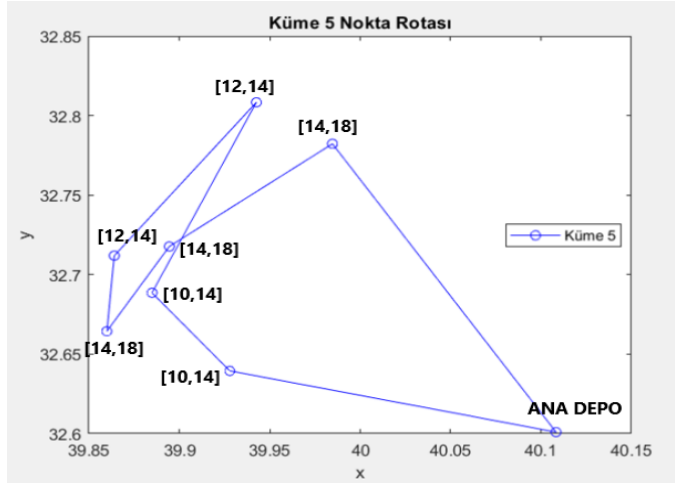


Şekil 4.14. Küme 3 Rotası

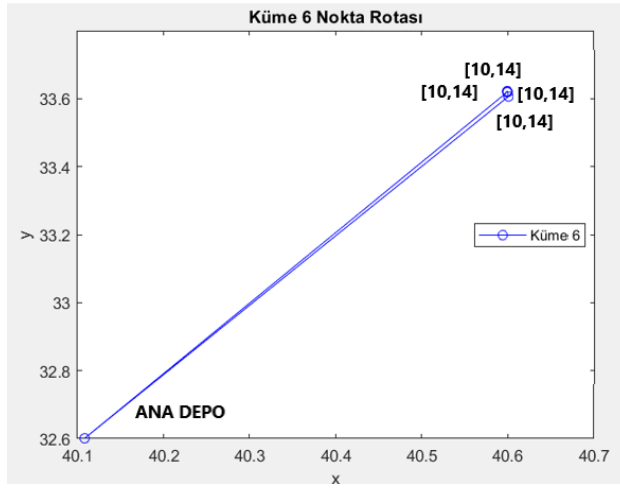


Şekil 4.15. Küme 4 Rotası

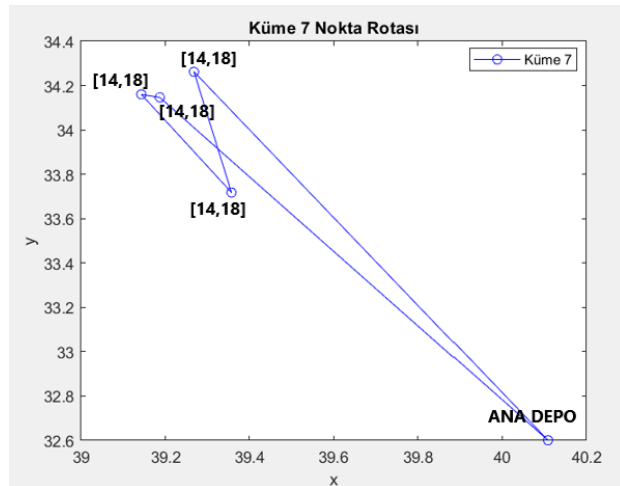
Beşinci kümeye ait rota gösterim grafiği Şekil 4.16'da, altıncı kümeye ait rota gösterim grafiği Şekil 4.17'de ve yedinci kümeye ait rota gösterim grafiği Şekil 4.18'de verilmiştir.



Şekil 4.16. Küme 5 Rotası



Şekil 4.17. Küme 6 Rotası



Şekil 4.18. Küme 7 Rotası

Firma personelinin bilgi ve deneyimlerine dayalı yapmış olduğu planlama sonucunda elde edilen rotalama Tablo 4.19’da verilmiştir.

Tablo 4.19. Çalışma Öncesi Durum

ARAÇLAR														
1	14	15	16	18										
2	4	47	59	60	61	65	69	70						
3	11	12	13	17	63	64	66	73						
4	7	8	9	10	72									
5	20	21	27	40	42	43	44	45	48					
6	56	57	58	62	67	68	75							
7	1	2	3	4	5	6	22	23	24	25	26	41	54	55
8	50	51	52	53	71									
9	28	29	30	31	32	33								
10	19	34	35	36	37	38	39	49	74					

Firma tarafından tablodaki gibi kümelenen 10 araç zaman pencereleri dikkate alınarak teslimatlarını gerçekleştirmektedir.

Yapılan çalışmada, planlama personelinin deneyim ve gözlemlerine dayalı rotalamadan elde edilen 10 araç sayısı ortalama %94.89 araç doluluğuyla 7 araca düşürülmüştür.

5. SONUÇ VE ÖNERİLER

Çalışmada, Ankara’da yer alan ana dağıtım deposundan Ankara ve çevre illerdeki 75 adet bayiye belirli zaman pencerelerinde teslimat yapması gereken bir beyaz eşya deposunun gerçek zamanlı bir problemi incelenmiştir.

Öncelikle teslimat adreslerinin Ankara çevresinde yoğunlaştığı analiz edilmiştir ve ürün grupları yerleştirme kısıtlarına göre sınıflandırılmıştır. Teslimat noktaları yoğunluk tabanlı DBSCAN algoritması ile kümeleneştir. Kümeleme sırasında aracın kapasitesini aşmayacak şekilde araç içi yerleştirme yapılmıştır. Üst üste istiflenebilme kriterini sağlayan ürün grupları üst üste yerleştirilerek araç doluluk oranı yükseltilmiştir.

Her bir küme bir aracı ifade etmektedir. Araçların kapasite, yerleşim kısıtları birbirlerine olan yakınlıkları bazında kümelendikten sonra zaman pencerelerine göre hangi sıralamada teslimat yapacağı problemi ortaya çıkmıştır.

Bu durumda sipariş boyutunun değişikliği ve adres dinamikliği nedeniyle metasezgisel yöntemlerden biri olan Karınca kolonisi algoritması rotalama adımı için seçilmiştir. Bu aşamada klasik karınca kolonisi algoritmasının dışına çıkarak algoritma içine zaman pencereleri kısıtlarını sağlayacak şekilde rotalama kısıtı eklenmiştir.

Teslimat noktasına zaman başlangıcından daha erken gimesi durumunda bekleyebilen ancak geç gitmesi durumunda ceza maliyeti ekleyen kısıt algoritmaya entegre edilmiştir.

Bu rotalama sonucunda hem kapasite, hem yerleştirme kısıtları hem de zaman pencereleri kısıtlarını sağlayan bir yaklaşım önerilmiştir.

Gelecek çalışmalarda, bu çalışmada varsayılan ya da kabul edilen durumlar üzerine çalışmalar planlanmıştır.

Tek tip araç tipi yerine deęişken kapasiteli araçlar, ürün gruplarının taban alanlarının ürün modeline göre deęişkenlik göstermesi gibi durumlar bu çalışmada varsayılan durumlardır.

Aynı zamanda küçük ölçekli siparişlere sahip çevre iller için “sadece haftanın belirli günlerinde teslimat” yapma ve böylece siparişler için belirli günlere randevu verilerek aracın doluluk oranı düşük teslimat yapması engellenebilir. Çalışmanın gelecek aşamalarında hedeflenen geliştirme noktalarından biridir.

Önerilen yaklaşım, kısıtlar ile algoritmaların entegresini sağlamış olup araç sayısında %30 iyileştirme sağlamıştır. Araç doluluk oranları ortalama %94.89 olacak şekilde artırılmıştır.

KAYNAKLAR

- Akdaş, H. Ş. (2021). Vehicle Route Optimization for Solid Waste Management: A Case Study of Maltepe, Istanbul. 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI). IEEE.
- Aydın, Ş. (2022). *Ülkelerin Yolsuzluk Göstergelerine Göre DBSCAN ile kümelenmesi* [Yüksek Lisans Tezi]. Marmara Üniversitesi, Sosyal Bilimler Enstitüsü, Ekoometri Anabilim Dalı, İstatistik Bölümü.
- Bozdemir, M. K., Bozdemir, M., Özcan, B. (2019). Route First-Cluster Second Method for Personal Service Routing Problem. *Journal of Engineering Studies and Research*, 25(2), 18-24. <https://doi.org/10.29081/jesr.v25i2.31>.
- Bujel, K., Lai, F., Szczecinski, M., So, W., Fernandez, M. (2018). Solving High Volume Capacitated Vehicle Routing Problem with Time Windows using Recursive-DBSCAN clustering algorithm. *ArXiv*. <https://doi.org/10.48550/arXiv.1812.02300>.
- Choudhari, A., Ekbote, A., Chaudhuri, P. (2022). Capacitated Vehicle Routing Problem Using Conventional and Approximation Method. *ArXiv*. <https://doi.org/10.48550/arXiv.2208.00046>.
- Cordeau, J. F., Gendreau, M., Hertz, A., Laporte, G. & Sormany, J. S. (2005). New Heuristics for the Vehicle Routing Problem. *In Logistics Systems: Design and Optimization*, 279-297, ISBN : 978-0-387-24971-1.
- Cömert, S. E., Yazgan, H. R., Görgülü, N. (2019). Eş Zamanlı Topla Dağıt Araç Rotalama Problemi için İki Aşamalı Bir Çözüm Yöntemi Önerisi. *Int. J. Adv. Eng. Pure Sci.*, 2, 107-117. <https://doi.org/10.7240/jeps.461528>.
- Cömert, S. E., Yazgan, H. R., Çakır, B., Sarı N. (2020). Esnek Zaman Pencereli Araç Rotalama Probleminin Çözümü için Önce Kümele Sonra Rotala Temelli Bir Yöntem Önerisi, Bir Süpermarket Örneği. *Konya Journal of Engineering Sciences*. Cilt: 8 Sayı: 1, 18-31. <https://doi.org/10.36306/konjes.698326>.
- Çiftçi, E. (2011). *Yüksek Dereceden Yanıt Yüzeyi Modelleri Üzerine Bir Çalışma* [Yüksek Lisans Tezi]. Hacettepe Üniversitesi, İstatistik Ana Bilim Dalı.
- Dereli, T., Daş, G. S., (2010). Konteyner Yükleme Problemleri İçin Karınca Kolonisi Optimizasyonu Yaklaşımı. *Gazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi*, Cilt 25, No 4, 881-894.
- Eren, İ. (2004). *Patateslerin Osmotik Dehisrasyonunun "Response Surface" Metodu Kullanılarak Optimizasyonu* [Yüksek Lisans Tezi]. Ege Üniversitesi, Fen Bilimleri Enstitüsü, Gıda Mühendisliği Anabilim Dalı.

- Gocken, T., Yaktubay, M. (2019). Comparison of Different Clustering Algorithms via Genetic Algorithm for VRPTW. *International Journal of Simulation Modelling*, Cilt 18(4), 574-585.
- Hoff A., Gribkovskaia I., Laporte G., Løkketangen A. (2009). Lasso solution strategies for the vehicle routing problem with pickups and deliveries, *European Journal of Operational Research*, Cilt 192, 755-766.
- Kasap, T. (2017). *Kağıt Endüstrisi Atıksularının Peroksit İlaveli Elektrokoagülasyon Yöntemi ile Arıtımında Cevap Yüzey Yöntemi Kullanılarak Proses Optimizasyonu* [Yüksek Lisans Tezi]. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Çevre Mühendisliği Ana Bilim Dalı, Çevre Mühendisliği Programı.
- Kumar, S. N., Panneerselvam, R. (2012). A Survey on the Vehicle Routing Problem and Its Variants. *Intelligent Information Management*, Cilt 4, 66-74.
- Le T. D. C., Nguyen, D. D., Oláh, J., Pakurár, M. (2022). Clustering Algorithm for a Vehicle Routing Problem with Time Windows. *Transport*, Cilt 37(1), 17–27. <https://doi.org/10.3846/transport.2022.16850>.
- Özdağ, H., Aygör, N., Parlak, A. (2012). Karınca Kolonisi Algoritmasının Zaman Çizelgelemesi Üzerine: Bir Modellemesi ve Uygulaması, *Akademik Bilişim '12*, Uşak, Türkiye.
- Özdemir, Y.S. (2008). *Karınca Kolonisi Algoritması İle Bilgisayar Ağlarının Topolojik En İyilenmesi* [Yüksek Lisans Tezi]. Başkent Üniversitesi Fen Bilimleri Enstitüsü.
- Özkök, Z. (2021). *Zaman Pencere Araç Rotalama Probleminin Çözümü için Hibrit Bir Yöntemin Geliştirilmesi* [Yüksek Lisans Tezi]. Necmettin Erbakan Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı.
- Rezaei, M. (2016). *Clustering Validation* [Doktora Tezi]. University of Eastern Finland, Publications of the University of Eastern Finland Dissertations in Forestry and Natural Sciences.
- Sánchez, D. G., Tabares, A., Faria, L. T., Rivera, J. C., Franco J.F. (2022). A Clustering Approach for the Optimal Siting of Recharging Stations in the Electric Vehicle Routing Problem with Time Windows. *Energies* 2022, 15, 2372.
- Ünsal, Ö., Yiğit, T. (2018). Yapay Zeka ve Kümeleme Tekniklerini Kullanarak Geliştirilen Yöntem ile Okul Servisi Rotalama Probleminin Optimizasyonu, *Mühendislik Bilimleri ve Tasarım dergisi*, Cilt: 6 Sayı: 1, 7 – 20.
- Villalba, A. F. L., Rotta, E. C. G. L. (2020). Comparison of DbSCAN and K-means clustering methods in the selection of representative clients for a vehicle routing model, *Congreso Internacional de Innovación y Tendencias en Ingeniería (CONIITI)*, 1-6. <https://doi.org/10.1109/CONIITI51147.2020.9240399>.
- Villalba, A. F. L., Rotta, E. C. G. L. (2022). Clustering and heuristics algorithm for the vehicle routing problem with time Windows. *International Journal of Industrial Engineering Computations*, 13,165–184.

- Wang, Y. (2023). A clustering-based extended genetic algorithm for the multidepot vehicle routing problem with time windows and three-dimensional loading constraints. *Applied Soft Computing*, 133. <https://doi.org/10.1016/j.asoc.2022.109922>.
- Yılmaz, Ş. (2008). *Çok depolu araç rotalama probleminin karınca kolonisi optimizasyonu ile modellenmesi ve bir çözüm önerisi* [Yüksek Lisans Tezi]. İstanbul: Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği.
- Zhang J. (2017). An Efficient Density-Based Clustering Algorithm for the Capacitated Vehicle Routing Problem, *International Conference on Computer Network, Electronic and Automation, ICCNEA*, 2(4), 465–469. <http://dx.doi.org/10.1109/ICCNEA.2017.96>.

EKLER

EK A. Uzaklık Matrisi

EK A

Tablo A.4. Uzaklık Matrisi

Table with 76 columns and 76 rows, containing numerical data representing a distance matrix. The diagonal elements are all 0, and the matrix is symmetric. The values range from 0 to approximately 74.

ÖZGEÇMİŞ

Ad-Soyad :Tuğba Gül YANTUR

ÖĞRENİM DURUMU:

- **Lisans** : 2018, Sakarya Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü
- **Yükseklisans** : 2023, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Bölümü

MESLEKİ DENEYİM VE ÖDÜLLER:

- 2018-2019 yılları arasında Colin's Eroğlu Holding bünyesinde Alokasyon Uzmanı olarak görev yaptı.
- 2020-2021 yılları arasında Chemnitz Teknoloji Üniversitesi'nde Erasmus Proje Stajyeri olarak Yalın Üretim - Kanban projesinde yer aldı.
- Nisan 2021 yılından itibaren Ceva Lojistik bünyesinde Operasyonel Mükemmellik Uzmanı olarak görev yapmaktadır.