



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



ŞEHİR İÇİ TRAFİK SİNYAL AĞININ
TAKVİYELİ ÖĞRENME ALGORİTMALARI
VE NESNELERİN İNTERNETİ TABANLI
KONTROLÜ

Seyit Alperen ÇELTEK

DOKTORA TEZİ

Elektrik Elektronik Mühendisliği Anabilim Dalı

Temmuz- 2021
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Seyit Alperen ÇELTEK tarafından hazırlanan “Şehir İçi Trafik Sinyal Ağının Takviyeli Öğrenme Algoritmaları ve Nesnelerin İnterneti Tabanlı Kontrolü” adlı tez çalışması **09/07/2021** tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü **Elektrik Elektronik Mühendisliği** Anabilim Dalı’nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Doç. Dr. Ahmet KAYABAŞI

.....

Danışman

Doç. Dr. Akif DURDU

.....

Üye

Dr. Öğr. Üyesi Ersin KAYA

.....

Üye

Doç. Dr. Kadir SABANCI

.....

Üye

Doç. Dr. Seyfettin Sinan GÜLTEKİN

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

Bu tez çalışması TÜBİTAK tarafından 2211-C Yurt İçi Öncelikli Alanlar Doktora programı kapsamında desteklenmiştir.

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Seyit Alperen ÇELTEK

Tarih:

ÖZET

DOKTORA TEZİ

ŞEHİR İÇİ TRAFİK SİNYAL AĞININ TAKVİYELİ ÖĞRENME ALGORİTMALARI VE NESNELERİN İNTERNETİ TABANLI KONTROLÜ

Seyit Alperen ÇELTEK

Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Akif DURDU

2021, 87 Sayfa

Jüri

Doç. Dr. Akif DURDU

Doç. Dr. Seyfettin Sinan GÜLTEKİN

Dr. Öğr. Üyesi Ersin KAYA

Doç. Dr. Kadir SABANCI

Doç. Dr. Ahmet KAYABAŞI

Uyarlanabilir trafik sinyal kontrolü, sinyal süresini trafiğin durumuna göre ayarlayan ve trafiği gerçek zamanlı olarak yöneten zorlu bir konudur. Klasik trafik sinyal kontrol sistemlerinin performansı, uygun sinyal planlamasını yapmak için yeterli değildir. Bu tez çalışmasında trafikteki gecikmeyi azaltmak için nesnelerin interneti tabanlı üç katmanlı bir kontrol yöntemi önerilmektedir. Önerilen yaklaşım ile her bir katmanda optimizasyon işlemi gerçekleştirilmektedir. En alt katman olan kenar bilişim katmanı, gerçek zamanlı ve yerel bir optimizasyon sağlamaktadır. Orta katman olan sis bilişim katmanı, gerçek zamanlı ve küresel bir optimizasyon işlemi gerçekleştirmektedir. En üst katman olan bulut bilişim ise alt katmanlardan gelen veriler çevrimdışı olarak değerlendirilmekte ve alttaki iki katmanın performansını arttıracak parametreler araştırılmaktadır. Ayrıca hem izole kavşakta hem de trafik ağında kullanılmak üzere yenilikçi bir yaklaşım önerilmektedir. Literatürdeki yöntemler araçların yalnızca kırmızı sinyal fazından dolayı beklemesinden kaynaklanan gecikme süresini minimize etmeye çalışmaktadır. Oysa kavşaklarda gecikmenin araçların yavaşlaması, hızlanması ve beklemesi olmak üzere üç farklı nedeni vardır. Bu çalışmada üç farklı gecikme türünün de minimize edilmesi hedeflenmiştir. Hızlanma ve yavaşlamadan kaynaklı gecikmeyi azaltmak için uygulanan kavşak bilgisinin yanı sıra bitişik kavşakların verilerini de girdi olarak kullanan bir kontrol önerilmiştir. Bu çalışmada yöntem geliştirmek ve geliştirilen yöntemlerin performanslarını değerlendirmek için SUMO trafik simülatörü kullanılmıştır. Hem izole bir kavşakta hem de çoklu trafik sinyallerinin oluşturduğu trafik ağlarında uygulanabilir bir mimari tasarlanmıştır. Böylece literatüre ve akıllı trafik kontrol sistemlerinin gelişimine büyük bir katkı sağlamak amaçlanmıştır.

Anahtar Kelimeler: Adaptif Trafik Sinyal Kontrolü, Gerçek Zamanlı Trafik Kontrolü, Nesnelerin İnterneti, Sosyal Öğrenme Parçacık Sürü Algoritması, Takviyeli Öğrenme.

ABSTRACT

PhD THESIS

THE REINFORCEMENT LEARNING ALGORITHMS and INTERNET OF THINGS BASED CONTROL FOR URBAN TRAFFIC SIGNAL NETWORK

Seyit Alperen CELTEK

**Konya Technical University
Institute of Graduate Studies**

Department of Electric and Electronic Engineering

Advisor: Assoc. Prof. Dr. Akif DURDU

2021, 87 Pages

Jury

Assoc. Prof. Dr. Akif DURDU

Assoc. Prof. Dr. Seyfettin Sinan GÜLTEKİN

Asst. Prof. Dr. Ersin KAYA

Assoc. Prof. Dr. Kadir SABANCI

Assoc. Prof. Dr. Ahmet KAYABAŞI

Adaptive traffic signal control is a challenging issue that adjusts the signal duration according to the traffic situation and manages the traffic in real-time. The performance of classical traffic signal control approaches is not sufficient to find the appropriate signal planning. In this thesis, a three-layer control method based on the Internet of Things is proposed to reduce the delay in traffic. With the proposed approach, optimization is performed at each layer. The bottom layer, the edge computing layer, provides real-time and local optimization. The middle layer, the fog computing layer, performs a real-time and global optimization process. In cloud computing, which is the top layer, the data from the lower layers are evaluated offline and parameters that will increase the performance of the lower two layers are investigated. In addition, an innovative approach is proposed to be used both in isolated intersections and in the traffic network. The methods in the literature only try to minimize the delay at intersections caused by vehicles waiting due to the red signal phase. There are three different reasons for delay at intersections: vehicles slowing down, accelerating and waiting. In this study, three different types of delay were also investigated. Proposed control uses not only the applied intersection information but also use the adjacent intersection data as an input. In this study, the SUMO traffic simulator was used for performance evaluation and method improvements. In addition, a model has been developed in which the internet of things, deep learning and reinforcement learning algorithms are used together. An architecture that can be applied both in an isolated intersection and in traffic networks formed by multiple traffic signals is designed, and it is aimed to make a great contribution to the literature and the development of intelligent traffic control system.

Keywords: Adaptive Traffic Signal Control, Real Time Traffic Control, Internet of Things, Social Learning Particle Swarm Algorithm, Reinforcement Learning.

ÖNSÖZ

Her şeyden önce bu tez çalışmasını güven ve huzur ortamında yapmamı sağlayan güvenlik güçlerimize ve bu uğurda canlarını feda eden aziz şehitlerimize,

Bilgi ve tecrübeleriyle bana yol gösteren, benden yardımlarını hiçbir zaman esirgemeyen, her zaman olumlu yönde teşvik eden ve bana ilham kaynağı olan değerli danışmanım Doç. Dr. Akif DURDU'ya,

Tez hakkında birçok yapıcı öneride bulunan değerli tez izleme komite üyeleri, Doç. Dr. Seyfettin Sinan GÜLTEKİN ve Dr. Öğr. Üyesi Ersin KAYA'ya,

Bu çalışmada kullanılan trafik verilerinin temini konusunda yardım aldığım Mosaş Akıllı Ulaşım ve Tartım Teknolojileri Şirketine, tez çalışmamda bana yardımcı olan Müzammil Muhammed Eltejani ALİ'ye,

Tez çalışmam süresince 2211-C (1649B031900701) programıyla bana destek sağlayarak ülkemizde bilimin gelişmesi ve yeni araştırmacıların yetiştirilmesi için çaba gösteren TÜBİTAK'a,

Bu günlere gelmemi borçlu olduğum, hayallerime ulaşmam için elinden gelen hiçbir fedakarlığı esirgemeyen canım babam ve canım anneme,

Her daim sevgilerini yanımda hissettiğim canım kız kardeşim Merve ve canım erkek kardeşim Oğuz'a,

Hayatımın her döneminde beni her daim motive eden, sevgi ve desteğini her zaman yüreğimde hissettiğim canım eşim Meltem'e ve bebeğim İtir Bilge'ye,

Minnettarlığımı sunar, çok teşekkür ederim.

Seyit Alperen ÇELTEK
KONYA-2021

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
ŞEKİLLER TABLOSU	x
ÇİZELGELER TABLOSU	xi
SİMGELER VE KISALTMALAR	xii
1. GİRİŞ	1
1.1. Motivasyon ve Amaç	1
1.2. Tezin Katkısı	2
1.3. Tezin Organizasyonu	3
2. TRAFİK KAVRAMLARI	4
2.1. Genel Trafik Sinyal Kavramları	4
2.1. Trafik Sinyal Kontrol Yöntemleri Performans Ölçütleri	5
3. KAYNAK ARAŞTIRMASI	8
3.1. Trafik Sinyal Kontrol Yöntemleri.....	9
3.2. Adaptif Trafik Sinyal Kontrol Yöntemleri	10
3.3. Takviyeli Öğrenme ile Trafik Sinyal Kontrol Sistemleri	11
3.4. Trafik Sinyal Kontrol Sistemlerinde Kullanılan Takviyeli Öğrenme Bileşenleri	13
3.5. Nesnelerin İnterneti ve Trafik Sinyal Kontrolü	14
4. MATERYAL VE YÖNTEM.....	16
4.1. Simülasyon Programı.....	16
4.1.1. SUMO Simülatörü	17
4.2. Sürü Tabanlı Optimizasyon Algoritmaları	18
4.2.1. Parçacık Sürü Algoritması	18
4.2.1. Sosyal Öğrenme Parçacık Sürü Algoritması	20
4.2.3. Edimsel Koşullanma ile Sosyal Öğrenme Parçacık Sürü Algoritması	21
4.2.4. Sürü Algoritmalarının Test Edilmesi	22
4.3. Takviyeli Öğrenme	23

4.3.1. Q-Öğrenme Algoritması	24
4.3.2. Derin Q-Öğrenme Algoritması	25
4.4. Nesnelerin İnterneti Kavramı.....	26
4.4.1. Bulut Bilişim.....	27
4.4.2. Sis Bilişim.....	28
4.4.3. Kenar Bilişim.....	29
5. SİMULASYON ORTAMI ve TRAFİK SİNYAL OPTİMİZASYONU	30
5.1. Simülasyon Ortamının Oluşturulması ve Deneysel Parametreler	30
5.1.1. İzole Kavşak Simülasyon Ortamının Oluşturulması ve Parametreler	30
5.1.2. Koordineli Kavşak Simülasyon Ortamının Oluşturulması ve Parametreler ..	32
5.1.3. Trafik Sinyal Ağı Simülasyon Ortamının Oluşturulması ve Parametreler ...	33
5.2. Adaptif Trafik Sinyal Kontrolü.....	33
5.2.1. Sürü Tabanlı Algoritmalar ile Faz Süresi Optimizasyonu.....	34
5.2.2. Takviyeli Öğrenme ile Optimum Faz Sıralamasının Ayarlanması.....	36
5.2.3 Öğrenme Mekanizması	40
5.3. Nesnelerin İnterneti Platformunda Adaptif Trafik Sinyal Kontrolü	42
6. ADAPTİF TRAFİK SİNYAL KONTROL UYGULAMALARI ve SONUÇLAR	46
6.1. Sürü Tabanlı Trafik Sinyal Kontrol Uygulaması ve Sonuçları.....	46
6.1.1. Sürü Tabanlı Trafik Sinyal Kontrol Uygulamasının Değerlendirilmesi... 53	
6.2. Edimsel Koşullanma ile Sürü Tabanlı Trafik Sinyal Kontrol Uygulaması ve Sonuçları	54
6.2.1. Edimsel Koşullanma ile Sürü Tabanlı Trafik Sinyal Kontrol Uygulaması ve Sonuçları	56
6.3. Adaptif Trafik Sinyal Kontrolünün DQÖ ile Gerçekleştirilmesi.....	56
6.4. DQÖ Tabanlı Adaptif Trafik Sinyal Kontrol Sistemleri için Optimum Faz Eylem Süresinin İncelenmesi	59
6.4.1. Optimum Faz Eylem Süresi ile DQÖ Tabanlı Adaptif Trafik Sinyal Kontrol Sonuçlarının Değerlendirilmesi	62
6.5. Koordineli Kavşak Trafik Sinyal Kontrolü Uygulaması	62
6.5.1. Önerilen Yenilikçi Yaklaşımın Değerlendirilmesi	68
6.6. Trafik Sinyal Ağı Kontrolü Uygulaması.....	69
6.7. Nesnelerin İnterneti Platformunda Adaptif Trafik Sinyal Ağı Uygulaması	72
6.7.1. Nesnelerin İnterneti Platformunda Adaptif Trafik Sinyal Kontrol Değerlendirmesi.....	77

7. TARTIŞMA VE ÖNERİLER.....	78
7.1. Öneriler	79
KAYNAKLAR	81



ŞEKİLLER TABLOSU

Şekil 4.1. Sumo konfigürasyon dosyasına ait görsel.	17
Şekil 4.2. SUMO_GUI ile simulasyonun görselleştirilmesi.	18
Şekil 4.3. DQÖ çalışma prensibi	26
Şekil 5.1. Uygulama yapılan izole kavşak, Kilis.	31
Şekil 5.2. Uygulama yapılan kavşağın simülasyon model görseli.	31
Şekil 5.3. Uygulama yapılan koordineli kavşak	32
Şekil 5.4. Manhattan tipi 5x5 trafik ağı	33
Şekil 5.5. Takviyeli öğrenme ile adaptif trafik sinyal kontrolü	37
Şekil 5.6. Eylem temsili	38
Şekil 5.7. Aynı aksiyonların ardı ardına uygulanması	39
Şekil 5.8. Farklı aksiyona geçiş.	40
Şekil 5.9. Öğrenme mekanizması	41
Şekil 5.10. Trafik düğümleri ve düğümlerin oluşturduğu kavşak kümesi.	42
Şekil 5.11. İki kavşak kümesinin birbirleri ile haberleşmesi.	43
Şekil 5.12. Trafik sinyal ağı görseli	43
Şekil 5.13. Nesnelerin interneti mimarilerinin trafik ağına uygulanması.	45
Şekil 6.1. Kavşaklardaki gecikme	63
Şekil 6.2. Koordineli kavşak trafik sinyal uygulaması	64
Şekil 6.3. Kavşak kapasitesi	65
Şekil 6.4. Ortalama bekleme süresi karşılaştırması	66
Şekil 6.5. Ortalama araç seyahat süresi	67
Şekil 6.6. Karayolu üzerindeki ortalama araç hızı	68
Şekil 6.7. Trafik sinyal ağı senaryosu	72
Şekil 6.8. Sis düğümü ile bulut merkezi arasındaki haberleşme.	74
Şekil 6.9. Bekleme süresine ait performans çıktıları	75
Şekil 6.10. Ortalama seyahat süresine ait performans çıktıları	75
Şekil 6.11. Ortalama araç hızı	76
Şekil 6.12. Durma sayısı kriteri	76

ÇİZELGELER TABLOSU

Çizelge 4.1. Kullanılan CEC 13 Fonksiyonları	22
Çizelge 4.2. OCA-SL-PSO'nun Klasik yöntemlerle karşılaştırılması	23
Çizelge 5.1. Eylem kümesi ve açıklamaları.....	39
Çizelge 6.1. Parçacık sayısının sonuca etkisi	48
Çizelge 6.2. Hesaplama ve simülasyon süre toplamaları	49
Çizelge 6.3. Standart sapma sonuçları.	50
Çizelge 6.4. PSO ve SL-PSO karşılaştırılması	51
Çizelge 6.5. Farklı trafik rejimlerinde performans değerlendirmesi	52
Çizelge 6.6. OCA-SL-PSO sonuçları	55
Çizelge 6.7. Sabit zamanlı kontrolör sonuçları.....	57
Çizelge 6.8. DQÖ simülasyon parametreleri	57
Çizelge 6.9. DQÖ Sonuçları	58
Çizelge 6.10. Sabit zamanlı ve DQÖ karşılaştırması.....	58
Çizelge 6.11. DQÖ tabanlı yöntemde farklı faz eylem sürelerinin incelenmesi	60
Çizelge 6.12. Kısa eylem süresi ile uzun eylem süresinin karşılaştırılması	61
Çizelge 6.13. En iyi faz eylem süresi ile sabit zamanlı kontrol karşılaştırılması.	61
Çizelge 6.14. Simülasyon parametreleri	70
Çizelge 6.15. Farklı faz eylem süreleri ile DQÖ tabanlı dağıtılmış yönetim sonuçları	71

SİMGELER VE KISALTMALAR

Simgeler

P_{best}	: En iyi yerel parçacık
g_{best}	: En iyi küresel parçacık
$x_i(t)$	: Parçacık konumu
$V_i(t)$	: Parçacık hızı
r_1, r_2	: Rastgele seçilen parametreler
w	: Atalet ağırlığı
c_1, c_2	: Parçacık hızlanma katsayıları
$i_i(t)$	: Öğreten parçacık
$c_i(t)$	: Sosyal etki faktörü
p_i^L	: Öğrenme oranı
$d(C)_{ij}$	: Kontrol gecikmesi
$d(\theta)_{ij}$	: Ortalama gecikme süresi
Λ	: Weibull katsayısı
t_{ij}	: Ortalama seyahat süresi
θ_{ij}	: Offset değeri
T	: Toplam seyahat süresi
PF_k	: K yolunun düzeltme faktörü
C_k	: K yolunun kapasitesi
C	: Döngü süresi
g_k	: Etkin kırmızı süresi
r_k	: Etkin yeşil süresi etkin yeşil süresi
x_k	: Yolun doygunluk süresi
$d_{1k} d_{2k}$	: Birim gecikme

Kısaltmalar

TTT	: Toplam seyahat süresi
PSO	: Parçacık sürü optimizasyon algoritması
SL-PSO	: Sosyal öğrenme parçacık sürü algoritması
DQÖ	: Derin Q öğrenme algoritması
RL	: Takviyeli öğrenme algoritması
SUMO	: Kentsel hareketlilik simülatörü
IoT	: Nesnelerin interneti



1. GİRİŞ

Bu bölümde, tez çalışmasının motivasyonu, literatüre katkısı ve organizasyonu sunulmaktadır. İlk olarak tez çalışmasının motivasyonları sıralanarak tezin amacına yer verilmiştir. Ardından tez çalışması kapsamında elde edilen sonuçların literatüre katkısından bahsedilmiştir. Son olarak tezin organizasyonuna yer verilmiştir.

1.1. Motivasyon ve Amaç

Trafik, tüm dünyada bilimsel ve pratik geçerliliği olan önemli bir sorundur (Ge ve ark., 2019). Büyük şehirlerin nüfusu ve kişi başına düşen özel araç sayısındaki hızlı artış, orantısız kentleşme ve altyapı eksiklikleri birleşince trafiğin yönetilmesi daha da zorlaşmaktadır. Her geçen gün daha karmaşık hale gelen trafik kontrolü, insanların ve araçların hızlı, konforlu, ekonomik ve güvenli bir şekilde transferini gerçekleştirmek için büyük önem arz etmektedir. Verimsiz trafik yönetimi, çevre kirliliği, trafik sıkışıklığı, can güvenliği, park etme ve araç trafiği gibi birçok olumsuz sonuca yol açmaktadır (Spall ve Chin, 1997; McCrea ve Moutari, 2010; Ye ve ark., 2017; Sun ve ark., 2018). Kent içi ulaşımında yaşanan sorunlar sadece maddi değil, aynı zamanda günlük yaşamı etkileyen birçok sosyal sorunun da başlıca sebebidir (Bull ve Thomson, 2002).

Kentsel trafik sorunlarının üstesinden gelmek için trafikteki taşıt sayısının artışı gözetilerek mevcut trafik kapasitesi artırılmalıdır. Mevcut trafik kapasitesi iki farklı şekilde arttırılabilir. Birinci çözüm mevcut trafik altyapısında değişiklik yapmaktır. Kentsel trafik altyapısında düzenlemeler yapmak trafik hacmini ve kapasitesini arttıran uzun vadede etkin bir çözümdür (Roess ve ark., 2004). Ancak kısa ve orta vadede trafik problemlerini daha da arttırır. Ayrıca mevcut alt yapıyı arttırma işlemi yüksek maliyet ve çaba gerektirdiğinden esnek ve uygulanabilir bir çözüm değildir. İkinci ve daha esnek bir çözüm ise trafik akımlarının kontrolüdür.

Trafik akımlarının verimli bir şekilde kontrol edilmesi ile mevcut altyapının hacmini arttırmak mümkündür (Garcia-Nieto ve ark., 2013; Zhou ve ark., 2017). Şehirlerdeki trafik sorunlarının çoğu, kavşaklardaki çakışan trafik akımlarından kaynaklanmaktadır (Celtek ve ark., 2020). Bu kritik noktalarda yapılacak her bir iyileştirmenin trafiğin genelinde olumlu bir etkisi olacaktır (Yau ve ark., 2017). Kavşaklardaki trafik akışları trafik sinyal kontrol sistemleri (TSCS) tarafından kontrol edilir.

Mevcut trafik sinyal kontrol çözümleri, kentsel nüfusun hızla artması, mevcut yolların kapasitelerinin sınırlı olması ve şehirlerimizin altyapısının dijitalleşmesi, optimizasyon için gereken hesaplama süresinin fazla olması ve verimsiz veri iletimi nedeniyle günümüz sinyalizasyon problemleri için yetersiz kalmaktadır (Cui ve Lee, 2013). Günümüz sorunlarına yeteri kadar çözüm üretemeyen mevcut trafik kontrol yöntemlerinin, yakın gelecekteki otonom araçlarında yer alacağı trafik problemlerine çare bulması umutsuz bir beklentidir. Sürücüsüz araçların sayısının her geçen gün arttığı yeni nesil bir trafik için daha modern ve daha adaptif çözümün bulunması bir zorunluluktur. Bütün bunlar göz önüne alındığında tek bir kavşağı değil; birden fazla kavşağın oluşturduğu ağın bütününe hedefleyen, uyarlanabilir, trafiğin tüm dinamik parametrelerini dikkate alan, pratikte uygulanabilir ve gerçek zamanlı bir trafik sinyalizasyon kontrolüne ihtiyaç vardır.

Nitekim bu tez çalışmasında şehirlerdeki trafik sıkışıklığını azaltmak için yapay optimizasyon algoritmaları ve nesnelerin interneti tabanlı trafik kontrol sistemi modeli tasarlanması amaçlanmıştır. Yapılan trafik sinyal kontrol tasarımları ile trafikteki ortalama gecikme süresinin ve kavşaklardaki kuyruk uzunluğunun azaltılması ve bununla birlikte araçların ortalama hızının artırılması sağlanmıştır.

1.2. Tezin Katkısı

Bu çalışmada gerçek zamanlı adaptif trafik sinyal kontrol sistemi geliştirilmiştir. Hali hazırda kullanılmakta olan klasik yöntemlerin aksine yenilikçi kontrol ve optimizasyon algoritmaları kullanılmıştır. Şehir içi trafik sinyallerinden oluşan bir ağın gerçek zamanlı kontrolü dağıtılmış yönetim ile koordine edilmiştir. Bu koordinasyon nesnelerin interneti teknolojisi mimarileri kullanılarak gerçekleştirilmiştir.

Bu çalışmanın literatüre katkısı şu şekilde özetlenmiştir:

- Trafiğin her rejiminde verimli çalışan bir kontrol sistemi geliştirmek: Doymamış, doymuş ve aşırı doymuş trafik durumlarına gerçek zamanlı olarak uyum sağlayan trafik kontrol yöntemi bu tezin en önemli katkısıdır.
- Üç katmanlı adaptif trafik sinyal kontrol sistemi geliştirmek: Trafiğin gerçek zamanlı olarak hem yerel hem de küresel optimizasyonun gerçekleştirilmesi bu tezin başlıca katkılarındandır.
- Çevrim dışı yapılan optimizasyon ile çevrim içi yapılan optimizasyon performansının iyileştirilmesi.

- Trafiğin anlık değişimlerine (mevsimsel değişiklik, kaza, konvoy v.b) uyum sağlama, Trafiği oluşturan tüm unsurların (sürücü, yaya, sağa- sola dönen araç oranı v.b) dikkate alınması.
- Nesnelerin interneti ve bilişim mimarileri tabanlı çözüm: Trafik sinyal ağında hızlı bir iletişim ve etkin bir koordinasyon. Hem merkezci hem de dağıtılmış merkez yöntemlerine uygun yenilikçi bir kontrol sisteminin uygulanması.

1.3. Tezin Organizasyonu

Bu çalışma yedi bölüm olarak düzenlenmiştir. Bu bölümde tezin ana konusunu kısaca tanıtılmıştır. Tezin motivasyonuna ve hedefine bu bölümde yer verilmektedir. Bölüm 2'de, bu kavramsal çerçeveye yer verilmiştir. Bölüm 3, trafik sinyal kontrolü ile ilgili mevcut literatüre genel bir bakış sunmaktadır. Bölüm 4'te, tez kapsamında kullanılan trafik simülatörü ve bu tezde kullanılan algoritmalar tanıtılmaktadır. Bölüm 5, simülasyon ortamının oluşturulmasını, gerçek zamanlı adaptif trafik sinyal kontrol yöntemi için önerilen yöntemleri detaylı olarak sunmaktadır. Tez kapsamında yapılan tüm çalışmalar ve sonuçları kronolojik sıra ile Bölüm 6'da açıklanmıştır. Son olarak Bölüm 7'de, yapılan çalışmaların sonuçları ve performans testleri değerlendirilerek tez sonlandırılmıştır. Gelecekteki çalışmalar için kullanılabilecek fikirler de bu bölümde verilmiştir.

2. TRAFİK KAVRAMLARI

Bu bölümde kavramsal çerçeve ve trafik sinyal kontrol yöntemlerinin performans ölçütlerine ait temel bilgiler verilmektedir. İlk olarak genel trafik sinyal kavramları tanımlanmıştır. Son olarak ise trafik sinyal kontrol yöntemlerinin performans ölçütleri hakkında bilgi vermektedir.

2.1. Genel Trafik Sinyal Kavramları

Bu bölümde trafik kontrol sistemlerinin ve sistemleri oluşturan alt birimlerin tanımlarına yer verilmiştir (Gordon ve ark., 2005);

- **İzole kavşak trafik sinyal kontrol birimi:** Diğer sinyal kontrol birimlerinden bağımsız olarak çalışır. İzole sinyal kontrol biriminde bir veya daha fazla faz, trafik uyarmalı olarak çalışırsa, çevrim uzunluğu bir önceki çevrim uzunluğuna göre değişebilir.
- **Koordineli trafik sinyal kontrol birimi:** Birden fazla sinyal kontrolü birbirlerine bağlı olarak çalışır. Koordine edilen kavşaklar arasındaki araç grubu durmadan seyahatine devam eder. Araç grubunun sürekli hareket halinde olması için ardışık kavşakların sinyal fazlarını birbirleri ile uyumlu olacak şekilde koordine edilir.
- **İzole edilmiş kavşak:** Komşu kavşaklara mesafe olarak uzak olup ve bu yüzden sinyal koordinasyonu işlemlerine dahil edilmeyen kavşak türüdür.
- **Anayol kavşağı:** Ana yoldaki trafik koordinasyonunun sağlandığı kavşakların her birine verilen addır.
- **Trafik sinyal ağı:** Yoğunluğun yüksek olduğu kentsel alanlarda ve merkezi iş bölgelerinde yaygın olarak bulunan, oldukça kısa bloklardan oluşan ve sinyalize edilmiş kavşaklar dizisini oluşturan kavşakların her birine verilen addır.
- **Bağlantı:** İki kavşağı bağlayan ara yoldur.
- **Faz:** Ardışık iki yeşil ışık arasında geçen süredir.
- **Etkin Kırmızı:** Araçların kavşakta beklediği ve kuyruk oluşturduğu süredir. Saha uygulamalarında sarı ışık ve kırmızı ışık sinyal sürelerinin toplamına eşittir (Zheng ve Recker, 2013). Literatürde bazı çalışmalarda “kayıp süre” ifadesi olarak da geçmektedir (Akçelik ve Roupail, 1993).

- **Etkin Yeşil:** Araçların kavşaktan geçiş yaptığı ve dolayısı ile kavşaktaki kuyruk uzunluğun azaldığı süredir (Zheng ve Recker, 2013). Bir kavşağın kapasitesi o kavşağın etkin yeşil süresi ile doğrudan orantılıdır(Çetinkaya, 2008).
- **Döngü uzunluğu:** Fazların döngüsünü tamamlamak için gereken süredir.
- **Gecikme süresi:** Uygulanan trafik kontrol yöntemlerinin performans parametrelerinden biridir. Bir yol üzerindeki araçların ne kadar zaman kaybettiğini belirtir.
- **Doymuş Trafik Rejimi:** Etkin kırmızı süresinde kavşağa gelen araç sayısı ile etkin yeşil süresinde kavşaktan geçen araç sayısının birbirine eşit olma durumudur. Doymuş trafik rejiminde kavşaktaki taşıt kuyruğu bir döngüdeki yeşil fazı ile tamamen sıfırlanır.
- **Doymamış Trafik Rejimi:** Kavşaktaki taşıt kuyruğunun etkin yeşil süresi tamamlanmadan önce sıfırlanma durumudur. Doymamış trafik rejiminde kavşakta araç kalmaması için etkin yeşil süresinin bir kısmı yeterlidir. Böyle bir durumda etkin yeşil süresinin kısaltılması gerekmektedir.
- **Aşırı Doymuş Trafik Rejimi:** Kavşaktaki taşıt kuyruğunun etkin yeşil süresi boyunca sıfırlanamaması durumudur.
- **Ortalama Araç Bekleme Süresi:** Bir aracın tüm yolculuklar için ne kadar beklediğini ölçer. Araç hızının 0,5 m/s'nin altında olduğu süreyi ölçer.
- **Ortalama Araç Seyahat Süresi:** Bir aracın yolculuk süresini ölçer. Simülasyondaki her aracın yolculuğu farklı ve spesifiktir. Bu parametre her araba için başlangıç noktasından bitiş noktasına kadar ölçüm yapmaya başlar.
- **Ortalama Araç Seyir Hızı:** Bir aracın ortalama hızını ölçer. Güzergah uzunluğunun her araç için seyahat süresine bölünmesine eşittir.
- **Ortalama Durma Sayısı:** Hızı 0 m/s olan bir aracın durma sayısını ölçer.

2.1. Trafik Sinyal Kontrol Yöntemleri Performans Ölçütleri

Trafik işareti kontrol sistemlerinin kalitesini değerlendirmek için çeşitli ölçütler kullanılmıştır. Literatüre bakıldığında gecikme süresi ve sıra uzunluğu en popüler göstergeler olarak öne çıkmaktadır. Gecikme süresi, bir trafik sinyal kontrol stratejisinin verimliliğinin değerlendirmesinde en fazla kullanılan göstergedir. Toplam seyahat süresi, yolda kaybedilen zaman ve yakıt tüketimi gibi kritik parametreleri doğrudan etkiler.

Gecikme süresi hesaplanırken, aracın belirli mesafeyi kat etmesi için gereken geçen süre (pratik hesap sonucu elde edilen süre), gerçek hayatta harcadığı süre ile kıyaslanır. Bir aracın toplam gecikme süresini üç farklı bileşen oluşturur. Aracın karayolunda kırmızı ışığa yakalanarak hızını azaltması ya da tamamen durması için harcanan zamana yavaşlama gecikmesi denir. Araç stop halinde iken geçen süre durmadan kaynaklı gecikme olarak ifade edilebilir. Hızlanma gecikmesi, yeşil fazın başlangıcında araç hızlanmaya başladığında başlar ve araç normal hıza ulaştığında sona erer, bu da herhangi bir engel olmadan hareket eden hızdır. Gecikme tahmininin doğruluğu çok önemlidir, ancak, tekdüzen varış hızı nedeniyle gecikmeyi hesaplamak karmaşık bir iştir. Gecikme, gerçek trafik ağlarında, simülasyonda ve analitik modellerde ölçümle tahmin edilebilir. Analitik modeller kullanılarak gecikme ölçümü basit ve mümkündür. Sinyalize kavşak noktasındaki gecikmeyi tahmin etmek için yaygın olarak kullanılmıştır.

Bir kavşaktaki aracın ortalama gecikmesini tahmin etmek için uzun yıllardır kullanılan bazı gecikme modelleri vardır (HCM, 2000) (Webster, 1958b). Bu modellerin en büyük dezavantajı bazı varsayımlara dayanmasıdır. Bu modellere göre araçlar trafik ışığına bir Poisson işlemine göre yaklaşmaktadır. Bu tür modeller kullanılarak hesaplanan gecikmeler yalnızca teorik konsepte dayalıdır. Oysa gerçek trafik oldukça dinamiktir ve bundan dolayı karakteristikleri, matematiksel formüller tarafından yeterince temsil edilemediğinden gecikme değerleri gerçekte ölçülenden farklı olabilir.

Kuyruk uzunluğu ise kavşaklardaki kontrol sistemlerinin değerlendirilmesinde kullanılan bir diğer önemli faktördür. Kuyruk uzunluğu komşu kavşak kontrolünün verimli olup olmadığının da bir göstergesidir. Kuyruk uzunluğu tahmin yaklaşımları ikiye ayrılabilir (Liu ve ark., 2009). Birinci model kümülatif giriş-çıkış tabanlı bir yaklaşımdır (Webster, 1958b; Sharma ve ark., 2007). Bu model sadece trafik yoğunluğu düşük kavşaklar için kullanılabilir. Kuyruk uzunluğu kavşak ile detektör arası mesafeden fazla olan kavşaklarda kullanılamaz. İkinci model ise şok dalgası modelidir. Doymuş ve aşırı doymuş durumlarda kullanılır. Bu modelin en büyük dezavantajı ise araçların kavşağa varış hızının bilindiği varsayımdır (Ban ve ark., 2011).

Gecikme ve kuyruk uzunluğundan başka performans ölçütü olarak kullanılan kriterler de vardır. Bunlara örnek egzoz emisyonu, yakıt tüketimi veya güvenlik kriterleri verilebilir. Egzoz emisyonu özellikle son yıllarda çevreci araştırmacılar ve politikacılar tarafından sık sık kullanılmaktadır. Sinyalize kavşaklarda dikkat edilmesi gereken bir

önemli kriter güvenlidir. Trafik sinyalleri trafikteki karışıklığı önleyerek ciddi oranda kavşak güvenliğini artırır.

Bölüm 2’de ilk olarak genel trafik sinyal kavramlarına ve tanımlarına yer verilmiştir. Ardından önerilen trafik sinyal kontrol sistemlerinin performanslarını değerlendirmek için kullanılan performans ölçütleri tanımlanmıştır.



3. KAYNAK ARAŞTIRMASI

Bu bölüm, tez çalışması kapsamında yapılan literatür araştırmalarını sunmaktadır. İlk olarak halen sıkça kullanılan klasik kontrol sistemlerine yer verilmiştir. Ardından adaptif trafik sinyal kontrol sistemleri tanıtılmıştır. Bu çalışmada kullanılan yöntem olan takviyeli öğrenme ve nesnelerin interneti ile ilgili yapılan çalışmalara yer verilmiştir.

Etkin bir trafik sinyali kontrolü için aşılması gereken sorunlar genel olarak üç ana madde altında özetlenmiştir (Pavlis ve Recker, 2004);

- Mevcut ve beklenen trafik durumunu temsil eden matematiksel bir modelin geliştirilmesi (Modelleme, amaç fonksiyonunun ve parametrelerin belirlenmesi),
- Belirli bir performans indeksi olarak ifade edilebilecek gerçek zamanlı kontrol hedefinin belirlenmesi (Amaç fonksiyonuna ve giriş parametrelerine sınır tanımlama),
- İstenen kriterlerin gerçek zamanlı bir şekilde karşılandığı şekilde uygun bir optimizasyon tekniğinin tasarımı (Optimizasyon yöntemi).

Trafiğin matematiksel modellenmesi ve amaç fonksiyonunu tanımlama optimizasyon işleminin başarısına, dolayısıyla trafik kontrol sisteminin başarısına doğrudan etki eder. Trafiğin durumu anlık olarak modellenemez ise optimizasyon işleminin başarısı sınırlı olacaktır.

Klasik modelleme yöntemleri, genellikle mevcut kontrol sistemleri gözetilerek önerilen modelleme yöntemleridir (Dion ve Rakha, 2003). Çalışma prensipleri incelendiğinde, gerçek zamanlı bir uygulama için yetersiz olduğu ortaya çıkmaktadır. Gerçek zamanlı modern optimizasyon sistemleri için daha modern modelleme teknikleri kullanılmış ya da mevcut teknikler modifiye edilmiştir (Araghi ve ark., 2015; Jovanović ve ark., 2017; Ali ve ark., 2020; Celtek ve ark., 2020). Bu modelleme tekniklerine farklı optimizasyon teknikleri uygulanarak sinyalizasyon kavşaklarında başarılı sonuçlar elde edilmiştir (Shaikh ve ark., 2020).

Trafiğin tam anlamıyla matematiksel modellenmesi imkansızdır. Trafik, dinamik unsurlardan (yaya, sürücü, trafik rejimi, hava durumu, v.b.) meydana gelmektedir ve trafikteki parametrelerin değişimi stokastiktir (Abdoos ve ark., 2011). Modelleme tabanlı çalışmalar, trafiğin bazı parametrelerini göz ardı ederken bazı çalışmalarda ise bu parametreler yerine katsayılar kullanılmıştır (Zhao ve ark., 2011; Yau ve ark., 2017). Modellemede yapılan bu kısıtlamalar, optimizasyon algoritmalarının başarısını sınırlamaktadır.

3.1. Trafik Sinyal Kontrol Yöntemleri

Trafik sinyal kontrol sistemleri, kavşak kapasitesini ve güvenliği artırmak için bekleme süresini ve gecikme süresini azaltmayı amaçlamaktadır. İnsanların ve araçların rahat, ekonomik ve güvenli bir şekilde taşınmasını sağlamak için birçok farklı trafik ışığı kontrol sistemleri sunulmuştur.

İlk önerilen trafik sinyal denetleyicilerinden biri olan sabit zamanlı stratejiler, trafik sinyalleri için uygun faz süresini belirlemek için uygulanır. Optimal döngü ve faz süresinin belirlenmesi, sabit zamanlı yöntemler için önemli bir faktördür. Faz süresi optimum süreden kısa seçilmesi, kavşağın boşalmamasına sebep olur. Uzun faz süreleri ise, daha uzun kuyruk ve daha uzun gecikmeye neden olabilir (Araghi ve ark., 2015). Bugün birçok şehirde aktif olarak kullanılan Webster yöntemi, optimal döngü ve optimal faz sürelerini bulmak için kullanılan yöntemlerden biridir (Webster, 1958a). Sabit zamanlı sinyal kontrol yöntemlerinde, günün farklı dönemleri için değişik döngü planlaması yapılır. Bu yöntem, trafik akışının sabit olduğu durumlarda başarılı sonuçlar vermektedir (Gartner ve ark., 2002). Yapılan trafik sinyal planı anlık trafik durumuna veya genel trafik rejimine göre değişmez. Bu nedenle, trafik akışındaki beklenmedik durumlarla (kaza, yol bakım v.b.) baş edilemez (Zhou ve ark., 2017).

Trafik kontrolü için trafik ortamından bilgi alma ihtiyacı araştırmacıları trafik uyarmalı sinyal kontrolüne yönlendirmiştir. Trafik uyarmalı sinyal kontrolü geri beslemeli trafik sinyal yönetim sistemleridir. Trafik ortamından veri toplamak için endüktif dedektörler kullanılır. Elde edilen veriler, yeşil faz süresinin verimli bir şekilde uzatılması veya sona ermesi için kullanılmaktadır (Wei ve ark., 2019). Günümüzde bu tür trafik kontrol sistemlerinden olan SCOOT (Hunt ve ark., 1982) ve SCATS (Lowrie, 1990), birçok modern ülkede yaygın olarak kullanılmaktadır. Trafik uyarmalı sinyal kontrolünde döngü uzunluğu, faz bölünmeleri ve faz sırası, döngü dedektörlerinden veya diğer trafik sensörlerinden elde edilen trafik durumuna göre değiştirilebilir. Ancak yapılan bu değişiklikler önceden belirlenmiş sabit sınırlar içindedir.

Sabit zamanlı ve trafik uyarmalı gibi geleneksel yöntemler, trafik yönetiminde kontrol kurallarını veya politikalarını geliştiremez. Sabit bir süre veya önceden belirlenmiş düzenlemelerle plan yapmaya çalışırlar ve anlık trafik durumunu görmezden gelirler (Zhou ve ark., 2017; Wei ve ark., 2019). Bunun yanı sıra hava durumu, mevsim, trafiğin yoğun olduğu saatler veya özel olaylar (trafik kazası, yol çalışması v.b.) gibi trafik rejimi üzerinde oldukça etkili olan önemli parametreler vardır. Klasik trafik kontrol yaklaşımları, bu önemli parametreleri görmezden gelerek ya da varsayımlar kullanarak

(sabit sayı kullanımı gibi) çözümler üretmeye çalışmaktadırlar (Zhao ve ark., 2012). Bu varsayımlar genellikle trafik ortamındaki gerçek parametrelerden farklıdır ve kontrol yöntemleri sonuçlarını olumsuz etkilemişlerdir.

3.2. Adaptif Trafik Sinyal Kontrol Yöntemleri

Trafik parametrelerinin değişkenliği ve bu değişkenliğin doğrusal olmaması nedeniyle, araştırmacılar adaptif sinyal kontrol modellerine yönelmişlerdir (Mannion ve ark., 2016). Adaptif trafik sinyal kontrol sistemleri trafik koşullarını dikkate alarak trafik sinyallerini ayarlamak için kullanılan trafik kontrol sistemleridir. Adaptif trafik sinyal kontrol sistemlerinde trafik parametreleri (kuyruk uzunluğu, kavşağa gelen araç sayısı, araç hızı v.b.) sürekli izlenir. Trafik kontrol politikaları veya fonksiyonları, trafik ortamından gelen geri bildirimlerle güncellenir.

Adaptif trafik sinyal kontrolü, sınırlı bir sürede çözülmesi gereken birden çok optimum noktaya sahip karmaşık bir optimizasyon problemidir (Celtek ve ark., 2020). Trafik ışığı sayısı ve kullanıcı sayısı (yayalar, sürücüler, araçlar vb.) arttıkça bu optimizasyon probleminin çözümleri daha karmaşık bir kombinasyon seti oluşturur (Joo ve ark., 2020). Birçok yerel optimum noktaya sahip optimizasyon problemlerinde, klasik optimizasyon algoritmalarının performansı zayıftır ve bekleneni verememektedir (Wang ve ark., 2013).

Trafik kontrolünün karmaşık ve doğrusal olmayan yapısı nedeni ile kullanılan yöntemlerin başarı oranının sınırlı olduğu birçok çalışmada ifade edilmiş ve çözüm için yapay zeka optimizasyon algoritmaları önerilmiştir (Van de Weg ve ark., 2018). Son yıllarda birçok farklı makine öğrenim modeli adaptif trafik sinyal kontrol sistemine uygulanmıştır (Dimitriou ve ark., 2008; De Oliveira ve de Almeida Neto, 2014; Jovanović ve ark., 2017). Verimli bir makine öğrenimi tabanlı adaptif trafik sinyal kontrolünün tasarlamak için aşılması gereken ciddi engeller mevcuttur. Denetimli öğrenmeye dayalı modellerin en kritik noktası öğrenme mekanizmasıdır. Ne yazık ki, trafik sinyal kontrol sistemlerinde iyi ve kötü modelleri ayırt etmek için yeterli eğitim verisi yoktur (Wei ve ark., 2019).

Denetimli öğrenme tabanlı makine öğrenme yöntemlerinin başarısının eğitim veri setine bağlı olması, araştırmacıları eğitilmesi için veri setine bağlı olmayan yöntemlere yönlendirmiştir. Araştırmacılar, bir çok farklı sezgisel optimizasyon yöntemlerini adaptif trafik sinyal kontrolüne uygulamıştır (Garcia-Nieto ve ark., 2013;

Celtek ve ark., 2020; Shaikh ve ark., 2020). Parçacık sürü optimizasyon (PSO) algoritması kullanılarak yapılan bir çalışma ile sezgisel tabanlı bir yöntemin klasik trafik kontrol sistemlerinden daha iyi bir kontrol performansına sahip olduğu ortaya konmuştur (Zhang ve Ni, 2017). Bir başka çalışma da PSO tabanlı kontrol strateji Sevilla (İspanya) şehrine ait trafik verilerine uygulanmıştır. PSO ile elde edilen optimum döngü süreleri ile, hedeflerine zamanında ulaşan araç sayısında ve toplam yolculuk süresi üzerinde önemli iyileştirmeler sağladığı ifade edilmiştir (García-Nieto ve ark., 2012). Bir başka çalışma da arı koloni algoritması ve benzetilmiş tavlama algoritmaları trafik sinyal kontrolüne uygulanmış ve sonuçları karşılaştırılmıştır (Jovanović ve ark., 2017).

Sezgisel optimizasyon yöntemleri ile trafik sinyal ağlarının optimizasyonun başarılı sonuçlar vermesine rağmen, gerçek zamanlı bir yaklaşım modeli için hala aşılması gereken ciddi engeller vardır. Bir trafik ağının anlık verilerle optimizasyonu için gerekli hesaplama süresinin minimize edilmesi çözülmesi gereken bir problemdir (Zhang ve Zhou, 2018). Sezgisel hesaplama yöntemlerinin zaman alıcı olması ise bu problemi daha da kritik hale getirmektedir (Zhao ve ark., 2012).

Yapay optimizasyon algoritmaları için bir diğer kritik konu, trafiğin anlık olarak modellenmesidir. Modellemede yapılan kabullerden dolayı, mevcut matematiksel modellerle yapılan optimizasyon çözümlerin başarısının her zaman bir limiti olacaktır. Birçok izole edilmiş kavşağın kooperatif çalışması, büyük ölçekli trafik sinyal ağını oluşturur. Ağdaki sinyal sayısının artması, dinamik trafik koşulları ile başa çıkmayı ve trafik akışını kontrol etmeyi daha da zorlaştırır (Wang ve ark., 2020).

3.3. Takviyeli Öğrenme ile Trafik Sinyal Kontrol Sistemleri

Özellikle son yıllarda takviyeli öğrenme (Reinforcement Learning-RL) tabanlı kontrol modelleri popüler hale gelmektedir. Trafik sinyal kontrolü için RL veya RL'nin varyant algoritmaların (Q-Learning, SARSA, Deep Reinforcement Learning) başarısı yapılan birçok çalışma ile kanıtlanmıştır (El-Tantawy ve ark., 2013; Chu ve ark., 2019) (Abdoos, 2020; Aragon-Gómez ve Clempner, 2020). Model tabanlı takviyeli öğrenme yöntemleri trafik sinyal kontrolüne uygulanmış ve kavşaklarda toplam bekleme süresi en aza indirilmeye çalışılmıştır (Wiering, 2000). Ancak modelleme, araştırmacılar için hala önemli ölçüde zorlayıcıdır ve uygun olmayan modelleme, kontrol sisteminin verimini olumsuz yönde etkilemektedir. Bu nedenle, araştırmacılar modelsiz RL tabanlı kontrol yöntemlerine yönelmişlerdir (Abdoos, 2020). Modelsiz RL tabanlı kontrol yöntemlerinin

dinamik trafik rejimine uyum sağlayarak başarılı bir şekilde adaptif kontrolün sağlandığı, yapılan çalışmalarda gösterilmektedir (Ge ve ark., 2019; Abdoos, 2020).

Etkin bir trafik stratejisi hem izole edilmiş sinyalize kavşağı hem de kavşaklardan oluşan trafik ağının bütünü kontrol edebilmelidir. Büyük ölçekli trafik sinyal ağlarında, trafik sinyal problemi ilk olarak tek bir merkezi ajan tarafından yönetilmeye çalışılmıştır (Arel ve ark., 2010; Van der Pol ve Oliehoek, 2016). Merkezi yönetim yaklaşımlarında, ağdaki trafik performans verileri durum olarak toplanır ve kontrol merkezine gönderilir (Casas, 2017). Merkezci yönetim trafik ağını optimize ederken yerel ve global tüm gözlemleri dikkate alır. Bu sayede yapılan optimizasyon işlemi yerelle sınırlı kalmaz. Ancak, merkezci yönetimde ağdaki kavşak sayısı arttıkça, durum uzayı ve eylem alanı büyük ölçüde artacaktır (boyut laneti problemi). Optimizasyon kararının sonucu trafik sinyali denetleyicisine geri döndürüldüğünde trafik akışı zaten büyük ölçüde değişmiş olabilir. Bu durumda, trafik ışığı için yapılan optimizasyon işlemi, mevcut trafik durumu için uygun değildir. Merkezi bir yöntem çok ağır hesaplama ve iletişim yükü gerektirir ki bu yüksek gecikme ve başarısızlığa yol açabilir (Wang ve ark., 2020).

Merkezci yönetimin alternatifi, dağıtılmış yönetim şeklidir. Çoklu ajan takviyeli öğrenme (Multi agent reinforcement learning, MARL), her bir ajanın farklı birer sinyalize kavşağı temsil ettiği dağıtılmış kontrol yöntemidir (Balaji ve Srinivasan, 2010). MARL’da, bir kavşakta olan ajan (trafik sinyal kontrolörü) ile çevre (sinyalize kavşak) arasındaki dinamik etkileşimin diğer kavşaklara bir etkisi vardır. Bir kavşaktaki etkileşim diğer kavşaktaki adaptif karar verme sürecini önemli ölçüde etkiler. Bu etkiyi minimize etmek ve engellemek için ajanların bağımsız bir şekilde öğrenmesi önerilmiştir (Liu ve ark., 2017). Bağımsız öğrenen ajanlar, kendi ödülleri yalnızca yerel gözlemlere göre optimize eder, global kriterler için optimizasyon yapamaz. Bu durumda yapılan iyileştirmeler yerelle sınırlanmaktadır ve trafik ağının geneli optimize edilememektedir.

Özetlemek gerekirse son yıllarda, yapay sinir ağları tabanlı önerilen yöntemlerin sayısı gittikçe azalmakta ve takviyeli öğrenme tabanlı yöntemlerin sayısı artmaktadır. Takviyeli öğrenmeye olan yönelimin motivasyonları şu şekildedir;

- Takviyeli öğrenme yöntemi, dinamik bir ortamda en iyi eylemi belirlemek için birçok farklı disiplinde önerilmiş ve başarılı sonuçlar elde edilmiştir (Clune ve ark., 2020; Goldwasser ve Thielscher, 2020). Trafik sinyal kontrolü son derece dinamik ve durağan olmayan bir mühendislik problemidir (Abdoos ve ark., 2011). Takviyeli öğrenmenin dinamik mühendislik problemlerindeki başarısı dikkate alındığında trafik sinyal kontrolünün de başarılı olacağı öngörülmektedir.

- Trafik ortamının modellenmesi zorlu bir iştir çünkü trafik parametreleri stokastiktir. Birçok yapay zeka tabanlı yöntemin aksine, takviyeli öğrenme modellemeye ihtiyaç duymaz (Joo ve ark., 2020).
- Sistem performans ölçüsü ödül olarak belirlenirse, ortam hakkında herhangi bir ön bilgiye ihtiyaç yoktur. Takviyeli öğrenme ajanları, çevre hakkında önceden bilgi sahibi olmadan kendi başlarına öğrenebilirler.
- Takviyeli öğrenme ajanları, trafik kazaları veya kötü hava koşulları gibi farklı stokastik koşullara uyum sağlayabilir (Yau ve ark., 2017).

3.4. Trafik Sinyal Kontrol Sistemlerinde Kullanılan Takviyeli Öğrenme Bileşenleri

Trafik sinyali programlama problemlerine takviyeli öğrenme tabanlı bir çözüm uygulamak için durum, eylem, ödül ve çevre bileşenlerinin tasarlanması gereklidir. Takviyeli öğrenme algoritmasında eylem, öğrenme üzerinde kritik bir etkiye sahiptir (El-Tantawy ve ark., 2013). Literatürde temel olarak dört tür eylem vardır: “mevcut faz süresini ayarla”, “faz ayrımını ayarla”, “mevcut fazı koru veya değiştir” ve “sonraki fazı seç” (Wei ve ark., 2019). “Geçerli faz süresini ayarla” eylemi, trafik sinyali kontrolü için sık kullanılan eylem türlerinden biridir (Abdoos ve ark., 2011; Abdoos, 2020). Bu, algoritmanın önceden tanımlanmış bir zaman setinden seçim yaparak mevcut faz süresini ayarlamayı öğrendiği anlamına gelir. “Faz ayrımını ayarla” eylemi, faz bölme oranı olarak tanımlanabilir. Ajan genellikle bir sonraki döngü için önceden tanımlanmış faz oranını seçer (Chin ve ark., 2012; Genders ve Razavi, 2019; Liang ve ark., 2019). “Mevcut fazı koru veya değiştir” eylemi, mevcut fazı koruma veya bir sonraki faza geçme kararını veren bir eylemdir (Aslani ve ark., 2018; Tan ve ark., 2019; Haydari ve Yilmaz, 2020). “Sonraki aşamayı seç” eylemi, bir temsilcinin aşama sırasına karar verebileceği en esnek eylemdir. Bu eylem, uzun yıllar boyunca takviyeli öğrenme tabanlı trafik kontrol uygulamaları için başarıyla kullanılmıştır (Gokulan ve Srinivasan, 2010; Xu ve ark., 2013; Aslani ve ark., 2018; Wei ve ark., 2018; Wei ve ark., 2019).

Eylem süresi, tüm eylem türleri için önemli bir faktördür. Gerçek trafik ortamı dinamiktir ve eylem kararı alınmadan önce gözlem gerektirir. Düşük eylem süresi ile daha fazla geri bildirim mümkündür. Ancak düşük eylem süresi, fazlar arasında daha fazla geçişe neden olur. Her geçiş, trafik sinyal kuralları gereği döngü süresini artıran sarı sinyal süresini gerektirir (HCM, 2000). Uzun eylem süresi daha az fazlar arası geçişe neden olur ve toplam sarı sinyal süresini azaltır. Sarı sinyal süresinden kaynaklanan trafik

gecikmesini ve bekleme sürelerini de minimuma indirir. Öte yandan, takviyeli öğrenme ajanı, uzun bir eylem süreli uygulamada trafik parametrelerindeki değişikliği yakalayamayabilir. Eylem süresinin önemli bir faktör olmasına rağmen, literatürdeki çalışmaların çoğu sabit bir süre kullanmakta ve optimum süre arayışına girmemektedir. Bunun nedeni de literatürde yapılan çalışmaların çoğunun tasarım olarak önerilmesi ve önerilen tasarımlara henüz iyileştirme yapılmamasıdır.

Bu çalışma da proje de "Mevcut fazı koru veya değiştir" ve "sonraki fazı seç" eylemleri için optimum eylem süreleri araştırılmıştır. İlk olarak sabit süreli uygulamalar simülasyon ortamında gerçek zamanlı olarak gerçekleştirilmiştir. Bu çalışma ile kısa eylem süreleri ile uzun eylem sürelerinin trafik performanslarının üzerindeki etkileri araştırılmıştır. Çalışma çıktıları hem izole kavşak hem koordineli kavşaklar hem de trafik ağı açısından ayrı ayrı incelenmiştir.

3.5. Nesnelerin İnterneti ve Trafik Sinyal Kontrolü

Teknolojide son gelişmeler, kontrol sistemlerinin veri algılama ve veri toplama yeteneklerini geliştirmektedir. Bu durum değişken trafik parametrelerinin daha hassas algılanmasını sağlar ve gerçek zamanlı trafik veri akışını mümkün kılar. Etkin bir trafik sinyal kontrol yönteminin, toplanan trafik verilerine dayalı olarak faz zamanlamasını verimli bir şekilde optimize etmesi gereklidir. Ek olarak, etkin bir trafik sinyal kontrol stratejisinin, veri iletimi ve hesaplama süresinden kaynaklanan uzun karar vermesinin ve yanıt gecikmesinin üstesinden gelmesi gerekir.

Yukarıda bahsi geçen sorunların üstesinden gelmek için nesnelerin interneti ve bilişim yöntemleri (bulut, sis ve kenar bilişim) potansiyel bir çözümdür (Zhang ve Zhou, 2018; Tang ve ark., 2019; Yang, 2020). Bulut bilişim, sis bilişim ve kenar bilişim gelecekteki birçok ulaşım sorununa çözüm olabilecek paradigmalardır. Özellikle son on yılda, bu paradigmlar ile ulaşım sorunlarına birçok çözümler önerilmiştir (Huang ve ark., 2017; Liu ve ark., 2018; Tang ve ark., 2018). Tüm bu çalışmalar, trafik sinyal kontrolü yerine esas olarak veri aktarımına veya veri toplamaya odaklanmaktadır. Yapılan bir çalışma da sis hesaplama ve genetik algoritma tabanlı trafik sinyal kontrolü üzerinde çalışılmıştır. Çalışma sonuçları, sis hesaplamanın trafik sinyali kontrol uygulamaları için potansiyele sahip olduğunu açıkça göstermektedir (Tang ve ark., 2019).

Bölüm 3'te literatür araştırmalarına yer verilmiştir. İlk olarak klasik kontrol sistemleri anlatılmış ve ardından adaptif trafik sinyal kontrol sistemleri tanıtılmıştır.

Takviyeli öğrenme tabanlı çalışmalar ve bu çalışmalarda kullanılan takviyeli öğrenme kavramları açıklanmıştır. Son olarak trafik sinyal ağında kontrol ve koordinasyonun sağlanması için önerilen nesnelerin interneti ile ilgili yapılan çalışmalara yer verilmiştir.



4. MATERYAL VE YÖNTEM

Bu bölümde, bu çalışmada kullanılan simülasyon programı ve trafik sinyal optimizasyonu için kullanılan yapay optimizasyon teknikleri tanıtılmıştır. Sürü tabanlı optimizasyon yöntemleri olarak, sırasıyla parçacık sürü optimizasyon algoritması, sosyal öğrenme parçacık sürü optimizasyon algoritması ve edimsel koşullanma ile sosyal öğrenme parçacık sürü optimizasyon algoritması açıklanmıştır. Ardından takviyeli öğrenme yöntemleri ve varyant teknikleri tanıtılmıştır.

4.1. Simülasyon Programı

Son yıllarda, önerilen ve geliştirilen trafik sinyal kontrol yöntemlerinin uygulamalarının sayısının artması, bu uygulamaların test edileceği ve değerlendirileceği platformlara olan ihtiyacı da arttırmıştır. Bu amaç doğrultusunda trafik simülatörleri, trafik sinyal kontrol yöntemlerinin tasarımında ve değerlendirilmesinde yaygın olarak kullanılmaktadır. Trafik simülatörlerinin trafik araştırmalarında önemli bir rol oynamaktadır. Bunun başlıca nedenleri şöyle sıralanmıştır;

- Önerilen trafik stratejilerinin gerçek dünyada test edilmesi ve değerlendirilmesi maliyetli ve zor bir işlemdir,
- Yapılan birçok trafik sinyal kontrol uygulamalarında trafiği oluşturan bazı dinamik parametreler göz ardı edilmiş ya da bir sabit olarak değerlendirilmiştir. Bunun gibi birçok kısıtlama veya kabullerin trafik yönetimine etkisi simülatör yardımıyla tespit edilebilir,
- Önerilen yöntemin doğruluğu ve verimi ucuz bir şekilde tespit edilebilir,
- Simülatörler sadece değerlendirme aracı değil, modelleme ve planlama aracı olarak da kullanılabilir (Kotusevski ve Hawick, 2009).

Trafik simülatörleri temsil ettiği detay seviyesine göre mikroskobik, mezoskopik ve makroskobik simülatörler olmak üzere üç kategoriye ayrılır. Makroskopik simülatörler, trafik ağını oluşturan küçük bölümleri (sokak, cadde, tali yol v.b.) dikkate almadan, trafiği bir bütün olarak ele alır. Doğrudan genel ağın kümülatif ya da ortalama değerlerini hesaplar. Genel trafik akış analizinde kullanılır. Mezoskopik yaklaşımda ise, her bir araç dikkate alınmazken genel etkileşim dikkate alınır. Mezoskopik yaklaşımlar orta düzeyde bir ayrıntı seviyesine sahiptir. Mikroskopik simülatörlerde ise daha detaylı bir analiz söz konusudur. Her bir aracın yol ya da mahalle ile etkileşimi göz önüne alınarak hesaplama yapılır (Pell ve ark., 2017).

4.1.1. SUMO Simülatörü

SUMO (Simulation of Urban Mobility- Kentsel Hareketlilik Simülatörü), yol ağlarını kontrol etmek için oluşturulan açık kaynaklı mikroskopik simülasyon yazılımıdır. Araba, otobüs, kamyon, bisiklet ve tren gibi kara yolunda bulunan tüm dinamik parametreleri dikkate alan çok yönlü (multimodal) bir simülasyon paketidir.

```
<configuration>

  <input>
    <net-file value="intersection.net.xml"/>
    <route-files value="intersection.rou.xml"/>
  </input>

  <time>
    <begin value="0"/>
  </time>

  <report>
    <verbose value="true"/>
    <no-step-log value="true"/>
  </report>
  <time-to-teleport value="-1"/>
</configuration>
```

Şekil 4.1. Sumo konfigürasyon dosyasına ait görsel.

SUMO kullanılarak oluşturulan simülasyonun, üç ana bileşeni vardır. Bu üç bileşen yol, rota ve konfigürasyon dosyalarıdır. Yol dosyaları trafik ağına ait teknik (trafik ışığı, şerit bilgisi, trafiğe kapalı yol, koordinat ve coğrafi) bilgileri içerir ve simülasyonda mevcut karayolunu temsil eder. Yol dosyası SUMO fonksiyonları kullanılarak oluşturulabilir. Ayrıca OPEN STREET MAP aracılığıyla istenen yol, cadde veya bölgeye ait gerçek trafik senaryosu indirilerek SUMO'da simülasyon gerçekleştirilebilir. Rota dosyaları ise karayolunu kullanan araçların karayolu üzerindeki güzergahlarını temsil eder. Rota dosyaları mevcut başlangıç ve varış matrisleri (koordinat bilgilerini içeren) kullanılarak ya da manuel olarak oluşturulabilir. Konfigürasyon dosyası ise yol ve rota dosyalarını birleştirmek için kullanılır. Yol dosyaları gerçek dünya bilgilerini içerirken,

rota bileşeni ise kullanıcın kendi isteklerini ve yönlendirmelerini bulundurur. Konfigürasyon dosyasına ait görsel Şekil 4.1’de verilmiştir.

SUMO_GUI kullanılarak yapılması istenen simülasyon görselleştirilebilir. Şekil 4.2’de yapılan görsel bir simülasyonun görüntüsü verilmiştir. SUMO’da her simülasyon işlemi için birçok ölçüm yapılabilir. Konum, ortalama hız, araç hızı, araç bekleme süresi gibi değerlerin bireysel, kümülatif ya da ortalama değerleri çıktı olarak elde edilebilir. Bunların haricinde gürültü, gaz emisyonu, yakıt tüketimi gibi değerlere de ulaşılabilir.



Şekil 4.2. SUMO_GUI ile simülasyonun görselleştirilmesi.

4.2. Sürü Tabanlı Optimizasyon Algoritmaları

4.2.1. Parçacık Sürü Algoritması

Parçacık sürü optimizasyon algoritması (PSO) , 1995 yılında Kennedy ve Eberhart tarafından; kuş sürülerinin davranışlarından esinlenilerek ortaya çıkarılmış popülasyon tabanlı optimizasyon tekniğidir (Karaboğa, 2014) (Eberhart ve Kennedy, 1995).

PSO'da sürüdeki her parçacığın, sürünün global ve yerel en iyi değerlerini kullanarak konumunu ve hızını güncelleyebileceği varsayılmaktadır. PSO, optimum ya da optimuma yakın çözüm bulmak için önce her biri aday çözümü sunan parçacıklar oluşturur. Bir sürü D -boyutlu arama uzayında hareket eden N tane parçacıktan oluşur. i 'nci parçacığın t anındaki pozisyonu herhangi bir arama ya da optimizasyon problemi için parçacığın çözümünü değerlendirmede kullanılır ve aday çözüm ya da çözümleri temsil eder. $x_i(t) = (x_{i1}, x_{i2}, \dots, x_{iD})$ ile temsil edilir. Burada $x_{i,n}(t) \in [l_n, u_n]$, $1 \leq n \leq N$ l_n ve u_n sırayla n . boyutun alt ve üst sınırlarını gösterir. PSO, bireyler arasındaki bilginin paylaşımını esas alır. Her bir parçacık arama boyunca kendi pozisyonunu sürüdeki en iyi pozisyona doğru ayarlarken, bir önceki tecrübesinden de yararlanır. Yani pozisyonunu iki faktöre göre ayarlar: $p_{i,j} = (p_{i1}, p_{i2}, \dots, p_{iD})$ şeklinde gösterilen kişisel en iyi pozisyonu (p_{best}) ve $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$ şeklinde gösterilen tüm sürüdeki en iyi pozisyon (g_{best}). t . iterasyondaki hızı $v_i(t) = (v_{i1}, v_{i2}, \dots, v_{iD})$ ile gösterilir ve $v_{imax} = (v_{imax1}, v_{imax2}, \dots, v_{imaxD})$ ile sınırlandırılır.

Geleneksel bir PSO için hız ve konum denklemleri sırasıyla Denklem 4.1 ve Denklem 4.2 ile hesaplanır;

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1r_{1,j}(t)(p_{best_{i,j}}(t) - x_{i,j}(t)) + c_2r_{2,j}(t)(g_{best_i}(t) - x_{i,j}(t)) \quad (4.1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (4.2)$$

r_1 ve r_2 , PSO'nun stokastik doğası için kullanılan $[0, 1]$ aralığında değer alan rastsal parametrelerdir. Hızlanma katsayıları olarak adlandırılan c_1 ve c_2 katsayıları ($0 < c_1, c_2 \leq 2$) öğrenme faktörleridir. Denklem 4.1'de geçen $w \in [0.8, 1.2]$ atalet ağırlığıdır. PSO'da atalet ağırlığı global ve yerel arama yeteneğini dengelemek için kullanılır. Atalet ağırlığının yüksek seçimi global aramayı (daha hızla yakınsama), küçük değerlerde seçimi ise yerel arama yapılmasını (daha hassas yaklaşım) kolaylaştırır. Atalet ağırlığı yerel ve global arama arasındaki dengeyi sağlar ve bunun sonucunda yeterli optimal sonuca daha az iterasyonla ulaşılır. Bu yüzden uygun atalet ağırlığı değerinin belirlenmesi önem arz etmektedir.

4.2.1. Sosyal Öğrenme Parçacık Sürü Algoritması

Optimizasyon probleminin boyutu arttıkça PSO'nun performansı düşmektedir. PSO'nun yüksek boyutlu problemlerde arama performansını iyileştirmek için birçok yöntem önerilmiştir. Sosyal öğrenme parçacık sürü algoritması (SL-PSO) bu amaç doğrultusunda önerilen PSO'nun varyantlarından biridir.

Klasik PSO gibi, önerilen SL-PSO yaklaşımında da, ilk olarak optimizasyon problemine aday bir çözüm oluşturan rasgele başlatılmış bir başlangıç vektörü ($X_{i,j}$) oluşturulur. Sürü daha sonra parçacıkların uygunluk değerlerinin artan sırasına göre sıralanır. Sonuç olarak, (aynı sosyal öğrenme mekanizmasında olduğu gibi) her parçacık diğer parçacıklardan öğrenerek davranışlarını düzeltmektedir. Parçacıklar arası sosyal öğrenme, Denklem 4.3 ile gerçekleşir:

$$X_{i,j}(t+1) = \begin{cases} X_{i,j}(t) + \Delta X_{i,j}(t+1), & \text{eğer } p_i(t) \leq P_i^L \\ X_{i,j}(t), & \text{aksi durumda} \end{cases} \quad (4.3)$$

Denklem 4.3'te $X_{i,j}(t+1)$ öğrenen, $X_{i,j}(t)$ öğreten parçacık, PL'de öğrenme olasılığıdır. $\Delta X_{i,j}(t+1)$ Denklem 4.4 ile hesaplanır;

$$\Delta X_{i,j}(t+1) = r_1(t)\Delta X_{i,j}(t) + r_2(t)I_{i,j}(t) + r_3(t)C_{i,j}(t) \quad (4.4)$$

$$I_{i,j}(t) = X_{k,j}(t) - X_{i,j}(t) \quad (4.5)$$

$$C_{i,j}(t) = \bar{X}_j(t) - X_{i,j}(t) \quad (4.6)$$

Sosyal öğrenmeden esinlenen SL-PSO'da davranış güncelleme operatörü olan $\Delta X_{i,j}(t+1)$ üç bileşenden meydana gelmektedir. Birinci bileşen olan $X_{i,j}(t)$ yerel ile global arama arasındaki dengeyi sağlamaktadır. İkinci bileşen olan $I_{i,j}(t)$ gösterici (demonstrator) olarak adlandırılır (Denklem 4.5). SL-PSO konum güncelleme için PSO'da kullanılan g_{best} veya p_{best} yerine göstericiler kullanır. Üçüncü bileşen $C_{i,j}(t)$, sosyal etki faktörü olarak adlandırılır ve parçacığın sürünün içindeki konumunu belirler (Denklem 4.6). Denklem 4.4'te kullanılan r_1 , r_2 ve r_3 parametreleri ise bu üç bileşenin ağırlığını belirler.

Arama boyutu ne kadar yüksek olursa, sorunu çözmek o kadar zor olur ve bir parçacığın diğerlerinden öğrenmeye istekli olma olasılığı o kadar düşük olur. SL-PSO'da, öğrenme olasılığı ve problem boyutu Denklem 4.7'de gösterildiği gibi ters orantılıdır.

$$P_i^L = \left(1 - \frac{i-1}{m}\right)^{\alpha \log\left(\frac{m}{|M|}\right)} \quad (4.7)$$

Denklem 4.7’de α $[0,1]$ aralığında rastsal seçilen bir sayıyı, M problem boyutunu, m ise sürü boyutunu göstermektedir.

4.2.3. Edimsel Koşullanma ile Sosyal Öğrenme Parçacık Sürü Algoritması

Bu tez çalışmasında kullanılan sürü algoritmalarının yakınsama hızını arttırmak için edimsel koşullandırma yönteminden esinlenen bir yaklaşım önerilmiştir (OCA-SL-PSO) (Çeltek ve Durdu, 2020). Önerilen yaklaşım sosyal öğrenme parçacık sürüsü optimizasyonuna (SL-PSO) uygulanmıştır.

Klasik SL-PSO’da sürü parçacıkları amaç işlevine göre sıralanır ve her bir parçacık için rastsal bir eğitim işlemi gerçekleştirilir. Önerilen yaklaşım ile, her bir parçacığın öğrenme oranı, deterministik olarak matematiksel olarak hesaplanır. Önerilen yaklaşım, her parçacık için öğrenme oranını ayarlamaktadır. Öğrenme oranı kullanılarak, optimum çözüme yakın bir parçacığın daha az öğrenmesi amaçlanmaktadır. Öğrenme oranı sayesinde bir parçacığın optimum çözüme yakın parçacık ile optimum çözüme uzak parçacıklardan aynı oranda etkilenmesinin önüne geçilmektedir.

Bu çalışmada öğrenme oranı, sosyal hayatta kullanılan edimsel koşullanmadan esinlenmiştir. Edimsel koşullanmada, bireyin önceden deneyimlediği her durum için, ortaya pozitif veya negatif sonuçlar çıkmaktadır. Bu olumlu ya da olumsuz sonuçlar, geçmişteki deneyimlerden kaynaklanır ve bireyin ilerideki davranışlarında büyük ölçüde etkili olmaktadır. Pozitif veya negatif sonuçların, bir bireyin davranışlarında yol açtığı değişikliklere “Edimsel Koşullanma” denir. Eğer birey daha önce bir davranışta bulunmuş ve bu davranışın sonucunda olumlu bir şekilde sonuç almışsa, tekrar aynı davranışta bulunmaya yönelmektedir. Aynı şekilde, eğer birey daha önce bir davranışta bulunmuş ve bu davranışın sonucunda olumsuz bir tepki almışsa, bir daha benzer bir davranışta bulunmamaya gayret etmektedir.

Bu çalışmadaki yaklaşımda edimsel koşullanma birey için değil sürü için uygulanmıştır. Bu sayede sürüyü oluşturan tüm bireylerin deneyimlerinden yararlanma imkânı doğmuştur. Bu çalışmadaki öğrenme olasılığı parçacığın en iyiye olan uzaklığı ve en kötüye olan uzaklığına bağlı bir fonksiyon ile tespit edilir (Denklem 4.8 ve Denklem 4.9). Parçacık iyi bir sonuç vermişse iki durum söz konusudur; birinci durumda parçacık daha iyi parçacıkları taklit ederek daha iyiye ulaşmalıdır. İkinci durumda ise parçacığın verdiği sonucun lokal değil yerel nokta olmasıdır ki bu da kötü parçacıkları taklit etmesiyle aşilmektedir.

$$P_{best}^L = \left(\frac{|Xi - X_{best}|}{|X_{best} - X_{worst}|} \right)^{\log\left(\frac{m}{M}\right)} \quad (4.8)$$

$$P_{worst}^L = \left(\frac{|Xi - X_{worst}|}{|X_{best} - X_{worst}|} \right)^{\log\left(\frac{m}{M}\right)} \quad (4.9)$$

P_{best}^L X , parçağın optimum çözüme yakın olan parçacıklardan öğrenmesini temsil etmekte iken, P_{worst}^L ise çözümden uzak parçacıklardan öğrenmeyi temsil eder.

4.2.4. Sürü Algoritmalarının Test Edilmesi

Sezgisel optimizasyon yöntemlerinin bir probleme uygulanması için yazılan kodun doğruluğunun test edilmesi gerekmektedir. Çalışmada kullanılan sezgisel yöntemler CEC 13 fonksiyonlarında SL-PSO ve klasik PSO ile karşılaştırılmıştır. Çizelge 4.1’de kullanılan CEC 13 fonksiyonlarına yer verilmiştir.

Çizelge 4.1. Kullanılan CEC 13 Fonksiyonları

No	Fonksiyon Adı
f1	Sphere Fonksiyonu
f2	Rotated High Conditioned Elliptic Fonksiyonu
f3	Rotated Bent Cigar Fonksiyonu
f4	Rotated Discus Fonksiyonu
f5	Different Powers Fonksiyonu
f6	Rotated Rosenbrock’s Fonksiyonu
f7	Rotated Schaffers F7 Fonksiyonu
f8	Rotated Ackley’s Fonksiyonu
f9	Rotated Weierstrass Fonksiyonu
f10	Rotated Griewank’s Fonksiyonu
f11	Rastrign’s Fonksiyonus
f12	Rotated Rastrign’s Fonksiyonu
f13	Non-Contious Rotated Rastrign Fonksiyonu

Çizelge 4.2’de OCA-SL-PSO ile SL-PSO ve saf PSO arasındaki karşılaştırma verilmiştir. Çizelgedeki değerlerin elde edilmesi için her bir yöntem 10 kez çalıştırılmış

ve elde edilen en iyi değerlerin ortalaması alınmıştır. Elde edilen PSO sonuçları literatürde elde edilen sonuçlardan daha iyidir (Zambrano-Bigiarini ve ark., 2013). SL-PSO sonuçları ise literatürdeki sonuçlar ile birebir aynıdır (Cheng ve Jin, 2015).

Çizelge 4.2. OCA-SL-PSO'nun Klasik yöntemlerle karşılaştırılması

	OCA-SL-PSO (n=100)	OCA-SL-PSO (n=500)	SL-PSO (n=50)	PSO (n=50)
f1	0,00E+00	5,88E-27	7,19E-13	-1,400E+3
f2	6,27E-229	9,59E-01	5,90E+5	3,776E+05
f3	9,18E+00	4,46E+05	2,13E+07	1,99E+07
f4	3,21E-36	5,96E+01	3,11E+6	3,113E+04
f5	9,57E+01	1,07E+03	6,23E-11	-1,000E+03
f6	0,00E+00	1,00E+01	2,33E0	-8,816E+02
f7	5,72E-02	8,06E-01	1,51E0	-7,439E+02
f8	5,30E+03	3,84E+04	3,44E-02	-6,790E+02
f9	7,96E+01	1,46E+03	1,75E0	-5,548E+02
f10	1,33E-14	9,83E-01	1,06E-1	-4,999E+02
f11	0,00E+00	6,66E-16	5,87E0	-2,498E+02
f12	4,71E-33	2,89E+00	2,11E+1	-1,378E+02
f13	1,35E-32	7,79E-23	2,45E+1	1,196E+02

Ayrıca Çizelge 4.2'ye göre önerilen yaklaşım, hem standart PSO'dan hem de SL-PSO'dan daha başarılı sonuçlar vermektedir. 13 test fonksiyonunun 10'unda 100 boyutlu edimsel koşullanma en başarılı sonucu üretirken klasik yöntemleri geride bırakmaktadır.

4.3. Takviyeli Öğrenme

Takviyeli öğrenme (Reinforcement Learning-RL-TÖ), deneme yanılma üzerine kurulu bir makine öğrenmesi tekniğidir. TÖ, Markov Karar Süreçleri stratejini kullanır. Markov Karar Süreçleri aksiyonların sadece bir sonraki ödülü değil gelecek durumları da etkilediği sıralı karar verme sürecini modelleyen klasik bir formüldür.

Takviyeli öğrenmede ajan olarak adlandırılan eylemleri yerine getiren birim karşılaştığı durumlara göre bir aksiyon alır. Alınan her bir aksiyonun karşılığında da sayısal bir ödül alır. Ajan farklı farklı aksiyonları deneyerek aldığı bu ödül puanını maksimuma çıkartmak için çalışır (Sutton ve Barto, 2018).

Takviyeli öğrenme tekniğinin temel bileşenleri şu şekilde tanımlanabilir;

- **Çevre:** Ajanın iletişim halinde olduğu dış dünyadır. Çevrenin bir simülasyonu olup ajanın bir aksiyonu gerçekleştirmeden önce bu aksiyon sonucunda alabileceği ödülü ve doğuracağı durumu tahmin etmesini sağlamaktadır.
- **Ajan:** Takviyeli öğrenmede, öğrenme faaliyetini gerçekleştirip, eylemleri gerçekleştiren temel öğedir. Bir ajanının amacı, uzun vadede aldığı ödülleri maksimum seviyeye ulaştırmaktır.
- **Aksiyon (A):** Ajanın çevreden gelen durum bilgisine göre aldığı karardır.
- **Ödül (R):** Ajanının gerçekleştirmiş olduğu bir aksiyona karşılık çevreden aldığı puandır. Ödül alınan eylemin ne kadar iyi veya kötü olduğunu belirleyen değerdir. Takviye öğrenmede ödül, ortamın aksiyona tepkisi ya da ortamdaki geri bildirim olarak değerlendirilir. Ajan, eylemin sonucunu anlamak ve ortam modeli geliştirmek için ödülü kullanır. Ödül, öğrenme sürecinin en önemli bir bileşenidir. Trafik sinyal kontrolünde amaç, trafik kapasitesini arttırmaktır. Bu hedefe ulaşmak için ödül, trafik verimliliğinin bazı performans ölçülerinden (gecikme, bekleme süresi, ortalama hız, kavşaktan geçen araç sayısı gibi) türetilmelidir. Böylece ajan, alınan eylemin kavşak verimliliğini azaltıp azaltmadığını veya artırıp artırmadığını anlayabilir.
- **Durum değeri (S):** Ajanın algılayabildiklerinin temsilidir. Ajanın çevreyi her bir adımda algıladığı değeri temsil eder.
- **Politika (P):** Ajanın herhangi bir zaman diliminde yapabileceği aksiyonu belirler.

4.3.1. Q-Öğrenme Algoritması

Q-Öğrenme (Q-Learning, QÖ) modelsiz takviyeli öğrenme tekniğidir (Jang ve ark., 2019). Karmaşık ve stokastik çevreyi keşfetmek için bir deneme yanılma yaklaşımı kullanır. Q-Öğrenme, negatif veya pozitif ödüller ile durum-aksiyon çiftini kullanarak bir öğrenme süreci oluşturmayı hedeflemektedir.

Q-Öğrenmede, ajanın içinde bulunduğu durumları ve ajanın hareketlerini kayıt eden bir tablo kullanılır. Öğrenme işleminin her bir adımında tablodaki Q değerleri Denklem 4.10'daki gibi güncellenmektedir. Ajan bu tabloyu kullanarak en uygun hareketi seçer. Burada $Q_{(s,a)}^{t+1}$ yeni Q değerini, $Q_{(s,a)}$ eski Q değerini, $R_{(s,a)}$ ödül değerini, $\max Q_{(s,a)}$, α öğrenme katsayısını, durum eylem çiftinin alabileceği maksimum Q değerini ve γ ise azaltma faktörünü temsil etmektedir.

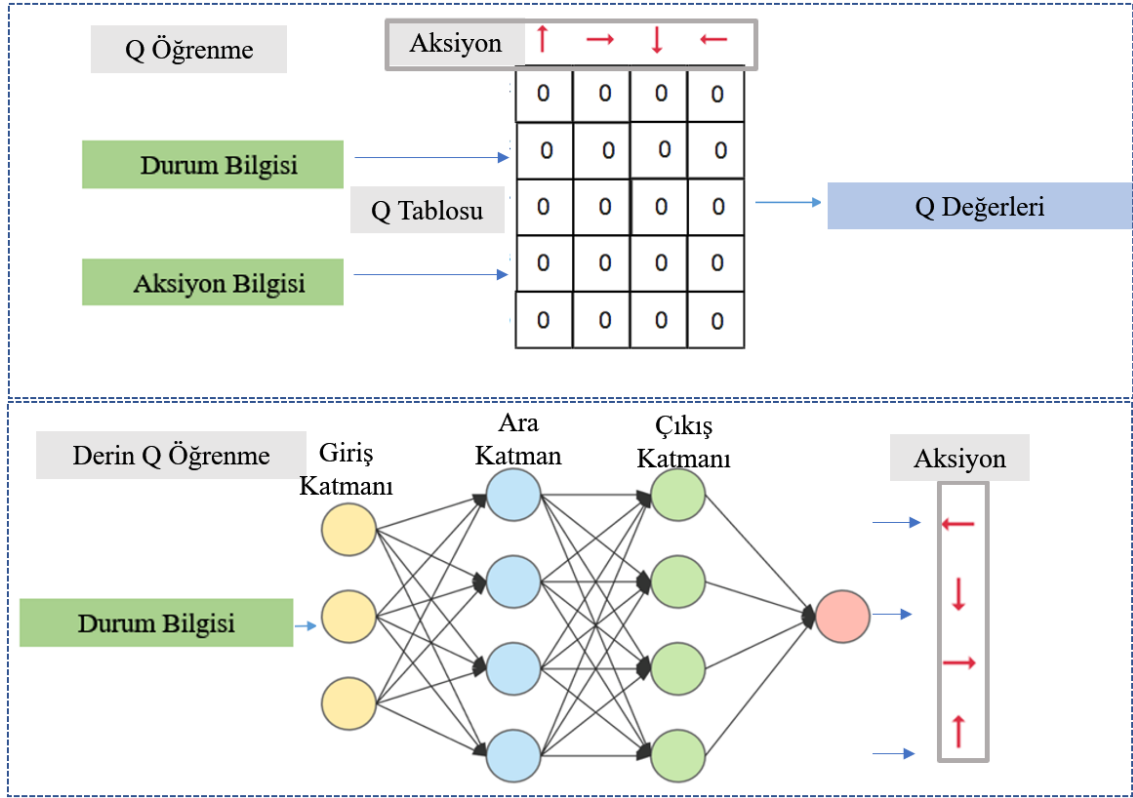
$$Q_{(s,a)}^{t+1} = [Q_{(s,a)} + \alpha \cdot R_{(s,a)} + \gamma \cdot \max Q_{(s,a)} - Q_{(s,a)}] \quad (4.10)$$

4.3.2. Derin Q-Öğrenme Algoritması

Takviyeli Öğrenme ile Derin Öğrenmenin birlikte kullanılması ile ortaya çıkan Derin Q-Öğrenme (Deep Q-Learning, DQÖ), eylemler için en uygun Q-değerleri bulmak amacıyla fonksiyon yaklaşırıcısı (function approximator) olarak derin sinir ağlarını kullanmaktadır.

Derin öğrenme yöntemlerinin kullanılmasıyla beraber Q fonksiyonu ile elde edilen değerler sinir ağları ile çözülmektedir. Takviyeli öğrenme ve derin yapay sinir ağlarını birleştirmenin temel motivasyonlarından biri, takviyeli öğrenmede bulunan büyük boyuttaki durumlar uzayını derin yapay sinir ağlarının çözebilme potansiyelidir.

DQÖ'nün çalışma prensibi QÖ ile karşılaştırılarak Şekil 4.3'te görsel olarak açıklanmıştır. QÖ'de durum-aksiyon çifti kullanılarak bir Q tablosu oluşturulur. Ajan Q tablosuna göre yapacağı eylemi belirler. DQÖ'de ise eylemin belirlenmesi için derin sinir ağları kullanılır. Ortamdan elde edilen durum bilgileri giriş katmanı olarak alınır. Giriş katman sayısı ortamdan elde edilen durum bilgi sayısı kadardır. Çıkış katmanı ise ajanın alabileceği aksiyonlara göre belirlenir. Her bir çıkış katmanı bir aksiyonu temsil etmektedir. Ara katman sayısı ve ara katmanda kullanılan nöron sayısı da derin öğrenme uygulamalarında olduğu gibi rastsal olarak belirlenmektedir (Fan ve ark., 2020).



Şekil 4.3. DQÖ çalışma prensibi

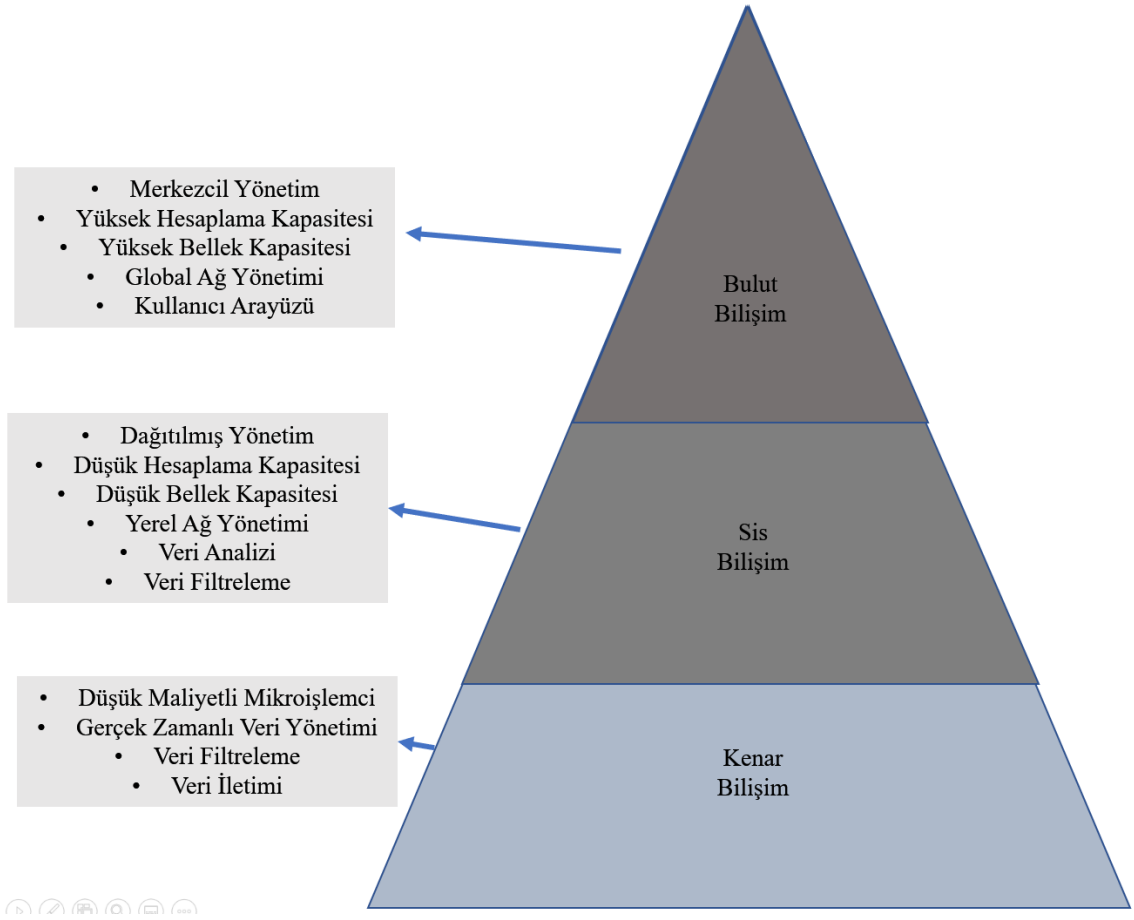
4.4. Nesnelerin İnterneti Kavramı

Haberleşmede kullanılan elektronik komponentlerin boyutlarının mikro düzeye indirilmesi ve maliyetlerinin düşürülmesi ile birlikte günümüzde her alanda hızlı bir değişim gerçekleşmektedir. Sağlıktan enerjiye, tarımdan gıda sektörüne kadar birçok farklı alanda dijital dönüşüm başlamış ve bu dönüşüm artan bir hızla devam etmektedir (Çeltek ve ark., 2015; Çeltek ve ark., 2017; Durgun ve ark., 2017). Öte yandan dünya nüfusunda aktif internet kullanımı (dünya nüfusunun yaklaşık %50'si (Khalil ve Özdemir, 2018) sürekli artmakta ve bu artış dijital dönüşümü daha da ivmelendirmektedir.

Kavram olarak nesnelerin interneti (Internet of Things-IoT), fiziksel nesnelerin dijitalleştirildiği küresel bir altyapıdır. Nesnelerin interneti, benzersiz bir şekilde adreslenebilir nesnelerin kendi aralarında oluşturduğu, dünya çapında yaygın bir ağ ve bu ağdaki nesnelerin belirli bir protokol ile birbirleriyle etkileşim içinde olmalarını sağlar (Lee ve ark., 2014). Nesnelerin interneti bilişim teknolojisi, iletişim teknolojisi ve internet teknolojisinin kesişim kümesi olarak da ifade edilebilir. Nesnelerin İnterneti ile her

zaman, her yerde, her şeyin interneti kavramı ortaya atılmış ve canlı cansız tüm nesnelerin internete bağlanabilmesi amaçlanmıştır.

Nesnelerin interneti de nesnelerin oluşturduğu bir ekosistemdir. Her türlü izleme cihazları, canlılar, robotlar, sensörler veya servisler nesnelerin internetinin ‘nesne’si olabilir (Rose ve ark., 2015). Nesnelerin interneti fiziksel ortamlardan algılanan sensör verilerini, filtreleyerek yapılan değerlendirmelerin ardından anlamlandırır. Bu anlamlandırmadan sonra ise ilgili kişilere iletilmesini sağlar (Gubbi ve ark., 2013). Bulut, Sis ve Kenar bilişim mimarileri Şekil 4.4’te görsel olarak ifade edilmiştir.



Şekil 4.4. Nesnelerin interneti mimarileri

4.4.1. Bulut Bilişim

Bulut bilişim, uygulamaların internet üzerinden uzak bir sunucuda çalıştırılmasını sağlayan bilgi işlem paradigması (Al-Qamash ve ark., 2018) ve internet tabanlı bilişim hizmetlerinin genel adıdır (Hurwitz ve Kirsch, 2020). Bulut bilişim ile hesaplama ve veri işleme uzak bir sunucuda işlenebilir. Bu sayede bilgi işleme ve veri yönetim maliyetleri düşürülmekte ve bakım maliyeti de minimuma indirilmektedir.

Bulut bilişim ile nesnelerin interneti uygulamaları birçok yönden desteklenmektedir. Düşük bellek kapasitesine sahip olan cihazlarla yüksek hesaplama ve bellek gerektiren uygulamalar yapmak mümkündür. Bulut bilişimin birçok avantajı ve verimli kullanımları olmasına rağmen, merkezi yapısının, verilerin aktarılması ve işlenmesi açısından gecikmeye duyarlı uygulamalar için verimsiz olduğu kanıtlanmıştır. Her saniye üretilen veri miktarının artması nedeniyle, bulutta gerçekleşen hesaplama, yoğun veri trafiğine ve yavaş aktarım hızlarına neden olmaktadır. Hesaplama gücü ağ bant genişliği sınırlı olduğu için, bulut bilişim büyük miktarda verinin iletimi ve gerçek zamanlı veri işleme uygulamalarında başarısı oldukça sınırlıdır (Hu ve ark., 2017).

4.4.2. Sis Bilişim

Bulut bilişimin dezavantajlarını gidermek için 2012’de sis bilişim paradigması tanıtılmıştır. Sis bilişim geleneksel bulut bilişim mimarisini ağıncı ucuna kadar genişleterek verilerin yerel olarak işlenmesine olanak tanımaktadır (Mouradian ve ark., 2017; Tang ve ark., 2019). İşlem birimlerini uç cihazlara yaklaştırmak, hesaplama gücünün kullanımını, görev yürütme süresini ve işlem süresini iyileştirmesine olanak tanımaktadır. Sis bilişim hızlı yanıt süresi ve esnek mobilite gerektiren uygulamaların çoğunda gerekli hizmetleri kolaylaştırmak için gereklidir (Dastjerdi ve Buyya, 2016).

Sis bilişim paradigması, dağıtılmış ve merkezi olmayan bir platform sunarak geleneksel bulut bilişim modelini genişleten katmanlı bir modele benzer. Bulut, sis ve terminal olmak üzere göre model üç ana katmana ayrılabilir. Bulut katmanı, sis katmanından alınan verileri yöneten, yüksek depolama ve hesaplama gücüne sahip veri merkezlerini temsil eder. Sis katmanı, her biri yeterli işleme ve geçici depolama yeteneklerine sahip bir dizi ağ cihazı içeren sis hücreleri olarak da adlandırılan çoklu sis düğümlerinden oluşur (Balevi ve Gitlin, 2018). Sis düğümleri, IoT veya son kullanıcı cihazlarından gelen istekleri alır ve gerekli verileri buluta iletir (Dastjerdi ve Buyya, 2016). Bunu takiben, cihaz katmanı olarak da adlandırılan terminal katmanı fiziksel dünyadan verileri algılayıp toplamaktan ve verileri sis katmanına göndermekten sorumludur. Böylece, bu paradigmanın katmanlı doğası, verileri zamanında işlemek için her katmanın komşu katmanlarla verimli bir şekilde iletişim kurmasını sağlar.

Sis bilişimi, gerçek zamanlı uygulamalar için bulut bilişimden sonra büyük bir gelişme olarak görülse de performansı hala gecikme ve bant genişliği açısından sınırlıdır.

Ayrıca buluta bağımlılığı hala bir dezavantaj olarak görülmektedir (Al-Qamash ve ark., 2018).

4.4.3. Kenar Bilişim

Kenar bilişim, bilgi işlem kaynaklarını ve depolamayı son kullanıcıya daha yakın olan ağın kenarında gerçekleştiren bir bilgi işlem modelidir (Abbas ve ark., 2017; Zhang ve ark., 2018). Bu nedenle, kenar bilişim, sis ve bulut bilişimin daha yerleştirilmiş bir versiyonu olarak tanımlanabilir.

Kenar bilişim, uç cihazlar ve bulut arasındaki iletişimin gecikmesini azaltır ve bulutun kaynaklarını ve yerel ağ cihazlarını verimli bir şekilde kullanır (Mach ve Becvar, 2017). Verilerin merkezde işlenmesi yerine ağın en uç kısmında işlenmesi ile merkezdeki işlem yükü azaltılmaktadır. Ayrıca kenar bilişim düğümleri sahadan toplanan fiziksel parametreleri merkeze göndermeden işlemekte ve gereksiz verileri ağa iletmeyecektir. Bu işlem ağ trafiğinin gereksiz yere meşgul edilmesini önlemektedir (Bonomi ve ark., 2012).

Bölüm 4'te, ilk olarak bu çalışmada kullanılan simülasyon programı olarak kullanılan SUMO programı tanıtılmıştır. Ardından tez kapsamında trafik sinyal optimizasyonu için kullanılan yapay optimizasyon teknikleri tanıtılmıştır. Bu tez çalışmasında optimizasyon yöntemi olarak sürü tabanlı yöntemler ve takviyeli öğrenme yöntemleri kullanılmıştır. Tez çalışmasında sezgisel optimizasyon yöntemi olarak kullanılan üç algoritma vardır; parçacık sürü optimizasyon algoritması, sosyal öğrenme parçacık sürü optimizasyon algoritması ve edimsel koşullanma ile sosyal öğrenme parçacık sürü optimizasyon algoritmasıdır. Bu bölümde kullanılan sezgisel algoritmalar açıklandıktan sonra takviyeli öğrenme yöntemlerine de yer verilmiştir. Tez kapsamında yapılan çalışmalarda kullanılan takviyeli öğrenme, Q-öğrenme, Derin Q-öğrenme yöntemlerinin açıklamasına da bu bölümde yer verilmiştir.

5. SİMULASYON ORTAMI ve TRAFİK SİNYAL OPTİMİZASYONU

Bu bölümde, önerilen optimizasyon tekniğinin performansını değerlendirmek için oluşturulan simülasyon ortamları ve izlenen deneysel çerçeve sunulmaktadır. Öncelikle bu çalışmada kullanılan trafik ışığı senaryoları açıklanmış, ardından kullanılan algoritmaların nasıl uygulandığı hakkında detaylı bilgiye yer verilmiştir.

5.1. Simülasyon Ortamının Oluşturulması ve Deneysel Parametreler

Bu tez çalışmasında üç farklı trafik senaryosu için trafik sinyal kontrol sistemi tasarlanmıştır. İlk senaryo gerçek saha verilerinden elde edilmiş olup, MOSAŞ Akıllı Ulaşım ve Tartım Teknolojileri şirketinden temin edilmiştir. İkinci senaryo ise koordineli trafik uygulamalarını temsil etmek için oluşturulmuştur ve ardışık iki kavşaktan meydana gelmektedir. Sonuncu ise şehir içi trafik ağını temsil eden 5×5 Manhattan ızgara tipi bir senaryodur.

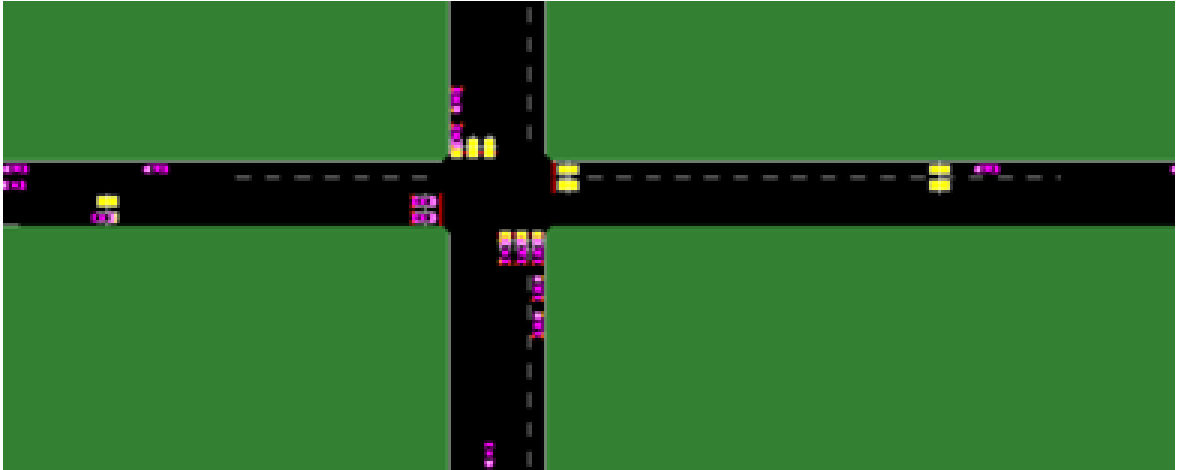
5.1.1. İzole Kavşak Simülasyon Ortamının Oluşturulması ve Parametreler

Bu tez çalışmasında oluşturulan ilk trafik senaryosu (Senaryo 1) Kilis (Türkiye) ilinde bir izole kavşaktan elde edilen gerçek trafik verileri ile oluşturulmuştur. Sahadan alınan yol ve araç güzergahı bilgileri simülasyon ortamına aktarılmıştır. Şekil 5.1'de uygulama yapılan kavşak yapısına yer verilmiştir.



Şekil 5.1. Uygulama yapılan izole kavşak, Kilis.

Şekil 5.2, incelenmekte olan kavşağın inşa edilmiş simülasyon modelini göstermektedir. İzole kavşak simülasyon modelinin oluşturulması için gereken dosyalar ve kodlar açık kaynak olarak verilmiştir (KilisData, 2021).

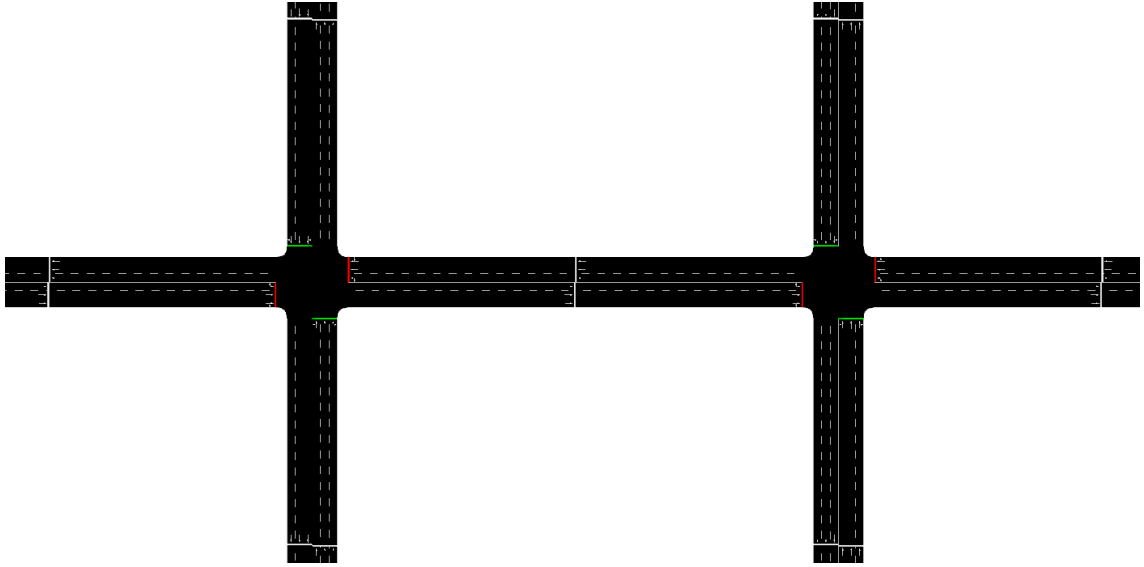


Şekil 5.2. Uygulama yapılan kavşağın simülasyon model görseli.

Senaryo 1 için önerilen yöntemin performans ölçütü olarak araç başına ortalama seyahat süresi kullanılmıştır. Bu parametrenin kullanılmasının nedeni, gerçek trafik verilerinde ortalama seyahat süresinin de ölçülmesidir. Saha verilerine göre araç başına ortalama seyahat süresi 59 saniyedir.

5.1.2. Koordineli Kavşak Simülasyon Ortamının Oluşturulması ve Parametreler

Bu tez çalışmasında koordineli kavşakların optimizasyonunu sağlayan sistemin değerlendirilmesi için Şekil 5.3'te görseli verilen trafik kavşaklar kullanılmıştır (Senaryo 2). Her iki kavşağa gelen her bir kol 3 şeritlidir. Şekil 5.3'te gösterildiği gibi her bir şerit bir aracın takip edebileceği olası yönleri tanımlamaktadır; en sağdaki şerit ve en soldaki şerit araçların sırasıyla sağa ve sola dönmesini veya düz gitmesini sağlar, ortadaki şerit sadece düz giden araçları temsil eder.

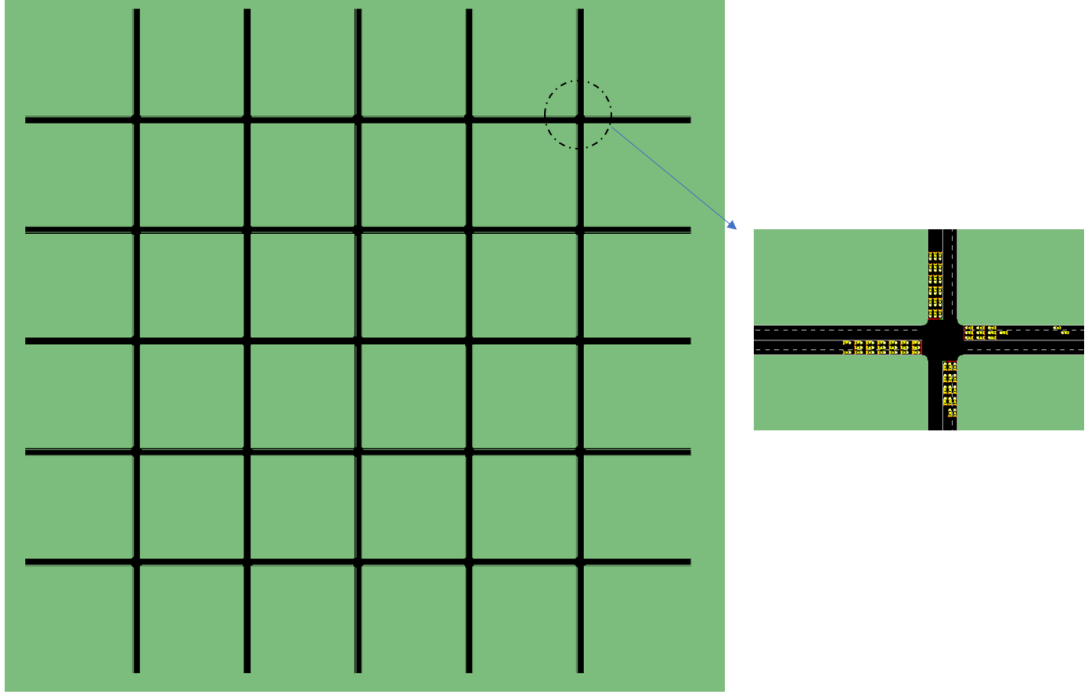


Şekil 5.3. Uygulama yapılan koordineli kavşak

Senaryo 2’de yol ve rota planlaması tamamen yapay olarak üretilmiştir. Trafik rejimi ve araç yoğunluğu gibi trafik parametreleri, trafik sinyal kontrol sistemlerinin performansını doğrudan etkilemektedir. Bu yüzden, Senaryo 2’ye, tasarlanan sistem performansını adil bir şekilde ölçmek için, literatürde sık kullanılan bir yöntem olan Weibull dağılımı ($\lambda=1$ ve $k=2$) kullanılmıştır (Vidali ve ark., 2019). Weibull dağılımı ile oluşturulan senaryoda trafiğin üç farklı rejimini de test edilebilmektedir. Ayrıca Weibull dağılımı ile trafik yoğunluğunu kademeli olarak azaltmak mümkündür. Weibull Dağılımı hakkında ayrıntılı bilgi için lütfen bakınız (Rinne, 2008). Koordineli kavşak simülasyon modelinin oluşturulması için gereken SUMO kodları ve dosyalar açık kaynak olarak verilmiştir (CoordinatedData, 2021).

5.1.3. Trafik Sinyal Ağı Simülasyon Ortamının Oluşturulması ve Parametreler

Bu çalışmada trafik sinyal ağı için oluşturulan optimum trafik sinyal yönetim planını değerlendirmek için Manhattan tipi 5×5 trafik ağı (Senaryo 3) oluşturulmuştur. Şekil 5.4'te gösterildiği gibi her kavşağa gelen her bir kol üç şeritlidir. En sağdaki şerit ve en soldaki şerit araçların sırasıyla sağa ve sola dönmesini veya düz gitmesini sağlar, ortadaki şerit düz giden araçları temsil eder. Senaryo 3 oluşturulurken Senaryo 2'deki gibi Weibull dağılımından ($\lambda=1$ ve $k=2$) yararlanılmıştır.



Şekil 5.4. Manhattan tipi 5x5 trafik ağı

5.2. Adaptif Trafik Sinyal Kontrolü

Adaptif trafik sinyal kontrolü uygulamalarında iki farklı aksiyon tipi mevcuttur. Birinci aksiyon sinyal programındaki faz sıralamasını değiştirmeden faz sürelerini optimize etmeyi amaçlar. İkinci aksiyon ise sabit faz süreleri ile faz dizilimini değiştirmektedir. Bu tez çalışmasında adaptif trafik sinyal kontrolüne hibrit bir yaklaşım tasarlanmıştır. Tasarlanan yaklaşım ile hem faz süresi hem de faz sıralaması optimize edilerek trafiğin etkin kontrolü sağlanmaya çalışılmıştır. Faz süresi optimizasyonu için sürü tabanlı algoritmalar, faz sıralaması için DQÖ algoritması kullanılmıştır.

5.2.1. Sürü Tabanlı Algoritmalar ile Faz Süresi Optimizasyonu

Bu bölüm, trafik sinyallerinin optimal çevrim programları için önerilen optimizasyon yöntemini açıklamaktadır. Ayrıntılı olarak çözüm vektörü, amaç fonksiyonu ve son olarak optimizasyon prosedürü hakkında bilgilere yer verilmiştir.

Bu çalışmada sinyal faz süresi optimizasyonu sezgisel sürü tabanlı algoritmalar kullanılarak gerçekleştirilmiştir. Tasarlanan model, kavşaktaki araç sayısı ve gecikme süresine göre çevrim süresini optimize eder. Bu çalışma da her bir kol için yeşil faz oranı -aktif yeşil süresinin kavşağın çevrim süresine oranı- eşit alınmıştır. Böylece çevrim süresi kol sayısına bölünerek fazın aktif yeşil süresi kolayca hesaplanmaktadır.

Çözüm vektörü: Bu çalışmada trafik sinyal ışıklarının faz süresi, SUMO tarafından desteklenen bir çözüm vektörü ile temsil edilmektedir.

Amaç fonksiyonu: Trafik sinyal kontrol sistemlerinin amaçlarından biri de toplam seyahat süresinin (TTT) en aza indirilmesidir. Bu nedenle, trafik sinyal kontrol uygulamaları için popüler bir performans metriği olan bir amaç fonksiyonu olarak toplam seyahat süresi fonksiyonu seçilmiştir (Jovanović ve ark., 2017; Celtek ve ark., 2020). Toplam seyahat süresi (TTT) Denklem 5.1 kullanılarak hesaplanır;

$$TTT = \sum_{(i,j)} TTT_{ij} = \sum_{(i,j) \in A} q_{ij} \cdot d_{ij} \quad (5.1)$$

Burada q_{ij} , i kavşağının j bağlantısı üzerindeki araç sayısını temsil eder. d_{ij} toplam gecikmeyi temsil etmektedir. Toplam gecikme (d_{ij}), kontrol gecikmesi ($d(C)_{ij}$ ortalama gecikme ve seyahat süresinin (t_{ij}) toplamıdır (Denklem 5.2);

$$d_{ij} = (d(C)_{ij} + d(\theta)_{ij} + t_{ij}) \quad (5.2)$$

Kontrol gecikmesi $d(C)_{ij}$ is Denklem 5.3 ile şu şekilde hesaplanır;

$$d(C)_{ij} = \sum_{k \in K} \alpha_{kij} \cdot (d_{1k} \cdot (PF_k) + d_{2k}) \quad (5.3)$$

burada;

$$d_{1k} = \frac{0,5 \cdot C \cdot (1 - \frac{g_k}{C})^2}{1 - [\min(1, X_k) \cdot \frac{g_k}{C}]} \quad (5.4)$$

$$d_{2k} = 900 \cdot T \cdot [(X_k - 1) + \sqrt{1 + \frac{4 \cdot X_k}{C_k \cdot T}}] \quad (5.5)$$

Ortalama gecikme süresi $d(\theta)_{ij}$ Denklem 5.6 ile hesaplanır;

$$d(\theta)_{ij} = \begin{cases} \sum_{k \in K} \frac{r_f}{g_f} \cdot (t_{ij} - \theta_{ij}) \cdot B_{fij} & \text{if } (t_{ij} - g_f) < \theta_{ij} < t_{ij} \\ (t_{ij} - \theta_{ij}) & \text{if } t_{ij} < \theta_{ij} < \sum_{k \in K} t_{ij} + (C - g_f) B_{fij} \end{cases} \quad (5.6)$$

$\alpha_{kij} = 1$ yada 0 (şerit α_{kij} , link(i,j)'nin elemanı ise 1 aksi durumda 0)

$B_{fij} = 1$ yada 0 (link(i,j) üzerindeki f kavşağı ise 1 aksi durumda 0)

θ_{ij} = sinyal offset süresi

T = periyod (saat)

PF_k = k yolunun düzeltme faktörü (saat)

C_k = k yolunun kapasitesi (araç/saat)

C = döngü süresi (saniye)

g_k = etkin yeşil süresi (saniye)

x_k = k yolunun doygunluk miktarı (araç)

d_{1k} = k yolundaki birim gecikme (saniye/araç)

d_{2k} = k yolundaki gecikme artış miktarı (saniye/araç)

Link (i,j) üzerindeki bir aracın seyahat süresinin (t_{ij}) hesaplanması Denklem 5.8'deki gibidir;

$$t_{ij} = t_{ij}(0) \cdot \left[1 + 0,15 \cdot \left(\frac{q_{ij}}{C_{ij}} \right)^4 \right] \quad (5.8)$$

Link (i,j) üzerindeki bir aracın serbest seyahat süresinin $t_{ij}(0)$ hesaplanması şu şekildedir (Denklem 5.9);

$$t_{ij}(0) = \frac{L_{ij}}{v_{ij}} \quad (5.9)$$

L_{ij} = link (i,j)'nin uzunluğu [metre]

V_{ij} = link (i,j) üzerindeki serbest akış hızı [metre/saniye]

q_{ij} = link (i,j) 'in üzerindeki anlık araç sayısı [araç/saat]

c_{ij} = link (i,j) 'in kapasitesi [araç/saat]

Kısıtlamalar:

Döngü uzunluğu (C), yeşil faz süresi (g_f) ve kayıp zaman (L_i) arasındaki bağlantı Denklem 5.10'daki gibidir.

$$\sum (g_f) = C - L_i \quad (5.10)$$

Ağıdaki sinyal ofsetini $(\theta)_{ij}$ hesaplamak için gereken koşul, aşağıdaki kısıtlama ile açıklanmıştır (Denklemler 5.11). (Burada n_p faz sayısıdır, kavşaktaki link sayısı ile doğru orantılıdır);

$$\sum (\theta)_{ij} + \sum (C - (\theta)_{ij}) = n_p \cdot C \quad (5.11)$$

Çevrim uzunluğu, yeşil faz süresi ve sinyal offset değerlerinin tanım aralığı aşağıda verilmiştir;

$$C_{min} \leq C \leq C_{max} \quad (5.12)$$

$$g_{fmin} \leq g_f \leq g_{fmax} \quad (5.13)$$

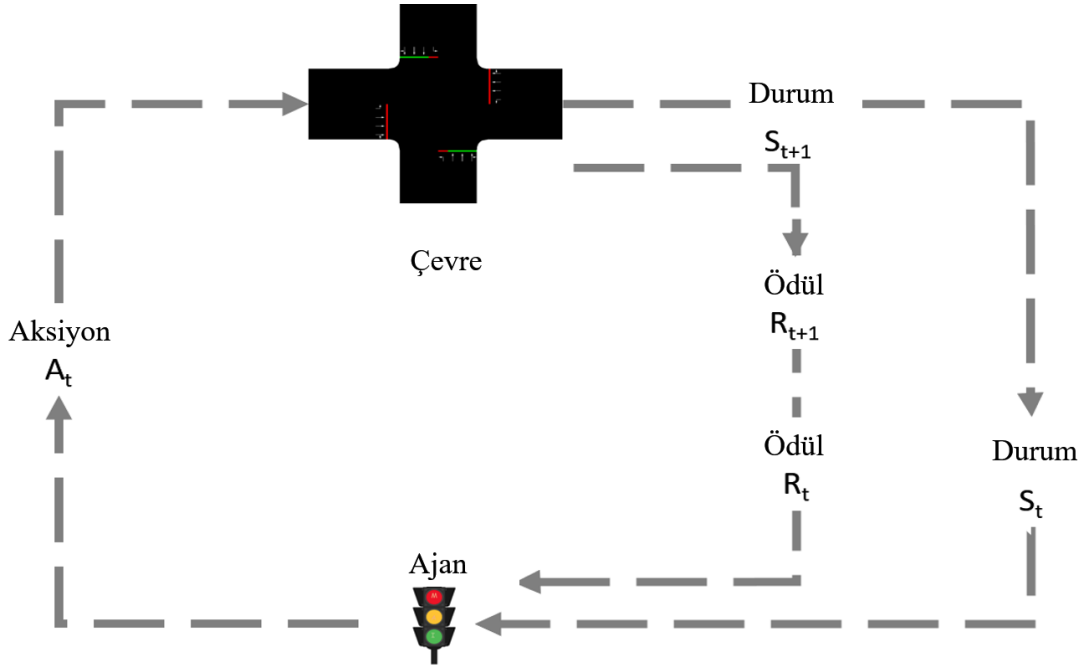
$$0 \leq (\theta)_{ij} \leq C \quad (5.14)$$

Bu çalışma da $C_{min} = 12 \text{ s}$, $C_{max} = 120 \text{ s}$, $g_{fmin} = 0 \text{ s}$ ve $g_{fmax} = 30 \text{ s}$ olarak alınmıştır.

5.2.2. Takviyeli Öğrenme ile Optimum Faz Sıralamasının Ayarlanması

5.2.2.1. Takviyeli Öğrenme Kavramlarının Tanımlanması

Tasarlanan kontrol modelinde faz sıralaması optimizasyonu Şekil 5.5'te gösterildiği gibi takviyeli öğrenme tekniği ile gerçekleştirilir. Trafik sinyal kontrol problemine takviyeli öğrenme tabanlı bir çözüm bulmak için yapılması gereken ilk adım takviyeli öğrenme kavramlarını (çevre, ajan, eylem, ödül) tanımlamak olmalıdır;



Şekil 5.5. Takviyeli öğrenme ile adaptif trafik sinyal kontrolü

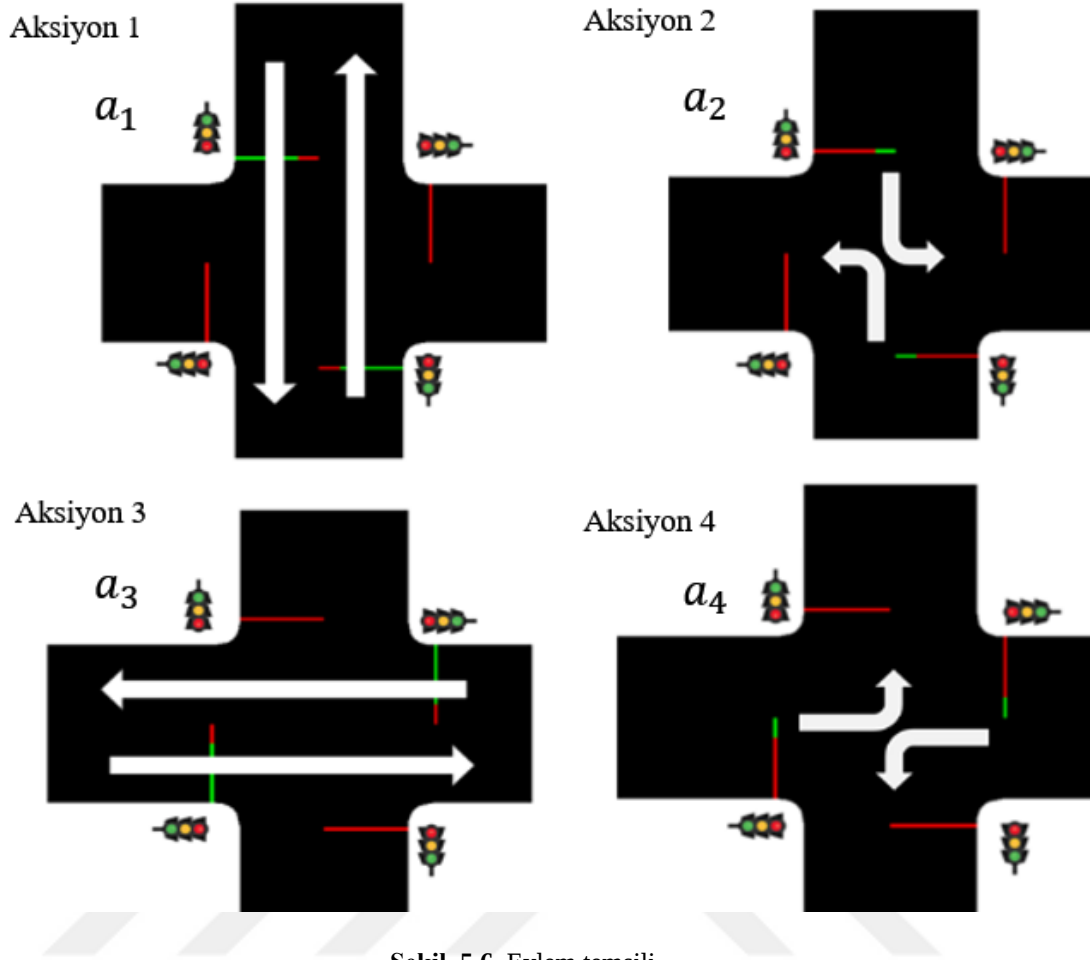
Çevre: Bu çalışmada çevre uygulama yapılan kavşağı temsil etmektedir.

Durum: Bu çalışmada durum olarak kavşaktaki kuyruk uzunlukları dikkate alınmıştır. Kavşağa gelen her bir kol iki farklı kuyruk uzunluğu sayısı ile temsil edilmiştir. Sola dönmek veya düz devam etmek isteyen araçlar (sağa dönüş de dahil) farklı kuyruk uzunluğu sayısı ile temsil edilmiştir.

Ajan: Bu çalışmada kavşakta bulunan trafik ışıkları ajan olarak temsil edilmiştir.

Eylem: Bu çalışmada kullanılan eylem kümesi (A) Şekil 5.6’da görsel olarak verilmiştir. Eylem kümesi ajanın gerçekleştirebileceği olası eylemleri tanımlar. Bu çalışma da kullanılan eylem kümesi (A) Denklem 5.15’te verilmiştir.

$$A = \{a_1, a_2, a_3, a_4\} \quad (5.15)$$



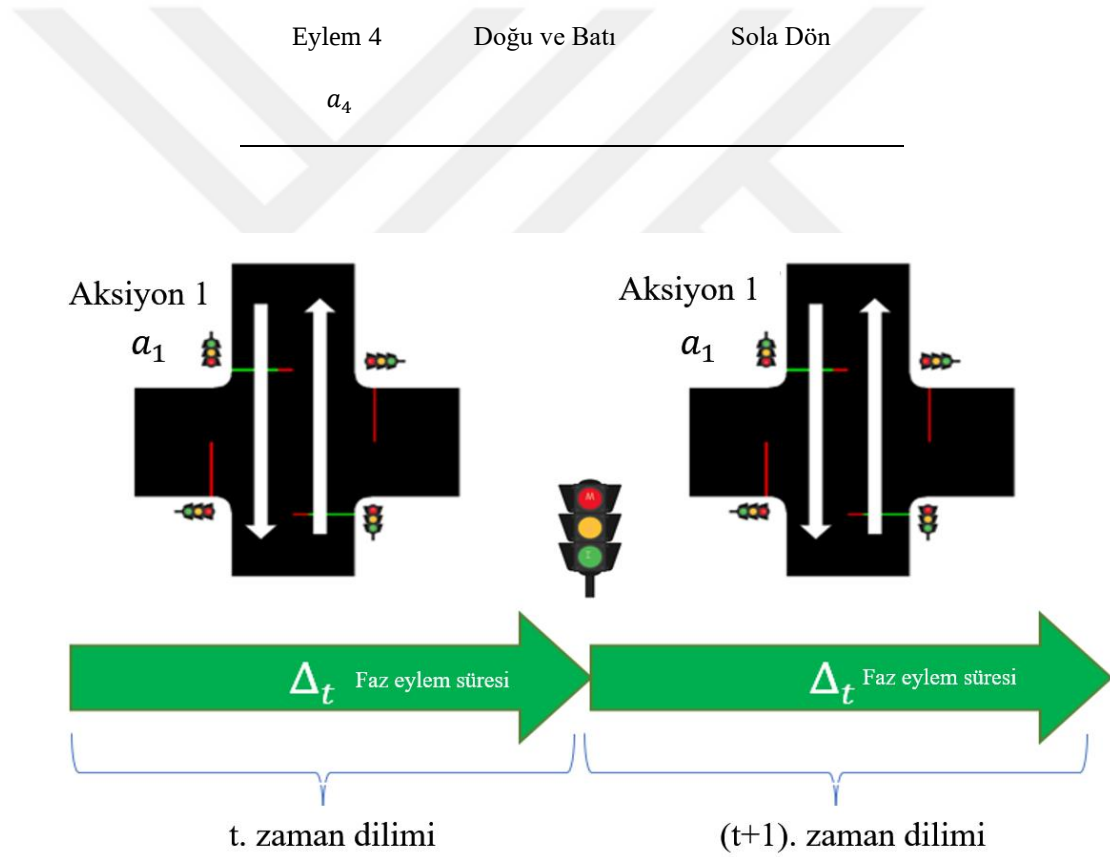
Şekil 5.6. Eylem temsili

Eylem kümesi dört farklı eylemden meydana gelmektedir. Bu eylemlerin açıklamaları Çizelge 5.1’de verilmiştir.

Ajan A eylem kümesinden eylemi seçerek, bazı yollar için faz eylem süresi (Δ_t) kadar yeşil fazı etkinleştirir. Aktif olan faz sürecince (t zaman aralığı) kavşağa yaklaşan araç sayıları ölçülür (Durum- s_t) ve Ödül (r_t) değerleri hesaplanır. Ajan hesaplanan Ödül (r_t) değerine göre yeni bir aksiyona (a_{t+1}) karar verir. Eğer $t+1$. zaman aralığında seçilen aksiyon bir önceki zaman dilimi olan t . Zaman aralığındaki aksiyonla aynıysa ($a_t = a_{t+1}$), mevcut aktif fazın süresi faz eylem süresi kadar uzatılır. Şekil 5.7’de bu durum görsel bir şekilde açıklanmıştır.

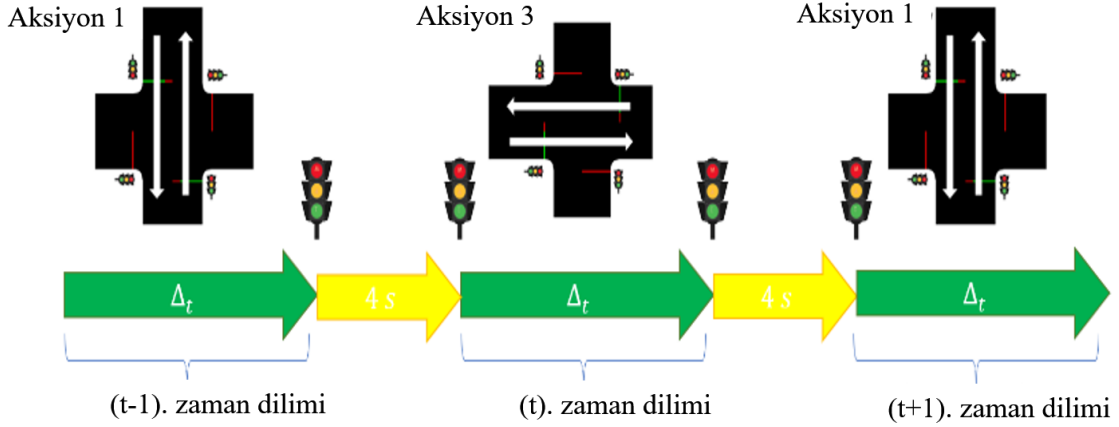
Çizelge 5.1. Eylem kümesi ve açıklamaları

Eylem	Etkin Yeşil Faz	İzin verilen Hareketler
Eylem 1	Güney ve Kuzey	Düz git
a_1		Sağa dön
Eylem 2	Güney ve Kuzey	Sola Dön
a_2		
Eylem 3	Doğu ve Batı	Düz git
a_3		Sağa dön
Eylem 4	Doğu ve Batı	Sola Dön
a_4		



Şekil 5.7. Aynı aksiyonların ardı ardına uygulanması

Eğer $t+1$. zaman aralığında seçilen aksiyon bir önceki t . zaman dilimindeki aksiyondan farklı ise ($a_t \neq a_{t+1}$), mevcut aktif fazın süresi bittikten sonra aktif faz sarı sinyal ile sonlandırılır. En uygun sarı sinyal süresi için literatürde sıkça alınan 4 sn değeri alınmıştır (Çakıcı ve Murat, 2016). Sarı sinyal süresi bittikten sonra yeni bir sinyale geçilir. Bu durum Şekil 5.8’de gösterilmiştir.



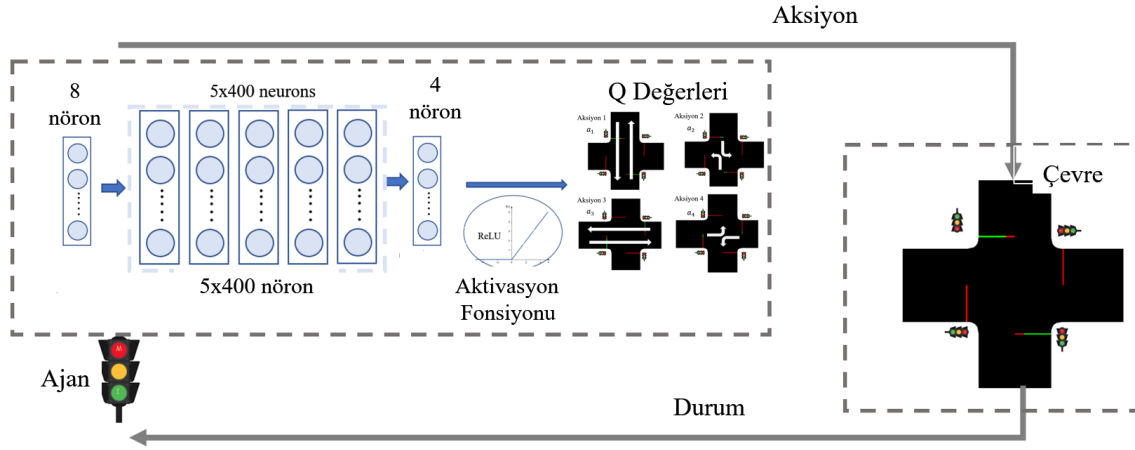
Şekil 5.8. Farklı aksiyona geçiş.

Ödül: Takviye öğrenmede ödül, ortamın eyleme tepkisi olarak değerlendirilir. Bu uygulamada amaç, kavşaktan geçen araçların gecikme zamanlarını minimuma indirmektir. Bu hedefe ulaşmak için ödül, trafik verimliliğinin temel performans ölçütlerinden biri olan kümülatif gecikme zamanı (Denklem 5.2) olarak belirlenmiştir.

5.2.3 Öğrenme Mekanizması

Bu çalışmada ajanın öğrenme mekanizması için Derin Q-öğrenme (DQÖ) kullanılmıştır. Derin sinir ağları ve Q-Öğrenme birleşiminden oluşan DQÖ modellemeye ihtiyaç duymayan bir öğrenme yöntemi olduğu için kullanılmıştır. Yapılan bir eylem ortamdan alınan geri bildirim göre Q değeri atanır.

Trafik sinyal kontrol problemi birçok olasılığı içeren zor ve karmaşık bir problemdir. Çok sayıda olasılık içermesi durum-eylem çiftini keşfetmeyi ve kaydetmeyi zorlaştırır. Bu nedenle, Q-öğrenme fonksiyonu bir sinir ağı kullanılarak yakınlştırılır.



Şekil 5.9. Öğrenme mekanizması

Bu çalışmada, 8 nöronlu giriş katmanı, 400 nöronlu 5 gizli katman ve 4 nöronlu bir çıkış katmanı kullanılmıştır. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır. ReLU, hesaplama yükünün sigmoid ve hiperbolik tanjant fonksiyonlarına göre az olmasından dolayı tercih edilmiştir. Bu çalışmada kullanılan derin sinir ağının grafiksel temsili Şekil 5.9’da gösterilmiştir.

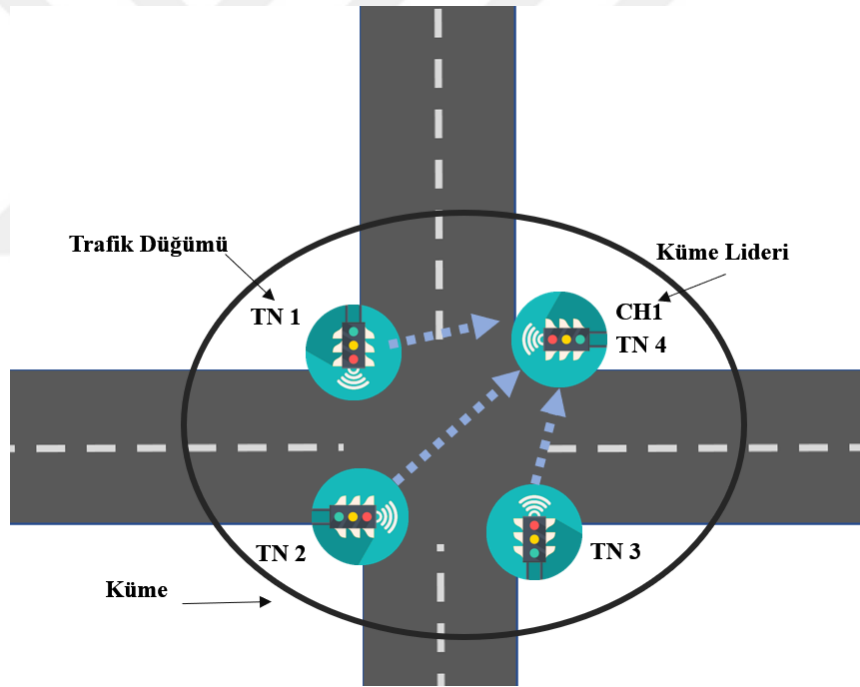
Bu çalışmada ajanın performansını ve öğrenme verimliliğini artırmak için eğitim aşamasında Deneyim Tekrarı (Experience Replay) tekniği uygulanmıştır. Deneyim tekrarı ile bir ajanın deneyimlerini saklamak ve daha sonra ağı bu deneyimlerle eğitmek amaçlanmıştır. Deneyim tekrarı tekniğinde, ajan simülasyon sırasında topladığı bilgileri hemen göndermek yerine bir bilgi seti oluşturur ve bu bilgi setini ‘deneyim’ olarak belleğe (memory) gönderir. Deneyim, bellekte <durum, aksiyon, ödül, bir sonraki durum> şeklinde saklanır.

Burada bahsi geçen bellek donanım olmayıp, kapasite ve küme boyutu (batch size) ile karakterize edilen sezgisel bir bellektir. Bellek kapasitesi, belleğin kaç örneği saklayabildiğini gösterir. Küme boyutu ise bir eğitim örneğinde bellekten alınan örnek sayısı olarak tanımlanır. Bu çalışmada bellek kapasitesi ve küme boyutu 40000 ve 80 olarak ayarlanmıştır.

5.3. Nesnelerin İnterneti Platformunda Adaptif Trafik Sinyal Kontrolü

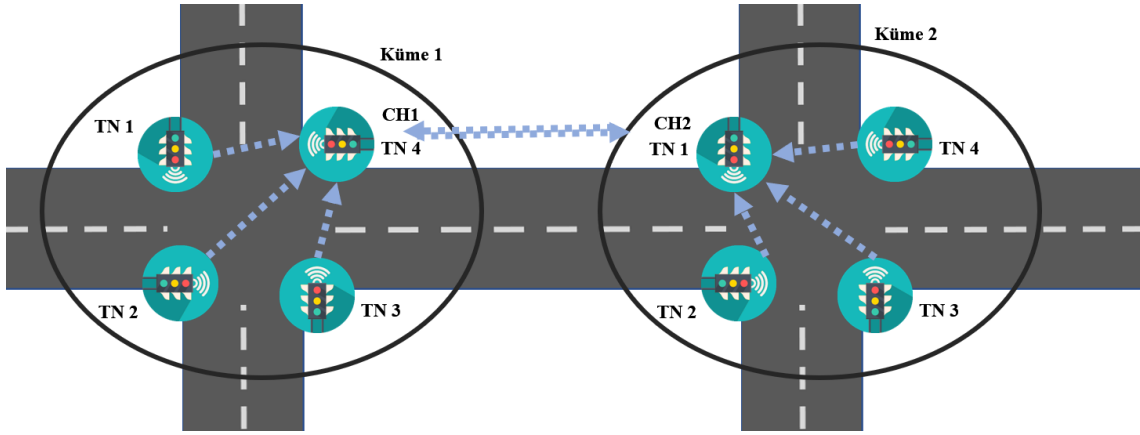
Bu çalışmada her bir kavşak birer kavşak kümesi olarak nitelendirilmiştir. Şekil 5.10’da gösterildiği gibi, izole bir kavşak kümesi trafik düğümlerinden (traffic node-TN) oluşmaktadır. Trafik düğümleri görevlerine göre trafikten anlık bilgi almayı sağlayan düğümler (trafik kameraları, endüktif sensörler v.b) ve trafiğin kontrolünü sağlayan düğümler (trafik sinyal lambaları) olmak üzere ikiye ayrılır. Bu projede trafik simülatörü kullanıldığı için trafik düğümleri trafik sinyal lambaları olarak gösterilmiştir.

Her bir kavşak kümesinde bir adet küme lideri (cluster head-CH) vardır. Trafik düğümleri trafik ortamından alınan anlık verileri (kuyruk uzunluğu, bekleyen araç sayısı, yaklaşan araç sayısı, hız v.b.) küme liderine gönderir. Küme lideri elde edilen trafik parametrelerine göre optimizasyon işlemini gerçekleştirir. Optimizasyon sonucu hesaplanan değerler kümedeki düğümlere iletilir.



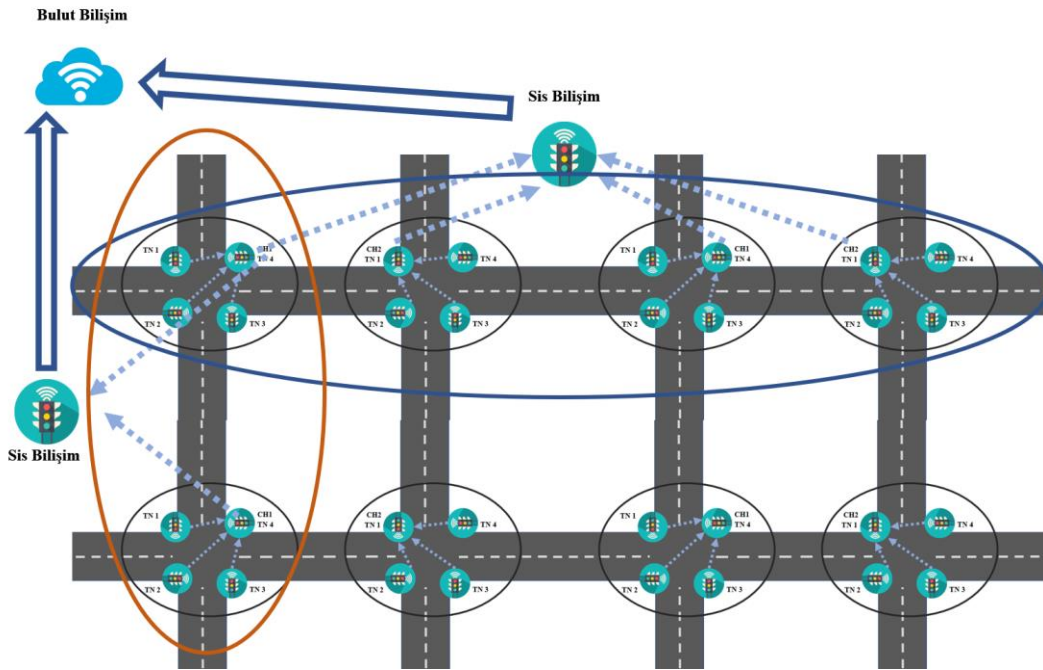
Şekil 5.10. Trafik düğümleri ve düğümlerin oluşturduğu kavşak kümesi.

Eğer kavşak izole bir kavşak değil ve Şekil 5.11’teki gibi koordineli bir kavşak ise, küme lideri sahadan alınan verileri (yaklaşan araç sayısı, bekleme süresi, faz bilgileri) filtre ederek bağlı olduğu diğer kavşak kümesinin liderine gönderir. Bağımlı kavşaklarda optimizasyon işlemi yapılırken diğer kavşaklardan gelen bilgiler de dikkate alınır.



Şekil 5.11. İki kavşak kümesinin birbirleri ile haberleşmesi.

Koordineli kavşak sayısının ikiden fazla olması durumunda, Şekil 5.12’teki gibi kavşakların oluşturduğu trafik ağı kümelerine ayrılarak optimizasyon işlemi gerçekleştirilir. Kümelere ayırma işlemi karayolundaki önceliklere göre yapılır. Şekil 5.12’te mavi renk ile gösterilen küme karayolunda birinci önceliğe sahip olan anayolu temsil etmektedir. Turuncu renk ise ana yola bağlı olan tali yolu temsil eder. Siyah renkteki kümeler ise en son önceliğe sahip bağımsız izole kavşaklardır.

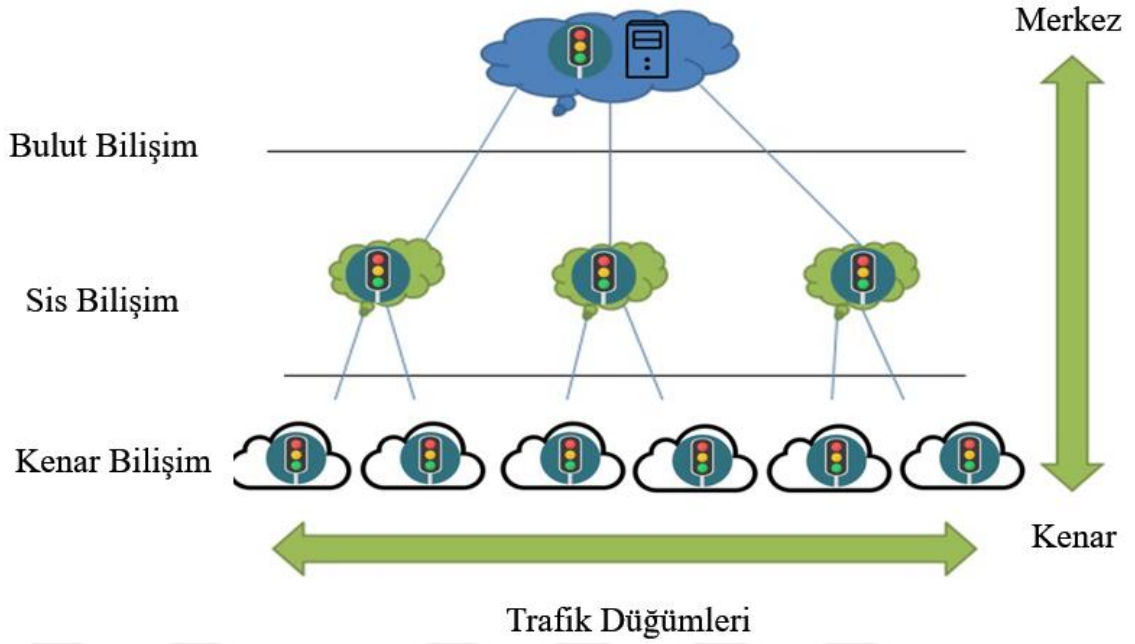


Şekil 5.12. Trafik sinyal ağı görseli

Trafik sinyal ağındaki en büyük problem koordinasyon ve ağı oluşturan her bir trafik sinyallerinin iletişimidir. Ağıdaki sinyal sayısı arttıkça bu problem daha da karmaşıklaşmaktadır. Bu sorunların üstesinden gelmek için nesnelerin interneti ve bilişim yöntemleri potansiyel bir çözümdür.

Bu tez çalışmasında üç farklı bilişim yönteminin de kullanıldığı trafik sinyal kontrol mimarisi önerilmektedir. Toplanan trafik verileri, gerçek zamanlı kararlar almak için verimli ve hızlı algoritmalar kullanılarak trafik verilerinin oluşturulduğu yerde işlenmektedir ve yapılan optimizasyon işlemlerin başarısını daha da arttırmak için bir üst katmanda alt katmandan bağımsız farklı optimizasyon işlemi gerçekleştirilmektedir.

Şekil 5.13'teki siyah kümedeki trafik düğümleri Şekil 5.14'te sınır cihazı olarak temsil edilmiş ve optimizasyon işlemi kenar bilişim ile çözümlenmeye çalışılmıştır. Sınır bilişim ile veriler mümkün olduğunca kaynağına yakın olarak işlenir. Bu sayede veri iletiminden kaynaklı gecikme minimuma indirilmektedir. Sınır bilişim katmanında yapılan optimizasyon işleminin giriş verileri kavşaklardan elde edilen yerel (tek bir kavşağa ait) verilerdir. Yerel veriler ile yapılan işlemlerde, optimizasyon işleminin başarısının bir limiti vardır. Bu başarıyı daha da arttırmak için yerel veriler bir üst katman olan sis bilişimde birleştirilerek yeniden optimizasyona tabii tutulur. Sınır bilişim ve sis bilişim katmanlarında yapılan optimizasyon işlemleri takviyeli öğrenme tabanlıdır ve “mevcut fazı koru veya değiştir” eylemi ile kontrol gerçekleştirilir. Bir üst katman olan bulut bilişim tarafından belirlenmiş bir faz eylem süresi ile fazlarda kullanılır. Bulut bilişim katmanı kullanıcının tüm ağın bilgilerine eriştiği katmandır. Bu katmanda yapılan optimizasyon işlemi diğer katmanlardaki gibi gerçek zamanlı değildir. Bu katmanda elde edilen veriler kaydedilip, raporlanır ve analiz edilir. Anlık veriler, geçmiş dönemde edilen verilerle harmanlanıp optimizasyon gerçekleştirilir. Bu katmandaki optimizasyonun temel amacı, alt katmanlarda (sis ve sınır) yapılan optimizasyon işlemi için gerekli olan faz eylem süresinin hesaplanmasıdır.



Şekil 5.13. Nesnelerin interneti mimarilerinin trafik ağına uygulanması.

Bölüm 5’te deneysel çerçeve ve kullanılan optimizasyon yöntemlerinin matematiksel altyapısı açıklanmıştır. İlk olarak simülasyonda uygulama yapılan trafik ortamları tanıtılmıştır. Bu çalışmada hem gerçek trafik verileri ve hem de yapay olarak üretilen trafik verileri kullanılarak trafik senaryoları elde edilmiştir. Ardından elde edilen senaryolar, optimizasyon yöntemlerinin ihtiyaç duyduğu parametreler için matematiksel olarak modellenmiştir. Bölüm 5, ayrıca takviyeli öğrenme tabanlı trafik kontrolörüne ait tasarım bilgileri de içermektedir. Takviyeli öğrenmenin “çevre”, “ajan”, “durum” ve “ödül” bileşenleri ayrıntılı olarak açıklanmıştır. Bu çalışma ile kontrol edilmesi gereken trafik sinyal sayısının artması ise ortaya çıkacak karmaşanın önüne geçmek için nesnelerin interneti tabanlı bir mimari önerilmiştir. Bölüm 5, bu mimarideki katmanları ve katmalarda yapılan işlemler açıklanmıştır.

6. ADAPTİF TRAFİK SİNYAL KONTROL UYGULAMALARI ve SONUÇLAR

Bu bölümde, bu tez çalışmasında yapılan çalışmalar kronolojik olarak açıklanmıştır. Bölüm 6'nın alt bölümlerinde yapılan çalışmalar farklı trafik problemleri ve senaryoları içermektedir. Önerilen kontrol yöntemlerini oluşturan algoritmalara ve çalışma prosedürlerine ilgili alt bölümlerde yer verilmiştir. Tez kapsamında yapılan çalışmalara ilham ve motivasyon kaynağı olan sonuçlar vurgulanarak detaylı bir değerlendirme yapılmıştır.

6.1. Sürü Tabanlı Trafik Sinyal Kontrol Uygulaması ve Sonuçları

Bu bölümde yapılan çalışma ile sürü algoritmalarının trafik sinyal kontrol yöntemlerindeki başarısı incelenmiştir. Bu kapsamda optimum faz süresini hesaplamak (faz sıralaması sabit) için ilk olarak klasik PSO algoritması uygulanmıştır. PSO'nun seçilme sebebi, hızlı yakınsama gerektiren problemlerde sağladığı başarılarıdır (Hansen ve ark., 2008). Bu çalışma, SL-PSO'nun trafik sinyal kontrol yöntemlerinde kullanıldığı ilk ve tek çalışma olarak literatüre geçmiştir (Celtek ve ark., 2020).

Bu çalışmanın amacı olan trafik ışıklarının hızlı optimize edilmesi işlemi, PSO'nun yakınsama hızı ile doğrudan ilgilidir. Bundan dolayı PSO'nun yakınsama hızını belirleyen parametreler diğer çalışmalarda olduğu gibi rastsal seçilmemiş, literatürde önerilen spesifik değerler alınmıştır. Buna göre ivme katsayıları ($c1$ ve $c2$), 2.05'e ayarlanmış ve atalet ağırlığı (ω), ise 0.5 - 0.1 aralığında olup doğrusal olarak azalan şekilde ayarlanmıştır (Eberhart ve Shi, 2000; Clerc ve Kennedy, 2002).

Bu bölümde yapılan çalışmada trafik ışıklarının kontrolü optimizasyon ve simülasyon süreci olmak üzere iki ana bölümden oluşmaktadır. Optimizasyon sürecinde, en uygun faz sürelerini bulmak için sürü tabanlı algoritma çalıştırılır. Tasarlanan PSO tabanlı kontrol prosedürü Algoritma 6.1'de adım adım anlatılmaktadır;

Algoritma 6.1. PSO tabanlı kontrol prosedürü

Adım 1- Başlangıç durumunda faz sürelerini temsil eden çözüm kümesi, faz sürelerini rastgele seçilmiş parçacıklardan oluşur. Bu parçacıklar $[\min_phase_value, \max_phase_value] \in \mathbb{Z} +$ zaman aralığındadır. $\min_phase_value$ ve $\max_phase_value$ parametreleri kullanıcının başlangıçta belirlemesi gereken sınır değerleridir.

Adım 2- Parçacıkların hız güncellemesi yapılır.

Adım 3- Konum güncellemesi yapılır.

Adım 4- Uygulama fonksiyonu çalıştırılır.

Adım 5- Elde edilen en iyi değerler simülasyon işlemine gönderilir.

Adım 6- Adım 2'ye geri döndürülür.

Bu uygulama ile sürü algoritmalarının parçacık sayısının trafik performansına etkisi de araştırılmıştır. Bunun için PSO beş farklı parçacık sayısı ile (5, 10, 25, 50 ve 125) ayrı ayrı uygulanmış ve sonuçları karşılaştırılmıştır. Karşılaştırılmanın adil olması için uygunluk değerlendirmelerinin sayısının (FEs, Fitness Evaluations) eşit olması kriteri dikkate alınmıştır.

Uygunluk değerlendirme sayısı, meta-sezgisel algoritmalarda bir çözümün kaç kez değerlendirildiğini ifade etmektedir. Uygunluk fonksiyonunu tek bir iterasyonda birden fazla değerlendirildiği için (parçacık sayısı ile doğru orantılı şekilde artış gösterir) iterasyon sayısı ile uygunluk değerlendirme sayısı farklı anlama gelmektedir. Literatürde sezgisel yöntemler değerlendirilirken maksimum uygunluk değerlendirmesi sayısı, maksimum iterasyon sayısına tercih edilmektedir (Engelbrecht, 2014). Bu durumun nedeni, her bir uygunluk değerlendirmesinin çözüm hakkında bazı bilgiler sağlamasıdır. Her bir iterasyonda çok parçacıklı bir sürü, az parçacıklı sürüden daha fazla bilgi toplamış olur. Uygunluk değerlendirmelerinin sayısını sınırlandırılarak, bir algoritmanın probleminden elde edeceği "bilgi miktarı" sınırlandırılır. Bu şekilde iki algoritmayı da karşılaştırmak adil olur. Bir karşılaştırma yaparken, tüm algoritmalar aynı sayıda maksimum uygunluk değerlendirmesi için çalıştırılmalıdır. Aksi takdirde, uygunluk fonksiyonunu daha fazla çalıştırılan yöntemin, problem hakkında daha fazla bilgi sahibi olacağından, daha iyi bir çözüm sunma şansı daha fazla olur.

Uygunluk deęerlendirmelerinin sayısı kriterine uyum saęlamak için 5, 10, 25, 50 ve 125 parçacıklı uygulamalarda iterasyon sayısı sırasıyla 50, 25, 10, 5 ve 2 olarak seçilmiştir. Bu çalışmada uygulanan uygunluk deęerlendirmelerinin maksimum sayısı 250 olarak belirlenmiştir.

Bu bölümde yapılan çalışma için trafik verisi olarak simülasyon programında Bölüm 5.1’de açıklanan Senaryo 1 kullanılmıştır. Algoritmaların performansını deęerlendirmek için araç başına ortalama seyahat süresi deęeri elde edilmiştir. Bu parametrenin kullanılmasının nedeni, gerçek trafik verilerinde ortalama seyahat süresinin ölçülmesidir. Saha verilerine göre araç başına ortalama seyahat süresi 59 saniyedir.

Çizelge 6.1 ile farklı parçacıklı PSO yaklaşımlarının sonuçları sunulmaktadır. Verilen çizelge de parçacık sayısı N ile sembolize edilmiştir. Simülasyondan elde edilen en iyi sonuçlara göre tüm yaklaşımlarda toplam seyahat süresi iyileştirilmiştir. En iyi sonuç, 10 parçacıklı ve 50 parçacıklı yaklaşım ile 46 saniye olarak hesaplanmıştır. Bu ortalama seyahat süresinde yaklaşık %28’lik bir iyileşme saęlandığı anlamına gelmektedir.

Çizelge 6.1. Parçacık sayısının sonuca etkisi

Sonuçlar	PSO N=5	PSO N=10	PSO N=25	PSO N=50	PSO N=125
En iyi [sn]	47	46	50,5	46	47
En kötü [sn]	75	113	126	75	124
Ortalama [sn]	54,95	50,4	59,6	67,3	68,5

Parçacıklardan elde edilen deęerlerinin ortalaması dikkate alındığında, 5 parçacıklı ve 10 parçacıklı PSO yaklaşımları sırasıyla %10 ve %18’lik bir iyileştirme gerçekleştirmiştir. Parçacıkların en kötü deęerleri dikkate alınırse herhangi bir iyileştirme elde edilememiştir. En iyi deęer ve ortalama deęerler dikkate alınırse, bu optimizasyon problemi için optimum parçacık sayısı N=10 şeklinde tespit edilmiştir.

Çizelge 6.1’deki ortalama sonuçlar, büyük boyutlu (N>25) PSO algoritmalarının sonuçlarının istenen iyileştirmeyi gerçekleştiremediğinin ispatı niteliğindedir. Parçacık sayısı arttıkça optimum çözüme yakınsama azalmıştır. PSO, bir sürü algoritmasıdır.

Dolayısıyla, bir parçacığın global bir optimumu varsa, tüm parçacıkların hareketi o yönde olacaktır. Bu çalışmada, yüksek boyutlu PSO uygulamalarının istenilen çözüme yakınsayamamasının nedeni gerekenden daha az iterasyon ile çalıştırılmasıdır. İterasyon sayısı ne kadar yüksek olursa, sonuçların optimum noktalara yakınsaması artacaktır. Ancak iterasyon sayısının artması hesaplama süresini de arttırmaktadır.

Önerilen kontrol modelleri bir günlük gerçek trafik verileri ile test edilmiştir. Çizelge 6.2’de gösterilen süre, simülasyon süresi ile hesaplama süresinin toplamıdır. Çizelge 6.2’de görüldüğü gibi parçacık sayısı arttıkça süre de artmaktadır. Bunun nedeni parçacık sayısının arttıkça gerekli hesaplama süresinin de artmasıdır. Gerçek zamanlı trafik sinyal kontrol uygulamalarında hesaplama süresinin arttırılması istenmeyen bir durumdur ve kontrolü daha da zorlaştırmaktadır.

Çizelge 6.2. Hesaplama ve simülasyon süre toplamları

	PSO	PSO	PSO	PSO	PSO
	N=5	N=10	N=25	N=50	N=125
Süre [dak]	50	239	326	377	382

Sezgisel algoritmaların kullanıldığı uygulamalarda çözümlerin rastgele olup olmadığı tartışılmalıdır. Çizelge 6.3, elde edilen sonuçların standart sapmasını göstermektedir. Hesaplanan standart sapma 0'a yakınsa, elde edilen çözümlerin rastgele olmadığı ve her zaman optimal değerlere yakınsayacağı anlamına gelmektedir. Kullanılan standart sapma denklemi şu şekildedir;

$$\sigma = \sqrt{\frac{1}{N-1} \cdot \sum (a_i - a_{mean})^2} \quad (7.1)$$

σ = standart sapma değeri

N = parçacık sayısı

a_i = i . parçacığın değeri

a_{mean} = sürünün ortalama değeri

Çizelge 6.3’e göre, tüm yaklaşımlarda iterasyon değeri arttıkça standart sapma değeri azalmıştır. Bu azalma değerleri, tüm parçacıkların optimum noktalara yakınsadığını göstermektedir. 50 ve 125 parçacıklı uygulamalarda başlangıç standart

sapma değerleri çok yüksek olmasına rağmen iterasyon sayısı arttıkça azalma olmaktadır. Çizelgeye göre, PSO'nun 5 ve 10 parçacık ile çalıştırıldığı uygulamalara ait standart sapma değerleri istenen seviyelere ulaşmıştır.

Çizelge 6.3. Standart sapma sonuçları.

İterasyon Sayısı	PSO N=5	PSO N=10	PSO N=25	PSO N=50	PSO N=125
1	7	1,023	23	30	31,16
2	9	1,023	25	22	23,55
3	8	1,023	23	19	
4	9	1,028	14,9	16	
5	9	1,028	14,2	8	
6	4	1,028	9,34		
7	4	1,090	9,28		
8	2	1,090	3,33		
9	2	1,043	3,39		
10	1,24	1,010	3,40		
25	1,15	1,010			
50	1,15				

Aynı senaryoya trafik sinyali optimizasyonu için SL-PSO algoritması uygulanmıştır. SL-PSO tabanlı yöntemin işleyişi, PSO'dan farklılık göstermektedir. SL-PSO tabanlı yöntemin prosedürü Algoritma 6.2'de verilmiştir;

Algoritma 6.2. SL-PSO tabanlı kontrol prosedürü

Adım 1- Başlangıç durumunda faz sürelerini temsil eden çözüm kümesi, faz sürelerini rastgele seçilmiş parçacıklardan oluşur. Bu parçacıklar $[\min_phase_value, \max_phase_value] \in \mathbb{Z} +$ zaman aralığındadır. $\min_phase_value$ ve $\max_phase_value$ parametreleri kullanıcının başlangıçta belirlemesi gereken sınır değerleridir.

Adım 2- Parçacıkların konumları en iyiden en kötüye doğru sıralanır.

Adım 3- Sosyal öğrenme gerçekleştirilir.

Adım 4- Uygulama fonksiyonu çalıştırılır.

Adım 5- Elde edilen en iyi değerler simülasyona gönderilir.

Adım 6- Adım 2'ye geri döndürülür.

Çizelge 6.4, SL-PSO'nun, PSO'dan daha hızlı sonuç verdiğini göstermektedir. Çizelgeye bakıldığında en iyi değerler arasında bir eşitlik söz konusudur. En kötü ve ortalama değerler SL-PSO ile iyileştirilmiştir.

Çizelge 6.4. PSO ve SL-PSO karşılaştırılması

	PSO	SL-PSO
	N=10	N=10
En iyi [sn]	46	46
En kötü [sn]	113	105
Ortalama [sn]	50,4	49,2
Süre [dk]	239	212

SL-PSO, PSO gibi optimizasyon probleminde bir çözüm bulmuş ve optimum nokta olan en iyi değere ulaşmıştır. SL-PSO bu sonucu ortalama 212 saniyede yakalarken, PSO yaklaşık %10 daha fazla süre harcamıştır. Simülasyon sürelerine göre, SL-PSO, PSO varyantlarına kıyasla daha yüksek bir hesaplama verimliliğine sahiptir. Literatürde SL-PSO'nun yakınsama hızının PSO'dan daha hızlı olduğu iddia edilmiştir (Cheng ve Jin,

2015). Bu çalışmanın sonuçları, gerçek zamanlı mühendislik problemleri için bu ifadeyi doğrulamaktadır.

Önerilen kontrol yaklaşımları, bir günlük gerçek trafik verileriyle test edilmiştir. Elde edilen sonuçlara göre, sürü tabanlı bir optimizasyon stratejisi, bir aracın ortalama seyahat süresini yaklaşık %28'lik bir iyileştirme ile 59 saniyeden 46 saniyeye düşürmektedir. Bu gelişme, bir günlük verilerin ortalama sonucudur.

Adaptif trafik sinyal kontrolü sistemlerinden trafiğin farklı rejimine (doymuş trafik durumu, aşırı doymuş trafik durumu ve doymamış trafik durumu) de uyum sağlaması beklenmektedir. Bu nedenle, farklı trafik rejimlerindeki performansı araştırmak için sürü tabanlı yaklaşımlarımızı iki farklı zaman dilimine uyguladık. Trafiğin yoğun olduğu saat dilimleri (08.00-10.00) aşırı doymuş trafiği, trafiğin az yoğun olduğu 13.00-15.00 saat dilimleri de doymamış trafiği temsil etmektedir.

Çizelge 6.5. Farklı trafik rejimlerinde performans değerlendirmesi

	Saha	PSO	SL-PSO
	Verisi	N=10	N=10
Tüm gün	59 s	46 s	46 s
08.00-10.00	79 s	62 s	62 s
13.00-15.00	49 s	42 s	42 s

Hali hazırda kullanılan trafik sinyal kontrol sistemleri doymamış rejimlerde başarılı sonuçlar verirken, aşırı doymuş rejimlerde istenen performansı vermemektedir (Gartner ve ark., 2002). Çizelge 6.5'e bakıldığında da kavşakta hali hazırda kullanılan sistem düşük yoğunluklu trafikte daha verimli sonuç vermektedir. Sürü tabanlı çözümler bir aracın yoldaki ortalama seyahat süresini 49 saniyeden 42 saniyeye düşürülmüştür. Yüksek yoğunluklu trafikte yapılan uygulamalarda yapılan iyileştirmeler daha fazladır. Araç başına ortalama seyahat süresi 79 saniyeden 62 saniyeye düşmüştür.

Bu bölümde yapılan çalışmaları özetlemek gerekirse, sürü tabanlı sezgisel yöntemler ile anlık trafik sinyal durumuna yeni bir çözüm vektörü üretilir. Bu çözüm vektörü ile döngü programı güncellenir. Döngü süresi güncellenen trafik ağı senaryosu SUMO ile simüle edilir. Simülasyon sonucunda elde edilen trafik parametrelerine göre

optimizasyon fonksiyonu tekrar çalıştırılır. Bu işlem, kullanıcı tarafından belirlenen durdurma kriterine ulaşılan kadar devam eder.

Bu çalışmada kullanılan optimizasyon algoritmaları Python platformu üzerinde yazılmış ve kodların doğruluğu CEC 07 fonksiyonları ile test edilmiştir. Simülasyon için Windows işletim sisteminde SUMO trafik simülatörü kullanılmaktadır. Simülasyon, I7 7500U Intel 2.7 GHz işlemci ve 16 GB RAM kullanılarak gerçekleştirilmiştir. PSO ve SL-PSO sezgisel tabanlı algoritmalar olduğu için elde edilen sonuçlar değişkenlik gösterebilmektedir. Bu değişkenliği azaltmak için, trafik senaryosu simülatörde simüle edilmeden algoritmalar 25 kez çalıştırılmıştır. Elde edilen değerlerin ortalaması alınmış ve bu ortalama değerlerin en iyi, en kötü ve ortalama değerleri hesaplanmıştır. Simülasyon programı en iyi, en kötü ve ortalama değerler için ayrı ayrı çalıştırılmış ve sonuçlar tablo şeklinde verilmiştir.

6.1.1. Sürü Tabanlı Trafik Sinyal Kontrol Uygulamasının Değerlendirilmesi

Bu bölümde yapılan çalışma ile gerçek zamanlı trafik kontrol sistemleri için sürü tabanlı optimizasyon modeli önerilmiştir. Önerilen modelin kapsamı izole kavşaktır. Önerilen model ile seyahat hızının azaltılması hedeflenmiştir. Bu bölümde yapılan çalışma ile sürü tabanlı yöntemlerin standart sapma ile gerçek zamanlı trafik sinyal kontrolünde başarılı olduğu istatistiksel olarak kanıtlanmıştır. Sürü tabanlı modellerin tüm trafik rejimleri için mevcut yöntemlerden daha başarılı olduğu ispat edilmiştir.

Sürü tabanlı stratejiler de parçacık sayısının sonuca etkisi ve sonuçların rastsallığı bu çalışmada araştırılmıştır. Sürü yöntemlerinin mevcut problemlerin çözümünde yakınsaması ve başarısı farklı parçacık uygulamalarında istatistiksel olarak kanıtlanmıştır.

Çalışmadan elde edilen çıkarımlar şu şekilde sıralanabilir;

1) PSO algoritması, gerçek trafik senaryolarında trafik sinyallerini optimize etmek için başarıyla kullanılabilir. Araç başına ortalama seyahat süresindeki %28'lik iyileşme bu çıkarımın kanıtı olarak gösterilebilir.

2) Önerilen sürü tabanlı modeller, hem aşırı doymuş hem de doymamış trafik rejimlerinde mevcut sistemden başarılıdır. Araç başına ortalama seyahat süresindeki iyileşmeler bu çıkarıma kanıt olarak verilebilir.

3) Gerçek zamanlı trafik sinyali kontrolü, hızlı bir şekilde çözülmesi gereken optimizasyon problemidir. PSO uygulamalarında gerekli olan hesaplama süresi parçacık

sayısı ile doğrudan ilişkilidir. Parçacık sayısının artması, algoritmanın matematiksel hesap yükünü ve hesaplama sürecini arttırmakta ve çözüm süresini uzatmaktadır. Bu çalışmada 5, 10, 25, 50 ve 125 parçacıklı PSO yaklaşımları ile ortalama seyahat süresi iyileştirilmeye çalışılmıştır. Mevcut şartlarda elde edilen en hızlı çözüm, 10 parçacıklı PSO çözümüdür. Elde edilen çözümlerin ortalama, en iyi, en kötü ve standart sapma değerleri arasındaki fark bu çıkarımın kanıtıdır.

4) Elde edilen sonuçlara göre, araç başına ortalama seyahat süresinde önemli bir iyileşme söz konusudur. Uygulama yapılan kavşak Kilis şehir merkezindedir. Bu iyileştirme ile mevcut yolun trafik potansiyeli de arttırılmıştır.

5) Çalışma sonunda önerilen yöntemin (10 parçacıklı sürü algoritması) hesaplama süresi 35 saniye olarak ölçülmüştür. Trafik uygulamalarında çevrim süreleri aralığı 60 saniye ve 120 saniye olarak alındığından önerilen yöntemin gerçek zamanlı bir uygulamada çalışması için zaman açısından herhangi bir engel bulunmamaktadır.

6) SL-PSO ilk kez trafik sinyali optimizasyonu için kullanılmıştır.

6.2. Edimsel Koşullanma ile Sürü Tabanlı Trafik Sinyal Kontrol Uygulaması ve Sonuçları

Edimsel koşullanma ile sürü tabanlı yöntemlerin yerel noktalara daha az takılmaları ve hedefe daha hızlı ulaşması hedeflenmiştir. Bu kapsamda trafik sinyal kontrol yöntemlerinde optimum faz sürelerinin daha hızlı bir şekilde elde edilmesi için edimsel koşullanma kullanılmıştır. OCA-SL-PSO tabanlı yöntemin prosedürü Algoritma 6.3'te verilmiştir;

Algoritma 6.3. OCA-SL-PSO tabanlı kontrol prosedürü

Adım 1- Başlangıç durumunda faz sürelerini temsil eden çözüm kümesi, faz sürelerini rastgele seçilmiş parçacıklardan oluşur.

Adım 2- Parçacıkların konumları en iyiden en kötüye doğru sıralanır.

Adım 3- Her bir parçacığın en iyi ve en kötü değerlere olan mesafesi hesaplanır.

Adım 4- Mesafeye göre öğrenme oranı hesaplanır.

Adım 5- Sosyal öğrenme gerçekleştirilir.

Adım 6- Uygulama fonksiyonu çalıştırılır.

Adım 7- Elde edilen en iyi değerler simülasyona gönderilir.

Adım 8- Adım 2'ye geri döndürülür.

Edimsel koşullanma ile kullanılarak elde edilen sonuçlar ve bu sonuçların klasik yöntemler ile karşılaştırılması Çizelge 6.6'da verilmiştir. Elde edilen sonuçlara göre OCA-SL-PSO klasik yöntemlerden daha başarısız bir grafik çizmiştir. En iyi süre olan 46 sn'yi yakalayamamıştır. Elde edilen en kötü süre kriterinde klasik PSO'yu geçse de SL-PSO'nun gerisinde kalmıştır.

Çizelge 6.6. OCA-SL-PSO sonuçları

	OCA-SL-PSO	PSO	SL-PSO
	N=10	N=10	N=10
En iyi [sn]	48	46	46
En kötü [sn]	106	113	105
Ortalama [sn]	49,4	50,4	49,2
Süre [dk]	221	239	212

6.2.1. Edimsel Koşullanma ile Sürü Tabanlı Trafik Sinyal Kontrol Uygulaması ve Sonuçları

OCA-SL-PSO'nun yüksek boyutlu ($N > 50$) problemlerde klasik sürü tabanlı yöntemlerin başarısını iyileştirdiği Bölüm 4.2.4'te tartışılmıştır. Bu problemde aynı iyileştirmeyi gösterememesinin sebebi olarak düşük boyutlarda çalışılması gösterilebilir. OCA-SL-PSO, öğrenme tabanlı bir yöntemdir ve öğrenen parçacık sayısının az olması sonucu olumsuz etkilemiştir. Aynı işlemin daha yüksek boyutlarda tekrar edilmesi OCA-SL-PSO'nun performansını arttıracak düşünülmemektedir. Fakat boyut sayısının daha da artması hesaplama süresini arttırmaktadır.

6.3. Adaptif Trafik Sinyal Kontrolünün DQÖ ile Gerçekleştirilmesi

Bu çalışma ile takviyeli öğrenmenin trafik sinyal kontrol yöntemlerindeki başarısı incelenmiştir. Önerilen yöntemde, trafiğin anlık durumuna göre DQÖ kullanılarak optimum faz dizilimi (sabit faz süresi ile) gerçekleştirilmiştir. Bu bölümde önerilen yöntemin kapsamı izole kavşak olduğu için sadece tek bir kavşaktan alınan kuyruk uzunluğu bilgisi girdi olarak kullanılmaktadır. Alınan kuyruk uzunluğu bilgisine göre aktif olan faza faz eylem süresi miktarınca (önceden belirlenmiş ek süre) verilmekte ya da yeni bir faza geçiş sağlanmaktadır.

Bu çalışmada performans kriteri için araç bekleme süresi, araç gecikme süresi, araç seyahat süresi ve karayolundaki araç hızı olmak üzere dört farklı trafik parametreleri kullanılmıştır. Bu bölümde elde edilen sonuçlar, sonraki bölümlerde yapılan çalışmalar için referans niteliğindedir. Tez kapsamında yapılan tüm çalışmalar bu bölümde elde edilen sonuçları iyileştirmeye yöneliktir. Dolayısıyla burada elde edilen sonuçlar bir sonraki bölümde karşılaştırılmak için kullanılmıştır.

Bu bölümde tasarlanan sistemin performansını değerlendirmek için sabit zamanlı trafik sinyal kontrol sonuçları kullanılmıştır. SUMO trafik simülatörü, sabit zamanlı kontrol çıktılarını elde etmek için aynı trafik verileriyle çalıştırılmıştır. Sabit zamanlı trafik sinyal kontrolünün sonuçları Çizelge 6.7'de verilmiştir.

Çizelge 6.7. Sabit zamanlı kontrolör sonuçları.

Sonuçlar	5	10	15	20	25	30	35	40	45
Kalkış gecikme süresi (sn)	32	27	34	35	29	35	32	31	43
Ortalama bekleme süresi (sn)	24	18	22	28	33	39	45	52	57
Ortalama seyahat süresi (sn)	149	141	145	150	156	161	167	175	180
Ortalama hız (m/s)	10	10	10	10	10	9	9	9	9

Bu çalışmada, dokuz farklı sabit zamanlı trafik sinyal kontrol yöntemi denenmiş ve en başarılı olanları DQÖ tabanlı yaklaşımla karşılaştırmak için kullanılmıştır. Çizelge 6.7'ye bakıldığında yeşil faz süresi 10 saniye olan sabit zamanlı uygulama 4 performans kriterinden de en iyi sonucu vermektedir. Bu nedenle, karşılaştırma için sabit zaman süresi 10 saniye olarak ayarlanır.

Çizelge 6.8. DQÖ simülasyon parametreleri

Parametre	Değerler
Trafik ağ tipi	4 kollu kavşak, 4 şerit giriş, 4 şerit çıkış
Kavşak tipi	750 m
Yol uzunluğu	19,44 m/s – 70 km/h
Maksimum hız	1,3 m
İki araç arası mesafe	4,7 m
DQÖ öğrenme oranı	0,1
Öğrenme katsayısı	0,9
ε	0,1
Sigma	0,5
İterasyon	100

Bu çalışmada DQÖ Python platformunda yazılmıştır. Simülasyon için Windows işletim sisteminde SUMO trafik simülatörü kullanılmaktadır. Simülasyonda kullanılan parametreler Çizelge 6.8'de verilmiştir. Elde edilen sonuçlar 10 kez çalıştırılıp ve ortalama değerleri alınmıştır (Çizelge 6.9). Bu çalışmada faz eylem süresi olarak en iyi sabit süre olan 10 saniye alınmıştır.

Çizelge 6.9. DQÖ Sonuçları

Results	DQL
Kalkış gecikme süresi (sn)	33
Ortalama bekleme süresi (sn)	13
Ortalama seyahat süresi (sn)	135
Ortalama hız (m/s)	11

Sabit zamanlı ile DQÖ tabanlı yöntemin sonuçlarının karşılaştırılması Çizelge 6.10'da verilmiştir. Çizelge 6.10'daki sonuçlara göre, DQÖ tabanlı kontrol stratejisi sabit zamanlı stratejiyi dört performans kriterinde de geçmiştir.

Çizelge 6.10. Sabit zamanlı ve DQÖ karşılaştırması

Sonuçlar	Sabit Zamanlı (10sn)	DQL
Kalkış gecikme süresi (sn)	27	33
Ortalama bekleme süresi (sn)	18	13
Ortalama seyahat süresi (sn)	141	135
Ortalama hız (m/s)	10	11

6.4. DQÖ Tabanlı Adaptif Trafik Sinyal Kontrol Sistemleri için Optimum Faz Eylem Süresinin İncelenmesi

Takviyeli öğrenmenin, uyarlanabilir trafik sinyali denetleyicileri için kullanılan en iyi algoritmalarından biri olduğu hem Bölüm 6.3’de yapılan çalışma sonuçları ile hem de literatürdeki çalışmalar ile ispatlanmıştır. Takviyeli öğrenme tabanlı trafik kontrolü ile ilgili birçok başarılı çalışmaya rağmen, adaptif trafik sinyal kontrolünü gerçekleştirmek için en iyi eylemlerin nasıl olması gerektiği konusunda belirsizlik devam etmektedir (Mannion ve ark., 2016; Genders ve Razavi, 2018). Bu bölümde yapılan çalışma ile, adaptif trafik yönetimi için farklı eylem sürelerindeki performans farklılıkları incelenmiştir.

Bu bölümde yapılan çalışma ile DQÖ tabanlı adaptif trafik sinyal kontrol yöntemlerindeki performansının artırılması hedeflenmiştir. Takviyeli öğrenme tabanlı yöntemler, ortamdan gelen geri bildirimine göre faz dizilimi gerçekleştirmektedir. Mevcut fazın faz eylem süresi kadar artırılması ya da bir sonraki faza geçmek alınabilecek eylemlerdir. Faz eylem süresi miktarı, DQÖ tabanlı yaklaşımın performansını doğrudan etkiler. Bu nedenle, bu bölümde yapılan çalışma ile birçok farklı aktif faz süresi ile denenmiş ve kendi aralarında karşılaştırılmış ve en iyi sonucu veren faz süresi değeri araştırılmıştır.

Çizelge 6.11’da sütunlar, faz eylem sürelerini temsil etmektedir. Kısa faz eylem süresi, DQÖ tabanlı yöntemin daha kısa ve daha fazla eylem gerçekleştirmesini sağlar. Uzun faz aksiyon süresi kullanılarak az aksiyon ile uzun vadeli yönetim hedeflenir. Öte yandan, kısa süre daha fazla sarı sinyal süresine neden olur. Daha fazla sarı sinyal süresi nedeniyle araçlar trafikte daha fazla zaman harcar. Bu durum trafikte bekleme sürelerini ve gecikme sürelerini artırır. Yapılan bu çalışma ile, uyarlanabilir trafik yönetimi için farklı eylem sürelerindeki performans farklılıklarını anlamayı amaçlamaktadır. Bu amaçla beş farklı faz eylem süresi on kez test edilmiş ve ortalama değerleri tabloda verilmiştir.

Çizelge 6.11. DQÖ tabanlı yöntemde farklı faz eylem sürelerinin incelenmesi

Sonuçlar	5	10	20	30	40
Kalkış gecikme süresi (sn)	27	33	37	41	31
Ortalama bekleme süresi (sn)	13	13	15	17	17
Ortalama seyahat süresi (sn)	135	135	139	140	140
Ortalama hız (m/s)	11	11	10	10	10

Çizelge 6.11'e göre, en iyi performans kısa süreli eylem süreleri ile (5 sn ve 10 sn) elde edilmektedir. Gecikme süresi kriteri hariç diğer üç performans kriterlerine bakarak, faz eylem süresi arttıkça sistemin performansının düştüğü yorumu çıkarılabilir.

Elde edilen gecikme sonuçları ilginçtir. Araç gecikme değerleri sırasıyla kısa zamanlı faz süresinden uzun faz eylem süresine 27 sn, 33 sn, 37 sn, 41 sn, 31 sn olarak hesaplanmıştır. 5 sn'den 30 sn'ye kadar performans düşmekte, 30 sn'den sonra gecikme değerinde ciddi bir azalma söz konusudur. Bu düşüşün sebebini bulmak için simülasyon görsel olarak çalıştırılmış ve araçların durumu tek tek analiz edilmiştir.

Yapılan analizler sonucunda, 40 sn'lik faz sürelerinde ana yoldaki araçların kavşağa yaklaşırken hızını daha az azalttığı tespit edilmiştir. Faz eylem süresinin yüksek bir rakam seçilmesi, Bölüm 2.1'de ayrıntılı olarak açıklanmış olan Gecikme 2 süresinin artmasına, Gecikme 1 ve Gecikme 3 değerlerinin azalmasını sağlamıştır.

Çizelge 6.12'de kısa ve uzun faz eylem süresinin etkileri karşılaştırılmış ve elde edilen iyileştirme değerleri hesaplanmıştır. 5 sn'lik uygulama kısa ve çok eylem yapılan uygulamayı, 40 sn'lik uygulama ise uzun ve az eylem yapılan uygulamayı temsil etmektedir. Daha fazla sarı sinyal süresine neden olmasına rağmen kısa süreli faz eylem süresi tüm kriterlerde uzun süreli faz eylem süresinden daha iyi sonuç vermiştir. Özellikle bekleme performans kriterlerinde yaklaşık %24'lük bir iyileşme göze çarpmaktadır. Gecikme performans kriterinde %13 ve seyir hızında %10'luk bir iyileşme gözlemlenmiştir. Araçlarının karayolu üzerindeki hızının artması, yolculuk süresini azaltmaktadır. Yolculuk süresindeki %3,5'lik iyileşme bu duruma kanıt olarak gösterilebilir.

Çizelge 6.12. Kısa eylem süresi ile uzun eylem süresinin karşılaştırılması

Sonuçlar	5	40	İyileştirme %
Kalkış gecikme süresi (sn)	27	31	12,9
Ortalama bekleme süresi (sn)	13	17	23,52
Ortalama seyahat süresi (sn)	135	140	3,5
Ortalama hız (m/s)	11	10	10

Çizelge 6.13'te en iyi faz eylem süreli DQÖ uygulaması ile sabit zamanlı kontrolör sonuçları karşılaştırılmıştır. Elde edilen sonuçlara göre, en iyi faz eylem süresi ile yapılan uygulama, ortalama araç seyir hızını %10 arttırmaktadır. Karayolu üzerindeki araçların hızındaki bu iyileşme, ortalama araç seyahat süresini de %4,2 oranında azaltmıştır.

Çizelge 6.13. En iyi faz eylem süresi ile sabit zamanlı kontrol karşılaştırılması.

Sonuçlar	DQN (5 sn)	Sabit Zaman (10 sn)	İyileştirme %
Kalkış gecikme süresi (sn)	27	27	-
Ortalama bekleme süresi (sn)	13	18	27,7
Ortalama seyahat süresi (sn)	135	141	4,2
Ortalama hız (m/sn)	11	10	10

Çizelge 6.13'e göre bir aracın yolda bekleme süresi %27,7'lik bir azalma ile 18 saniyeden 13 saniyeye düşürülmüştür. Ayrıca, bir otomobilin yolda geçirdiği toplam seyahat süresi 141 saniyeden 135 saniyeye düşürülerek 6 saniyelik bir iyileşme gerçekleştirilmiştir.

Burada dikkat edilmesi gereken karşılaştırma yapılırken kullanılan en uygun sabit zaman değeri araştırılmış ve en iyi sabit zaman değeri ile karşılaştırma yapılmıştır. Gerçek trafik koşullarında birçok uygulamada sabit zaman değeri hesaplanırken bu kadar hassas davranılmamaktadır. Bu kriter göz önüne alındığında, DQL stratejisi ile gerçek trafik koşullarındaki kontrol sonuçları çok daha başarılı olacaktır.

6.4.1. Optimum Faz Eylem Süresi ile DQÖ Tabanlı Adaptif Trafik Sinyal Kontrol Sonuçlarının Değerlendirilmesi

Bu çalışma ile takviyeli öğrenme tabanlı adaptif trafik sinyal kontrol çalışmalarında kullanılan optimal faz eylem süresi incelenmiştir. Çalışmadan elde edilen çıkarımlar şu şekilde sıralanabilir;

1) Q-öğrenme algoritması, trafik senaryolarında trafik sinyallerini başarılı bir şekilde optimize etmek için kullanılabilir. Tablolardaki performans ölçütlerindeki iyileştirmeler bu çıkarımın kanıtı olarak verilebilir.

2) Daha fazla sarı sinyal süresine rağmen, çok ve kısa vadeli eylemlerin (5 sn ile DQN), az ve uzun vadeli eylemlerden (40 sn ile DQN) daha verimli olduğu kanıtlanmıştır. Tablo 5'teki performans ölçütlerindeki iyileştirmeler bu çıkarımın kanıtı olarak verilebilir. Bu bulgu, mevcut DQN uygulamaları için önemlidir.

3) Trafik kontrol sistemlerinde uzun süreli eylemler, araçların kavşağa yaklaşırken yavaşladığı gecikme süresini ve yeşil ışık sinyalinden sonra hızlanmayı azaltır. Tablo 4'teki performans ölçütleri bu çıkarımın kanıtı olarak verilebilir.

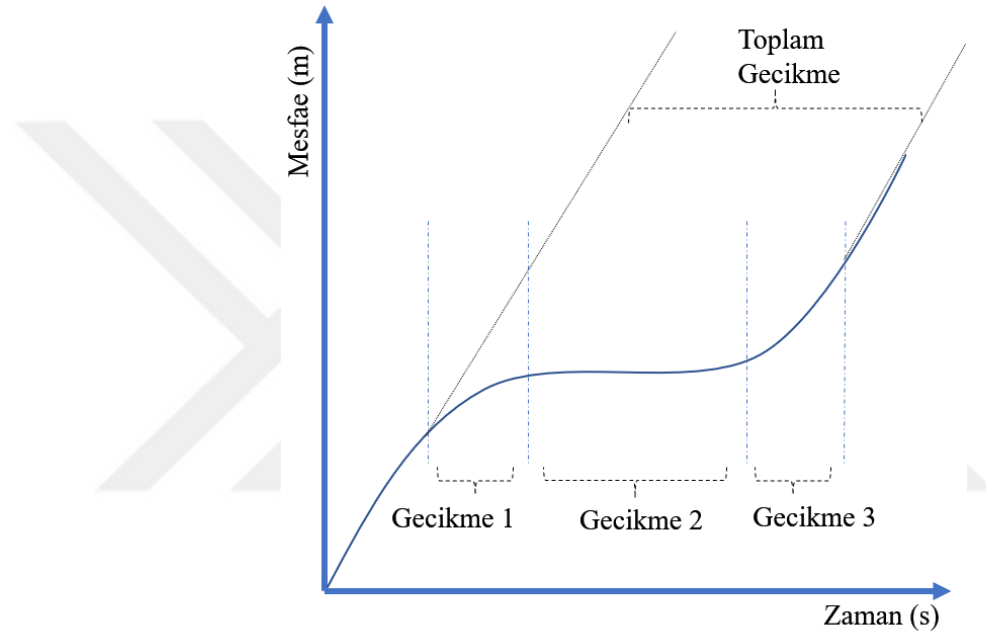
6.5. Koordineli Kavşak Trafik Sinyal Kontrolü Uygulaması

Bu bölümde, kavşaklardaki toplam gecikme süresini minimize etmek için yenilikçi bir yaklaşım önerilmiştir. Bu bölümde yapılan çalışmanın ilham kaynağı, Bölüm 6.4'te yapılan çalışmada elde edilen gecikme performans kriter sonuçlarıdır. Çizelge 6.11'deki gecikme sonuçları göstermiştir ki trafikte gecikmeye neden olan etken sadece kırmızı sinyal süresince beklemek değildir. Araçların sinyal değişimlerinden dolayı yavaşlaması da gecikme süresini etkilemektedir. Bu tez çalışmasında üç gecikme tipinin de minimize edilmesi hedeflenmiştir. Bu kapsamda girdi olarak sadece uygulama yapılan kavşak verileri değil, ardışık kavşak verileri de kullanılmıştır.

Bu bölümde yapılan çalışma ile, koordineli kavşaklardaki toplam gecikme süresini en aza indirmek için DQÖ kullanılmıştır. Toplam gecikme süresi bir sistem performans metriği olarak belirlenmiştir. Bundan dolayı DQÖ yöntemi her türlü

kombinasyonu denemekte ve toplam gecikme süresini azaltmak için en uygun çözümü aramaktadır.

Gecikme süresi en sık kullanılan trafik performans parametrelerinden biridir ve araçların yoldaki kayıp sürelerini belirler. Şekil 6.1'de görüldüğü gibi, kavşaklardaki toplam gecikmenin üç farklı nedeni vardır (Dion ve ark., 2004). Gecikme 1, kavşağa yaklaşırken yavaşlayan araçların neden olduğu gecikme süresidir. Gecikme 2, kırmızı ışık sinyali sırasında beklemeye neden olan en çok bilinen gecikme süresidir. Sonuncusu, yeşil ışık sinyalinden sonra hızlanmak için gereken süre olan gecikme 3'tür.



Şekil 6.1. Kavşaklardaki gecikme

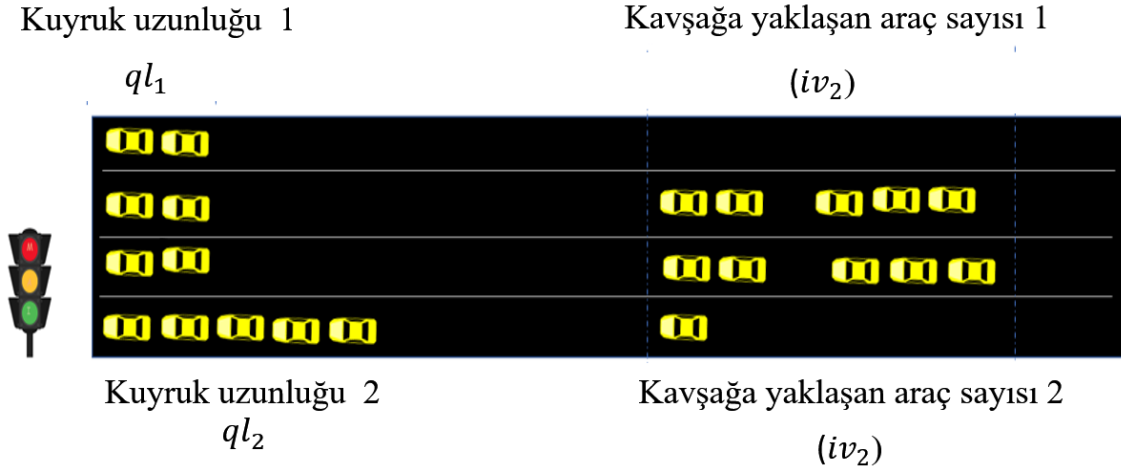
Bu çalışmada üç gecikme tipinin de azaltılması hedeflenmiştir. Bu kapsamda, Bölüm 5.2.2.1’de açıklanan takviyeli öğrenme kavramlarından farklı olarak, bu bölümde önerilen strateji, sadece uygulama yapılan kavşak bilgisini kullanmakla kalmaz, aynı zamanda bitişik kavşaklardan gelen araç verilerini de girdi olarak kullanır.

$$S_1 = ql_1 + iv_1 \quad (6.1)$$

$$S_2 = ql_2 + iv_2 \quad (6.2)$$

Bölüm 5.2.2.1’de tanıtilen kontrol stratejisinde durum bilgisi olarak kavşaktaki araç sayısı dikkate alınmıştır. Bunun için her yoldaki kuyruk uzunlukları ve gelen araçlar dikkate alınır. Bir yol için iki farklı durum vektörü vardır. İlk vektör (S_1), araçları sola

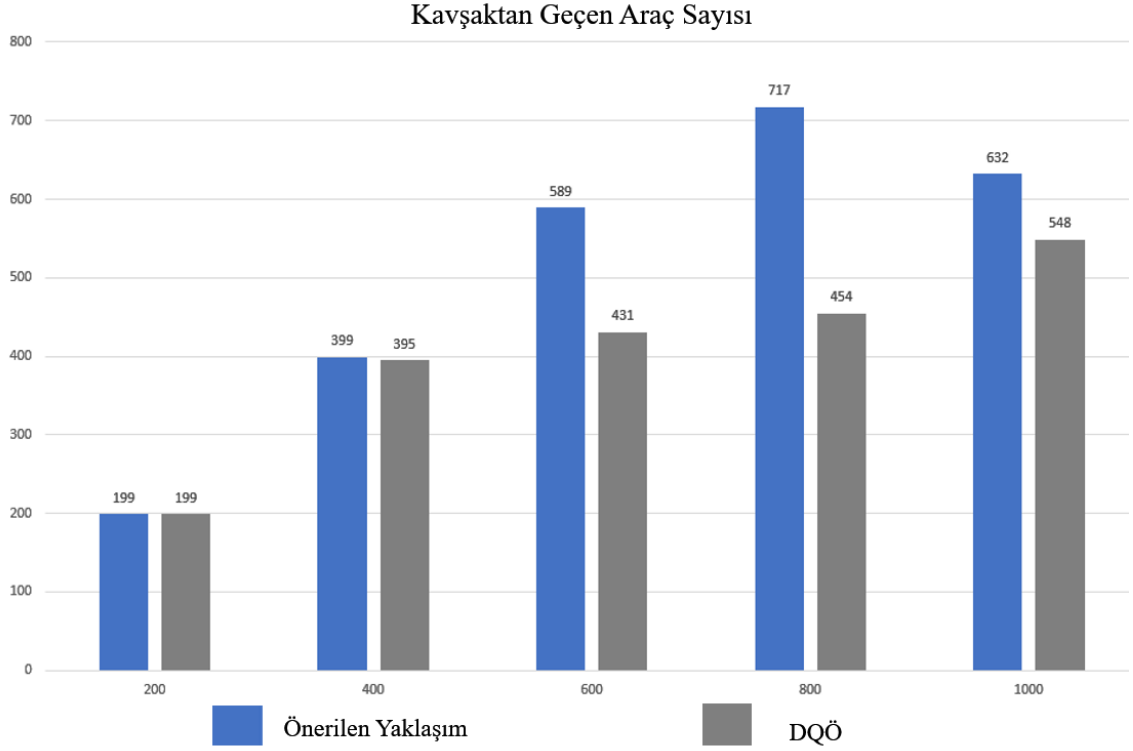
dönüşle temsil eder (Denklem 6.1). İkincisi (S_2) düz veya sağa dönmeyi temsil eder (Denklem 6.2). Denklem 6.1 ve Denklem 6.2'deki kuyruk uzunlukları (ql_1, ql_2) ile kavşakta 0,1m/s hızın altında bekleyen araçlar temsil edilmektedir. Şekil 6.2'de, kuyruk uzunluğu ile kavşağa yaklaşan araç sayısını görsel olarak verilmiştir.



Şekil 6.2. Koordineli kavşak trafik sinyal uygulaması

Önerilen yenilikçi kavşak kontrol stratejisi sadece belirli trafik koşulları için değil, tüm trafik rejimleri göz önünde bulundurularak tasarlanmıştır. Bundan dolayı performans testi yapılırken beş farklı trafik rejimi ile test gerçekleştirilmiştir. Koordineli kavşaklara farklı sayıda araç gönderilerek farklı trafik rejimleri uygulanmıştır. 200 ve 400 araçlık uygulamalar, doymamış trafik rejimlerini temsil etmektedir. Doymuş trafik rejimi 600 araba ile temsil edilmektedir. 800 araç ve 1000 araç ile aşırı doymuş trafik rejimleri temsil edilmektedir. 200 araba ve 1000 araba, sırasıyla çok düşük ve çok yüksek yoğunluklu trafik rejimlerini temsil etmektedir.

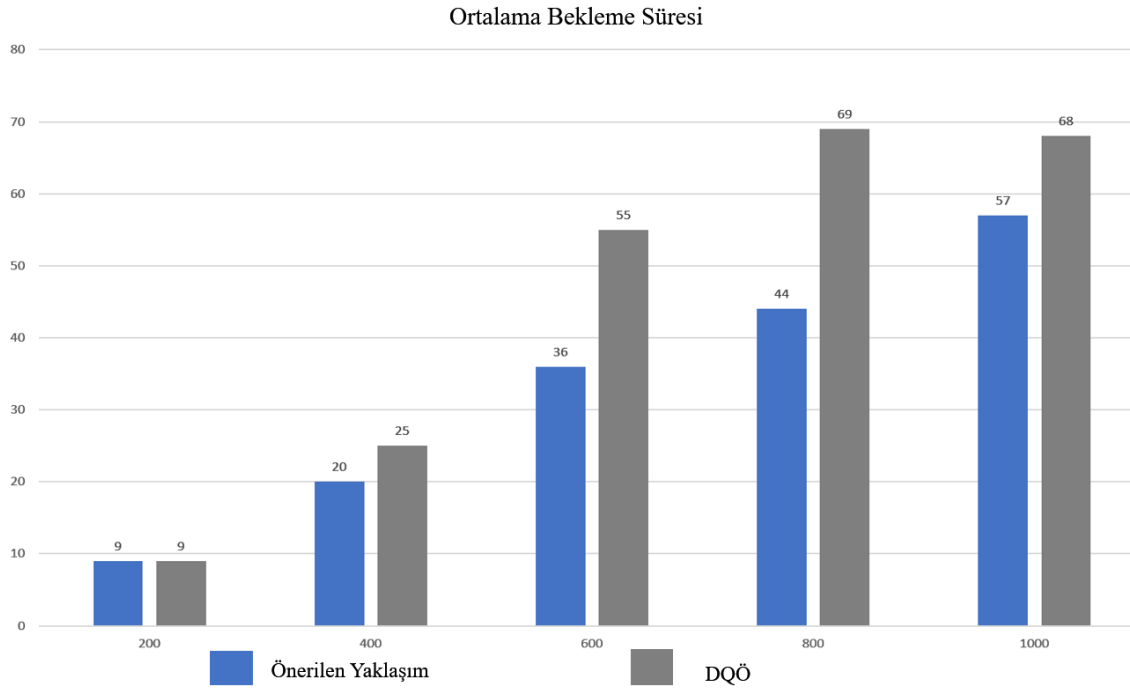
Önerilen yenilikçi yaklaşımla, DQÖ tabanlı trafik kontrol yöntemlerinin performansının artırılması hedeflenmiştir. Bundan dolayı önerilen yaklaşım, klasik DQÖ algoritmasının sonuçları ile karşılaştırılmıştır.



Şekil 6.3. Kavşak kapasitesi

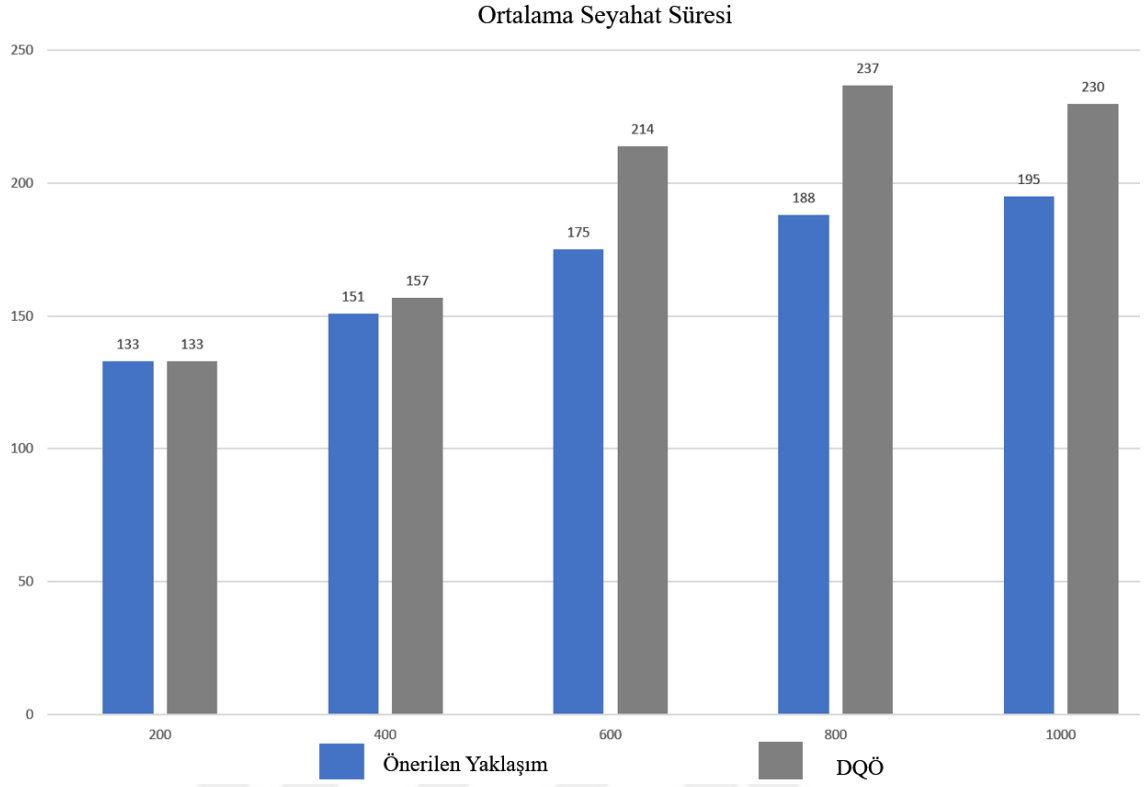
Şekil 6.3'te bir simülasyon süresi boyunca kavşaktan geçen araç sayısı görsel olarak gösterilmiştir. Örneğin, simülasyon programına 800 araba göndermek için komut verilmiştir. Önerilen yöntemle belirlenen süre zarfında kavşaktan 717 araç geçerken, klasik teknik ile 454 araç kavşaktan geçmektedir. Her iki model de aşırı düşük trafik rejimlerinde (200 araç) başarılıdır. 400 araçlık uygulama da küçük bir iyileştirme (5 araç) söz konusudur. Önerilen yöntemin dokunuşu, 600 araçlı uygulamada daha belirgindir. Klasik yöntemle 600 araçtan 170 araç kavşağı geçemezken, önerilen teknikle sadece 11 araç geçememiştir. Şekil 6.3'te verilen grafik, kontrol modelimizin kavşağın kapasitesini arttırdığının bir delilidir.

Şekil 6.4 ile ortalama araç bekleme süresinin karşılaştırma sonuçları verilmiştir. Önerilen yöntemle arabaların tüm trafik koşullarında daha az beklediği görsel olarak gösterilmiştir. Doymuş rejimde (600 araçlı), beklemeye ait performans kriterleri önerilen kontrol stratejisi ve klasik kontrol stratejisi ile sırasıyla 36 saniye ve 55 saniyedir. Çok düşük ve aşırı yüksek yoğunluklu trafik rejimlerinde, kontrol yöntemimiz ile araçlar sırasıyla 5 saniye ve 25 saniye daha az beklemektedir. Önerilen kontrol modeli ile, az doymuş, doymuş ve aşırı doymuş trafik rejimlerinin bekleme değerlerinde sırasıyla %5, %35 ve %37'lik iyileştirmeler sağlanmıştır.



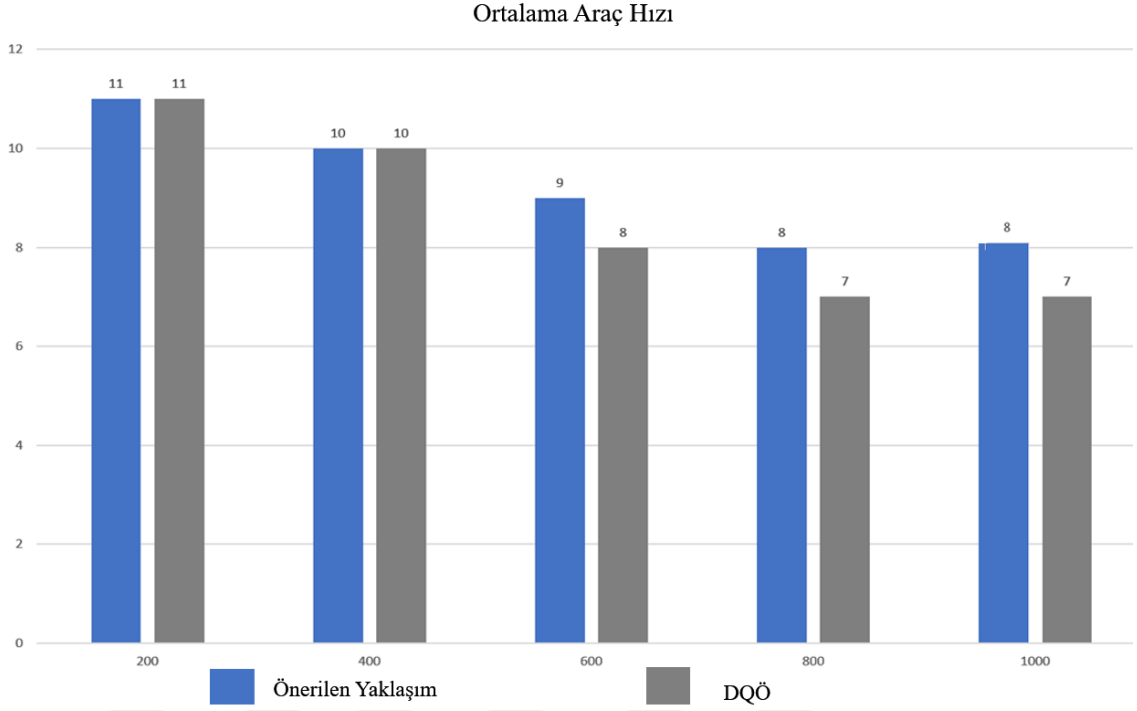
Şekil 6.4. Ortalama bekleme süresi karşılaştırması

Önerilen yaklaşımla elde edilen Ortalama Araç Seyahat Süresi sonuçları Şekil 6.5'te verilmiştir. Şekil 6.5'e göre tüm trafik rejimlerinde, önerilen strateji klasik sisteme göre daha üstün bir sonuç vermektedir. Yöntemler arasında doymuş ve aşırı doymuş rejim için sırasıyla 39 sn ve 48 sn'lik belirgin bir fark vardır. Doymamış rejim için bu fark değeri 5 sn'dir. Şekil 6.5, önerin kontrol modelinin ortalama yolculuk süresini azalttığının kanıtı olarak verilebilir.



Şekil 6.5. Ortalama araç seyahat süresi

Şekil 6.6'da ortalama seyir hızı değerleri (m/sn) karşılaştırılmıştır. Önerilen yaklaşım ile doymuş ve aşırı doymuş rejimlerde daha iyi sonuçlar elde edilmiştir. Doymuş ve aşırı doymuş trafik koşullarında 3,6 km/sa'e eşit olan 1 m/sn'lik bir iyileşme vardır.



Şekil 6.6. Karayolu üzerindeki ortalama araç hızı

Önerilen strateji, kavşak yönetimi için bitişik kavşaklardan gelen bilgileri de dikkate alır. Bu strateji sayesinde, araçlar daha az yavaşlamaya ihtiyaç duymaktadır. Ortalama seyahat hızındaki iyileştirmenin sebebi araçların daha az yavaşlamasıdır.

Bu çalışmada, çok düşük trafik rejimi ve çok yüksek trafik rejimi gibi olağanüstü durumlar da test edilmiştir. Sonuçlarda da görüldüğü gibi, düşük yoğunluklu trafik koşullarında (200 araç), her iki kontrol tekniğinde de elde edilen tüm sonuçlar eşittir. Öte yandan aşırı yoğun trafik rejiminde ise (1000 araç) kavşak kapasitesinden daha fazla araç gönderildiği için her iki algoritma da trafiği düzgün yönetememektedir.

6.5.1. Önerilen Yenilikçi Yaklaşımın Değerlendirilmesi

Bu çalışmada koordineli kavşaklar için yenilikçi bir yaklaşım önerilmiştir. Önerilen teknik, bitişik kavşakların verilerini ve mevcut kavşak parametreleri ile birlikte kullanılmaktadır.

Bu araştırmada önerilen yöntemin sonuçları klasik DQL trafik sinyal denetleyicisi ile karşılaştırılmıştır. DQÖ'nin en iyi modern trafik kontrol yöntemlerinden biri olduğu birçok çalışma ile ispatlanmıştır (Yau ve ark., 2017; Haydari ve Yılmaz, 2020; Shaikh ve ark., 2020). Gerçek hayatta trafik sinyali faz değeri klasik kontrol yöntemleri kullanılarak

hesaplanır. Bu kriter dikkate alındığında, önerilen strateji ile gerçek yaşam sonuçları çok daha başarılı olacaktır. Çalışmanın çıkarımları şu şekilde sıralanmıştır;

1) Önerilen yaklaşım, tüm trafik rejimleri için trafik senaryolarında gerçek zamanlı trafik sinyallerini başarılı bir şekilde optimize etmek için kullanılabilir.

2) Önerilen yöntem, kavşağın kapasitesini artırmaktadır. Şekil 6.3'te verilen performans ölçütlerindeki iyileştirmeler bu çıkarımın kanıtı olarak verilebilir. Bu iyileştirme, mevcut karayolunun potansiyelini önemli ölçüde artırmaktadır.

3) Doygun ve aşırı doymuş trafik akışlarında araç başına ortalama bekleme süresi %35 ve %37 oranında azaltılır. Bu iyileştirme ile dolaylı olarak karayolunu kullanan araç sayısı artırmaktadır.

4) Önerilen yaklaşım, doymuş ve aşırı doymuş trafik akışları için ortalama araç yolculuk süresini sırasıyla 39 saniye ve 48 saniye azaltır. Bu iyileştirme kavşağı kullanan araç sayısını artırmaktadır.

5) Önerilen yöntemle araçların kara yolu üzerindeki hızı artırılmaktadır.

6) Bu çalışmada, eylem sürelerinin değerlendirilmesi için gerçek zamanlı kontrol modeli kullanılmıştır. Gerçek trafikte çevrim süreleri genellikle 60 sn ile 120 sn arasında alınır. Önerilen yöntem ile bu süre içerisinde 12-24 işlem gerçekleştirilebilmektedir. Bu, önerilen yöntemin trafik sinyal sürelerini gerçek zamanlı olarak başarılı bir şekilde kontrol edebileceğinin kanıtıdır.

6.6. Trafik Sinyal Ağı Kontrolü Uygulaması

Bu bölümde, trafik sinyal ağının kontrolü için kullanılan DQÖ tabanlı yöntemin sonuçlarına ve performans analizine yer verilmiştir. Bu bölümde merkezi bir yönetim yerinde dağıtılmış yönetim merkezi kullanılmıştır. Trafik sinyal ağında her bir kavşak bağımsız bir küme gibi optimizasyon gerçekleştirmiştir. Bölüm 5.2.2.1'de ayrıntılı olarak açıklanan takviyeli öğrenme tabanlı kontrolör kullanılmıştır.

Bu bölümde yapılan çalışmada, dağıtılmış trafik sinyal kontrolörleri kullanılmıştır. 5×5 boyutlu trafik ağındaki her bir kavşak izole bir kavşak gibi davranarak yönetim gerçekleştirmiştir. Kavşaklar arası herhangi bir senkronizasyon ve koordinasyon söz konusu değildir. Her bir kavşak kontrolörü klasik bir DQÖ tabanlı strateji (Bölüm 6.3'te ayrıntılı olarak açıklanmıştır) ile yönetilmekte ve girdi olarak sadece kendi kavşak bünyesindeki trafik parametrelerini kullanmaktadır.

Bu çalışmada, Bölüm 5.1’de tanıtilan Senaryo 3 adlı 5×5 Manhattan tipi ızgara trafik ağı kullanılmıştır. Gerekli tüm kodlar, Python platformunda yazılmıştır. Simülasyon için Ubuntu işletim sisteminde SUMO trafik simülatörü kullanılmaktadır. Simülasyon, I7-7500U Intel 2.7 GHz işlemci ve 16 GB RAM kullanılarak gerçekleştirilir. Elde edilen sonuçlar 25 kez çalıştırılmış ve ortalama değerleri alınmıştır. Simülasyonda kullanılan parametrelere Çizelge 6.14’te yer verilmiştir.

Çizelge 6.14. Simülasyon parametreleri

Parametre	Değeri
Trafik ağ tipi	5x5 ızgara tipi
Kavşak tipi	4 kollu, 3 şerit giriş, 3 şerit çıkış
Yol uzunluğu	500 m
Maksimum hız	13,89 m/s – 50 km/h
İki araç arası mesafe	1,3 m
Araç uzunluğu	4,7 m
DQÖ öğrenme katsayısı	0,1
DQÖ indirim katsayısı	0,9
ϵ	0,1
Sigma	0,5
İterasyon	100

Çizelge 6.15. Farklı faz eylem süreleri ile DQÖ tabanlı dağıtılmış yönetim sonuçları

Sonuçlar	5 sn	10 sn	15 sn	20 sn	25 sn	30 sn	35 sn	40 sn
Ortalama Bekleme Süresi (sn)	102.79	108.54	188.06	271.46	320.73	295.93	274.82	292.0
Ortalama Seyahat Süresi (sn)	307.63	282.55	363.77	449.81	500.46	468.57	441.92	448.12
Ortalama Hız (m/sn)	6.82	6.39	4.95	4.01	3.63	3.93	4.13	3.96
Ortalama Durma Sayısı	6.53	4.90	4.82	4.83	4.51	3.65	3.05	2.86

Çizelge 6.15, senkronize edilmemiş DQÖ tabanlı kontrolörlerin sonuçlarını sunmaktadır. Faz eylem süresinin (faz eylem süresi=yeşil faz süresi) doğrudan kontrolör performansını etkilediği önceki bölümlerde yapılan çalışmalar ile ispatlanmıştı. Bu nedenle simülasyon sekiz farklı faz eylem süresi ile çalıştırılmıştır. Çizelge 6.15'te sütunlar, faz eylem süresindeki değişikliği temsil eder. Kısa faz eylem süresi, önerilen yöntemin daha kısa ve daha fazla işlem yapmasına izin verir. Uzun aksiyon süresi kullanılarak az aksiyon ile uzun vadeli yönetim hedeflenir.

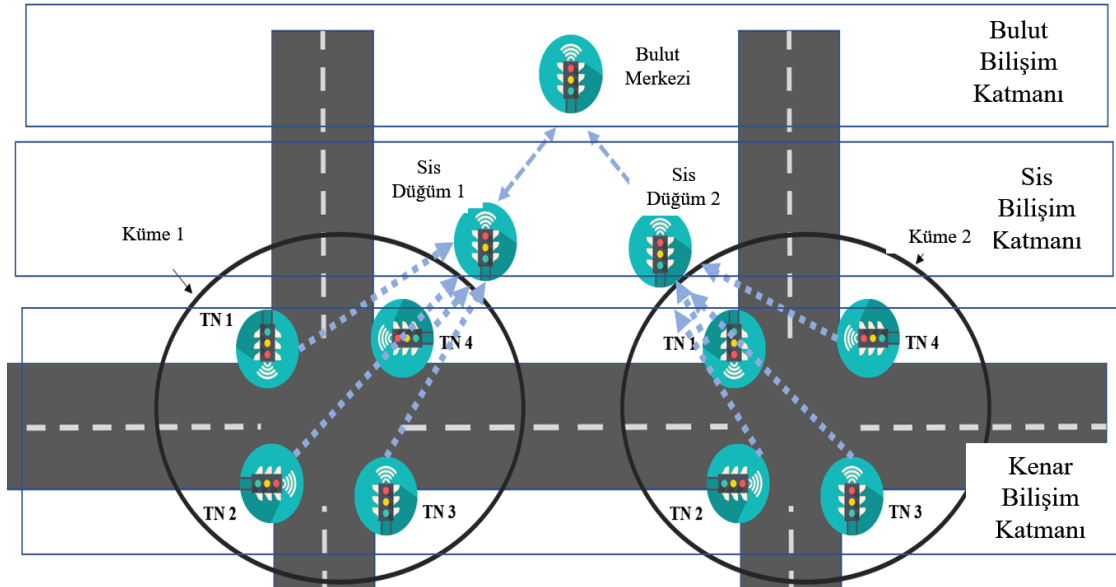
Çizelge 6.15'e göre, trafik sinyal ağında da kısa vadeli eylemler (5 sn veya 10 sn ile DQN), daha fazla sarı sinyal süresine neden olmasına rağmen, uzun vadeli eylemlerden daha efektif sonuçlar vermektedir. Bekleme süresi, seyahat süresi ve seyahat hızı sonuçları bu çıkarımın kanıtı olarak verilebilir. Ortalama bekleme süresi değeri, faz eylem süresi artışı ile neredeyse üç katına çıkmakta, ortalama seyahat süresi parametresi, 282 sn'den 458 sn'ye yükselmekte ve hızı parametresi 6,82'den 3,96'ya düşmektedir.

Çizelge 6.15'teki en ilginç sonuç araçların ortalama durma sayısıdır. Durma kriteri, diğer üç performans kriterine zıt sonuçlar vermektedir. Sonuçlar, araçların uzun vadeli eylemlerde daha az durduğunu açıkça göstermektedir. Ancak uzun süreli aksiyonlarda araç durduğunda daha fazla süre bekler. Bu nedenle, daha az durmasına rağmen daha fazla bekleme süresi ortalama seyahat süresini artırmaktadır.

6.7. Nesnelerin İnterneti Platformunda Adaptif Trafik Sinyal Ağı Uygulaması

Ağıdaki trafik sinyal sayısı arttıkça, kontrolörün güçlü bilgi işlem yeteneklerine duyulan ihtiyacı da artmakta, trafik sinyal optimizasyon işlemi daha da karmaşıklıklaştırmaktadır. Mevcut merkezi kontrol yöntemlerinin performansı, uzun yanıt nedeniyle bu karmaşıklığı çözmek için yeterli değildir. Bu tezde gerçek zamanlı optimum kontrol için nesnelerin interneti tabanlı uyarlanabilir bir trafik kontrol sistemi önerilmiştir. Bu bölümde yapılan çalışma ile, uyarlanabilir trafik kontrol sistemlerinin veri aktarımını ve gerçek zamanlı veri işleme yeteneklerini geliştirmek amaçlanmaktadır.

Önerilen model Şekil 6.7’de gösterildiği gibi üç farklı katmandan oluşmaktadır. El alt katman olan fiziksel katman, kavşaktan trafik verilerini toplar ve bir üst katman olan sis bilişim katmanına yönlendirir. Sis bilişim katmanı, kendi kümesi içinde yerel bir optimizasyon işlemi gerçekleştirir. Bu optimizasyon işlemi için DQÖ algoritması kullanılmaktadır. Sis katmanındaki DQÖ algoritması, trafiğin anlık durumuna göre faz sırasına karar verir. Faz eylem süresi miktarınca aktif faza ek süre verilir veya yeni bir faza geçilir. Faz eylem süresi, sis hesaplama katmanının performansını doğrudan etkilemektedir. En üst katmanda yer alan bulut merkezi, komşu kavşak trafik verilerini kullanarak faz eylem süresine karar verir.



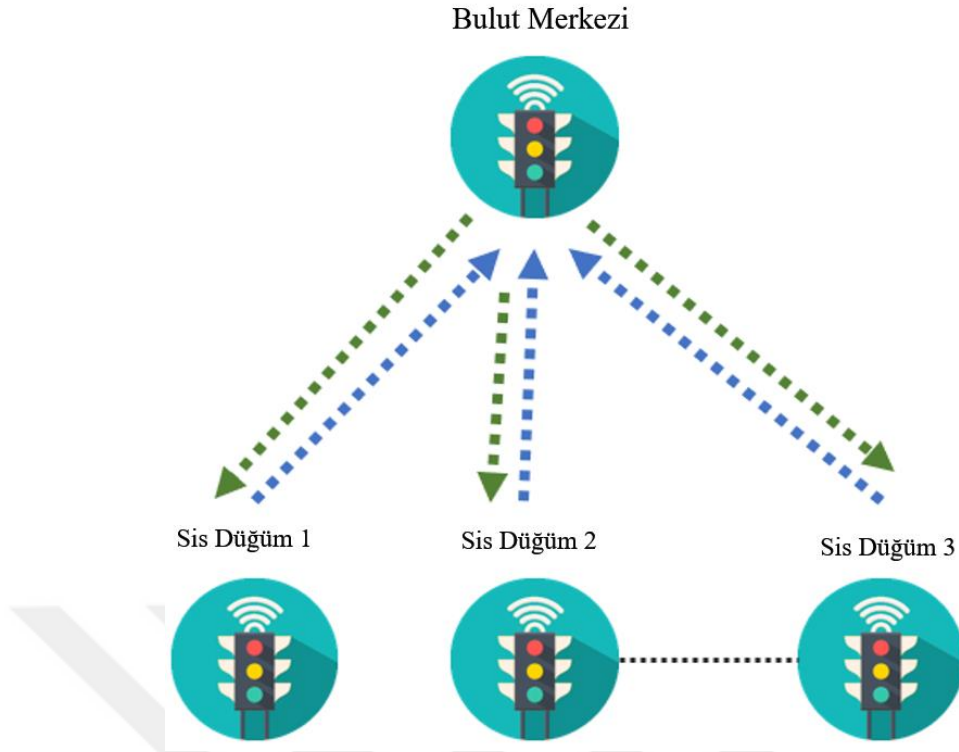
Şekil 6.7. Trafik sinyal ağı senaryosu

Trafik düğümleri, trafik ortamından gerçek zamanlı verileri toplayan ve sis düğüme ileten uç cihazlardır. Trafik düğümleri fiziksel katmanda konuşlandırılır.

Trafik düğümlerinin iki görevi vardır. Birincisi, trafik ortamından gerçek zamanlı veriler (sıra uzunluğu, yaklaşan araç sayısı, araçların hızı, çevrim süresi, faz süresi) toplamaktır. İlk görev için trafik kameraları veya dedektörler kullanılabilir. İkinci görev, trafik parametrelerini sis düğümüne göndermektir. 4G, WiFi, WLAN gibi iletişim teknolojileri ve diğer kısa mesafeli iletişim teknikleri, fiziksel ve sis bilgi işlem katmanı arasında iletişim kurmak için kullanılır.

Sis hesaplama katmanı, sis düğümlerinden oluşur. Sis düğümleri, güçlü bilgi işlem yeteneklerine sahip trafik sinyali denetleyicileridir. Sis düğümleri, trafik düğümlerinden gelen trafik verilerini analiz eder ve DQÖ kullanarak kavşak trafik sinyali programını planlar. Sis düğümleri harekete karar verir ve faz eylem süresi ile bazı yönler için yeşil fazı etkinleştirir. Sis hesaplama katmanındaki optimizasyon, tüm trafik ağlarını değil, yalnızca bir kümeyi dikkate alan yerel optimizasyondur. Her sis düğümü, bulut merkezine bir rapor gönderir.

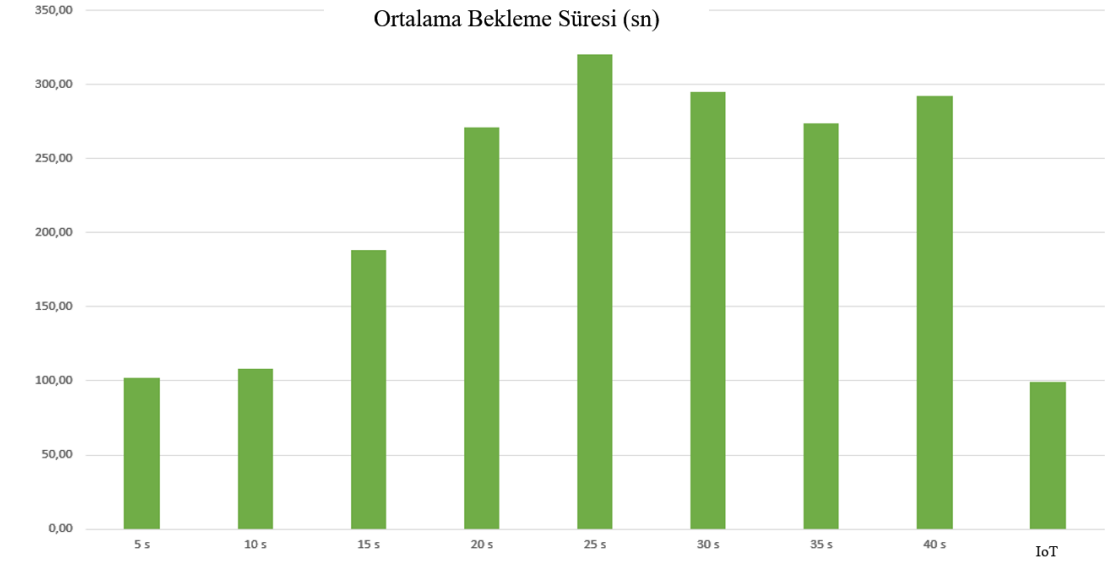
Bulut bilgi işlem katmanı, trafik sinyali kontrol sisteminin merkezidir. Bulut merkezi, sis düğümlerinden gelen verileri dikkate alarak tüm ağı optimize etmek için kullanılır. Şekil 6.8, bulut merkezi ve sis düğümleri arasındaki iletişimi göstermektedir. Mavi çizgi, sis düğümlerinden toplanan trafik parametresini temsil eder. Yeşil çizgi, bulut merkezinden sis düğümüne giden sinyali temsil eder. Bulut merkezi, bitişik küme trafik parametrelerini kullanarak küme için faz eylem süresine karar verir. Yeşil sinyal, bulut merkezi tarafından kararlaştırılan faz eylem süresini içerir. Örneğin, küme 2 için faz eylem zamanını belirlemek için bulut merkezi sadece küme 2'yi değil, aynı zamanda küme 1 ve küme 3 olan bitişik kümelerle ilgili trafik parametrelerini de kullanır. Böylece bulut merkezi, bitişik kümeler arasındaki koordinasyonu sağlar.



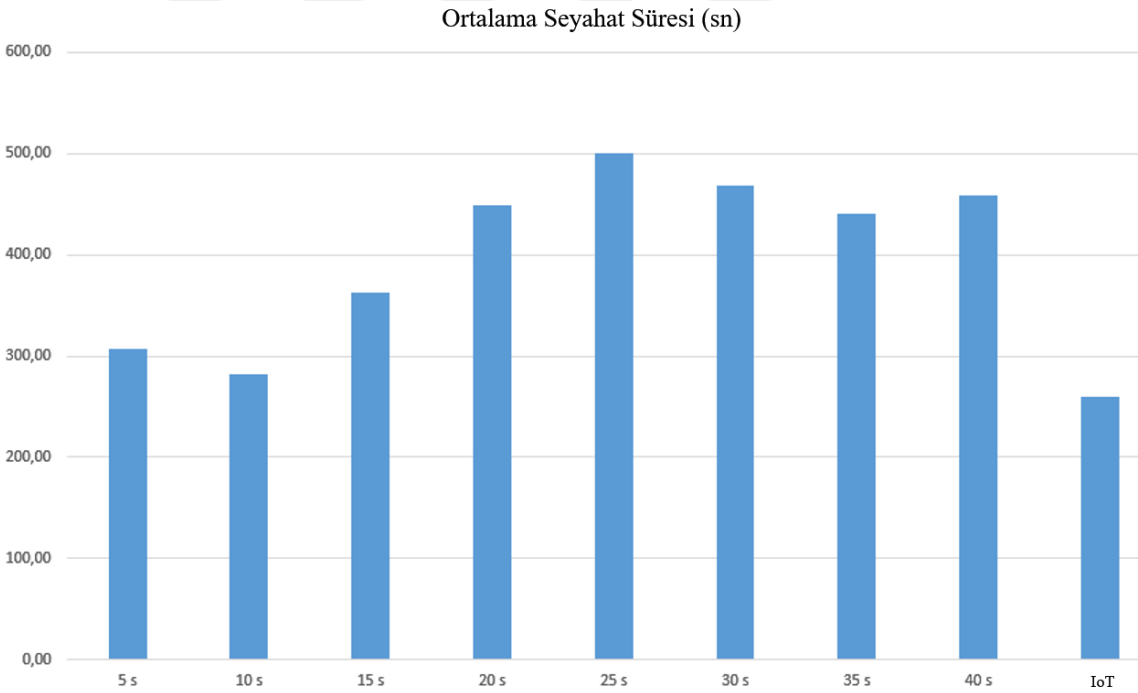
 ekil 6.8. Sis d ğ m  ile bulut merkezi arasındaki haberle me.

Bu  alı mada, d ğ mleri ba lamak i in web tabanlı a ık API IoT kaynak bilgi platformu olan thingspeak (<https://thingspeak.com/>) kullanılmı tır. Thingspeak, t m kav ak verilerini bir araya getirmeye yardımcı olur ve kullanıcıların trafik sinyallerini kontrol etmesine izin verir.

 nerilen nesnelerin interneti tabanlı modelin performansı, B l m 6.6'da yapılan uygulamalarla kar ıla tırılmı  ve  alı manın  ıktıları g rsel olarak verilmi tir. Bekleme s resine ait performans  ıktıları  ekil 6.9'da verilmi tir. Klasik DQ  sonu larında en iyi de er 102.9 sn'dir.  nerilen model i in ortalama ara  bekleme s resi 99.23 sn'dir. Bu sonu lara g re, bekleme s resi performans kriterinde yaklaşık %3'l k bir iyile me vardır.

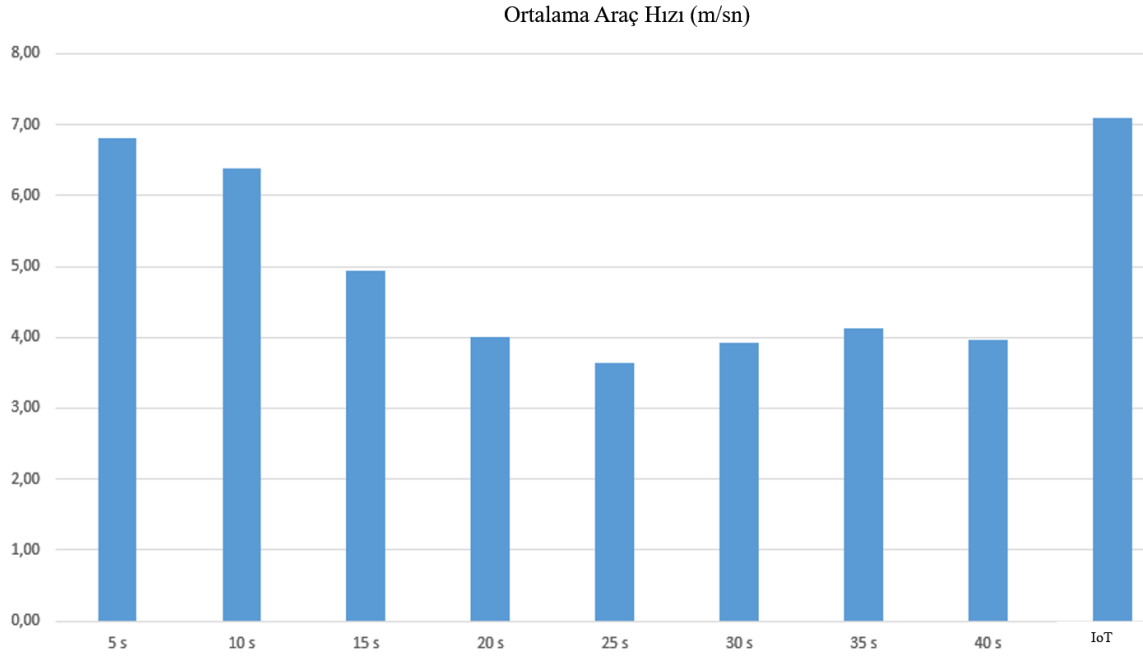


Şekil 6.9. Bekleme süresine ait performans çıktıları



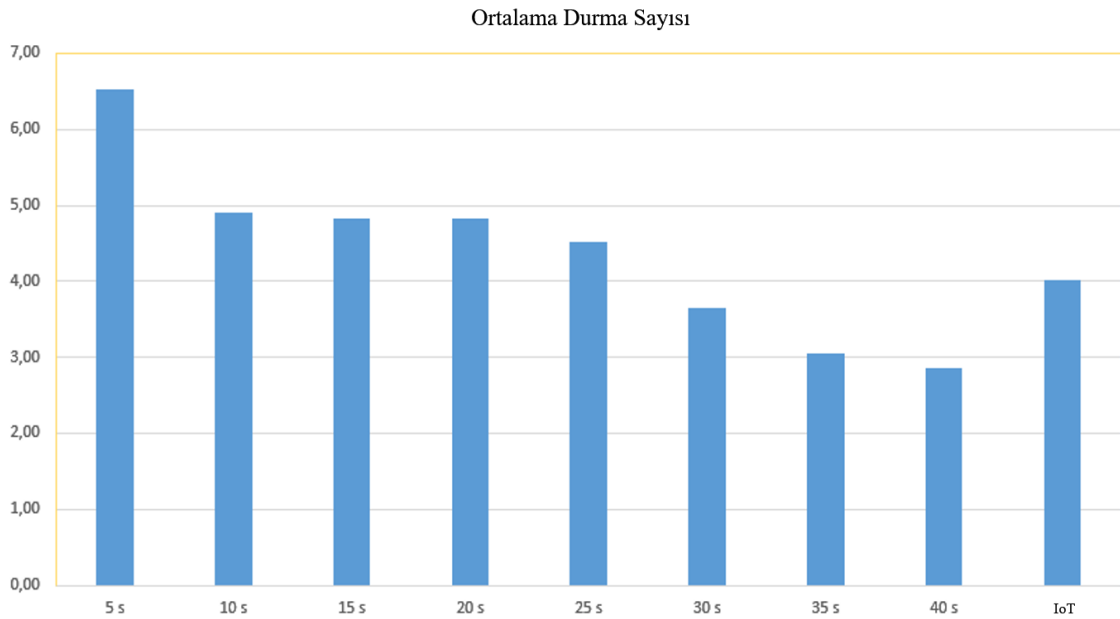
Şekil 6.10. Ortalama seyahat süresine ait performans çıktıları

Şekil 6.10, her bir araç için seyahat süresi performans kriterini göstermektedir. Seyahat süresi, duraklar, beklemler ve yoldaki gecikmeler dahil olmak üzere, başlangıç noktasından bitiş noktasına kadar geçen tüm süreyi temsil eder. En iyi performans değeri olan 260.55 sn önerilen modele aittir. Önerilen model ile seyahat süresi performans kriterinde yaklaşık %10'luk iyileştirme yapılmıştır.



Şekil 6.11. Ortalama araç hızı

Ortalama araç hızına ait performans çıktıları Şekil 6.11'de verilmiştir. Seyahat hızı değeri, her bir araç için güzergâh uzunluğunun seyahat süresine bölünmesiyle hesaplanmaktadır. Önerilen kontrol stratejisi ile hız kriteri 0.28 m/sn'lik bir iyileşme ile 7.10 m/sn'ye çıkarılmıştır.



Şekil 6.12. Durma sayısı kriteri

Ortalama durma sayısı kriteri sonuçları Şekil 6.12'de verilmiştir. Durma değeri kısa süreli eylem süresinden uzun süreli eylem süresine doğru azalmaktadır. Durma sayısı, durma süresi parametresi ile toplam seyahat süresi için kritik bir parametredir. Durma süresi yüksek ise, durma sayısı az olsa da toplam seyahat süresi artar. Dolayısıyla Şekil 6.10 ve Şekil 6.12, önerilen yaklaşım ile durma süresinin azaltılarak toplam seyahat süresini en aza indirdiğinin kanıtı olabilir.

6.7.1. Nesnelerin İnterneti Platformunda Adaptif Trafik Sinyal Kontrol Değerlendirmesi

Bu araştırma ile, nesnelerin interneti bilişim mimarilerine dayalı koordineli uyarlanabilir trafik sinyali kontrolünü önerilmiştir. Önerilen modelin kapsamı trafik sinyal ağıdır. Önerilen yöntem ile adaptif trafik kontrol sistemlerinde veri iletimi koordine edilmesi ve yanıt gecikmesinin minimize edilmesi amaçlanmıştır. Deneysel sonuçlar, önerilen modelin, klasik stratejilerden daha verimli olduğunu göstermiştir. Önerilen kontrol yönteminin değerlendirilmesi ve test edilmesi için SUMO trafik simülatörü kullanılmıştır. Çalışmadan elde edilen çıkarımlar şu şekilde verilebilir;

1) Nesnelerin interneti ve Derin Q-Öğrenme, trafik sinyallerini başarılı bir şekilde optimize etmek için kullanılabilir. Şekil 6.9, Şekil 6.10, Şekil 6.11 ve Şekil 6.12'de verilen sonuçlar bu çıkarıma kanıt olarak verilebilir.

2) Karayolu üzerindeki araçların ortalama hızlarında %10'luk bir artış sağlanmıştır. Bu iyileştirme toplam seyahat süresini azaltmakta ve mevcut karayolunun potansiyelini önemli ölçüde artırmaktadır.

3) Trafik sinyal ağlarında da kısa vadeli eylemlerin, uzun vadeli eylemlerden daha verimli olduğu kanıtlanmıştır. Bu bulgu, mevcut DQN uygulamaları için önemlidir.

4) Bu çalışma, nesnelerin internetinin büyük ölçekli trafik ağları için kullanılabileceğini kanıtlamaktadır.

7. TARTIŞMA VE ÖNERİLER

Trafik ağının efektif kontrolü günümüz şehirlerinin en kritik konularından biridir. Bu konuda yapılan çalışmaların ve yatırımların sayısı her geçen gün artmaktadır. Bu artışa rağmen halen sorun devam etmekte ve optimum trafik sinyal kontrol arayışı devam etmektedir.

Bu tez ile yapılan araştırmalarda gerçek zamanlı adaptif trafik sinyal kontrolü yöntemlerinde iyileştirme sağlamak hedeflenmiştir. Bu çalışma kapsamında önerilen yöntemler tasarlanırken, ucuz maliyet, kolay kurulum ve uygulanabilir olması kriterleri dikkate alınmıştır. Önerilen yöntemler girdi olarak kavşaktaki kuyruk uzunluğunu ya da kavşağa gelen araç sayısını kullanmaktadır. Bu parametreler halihazırda birçok kavşakta kullanılan yüksek hesaplamalı trafik kamerası ile elde edilmektedir. Ondan dolayı önerilen yöntemlerin uygulanması için herhangi bir ek donanım ihtiyacı söz konusu değildir.

Önerilen modeller ile sadece ulusal trafik problemleri değil, evrensel trafik problemlerinin çözümü de hedeflenmiştir. Bu bağlamda simülasyonda kullanılan parametreler bu konuda rehber niteliği olan kitaplar (Orcutt Jr, 1993; Papacostas ve Prevedouros, 1993; HCM, 2000) incelenerek belirlenmiştir.

Tez kapsamında yapılan ilk araştırma, optimum trafik sinyal kontrolünün sezgisel sürü tabanlı yöntemler ile gerçekleştirilmesidir. İlk olarak klasik bir yöntem olan PSO başarıyla uygulanmıştır. Yapılan uygulama da parçacık sürü algoritmasının hesaplama süresinin fazla olduğu tespit edilmiştir.

Gerçek zamanlı bir optimizasyon işleminde bu hesaplama süresinin minimize edilmesi gerekmektedir. Bu sebeple, PSO'nun yakınsama hızı yüksek bir varyantı olan SL-PSO trafik sinyal kontrol problemine uygulanmıştır. Elde edilen sonuçlar, SL-PSO'nun daha kısa sürede daha etkili sonuçlar verdiğini göstermektedir. Bu çalışma SL-PSO'nun trafik sinyal kontrol yöntemlerine uygulandığı ilk çalışmadır ve sonuçları ilgili literatüre katkı vermektedir (Celtek ve ark., 2020).

SL-PSO'nun yakınsama hızını daha da arttırmak adına sosyal bilimlerde sık kullanılan edimsel koşullanmadan esinlenen bir yaklaşım (OCA-SL-PSO) kullanılmıştır. Önerilen yaklaşım ile SL-PSO'nun sonuçları iyileştirilmiş ve ilgili literatüre sunulmuştur (Çeltek ve Durdu, 2020).

Sezgisel optimizasyon yöntemlerinin ihtiyaç duyduğu hesaplama süresi gerçek zamanlı adaptif trafik sinyal kontrolü için aşılması gereken bir problemdir. Hesaplama süresini daha da azaltmak için TÖ tabanlı yöntemler araştırılmıştır.

Tez çalışması kapsamında TÖ, izole kavşak, koordineli kavşak ve trafik ağına uygulanmış ve başarısı ispatlanmıştır. Kavşaklarda gecikmeyi azaltmak için gecikmenin nedenleri irdelenmiş ve her bir gecikme nedenini minimize etmeyi hedefleyen yenilikçi bir trafik kontrol sistemi literatüre sunulmuştur (Celtek ve ark., 2021).

Trafik sinyal ağının boyutunun artması ile kontrol ve koordinasyonda yaşanan zorluklar daha da artmaktadır. Bu tez ile trafik sinyal ağının verimli bir şekilde kontrol edilmesi için nesnelerin interneti tabanlı çok katmanlı bir optimizasyon mimarisi sunulmuştur.

7.1. Öneriler

Bu tez çalışmasında sezgisel optimizasyon yöntemleri sürü tabanlı stratejiler seçilmiştir. Yapılan çalışmalar diğer optimizasyon yöntemleri ile denenebilir. Elde edilen sonuçların gerçek zamanlı olmayan uygulamalar için değerli olacağı öngörülmektedir.

Bu çalışmada önerilen yöntemlerde bulut merkezi uzman olarak kullanılmaktadır. Faz süresini belirlemek için daha farklı makine öğrenimi yöntemleri kullanılabilir. Ayrıca bulut merkezi, alt katmanlarda yapılan optimizasyonun farklı parametreler kullanılarak bir iyileştirme gerçekleştirebilir. Kullanılan DQÖ algoritmasındaki öğrenme parametreleri değiştirilerek en uygun parametre değerleri incelenebilir. DQÖ performansını arttırmak adına başka bir makine öğrenme yöntemi ile hibrit bir sistem tasarlanabilir.

Tez kapsamında modelsiz TÖ yöntemi olan QÖ yöntemi kullanılmıştır. Uygun bir politika ile politika tabanlı yöntemler ya da uygun bir modelleme ile model tabanlı yöntemler ile uygulama gerçekleştirilebilir.

Önerilen nesnelerin interneti tabanlı mimari simülasyon ortamında gerçekleştirilmiştir. Trafiğin internet ortamına açılması çeşitli siber riskleri içermektedir. Bu riskleri önleme adına şifreleme gibi uygulamalar üzerine çalışmalar geliştirilebilir.

Bu çalışmada önerilen yöntemler SUMO simülatörü ile test edilmiştir. Simülasyon programlarının trafik ortamını modellemek için belirlediği bazı kabuller vardır. Performans testleri başka simülatörlerde denenip bu kabullerin etkisi

değerlendirilebilir. Önerilen yöntemler, gerçek trafik koşullarında test edilip, performansı simülasyon sonuçları ile karşılaştırılabilir.



KAYNAKLAR

- Abbas, N., Zhang, Y., Taherkordi, A. ve Skeie, T., 2017, Mobile edge computing: A survey, *IEEE Internet of Things Journal*, 5 (1), 450-465.
- Abdoos, M., Mozayani, N. ve Bazzan, A. L., 2011, Traffic light control in non-stationary environments based on multi agent Q-learning, *2011 14th International IEEE conference on intelligent transportation systems (ITSC)*, 1580-1585.
- Abdoos, M., 2020, Fuzzy Graph and Collective Multi-Agent Reinforcement Learning for Traffic Signals Control, *IEEE Intelligent Systems*.
- Akçelik, R. ve Roupail, N. M., 1993, Estimation of delays at traffic signals for variable demand conditions, *Transportation Research Part B: Methodological*, 27 (2), 109-131.
- Al-Qamash, A., Soliman, I., Abulibdeh, R. ve Saleh, M., 2018, Cloud, fog, and edge computing: A software engineering perspective, *2018 International Conference on Computer and Applications (ICCA)*, 276-284.
- Ali, M. E. M., Durdu, A., Çeltek, S. A. ve Gültekin, S. S., 2020, Fuzzy logic and webster's optimal cycle based decentralized coordinated adaptive traffic control method.
- Araghi, S., Khosravi, A. ve Creighton, D., 2015, A review on computational intelligence methods for controlling traffic signal timing, *Expert Systems with Applications*, 42 (3), 1538-1550.
- Aragon-Gómez, R. ve Clempner, J. B., 2020, Traffic-signal control reinforcement learning approach for continuous-time Markov games, *Engineering Applications of Artificial Intelligence*, 89, 103415.
- Arel, I., Liu, C., Urbanik, T. ve Kohls, A. G., 2010, Reinforcement learning-based multi-agent system for network traffic signal control, *IET Intelligent Transport Systems*, 4 (2), 128-135.
- Aslani, M., Seipel, S., Mesgari, M. S. ve Wiering, M., 2018, Traffic signal optimization through discrete and continuous reinforcement learning with robustness analysis in downtown Tehran, *Advanced Engineering Informatics*, 38, 639-655.
- Balaji, P. ve Srinivasan, D., 2010, Multi-agent system in urban traffic signal control, *IEEE Computational Intelligence Magazine*, 5 (4), 43-51.
- Balevi, E. ve Gitlin, R. D., 2018, Optimizing the number of fog nodes for cloud-fog-thing networks, *IEEE Access*, 6, 11173-11183.
- Ban, X. J., Hao, P. ve Sun, Z., 2011, Real time queue length estimation for signalized intersections using travel times from mobile sensors, *Transportation Research Part C: Emerging Technologies*, 19 (6), 1133-1156.
- Bonomi, F., Milito, R., Zhu, J. ve Addepalli, S., 2012, Fog computing and its role in the internet of things, *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, 13-16.
- Bull, A. ve Thomson, I., 2002, Urban traffic congestion: its economic and social causes and consequences, *Cepal review*.
- Casas, N., 2017, Deep deterministic policy gradient for urban traffic light control, *arXiv preprint arXiv:1703.09035*.

- Celtek, S. A., Durdu, A. ve Ali, M. E. M., 2020, Real-time Traffic Signal Control with Swarm Optimization Methods, *Measurement*.
- Celtek, S. A., Durdu, A. ve Ali, M. E. M., 2021, Evaluating Action Durations for Adaptive Traffic Signal Control Based On Deep Q-Learning, *International Journal of Intelligent Transportation Systems Research*.
- Cheng, R. ve Jin, Y. J. I. S., 2015, A social learning particle swarm optimization algorithm for scalable optimization, 291, 43-60.
- Chin, Y. K., Kow, W. Y., Khong, W. L., Tan, M. K. ve Teo, K. T. K., 2012, Q-learning traffic signal optimization within multiple intersections traffic network, 2012 *Sixth UKSim/AMSS European Symposium on Computer Modeling and Simulation*, 343-348.
- Chu, T., Wang, J., Codecà, L. ve Li, Z., 2019, Multi-agent deep reinforcement learning for large-scale traffic signal control, *IEEE Transactions on intelligent transportation systems*, 21 (3), 1086-1095.
- Clerc, M. ve Kennedy, J., 2002, The particle swarm-explosion, stability, and convergence in a multidimensional complex space, *IEEE Transactions on Evolutionary Computation*, 6 (1), 58-73.
- Clune, J. M., Ecoffet, A. L., Stanley, K. O., Huizinga, J. ve Lehman, J. A., 2020, Deep reinforcement learning based models for hard-exploration problems, Google Patents.
- CoordinatedData, 2021, Data For Coordinated Traffic, <https://github.com/salperenceltek/Evaluating-Action-Durations-for-ATSC-DQ>: [01.08.2021].
- Cui, C.-Y. ve Lee, H.-H., 2013, Distributed traffic signal control using PSO based on probability model for traffic jam, In: *Intelligent Autonomous Systems 12*, Eds: Springer, p. 629-639.
- Çakıcı, Z. ve Murat, Y. Ş., 2016, Sinyalize dönel kavşaklar için hesap yöntemi önerisi ve performans analizi, *Teknik Dergi*, 27 (4), 7569-7592.
- Çeltek, S. A., Soy, H. ve Hacıbeyoğlu, M., 2015, Nesnelerin İnternetine Doğru: Güncel Konular ve Gelecekteki Eğilimler, *Akademik Bilişim*, Eskişehir.
- Çeltek, S. A., Durgun, M., GöKrem, L. ve Durgun, Y., 2017, Nesnelerin İnterneti Tabanlı Yangın Alarm Sistemi Tasarımı ve Uygulaması, *Gaziosmanpaşa Bilimsel Araştırma Dergisi*, 6 (3), 66-72.
- Çeltek, S. A. ve Durdu, A., 2020, An Operant Conditioning Approach For Larga Scale Social Optimization Algorithms, *Konya Mühendislik Bilimleri Dergisi*, 8, 38-45.
- Çetinkaya, G., 2008, Işıklı Kavşaklarda Değişik Hesaplama Yöntemlerinin Karşılaştırılması.
- Dastjerdi, A. V. ve Buyya, R., 2016, Fog computing: Helping the Internet of Things realize its potential, *Computer*, 49 (8), 112-116.
- De Oliveira, M. B. ve de Almeida Neto, A., 2014, Optimization of traffic lights timing based on Artificial Neural Networks, *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 1921-1922.

- Dimitriou, L., Tsekeris, T. ve Stathopoulos, A., 2008, Adaptive hybrid fuzzy rule-based system approach for modeling and predicting urban traffic flow, *Transportation Research Part C: Emerging Technologies*, 16 (5), 554-573.
- Dion, F. ve Rakha, H., 2003, Estimating spatial travel times using automatic vehicle identification data, 12-16.
- Dion, F., Rakha, H. ve Kang, Y.-S., 2004, Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections, *Transportation Research Part B: Methodological*, 38 (2), 99-122.
- Durgun, M., Celtek, S. A., Gokrem, L. ve Durgun, Y., 2017, The design and implementation of web of low-power wireless temperature monitoring and alarm system, *2017 2nd International Conference on Advanced Information and Communication Technologies (AICT)*, 19-21.
- Eberhart, R. ve Kennedy, J., 1995, A new optimizer using particle swarm theory, *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 39-43.
- Eberhart, R. C. ve Shi, Y., 2000, Comparing inertia weights and constriction factors in particle swarm optimization, *Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512)*, 84-88.
- El-Tantawy, S., Abdulhai, B. ve Abdelgawad, H., 2013, Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown Toronto, *IEEE Transactions on intelligent transportation systems*, 14 (3), 1140-1150.
- Engelbrecht, A. P., 2014, Fitness function evaluations: A fair stopping condition?, *2014 IEEE Symposium on Swarm Intelligence*, 1-8.
- Fan, J., Wang, Z., Xie, Y. ve Yang, Z., 2020, A theoretical analysis of deep Q-learning, *Learning for Dynamics and Control*, 486-489.
- Garcia-Nieto, J., Olivera, A. C. ve Alba, E., 2013, Optimal cycle program of traffic lights with particle swarm optimization, *IEEE Transactions on Evolutionary Computation*, 17 (6), 823-839.
- García-Nieto, J., Alba, E. ve Olivera, A. C., 2012, Swarm intelligence for traffic light scheduling: Application to real urban areas, *Engineering Applications of Artificial Intelligence*, 25 (2), 274-283.
- Gartner, N. H., Pooran, F. J. ve Andrews, C. M., 2002, Optimized policies for adaptive control strategy in real-time traffic adaptive control systems: Implementation and field testing, *Transportation Research Record*, 1811 (1), 148-156.
- Ge, H., Song, Y., Wu, C., Ren, J. ve Tan, G., 2019, Cooperative deep Q-learning with Q-value transfer for multi-intersection signal control, *IEEE Access*, 7, 40797-40809.
- Genders, W. ve Razavi, S., 2018, Evaluating reinforcement learning state representations for adaptive traffic signal control, *Procedia computer science*, 130, 26-33.
- Genders, W. ve Razavi, S., 2019, Asynchronous n-step q-learning adaptive traffic signal control, *Journal of Intelligent Transportation Systems*, 23 (4), 319-331.
- Gokulan, B. P. ve Srinivasan, D., 2010, Distributed geometric fuzzy multiagent urban traffic signal control, *IEEE Transactions on intelligent transportation systems*, 11 (3), 714-727.

- Goldwasser, A. ve Thielscher, M., 2020, Deep Reinforcement Learning for General Game Playing, *AAAI*, 1701-1708.
- Gordon, R. L., Tighe, W. ve Siemens, I. T. S., 2005, Traffic control systems handbook, *United States. Federal Highway Administration. Office of Transportation*.
- Gubbi, J., Buyya, R., Marusic, S. ve Palaniswami, M., 2013, Internet of Things (IoT): A vision, architectural elements, and future directions, *Future generation computer systems*, 29 (7), 1645-1660.
- Hansen, N., Ros, R., Mauny, N., Schoenauer, M. ve Auger, A., 2008, PSO facing non-separable and ill-conditioned problems.
- Haydari, A. ve Yilmaz, Y., 2020, Deep Reinforcement Learning for Intelligent Transportation Systems: A Survey, *arXiv preprint arXiv:2005.00935*.
- HCM, 2000, Highway capacity manual.
- Hu, P., Dhelim, S., Ning, H. ve Qiu, T., 2017, Survey on fog computing: architecture, key technologies, applications and open issues, *Journal of Network and Computer Applications*, 98, 27-42.
- Huang, C., Lu, R. ve Choo, K.-K. R., 2017, Vehicular fog computing: architecture, use case, and security and forensic challenges, *IEEE Communications Magazine*, 55 (11), 105-111.
- Hunt, P., Robertson, D., Bretherton, R. ve Royle, M. C., 1982, The SCOOT on-line traffic signal optimisation technique, *Traffic engineering & control*, 23 (4).
- Hurwitz, J. S. ve Kirsch, D., 2020, Cloud computing for dummies, John Wiley & Sons, p.
- Jang, B., Kim, M., Harerimana, G. ve Kim, J. W., 2019, Q-learning algorithms: A comprehensive classification and applications, *IEEE Access*, 7, 133653-133667.
- Joo, H., Ahmed, S. H. ve Lim, Y., 2020, Traffic signal control for smart cities using reinforcement learning, *Computer Communications*.
- Jovanović, A., Nikolić, M. ve Teodorović, D., 2017, Area-wide urban traffic control: A Bee Colony Optimization approach, *Transportation Research Part C: Emerging Technologies*, 77, 329-350.
- Karaboğa, D., 2014, Yapay Zeka Optimizasyon Algoritmaları, Nobel Akademik Yayıncılık, p.
- Khalil, E. A. ve Özdemir, S., 2018, Nesnelerin internetine genel bir bakış: Kavram, özellikler, zorluklar ve fırsatlar, *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 24 (2), 311-326.
- KilisData, 2021, Data For Kilis Traffic, <https://github.com/salperenceltek/KilisData>: [01.08.2021].
- Kotusevski, G. ve Hawick, K., 2009, A review of traffic simulation software.
- Lee, J., Kao, H.-A. ve Yang, S., 2014, Service innovation and smart analytics for industry 4.0 and big data environment, *Procedia Cirp*, 16, 3-8.
- Liang, X., Du, X., Wang, G. ve Han, Z., 2019, A deep reinforcement learning network for traffic light cycle control, *IEEE transactions on vehicular technology*, 68 (2), 1243-1253.

- Liu, H. X., Wu, X., Ma, W. ve Hu, H., 2009, Real-time queue length estimation for congested signalized intersections, *Transportation Research Part C: Emerging Technologies*, 17 (4), 412-427.
- Liu, J., Li, J., Zhang, L., Dai, F., Zhang, Y., Meng, X. ve Shen, J., 2018, Secure intelligent traffic light control using fog computing, *Future generation computer systems*, 78, 817-824.
- Liu, M., Deng, J., Xu, M., Zhang, X. ve Wang, W., 2017, Cooperative deep reinforcement learning for traffic signal control, *The 7th International Workshop on Urban Computing (UrbComp 2018)*.
- Lowrie, P., 1990, SCATS—A Traffic Responsive Method of Controlling Urban Traffic. Roads and Traffic Authority, Sydney, *New South Wales, Australia*.
- Mach, P. ve Becvar, Z., 2017, Mobile edge computing: A survey on architecture and computation offloading, *IEEE communications surveys & tutorials*, 19 (3), 1628-1656.
- Mannion, P., Duggan, J. ve Howley, E., 2016, An experimental review of reinforcement learning algorithms for adaptive traffic signal control, In: *Autonomic road transport support systems*, Eds: Springer, p. 47-66.
- McCrea, J. ve Moutari, S., 2010, A hybrid macroscopic-based model for traffic flow in road networks, *European Journal of Operational Research*, 207 (2), 676-684.
- Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R. H., Morrow, M. J. ve Polakos, P. A., 2017, A comprehensive survey on fog computing: State-of-the-art and research challenges, *IEEE communications surveys & tutorials*, 20 (1), 416-464.
- Orcutt Jr, F. L., 1993, *The traffic signal book*, p.
- Papacostas, C. S. ve Prevedouros, P. D., 1993, *Transportation engineering and planning*, p.
- Pavlis, Y. ve Recker, W. W., 2004, Inconsistencies in the problem of optimal signal control for surface street networks, 361-366.
- Pell, A., Meingast, A. ve Schauer, O., 2017, Trends in real-time traffic simulation, *Transportation research procedia*, 25, 1477-1484.
- Rinne, H., 2008, *The Weibull distribution: a handbook*, CRC press, p.
- Roess, R. P., Prassas, E. S. ve McShane, W. R., 2004, *Traffic engineering*, Pearson/Prentice Hall, p.
- Rose, K., Eldridge, S. ve Chapin, L., 2015, The internet of things: An overview, *The Internet Society (ISOC)*, 1-50.
- Shaikh, P. W., El-Abd, M., Khanafer, M. ve Gao, K., 2020, A Review on Swarm Intelligence and Evolutionary Algorithms for Solving the Traffic Signal Control Problem, *IEEE Transactions on intelligent transportation systems*.
- Sharma, A., Bullock, D. M. ve Bonneson, J. A., 2007, Input–output and hybrid techniques for real-time prediction of delay and maximum queue length at signalized intersections, *Transportation Research Record*, 2035 (1), 69-80.
- Spall, J. C. ve Chin, D. C., 1997, Traffic-responsive signal timing for system-wide traffic control, *Transportation Research Part C: Emerging Technologies*, 5 (3-4), 153-163.

- Sun, C., Luo, Y. ve Li, J., 2018, Urban traffic infrastructure investment and air pollution: Evidence from the 83 cities in China, *Journal of Cleaner Production*, 172, 488-496.
- Sutton, R. S. ve Barto, A. G., 2018, Reinforcement learning: An introduction, MIT press, p.
- Tan, T., Bao, F., Deng, Y., Jin, A., Dai, Q. ve Wang, J., 2019, Cooperative deep reinforcement learning for large-scale traffic grid signal control, *IEEE transactions on cybernetics*, 50 (6), 2687-2700.
- Tang, C., Wei, X., Zhu, C., Chen, W. ve Rodrigues, J. J., 2018, Towards smart parking based on fog computing, *IEEE Access*, 6, 70172-70185.
- Tang, C., Xia, S., Zhu, C. ve Wei, X., 2019, Phase timing optimization for smart traffic control based on fog computing, *IEEE Access*, 7, 84217-84228.
- Van de Weg, G. S., Vu, H. L., Hegyi, A. ve Hoogendoorn, S. P., 2018, A Hierarchical Control Framework for Coordination of Intersection Signal Timings in All Traffic Regimes, *IEEE Transactions on intelligent transportation systems* (99), 1-13.
- Van der Pol, E. ve Oliehoek, F. A., 2016, Coordinated deep reinforcement learners for traffic light control, *Proceedings of Learning, Inference and Control of Multi-Agent Systems (at NIPS 2016)*.
- Vidali, A., Crociani, L., Vizzari, G. ve Bandini, S., 2019, A Deep Reinforcement Learning Approach to Adaptive Traffic Lights Management, *WOA*, 42-50.
- Wang, H., Sun, H., Li, C., Rahnamayan, S. ve Pan, J.-S., 2013, Diversity enhanced particle swarm optimization with neighborhood search, *Information Sciences*, 223, 119-135.
- Wang, X., Ke, L., Qiao, Z. ve Chai, X., 2020, Large-scale traffic signal control using a novel multiagent reinforcement learning, *IEEE transactions on cybernetics*.
- Webster, F., 1958a, Traffic signal settings, road research technical paper no. 39, *Road Research Laboratory*.
- Webster, F. V., 1958b, Traffic signal settings.
- Wei, H., Zheng, G., Yao, H. ve Li, Z., 2018, Intellilight: A reinforcement learning approach for intelligent traffic light control, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2496-2505.
- Wei, H., Zheng, G., Gayah, V. ve Li, Z., 2019, A survey on traffic signal control methods, *arXiv preprint arXiv:1904.08117*.
- Wiering, M. A., 2000, Multi-agent reinforcement learning for traffic light control, *Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000)*, 1151-1158.
- Xu, L.-H., Xia, X.-H. ve Luo, Q., 2013, The study of reinforcement learning for traffic self-adaptive control under multiagent markov game environment, *Mathematical Problems in Engineering*, 2013.
- Yang, J., 2020, Design Traffic Signal Node Based on Edge Computing, *Journal of Physics: Conference Series*, 012198.

- Yau, K.-L. A., Qadir, J., Khoo, H. L., Ling, M. H. ve Komisarczuk, P., 2017, A survey on reinforcement learning models and algorithms for traffic signal control, *ACM Computing Surveys (CSUR)*, 50 (3), 1-38.
- Ye, B.-L., Wu, W., Gao, H., Lu, Y., Cao, Q. ve Zhu, L., 2017, Stochastic model predictive control for urban traffic networks, *Applied Sciences*, 7 (6), 588.
- Zambrano-Bigiarini, M., Clerc, M. ve Rojas, R., 2013, Standard particle swarm optimisation 2011 at cec-2013: A baseline for future pso improvements, *2013 IEEE Congress on Evolutionary Computation*, 2337-2344.
- Zhang, J., Chen, B., Zhao, Y., Cheng, X. ve Hu, F., 2018, Data security and privacy-preserving in edge computing paradigm: Survey and open issues, *IEEE Access*, 6, 18209-18237.
- Zhang, Y. ve Ni, Q., 2017, A coordinated traffic control on urban expressways with modified particle swarm optimization, *KSCE Journal of Civil Engineering*, 21 (2), 501-511.
- Zhang, Y. ve Zhou, Y., 2018, Distributed coordination control of traffic network flow using adaptive genetic algorithm based on cloud computing, *Journal of Network and Computer Applications*, 119, 110-120.
- Zhao, D., Dai, Y. ve Zhang, Z., 2011, Computational intelligence in urban traffic signal control: A survey, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42 (4), 485-494.
- Zhao, D., Dai, Y. ve Zhang, Z., 2012, Computational intelligence in urban traffic signal control: A survey, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42 (4), 485-494.
- Zheng, X. ve Recker, W., 2013, An adaptive control algorithm for traffic-actuated signals, *Transportation Research Part C: Emerging Technologies*, 30, 93-115.
- Zhou, P., Fang, Z., Dong, H., Liu, J. ve Pan, S., 2017, Data analysis with multi-objective optimization algorithm: A study in smart traffic signal system, *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, 307-310.