



Optimization of traffic signals on roundabouts via reinforcement learning

Onur Ağbulut*^{ID}, Korhan Karabulut^{ID}

Department of Computer Engineering, Faculty of Engineering, Yaşar University, 35100, İzmir, Türkiye

Highlights:

- Traffic light control via reinforcement learning
- Optimization of selected traffic metrics on roundabouts using reinforcement learning
- Optimization of selected traffic metrics on high traffic volume

Keywords:

- Reinforcement Learning
- Traffic Lights
- Traffic Light Optimization
- Roundabouts

Article Info:

Research Article
Received: 19.08.2024
Accepted: 12.02.2026

DOI:

10.17341/gazimmfd.1535848

Correspondence:

Author: Onur Ağbulut
e-mail: agbulutonur@gmail.com
phone: +90 546 813 9866

Graphical/Tabular Abstract

In this study, a reinforcement learning-based solution to optimize traffic flow in roundabouts is developed. Several experiments for different traffic flow scenarios are executed. Results with mean values of metrics and standard deviations (separated by /) are shown in Table A.

Table A. Experiment Results for different scenarios

Case	Type	Mean Stop Veh.	Mean Wait Time	Mean Speed	Mean Speed 2	Mean Speed 3
1	Test	7.461 / 0.799	241.917 / 168.501	0.362 / 0.016	6.709 / 0.305	7.916 / 0.217
	Baseline	144.040 / 0.084	3240.663 / 1.902	0.038 / 0.000	0.705 / 0.005	1.261 / 0.005
2	Test	5.323 / 1.311	143.903 / 131.243	0.374 / 0.030	6.930 / 0.561	8.076 / 0.416
	Baseline	5.268 / 0.218	76.368 / 4.869	0.367 / 0.006	6.828 / 0.118	8.007 / 0.080
3	Test	5.777 / 1.084	216.478 / 174.338	0.360 / 0.024	6.690 / 0.442	7.896 / 0.327
	Baseline	53.582 / 0.714	1223.557 / 16.437	0.074 / 0.001	1.377 / 0.012	2.378 / 0.020
4	Test	2.115 / 0.964	131.337 / 133.111	0.355 / 0.056	6.515 / 1.022	8.028 / 0.966
	Baseline	1.192 / 0.003	12.336 / 0.022	0.406 / 0.001	7.457 / 0.004	8.956 / 0.006
5	Test	28.331 / 7.559	579.920 / 174.651	0.200 / 0.039	3.708 / 0.722	5.290 / 0.743
	Baseline	201.625 / 0.036	4587.538 / 0.836	0.027 / 0.000	0.496 / 0.001	0.892 / 0.001
6	Test	6.546 / 0.582	119.658 / 55.171	0.380 / 0.013	7.053 / 0.248	8.159 / 0.174
	Baseline	143.259 / 0.271	3225.640 / 6.176	0.038 / 0.000	0.704 / 0.000	1.261 / 0.001
7	Test	7.391 / 0.779	125.316 / 42.310	0.360 / 0.016	6.690 / 0.293	7.906 / 0.208
	Baseline	127.056 / 1.657	2907.518 / 38.239	0.041 / 0.001	0.758 / 0.009	1.356 / 0.016
8	Test	3.902 / 1.527	191.949 / 194.151	0.368 / 0.048	6.822 / 0.887	8.022 / 0.717
	Baseline	2.659 / 0.173	33.126 / 3.766	0.411 / 0.010	7.589 / 0.183	8.678 / 0.116

Purpose: This study aims to propose a new approach by developing a reinforcement learning-based solution for traffic congestion problem to optimize traffic flow in roundabouts by controlling traffic lights.

Theory and Methods: A solution based on Q-Learning with custom reward and state mechanism is designed to optimize various metrics. Experiments for different traffic flow scenarios are designed and executed multiple times to have meaningful results. To compare the performance of our method, sequentially changing traffic light experiments are executed. Outputs for different traffic metrics are calculated.

Results: As a result, in scenarios with high-traffic density and asymmetric volume of vehicles, our study performs better and has lower values than baseline in mean stopped vehicles and mean waiting time metrics. Moreover, our study has better results than baseline in all mean speed metrics and resulted in higher average speeds. In scenarios with lower-traffic density our study performs slightly worse but has virtually identical values compared to baseline in mean stopped vehicles metric. In all mean speed metrics, our solution has extremely close values to baseline values.

Conclusion: Our solution produced better results in 5 out of 8 scenarios in 5 different metrics than baseline. In rest of 3 scenarios, in 4 different metrics, our solution has similar values with baseline. Only in 1 metric, it has worse values than baseline. In conclusion, in terms of vehicle density, our solution is remarkably successful, in high-volume traffic cases. In low-volume traffic, it has similar values to baseline results.



Pekiştirmeli öğrenme ile dönel kavşaklarda trafik ışıklarının optimizasyonu

Onur Ağbulut*^{ID}, Korhan Karabulut^{ID}

Yaşar Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 35100, Bornova, İzmir, Türkiye

Ö N E Ç İ K A N L A R

- Pekiştirmeli öğrenme ile trafik ışık kontrolü
- Dönel kavşaklarda pekiştirmeli öğrenme kullanarak belirlenen trafik ölçütlerinin eniyilemesi
- Çok fazla araç hacmi durumlarında belirlenen trafik ölçütlerinin eniyilemesi

Makale Bilgileri

Araştırma Makalesi

Geliş: 19.08.2024

Kabul: 12.02.2026

DOI:

10.17341/gazimmfd.1535848

Anahtar Kelimeler:

Pekiştirmeli öğrenme,
trafik ışıkları,
dönel kavşaklar,
trafik ışık optimizasyonu

ÖZ

Bu çalışmada, trafik ışıklarını kontrol ederek dönel kavşaklardaki trafik akışını optimize etmek için pekiştirmeli öğrenmeyi temel alan bir çözüm geliştirilmiştir. Q-Öğrenme algoritmasını kullanan bir çözüm geliştirilmiş ve farklı trafik ölçütlerini eniyileme amacı ile ödül ve durum mekanizmaları tanımlanmıştır. Dönel kavşaklarda ortalama bekleme süresi, duran araç sayısı, araçların ortalama hızı vb. gibi ölçütler dikkate alınarak trafik akışının eniyilenmesi hedeflenmiştir. Bu amaçla, farklı trafik akış senaryoları için çok sayıda deney oluşturulmuş ve istatistiksel olarak anlamlı sonuçlar elde edebilmek için her bir senaryo birden fazla kez bağımsız olarak simüle edilmiştir. Önerilen yöntemin başarımının ölçülebilmesi için, trafik ışıklarının sırayla değiştiği deneyler de tasarlanmış ve çalıştırılmıştır. Bu deneylerde belirlenen farklı trafik ölçütleri kullanılarak ölçümler yapılmıştır. Elde edilen sonuçlar trafik ışıklarının sıra ile değiştiği temel durum değerleriyle karşılaştırılmıştır. Sonuç olarak, Q-Öğrenme algoritmasını kullanan pekiştirmeli öğrenme bazlı çözüm, temel durum deneylerinin sonuçları ile kıyaslandığı zaman, özellikle çok fazla araç hacmi olduğu durumda ortalama bekleme süresi, duran araç sayısı, araçların ortalama hızı gibi birçok ölçüt cinsinden daha iyi sonuçlar verdiği gözlenmiştir.

Optimization of traffic signals on roundabouts via reinforcement learning

H I G H L I G H T S

- Traffic light control via reinforcement learning
- Optimization of selected traffic metrics on roundabouts using reinforcement learning
- Optimization of selected traffic metrics on high traffic volume

Article Info

Research Article

Received: 19.08.2024

Accepted: 12.02.2026

DOI:

10.17341/gazimmfd.1535848

Keywords:

Reinforcement learning,
traffic lights,
roundabouts,
traffic light optimization

ABSTRACT

In this study, a reinforcement learning-based solution for congestion problems in traffic flow was developed to optimize traffic flow in roundabouts by controlling traffic lights. A solution based on the Q-Learning Algorithm is developed, and reward and state mechanisms are implemented to optimize various traffic metrics. This study aims to optimize traffic flow in roundabouts in terms of mean waiting time, the number of stopped vehicles, average speed of cars, etc. It aims to optimize them by controlling the change of traffic lights. Several experiments have been designed for different traffic flow scenarios. They are simulated independently multiple times to obtain statistically significant results. To compare the performance of the proposed method, experiments in which traffic lights change sequentially are also designed and executed. In these experiments, outputs for different traffic metrics are measured. Later, the results of the proposed method are compared with these baseline results. As a result, the proposed method using the Q-Learning algorithm-based reinforcement learning has been observed to produce better results than baseline results in multiple metrics like mean waiting time, number of stopped vehicles, and average of cars, especially when there is a high traffic volume.

1. Giriş (Introduction)

Donanım limitleri nedeniyle önceki dönemlerde aktif olarak çalışılmamış bilgisayar bilimleri konuları, yakın zamanda gerçekleşen donanım teknolojisindeki gelişmeler ile birlikte şimdi aktif ve popüler olmaya başlamışlardır. Bilgisayar biliminin öne çıkan alanlarında biri olan Yapay Zekâ (YZ) son yıllarda en popüler konulardan biri haline gelmiştir (Hollander [1]). Gelişen donanım teknolojisi YZ etmenlerini daha hızlı bir şekilde eğitmeyi ve test etmeyi mümkün hale getirdi ve böylece, daha önce başarması oldukça zor olan etmen eğitimini daha kısa sürede tamamlamaya olanak sağladı. Bununla birlikte, YZ ve makine öğrenmesi gibi YZ alt alanlarını daha akıllı hale getiren yeni çözümler ve algoritmalar geliştirildi. Bu gelişmeler yapay zekânın günlük hayatla daha fazla iç içe olmasına olanak sağladı. Örneğin, temelinde yapay zekâyı kullanan bilgisayar görüşünün ilerlemesi ile yüz tanıma sistemleri ve sürücüsüz araçlar kullanılabilir hale geldi (Srivastava [2]). Trafik sistemleri de teknolojiye bu ilerlemeden yararlanabilecek alanlardan birisidir. Teknoloji geliştikçe, trafik unsurları da daha akıllı olmaya başladılar ve çevrelerindeki ortam hakkında bilgi toplama ve işleme yeteneğine sahip oldular. Örneğin, bütünlük bir kamera aracılığıyla trafik işaretlerini veya sürücünün şeritten çıkıp çıkmadığını tespit edebilen ve onları uyararak araçlar mevcuttur (Pilgrim [3], Howard [4]). Ayrıca, trafiğin başka bir parçası olan trafik ışıklarını daha akıllı yapmak için kuruldukları yollar/kavşaklar hakkında bilgi toplamaya başlayan sensörler de mevcuttur. Bu bilgiler, geçen araç sayısı, araçların hızları vb. bilgilerdir (Wheatley [5]). Bu teknik ilerlemeler ve yetenekler ile birlikte, toplanan verilerin yapay zekâdaki ilerleme ile birleştirilip trafik sıkışıklığı gibi önemli trafik problemlerini çözmek için kullanılabileceğini düşünmek mümkündür.

Şehirler ve kırsal alanların kalabalıklaşmasıyla birlikte yollardaki araç sayısındaki artış trafik sıkışıklığını önemli bir problem haline gelmiştir. Araçların sayısındaki bu artış, yetersiz altyapı ve her gün trafiğe çıkan araç sayısını karşılamakta kapasite olarak yetersiz olan yollar, hava şartları ve yollarda meydana gelen kazalar ile birleşince trafik sıkışıklığını artırmaktadır. Trafik ışıkları bu sistemin en önemli parçalarından birisidir. Genellikle iki tür trafik ışığı vardır; her bir faz için sabit zamanlı klasik trafik ışıkları ve trafiğe göre uyarlanabilen trafik ışıkları. Sıkışıklığı azaltmaya olan etkileri dikkate alındığında, ilki o anki trafik akışına göre değişken olmadığı için, diğeri de daha adaptif olmasına karşın, problemi daha kapsamlı ve geniş zaman menzili biçiminde dikkate alamadığı ve bu iki özelliği aynı anda destekleyemedikleri için, her iki tür trafik ışığı da çok verimli olarak değerlendirilmemektedir (Yau vd. [6]).

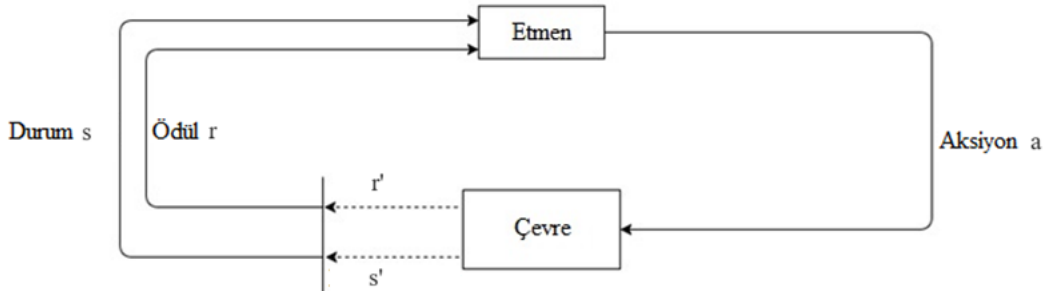
Trafik elemanlarının çevreleri ve araçlar hakkında veri toplayabilmeleri sayesinde yapay zekâ teknolojisi trafik sıkışıklığını azaltmak ve trafik akışını birçok yönden geliştirmek için (örneğin araçların toplam seyahat süresinin azaltılması, trafik ışıklarının neden

olduğu toplam gecikmenin azaltılması vb.) kullanılabilir. Trafik akışının kontrolünde büyük bir rol oynayan trafik ışıkları teknolojik açıdan zaman içinde değişiklikler göstermiştir. Genel olarak, bir trafik ışığında 3 farklı faz vardır, bunlar kırmızı, sarı ve yeşildir. Geleneksel bir trafik ışığı bu fazların her biri için sabit zamana sahiptir. Gelişen teknoloji ile trafik ışıkları, trafik yoğunluğuna göre harekete geçmek için kendilerini ayarlayabilecekleri biçimde daha adaptif olmuşlardır. Daha da ötesi, şu an daha “akıllı” olmuşlardır ve yollara yerleştirilen sensörlerin ve video kameraların yardımıyla hız ölçmek ve ücret toplamak gibi işleri yapabilmektedir. Ek olarak, trafik sıkışıklığı, araçların ya da motosikletlerin sayısı, varsa bir kazanın raporlanması gibi yoldaki trafik hakkında farklı bilgileri de toplayabilmektedir (Bullock [7]). Sonuç olarak, bu bilgiler, trafik ışıklarının her bir faz için sürelerini uyarlaması ve trafik yoğunluğuna göre bu fazlar arasında değişiklik yapması için makine öğrenmesi algoritmaları ile birlikte kullanılabilir. Bu şekilde, trafik sıkışıklığını azaltabilir ya da ortadan kaldıracaklardır. Bu durum aynı zamanda tüketilen yakıtın azalmasına ve daha az motor yıpranmasına sebep olacağı için, maliyetlerin azaltılmasını sağlayacak ve ekonomik katkı da sağlayacaktır. Daha önemlisi, azaltılmış CO₂ emisyon oranlarından dolayı çevreye daha az zarar verilmesine sebep olacaktır (El-Tantawy vd. [8]).

Pekiştirmeli Öğrenme (PÖ), deneme-yanılma bazlı bir öğrenme yaklaşımıdır. Etmen, ortam, durum uzayı ve aksiyon uzayı olarak isimlendirilen elemanlara sahiptir. Çok özlü bir şekilde, bir problem ortamı vardır ve etmen bu ortamda deneyimleyerek öğrenmektedir. Etmen aksiyonlar alır ve buna göre durumunu değiştirir. Bunların sonucu olarak, seçilen aksiyona göre ödül veya ceza alır. Aldığı aksiyonlar ile ödüllerin toplamını maksimize etmeyi hedefler. Pekiştirmeli öğrenmenin mimarisi özet olarak Şekil 1’de gösterilmiştir.

Trafik sistemindeki trafik ışığı o anki aktif faz ve kavşakların farklı kollarında bekleyen araçların göreceli veya toplam bekleme süresinin matematiksel değerlerinin birleşmesiyle oluşturulan durumlara sahiptir. Trafik ışığı, aynı zamanda o anki sistem durumunun da değiştirmesine sebep olacak şekilde fazını değiştirebilir. Böylece, basit bir şekilde, fazları arasında ne zaman değişiklik yapacağına karar vererek durum uzayını kontrol edebilir. Bu nedenle bir trafik ışığı bir pekiştirmeli öğrenme etmeni olarak deneyimleyerek öğrenebilir.

Diğer pekiştirmeli öğrenme etmenlerine benzer şekilde, trafik ışığı da değişen trafik akışına göre fazlarının süresini kontrol ederek ve ödülleri toplamını maksimize etmeyi hedefleyerek kendini uyarlayabilir. Bir etmen için en önemli amaç olan ve öğrenmesini önemli bir şekilde etkileyen ödül, araçların toplam bekleme süreleri, seyahat süresi, toplam gecikme vb. gibi farklı niteliklerden sadece biri veya onlardan birkaçının birleşimi dikkate alınarak hesaplanabilir.



Şekil 1. Pekiştirmeli öğrenmenin temel mimarisi (Fundamental architecture of Reinforcement Learning)

Etmen öğrendikçe, araçların azaltılmış bekleme süresi veya azaltılmış gecikmesine neden olduğu durumlar olacak ve böylece daha çok ödüle sahip olarak daha verimli durumlara yol açacak aksiyonlara yönelecektir (Gollapudi [9]).

Bu pekiştirmeli öğrenme modeli hem kavşaklar hem de dönel kavşaklarda trafiğin düzenlenmesi için kullanılabilir. Dönel kavşaklar normal kavşaklara alternatif olarak kullanılan trafik elemanlarıdır. Kavşaklardan farklı olarak, dönel kavşaklar genelde trafik ışıklarına ihtiyaç duymadan trafik akışını düzenlemek için tasarlanmıştır. Ancak genellikle düşük hacimli trafik akışını kontrol etmekte başarılıdır (Rizzo vd. [10]). Kavşaklarda, giren ve çıkan şeritler vardır, akış genellikle trafik ışıkları tarafından yönetilir. Dönel kavşaklarda kavşaklardan farklı biçimde, giren ve çıkan şeritlere ek olarak ayrıca, giren ve çıkan şeritler arasındaki bağlantı şeritleri olan dolaşım şeritleri vardır. Buna ek olarak, kavşaklara benzer bir şekilde dönel kavşaklar da trafik ışıklarına sahip olabilirler. Dönel kavşaklar, trafik ışıklarına ihtiyaç olmadan trafiği kontrol etmek için tasarlanmış olmasına rağmen, trafik ışıklarının kullanıldığı dönel kavşaklar da vardır. Amerika Birleşik Devletleri (ABD)'de, trafik ışıklarına sahip olan dönel kavşaklar Federal Karayolu İdaresi (FHWA) tarafından teşvik edilmez. ABD'den farklı olarak, Avrupa'da ise yaygın olarak kullanılmaktadırlar. Normal (trafik ışiksiz) dönel kavşak ve trafik ışıklarına sahip olan dönel kavşak modellerinin her ikisi de Birleşik Krallık, Hollanda, Fransa, Türkiye gibi ülkelerde mevcuttur. Özellikle Türkiye'de, dönel kavşaklar trafik akışı düzenlemesinde önemli bir rol oynamaktadır (Gökçe vd. [11]).

Kavşaklara benzer şekilde, dönel kavşaklar da kol adı verilen bağlantı yollarına sahiptir. Bunlar birbirlerine dönel kavşaklar ile bağlanabilir. Dönel kavşaklarda, dolaşım şeritlerinin içindeki araçların, kollardaki araçlara göre önceliği vardır. Bu nedenle, kural olarak, kollardaki araçlar, dönel kavşaklardaki dolaşım şeritlerinin içindeki araçların dolaşım şeritlerinden çıkmalarını beklemelidirler. Bir dönel kavşağın elemanları Şekil 2'de gösterilmiştir.

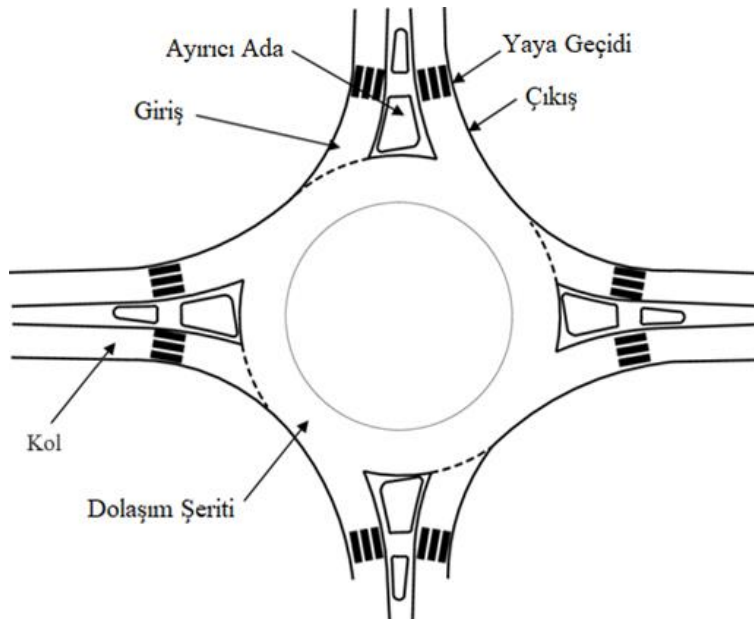
Alanyazında dönel kavşaklar, kavşaklar ve dönel kavşaklar arasında karşılaştırmalar, üzerinde trafik ışıkları olan dönel kavşaklar ve dönel kavşaklarda sürücüsüz araçları optimize etmeye odaklanan çalışmalar vardır. Dönel kavşaklar dünyanın her yerinde, Avrupa'da olduğu

kadar yaygın olmadığı için ve ayrıca Avrupa'da da yalnızca bazı ülkelerde kullanıldığı için, dönel kavşaklarla ilgili olarak alan yazında kavşaklarla ilgili makalelere göre çok daha az sayıda çalışma bulunmaktadır. Buna ek olarak, bu çalışmada göz önüne alınan dönel kavşaklarda trafik ışığı kontrol mekanizmalarında pekiştirmeli öğrenmenin kullanımı ile ilgili çalışmalar oldukça sınırlıdır. Yapılan alanyazın taramasında var olan çalışmalarda, kullandıkları algoritmalar, optimize etmeye çalıştıkları noktalar, eksik/tamamlanmamış iş ve çalışmaları için genişletmeyi düşündükleri gelecekteki noktalar gibi bazı kilit noktalar tespit edilmeye çalışılmıştır.

Erdem [12] kavşaklarda trafik sinyalizasyonu üzerine çalışmıştır. Bulanık mantık kullanarak, trafik ışıklarına yaklaşan araçları dedektörler ile sayıp sinyalleri buna göre değiştirmiştir. Özellikle değişken trafik yoğunluğunun olduğu kavşaklarda, bulanık mantık kullanılarak bekleme sürelerinin klasik yöntemle göre azaltılabileceğini göstermiştir. Buna ek olarak yakıt tüketimi ve hava kirliliği gibi değerlerin de azalmasına katkıda bulunduğunu göstermiştir.

Chin vd. [13], Q-Öğrenme kullanarak birden fazla kavşakta trafik akışını optimize etmeye çalışmıştır. Durum temsili olarak her kavşakta bekleyen araç sayısının yoğunluk seviyesini ve aksiyon kümesi olarak yeşil ışık süresini kullanmıştır. Ödül fonksiyonu olarak ise araç yoğunluğunu azaltacak bir fonksiyon tasarlamıştır. Sonuç olarak bakıldığında birden fazla kavşakta trafik ışık dağılımını daha iyi hale getirerek, akışı optimize etmiştir. Gelecek çalışma planı olarak ise, iki ardışık trafik fazı arasındaki sürenin geliştirilmesi üzerinde çalışmayı hedeflemiştir.

Chin vd. [14], Q-Öğrenme kullanarak 4 kollu bir kavşakta trafik sinyal iyileştirmesi yapmıştır. Her bir fazdaki bekleyen araç sayısını durum temsili olarak kullanmıştır. Aksiyon kümesi olarak 1 saniye ve 5 saniye arasında değişen yeşil ışık sürelerini kullanmıştır. Ödül fonksiyonu olarak ise bekleyen araç sayısına göre bir fonksiyon tasarlamıştır. Veri seti olarak 4 kollu bir kavşağı simülasyon ortamı olarak seçip buradan toplanan verileri kullanmıştır. Sonuç olarak önerilen algoritmanın trafik sinyal zamanlamasını başarılı bir şekilde yönettiğini göstermiştir.



Şekil 2. Dönel kavşak ve elemanları (Roundabout and its elements)

Zeng vd. [15], Tekrarlayan Sinir Ağı (Recurrent Neural Network – RNN) ve Derin Q Ağını (Deep Q-Network – DQN) birleştirip Derin Tekrarlayan Q Ağı (Deep Recurrent Q-Network) kullanarak tek bir kavşak üzerinde trafik sinyal zamanlamasını optimize etmeye çalışmıştır. GPS’den alınan gerçek zamanlı veriyi kullanmıştır. Sonuç olarak, DRQN yönteminin, sabit zamanlı sinyal metoduna oranla trafiği optimize ettiğini ve araçlardaki gecikmelerin süresini azalttığını göstermiştir. Ayrıca, DRQN metodunun standart Q-Öğrenmesine göre daha iyi simülasyon sonuçları aldığını da gözlemlemiştir. Gelecek çalışma olarak ise, bu probleme hafıza ağı yaklaşımını uygulamayı planlamaktadır.

Ducrocq ve Fahri [16], derin pekiştirmeli Q-Öğrenmesi (Deep Reinforcement Q-Learning – DRQL) için yeni bir model geliştirerek izole bir kavşakta trafik sinyalizasyonunu optimize etmeye çalışmıştır. Bunun için kısmi gözlemlenebilir bir ortamda bağlı araçlar (connected vehicles) ile çalışmıştır. Bu yöntemi kullanarak başarılı sonuçlar almıştır ve gelecek çalışma olarak birden fazla trafik ışığının iletişimi için çok etmenli pekiştirmeli öğrenme sistemlerle çalışma, farklı kavşak tipleri üzerinde çalışma gibi farklı konuları planlamıştır.

Wang vd. [17], Derin Q-Öğrenme (Deep Q-Learning – DQL) kullanarak trafik sinyal zamanlama problemini çözmeye çalışmıştır. Deneylerde Çin’de Suzhou Endüstriyel Parkında bulunan sensörler vasıtasıyla toplanan veriden faydalanmıştır. Sonuç olarak kentsel bölgede bulunan kavşaklarda trafik sinyal zamanlamasını optimize etmiştir. Gelecek çalışma olarak ise, bu yöntemi genişletip Model Öngörülü Kontrol (Model Predictive Control – MPC) ile karşılaştırmayı ve mevcut yöntemi ölçeklendirip daha büyük senaryolarda kullanmayı düşünmektedir.

Ahmed vd. [18], Derin Q-Öğrenme kullanarak kavşaklardaki yoğunluğu azaltmayı hedeflemiştir. Yoğunluk ve hız ikilisini kullanarak bir politika geliştirmiştir. Sonuç olarak önerilen yöntemin başarılı olduğunu göstermiş ve gelecek planı olarak başka trafik metriklerinin de kullanılarak yeni bir politika geliştirmeyi düşünmektedir.

Meneguzzar vd. [19], var olan bir kavşağı bir dönel kavşak ile değiştirerek bir karşılaştırma yapmıştır. İkisini de araç emisyonları cinsinden karşılaştırıp değerlendirmişlerdir. Arabalara yükledikleri Portatif Emisyon Ölçüm Sistemi’ni (PEMS) kullanmışlar ve her iki çözümün performansını CO₂, NO_x ve CO gazları salınımı cinsinden ölçmüşlerdir. Veri kümesi olarak ise, tüm sinyalleri ve ilgili veriyi toplayan bir test aracı kullanmışlardır. Sonuç olarak, trafik ışıklarına sahip olan kavşağı, üzerinde hiçbir trafik ışığı olmayan bir dönel kavşak ile değiştirerek, NO_x emisyonları kavşaklarda daha iyi iken ve CO emisyonları her ikisinde daha iyi değilken, dönel kavşaklarda CO₂ emisyonlarını azaltmayı başarmışlardır. Gelecek çalışma olarak, sürüş tarzı, yaya akışı gibi bazı diğer faktörleri de dahil etmeyi amaçlamışlardır.

Kubíková vd. [20], Slovakya’nın Hlohovec şehrindeki trafik akışını optimize etmeye odaklanmışlardır. Dönel kavşaklar, trafik ışıkları olan kavşaklar ve turbo dönel kavşaklarda deneyler yapmışlardır. Küçük bir dönel kavşağı, trafik ışıkları olan bir kavşak ve bir turbo dönel kavşak ile değiştirmeyi denemişlerdir. Trafik akışının arttığını gözlemlemişler, bu sebeple, bu durum azaltılması gereken bir trafik sıkışıklığına sebep olmuştur. Ancak, bazı durumlar için, iki şeritli / küçük geçirgenliği ve kapasitesi olan küçük dönel kavşakların avantajlı olduklarını tespit etmişlerdir. Sabah 6’dan, akşam 6’ya kadar 12 saatlik veri toplamış ve her türlü aracı tespit etmeye çalışmışlardır. Ayrıca, sabah 15 dakika ve 1 saat aralıkları ile ek veriler de toplamışlardır. Tüm bu veriyi toplamak için dönel kavşağın dışına yüksekte yerleştirilmiş bir kamera kullanmışlardır. Sonuç olarak, küçük dönel kavşaklar yerine turbo dönel kavşaklar kullanarak daha

az durma süresi ve daha iyi trafik akışı elde etmeyi başarmışlardır. Dahası, hız ve seyahat süreleri gibi diğer trafik parametreleri cinsinden de özellikle turbo dönel kavşaklarda diğer alternatiflere göre daha iyi sonuç almışlardır. Emisyon gibi çevre parametreleri ile ilgili olarak, turbo dönel kavşak ve trafik ışıkları olan kavşaklar daha iyi sonuçlara sahiptir. Gelecek çalışmalar için, Hlohovec Şehri’nde mevcut dönel kavşakları, trafik ışıkları olan kavşaklarla veya turbo dönel kavşaklarla değiştirmeleri için tavsiyede bulunmuşlardır.

Gökçe vd. [11], dönel kavşaklarda bekleme sürelerini azaltmayı amaçlamışlardır. İzmir, Türkiye’deki en büyük dönel kavşaklardan birinde simülasyon-optimizasyon modeli için Parçacık Sürü Optimizasyonu (PSO) kullanmışlardır. Simulator olarak, VISSIM’i, bir aylık sürede her gün 30 dakika boyunca özellikle yoğun saatlerde topladıkları gerçek veriyle birlikte kullanmışlardır. Gecikme süresini yaklaşık %55,9 azaltan ve geçen araç sayısını yaklaşık %9,3 artıran bir optimizasyona ulaşmışlardır. Gelecek çalışma için, bu çalışmayı, birden fazla dönel kavşak ekleyerek, böylece bir dönel kavşak grubu oluşturarak ve trafik karakteristikleri ile ilgili daha fazla veri yakalayarak genişletmeyi hedeflemektedirler.

Shaaban vd. [21], Doha, Katar’daki en yoğun alanlardan birindeki trafik akışına odaklanmışlardır. İki farklı çözümü karşılaştırmışlardır. Halihazırda trafik ışıklarının olduğu dönel kavşakları trafik ışıkları olan kavşaklar ile simüle etmeye çalışmışlardır. Ulaştırma Bakanlığı’ndan dönel kavşaklar için özellikle en yoğun saatler için veri toplamışlardır. Ayrıca, bakım ajansının yardımı ile kavşaklar ile değiştirdikten sonra, sinyal faz zamanlaması için veri toplamışlardır. Sonuç olarak, emisyonu %37-43 arasında azaltmayı başarmışlardır. Kullandıkları US EPA MOVE, Katar koşullarına uyabilmek için ayarlanmıştır. Gelecek çalışma için, aynı bölgedeki diğer ülkeler için bu yaklaşımın genişletilebileceğini belirtmişlerdir.

Rizzo vd. [10], trafik ışığı kontrolü problemini, özellikle yoğun bir şekilde sıkışmış trafik senaryoları için sıkışıklığı azaltırken, akışı maksimize ederek, optimize etmeye çalışmışlardır. Bunu başarmak için Zaman Kritiği metodunu temel alan Derin Politika Gradyanları kullanmışlardır. Katar’daki Doha şehrinde bulunan Al Gharrafa dönel kavşağında yoğun saatlerde veri toplamışlar ve bunu daha sonra simülatörde tekrar üretmişlerdir. Ödül hesaplaması olarak araç sayısını kullanmışlardır. Gelecek çalışma için bunu gerçek hayatta kullanmak istediklerini belirtmişlerdir.

Rizzo vd. [22] kavşaklar için gerçek veri kullanarak yapay sinir ağı ile eğitilmiş olan bir etmeden, bir simüle edilmiş yol ağının girdi durumu ve pekiştirmeli öğrenme etmeninin çıktısı ve etmenin faz tercihleri ve gerçek trafik arasında ilişki bulmaya ve etmenin, o anki detektörlerin durumuna tepkisinin başarımını ölçmeye çalışmışlardır. Bir sinir ağı etmenini, Shapley Additive exPlanation (SHAP) model-bağımsız teknikle, trafik ışıkları olan bir dönel kavşak üzerine Politika Gradyanı (PG) ile eğitmişlerdir. Veri kümesi olarak, trafik ağı için hafta içi bir gün yoğun saatler için veri toplamışlardır. Durum tasarımı olarak tespit döngüleri boyunca gelen araç sayısını ve aksiyon olarak trafik ışığı fazlarını kontrol eden bir aksiyon kümesi kullanmışlardır. Ödül olarak belirli bir zamanda dönel kavşağı terk eden araç sayısını kullanmışlardır. Buna ek olarak, tüm ödüllerin durumu olan nihai durumlar tanımlamışlardır. Sonuç olarak, deney çıktıları SHAP model bağımsız tekniğini kullanarak karmaşık bir politika üzerine karar almış bir etmeden bilgi çıkarmanın mümkün olduğunu göstermişlerdir. Yöntemin eksikliği olarak, görülen trafik için bir eğilim olduğu ve bu nedenle bir aşırı uyum olduğu görülmektedir. Ayrıca, yazarlar, dedektör değerleri ve faz seçimleri arasındaki ilişkiyi açıklayamamışlardır. Gelecek çalışma olarak ise, daha basit modelleri doğrudan eğitmeyi ve karmaşık Derin Öğrenme (DÖ) modeline kıyasla doğruluğundan ödün vermesini gözlemlemeyi amaçlamaktadırlar.

Paul ve Mitra [23], pekiştirmeli öğrenme ve derin öğrenme kullanarak, trafik akışı için bekleme süresi, CO₂ emisyonu ve yakıt tüketimi gibi birçok ölçütü optimize etmeye çalışmışlardır. Politika gradyan (PG) algoritması ile eğitilmiş olan etmenleri kullanmışlar ve daha sonra üç derin sinir ağı (DNN) modeli ile değerlendirmişlerdir. Veri kümesi olarak, Küresel Konumlama Sistemi'nden (GPS) gerçek zamanlı veri toplamışlardır. Durum olarak, araç yoğunluğu, araçların hızları, araç hareketinin yönü gibi parametreler kullanmışlardır. Aksiyon olarak ise, ortamın mevcut durumunu kontrol eden bir etmen kullanmışlardır. Etmen bir sonraki zaman adımında hangi şeridin yeşil ışığa sahip olacağına karar vermektedir. Ödül olarak, amaç araçların bekleme süresini azaltmak olduğu için, araçların bekleme süresini kullanmışlardır. Bu süre, önceki zaman adımına kıyasla araçların kaç kez durduğunu ölçmektedir. Sonuç olarak, bekleme süresini, CO₂ emisyonunu ve araçların yakıt tüketimini azaltmayı başarmışlardır. Gelecek için, bu çalışmayı, gerçek trafik senaryoları, yinelemeli sinir ağı (RNN) gibi daha ileri derin sinir ağları (DNN), ve ayrıca daha karmaşık pekiştirmeli öğrenme algoritmaları kullanarak geliştirmeyi amaçlamışlardır.

Qadri vd. [24], İzmir'de trafik ışıkları olan bir dönel kavşak üzerinde çalışmışlardır. Mevcut altyapı olan trafik ışıklarına sahip dönel kavşağı, bir alternatif olarak trafik ışıkları olan kavşak ile karşılaştırmaya çalışmışlardır. Şu anki altyapıyı bir alternatifle değiştirmenin, sıkışıklığı, araçların her biri için ortalama gecikme süresi cinsinden azaltıp azaltmayacağını gözlemlemeyi amaçlamışlardır. Simülasyon olarak VISSIM'i ve üç farklı döngü uzunluğu için üç farklı trafik yoğunluk verisini kullanmışlardır. Sonuç olarak, simülasyondan, mevcut altyapıyı bir alternatif ile değiştirmenin, araç trafiği akışı için daha iyi olduğu görülmüştür. Altyapıdaki bir iyileşmenin her bir araç için ortalama gecikme süresini azaltarak trafik akışında daha iyi performans gösterdiğini gözlemlemişlerdir. Gelecek çalışma olarak, yaya ve diğer kavşakları da dikkate alarak simülasyonu genişletmeyi planlamaktadırlar. Ek olarak, trafik akışından daha fazla ölçüt toplamayı ve simülasyondan elde edilen sonuçları hesaplama cinsinden daha yetkin ve verimli olan bir modele dahil etmeyi planlamaktadırlar.

Kunjir vd. [25], A-DAC metodunu kullanarak, sinyalizasyon kavşaklar, çoklu kavşaklar ve dönel kavşaklarda üzerinde çevrim dışı (offline) öğrenme kullanarak kavşaklardan geçen araç sayısını maksimize etmeye çalışmışlardır. Makri ve Charalambous [26], pekiştirmeli öğrenme ve müfredat (curriculum) öğrenmesi kullanarak araçlar için bir model eğitmiş ve çarpışmaları minimize etmek için kullanmıştır. Almansoori vd. [27], dönel kavşaklara girmek üzere olan araçları saptayıp ve bu araçların mesafesini tahmin ederek, sürücülere yardımcı olacak yapay zekâ destekli bir asistan geliştirmişlerdir. Böylece kazaları önleyip trafik akışını artırmayı hedeflemişlerdir.

Baumgart ve Burger [28], pekiştirmeli öğrenme ile kontrol edilen bir otonom araç kullanmanın trafik akışını en optimal şekilde kontrol edebileceğini savunan bir çalışma yapmışlardır. Sistemdeki bulunan araçların sadece 20% ve 40%'ının gönderdiği verinin bile bunu başarmaya yeterli olduğunu kanıtlamışlardır.

Kunjir ve Chawla [29], trafik sinyal kontrolünü bir offline/batch pekiştirmeli öğrenme problemi olarak modelleyip, gerçek bir simülasyon ortamı ile etkileşime girmeden veriden bir politika üretmeye çalışmışlardır. Verinin yoğun olduğu yöntemde yüksek performanslı politikalar yaratmak için önemli bir potansiyel olduğunu gösteren bir değerlendirme yapmışlardır.

Baumgart ve Burger [30], trafik akışını, araçların ortalama hızlarına göre geliştirmeyi hedeflemiş ve sanal ortamda pekiştirmeli öğrenmenin trafik akış problemlerine uygulanabilirliği üzerine

çalışmıştır. Yaklaşımları sağlamlık ve stabilite olarak hala araştırmaya ihtiyaç duymaktadır.

Yuan vd. [31], dönel kavşaklarda otomatize araçların navigasyon ve kontrolünü hızlandırmak için derin pekiştirmeli öğrenme (DRL) kullanmış ve bazı kritik araştırma boşluklarını doldurmaya çalışmıştır. Wang vd. [32], sinyalizasyon olmayan dönel kavşaklarda ego-aracının güvenliğini artırmak üzerine Soft Aktör-Kritik (Soft Actor-Critic) algoritmasını kullanarak çalışmışlardır. Çarpışma önleme oranlarında 15%'lik bir iyileştirme sağlamasına karşın çalışmaları simülasyon üzerinde çalışmıştır.

Dong vd. [36] karma-otonom ortamlarda, güvenlik ve verimliliği geliştirmek için çok amaçlı derin pekiştirmeli öğrenme tabanlı adaptif transit sinyal önceliği kontrol yapısını önermiştir. Durum, aksiyon ve ödülleri tanımlamak için, bağlı ve otonom araçlardan gerçek zamanlı verileri kullanmıştır. Trafik çatışmalarını güven ödülü, araç bekleme sürelerini ise verimlilik ödülü olarak kullanmıştır. Rekabet halinde olan bu amaçları dengelemek amacıyla, araç tipi ve yolcu kapasitesine bağlı ağırlıklar atanarak transit sinyal önceliği stratejileri uygulanmıştır. Sonuç olarak 5:5 ağırlık oranı uygulandığında güven ve verim ödülllerinin en iyi sonuca ulaştığı ve tüm araç tipleri için gecikme ve çatışmaların en aza indirildiği görülmüştür. %100 bağlantılı otonom araçların olduğu ortamda gecikme ve çatışmalar en dengeli seviyede olurken, bağlantılı otonom araçların ve insan sürücülü araçların olduğu karma ortamlarda ise; bağlantılı otonom araçların %75 oranında olduğu durumun optimum güven ve verimlilik gösterdiği görülmüştür.

Agrahari vd. [37] adaptif trafik sinyal kontrol sistemlerdeki bulguları, teknik özellikleri ve çeşitli araştırmaların durumları üzerine kapsayıcı bir çalışma sunmuştur. Pekiştirmeli öğrenme, derin pekiştirmeli öğrenme ve hibrit yaklaşımlar gibi ileri tekniklerin, trafik yönetiminde özellikle seyahat sürelerini ve emisyonları azaltarak önemli gelişmeler gösterdiğini belirtmiştir. Genel olarak çalışmaların tek kavşaklı sistemlere odaklandığını ama sinyaller arasındaki koordinasyonun önemli olduğu çoklu kavşak sistemlerini eniyilemede de dikkate değer bir araştırma boşluğu olduğunu da belirtmiştir.

Karbasi vd. [38] bir derin pekiştirmeli öğrenme yaklaşımı kullanarak adaptif trafik sinyal kontrolünün, bağlı ve otonom araçların (CAV) kavşak güvenliğindeki etkilerini araştırmaya odaklanmıştır. Amaç CAVların ve adaptif trafik sinyal kontrolünün trafik güvenliği üzerindeki bireysel ve birleşik etkilerini, arkadan çarpma ve kesişerek çarpışmaları hesaba katarak değerlendirmektir. Trafik sinyallerini, CAVların ve sürücülü araçların sürüş davranışlarını düzenlemek için derin Q ağı (DQN)'i uygulamış ve güvenliğini değerlendirmek için çarpışma zamanı metriklerini kullanmıştır. Bulgular, CAVların ve DQN-bazlı trafik sinyal kontrolünün birleştirilmiş bir şekilde uygulanmasıyla, arkadan ve kesişerek çarpışmalarda önemli bir ölçüde azalma olduğunu göstermiştir.

Xiao vd. [39] kentsel ortamlarda yaygın olan tıkanıklık sorunlarını adresleyerek, farklı yol topolojilerinde trafik verimliliğini eniyilemek için bir karma trafik kontrol politikası sunmuştur. Sürücülü araçları yönlendirmek için otonom araçlardan toplanan verileri kullanarak, geniş ölçekli trafik akışını yönetmek için bir model tabanlı olmayan (model-free) pekiştirmeli öğrenme yaklaşımı geliştirmiştir. Bu çalışma, geniş coğrafik kapsamı ve farklı türdeki senaryo ve yol topolojilerini kapsamıyla aynı zamanda, sonraki çalışmalar için bir temel olarak kullanılabilir bir kıyaslama zemini de sunmuştur.

Usanmaz ve Güney [40] derin öğrenme tekniklerini kullanarak trafik güvenliği konusuna odaklanan bir çalışma sunmuştur. Günümüzde araçların gelişmiş sürücü destek sistemlerinde bulunan ve trafik işaretlerinin tanınmasını sağlayan yapıyı daha iyi hale getirmeyi

amaçlayan bir çalışma yapmıştır. Bu sistemlerdeki trafik işareti algılama yöntemlerinin gerçek zamanlı olarak, farklı ışık ve hava koşullarında daha iyi performans göstermesi için derin öğrenme algoritmaları ve transfer öğrenim yöntemini kullanıp sonuçları gözlemlemiştir. Ayrıca bu algoritmaların en iyi performans gösterdiği hiper parametreleri belirlemiştir. Böylece, ResNet50 ve SqueezeNet algoritmalarının daha iyi performans gösterdiğini belirtmiştir. İlerleyen dönemlerde bu çalışmaya gece trafiğine ait görüntüleri de dahil etmeyi ve araç içerisine yerleştirilen araç içi kameradan alınan görüntüleri kullanarak yol taraması gerçekleştirmeyi hedeflemektedir.

Turan vd. [41] kavşaklardaki sinyalizasyon problemine odaklanmıştır. Kavşaklardaki bekleme süresini ve kuyruk uzunluklarını azaltmak için faz süresi ve faz sırasını hesaplayıp birleştiren GDRL isimli yeni bir yaklaşım kullanmıştır. Faz sırasını derin pekiştirmeli öğrenme ile, faz süresini ise Ford-Fulkerson algoritmasına dayalı graf yöntemi kullanarak hesaplamıştır. Daha sonra bu modeli, Elazığ'da bulunan 2 kavşaktaki gerçek araç verileri kullanarak SUMO simülatorü üzerinde test etmiştir. Yapılan testler sonucunda bu çalışmanın sabit faz ve süre yöntemine göre kuyruk uzunluklarını %45 oranında azalttığını belirtmiştir.

Bugüne kadar olan alan yazılarını incelediğimizde, dönel kavşaklar, trafik ışıkları veya pekiştirmeli öğrenme üzerine yapılmış olan birçok çalışma mevcuttur, fakat bu üç konuya aynı anda odaklanan bir çalışma mevcut değildir. Bu çalışmanın önceki çalışmalara göre öne çıkan katkıları aşağıdaki gibi sıralanabilir:

- Dönel kavşaklarda, trafik ışıklarının pekiştirmeli öğrenme kullanılarak optimizasyonu
- Gerçek hayat veri kullanımı
- Deney tasarımı yöntemi ile hiper parametre optimizasyonu

2. Deneysel Metot (Experimental Method)

2.1. Q-Öğrenme Algoritması (Q-Learning Algorithm)

Q-Öğrenme algoritması, pekiştirmeli öğrenmede, en iyi bilinen ve aynı zamanda en temel algoritmalarından biridir. Temel amaçlarından bir tanesi, gelecekteki aksiyonları ve ödülleri öngörmek ve uzun vadedeki maksimum ödülü ve gelecekteki ödülleri de dikkate alarak, etmene bir sonraki aksiyonuna karar vermesinde yardımcı olmaktır. Durumların önemine göre hesaplanmış değerler ile doldurulmuş olan, örneğin, etmenin yönelmesi istenilen durumlar daha yüksek ödül değerlerine sahipken, etmenin ziyaret etmesi istenmeyen durumların daha düşük veya negatif değerlere sahip olduğu bir ödül tablosu tutar. Q-Öğrenme algoritması bu tabloyu, her yinelemede, yani eğitim döngüsü içindeki hesaplamasında kullanır. Ayrıca, algoritmanın adından da anlaşılabilir gibi, Q-Öğrenme algoritması, etmenin deneyim değerlerini kaydetmek için bir Q tablosuna da sahiptir. Bu tabloyu düzenli bir şekilde günceller ve bunu gelecekteki kararları aldığı sırada kullanır. Karar almak için kullandığı denklemde ödüllerin önemini belirlemek için, öğrenme faktörü ve γ (gama) gibi bazı kilit parametrelere sahiptir. Temel olarak denklem aşağıdaki gibidir (Eş. 1).

$$Q(s, a) = Q(s, a) + \alpha * (r(s, a) + \gamma * \max_{a'}(Q(s', a')) - Q(s, a)) \quad (1)$$

Buradaki denklemde, Q, eğer etmen s durumunda a aksiyonunu yaparsa ilgili Q değerini alabileceğimiz Q-tablosunu temsil eder. α (alfa) etmenin öğrenme oranını gösterir ve γ (gama) ise azaltma oranıdır. Denklemde göre etmen hem şu anki eyleminin ödülünü hem de bir sonraki yineleme için planladığı aksiyon ve durumun Q değerlerinden en büyüğünü kullanarak Q değerini günceller. Bir sonraki aksiyonunda Q değerini dikkate alarak, en iyi kararı verebilir ve ödülünü maksimize edebilir. Burada, algoritma, sonraki tüm

hareketleri araştırır ve en büyük ödül değerine sahip olanı seçer ve onunla devam eder. Ancak, ilk başta etmen hiçbir deneyime sahip olmadığı için Q-Tablosu tamamen sıfırlarla doludur. Q-Tablosundan farklı olarak, ödül tablosu, önceden bahsedildiği gibi durumun etmenine göre pozitif ve negatif değerlerle doldurulur. Ödül tablosu etmenin ziyaret etmesi istenilen durumlar için pozitif, etmenin kaçınmasını istediğimiz durumlar için ise negatif veya sıfır değere sahiptir. Sonuç olarak, ilk yinelemelerde Q-Tablosu sıfırlarla dolu olduğu için, ilk aksiyonlarına tamamen rasgele karar verecektir ve sıfır olmayan pozitif ödül değeri olan bir durum bulana kadar bunu yapmaya devam edecektir. Ayrıca, denklemde γ (gama) parametresi, Q değerinin hesaplanmasında gelecekteki ödüllerin katkısını belirlediği için önemli bir rol oynar. γ değeri 0 ve 1 arasındadır. 1 olması durumunda, etmen gelecekteki hareketleri dikkate alacaktır, ancak, 0 değerine sahip olduğunda gelecekteki aksiyonların çıktısı önemsiz olacaktır. Bu değer ayrıca keşif ve faydalanma dengesi de (trade-off) belirler.

Diğer makine öğrenmesi/yapay zekâ algoritmalarının çoğunda olduğu gibi, pekiştirmeli öğrenmede de keşif ve faydalanma dengesi vardır. Tahmin edilebileceği gibi, bir pekiştirmeli öğrenme probleminde, problemin karmaşıklığı nedeniyle arama uzayı bazen çok büyük boyutlara ulaşabilmektedir ve bu durum etmeni keşif ve faydalanma arasında denge kurmak için bir politika geliştirmek zorunda bırakır. Temel olarak, keşif, etmenin daha fazla durumu denemesini sağlarken, faydalanma ise, halihazırda elde edilmiş bilgiyi kullanarak ödülü maksimize etmeye çalışır. Etmen, tüm durumları keşfetmeyi seçebilir ve maksimum ödüle asla ulaşamama olasılığı dahilindedir. Çok fazla zaman harcayarak maksimum ödülü bulmayı deneyebilir. Benzer şekilde, etmen, arama uzayındaki ulaşabileceği en iyi ve maksimum ödül olduğunu düşündüğü için, aynı durum üzerinde aynı aksiyonu yapmayı devam ettirerek bu maksimum ödül değerinde kalmayı seçebilir, yani var olan durumdan faydalanmayı tercih edebilir. Ancak, bu yanıltıcı olabilir ve etmen yerel bir optimumda takılı kalabilir. Bu dengeyi iyileştirmek için, etmenin daha iyi karar vermesine yardımcı edebilecek bazı yöntemler vardır. Bu yöntemlerin en çok bilineni epsilon-açgözlü (epsilon-greedy) stratejisidir (Dutta [33]). Bu stratejide, etmenin davranışında kilit rol oynayan bir epsilon değeri vardır. Her yinelemede, rasgele bir sayı üretilir ve epsilon ile karşılaştırılır. Eğer değer epsilondan daha büyük ise, algoritma var olan çözümünden yararlanır. Aksi takdirde, yani epsilondan daha az olduğunda ise, etmen keşfedecektir demektir. Bunun yanı sıra, epsilon değerinin de eğitim süresince zamanla azaldığı dinamik ve böylece uyarlanabilen yaklaşımlar da olabilir. Pekiştirmeli öğrenme ve Q-Öğrenme algoritması hakkında daha fazla bilgi için Sutton ve Barto'nun kitabı incelenebilir [34].

Simülatorler gerçek hayatta olan sistemlerin veya süreçlerin taklit edilmesi için kullanılan araçlar olarak tanımlanabilir. Trafik akışında da her ne kadar sistemin işleyişi ve davranışı hakkında teorik bilgimiz olsa da simülatorler trafik sistemlerinin birçok farklı şartlarda tepkilerini analiz etmede önemli bir yere sahiptirler. Trafik akışını tanımlama konusunda yardımcı olup, ayrıca bu akışların güvenli ve uygun bir ortamda kapsamlı bir şekilde araştırılması ve incelenmesi konusunda oldukça önemlidirler.

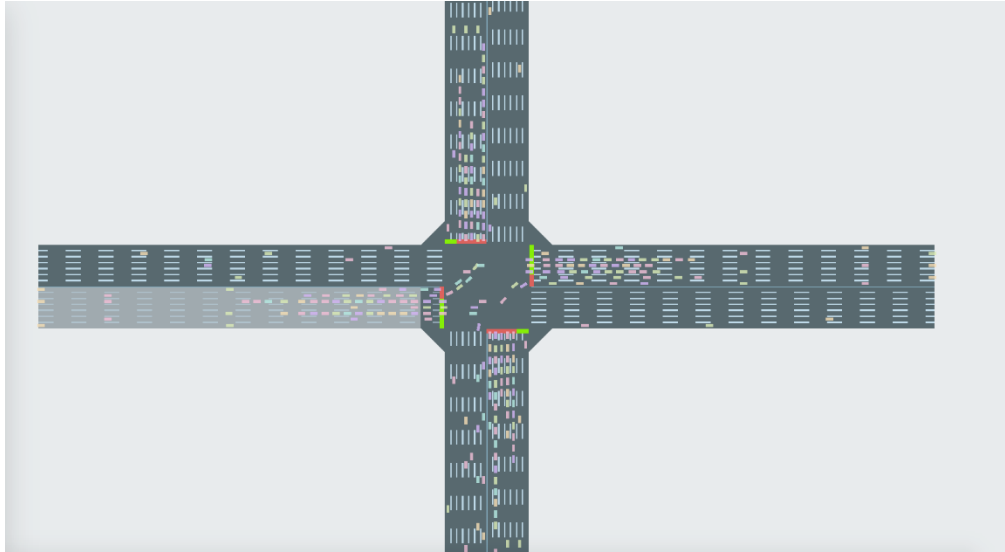
Trafik akışı için birçok simülator bulunmaktadır. SUMO en yaygın olarak kullanılan simülatorlerden biridir. Kısaltmasının açılımı kentsel hareketliliğin simülasyonu olan SUMO, Alman Havaçılık Merkezi (Deutsches Zentrum für Luft- und Raumfahrt – DLR) tarafından geliştirilmiş açık kaynaklı bir simülatördür. DLR ve Fraunhofer tarafından desteklenmektedir. Farklı işletim ortamlarına kolayca taşınabilir ve uygulanabilir. Dahası, kendini yukarı yönde ölçekleyerek çok geniş trafik ağlarını desteklemek için uyarlanabilir ve gerçek zamanda çok detaylı trafik bilgisi de sağlayabilmektedir.

CityFlow, geniş-ölçekli trafik simülasyonları için geliştirilmiş bir ortamdır. Ek olarak, çok etmenli pekiştirmeli öğrenme deneylerini destekler. SUMO'ya benzer bir şekilde, açık kaynaktır ve çok karmaşık trafik ağlarını destekleyebilir. SUMO'dan yirmi kat daha hızlı olduğu iddia edilmektedir ve SUMO'da olmayan çoklu iş parçacığı desteğine sahiptir. Bundan başka, Şekil 3'te gösterildiği gibi daha esnek kontrollere sahip ve daha kullanıcı dostu web tabanlı bir kullanıcı arayüzü sunmaktadır. Bu kullanıcı arayüzü, o an çalışan simülasyondaki araç sayısı, simülasyonun iterasyon sayısı gibi bilgiler içermektedir. Ayrıca, bu arayüz kullanılarak senaryo ve trafik ağı dosyaları yüklenebilir. Simülasyondaki çizelgeler için özel bir ayar dosyası da kullanılabilir. Buna ek olarak simülasyonun hızını artırıp azaltma veya simülasyonu durdurmak gibi kontrol seçenekleri de vardır. Web arayüzü ayrıca yol ve araç isimlerini ekranda göstermek gibi bazı hata ayıklama seçeneklerine de sahiptir. SUMO için üretilmiş olan yapılandırma ve trafik dosyalarını CityFlow'da kullanılmak üzere uyumlu hale getirmek için, bazı dönüştürme araçlarına da sahiptir.

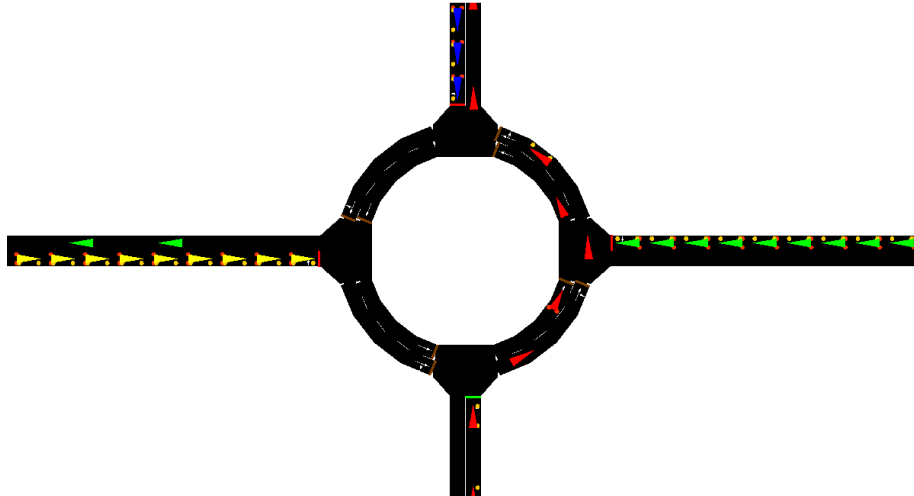
Bu çalışmada hem SUMO hem de CityFlow kullanılmıştır. CityFlow kullanımı kolaylığı, çoklu iş parçacığı desteği gibi özellikleri nedeniyle deneylerde kullanılmıştır. SUMO ise dönel kavşak desteği

nedeniyle kullanılmıştır. Orijinal olarak aslında ne CityFlow ne de SUMO dönel kavşakları doğrudan desteklememektedir. Fakat, SUMO için dolaylı bir yol vardır. Trafik ağı dosyasını, netedit olarak adlandırılan trafik ağı editörüne yükleyip, trafik ağı yapısını el ile düzenleyerek bir kavşağı dönel kavşağa çevirmek mümkündür. Onun dışında, CityFlow, SUMO trafik ağı ve trafik rotası dosyalarını CityFlow yol ağı ve yol akışı dosyalarına bir betik kullanarak çevirmek için desteğe sahiptir. Bu nedenle, CityFlow'da çalıştırmak için, ilk başta SUMO dosya formatına göre kavşak-bazlı trafik ağı oluşturulmuştur. Sonra, bu dosya, SUMO trafik ağı düzenleyicisinde düzenlenmiş ve bir dönel kavşağa çevrilmiştir. Daha sonra, trafik ağı ve trafik rotası dosyalarını CityFlow yol ağı ve akış dosyalarına çevirmek için CityFlow'un sağladığı destekleyici betik kullanılmıştır. Tüm bu adımlardan sonra, deneyler CityFlow'da yürütülebilir. Deneyler her iki simülatörün de artılarından faydalanmak amacıyla hem CityFlow'da hem de SUMO'da çalıştırılmıştır. Ayrıca bu iki simülatör daha önceki çalışmalarda da en çok tercih edilen sistemlerdir. Oluşturulan döner kavşağın SUMO simülatörünün kullanıcı arayüzünde görünümü Şekil 4'te verilmiştir.

Pekiştirmeli öğrenme, doğası nedeniyle, herhangi bir öncel veri olmadan da öğrenebilir. Trafik ışığı etmenlerini eğitmek ve değişen



Şekil 3. CityFlow web önyüz kullanıcı arayüzü (CityFlow web frontend user interface)



Şekil 4. SUMO simülatörünün kullanıcı arayüzü (User interface of SUMO simulator)

trafik akışı şartlarına göre öğrenmelerini sağlamak için SUMO veya Aimsun gibi simülátörler ve farklı pekiştirmeli öğrenme kütüphaneleri de daha önceki birçok çalışmada kullanılmıştır. Bu çalışmada OpenAI tarafından geliştirilmiş olan Gym kütüphanesi kullanılmıştır.

2.2. Deneyisel Çalışma (Experimental Study)

Çalışmada iki ana kategoride deneyler yapılmıştır. Bunlardan ilki deney tasarımı (Design of Experiment) adı verilen, algoritmalar için kullanılacak olan hiper parametrelerin değerlerinin belirlenmesi için yapılan deneylerdir. Diğer ise ilk aşamada deneyel olarak belirlenen hiper parametreleri kullanarak, sistemin farklı senaryolar üzerindeki davranışını inceleyen deneylerdir.

Deney tasarımı setinde, Q-Öğrenme algoritmasında kullanılan Bellman denkleminin, Q değerini, ödülleri ve etmenin keşif ve faydalanma oranlarını belirleyen sırasıyla, α (alfa) veya öğrenme oranı, γ (gama) veya indirim oranı ve ϵ (epsilon) veya keşif oranı (exploration rate) ve ödül tipi (reward type) parametrelerinden en iyi sonuçları veren kombinasyonu bulmak için deneyler yapılmıştır. Her bir parametre için olası değer kümelerini oluşturup bu deneylerin, bütün parametrelerin birbirleriyle olabilecek tüm kombinasyonlarını kapsayacak biçimde tam etkensel (full factorial) deney tasarımı yapılmıştır. Bu deneylerde eğitim aşamasında her bir yenilemede (replication) beş kez, bunun sonucunda oluşan Q-Tablosu kullanılarak bir kez de test aşaması için çalıştırılmıştır. Test aşaması sonucunda toplam bekleme süresi, toplam bekleyen araç sayısı, araçların ortalama hızı ölçütlerinin değerleri kaydedilmiştir. İstatistiksel olarak daha anlamlı sonuç elde etmek için bu deneyler farklı rasgele sayı üretici başlangıç değerleri (seed) kullanarak beş kez tekrar edilmiştir. Deneyde ele alınan hiper parametreler için değer kümeleri Tablo 1'de verilmiştir. Bu deneyde $3 \times 3 \times 3 \times 2 = 54$ parametre kombinasyonu 5 kere çalıştırılarak, en iyi değer veren hiper parametrelerin belirlenmesi için toplamda 270 değer elde edilmiştir.

Tablo 1. Deney tasarımında ele alınan parametre değerleri
(Parameter values considered for design of experiment)

Simge	Açıklama	Ele Alınan Değerler
α	Öğrenme Oranı	0,1; 0,3; 0,5
ϵ	Keşfetme Oranı	0,2; 0,3; 0,4
γ	İndirim Oranı	0,8; 0,9; 0,99
-	Ödül Tipi	delta bekleme süresi; ortalama hız

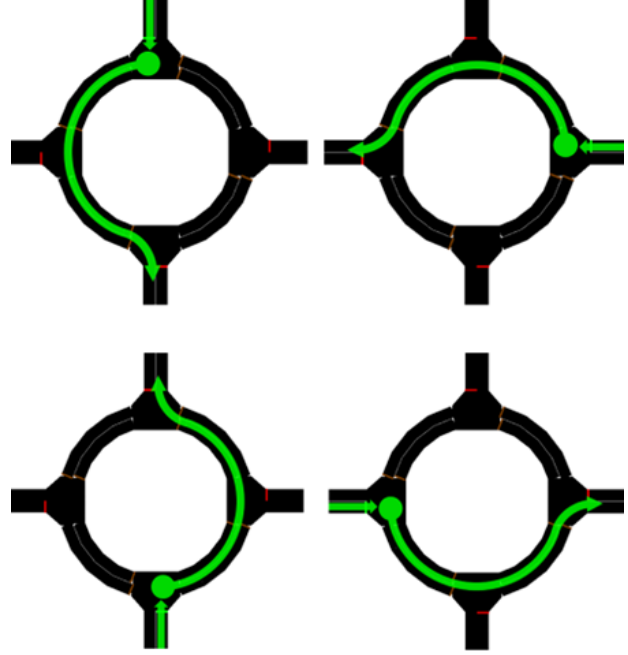
Bu deney için simülasyondaki trafik yoğunluğunun belirlenmesi için bir saat içinde sistemde oluşturulacak araç sayısı parametresi kullanılmıştır. Saatteki araç sayısı belirlenirken simülasyon sırasında hangi yönden ne kadar aracın zaman içerisinde sisteme ekleneceği ve araçların dönel kavşaktan geliş yönü yanında hangi yöne gitmeyi hedefleyeceklerinin belirtilmesi gerekmektedir. Yapılan deney tasarımında kullanılan saatteki araç sayıları, başlangıç ve hedef yönleri Tablo 2'de gösterilmiştir.

Tablo 2. Referans senaryo trafik araç yoğunluğu
(Baseline scenario traffic vehicle density)

Başlangıç Bacağı	Hedef Bacağı	Saatteki Araç Sayısı
Batı	Doğu	60
Güney	Kuzey	300
Doğu	Batı	300
Kuzey	Güney	300

Bu dönel kavşak tasarımında, dönel kavşağın dört tane ana bacağı vardır. Her bir ana kol, bir giriş ve bir çıkış olarak alt kollara ayrılmaktadır. Ayrıca her bir giriş bacağına dönel kavşağa bağlayan

noktada birer trafik ışığı bulunmaktadır. Bu dönel kavşaktaki rota tasarımına göre dönel kavşağın dört farklı kolundan dönel kavşağa giriş yapan araçlar, tam karşılarındaki koldan dönel kavşaktan çıkış yapmaktadırlar. Örnek olarak batı bacağından dönel kavşağa girmiş olan bir araç dönel kavşağa girip, güney bacağına pas geçip, doğu bacağına geldiğinde dönel kavşaktan çıkmaktadır. Yukarıdaki trafik yoğunluğu değerlerini göz önüne aldığımızda, her saatte 60 aracın batı bacağından dönel kavşağa girip doğu bacağından çıktığı, 300 aracın güney bacağından girip kuzey bacağından çıktığı, 300 aracın doğu bacağından girip batı bacağından çıktığı ve 300 aracın kuzey bacağından girip güney bacağından çıktığını söyleyebiliriz. Bu rotalar sol üstten başlayıp saat yönünde ilerleyecek şekilde sırasıyla Kuzey → Güney, Doğu → Batı, Batı → Doğu ve Güney → Kuzey olarak Şekil 5'te gösterilmiştir.



Şekil 5. Trafik akış rotaları (Traffic flow routes)

Bu deneylerde, test aşamasında toplam ve ortalama bekleme süresi, toplam ve ortalama bekleyen araç sayısı, araçların ortalama hızı ölçütleri toplanmıştır. Bu toplanan sonuçlar ANOVA metodu kullanılarak değerlendirilmiştir. ANOVA, parametreler arasındaki ilişkileri belirlemek ve hangi parametre değerlerini beraber veya tek başına kullanmanın en iyi olduğunu saptamak için kullanılabilecek istatistiksel bir yöntemdir. ANOVA metoduna göre, eğer tek bir parametre veya iki veya üç parametrenin etkileşimi için p değeri, 0,05'in altındaysa, bu değer istatistiksel olarak önemli sayılır. Ortalama bekleme süresine göre yapılan ANOVA analizi sonuçları Tablo 3'te verilmiştir. Bu tabloya göre tüm parametreler ayrı ayrı, yani öğrenme oranı, keşfetme oranı, indirim oranı ve ödül tipi, aynı zamanda öğrenme oranı ve keşfetme oranı, öğrenme oranı ve ödül tipi, keşfetme oranı ve ödül tipi, indirim oranı ve ödül tipi ve son olarak öğrenme oranı, keşfetme oranı ve ödül tipi etkileşimleri p değerleri 0,05'in altında olduğu için istatistiksel olarak anlamlıdır.

Parametre değerlerini belirleyebilmek için temel etki ve programın etkileşim grafik çıktıları incelenmiştir. Bu grafiklerde her bir parametrenin bağımsız olarak ve diğer parametrelerle etkileşim halinde sonuca etkisi görülmektedir. Deneyde amaç, bekleyen araç sayısını azaltmak olduğu için 4 parametrenin her biri için grafikteki en düşük değer en iyi performansı vermektedir. Aynı şekilde, birbirleriyle olan etkileşimlerinde de en düşük değere sahip

parametreler en iyi performansı vermektedir. Yapılan analiz sonucunda parametrelerin aşağıdaki değerlerinin en iyi sonuçları verdiği görülmüştür (Tablo 4).

Tablo 3. Deney tasarımı ANOVA sonuçları
(ANOVA results for design of experiment)

Parametre	F	p
öğrenme oranı	7,68	0,001
epsilon	5,14	0,007
gama	3,53	0,031
ödül tipi	11,85	0,001
öğrenme oranı*epsilon	2,97	0,020
öğrenme oranı*gama	1,75	0,140
öğrenme oranı*ödül tipi	7,68	0,001
epsilon*gama	1,01	0,403
epsilon*ödül tipi	5,13	0,007
gama*ödül tipi	3,53	0,031
öğrenme oranı*epsilon*gama	0,74	0,658
öğrenme oranı*epsilon*ödül tipi	2,97	0,020
öğrenme oranı*gama*ödül tipi	1,75	0,140
epsilon*gama*ödül tipi	1,01	0,402
öğrenme oranı*epsilon*gama* ödül tipi	0,74	0,657

Tablo 4. Deneyler için belirlenen parametre değerleri
(Chosen parameter values for experiments)

Simge	Açıklama	Değer
α	Öğrenme Oranı	0,5
ϵ	Keşfetme Oranı	0,2
γ	İndirim Oranı	0,99
-	Ödül Tipi	delta-bekleme-süresi

Daha sonra bu parametreleri kullanarak önerilen sisteminin üreteceği sonuçları belirlemek için farklı trafik akış senaryoları geliştirilmiştir. Deney tasarımında kullanılan senaryoyu çeşitlendirmek için 8 farklı senaryo ele alınmıştır. Bu senaryolardaki trafik yoğunlukları Tablo 5'te verilmiştir. Tablodaki sayılar ilgili koldan saatte geçen araç sayısını belirtmektedir. İstanbul Büyükşehir Belediyesi, farklı tarih ve zaman dilimleri için, İstanbul'daki kavşak ve dönel kavşaklardaki trafik verilerini paylaştığı bir web sitesi bulunmaktadır [35]. Deneylerimizde kullandığımız sayıları belirlerken, olabildiğinde gerçek veri ile çalışmak için, bu web sitesindeki verileri kontrol edip en yoğun araç trafiği sayılarını alıp kullandık.

Tablo 5. Farklı senaryoların saatlik araç yoğunluk oranları
(Vehicle hourly density rates of different scenarios)

Senaryo No	Batı	Doğu	Güney	Kuzey
1	60	300	300	300
2	60	60	300	300
3	60	300	60	300
4	60	60	60	60
5	300	300	300	300
6	300	60	300	300
7	300	300	60	300
8	300	60	60	60

Her bir deney beş kez eğitim ve bir kez de test olarak çalıştırılmıştır. Testlerde, ortalama delta bekleme süresi, ortalama bekleyen araç sayısı, araçların üç farklı şekilde hesaplanan ortalama hızları ölçütlerine göre veri toplanmıştır. Daha sonra karşılaştırma yapabilmek için, temel durum değerleri deneyleri çalıştırılmıştır. Temel durum değeri belirleme deneylerinde, bir pekiştirmeli öğrenme etmeni ve algoritması kullanmak yerine, trafik ışığını sabit sürelerle belirli biçimde, yani saat yönünde sırayla her bir bacağı yeşil ışık yakıp geçiş hakkı vereceği şekilde tasarlanmıştır. Sabit süre değeri olarak, simülatördeki varsayılan değer olan 23 saniye kullanılmıştır.

Böylece temel durum değeri belirleme deneyi herhangi bir başka girdiyi dikkate almadan ve herhangi bir yöne öncelik vermeden sırayla bu kollardaki araçlara yeşil ışık vermektedir. Karşılaştırma yapabilmek için temel durum değeri belirleme deneylerinde de ortalama bekleme süresi, ortalama bekleyen araç sayısı, araçların 3 farklı hesaplama ile ortalama hızları ölçütlerine göre veri toplanmıştır. Aynı trafik parametreleri kullanılsa bile simülasyon rasgelelik içerdiği için her bir deney seti (eğitim-test-temel durum değeri) beş kez tekrar edilip bu beş deneyin sonucunda minimum, maksimum, ortalama ve standart sapma değerleri hesaplanmıştır.

Deneyler, Windows 11 (21H2) işletim sistemi çalıştıran, 2.90 GHz AMD Ryzen 7 4800H işlemci, 16GB RAM bellek, 8367 MB disk ve Radeon Graphics GPU ekran kartı olan bir dizüstü bilgisayarda çalıştırılmıştır. Yazılım olarak SUMO simülatörünün 1.12.0 sürümü, OpenAI tarafından sunulan Gym kütüphanesinin 0.26.2 sürümü ve Python dilinin 3.7 sürümü kullanılmıştır. Ayrıca, daha düzenli bir çalışma ortamı sağlamak için Python dilinin sunmuş olduğu sanal ortam (virtualenv) yapısını kullanarak izole bir ortam oluşturup TensorFlow, Keras, NumPy vb. gibi diğer tüm gerekli yardımcı kütüphaneler de bu ortama yüklenip birlikte kullanılmıştır.

2.2.1. Kullanılan Q-Öğrenme algoritmasının tasarımı (Design of Utilized Q-Learning Algorithm)

Deneyler için tasarlanan trafik senaryoları, Batı, Kuzey, Doğu ve Güney kollarından aynı anda sadece bir tanesine yeşil ışık verilecek şekilde tasarlanmıştır. Dolayısıyla, bu 4 koldan birine yeşil yandığı sırada diğer kalan 3 kolda kırmızı veya sarı ışık yanmaktadır. Bu tasarımda sistemin dört koldan birisini yeşil ışık yakmak için seçmesi gerekmektedir. Bu da sistemde 4 tane aksiyon seçeneği olduğu anlamına gelmektedir.

Oluşturulan Q-Öğrenme algoritmasını temel alan sistemdeki etmen, öğrenme aşaması deneylerinde o sıradaki sistemin durumuna bakıp, epsilon-greedy yöntemi kullanarak keşfetmek ve faydalanmak arasında bir karar verecektir. Keşfetmeyi seçerse rasgele bir durum seçecektir. Faydalanmayı seçmesi halinde Q-Tablosundan o anki bulunduğu duruma göre, seçebileceği dört aksiyon için Q değerlerini inceleyecek ve ödül değerini maksimize etmeye çalışacaktır. Böylece etmen, en iyi ödül değerine ulaştıracak olan aksiyonu seçip o bacağı yeşil ışık yakacaktır. Test aşaması deneylerinde ise, Q-Tablosu son halini aldığı ve artık sistemin öğrenmiş olduğu varsayıldığı için, epsilon-greedy yaklaşımı yerine, etmen Q-Tablosuna bakarak o anki bulunduğu duruma göre, seçebileceği dört aksiyon için Q değerlerini inceleyecek ve maksimum değeri veren ödüllü seçecektir. Böylece rastlantısallık yerine, eğitim aşamasında oluşan Q-Tablosundaki en iyi değeri kullanarak ilerleyecektir. Burada not edilmesi gereken önemli bir nokta ise, eğer etmen Q-Tablosundaki değerlere göre en iyi faydayı vereceğini düşünürse, halihazırda yeşil yanmış olan bacağı tekrardan yeşil yakmayı seçebilir.

Öğrenme aşaması deneyleri için aksiyon seçimi algoritması aşağıda gibidir (Şekil 6).

rasgele bir sayı seç

eğer rasgele seçilen sayı, keşfetme oranı'ndan küçük ise

aksiyon uzayından yeni bir aksiyon seç

değilse

Q-Tablosundaki maksimum ödüllü veren aksiyonu seç

seçilen aksiyon değerini döndür

Şekil 6. Öğrenme fazı için aksiyon seçimi algoritması
(Action selection algorithm for training phase)

Test aşaması deneyleri için ise aksiyon seçimi algoritması aşağıdaki gibidir (Şekil 7).

Q-Tablosu'ndan maksimum ödüllü veren aksiyonu seç

Şekil 7. Test fazı aksiyon seçimi algoritması
(Action selection algorithm for testing phase)

Temel durum değeri deneyleri ise, yukarıdaki pekiştirmeli öğrenmeyi kullanan Q-öğrenme algoritması tabanlı sistemin aksine, herhangi bir ödül, fayda hesabı yapmadan ve keşif, faydalanma stratejisi gütmeyen, geleneksel trafik ışıklarında olduğu gibi, basit bir şekilde bütün kollarla sırasıyla ve sabit süreyle yeşil ışık yakacak şekilde tasarlanmıştır. Bu deneylerde tüm kollarla saat yönünde sırasıyla yeşil ışık yanar. Daha net belirtmek gerekirse, kuzey bacağında yeşil ışık yandıktan sonra, doğu bacağında yeşil ışık yanmaya başlayacaktır. Daha sonrasında ise sırasıyla güney ve batı kollarına yeşil ışık yanacaktır. Temel durum deneyleri için aksiyonların algoritması aşağıda verilmiştir (Şekil 8).

aksiyon değişkenini 0'a eşitle

simülasyon devam ettiği sürece

aksiyon değişkenini simülasyona'a ver

dönen sonuçları ilgili değişkenlere ata

aksiyon değişkeninin değerini 1 artır

aksiyon değişkeninin 4'e göre modunu al

Şekil 8. Referans fazı aksiyon seçimi algoritması
(Action selection algorithm for baseline phase)

Algoritmada görüldüğü gibi, etmenin seçebileceği dört tane aksiyon olduğundan aksiyon 0, 1, 2 ve 3 değerlerini almaktadır ve değeri 4 olduğunda, mod operatörü ile değeri, ilk değer olan 0'a geri dönmektedir. Buradaki 0, 1, 2 ve 3 değerlerinin her biri sırasıyla dönel kavşağa bağlanan Kuzey, Doğu, Güney ve Batı kollarını temsil etmektedir.

2.2.2. Durum temsili ve Q-tablosu tasarımı (State Representation and Q-Table Design)

Sistemin herhangi bir andaki durumunu temsil etmek için, trafik ışıklarının ve trafik akışının o andaki durumu kullanılmıştır. Bunun için bir dizi kullanılmıştır. Bu dizinin ilk dört elemanı trafik sisteminde o an hangi kol için yeşil ışık yandığını gösteren bir bit dizisidir. Dönel kavşağın kolları sırasıyla 0, 1, 2 ve 3 indisleriyle gösterildiği için, o an yeşil ışık yanan bacağın bit dizisindeki değeri 1, geri kalanlar ise 0 olacak şekilde düzenlenmiştir. Daha sonra ise, halihazırda yeşil yanan kol için, yeşil süresinin belirlenmiş olan minimum yeşil süresini geçip geçmediğini belirten bir etiket eklenmiştir. Başka bir deyişle, aynı kol için bir önceki durumda da yeşil yanıp yanmadığını göstermek için, bir değer eklenmiştir. Eğer ışık kendisine ayrılan minimum yeşil zamanını geçmişse, yani daha önceden de bu bacağı yeşil yanmışsa ve tekrar bu kol seçilmişse 1, bu bacağı yeşil ilk kez yanmışsa 0 olacak şekilde düzenlenmiştir. Bunlara ek olarak ise, her bir koldaki araç yoğunluğu da durum temsiline eklenmiştir. Böylece dört bacağın her birinde bulunan araç yoğunluğu da dört ayrı değer olarak durum dizisine eklenmiştir. Araç yoğunluğunu hesaplamak için ise, o koldaki son zaman adımıdaki araç sayısı, o bacağın toplam araç sayısı kapasitesine bölünmüştür. Somut bir şekilde göstermek gerekirse, aşağıdaki örnek dizi kullanılabilir.

[1; 0; 0; 0; 0,01594184; 0,01594184; 0,01594184; 0,01594184]

[0; 0; 1; 0; 0; 0,04782553; 0,04782553; 0,04782553; 0,01594184]

Son satırdaki örnek dikkate alındığında, ilk dört eleman hangi kolda o an yeşil ışık yandığını göstermektedir ve şu anki durumda ise iki numaralı kolda, yani güney bacağında yeşil ışık yanmaktadır. Dört elemmandan sonraki ilk eleman ise, 0 olduğu için, güney bacağında bu adımda ilk defa yeşil yandığı söylenebilir. Dizideki geri kalan son dört rakam ise her bir kol için araç yoğunluğunu göstermektedir. Yoğunluk değeri için, o şeritteki o anda bulunan araç sayısının, o şeritte olabilecek toplam araç sayısı, varsayılan araç boyu ve iki araç arasındaki mesafe dikkate alınarak bir hesaplama yapılmıştır. Hesaplamaların formülü aşağıda gösterilmiştir (Eş. 2). Bu formülde D yoğunluk değerini, N o anda şeritteki araç sayısını, N_{max} o şeride sığabilecek maksimum araç sayısını, l_v varsayılan araç boyunu, d araçlar arası mesafeyi ve L ise şerit uzunluğunu ifade etmektedir. Buna göre sırasıyla, kuzey bacağı 0,04782553, doğu bacağı 0,04782553, güney bacağı 0,04782553 ve son olarak batı bacağı 0,01594184 yoğunluğa sahiptir. Var olan durumda batı bacağında bekleyen araç sayısının diğer kolları bekleyen araçların dörtte biri kadar olduğu gözlenmektedir.

$$D = \frac{N}{N_{max}} = \frac{N}{\frac{L}{(l_v + d)}} = \frac{N * (l_v + d)}{L} \quad (2)$$

Q-tablosu ise, sistemin o an içinde bulunduğu durumu, olası aksiyonlarla birleştirip, etmenin o anki durumdan bir sonraki duruma geçmek için seçeceği her bir aksiyonun kendisine ne kadar fayda sağlayacağını değerini tutan bir tablodur. Etmenin yeşil ışık yakabileceği 4 tane kol olduğu düşünüldüğünde, etmenin her durumda seçebileceği 4 tane aksiyon vardır. Dolayısıyla, Q-tablosuna durum değeri verildiğinde dört farklı değer alınmalıdır. Bunu sağlamak için anahtar-değer yapısının kullanılabileceği Python dilinde mevcut olan sözlük (dictionary) yapısı kullanılmıştır. Fakat durum temsili tek bir değer yerine birden fazla değerden oluştuğu için, anahtar olarak bir dizi (array) tipi kullanılmıştır. Ayrıca, durum temsili ilk dört eleman her bir bacağı temsil edip, hangi kolda o an yeşil ışık yanıyorsa onu 1, diğerlerini ise 0 değeri ile temsil ediyorken, Q-Tablosunda bunu biraz sadeleştirip, sadece o an yeşil ışık yanan bacağın indisi kullanılmıştır. Böylece anahtarın uzunluğu da biraz kısaltılarak başarımın artırılması sağlanmıştır. Q-tablosunun herhangi bir andaki değerleri aşağıdaki gibi olabilir.

(0; 0; 0; 0; 0; 0; [3,2046378738948462; 0; 0; 1,881745461696459])

(3; 1; 1; 2; 2; 1; [4,0273408062275005; 4,209953951922076; 5,538714258411689; 2,797867215882488])

Örnek olarak buradaki son girdi incelendiğinde, ilk eleman olan 3, 3 numaralı kolda yani batı bacağında o an yeşil ışık yandığını ifade eder. Bir sonraki değer olan 1 ise, yeşil ışığın minimum yeşil süresini geçtiğini, yani bir önceki adımda da 3 numaralı bacağı yeşil yandığını ifade eder. Daha sonraki 1, 2, 2, 1 değerleri ise, sırasıyla 0, 1, 2, ve 3 numaralı kollarıdaki araç yoğunluklarının ayrılaştırılarak 0-9 arası tam sayı bir değere dönüştürülmüş halini temsil eder, yani, araç yoğunluğunu ayrılaştırılmış hali kuzey bacağında 1, doğu bacağında 2, güney bacağında 2 ve batı bacağında ise 1 olarak gerçekleşmiştir. Ayrılaştırma için, durum temsiliindeki araç yoğunluğu değerini alıp 10 ile çarpıp çıkan değeri bir üst tam sayıya yuvarladıktan sonra oluşan tam sayı değeri 0-9 arası olacak şekilde döndürülür. Sonuç olarak hesaplanan değer 9'un altındaysa, sayının kendisi doğrudan alınmakta, ancak çıkan sayı 9'dan büyük olursa, 9 değeri alınmaktadır. Bu hesaplamanın formülü x araç yoğunluğunu ifade edecek şekilde aşağıda gösterilmiştir (Eş. 3). Burada etmen, bir duruma geldiğinde ve bir adımı tamamladığında bu duruma uygun şekilde Bellman eşitliğine göre yeni Q değerini hesaplayıp, Q-Tablosunda ilgili yeri günceller. Eğer, Q-Tablosunda, o durumu ifade eden bir girdi yok ise yeni girdi olarak ekler, eğer girdi var ise onu yeni değerler ile günceller.

$$f(x) = \{[10x], \text{ eğer } [10x] \leq 9, \text{ eğer } [10x] > 9, x \in [0, 1] \quad (3)$$

Tablo 6. Farklı senaryolar için elde edilen deney sonuçları (Experiment results for different scenarios)

Senaryo	Deney Tipi	Değer	Ort. Duran Araç	Ort. Bekleme Süresi	Ort. Hız	Ort. Hız 2	Ort. Hız 3
1	Test	Ortalama	7,461	241,917	0,362	6,709	7,916
		std. sapma	0,799	168,501	0,016	0,305	0,217
	Temel Durum	Ortalama	144,040	3240,663	0,038	0,705	1,261
		std. sapma	0,084	1,902	0,000	0,005	0,005
2	Test	Ortalama	5,323	143,903	0,374	6,930	8,076
		std. sapma	1,311	131,243	0,030	0,561	0,416
	Temel Durum	Ortalama	5,268	76,368	0,367	6,828	8,007
		std. sapma	0,218	4,869	0,006	0,118	0,080
3	Test	Ortalama	5,777	216,478	0,360	6,690	7,896
		std. sapma	1,084	174,338	0,024	0,442	0,327
	Temel Durum	Ortalama	53,582	1223,557	0,074	1,377	2,378
		std. sapma	0,714	16,437	0,001	0,012	0,020
4	Test	Ortalama	2,115	131,337	0,355	6,515	8,028
		std. sapma	0,964	133,111	0,056	1,022	0,966
	Temel Durum	Ortalama	1,192	12,336	0,406	7,457	8,956
		std. sapma	0,003	0,022	0,001	0,004	0,006
5	Test	Ortalama	28,331	579,920	0,200	3,708	5,290
		std. sapma	7,559	174,651	0,039	0,722	0,743
	Temel Durum	Ortalama	201,625	4587,538	0,027	0,496	0,892
		std. sapma	0,036	0,836	0,000	0,001	0,001
6	Test	Ortalama	6,546	119,658	0,380	7,053	8,159
		std. sapma	0,582	55,171	0,013	0,248	0,174
	Temel Durum	Ortalama	143,259	3225,640	0,038	0,704	1,261
		std. sapma	0,271	6,176	0,000	0,000	0,001
7	Test	Ortalama	7,391	125,316	0,360	6,690	7,906
		std. sapma	0,779	42,310	0,016	0,293	0,208
	Temel Durum	Ortalama	127,056	2907,518	0,041	0,758	1,356
		std. sapma	1,657	38,239	0,001	0,009	0,016
8	Test	Ortalama	3,902	191,949	0,368	6,822	8,022
		std. sapma	1,527	194,151	0,048	0,887	0,717
	Temel Durum	Ortalama	2,659	33,126	0,411	7,589	8,678
		std. sapma	0,173	3,766	0,010	0,183	0,116

2.2.3. Ödül fonksiyonu tasarımı ve algoritması (Reward Function Design and Algorithm)

Pekiştirmeli öğrenme algoritmalarında etmenin amacı ödülün en iyi değerine ulaşmaktır. Problemin türüne göre bu en yüksek ödül veya en düşük ceza olabilir. Ödül fonksiyonu tasarımı genelde optimize edilmek istenilen değer seçilir. Bu problemde trafik akışındaki anahtar iki ölçüt olan bekleme süresi ve ortalama hız iki farklı ödül olarak seçilmiştir. Deney tasarımı aşamasında, en iyi parametreleri bulmak için yapılan deneylerde de bu iki farklı ödül türü kullanılarak, bu ödüllerden hangisinin daha iyi sonuç verdiği belirlenmiştir. Sonuç bölümünde de görüleceği üzere, delta bekleme süresini ödül parametresi olarak kullanılmanın en iyi sonuçları verdiği gözlenmiştir. Bu nedenle, sistemdeki araçların bekleme süresini azaltacak bir ödül fonksiyonu tasarımı gerçekleştirilmiştir. Aşağıda Şekil 9'da ödül fonksiyonunun algoritması verilmiştir.

tüm şeritlerdeki bekleme süresini toplayıp 100'e böl

sonucu bekleme süresi değişkenine ata

önceki bekleme süresinden, yeni bekleme süresini çıkar

sonucu ödül değeri değişkenine ata

yeni bekleme süresini, önceki bekleme süresine ata

ödül değeri sonucunu döndür

Şekil 9. Ödül fonksiyonu algoritması (Reward function algorithm)

Her bir şeritteki bekleyen araçların bekleme sürelerinin toplamı ölçeği küçültmek için 100'e bölünmüş, daha sonra ise bu yeni değer, bir önceki yinelemeye hesaplanmış olan eski değerden çıkartılıp farkı alınmıştır ve bu fark ödül değeri olarak döndürülmüştür.

3. Sonuçlar ve Tartışmalar (Results and Discussions)

Tablo 6'da geliştirilen sistemin yukarıda belirtilen sekiz farklı senaryonun her biri için test ve temel durum belirleme deneylerinin sonuçları verilmiştir. Tablo incelendiğinde, trafiğin yoğun olduğu yani bazı kollardaki bekleyen araç sayısının diğer senaryolara göre yaklaşık 5 kat daha fazla olduğu ve araç hacminin asimetric olduğu 1, 3, 5, 6 ve 7 numaralı senaryolarda ortalama duran araç sayısı ve ortalama delta bekleme süresi açısından, geliştirilen çözüm ciddi bir iyileştirme gerçekleştirmiş ve temel durum değerlerine kıyasla çok düşük değerlere ulaşabilmiştir. Bununla birlikte üç farklı ortalama hız değerlerinde de aynı şekilde temel durum değerlerine göre çok daha iyi bir başarıyı gösterip, oldukça yüksek bir ortalama hıza ulaşmıştır. Somut olarak etkisini ifade etmek gerekirse, ortalama duran araç sayısı %91 civarında, ortalama bekleme süresi de %90 civarında bir azalma göstermiştir. Bu iki ölçütün bu oranda iyileşmesi, her bir sürücüyü 2 kata yakın zaman kazandırmakla birlikte, birim zamanda bu dönel kavşaktan geçen araç sayısı da bu oranda artmış, ek olarak CO₂ ve CO emisyonları da azalmıştır.

Bununla beraber, 2, 4 ve 8 numaralı senaryolarında ise, ortalama bekleyen araç sayısı bakımından temel durum değerlerine yakın, ancak biraz daha kötü değerler elde etmiştir. Aynı senaryolarda,

ortalama delta bekleme süresi ölçütünde ise, temel durum değerleri daha iyi sonuçlar göstermektedir. Ortalama hız değerlerine bakıldığında ise, geliştirilen çözüm üç farklı hız hesaplamasında da temel durum değerlerine oldukça yakın değerlere ulaşmıştır.

4. Sonuçlar (Conclusions)

Çalışmanın sonuçlarına genel olarak bakıldığında geliştirilen sistem 8 senaryonun 5 tanesinde, 5 farklı ölçütte temel durum değerlerinden oldukça iyi performans gösterip, geri kalan 3 senaryoda da 5 ölçütün 4 tanesinde temel durum değerleriyle oldukça yakın değerler gösterip, sadece tek bir ölçütte temel durum değerlerine göre kötü bir değere ulaşmıştır. Ayrıca başarılı olduğu senaryolarındaki araç yoğunluklarına bakıldığında ise, araç yoğunluğunun fazla olduğu durumlarda oldukça başarılıdır. Araç yoğunluğunun nispeten daha az olduğu durumlarda ise, temel durum değerlerine yakın sonuçlar göstermiştir. 4 ve 5 numaralı senaryolarda dönel kavşağın her bacağında eşit sayıda araç beklemektedir. Fakat 5 numaralı senaryodaki her bir koldaki araç sayısı 4 numaralı senaryoya göre 5 kat daha fazla olduğu için, bu sistemde bir yoğunluk yaratmaktadır ve dolayısıyla Q-Öğrenme algoritması yoğunluğun daha fazla olduğu 5 numaralı senaryoda, temel durum değerinden daha iyi bir sonuç göstermektedir. Araç yoğunluklarının aynı olduğu 2 ve 3 numaralı senaryolarda farklı sonuçlar alınmasının sebebi ise, 2 numaradaki araç yoğunluklarının az olduğu kollar birbirine tam karşı yönden baktığı için yoğunluk dengelenmiştir ve fazla bir yoğunluk oluşmamıştır. Dolayısıyla temel durum değerlerine yakın sonuçlar üretmiştir. Ancak 3 numaralı senaryoda ise birbirine komşu olan kollarla yüksek araç yoğunluğu olduğu için trafik sıkışmalarına sebep olmuştur ve bu durumda önerilen sistem daha iyi performans göstermiştir. Alan yazındaki çalışmalar genelde aynı veya benzer algoritmaları kullanıp normal sinyalizasyon kavşakları üzerinde çalışmışlar ve genellikle sadece tek bir ölçütü eniyilemişlerdir. Onlara kıyasla, bu çalışma birden fazla ölçütü aynı anda eniyilemeyi başarmıştır.

Bu çalışmada önerilen çözüm tek bir dönel kavşak üzerinde etkili olmuştur, fakat gerçek dünyada birden fazla dönel kavşak veya şehir çapında trafik ağları mevcuttur. Birden fazla dönel kavşak ya da trafik ağında, dönel kavşakların herhangi birinde oluşacak bir trafik sıkışıklığı diğer dönel kavşakları da etkileyecektir. Dolayısıyla bu, çoklu sistemlerde iletişimi önemli bir etken haline getirmektedir. Halihazırda bu çözümde kullanılan algoritma bu konuda yetersiz kalmaktadır. Bu etkeni göz önünde bulundurarak, ilerisi için bu çözümü çoklu dönel kavşaklarda uygulayabilmek için Değer Ayırıştırma Ağları (VDN), Monotonik Değer Karışım Ağı (QMIX), Çoklu Etmen Yakınsal Politika Optimizasyonu (MAPPO) gibi çok-etmenli pekiştirmeli öğrenme algoritmaları kullanılarak var olan çözümün genişletilmesi konusunda çalışmalar yapılabilir. Bunun yanı sıra bu çalışma, her ne kadar CO₂ ve CO emisyon değerleri gibi çevresel ölçütlere odaklanan bir çalışma olmasa bile; ortalama duran araç sayısı ve ortalama bekleme süresi gibi ölçütlerin çevresel emisyon değerleriyle arasında bir korelasyon oluşturduğu bilinir. Bu sebeple bahsedilen ölçütlerde yapılan iyileşme daha düşük emisyon değerleri oluşmasına pozitif katkı sağladığı düşünülmektedir.

Gelecek çalışma olarak ise, şu an optimize edilen ortalama duran araç sayısı, ortalama bekleme süresi, ortalama hız gibi ölçütlerin yanında, CO₂ ve / veya CO emisyonu, yakıt tüketimi ve gürültü emisyonu gibi çevresel ölçütler konusunda da sistemin optimize edilip temel durum değerlerine göre daha iyi performans gösterebilecek biçimde sistemin geliştirilmesi planlanmaktadır.

Kaynaklar (References)

1. EDMS Consultants Sdn Bhd. 4 Artificial Intelligence (AI) Trend Predictions for 2020. <https://www.edms-consultants.com/4-artificial-intelligence-ai-trend-predictions-for-2020/>. Yayın tarihi Şubat 6, 2020. Erişim tarihi Nisan 24, 2023.
2. Srivastava S. Computer Vision Makes Autonomous Vehicles Intelligent and Reliable. <https://www.analyticsinsight.net/computer-vision-makes-autonomous-vehicles-intelligent-and-reliable/>. Yayın tarihi 2019. Erişim tarihi Nisan 24, 2023.
3. Pilgrim T. Traffic sign recognition “most influential” innovation of past decade. <https://phys.org/news/2019-05-traffic-recognition-influential-decade.html>. Yayın tarihi Mayıs 29, 2019. Erişim tarihi Nisan 24, 2023.
4. Howard B. How does lane departure warning work?. <https://www.extremetech.com/cars/165320-what-is-lane-departure-warning-and-how-does-it-work>. Yayın tarihi Şubat 27, 2017. Erişim tarihi Nisan 24, 2023.
5. Wheatley M. Big Data Traffic Jam: Smarter Lights, Happy Drivers. <https://siliconangle.com/2013/04/03/big-data-traffic-jam-smarter-lights-happy-drivers/>. Yayın tarihi Nisan 3, 2013. Erişim tarihi Nisan 18, 2023.
6. Yau K.L.A., Qadir J., Khoo H.L., Ling M.H., Komisarczuk P., A Survey on Reinforcement Learning Models and Algorithms for Traffic Signal Control, ACM Computing Surveys (CSUR), 50 (3), 1-38, 2017.
7. Bullock D. Traffic signals: The perfect smart city tool. <https://njac.org/traffic-signals-the-perfect-smart-city-tool/>. Yayın tarihi 2016. Erişim tarihi 2023.
8. El-Tantawy S., Abdulhai B., Abdelgawad H., 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, 2013.
9. Gollapudi S., Practical Machine Learning, Packt Publishing, Birmingham, İngiltere, 2016.
10. Rizzo S.G., Vantini G., Chawla S., Time Critic Policy Gradient Methods for Traffic Signal Control in Complex and Congested Scenarios, KDD '19: The 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK-A.B.D., 1654–1664, 4-8 Ağustos, 2019.
11. Gökçe M.A., Öner E., Işık G., Traffic signal optimization with Particle Swarm Optimization for signalized roundabouts, Simulation, 91 (5), 456-466, 2015.
12. Erdem O.A., Kavşak Trafik Sinyalizasyon Sistemi İçin Bulanık Mantık Tabanlı Gerçek Zamanlı Denetleyici Tasarımı ve Uygulaması, Engineering Sciences, 2 (4), 241-255, 2007.
13. Chin Y.K., Kow W.Y., Khong W.L., Tan M.K., Teo K.T.K., Q-Learning Traffic Signal Optimization within Multiple Intersections Traffic Network, 6th UKSim/AMSS European Symposium on Computer Modelling and Simulation, Malta-Malta, 343–348, 14-16 Kasım, 2012.
14. Chin Y.K., Lee L.K., Bolong N., Yang S.S., Teo K.T.K., Exploring Q-learning Optimization in Traffic Signal Timing Plan Management, 3rd International Conference on Computational Intelligence, Communication Systems and Networks, Bali-Indonesia, 269-274, 26-28 Temmuz, 2011.
15. Zeng J., Hu J., Zhang Y., Adaptive Traffic Signal Control with Deep Recurrent Q-learning, 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu-Çin, 1215-1220, 26-30 Haziran, 2018.
16. Ducrocq R., Farhi N., Deep Reinforcement Q-Learning for Intelligent Traffic Signal Control with Partial Detection, International Journal of Intelligent Transportation Systems Research, 21, 192–206, 2023.
17. Wang H., Yuan Y., Yang X.T., Zhao T., Liu Y., Deep Q learning-based traffic signal control algorithms: Model development and evaluation with field data, Journal of Intelligent Transportation Systems, 27 (3), 314-334, 2023.
18. Ahmed M.A.A., Khoo H.L., Ng O.E., Discharge control policy based on density and speed for deep Q-learning adaptive traffic signal, Transportmetrica B: Transport Dynamics, 11 (1), 1707-1726, 2023.
19. Meneguzzo C., Gastaldi M., Rossi R., Gecchele G., Prati M.V., Comparison of exhaust emissions at intersections under traffic signal versus roundabout control using an instrumented vehicle, World Conference on Transportation Research, Shanghai-Çin, 1597-1609, 10-15 Temmuz, 2016.
20. Kubiková S. S., Kalašová A., Čulík K., Palúch J., Comparison of Traffic Flow Characteristics of Signal Controlled Intersection and Turbo Roundabout, Archivum Motoryzacji, 88 (2), 19-36, 2020.
21. Shaaban K., Abou-Senna H., Elnashar D., Radwan E., Assessing the impact of converting roundabouts to traffic signals on vehicle emissions

- along an urban arterial corridor in Qatar, *Journal of the Air & Waste Management Association*, 69 (2), 178-191, 2019.
22. Rizzo S.G., Vantini G., Chawla S., Reinforcement Learning with Explainability for Traffic Signal Control, 2019 IEEE Intelligent Transportation Systems Conference, Auckland-Yeni Zelanda, 3567-3572, 27-30 Ekim, 2019.
 23. Paul A., Mitra S., Deep Reinforcement Learning based Traffic Signal optimization for Multiple Intersections in ITS, 2020 IEEE International Conference on Advanced Networks and Telecommunication Systems, New Delhi-Hindistan, 1-6, 14-17 Aralık, 2020.
 24. Qadri S.S.S.M., Gökçe M.A., Öner E., Gökçe E.G., Analysis of Various Scenarios to Mitigate Congestion at a Signalized Roundabout using Microsimulation, 2019 Innovations in Intelligent Systems and Applications Conference, İzmir-Türkiye, 1-6, 31 Ekim-2 Kasım, 2019.
 25. Kunjir M., Chawla S., Chandrasekar S., Jay D., Ravindran B., Optimizing Traffic Control with Model-Based Learning: A Pessimistic Approach to Data-Efficient Policy Inference, KDD '23: The 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Long Beach-CA-A.B.D., 1176-1187, 6-10 Ağustos, 2023.
 26. Makri S., Charalambous P., Navigating a road network using Reinforcement Learning, 2022 International Conference on Interactive Media, Smart Systems and Emerging Technologies, Limassol-Kıbrıs, 1-8, 4-7 Ekim, 2022.
 27. Almansoori B.A., Almansoori S.S., Almansoori H., Almansoori R.A., Ahmed I., Shahid K., AI-Based Adaptive Signaling for Traffic Control Around Roundabouts, 2022 Advances in Science and Engineering Technology International Conferences, Dubai-B.A.E, 1-5, 21-24 Şubat, 2022.
 28. Baumgart U., Burger M., A Reinforcement Learning Approach for Traffic Control, 7th International Conference on Vehicle Technology and Intelligent Transport Systems, Prag-Çek Cumhuriyeti, 133-141, 28-30 Nisan, 2021.
 29. Kunjir M., Chawla S., Offline Reinforcement Learning for Road Traffic Control, arXiv 2201.02381, 2022.
 30. Baumgart U., Burger M., Optimal Control of Traffic Flow Based on Reinforcement Learning, *Communications in Computer and Information Science*, Cilt 1612, Springer, Cham, Almanya, 313-329, 2022.
 31. Yuan H., Li P., van Arem B., Kang L., Dong Y., Safe, Efficient, Comfort, and Energy-saving Automated Driving through Roundabout Based on Deep Reinforcement Learning, 26th IEEE International Conference on Intelligent Transportation Systems, Bilbao-İspanya, 6074-6079, 24-28 Eylül, 2023.
 32. Wang Z., Liu X., Wu Z., Design of Unsignalized Roundabouts Driving Policy of Autonomous Vehicles Using Deep Reinforcement Learning, *World Electric Vehicle Journal*, 14 (2), 52, 2023.
 33. Dutta S., Reinforcement Learning with TensorFlow, Packt Publishing, Birmingham, İngiltere 2018.
 34. Sutton R.S., Barto A.G., Reinforcement Learning: An Introduction, Cilt 2, MIT Press, Cambridge, MA, A.B.D., 2018.
 35. İstanbul Büyükşehir Belediyesi. İBB Açık Veri Portalı. <https://data.ibb.gov.tr/dataset/>. Yayın tarihi 2020. Güncellenme tarihi Ocak 2024. Erişim tarihi Ocak 4, 2024.
 36. Dong Y., Huang H., Zhang G., Jieling J., Adaptive Transit Signal Priority Control for Traffic Safety and Efficiency Optimization: A Multi-Objective Deep Reinforcement Learning Framework, *Mathematics*, 12 (24), 3994, 2024.
 37. Agrahari A., Dhabu M.M., Deshpande P.S., Tiwari A., Baig M. A., Sawarkar A. D., Artificial Intelligence-Based Adaptive Traffic Signal Control System: A Comprehensive Review, *Electronics*, 13 (19), 3875, 2024.
 38. Karbasi A.H., Yang H., Razavi S., Exploring the impact of traffic signal control and connected and automated vehicles on intersections safety: A deep reinforcement learning approach, arXiv 2405.19236, 2024.
 39. Xiao C., Wang D., Tang X., Pan J., Ma Y., Optimizing Efficiency of Mixed Traffic through Reinforcement Learning: A Topology-Independent Approach and Benchmark, 2025 IEEE International Conference on Robotics and Automation, Atlanta, GA-A.B.D., 3452-3458, 19-23 Mayıs, 2025.
 40. Usanmaz K., Güney S., Real time classification of traffic sign with deep learning methods, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 40 (2), 1311-1324, 2025.
 41. Turan E., Dandıl B., Avcı E., A new traffic signaling model based on graph and deep reinforcement learning, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 40 (1), 85-102, 2024.