



**ARAÇ ROTALAMA PROBLEMİNDE GPS
VERİLERİNİN YAPAY ZEKÂ ALGORİTMALARI İLE
ANALİZİ**

Ebru ERDEM

Danışman: Doç. Dr. Tolga AYDIN

Doktora Tezi

Bilgisayar Mühendisliği Ana Bilim Dalı

2025

(Her hakkı saklıdır.)

T.C.
ATATÜRK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

**ARAÇ ROTALAMA PROBLEMİNDE GPS VERİLERİNİN YAPAY ZEKÂ
ALGORİTMALARI İLE ANALİZİ**

(Analysis of GPS Data Using Artificial Intelligence Algorithms in the Vehicle Routing Problem)

DOKTORA TEZİ

Ebru ERDEM

Danışman: Doç Dr. Tolga AYDIN
İkinci Tez Danışmanı: Prof. Dr. Burak ERKAYMAN

Erzurum
Temmuz, 2025

KABUL VE ONAY TUTANAĞI

Ebru ERDEM tarafından hazırlanan “**Araç Rotalama Probleminde GPS Verilerinin Yapay Zekâ Algoritmaları İle Analizi**” başlıklı çalışması 14 / 07 / 2025 tarihinde yapılan tez savunma sınavı sonucunda başarılı bulunarak jürimiz tarafından Bilgisayar Mühendisliği Ana Bilim Dalı, Bilgisayar Mühendisliği Bilim Dalında doktora tezi olarak kabul edilmiştir.

Jüri Başkanı:	Prof. Dr. İbrahim Yücel ÖZBEK Atatürk Üniversitesi	Aslı ıslak imzalıdır
Danışman:	Doç. Dr. Tolga AYDIN Atatürk Üniversitesi	Aslı ıslak imzalıdır
Jüri Üyesi:	Doç. Dr. Mahmut TUTAM Erzurum Teknik Üniversitesi	Aslı ıslak imzalıdır
Jüri Üyesi:	Dr. Öğr. Üyesi Bilal USANMAZ Atatürk Üniversitesi	Aslı ıslak imzalıdır
Jüri Üyesi:	Dr. Öğr. Üyesi Işıl KARABEY AKSAKALLI Erzurum Teknik Üniversitesi	Aslı ıslak imzalıdır

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili maddelerinde belirtilen şartları yerine getirdiğini onaylarım.

Aslı ıslak imzalıdır

Prof. Dr. Alper Nuhoglu
Enstitü Müdürü

Bu çalışma FDK-2022-10666 numaralı Doktora düzeyindeki BAP projesi kapsamında desteklenmiştir.

Proje No: FDK-2022-10666

Not: Bu tezde kullanılan özgün ve başka kaynaklardan yapılan bildiriş, çizelge, şekil ve fotoğrafların kaynak olarak kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunundaki hükümlere tabidir.

ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU

Doktora Tezi olarak Doç Dr. Tolga AYDIN danışmanlığında sunulan “Araç Rotalama Probleminde GPS Verilerinin Yapay Zekâ Algoritmaları İle Analizi” başlıklı çalışmanın tarafımızdan bilimsel etik ilkelere uyularak yazıldığını, yararlanılan eserlerin kaynakçada gösterildiğini, Fen Bilimleri Enstitüsü tarafından belirlenmiş olan Turnitin Programı benzerlik oranlarının aşılmadığını ve aşağıdaki oranlarda olduğunu beyan ederiz.

Tez Bölümleri	Tezin Benzerlik Oranı (%)	Maksimum Oran (%)
Giriş	1	30
Kuramsal Temeller	0	30
Materyal ve Metot	1	35
Araştırma Bulguları ve Tartışma	1	20
Sonuçlar	0	20
Tezin Geneli	2	25

Not: Yedi kelimeye kadar benzerlikler ile Başlık, Kaynakça, İçindekiler, Teşekkür, Dizin ve Ekler kısımları tarama dışı bırakılabilir. Yukarıdaki azami benzerlik oranları yanında tek bir kaynaktan olan benzerlik oranlarının %5'den büyük olmaması gerekir.

Beyan edilen bilgilerin doğru olduğunu, aksi halde doğacak hukuki sorumlulukları kabul ve beyan ederiz.

Tez Yazarı (Öğrenci)	Tez Danışmanı
Ebru ERDEM	Doç. Dr. Tolga AYDIN
14.7.2025	14.7.2025
İmza:Aslı ıslak imzalıdır	İmza:Aslı ıslak imzalıdır

* Tez ile ilgili YÖKTEZ’de yayınlamasına ilişkin bir engelleme var ise aşağıdaki alanı doldurunuz.

☐ Tezle ilgili patent başvurusu yapılması / patent alma sürecinin devam etmesi sebebiyle Enstitü Yönetim Kurulunun/....../.... tarih ve sayılı kararı ile teze erişim 2 (iki) yıl süreyle engellenmiştir.

☐ Enstitü Yönetim Kurulunun/....../.... tarih ve sayılı kararı ile teze erişim 6 (altı) ay süreyle engellenmiştir.

TEŞEKKÜR

Bu tezin hazırlanmasında akademik bilgi, rehberlik ve desteğiyle katkı sağlayan danışman hocam Sayın Doç. Dr. Tolga AYDIN'a ve ikinci danışman hocam Sayın Prof. Dr. Burak ERKAYMAN'a en içten teşekkürlerimi sunarım. Tez sürecinde fikirleri, eleştirileri ve yol gösterici yorumlarıyla çalışmama katkılar sağlayan University of Twente'den Sayın Dr. Engin TOPAN'a teşekkür ederim.

Tez izleme jürisinde yer alarak değerlendirme, yönlendirme ve önerileriyle süreci zenginleştiren Sayın Prof. Dr. İbrahim Yücel ÖZBEK'e ve Sayın Dr. Öğr. Üyesi Bilal USANMAZ'a katkılarından dolayı teşekkür ederim.

Ayrıca, doktora araştırması süresince saha verilerinin sağlanması, kurum içi lojistik süreçlerinin anlaşılması ve veri kullanımına izin verilmesi konularında destekleriyle katkı sağlayan Aktif Gıda Meşrubat San. ve Tic. A.Ş. Baş Yönetici'si Sayın İzzet TAVAŞI'ya ve Per Retail Solutions (PRS) firmasında Genel Müdür olarak görev yapan Sayın Korkut KUTLUCA'ya içtenlikle teşekkür ederim. Staj sürecim boyunca bilgi ve deneyimleriyle gelişimime değerli katkılar sunan, bu süreçte tarafıma layık görülen “Yılın En Başarılı Stajyer Öğrencisi” ödülüyle motivasyonumu pekiştiren ve doktora çalışmam kapsamında Aktif Gıda ile akademik bağ kurmamı sağlayan kişi olarak, Sayın Korkut KUTLUCA'ya özellikle müteşekkirim.

Ayrıca, bu süreçte manevi destekleriyle yanımda olan biricik annem Meryem'e, babam Turgay'a, hayatıma neşe katan Elmas'a, Şurup'a, kıymetli Burcu'ya, Eren'e, Mehmet'e ve Muhammet Salih'e ve değerli arkadaşlarıma teşekkür ederim.

Bu tez, Atatürk Üniversitesi Bilimsel Araştırma Projeleri Birimi tarafından desteklenen “Araç Rotalama Probleminde GPS Verilerinin Yapay Zekâ Algoritmaları ile Analizi” başlıklı, FDK-2022-10666 numaralı doktora projesi kapsamında gerçekleştirilmiştir. Sağladıkları destekten ötürü Atatürk Üniversitesi BAP birimine teşekkür ederim.

Ebru ERDEM

ÖZET

DOKTORA TEZİ

ARAÇ ROTALAMA PROBLEMİNDE GPS VERİLERİNİN YAPAY ZEKÂ ALGORİTMALARI İLE ANALİZİ

Ebru ERDEM

Danışman: Doç. Dr. Tolga AYDIN

İkinci Tez Danışmanı: Prof. Dr. Burak ERKAYMAN

Amaç: İstanbul’da faaliyet gösteren bir dağıtım şirketinin operasyonel verilerini kullanarak Çok Depolu Araç Rotalama Problemi (ÇDARP) için yapay zekâ (AI) tabanlı rota optimizasyon çözümleri geliştirmektir. GPS ve sevkiyat verilerinin entegrasyonu ile veri odaklı ve uyarlanabilir çözümler hedeflenmiştir.

Yöntem: İki ana veri kaynağı kullanılmıştır: Şirketin sevkiyat kayıtlarını içeren Micro V15 veri seti ve araçların konum, hız vb. bilgilerini kapsayan Mobiliz GPS verileridir. 4 Boyutlu (4B) dönüşümler ile mekânsal ilişkiler analiz edilmiştir. Bu yöntem sınırlı avantajlar sunduğundan, ilerleyen aşamalarda 1B veriler kullanılmıştır. Rota optimizasyonu için meta-sezgisel algoritmalar uygulanmıştır. Bu algoritmaların başarımını artırmak amacıyla makine öğrenimi (ML) destekli başlangıç çözümleri geliştirilmiştir. *S_{mld}*, tarihsel GPS verilerine dayanan güçlü başlangıç çözümleri sunmakta ve meta-sezgisel algoritmaların verimli çalışmasına katkı sağlamaktadır. Ayrıca, Graf Sinir Ağı (GNN), Genişleyen Ağ Yapılarıyla Nöroevrimsel Öğrenme (NEAT), Modelden Bağımsız Meta-Öğrenme (MAML) vb. modeller ile hibrit yapılar oluşturulmuştur.

Bulgular: *S_{mld}* yöntemi, meta-sezgisel algoritmaların başlangıç koşullarını iyileştirerek çözüm kalitesini artırmıştır. GPS tabanlı zaman-mesafe matrisleri, Açık Kaynak Rotalama Motoru (OSRM) API temelli matrislere göre daha güvenilir sonuçlar vermiştir. Dikkat Tabanlı Balçık Küf algoritması (DTBKA), hem yüksek doğruluk hem de güçlü genelleme yeteneği sergilemiştir. Dikkat tabanlı model stabil çözümler üretme başarısıyla öne çıkmıştır. Liste Tabanlı Zamanlama-Transformer (LST-Former) modeli, hata metrikleri açısından iyi performans göstermiştir. MAML tabanlı model, değişken müşteri taleplerine karşı hızlı adaptasyon sağlamış. NEAT ise daha hafif yapıları sayesinde operasyonel uygulanabilirliği artırmıştır.

Sonuç: AI tabanlı modeller, ÇDARP için dinamik, veriye dayalı ve ölçeklenebilir çözümler sunmuştur. ML destekli başlangıç çözümleri, hibrit modellerin başarımını artırmış; düşük maliyetli, yüksek doğruluklu ve donanım uyumlu sonuçlar vermiştir. Klasik yöntemlerin ötesine geçen, pratikte uygulanabilir ve ileri düzeyde optimize edilmiş AI tabanlı çözümler ortaya konulmuştur.

Anahtar Kelimeler: ÇDARP, GPS verisi, AI tabanlı modeller, rota optimizasyonu, meta-sezgisel algoritmalar.

Temmuz 2025, 240 sayfa

ABSTRACT

DOCTORAL DISSERTATION

ANALYSIS OF GPS DATA USING ARTIFICIAL INTELLIGENCE ALGORITHMS IN THE VEHICLE ROUTING PROBLEM

Ebru ERDEM

Supervisor: Assoc. Prof. Dr. Tolga AYDIN

Co-supervisor: Prof. Dr. Burak ERKAYMAN

Purpose: The aim of this thesis is to develop artificial intelligence (AI)-based route optimization solutions for the multi-depot vehicle routing problem (MDVRP) using operational data from a distribution company operating in Istanbul. By integrating GPS and dispatch data, the study aims to propose data-driven and adaptive solutions.

Method: Two primary data sources were used: The Micro V15 dataset, which contains the company's dispatch records, and the Mobiliz GPS dataset, which contains vehicle-related information such as location, speed, and route history. The spatial relationships were initially analysed using 4-dimensional (4D) transformations. However, as this method offered only limited advantages, later phases of the study focussed on the use of 1D data representations. Metaheuristic algorithms were used for route optimization, and to improve their performance, initial solutions based on machine learning (ML) were developed. These solutions, especially those using historical GPS data, provided robust initializations that significantly improved the efficiency of the metaheuristic processes. In addition, hybrid architectures were constructed by integrating advanced models such as Graph Neural Networks (GNN), NeuroEvolution of Augmenting Topologies (NEAT) and Model-Agnostic Meta-Learning (MAML).

Findings: The S_{mld} method improved the solution quality by improving the initial conditions of the metaheuristic algorithms. Time-distance matrices based on GPS data provided more reliable results than those created with the Open Source Routing Machine (OSRM) API. The Attention-Based Slime Mold Algorithm (DTBKA) showed both high accuracy and strong generalization capability, with the attention mechanism contributing to the generation of stable solutions. The List Scheduling–Transformer (LST-Former) model showed favourable performance in terms of error metrics. The MAML-based model enabled quick adaptation to changing customer requirements, while the NEAT model with its lightweight structure increased operational applicability.

Results: The developed AI-based models provided dynamic, scalable and data-driven solutions for MDVRP. ML-assisted initialisation significantly improved the performance of the hybrid models, leading to cost-effective, highly accurate and hardware-compatible results. Overall, the study presents practical, sophisticated and optimised AI solutions that go beyond classical methods.

Keywords: MDVRP, GPS data, AI-based models, route optimization, metaheuristic algorithms.

July 2025, 240 pages

İÇİNDEKİLER

KABUL VE ONAY TUTANAĞI.....	i
ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU	ii
TEŞEKKÜR	iii
ÖZET	iv
ABSTRACT	v
İÇİNDEKİLER.....	vi
TABLolar DİZİNİ.....	viii
ŞEKİLLER DİZİNİ	x
KISALTMALAR VE SİMGELER DİZİNİ	xvi
GİRİŞ.....	1
Problem Tanımı ve Gerçek Dünya Bağlamı	2
Motivasyon ve Literatüre Katkı	3
Veri Tabanlı Yaklaşım ve Çalışmanın Kapsamı	3
GPS Verisinin Önemi ve AI Tabanlı Modellerle Entegrasyonu	4
Çalışmanın Katkısı ve Uygulama Potansiyeli	5
Kademeli Geliştirme Süreci	5
İlk Aşama: keşif çalışması	8
İkinci Aşama: gerçek veri çalışması	9
Üçüncü Aşama: optimizasyon modellerinin geliştirilmesi ve gerçek zamanlı gps verileriyle entegrasyonu	10
Dağıtım sistemlerinde ARP	11
Motivasyon.....	12
Şirket ve operasyonel süreçler	12
Dördüncü Aşama: ileri seviye algoritmaların test edilmesi	18
Beşinci Aşama: hibrit ve AI destekli modellerin geliştirilmesi	21
Motivasyon.....	22
Altıncı Aşama: HLS tabanlı, derinlemesine öğrenme ve evrimsel modellerin bütünleşik uygulaması	24
Motivasyon.....	25
KURAMSAL TEMELLER.....	28

ÇDARP İçin AI Tabanlı Çözümlerdeki Gelişmeler: Son Trendlerin ve Gelecekteki	
Yönlerin İncelenmesi	28
AI teknikleri	36
AI tabanlı meta-sezgisel yaklaşımlar	38
RL ve DL yaklaşımları.....	41
Optimizasyon teknikleri	43
MATERYAL VE METOT	54
Materyal	54
Veri ön işleme	66
Arka Plan.....	70
Metot	72
GPS verisinin 1B'den 4B'ye dönüştürülmesi	72
ML tabanlı topluluk modelleri	72
Yığınlama uygulanma süreci.....	73
DL tabanlı hibrit modeller.....	74
Optimizasyon modellerinin ÇDARP için geliştirilmesi.....	77
Çözüm yöntemleri	80
Takviye öğrenme, DL ve Meta-Sezgisel algoritmaların yenilikçi entegrasyonlarıyla	
ÇDARP optimizasyonu	110
HLS tabanlı, derinlemesine öğrenme ve evrimsel modellerin bütünleşik uygulaması .	128
ARAŞTIRMA BULGULARI	132
TARTIŞMA ve SONUÇ	204
Sonuç ve Öneriler.....	205
KAYNAKÇA	208
ÖZGEÇMİŞ.....	220

TABLÖLAR DİZİNİ

Tablo 1. Başlangıç Çözüm Kümesinin Optimizasyon Sürecine Etkileri.....	17
Tablo 2. En Çok Atıf Alan Makaleler	33
Tablo 3. 51 Kümeye Ait Etiketler	35
Tablo 4. AI Tabanlı Meta-Sezgisel Algoritmalar Üzerine Çalışmaların Analizi	39
Tablo 5. RL ve DL Uygulamaları için Yapılan Çalışmaların Analizi	42
Tablo 6. Veri Setlerinde Mevcut Ayrıntılı Bilgiler.....	54
Tablo 7. Micro V15 Veri tabanında Örnek Bir Kesitin Ayrıntıları	56
Tablo 8. GPS Veri Kümesinin Özellikleri	57
Tablo 9. Araçların Özelliklerine İlişkin Detaylar	65
Tablo 10. Bölütleme Algoritması Sözde Kodu	68
Tablo 11. Farklı Yöntemlerle İşlenen Veri Setinin Bellek Kullanım Sonuçları	68
Tablo 12. Veri Boyutlarındaki Değişiklikler	69
Tablo 13. Geriye Doğru Eleme Yöntemi Sahte Kodu	71
Tablo 14. CHD Veri Setinde Katsayılar (Coef) ve P-Değerlerinin Özeti	71
Tablo 15. WnvD Veri Setinde Katsayılar (coef) ve P-Değerlerinin Özeti	71
Tablo 16. GPS Veri setindeki Katsayı ve p-değeri Özet Bilgisi.....	72
Tablo 17. Problemin Matematiksel Formülasyonu	78
Tablo 18. Sevkiyat Veri Modeli ve Coğrafi Kodlama Sonrasında Elde Edilen Veri Çerçevesinin Özeti	83
Tablo 19. OSRM API Parametre Detayları	84
Tablo 20. OSRM API ile Zaman-Mesafe Matrisi Oluşturulması	84
Tablo 21. Mikro V15 ve GPS Adres Benzerlik Sonucu Oluşturulan Veri Seti	87
Tablo 22. GPS Verileriyle Zaman-Mesafe Matrisi Oluşturulması	88
Tablo 23. ÇDARP Çözümü için ML Modelleri	93
Tablo 24. Farklı Başlangıç Çözüm Yaklaşımlarıyla ÇDARP Çözümü için BT Algoritması Sözde Kodu	99
Tablo 25. Çeşitli Başlangıç Çözüm Yaklaşımlarıyla ÇDARP Çözümü için MBT Algoritması Yalancı Kodu	101
Tablo 26. Çeşitli Başlangıç Çözüm Yaklaşımlarıyla ÇDARP Çözümü için TA Algoritması Yalancı Kodu	103

Tablo 27. Karşılaştırmalı MAE ve MSE Değerleri: CHD ve WnvD Veri Setleri Üzerinde Uygulanan Yöntemlerin Performans Analizi.....	133
Tablo 28. Karşılaştırmalı R^2 ve RMSE Değerleri: CHD ve WnvD Veri Setleri Üzerinde Uygulanan Yöntemlerin Performans Analizi.....	134
Tablo 29. Karşılaştırmalı MAE, MSE, R^2 ve RMSE değerleri: GPS Veri Seti Üzerinde Uygulanan Yöntemlerin Performans Analizi.....	141
Tablo 30. ML Modellerinin Performansının Karşılaştırılması	147
Tablo 31. BT ve MBT Sayısal Parametreleri	151
Tablo 32. TA Sayısal Parametreleri.....	151
Tablo 33. Dikkat Mekanizması Parametre Detayları.....	162
Tablo 34. Q-Değerlerinin Güncellenmesi (Bellman Eşitliği).....	163
Tablo 35. Replay Memory ve Mini-Batch Eğitimi.....	163
Tablo 36. Q-Ağının Eğitilmesi ve Nihai Rota Optimizasyonu.....	164
Tablo 37. BMEO Modelindeki Parametreler.....	164
Tablo 38. Diferansiyel Evrim Algoritmasının Parametre Detayları	165
Tablo 39. GNN-LSTM-RL ve GNN-RL Modellerindeki GNN Katmanlarının Sayısal Parametreleri	165
Tablo 40. GNN-LSTM-RL modelindeki LSTM Katmanlarının Sayısal Parametreleri	166
Tablo 41. PPO Modeli Parametreleri Sayısal Parametreleri.....	166

ŞEKİLLER DİZİNİ

Şekil 1. AI destekli ÇDARP çözüm sürecinin aşamalı yol haritası	7
Şekil 2. Altı aşamalı modelleme sürecine ait yöntemsel açıklama ve karar noktaları	8
Şekil 3. Ürün hazırlama ve dağıtım süreçlerine ilişkin bir örnek	13
Şekil 4. Şirket sevk işlemi ayrıntıları	15
Şekil 5. 2013-2023 Sevkiyat haritası QGIS gösteriminde: Tüm müşteri düğümlerinin görselleştirilmesi	15
Şekil 6. Gerçek zamanlı GPS ve operasyonel verilere dayalı çok aşamalı dağıtım ağı modelleme ve rotalama süreci	16
Şekil 7. Zaman-mesafe verisi ile güçlendirilmiş araç rotalama için optimizasyon algoritmalarının karşılaştırmalı süreci	18
Şekil 8. Zaman-mesafe verisi ile güçlendirilmiş araç rotalama için optimizasyon algoritmalarının karşılaştırmalı süreci	22
Şekil 9. Zaman-mesafe verisi ile güçlendirilmiş araç rotalama için optimizasyon algoritmalarının karşılaştırmalı süreci	25
Şekil 10. Sistematiik derleme sürecinin adımları	31
Şekil 11. Arama sonuçlarına göre incelenen makale sayısı	32
Şekil 12. Scopus ve WoS için 2015'ten 2024'e kadar olan makale yayın eğilimleri	32
Şekil 13. Etiket bazlı anahtar kelime eş-zamanlı haritası	34
Şekil 14. CHD ve WnvD'ye ait coğrafi noktaların koordinatlarının çizilmesi ve harita üzerinde gösterilmesi	55
Şekil 15. Veri kümesindeki enlem ve boylam değerlerinin istatistiksel özeti	56
Şekil 16. Mobiliz araç takip sisteminin yetkili etkileşimli web sayfası arayüzü	57
Şekil 17. ÇDARP örnek model	58
Şekil 18. Gerçek GPS verilerine dayalı araç hareket grafiği, depo-müşteri etkileşimleri	59
Şekil 19. Haziran 2021'e ait merkez-uç ağı	60
Şekil 20. Haziran 2022'ye ait merkez-uç ağı	60
Şekil 21. 4 depolu, öğle saatinde sipariş veren 29 müşterili ve 31 araçlı ÇDARP'nin yapısal karmaşıklığını gösteren örnek model	61
Şekil 22. Müşteri ve depo lokasyonları ile talep büyüklüklerinin coğrafi dağılımı	62
Şekil 23. Müşteri konumları ve sipariş yoğunluklarının coğrafi dağılımı	63
Şekil 24. Araç kapasite dağılımı ve talep karşılama analizi	65

Şekil 25. Müşteri ve depolar arasındaki normalize mesafe ısı haritası	65
Şekil 26. 6 Aylık GPS verilerine ait bilgiler	67
Şekil 27. Veri ön-işleme adımları	67
Şekil 28. Şirket sevkiyat işlemleri-dağıtım veri tabanı (DB) yapısı (Sevkiyat veri modeli) ..	69
Şekil 29. CHD ve WnvD'ye ait korelasyon ısı haritası.....	72
Şekil 30. GPS veri setinin korelasyon ısı haritası	72
Şekil 31. ML ve DL yöntemleriyle sürücü seyahat verileri için özellik seçimi ve tahmin modelleme çerçevesi	77
Şekil 32. Geleneksel ÇDARP çözümü	80
Şekil 33. Önerilen ÇDARP çözümü.....	81
Şekil 34. ÇDARP çözüm yaklaşımı	81
Şekil 35. Önerilen ÇDARP modelinde teslimatların entegrasyonu ve etki alanları	82
Şekil 36. Müşteri ve depo konumları için coğrafi kodlama süreci	83
Şekil 37. GPS verilerine dayalı zaman-mesafe matrisi oluşturma ve rota optimizasyonu süreci	89
Şekil 38. Farklı araç gruplarının adres ziyaret yoğunluklarının dağılımı.....	91
Şekil 39. Farklı araç gruplarına ait ziyaret edilen kayıtların indeks dağılımı	91
Şekil 40. Araç gruplarının zaman içindeki toplam mesafe (km) dağılımları	91
Şekil 41. ÇDARP çözümüne yönelik AI tabanlı yaklaşımın I. aşamasının gösterimi	94
Şekil 42. Hibrit modelin genel mimarisi	96
Şekil 43. ÇDARP çözümüne yönelik AI tabanlı yaklaşımın II. aşamasının gösterimi	97
Şekil 44. Farklı başlangıç yaklaşımlarıyla BT tabanlı ÇDARP çözüm süreci	100
Şekil 45. Farklı başlangıç yaklaşımlarıyla MBT tabanlı ÇDARP çözüm süreci	102
Şekil 46. Farklı başlangıç yaklaşımlarıyla TA tabanlı ÇDARP çözüm süreci	103
Şekil 47. Araç gruplarına göre depoya geri dönüş süresi dağılımları	107
Şekil 48. Depoya geri dönüş süresi değişkeninin histogram grafiği	108
Şekil 49. Araç palet kapasitelerine göre toplam yükleme miktarları	108
Şekil 50. Araç grubu ve palet kapasitesine göre yükleme miktarlarının ısı haritası	108
Şekil 51. Araç palet kapasitelerine göre depoya geri dönüş süresi dağılımı	109
Şekil 52. Günlük zaman dilimine göre araçların depoya geri dönüş süreleri.....	109
Şekil 53. Araç gruplarına göre kat edilen mesafe dağılımı (yol (km)).....	109
Şekil 54. ÇDARP çözüm aşamaları	110
Şekil 55. ÇDARP çözüm yaklaşımlarına yönelik ileri seviye algoritmalar (BKA, DQN, Dikkat Mekanizması ve DTBKA)	110

Şekil 56. ÇDARP çözüm yaklaşımlarına yönelik hibrit Algoritmalar (GNN-LSTM-RL, GNN-RL ve BMEO).....	111
Şekil 57. BKA akış diyagramı: Rota keşfi ve optimizasyonu	112
Şekil 58. Araç rota optimizasyonu için BKA tabanlı yaklaşım	112
Şekil 59. DQN optimizasyon süreci	114
Şekil 60. Araç rota optimizasyonu için DQN (Derin Q-Ağları) tabanlı yaklaşım	114
Şekil 61. Dikkat mekanizması optimizasyon süreci.....	116
Şekil 62. Araç rota optimizasyonu için dikkat tabanlı yaklaşım	116
Şekil 63. DTBKA optimizasyon süreci	119
Şekil 64. Araç rota optimizasyonu için DTBKA yaklaşımı	119
Şekil 65. GNN-LSTM-RL optimizasyon süreci	120
Şekil 66. Araç rota optimizasyonu için GNN-LSTM-RL yaklaşımı.....	121
Şekil 67. GNN-RL optimizasyon süreci.....	123
Şekil 68. Araç rota optimizasyonu için GNN-RL yaklaşımı.....	123
Şekil 69. BME optimizasyon süreci	124
Şekil 70. Araç rota optimizasyonu için BME yaklaşımı	124
Şekil 71. Araç kapasitesi, zaman maliyeti ve ceza değişkenleri için üyelik fonksiyonlarının tanımlanması	125
Şekil 72. Araç kapasitesi ve zaman maliyetine dayalı ceza değeri hesaplama süreci.....	126
Şekil 73. ÇDARP optimizasyonu için hibrit AI tabanlı modellerin akış şeması	129
Şekil 74. Eğitim ve test noktalarının görsel dağılımı	132
Şekil 75. Model performanslarının grafiksel özeti	135
Şekil 76. Veri setlerindeki gerçek boylam değerlerinin (y) grafiksel dağılımları.....	135
Şekil 77. CHD verisetinde CNN ve GPSCNN modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	135
Şekil 78. CHD verisetinde GPSLSTM modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon.....	136
Şekil 79. CHD verisetinde GPSCNN ve ön-eğitilmiş modellerle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	136
Şekil 80. CHD verisetinde ML ve topluluk modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	137
Şekil 81. WnvD verisetinde CNN ve GPSCNN modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	138
Şekil 82. WnvD verisetinde GPSLSTM modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon.....	138

Şekil 83. WnvD verisetinde GPSCNN ve ön-eğitilmiş modellerle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	139
Şekil 84. WnvD verisetinde ML ve topluluk modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	140
Şekil 85. GPS verisetinde ML ve topluluk modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	142
Şekil 86. GPS verisetinde ML modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon	143
Şekil 87. Diğer algoritmalar için pivot özelliği taşıyan GPS-tabanlı zaman mesafe matrisi .	143
Şekil 88. ML modelleri uygulandıktan sonra performans sonuçları	144
Şekil 89. MAE, MSE ve R^2 ölçütlerine göre ML modellerinin karşılaştırmalı performans sıralamaları	146
Şekil 90. ML modelleri (KNR, RFR, MLPR, XGB) uygulandıktan sonra elde edilen minimum maliyetli rota ve test metriği sonuçları	147
Şekil 91. ML modelleri (DTR, LR, RFR-DL) uygulandıktan sonra elde edilen minimum maliyetli rota ve test metriği sonuçları	148
Şekil 92. ML tabanlı başlangıç modelleri (sırasıyla, XGB, RFR, MLPR, LR, KNR ve DTR) için özellik etkilerinin SHAP kuvvet grafiği ile görselleştirilmesi	149
Şekil 93. ML tabanlı başlangıç modellerine ait SHAP kuvvet diyagramı görselleştirmeleri (sırasıyla MLPR, XGB, KNR, LR, DTR ve RFR)	150
Şekil 94. Üç farklı başlangıç çözümüne göre filonun kat ettiği toplam süre	152
Şekil 95. Sırasıyla OSRM API ve GPS verileriyle oluşturulan zaman-mesafe matrisleri kullanılarak elde edilen MBT performans sonuçları	155
Şekil 96. Sırasıyla OSRM API ve GPS verileriyle oluşturulan zaman-mesafe matrisleri kullanılarak elde edilen TA performans sonuçları	155
Şekil 97. BT algoritması kullanılarak başlangıç stratejileri ve test koşulları boyunca toplam seyahat süresi ısı haritası	157
Şekil 98. Farklı başlangıç stratejilerine göre BT algoritmasının iterasyonlar boyunca yakınsama davranışı	158
Şekil 99. BT algoritması için başlangıç stratejilerine (S_{mld} , S_{rd} , S_{dyd}) göre kutu grafiği	158
Şekil 100. MBT algoritması kullanılarak başlangıç stratejileri ve test koşulları boyunca toplam seyahat süresi ısı haritası	158
Şekil 101. MBT algoritmasının farklı başlangıç stratejilerine göre iterasyonlar boyunca yakınsama davranışı	159
Şekil 102. MBT algoritması için başlangıç stratejilerine (S_{mld} , S_{rd} , S_{dyd}) göre kutu grafiği	159

Şekil 103. TA algoritması kullanılarak başlangıç stratejileri ve test koşulları boyunca toplam seyahat süresi ısı haritası	159
Şekil 104. TA algoritmasının farklı başlangıç stratejilerine göre iterasyonlar boyunca yakınsama davranışı	160
Şekil 105. TA algoritması için başlangıç stratejilerine (S_{mld} , S_{rd} , S_{dyd}) göre kutu grafiği	160
Şekil 106. Train ve test noktalarının görsel dağılımı	161
Şekil 107. DQN algoritması için rota sonuçları	177
Şekil 108. BKA algoritması için rota sonuçları	178
Şekil 109. Dikkat mekanizması için rota sonuçları	178
Şekil 110. DTBKA algoritması için rota sonuçları	178
Şekil 111. GNN-LSTM-RL algoritması için rota sonuçları	179
Şekil 112. GNN-RL algoritması için rota sonuçları	179
Şekil 113. BME algoritması için rota sonuçları	179
Şekil 114. Dikkat mekanizması için eğitim ve test ödül analiz grafiği	180
Şekil 115. BKA için eğitim ve test ödül analiz grafiği	180
Şekil 116. DTBKA için eğitim ve test ödül analiz grafiği	180
Şekil 117. DQN için eğitim ve test ödül analiz grafiği	181
Şekil 118. GNN-LSTM-RL eğitim ve test ödül analiz grafiği	181
Şekil 119. GNN-RL eğitim ve test ödül analiz grafiği	182
Şekil 120. BME eğitim ve test ödül analiz grafiği	183
Şekil 121. Dikkat mekanizması ve DTBKA için sırasıyla eğitim ve test kayıp karşılaştırması	183
Şekil 122. BKA için foldlar arasında fitness değeri dağılımı ve kullanım analizi	184
Şekil 123. Dikkat mekanizması için foldlar arasında fitness değeri dağılımı ve kullanım analizi	185
Şekil 124. DTBKA için foldlar arasında fitness değeri dağılımı ve kullanım analizi	186
Şekil 125. DQN için foldlar arasında fitness değeri dağılımı ve kullanım analizi	186
Şekil 126. GNN-LSTM-RL için foldlar arasında fitness değeri dağılımı ve kullanım analizi	187
Şekil 127. GNN-RL için foldlar arasında fitness değeri dağılımı ve kullanım analizi	188
Şekil 128. BME algoritmasında bulanık kümeleme sonrasında eğitim ve test verisi	189
Şekil 129. BME için foldlar arasında fitness değeri dağılımı ve kullanım analizi	189
Şekil 130. Modellere göre eğitim ve test hatalarının karşılaştırması (logaritmik ölçekle) ...	190
Şekil 131. Her modelin performans eğilimi	190
Şekil 132. Eğitim ve test arasındaki hata oranında yüzdelik iyileşme	190

Şekil 133. Modeller arasındaki hata dağılımının sankey diyagramı (BKA=SMA, DTBKA=AT SMA, GNN-RL=GNN+RL, GNN-LSTM-RL=GNN+LSTM+RL, Dikkat Mekanizması=Attention)	191
Şekil 134. Modellere göre hata dağılımı boxplot grafikleri	191
Şekil 135. Modeller arasında MAE ve MSE değerlerinin ısı haritası	192
Şekil 136. Tüm modeller için MAE ve MSE dağılım grafiği (Scatter Plot)	192
Şekil 137. Tüm algoritmalar için eğitim ve test ödül analizi özeti (BKA=SMA, DTBKA=AT SMA, GNN-RL=GNN+RL, GNN-LSTM-RL=GNN+LSTM+RL, Dikkat Mekanizması=Attention)	192
Şekil 138. AST-Former modeli sonuçları	195
Şekil 139. ALT-Former modeli sonuçları	196
Şekil 140. LST-Former modeli sonuçları	197
Şekil 141. LST-MetaDRL modeli sonuçları	198
Şekil 142. ALT-MetaDRL modeli sonuçları	199
Şekil 143. AST-MetaDRL modeli sonuçları	200
Şekil 144. NRL modeli sonuçları	201
Şekil 145. AI tabanlı yönlendirme modelleri genelinde normalleştirilmiş başarı oranlarının histogramı	202
Şekil 146. Model performans değerlendirmesinin radar grafik gösterimi	202
Şekil 147. %95 güven aralıklarıyla başarı oranı karşılaştırması	202
Şekil 148. Metodolojik kalite değerlendirmesi-ısı haritası	203
Şekil 149. Metodolojik kalite değerlendirme radar çizelgesi	203

KISALTMALAR VE SİMGELER DİZİNİ

1B	: 1 Boyut
4B	: 4 Boyut
ABC	: Artificial Bee Colony (Yapay Arı Kolonisi)
ACO	: Ant Colony Optimization (Karıncı Kolonisi Optimizasyonu)
AI	: Artificial Intelligence (Yapay Zekâ)
ALAP	: As Late as Possible Scheduling (En Geç Zamanlama)
ALT	: As Late as Possible Time Strategy (En Geç Zamanlama Stratejisi)
ALT-Former	: En Geç Zamanlama Stratejisi -Transformer Hibrit Optimizasyon Yöntemi
ALT-MetaDRL	: ALT-DRL-MAML Mekanizması
ANN	: Artificial Neural Network (Yapay Sinir Ağı)
ARP	: Araç Rotalama Problemi
ASAP	: As Soon As Possible Scheduling (En Erken Zamanlama)
AST	: As Soon As Possible Time Strategy (En Erken Zamanlama Stratejisi)
AST-Former	: En Erken Zamanlama Stratejisi -Transformer Hibrit Optimizasyon Yöntemi
AST-MetaDRL	: AST-DRL-MAML Mekanizması
BKA	: Balçık Küfü Algoritması
BME	: Bulanık Mantık-Evrimsel
BMEO	: Bulanık Mantık Evrimsel Optimizasyon
BT	: Benzetilmiş Tavlama
CHD	: California Housing Data Set (California Konut Veri Seti)
CNN	: Convolutional Neural Network (Evrimsel Sinir Ağı)
Coef	: Coefficients (Katsayılar)
ÇDARP	: Çok Depolu Araç Rotalama Problemi
DL	: Deep Learning (Derin Öğrenme)
DQN	: Deep Q-Network (Derin Q-Ağı)
DRL	: Deep Reinforcement Learning (Derin Pekiştirmeli Öğrenme)
DTBKA	: Dikkat Tabanlı Balçık Küf Algoritması
DTR	: Decision Tree Regression (Karar Ağacı Regresyonu)
DTW	: Dynamic Time Warping (Dinamik Zaman Savaşımı)
EA	: Evrimsel Algoritma

FPGA	: Field-Programmable Gate Array (Saha Programlanabilir Kapı Dizisi)
GA	: Genetic Algorithm (Genetik Algoritma)
GAN	: Generative Adversarial Network (Üretici Çekişmeli Ağlar)
GCN	: Graph Convolutional Network (Graf Evrişimsel Ağ)
GIS	: Geographic Information Systems (Coğrafi Bilgi Sistemleri)
GNN	: Graph Neural Network (Graf Sinir Ağı)
GPS	: Global Positioning System (Küresel Konumlandırma Sistemi)
HLS	: High-Level Synthesis (Yüksek Seviyeli Sentez)
INS	: Inertial Navigation System (Ataletsel Seyrüsefer Sistemi)
IoT	: Internet of Things (Nesnelerin İnterneti)
IWD	: Intelligent Water Drops (Zeki Su Damlası)
KDE	: Kernel Density Estimation (Frekans ve Yoğunluk Dağılım Grafiği)
kNN	: k-Nearest Neighbor (k-En Yakın Komşu Algoritması)
KNR	: K-Nearest Neighbor Regression (K-En Yakın Komşu Regresyonu)
LR	: Linear Regression (Doğrusal Regresyon)
LST	: List Scheduling Technique (Liste Tabanlı Zamanlama)
LST-Former	: Liste Tabanlı Zamanlama–Transformer
LSTM	: Long Short-Term Memory (Uzun Kısa Süreli Bellek)
LST-MetaDRL	: LST-DRL-MAML Mekanizması
MAE	: Mean Absolute Error (Ortalama Mutlak Hata)
MAML	: Model-Agnostic Meta-Learning (Modelden Bağımsız Meta-Öğrenme)
MAPE	: Mean Absolute Percentage Error (Ortalama Mutlak Yüzde Hata)
MBT	: Modifiye Benzetilmiş Tavlama
MetaDRL	: Model-Agnostic Meta-Pekiştirmeli Öğrenme (Model-Agnostic Meta-Pekiştirmeli Öğrenme)
ML	: Machine Learning (Makine Öğrenimi)
MLP	: Multi-Layer Perceptron (Çok Katmanlı Algılayıcı)
MLPRegressor	: Multi-Layer Perceptron Regressor (Çok Katmanlı Algılayıcı Regresyonu)
MLSSA	: Multi-Leader Salp Swarm Algorithm (Çoklu Liderli Salp Sürüsü Algoritması)
MSE	: Mean Squared Error (Ortalama Kare Hata)
NEAT	: NeuroEvolution of Augmenting Topologies (Genişleyen Ağ Yapılarıyla Nöroevrimsel Öğrenme)
NGL	: Neural Graph Learning (Sinirsel Grafik Öğrenmesi)

NRL	: NeuroEvolution of Augmenting Topologies with Reinforcement Learning (NEAT-RL)
OpenStreetMap	: Açık Kaynak Harita Projesi
OSRM	: Open Source Routing Machine (Açık Kaynak Rotalama Motoru)
PAP10	: Prediction Accuracy within 10% (%10 İçindeki Doğru Tahmin Oranı)
POFP	: Proportion of Over-Fulfilled Predictions (Aşırı Karşılanan Tahmin Oranı)
PPO	: Proximal Policy Optimization (Yakınsak Politika Optimizasyonu)
PSO	: Particle Swarm Optimization (Parçacık Sürü Optimizasyonu)
QGIS	: Quantum Geographic Information System (Kuantum Coğrafi Bilgi Sistemi)
RBF	: Radial Basis Function (Radyal Tabanlı Fonksiyon)
RFR	: Random Forest Regression (Rastgele Orman Regresyonu)
RL	: Reinforcement Learning (Pekiştirmeli Öğrenme)
RMSE	: Root Mean Squared Error (Kök Ortalama Kare Hata)
RNN	: Recurrent Neural Network (Yinelenen Sinir Ağı)
SepCNN	: Separable Convolutional Neural Network (Ayrılabilir Evrişimsel Sinir Ağı)
SHAP	: SHapley Additive exPlanations (Shapley Katkı Açıklamalar)
SVR	: Support Vector Regression (Destek Vektör Regresyonu)
TA	: Tabu Search (Tabu Arama)
TF-IDF	: TfidfVectorizer (Terim Frekansı Temelli Metin Vektörleştirici)
UF-MEP	: Urban Freight – Mobility and Energy Performance (Kentsel Yük Taşımacılığı Enerji Verimliliği)
VAE	: Variational Autoencoder (Varyasyonel Otokodlayıcılar)
WdCNN	: Wide and Deep Convolutional Neural Network (Geniş ve Derin Evrişimli Sinir Ağı)
WMRA	: Wavelet Multi-Resolution Analysis (Wavelet Çoklu Çözünürlük Analizi)
WnvD	: West Nile Virus Data Set (Batı Nil Virüsü Veri Seti)
XGB	: eXtreme Gradient Boosting (Aşırı Gradyan Artırma)

GİRİŞ

Çağdaş dağıtım sistemlerinin artan karmaşıklığı, firmaların etkin lojistik planlamaya yönelik daha entegre, veri odaklı ve AI destekli çözümler geliştirmesini zorunlu kılmıştır. Özellikle büyük şehirlerde faaliyet gösteren dağıtım şirketleri için, çoklu depo kullanımı, birden fazla sefer gereksinimi ve bölünmüş teslimatlar gibi karmaşık lojistik koşullar altında optimize rota belirlemek, klasik Araç Rotalama Problemi (ARP) çözümlerinin yetersiz kalmasına neden olmaktadır. Artan müşteri talepleri, sık değişen trafik koşulları ve zaman kısıtları, lojistik problemlerini stratejik açıdan daha kritik hâle getirmiştir. Bu bağlamda, yalnızca mesafeye dayalı planlama sunan geleneksel ARP yaklaşımlarının yerine; trafik yoğunluğu, teslimat sıralaması, araç kapasiteleri ve depo geri dönüş süreleri gibi çok boyutlu değişkenleri eşzamanlı yöneten daha esnek modeller olan Çok Depolu Araç Rotalama Problemi (ÇDARP) (Renaud *et al.* 1996) çözümleri gerekmektedir. Bu tez çalışması, gerçek dünya koşullarına uyumlu bir çözüm geliştirmek amacıyla, 2013–2023 yılları arasında bir içecek dağıtım şirketinden elde edilen 10 yıllık Mikro Sevkiyat Veri Tabanı (Mikro V15) (Belbag *et al.* 2009) sipariş verisi ile 2020–2023 arası Mobiliz GPS (Anonymous 2023) verilerinin bütünleştirilmesini esas alarak, AI destekli büyük ölçekli bir ÇDARP çözümü önermektedir. Mevcut sistemde, araçlar önceden belirlenmiş bir depodan çıkış yapmakta ve sahadaki talepler doğrultusunda günlük olarak teslimatlarını tamamlamaktadır. Ancak bu süreçte klasik OSRM API (Anonymous 2023) verilerine dayalı sabit süreli tahminlerin, trafik ve coğrafi değişkenlikleri yeterince yansıtmadığı görülmüştür. GPS verilerinin entegrasyonu sayesinde, trafik yoğunluğu, seyahat süresi ve müşteri davranışı gibi lojistik parametreler doğrudan öğrenilmiş ve veri odaklı tahminleme algoritmalarıyla desteklenmiştir. Ayrıca önerilen AI tabanlı sistemde, Transformer mimarileri, pekiştirmeli öğrenme (RL), MAML ve NEAT gibi ileri seviye algoritmalar kullanılarak klasik rotalama yaklaşımlarının ötesine geçilmiştir. Bu yapılar, özellikle sevkiyat işlemlerinde ortaya çıkan zaman duyarlılığı, teslimat sırası önceliği ve çoklu sefer gibi karmaşık operasyonel kısıtları optimize etmede önemli başarı sağlamıştır. GPS verilerinden türetilen zaman farkları, pozisyonel kodlama ve dikkat katmanları (Vaswani *et al.* 2017) aracılığıyla AI modellerine entegre edilerek daha gerçekçi rota tahminleri elde edilmiştir. Çalışmada kullanılan veri temelli yaklaşım, yalnızca yazılım düzeyinde değil, aynı zamanda yüksek seviyeli sentez (HLS) uyumlu donanım tasarımlarına da olanak tanımakta ve bu yönüyle lojistikte gerçek zamanlı entegrasyon için uygulamaya dönük güçlü bir çözüm sunmaktadır.

Problem Tanımı ve Gerçek Dünya Bağlamı

ÇDARP, birden fazla deponun bulunduğu ve her bir deponun farklı müşterilere hizmet verdiği karmaşık lojistik problemlerden biridir. Bu problem türü, gerçek dünya lojistiğinde sıklıkla karşılaşılmaya rağmen hem hesaplama karmaşıklığı hem de zaman kısıtları nedeniyle geleneksel yöntemlerle çözülmesi oldukça zordur. Dağıtım sistemleri, yalnızca mesafe optimizasyonu ile değil; aynı zamanda aracın kapasitesi, teslimat zamanı ve depo geri dönüş süreleri gibi birden çok öz nitelikli parametreyi içermektedir. Gerçek zamanlı GPS verileri, lojistik süreçlerin planlanmasında ve özellikle zaman-mesafe matrislerinin oluşturulmasında önemli avantajlar sunar. Bu veriler, rota optimizasyonu sürecinde zamanlamayı doğrudan etkileyen en önemli faktörlerden biri olan "araç dönüş sürelerinin" hassasiyetle hesaplanmasına olanak tanır. Son yıllarda yapılan çalışmalar, GPS verileri ile rota süresi, konum tespiti, müşteri davranışı ve rota planlama doğruluğu üzerindeki etkisi gibi lojistik parametreler tahmin edilmiş ve bu tahminlerin doğruluğu yüksek bulunmuştur (Ghazikhani *et al.* 2024; Žunić *et al.* 2018; Hughes *et al.* 2019; Elmasri *et al.* 2023; Kodera *et al.* 2025; Soh *et al.* 2024; Choudhry & Qian, 2024). Žunić *et al.* (2018) tarafından yapılan bir çalışmada, 15.000'den fazla müşteriye hizmet veren bir dağıtım firmasının GPS verileri kullanılarak geliştirilen sistemin, rota planlamasında klasik yöntemlere göre daha başarılı sonuçlar verdiği gösterilmiştir. Ayrıca, Hughes *et al.* (2019) çalışmasında GPS ve Açık Kaynak Harita Projesi (OpenStreetMap) verileri ile teslimat sürelerinin tahmin edilmesinde k-En Yakın Komşu Algoritması (kNN) algoritmasının en başarılı yöntem olduğu vurgulanmıştır. Ghazikhani *et al.* (2024), GPS verileri ve regresyon algoritmaları kullanarak kamyonların seyahat sürelerini tahmin etmeye yönelik bir çalışma sunmuştur. Bu çalışma, GPS verilerinin seyahat süresi tahmininde yüksek doğruluk sağladığını göstermektedir. Kodera *et al.* (2025), Tokyo Metropolen Bölgesi'nden elde edilen GPS verilerini kullanarak kentsel taşımacılıkta teslimat zaman dilimi seçim modelini geliştirmiştir. Bu model, GPS verilerinin teslimat zamanlaması tahmininde etkili olduğunu ortaya koymaktadır. Elmasri *et al.* (2023), büyük ölçekli GPS verileri kullanarak ulaşım ağlarında seyahat süresi tahmini için istatistiksel yöntemler geliştirmiştir. Bu çalışma, GPS verilerinin seyahat süresi dağılımını tahmin etmede etkili olduğunu göstermektedir. Lojistik problemler kapsamında yapılan güncel çalışmalar, GPS verilerinin rota planlamasında sağladığı avantajları daha da pekiştirmektedir. Soh *et al.* (2024) tarafından yapılan kapsamlı bir analiz çalışmasında, GPS koordinatlarının gecikme kaynaklı örüntüleri tanımlamak ve teslimat rotalarındaki sapmaları belirlemek için kullanıldığı ve bu sayede zamanında teslimat performansının artırılabilirdiği gösterilmiştir. Choudhry ve Qian (2024) ise, kamyon erişilebilirliği ve zaman-mesafe verimliliğini ölçmek amacıyla geliştirdikleri Kentsel Yük Taşımacılığı Enerji

Verimliliği (UF-MEP) modeli ile, GPS verilerinin kent içi teslimat erişimindeki potansiyel faydalarını vurgulamıştır.

Tüm bu çalışmalar, GPS verilerinin yalnızca rota uzunluğunun değil; aynı zamanda trafik yoğunluğu, teslimat noktası sapmaları, sürücü davranışları ve zaman pencereleri gibi değişkenlerin yönetiminde nasıl kritik bir rol oynadığını ortaya koymaktadır. Böylece, teslimat işlemlerinde daha esnek, çevik ve öğrenebilen rota planlama stratejilerinin geliştirilmesi mümkün olmaktadır.

Motivasyon ve Literatüre Katkı

Literatürdeki çoğu ÇDARP çalışması, ya sınırlı sentetik verilerle test edilmiş ya da GPS tabanlı trafik ve rota bilgilerini ihmal etmiştir (Noori & Ghannadpour, 2012; Stodola, 2018; Ramos *et al.* 2020; Correll *et al.* 2022; Arishi & Krishnan, 2023). Ayrıca, çoğu çalışma, teslimatların sadece tek turda ve tek depo üzerinden yapıldığını varsaymaktadır (Liu & Wang, 2022; Gupta *et al.* 2022; Yang & Chen, 2024; Xu *et al.* 2022; Yang *et al.* 2023). Bu çalışmanın temel motivasyonu, dağıtım ağının doğal akışını GPS verileriyle öğlenerek modellemek, günlük operasyonel kısıtları entegre etmek ve özelleştirilmiş çözümler sunabilmektir. Bu bağlamda çalışma, klasik meta-sezgisel ve sezgisel algoritmalarla sınırlı kalmayıp, Transformer tabanlı mimariler, RL, MAML ve HLS uyumlu sistemler geliştirerek literatüre büyük ölçekli, veri odaklı ve uygulamaya dönük katkı sunmayı hedeflemiştir.

Ayrıca, önerilen ÇDARP çözümünün şehir içi dağıtım operasyonlarındaki rolü de dikkate değerdir. Özellikle şehir içi dağıtım süreçlerinde, GPS verilerine dayalı olarak geliştirilen zaman-mesafe matrisleri, teslimat rotalarının trafik yoğunluğu ve seyahat süresi değişkenliğine göre gerçek zamanlı olarak optimize edilmesini sağlamaktadır. Çalışmada kullanılan ML destekli başlangıç rotalama stratejileri ve geçmişe dayalı rota öğrenimi, kentsel lojistik uygulamalarında rota tahmin performansını iyileştirmiştir. Böylece, önerilen sistem yalnızca teorik bir çözüm önermekle kalmayıp, kentsel dağıtım sistemleri kapsamında zaman duyarlı teslimatlar ve çoklu sefer planlamaları gibi operasyonel zorluklara da gerçekçi yanıtlar sunmuştur. Bu yönüyle, çalışma yalnızca ÇDARP çözümünü değil, aynı zamanda kentsel ölçekte rota optimizasyonunun operasyonel faydalarını da ortaya koymaktadır.

Veri Tabanlı Yaklaşım ve Çalışmanın Kapsamı

Çalışma, 2013-2023 yılları arasında bir dağıtım şirketinin Mikro V15 sisteminden alınan 1,619 tabloyu ve 4,784,003 satırlık Mobiliz GPS verisini kapsamaktadır. Sipariş, stok, cari hesap ve sevk irsaliyesi gibi ticari belgeler ile GPS tabanlı konum, hız ve süre bilgileri birleştirilerek zaman-mesafe matrisi oluşturulmuştur. Ayrıca, 31 Mart 2023 gününe ait 29

gerçek teslimat verisi kullanılarak, günlük operasyonel senaryolara dayalı test ortamı sağlanmıştır. Bu çok boyutlu veri alt yapısı, hem modelin öğrenme kapasitesini artırmış hem de sahaya uygun çözümler üretmesini mümkün kılmıştır.

Özellikle şehir içi dağıtım süreçlerinin etkinliğini artırmak amacıyla, veri tabanlı yaklaşım kapsamında oluşturulan zaman-mesafe matrisi kritik rol oynamaktadır. Çalışma kapsamında kullanılan gerçek GPS verileri, sadece rota mesafelerinin değil, aynı zamanda teslimat sürelerinin, depo geri dönüş zamanlarının ve trafik kaynaklı sapmaların da modele entegre edilmesini sağlamıştır. 31 Mart 2023 tarihli günlük operasyonel test seti, yoğun dağıtım döngüleri ve çoklu sefer dinamiklerini içermekte ve bu doğrultuda geliştirilen modellerin zaman duyarlı planlama kapasitesini sahada test etmeye olanak tanımaktadır. Bu yönüyle veri altyapısı, klasik sistemlerin çoğunlukla göz ardı ettiği şehir içi dağıtım operasyonlarının dinamiklerini doğrudan göz önünde bulundurmuş ve çoklu depo-temelli dağıtım ağlarının gerçek zamanlı planlamasına katkı sunmuştur.

GPS Verisinin Önemi ve AI Tabanlı Modellerle Entegrasyonu

GPS verilerinin kullanımı, klasik OSRM tabanlı mesafe tahminlerine göre daha dinamik ve gerçek zamanlı yol süresi bilgisinin elde edilmesini sağlamaktadır. Bu tezin özünde, GPS verileri hem rota tahmininde hem de depo dönüş sürelerinin belirlenmesinde aktif olarak kullanılmıştır. Ayrıca, bu veriler üzerinden öğrenen modeller sayesinde, geçmiş teslimat desenlerinden çıkarımlar yapılarak yeni siparişler için daha doğrusal tahminler üretilmiştir. Tensör, gömme, pozisyonel kodlama ve dikkat katmanları kullanılarak bu veriler AI mimarilerine uygun hale getirilmiştir.

Bu genel AI entegrasyonuna ek olarak, GPS verilerinin özellikle şehir içi dağıtım operasyonlarında sunduğu dinamik analiz yeteneği, zaman ve lokasyon temelli karar süreçlerini doğrudan etkileyen ileri düzey uygulamaların geliştirilmesine olanak tanımıştır. Ağırlıklı olarak, trafik yoğunluğu, teslimat sırası ve araçların tekrar kullanım sıklığı gibi değişkenlerin dinamik olarak modellenebilmesi için GPS verileri kritik bir rol oynamaktadır. Bu çalışma kapsamında, araçların her bir teslimattan sonra depoya dönüş süreleri GPS verileri aracılığıyla modellenmiştir. Böylece çoklu sefer planlaması gibi kentsel dağıtım senaryolarına uygun çözümler geliştirilmiştir. Ayrıca, teslimat noktalarının zaman bazlı dağılımı ve yoğunluk bölgeleri analiz edilerek, şehir içi sıklığının yoğun olduğu alanlarda rota verimliliğini artırmayı hedefleyen AI destekli tahmin modelleri tasarlanmıştır. GPS temelli veri akışı sayesinde, klasik tahmin yöntemlerinin ötesine geçilerek gerçek zamanlı güncellenebilen ve zaman duyarlı dağıtım süreçlerini doğrudan etkileyen öğrenen sistemler oluşturulmuştur.

Çalışmanın Katkısı ve Uygulama Potansiyeli

Geliştirilen her bir model, 2013–2023 yılları arasında elde edilen büyük ölçekli kurumsal GPS ve iş verisiyle test edilerek, saha koşullarına uygunluk bakımından gerçekçi çözümler üretmiştir.

Bu yönüyle tez hem teorik hem de pratik düzeyde literatüre katkı sunmaktadır. Önerilen modeller; Transformer tabanlı dikkat mekanizmaları, Derin Pekiştirmeli Öğrenme (DRL), MAML, NEAT ve yüksek seviyeli donanım sentezi mimarilerini birleştirerek, akademik literatürde nadiren bütüncül olarak ele alınan veri-temelli ve donanım uyumlu optimizasyon stratejileri sunmaktadır. Bu modellerin hem Python tabanlı yazılım prototipleri hem de Vivado HLS gibi Saha Programlanabilir Kapı Dizisi (FPGA) donanımlarına entegrasyon potansiyelleri değerlendirilmiştir. Böylece, önerilen yaklaşımlar yalnızca akademik simülasyonlarda değil, aynı zamanda endüstriyel üretim ortamlarında da doğrudan uygulanabilir hâle gelmiştir.

Ek olarak, çalışma kapsamında geliştirilen modellerin, şehir içi dinamikler, trafik yoğunluğu ve teslimat zamanlamaları gibi kentsel dağıtım sistemlerine özgü değişkenleri GPS verisi üzerinden öğrenebilme yeteneği, dağıtım sistemlerinin çevikliği ve müşteri memnuniyetine doğrudan katkı sağlayabilecek niteliktedir. Ayrıca, günlük operasyonel verilerle yapılan testler ve grafik analizler, modelin gerçek zamanlı sistemlerle entegrasyon kabiliyetini ortaya koymakta ve gelecekte akıllı lojistik sistemlerinin temel yapı taşlarını oluşturabilecek altyapıyı sunmaktadır. Donanım düzeyinde doğrudan bir entegrasyon gerçekleştirilmemiş olmakla birlikte, geliştirilen algoritmalar donanım uyumlu biçimde tasarlanmış ve gelecekte FPGA gibi gömülü sistemlere entegre edilebilme potansiyeli taşımaktadır. Sonuç olarak, bu tez çalışması, çok depolu dağıtım sistemleri için AI tabanlı rota optimizasyonunun yazılım ve donanım düzeyinde entegre biçimde ele alınmasına yönelik anlamlı bir katkı sunmakta ve bu alandaki uygulamalı araştırmalara yeni bir perspektif kazandırmaktadır.

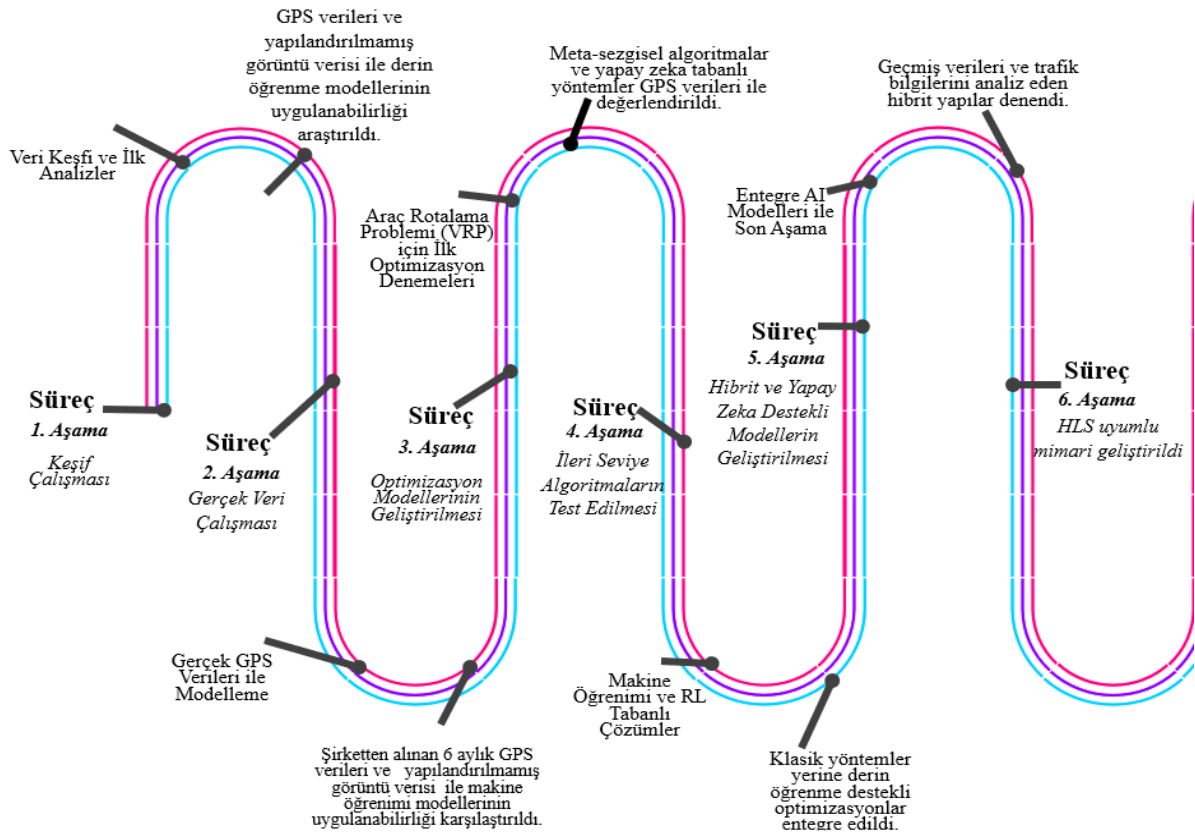
Kademeli Geliştirme Süreci

Çalışmanın devamında izlenen yöntemsel yol haritası, altı aşamalı bir yapı çerçevesinde planlanmıştır. Bu yapının ilk iki aşaması, GPS verilerinin yapay zekâ tabanlı modellere entegrasyonu ile veri formatı dönüşümlerine odaklanan keşif temelli analizleri içermektedir. Şekil 1’de bu sürecin zamansal ve mantıksal akışı sunulurken, Şekil 2’de her aşamada kullanılan yöntemler, veri türleri, tercih nedenleri ve karşılaşılan zorluklar detaylı şekilde açıklanmıştır.

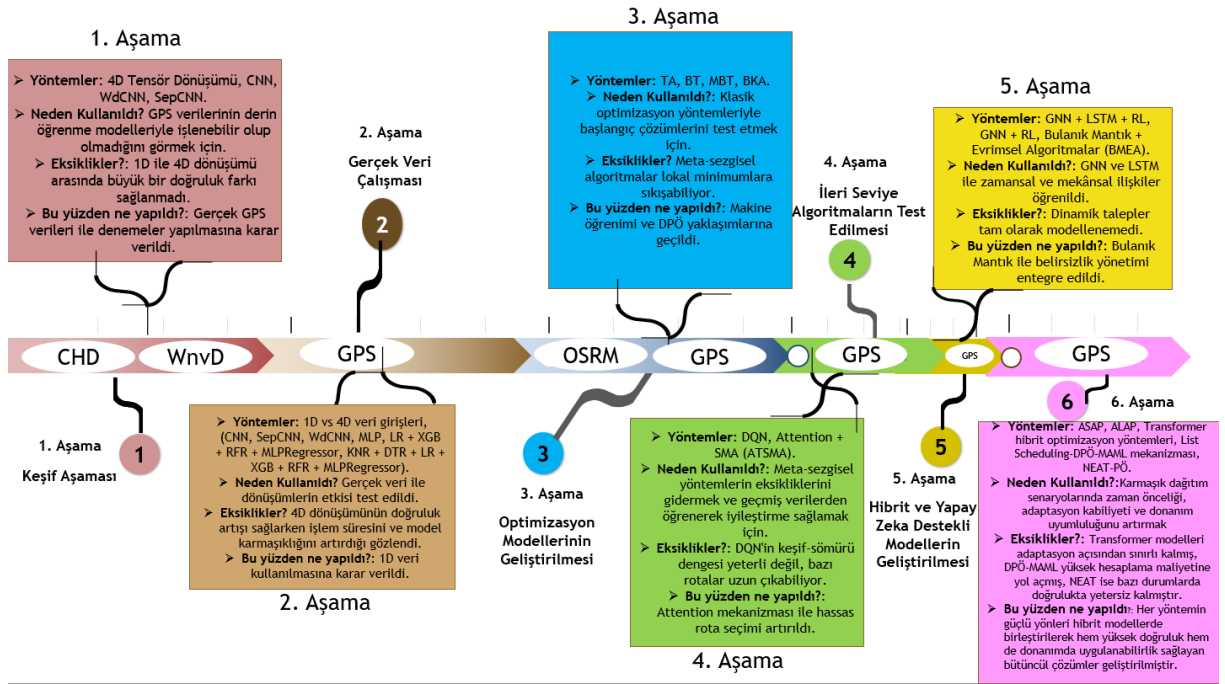
- ✓ 1. ve 2. Aşamalar: Veri keşfi ve gerçek veri çalışmaları kapsamında, 4B tensör dönüşümleri ve Evrişimsel Sinir Ağı (CNN) tabanlı yaklaşımlar kullanılarak GPS verilerinin modellemeye uygunluğu test edilmiş; 1B/4B veri karşılaştırmaları ile en uygun yapılandırma belirlenmiştir (Aydın & Erdem, 2023; Erdem *et al.* under review e).
- ✓ 3. Aşama: İlk optimizasyon denemeleri, klasik meta-sezgisel ve önerilen MBT modeli (Erdem *et al.* under review b) ile gerçekleştirilmiştir. Bu aşamada yapılan karşılaştırmalı analizlerde, MBT modelinin, özellikle büyük ölçekli ve karmaşık dağıtım problemleri üzerinde, BT gibi klasik yöntemlere kıyasla daha kısa rotalar ve daha düşük toplam maliyet ile daha yüksek performans sergilediği ortaya konmuştur. Böylece, MBT'nin önerilen çözümler arasında daha etkili bir alternatif olduğu deneysel olarak doğrulanmıştır.
- ✓ 4. Aşama: Bu aşamada, Derin Q-Ağı (DQN), dikkat mekanizması ve DTBKA gibi ileri seviye optimizasyon yaklaşımları test edilmiştir (Erdem *et al.* under review d). Modeller, eğitim ve test döngüleriyle ödül fonksiyonu, rota verimliliği ve toplam maliyet performansı açısından değerlendirilmiştir. Yapılan analizler sonucunda, özellikle dikkat destekli modellerin, rotalama süreçlerinde daha duyarlı, hassas ve tutarlı çözümler üretebildiği gözlemlenmiştir. DQN algoritması ise çevresel değişkenleri dikkate alarak dinamik ortamlarda daha etkili yönlendirme stratejileri geliştirmiştir.
- ✓ 5. Aşama: Bu aşamada, GNN-RL, GNN-Uzun Kısa Süreli Bellek (LSTM) (Hochreiter & Schmidhuber, 1997)-RL ve Bulanık Mantık-Evrimsel (BME) gibi hibrit optimizasyon yöntemleri kullanılarak ÇDARP çözümleri test edilmiştir. GNN-LSTM-RL hibrit optimizasyon modeli, grafik yapılarla konumsal ilişkileri öğrenme, zaman serileriyle talepleri modelleme ve RL ile rotaları optimize etme süreçlerini birleştirerek dinamik ortamlarda güçlü sonuçlar üretmiştir. GNN-RL hibrit optimizasyon yaklaşımı, yüksek genelleme kabiliyeti ve doğruluk seviyesi ile statik senaryolarda dengeli bir performans sunmuştur. Öte yandan BME hibrit optimizasyon yöntemi, geniş çözüm alanlarında etkin tarama ve belirsiz verilerle esnek kararlar üretme becerisi sayesinde, eksik ya da değişken bilgilere sahip senaryolarda etkili çözümler sağlamıştır. Genel değerlendirmede, GNN-RL hibrit optimizasyon yöntemi, doğruluk ve ödül skorları açısından en başarılı sonuçları sunmuştur (Erdem *et al.* under review d).
- ✓ 6. Aşama: Bu son aşamada, daha önce geliştirilen AI tabanlı modeller (örneğin; En Erken Zamanlama (ASAP) Stratejisi-Transformer hibrit optimizasyon yöntemi

(AST-Former), En Geç Zamanlama (ALAP) Stratejisi-Transformer hibrit optimizasyon yöntemi (ALT-Former), LST-DRL-MAML mekanizması (LST-Model-Agnostic Meta-Pekiştirmeli Öğrenme (MetaDRL)), NEAT-RL (NRL)), HLS ilkeleriyle uyumlu şekilde yeniden yapılandırılmıştır. Bu sayede, özellikle gerçek zamanlı ve düşük gecikmeli lojistik sistemlerde kullanılabilecek modüler ve donanıma entegre edilebilir çözümler hedeflenmiştir. Tasarlanan sistem, çoklu depo, bölünmüş teslimat ve çoklu tur gibi karmaşık dağıtım senaryolarına uygun şekilde ölçeklenebilir yapıda tasarlanmıştır. HLS uyumlu bu mimari, rota tahmini, müşteri önceliklendirme ve zaman-mesafe hesaplamaları gibi işlemlerin gelecekte gömülü sistemler üzerinde hızlandırılmış olarak çalıştırılabilmesine olanak sunmaktadır. Böylece, yoğun şehir içi dağıtım operasyonlarında gerçek zamanlı yönlendirme ve karar desteği sağlanarak operasyonel etkinlik ve yanıt süresi önemli ölçüde artırılmıştır (Erdem *et al.* under review c).

Şekil 1 bu genel süreci akış temelli olarak temsil ederken, Şekil 2 her aşamanın içeriğini, kullanılan yöntemleri ve karar gerekçelerini sistematik olarak ortaya koymaktadır. Böylece çalışma, sadece teorik değil, uygulamaya dönük bir çözüm geliştirme yaklaşımı benimseyerek, gerçek zamanlı lojistik uygulamaları için güçlü bir AI temelli optimizasyon çerçevesi sunmuştur.



Şekil 1. AI destekli ÇDARP çözüm sürecinin aşamalı yol haritası



Şekil 2. Altı aşamalı modelleme sürecine ait yöntemsel açıklama ve karar noktaları

İlk Aşama: keşif çalışması

Tezin ilk iki aşamasında yapılan çalışmaların ilerleyişi, GPS konumlandırma doğruluğunu artırmaya yönelik sistematik yaklaşım içermektedir (Aydın & Erdem, 2023). Bu süreç, anonim verilerle yapılan başlangıç çalışmalarından başlayarak gerçek dünya verileri ile gerçekleştirilen uygulama tabanlı değerlendirmelere kadar uzanmıştır. Yapılan deneylerin, tezin ilerleyen aşamalarına yönelik bir ön hazırlık niteliği taşıdığı ve genel tez hedeflerine önemli katkılar sağladığı açıklanmıştır.

Önerilen sistematik yaklaşımla, yapılandırılmamış görüntü verisinin başarı oranını değerlendirmek amacıyla iki ayrı çalışma gerçekleştirilmiştir. Bu çalışmalar, dönüşümün modeller üzerindeki etkisini anlamak ve en uygun yaklaşımı belirlemek için tasarlanmıştır.

Yapılandırılmamış görüntü verisi

Yapılandırılmamış görüntü verisi, GPS verilerinin tensör yapısına dönüştürülerek, derin öğrenme (DL) modellerinin mekânsal ilişkileri daha iyi öğrenmesini sağlamayı hedefleyen bir işlem olarak tanımlanabilir (Aydın & Erdem, 2023). Özellikle DL modelleri (CNN, Geniş ve Derin Evrişimli Sinir Ağı (WdCNN), Ayrılabilir Evrişimli Sinir Ağı (SepCNN) vb.), mekânsal desenleri ve ilişkileri algılama konusunda oldukça başarılıdır. Nitekim CNN ve Yinelenen Sinir Ağı (RNN) gibi mimariler, mekânsal yakınlık ve zamansal korelasyonları etkili biçimde öğrenerek yüksek doğruluk oranlarına ulaşabilmektedir (Arun, 2021). GPS verileri gibi 1B yapılar, doğrudan mekânsal bir bağlam sunmadığı için, bu verilerin 4B yapılandırılmamış görüntü formatına dönüştürülmesi, DL modellerinin potansiyelini daha iyi kullanmasını sağlar.

Bu yaklaşıma örnek olarak, Ribeiro *et al.* (2024), GPS iz verilerini görüntü temsiline dönüştürerek ulaşım modlarını tanıyan bir DL sistemi geliştirmiş ve gerçek dünya senaryolarında yüksek doğruluk oranı elde etmişlerdir. Bu durum, yapılandırılmamış görüntü dönüşümünün özellikle karmaşık ve düzensiz GPS verilerinde yüksek performans sağlayabileceğini göstermektedir. Benzer şekilde, Noureldin *et al.* (2003), GPS ve Ataletsel Seyrüsefer Sistemi (INS) verilerini birleştirerek konum doğruluğunu artırmak amacıyla, yapay sinir ağı (ANN) ile Wavelet çoklu çözünürlük analizini (WMRA) entegre ettikleri bir model önermiştir. Bu model sayesinde, GPS verilerindeki konum hatalarının sinir ağı üzerinden öğrenilerek düzeltildiği ve konumlama performansının anlamlı ölçüde iyileştirildiği raporlanmıştır. Bu dönüşüm, yalnızca modelin öğrenme kabiliyetini artırmakla kalmayıp, GPS verilerinin görsel olarak temsil edilmesine de olanak tanımaktadır. Tensör yapısı, ham GPS verilerinin daha iyi anlaşılması ve işlenmesi için görselleştirilebilir bir format sunar. Böylece, GPS verilerinin bir nevi görselleştirilmiş hali elde edilir. Örneğin, bir görüntüdeki piksellerin birbirine yakınlığı mekânsal ilişkileri temsil ederken, GPS verilerinde enlem ve boylam değerlerinin konumlar arasındaki ilişkileri temsil ettiği düşünülür. Bu nedenle, 4B veri dönüşümü, GPS verilerini adeta bir görüntü gibi işlemeyi mümkün kılar. Önerilen modellerin, konumlandırma veya rota planlama gibi karmaşık görevlerde daha hassas tahminler yapması hedeflenir.

Keşif çalışması, GPS konumlandırma doğruluğunu artırmak amacıyla 4B yapılandırılmamış görüntü verisi dönüşümünün DL modellerinde uygulanabilirliğini araştırmak için bir ilk adım olarak tasarlanmıştır. Bu aşamada, şirketten gerçek GPS verileri henüz temin edilemediği için anonim iki farklı veri seti (California Konut Veri Seti (CHD) (Anonymous 2018) ve Batı Nil Virüsü Veri Seti (WnvD) (Anonymous 2018) kullanılmıştır. Bu aşamada amaç, 1B GPS verilerinin 4B tensör temsillerine dönüştürülmesinin, CNN tabanlı ve hibrit modeller dahil olmak üzere DL modellerinin (Xception, VGG16, VGG19, Alexnet ve CNN tabanlı 7 farklı model) performansını artırıp artırmayacağını değerlendirmektir. Elde edilen temel bulgulara, 4B dönüşümlerin DL modellerinin mekânsal desenleri etkili bir şekilde yakalamasını sağladığını göstermiştir. Ancak, bu dönüşüm, ML tabanlı modellerle (kNN-Doğrusal Regresyon (LR) hibrit optimizasyon yöntemi, kNN-Karar Ağacı Regresyonu (DTR) hibrit optimizasyon yöntemi ve Yığın Tabanlı Topluluk Modeli) karşılaştırıldığında tahmin doğruluğunda önemli bir iyileşme sağlamamıştır.

İkinci Aşama: Gerçek veri çalışması

Gerçek Veri Çalışması, şirketten alınan 6 aylık GPS verileri üzerinde deneyler gerçekleştirilmiştir. Bu aşamada amaç, gerçek GPS verileri üzerinde 1B ve 4B veri girişlerinin

performansını karşılaştırmak ve tezin sonraki aşamalarında hangi veri formatının kullanılmasının daha uygun olacağına karar vermektir. Esas bulgular incelendiğinde, 4B dönüşümlü veriler DL modellerinde (CNN, SepCNN, WdCNN, Çok Katmanlı Algılayıcı (MLP)) belirli başarılar göstermiştir, ancak 1B verilerle çalışan modellerle karşılaştırıldığında işleme süresi ve bellek kullanımında dezavantajlar gözlenmiştir. Ayrıca, DL tabanlı modellerin daha fazla parametre gereksinimi nedeniyle bazı durumlarda daha az etkili olduğu belirlenmiştir (Erdem *et al.* under review e).

İki aşamanın da tezin ilerleyen süreçlerine etkileri ele alındığında;

1. Model Performans Analizi: Farklı veri işleme yöntemlerinin DL ve ML modellerinde nasıl performans gösterdiğini anlamak için bir temel oluşturuldu.
2. GPS Teknolojisinin Önemi ve Önerilen Yaklaşımlar: GPS teknolojisinin çeşitli uygulamalardaki önemine değinilmiştir. Çalışma, GPS verilerindeki konum hatalarını minimize etmek için önerilen geleneksel yöntemlerin sınırlamalarını açıklamış ve düşük maliyetli AI tabanlı yeni yaklaşımlar sunmuştur. Ayrıca, GPS verilerindeki doğruluğun artırılmasında DL ve hibrit algoritmaların potansiyeline dikkat çekilmiştir. Önerilen modeller farklı veri setleri üzerinde test edilmiştir.
3. Veri Formatı Seçimi: Anonim veri setleri üzerinde yapılan testlerde, 1B'den 4B'ye dönüştürülen verilerin bazı DL modellerinde kısmen başarılı olduğu görülmüştür. Ancak bu dönüşüm her zaman üstün bir performans sağlamamıştır. Gerçek GPS verileri üzerinde yapılan testlerde, 4B dönüşümünün işleme süresini ve model karmaşıklığını artırdığı, ancak doğrulukta belirgin bir fark yaratmadığı gözlemlenmiştir. Sonuç olarak, yapılandırılmamış görüntü verisi dönüşümü, mekânsal ilişkilerin öğrenilmesi için fayda sağlayabilir; ancak bu tür dönüşümlerin etkisi, veri setinin yapısına ve kullanılan modele göre değişkenlik gösterebilmektedir. Bu durum, çalışmalarda elde edilen sonuçlarla da desteklenmiştir. 4B dönüşümlerin avantajlarının sınırlı olduğu göz önüne alındığında, tezin ilerleyen aşamalarında dönüşüm yapılmadan 1B verilerin kullanılması tercih edilmiştir. Bu, daha düşük işlem maliyetleri ve daha kısa sürelerde sonuç alınmasına olanak tanımıştır.

Üçüncü Aşama: Optimizasyon modellerinin geliştirilmesi ve gerçek zamanlı gps verileriyle entegrasyonu

Bu evre, önceki aşamalarda oluşturulan veri işleme ve modelleme temellerini kullanarak, özellikle optimizasyon algoritmalarının geliştirilmesi ve gerçek dünya verileri ile

test edilmesine odaklanan çalışmayı temsil eder. Önceki iki kademedeki yapılan ön keşif ve gerçek veri analizlerinin çıktılarından yararlanarak, daha sofistike ve uygulanabilir çözümler geliştirme sürecini ifade etmektedir. Modern dağıtım sistemlerinin karmaşıklığı ve işletmelerin ARP çözmek için karşılaştığı zorluklar ele alınmıştır. Geleneksel yöntemlerin eksiklikleri vurgulanmış ve büyük ölçekli gerçek GPS verileri ile şirket kaynaklı verilerin entegrasyonunun potansiyel faydaları açıklanmıştır. GPS verileriyle çözülmesi gereken lojistik ve araç rotalama problemlerindeki zorluklar ele alınmış ve mevcut yöntemlerin sınırlamaları vurgulanmıştır. Bu aşamada ayrıca, şirketin İstanbul'daki içecek dağıtım ağı ve kullanılan veri kaynakları tanıtılmıştır. Çalışmanın temel amacı, ÇDARP daha verimli çözmek için heterojen filo kullanımı ve AI temelli yöntemleri uygulayıp, operasyonel maliyetleri azaltmak ve müşteri memnuniyetini artırmaktır.

Dağıtım sistemlerinde ARP

İyi yapılandırılmış bir dağıtım planına sahip bir şirket, operasyonel verimliliğini artırmak ve maliyet avantajı sağlamak için yenilikçi stratejiler geliştirebilir (Castillo *et al.* 2024; Chen *et al.* 2024). Ancak, kötü bir dağıtım planı, teslimat maliyetlerinde önemli artışlara ve müşteri memnuniyetsizliğine yol açabilir (Chen *et al.* 2023; Maleki *et al.* 2024). Modern dağıtım sistemlerinin karmaşıklığı; küreselleşme, teknolojik gelişmeler, artan rekabet ve müşteri taleplerine zamanında çözüm sağlama gerekliliği gibi faktörlerden kaynaklanmaktadır (Bai *et al.* 2023; Chen *et al.* 2024). Bu bağlamda, dağıtım sistemlerinin verimli bir şekilde yönetimi şirketlere stratejik avantaj sağlar (Shahbazian *et al.* 2024; Feng *et al.* 2020). ARP, modern dağıtım sistemlerinde karşılaşılan temel zorluklardan biridir (Asghari & Al-e, 2021; Bai *et al.* 2023). Bu problem, araç filolarının kapasite sınırlamalarını ve müşteri ihtiyaçlarını dikkate alarak etkin rotaları belirlemeyi hedefler (Du *et al.* 2017; Khodashenas *et al.* 2023). Bu hedef, daha az yakıt tüketimi, daha kısa mesafeler ve zaman tasarrufu gibi iyileştirmeler sağlar (Castillo *et al.* 2024; Chen *et al.* 2024). Neticede hem çevresel hem de ekonomik faydalar sunulabilir (Chen *et al.* 2024; Li *et al.* 2019). Örneğin, yakıt tüketiminin azaltılmasıyla hava kirliliği ve zararlı gaz emisyonları düşürülerek insan sağlığı ve çevre korunabilir (Chen *et al.* 2024; Asghari & Al-e, 2021). Şirketler genellikle rotalarını sevkiyat sürelerini tahmin ederek planlamaktadır (Chen *et al.*, 2023; Hao *et al.* 2024). Ancak yanlış tahminler, teslimatların gecikmesine, müşteri memnuniyetsizliğine ve finansal kayıplara yol açabilir (Shahbazian *et al.* 2024). ARP'nin karmaşıklığı nedeniyle bu problem genellikle sezgisel yöntemlerle çözülmektedir (Azevedo *et al.* 2024). Etkili bir planlama stratejisi, teslimat maliyetlerini kontrol altında tutmak ve piyasa dalgalanmalarına uyum sağlamak açısından kritik öneme sahiptir (Shahbazian *et al.* 2024; Feng *et al.* 2020).

Motivasyon

Bu aşamanın temel motivasyonu, geleneksel ARP çözümlerinin sınırlamalarını aşarak geniş çaplı tarihsel GPS verileri ve Mikro V15 kullanarak daha doğru ve etkili bir çözüm sunmaktır. Geleneksel yöntemler genellikle gerçek zamanlı GPS verilerine dayanarak küçük ölçekli senaryolara odaklanmaktadır. Bu çalışma ise büyük veri setlerini kullanarak daha kapsamlı ve optimize edilmiş çözümler sunmayı hedeflemektedir. Rekabetçi bir ortamda operasyonel verimliliği artırmak, kaynak tahsisini optimize etmek ve müşteri memnuniyetini sağlamak bu çalışmanın öncelikli amaçları arasındadır. Ayrıca, çalışma şirketin dağıtım stratejisini optimize etmeyi, araç rotalama ve kaynak tahsisi gibi lojistik yönetimindeki yaygın sorunları ele almayı amaçlamaktadır. İş süreçleri, dokümantasyon, veri tabanları ve cari hesap yönetimi gibi unsurların araç rotalama ve dağıtım optimizasyonundaki etkileri incelenmiştir. Örneğin, lojistik süreçlerde kullanılan satın alma faturaları, satış verileri ve sevkiyat notları gibi iş belgeleri, stok hareketlerinin ve finansal işlemlerin gerçek zamanlı takibiyle entegre edilmiştir. Bu veri kaynakları, işletme süreçlerinin daha verimli ve bilgiye dayalı kararlarla yönetilmesine katkı sağlamaktadır.

Şirket ve operasyonel süreçler

Şirket, işletme süreçleri, işletme belgeleri ve veri tabanları gibi temel unsurları bünyesinde barındırmaktadır (Belbag *et al.* 2009). İşletme süreçleri; firma, müşteri, depo ve tedarikçiden oluşmaktadır. Firma, mal olarak içecek ürünler, demirbaş olarak araçlar ve hizmet olarak danışmanlık, teknik destek, lojistik, bakım ve onarım gibi çeşitli hizmetler sağlamaktadır.

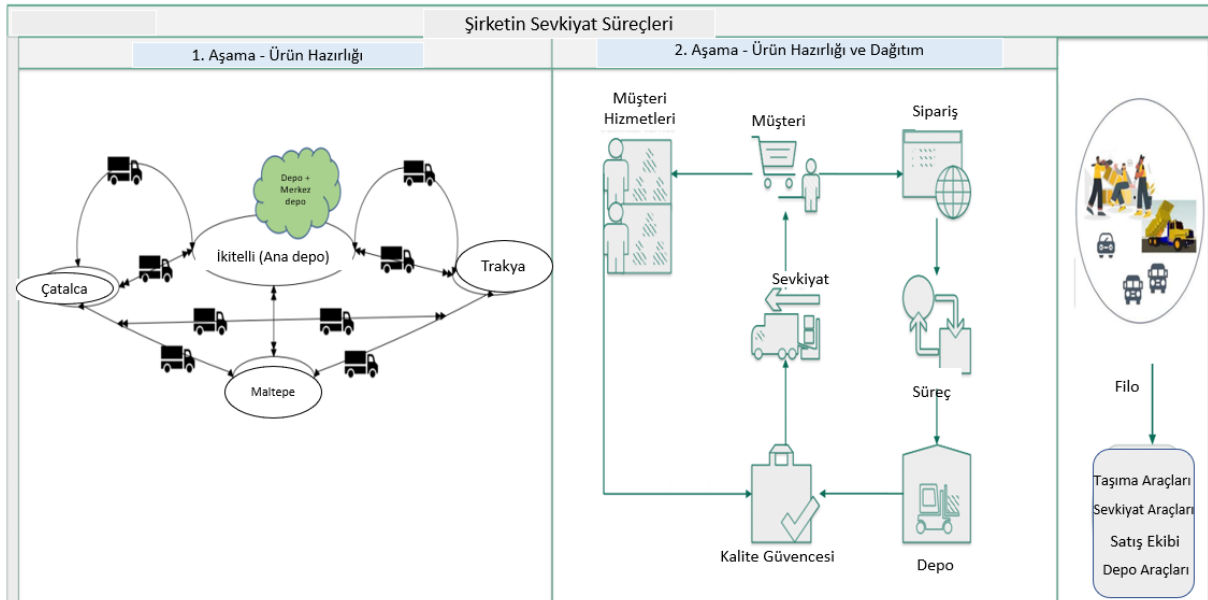
Şirketin lojistik ve finansal süreçlerini destekleyen işletme belgeleri; satın alma faturası, satış verisi ve sevk irsaliyesinden oluşmaktadır. Satın alma faturası, bir işletmenin tedarikçisinden mal veya hizmet alımını resmi olarak belgelendirirken; satış verisi, müşteri talepleri ve satışı yapılan ürünlerin detaylarını içermektedir. Sevk irsaliyesi ise, ürünlerin taşınma sürecine ilişkin detayları (miktar, nitelik, alıcı bilgisi) kayıt altına alarak lojistik süreçleri desteklemektedir (Pessi, 2017; Chu *et al.* 2014).

Veri tabanları ve izleme tabloları, işletmenin operasyonel süreçlerini etkili bir şekilde yönetmesine olanak sağlamaktadır. Bu sistemler, stok tablosu, cari hesaplar tablosu, stok hareketleri tablosu, cari hesap hareketleri tablosu, siparişler tablosu ve cari hesap adresleri tablosundan oluşmaktadır (Tatsis *et al.* 2006). Stok tablosu, ürünlerin mevcut miktarını ve stok kodlarını içerirken; cari hesaplar tablosu, şirketin finansal işlemlerini ve müşteri/tedarikçi ilişkilerini detaylandırmaktadır. Stok hareketleri tablosu, ürünlerin giriş, çıkış, transfer ve satış

gibi tüm hareketlerini kaydederek stok yönetimine destek olmaktadır. Cari hesap hareketleri tablosu, borç, alacak ve ödeme durumları gibi finansal işlemleri detaylandırır. Siparişler tablosu, müşteri taleplerini ve siparişlerin durumunu (beklemede, gönderildi, iptal edildi vb.) kayıt altına alırken, cari hesap adresleri tablosu lojistik süreçleri kolaylaştırmak için adres bilgilerini saklamaktadır (Mustafa 2011; Purba & Firdaus, 2021; Lopes *et al.* 2022).

Şirketin dağıtım süreçleri, ürün bulunabilirliği, araç planlaması ve rota oluşturma gibi faktörlere dayalı olarak optimize edilmektedir. Örneğin, tedarikçiden yapılan satın alma işlemleri; fiyat, maliyet, araç kapasitesi ve rota gibi unsurlar göz önüne alınarak planlanmaktadır. Firma, ürünleri depolara taşıma işlemlerini kendi araçlarıyla gerçekleştirebilmekte veya lojistik operasyonlarını araçların durumuna göre düzenleyebilmektedir. Araçlar gerektiğinde farklı lojistik operasyonlar için de kullanılmaktadır. Bu süreçler, maliyetleri en aza indirirken operasyonel etkinliği artırmayı hedeflemektedir (Şekil 3).

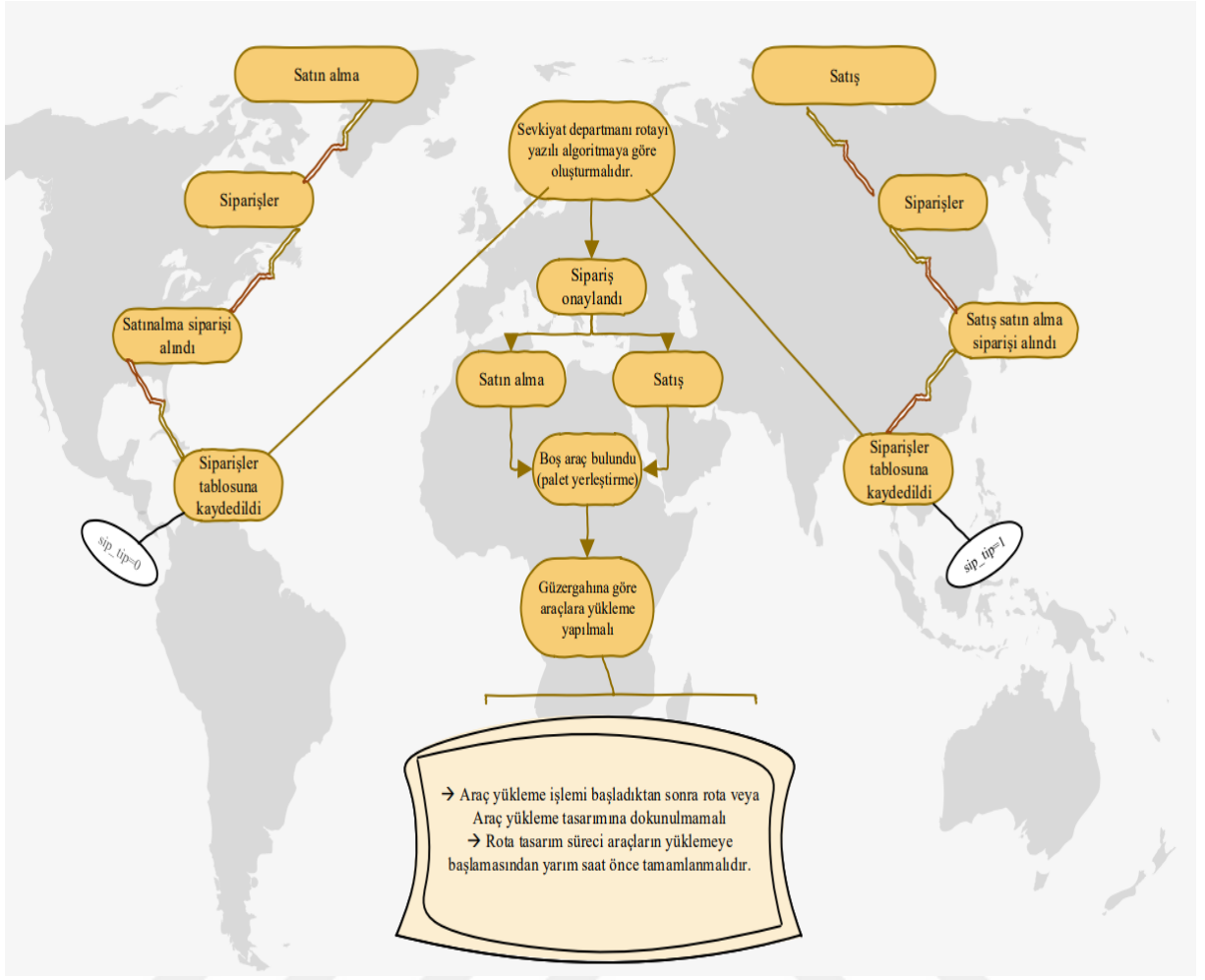
Bu çalışma, dağıtım ağlarının optimizasyonuna yönelik veri odaklı bir yaklaşım sunarak ÇDARP'yi ele almaktadır. Tarihsel GPS verileri ve Mikro V15 kullanılarak oluşturulan zaman-mesafe matrisleri, daha doğru planlama ve maliyet optimizasyonu sağlamıştır. Çalışmanın sonuçları, işletmelerin dağıtım sistemlerini daha etkin yönetmek ve müşteri memnuniyetini artırmak için veri odaklı stratejiler geliştirebileceğini göstermektedir. Bu yaklaşım, gelecekteki lojistik yönetimi çalışmalarına rehberlik edebilecek önemli bir katkı sunmaktadır.



Şekil 3. Ürün hazırlama ve dağıtım süreçlerine ilişkin bir örnek (Erdem *et al.* under review b)

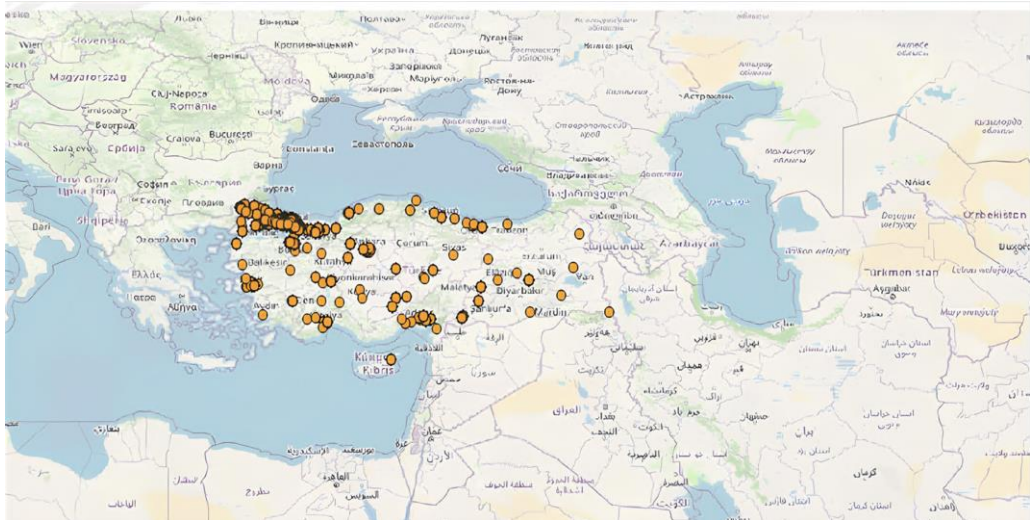
Büyük ölçekli ve gerçek zamanlı GPS verilerinin ÇDARP çözümlerine katkısı

Birçok ARP çalışması, çeşitli meta-sezgisel algoritmaların geliştirilmesine ve bu algoritmaların belirli standart veri kümeleri üzerinde test edilmesine odaklanmıştır. Ancak, bu çalışmaların büyük bir kısmı gerçek zamanlı GPS verilerini kullanmamakta ve bu nedenle trafik ve hava koşulları gibi dinamik faktörleri göz ardı etmektedir. Çalışmamız, büyük ölçekli, gerçek dünya GPS verilerini entegre ederek bu boşluğu doldurmayı amaçlamakta ve bu şekilde daha doğru ve dinamik rota planlaması sağlamaktadır. Gerçek zamanlı değişkenlikleri dikkate alan bu yaklaşım, özellikle dağıtım sektörü gibi yüksek talep içeren ortamlarda ARP çözümlerinin güvenilirliğini ve uygulanabilirliğini artırarak literatüre önemli bir katkı sunmaktadır. Ayrıca, mevcut yöntemlerde sıklıkla karşılaşılan veri kalitesi sorunlarını aşmayı hedeflemektedir. Şirketin operasyonlarında, ürünler (stoklanmış ürünler) İkitelli'deki ana merkez depodan alınarak Çatalca, Trakya ve Maltepe gibi diğer depolara dağıtılmaktadır. Depolar, hem kendi aralarında hem de doğrudan müşterilere sevkiyat gerçekleştirebilmektedir (Şekil 3). Şirketin sevkiyat operasyonlarındaki taşıma planlaması Şekil 4'de verilmiştir. ÇDARP çözümünde kullanılan verilerin kalitesi, navigasyon ve rota planlaması gibi konum tabanlı hizmetler için büyük önem taşımaktadır. Mevcut yöntemlerin neredeyse tamamı, anlık olarak değişebilen bilgilere (hava durumu, trafik koşulları vb.) dayanmaktadır. Bu durum, mesafeler arasındaki zaman değerlerinde tutarsızlıklara yol açmaktadır. Ne yazık ki, ÇDARP çözümlerinde çıkarılan verilerin kalitesi genellikle düşüktür. Bu çalışmada, İstanbul şehrinde üç yıl boyunca toplanan büyük ölçekli GPS konum verileri kullanılarak seyahat sürelerini elde etmek için etkili bir büyük veri tabanlı yaklaşım önerilmektedir. GPS verileri, navigasyon hizmetleri sağlayabilen heterojen bir platformdan elde edilmiştir. Bu verilerin kullanım amacı, araçların sevkiyat sürecinde optimum maliyet sağlanarak maksimum verimliliği artırmak ve gerçek dünyadaki etkinliğini değerlendirmektir. Araçlar, operasyonel günlük faaliyetlerine önceden belirlenmiş bir depodan başlamakta ve dış kullanıcılar tarafından sağlanan talepler veya planlar doğrultusunda belirli bir coğrafi bölgede çeşitli operasyonları gerçekleştirmektedir. Bu süreçte araçlar, GPS cihazları aracılığıyla veri toplayabilmektedir. Bu tür bir altyapıda, araçlara yerleştirilen cihazlardan alınan veriler Mobiliz merkezinde hiyerarşik bir şekilde düzenlenmektedir. Ağ izleme işlemleri ise, belirli bir coğrafi alanda hareket eden araçlar tarafından sensör ekipmanları aracılığıyla gerçekleştirilmekte ve trafik, kesintiler, yoğunluk ve sapsmalar gibi ağ durumunu sürekli ve gerçek zamanlı olarak izlemek için veri toplanmaktadır. Çalışmada, GPS verilerinden çıkarılan zaman bilgilerinin, klasik ÇDARP çözümünde kullanılan veri kaynağı (OSRM API) ile elde edilen zaman değerlerine göre daha iyi ve daha güvenilir sonuçlar verdiği vurgulanmıştır.



Şekil 4. Şirket sevk işlemi ayrıntıları (Erdem *et al.* under review b)

2020-2023 yılları arasında sevkiyat yapılan müşteriler, Kuantum Coğrafi Bilgi Sistemi (QGIS) kullanılarak harita üzerinde görselleştirilmiştir. Bu süre boyunca sipariş veren müşterilerin coğrafi dağılımı incelenmiştir (Şekil 5).



Şekil 5. 2013-2023 Sevkiyat haritası QGIS gösteriminde: Tüm müşteri düğümlerinin görselleştirilmesi (Erdem *et al.* under review b)

Tablo 1. Başlangıç Çözüm Kümesinin Optimizasyon Sürecine Etkileri

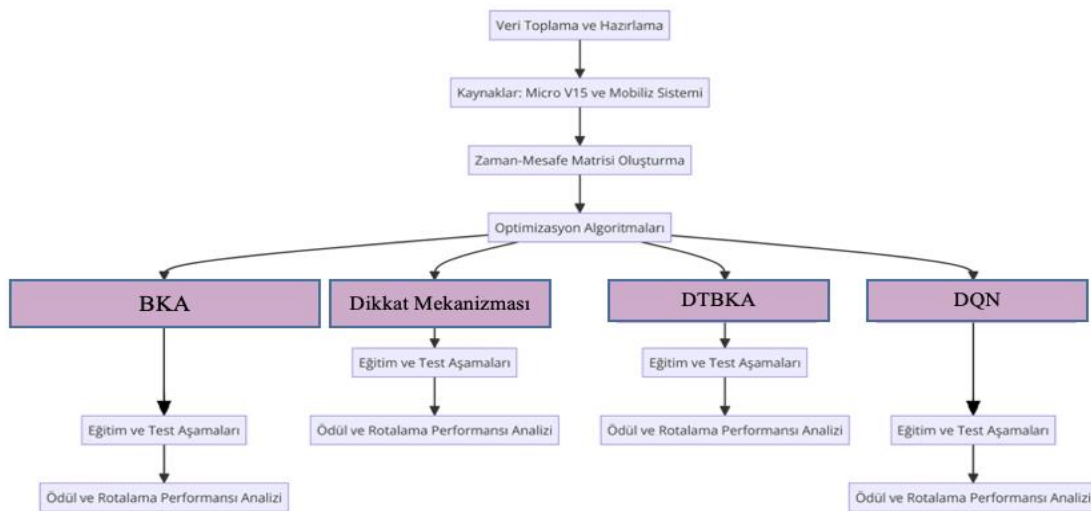
Daha İyi Yakınsama	Hızlı Yakınsama: İyi bir başlangıç çözüm kümesi, optimizasyon algoritmasının daha hızlı yakınsamasını sağlayabilir. Bu, modelin daha kısa sürede optimal veya yakın-optimal çözüme ulaşması anlamına gelir (Zhang <i>et al.</i> 2022; Feng <i>et al.</i> 2020). Yerel Minimumdan Kaçınma: İyi bir başlangıç noktası, optimizasyon sürecinin yerel minimumlara takılma olasılığını azaltabilir ve global minimuma daha yakın bir çözüm bulma şansını artırabilir (Hinton <i>et al.</i> 2006; Bai <i>et al.</i> 2023).
Model Performansı	Daha Yüksek Doğruluk: İyi bir başlangıç çözüm kümesi, modelin daha yüksek doğruluk oranlarına ulaşmasını sağlayabilir. Bu, özellikle karmaşık modellerde ve büyük veri setlerinde önemlidir (Zhang <i>et al.</i> 2023; Wang <i>et al.</i> 2023). Dengeli Öğrenme: Başlangıçta iyi ayarlanmış parametreler, modelin eğitim sırasında daha dengeli bir şekilde öğrenmesini sağlayabilir, bu da genelleme performansını artırabilir (Sahar <i>et al.</i> 2021; Bai <i>et al.</i> 2023).
Hesaplama Kaynaklarının Verimli Kullanımı	Kaynak Tasarrufu: İyi bir başlangıç çözüm kümesi, hesaplama kaynaklarının daha verimli kullanılmasını sağlar. Bu, özellikle büyük ölçekli modellerde ve sınırlı kaynaklara sahip ortamlarda önemlidir (Chen <i>et al.</i> 2023; Wang <i>et al.</i> 2023). Daha Az İterasyon: Optimizasyon sürecinin daha az iterasyonla tamamlanması, hesaplama yükünü azaltır ve kaynakların daha etkin kullanılmasını sağlar (Bai <i>et al.</i> 2023; Stodola, 2020).
Model Kararlılığı	Kararlı Öğrenme: İyi bir başlangıç çözüm kümesi, modelin eğitim sırasında daha kararlı olmasını sağlayabilir. Bu, özellikle DL modellerinde önemlidir, çünkü bu modeller genellikle eğitim sırasında kararsızlıklar yaşayabilir (Chaabani & Said, 2019; Mak <i>et al.</i> 2023). Daha Az Aşırı Uyum: İyi bir başlangıç noktası, modelin aşırı uyum sorununu azaltabilir ve genelleme yeteneğini artırabilir (Bai <i>et al.</i> 2023; Shahbazian <i>et al.</i> 2024).
Hiperparametre Optimizasyonu	Daha Etkili Hiperparametre Ayarı: Başlangıç çözüm kümesi, hiperparametre optimizasyonu sürecini daha etkili hale getirebilir. Bu, modelin daha iyi performans göstermesini sağlayabilir (Zhou <i>et al.</i> 2024; Hutter <i>et al.</i> 2019). Daha Az Hiperparametre Denemesi: İyi bir başlangıç noktası, hiperparametre optimizasyonu için daha az deneme yapılmasını gerektirebilir, bu da süreci hızlandırır (Yang 2010; Karakatić <i>et al.</i> 2021).
Transfer Öğrenmesi ve İnce Ayar	Transfer Öğrenmesi: Başlangıç çözüm kümesi, transfer öğrenmesi sürecini daha etkili hale getirebilir. Önceden eğitilmiş modellerin ağırlıkları, yeni bir görev için iyi bir başlangıç noktası olabilir (Chaabani & Said, 2019 ; Feng <i>et al.</i> 2020). İnce-Ayar: İyi bir başlangıç noktası, ince ayar sürecini daha verimli hale getirir ve modelin yeni verilere daha hızlı uyum sağlamasını sağlar (Zhang <i>et al.</i> 2022; Pan & Liu; 2023).
Algoritma Seçimi ve Uyarlanabilirlik	Algoritma Seçimi: İyi bir başlangıç çözüm kümesi, farklı optimizasyon algoritmalarının daha etkili bir şekilde çalışmasını sağlayabilir. Bu, algoritma seçimi sürecini kolaylaştırır (Gu <i>et al.</i> 2022 ; Bai <i>et al.</i> 2023). Uyarlanabilirlik: Başlangıç çözüm kümesi, modelin farklı veri setlerine ve problemlere daha iyi uyum sağlamasını kolaylaştırabilir (Ayyıldız <i>et al.</i> 2022; Wang <i>et al.</i> 2022).
Güvenilirlik ve Tekrarlanabilirlik	Güvenilir Sonuçlar: İyi bir başlangıç noktası, modelin eğitim sürecinde daha güvenilir sonuçlar üretmesini sağlar. Bu, özellikle endüstriyel uygulamalarda önemlidir (Chen <i>et al.</i> 2023). Tekrarlanabilirlik: Başlangıç çözüm kümesi, modelin farklı ortamlarda ve farklı zamanlarda tekrarlanabilir sonuçlar üretmesini sağlayabilir (Shahbazian <i>et al.</i> 2024; Zhou <i>et al.</i> 2024).

Bu çok aşamalı optimizasyon sürecinin sonunda, özellikle kentsel alanlardaki şehir içi dağıtım süreçlerinin etkisi dikkate alınarak daha esnek ve ölçeklenebilir rota planları oluşturulmuştur. Önerilen modelde şehir içi trafik ve teslimat yoğunluğunun etkili olduğu kritik bileşenler vurgulanmıştır. Öncelikle, trafik yoğunluğu etkisi ve rota tahmini, GPS verilerinin zaman-mesafe matrisine dahil edilmesiyle daha gerçekçi biçimde modellenmiştir. Bu sayede, dinamik seyahat süresi farkındalığı sağlayan bir araç yönlendirme sistemi geliştirilmiştir. GPS tabanlı seyahat süresi kısıtları sayesinde, rotalar yoğun trafik koşullarına göre yeniden

şekillendirilmiş ve şehir içi teslimat sıralaması daha hassas hale getirilmiştir. Ayrıca, geri dönüş süresinin dikkate alınması ile çoklu seferli araç yönlendirme yapılmış, bu sayede teslimat döngüsünün süre ve kaynak kullanımı açısından optimizasyonu sağlanmıştır. Sonuç olarak, gerçek seyahat sürelerine dayalı araç yönlendirme stratejisi, özellikle trafik yoğunluğu yüksek bölgelerde teslimat başarımını ve zamanında ulaştırma oranını artırmıştır. Bu çalışma, kentsel lojistik operasyonlarının başarı üzerindeki belirleyici rolünü ortaya koymakla kalmamış, aynı zamanda AI destekli veri kullanımıyla trafik odaklı dağıtım sistemlerine entegre edilen dinamik karar destek çözümlerinin uygulanabilirliğini de kanıtlamıştır (Erdem *et al.* under review b).

Dördüncü Aşama: İleri seviye algoritmaların test edilmesi

Son yıllarda, meta-sezgisel, RL ve DL yöntemleri, ÇDARP çözümünde önemli bir ilgi görmüştür. Meta-sezgisel yöntemler çözüm alanını verimli bir şekilde tarayarak çeşitli iyileştirmeler sunarken, RL algoritmaları ödül-ceza mekanizması kullanarak rotaları optimize eder. DL yaklaşımları ise karmaşık veri yapılarını öğrenerek daha sağlam ve etkili çözümler sunar. Bu yöntemlerin ÇDARP'ye uygulanması, klasik algoritmalara kıyasla daha iyi sonuçlar elde edilmesini sağlamaktadır. Bu amaçla, Balçık Küfü Algoritması (BKA) (Li *et al.* 2020), DQN (Mnih *et al.* 2015) ve yenilikçi bir hibrit model olan DTBKA'nın birleşiminden oluşan çeşitli hesaplama yaklaşımlarının karşılaştırması incelenmiştir. Üçüncü aşamada kapsamlı GPS verileri ile oluşturulan zaman-mesafe matrisi kullanılmıştır. Bu kapsamda Şekil 7'de, BKA, DQN, dikkat mekanizması ve DTBKA modellerinin zaman-mesafe matrisleri üzerinde eğitilmesi, test edilmesi ve performanslarının karşılaştırılması adımlarını içeren genel süreç şematik olarak sunulmuştur. Görselde, algoritmaların eğitim-test döngüleri sonucunda elde edilen çıktıların ödül skorları ve rota verimliliği üzerinden değerlendirilmesi gösterilmektedir (Erdem *et al.* under review d).



Şekil 7. Zaman-mesafe verisi ile güçlendirilmiş araç rotalama için optimizasyon algoritmalarının karşılaştırmalı süreci

Motivasyon

Bu aşamada çalışmanın temel motivasyonu, klasik meta-sezgisel algoritmaların keşif-sömürü dengesindeki yetersizliklerini gidererek daha hassas rota seçimleri gerçekleştirmektir. DQN'nin bazı durumlarda uzun rotalar üretmesi ve genel performans dengesizliği nedeniyle, geçmiş verilerden öğrenerek iyileştirme sağlayan RL ve dikkat mekanizmalarıyla bu eksiklikler giderilmeye çalışılmıştır. Özellikle DTBKA yaklaşımı ile, geçmiş deneyimlerden öğrenen ve daha duyarlı rota kararları alabilen bir sistem hedeflenmiştir. Bu bağlamda, ileri seviye algoritmaların entegrasyonu ile hem rota verimliliği hem de model başarımı artırılmak istenmiştir (Erdem *et al.* under review d).

Yöntemlerin ÇDARP çözümündeki önemi: Kavramsal bir bakış

BJA, biyolojik ilhamla geliştirilmiş bir meta-sezgisel optimizasyon yöntemidir. Doğadaki küflerin besin kaynaklarını verimli bir şekilde bulma davranışını taklit eden BJA, çok geniş çözüm alanlarında küresel optimuma yakın çözümler elde etmek için güçlü bir yeteneğe sahiptir. BJA'nın ÇDARP çözümünde önemi şunlardan kaynaklanır:

- ✓ Çözüm alanını kapsamlı tarama yeteneği (Altay, 2022): BJA, rastgelelik ve keşif yeteneği sayesinde yerel optima takılmadan geniş bir çözüm alanını tarar ve yüksek kaliteli çözümler elde eder (Gao *et al.* 2020).
- ✓ Karmaşık ve büyük ölçekteki problemler için uygunluk (Gao *et al.* 2020; Jia *et al.* 2022; Dokeroglu *et al.* 2024; Kaveh *et al.* 2022; Ghannadi & Kourehli, 2022): Çok sayıda müşteri ve depo içeren ÇDARP gibi büyük ölçekli optimizasyon problemlerinde verimli sonuçlar sunabilir.
- ✓ Belirsizlikleri yönetme yeteneği (Yang *et al.* 2023): BJA'nın doğası gereği, belirsiz veya dinamik koşullarda bile iyi çözümler sunma potansiyeli yüksektir (Dokeroglu *et al.* 2024).

DQN, RL ile derin sinir ağlarını birleştiren bir yöntemdir. DQN, bir ortamda bulunan bir ajanı eğiterek, deneme-yanılma yoluyla en iyi eylemleri öğrenmesini sağlar. ÇDARP çözümünde DQN'nin önemi şu nedenlere dayanmaktadır:

- ✓ Dinamik ortamları öğrenme yeteneği (Arulkumaran *et al.* 2017): ÇDARP gibi müşteri taleplerinin ve teslim sürelerinin sürekli değişebildiği problemler için DQN, çevresel değişikliklere hızlı adaptasyon sağlar (Pan & Liu, 2023; Joe & Lau, 2020; Chen *et al.* 2022).

- ✓ Yüksek kaliteli politikalar öğrenme: DQN, müşteriler ve depolar arasındaki etkileşimleri modelleyerek, minimum toplam süre veya maliyet hedefleyen rotalar oluşturur (Cai *et al.* 2024).
- ✓ Sürekli iyileştirme ve genel performans: DQN, eğitim süreci boyunca sürekli olarak performansını iyileştirir ve böylece her senaryoya uygun yüksek kaliteli politikalar sunar (Zhang *et al.* 2024).

Dikkat mekanizması, veri içindeki önemli özelliklere odaklanarak modelin daha etkili ve doğru kararlar almasını sağlar. Klasik DL yöntemlerinde tüm girdilere eşit ağırlık verilirken, dikkat mekanizması kritik öneme sahip bilgilere daha fazla ağırlık atayarak, veri yoğunluğu yüksek problemlerde verimliliği artırır. ÇDARP çözümünde dikkat mekanizmasının önemi şu şekilde özetlenebilir:

- ✓ Önemli konumlara odaklanma: ÇDARP’de tüm müşteriler ve depolar aynı öneme sahip değildir. dikkat mekanizması, önemli müşteri-depo ilişkilerini öğrenerek, rotalama sırasında kritik noktalara öncelik verir (Yang & Chen, 2024).
- ✓ Büyük veri setlerinde verimlilik: Çok sayıda müşteri ve depo içeren büyük veri setlerinde, dikkat mekanizması önemli bilgilerin çıkarılmasını hızlandırır, böylece çözüm süresini kısaltır ve daha iyi optimizasyon sağlar (Yan *et al.* 2025).
- ✓ Esnek çözüm yapısı: Dikkat mekanizması, dinamik talepler ve değişken ortam koşullarına hızlı uyum sağlama yeteneği ile klasik yöntemlere kıyasla daha esnektir. Böylece değişen müşteri talepleri ve koşullarda bile etkili rotalar oluşturabilir (Luo, G., & Luo, 2024).
- ✓ Hibrit modellerle entegrasyon: Dikkat mekanizması, BKA ve DQN gibi meta-sezgisel ve RL yaklaşımlarıyla birleştirildiğinde, genel performansı artıran güçlü bir modül olarak işlev görür (Chen, J., & Luo, 2023).

DTBKA hibrit optimizasyon yöntemi, dikkat mekanizmasını BKA ile birleştirerek yenilikçi bir hibrit yaklaşım sunar. Bu modelin temel avantajı, BKA’nın geniş çözüm alanını tarama yeteneğini, dikkat mekanizmasının odaklanma ve kritik bilgileri önceliklendirme yeteneğiyle desteklemesidir. DTBKA’nın ÇDARP çözümündeki önemi şu şekilde açıklanabilir:

- ✓ Karmaşık veri ilişkilerini öğrenme: Dikkat mekanizması, müşteri ve depo konumları arasındaki önemli ilişkileri ve desenleri öğrenir (Li *et al.* 2024), böylece BKA’nın arama sürecini hızlandırır ve daha iyi yönlendirir.

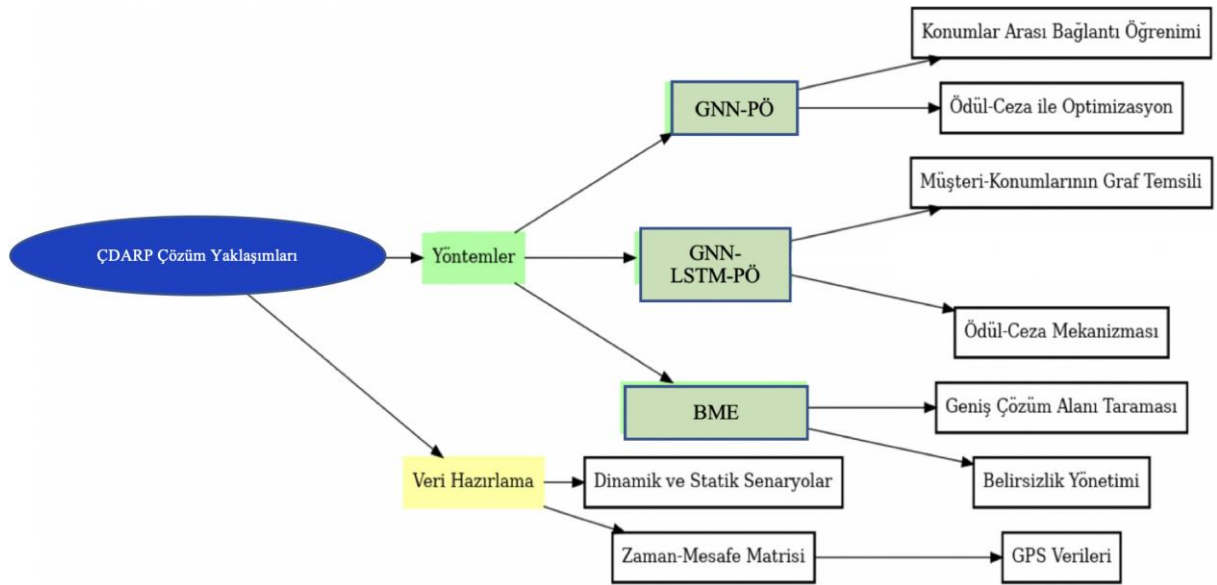
- ✓ Çözüm kalitesinde artış: BKA, çözüm alanını tararken yerel optima takılma riskine sahiptir (Chakraborty *et al.* 2023). Dikkat mekanizması, kritik alanlara odaklanarak BKA'nın daha hızlı küresel optimuma ulaşmasını sağlar.
- ✓ Esnek ve genelleştirilebilir yapı: DTBKA, hem statik hem de dinamik ÇDARP senaryolarında kullanılabilir ve farklı problem türlerine kolayca adapte edilebilir (Zhao *et al.* 2020).
- ✓ Büyük ölçekli problemler için uygunluk: Dikkat mekanizması sayesinde, büyük veri kümelerinde bile önemli bilgilere odaklanarak hesaplama süresini düşürür ve BKA'nın performansını artırır (Wang *et al.* 2024).

Çalışmada DTBKA modeli, rota verimliliği ve doğruluğu açısından üstün bir performans sergileyerek, karmaşık ÇDARP örnekleri için umut verici çözümler sunmuştur. Özellikle DTBKA modeli aracılığıyla meta-sezgisel ve DL yöntemlerinin entegrasyonu, gerçek dünya veri setlerini kullanarak pratik çözümler sağlamış ve simülasyonlar yerine gerçek veri setlerine odaklanarak mevcut literatüre katkı sağlamıştır (Erdem *et al.* under review d).

Dördüncü aşamada geliştirilen ileri düzey algoritmalar, geçmişe dayalı zaman-mesafe verilerini kullanarak rotaları optimize etme konusunda klasik yöntemlere kıyasla üstün performans sergileyerek, şehir içi dağıtım operasyonlarında hem rota verimliliğini hem de karar doğruluğunu artırmış; özellikle DTBKA modeli ile, gerçek zamanlı koşullara duyarlı, dinamik ve optimize edilmiş teslimat stratejileri geliştirilmiştir (Erdem *et al.* under review d).

Beşinci Aşama: Hibrit ve AI destekli modellerin geliştirilmesi

Bu aşamada, GNN-LSTM-RL, GNN-RL ve BME algoritmaların birleşiminden oluşan çeşitli methodların karşılaştırmasını içermektedir. Üçüncü aşamada kapsamlı GPS verileri ile oluşturulan zaman-mesafe matrisi kullanılmıştır (Şekil 8).



Şekil 8. Zaman-mesafe verisi ile güçlendirilmiş araç rotalama için optimizasyon algoritmalarının karşılaştırmalı süreci

Motivasyon

Bu aşamadaki çalışmanın temel motivasyonu, hem zamansal hem de mekânsal ilişkileri aynı anda modelleyebilen GNN ve LSTM yapılarını bir araya getirerek, çok değişkenli ve gerçek dünya verilerine dayalı karar destek sistemleri geliştirmektir. GNN, müşteri ve depo arasındaki topolojik bağlantıları öğrenmekte, LSTM ise geçmiş siparişlerin zaman serisi özelliklerini kavramaktadır. Bu sayede, dağıtım ağlarının yapısal ve zamansal dinamikleri eşzamanlı olarak analiz edilmiştir. Ancak bu iki modelin birlikte kullanımı, talep belirsizliği ve ani değişim gibi çevresel dinamikleri doğrudan modellemek konusunda yetersiz kalmıştır. Bu nedenle, sistemin değişken ortamlara adaptasyon kabiliyetini artırmak amacıyla Bulanık Mantık tabanlı Evrimsel Algoritmalar entegre edilmiştir. Böylece, belirsizliklerin yönetimi daha güçlü bir biçimde sağlanmış ve karar mekanizmasına esneklik kazandırılmıştır. Önceki aşamalarda kullanılan BKA, DQN ve dikkat mekanizması gibi algoritmalar zaman ve mekân bilgilerine belirli düzeylerde tepki verebilmiş olsa da, bu bilgileri doğrudan ve bütüncül biçimde öğrenmeye odaklı değildir. Örneğin, BKA algoritması çözüm uzayını biyolojik davranışlara dayalı keşif stratejileriyle tarayarak mekânsal ilişkileri dikkate alır, ancak zaman boyutunu doğrudan modelleyemez. DQN, çevreden aldığı geri bildirimler ile öğrenme gerçekleştiren bir RL algoritmasıdır, fakat klasik yapısı gereği zaman serilerini ya da müşteri-depo arasındaki topolojik bağları doğrudan modelleyemez. Zaman ve mekân verilerini yalnızca anlık karar süreçleri içinde değerlendirir ve uzun vadeli örüntüleri yakalayamaz. Benzer şekilde, dikkat mekanizması veri içindeki önemli özniteliklere odaklanmayı sağlar ve kritik bilgilere ağırlık verir. Ancak bu yaklaşım da zamansal desenleri ya da coğrafi yapıları doğrudan modellemeye yeterli değildir. Bu bağlamda, beşinci aşamada geliştirilen GNN-LSTM-RL ve BME

Algoritmalar gibi yaklaşımlar, zaman ve mekân bileşenlerini aynı model içinde bütüncül şekilde öğrenme hedefiyle geliştirilmiştir. GNN yapısı, müşteri ve depo noktaları arasındaki topolojik ilişkileri doğrudan öğrenebilmektedir. LSTM, geçmiş sipariş verilerinden oluşan zaman serisi örüntülerini yakalayarak talep dinamiklerini kavramakta etkili bir yapı sunmaktadır. Bulanık Mantık, çevresel belirsizlikleri ve ani değişkenlikleri modellemekte güçlüdür. RL mekanizması ise bu yapıların çıktıları doğrultusunda optimum kararları üretebilmektedir. Böylece, bu aşamada önerilen bütünleşik modelleme yaklaşımı, önceki yöntemlerde ayrı ayrı ele alınan zamansal ve mekânsal bilgileri tek bir yapı altında doğrudan öğrenerek, daha gerçekçi, esnek ve belirsizliklere karşı dayanıklı çözümler sunmaktadır (Erdem *et al.* under review d).

Yöntemlerin ÇDARP çözümündeki önemi: Kavramsal bir bakış

GNN-LSTM-RL yöntemi, graf yapılarının öğrenilmesi (GNN), zaman serisi özelliklerinin yakalanması (LSTM) ve ödül-ceza mekanizmasıyla optimizasyon (RL) süreçlerini birleştirerek ÇDARP’de kapsamlı bir çözüm sunar:

- ✓ GNN bileşeni, müşteriler arasındaki bağlantıları (örneğin konumlar arasındaki ilişkileri) öğrenerek rota optimizasyonunda güçlü temsiller oluşturur (Fellek *et al.* 2023).
- ✓ LSTM, müşteri talepleri veya sipariş değişimlerinin zaman içerisindeki desenlerini öğrenir ve bu bilgiyi gelecekteki rotaları tahmin etmek için kullanır (Yu *et al.* 2020).
- ✓ RL, bu öğrenilen temsiller ve zaman serisi bilgilerini kullanarak rotaları ödül-ceza mekanizmasıyla optimize eder (Delamer *et al.* 2024; He *et al.* 2023).

Bu kombinasyon, özellikle dinamik ve değişken talep durumlarında daha sağlam ve esnek çözümler sunar.

GNN-RL Yöntemi, graf tabanlı öğrenme ve RL süreçlerini bir araya getirerek, ÇDARP’nin temel yapısını (müşteri-depo bağlantıları, mesafe ilişkileri gibi) doğrudan modelleyebilir.

- ✓ GNN, müşterilerin ve depoların konumlarını bir grafik olarak temsil ederek bu yapıyı öğrenir ve rota optimizasyonu için uygun temsiller çıkarır (Wang & Liang, 2025; Almasan *et al.* 2022).
- ✓ RL, bu temsiller üzerinde kararlar alarak ödül-ceza mekanizması ile rotaları optimize eder.

Bu yaklaşım, statik ÇDARP problemlerinde yüksek performans sağlamakla birlikte, graf yapısının doğrudan kullanımı sayesinde genelleme yeteneği yüksek çözümler sunar (Chen *et al.* 2022; He *et al.* 2023).

BME Yöntemi, evrimsel algoritmaların geniş çözüm alanını tarama yeteneğini ve bulanık mantığın belirsizlikleri yönetme kabiliyetini birleştirir.

- ✓ Evrimsel algoritmalar, ÇDARP gibi çok büyük çözüm alanlarına sahip problemleri etkili bir şekilde tarayarak küresel optimuma yakın çözümler (Pytel, 2025) elde eder.
- ✓ Bulanık mantık, belirsiz veya eksik bilgilere dayalı karar verme süreçlerinde güçlüdür. Örneğin, müşteri taleplerindeki belirsizlikleri veya teslim sürelerindeki değişkenlikleri yönetmek için kullanılabilir (Sherinov, 2011).

Bu kombinasyon, özellikle belirsiz ve eksik verilere sahip senaryolarda etkili çözümler sunar ve klasik algoritmalara kıyasla daha esnek sonuçlar üretir.

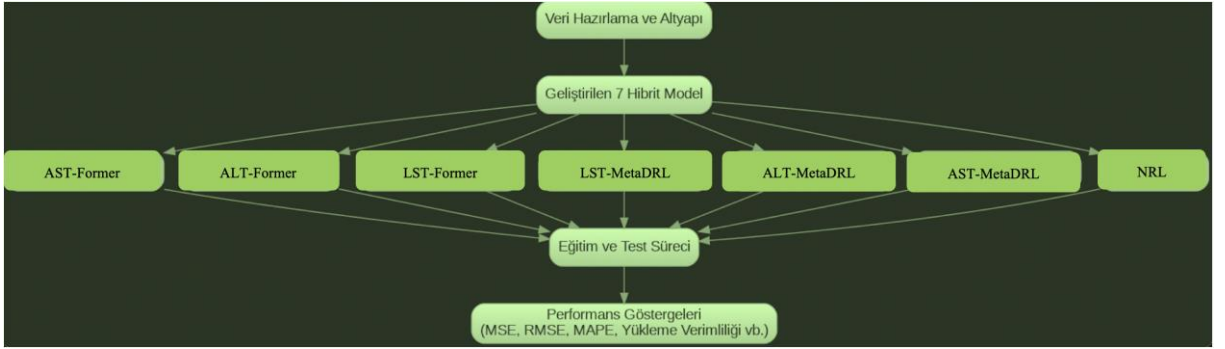
Elde edilen sonuçlarda, GNN-RL modeli, doğruluk, genelleme kapasitesi ve optimizasyon performansı açısından diğer modellere kıyasla en başarılı sonuçları sunmuştur. Eğitim ve test süreçlerinde düşük kayıp ve yüksek doğruluk oranlarıyla dengeli bir performans sergilemiştir.

Beşinci aşamada önerilen GNN-LSTM-RL ve BME algoritmaları, müşteri-depo ilişkilerini ve geçmiş sipariş desenlerini öğrenerek zaman ve mekân boyutlarını aynı anda modelleyebilmiş; bu sayede, şehir içi dağıtım operasyonlarında belirsizliklere karşı daha esnek, genellenebilir ve veri güdümlü karar destek mekanizmaları oluşturulmuştur (Erdem *et al.* under review d).

Altıncı Aşama: HLS tabanlı, derinlemesine öğrenme ve evrimsel modellerin bütünleşik uygulaması

Bu aşamada, önceki bölümlerde oluşturulan veri hazırlama süreçleri, zaman-mesafe matrisleri ve modelleme altyapıları temel alınarak, ileri düzey AI tabanlı algoritmaların karşılaştırmalı değerlendirilmesi gerçekleştirilmiştir (Erdem *et al.* under review c). Özellikle transformer tabanlı modeller, MAML, DRL, NEAT ve HLS uyumlu mimariler bu kapsamda değerlendirilmiştir. Gerçek dünya verileriyle (GPS, Mikro V15) desteklenen deneysel çalışmalar sayesinde, algoritmaların rota doğruluğu, yükleme verimliliği, genelleme kabiliyeti ve sistem düzeyinde uygulanabilirliği gibi çok boyutlu performans ölçütleri üzerinden detaylı analizler gerçekleştirilmiştir (Erdem *et al.* under review c). Bu aşama, geleneksel optimizasyon tekniklerinin ötesine geçen, donanım uyumlu ve öğrenen sistemlerin test edilerek, gerçek

zamanlı dağıtım ağlarına entegrasyon potansiyelini araştırmayı amaçlamaktadır. Böylece, akademik literatürde nadiren birlikte ele alınan algoritmik başarı ile donanımsal verimlilik, bu tez çalışmasında bütüncül olarak değerlendirilmiştir (Şekil 9).



Şekil 9. Zaman-mesafe verisi ile güçlendirilmiş araç rotalama için optimizasyon algoritmalarının karşılaştırmalı süreci

Motivasyon

Bu aşamadaki temel motivasyon, önceki aşamalarda geliştirilen BKA, DQN, Dikkat tabanlı model, GNN ve LSTM tabanlı modellerin sahip olduğu sınırlı bütüncül öğrenme yeteneklerini aşmak ve hem rota verimliliğini hem de model başarımını artırmak amacıyla daha ileri seviyeli, entegre AI algoritmalarının sistematik biçimde test edilmesidir. Önceki aşamalarda ele alınan algoritmalar, ya zamansal ya da mekânsal boyutlara odaklanmış; ancak bu bilgileri aynı anda ve doğrudan öğrenen birleşik yapılara sahip olmamıştır. Ayrıca, bu algoritmaların çoğu, değişken talep yapıları, çoklu araç turları ve dinamik ortam koşulları gibi gerçek dünya zorluklarına karşı sınırlı adaptasyon kapasitesi sunmuştur. Bu bağlamda, 6. aşamada kullanılan modellerin seçimi ve entegrasyonu şu motivasyonlara dayanmaktadır: Transformer tabanlı AST-Former, ALT-Former ve LST-Former modelleri sırasıyla erken, geç ve dengeleyici teslimat stratejilerini temel alarak müşteri önceliklendirme ve sıralama süreçlerini dikkat mekanizmalarıyla optimize etmektedir. Ancak bu modellerdeki temel eksiklik, ani değişen taleplere karşı adaptasyon kabiliyetinin sınırlı olması ve genellikle deterministik kalıplar üzerinden işlem yapmalarıdır. Bu nedenle, DRL ve MAML tabanlı modeller (örneğin LST-MetaDRL, ALT-DRL-MAML mekanizması (ALT-MetaDRL), AST-DRL-MAML mekanizması (AST-MetaDRL)), çevresel değişikliklere karşı daha yüksek öğrenme kapasitesi sağlamak için tasarlanmış, ancak bu modeller de yüksek hesaplama maliyetleri ve eğitim süresindeki dengesizliklerle öne çıkmıştır. Bu eksiklikleri gidermek adına NEAT-RL tabanlı NRL modeli gibi evrimsel yapılar devreye alınmış ve sinir ağı yapısının evrimsel olarak optimize edilmesiyle yapısal adaptasyon yeteneği artırılmıştır (Erdem *et al.* under review c). Tüm bu modellerin geliştirilmesi ve test edilmesindeki ortak hedef, yalnızca yazılım düzeyinde başarılı sonuçlar üretmek değil; aynı zamanda uyumluluğu sayesinde

donanım düzeyinde gelecekteki uygulamalara uygun, ölçeklenebilir ve gerçek zamanlı çözümler sunabilmektir. Böylece 6. aşamada önerilen çözümler, hem geçmiş aşamalarda gözlemlenen eksiklikleri adreslemekte hem de algoritmik zekâyı donanımsal verimlilikle birleştirerek yeni nesil lojistik sistemler için bütünsel bir çözüm çerçevesi sunmaktadır.

Yöntemlerin ÇDARP çözümündeki önemi: Kavramsal bir bakış

Altıncı aşamada geliştirilen yedi hibrit AI tabanlı model, çoklu depo, parçalı teslimat ve çoklu sefer gibi gerçek dünya lojistik kısıtlarını dikkate alan, güçlü, genellenebilir ve yüksek performanslı çözüm yapıları sunmaktadır. Bu yöntemler, klasik algoritmaların sınırlı genelleme ve adaptasyon kabiliyetlerine karşı, zamansal, mekânsal ve operasyonel belirsizliklerle başa çıkabilen DL ve RL yapıları ile tasarlanmıştır. Kavramsal olarak her bir modelin ÇDARP çözümündeki katkısı aşağıda özetlenmiştir:

- ✓ AST-Former: ASAP öncelikleme stratejisini kullanan bu model, acil müşteri taleplerini öne alarak zaman açısından kritik rotalama problemleri için uygundur. Transformer tabanlı yapısı sayesinde (Munir *et al.* 2024), erken teslimat önceliği olan dağıtım senaryolarında yüksek doğruluk ve düşük gecikme ile çözüm sunar.
- ✓ ALT-Former: ALAP stratejisiyle çalışan bu model, zaman esnekliği olan senaryolarda, geç teslimata tolerans sağlayarak araç kapasitesini daha verimli kullanmayı hedefler. Transformer mimarisinin zaman esnekliğini öğrenme kabiliyeti, bu tip problemler için uygundur (Ryu *et al.* 2023). Gevşek zaman kısıtlı ÇDARP durumlarında dengeli yük dağılımı ve rota planlaması sunar.
- ✓ LST-Former: Talep hacmine göre listelenen müşteri önceliklerini dikkate alarak, Transformer mimarisi ile rotaları optimize eden bu yapı, özellikle yüksek hacimli dağıtım operasyonlarında kapasite kullanımını maksimize eder. Mekânsal ilişkileri başarıyla öğrenerek rota sıralamasında kararlı çözümler üretir (Zhang *et al.* 2024).
- ✓ LST-MetaDRL: Listeleme temelli planlamayı DRL ve MAML ile birleştirir. Bu yapı, hem rotalama öğrenimini pekiştirir hem de MAML ile farklı lojistik senaryolara hızlı adaptasyon sağlar. Özellikle sık değişen müşteri talepleri veya trafik koşullarında güçlü bir çözüm sunar. Bu tür yapıların performansı, zaman-bağımlı ve çoklu seferli problemler için önerilen DRL tabanlı modeller de doğrulanmıştır (Mozhdehi *et al.* 2025).
- ✓ ALT-MetaDRL: Ertelenebilir teslimatların önceliklendirildiği ALAP yaklaşımına DRL ve MAML mekanizmaları entegre edilmiştir. Teslimatların daha geç planlanmasına olanak tanıyarak araç yüklemelerinde denge sağlanır. Aynı zamanda meta-öğrenme modülü sayesinde farklı dağıtım bölgelerine kolayca adapte olur.

DRL tabanlı farklı yapıların zaman-bağımlı çoklu tur optimizasyonlarında nasıl avantaj sağladığı detaylı biçimde görülmüştür (Mozhdehi *et al.* 2025).

- ✓ AST-MetaDRL: Erken teslimatların ön planda tutulduğu ASAP yaklaşımını DRL ve MAML ile birleştiren bu model, acil teslimat gereksinimlerinde hızlı ve duyarlı rota önerileri üretir. Anlık talep değişimlerine karşı yüksek tepki kabiliyeti sunar. Zaman hassasiyeti yüksek karar destek sistemlerinde DRL kombinasyonlarının başarı sağladığı görülmüştür (Munir *et al.* 2024).
- ✓ NRL: NEAT algoritmasıyla evrimsel olarak yapılandırılmış sinir ağı mimarisine sahip olan bu model (Peng *et al.* 2018; Kogan *et al.* 2022), ağ topolojisini veri karmaşıklığına göre özelleştirerek DRL'nin performansını artırır. Farklı araç türleri, müşteri profilleri ve talep desenlerine karşı esnek ve kendini uyarlayan çözümler sunar.

Özellikle LST-Former modeli; rota doğruluğu, yükleme verimliliği, hata metrikleri (MSE, RMSE, MAPE) ve genelleme kapasitesi açısından diğer tüm modellerden üstün performans sergilemiştir. Eğitim ve test süreçlerinde düşük kayıp oranları, yüksek doğruluk ve dar varyans değerleriyle öne çıkan bu model, dinamik talep koşulları altında dahi kararlı sonuçlar üretmiştir. Transformer tabanlı yapının sunduğu dikkat mekanizması, müşteriler arası mekânsal ilişkilerin başarılı biçimde modellenmesini sağlamıştır. Ayrıca, DRL ve MAML ile güçlendirilmiş modeller (örneğin LST-MetaDRL ve AST-MetaDRL), yüksek adaptasyon yetenekleriyle değişken ortam koşullarında stratejik esneklik sunmuştur. Bununla birlikte, bu hibrit modeller, hesaplama yükü açısından daha maliyetli olmuş ve bazı senaryolarda daha yüksek varyans değerleri gözlenmiştir. NRL modeli ise NEAT algoritması ile DRL'yi birleştirerek yapısal uyarlanabilirlik ve düşük hesaplama yükü arasında dengeli bir performans sergilemiştir. Sonuçlar genel olarak, klasik yöntemlerle kıyaslandığında, Transformer, DRL, MAML ve NEAT gibi AI destekli modellerin çoklu depo, parçalı teslimat ve çoklu sefer gibi karmaşık dağıtım senaryolarında önemli performans iyileştirmeleri sağladığını göstermektedir (Erdem *et al.* under review c).

Altıncı aşamada geliştirilen hibrit AI modelleri, çoklu depo ve çoklu sefer içeren parçalı teslimat senaryolarında, şehir içi dağıtım süreçlerinin dinamik, zaman kısıtlı ve yüksek hacimli doğasına doğrudan uyum sağlayarak gerçek zamanlı ve ölçeklenebilir çözüm yapıları sunmuştur.

KURAMSAL TEMELLER

ÇDARP, birden fazla depodan hareket eden araç filosu için kapasite, zaman aralığı ve müşteri talepleri gibi kısıtlamalar göz önünde bulundurularak en verimli rotaların belirlenmesini içeren karmaşık bir kombinatoriyal optimizasyon problemidir. NP-zor yapısı nedeniyle, ÇDARP için optimal çözümler bulmak oldukça zordur (Karakatič & Podgorelec, 2015). Son yıllarda, doğal süreçlerden ilham alan meta-sezgisel algoritmalar ve DL ile RL yöntemleri gibi AI teknikleri, ÇDARP'ye ölçeklenebilir ve verimli çözümler sunma konusundaki yetenekleri nedeniyle ön plana çıkmıştır (Yoo *et al.* 2024; Pitakaso *et al.* 2024; Chen *et al.* 2024; Li *et al.* 2024; Pan & Liu, 2023). Özellikle, meta-sezgisellerle DL'nin bir araya getirildiği hibrit yaklaşımlar, geliştirilmiş çözüm kalitesi ve hesaplama verimliliği sağlayarak dikkat çekici bir alan olmuştur (Li *et al.* 2024; Londoño *et al.* 2024; Bai *et al.* 2023; Pan & Liu, 2023; Shahbazian *et al.* 2024). Çalışmalar, genetik algoritma (GA) gibi meta-sezgisellerin kümeleme teknikleriyle birlikte kullanıldığında müşteri atamalarını ve genel rota planlamasını optimize ettiğini göstermiştir (Prabu *et al.* 2019; Su *et al.* 2024; Wang *et al.* 2023; Erdem *et al.* under review a).

ÇDARP İçin AI Tabanlı Çözümlerdeki Gelişmeler: Son Trendlerin ve Gelecekteki Yönlerin İncelenmesi

ÇDARP'yi ele almak için son on yılda önerilen AI tabanlı yaklaşımları belirlemek üzere sistematik bir literatür incelemesi sunuldu ve bu alanda gelecekteki araştırma yönleri tartışıldı. Gerçek zamanlı verilerin Nesnelerin İnterneti (IoT) ile entegrasyonu, daha dinamik ve uyarlanabilir çözümler oluşturulması açısından gelecekteki araştırmaların odak noktası olmalıdır. Ayrıca, meta-sezgisellerle eşleştirilen RL algoritmaları, daha büyük ölçekli ve gerçek dünya ÇDARP örneklerini ele almak için önemli bir potansiyele sahiptir. E-ticaret lojistiği, sağlık lojistiği ve acil durum müdahalesi gibi pratik alanlarda AI'nın uygulanması, operasyonel iyileştirmeler sağlar. Gerçek dünya karmaşıklıklarını, özellikle talep ve kapasite belirsizliklerini ele almak için maliyet, rota dengesi ve enerji verimliliği arasında denge sağlayan çok amaçlı optimizasyon çalışmalarına devam edilmesi kritik öneme sahiptir.

ÇDARP, klasik ARP'nin bir genişlemesi olarak, çoklu depo ve araçlar arasında müşteri hizmetini optimize etmeye odaklanır. AI tabanlı yaklaşımlar, bu karmaşık problemin getirdiği zorlukları ele almak için pratik araçlar sunmaktadır. Geleneksel yöntemler genellikle büyük

veri kümelerini ve birçok değişkeni ele almakta zorlanırken, AI teknikleri bu karmaşıklıkları etkili bir şekilde yöneterek daha yalın ve ölçeklenebilir çözümler sağlar.

AI teknikleri, özellikle GA (Holland, 1975), Karınca Kolonisi Optimizasyonu (ACO) (Dorigo & Gambardella, 1997), Parçacık Sürü Optimizasyonu (PSO) (Kennedy & Eberhart, 1995), DL (Hinton *et al.* 2006) ve RL (Sutton & Barto 1999) gibi meta-sezgisel algoritmalar, ÇDARP gibi karmaşık problemleri çözmek için geniş bir şekilde uygulanmaktadır. GA, ÇDARP gibi NP-zor problemler için genetik işlemler kullanarak etkili çözümler sunar. Araştırmalar, GA'nın özellikle büyük veri setleri için etkili olduğunu ve geleneksel yöntemlerden daha hızlı optimal ya da optimale yakın çözümler sağladığını göstermiştir (Castillo *et al.* 2024; Yoo *et al.* 2024). Benzer şekilde, ACO, karıncaların yiyecek bulma davranışını taklit eder ve çoklu depo ve müşteri rotalama problemleri için optimal çözümler sunar; bu da bu zorlukların ele alınmasında önemli bir araçtır (Asghari & Al-e, 2021). Bu meta-sezgisel yöntemler, çözüm uzayını etkili bir şekilde keşfederek daha hızlı sonuçlar sunar. PSO ve DL yöntemleri, büyük veri setlerini işleme ve anlamlı içgörüler çıkarma konusunda güçlü bir performans sergiler. PSO, kuşların sosyal davranışlarından esinlenmiştir ve çoklu depo ARP'ler için hız ve verimlilik açısından optimizasyon sağlar. PSO'nun büyük veri kümelerine uyarlanabilirliği, ona geleneksel yöntemler karşısında önemli bir avantaj sağlar (Pitakaso *et al.* 2024). Öte yandan, DL teknikleri, ÇDARP'de müşteri talepleri ile depo konumları arasındaki karmaşık ilişkileri tanımlamada mükemmeldir ve bu süreçte genelleme yeteneklerini artırır. Ayrıca, RL, çevresel değişikliklere yanıt vererek araç rotalarını optimize etmek için dinamik lojistik ve rotalama problemleri için önemli bir AI yöntemi olarak öne çıkmaktadır. RL, trafik, müşteri talepleri ve diğer değişkenlere dayalı rota güncellemeleri sağlayarak gerçek zamanlı optimizasyon sunar (Hao *et al.* 2024; Anuar *et al.* 2022). Bu AI teknikleri, ÇDARP çözümünde daha verimli, ölçeklenebilir ve uyarlanabilir çözümler sunar.

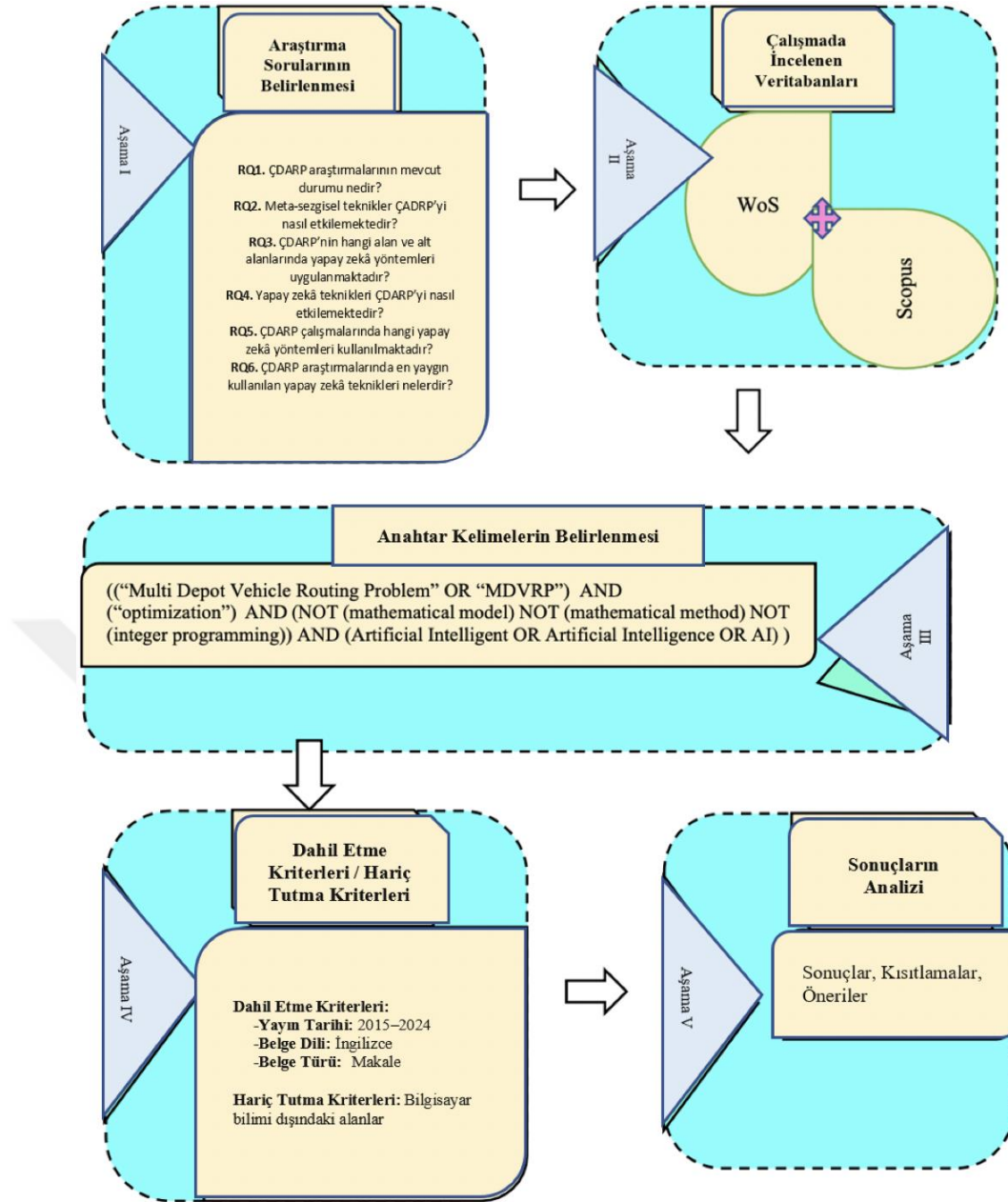
Son yıllarda ÇDARP ve AI tabanlı çözümler, lojistik, çevre dostu ulaşım ve operasyonel verimlilik açısından kritik rolleri nedeniyle dikkat çekmektedir. Örneğin, çevresel olarak sürdürülebilir çözümlerle ÇDARP problemlerinin ele alınmasının önemi giderek daha fazla kabul görmektedir. Araştırmacılar, kentsel lojistikte mikro-hub dağıtımı için Agile yönlendirme algoritmaları geliştirmiş (Chen *et al.* 2024) ve iş birliğine dayalı ARP'lerde emisyonları azaltmaya yönelik yöntemler geliştirmiştir (Asghari & Al-e, 2021). Yoo *et al.* (2024), ARP'leri çözmede meta-sezgisellerin kritik rolünü vurgularken, Feng *et al.* (2020), AI'nın bilgi aktarımı yoluyla ARP'leri hızlandırdığını göstermiştir. Wang *et al.* (2022), çoklu depo iş birliğine dayalı rotalama problemlerine zaman aralıkları ekleyerek operasyonel verimliliği artırmayı

hedeflemiştir. Bu çalışmalar, lojistikte optimizasyon problemlerini çözmede AI ve meta-sezgisel algoritmaların önemini vurgulamaktadır.

Yukarıda özetlenen gelişmeler doğrultusunda, bu çalışma 2015–2024 yılları arasında yayımlanan literatürü sistematik bir yaklaşımla incelemiş ve ÇDARP ile AI tabanlı tekniklerin etkileşimini çok boyutlu olarak analiz etmiştir. ÇDARP çözümleri ve ilgili AI tekniklerini, meta-sezgiseller, DL ve RL gibi yöntemleri kategorize edilmiştir. Araştırmacıların giderek karmaşık lojistik problemleri çözmek için uyguladığı AI yöntemleri bibliyometrik bir analizle ayrıntılı olarak incelenmiştir. Bu çalışma, ÇDARP araştırmalarında en sık kullanılan AI yöntemlerini inceleyerek mevcut uygulamaları özetler ve çoklu depo senaryolarında araç rotalamayı optimize etmek için gelecekteki araştırma yönlerini önerir. Şekil 10, inceleme aşamalarını; Şekil 11, analiz edilen makale sayısını sunmaktadır (Erdem *et al.* under review a).

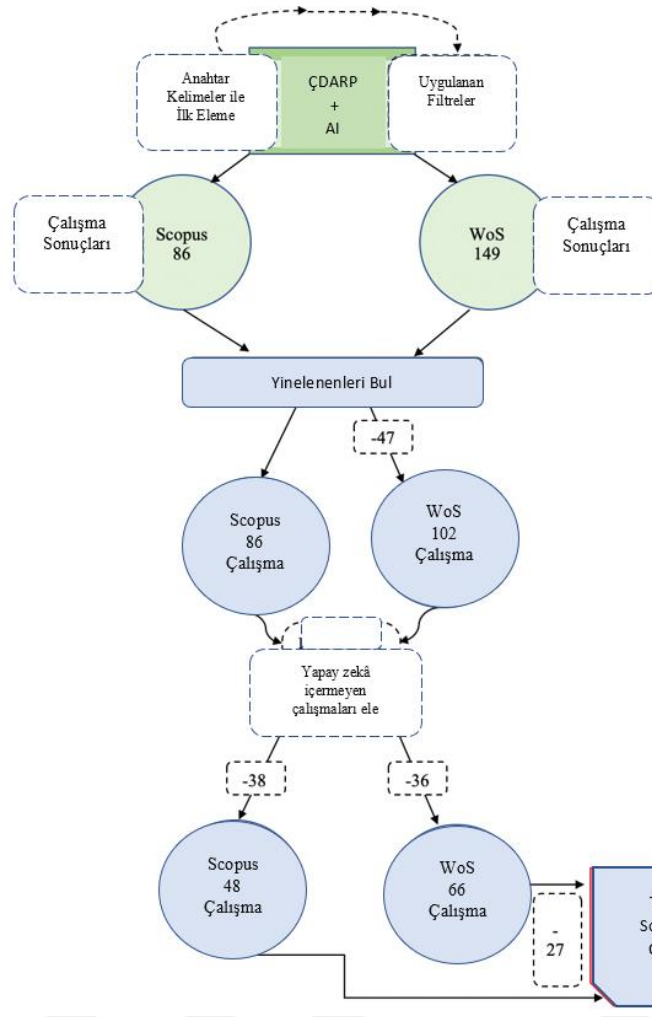
Yazarlar, çalışmanın amaçlarını açıkça tanımlamak ve kapsamla uyumlu kalmasını sağlamak için aşağıdaki araştırma sorularını tasarlamışlardır (Erdem *et al.* under review a):

- RQ1. ÇDARP araştırmalarının mevcut durumu nedir?
- RQ2. Meta-sezgisel teknikler ÇDARP'yi nasıl etkiler?
- RQ3. ÇDARP'nin hangi alanlarında ve alt alanlarında AI yöntemleri uygulanmaktadır?
- RQ4. AI teknikleri ÇDARP'yi nasıl etkiler?
- RQ5. ÇDARP çalışmalarında hangi AI yöntemleri kullanılmaktadır?
- RQ6. ÇDARP araştırmalarında en yaygın kullanılan AI teknikleri nelerdir?



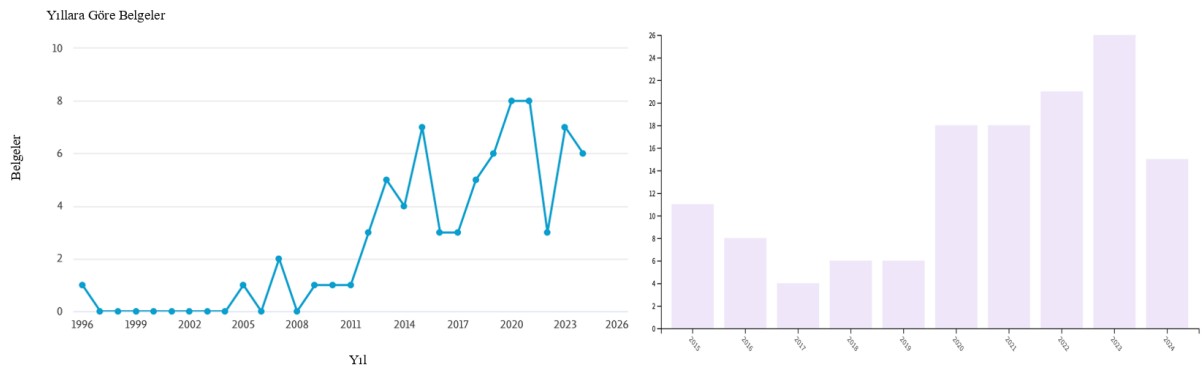
Şekil 10. Sistematik derleme sürecinin adımları (Erdem *et al.* under review a)

ÇDARP ile yapılan çalışmalar ile ilgili WoS'ta 149 ve Scopus'ta 86 çalışma taranmıştır. Bu sonuçların detayları Şekil 11'de sunulmaktadır. Tarama sonucunda toplam 87 özgün çalışmaya ulaşılmış, yinelenen kayıtlar (n=47) ve yapay zekâ içeriği barındırmayan ya da kapsam dışı olan çalışmalar (n=27) elenmiştir. Böylece, AI odaklı ve uygulama yönü olan 87 çalışmayla sistematik değerlendirme gerçekleştirilmiştir.



Şekil 11. Arama sonuçlarına göre incelenen makale sayısı (Erdem *et al.* under review a)

Şekil 12’de gösterildiği üzere, yıllar içinde yayın sayısında istikrarlı bir artış gözlemlenmiş, özellikle 2021 ve 2023 yıllarında önemli artışlar kaydedilmiştir. İncelenen çalışmaların yaklaşık %19,5’i 2023 yılında yayınlanmış olup, bu durum bu alandaki araştırma büyümesini açıkça ortaya koymaktadır. Bu eğilim, AI tekniklerinin ÇDARP’deki artan önemini ve uygulamasını vurgulamaktadır.



Şekil 12. Scopus ve WoS için 2015’ten 2024’e kadar olan makale yayın eğilimleri (Erdem *et al.* under review a)

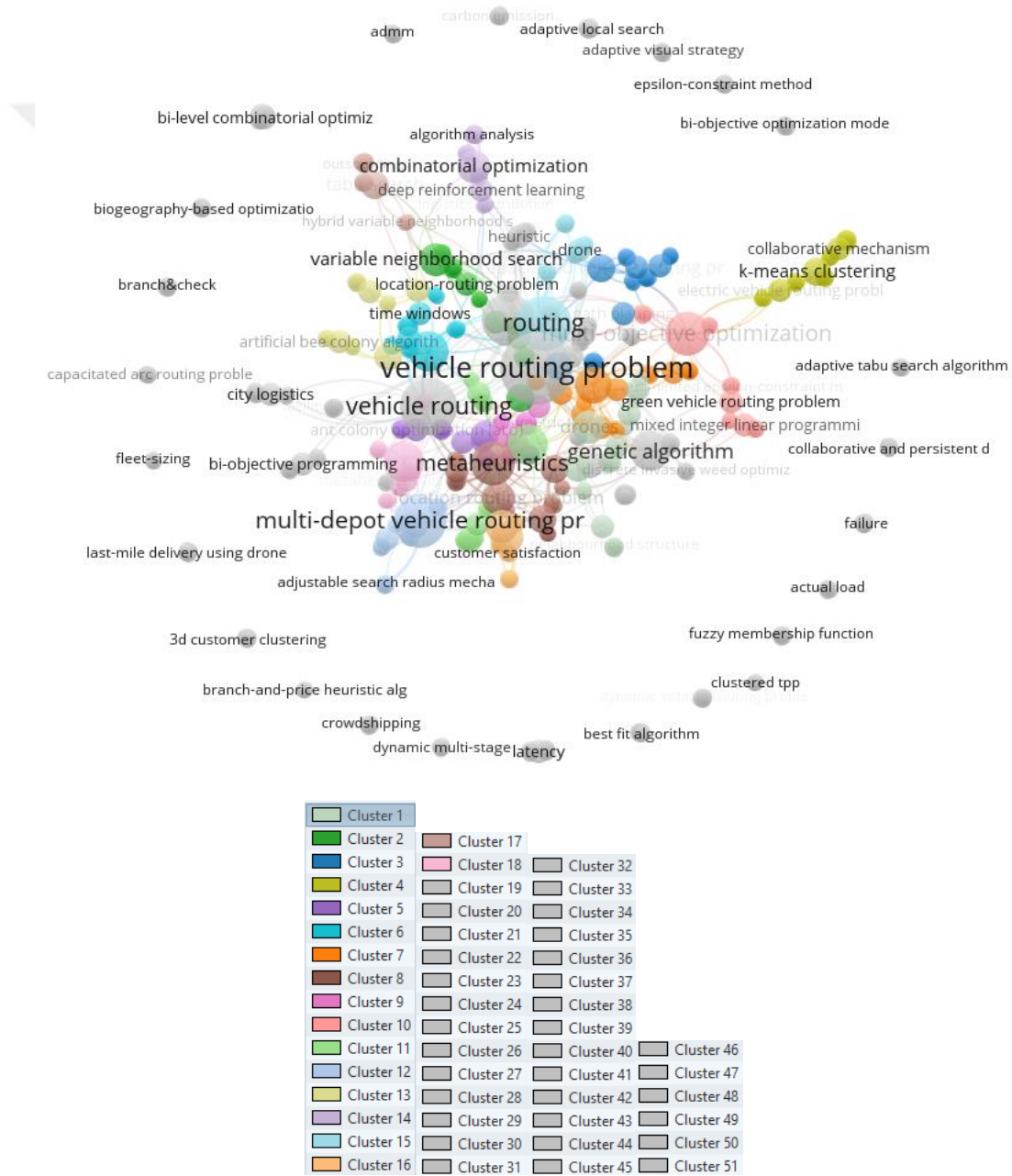
Tablo 2, WoS ve Scopus'taki en çok atıf alan üç çalışmayı ve bu çalışmaların odaklandığı özel uygulama alanlarını göstermektedir. WoS ve Scopus'ta en çok atıf alan makale (Karakatič & Podgorelec, 2015), ÇDARP'nin çözümünde karşılaşılan zorlukları ele almak için tasarlanmış çeşitli GA'ları incelemektedir. Bu çalışma, WoS'ta 200 atıf ve Scopus'ta 238 atıf almıştır. Çalışma, GA'ların ÇDARP için geniş çözüm uzaylarını etkili bir şekilde keşfetmedeki önemini vurgulamakta ve karmaşık lojistik operasyonların optimize edilmesindeki faydasını ortaya koymaktadır. İkinci en çok atıf alan çalışma (Du *et al.* 2017), tehlikeli maddelerin taşınmasında ulaşım riskini en aza indirmek için bir bulanık iki seviyeli programlama modeli sunmaktadır. Bu çalışma, WoS'ta 106 kez ve Scopus'ta 125 kez atıf almıştır. Bulanık Mantık entegrasyonu ile, bu araştırma, belirsizlikleri ele almak ve ÇDARP senaryolarında rota çözümlerini optimize etmek için AI tabanlı karar verme süreçlerinin kullanımını göstermektedir. Çalışma, araç rotalama problemlerinde AI uygulamalarını geliştirmede kritik bir rol oynadığını vurgulamaktadır. Üçüncü en çok atıf alan çalışma (Karakatič, 2021), elektrikli araç rotalamasında doğrusal olmayan şarj sürelerini optimize etmek için bir Genetik Algoritma kullanmaktadır. Bu makale, WoS'ta 89 kez, Scopus'ta ise 103 kez atıf almıştır. GA'ların bu bağlamda uygulanması, elektrikli araç şarjı gibi özel rotalama zorluklarını ele almada adaptasyon ve etkinliğini göstermektedir. Bu araştırma, GA'ların ÇDARP ve ilgili problemlerin çözümünde AI tabanlı bir yaklaşım olarak önemini pekiştirmektedir.

Tablo 2. En Çok Atıf Alan Makaleler (Erdem *et al.* under review a)

Yazarlar	Yıl	Max Atıf Sayısı
Karakatič and Podgorelec	2015	238
Du <i>et al.</i>	2017	125
Karakatič	2021	103

ÇDARP çözümleri ve AI teknikleriyle ilgili yaygın olarak kullanılan anahtar kelimelerin belirlenmesi, mevcut araştırma eğilimlerinin odak noktalarını anlamak için kritik bir öneme sahiptir. Bu amacı gerçekleştirmek için yazarlar, “eş-zamanlı terim haritası” analizi gerçekleştirmiştir. Bu analiz, ilgili kavramları temsil eden 51 farklı küme ortaya çıkarmış ve sonuçlar Şekil 13 ve Tablo 3’de gösterilmiştir. Haritalama, belirgin anahtar kelimeleri, bunların birbirleriyle olan ilişkilerini ve kümeleme desenlerini vurgulamaktadır. Her küme, aynı küme içindeki terimlerin farklı kümelerdeki terimlere göre daha fazla benzerlik taşıdığını ifade eden belirli bir renk ile temsil edilmiştir. Anahtar kelimeler arasındaki bağlantılar, terimler arasındaki bağlantı gücünü gösteren çizgilerle görselleştirilmiştir. Analiz, seçilen literatürde en sık kullanılan anahtar kelimeleri belirlemiştir. Bunlar arasında “ARP” (yüksek sıklık ve toplam bağlantı gücü), “ÇDARP” (belirgin sıklık, güçlü bağlantı gücü) ve “meta-sezgisel” (önemli sıklık, sağlam bağlantı gücü) bulunmaktadır. Bu anahtar kelimeler, “GAs,” “Deep Learning”

ve “optimization” gibi diğer terimlerle birlikte, ÇDARP’yi çözmek için gelişmiş AI ve hibrit tekniklerin kullanımına yönelik süregelen araştırma odağını vurgulamaktadır. Tablo 3, kümeleme ve anahtar kelime etkileşimlerini ayrıntılı olarak açıklayarak her kümeyi belirli etiketlerle ilişkilendirmektedir. Örneğin: “Adaptive ACO, Machine Learning (ML)” (Küme 1), “GAs, Optimization Techniques” (Küme 2), “RL, Deep Q-Networks (DQNs)” (Küme 4). Bu analiz, ÇDARP’de uyarlanabilir ve hibrit algoritmaların yaygın kullanımını ortaya koyarak, DL ve gerçek zamanlı optimizasyon stratejilerinin entegrasyonuna yönelik eğilimi doğrulamaktadır. Bu yaklaşımın, çözüm kalitesini ve ölçeklenebilirliği artırmada önemli bir rol oynadığı görülmektedir (Erdem *et al.* under review a).



Şekil 13. Etiket bazlı anahtar kelime eş-zamanlı haritası (Erdem *et al.* under review a)

Tablo 3. 51 Kümeye Ait Etiketler (Erdem *et al.* under review a)

Kümeler	Etiketler
1	Adaptive ACO, ML, Heuristic Algorithms
2	GAs, Optimization Techniques, Computational Efficiency
3	Swarm Intelligence, Particle Swarm Optimization, Collaborative Search
4	Reinforcement Learning, DQNs, Policy Optimization
5	Multi-Depot ARP, Fleet Management, Route Planning
6	Neural Networks, Training Models, Backpropagation
7	Data Preprocessing, Feature Selection, Normalization
8	Graph Theory, Shortest Path, Dijkstra's Algorithm
9	Supply Chain Optimization, Inventory Management, Logistics
10	ARP, Customer Demands, Delivery Scheduling
11	Time Complexity, Computational Costs, Scalability
12	DL, Convolutional Neural Networks, Image Processing
13	Dynamic Programming, State Transition, Optimal Solutions
14	Metaheuristic Methods, SA, Global Optimization
15	Reinforcement Strategies, Learning Rates, Rewards Functions
16	Constraint Satisfaction, Feasibility Checks, Optimization Constraints
17	Big Data Analytics, Data Mining, Pattern Recognition
18	Evrimsel Computation, Selection Mechanisms, Genetic Diversity
19	Hybrid Algorithms, Combining Heuristics, Enhanced Solutions
20	Artificial Intelligence, Expert Systems, Automated Decision-Making
21	Bulanık MantıkLogic, Uncertainty Management, Decision Theory
22	Stochastic Processes, Random Variables, Probability Distributions
23	Computational Geometry, Spatial Data Structures, Pathfinding
24	Algorithmic Complexity, Approximation Algorithms, NP-Hard Problems
25	Load Balancing, Parallel Computing, Resource Allocation
26	Cloud Computing, Distributed Systems, Virtualization
27	Optimization in Transportation, Traffic Management, Route Efficiency
28	Scheduling Algorithms, Job Sequencing, Priority Queues
29	Heuristic Evaluation, Search Spaces, Solution Quality
30	Monte Carlo Simulations, Random Sampling, Probabilistic Analysis
31	Machine Vision, Object Detection, Edge Detection
32	Ant Colony System Variants, Trail Pheromone, Solution Convergence
33	Traveling Salesman Problem, Path Optimization, Tour Construction
34	Network Optimization, Graph Algorithms, Connectivity
35	Clustering Methods, K-Means, Hierarchical Clustering
36	Data Augmentation, Synthetic Data Generation, Training Improvements
37	Algorithmic Frameworks, Benchmarking, Performance Metrics
38	ARP Extensions, Time Windows, Service Constraints
39	Decision Trees, Random Forests, Ensemble Learning
40	Evrimsel Strategies, Mutation Operators, Chromosomal Encoding
41	Learning Algorithms, Gradient Descent, Overfitting Control

Tablo 3. (devamı)

42	Parallel Heuristics, Load Distribution, Multi-Threading
43	Predictive Modeling, Regression Analysis, Forecasting
44	Semantic Analysis, Natural Language Processing, Text Mining
45	Energy Efficiency in Computing, Power Management, Green IT
46	Optimization in Logistics, Depot Assignments, Vehicle Capacities
47	Data Visualization, Graphical Models, Reporting Tools
48	Model Training Pipelines, Hyperparameter Tuning, Validation Techniques
49	Route Optimization Algorithms, Distance Matrices, Cost Functions
50	Bio-Inspired Algorithms, Natural Process Simulation, Adaptive Learning
51	Adaptive Systems, Self-Learning Mechanisms, Continuous Improvement

Analiz edilen makaleler detaylı bir şekilde incelenmiş ve AI teknikleri, AI tabanlı meta-sezgisel algoritmalar, RL ve DL uygulamaları ile ÇDARP için optimizasyon teknikleri ve uygulamaları olmak üzere dört ana başlık altında tartışılmıştır. Her bir bölüm, alt başlıklara ayrılmış ve çalışmada analiz edilen makaleler tablo formatında özetlenerek Şekil 13 ve Tablo 3’de belirtilen özelliklere göre değerlendirilmiştir. Ayrıca, her kategorideki bazı çalışmalar daha ayrıntılı bir şekilde ele alınmıştır. Kısaltmaları özet tablosunda verilen algoritmaların tam adlarının yer aldığı kapsamlı bir liste ise Tablo 4 ve Tablo 5’de sunulmuştur. Çalışma, Şekil 13’de belirtilen özelliklerle sınırlı olmakla birlikte, bu konuda önemli çalışmaları şu şekilde vurgulamaktadır:

- DRL’ye dayalı bir çalışmada, ölçeklenebilir bir çoklu karınca kolonisi yaklaşımı ile birleştirilen reaktif çok ajanlı bir sistem kullanılmış ve ÇDARP’deki karmaşık rotalama sorunlarını çözerek verimlilik artırılmıştır (Sahar *et al.* 2021).
- Ayrıca, farklı dikkat mekanizmalarının uygulanması, ÇDARP çözümleri için daha hassas ve hedefe yönelik stratejilerin geliştirilmesine olanak tanımıştır (Li *et al.* 2024).
- Chaabani & Said (2019), iki seviyeli üretim-dağıtım planlama sistemi içinde transfer öğrenimini uygulayarak daha verimli ÇDARP çözümleri geliştirmiştir.
- Son olarak, de Oliveira *et al.* (2016), ÇDARP için iş birliğine dayalı bir ortak evrim algoritması kullanarak çözümlerin etkinliğini ve esnekliğini artırmıştır.

AI teknikleri

AI teknikleri, ARP üzerinde önemli bir rol oynamaktadır. Özellikle, AI’nın çeşitli teknikleri keşfetmesiyle bu alanda önemli ilerlemeler kaydedilmiştir. Son yıllarda, sürü zekâsı ve hibrit algoritmalar gibi farklı AI teknikleri, ÇDARP çözümleri için incelenmiştir. Bu teknikler arasında, sürü zekâsı karmaşık rotalama zorluklarını ele almakta önemli bir potansiyel

göstermiştir. ACO, PSO ve Yapay Arı Kolonisi (ABC) gibi sürü zekâsı teknikleri, çeşitli araştırma çalışmalarında yaygın olarak kullanılmıştır.

- Gu *et al.* (2022), ABC algoritmasını kullanarak genetik ve ağgözlü algoritmalara kıyasla önemli performans iyileştirmeleri göstermiştir.
- Ayrıca, hibrit yaklaşımlar, yenilikçi çözüm stratejileri olarak ortaya çıkmıştır. Zeki Su Damlası (IWD) algoritması ile ACO'yu birleştiren bir yöntem, ÇDARP için hibrit bir çözüm sunmuştur. Samsuddin *et al.* (2020), bu yaklaşımları birleştirerek karmaşık ÇDARP zorluklarını ele almış ve önemli sonuçlar elde etmiştir.
- Benzer şekilde, Shi *et al.* (2020), atık toplama problemlerini ÇDARP çerçevesinde ele alarak sezgisel bir çözüm önermiş ve sezgisel yöntemlerin farklı alanlarda çok yönlü uygulanabilirliğini vurgulamıştır.
- Prabu *et al.* (2019), ÇDARP'de daha yüksek uyumlu çözümler elde etmek için başlangıç popülasyonu oluşturmayı iyileştiren gelişmiş bir genetik algoritma geliştirmiştir.
- Ateşböceği algoritmasının iyileştirilmesi (Yang, 2009) ve çeşitli optimizasyon tekniklerini entegre eden hibrit yaklaşımların geliştirilmesi (Londoño *et al.* 2024), ÇDARP için daha sağlam çözümleri ileriye taşımıştır.

Transfer öğrenimi, ARP çözümlerine önemli katkılar sağlayan bir başka yaklaşımdır. Örneğin, Huo, müşteri verilerine transfer öğrenimi uygulayarak ARP çözümlerini hızlandırmıştır (Huo, 2021). Bu çalışma, ML ve transfer öğrenimi tekniklerinden elde edilen içgörülerini kullanarak rotalama süresinde önemli bir azalma sağlamıştır. Ayrıca, farklı modellerin ve metodolojilerin entegrasyonu, daha sağlam ve verimli çözümlerin geliştirilmesini kolaylaştırmıştır. Ayyıldız ve arkadaşları, zaman aralıklarına dayalı nakit taşıma süreçlerini optimize etmek için yapay sinir ağlarını karma tam sayılı matematiksel bir modelle birleştirmiştir (Ayyıldız *et al.* 2022). Kapsamlı incelemeler, AI'nın ARP ile ilgili araştırmalar üzerindeki derin etkisini vurgulamaktadır. Özellikle, Bai *et al.* (2023), ARP çözümlerini iyileştirmede AI ve ML yöntemlerinin etkinliğini incelemiştir. Ayrıca, analitik yöntemler, ARP çözümlerinin geliştirilmesine dair daha derin içgörüler sağlamada etkili olmuştur. ML ve analitik yöntemlerin entegrasyonu, rotalama çözümlerinin geliştirilmesi ve sonuçların optimize edilmesi üzerinde önemli bir etki yaratmıştır. AI teknikleri, acil durum müdahalesi ve halk sağlığı lojistiği gibi alanlarda da uygulanmıştır. Maleki *et al.* (2024), COVID-19 aşılarının dağıtımını ve atık yönetimini optimize etmek için bulanık zaman aralıklarını kullanmıştır. Bu teknik, belirsizlikleri modellemek ve aşı merkezlerinin memnuniyetini artırmak için AI tabanlı Bulanık Mantık yöntemlerinden yararlanmıştır. Benzer bir yaklaşım, pandemi sırasında dağıtım

lojistiğini optimize etmek için kullanılmıştır. Chen *et al.* (2023), COVID-19 maskesi dağıtımını optimize etmek için ML ve sezgisel optimizasyonu birleştirerek lojistik süreçlerde zaman ve maliyet verimliliği sağlamıştır (Erdem *et al.* under review a).

AI tabanlı meta-sezgisel yaklaşımlar

Son yıllarda, AI ile geleneksel optimizasyon tekniklerini birleştiren çeşitli hibrit yaklaşımlar, ÇDARP'nin çözüm etkinliğini artırmak için geliştirilmiştir. Örneğin, Moonsri *et al.* (2022), ÇDARP örneklerini ele almak için önemli ilerlemeler sağlayan geliştirilmiş bir diferansiyel evrim algoritması sunmuş ve hibrit optimizasyon stratejilerinin etkinliğini vurgulamıştır. Ayrıca, Stodola (2018), ÇDARP çerçevesinde operasyonel ihtiyaçlara uyum sağlamak için meta-sezgisel yöntemlerin esnekliğini ön plana çıkararak farklı optimizasyon kriterlerini incelemiştir. Meta-sezgisel algoritmalar, büyük veri kümelerini yönetme ve ÇDARP gibi NP-zor problemlerle ilişkili belirsizlikleri ele alma yetenekleri nedeniyle yaygın olarak kullanılmaktadır (Stodola & Mazal, 2016). Doğadan ilham alan yöntemler, örneğin ACO (Stodola & Mazal, 2016), rotalama zorluklarında dikkate değer bir uyarlanabilirlik sergilemiştir. Benzer şekilde, Balina Optimizasyon Algoritması (Dewi & Utama 2021), hesaplama verimliliğini artırmada etkili olmuştur. Gerçek dünya uygulamalarında, Gu ve arkadaşları, karmaşık rotalama problemlerini ele almak için ABC algoritmasının etkinliğini araştırmış ve geleneksel açgözlü yöntemlere kıyasla %70'lik bir iyileşme bildirmiştir (Gu *et al.* 2022). Benzer şekilde, Khodashenas *et al.* (2023), bulanık-güçlü optimizasyon çerçevesi kullanarak alma ve teslimat maliyeti belirsizliklerini ele almış ve AI ile sağlam optimizasyon tekniklerinin başarılı entegrasyonunu göstermiştir. Ayrıca, Çoklu Liderli Salp Sürüsü Algoritması (MLSSA) gibi gelişmiş hibrit yöntemler, ileri beslemeli sinir ağlarının eğitimini iyileştirerek karmaşık problemleri etkili bir şekilde çözmüştür ve hibrit algoritmaların çok boyutlu problem çözmedeki çok yönlülüğünü bir kez daha kanıtlamıştır (Bairathi & Gopalani 2021). Meta-sezgisel algoritmalar, ARP'lerde, özellikle çoklu depo ve işbirlikçi rotalama problemlerinde sıkça kullanılmaktadır. AI tabanlı meta-sezgisel algoritmalar, ARP'lerde enerji ve öncelik kısıtlanmalı operasyonlar için GA kullanılması (Chen *et al.* 2019)), bulut bilişim iş akışlarını optimize etmek için kaotik bir çerçeve ile entegre edilen Gri Kurt Optimizasyon Algoritması (Mohammadzadeh *et al.* 2021) ve özel servis süresi gereksinimlerini çözmek için ACO Algoritması gibi karmaşık optimizasyon görevlerine uygulanmıştır (Wang *et al.* 2019). ARP çözümleri için geliştirilen hibrit yayılma tahmin algoritması gibi birkaç algoritma yenilikçi problem çözme yaklaşımlarını temsil etmektedir (Pérez-Rodríguez & Hernández-Aguirre 2019). Harris Hawks Optimizasyonu, ARP'leri etkili bir şekilde çözmek için kullanılmıştır (Alweshah *et al.* 2022). Bununla birlikte, ACO, büyük veri kümeleri için sağlam

sonuçlar sağlmasına rağmen, uzun işlem süreleri ve yerel optimumlara takılma gibi sınırlamalara sahiptir (Xie *et al.* 2023; Gu *et al.* 2022). Benzer şekilde, GA'lar (Yao *et al.* 2015) genellikle etkili olsalar da yerel optimuma takılma riski taşır. Hibrit ve sezgisel algoritmalar, karmaşık veri kümelerini yönetme yetenekleri nedeniyle optimizasyon süreçlerinde tutarlı bir şekilde tercih edilmektedir. Chen *et al.* (2024), lojistik ağlarını optimize etmek için yerel arama algoritmaları önermiş, Wang *et al.* (2023), dinamik rotalama senaryolarında meta-sezgisellerin performansını test etmiştir.

Bu çalışmalar, araştırmacıları gerçek zamanlı uygulamalarda benzer hibrit yaklaşımları incelemeye teşvik etmektedir. Birden fazla tekniği birleştiren hibrit algoritmalar, ÇDARP'nin çok boyutlu doğasını ele almak için özellikle etkili hale gelmiştir. Son beş yılda, araştırmacılar AI tekniklerini ÇDARP çözümüne uygulamada önemli ilerlemeler kaydetmiştir. Doğadan ilham alan algoritmalar ve hibrit yöntemler bu alanda özellikle etkili olmuştur. Bu ilerlemelere rağmen, bazı algoritmaların daha büyük veri kümelerinde performansını tam olarak keşfetmek için daha fazla çalışmaya ihtiyaç duyulmaktadır. Ayrıca, enerji tüketimi belirsizliklerini ele alan çalışmalar, lojistik operasyonlarda daha dayanıklı çözümler sağlamıştır (Wang *et al.* 2024). Gelecekteki araştırmalar, bu algoritmaların daha büyük veri kümelerine uygulanmasına ve dinamik rotalama ile filo yönetimi için yeni hibrit yaklaşımların araştırılmasına odaklanmalıdır. Teknoloji ilerlemeye devam ettikçe, araştırmacıların daha sofistike AI tabanlı çözümler geliştirmesi ve bu karmaşık problem için verimli ve ölçeklenebilir yaklaşımlar sunması beklenmektedir (Erdem *et al.* under review a).

Tablo 4. AI Tabanlı Meta-Sezgisel Algoritmalar Üzerine Çalışmaların Analizi (Erdem *et al.* under review a)

Kümeler	Alt-Alanlar	Çalışma	Algoritma	Amaç/Yaklaşım
2, 18, 19, 46	Optimizasyon/Meta-sezgisel/ Sezgisel Yöntemler	Karakatić, 2021	GA	ÇDARP için Genetik Algoritmaların derlemesi
2, 19, 14	Optimizasyon/Meta-sezgisel/ Sezgisel Yöntemler	Chaabani <i>et al.</i> 2018	Ortak Evrimsel Algoritma	İki düzeyli kombinatoryal optimizasyon için ayrıştırma tabanlı yeni yaklaşım
9,10, 46	Lojistik / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Huang <i>et al.</i> 2024	ACO	Tıbbi numune toplama yolu optimizasyonu için uygulandı
14, 19, 49, 27	Rotalama / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Stodola, 2020	Hibrit ACO	Değiştirilmiş optimizasyon kriterleriyle ÇDARP'ye uygulandı.
2, 18, 29	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Quintero-Araujo <i>et al.</i> 2021	Dağılım Tahmini Algoritması	Zaman pencereli araç rotalama için hibrit yaklaşım

Tablo 4. (devamı)

14, 38, 49	Rotalama / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Stodola & Mazal, 2016	ACO	Kapasiteli ÇDARP uygulaması
2, 18, 29	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Su <i>et al.</i> 2024	GA	Zaman pencereli ÇDARP için hafif GA
14, 19, 38	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Stodola & Nohel, 2022	ACO	ÇDARP için düğüm kümelemeli uyarlanabilir algoritma
50	Biyolojik İlhamlı Yöntem	Yesodha & Amudha, 2022	Ateşböceği Algoritması	Zaman pencereli ÇDARP için biyolojik ilhamlı yaklaşım
2, 19, 49	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Xie <i>et al.</i> 2023	GA	Entegre sipariş gruplama ve rotalama optimizasyonu için hibrit GA geliştirildi
3, 50	Sürü Zekâsı / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Yao <i>et al.</i> 2015	Arı Koloni Algoritması	Geliştirilmiş ABC algoritmasıyla teslimat problemlerini optimize eder
2, 35	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Xu & Xiao (2015)	Genetik Kümeleme	Çözüm kalitesini artırmak için genetik kümeleme algoritması önerildi
2, 19, 29	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Prabu <i>et al.</i> 2019	Geliştirilmiş ODV tabanlı GA	ÇDARP çözümü için geliştirilmiş GA varyantı önerildi
19, 29, 35	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Li <i>et al.</i> 2021	Geniş Komşuluk Arama	Çoklu depo İHA sürü rotalaması için hibrit yaklaşım
3, 27	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Kachitvichyanukul <i>et al.</i> 2015	PSO	Çoklu alım ve teslimatlı çoklu depo ARP'yi çözer
9, 27, 46	Lojistik Optimizasyonu / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Gruler <i>et al.</i> 2017	Benzetim-Optimizasyon	Çoklu depo ve olasılıksal atık toplama yönetimini destekler
2, 19, 49	Araç Rotalama / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Wang <i>et al.</i> 2016	GA	Çoklu depo araç rotalamasını optimize etmek için uygunluk ölçekli uyarlanabilir GA kullanır
2, 19, 29	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Wang <i>et al.</i> 2023	Kümeleme Tabanlı GA	Zaman pencereli ve 3D yükleme kısıtlı ÇDARP için GA uygular
2, 33, 44	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Chaabani <i>et al.</i> 2018	Ortak Evrimsel Algoritma	İki düzeyli kombinatorial optimizasyon için yeni ayrıştırma tabanlı yaklaşım

Tablo 4. (devamı)

8, 27, 45	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Kachitvichyanukul <i>et al.</i> 2015	PSO	Çoklu alım ve teslimatlı ARP'yi optimize eder
2, 9, 29	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	El Midaoui <i>et al.</i> 2021	GA	Hastane eczane ilaç teslimat zamanlamasını optimize eder
21, 5	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Du <i>et al.</i> 2017	Bulanık Mantık / İki Düzeyli Programlama	Tehlikeli maddeler için ÇDARP'de kullanılmıştır
3, 18, 50	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Bairathi D & Gopalani D (2021)	MLSSA	MDRP için geliştirilmiş salp sürüsü algoritmasıyla verimli rota planlaması sağlar
2, 18, 19	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Kasparavičiūtė & Ludvigsen, 2023	GA	Enerji ve öncelik kısıtlarıyla AUV görevini optimize eder

RL ve DL yaklaşımları

RL ve DL, ARP çözümünde kritik bir rol oynamaktadır. Bu alandaki önemli bir gelişme, Q-öğrenme gibi RL tekniklerinin ÇDARP'ye uygulanmasıdır. Mak *et al.* (2023), araç rotalamasını optimize etmek amacıyla, güzergâh tahsisi ve kâr paylaşımını aynı anda ele alan derin çoklu ajanlı RL destekli bir koalisyon pazarlık mekanizması önermiştir. Han (2023) ise, AI'nın gerçek zamanlı uygulamalarda rotalamayı optimize etme potansiyelini ortaya koyan, paralel kümeleme tabanlı bir yol planlama algoritması önermiştir. Geleneksel meta-sezgisel yöntemler (örneğin GA ve ACO) kadar yaygın olmasa da, DL teknikleri, karmaşık rotalama senaryolarını optimize etme potansiyelleri nedeniyle ÇDARP araştırmalarında giderek daha fazla ilgi görmeye başlamıştır. Bazı araştırmacılar, ÇDARP kapsamında GA, ACO ve PSO gibi meta-sezgisel algoritmalar dâhil olmak üzere standart AI tekniklerini, büyük çözüm uzaylarını verimli bir şekilde keşfetmek amacıyla yaygın olarak uygulamışlardır. Nahar & Das (2023) ile Alqahtani & Kumar (2024), meta-sezgisel algoritmaların RL ile bütünleştirilmesinin, gerçek zamanlı verilere dayalı uyarlanabilir rota keşfini kolaylaştırarak çözüm kalitesini artırdığını göstermiştir. Ayrıca araştırmacılar, gerçek zamanlı verilerden yararlanan hibrit yaklaşımlar ve optimizasyon süreçlerine ilişkin önemli içgörüler sunmuşlardır. Örneğin, Gu *et al.* (2022), yapay sinir ağlarının teslimat süresi tahminlerini ve kaynak tahsisini önemli ölçüde iyileştirebileceğini ve geleneksel yöntemlere kıyasla rota verimliliğinde %70'e varan bir artış sağladığını göstermiştir. Bu bulgu, ÇDARP'de daha etkili çözümler elde etmek için DL yaklaşımlarının entegrasyonunun artan önemini vurgulamaktadır. Li *et al.* (2024), AGV'lerin (Otonom Güdümlü Araçlar) görev atama ve rotalama problemlerini çözmek için RL tabanlı bir

hiper-heuristik algoritma geliřtirmiřtir. Huo (2021), zaman aralıkları ile ARP problemlerini çözmek için genetik algoritmaları kullanmıřtır. Zhang vd. (2022), teslimat gecikmeleri için cezaların olduđu ARP problemlerini RL ve Transformer tabanlı DL modelleri kullanarak ele almıřtır. Zheng *et al.* (2024), min-max ARP problemlerini nöral çözücüler ve navigasyon algoritmaları kullanarak çözmüřtür. Yazarlar, seçilen makaleleri analiz ederek ÇDARP'ye uygulanan AI tekniklerinin araştırma alanını haritalamıřtır. Bulgular, GA ve ACO gibi meta-sezgisellerin, DL ve RL gibi geliřmiř AI teknikleriyle entegrasyonuna yönelik bir eğilimi vurgulamaktadır. Bu kombinasyonlar, çözüm doğruluğunu artırmayı, hesaplama süresini azaltmayı ve müşteri talebindeki dalgalanmalar ve dinamik trafik kořulları gibi gerçek dünya belirsizliklerini daha iyi ele almayı amaçlamaktadır. Meta-sezgisel algoritmalar, ÇDARP problemlerinde maliyet ve zaman tasarrufu sağlama açısından sıkça kullanılmaktadır. Meta-sezgisel yöntemlerin ötesinde, arařtırmacılar ÇDARP bağlamında karřılařılan kapsamlı ve karmařık veri kümelerini yönetmek amacıyla giderek daha fazla ML yaklařımlarını — özellikle RL ve derin sinir ağı tekniklerini — kullanmaya bařlamıřtır. Shahbazian ve arkadaşları (2024), Araç Rotalama Problemleri (ARP) kapsamında ML uygulamalarını kapsamlı bir řekilde incelemiř; dinamik ve büyük ölçekli problemler için RL tabanlı yöntemlerin verimliliğı artırma ve karmařık kısıtları etkin řekilde ele alma potansiyelini vurgulamıřtır. Bu örnekler, ÇDARP problemlerinde AI'nın giderek artan önemini ve çözüm kalitesini artırma potansiyelini göstermektedir (Erdem *et al.* under review a).

Tablo 5. RL ve DL Uygulamaları için Yapılan Çalıřmaların Analizi (Erdem *et al.* under review a)

Etiketler	Alt-Alanlar	Çalıřma	Algoritma	Amaç/Yaklařım
9, 46	Lojistik Optimizasyonu / Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Serrano-Hernandez <i>et al.</i> 2021	Ajan Tabanlı Simülasyon	Yatay iř birliğı senaryoları kullanarak kentsel alanlar için e-market dağıtım modeli tasarlar
19, 27, 49	Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Shen <i>et al.</i> 2022	Sezgisel Algoritma	Çarpıřma önleyici çoklu depo insansız hava aracı rotaları planlar
42, 15, 29	ML/ Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Bai <i>et al.</i> 2023	ML Analitiğı	Araç rotalama arařtırmaları için analitik ve ML uygulamalarını inceler
1, 17, 43	ML/ Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Feng <i>et al.</i> 2020	ML	Müşteri temsili aktarımıyla araç rotalamayı hızlandırmak için ML uygulamalar
4	RL/ Optimizasyon / Meta-sezgisel / Sezgisel Yöntemler	Yoo <i>et al.</i> 2024	RL	Uyarlanabilir öğrenme ve genetik çeřitliliğı teřvik ederek otonom araç sistemlerinde çoktan çoğa rotalama problemlerini

Optimizasyon teknikleri

ARP çözümündeki karmaşıklıkları ele almak için geleneksel meta-sezgisel yöntemlerden sofistike hibrit ve AI tabanlı yaklaşımlara kadar geniş bir yelpazede optimizasyon teknikleri geliştirilmiş ve uygulanmıştır. Her bir teknik, lojistik ve operasyonel süreçleri iyileştirmek için etkili ve verimli çözümler sunarak alana önemli katkılarda bulunmuştur. Aşağıda, bu tekniklerin uygulamalarına ilişkin bir inceleme, ARP alanında sağlanan ilerlemeleri ve iyileştirmeleri öne çıkarmaktadır.

Mańdziuk (2018), ARP'nin yenilikçi formülasyonlarını ve hesaplama zekasına dayalı çözüm metodolojilerini inceleyerek, ARP'nin karmaşıklıklarına ilişkin temel bir anlayış sunmuştur. Bu temel bilgilere dayanarak, Darvish *et al.* (2019), iki aşamalı esnek konum-rotalama problemlerini ele almış ve hibrit optimizasyon tekniklerini kullanarak lojistik süreçleri iyileştirmiştir. Benzer şekilde, Yang *et al.* (2020), bölünmüş teslimat ARP'leri sırasında mal tüketimini modellemiş ve bu lojistik süreçleri optimize ederek verimliliği artırmıştır. Jiansheng *et al.* (2020), çoklu depo senaryolarında heterojen ARP'leri ele almak için hibrit optimizasyon tekniklerini uygulamış ve hibrit yöntemlerin heterojen bağlamlardaki karmaşık rotalama sorunlarını çözmedeki etkinliğini göstermiştir. Ayrıca, Fagbola & Thakur (2019), hibrit meta-sezgisel yaklaşımların ARP'leri çözmede uyarlanabilirliğini ve etkinliğini vurgulamış, bu kombinasyon metodolojilerinin çok yönlülüğünü öne çıkarmıştır. Žunić *et al.* (2020), veri merkezli uyarlanabilir bir yaklaşım önererek gerçek dünya ARP zorluklarını ele almış ve lojistik verimliliğini önemli ölçüde artırmıştır. Benzer şekilde, Belhaiza *et al.* (2019), birden çok zaman penceresini içeren üç evrimsel sezgisel yöntem sunarak ARP'ler için hızlı ve etkili çözümler üretmiştir. Araştırmacılar ayrıca, karmaşık ARP sorunlarını çözmek için AI tabanlı tekniklerin uyarlanabilirliğini ve çok yönlülüğünü vurgulayan veri odaklı evrimsel sezgisel algoritmalar geliştirmiştir. Agárdi *et al.* (2019), zaman pencerelerini entegre ederek çoklu depo periyodik ARP'leri optimize etmiş ve karmaşık zamanlama zorluklarına pratik bir çözüm sunmuştur. Yan *et al.* (2019), iki aşamalı ARP'ler için evrimsel algoritmalarla birleştirilen grafik tabanlı Bulanık Mantık kullanarak gelişmiş karar verme süreçlerini içeren çözümler geliştirmiştir. Benzer şekilde, Oudouar *et al.* (2020), AI ile geleneksel optimizasyonun çekici bir kombinasyonunu sergileyerek, kapasiteli konum-rotalama problemlerini çözmek için sınır ağlarını sezgisel yöntemlerle birleştirmiştir. Gerçek dünya zorlukları bağlamında, Jafarian *et al.* (2019), araç arızalarını içeren envanter-rotalama problemlerini ele almış ve COVID-19 pandemisinin getirdiği kısıtlamalar altında lojistik sistemlerin dayanıklılığını ve verimliliğini artırmak için hem envanter yönetimini hem de araç operasyonlarını optimize etmiştir.

İncelenen çalışmalar sonucunda ÇDARP'deki araştırma sorularının her birinin yanıtı aşağıdaki gibidir (Erdem *et al.* under review a).

RQ1: ÇDARP Araştırmalarının Mevcut Durumu Nedir?

Son yıllardaki ÇDARP çalışmaları, daha karmaşık ve gerçekçi lojistik zorlukları ele almak için klasik modellerin genişletilmesine odaklanmıştır. Hibrit yaklaşımlar, klasik optimizasyon tekniklerini gelişmiş meta-sezgiseller ve AI yöntemleriyle birleştirerek önem kazanmıştır.

- ✓ Örneğin, Tabu Arama (TA), PSO ve ACO gibi meta-sezgisellerin entegrasyonu, dinamik ortamlarındaki ÇDARP problemlerini etkili bir şekilde ele almak için vurgulanmıştır (Wang *et al.* 2016).
- ✓ Bu yaklaşımlar, büyük ölçekli lojistik uygulamalarda, örneğin ilaç dağıtımı ve sağlık lojistiği gibi alanlarda uygulanmış ve önemli maliyet ve zaman tasarrufu sağlamıştır.
- ✓ Ayrıca, evrimsel algoritmalar ve global arama teknikleri, çözüm alanını önemli ölçüde genişleterek büyük ölçekli ÇDARP problemlerini optimize etmek için araştırılmıştır.
- ✓ Çevresel etkilerin en aza indirilmesine yönelik artan bir ilgi, yakıt tüketimine ve karbon ayak izini azaltmaya odaklanan yeşil ÇDARP yaklaşımlarının geliştirilmesine yol açmıştır (Li *et al.* 2019).

RQ2: Meta-sezgisel Teknikler ÇDARP'yi Nasıl Etkiler?

Meta-sezgisel teknikler, büyük ölçekli ÇDARP gibi karmaşık problemler için sağlam, esnek ve ölçeklenebilir çözümler sunarak bu alanda önemli bir etki yaratmıştır.

- ✓ Optimizasyon: ACO ve PSO gibi teknikler, geniş çözüm alanlarını hızlı bir şekilde keşfederek büyük ölçekli ÇDARP örneklerini etkili bir şekilde ele alır, hesaplama süresini azaltır ve çözüm kalitesini korur (Soeanu *et al.* 2020).
- ✓ Gerçek Dünya Kısıtlarını Ele Alma: Meta-sezgiseller, zaman pencereleri, araç kapasiteleri ve değişken müşteri talepleri gibi gerçek dünya kısıtlarını kolayca entegre edebilir.
- ✓ Maliyet ve Zaman Verimliliği: Çalışmalar, meta-sezgisellerin yakıt tüketimini optimize ederek, rota mesafelerini kısaltarak ve seyahat sürelerini azaltarak operasyonel maliyetleri önemli ölçüde düşürdüğünü göstermektedir.
- ✓ Hibrit Yaklaşımlar: ACO gibi klasik algoritmaların entegrasyonu, çözüm kalitesini artırmıştır.

- ✓ Çevresel Sürdürülebilirlik: GAs ve ACO gibi meta-sezgiseller, sürdürülebilir lojistik yönetimini ilerletmek için rota mesafelerini optimize ederek yakıt tüketimini ve CO₂ emisyonlarını azaltmak için kullanılmıştır (Li *et al.* 2019).

RQ3: ÇDARP Araştırmalarında AI Yöntemleri Hangi Alan ve Alt Alanlarda Kullanılmaktadır?

AI yöntemleri, ÇDARP araştırmalarında çeşitli alanlarda uygulanmaktadır:

- ✓ Dinamik ÇDARP: ML, gerçek zamanlı rotalama için kullanılarak müşteri talepleri ve yol koşullarındaki değişimlere anında uyum sağlamaktadır.
- ✓ Yeşil ÇDARP: AI, yakıt tüketimini ve karbon emisyonlarını en aza indiren sürdürülebilir lojistik modellerini optimize etmek için kullanılmaktadır (Li *et al.* 2019).
- ✓ Sağlık ve Acil Durum Lojistiği: Hasta bekleme sürelerini ve seyahat mesafelerini minimize ederek sağlık lojistiğinde kritik bir rol oynamaktadır.

RQ4: AI Teknikleri ÇDARP'yi Nasıl Etkiler?

AI teknikleri, ÇDARP çözüm yöntemlerini önemli ölçüde etkileyerek hem verimliliği hem de esnekliği artırmıştır.

- ✓ ACO, araç kapasiteleri göz önüne alınarak rota mesafelerini en aza indirmek için ÇDARP'ye uyarlanmıştır.
- ✓ Hibrit AI yöntemleri, özellikle karmaşık çoklu depo ortamlarında dinamik rota planlamayı iyileştirmiştir (Wang *et al.* 2023; Wu & Gao, 2023).

RQ5: ÇDARP Çalışmalarında Hangi AI Yöntemleri Kullanılmaktadır?

ÇDARP çalışmalarında yaygın olarak kullanılan AI yöntemleri:

- ✓ GAs: Dinamik ve yeşil ÇDARP ayarlarında rotaları optimize etmek için yaygın olarak kullanılmaktadır.
- ✓ PSO: Müşteri talebi veya araç erişilebilirliğindeki değişimlere hızlı uyum gerektiren ÇDARP problemleri için etkilidir.
- ✓ ACO: Karmaşık, büyük ölçekli rotalama problemlerinde ve çevresel etkileri en aza indirmede yaygın olarak uygulanmaktadır (Li *et al.* 2019; Stodola P & Nohel, 2022; Huang *et al.* 2024).
- ✓ Sinir Ağları ve DL: Trafik desenlerini tahmin etmek ve araç yüklemeyi optimize etmek için kullanılmaktadır.

- ✓ Hibrit Algoritmalar: Klasik yöntemlerle AI tekniklerini birleştirerek büyük, karmaşık ÇDARP örneklerini etkili bir şekilde çözmektedir (Jabir *et al.* 2017).

RQ6: ÇDARP için En Yaygın AI Teknikleri Nelerdir?

- ✓ GAs: Geniş çözüm alanlarını etkili bir şekilde keşfederek büyük ölçekli rotalama problemlerinde yakın optimal çözümler sağlamaktadır.
- ✓ ACO: Yeşil ARP için yakıt tüketimini ve CO₂ emisyonlarını azaltmada başarılı bir şekilde uygulanmıştır (Li *et al.* 2019).
- ✓ PSO: Dinamik ÇDARP problemlerinde rota planlamayı iyileştirmiştir.
- ✓ DL ve RL: Gerçek zamanlı dinamik rotalama için AI tabanlı modellerin gelişimini desteklemektedir (Pan & Liu, 2023).

Sonuç olarak, AI tekniklerinin ÇDARP'ye uygulanması, 2015-2024 yılları arasında önemli ilerlemeler kaydetmiştir. Bu durum, bu optimizasyon probleminin karmaşıklığı ve NP-zor doğasının artan bir şekilde tanınmasını yansıtmaktadır. ÇDARP, birden fazla depodan hareket eden araç filosunun, bir dizi müşteriye en verimli rotalarla hizmet sunmasını içeren karmaşık bir optimizasyon problemidir ve özellikle heuristic ve meta-sezgisel algoritmaların entegrasyonundan büyük ölçüde yararlanmıştır. Bu yöntemler, özellikle dinamik ve belirsiz ortamlarda çözüm etkinliğini ve verimliliğini artırmada etkili olmuştur. ACO ve IWD gibi çeşitli sezgisel algoritmalar geliştirilmiş ve test edilerek büyük ölçekli ARP'leri başarılı bir şekilde ele almıştır. Ayrıca, AI uygulamalarının evrimi, yalnızca bağımsız sezgisel yöntemlerle sınırlı kalmayıp, birden fazla algoritmayı entegre eden hibrit yaklaşımları da içerecek şekilde genişlemiştir. Bu hibrit yaklaşımlar, performans açısından önemli iyileştirmeler göstermiştir. Bu sistematik inceleme, AI tekniklerinin ÇDARP alanını dönüştürdüğünü, geleneksel yöntemlerden daha verimli ve ölçeklenebilir çözümler sunduğunu vurgulamaktadır. RL ve hibrit optimizasyon teknikleri gibi gelişmiş algoritmaların entegrasyonu, çözüm kalitesinde ve hesaplama verimliliğinde dikkat çekici iyileştirmelere yol açmıştır. Örneğin, kümeleme teknikleri ile GA ve diğer meta-sezgisel yaklaşımların birleştirilmesi, müşteri atamaları ve genel rotalama süreçlerinin optimize edilmesinde etkili olduklarını göstermiştir. Acil durum lojistiği ve salgın yönetimi gibi örnekler, AI tabanlı çözümlerin operasyonel verimliliği önemli ölçüde artırabileceğini göstermiştir. Stokastik unsurların ve çok amaçlı optimizasyon yaklaşımlarının araştırılması, talep ve kapasite belirsizliği gibi gerçek dünya karmaşıklıklarını ele almak için aynı anda birden fazla kriteri işleyebilen algoritmaların önemini ortaya koymaktadır. Ayrıca, ÇDARP problemini çözerken genel olarak kullanılan girdi verileri; müşteri talepleri ve lokasyonları, depo lokasyonları ve kapasiteleri, araç kapasiteleri, zaman penceresi kısıtlamaları ve internet tabanlı harita verileridir.

Meta-sezgiseller (GA, ACO, PSO) ve hibrit teknikler, ÇDARP çözümlerinde gerçek dünya lojistik problemleri için ölçeklenebilir, uyarlanabilir ve maliyet açısından etkili çözümler sağlamıştır. Bu yaklaşımlar, karmaşık kısıtlamalar ve dinamik ortamlarla başa çıkmada kritik bir rol oynamaktadır. Ancak, artan problem boyutu ve karmaşıklıkla birlikte ölçeklenebilir algoritmalar geliştirilmesi, araştırmacılar için bir zorluk oluşturmaktadır. AI tekniklerinin ÇDARP'ye entegrasyonu, lojistik verimliliğini artırmak, operasyonel maliyetleri minimize etmek ve sürdürülebilirliği teşvik etmek için önemli bir potansiyele sahiptir. Ayrıca, DL ve RL'nin uygulanması, özellikle gerçek dünya ve büyük ölçekli problem örneklerine adaptasyonu artırma açısından önemli bir potansiyele sahiptir. Algoritmaların daha büyük veri kümelerinde performansının kapsamlı bir şekilde test edilmesi, sağlamlık ve verimliliklerini garanti altına almak için önemlidir. Akademi ve endüstri arasındaki iş birliği, mevcut modellerin pratik sınırlamalarını ele almak ve gerçek dünya lojistik sistemlerinde etkili ve uygulanabilir AI tabanlı çözümleri ilerletmek için kritik bir rol oynamaktadır (Erdem *et al.* under review a).

Yukarıda belirtilen literatürdeki boşlukları gidermek amacıyla, tezin ilk iki aşamasında gerçekleştirilen çalışmalar, GPS verilerinin yapılandırılmamış görüntü verisi formatına dönüştürülmesi ve bu verilerin DL modelleri üzerindeki etkisinin değerlendirilmesi üzerine odaklanmıştır. Bu çalışmalar, ölçeklenebilir ve uyarlanabilir çözümler geliştirilmesine katkıda bulunmayı hedeflemiştir. Gerçek dünya lojistik problemlerine uygulanabilirliği sağlamak için hem anonim veri setleri hem de şirketten temin edilen gerçek GPS verileri kullanılmıştır. Bu yaklaşım, literatürde sıklıkla vurgulanan büyük ölçekli veri kümelerinde algoritmaların performansını test etme ihtiyacına yanıt vermiş ve AI tabanlı yöntemlerin etkili bir şekilde uygulanması için sağlam bir zemin oluşturmuştur. Böylece, meta-sezgisel ve hibrit algoritmaların gerçek dünya problemlerindeki sınırlamalarını aşmaya yönelik pratik ve düşük maliyetli çözümler geliştirilmiştir. İlk iki aşamada yapılan somut kazanımlar ayrıntılı incelendiğinde:

Tezin başlangıç safhasında literatüre yapılan katkılar (Aydin & Erdem, 2023):

- ✓ GPS konumlandırma sistemlerinin doğruluğunu artırmak için ML ve DL tabanlı hibrit ve topluluk algoritmalarını kullanarak yenilikçi yaklaşımlar sunuldu. Önerilen modeller, GPS verilerinden elde edilen enlem ve boylam bilgilerini daha doğru hale getirerek, mevcut yöntemlere kıyasla daha hassas ve güvenilir konumlandırma sonuçları vermektedir.
- ✓ GPS verilerindeki hataları azaltmaya yönelik analiz ve işleme süreçlerinde uygulanan algoritmaların performansını iki farklı veri setinde test ederek, hibrit ve topluluk modellerinin tekil modellere göre üstün olduğu ortaya konuldu. Bu

bağlamda, çalışma hem teorik hem de uygulamalı açıdan GPS doğrulama sistemlerinin geliştirilmesine yönelik yeni bir perspektif sunuldu.

Tezin ikinci safhasında, literatüre yapılan katkılar (Erdem *et al.* under review e):

- ✓ Büyük Ölçekli GPS Verisi Analizi: Çalışma, İstanbul'daki bir şirketin GPS verileri üzerinde yapılmış olmasıyla, geniş ölçekli veri analizi konusunda literatüre önemli bir katkı sunmaktadır.
- ✓ ML ve DL Modellerinin Karşılaştırılması: Çalışma, topluluk ve hibrit modellerin, ayrik modellere kıyasla üstün performans sağladığını göstermiştir.
- ✓ Gerçek Dünya Sorunlarına Çözüm: Literatürdeki birçok çalışmanın aksine, gerçek dünyadaki lojistik operasyonlarına yönelik çözüm önerileri geliştirilmiştir.

İlk iki çalışmadan elde edilen bulgular, GPS verilerinin işlenmesi ve konumlandırma doğruluğunu artırma alanında literatüre yeni bir bakış açısı kazandırmıştır. Özellikle, yapılandırılmamış veri dönüşümlerinin potansiyeli ve sınırlamaları hakkında önemli çıkarımlar yapılmıştır.

Çalışmanın sonraki aşamalarında, GPS verilerinin lojistik operasyonlarda daha etkin kullanılması ve ÇDARP gibi karmaşık problemlerin çözümüne yönelik yeni yöntemler geliştirilmesi hedeflenmiştir. Bu doğrultuda, üçüncü, dördüncü, beşinci ve altıncı aşamalar aşağıdaki şekilde gerçekleştirilmiştir:

Üçüncü aşamada, İstanbul'daki bir içecek dağıtım şirketinin gerçek operasyonel verileri (Mobiliz araç takip sistemiyle kaydedilen GPS verileri ve Micro V15 sisteminden elde edilen tarihsel veri kümeleri) kullanılarak ölçeklenebilirlik ve uyarlanabilirlik gibi literatürde belirtilen temel zorlukların üstesinden gelinmeye çalışılmıştır. Gerçek zamanlı dinamik zaman-mesafe matrisleri, geniş bir tarihsel veri kümesinden türetilmiş ve rota optimizasyon süreçlerinde kullanılmıştır. Bu durum, literatürde sıklıkla vurgulanan teorik yaklaşımları pratik uygulamalara taşımıştır. MBT algoritması sayesinde daha kısa rotalar ve maliyet tasarrufları sağlanmıştır. Ayrıca, geliştirilen modelin performansı, hem eğitim hem de test veri kümeleri üzerinde kapsamlı bir şekilde değerlendirilerek, gerçek dünya koşullarındaki uygulanabilirliği doğrulanmıştır.

Tezin üçüncü safhasında literatüre yapılan somut kazanımlar ayrıntılı incelendiğinde (Erdem *et al.* under review b):

- ✓ Literatürde, ARP çözümleri genellikle küçük ölçekli senaryolar ve GPS verilerine dayalı analizlerle sınırlı kalmıştır. Ancak bu çalışma, büyük ölçekli GPS verileri ile şirket kaynaklı Mikro V15 kullanarak gerçek dünya problemleri için yeni bir çözüm

yaklaşımı sunmaktadır. Bu durum, literatürde ilk kez bu kapsamda bir çözüm önerisi sağlayarak önemli bir yenilik ortaya koymaktadır.

- ✓ Önceki çalışmaların çoğunda ARP çözümleri, AI yöntemleriyle hazır çözümleyici mantığına indirgeyen çalışmalar gerçekleştirilmiştir (Ara *et al.* 2023; Máximo *et al.* 2024). Ancak bu çalışmada, problemin ölçeği ve hızlı çözüm gereklilikleri göz önünde bulundurularak AI tabanlı meta-sezgisel algoritmalar tercih edilmiştir. Ayrıca, yeni bir algoritma olan MBT önerilmiştir. Bu çalışmada kullanılan meta-sezgisel algoritmalar, titizlikle seçilmiş ve üç farklı yöntemle başlatılmıştır: Depolara yakın dağıtım, rastgele dağıtım ve ML ile oluşturulan başlangıç çözümüdür. Algoritmaların bu şekilde özenle başlatılması, araştırma metodolojisinin dikkat çeken bir yönüdür.
- ✓ OSRM API ve Google Maps gibi araçlar genellikle "araba, uçak, yaya, bisiklet" türü taşıtlar için rota planlaması sağlarken, bu çalışmada kamyon veya tır gibi büyük araçların şehir içi bölgelerde karşılaşılabileceği erişim zorlukları dikkate alınmıştır. Böylece daha hassas rota planlaması ve trafik analizi yapılmış; bu yenilikçi yaklaşım, araç türüne özgü özelleştirmeleri mümkün kılmıştır. Ayrıca, Google Maps gibi platformlarda trafik ve hava koşullarına bağlı değişkenlikler nedeniyle zaman bilgisi değişebilirken, bu çalışma sadece gerçek zamanlı verilerle sınırlı kalmamış, tarihsel GPS verilerini içeren kapsamlı bir yöntem benimsemiştir. Bu yaklaşım, araştırmaya hem derinlik hem de özgünlük kazandırmıştır.
- ✓ GPS verilerine dayalı olarak AI tabanlı sezgisel algoritmalar geliştirilmiştir. Bu algoritmalar, küresel bilgi yakalama kapasitesini artırmış ve OSRM API kullanılarak gerçekleştirilen meta-sezgisel çözümlerle kıyaslandığında daha başarılı sonuçlar elde etmiştir. Çalışmada, araç sevkiyat planlamasında, belirli bir coğrafi bölgede gerçekleştirilmesi gereken "hedef operasyonlar" sağlandığında detaylı bir görev planı oluşturulmuştur. Bu plan; izlenecek rotaları, ziyaret edilecek yerlerin sırasını ve taşınacak ürünleri kapsamaktadır. Önerilen çözüm, bu çalışmada etkili sonuçlar sağlamış ve gelecekte akıllı şehirlerde otonom araç rotalarının veya araç paylaşım hizmetlerinin optimize edilmesi gibi farklı uygulamalarda kullanılabilir olduğunu göstermiştir.
- ✓ Çalışmada dağıtım ağı, iki farklı seviyede yapılandırılmıştır (Şekil 4):
- ✓ Adım 1: Mobiliz araç takip sistemi tarafından sağlanan tarihsel GPS verileri kullanılarak bir zaman-mesafe matrisi oluşturulmuş ve bu matris üzerinde meta-sezgisel algoritmalarla rota planlaması yapılmıştır. Ayrıca OSRM API ile zaman-mesafe matrisi oluşturulmuş ve benzer rota planlama işlemleri gerçekleştirilmiştir.

- ✓ Adım 2: GPS verileri ile oluşturulan zaman-mesafe matrisine AI entegrasyonu sağlanmıştır. Bu kapsamda, tarihsel Micro V15 veri tabanından 2013-2023 yıllarına ait veriler ile 2020-2023 dönemine ait GPS verileri birleştirilmiştir. Bu entegrasyonla oluşturulan başlangıç çözümü, meta-sezgisel algoritmaların başlangıç noktası olarak kullanılmıştır.

Dördüncü aşamada, literatürde sıkça dile getirilen eksikliklerin giderilmesi amacıyla, daha önce tanımlanan veri setleri (Mobiliz ve Micro V15) kullanılarak farklı algoritmaların performansları karşılaştırılmıştır. Bu çalışmada, özellikle BKA, DQN, Dikkat Mekanizması ve DTBKA gibi dört farklı algoritma test edilmiştir. DTBKA modeli, meta-sezgisel ve DL yöntemlerini bir araya getirerek diğer yöntemlere kıyasla üstün performans sergilemiştir. Modelin, lojistik operasyonlarda maliyetleri azaltma ve verimliliği artırma potansiyeli, gerçek dünya veri setleriyle yapılan değerlendirmelerde ortaya konulmuştur. Bu aşama, literatürde genellikle simülasyon verilerine dayanan çalışmaları aşarak işlevsel uygulanabilirlik açısından önemli bir katkı sunmuştur.

Tezin dördüncü safhasında literatüre yapılan somut kazanım ayrıntılı incelendiğinde (Erdem *et al.* under review d):

- ✓ Meta-sezgisel ve DL yöntemlerini entegre ederek yenilikçi bir DTBKA modeli sunulmuştur. yalnızca çözüm performansı değil, genel model esnekliği, ölçeklenebilirliği ve öğrenme kapasitesi açısından da literatüre katkı sağlanmıştır.
- ✓ MAE, MSE gibi metriklerle tüm modellerin genelleme yetenekleri değerlendirilmiştir. Özellikle DTBKA ve dikkat mekanizmasının eğitim-test performans dengesinde istatistiksel olarak anlamlı üstünlük sağladığı gösterilmiştir.
- ✓ DTBKA modeli, meta-sezgisellerin keşif gücü ile dikkat mekanizmasının öğrenme yeteneğini birleştirerek, rota optimizasyonunda hem hata oranı hem de rota uzunluğu açısından en iyi sonuçları vermiştir. Literatürde bu düzeyde bir hibrit yapı, gerçek GPS verileriyle başarılı şekilde test edilmiştir.

Dördüncü aşamada geliştirilen algoritmalar, özellikle mekânsal verilerin işlenmesinde başarılı çözümler sunmuş olsa da, zaman serisi bileşenlerinin modelleme kapasitesi sınırlı kalmıştır. Bu nedenle, beşinci aşamada hem zamansal hem de mekânsal ilişkileri aynı anda bütüncül şekilde öğrenebilen yapılar geliştirilmeye odaklanılmıştır.

Tezin beşinci safhasında literatüre yapılan somut kazanım ayrıntılı incelendiğinde (Erdem *et al.* under review d):

- ✓ GNN tabanlı modellerin gerçek zamanlı lojistik verilerde uygulanabilirliği, bu tez kapsamında test edilmiş ve GNN-RL modelinin istikrarlı bir genel performansa sahip olduğu gösterilmiştir. GNN-RL modeli, GNN yapısını Yakınsak Politika Optimizasyon (PPO) ile birleştirerek yüksek ödül ve dengeli rota üretimi ile ön plana çıkmıştır.
- ✓ Bulanık Mantık ile evrimsel algoritmaların birleştirildiği BME modeli, belirsiz talep ve kapasite koşullarında rota planlamasının nasıl geliştirilebileceğini göstermiştir. Bu yaklaşım, özellikle kısıtların belirsiz olduğu durumlarda literatürde az çalışılan bir boşluğu doldurmuştur. Literatürde birçok evrimsel algoritma veya Bulanık Mantık uygulaması (bulanık kurallar vb. (EL YADARI *et al.* 2024)) yer alsa da bu iki yöntemin eş zamanlı (Zaman-mekân eşzamanlı öğrenme (GNN-LSTM-RL)) ve optimize edici bir yapı içinde birleşimi nadirdir.
- ✓ Literatürdeki birçok çalışma (Tsehaie, 2023; Bai *et al.* 2023; Jin *et al.* 2024) yalnızca mekânsal yapıyı (GNN tabanlı) ya da zamansal örüntüleri (LSTM) modellemeye odaklanmıştır. Ancak bu aşamada, GNN-LSTM-RL modeli ile hem müşteri-depo bağlantılarının topolojisi hem de siparişlerin zaman serisi dinamikleri aynı çatı altında bütünleştirilmiştir.
- ✓ K-Fold, MAE/MSE dağılımı, ödül eğrileri gibi ileri düzey analizler ile özellikle Sinirsel Grafik Öğrenmesi (NGL)-RL vb. modellerinin genelleme başarımı detaylı incelenmiş, bu da literatürde az görülen çok yönlü validasyon katkısı sunmuştur.

Beşinci aşamadaki bütünleşik modeller, zamansal ve mekânsal ilişkileri aynı yapı altında modelleme konusunda önemli başarılar sunarken, donanımsal uyumluluk, çevresel adaptasyon yetisi ve operasyonel esneklik açısından hâlâ sınırlılıklar taşımaktadır. Bu eksiklikleri aşmak amacıyla altıncı aşamada, DRL, meta-öğrenme ve evrimsel yaklaşımların HLS uyumlu mimarilerle bütünleştirildiği ileri seviye AI tabanlı modeller geliştirilmiştir.

Tezin altıncı safhasında literatüre yapılan somut kazanım ayrıntılı incelendiğinde (Erdem *et al.* under review c):

- ✓ Çok yönlü hibrit AI modelleri geliştirilmiştir. Literatürde çoğu çalışma yalnızca tek bir algoritma türüne (örneğin sadece Transformer veya sadece DRL) odaklanırken (Wang, & Chen, 2022; Munir *et al.* 2024; Wu *et al.* 2024), bu tezde yedi farklı hibrit model (AST-Former, ALT-Former, LST-Former, AST-MetaDRL, ALT-MetaDRL, LST-MetaDRL, NRL) geliştirilmiş ve karşılaştırılmıştır. Bu modellerin her biri, farklı zaman önceliği stratejileri (ASAP, ALAP, Listeleme) ve öğrenme türlerini

(DRL, MAML, NEAT) bütünleştirerek zaman-mekân-öğrenme boyutlarını aynı anda optimize eden birleşik yapılar sunmuştur.

- ✓ Transformer tabanlı modellerle yüksek doğruluk ve genelleme kapasitesi sağlanmıştır. LST-Former modeli, gerçek dünya senaryolarında rota doğruluğu, yükleme verimliliği ve genelleme açısından en üstün performansı göstermiştir. Bu, literatürde Transformer tabanlı modellerin genellikle simülasyon verileriyle test edildiği çalışmalara kıyasla, gerçek GPS verisi kullanımıyla yapılan ilk kapsamlı doğrulamalardan biridir.
- ✓ DRL-MAML entegrasyonu ile uyarlanabilirlik artırılmıştır. DRL modelleri ile MAML stratejilerinin birleştirilmesi, farklı dağıtım bölgelerine veya değişen taleplere hızlı adaptasyon sağlayan çözümler geliştirilmesini mümkün kılmıştır. Bu da, dinamik ortam koşullarında stratejik karar verme kapasitesini ciddi şekilde artırmış ve özellikle değişken müşteri taleplerinde literatürde eksik kalan adaptif çözümler sunmuştur.
- ✓ NEAT Tabanlı Yapılarla Sinir Ağı Mimarisinin Evrimsel Optimizasyonu gerçekleştirilmiştir. NRL modeli, sinir ağı yapısını veri karmaşıklığına göre evrimsel olarak optimize ederek hafif yapılı ve genellenebilir çözümler sunmuştur. Bu da, düşük kaynaklı donanımlarda dahi uygulanabilecek modeller için önemli bir katkıdır.
- ✓ HLS uyumluluğunun konsept entegrasyonu sağlanmıştır. Literatürde HLS, genellikle donanım hızlandırması için kullanılırken; bu tezde HLS uyumlu mimarilerle, AI tabanlı optimizasyonun donanıma uyarlanabilirliği ilk kez konsept düzeyde ÇDARP alanında ele alınmıştır. Özellikle LST-Former gibi modellerin gelecekte FPGA gibi donanımlara gömülerek gerçek zamanlı çalıştırılmasına zemin hazırlanmıştır.
- ✓ Performans sadece klasik hata metrikleriyle değil (MSE, RMSE, MAPE), aynı zamanda DTW, PAP10, POFP, Kapsam gibi operasyonel metriklerle de analiz edilmiştir. Bu çeşitlilik, literatürde nadir rastlanan çok boyutlu değerlendirme yaklaşımlarına önemli bir katkı sağlamaktadır.

Bu yenilikçi yaklaşımlarda, ARP çözümünde GPS verilerinin kullanımı algoritmaların performansını doğrudan etkilemiştir. Çünkü önceki araştırmalarda trafik sıklık ve yol koşullarını zaman bağımlı seyahat sürelerini kullanarak ele alan modeller geliştirilmiştir (Mousavipour & Hojjati, 2024; Kritzinger *et al.* 2012). Buna karşılık, tez kapsamında, trafik ve hava durumu gibi dinamik değişkenleri yakalamak için tarihsel GPS verileri doğrudan

kullanılmaktadır. Önceki çalışmalar, heterojen ARP için hibrit algoritmalar ve meta-sezgiseller uygulamıştır (Kocatürk *et al.* 2021; Zahedi *et al.* 2023). Bu çalışmada ise kamyonlar ve tırlar için araç bazında özelleştirmeler yapılarak daha doğru rota planlaması sağlanmaktadır. Bu bağlamda önerilen yaklaşımlar ile ÇDARP'deki esneklik, ölçeklenebilirlik, pratik uygulamalar vb. mevcut kısıtlamalar aşılarak literatürdeki önemli bir boşluk doldurulmuştur. İlave olarak, lojistik sektörüne de önemli bir katkı sunulmuştur. Gerçek dünya verileri ile simülasyonlardan ayrılan derinlemesine değerlendirme yapılmıştır. Diğer çalışmaların aksine (Mousavipour & Hojjati, 2014; Kritzing *et al.* 2012; Kocatürk *et al.* 2021; Zahedi *et al.* 2023), bu tezde modeller sadece teorik ya da sentetik verilerle değil, İstanbul merkezli bir dağıtım şirketinden alınan gerçek zamanlı GPS ve müşteri sipariş verileri ile test edilmiştir. Bu da, önerilen modellerin uygulanabilirlik ve ölçeklenebilirlik açısından literatürdeki çoğu çalışmadan ileri olduğunu göstermektedir. Ayrıca, önerilen modellerin çoklu depo, çoklu sefer ve parçalı teslimat senaryolarına uyarlanabilir yapısı, şehir içi dağıtım süreçlerinin dinamik doğasına doğrudan uyum sağlamıştır. Özellikle şehir içi teslimatlarda karşılaşılan trafik yoğunluğu, teslimat zaman kısıtları ve talep belirsizlikleri gibi değişkenler, tarihsel GPS verilerinin entegrasyonu ve AI tabanlı öğrenme yapıları sayesinde daha etkin biçimde yönetilebilmiştir. Böylece, bu tez çalışması, lojistik kapsamda rota doğruluğunu artıran, yükleme verimliliğini iyileştiren ve gerçek zamanlı koşullara duyarlı karar destek mekanizmaları sunarak, şehir içi teslimatlarda hem operasyonel hem de stratejik düzeyde uygulanabilir çözümler geliştirmiştir.

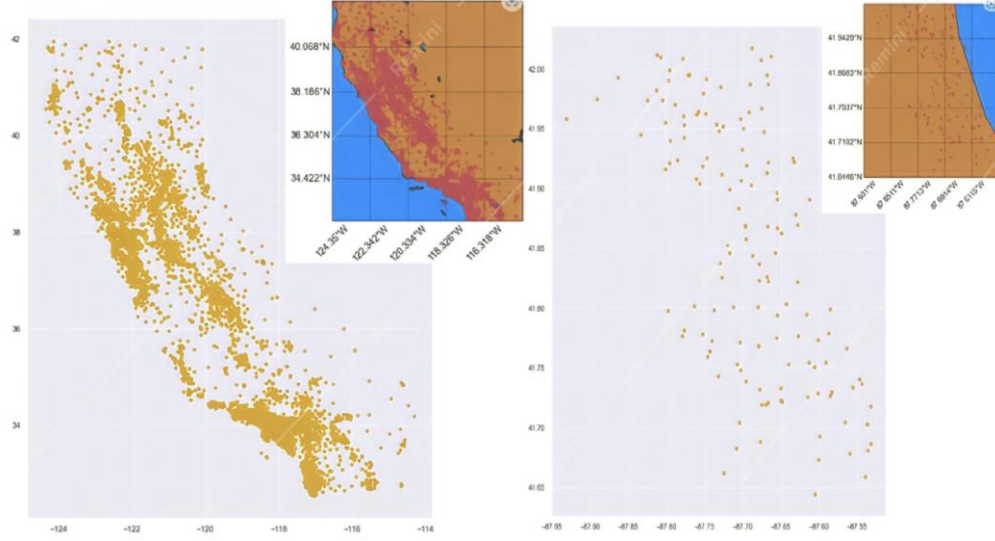
MATERYAL VE METOT

Materyal

Tezin başlangıç safhasında, iki veri seti kullanılmıştır ve bu veri setlerinin özellikleri Tablo 6’da sunulmaktadır. CHD veri setinde (2022), 17,000 adet örnek bulunmaktadır ve bu veri seti, Kaliforniya’daki konutlarla ilgili coğrafi ve demografik bilgiler (konut bilgileri, enlem, boylam, yaş, nüfus vb.) içermektedir. WnvD veri seti (2022), 2007-2013 yılları arasında Chicago’da toplanmış 10,506 örnek içermekte olup, sivrisineklerin tür bilgisi ve hastalık durumu (sivrisineklerin konum bilgileri, enlem, boylam, vb.) detaylarını kapsamaktadır. Şekil 14, CHD ve WnvD veri setlerindeki koordinatlar ve coğrafi konumlarla ilgili bilgiler görselleştirilmiştir (Aydin & Erdem, 2023).

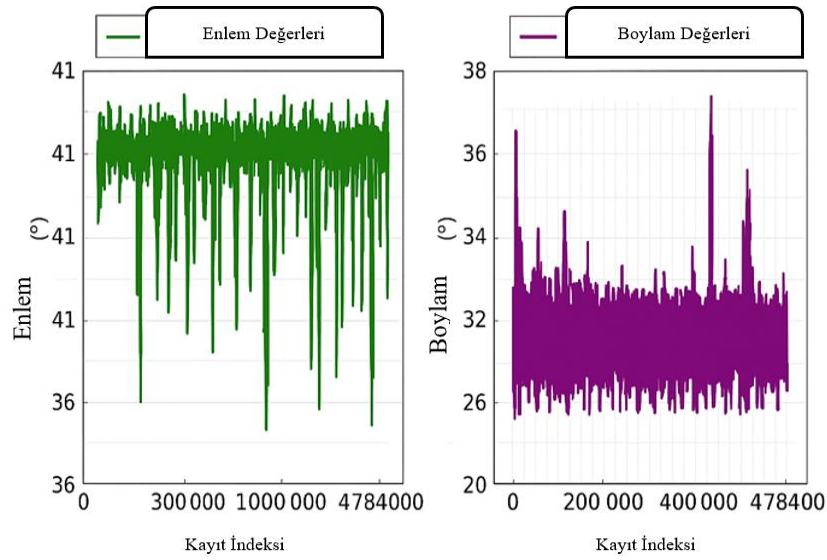
Tablo 6. Veri Setlerinde Mevcut Ayrıntılı Bilgiler (Aydin & Erdem, 2023)

Veri Seti	Öznitelikler	Veri Tipi
CHD	Boylam	decimal
CHD	Enlem	decimal
CHD	Konutların Medyan Yaşı	float
CHD	Toplam Oda Sayısı	float
CHD	Toplam Yatak Odası Sayısı	float
CHD	Nüfus	float
CHD	Hane Sayısı	float
CHD	Medyan Gelir	float
CHD	Medyan Konut Değeri	float
WnvD	Tarih	date
WnvD	Adres	string
WnvD	Tür	string
WnvD	Blok	int
WnvD	Sokak	string
WnvD	Tuzak	string
WnvD	Adres Numarası ve Sokak	string
WnvD	Enlem	decimal
WnvD	Boylam	decimal
WnvD	Adres Doğruluğu	int
WnvD	Sivrisinek Sayısı	int
WnvD	WNV (Batı Nil Virüsü) Varlığı	int



Şekil 14. CHD ve WnvD'ye ait coğrafi noktaların koordinatlarının çizilmesi ve harita üzerinde gösterilmesi (sırasıyla) (Aydın & Erdem, 2023)

Tezin ikinci safhasında, İstanbul'daki bir dağıtım şirketine ait 6 aylık GPS veri seti kullanılmıştır. Bu veriler, Mobiliz Araç Takip Sisteminde, dağıtım şirketine ait kayıtların tutulduğu verilerdir. Mobiliz Araç Takip Sistemi, GPS teknolojisini kullanan, araçların konumunu, hızını, mesafesini, sürücü davranışlarını, rotalarını ve filo yönetimini takip eden sistemdir. Veri seti, 21 sütun ve toplamda 4,784,003 satırdan oluşmaktadır. Her kayıt, "konum (enlem, boylam), zaman, plaka, araç etiketi, filo, grup, sürücü, kimlik numarası, adres, hız, rota, mesaj türü, alan adı, araç türü, araç alt türü, araç markası, araç modeli, yıl, yakıt türü ve mesaj işleme süresi" içerir. İstanbul'daki bir dağıtım şirketinin ulaşım operasyonlarına odaklanan bu veri seti, çalışmanın yerel operasyonel bağlamlarla uyumunu güçlendirir. Ancak, İstanbul'la coğrafi sınırlaması ve altı aylık zaman dilimi, bulguların diğer bölgelere veya genişletilmiş operasyonel dönemlere genelleştirilebilirliğini kısıtlayabilir. Yine de bu veri seti kentsel lojistik zorluklarının derinlemesine araştırılması için mükemmel bir temeldir. Veri setindeki her bir kayda ait coğrafi koordinat bilgilerinin dağılımını görselleştirmek için Şekil 15'de istatistiksel özet sunulmuştur (Erdem *et al.* under review e).



Şekil 15. Veri kümesindeki enlem ve boylam değerlerinin istatistiksel özeti (Erdem *et al.* under review e)

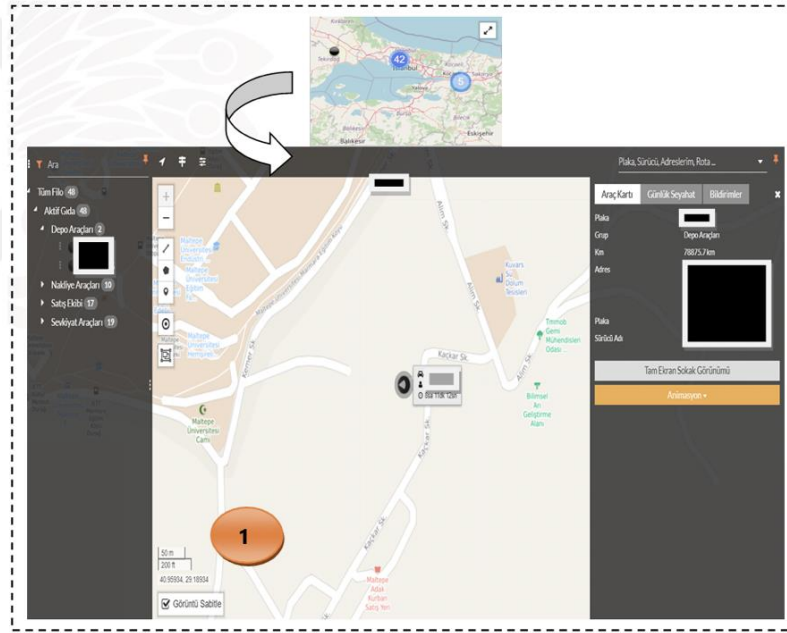
Tezin üçüncü safhasında, çalışmada kullanılan tüm veriler iki farklı sistemden alınmıştır. İlk veri tabanı, firmanın satın alma faturalarını, satış verilerini ve sevk irsaliyelerini içeren Mikro V15 veri tabanıdır. Diğer veri seti Mobiliz'den alınmıştır. Kullanılan veri kaynakları aşağıda ayrıntılı olarak açıklanmıştır:

- ✓ Micro V15 Veri Seti: Şirketin satın alma, satış ve sevkiyat verilerini içeren bir veri tabanıdır. Micro V15 veri seti, firmanın yaklaşık 600 dağıtım noktasına sağladığı verilere ilişkin bilgileri içermektedir ve detaylar Tablo 7'de sunulmuştur. Her yıl dağıtım noktası ~%10 oranında değişebilmektedir. Veri tabanında toplam 1619 tablo (Cari Hesap Adresleri, Cari Hesap Hareketleri, Cari Hesap Yetkilileri, Cari Hesaplar, Demirbaşlar, Personeller, Siparişler, Stok Alt Grupları, Stok Ana Grupları, Stok Hareketleri, Stok Markaları, Stoklar, vb.) mevcuttur. Bu çalışmada, 2013-2023 yılları arasında alınan 178,000 müşteri siparişine ait veriler kullanılmıştır (Erdem *et al.* under review b).

Tablo 7. Micro V15 Veri tabanında Örnek Bir Kesitin Ayrıntıları (Erdem *et al.* under review b)

Tablo	Toplam Kayıt Sayısı	Toplam Sütun Sayısı
Cari hesap adresleri	6899	55
Cari hesap hareketleri	849320	155
Cari Hesaplar	7024	182
Demirbaşlar	204	116
Siparişler	922396	126
Stoklar	1326	215
Stok Hareketleri	1459223	138
...		

- ✓ Mobiliz GPS Verileri: Mobiliz sistemine ait yetkili kullanıcı arayüzü Şekil 16’da sunulmuştur. Sistem üzerinden temin edilen GPS tabanlı ticari araç takip verileri kullanılmıştır. Bu veri seti, Tablo 8’de verilen, konum, hız, seyahat mesafesi ve sürücü davranışlarını içermekte olup yaklaşık 2–3 yıllık bir dönemi kapsamaktadır. Toplamda 4,784,003 GPS kaydı, rota optimizasyonu için gerekli zaman-mesafe matrisinin oluşturulmasında kullanılmıştır. Gerçek dünya verilerinin yanı sıra, ölçeklenebilir deneyler gerçekleştirebilmek amacıyla 4 depolu ve 600 müşterili sentetik bir örnek model geliştirilmiştir. Şekil 17’de gösterildiği üzere, depolar renkli karelerle ve müşteriler aynı renkteki dairelerle gösterilmiş olup, her bir müşteri ilgili depoya atanmıştır. Müşteri talepleri, sipariş hacmindeki değişkenliği simüle edebilmek amacıyla önceden tanımlanmış bir dağılımdan rastgele örneklenmiştir. Bu yapılandırma, lojistik ağ üzerindeki mekânsal dağılımı ve talep yoğunluğunu analiz etmeyi mümkün kılmaktadır (Erdem *et al.* under review c).



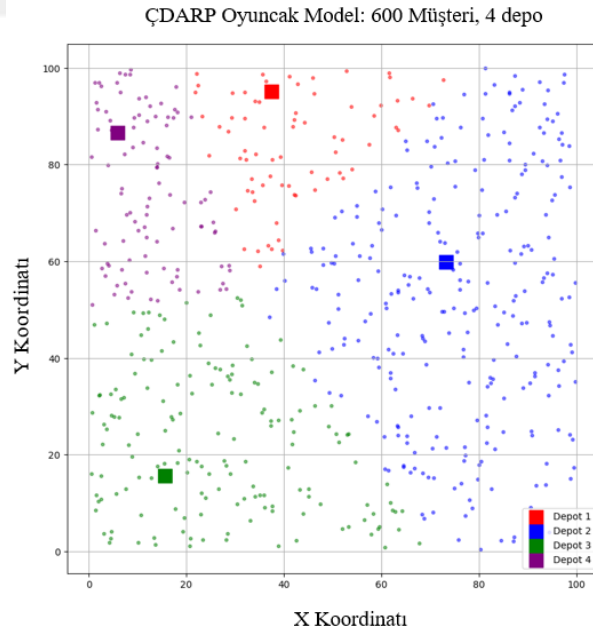
Şekil 16. Mobiliz araç takip sisteminin yetkili etkileşimli web sayfası arayüzü (Erdem *et al.* under review d)

Tablo 8. GPS Veri Kümesinin Özellikleri (Erdem *et al.* under review e)

Özellik	Veri Tipi	Açıklama
Tarih	Date	GPS sinyallerinin alındığı tarih
Zaman	Time	GPS sinyallerinin alındığı zaman dilimi (saat:dakika:saniye cinsinden)
Plaka	String	Aracın tescil edildiği ülke ve eyalet/kent kodunu içeren levha
Araç Etiket Kodu	String	Araca ait dahili etiket kodu
Filo	Categorical	Şirketin sahip olduğu tüm araçlar

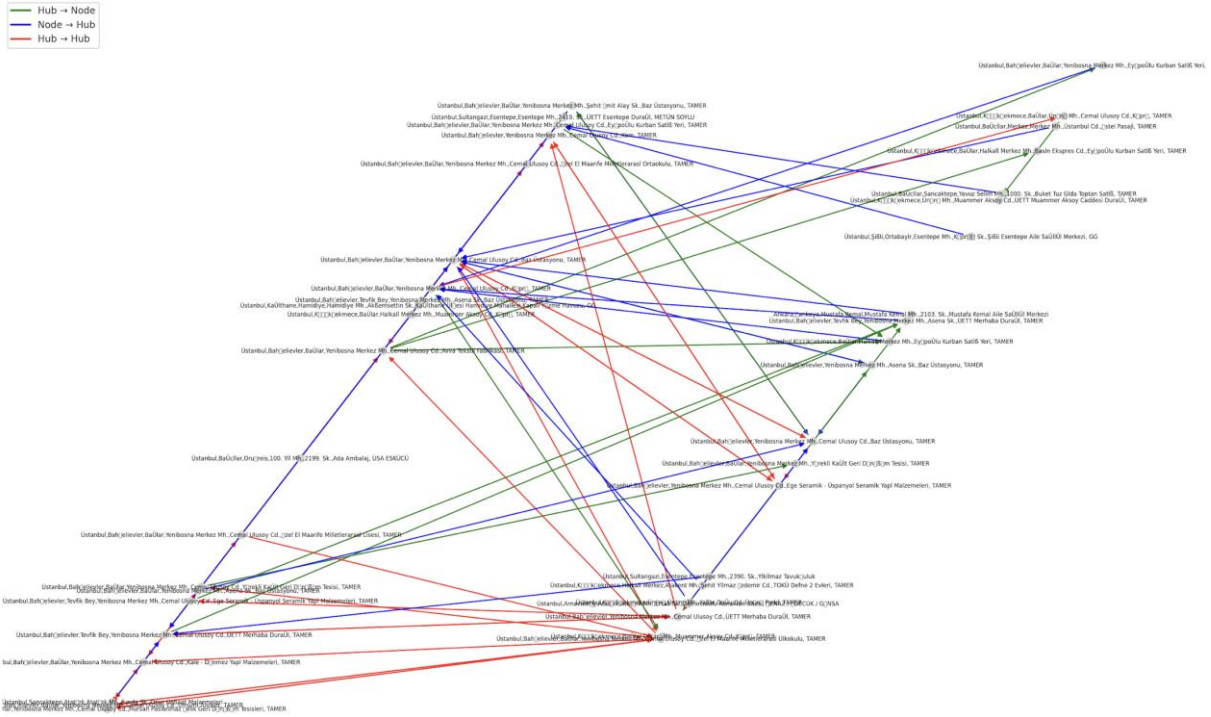
Tablo 8. (devamı)

Grup	Categorical	Benzer işlemlere sahip araçların bir araya getirildiği birim (Nakliye Araçları, Sevkiyat Araçları, Depo Araçları, Satış Ekibi)
Sürücü Adı	String	Sürücünün tam adı
Sürücü ID	Integer	Sürücüye ait benzersiz kimlik numarası
Adres	String	GPS noktasının kaydedildiği tam adres
Hız (km/sa)	Float	Araçın km/s cinsinden hızı
Boylam	Float	GPS'ten alınan boylam değeri
Enlem	Float	GPS'ten alınan enlem değeri
Yol (km)	Float	Araçların izlediği güzergâh
Mesaj Türü	String	GPS verilerinin veya durum bildirimlerinin türü (Ani hareket, aşırı hız başlangıcı, kontak açıldı, rölanti başlangıcı, vb.)
Alan Adı	String	Mesaj/araca atanmış alan grubu
Araç Türü	Categorical	Araçın genel türü (ör. Çekici)
Araç Alt Türü	String	Araçın alt türü (ör. Tır, kamyon)
Araç Markası	String	Araç markası (ör. FORD)
Araç Modeli	String	Araç modeli (ör. F-MAX)
Yıl	Integer	Araç üretim yılı
Yakıt Türü	String	Yakıt türü (ör. Dizel, Benzin)
Mesaj İşlem Zamanı	Timestamp	GPS mesajının işlendiği zaman

**Şekil 17.** ÇDARP örnek model (Erdem *et al.* under review c)

Simülasyon tabanlı görselleştirmelere ek olarak, gerçek GPS verileri kullanılarak araç hareketlerinin rotaları da görselleştirilmiştir. Bu yaklaşım, operasyonel davranışların kapsamlı bir şekilde anlaşılmasını sağlamaktadır. Şekil 18’de sunulan hareket görselleştirmesinde mavi oklar depodan müşteriye yapılan teslimatları, yeşil oklar müşteri dönüşlerini ve kırmızı oklar

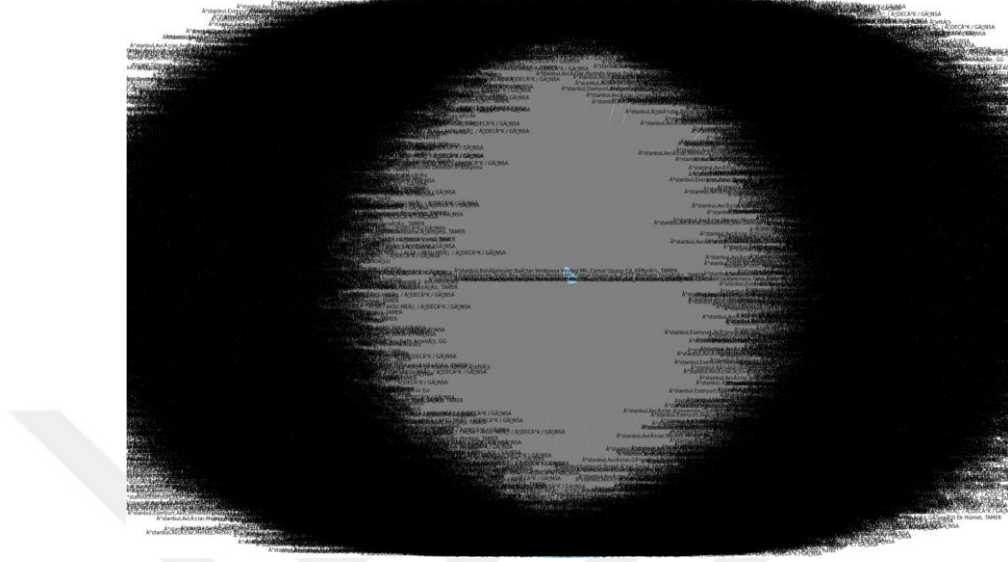
ise depo-depo arası transferleri temsil etmektedir. Gerçek GPS verileri kullanılarak araç hareket yönleri analiz edilmiş ve Mobiliz veri setinden elde edilen zaman sıralı ilk 250 hareket kaydı baz alınarak araç yolları oluşturulmuştur. Her aracın teslimat sırası önceliklendirilmiş, bu da teslimatların sıklığından çok operasyonel zamanlamasına odaklanılmasını sağlamıştır. Bazı araçlar 250’den az kayda sahip olsa da tüm veriler kronolojik olarak işlenmiştir. Bu analiz, “hub-to-customer” ve “customer-to-hub” lojistik ilişkilerinin pratikte nasıl gerçekleştiğini ortaya koyarak dağıtım ağındaki gerçek hareket örüntülerini tanımlamayı mümkün kılmıştır. Renk kodlaması sayesinde bu örüntüler tablolarla tespit edilmesi zor olan davranışları da ortaya koymuştur (Erdem *et al.* under review).



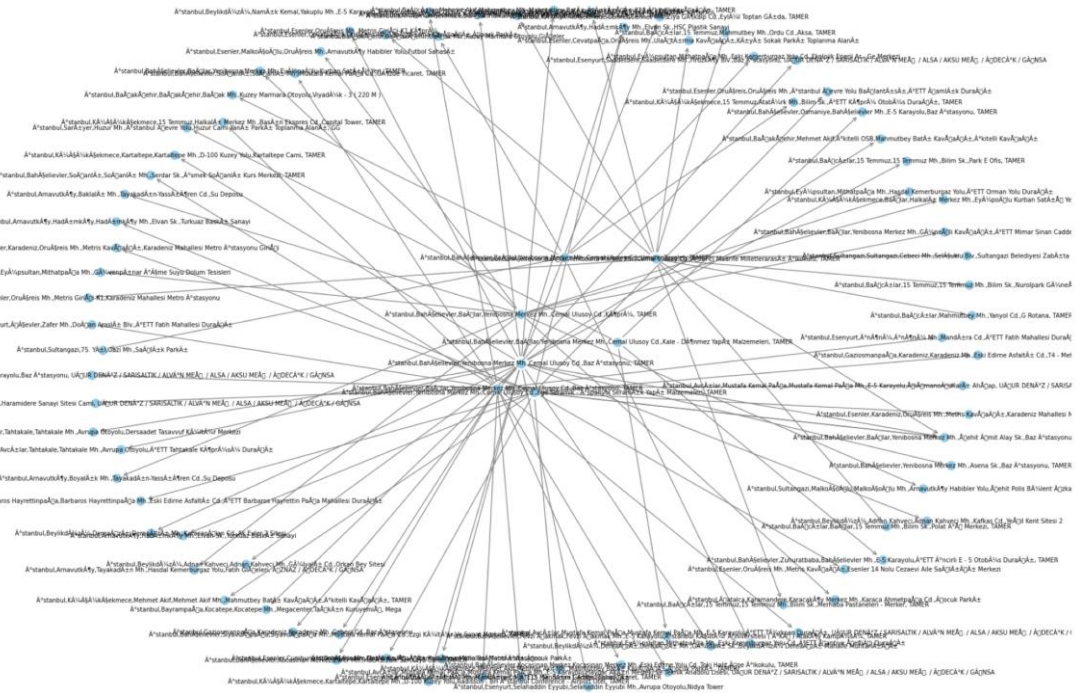
Şekil 18. Gerçek GPS verilerine dayalı araç hareket grafiği, depo-müşteri etkileşimleri (Erdem *et al.* under review c)

Şekil 19 ve 20’de gösterildiği üzere, “hub-and-spoke” yapıları (merkezî dağıtım noktaları ile bunlara bağlı müşteri hatlarından oluşan yıldız tipi dağıtım ağı) Haziran 2021 ve Haziran 2022 tarihlerindeki gerçek GPS verileriyle oluşturulmuş ve zaman içinde operasyonel yapının nasıl değiştiği görselleştirilmiştir. 2022 yılına ait ağ yapısı daha az yoğun, daha sade ve yorumlanması daha kolay bir görünüm sergilemektedir. Merkez sayısı sınırlı, her merkezden çıkan hat (yani müşteri bağlantısı) sayısı ise görece düşüktür. Bu durum, 2022 yılında yapılan dağıtım stratejisinin daha merkezî, belirli müşteri gruplarına ya da coğrafi bölgelere odaklanan bir yapıda olduğunu göstermektedir. Buna karşılık, 2021 yılına ait ağ yapısı çok daha yoğun ve dağınıktır. Merkez sayısı fazladır ve her birinden çok sayıda müşteri bağlantısı çıkmaktadır. Bu, daha geniş bir müşteri kitlesine hizmet verildiğini, daha fazla araç faaliyeti olduğunu ve

daha karmaşık bir dağıtım yapısının kullanıldığını göstermektedir. Söz konusu farklar; operasyonel optimizasyon çabaları, müşteri profil değişimleri ya da talep yoğunluğu gibi dinamiklerden kaynaklanmış olabilir. Her iki görsel de 9 Haziran 2021 ve Haziran 2022’ye ait gerçek araç hareket verilerine dayanmaktadır.



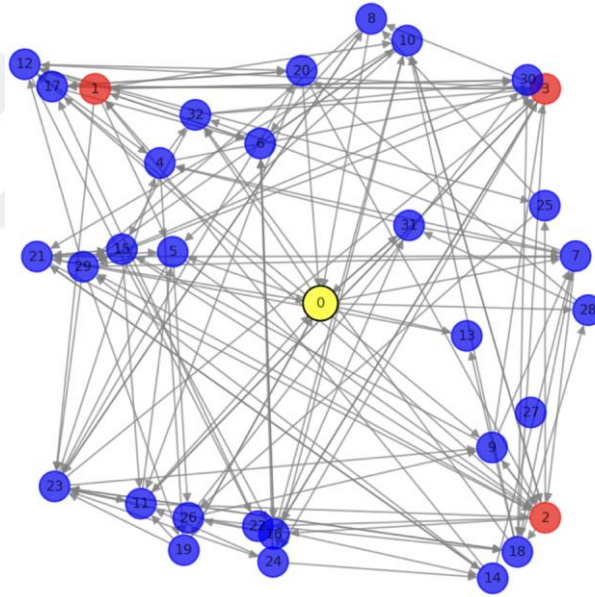
Şekil 19. Haziran 2021’e ait merkez-uç ağı (Erdem *et al.* under review c)



Şekil 20. Haziran 2022'ye ait merkez-uç ağı (Erdem *et al.* under review c)

Sentetik ve tarihsel verilere ek olarak, önerilen modellerin deneysel doğrulanması, 31 Mart 2023 tarihine ait ve 178,000 müşteri siparişini içeren Mikro V15 veri tabanından seçilen 29 gerçek sipariş ile gerçekleştirilmiştir. Bu seçim, şirketin dinamik ve günlük talebe dayalı rota optimizasyon ihtiyacını yansıtmak açısından anlamlıdır. Her ne kadar bu gündeki sipariş sayısı görece az olsa da, taşınan ürünlerin hacimli paletlerden oluşması araç doluluk ve kapasite

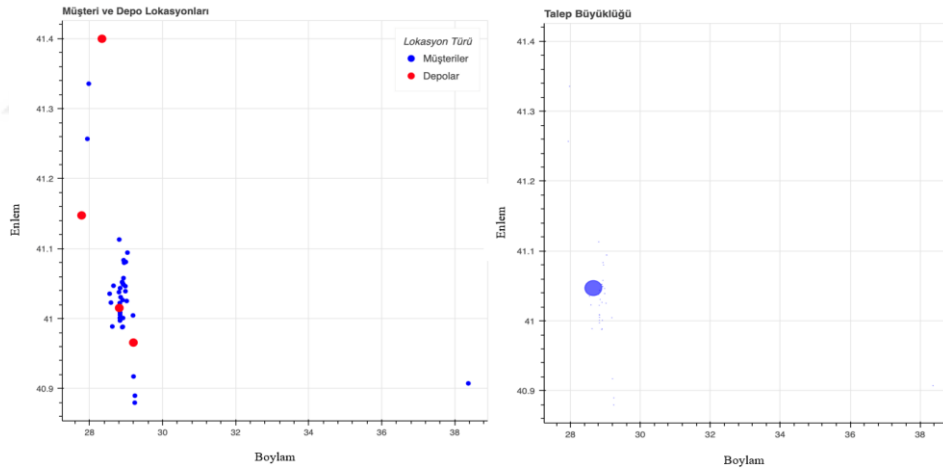
dengelemesi açısından önemli zorluklar barındırmaktadır. Dolayısıyla, bu koşullar altında rota optimizasyonu hâlâ kritik önem taşımaktadır. Bu durum, gerçek dünyadaki lojistik senaryolarında sıkça karşılaşılan “az sayıda ancak lojistik açıdan zorlayıcı teslimatlar” örüntüsünü temsil etmektedir. Bu nedenle, seçilen alt küme önerilen hibrit AI modellerinin gerçekçi ve operasyonel açıdan anlamlı kısıtlar altında test edilmesi için elverişli bir deney ortamı sunmaktadır. GPS tabanlı rota analizinin doğruluğunu sağlamak adına, Mobiliz ve Mikro V15 veri setleri arasında adres eşleştirmesi yapılmış, biçimsel farklardan veya eksik bilgilerden kaynaklanan tutarsızlıklar benzerlik tabanlı veri hizalama yöntemleri ile çözülmüştür. Şekil 21’de basitleştirilmiş bir örnek model sunulmuştur. Bu model dört depo, 29 müşteri ve 31 araçtan oluşmaktadır. Ana depo, rolünü vurgulamak amacıyla farklı bir renkle gösterilmiştir. Müşteriler hizmet bölgesi genelinde dağıtılmış olup, örnek araç rotaları teslimatların nasıl koordine edildiğini gösterecek şekilde görselleştirilmiştir. Bu örnek model, ÇDARP’nin temel zorluklarını — çoklu depo kullanımı, araç tahsisi ve kapasite kısıtları altında rota optimizasyonu — açıklamak açısından görsel bir açıklık sağlamaktadır.



Şekil 21. 4 depolu, öğle saatinde sipariş veren 29 müşteri ve 31 araçlı ÇDARP’nin yapısal karmaşıklığını gösteren örnek model (Erdem *et al.* under review d)

Şekil 22’de, 31 Mart 2023 tarihli gerçek sipariş verisine dayalı olarak oluşturulan müşteri ve depo lokasyonları ile talep büyüklüklerini görselleştiren iki ayrı grafik sunulmaktadır. Bu görseller, gerçek-dünya ÇDARP senaryosunun mekânsal ve operasyonel karmaşıklığını anlamaya yardımcı olmaktadır. Sol taraftaki grafik, müşteri ve depo noktalarının coğrafi konumlarını göstermektedir. Mavi noktalar müşterileri, kırmızı noktalar ise depoları temsil etmektedir. Dört ayrı deponun İstanbul’un farklı bölgelerine konumlandığı ve müşterilerin ağırlıklı olarak şehir merkezine yakın alanlarda kümелendiği gözlemlenmektedir. Bu mekânsal yoğunlaşma, bazı depoların daha fazla müşteri talebini karşılaması gerektiğini,

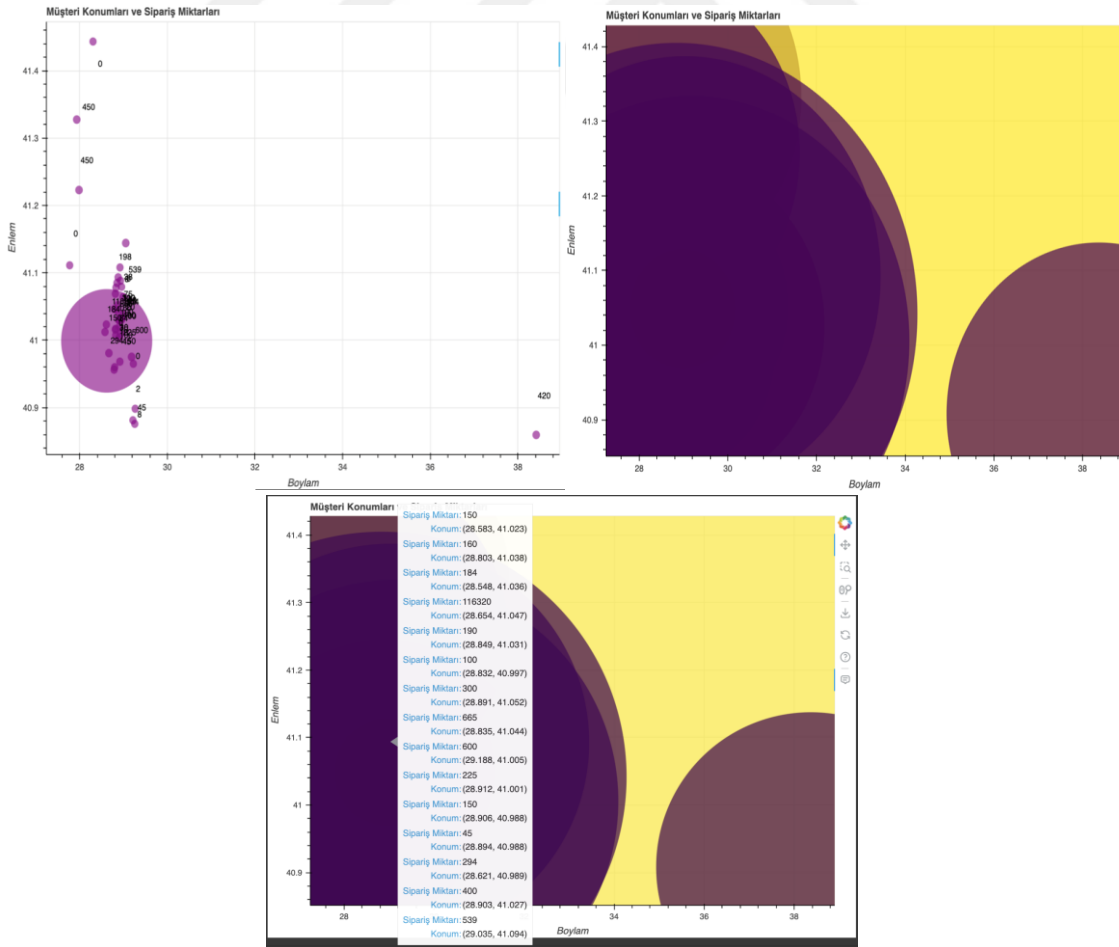
dolayısıyla rota optimizasyonunun bu yoğunluklara duyarlı olarak planlanmasının kritik olduğunu göstermektedir. Ayrıca, bazı müşterilerin oldukça uzak lokasyonlarda yer aldığı, bu durumun rota planlamasında sapma ve araç kapasitesi yönetimi açısından zorluk yaratabileceği görülmektedir. Sağ taraftaki grafik ise aynı müşteri lokasyonları üzerinde talep büyüklüklerini temsil etmektedir. Nokta büyüklüğü, her müşterinin sipariş ettiği ürün miktarına karşılık gelmektedir. Grafik, müşteriler arasındaki talep büyüklüğü açısından ciddi bir heterojenlik olduğunu ortaya koymaktadır. Özellikle şehir merkezine yakın konumlanan bir müşterinin talep büyüklüğünün, diğerlerine kıyasla oldukça yüksek olduğu dikkat çekmektedir. Bu tür yüksek hacimli siparişler, tek başına bir aracın kapasitesini doldurabilecek nitelikte olabilir ve bu nedenle bölünmüş teslimat veya çoklu tur gibi stratejilerin kullanılmasını gerektirebilir. Bu iki grafik birlikte değerlendirildiğinde, veri setinin hem mekânsal dağılım (müşteri ve depo lokasyonları) hem de lojistik zorluk seviyesi (talep büyüklüğü, mesafe, rota karmaşıklığı) açısından gerçek dünyaya oldukça yakın bir yapı sunduğu anlaşılmaktadır. Dağılımın dengesiz olması, klasik algoritmalarla çözülemeyecek kadar karmaşık rotalama problemlerine yol açabilmektedir. Bu bağlamda, önerilen hibrit AI modellerinin bu zorlukları dikkate alarak geliştirildiği ve bu tür karmaşık teslimat senaryoları için test edildiği görülmektedir.



Şekil 22. Müşteri ve depo lokasyonları ile talep büyüklüklerinin coğrafi dağılımı

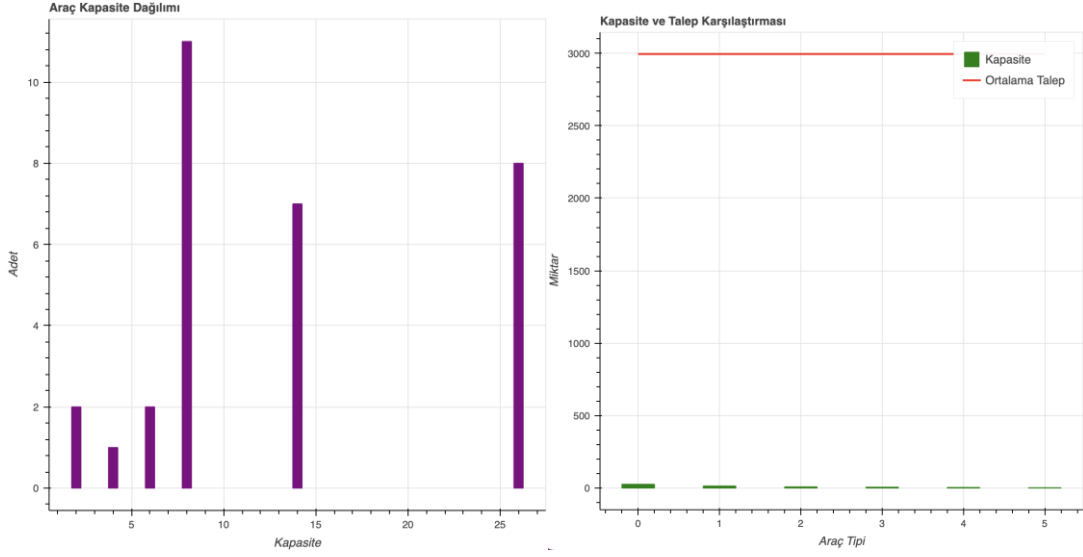
Şekil 23’de, 31 Mart 2023 tarihli gerçek sipariş verisine dayalı olarak oluşturulan müşteri konumları ve sipariş miktarlarına ilişkin üçlü bir görselleştirme sunulmaktadır. Sol üstte yer alan grafik, müşterilerin enlem-boylam koordinatlarına göre konumlarını göstermekte olup, her bir dairenin büyüklüğü o müşterinin sipariş miktarını temsil etmektedir. Daire içlerine yazılmış olan sayısal değerler, her bir müşterinin verdiği sipariş miktarını ifade etmektedir. Bu görselleştirmede İstanbul çevresinde önemli bir yoğunluk gözlemlenmekte, bazı müşterilerin 450 birime kadar çıkan sipariş talepleriyle öne çıktığı görülmektedir. Bu durum, aynı gün içinde az sayıda ama hacimsel olarak büyük teslimatların yapılmasının gerektiğini göstermektedir. Sağ üstteki ikinci grafik ise sipariş miktarlarının mekânsal yoğunluğunu daha bütünsel bir şekilde

sunan bir yoğunluk haritasıdır. Sarı alanlar görece düşük yoğunluğu, koyu mor alanlar ise yüksek sipariş hacmini temsil etmektedir. Özellikle İstanbul'un batı yakasında yer alan bölgelerdeki yoğun mor alanlar, sipariş taleplerinin bu bölgede yüksek yoğunlukta toplandığını ortaya koymaktadır. Uzakta bulunan bazı müşteriler ise daha seyrek ama yüksek hacimli siparişlerle dikkat çekmektedir. Bu da bazı rotaların yalnızca tek bir büyük siparişi taşıyan özel turlarla optimize edilmesini gerekli kılmaktadır. Alt kısımdaki interaktif grafik ise müşteri konumları ile birlikte her bir siparişe ilişkin detaylı nicel bilgileri metin formatında sunmaktadır. Her veri noktası, müşterinin verdiği sipariş miktarı ve bulunduğu coğrafi koordinat bilgisiyle birlikte görselleştirilmiştir. Bu yapı, özellikle rota planlamasında hangi müşterinin nerede bulunduğu ve ne kadar yük taşıdığı gibi sorulara doğrudan yanıt sunmakta; aynı zamanda veri setinin manuel olarak doğrulanmasına ve istisnai noktaların hızlıca tespit edilmesine olanak sağlamaktadır. Genel olarak bu görsel yapı, hem operasyonel planlamaya destek sağlayan analiz derinliğini hem de ÇDARP kapsamındaki gerçekçi kısıtların mekânsal etkilerini açık biçimde ortaya koymaktadır. Rotaların oluşturulmasında müşteri dağılımı, yoğunluk ve sipariş hacminin birlikte dikkate alınması gerektiğini vurgulayan önemli bir analiz sunulmuştur.

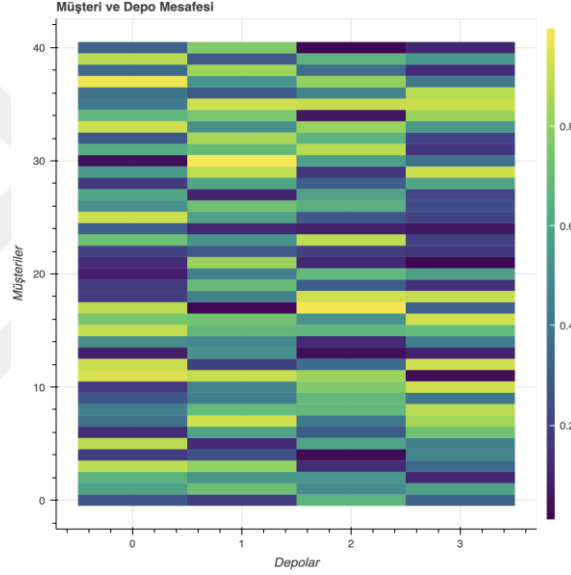


Şekil 23. Müşteri konumları ve sipariş yoğunluklarının coğrafi dağılımı (Erdem *et al.* under review d)

Şekil 24’de, çalışmada kullanılan araçlara ait kapasite dağılımı ve bu kapasitenin toplam müşteri talebini karşılamadaki yeterliliği görsel olarak sunulmuştur. Sol taraftaki grafik, farklı kapasiteye sahip araçların kaç adet olduğunu göstermektedir. En çok sayıda bulunan araçlar, 8 ve 26 birim kapasiteli olanlardır. Bu dağılım, filo yapısının hem küçük hem de büyük ölçekli teslimatlar için esnekliğe sahip olduğunu göstermektedir. Sağ taraftaki grafikte ise, her bir araç tipi için toplam taşıma kapasitesi ile sistemdeki ortalama müşteri talebi karşılaştırılmıştır. Yeşil çubuklar araç tiplerine ait toplam kapasiteleri, kırmızı çizgi ise 31 Mart 2023 tarihli veri setinde gözlemlenen ortalama toplam müşteri talebini temsil etmektedir. Görüldüğü üzere, tüm araçların toplam taşıma kapasitesi ortalama talebin oldukça altında kalmaktadır. Bu durum, sistemin toplam müşteri talebini karşılayabilmesi için çoklu tur uygulamasının zorunlu olduğunu göstermektedir. Bu analiz, çalışmanın genel akışında yer alan ilk veri toplama ve ön hazırlık aşamasına denk gelir. Bu aşamada GPS verileri ve sevkiyat kayıtları toplanarak, müşteri talepleri, depo konumları ve araç kapasiteleri belirlenmektedir. Takip eden adımda, bu veriler kullanılarak müşteri-depo mesafe ilişkilerini gösteren bir zaman-mesafe matrisi oluşturulmakta, bu da güzergâh optimizasyonunun çözümünde temel teşkil etmektedir. Şekil 25’de sunulan ısı haritası, müşteri ve depo mesafelerini görselleştirmektedir. Yatay ekseninde dört depo (0–3), dikey ekseninde ise her bir müşteri yer almakta; renk skalası ise müşteri-depo çiftleri arasındaki normalize edilmiş mesafe değerlerini temsil etmektedir. Koyu mor tonlar, müşterinin ilgili depoya yakın olduğunu, açık sarı-yeşil tonlar ise mesafenin daha uzak olduğunu göstermektedir. Görsel incelendiğinde, müşteri-depo mesafelerinin oldukça düzensiz ve heterojen bir yapı sergilediği açıkça görülmektedir. Bazı müşteriler birden fazla depoya yakınen, bazıları tüm depolara uzak konumda bulunmaktadır. Bu çeşitlilik, güzergâh planlamasında sabit bir strateji yerine, veri odaklı ve dinamik olarak ayarlanabilen yaklaşımların kullanılmasını zorunlu kılmaktadır. Bu görsel analiz, rota planlamasının yalnızca mesafe değil, aynı zamanda taşıma kapasitesi ve taleple dengelenmesi gerektiğini vurgulamakta; çalışmada önerilen hibrit modellerin bu tür çoklu kısıtlar altında nasıl optimize edildiğine dair önemli bir ön bilgi sunmaktadır.



Şekil 24. Araç kapasite dağılımı ve talep karşılama analizi (Erdem *et al.* under review d)



Şekil 25. Müşteri ve depolar arasındaki normalize mesafe ısı haritası (Erdem *et al.* under review d)

Şirket tarafından bilgileri sağlanan güncel filolara ait yer alan araçların kantar toleransı ve yüklenebilir miktarı Tablo 9’da verilmiştir.

Tablo 9. Araçların Özelliklerine İlişkin Detaylar (Erdem *et al.* under review b)

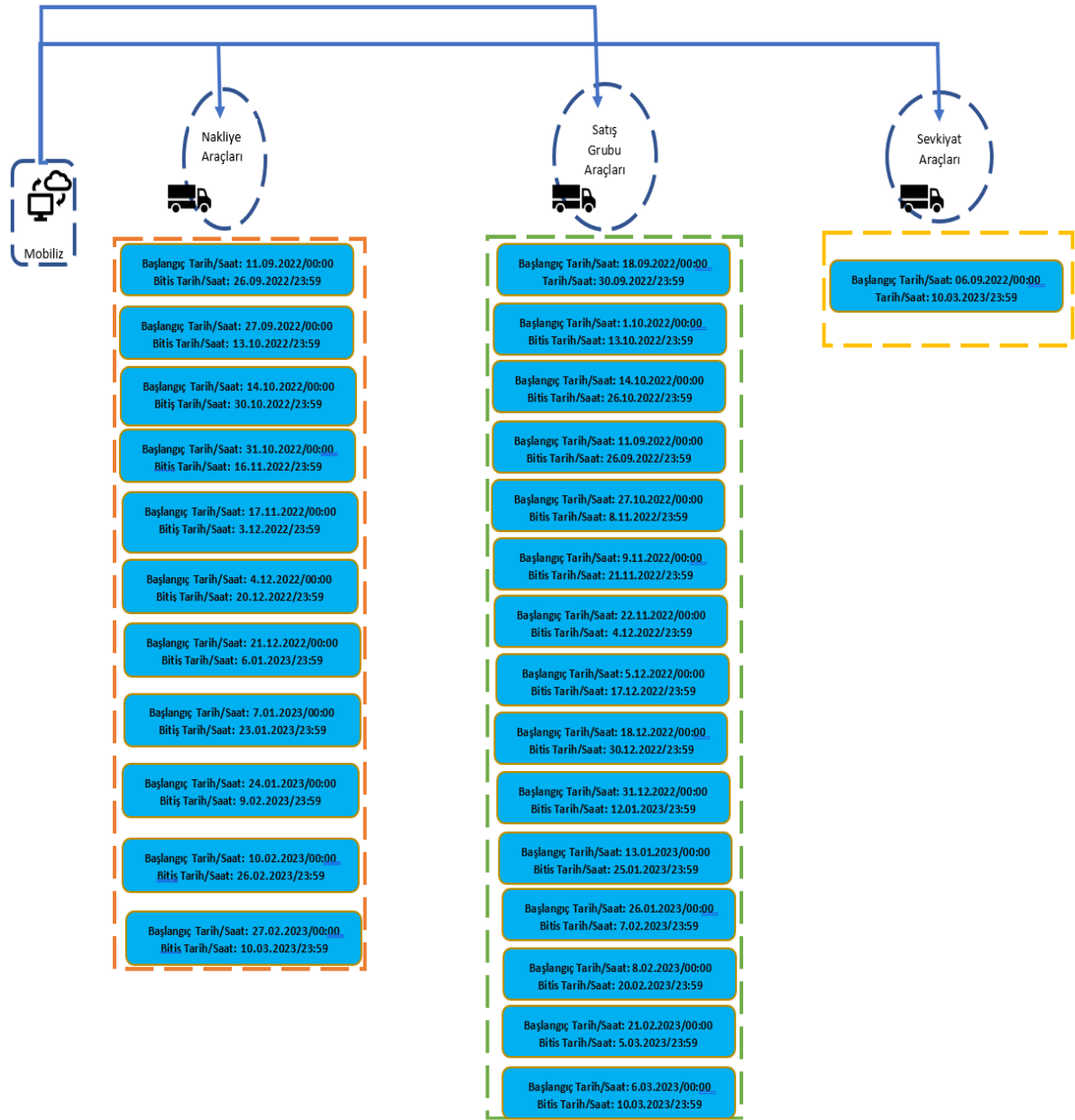
Demirbaş İsmi (Marka/Model Kısaltma)	Kantar Boş Ağırlık	Brüt Ağırlık	Yükleme Kapasitesi (Kg)	Azami Yüklü Ağırlık (3,75 + 500 Kg Tolerans Dahil)	Max Yüklenebili r (Kg)	Yüklemeye Baz Alınacak (Kg)	Palet Sayısı
CRG-2526	11,280	25,000	13,720	26,438	15,158	15,058	14
CRG-2533 MDL	11,300	25,000	13,700	26,438	15,138	15,038	14
...							
SZ-NR-3D	5,430	15,000	9,570	16,063	10,633	10,533	8
SZ-NR-3D	5,810	15,000	9,190	16,063	10,253	10,153	8

Tablo 9. (devamı)

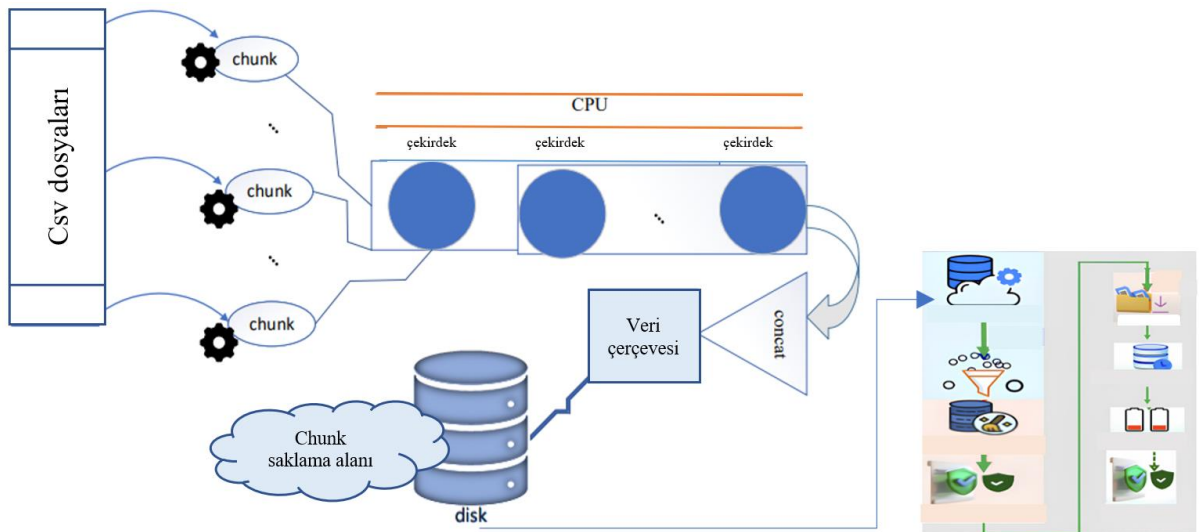
				...			
SZ-NR-3D	4,330	9,800	5,470	10,668	6,338	6,238	6
VC-DLY	3,800	7,200	3400	7,970	4170	4,070	4
RD-TRNST	2,830	4,700	1,870	5,376	2,546	2,446	2
RDFMAX-500	15,400	40,000	24,600	42,000	26,600	26,500	26
				...			
RDFMAX-500	15,400	40,000	24,600	42,000	26,600	26,500	26

Veri ön işleme

Şekil 26’da gösterildiği gibi, GPS Veri seti, “Nakliye Araçları”, “Satış Grubu Araçları” ve “Sevkiyat Araçları’na” ait verileri içermektedir. Veri kaybını önlemek amacıyla farklı yöntemlerle birleştirme işlemleri gerçekleştirilmiştir. İlk olarak, “cmd” komut satırı kullanılarak birleştirme yapılmıştır. Bu işlem sonucunda, veri setinin toplamda 4,784,003 satır ve 21 sütundan oluştuğu görülmüştür. Büyük verilerle çalışıldığı için veri okuma süresi optimize edilmiştir. Veri seti küçük dosyalara bölünerek paralel okuma “bölütleme (chunk)” yapıları halinde işlenmiştir. Bu işlem sırasında veri seti bölünmüş ve hafıza kullanımı azaltılmıştır (Tablo 11). İşlem süresi hızlandırılmış ve veriler küçük parçalara ayrılarak daha verimli bir şekilde yönetilmiştir. İşlem sırasında kullanılan bölütleme algoritmasının yalancı kodu Tablo 10’da verilmiştir. Bölütleme işlemi sırasında Python Pandas kütüphanesi kullanılmıştır. Büyük veri setleriyle çalışırken, “bölütleme boyutu” parametresi bir seferde okunan veri miktarını temsil etmektedir. Tüm “.csv” dosyaları tek bir klasörde toplanmıştır. Bölütlemeler halinde işlenen her dosya daha sonra dikey olarak birleştirilerek tek bir veri çerçevesi oluşturulmuştur. Daha küçük “bölütleme boyut” değerleri, daha uzun işlem sürelerine neden olmuştur. Ayrıca uygulanacak modellerin daha tutarlı genelleme yapmasını sağlamak için özellik seçimi ile gereksiz özellikler çıkarılmıştır (Şekil 27).



Şekil 26. 6 Aylık GPS verilerine ait bilgiler (Erdem *et al.* under review e)



Şekil 27. Veri ön-işleme adımları (Erdem *et al.* under review b)

Tablo 10. Bölütleme Algoritması Sözde Kodu (Erdem *et al.* under review b)

Bölütleme Algoritması
1: Tum_Dosya_Listeleri=input("./all_csv_Files/*.csv")
2: chunk_size←[10 100 1000]
3: başla
4: chunks=[]
5: for dosya in Tum_Dosya_Listeleri:
6: for chunk in read(dosya, chunk_size):
7: chunks.append(chunk)
8: Dataframe=concat(chunks)
9: print (len(Dataframe))
10: print (info(Dataframe))
11: son

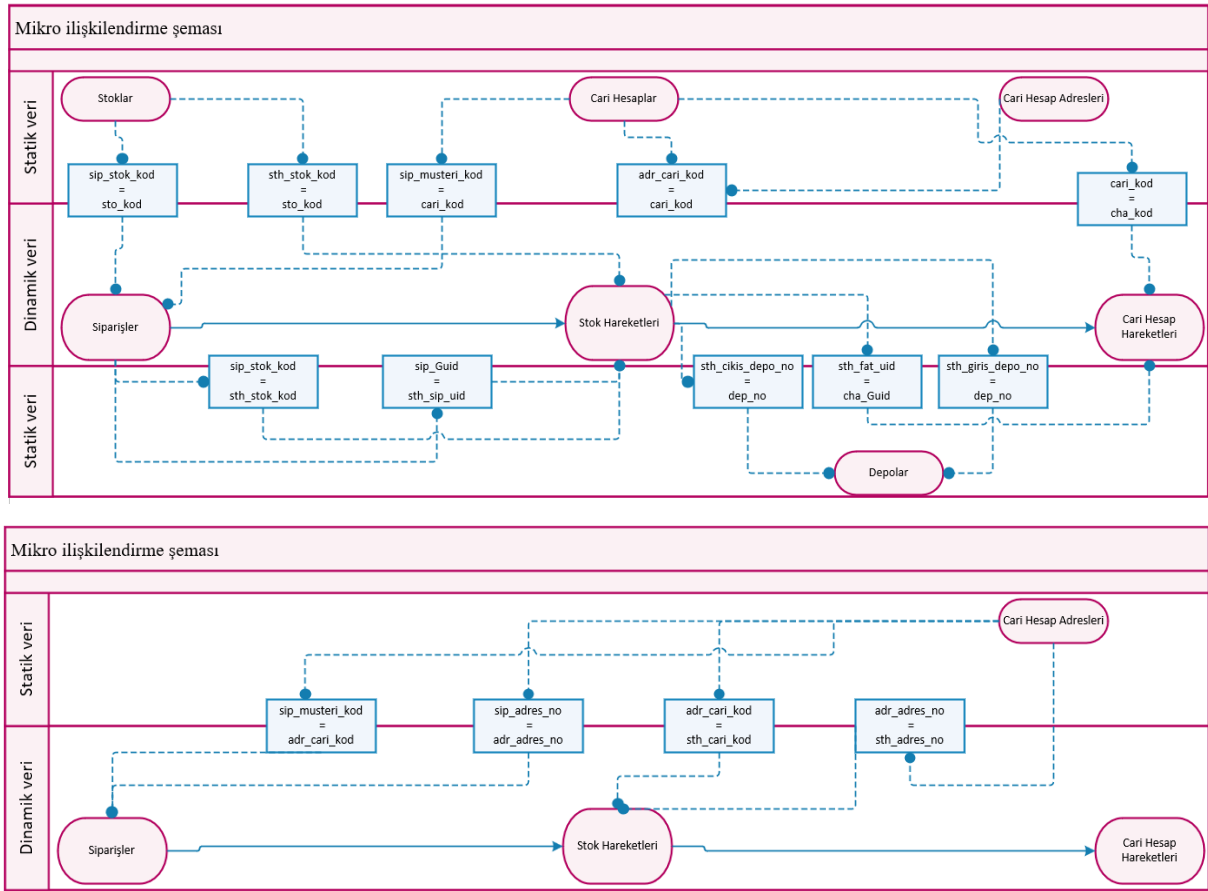
Tablo 11. Farklı Yöntemlerle İşlenen Veri Setinin Bellek Kullanım Sonuçları (Erdem *et al.* under review b)

Metod	Bellek Kullanımı
cmd	803.0+ MB
chunk	69.3+ MB

Micro V15 veri setinin karmaşıklığı nedeniyle, problem çözümünde kullanılacak tablolar seçilmiştir. İlgili tablolardaki ilişkilendirme adımları, firma bilgileri doğrultusunda gerçekleştirilmiştir (Şekil 28). İş süreçleri, dokümantasyon, veri tabanları ve cari hesap yönetimi gibi unsurların araç rotalama ve dağıtım optimizasyonundaki etkileri incelenmiştir. Örneğin, lojistik süreçlerde kullanılan satın alma faturaları, satış verileri ve sevkiyat notları gibi iş belgeleri, stok hareketlerinin ve finansal işlemlerin gerçek zamanlı takibiyle entegre edilmiştir. Bu veri kaynakları, işletme süreçlerinin daha verimli ve bilgiye dayalı kararlarla yönetilmesine katkı sağlamaktadır. Sevkiyat sürecinde, Şekil 28’de ok yönünde gösterilen süreç sırasına göre, müşterilerden siparişler alındıktan sonra stok hareketleri (stok miktarlarını etkileyen, stok giriş/çıkış işlemlerinin kaydedilmesi vb.) kaydedilir. Daha sonra cari hareketler (müşterilerin yaptığı ödemeler, tahsilatlar, fatura işlem detaylarına ait bilgiler vb.) işlenir. Mikro V15’den alınan verilerde, Şekil 28’de gösterilen ilişkilendirme işlemleri yapıldıktan sonra sipariş veren ve sevkiyat işlemi gerçekleştirilen müşteriler işleme alınmıştır. Yapılan işlemler sonucunda elde edilen veriseti Mikro Sevkiyat Veri Model Veriseti olarak adlandırılmıştır. Daha sonra, veri setleri üzerinde veri temizliği yapılmıştır; eksik (NaN, null) veya hatalı veriler işleme alınmamıştır. “Stok hareketleri” ve “Siparişler” tablolarında, veri kaybının önüne geçmek için “sth_stok_kod” ve “sip_stok_kod” sütunlarında yer alan veriler dönüştürülmüştür. “99,010,002,910”, “1,010,000,016” vb. formatlar ve float veri tipinde olan değerler, veri dönüşümü ile “str” veri tipine çevrilmiştir. “replace” işlemi ile de kodun en

sağında binlik ayırıcındaki nokta kaldırılmıştır. Çünkü binlik ayırıcı ile ayrılmayan ancak aynı stok koduna sahip verilerin varlığı tespit edilmiştir. Ayrıca, boşluklar ve diğer olası istenmeyen karakterlerin silinmesi “strip” işlemiyle yapılmıştır. Kodun başında fazladan bulunan “0” değerlerinin silinmesi ise “lstrip” ile gerçekleştirilmiştir. Tüm bu işlemlerin ardından veri boyutlarındaki değişim Tablo 12’de verilmiştir.

Veri dönüşümü yapılarak ML algoritmalarına girdi olarak verilecek kategorik veriler (müşterilerin fiziksel adres bilgileri) Python LabelEncoder kütüphanesi kullanılarak LabelEncoder() ile sayısal değerlere çevrilmiştir. ML modellerine yönelik giriş verileri genellikle farklı aralıklara sahiptir. Bu durumu çözmek için, Python StandardScaler kütüphanesinden StandardScaler() fonksiyonu kullanılarak veriler yeniden ölçeklendirilmiştir.



Şekil 28. Şirket sevkiyat işlemleri-dağıtım veri tabanı (DB) yapısı (Sevkiyat veri modeli) (Erdem *et al.* under review b)

Tablo 12. Veri Boyutlarındaki Değişiklikler (Erdem *et al.* under review b)

Tarih	Veri Boyutu (Mikro Veri tabanı İlişkilendirme Şeması Uygulanmadan Önce)	Veri Boyutu (Mikro Veri tabanı İlişkilendirme Şeması Uygulandıktan Sonra)
31.03.2023	29	23
2013-2023	922396	2903

Arka Plan

Tezin ilk ve ikinci aşamasında, veri setleri içindeki değişkenlerin birbiriyle olan ilişkileri korelasyon matrisi ile analiz edilmiştir. Korelasyon matrisi, veri setindeki değişkenlerin GPS konumlandırma üzerindeki etkisini görselleştirmek ve hangi değişkenlerin daha etkili olduğunu anlamak için temel bir araç olarak kullanılmıştır. Korelasyon ısı haritaları, değişkenler arasındaki pozitif ve negatif ilişkileri Şekil 29’da görselleştirmiştir. Çalışmada, Tablo 13’de verilen geri eleme yöntemi kullanılarak bağımsız değişkenler arasında multikolineerite analizi yapılmıştır. Bağımsız değişkenler, açıklayıcı değişken, ve p-değeri anlamlılık seviyesi (önem düzeyi = 0.05) algoritmanın girdileridir. En Küçük Kareler Yöntemi (OLS) ile model eğitimi gerçekleştirilmiştir. Algoritmanın 7-16. adımları, modeldeki tüm bağımsız değişkenlerin p-değeri 0,05’in altına düşene kadar tekrar edilmiştir. Ayrıca, bağımsız değişkenlerin modeldeki önemine göre seçimi optimize edilmiştir.

Tezin ilk aşamasında, CHD ve WnvD veri setleri üzerinde bu işlemler yapıldığında;

- ✓ Değişkenlerin Seçimi ve Analizi: Geri eleme algoritması kullanılarak, istatistiksel öneme sahip olmayan değişkenler modelden çıkarılmıştır. Ancak, iki veri seti için yapılan analiz sonucunda, tüm değişkenler modelde tutulmuştur (p-değeri < 0,05) (Tablo 14, 15).
- ✓ Korelasyon Haritası: İki veri setinde de enlem ve boylam değişkenleri arasında güçlü bir negatif korelasyon olduğu gözlenmiştir. Özellikle CHD veri setinde, “Total Rooms”, “Total Bedrooms”, “Households” ve “Population” değişkenleri arasında güçlü bir pozitif ilişki olduğu belirtilmiştir (Şekil 29).
- ✓ Multikolineerite Analizi: Tüm bağımsız değişkenlerin korelasyon katsayıları 0,7’nin altında bulunmuştur. Bu durum, değişkenler arasında multikolineerite olmadığını göstermiştir (Tablo 14 ve Tablo 15).

Tezin ikinci aşamasında, geçmiş GPS veri seti üzerinde bu işlemler yapıldığında;

- ✓ Değişkenlerin Seçimi ve Analizi: Model eğitimin sonunda, bağımsız değişkenlerin tamamı modelde tutulmuştur. Yani, veri setindeki hiçbir predictor çıkarılmamıştır (Tablo 16).
- ✓ Korelasyon Haritası: Enlem ve Boylam değişkenleri arasında güçlü bir negatif korelasyon gözlemlenmiştir. Veri setindeki diğer değişkenler arasında anlamlı bir korelasyon bulunmamıştır (Şekil 30).
- ✓ Multikolineerite Analizi: Enlem değişkeninin korelasyon katsayısı, eşik değer ($\varphi = 0,7$)’yi aşmamıştır (Tablo 16). Bu nedenle veri setindeki değişkenler arasında çoklu doğrusal bağlantı olmadığı belirlenmiştir.

Tablo 13. Geriye Doğru Eleme Yöntemi Sahte Kodu (Aydın & Erdem, 2023; Erdem *et al.* under review e)

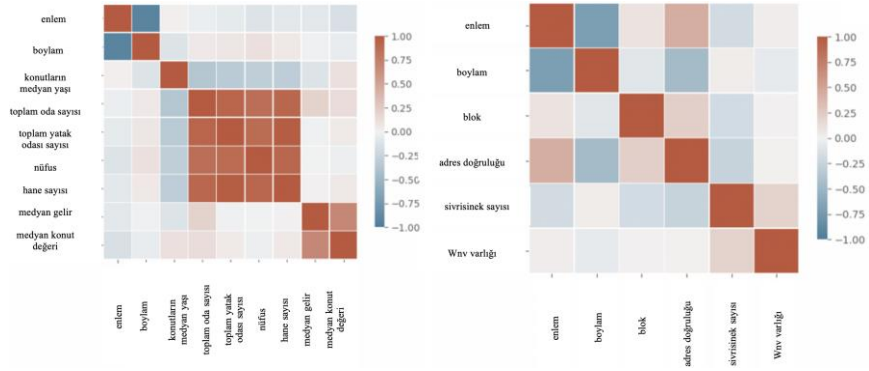
Algoritma: CHD ve WnvD Veri Kümeleri İçin Algoritma		
1:	veeri seti V^i : $V^i = \{(X_1, y_1), \dots, (X_j, y_j)\}$ (X veri tipi: “Dataframe” and y veri tipi: “Series”)	
2:	$p \leftarrow 0.05$ (anlamlılık düzeyini başlat)	
3:	başla	
4:	model yapısını yükle (“statsmodels.api” kütüphanesinden “OLS” sınıfını oluştur)	
5:	“OLS” sınıfının “fit()” fonksiyonunu kullan	
6:	modeli tüm olası bağımsız değişkenlerle (X ’ler) eğit	
7:	repeat	
8:	diğer değişkenlerin p -değerleriyle karşılaştırıldığında en yüksek P -değerine sahip olanı bul	
9:	if ($P\text{-degeri} > p$)	
10:	go to adım 14	
11:	else	
12:	go to adım 15	
13:	endif	
14:	mevcut değişkeni çıkar ve kalan tüm değişken listesini güncelle	
15:	go to adım 6	
16:	until (tüm değişkenlerin p -değeri $< p$)	
17:	son	

Tablo 14. CHD Veri Setinde Katsayılar (Coef) ve P-Değerlerinin Özeti

CHD VERİ SETİ	coef	P> t
enlem	-3,1336	0,000
konutların medyan yaşı	-0,0666	0,000
toplam oda sayısı	0,0018	0,000
toplam yatak odası sayısı	-0,0087	0,000
nüfus	-0,0023	0,000
hane sayısı	0,0041	0,000
medyan gelir	-0,6680	0,000
medyan konut değeri	-9,113e-06	0,000

Tablo 15. WnvD Veri Setinde Katsayılar (coef) ve P-Değerlerinin Özeti

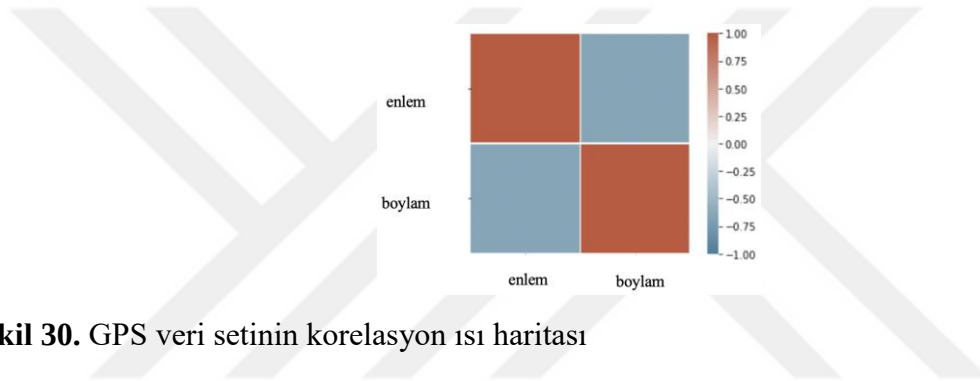
WNVD VERİ SETİ	coef	P> t
enlem	-2,1021	0,000
blok	-0,0002	0,000
adres doğruluğu	0,0363	0,000
sivrisinek sayısı	-0,0018	0,000
Wnv (Batı Nil Virüsü) varlığı	0,0282	0,000



Şekil 29. CHD ve WnvD'ye ait korelasyon ısı haritası

Tablo 16. GPS Veri setindeki Katsayı ve p-değeri Özet Bilgisi

VERİ SETİ	coef	P> t
enlem	0,7	0,000



Şekil 30. GPS veri setinin korelasyon ısı haritası

Metot

Tezin ilk aşamasında, hibrit ve topluluk modelleri, GPS konumlandırma doğruluğunu artırmak için önerilmiştir.

GPS verisinin 1B'den 4B'ye dönüştürülmesi

- 1B Veri: GPS verileri başlangıçta enlem, boylam, hız gibi tek boyutlu bir yapıdadır. Bu format, genellikle düzenli tablo yapısında işlenir.
- 4B Veri: GPS verileri, mekânsal ilişkilerin modellenenebilmesi için yapılandırılmamış görüntü formatına dönüştürülür. Bu dönüşümde:
 - Veriler bir tensör yapısına getirilir.
 - 4B tensörler, DL modellerine giriş olarak kullanılır.

ML tabanlı topluluk modelleri

Çeşitli regresyon modellerinin tahminlerinin birleştirilmesi için topluluk yöntemleri (oylama yöntemi ve yığma yöntemi) uygulanmıştır.

Yığınlama

Çalışmada, yığıma yöntemi şu şekilde uygulanmıştır: Baz modellerden gelen tahminler bir meta modele giriş olarak kullanılarak en iyi kombinasyon öğrenilmiştir. Meta öğrenme algoritması, ML bağlamında bir modelin nasıl daha iyi öğrenebileceğini öğrenmesini sağlayan bir yaklaşımdır. Bu yaklaşım, birden fazla modelin çıktılarından veya tahminlerinden faydalanarak, daha iyi bir genel performans elde etmek için bir üst modelin (meta model) eğitilmesiyle gerçekleştirilir.

Yığınlama uygulanma süreci

✓ Baz Modellerin Eğitilmesi:

- Farklı ML modelleri, eğitim veri seti üzerinde eğitilmiş ve tahminler üretilmiştir (Kullanılan baz modeller: kNN, DTR ve SVR).

✓ Meta Model İçin Özellik Oluşturma:

- Baz modellerin tahmin sonuçları, meta model için yeni bir özellik kümesi olarak kullanılmıştır. Tahmin sonuçları bir dizi sütun oluşturmuş ve bu sütunlar meta modelin girişleri olmuştur. LR, baz modellerin tahminlerinin ağırlıklarını öğrenmiş ve bu ağırlıklara göre final tahminini üretmiştir.

Oylama

Çalışmada, oylama yöntemi, modellerin tahminlerinin ortalaması alınarak oluşturulmuştur.

Oylama uygulanma süreci

✓ Baz Modellerin Eğitilmesi:

- İki farklı oylama modeli oluşturulmuştur:
 - İlk modelde, kNN ve LR kullanılmıştır.
 - İkinci modelde, kNN ve DTR kullanılmıştır.
- Bu modeller, eğitim veri seti üzerinde bağımsız olarak eğitilmiş ve her biri test veri seti için tahminler üretmiştir.

✓ Tahminlerin Birleştirilmesi:

- Baz modellerin tahminleri ortalama oylama yöntemiyle birleştirilmiştir.
- Ortalama oylama şu şekilde gerçekleşir: Her baz modelin tahminleri alınır ve aritmetik ortalaması hesaplanır.

DL tabanlı hibrit modeller

Çalışmada önerilen DL tabanlı yedi farklı hibrit model tasarlanmıştır. Bu modeller, GPSCNNs ve GPSLSTM olmak üzere iki ana grupta ele alınmıştır. Modeller, GPS konumlandırma verilerinin doğruluğunu artırmayı amaçlayan özel tasarımlar içerir. GPSCNNs, önceden eğitilmiş modeller ve özel CNN tabanlı modeller olmak üzere iki gruptan oluşurken; GPSLSTM modeli LSTM-SVR yapısındadır. Modellerin detaylı yapıları ve parametreleri verilmiştir.

GPSCNNs modelleri

GPSCNNs, CNN tabanlı modellerden oluşur ve şu şekilde sınıflandırılır: Önceden Eğitilmiş Modeller (Xception-SVR, VGG16-SVR, VGG19-SVR, AlexNet-SVR) ve Özel CNN Tabanlı Modeller (CNN1-SVR, CNN2-SVR, CNN3-SVR). GPSCNN’lerde yer alan önceden eğitilmiş modeller, ImageNet veri seti üzerinde eğitilmiş ağ yapılarıdır. Xception modeli (Chollet, 2017), derinlik ayrımlı konvolüsyon yapısına sahiptir. VGG16 ve VGG19 modelleri, 3x3 filtre boyutuna sahip konvolüsyon katmanları ve 2x2 filtre boyutuna sahip havuzlama katmanlarından oluşur (Russakovsky *et al.* 2015; Simonyan & Zisserman, 2014). AlexNet modeli ise 60 milyon parametreye sahiptir ve 3 konvolüsyon katmanı ile 2 tam bağlı katman içerir (Krizhevsky *et al.* 2017). GPSCNN’lerde yer alan Özel CNN tabanlı modellerden, CNN1 modeli, 2 konvolüsyon katmanı, 2 ortalama havuzlama katmanı, 1 düzleştirme katmanı ve 5 tam bağlı katmandan oluşur. İlk iki konvolüsyon katmanında 256 filtre kullanılmıştır ve bırakma katmanı (dropout) oranı %30’dur. CNN2 modeli, 1 konvolüsyon katmanı, 1 maksimum havuzlama katmanı, 1 düzleştirme katmanı ve 4 tam bağlı katmandan oluşur. Konvolüsyon katmanında 16 filtre bulunmaktadır ve bırakma katmanı oranı %10’dur. CNN3 modeli ise 4 konvolüsyon katmanı, 4 toplu normalizasyon katmanı, 2 maksimum havuzlama katmanı, 1 düzleştirme katmanı ve 3 tam bağlı katmandan oluşur. İlk iki konvolüsyon katmanında 32 filtre, son iki konvolüsyon katmanında ise 64 filtre kullanılmıştır ve bırakma katmanı oranı %50’dir (Aydin & Erdem, 2023).

GPSLSTM modeli

GPSLSTM, LSTM ağına dayalı bir modeldir. LSTM modeli, bir LSTM katmanı ve bir tam bağlı katman ile tasarlanmıştır. LSTM katmanında “tanh” aktivasyon fonksiyonu kullanılmıştır. Modelin özellikleri arasında LSTM katmanında 256 nöron, tam bağlı katmanında ise 1 nöron bulunmaktadır. Optimizasyon fonksiyonu olarak “Adam” kullanılmış ve öğrenme oranı 0.001 olarak belirlenmiştir.

GPSCNNs ve GPSLSTM çalışma prensibi

GPSCNNs, GPS verilerindeki doğruluğu artırmak için CNN tabanlı bir yapı kullanır. GPS verileri, bir boyutlu dizilerden dört boyutlu tensörlere dönüştürülerek CNN modellerine giriş olarak verilir. CNN modelleri, konvolüsyon katmanlarında mekânsal özellikleri çıkarır, havuzlama katmanlarında boyutları küçültür ve tam bağlı katmanlarda bu bilgiyi işleyerek anlamlı tahminler üretir. CNN tabanlı modellerden elde edilen bu çıktı, daha sonra bir meta model olan SVR ile birleştirilir. SVR, bu çıktıların doğruluğunu artırır ve GPS konumlandırma tahmini yapar. GPSLSTM modeli, GPS verileri arasındaki ardışık düzenleri ve ilişkileri öğrenmek için tasarlanmıştır. Ancak, çalışmada zaman bilgisi kullanılmadığından, model zamansal bağımlılıkları değil, GPS verileri arasındaki yapısal ve konumsal ilişkileri öğrenir. GPS verileri, LSTM modeline girdikten sonra, LSTM katmanı bu verilerdeki ardışık düzenleri işler ve özellikleri çıkarır. LSTM'nin çıktısı, tam bağlantılı bir katman aracılığıyla işlenir ve SVR meta modeline giriş olarak verilir. SVR, LSTM'den gelen çıktıyı optimize ederek nihai konum tahmini sağlar. SVR modeli, *Radyal Tabanlı Fonksiyon (RBF)* çekirdeği ile yapılandırılmıştır. Modelde *gamma* değeri 0,12, *C* parametresi 1,7 ve *epsilon* değeri 0,11 olarak belirlenmiştir.

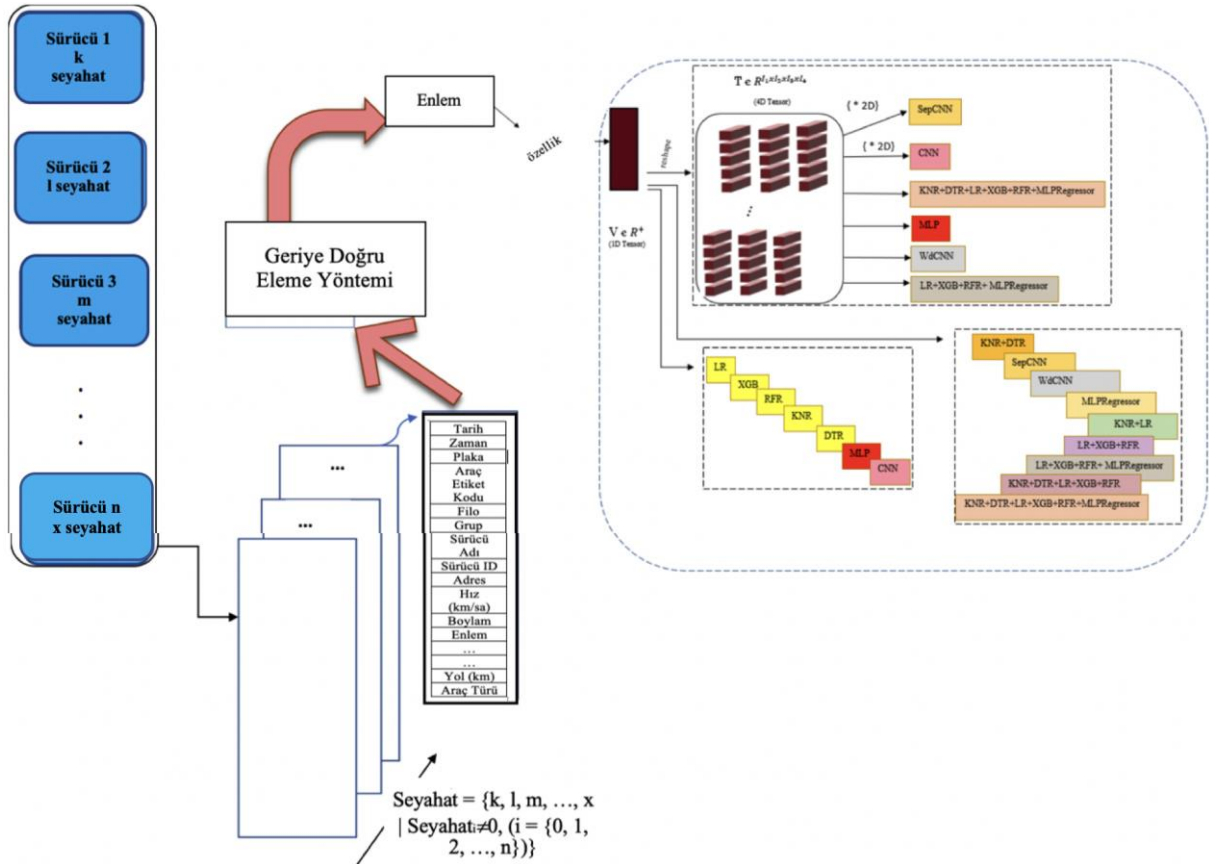
Tezin ikinci aşamasında, model seçim sürecinde; veri setinin yapısal özelliklerine en uygun modellerin belirlenmesi ve konum doğruluğunun artırılması amaçlanmıştır. Yüksek boyutluluk, eksik değerlerin varlığı ve GPS verilerinin karmaşıklığı gibi zorluklarla başa çıkabilecek modeller tercih edilmiştir. ML modellerinden, Aşırı Gradyan Artırma (XGB) modelinin seçilme nedenleri; heterojen ve yüksek boyutlu veri setlerini verimli bir şekilde işleyebilme kapasitesine sahiptir. Dahili düzenleme (regularization) mekanizmaları sayesinde aşırı öğrenmeyi önler. Büyük veri setlerinde hızlı tahmin yapma ve doğruluk sağlama avantajı sunar. Uygulamada GPS verileri üzerinde üstün performansı kanıtlanmıştır. DTR, KNR ve Rastgele Orman Regresyonu (RFR) uygulanma nedenleri ise, daha hızlı eğitim süreleri ve düşük hesaplama maliyetleri nedeniyle seçilmiştir. Bu modeller, topluluk modellerine entegre edilerek bireysel modellerin eksiklikleri giderilmiş ve genel doğruluk artırılmıştır. DL modellerinden, CNN seçilme nedenleri; mekânsal özellikleri yakalama kapasitesi nedeniyle seçilmiştir. GPS verilerinin 4B tensörlere dönüştürülmesiyle bu modellerin performansının artırılacağı öngörülmüştür. Ayrılabilir Evrişimsel Sinir Ağı (SepCNN) ve Geniş ve Derin CNN (WdCNN) modellerinin seçilme nedenleri; farklı katman yapıları sayesinde daha karmaşık veri ilişkilerini modelleme yeteneği sunar.

Çalışmada uygulanan yöntemlerin genel bir özet diyagramı Şekil 31'de verilmiştir. Diyagramda üç aşama bulunmaktadır. Birinci aşamada, 4B GPS verisi yapay sinir ağları içeren

modellere girdi olarak verilmiştir (CNN, SepCNN, WdCNN, Çok Katmanlı Algılayıcı (MLP), LR-XGB-RFR-Çok Katmanlı Algılayıcı Regresyonu (MLPRegressor), K-En Yakın Komşu Regresyonu (KNR)-DTR-LR-XGB-RFR-MLPRegressor). İkinci aşamada, 1B GPS verisi ML modellerine (LR, XGB, RFR, KNR, DTR, MLP, CNN) girdi olarak verilmiştir. Son aşamada ML tabanlı topluluk modelleri ve DL tabanlı hibrit modellere 1 boyutlu GPS verileri verilmiştir.

MLP modeli, bir giriş vektörünü alarak gizli katmanlardan geçirir ve tahmin sonucunu çıktı katmanında üretir. Modelde, her gizli katman giriş verilerini ağırlıklar ve bias değerleri ile işler. Gizli katmanlarda ReLU aktivasyon fonksiyonu kullanılırken, çıktı katmanında lineer bir aktivasyon fonksiyonu uygulanmıştır. Modelin gizli katmanında 30 nöron, son çıktı katmanında ise 1 nöron bulunmaktadır. Çalışmada, MLP'nin regresyon versiyonu olan MLPRegressor, scikit-learn kütüphanesi kullanılarak uygulanmıştır. CNN modelleri, tensör işlemlerinde mekânsal özellikleri çıkarmak için filtre (kernel) kullanır. CNN'ler hem tek boyutlu tensörler (Conv1D) hem de iki boyutlu tensörler (Conv2D) üzerinde çalışabilir. Conv1D, veri boyunca bir boyutta hareket ederek mekânsal ilişkileri öğrenirken, Conv2D, iki boyutta hareket ederek daha karmaşık mekânsal ilişkileri öğrenir. Dolayısıyla, Conv1D tek boyutlu GPS verileri ile çalışırken, Conv2D GPS verilerinin 4B tensöre dönüştürülmüş haliyle çalışır. Çalışmada önerilen CNN modeli, bir konvolüsyon katmanı (3x3 filtre boyutunda), bir maksimum havuzlama katmanı (2x2 filtre boyutunda), bir düzleştirme katmanı ve iki tam bağlı katman içermektedir. İlk tam bağlı katmanda 30 nöron, son katmanda ise 1 nöron bulunmaktadır. SepCNN modeli, CNN'lerin bir çeşididir ve konvolüsyon işlemini iki aşamada gerçekleştirir: derinlik konvolüsyonu ve nokta konvolüsyonu. Derinlik konvolüsyonu, her renk kanalını ayrı ayrı işlerken, nokta konvolüsyonu bu sonuçları birleştirir. Çalışmada kullanılan SepCNN modeli, bir ayrılabilir konvolüsyon katmanı (7x7 filtre boyutunda), bir maksimum havuzlama katmanı (5x5 filtre boyutunda), bir global maksimum havuzlama katmanı ve bir tam bağlı katmandan oluşmaktadır. Son katmanda 1 nöron bulunmaktadır. WdCNN, geniş ve derin katmanları birleştiren bir hibrit DL yapısıdır. Model, geniş katman üzerinden genel özellikleri öğrenirken, derin katman üzerinden karmaşık ilişkileri anlamaya çalışır. Çalışmada kullanılan WdCNN modeli, iki farklı giriş katmanı ile başlamaktadır: biri geniş, diğeri derin katman içindir. İlk gizli katmanda 30 nöron ve bir düzleştirme katmanı, ikinci gizli katmanda ise 30 nöronlu iki tam bağlı katman ve bir düzleştirme katmanı bulunmaktadır. Bu katmanların çıktıları, bir birleştirme (concatenate) katmanında birleştirilmiş ve son olarak bir tam bağlı katmanda tek bir çıktı üretilmiştir. Topluluk modeller, farklı ML algoritmalarının tahminlerini birleştirmek için kullanılmıştır. Tekil (ayrık) modellerin eksikliklerini gidermiş ve genel doğruluğu artırmıştır. Ortalama oylama yöntemiyle tahminler birleştirilmiş ve çeşitli kombinasyonlar, özellikle heterojen modellerle uygulanmıştır. Kullanılan topluluk modelleri;

KNR-DTR-LR-XGB-RFR-MLPRegressor, LR-XGB-RFR-MLPRegressor, LR-XGB-RFR, KNR-DTR-LR-XGB-RFR, KNR-DTR ve KNR-LR kombinasyonlarıdır.



Şekil 31. ML ve DL yöntemleriyle sürücü seyahat verileri için özellik seçimi ve tahmin modelleme çerçevesi (Erdem *et al.* under review e)

Optimizasyon modellerinin ÇDARP için geliştirilmesi

Problem tanımı

ÇDARP, matematiksel formülasyonu detaylı bir şekilde Tablo 17’de açıklanmıştır. Modelin temel parametreleri, kısıtları, çözüm yöntemleri vb. verilmiştir. ARP, müşteri taleplerini en düşük maliyetle karşılamak için araçların rotalarını optimize etmeyi amaçlar. Bu problemde:

- **C:** Hizmet verilecek müşteri setidir. $C=\{1,2,...,m\}$
- **S:** Şirket ürünlerini satan tedarikçi setidir. $S=\{1,2,...,z\}$
- **D:** Depolama alanlarının setidir. $D=\{1,2,...,n\}$
- c_{qw} : q ve w arasındaki rota maliyetidir.

Amaç fonksiyonu, toplam maliyeti (I) minimize etmeyi hedefler. Maliyet, sevkiyat işlem süresi olarak tanımlanmıştır. Her müşteri, bir veya birden fazla araçla hizmet alabilir. Eğer bir müşterinin toplam talebi bir araçla karşılanamazsa, talep birden fazla turla tamamlanır. İkitelli’de bulunan merkezi depo, Çatalca, Trakya ve Maltepe’deki diğer depolara ürünlerin

dağıtımı için ana dağıtım merkezi olarak işlev görür. Her rota bir depodan başlamalı ve aynı depoda bitmelidir.

Kısıtlar;

- ✓ Araç Kapasite Kısıtı: Araçların taşıma kapasitesi, müşteri taleplerini karşılamak için yeterli olmalıdır.
- ✓ Rota Sürekliliği: İki depo ardışık olarak aynı rotada yer alamaz.
- ✓ Depo Konum Kısıtlaması: Depolar, rotanın ikinci veya sondan ikinci pozisyonunda yer almamalıdır. Araçların başlangıç noktaları depolar olmalıdır.

Zaman-mesafe matrisi, OSRM API ve GPS verileri kullanılarak iki farklı yöntemle oluşturulmuştur. OSRM API, düğümlerin coğrafi koordinatlarını kullanarak düğümler arasındaki en kısa süreyi *API_Call* ile hesaplar. GPS tabanlı yöntem, gerçek zamanlı verilerden faydalanarak düğümler arasındaki seyahat sürelerini hesaplar. Her iki düğüm arasındaki süre, varış ve çıkış zamanları arasındaki fark alınarak hesaplanır.

Tablo 17. Problemin Matematiksel Formülasyonu (Erdem *et al.* under review b)

ÇDARP Probleminin Parametreleri, Değişkenleri, Kısıtları ve Amaç Fonksiyonu
<i>Parametreler ve Kümeler</i>
C: Müşteri kümesi, $C = \{1, 2, \dots, m\} \mid m$ müşteri sayısı
S: Tedarikçi kümesi, $S = \{1, 2, \dots, z\} \mid z$ tedarikçi sayısı
D: Depo kümesi, $D = \{1, 2, \dots, n\} \mid n$ depo sayısı
V: Düğüm kümesi, $V = C \cup S \cup D$
A: Yönlü kenarlar kümesi, $A = \{(q, w) \mid q \in V, w \in V, q \neq w\}$
o_{V_q} : V_q düğümündeki sipariş (talep) miktarı $\mid q \in (C \cup S) \setminus D$
c_{qw} : V_q düğümünden V_w düğümüne hareketin maliyeti (zaman-temelli mesafe)
$(T_d)_{qw}$: V_q ve V_w arası zaman-mesafe matrisi
T_m : Zaman-mesafe matrisi
G: GPS koordinatları, $G = \{(lat_1, lon_1), (lat_2, lon_2), \dots, (lat_r, lon_r)\} \mid r \in V$
<i>Araç ve Palet Parametreleri</i>
T: Taşıma araçları kümesi
S: Satış grubu araçları kümesi
D_s : Dağıtım araçları kümesi
D_p : Depo araçları kümesi
F: Filo kümesi, $F = T \cup S \cup D_s \cup D_p$
Q: Araç kapasitesi
Ψ : Rota kümesi, $\Psi = \{V_q, V_a, \dots, V_w\}$

Tablo 17. (devamı)

p : Palet tipleri, $p = \{26, 14, 8, 6, 4, 2\}$

n_p : p tipinde palet taşıyan araçların sayısı

Karar Değişkenleri

Γ : Toplam maliyet

x_{qw} : İkili karar değişkeni:

- $x_{qw} = 1 \rightarrow$ Eğer V_q düğümünden V_w düğümüne hareket varsa
- $x_{qw} = 0 \rightarrow$ diğer durumlar

Amaç Fonksiyonu

Toplam maliyeti minimize et:

$$\min_{\Gamma} = \sum_{w \in V} \sum_{q \in V} x_{qw} \cdot c_{qw} \quad (1)$$

Kısıtlar

1. Araç Kapasite Kısıtı:

$$\sum_{w \in V} \sum_{q \in V} x_{qw} \cdot o_{V_q} \leq Q_v, \forall \{q, w\} \in V, q \neq w, \forall v \in F \quad (2)$$

2. Herhangi bir araç, başladığı düğümde rotasını sonlandırmamalıdır:

$$\sum_{q \in V} x_{qq} = 0, \forall q \in V \quad (3)$$

3. Araç çıkış ve varış dengesi:

$$\sum_{q \in V} x_{dq} \geq \sum_{q \in V} x_{wq}, \forall d \in V_D \text{ ve } \forall w \in V_C, V_D \subseteq D, V_C \subseteq C \quad (4)$$

4. Rota sürekliliği:

$$\Psi_i \neq \Psi_{i+1}, \forall_i \in V \text{ ve } \Psi_i \in D \quad (5)$$

5. Depo kısıtları:

$$\Psi_i \in D, \forall_i \in V, i \neq 2 \text{ ve } i \neq n-2 \quad (6)$$

Zaman-Mesafe Matrisi Oluşturulması

1-OSRM API:

$$(T_d)_{qw} = \text{API_Call}(lon_q, lat_q, lon_w, lat_w) \quad (7)$$

$$T_m = \begin{bmatrix} T_{d11} & \cdots & T_{d1r} \\ \vdots & \ddots & \vdots \\ T_{dr1} & \cdots & T_{drr} \end{bmatrix} \quad (8)$$

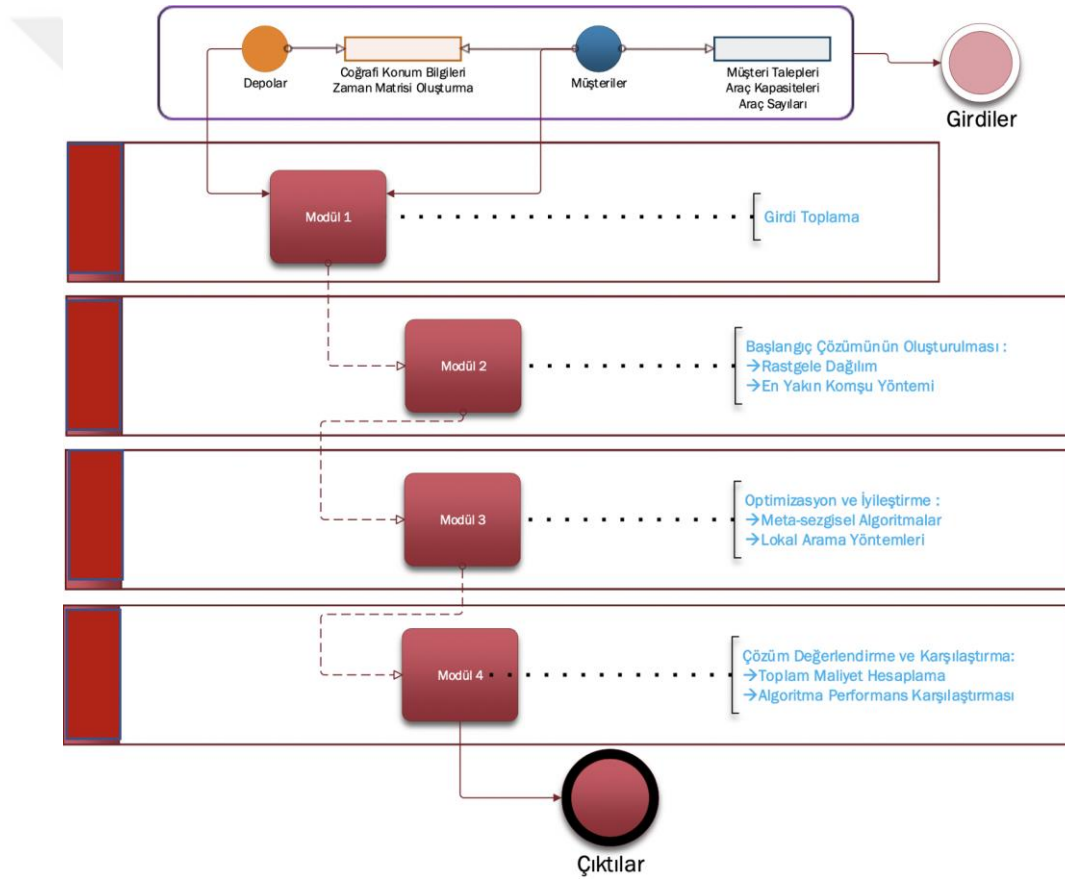
2-GPS:

$$(T_d)_{qw} = \text{time}_q(lat, lon) - \text{time}_w(lat, lon) \quad (9)$$

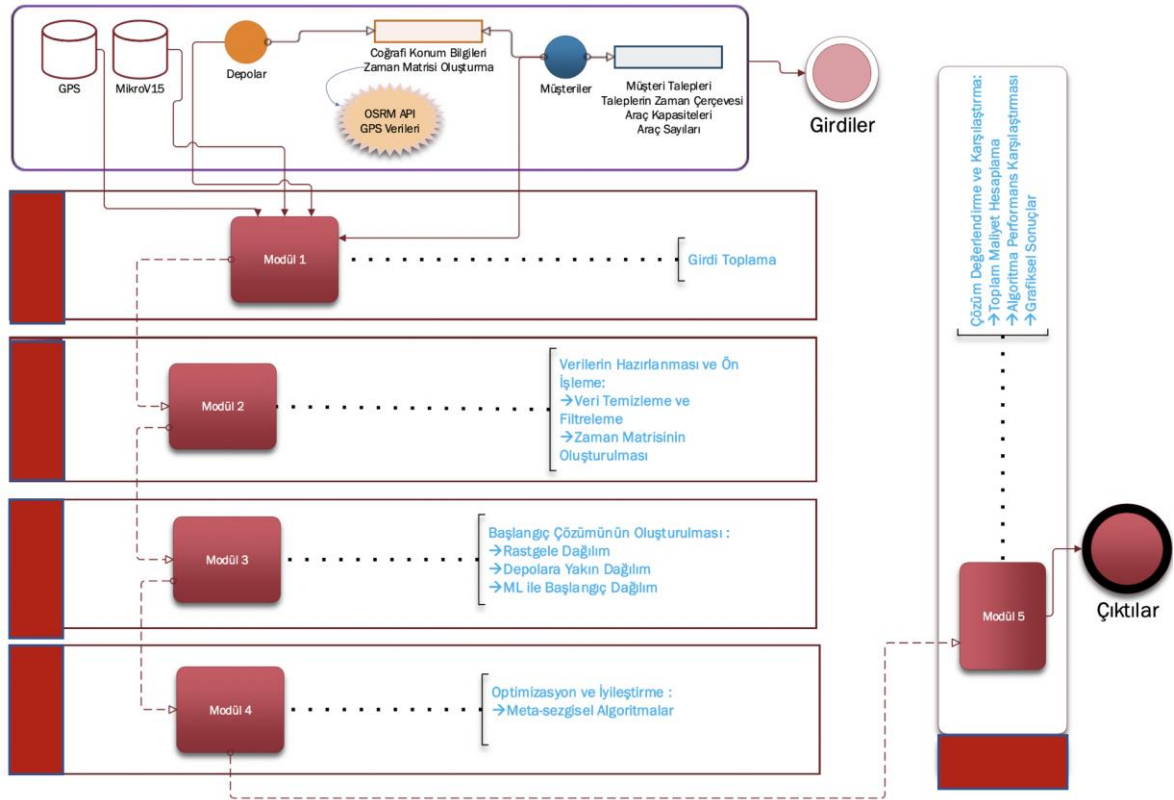
$$T_m = \begin{bmatrix} T_{d11} & \cdots & T_{d1r} \\ \vdots & \ddots & \vdots \\ T_{dr1} & \cdots & T_{drr} \end{bmatrix} \quad (10)$$

Çözüm yöntemleri

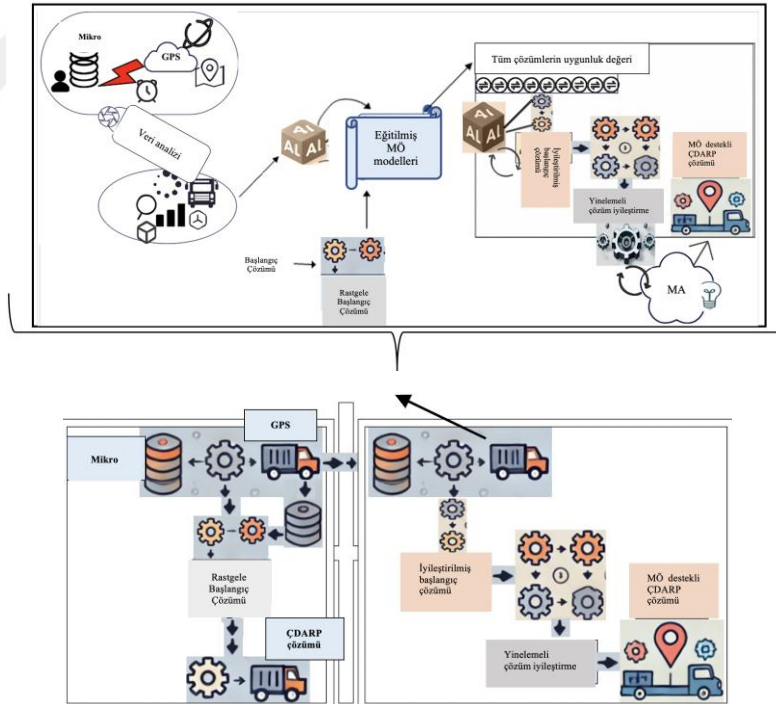
ÇDARP çözümüne yönelik iki farklı yaklaşım karşılaştırıldı: Geleneksel meta-sezgisel algoritmalar ile çözülen ÇDARP ve ML Entegrasyonu ile çözülen ÇDARP (Şekil 32 ve Şekil 33). Geleneksel meta-sezgisel algoritmalar, genellikle rastgele yöntemlerle başlangıç çözümleri oluşturur. Bu çözümler, çözüm uzayında herhangi bir noktada bulunabilir ve iyi bir sonuç elde etmek için algoritmanın çok sayıda yineleme yapması gerekir. Bu süreç, daha fazla hesaplama süresi ve deneme-yanılma yöntemiyle karakterizedir. Öte yandan, ML entegrasyonu, tarihsel verilere dayanarak daha verimli ve doğru başlangıç çözümleri sağlar. ML, meta-sezgisel algoritmalarla başlangıçta daha kaliteli çözümler sunarak çözüm sürecini hızlandırır ve genel çözüm kalitesini artırabilir. ML tarafından oluşturulan bu çözümler, meta-sezgisel algoritmalar tarafından daha da iyileştirilerek en iyi veya en uygun çözüme ulaşılmasını sağlar (Şekil 34).



Şekil 32. Geleneksel ÇDARP çözümü



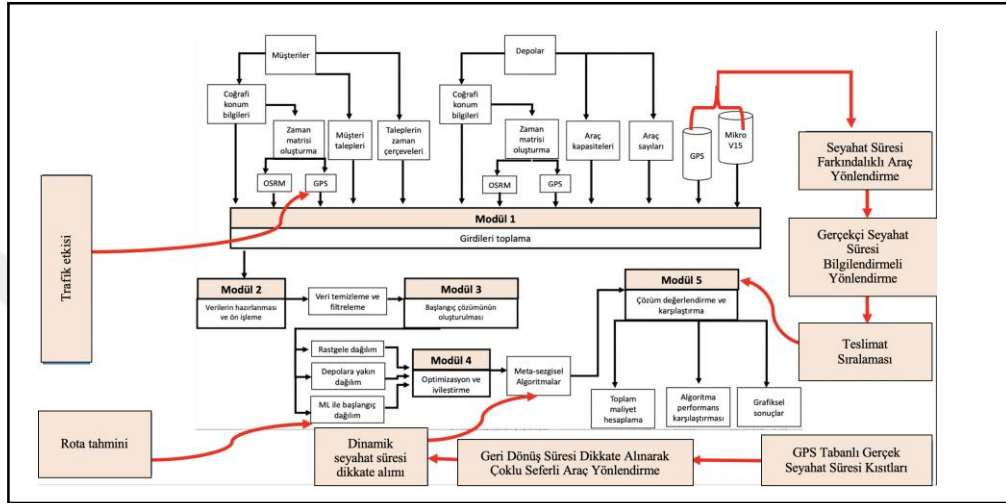
Şekil 33. Önerilen ÇDARP çözümü



Şekil 34. ÇDARP çözüm yaklaşımı (Erdem et al. under review b)

Önerilen yaklaşım, klasik çözümlerin ötesine geçerek gerçek GPS verileri ve araçların gerçek zamanlı ve yüksek hassasiyetli takibiyle desteklenen bir şehir içi dağıtım optimizasyonu çerçevesi sunmaktadır (Şekil 35). Bu yapıda, müşteri ve depo konumları, araç kapasiteleri, zaman ve geçmiş rota bilgileri gibi çoklu kaynaklardan gelen veriler bir araya getirilerek

kapsamlı bir veri havuzu oluşturulmuştur. Modüler yapıyla geliştirilen çözüm sisteminde, veri ön işleme (Modül 2), başlangıç çözümü oluşturma (Modül 3) ve optimizasyon (Modül 4) adımlarının her biri, gerçek zamanlı ve tarihsel veri bütünleşmesiyle desteklenmektedir. Teslimat seyahat sürelerinin dinamik şekilde hesaba katılması, araçların dönüş sürelerinin yönlendirmeye entegre edilmesi ve GPS tabanlı kısıtlarla algoritmanın beslenmesi, rota tahmininin doğruluğunu artırmaktadır. Ayrıca, dağıtım sıralaması ve seyahat süresi farkındalığı gibi çıktılar sayesinde operasyonel verimlilik önemli ölçüde iyileştirilmektedir.

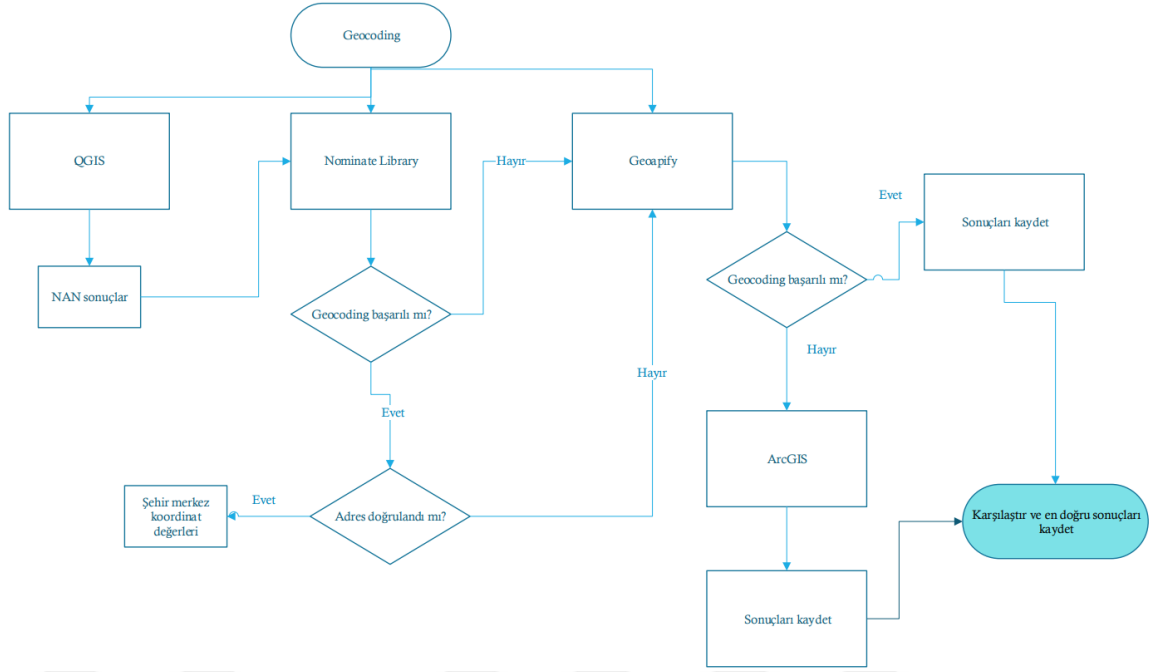


Şekil 35. Önerilen ÇDARP modelinde teslimatların entegrasyonu ve etki alanları

ÇDARP çözüm sürecinin temel adımlarından biri olan konum verilerinin işlenmesi, zaman-mesafe matrisi oluşturma, algoritmaların uygulanması, başlangıç çözümü geliştirme stratejileri, ürünlerin araçlara yerleştirilmesi vb. aşamalar incelendiğinde;

✓ Aşama I: Veri Toplama, Veri İşleme ve Hazırlık Aşaması

ÇDARP problemini çözmek için ilk adım, müşteri konum verilerinin (enlem ve boylam bilgileri) toplanması ve işlenmesidir. Bu veriler, zaman-mesafe matrisini doldurmak için gereklidir. Bu amaçla, müşterilerin 2013-2023 yılları arasında verdiği siparişlere ait fiziksel adres bilgileri, coğrafi koordinatlara dönüştürülmüştür. Bu dönüşüm işlemi için coğrafi kodlama (geocoding) yöntemi kullanılmıştır. Coğrafi koordinatların elde edilmesinde kullanılan farklı programlar arasında yapılan deneyler, ArcGIS API'nin diğer programlara kıyasla en güvenilir, en doğru ve en kapsamlı olduğunu göstermiştir (Şekil 36).



Şekil 36. Müşteri ve depo konumları için coğrafi kodlama süreci (Erdem *et al.* under review b)

Sipariş veren müşterilerin, konum, sipariş miktarları, sipariş tarihi ve sipariş saati gibi bilgileri sonraki aşamalarda kritik öneme sahiptir. Bu nedenle, Şekil 28’de gösterilen Sevkiyat Veri Modeli ve coğrafi kodlama işlemleri (Şekil 36) uygulanmıştır. Bu işlemler sonucunda, veriler veri çerçevesine (dataframe) kaydedilmiştir. Veri çerçevesinin genel bir kesiti ise Tablo 18’de sunulmuştur.

Tablo 18. Sevkiyat Veri Modeli ve Coğrafi Kodlama Sonrasında Elde Edilen Veri Çerçevesinin Özeti (Erdem *et al.* under review b)

Tarih	Saat	Müşteri Kodu	City	İlçe	x_koordinatı	y_koordinatı	Talep miktarı (palet cinsinden)
31.03.2023	16:34	120.10.100.13688	İstanbul	Güngören	41.0186	28.8643	160
31.03.2023	16:34	120.10.100.13688	İstanbul	Güngören	41.0186	28.8643	45
...							
31.03.2023	13:53	120.10.100.15335	İstanbul	Gaziosmanpaşa	41.0726	28.8952	264
31.03.2023	14:03	120.01.01.15736	İstanbul	Sultangazi	41.0994	28.8642	225
31.03.2023	17:04	120.10.100.15527	İstanbul	Bağcılar	41.0606	28.839	40
...							

✓ Aşama II: Zaman-Mesafe Matrisi Oluşturma

Zaman-mesafe matrisi, OSRM API ve GPS verileri kullanılarak iki farklı yöntemle oluşturulmuştur. Oluşturulan matrisler, düğümler arasındaki süre bilgilerini içerir ve rota planlamasının temelini oluşturur. Bu aşamada, OSRM API kullanılarak teorik ve ideal zaman-mesafe bilgisi elde edilmiştir. Buna karşılık, GPS verileri gerçek dünyadaki koşulları ve araç

hareketlerini dikkate alarak daha pratik bir veri sağlamıştır. Uygulanan yöntemlere ait detaylı bilgiler aşağıda verilmiştir.

OSRM API, düğümler (müşteri veya depo konumları) arasındaki en iyi rotayı bulmak için OpenStreetMap verilerini kullanır. HTTP üzerinden standart bir genel API yapısı kullanılarak istekler yapılır. OSRM API'si için temel kullanılan parametre detayları Tablo 19'da verilmiştir. Bu çalışmada, "araba" seyahat modu seçilmiştir. İsteklere verilen yanıtlar JSON formatında alınmıştır.

Mesafe matrisi, düğümler arası seyahat sürelerini içerir. Her bir düğüm satır olarak "başlangıç" (origin) ve diğer satırlar "hedef" (destination) olarak ele alınır. JSON yanıtlarından alınan mesafe verileri, bir mesafe matrisi oluşturmak için işlenmiştir. Matrisin her bir noktası, düğümler arasındaki seyahat sürelerini temsil etmiş ve bu bilgiler bir sözlük yapısında düzenlenmiştir (Tablo 20).

Tablo 19. OSRM API Parametre Detayları (Erdem *et al.* under review b)

OSRM API		Sonuç: n_{∞} düğümünden n_{∂} düğüme olan seyahat süresi (sn cinsinden)
Parameter	Kullanılan Parametre Değeri	
servis	route	$(n_{\infty}(\infty, \text{müşterinin konumu}), n_{\partial}(\partial, \text{müşterinin konumu})): (x_{\partial}, y_{\partial}), x_{(\infty \partial)}: \text{enlem}, y_{(\infty \partial)}: \text{boylam}$
versiyon	v1	
profil (seyahat modu)	car	
zaman aşımı	30000	
genel bakış	false	

Tablo 20. OSRM API ile Zaman-Mesafe Matrisi Oluşturulması (Erdem *et al.* under review b)

Algoritma: OSRM API tabanlı zaman-mesafe matrisi algoritması

```

1: satırlar ← read_csv (veri_seti)
2: satırlar ← veri_seti.drop_duplicates([x( $\infty|\partial$ ), y( $\infty|\partial$ )])
3: for i, originRow in satırlar:
4:   for j, destRow in satırlar:
5:     r ← session.get
6:     routes ← r.json()
7:     distanceMatrix[i][j] ← {'duration': routes[duration]}
8: for i in range(len(distanceMatrix)):
9:   for j in range(len(distanceMatrix)):
10:    if i != j and distanceMatrix[i].get(j):
11:      nodes.add(i)
12:      nodes.add(j)
13: for node in nodes:
14:   nodeCoordinates[node] ← ([x( $\infty|\partial$ ), y( $\infty|\partial$ )])
15: for node in nodes:
16:   latitude, longitude ← nodeCoordinates[node]
17:   distanceMatrix.at[node, 'enlem'] ← enlem
18:   distanceMatrix.at[node, 'boylam'] ← boylam

```

GPS verilerinin işlenmesi için Mobiliz Araç Takip Sistemi kullanılmıştır. Mobiliz, araç hareketlerini izlemek, müşteri adreslerini belirlemek, adres benzerliği ve veri eşleştirme işlemlerini gerçekleştirmek, ayrıca zaman farklılıklarını hesaplamak için kullanılan bir GPS tabanlı araç takip sistemidir. Sistem, iki yıldan fazla süredir teslimat verilerini kayıt altına almıştır. Tablo 22’de yalancı kod ile gerçekleştirilen işlem basamakları, Mobiliz’den temin edilen GPS verilerinin müşteri ve depo adresleriyle doğru şekilde eşleştirilmesini, adresler arasındaki zaman farklarının hesaplanmasını ve bu bilgilerin zaman-mesafe matrisine eklenmesini kapsamaktadır. Paralel işleme ve Cosine benzerliği gibi yöntemlerle işlem performansı artırılmış ve müşteri konumlarının doğruluğu sağlanmıştır. Bu bulgular, rota optimizasyonu ve lojistik süreçlerin iyileştirilmesi için kullanılmıştır.

✓ Paralel İşleme:

Bu süreçte, maksimum 8 çekirdekli işlem havuzu ("ThreadPoolExecutor") kullanılmıştır. Veri setindeki her 10,000 kayıt bir "bölüt" olarak ayrılmış ve bu parçalar "concat()" fonksiyonu ile birleştirilmiştir (Erdem *et al.* under review b).

✓ Müşteri Adreslerinin Belirlenmesi:

GPS verileri, araçların geçtiği noktaları kaydederek müşteri adreslerini doğrulamak ve belirlemek için kullanılmıştır. Ancak, kaydedilen tüm noktalar müşteri adresi değildir. Doğru müşteri adreslerini tespit etmek için, GPS verileri, Mikro V15’den alınan adreslerle karşılaştırılmıştır. GPS veri setinde yer alan fiziksel adres bilgileri, şirket tarafından sağlanan referans bilgiler doğrultusunda filtrelenerek depoların konumları tespit edilmiştir. Bu bağlamda, "Adres" öz niteliğinde yer alan ve “Şehit Ümit Alay Sk.”, “Cemal Ulusoy Cd.”, “Ali Osman Çelebi”, “Büyükbakkalköy Mh.”, “Karacaköy Merkez Mh.” ve “Karaca Ahmetpaşa Cd.” ifadelerini içeren tüm kayıtların, depolara ait olduğu belirlenmiştir. Bu amaçla, "depoRowsDf" veri çerçevesinde plaka bilgilerine göre bir filtreleme işlemi gerçekleştirilmiştir. Elde edilen plaka bilgileri, her biri benzersiz olacak şekilde düzenlenmiştir. Araçlar depodan ayrıldığında, sevkiyatlarını müşteri adreslerine teslim etmektedir (Erdem *et al.* under review b).

✓ Araç Hareketlerinin İzlenmesi ve Analizi:

GPS verileri, araç hareketlerinin izlenmesi, analiz edilmesi, durma sürelerini ve genel rotaları belirlemek için kullanılmıştır. Araçların duraklama süreleri ve genel rotaları analiz edilmiştir. Araçların uzun süre hareketsiz kaldığı durumlar tespit edilmiş ve bu tür veriler analizden çıkarılmıştır. Bu durum, operasyonel verimliliği yanlış değerlendirmemek için önemli bir adımdır. Depodaki araçlarda bulunan GPS cihazı, aracın hareketsiz olduğu durumlar,

kontak açık veya kapalı olduğu anlar, periyodik mesajlar, aracın rölantide çalışmaya başladığı zamanlar, ani hareketler ve kontrol mesajları gibi çeşitli veri türlerini kaydetmektedir. Ancak GPS veri setinde, araç konumunun uzun süre değişmemesi istenmeyen bir durum olarak değerlendirilmektedir. Bu tür sabit verilerin veri setine (samePlatesDf) dahil edilmesi, araç aktivite desenlerinin yanlış anlaşılmasına ve operasyonel verimlilik analizlerinin yanıltıcı sonuçlar vermesine neden olabilir. Bu nedenle, yalancı kodun 22. satırında, yalnızca gerçek araç hareketlerini yansıtan doğru ve ilgili verilerin kullanılmasını sağlamak amacıyla gerekli düzenlemeler yapılmıştır. Sonuçta, araçların aktif bir şekilde rotalarında ilerlediği veya teslimat yaptığı durumlara odaklanılmıştır. Ayrıca, çalışmada adreslerin Mikro V15 veri tabanı ile eşleştirilmesi için zaman damgalı GPS verileri kullanılmıştır (Erdem *et al.* under review b).

✓ Adres Benzerliği ve Veri Eşleştirme:

GPS cihazları, araç hareketi sırasında birçok adres kaydeder. GPS veri setinde plaka numarası “X” olan bir aracın gün boyunca baz istasyonu, cami, otobüs durağı ve hastane gibi farklı noktalardan geçtiği veri setinde gözlemlenmiştir. Tutarlı bilgi veya format eksikliği olan müşteri adreslerini doğru bir şekilde belirleyebilmek için, kaydedilen GPS konumlarının Mikro V15’de saklanan müşteri adresleriyle karşılaştırılması gerekmektedir. Adres benzerlik kontrolleri sırasında, GPS veri formatı ile Mikro V15’deki adres formatının doğrudan uyumlu olmaması nedeniyle zorluklar yaşanmıştır. Bu sorunu çözmek amacıyla, veri setleri arasındaki ilişkileri değerlendirmek için bir benzerlik matrisi oluşturulmuştur. Benzerlik matrisinde yüksek değerler, her iki veri setindeki adresler arasında güçlü bir eşleşme olduğunu göstermektedir. Bu kontroller sırasında işlem verimliliğini artırmak amacıyla çoklu işlem ve iş parçacığı (threading) yöntemleri uygulanmıştır. Cosine benzerliği, Jaccard ve Pearson gibi birçok yöntem test edilmiş olmasına rağmen, Cosine benzerliği en doğru sonuçları sürekli olarak sağlamıştır. Bu yöntem, adres eşleşmelerini ve tanımlamalarını daha doğru bir şekilde gerçekleştirmiş ve müşteri konumlarının belirlenmesinde güvenilirliği artırmıştır (Erdem *et al.* under review b).

✓ Zaman Hesaplamaları ve Zaman-Mesafe Matrisi:

Benzerlik oranı %30 olarak belirlendiğinde, birçok adresle ilgili bilgi benzerlik göstermiştir. Ancak, bu sonuçlar tutarsızdır. Bu tutarsızlığın, adreslerdeki kelimelerin yazım, kısaltma veya format farklarını dikkate almayan doğrudan eşleştirme yöntemiyle yapılan kelime kelime karşılaştırmadan kaynaklandığı düşünülmektedir. Bu tür bir doğrudan eşleştirme, veri girişleri arasında küçük farklılıklar olduğunda yanlış sonuçlara yol açabilmektedir. Benzerlik oranı %60-70 olarak belirlendiğinde ise çok az veri eşleşmesi sağlanmıştır. Bu nedenle, benzerlik oranı %50 olarak optimize edilmiştir. Cosine benzerliği

yöntemi uygulanarak, Mikro V15 veri setindeki “fullAdres” alanı ile GPS veri setindeki “Address” değerleri arasındaki eşleşmeler %50 benzerlik oranına göre değerlendirilmiştir. En yüksek benzerlik değerine sahip adresin belirlenmesi için GPS verisi kullanılmıştır. Bu adres, “data” listesine eklenmiştir. “Data” içeriği, Tablo 21’de örnek formatta sunulmuştur. Araçların depo ile müşteriler arasındaki yolculuk süreleri zaman farkı (“timeDifference”) olarak kaydedilmiştir. Zaman farkı, aynı araç plakasına ait kayıtlar için toplanmış, ancak plaka değiştiğinde sıfırlanmıştır. Böylece, depodan müşteriye toplam seyahat süresi, aynı plaka için kaydedilen zaman farkı değerlerinin toplamı olarak belirlenmiştir. GPS verileri sayesinde, araçların belirli lokasyonlar arasında ne kadar süre yolculuk yaptığını ölçülmüştür. Dolaylı olarak, müşterilerin birbirlerine göre konumları arasındaki süre değerleri de hesaplanmıştır. Hesaplanan bu süre bilgileri zaman-mesafe matrisine eklenmiştir. Eğer GPS verilerinde bir müşteriye ait bilgi bulunmuyorsa, bu müşterinin diğer konumlara göre süre bilgisi zaman-mesafe matrisine eklenemez. Bunun temel nedeni, şirketin son 2-3 yıldır Mobiliz araç takip sistemini kullanıyor olmasıdır. Ancak sipariş veren müşterilere ait bilgilerin yer aldığı Mikro V15 veri seti son 10 yılı kapsamaktadır. Bu nedenle, müşteri geçmişte, özellikle daha eski yıllarda sipariş vermiş ve GPS verilerinde kaydı bulunmuyorsa, bu durum bir sorun oluşturmaktadır. Bu tür durumlarda, zaman-mesafe matrisindeki veri kaybını önlemek için OSRM API kullanılarak eksik süre bilgisi elde edilmiş ve matrise eklenmiştir.

Sonuçta, araçların depo ve müşteri konumları arasındaki toplam seyahat süreleri belirlenmiştir. GPS verileriyle belirlenen adresler arasındaki zaman farkları, müşteri ile depo veya müşteriler arasındaki göreceli mesafenin belirlenmesinde kullanılmıştır. Göreceli mesafe, ulaşım ve lojistik problemlerinde sıklıkla kullanılan bir ölçüttür ve gerçek yol koşulları ile seyahat sürelerini dikkate alarak rota optimizasyonu için hesaplanır. Bu yöntem, düz coğrafi mesafeden farklıdır çünkü yol durumu, hız sınırları, trafik yoğunluğu ve diğer çevresel faktörler gibi gerçek dünya koşullarını göz önünde bulundurmaz. Çalışmada, göreceli mesafe, konumlar arasındaki kat edilen yol mesafesi olarak ele alınmamıştır. Bunun yerine, sevkiyat işlemleri sırasında ürünlerin teslimatı için zaman parametresini kapsayan bir ölçüm yöntemi olarak ele alınmıştır (Erdem *et al.* under review b).

Tablo 21. Mikro V15 ve GPS Adres Benzerlik Sonucu Oluşturulan Veri Seti (Erdem *et al.* under review b)

plaka	...	enlem	boylam	zaman	
Plaka formatı 1				00:23:01	
Plaka formatı 1				08:42:47	
					Süre fark=Zaman formatı 2 - Zaman formatı 1
Plaka formatı 1				09:10:40	
Plaka formatı 1				09:10:59	
Plaka formatı 2				...	Süre fark=Zaman formatı 2 - Zaman formatı 1

Tablo 22. GPS Verileriyle Zaman-Mesafe Matrisi Oluşturulması (Erdem *et al.* under review b)**Algoritma: GPS Verisi ile Zaman-Mesafe Matrisi Oluşturma Algoritması**1: *Tablo 20, 1 ve 2. satırlar*

2: chunk_size←10000 & chunks, added_plakas, matched_indices, data←[]

3: depolar_adresleri←['Şehit Ümit Alay Sk.', 'Cemal Ulusoy Cd.', 'Ali Osman Çelebi', 'Büyükbakkalköy Mh.', 'Karacaköy Merkez Mh.', 'Karaca Ahmetpaşa Cd.']

4: concurrent.futures.ThreadPoolExecutor(cpu_counts=8):

5: for chunk in read_csv(GPS dataset, chunksize=chunk_size):

6: chunks.append(chunk)

7: df_gpsdata←concat(chunks)

8: for depo_adresi in depolar_adresleri:

9: deporows←df_gpsdata[df_gpsdata['Adres'].str.contains(depo_adresi)]

10: deporows_df←concat([deporows_df, deporows])

11: for index, row in deporows_df.iterrows(): #depolardan kalkan araçlar saptanıyor

12: plaka←satır['Plaka']

13: if plaka in added_plakas:

14: continue

15: added_plakas.add(plaka)

16: plaka_df←deporows_df[deporows_df['Plaka'] == plaka]

17: mergedPlaka_df←concat([mergedPlaka_df, plaka_df])

18: onceki_adres←None

19: for plaka_row in mergedPlaka_df.iterrows():

20: if plaka_row['Adres'] != onceki_adres:

21: yeni_satır←{plaka_row['Plaka'], 'Adres', 'Tarih', 'Zaman', 'Grup', 'Sürücü', 'Hız (km/sa)', 'Yol (km)', 'İleti Tipi' }

22: ayni_plakalar_df←ayni_plakalar_df.append(yeni_satır)

23: onceki_adres←plaka_row['Adres']

24: def calculate_similarity(X1, X2):

25: return cosine_similarity(X1, X2)

26: concurrent.futures.ThreadPoolExecutor(cpu_counts=8):

27: X1,X2←vectorizer.fit_transform(dfadresCapacities["fulladres"], ayni_plakalar_df["Adres"]).astype(str).tolist(),

28: cosine_similarities←calculate_similarity(X2, X1)

29: for i, similarity_matrix in enumerate(cosine_similarities):

30: similar_rows←(similarity_matrix > 0,5).any(axis=1)

31: for idx, satır in ayni_plakalar_df[similar_rows].iterrows():

32: matched_indices.append((idx, similarity_matrix[idx].argmax()))

33: for ayni_plakalar_idx, dfadresCapacities_idx in matched_indices:

34: data.append({ayni_plakalar_df.loc[ayni_plakalar_idx, "Adres"],
dfadresCapacities.loc[dfadresCapacities_idx, "fulladres"], ayni_plakalar_idx, dfadresCapacities_idx})

35: onceki_plaka=None

36: for idx, satır in data.iterrows():

37: result_df←result_df.append(satır[plaka, tarih, zaman, grup, surucu, hiz, yol, iletitipi, lat_gc, lon_gc, sip_miktar])

38: if onceki_plaka==plaka:

39: surefark←(to_datetime(zaman)- to_datetime(result_df.loc[onceki_adres_sure_hesaplamaicin - 1, "Zaman"])).seconds

40: else:

41: surefark←0

42: result_df.loc[idx, "SüreFark"]←surefark

43: onceki_plaka←plaka

44: onceki_adres_sure_hesaplamaicin += 1

45: for i, origin_row in satırlar: # Her satır sırayla bir 'origin' olarak kabul edilir

46: total_sure_farki←0

47: for j, dest_row in satırlar: # diğer satırlar 'destination' olarak kabul edilir

48: for index, satır in result_df.iterrows():

49: if (origin_row and dest_row == [[index, "lat_gc"] & [[index, "lon_gc"]]])

50: maxBenzerIndexBul←paralelIslemleriBaslat(result_df['DfadresCapacities_Fulladres'],index)

51: total_sure_fark+=result_df[maxBenzerIndexBul,'surefark']

52: distance_matrix[i][j]←{'duration': total_sure_fark}

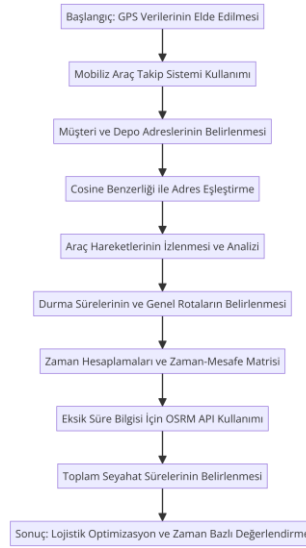
53: else

54: r←session.get # OSMR API'ye istek yap

55: distance_matrix[i][j]←{'duration': routes [duration]} # Origin ve destination arasındaki süreleri

depolayan sözlük

56: *Table 20, 8-18 aralığındaki satırlar*

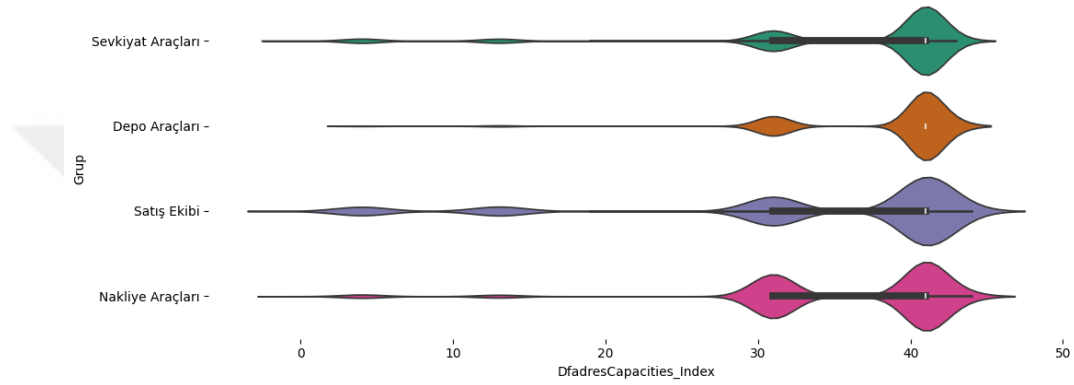


Şekil 37. GPS verilerine dayalı zaman-mesafe matrisi oluşturma ve rota optimizasyonu süreci (Erdem *et al.* under review b)

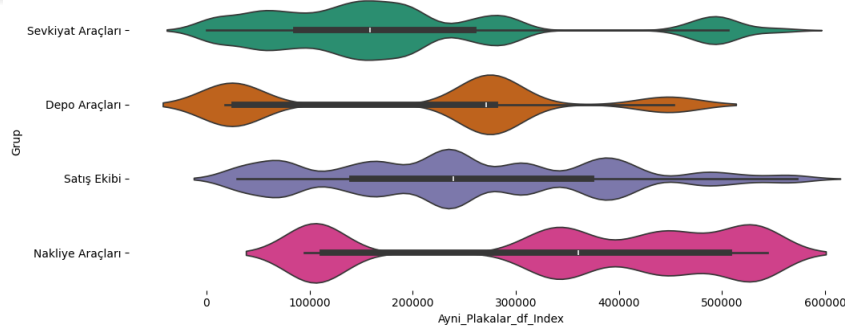
Mikro V15 veri tabanında gerçekleştirilen Mikro Veri Tabanı İlişkilendirmesi sürecinin (Şekil 28) ardından, en fazla sipariş veren ilk 100 müşteri grubu arasından öğle saatlerinde sipariş veren 59 müşteri dikkate alınarak aşağıdaki analizler gerçekleştirilmiştir. Her ne kadar rota optimizasyonu 31 Mart 2023 tarihli 29 gerçek müşteri siparişi üzerinden yürütülmüş olsa da, görsel analizlerin daha tutarlı örüntüler sunabilmesi, adres bazlı dağılımların anlamlı şekilde gözlemlenebilmesi ve araç grubu temsiliyetinin güçlendirilmesi amacıyla eşleşme başarısı yüksek olan 59 müşteri kaydı analiz kapsamına alınmıştır. Bu kapsamda, GPS verileriyle eşleştirilen fiziksel adreslerin, Mikro V15 veri setinde hangi satırlara karşılık geldiğini belirlemek amacıyla “dfadresCapacities_Index” sütunu kullanılmıştır. Bu sütun, Mikro V15 verisetindeki fiziksel adreslerin satır numaralarını temsil etmektedir. Eşleştirme sürecinde, GPS veri setinde yer alan her bir araç hareketi, Terim Frekansı Temelli Metin Vektörleştirici (TF-IDF) ve Cosinüs Benzerliği yöntemleriyle analiz edilerek en yüksek benzerlik oranına sahip fiziksel adres ile eşleştirilmiş ve ilgili satırın indeks değeri bu sütuna atanmıştır. Böylece, GPS kayıtları ile eşleşen fiziksel adreslerin sistematik biçimde izlenmesi ve bu adresler üzerinden süre farkı hesaplamaları, grup bazlı dağılım analizleri gibi işlemler gerçekleştirilmiştir. “DfadresCapacities_Index”, özellikle fiziksel adreslerin konumsal yoğunluğu, görev dağılımı vb. yönelik yapılan değerlendirmelerde referans noktası olarak kullanılmıştır. “DfadresCapacities_Index” ile araç grupları arasındaki ilişki, violin grafiği ile görselleştirilmiştir (Şekil 38). Grafik, her bir araç grubunun ağırlıklı olarak hangi adreslerle eşleştiğini ve bu adreslere yönelimin yoğunluk düzeyini ortaya koymuştur. Grafik yorumlandığında, “Sevkiyat Araçları” grubunun, hem Avrupa Yakası’ndaki Yenibosna Cemal Ulusoy Caddesi üzerindeki merkez depoyla hem de Çorlu - Bülent Ecevit Bulvarı üzerindeki bölge dağıtım noktasına yüksek oranda eşleştiği görülmüştür. Bu durum, söz konusu grubun iki

büyük dağıtım merkezi arasında yaygın bir sevkiyat görevi üstlendiğini göstermiştir. Benzer şekilde, Maltepe Çiftlik Yolu Kaçkar Sokak üzerindeki Anadolu Yakası deposu ile en çok eşleşen grup “Nakliye Araçları” olarak dikkat çekmiştir. Bu da Anadolu yakasında yapılan dağıtım faaliyetlerinin genellikle bu gruba ait araçlarla gerçekleştirildiğine işaret etmiştir. Öte yandan, “Satış Ekibi” ve “Depo Araçları” gibi grupların daha sınırlı adres indeksleriyle eşleştiği, yoğunluğun belirli birkaç adrese toplandığı görülmüştür. Özellikle Karacaköy Merkez Mahallesi - Karacaahmetpaşa Caddesi üzerindeki Çatalca deposu ile yapılan eşleşmelerde, bu araç gruplarının daha lokal ve sabit görevlerde kullanıldığı anlaşılmıştır. Violin şekillerinin dar ama yoğun olduğu bu gruplarda, operasyonlar daha spesifik lokasyonlara odaklanmıştır. Bu grafiksel analiz sayesinde, operasyonel planlamada görevlerin lokasyon bazında nasıl dağıldığı, hangi araç grubunun hangi fiziksel bölgeyle yoğun ilişkide bulunduğu ve bu eşleşmelerin ne kadar geniş bir alana yayıldığı analiz edilmiştir. Öte yandan, eşleştirme sürecinin diğer ucunda yer alan ve GPS verilerindeki araç hareketlerinin hangi adreslerle eşleştiğini temsil eden indeks bilgisi, “ayni_plakalar_df_Index” sütunu aracılığıyla takip edilmiştir. “ayni_plakalar_df_Index” sütunu, GPS verileri ile Mikro V15 veri setindeki fiziksel adresler TF-IDF ve Cosinüs benzerliği yöntemleri kullanılarak gerçekleştirilen eşleştirme sürecinin sonucunda elde edilmiştir. Bu indeksler, GPS veri setinde yer alan araç hareketlerine ait konum kayıtlarının, adres benzerliği esasına göre en yakın eşleşmeyle ilişkilendirildiği satır numaralarını temsil etmiştir. Böylece her bir araç hareketi, en yüksek benzerlik skoruna sahip fiziksel adrese bağlanmış ve bu eşleşme sonucunda “Ayni_Plakalar_df_Index” değeri atanmıştır. Bu indeksler, araçların günlük operasyonları sırasında hangi adreslerle en çok eşleştiğini ve hareket yoğunluklarının zaman içerisindeki dağılımını analiz etmek için bir referans noktası oluşturmuştur. Ayni_Plakalar_df_Index ile araç grupları arasındaki ilişki, violin grafiği kullanılarak Şekil 39’da incelenmiştir. Elde edilen sonuçlara göre, Sevkiyat ve Nakliye Araçları’nın veri seti boyunca geniş bir dağılıma sahip olmuş, bu grupların günün çeşitli zamanlarında ve farklı güzergahlarda aktif görev aldığı anlaşılmıştır. Depo araçları ve Satış Ekibi’ne ait araçlar ise daha sınırlı ve belirli zaman aralıklarında yoğunlaşarak, operasyonel açıdan daha sabit görevler üstlenmiştir. Şekil 40’da yapılan analizde, araç gruplarının zaman içerisinde yaptığı toplam yol (km) miktarı, “Ayni_Plakalar_df_Index” değişkeni ile karşılaştırılarak görselleştirilmiştir. Grafik, her bir araç grubunun belirli indeks aralıklarında yaptığı yol miktarını, çizgi eğilimleri üzerinden ortaya koymuştur. X ekseninde yer alan “Ayni_Plakalar_df_Index”, GPS kayıtlarının zaman sırasına göre satır indekslerini; y eksenini ise her bir araç hareketine karşılık gelen anlık yol uzunluklarını temsil etmiştir. Grafik yorumlandığında, “Sevkiyat Araçları” grubunun indeks aralığı boyunca en yüksek yol değerlerine ulaşmış ve rota açısından en aktif grup olduğu görülmüştür. Özellikle 100,000–

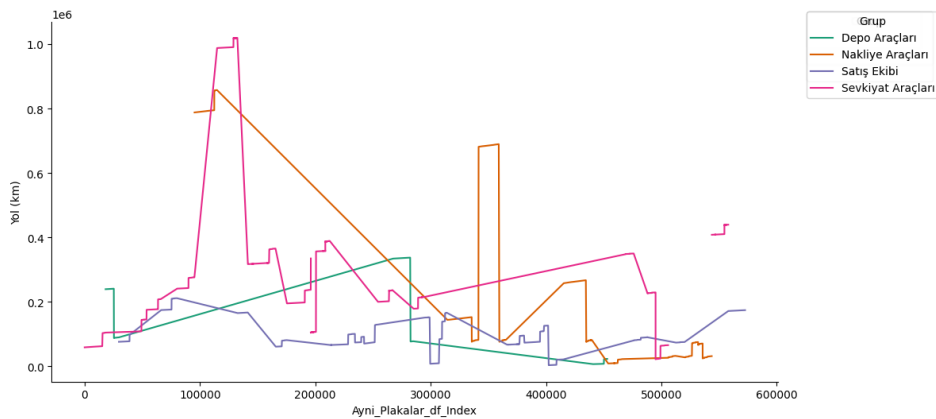
150,000 aralığında hızlı bir artış gözlemlenmiş, bu durum yoğun sevkiyat faaliyetiyle ilişkilendirilmiştir. “Nakliye Araçları” ise belirli indeks aralıklarında ani ve yüksek yol değerleri üretmiş, bu da belirli zaman dilimlerinde yoğun taşıma görevlerinin yapıldığını göstermiştir. “Depo Araçları” ve “Satış Ekibi” ise genellikle düşük mesafe aralıklarında hareket etmiş, bu durum bu grupların daha lokal, kısa mesafeli görevler üstlendiğini ortaya koymuştur. Satış Ekibi’nin yol eğrisinin zaman içinde dalgalı ama düşük düzeyde kalması, gün içinde çok sayıda kısa mesafeli müşteri ziyareti yaptığını göstermiştir. Bu grafiksel analiz sayesinde, her bir grubun gün içindeki yol kat etme davranışı, görev yoğunluğu ve zamanla olan ilişkisi detaylı olarak analiz edilmiştir.



Şekil 38. Farklı araç gruplarının adres ziyaret yoğunluklarının dağılımı



Şekil 39. Farklı araç gruplarına ait ziyaret edilen kayıtların indeks dağılımı



Şekil 40. Araç gruplarının zaman içindeki toplam mesafe (km) dağılımları

✓ Aşama III: Başlangıç Çözümünün Oluşturulması

Aşama III, müşteri taleplerinin depolar ve araçlar arasında dağıtılmasını optimize etmek için kullanılan başlangıç çözümlerinin oluşturulmasını ifade eder. Bu aşama, herhangi bir stratejiye dayalı olmadan, algoritmalar için bir temel başlangıç noktası belirlemeyi amaçlar. Bu başlangıç çözümleri rastgele üretilir. Amaç, algoritmalara uygulanacak ilk çözüm kümesini sağlamaktır.

✓ Aşama IV: Dağıtım Stratejileri

Aşama IV, Aşama III'te oluşturulan başlangıç çözümlerinin, müşteri taleplerinin depolar ve araçlar arasında nasıl dağıtılacağına ilişkin belirli stratejilere göre optimize edilmesini sağlar. Bu aşamada üç farklı strateji (Depoya Yakın Dağılım (S_{dyd}), Rastgele Dağılım (S_{rd}), ML Başlangıç Çözümü (S_{mld})) uygulanarak başlangıç çözümleri daha yapılandırılmış ve stratejik bir yaklaşımla geliştirilir. Bu stratejiler, coğrafi yakınlık, rastgele dağılım ve ML tabanlı çözümleri içerir. Temel amaç, Aşama V'te meta-sezgisel algoritmalara daha iyi başlangıç noktaları sağlamaktır (Erdem *et al.* under review b).

- ✓ S_{dyd} : Her müşteri, coğrafi olarak en yakın olduğu depoya atanır. Yaklaşım, depo-müşteri yakınlığına dayalı en kısa mesafeli rotaları oluşturmayı hedefler. Bu işlemde Haversine formülü kullanılarak müşteriler ile depolar arasındaki mesafeler hesaplanmıştır. Formül, dünya yüzeyinin eğimini dikkate alarak iki koordinat arasındaki gerçek mesafeyi ölçer.
- ✓ S_{rd} : Depo ve müşteri konumları rastgele dağıtılmıştır. Bu yöntem, sistemin farklı olasılıklarla nasıl performans gösterdiğini anlamak ve karşılaştırma yapmak için uygulanmıştır.
- ✓ S_{mld} : S_{mld} yöntemi, AI tabanlı yaklaşımın temel bir unsuru olarak, ML modellerinden elde edilen çıktılar aracılığıyla optimize edilmiş bir başlangıç çözümü sunmaktadır. Bu süreç, geçmiş veri setlerinden türetilen en uygun başlangıç çözümünün meta-sezgisel algoritmalara girdi parametresi olarak sağlanmasını içermektedir.

AI tabanlı yaklaşım, iki fazda gerçekleştirilmiştir. Bu iki faza ait sözde kodlar Tablo 23'de detaylı olarak verilmiş, görsel özetleri ise sırasıyla Şekil 41 ve Şekil 42'de sunulmuştur.

Tablo 23. ÇDARP Çözümü için ML Modelleri (Erdem *et al.* under review b)**Algoritma: ÇDARP çözümü için ML modelleri kullanılarak başlangıç çözümünün oluşturulması**

```

1: df_totalmikro_sabah, df_totalmikro_ogle, df_totalmikro_aksam ← read_csv(dataset)
2: result_df_sabah, result_df_ogle, result_df_aksam ← read_csv(dataset)
3: df_totalmikro ← concat([df_totalmikro_sabah, df_totalmikro_ogle, df_totalmikro_aksam])
4: result_df ← concat([result_df_sabah, result_df_ogle, result_df_aksam])
5: dfadresCapacities ← read_csv(dataset) #31.03 ogle
6: def MLUygunlukBulParallelVectorized(row):
7:   plaka ← row['Plaka'], tarih ← row['Tarih']
8:   matching_rows ← result_df[(result_df['Plaka'] == plaka) & (result_df['Tarih'] == tarih) & (result_df.index !=
row.name)]
9:   totalsurefark ← matching_rows['SüreFark'].sum()
10:  rota ← '.join(matching_rows['DfadresCapacities_Fulladres'])
11:  return totalsurefark, rota
12: gecmisrotalarwithTimeValue ← DataFrame(columns=['Rota Fiziksel Adresler', 'Maliyet(Süre bazında)'])
13: rota ← []
14: with ThreadPoolExecutor(max_workers=5) as executor:
15:   futures ← []
16:   for _, row in result_df.iterrows():
17:     futures.append(executor.submit(MLUygunlukBulParallelVectorized, row))
18:   for future in as_completed(futures):
19:     result ← future.result()
20:     gecmisrotalarwithTimeValue.append({'Rota Fiziksel Adresler yan yana': result[1],
'Maliyet(Süre bazında)': result[0]})
21: df ← read_csv(gecmisrotalarwithTimeValue)
22: X ← df.iloc[:, 1]
23: y ← df.iloc[:, 0]
24: X, y ← LabelEncoder().fit_transform(X), LabelEncoder().fit_transform(y)
25: x_train, x_test, y_train, y_test ← train_test_split(X, y, test_size=0.2, random_state=0)
26: model ← MLModel()
27: model ← model.fit(x_train, y_train)
28: preds ← model.predict(x_test)
29: calculate(mae(y_test, preds), mse(y_test, preds), rmse(y_test, preds), r2(y_test, preds))
30: plot_predictions(y_test, preds)
31: def match_dimensions(y_true, y_pred):
32:   min_len ← min(len(y_true), len(y_pred))
33:   y_true_matched, y_pred_matched ← y_true[:min_len], y_pred[:min_len]
34:   return y_true_matched, y_pred_matched
35: def MLUygunlukBul(rota_tahmin):
36:   for i in range(len(rota_tahmin)):
37:     for index, fullAdres_row in df_totalmikro.iterrows():
38:       fullAdres_degisken ← fullAdres_row['fulladres'].lower().strip()
39:       dfadresCapacities_degisken ← dfadresCapacities.at[rota_tahmin[i]-1, 'fulladres'].lower().strip()
40:       if fullAdres_degisken == dfadresCapacities_degisken:
41:         x_test_tahmin.append(dfadresCapacities.at[rota_tahmin[i] - 1, 'fulladres'])
42:         break
43:   x_test_tahmin ← LabelEncoder().fit_transform(x_test_tahmin)
44:   pretrained_features_scaled, new_features_scaled ← StandardScaler().(fit_transform(x_train),
transform(x_test_tahmin))
45:   model.fit(pretrained_features_scaled, y_train)
46:   prediction ← model.predict(new_features_scaled)
47:   y_test, prediction ← match_dimensions(y_test, prediction)
48:   calculate(mae(y_test, prediction), mse(y_test, prediction), rmse(y_test, prediction), r2(y_test, prediction))
49:   plot_predictions(y_test, prediction)
50:   yTahminEdilenMaliyetHataDegeri ← mse
51:   return yTahminEdilenMaliyetHataDegeri
52: while T > cr
53:   for 1:iterPerMax
54:     next_rnd_sol_1 ← PerturbASolution(rnd_sol_1)
55:     hataOrani ← MLUygunlukBul(next_rnd_sol_1)
56:     hataOraniDf ← hataOraniDf.append({'Hata Orani': hataOrani, 'Next Rnd Sol': next_rnd_sol_1},
ignore_index=True)
57:     rnd_sol_1 ← next_rnd_sol_1
58:     T ← T*cr
59:   siralanmisRotalarListesi ← hataOraniDf.sort_values(by='Hata Orani')
60:   findMinHataliRota ← siralanmisRotalarListesi.iloc[0]['Next Rnd Sol']

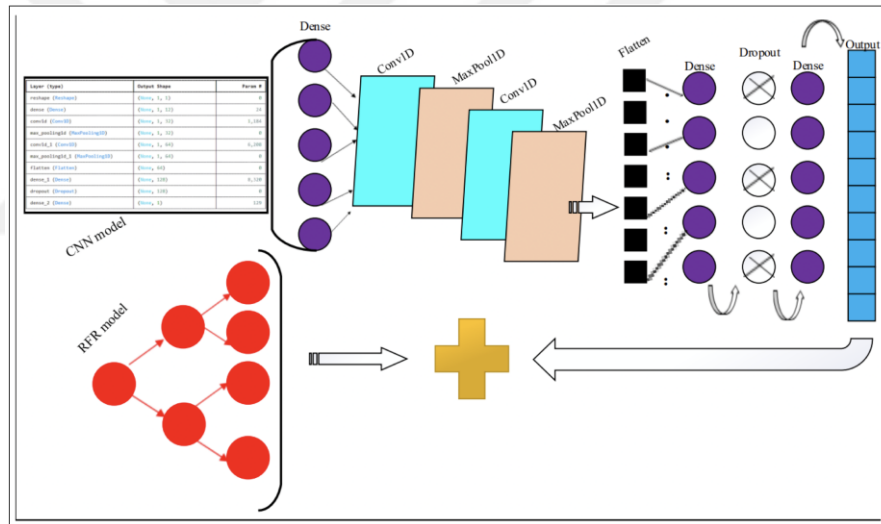
```

diğer satırları bulmuştur. Bu işlem sırasında, eşleşen satırlardaki toplam zaman farkını hesaplamış ve bu satırlara ait adresleri birleştirerek rotaları oluşturmuştur (Satır 6-11). Elde edilen rota bilgisi, "Rota Fiziksel Adresler" sütununda saklanırken, toplam zaman farkı ise "Maliyet" sütununa kaydedilmiştir. Paralel işleme sonucu hesaplanan rotalar ve maliyetler "gecmisrotalarwithTimeValue" adlı veri çerçevesinde saklanmıştır (Satır 20). Rota ve maliyet verilerinin oluşturulması sırasında, ThreadPoolExecutor kullanılarak maksimum 5 çekirdek sayısı ile paralel olarak işlemler gerçekleştirilmiştir (Satır 14-20). Bu rotalar, ML modelleri için girdi (X) ve maliyetler hedef değişkenler (y) olarak ayrılmıştır. Eğitim ve test verileri %80 eğitim, %20 test olarak ayrılmıştır (satır 21-24). ML Modellerinin Entegrasyonu için KNR, RFR, XGB, LR, DTR, MLPR ve RFR-DL algoritmaları kullanılmıştır. ML algoritmalarının veri setindeki örüntüleri (rotalar ve süreler arasındaki ilişkiyi kavrayarak) öğrenip daha isabetli tahminler yapabilmesi sağlanmıştır. Tahmin performansları Ortalama Mutlak Hata (MAE), Ortalama Kare Hata (MSE), Kök Ortalama Kare Hata (RMSE) ve R^2 gibi metriklerle değerlendirilmiştir (satır 26-30).

Tahmin sonuçlarının alındığı 7 farklı (KNR, RFR, XGB, LR, DTR, MLPRRegressor (MLPR) ve RFR-DL) ML modeli incelenecek olursa;

KNR, özellik uzayında veri noktalarının yakınlıklarını kullanarak tahmin yapar. Bu model, en yakın komşuların değerlerini ortalayarak rotanın belirli segmentlerine uyarlanmış tahminler sunar. Bu yetenek, geçmiş verilere dayanarak mesafeler veya zaman aralıkları arasında yerel eğilimleri ve benzerlikleri tanımlamak için değerlidir. RFR, birden fazla karar ağacından elde edilen tahminleri bir araya getirerek daha doğru ve genellenebilir sonuçlar sunan topluluk öğrenme yöntemini kullanır. Bu model, geçmiş rotalardaki karmaşık desenleri ve anormallikleri yakalamada etkilidir. XGB, başlangıçta basit bir model oluşturur ve ardından önceki modellerin hatalarını düzelten ek modeller ekleyerek tahmin doğruluğunu artırır. XGB, büyük veri setlerini işleyebilme yeteneği ile geçmiş verilerdeki karmaşık ilişkileri ve desenleri yakalamakta oldukça başarılıdır. Bu model her adımda türevlenebilir bir kayıp fonksiyonunu optimize eder. LR, bağımlı bir değişken ile bir veya daha fazla bağımsız değişken arasındaki ilişkiyi modelleyen doğrusal bir regresyon modelidir. Bu model, geçmiş rota verilerindeki doğrusal ilişkileri anlamak için faydalıdır. DTR, veri özelliklerinden türetilen karar kurallarını öğrenerek hedef değişkeni tahmin eder. Bu model, geçmiş verilere dayalı olarak belirli kurallar türetmek için uygundur. MLPR, yapay sinir ağlarından oluşur ve geri yayılım yöntemiyle giriş özellikleri ile hedef değerler arasındaki doğrusal olmayan ilişkileri öğrenir. Bu model, geçmiş verilerdeki karmaşık desenleri analiz ederek daha doğru rota tahminleri sağlar. RFR-DL, ML ve DL tekniklerini birleştiren hibrit bir modeldir. Bu modelde, rastgele orman ve DL tahminleri

birleştirilir ve ortalaması alınır. DL modeli, araç rotaları veya konum verileri gibi sayısal verilere evrişim uygulayarak yerel desenleri ve özellikleri çıkarır. Bu amaçla Conv1B katmanları kullanılarak, rotalardaki komşu lokasyonlar arasındaki mesafe veya zaman farkı gibi yerel ilişkiler modellenir. Böylelikle, verideki gizli desenler ortaya çıkarılarak daha optimize sonuçlara ulaşılır. Önerilen hibrit modelde, DL modeli; 3 tam bağlı katman, 2 konvolüsyon katmanı, 2 maksimum havuzlama katmanı, bırakma katmanı ve düzleştirme katmanından oluşmaktadır. Modele verilen giriş verisinde her bir özellik için ayrı bir kanal oluşturulmuştur. Tam bağlı ve konvolüsyon katmanlarında ReLU aktivasyon katmanları kullanılmıştır. Maksimum havuzlama ve konvolüsyon katmanlarında, dolgu (padding) “same” olarak alındı. Konvolüsyon katmanlarında 3x3'lük filtreler uygulandı. Konvolüsyon katmanlarında filtre sayıları, sırasıyla 32 ve 64 olarak uygulanmıştır. Maksimum havuzlama penceresinin boyutu 2 olarak ayarlanmıştır. Aşırımı uyumu önlemek için kullanılan bırakma katmanı oranı %50 verildi. İlk tam bağlı katman 12 nöron, ikinci tam bağlı katman 128 nöron ve son tam bağlı katman 1 nöron oluşmaktadır (Şekil 42) (Erdem *et al.* under review b).



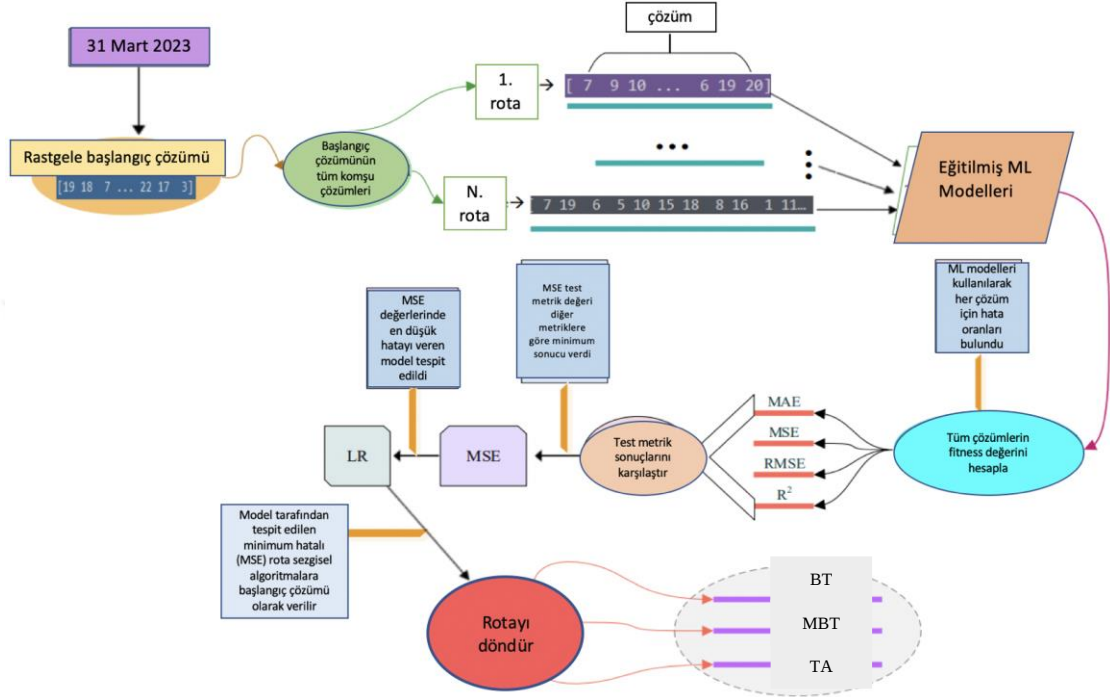
Şekil 42. Hibrit modelin genel mimarisi (Erdem *et al.* under review b)

Faz 1 sonucunda eğitilmiş ML modelleri, Faz 2'de başlangıç çözümünün iyileştirilmesi için girdi olarak verilmiştir. Bu modeller, toplam teslimat sürelerini tahmin etmek ve bu tahminler doğrultusunda başlangıç çözümünü optimize etmede spesifik bir amaçla kullanılmıştır (Erdem *et al.* under review b).

Faz II: ML ve Meta-sezgisel Algoritmaların Entegrasyonu

Tablo 23'de 31 ile 60. satırlar arasında yer alan adımlar, Faz II kapsamındadır. Süreç boyunca, rastgele bir başlangıç çözümünden hareketle 1'den N'e kadar komşu rotalar oluşturulmuştur. Bu rotalar, Benzetilmiş Tavlama (BT) algoritmasına benzer bir yaklaşım ile, başlangıç parametreleri (sıcaklık, epsilon, soğuma oranı) kullanılarak mevcut çözümün

yinelemeli olarak pertürbe edilmesiyle üretilmiştir (satır 52-58). Oluşturulan yeni aday çözümler, eğitilmiş ML modelleriyle değerlendirilmiş ve iyileştirilmiştir (satır 55). Hata oranları farklı metriklerle (MAE, MSE, RMSE, R^2) karşılaştırılıp en düşük hata oranına sahip çözüm belirlenmiştir. Belirlenen bu çözüm, meta-sezgisel algoritmalar (BT, MBT, TA) için başlangıç çözümü olarak kullanılmıştır. Bu süreç, minimum hata ile en uygun rota bulma ve daha etkin sonuçlar elde etme üzerine odaklanmıştır (Şekil 43) (Erdem *et al.* under review b).



Şekil 43. ÇDARP çözümüne yönelik AI tabanlı yaklaşımın II. aşamasının gösterimi (Erdem *et al.* under review b)

Faz 2’deki işlem basamakları ayrıntılı incelendiğinde (Erdem *et al.* under review b),

Tablo 23’de verilen sözde kodda ilk olarak 31 Mart 2023 tarihindeki müşteri siparişleri dikkate alınarak rastgele bir başlangıç çözümü oluşturulmuştur. BT algoritmasına benzer bir yaklaşımla her iterasyonda başlangıç çözümü pertürbe edilerek yeni aday çözümler (rotaları) değerlendirilmiştir. Bu değerlendirme sürecinde, “MLUygunlukBul” fonksiyonu kullanılarak aday çözümlerin maliyetleri hesaplanmış ve her bir çözüm için bir hata oranı belirlenmiştir. Fonksiyonun işleyişine bakıldığında, her rota düğümü, müşteri adreslerini içeren “dfadresCapacities” ve “df_total_micro” veri çerçevelerindeki bilgilerle eşleştirilmiştir. Tahmin edilen rota, gerçek teslimat adresleriyle karşılaştırılmıştır. Eğer bir eşleşme bulunursa, ilgili adresler “x_test_tahmin” listesine eklenmiştir (satır 36-42). Bu eşleşen adresler daha sonra sayısal verilere dönüştürülmüş ve ölçeklendirilmiştir (satır 43-44). Eğitilmiş ML modeli (model), ölçeklendirilmiş test verileri üzerinde tahmin yaparak rota maliyetlerini öngörmüştür (satır 45-46). Gerçek değerler ve tahmin edilen rota maliyetleri arasındaki hata, çeşitli test

metrikleriyle (örneğin, MAE, MSE, RMSE, R^2) hesaplanmıştır (satır 47-50). Gerçek değerler, geçmiş teslimatlara ait rotaların maliyetlerini içermiş ve rota adreslerinin özelliklerine dayalı olarak oluşturulan "Maliyet" sütunundaki verilerden türetilmiştir. Bu veri yapısı, her bir rota için toplam teslimat maliyetini içerir. Tahmin edilen değerler ise ML modelinin test setindeki tahmin sonuçlarını ifade eder. Ayrıca, y_{True} ve y_{Pred} veri yapılarındaki boyut uyumsuzluklarının giderilmesi, hatasız bir değerlendirme süreci sağlar. Bu düzenleme, modelin farklı test setleri veya rastgele çözümler için hatalarını tutarlı bir şekilde analiz edebilmesini mümkün kılmıştır. Bu süreç, özellikle ÇDARP çözümünde geçmiş verilere dayalı optimize edilmiş çözümler oluştururken kritik bir öneme sahiptir. Boyut uyumsuzluğunu önlemek için "matchDimensions" fonksiyonu kullanılmıştır. Bu fonksiyon, her iki veri yapısının uzunluklarının minimumunu (minLen) alır ve bu uzunluğa göre her iki veri yapısını da kırpır. Sonuç olarak, eşit uzunlukta "yTrueMatched" ve "yPredMatched" veri yapıları elde edilmiştir. Böylece, hata metriklerinin (MAE, MSE, RMSE, R^2) hesaplanması sırasında oluşabilecek boyut uyumsuzluklarından kaynaklanan istisnalar önlenerek tutarlı karşılaştırma yapılmıştır. Hesaplanan hata metrikleri arasında MSE, başlangıç çözümünü değerlendirirken en düşük hata değerini sağlamıştır. Bu nedenle, "MLUygunlukBul" fonksiyonu, tahmin edilen rota maliyeti olarak MSE değerini geriye döndürmektedir. Algoritma boyunca oluşturulan her yeni çözüm ve bu çözümlere ait hata oranları "hataOranıDf" adlı veri çerçevesinde saklanmıştır (satır 55). BT algoritması, belirlenen tamamlama kriterlerine (örneğin, sıcaklık veya soğutma oranı) ulaştığında sona ermiştir. Bu sürecin sonunda, "hataOranıDf" veri çerçevesi içerisinde en düşük hata oranına sahip çözüm ("findMinHatalıRota") seçilmiş ve nihai çözüm olarak belirlenmiştir (satır 59-60). Nihai çözüm, meta-sezgisel algoritmalar (BT, MBT, TA) için başlangıç çözümü olarak atanmıştır.

Bu yöntem, özellikle ÇDARP çözümünde geçmiş verilere dayalı optimize edilmiş çözümler üretmek için kritik bir öneme sahiptir. ML ile geçmiş teslimat verilerini entegre ederek rota optimizasyonunda daha etkin ve doğru sonuçlar elde edilmesini sağlamıştır.

✓ Aşama V: Rota Optimizasyonu

Bu aşamada, önceki aşamalarda oluşturulan zaman-mesafe matrisleri optimizasyon algoritmaları ile birlikte kullanılarak en verimli rotalar belirlenmiştir. Bu adım, müşterilerin araçlara atanmasını ve ziyaret sırasını optimize etmeyi içerir. Bu aşamada, BT, MBT ve TA gibi algoritmalar seçilmiş ve uygulanmıştır. Algoritmaların başlangıç çözümü belirlenirken, 3 farklı strateji uygulanmıştır: S_{dyd} , S_{rd} ve S_{mld} (Erdem *et al.* under review b).

- BT

BT algoritması, ÇDARP çözümünde optimize edilmiş rotaların belirlenmesi için kullanılan bir meta-sezgisel yöntemdir. Bu algoritma, başlangıç çözümü (S_0), başlangıç sıcaklığı (T_0), soğutma oranı (cr) ve her sıcaklık seviyesinde gerçekleştirilecek maksimum deneme sayısı ($iterPerMax$) gibi temel girdilere dayanır. Nihai hedef, optimize edilmiş bir çözüm (S^* , en düşük maliyetli rota) bulmaktır (Şekil 44). Sözde kodu Tablo 24’de verilen algoritmanın işleyişi incelendiğinde, algoritma, başlangıç çözümü (S_0) ile başlar. Başlangıç çözümünün maliyeti, $cost(S_0)$ fonksiyonu ile hesaplanır ve bu maliyet, en iyi çözüm olarak başlangıç değerine atanır (satır 1-2). Her iterasyon sonunda sıcaklık, soğutma oranı ile çarpılarak azaltılır. Algoritma, sıcaklık değeri (T) soğutma oranı (cr) altına düşene kadar bir döngü içerisinde çalışır. Her sıcaklık seviyesinde, maksimum deneme sayısı kadar iterasyon gerçekleştirilir. Her bir iterasyonda çözüm üzerinde iyileştirmeler yapılır (satır 3-4). Her iterasyonda, mevcut çözümden (S_0) bir komşu çözüm (S_1) oluşturulur. Bu işlem, rotalardaki düğümlerin yer değiştirilmesiyle gerçekleştirilir (satır 5). Komşu çözümün maliyeti hesaplanır ve mevcut çözümle arasındaki maliyet farkı (ΔC) belirlenir (satır 6). Eğer $\Delta C < 0$ ise, komşu çözüm mevcut çözümden daha iyi olduğundan mevcut çözüm olarak güncellenir ($S_0 \leftarrow S_1$). Ayrıca, eğer bu çözüm en iyi çözümden daha düşük maliyete sahipse, en iyi çözüm de (best) güncellenir. Eğer $\Delta C > 0$ ise, algoritma yerel minimumlardan kaçınmak için daha kötü çözümleri kabul edebilir. Bu olasılık, rastgele bir sayı ile $\exp(-\Delta C / T)$ olasılık fonksiyonu kullanılarak belirlenir (satır 7-13). Döngü tamamlandığında, en düşük maliyete sahip çözüm, algoritmanın nihai çözümü olarak atanır ($S \leftarrow best^*$). Bu çözüm, algoritmanın optimize edilmiş son çıktısıdır (Erdem *et al.* under review b).

Tablo 24. Farklı Başlangıç Çözüm Yaklaşımlarıyla ÇDARP Çözümü için BT Algoritması Sözde Kodu (Erdem *et al.* under review b)

Algoritma: ÇDARP Çözümü için BT Algoritması Yalancı Kodu

Girdi: Başlangıç çözümü ($S_0 = S_{dyd} \mid S_{rd} \mid S_{mld}$), başlangıç sıcaklığı (T_0), soğutma oranı (cr), ve bir sıcaklık değeri başına en fazla deneme sayısı ($iterPerMax$)

Çıktı: Final çözüm (S^*)

1: $best \leftarrow S_0$, $T \leftarrow T_0$

2: $cost(best) \leftarrow cost(S_0)$

3: while $T > cr$

4: for 1: $iterPerMax$

5: $S_1 \leftarrow \text{perturbFindNeighbourSolution}(S_0)$

6: $\Delta C \leftarrow cost(S_1) - cost(S_0)$

7: if $\Delta C < 0$

8: $S_0 \leftarrow S_1$

9: if $cost(S_0) < cost(best)$

10: $best \leftarrow S_0$

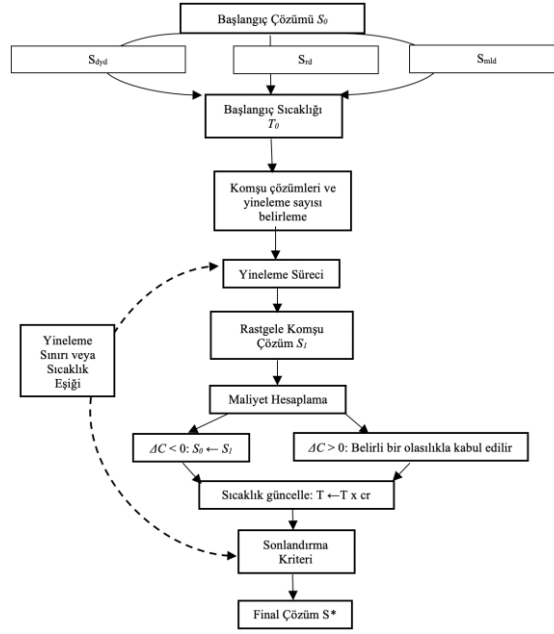
11: $cost(best) \leftarrow cost(S_0)$

12: elseif $\text{random}(0,1) < \exp(e, -(\Delta C/T))$

13: $S_0 \leftarrow S_1$

14: $T \leftarrow T * cr$

15: $S^* \leftarrow best$



Şekil 44. Farklı başlangıç yaklaşımlarıyla BT tabanlı ÇDARP çözüm süreci (Erdem *et al.* under review b)

- MBT

MBT Algoritması (Şekil 45), BT algoritmasının temel yapısını kullanmakla birlikte, çözüm sürecini daha etkili hale getirmek ve çözüm uzayında daha geniş bir çeşitlilik sağlamak amacıyla bazı ek düzenlemeler içermektedir (Tablo 25). Bu düzenlemeler, algoritmanın daha iyi sonuçlara ulaşabilmesi için kritik önem taşımaktadır. MBT algoritmasının BT algoritmasından farklılaşan yönleri ek kontrol mekanizması ve mutasyon süreci olarak ifade edilebilir (Erdem *et al.* under review b).

- Ek Kontrol: MBT algoritması, her 30 iterasyonda mevcut çözümün en iyi çözümle karşılaştırılmasını sağlar. Eğer mevcut çözüm kötü bir noktadaysa, algoritma bu çözümü en iyi çözümle değiştirir (Satır 14-17). Bu mekanizma, algoritmanın alt-optimal çözümlere takılmasını önler.
- Mutasyon İşlemi: MBT algoritmasının BT algoritmasından ayrıldığı temel nokta mutasyon işlemidir. Mutasyon, çözüm uzayında daha geniş bir çeşitlilik sağlar ve algoritmanın yeni çözümler keşfetme kabiliyetini artırır (Satır 18-34). Ters Çevirme (Inversion) ve Yer Değiştirme (Swap) gibi iki aşamalı bir mutasyon işlemi uygulanır.
- Ters Çevirme:

Eğer rastgele seçilen bir mutasyon olasılığı (randomMutation) tanımlı mutasyon olasılığından ($p_{\text{Mutation}} = \%20$) küçükse, iki rastgele indeks (randMut1 ve randMut2) alınır. Bu indeksler arasındaki elemanlar ters çevrilir ve mutasyona uğramış çözüm (mutatedS0) oluşturulur (satır 20-30).

- Yer Değiştirme:

Mutasyona uğramış çözümde, iki farklı rastgele indeks (randMut1 ve randMut3) alınır ve bu indekslerdeki elemanlar yer değiştirir. Bu ek düzenleme, çözüm uzayında daha geniş bir araştırma yapılmasını sağlar (satır 31-34).

Tablo 25. Çeşitli Başlangıç Çözüm Yaklaşımlarıyla ÇDARP Çözümü için MBT Algoritması Yalancı Kodu (Erdem *et al.* under review b)

Algoritma: ÇDARP Çözümü için MBT Algoritması Yalancı Kodu

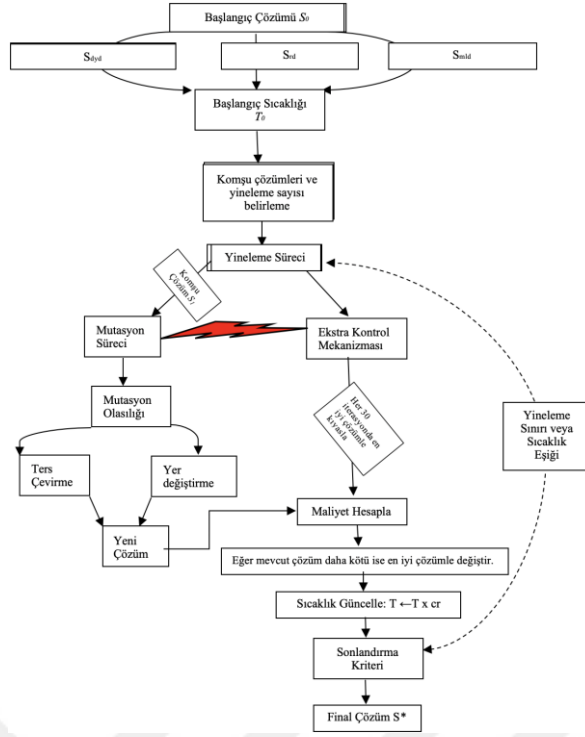
Girdi: Başlangıç çözümü ($S_0 = S_{dyd} \mid S_{rd} \mid S_{mld}$), başlangıç sıcaklığı (T_0), soğuma oranı (cr), ve bir sıcaklık değeri başına en fazla deneme sayısı ($iterPerMax$), Mutasyon olasılığı ($pMutation$)

Çıktı: Final çözüm (S^*)

```

1: best ←  $S_0$ ,  $T \leftarrow T_0$ 
2: cost(best) ← cost( $S_0$ )
3: while  $T > cr$ 
4:   for 1:  $iterPerMax$ 
5:      $S_1 \leftarrow \text{perturbFindNeighbourSolution}(S_0)$ 
6:      $\Delta C \leftarrow \text{cost}(S_1) - \text{cost}(S_0)$ 
7:     if  $\Delta C < 0$ 
8:        $S_0 \leftarrow S_1$ 
9:       if cost( $S_0$ ) < cost(best)
10:        best ←  $S_0$ 
11:        cost(best) ← cost( $S_0$ )
12:     elseif random (0,1) < exp(e,  $-(\Delta C/T)$ )
13:        $S_0 \leftarrow S_1$ 
14:     modIteration ← iteration % 30
15:     if modIteration == 0
16:       if cost( $S_0$ ) > cost(best)
17:          $S_0 \leftarrow \text{best}$ 
18:       if randomMutation ≤ pMutation
19:         mutated $S_0$  ← []
20:         randMut1 ← rand() % len( $S_0$ ) + 1
21:         randMut2 ← rand() % len( $S_0$ ) + 1
22:         randMut3 ← rand() % len( $S_0$ ) + 1
23:         if randMut1 < randMut2
24:           randomIndex1 ←  $S_0$ [randMut1]
25:           randomIndex2 ←  $S_0$ [randMut2]
26:           mutated $S_0$  ← reversed( $S_0$ [randomIndex1:randomIndex2])
27:         if randMut2 < randMut1
28:           randomIndex1 ←  $S_0$ [randMut2]
29:           randomIndex2 ←  $S_0$ [randMut1]
30:           mutated $S_0$  ← reversed( $S_0$ [randomIndex1:randomIndex2])
31:         temp ← mutated $S_0$ [randMut1]
32:         mutated $S_0$ [randMut1] ← mutated $S_0$ [randMut3]
33:         mutated $S_0$ [randMut3] ← temp
34:          $S_0 \leftarrow \text{mutated}S_0$ 
35:       iteration ← iteration + 1
36:      $T \leftarrow T * cr$ 
37:  $S^* \leftarrow \text{best}$ 

```



Şekil 45. Farklı başlangıç yaklaşımlarıyla MBT tabanlı ÇDARP çözüm süreci (Erdem *et al.* under review b)

- TA

TA algoritması (Şekil 46), ÇDARP için uygulanmış ve çözüm sürecini optimize etmek için yerel arama yöntemlerinden faydalanılmıştır. Bu algoritma, döngüsel hareketleri önlemek ve aynı çözüm yollarının tekrar edilmesini engellemek için “tabu listesi” adı verilen bir yapı kullanır. Tabu listesi, çözüm arama sırasında yapılan hareketleri ve belirli kriterlere uymayan durumları kaydederek, çözüm sürecinin çeşitliliğini artırmayı ve yerel optimal tuzağından kurtulmayı hedefler. Tablo 26’da algoritmanın sözde kodu incelendiğinde: Algoritma, başlangıç çözümü olarak S_{dyd} , S_{rd} veya S_{mld} yöntemlerinden biriyle oluşturulan bir çözümle başlar (satır 1). Tabu listesi başlangıçta boş olarak tanımlanır (satır 3). İlk çözümün maliyeti hesaplanır ve bu çözüm "en iyi çözüm" olarak işaretlenir (satır 1-2). Her iterasyonda, mevcut çözümden bir komşu çözüm oluşturulur (satır 6). Yeni çözümün maliyeti, mevcut çözümün maliyetiyle karşılaştırılarak bir fark değeri (Δ) hesaplanır (satır 7). Yeni bir çözüm, çözüm tabu listesinde yer almıyorsa, maliyet farkı (Δ) sıfırdan küçükse (yani daha iyi bir çözümse) veya çözüm tabu listesindeki bir hareket olsa dahi “aspiration” kriterini karşılıyorsa kabul edilir (satır 8) (Erdem *et al.* under review b). Kabul edilen çözüm, mevcut çözüm olarak güncellenir (satır 9). Eğer mevcut çözüm, en iyi çözümden daha düşük bir maliyete sahipse, bu çözüm “en iyi çözüm” olarak kaydedilir ve maliyet bilgisi güncellenir (satır 10-12). Kabul edilen hareketler tabu listesine eklenir. Eğer tabu listesinin boyutu belirlenen sınırı (L) aşarsa, en eski hareket listeden çıkarılır (satır 13-15). Bu mekanizma, çözüm arama sürecinde esneklik ve çeşitlilik

sağlar. Algoritma, belirlenen iterasyon sayısına ($maxIter$) ulaşana veya daha fazla iyileştirme sağlanamayana kadar çalışır (*satır 16*). Bu kriterlere ulaşıldığında, en iyi çözüm (S^*) döndürülür (*satır 17*).

TA algoritması, temelde yerel arama yöntemlerini genişletmek ve döngüsel hareketleri önlemek amacıyla tabu listesini kullanır. Ancak, mutasyon işlemi, çözüm çeşitliliğini artırmak ve yerel optima tuzaklarından kurtulmak için daha gelişmiş varyasyonlarda eklenebilir.

Tablo 26. Çeşitli Başlangıç Çözüm Yaklaşımlarıyla ÇDARP Çözümü için TA Algoritması Yalancı Kodu (Erdem *et al.* under review b)

Algoritma: ÇDARP Çözümü için TA Algoritması Yalancı Kodu

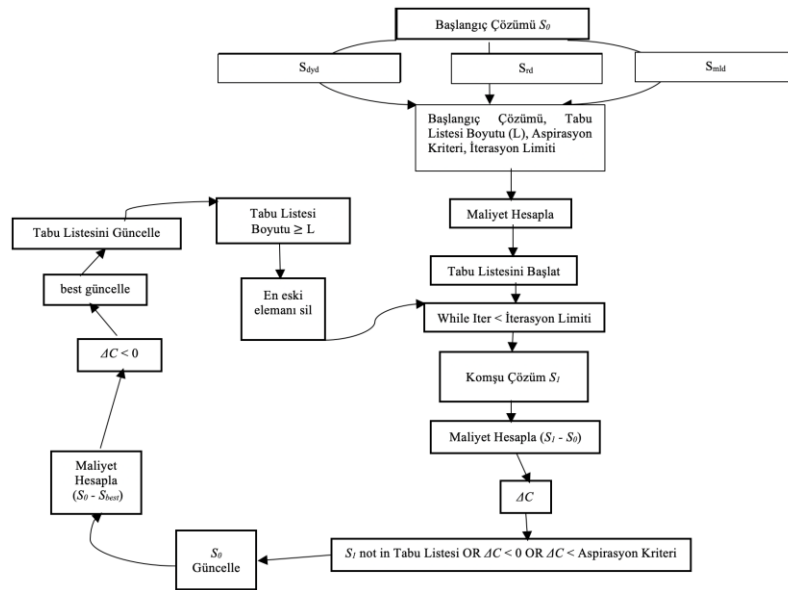
Girdi: Başlangıç çözümü ($S_0 = S_{dyd} \mid S_{rd} \mid S_{mld}$), başlangıç sıcaklığı (T_0), soğuma oranı (cr), ve bir sıcaklık değeri başına en fazla deneme sayısı ($iterPerMax$), Mutasyon olasılığı ($pMutation$)

Çıktı: Final çözüm (S^*)

```

1: best ←  $S_0$ 
2: cost(best) ← cost( $S_0$ )
3: tabu list = []
4: Iter ← 0
5: while Iter <  $maxIter$ 
6:    $S_1$  ← findNeighbourSolution( $S_0$ )
7:    $\Delta C$  ← cost( $S_1$ ) - cost( $S_0$ )
8:   if ( $S_1$  is not in tabu list) or ( $\Delta C < 0$ ) or ( $\Delta C < aspiration\ criterion$ )
9:      $S_0$  ←  $S_1$ 
10:  if cost( $S_0$ ) < cost(best)
11:    best ←  $S_0$ 
12:    cost(best) ← cost( $S_0$ )
13:    Tabu listesini mevcut hareketle güncelle
14:    if size(tabu list) ≥  $L$ 
15:      Tabu listesindeki en eski elemanı sil
16:  Iter ← Iter + 1
17:  $S^* \leftarrow best$ 

```



Şekil 46. Farklı başlangıç yaklaşımlarıyla TA tabanlı ÇDARP çözüm süreci (Erdem *et al.* under review b)

✓ Aşama VI: Ürünlerin Araçlara Yerleştirilmesi ve Müşteri Teslimatı

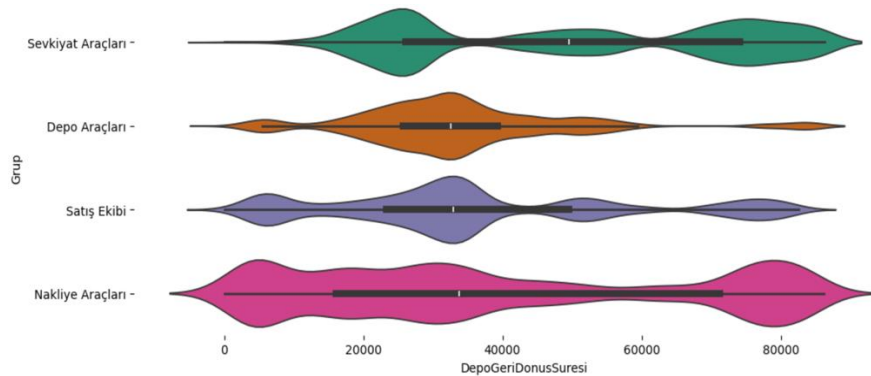
Bu aşamada, ÇDARP çözümünde, araçlar müşteri taleplerine ve minimum maliyetli rota planına göre yüklenmiştir. Aşama V’de oluşturulan optimum rotalar, müşteri konumlarını temsil eden düğümlerden oluşmaktadır. Bu rotalar doğrultusunda, müşteri siparişlerine uygun şekilde araçlara ürün yerleştirilmiştir. Ürünlerin müşterilere teslimatı için araçların bir alt kümesi kullanılmıştır. Her araç farklı palet kapasitelerine sahip olabileceğinden, araç türlerine göre kapasite bilgileri ayrı ayrı tanımlanmıştır. Kapasite uygunluğunu sağlamak amacıyla, araçlar her bir müşterinin sipariş talebine göre tahsis edilmiştir. Bazı durumlarda tüm talepleri karşılamak için yeterli araç bulunmadığından, araçların birden fazla tur yapmasına izin verilmiştir.

Başlangıç çözümleri, araç kapasiteleri ve rota planlama gereksinimleri gibi kritik kısıtları dikkate almadığında uygulanabilirliğini kaybedebilir. Bu tür geçersizlik durumlarını çözmek için cezalandırma yöntemleri (Ceza 1, Ceza 2 ve Ceza 3) uygulanmıştır. Bu yöntemler, çözümlerin belirtilen tüm kısıtlara ve gereksinimlere uygun hale gelmesini sağlamaktadır. Çözüm oluşturma sürecinde öncelikle kısıtlar eklenmiş ve araç kapasitesinin aşılması durumunda Ceza 1 uygulanmıştır. Kapasite bilgileri, araçları temsil eden veri yapısında saklanmıştır. Dağıtım sürecinde, her aracın kapasitesinin kontrol edilmesini sağlayan bir mekanizma kurulmuştur. Teslimat sırasında, her ürünün hangi araca yükleneceği bu kısıtlara göre belirlenmiştir. Her aracın kapasitesinin aşılmaması sağlanarak, dağıtım buna göre optimize edilmiştir. Ürünler en büyük kapasiteye sahip araçtan başlayarak araçlara yerleştirilmiştir. Müşteri talebine göre araç kapasitesinin aşılması durumunda, yükleme diğer araçlara yapılmıştır. Teslimat işlemini gerçekleştiren aracın depoya dönüş durumu da kontrol edilmiştir. Dağıtım sürecini tamamlayıp depoya dönen araçlar arasında öncelik, araçların kapasite ve yük dengesi göz önünde bulundurularak belirlenmiştir. En yüksek yük taşıma kapasitesine sahip ve yeniden yüklenebilir araçlara öncelik verilmiştir. Bu kapsamda, tüm GPS verilerinde, farklı araç gruplarının gün içerisindeki depoya dönüş davranışlarını incelemek amacıyla, dönüş sürelerinin dağılımı violin grafiği ile analiz edilmiştir. Şekil 47’de, dört farklı araç grubunun (Sevkiyat Araçları, Depo Araçları, Satış Ekibi, Nakliye Araçları) gün içerisinde depoya dönüş sürelerinin dağılımı violin grafiği ile görselleştirilmiştir. Yatay eksen, araçların depoya dönüş sürelerini saniye cinsinden gösterirken; dikey eksen ise her bir araç grubu yer almaktadır. Her bir violin şekli, ilgili gruba ait dönüş süresi dağılımının yoğunluğunu, simetrisini ve yayılımını ortaya koymuş, ortasındaki kutu grafikler (boxplot), medyan ve çeyrek değerler hakkında bilgi vermiştir. Grafikte, bazı grupların dönüş süreleri oldukça dar bir aralıkta yoğunlaşmışken, bazı gruplarda yayılımın oldukça geniş olduğu gözlemlenmiştir. Bu durum,

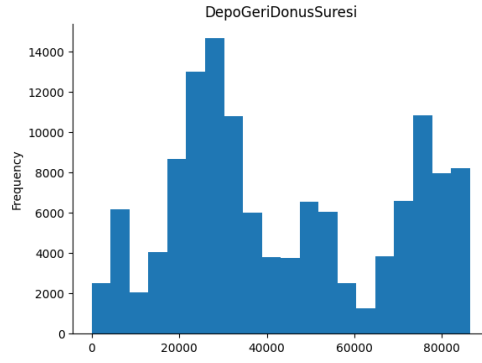
bazı araç gruplarının rotalarını daha tutarlı ve dengeli bir şekilde tamamladığını, bazı gruplarda ise dönüş sürelerinin önemli ölçüde değişkenlik gösterdiğini ortaya koymuştur. Özellikle violin şeklinin alt ya da üst kısımlarında yoğunlaşma görülen gruplarda, araçların çoğunun belirli bir zaman aralığında depoya döndüğü anlaşılmış; buna karşın şeklin oldukça yaygın olduğu gruplarda ise operasyonel düzensizlikler ya da rota uzunluklarındaki farklılıkların etkili olduğu değerlendirilmiştir. Sevkiyat Araçları grubunda ise dağılım oldukça simetrik ve geniştir. Geri dönüş süreleri yaklaşık 20,000 ila 80,000 saniye arasında değişmekte olup, medyan değer yaklaşık 50,000 saniye civarındadır (yaklaşık 13,8 saat). Dağılımın ortasında yoğunluğun fazla olması, çoğu aracın bu zaman aralığında depoya döndüğünü göstermiştir. Bu sonuç, Sevkiyat Araçları'nın gün boyunca geniş bir zaman diliminde dönüş yaptığını ve ortalamanın üzerinde sürelerle sahip olduğunu ortaya koymuştur. Bu durum, bu araçların çok sayıda müşteriye uğradığını, rotalarının uzun olduğunu veya tekrar sahaya çıkmadan önce geç döndüklerini düşündürmüştür. Bu bağlamda, Şekil 48'de, araçların depoya dönüş sürelerinin dağılımı histogram grafiği ile görselleştirilmiştir. Yatay ekseninde dönüş süreleri saniye cinsinden yer almakta; dikey ekseninde ise her bir süre aralığında gözlenen araç sayısı (frekans) gösterilmiştir. Grafik, dönüş sürelerinin büyük bir kısmının 20,000 ila 40,000 saniye aralığında yoğunlaştığını ortaya koymuş, bu durum dağıtıma çıkan araçların önemli bir bölümünün yaklaşık 5,5 ila 11 saat içerisinde depoya döndüğünü göstermiştir. Bununla birlikte, 60,000 saniyeyi aşan ve 80,000 saniyeye kadar uzanan süre aralığında da anlamlı bir yoğunluk gözlenmiştir. Bu durum, bazı araçların oldukça uzun rotalarda görev aldığını veya sahada beklemeye bağlı olarak geç döndüğünü düşündürmüştür. Daha düşük süreli dönüşlerin (örneğin 0–10,000 saniye arası) az sayıda gözlemlenmesi ise kısa mesafeli teslimatlar ya da iptal edilen görevler gibi istisnai operasyonel durumlara işaret etmiştir. Elde edilen bu dağılım, filo genelinde rotaların yoğunlukla orta uzunlukta planlandığını; ancak bazı uç değerlerin zaman açısından maliyetli dağıtımlara işaret ettiğini ortaya koymuştur. Bu analiz, araçların zaman yönetimi performansı hakkında bütünsel bir bakış sunmuştur. Şekil 47'de görüldüğü gibi, Depo Araçları grubunda dağılım görece daha dar ve dengelidir. Geri dönüş süreleri yaklaşık 0 ila 60,000 saniye arasında değişmekte olup, medyan değer yaklaşık 35,000 saniye civarındadır (yaklaşık 9,7 saat). Diğer gruplara kıyasla uç değerlerin daha az olması, bu grubun dönüş sürelerinin daha öngörülebilir ve operasyonel açıdan daha istikrarlı olduğunu göstermiştir. Bu durum, depo araçlarının daha kısa mesafeli görevlerle sınırlandırılmış olmasından kaynaklanıyor olabilir. Satış Ekibi grubunda dağılım, tüm gruplar arasında en dar olanıdır. Geri dönüş süreleri yoğunlukla 10,000 ila 40,000 saniye arasında (yaklaşık 2,8 ila 11 saat) yoğunlaşmış olup, medyan değer yaklaşık 30,000 saniyedir (yaklaşık 8,3 saat). Violin şeklinin simetrik olması, satış ekiplerinin dönüş sürelerinde tutarlılığı göstermiştir. Bu düzen, büyük olasılıkla planlı müşteri ziyaretlerine

dayalı, sabit ve kontrollü saha görevlerinden kaynaklanmaktadır. Nakliye Araçları grubu ise en geniş dağılıma sahip gruptur. Geri dönüş süreleri neredeyse 0'dan 90,000 saniyeye kadar (yaklaşık 25 saat) yayılmakta olup, medyan değer yaklaşık 40,000 saniye (yaklaşık 11,1 saat) civarındadır. Şeklin asimetrik yapısı ve uç değerlerin oldukça uzun sürelerle işaret etmesi, bu gruptaki bazı araçların çok uzak mesafelere gönderildiğini veya rotalarında duraksamalar yaşandığını göstermiştir. Bu yüksek varyasyon, lojistik planlamada dikkatle ele alınması gereken önemli bir operasyonel riske işaret etmiştir. Genel olarak grafik, araç gruplarının operasyonel görev tanımlarına göre farklı dönüş süreleri profillerine sahip olduğunu açıkça ortaya koymuştur. Satış Ekibi grubu en tutarlı ve öngörülebilir dönüş sürelerine sahipken, Nakliye Araçları grubu oldukça dağınık ve uç değerlere sahip bir dağılım sergilemiştir. Depo ve Sevkiyat Araçları grupları ise orta seviyede bir yayılıma sahiptir; ancak özellikle Sevkiyat Araçları'nda üst limitlerin dikkat çekici şekilde yüksek olduğu gözlemlenmiştir. Bu analizleri desteklemek amacıyla, araçların palet kapasitelerine göre dağıtıma katılım sıklığı çubuk grafik ile görselleştirilmiştir. Şekil 49'da, dağıtıma katılan araçların sahip oldukları palet kapasitesine göre frekans dağılımı yatay çubuk grafik biçiminde sunulmuştur. Yatay eksen, her bir kapasite grubunun dağıtıma kaç kez katıldığını göstermiş, dikey eksen ise palet kapasitesi değerleri yer almıştır. Grafikten elde edilen bulgulara göre, en fazla kullanılan araçların 8 palet kapasiteli olduğu, bunu sırasıyla 14 ve 26 paletlik araçların takip ettiği görülmüştür. Düşük kapasiteli araçlar (2, 4 ve 6 palet) ise daha sınırlı sayıda kullanılmıştır. Bu durum, dağıtım sürecinde daha fazla ürünü tek seferde taşıyabilen araçlara öncelik verildiğini ve operasyonun daha az tur ile tamamlanmasının hedeflendiğini göstermiştir. Bu analizleri daha ayrıntılı değerlendirmek amacıyla, araç grupları ile palet kapasiteleri arasındaki ilişkileri gösteren bir ısı haritası hazırlanmıştır. Şekil 50'de, yatay eksen araç grupları, dikey eksen ise araçlara ait palet kapasiteleri yer almış, her hücredeki renk yoğunluğu ilgili gruptaki araçların belirli bir palet kapasitesinde kaç kez kullanıldığını göstermiştir. Görsel bulgular, özellikle Sevkiyat Araçları grubunun 8 ve 14 palet kapasiteli araçlarla yoğun biçimde temsil edildiğini ortaya koymuştur. Nakliye Araçları grubunda yalnızca 26 palet kapasiteli araçların kullanılması, bu grubun büyük hacimli dağıtımlarda görev aldığını göstermiştir. Depo Araçları ve Satış Ekibi gruplarında ise sınırlı kapasite çeşitliliği gözlemlenmiştir. Bu durum, her grubun operasyonel rolüne göre farklı taşıma kapasitesi yapılarıyla optimize edildiğini ve sistemin toplam verimliliğini artırmak üzere kapasite-grup eşleşmesinin bilinçli olarak yapılandırıldığını ortaya koymuştur. Ayrıca, araçların sahip olduğu palet kapasiteleri ile depoya dönüş süreleri arasındaki ilişkiyi incelemek amacıyla violin grafiği oluşturulmuştur. Şekil 51'de sunulan bu grafik, her bir palet kapasitesine sahip araç grubunun dönüş süreleri bakımından gösterdiği dağılımı görselleştirmiştir. Elde edilen sonuçlara göre, özellikle 8 ve 14 palet kapasiteli araçların dönüş sürelerinde geniş bir

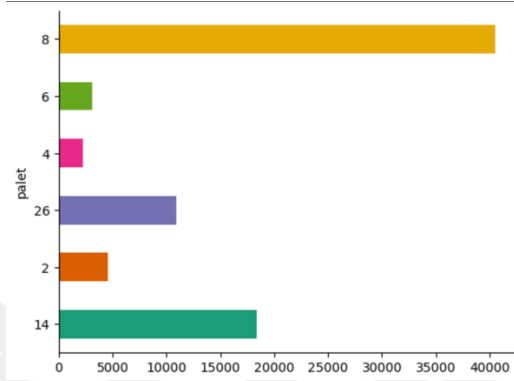
yayılım olduğu ve bu araçların hem kısa hem uzun rotalarda görev aldığı anlaşılmıştır. Öte yandan, 2 ve 26 palet kapasiteli araçların dönüş sürelerinin görece daha kısa ve dar bir aralıkta yoğunlaştığı gözlemlenmiştir. Bu durum, 2 palet kapasiteli araçların kısa mesafeli teslimatlarda; 26 paletlik araçların ise büyük hacimli ancak optimize rotalarda kullanıldığını düşündürmüştür. Araç gruplarının gün içerisindeki dönüş davranışlarını zamansal olarak değerlendirmek amacıyla, zaman eksenine karşılık dönüş sürelerini gösteren çizgi grafik oluşturulmuştur (Şekil 52). Grafik, farklı grupların belirli saat aralıklarında depoya dönüş eğilimlerini ortaya koymuştur. Özellikle Sevkiyat Araçları'nın gün boyunca aktif olduğu, yoğunlukla yüksek sürelerle sahada kaldığı ve geç saatlerde dönüş yaptığı gözlemlenmiştir. Buna karşın, Satış Ekibi ve Depo Araçları gruplarının daha kısa süreli ve erken saatlerde dönüş gerçekleştirdiği anlaşılmıştır. Nakliye Araçları ise gün boyunca geniş bir zaman aralığında dönüş yapmış olup, dönüş sürelerinde büyük değişkenlik sergilemiştir. Araç gruplarının toplam kat ettikleri yol uzunluklarını analiz etmek amacıyla oluşturulan Şekil 53, her grubun operasyonel yoğunluğunu ve kullanım farklılıklarını ortaya koymuştur. Özellikle Nakliye Araçları grubunun çok geniş bir dağılıma sahip olması, bu araçların hem mesafe hem görev sıklığı açısından sistemde en yüksek operasyonel yükü taşıdığını göstermiştir. Bu gruba ait araçların toplamda kat ettiği yol uzunluğu 0 ila 1,000,000 km arasında değişmekte olup, medyan değer yüksek olması ve dağılımın oldukça yaygın olması, bu grubun çok uzun rotalarda ve yüksek frekansta kullanıldığını göstermiştir. Sevkiyat Araçları grubunda ise yol uzunlukları daha çok 200,000 ila 800,000 km arasında yoğunlaşmakta, bu da bu grubun hem kısa hem uzun rotalarda görev aldığını ve yüksek operasyonel yoğunluk taşıdığını göstermiştir. Depo Araçları grubunda dağılım daha dar olup, genellikle 100,000 ila 400,000 km aralığında toplanmıştır; bu durum, bu grubun daha sabit görevlerle ve sınırlı mesafelerde çalıştığını düşündürmüştür. Satış Ekibi grubunun ise yol uzunlukları 0 ila 200,000 km arasında sıkışmış ve dağılım oldukça dar bir aralıkta yoğunlaşmıştır. Bu durum, satış ekiplerinin sabit, kısa mesafeli ve planlı müşteri ziyaretleri kapsamında çalıştığını ve oldukça öngörülebilir bir operasyonel yapıya sahip olduğunu ortaya koymuştur.



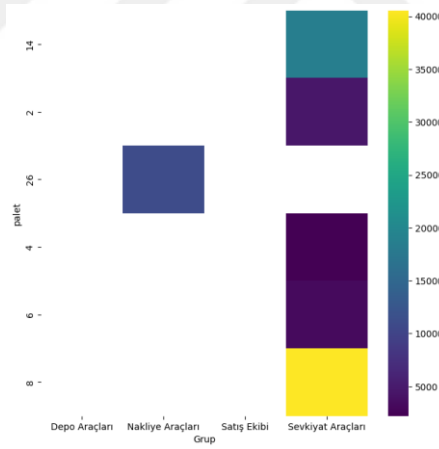
Şekil 47. Araç gruplarına göre depoya geri dönüş süresi dağılımları



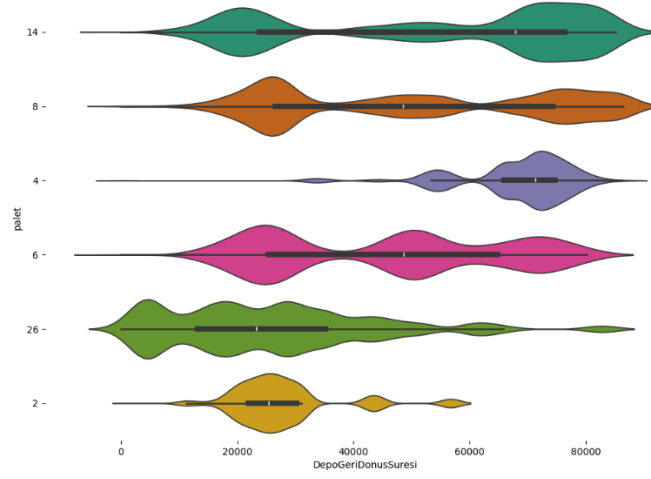
Şekil 48. Depoya geri dönüş süresi değişkeninin histogram grafiği



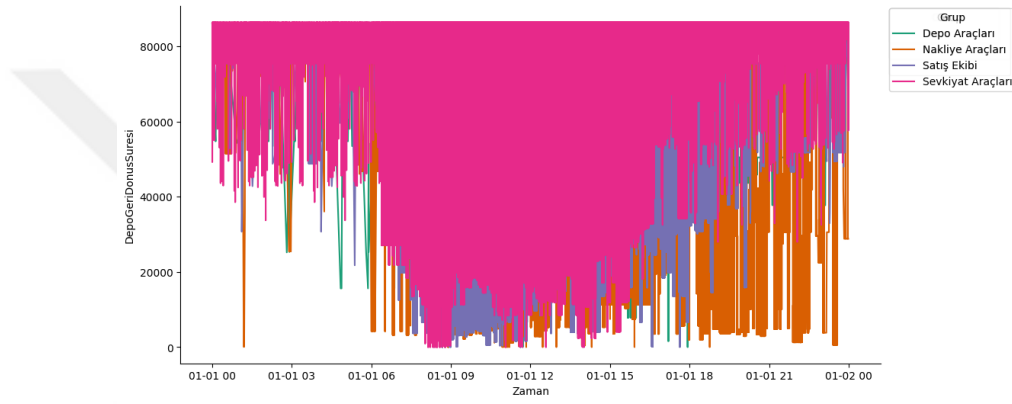
Şekil 49. Araç palet kapasitelerine göre toplam yükleme miktarları



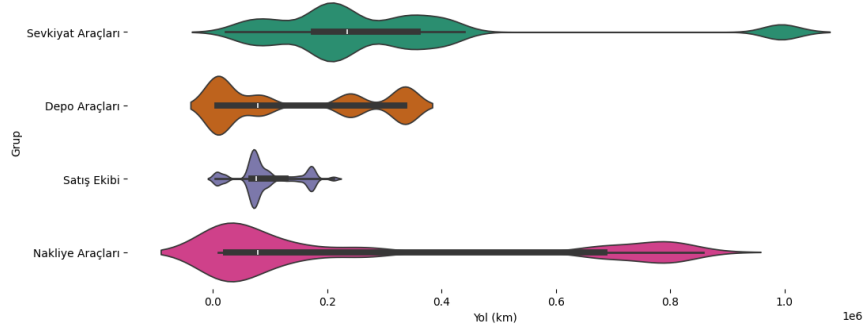
Şekil 50. Araç grubu ve palet kapasitesine göre yükleme miktarlarının ısı haritası



Şekil 51. Araç palet kapasitelerine göre depoya geri dönüş süresi dağılımı



Şekil 52. Günlük zaman dilimine göre araçların depoya geri dönüş süreleri

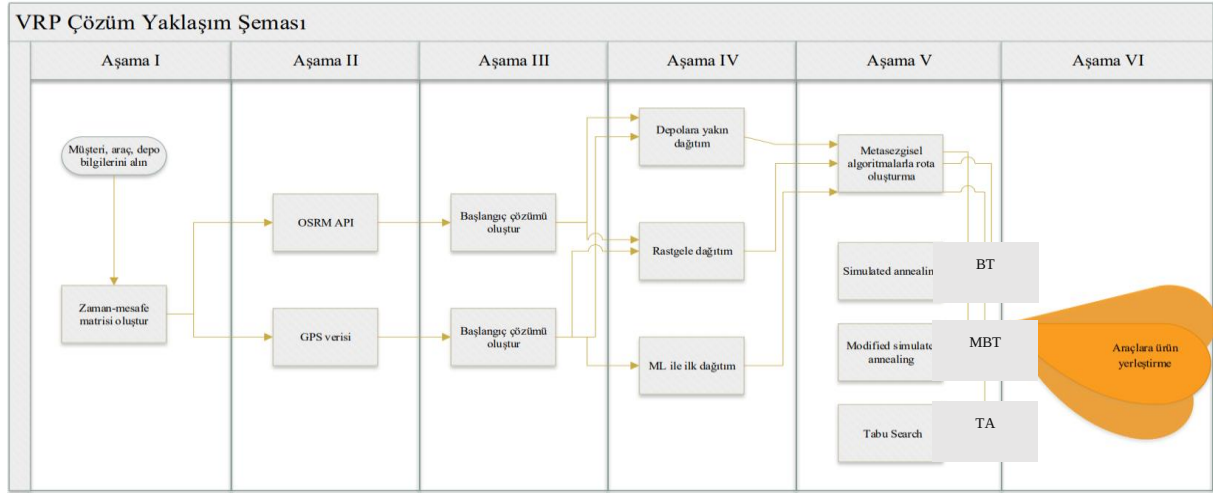


Şekil 53. Araç gruplarına göre kat edilen mesafe dağılımı (yol (km))

Rota planlamasında, depo sıralamasının başlangıçtan ikinci veya sondan ikinci sırada olmaması ve iki deponun yan yana bulunması durumunda Ceza 2 uygulanmıştır. Ayrıca, depo konumlarının rota başlangıcı ve sonunda olmaması durumunda Ceza 3 devreye girmiştir.

Önerilen tedarik zinciri senaryosunda (Şekil 54), her depo mevcut bir müşteri segmentine hizmet etmek üzere tanımlanmıştır. Ürünler, ana merkez depodan diğer depolara dağıtılmış ve ardından müşterilere teslim edilmiştir. Rota üzerindeki müşteri listesi gözden geçirilirken, her depo numarasına ait siparişler ilgili araçlara yüklenmiş ve dağıtılmıştır. Bu

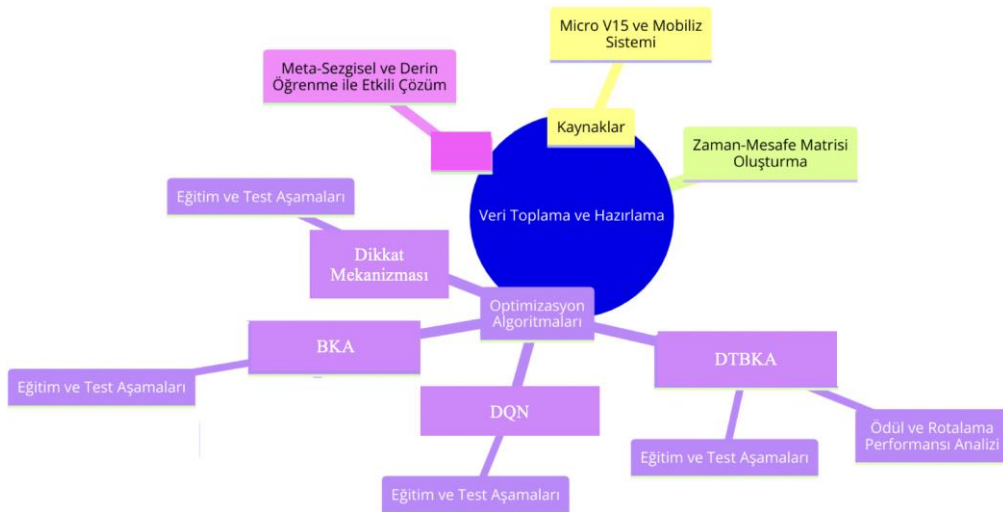
siparişler, en yüksek verimliliği sağlamak amacıyla uygun araçlara tahsis edilmiştir. Siparişler için yeterli araç bulunmaması durumunda, araçlar depolara döndükçe önceliklendirilerek yeniden yükleme yapılmıştır. Bu yöntem, hem araç kapasite kısıtlarını hem de rota planlama gereksinimlerini göz önünde bulundurarak optimize edilmiş bir dağıtım süreci sunmaktadır (Erdem *et al.* under review b).



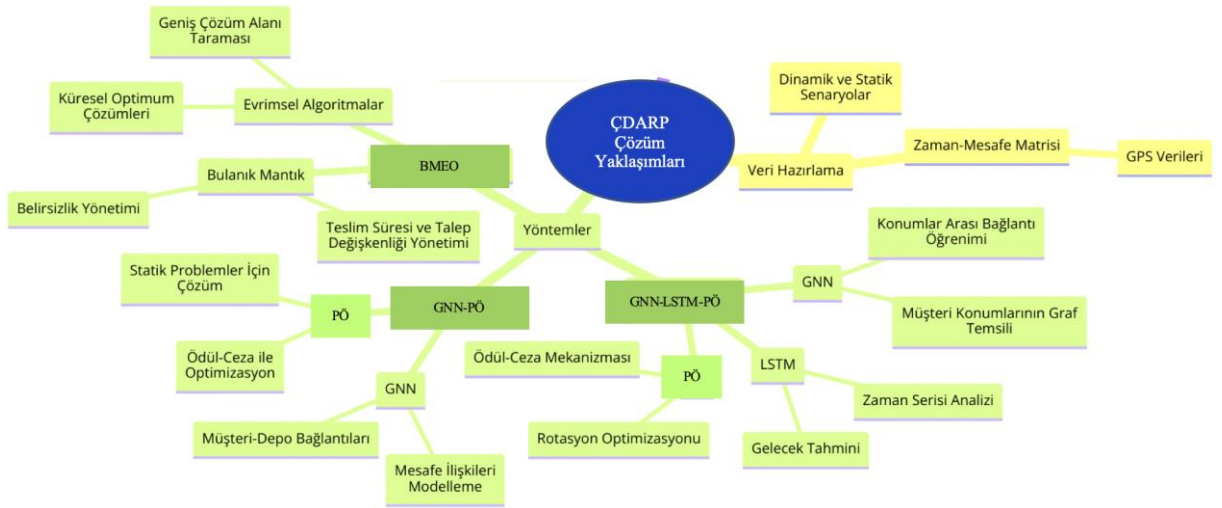
Şekil 54. ÇDARP çözüm aşamaları (Erdem *et al.* under review b)

Takviye öğrenme, DL ve Meta-Sezgisel algoritmaların yenilikçi entegrasyonlarıyla ÇDARP optimizasyonu

Tablo 22'deki işlem adımları temel alınarak GPS verileri oluşturulmuştur. Bu veriler, BKA, DQN, Dikkat Mekanizması, DTBKA, GNN-LSTM-RL, GNN-RL ve Bulanık Mantık Evrimsel Optimizasyon (BMEO) olmak üzere yedi farklı algorithmada kullanılmıştır (Şekil 55, 56). Algoritmalar, tahmin edilen rotalar ile gerçek rotalar arasındaki hata oranını ölçmüştür. Elde edilen sonuçlar, bu ölçümler doğrultusunda optimize edilmiştir (Erdem *et al.* under review d).



Şekil 55. ÇDARP çözüm yaklaşımlarına yönelik ileri seviye algoritmalar (BKA, DQN, Dikkat Mekanizması ve DTBKA)



Şekil 56. ÇDARP çözüm yaklaşımlarına yönelik hibrit Algoritmalar (GNN-LSTM-RL, GNN-RL ve BMEO)

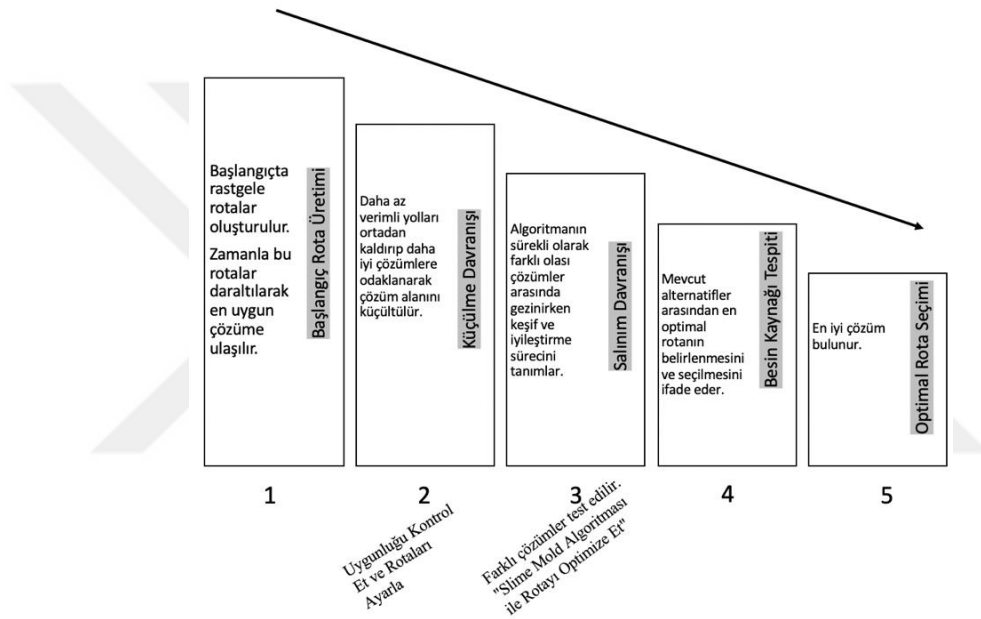
✓ BKA

BKA, mikroorganizmalardan, özellikle de cıvık mantar (slime mold) canlılarından ilham alınarak geliştirilmiş doğa temelli bir optimizasyon algoritmasıdır. BKA, cıvık mantarın besin kaynaklarına doğru hareket ederken izlediği yolları ve davranışlarını taklit eder. Bu süreçte slime mold, rotalarını sürekli güncelleyip optimize ederek hedefe doğru ilerler. Slime mold'un hedefe ulaşırken dinamik olarak uyum sağladığı bu strateji, ÇDARP gibi problemler için en kısa veya en verimli rotayı bulmada kullanılabilir. BKA birkaç aşamadan oluşur (Erdem *et al.* under review d):

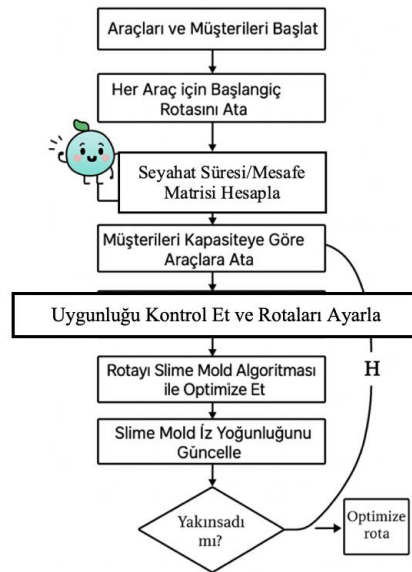
- ✓ Slime mold başlangıç yolu: Slime mold algoritmasında çözüm arayışına başlamadan önce, her aracın ve müşterinin başlangıç durumu tanımlanarak algoritmanın ilk popülasyonu oluşturulur. Bu aşamada, her araç için başlangıç rotası belirlenir ve bu rotalar algoritmanın başlangıç bireyleri olarak değerlendirilir. Bu aşama, Şekil 58'de "Araçları ve Müşterileri Başlat" ve "Her Araç için Başlangıç Rotasını Ata" adımları ile gösterilmiştir. Algoritma, optimizasyon sürecinin ilk adımı olarak rastgele bir başlangıç yolu oluşturur. Bu ilk yol, zamanla optimize edilir.
- ✓ Daralma davranışı: Şekil 58'de "Uygunluğu Kontrol Et ve Rotaları Ayarla" adımıyla temsil edilir. Bu aşamada algoritma, daha az verimli yolları daraltarak çözüm uzayını küçültür ve daha iyi çözümlere odaklanır. Şekil 57'de daralan yollar bu davranışı gösterir (2 numaralı adım).
- ✓ Salınım davranışı: Şekil 58'de "BKA ile Rotayı Optimize Et" adımıyla açıklanmıştır. Slime mold'un sürekli hareket edip keşif yapması, her bireyin kendi rotasını takip etmesi ve diğerleriyle etkileşimde bulunması bu aşamada gerçekleşir.

Şekil 57’de salınım davranışı, sinüzoidal desenlerle gösterilir (3 numaralı adım). Bu davranış, organizmanın besin kaynaklarına yönelme yeteneğini simüle eder.

- ✓ Besin kaynağı tespiti: Bu aşama, Şekil 58’de "Seyahat Süresi/Mesafe Matrisini Hesapla" ve "Müşterileri Kapasiteye Göre Araçlara Ata" adımlarıyla temsil edilir. Slime mold, mevcut seçenekler arasından en iyi besin kaynağını (rota) seçer. Şekil 57’de, Slime mold’un keşfinde yoğun düğümlerle gösterilir (4 numaralı adım).
- ✓ Optimal yol seçimi: Bu aşama, algoritmanın en verimli çözümü bulduğu ve optimize ettiği süreçtir. Şekil 58’de "Yakınsadı mı?" ve "Optimize rota" adımlarıyla gösterilir. Şekil 57’de bu aşama, Slime mold’un en iyi rotayı belirlediği ve algoritmanın sonuçlandığı nokta olarak gösterilir (5 numaralı adım).



Şekil 57. BKA akış diyagramı: Rota keşfi ve optimizasyonu (Erdem *et al.* under review d)



Şekil 58. Araç rota optimizasyonu için BKA tabanlı yaklaşım (Erdem *et al.* under review d)

ÇDARP'nin çözümü için BKA'nın genel adımları aşağıdaki şekilde matematiksel olarak ifade edilebilir: N müşteri sayısını, V araç sayısını, D her müşterinin talebini, C her aracın kapasitesini, T iki lokasyon (depo veya müşteri) arasındaki süreyi, d konumsal kodlamanın boyutunu ve α ise optimizasyon sürecindeki farklı bileşenler arasında denge sağlayan bir öğrenme ya da ağırlık faktörünü temsil eder. Süreç, her araç i için rastgele rotalar R_i başlatılarak ($i=1,2,...,V$), keşif ve sömürüyü dengelemek için başlangıçta bir slime mold iz yoğunluğu λ belirlenerek başlar. Araçların başlangıç konumu depoya yerleştirilir ve her bir araca ait başlangıç kapasitesi C_i olarak tanımlanır. Amaç, tüm rotalar için toplam mesafeyi veya süreyi minimize etmektir. Burada $T_{i,j}$, lokasyon i ile j arasındaki seyahat süresini ifade eder (Denklem 1). Her müşteri k için, eğer $D_k \leq C_i$ ise müşteri k numaralı araca atanır; aksi takdirde müşteri atanmadan sonraki iterasyonu bekler. Her araç için, toplam müşteri talebinin aracın kapasitesini aşmadığı geçerli bir rota R_i oluşturulur (Denklem 2). Optimizasyon aşaması, slime mold izleri arasındaki etkileşim tarafından yönlendirilir; bu süreçte yeni rotalar keşfedilir ve mevcut rotalar iyileştirilir. Mevcut çözümün kalitesine (toplam seyahat süresiyle ölçülen) bağlı olarak, algoritma uyarlanabilir bir mekanizma kullanarak rotaları günceller ve slime mold iz yoğunluğu λ 'yı zamanla güncelleyerek keşif-sömürü dengesini etkiler. λ 'nın güncelleme kuralı Denklem 3 ile ifade edilebilir (Erdem *et al.* under review d).

$$\min \sum_{i=1}^V \sum_{j=1}^N T_{i,j} \quad (1)$$

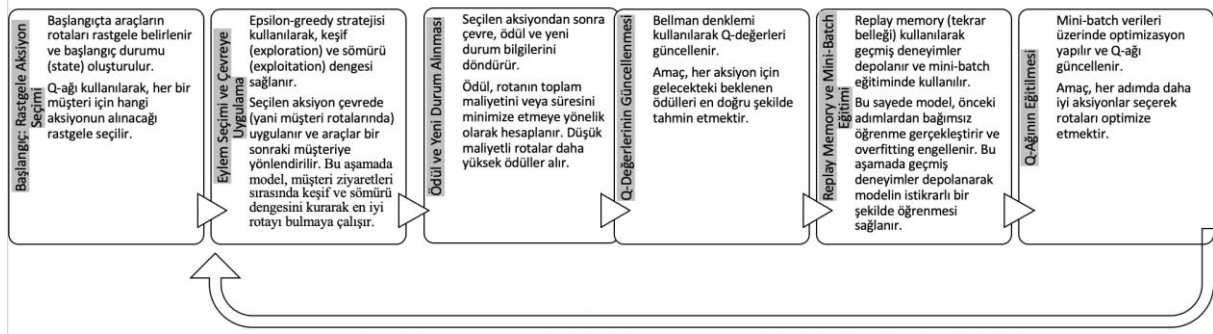
$$\sum_{k \in R_i} D_k \leq C_i \quad R_i = \{k \mid D_k \leq C_i\} \quad (2)$$

$$\lambda(t+1) = \lambda(t) \times \alpha + (1 - \alpha) \cdot \text{ÇözümKalitesi} \quad (3)$$

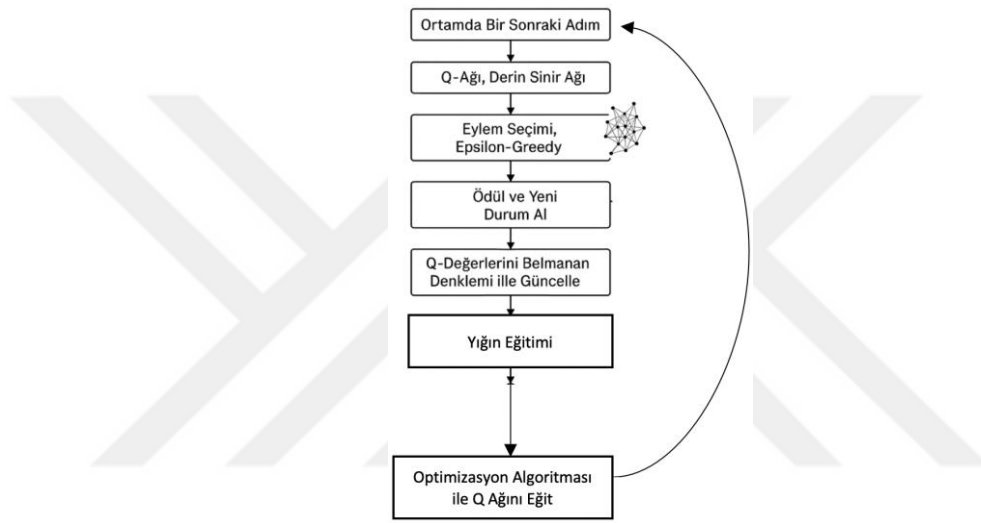
✓ DQN

DQN algoritması, belirli bir durumda (state) belirli bir eylemin (action) gelecekte sağlayacağı ödülleri tahmin etmek için derin sinir ağlarını kullanan bir RL yönteminin bir uzantısıdır. Şekil 60'da DQN algoritmasının temel bileşenleri görselleştirilmiştir. “Q-Ağı, Derin Sinir Ağ” kutucuğu, durum-eylem değerlerini tahmin eden sinir ağı yapısını temsil ederken, “Eylem Seçimi, Epsilon-Greedy” bölümü, keşif ve sömürü arasında denge kurarak uygun eylemin nasıl seçildiğini açıklamaktadır. Ayrıca, “Ödül ve Yeni Durum Al” aşaması, çevredeki değişimin ve ödül mekanizmasının nasıl işlediğini gösterir. Şekil 59'daki her aşama, DQN algoritmasının bir adımını açık biçimde tanımlar; başlangıçta rastgele aksiyon seçimiyle başlayan süreç, Q-değerlerinin Bellman denklemi ile güncellenmesi, Replay Memory kullanımıyla istikrarlı öğrenmenin sağlanması ve Mini-batch optimizasyonu ile Q-ağının eğitilmesiyle tamamlanır. Bu yapı, ÇDARP çözümünde karar süreçlerinin nasıl kademeli olarak geliştirildiğini net bir şekilde ortaya koymaktadır. ARP'de her adımda hangi müşteriye hizmet

edileceğine karar vermek, Q-fonksiyonuna göre en iyi eylemin seçilmesiyle gerçekleştirilir. (Erdem *et al.* under review d).



Şekil 59. DQN optimizasyon süreci



Şekil 60. Araç rota optimizasyonu için DQN (Derin Q-Ağları) tabanlı yaklaşım (Erdem *et al.* under review d)

DQN'nin ÇDARP'yi çözmek için izlediği genel adımlar, bir Q-fonksiyonu $Q(s, a)$ 'nın yaklaşık olarak hesaplanmasıyla matematiksel olarak ifade edilebilir. Bu Q-fonksiyonu, durum s 'de yapılan a eylemi sonucunda elde edilmesi beklenen toplam ödülü tahmin eder. Bu bağlamda, durum s müşteri ve araçların mevcut konumlarını temsil ederken, eylem a belirli bir müşteriye bir aracın atanmasını ifade eder. Eylem a gerçekleştirildikten sonra elde edilen ödül r ; müşteri talebi D , aracın kalan kapasitesi C ve lokasyonlar (depolar veya müşteriler) arasındaki seyahat süresi T gibi faktörlerden etkilenir (çevrede eylem gerçekleştirme ve ödül ile yeni durum alma aşamaları). Q-değerinin güncellenmesi Bellman Denklemi (Denklem 4) ile yapılır. Burada α , yeni bilginin mevcut Q-değerleri üzerindeki etkisini belirleyen öğrenme oranıdır; γ ise gelecekteki ödüllerin etkisini ayarlayan indirim katsayısıdır (Discount factor). $\max_{a'} Q(s', a')$, sonraki durum s' için elde edilebilecek en yüksek Q-değerini ifade ederek en iyi gelecekteki eylemi gösterir. Eğitim sürecini iyileştirmek amacıyla, deneyim tekrar belleği kullanılır. Bu bellek, geçmişteki durum-eylem-ödül geçişlerini (s, a, r, s') depolar ve bunları

rastgele örnekleyerek ardışık öğrenme adımları arasındaki korelasyonu kırar. Böylece eğitim süreci daha kararlı hale gelir. ÇDARP için DQN'nin optimizasyon hedefi, toplam seyahat süresi veya mesafesini en aza indirecek rotalar oluşturmaktır (Denklem 5). Q-ağını eğitmek için kullanılan kayıp fonksiyonu, hedef Q-değeri y_t ile tahmin edilen Q-değeri $Q(s_t, a_t)$ arasındaki ortalama kare hata (MSE) esasına dayanır (Denklem 6). Bu durumda θ , Q-ağının parametrelerini; θ^- ise daha seyrek güncellenerek eğitim kararlılığını artıran hedef ağı parametrelerini temsil eder. DQN algoritması, keşif ve sömürü arasında denge kurarak rotaları yinelemeli şekilde geliştirir. Bu geliştirme süreci, belirli bir iterasyon sayısına ulaşılması veya tatmin edici bir çözüme yakınsama gibi bir durdurma kriterine ulaşılan kadar devam eder (Denklem 7) (Erdem *et al.* under review d).

$$Q(s, a) \leftarrow Q(s, a) + \alpha (r + \gamma \max_{a'} Q(s', a') - Q(s, a)) \quad (4)$$

$$\min \sum_{i=1}^V \sum_{j=1}^N T(s_i, s_j) \quad (5)$$

$$L(\theta) = E \left[\exp \left((y_t - Q(s_t, a_t; \theta))^2 \right) \right] \quad (6)$$

$$y_t = r_t + \gamma \max_{a'} Q(s_{t+1}, a'; \theta^-) \quad (7)$$

✓ Dikkat Mekanizması

Dikkat Tabanlı Mekanizma, ML'de modelin giriş verisinin belirli bölümlerine odaklanmasını sağlayan güçlü bir mekanizmadır. Bu model, çeşitli öğelerin önem derecelerini belirleyerek dikkat ağırlıkları atar ve bu sayede karmaşık ilişkileri yakalayabilir. ÇDARP gibi optimizasyon problemlerinde dikkat mekanizmaları, müşteri talepleri ile araç kapasiteleri arasındaki eşleşmeleri verimli bir şekilde yaparak daha optimal rotalama çözümleri sağlar. Çok başlı dikkat sayesinde model, çeşitli bağımlılıkları aynı anda analiz edebilir ve bu da karmaşık ve dinamik ortamlarda daha etkili çözümler sunar. Dikkat tabanlı algoritmanın ÇDARP'yi çözmek için genel adımları matematiksel olarak aşağıdaki şekilde açıklanabilir (Erdem *et al.* under review d):

- ✓ Konumsal Kodlama: Müşteri ve araçların pozisyon bilgileri vektörler hâlinde modele eklenir. Burada, pos_n müşteri ya da araç n 'yi temsil eder ve d , pozisyonel kodlamanın boyutudur. j ise vektör boyutundaki indeksleri gösterir (Denklem 8). Bu adım, modelin müşteri ve araçlar arasındaki mekânsal ilişkileri anlamasına yardımcı olur.

$$PE(i, 2j) = \sin(i/10000^{2j/d}) \text{ and } PE(i, 2j + 1) = \cos(i/10000^{2j/d}) \quad (8)$$

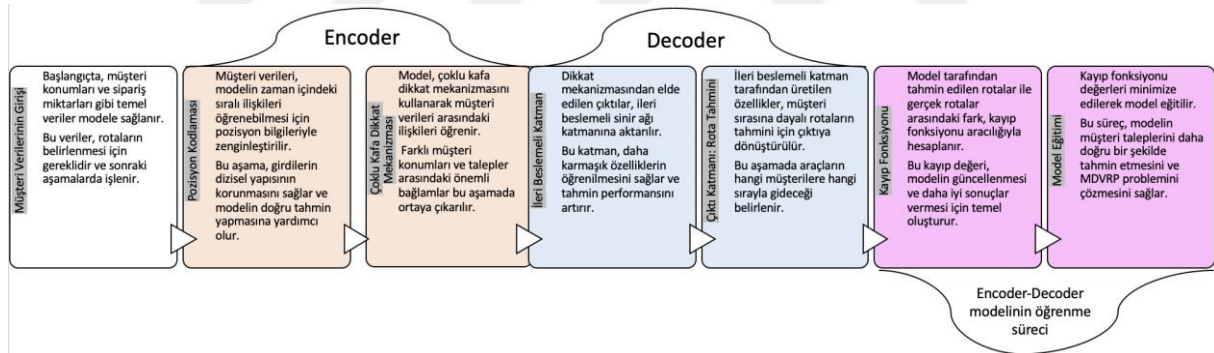
- ✓ Dikkat Mekanizması: Bu aşamada müşteri talepleri ve konumları arasındaki ilişkiler, Q (query), K (key) ve V (value) matrisleriyle tanımlanır ve ölçeklenmiş iç

çarpım dikkat hesaplamaları yapılır (Denklem 9). Çok başlı dikkat yapısı, h farklı dikkat mekanizmasını aynı anda çalıştırarak farklı bağımlılıkları öğrenir. Bu yapı Denklem 14 ile gösterilir. Burada W_O çıkış ağırlık matrisi, $head_i$ ise her bir dikkat başlığını ifade eder.

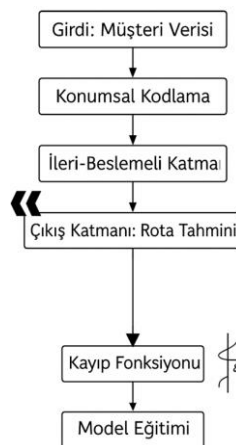
$$Attention(Q, K, V) = \text{softmax}((QK^T)/\sqrt{d_k})V \quad (9)$$

- ✓ İleri Beslemeli Katman: Dikkat mekanizmasından sonra, çıktı verileri ileri beslemeli sinir ağına aktarılır (Denklem 15). Burada W_1 , W_2 ağırlık matrisleri; b_1 , b_2 ise bias terimleridir.
- ✓ Çıkış Katmanı (Rota tahmini): Modelin çıktısı, her araç için optimize edilmiş rotalardır. Bu rotalar, toplam seyahat süresini en aza indirecek şekilde seçilir ve aynı zamanda müşteri talepleri D ve araç kapasite kısıtları C ile uyumlu olacak biçimde belirlenir (Kayıp Fonksiyonu).
- ✓ Model Eğitimi: Model, geri yayılım ve ağırlık güncellemeleri ile kayıp fonksiyonunu minimize etmek üzere eğitilir.

Bu yapı, ÇDARP'nin çözümünde dikkatli, veriye duyarlı ve güçlü sonuçlar üreten modern bir yaklaşımı temsil eder.



Şekil 61. Dikkat mekanizması optimizasyon süreci



Şekil 62. Araç rota optimizasyonu için dikkat tabanlı yaklaşım

Yukarıda sunulan dikkat tabanlı mekanizmasının teorik temelleri ve matematiksel açıklamaları, görsellerle somut bir şekilde desteklenmiştir (Şekil 61 ve Şekil 62). Şekil 61, Encoder-Decoder mimarisinin ÇDARP çözüm sürecine nasıl entegre edildiğini ayrıntılı olarak göstermektedir. Müşteri verilerinin girişinden başlayarak pozisyonel kodlama, çoklu dikkat mekanizmaları, ileri beslemeli katmanlar ve kayıp fonksiyonu hesaplamalarına kadar tüm adımlar şematik olarak ifade edilmiştir. Bu yapı, modelin müşteri konumlarını ve sipariş miktarlarını dikkate alarak, araçların hangi sırayla hangi müşterilere gideceğini belirlemesini sağlar. Şekilde ayrıca modelin eğitilme süreci, tahmin edilen rotaların kayıp fonksiyonu ile karşılaştırılması ve bu farkın minimize edilmesiyle sonuçların iyileştirilmesi açıkça sunulmuştur. Şekil 62 ise dikkat tabanlı algoritmanın matematiksel bileşenlerinin akışını daha sade bir şekilde özetlemektedir. Girdi olarak müşteri verileri alınmakta, ardından pozisyonel kodlama ile mekânsal ilişkiler modellenmekte, çok başlı dikkat mekanizması uygulanmakta ve elde edilen çıktılar ileri beslemeli ağda işlenerek rota tahminleri yapılmaktadır. Son adımda kayıp fonksiyonu ile model eğitimi gerçekleştirilerek çözüm kalitesi artırılmaktadır (Erdem *et al.* under review d).

✓ DTBKA Model

DTBKA, BKA'yı başlangıç rotalarını oluşturmak için, dikkat mekanizmasını ise optimizasyon süreçlerinde kullanan hibrit bir modeldir. BKA, başlangıçta rastgele bir rota oluşturur ve bu rotayı optimize etmeye başlar. Bu süreçte, en iyi çözüm güncellenir ve optimal çözüme yakın yollar belirlenir. BKA, salınım ve daralma davranışları kullanarak bu rotaları iyileştirir ve daha verimli hale getirir. Dikkat mekanizması, bu başlangıç rotasını daha da optimize eder. Dikkat mekanizmasında, müşteri ve araçların konumsal bilgileri, Pozisyon Kodlama yöntemiyle vektörlere dönüştürülür. Bu, müşteriler ve araçlar arasındaki mekansal ilişkileri temsil eder (Erdem *et al.* under review d).

- Ölçeklenmiş İç Çarpım Dikkat aşamasında (Scaled Dot-Product Attention), müşterilerin talepleri ve konumları arasındaki ilişkileri anlamak için dikkat mekanizması kullanılır. Bu mekanizma, talepler ile pozisyonlar arasındaki ilişkileri hesaplayarak en uygun rotayı bulur.
- Çoklu Başlık Dikkat aşamasında ise birden fazla dikkat başlığı kullanılarak müşteriler arasındaki ilişkiler daha iyi öğrenilir. Bu aşama, modelin müşterilerin taleplerini daha etkili bir şekilde öğrenmesini ve rotaları iyileştirmesini sağlar (Şekil 64).

Bu hibrit modelde, BKA'nın güçlü keşif yetenekleri ile dikkat mekanizmasının detaylı öğrenme kapasitesi bir araya getirilmiştir. Böylece, hem müşteri taleplerini karşılayan hem de

araç kapasitesine uygun rotalar oluşturulur. Modelde kullanılan kayıp fonksiyonu, tahmin edilen rotalar ile gerçek rotalar arasındaki hatayı en aza indirmeyi amaçlar. Model, bu hatayı adım adım azaltarak rotaları iyileştirir. Bu süreç Şekil 63'te ayrıntılı bir şekilde gösterilmiştir. Görselde, BKA algoritmasının başlangıç rotalarını oluşturduğu ve ardından dikkat mekanizması ile rotaların optimize edildiği adımlar sırasıyla sunulmuştur. Modelin temel bileşenlerinin (pozisyon kodlama, müşteri talep işleme, dikkat mekanizması, rota güncelleme ve performans değerlendirme) nasıl bir etkileşim içinde çalıştığı açıkça görülmektedir.

DTBKA'nın ÇDARP çözümü için matematiksel formülasyonu, iki temel bileşeni bütünleştirir: BKA ve Dikkat Mekanizması Tabanlı Transformer Modeli. BKA, N müşteriye ve V araca karşılık gelen rotalardan oluşan rastgele bir başlangıç popülasyonu üretir: $R_{\text{initial}} = \{R_1, R_2, \dots, R_V\}$, Her iterasyonda (t), BKA müşteri konumlarını $X(t)$ olarak güncelleyerek yeni rotaları keşfetmeyi hedefler. Bu konumlar, salınım frekansı λ 'ya bağlı olarak osilasyon davranışı gösterir.

$X(t)$, t zamanındaki müşteri konumunu; λ ise osilasyon frekansını ifade eder (Denklem 10). Çözüm iyileştikçe, α daralma faktörü ile arama alanı küçülür ve daha hassas sonuçlara ulaşılır: α , zamanla arama alanını daraltan faktördür (Denklem 11). Her iterasyonda BKA, çözümleri toplam mesafe T üzerinden değerlendirir; burada $T(X)$, çözüm X 'in toplam mesafesini; X_{best} , t iterasyonundaki en iyi çözümü gösterir (Denklem 12). Transformer modelinde, müşteri konumlarına konumsal kodlama uygulanarak bu verilerin uzamsal temsili sağlanır. Her pozisyon pos için, boyut d ve boyut indeksi i kullanılarak $PE(pos, 2i)$ hesaplanır. Dikkat mekanizması (scaled dot-product dikkat), müşteri konumlarının ve taleplerinin önemini belirlemek için sorgular Q , anahtarlar K ve değerler V matrislerini kullanır. Burada d_k , sorgu/anahtar vektörlerinin boyutunu ifade eder ve Q, K, V müşteri ve araçlara ilişkin talep ve pozisyon bilgilerini temsil eder (Denklem 13). Öğrenme sürecini artırmak amacıyla çoklu dikkat başlıkları kullanılır. Bu sayede model, müşteri ilişkilerinin farklı yönlerini aynı anda öğrenebilir. Her bir dikkat başlığının çıktısı birleştirilip dönüşüme tabi tutulur: h , dikkat başlığı sayısını; W_o , çıktı ağırlık matrisini gösterir (Denklem 14). Dikkat mekanizmasından elde edilen çıktılar, ileri beslemeli tam bağlı katmanlardan geçirilerek tahminler daha da iyileştirilir. Her katman, aşağıdaki dönüşümü uygular: W_1, W_2 ağırlık matrisleri, b_1, b_2 ise bias terimleridir (Denklem 15). DTBKA modeli, ilk olarak BKA ile bir başlangıç çözümü oluşturur. Bu çözüm daha sonra, müşteri talepleri ve araç kapasitelerine odaklanarak rotaları optimize eden Transformer yapısı tarafından iyileştirilir. Son rotalar, her aracın müşterileri arasında toplam seyahat süresi ya da mesafesini en aza indirecek şekilde seçilir: $T(X_{ij}, X_{i(j+1)})$, araç i 'nin

rotasında müşteri j ile $j+1$ arasındaki seyahat süresini temsil eder (Denklem 16) (Erdem *et al.* under review d).

$$X(t+1) = X(t) + \Delta \cdot \sin((2\pi/T)t) \quad (10)$$

$$X(t+1) = \alpha \cdot X(t) \quad (11)$$

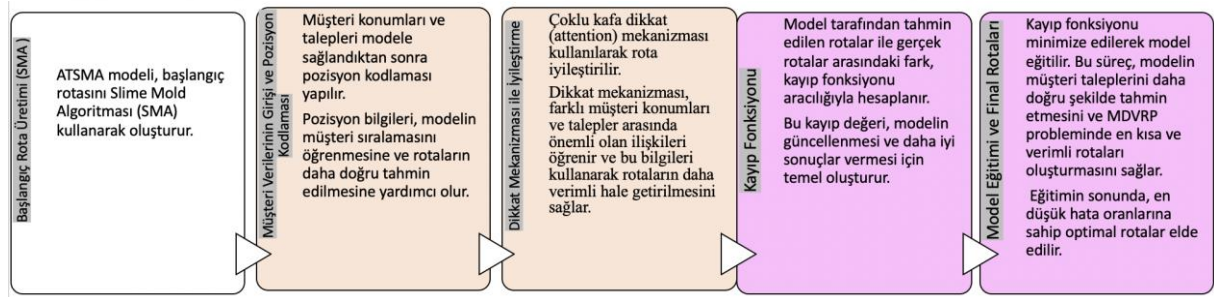
$$X_{\text{best}} = \text{argmin } T(X) \quad (12)$$

$$PE(pos, 2i) = \sin(pos/10000^{2i/d}) \text{ and } PE(pos, 2i+1) = \cos(pos/10000^{2i/d}) \quad (13)$$

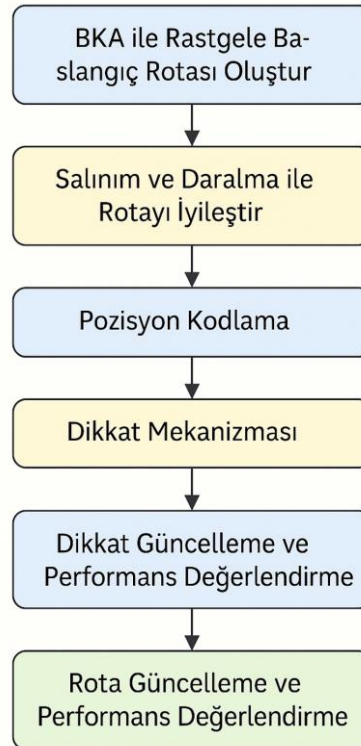
$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h) W_O \quad (14)$$

$$\text{FFN}(x) = \max(0, xW_1 + b_1) W_2 + b_2 \quad (15)$$

$$\min \sum_{i=1}^V \sum_{j=1}^N T(X_{ij}, X_{i(j+1)}) \quad (16)$$



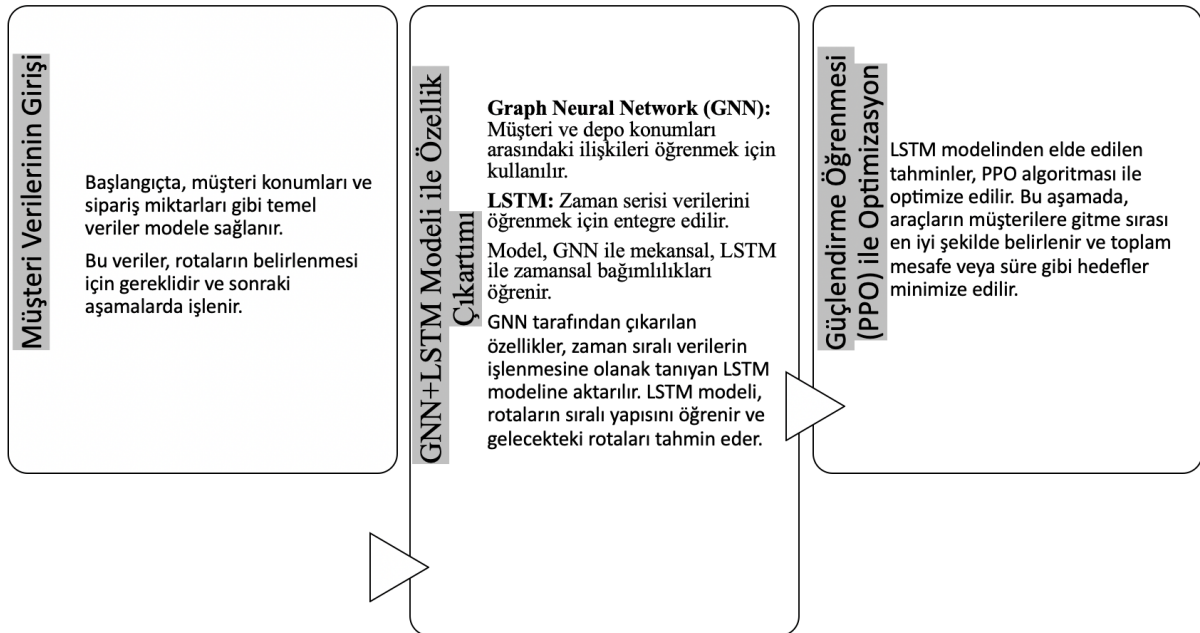
Şekil 63. DTBKA optimizasyon süreci



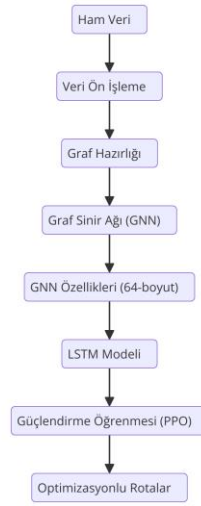
Şekil 64. Araç rota optimizasyonu için DTBKA yaklaşımı

✓ GNN-LSTM-RL Model

GNN-LSTM-RL modelinde, müşteri ve depo konumları arasındaki mekansal ilişkileri öğrenmek için GNN, zaman serisi verilerini modellemek için ise LSTM kullanılarak tasarlanmıştır. Model, GNN ile mekansal, LSTM ile zamansal bağımlılıkları öğrenerek RL ile araç rotalarını optimize eder. Veri hazırlık aşamasında, müşteri ve depo konumları arasındaki zaman-mesafe matrisleri GPS verileri ve OSRM API kullanılarak hesaplanır ve bu verilerden GNN için gerekli düğüm ve kenar özellikleri organize edilir. GNN, müşteri ve depo arasındaki ilişkileri öğrenerek her düğüm için 64 boyutlu özellikler çıkarır; bu özellikler, LSTM'ye giriş olarak zaman serisi formatında (10,64) boyutunda hazırlanır. Bu, LSTM'nin 10 zaman adımı boyunca 64 özellikten oluşan bir zaman serisi yapısı oluşturması anlamına gelir. Her zaman adımında 64 özellikten oluşan bir giriş vektörü alınır. Kodda kullanılan LSTM, her araç/müşteri için 10 zaman adımında 64 özellikten oluşan oluşturulmuş gözlemleri işler. Bu gözlemler, gerçek zaman serisi verilerinin simülasyonudur ve LSTM'nin zaman bağımlılıklarını öğrenebilmesi için bir çerçeve sağlar. LSTM, bu özellikleri zaman bağımlılıklarını modellemek için işler ve sonuçları RL ajanına aktarır. RL modeli, Stable-Baselines3 PPO algoritması ile müşteri taleplerini karşılamak üzere araç rotalarını optimize eder. Bu yapı, hem mekansal hem de zamansal bağımlılıkları dikkate alarak rota optimizasyonu için etkili bir çözüm sunar (Erdem *et al.* under review d) (Şekil 65-66).



Şekil 65. GNN-LSTM-RL optimizasyon süreci



Şekil 66. Araç rota optimizasyonu için GNN-LSTM-RL yaklaşımı

GNN-LSTM-RL modeli, müşteri ve depo düğümlerinin bağlantılarını grafik temsiliyle işleyen ve RL ile karar veren bir yapay öğrenme yaklaşımıdır. Modelin temel bileşenleri, düğüm kümesi V (müşteriler ve depolar) ve kenar kümesi E (seyahat maliyetlerini temsil eder) ile tanımlanan bir grafik $G=(V,E)$ üzerinde çalışır. İlk olarak, GNN katmanlarında düğüm temsil vektörleri güncellenir. Her katmandaki güncelleme işlemi Denklem 17’de tanımlanır. Burada $H^{(l)}$, l . katmandaki düğüm gömmesi; $W^{(l)}$, ağırlık matrisi; $b^{(l)}$, bias terimi; σ ise aktivasyon fonksiyonudur.

$$H^{(l+1)} = \sigma(W^{(l)}H^{(l)} + b^{(l)}) \quad (17)$$

GNN çıktısı, zaman bağımlı bilgileri yakalamak için LSTM yapısına aktarılır. Her zaman adımında LSTM güncellemesi Denklem 18’deki gibi yapılır. Burada, x_t anındaki girdi vektörü, h_t , gizli durum (hidden state), W_x ve W_h , öğrenilebilir ağırlık matrisleri, b ise bias terimidir.

$$h_t = f(W_x x_t + W_h h_{t-1} + b) \quad (18)$$

RL aşamasında, ajan ortamda eylemler alarak ödül elde eder ve Q -değerleri Bellman denklemi ile güncellenir. Denklem 19’deki $Q(s_t, a_t)$, belirli bir durum-eylem çiftinin değerini (beklenen toplam ödül) temsil eder: α öğrenme oranı; r , anlık ödül; γ , indirim faktörü; s' , yeni durumdur. Bu yapı sayesinde, GNN-LSTM-RL modeli hem mekansal ilişkileri (GNN) hem de zaman serisi dinamiklerini (LSTM) öğrenerek rotalama kararlarını RL temelli şekilde optimize eder.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha (r + \gamma \max_{a'} Q(s', a') - Q(s_t, a_t)) \quad (19)$$

✓ GNN-RL Model

GNN-RL modeli, müşteri ve depo konumları arasındaki mekansal ilişkileri öğrenmek için GNN ve rota optimizasyonu için RL kullanılarak tasarlanmıştır. Model, GNN ile mekansal bağımlılıkları öğrenir ve bu bilgileri RL algoritmasına aktararak araç rotalarını optimize eder. Veri hazırlık aşamasında, müşteri ve depo konumları arasındaki zaman-mesafe matrisleri GPS verileri ve OSRM API kullanılarak hesaplanmıştır. Bu veriler, GNN için gerekli düğüm ve kenar özelliklerini oluşturmak için organize edilmiştir. GNN, müşteri ve depo arasındaki ilişkileri öğrenmek için kullanılmış ve her düğüm için 64 boyutlu bir özellik vektörü çıkarmıştır. RL ortamında, GNN tarafından öğrenilen bu özellikler, rota optimizasyonunda gözlem uzayı olarak kullanılmıştır. RL modeli, Stable-Baselines3 kütüphanesindeki PPO algoritması ile eğitilmiştir. Model, müşteri taleplerini karşılamak ve araç kullanımını optimize etmek amacıyla rota planlarını öğrenmiştir. Eğitim ve test süreçlerinde modelin performansı, toplam yolculuk süresi, kullanılan araç sayısı ve yük kullanım oranı gibi metrikler ile değerlendirilmiştir. Bu yapı, mekansal ilişkileri dikkate alarak etkili bir rota optimizasyon çözümü sunar (Şekil 67-68) (Erdem *et al.* under review d).

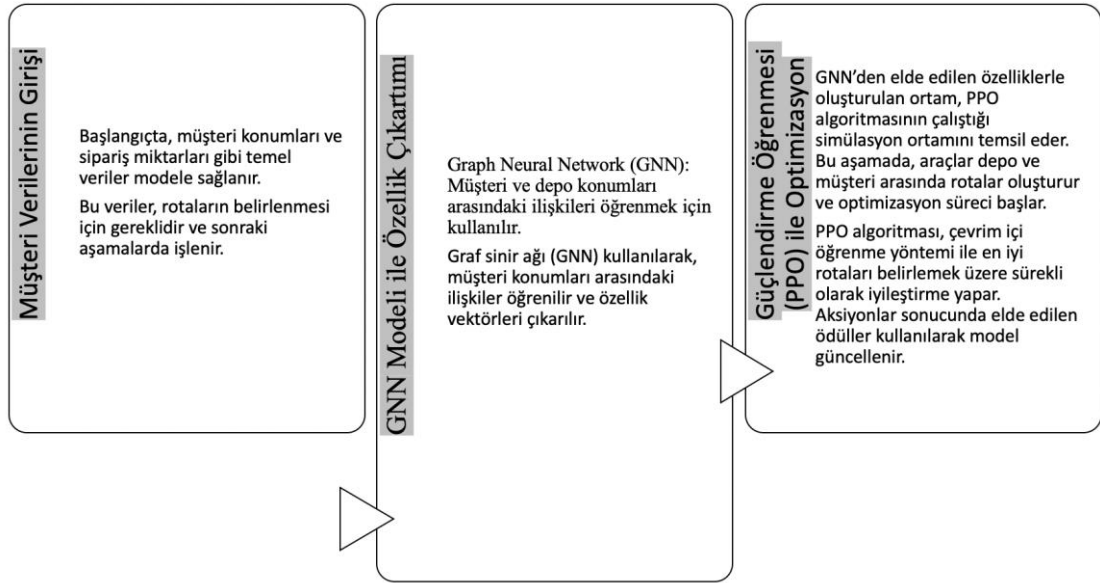
GNN yapısı ayrıntılı incelendiğinde;

- GNN Katmanları (GCNConv): İlk katman, düğümlere ait özellikleri (örneğin konum koordinatları gibi) daha anlamlı ve yüksek boyutlu temsillere dönüştürerek öğrenme süreci için uygun bir forma getirir. İkinci katman, bu özellikleri işleyerek problemle ilgili daha anlamlı temsiller oluşturur.
- Graf Veri Yapısı: Lokasyonlar düğümler olarak temsil edilir. İki lokasyon arasındaki mesafeler kenar özellikleri olarak eklenmiştir.
- Modelin Eğitimi: Eğitim verilerinde, müşteriler ve depolar arasındaki zaman-mesafe matrisi kullanılarak GNN modelinde düğüm temsilleri öğrenilmiştir. Eğitim sırasında ‘Adam’ optimizasyon algoritması ile ağırlıklar güncellenmiştir. Kaybı ölçmek için ‘mean’ kayıp fonksiyonu kullanılmıştır.
- Çıkış: Her müşteri ve deponun öğrenilmiş temsilleri (gömme vektörleri) elde edilir.

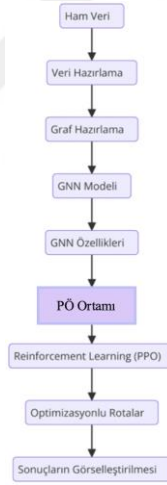
RL yapısı ayrıntılı incelendiğinde;

- Stable-Baselines3 kütüphanesinden PPO algoritması kullanılmıştır.
 - Durum: GNN tarafından üretilen düğüm temsilleri.
 - Eylemler (Actions): Araçların hangi müşteriye veya depoya hareket edeceği.
 - Ödül (Reward): Toplam seyahat süresinin azaltılmasını teşvik ederken, aynı noktaya yapılan tekrar ziyaretler için cezalandırma yapacak şekilde tasarlanmıştır.

- Modelin Eğitimi: RL modeli, GNN tarafından üretilen düğüm temsillerini giriş olarak almıştır. PPO algoritması, toplam ödülü maksimize etmek için politikayı iteratif olarak güncellemiştir.



Şekil 67. GNN-RL optimizasyon süreci



Şekil 68. Araç rota optimizasyonu için GNN-RL yaklaşımı

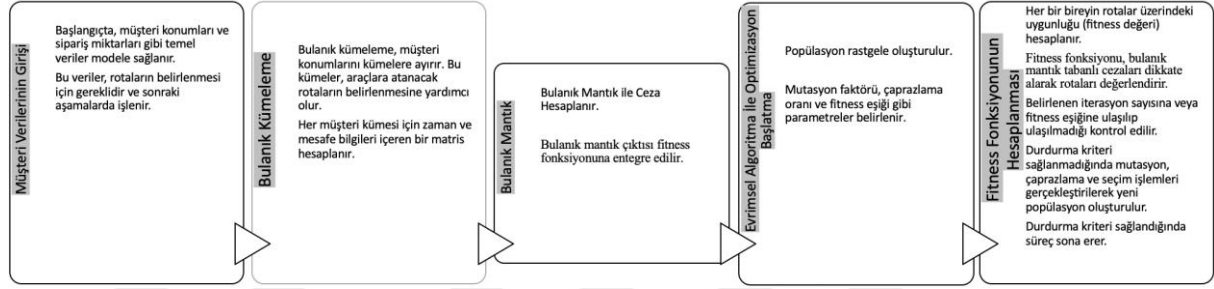
GNN-RL matematiksel modeli incelendiğinde, Denklem 17'deki gibi GNN'den çıkartılan düğüm temsilleri, RL ortamında durum uzayı olarak Denklem 20'deki gibi tanımlanır. Burada $T_{i,j}$ seçilen iki düğüm arasındaki seyahat süresi ve β ceza ağırlık katsayısıdır. Politika π_θ , PPO ile Denklem 21'deki gibi optimize edilir. Burada, $r_t(\theta)$ politika olasılık oranı, $A^{(t)}$, avantaj tahmini, ϵ politikadaki güncelleme oranını sınırlayan klipsleme eşikini ve E_t ise zaman adımı t 'deki beklenen değeri temsil eder.

$$r_t = -T_{i,j} - \beta \cdot \text{Ceza}(a_t) \quad (20)$$

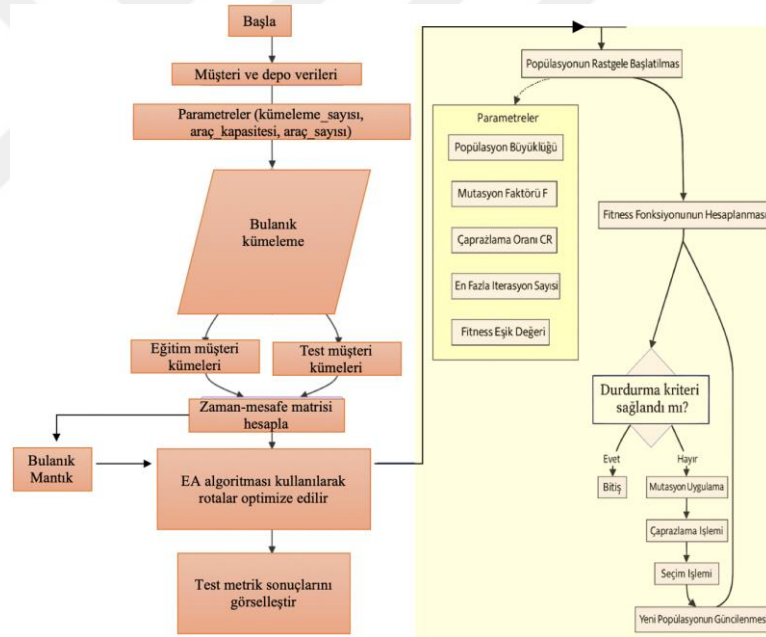
$$L^{\text{clip}}(\theta) = E_t[\min(r_t(\theta) A^{(t)}, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon) A^{(t)})] \quad (21)$$

✓ BME

Bulanık Mantık tabanlı ceza fonksiyonu ve Evrimsel Algoritma entegrasyonu ile ÇDARP çözümü sunar. Bulanık mantık, kapasite kullanımı ve zaman maliyetine dayalı ceza değerleri hesaplayarak evrimsel algoritmaya doğrudan etki eder. Bulanık Mantık ile hesaplanan ceza değerleri Evrimsel Algoritma tarafından optimize edilen rotaların toplam maliyetine dahil edilir. Bu sayede optimize edilen rotalar daha verimli ve gerçekçi olur (Erdem *et al.* under review d).



Şekil 69. BME optimizasyon süreci



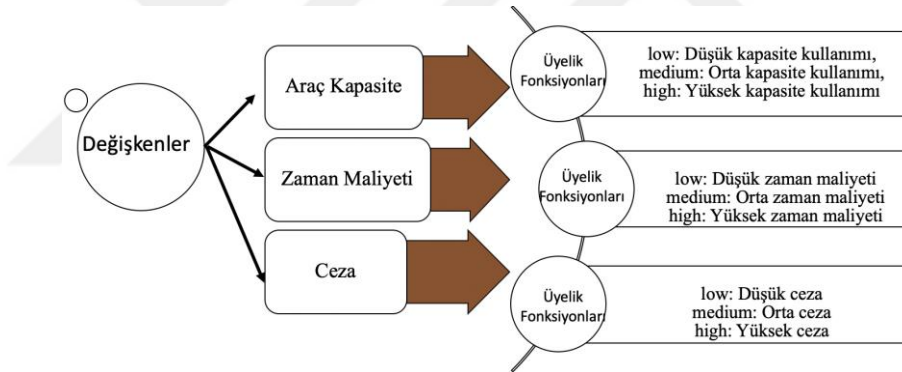
Şekil 70. Araç rota optimizasyonu için BME yaklaşımı

BMEO aşamaları adım adım incelendiğinde (Şekil 69-70);

- 1- Bulanık Kümeleme: Bulanık kümeleme, müşteri konumlarını kümeler ayırır. Bu kümeler, araçlara atanacak rotaların belirlenmesine yardımcı olur. Giriş Parametreleri: Müşteri konumları ve küme sayısıdır. Çıkış Parametreleri: Kümelere atanmış müşteri indeksleri ve her müşterinin küme üyelik derecesidir.
- 2- Bulanık Mantık ile Ceza Hesaplama: Araç kapasitesi, kullanım oranı ve zaman maliyetine göre ceza hesaplanır. Bulanık mantık, araç kapasitesi aşımı veya yüksek

zaman maliyeti durumlarını cezalandırır. Giriş Parametreleri: Araç kapasitesi kullanım oranı ve toplam zaman maliyetidir. Çıkış Parametresi: Ceza değeridir. Bulanık mantık yaklaşımıyla ceza hesaplamasının nasıl gerçekleştirildiğine dair işlemler ayrıntılı olarak aşağıda sırasıyla açıklanmıştır.

→Bulanık Değişkenler ve Üyelik Fonksiyonları: Üç adet bulanık değişken tanımlanmıştır (Şekil 71). Bunlardan ilki, araçların ne kadar kapasite kullandığını ifade eden "Araç Kapasite Kullanımı" değişkenidir. Bu değişken için üçgen üyelik fonksiyonları (trimf) kullanılarak "low" (düşük kapasite kullanımı), "medium" (orta kapasite kullanımı) ve "high" (yüksek kapasite kullanımı) üyelik seviyeleri tanımlanmıştır. İkinci bulanık değişken "Zaman Maliyeti" olup, belirli bir rotanın ne kadar süre veya maliyet gerektirdiğini ifade eder. Bu değişken için de "low" (düşük zaman maliyeti), "medium" (orta zaman maliyeti) ve "high" (yüksek zaman maliyeti) olmak üzere üç üyelik fonksiyonu belirlenmiştir. Üçüncü değişken ise "Ceza" olup, kapasite kullanımı ve zaman maliyetine bağlı olarak uygulanacak ceza miktarını temsil etmektedir. Bu değişkenin üyelik fonksiyonları da yine "low" (düşük ceza), "medium" (orta ceza) ve "high" (yüksek ceza) şeklinde tanımlanmıştır.



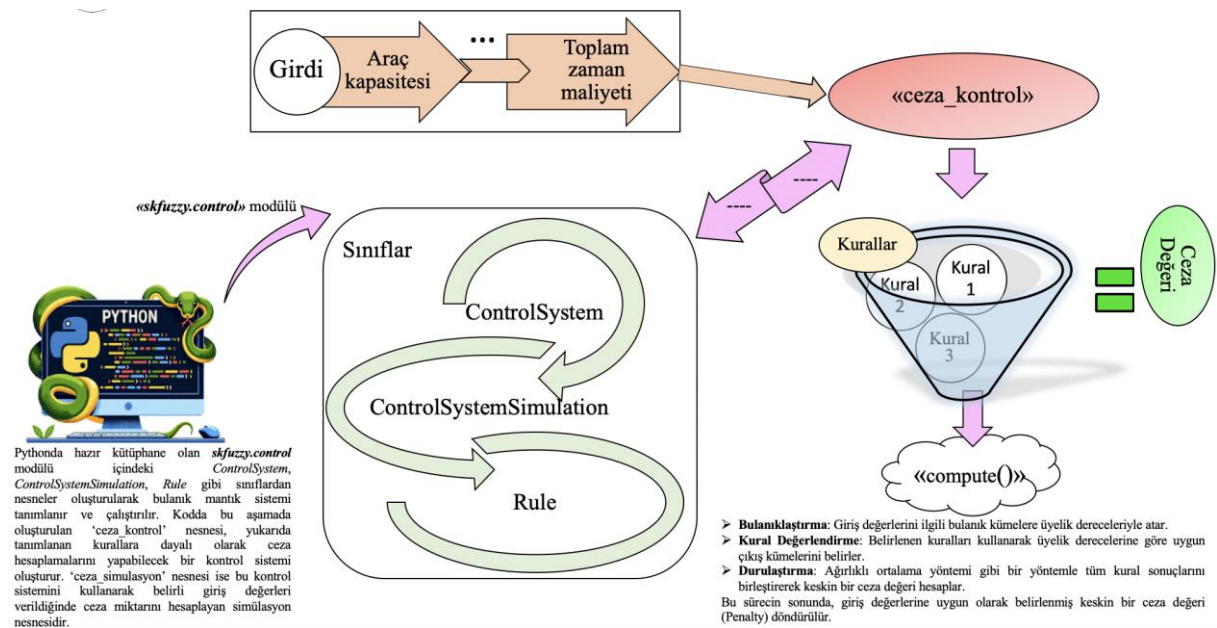
Şekil 71. Araç kapasitesi, zaman maliyeti ve ceza değişkenleri için üyelik fonksiyonlarının tanımlanması

→Bulanık Kurallar: Üç adet bulanık kural tanımlanmıştır. Birinci kurala göre, araç kapasite kullanımı düşük ve zaman maliyeti de düşük olduğunda, cezanın da düşük olması gerekmektedir. İkinci kurala göre, hem araç kapasite kullanımı hem de zaman maliyeti orta seviyedeysen, uygulanacak ceza da orta seviyede olmalıdır. Üçüncü ve son kurala göre ise, araç kapasite kullanımı yüksek ya da zaman maliyeti yüksek olduğunda, ceza da yüksek seviyede belirlenmelidir.

→Bulanık Mantık Kontrol Sistemi: Bulanık kurallar doğrultusunda, Python programlama dili içerisindeki hazır kütüphane olan “skfuzzy.control” modülündeki “ControlSystem, ControlSystemSimulation ve Rule” gibi sınıflar kullanılarak bir Bulanık Mantık Kontrol Sistemi oluşturulmuştur. Bu aşamada tanımlanan “ceza_kontrol” nesnesi,

yukarıda belirtilen kurallara dayalı olarak ceza hesaplamalarını gerçekleştirebilecek bir kontrol sistemi olarak yapılandırılmıştır. Bu sisteme bağlı olarak oluşturulan “ceza_simulasyon” nesnesi ise, belirli giriş değerleri verildiğinde ceza miktarını hesaplayan bir simülasyon bileşeni olarak görev yapmaktadır (Şekil 72).

→Ceza Hesaplama Fonksiyonu: Kapasite kullanımı ve zaman maliyeti verildiğinde ceza değerini hesaplayan bir fonksiyon tanımlanmıştır. Fonksiyonun çalışma mantığına göre, önce araç kapasitesi ve maliyet değerleri belirlenir. Bu değerler, daha önce tanımlanmış olan “ceza_simulasyon” nesnesine giriş olarak aktarılır. Ardından, bulanık kurallar doğrultusunda “compute()” fonksiyonu kullanılarak ceza değeri hesaplanır ve bu değer fonksiyonun çıktısı olarak döndürülür.



Şekil 72. Araç kapasitesi ve zaman maliyetine dayalı ceza değeri hesaplama süreci

3- Fitness Fonksiyonu: Toplam zaman maliyeti ve ceza değerinin birleşimiyle toplam maliyet hesaplanır. Bulanık Mantık çıktısı burada fitness fonksiyonuna entegre edilmiştir. Giriş Parametreleri: Rota, zaman-mesafe matrisi, araç kapasitesi, müşteri talepleri ve araç kimliği, Bulanık Mantık cezasıdır. Çıkış Parametresi: Toplam maliyettir.

4- Evrimsel Algoritma ile Optimizasyon: “Differential Evolution” algoritması kullanılarak rotalar optimize edilmiştir. Fitness fonksiyonu, Bulanık Mantık tabanlı cezaları dikkate alarak rotaları değerlendirmiştir. Giriş Parametreleri: Zaman-mesafe matrisi, müşteri kümeleri, araç kapasiteleri ve müşteri talepleridir. Çıkış Parametreleri: Optimize edilmiş rotalar ve fitness değeridir.

Evrimsel Algoritma, popülasyon tabanlı bir evrimsel optimizasyon algoritması olup, karmaşık ve doğrusal olmayan sürekli optimizasyon problemlerini çözmek için kullanılır. Bu çalışmada, Scipy kütüphanesinden alınan bir Differential Evolution algoritması kullanılmıştır. Differential Evolution Algoritmasının aşamaları ayrıntılı olarak aşağıda sırasıyla açıklanmıştır.

→Popülasyonun Başlatılması: Algoritma, çözümlerden oluşan bir başlangıç popülasyonu ile başlar. Her çözüm (vektör), belirli sınırlar (bounds) dahilinde rastgele olarak oluşturulur. Bu sınırlar bounds parametresi ile belirlenmiştir. Her müşteri ve depo konumunu temsil eden noktalar için sıralı indeks sınırları tanımlanmıştır.

→Mutasyon: Her bir çözüm (hedef vektör) için, popülasyondaki farklı çözüm seçilir ve bu çözümler kullanılarak mutasyon vektörü oluşturulur.

→Çaprazlama: Hedef vektör ve mutasyon vektörü arasında çaprazlama yapılır. Bu işlem sonucunda yeni bir aday çözüm elde edilir. Yeni aday çözüm ile mevcut çözüm karşılaştırılır. Daha iyi olan çözüm bir sonraki nesile aktarılır.

→Durdurma Kriteri: Algoritma, belirli bir iterasyon sayısına ulaşıldığında veya iyileşme belirli bir tolerans seviyesinin altına düştüğünde durdurulur.

BMEO matematiksel modeli incelendiğinde, kullanılan ceza tabanlı uygunluk fonksiyonu, doğrudan seyahat maliyetlerini ve bulanık ceza değerlerini birleştirir. Bu fonksiyon, toplam seyahat süresine (T) ek olarak kapasite ve zaman kısıtlarına dayalı olarak hesaplanan bulanık ceza (P) terimini içerir (Denklem 22). Burada $T(i,j)$, i ve j konumları arasındaki seyahat süresini; P ise araç kapasitesi ve zaman kısıtlarına dayalı olarak hesaplanan bulanık ceza değerini temsil eder.

$$F(R)=\sum_{i=1}^V \sum_{j=1}^N T(i,j) + P \quad (22)$$

Evrimsel algoritmada mutasyon güncelleme kuralı Denklem 23'deki gibi tanımlanır. Burada X_{best} popülasyondaki en iyi çözümü, X_{rand1} ve X_{rand2} ise rastgele seçilmiş iki çözümü temsil eder. F , mutasyon faktörüdür ve çözüm arama alanındaki çeşitliliği artırmak için kullanılır.

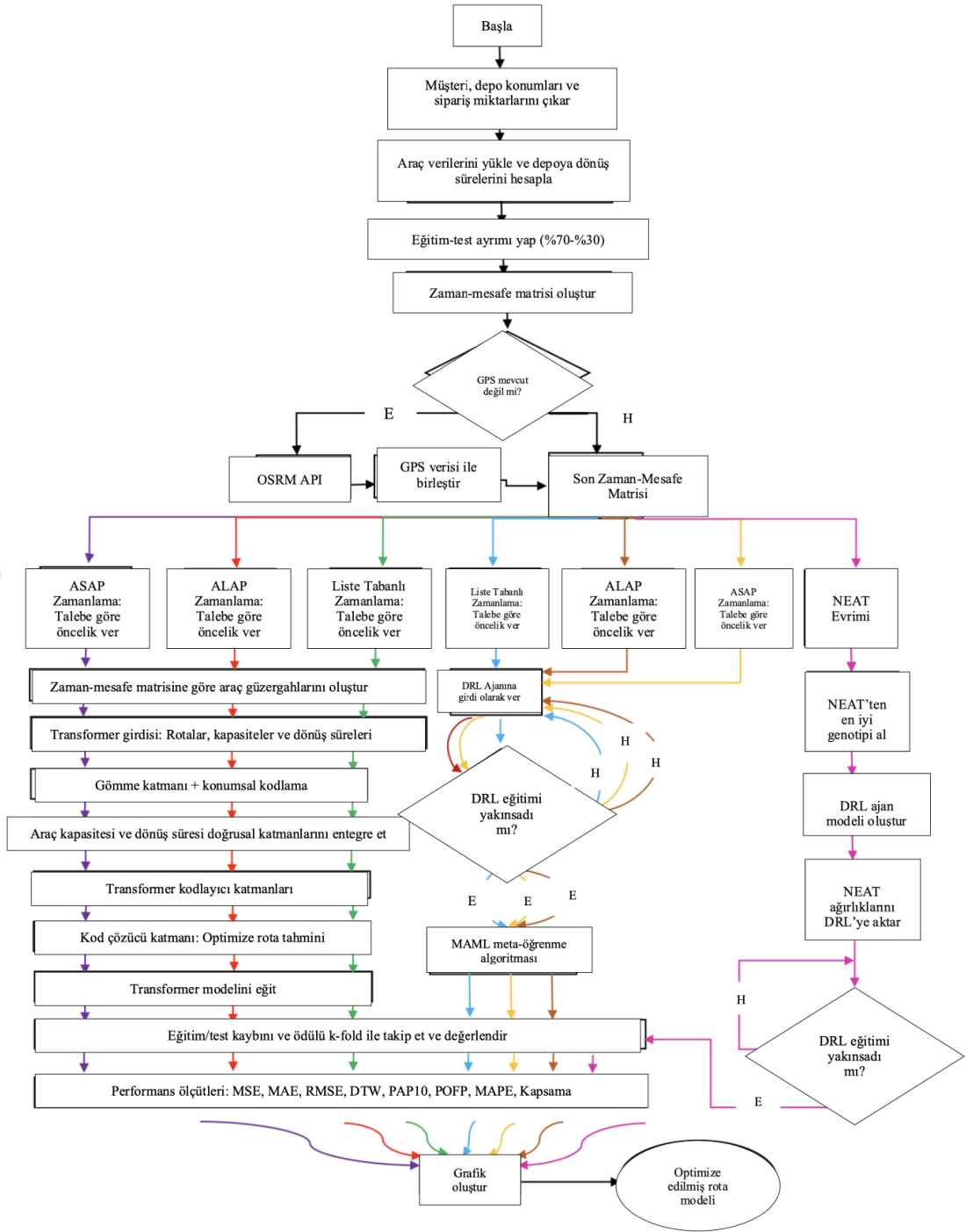
$$X_{mut}= X_{best} + F \cdot (X_{rand1} - X_{rand2}) \quad (23)$$

Çaprazlama işlemi, yeni bireylerin (çözümlerin) oluşturulmasında kullanılır ve Denklem 24'deki gibi tanımlanır. Burada C_r , çaprazlama olasılığı; r_i , $[0,1]$ aralığında rastgele üretilmiş bir sayıdır. Eğer r_i , C_r 'den küçükse mutasyon sonucu alınır, aksi takdirde hedef çözüm korunur.

$$X_{trial,i} = \begin{cases} X_{mut,i} & \text{if } r_i \leq C_r \\ X_{target,i} & \text{diğer durumlar} \end{cases} \quad (24)$$

HLS tabanlı, derinlemesine öğrenme ve evrimsel modellerin bütünleşik uygulaması

Yukarıda detaylı biçimde açıklanan klasik ve hibrit algoritmaların deneysel başarıları, daha ölçeklenebilir, genellenebilir ve donanıma entegre edilebilir çözümler geliştirilmesi gerekliliğini ortaya koymuştur. Bu doğrultuda, çalışmanın ileri aşamasında, çoklu depo, çoklu tur ve bölünmüş teslimat özelliklerine sahip gerçek-dünya ÇDARP senaryolarına uygulanabilecek yedi yenilikçi AI tabanlı model önerilmiştir. Bu modeller; klasik sıralama sezgiselleri (ASAP, ALAP, LST) ile Transformer ağları, DRL, MAML ve NEAT gibi ileri düzey yöntemlerin birleşiminden oluşmaktadır. Her bir model, müşteri taleplerinin değişkenliği, araç kapasitesi ve depo dönüş süreleri gibi operasyonel sınırlamaları dikkate alarak optimize edilmiş rotalar üretmeyi amaçlamıştır. Modellerin genel performansını değerlendirmek ve genellenebilirliğini test etmek amacıyla 5 katlı çapraz doğrulama uygulanmıştır. Model çıktıları; MSE, MAE, RMSE, Dinamik Zaman Savaşımı (DTW (Dynamic Time Warping)), MAPE, %10 içindeki Doğru Tahmin Oranı (PAP10), Aşırı Karşılana Tahmin Oranı (POFP (Proportion of Over-Forecasted Predictions)) ve Kapsama Skoru (Coverage Score) gibi hata ve operasyonel metriklerle değerlendirilmiştir (Şekil 73). Eğitim süreçlerinin analizi, modeller arası karşılaştırmalar ve farklı teslimat senaryolarındaki rota performanslarının yorumlanması amacıyla çubuk grafikler, logaritmik ölçekli grafikler, radar diyagramları ve histogramlar gibi görsel analiz araçları kullanılmıştır. Öte yandan, yalnızca tahmin başarıımı değil, her modelin hesaplama maliyeti ve pratik uygulanabilirliği de dikkate alınmıştır. Bu değerlendirme; mimari karmaşıklık (parametre sayısı), eğitim süresi ve farklı kentsel senaryolarda genellenebilirlik yeteneği gibi ölçütlere dayalı olarak yapılmıştır. Ayrıca, teslimat zaman aralığı kısıtları ve depo uygunluğu gibi lojistik faktörler kodlanmış, modellerin gerçek zamanlı operasyonel değişkenlere uyum sağlayabilmesi sağlanmıştır.



Şekil 73. ÇDARP optimizasyonu için hibrit AI tabanlı modellerin akış şeması (Erdem *et al.* under review c)

✓ AST-Former

Bu model, öncelikli müşteri taleplerini karşılamayı hedefleyen ASAP zamanlama stratejisini temel alır. Müşteri siparişleri, talep miktarına göre azalan şekilde sıralanır ve erken teslimat önceliği vurgulanır. Önceliklendirilmiş liste, optimize edilmiş teslimat rotaları üretmek amacıyla Transformer tabanlı bir encoder-decoder mimarisi ile işlenir. Mimari, zamansal ve mekânsal bağımlılıkları modellemek için çoklu başlık dikkat ve encoder-decoder katmanları kullanır. Transformer girişi; ASAP sıralı müşteri konumları, araç kapasite vektörleri ve depo

dönüş sürelerinin gömme, konumsal kodlama ve doğrusal dönüşümlerle kodlanmasının ardından birleştirilmesiyle elde edilen tensörden oluşur. Modelin çekirdeğinde, her biri sekiz dikkat başlığına sahip altı katmanlı bir encoder yer alır. Ardından gelen decoder, müşteri ziyaret sırasını tahmin eder. AST-Former, teslimat maliyetini minimize ederken araç kapasitesi ve teslimat önceliklerine uygun optimize müşteri sıralamaları üretir (Erdem *et al.* under review c).

✓ ALT-Former

ALT-Former, zaman kısıtlarının daha esnek olduğu senaryolarda teslimatları erteleterek optimizasyon sağlayan ALAP sezgisini kullanır. Müşteriler, talep miktarına göre artan sırada sıralanır ve bu sıra Transformer tabanlı encoder-decoder mimarisine giriş olarak verilir. Bu strateji, geç teslimatların ceza oluşturmadığı durumlarda lojistik süreçlerin zaman açısından daha esnek yönetilmesini sağlar. ALAP, teslimatların ertelenmesini vurgulayarak araç yük dağılımını iyileştirir ve zamansal kaynak kullanımını dengeler. Sıralanmış teslimat verileri, araç kapasite bilgileri ve depo dönüş süreleri gömme ve doğrusal dönüşüm katmanlarıyla kodlandıktan sonra altı katmanlı ve sekiz dikkat başlıklı encoder tarafından işlenir. Ardından gelen decoder, optimize edilmiş müşteri sıralamasını üretir. ALAP stratejisini dikkat temelli yapay öğrenme yapısına entegre eden ALT-Former, zaman açısından esnek dağıtım senaryolarında verimliliği artırır (Erdem *et al.* under review c).

✓ LST-Former

LST-Former, araç yüklerini dengelemeye odaklanan LST algoritmasını kullanır ve müşteri taleplerini azalan sıraya göre düzenleyerek araç kapasitesinin maksimum düzeyde kullanılmasını sağlar. Talebe göre önceliklendirme ve araç atama işlemleri tamamlandıktan sonra, Transformer tabanlı encoder-decoder mimarisi kullanılarak optimize edilmiş teslimat sıralamaları üretilir. Listeleme aşaması, yüksek talepli siparişlerin mevcut araç kapasitesine en uygun şekilde eşleştirilmesini ve mekânsal-zamansal kısıtların dikkate alınmasını sağlar. Transformer mimarisi, gömme ve konumsal kodlama teknikleriyle teslimat sırasındaki desenleri öğrenir. Encoder katmanları müşteri arası ilişkileri çıkarırken, decoder en verimli rotalama sırasını oluşturur. LST-Former, yüksek hacimli ve dinamik teslimat ortamlarında sağlam bir performans sergileyerek araç kullanımını optimize eder ve lojistik maliyetleri azaltır (Erdem *et al.* under review c).

✓ LST-MetaDRL

LST-MetaDRL, klasik LST algoritmasını DRL ajanı ile birleştirir ve MAML mekanizmasını içerir. Müşteri talepleri talep büyüklüğü ve mekânsal yakınlığa göre sezgisel olarak araçlara atanır. Ardından, DRL ajanı ödül sinyalleri yardımıyla rotalama stratejilerini

aşamalı olarak iyileştirir; bu sinyaller, fazla seyahat süresi ve kapasite ihlalleri gibi verimsizlikleri cezalandırır. MAML bileşeni, modelin yeni teslimat koşullarına hızlıca uyum sağlayabilmesini mümkün kılarak genelleme yeteneğini artırır. Bu yapı, deterministik zamanlama ile adaptif öğrenmeyi bir araya getirerek sağlam ve ölçeklenebilir bir çözüm sunar (Erdem *et al.* under review c).

✓ ALT-MetaDRL

ALT-MetaDRL modeli, teslimat esnekliğini artırmak üzere ALAP zamanlama sezgisini kullanır. Talep sırasına göre artan şekilde sıralanan siparişler, daha dengeli bir araç yükleme dağılımı sağlar. Bu sıralamalar, DRL ajanı için giriş olarak kullanılır ve ajan, zaman, yük dengesi ve seyahat maliyeti açısından verimsizlikleri cezalandıran bir ödül fonksiyonu aracılığıyla politikaları iteratif olarak iyileştirir. MAML, farklı teslimat bölgeleri veya depo konfigürasyonları için modelin hızlıca uyum sağlamasını mümkün kılar. ALT-MetaDRL, zamanla değişen kısıtlar ve ertelenebilir teslimatların yoğun olduğu senaryolarda esnek ve akıllı bir çözüm sunar (Erdem *et al.* under review c).

✓ AST-MetaDRL

AST-MetaDRL, ASAP zamanlamasını DRL ve MAML ile birleştirerek hızlı ve uyarlanabilir bir teslimat çözümü sunar. Yüksek öncelikli teslimatları önceleyen sıralamalar, DRL ajanı tarafından işlenerek toplam teslimat maliyetini en aza indiren politikalar öğrenilir. MAML bileşeni, ani değişen müşteri talepleri veya depo durumlarına hızlı yanıt verebilme yeteneği sağlar. Meta seviye başlatma (Meta-initialization) yaklaşımı sayesinde model minimum güncelleme ile farklı koşullara adapte olabilir. Bu hibrit mimari, belirsizlik altında karar verme gerektiren gerçek zamanlı lojistik uygulamaları için uygundur (Erdem *et al.* under review c).

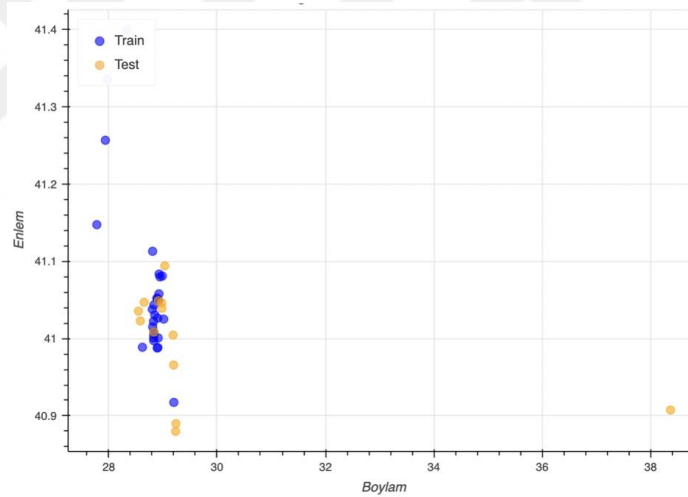
✓ NRL

NRL modeli, DRL ajanının sinir ağı yapısını hem ağırlıklar hem de mimari düzeyinde geliştirmek için NEAT algoritmasını entegre eder. Evrimsel süreç ARP'nin karmaşıklığına özel en uygun politika mimarisini arar. En uygun aday çözüm seçildikten sonra, bu yapı DRL ajanının başlangıç ağırlıklarını belirler ve öğrenme süreci hızlandırılır. Evrim ve RL'den oluşan bu iki aşamalı yapı, dinamik teslimat ortamlarında uyarlanabilirlik ve ölçeklenebilirlik sağlar. Politika yapısının veri karmaşıklığına göre evrilmesine olanak tanıyan NRL, heterojen filo ve değişken talep senaryolarına uygun esnek bir rotalama çözümü sunar (Erdem *et al.* under review c).

ARAŞTIRMA BULGULARI

Tezin tüm aşamalarında, algoritmaların potansiyeli, Intel® Core™ i7-7700 işlemci (~3,6 GHz), 24 GB bellek, Intel® HD Graphics 630 ve NVIDIA GeForce GT 730 grafik kartına sahip üç bilgisayar ile Intel® Xeon® Gold 5218 işlemciye (64 çekirdek, 2,3 GHz) ve 131072 MB belleğe sahip bilgisayarda gerçekleştirilmiştir. Deneyler, Python programlama dili kullanılarak, Scikit-learn, TensorFlow, Keras, Matplotlib ve Bokeh vb. kütüphaneler ile yapılmıştır.

Tezin ilk üç aşamasında, DL modelleri için Keras, ML modelleri için ise sklearn kütüphanesi kullanılmıştır. Eğitim ve test veri oranları %80-%20 olarak belirlenmiştir (Şekil 74). Bu yaklaşım, veri bütüneşik yapısını kullanarak daha etkili rota optimizasyonu sağlamayı hedeflemiştir.



Şekil 74. Eğitim ve test noktalarının görsel dağılımı

Tezin ilk iki aşamasında, Modellerin performansı, MAE, MSE, R^2 ve RMSE metrikleri kullanılarak değerlendirilmiştir. Adam optimizasyon algoritması kullanılmıştır. DL tabanlı modeller için öğrenme oranı 0,001, epoch sayısı 100 ve batch boyutu 150 olarak belirlenmiştir.

Tablo 27, 28 ve Şekil 75’de elde edilen sonuçlar incelendiğinde, kNN-DTR modeli, CHD veri setinde 0,9446 R^2 değeriyle en iyi sonuçları elde etmiştir. WnvD veri setinde ise Yığın Tabanlı Topluluk Modeli, 0,9999 R^2 değeriyle iyi sonuç vermiştir. Yığın tabanlı topluluk modeli, WnvD veri setinde sıfır hata oranıyla (MAE = 0, MSE = 0, RMSE = 0) tüm metriklerde en iyi performansı göstermiştir. CHD veri setinde kNN-DTR modeli, en düşük RMSE (0,4699) değeriyle öne çıkmıştır. Negatif R^2 değerleri, bazı modellerin veri setine uygun olmadığını

göstermiştir. CNN1 ve CNN2 modelleri negatif R^2 değerleriyle düşük performans sergilemiştir. CNN1-SVR ve CNN2-SVR gibi hibrit modeller, tekil modellerden (CNN1 ve CNN2) daha iyi sonuçlar vermiştir. AlexNet modeli, regresyon hattından en uzak tahminleri yaparken, AlexNet-SVR hibrit modeli bu mesafeyi azaltarak performansı iyileştirmiştir. Topluluk modelleri, en düşük hata oranlarını sağlayarak DL modellerine göre üstün performans göstermiştir. Çalışma genelinde, yığma yöntemi oylama yöntemine göre daha üstün sonuçlar vermiştir. Özellikle yığın tabanlı topluluk modeli, tüm test metriklerinde en yüksek doğruluğu sağlamıştır. Oylama yöntemleri, daha az karmaşık olduğu için hızlı bir çözüm olarak kullanılırken, yığın yöntemi daha yüksek doğruluk ve genelleme kabiliyeti sunmuştur. Daha karmaşık modeller her zaman daha iyi performans göstermemiştir. Örneğin, AlexNet modeli daha fazla parametre içermesine rağmen VGGNet'e göre daha kötü sonuçlar vermiştir. Bu durum, modelin karmaşıklığının artmasıyla daha fazla veri ve özellik gereksinimi doğurduğunu göstermektedir. Şekil 75-76'da verilen gerçek boylam değerleri ve tahmin edilen boylam değerleri incelendiğinde, hibrit ve topluluk modelleri (Yığın tabanlı topluluk modeli, kNN-DTR) genellikle tekil modellerden (CNN, SVR, AlexNet) daha üstün performans göstermiştir. Çalışma, GPS verilerindeki konum doğruluğunu artırmada topluluk ve hibrit modellerin etkili olduğunu göstermiştir. Bu yöntemler, navigasyon sistemlerinin hassasiyetini artırmada üstün bir yaklaşım sunmaktadır (Şekil 77-84) (Aydin & Erdem, 2023).

Tablo 27. Karşılaştırmalı MAE ve MSE Değerleri: CHD ve WnvD Veri Setleri Üzerinde Uygulanan Yöntemlerin Performans Analizi (Aydin & Erdem, 2023)

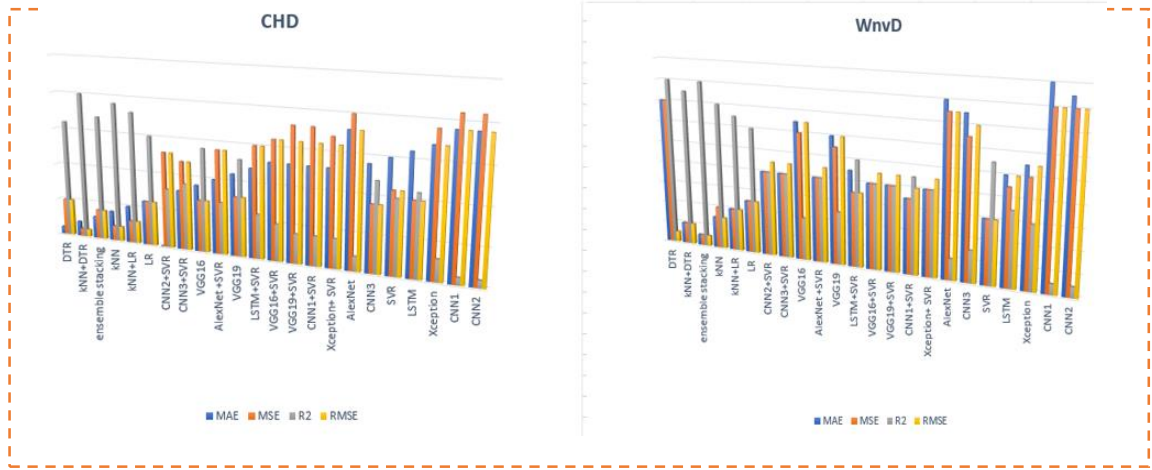
Veri Seti	Metod	MAE	MSE	Veri Seti	Metod	MAE	MSE
CHD	kNN-LR	0,3734	0,2842	WnvD	kNN-LR	0,0308	0,0014
CHD	kNN-DTR	0,2755	0,2208	WnvD	kNN-DTR	0,0047	0,0001
CHD	Yığın Tabanlı Topluluk Modeli	0,2893	0,3150	WnvD	Yığın Tabanlı Topluluk Modeli	0	0
CHD	CNN1-SVR	1,7440	5,0164	WnvD	CNN1-SVR	0,0782	0,0094
CHD	CNN2-SVR	1,6277	4,4649	WnvD	CNN2-SVR	0,0783	0,0095
CHD	CNN3-SVR	1,6452	4,2695	WnvD	CNN3-SVR	0,0783	0,0095
CHD	VGG16-SVR	1,7440	5,0162	WnvD	VGG16-SVR	0,0783	0,0095
CHD	VGG19-SVR	1,7440	5,0164	WnvD	VGG19-SVR	0,0783	0,0095
CHD	Xception-SVR	1,7440	5,0163	WnvD	Xception-SVR	0,0783	0,0095
CHD	AlexNet-SVR	1,6976	4,8529	WnvD	AlexNet-SVR	0,0783	0,0095

Tablo 27. (devamı)

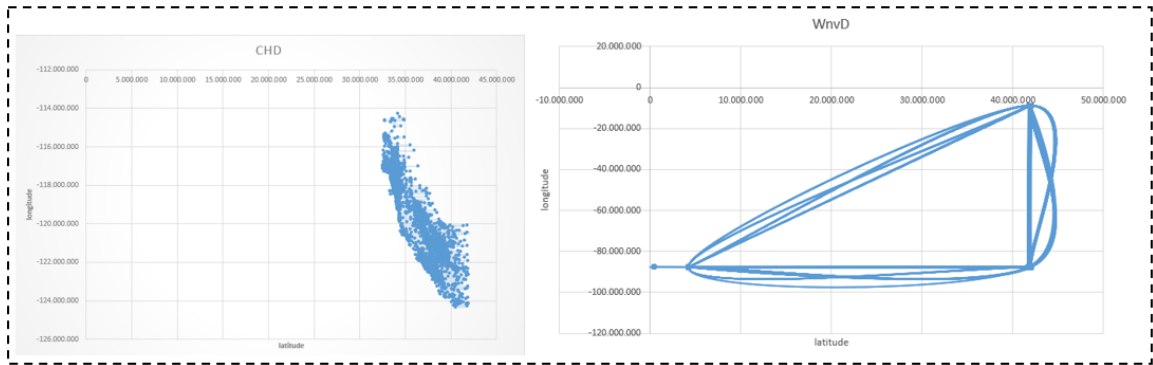
CHD	LSTM-SVR	1,7193	4,8616	WnvD	LSTM-SVR	0,0789	0,0094
CHD	CNN3	1,7602	3,9736	WnvD	CNN3	2,7944	7,8778
CHD	SVR	1,7890	4,0369	WnvD	SVR	0,0659	0,0056
CHD	VGG16	1,6494	3,7653	WnvD	VGG16	1,1617	1,3603
CHD	VGG19	1,6983	3,9016	WnvD	VGG19	0,7678	0,5990
CHD	Xception	2,0527	5,8520	WnvD	Xception	0,5328	0,3300
CHD	LSTM	1,8217	3,9942	WnvD	LSTM	0,1026	0,0191
CHD	kNN	0,3377	0,2821	WnvD	kNN	0,0094	0,0005
CHD	LR	0,4707	0,3983	WnvD	LR	0,0537	0,0044
CHD	DTR	0,2708	0,3198	WnvD	DTR	2,2826	8,0930

Tablo 28. Karşılaştırmalı R^2 ve RMSE Değerleri: CHD ve WnvD Veri Setleri Üzerinde Uygulanan Yöntemlerin Performans Analizi (Aydın & Erdem, 2023)

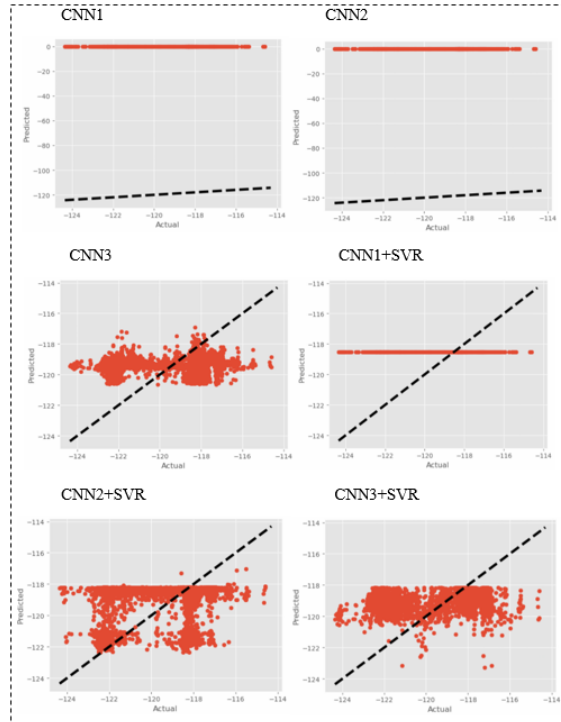
Veri Seti	Metod	R^2	RMSE	Veri Seti	Metod	R^2	RMSE
CHD	kNN-LR	0,9287	0,5331	WnvD	kNN-LR	0,8425	0,0386
CHD	kNN-DTR	0,9446	0,4699	WnvD	kNN-DTR	0,9864	0,0113
CHD	Yığın Tabanlı Topluluk Modeli	0,9210	0,5612	WnvD	Yığın Tabanlı Topluluk Modeli	0,9999	0
CHD	CNN1-SVR	-0,2580	2,2397	WnvD	CNN1-SVR	0,0006	0,0974
CHD	CNN2-SVR	-0,1197	2,1130	WnvD	CNN2-SVR	-0,0009	0,0975
CHD	CNN3-SVR	-0,0707	2,0662	WnvD	CNN3-SVR	-0,0009	0,0975
CHD	VGG16-SVR	-0,2579	2,2397	WnvD	VGG16-SVR	-0,0009	0,0975
CHD	VGG19-SVR	-0,2580	2,2397	WnvD	VGG19-SVR	-0,0009	0,0975
CHD	Xception-SVR	-0,2580	2,2397	WnvD	Xception-SVR	-0,0009	0,0975
CHD	AlexNet-SVR	-0,2170	2,2029	WnvD	AlexNet-SVR	-0,0009	0,0975
CHD	LSTM-SVR	-0,2192	2,2049	WnvD	LSTM-SVR	0,0087	0,0970
CHD	LR	0,9001	0,6311	WnvD	LR	0,5288	0,0669
CHD	DTR	0,9197	0,5655	WnvD	DTR	0,9999	0



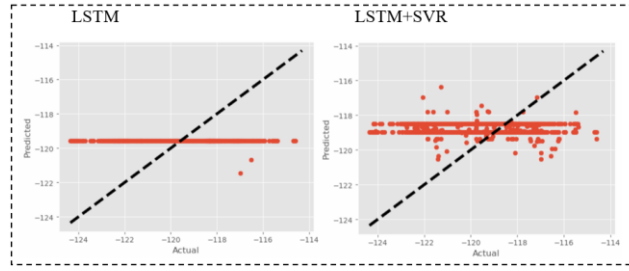
Şekil 75. Model performanslarının grafiksel özeti (Aydin & Erdem, 2023)



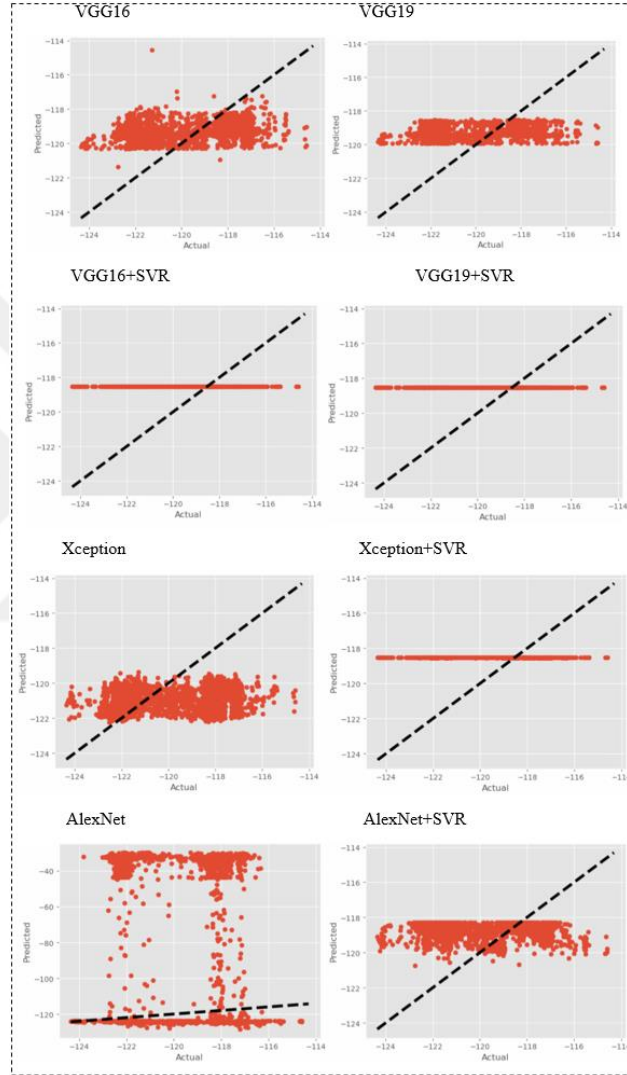
Şekil 76. Veri setlerindeki gerçek boylam değerlerinin (y) grafiksel dağılımları (Aydin & Erdem, 2023)



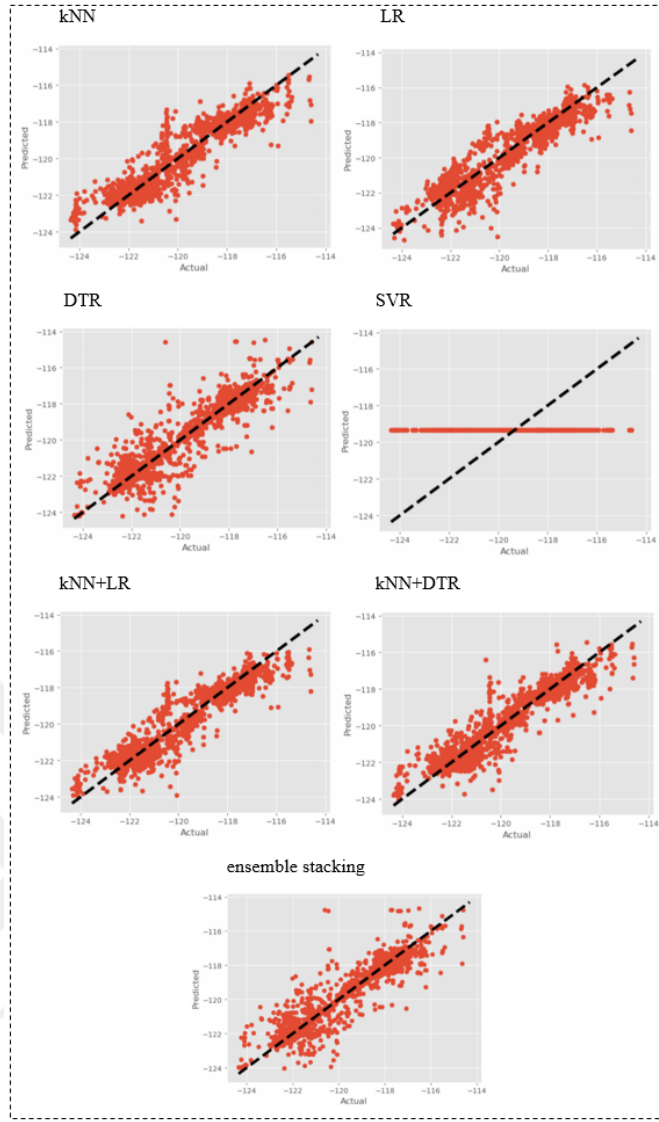
Şekil 77. CHD verisetinde CNN ve GPSCNN modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydin & Erdem, 2023)



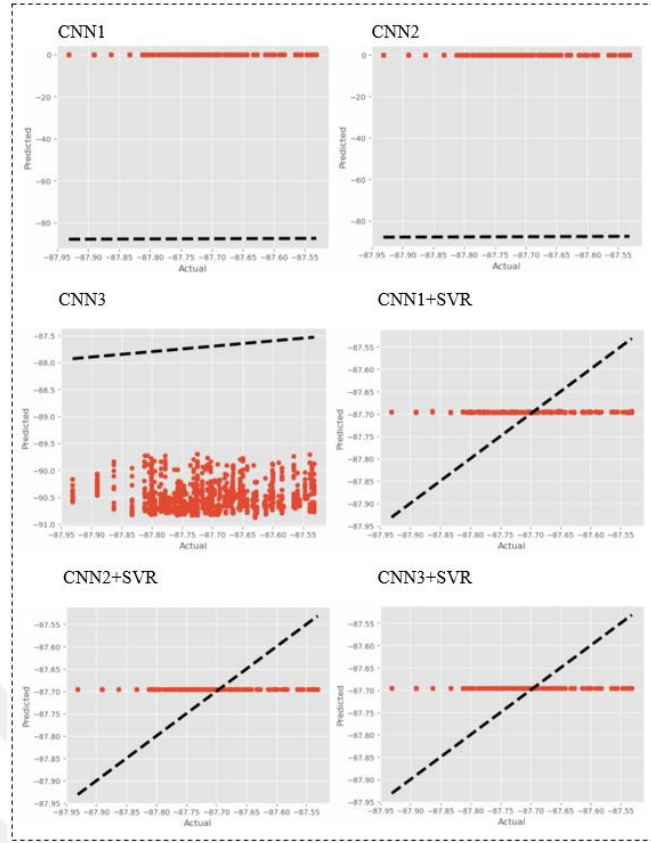
Şekil 78. CHD verisetinde GPSLSTM modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)



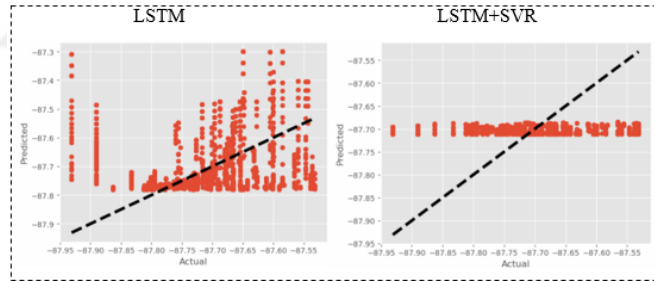
Şekil 79. CHD verisetinde GPSCNN ve ön-eğitilmiş modellerle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)



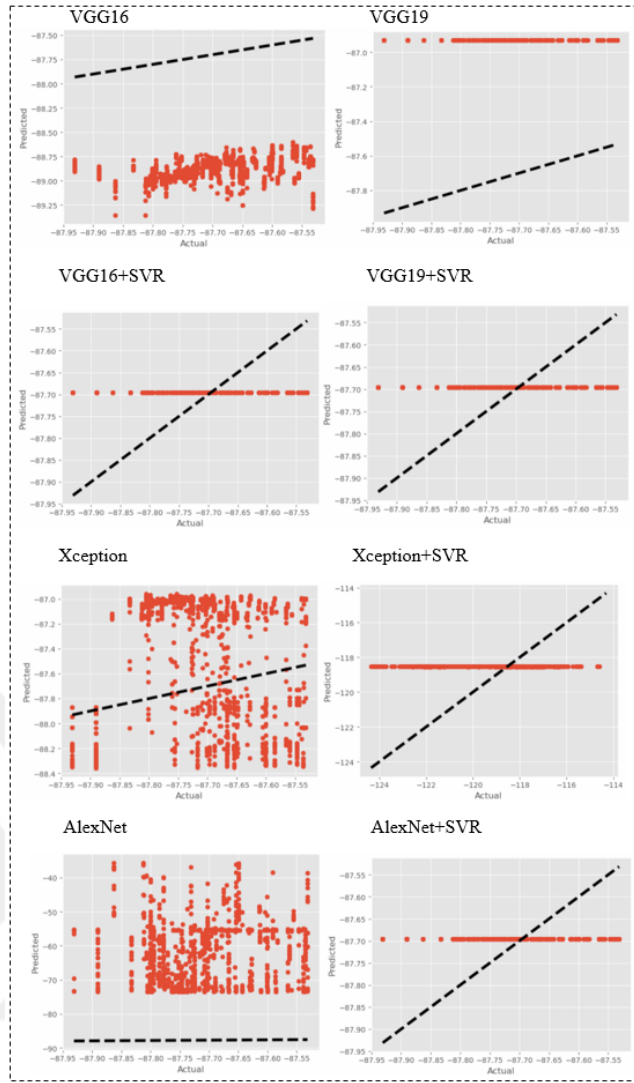
Şekil 80. CHD verisetinde ML ve topluluk modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)



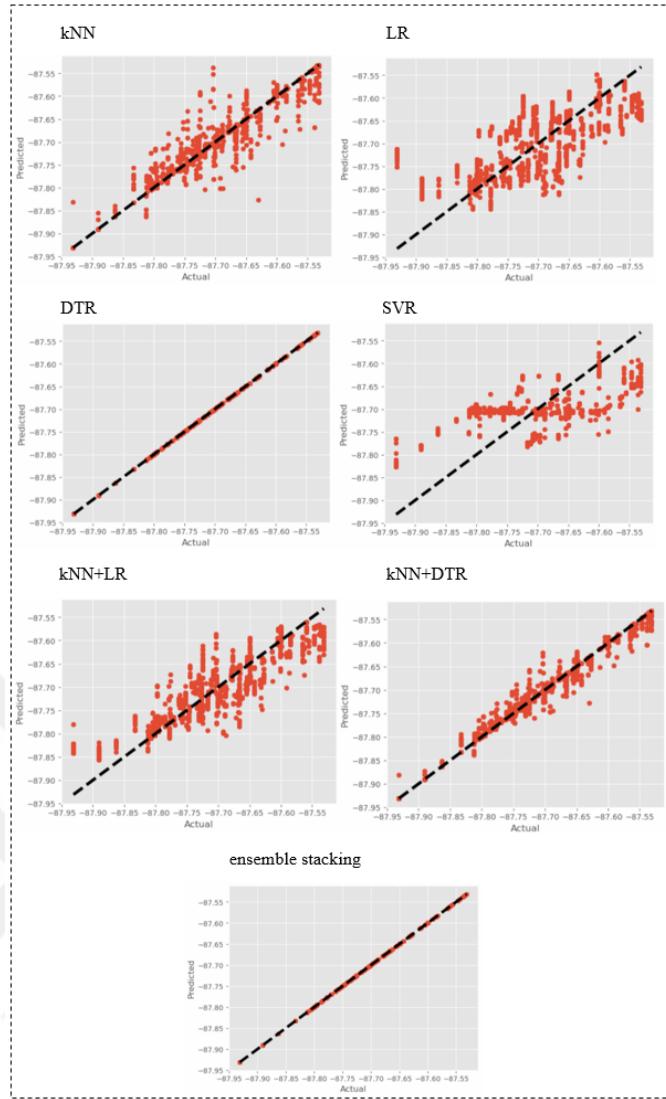
Şekil 81. WnvD verisetinde CNN ve GPSCNN modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)



Şekil 82. WnvD verisetinde GPSLSTM modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)



Şekil 83. WnvD verisetinde GPSCNN ve ön-eğitilmiş modellerle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)



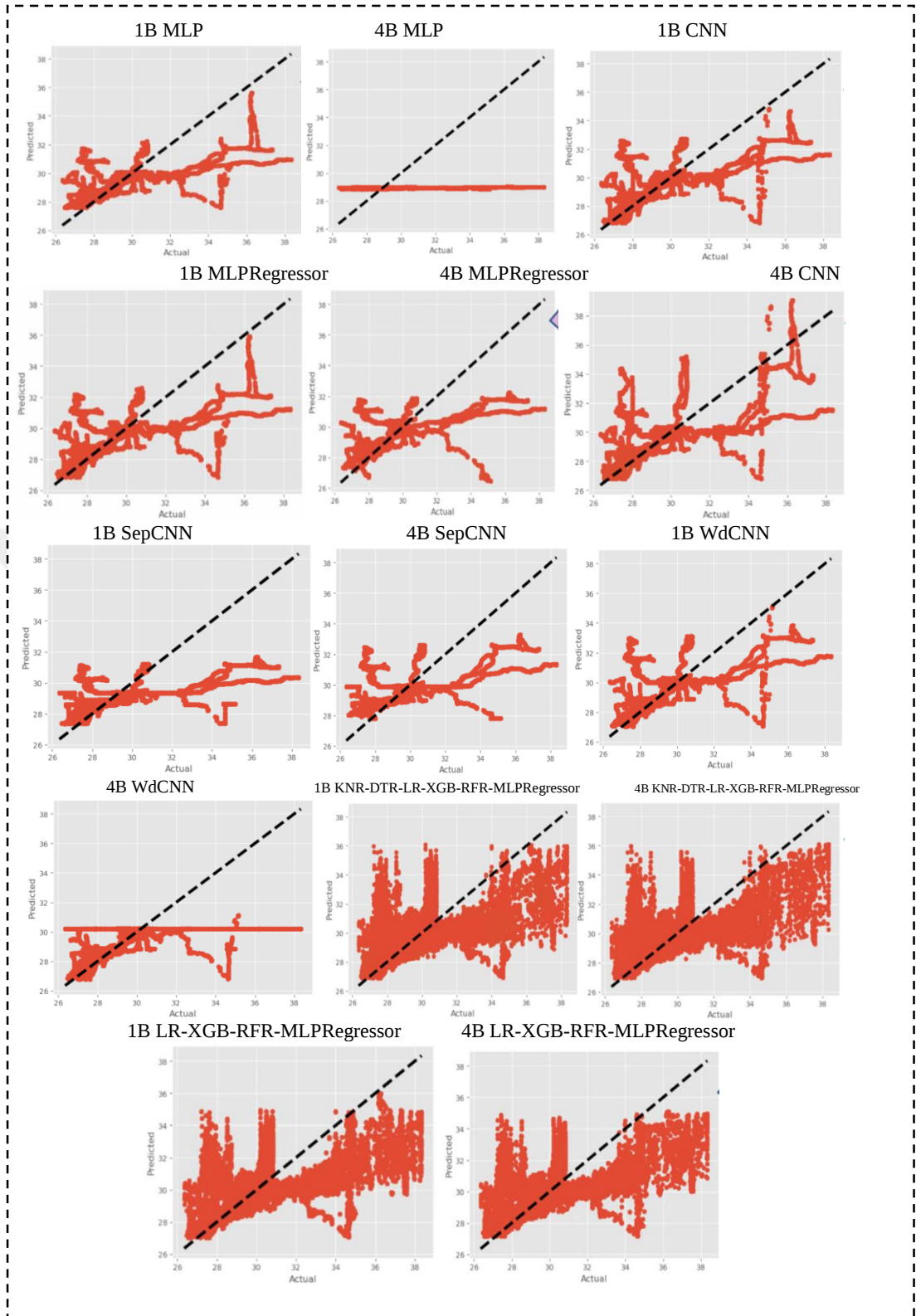
Şekil 84. WnvD verisetinde ML ve topluluk modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Aydın & Erdem, 2023)

Tezin ikinci aşamasında, çalışmada toplam 16 farklı model test edilmiştir. Bu modeller arasında ML tabanlı algoritmalar (LR, XGB, RFR, KNR, DTR, MLPRegressor) ve DL tabanlı modeller (MLP, CNN, WdCNN, SepCNN) yer almıştır. Ayrıca birleştirilmiş modeller de kullanılmıştır. Tablo 29 incelendiğinde, XGB modeli en yüksek R^2 , MSE, ve RMSE değerleriyle GPS verileri üzerinde en başarılı model olmuştur (MSE: 0,1402, RMSE: 0,3745, R^2 : 0,7008). Bu model, tahmin edilen ve gerçek değerler arasında iyi bir uyum olduğunu göstermektedir. MAE açısından en düşük hata oranı, KNR-DTR modeliyle (0,1795) elde edilmiştir. MLP modeli (4B tensör girişi), tüm metriklerde en kötü sonuçları vermiştir. Topluluk modelleri, tekil modellere kıyasla daha düşük hata oranlarıyla üstün performans sergilemiştir. KNR, KNR-DTR-LR-XGB-RFR, KNR-DTR-LR-XGB-RFR-MLPRegressor (4B tensör girişi), KNR-DTR ve LR-XGB-RFR modelleri yüksek R^2 değerleri sağlamıştır. Şekil 85 ve 86, tahmini boylam sonuçlarını göstermektedir. Veri setinin gerçek boylam değerleri ile karşılaştırıldığında, genel olarak ML tabanlı modellerin DL tabanlı modellere göre

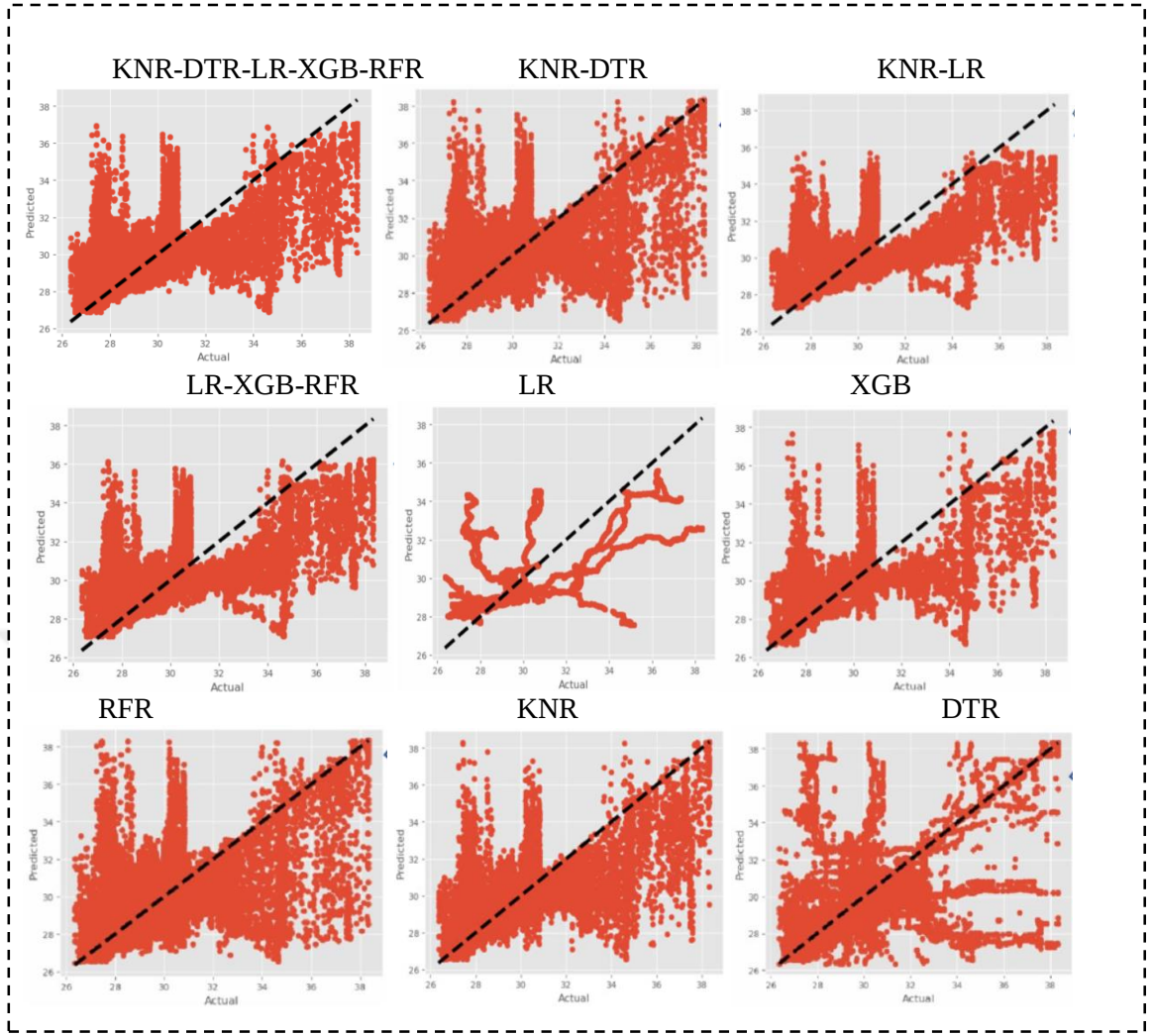
gerçeğe daha yakın tahmin sonuçları verdiği gözlemlenmiştir. En iyi konumlandırma doğruluğunu sağlayan model XGB'dir. XGB modeli, GPS verilerinin heterojen yapısına ve yüksek boyutluluğuna uyum sağlayarak en iyi sonuçları vermiştir. Bu başarı, modelin veri setinin özelliklerine uygunluğuna, dahili düzenleme mekanizmalarına ve düşük hesaplama karmaşıklığına dayanmaktadır. CNN, WdCNN, ve SepCNN gibi modeller, özellikle mekânsal özelliklerin öğrenilmesinde başarılı olmuştur. Ancak, katman veya nöron sayısının artırılması performansı önemli ölçüde iyileştirmemiştir. 4B tensör girişli modeller, 1B girişe göre daha yüksek doğruluk sağlamıştır. Ancak, MLP ve MLPRegressor modelleri, 1B girişle daha iyi performans göstermiştir.

Tablo 29. Karşılaştırmalı MAE, MSE, R^2 ve RMSE değerleri: GPS Veri Seti Üzerinde Uygulanan Yöntemlerin Performans Analizi (Erdem *et al.* under review e)

Model		MAE	MSE	RMSE	R^2
KNR-DTR-LR-XGB-RFR		0,1865	0,1480	0,3847	0,6842
KNR-DTR		0,1795	0,1549	0,3935	0,6696
KNR-LR		0,2181	0,1710	0,4135	0,6352
LR-XGB-RFR		0,1984	0,1549	0,3936	0,6695
XGB		0,1815	0,1402	0,3745	0,7008
RFR		0,1796	0,1653	0,4066	0,6473
KNR		0,1831	0,1455	0,3814	0,6896
Boyut	Model	MAE	MSE	RMSE	R^2
1B	KNR-DTR-LR-XGB-RFR- MLPRegressor	0,2039	0,1788	0,4228	0,6186
4B	KNR-DTR-LR-XGB-RFR-MLPRegressor	0,1929	0,1507	0,3882	0,6784
1B	LR-XGB-RFR-MLPRegressor	0,2061	0,1774	0,4212	0,6216
4B	LR-XGB-RFR-MLPRegressor	0,2065	0,1622	0,4028	0,6539



Şekil 85. GPS verisetinde ML ve topluluk modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Erdem *et al.* under review e)



Şekil 86. GPS verisetinde ML modelleriyle test aşamasında elde edilen gerçek ve tahmin edilen değerler arasındaki korelasyon (Erdem *et al.* under review e)

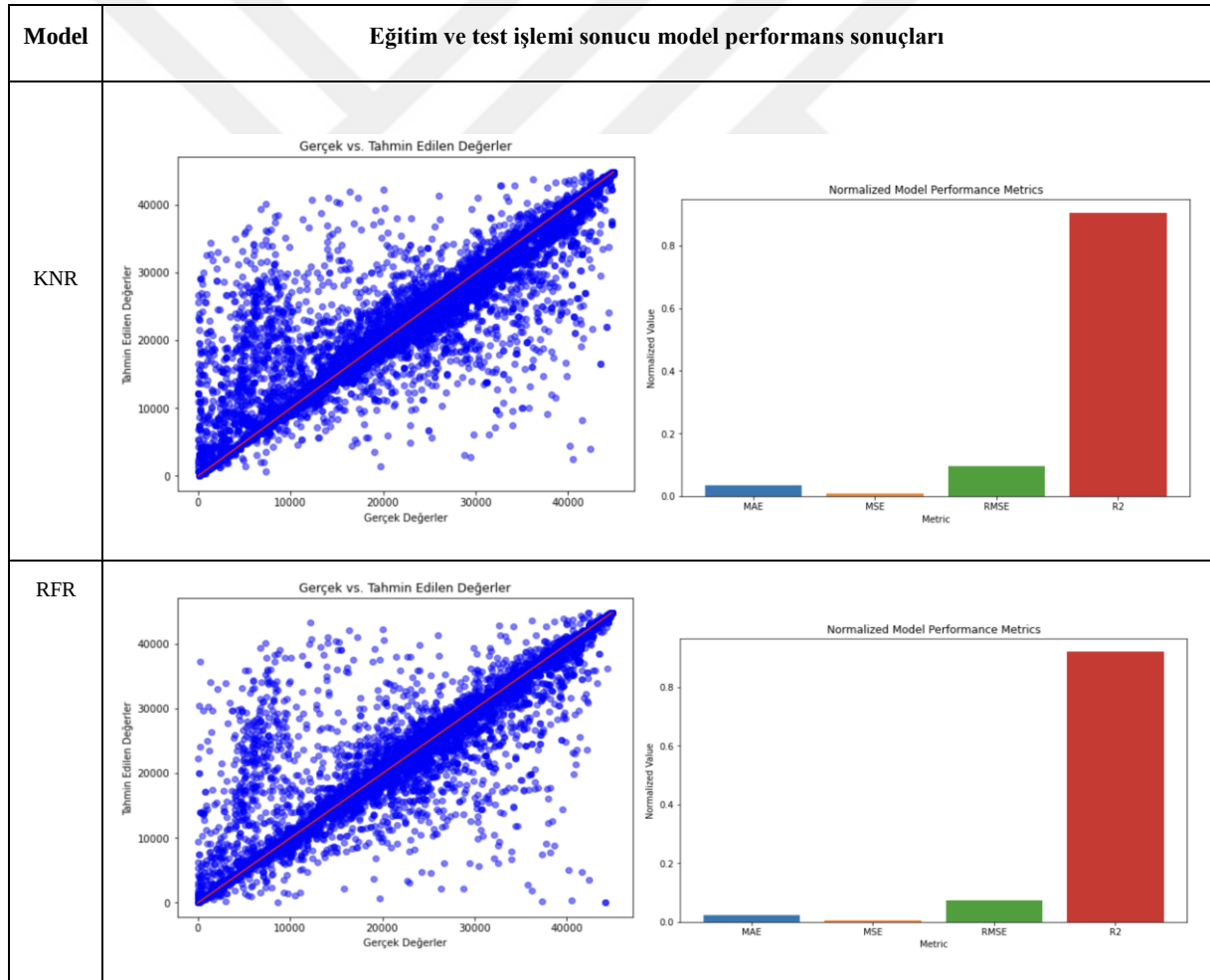
Tezin üçüncü aşamasında, algoritmalar Mikro V15 ve GPS veri setleri kullanılarak test edilmiştir. Mikro V15-GPS entegrasyonu ve ML algoritmaları ile yapılan ARP çözümünde, GPS verileri zaman-mesafe matrisinin oluşturulmasında temel bir veri kaynağı olarak kullanılmıştır (Şekil 87).



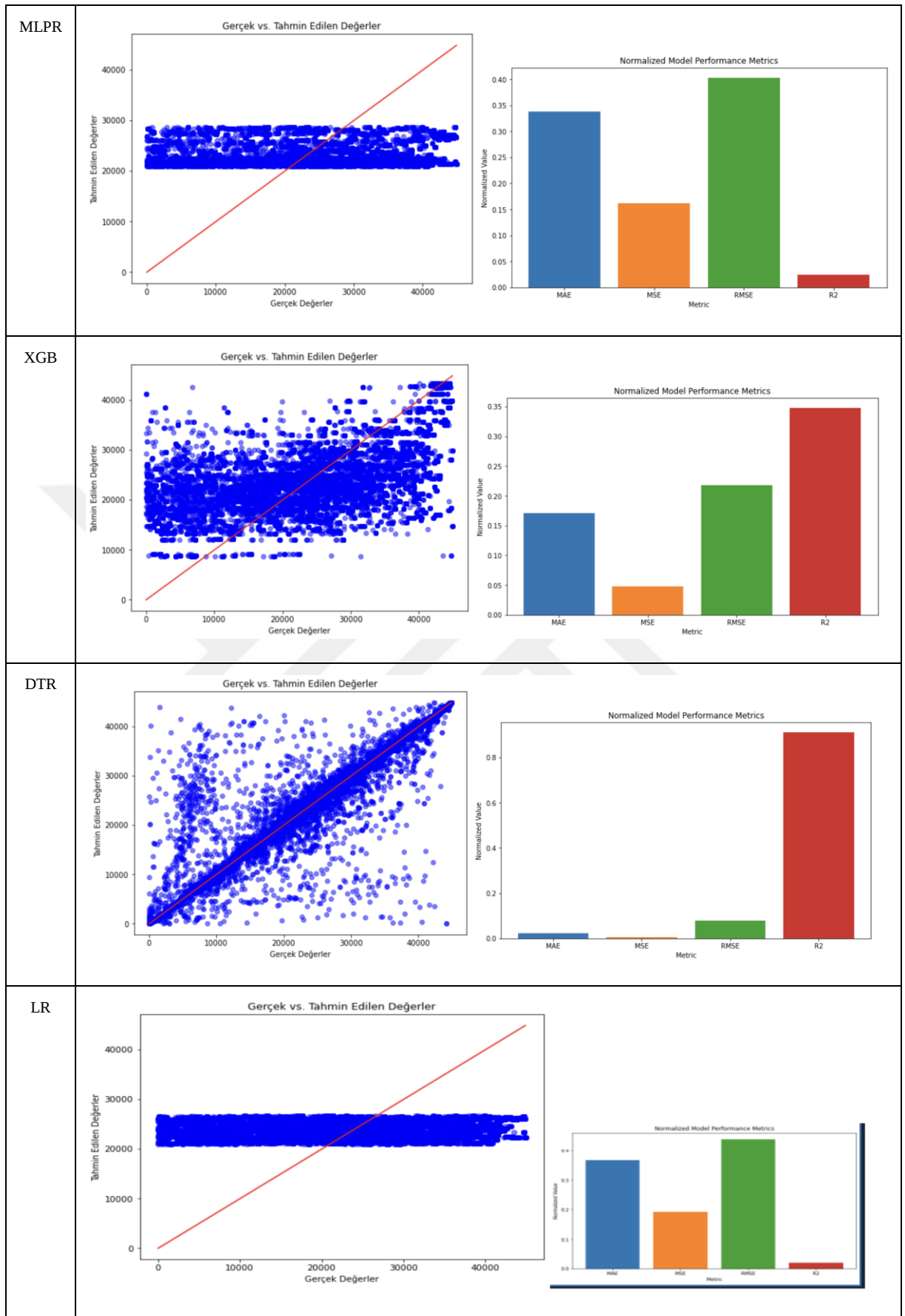
Şekil 87. Diğer algoritmalar için pivot özelliği taşıyan GPS-tabanlı zaman mesafe matrisi

Bu çalışma kapsamında gerçekleştirilen deneyler, Mikro V15’de 31 Mart 2023 tarihindeki öğlen siparişlerinden seçilen 29 siparişe dayanmaktadır. Deney sonuçları, tüm algoritmaların veri seti üzerinde ortalama beş kez test edilmesiyle elde edilmiştir.

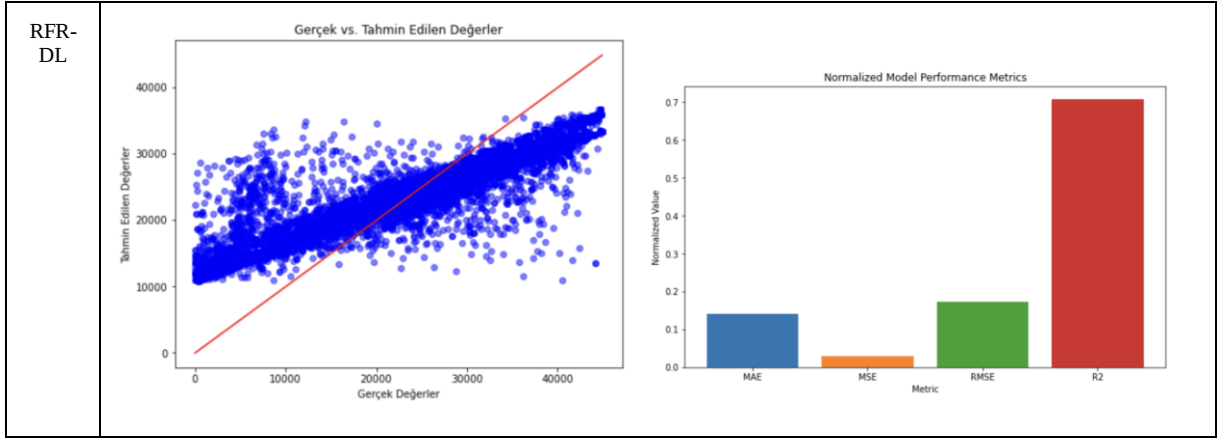
Mikro V15-GPS ve ML algoritmalarını entegre ederek ARP problemini çözmeyi hedefleyen sistem, ilk olarak Şekil 41’de sunulan ARP için AI Tabanlı Yaklaşımın Faz I aşamasını tamamlamıştır. Bu aşamada, ML modellerinin eğitimi gerçekleştirilmiştir. Şekil 88 ve 89, çeşitli test metrikleri (MAE, MSE, RMSE ve R^2 hata oranı) kullanılarak her bir modelin performansını göstermektedir. MAE ve RMSE değerleri incelendiğinde, en yüksek performans (MAE ve RMSE < 0,2) KNR, RFR, DTR ve RFR-DL modellerinden elde edilmiştir. MSE değerleri analiz edildiğinde, en iyi sonuçlar (MSE < 0,05) yine KNR, RFR, DTR ve RFR-DL modellerinden elde edilmiştir. R^2 hata oranı değerlendirildiğinde ise en yüksek performansı ($R^2 > 0,90$) RFR, DTR ve KNR modelleri göstermiştir (Erdem *et al.* under review b).



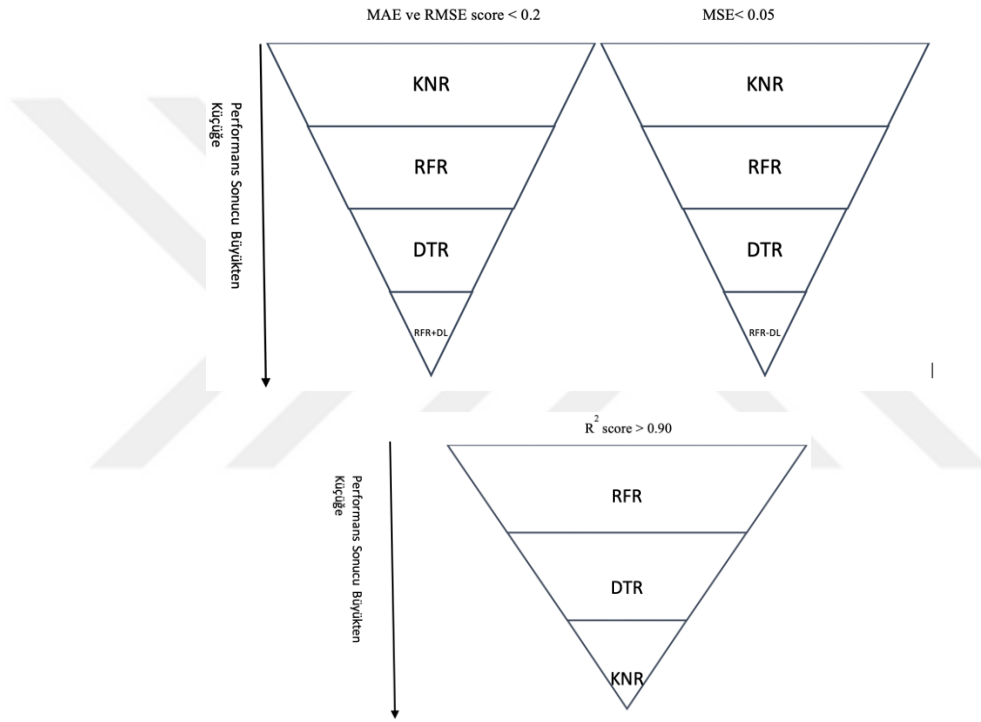
Şekil 88. ML modelleri uygulandıktan sonra performans sonuçları (Erdem *et al.* under review b)



Şekil 88. ML modelleri uygulandıktan sonra performans sonuçları (Erdem *et al.* under review b) (devamı)



Şekil 88. ML modelleri uygulandıktan sonra performans sonuçları (Erdem *et al.* under review b) (devamı)

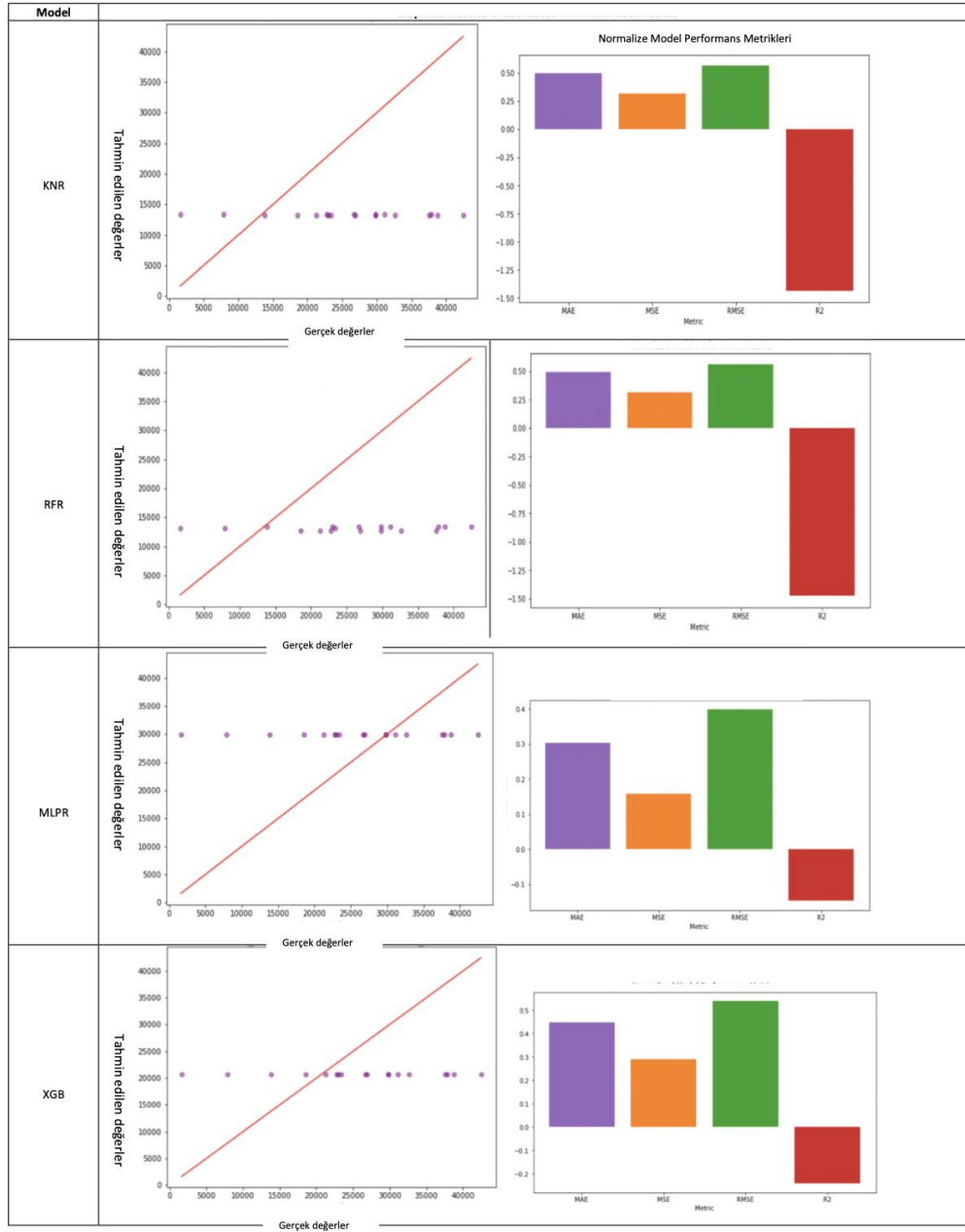


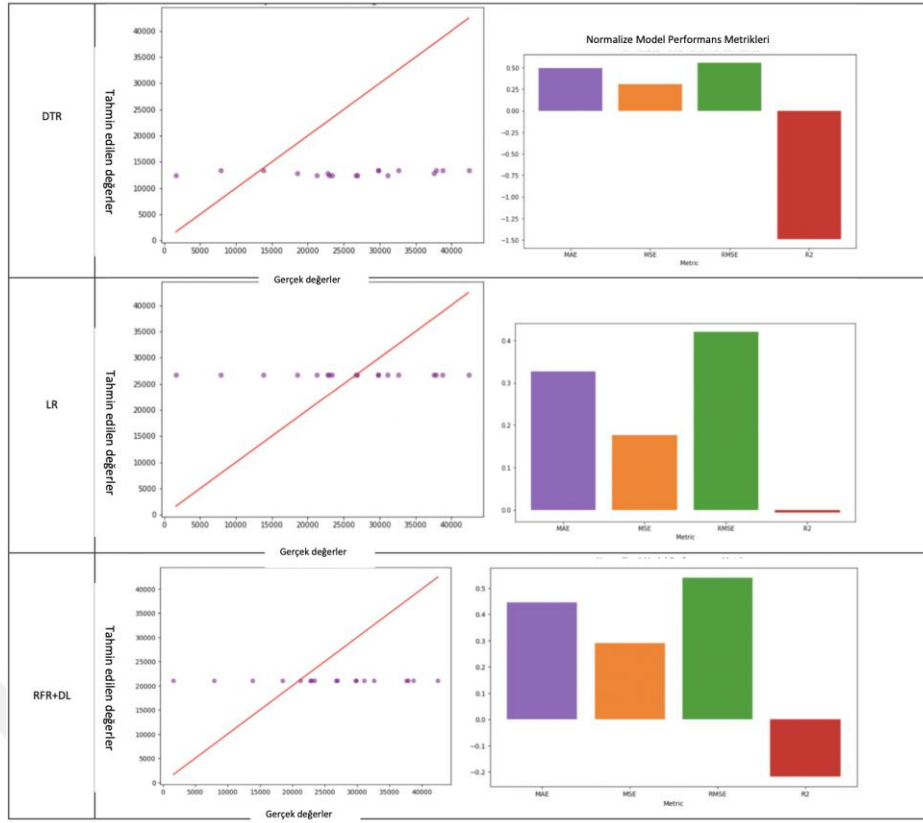
Şekil 89. MAE, MSE ve R² ölçütlerine göre ML modellerinin karşılaştırmalı performans sıralamaları (Erdem *et al.* under review b)

Ardından, Şekil 43’de gösterilen ARP için AI Tabanlı Yaklaşımın Faz II aşaması tamamlanmıştır. Minimum maliyetli rota ve her bir ML modeli için test metriklerine ait sonuçlar, Şekil 90 ve 91’de sunulmuştur. Tablo 30, tarihsel GPS ve Mikro V15 verilerinin analizi sonucunda ML kullanılarak elde edilen minimum maliyetli rotanın maliyet sonuçlarını içermektedir. ML hata skorlarında en yüksek performans, sırasıyla LR, MLPR, RL-DL, XGB, KNR, RFR ve DTR modelleri ile elde edilmiştir. Hibrit model, en başarılı modelle oldukça benzer sonuçlar verdiği için etkili bir seçenek olarak değerlendirilmiştir. Bunun yanı sıra, LR modelinin başarı oranı, tüm test metriklerinde pozitif eğilim gösteren grafiğin eğrisiyle genel olarak desteklenmiştir.

Tablo 30. ML Modellerinin Performansının Karşılaştırılması (Erdem *et al.* under review b)

Model	Minimum Cost
KNR	2,68174e+08
RFR	2,69447e+08
MLPR	1,2606e+08
XGB	1,36582e+08
DTR	2,7089e+08
LR	1,10702e+08
RFR-DL	1,771266e+08

**Şekil 90.** ML modelleri (KNR, RFR, MLPR, XGB) uygulandıktan sonra elde edilen minimum maliyetli rota ve test metriği sonuçları (Erdem *et al.* under review b)

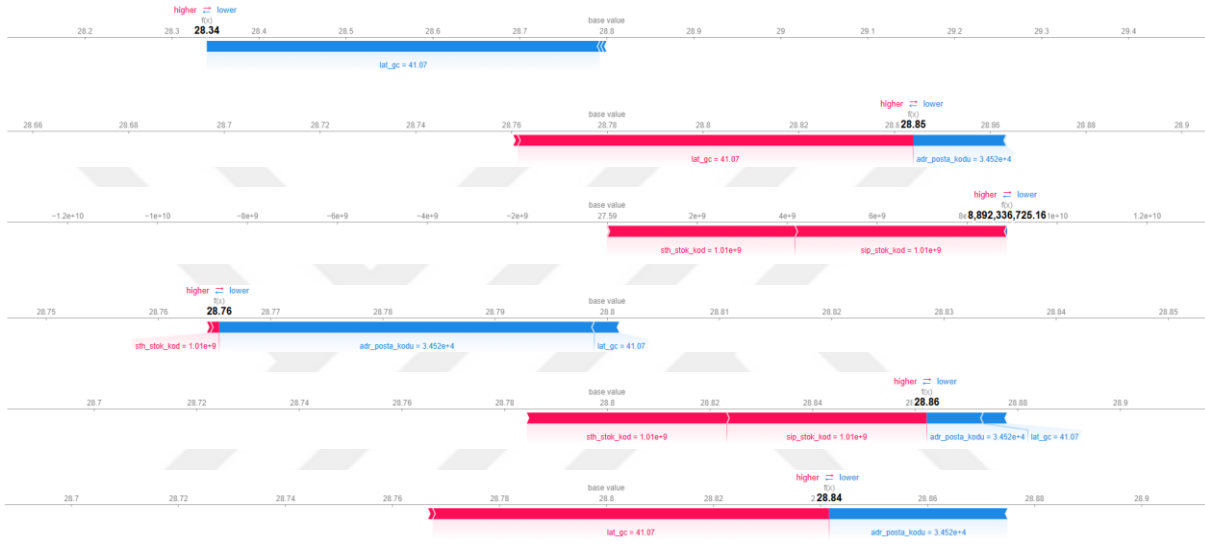


Şekil 91. ML modelleri (DTR, LR, RFR-DL) uygulandıktan sonra elde edilen minimum maliyetli rota ve test metriği sonuçları (Erdem *et al.* under review b)

Şekil 94, Mikro V15-GPS entegrasyonu kullanılarak gerçekleştirilen rota planlama sürecinin grafiksel sonuçlarını sunmakta ve filonun ÇDARP çözümünü tamamlama süresini göstermektedir. Bu süreçte, algoritmalara üç farklı başlangıç çözümü (S_{rd} , S_{dyd} ve S_{mld}) giriş olarak verilmiştir. ML modelleri kullanılarak sağlanan başlangıç çözümlerinde en yüksek başarıya sahip ML modeli (Tablo 30) dikkate alınmış ve bu nedenle S_{mld} başlangıç çözümü için LR algoritması tarafından belirlenen minimum maliyetli rota tercih edilmiştir.

ML tabanlı başlangıç çözüm stratejilerinin etkinliğini değerlendirmek amacıyla, DTR, KNR, LR, MLPR, RFR ve XGB modellerine ait Shapley Katkılı Açıklamalar (SHAP) kuvvet diyagramları oluşturulmuş ve yorumlanmıştır (Şekil 92). Bu diyagramlar, her bir girdinin model tahminine olan etkisini yerel düzeyde şeffaf biçimde ortaya koyarak, model iç görülerinin analizine olanak tanımıştır. RFR modeli, “lat_gc” (enlem) değişkeninin rota süresini azaltıcı etkisini tutarlı biçimde göstermiş; coğrafi desenleri başarılı şekilde öğrenerek istikrarlı ve yorumlanabilir tahminler üretmiştir. XGB modeli ise enlem ve adres temelli değişkenleri içeren daha karmaşık etkileşimleri yakalamış, ancak giriş değişimlerine karşı daha hassas sonuçlar vermiştir. MLPR modeli doğrusal olmayan yapılarla çalışsa da, SHAP değerleri dağınık olmuştur. LR modeli ise yüksek şeffaflık sunmakla birlikte, doğrusal sınırlılığı nedeniyle karmaşık ilişkileri yeterince temsil edememiştir. KNR modeli örnek tabanlı yapısı gereği

katkıları homojen dağıtmış; DTR modeli ise kategorik değişkenlere odaklanarak ani tahmin değişimlerine yol açmıştır. SHAP analizleri, en yüksek açıklanabilirlik ve tahmin istikrarının RFR ve XGB modellerinde sağlandığını göstermiştir. Bununla birlikte, rota maliyeti açısından en iyi sonuç LR modeli ile elde edilmiştir. Bu nedenle meta-sezgisel algoritmalara başlangıç çözümü olarak LR modelinin ürettiği rota (S_{mld}) verilmiştir. Bu durum, sınırlı doğrusal yapısına rağmen LR'nin operasyonel verimlilik açısından güçlü çözümler sağlayabildiğini ortaya koymuştur. Sonuç olarak, SHAP görselleştirmeleri yalnızca model iç görülerini derinleştirmemiş, aynı zamanda ML tabanlı başlangıç çözümlerinin yorumlanabilirliğini artırarak ARP optimizasyon süreçlerinde stratejik fayda sağlamıştır.

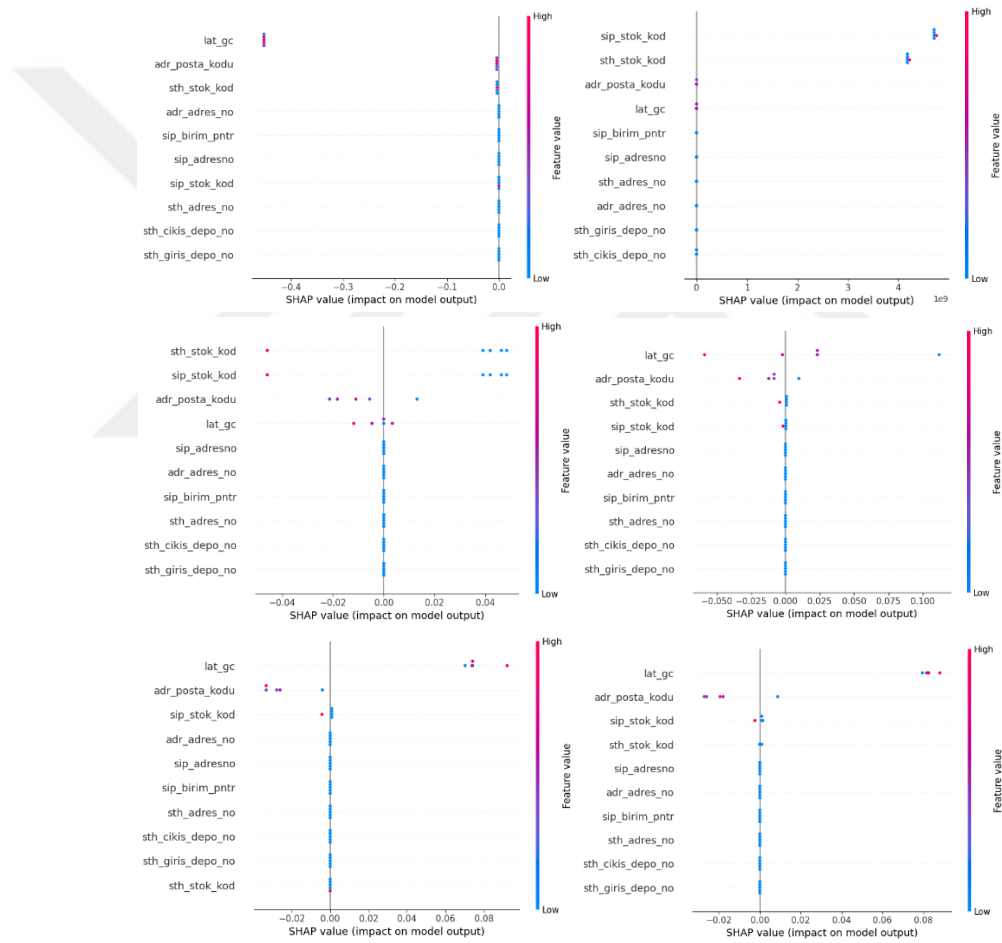


Şekil 92. ML tabanlı başlangıç modelleri (sırasıyla, XGB, RFR, MLPR, LR, KNR ve DTR) için özellik etkilerinin SHAP kuvvet grafiği ile görselleştirilmesi (Erdem *et al.* under review e)

ML modellerinin performans metrikleriyle yapılan ilk değerlendirmesinin ardından, model tahminlerinde özelliklerin katkılarını incelemek amacıyla bir SHAP analizi gerçekleştirilmiştir (Şekil 93). Kuvvet (force) ve özet grafikler, tüm modellerin “lat_gc”, “adr_posta_kodu” ve “sip_stok_kod” gibi bazı ortak özellikleri tutarlı biçimde önemli olarak tanımladığını; ancak bu özelliklerin etkisinin büyüklüğü ve yönünün modele göre değişiklik gösterdiğini ortaya koymuştur. XGB modeli, stokla ilişkili kodlara (“sip_stok_kod”, “sth_stok_kod”) güçlü ve istikrarlı bir şekilde bağımlı olduğunu göstermiş; bu değişkenlerin öngörü performansındaki önemini vurgulamıştır. Öte yandan, RFR modeli SHAP değerlerini daha dengeli dağıtarak hem coğrafi (enlem) hem de adres tanımlayıcılarından orta düzeyde katkılar göstermiştir. MLPRegressor modelinin SHAP çıktıları, bazı baskın özellikleri tutarlı şekilde vurgulasa da, ağır doğrusal olmayan yapısı nedeniyle yorumlanabilirlik açısından daha düşüktür; bu durum, doğrusal modellerde gözlemlenen daha net ilişkilerle tezat oluşturmuştur. Ayrıca, DTR ve KNR modelleri daha yerel SHAP dağılımları üretmekte olup, küresel

desenlerden ziyade belirli örneklere duyarlılık sergilemiştir. LR modeli ise SHAP grafiğinde sade ve doğrudan bir yapı sergilemiştir. “lat_gc” ve “adr_posta_kodu” gibi özelliklerin etkileri düşük ancak dengelidir. Özellik katkıları model katsayılarıyla doğrudan örtüştüğü için yüksek şeffaflık sunmuş; ancak doğrusal olmayan ilişkilerde sınırlı kalmıştır. Bununla birlikte, LR modelinin minimum rota maliyeti sağlayan çözümler üretmesi, onu yorumlanabilirliği yüksek ve operasyonel olarak avantajlı bir başlangıç modeli haline getirmiştir.

SHAP özet ve güç grafikleri ile temsil edilen bu karşılaştırmalı görselleştirmeler, model mimarisinin özellik katkılarının hem boyutunu hem de yorumlanabilirliğini nasıl şekillendirdiğini ortaya koymuş; böylece her bir algoritmanın temel mekânsal lojistik veri kümesinin yapısını yakalama uygunluğuna dair önemli iç görüler sunmuştur.



Şekil 93. ML tabanlı başlangıç modellerine ait SHAP kuvvet diyagramı görselleştirmeleri (sırasıyla MLPR, XGB, KNR, LR, DTR ve RFR) (Erdem *et al.* under review e)

→ BT ve MBT Sayısal Parametreleri Tablo 31’de verilmiştir (Erdem *et al.* under review b).

Tablo 31. BT ve MBT Sayısal Parametreleri

Algoritma	Başlangıç Sıcaklığı	Soğutma Oranı	Durdurma Kriteri	İterasyon Sayısı
TB	100	0,9	0,001	100
MBT	100	0,9	0,001	100

→ TA Sayısal Parametreleri Tablo 32’de verilmiştir (Erdem *et al.* under review b).

Tablo 32. TA Sayısal Parametreleri

Algoritma	İterasyon Sayısı	Tabu Liste Boyutu
TA	250	20

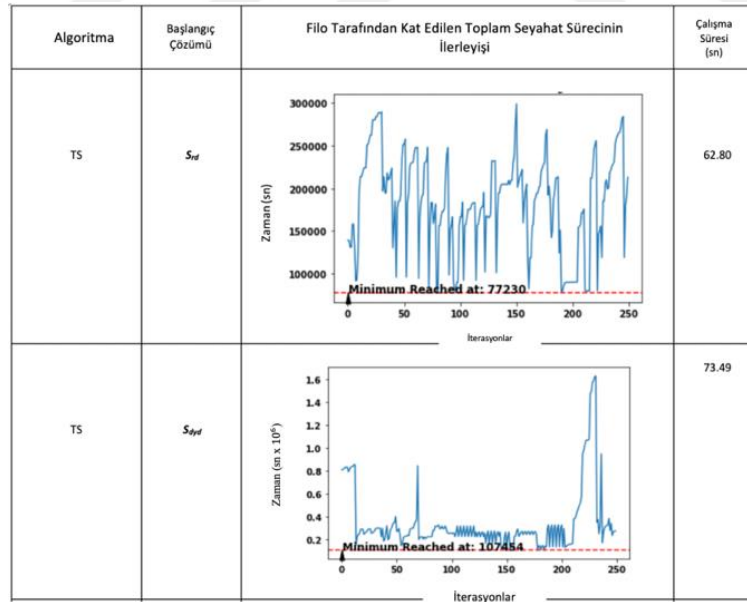
Şekil 94 (Erdem *et al.* under review b) incelendiğinde, S_{mld} başlangıç çözümü ile ARP çözümünde en düşük maliyetlerin sırasıyla MBT, TA ve BT algoritmaları ile elde edildiği görülmektedir. Özellikle, S_{mld} başlangıç çözümünü kullanan MBT algoritması, siparişlerin sevkiyat sürecini 11 saat içerisinde tamamlayarak en iyi sonucu sağlamıştır. Bu bulgu, S_{mld} başlangıç çözümünün rota planlama sürecinde minimum maliyet elde edilmesinde önemli bir avantaj sağladığını göstermektedir. Üç başlangıç çözümüne göre değerlendirildiğinde, önerilen MBT algoritması, BT algoritmasına kıyasla daha düşük rota maliyetleri sunmuş ve böylelikle BT algoritmasının geliştirilmiş bir versiyonu olarak tanımlanmıştır. Diğer taraftan, S_{rd} ve S_{dyd} başlangıç çözümleri ile yapılan ARP çözümünde ise en düşük maliyetler sırasıyla TA, MBT ve BT algoritmaları ile elde edilmiştir.

Algoritmaların çalışma süreleri analiz edildiğinde, üç başlangıç çözümü için de en kısa süre TA algoritması ile elde edilmiştir. TA ve MBT algoritmalarında toplam sevkiyat süresi, $S_{mld} < S_{rd} < S_{dyd}$ sırasını takip etmiştir. Bu durumun sebebi, müşterilerin depolara yakın bir şekilde dağıtıldığı durumlarda algoritmaların daha fazla yerel optimizasyon yapma eğiliminde olmasıdır. Buna karşın, rastgele oluşturulan başlangıç çözümleri daha az optimize edilmiş olduğu için algoritmaların daha geniş bir çözüm alanını keşfetmesine olanak tanımaktadır. ML modelleri ile oluşturulan başlangıç çözümleri ise genellikle daha iyi optimize edilmiş bir başlangıç noktası sunmakta ve bu durum algoritmaların daha az geliştirme yapmasını gerektirerek genel işlem süresini kısaltmaktadır. BT algoritması için toplam sevkiyat süresi, $S_{dyd} < S_{rd} < S_{mld}$ sırasındadır. Bu durum, ML tarafından oluşturulan başlangıç çözümlerinin daha karmaşık olabileceği ve BT algoritmasının daha uzun arama süreçlerine girmesine neden olabileceği şeklinde açıklanabilir. Rastgele dağıtılan çözümler, algoritmanın daha geniş bir çözüm alanını keşfetmesine olanak tanırken, depolara yakın müşteri dağılımları BT

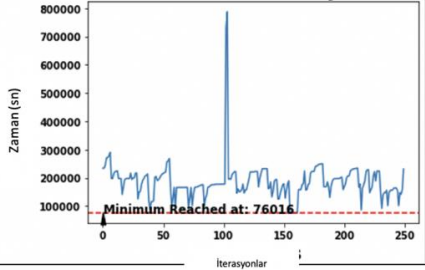
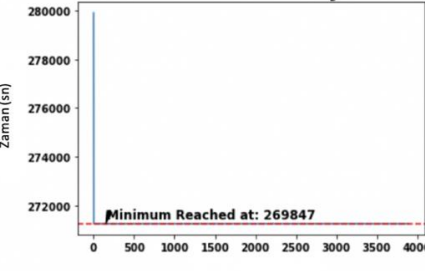
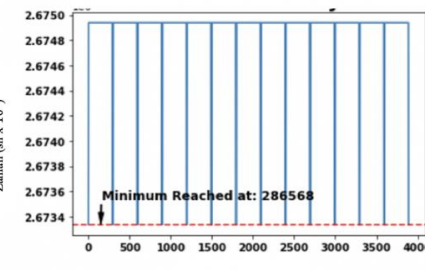
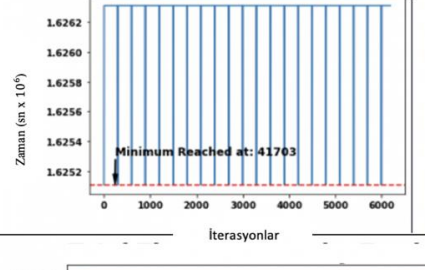
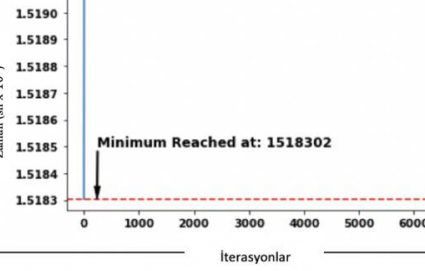
algoritmasının daha hızlı stabilizasyon sağlamasına ve yerel optimumlara daha hızlı ulaşmasına neden olmaktadır.

Grafikte gözlemlenen dalgalanmalar, algoritmaların çözüm iyileştirme süreçlerini yansıtmaktadır. TA algoritmasındaki belirgin dalgalanmalar, algoritmanın geniş bir çözüm alanını keşfetme ve yerel optima tuzaklarından kaçma çabasını göstermektedir. Buna karşın, BT algoritması daha erken bir stabilizasyon göstererek çözümler arasında daha az hareket ve sınırlı iyileştirme potansiyeli sunmaktadır. MBT algoritmasındaki daha büyük dalgalanmalar, algoritmanın daha kapsamlı bir arama gerçekleştirdiğini ve yerel optimum tuzaklarını daha etkili bir şekilde aşabildiğini göstermektedir.

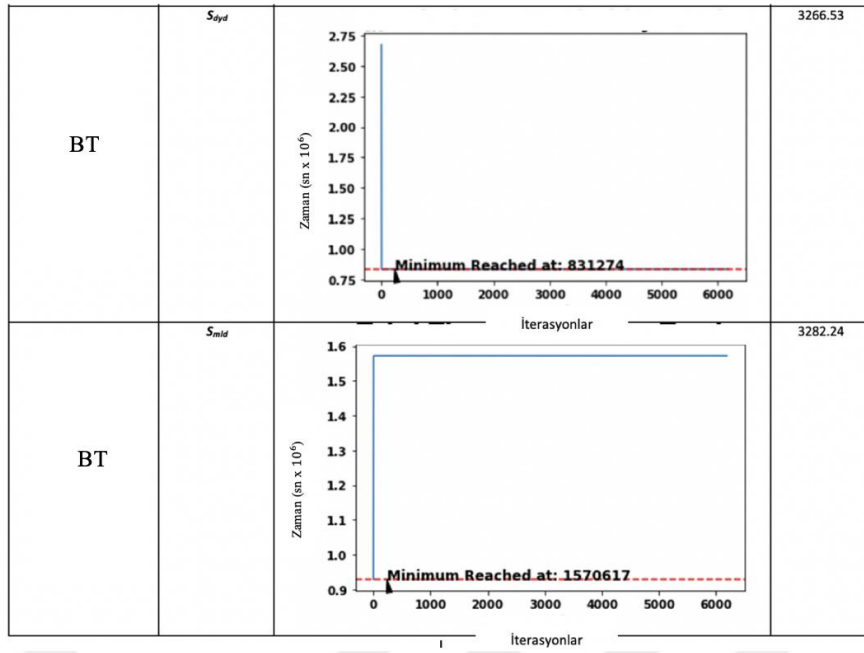
BT algoritmasının diğer iki sezgisel algoritmaya göre daha düşük performans göstermesi, yerel optimum tuzaklarına yakalanma eğiliminin artması ve küresel optimum ulaşma yeteneğinin sınırlı olmasına bağlanabilir. Buna karşın, TA ve MBT algoritmaları, tabu listesi, mutasyon ve ekstra kontrol süreçleri gibi ek mekanizmalar kullanarak bu sorunların üstesinden gelmektedir. TA algoritması, tabu listesi sayesinde yinelenen çözümlerden kaçınarak daha geniş bir arama alanı sağlar ve parametre duyarlılıklarından bağımsız olarak daha tutarlı sonuçlar sunar. MBT ise BT algoritmasını aşan çeşitli iyileştirme ve modifikasyonlarla daha kapsamlı bir arama gerçekleştirme kabiliyetine sahiptir.



Şekil 94. Üç farklı başlangıç çözümüne göre filonun kat ettiği toplam süre (Erdem *et al.* under review b)

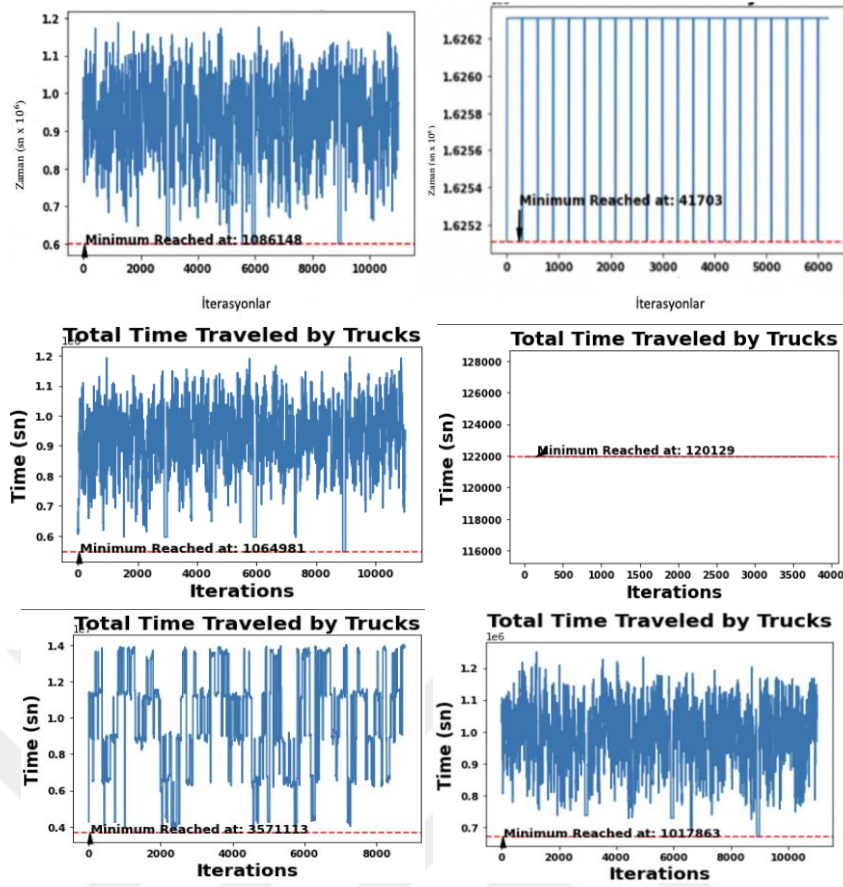
TA	S_{mid}		63.71
MBT	S_{id}		833.68
MBT	$\dot{\phi}_{dayd}$		832.53
MBT	S_{mid}		3313.88
BT	S_{id}		3285.96

Şekil 94. Üç farklı başlangıç çözümüne göre filonun kat ettiği toplam süre (Erdem *et al.* under review b) (devamı)

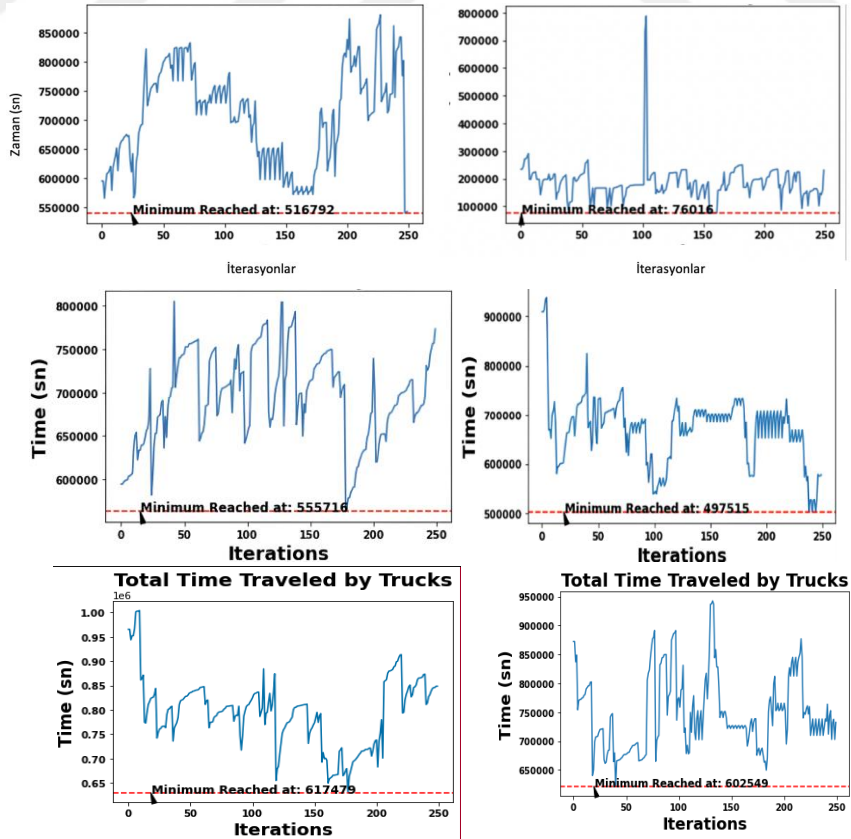


Şekil 94. Üç farklı başlangıç çözümüne göre filonun kat ettiği toplam süre (Erdem *et al.* under review b) (devamı)

Şekil 94’de (Erdem *et al.* under review b) sunulan sonuçlara göre, MBT ve TA algoritmaları, en yüksek performansı sergileyen algoritmalar olarak ön plana çıkmıştır. Bu nedenle, bu algoritmaların etkinliği, OSRM API kullanılarak çözölen klasik ÇDARP problemi ile karşılaştırılmıştır. Şekil 95 ve 96’da göröldüğü üzere (Erdem *et al.* under review b), geçmiş GPS verilerinden türetilen zaman-mesafe matrisi kullanılarak yapılan ARP çözümü, OSRM API tarafından oluşturulan zaman-mesafe matrisine dayalı ARP çözöümünden daha iyi sonuçlar vermiştir. Bu bulgular, tarihsel GPS verileriyle oluşturulan zaman-mesafe matrislerinin, ARP problemlerinde OSRM API’ye kıyasla önemli bir avantaj sunduğunu ortaya koymaktadır. Tarihsel GPS verileri, zaman içinde biriken trafik eğilimleri, yol koşulları ve diğör dinamik faktörler hakkında değörlü içgöröler sağlamaktadır. Belirli rotalardaki trafik yoğunluğu desenleri ve tekrarlayan problemleri belirlemeye olanak tanıyan bu veriler, uzun vadeli eğilimlerin anlaşılmasını kolaylaştırmaktadır. Buna karşın, OSRM API yalnızca mevcut trafik koşullarını temel alarak çalışmakta ve tarihsel verileri hesaba katmamaktadır. Sonuç olarak, tarihsel GPS verileri, daha kapsamlı ve ayrıntılı bir veri temeli sunarak, ARP problemlerinin çözöümünde OSRM API’ye kıyasla daha doğöru ve etkili sonuçlar sağlamaktadır.



Şekil 95. Sırasıyla OSRM API ve GPS verileriyle oluşturulan zaman-mesafe matrisleri kullanılarak elde edilen MBT performans sonuçları (Erdem *et al.* under review b)

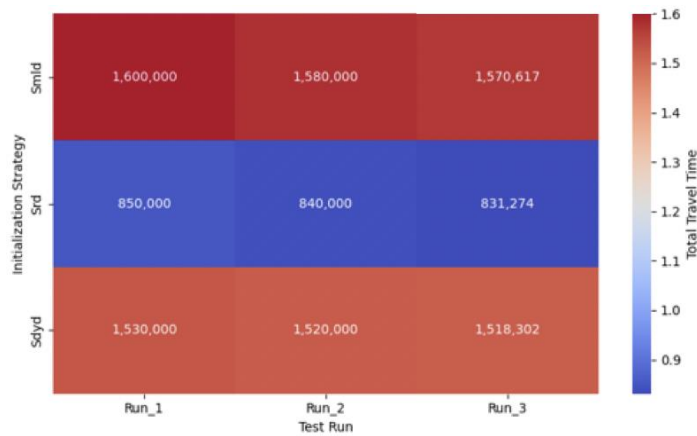


Şekil 96. Sırasıyla OSRM API ve GPS verileriyle oluşturulan zaman-mesafe matrisleri kullanılarak elde edilen TA performans sonuçları (Erdem *et al.* under review b)

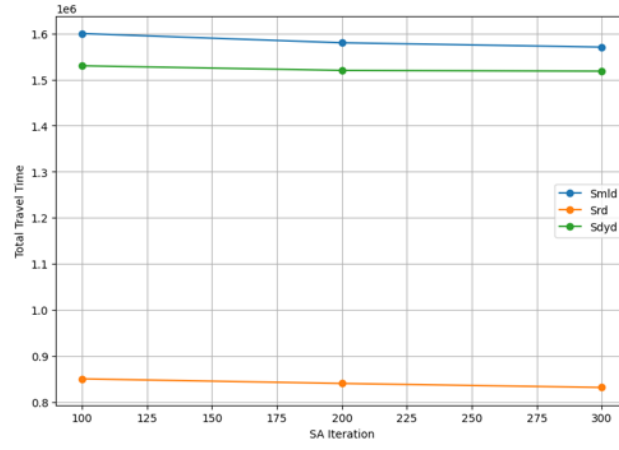
Önerilen MBT- S_{mld} tabanlı rota optimizasyon modeli, klasik çözüm yöntemleriyle karşılaştırılmış ve iki temel performans göstergesi olan toplam rota süresi ile toplam maliyet açısından anlamlı iyileştirmeler elde edilmiştir. Klasik yöntemle elde edilen ortalama toplam rota süresi yaklaşık 16,9 saat olarak hesaplanmıştır. Önerilen MBT- S_{mld} çözümü ise bu süreyi yaklaşık 11 saate düşürmüştür. Bu iyileşme, $(16,9 - 11) / 16,9 \approx 0,35$ hesaplama adımıyla oranlanmıştır. Toplam maliyet hesaplamasında, araç sayısı ve sürüş süresine dayalı operasyonel giderler esas alınmıştır. Karşılaştırma amacıyla klasik modelin maliyeti normalleştirilmiş değer olarak 100 birim kabul edilmiştir. Önerilen modelle bu değer yaklaşık 70 birime düşmüştür. Bu iyileşme, $(100-70)/100 = 0.30$ hesaplama adımıyla oranlanmıştır. Böylece rota süresinde yaklaşık %35, maliyetlerde ise %30 oranında iyileşme sağlanmıştır.

“ S_{mld} ”, “ S_{dyd} ” ve “ S_{rd} ” stratejileri için BT, MBT ve TA algoritmaları özelinde yapılan deneylerde, önceki bölümlerde ortalama 5 tekrar üzerinden sonuçlar raporlanmışken; bu bölümdeki ısı haritası ve çizgi grafik görselleştirmeleri üç bağımsız çalıştırmanın (Run_1, Run_2, Run_3) çıktıları esas alınarak oluşturulmuştur. Bu yapılandırma, görsel netliği artırmak ve her stratejinin algoritmalara özgü davranış örüntülerini daha açık bir biçimde sunmak amacıyla tercih edilmiştir. BT algoritmasına ait ısı haritasında (Şekil 97) açıkça görüldüğü üzere, S_{mld} stratejisi tüm test koşullarında en yüksek seyahat sürelerine yol açarken, S_{rd} stratejisi en düşük toplam seyahat süresiyle en iyi performansı sergilemiştir. Boxplot grafiği analizi (Şekil 99) bu bulguyu desteklemiştir. S_{mld} tabanlı başlangıç çözümleri en geniş seyahat süresi dağılımına sahipken, S_{rd} en düşük ve en istikrarlı dağılımı göstererek daha etkili bir başlangıç keşfi yeteneği sunmuştur. Bu durum, S_{mld} stratejisinin geçmiş verilerden öğrenilmesine rağmen, mevcut müşteri veya depo dağılımlarının eğitim sırasında gözlemlenen yapılardan önemli ölçüde farklı olması halinde optimalden uzak çözümlerle sonuçlanabileceğini göstermiştir. Şekil 98’de yer alan çizgi grafiği, BT algoritmasının iterasyonlar boyunca yakınsama davranışını ortaya koymuştur. S_{mld} ile başlatılan BT algoritması, yüksek toplam seyahat süresiyle başlamış ve bu seviyeyi sürdürmüştür. Öte yandan, S_{rd} stratejisi en dik ve tutarlı iyileşme eğrisine sahip olup en düşük değerlerde yakınsamıştır. Bu sonuçlar, S_{mld} ’nin başarımının eğitim verisi ile test senaryosu arasındaki dağılım uyumuna oldukça duyarlı olduğunu ortaya koymuştur. S_{mld} , rotalama yapısı öğrenilen örüntülerle örtüştüğünde oldukça etkili olabilse de, daha dinamik ya da yeni rotalama senaryolarında yakınsama dengesizliği veya yerel tuzaklara yol açabilir. Bu da, farklı çalışma koşullarında sağlamlık sağlamak adına ML tabanlı başlangıç çözümlerinin hibrit ya da uyarlanabilir stratejilerle geliştirilmesi gerektiğini vurgulamaktadır. MBT algoritmasına ait ısı haritası (Şekil 100) ve çizgi grafiği (Şekil 101), S_{mld} tabanlı başlangıç stratejisinin Run_3’te en düşük toplam seyahat süresine ulaşarak S_{dyd} ve S_{rd} stratejilerinden açık biçimde üstün olduğunu göstermiştir. Bu üstünlük, kutu grafiği

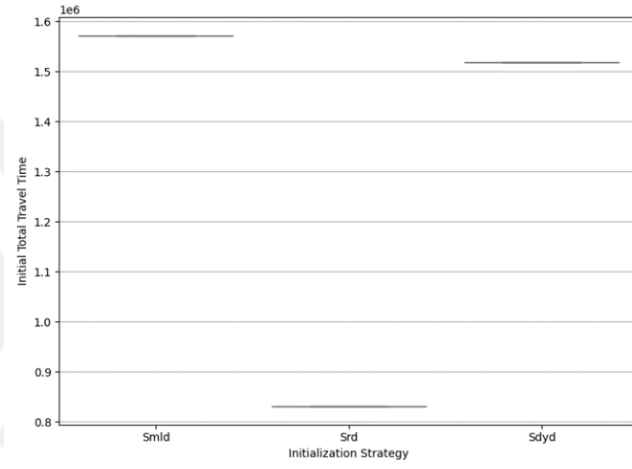
analiziyle de (Şekil 102) desteklenmiştir: S_{mld} yalnızca en düşük medyan değeri üretmekle kalmamış, aynı zamanda seyahat sürelerinin dağılımı açısından da en düşük yayılma değerine sahip olarak hem verimlilik hem de kararlılık göstermiştir. Bu sonuçlar, MBT algoritmasının ML tabanlı giriş çözümlerinin yapısal avantajlarını daha etkili biçimde kullanabildiğini göstermiştir. BT algoritmasında S_{mld} alt optimal yakınsamaya ve yüksek varyansa yol açarken, MBT erken yerel optimalara düşmeden, S_{mld} yapısındaki örüntüleri etkili biçimde değerlendirmiştir. Uyarlanabilir arama dinamikleri, algoritmanın başlangıç yapısı tarihsel önyargılar taşısa bile çözüm üzerinde daha verimli iyileştirmeler yapabilmesini sağlamıştır. Bu sonuçlar, MBT'nin hibrit ve keşifçi yapısını doğrulamakta ve S_{mld} gibi ML tabanlı stratejilerin karmaşık rotalama koşullarında sağlamlığını artırdığını göstermiştir. Şekil 103 ve 104'deki TA algoritmasına ait ısı haritası ve çizgi grafiği, S_{mld} başlangıcının tüm test koşullarında en düşük toplam seyahat süresine ulaşarak diğer algoritmalarından üstün olduğunu göstermiştir. Bu durum, ML tabanlı S_{mld} stratejisinin, TA algoritması ile birlikte kullanıldığında daha etkili şekilde değerlendirildiğini ve BT çerçevesindekinden farklı bir davranış sergilediğini ortaya koymuştur. TA içindeki tabu mekanizması, fazla yapılandırılmış başlangıç çözümlerinin olası dezavantajlarını azaltmış; algoritmanın yinelenen yerel döngülerden kaçınmasını ve çözüm uzayında daha umut vadeden bölgelere yönelmesini sağlamıştır. Ayrıca, Şekil 105'deki kutu grafiği, S_{mld} ile başlatılan TA algoritmasının seyahat süresi dağılımında daha kompakt ve düşük varyanslı bir yapı sunduğunu; bunun da rota optimizasyonunda daha tutarlı ve verimli bir performansa işaret ettiğini göstermiştir. Elde edilen sonuçlar, TA algoritmasının veri odaklı başlangıçlardan daha iyi yararlandığını ve rastgele yapıli başlangıçlara kıyasla daha yüksek uyarlanabilirlik sunduğunu ortaya koymuştur.



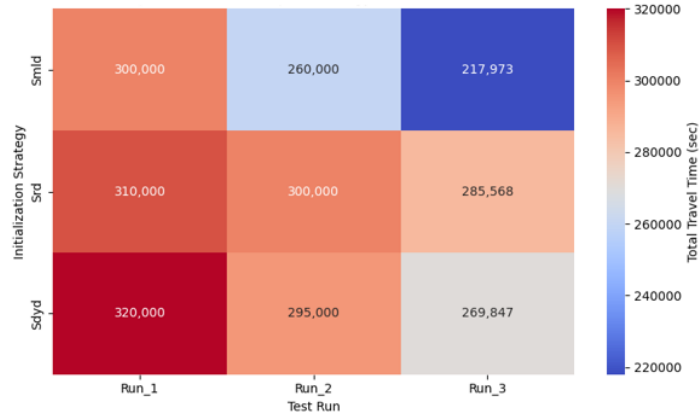
Şekil 97. BT algoritması kullanılarak başlangıç stratejileri ve test koşulları boyunca toplam seyahat süresi ısı haritası (Erdem *et al.* under review b)



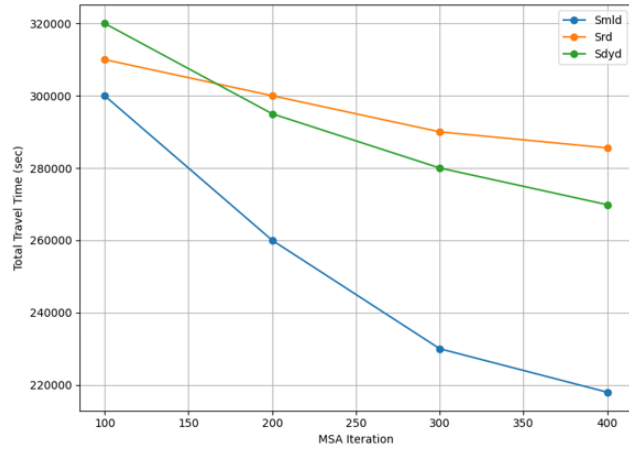
Şekil 98. Farklı başlangıç stratejilerine göre BT algoritmasının iterasyonlar boyunca yakınsama davranışı (Erdem *et al.* under review b)



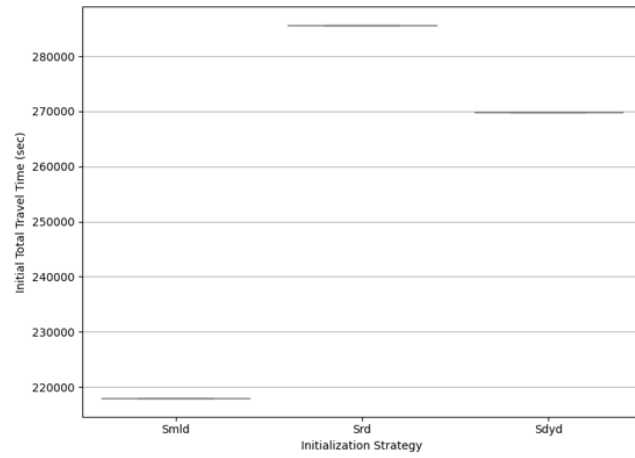
Şekil 99. BT algoritması için başlangıç stratejilerine (S_{ml} , S_{rd} , S_{dyd}) göre kutu grafiği (Erdem *et al.* under review b)



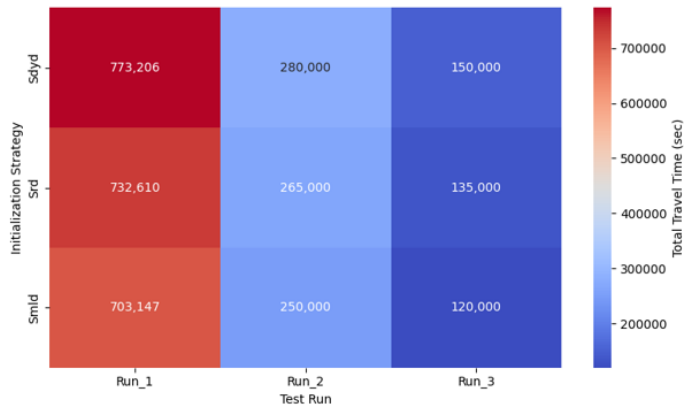
Şekil 100. MBT algoritması kullanılarak başlangıç stratejileri ve test koşulları boyunca toplam seyahat süresi ısı haritası (Erdem *et al.* under review b)



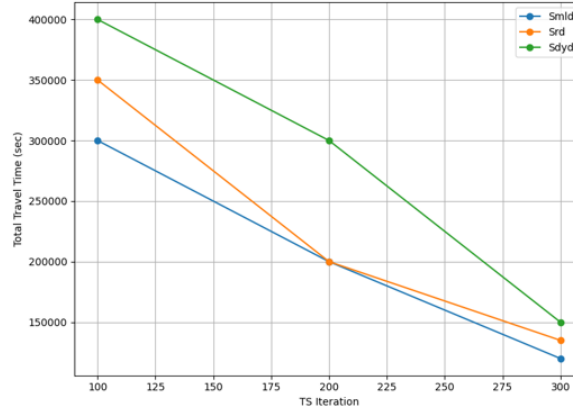
Şekil 101. MBT algoritmasının farklı başlangıç stratejilerine göre iterasyonlar boyunca yakınsama davranışı (Erdem *et al.* under review b)



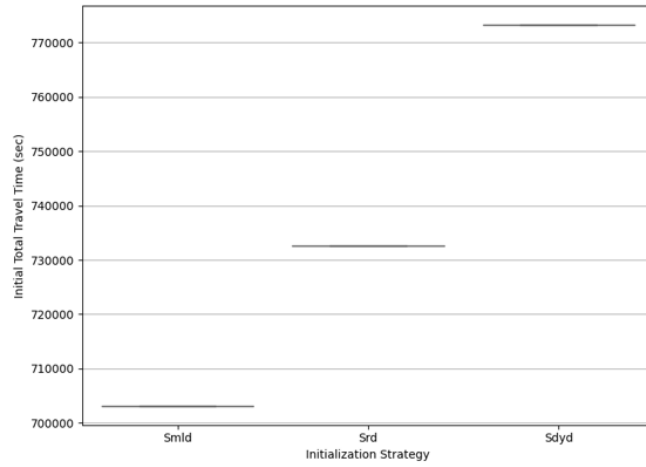
Şekil 102. MBT algoritması için başlangıç stratejilerine (S_{mld} , S_{rd} , S_{dyd}) göre kutu grafiği (Erdem *et al.* under review b)



Şekil 103. TA algoritması kullanılarak başlangıç stratejileri ve test koşulları boyunca toplam seyahat süresi ısı haritası (Erdem *et al.* under review b)

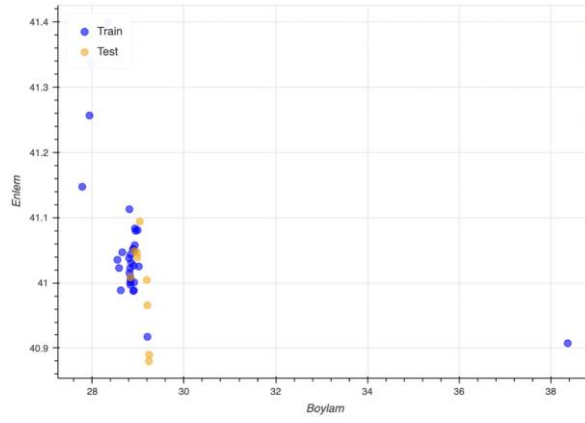


Şekil 104. TA algoritmasının farklı başlangıç stratejilerine göre iterasyonlar boyunca yakınsama davranışı (Erdem *et al.* under review b)



Şekil 105. TA algoritması için başlangıç stratejilerine (S_{mld} , S_{rd} , S_{dyd}) göre kutu grafiği (Erdem *et al.* under review b)

Tezin dördüncü, beşinci ve altıncı aşamasında, veri seti, model performansının dengeli bir şekilde değerlendirilmesi için %70 eğitim ve %30 test oranında bölünmüştür (Şekil 106). Bu oran, modelin genelleme kapasitesini daha dengeli bir şekilde değerlendirebilmek amacıyla tercih edilmiştir. Özellikle %80-20 gibi daha bölmeler, test verisi miktarını azaltarak modelin doğrulama performansını tam olarak yansıtamayabilir. %70-30 oranı ise, hem eğitim için yeterli veri sağlamak hem de modelin gerçek dünyadaki başarımını daha güvenilir bir şekilde ölçmek için yeterli test verisi sunmaktadır.



Şekil 106. Train ve test noktalarının görsel dağılımı

Tezin dördüncü ve beşinci aşamalarında, optimizasyon için kullanılan yedi farklı modelin performansı, rotaların verimliliği ve ödül değerleri dikkate alınarak değerlendirilmiştir. Ayrıca, performans ölçümleri, MAE ve MSE, kayıp, fitness değeri dağılımı, kapasite kullanımı, k-fold bazlı analizler, korelasyon matrisi, hata dağılım şemaları (radar, vb.), fitness değerleri frekans ve yoğunluk (KDE) Dağılım Grafiği ile gerçekleştirilmiştir (Erdem *et al.* under review d).

RL modellerinde doğrudan bir eğitim-test ayrımı olmamasına rağmen, bu hibrit uygulamada train-test bölünmesi bazı belirli nedenlerle yapılmıştır. İlk olarak, modelin yalnızca eğitim verilerine uyum sağlaması değil, farklı müşteri ve talep dağılımlarında da başarılı olması beklenmektedir. Bu amaçla test süreci, modelin genelleme kapasitesini değerlendirmek ve "aşırı öğrenme" yapıp yapmadığını belirlemek amacıyla kullanılmıştır. İkinci olarak, eğitim ortamı simülasyon performansını optimize etmeye odaklansa da, modelin gerçek dünya problemlerinde karşılaşabileceği farklı veri dağılımları üzerindeki performansı test edilerek bu fark analiz edilmiştir. Üçüncü olarak, eğitim ve test ödüllерinin karşılaştırılması yoluyla modelin performansında bir düşüş olup olmadığı incelenmiştir. Son olarak, modelin eğitim sürecinde öğrendiği politikaların test ortamında ne kadar etkili olduğu değerlendirilmiş ve bu yaklaşım, RL tabanlı modellerin eğitim sırasında aşırı öğrenme yapmasını engellemek için kritik bir adım olarak görülmüştür.

BKA ve dikkat tabanlı modellerde ödül kavramı klasik RL modellerindeki gibi doğrudan bir ödül sinyali değil, daha çok fitness değerlerinden veya kaybın tersine çevrilmiş halinden türetilen bir ölçüt olarak kullanılır. BKA ve dikkat tabanlı modellerin temel mantığı, RL tabanlı bir politika öğrenmekten ziyade, çözüm kalitesini artırmak için doğrudan optimizasyon yapmaktır. Yani, BKA ve dikkat modellerinde ödül değeri, doğrudan RL'deki gibi çevresel bir geri bildirim değil, modelin optimizasyon başarısını ölçen bir metrik olarak

kullanılır. Bu nedenle, BKA’da fitness fonksiyonu, dikkat modelinde ise kayıp fonksiyonu ödülün türetilmesinde temel rol oynamıştır (Erdem *et al.* under review d).

1. BKA Modeli için Ödül Açıklaması:

- ✓ Fitness değeri: BKA’nın doğrudan minimize etmeye çalıştığı toplam süredir. Bu nedenle BKA’da ödül değeri, fitness fonksiyonunun negatif hali olarak tanımlanır. Yani, fitness değeri ne kadar düşükse, ödül o kadar yüksek kabul edilir.

2. Dikkat Mekanizması için Ödül Açıklaması:

- ✓ Kayıp fonksiyonu: Modelin tahmin ettiği müşteri sıralamaları ile gerçek sıralamalar arasındaki fark “CrossEntropyLoss” kullanılarak ölçülür.
- ✓ Ödül türetme: Modelin başarısını izlemek için kullanılan ödül değeri, doğrudan kayıp fonksiyonundan türetilir. Kayıp değeri normalize edilerek (maksimum kayıp - mevcut kayıp) şeklinde pozitif bir ödül değeri elde edilir.

Dikkat mekanizması parametre detayları Tablo 33’de verilmiştir (Erdem *et al.* under review d).

Tablo 33. Dikkat Mekanizması Parametre Detayları

Parametre	Aşama-Açıklama
input_dim	Müşteri Verilerinin Girişi: Modelin girdi boyutu olup, müşterilerin konumlarını ve sipariş bilgilerini temsil eder.
hidden_dim (128)	İleri Beslemeli Katman: Encoder ve Decoder katmanlarında kullanılan gizli katman boyutudur. Öğrenilen temsillerin vektör boyutunu belirler.
output_dim	Modelin ürettiği çıktının boyutudur. Burada tahmin edilen müşteri sıralamaları (rota optimizasyonu) belirlenir.
nheads (8)	Çoklu Kafa Dikkat Mekanizması: Çoklu başlı dikkat mekanizmasında kullanılan başlık sayısını belirler. Dikkatin farklı yönlere odaklanmasını sağlar (1).
transformer_encoder	Transformer Encoder katmanlarının kaç tane olduğunu belirler. Modelin derinliğini artırır ve daha karmaşık ilişkileri öğrenmesini sağlar (2).
num_layers(6)	Transformer Encoder katmanlarının kaç tane olduğunu belirler. Modelin derinliğini artırır ve daha karmaşık ilişkileri öğrenmesini sağlar (3).
dropout (0,1)	Modelin aşırı öğrenmesini önlemek için kullanılır.
encoder	Encoder-Müşteri konumlarını ve siparişleri gömme vektörlerine çevirir.
positional_encoding	Pozisyon Kodlaması: Müşteri verilerinin zaman içindeki sıralı ilişkilerini öğrenmeye yardımcı olur.
decoder	Çıktı Katmanı: Rota Tahmini
criterion	Kayıp Fonksiyonu-Modelin kayıp fonksiyonu olup, tahmin edilen müşteri rotaları ile gerçek rotalar arasındaki farkı hesaplar.
optimizer	Model Eğitimi-Modelin ağırlıklarını güncellemek için kullanılan optimizasyon algoritmasıdır.
num_epochs (100)	

3. DTBKA Modeli için Ödül Açıklaması:

- ✓ Kayıp fonksiyonu: DTBKA modelinde, Transformer tabanlı Dikkat modeli ile tahmin edilen müşteri sıralamaları ile gerçek müşteri sıralamaları arasındaki fark, CrossEntropyLoss kullanılarak ölçülür. Bu kayıp fonksiyonu, modelin sıralama doğruluğunu artırmaya yönelik optimizasyon sürecinde kullanılır.
- ✓ Ödül türetme: DTBKA modelinde ödül değeri, doğrudan kayıp fonksiyonundan türetilir. Kayıp değeri negatif alınarak (Ödül = -Loss) pozitif bir ödül elde edilir. Ayrıca, elde edilen ödül değerleri normalize edilerek eğitim ve test performansları arasında karşılaştırma yapılır (Erdem *et al.* under review d).

4. DQN Modeli için Ödül Açıklaması

- ✓ Ödül değeri: DQN modelinde ödül, her adımda alınan aksiyonun sonucuna göre çevreden anlık olarak elde edilir. Model, pozitif ödülleri maksimize ederken ceza değerlerini minimize eder. Modelde dört farklı ceza türü tanımlanmıştır. Aynı müşteriye tekrar gidilmesi durumunda, model -1000 puanlık yüksek bir tekrar ziyaret cezası alır. Eğer bir aracın kapasitesi aşılsa, kapasite aşımı cezası uygulanır ve model -1000 puan ceza alarak süreç sonlandırılır. Araçların belirlenen azami (maksimum) mesafeyi aşması durumunda, model azami mesafe aşımı cezası olarak -100 puanla cezalandırılır. Ayrıca, her müşteri seçiminde müşteri ile depo arasındaki mesafe, negatif bir değer olarak zaman-mesafe maliyeti şeklinde ödül fonksiyonuna yansıtılır.

DQN parametre detayları Tablo 34-36'da verilmiştir (Erdem *et al.* under review d).

Tablo 34. Q-Değerlerinin Güncellenmesi (Bellman Eşitliği)

Parametre	Açıklama
gamma (0,95)	Gelecekteki ödülleri belirler (Bellman güncellemesi için).
batch_size (256)	Mini-batch boyutunu belirler, Bellman güncellemesinde kullanılır.
learning_rate	Öğrenme oranı belirlenir, Bellman eşitliğiyle güncellemeleri etkiler.

Tablo 35. Replay Memory ve Mini-Batch Eğitimi

Parametre	Açıklama
Replay_buffer_size (1000000)	Daha önce yapılan hamleleri saklar, tekrar eğitmek için kullanılır.
Target_update_interval (1000)	Hedef Q-ağı belirli aralıklarla güncellenir.

Tablo 36. Q-Ağının Eğitilmesi ve Nihai Rota Optimizasyonu

Parametre	Açıklama
num_training_epochs (100)	Modelin 100 dönem boyunca eğitilmesini sağlar.
total_timesteps (10000)	Model, 10.000 adım boyunca öğrenir.
training_batch_size (256)	Her batch için 256 veri kullanılır.

5. BME Modeli için Ödül Açıklaması

- ✓ Fitness değeri: Evrimsel algoritmalarda fitness fonksiyonu, bir çözümün kalitesini ölçen temel bileşendir. Bu modelde fitness değeri, araç kapasitesi, zaman maliyeti ve ceza fonksiyonlarının toplamından oluşur. Ödül değeri, bu fitness fonksiyonunun negatif hali olarak tanımlanır (Ödül = -Fitness Value).

BMEO modelindeki Sayısal Parametrelerin Tablo 37’de detayları verilmiştir. Bu modelde üç ana parametre grubu bulunmaktadır. İlk grup, modelin Bulanık Mantık bileşenine ait değişkenleri temsil eder ve toplamda üç adet değişken tanımlanmıştır. İkinci grup, yine Bulanık Mantık sistemine ait üç kuraldan oluşmaktadır. Üçüncü grup ise diferansiyel evrim algoritmasına ait parametreleri kapsamaktadır ve bu grup beş farklı parametreden oluşmaktadır: maksimum iterasyon sayısı, algoritmanın durma kriteri olarak belirlenen eşik değeri, mutasyon faktörü, rekombinasyon oranı ve kullanılan evrim stratejisi. Diferansiyel evrim algoritmasının parametre detayları ise Tablo 38’de sunulmuştur. Algoritmanın maksimum iterasyon sayısı 200 olarak belirlenmiştir. Durma kriteri için kullanılan eşik değeri, çözümün iyileşme oranı baz alınarak 3 olarak tanımlanmıştır. Mutasyon faktörü, çözüm arama sürecinde ağırlık katsayısı olarak görev yapmakta ve (0,5, 1) aralığında bir değer almaktadır. Rekombinasyon oranı, yani çaprazlama sırasında bilgi alışverişi oranı ise 0,7 olarak belirlenmiştir. Modelde tercih edilen evrim stratejisi olarak ise “best1bin” stratejisi kullanılmıştır (Erdem *et al.* under review d).

Tablo 37. BMEO Modelindeki Parametreler

Parametre Grubu	Parametre Sayısı
Bulanık mantık değişkenleri	3
Bulanık mantık kuralları	3
Diferansiyel evrim	5 (max iterasyon sayısı, eşik değeri, mutasyon faktörü, rekombinasyon oranı, kullanılan evrim stratejisi)

Tablo 38. Diferansiyel Evrim Algoritmasının Parametre Detayları

Parametre	Değer
Maksimum iterasyon sayısı	200
Algoritmanın durma kriteri için eşik değeri (çözümün iyileşme oranı)	3
Mutasyon faktörü (ağırlık katsayı ağırlığı)	(0,5,1)
Rekombanasyon (Çaprazlama esnasında bilgi alışverişi oranı)	0,7
Strategy	best1bin

6. GNN-LSTM-RL Modeli için Ödül Açıklaması

- ✓ Ödül değeri: Modelde ödül (ödül) doğrudan zaman-mesafe matrisi üzerinden hesaplanır. Seçilen müşteri ile depo arasındaki mesafe veya süre farkı negatif olarak alınır ve bu değer, toplam maliyeti düşürmek için minimize edilmesi gereken bir faktör olarak değerlendirilir.

GNN-LSTM-RL ve GNN-RL modellerindeki GNN Katmanlarının Sayısal Parametreleri Tablo 39’da detayları verilmiştir. Her bir GCNConv katmanının toplam parametre sayısını hesaplamak için Denklem 25’deki formül kullanılmıştır. Modelde kullanılan iki GCNConv katmanının toplam parametre sayısı 4352’dir (Tablo 39). İlk katman input_dim = 2, hidden_dim = 64, ikinci katman ise input_dim = 64, hidden_dim = 64 şeklindedir.

$$\text{Parametre Sayısı} = (\text{Girdi boyutu} \times \text{Çıktı boyutu}) + \text{Çıktı boyutu} \quad (25)$$

Tablo 39. GNN-LSTM-RL ve GNN-RL Modellerindeki GNN Katmanlarının Sayısal Parametreleri

Katman	input_dim	hidden_dim	Çıktı boyutu	hidden_dim	Parametre sayısı
Birinci GCNConv Katmanı	2	64	X	64	(2x64)+64
İkinci GCNConv Katmanı	64	64	X	64	(64x64)+64
Toplam parametre sayısı: 4352					

GNN-LSTM-RL modelindeki LSTM Katmanlarının Sayısal Parametreleri Tablo 40’da verilmiştir. Modelde kullanılan LSTM katmanlarının toplam parametre sayısı 99328’dir (Tablo 40). LSTM katmanlarında lstm_input_dim = 64 ve lstm_hidden_dim = 128 dir.

Tablo 40. GNN-LSTM-RL modelindeki LSTM Katmanlarının Sayısal Parametreleri

Kapı	input_size	hidden_size	hidden_size	hidden_size	bias	Parametre sayısı
Forget gate	64	128	128	128	2x128	$[(64 \times 128) + (128 \times 128) + (2 \times 128)]$
Input gate	64	128	128	128	2x128	$[(64 \times 128) + (128 \times 128) + (2 \times 128)]$
Cell state update	64	128	128	128	2x128	$[(64 \times 128) + (128 \times 128) + (2 \times 128)]$
Output gate	64	128	128	128	2x128	$[(64 \times 128) + (128 \times 128) + (2 \times 128)]$
Toplam parametre sayısı: 99328						

GNN-LSTM-RL ve GNN-RL modellerindeki PPO Modeli Sayısal Parametreleri Tablo 41’de verilmiştir. Politika yapısı olarak varsayılan çok katmanlı algılayıcı tercih edilmiş ve bu yapıya LSTM tabanlı bir özel özellik çıkarıcı entegre edilmiştir. Bu özellik çıkarıcı, gözlemleri işleyerek özellik uzayına dönüştürmektedir. LSTM giriş boyutu 64, gizli katman boyutu ise 128 olarak belirlenmiştir. PPO algoritması, özel olarak tanımlanan ÇDARPEnv ortamında çalıştırılmıştır. Modelin optimizasyon sürecinde kullanılan varsayılan PPO parametreleri şu şekildedir: Öğrenme oranı 0,0003, adım sayısı (n_steps) 2048, mini-batch boyutu (batch_size) 64 olarak ayarlanmıştır. İndirim faktörü (gamma) 0,99, GAE lambda değeri 0,95’tir. Her bir güncelleme için 10 epoch kullanılmakta, clip_range değeri 0,2 olarak belirlenmiştir. Entropi katsayısı (ent_coef) 0, değer fonksiyonu katsayısı (vf_coef) 0,5 ve maksimum gradyan normu (max_grad_norm) 0,5 olarak ayarlanmıştır. GNN-LSTM-RL modelindeki toplam parametre sayısı 112065’dir. GNN-RL modelindeki toplam parametre sayısı 12737’dir.

Tablo 41. PPO Modeli Parametreleri Sayısal Parametreleri

Bileşen	Parametre	Değer / Açıklama
Politika	MlpPolicy	Varsayılan çok katmanlı algılayıcı
	LSTM Giriş Boyutu	64
	LSTM Gizli Katman Boyutu	128
PPO Varsayılan Parametreleri	learning_rate	3e-4 (0,0003)
	n_steps	2048
	batch_size	64
	gamma (Discount Factor)	0,99
	gae_lambda	0,95
	epoch sayısı	10
	clip_range	0,2
	ent_coef	0,0
	vf_coef	0,5
	max_grad_norm	0,5
GNN-LSTM-RL modelindeki toplam parametre sayısı: $4352 + 99328 + 8385 = 112065$		
GNN-RL modelindeki toplam parametre sayısı: $4352 + 8385 = 12737$		

7. GNN-RL Modeli için Ödül Açıklaması

- ✓ Ödül değeri: Modelde ödül, doğrudan zaman-mesafe matrisi üzerinden hesaplanır. Seçilen müşteri ile depo arasındaki mesafe veya süre farkı negatif olarak alınır.

Bulgular incelendiğinde (Erdem *et al.* under review d);

- ✓ BKA'nın hem eğitim hem de test aşamalarında en düşük performansı sergilediğini göstermiştir. BKA tarafından üretilen rotalar daha uzun ve daha az verimli olup, müşteri talepleri ile araç kapasiteleri arasında dengeli bir çözüm sağlayamamıştır. Eğitim ve test aşamalarında elde edilen ödül değerleri düşük kalmış, bu da modelin sınırlı keşif yeteneklerini ortaya koymuştur. BKA'nın eğitim ve test ödül sonuçları incelendiğinde, grafik boyunca yüksek dalgalanma olduğu görülmüştür. Belirli aralıklarla yüksek ödüller elde edilse de genel kararlılık düşük kalmıştır. Ayrıca, test ödülleri eğitim ödüllerine göre daha fazla dalgalanma göstermekte ve genellikle daha yüksekte seyretmiştir; bu durum genel performans üzerinde olumsuz etkilere yol açabilir (Şekil 115). BKA'nın rotaları mesafe ve zaman açısından daha uzun ve verimsiz olmuş, bu da gerçek değerlerle uyumsuzluğa yol açarak yüksek hata oranlarına neden olmuştur (Şekil 108). BKA hem eğitim hem de test aşamalarında en yüksek MAE ve MSE değerlerini göstermiştir (Şekil 130-136). Modelin K-Fold çapraz doğrulama analizine göre, eğitim fold'larında elde edilen fitness değerleri oldukça dalgalı bir yapı sergilemiştir. Şekilde verilen normalleştirilmiş fitness grafiğinde, bazı fold'larda eğitim ve test değerleri arasında büyük farklar olduğu, üçüncü fold'da ise her iki fitness değerinin neredeyse sıfırlandığı görülmüştür. Bu durum, modelin veri setine istikrarlı şekilde uyum sağlayamadığını ve özellikle bazı senaryolarda başarısız çözümler ürettiğini göstermiştir. Boxplot analizine göre, eğitim fold'larındaki fitness değerlerinin ortanca değeri test fold'una göre daha düşük olup, geniş bir dağılım göstermiştir. Test fold'larındaki değerler daha yüksek ortanca ile dar bir aralıkta yoğunlaşmış, ancak bu durum test verisine fazla uyum sağlandığını düşündürmüştür. Fitness değerlerinin histogramı incelendiğinde, özellikle -24,000 ile -14,000 aralığında fitness değerlerinin tekrar eden frekansta yoğunlaştığı görülmüştür. Bu geniş ve çok düşük aralıklı fitness dağılımı, BKA'nın bazı durumlarda aşırı maliyetli ve verimsiz rotalar ürettiğini göstermiştir. Dağılımın bu kadar keskin ve düşük değerlerde yoğunlaşması, modelin genel olarak kötü performans sergilediğini ve genelleme yeteneğinin zayıf olduğunu ortaya koymuştur. Modelin yük kullanım oranı grafiklerine göre, eğitim ve test aşamalarında %100'e yakın kapasite kullanımı elde edilmiş olsa da bu durum

BKA'nın rota kalitesindeki zayıflığı telafi edememiştir. Tüm araçlar kapasiteyi tam kullanmasına rağmen rotaların toplam süresi ve verimliliği açısından istenilen optimizasyon başarısı sağlanamamıştır (Şekil 122).

- ✓ DQN, BKA'ya göre daha iyi sonuçlar vermiş olsa da, test aşamasında genelleme yeteneği yetersiz kalmış ve beklenenden düşük bir performans sergilemiştir. Eğitim sırasında ödül değerleri artmasına rağmen test aşamasında bu değerler sınırlı kalmıştır. Eğitim ve test ödülleri boyunca dengesiz ve düzensiz bir yapı gözlemlenmiştir; sık sık sıfıra yakın değerlere düşüp tekrar yükselmiştir. Bu durum, modelin kararlı bir şekilde öğrenemediğini ve eğitimin dalgalı ilerlediğini göstermiştir (Şekil 117). DQN modeli takviyeli öğrenme yöntemi ile rotaları optimize etmeye çalışsa da bazı rotalarda gereksiz uzama ve dengesizlik gözlemlenmiştir (Şekil 107). Eğitim sırasında MSE ve MAE değerlerinde azalma görülmesine rağmen, test aşamasındaki yüksek hatalar, modelin gerçek dünya uygulamalarındaki performans düşüşünü ortaya koymuştur (Şekil 130-136). Modelin K-Fold analizinde fitness boxplot sonuçları incelendiğinde, eğitim fold'undaki fitness değerlerinin çok düşük düzeylerde yoğunlaştığı, test fold'undaki değerlerin ise 1,0 seviyesinde sabit kaldığı görülmüştür. Bu durum, eğitim sürecinde modelin bazı senaryolarda kötü çözümler ürettiğini, test setinde ise modelin veri yapısına fazla uyum gösterdiğini düşündürmüştür. Eğitim verisinde fitness değerlerinin düşük kalması, modelin rota optimizasyonunu başarıyla gerçekleştiremediğini göstermiştir. Fitness histogramına bakıldığında, fitness değerlerinin büyük ölçüde -200,000 ile 0 arasında yoğunlaştığı ve tekrar eden belirli aralıklarda sabitlendiği görülmüştür. Bu yapı, modelin büyük çoğunlukla yüksek maliyetli çözümler sunduğunu ve düşük fitness değerlerinde sıkıştığını göstermiştir. Bu da, DQN modelinin hem eğitim hem de test veri setlerinde istikrarlı optimizasyon sağlayamadığına işaret etmiştir. Yük kullanım oranı grafiklerine göre, eğitim ve test aşamalarında araçların %100'e yakın kapasiteyle kullanıldığı görülmüştür (Şekil 125). Bu durum, DQN modelinin kapasite kullanımında başarılı olduğunu gösterse de rota kalitesindeki düşüklük nedeniyle bu başarı genel performansa tam olarak yansımamıştır.
- ✓ Dikkat mekanizması algoritması, diğer modellere en kararlı ve en yüksek ödül değerlerine ulaşmıştır. Eğitim sürecinde oldukça hızlı bir şekilde yüksek ödül değerlerine ulaşılmış (kararlı ve hızlı yakınsama) ve eğitim-test ödülleri arasında oldukça iyi bir uyum sağlanmıştır. Her iki ödül eğrisi benzer bir artış göstererek sabitlenmiştir. Bu durum, modelin kararlı bir öğrenme gerçekleştirdiğini ve güçlü

genelleme yeteneklerine sahip olduğunu göstermiştir (Şekil 114). Kayıp fonksiyonları incelendiğinde ise eğitim ve test kayıpları arasındaki fark oldukça azdır ve ilk 20 epoch içerisinde hızlı bir azalma ile kayıplar sıfıra yaklaşmıştır. Sonraki epochlarda ise kayıplar düşük seviyelerde stabil kalmıştır (Şekil 121). Dikkat mekanizması, hem eğitim hem de test aşamalarında daha kısa ve optimize edilmiş rotalar üretmiş, müşteri taleplerini araç kapasiteleriyle etkili bir şekilde dengelemiştir. Müşteri grupları arasında belirgin ayrımlar yaparak rotaları optimize etmiştir; ancak eğitim ve test setleri arasında bazı farklar gözlenmiştir (Şekil 109). Model, MAE ve MSE bakımından önemli iyileşmeler sağlamış, müşteri talepleri ile araç kapasiteleri arasındaki ilişkileri başarıyla öğrenmiş ve son derece optimize rotalar üretmiştir (Şekil 130-136). Modelin K-Fold analizinde fitness boxplot sonuçları incelendiğinde, eğitim fold'undaki fitness değerlerinin oldukça düşük seviyelerde ve dar bir aralıkta yoğunlaştığı, test fold'undaki fitness değerlerinin ise daha geniş bir aralıkta ve genellikle 0,7 ile 1 arasında toplandığı görülmüştür. Bu durum, modelin test verisi üzerinde daha iyi genelleme yapabildiğini ve daha kararlı çözümler sunduğunu göstermiştir. Eğitim ve test fitness eğrileri birlikte incelendiğinde, özellikle bazı fold'larda (örneğin 2. ve 5. fold'da), eğitim verisinin test verisine kıyasla daha yüksek uygunluk değerleri ürettiği gözlemlenmiştir. Bu, dikkat mekanizmasının eğitim verisine uyum sağlama konusunda etkili olduğunu, ancak test verisi üzerindeki genelleme performansının sınırlı kaldığını göstermiştir. Fitness histogramı incelendiğinde, fitness değerlerinin büyük çoğunluğunun 0,9 ile 1 aralığında yoğunlaştığı ve oldukça sık tekrarlandığı görülmüştür. Bu, modelin çok sayıda optimal veya optimal yakın çözümler üretebildiğini ortaya koymuştur. Fitness dağılımının bu denli dar ve yüksek olması, modelin öğrenme sürecinde belirli bir kararlılıkla rotaları optimize ettiğini ve müşteri taleplerine uygun çözümler ürettiğini göstermiştir. Dış veri noktası (outlier) içermeyen bu dağılım, dikkat mekanizmasının öngörülebilir ve güvenilir çıktılar sunduğunu kanıtlamıştır. Yük kullanım grafikleri incelendiğinde, hem eğitim hem de test setinde araç kapasitesi kullanım oranının %100 olduğu ve araçların tamamının etkili bir şekilde kullanıldığı görülmüştür (Şekil 123). Bu da modelin sadece rota optimizasyonunu değil, aynı zamanda araçların yük kapasitesini de maksimum düzeyde değerlendirdiğini ortaya koymuştur.

- ✓ DTBKA algoritması, her iki yaklaşımın güçlü yönlerini (BKA'nın keşif kapasitesini ve Dikkat mekanizmasının öğrenme gücü), birleştirerek Dikkat mekanizması algoritmasından sonra en kararlı ve yüksek ödül değerine sahip

algoritma olarak öne çıkmaktadır. Eğitim sürecinde ödöl değerleri düzenli bir artışla yüksek seviyelere ulaşmış ve eğitim-test uyumu sağlanmıştır (Şekil 116). Eğitim ve test setlerinde oluşturulan rotalar genellikle kısa ve belirgin görünmüştür. DTBKA, hem kısa hem de dengeli rotalar oluşturmuştur. Şekil 135-136 incelendiğinde, en düşük hata oranlarına sahip model DTBKA olmuştur. DTBKA, eğitim sürecinde kayıp fonksiyonunu hızla düşürmüş ve test kayıplarıyla birlikte düşük seviyelerde sabitlenmiştir. Bu durum, modelin hem öğrenme hem de genelleme başarısını doğrulamıştır. Dikkat mekanizmasına kıyasla ilk epochlarda kayıp değerleri biraz daha yüksek olsa da sonraki aşamalarda benzer düzeylere inmiştir (Şekil 121). Hem eğitim hem de test süreçlerinde düşük MAE ve MSE değerleriyle en yüksek doğruluğa ulaşmıştır (Şekil 130-136). Modelin K-Fold çapraz doğrulama sonuçları incelendiğinde, eğitim fold'undaki fitness değerlerinin düşük seviyelerde ve dar bir aralıkta yoğunlaştığı, test fold'undaki fitness değerlerinin ise 0,6 ile 1 aralığında kararlı bir şekilde dağıldığı görülmüştür. Özellikle 4. fold'da, test fitness değerlerinin eğitim değerlerinin üzerinde olduğu ve modelin bazı durumlarda daha iyi genelleme yaptığı gözlemlenmiştir. Test verilerindeki boxplot analizinde değerlerin istikrarlı olduğu ve genelleme başarısının dikkat çekici düzeyde yüksek olduğu anlaşılmıştır. Eğitim ve test fitness eğrileri birlikte değerlendirildiğinde, test fold'larının eğitim fold'larına kıyasla bazı durumlarda daha iyi sonuçlar sunduğu görülmüştür. Bu, modelin gerçek veri senaryolarında başarılı çözümler sunduğunu göstermiştir. Histogramdaki dağılıma göre fitness değerleri, -5 ile -4 aralığında tekrarlayan yüksek frekanslara sahiptir. Değerler belirli noktalarda yoğunlaşmakla birlikte, dağılım tamamen dar bir aralıkta sıkışmamış, ayrık ve çok modlu (çok tepe noktalı) bir yapı sergilemiştir. Bu durum, modelin belirli senaryolarda benzer kalitede çözümler üretme eğiliminde olduğunu, ancak tam anlamıyla dar ve homojen bir dağılım göstermediğini ortaya koyar. Bu dağılım, DTBKA modelinin genellikle benzer kalitede çözümler sunduğunu ve farklı veri setlerine karşı tutarlı performans sergileyebileceğini göstermiştir. Outlier'ların bulunmaması, modelin istikrarlı bir şekilde optimizasyon yaptığını kanıtlamıştır. Geniş bir aralığa yayılmayan bu dağılım, modelin hem müşteri taleplerini hem de araç kapasitelerini etkin bir şekilde optimize ettiğini ve rota planlamasında istikrar sağladığını doğrulamıştır. Yük kullanımına ait grafikler incelendiğinde, eğitim ve test setlerinde araç kapasitelerinin %100 kullanıldığı görülmüştür. Bu, modelin sadece rotaları değil, aynı zamanda kaynak kullanımını da optimize ettiğini ve kapasite

planlamasında etkin çalıştığını göstermiştir. Her iki süreçte de kullanılmayan kapasitenin olmaması, yüksek verimliliği desteklemektedir (Şekil 124).

- ✓ GNN-LSTM-RL algoritmasında uygulanan K-Fold çapraz doğrulama sonuçları, modelin eğitim ve test süreçleri arasındaki performans farklarını ortaya koymuştur. Eğitim sürecinde fitness değerleri genel olarak artış eğiliminde olsa da, test sürecinde gözlenen dalgalanmalar, modelin eğitim verisinde iyi performans sergilerken test verisinde genelleme kapasitesinin azaldığını göstermiştir. K-Fold Fitness Box Plot sonuçları incelendiğinde, eğitim fold'undaki ortalama fitness değeri yaklaşık 0,6 seviyesindedir ve fitness değerleri 0 ile 1 arasında geniş bir aralığa yayılmıştır. Boxplot kutusunun büyük olması, modelin farklı senaryolara karşı değişken sonuçlar üretebildiğini göstermiştir. Test fold'undaki ortalama fitness değeri yaklaşık 0,5 olup, bu değer eğitim fold'una göre daha düşüktür. Test verisindeki fitness değerleri 0,2 ile 1 arasında daha dar bir aralıkta toplanmış ve herhangi bir outlier bulunmamıştır. Bu durum, modelin test verileri üzerinde sınırlı genelleme kapasitesine sahip olduğunu göstermiştir. Fitness değerlerinin frekans dağılımı incelendiğinde, değerlerin geniş bir aralıkta dağılması dikkat çekicidir; özellikle -2000 ile 0 aralığında yüksek frekansa sahip değerler gözlemlenmiştir. Bununla birlikte, -10,000 gibi çok düşük fitness değerlerine sahip az sayıda örnek bulunması, modelin bazı durumlarda kötü sonuçlar ürettiğini göstermiştir. Genel olarak dağılım, modelin çoğunlukla optimal veya yakın-optimal çözümler sunduğunu, ancak bazı veri noktalarında stabilite sorunları yaşadığını ortaya koymuştur. Bu durum modelin genelleme kapasitesinin sınırlı olduğunu ve farklı veri setleri üzerinde tutarlı sonuçlar elde edebilmesi için iyileştirmeye ihtiyaç duyduğunu göstermiştir. Eğitim ve test aşamalarında tüm araçların %100 kapasiteyle kullanıldığı görülmektedir. Bu durum, modelin araç yüklemelerini oldukça dengeli bir şekilde gerçekleştirdiğini ve araç kullanımında etkinlik sağladığını ortaya koymaktadır (Şekil 126). GNN-LSTM-RL modelinin eğitim sürecinde ödül değerlerinde zamanla bir artış eğilimi bulunsa da, bu artış istikrarsız ve yüksek dalgalanmalıdır. Test süreci ise oldukça düzensiz bir yapı sergilemiş; farklı adımlarda ödüller büyük farklılıklar göstermiştir. Bu durum, modelin eğitim verisine tam olarak uyum sağlayamadığını ve test verisi üzerinde düşük genelleme kabiliyetine sahip olduğunu göstermektedir. Hem eğitim hem test grafikleri, modelin kararsız öğrenme dinamikleri ile çalıştığını ve stabilite eksikliği yaşadığını ortaya koymaktadır (Şekil 118). Eğitimdeki rota sonuçları oldukça çeşitlilik göstermekte ve düzensizlik içermektedir; bu da modelin tüm müşteri noktalarını

etkili bir biçimde öğrenemediğini göstermiştir. Test setindeki rotalar daha belirgin olsa da hâlen bazı karmaşıklıklar barındırmakta ve modelin genelleme kapasitesinin kısmen iyi ancak geliştirilmesi gerektiğini göstermiştir (Şekil 111). Eğitim sürecinde elde edilen kayıp değerlerinin grafik üzerinde sabit ve neredeyse sıfır seviyesinde olması, modelin çok düşük hata ile çalıştığını ve fazla iyileştirme gerektirmediğini göstermektedir. Bu durum, eğitim verisi ile modelin uyumunun yüksek olduğunu göstermektedir. Eğitim doğruluğu ise dalgalanmalara rağmen genel olarak 0,8 ile 1 arasında seyretmiş ve modelin eğitim verisinde iyi performans gösterdiğini ortaya koymuştur. Test sürecinde ise kayıp değerlerinde 0 ile 1 arasında ani dalgalanmalar görülmüş ve bu durum modelin test verisinde tutarlı performans sergilemekte zorlandığını ve genelleme yeteneğinin düşük olabileceğini göstermiştir. Benzer şekilde, test doğruluğunda da dalgalanmalar gözlemlenmiş, bazı adımlarda yüksek doğruluk elde edilse de sık sık düşüşler yaşanmıştır (Şekil 126). Görselde yer alan GNN ve LSTM tabanlı modelin eğitim sonuçları, rotaların karmaşık ve düzensiz olduğunu, araç rotalarının birbiriyle kesiştiğini ve modelin bazı müşteri noktalarını gereksiz yere tekrarladığını göstermiştir. Test sonuçları ise daha düzenli görünmekle birlikte bazı rotalarda hâlen karmaşıklık ve dengesizliklerin sürdüğünü ortaya koymuştur (Şekil 111).

- ✓ GNN-RL algoritmasında uygulanan performans analizleri, modelin eğitim ve test aşamalarında gösterdiği genel başarımı ortaya koymaktadır. K-Fold sonuçlarına göre, eğitim sürecinde bazı fold'larda fitness değerleri artış gösterirken, bazı fold'larda dalgalanmalar gözlemlenmiştir. Bu dalgalanmalar, modelin eğitim sürecinde istikrarlı bir öğrenme gerçekleştiremediğini ve test verisinde genelleme kapasitesinin sınırlı olabileceğini göstermiştir. K-Fold Fitness Box Plot analizine göre, eğitim fold'larındaki ortalama fitness değeri yaklaşık 0,4 olup, bu değer GNN-LSTM-RL modeline kıyasla daha düşüktür. Eğitim verisinde fitness değerleri 0 ile 1 arasında dağılmış olsa da, büyük çoğunluğu daha düşük değerlerde yoğunlaşmıştır. Kutu boyutunun dar olması, modelin eğitim verisinde tutarlı çözümler üretebildiğini göstermektedir. Test fold'larında ortalama fitness değeri yaklaşık 0,5 olup, bu da eğitim setine göre bir miktar daha iyi genelleme başarısı olduğunu işaret eder. Bununla birlikte, test setinde 0 civarında bir outlier değeri gözlemlenmiştir; bu, bazı durumlarda modelin düşük kaliteli çözümler de üretebildiğine işaret eder. Genel olarak, fitness değerlerinin dar bir aralıkta ve 0,4–0,9 arasında yoğunlaşması, modelin müşteri taleplerini ve araç rotalarını kararlı ve tahmin edilebilir şekilde optimize edebildiğini göstermektedir. Eğitim ve test

sonuçları birbirine yakın olup, genelleme yeteneği güçlü bir model ortaya koyulmuştur. Ortalama fitness değerlerine bakıldığında, GNN-RL modeli hem eğitim hem de test fold'larında daha düşük ortalama fitness değerine sahiptir. Bu durum, zamanın minimize edilmesinin amaçlandığı problemler açısından daha iyi bir performans sunduğunu göstermiştir. Stabilité açısından GNN-RL modeli, özellikle eğitim fold'unda daha dar bir dağılıma sahiptir ve bu durum modelin daha stabil sonuçlar verdiğini göstermiştir. Fitness değerlerinin frekans dağılımı incelendiğinde, bu değerlerin genellikle -2000 ile 0 aralığında yoğunlaştığı, daha düşük fitness değerlerinin ise -6000 civarında az sayıda ve düşük frekansla gözlemlenmiştir. Bu dağılım, modelin genel olarak tutarlı ve iyi çözümler sunduğunu, sadece az sayıda düşük performanslı çözüm ürettiğini ortaya koymuştur. Fitness değerlerinin büyük kısmı düşük seviyelerde yoğunlaşmıştır; bu da modelin genellikle benzer kalite seviyesinde çözümler ürettiğini göstermektedir. Ancak bu değerler optimalden uzak olduğu için, modelin genel çözüm kalitesi düşüktür. Dağılımın dar olması ise modelin nispeten tutarlı ama sınırlı performans sunduğunu göstermiştir. Bu durumun veri setindeki karmaşıklıklar ya da zorlu müşteri taleplerinden kaynaklanabileceği düşünülmüştür. Modelin yük kullanımı incelendiğinde, kapasite kullanım oranının %100'e yakın olması, araçların kapasitelerinin verimli bir şekilde kullanıldığını ve optimizasyon sürecinin başarılı bir şekilde gerçekleştiğini göstermiştir (Şekil 127). Eğitim-test ödül değişim grafiklerine göre, eğitim sürecinde ödül değerleri genel olarak artış göstermiş, ancak tam anlamıyla sabitlenmeden önce dalgalı bir yapı sergilemiştir. Test sürecindeki ödül eğrisi ise stabil olmaktan uzaktır; bu durum modelin test verisi üzerinde sınırlı genelleme yeteneğine sahip olduğunu göstermektedir (Şekil 119). Ancak fitness histogramı ve araç kullanım analizleri dikkate alındığında, GNN-RL modeli GNN-LSTM-RL'ye kıyasla daha düzenli ve tutarlı rota çözümleri üretmiş, bu da bazı veri senaryolarında modelin başarılı çıktılar sunduğunu göstermiştir (Şekil 112). Bu da modelin genelleme kapasitesinin iyi olduğuna işaret etmiştir. Eğitim süreci boyunca kayıp grafiği incelendiğinde, kayıp değerinin sabit ve neredeyse sıfır seviyesinde olduğu görülmüştür. Bu durum modelin başlangıçtan itibaren düşük hata ile çalıştığını ve fazla iyileştirme gereksinimi olmadığını göstermiştir (Şekil 127). Bu aynı zamanda eğitim verisinin model tarafından kolaylıkla öğrenildiğini ya da model ile veri uyumunun yüksek olduğunu da göstermiştir. Doğruluk (accuracy) grafiği ise dalgalı bir seyir izlese de genel olarak %60 ile %100 arasında seyretmiş ve ortalamada yüksek doğruluk değerine

ulaşmıştır (Şekil 127). GNN-RL modelinde eğitim doğruluğu daha tutarlı seyretmiş ve GNN-LSTM-RL modeline göre daha az dalgalanma göstermiştir. Test sürecinde ise kayıp değerleri dalgalanma göstermeye devam etmiştir. Bu durum, modelin test verisi üzerinde kararlı bir hata oranı sağlayamadığını göstermiştir (Şekil 127). GNN-RL modelinde genel kayıp seviyeleri, GNN-LSTM-RL modeline kıyasla daha düşük ve kontrollü seyretmiştir. Ancak doğruluk grafikleri incelendiğinde, test aşamasında doğruluk seviyeleri zaman zaman keskin düşüşler göstermiş ve genel olarak istikrarsız bir yapı sergilemiştir. Bu durum, modelin bazı veri senaryolarında yeterince genelleme yapamadığını göstermektedir. GNN için çizdirilen loss-accuracy grafiği incelendiğinde, eğitim süreci boyunca kayıp değerlerinin hızla azaldığı ve modelin başarılı bir şekilde öğrendiği anlaşılmıştır. GNN doğruluk grafiğinde ise doğruluğun sabit kaldığı ve 1 seviyesine ulaştığı görülmüştür (Şekil 127). Bu durum, modelin test verisini iyi şekilde genelleyeabildiğini ve doğruluk eğrisinde dalgalanma yaşanmadığını göstermiştir. GNN-RL ve sadece GNN ile çizilen loss-accuracy (kayıp-doğruluk) grafiklerinin karşılaştırılmasında ise, GNN-RL grafiklerinde daha karmaşık bir yapı gözlemlenmiş ve özellikle eğitim sürecinde kayıp ile doğruluk eğrilerinde daha fazla dalgalanma yaşanmıştır. Bu dalgalanmaların, RL'nin ödül tabanlı optimizasyon yaklaşımına ve uzun vadeli öğrenme hedeflerine odaklanmasına bağlı olduğu söylenebilir. GNN-RL modelinde doğruluk ve kayıp eğrileri, GNN modeline kıyasla daha dalgalı ve kararsız bir seyir izlemiştir. Özellikle test doğruluk eğrisinde gözlenen ani dalgalanmalar, modelin genelleme kapasitesinin sınırlı olduğunu göstermiştir. Dolayısıyla GNN-RL, genel optimizasyon başarımı açısından GNN modeline göre daha düşük ve istikrarsız bir performans sergilemiştir. Sadece GNN ile elde edilen grafiklerin avantajları; GNN'nin tekil bileşen katkısını analiz etmek, RL ile entegrasyonun etkisini gözlemlemek ve hiperparametre ayarları açısından modelin optimize edilmesi için fırsatlar yaratması bulunmaktadır. GNN'nin tek başına yüksek doğruluk ve düşük kayıp üretmesi, modelin yapılandırılmasında temel bileşen olarak güçlü bir performansa sahip olduğunu ortaya koymuştur (Şekil 126).

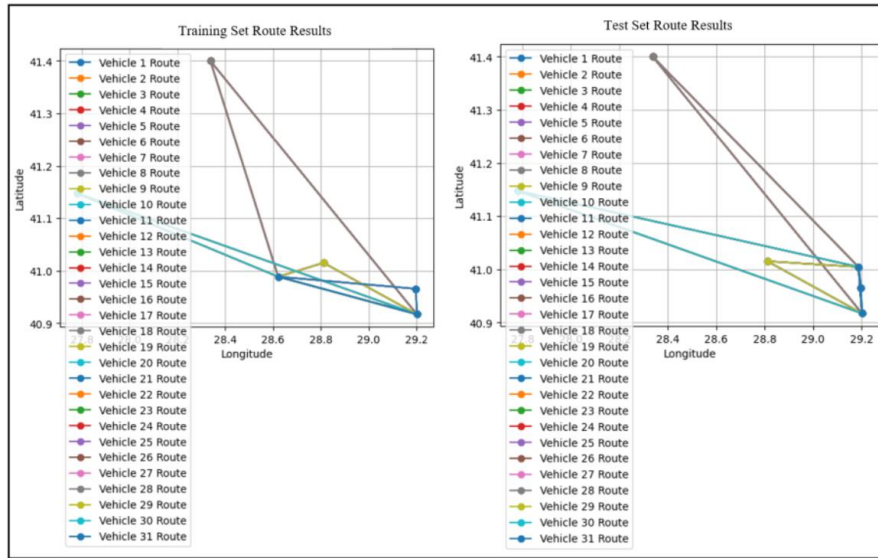
- ✓ BME modelinde hem eğitim hem de test verileri için kümeleme sonuçları analiz edilmiştir. Eğitim verisinde müşteri konumları belirli bölgelere dağılmış ve beş farklı küme oluşturulmuştur. Bu kümeler arasında özellikle kırmızı ve turuncu kümelerin yoğun müşteri içeriğiyle öne çıktığı görülmektedir. Küme dağılımı, araçların belirli alanlara yoğunlaşarak rotalarını daha verimli bir şekilde planlamasına olanak tanımaktadır. Bazı kümelerin sınırlarının birbirine yakın

olması, bulanık kümeleme yöntemiyle esnek müşteri atamaları yapılmasını ve dolayısıyla rotaların daha iyi optimize edilmesini sağlamıştır. Test verisinde de benzer şekilde beş farklı küme oluşturulmuş; ancak müşteri konumlarının daha geniş bir alana yayılması ve kümelerin birbirinden uzak olması, araçların daha uzun mesafeler kat etmesini gerektirmiş ve bu durum test verisinde daha zorlu optimizasyon problemlerine yol açmıştır (Şekil 128). K-Fold sonuçları incelendiğinde, fitness değerlerinin büyük çoğunluğunun dar bir aralıkta toplandığı ve dağılımın oldukça sabit olduğu görülmüştür. Bu durum, modelin müşteri talepleri ile araç kapasiteleri arasında tutarlı ve istikrarlı optimizasyon sağladığını, ayrıca farklı veri bölümlerinde güvenilir çözümler üretebildiğini göstermiştir. Fitness Box Plot çıktılarında belirgin sapmaların bulunmaması, modelin stabilite düzeyinin yüksek olduğunu ortaya koymuştur. Fitness değerlerinin frekans dağılımı analiz edildiğinde, 0-100 aralığında yoğunluk gözlenmiş, yüksek fitness değerlerinin (800-1000) ise nadiren ortaya çıktığı görülmüştür. Bu dağılım, modelin genellikle düşük fitness değerleri üretmesi sayesinde iyi çözümler sunduğunu ve sadece bazı istisnai durumlarda yüksek maliyetli rotalar oluşturduğunu göstermektedir. Genel olarak düşük seviyelerde yoğunlaşan fitness değerleri, modelin çoğunlukla optimal veya optimal çözüme yakın sonuçlar ürettiğini desteklemekte; az sayıdaki yüksek değer ise veri setindeki karmaşıklık veya müşteri taleplerinden kaynaklı zorluklara işaret etmektedir. Araç yük kullanım oranları hem eğitim hem de test setinde %100'e yakın gerçekleşmiş olup, tüm araçların tam kapasiteyle kullanıldığı gözlemlenmiştir. Bu durum, modelin araç yükleme sürecini etkin şekilde optimize ettiğini ve taşıma kapasitesini maksimum seviyede değerlendirdiğini göstermiştir (Şekil 129). Eğitim sürecindeki ödül eğrisi, bazı adımlarda keskin artışlar gösterdikten sonra sabitlenmiş, bu da modelin belirli dönemlerde hızlı öğrenme gerçekleştirdiğini ve ardından stabil performans sergilediğini ortaya koymuştur. Test sürecinde ise ödül eğrisi başlangıçta hızlı adaptasyon göstermiş ancak kısa süre sonra sabit bir seviyeye durmuştur, bu durum modelin test verisine uyum sağlama kapasitesinin sınırlı olduğunu göstermiştir (Şekil 120). Eğitim sürecine ait kayıp (loss) grafiği, modelin öğrenme sürecinde tutarsızlık yaşadığını ortaya koymuştur. Başlangıçta yüksek olan kayıp değeri bazı jenerasyonlarda sıfıra inince de tekrar yükselmiş, benzer şekilde test sürecinde de belirgin dalgalanmalar gözlenmiştir. Bu durum modelin genelleme kapasitesinin sınırlı olduğunu ve farklı veri setlerinde istikrarlı sonuçlar elde etmekte zorlandığını göstermiştir. Şekil 129'daki doğruluk grafiklerine göre, hem eğitim hem de test

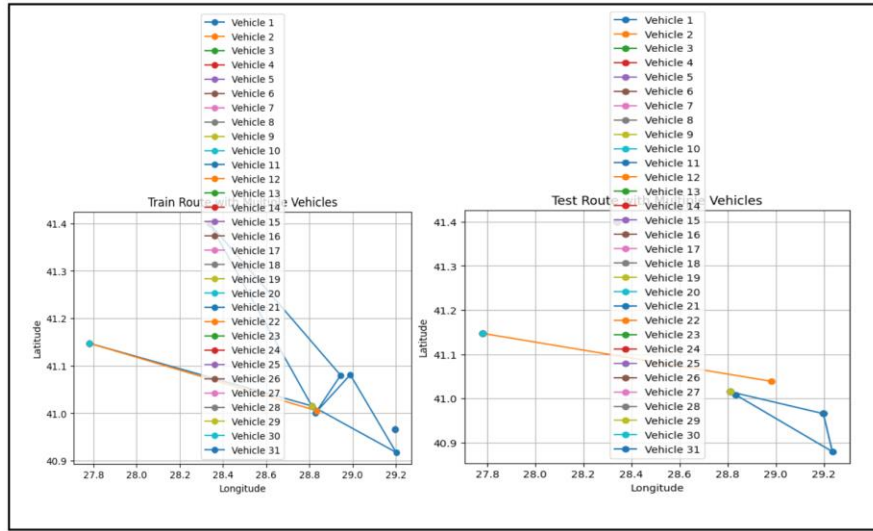
doğruluğu genellikle sıfıra yakın değerlerde seyretmiş; yalnızca bazı epoch'larda ani ve kısa süreli sıçramalar gözlemlenmiştir. Ancak bu doğruluk artışları sürdürülebilir olmamış ve model doğruluğu tekrar düşük seviyelere gerilemiştir. Bu durum, modelin öğrenme sürecinde kararlılık sağlayamadığını ve özellikle test verisi üzerinde genelleme yeteneğinin zayıf kaldığını ortaya koymuştur.

Sonuçlar, DTBKA modelinin hem öğrenme başarısı hem de genelleme yeteneği açısından en üstün performansı sergilediğini göstermiştir. Eğitim sürecinde hızlı yakınsama ile yüksek ödül değerlerine ulaşan DTBKA, test aşamasında da bu başarıyı sürdürmüştür. MAE ve MSE değerleri açısından tüm modeller arasında en düşük hata oranlarına ulaşarak yüksek doğruluk sağlamıştır. Grafiklerdeki hata dağılımları, DTBKA'nın hem eğitim hem de test aşamalarında kararlı ve tutarlı çıktılar verdiğini doğrulamıştır. Bu model, BKA'nın keşif kapasitesi ile dikkat tabanlı modelin güçlü öğrenme becerisini birleştirerek kısa ve dengeli rotalar oluşturmuş, genelleme performansını en üst düzeye taşımıştır. Dikkat mekanizması, DTBKA'nın ardından en iyi performansı sergileyen ikinci model olarak öne çıkmıştır. Eğitim ve test sürecinde ödül eğrileri birbirine oldukça yakın seyretmiş, kayıplar hızla düşerek düşük seviyelerde sabitlenmiştir. MAE ve MSE değerleri bakımından önemli bir iyileşme sağlamış ve müşteri talepleri ile araç kapasiteleri arasında etkin denge kurarak optimize rotalar oluşturmuştur. Scatter ve heatmap grafiklerinde bu başarıyı net şekilde gözlenmiştir. Bununla birlikte, DTBKA kadar güçlü olmasa da geniş veri setlerinde güvenilir bir çözüm sunabileceğini göstermiştir. DQN modeli, eğitim sürecinde makul bir ödül artışı göstermesine rağmen test aşamasında belirgin dalgalanmalar ve düşük genelleme başarısı sergilemiştir. MAE ve MSE değerleri diğer modellere göre daha yüksek kalmış, özellikle test sırasında bu hataların arttığı gözlemlenmiştir (Şekil 136). Scatter plot ve boxplot analizlerinde DQN'nin genellikle yüksek hata ürettiği, sadece bazı durumlarda tatmin edici çözümler sunduğu ortaya çıkmıştır. Bu durum, modelin tutarsızlık yaşadığını ve aşırı öğrenme etkilerinin görüldüğünü göstermektedir. BKA modeli, düşük performans sergileyen algoritma olmuştur. Eğitim ve test sürecinde ödül değerlerinde belirgin dalgalanmalar, yüksek MAE-MSE hataları ve düşük doğruluk oranları gözlemlenmiştir. Özellikle grafiklerde yer alan performans trendleri, BKA'nın genel kararsızlık içinde olduğunu, bazı zamanlarda yüksek hatalı çıktılar ürettiğini göstermiştir. Sankey ve dağılım analizlerinde de BKA'nın hem eğitim hem test sürecinde yüksek hata katkısına sahip olduğu görülmüştür. Bu modelin keşif kapasitesi güçlü olsa da, genel performansı zayıf kalmıştır. GNN-RL modeli ise güçlü genelleme yeteneği ve tutarlı performansı ile öne çıkmıştır. Eğitim ve test süreçlerinde düşük kayıp, yüksek doğruluk ve dengeli ödül eğrileri ile kararlı sonuçlar üretmiştir. K-Fold analizinde de stabil fitness değerleri ve az sayıda outlier ile dikkat çekmiştir. Bu model, geniş ölçekli ve karmaşık problemler için oldukça uygundur ve

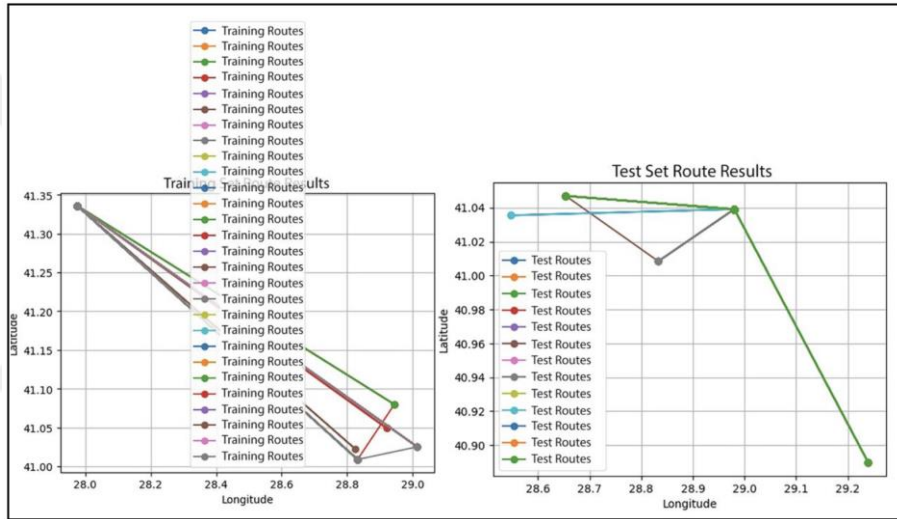
veri setinde tutarlı çözümler sunmuştur. GNN-LSTM-RL modeli, geçmiş müşteri taleplerini öğrenmede başarılı olmuş; ancak eğitim sürecinde elde ettiği olumlu sonuçlara rağmen test aşamasında dalgalanmalar sergilemiştir. MAE ve MSE değerleri zaman zaman yükselmiş, bu durum modelin genelleme kapasitesinin sınırlı olduğunu göstermiştir. Bu modelin rotaları düzensiz ve karmaşık olma eğilimindedir ve her zaman istikrarlı bir öğrenme süreci sunmamıştır. BME modeli, düşük hesaplama maliyeti ve stabil yapısı ile özellikle küçük ve orta ölçekli problemler için etkili bir alternatif sunmuştur. Eğitim ve test kayıplarında gözlenen sabitlik, aşırı öğrenme veya underfitting problemlerinin yaşanmadığını göstermiştir. Ancak, hataların diğer modellere kıyasla biraz daha yüksek olması bu modelin özellikle karmaşık senaryolarda sınırlı kalabileceğini düşündürmüştür. Sonuç olarak, DTBKA modeli zaman ve mesafe açısından en verimli rotaları oluşturmuş ve hata oranlarını en aza indirerek en güçlü çözüm olarak öne çıkmıştır. Dikkat mekanizması ve GNN-RL modelleri, özellikle genelleme kapasitesi ve doğruluk açısından dikkate değer başarı göstermiştir. DQN ve BKA modelleri ise test performansı açısından zayıf kalmış, tutarsızlık ve yüksek hata üretme eğiliminde olmuştur (Şekil 130-137). GNN-LSTM-RL modeli veri geçmişine duyarlı yapı sunarken stabilite açısından sınırlıdır. BME modeli ise düşük kaynaklı uygulamalarda güvenilir bir çözüm sağlayabilir. Tüm bu bulgular, meta-sezgisel ve DL tabanlı yaklaşımların birleşiminin ÇDARP gibi karmaşık problemler için etkili çözümler üretebileceğini ortaya koymaktadır (Erdem *et al.* under review d).



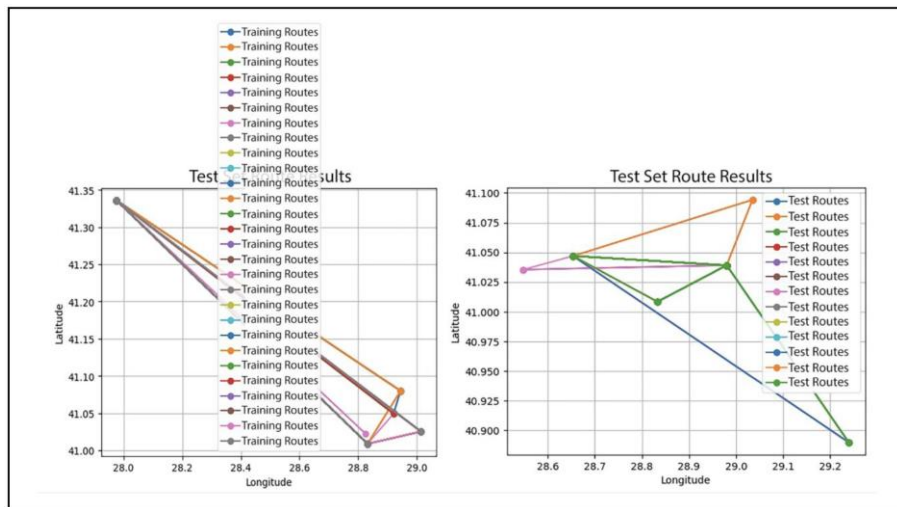
Şekil 107. DQN algoritması için rota sonuçları (Erdem *et al.* under review d)



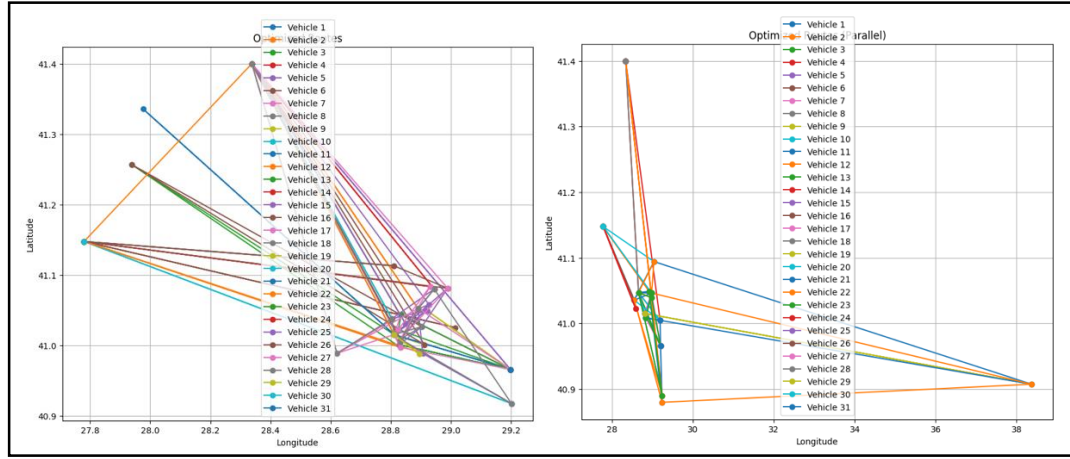
Şekil 108. BKA algoritması için rota sonuçları (Erdem *et al.* under review d)



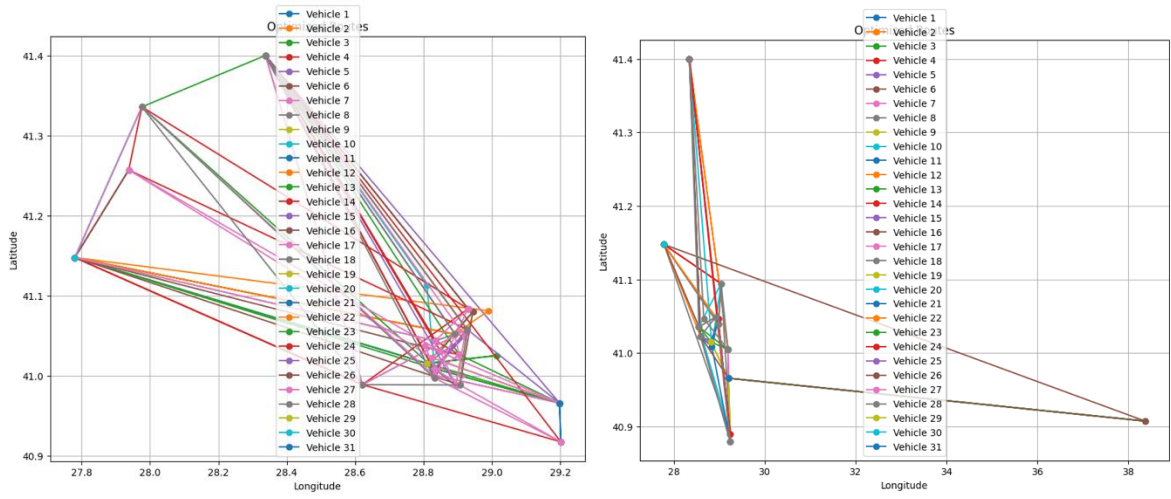
Şekil 109. Dikkat mekanizması için rota sonuçları (Erdem *et al.* under review d)



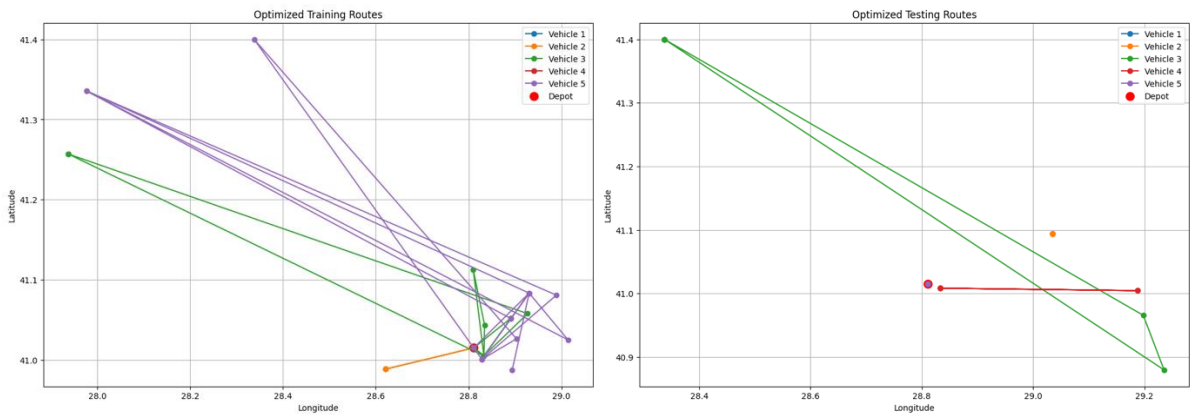
Şekil 110. DTBKA algoritması için rota sonuçları (Erdem *et al.* under review d)



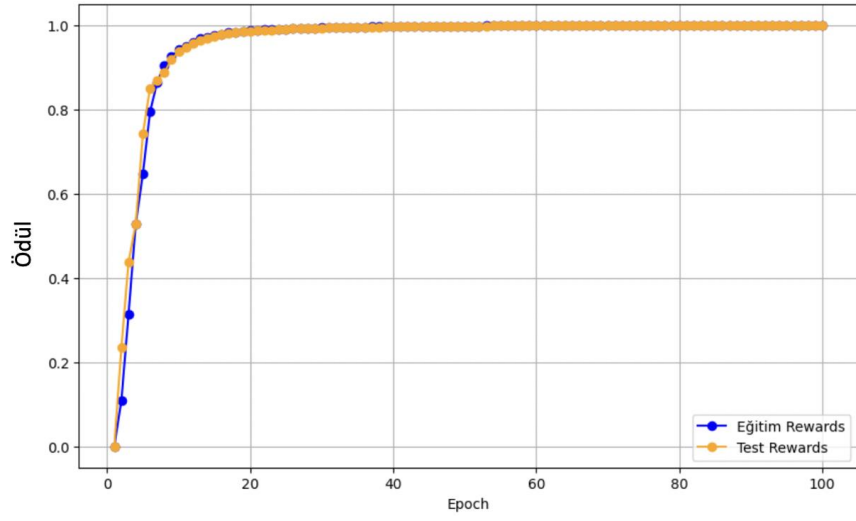
Şekil 111. GNN-LSTM-RL algoritması için rota sonuçları (Erdem *et al.* under review d)



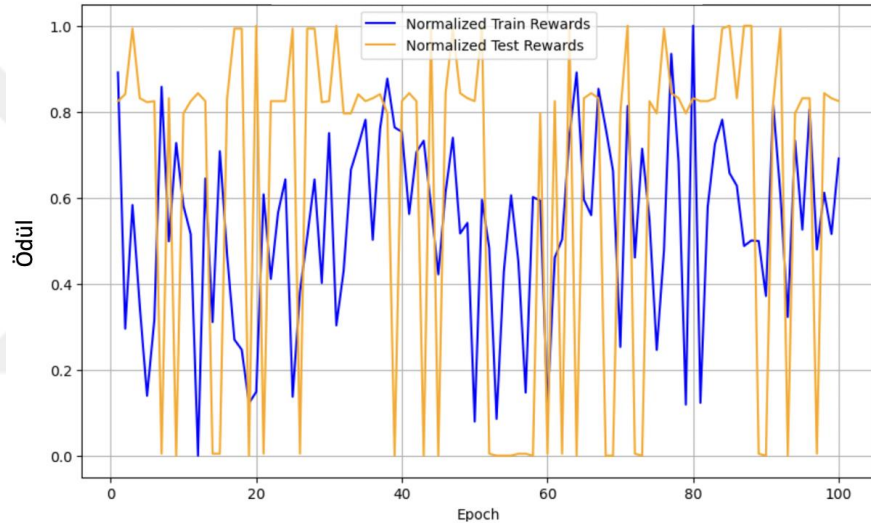
Şekil 112. GNN-RL algoritması için rota sonuçları (Erdem *et al.* under review d)



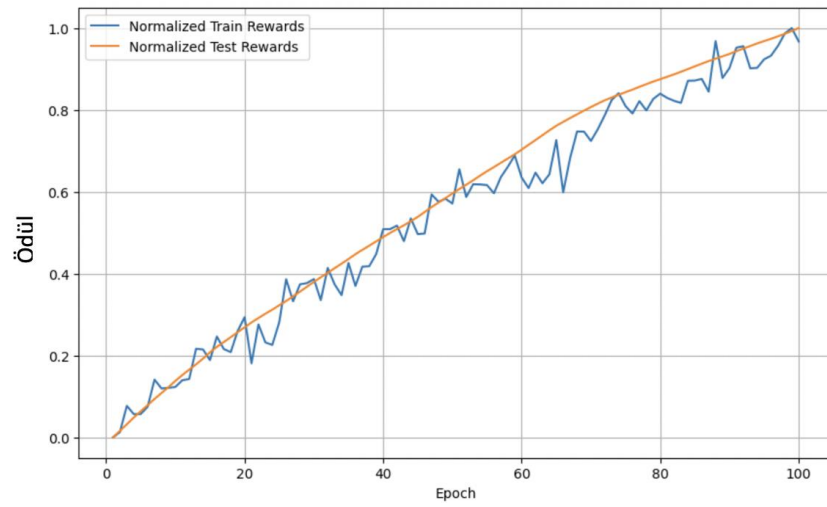
Şekil 113. BME algoritması için rota sonuçları (Erdem *et al.* under review d)



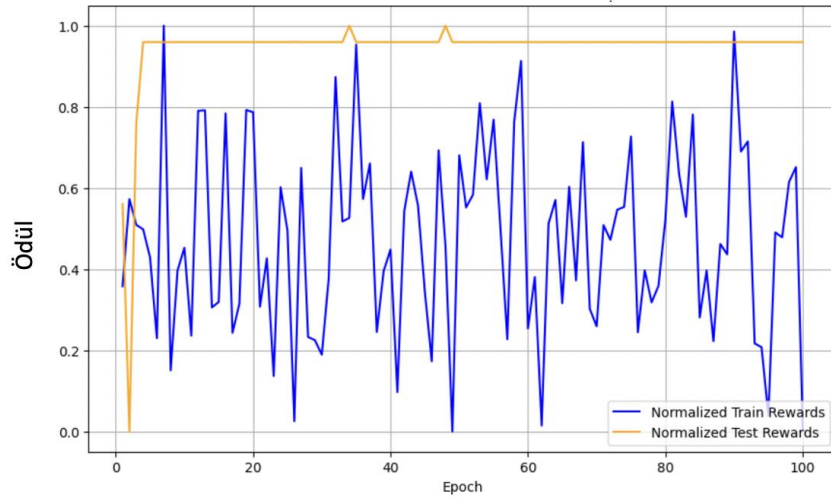
Şekil 114. Dikkat mekanizması için eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



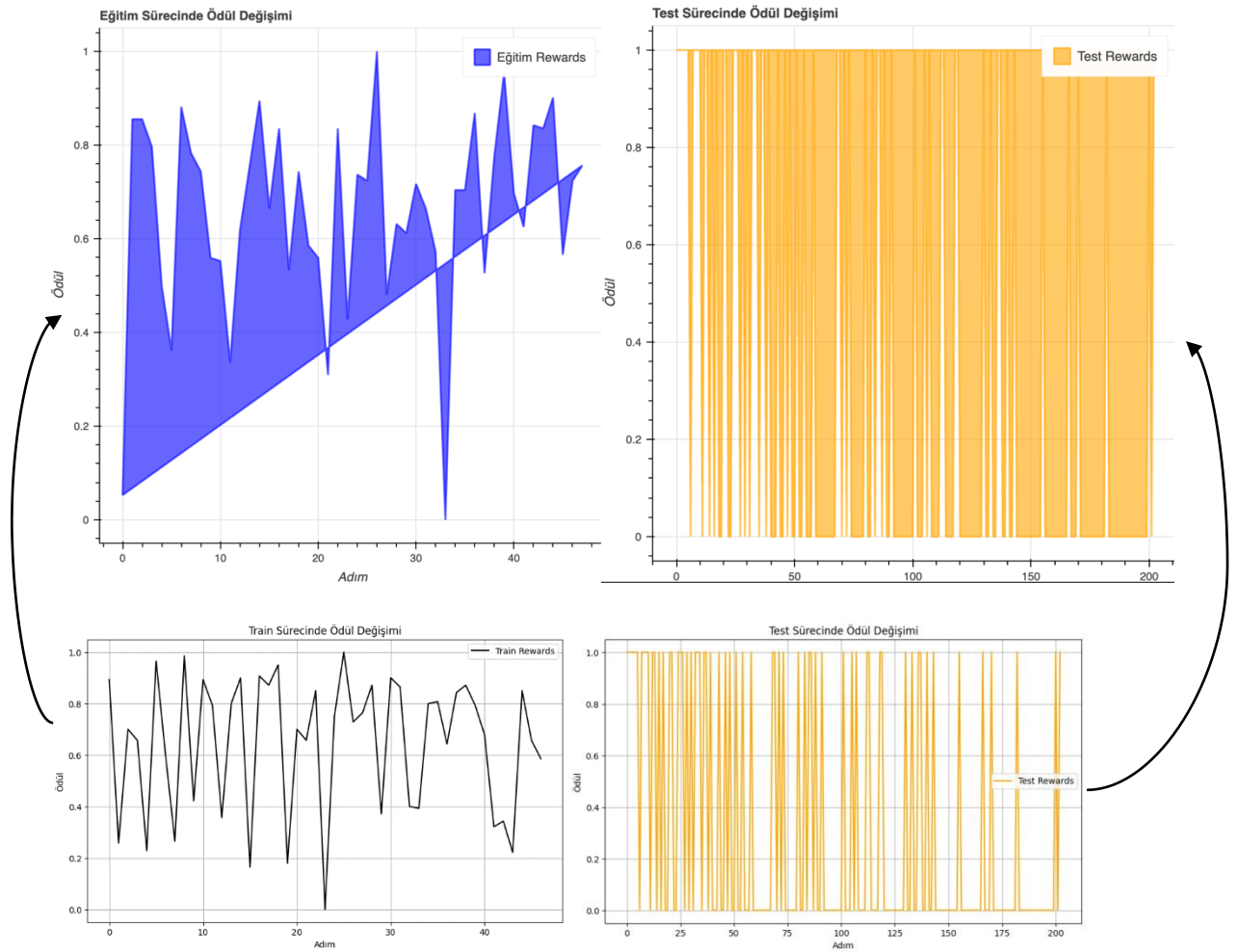
Şekil 115. BKA için eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



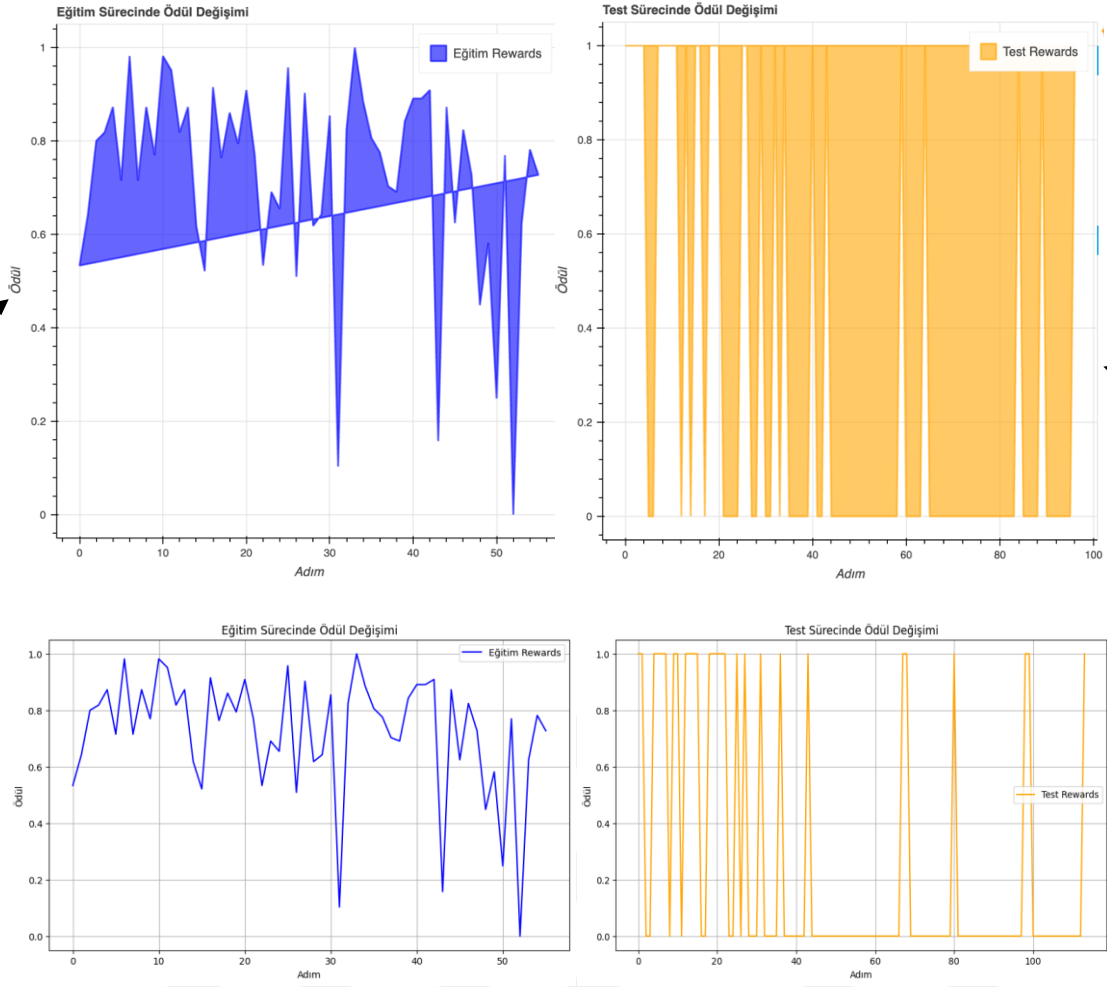
Şekil 116. DTBKA için eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



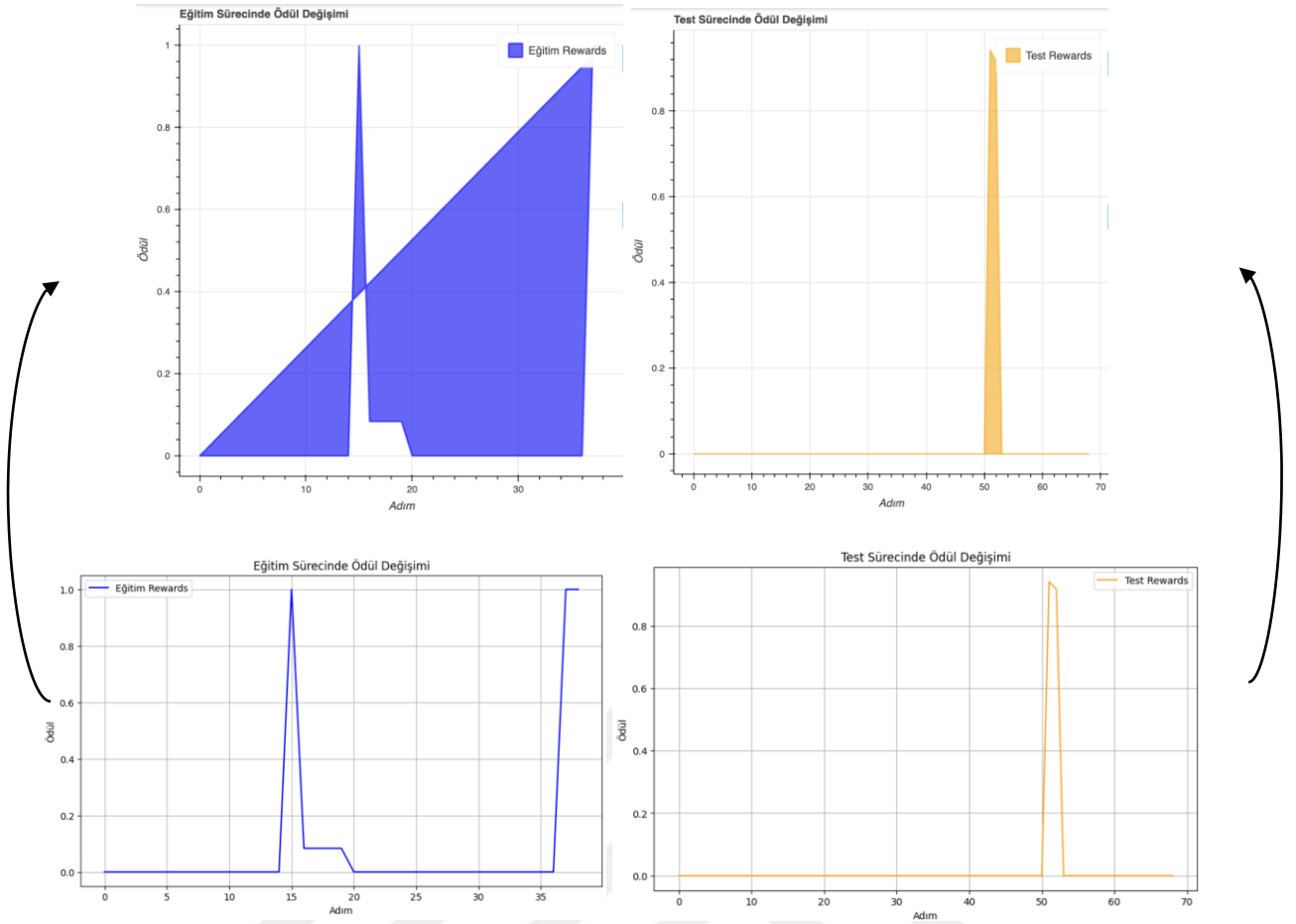
Şekil 117. DQN için eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



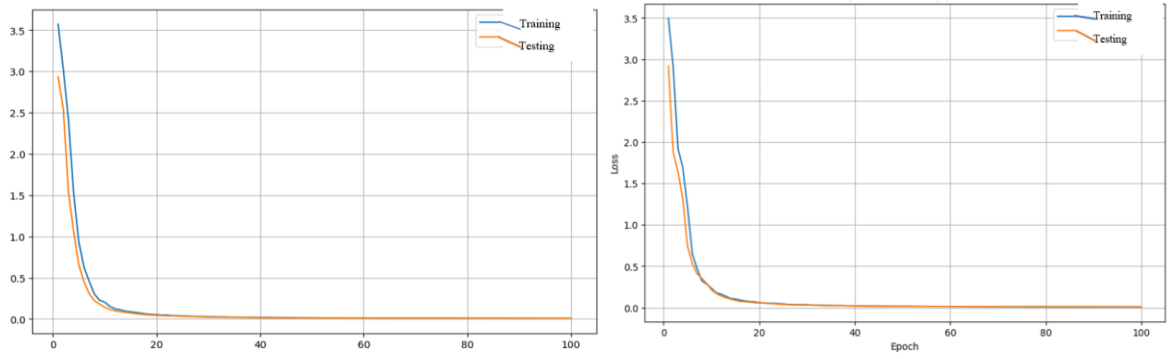
Şekil 118. GNN-LSTM-RL eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



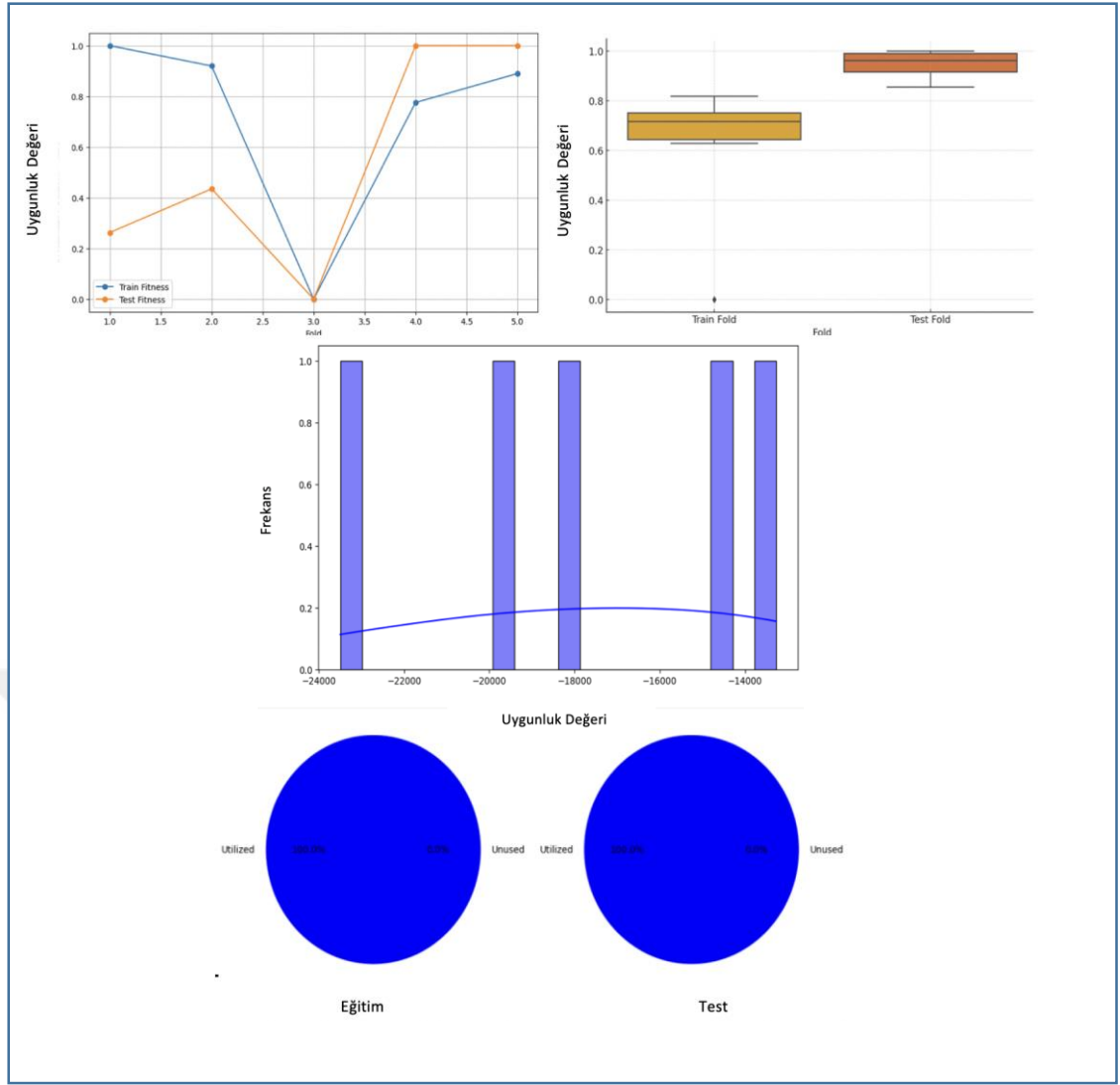
Şekil 119. GNN-RL eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



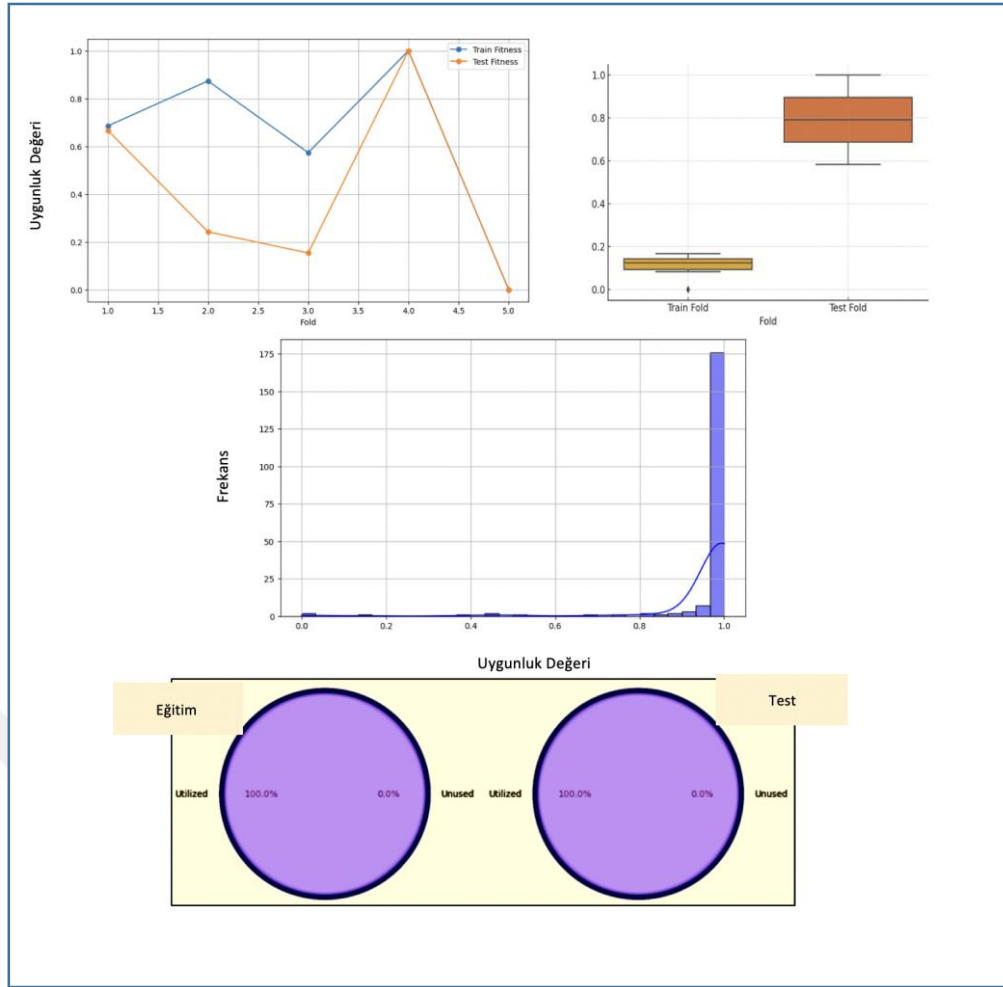
Şekil 120. BME eğitim ve test ödül analiz grafiği (Erdem *et al.* under review d)



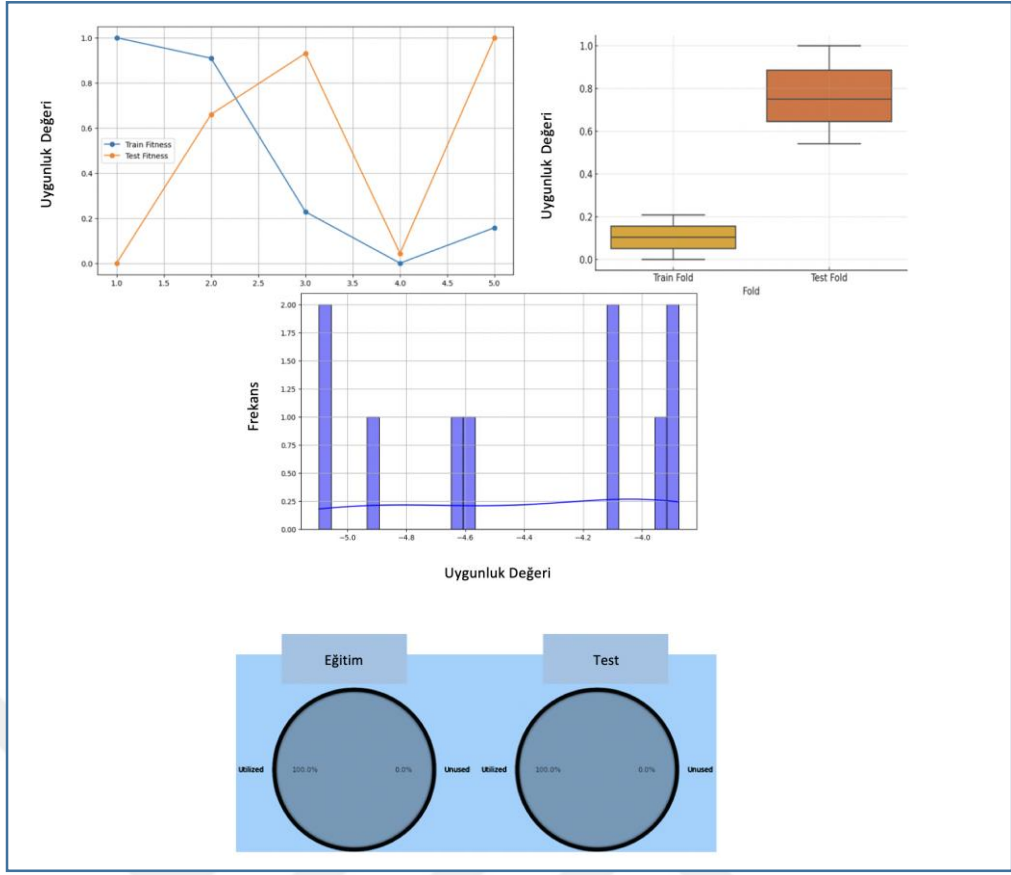
Şekil 121. Dikkat mekanizması ve DTBKA için sırasıyla eğitim ve test kayıp karşılaştırması (Erdem *et al.* under review d)



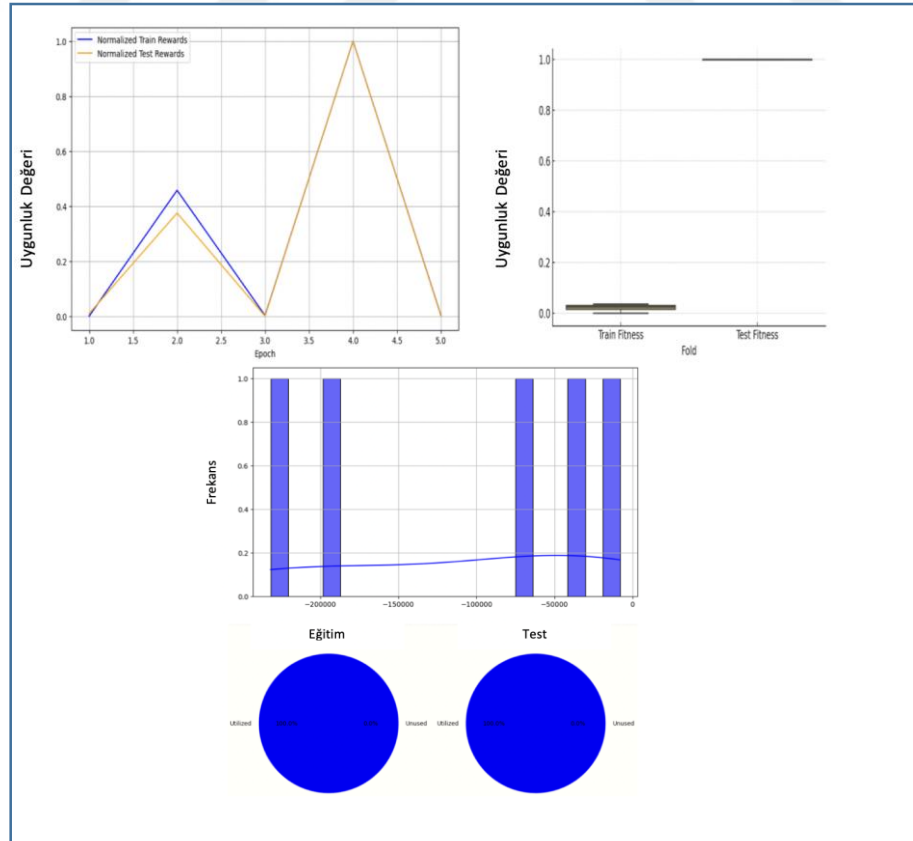
Şekil 122. BKA için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



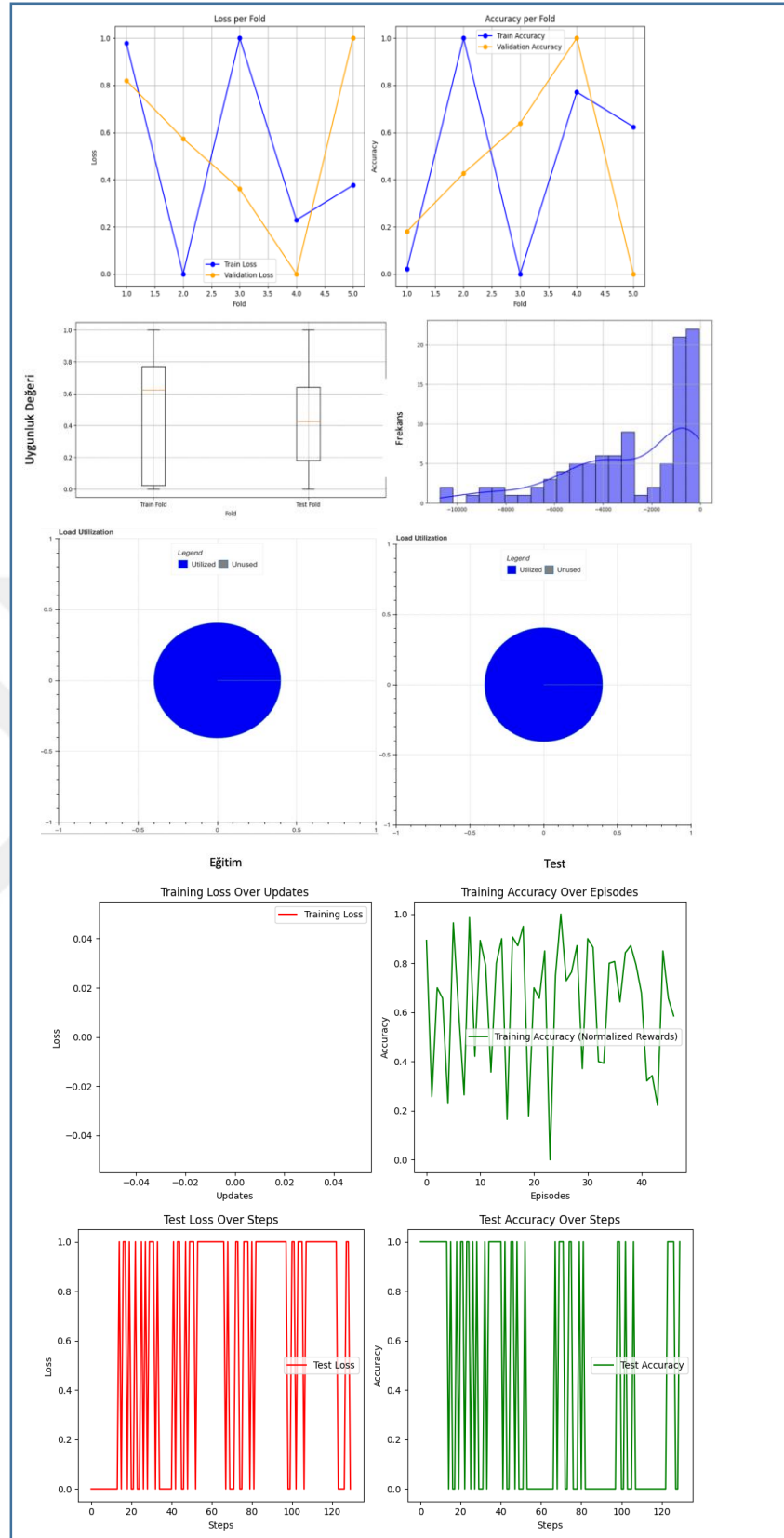
Şekil 123. Dikkat mekanizması için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



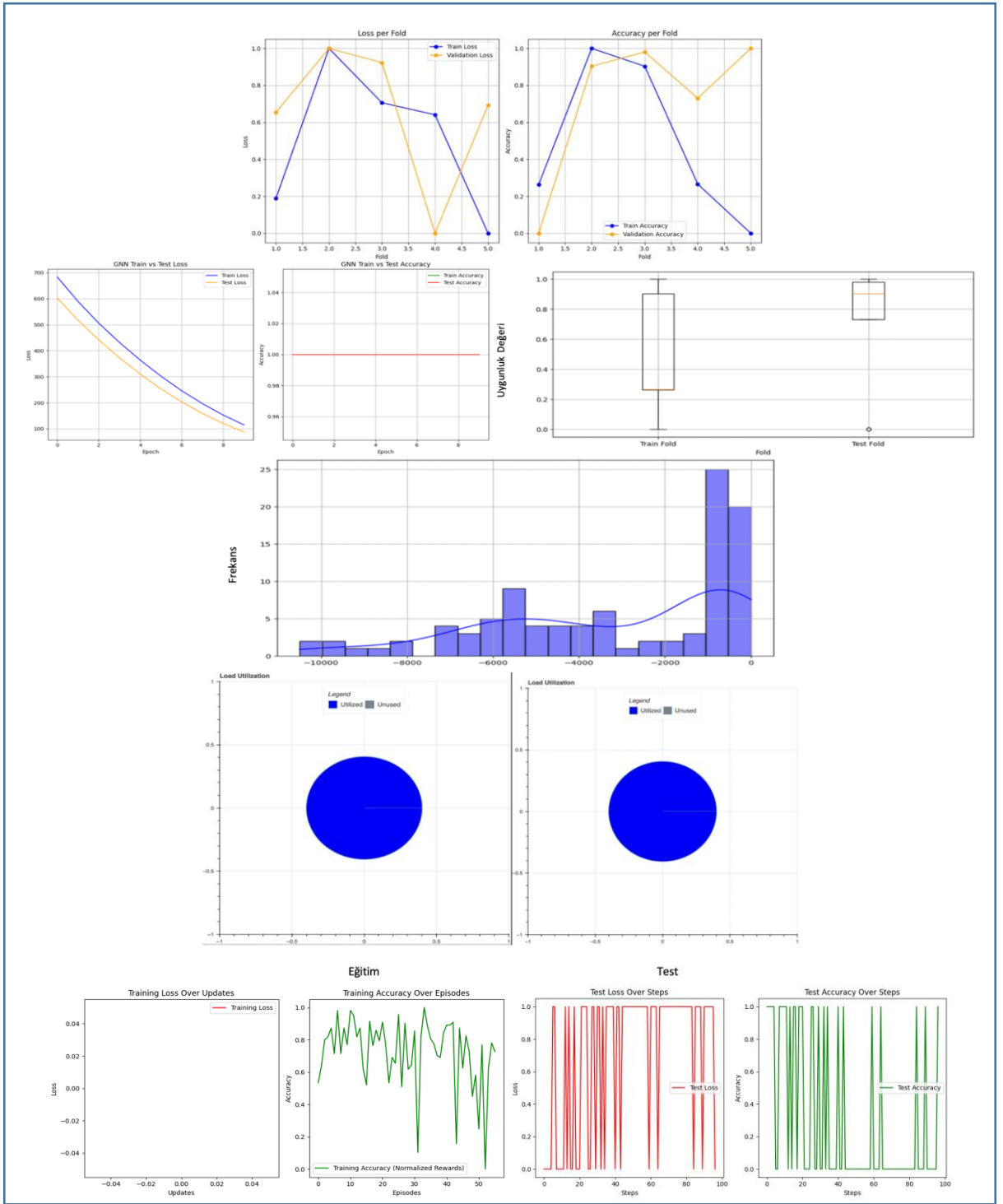
Şekil 124. DTBKA için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



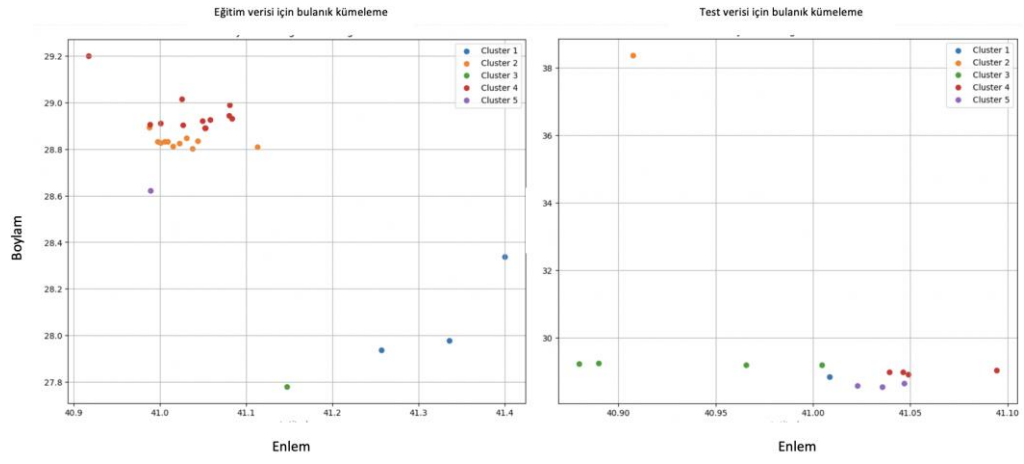
Şekil 125. DQN için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



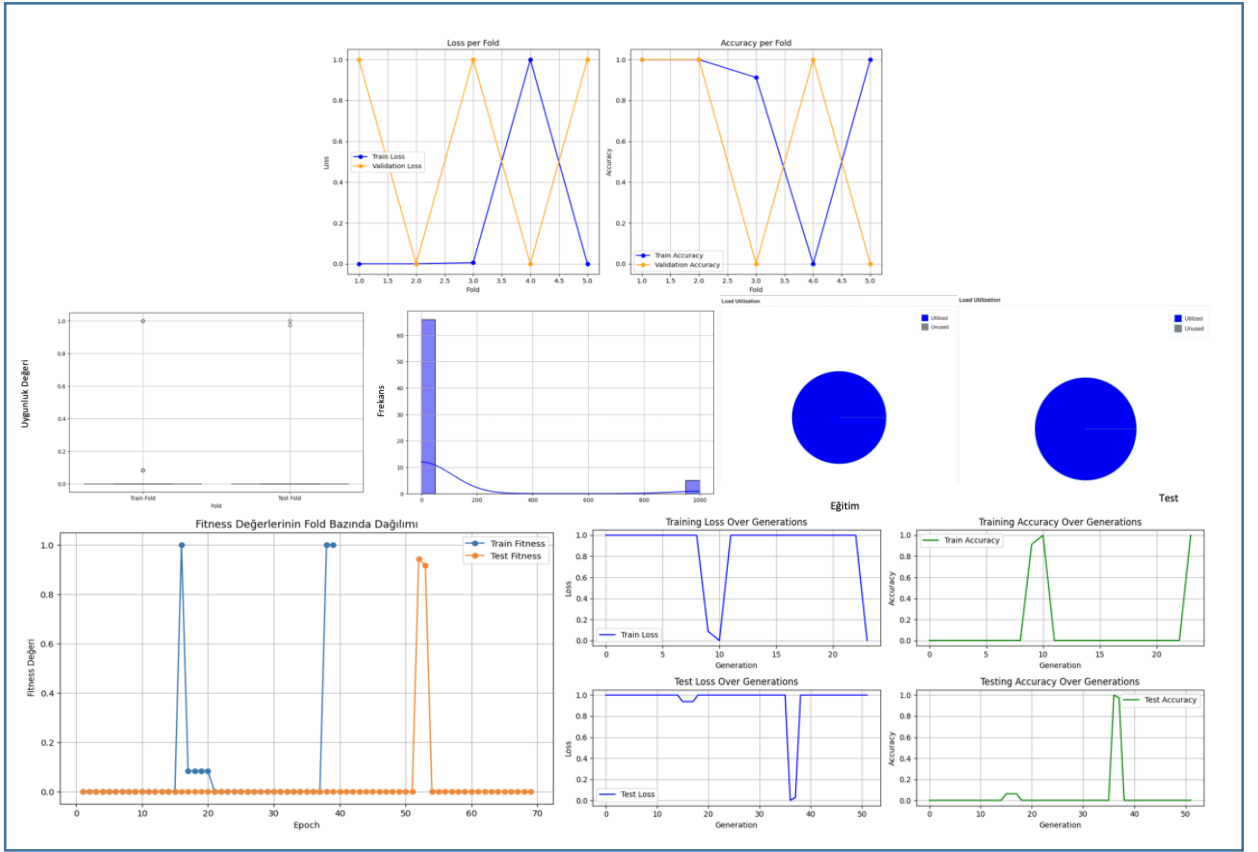
Şekil 126. GNN-LSTM-RL için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



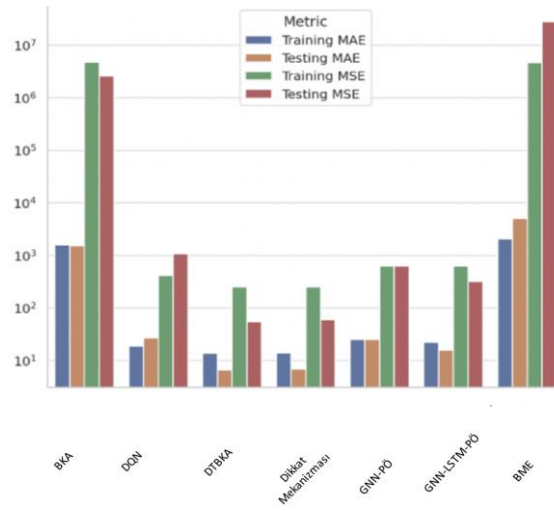
Şekil 127. GNN-RL için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



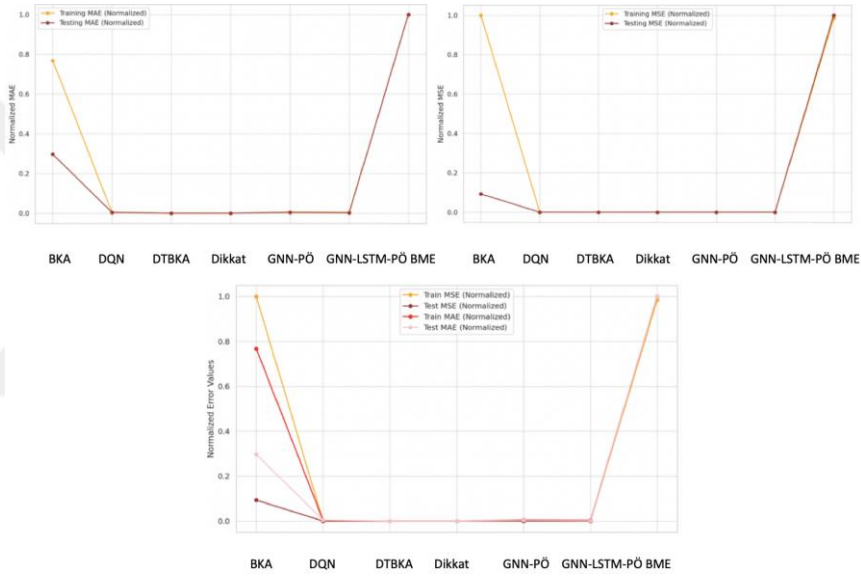
Şekil 128. BME algoritmasında bulanık kümeleme sonrasında eğitim ve test verisi (Erdem *et al.* under review d)



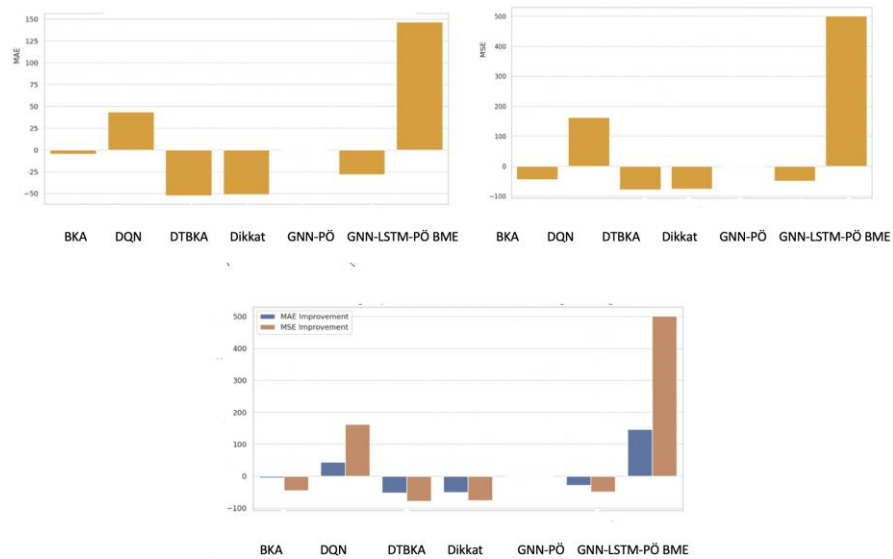
Şekil 129. BME için foldlar arasında fitness değeri dağılımı ve kullanım analizi (Erdem *et al.* under review d)



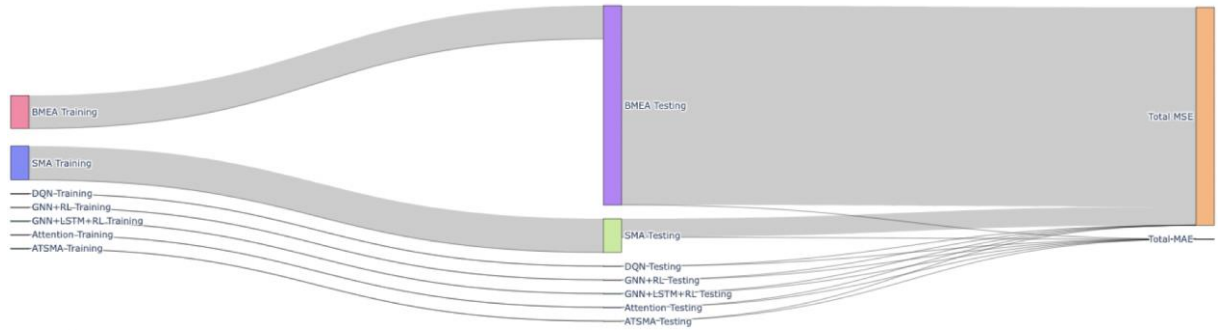
Şekil 130. Modellere göre eğitim ve test hatalarının karşılaştırması (logaritmik ölçekle) (Erdem *et al.* under review d)



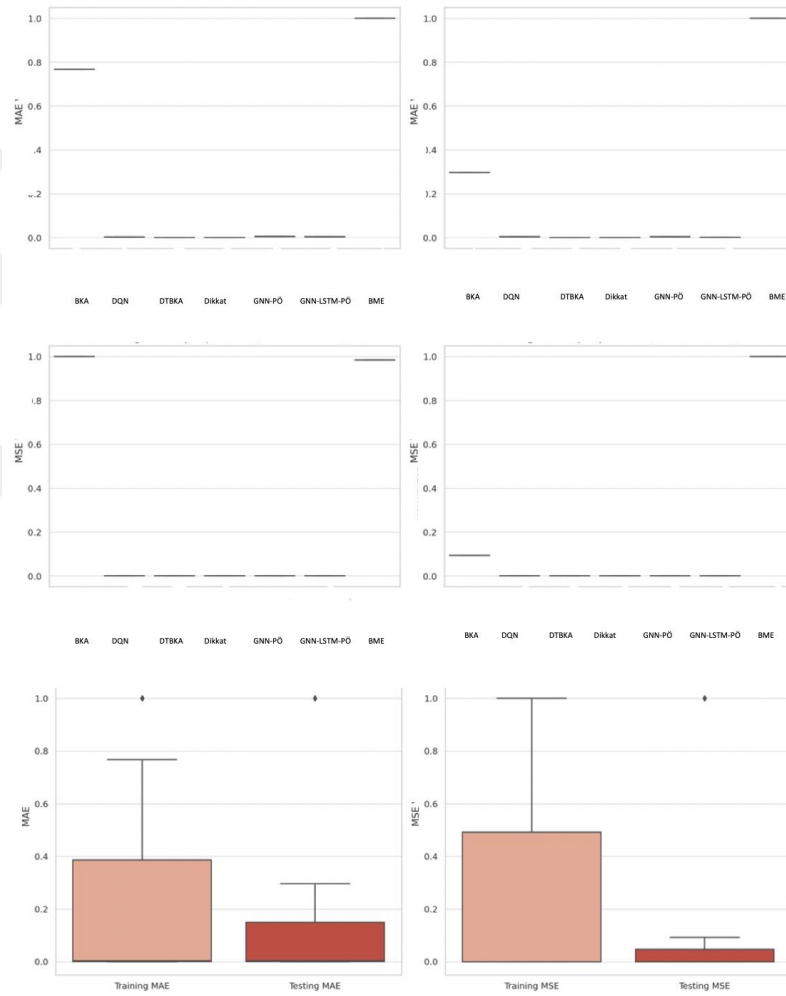
Şekil 131. Her modelin performans eğilimi (Erdem *et al.* under review d)



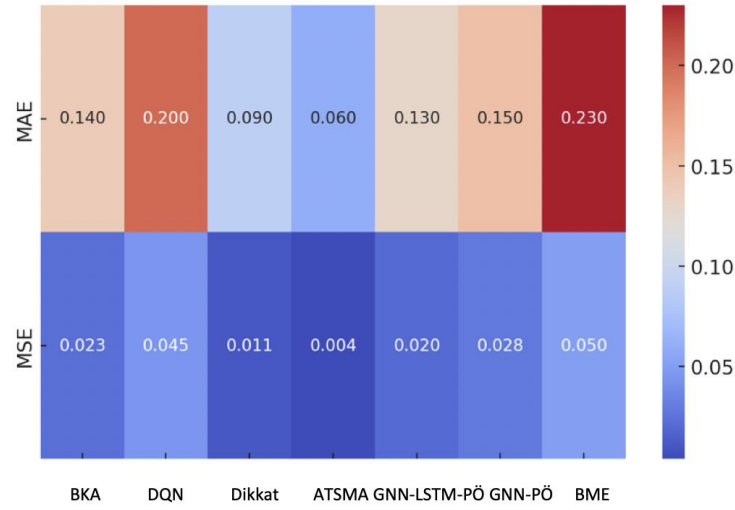
Şekil 132. Eğitim ve test arasındaki hata oranında yüzdelik iyileşme (Erdem *et al.* under review d)



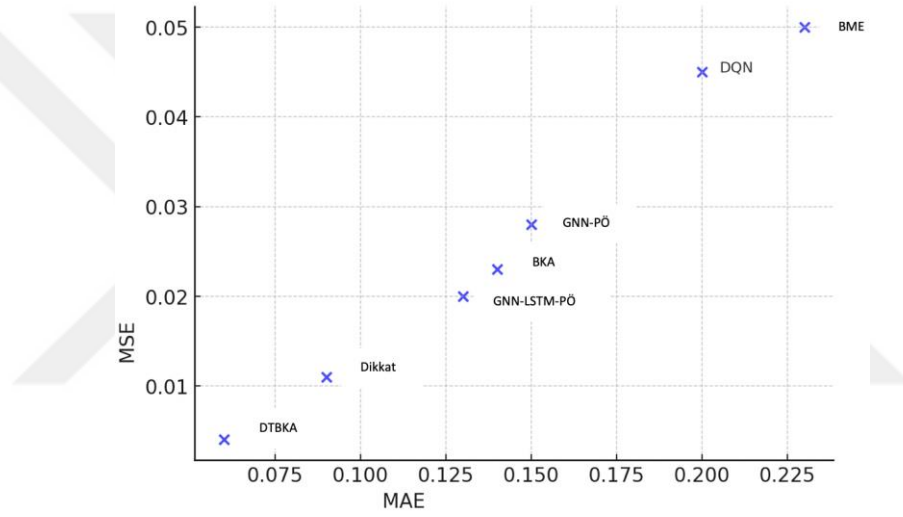
Şekil 133. Modeller arasındaki hata dağılımının sankey diyagramı (BKA=SMA, DTBKA=ATsMA, GNN-RL=GNN+RL, GNN-LSTM-RL=GNN+LSTM+RL, Dikkat Mekanizması=Attention) (Erdem *et al.* under review d)



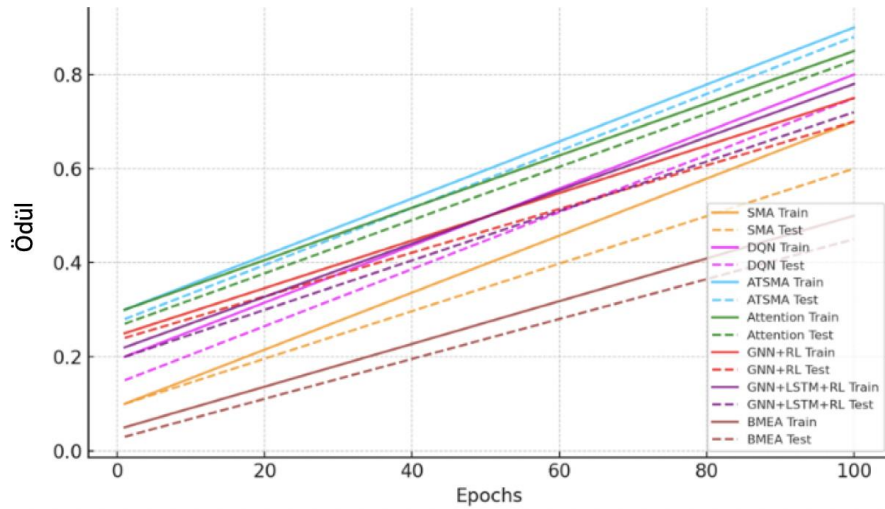
Şekil 134. Modellere göre hata dağılımı boxplot grafikleri (Erdem *et al.* under review d)



Şekil 135. Modeller arasında MAE ve MSE değerlerinin ısı haritası (Erdem *et al.* under review d)



Şekil 136. Tüm modeller için MAE ve MSE dağılım grafiği (Scatter Plot) (Erdem *et al.* under review d)



Şekil 137. Tüm algoritmalar için eğitim ve test ödül analizi özeti (BKA=SMA, DTBKA=ATSMa, GNN-RL=GNN+RL, GNN-LSTM-RL=GNN+LSTM+RL, Dikkat Mekanizması=Attention) (Erdem *et al.* under review d)

Tezin altıncı aşamasında, değerlendirme süreci, MSE, MAE, RMSE, DTW, MAPE, PAP10, Aşırı Tahmin Oranı (POFP) ve Kapsam Skoru gibi ortak metrikler çerçevesinde; ayrıca K-katlı çapraz doğrulama ve çeşitli görsel analizler (çubuk grafikler, logaritmik dönüşümler, normalize edilmiş ölçekler) ile gerçekleştirilmiştir. Şekil 138-144'e göre (Erdem *et al.* under review c), hata bazlı metriklerde modeller arasında küçük farklar gözlemlenmiş; ancak PAP10 ve kapsam skoru gibi zamana duyarlılık ve dayanıklılıkla ilgili metriklerde ayrışma daha belirgin hâle gelmiştir. LST-Former modeli, en düşük hata değerlerini ve en yüksek kapsam skorlarını sunarak genel olarak en başarılı performansı sergilemiştir. Ayrıca Şekil 145-148'de de görüldüğü gibi bu model, hem doğruluk hem de uygulanabilirlik açısından en yüksek bileşik başarı puanına ulaşmıştır (8,62). AST-Former, özellikle zamana duyarlı teslimat senaryolarında güçlü PAP10 ve kapsam skorları ile dikkat çekmiştir. ALT-Former modeli ise, daha yüksek hata oranlarına ve geç teslimatlara daha fazla tolerans gösteren yapısıyla esneklik sağlamıştır. Meta-öğrenme tabanlı hibrit modellerden LST-MetaDRL, uyarlanabilirlik ile doğruluk arasında güçlü bir denge yakalamıştır. Bu model, değişken müşteri talepleri altında hızlı adaptasyon kabiliyeti sergilemiş ve öğrenme süreci boyunca genel olarak stabil bir başarı ortaya koymuştur. ALT-MetaDRL ve AST-MetaDRL modelleri ise, eğitim süreci boyunca özellikle DTW ve MAPE metriklerinde yüksek varyans göstererek dinamik ve seyrek müşteri senaryolarına karşı daha hassas olduklarını ortaya koymuştur. Bununla birlikte, bu modeller MAML sayesinde yüksek düzeyde uyarlanabilirlik kapasitesine sahiptir ve farklı dağıtım senaryolarında hızlı öğrenme avantajı sunmaktadır. NRL modeli ise, NEAT algoritması sayesinde sinir ağlarının yapısını evrimleştirerek hafif yapılı ve genellenebilir çözümler üretmiştir. Ham hata performansında LST-Former kadar güçlü olmasa da, genel metriklerde istikrarlı sonuçlar sunmuş ve özellikle test verilerinde yüksek yapısal uyum sergilemiştir. Görsel analizler (kayıp eğrileri, ödül eğrileri, fitness dağılımları, başarı histogramları), Transformer tabanlı modellerin (özellikle LST-Former ve AST-Former) yüksek genelleme kabiliyetine sahip olduğunu ortaya koymuştur. DRL-MAML tabanlı modeller (LST-MetaDRL ve AST-MetaDRL), erken evrelerde hızlı öğrenme sağlarken, daha sonra hafif dalgalanmalar göstermiştir. 5 katlı çapraz doğrulama ile hesaplanan fitness skorlarının dağılımı, LST-Former, AST-Former ve NRL modellerinin düşük varyansla daha genellenebilir çözümler sunduğunu göstermiştir. DRL tabanlı modeller (ALT-MetaDRL, LST-MetaDRL), çevresel değişkenliklere karşı daha yüksek varyans sergilemiştir. Şekil 145, PAP10, kapsam skoru ve normalize edilmiş hata skorlarını ağırlıklı bir formülle birleştirerek oluşturulan normalize edilmiş başarı oranlarını göstermiştir (Denklem 26) (Erdem *et al.* under review c).

$$\text{Başarı Oranı} = 0.4 \times \text{PAP10} + 0.3 \times \text{Coverage} - 0.3 \times \text{Normalize Hata} \quad (26)$$

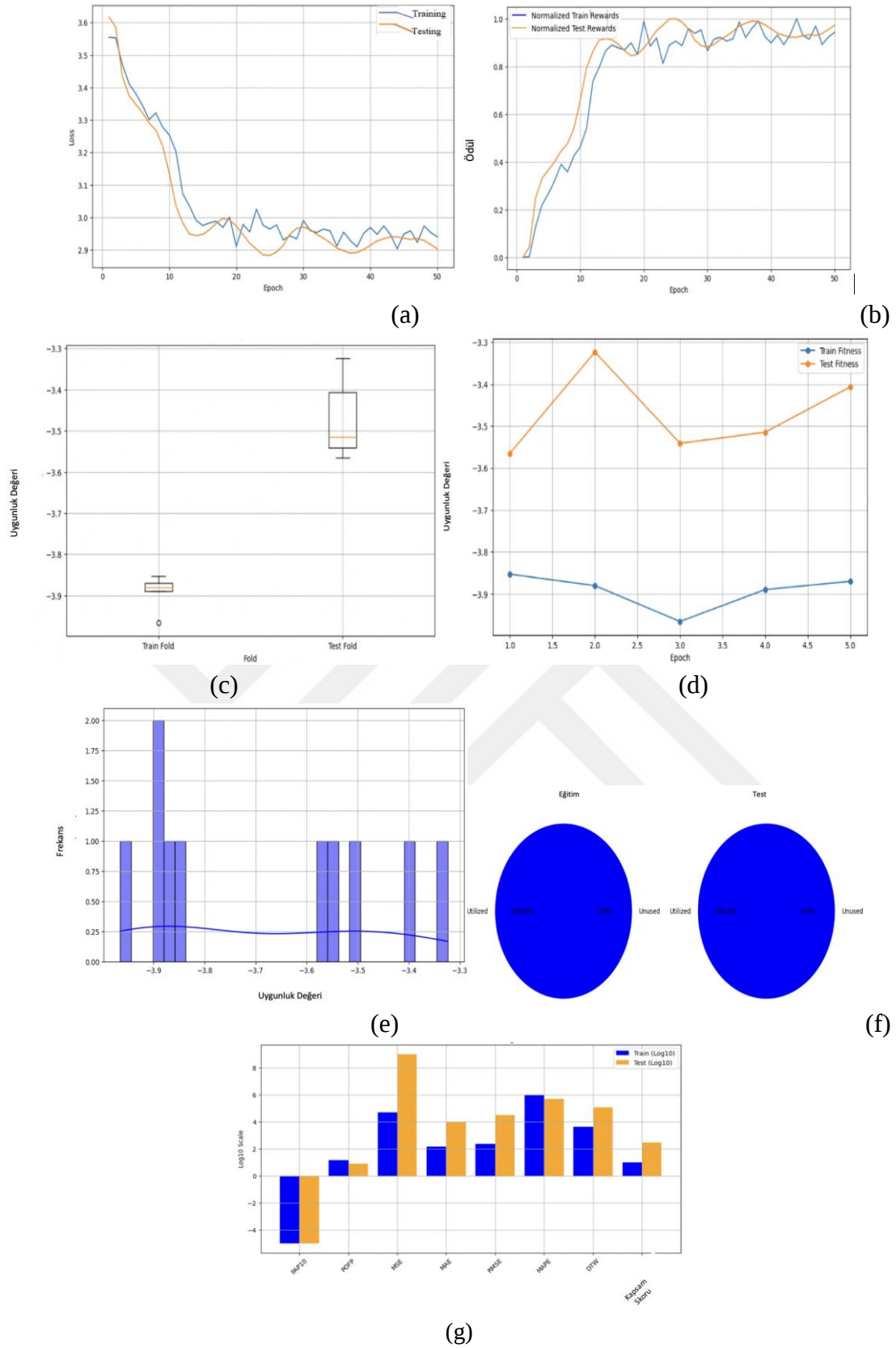
Bu formül, kısa vadeli teslimat doğruluğu (PAP10), örnekleme genelleme (kapsam skoru) ve rota tahmin hatasını dengeli biçimde değerlendirir. LST-Former, en yüksek başarı oranını elde etmiştir. AST-Former, yüksek kapsam ve erken teslimat doğruluğuyla onu takip etmiştir. AST-MetaDRL ve LST-MetaDRL, daha geniş dağılım ve düşük skorlarla eğitim varyansına daha duyarlı yapıdadır. Şekil 146, tüm modellerin MSE, MAE, RMSE, MAPE, DTW ve kapsam skorlarını karşılaştıran radar diyagramıdır. En küçük alan kaplayan modeller, hata metriklerinde daha düşük değerler elde etmiş ve genel başarıyı göstermiştir. Şekil 147, 5 katlı çapraz doğrulamada her modelin ortalama başarı oranı ve %95 güven aralıklarını göstermiştir. LST-Former ve AST-Former gibi Transformer tabanlı modeller daha dar aralıklara sahipken, DRL-MAML tabanlı modeller (özellikle AST-MetaDRL) daha yüksek varyans sergilemiştir. Bu durum, DRL ajanlarının stokastik çevresel faktörlere karşı daha hassas olduğunu göstermiştir (Erdem *et al.* under review c).

Şekil 148 ve Şekil 149, performans, hesaplama maliyeti, uygulanabilirlik ve uyarlanabilirlik olmak üzere dört temel boyutta metodolojik değerlendirme sonuçlarını göstermiştir: Puanlandırma süreci, hem ampirik verilerden elde edilen sonuçlara hem de lojistik optimizasyon ve AI tabanlı rota planlama konularında deneyimli uzmanların bilgisine dayalı göre gerçekleştirilmiştir (Erdem *et al.* under review c).

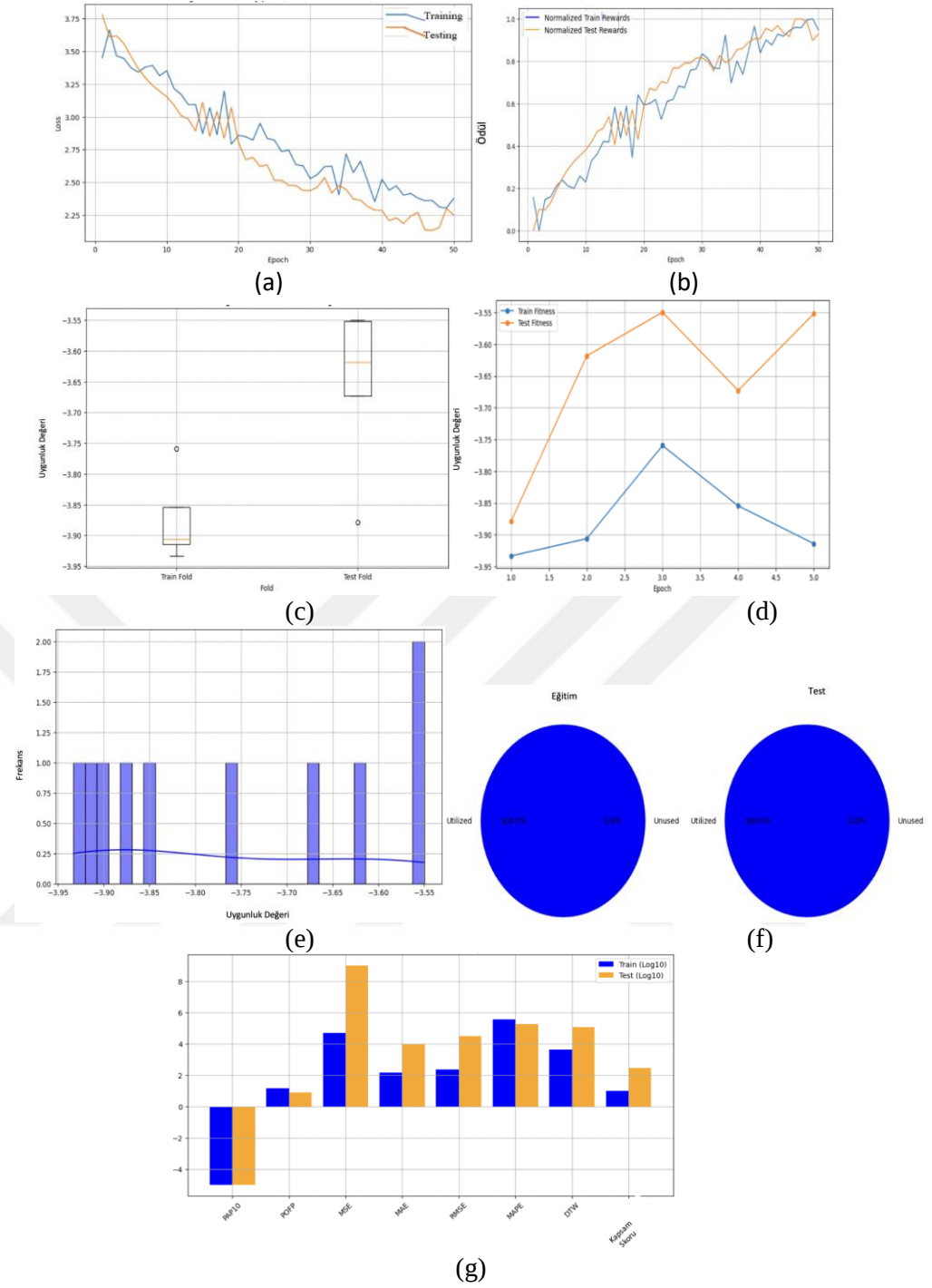
- ✓ Performans Puanı: Hata bazlı metriklere dayalıdır; LST-Former en yüksek (9,2) puanı almıştır.
- ✓ Hesaplama Maliyeti: Parametre sayısı ve mimari karmaşıklığa göre değerlendirilmiş; AST-Former ve ALT-Former en iyi skorları (9) almıştır.
- ✓ Uygulanabilirlik: Gerçek senaryolara uygunluk açısından değerlendirilmiş; LST-Former en yüksek (9) puana sahiptir.
- ✓ Uyarlanabilirlik: MAML entegrasyonu sayesinde LST-MetaDRL en yüksek uyarlanabilirlik skorunu (9) almıştır.

Toplamda LST-Former, 8,62 bileşik skorla genel anlamda en başarılı model olarak öne çıkmıştır.

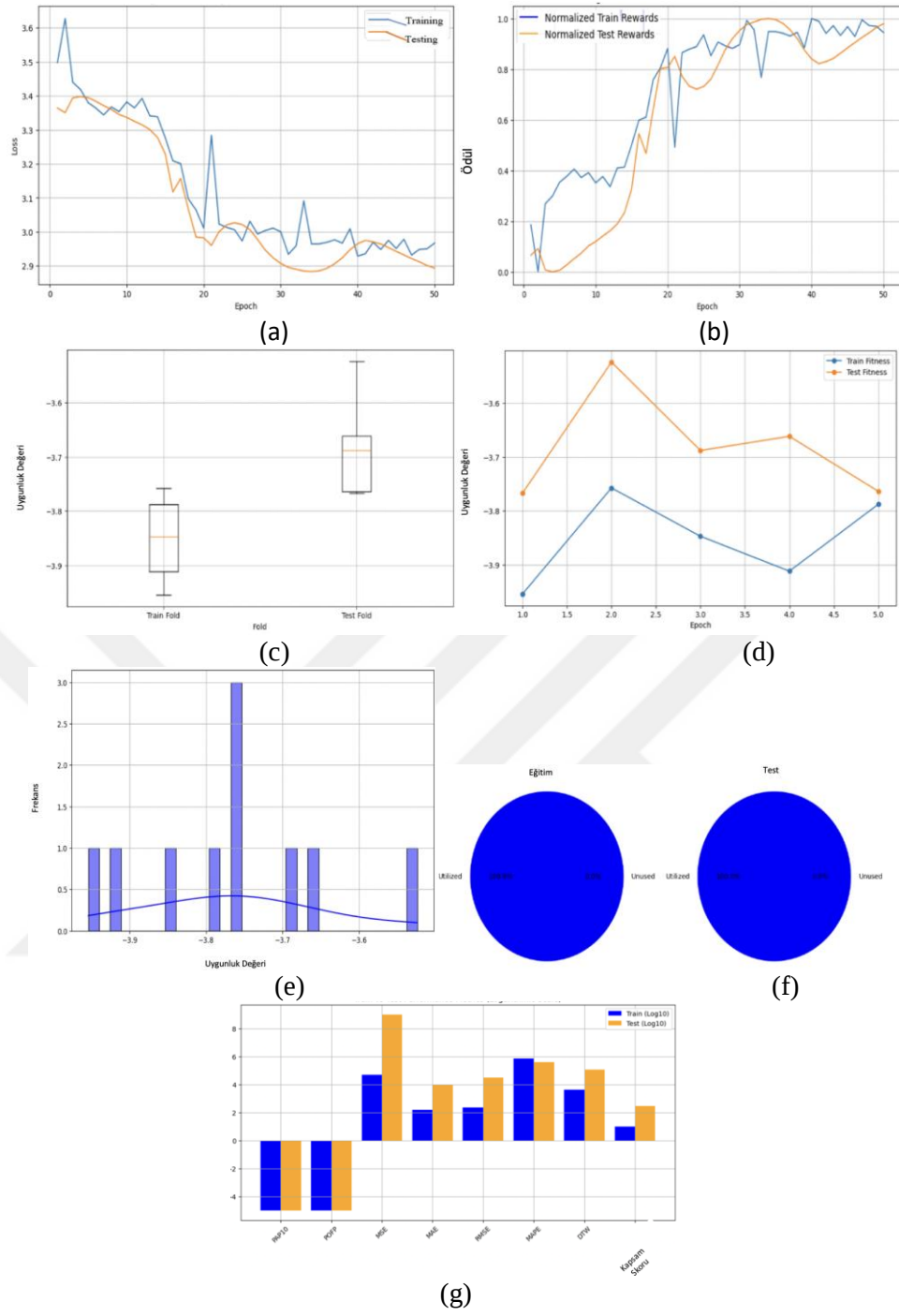
Son olarak, yorumlanabilirlik (interpretability) açısından Transformer tabanlı modeller (örneğin LST-Former), dikkat ağırlıkları sayesinde analiz edilebilir yapı sunarken; meta-öğrenme ve NEAT tabanlı modeller, daha esnek olmalarına rağmen sınırlı yorumlanabilirlik sergilemiştir. Bu kapsamda, hem teorik derinlik hem de pratik uygulanabilirlik açısından geliştirilen çözümler, gerçek zamanlı lojistik ağlar için önemli bir katkı sunmuştur.



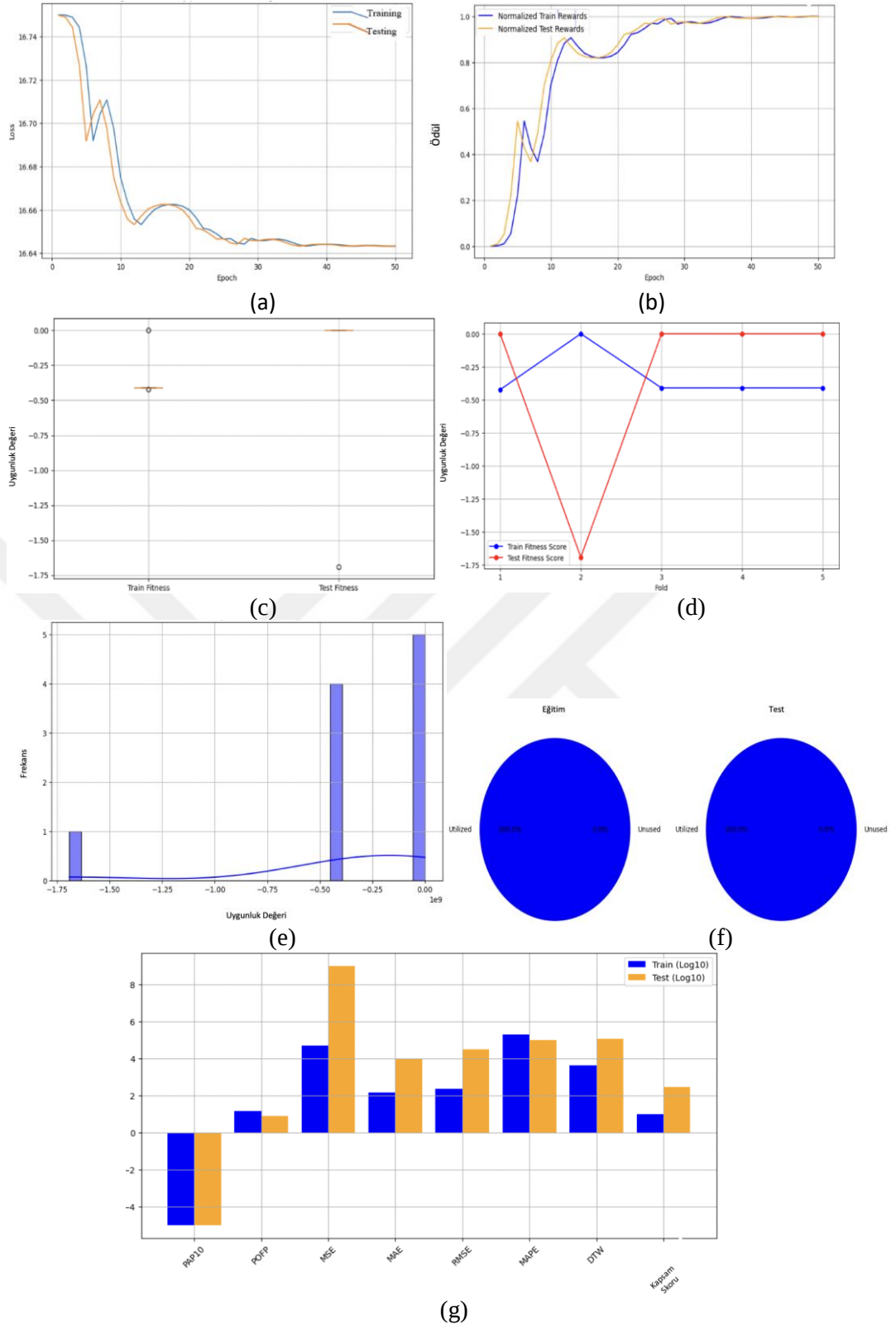
Şekil 138. AST-Former modeli sonuçları (Erdem *et al.* under review c)



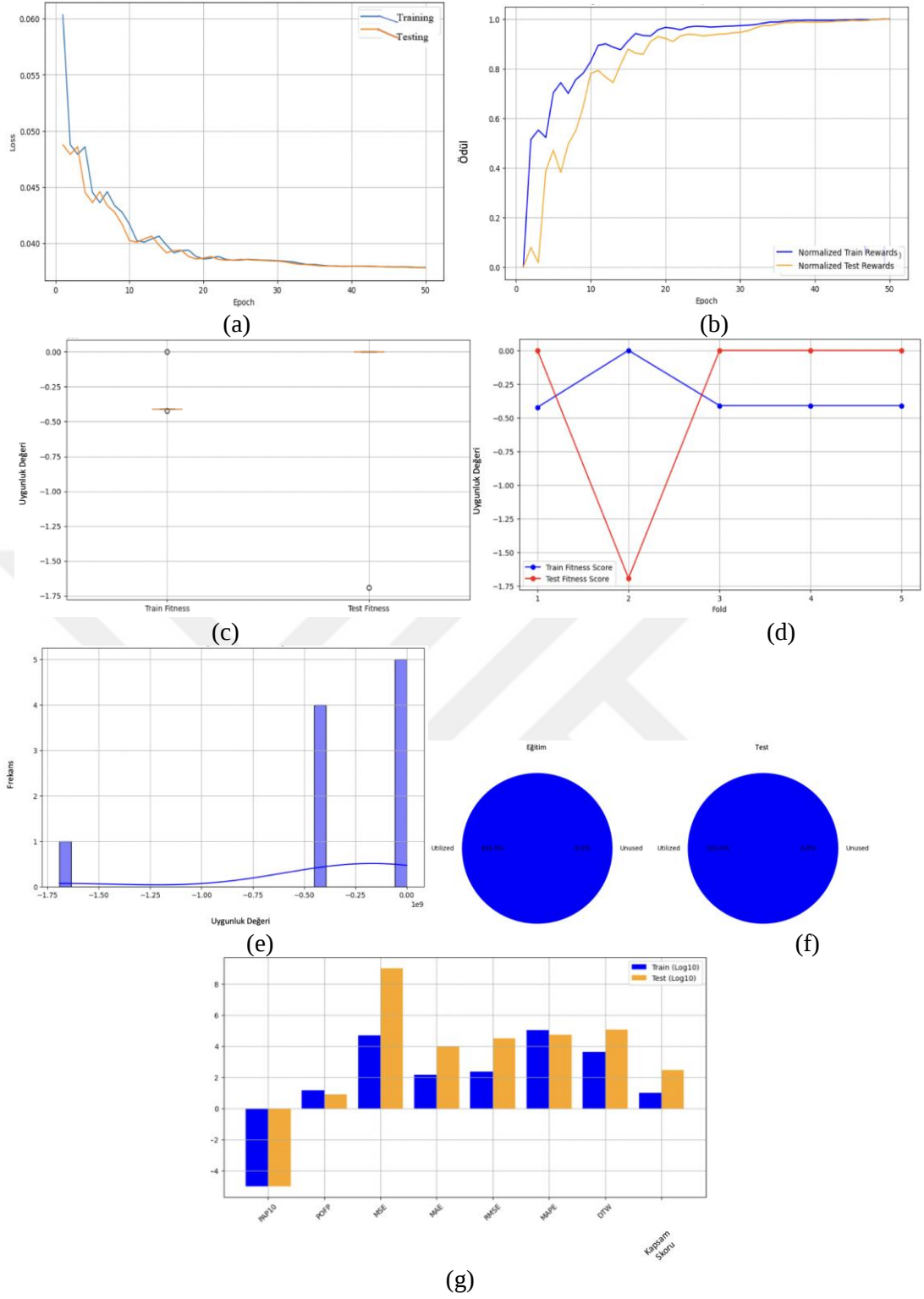
Şekil 139. ALT-Former modeli sonuçları (Erdem *et al.* under review c)



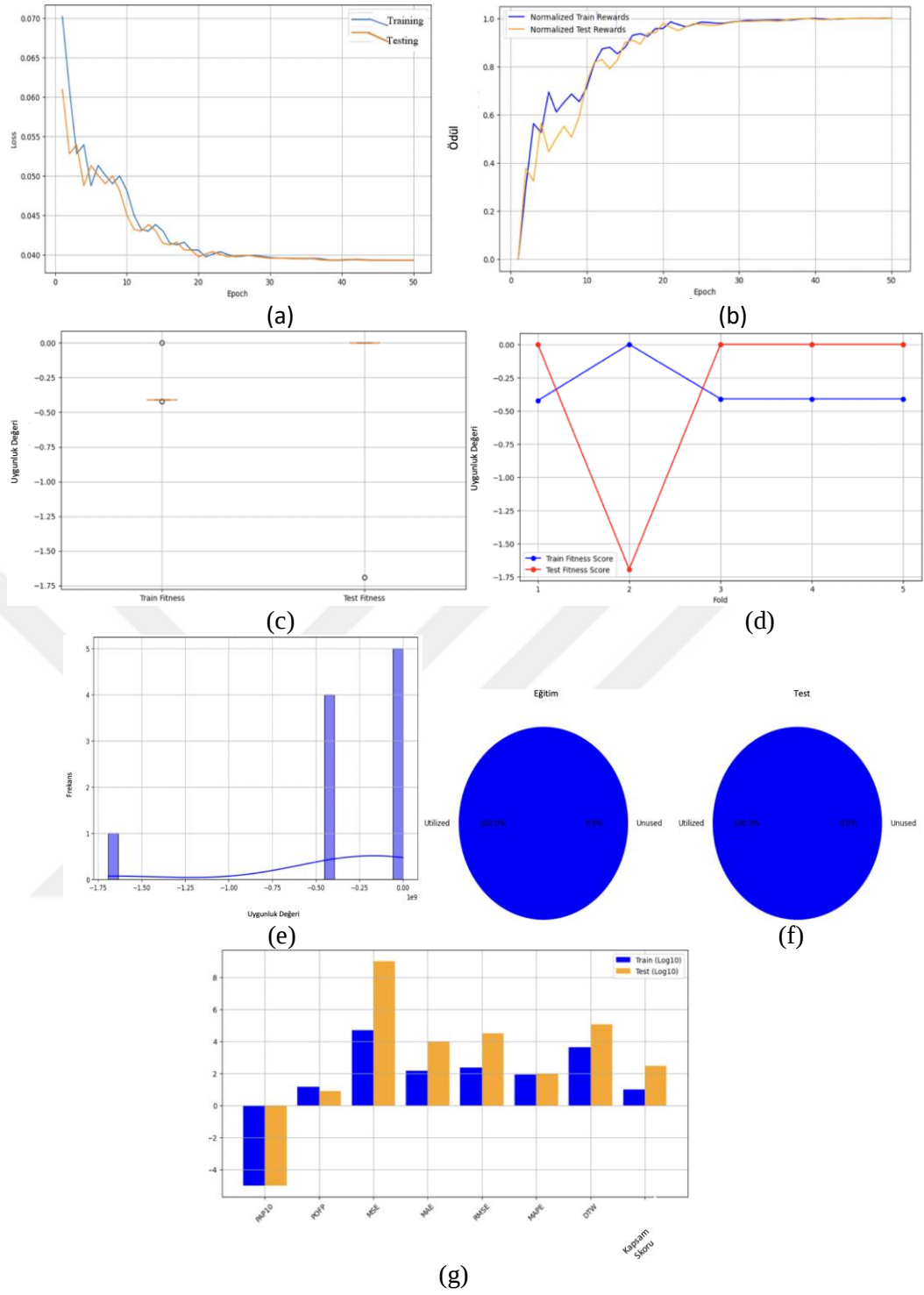
Şekil 140. LST-Former modeli sonuçları (Erdem *et al.* under review c)



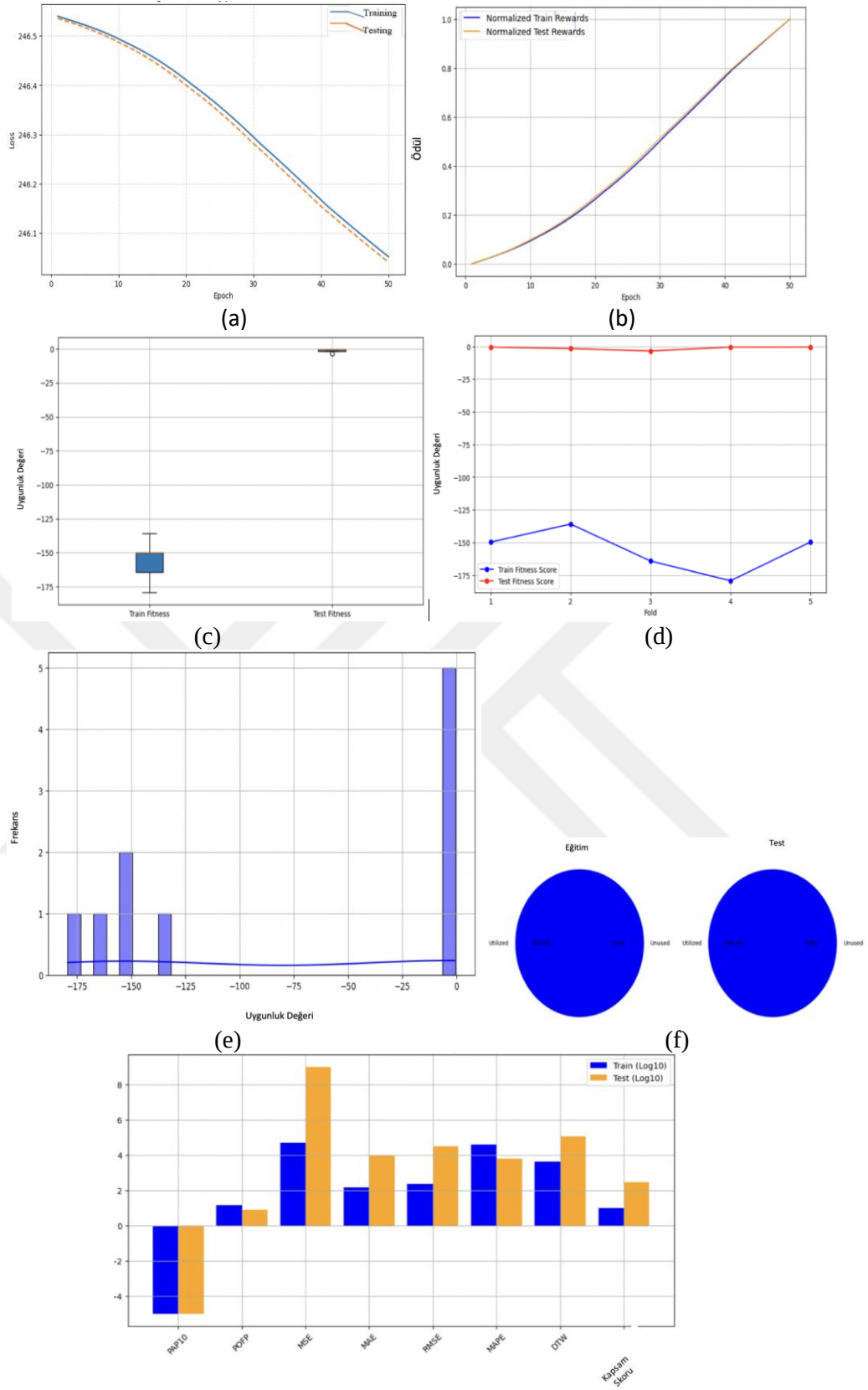
Şekil 141. LST-MetaDRL modeli sonuçları (Erdem *et al.* under review c)



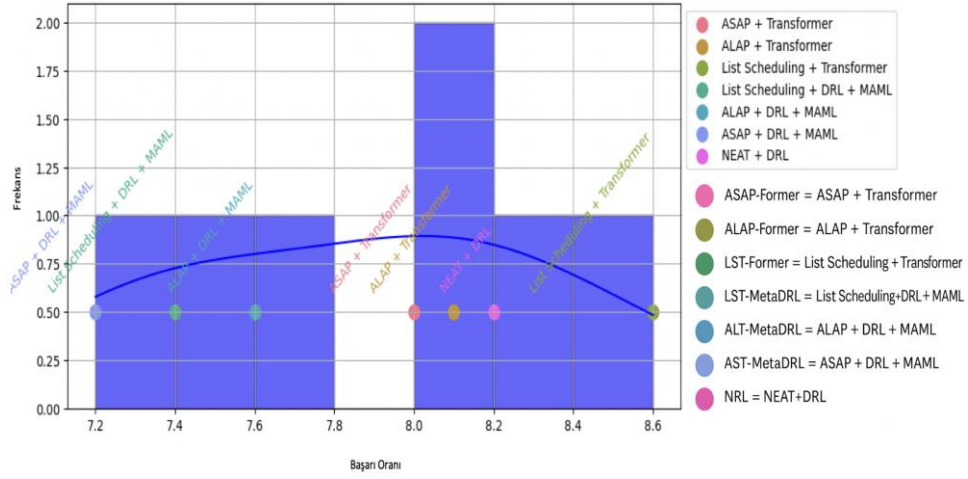
Şekil 142. ALT-MetaDRL modeli sonuçları (Erdem *et al.* under review c)



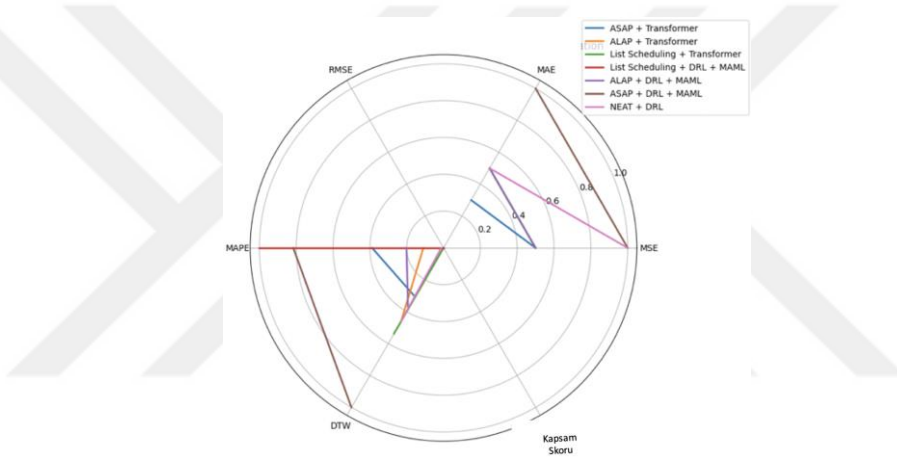
Şekil 143. AST-MetaDRL modeli sonuçları (Erdem *et al.* under review c)



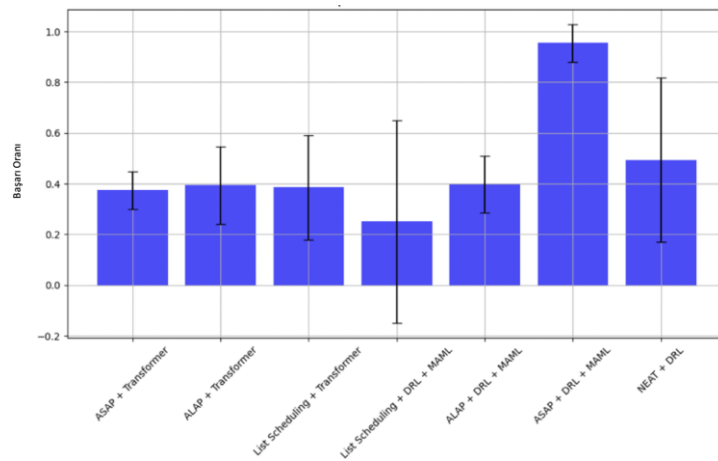
Şekil 144. NRL modeli sonuçları (Erdem *et al.* under review c)



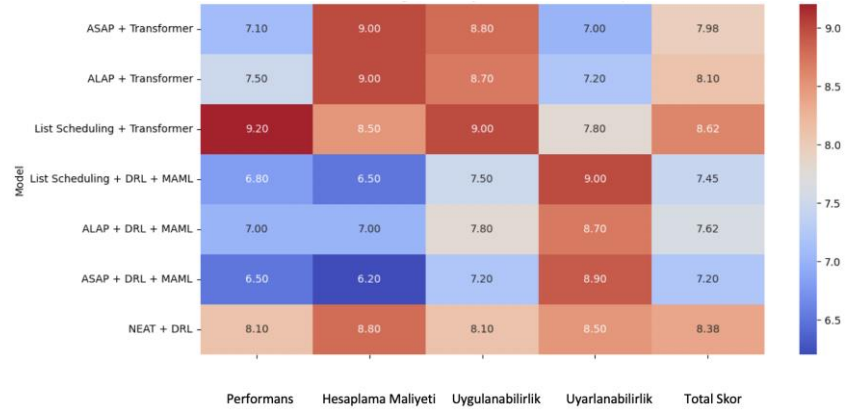
Şekil 145. AI tabanlı yönlendirme modelleri genelinde normalleştirilmiş başarı oranlarının histogramı (Erdem *et al.* under review c)



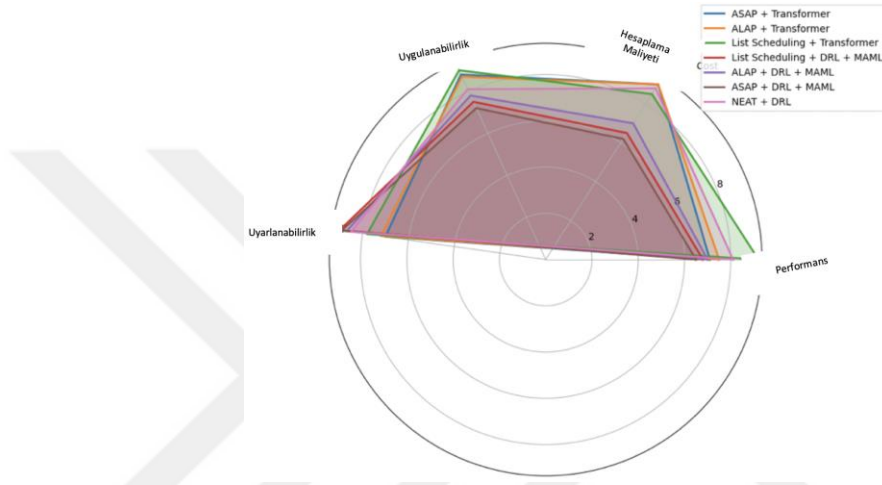
Şekil 146. Model performans değerlendirmesinin radar grafik gösterimi (Erdem *et al.* under review c)



Şekil 147. %95 güven aralıklarıyla başarı oranı karşılaştırması (Erdem *et al.* under review c)



Şekil 148. Metodolojik kalite değerlendirme-ısı haritası (Erdem *et al.* under review c)



Şekil 149. Metodolojik kalite değerlendirme radar çizelgesi (Erdem *et al.* under review c)

TARTIŞMA ve SONUÇ

2015-2024 dönemi, ÇDARP alanında AI ve sezgisel algoritmalara olan ilgide belirgin bir artış olduğunu göstermektedir. Bu dönemde yapılan çalışmalar, ÇDARP'nin çözümüne yönelik olarak yenilikçi algoritmaların geliştirilmesine, Coğrafi Bilgi Sistemleri (GIS) entegrasyonunun sağlanmasına ve çok amaçlı optimizasyon tekniklerinin uygulanmasına odaklanmıştır. Bu gelişmeler, teorik anlayışın derinleştirilmesini sağlarken, aynı zamanda çeşitli sektörlerde pratik ve uygulanabilir çözümler sunmuştur. Bununla birlikte, söz konusu ilerlemelere rağmen, problem yapısının karmaşıklığı ve değişken koşullara anlık uyum sağlama gerekliliği gibi bazı temel zorluklar varlığını sürdürmektedir.

Bu nedenle, gelecekteki araştırmaların yönelmesi gereken başlıca alanlar arasında, gerçek zamanlı verilerin IoT cihazları aracılığıyla sistemlere entegre edilmesi ve bu sayede daha dinamik ve ölçeklenebilir çözümler geliştirilmesi yer almaktadır. Aynı zamanda, daha uyarlanabilir ve açıklanabilir model yapılarına ihtiyaç duyulmakta; bu doğrultuda, RL ile sürü zekâsı tekniklerini birleştiren hibrit AI yaklaşımlarının keşfi önem kazanmaktadır.

Gelecek çalışmalar ayrıca, üretken AI teknolojilerinin ÇDARP için sunduğu potansiyelleri araştırabilir. Özellikle Üretici Çekişmeli Ağlar (GAN) ve Varyasyonel Otokodlayıcılar (VAE) gibi üretken modeller, çeşitli trafik koşulları, müşteri talepleri ve araç kapasitesi senaryolarını simüle ederek rota planlamasında dayanıklılığı ve model genellemesini artırabilir. Üretken modellerin RL, DTBKA vb. algoritmalar ile entegre edilmesi, BT'nin güçlü keşif yeteneklerini dikkat mekanizmalarının öğrenme kapasitesiyle birleştirerek, dinamik rota senaryolarına gerçek zamanlı uyum sağlamayı mümkün kılabilir. Ayrıca, Transformer mimarisinin GNN'ler ile birlikte kullanılması, mekânsal ilişkilerin daha derinlemesine modellenmesini sağlayarak büyük ölçekli ÇDARP problemlerinin çözümünde önemli avantajlar sunabilir.

Bu gelişmeler ışığında, DTBKA ile entegre edilen bu ileri düzey yöntemler, sistemlerin ölçeklenebilirliğini ve lojistik operasyonlardaki verimliliği önemli ölçüde artırma potansiyeline sahiptir. Özellikle karmaşık dağıtım ihtiyaçlarına sahip endüstriler için yüksek performanslı ve esnek çözümler geliştirilmesine olanak tanıyabilir. Sonuç olarak, ÇDARP alanı, hem akademik hem de pratik düzeyde araştırmaya açık ve aktif bir saha olmaya devam etmektedir. Artan bilgisayar gücü ve gelişen optimizasyon teknikleri sayesinde, lojistik firmalarının

operasyonlarını optimize etmelerine yönelik daha etkin, gerçek zamanlı ve uygulanabilir çözümlerin geliştirilmesi beklenmektedir.

Sonuç ve Öneriler

Bu tez çalışmasında, ÇDARP için gerçek dünya GPS verilerine dayalı, ölçeklenebilir, dinamik ve uyarlanabilir AI tabanlı çözümler geliştirilmiştir. Tezin ilk aşamasında, GPS konumlandırma doğruluğunu artırmaya yönelik olarak topluluk ve DL tabanlı hibrit model yaklaşımları test edilmiştir. Topluluk modellerde özellikle kNN-DTR kombinasyonu öne çıkmış, yığın yöntemi, birden fazla modelin güçlü yönlerini bir araya getirerek genelleme kabiliyeti yüksek sonuçlar üretmiştir. Oylama yöntemi daha basit ve hesaplama açısından avantajlı olmasına rağmen, doğruluk bakımından yığın modeline göre daha sınırlı kalmıştır. Bu aşamada, 1B GPS verileri, mekânsal ilişkileri daha iyi yakalamak amacıyla 4B tensör yapısına dönüştürülerek CNN tabanlı modellere giriş olarak verilmiştir. Tez kapsamında geliştirilen DL tabanlı hibrit modeller, iki ana grupta incelenmiştir: GPSCNN'ler ve GPSLSTM. GPSCNN'ler grubu; önceden eğitilmiş CNN modelleri (Xception, VGG16, VGG19, AlexNet) ile özel tasarlanmış CNN yapılarından (CNN1, CNN2, CNN3) oluşmaktadır. Bu modeller, mekânsal özellikleri çıkarmakta yüksek performans sergilemiştir. Öte yandan GPSLSTM modeli, LSTM katmanı ile ardışık yapıları işleyip SVR meta modeli aracılığıyla tahmin gücünü artırmıştır. 4B tensör dönüşümü, özellikle CNN tabanlı modellerde fayda sağlarken, bazı senaryolarda daha basit 1B yapılarla çalışan modellerin (örneğin MLPRegressor, LR, XGB) daha hızlı ve verimli sonuçlar ürettiği gözlemlenmiştir. Bu bulgu, veri dönüşüm stratejilerinin, kullanılan veri setinin yapısına ve model mimarisine göre dikkatle seçilmesi gerektiğini ortaya koymuştur. Ayrıca, kullanılan topluluk modeller (örneğin KNR-DTR-LR-XGB-RFR-MLPRegressor gibi) farklı algoritmaların çıktılarından faydalananarak genel doğruluğu artırmıştır. Tezin ikinci aşamasında, hem klasik ML hem de DL tabanlı modellerin GPS konum verileri üzerindeki performansları karşılaştırılmış ve veri setinin yapısal özelliklerine en uygun modellerin belirlenmesi hedeflenmiştir. Bu aşamada, yüksek boyutluluk, eksik veri ve GPS verisinin düzensizliği gibi problemlerle başa çıkabilecek modeller (XGB, CNN, SepCNN, WdCNN gibi) tercih edilmiştir. CNN tabanlı modellerin mekânsal ilişkileri yakalama gücü ön plana çıkarken, XGB modeli, büyük ve heterojen veri setlerinde doğruluk avantajı ile en etkili klasik yöntemlerden biri olarak öne çıkmıştır. Sonuç olarak, bu çalışmada geliştirilen sistematik yaklaşım, hem 1B hem de 4B formatındaki GPS verilerinin etkin şekilde işlenmesini sağlamış; stacking, CNN ve XGB gibi güçlü modellerin kullanımı ile yüksek doğrulukta konum tahmini elde edilmiştir. Tezin bu iki aşaması, hem veri ön işleme stratejilerinin hem de model seçiminin doğruluk ve verimlilik üzerindeki kritik rolünü açıkça ortaya koymuştur. Tezin üçüncü aşamasında, İstanbul'daki bir

dağıtım şirketinden alınan büyük ölçekli GPS verileri üzerinde AI modellerinin mekânsal doğruluk artırmadaki performansı analiz edilmiştir. Veri seti, ML algoritmaları (KNR, XGB, LR, DTR, MLPR vb.) ve hibrit modellerin rotalara ilişkin uygunluk değerleri hesaplanmıştır. En başarılı LR modeli, meta-sezgisel algoritmalarla başlangıç çözümü olarak verilmiştir. Bu hibrit yapı sayesinde, başlangıç çözümü iyileştirilmiş ve meta-sezgisel algoritmaların (BT, TA, MBT) daha düşük maliyetlerle optimizasyon gerçekleştirmesi sağlanmıştır. ARP'nin çözümüne yönelik sezgisel ve meta-sezgisel algoritmalar kullanılarak şirket verileri üzerinde uygulamalar geliştirilmiştir. Geniş GPS veri seti paralel işlenerek işlem süresi optimize edilmiştir. "ThreadPoolExecutor" sınıfı ile CPU kaynakları etkili biçimde kullanılarak "out of memory" sorunlarının önüne geçilmiştir. GPS verileri, araç plakası ve tarih bilgisine göre eşleştirilmiş, geçmiş sevkiyatlara dayalı zaman farkları hesaplanarak rotalar oluşturulmuştur. Başlangıç çözümü üç farklı yaklaşımla ele alınmıştır. Bu çözümler üzerinde MBT, BT ve TA algoritmaları çalıştırılarak maliyet fonksiyonu minimize edilmiştir. Zaman-mesafe matrisi iki farklı yöntemle (OSRM API ve GPS verisi) oluşturulmuş, GPS verisine dayalı asimetrik yapı trafik, hava durumu ve yol koşulları gibi faktörlerin etkisini gerçekçi şekilde yansıtmıştır. DTBKA modeli, yapılan karşılaştırmalarda en düşük MAE ve MSE değerlerine ulaşarak en verimli rotaları üretmiştir. BKA en zayıf performansı gösterirken; DQN ve Dikkat tabanlı modeller daha başarılı olsa da DTBKA tarafından geride bırakılmıştır. DTBKA, BKA'nın keşif gücünü Dikkat tabanlı modelin odaklanma yeteneği ile birleştirerek daha kısa, etkili ve genellenebilir rotalar üretmiştir. Bu bağlamda, meta-sezgisel yöntemlerin DL ile entegrasyonu, ÇDARP için üstün performans sergileyen hibrit çözümler sunmuştur. RL algoritmalarının genelleme sorunları, DL modellerinin hesaplama yükü, meta-sezgisellerin yerel optimumlara takılma riski gibi sınırlamalar, DTBKA gibi hibrit yapıların önemini ortaya koymuştur. Ayrıca, GNN-LSTM-RL modeli mekânsal-zamansal ilişkileri öğrenmede öne çıkarken, BME modeli düşük hesaplama maliyeti ile sade problemler için uygun bir alternatif sunmuştur. GNN-LSTM-RL modeli ise veri değişkenliğine hızlı uyum sağlama yeteneği ile dikkat çekmiştir. K-Fold analiz sonuçlarına göre DTBKA ve Dikkat tabanlı mekanizma en istikrarlı genelleme performansını sergilemiş; BKA ve DQN modelleri ise düşük doğruluk ve yüksek varyans ile sınırlı kalmıştır. BME, GR ve GNN-LSTM-RL gibi modeller stabil ancak bazı senaryolarda dalgalanan sonuçlar üretmiştir. Kaynak kullanımı, eğitim süresi ve genelleme başarısı açısından DTBKA en dengeli ve başarılı çözüm olarak öne çıkmıştır. Son aşamada, HLS uyumlu modellerin FPGA ve gömülü sistemlerde gerçek zamanlı uygulanabilirliği hedeflenmiştir. LSTFormer modeli, sıralı listeleme ve Transformer mimarisini birleştirerek yüksek doğruluk, kapasite kullanımı ve genelleme başarısı sağlamıştır. Diğer modeller olan ALTFormer, ASTFormer, MetaDRL tabanlı yapılar, değişken müşteri talepleri gibi senaryolara özel uyarlanabilir

özmler retmiřtir. NEAT tabanlı NRL modeli ise hafif yapısı ve stabil sonuçları ile operasyonel sahalarda uygulanabilirliğini kanıtlamıřtır. Sonuç olarak, bu tez alıřması klasik yöntemlerin ötesine geçerek AI, DL, RL ve EA yaklaşımlarını donanım uyumlu ve gerçek zamanlı sistemler için ölçeklenebilir bir özüm çerçevesi altında bütünleřtirmiřtir. Fitness fonksiyonu, toplam seyahat süresini araç kapasitesi, ceza fonksiyonu ve mesafe üzerinden optimize etmeye dayanmıřtır. Gerçek verilerle yapılan deneyler, DTBKA ve LST-Former gibi hibrit modellerin DARP'nin karmařıklık düzeyi yüksek versiyonları için en başarılı özümleri sunduğunu göstermiřtir. Bu alıřma, lojistik alanında AI temelli optimizasyon özümlerinin geliştirilmesine katkı saęlamıř ve gelecekteki donanım tabanlı uygulamalar için temel oluřturmuřtur.

Gelecekteki alıřmalar, GPS verileri ve müşteri taleplerinin anlık olarak deęiřebildięi lojistik sistemlerde, IoT cihazları ve mobil veri akıřlarını entegre ederek gerçek zamanlı ve adaptif rota optimizasyonu geliřtirmeye odaklanmalıdır. DTBKA ve LST-MetaDRL gibi hibrit modellerin eęitim süreçlerinde gözlemlenen dalgalanmaların azaltılması için, hiperparametre optimizasyonu ve Kademeli Öğrenme gibi öğrenme stratejilerinin uygulanmasıyla model eęitim stabilitesi artırılmalıdır. HLS uyumlu modellerin FPGA gibi gömülü sistemlerde alıřtırılabilirliği teorik olarak gösterilmiř olsa da, gelecekte Vivado HLS gibi araçlarla bu modellerin donanım ortamında prototiplerinin geliştirilmesi ve gecikme ile enerji verimlilięi gibi metriklerin ölçülmesi önerilmektedir. Ayrıca, GNN-RL ve BME gibi mevcut başarılı modellerin yanı sıra GPT-4, Perceiver veya Swin Transformer gibi yeni nesil AI mimarileri ile karřılařtırmalı analizler yapılarak modelleme kabiliyetlerinin karřılařtırılması faydalı olacaktır. Son olarak, GPS verilerinin 1B'den 4B'ye dönüşümü gibi zaman alıcı ve veri setine duyarlı işlemler için AutoML ve NAS gibi otomatik özellik mühendisliği yöntemlerinin kullanılması, veri ön işleme sürecinde dinamik ve veriye özgü stratejilerin geliştirilmesini saęlayacaktır.

KAYNAKÇA

- Agárdi, A., Kovács L., & Bányai T. (2019). Optimization Of Multi-Depot Periodic Vehicle Routing Problem With Time Window. *Academic Journal of Manufacturing Engineering*, 17(4).
- Almasan, P., Suárez-Varela, J., Rusek, K., Barlet-Ros, P., & Cabellos-Aparicio, A. (2022). Deep reinforcement learning meets graph neural networks: Exploring a routing optimization use case. *Computer Communications*, 196, 184-194.
- Alqahtani H. & Kumar G. (2024). Efficient Routing Strategies for Electric and Flying Vehicles: A Comprehensive Hybrid Metaheuristic Review. *IEEE Transactions on Intelligent Vehicles*.
- Altay, O. (2022). Chaotic slime mould optimization algorithm for global optimization. *Artificial Intelligence Review*, 55(5), 3979-4040.
- Alweshah M., Almiani M., Almansour N., Al Khalaileh S., Aldabbas H., Alomoush W., & Alshareef A. (2022). Vehicle routing problems based on Harris Hawks optimization. *Journal of Big Data*, 9(1):42.
- Anonymous. GitHub, 2018. project-4-GADSI. https://github.com/rylativity/identifying_wnv_locations-raw_repo- (2022).
- Anonymous. Kaggle, 2018. California Housing Dataset - ML Crash Course. https://www.kaggle.com/datasets/vikramtiwari/california-housing-dataset-ML-crash-course?resource=download&select=california_housing_train.csv (2022).
- Anonymous. Mobiliz, <https://mobiliz.com.tr/> (2023).
- Anonymous. OSRM API Documentation, <https://project-osrm.org/docs/v5.5.1/api/#general-options> (2023).
- Anuar, W. K., Lee, L. S., Seow, H. V., & Pickl, S. (2022). A multi-depot dynamic vehicle routing problem with stochastic road capacity: An MDP model and dynamic policy for post-decision state rollout algorithm in reinforcement learning. *Mathematics*, 10(15), 2699.
- Ara, S., Akib, M. M. M. K., Oion, M. S. R., Shohel, M. H. R., Ridoy, M. N. F., Kabita, F. A., & Shahiduzzaman, M. (2023, December). Vehicle Routing Problem Solving Using Reinforcement Learning. In 2023 26th International Conference on Computer and Information Technology (ICCIT) (pp. 1-6). IEEE.
- Arishi, A., & Krishnan, K. (2023). A multi-agent deep reinforcement learning approach for solving the multi-depot vehicle routing problem. *Journal of Management Analytics*, 10(3), 493-515.
- Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). A brief survey of deep reinforcement learning. *arXiv preprint arXiv:1708.05866*.
- Arun, M., (2021). Spatio-temporal data analysis using deep learning. Medium. <https://morgenrot.medium.com/spatio-temporal-data-analysis-using-deep-learning-fadbcd79d224> (2023).

- Asghari M, & Al-e SMJM (2021) Green vehicle routing problem: A state-of-the-art review. *International Journal of Production Economics* 231:107899.
- Aydin, T., & Erdem, E. (2023). Novel Deep Hybrid and Ensemble Algorithms for Improving GPS Navigation Positioning Accuracy. *IEEE Access*, 11, 53518-53530.
- Ayyıldız E, Taşkın A, Yıldız A, & Özkan C (2022) Artificial neural networks integrated mixed integer mathematical model for multi-fleet heterogeneous time-dependent cash in transit problem with time windows. *Neural Computing and Applications* 34(24): 21891-21909.
- Azevedo, B. F., Rocha, A. M. A., & Pereira, A. I. (2024). Hybrid approaches to optimization and machine learning methods: a systematic literature review. *Machine Learning*, 113(7), 4055-4097.
- Bai R, Chen X, Chen ZL, Cui T, Gong S, He W et al (2023) Analytics and machine learning in vehicle routing research. *International Journal of Production Research* 61(1):4-30.
- Bairathi D & Gopalani D (2021) An improved salp swarm algorithm for complex multi-modal problems. *Soft Computing* 25(15):10441-10465.
- Bairathi D & Gopalani D (2021) Numerical optimization and feed-forward neural networks training using an improved optimization algorithm: multiple leader salp swarm algorithm. *Evolution Intelligence* 14(3):1233-1249.
- Belbag, S., Çimen, M., Tarim, S., & Tas, A. (2009). A research on corporate Enterprise Resource Planning (ERP) systems used for supermarket supply chain inventory management in Turkey. *European Journal of Scientific Research*, 38(3), 486-499.
- Belhaiza S, M'Hallah R, Ben Brahim G, & Laporte G (2019) Three multi-start data-driven Evolution heuristics for the vehicle routing problem with multiple time windows. *Journal of Heuristics* 25(3):485-515.
- Cai, J., Zhang, X., Lin, Q., Dong, L., Chen, W., & Ming, Z. (2024, June). Deep Reinforcement Learning for Solving the Vehicle Routing Problem in Practical Logistics. In *2024 IEEE Congress on Evolution Computation (CEC)* (pp. 1-8). IEEE.
- Castillo C, Panadero J, Alvarez-Palau E J, & Juan AA (2024) Towards greener city logistics: an application of agile routing algorithms to optimize the distribution of micro-hubs in Barcelona. *European Transport Research Review* 16(1):44.
- Chaabani A, Bechikh S, & Said L B (2018) A new co- Evolution decomposition-based algorithm for bi-level combinatorial optimization. *Applied Intelligence* 48:2847-2872.
- Chaabani, A., & Said, L. B. (2019). Transfer of learning with the co-evolutionary decomposition-based algorithm-II: a realization on the bi-level production-distribution planning system. *Applied Intelligence*, 49, 963-982.
- Chakraborty, P., Nama, S., & Saha, A. K. (2023). RETRACTED ARTICLE: A hybrid slime mould algorithm for global optimization. *Multimedia Tools and Applications*, 82(15), 22441-22467.
- Chen, B., Zhu, D., Wang, Y., & Zhang, P. (2022). An approach to combine the power of deep reinforcement learning with a graph neural network for routing optimization. *Electronics*, 11(3), 368.
- Chen, J., & Luo, J. (2023, September). Enhancing Vehicle Routing Solutions Through Attention-Based Deep Reinforcement Learning. In *2023 5th International Conference on Data-driven Optimization of Complex Systems (DOCS)* (pp. 1-9). IEEE.

- Chen, R. M., Shen, Y. M., & Hong, W. Z. (2019). Neural-like encoding particle swarm optimization for periodic vehicle routing problems. *Expert Systems with Applications*, 138, 112833.
- Chen, W., Men, Y., Fuster, N., Osorio, C., & Juan, A. A. (2024). Artificial intelligence in logistics optimization with sustainable criteria: A review. *Sustainability*, 16(21), 9145.
- Chen, X., Ulmer, M. W., & Thomas, B. W. (2022). Deep Q-learning for same-day delivery with vehicles and drones. *European Journal of Operational Research*, 298(3), 939-952.
- Chen, X., Yan, H. F., Zheng, Y. J., & Karatas, M. (2023). Integration of machine learning prediction and heuristic optimization for mask delivery in COVID-19. *Swarm and Evolutionary Computation*, 76, 101208.
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1251-1258).
- Choudhry, A., & Qian, S. (2024). Quantification of truck accessibility in urban last-mile deliveries using GPS probe data. *Transportation Research Part E: Logistics and Transportation Review*, 186, 103536.
- Chu, H., Chai, Y., Liu, Y., & Sun, H. (2014, May). A novel E-Invoice Framework towards data-oriented taxation system. In *Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD)* (pp. 242-246). IEEE.
- Correll, R., Weinberg, S. J., Sanches, F., Ide, T., & Suzuki, T. (2022). Reinforcement Learning for Multi-Truck Vehicle Routing Problems. *arXiv preprint arXiv:2211.17078*.
- Darvish, M., Archetti, C., Coelho, L. C., & Speranza, M. G. (2019). Flexible two-echelon location routing problem. *European Journal of Operational Research*, 277(3), 1124-1136.
- de Oliveira, F. B., Enayatifar, R., Sadaei, H. J., Guimarães, F. G., & Potvin, J. Y. (2016). A cooperative coevolutionary algorithm for the multi-depot vehicle routing problem. *Expert Systems with Applications*, 43, 117-130.
- Delamer, J. A., Cupido, K., & Givigi, S. (2024, April). Graph Neural Network to Solve the Dynamic Emergency Vehicles Routing Problems. In *2024 IEEE International Systems Conference (SysCon)* (pp. 1-8). IEEE.
- Dewi, S. K., & Utama, D. M. (2021). A new hybrid whale optimization algorithm for green vehicle routing problem. *Systems Science & Control Engineering*, 9(1), 61-72.
- Dokeroglu, T., Canturk, D., & Kucukyilmaz, T. (2024). A survey on pioneering metaheuristic algorithms between 2019 and 2024. *arXiv preprint arXiv:2501.14769*.
- Dorigo, M., & Gambardella, L. M. (1997). Ant colonies for the travelling salesman problem. *biosystems*, 43(2), 73-81.
- Du, J., Li, X., Yu, L., Dan, R., & Zhou, J. (2017). Multi-depot vehicle routing problem for hazardous materials transportation: A fuzzy bilevel programming. *Information sciences*, 399, 201-218.
- El Midaoui, M., Qbadou, M., & Mansouri, K. (2021). Logistics chain optimization and scheduling of hospital pharmacy drugs using genetic algorithms: Morocco case. *International Journal of Web-Based Learning and Teaching Technologies (IJWLTT)*, 16(2), 54-64.

- EL YADARI, M., JAWAB, F., & MOUFAD, I. (2024). A Supervised Learning and Metaheuristic Approach for Energy-Efficient Vehicle Routing Problem: Model Development and Empirical Case Validation. Available at SSRN 5072833.
- Elmasri, M., Labbe, A., Larocque, D., & Charlin, L. (2023). Predictive inference for travel time on transportation networks. *The Annals of Applied Statistics*, 17(4), 2796-2820.
- Erdem, E., Aydın, T., and Erkeyman B. (under review a). Advances in AI-Based Solutions for the Multi-Depot Vehicle Routing Problem: A Review of Recent Trends and Future Directions. *Archives of Computational Methods in Engineering*.
- Erdem, E., Aydın, T., and Erkeyman B. (under review c). AI-Driven Optimization for Scalable MDVRP High-Level Synthesis, Transformer, Deep RL, and Evolutionary Approaches for Last-Mile Split Delivery with Multiple Trips. *Engineering Science and Technology, an International Journal*.
- Erdem, E., Aydın, T., and Erkeyman B. (under review d). AI-Infused Evolutionary Optimization for Multi-Depot Vehicle Routing: A Case Study of Beverage Distribution. *The Journal of Supercomputing*.
- Erdem, E., Aydın, T., and Erkeyman B. (under review e). Enhancing Location Accuracy in Urban Logistics Using Hybrid Machine Learning and Deep Learning Models on Large-Scale Global Positioning System Data. *Connection Science*.
- Erdem, E., Topan E., Aydın, T., and Erkeyman B. (under review b). AI-Assisted Optimization of Multi-Depot Vehicle Routing with Heterogeneous Fleet, Split Deliveries, and Multiple Trips using GPS Data: A Real-World Case. *Concurrency and Computation: Practice and Experience*.
- Fagbola TM & Thakur SC (2019) TERMHIGEN – A Hybrid Metaheuristic Technique for Solving Large-Scale Vehicle Routing Problem with Time Windows. *International Journal of Engineering Research and Technology* 12 (2):180 – 195.
- Fellek, G., Farid, A., Gebreyesus, G., Fujimura, S., & Yoshie, O. (2023, July). Deep graph representation learning to solve vehicle routing problem. In *2023 International Conference on Machine Learning and Cybernetics (ICMLC)* (pp. 172-180). IEEE.
- Feng L, Huang Y, Tsang IW, Gupta A, Tang K, Tan KC, & Ong YS (2020) Towards faster vehicle routing by transferring knowledge from customer representation. *IEEE Transactions on Intelligent Transportation Systems* 23(2):952-965.
- Gao, Z. M., Zhao, J., Yang, Y., & Tian, X. J. (2020, August). The hybrid grey wolf optimization-slime mould algorithm. In *Journal of Physics: Conference Series* (Vol. 1617, No. 1, p. 012034). IOP Publishing.
- Ghannadi, P., & Kourehli, S. S. (2022). Efficiency of the slime mold algorithm for damage detection of large-scale structures. *The Structural Design of Tall and Special Buildings*, 31(14), e1967.
- Ghazikhani, A., Davoodipoor, S., Fathollahi-Fard, A. M., Gheibi, M., & Moezzi, R. (2024). Robust Truck Transit Time Prediction through GPS Data and Regression Algorithms in Mixed Traffic Scenarios. *Mathematics*, 12(13), 2004.
- Gruler, A., Fikar, C., Juan, A. A., Hirsch, P., & Contreras-Bolton, C. (2017). Supporting multi-depot and stochastic waste collection management in clustered urban areas via simulation–optimization. *Journal of simulation*, 11(1), 11-19.
- Gu, Z., Zhu, Y., Wang, Y., Du, X., Guizani, M., & Tian, Z. (2022). Applying artificial bee colony algorithm to the multidepot vehicle routing problem. *Software: Practice and Experience*, 52(3), 756-771.

- Gupta, A., Ghosh, S., & Dhara, A. (2022, January). Deep reinforcement learning algorithm for fast solutions to vehicle routing problem with time-windows. In Proceedings of the 5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD) (pp. 236-240).
- Han, X. (2023). Path planning algorithm for the multiple depot vehicle routing problem based on parallel clustering. *Scientific Programming*, 2023(1), 7588595.
- Hao, H., Yao, E., Chen, R., Pan, L., & Wang, Y. (2024). Dynamic Passenger Route Guidance in the Multimodal Transit System With Graph Representation and Attention Based Deep Reinforcement Learning. *IEEE Transactions on Intelligent Transportation Systems*.
- He, Q., Wang, Y., Wang, X., Xu, W., Li, F., Yang, K., & Ma, L. (2023). Routing optimization with deep reinforcement learning in knowledge defined networking. *IEEE Transactions on Mobile Computing*, 23(2), 1444-1455.
- Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527-1554.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press google schola, 2, 29-41.
- Huang, S. H., Huang, Y. H., & Chiu, L. C. (2024). Solve the medical specimen collection problem by Ant Colony Optimization. *International Journal of Systems Science: Operations & Logistics*, 11(1), 2374968.
- Hughes, S., Moreno, S., Yushimito, W. F., & Huerta-Cánepa, G. (2019). Evaluation of machine learning methodologies to predict stop delivery times from GPS data. *Transportation Research Part C: Emerging Technologies*, 109, 289-304.
- Huo, L. (2021). [Retracted] Intelligent Recognition Method of Vehicle Path with Time Window Based on Genetic Algorithm. *Security and Communication Networks*, 2021(1), 3614291.
- Hutter F, Kotthoff L, & Vanschoren J (2019) *Automated machine learning: methods, systems, challenges* (p. 219). Springer Nature.
- Jabir, E., Panicker, V. V., & Sridharan, R. (2017). Design and development of a hybrid ant colony-variable neighbourhood search algorithm for a multi-depot green vehicle routing problem. *Transportation Research Part D: Transport and Environment*, 57, 422-457.
- Jafarian, A., Asgari, N., Mohri, S. S., Fatemi-Sadr, E., & Farahani, R. Z. (2019). The inventory-routing problem subject to vehicle failure. *Transportation Research Part E: Logistics and Transportation Review*, 126, 254-294.
- Jia, H., Zhang, W., Zheng, R., Wang, S., Leng, X., & Cao, N. (2022). Ensemble mutation slime mould algorithm with restart mechanism for feature selection. *International Journal of Intelligent Systems*, 37(3), 2335-2370.
- Jiansheng L, Wenyue T, Hai J, & Gong Y (2020) A hybrid optimization approach for the heterogeneous vehicle routing problem with multiple depots cooperative operation. *High Technology Letters* 26(1):108-117.
- Jin, Y., Yan, X., Liu, S., & Wang, X. (2024). A Unified Framework for Combinatorial Optimization Based on Graph Neural Networks. *arXiv preprint arXiv:2406.13125*.

- Joe, W., & Lau, H. C. (2020, June). Deep reinforcement learning approach to solve dynamic vehicle routing problem with stochastic customers. In *Proceedings of the international conference on automated planning and scheduling* (Vol. 30, pp. 394-402).
- Kachitvichyanukul, V., Sombuntham, P., & Kunnapapdeelert, S. (2015). Two solution representations for solving multi-depot vehicle routing problem with multiple pickup and delivery requests via PSO. *Computers & Industrial Engineering*, 89, 125-136.
- Karakatič, S. (2021). Optimizing nonlinear charging times of electric vehicle routing with genetic algorithm. *Expert Systems with Applications*, 164, 114039.
- Karakatič, S., & Podgorelec, V. (2015). A survey of genetic algorithms for solving multi depot vehicle routing problem. *Applied Soft Computing*, 27, 519-532.
- Kasparavičiūtė, G., & Ludvigsen, M. (2023, June). Genetic Algorithm-based AUV Mission Optimisation With Energy and Priority Constraints. In *OCEANS 2023-Limerick* (pp. 1-8). IEEE.
- Kaveh, A., Hamedani, K. B., & Kamalinejad, M. (2022). Improved slime mould algorithm with elitist strategy and its application to structural optimization with natural frequency constraints. *Computers & Structures*, 264, 106760.
- Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). IEEE.
- Khodashenas, M., Najafi, S. E., Kazemipoor, H., & Sobhani, M. (2023). Providing an integrated multi-depot vehicle routing problem model with simultaneous pickup and delivery and package layout under uncertainty with fuzzy-robust box optimization method. *Decision Making: Applications in Management and Engineering*, 6(2), 372-403.
- Kocatürk, F., Tütüncü, G. Y., & Salhi, S. (2021). The multi-depot heterogeneous VRP with backhauls: formulation and a hybrid VNS with GRAMPS meta-heuristic approach. *Annals of Operations Research*, 307(1), 277-302.
- Kodera, R., Sakai, T., & Hyodo, T. (2025). Development of a Delivery Time-Period Selection Model for Urban Freight Using GPS Data. *Smart Cities*, 8(1), 31.
- Kogan, M., Karns, J., & Desell, T. (2022, April). Self-adaptation of Neuroevolution Algorithms Using Reinforcement Learning. In *International Conference on the Applications of Evolution Computation (Part of EvoStar)* (pp. 452-467). Cham: Springer International Publishing.
- Kritzinger, S., Doerner, K. F., Hartl, R. F., Kiechle, G. Y., Stadler, H., & Manohar, S. S. (2012). Using traffic information for time-dependent vehicle routing. *Procedia-Social and Behavioral Sciences*, 39, 217-229.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90.
- Li, J., Dai, B. T., Niu, Y., Xiao, J., & Wu, Y. (2024). Multi-type attention for solving multi-depot vehicle routing problems. *IEEE Transactions on Intelligent Transportation Systems*.
- Li, K., Liu, T., Kumar, P. R., & Han, X. (2024). A reinforcement learning-based hyper-heuristic for AGV task assignment and route planning in parts-to-picker warehouses. *Transportation research part E: logistics and transportation review*, 185, 103518.
- Li, S., Chen, H., Wang, M., Heidari, A. A., & Mirjalili, S. (2020). Slime mould algorithm: A new method for stochastic optimization. *Future generation computer systems*, 111, 300-323.

- Li, X., Li, P., Zhao, Y., Zhang, L., & Dong, Y. (2021). A hybrid large neighborhood search algorithm for solving the multi depot UAV swarm routing problem. *IEEE Access*, 9, 104115-104126.
- Li, Y., Soleimani, H., & Zohal, M. (2019). An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. *Journal of cleaner production*, 227, 1161-1172.
- Liu, M., Wang, Z., & Li, J. (2022, May). A deep reinforcement learning algorithm for large-scale vehicle routing problems. In *International Conference on Electronic Information Technology (EIT 2022)* (Vol. 12254, pp. 824-829). SPIE.
- Londoño A, González W, Giraldo O, & Escobar J (2024) A hybrid heuristic approach for the multi-objective multi depot vehicle routing problem. *International Journal of Industrial Engineering Computations* 15(1):337-354.
- Lopes, A. I. G., Andrade, E. L. G., & de Meneses, V. P. (2022). Analysis of Strategic Logistics Tools in a Beverage Distributor. *Revista Interdisciplinar E Do Meio Ambiente (RIMA)*, 4(1), e144-e144.
- Luo, G., & Luo, J. (2024, February). Multi-objective adaptive dynamics attention model to solve multi-objective vehicle routing problem. In *Asian Conference on Machine Learning* (pp. 834-849). PMLR.
- Mak, S., Xu, L., Pearce, T., Ostroumov, M., & Brintrup, A. (2023). Fair collaborative vehicle routing: A deep multi-agent reinforcement learning approach. *Transportation Research Part C: Emerging Technologies*, 157, 104376.
- Maleki, A., Hemmati, V., Abazari, S. R., Aghsami, A., & Rabbani, M. (2024). Optimal distribution and waste management of COVID-19 vaccines from vaccination centers' satisfaction perspective—A fuzzy time window-based VRP. *Transportation Research Part E: Logistics and Transportation Review*, 183, 103454.
- Mańdziuk J (2018) New shades of the vehicle routing problem: Emerging problem formulations and computational intelligence solution methods. *IEEE Transactions on Emerging Topics in Computational Intelligence* 3(3):230-244.
- Máximo, V. R., Cordeau, J. F., & Nascimento, M. C. (2024). AILS-II: An adaptive iterated local search heuristic for the large-scale capacitated vehicle routing problem. *INFORMS Journal on Computing*, 36(4), 974-986.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540), 529-533.
- Mohammadzadeh, A., Masdari, M., Gharehchopogh, F. S., & Jafarian, A. (2021). Improved chaotic binary grey wolf optimization algorithm for workflow scheduling in green cloud computing. *Evolutionary Intelligence*, 14, 1997-2025.
- Moonsri, K., Sethanan, K., & Worasan, K. (2022). A novel enhanced differential evolution algorithm for outbound logistics of the poultry industry in Thailand. *Journal of Open Innovation: Technology, Market, and Complexity*, 8(1), 15.
- Mousavipour, S., & Hojjati, S. M. H. (2014). A particle swarm optimisation for time-dependent vehicle routing problem with an efficient travel time function. *International Journal of Operational Research*, 20(1), 109-120.
- Mozhdehi, A., Wang, Y., Sun, S., & Wang, X. (2025). SED2AM: Solving Multi-Trip Time-Dependent Vehicle Routing Problem using Deep Reinforcement Learning. *arXiv preprint arXiv:2503.04085*.

- Munir, A., Parasuraman, R., Ye, J., & Song, W. (2024, October). Route Planning for Electric Vehicles with Charging Constraints. In 2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall) (pp. 1-5). IEEE.
- Mustafa, H. (2011). Stock and customer order management system case study: Hargeisa Keyse Bushaaro, Somaliland.
- Nahar A & Das D (2023) MetaLearn: Optimizing routing heuristics with a hybrid meta-learning approach in vehicular ad-hoc networks. *Ad Hoc Networks* 138:102996.
- Noori, S., & Ghannadpour, S. F. (2012). High-level relay hybrid metaheuristic method for multi-depot vehicle routing problem with time windows. *Journal of Mathematical Modelling and Algorithms*, 11, 159-179.
- Noureldin, A., Osman, A., & El-Sheimy, N. (2003). A neuro-wavelet method for multi-sensor system integration for vehicular navigation. *Measurement science and technology*, 15(2), 404.
- Oudouar, F., Lazaar, M., & El Miloud, Z. (2020). A novel approach based on heuristics and a neural network to solve a capacitated location routing problem. *Simulation modelling practice and theory*, 100, 102064.
- Pan, W., & Liu, S. Q. (2023). Deep reinforcement learning for the dynamic and uncertain vehicle routing problem. *Applied Intelligence*, 53(1), 405-422.
- Peng, Y., Chen, G., Singh, H., & Zhang, M. (2018, July). NEAT for large-scale reinforcement learning through Evolution feature learning and policy gradient search. In *Proceedings of the genetic and Evolution computation conference* (pp. 490-497).
- Pérez-Rodríguez, R., & Hernández-Aguirre, A. (2019). A hybrid estimation of distribution algorithm for the vehicle routing problem with time windows. *Computers & Industrial Engineering*, 130, 75-96.
- Pessi, B. (2017). The Impact of Implementation of Electronic Purchase Invoice System on a Company on the Example of Hahle Group. Tallinn University of Technology.
- Pitakaso, R., Sethanan, K., Kumar, A., Tan, K. H., & Nanthasamroeng, N. (2024). An application of artificial intelligence for solving multi-visit scheduling and routing of multi-heterogeneous resources. *Annals of Operations Research*, 1-50.
- Prabu, U., Ravisasthiri, P., Sriram, R., Malarvizhi, N., & Amudhavel, J. (2019). EODVGA: an enhanced ODV based genetic algorithm for multi-depot vehicle routing problem. *EAI Endorsed Transactions on Scalable Information Systems*, 6(21).
- Purba, T. N., & Firdaus, D. (2021). Determination for Consumer Patterns in Beverage Product Sales Using the Frequent Pattern Growth Algorithm. *IJISCS (International Journal of Information System and Computer Science)*, 5(2), 84-92.
- Pytel, K. (2025). Fuzzy Guiding of Roulette Selection in Evolutionary Algorithms. *Technologies*, 13(2), 78.
- Quintero-Araujo, C. L., Guimarans, D., & Juan, A. A. (2021). A simheuristic algorithm for the capacitated location routing problem with stochastic demands. *Journal of Simulation*, 15(3), 217-234.
- Ramos, T. R. P., Gomes, M. I., & Barbosa-Póvoa, A. P. (2020). A new matheuristic approach for the multi-depot vehicle routing problem with inter-depot routes. *OR Spectrum*, 42, 75-110.
- Renaud, J., Laporte, G., & Boctor, F. F. (1996). A tabu search heuristic for the multi-depot vehicle routing problem. *Computers & Operations Research*, 23(3), 229-235.

- Ribeiro, R., Trifan, A., & Neves, A. J. (2024). A deep learning approach for transportation mode identification using a transformation of GPS trajectory data features into an image representation. *International Journal of Data Science and Analytics*, 1-10.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115, 211-252.
- Ryu, H., Yu, S., & Lee, K. Y. (2023, May). TI-former: A Time-Interval Prediction Transformer for Timestamped Sequences. In *2023 IEEE/ACIS 21st International Conference on Software Engineering Research, Management and Applications (SERA)* (pp. 319-325). IEEE.
- Sahar, G. S., Okba, K., Abdelkader, L., Amine, Y. M., Euler, R., Bounceur, A., & Hammoudeh, M. (2021). An optimized scalable multi-ant colony system for multi-depot vehicle routing problems using a reactive multi-agent system. *WSEAS Transactions on Systems*, 20, 249-259.
- Samsuddin, S., Othman, M. S., & Yusuf, L. M. (2020, May). Utilizing ant colony optimization and intelligent water drop for solving multi depot vehicle routing problem. In *IOP Conference Series: Materials Science and Engineering* (Vol. 864, No. 1, p. 012095). IOP Publishing.
- Serrano-Hernandez, A., de la Torre, R., Cadarso, L., & Faulin, J. (2021). Urban e-Grocery distribution design in Pamplona (Spain) applying an agent-based simulation model with horizontal cooperation scenarios. *Algorithms*, 14(1), 20.
- Shahbazian, R., Pugliese, L. D. P., Guerriero, F., & Macrina, G. (2024). Integrating Machine Learning Into Vehicle Routing Problem: Methods and Applications. *IEEE Access*.
- Shen, K., Shivgan, R., Medina, J., Dong, Z., & Rojas-Cessa, R. (2022). Multidepot drone path planning with collision avoidance. *IEEE Internet of Things Journal*, 9(17), 16297-16307.
- Sherinov, Z. (2011). An Evolutionary Multi-Objective Approach for Fuzzy Vehicle Routing Problem (Doctoral dissertation, Eastern Mediterranean University (EMU)).
- Shi, Y., Lv, L., Hu, F., & Han, Q. (2020). A heuristic solution method for multi-depot vehicle routing-based waste collection problems. *Applied Sciences*, 10(7), 2403.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Soeanu, A., Ray, S., Berger, J., Boukhtouta, A., & Debbabi, M. (2020). Multi-depot vehicle routing problem with risk mitigation: Model and solution algorithm. *Expert systems with applications*, 145, 113099.
- Soh, N. W., Zhang, A. N., Wan, F., & Xu, C. (2024, January). Urban Last Mile Delivery Data Mining for Performance Improvement. In *Proceedings of the 2024 7th International Conference on Computers in Management and Business* (pp. 86-93).
- Stodola, P. (2018). Using metaheuristics on the multi-depot vehicle routing problem with modified optimization criterion. *Algorithms*, 11(5), 74.
- Stodola, P. (2020). Hybrid ant colony optimization algorithm applied to the multi-depot vehicle routing problem. *Natural computing*, 19(2), 463-475.
- Stodola, P., & Mazal, J. (2016). Applying the ant colony optimisation algorithm to the capacitated multi-depot vehicle routing problem. *International Journal of Bio-Inspired Computation*, 8(4), 228-233.

- Stodola, P., & Nohel, J. (2022). Adaptive ant colony optimization with node clustering for the multidepot vehicle routing problem. *IEEE Transactions on Evolutionary Computation*, 27(6), 1866-1880.
- Su, Y., Zhang, S., & Zhang, C. (2024). A lightweight genetic algorithm with variable neighborhood search for multi-depot vehicle routing problem with time windows. *Applied Soft Computing*, 161, 111789.
- Sutton, R. S., & Barto, A. G. (1999). Reinforcement learning. *Journal of Cognitive Neuroscience*, 11(1), 126-134.
- Tatsis, V., Mena, C., Van Wassenhove, L. N., & Whicker, L. (2006). E-procurement in the Greek food and drink industry: drivers and impediments. *Journal of Purchasing and Supply management*, 12(2), 63-74.
- Tsehaie, N., 2023. Graph Neural Networks for Inland Waterway Ship Scheduling. MS Thesis, Delft University of Technology, Delft, Netherlands.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Wang, R., Keyantuo, P., Zeng, T., Sandoval, J., Vishwanath, A., Borhan, H., & Moura, S. (2024). Robust routing for a mixed fleet of heavy-duty trucks with pickup and delivery under energy consumption uncertainty. *Applied Energy*, 368, 123407.
- Wang, S., Lu, Z., Wei, L., Ji, G., & Yang, J. (2016). Fitness-scaling adaptive genetic algorithm with local search for solving the Multiple Depot Vehicle Routing Problem. *Simulation*, 92(7), 601-616.
- Wang, Y., & Chen, Z. (2022, September). A Deep Reinforcement Learning Algorithm Using A New Graph Transformer Model for Routing Problems. In *Proceedings of SAI Intelligent Systems Conference* (pp. 365-379). Cham: Springer International Publishing.
- Wang, Y., & Liang, X. (2025). Application of Reinforcement Learning Methods Combining Graph Neural Networks and Self-Attention Mechanisms in Supply Chain Route Optimization. *Sensors*, 25(3), 955.
- Wang, Y., Wang, L., Peng, Z., Chen, G., Cai, Z., & Xing, L. (2019). A multi ant system based hybrid heuristic algorithm for vehicle routing problem with service time customization. *Swarm and Evolutionary Computation*, 50, 100563.
- Wang, Y., Wei, Y., Wang, X., Wang, Z., & Wang, H. (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, 109922.
- Wang, Y., Zhe, J., Wang, X., Sun, Y., & Wang, H. (2022). Collaborative multidepot vehicle routing problem with dynamic customer demands and time windows. *Sustainability*, 14(11), 6709.
- Wu, H., & Gao, Y. (2023). An ant colony optimization based on local search for the vehicle routing problem with simultaneous pickup–delivery and time window. *Applied Soft Computing*, 139, 110203.
- Wu, R., Wang, R., Hao, J., Wu, Q., Wang, P., & Niyato, D. (2024). Multiobjective Vehicle Routing Optimization With Time Windows: A Hybrid Approach Using Deep Reinforcement Learning and NSGA-II. *IEEE Transactions on Intelligent Transportation Systems*.

- Xie, L., Li, H., & Luttmann, L. (2023). Formulating and solving integrated order batching and routing in multi-depot AGV-assisted mixed-shelves warehouses. *European Journal of Operational Research*, 307(2), 713-730.
- Xu, D., & Xiao, R. (2015). An improved genetic clustering algorithm for the multi-depot vehicle routing problem. *International Journal of Wireless and Mobile Computing*, 9(1), 1-7.
- Xu, Y., Fang, M., Chen, L., Xu, G., Du, Y., & Zhang, C. (2022). Deep reinforcement learning for solving the heterogeneous capacitated vehicle routing problem. *IEEE Trans. Cybern.*, 52(10), 11107-11120.
- Yan, D., Ou, B., Guan, Q., Zhu, Z., & Cao, H. (2025). Edge-Driven Multiple Trajectory Attention Model for Vehicle Routing Problems. *Applied Sciences*, 15(5), 2679.
- Yan, X., Huang, H., Hao, Z., & Wang, J. (2019). A graph-based fuzzy evolutionary algorithm for solving two-echelon vehicle routing problems. *IEEE Transactions on Evolutionary Computation*, 24(1), 129-141.
- Yang, J., Zhang, Y., Jin, T., Lei, Z., Todo, Y., & Gao, S. (2023). Maximum Lyapunov exponent-based multiple chaotic slime mold algorithm for real-world optimization. *Scientific Reports*, 13(1), 12744.
- Yang, W., Wang, D., Pang, W., Tan, A. H., & Zhou, Y. (2020). Goods consumed during transit in split delivery vehicle routing problems: Modeling and solution. *IEEE Access*, 8, 110336-110350.
- Yang, X. S. (2009, October). Firefly algorithms for multimodal optimization. In *International symposium on stochastic algorithms* (pp. 169-178). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Yang, X. S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver press.
- Yang, X., & Chen, Z. (2024, November). Dynamic Hybrid Temporal Attention With Attention on Attention for the Capacity Vehicle Routing Problem. In *2024 China Automation Congress (CAC)* (pp. 31-36). IEEE.
- Yang, X., Ke, L., & Chen, Z. (2023, March). A deep reinforcement learning framework with generalization performance for the large-scale capacitated vehicle routing problem. In *Second International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2022)* (Vol. 12597, pp. 816-825). SPIE.
- YAO, B., HU, P., YU, L., ZHANG, M., & GAO, J. (2015). Merged automobile parts maintenance delivery problem using an improved artificial bee colony algorithm.
- Yesodha, R., & Amudha, T. (2022). A bio-inspired approach: firefly algorithm for multi-depot vehicle routing problem with time windows. *Computer Communications*, 190, 48-56.
- Yoo, S., Kim, H., & Lee, J. (2024). Combining reinforcement learning with genetic algorithm for many-to-many route optimization of autonomous vehicles. *IEEE Access*, 12, 26931-26942.
- Yu, Z., Guo, Y., & Chen, Y. (2020, May). Learning trajectory routing with graph neural networks. In *Proceedings of the 5th International Conference on Big Data and Computing* (pp. 121-126).
- Zahedi, F., Kia, H., & Khalilzadeh, M. (2023). A hybrid metaheuristic approach for solving a bi-objective capacitated electric vehicle routing problem with time windows and partial recharging. *Journal of Advances in Management Research*, 20(4), 695-729.

- Zhang, K., Lin, X., & Li, M. (2022). Transformer-based reinforcement learning for pickup and delivery problems with late penalties. *IEEE Transactions on Intelligent Transportation Systems*, 23(12), 24649-24661.
- Zhang, X., Liu, G., Zhao, K., & Niu, Y. (2023). Improved salp swarm algorithm based on gravitational search and multi-leader search strategies. *AIMS Mathematics*, 8(3), 5099-5123.
- Zhang, X., Lyu, H., & Cheng, R. (2024). ASTNAT: an attention-based spatial-temporal non-autoregressive transformer network for vehicle trajectory prediction. *Neural Computing and Applications*, 1-16.
- Zhang, X., Yang, Y., Cai, J., Zhu, Q., Chen, W., & Lin, Q. (2024, June). Deep Reinforcement Learning-Based Multi-Agent Algorithm for Vehicle Routing Problem in Complex Logistics Scenarios. In *2024 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.
- Zhao, J., Mao, M., Zhao, X., & Zou, J. (2020). A hybrid of deep reinforcement learning and local search for the vehicle routing problems. *IEEE Transactions on Intelligent Transportation Systems*, 22(11), 7208-7218.
- Zheng, Z., Yao, S., Wang, Z., Tong, X., Yuan, M., & Tang, K. (2024). Dpn: Decoupling partition and navigation for neural solvers of min-max vehicle routing problems. *arXiv preprint arXiv:2405.17272*.
- Zhou G, Li X, Li D, & Bian J (2024) Learning-Based Optimization Algorithms for Routing Problems: Bibliometric Analysis and Literature Review. *IEEE Transactions on Intelligent Transportation Systems*.
- Žunić E, Đonko D, & Buza E (2020) An Adaptive Data-Driven Approach to Solve Real-World Vehicle Routing Problems in Logistics. *Complexity* 2020(1):7386701.
- Žunić, E., Hindija, H., Beširević, A., Hodžić, K., & Delalić, S. (2018, November). Improving performance of vehicle routing algorithms using GPS data. In *2018 14th Symposium on Neural Networks and Applications (NEUREL)* (pp. 1-4). IEEE.

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı:	Ebru ERDEM
Doğum tarihi:	
Doğum Yeri:	
Uyruğu:	
Adres:	
Tel:	
E-mail:	
Eğitim	
Lisans:	Atatürk Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
Yüksek lisans:	Atatürk Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı (2019)
Doktora:	Atatürk Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı (2025)
Yabancı Dil Bilgisi	
İngilizce:	İyi
Tezden Üretilmiş Yayınlar	
1. Aydın, T., & Erdem, E. (2023). Novel Deep Hybrid and Ensemble Algorithms for Improving GPS Navigation Positioning Accuracy. <i>IEEE Access</i> , 11, 53518-53530.	