Elektrikli Araçlar için Pekiştirmeli Öğrenme Tabanlı Rota Planlama

Özge ASLAN YILDIZ

DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Mayıs 2025

Reinforcement Learning Based Route Planning for Electric Vehicles

Özge ASLAN YILDIZ

DOCTORAL DISSERTATION

Department of Computer Engineering

May 2025

Elektrikli Araçlar için Pekiştirmeli Öğrenme Tabanlı Rota Planlama

Özge ASLAN YILDIZ

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Eğitim ve Öğretim Yönetmeliği Uyarınca
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Yazılımı Bilim Dalında
DOKTORA TEZİ
olarak hazırlanmıştır.

Danışman: Prof. Dr. Ahmet YAZICI
İkinci Danışman: Prof. Dr. İnci SARIÇİÇEK

Bu tez Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından 22AG040 Numaralı proje ile desteklenmiştir.

Mayıs 2025

ONAY

Bilgisayar Mühendisliği Anabilim Dalı Doktora öğrencisi **Özge ASLAN YILDIZ**'ın DOKTORA tezi olarak hazırladığı "**Elektrikli Araçlar için Pekiştirmeli Öğrenme Tabanlı Rota Planlama**" başlıklı bu çalışma, jürimizce lisansüstü yönetmeliğin ilgili maddeleri uyarınca değerlendirilerek oybirliği ile kabul edilmiştir.

Danışman : Prof. Dr. Ahmet YAZICI

İkinci Danışman : Prof. Dr. İnci SARIÇİÇEK

Doktora Tez Savunma Jürisi:

Üye: Prof. Dr. Ahmet YAZICI

Üye: Prof. Dr. Kemal ÖZKAN

Üye: Doç. Dr. Uğur YAYAN

Üye: Doç. Dr. Emre ÇİMEN

Üye: Dr. Öğr. Üyesi Altan ONAT

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ……..…….. tarih ve ……..……. sayılı kararıyla onaylanmıştır.

Prof. Dr. ....

Enstitü Müdürü

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Prof. Dr. Ahmet YAZICI ve Prof. Dr. İnci SARIÇİÇEK danışmanlığında hazırlamış olduğum "**Elektrikli Araçlar için Pekiştirmeli Öğrenme Tabanlı Rota Planlama**" başlıklı Doktora tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallarına uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 30/05/2025

Özge ASLAN YILDIZ

ÖZET

Kent nüfusunun hızlı artışı ve e-ticaret faaliyetlerindeki yükseliş, son kilometre teslimat süreçlerinde önemli zorlukları beraberinde getirmiştir. Bu bağlamda, çevre dostu bir alternatif olarak elektrikli araçların (EV) son kilometre teslimatta kullanımı ön plana çıkmaktadır. Elektrikli araçlar, hava kirliliğini azaltmanın yanı sıra enerji verimliliği ve sürdürülebilir şehir içi ulaşım sistemlerinin geliştirilmesine de önemli katkılar sağlamaktadır. Bu dinamikler içinde, Elektrikli Araç Rotalama Problemleri (EVRP), geleneksel Araç Rotalama Problemlerinin (VRP) yerini almaya başlamıştır. Klasik araç rotalamada yakıt ikmali göz ardı edilirken, elektrikli araç rotalama problemlerinde şarj istasyonlarının konumu ve uzun şarj süreleri dikkate alınmalıdır. Bu tez çalışmasında elektrikli araç rotalama problemlerinden biri olan Kapasiteli Elektrikli Araç Rotalama Problemi ve Topla Dağıt Elektrikli Araç Rotalama Problemi ele alınmaktadır. Kapasiteli Elektrikli Araç Rotalama Problemi (CEVRP) için pekiştirmeli öğrenme yöntemlerinden biri olan yeni bir Q-öğrenme algoritması önerilmiştir. Zaman Pencereli Kapasiteli Elektrikli Araç Rotalama Problemi (CEVRPTW) ve Zaman Pencereli Topla Dağıt Elektrikli Araç Rotalama Problemi (EVRP-PDP-TW) için derin pekiştirmeli öğrenme yöntemlerinden biri olan yeni bir Derin Q-Ağı (DQN) algoritması geliştirilmiştir. Bildiğimiz kadarıyla EVRP-PDP-TW problemi için pekiştirmeli öğrenme yaklaşımını kullanan ilk çalışma budur. Ayrıca çalışmada, Eskişehir Osmangazi Üniversitesi Meşelik Kampüsü'ndeki gerçek coğrafi konumlar esas alınarak pekiştirmeli öğrenme yöntemleri için ortam ihtiyacını karşılayacak iki yeni veri seti oluşturulmuştur. Kapasiteli elektrikli araç rotalama problemlerinde kullanılabilecek ESOGU-CEVRPTW veri seti ve topla dağıt elektrikli araç rotalama problemlerinde kullanılabilecek ESOGU-EVRP-PDP-TW veri seti için matematiksel model tarafından elde edilen optimal çözümler sunulmuştur. Geliştirilen algoritmaların performansı, önerilen özgün veri setlerindeki optimal sonuçlarla karşılaştırılmıştır. Deneysel sonuçlar, önerilen Q-öğrenme ve DQN algoritmalarının elektrikli araçların menzil kısıtları ve şarj gereksinimlerini etkin bir şekilde dikkate alarak optimal sonuçlara çok yakın sonuçları oldukça kısa sürede elde ettiğini göstermektedir. Ayrıca, derin pekiştirmeli öğrenme tabanlı algoritmaların eğitim süreci yalnızca bir defa gerçekleştirildiğinden, aynı dağılıma sahip yeni problemlere doğrudan uygulanabilmesi, geleneksel optimizasyon yöntemlerine göre büyük bir avantaj sağlamaktadır. Önerilen algoritmaların sağladığı bu avantajlar lojistik süreçlerinin verimliliği artırmaktadır.

Anahtar Kelimeler: Elektrikli araçlar, kapasiteli araç rotalama problemi, topla-dağıt problemi, zaman pencereleri, pekiştirmeli öğrenme, Q-Öğrenme, derin Q-Ağı

SUMMARY

The rapid increase in urban populations and the rise of e-commerce activities have introduced significant challenges in last-mile delivery processes. In this context, electric vehicles (EVs) have emerged as an environmentally friendly alternative for last-mile delivery operations. Electric vehicles not only reduce air pollution but also substantially contribute to improving energy efficiency and developing sustainable urban transportation systems. Within this framework, Electric Vehicle Routing Problems (EVRP) have started to replace conventional Vehicle Routing Problems (VRP). Unlike traditional vehicle routing, which typically overlooks refueling constraints, electric vehicle routing problems must consider the locations of charging stations and extended charging durations. This thesis addresses two specific variants of electric vehicle routing problems: the Capacitated Electric Vehicle Routing Problem (CEVRP) and the Electric Vehicle Routing Problem with Pickup and Delivery (EVRP-PDP). For the CEVRP, a novel Q-learning algorithm approaches is proposed. Additionally, for the Capacitated Electric Vehicle Routing Problem with Time Windows (CEVRPTW) and Pickup Delivery Electric Vehicle Routing Problem with Time Windows (EVRP-PDP-TW), a novel Deep Q-Network (DQN) algorithm is developed. To the best of our knowledge, this is the first study employing reinforcement learning methods specifically for the EVRP-PDP-TW. Furthermore, two new datasets are introduced based on real geographic locations from Eskişehir Osmangazi University Meşelik Campus. Optimal solutions obtained through mathematical model are presented for these datasets: the ESOGU-CEVRPTW dataset, applicable to capacitated electric vehicle routing problems, and the ESOGU-EVRP-PDP-TW dataset, applicable to pick up delivery electric vehicle routing problems. The performance of the proposed algorithms is benchmarked against the optimal solutions from the generated datasets. Experimental results indicate that the proposed Q-learning and DQN algorithms achieving near-optimal results within remarkably short computational times. Moreover, once trained, these deep reinforcement learning-based algorithms can be directly applied to new problems drawn from the same distribution without requiring additional training, offering significant advantages over traditional optimization methods. These advantages notably enhance the efficiency of logistics operations.

Keywords: Electric vehicles, capacitated vehicle routing problem, pickup delivery problem, time windows, reinforcement learning, Q-Learning, deep Q-Network

TEŞEKKÜR

Doktora eğitimim ve tez çalışmam boyunca yardımları, yönlendirmeleri ve katkılarıyla beni destekleyen değerli danışman hocalarım Sayın Prof. Dr. Ahmet YAZICI'ya ve Sayın Prof. Dr. İnci SARIÇİÇEK'e sonsuz saygılarımı ve teşekkürlerimi sunuyorum.

Tez çalışmam sırasında desteklerini ve önerilerini esirgemeyen Tez İzleme Komitesi üyeleri Sayın Prof. Dr. Kemal ÖZKAN'a ve Sayın Doç. Dr. Uğur YAYAN' a katkılarından dolayı teşekkür ederim.

Her zaman bana destek olan çok değerli sevgili aileme ve eşime sonsuz desteklerinden dolayı saygı ve sevgilerimi sunar teşekkür ederim.

Bu tez, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından 22AG040 Numaralı proje ile desteklenmiştir. Projeye verdiği destekten ötürü TÜBİTAK'a teşekkürlerimizi sunarız.

İÇİNDEKİLER

Sayfa

ÖZET ..... **vi**
SUMMARY ..... **vii**
TEŞEKKÜR ..... **viii**
İÇİNDEKİLER ..... **ix**
ŞEKİLLER DİZİNİ ..... **xi**
ÇİZELGELER DİZİNİ ..... **xiii**
SİMGELER VE KISALTMALAR DİZİNİ ..... **xiv**
1. GİRİŞ VE AMAÇ **1**
2. LİTERATÜR ARAŞTIRMASI **7**
2.1. Pekiştirmeli Öğrenme 7
2.1.1. Değer Tabanlı Yöntemler 12
2.1.2. Politika Tabanlı Yöntemler 13
2.1.3. Aktör-Kritik Yöntemler 14
2.2. Pekiştirmeli Öğrenme Tabanlı Kapasiteli Elektrikli Araç Rotalama Problemi 14
2.3. Pekiştirmeli Öğrenme Tabanlı Topla-Dağıt Elektrikli Araç Rotalama Problemi 24
2.4. Literatürdeki Mevcut Veri Setleri 30
3. MATERYAL VE YÖNTEM **34**
3.1. Materyal 34
3.1.1. Oluşturulan ESOGU-CEVRPTW veri seti 34
3.1.2. Oluşturulan ESOGU-EVRP-PDP-TW veri seti 41
3.2. Kapasiteli Elektrikli Araç Rotalama Problemi ve Matematiksel Modeli 43
3.3. Topla Dağıt Elektrikli Araç Rotalama Problemi ve Matematiksel Modeli 46
3.4. Q-Öğrenme ile Kapasiteli Elektrikli Araç Rotalama Problemi 48
3.5. Derin Pekiştirmeli Öğrenme ile Zaman Pencereli Kapasiteli Elektrikli Araç Rotalama Problemi 55
3.6. Derin Pekiştirmeli Öğrenme ile Zaman Pencereli Topla Dağıt Elektrikli Araç Rotalama Problemi 66
4. BULGULAR VE TARTIŞMA **68**
4.1. Matematiksel Model Sonuçları 68
4.2. CEVRP için Q-Öğrenme Algoritmasının Sonuçları 77

İÇİNDEKİLER

Sayfa

4.3. CEVRPTW için DQN Modelinin Sonuçları 82
4.4. EVRP-PDP-TW için DQN Modelinin Sonuçları 94
5.SONUÇ VE ÖNERİLER 104
KAYNAKLAR DİZİNİ 106

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

2.2. Pekiştirmeli Öğrenme Taksonomisi 10
2.3. Yıl bazında EVRP çalışmaları 17
2.4. Pekiştirmeli öğrenme yöntemini kullanan EVRP çalışmaları 19
2.5. Topla Dağıt Araç Rotalama Problemlerinin Sınıflandırılması (Parragh vd.,2008) 25
3.1. ESOGÜ haritası 35
3.2. ESOGÜ haritasının netedit ortamında gösterilmesi 35
3.3. ESOGÜ kampüsündeki rastgele müşteri dağılımı örneği 36
3.4. ESOGÜ kampüsündeki müşteri noktalarının k-means ile kümelenmesi 37
3.5. ESOGÜ kampüsündeki kümelenmiş müşteri dağılımı örneği 37
3.6. ESOGÜ kampüsündeki rastgele -kümelenmiş müşteri dağılımı örneği 38
3.7. Öklid mesafe ve gerçek mesafenin karşılaştırılması 39
3.8. ESOGU-CEVRPTW veri seti özeti 40
3.9. CEVRPTW için bir örnek çözüm 43
3.10. Zaman pencereli topla dağıt elektrikli araç rotalama problemi 47
3.11. Önerilen Q-Öğrenme algoritması 48
3.12. Oluşturulan Q Tablosu 49
3.13. Q-Öğrenme algoritmasının adım adım gösterimi 51
3.14. Önerilen DQN algoritması 58
3.15. Ana ve hedef ağın yapısı 60
3.16. Durum gösterimi 61
3.17. EVRP-PDP-TW problemi için durum gösterimi 66
4.1. ESOGU_C10 probleminin matematiksel model sonucunun SUMO'da gösterimi 71
4.2. Önerilen Q-Öğrenme algoritmasının öğrenme eğrisi 79
4.3. CPLEX çözücü ve Q-Öğrenme sonuçlarının toplam mesafe açısından karşılaştırılması 81
4.4. CPLEX çözücü ve Q-Öğrenme sonuçlarının toplam çalışma süresi açısından karşılaştırılması 81
4.5. Önerilen DQN algoritmasının eğitimdeki toplam mesafe eğrisi 84
4.6. Önerilen DQN algoritmasının eğitimdeki toplam ödül eğrisi 85
4.7. ESOGU_R5 için önerilen DQN ile üretilen birinci rota 87
4.8. ESOGU_R5 için önerilen DQN ile üretilen ikinci rota 88
4.9. CEVRP için CPLEX çözücü ve DQN sonuçlarının toplam mesafe açısından karşılaştırılması 89
4.10. CEVRP için CPLEX çözücü ve DQN sonuçlarının toplam çalışma süresi açısından karşılaştırılması 90
4.11. CEVRPTW için CPLEX çözücü ve DQN sonuçlarının toplam mesafe açısından karşılaştırılması 93

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

4.12. CEVRPTW için CPLEX çözücü ve DQN sonuçlarının toplam çalışma süresi açısından karşılaştırılması 94
4.13. Önerilen DQN yaklaşımının EVRP-PDP-TW probleminde eğitim süresince toplam mesafe eğrisi 96
4.14. Önerilen DQN yaklaşımının EVRP-PDP-TW probleminde eğitim süresince toplam ödül eğrisi 96
4.15. EVRP-PDP-TW için CPLEX çözücü ve DQN sonuçlarının toplam mesafe açısından karşılaştırılması 100
4.16. EVRP-PDP-TW için CPLEX çözücü ve DQN sonuçlarının çalışma süresi açısından karşılaştırılması 101
4.17. Küçük ölçekli modelin büyük problem örneklerine genelleme performansı 102
4.18. Büyük ölçekli modelin küçük problem örneklerine genelleme performansı 102

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

2.1. EVRP için farklı çözüm yaklaşımlarının karşılaştırılması 16
2.2. Literatürdeki EVRP çalışmaları 23
2.3. Literatürdeki EVRP-PDP çalışmaları 29
2.4. Literatürdeki EVRP veri setleri 31
2.5. Önerilen veri setinde tutulacak bilgiler 32
3.1. ESOGU-CEVRPTW veri setinden bir örnek 41
3.2. ESOGU-EVRP-PDP-TW veri setinden bir örnek 42
4.1. CEVRP'nin tam şarj stratejisine göre CPLEX'den elde edilen optimal sonuçları 68
4.2. CEVRPTW'nin tam şarj stratejisine göre CPLEX'den elde edilen optimal sonuçları 72
4.3. EVRP-PDP-TW'nin tam şarj stratejisine göre CPLEX'den elde edilen optimal sonuçları 75
4.4. Eğitim için kullanılan hiper parametreler 78
4.5. Matematiksel model ve önerilen Q-Öğrenme'nin karşılaştırmalı sonuçları 80
4.6. DQN eğitimi için kullanılan hiper parametreler 83
4.7. CEVRP'nin tam şarj stratejisine göre önerilen DQN modelinden elde edilen sonuçları 86
4.8. CEVRP için önerilen DQN modelinin CPLEX çözücüden elde edilen sonuçlarla karşılaştırılması 88
4.9. CEVRPTW'nin tam şarj stratejisine göre önerilen DQN modelinden elde edilen sonuçları 90
4.10. CEVRPTW için önerilen DQN modelinin CPLEX çözücüden elde edilen sonuçlarla karşılaştırılması 92
4.11. EVRP-PDP-TW'nin tam şarj stratejisine göre önerilen DQN modelinden elde edilen sonuçları 97
4.12. EVRP-PDP-TW için önerilen DQN modelinin CPLEX çözücüden elde edilen sonuçlarla karşılaştırılması 99

SİMGELER VE KISALTMALAR DİZİNİ

Simgeler	**Açıklama**
A	Eylemler kümesi
S	Durumlar Kümesi
$R(\,)$	Ödül Fonksiyonu
$T(\,)$	Geçiş Fonksiyonu
$\pi(\,)$	Politika
a	Eylem
s	Durum
r	Ödül
α	Öğrenme Katsayısı
γ	İndirim Faktörü
h	Şarj Tüketim Oranı
g	Şarj Oranı
C	Araç yük kapasitesi
ϵ_{min}	Minimum Epsilon
Δ	Epsilon İndirim Oranı

Kısaltmalar	**Açıklama**
A2C	Eşzamanlı Avantaj Aktör Kritik (Synchronous Advantage Actor-Critic)
A3C	Eşzamansız Avantaj Aktör Kritik (Asynchronous Advantage Actor-Critic)
ABC	Yapay Arı Kolonisi (Artificial Bee Colony)
ACO	Karınca Kolonisi Optimizasyonu (Ant Colony Optimization)

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
AEV	Elektrikli Otonom Araçlar (Autonomous Electric Vehicles)
ALNS	Uyarlanabilir Büyük Komşuluk Araması (Adaptive Large Neighborhood Search)
CEVRP	Kapasiteli Elektrikli Araç Rotalama Problemi (Capacitated Electric Vehicle Routing Problem)
CEVRPTW	Zaman Pencereli Kapasiteli EVRP (CEVRP with Time Windows)
DARP	Ara ve Bin Problemi (Dial-a- Ride Problem)
DDPG	Derin Deterministik Politika Gradyan (Deep Deterministic Policy Gradient)
DDQN	Çift Derin Q-Ağı (Double Deep Q-Network)
DP	Dinamik Programlama (Dynamic Programming)
DPG	Deterministik Politika Gradyan (Deterministic Policy Gradient)
DRL	Derin Pekiştirmeli Öğrenme (Deep Reinforcement Learning)
DQN	Derin Q-Ağı (Deep Q-Network)
EV	Elektrikli Araç (Electric Vehicle)
EVRP	Elektrikli Araç Rotalama Problemi (Electric Vehicle Routing Problem)
EVRP-PDP	Topla Dağıt Elektrikli Araç Rotalama Problemi (Electric Vehicle Routing Problem with Pickup and Delivery)

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
EVRP-PDP-TW	Zaman Pencereli Topla Dağıt EVRP (EVRP with Pickup and Delivery with Time Windows)
EVTSP	Elektrikli Araç Gezgin Satıcı Problemi (Electric Vehicle Traveling Salesman Problem)
GA	Genetik Algoritma (Genetic Algorithm)
GAT	Çizge Dikkat Mekanizması (Graph Attention Mechanism)
GTS	Granüler Yasaklı Arama (Granular Tabu Search)
HEV	Hibrit Elektrikli Araçların (Hybrid Electric Vehicles)
IDE	Geliştirilmiş Diferansiyel Evrim (Improved Differential Evolution)
MDP	Markov Karar Süreçleri (Markov Decision Processes)
MILP	Karma Tamsayılı Doğrusal Programlama (Mixed Integer Linear Programming)
LNS	Geniş Komşuluk Araması (Large Neighborhood Search)
PDP	Topla Dağıt Problemi (Pickup Delivery Problem)
PDPVRP	Topla Dağıt VRP (Pickup Delivery VRP)
PPO	Yakınsal Politika Optimizasyonu (Proximal Policy Optimization)
RL	Pekiştirmeli Öğrenme (Reinforcement Learning)

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
RNN	Tekrarlayan Sinir Ağı (Recurrent Neural Network)
SA	Tavlama Benzetimi (Simulated Annealing)
SUMO	Kentsel Hareketlilik için Simülasyon (Simulation of Urban Mobility)
TS	Yasaklı Arama (Tabu Search)
VNS	Değişken Komşuluk Araması (Variable Neighborhood Search)
VRP	Araç Rotalama Problemi (Vehicle Routing Problems)
VRPB	Önce Dağıt Sonra Topla Araç Rotalama Problemi (Vehicle Routing Problems with Backhaul)
VRPCB	Kümelenmiş Önce Dağıt Sonra Topla VRP (VRP with Clustered Backhauls)
VRPDDP	Bölünebilir Al Bıraklı VRP (VRP with Divisible Delivery and Pickup)
VRPMB	Dağıtım ve Toplamaların Karışık Olduğu VRP (VRP with Mixed Linehauls and Backhauls)
VRPSDP	Eş Zamanlı Topla Dağıt VRP (VRP with Simultaneous Delivery and Pickup)

1. GİRİŞ VE AMAÇ

Son yıllarda, e-ticaretin küresel ölçekte hız kazanması, dağıtım ve son kilometre teslimat lojistiğini önemli ölçüde etkilemiştir. Son kilometre teslimat lojistiği, bir ürünün teslimat sürecinin nihai aşamasını, yani ürünün son kullanıcıya ulaştırılmasını ifade eder. Müşteri memnuniyeti, teslimat hızı, maliyet ve kolaylık üzerinde kritik etkileri nedeniyle bu aşama lojistik stratejilerde odak noktası haline gelmiştir. Sürdürülebilirlik açısından; artan müşteri talepleri şehir merkezlerine çok fazla sayıda teslimat kamyonunun girmesine neden olmakta ve bu durum mevcut alt yapıyı olumsuz yönde etkilemekte, trafik tıkanıklığını arttırmakta, sağlık, çevre ve güvenlik üzerinde olumsuz etkilere sebep olmaktadır Artan müşteri bilinci ve yeni hükümet mevzuatları, sürdürülebilir ve çevre dostu uygulamalara yönelik kargo operasyonları için gerekli çabalar üzerine yoğunlaşmaktadır (Özden ve Sarıçiçek, 2021). Geçtiğimiz on yılda, elektrikli araçlar (EV-Electric Vehicle), sera gazı emisyonlarını azaltma, yenilenebilir enerjiyi destekleme ve sürdürülebilir ulaşım sistemleri oluşturma kapasiteleri sayesinde son kilometre teslimat hizmetlerinde yaygınlaşmıştır. EV'lerin avantajları arasında düşen ulaşım maliyetleri, artırılmış enerji verimliliği ve önemli ölçüde azalmış çevre kirliliği yer almaktadır. Elektrikli araçların kullanımındaki artışla birlikte, fosil yakıtlı araçlarda farklı olarak şarj ile ilgili konuları da içeren elektrikli araç rotalama problemi ortaya çıkmıştır. Literatürde bu probleme Elektrikli Araç Rotalama Problemi (EVRP-Electric Vehicle Routing Problem) adı verilmektedir. EVRP ile klasik Araç Rota Planlama Problemi (VRP-Vehicle Routing Problem) arasındaki temel fark, EV'lerin enerji kaynağı olarak bataryaya sahip olmaları ve batarya seviyelerinin araç yolculuğu sırasında azalmasıdır. Bu durum, EVRP'de şarj kapasitelerinin hesaplanması ve şarj istasyonlarının stratejik yerleştirilmesinin, EV filosunun verimli çalışması için kritik faktörler olduğunu ortaya koymaktadır.

Tüm EVRP varyasyonlarında ortak bir kısıtlama, elektrikli araçların sınırlı menzili nedeniyle şarj istasyonunu ziyaret etme gerekliliğidir. Geleneksel VRP kapsamında, benzinli ve dizel araçlar genellikle tek bir yakıt ikmaliyle yüzlerce kilometre yol alabilmekte olup, günlük rota planlamasında yakıt ikmali ihtiyacı daha az gözlemlenir. Ayrıca, yakıt ikmali süreleri genellikle 5 ila 10 dakika arasında olup, güzergahın toplam seyahat süresi üzerinde

önemli bir etki yaratmamaktadır. Benzin ve dizel istasyonlarının kent merkezlerinden kırsal alanlara kadar yaygın dağılımı, rota planlama modellerine yakıt ikmali noktalarının eklenme gerekliliğini daha da azaltmaktadır. Buna karşılık, EVRP çerçevesinde, elektrikli araçların sınırlı batarya kapasitesi ve nispeten uzun şarj süreleri nedeniyle rotalara şarj istasyonlarının entegrasyonu kaçınılmaz hale gelmektedir. Elektrikli araçların, menzil sınırlamalarını gidermek amacıyla belirli aralıklarla şarj olması gerekmekte ve şarj istasyonlarının stratejik seçimi bu süreci doğrudan etkilemektedir. Sonuç olarak, EVRP, şarj süreleri ve istasyon ziyaret kararlarını rota planlamada ek değişkenler olarak dikkate almayı gerektirdiğinden, klasik VRP'ye kıyasla daha karmaşık bir modelleme sorunu ortaya çıkmaktadır.

Tez çalışmasında EVRP'nin türlerinden biri olan Kapasiteli Elektrikli Araç Rotalama Problemi (CEVRP) ele alınmaktadır. CEVRP, elektrikli araçların sınırlı sürüş menzilini ve yük kapasitelerini hesaba katarak operasyonel ve çevresel hedeflere ulaşmak amacıyla çözüm aranan bir problemdir. Kapasite kısıtlamalarını dahil etmek daha gerçekçi endüstriyel planlamayı kolaylaştırır. CEVRP'ye her müşteri için zaman pencereleri kısıtlaması getirildiğinde, Zaman Pencereli Kapasiteli Elektrikli Araç Rotalama Problemi (CEVRPTW-Capacitated Electric Vehicle Routing Problem with Time Windows) olarak bilinen bir varyant ortaya çıkar. Zaman penceresi kısıtlaması, her müşteriye önceden belirlenmiş bir zaman dilimi içinde hizmet verilmesi gerektiğini belirtir ve bu da rotalama sürecini daha da karmaşık hale getirir. Bu problemleri ele almanın birincil amacı, bir elektrikli araç filosu için en uygun maliyetli rotaları belirlemektir. Bu, genellikle tüm operasyonel kısıtlamalara uyumu sağlarken seyahat mesafelerini en aza indirmeyi gerektirir. CEVRP, CEVRPTW problemlerine çözümler, enerji verimliliği için rotaları optimize ederek ve karbon emisyonlarını en aza indirerek çevresel sürdürülebilirliği teşvik eder ve operasyonel maliyetleri azaltır. Daha az emisyon üreten ve enerji verimliliğini dikkate alan bir filo yönetimi, kurumsal çevre politikalarını destekler. Sonuç olarak, lojistik sektöründe çevre dostu çözümlerin daha geniş bir şekilde uygulanmasına öncülük ederek günümüzün son kilometre teslimat operasyonlarında giderek daha fazla benimsenmektedir.

Tez çalışmasında CEVRP problemini çözmek için, iyi tanımlanmış ödül yapıları ve gelişmiş performans için ince ayarlı hiper parametreler içeren bir Q-öğrenme algoritması önerilmektedir. Q-öğrenme algoritmasına ilişkin EVRP literatürünü gözden geçirildiğinde, iki önemli çalışma ortaya çıkmaktadır. Ottoni vd., (2022) çalışması, Q-öğrenme ve SARSA

(State-Action-Reward-State-Action) olmak üzere iki RL algoritmasının performansını karşılaştırarak Elektrikli Araç Gezgin Satıcı Problemi'ni (EVTSP- Electric Vehicle Traveling Salesman Problem) ele almıştır. Çalışma tek bir rota için elektrikli aracın rota optimizasyonunu sağlamaktadır. Bu yaklaşım, CEVRP gibi filo tabanlı senaryoları ele almak için gerekli olan birden fazla elektrikli aracın yönetimini içermemektedir. Sonuç olarak, metodoloji doğrudan çok araçlı bağlamlara aktarılabilir değildir. Diğer çalışma Dorokhova vd., (2021) şarj istasyonlarında şarj etme olasılığını göz önünde bulundurarak, elektrikli araçlar için bir kaynak ve bir hedef arasında enerji açısından verimli rotalar oluşturmaya odaklanmıştır. Yol planlama (path planning) çalışmaları, bir noktadan (kaynak) başka bir noktaya (hedef) giden görevleri içerir. Bu çalışmalarda, iki nokta arasındaki en uygun rota elde edilir ve başlangıç-bitiş noktaları dışında herhangi bir noktaya gitme zorunluluğu yoktur. Ancak, son kilometre teslimatında çoklu teslimat görevlerine ihtiyaç vardır. Bu yaklaşım, araçların yük kapasitesi gibi kritik kısıtlamaları göz ardı eder. Sonuç olarak, yöntem doğrudan CEVRP'ye uygulanabilir değildir. Rota üzerindeki çoklu görev kısıtlarının dikkate alınması ihtiyacı vardır.

Tez çalışmasında, Zaman Pencereli Kapasiteli Elektrikli Araç Rotalama Problemi (CEVRPTW) için Derin Pekiştirmeli Öğrenme (DRL-Deep Reinforcement Learning) yöntemlerinden biri olan hızlı ve güvenilir çözümler üreten yeni bir derin Q-Ağı (DQN-Deep Q Network) algoritması önerilmektedir. DRL'de eğitim, belirli bir veri kümesi üzerinde gerçekleştirilir. Ek eğitim olmadan, eğitim verileriyle aynı dağılımı paylaşan yeni örnekler için etkili sonuçlar elde edilebilir. Önerilen DQN yönteminde kapsamlı ve ayrıntılı bir durum gösterimini kolaylaştırarak, arama sürecinden özellikleri çıkarmak için yeni bir durum kodlama yöntemi tasarlanmıştır. Ek olarak, CEVRPTW için uygun bir eylem seçmesine rehberlik etmek için yeni bir ödül fonksiyonu ve maskeleme mekanizması tasarlanmıştır.

Bu tez kapsamında ayrıca EVRP problemlerinin varyantlarından biri olan Zaman Pencereli Topla-Dağıt Elektrikli Araç Rotalama Problemi (EVRP-PDP-TW) de ele alınmıştır. Bu problemde müşterilerin bir kısmı dağıtım talebine sahipken bir kısmı da toplama talebine sahiptir. Müşterilerin zaman pencereleri ve EV’nin şarj gereksinimi göz önünde bulundurulduğunda çözülmesi zor bir optimizasyon problemi ortaya çıkar. Tez çalışmasında bu problemi çözmek için CEVRPTW problemi için önerilen DQN yöntemi

EVRP-PDP-TW probleminin kısıtlarını karşılayacak şekilde modifiye edilerek kullanılmıştır. Önerilen DQN temelli yaklaşım ile, zaman penceresi ve şarj kısıtlarına uyularak toplam seyahat mesafesini minimize etmek amaçlanmıştır. Önerilen yöntem DRL yöntemlerinden biridir ve DRL, son yıllarda VRP, EVRP gibi rotalama problemleri için yaygın olarak uygulanmış olsa da, EVRP-PDP problemine henüz uygulanmamıştır. Literatürdeki bu boşluk, çalışmamız için güçlü bir motivasyon sağlar. DRL tekniklerini EVRP-PDP-TW'nin özel gereksinimlerini karşılayacak şekilde uyarlayarak ve iyileştirerek, gerçek dünyadaki önemli problemlerden biri olan bu problem için DRL'nin uygulanabilirliğini ve faydaları gösterilmiştir.

Çalışmada, birden fazla EV içeren bir araç filosu için rota planlaması yapılmaktadır. Yerine getirilecek görevler için belirli bir amaç doğrultusunda rota optimizasyonu ile müşteri talebine göre rota sayısı belirlenmektedir. Tüm müşteri taleplerinin karşılandığı ve filonun kat ettiği toplam mesafenin en aza indirildiği rotalar bulunmaktadır. Her rotayı bir EV'nin gerçekleştireceği varsayımıyla, çözümdeki EV sayısı, Lin vd., (2022) ve Nouicer vd., (2023) çalışmalarında olduğu gibi model tarafından önerilmekte ve optimize edilmektedir. Bu, daha az sayıda EV'nin müşterilerin talepleri için yeterli olmayacağı ve daha fazlasının aşırı olacağı ve gereksiz enerji tüketimine yol açacağı anlamına gelir. Önerilen yöntemde, her EV için bağımsız planlama süreçleri yürütmek yerine, entegre bir çerçeve içinde hem rotaları hem de EV sayısını ortaklaşa belirlemek için tek ajanlı bir yaklaşım benimsenmektedir.

Önerilen algoritmalar ESOGU-CEVRPTW isminde yeni sunulan bir veri seti kullanılarak test edilmiştir. Bu veri setinde yapılan çeşitli düzenlemelerle, EVRP-PDP problemlerine yönelik ESOGU-EVRP-PDP-TW adlı ikinci bir veri seti de türetilmiştir. Bu veri setleri gerçek dünya verilerine (gerçek lokasyonlar) dayanmaktadır, bu nedenle EVRP ve varyantları için veri kümelerinin ağırlıklı olarak sentetik verilerden türetildiği literatürdeki önemli bir boşluğu ele almaktadır. Sentetik veri setleri, gerçek dünya lojistik problemlerinin tüm karmaşıklığını ve inceliklerini yakalamada sıklıkla başarısız olmaktadır. Literatürdeki EVRP için mevcut veri kümeleri genellikle Öklid mesafe metriği kullanılarak hesaplanan düğümler arasındaki mesafelerle müşteri koordinatlarından (x, y) oluşur. Ancak, RL gibi çevresel verileri kullanan yöntemler, sokaklar, kavşaklar ve trafik yoğunluğu dahil olmak üzere çevreyle ilgili ek bilgiler gerektirir. RL genel bir yöntemdir ve herhangi bir veriyle çalışabilir; ancak, çevredeki durum geçişlerini doğru bir şekilde modellemek için

sokak yapısı, kavşak bilgileri ve trafik durumu gibi çevresel verilere erişim genellikle bu yaklaşımlarda daha kritik bir rol oynar. Bu nedenle, "çevresel veri kullanma ihtiyacı" RL'ye özgü bir gereklilik değildir; ancak, RL'deki "durum" ve "ödül" tanımları doğrudan çevresel faktörler tarafından şekillendirildiğinden, bu veri ne kadar zenginse, öğrenme süreci ve karar kalitesi o kadar yüksek olma eğilimindedir. Sonuç olarak, çevresel verinin zenginliği ve kalitesi, öğrenme sürecini ve ortaya çıkan karar alma kalitesini doğrudan artırır.

ESOGU-CEVRPTW veri seti için optimal çözümler, her bir EV'nin önceden tanımlanmış zaman pencereleri içinde müşteri teslimat taleplerini karşılayarak kargo yükü ve batarya kapasitesiyle ilgili kısıtlamalar altında çalıştığı CEVRPTW'yi ele almak üzere tasarlanmış bir matematiksel modelden elde edilmiştir. Problemin zaman penceresi kısıtı içermeyen hali için de matematiksel model kullanılarak optimal çözümler elde edilmiştir. Buna ek olarak, EVRP-PDP-TW problemine özgü olarak geliştirilen matematiksel model uygulanarak ESOGU-EVRP-PDP-TW veri seti üzerinde optimal çözümler elde edilmiştir.

Bu bölümde yapılan çalışmanın literatüre katkıları şu şekilde özetlenebilir:

- Pekiştirmeli öğrenme temelli çalışmalarda "çevre" gereksinimini karşılamak amacıyla iki özgün veri seti geliştirilmiştir.
- Son kilometre teslimat perspektifinden oluşturulan ESOGU-CEVRPTW veri setinde zaman pencereli ve zaman penceresiz senaryolar için optimal çözümler sunulmaktadır.
- Topla dağıt rotalama problemlerine yönelik oluşturulan ESOGU-EVRP-PDP-TW veri seti için de optimal çözümler sunulmaktadır.
- CEVRP problemi için Q-öğrenme temelli bir yöntem önerilmiş ve elde edilen sonuçlar matematiksel modelle karşılaştırılmıştır.
- CEVRPTW problemi için DQN temelli bir çözüm yaklaşımı geliştirilmiş ve performansı matematiksel modelle karşılaştırılmıştır.
- EVRP-PDP-TW problemi de DQN tabanlı olarak çözülmüş ve elde edilen bulgular matematiksel modelle doğrulanmıştır.

Tez raporu yedi ana bölümden oluşmaktadır. İkinci bölümde, elektrikli araç rotalama problemi ile ilgili çalışmalar hakkında ayrıntılı bir literatür taraması sunulmuştur. Üçüncü

bölümde ilk kısmında geliştirilen veri setleri anlatılmıştır ve ikinci kısmında önerilen Q-öğrenme ve DQN yaklaşımlarının detayları açıklanmıştır. Dördüncü bölümde, kapasiteli elektrikli araç rotalama probleminin tanımı, matematiksel modeli ve önerilen çözüm yöntemi anlatılmıştır. Beşinci bölümde topla dağıt elektrikli araç rotalama probleminin tanımı, matematiksel modeli ve önerilen çözüm yöntemi anlatılmıştır. Altıncı bölümde CEVRP, CEVRPTW ve EVRP-PDP-TW problemleri için elde edilen tüm bulgular istatistiksel olarak değerlendirilmiş ve karşılaştırmalı sonuçlar sunulmuştur. Son olarak yedinci bölümde, çalışmanın genel sonuçlarını özetlemekte ve gelecekteki araştırma yönlerine ilişkin önerilerde bulunulmaktadır.

2. LİTERATÜR ARAŞTIRMASI

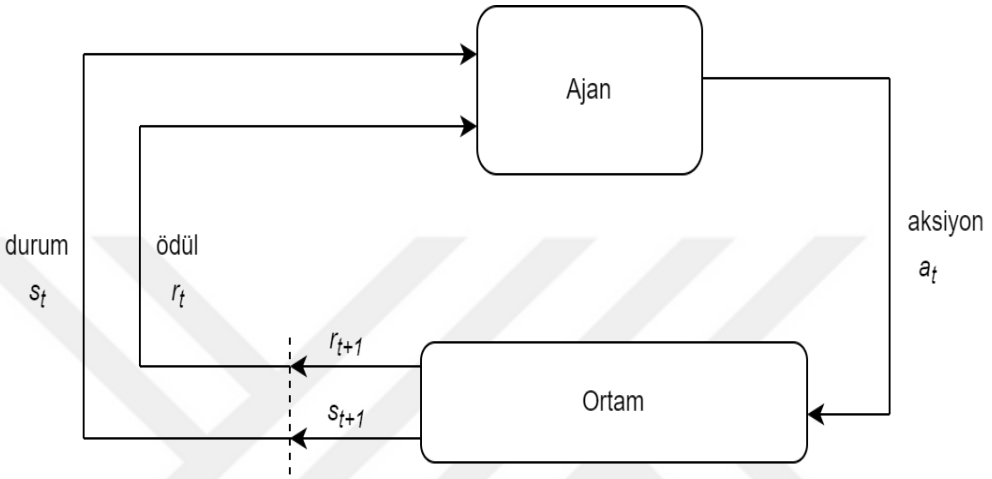
Literatür araştırması dört ana başlık altında sunulmaktadır. Birinci bölümde pekiştirmeli öğrenme ve yaklaşımları hakkındaki genel literatür verilmiştir. İkinci bölümde, pekiştirmeli öğrenme tabanlı kapasiteli elektrikli araç rotalama problemine ilişkin çalışmalar ele alınmıştır. Üçüncü bölümde, pekiştirmeli öğrenme topla dağıt elektrikli araç rotalama problemine odaklanan çalışmalar ayrıntılı olarak incelenmiştir. Dördüncü bölümde ise literatürdeki veri setleri sunulmuştur.

2.1. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme (RL-Reinforcement Learning), makine öğrenmesinin önemli alt alanlarından biridir ve sıralı karar verme problemlerini çözmeyi amaçlayan algoritmalardan oluşur. Bu öğrenme yönteminde, bir ajan (öğrenen varlık) ortamla etkileşim kurarak her adımda bir karar (eylem) verir ve ortam bu karara karşılık bir geri bildirim (ödül) verir. RL algoritmaları, ajanın bulunduğu ortamda hangi eylemler dizisini gerçekleştirmesi gerektiğini belirlemeye çalışır ve amaç, önceden tanımlanmış genel bir ödül fonksiyonunu maksimize etmek yani uzun vadeli toplam getiriyi en üst düzeye çıkarmaktır. Ajan, başlangıçta ortamda nasıl hareket etmesi gerektiğini bilmez ve bu nedenle farklı eylemleri deneyerek ortamdan geri bildirim (ödül veya ceza) alır. Elde edilen bu ödüller veya cezalar doğrultusunda, ajan zamanla hangi eylemlerin daha yüksek ödül sağladığını öğrenir ve bu bilgiye dayanarak bir davranış politikası geliştirir. Çok sayıda deneme ve öğrenme döngüsü sonucunda algoritma, ajana belirli durumlarda hangi eylemleri seçmesi gerektiğini belirleyen bir strateji sunabilir.

Bu yaklaşım, özellikle ortamın tam olarak modellenemediği veya yüksek karmaşıklığa sahip olduğu durumlarda etkili sonuçlar üretmektedir. Bu yönüyle pekiştirmeli öğrenme, denetimli ve denetimsiz öğrenmeden farklı olarak, doğrudan örnek verilerden değil, ortamla etkileşimden öğrenilen deneyimlere dayanır. Şekil 2.1'de ajan ve ortam arasındaki bu etkileşim süreci şematik olarak gösterilmektedir (Richard S. S. ve Andrew G. B., 2018).

Şekil 2.1'de, r_t ajanın ortamdan almış olduğu ödülü, a_t ajanın ortama gönderdiği eylemleri ifade etmektedir. Ajanın ortam içerisinde bulunduğu durumlar ise s_t ile ifade edilmektedir. Ajan herhangi bir t anında, s_t durumunda bulunurken a_t eylemini gerçekleştirerek r_{t+1} ödülünü almakta ve buna bağlı olarak s_{t+1} durumuna geçmektedir.



Şekil 2.1. RL modeli

RL'de, birçok makine öğrenimi yönteminde olduğu gibi hangi eylemleri yapması gerektiği söylenmez. Bunun yerine hangi eylemlerin en fazla ödül kazandığını o eylemleri deneyerek keşfetmesi gerekir. En ilginç ve zorlu durumlarda, eylemler yalnızca anlık ödülü değil, bir sonraki durumu ve bu sayede sonraki tüm ödülleri de etkileyebilir. Bu nedenle karar süreci yalnızca eylemin mevcut anda ne kadar ödül elde edeceği değil aynı zamanda seçilen eylemin gelecekte doğuracağı durumları ve bunların beklenen toplam getiri üzerindeki etkisini de hesaba katar. Bu özellik, pekiştirmeli öğrenmeyi anlık ödülleri maksimize eden diğer yaklaşımlardan ayırır. RL'nin temel hedefi, eylemlerin uzun vadeli sonuçlarını göz önünde bulundurarak beklenen kümülatif ödülü maksimize edecek bir politika öğrenmektir (Denklem 2.1). Denklem 2.1'deki γ indirim faktörünü R_{t+k+1} ise gelecekteki anlık ödülleri temsil eder. İndirim faktörü gelecekteki ödüllerin bugünkü ödül üzerindeki ağırlığını belirleyen bir parametredir.

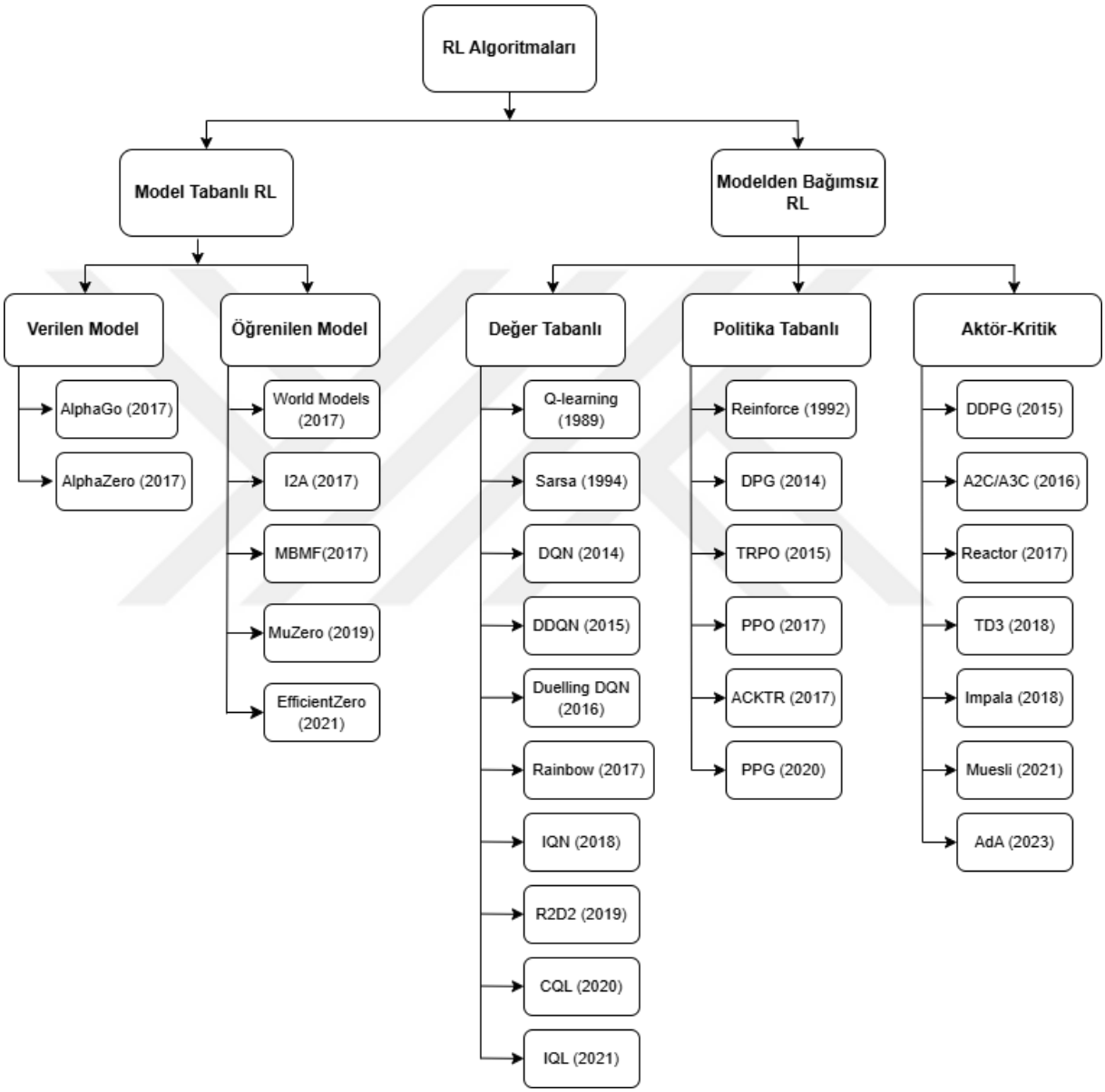
$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.1}$$

RL algoritmaları Markov Karar Sürecine (MDP-Markov Decision Prosess) dayanmaktadır. Markov Karar Süreçleri, karar verme için özel bir stokastik zaman kontrol sürecidir. RL algoritmasının ana aktörleri şunlardır:

• **Ajan (Agent):** İçinde bulunduğu ortamı algılayan, bu algılar doğrultusunda eylemler gerçekleştiren ve bu etkileşimlerden öğrenerek strateji geliştiren yapay bir karar vericidir.
• **Ortam (Environment):** Ajanın etkileşimde bulunduğu fiziksel ya da simüle edilmiş sistemdir. Ortam, ajanın içinde yer aldığı bağlamı ve kararlarını etkileyen dinamik yapıyı tanımlar. Ajanın yaptığı her eylemin sonucu, ortam tarafından belirlenir ve bu sonuçlar aracılığıyla ajan yeni durumlara geçiş yapar ve ödüller alır.
• **Durum (State):** Durum ajanın ortam içerisinde ne şekilde bulunduğu bilgisidir. Ortamın içerisinde, ajan için gerekli minimum bilgiyi taşımakta olup, ortama ve problemin tanımına göre sürekli zamanlı ve ayrık zamanlı olarak ifade edilmektedir.
• **Eylem (Action):** Ajanın, belirli bir durumdayken gerçekleştirebileceği karar veya hareketlerdir. Eylemler, ajan tarafından politika aracılığıyla seçilir ve bu seçim ortamla etkileşimi başlatır. Ajanın alacağı eylemler sürekli veya ayrık değerli olabilmektedir.
• **Geçiş Fonksiyonu (Transition Function)**: Geçiş fonksiyonu, ajanın belirli bir durumda bir eylem gerçekleştirdikten sonra hangi yeni duruma geçeceğini tanımlayan olasılık dağılımıdır.
• **Ödül (Reward)***:* Ajanın gerçekleştirdiği eylemler sonucunda ortamdan almış olduğu geribildirimlerdir. Ödül, ajanın kararlarının ne derece uygun olduğunu belirlemede temel kriter olarak kullanılır. Ödül pozitif olabileceği gibi negatifte olabilir.
• **Politika (Policy):** Politika, t anında s_t durumunda bulunan ajanın eylem kümesi içinden nasıl bir eylem a_t seçeceğini belirlemektedir. Politika, durum uzayından eylem uzayına bir haritalama fonksiyonu olarak tanımlanır. Ajanın takip edeceği politika stratejisi, deterministik veya stokastik olarak modellenebilmektedir.

Pekiştirmeli öğrenme algoritmaları Şekil 2.2'de gösterildiği gibi model tabanlı ve modelden bağımsız olmak üzere iki grupta incelenebilir. Model tabanlı algoritmalar, öğrenmeyi geliştirmek için içinde çalıştıkları ortamın kesin bilgisini kullanır. Öğrenme aşamasında gerçek ortamla daha az sayıda etkileşim kullanır, daha sonra bu etkileşimlere dayalı olarak yeni bir model oluşturur. Bu modeli daha sonraki bölümleri simüle etmek için kullanır ve sanal model tarafından döndürülen sonuçları alır. Modelden bağımsız

algoritmalar ortam hakkında herhangi bir bilgiye sahip değillerdir ve teorik olarak herhangi bir probleme uygulanabilir. Ortam MDP ile modellenerek algoritmalar geliştirilebilir. Süreçteki tüm çevresel kuralları özümseyerek etkileşim yoluyla stratejileri öğrenirler (Winder P., 2020).



Şekil 2.2. Pekiştirmeli Öğrenme Taksonomisi

Modelden bağımsız algoritmalar kendi içerisinde değer tabanlı, politika tabanlı ve aktör-kritik olmak üzere üç grupta incelenebilir. Değer tabanlı bir algoritma, sürekli olarak tahminlerini geliştirerek optimal değer fonksiyonuna veya optimal eylem değeri fonksiyonuna yaklaşır. Genellikle, değer fonksiyonu veya eylem değeri fonksiyonu rastgele

başlatılır, sonra yakınsayana kadar sürekli olarak güncellenir. Değer tabanlı bir algoritmanın optimal değerlere yakınsaması garanti edilir. Politika tabanlı bir algoritma, gelecekte maksimum ödül elde etmek için her durumda gerçekleştirilen eylemin optimal olduğu bir politika arar. Politikayı her adımda yeniden tanımlar ve politika yakınsayana kadar bu yeni politikaya göre değer fonksiyonunu hesaplar. Politika tabanlı bir yöntemin de optimal politikaya yakınsaması garanti edilir ve yakınsaması genellikle değer tabanlı algoritmalardan daha az yineleme gerektirir (Sutton ve Barta, 2018). Aktör-kritik yaklaşımda ise değer tabanlı ve politika tabanlı yaklaşım birlikte kullanılır. Değer tabanlı algoritmalardan bazıları Q-Öğrenme, Sarsa, Derin Q-Ağı (DQN-Deep Q-Network) ve Çift Derin Q-Ağı (DDQN-Double Deep Q-Network) dır. Politika tabanlı algoritmalardan bazıları Reinforce, Deterministik Politika Gradyan (DPG-Deterministic Policy Gradient), ve Yakınsal Politika Optimizasyonu (PPO-Proximal Policy Optimization) dur. Aktör-kritik algoritmalardan bazıları ise Derin Deterministik Politika Gradyan (DDPG-Deep Deterministic Policy Gradient), Eşzamansız Avantaj Aktor Kritik (A3C-Asynchronous Advantage Actor-Critic) dir.

Pekiştirmeli öğrenme algoritmaları derin öğrenme teknikleriyle bütünleştirildiğinde Derin Pekiştirmeli Öğrenme (DRL-Deep Reinforcement Learning) yaklaşımı ortaya çıkar. DRL yaklaşımlarının birincil amacı karmaşık, yüksek boyutlu problemleri ele almaktır. DRL, robotik kontrol, video oyunu, doğal dil işleme ve otonom sürüş dahil olmak üzere çeşitli alanlarda kayda değer başarı göstermiştir. Bu önemli başarılar nedeniyle DRL son yıllarda oldukça ilgi ve popülerlik kazanmıştır. RL ve DRL arasındaki temel fark, derin sinir ağlarının (DNN-Deep Neural Network) kullanılmasıdır. DRL'de, DNN'ler değer fonksiyonlarını veya politikaları tahmin etmek için fonksiyon yaklaştırıcıları olarak hizmet eder ve yüksek boyutlu ve sürekli durum alanlarını yönetebildiğinden gerçek dünya problemlerine uygulanmasını sağlar. DRL algoritmaları politika içi ve politika dışı olmak üzere iki kategoriye ayrılabilir. Derin Q Ağı (DQN) ve Derin Deterministik Politika Gradyanı (DDPG) gibi politika dışı algoritmalar, kaydedilen deneyimlerden öğrenirken, Asenkron Avantaj Aktör-Kritik (A3C) ve Yakınsal Politika Optimizasyonu (PPO) gibi politika içi algoritmalar, doğrudan çevreyle etkileşime girerek öğrenir.

2.1.1. Değer Tabanlı Yöntemler

Değer tabanlı yöntemler, RL çerçevesinde optimal politikayı elde etmek amacıyla değer fonksiyonlarını temel alan yaklaşımlardır. Bu yöntemlerde, öğrenme süreci sırasında ya yalnızca durumlara (state) ya da durum-eylem çiftlerine (state-action) karşılık gelen değerler tahmin edilir. Değer fonksiyonu, belirli bir politika izlendiğinde uzun vadede elde edilmesi beklenen toplam ödülü temsil eder ve bu fonksiyon aracılığıyla politika değerlendirilip iyileştirilebilir. Değer tabanlı bir yöntemde, genellikle öğrenme süreci rastgele başlatılan bir değer fonksiyonu ile başlar. Ardından, ortamla etkileşim yoluyla elde edilen geri bildirimler kullanılarak bu değer fonksiyonu güncellenir. Bu güncelleme işlemi, optimal değer fonksiyonu elde edilene kadar iteratif biçimde devam eder. Buradaki temel varsayım, optimal değer fonksiyonuna karşılık gelen politikanın da optimal politika olacağıdır. Dolayısıyla, bu yöntemlerde politika açık bir şekilde temsil edilmez; bunun yerine, değer fonksiyonu üzerinden dolaylı olarak elde edilir ve yönlendirilir. Değer tabanlı öğrenmeye örnek olarak, durum-eylem değer fonksiyonu üzerinden öğrenme gerçekleştiren Q-öğrenme (Q-Learning) algoritması verilebilir.

Q-öğrenme algoritması, ortamla etkileşim yoluyla öğrenen bir yöntemdir. Hangi koşullarda hangi eylemlerin seçilmesi gerektiğini gösteren stratejileri öğrenmeyi amaçlar. Q-öğrenme, Markov karar süreçlerinin (MDP) her biri için en uygun stratejiyi bulabilir. Bu doğrultuda, mevcut durumdan başlayarak tüm ardışık adımlarda elde edilecek toplam getiri beklentisini en üst düzeye çıkarır. Sonsuz arama süresi ve kısmen rastgele strateji koşulları altında, Q-öğrenme herhangi bir MDP için optimal eylem seçme stratejisini belirleyebilir. Bu algoritma, zamansal fark algoritması ve değer tabanlı politika kullanan bir algoritmadır. Her adımda olası eylemler arasında en yüksek değere sahip eylem seçilir. Algoritma, her adımda mevcut durumdan başlanarak alınabilecek eylemler arasından en yüksek Q değerine sahip olan eylemi seçer. Bu süreç, açgözlü (greedy) bir seçim stratejisine dayanmaktadır ve ajan her zaman kısa vadede en yüksek ödül beklentisi taşıyan aksiyonu tercih eder. Ancak keşif ve sömürü arasındaki dengeyi sağlamak amacıyla uygulamalarda genellikle ε-greedy gibi stratejiler tercih edilir. Bu algoritmada durum-eylem çiftleri kullanılarak Q-tablosu adı verilen bir tablo oluşturulur. Tablo boyutu “durum_sayısı x eylem_sayısı”dır. Mevcut durumdan hangi eylemin seçileceğine karar vermek için tablodaki Q değerlerine bakılır ve maksimum Q değeri seçilir. Bu algoritma ayrık eylem kümelerinde ve deterministik

ortamlarda çalışır. Q-öğrenmenin temelini oluşturan güncelleme kuralı Denklem 2.2'de verilmiştir.

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha.[r_{t+1} + \gamma.maxQ_t(s_{t+1}, a') - Q_t(s_t, a_t)] \quad (2.2)$$

Denklem 2.2'deki $Q(s_t, a_t)$ (durum, eylem) mevcut durumu, α öğrenme katsayısını, r_{t+1} ödül tablosundaki ödül değerini, γ indirim faktörünü ve $maxQ(s_{t+1}, a_t)$ ziyaret edilebilecek yerlerin en yüksek Q değerini temsil etmektedir.

2.1.2. Politika Tabanlı Yöntemler

Politika tabanlı yöntemler, değer fonksiyonlarını açıkça modellemeden doğrudan en uygun politikayı öğrenmeyi hedefleyen pekiştirmeli öğrenme yaklaşımlarıdır. Bu yöntemlerde politika, genellikle öğrenilebilir parametrelerden oluşan bir vektör olan θ ile parametreleştirilir. Yani, öğrenme süreci doğrudan π olarak ifade edilen parametreli politikanın güncellenmesi yoluyla gerçekleştirilir. Bu yaklaşımın temel avantajlarından biri, aracının (ajanın), hem sürekli hem de stokastik eylem uzaylarında etkili politikalar öğrenebilmesine imkân tanımasıdır. Özellikle, değer tabanlı yöntemlerin sınırlı kaldığı yüksek boyutlu veya sürekli eylem alanlarında politika tabanlı yöntemler önemli bir esneklik sağlar.

Politika tabanlı yaklaşımlarda, parametre güncellemeleri belirli bir amaç fonksiyonu doğrultusunda gerçekleştirilir. Bu amaç fonksiyonu, politikanın ne derece iyi olduğunu, başka bir deyişle uzun vadede ne kadar ödül elde edebileceğini nicel olarak ifade eder. Ortamın yapısına bağlı olarak (yani episodik mi yoksa sürekli mi olduğuna göre) bu amaç fonksiyonu farklı biçimlerde tanımlanabilir. Örneğin, episodik ortamlarda amaç fonksiyonu genellikle başlangıç durumundan bölüm sonuna kadar elde edilen ortalama ödül üzerinden hesaplanırken, sürekli ortamlarda tüm durumlar boyunca beklenen ortalama değer ya da zaman adımı başına düşen ortalama ödül esas alınabilir (Blakeman ve Mareschal, 2020).

Değer tabanlı yöntemlerde genellikle eylemler üzerinden maksimum değer alınması gerektiğinden, bu yöntemler çoğunlukla ayrık eylem uzaylarıyla sınırlandırılmıştır. Özellikle yüksek boyutlu eylem uzaylarında, maksimum değerin hesaplanması hesaplama açısından oldukça maliyetli hale gelebilir. Bununla birlikte, sürekli eylem uzaylarında değer fonksiyonlarına dayalı yöntemlerin uygulanabilirliği ciddi şekilde kısıtlanmaktadır. Bu

bağlamda, politika tabanlı yöntemlerin stokastik politikaları daha etkili biçimde öğrenebilmesi önemli bir avantaj sunar. Stokastik politikaların öğrenilmesi, özellikle kısmen gözlemlenebilir ortamlarda ajanın daha istikrarlı ve başarılı performans göstermesini mümkün kılar (Morales, 2020).

2.1.3. Aktör-Kritik Yöntemler

Aktör-Kritik yöntemler, RL'de hem bir değer işlevini hem de doğrudan bir politikayı birlikte öğrenerek değer tabanlı ve politika tabanlı yaklaşımların avantajlarını birleştiren hibrit bir yapıya sahiptir. Bu yöntemlerde, politika parametrelerinin güncellenmesinde karşılaşılan yüksek varyans problemi, değer işlevinden elde edilen tahminler kullanılarak azaltılır. Böylece, öğrenme süreci hem daha kararlı hale gelir hem de ajan, stokastik ve/veya sürekli eylem uzaylarında etkili politikalar öğrenme yeteneğine sahip olur. Politika, ajan tarafından eylem seçimini yönlendirdiği için bu bileşene aktör adı verilir. Öte yandan, bu seçilen eylemlerin kalitesini değerlendiren değer işlevi ise kritik olarak adlandırılır. Kritik, belirli bir politika altında verilen durum-eylem çiftlerinin ne kadar ödül getireceğini tahmin ederek aktöre geri bildirim sağlar. Bu yapı, aktörün güncellenmesini daha bilinçli ve düşük varyansla gerçekleştirmesine olanak tanır.

Aktör-Kritik yöntemlerde amaç, ajan tarafından izlenen politika π ye göre tanımlanan durum-eylem değer fonksiyonunu $Q^{\pi}(s,a)$ doğru tahmin etmeye çalışmaktır. Kritiğin ürettiği $Q^{\pi}(s,a)$ tahminleri, politika gradyanını hesaplarken kullanılır ve böylece doğrudan politika öğreniminde karşılaşılan yüksek varyans sorunu büyük ölçüde azalır. Sonuçta öğrenme süreci hem daha istikrarlı hem de daha örnek verimli hale gelir.

2.2. Pekiştirmeli Öğrenme Tabanlı Kapasiteli Elektrikli Araç Rotalama Problemi

Bu bölümde, EVRP ve varyantlarını ele almak için kullanılan mevcut yöntemlerin kapsamlı bir incelemesi sunulmaktadır. VRP NP-zor sınıfına ait bir problemdir ve EVRP de VRP'nin ek kısıtlar gerektiren bir hali olduğundan, EVRP de NP-zor sınıfında bir problem olarak kabul edilir (Kucukoglu vd., 2021). EVRP ve varyantlarını çözmek için literatürde

önerilen çözüm metodolojileri, kesin çözüm veren yöntemler, sezgisel yöntemler, meta-sezgisel yöntemler ve öğrenmeye dayalı yöntemler olarak kategorize edilebilir (Lin vd., 2022; Nouicer vd., 2023; Rodríguez-Esparza vd., 2024; Tang vd., 2023).

Kesin çözüm veren yöntemler genellikle EVRP'yi formüle eden Karma Tamsayılı Doğrusal Programlama (MILP) modeli gibi matematiksel modelleri içerir ve bunları çözmek için Dal-ve-Sınır Algoritması, Dal-ve-Kesme Algoritması, Dal-ve-Fiyat Algoritmaları, Dal-ve-Fiyat-Ve-Kesme Algoritmaları gibi yaklaşımları kullanır (Desaulniers vd., 2016; Lam vd., 2022). Problemin karmaşıklığından dolayı kesin çözüm veren yöntemler, ya büyük ölçekli problemler için çözüm sağlamada başarısız olur ya da aşırı hesaplama süresi gerektirir. Bu sebeple araştırmacılar bu zorlukları ele almak için sezgisel veya meta-sezgisel yöntemler kullanmışlardır. Bu yöntemler en iyi çözümleri garanti etmese de makul bir zaman dilimi içinde yüksek kaliteli çözümler üretir. Clarke ve Wright Tasarruf Algoritması gibi geleneksel sezgisel yöntemler ve Tavlama Benzetimi (SA-Simulated Annealing), Yasaklı Arama (TS-Tabu Search), Karınca Kolonisi Optimizasyonu, Değişken Komşuluk Arama (VNS- Variable Neighbourhood Search) ve Uyarlanabilir Büyük Komşuluk Arama (ALNS-Adaptive Large Neighbourhood Search) gibi meta sezgisel yöntemler uzun zamandır karmaşık rotalama problemlerini çözmek için kullanılmaktadır (Akbay vd., 2022; Jia vd., 2022; Luo vd., 2024). Sezgisel ve meta sezgisel yaklaşımlar her bir belirli örnek için sıfırdan yürütülür ve genellikle yeni problemler için parametre ayarlamaları gerektirir. Bu yöntemler genelleştirilebilirlik ve uyarlanabilirlik açısından sınırlıdır, yani problemde yapılan küçük değişiklikler bile genellikle bir çözüm elde etmek için algoritmanın yeniden yürütülmesini gerektirir.

Meta sezgisel algoritmaların ortamı yansıtamama gibi sınırlamaları göz önüne alındığında, ortam verisini de yansıtma ihtiyacı duyulan kombinatoryal optimizasyon problemlerini ele almak için öğrenmeye dayalı yöntemler iyi bir alternatif olabilir. Son yıllarda, kendi kendine öğrenme metodolojilerini entegre eden araştırmalar önemli ölçüde öne çıkmış ve pekiştirmeli öğrenme (RL), araç rotalama problemlerinin çözümünde kullanılmaya başlanmıştır (Aljohani vd., 2021; Munir vd., 2024; Narayanan vd., 2022.; Tang vd., 2023; Y. Zhang vd., 2023). RL yöntemleri, ajanların ortamın durumuna veya gözlemine dayanarak en uygun eylem yolunu belirlediği stratejileri içerir. Ajan, mevcut seçenekler kümesinden eylemleri seçer ve belirli bir zaman adımında gerçekleştirilen eylemlerin faydasına göre ödüller alır. Ajanın seçtiği eylemler, içinde çalıştığı ortamın durumunu

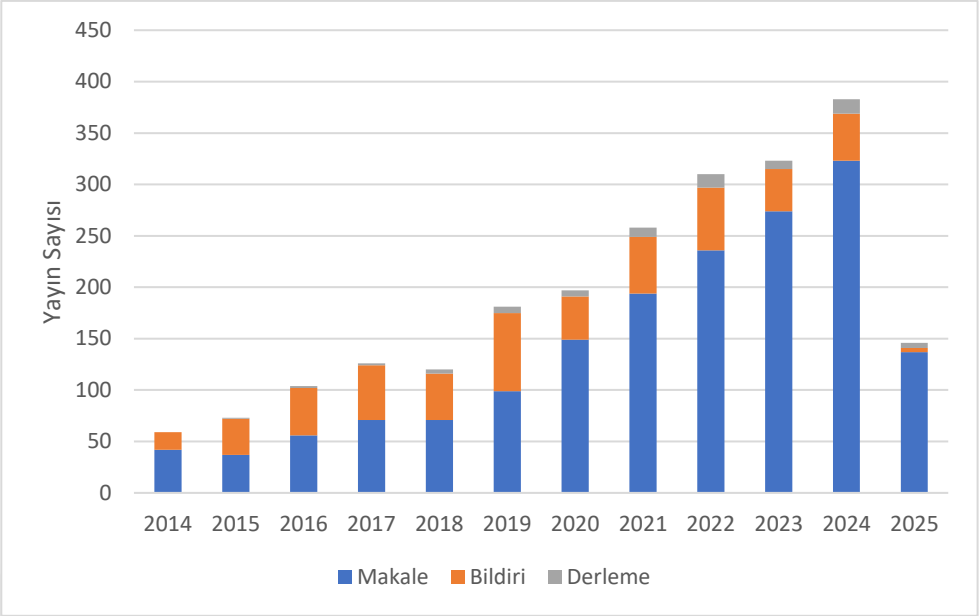
değiştirebilir. RL algoritmaları, deneysel verilerden uyarlanabilir bir şekilde öğrenme ve problem örneklerinin belirli özelliklerine ve çözümlerden gelen geri bildirimlere yanıt olarak stratejilerini dinamik olarak iyileştirme yeteneğine sahiptir. Klasik araç rotalama için geliştirilen geleneksel modeller ve çözümler, şarj kısıtlamaları nedeniyle elektrikli araçlara doğrudan uygulanamaz. Sonuç olarak, özellikle elektrikli araç rotalama problemlerine göre uyarlanmış öğrenmeye dayalı yaklaşımlar geliştirme ihtiyacı giderek daha kritik hale gelmiş ve bu alanda etkili, elektrikli araçlara özgü çözümler önermenin önemi vurgulanmıştır.

EVRP ve varyantlarının çözümü için önerilen kesin çözüm yöntemleri, sezgisel yöntemler, meta-sezgisel yöntemler ve öğrenmeye dayalı tekniklerin her biriyle ilişkili avantaj ve dezavantajların bir özeti Çizelge 2.1'de verilmiştir.

Çizelge 2.1. EVRP için farklı çözüm yaklaşımlarının karşılaştırılması

Çözüm Yöntemi	**Avantajı**	**Dezavantajı**
Kesin methotlar	-Kesin (optimal) çözümler verir. -Küçük ve orta büyüklükteki problemler için uygundur.	-Büyük boyutlu problemler için uzun çözüm süreleri
Sezgiseller, meta sezgiseller	-Kısa çözüm süresi	- Optimal çözümü garanti etmez
Öğrenme tabanlı yöntemler	- Değişen koşullara ve dinamik ortamlara uyum sağlar. -Farklı problem tiplerine ve kısıtlamalara kolayca uyarlanabilir. -Belirsizlik ve rastgelelik içeren ortamlarda karar alma yeteneği	- Optimal çözümü garanti etmez - Eğitim süreci uzun olabilir ve yüksek hesaplama gücü gerektirebilir

Web of Science'da (WOS) 2014 ile 2025 yılları arasında EVRP ile ilgili yayınların sayısı Şekil 2.3'te gösterilmiştir. Şekil 2.3'ü elde etmek için, WOS veritabanı "ALL="electric vehicle" AND (ALL=route planning OR ALL=path planning OR ALL=task scheduling OR ALL=routing)" sorgusuyla arandı. Bu sorgu, "electric vehicle" ifadesini içeren ve ayrıca "route planning", "path planning", "task scheduling" veya "routing" kelimelerinden en az birini içeren tüm makaleleri içerir. Şekil 2.3 incelendiğinde son yıllarda konuya olan ilginin arttığı görülmektedir.



Şekil 2.3. Yıl bazında EVRP çalışmaları

Literatürde EVRP ve varyantları hakkında çeşitli derleme çalışmaları bulunmaktadır (Aslan O vd., 2022; Erdelic vd., 2019; Kucukoglu vd., 2021; Qin vd., 2021). Erdelic vd., (2019) EVRP ile ilgili son gelişmeler hakkında bir literatür incelemesi sunmaktadır. Bu çalışma, EVRP ve ilgili problemleri çözmek için en son teknolojilere genel bir bakış sunmaktadır. Qin vd., (2021) çalışması, EVRP problemleri ve bunların birçok türü hakkında kapsamlı bir araştırma sunmaktadır. Çalışmada, EVRP problemleri dokuz sınıfa ayrılmıştır. Bu dokuz sınıfın her biri için, problem değişkenlerinin ayarları ve çözümlerini elde etmek için kullanılan algoritmalar incelenmiştir.

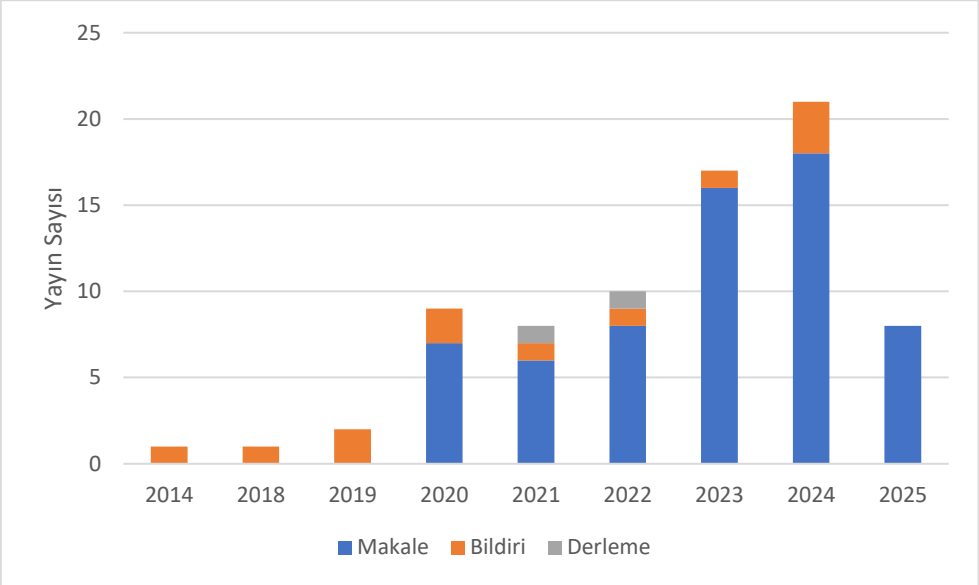
Literatürde birçok araştırmacı, EVRP ve varyantlarının çözümü için kesin çözüm veren algoritmaları (Dal-ve-Kesme Algoritması, Dal-ve-Fiyat Algoritması, Dal-ve-Fiyat-ve-Kesme Algoritması) önermişlerdir (Desaulniers vd., 2016; Keskin & Çatay, 2018; Lam vd., 2022. Desaulniers vd., (2016), EVRP'nin dört varyantını ele almaktadır: (i) rota başına en fazla tek bir şarj yapılmasına izin verilir ve batarya bir şarj istasyonunun ziyareti sırasında tam olarak şarj edilir; (ii) rota başına birden fazla şarj (sadece tam şarj); (iii) en fazla bir tek şarj (kısmi şarj); (iv) çoklu kısmi şarj. Çalışmada problemin çözümü için dal-fiyat-ve-kesme algoritması önerilmiştir. Deneysel sonuçlar, dört varyantın da 100 müşteriye ve 21 şarj istasyonuna kadar çözülebilir olduğunu göstermektedir. Keskin ve Çatay, (2018) EVRPTW'yi normal, hızlı ve süper hızlı olmak üzere üç farklı şarj konfigürasyonuyla kısmi şarjlara izin vererek ele almışlardır. Çalışma, minimum sayıda EV çalıştırırken toplam şarj

maliyetini en küçüklemeyi amaçlamaktadır. Problem, Karma Tamsayılı Doğrusal Programlama (MILP) olarak formüle edilmiştir ve küçük örnekler CPLEX çözücüsü kullanılarak optimal sonuçlar elde edilmiştir. Lam vd., (2022) çalışmasında, zaman pencereleri, parça parça doğrusal şarj ve kapasiteli şarj istasyonları olan Elektrikli Araç Rotalama Problemi için bir Dal-Ve-Kes-ve-Fiyatlandır Algoritması (Branch-and-Cut-and-Price) önermiştir. Önerilen algoritma sonuçları, bu hibrit algoritmanın 100'e kadar müşteriye sahip örnekleri çözdüğünü göstermektedir.

Literatürde yer alan çalışmaların bir kısmında, EVRP problemlerinin çözümü için sezgisel veya meta-sezgisel yöntemler kullanılmıştır (Akbay vd., 2022; Erdelić ve Carić, 2022; Jia vd., 2022; Rastani ve Çatay, 2023; Wang vd., 2023; Luo vd., 2024; Deng vd., 2024; Zhang vd., 2024). Jia vd., (2022) kapasite ve batarya kısıtlarını göz önünde bulundurarak elektrikli araçlar için en uygun rotaların bulunmasına odaklanan bir CEVRP probleminin çözümü için iki düzeyli bir Karınca Kolonisi Optimizasyon Algoritması önermiştir. Bu yaklaşım, CEVRP'yi iki düzeyli bir problem olarak ele almakta ve üst düzeyde kapasite kısıtlı araç rotalama problemi (CVRP), alt düzeyde ise sabit rotalı araç şarj etme problemi olarak ayrıştırmaktadır. Akbay vd., (2022) EVRPTW için Clarke ve Wright tasarruf sezgiseli sunmuş ve bu yöntemle elde edilen çözümleri, Değişken Komşuluk Arama (VNS-Variable Neighborhood Search) meta-sezgiselinde başlangıç çözümü olarak kullanmıştır. Elde edilen sonuçlar, küçük boyutlu örneklerde VNS'nin CPLEX'e yakın bir performans sergilediğini ortaya koymaktadır. Rastani ve Çatay, (2023) elektrikli araç rotalamada yük ağırlığının enerji tüketimi üzerindeki etkisini vurgulayarak problemin çözümü için Geniş Komşuluk Araması (LNS-Large Neighborhood Search) meta-sezgiselini önermiştir. Yapılan çalışma, yük ağırlığının göz ardı edilmesinin rotalarda ve filo boyutunda önemli değişikliklere, dolayısıyla da artan maliyetlere yol açabileceğini göstermektedir. Deng vd., (2024) zaman pencerelerine ve doğrusal olmayan kısmi şarj özelliğine sahip bir EVRP varyantı için Geliştirilmiş Diferansiyel Evrim (IDE- Improved Differential Evolution) algoritması geliştirmiş ve bu algoritmayı Genetik Algoritma, Yapay Arı Kolonisi Algoritması ve Uyarlanabilir Geniş Komşuluk Arama Algoritması ile karşılaştırmıştır. Deneysel sonuçlar, IDE'nin diğer yöntemlere kıyasla daha üstün performans sergilediğini göstermektedir.

Son yıllarda EVRP ve varyantlarının çözümü için öğrenme tabanlı yöntemlerden biri olan pekiştirmeli öğrenme yöntemi ön plana çıkmaktadır. EVRP'nin çözümünde

pekiştirmeli öğrenmeyi kullanan çalışmalara ilişkin literatür, Şekil 2.4'te sunulmaktadır. Şekil 2.4'ü elde edebilmek için, WOS veri tabanında "ALL="electric vehicle" AND ALL="reinforcement learning" AND (ALL=route planning OR ALL=path planning OR ALL=routing)" sorgusu kullanılarak arama yapılmıştır. Bu sorgu, hem "electric vehicle" hem de "reinforcement learning" ifadelerini içeren ve ayrıca "route planning", "path planning", veya "routing" terimlerinden en az birini barındıran yayınları hedeflemektedir.



Şekil 2.4. Pekiştirmeli öğrenme yöntemini kullanan EVRP çalışmaları

Çalışmalar incelendiğinde, Nazari vd., (2018) VRP'ye daha verimli çözümler sağlayan tekrarlayan sinir ağı (RNN- recurrent neural network) tabanlı kodlayıcıyı basitleştirerek işaretçi ağını (pointer network) iyileştirmiştir. Kool vd., (2019)VRP için sinir ağının hem kodlayıcı hem de kod çözücü kısmında dikkat mekanizmasının kullanıldığı transformer tabanlı bir mimari tanıtmıştır. Bu sayede, hesaplama hızı ve çözüm kalitesi önemli ölçüde artmıştır. Basso vd., (2022) yaptıkları çalışmada Dinamik Stokastik Elektrikli Araç Rotalama Problemine (Dynamic Stochastic Electric Vehicle Routing Problem DS-EVRP) odaklanmışlardır. Problemin çözümü için pekiştirmeli öğrenme metodu önerilmiştir. Çalışmanın amacı, enerji tüketimini azaltırken, batarya tükenme riskini en aza indirmek ve gerektiğinde şarj işlemini verimli bir şekilde planlamaktır. Çalışmada, stokastik enerji tüketimi ve dinamik müşteri talepleri dikkate alınarak, tek bir EV için güvenilir şarj planlaması yapılmıştır. EV'nin rotasını tahmine dayalı ve güvenli bir şekilde çevrimiçi planlayabilmek için Monte Carlo simülasyonunu kullanarak stokastik müşteri istekleri ve

enerji tüketimi hakkında çevrimdışı bilgi edinilir. Sonuçlar, rotaları planlayarak ve öngörülü bir şekilde şarj ederek aynı zamanda güvenilirliği koruyarak enerji tasarrufu yapmanın mümkün olduğunu göstermektedir.

Soni vd., (2019) EVRP için pekiştirmeli öğrenme tekniklerini uyguladıkları çalışmada, batarya tüketimini optimize etmeye ve çeşitli takviyeli öğrenme şemaları altında EV seyahat süresini en küçüklemeye odaklanmıştır. Kentsel Hareketlilik için Simülasyon (SUMO) yazılımı kullanılarak bir trafik simülasyon ağı oluşturulmuş ve hem model tabanlı hem de modelden bağımsız yöntemler uygulanmıştır. Bulgular, pekiştirmeli öğrenmenin batarya tüketimi, ortalama seyahat hızı, araç seyahat süresi ve hesaplama karmaşıklığı açısından verimli bir algoritma olduğunu ortaya koymuştur. Li vd., (2021) yaptıkları çalışmada, düğümleri otomatik olarak seçmek için derin pekiştirmeli öğrenme politikasını güçlendirmek amacıyla heterojen bir dikkat mekanizmasıyla bütünleştirilmiş bir sinir ağı kullanmıştır. Bu şekilde, düğümler arasındaki öncelik ilişkileri dikkate alınarak çözümler üretilir. Nolte vd., (2021) yaptıkları çalışmada, Elektrikli Otonom Araçların (AEV) potansiyelinden yararlanarak, özellikle akıllı şehirler gibi yüksek şehirleşmenin olduğu bölgelerde, son mil paket teslimatının verimliliğini artırmayı amaçlayan Randevu Teslimatı'nı tanıtmışlardır. Ayrıca, belirli bir teslimat için en uygun teslimat rotasını bulmak için, pekiştirmeli öğrenme (aktör-kritik yöntemi) ile birlikte en son teknolojiye sahip makine öğrenme yöntemlerine dayanan (dikkat mekanizması, RNN) iki sezgisel çözüm yaklaşımı önermişlerdir. Deneysel sonuçlar, önerilen iki yaklaşımın ilgili probleme çözüm bulmak ve AEV'lerin kullanımının, teslimat turunu tamamlama süresi açısından son mil teslimatının verimliliğini %30'a varan bir marjla artırdığını ve gerçekten başarılı olduğunu göstermektedir.

Lin vd., (2022) EVRPTW'yi çözmek için DRL yöntemini kullanan politika gradyanı tabanlı bir yöntem önermektedir. Bu yöntem, depo konumları, batarya seviyeleri ve şarj istasyonlarına ziyaretler gibi hususları dahil ederek EV'ler için rota kararlarını geliştirmeyi amaçlamaktadır. EV'ler için, zaman pencereleri boyunca tüm müşteri taleplerinin karşılanacağı ve filonun kat ettiği toplam mesafenin en aza indirileceği rotalar bulmaya çalışılır. Önerilen algoritmanın küçük örnekler için ürettiği çözümler optimal olmasa da, büyük ölçekli gerçek zamanlı EV operasyonlarını desteklemek için kullanılabilir. Şarj ile ilgili olarak, RL aracısı, çoğunlukla EV'lerin batarya seviyelerine dayalı olarak şarj kararları

verir. Böylece, bir EV'nin enerjisi tükendiğinde şarj olmasını sağlar, ancak özellikle erken aşamalarda bazı şarj fırsatlarını kaçırabilir. Qi vd., (2022) Hibrit Elektrikli Araçların (HEV') enerji yönetimi stratejisi için yeni bir RL tabanlı algoritma önermektedir. Enerji yönetiminin optimal çözümünü elde etmek için derin Q-öğrenme algoritmasında (DQL-H) hiyerarşik yapı kullanılır. Bu yeni RL yöntemi sadece eğitim sürecindeki seyrek ödül sorununu çözmekle kalmaz, aynı zamanda optimal güç dağılımını da sağlar. DQL-H, her seviyeyi bağımsız olarak eğitir. Deneysel sonuçlar, önerilen DQL-H yöntemi, diğer RL tabanlı yöntemlere kıyasla daha iyi eğitim verimliliği ve daha düşük yakıt tüketimi sağladığını göstermektedir.

Nouicer vd., (2023) EVRPTW problemini çözmek için problemi Structure2vec (S2V) ile Pekiştirmeli Öğrenme yaklaşımını birleştirmiştir. Çalışmanın amacı, Structure2vect ve DDQN kombinasyonunu kullanarak belirli zaman aralıklarında müşterilere hizmet verirken kat edilen mesafeyi en aza indirmektir. Önerilen yöntem, Tunus'taki bir kamu hizmeti filosu şirketinden alınan gerçek dünya verileri üzerinde test edilmiştir. Sonuçlar, önerilen modelin kat edilen mesafeyi %50'ye kadar azalttığını ve ayrıca çalışma süresini optimize ettiğini ortaya koymaktadır. Tang vd., (2023) EVRPTW'yi ele almak için A2C yaklaşımına bir çizge dikkat mekanizması (GAT- Graph Attention Network) entegre etmiştir. Çalışmada ayrıca elektrikli araç kütlesi, hızı, ivmesi ve yol eğimi gibi faktörleri hesaba katan bir enerji tüketim modeli de tanıtılmıştır. Wang vd., (2024) DRL ve ALNS algoritmasını entegre ederek CEVRP'ye yeni bir yaklaşım sunmuştur. Bu yaklaşımda operatör seçimi, yok etme, onarma ve yükleme operatörlerinin uyarlanabilir seçimi için DRL'nin kullanıldığı bir Markov Karar Süreci (MDP) olarak modellenmiştir.

Rodríguez-Esparza vd., (2024) CEVRP'yi ele almak için pekiştirmeli öğrenme ile Hiper-Sezgisel Uyarlamalı Tavlama Benzetimi (HHASARL- Hyper-heuristic Adaptive Simulated Annealing with Reinforcement Learning) algoritmasını önermiştir. Algoritma, seçim mekanizması olarak bir RL modülünü entegre eder ve hareket kabul bileşeni olarak Tavlama Benzetimi (SA) metasezgiselinden Metropolis kriterini kullanır. Deneysel sonuçlarda, HHASARL'nin IEEE WCCI2020 yarışmasında üstün performans gösterdiği vurgulanmıştır. Narayanan vd., (2022) yaptıkları çalışmada zaman pencereleri ve araçtan şebekeye enerji tedarikini içeren kapasiteli elektrikli araç rotalama problemini (CEVRPTW-D) ele almıştır. Çalışmada büyük boyutlu problemlerde verimli bir şekilde çalışması için

tasarlanmış RL tabanlı bir yaklaşım olan QuikRouteFinder'ı önerilmiştir. Solomon veri setleri üzerinde yapılan testler, RL yönteminin Karma Tamsayılı Doğrusal Programlama ve Genetik Algoritma yöntemlerinden 24 kat daha hızlı olurken, en iyi bilinen sonuçların %20'si içinde neredeyse optimal çözümlere ulaştığını ortaya koymuştur. Y. Zhang vd., (2023) geleneksel elektrikli araç rota planlama çözümlerinin yüksek hesaplama karmaşıklığını ve düşük verimliliğini aşmak için derin pekiştirmeli öğrenmeye dayalı bir iki katmanlı bir model önermiştir. İlk katmanda uygulanabilir, zaman açısından verimli bir rota oluşturulur ve ikinci katmanda ise bu rota boyunca optimum bir şarj şeması tasarlanır.

Wang vd., (2024) EVRP için sınırlı batarya kapasitesinin oluşturduğu zorlukları ele alarak uçtan uca derin pekiştirmeli öğrenme mimarisi sunmuştur. Önerilen mimari, düğüm ve kenar bilgilerini yüksek boyutlu yerleştirmelere işleyen gelişmiş bir GAT (Graph Attention Mechanism) tabanlı kodlayıcıya ve rota çözümlerini sırayla üreten çift katmanlı dikkat tabanlı bir kod çözücüye sahiptir. Mozhdehi vd., (2024) Edge-DIRECT adlı derin pekiştirmeli öğrenme yaklaşımını önererek zaman penceresi kısıtlamaları ile heterojen elektrikli araç rotalama problemini (HEVRPTW) ele almıştır. Yöntem, modeldeki çizge yapısına kesişen müşteri zaman pencerelerine dayalı düğüm bağlantısını da dahil ederek geleneksel çizge gösterimlerini geliştirir ve ayrıca enerji tüketimi, seyahat süresi verilerini kullanarak öz dikkat mekanizmasını iyileştirir. Munir vd., (2024) EVRPTW problemini ele almak için transformer tabanlı derin pekiştirmeli öğrenme modelini önermiştir. Önerilen yaklaşım, genellikle uzun vadeli bağımlılıklar ve farklı problem boyutları arasında genelleme ile mücadele eden geleneksel yöntemlerin sınırlamalarını aşmak için heterojen bir dikkat mekanizmasından yararlanır. Son teknoloji çözümlerle yapılan karşılaştırmalı analizler, yeni modelin yalnızca seyahat mesafesini etkili bir şekilde en aza indirmekle kalmayıp aynı zamanda çeşitli EVRPTW örnekleri arasında sağlam bir performans sağladığını ortaya koyduğunu göstermektedir.

Çizelge 2.2, EVRP ve varyantlarını inceleyen çalışmaların bir özetini sunmaktadır. Çizelge 2.2'de sunulan her çalışma için problem tipi, zaman penceresi kısıtı, kullanılan yöntem ve çalışmanın test edildiği veri seti yer almaktadır.

Çizelge 2.2. Literatürdeki EVRP çalışmaları

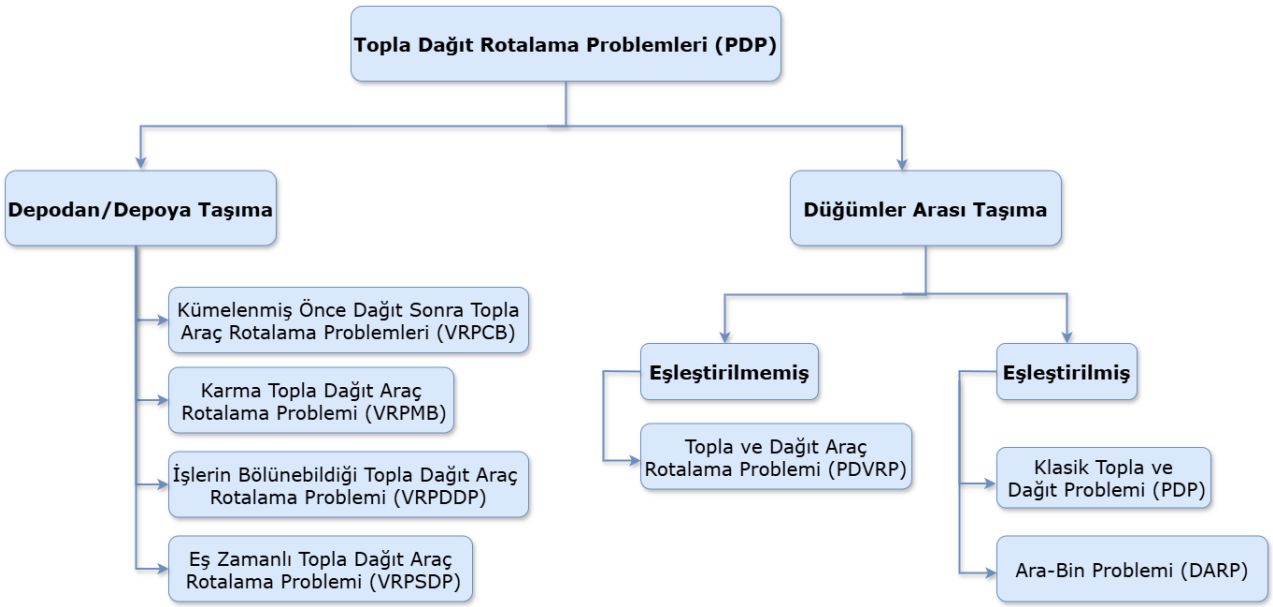
Kaynak	**Problem tipi**	**Zaman penceresi**	**Kullanılan yöntem**	**Kullanılan veri seti**
Desaulniers vd., (2016)	EVRPTW	✓	Dal fiyat ve kesme algoritması	Schneider vd., (2014) veri seti
Taş, (2021)	EVRPFTW	✓	Sütun oluşturma yöntemi	(Schneider vd., 2014)
Lam vd., (2022)	EVRPTW	✓	Dal fiyat ve kesme algoritması	Goeke, (2019a) veri seti
Akbay vd., (2022)	EVRPTW	✓	Clarke ve Wright, VNS	Genişletilmiş Schneider vd., (2014) veri seti
Jia vd., (2022)	CEVRP		İki seviyeli ACO	IEEE WCCI2020 veri seti
Rastani ve Çatay, (2023)	CEVRP		LNS	Schneider vd., (2014) veri seti Desaulniers vd., (2016) veri seti
Ottoni vd., (2022)	EVTSP		Q-Öğrenme, Sarsa	Brezilyadaki 2 eyaletten gerçek dünya verileri
Dorokhova vd., (2021)	EVTSP		Q-Öğrenme	İsviçre'deki bir bölgeden gerçek dünya verileri
Aljohani vd., (2021)	EVTSP		DDQN	Kaliforniya ve Florida'dan gerçek dünya veri seti
Lin vd., (2022)	EVRPTW	✓	DRL Reinforce, GAT, S2V	Schneider vd., (2014) veri seti
Tang vd., (2023)	EVRPTW	✓	A2C, GAT	Rastgele üretilmiş sentetik veri seti
Nouicer vd., (2023)	EVRPTW	✓	DDQN, S2V	Schneider vd., (2014) veri seti, Tunus'un Bizerte kentinden gerçek dünya verileri.
Wang vd., (2024)	CEVRP		DRL ve ALNS	IEEE WCCI2020 veri seti
Rodríguez-Esparza vd., (2024)	CEVRP		SA ve RL	IEEE WCCI2020 veri seti
Narayanan vd., (2022)	CEVRPTW	✓	DQN	Solomon veri seti
Zhang vd., (2023)	CEVRP		DQN	Çin'in Pekin kentinden

				gerçek dünya verileri
Wang vd., (2024)	EVRP		DRL, Reinforce, GAT	Rastgele üretilmiş sentetik veriseti
Mozhdehi vd., (2024)	CEVRPTW	✓	DRL	Kanada'nın şehri olan Edmonton ve Calgary'den gerçek dünya veri seti
Munir vd., (2024)	EVRPTW	✓	Transformer tabanlı DRL	Schneider vd., (2014) veri seti

2.3. Pekiştirmeli Öğrenme Tabanlı Topla-Dağıt Elektrikli Araç Rotalama Problemi

EVRP varyantlarından biri de Topla Dağıt Araç Rotalama Problemleridir (VRPPD-Vehicle Routing Problem with Pickup and Delivery). Topla Dağıt Araç Rotalama Problemleri VRP problemlerinden biri olup PDP'nin problem sınıfları Şekil 2.5'te gösterilmektedir.

Topla Dağıt Araç Rotalama Problemi (Vehicle Routing Problem with Pickup and Delivery) depodan depoya taşımanın olduğu grup, müşteriden toplanan ürünlerin doğrudan diğer bir müşteriye taşınmasının söz konusu olmadığı problemlerdir. Bu problem türlerinde bütün talepler depodan müşteriye ya da müşteriden depoya taşınabilmektedir. Önce Dağıt Sonra Topla Araç Rotalama Problemi (VRPB- Vehicle Routing Problems with Backhaul) depodan çıktıktan sonra önce dağıtım görevlerinin yerine getirildiği daha sonra toplama (alma) görevlerinin planlandığı problem tipidir. VRPB dört alt kategori altında incelenir (Lin vd., 2014).



Şekil 2.5. Topla Dağıt Araç Rotalama Problemlerinin Sınıflandırılması (Parragh vd.,2008)

Kümelenmiş Önce Dağıt Sonra Topla Araç Rotalama Problemi (Vehicle Routing Problem with Clustered Backhauls –VRPCB), Karma Topla Dağıt Araç Rotalama Problemi (Vehicle Routing Problem with Mixed Linehauls and Backhauls –VRPMB), İşlerin Bölünebildiği Topla Dağıt Araç Rotalama Problemi (Vehicle Routing Problem with Divisible Delivery and Pickup –VRPDDP) ve Eş Zamanlı Topla Dağıt Araç Rotalama Problemi (Vehicle Routing Problem with Simultaneous Delivery and Pickup –VRPSDP) ilk grupta yer almaktadır. İlk iki alt kategoride müşteriler ya yalnızca teslimat (delivery) ya da yalnızca toplama (pickup) müşterisidir ancak aynı anda ikisi birden olamazlar. Son iki alt kategoride, müşteri hem teslimat hem toplama gerektirir. Birinci alt kategori olan VRPCB probleminde, bir küme ya da gruptaki tüm teslimat müşterileri ziyaret edilmeden ilk toplama müşterisine gidilmesine izin verilmez. İkinci alt kategori dağıtım ve toplamaların karışık olduğu VRPMB olup, teslimat ve toplama taleplerinin karışık şekilde aynı tur içinde yerine getirildiği durumları kapsar. Üçüncü alt kategori VRPDDP problemidir ve burada bir müşteri teslimat ve/veya toplama için bir ya da iki kez ziyaret edilebilir. Dördüncü alt kategori ise VRPSDP problemidir, müşterilerin toplama ve dağıtma talepleri aynı anda karşılanır.

VRPPD sınıfı ise iki alt kategoriye ayrılır. Birinci alt kategori, toplama ve dağıtma noktalarının eşleşmediği durumları içerir. Toplama noktasından alınan her bir yük, herhangi bir dağıtım talebini karşılamak için kullanılabilir ve bu problem literatürde Topla Dağıt VRP

(PDVRP) olarak anılır. İkinci alt kategori ise toplama ve dağıtma noktalarının eşleştiği durumları içerir. Klasik Topla Dağıt Problemi (PDP) yük taşımaya odaklanırken Ara-Bin Problemi (DARP- Dial-a- Ride Problem) insan taşımacılığını ele alır.

Topla dağıt araç rotalama problemleri NP-zor problem sınıfındadır (Agrali ve Lee, 2023). Topla dağıt elektrikli araç rotalama problemi PDP'nin elektrikli araçlar için ele alınmış hali olduğundan NP-zor problem sınıfında kabul edilir. Problemi ele almak için literatürde önerilen çözüm metodolojileri kesin çözüm yöntemleri, sezgisel yöntemler, meta-sezgisel yöntemler ve öğrenmeye dayalı yöntemler olarak kategorize edilebilir. Problemin karmaşıklığından dolayı, kesin çözüm yöntemleri genellikle büyük ölçekli durumlar için çözümler sağlamada zorlanır veya uzun hesaplama süreleri gerektirir. Bu nedenle, araştırmacılar bu problem için giderek daha fazla sezgisel ve meta-sezgisel yöntemlere yönelmiştir. Son zamanlarda ise makine öğrenmesinin bir alt alanı olan kendi kendine öğrenme tekniklerini içeren çalışmalar ön plana çıkmıştır. Pekiştirmeli Öğrenme, öğrenmeye dayalı yöntemlerde dikkate değer bir yaklaşım olarak ortaya çıkmıştır. Özellikle, derin pekiştirmeli öğrenme rotalama problemlerini çözmek için dikkat çekmiştir

Literatürde birçok araştırmacı, EVRP-PDP probleminin çözümü için kesin çözüm veren algoritmaları önermişlerdir (J. Lin vd., 2016; Soysal vd., 2020; Yang vd., 2021).
Lin vd. (2016) Topla Dağıt Elektrikli Araç Rotalama Probleminde (EVRP-PDP Electric Vehicle Routing Problem with Pickup and Delivery) araç yükünün batarya tüketimi üzerindeki etkisini incelemiştir. Problemin çözümü için matematiksel bir model önerilmiş olup, testler Austin, Texax bölgesinde gerçekleştirilmiştir. Yapılan testler sonucunda araç yükünün rota stratejisi üzerindeki etkisinin önemli olduğu gösterilmiştir. Soysal vd., (2020) stokastik batarya tüketimi varsayımı altındaki EVRP-PDP için şansa bağlı Karma Tamsayılı Doğrusal Olmayan Programlama Modeli (C-MILP Chance-Constrained Mixed Integer Non-Linear Programming Model) ve Doğrusal Yaklaşım (LA- Linear Approximation) önermiştir. Çalışmada EV'ler şarj istasyonları yerine batarya takas istasyonlarını ziyaret eder. Önerilen yöntem İstanbul'dan elde edilen gerçek dünya verileri kullanılarak test edilmiştir. Yang vd., (2021) çalışmasında zaman penceresi ve şarj stratejileri içeren Karma Topla Dağıt Elektrikli Araç Rotalama Problemine (EVRPMBTW-RS Electric Vehicle Routing Problem with Mixed Backhauls, Time Windows, And Recharging Strategies) odaklanılmıştır. Çalışmada problem önce ikili karesel bir probleme dönüştürülmüştür ve ardından artırılmış Lagrangian

Gevşeme Tekniği ve alternatif yön çarpanı yöntemi (ADMM- Alternating Direction Multiplier Method) ile problem “en düşük maliyetli zaman-bağımlı yollar” dizisine ayrıştırılmıştır. Her yolu verimli bir Dinamik Programlama (DP) algoritmasıyla çözülmektedir. Önerilen çözüm yaklaşımı, basit bir yedi düğümlü ağ ve gerçek dünya Yizhuang yol ağı örnekleri üzerinde test edilmiştir.

Literatürde yer alan çalışmaların büyük bir kısmında, EVRP-PDP problemlerinin çözümü için sezgisel veya meta-sezgisel yöntemler kullanılmıştır. Bu çalışmaların bir kısmında zaman penceresi kısıtı dahil edilmezken bir kısmında ise dahil edilir. Goeke, (2019c), zaman penceresi kısıtı içeren Topla Dağıt Elektrikli Araç Rotalama Problemi için (EVRP-PDP-TW Pickup and Delivery Electric Vehicle Routing Problem with Time Windows) Granüler Yasaklı Arama (GTS-Granular Tabu Search) algoritması önermiştir. Yapılan sayısal çalışmalarda, önerilen yaklaşım küçük boyutlu örneklerde doğrulanmıştır. Ayrıca yapılan çalışmada ilgili problem için yeni bir veri seti önerilmiştir. Zhao ve Lu, (2019) EVRP-PDP-TW probleminin heterojen filolu halini çözmek için ALNS algoritması ile Tamsayılı Programlamaya dayalı bir sezgisel yaklaşım önermiştir. ALNS tarafından elde edilen en iyi çözüm tamsayı programlama ile iyileştirilmiştir. Gerçek dünya verilerinden oluşturulan yirmi örnek önerilen algoritmanın etkinliğini doğrulamak için kullanılmıştır. Li vd., (2019) çalışmasında hetorojen EV’lerden oluşan bir filo için EVRP-SPD-TW problemini ele almıştır. Problem EV'lerin seyahat süresi ve yüküyle ilişkili enerji tüketim analizini dikkate alan MILP olarak formüle edilmiştir. Önerilen modeli çözmek için VNS algoritması kullanılmıştır.

Ghobadi vd., (2021) yaptıkları çalışmada Çok Depolu ve Bulanık Zaman Pencereli Topla Dağıt Elektriki Araç Rotalama Problemini (MD-EVRP-FTW-PD Multi-Depot Electric Vehicle Routing Problem with Fuzzy Time Windows and Pickup/Delivery) çözmek için üç meta sezgisel yöntem önermiştir. Bu meta sezgisel yöntemler SA, VNS ve bu iki yöntemin hibrit olarak kullanılmasıdır. Önerilen hibrit VNS-SA algoritması, diğer algoritmalarla karşılaştırıldığında daha verimli sonuçlar vermiştir. Nolz vd., (2022) sınırlı sayıda şarj istasyonu içeren EVRP-PDP için ALNS algoritmasını önermiştir. Ayrıca çalışmadaki EV filosu hetorojendir ve çalışmada kısmi şarj politikası kullanılmıştır. Avusturyalı bir paket teslimat şirketinden alınan verilere dayanan farklı ortamlar ve senaryolar için önerilen yöntem test edilmiştir.

Xu vd., (2022) EVRP-SPD problemini doğrusal olmayan şarj ve yüke bağlı batarya tüketimi kısıtlamaları altında ele almıştır. Problemi MILP olarak formüle edilmiştir ve küçük boyutlu problemler için CPLEX ile çözümler elde edilmiştir. Büyük boyutlu problemleri çözmek için ise ALNS metasezgiseli önerilmiştir. Yilmaz ve Kalayci, (2022) yaptıkları çalışmada eş zamanlı topla dağıt elektrikli araç rotalama problemini (EVRP-SPD Electric Vehicle Routing Problem with Simultaneous Pickup and Delivery) ele almıştır. Problem tamsayı doğrusal bir model olarak formüle edilmiştir ve toplam kat edilen mesafeyi en aza indirmek amaçlanmıştır. Problemin başlangıç çözümü için Clarke ve Wright'ın tasarruf algoritması kullanılmıştır. Ardından VNS algoritmasının çeşitli varyantları test edilerek makul bir sürede en iyi çözümü veren azaltılmış VNS algoritması önerilmiştir. Yapılan testler sonunda elde edilen sonuçlar, önerilen yöntemin tüm kıyaslama örneklerinde çözüm kalitesi ve zaman açısından etkili çözümlere ulaşabileceğini göstermiştir.

Agrali ve Lee, (2023)elektrikli araçlar ve transferlerle ilgili yeni bir topla dağıt problemi sunmuştur. Her talebin tek bir araç tarafından gerçekleştirilmesi şeklindeki geleneksel kısıtlama, bu problemde transferlerle gevşetilmiştir. Problemde çoklu depo kısıtı vardır. Tüm bu kısıtlamaları kapsayan MILP modeli geliştirilmiştir. Problemin hesaplama zorluğunu ele almak için SA ve LNS algoritmasını birleştiren hibrit bir yöntem önerilmiştir.

Zhou vd., (2024) EVRP-PDP-TW problemindeki EV'lerin kuyruk planlamasını dikkate almıştır. Seyahat mesafesi ve kuyruk süresinin maliyetini en aza indirmek için yeni bir MILP modeli önerilmiştir. Orta ölçekli problemleri çözmek için uyarlanabilir bir hibrit mahalle arama algoritması (AHVS- Adaptive Hybrid Neighborhood Search Algorithm) geliştirilmiştir. Deneysel sonuçlarda, modelin ve AHNS'nin etkinliğini gösterilmiştir.

L. Wang vd., (2024) yaptıkları çalışma heterojen araç rotalama problemi (HVRP-Heterogeneous Vehicle Routing Problem), çoklu yolculuk araç rotalama problemi (MTVRP-Multi-Trip Vehicle Routing Problem), ile EVRP-PDP-TW problemini birleştirerek yeni bir problem sunmuştur (H-MT-EVRP-PD Heterogeneous and Multi-Trip Electric Vehicle Routing Problem with Pickup and Delivery). Çalışmadaki amaç fonksiyonu, seyahat masraflarını, bekleme sürelerini, şarj maliyetlerini ve sabit araç maliyetlerini kapsayan genel

maliyeti en aza indirmektir. Problemin çözümü için VNS ve ALNS algoritması önerilmiş olup bu iki yönyemin sonuçları karşılaştırılmıştır.

Çizelge 2.3, topla dağıt EVRP problemlerini inceleyen çalışmaların bir özetini sunmaktadır. Çizelge 2.3'te sunulan her çalışma için problem tipi, zaman penceresi kısıtı, kullanılan yöntem, çalışmanın test edildiği veri seti ve ilişkili amaç fonksiyonu yer almaktadır.

Çizelge 2.3. Literatürdeki EVRP-PDP çalışmaları

Kaynak	**Problem tipi**	**Zaman penceresi**	**Kullanılan yöntem**	**Kullanılan veri seti**	**Amaç Fonksiyonu**
J. Lin vd., (2016)	EVRP-PDP		Matematiksel model	Texax Austin'den gerçek dünya veri seti	Toplam seyahat süresini ve enerji maliyetini minimize etmek
Goeke, (2019c)	EVRP-PDP-TW	✓	GTS	Goeke D., (2019) veri seti	Toplam seyahat mesafesini minimize etmek
Zhao ve Lu, (2019)	EVRP-PDP-TW	✓	ALNS	Çin'in Wuhan kentinden gerçek dünya veri seti	Toplam EV sayısı, toplam seyahat mesafesi ve toplam şarj olma maliyetini minimize etmek
(Li vd., 2019)	EVRP-SPD	✓	VNS	Çin'den gerçek dünya veri seti	Toplam seyahat mesafesini minimize etmek
Soysal vd., (2020)	EVRP-PDP		C-MILP, LA	İstanbul'dan gerçek dünya veri seti	Toplam enerji tüketimini minimize etmek
Ghobadi vd., (2021)	EVRP-PDP	✓	SA, VNS/hibrit VNS-SA	Solomon veri setinden türetilen veri seti	Toplam seyahat mesafesini minimize etmek
Yang vd., (2021)	EVRPMBTW-RS	✓	Artırılmış Lagrangian gevşeme, ADMM, DP	Beijing Yizhuang'dan gerçek dünya veri seti	Toplam seyahat maliyetini minimize etmek
Xu vd., (2022)	EVRP-SPD		ALNS	Montoya vd., (2017) çalışmasından	EV sayısını ve seyahat süresi minimize etmek

				türetilen veri seti	
Nolz vd., (2022)	EVRP-PDP		ALNS	Avusturya'dan gerçek dünya veri seti	Toplam seyahat maliyetini minimize etmek
Yilmaz ve Kalayci, (2022)	EVRP-SPD		Clarke ve Wright sezgiseli, VNS	Schneider vd., (2014) veri setinden türetilmiş veri seti	Toplam seyahat mesafesini minimize etmek
Agrali ve Lee, (2023)	EVRP-PDP	✓	Hibrit LNS ve SA	Agrali C., (2021) veri seti	Toplam seyahat mesafesini minimize etmek
Zhou vd., (2024)	EVRP-PDP	✓	AHNS	Goeke D., 2019 veri setinden türetilmiş veri seti	Seyahat mesafesi ve EV kuyruk süresini minimize etmek
L. Wang vd., (2024)	H-MT-EVRP-PD	✓	VNS, ALNS	(Yao et al., 2019)çalışmasında sunulan gerçek dünya veri seti	Seyahat masraflarını, bekleme sürelerini, şarj maliyetlerini ve sabit araç maliyetlerini kapsayan genel maliyeti minimize etmek

2.4. Literatürdeki Mevcut Veri Setleri

Literatürdeki veri setleri araştırılarak EVRP ye uygun toplamda dokuz veri seti incelenmiştir. EVRP problemi için yaygın olarak kullanılan veri setlerinden biri Schneider vd., (2014) tarafından hazırlanan veri setidir. Bu veri seti VRP için iyi bilinen Solomon (1987) veri seti kullanılarak oluşturulmuştur. Veri seti kümelenmiş (C-Clustered), rasgele (R-Random) ve rasgele kümelenmiş (RC-Random-Clustered) verilerden oluşur. Batarya kapasitesi, araç yük kapasitesi, yakıt tüketim oranı, şarj oranı ve ortalama hız parametre değerleri veri setinde verilmektedir. İncelenen tüm veri setlerinde hangi bilgilerin tutulduğu Çizelge 2.4'te gösterilmektedir

Çizelge 2.4. Literatürdeki EVRP veri setleri

	EV							Lojistik								EV Filo Tipi		Şarj Stratejisi	
Makale/Veri seti	**Araç yük kapasitesi**	**Araç batarya kapasitesi**	**Enerji tüketim oranı**	**Ortalama hız**	**Şarj durumu**	**Şarj teknolojisi**	**Şarj süresi**	**Konum (pos_x, pos_y)**	**Talep**			**Zaman pencere si**	**Servis süresi**	**Kenar seyahat süresi**		**Homojen**	**Heterojen**	**Tam**	**Kısmi**
									P	**D**	**P/D**								
(Schneider vd., 2014)	✓	✓	✓	✓			✓	✓		✓		✓	✓				✓	✓	
(Goeke, 2019c)	✓	✓	✓	✓			✓	✓			✓	✓	✓						✓
(Mavrovouniotis vd., 2020)	✓	✓	✓					✓		✓						✓		✓	
(Doppstadt vd., 2020)								✓				✓	✓						
(S. Zhang vd., 2020)	✓	✓	✓					✓		✓		✓	✓	✓					
(Ding, S., 2022)								✓		✓		✓	✓						
(Agrali & Dissertation, 2022)								✓			✓	✓	✓						
(Futalef vd., 2023)	✓	✓			✓	✓	✓	✓		✓		✓	✓	✓		✓			✓
(Zhou vd., 2023)								✓		✓		✓	✓						
(Bac & Erdem, 2021)	✓	✓	✓				✓	✓		✓		✓	✓	✓			✓		✓

Tez kapsamında EVRP için literatür de dikkate alınarak bir veri seti gereksinim çalışması yapılmıştır. İncelenen tüm veri setlerinde tutulan bilgiler ışığında Çizelge 2.5'te önerilen veri setinden tutulacak bilgiler yer almaktadır.

Çizelge 2.5. Önerilen veri setinde tutulacak bilgiler

ELEKTRİKLİ ARAÇ (EV)/ŞARJ (CHARGE)	Aracın yük kapasitesi (Vehicle load capacity)
	Aracın batarya kapasitesi (Vehicle battery capacity)
	Aracın enerji tüketim oranı (Energy consumption rate:reduction of battery capacity when traveling one unit of distance)
	Aracın ortalama hızı (Average velocity)
	Aracın batarya şarj oranı (Battery recharging rate)
LOJİSTİK (LOGISTICS)/ORTAM (ENVIRONMENT)	Deponun konumu (Location (pos x, pos y) of depot)
	Dağıtım noktalarının konumları (Location (pos x, pos y) of delivery nodes)
	Şarj istasyonlarının konumları (Location (pos x, pos y) of charging stations)
	Dağıtım noktalarının talep miktarı (Demand of delivery nodes)
	Dağıtım noktalarındaki bırakma süresi (Unloading time)
	İki düğüm arası seyahat süresi (Edge travel time)
	İki düğüm arası seyahat mesafesi (Edge travel distance)
	Müşterinin alma bırakma için tercih ettiği zaman aralığı (Time Windows)

Çizelge 2.5'te yer alan önerilen veri setinde tutulacak bilgiler EV/Şarj ve Lojistik/Ortam olmak üzere iki ana başlık altında toplanmıştır. Aracın yük kapasitesi, batarya kapasitesi, enerji tüketim oranı, ortalama hızı ve şarj oranı bilgileri elektrikli araca ait bilgilerdir. Lojistikle ilgili ilk olarak ortamda yer alan düğümlerin (depo, dağıtım noktaları, şarj istasyonları) konum bilgisi tutulur.

Literatürdeki veri setleri ile oluşturulan veri seti arasındaki farklar şu şekildedir:

- Literatürdeki veri setlerinde gerçek ortamdan oluşturulan veriler yer almamaktadır. Veri setlerinde yer alan müşteri ve şarj istasyonlarına ait konum bilgileri belli bir aralıkta rasgele üretilen sayılardan oluşturulmuştur. Tez kapsamında oluşturulan veri seti gerçek ortam konum bilgileri doğrultusunda üretilmiştir. Eskişehir Osmangazi Üniversitesi Meşelik kampüsünde yer alan bina/birimler müşteri olarak belirlenmiştir. Toplamda 118 müşteri vardır. Veri setindeki müşterilerin konum bilgisi binaların gerçekteki konum bilgisini içermektedir. Veri setindeki şarj

istasyonları 10 tane olup, şarj istasyon noktaları olarak kampüsteki otoparklar seçilmiştir. Veri setindeki servis süresi hesaplanırken bina önünden bina içindeki müşteriye kadar olan yaya yolu mesafesi dikkate alınmıştır.

- Literatürdeki veri setlerinde düğümler arası mesafeler hesaplanırken kuş uçuşu uzaklık kullanılırken, oluşturulan veri setinde yolların gerçek mesafesi kullanılmıştır.

3. MATERYAL VE YÖNTEM

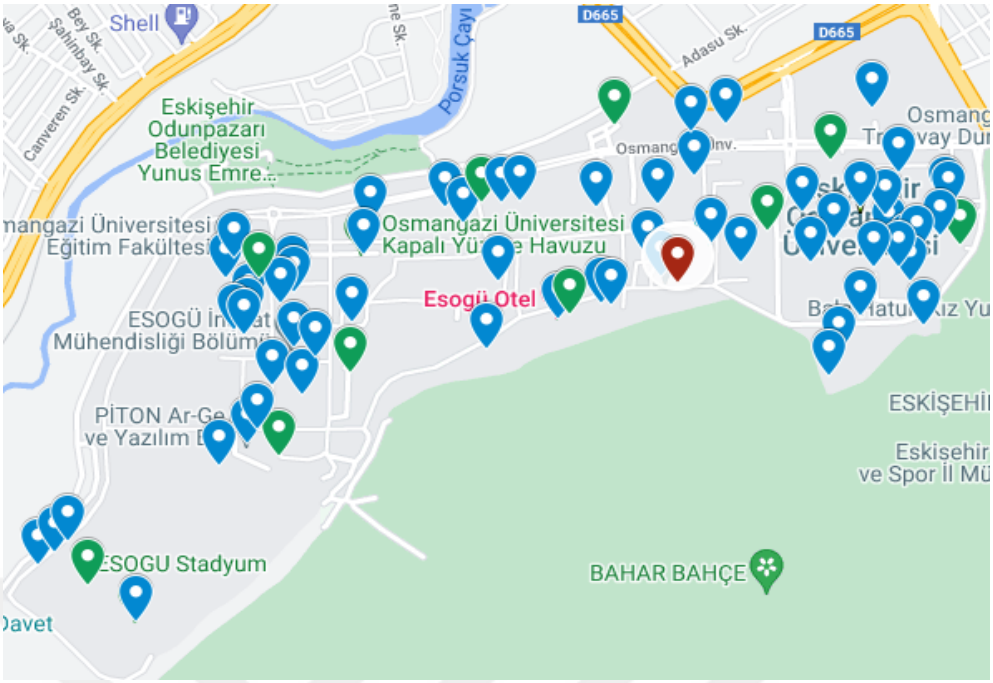
Çalışmanın temel girdileri ve uygulanan çözüm yaklaşımları sistematik biçimde sunulması amacıyla iki bölüm altında ele alınmıştır. İlk bölümde, tez kapsamındaki çözülen problemler için özgün olarak geliştirilmiş veri setlerinin özellikleri, oluşturulma süreçleri ve sağladıkları olanaklar ayrıntılı olarak tanıtılmaktadır. İkinci bölümde problemlerin çözümünde kullanılan önerilen yöntemlerin kuramsal temelleri ve algoritmik ayrıntıları ve açıklanmaktadır.

3.1. Materyal

Bu alt başlıkta, tez kapsamında geliştirilen ESOGU-CEVRPTW ve ESOGU-EVRP-PDP-TW veri setlerinin tasarım ilkeleri ve içerikleri ayrıntılandırılmaktadır. Her iki veri seti de gerçek dünyadan elde edilen coğrafi konum, müşteri talebi, zaman penceresi ve araç parametreleri göz önüne alınarak oluşturulmuştur. Veri setlerinin geçerliliği, hem literatürde matematiksel modellerle ve pekiştirmeli öğrenme yöntemleri ile doğrulanmıştır.

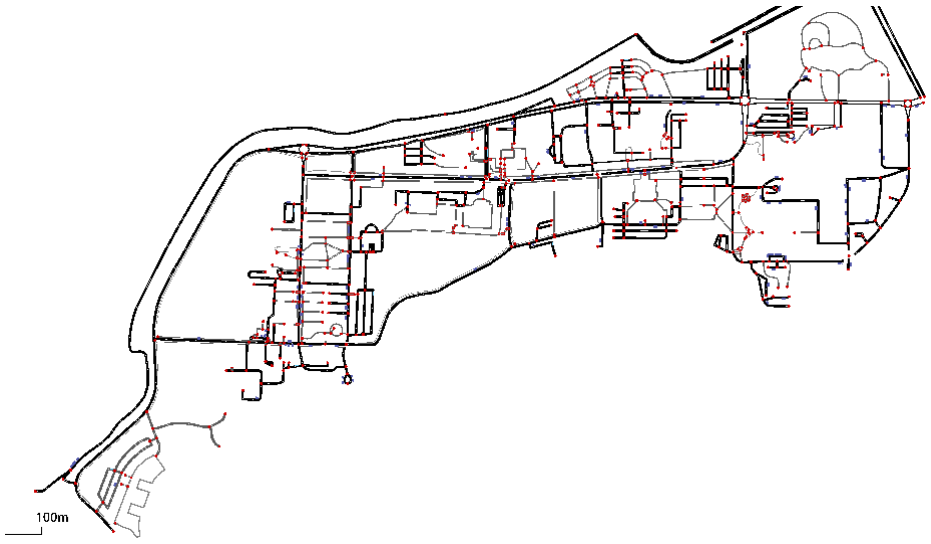
3.1.1. Oluşturulan ESOGU-CEVRPTW veri seti

Eskişehir Osmangazi Üniversitesi (ESOGU) Meşelik Kampüsü için ESOGU-CEVRPTW adlı yeni bir veri seti oluşturulmuştur. Oluşturulan veri seti Mendeley Data'ya yüklenmiştir (Aslan vd., 2025). Veri setinin ayrıntılı çözümleri ise Aslan Yıldız vd., (2025) çalışmasında yer almaktadır. Bu bölüm, veri seti oluşturma sürecinin ayrıntılı bir açıklamasını sunar. Oluşturulan veri seti, literatürde yaygın olarak bulunanlardan farklı olarak gerçek dünya verilerinden oluşur. Veri setindeki depo, müşteri ve şarj istasyonlarının konumları, Eskişehir Osmangazi Üniversitesi Meşelik Kampüsü'ndeki coğrafi konumlara dayanmaktadır. Şekil 3.1'deki harita, belirtilen noktalara farklı renklerde coğrafi işaretleyiciler eklenerek Google Haritalar kullanılarak oluşturulmuştur. Şekil 3.1'de mavi işaretli noktalar müşterileri, kırmızı işaretli nokta depoyu ve yeşil işaretli noktalar şarj istasyonlarını göstermektedir.



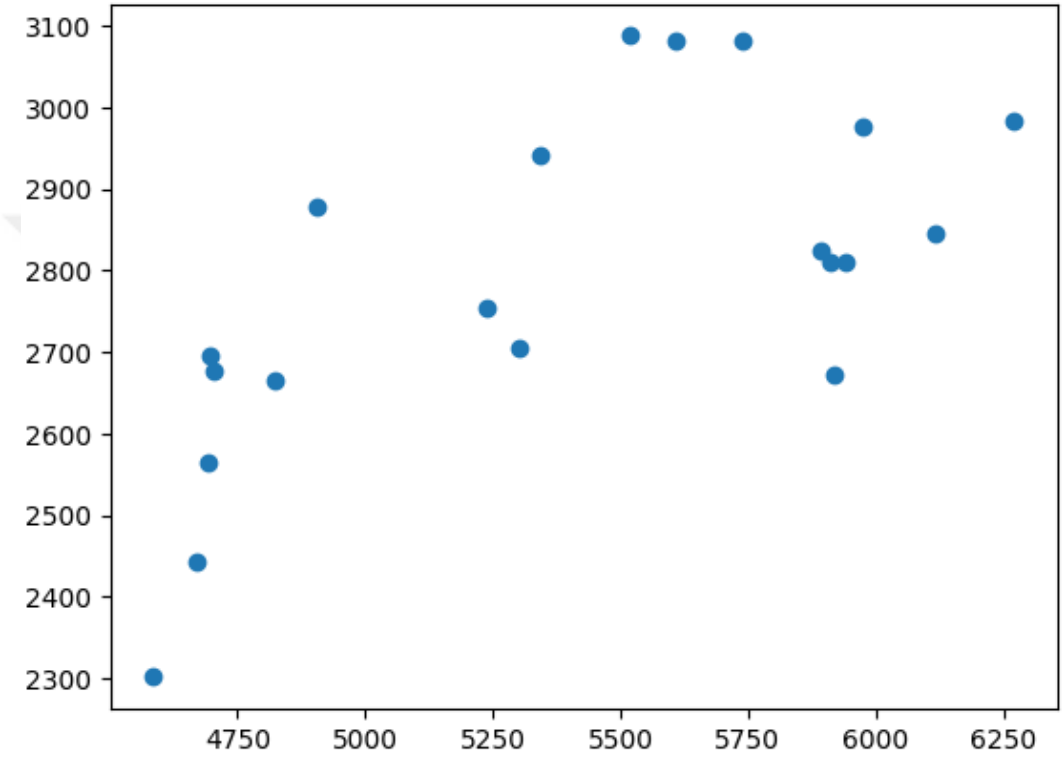
Şekil 3.1. ESOGÜ haritası

Veri seti oluşturulurken ESOGÜ kampüsünün yol ağı SUMO (Simulation of Urban Mobility) simülasyon ortamına aktarılmıştır. SUMO ortamında araç ve yaya yollarının mesafesine doğrudan ulaşılabilir. ESOGÜ kampüsünde yer alan bina bilgileri SUMO'daki kampüs haritası üzerine eklenmiştir. Kampüsteki binaları haritada göstermek için, tüm binaların giriş noktasına konteyner durak noktası (containerStop) eklenmiştir. ESOGÜ kampüsünün SUMO ortamındaki gösterimi Şekil 3.2'de gösterilmektedir. Şekil 3.2'de mavi konteyner noktaları müşterileri, yeşil dikdörtgenler ise şarj istasyonlarını göstermektedir.



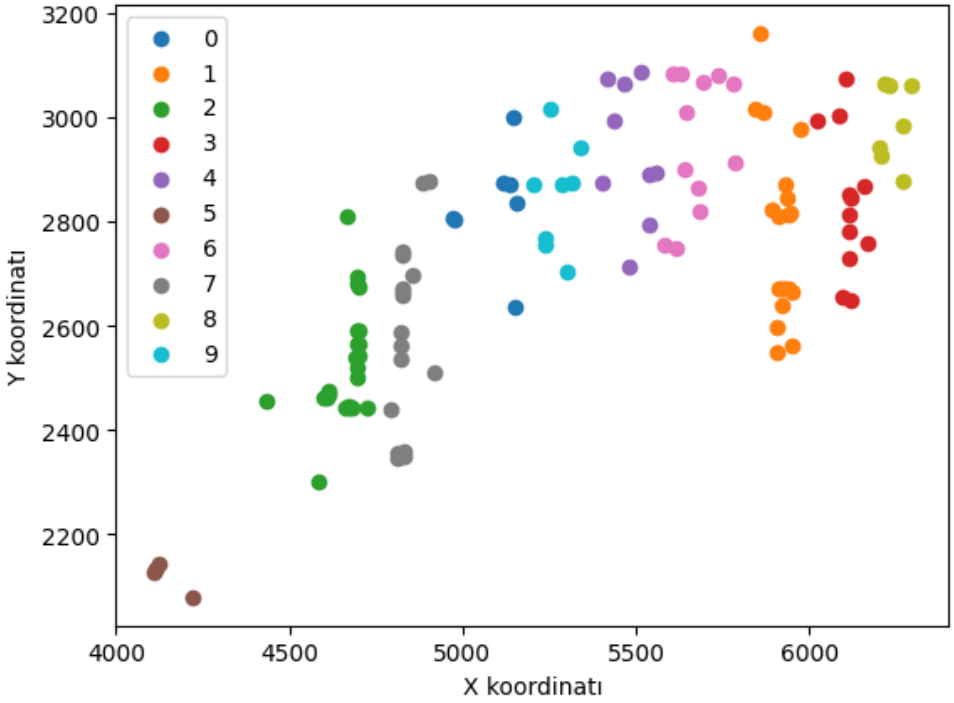
Şekil 3.2. ESOGÜ haritasının netedit ortamında gösterilmesi

Veri seti oluşturulurken 5, 10, 20, 40, 60 müşteri seçilerek farklı büyüklüklerde örnek problemler oluşturulmuştur. Ayrıca, müşteri konumlarının coğrafi dağılımına bağlı olarak üç sınıf mevcuttur. Bu sınıflar rastgele müşteri dağılımı (R), kümelenmiş müşteri dağılımı (C) ve her ikisinin bir karışımı olan müşteri dağılımıdır (RC). Rastgele müşteri dağılımı için kampüste yer alan 118 müşteri arasından rasgele 20 müşterinin seçildiği durum Şekil 3.3'te gösterilmiştir.



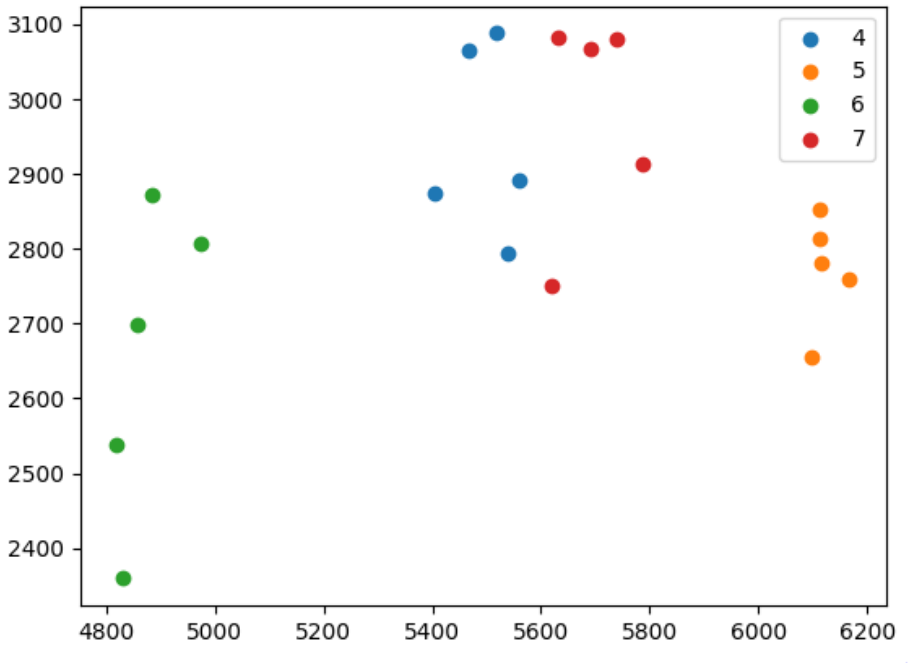
Şekil 3.3. ESOGÜ kampüsündeki rastgele müşteri dağılımı örneği

Kümelenmiş müşteri dağılımı için öncelikle kampüs ortamındaki müşteriler K-Means algoritması ile kümelere ayrılmıştır. Şekil 3.4'te müşterilerin kümelere ayrılmış hali gösterilmektedir.



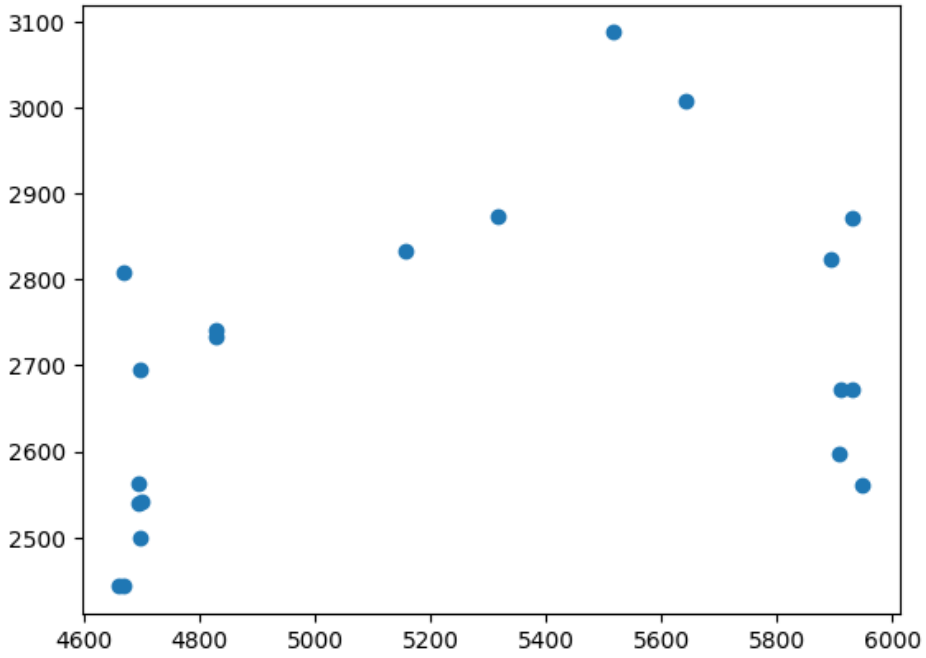
Şekil 3.4. ESOGÜ kampüsündeki müşteri noktalarının k-means ile kümelenmesi

Daha sonra bu kümelere ayrılmış müşteriler arasından 20 müşteri seçilerek Şekil 3.5'te gösterildiği gibi 20 müşterili kümelenmiş müşteri dağılımı oluşturulmuştur.



Şekil 3.5. ESOGÜ kampüsündeki kümelenmiş müşteri dağılımı örneği

Her iki farklı tipteki müşteri dağılımı birleştirilerek Şekil 3.6'daki rastgele-kümelenmiş müşteri dağılımı oluşturulur.



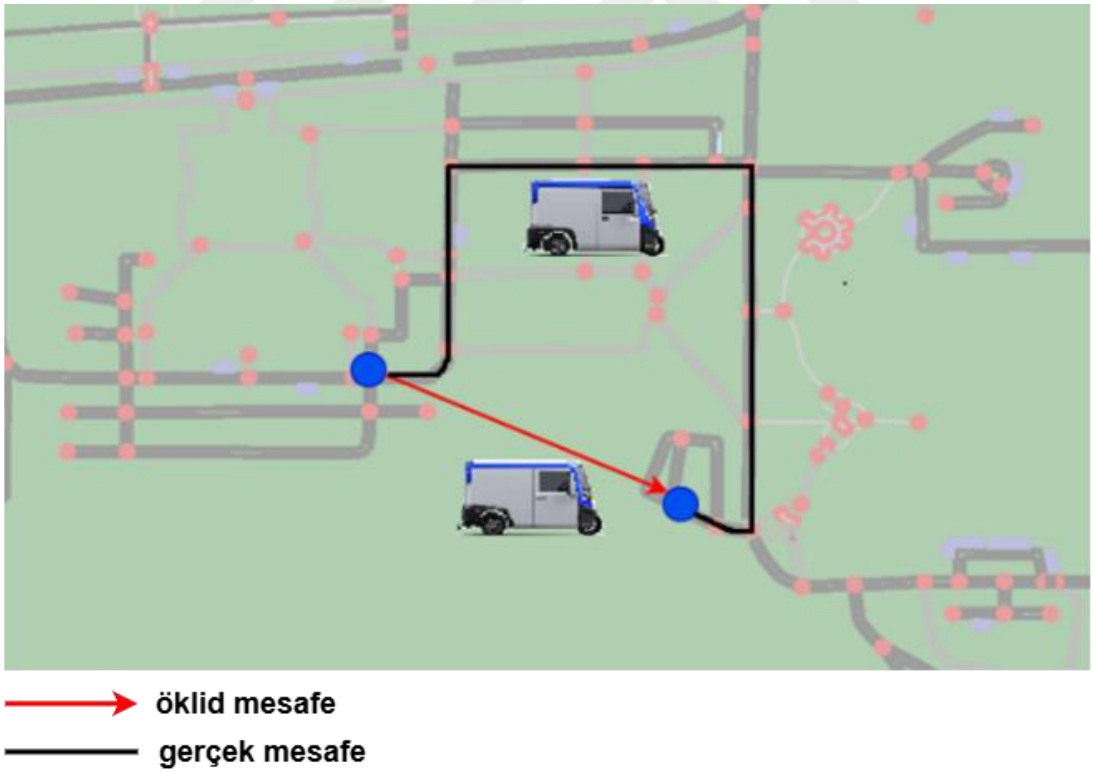
Şekil 3.6. ESOGÜ kampüsündeki rastgele -kümelenmiş müşteri dağılımı örneği

Veri setindeki müşterilerin tanımlanmasının ardından kampüsteki 10 park yeri şarj istasyonu olarak belirlenmiştir. Bu veri setindeki parametreler gerçek dünya problemlerini yansıtacak şekilde oluşturulmuştur. Veri setindeki zaman penceresi kısıtını oluşturmak için her müşteri için "servise en erken başlama zamanı" ve "servise en geç başlama zamanı" yer almaktadır. Zaman penceresi kısıtı belirlenirken, başlangıçta makine planlama için kullanılan Kim vd., (2004) çalışmasındaki zaman penceresi türetme mekanizması araç rotalama problemine uyarlanmıştır. Zaman penceresindeki, "servise en erken başlama zamanı " ve "servise en geç başlama zamanı", Denklem 3.1'de sunulan formül kullanılarak ayrık düzgün bir dağılımdan türetilir. Denklemdeki t_i depodan müşteriye normal hızda seyahat süresini, s_i, müşteri için teslim alma/teslim etme görevinin süresini (araçtan gidip gelme süreleri ve teslim alma/teslim etme süreleri dahil) ve k, araç sayısını temsil eder. Geniş, orta ve dar zaman penceresi, (α, β) değerleri kullanılarak belirlenir. Örneğin, dar zaman penceresi için $(\alpha, \beta) = (0, 1.2)$ seçilir. Formül 118 müşteri için uygulandığında ve $(\alpha, \beta) = (0, 1.2)$ olduğunda, zaman penceresi [0,1789] olarak belirlenir.

$$\left[\alpha \times \frac{\sum(t_i + s_i)}{k}, \beta \times \frac{\sum(t_i + s_i)}{k}\right] \tag{3.1}$$

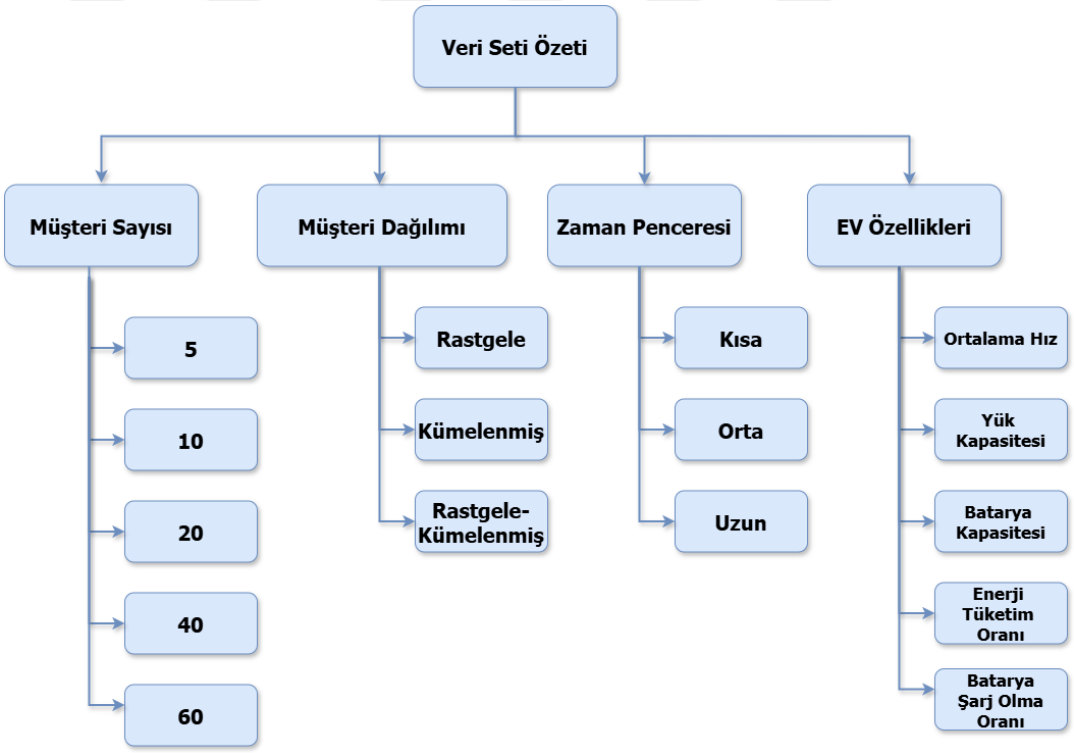
"Seyahat süresi", aracın depo ile müşteri arasındaki mesafeyi katettiği süreyi ifade eder. Bu süreyi hesaplamak için Google Mesafe Matrisi API'si kullanılır. Mesafe Matrisi API'si belirli başlangıç ve bitiş noktaları için seyahat mesafesini ve süresini hesaplayabilir. Bu hesaplamalarda Öklid mesafesi yerine gerçek mesafe kullanılır. “Servis süresi”, bir kişinin araç ile müşteriye en yakın noktaya ulaşmasından sonra, yaya yolu ile ürünü müşteriye teslim etmesi ve araca geri dönmesi için gereken toplam süreye karşılık gelir. Burada kişinin 100 m’yi 120 sn de yürüdüğü varsayılmıştır. Ayrıca katlar arası mesafe 60 sn çıkış ve 60 sn iniş olmak üzere toplam 120 sn olarak düşünülmüştür. Problemdeki her müşteri için “Talep” [19,95] arasındaki kesikli düzgün bir dağılımdan üretilmiştir.

Veri seti, ESOGU Kampüsü'ndeki tüm müşteriler ve şarj istasyonları arasındaki en kısa mesafeleri içeren bir mesafe matrisi içerir. Mesafe matrisi SUMO kullanılarak oluşturulmuştur. Şekil 3.7, gerçek mesafeler ve Öklid mesafeleri kullanılarak hesaplanan en kısa yollar arasındaki bir karşılaştırmayı sunar.



Şekil 3.7. Öklid mesafe ve gerçek mesafenin karşılaştırılması

Öklid mesafesi gerçek mesafeyi doğru bir şekilde temsil etmediğinden, gerçek dünya senaryolarının karmaşıklıklarını yakalamada başarısız olur. Önerilen veri seti, binalar, sokaklar gibi özellikleri dahil ederek bu ihtiyacı karşılar ve böylece birden fazla özniteliğe sahip gerçekçi bir ortamda rota planlamasını mümkün kılar. Gerçek dünya koşulları için tasarlanan bu veri seti, gelişmiş lojistik yöntemlerinin uygulanmasını kolaylaştırır. Özellikle yol özellikleri (örneğin, tek yönlü, çift yönlü, eğimli, dar), trafik düzenlemeleri ve diğer faktörler gibi çevresel bilgilere dayanan yaklaşımları kullanan araştırmacılar için değerli bir kaynak sağlar. Dahası, veri seti, trafik kazaları nedeniyle yol kapanmaları gibi senaryoları simüle etme yeteneği sunarak bu tür durumların derin öğrenme modellerinde modellenmesine olanak tanır.



Şekil 3.8. ESOGU-CEVRPTW veri seti özeti

Şekil 3.8, ESOGU-CEVRPTW veri setindeki örneklerin bir açıklamasını sunar. Her örnekteki müşteri sayısı, müşterilerin dağılımı ve uygulanan zaman penceresi hakkında bilgi içerir. Veri setinden bir örnek problem Çizelge 3.1'de sunulmuştur.

Çizelge 3.1. ESOGU-CEVRPTW veri setinden bir örnek

Düğüm ID	Lokasyon bilgisi	Servise en erken başlama zamanı	Servise en geç başlama zamanı	Servis süresi	Talep
32	39.752487 - 30.488123	1322	1378	180	76
19	39.748319 - 30.489034	296	350	120	57
31	39.752941 - 30.483072	1372	1448	120	57
60A/2	39.749343 - 30.474854	764	840	240	38
75	39.746589 - 30.474760	424	487	120	38

Örnekteki her düğümün bir id değeri, enlem ve boylamı, servise en erken başlama zamanı, servise en geç başlama zamanı, talep ve servis zamanı vardır. Veri setindeki örnekler için, CEVRP'nin hem zaman pencereli hem de zaman penceresiz versiyonları, optimal çözümler elde etmek için matematiksel bir model kullanılarak çözülmüştür.

3.1.2. Oluşturulan ESOGU-EVRP-PDP-TW veri seti

Literatürdeki veri setlerinden sadece ikisi (Goeke, 2018; Agrali, 2021) topla-dağıt elektrikli araç rotalama problemine uygundur (Çizelge 3.1). Goeke, (2018) veri seti Schneider vd., (2014)veri setinden türetilmiştir. Veri setindeki talep kısımları "+"/ "-" diye gruplandırılarak pozitifler toplama (pickup) negatifler ise dağıtım (delivery) noktalarını ifade etmektedir. Agrali., (2021) veri seti oluşturulurken Solomon, (1987) veri seti kullanılmıştır. Bu veri setinde birden fazla depo mevcuttur ve transfer noktaları içeren eşzamanlı topla dağıt problemi içindir. Depolar, araçların başlangıç ve son düğümleridir ve her araç farklı bir depo kullanabilir.

ESOGU-EVRP-PDP-TW veri seti oluşturulurken depo, müşteri ve şarj istasyonlarının konumları, Eskişehir Osmangazi Üniversitesi Meşelik Kampüsü'nden seçilmiştir. Veri seti problem boyutunun etkisini inceleyebilmek amacıyla 5, 10, 20, 40, 60, 80 ve 100 müşteriden oluşan yedi farklı örnek seti içermektedir. Her bir müşteri büyüklüğü için, rastgele dağılım (R), kümelenmiş dağılım (C) ve rastgele-kümelenmiş karışık dağılım

(RC) dan örnekler yer almaktadır. Veri setindeki şarj istasyonları kampüs sınırları içindeki on park yeri olarak belirlenmiştir. Tüm parametreler, gerçek-dünya operasyonlarını olabildiğince doğru yansıtacak şekilde oluşturulmuştur. Veri setindeki müşteri grubunun %50'si için toplama (alma), kalan %50'si için teslimat (bırakma) talebi tanımlanmıştır. Problemdeki her müşteri için "Talep" değeri [5,20] arasındaki kesikli düzgün bir dağılımdan üretilir. Veri setindeki zamanla ilişkili "servis süresi", "servise en erken başlama zamanı" ve "servise en geç başlama zamanı" değerleri ESOGU-CEVRPTW veri setindeki uygulanan aynı kurallar izlenerek türetilmiştir.

ESOGU-EVRP-PDP-TW veri setinde de dar, orta ve geniş olmak üzere üç farklı zaman penceresi kısıtına ait örnekler vardır. Veri setinden kümelenmiş dağılım ile seçilen beş müşterili ve orta zaman pencereli bir örnek problem Çizelge 3.2'de sunulmuştur.

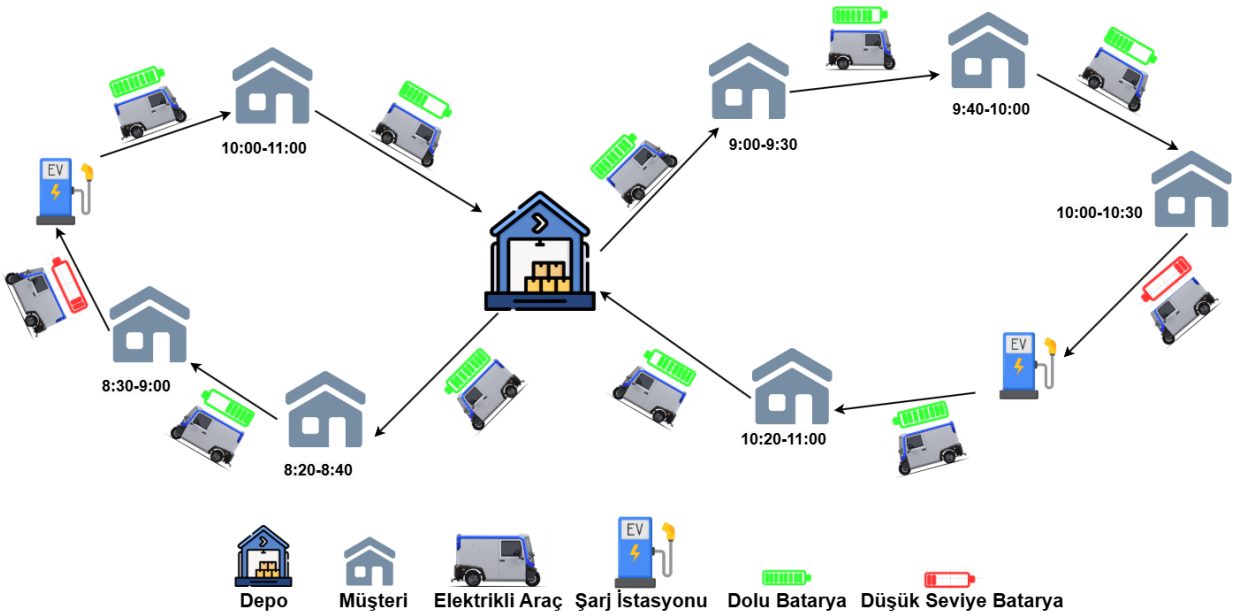
Çizelge 3.2. ESOGU-EVRP-PDP-TW veri setinden bir örnek

Düğüm ID	**Düğüm tipi**	**Enlem**	**Boylam**	**Dağıtım talebi**	**Toplama talebi**	**Servis süresi**	**Servise en erken başlama zamanı**	**Servise en geç başlama zamanı**
48	d	39750124	30485116	0	0	0	0	7000
10	c	39750316	30491958	19	0	120	81	6301
13	c	39750779	30491320	12	0	120	106	6734
14	c	39750497	30491334	17	0	120	87	5881
154	c	39750448	30478030	0	19	240	732	6070
22A	c	39750725	30489306	0	20	240	227	6336
cs1	cs	39744075	30469333	0	0	0	0	7000
cs2	cs	39748305	30477105	0	0	0	0	7000
cs3	cs	39746605	30474623	0	0	0	0	7000
cs4	cs	39753933	30484572	0	0	0	0	7000
cs5	cs	39751670	30480422	0	0	0	0	7000
cs6	cs	39751182	30487697	0	0	0	0	7000
cs7	cs	39752908	30489350	0	0	0	0	7000
cs8	cs	39750889	30492715	0	0	0	0	7000
cs9	cs	39750132	30474637	0	0	0	0	7000
cs10	cs	39749569	30482477	0	0	0	0	7000

Örnekteki her düğümün bir id değeri, düğüm tipi, enlem ve boylamı, toplama veya dağıtım talebi, servis zamanı, servise en erken başlama zamanı ve servise en geç başlama zamanı vardır. Veri setindeki örnekler için, EVRP-PDP-TW probleminin optimal çözümleri matematiksel bir model kullanılarak çözülmüştür.

3.2. Kapasiteli Elektrikli Araç Rotalama Problemi ve Matematiksel Modeli

Bu bölümde, Zaman Pencereli Kapasiteli Elektrikli Araç Rotalama Problemi ele alınmaktadır. CEVRPTW bir kombinatoryal optimizasyon problemidir. Problemde, belirli bir yük ve batarya kapasitesine sahip EV'ler tam şarjlı olarak müşterilere hizmet etmek için depodan ayrılırlar ve rotaları boyunca müşterileri ziyaret ederler. Gerekirse bataryalarını şarj etmek için şarj istasyonuna uğrarlar (Şekil 3.9). EV bir şarj istasyonunu ziyaret ettiğinde, görevini tamamlamak için tam şarj stratejisi ile şarj edilecektir. EV rotanın sonunda depoya geri dönmelidir. EV'ler için amaç, tüm müşteri taleplerinin karşılanacağı ve filonun toplam mesafesinin veya seyahat süresinin en aza indirileceği rotaları bulmaktır.



Şekil 3.9. CEVRPTW için bir örnek çözüm

CEVRPTW'de başlangıçta tam yönlendirilmiş bir çizge $G = (V, E)$ tanımlanır. $V = \{0 \cup C \cup F\}$ çizgedeki düğüm kümesini temsil eder. İfadedeki 0 depoyu, C müşterileri ve F şarj istasyonlarını temsil eder. $E = \{(i,j)|\, i,j \in V, i \neq j\}$ ise çizgedeki ayrıt kümesidir. Buradaki i ve j düğümleri arasındaki her ayrıtın d_{ij} olarak gösterilen bir mesafe değeri vardır. $[e_i, l_i]$ müşteri i 'nin zaman penceresini temsil eder. Bir elektrikli araç bir ayrıtta seyahat ettiğinde, o kenarda harcayacağı enerji $e_{ij} = h(r).d_{ij}$ formülü kullanılarak hesaplanır. Formüldeki $h(r)$ enerji tüketim oranını ifade eder.

Bu problemde, müşterilere zaman pencereleri içinde hizmet verilirken tüm elektrikli araçların toplam seyahat mesafesini en aza indirmek amaçlamaktadır. Her müşteriye yalnızca bir araçla hizmet verilebileceği kısıtlaması vardır. EV'ler, tüm müşterilerin taleplerini karşıladıktan sonra depoya geri dönmelidir. EV'lerin şarj istasyonlarında her zaman tam olarak şarj edildiğini varsayımı kabul edilmiştir.

Literatürde, Schneider vd., (2014) CEVRPTW için bir matematiksel model önermiştir. Model:

Kümeler

$0, N+1$: Depo

F' : Şarj istasyonlarına yapılan ziyaretler kümesi

F_0': Depo 0 dahil olmak üzere şarj ziyaretleri kümesi

V: Müşteri dizisi $V = \{1, \ldots., N\}$

V_0: Depo 0 dahil olma üzere müşteri kümesi

V': Şarj istasyonlarına yapılan ziyaretleri de içeren müşteri kümesi, $V' = V \cup F'$

V_0': Depo 0, şarj istasyonu ve müşteri kümesi: $V_0' = V' \cup \{0\}$

V_{N+1}': Depo N+1, şarj istasyonu ve müşteri kümesi: $V_{N+1}' = V' \cup \{N+1\}$

$V_{0,N+1}'$: Depo 0, depo N+1, şarj istasyonu ve müşteri kümesi: $V_{0,N+1}' = V' \cup \{0\} \cup \{N+1\}$

Parametreler

q_i: i düğümünün talebi, 0 eğer $i \notin V$

d_{ij}: i ve j düğümü arasındaki mesafe

t_{ij}: i ve j düğümü arasındaki seyahat süresi

e_i: i düğümünün servise en erken başlama zamanı

l_i: i düğümünün servise en geç başlama zamanı

s_i: i düğümünün servis süresi $(s_0, s_{N+1} = 0)$

C: Araç kapasitesi

g: Şarj oranı

h: Şarj tüketim oranı

Karar Değişkenleri

P_i: i düğümüne varıldığındaki zaman

u_i: i düğümüne varıldığında aracın kalan yük kapasitesi

y_i: i düğümüne varıldığında aracın kalan batarya kapasitesi

x_{ij}: Eğer araç i düğümünden j düğümüne geçiyorsa 1, diğer durumlarda 0

CEVRPTW'nin matematiksel modeli, aşağıdaki gibi Karma Tamsayılı Doğrusal Programlama (MILP) olarak formüle edilmiştir:

Amaç Fonksiyonu:

$$\sum_{i \epsilon V_0', j \epsilon V_{N+1}', i \neq j} d_{ij} x_{ij} \tag{3.2}$$

Kısıtlar

$$\sum_{j \epsilon V_{N+1}', i \neq j} x_{ij} = 1 \ \ \forall i \epsilon V \tag{3.3}$$

$$\sum_{j \epsilon V_{N+1}', i \neq j} x_{ij} \leq 1 \ \ \forall i \epsilon F' \tag{3.4}$$

$$\sum_{i \epsilon V_{N+1}', i \neq j} x_{ij} - \sum_{i \epsilon V_0', i \neq j} x_{ij} = 0 \ \ \forall j \epsilon V' \tag{3.5}$$

$$P_i + (t_{ij} + s_i) x_{ij} - l_0 (1 - x_{ij}) \leq P_j \quad \forall i \epsilon V_0, \forall j \epsilon V_{N+1}', i \neq j \tag{3.6}$$

$$P_i + t_{ij} x_{ij} + g((Qc) - y_i) - (l_0 + M * g(Qc))(1 - x_{ij}) \leq P_j \quad \forall i \epsilon F', \forall j \epsilon V_{N+1}', i \neq j \tag{3.7}$$

$$e_j \leq P_j \leq l_j \ \ \forall j \epsilon V_{0,N+1}' \tag{3.8}$$

$$0 \leq u_j \leq u_i - q_i x_{ij} + C(1 - x_{ij}) \quad \forall i \in V_0', \forall j \in V_{N+1}', i \neq j \tag{3.9}$$

$$0 \leq u_0 \leq C \tag{3.10}$$

$$0 \leq y_j \leq y_i - (d_{ij}h)x_{ij} + Qc(1 - x_{ij}) \quad \forall i \in V, \forall j \in V_{N+1}', i \neq j \tag{3.11}$$

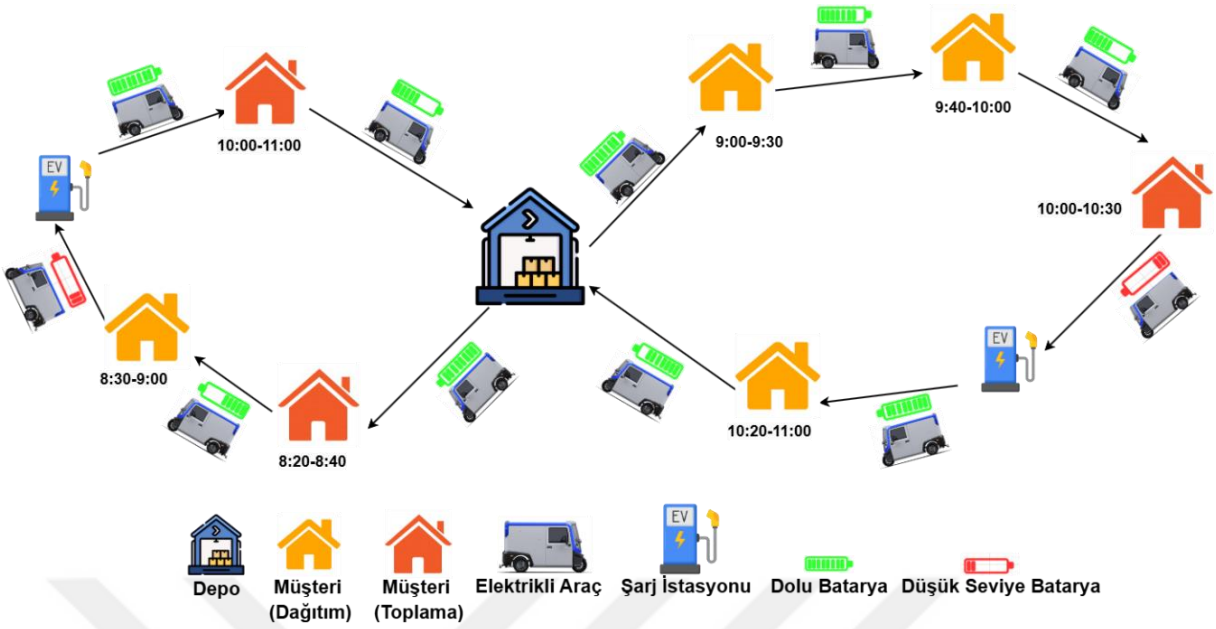
$$0 \leq y_j \leq Qc - (d_{ij}h)x_{ij} \quad \forall i \in F_0', \forall j \in V_{N+1}', i \neq j \tag{3.12}$$

$$x_{ij} \epsilon [0,1] \ \forall i \in V_0', j \in V_{N+1}', i \neq j \tag{3.13}$$

Amaç fonksiyonu toplam seyahat mesafesini en küçüklemektir ve denklem 3.2'de ifade edilmiştir. Kısıt (3.3), her müşteri düğümüne hizmet sağlanmasını garanti ederken, kısıt (3.4), şarj istasyonunun ziyaret edilebilmesini garanti eder. Kısıt (3.5) ise, gereksiz alt turların oluşmasını önler. Kısıt (3.6) ve (3.17), düğümlere varıldığı zamanı kontrol içindir. Kısıt (3.8), zaman penceresi koşullarını ifade eder. Kısıt (3.9) ve (3.10), aracın yükünü kontrol eden kısıtlamalardır. Burada, araç, kapasitesini aşmayan bir yük ile depodan ayrılacaktır. Kısıt (3.11) ve (3.12), aracın şarj durumunu kontrol eder. Kısıt (3.13), x_{ij}'nin 0 ya da 1 değer alacağını ifade eder.

3.3. Topla Dağıt Elektrikli Araç Rotalama Problemi ve Matematiksel Modeli

Bu bölümde EVRP problemlerinden biri olan zaman pencereli topla dağıt elektrikli araç rotalama problem ele alınmaktadır. EVRP-PDP-TW, bir bölgeye dağılmış müşterilerin teslimat ve toplama taleplerinin tanımlı zaman pencereleri içinde ve elektrikli araçların (EV) batarya kısıtları gözetilerek karşılandığı kombinatoryal bir optimizasyon problemidir. Problemde, belirli bir batarya kapasitesine sahip EV'ler tam şarjlı olarak müşterilere hizmet etmek için depodan ayrılırlar ve rotaları boyunca bataryalarını şarj etmek için istasyonları ziyaret ederler. EV bir şarj istasyonunu ziyaret ettiğinde, görevini tamamlamak için tam şarjlı olarak şarj edilecektir. EV rotanın sonunda depoya geri dönmelidir. EV'ler için amaç, tüm müşteri taleplerinin karşılanacağı ve filonun toplam mesafesinin veya seyahat süresinin en aza indirileceği rotaları bulmaktır. EVRP-PDP-TW problem Şekil 3.10'da gösterilmektedir.



Şekil 3.10. Zaman pencereli topla dağıt elektrikli araç rotalama problemi

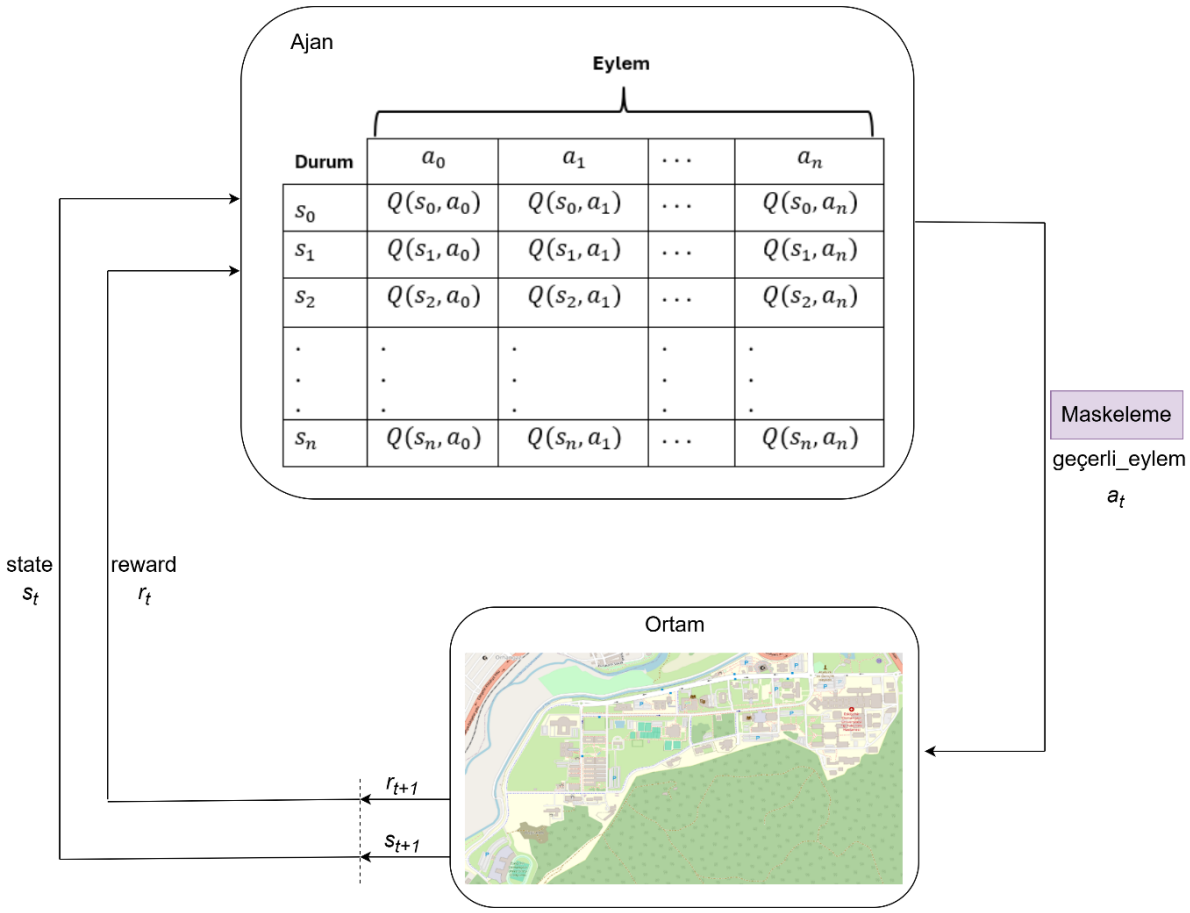
Problemde ortam düğüm ve kenarlardan oluşan tam yönlendirilmiş bir çizge şeklinde ifade edilir ve $G = (V, E)$ olarak tanımlanır. $V = \{0 \cup C \cup F\}$ çizgedeki düğüm kümesini temsil eder. İfadedeki 0 depoyu, C müşteri küümesini ve F şarj istasyonu kümesini temsil eder. $E = \{(i,j) | \, i,j \in V, i \neq j\}$ ise çizgedeki kenar kümesidir. Buradaki i ve j düğümleri arasındaki her kenarın d_{ij} olarak gösterilen bir seyahat mesafesi değeri vardır. Müşteri i için belirlenen zaman penceresi $\lfloor e_i, l_i \rfloor$ ile gösterilir ve e_i müşteriye hizmetin başlayabileceği en erken zamanı, l_i ise hizmetin başlamak zorunda olduğu en geç zamanı ifade eder. EV belirlenen l_i zamanından sorra servise başlayamaz ve e_i 'den önce servise başlarsa e_i zamanına kadar beklemek zorundadır. Bir elektrikli araç bir kenarda seyahat ettiğinde, o kenarda harcayacağı enerji $e_{ij} = h(r).d_{ij}$ formülü kullanılarak hesaplanır. Formüldeki $h(r)$ enerji tüketim oranını ifade eder.

EVRP-PDP-TW'da amaç, müşteri zaman pencerelerine uyumlu bir şekilde hizmet sunulurken filo genelindeki elektrikli araçların kat ettiği toplam mesafeyi en düşük düzeye indirmektir. Her müşteriye yalnızca bir araç tarafından hizmet verilebileceği kısıtı bulunmakta ve araçlar tüm müşteri taleplerini tamamladıktan sonra depoya geri dönmek zorundadır. Optimal sonuçlar, CEVRP-TW matematiksel modeli revize edilerek elde edilmiştir. Senaryo kampüs içinde birimler arası evrak vb. taşımaların elektrikli araçla yapılması olarak kurulmuştur. Taşınacak yüklerin hafif olması sebebiyle araç yük kapasite

kısıtı ihmal edilmiştir. Araç batarya kısıtı ve zaman penceresi kısıtlarının önem kazandığı problem için test problemleri CPLEX çözücü ile çözülerek optimal sonuçlar elde edilmiştir.

3.4. Q-Öğrenme ile Kapasiteli Elektrikli Araç Rotalama Problemi

Q-öğrenme algoritması, Bölüm 2.1'de sınıflandırması verilen Pekiştirmeli Öğrenme algoritmaları arasından, modelden bağımsız ve değer tabanlı olan RL algoritmaları sınıfında yer almaktadır. CEVRP için önerilen Q-Öğrenme algoritması Şekil 3.11'de gösterilmektedir



Şekil 3.11. Önerilen Q-Öğrenme algoritması

Q-öğrenme algoritmasında, bir RL ajanı bir periyot boyunca bir ortamla etkileşime girer. Her zaman adımı t 'de, ajan bir durum alanı S 'den bir durum s_t alır ve bir eylem alanı A 'dan bir eylem a_t seçer. Önerilen yaklaşımda, durum temsili mevcut düğüm id değeri, mevcut batarya kapasitesi, mevcut yük kapasitesi bilgilerinden oluşur. Eylem a_t, EV'nin adım t 'de gideceği düğümü belirler. Eylem dizisi ilk adım 0'dan son adım T'ye kadar üretilir.

Burada, a_0 depodan ayrılma eylemidir, a_1 ilk adımda seçilen eylemdir ve a_T depoya dönme eylemidir. Eylem uzayı A, bir dizi ayrık eylemi temsil eder. Ajan bir müşteriye, bir şarj istasyonuna veya bir depoya gitme eylemlerinden birine karar verir. Eylem seçiminde maskeleme yöntemi ile tüm eylemler arasından o anki durum için geçerli eylemler belirlenir ve seçim bu geçerli eylemler arasından yapılır. Eylem seçimi sürecinde kullanılan maskeleme yöntemi ile, ajan tarafından tüm eylemler arasından, yalnızca mevcut durumda gerçekten uygulanabilir olanları filtrelenir. Ajan o durum için sadece geçerli eylemler arasından bir seçim yapar.

Şekil 3.11'de genel görünümü verilen Q tablosunun detaylı gösterimi Şekil 3.12'de yer almaktadır. Tablo boyutu "durum_sayısı x eylem_sayısı" şeklinde tanımlanır ve problemin boyutuna göre değişir. Örneğin tek bir depo, beş müşteri ve 10 şarj istasyonundan oluşan bir problem için tablo boyutu 16x16=256 olarak hesaplanır.

		Eylem: a_0 (D)	a_1 (C)	...	a_n (C)	a_{n+1} (CS)	...	a_{n+10} (CS)
	Durum							
D	s_0	$Q(s_0, a_0)$	$Q(s_0, a_1)$	...	$Q(s_0, a_n)$	$Q(s_0, a_{n+1})$	...	$Q(s_0, a_{n+10})$
C	s_1	$Q(s_1, a_0)$	$Q(s_1, a_1)$	...	$Q(s_1, a_n)$	$Q(s_1, a_{n+1})$	...	$Q(s_1, a_{n+10})$
C	s_2	$Q(s_2, a_0)$	$Q(s_2, a_1)$	...	$Q(s_2, a_n)$	$Q(s_2, a_{n+1})$	...	$Q(s_2, a_{n+10})$
C	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
C	s_n	$Q(s_n, a_0)$	$Q(s_n, a_1)$	...	$Q(s_n, a_n)$	$Q(s_n, a_{n+1})$	...	$Q(s_n, a_{n+10})$
CS	s_{n+1}	$Q(s_{n+1}, a_0)$	$Q(s_{n+1}, a_1)$	...	$Q(s_{n+1}, a_n)$	$Q(s_{n+1}, a_{n+1})$	...	$Q(s_{n+1}, a_{n+10})$
CS	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
CS	s_{n+10}	$Q(s_{n+10}, a_0)$	$Q(s_{n+10}, a_1)$	...	$Q(s_{n+10}, a_n)$	$Q(s_{n+10}, a_{n+1})$	...	$Q(s_{n+10}, a_{n+10})$

Şekil 3.12. Oluşturulan Q Tablosu

- **s_0:** Tez kapsamında çözülen her problemde tek depo olduğu için s_0 depo ile ilgili bir durumu temsil eder. Bu durum için EV'nin mevcut yükü maksimum kapasiteye, mevcut batarya seviyesi ise tam dolu kapasiteye eşittir.

- $\boldsymbol{s_1 \ldots s_n}$: Müşteri sayısı n tane olduğu için s_1 ve s_n arası tüm durumlar müşteriler içindir. Bu durumlar için EV'nin mevcut yük miktarı bir önceki durumdaki yükten ilgili müşterinin talebinin düşülmesiyle hesaplanır. Batarya seviyesi ise önceki durumdan bu müşteriye kadar tüketilen enerji çıkarılarak güncellenir.
- $\boldsymbol{s_{n+1} \ldots s_{n+10}}$: Tüm problemlerdeki şarj istasyonu sayısı da sabit ve on tanedir. s_{n+1} ve s_{n+10} durumları ise şarj istasyonları ile ilgilidir. Bu durumlar için EV'nin mevcut yük miktarı bir önceki durumdaki değerle aynıdır. Batarya seviyesi ise istasyonda tam şarj edildiği varsayımıyla maksimum düzeydedir.

Bu yapı sayesinde her durum, EV'nin konumunu, kalan yükünü ve batarya seviyesini tutarlı biçimde yansıtarak Q-öğrenme algoritmasının gerektirdiği ayrıntılı durum-eylem eşlemelerini mümkün kılar. Q tablosundaki eylem gösterimi de durum indekslemesiyle tutarlı biçimdedir. a_0 aracı depoya yönlendiren eylemi temsil ederken, $a_1 \ldots a_n$ sırasıyla müşterilere gitmeyi ifade eder. Şarj istasyonuyla ilişkin eylemler ise $a_{n+1} \ldots a_{n+10}$ aralığında tanımlanmış olup, her biri belirli bir şarj istasyonuna yönelmeyi göstermektedir. Böylece eylem kümesi, aracın depoya dönüşü, müşteri hizmeti ve şarj işlemleri arasında açık bir ayrım sağlayacak şekilde numaralandırılmıştır.

Q-öğrenme algoritmasının önemli adımlarından biri keşif-sömürü dengesidir. Başlangıçta, ajan ortam hakkında yeterince bilgi sahibi değildir ve mevcut bilgiyi kullanmaktan ziyade keşfetmeye daha yatkındır. Zamanla, ajan ortamla etkileşime girdikçe, ortam hakkında bilgi edinir ve zamanla bu edinilen bilgiye güvenir. Bu bilgiyi kullanma ve yeni olasılıkları keşfetmeye devam etme arasında bir denge kurmak çok önemlidir. Bu denge olmadan, Q-değer fonksiyonu optimal çözüm yerine yerel bir minimuma yakınsayabilir. Q-öğrenmede keşif-sömürü dengesini sağlamak için en yaygın kullanılan yöntem **ε-** açgözlü stratejidir ve Denklem 3.14'te gösterilmiştir.

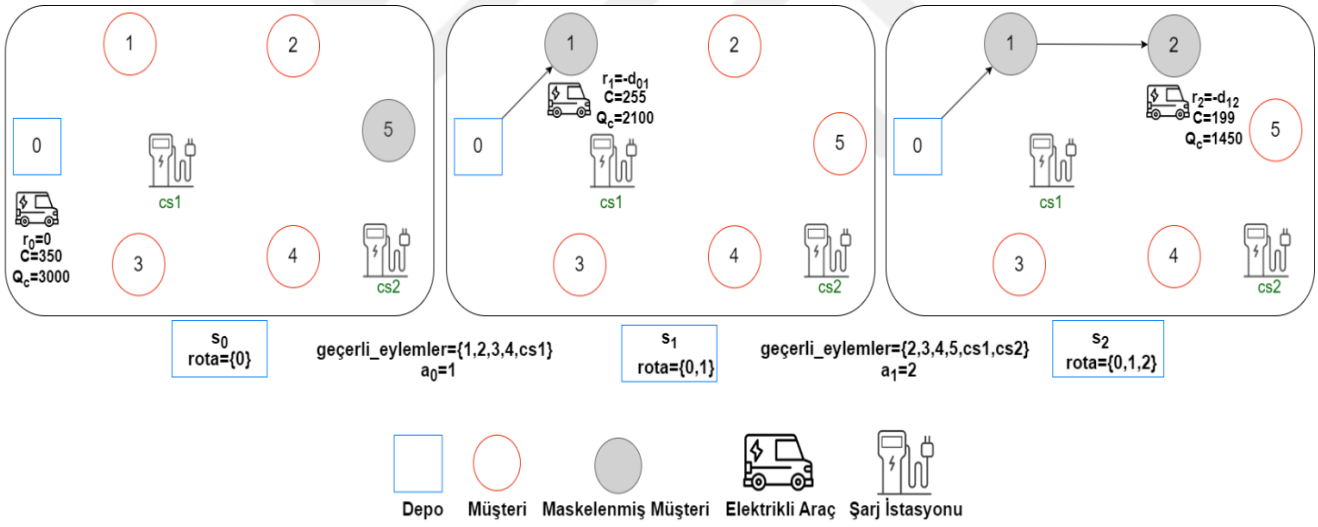
$$\pi(s_t) = \begin{cases} argmax_{a \in geçerli_eylemler} Q(s,a), & eğer\ rastgele\ [0,1] > \epsilon \\ rastgele\ a \in geçerli_eylemler, & değilse \end{cases} \tag{3.14}$$

Klasik Q-öğrenme algoritmasında kullanılan epsilon-açgözlü stratejisinin keşif ve sömürüyü dengelemek için sabit bir epsilon değerini kullandığı klasik yaklaşımın yerine, önerilen Q-öğrenme algoritmasında bir epsilon bozunma stratejisi benimsenmiştir ve strateji Denklem 3.15'te gösterilmiştir. Denklemdeki ϵ_0 epsilon değerinin başlangıç miktarını,

ϵ_{min} epsilon değerinin alt sınırını, Δ ise her adımdaki azalış miktarını temsil etmektedir. Denklem 3.15'te gösterilen epsilon bozunma stratejisi doğrusal bozunmayı ifade eder. Epsilon bu şekilde değerinin dinamik olarak ayarlaması, öğrenme süreci boyunca statik bir epsilondan daha etkili olabilen daha uyarlanabilir bir keşif-sömürü dengelenmesini sağlar. Bu güncellemeler, CEVRP'yi çözmede Q-öğrenmesinin performansını ve uygulanabilirliğini iyileştirmeyi amaçlamaktadır.

$$\epsilon_t = \max(\epsilon_{min}\,,\, \epsilon_0 - \Delta t) \tag{3.15}$$

Önerilen Q-öğrenme algoritmasının adım adım gösterimi Şekil 3.13'te gösterilmiştir. Algoritmanın ilk adımında, durum depo $(\boldsymbol{s_0})$ olarak belirlenir ve rota listesine eklenir. Bu ilk durumdan erişilebilir eylemler maskeleme yoluyla belirlenir ve geçerli_eylemler listesine dahil edilir. 5 numaralı müşteri bu adımda maskelenir, çünkü 3000 batarya kapasitesiyle $(\boldsymbol{s_0})$ konumundan ulaşılamaz.



Şekil 3.13. Q-Öğrenme algoritmasının adım adım gösterimi

CEVRP problemini çözmek için literatürdeki yer alan Q-öğrenme (Ottoni vd., 2022) algoritmasının son hali üzerinden çeşitli geliştirmeler yapılmıştır. Temel değişikliklerden biri, EV'nin o anda ulaşamadığı düğümleri maskeleyerek olası eylemler üreten bir maskeleme tekniğinin kullanılmasıdır. Bu, EV'nin her adımda yalnızca uygun olan eylemler arasından seçim yapmasını sağlar. Uygun olmayan eylemleri filtreleyerek, maskeleme mekanizması ajanın geçerli eylemlere odaklanmasını artırır ve böylece uygulanabilir

çözümlerin oluşturulmasına yardımcı olur. Maskeleme mekanizması, batarya seviyesi (SoC), araç kapasitesi ve daha önce ziyaret edilen müşteriler gibi faktörleri göz önünde bulundurarak mevcut duruma göre uygulanabilir eylemlerin bir listesini oluşturur. Yetersiz batarya veya kapasite nedeniyle ulaşılamayan bir düğüme gitmek gibi geçersiz eylemler maskelenir.

- Müşteri düğümü i daha önce ziyaret edilmişse maskelenir.
- Eğer düğüm i bir müşteriyi temsil ediyorsa ve müşterinin talebi EV'nin kalan kargo kapasitesini aşıyorsa, düğüm i maskelenir.
- Eğer EV şu anda müşteri i'de ise ve batarya seviyesi bir sonraki müşteri j'yi ziyaret etmek ve daha sonra depoya veya herhangi bir şarj istasyonuna dönmek için yeterli değilse, müşteri j maskelenir.
- Eğer EV şu anda herhangi bir i düğümünde ise ve batarya seviyesi bir sonraki şarj istasyonu j'yi ziyaret etmek ve daha sonra depoya veya herhangi bir şarj istasyonuna geri dönmek için yeterli değilse, şarj istasyonu j maskelenir.
- Eğer EV'nin yeni teslimatları tamamlamadan önce gereksiz yere şarj istasyonlarını veya depoyu tekrar ziyaret edeceği bir kendi kendine döngü tespit edilirse, daha verimli bir rotalamayı teşvik etmek için ilgili eylem maskelenir.

CEVRP'yi çözmek için önerilen Q-öğrenme algoritmasının sözde kodu Algoritma 1 ve Algoritma 2'de sunulmuştur. İlk olarak, "durum_boyutu x eylem_boyutu" boyutunda bir Q tablosu oluşturur. Oluşturulan Q tablosu başlangıçta sıfırlarla doldurulur çünkü ajanın deneyimi yoktur. Eğitim süreci, hiper parametrelerin başlatılmasıyla başlar. Daha sonra, belirli bir iterasyon boyunca içteki *while* döngüsü çalıştırılır. İçteki *while* döngüsünde ajan depodan başlayarak tüm müşterilerin taleplerinin karşılanmasıyla ve tekrar depoya dönülmesiyle bir rotanın elde edilmesi sağlanır. Her yeni durum ($\boldsymbol{s_t}$)'de ajan, keşif ve sömürüyü dengeleyen epsilon-açgözlü bir politika kullanarak bir eylem seçer. Eğitimin başlarında, eylemleri rastgele seçme olasılığı daha yüksektir (keşif). Zamanla, daha akıllı hale geldikçe, epsilon değeri azalır, bu da etkenin model tarafından tahmin edilen Q değerlerine göre en iyi eylemi seçerek şimdiye kadar edindiği bilgiyi giderek daha fazla kullandığı anlamına gelir. Ajan keşfetmek istediğinde (yani, rastgele sayı epsilonun altındaysa), maskeleme şeması tarafından belirlenen uygun eylemler kümesinden rastgele bir eylem seçer. Aksi takdirde, geçerli durum ve uygun eylemler için Q değerlerini hesaplar,

ardından en yüksek Q değerine sahip eylemi seçer. Seçilen eyleme (örneğin, bir müşteriye, şarj istasyonuna veya depoya gitme) göre EV'nin rotası, batarya seviyesini ve kapasitesi güncellenir. Eylem için ödül, Denklem 3.16-3.17'deki ödül fonksiyonu kullanılarak hesaplanır ve bu, daha uzun mesafeler için ajanı cezalandırırken verimli rota planlamasını teşvik eder. Ajan bir eylemi gerçekleştirdikten sonra, yeni duruma geçilir ve EV'nin durumunu (batarya, kapasite), konumunu ve geçerli düğüm bilgileri güncellenmiş olur.

ALGORİTMA 1: ÖNERİLEN Q-ÖĞRENME ALGORİTMASI

1: Q tablosunu oluştur ve köşegenleri büyük negatif sayılarla diğer kısımları sıfırlarla doldur.
2: Maksimum iterasyon sayısını belirle $max_iterations$
3: İndirim faktörü γ, öğrenme hızı α değerlerini belirle
4: Tüm epsilon değerlerini ϵ_0, ϵ_{min} ve Δ belirle
5: Başla $i = 0$.
6: **while** $i \leq max_iterations$
7: Ziyaret edilen müşteriler listesi oluştur
8: *rota_listesi'n*i boş liste olarak oluştur.
9: $toplam_mesafe = 0$.
10: İlk $durum$ (s_t)'i başlat.
11: EV'nin toplam yük ve batarya kapasitesini belirle
12: **while** True **do**
13: **if** tüm müşteriler ziyaret edildiyse ve mevcut durum depo ise
14: *tamamlanan_rota = True*.
15: **end if**
16: Maskeleme şemasına göre *geçerli_eylemler* listesini oluştur
17: **if** *geçerli_eylemler* listesi boşsa **then**
18: *tamamlanan_rota = False*
19: **end if**
20: Epsilon açgözlü politikasını kullanarak *geçerli_eylemler* listesinden bir *eylem* seç.
21: Seçilen *eylem* için adım fonksiyonunu çalıştır ve ödül r_t ile gelecek durumu s_{t+1} elde et
22: Bellman denklemini kullanarak Q değerini güncelle.
23: $TD = reward + \gamma \times Q[next_{state}, action] - Q[state, action]$
24: $Q[state, action] = Q[state, action] + \alpha \times TD$
25: $s_t = s_{t+1}$
26: Epsilon değerini güncelle $\epsilon = max(\min\ _epsilon, \epsilon \times \Delta)$
27: **end while**

28: **if** *tamamlanan_rota = True* **then**
29: *rota_listesi'n*e rotayı ve toplam uzaklığı ekle.
30: **end if**
31: artır i.
32: **end while**

$$R = \sum_{t=1}^{T} r^t \tag{3.16}$$

$$r_t = -(d_{ij}) \qquad eğer\ i = V_{t-1}\ ve\ j = V_t \tag{3.17}$$

ALGORİTMA 2: Adım Fonksiyonu

input: *durum, eylem, mevcut_yük_kapasitesi, mevcut_batarya_kapasitesi*
output: *ödül, durum, mevcut_yük_kapasitesi, mevcut_batarya_kapasitesi*
1: Şarj tüketim oranı h=1.
2: $şarj_tüketimi \leftarrow d_{ij} \times h$
3: **if** şarj tüketimine ve müşteri talebine göre ilgili eyleme gidilebilirse
4: durum=müşteri
5: $r_t \leftarrow -(d_{ij})$
6: *mevcut_yük_kapasitesi, mevcut_batarya_kapasitesi*'ni güncelle.
7: **end if**
8: **else if** şarj tüketimine bağlı olarak herhangi bir müşteriye gidilemezse **then**
9: rastgele şarj istasyonunu seçin
10: *durum* = şarj istasyonu
11: $r_t \leftarrow -(d_{ij})$
12: *mevcut_batarya_kapasitesi* 'ni güncelle.
13: **end if**
14: **else if** Müşteri talebine bağlı olarak herhangi bir müşteriye gidilemezse **then**
15: *durum* = depo
16: $r_t \leftarrow -(d_{ij})$
17: *mevcut_yük_kapasitesi, mevcut_batarya_kapasitesi* 'ni güncelle.
18: **end if**
19: Return *ödül, durum, mevcut_yük_kapasitesi, mevcut_batarya_kapasitesi*

Algoritma, ajan tüm müşterileri başarıyla ziyaret edip bir depoya döndüğünde sona erer. EV'nin bataryası biterse veya rotayı tamamlayamazsa, bölüm sona erer ve ajan

başarısızlıktan ders çıkarır. Başarıyla tamamlanan rotalar kaydedilir ve ajan müşteri talebini karşılayarak kat edilen toplam mesafeyi en aza indirmeye çalışır. Algoritma önceden tanımlanmış maksimum iterasyon sayısıyla başlar. Her 100. iterasyonda, tamamlanmış rotalar listesinden en düşük maliyetli rota belirlenir. Daha sonra, algoritma en son belirlenen en düşük maliyetli rotanın daha önce belirlenen en düşük maliyetli rota ile aynı olup olmadığını değerlendirir. En düşük maliyetli rota ardışık bir yineleme serisi boyunca (özellikle 2000 yineleme) değişmeden kalırsa, algoritmanın yakınsamayı başardığı kabul edilir ve böylece yinelemeli süreç sonlandırılır.

3.5. Derin Pekiştirmeli Öğrenme ile Zaman Pencereli Kapasiteli Elektrikli Araç Rotalama Problemi

Bu çalışmada, CEVRPTW problemi ayrık, yüksek boyutlu bir durum alanını içermektedir. DQN, ayrık eylemlere sahip problemleri çözmek için kullanılan modelden bağımsız ve değer tabanlı bir algoritmadır. CEVRPTW problemi için karar alma çerçevesini modellemede MDP ilkelerini uygulayan DQN yöntemi kullanılmıştır. MDP, her bir bileşenin CEVRPTW ve DQN yaklaşımının farklı özelliklerini hesaba katmak üzere dikkatlice tasarlandığı belirli bir modelle tanımlanır. MDP'nin unsurları durum, eylem, geçiş kuralı ve ödül fonksiyonu olarak tanımlanır. Bu problem için, durumlar düğümlerdir (depo, müşteriler, şarj istasyonları) ve eylemler bir düğümden diğerine hareket etmektedir. RL, bir ajanın hedefine ulaşmak için etkileşimde bulunduğu ortamın temel özelliklerini yakalamayı amaçlamaktadır. Ajan, ortamın durumunu algılayabilmeli ve durumu etkileyen eylemlerde bulunabilmelidir.

1) *CEVRPTW'de durum:* Bir RL ajanı bir periyot boyunca bir ortamla etkileşime girer. Her zaman adımında t, ajan bir durum uzayı S 'den bir durum s_t alır ve bir eylem uzayı A 'dan bir eylem a_t seçer. Önerilen yaklaşımda durum temsili önemli kısımlardan biridir. Durum, aşağıdakileri içeren sabit boyutlu bir vektör olarak kodlanır:
   - Mevcut düğüm tipi (müşteri, şarj istasyonu veya depo)
   - EV'nin mevcut batarya seviyesi
   - EV'nin mevcut yük kapasitesi
   - Mevcut zaman

- Geçerli eylemler hakkındaki bilgiler

Bu durum kodlaması, ajanın mevcut konumun, kaynaklarına ve kullanılabilir seçeneklerine göre kararlar almasını sağlar. Durum kodlamasındaki bilgiler normalize edilerek kullanılır. Bu sayede, farklı problem örnekleri arasında tutarlılığın korunması sağlanır.

2) *CEVRPTW'de eylem:* Eylem a_t , EV'nin t zaman adımında gideceği düğümü belirler. Burada, a_0 depodan ayrılma eylemi, a_1 ilk adımda seçilen eylem ve a_T ise depoya dönme eylemidir. Eylem uzayı A, ayrık eylemler kümesini temsil eder $a\ (a \in A \coloneqq C, F, D)$. Ajan bir müşteriye, bir şarj istasyonuna veya bir depoya gitmeye karar verir. Kalan batarya, kapasite, zaman penceresi ve tüm müşterilerin ziyaret edilip edilmediği gibi faktörleri göz önünde bulundurarak mevcut durum s_t 'ye göre gidilebilecek olası eylemleri belirlemek için maskeleme yöntemi uygulanır. Bu maskeleme adımı, her zaman adım t 'de uygulanamaz eylemleri ortadan kaldırarak sorunun karmaşıklığını azaltmaya yardımcı olur.

3) *CEVRPTW'de Ödül:* Ödül fonksiyonu tasarlanırken müşteri ziyaretlerine yalnızca mesafeye dayalı bir ceza verilirken, şarj istasyonlarına veya depolara yapılan ziyaretlere ek cezalar verildi. Bu tasarım, ajanı müşteri ziyaretlerine öncelik vermeye ve gereksiz duraklamaları en aza indirmeye teşvik eder. Eğitim sinyali yalnızca mesafeye göre hesaplandığında, EV müşterilere hizmet etmek yerine gereksiz yere şarj istasyonlarına veya depoya gidebilir. Bu eğilim, pratik olmayan rotalar oluşturur ve böylece modelin gerçek dünya senaryolarına uygulanabilirliğini zayıflatır. Bu sorunu ele almak için, ödül fonksiyonuna şarj istasyonlarını ve depoları ziyaret etme durumunda ek cezalar eklenir ve ajanı gereksiz ziyaretler yerine müşteri hizmetlerine odaklanmaya yönlendirir. Ödül sinyallerinin durum kodlamasıyla uyumlu olarak dikkatli bir şekilde tasarlanması, ağ yakınsamasını hızlandırmada etkilidir. Özellikle, daha düşük şarj sıklığının teşvik edildiği DQN çerçevesine ödül R'yi dahil ederek, ajan her iki seçenek de mevcut olduğunda şarj işlemlerinden ziyade müşteri hizmetlerine öncelik vermeyi öğrenir. Toplam ödül R, Denklem 3.18-3.19'da belirtildiği gibi her t adımında elde edilen anlık ödüllerin r_t toplamıdır. İki düğüm arasındaki d_{ij} mesafesi 0 ile 1 arasında normalize edilir.

$$R = \sum_{t=1}^{T} r^t \tag{3.18}$$

$$r_1 = \begin{cases} normalize(-d_{ij}) & eğer\ V = C \\ normalize(-d_{ij}) - 1 & eğer\ V = F \\ normalize(-d_{ij}) - 0.5 & eğer\ V = D \end{cases} \tag{3.19}$$

4) *CEVRPTW'de geçiş fonksiyonu:* Sistem durumu, şu anda yürütülen r_t eylemine göre s_t 'den s_{t+1} 'e güncellenecektir. Düğüm için, tüm düğümlerin statik bilgileri değişmeden kalır. Müşteri düğüm ziyaret edildiğinde, müşteri talebi sıfırlanır ve EV yük ve batarya kapasitesi güncellenir. Her t adımında, müşterilerin talepleri ve EV'nin mevcut kapasitesi Denklem 3.20-3.21'de olduğu gibi güncellenir.

$$q_i^t = \begin{cases} 0 & eğer\ i \in C \\ d_i^{t-1} & değilse \end{cases} \tag{3.20}$$

$$kapasite_i^t = \begin{cases} kapasite_i^{t-1} - d_i^{t-1} & eğer\ i \in C \\ kapasite_i^{t-1} & eğer\ i \in F \\ kapasite_i^{t_0} & eğer\ i \in D \end{cases} \tag{3.21}$$

EV'nin SoC'si Denklem 3.22'deki şekilde güncellenir.

$$soc_j^t = soc_i^{t-1} - e_{ij} \quad eğer\ i = V^{t-1} ve\ j = V^t \tag{3.22}$$

Sistem zamanı Denklem 3.23'e göre güncellenir.

$$T^t = \begin{cases} T^{t-1} + t_{ij} + w_j + s_j & eğer\ i = V^{t-1}\ ve\ j = V^t, V = C \\ T^{t-1} + t_{ij} + c_j & eğer\ i = V^{t-1}\ ve\ j = V^t, V = F \\ 0 & eğer\ V = D \end{cases} \tag{3.23}$$

$$c_j = (Q_c - Q_j).g \tag{3.24}$$

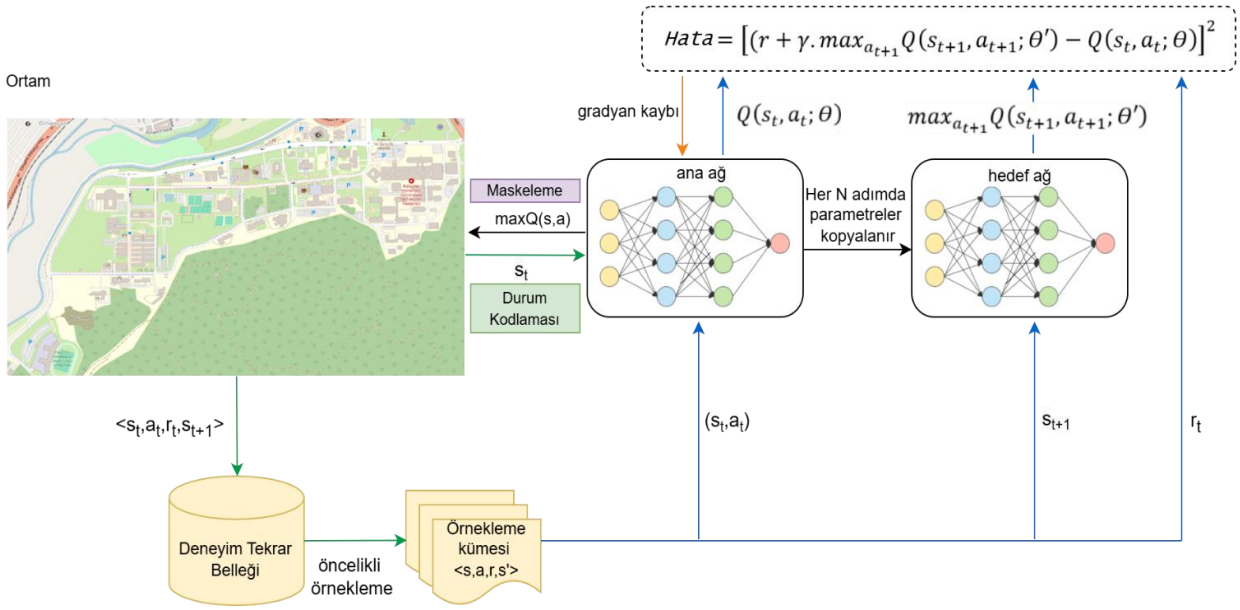
Denklem 3.23'te t_{ij} iki düğüm arasındaki seyahat süresini, s_j her müşteri için gereken hizmet süresini, w_j servise başlamak için beklenen süreyi ifade eder. Eğer EV müşteri j 'ye servise başlama süresinden erken ulaşırsa servise başlama zamanına kadar

bekler. Şarj süresi c_j, Denklem 3.24 kullanılarak belirlenir. Burada Q_c EV'nin toplam batarya kapasitesini, Q_j EV'nin mevcut batarya seviyesini ve g şarj oranını belirtir.

5) CEVRPTW'de politika: Politika $\pi(s_t)$, ajanın mevcut durum s_t 'deki eylemini belirlemek için kullandığı bir olasılık dağılımını ifade eder. ε-açgözlü keşif stratejisi (ε-greedy) kullanılarak türetilebilir (Denklem 3.14).

Önerilen DQN algoritması, belirli bir durumdaki her olası eylem için beklenen gelecekteki ödülü tahmin eden Q-değeri fonksiyonunu yaklaşık olarak hesaplamak için bir sinir ağı kullanır. Önerilen mimari, Şekil 3.14'te gösterilmiştir. DQN klasik Q-öğrenmedeki Bellman Denklemi'nden türetilen zamansal farkı kayıp fonksiyonu olarak kullanır ve Q değerleri sinir ağı parametreleri üzerinden geriye yayılım yapılarak güncellenir. Her öğrenme adımında deneyim tekrar belleğinden seçilen örnekler için hedef değer Denklem 3.25 ile hesaplanır. Ardından ağın güncel çıktısı $Q(s_t, a_t)$ ile bu hedef arasındaki karesel hata hesaplanır ve ağ ağırlıkları θ geri yayılımla güncellenir.

$$y_t = \begin{cases} r_t & eğer\ durum\ s_{t+1}\ son\ durum\ ise \\ r_t + \gamma . max_a Q(s_{t+1}, a; \theta) & eğer\ s_{t+1}\ son\ durum\ değilse \end{cases} \quad (3.25)$$

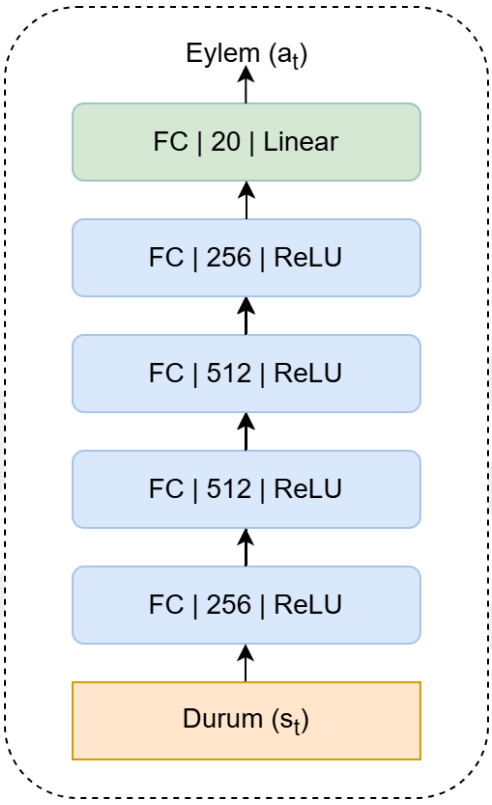


Şekil 3.14. Önerilen DQN algoritması

Model eğitildikten sonra, kümülatif ödülleri maksimize etmeyi amaçlayan eylemler için Q değerlerini yaklaşık olarak hesaplayabilir. Ancak, $Q(s_t, a_t)$ hesaplanırken mimaride önemli bir zorluk ortaya çıkar. Bellman denklemine göre, $Q(s_t, a_t)$ değeri $Q(s_{t+1}, a_{t+1})$'ye bağlıdır, burada s_{t+1}, s_t 'yi izleyen bir sonraki durumdur. DQN de bu adımlar sinir ağı ile hesaplandığından hem $Q(s_t, a_t)$ değeri hem de hedef $Q(s_{t+1}, a_{t+1})$ değeri aynı ağdan elde edilmiş olur. Eğitim sırasında ağ parametreleri güncellenince hedefi oluşturan çıktılar da aynı anda değişir. Bu da eğitim sırasında tahminler sürekli değiştiğinden hareket eden bir hedefi kovalama sorununa yol açar ve model kararsız bir öğrenme döngüsüne girer. Bu dengesizliği azaltmak amacıyla, $Q(s_{t+1}, a_{t+1})$'yi tahmin etmek için Hedef Ağ olarak adlandırılan ikinci bir sinir ağı kullanılır. Ana ağın parametreleri her adımda güncellenirken hedef ağın parametreleri sabit tutulur ve yalnızca belirli aralıklarla (örneğin her N adımda) güncellenir. Bu durum öğrenmeyi stabilize eder ve ajanın gelecekteki ödülleri aşırı tahmin etmesini önlemeye yardımcı olur. Eğitim aşaması, batarya kapasitesi, araç yükü, zaman penceresi ve şarj istasyonlarının konumu gibi kısıtlamaları göz önünde bulundurarak aracının optimal rotaları öğrenmesine rehberlik etmek için tasarlanmıştır.

DQN'de kullanılan ağ mimarisi 1 girdi katmanı, 1 çıktı katmanı ve 4 gizli katmandan oluşmaktadır. Ağın girdisi, mevcut düğüm özellikleri, EV'nin durumu ve mevcut düğümden seçilebilecek uygun eylemlerin özelliklerinin bir kombinasyonudur. Gizli katman, her biri bir ReLU aktivasyon fonksiyonu tarafından takip edilen sırasıyla 256, 512, 512 ve 256 birimli dört tam bağlı katmandan oluşur. Bu derin yapı, modelin rotalama problemindeki karmaşık ilişkileri yakalamasına olanak tanır. Çıktı, her olası eyleme (yani, rotadaki bir sonraki düğümü seçme) karşılık gelen bir Q değerleri vektörüdür. Q değerleri, ajanın optimum eylemi seçmesine rehberlik eder. Model, ortalama karesel hata kullanılarak eğitilir ve 0,0001 öğrenme oranına sahip Adam optimize edicisi kullanılarak optimize edilir. Seçilen ağ yapısının temel nedeni CEVRP'deki durum eylem uzayının doğasında yatan karmaşıklıktır. Problemdeki durum girdisi birden çok özelliği içinde barındırır ve bu yüksek boyutlu yapıyı modelleyebilmek için 256, 512, 512, 256 şeklindeki mimari hem yeterli temsil gücü hem de aşırı öğrenme riskini dengeleyen bir çözüm sunmaktadır. Daha az katmanlı basit bir ağ durum eylem etkileşimlerinin karmaşıklığını yeterince modelleyemez. Literatürde oyun ve görüntü tabanlı DQN çalışmalarında üç konvolüsyon katmanı ile 512 birimli tam bağlı katman kullanılmıştır (Mnih vd., 2015). Oyunlardaki konvolüsyon katmanları, bitişik pikseller arasındaki desen ve hareket bilgisini yakalamak için

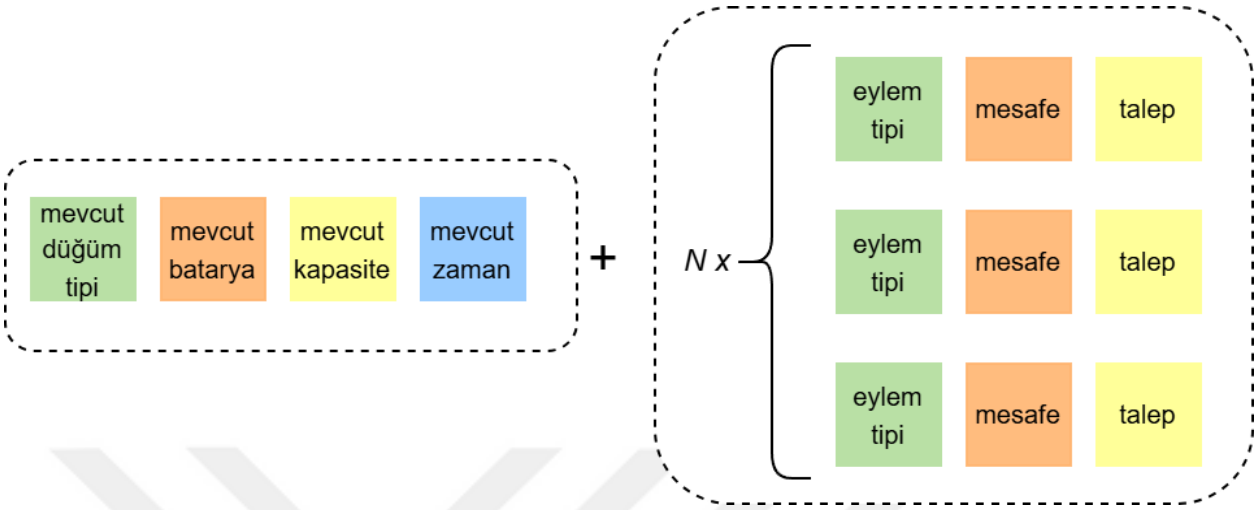
tasarlanmıştır ancak CEVRP'de girdi sabit uzunluklu sayısal özelliklerden oluşan bir vektördür ve burada konvolüsyon kullanılması uygun değildir. Tam bağlı katmanlar ise tüm öznitelikleri eşit ağırlıkla işleyerek bu tür veri yapısında çok daha etkili genelleme sağlar. Önerilen modeldeki ana ağın ve hedef ağın yapısı Şekil 3.15'te gösterilmiştir.



Şekil 3.15. Ana ve hedef ağın yapısı

DQN tabanlı yaklaşımlarda, hangi bilgilerin duruma dahil edildiğini ve nasıl temsil edildiğini belirlemek çok önemlidir. Durum temsil edilirken, gereksiz özelliklerden kaçınırken tüm temel çevresel ayrıntıları içermelidir. Genellikle, DQN mimarileri sabit boyutlu bir giriş katmanı gerektirir ve bu da durumu sabit uzunlukta bir vektör olarak temsil etmeyi gerekli kılar. Ancak, CEVRPTW gibi problemlerde, müşteri sayısı sabit değildir. Bu zorluğun üstesinden gelmek için, bu tez çalışmasında, değişen sayıda müşteriye sahip örnekleri esnek bir şekilde barındıran yeni bir durum kodlama yöntemi önerilmektedir. *durum_kodlaması()* yöntemi, EV'nin durumunu ve olası eylemleri bir durum vektörüne kodlarken, karşılık gelen eylem haritası yalnızca geçerli eylemleri içerir. Bu, EV'nin yalnızca bir şarj istasyonuna ulaşmak için yeterli bataryaya sahip olmak veya tüm müşterilere hizmet verdikten sonra bir depoya geri dönebilmek gibi tüm kısıtlamaları

karşılayan eylemleri dikkate almasını sağlar. *durum_kodlaması()* yöntemi ile oluşturulan durum gösterimi Şekil 3.16'da gösterilmiştir.



Şekil 3.16. Durum gösterimi

Durum gösterimi, mevcut düğüm, EV ve uygun eylemlerle ilgili ayrıntılı bilgileri kapsar. Özellikle, mevcut düğümün türünü, EV'nin kalan batarya seviyesini ve kapasitesini ve ayrıca mevcut zamanı içerir. Ek olarak, mevcut düğümden maskeleme şeması ile elde edilen uygun eylemlerle ilgili veriler dahil edilir. Bu uygun eylemler, mevcut düğümden ulaşılabilen müşteriler, şarj istasyonları ve depo gibi tüm erişilebilir düğümleri kapsar. Uygun eylemler kümesinden, N adet eylem seçilir. Bu N eylemin her biri için gösterim, düğümün türünü, mevcut düğümden seçilen düğüme olan mesafeyi ve seçilen düğümle ilişkili talebi yakalar. Bu çalışmada, $N = 20$ olarak seçildi ve (20×3+4)=64'lük bir durum boyutu elde edildi. Uygun eylemler arasından 20 düğüm seçmek, sabit boyutlu girdiler gerektiren DQN gibi algoritmalar için oldukça faydalı olan sabit bir durum boyutu sağlar. Ayrıca durum gösteriminde, batarya seviyesi, yük miktarı, sistem zamanı ve mesafe gibi değişkenler normalize edilir, bu da daha hızlı ve daha kararlı öğrenmeyi destekler. Uygulanabilir eylem sayısının 20'den az olduğu durumlarda, 64 boyutlu yapıyı koruyarak kalan düğümler için -1 değeri ile dolgu uygulanır. Normalize edilen değerler [0 1] aralığında olduğu için -1 ile dolgu yapmak diğer değerler ile çakışma olmamasını sağlar. Bu gösterim genellikle verimli DQN eğitimi için yeterince bilgilendiricidir.

CEVRPTW'yi çözmek için önerilen DQN algoritmasının sözde kodu Algoritma 3'te sunulmuştur. Eğitim süreci, sinir ağı ağırlıklarının başlatılmasıyla başlar. Her eğitim yinelemesinde, yeni bir problem örneği B başlatılır ve B'den seçilen b örneği için bir

başlangıç durumu oluşturulur. Ödül başlangıçta sıfıra ayarlanır ve ajanın etkileşimlerini kaydetmek için bir deneyim tekrar belleği (experience replay memory) oluşturulur. Daha sonra, ajan tüm müşterilere hizmet verilen turun tamamlanmasıyla belirlenen bir son duruma ulaşana kadar bir while döngüsü yürütülür.

ALGORİTMA 3: ÖNERİLEN DQN ALGORİTMASI

1: Ana ağın ağırlıklarını Q ile rasgele başlat
2: Hedef ağın ağırlıklarını $\theta = Q$ olarak ayarla
3: Önceliklendirilmiş kapasitesi D olan deneyim tekrar belleği oluştur
4: Kullanılacak tüm hiper parametreleri belirle (epsilon değerleri ϵ_0, ϵ_{min} ve Δ, indirim faktörü γ, öğrenme hızı α, iterasyon sayısı N, hedef ağı güncelleme aralığı)
5: **foreach** iterasyon $= 1,2,\cdots,N$ **do**
6: B adet problem örneği kullan
7: **foreach** b $= 1,2,\cdots,B$ **do**
8: Başlangıç durumunu depo olarak belirle s_0^b
9: **while** tur bitene kadar tekrarla **do**
10: geçerli_eylemler=*maskeleme ( )*
11: s_0^b =*durum_kodlaması* (s_0^b,*geçerli_eylemler)*
12: Epsilon aç gözlü seçim stratejisi ile bir eylem a_t^b seç
13: Seçilen eyleme göre ödül değerini r_t^b ve bir sonraki durumu s_{t+1}^b hesapla
14: s_{t+1}^b= *durum_kodlaması* (s_{t+1}^b)
15: Deneyimleri $< s_t^b, a_t^b, r_t^b, s_{t+1}^b >$ deneyim tekrar belleğine D kaydet
16: **if** D'de yeterli deneyim varsa **then**
17: D'deki deneyimlerden örnekleme kümesi seç
18: $$y_i = \begin{cases} r_i & eğer\ durum\ s_{i+1}\ son\ durum\ ise \\ r_i + \gamma . max_a Q(s_{i+1}, a; \theta) & eğer\ s_{i+1}\ son\ durum\ değilse \end{cases}$$
19: Ağın ağırlıklarını gradyan inişi ile güncelle $\left(y_i - Q(s_i, a; \theta)\right)^2$
20: **end if**
21: **end while**
22: **end for**
23: **end for**

Her yeni durum s_t'de, ajan epsilon-açgözlü bir politika kullanarak (Denklem 19) keşif ve sömürüyü dengeleyen bir eylem seçer. Eğitimin başlarında, eylemleri rastgele seçme olasılığı daha yüksektir (keşif). Zamanla, eğitim gerçekleştikçe, epsilon değeri azalır, bu da ajanın model tarafından tahmin edilen Q değerlerine göre en iyi eylemi seçerek şimdiye kadar edindiği bilgiyi daha fazla kullandığı anlamına gelir. Ajan keşfetmek istediğinde (yani, rastgele sayı epsilonun altında olduğunda), Algoritma 4'te ayrıntılı gösterilen maskeleme fonksiyonu tarafından belirlenen uygun eylemler kümesinden rastgele bir eylem seçer. Aksi takdirde, geçerli durum ve uygun eylemler için Q değerlerini hesaplar, ardından en yüksek Q değerini elde eden eylemi seçer. Seçilen eylem, EV'nin rotasını, batarya seviyesini ve kapasitesini seçilen düğüme (örneğin, bir müşteriye, şarj istasyonuna veya depoya gitme) göre günceller. Eylemin ödülü, Denklem 3.18-3.19'daki ödül fonksiyonu kullanılarak hesaplanır ve bu fonksiyon, aracı daha uzun mesafeler için cezalandırırken verimli rota planlamasını teşvik eder. Aracı bir eylemi gerçekleştirdikten sonra, EV'nin durumu (batarya, kapasite), konumu ve mevcut düğüm bilgileri güncellenerek yeni bir duruma geçilir.

ALGORİTMA 4: MASKELEME FONKSİYONU

input: mevcut_düğüm, mevcut_yük_kapasitesi, mevcut_batarya_kapasitesi, mevcut_zaman, rota_listesi, ziyaret_edilen_müşteriler
output: geçerli_eylemler
foreach depo **in** tüm_depolar
    if mevcut_düğüm **!=** depo
        gitme_durumu, yeni_kapasite, yeni_batarya, mesafe = gitmeKontrol (mevcut_düğüm, depo, mevcut_kapasite, mevcut_batarya, mevcut_zaman)
        döngü_durumu=döngüKontrol (rota_listesi + depo)
        if gitme_durumu ==true ve döngü_durumu==false
            geçerli_eylemler.ekle(depo)
foreach müşteri **in** tüm_müşteriler
    if mevcut_düğüm **!=** müşteri ve müşteri **not in** ziyaret_edilen_müşteriler
        gitme_durumu, yeni_kapasite, yeni_batarya, mesafe = gitmeKontrol (mevcut_düğüm, müşteri, mevcut_kapasite, mevcut_batarya, mevcut_zaman)
        cs_durumu=csKontrol (düğüm, yeni_kapasite, yeni_batarya)
        depo_durumu=depoKontrol (düğüm, yeni_kapasite, yeni_batarya)
        if gitme_durumu ==true ve (cs_durumu ==true ya da depo_durumu==true

```
13:                 geçerli_eylemler.ekle(müşteri)5:
14:  foreach cs in tüm_csler
15:      if mevcut_düğüm != cs
16:          gitme_durumu, yeni_kapasite, yeni_batarya, mesafe = gitmeKontrol (mevcut_
düğüm, cs, mevcut_kapasite, mevcut_batarya, mevcut_zaman)
17:          döngü_durumu=döngüKontrol (rota_listesi + cs)
18:          if gitme_durumu ==true ve döngü_durumu==false
19:                 geçerli_eylemler.ekle (cs)
```

Yeni duruma geçildikten sonra bazı bilgiler (mevcut durum, eylem, ödül, sonraki durum), $< s_t, a_t, r_t, s_{t+1} >$ olarak gösterilir ve bu deneyimler önceliklendirilmiş deneyim tekrar belleğinde saklanır. Bunun sebebi bu deneyimlerin daha sonraki bir aşamada model eğitimi için kullanılacak olmasıdır. Bu süreç, DQN'nin geçmiş deneyimlerden genelleme yapmasını ve gelecekteki karar vermeyi optimize etmesini sağlar. Eğitim sırasında kararlılığı artırmak için, ajan periyodik olarak önceliklendirilmiş deneyim tekrar belleğinden örnekler alır. Bir deneyim grubu, tahmin edilen ve gerçek Q değerleri arasındaki kaybı en aza indirerek ağı güncellemek için kullanılır. Bu adımda, model gruptaki her durum-eylem çifti $Q(s_t, a_t)$ değerini hesaplarken, hedef ağ tüm olası sonraki eylemler için sonraki durum s_{t+1} için $Q(s_{t+1}, a_{t+1})$ yı değerlendirir. Sonraki eylem a_{t+1}, $Q(s_{t+1}, a_{t+1})$ 'yi maksimize edecek şekilde seçilir. Son olarak, model bu değerlendirmeye dayalı bir eğim adımı gerçekleştirir. Her 100 bölümde bir güncellenen hedef ağı, Q değeri güncellemeleri için istikrarlı bir referans sağlayarak daha iyi bir öğrenme sağlar ve gelecekteki ödüllerin aşırı tahmin edilmesini önler. Bölüm, ajan tüm müşterileri başarıyla ziyaret edip bir depoya döndüğünde sona erer. Aracın bataryası biterse veya rotayı tamamlayamazsa, bölüm sona erer ve aracı başarısızlıktan ders çıkarır. Başarıyla tamamlanan rotalar kaydedilir ve ajan, müşteri talebini karşılayarak kat edilen toplam mesafeyi en aza indirmeye çalışır.

DQN birden fazla bölüm boyunca eğitilir ve bu eğitim aşamasındaki kritik adımlardan biri maskeleme şemasıdır. Maskeleme, eylem kümesinden uygulanamaz, mantıksız veya istenmeyen eylemleri kaldırarak öğrenme sürecini hızlandırmakla kalmaz, aynı zamanda kararlılığını da artırır. Çok sayıda olası eylemin olduğu senaryolarda, geçersiz

olanları filtrelemek etkili eylem alanını daraltır ve böylece öğrenmeyi hızlandırır. Bu, DQN'nin uygulanamaz eylemleri değerlendirmek için hesaplama kaynaklarını harcamasını da önler. DQN her eylem için Q değerleri ürettiğinden, geçersiz eylemleri tutmak ve bunlara yapay olarak düşük veya yüksek cezalar atamak Q işlevi tahminlerinde istikrarsızlığa yol açabilir. Maskeleme, geçersiz eylemleri kaldırarak, öğrenme sürecinde büyük hatalar ve istikrarsız güncellemeler riskini azaltarak kararlılığı artırır.

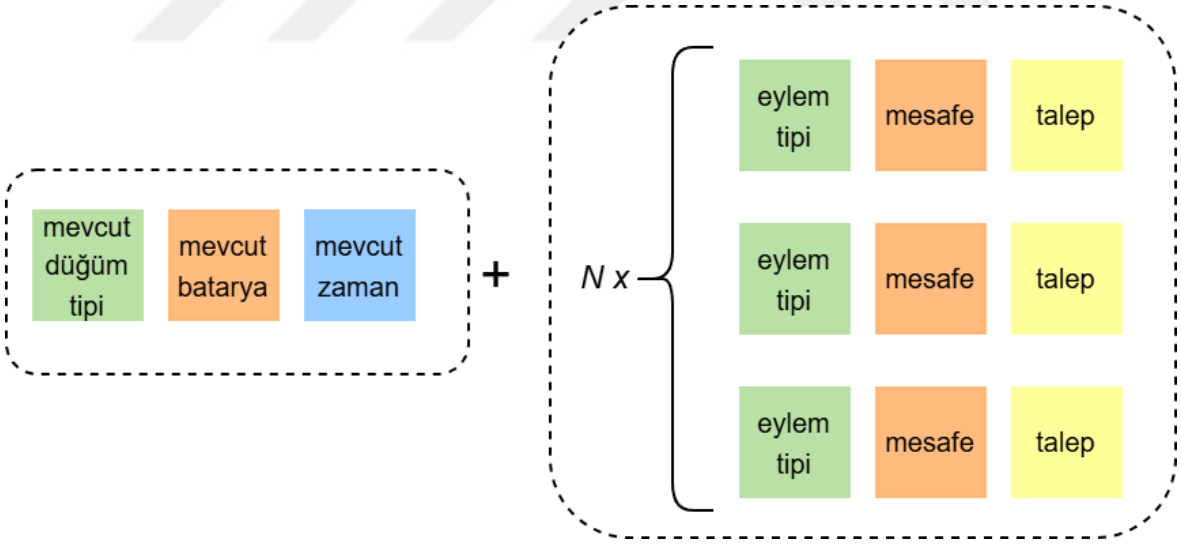
Maskeleme Şeması: Eğitim sürecini hızlandırmak ve çözümlerin uygulanabilirliğini sağlamak için, uygun olmayan rotaları ortadan kaldırmayı amaçlayan maskeleme stratejileri geliştirildi. Tüm eylemler arasından uygun olmayan eylemleri filtreleyerek, aracının geçerli eylemlere odaklanması artırılır ve böylece uygulanabilir çözümlerin oluşturulmasına yardımcı olunur. Maskeleme fonksiyonu ile EV'nin batarya ve yük seviyesi, zaman pencereleri ve ziyaret edilen müşteriler gibi faktörleri göz önünde bulundurarak geçerli duruma göre uygun eylemlerin bir listesi oluşturulur. Yetersiz batarya, kapasite veya zaman pencerelerinin uymaması nedeniyle ulaşılamayan bir düğüme gitmek gibi geçersiz eylemler maskelenir.

- Eğer müşteri *i* önceden ziyaret edilmişse müşteri *i* maskelenir.
- Eğer düğüm *i* müşteriyi temsil ediyorsa ve düğüm i nin talebi EV'nin mevcut yük kapasitesi tarafından karşılanamıyorsa düğüm *i* maskelenir.
- Eğer EV şu anda müşteri i'de ise ve batarya seviyesi bir sonraki müşteri j'yi ziyaret etmek ve daha sonra depoya veya herhangi bir şarj istasyonuna dönmek için yeterli değilse, müşteri j maskelenir.
- Eğer EV şu anda herhangi bir i düğümünde ise ve batarya seviyesi bir sonraki şarj istasyonu j'yi ziyaret etmek ve daha sonra depoya veya herhangi bir şarj istasyonuna geri dönmek için yeterli değilse, şarj istasyonu j maskelenir.
- Eğer EV'nin yeni teslimatları tamamlamadan önce gereksiz yere şarj istasyonlarını veya depoyu tekrar ziyaret edeceği bir kendi kendine döngü tespit edilirse, daha verimli bir rotalamayı teşvik etmek için ilgili eylem maskelenir.
- Eğer EV'nin düğüm *i* ye gelme zamanı, düğüm *i* nin hizmete en geç başlama zamanından büyükse düğüm *i* maskelenir. Burada düğüm *i* ye gelme zamanı hesaplanırken mevcut zaman değeriyle bir önceki düğüm ve düğüm *i* arası harcanan süre toplanır.

3.6. Derin Pekiştirmeli Öğrenme ile Zaman Pencereli Topla Dağıt Elektrikli Araç Rotalama Problemi

Bu bölümde, zaman pencereli topla dağıt elektrikli araç rotalama problemi (EVRP-PDP-TW) için Bölüm 3.5'te ayrıntılarıyla anlatılan DQN modeli uygulanmıştır. CEVRPTW için önerilen DQN yaklaşımı, yalnızca minimal uyarlamalar ile EVRP-PDP-TW problemine entegre edilmiştir. Toplama ve dağıtım işlerini kapsayan EVRP-PDP-TW probleminde EV için yük kapasitesi kısıtını bulundurmaz. Bu nedenle DQN modelindeki durum vektöründen yük kapasitesi bileşeni çıkarılmıştır. EVRP-PDP-TW problemi için yeni durum gösterimi Şekil 3.17'de gösterilmektedir.

Buna bağlı olarak maskeleme mekanizmasından da yük kapasitesi ile ilgili olan kontroller çıkarılarak, batarya kapasitesi ve zaman penceresi kriterlerini göz önünde bulundurulmaktadır. Modeldeki diğer tüm bileşenler (geçerli eylem seçimi, ödül fonksiyonu ile ilgili tanımlamalar, ana ve hedef ağın yapısı vb.) CEVRPTW modelinin birebir aynısıdır.



Şekil 3.17. EVRP-PDP-TW problemi için durum gösterimi

Bu uyarlama, önerilen DQN yaklaşımının farklı rotalama problemlerine yalnızca duruma ilişkin kodlama ve maskeleme kurallarında küçük değişiklikler yapılarak hızlıca adapte edilebildiğini göstermektedir. Kapasite kısıtı yerine farklı lojistik kısıtlar eklendiğinde veya zaman penceresi mantığı farklılaştırıldığında, önerilen mimarisi aynı

kalırken yalnızca durum gösterimi maskeleme fonksiyonda ilgili yerler güncellenir. Bu sayede önerilen yöntem, EVRP-PDP-TW'nin yanı sıra geniş bir problem sınıfında yüksek yeniden kullanılabilirlik ve ölçeklenebilirlik sunmaktadır.

4. BULGULAR VE TARTIŞMA

Bu bölümde, tez çalışması kapsamında ele alınan her iki veri seti üzerinde hem geliştirilen matematiksel modelin hem de önerilen pekiştirmeli öğrenme yaklaşımlarının performansı çoklu test senaryoları altında değerlendirildi. Öncelikle, matematiksel model CPLEX ile çözülerek elde edilen optimal sonuçlar toplam mesafe ve yürütme süresi bakımından raporlandı. Ardından, aynı veri setleri üzerinde uygulanan Q-Öğrenme ve DQN derin algoritmasıyla ulaşılan çözümler karşılaştırılarak modelin genellenebilirliği, hesaplama verimliliği ve çözüm kalitesi incelenmiştir.

4.1. Matematiksel Model Sonuçları

Bölüm 3.2.2' de sunulan matematiksel model, ESOGU_CEVRPTW veri setinde optimal rotaları elde etmek için kullanılır. Oluşturulan veri seti için CPLEX çözücüsü ile optimal sonuçlar elde edilir. Çizelge 4.1, veri setindeki bazı örnekler için zaman penceresi kısıtlamasını dahil etmeden CEVRP için matematiksel model sonuçlarını gösterir. Yürütme süresi, minimum mesafe, rota sayısı ve çözülen örnekler için optimal rota Çizelge 4.1'de gösterilmiştir. Matematiksel model, 5, 10 müşteriyi içeren küçük boyutlu problemler için makul bir sürede sonuç üretebilir. 20, 40 ve 60 müşteriyi içeren büyük boyutlu problemlerin çözümü çok uzun sürdüğü için model yaklaşık 3 saat çalıştırılmış ve bu süre sonunda elde edilen sonuçlar verilmiştir. Büyük boyutlu problemler için optimal sonuçlar makul bir sürede elde edilememiştir.

Çizelge 4.1. CEVRP'nin tam şarj stratejisine göre CPLEX'den elde edilen optimal sonuçları

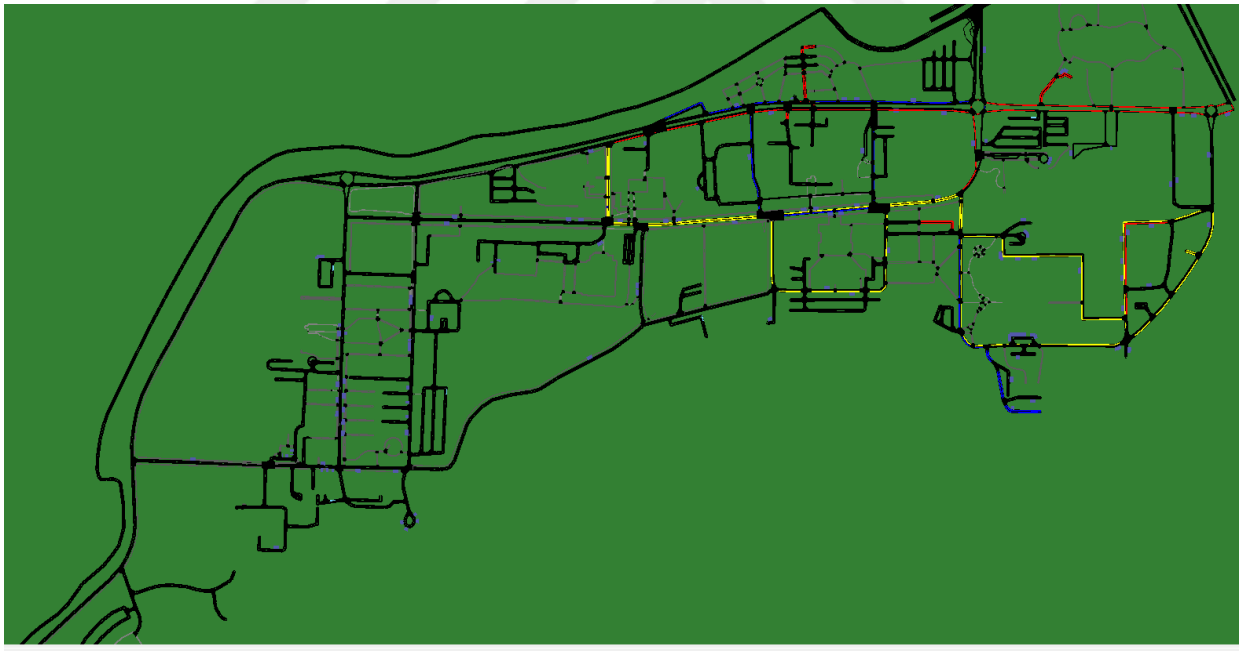
Örnek Adı	**Mesafe (m)**	**Yürütme Süresi (s)**	**Rotalar**
ESOGU_C5	3956.00	0.50	R1: Depot-35-cs7-115-2-32-Depot R2: Depot-42B-Depot
ESOGU_R5	4977.00	0.53	R1: Depot- 31-32-19-cs10-75-60A/2-Depot
ESOGU_RC5	4944.00	0.41	R1: Depot- 31-42B-115-32-Depot R2: Depot- 75-Depot

ESOGU_C10	5646.00	124.33	R1: Depot- 31-119-34-24-22A-Depot R2: Depot- 45-13-14-cs8-113-26-Depot
ESOGU_R10	4889.00	389.74	R1: Depot- 56-122-70-74-77-78-60A/2-Depot R2: Depot- 22E-21-22A-Depot
ESOGU_RC10	5843.00	345.63	R1: Depot- 13-14-22E-21-24-22A-Depot R2: Depot- 56-122-74-45-Depot
ESOGU_C20	*9125.00	11160.69	R1: Depot- 30-115-26-116-27-37B-Depot R2: Depot- 44-58/1-58/1-43C-42B-31-39-Depot R3: Depot- 49-2-8-cs8-9-50-51-Depot R4: Depot- 153-Depot
ESOGU_R20	*9974.00	11134.66	R1: Depot- 35-119-116-115-cs8-10-45-Depot R2: Depot- 152-22D-7-6-Depot R3: Depot- 154-44-55-43C-Depot R4: Depot- 60A/2-62-74-82-64-61B-Depot
ESOGU_RC20	*9161.00	11113.25	R1: Depot- 35-115-2-cs8-10-22D-Depot R2: Depot- 44-43C-42B-39-Depot R3: Depot- 152-153-51-50-45-Depot R4: Depot- 58/1-58/1-61B-64-82-62-Depot
ESOGU_C40	*16102.00	10800.05	R1: Depot- 25-1-5-2-8-15-Depot R2: Depot- 38-35-116-27-28-29-30-39-Depot R3: Depot- 44-60A/2-58/1-58/1-57-55-Depot R4: Depot- 49-120-18-17-22B-22C-Depot R5: Depot- 52-51-50-3-37A-Depot R6: Depot- 117-16-9-cs8-113-115-Depot R7: Depot- 152-42C-42B-41-Depot
ESOGU_R40	*18276.00	10800.02	R1: Depot- 10-11-12-13-23-20-Depot R2: Depot- 38-41-Depot R3: Depot- 46-24-22A-27-28-Depot R4: Depot- 74-76-124-88-cs3-73-60B/2-Depot R5: Depot- 119-1-113-125-8-Depot R6: Depot- 120-19-17-22C-21-Depot R7: Depot- 42A-42C-154-53-153-Depot

			R8: Depot- 43B-122-61B-64-57-44-Depot
ESOGU_RC40	*17053.00	10800.02	R1: Depot- 2-8-10-11-13-15-20-Depot R2: Depot- 19-17-cs8-125-113-115-1-Depot R3: Depot- 38-35-119-3-25-28-39-Depot R4: Depot- 49-50-120-21-24-22A-Depot R5: Depot- 42B-57-60B/2-55-Depot R6: Depot- 42C-152-153-Depot R7: Depot- 58/1-58/1-61B-64-76-124-51-Depot
ESOGU_C60	*26773.00	10800.02	R1: Depot- 3-25-115-1-36-37B-Depot R2: Depot- 10-23-16-21-Depot R3: Depot- 34-2-4-5-113-28-39-Depot R4: Depot- 38-30-18-19-118-Depot R5: Depot- 47-26-116-35-119-32-Depot R6: Depot- 49-14-9-12-13-22C-Depot R7: Depot- 56-61A/1-61B-65-70-71-Depot R8: Depot- 59-60A/2-60B/2-43C-31-40-Depot R9: Depot- 62-68-69-66-63-Depot R10: Depot- 154-53-52-42C-Depot R11: Depot- 42A-7-24-22D-22A-Depot
ESOGU_R60	*26159.00	10800.03	R1: Depot- 2-4-114-5-113-Depot R2: Depot- 16-8-cs8-12-13-23-20-Depot R3: Depot- 25-116-27-28-29-30-Depot R4: Depot- 38-40-41-50-Depot R5: Depot- 59-62-67-68-73-72-63-Depot R6: Depot- 74-65-122-56-57-60A/2-60B/2-Depot R7: Depot- 75-79-77-88-cs1-78-81-Depot R8: Depot- 123-82-84-52-Depot R9: Depot- 154-53-153-42A-Depot R10: Depot- 37A-45-19-22D-33-36-37B-Depot R11: Depot- 43A-44-43C-42B-Depot
ESOGU_RC60	*27905.00	10800.02	R1: Depot- 13-14-10-16-23-Depot

	R2: Depot- 25-26-116-27-28-Depot
	R3: Depot- 29-30-31-38-119-Depot
	R4: Depot- 34-114-5-113-8-4-Depot
	R5: Depot- 45-24-22D-22A-33-36-37B-Depot
	R6: Depot- 50-19-18-22C-21-Depot
	R7: Depot- 52-65-56-57-122-61A/1-61B-Depot
	R8: Depot- 77-79-88-cs3-60A/2-154-53-153-Depot
	R9: Depot- 43A-62-68-73-63-59-Depot
	R10: Depot- 43C-42B-40-Depot
	R11: Depot- 60B/2-78-123-81-Depot

* En iyi çözüm değildir. Büyük ölçekli problemlerde, matematiksel model makul bir sürede çözümler sağlayamamıştır. CPLEX tarafından üç saatlik bir süre sınırlaması ile elde edilen sonuçlar. 5 ve 10 müşterili problemler için elde edilen çözümler optimal çözümlerdir.



Şekil 4.1. ESOGU_C10 probleminin matematiksel model sonucunun SUMO'da gösterimi

Şekil 4.1, veri kümesindeki ESOGU_C10 örneği için matematiksel model tarafından elde edilen optimal rotayı göstermektedir. Elde edilen optimal rotalar SUMO'da görselleştirilmiştir. ESOGU_C10 için rotalar şu şekildedir: R1: D-45-22A-CS8-14-D R2: D-

CS4-26-113-CS6-24-13- D R3: D-119-34-31- D. Şekil 13'te sarı renkle gösterilen rota R1'i, kırmızı renkle gösterilen rota R2'yi ve mavi renkle gösterilen rota R3'ü temsil etmektedir.

Çizelge 4.2 ise, veri setindeki bazı örnekler için zaman penceresi kısıtlamasını dahil edilerek CEVRPTW için matematiksel model sonuçlarını gösterir. Yürütme süresi, minimum mesafe, rota sayısı ve çözülen örnekler için optimal rota Çizelge 6.2'de gösterilmiştir. CEVRPTW için matematiksel model, 5, 10, 20, 40 müşteri içeren problemler için makul kabul edilebilecek bir sürede sonuç üretebilir. Ancak 60 müşteriyi içeren büyük boyutlu problemlerin çözümü çok uzun sürdüğü için model yaklaşık 3 saat çalıştırılmış ve bu süre sonunda elde edilen sonuçlar verilmiştir. Büyük boyutlu problemler için optimal sonuçlar makul bir sürede elde edilememiştir.

Çizelge 4.2. CEVRPTW'nin tam şarj stratejisine göre CPLEX'den elde edilen optimal sonuçları

Örnek Adı	**Mesafe(m)**	**Yürütme Süresi (s)**	**Rotalar**
ESOGU_C5_TW1	5006.00	0.05	**R1:**D-2-115-D **R2:**D-42B-32-35-D
ESOGU_R5_TW1	5727.0	0.12	**R1:**D-19-32-D **R2:**D-31-D **R3:**D-75-60A/2-D
ESOGU_RC5_TW1	7094.00	0.05	**R1:**D-32-D **R2:**D-75-D **R3:**D-115-D **R4:**D-42B-31-D
ESOGU_C10_TW1	7531.00	0.23	**R1:**D-24-22A-D **R2:**D-34-23-45-31-D **R3:**D-119-113-13-14-D
ESOGU_R10_TW1	6822.0	0.06	**R1:**D-21-22E-22A-D **R2:**D-56-122-60A/2-D **R3:**D-78-74-77-70-D
ESOGU_RC10_TW1	7350.00	0.19	**R1:**D-21-22E-13-14-D **R2:**D-24-45-22A-D **R3:**D-56-122-74-D
ESOGU_C20_TW1	12737.00	0.44	**R1:**D-2-26-37B-D **R2:**D-9-8-cs8-115-D **R3:**D-44-58/1-58/2-D **R4:**D-49-51-50-153--D **R5:**D-116-27-30-31-39-D **R6:**D-42B-43C-D
ESOGU_R20_TW1	14022.00	4.51	**R1:**D-6-7-45-22D-35-D **R2:**D-44-64-61B-55-D **R3:**D-82-74-62-D

			R4:D-116-119-cs7-10-115 -D **R5:**D-152-D **R6:**D-154-60A/2-43C-D
ESOGU_RC20_TW1	12447.00	0.36	**R1:**D-2-cs8-10-115-D **R2:**D-44-64-61B-62-43C-39-D **R3:**D-50-45-22D-35-D **R4:**D-51-82-58/1-58/2-D **R5:**D-152-D **R6:**D-42B-153-D
ESOGU_C40_TW1	21151.00	3.41	**R1:**D-2-8-113-29-D **R2:**D-3-116-27-37A-41-D **R3:**D-9-17-18-D **R4:**D-25-1-5-115-35-D **R5:**D-38-28-30-39-D **R6:**D-44-60A/2-55-D **R7:**D-49-50-120-D **R8:**D-51-52-D **R9:**D-57-58/1-58/2-D **R10:**D-22B-117-15-16-22C-D **R11:**D-42C-42B-152-D
ESOGU_R40_TW1	26317.00	75.09	**R1:**D-19-17-12-13-D **R2:**D-21-20-120-41-46-D **R3:**D-23-8-11-cs8-10-22C-D **R4:**D-24-27-119-22A-D **R5:**D-38-28-1-125-113-cs4-D **R6:**D-44-64-74-73-D **R7:**D-57-122-61B-43B-D **R8:**D-76-88-cs1-124-60B/2 -D **R9:**D-154-53-D **R10:**D-42A-D **R11:**D-42C-153-D
ESOGU_RC40_TW1	23375.00	19.31	**R1:**D-2-8-20-15-10-cs8-D **R2:**D-3-cs7-1-125-113-115-35-D **R3:**D-19-21-17-11-13-D **R4:**D-24-25-119-22A-D **R5:**D-38-28-39-D **R6:**D-49-51-50-120-D **R7:**D-57-64-61B-55-D **R8:**D-76-124-60B/2-D **R9:**D-152-D **R10:**D-153-D **R11:**D-42C-42B-58/1-58/1-D
ESOGU_C60_TW1	*28714.00	10800.50	**R1:**D-1-5-113-cs8-4-2-36-D **R2:**D-7-22D-22A-24-D **R3:**D-9-12-13-14-D **R4:**D-16-23-10-22C-D **R5:**D-21-18-118-19-D **R6:**D-30-35-119-34-116-32-D **R7:**D-47-3-25-115-26-28-D **R8:**D-59-60A/2-60B/2-52-D **R9:**D-65-61B-56-61A/1-70-71-D **R10:**D-66-63-62-68-69-D **R11:**D-37B-40-39-38-31-D

			R12:D-42A-49-D **R13:**D-43C-154-53-42C-D
ESOGU_R60_TW1	*29392.00	10801.17	**R1:**D-13-16-23-20-45-D **R2:**D-30-cs7-113-2-8-12-19-D **R3:**D-36-37B-37A-38-116-33-D **R4:**D-50-82-84-52-D **R5:**D-72-68-74-78-81-D **R6:**D-123-cs3-77-75-79-88-D **R7:**D-154-53-153-D **R8:**D-22D-27-28-29-D **R9:**D-42A-25-114-5-4-D **R10:**D-43A-44-59-cs9-56-122-65-57-D **R11:**D-43C-42B-40-41-D **R12:**D-60A/2-60B/2-63-62-67-73-D
ESOGU_RC60_TW1	*29997.00	10801.78	**R1:**D-23-16-21-D **R2:**D-24-22D-22A-33-18-19-D **R3:**D-26-27-29-30-31-D **R4:**D-28-38-119-34-25-116-D **R5:**D-50-52-62-68-73-63-D **R6:**D-57-56-122-61A/1-61B-65-D **R7:**D-77-79-59-60A/2-60B/2-D **R8:**D-114-5-113-8-4-D **R9:**D-123-78-81-cs3-88-D **R10:**D-154-53-153-D **R11:**D-22C-10-13-14-45-36-37B-D **R12:** D-43A-43C-42B-40-D

* En iyi çözüm değildir. Büyük ölçekli problemlerde, matematiksel model makul bir sürede çözümler sağlayamamıştır. CPLEX tarafından üç saatlik bir süre sınırlaması ile elde edilen sonuçlar. 5, 10, 20, 40 müşterili problemler için elde edilen çözümler optimal çözümlerdir.

Bölüm 4.2'de sunulan matematiksel model, ESOGU_EVRP-PDP-TW veri setindeki optimal rotaları elde etmek için kullanılmıştır. Bu veri seti için CPLEX çözücüsü ile optimal sonuçlar elde edilmiştir ve bu sonuçlar Çizelge 4.3'te gösterilmektedir. Çizelge 4.3, veri setindeki orta zaman penceresi kısıtına ait 18 problem için matematiksel model sonuçlarını gösterir. Yürütme süresi, toplam mesafe, elde edilen rotalar Çizelge 4.3'te gösterilmiştir. Matematiksel model, 5, 10 müşteriyi içeren küçük boyutlu problemler için makul bir sürede sonuç üretebilir. 20, 40, 60 ve 80 müşteriyi içeren büyük boyutlu problemlerin çözümü çok uzun sürdüğü için model yaklaşık 3 saat çalıştırılmış ve bu süre sonunda elde edilen sonuçlar verilmiştir. Büyük boyutlu problemler için optimal sonuçlar makul bir sürede elde edilememiştir.

Çizelge 4.3. EVRP-PDP-TW'nin tam şarj stratejisine göre CPLEX'den elde edilen optimal sonuçları

Örnek Adı	**Mesafe(m)**	**Yürütme Süresi (s)**	**Rotalar**
ESOGU_C5_TW1	3705.48	0.64	**R1:**D-154-cs5-22A-10-13-14-D
ESOGU_R5_TW1	4013.27	0.36	**R1:**D-7-25-cs5-42C-43A-74-D
ESOGU_RC5_TW1	4235.62	0.75	**R1:**D-43A-74-cs3-10-13-14-D
ESOGU_C10_TW1	3255.85	0.61	**R1:**D-11-12-33-36-37B-37A-cs5-42A-42C-40-41-D
ESOGU_R10_TW1	5203.27	2.69	**R1:**D-9-14-6-33-37A-cs5-121-58/2-61A/2-46-45-D
ESOGU_RC10_TW1	4566.68	1.97	**R1:**D-11-9-12-6-33-36-37B-37A-cs5-58/2-45-D
ESOGU_C20_TW1	*4590.68	10800.02	**R1:**D-52-122-61A/2-61A/1-61B-56-60A/2-60A/1-60B/1-60B/2-121-cs5-18-17-117-16-20-22B-22E-22C-21-D
ESOGU_R20_TW1	7797.43	7814.91	**R1:**D-37A-31-43A-61A/1-61B-65-76-80-cs3-73-72-67-60A/1-54-39-38-34-cs7-9-23-20-22C-D
ESOGU_RC20_TW1	*6542.60	10800.01	**R1:**D-38-31-43A-122-76-80-cs3-73-72-121-60A/1-54-cs5-18-17-117-16-23-20-22B-22E-21-D
ESOGU_C40_TW1	*7694.95	10800.02	**R1:**D-18-22B-22E-22C-21-17-117-16-8-11-12-20-D **R2:**D-47-46-7-6-33-36-37B-37A-41-42A-42C-43A-43B-44-55-cs9-60A/1-60A/2-121-60B/2-60B/1-56-122-61A/1-61A/2-61B-52-51-49-D
ESOGU_R40_TW1	*10724.85	10800.02	**R1:**D-36-46-35-29-2-114-5-125-cs8-9-12-20-22B-22C-19-22D-33-D **R2:**D-40-39-38-42B-43C-43A-57-122-61A/2-64-70-84-cs3-73-66-68-74-75-79-124-82-52-153-49-120-D
ESOGU_RC40_TW1	*9091.72	10800.02	**R1:**D-51-52-84-cs3-66-121-60A/1-60A/2-60B/1-60B/2-68-74-79-82-122-61A/1-61A/2-61B-56-43C-49-D **R2:**D-120-33-36-46-35-29-38-5-2-cs8-12-16-20-117-17-22B-22E-22C-21-19-18-D
ESOGU_C60_TW1	*12935.75	10800.02	**R1:**D-45-35-119-116-32-24-22D-22A-3-25-cs7-115-1-2-4-114-113-125-10-13-14-23-16-D **R2:**D-18-17-117-20-22B-22E-22C-21-D **R3:**D-52-74-76-77-75-79-124-78-80-81-73-123-82-cs2-122-61A/2-61A/1-61B-56-60A/1-60A/2-60B/1-60B/2-121-59-cs9-43C-42B-53-54-154-39-D

ESOGU_R60_TW1	*14922.99	10800.02	**R1:**D-51-52-43C-42B-38-35-29-30-40-41-42A-cs5-53-7-24-22D-6-D **R2:**D-55-58/1-56-122-61A/1-61A/2-64-65-71-85-86-83-82-cs3-77-75-79-124-87-80-72-66-62-60B/1-121-60A/1-59-cs9-D **R3:**D-19-118-22E-21-17-16-23-12-13-14-cs8-8-4-5-113-1-26-3-33-D
ESOGU_RC60_TW1	*13613.28	10800.01	**R1:**D-45-3-25-115-1-2-4-5-113-125-8-cs8-10-13-14-23-16-22E-21-118-24-22D-22A-6-33-D **R2:**D-58/1-56-61A/1-61A/2-64-71-82-85-cs3-74-78-76-77-75-79-124-87-80-81-123-73-72-60B/1-121-60A/1-59-cs9-55-43C-51-D **R3:**D-38-35-29-30-41-154-54-53-D
ESOGU_C80_TW1	*18013.20	10800.03	**R1:**D-51-52-cs3-72-78-80-81-74-88-cs1-79-76-77-75-124-82-70-71-122-56-61A/1-61A/2-cs2-43C-42B-40-41-39-36-37B-37A-D **R2:**D-42C-43A-43B-44-55-61B-64-65-73-69-68-63-62-67-66-121-60A/1-cs9-60A/2-60B/2-60B/1-123-cs5-42A-49-6-7-33-D **R3:**D-47-46-35-32-3-25-115-1-125-4-2-cs8-8-11-9-12-20-117-16-17-22B-22E-22C-21-18-118-19-D
ESOGU_R80_TW1	*27345.64	10800.02	**R1:**D-120-18-cs10-83-70-71-74-78-80-123-86-75-79-124-cs1-88-87-76-77-67-66-68-73-62-60A/2-60B/1-60B/2-121-59-cs9-D **R2:**D-57-56-122-61A/2-61A/1-61B-cs4-28-29-5-113-2-125-8-10-15-23-11-cs8-D **R3:**D-34-1-26-116-27-47-cs6-35-22D-7-6-33-36-37B-cs7-51-52-49-50-D **R4:**D-9-12-13-14-16-20-117-17-22B-22E-cs5-42A-153-42C-154-43B-43C-42B-41-39-D
ESOGU_RC80_TW1	*19748.42	10800.01	**R1:**D-17-117-16-10-11-9-12-13-14-15-23-20-22B-22E-21-18-22C-D **R2:**D-50-51-52-73-66-67-68-74-78-80-81-cs3-87-124-75-79-77-76-123-82-cs2-43C-42B-41-39-42A-D **R3:**D-154-42C-153-43B-59-cs9-60A/2-60B/1-121-60B/2-57-56-122-61A/1-70-71-D **R4:**D-47-34-1-26-3-116-32-25-cs7-115-2-4-5-125-8-113-27-28-35-cs6-7-6-33-36-37B-120-D

* En iyi çözüm değildir. Büyük ölçekli problemlerde, matematiksel model makul bir sürede çözümler sağlayamamıştır. CPLEX tarafından üç saatlik bir süre sınırlaması ile elde edilen sonuçlar. 5, 10 müşterili problemler için elde edilen çözümler optimal çözümlerdir.

Matematiksel modeller, kısıtları ve amaç fonksiyonlarını açıkça tanımlayarak Elektrikli Araç Yönlendirme Problemi (EVRP) gibi rotalama problemlerinin kesin bir formülasyonunu sağlar. Buna karşılık, Q-öğrenme, DQN gibi RL yöntemleri aynı problemi eylemler, durumlar ve ödüller aracılığıyla temsil eden bir RL çerçevesine dönüştürür. Matematiksel modeller probleme kesin bir çözüm bulmayı hedeflerken, RL yöntemleri

ödüller ve cezalar temelinde bir öğrenme süreci yürüterek hedef fonksiyon değerine yaklaşır. Matematiksel modellerde, kısıtlamalar açıkça tanımlanır ve çözümün bunlara kesinlikle uyması gerekir. Ancak RL'de, bu kısıtlamalar ödül fonksiyonu aracılığıyla dolaylı olarak öğrenilir. Matematiksel modeller kesin çözümler üretirken, öğrenmeye dayalı bir yaklaşım olan RL zamanla iyileşen ve yakınsayan çözümler üretir. Bu özellik, kesin yöntemlerin hesaplama açısından engelleyici olabileceği büyük ölçekli ve karmaşık problemleri çözmek için özellikle avantajlıdır.

4.2. CEVRP için Q-Öğrenme Algoritmasının Sonuçları

Tezin bu bölümünde, CEVRP için önerilen Q-öğrenme modeli Bölüm 3.1.2'de tanıtılan ESOGU-CEVRPTW veri seti kullanılarak test edilmiştir. Ele alınan problem bir zaman penceresi kısıtlaması içermediğinden, bu kısıtlama veri setinden göz ardı edilmiştir. EV'nin yük kapasitesi 350 kg ve pil kapasitesi 3000 kWsec'dir. Şarj istasyonlarını ziyaret etmeyle ilgili kısıtlamaların test ortamında etkili bir şekilde gösterildiğinden emin olmak için nispeten küçük bir pil kapasitesi kullanılmıştır.

Önerilen yöntem veri setindeki 15 örnek üzerinde test edilmiştir. Her örnekte, rota sayısı müşteri sayısına ve taleplerine göre değişir. EV depodan başlar ve yük kapasitesinin izin verdiği ölçüde müşterilerin taleplerini karşılar. Mevcut yük kapasitesi yetersiz olduğunda, EV tam kapasiteyi yeniden yüklemek için depoya geri döner. Ayrıca, EV'nin şarj seviyesi yolda iken bir sonraki müşteriye ulaşmak için yeterli değilse, tam şarj için en uygun şarj istasyonuna yönlendirilir. Amaç, müşteri talebi ve EV'nin şarj durumu ile ilgili kısıtlamalara uyularak en kısa mesafeli rotayı hesaplamaktır.

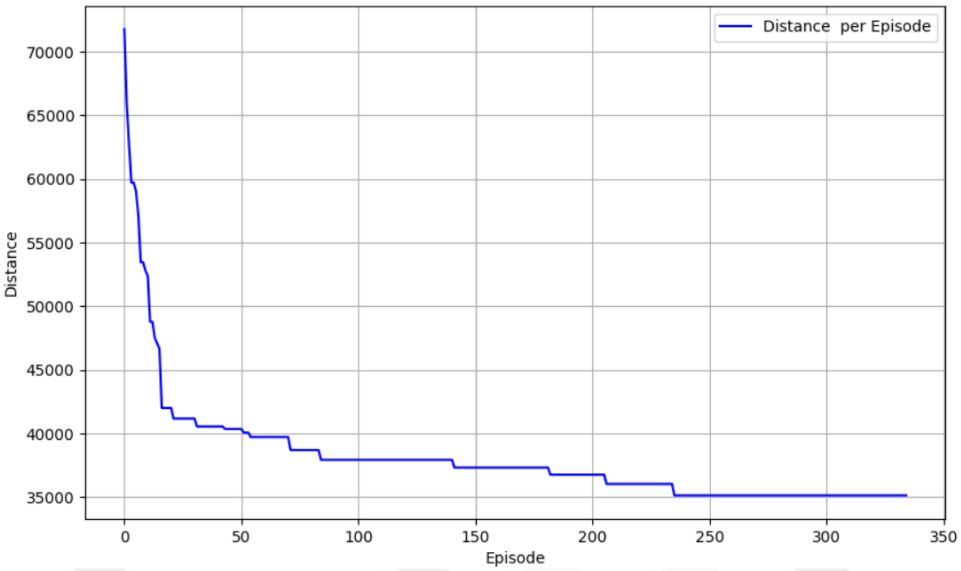
Önerilen Q-öğrenme yönteminin tüm testler i5-7500 CPU'lu (3,40 GHZ ve 12 RAM) bir masaüstü bilgisayarda gerçekleştirilmiştir. Kod Python 3.8 kullanılarak yazılmıştır. Önerilen modelin performansını değerlendirmek için model, n=5, 10, 20, 40 ila 60 müşteri sayısı arasında değişen çeşitli örnek boyutlarında test edilmiştir. Eğitim, Çizelge 4.4'te özetlenen hiper parametreler kullanılarak gerçekleştirilmiştir. Çizelge 4.4'te sunulan hiper parametreler, Cetiner vd., (2024) çalışmasındaki mutasyona dayalı beyaz kutu testi yoluyla belirlenmiştir. Bu çalışma, elektrikli araç rotalaması için geliştirilen bir RL modelinin, Uzun Kısa Süreli Bellek (LSTM) modelinin ve Transformatör tabanlı bir sinir ağı modelinin

mutasyona dayalı testlerine odaklanmaktadır. Çalışmadaki derin mutasyon modülü, derin öğrenme modellerinin sağlamlığını ve doğruluğunu değerlendirmek için geliştirilmiş bir araçtır. Çeşitli derin model türleri için kullanılabilir ve bu modeller farklı yapılar kullandığından, kullanılan değerlendirme ölçütleri model türüne bağlı olarak değişir. RL modellerinde, ölçütler modele bağlı olarak değişebilir. Elektrikli araçlar için en uygun rotayı bulmak için kullanılan bir ölçüt olan Reward_max, RL modelinde kullanılır. RL modeli için gama, alfa, epsilon ve bozunma parametreleri seçilerek her parametre için testler yapılmıştır.

Çizelge 4.4. Eğitim için kullanılan hiper parametreler

Parametre	Değer
İterasyon sayısı	20000
İndirim faktörü γ	0.6
Öğrenme oranı α	0.1
Maksimum epsilon	1.0
Minimum epsilon	0.1
Epsilon bozunma oranı	0.999

Şekil 4.2'de sunulan öğrenme eğrisi, önerilen Q-öğrenme algoritmasının ESOGU_RC60 örneği için belli bir iterasyon boyunca eğitilmesi sonucunda kat edilen toplam mesafeyi en aza indirmedeki performansını göstermektedir. Veri noktaları, algoritmanın öğrenme ilerlemesinin kapsamlı bir genel görünümünü sağlamak için her 20 bölümde bir örneklenir ve çizilir. Başlangıçta, toplam mesafe, ajanın keşif davranışını ve optimal olmayan yönlendirme kararlarını yansıtan daha yüksek değerler gösterebilir. Eğitim ilerledikçe, genel bir aşağı yönlü eğilim gözlemlenir ve bu, Q-öğrenme ajanının daha verimli rotalar elde etmek için politikasını etkili bir şekilde öğrendiğini ve iyileştirdiğini gösterir. Ardışık bölümler boyunca toplam mesafedeki bu azalma, algoritmanın rotalama kararlarını optimize etme kapasitesini ve dolayısıyla genel performansı artırdığını gösterir. Eğrinin daha düşük mesafe değerlerine doğru sabitlenmesi, algoritmanın yakınsamaya yaklaştığını ve CEVRP için sürekli olarak optimala yakın çözümler belirlediğini gösterir.



Şekil 4.2. Önerilen Q-Öğrenme algoritmasının öğrenme eğrisi

Önerilen yöntem ESOGU_CEVRPTW veri setinde test edilmiş ve araç kapasitesi 350, araç aküsü ise 3000 olarak kabul edilmiştir. Önerilen yöntem daha seyrek bir veri kümesi veya farklı pil kapasitelerine sahip EV'ler gibi farklı veri setlerinde de çalışmaktadır. Önerilen yöntemin bir rota üzerindeki iki müşteri arasında birden fazla şarj gerektiren durumlarda da çalışabileceğini göstermek amacıyla aracın pil kapasitesini düşürerek 1000 olarak kabul eden ek bir deney tasarlanmıştır. Bu durumda iki müşteri arasında bile ilerleyebilmek için aracın birden fazla şarj istasyonunda durması gerekmiştir. EV pil kapasitesi 3000 olduğunda önerilen algoritma tarafından üretilen rota "R1:D-35-32-115-2-CS9-D R2:D-42B-D" ve toplam mesafe 3989,19 m'dir. Pil kapasitesi 1000 olduğunda oluşturulan rota "R1: D-32-CS8-115-2-CS9-CS7-35-D R2:D-42B-D" olup toplam mesafe 4686,75 m'dir. Bu senaryoda elektrikli aracın iki müşteri lokasyonu arasında birden fazla şarj durağına ihtiyaç duyduğu görülmektedir.

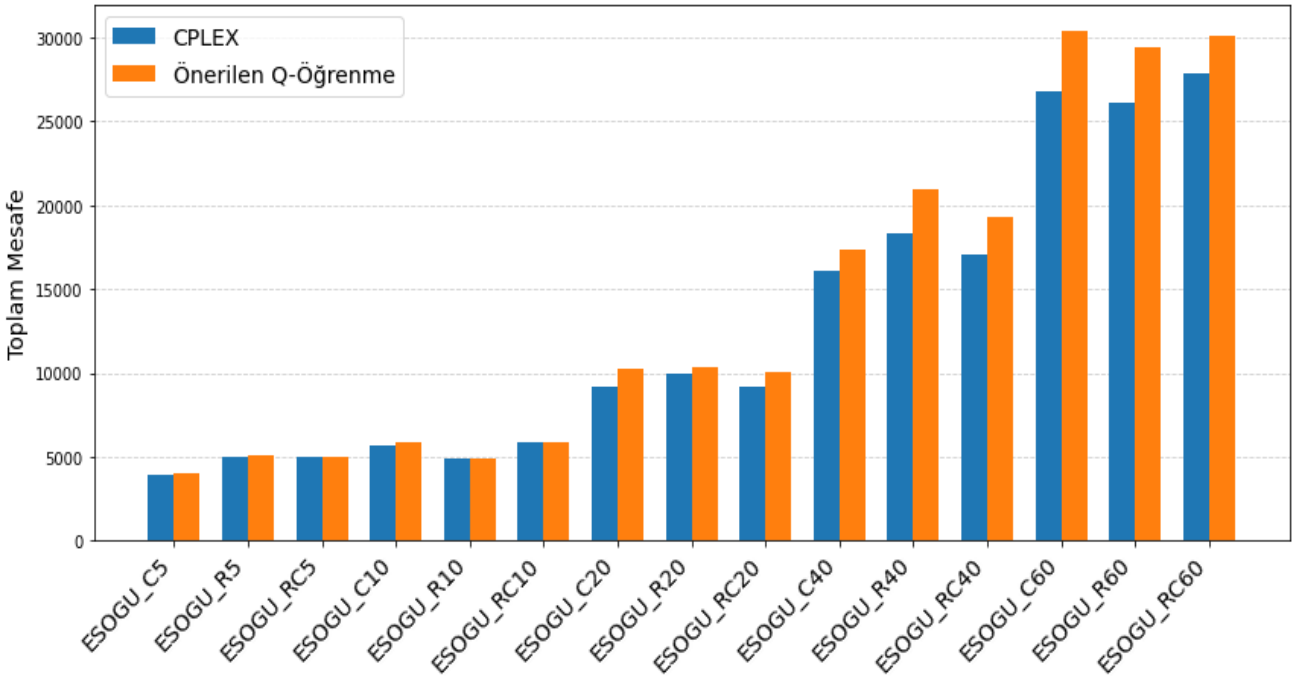
Bu çalışmada, önerilen algoritma matematiksel modelle elde edilen optimal değerlerle karşılaştırılmıştır. Çizelge 4.5'te matematiksel modelin ve önerilen Q-öğrenme algoritmasının karşılaştırmalı sonuçları gösterilmektedir. İki yaklaşım arasındaki GAP değeri Denklem 4.1'deki formülle hesaplanır.

$$Gap = \frac{Mesafe_{CPLEX} - Mesafe_{Q_Öğrenme}}{Mesafe_{CPLEX}} \; x100\% \qquad (4.1)$$

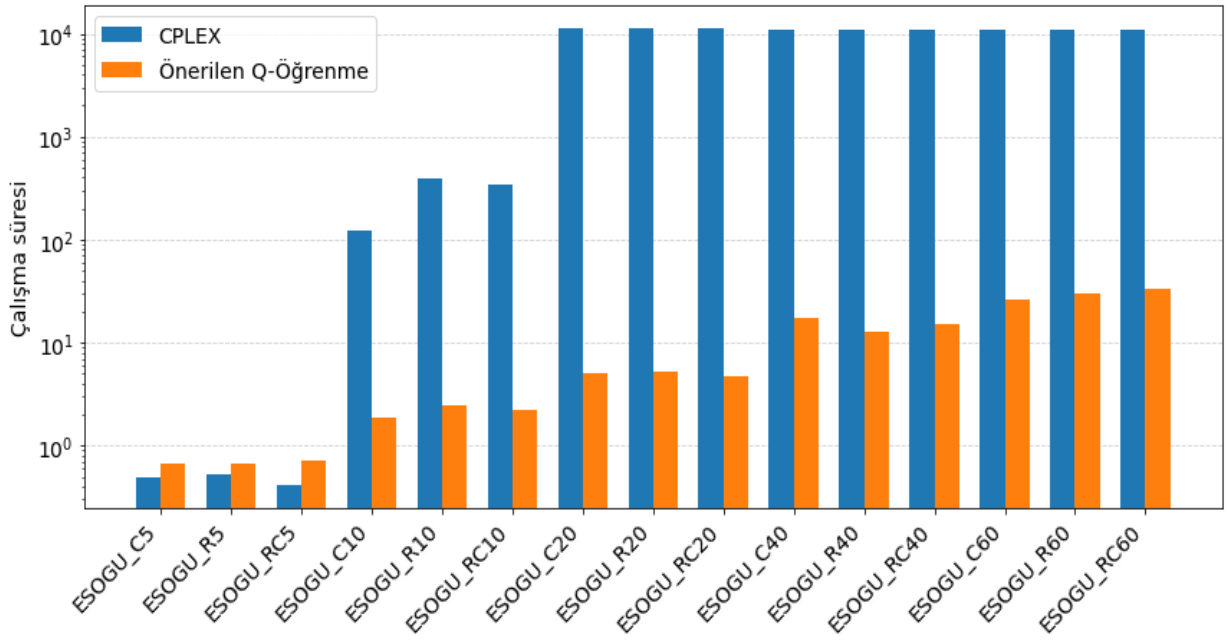
Çizelge 4.5. Matematiksel model ve önerilen Q-Öğrenme'nin karşılaştırmalı sonuçları

Veri seti örnekleri	CPLEX çözücü		Önerilen Q-Öğrenme		
	Mesafe (m)	Çalışma Süresi (s)	Mesafe (m)	Çalışma Süresi (s)	GAP
ESOGU_C5	3956.00	0.50	3989.19	0.66	0.008%
ESOGU_R5	4977.00	0.53	5064.28	0.67	0.01%
ESOGU_RC5	4944.00	0.41	4944.00	0.71	0.0%
ESOGU_C10	5646.00	124.33	5871.79	1.89	0.03%
ESOGU_R10	4889.00	389.74	4898.02	2.48	0.001%
ESOGU_RC10	5843.00	345.63	5851.72	2.26	0.001%
ESOGU_C20	9125.00	11160.69	10203.54	5.12	0.11%
ESOGU_R20	9974.00	11134.66	10292.60	5.15	0.03%
ESOGU_RC20	9161.00	11113.25	10082.14	4.76	0.10%
ESOGU_C40	16102.00	10800.05	17358.23	17.26	0.07%
ESOGU_R40	18276.00	10800.02	20906.46	12.74	0.14%
ESOGU_RC40	17053.00	10800.02	19293.44	15.01	0.13%
ESOGU_C60	26773.00	10800.02	30392.72	26.19	0.13%
ESOGU_R60	26159.00	10800.03	29459.82	29.88	0.12%
ESOGU_RC60	27905.00	10800.02	30143.03	33.31	0.08%

Önerilen Q-öğrenme modeli, Şekil 4.3'te gösterildiği gibi küçük boyutlu test problemlerinde optimal çözümlerle neredeyse aynı sonuçları verirken büyük boyutlu test problemlerde de oldukça yakın sonuçlar vermiştir. Önerilen Q-öğrenme modeli, matematiksel modele kıyasla yürütme süresini önemli ölçüde azaltmıştır. Büyük ölçekli problemler, matematiksel modelin gerektirdiği 3 saatin aksine, yaklaşık 30 saniyede çözülebilir. Bu nedenle, önerilen yöntem, Şekil 4.4'te gösterildiği gibi belirgin şekilde daha kısa yürütme sürelerine sahiptir.



Şekil 4.3. CPLEX çözücü ve Q-Öğrenme sonuçlarının toplam mesafe açısından karşılaştırılması



Şekil 4.4. CPLEX çözücü ve Q-Öğrenme sonuçlarının toplam çalışma süresi açısından karşılaştırılması

Matematiksel model, optimal çözümler sağlamasına rağmen, önemli miktarda hesaplama kaynağı ve zaman gerektirirken, RL yaklaşımları neredeyse anında çözümler sunmaktadır. Hesaplama süresindeki bu önemli azalma, zamanında karar almanın kritik

olduğu gerçek dünya lojistik operasyonlarında RL yöntemlerinin pratik olarak uygulanabilirliğini vurgulamaktadır.

Son kilometre teslimatı kapsamında, kısa sürede yüksek kaliteli çözümler üretebilen algoritmalar şirketler için kritik öneme sahiptir. Genel olarak, karşılaştırmalı analiz önerilen Q-öğrenme algoritmasının yalnızca yüksek doğrulukla optimal çözümlere yaklaşmakla kalmayıp aynı zamanda geleneksel matematiksel modellere göre üstün yürütme hızını vurgulamaktadır. Bu özellikler Q-öğrenmesini operasyonel ortamlarda CEVRP'yi çözmek için oldukça umut verici bir yaklaşım haline getirerek çözüm kalitesi ve hesaplama verimliliği arasında bir denge sunmaktadır.

4.3. CEVRPTW için DQN Modelinin Sonuçları

Önerilen DQN yöntemi ESOGU-CEVRPTW veri setindeki 15 örnekte test edilmiştir. EV depodan başlar ve yük kapasitesinin izin verdiği sürece müşterilerin taleplerini karşılar. Mevcut yük kapasitesi yetersiz olduğunda, EV tam kapasite için depoya geri döner. Ayrıca, EV'nin şarj seviyesi yolda iken bir sonraki müşteriye ulaşmak için yeterli değilse, tam şarj için en uygun şarj istasyonuna yönlendirilir. Amaç, müşteri talebi, zaman penceresi ve EV'nin şarj durumu ile ilgili gerekli tüm kısıtlamalara uyarak en kısa mesafeli rotayı hesaplamaktır.

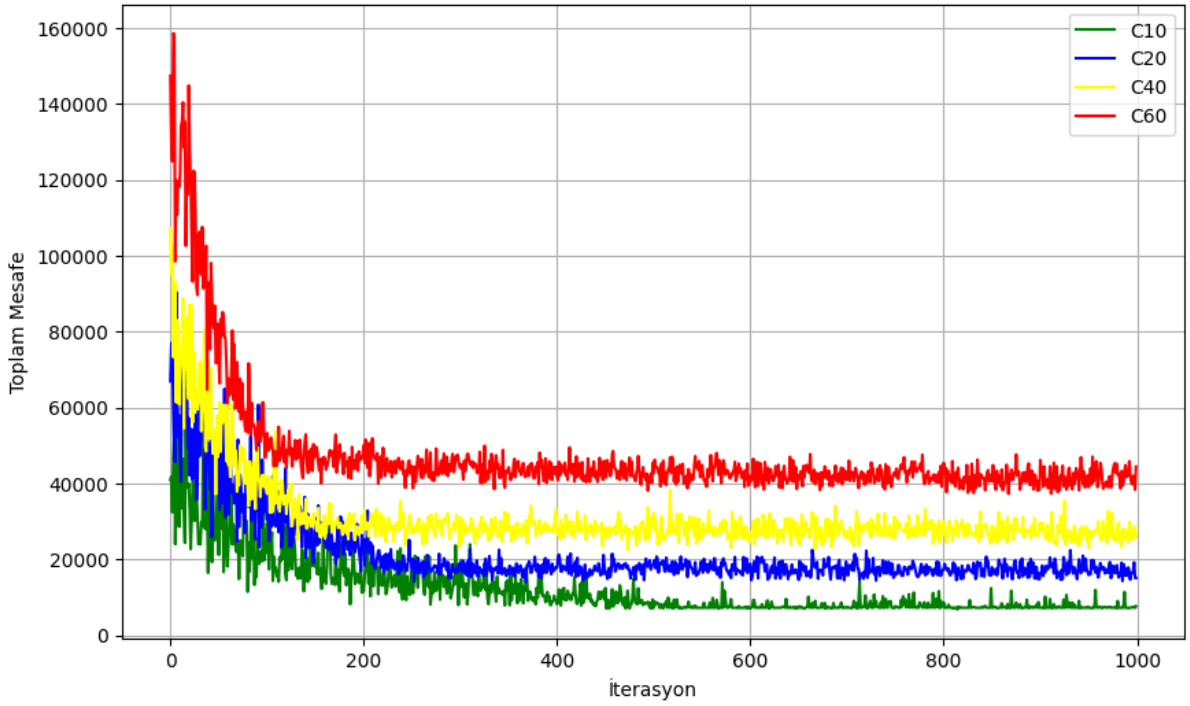
Tüm testler 3,40 GHz'de Intel i7-13700K ve bir NVDIA Geforce RTX 4070 GPU'ya sahip masaüstü bilgisayarda gerçekleştirilmiştir. Kod python'da tensorflow kütüphanesi kullanılarak yazılmıştır. Eğitim, Çizelge 4.6'da sunulan hiper parametreler kullanılarak gerçekleştirilmiştir. Önerilen DQN'deki hiper parametreler seçilirken literatürdeki çalışmalardan yararlanılmıştır. Hiper parametre seçimi sırasında öğrenme oranı için Wang vd., (2024) çalışması, bellek kapasitesi ve ağırlık kopyalama aralığı için ise Nouicer vd., (2023) çalışmasında önerilen değerler kullanılmıştır. Öğrenme oranı belirlenen değerden daha yüksek seçildiğinde, rota optimizasyonu başarısı düşerek daha uzun mesafeli çözümler elde edilmiştir. Önerilen model kapsamında diğer hiper parametrelerin belirlenmesi amacıyla literatürde önerilen değer aralıkları dikkate alınarak kapsamlı deneysel testler gerçekleştirilmiştir. Belirlenen hiper parametreler modelin eğitim süresince tutarlı ve kısa mesafeli rotalar elde etmesini sağlamıştır.

Çizelge 4.6. DQN eğitimi için kullanılan hiper parametreler

Parametre	**Değer**
İterasyon sayısı	500
Küme boyutu (Batch size)	32
Bellek kapasitesi	10000
Ağırlık kopyalama aralığı	100
Durum vektörü boyutu	64
İndirim faktörü γ	0.99
Öğrenme oranı α	0.0001
Maksimum epsilon	1.0
Minimum epsilon	0.01
Epsilon bozunma oranı	1e-4

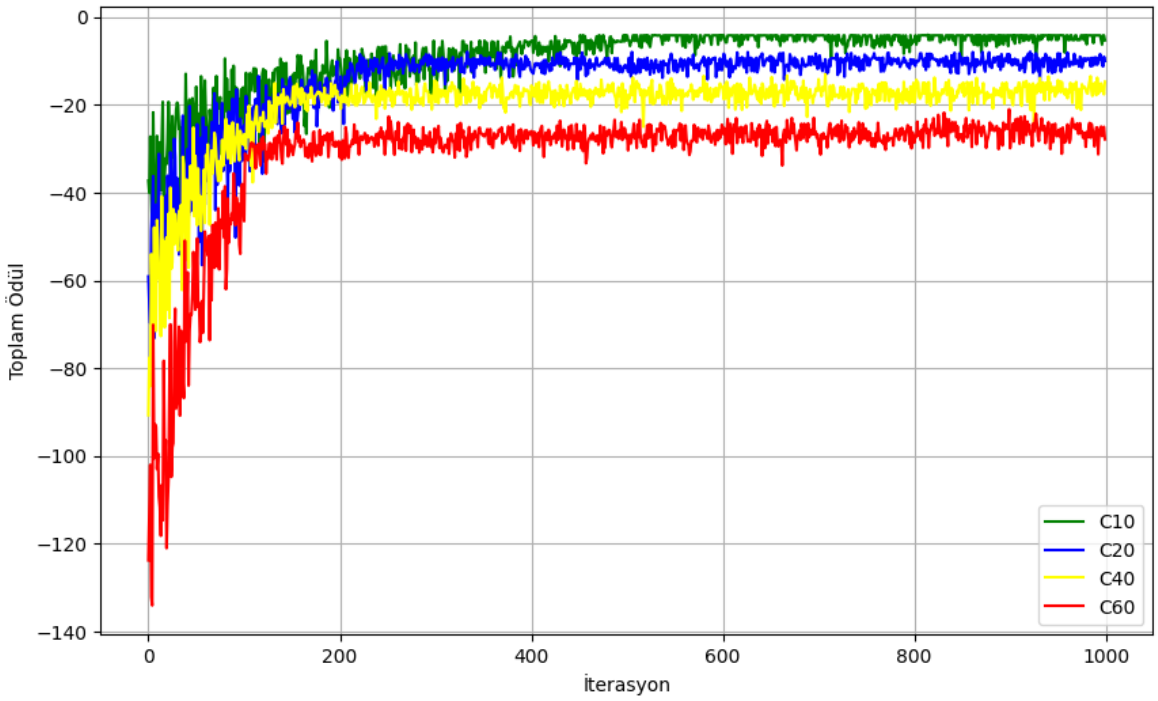
Önerilen modelin eğitim aşaması sırasında, her sayıdaki problem için (5,10,20,40,60 müşteri) 1.000 örnek üretilmiştir. 20 düğüm için yapılan eğitimde, her iterasyon yaklaşık 6 dakika sürmüş ve 100 iterasyon yaklaşık 10 saat sürmüştür. 40 ve 60 müşteri içeren eğitim verilerinin süresi ise daha uzundur.

Önerilen DQN algoritmasının öğrenme eğrisi Şekil 4.5'te gösterilmiştir. Burada belli bir iterasyon boyunca toplam mesafe değerinin C10, C20, C40 ve C60 örnekleri için nasıl değiştiği görülmektedir. C10, C20, C40 ve C60 sırasıyla 10, 20, 40 ve 60 müşteriden oluşan eğitim verisi örneklerini ifade eder.



Şekil 4.5. Önerilen DQN algoritmasının eğitimdeki toplam mesafe eğrisi

Eğitim sürecinin başlarında, DQN'deki ajan genellikle keşif eylemlerini seçtiğinden daha yüksek mesafeler üretir. Ancak eğitim ilerledikçe, daha verimli rotalar üretmek için politikasını iyileştirdiğinden net bir aşağı yönlü eğilim ortaya çıkar. Daha büyük boyutlu eğitim örneklerinde (C40, C60) toplam mesafe genellikle daha yüksek başlar ve zaman içinde eğitilerek eğitimin sonunda nispeten düşük seviyelerde sabitlenebilir. C10 gibi daha küçük boyutlu verilerde genellikle eğitim daha hızlı gerçekleşir ve daha düşük mesafe değerlerine daha erken ulaşır. Eğitimdeki dalgalanmalar DQN'nin keşif aşamasına normaldir ve burada periyodik politika değişiklikleri ve alternatif eylemlerin denenmesi performansta geçici değişikliklere neden olabilir. Genel olarak, tüm eğrilerdeki tutarlı azalma, DQN algoritmasının rotalama kararlarını optimize etme kapasitesini ve her eğrinin azaltılmış mesafe değerlerinde sabitlenmesi, algoritmanın CEVRPTW için neredeyse optimal çözümlere doğru yakınsadığını gösterir.



Şekil 4.6. Önerilen DQN algoritmasının eğitimdeki toplam ödül eğrisi

Önerilen DQN algoritmasının toplam ödül eğrisini Şekil 4.6'da gösterilmektedir. Genellikle azalan bir eğilim gösteren toplam mesafe eğrisinin (Şekil 4.5) aksine, toplam ödül eğrisi ardışık eğitim bölümleri boyunca net bir şekilde yukarı doğru ilerleme göstermektedir. Bu artış, daha yüksek ödüllerin genellikle daha iyi çözümlere karşılık gelmesi nedeniyle, ajanın daha verimli rotalama kararlarını belirleme ve uygulama becerisinin arttığını yansıtmaktadır. Toplam mesafe sonuçlarında olduğu gibi, küçük boyutlu örnekler daha yüksek ödüllere daha erken ulaşabilirken, büyük boyutlu örnekler nispeten daha geç ulaşır ve iyileşen bir ödül seviyesinde sabitlenmeden önce daha fazla eğitim bölümü gerektirebilir. Sonuç olarak, ödül eğrisinin daha yüksek değerlerde sabitlenme eğilimi, DQN'nin yakınsamasına işaret ederek algoritmanın CEVRPTW için güvenilir bir şekilde neredeyse optimal çözümler keşfettiğini göstermektedir.

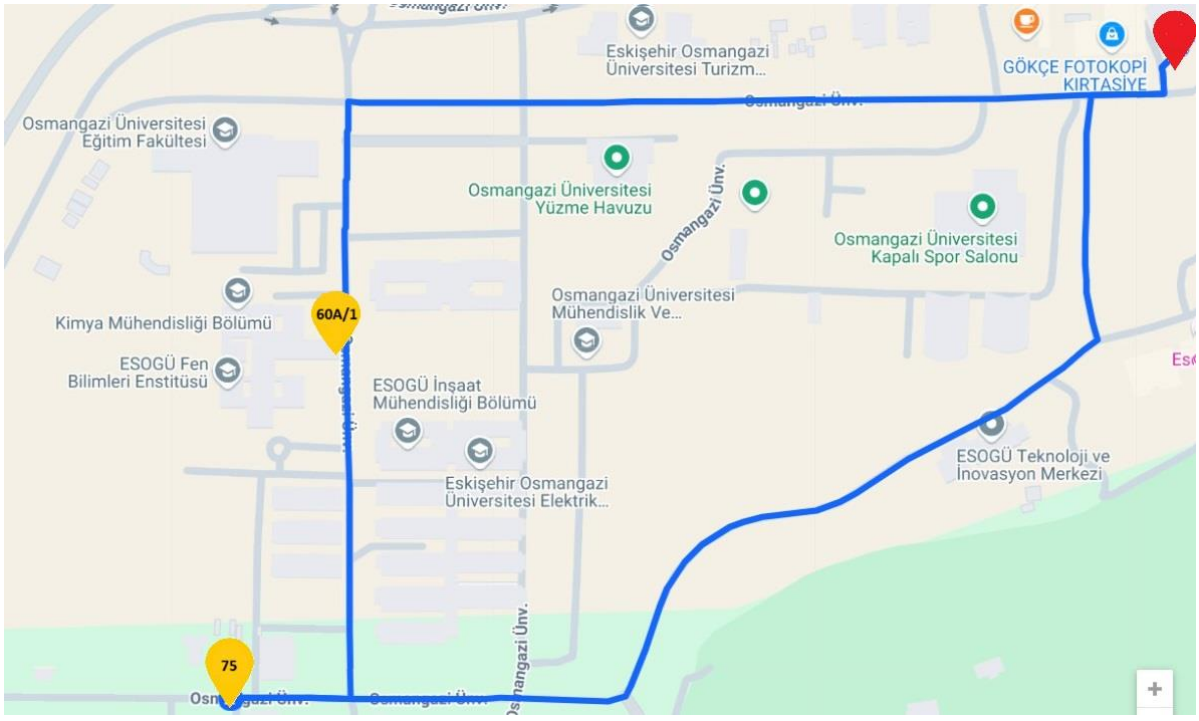
DQN algoritması ESOGU_CEVRPTW veri setinden seçilen 15 örnek üzerinde test edilmiştir ve sonuçlar Çizelge 4.7'de sunulmuştur. Çizelge 4.7, veri setindeki bazı örnekler için zaman penceresi dahil edilmeden CEVRP için önerilen modelin sonuçlarını göstermektedir. Çözülen örnekler için yürütme süresi, minimum mesafe ve optimal rota gösterilmiştir.

Çizelge 4.7. CEVRP'nin tam şarj stratejisine göre önerilen DQN modelinden elde edilen sonuçları

Örnek Adı	**Mesafe (m)**	**Yürütme Süresi (s)**	**Rotalar**
ESOGU_C5	3989	0.2192	R1:D-35-32-115-2-CS9-D R2:D-42B-D
ESOGU_R5	5064	0.2034	R1:D-60A/1-75-D R2: D-31-32-19-D
ESOGU_RC5	4994	0.2020	R1:D-31-42B-115-32-D R2:D-75-D
ESOGU_C10	5886	0.2988	R1:D-31-119-34-24-22A-D R2:D-13-14-113-26-45-CS7-D
ESOGU_R10	4898	0.2699	R1: D-56-122-70-74-77-78-60A/1-D R2: D-22A-22E-21-D
ESOGU_RC10	5853	0.2987	R1:D-56-122-74-45-D R2: D-22E-21-13-14-24-22A-D
ESOGU_C20	10760	0.4556	R1: D-153-51-50-37B-30-39-31-D R2:D-49-116-115-26-27-D R3:D-44-58/1-58/2-43C-42B-D R4:D-2-8-9-CS8-D
ESOGU_R20	10794	0.4418	R1:D-152-154-44-61B-64-D R2:D-45-35-119-116-115-10-CS8-D R3:D-55-60A/2-62-74-82-43C-D R4:D-7-6-22D-D
ESOGU_RC20	9950	0.4612	R1:D-152-153-51-50-45-22D-35-39-D R2:D-58/2-58/1-61B-64-82-62-D R3:D-42B-43C-44-D R3:D-115-2-10-D
ESOGU_C40	17929	0.9257	R1:D-42C-42B-39-120-28-29-30-35-D R2:D-49-3-25-116-27-38-D R3:D-152-52-51-50-41-D R4:D-22B-17-117-15-9-16-8-cs8-D R5:D-44-58/1-57-58/2-60A/2-55-D R6:D-5-113-2-115-1-cs8-D R7:D-22C-18-37A-D
ESOGU_R40	21773	0.9750	R1:D-42C-43B-61B-122-64-57-D R2:D-42A-53-153-44-154-120-D R3:D-20-17-11-13-12-23-8-cs8-28-D R4:D-73-74-76-124-88-cs1-60B/1-D R5:D-46-119-27-24-D R6:D-22A-22C-21-19-D R7:D-38-1-113-125-10-D R8:D-41-D
ESOGU_RC40	19686	1.0042	R1:D-49-50-120-19-20-17-21-D R2:D-42C-152-153-51-42B-39-D R3:D-38-28-119-25-1-D R4:D-55-58/1-58/2-57-61B-64-124-D R5:D-35-24-22A-3-113-125-8-cs8-13-D R6:D-60B/1-76-D R7:D-115-2-10-11-15-cs8-D
ESOGU_C60	29500	1.5297	R1:D-42A-42C-53-49-47-36-37B-39-D R2:D-154-52-71-65-56-D R3:D-40-38-35-28-30-31-43C-D R4:D-119-34-32-3-116-25-D R5:D-24-22D-7-18-D

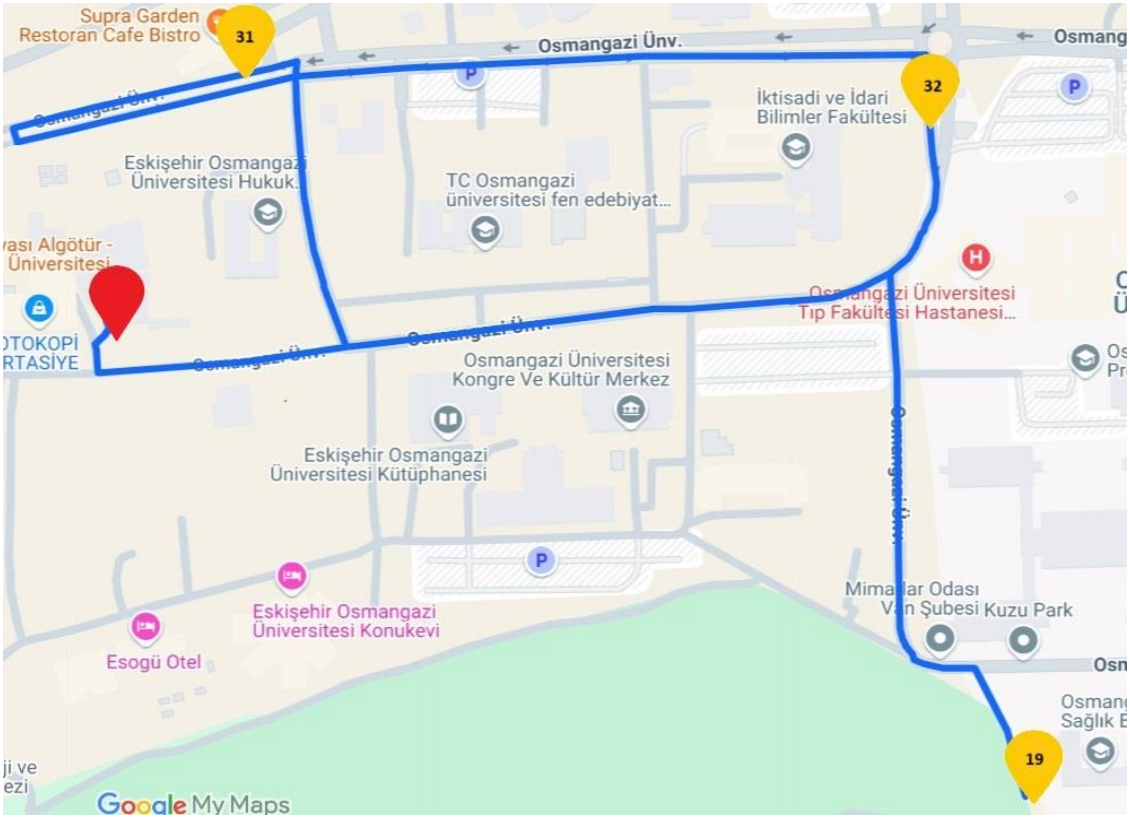
			R6:D-59-60A/2-60B/2-66-68-D R7:D-70-61A/2-69-62-63-61B-D R8:D-22C-21-118-16-D R9:D-19-14-13-9-12-113-CS8-D R10:D-2-4-5-26-D R11:D-115-1-10-23-CS8-D
ESOGU_R60	30191	1.4217	R1:D-53-154-43A-44-57-56-D R2:D-153-42B-30-38-28-29-36-37B-D R3:D-40-41-50-45-37A-33-D R4:D-52-84-123-74-81-D R5:D-122-65-78-73-68-D R6:D-25-116-32-27-113-5-cs8-D R7:D-43C-59-60A/2-60B/1-62-63-67-D R8:D-22D-20-17-16-12-23-cs8-D R9:D-19-8-4-114-2-cs8-D R10:D-72-75-79-77-87-88-cs1-D
ESOGU_RC60	30660	1.3676	R1:D-43A-43C-42B-40-31-D R2:D-53-154-153-50-45-36-37B-29-28-cs4-D R3:D-38-30-119-34-25-D R4:D-52-65-122-61A/2-61B-56-57-D R5:D-33-24-22D-22A-116-27-D R6:D-59-60A/2-60B/2-62-63-68-D R7:D-114-5-113-8-10-14-D R8:D-18-22C-21-16-D R9:D-123-79-77-78-81-D R10:D-73-88-cs1-D R11:D-19-13-23-4-cs8-26-D

Önerilen DQN modelinin ESOGU_R5 problemi için ürettiği rotalar Şekil 4.7 ve Şekil 4.8'de sunulmaktadır.



Şekil 4.7. ESOGU_R5 için önerilen DQN ile üretilen birinci rota

Şekillerdeki kırmızı işaretçiler depo konumunu, sarı işaretçiler müşterilerin konumlarını ve mavi çizgiler ise rota güzergahını göstermektedir. Model ESOGU_R5 problemi için iki ayrı rota üretmiştir. Birinci rota R1: D-60A/1-75-D Şekil 4.7'de, ikinci rota ise R2: D-31-32-19-D olarak Şekil 4.8'de gösterilmektedir.



Şekil 4.8. ESOGU_R5 için önerilen DQN ile üretilen ikinci rota

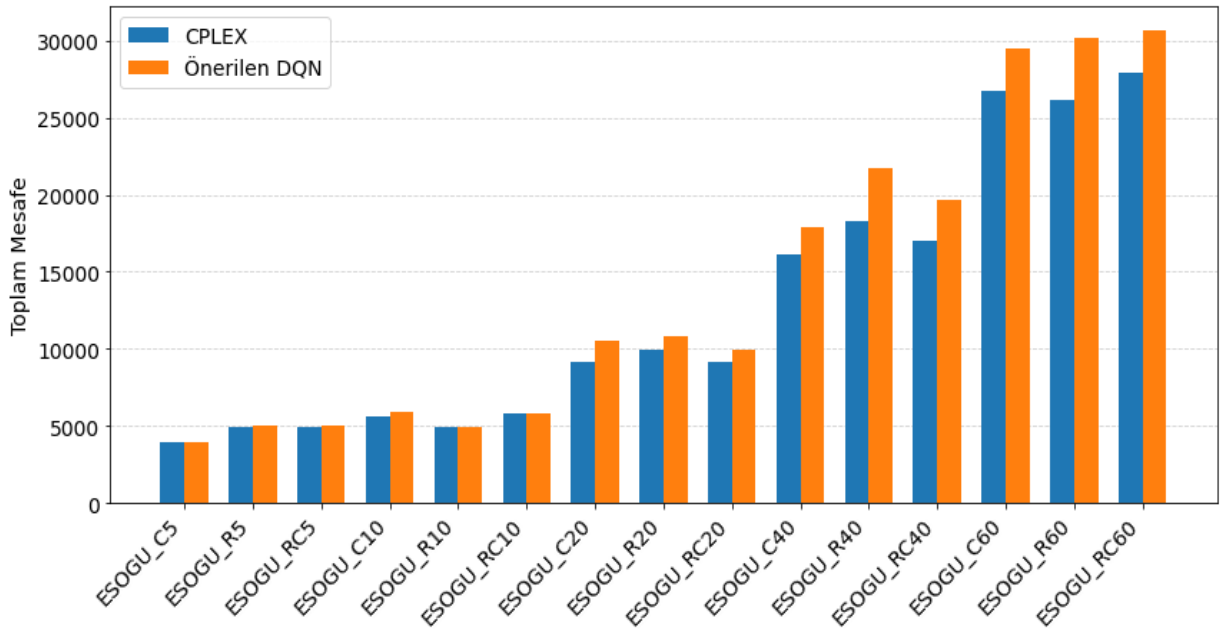
Önerilen DQN modelinin CEVRP için sonuçları matematiksel modelle elde edilen optimal sonuçlarla karşılaştırılmıştır ve sonuçlar Çizelge 4.8'de gösterilmektedir.

Çizelge 4.8. CEVRP için önerilen DQN modelinin CPLEX çözücüden elde edilen sonuçlarla karşılaştırılması

Örnek Adı	**CPLEX Çözücü**		**Önerilen DQN**		**GAP**
	Mesafe (m)	**Yürütme Süresi (s)**	**Mesafe (m)**	**Yürütme Süresi (s)**	
ESOGU_C5	3956	0.5	3989	0.2192	0.008%
ESOGU_R5	4977	0.53	5064	0.2034	0.01%
ESOGU_RC5	4944	0.41	4994	0.2212	0.01%

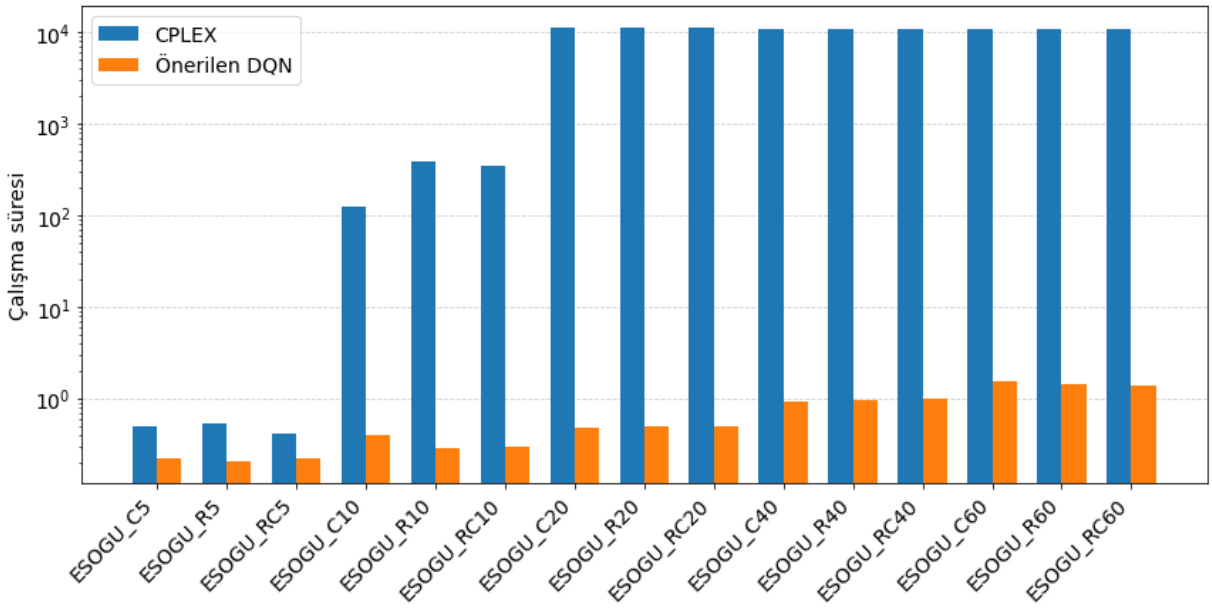
ESOGU_C10	5646	124.33	5886	0.3920	0.042%
ESOGU_R10	4889	389.74	4898	0.2814	0.001%
ESOGU_RC10	5843	345.63	5853	0.2987	0.001%
ESOGU_C20	9125	11160.69	10556	0.4845	0.15%
ESOGU_R20	9974	11134.66	10794	0.5021	0.08%
ESOGU_RC20	9161	11113.25	9950	0.5011	0.08%
ESOGU_C40	16102	10800.05	17929	0.9257	0.11%
ESOGU_R40	18276	10800.02	21773	0.9750	0.18%
ESOGU_RC40	17053	10800.02	19686	1.0042	0.15%
ESOGU_C60	26733	10800.02	29500	1.5297	0.10%
ESOGU_R60	26159	10800.03	30191	1.4217	0.15%
ESOGU_RC60	27905	10800.02	30660	1.3676	0.09%

Şekil 4.9'da önerilen modelin ve matematiksel modelin toplam mesafe bakımından karşılaştırılması görselleştirilmiştir. Şekil 4.9'da gösterildiği gibi küçük boyutlu test problemlerinde optimal çözümlerle neredeyse aynı sonuçları verirken büyük boyutlu test problemlerde de oldukça yakın sonuçlar vermiştir.



Şekil 4.9. CEVRP için CPLEX çözücü ve DQN sonuçlarının toplam mesafe açısından karşılaştırılması

Önerilen modelle matematiksel modelin yürütme süresi bakımından karşılaştırılması ise Şekil 4.10'da görselleştirilmiştir. Karşılaştırmalara bakıldığında, önerilen model yürütme süresini önemli ölçüde azaltmıştır. Büyük ölçekli problemler, matematiksel modelin gerektirdiği 3 saatin aksine, yaklaşık 1,5 saniyede çözülebilir. Yürütme süresi açısından, önerilen model matematiksel modele kıyasla önemli ölçüde daha hızlı hesaplama süreleri sergiler.



Şekil 4.10. CEVRP için CPLEX çözücü ve DQN sonuçlarının toplam çalışma süresi açısından karşılaştırılması

Çizelge 4.9, veri setindeki 15 örnek için zaman penceresi kısıtlamasının olduğu CEVRPTW problemi için önerilen model tarafından elde edilen sonuçları göstermektedir. Çözülen örnekler için yürütme süresi, minimum mesafe ve optimal rota bilgileri yer almaktadır.

Çizelge 4.9. CEVRPTW'nin tam şarj stratejisine göre önerilen DQN modelinden elde edilen sonuçları

Örnek Adı	**Mesafe (m)**	**Yürütme Süresi (s)**	**Rotalar**
ESOGU_C5_TW1	5006	0.22	R1:D-2-115-D R2:D-42B-32-35-D
ESOGU_R5_TW1	5727	0.23	R1:D-19-32-D R2:D-31-D R3:D-75-60A/2-D
ESOGU_RC5_TW1	7094	0.22	R1:D-32-D R2:D-75-D

			R3:D-115-D R4:D-42B-31-D
ESOGU_C10_TW1	7869	0.33	R1:D-24-22A-D R2:D-119-45-31-D R3:D-34-26-113-13-14-cs8-D
ESOGU_R10_TW1	7095	0.29	R1:D-56-122-74-77-70-D R2:D-21-22E-22A-D R3:D-78-60A/1-D
ESOGU_RC10_TW1	7350	0.30	R1:D-56-122-74-D R2:D-24-45-22A-D R3:D-21-22E-13-14-D
ESOGU_C20_TW1	14504	0.51	R1:D-49-51-50-37B-30-31-39-D R2:D-44-58/2-58/1-D R3:D-42B-153-43C-D R4:D-2-8-27-D R5:D-9-26-cs7-115-D R6:D-116-D
ESOGU_R20_TW1	15439	0.51	R1:D-152-45-22D-35-D R2:D-119-115-D R3:D-6-7-116-10-cs8-D R4:D-154-64-74-62-55-D R5:D-44-82-61B-43C-D R6:D-60A/2-D
ESOGU_RC20_TW1	13942	0.50	R1:D-51-50-45-22D-35-D R2:D-44-64-58/2-58/1-39-D R3:D-153-43C-D R4:D-42B-152-D R5:D-82-61B-62-D R6:D-2-10-115-cs8-D
ESOGU_C40_TW1	23552	0.91	R1:D-42C-49-152-37A-35-29-D R2:D-25-3-116-27-28-30-39-D R3:D-42B-38-51-52-50-D R4:D-44-58/2-60A/2-55-58/1-41-D R5:D-57-120-17-117-16-15-cs8-D R6:D-22B-9-8-22C-cs8-115-D R7:D-2-5-113-1-D R8:D-18-D
ESOGU_R40_TW1	27236	0.95	R1:D-154-44-64-122-61B-43B-D R2:D-42C-42A-153-119-1-D R3:D-120-24-8-113-125-cs8-10-13-D R4:D-57-74-124-73-60B/1-D R5:D-38-28-27-20-22C-D R6:D-41-46-22A-53-D R7:D-19-21-17-11-12-D R8:D-76-88-cs1-D R9:D-23-D
ESOGU_RC40_TW1	25064	0.99	R1:D-38-42B-49-50-120-35-39-D R2:D-42C-64-61B-58/2-58/1-60B/1-55-D R3:D-51-153-152-28-119-113-cs8-D R4:D-24-3-25-1-10-15-cs8-D R5:D-57-124-D R6:D-21-19-17-20-22A-D R7:D-2-8-11-13-125-cs8-115-D R8:D-76-D
ESOGU_C60_TW1	34416	1.53	R1:D-42C-49-47-28-30-31-53-D R2:D-154-38-36-37B-42A-39-D R3:D-40-7-24-32-22D-D R4:D-34-3-116-25-22A-D R5:D-61A/2-65-61B-70-62-69-D

			R6:D-56-71-66-68-60A/2-59-D R7:D-52-63-60B/2-43C-D R8:D-119-1-26-35-5-113-cs8-D R9:D-22C-21-18-16-D R10:D-118-19-13-14-10-D R11:D-23-12-9-4-cs8-D R12:D-115-2-D
ESOGU_R60_TW1	36463	1.42	R1:D-44-56-65-122-81-59-D R2:D-154-40-36-37B-37A-45-29-D R3:D-38-28-116-27-30-32-D R4:D-42B-43A-153-52-50-D R5:D-57-84-75-79-67-62-68-D R6:D-25-2-8-12-16-D R7:D-33-19-17-20-22D-D R8:D-41-114-5-113-4-cs8-D R9:D-53-43C-60A/2-60B/1-72-77-cs3-D R10:D-63-73-74-123-87-cs3-D R11:D-78-88-cs1-D R12:D-23-D
ESOGU_RC60_TW1	36398	1.38	R1:D-40-38-116-28-30-31-53-D R2:D-57-61A/2-122-61B-81-79-77-D R3:D-154-43A-60A/2-62-73-59-D R4:D-25-27-119-33-36-37B-D R5:D-42B-52-50-45-29-D R6:D-153-60B/2-68-123-D R7:D-34-24-22D-22A-22C-14-cs8-D R8:D-56-65-43C-D R9:D-18-19-21-16-13-cs8-D R10:D-114-113-5-8-10-D R11:D-63-88-cs1-D R12:D-78-D R13:D-23-4-26-cs7-D

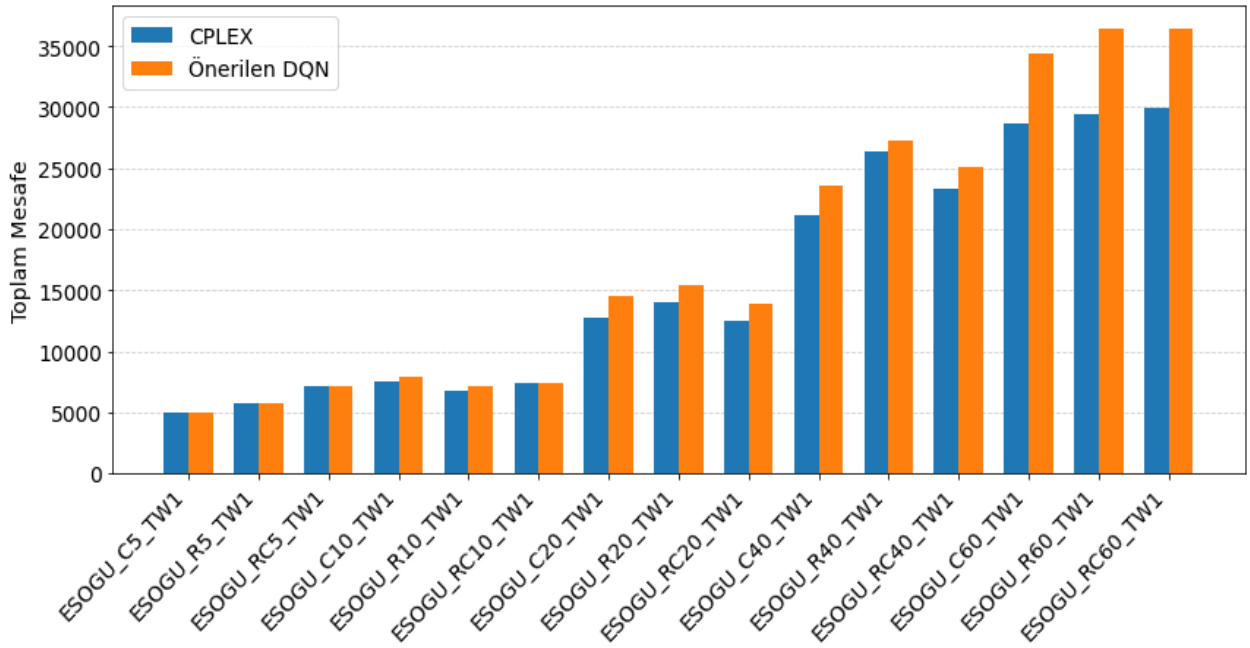
CEVRPTW problemi için önerilen modelin ve matematiksel modelin karşılaştırmalı sonuçlarını Çizelge 4.10'da sunulmaktadır. Çizelge 4.10'a göre, önerilen model beş müşterili test örnekler için optimal çözümü, on müşterili örnekler için optimale yakın çözümü elde eder ve diğer örnekler için ise en iyiye yaklaşır.

Çizelge 4.10. CEVRPTW için önerilen DQN modelinin CPLEX çözücüden elde edilen sonuçlarla karşılaştırılması

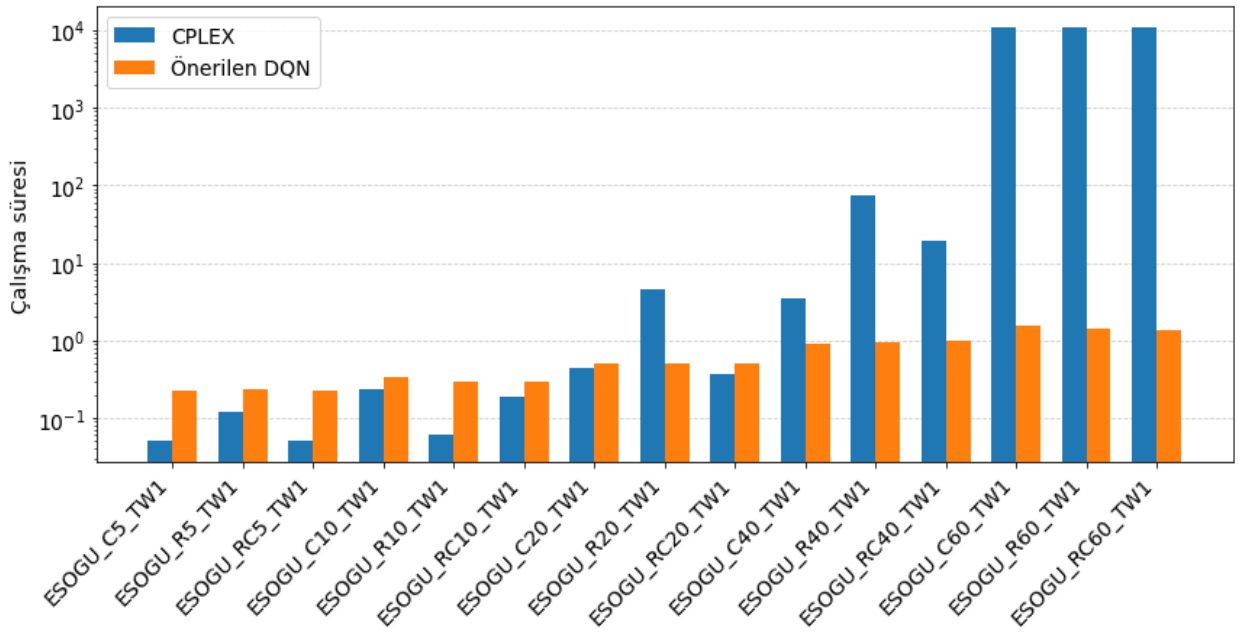
Örnek Adı	**CPLEX Çözücü**		**Önerilen DQN**		**GAP**
	Mesafe (m)	**Yürütme Süresi (s)**	**Mesafe (m)**	**Yürütme Süresi(s)**	
ESOGU_C5_TW1	5006.00	0.05	5006	0.22	0.0%
ESOGU_R5_TW1	5727.0	0.12	5727	0.23	0.0%
ESOGU_RC5_TW1	7094.00	0.05	7094	0.22	0.0%

ESOGU_C10_TW1	7531.00	0.23	7869	0.33	0.04%
ESOGU_R10_TW1	6822.0	0.06	7095	0.29	0.04%
ESOGU_RC10_TW1	7350.00	0.19	7350	0.30	0.0%
ESOGU_C20_TW1	12737.00	0.44	14504	0.51	0.13%
ESOGU_R20_TW1	14022.00	4.51	15439	0.51	0.10%
ESOGU_RC20_TW1	12447.00	0.36	13942	0.50	0.12%
ESOGU_C40_TW1	21151.00	3.41	23552	0.91	0.11%
ESOGU_R40_TW1	26317.00	75.09	27236	0.95	0.03%
ESOGU_RC40_TW1	23375.00	19.31	25064	0.99	0.07%
ESOGU_C60_TW1	28714.00	10800.50	34416	1.53	0.19%
ESOGU_R60_TW1	29392.00	10801.17	36463	1.42	0.24%
ESOGU_RC60_TW1	29997.00	10801.78	36398	1.38	0.21%

Şekil 4.11'de matematiksel modelin ve önerilen modelin toplam mesafe bakımından karşılaştırmalı sonuçları görselleştirilmiştir. Şekil 4.12 ise bu iki yaklaşımın yürütme sürelerini karşılaştırması yer alır. Büyük ölçekli problemler için, matematiksel modelin bir çözüm bulması üç saate kadar sürebilirken, önerilen model bir veya iki saniye kadar kısa bir sürede bir çözüm üretebilir.



Şekil 4.11. CEVRPTW için CPLEX çözücü ve DQN sonuçlarının toplam mesafe açısından karşılaştırılması



Şekil 4.12. CEVRPTW için CPLEX çözücü ve DQN sonuçlarının toplam çalışma süresi açısından karşılaştırılması

Yapılan testler incelendiğinde matematiksel modelin en iyi çözümleri sağlarken, önemli miktarda hesaplama kaynağı ve işlem süresi gerektirdiği görülmüştür. Buna karşılık, RL yöntemleri neredeyse anında çözümler üreterek, zamana duyarlı lojistik operasyonlarında pratik önemlerini vurgular. Karşılaştırmalı analiz, önerilen DQN modelinin çok kısa bir yürütme süresiyle neredeyse en iyi çözümlere ulaştığını göstermektedir. Bu özellikler, DQN'nin gerçek dünya uygulamalarında hem çözüm kalitesini hem de hesaplama verimliliğini dengeleyerek CEVRPTW'yi etkili bir şekilde ele alma potansiyelinin altını çizmektedir. Ayrıca, DQN belirli bir veri kümesi üzerinde eğitildiği için, ek eğitim gerektirmeden aynı dağılımdan üretilen yeni örnekler için güçlü sonuçlar sunabilir ve bu da pratikliğini artırır. Örneğin model 20 müşteri ile eğitildiğinden sonra, farklı 20 müşteri içeren örnekler üzerinde test edilebilir. Dahası farklı müşteri sayıları içeren örnekler için de sonuç üretebilir.

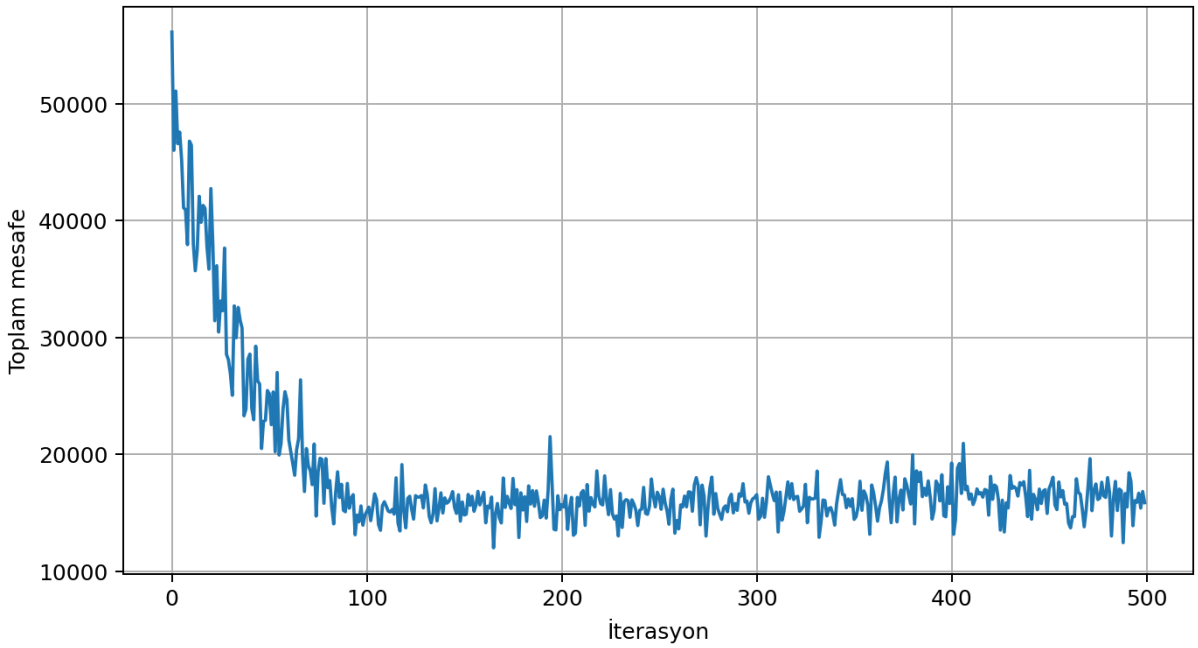
4.4. EVRP-PDP-TW için DQN Modelinin Sonuçları

Önerilen DQN yöntemi ESOGU-EVRP-PDP-TW veri setindeki orta genişlikte zaman pencerelerine sahip 18 örnekte test edilmiştir. EV depodan harekete başlayarak batarya kapasitesinin izin verdiği ölçüde müşterilerin toplama ya da dağıtım taleplerini

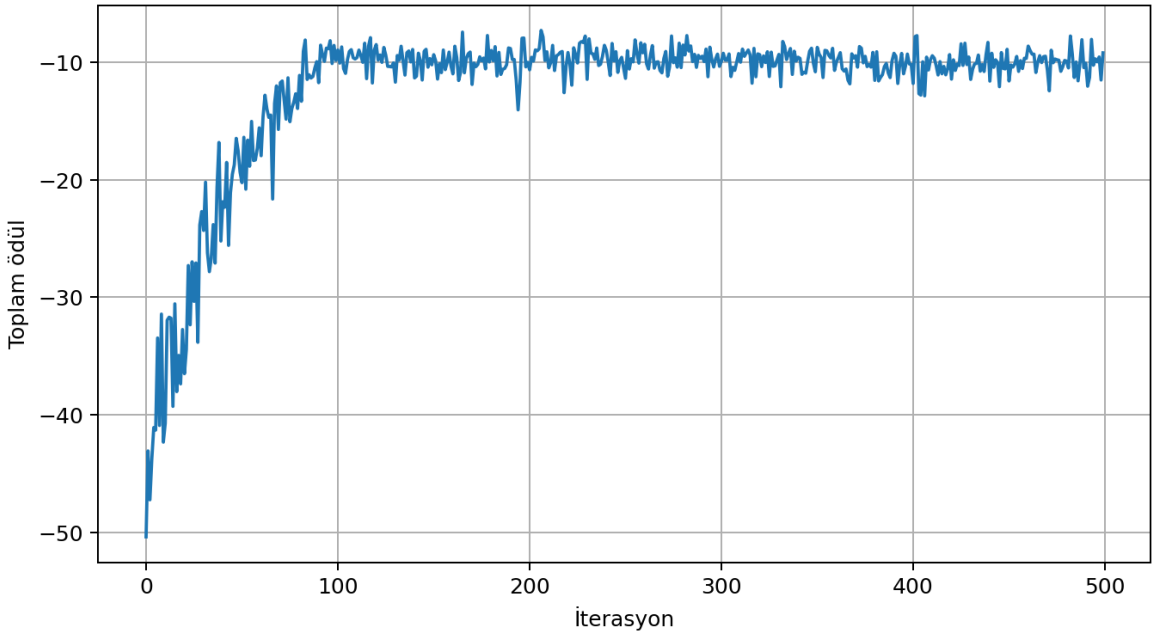
karşılar. Mevcut batarya kapasitesi yetersiz olduğunda, EV tam şarj için en uygun şarj istasyonuna yönlendirilir. Ayrıca her müşteriye ait bir zaman penceresi olduğundan müşteriye hizmete başlangıç zamanı bu belirlenen zaman penceresi aralığına uymalıdır. EV müşterinin servise en geç başlama zamanından sonra müşteriye gidemez. Ya da EV müşteriye geldiğinde servise en erken başlama zamanında önce gelmişse bekleme durumuna geçer. Amaç, müşteri talebi, zaman penceresi ve EV'nin şarj durumu ile ilgili gerekli tüm kısıtlamalara uyarak toplam seyahat mesafesini minimize eden rotayı elde etmektir.

EVRP-PDP-TW için yapılan tüm testler 3,40 GHz'de Intel i7-13700K ve bir NVDIA Geforce RTX 4070 GPU'ya sahip masaüstü bilgisayarda gerçekleştirilmiştir. Kod python'da tensorflow kütüphanesi kullanılarak yazılmıştır. Önerilen modelin eğitim aşaması sırasında, her sayıdaki problem için (5,10,20,40,60,80 müşteri) 1.000 örnek üretilmiştir. Bu örnekler ESOGU kampüs üzerindeki noktalardan rastgele seçilerek oluşturulur. Bu örnekler üzerinde eğitim gerçekleşir. Test için ise ESOGU-EVRP-PDP-TW veri setindeki örnekler kullanılır. Örneğin 40 müşteri içeren örnekler için yapılan eğitimde, her iterasyon yaklaşık 7 dakika sürmüş ve 100 iterasyon yaklaşık 12 saat sürmüştür.

Şekil 4.13'te önerilen DQN algoritmasının EVRP-PDP-TW problemindeki öğrenme eğrisi sunulmaktadır. Şekilde 4.13'te, 40 müşterili eğitim örnekleri için toplam seyahat mesafesinin 500 iterasyon boyunca nasıl değiştiği gösterilmektedir. Eğitim sürecinin başlarında, DQN'deki ajan genellikle keşif eylemlerini seçtiğinden dolayı toplam seyahat mesafesi değerleri yüksektir. Ancak eğitim ilerledikçe, daha verimli rotalar üretmek için politikasını iyileştirdiğinden grafikte belirgin bir düşüş eğilimi gözlemlenir. Grafiğe bakıldığında eğitimdeki dalgalanmalar görülür. Bu durum DQN'nin keşif-sömürü dengesi gereği alternatif eylemleri denemesinden kaynaklıdır. Sonuç olarak, grafikteki tutarlı azalma, önerilen DQN'nin EVRP-PDP-TW için neredeyse optimal çözümlere doğru yakınsadığını gösterir.



Şekil 4.13. Önerilen DQN yaklaşımının EVRP-PDP-TW probleminde eğitim süresince toplam mesafe eğrisi



Şekil 4.14. Önerilen DQN yaklaşımının EVRP-PDP-TW probleminde eğitim süresince toplam ödül eğrisi

Önerilen DQN algoritmasının EVRP-PDP-TW problemindeki toplam ödül eğrisini Şekil 4.14'te gösterilmektedir. Toplam mesafe eğrisinin (Şekil 4.13) iterasyon ilerledikçe

azalan seyrinin aksine, ödül eğrisi eğitim boyunca tutarlı bir yükseliş sergilemektedir. Bu artış, daha yüksek ödüllerin genellikle daha iyi çözümlere karşılık gelmesi nedeniyle, ajanın daha verimli rotalama kararlarını belirleme ve uygulama becerisinin arttığını yansıtmaktadır. Ödül değerlerinin üst seviyelerde yataya yakınsaması ise DQN algoritmasının yakınsama eğilimini ve EVRP-PDP-TW için neredeyse optimal çözümler keşfetme yetkinliğini ortaya koymaktadır.

DQN algoritması ESOGU_EVRP-PDP-TW veri setinden seçilen orta zaman penceresine sahip 18 örnek üzerinde test edilmiştir ve sonuçlar Çizelge 4.11'de sunulmuştur. Çizelge 4.11, veri setindeki bu örnekler için EVRP-PDP-TW için önerilen DQN modelinin sonuçlarını göstermektedir. Çözülen örnekler için yürütme süresi, minimum mesafe ve elde edilen rotalar gösterilmiştir.

Çizelge 4.11. EVRP-PDP-TW'nin tam şarj stratejisine göre önerilen DQN modelinden elde edilen sonuçları

Örnek Adı	**Mesafe(m)**	**Yürütme Süresi (s)**	**Rotalar**
ESOGU_C5_TW1	3705.48	0.18	**R1:**D-154-cs5-22A-10-13-14-D
ESOGU_R5_TW1	4123.49	0.19	**R1:**D-25-7-42C-43A-74-cs3-D
ESOGU_RC5_TW1	4235.62	0.18	**R1:**D-43A-74-cs3-10-13-14-D
ESOGU_C10_TW1	3563.61	0.25	**R1:**D-33-36-37B-37A-40-41-42A-42C-cs5-11-12-D
ESOGU_R10_TW1	5758.59	0.26	**R1:**D-6-14-9-45-37A-cs5-58/2-61A/2-121-46-33-D
ESOGU_RC10_TW1	5184.74	0.26	**R1:**D-33-6-11-12-9-45-cs6-36-37B-37A-58/2-D
ESOGU_C20_TW1	4720.56	0.43	**R1:**D-17-22B-22E-22C-21-20-117-16-18-D **R2:**D-60A/2-60A/1-60B/2-60B/1-121-122-61A/1-61A/2-61B-56-52-D
ESOGU_R20_TW1	8242.01	0.47	**R1:**D-31-39-43A-61A/1-61B-65-73-72-67-80-76-cs3-60A/1-54-37A-38-34-cs7-9-23-20-22C-D
ESOGU_RC20_TW1	7204.17	0.45	**R1:**D-17-20-22E-21-22B-117-16-23-18-D **R2:**D-38-31-43A-60A/1-121-72-73-80-76-cs3-122-54-D
ESOGU_C40_TW1	8225.08	0.77	**R1:**D-51-52-42C-43A-43B-44-55-122-61A/1-61A/2-61B-56-60A/2-60B/2-60B/1-121-60A/1-cs9-42A-49-47-46-36-37B-37A-41-7-6-33-D **R2:**D-20-17-22B-22C-21-22E-18-117-16-11-12-8-D

ESOGU_R40_TW1	12630.9	0.90	**R1:**D-120-36-46-33-35-29-38-39-40-42B-cs5-49-43A-57-122-61A/2-64-70-82-74-79-75-124-73-68-66-cs3-84-52-153-43C-D **R2:**D-20-22B-22C-19-22D-114-5-125-2-cs8-12-9-D
ESOGU_RC40_TW1	10288.33	0.84	**R1:**D-120-20-22B-22E-22C-21-17-117-16-12-5-2-cs8-18-19-33-36-46-35-29-38-D **R2:**D-51-52-56-61A/1-61A/2-61B-122-84-82-74-79-66-68-cs2-60A/1-60B/2-60B/1-121-60A/2-43C-49-D
ESOGU_C60_TW1	13856.93	1.11	**R1:**D-45-35-119-32-25-116-115-1-114-113-125-4-2-cs8-14-13-10-16-17-20-22E-22C-21-22B-18-117-23-D **R2:**D-52-80-81-74-78-123-82-76-77-75-79-124-73-121-60B/1-60B/2-60A/1-60A/2-59-cs9-61A/1-61A/2-61B-122-56-43C-42B-39-53-154-54-cs5-D **R3:**D-22D-22A-24-3-D
ESOGU_R60_TW1	15773.92	1.17	**R1:**D-42A-53-42B-40-41-38-30-29-35-33-D **R2:**D-17-22E-21-118-19-16-14-13-12-23-8-cs8-4-5-113-1-26-3-7-24-22D-6-D **R3:**D-51-72-66-62-121-60B/1-60A/1-61A/2-61A/1-122-64-65-71-85-86-83-82-80-75-79-cs3-77-124-87-56-58/1-55-59-cs9-43C-52-D
ESOGU_RC60_TW1	16575.14	1.17	**R1:**D-45-33-35-29-30-38-5-113-125-8-10-14-13-cs8-4-2-115-1-16-23-22E-21-D **R2:**D-51-78-80-81-74-75-79-76-77-124-87-123-82-85-cs3-72-73-121-60B/1-60A/1-59-55-58/1-61A/2-61A/1-64-71-56-43C-cs5-53-154-54-41-6-D **R3:**D-118-24-22D-22A-25-3-D
ESOGU_C80_TW1	18767.78	1.52	**R1:**D-47-46-36-37B-37A-42A-42C-43A-43B-44-122-61A/1-61A/2-61B-64-65-70-71-80-81-74-78-123-82-73-72-69-68-66-67-63-62-60B/1-60A/1-cs9-D **R2:**D-40-41-39-35-32-25-7-6-33-D **R3:**D-20-17-22B-22E-22C-21-18-118-19-117-16-11-12-9-8-cs8-4-2-125-1-115-3-D **R4:**D-51-52-79-75-76-77-124-88-cs1-55-56-60A/2-60B/2-121-cs9-43C-42B-49-D
ESOGU_R80_TW1	28114.02	1.55	**R1:**D-120-50-51-52-42C-43B-57-122-61A/1-61A/2-61B-56-70-71-68-74-75-79-78-80-cs3-123-73-66-67-62-121-60B/1-60B/2-60A/2-59-76-77-124-87-cs2-86-42A-D **R2:**D-47-36-37B-41-39-49-153-42B-29-cs4-28-35-34-116-27-33-7-6-22D-cs7-1-5-113-125-8-10-15-14-13-12-9-20-23-cs8-D **R3:**D-17-22B-22E-18-117-16-11-2-cs8-26-154-D **R4:**D-43C-83-cs3-88-cs1-D

ESOGU_RC80_TW1	21024.26	1.61	**R1:**D-120-50-51-52-42C-43B-57-61A/1-56-122-70-71-73-60B/2-60B/1-121-60A/2-67-68-cs3-74-78-80-81-123-82-66-59-75-79-76-77-124-87-cs3-42A-D **R2:**D-47-36-37B-41-39-33-28-35-34-32-25-116-cs7-1-5-113-125-8-10-15-16-20-17-22B-22E-22C-21-18-6-7-D **R3:**D-153-42B-43C-154-D **R4:**D-117-14-13-12-11-9-23-4-2-cs8-115-26-27-3-D

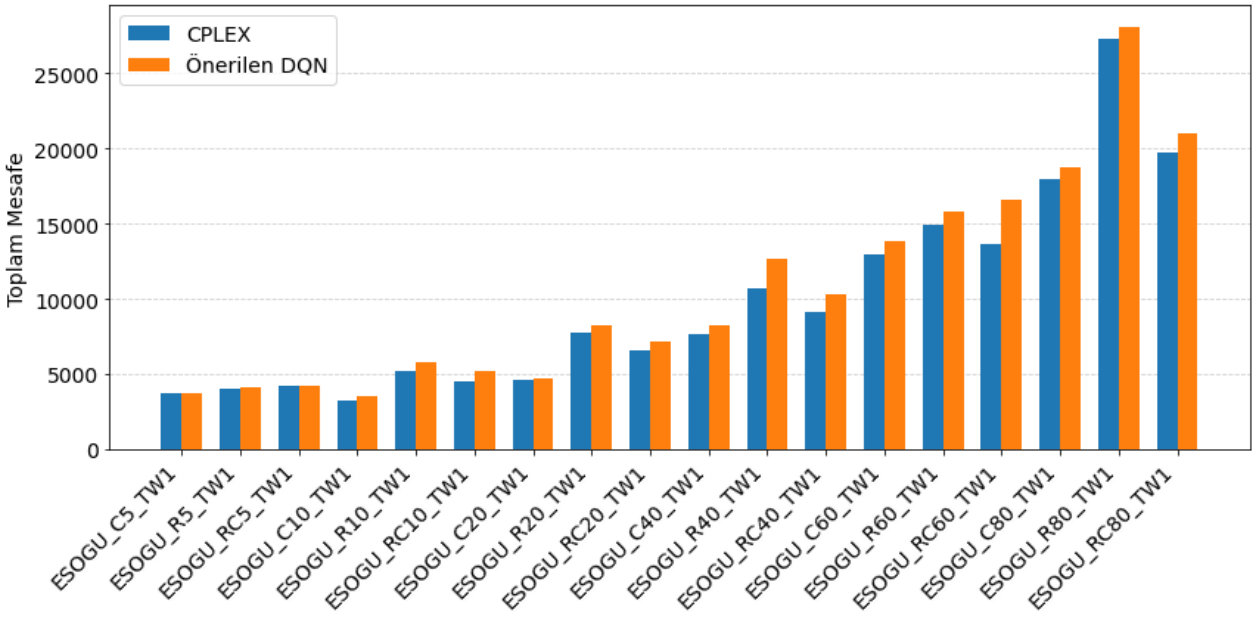
Önerilen DQN algoritmasının EVRP-PDP-TW için sonuçları matematiksel modelle elde edilen optimal sonuçlarla karşılaştırılmıştır ve sonuçlar Çizelge 4.12'de gösterilmektedir. Çizelge 4.12'deki GAP değeri Denklem 4.1 kullanılarak hesaplanmıştır.

Çizelge 4.12. EVRP-PDP-TW için önerilen DQN modelinin CPLEX çözücüden elde edilen sonuçlarla karşılaştırılması

Örnek Adı	**CPLEX Çözücü**		**Önerilen DQN**		**GAP**
	Mesafe (m)	**Yürütme Süresi (s)**	**Mesafe (m)**	**Yürütme Süresi(s)**	
ESOGU_C5_TW1	3705.48	0.64	3705.48	0.18	0.0%
ESOGU_R5_TW1	4013.27	0.36	4123.49	0.19	0.02%
ESOGU_RC5_TW1	4235.62	0.75	4235.62	0.18	0.0%
ESOGU_C10_TW1	3255.85	0.61	3563.61	0.25	0.09%
ESOGU_R10_TW1	5203.27	2.69	5758.59	0.26	0.10%
ESOGU_RC10_TW1	4566.68	1.97	5184.74	0.26	0.13%
ESOGU_C20_TW1	4590.68	10800.02	4720.56	0.43	0.02%
ESOGU_R20_TW1	7797.43	7814.91	8242.01	0.47	0.05%
ESOGU_RC20_TW1	6542.60	10800.01	7204.17	0.45	0.10%
ESOGU_C40_TW1	7694.95	10800.02	8225.08	0.77	0.06%
ESOGU_R40_TW1	10724.85	10800.02	12630.9	0.90	0.17%
ESOGU_RC40_TW1	9091.72	10800.02	10288.33	0.84	0.13%
ESOGU_C60_TW1	12935.75	10800.02	13856.93	1.11	0.07%
ESOGU_R60_TW1	14922.99	10800.02	15773.92	1.17	0.05%

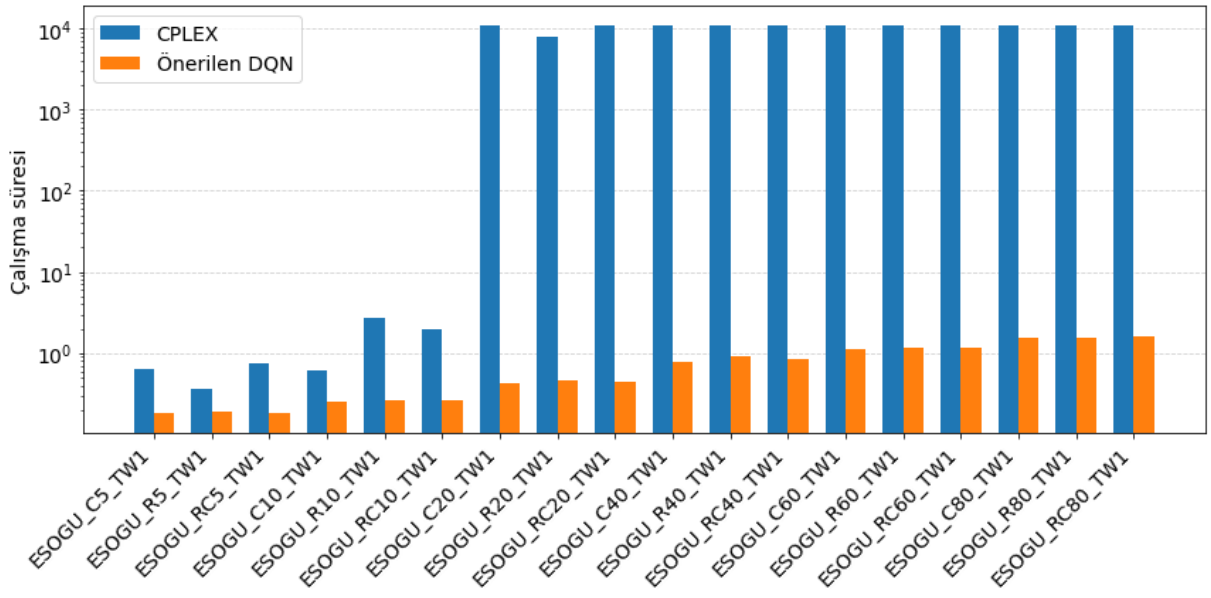
ESOGU_RC60_TW1	13613.28	10800.01	16575.14	1.17	0.21%
ESOGU_C80_TW1	18013.20	10800.03	18767.78	1.52	0.04%
ESOGU_R80_TW1	27345.64	10800.02	28114.02	1.55	0.02%
ESOGU_RC80_TW1	19748.42	10800.01	21024.26	1.61	0.06%

Şekil 4.15'te önerilen DQN modelinin ve matematiksel modelin toplam mesafe bakımından karşılaştırılması görselleştirilmiştir. Şekil 4.15'te gösterildiği gibi küçük boyutlu test problemlerinde optimal çözümlerle neredeyse aynı sonuçları verirken büyük boyutlu test problemlerde de çok yakın sonuçlar vermiştir.



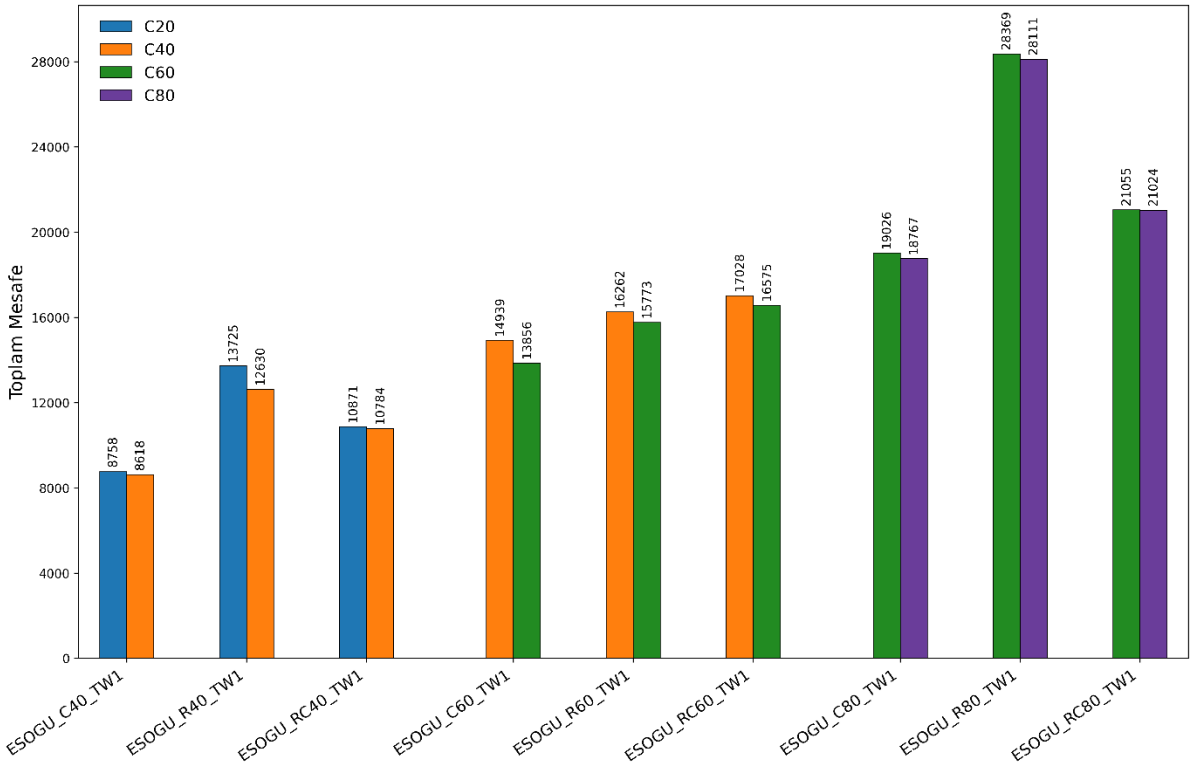
Şekil 4.15. EVRP-PDP-TW için CPLEX çözücü ve DQN sonuçlarının toplam mesafe açısından karşılaştırılması

Önerilen modelle matematiksel modelin yürütme süresi bakımından karşılaştırılması ise Şekil 4.16'da gösterilmiştir. Karşılaştırmalara bakıldığında, önerilen model yürütme süresini önemli ölçüde azaltmıştır. Büyük ölçekli problemler için, matematiksel modelin üç saati bulan çözümlerinin aksine önerilen yaklaşım çözümleri 1, 2 saniyede üretebilir. Dolayısıyla yürütme süresi açısından önerilen model, matematiksel modele göre çarpıcı bir hız avantajı sağlamaktadır.

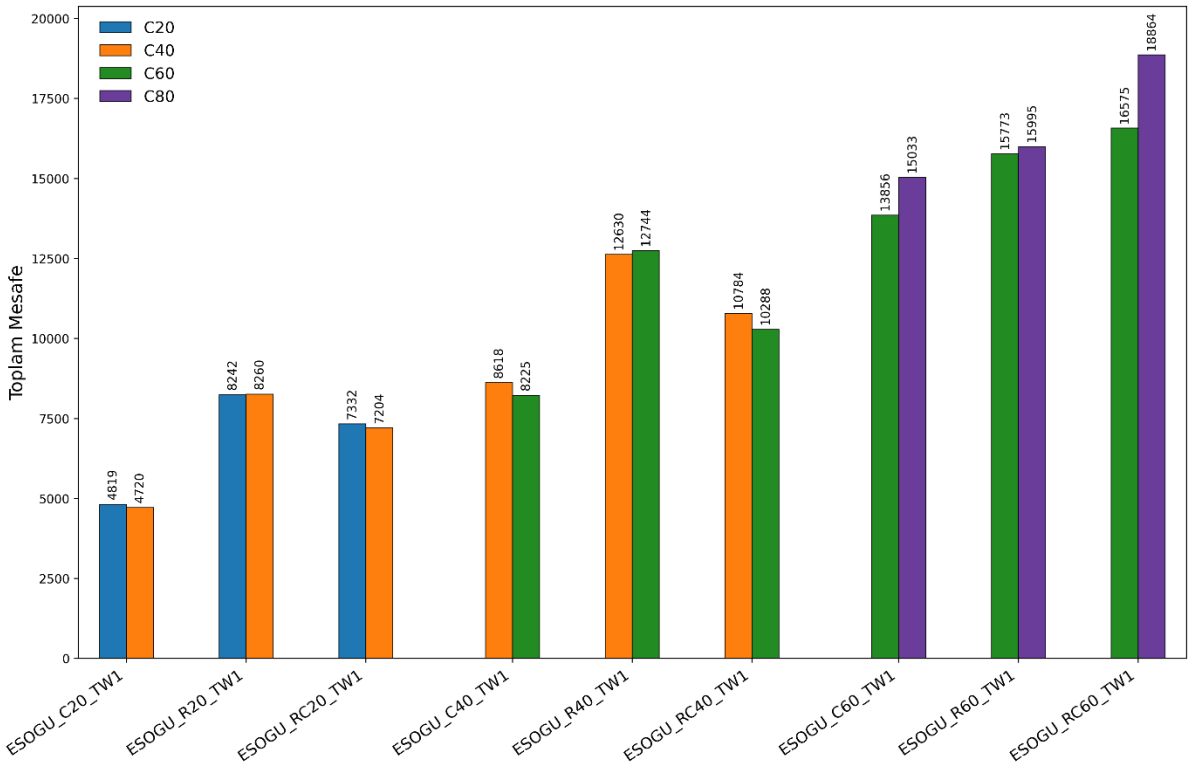


Şekil 4.16. EVRP-PDP-TW için CPLEX çözücü ve DQN sonuçlarının çalışma süresi açısından karşılaştırılması

Yapılan karşılaştırmalardan sonra eğitilen DQN modelinin farklı ölçeklerdeki problemlere genelleme yeteneği ayrıntılı olarak incelenmiştir. Genelleme yeteneğini test etmek için önerilen model dört farklı müşteri büyüklüğü (20, 40, 60 ve 80 müşteri) için ayrı ayrı eğitilmiş ve iki senaryo ele alınmıştır. İlk senaryoda, daha küçük problem ölçeğinde eğitilen model, daha büyük komşu problemlerin çözümünde kullanılmıştır. Örneğin ESOGU_C40_TW1, ESOGU_R40_TW1, ESOGU_RC40_TW1 problem örnekleri 20 ve 40 müşteri için eğitilen modelle çözülmüştür ve sonuçlar Şekil 4.17'de gösterilmektedir. İkinci senaryoda ise ters yönde bir değerlendirme yapılarak, büyük ölçekteki problemler için eğitilen model, daha küçük komşu problemlerde test edilmiştir. Örneğin ESOGU_C20_TW1, ESOGU_R20_TW1, ESOGU_RC20_TW1 problem örnekleri 20 ve 40 müşteri için eğitilen modelle çözülmüştür ve sonuçlar Şekil 4.18'de sunulmuştur. Şekil 4.17 için sonuçlar, her bir modelin kendi eğitim ölçeğinde en iyi performansı sergilediğini, ancak küçük ölçeklerde eğitilen modellerin daha büyük ölçekli problemlerde de rekabetçi çözümler üretebildiğini göstermektedir. Şekil 4.18'de ise büyük ölçekli modeller de küçük ölçekli problemlere başarıyla genellenmiş ve ölçekler arasındaki benzerlik arttıkça performans farkı azalmıştır. Bu bulgular, önerilen DQN modelinin farklı problem ölçekleri arasında güçlü bir genelleme potansiyeline sahip olduğunu göstermektedir.



Şekil 4.17. Küçük ölçekli modelin büyük problem örneklerine genelleme performansı



Şekil 4.18. Büyük ölçekli modelin küçük problem örneklerine genelleme performansı

Önerilen DQN yöntemi bildiğimiz kadarıyla literatürde EVRP-PDP problemlerine uygulanan ilk pekiştirmeli öğrenme yaklaşımıdır. Literatürdeki çalışmalarda pekiştirmeli öğrenme yaklaşımları özellikle VRP, EVRP ve benzeri rotalama problemlerinde yaygın şekilde kullanılmış olmasına rağmen, EVRP-PDP ve varyantları için henüz ele alınmamıştır. Bu boşluğu dolduran çalışmamızda, DQN algoritmasını EVRP-PDP-TW'nin zaman penceresi ve batarya kısıtlarını gözeterek özelleştirip iyileştirdik. Elde edilen sonuçlar hem çözüm kalitesi hem de hesaplama hızı açılarından üstün performans sergileyerek, DQN'nin bu problem sınıfı için uygulanabilirliğini ve sağladığı pratik avantajları ilk kez kanıtlamıştır.

EVRP ile ilgili yazılımlar, lojistik ve dağıtım sektörlerinde günlük operasyonel kararları desteklemede çok önemlidir. Bu kararlar rota planlama, araç atama, şarj istasyonu seçimi ve gerçek zamanlı güncellemeleri kapsar. Genellikle, bu süreçler günün çeşitli saatlerinde başlatılır ve hızlı ve verimli bir şekilde tamamlanmalıdır. Gerçekçi bir karar alma süresi olarak, algoritmalar orta büyüklükteki bir son kilometre teslimat problemi için yaklaşık 2 dakika içinde sonuç üretmelidir (D'Emidio vd., 2024). RL yöntemleri, geleneksel optimizasyon yöntemlerine kıyasla çözümlerin daha hızlı üretilmesini sağlayarak EVRP yazılımının etkinliğini artırır. Hızlı çözüm üretme yeteneği, özellikle lojistik operasyonlarının dinamik yapısı göz önüne alındığında, RL algoritmalarının pratik uygulanabilirliğini önemli ölçüde artırır. RL yaklaşımlarının hızlı ve uyarlanabilir çözümler üretme yeteneği, zamanında ve esnek karar almanın verimliliği korumak ve hizmet standartlarını karşılamak için gerekli olduğu lojistiğin operasyonel gereksinimleriyle iyi uyumludur.

5.SONUÇ VE ÖNERİLER

Günümüzde, fosil yakıtlı araçlardan elektrikli araçlara (EV) geçişin hızlanmasıyla birlikte, literatürdeki araştırmalar Elektrikli Araç Rotalama Problemlerini (EVRP) çözmeye odaklanmıştır. Bu çalışma, EVRP problemlerinden biri olan kapasiteli elektrikli araç rotalama problemi ve topla dağıt elektrikli araç rotalama problemini ele almaktadır. Bu problemlerin çözümünde son yıllarda oldukça popüler olan pekiştirmeli öğrenme yöntemleri kullanılmaktadır. Kapasiteli elektrikli araç rotalama problemi (CEVRP) için yeni bir Q-öğrenme algoritması, zaman pencereleri kapasiteli elektrikli araç rotalama problemi (CEVRPTW) ve zaman pencereleri topla dağıt elektrikli araç rotalama problemi için ise yeni bir DQN yöntemi önerilmektedir. Her ne kadar VRP ve EVRP gibi problemlere yönelik olarak daha önce pekiştirmeli öğrenme yöntemleri uygulanmış olsa da, literatürde EVRP-PDP-TW problemine yönelik olarak pekiştirmeli öğrenme yöntemlerinin kullanıldığı ilk çalışma bu tez çalışmadır.

Gerçek dünya coğrafi verilerine dayanan ESOGU-CEVRPTW ve ESOGU-EVRP-PDP-TW veri setleri, gerçekçi bir test ortamı sağlamak için oluşturulmuştur ve bu veri setlerinin matematiksel model ile optimal sonuçları elde edilmiştir. Önerilen yöntemler, bu veri setleri kullanılarak değerlendirilmiştir. Deneysel sonuçlar, önerilen Q-öğrenme ve DQN algoritmalarının elektrikli araçların sınırlı batarya kapasitesi ve uzun şarj süreleri gibi kısıtlarını etkin biçimde dikkate alarak optimal çözümlere çok yakın sonuçları kısa sürelerde elde ettiğini ortaya koymuştur. Bu bulgular, algoritmanın lojistik planlamasında, özellikle hızlı karar almanın kritik olduğu senaryolarda gerçek zamanlı uygulamalar için potansiyelini vurgulamaktadır. Önerilen yöntemlerin gerçek dünya verileri üzerinde test edilmesi, çözümlerin yalnızca teorik olarak sağlam değil, aynı zamanda pratik olarak da uygulanabilir olmasını sağlamaktadır.

Gelecekteki araştırmalar, dinamik trafik verileri, değişen enerji tüketim oranları veya seyahat ve şarj sürelerinin stokastik unsurları gibi değişkenleri göz önünde bulundurarak önerilen yöntemler daha da iyileştirebilir ve böylece uygulanabilirliğini daha karmaşık rotalama problemlerine genişletebilir. Ayrıca, farklı şehir yapıları ve müşteri dağılımlarına

sahip veri setlerinde algoritmaların performanslarının test edilmesi ve farklı derin öğrenme tabanlı mimarilerin (örneğin çizge sinir ağları) rotalama problemlerine uygulanması gelecek çalışmalar için önemli araştırma alanları olarak değerlendirilmektedir. Bunlara ek olarak son yıllarda Büyük Dil Modelleri (Large Language Models-LLM) VRP çözümlerine başarılı bir şekilde entegre edilmeye başlamıştır (Huang vd, 2024; Jiang vd., 2025; Li vd., 2025). Bu yaklaşımlarda problem tanımı, yapılandırılmış veri formatında LLM'e sunulur ve model bu tanımdan gerekli kısıtları ve parametreleri çıkararak ilgili kodu üretir. Bu sayede manuel kod yazma süresi ve hata riski büyük ölçüde azalır ve yeni kısıtlar söz konusu olduğunda ise yalnızca problem tanımı güncellenerek hızlıca uyarlama yapılabilir. Bu doğrultuda VRP'de başarıyla kullanılan LLM tabanlı yaklaşımlar, EVRP'ye de uyarlanabilir. Elektrikli araçlara özgü kısıtların problem tanımına eklenmesiyle otomatik kod üretimi gerçekleştirilebilir ve böylece geliştirme süreci hızlanarak esnek çözümler elde edilir.

KAYNAKLAR DİZİNİ

Agrali, C., & Dissertation, A. (2022). *PICKUP AND DELIVERY PROBLEM WITH TRANSFERS AND ELECTRIC VEHICLES*.

Agrali, C., & Lee, S. (2023). The multi-depot pickup and delivery problem with capacitated electric vehicles, transfers, and time windows. *Computers and Industrial Engineering*, *179*. https://doi.org/10.1016/j.cie.2023.109207

Akbay, M. A., Kalayci, C. B., Blum, C., & Polat, O. (2022). Variable Neighborhood Search for the Two-Echelon Electric Vehicle Routing Problem with Time Windows. *Applied Sciences (Switzerland)*, *12*(3). https://doi.org/10.3390/app12031014

Aljohani, T. M., Ebrahim, A., & Mohammed, O. (2021). Real-Time metadata-driven routing optimization for electric vehicle energy consumption minimization using deep reinforcement learning and Markov chain model. *Electric Power Systems Research*, *192*. https://doi.org/10.1016/j.epsr.2020.106962

Aslan Ö, Sarıçiçek İ, & Yazıcı A. (2025). ESOGU-CEVRPTW. *Mendeley Data*.

Aslan O, Yazıcı A, & Sarıçiçek İ. (2022). Elektrikli Araçlar İçin Rota Planlama Üzerine Bir Derleme. *2. Internatıonal Black Sea Modern Scıentıfic Research Congress*, 1–15.

Aslan Yıldız, Ö., Sarıçiçek, İ., & Yazıcı, A. (2025). A Reinforcement Learning-Based Solution for the Capacitated Electric Vehicle Routing Problem from the Last-Mile Delivery Perspective. *Applied Sciences (Switzerland)*, *15*(3). https://doi.org/10.3390/app15031068

Bac, U., & Erdem, M. (2021). Optimization of electric vehicle recharge schedule and routing problem with time windows and partial recharge: A comparative study for an urban logistics fleet. *Sustainable Cities and Society*, *70*. https://doi.org/10.1016/j.scs.2021.102883

Basso, R., Kulcsár, B., Sanchez-Diaz, I., & Qu, X. (2022). Dynamic stochastic electric vehicle routing with safe reinforcement learning. *Transportation Research Part E: Logistics and Transportation Review*, *157*. https://doi.org/10.1016/j.tre.2021.102496

Blakeman, S., & Mareschal, D. (2020). A complementary learning systems approach to temporal difference learning. *Neural Networks*, *122*, 218–230. https://doi.org/10.1016/j.neunet.2019.10.011

Cetiner, G., Yayan, U., & Yazici, A. (2024). Mutation Based White Box Testing of Deep Neural Networks. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3482114

D'Emidio, M., Delfaraz, E., Di Stefano, G., Frittella, G., & Vittoria, E. (2024). Route Planning Algorithms for Fleets of Connected Vehicles: State of the Art, Implementation, and Deployment. *Applied Sciences (Switzerland)*, *14*(7). https://doi.org/10.3390/app14072884

KAYNAKLAR DİZİNİ

Deng, J., Zhang, J., & Yang, S. (2024). Optimizing electric vehicle routing with nonlinear charging and time windows using improved differential evolution algorithm. *Cluster Computing*, *27*(4), 5423–5458. https://doi.org/10.1007/s10586-023-04243-z

Desaulniers, G., Errico, F., Irnich, S., & Schneider, M. (2016). Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, *64*(6), 1388–1405. https://doi.org/10.1287/opre.2016.1535

Doppstadt, C., Koberstein, A., & Vigo, D. (2020). The Hybrid Electric Vehicle—Traveling Salesman Problem with time windows. *European Journal of Operational Research*, *284*(2), 675–692. https://doi.org/10.1016/j.ejor.2019.12.031

Dorokhova, M., Ballif, C., & Wyrsch, N. (2021). Routing of Electric Vehicles With Intermediary Charging Stations: A Reinforcement Learning Approach. *Frontiers in Big Data*, *4*. https://doi.org/10.3389/fdata.2021.586481

Erdelić, T., & Carić, T. (2022). Goods Delivery with Electric Vehicles: Electric Vehicle Routing Optimization with Time Windows and Partial or Full Recharge. *Energies*, *15*(1). https://doi.org/10.3390/en15010285

Erdelic, T., Carić, T., & Lalla-Ruiz, E. (2019). A Survey on the Electric Vehicle Routing Problem: Variants and Solution Approaches. In *Journal of Advanced Transportation* (Vol. 2019). Hindawi Limited. https://doi.org/10.1155/2019/5075671

Futalef, J. P., Muñoz-Carpintero, D., Rozas, H., & Orchard, M. E. (2023). An online decision-making strategy for routing of electric vehicle fleets. *Information Sciences*, *625*, 715–737. https://doi.org/10.1016/j.ins.2022.12.108

Ghobadi, A., Tavakkoli-Moghaddam, R., Fallah, M., & Kazemipoor, H. (2021). Multi-Depot Electric Vehicle Routing Problem with Fuzzy Time Windows and Pickup/Delivery Constraints. *Journal of Applied Research on Industrial Engineering*, *8*(1), 1–18. https://doi.org/10.22105/jarie.2021.231764.1165

Goeke, D. (2019a). Granular tabu search for the pickup and delivery problem with time windows and electric vehicles. *European Journal of Operational Research*, *278*(3), 821–836. https://doi.org/10.1016/j.ejor.2019.05.010

Goeke, D. (2019b). Granular tabu search for the pickup and delivery problem with time windows and electric vehicles. *European Journal of Operational Research*, *278*(3), 821–836. https://doi.org/10.1016/j.ejor.2019.05.010

Goeke, D. (2019c). Granular tabu search for the pickup and delivery problem with time windows and electric vehicles. *European Journal of Operational Research*, *278*(3), 821–836. https://doi.org/10.1016/j.ejor.2019.05.010

Huang, Z., Shi, G., & Sukhatme, G. S. (2024). Can Large Language Models Solve Robot Routing. *arXiv e-prints*, arXiv-2403.

KAYNAKLAR DİZİNİ

Jia, Y. H., Mei, Y., & Zhang, M. (2022). A Bilevel Ant Colony Optimization Algorithm for Capacitated Electric Vehicle Routing Problem. *IEEE Transactions on Cybernetics*, *52*(10), 10855–10868. https://doi.org/10.1109/TCYB.2021.3069942

Jiang, X., Wu, Y., Zhang, C., & Zhang, Y. (2025). DRoC: Elevating large language models for complex vehicle routing via decomposed retrieval of constraints. In 13th international Conference on Learning Representations, ICLR 2025. OpenReview. net.

Keskin, M., & Çatay, B. (2018). A matheuristic method for the electric vehicle routing problem with time windows and fast chargers. *Computers and Operations Research*, *100*, 172–188. https://doi.org/10.1016/j.cor.2018.06.019

Kim, Y. D., Shim, S. O., Kim, S. B., Choi, Y. C., & Yoon, H. M. (2004). Parallel machine scheduling considering a job-splitting property. In *International Journal of Production Research* (Vol. 42, Issue 21, pp. 4531–4546). https://doi.org/10.1080/00207540410001720745

Kool, W., Van Hoof, H., & Welling, M. (n.d.). *ATTENTION, LEARN TO SOLVE ROUTING PROBLEMS!*

Kucukoglu, I., Dewil, R., & Cattrysse, D. (2021). The electric vehicle routing problem and its variations: A literature review. *Computers and Industrial Engineering*, *161*. https://doi.org/10.1016/j.cie.2021.107650

Lam, E., Desaulniers, G., & Stuckey, P. J. (2022). Branch-and-cut-and-price for the Electric Vehicle Routing Problem with Time Windows, Piecewise-Linear Recharging and Capacitated Recharging Stations. *Computers and Operations Research*, *145*. https://doi.org/10.1016/j.cor.2022.105870

Li, J., Xin, L., Cao, Z., Lim, A., Song, W., & Zhang, J. (2021). Heterogeneous Attentions for Solving Pickup and Delivery Problem via Deep Reinforcement Learning. In *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* (Issue 1).

Li, K., Liu, F., Wang, Z., Tong, X., Han, X., Yuan, M., & Zhang, Q. (2025). ARS: Automatic Routing Solver with Large Language Models. *arXiv preprint arXiv:2502.15359.*

Li, L., Li, T., Wang, K., Gao, S., Chen, Z., & Wang, L. (2019). Heterogeneous fleet electric vehicle routing optimization for logistic distribution with time windows and simultaneous pick-up and delivery service. *16th International Conference on Service Systems and Service Management (ICSSSM)*, 1–6.

Lin, B., Ghaddar, B., & Nathwani, J. (2022). Deep Reinforcement Learning for the Electric Vehicle Routing Problem With Time Windows. *IEEE Transactions on Intelligent Transportation Systems*, *23*(8), 11528–11538. https://doi.org/10.1109/TITS.2021.3105232

KAYNAKLAR DİZİNİ

Lin, C., Choy, K. L., Ho, G. T. S., Chung, S. H., & Lam, H. Y. (2014). Survey of Green Vehicle Routing Problem: Past and future trends. *Expert Systems with Applications*, *41*(4 PART 1), 1118–1138. https://doi.org/10.1016/j.eswa.2013.07.107

Lin, J., Zhou, W., & Wolfson, O. (2016). Electric Vehicle Routing Problem. *Transportation Research Procedia*, *12*, 508–521. https://doi.org/10.1016/j.trpro.2016.02.007

Luo, T., Heng, Y., Xing, L., Ren, T., Li, Q., Qin, H., Hou, Y., & Wang, K. (2024). A Two-Stage Approach for Electric Vehicle Routing Problem with Time Windows and Heterogeneous Recharging Stations. *Tsinghua Science and Technology*, *29*(5), 1300–1322. https://doi.org/10.26599/TST.2023.9010101

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533. https://doi.org/10.1038/nature14236

Morales, M. (2020). *Grokking deep reinforcement learning*.

Mozhdehi, A., Mohammadizadeh, M., & Wang, X. (2024). *The 37th Canadian Conference on Artificial Intelligence Edge-DIRECT: A Deep Reinforcement Learning-based Method for Solving Heterogeneous Electric Vehicle Routing Problem with Time Window Constraints*.

Munir, A., Parasuraman, R., Ye, J., & Song, W. Z. (2024). Route Planning for Electric Vehicles with Charging Constraints. *IEEE Vehicular Technology Conference*. https://doi.org/10.1109/VTC2024-Fall63153.2024.10757558

Narayanan, A., Misra, P., Ojha, A., Bandhu, V., Ghosh, S., & Vasan, A. (n.d.). *A Reinforcement Learning Approach for Electric Vehicle Routing Problem with Vehicle-to-Grid Supply; A Reinforcement Learning Approach for Electric Vehicle Routing Problem with Vehicle-to-Grid Supply*.

Nazari, M., Oroojlooy, A., Takáč, M., & Snyder, L. V. (n.d.). *Reinforcement Learning for Solving the Vehicle Routing Problem*.

Nolte, F. P., Wilken, N., & Bartelt, C. (2021). Rendezvous Delivery: Utilizing Autonomous Electric Vehicles to Improve the Efficiency of Last Mile Parcel Delivery in Urban Areas. *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events, PerCom Workshops 2021*, 148–153. https://doi.org/10.1109/PerComWorkshops51409.2021.9430987

Nolz, P. C., Absi, N., Feillet, D., & Seragiotto, C. (2022). The consistent electric-Vehicle routing problem with backhauls and charging management. *European Journal of Operational Research*, *302*(2), 700–716. https://doi.org/10.1016/j.ejor.2022.01.024

KAYNAKLAR DİZİNİ

Nouicer, D., Msadaa, I. C., & Grayaa, K. (2023). A Novel Routing Solution for EV Fleets: A Real-World Case Study Leveraging Double DQNs and Graph-Structured Data to Solve the EVRPTW Problem. *IEEE Access*, *11*, 122116–122127. https://doi.org/10.1109/ACCESS.2023.3327324

Ottoni, A. L. C., Nepomuceno, E. G., Oliveira, M. S. de, & Oliveira, D. C. R. de. (2022). Reinforcement learning for the traveling salesman problem with refueling. *Complex and Intelligent Systems*, *8*(3), 2001–2015. https://doi.org/10.1007/s40747-021-00444-4

Özden, G., & Sarıçiçek, İ. (2021). Dağıtım Operasyonlarında Kamyon-Dron Ekip Lojistiği Çalışmaları. *Yönetim Araştırmaları / Mühendislik Uygulamaları Sempozyumu*.

Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2008). A survey on pickup and delivery problems: Part II: Transportation between pickup and delivery locations. *Journal Fur Betriebswirtschaft*, *58*(2), 81–117. https://doi.org/10.1007/s11301-008-0036-4

Qi, C., Zhu, Y., Song, C., Yan, G., Xiao, F., Da wang, Zhang, X., Cao, J., & Song, S. (2022). Hierarchical reinforcement learning based energy management strategy for hybrid electric vehicle. *Energy*, *238*. https://doi.org/10.1016/j.energy.2021.121703

Qin, H., Su, X., Ren, T., & Luo, Z. (2021). A review on the electric vehicle routing problems: Variants and algorithms. *Frontiers of Engineering Management*, *8*(3), 370–389. https://doi.org/10.1007/s42524-021-0157-1

Rastani, S., & Çatay, B. (2023). A large neighborhood search-based matheuristic for the load-dependent electric vehicle routing problem with time windows. *Annals of Operations Research*, *324*(1–2), 761–793. https://doi.org/10.1007/s10479-021-04320-9

Richard S. S., & Andrew G. B. (2018). *Reinforcement learning: an introduction*.

Rodríguez-Esparza, E., Masegosa, A. D., Oliva, D., & Onieva, E. (2024). A new Hyper-heuristic based on Adaptive Simulated Annealing and Reinforcement Learning for the Capacitated Electric Vehicle Routing Problem. *Expert Systems with Applications*, *252*. https://doi.org/10.1016/j.eswa.2024.124197

Schneider, M., Stenger, A., & Goeke, D. (2014). The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, *48*(4), 500–520. https://doi.org/10.1287/trsc.2013.0490

Soni, H., Gupta, V., & Kumar, R. (n.d.). *Motion Planning using Reinforcem ent Learning for Electric Vehicle Battery Optimization(EVBO)*.

Soysal, M., Çimen, M., & Belbağ, S. (2020). Pickup and delivery with electric vehicles under stochastic battery depletion. *Computers and Industrial Engineering*, *146*. https://doi.org/10.1016/j.cie.2020.106512

KAYNAKLAR DİZİNİ

Tang, M., Zhuang, W., Li, B., Liu, H., Song, Z., & Yin, G. (2023). Energy-optimal routing for electric vehicles using deep reinforcement learning with transformer. *Applied Energy*, *350*. https://doi.org/10.1016/j.apenergy.2023.121711

Taş, D. (2021). Electric vehicle routing with flexible time windows: a column generation solution approach. *Transportation Letters*, *13*(2), 97–103. https://doi.org/10.1080/19427867.2020.1711581

Wang, C., Cao, M., Jiang, H., Xiang, X., & Zhang, X. (2024). A Deep Reinforcement Learning-Based Adaptive Large Neighborhood Search for CapacitatedElectric Vehicle Routing Problems. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 1–14. https://doi.org/10.1109/tetci.2024.3444698

Wang, D. L., Ding, A., Chen, G. L., & Zhang, L. (2023). A combined genetic algorithm and A* search algorithm for the electric vehicle routing problem with time windows. *Advances in Production Engineering And Management*, *18*(4), 403–416. https://doi.org/10.14743/apem2023.4.481

Wang, L., Ding, Y., Chen, Z., Su, Z., & Zhuang, Y. (2024). Heuristic Algorithms for Heterogeneous and Multi-Trip Electric Vehicle Routing Problem with Pickup and Delivery. *World Electric Vehicle Journal*, *15*(2). https://doi.org/10.3390/wevj15020069

Wang, M., Wei, Y., Huang, X., & Gao, S. (2024). An End-to-End Deep Reinforcement Learning Framework for Electric Vehicle Routing Problem. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2024.3432911

Xu, W., Zhang, C., Cheng, M., & Huang, Y. (2022). Electric Vehicle Routing Problem with Simultaneous Pickup and Delivery: Mathematical Modeling and Adaptive Large Neighborhood Search Heuristic Method. *Energies*, *15*(23). https://doi.org/10.3390/en15239222

Yang, S., Ning, L., Tong, L. C., & Shang, P. (2021). Optimizing electric vehicle routing problems with mixed backhauls and recharging strategies in multi-dimensional representation network. *Expert Systems with Applications*, *176*. https://doi.org/10.1016/j.eswa.2021.114804

Yao, Y., Zhu, X., Dong, H., Wu, S., Wu, H., Tong, L. C., & Zhou, X. (2019). ADMM-based problem decomposition scheme for vehicle routing problem with time windows. *Transportation Research Part B: Methodological*, 156–174.

Yilmaz, Y., & Kalayci, C. B. (2022). Variable Neighborhood Search Algorithms to Solve the Electric Vehicle Routing Problem with Simultaneous Pickup and Delivery. *Mathematics*, *10*(17). https://doi.org/10.3390/math10173108

KAYNAKLAR DİZİNİ

Zhang, S., Chen, M., Zhang, W., & Zhuang, X. (2020). Fuzzy optimization model for electric vehicle routing problem with time windows and recharging stations. *Expert Systems with Applications*, *145*. https://doi.org/10.1016/j.eswa.2019.113123

Zhang, S., Zhou, T., Fang, C., & Yang, S. (2024). A novel collaborative electric vehicle routing problem with multiple prioritized time windows and time-dependent hybrid recharging. *Expert Systems with Applications*, *244*. https://doi.org/10.1016/j.eswa.2023.122990

Zhang, Y., Li, M., Chen, Y., Chiang, Y. Y., & Hua, Y. (2023). A Constraint-Based Routing and Charging Methodology for Battery Electric Vehicles With Deep Reinforcement Learning. *IEEE Transactions on Smart Grid*, *14*(3), 2446–2459. https://doi.org/10.1109/TSG.2022.3214680

Zhao, M., & Lu, Y. (2019). A heuristic approach for a real-world electric vehicle routing problem. *Algorithms*, *12*(2). https://doi.org/10.3390/a12020045

Zhou, S., Zhang, D., Yuan, W., Wang, Z., Zhou, L., & Bell, M. G. H. (2024). Pickup and delivery problem with electric vehicles and time windows considering queues. Transportation Research Part C: Emerging Technologies, 167. https://doi.org/10.1016/j.trc.2024.104829