

**T.C.
ERCIYES ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
HARİTA MÜHENDİSLİĞİ ANABİLİM DALI**

**SEZGİSEL ARAMA ALGORİTMALARI İLE
TRAFİK SİNYAL OPTİMİZASYONU**

**Hazırlayan
Abdullah KARAAĞAÇ**

**Danışman
Doç. Dr. Bülent BOSTANCI**

Doktora Tezi

**Mart 2020
KAYSERİ**

**T.C.
ERCIYES ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
HARİTA MÜHENDİSLİĞİ ANABİLİM DALI**

**SEZGİSEL ARAMA ALGORİTMALARI İLE
TRAFİK SİNYAL OPTİMİZASYONU**

(Doktora Tezi)

**Hazırlayan
Abdullah KARAAĞAÇ**

**Danışman
Doç. Dr. Bülent BOSTANCI**

**Bu çalışma, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK)
tarafından 2211-C kodlu Öncelikli Alanlara Yönelik Doktora Bursu ile
desteklenmiştir.**

**Mart 2020
KAYSERİ**

BİLİMSEL ETİĞE UYGUNLUK

Bu çalışmadaki tüm bilgilerin, akademik ve etik kurallara uygun bir şekilde elde edildiğini beyan ederim. Aynı zamanda bu kural ve davranışların gerektirdiği gibi, bu çalışmanın özünde olmayan tüm materyal ve sonuçları tam olarak aktardığımı ve referans gösterdiğimi belirtirim.



Abdullah KARAAĞAÇ

“Sezgisel Arama Algoritmaları ile Trafik Sinyal Optimizasyonu” adlı Doktora tezi, Erciyes Üniversitesi Lisansüstü Tez Önerisi ve Tez Yazma Yönergesi’ ne uygun olarak hazırlanmıştır.

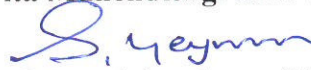


Hazırlayan
Abdullah KARAAĞAÇ



Danışman
Doç. Dr. Bülent BOSTANCI

Harita Mühendisliği ABD Başkanı



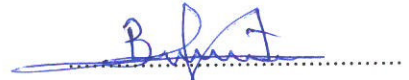
Prof. Dr. Abdurrahman EYMEN

Doç. Dr. Bülent BOSTANCI danışmanlığında **Abdullah KARAAĞAÇ** tarafından hazırlanan “**Sezgisel Arama Algoritmaları ile Trafik Sinyal Optimizasyonu**” adlı bu çalışma jürimiz tarafından Erciyes Üniversitesi Fen Bilimleri Enstitüsü **Harita Mühendisliği** Anabilim Dalında **Doktora** tezi olarak kabul edilmiştir.

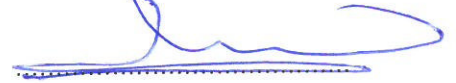
13 / 03 / 2020

JÜRİ:

Danışman : Doç. Dr. Bülent BOSTANCI



Üye : Prof. Dr. Mustafa ILICALI



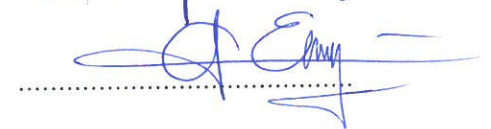
Üye : Prof. Dr. Erkan BEŞDOK



Üye : Prof. Dr. Alper BAŞTÜRK



Üye : Dr. Ö. Üyesi Ahmet Emin KARKINLI



ONAY:

Bu tezin kabulü Enstitü Yönetim Kurulunun 24/03/2020 tarih ve 2020/22-91 sayılı kararı ile onaylanmıştır.



Prof. Dr. Mehmet AKKURT

Enstitü Müdürü V.

TEŞEKKÜR

Bana çalışmalarım süresince her türlü yardımı ve fedakârlığı sağlayan, başta sevgili annem ve babama olmak üzere danışmanım sayın Doç. Dr. Bülent BOSTANCI'ya, Tez İzleme Komitemde bulunan Prof. Dr. Erkan BEŞDOK ve Doç. Dr. Alper Baştürk'e, Harita Mühendisliği Bölümü öğretim elemanlarına, beni sürekli destekleyen başta Arş. Gör. Talha TAŞKANAT olmak üzere tüm araştırma görevlilerine teşekkürlerimi sunarım.

Hepsinden öte, beni sabırla destekleyen ve tüm sıkıntılara şahit olan sevgili eşim Zehra ve oğlum Elyesa'ya teşekkürü bir borç bilirim.

Abdullah KARAAĞAÇ

Mart 2020, KAYSERİ

SEZGİSEL ARAMA ALGORİTMALARI İLE TRAFİK SİNYAL OPTİMİZASYONU

Abdullah KARAAĞAÇ

**Erciyes Üniversitesi, Fen Bilimleri Enstitüsü
Doktora Tezi, Mart 2020
Danışman: Doç. Dr. Bülent BOSTANCI**

ÖZET

Coğrafi Bilgi Sistemleri (CBS) uygulamaları dünya üzerindeki doğal ve yapay kaynakların verimli kullanımı için grafik ve grafik olmayan verilerin bir arada kullanılması ile sürekli olarak geliştirilmektedir. Bu uygulamaların geliştirilmesinde geomatik mühendislerinin katkısı çok önemli bir yer teşkil etmektedir. Son yıllarda CBS alanı disiplinler-arası bir hal almış ve özellikle kentsel alanlarda akıllı şehir uygulamaları adı altında anılmaya başlanmıştır. Akıllı şehir uygulamaları ise operasyonel manada çeşitli alt dallara ayrılmıştır.

Bu tez çalışmasında akıllı şehirlerin alt dallarından olan akıllı hareketlilik alanında, bir şehir trafiği ağında, sezgisel arama algoritmaları kullanılarak trafik sinyal optimizasyonu yapılmıştır. Çalışmada literatürde sıkça kullanılan Genetik Algoritma (GA) ve Standart Diferansiyel Gelişim Algoritması (sDE)'nin yanında literatüre yeni kazandırılan Bernstein Arama Diferansiyel Gelişim Algoritması (BSD) ile birkaç kavşaktan oluşan trafik ağlarındaki sinyal sürelerinin aynı anda optimize edilmesi amaçlanmaktadır.

BSD algoritması yeni ve kolay kontrol edilebilir, basit yapılandırılmış, özyinelemeli olmayan, yüksek verimli, hızlı ve pratikte parametresi olmayan bir evrensel diferansiyel gelişim algoritmasıdır. BSD çok elverişli rasgele çaprazlama ve mutasyon süreçlerine sahiptir ve standart diferansiyel gelişim (sDE) ve sDE'nin geliştirilmiş varyantlarının aksine bir kontrol parametresi ayar süreci bulunmamaktadır.

Çalışma alanı olarak; Kayseri merkezinde bulunan bir trafik ağı seçilmiştir. Çalışma alanına ait günün en yoğun saati olan sabah vaktinde ağa giriş yapan araçların sayı verileri Kayseri Büyükşehir Belediyesi aracılığı ile elde edilmiştir. Optimizasyon süreci iteratif yaklaşımlı olduğundan bu sürecin gerçek bir trafik ağında denenmesi mümkün değildir. Dolayısıyla trafikten elde edilen statik olan; yol geometrileri, sinyal lambalarının konumları gibi verilerin yanında, dinamik olan; araç verileri gerçek zamanlı simülasyon ortamında modellenmiştir.

Halihazırda piyasada pek çok trafik simülasyon yazılımı mevcuttur. Bu yazılımların gerçek trafiğe ait statik ve dinamik verileri doğru modelleyebilmeleri simülasyonun başarısı ile yakından ilişkilidir. Bu tez çalışmasında, günümüzde piyasa lideri konumunda bulunan VISSIM yazılımının akademik sürümü kullanılmıştır. VISSIM yazılımı pek çok açıdan çeşitli avantajlar sağlamaktadır. Bu avantajlardan ilki; harita altlıkları sayesinde statik verilerin program içerisinde doğrudan modellenmesine izin vermesidir. İkincisi; sürücü davranışları yazılım kapsamında çok başarılı şekilde modellenmiştir. Üçüncüsü; COM adı verilen eklentisi sayesinde pek çok programlama dili ile VISSIM yazılımındaki verilerin çekilmesi ve verilere müdahale edilmesi mümkün olmaktadır. Ek olarak VISSIM yazılımının geliştirme birimi tarafından geliştirilen, Vissig adı verilen istatistik temelli, kavşak bazlı ve kesin zamanlı bir trafik sinyal optimizasyon algoritması bulunmaktadır. Kayseri Büyük Şehir Belediyesi halihazırda çalışma alanı olarak seçilen bölgedeki trafik sinyal değerlerini Vissig algoritması ile optimize etmiş ve kullanıma açmıştır.

Bu tezde optimizasyon algoritmaları MATLAB programlama dili ile derlenmiştir. Optimizasyon algoritmaları ile üretilen sinyal sürelerinin doğrudan VISSIM yazılıma aktarılması için hazır bir VISSIM eklentisi bulunmamaktadır. Bu problemin çözümü için, yine MATLAB programlama dilinde bir sinyal çizelgeleme algoritması geliştirilmiştir. BSD, sDE, GA ve Vissig algoritması ile elde edilen sonuçlar birbirleri ile karşılaştırılmıştır.

Anahtar Kelimeler: Trafik Sinyal Optimizasyonu, Trafik Simülasyonu, Sezgisel Arama Algoritmaları, Bernstein Arama Diferansiyel Gelişim Algoritması

TRAFFIC SIGNAL OPTIMIZATION VIA HEURISTIC SEARCH ALGORITHMS

Abdullah KARAAĞAÇ

Erciyes University, Graduate School of Natural and Applied Sciences

PhD Thesis, Month Year

Supervisor: Assoc. Prof. Dr. Bülent BOSTANCI

ABSTRACT

Geographic Information Systems (GIS) applications are continuously developed by using graphic and non-graphic data together for efficient use of natural and artificial resources in the world. The contribution of geomatics engineers plays an important role in the development of these applications. In recent years, the field of GIS has become interdisciplinary and has become known as smart city applications especially in urban areas. Smart city applications are divided into several sub-branches in operational sense.

In this dissertation, traffic signal optimization has been performed in the smart mobility field, which is one of the sub-branches of smart cities, in a city traffic network by using evolutionary algorithms. In the study, besides Genetic Algorithm (GA) and Standard Differential Evolution Algorithm (sDE), which are frequently used in the literature, the Bernstein Search Differential Evolution Algorithm (BSD), which is newly introduced to the literature, aims to simultaneously optimize the signal times in traffic networks consisting of several intersections.

The BSD algorithm is a new and handy, simple structured, non-recursive, high efficiency, fast and practically universal parameter-free differential evolution algorithm. BSD has literally favorable random crossover and mutation processes and, unlike the standard differential evolution (sDE) and sDE's improved variants, there is no control parameter setting process.

As the study area; a traffic network in the center of Kayseri has been selected. The number data of the vehicles entering the network in the morning, which is the rush hour of the day, was obtained through Kayseri Metropolitan Municipality. Since the optimization process is iterative, it is not possible to try this process in a real traffic network. Therefore, alongside of the static traffic data; as well as such as path

geometries, locations of signal heads, dynamic data, as vehicle data has been obtained and modeled in a real-time simulation environment.

Currently, there are many pieces of traffic simulation software available on the market. The ability of these software to accurately model static and dynamic data of real traffic is closely related to the success of the simulation. In this thesis, the academic version of VISSIM software, which is the market leader today, has been used. VISSIM software offers several advantages in many aspects. The first of these advantages; It allows the modeling of static data directly in the program by means of map bases. Latter; driver behaviors are modeled very successfully within the software. Third; Thanks to its plugin called COM, it is possible to capture and interfere the data in the VISSIM software with many programming languages. In addition, there is a statistics-based, intersection-based and definite-time traffic signal optimization algorithm called Vissig developed by the development unit of the VISSIM software. Kayseri Metropolitan Municipality has optimized the traffic signal values in the region, which is currently selected as the study area, with the Vissig algorithm and put it into use.

In this thesis, optimization algorithms are compiled with MATLAB programming language. There is no ready-made VISSIM plug-in for transferring signal times generated by optimization algorithms directly to VISSIM software. To solve this problem, a signal scheduling algorithm has been developed in the MATLAB programming language. The results obtained with BSD, sDE, GA and Vissig algorithm were compared with each other.

Keywords: Traffic Signal Timing Optimization, Traffic Simulation, Heuristic Optimization Techniques, Bernstein Search Differential Evolution Algorithm

İÇİNDEKİLER

SEZGİSEL ARAMA ALGORİTMALARI İLE TRAFİK SİNYAL OPTİMİZASYONU

BİLİMSEL ETİĞE UYGUNLUK.....	ii
YÖNERGEYE UYGUNLUK.....	iii
KABUL VE ONAY	iv
TEŞEKKÜR	v
ÖZET	vi
ABSTRACT	viii
İÇİNDEKİLER.....	x
KISALTMALAR LİSTESİ.....	xiii
TABLO LİSTESİ.....	xiv
ŞEKİL LİSTESİ	xv
GİRİŞ	1

BÖLÜM 1

LİTERATÜR TARAMASI ve GENEL BİLGİLER

1.1. Literatür Taraması	6
1.1.1. Trafik Akış Kontrolü için Yapılan Çalışmalar	7
1.2. Trafik Dinamikleri	10
1.2.1. Trafik Ataması	12
1.2.2. Güzergâh Seçim Modelleri ve Dengeleri	13
1.2.3. Statik ve Dinamik Atama	14
1.3. Trafik Değişkenleri	15
1.3.1. Trafik Ağı Düzeyindeki Akış Değişkenleri.....	16
1.3.1.1. Link Bazlı Ölçümler	17
1.3.1.2. Güzergâh Bazlı Ölçümler	17
1.3.2. İki Linkli ve Serbest Yol Ağı.....	20
1.4. Trafik Sinyal Kontrolü İçin Temel Parametreler	23
1.4.1. Sinyal Fazı (Signal Phase)	23
1.4.2. Doygunluk (Saturation).....	24

1.4.3. Zaman Gecikmesi (Time Delay).....	24
1.4.4. Trafik Lamba Sayısı (Number of Traffic Stops).....	25
1.4.5. Trafik Kapasitesi (Traffic Capacity)	25
1.4.6. Optimum Sinyal Zamanı için Matematik Model	25
1.4.7. Ağırlık Katsayılarının Belirlenmesi.....	26
1.5. Trafik Sinyal Düzenleme Süreci	27
1.5.1. Sinyal Planlama Süreci	28
1.5.2. Sistem Kavşaklarının Tanımlanması.....	28
1.5.3. Mevcut verilerin toplanması ve düzenlenmesi	29
1.5.4. Saha Araştırması	30
1.5.5. Günlük senaryo sayısının belirlenmesi	31
1.5.6. Yerel zaman parametrelerinin hesaplanması.....	31
1.5.7. Sinyal gruplarının tanımlanması	31
1.5.8. Koordinasyon parametrelerinin hesaplanması	32
1.5.9. Yeni planın uygulaması ve değerlendirilmesi.....	33
1.6. Trafik Simulasyon Modelleri	33
1.6.1. Makroskobik Modeller.....	33
1.6.2. Mikroskobik Modeller	34
1.6.2.1. Araç Takip Modelleri	35
1.6.2.1.1. Gecikmeli Diferansiyel Denklemler.....	35
1.6.2.1.2. Hücresel Otomatlar ve Bağlantılı Haritalar	36
1.6.3. Mezokobik Modeller	39

BÖLÜM 2

YÖNTEM ve MATERYAL

2.1. Kullanılan Yazılımlar	41
2.1.1. VISSIM	41
2.1.1.1. Trafik Simülasyon Modeli.....	42
2.1.1.2. VISSIM ile ilgili Teknik Bilgiler	43
2.1.1.2.1. Rastgele Tohumlama (Random Seed)	43
2.1.1.2.2. Eriyod ve Simülasyon Çözünürlüğü.....	44
2.1.1.2.3. VISSIM’da Varsayılan Maksimum İvmelenme-Yavaşlama Eğrileri.....	45
2.1.1.2.3.1. Wiedemann Yaklaşımı	46
2.1.1.2.3.2. Wiedemann 74 Modeli Parametreleri	47

2.1.1.3. COM Arayüzü	48
2.1.1.4. VISSIM ile Otomatik Aşama Tabanlı Sabit Zamanlı Kontrolün Yeşil Sinyal Optimizasyonu (Vissig).....	49
2.2. Sezgisel Arama Algoritmaları.....	51
2.2.1. Genetik Algoritma (GA).....	54
2.2.1.1. Genetik Algoritmanın Adımları.....	54
2.2.2. Standart Diferansiyel Gelişim Algoritması (sDE)	57
2.2.2.1. sDE'nin adımları	57
2.2.3. Bernstein Arama Diferansiyel Gelişim Algoritması (Bernstrain-Search Differential Evolution Algorithm – BSD)	59
2.2.3.1. BSD Algoritmasının Adımları.....	60
2.2.3.2. Bernstein Polinomsalları	63

BÖLÜM 3

UYGULAMA ve BULGULAR

3.1. Çalışma Alanı ve Mevcut Durum	65
3.1.1. Kavşak Sayımları ve Mevcut Durum.....	66
3.1.1.1. K1 Kavşağı (30 Nolu Sanat Okulu Kavşağı).....	66
3.1.1.2. K2 Kavşağı (74 Nolu Şehitlik Kavşağı).....	69
3.1.1.3. K3 Kavşağı (80 Nolu Akansu Kavşağı).....	72
3.1.1.4. K4 Kavşağı (66 Nolu Battalgazi Kavşağı).....	75
3.2. Algoritmalar ile Performans Değerleme	77
3.2.1. Çalışma Alanının VISSIM Üzerinde Oluşturulması.....	78
3.2.2. Verilerin COM yardımı ile MATLAB Ortamına Aktarılması ve Simülasyon Yönetimi	79
3.2.3. Sezgisel Arama Algoritmaları ile Sinyal Değerlerinin Optimize Edilmesi. 81	
3.2.4. Değerlerin VISSIM Ortamına Aktarılması ve Simülasyon	82
3.3. Bulgular	86

BÖLÜM 4

SONUÇLAR VE TARTIŞMA

KAYNAKLAR.....	99
ÖZGEÇMİŞ.....	111

KISALTMALAR LİSTESİ

ABC	Yapay Arı Koloni Algoritması
ATŞ	Ağ Temel Şeması
BİT	Bilgi İletişim Teknolojileri
	Bernstein Arama Diferansiyel Gelişim
BSD	Algoritması
BVM	Başlangıç-Varış Matrisi
CBS	Coğrafi Bilgi Sistemleri
DE	Diferansiyel Gelişim Algoritması
DTA	Dinamik Trafik Ataması
GA	Genetik Algoritma
İGÇ	İleri Geri Çatma
JADE	Uyarlamalı Diferansiyel Evrim Algoritması
KW	Kinematik Dalga
MILP	Karışık Tam Sayılı Doğrusal Programlama
MTŞ	Makroskopik Temel Şeması
SAA	Sezgisel Arama Algoritması
SCOOT	Ayrık Döngülü Ofset Optimizasyon Tekniği
sDE	Standart Diferansiyel Gelişim Algoritması
SP	Stokastik Programlama
TUHE	Trafik Uzaysal Homojenite Endeksi
uDE	Evrensel Diferansiyel Gelişim Algoritması
YZ	Yapay Zeka

TABLO LİSTESİ

Tablo 1.	Ulaşım Planlaması Model Tablosu.....	11
Tablo 2.	K1 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları	69
Tablo 3.	K2 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları	71
Tablo 4.	K3 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları	74
Tablo 5.	K4 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları	77
Tablo 6.	Döngü süreleri için belirlenen alt ve üst sınırlar	82
Tablo 7.	Çizelgeleme matrisi	83
Tablo 8.	Algoritmalara ait en iyi performans değerleri ve çıkışa ulaşan araç sayıları.....	87
Tablo 9.	Algoritmaların kırılma yaşanan performans değerleri.....	88
Tablo 10.	Aşırı Doygun Durumda Algoritmaların Gösterdiği En İyi Performans Değerleri.....	92

ŞEKİL LİSTESİ

Şekil 1.	Bir Uzay-Zaman Diyagramındaki İki Boyutlu Güzergâhlar	18
Şekil 2.	Üç Boyutlu Uzay-Zaman Diyagramı	19
Şekil 3.	Üç Boyutlu Uzay-Zaman Diyagramının x-y Düzlemindeki Ağ Yapısı	19
Şekil 4.	(a) Serbest Trafik Ağı; (b) Linklerin Temel Diyagramı	20
Şekil 5.	Yoğunluk A-Yoğunluk B Faz Diyagramı (a) Ağın Ortalama Akışı; (b) Ağın Ortalama Yoğunluğu	21
Şekil 6.	(a) Ortalama Ağ Akışına karşılık gelen Ortalama Ağ Yoğunluğu; (b) Ağ Yoğunluğunun Standart Sapmasını Gösteren Yoğunluk A-Yoğunluk B Faz Diyagramı	22
Şekil 7.	Sinyal Faz Planları ve Döngüleri	23
Şekil 8.	Trafik Simülatörü ve Sinyal Durumu Üretici Arasındaki İletişim	42
Şekil 9.	VISSIM-COM Modeli Nesne Hiyerarşisi	49
Şekil 10.	Genetik Algoritmaya Ait Akış Diyagramı	55
Şekil 11.	İki Boyutlu Uzayda Yerel ve Global Çözümler	56
Şekil 12.	sDE'ye Ait Akış Diyagramı	58
Şekil 13.	BSD'ye ait Kod Mimarisi	63
Şekil 14.	2. dereceden Bernstein Polinomsalları	64
Şekil 15.	Ölçüm yapılan 4 adet kavşak	65
Şekil 16.	K1 kavşağının cadde görünümü	66
Şekil 17.	K1 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri	67
Şekil 18.	K1 kavşağına ait faz diyagramı	67
Şekil 19.	K1 kavşağına ait sinyal grup diyagramı	68
Şekil 20.	K2 kavşağının cadde görünümü	70
Şekil 21.	K2 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri	70
Şekil 22.	K2 kavşağına ait faz diyagramı	71
Şekil 23.	K2 kavşağına ait sinyal grup diyagramı	71
Şekil 24.	K3 kavşağının cadde görünümü	72
Şekil 25.	K3 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri	73
Şekil 26.	K3 kavşağına ait faz diyagramı	73
Şekil 27.	K3 kavşağına ait sinyal grup diyagramı	74
Şekil 28.	K4 kavşağının cadde görünümü	75
Şekil 29.	K4 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri	76

Şekil 30. K4 kavşağına ait faz diyagramı	76
Şekil 31. K4 kavşağına ait sinyal grup diyagramı	77
Şekil 32. Çalışma alanına ait VISSIM modeli	78
Şekil 33. Uygulamanın akış diyagramı	79
Şekil 34. VISSIM Simülasyon Ortamından bir Görüntü	86
Şekil 35. Algoritmaların iterasyon sayılarına göre başarı oranları	87
Şekil 36. BSD algoritması ile üretilen K1 kavşağına ait yeni sinyal grup diyagramı ...	90
Şekil 37. BSD algoritması ile üretilen K2 kavşağına ait yeni sinyal grup diyagramı ...	91
Şekil 38. BSD algoritması ile üretilen K3 kavşağına ait yeni sinyal grup diyagramı ...	91
Şekil 39. BSD algoritması ile üretilen K4 kavşağına ait yeni sinyal grup diyagramı ...	92
Şekil 40. BSD algoritmasının aşırı doygun durumda ürettiği K1 kavşağına ait yeni sinyal grup diyagramı	93
Şekil 41. BSD algoritmasının aşırı doygun durumda ürettiği K2 kavşağına ait yeni sinyal grup diyagramı	93
Şekil 42. BSD algoritmasının aşırı doygun durumda ürettiği K3 kavşağına ait yeni sinyal grup diyagramı	94
Şekil 43. BSD algoritmasının aşırı doygun durumda ürettiği K4 kavşağına ait yeni sinyal grup diyagramı	94

GİRİŞ

Özellikle son yüzyılda gelişen teknoloji sayesinde, insanlar ağır sanayi ve limanların olduğu bölgelere hızla göç etmiştir. Demografik ve ekonomik nedenlerden ötürü göç eden çoğu insana ait konutların bu sanayi ve ticaret bölgelerinin çevresinde olması, şehirlerin hızla büyümesine sebep olmuş, böylelikle şehirlerin yönetilmesi ciddi bir problem haline almıştır. Bu problemlerin başında ulaşım, enerji, sağlık, barınma, temizlik ve güvenlik gibi etkenler bulunmaktadır.

Son 15 yılda ise, kentsel alanlarda ortaya çıkan ihtiyaç ve zorluklara cevap olarak, kaynakların verimli kullanılması amacı ile “Akıllı Şehirler” kavramı ortaya çıkmış ve yönetimlerin dikkat odağı haline gelmiştir. Bilgi ve İletişim Teknolojileri (BİT) potansiyelinin de artması ile dünyanın dört bir yanındaki şehirler, ekonomik rekabetçiliği, sürdürülebilirliği, sosyal ve sermaye çekiciliğini ve en önemlisi de herkes için yaşam kalitesini artırmak için çeşitli stratejiler benimsemiştir. Şehirlerin daha akıllı hale gelmesi için geniş bir strateji yelpazesi bulunmaktadır, ancak şehrin ihtiyaçlarına ve isteklerine en uygun stratejiyi belirlemek zor bir süreçtir. Bu stratejik kararlar genellikle kıyas ve muhakeme edilir. Ancak doğru dengenin belirlenmesi için şehirlerin güçlü ve zayıf yönlerinin tam olarak bilinmesi gereklidir.

Akıllı şehir kavramı daha doğru planlama yapılabilmesi için; akıllı ekonomi, akıllı yönetim, akıllı çevre, akıllı yaşam, akıllı insanlar ve akıllı hareketlilik olmak üzere altı alt dala ayrılmıştır (Alizadeh vd., 2018). Bu tez çalışması kapsamında ise akıllı hareketlilik bileşenine odaklanılmaktadır.

Teknolojik yenilik ve stratejilerle geliştirilmeye devam eden modern ve sürdürülebilir ulaştırma sistemlerinin temel amacı, bir bölgede yer alan insan ve ticari malların, kendi bölgeleri veya dünyanın geri kalanıyla bağlantı kurabilmelerini sağlamaktır. Gelecek vaat eden teknolojiye dayalı stratejiler göz önüne alındığında da teknolojinin yalnızca

halihazırdaki şehirleri kapsayacağını unutmamak gerekir. Akıllı şehirlerin, daha geniş bağlamında akıllı hareketliliğin geliştirilmesi, akıllı şehir gelişiminin diğer boyutları ile olan ilişki ve sinerjisinin iyi bir şekilde anlaşılmasına dayanmaktadır.

İnsanlar her geçen gün gürültü, kirlilik, kazalar ve sıkışıklık gibi trafiğin olumsuz etkilerine katlanamaz hale gelmektedir. Bu nedenle, karar verme organları, artan ulaşım talebini yönetmek için sürekli yeni stratejiler aramayı sürdürmüşlerdir. Özellikle büyük şehirlerdeki nüfus yoğunluğu nedeniyle, artan ulaştırma talebin karşılanması, sadece yeni bir altyapı inşa edilerek çözülemeyeceği, Braess'in paradoksu ile açıklanmış; yeni yol açarak trafik talebinin çözülmeye çalışılmasının ters sezgisel ve hatta ters üretken sonuçlara neden olacağı net bir şekilde ortaya konmuştur (Rapoport vd., 2009).

Trafik sıkışıklığı yeni bir sorun değildir. On yıllardır trafik akışının modellenmesi ve tıkanıklığın azaltılması araştırmacılar için çekici bir konu olmuştur. Trafik modelleme araştırması geleneksel olarak, mikroskobik, makroskobik ve mezoskobik olarak üç gruba ayrılmaktadır. Mikroskobik modeller, tek tek araçların (veya daha genel olarak parçacıkların) diğer araçlarla etkileşimli bir ortamda davranışını incelemektedir. Yaygın olarak bilinen mikroskobik model örnekleri şerit değiştirme ve araç takip eden modellerdir (Herman ve Potts, 1959). Makroskobik modeller, araçların ortak dinamiklerini yani yoğunluk (araç / mesafe), akış (araç / saat) ve hız (toplam / saat) gibi değişkenleri toplam değişkenler olarak tanımlamaktadır. Makroskobik modellerin iyi bilinen bir örneği Lighthill-Whitham-Richards (LWR) modelidir (Lighthill ve Whitham, 1955), (Richards, 1956). Mezoskobik modeller ise bireysel taşıtların dinamiğini, toplam (makroskobik) ilişkilerin bir fonksiyonu olarak tanımlamaktadır. Mezoskobik modellere örnek olarak Prigogine'nin 1961'de ve Prigogine ve Herman'nın 1971'de yaptığı çalışmaları verilebilir (Prigogine, 1961)(Prigogine ve Herman, 1971).

Son yıllarda daha hızlı bilgisayarlar ve gelişmiş trafik simülasyon modelleri, büyük şehirlerdeki otoyol ağı gibi büyük ağların gerçek zamanlı simülasyonlarını mümkün kılmıştır. Bu simülasyonlar karar vericilere mevcut farklı seçeneklerin etkilerini değerlendirme konusunda yardımcı olabilmektedir. Böylelikle ağ genelinde ortalama akış, ortalama yoğunluk ve ortalama hız arasındaki; tekrarlanabilir ve iyi tanımlanmış bir ilişkinin varlığı literatürde belirlenmiştir (Geroliminis ve Daganzo, 2008; Herman ve Prigogine, 1979; Mahmassani vd., 1987, 1984; Mahmassani ve Peeta, 1993; Williams

vd., 1995, 1987). Bu ağ çapındaki ilişki daha sonraki zamanlarda “Makroskobik Temel Şeması” (MTŞ) (Geroliminis ve Daganzo, 2008) veya “Ağ Temel Şeması” (ATŞ) olarak adlandırılmıştır (Saber ve Mahmassani, 2012).

Trafik sıkışıklığı, faz geçişleri ve dur-kalk dalgalarının dinamikleri de dahil olmak üzere trafik akışı teorisi ve özellikleri, kısmi trafik akışının kümülatif fiziksel sistemlerle çok benzerlikleri olduğu için, fizikçiler için de çekici araştırma konusu olmuştur (Helbing, 1997; Helbing ve Treiber, 1998; Kerner ve Rehborn, 1997, 1996a; Lee vd., 1998; Schreckenberg ve Wolf, 1998). Örneğin; araç akışı ile sürekli basınçla sıkıştırılabilir akışkan ya da gaz arasındaki bağıntıyı kurarak farklı trafik akış modelleri geliştirilmiştir (Helbing, 1996, 1995; Kerner ve Konhäuser, 1994; Lighthill ve Whitham, 1955). Ayrıca, büyüyen yüzeyler, moleküler ışın epitaksi, süper iletkenler ve biyokimyasal ağlar gibi birçok dinamik sisteme benzer şekilde, trafik akışında dengede olmayan faz geçişleri de ele alınmıştır (Helbing ve Treiber, 1998). Ayrıca ayırık otoyol tesisleri ve ağlarında tıkanık trafikte histerezis durumu gözlemlenmiştir (Buisson ve Ladier, 2009; Geroliminis ve Sun, 2011; Saber ve Mahmassani, 2012). Histerezis, sisteme verilen yeni girdinin, sistemin yakın geçmişteki durumundan ötürü verilen girdiye olan tepkisindeki gecikme olarak ifade edilebilir. Örneğin; trafikteki araçların yeşil fazdaki beklenen akışı, fazın kırmızısından yeşile döndüğü anda gerçekleşmeyip zamanla fazın beklenen akışına yakınsayacaktır.

Ulaştırma talebinin yüksek bir zaman çözünürlüğü ile modellenmesi oldukça zordur. Mikroskobik trafik simülasyonları yapmak için ağ üzerinden yapılan seyahat bilgileri, yani bir dizi sürüş başlangıç zamanı ve güzergâhlar gibi birçok ayrıntılı girdi verisine ihtiyaç duyulmaktadır. Bu girdi verilerini sağlamak için ise bir ulaşım talebi veya güzergâh seçimi davranış modeli gereklidir.

Bununla birlikte güzergâh seçiminin modellenmesi, simülasyondan bağımsız olarak yapılamamaktadır. Çünkü güzergâh seçimi simülasyonun bir sonucu olan seyahat sürelerine bağlıdır. Güzergâh seçimi için kabul edilen en yaygın model Wardrop’un ilk prensibidir (Herman ve Prigogine, 1979)(Wardrop, 1952): Her sürücü maliyetin (genellikle seyahat süresinin) minimum olduğu bir güzergâh seçer. Her güzergâh seçiminin bu koşulu yerine getirdiği durum, *kullanıcı dengesi* olarak adlandırılır.

Ulaşım talebi, genellikle talebi içeren bir Başlangıç-Varış Matrisi (BVM) vasıtasıyla sağlanır ve bu matriste her bir başlangıç-varış ikilisine karşılık seyahat süresi bulunur. Kullanıcı dengesi güzergâh seçimlerini hesaplama problemine, *trafik atama problemi* denir. BVM'nin zamana bağımlılığını göz ardı ederek bu sorunu basitleştirmek ve trafik akışı ile seyahat süresi arasında zamandan bağımsız bir ilişki olduğunu varsayarak, statik bir trafik atama problemini çözmek hala yaygın bir uygulamadır. Gerçekte BVM zamana bağlıdır. Yoğun saatlerde, talep çoğu zaman trafik sıkışıklığının artmasına yol açan mevcut karayolunun kapasitesini aşmaktadır. Bu nedenle seyahat süresi sistemin yakın geçmişine, yani trafik sıkışıklığının mevcut uzunluklarına bağlıdır.

Birçok bilim insanı matematiksel programlama yöntemlerine dayanan dinamik kullanıcı dengesi problemi için modeller önermiştir. Bu modelleri çözme görevi *dinamik trafik ataması* olarak adlandırılır.

Zamana bağlı kullanıcı dengesini bulmanın etkili bir yolu da tekrarlı simülasyondur. Tekrarlı simülasyon temelde 3 ana aşamadan gerçekleşmektedir:

- Trafiğin hiç olmadığını varsayan ilk güzergâhlar seçilir.
- Simülasyon kullanılarak ağ yükü ve seyahat süreleri hesaplanır ve sürücülerin güzergâh seçenekleri güncellenir.
- Seyahat süreleri durağan olana kadar bu işlem yinelenir, yani yinelemenin sabit bir noktası bulunur.

Bu tezin temel amaçlarından birisi, bilgisayar ortamında gerçek trafik verileri kullanılarak doğru sonuçlar üreten trafik ataması modelinin geliştirilmesidir. Bu yaklaşım modeli tekrarlı simülasyonlara dayandığından, modele *simülasyon tabanlı trafik ataması* adı verilmektedir.

Bu tezin asıl amacı ise sezgisel arama algoritmalarının trafik sinyal optimizasyonuna başarılı şekilde uygulanabilirliğini göstermektir. Bu amaçla literatürde daha önce uygulanmamış olan Bernstein Arama Diferansiyel Gelişim Algoritması'nın (BSD) trafik sinyal optimizasyonda kullanılmıştır. Performansın ölçeklenebilmesi adına optimizasyonlar yine sezgisel sürece dayanan Genetik Algoritma ve Standart

Diferansiyel Gelişim Algoritmasının yanında, istatistik temelli olan ve VISSIM programının bir eklentisi olan Vissig Algoritmasının performansı ile kıyaslanmıştır.

Tez kapsamında Kayseri’de bulunan 4 kavşaktan oluşan bir trafik ağı modellenip, gerçek trafik ölçüm değerlerinin yanında trafiğin aşırı doymun olduğu varsayılan durumunun verileri kullanılarak simülasyon tabanlı trafik ataması yapılması amacı ile trafik sinyal süresi optimizasyonunda sezgisel arama algoritmalarının kullanımı konusundaki teori ve uygulama aşamaları hakkında bilgi verilecektir. Bölüm 1’de trafik sistemlerinin gelişimi hakkındaki teorik altyapı ve literatürden bahsedilmiştir. Bölüm 2’de projede kullanılan yazılımlar ve MATLAB dili kullanılarak geliştirilen sezgisel arama algoritmaların VISSIM programında nasıl kullanılabileceği hakkında bilgiler verilmiştir. Bölüm 3’te ise yapılan uygulama ve bulgular hakkında ayrıntılı bilgilendirme yapılmıştır. Bölüm 4’te ise tez kapsamında çıkarılan sonuçlardan bahsedilmiş ve önerilerde bulunulmuştur.

Yapılan simülasyonlar sonucunda trafik sinyal optimizasyonu için en başarılı tekniğin BSD algoritması olduğu gözlenmiştir. Tekniklerin başarıları ise sırasıyla BSD, Vissig, sDE ve GA şeklinde gözlemlenmiştir.

BÖLÜM 1

LİTERATÜR TARAMASI ve GENEL BİLGİLER

1.1. Literatür Taraması

Geçmişten günümüze kadar yapılan çalışmalarda makroskobik otoyol trafik akış modellerinin büyük çoğunluğu, şeritlerde toplanan tüm değişkenleri göz önünde bulundurmakta (Treiber ve Kesting, 2013), sınırlı sayıda çalışmada ise, bir otoyoldaki farklı şeritleri bağımsız varlıklar olarak ele almaktadır. Bu konuyu inceleyen ilk önemli eserlerden birinde Gazis vd. (1962), çok şeritli otoyoldaki şerit yoğunluklarının bir denge yoğunluğu etrafında salındığını varsaymışlardır ve sistemin hatalarını en aza indirmek ve istikrarını artırmak üzere bir yöntem geliştirmişlerdir (Gazis vd., 1962). Sonrasında Michalopoulos vd. (1984), şerit değiştirme davranışını yakalamak için üç model önermiştir. İlk model araçların yoğunluklarının denge değerlerinden sapmalarının farkına göre araçların şerit değiştirdiği varsayımına dayanan bir sürekli modeldir. İkinci model ise hız ve ivmelenme etkilerini göz önünde bulundurup, ilk modelin genişletilerek ikinci dereceden bir model haline getirilmesidir. Sokak genişlikleri de projeye dahil edilerek üçüncü bir uzatma model ortaya konmuştur. Bu modeller herhangi bir ayrıştırma şeması uygulamadan sürekli bir uzay-zaman alanı içinde formüle edilmiştir (Michalopoulos vd., 1984). Laval ve Daganzo (2006). Kinematik dalga (KW) teorisini kullanmışlar ve hareketli tıkanıklıkları, parçacıklar olarak ele alan ve karakterize edilen daha karmaşık bir modelin ilk modülü olarak çok şeritli KW tabanlı bir model geliştirmişlerdir (Laval ve Daganzo, 2006). Modelde sınırlı hızlanma oranları ile şerit değişiklikleri, iki komşu şerit arasındaki ortalama hız farkına göre değerlendirilmektedir. Jin (2013) tarafından LWR teorisine (Lighthill ve Whitham, 1955) dayanan çoklu emtia modelinin geliştirildiği bir yaklaşım ortaya konmuştur. Çalışmada; şerit değiştiren akış, şerit değiştiren yoğunluk (hız-yoğunluk ilişkisini etkileyen yeni bir değişken olarak tanımlanmaktadır) kavramına, yeni bir temel

diyagramın (fundamental diagram) tanımlanması ve girişin “sözde entropi durumu” tanımı ortaya çıkmıştır (Jin, 2013). Bir diğer yaklaşım ise, Helbing’in (1997) çalışmasında da olduğu gibi gaz-kinetik trafik akış modeline dayanır. Burada; etkileşimli şeritlerin makroskobik trafik akışı denklemleri, kuyruklama etkileri açıkça hesaba katılarak türetilmiştir ve trafik sürekli (trafik akış değişkenlerindeki düzgün değişiklikler) ve süreksiz süreçler (araçların yavaşlaması ve şerit değiştirme) tarafından yönetilen bir araç topluluğu olarak tanımlanmıştır (Helbing, 1997; Hoogendoorn ve Bovy, 1999). Bazı yeni çalışmalar şerit akış dağılımının, ağın belirli yerlerinde modellenmesi sorununu ele almaktadır. Knoop vd. (2010), serbest akış ve yoğun rejim sırasında toplam yoğunluk ve şerit yoğunlukları arasındaki bir ilişki üzerinde çalışmıştır (Knoop vd., 2010). Yine Knoop vd. (2012), insandaki psikolojik dürtülerin şerit değiştirme arzusuna etkisini incelemiş ve şerit değiştirmeye yatkın olmanın kritik durumunun link başlangıçlarına ve hedefe götürecek şeritlerin yoğunluğuna bağlı olduğunu tespit etmişlerdir (Knoop vd., 2012). Sonrasında Duret vd. (2012)(Duret vd., 2012), trafik akışının şerit dağılımını göz önüne alarak basit bir doğrusal modeli türeten, üç şeritli otoyol için gerçek verileri (serbest akış koşullarında) analiz etmişlerdir.

1.1.1. Trafik Akış Kontrolü için Yapılan Çalışmalar

Bu alanla ilgili yapılan çalışmalar, gerçek zamanlı trafik dinamiklerini ve tahmin edilen trafik akışlarını ele almak için trafik kontrol şemaları geliştirmiştir. SCOOT (Hunt vd., 1982) ve SCATS (Luk, 1984), yaygın şekilde uygulanan iki önemli gerçek zamanlı ve uyarlanabilir trafik kontrol sistemidir. SCOOT (Split Cycle Offset Optimization Technique), çevrim içi trafik gözetim bilgisine göre sinyal ayarlarını aşamalı olarak ayarlayan, TRANSYT (Robertson, 1969) ise trafik sinyali optimizasyon modelinin kullanımına dayanan bir çevrimdışı optimizasyon çözümüdür. SCATS (Sydney Koordinatlı Uyarlanabilir Trafik Sistemi), Avustralya’da, karayolu ağındaki araç durma ve gecikmelerini en aza indirmek veya bir karayolunun verimini artırmak için geliştirilmiş bilgisayarlı bir sinyal kontrol sistemidir.

Buna ek olarak, DYPIC (Robertson ve Bretherton, 1974), OPAC (Gardner, 1983), PROLYN (Henry vd., 1983), SPPORT (Yagar ve Han, 1994) ve UTOPIA (Mauro ve DiTaronto, 1990), sistemleri de gerçek zamanlı olarak uyarlanabilir trafik sinyali kontrol sistemleridir (Cai vd., 2009; Haijema ve Hendrix, 2014; Zheng ve Recker,

2013). Ayrıca, trafik kontrol sisteminin performansını artırmak için trafik sinyallerini yönlendirme (Teklu vd., 2007), trafik kontrolünü yüksek çözünürlüklü olay temelli verilerle optimize etme (Hu ve Liu, 2013), ağ çapında trafik denetleyicisi (Lin vd., 2012), çok amaçlı trafik sinyali optimizasyonu (Stevanovic vd., 2015) ve yay akışına dayalı ağ trafiği kontrol optimizasyonu (Wey ve Jayakrishnan, 1997) gibi çeşitli yaklaşımlar ortaya konmuştur. Bunların arasında Wey ve Jayakrishnan (Wey ve Jayakrishnan, 1997), Robertson'un platoon dağılım simülasyonunun (Robertson, 1969) analitiğini temel alarak, zamansal olarak genişletilmiş bir ağdaki ana arter akışlarını kullanan bir trafik ağı sinyal optimizasyonu geliştirip formüle etmişlerdir. Park (Park, 2009), Wey'in modelini yol kenarı kablosuz cihazlar tarafından yazılmış araç içi veri tabanı çerezlerine dayanan ve kalıcı trafik çerezleri (Marca vd., 2006) adlı yeni bir sensör veri teknolojisini içerecek şekilde genişletmiş ve ayrıca geliştirdiği Güney Kaliforniya'daki bir ağ üzerinde simülasyon çalışmaları için dinamik programlama (DP) çözüm algoritmasını uygulamıştır.

Trafik tıkanıklığı durumunda, kavşak optimizasyon modelinin bir çözümü olarak, Park vd. tarafından problemi çözmek için Genetik Algoritma (GA) kullanılmıştır (Park vd., 2000). GA ile trafiğin aktığı durumlarda optimizasyon ile ilk kontrol sinyalleri üretilip sonraki iterasyonlarda gelişim sağlanmıştır (Park vd., 1999). Buna ek olarak, GA ayrıca ağ gecikmesi ve yol kapasitesini de dikkate almaktadır. Bununla birlikte, GA'daki optimizasyon işlemi, trafiğin durumunun belirli bir süre boyunca sabit olduğu varsayımını gerektirmektedir. Lieberman ve Chang (Lieberman ve Chang, 2005) her kavşağın kuyruk uzunluğunu kısıt olarak kullanır ve bir sinyal döngüsünde, yeşil yanma süresini doğrusal olmayan kuadratik programlama ile ayarlayabilen optimum sinyal faz diagram eğrisini çözmek için Karışık Tamsayılı Doğrusal Programlamayı (KTDP) önermişlerdir. Abu-Lebdeh ve Benekohal (Abu-Lebdeh ve Benekohal, 2003) kuyruk uzunluğunu dinamik kontrol performansının negatif fayda fonksiyonuna dayalı olarak değerlendiren bir amaç fonksiyonu şeklinde kullanmıştır.

Literatürde, bir otoyola giren araç sayısını düzenlemek için Yapay Zekâ (YZ) kullanılan pek çok çalışma yapılmıştır. Örneğin rampa ölçüm cihazları, takviyeli öğrenme algoritmaları kullanılarak kontrol edilmiştir (Fares ve Gomaa, 2015, 2014). Benzer şekilde trafik akışının optimizasyonu, sıralı bir karar verme süreci olarak formüle edilebilmiştir. Bu yöntemler akışın optimize edilmesi için araçların yoğunluğunu kritik

yoğunluğa yakın tutmayı amaçlamaktadır. Rampa ölçüm cihazlarını Q-öğrenme ile kontrol etmek için bir başka yöntem Rezaee vd. (Rezaee vd., 2012) tarafından kullanılmıştır. Burada bir açılma rampası yakınında bir döngü detektörü ile üzerinden geçen araçların sayısı optimize edilmiştir. Davarynejad vd.'nin (Davarynejad vd., 2011) uyguladığı yaklaşımda, öğrenim süresince rampaların kuyruk uzunlukları hesaba katılarak benzer bir çalışma gerçekleştirilmiştir.

Yapay sinir ağları, trafik ışıklarında denetleyiciler oluşturmak için mevcut çalışmalarda uygulanmıştır (Spall ve Chin, 1994; Wei ve Zhang, 2002). Bu tarz çalışmalarda öğrenme için kullanılan yol ağları görece daha basittir ve talep hacminin sabit olduğu varsayılır. Yapay sinir ağları, karayollarının çıkış talebini tahmin etmek için de kullanılmıştır (Kwon ve Stephanedes, 1994).

Kentsel alanda trafik ışıklarının optimizasyonu tıkanıklığı azaltmak için de önemlidir. Kuyer vd. (Kuyer vd., 2008) trafik ışıklarının davranışlarını koordine eden akıllı ajanlar ile kurgulandığı takviyeli öğrenme yöntemini kullanarak kentsel alanda trafik ışıklarının koordinasyonunu irdlemiştir. Kendi içinde koordineli trafik lambalarının çok ajanlı sistem ile kontrolü modellenirken, çok sayıda kavşağa uygulanmasında özellikle zamana ve modele göre ölçeklemeler konusunda, sistem oldukça karmaşık hale gelmektedir. Abdoos vd. (Rezaee vd., 2012), sorunu çözmek için hiyerarşik bir kontrol yöntemi üzerine çalışmışlardır.

He ve Hou (He ve Hou, 2012), Başkan (Başkan, 2009) trafik sinyal optimizasyonu için karınca kolonisi algoritmasını kullanmışlardır.

Son zamanlarda, çalışmaların çoğunda aşırı doymamış kavşaklarda sinyal kontrolü için matematiksel programlama ve optimizasyon modelleri kullanılmaktadır. Abu-Lebdeh vd., trafiğin akış yönündeki birikimleri ve yeşil zaman kullanımı üzerindeki kısıtlamalarla verimliliği en üst düzeye çıkaran geniş çaplı bir optimizasyon modeli geliştirmiştir (Abu-Lebdeh ve Benekohal, 2000). Başka bir çalışmada kuyruk uzunluğu kontrolü için karışık tamsayı doğrusal programlama yaklaşımı geliştirilmiştir (Lieberman vd., 2000). Lo ve Chow (Lo ve Chow, 2004), hücre iletim modeline dayanılarak aşırı doygunluk durumu için dinamik kavşak sinyali kontrol optimizasyon yöntemi geliştirmişlerdir. Li ve Prevedouros (Li ve Prevedouros, 2004) izole edilmiş sinyaller için hibrid bir optimizasyon ve kural tabanlı aşırı doymamış kontrol

algoritması sunmuştur. Chang ve Lin (Chang ve Lin, 2000), Chang ve Sun (Chang ve Sun, 2004), aşırı doymuş koşulların tümü boyunca sinyal parametrelerini optimize etmek için ayrı bir dinamik model ve performans endeksi yaklaşımı geliştirmiştir. Liu vd. (Liu vd., 2008) aşırı doymamış sinyalizasyon kavşakları, normal otoyol darboğazları olarak hesaplama verimliliğini büyük ölçüde iyileştirmek için işlemiştir. Liu ve Chang (Liu ve Chang, 2011) kuyruk geri taşmalarının (queue spillback) dinamik etkileşimlerini yakalamak için ana arter bağlantılarındaki fiziksel sıranın evrimini şerit grubu ile modellemiştir. Hu vd. (Hu vd., 2013), aşırı doymuş kavşaklarda optimal akış elde etmek için basit İleri Geri Çatma (İGC) yöntemini önermişlerdir. Tong vd. (Tong vd., 2015), beklenen araç gecikmesini en aza indirgeyen uyarlamalı sinyal zaman planlamalarını gerçekleştirmek için bir Stokastik Programlama (SP) modeli ortaya koymuştur. Sun vd. (Sun vd., 2015) aşırı doymuş kavşak sistemi için bir yarı-optimal dağıtılmış Kuyruk Tabanlı Geri Bildirim Kontrol Stratejisi (KTGBKS (QUEUE)) geliştirmişlerdir. KTGBKS, gerçek trafikte her bir döngü üzerinde oluşan kuyruklanmalarda uygulanmakta ve çevrimdışı(simülasyon) çalışmalardan elde edilen sonuçların en uygun olanına göre yaklaşık olarak tahmin edilebilmektedir.

1.2. Trafik Dinamikleri

Trafik akışı dinamiklerinin matematiksel olarak ifade edilmesinin, ihtiyaçlara göre farklılık gösteren çeşitli yöntemleri bulunmaktadır. İlk olarak sürücülerin ve araçların ivmelenme özellikleri hakkında sabit detektörler tarafından elde edilen verilerin yanında, mobil cihaz kullanımının yaygınlaşmasıyla GPS, WiFi ve mobil cihazların sensörlerinin yardımı ile elde edilen makroskobik verilerle birlikte çok miktarda trafik akışı verisi mevcuttur. Doğa bilimleri alanındaki deneylere karşılık gelen ilgili ölçümler, matematiksel bir modellemenin temelini oluşturmaktadır (bkz. Tablo 1). Bir modelin tahmin değerlerini gerçek verilerle karşılaştırarak ve sistemin uyumlu olması için model parametre değerlerini değiştirerek, yeni bir uygulama için ön şart olan uygulamaya özel bir model üretilebilir.

Tablo 1. Ulaşım Planlaması Model Tablosu

Zaman Aralığı	İlgi Alanı	Model	Problem Örneği
<0,1 sn	Araç Dinamikleri	Alt-Mikroskobik	Motor ve Fren Kontrolü
1 sn	Trafik Akış Dinamikleri	Araç Takip Modelleri	Tepki Zamanı/Zaman Boşluğu
10 sn			İvmelenme ve Yavaşlama
1 dk		Makroskobik Modeller	Trafik Lamba Döngü Süreleri
10 dk			Dur-Kalk Dalgası
1 sa	Ulaşım Planlanma	Trafik Gereksinimine göre Güzergâh Ataması	Pik Saatler
1 gün			Dünlük Gereksinim Örüntüsü
1 yıl		Piramit Yaş İstatistikleri	Alyapı inşa/değişim
5 yıl			Sosyo-ekonomik Yapı
50 yıl			Demografik Değişim

İkincisi, trafik dinamikleri birçok araç ve sürücünün etkileşimini tanımlamaktadır. Teknik olarak sürücü araç birimleri olarak adlandırılan araçların ve sürücülerin etkileşimi, bireylerin detaylarına bağlı olmayan yeni kolektif etkilere yol açmaktadır. Örnek olarak, dur-kalk dalgalarının oluşumu sırasında meydana gelen trafik sıkışıklığının uzay-zamansal kalıpları verilebilir. Bu durum klasik fizikteki sıvı ve gazların termodinamik davranışlarına benzemektedir.

Son olarak sürüş dinamikleri bazı sınırlamalara tabidir. Örneğin, tıpkı kapalı kaplardaki sıvı ve gazların etkileşimi gibi sürücüler de tipik olarak çevresindeki diğer sürücülerle doğrudan etkileşime girerken bazı sınırlanmalara tabidirler. Sonuçta araçlar (sürücüler değil) klasik anlamda fiziksel nesneler oldukları için ivmeler ve yavaşlamalar da fiziksel olarak sınırlıdır.

Trafik Akış Dinamiklerinin Sınırlandırılması: Trafik akışı dinamikleri, yaklaşık bir saniye ile birkaç saat arasında değişen zaman ölçeklerini içerir. İnsan reaksiyon süreleri ve birbirini takip eden iki araç arasındaki zaman aralığı 1 saniye düzeyinde olduğundan frenleme ve hızlanma manevraları da tipik olarak birkaç saniye sürmektedir. Şehir trafiğinde, trafik sinyallerinin bir kırmızı-yeşil döngüsünün süresi 1 dakika iken otoyollarda trafik salınımlarının ve dur-kalk dalgalarının süresi 5 ile 20 dakika arasındadır. Son olarak, trafik akışı modelleri için dış değişken (model girişi) işlevi

gören trafik talebi, “yoğun saat” terimi ile gösterildiğinden dolayı bir saatlik zaman ölçeklerinde değişiklik gösterir.

Saatler ile yıllar arasında değişen daha uzun zaman ölçekleri ulaştırma planlamasının etki alanıdır. Bu etki alanı, demografik değişimin neden olduğu trafik talebindeki değişkenliklerin çok uzun süreli ölçeklerini içermektedir.

Ulaştırma planlaması ve trafik akışı dinamikleri birbirini tamamlamaktadır. Klasik ulaştırma planlaması dört aşamadan meydana gelmektedir:

- 1- Seyahatin oluşması
- 2- Seyahat dağılımı
- 3- Ulaşım türü
- 4- Güzergâh ataması

Klasik ulaştırma planının model çıktısı ve modern dinamik varyantları, çalışma yapılan ağın her linkindeki trafik talebidir (saatteki araç sayısı). Bu değişkenler trafik akışı simülasyonları için her adımda sınır koşulları şeklinde simülasyona haricen girdi olarak verilmelidir. (Treiber ve Kesting, 2013)

1.2.1. Trafik Ataması

Trafik atamasının odak noktası, trafik akışının gerçekçi bir tanımından ziyade güzergâh seçiminin matematiksel açıklamasıdır ve Wardrop’un güzergâh seçim ilkeleri gibi temel kavramları sağlar (Wardrop, 1952).

Bir trafik simülasyon modelindeki güzergâh seçeneklerini hesaplamak için başlangıç-variş matrisine (BVM) ve ağdaki seyahat süreleri hakkında da bilgiye ihtiyaç vardır. Diğer taraftan bu seyahat süreleri güzergâh seçimlerine de bağlıdır. Bu nedenle kendi içinde tutarlı olan, yani sonuçta ortaya çıkan seyahat süreleriyle tutarlı olan bir güzergâh seçimi belirlemek gerekmektedir. Aslında, BVM’nin dahi kendi kendine tutarlı bir şekilde seçilmesi gereklidir. Çünkü BVM belirli bir anlamda seyahat zamanlarının bir fonksiyonudur.

Bir trafik simülasyon modeli, olası tüm güzergâh seçimleri olan R 'den seçilen bir güzergâh seçimine bağlı seyahat zamanını $\tau \in \mathcal{T}$ 'ye bağımlılığını ifade eden bir fonksiyon şeklinde tanımlayarak, kendi içinde tutarlı güzergâh seçenekleri bulma problemi olarak Denklem 1'de gösterildiği gibi formüle edilebilir.

$$S: \mathcal{R} \rightarrow \mathcal{T} \quad (1)$$

$$r \mapsto \tau$$

Çoğu simülasyon modelinde $\mathcal{R}$, sürücüler kümesinden ağdaki $\mathcal{P}$ yolları kümesine kadar tüm haritaların kümesi; $\mathcal{T}$ ise, ağdaki bir yoldan oluşan tüm çiftlerin kümesindeki tüm haritaların kümesi olarak ifade edilmektedir. Ayrıca simülasyonlarda seyahat süresine karşılık gelen kalkış saati $\mathcal{T} = \mathbb{R}_{\geq 0}^{\mathcal{P} \times [0, T]}$, olarak tanımlanmaktadır. Burada $[0, T]$, simüle edilen zaman aralığı olarak tanımlanır.

Güzergâh seçim modeli fonksiyonu (C) Denklem 2 ile ifade edilmektedir.

$$C: \mathcal{T} \rightarrow \mathcal{R} \quad (2)$$

$$\tau \mapsto r$$

Denklem 2, güzergâh seçeneklerinin seyahat zamanlarına τ bağımlılığını tanımlayan bir fonksiyon olarak görülebilir. Eğer Denklem 3'teki durum geçerli olursa, her $r \in R$ için güzergâh seçim dizileri(setleri) kendi içinde tutarlı olmaktadır.

$$r = C(S(r)) \quad (3)$$

Eğer S bir simülasyon modeli tarafından değil de bir link maliyet fonksiyonu ile, yani link akış süresinin bağlantı akışına analitik bir değişken olarak verilmişse, kendi içinde tutarlı güzergâh seçimlerini belirleme problemi, BVM'yi ağa atamak gibi tanımlanabileceğinden *trafik atama problemi* olarak adlandırılmaktadır.

1.2.2. Güzergâh Seçim Modelleri ve Dengeleri

Temelde iki farklı güzergâh seçim tekniği bulunmaktadır. Birinci teknikte, bazı global modeller her sürücünün güzergâhını seçebilir. Bu durumda olağan varsayım, bu modellerin bazı küresel maliyet fonksiyonlarını optimize etmeye çalıştığı yönündedir

(Örneğin seyahat sürelerinin toplamı veya geçen araç sayısı). Global bir güzergâh seçim modeliyle trafik ataması, sistemin optimal trafik ataması olarak adlandırılır. İkinci teknikte, tüm sürücüler güzergâhlarını ayrı ayrı seçerler ve her bireyin kendi seyahat zamanını optimize etmeye çalıştığı varsayılır. Ortaya çıkan güzergâh seçimleri Wardrop'un ilk prensibi ile açıklanır (Wardrop, 1952): *“Kullanılan tüm güzergâhlar [sabit bir başlangıç varış yeri çifti için] eşit maliyetlere sahiptir ve kullanılmayan hiçbir güzergâh daha düşük maliyetli değildir.”* Bu durum kullanıcı dengesi trafik ataması olarak adlandırılır çünkü ortaya çıkan ağ durumu bir denge olarak görülebilir ve herhangi bir sürücü güzergâhını tek taraflı olarak değiştirerek seyahat süresini iyileştiremez.

Kullanıcı dengesi trafik atamasında, her bir bireyin ağ durumu hakkında mükemmel bir bilgiye sahip olduğu varsayılmaktadır. Akan trafik durumunda, trafik talebinin ve dolayısıyla ağ durumunun her gün aşağı yukarı aynı olduğu durumlarda bu varsayım yaklaşık tatmin edicidir. Ayrıca, her sürücünün mükemmel bir rasyonel ve tek tip davranış gösterdiği varsayılmaktadır.

Diğer bir varsayım ise, gerçek seyahat süresi ile birey tarafından algılanan seyahat süresi arasında bir ayırım yapılarak trafiğin rahatlatılabileceği yönündedir. Bu durumda algılanan seyahat süreleri, sürücülere dağıtılan rastgele değişkenler olarak tanımlanır. Hiçbir sürücünün, tek taraflı olarak değişen güzergâhlarla seyahat süresini geliştirebileceğine inanmadığı dengeye, stokastik kullanıcı dengesi denir (Gawron, 1998).

1.2.3. Statik ve Dinamik Atama

BVM ve bağlantı akışlarının, zamandan bağımsız olduğu varsayılırsa sorun statik atama olarak bilinir. Öte yandan zamana bağımlılık dikkate alındığında, dinamik atamadan bahsedilebilir. Dinamik atamanın, statik atamadan hem matematiksel hem de kavramsal olarak daha karmaşık olduğu görülmektedir (Saber, 2013).

Link akımları zamandan bağımsız değildir. Özellikle pik zamanların durumu açıklanırken statik atama iyi bir yakınsayıcı olarak kabul edilebilir. Çünkü durağan limit olarak kabul edilen trafik durumunda pik zamanlarda seyahat süresi sonsuza yakınsamaktadır. Sıkışık ağlar için bu durağan limitin gerçekçi olmadığı belirtilmelidir.

Bir linkten olabilecek çıkışlar linkin kapasitesini aşamayacağından, link kapasitesinden daha büyük bir giriş olması durumunda, link üzerinde kuyruklanma oluşmasına neden olmaktadır. Bu kuyruklanma, linkin akış girişi kapasitesini aştığı sürece büyüyecektir. Ancak akışın durağan olduğunu varsayarsak, kuyruk uzunluğunun sonsuz olması gerektiği düşünülebilir. Bu olay sonsuz bir seyahat süresi manasına geleceğinden gerçekçi olmayacaktır. Bu nedenle, “*Durağan limit güzergâh seçim modelinde de, bir linkin kapasitesi aşılamaz.*” olarak kabul edilmektedir (Gawron, 1998). Bir linkteki akış, yoğun saatlerde bağlantının kapasitesini aşabilmektedir. Çünkü en kısa fakat sıkışmış güzergâhtaki seyahat süresi, en kısa ikinci güzergâhtaki seyahat süresinden daha az olabilir. Bu durum araştırmacıların genellikle statik atama uygulamaları için kapasitenin üzerinde sınırlı kalan bir link maliyet fonksiyonu kullanmasının sebeplerinden birisidir. Özellikle sıkışık ağlardaki trafik akışına ilişkin durumlar, statik bir yaklaşımla tatmin edici bir şekilde tanımlanamamaktadır.

Statik ağ modellerinin bir diğer önemli eksikliği, bir seyahat talebinin tüm güzergâh üzerinde kapasite tahsis eden bir akışla sonuçlanmasıdır. Bu nedenle, ağın ortak bir arterini paylaşan güzergâhlardaki akışlar her zaman etkileşime girerken, gerçekte farklı güzergâhlardaki sürücüler büyük otoban ağlarında farklı zamanlarda farklı linkleri kullanabilir.

Dinamik Trafik Ataması (DTA) gibi trafik tahminleri içeren bir ortamda herhangi bir sinyal optimizasyon şemasının tüm avantajlarından yararlanmak için, güzergâha bağlı etkileri sisteme dahil etmek başarıyı arttıracaktır. Mevcut modellerin birçoğu bu yönü doğru bir şekilde ele almamaktadır.

1.3. Trafik Değişkenleri

Trafikteki temel değişkenler mikroskobik (araç bazında) ve makroskobik (trafik ağı ve link bazında) olmak üzere iki seviyede incelenmektedir. Mikroskobik seviyedeki değişkenler şu şekilde açıklanabilir:

Takip Mesafesi (Spacing): Akan trafikte iki araç arasındaki ön tampondan-ön tampona olan ortalama mesafedir. Cinsi metre (m)’dir

Takip süresi (Headway): Birbirini izleyen araçlar arasında, her araç için aynı noktadan ölçülen (ön tampondan-ön tampona gibi) bir şerit veya yol üzerindeki bir noktaya aracın ulaştığı andaki geçen süredir. Cinsi saniye (sn)'dir.

Hız (Speed): Bir aracın 1 saniyede ilerlediği mesafedir. Cinsi m/sn'dir.

Makroskobik seviyedeki değişkenler ise aşağıdaki gibidir:

Akış (Flow): Belirlenen bir nokta üzerinden bir saatte geçen araç sayısıdır. Akış en sade haliyle Denklem 4 ile ifade edilmektedir:

$$q = \frac{N}{t(sa)} \quad (4)$$

Yoğunluk (Density): Bir linkteki veya bölgedeki araçların link uzunluğuna veya bölge alanına oranı olarak ifade edilmektedir. Yoğunluk basitçe Denklem 5 ifade edilmektedir:

$$k = \frac{N}{x(m)} \quad (5)$$

Ortalama seyahat hızı (Average Travel Speed): Bir linkteki veya bölgedeki araç sayısının bölgeyi geçmek için harcadığı toplam süreye olan oranıdır. Denklem 6 ile ifade edilmektedir:

$$S = \frac{nL}{\sum_{i=1}^n t_i} \quad (6)$$

Burada $S(\text{km/sa})$ ortalama seyahat hızını, $L(\text{km})$ linkin uzunluğunu, $t(\text{sa})$ i . aracın linki veya bölgeyi geçmek için geçen süreyi ve n toplam gözlem sayısını ifade etmektedir.

Tez kapsamında yapılacak çalışma ağ bazında olduğundan makroskobik boyutta olan trafik ağı düzeyindeki akış değişkenleri olan akış (Q) ve yoğunluğun (K) link bazlı ve güzergâh bazlı ölçümlerindeki durumları incelenecektir.

1.3.1. Trafik Ağı Düzeyindeki Akış Değişkenleri

Ağ genelindeki trafik akışı değişkenleri genellikle link ölçümlerine dayanarak hesaplanmaktadır. Link bazlı ölçümler, ağın bireysel linklerden gelen trafik değişkenleri verileri mevcut olduğunda tahmin edilir. Burada ağ genelindeki trafik değişkenlerinin

bireysel araç yörüngelerini kullanarak da hesaplanabileceği önerilmiştir. Bağlantılı araç (Connected Cars) teknolojisi olarak bilinen, tam bağlantılı bir taşıma sisteminin geliştirilmesi ve yayılmasıyla bireysel araçlar sistemde “algılayıcılar” olarak hareket eder. Bu nedenle, bireysel araç güzergâhlarının iyi bilinmesi beklenmektedir. Bu amaçla bazı cep telefonu ve GPS navigasyon şirketleri, araç yörüngelerini elde etmek için, GPS özellikli cep telefonlarından veya GPS navigasyon cihazlarından alınan kalabalık temelli trafik verileri kullanmaktadır. Kalabalık kaynaklı araç güzergâh kullanımı günümüzde istenen düzeyde değildir ancak önümüzdeki birkaç yıl içinde artması beklenmektedir. Böyle bir durumda ağ genelindeki güzergâha dayalı ölçümler, kalabalık kaynaklı trafik verileri veya simülasyon verisi mevcut olduğunda tahmin edilebilir olacaktır.

1.3.1.1. Link Bazlı Ölçümler

Ağ genelindeki link bazlı ölçümleri tahmin etmek için kullanılan ortalama ağ akışı ve ortalama ağ yoğunluğu Mahmassani ve diğ. (1984) tarafından Denklem 7 ve 8 ile ifade edilmiştir (Mahmassani vd., 1984):

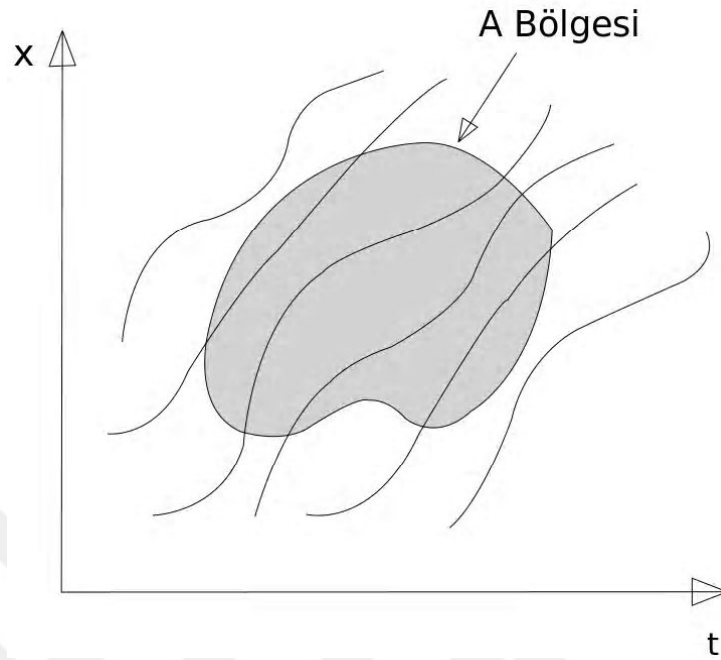
$$Q = \frac{\sum_{i=1}^M l_i q_i}{\sum_{i=1}^M l_i} \quad (7)$$

$$K = \frac{\sum_{i=1}^M l_i k_i}{\sum_{i=1}^M l_i} \quad (8)$$

Burada Q değeri ortalama ağ akışı, K ortalama ağ yoğunluğu, q_i , k_i ve l_i sırasıyla link bazında ortalama akış, gözlem süresi yoğunluğu ve $i = 1, \dots, M$ olan i adlı linke ait uzunluktur. M ise linklerin sayısıdır.

1.3.1.2. Güzergâh Bazlı Ölçümler

Güzergâh, bir aracın veya nesnenin, uzaydaki başlangıç ve bitiş noktaları arasında izleyeceği yol olarak tanımlanabilir. Güzergahtaki her bir link üzerinde, araçlar genellikle tek boyutlu bir yolda, örneğin bir otoyol şeridi boyunca hareket etmekle kısıtlıdır. Bundan dolayı ayrık güzergahların zaman-uzay diyagramları genellikle sadece zaman eksenini (t) ve mesafeyi (x) içermektedir. Bu tür güzergahlara iki boyutlu güzergâhlar adı verilmektedir (Şekil 1).



Şekil 1. Bir Uzay-Zaman Diyagramındaki İki Boyutlu Güzergâhlar (Saber, 2013)

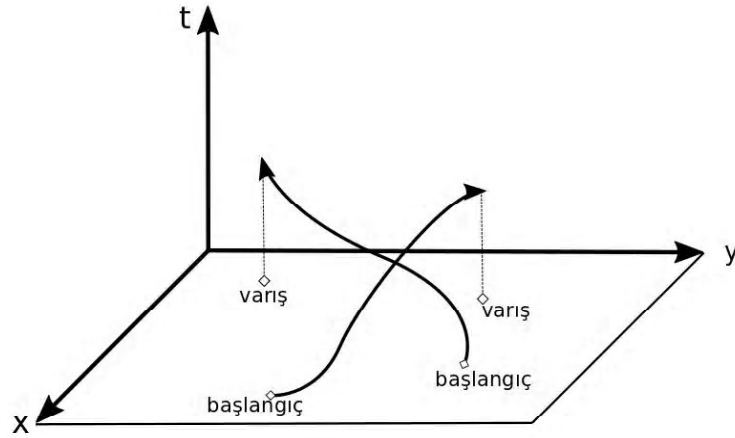
Edie (1961), Denklem 9 ve 10’da belirtilen genelleştirilmiş akış ve yoğunluk tanımlarını önermiştir (Edie, 1961):

$$q_A = \frac{d(A)}{A} \quad (9)$$

$$k_A = \frac{t(A)}{A} \quad (10)$$

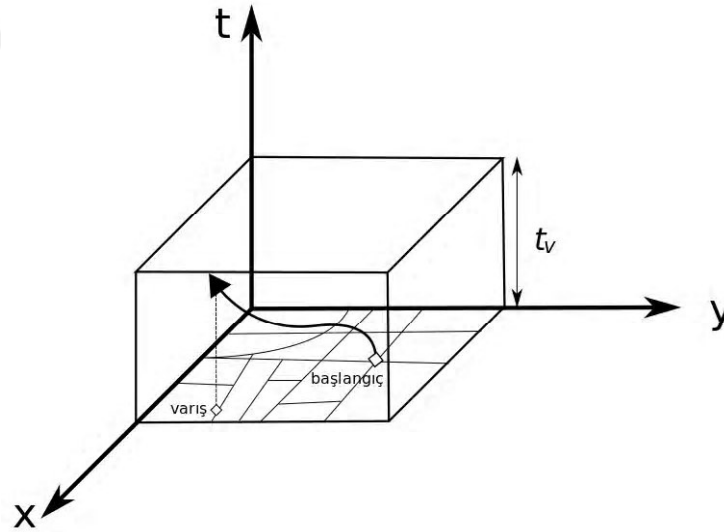
Burada $d(A)$, A bölgesindeki tüm araçların kat ettiği toplam mesafeyi, $t(A)$, bölgedeki tüm araçların trafikte geçirdiği toplam zamanı ve A , A bölgesinin kapladığı alanı ifade etmektedir.

Bireysel araç hareketlerini incelerken, iki boyutlu uzay-zaman diyagramlarının yaygın olarak uygulanmasından farklı olarak, Şekil 2’de gösterildiği üzere ağ trafiği üç boyutlu bir uzay-zaman diyagramına sahiptir. Trafik ağında, bir başlangıç kaynağından bir hedefe varışa ilişkin araç hareketleri karmaşıktır ve x-y düzleminde geniş bir x ve y aralığını kapsayabilir.



Şekil 2. Üç Boyutlu Uzay-Zaman Diyagramı (Saber, 2013)

Şekil 3'te görüldüğü gibi ağ genelindeki trafik değişkenlerini, yörüngeleri kullanarak tahmin etmek için iki boyutlu uzay-zaman diyagramındaki A bölgesine benzer bir küp gibi V hacmiyle kapalı üç boyutlu prizma kullanılmaktadır. Bu prizmanın zemini ağ yapısını oluşturmaktadır.



Şekil 3. Üç Boyutlu Uzay-Zaman Diyagramının x-y Düzlemindeki Ağ Yapısı (Saber, 2013)

Bu nedenle Edie'nin tanımlarına dayanan genel ağ genelinde trafik değişkenleri Denklem 11 ve 12 ile ifade edilebilir (Edie, 1961):

$$Q V = \frac{d(V)}{L_{xy}(V) \times t_v} \quad (11)$$

$$K V = \frac{t(V)}{L_{xy}(V) \times t_v} \quad (12)$$

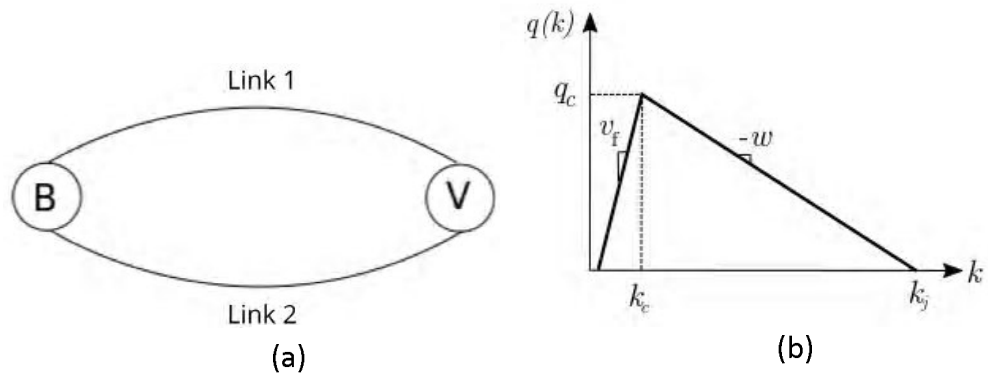
Burada $Q V$ ve $K V$ değerleri ile ağ bazında ortalama akış ve yoğunluk ifade edilmektedir. $d V$ değeri V hacmindeki bulunan araçların toplam kat ettiği mesafeyi, $t V$ ise mesafe kat edilirken geçen süre, $L_{xy} V$, V hacmi ile ilişkilendirilmiş x-y düzleminde ağın uzunluğu ve t_v , V hacminin zaman yüksekliği olarak ifade edilmektedir.

Konvansiyonel yoğunluk tanımına benzer şekilde, n 'nin dt düzlemindeki taşıt sayısı olduğundan hareketle ağ genelindeki yoğunluk, Denklem 13 ile gösterilebilir:

$$K t = \frac{n, dt}{L_{xy}, dt} \quad (13)$$

1.3.2. İki Linkli ve Serbest Yol Ağı

Daganzo ve diğ. (2011) ve Cassidy ve diğ. (2011) çalışmalarında teorik olarak basit iki linkli otoyol ağı olduğu varsayılır. Her iki link aynı uzunluğa sahiptir ve maksimum akış q_c , sıkışma yoğunluğu k_j , serbest ortalama hız v_f ve geriye doğru dalga hızı w ile üçgen olduğu varsayılan aynı ideal temel diyagramı takip etmektedir (Daganzo vd., 2011)(Cassidy vd., 2011) (Şekil 4(a) ve 4(b)).



Şekil 4. (a) Serbest Trafik Ağı; (b) Linklerin Temel Diyagramı(Daganzo vd., 2011)

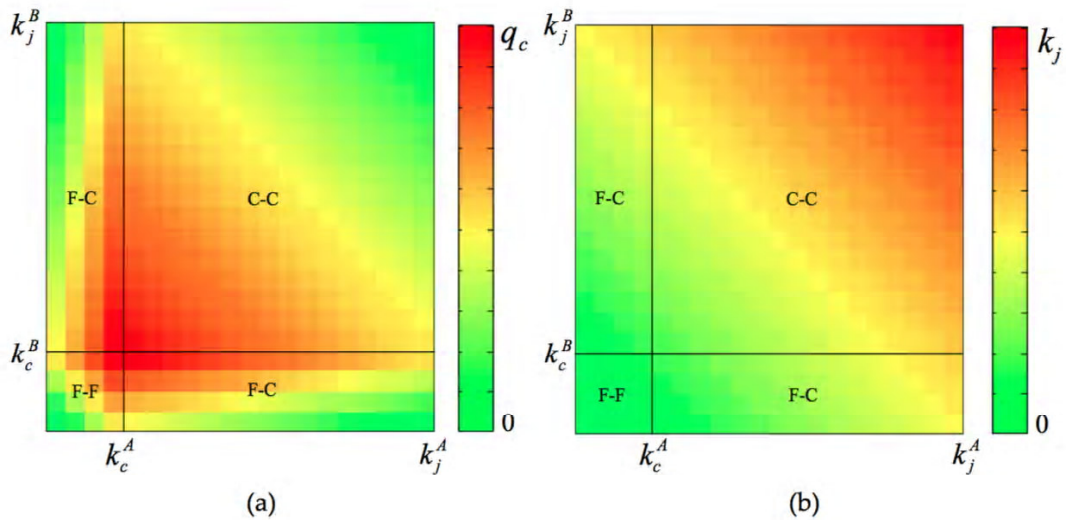
Daganzo ve diğ. (2011) (Daganzo vd., 2011), söz konusu ağın herhangi bir zamanda olası üç rejimden birinde olabileceğini göstermiştir:

- i) Her iki linkin serbest akış koşullarında çalıştığı Serbest-Serbest rejim (F-F);

- ii) Bir bağlantının linkin akış koşullarında, diğerinin sıkışık şartlarda çalıştığı, Serbest-Sıkışık rejim (F-C);
- iii) Her iki linkin d3 sıkışık koşullarda çalıştığı, Sıkışık-Sıkışık rejim (C-C).

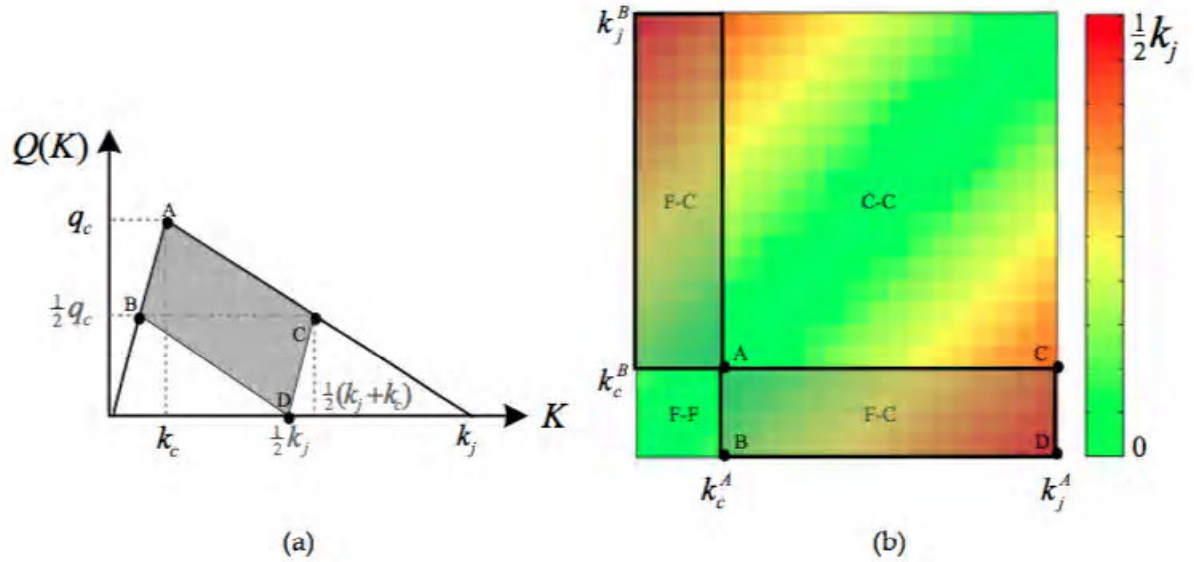
Bu rejimler iki bileşenli bir faz diyagramında gösterilebilir (bkz. Şekil 5); ve bu diyagrama bakılarak linklerden hangisinin sıkıştığına bağlı olarak iki Serbest-Sıkışık durumu da ayırt edebilmektedir. Serbest-Serbest ve Sıkışık-Sıkışık rejimlerinde, trafik ağda homojen olarak dağıtılmaktadır. Bu nedenle basit ağ için sadece sistemin durumu bu rejimlerden birine dönüşmesi durumunda, üçgen akış-yoğunluk eğrisi tanımlanabilir. Ancak, sistem Serbest-Sıkışık rejimi içindeyse histerezis oluşacaktır. Şekil 5, k_c^A , ve k_j^A 'nın (k_c^B ve k_j^B), güzergâh A (B) 'nin kritik yoğunluğu ve sıkışma yoğunluğu olduğu ortalama ağ akışını ve ortalama ağ yoğunluğunu gösteren iki faz diyagramını göstermektedir. Görüleceği üzere ortalama ağ akışı, rejimler arasındaki sınır çizgilerinin kesişme noktasına doğru artmaktadır. Şekil 4, ortalama ağ akışı $Q(K)$ ve ortalama ağ yoğunluğu K ile ilgili olarak Şekil 6(a)'te gösterilen ağ için akış yoğunluğu ilişkisini gösterir. Şekil 6(a)'daki gölgeli yamuk, histerezisin olabileceği alanı gösterir. Şekil 6(b), ağdaki trafik dağılımının homojen olmama ölçüsü olarak tanımlanan ağ yoğunluğunun δ_k standart sapmasını gösteren bir faz şemasını göstermektedir. Ağ yoğunluğunun standart sapması Denklem 14 ile ifade edilmiştir:

$$\delta_k = \frac{(k^A - K)^2 + (k^B - K)^2}{2} \quad (14)$$



Şekil 5. Yoğunluk A-Yoğunluk B Faz Diyagramı (a) Ağın Ortalama Akışı; (b) Ağın Ortalama Yoğunluğu (Saber, 2013)

Faz diyagramındaki gölgeli dikdörtgenler, Şekil 6 (a) 'daki gölgeli yamuk ile bağlantılı sistemin durumlarını göstermektedir. Sınır noktaları A'dan D'ye kadar işaretlenmiştir. Ağ yoğunluğunun standart sapması faz diyagramının sol üst ve sağ alt köşelerine doğru artmaktadır. Burada C-C rejimi dikdörtgeninin bazı alanlarındaki ağ yoğunluğunun standart sapmasının, en fazla ağın durumu F-C rejimi durumundayken olan ağ yoğunluğunun standart sapması kadar olabildiği görülmektedir. Bu nedenle daha önce belirtilen ağ trafiğinin durumuna bakılarak, bir ağın temel diyagramının üçgen şeklinde olması için “homojen olarak dağıtılmış” olması, ağın her iki linkinin akış halinde olması durumu ya da tıkanması durumu daha kesin olarak ifade edilebilir.



Şekil 6. (a) Ortalama Ağ Akışına karşılık gelen Ortalama Ağ Yoğunluğu; (b) Ağ Yoğunluğunun Standart Sapmasını Gösteren Yoğunluk A-Yoğunluk B Faz Diyagramı (Saber, 2013)

Sonuçlar ağ yoğunluğunun standart sapmasının, histerezisi modelleyebilmek amacıyla ağdaki trafik uzaysal heterojenliğini tam olarak yansıtamamaktadır. Bunun için de, Trafik Uzaysal Homojenite Endeksi (TUHE) olarak adlandırılan ek bir önlem Denklem 15 ile önerilmektedir:

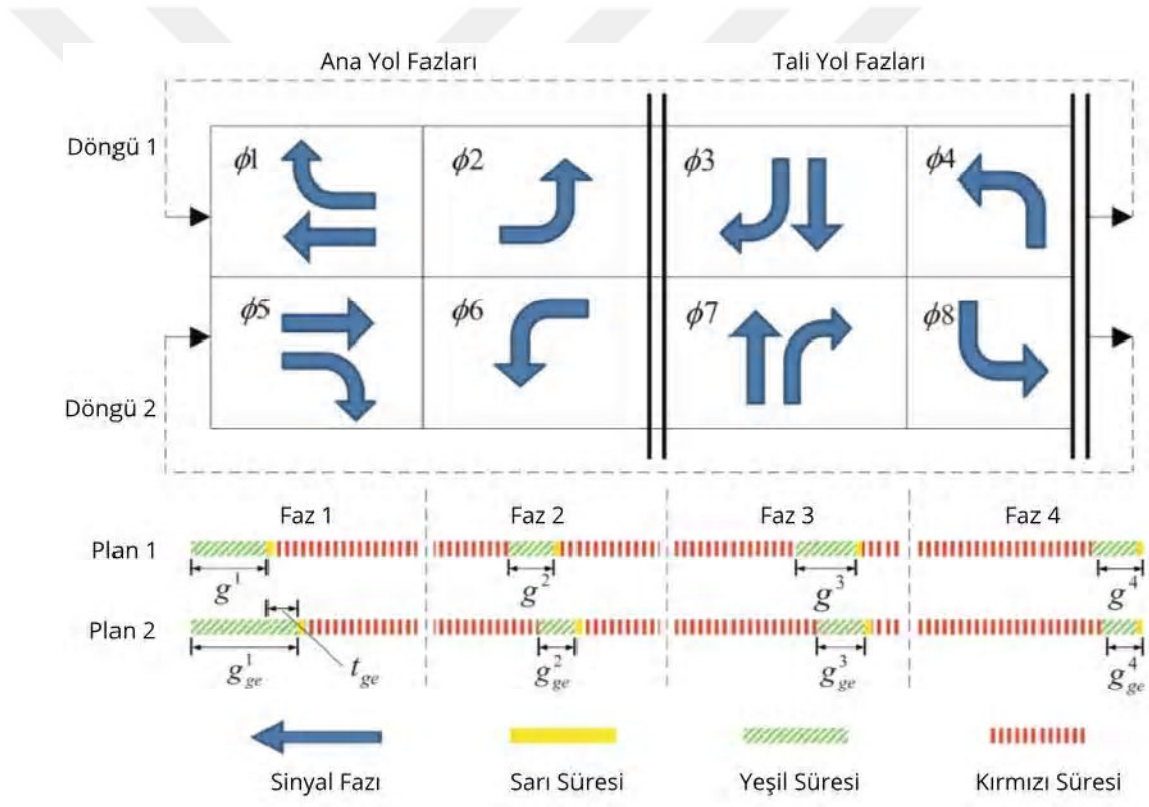
$$TUHE = \frac{L_U - L_C}{L} \quad (15)$$

Burada L_u yoğun olmayan, L_c ise yoğun olan link uzunluklarının toplamıdır. L ise ağdaki linklerin toplam uzunluğu olarak tanımlanır.

1.4. Trafik Sinyal Kontrolü İçin Temel Parametreler

1.4.1. Sinyal Fazı (Signal Phase)

Trafik sinyal kontrolünde, bütün yönlerdeki trafik akışı arasındaki tıkanıklığı önlemek için, tüm trafik akışının sinyal döngüsünde aynı anda aktif olması mümkün değildir. Genel olarak bir sinyal döngüsünde aynı renkli sinyal lambaları ile trafik akışı aynı sinyal fazına sahiptir.



Şekil 7. Sinyal Faz Planları ve Döngüleri (Li ve Jin, 2017)

Şekil 7’de sinyal faz planları ve döngüleri görülmektedir. Sinyalize bir kavşakta gün içerisinde yoğunluğa göre birden fazla sinyal faz planı ve döngüsü uygulanabilmektedir. Şekil 7’de görülen ϕ fazı, g yeşil faz süresini, t ise iki döngüde yeşil fazların farkını ifade etmektedir.

1.4.2. Doygunluk (Saturation)

Akış oranı olarak da bilinen bir kavşaktaki doyumluk, birim zamandaki trafik akışının doyumluk akışına oranı anlamına gelmektedir (Li, 2009). Burada, trafik akışı birim zaman içinde kavşaktan geçen araç sayısını; doyumluk akışı ise bir sinyal çevriminde yeşil ışık yanıyorsa, kavşaktan geçen maksimum araç sayısını ifade etmektedir.

1.4.3. Zaman Gecikmesi (Time Delay)

Zaman gecikmesi sürücülerin sürüş sırasında trafik kurallarını gördükleri andan, tepkime verdikleri ana kadar geçen süredir (Bando vd., 1995). Örneğin kırmızı fazda olan bir trafik lambasının, yeşile döndüğü andan itibaren, en öndeki aracın hareket etmeye başladığı ana kadar geçen süre zaman gecikmesidir.

Trafik sinyali kontrolü sırasında, zaman gecikmesi, trafik erişimi üzerinde direkt bir etkiye sahip olduğundan, genellikle trafik kazanımının anahtar indeksi olarak tanımlanmaktadır. Son yıllarda, geleneksel Webster sinyali gecikme formülü trafik alanında yaygın bir şekilde kullanılmaya başlanmıştır ve Denklem 16 ile ifade edilmektedir.

$$D_i = \frac{c(1-g_i)^2}{2(1-y_i)} + \frac{y_i^2}{2q_i g_i (g_i - y_i)} \quad (16)$$

Burada c döngü süresi, g_i etkin yeşil ışık süresinin döngü süresine oranı, q_i faz i 'nin trafik akışı, y_i faz i 'nin doyumluğu olarak tanımlanmaktadır.

Webster formülü yalnızca doyumluk daha küçük olduğunda kullanılabilir. Bu nedenle Yang tarafından geliştirilmiş ve zaman aşımı Denklem 17 ile tanımlanmıştır (Yang ve Yang, 2001):

$$D_i = \frac{cq_i(x-y_i)^2}{2x^2(1-y_2)} + \frac{x^2}{2(1-x)} \quad (17)$$

Burada x bir kavşağın doyumluğu olarak tanımlanmaktadır.

1.4.4. Trafik Lamba Sayısı (Number of Traffic Stops)

Bilindiği üzere lamba sayısı doyumluk ile ters orantılıdır. Başka bir deyişle daha az sayıda lamba, trafik akışını olumlu yönde etkiler. Yang'ın araştırmasına göre, faz i için beklenen lamba sayısı Denklem 18 ile tanımlanmaktadır:

$$H_i = \frac{y_i(x-y_i)}{2x^2(y_i-x^2)} \quad (18)$$

Burada y_i faz i 'nin doyumluğu ve x bir kavşağın doyumluğu olarak tanımlanmaktadır.

1.4.5. Trafik Kapasitesi (Traffic Capacity)

Trafik kapasitesi bir linkin alabileceği toplam araç sayısıdır. Genel olarak, sadece yeşil ışıktaki geçebilen araç sayısı kapasiteden az olabilir. Dolayısıyla durma sayısı ve trafik kapasitesi arasındaki ilişkiye göre trafik kapasitesi Denklem 19 ile ifade edilebilir:

$$Q_i = s_i \cdot 1 - \frac{y_i(x-y_i)(1-y_i)}{1.8x^2(y_i-x^2)} \quad (19)$$

Burada s_i , faz i 'nin doyum akışıdır.

1.4.6. Optimum Sinyal Zamanı için Matematik Model

Gerçek trafik talebi göz önüne alındığında en küçük gecikme, en az sayıda durma ve en büyük trafik kapasitesi nihai hedefler olarak seçilmektedir. Zaman gecikmesi, durma sayısı ve trafik kapasitesi, döngü süresi c 'nin ve bir kavşak kesişim noktasının doyumluğu olan x 'in bir fonksiyonudur. Trafik kapasitesini maksimize etmek ve zaman gecikmesini ve durak sayısını minimize etmek için sinyal zamanlamasının optimizasyon problemi, amaç fonksiyonunun iki ana değişkeni olan c ve x ile farklı trafik talepleri ile değişen üç ağırlıklandırma katsayısı kullanılarak, minimizasyon formülüne dönüştürülebilmektedir. Sinyal zamanlamasının optimizasyon modeli Denklem 20 ile yazılabilir:

$$\min f \ x, c = \frac{\sum_{i=1}^n (K_i^1 D_i + K_i^2 H_i)}{\sum_{i=1}^n K_i^3 Q_i} \quad (20)$$

$x \in [0.6, 0.8]$, $c \in [40, 120]$ olmak üzere, 0.6, zaman gecikmesinin artması ve düşük doyumluğun neden olduğu durakların önlenmesinden, 0.8 ise daha büyük doyumluktan

kaynaklanan trafik tıkanıklığından kaçınmaktır; çevrim süresi aralığı genellikle [40, 120] aralığı olarak tanımlanır (Yan ve Li, 2006) K_i^1 , K_i^2 ve K_i^3 sırasıyla zaman gecikmesi, durma sayısı ve trafik kapasitesinin ağırlık katsayılarıdır ve Denklem 21, 22 ve 23 ile hesaplanmaktadır (Gu ve Wang, 1998):

$$K_i^1 = 2 \cdot (1 - Y) \cdot \bar{s}_i \quad (21)$$

$$K_i^2 = \bar{s}_i \cdot \frac{1-Y}{0.9} \quad (22)$$

$$K_i^3 = 2Y \cdot \frac{c}{3600} \quad (23)$$

Burada $Y = \max_{j=1,2,..} y_i^j$, ve $y_i^j (j = 1,2,..)$ faz i deki a yönündeki doyumluktur. Böylelikle $f(x, c) \geq 0$ olacağından $\min f(x, c) \geq 0$ eşitsizliği de sağlanmış olur.

1.4.7. Ağırlık Katsayılarının Belirlenmesi

Ağırlıklandırma katsayıları verimlilik ve güvenlik arasındaki dengeyi temsil etmektedir (Bad ve Fahmy, 2004). Bunun için ağırlık katsayıları olan, K_i^1 , K_i^2 ve K_i^3 'nin belirlenmesi de önemlidir. Burada aşağıdaki hususlar göz önüne alınarak yapılır:

- 1- K_i^1 , K_i^2 'nin değeri yoğunluk artışı ile azalır, ancak K_i^3 yoğunluğun artmasıyla artmaktadır. Böylece pik saatler süresince trafik kapasitesini artıracak ve trafiğin serbest olduğu saatler süresince zaman gecikmesi ve durma sayılarının azalması hedeflenmektedir.
- 2- Trafik açık olduğu saatler boyunca herhangi bir kavşakta anlık bir yoğunlaşma oluşursa, zaman gecikmesine neden olma ihtimalinin daha yüksek olduğu göz önüne alınmalıdır. Bu yolla zaman gecikmesi olan K_i^1 'nin doyumluk akışı olan s_i 'sinin artmasıyla daha yoğun kavşaklarda gecikmenin azalması hedeflenmektedir.
- 3- Daha uzun döngü süresi, daha büyük yoğunluğa neden olmaktadır. Bu nedenle, trafik kapasitesini bir dereceye kadar arttırmak için, döngü süresi c , K_i^3 'ye dahil edilmektedir.

1.5. Trafik Sinyal Düzenleme Süreci

Yapılan araştırmalar ve edinilen tecrübeler, trafik sinyal düzenlemesinin trafik akışını iyileştirmek için yapılabilecek en doğru yöntem olduğunu göstermektedir.

Sinyal zamanlama optimizasyonu ve sahada uygulanmasına yönelik geleneksel yaklaşım, güncel trafik akışı verileri, optimizasyon modelleri, sinyal kontrol cihazı donanımına aşinalık ve sinyal zamanlaması ince ayarı dahil birçok saha tecrübesi gerektirir. Etkin trafik akışı için yeni sinyal zamanlama parametreleri ayarlamak zaman alıcı ve pahalı olmaktadır. Kaynaklara göre trafik sinyal düzenleme sürecinin uygulanması farklılık gösterebilse de kabul gören geleneksel süreçte trafik sinyallerinin düzenlenmesi için 5 adım takip edilmektedir (Henry, 2005):

1. Mevcut bilgilerin düzenlenmesi,
2. Alandaki yeni trafik akışı verilerini toplanması,
3. Sinyal zamanlama için en iyi programların değerlendirilmesi ve çalıştırılması,
4. En iyi sinyal zamanlama ayarlarının doğrulanması ve seçilmesi,
5. Sokaktaki sinyal denetleyicilerinde yeni sinyal zamanlama planlarının kurulması ve ince ayar yapılması.

Bununla birlikte, bu süreci kısaltmak için pratik ve uygun maliyetli yöntemler geliştirmiştir. Ayrıca hala geleneksel modelleme süreci kullanılarak geliştirilen sinyal zamanlamasının etkinliğini yaklaşık olarak tahmin edebilecek sinyal zamanlama planları üreten yöntemler bulunmaktadır. Bu planlara “yakın-optimum” (near-optimum) planlar adı verilmektedir. Elbette ki yakın-optimum yöntemlerden karmaşık ve maliyetli sezgisel optimizasyon süreci ile aynı kalitede sinyal zamanlaması sonuçlarının beklenmesi makul değildir. Fakat, sezgisel optimizasyon algoritmaları gibi karmaşık yöntemlerde sinyal zamanlaması mümkün olmayacak şekilde kaynak yetersizliği ile karşılaşıldığında, sinyal düzenlemesi için yakın optimum planlar kullanılmaktadır. Bu bölümde bir trafik ağındaki sinyal sürelerinin optimize edilebilmesi için gerekli sinyal planlama sürecinin teorik adımları hakkında bilgi verilecektir.

1.5.1. Sinyal Planlama Süreci

Sinyal planlama süreci birden fazla kavşak söz konusu olduğunda gerekli hale gelmektedir. Süreçte dikkat edilmesi gereken nokta çalışma alanının sınırlarının doğru belirlenmesidir. Bu işlem genellikle yeterli veri bulunuyorsa bu veriler üzerinden, veri yok ise tecrübeye dayalı olarak belirlenmektedir. Sonrasında sinyal planlama süreci başlatılır. Sinyal planlama sürecinde sekiz farklı adım vardır. Bunlar:

- 1- Sistem kavşaklarının tanımlanması
- 2- Mevcut verilerin toplanması ve düzenlenmesi
- 3- Saha araştırması
- 4- Günlük senaryo sayısının belirlenmesi
- 5- Yerel zaman parametrelerinin hesaplanması
- 6- Sinyal gruplarının tanımlanması
- 7- Koordinasyon parametrelerinin hesaplanması
- 8- Yeni planın uygulaması ve değerlendirilmesi

1.5.2. Sistem Kavşaklarının Tanımlanması

Adımın amacı, çalışma dahilinde analiz edilecek olan tüm kavşakları açıkça belirlemektir. Bu tespit önemli bir konudur çünkü, tüm kavşaklar trafik hareketinin durduğu ve başladığı noktalar olduğundan trafiğin akışını doğrudan etkilemektedir. Dolayısıyla sistem kavşaklarını doğru tanımlamak sistemin optimizasyonu için en önemli adım sayılabilir.

Sistem kavşakları tanımlanırken öncelikle her kavşak benzersiz bir ad ve numara ile tanımlanmalıdır. Numaralandırma şeması kavşakların geometrisini yansıtacak şekilde düzenlenip liste haline getirilmelidir. Örneğin kavşak noktaları genel olarak doğu ve batıya uzanan bir arter üzerinde ise, sayılar en batıdaki kavşak noktası için başlayabilir ve doğuya doğru arttırılabilir.

Sistem kavşaklarının tanımlanması noktasında belirlenen liste çalışma yapılacak tüm kavşakları içerecektir. Bu durum tüm kavşakların mutlaka bir grup veya sistem olarak birlikte çalışacağı anlamına gelmeyip; sadece seçilen kavşak noktalarının birlikte değerlendirileceği anlamına gelmektedir. Sinyal planlama sürecinin bu aşamasında yapılacak operasyonel analizler neticesinde; kavşakların tümünün birden, tek tek veya gruplar halinde çalışmasına karar verilmektedir.

1.5.3. Mevcut verilerin toplanması ve düzenlenmesi

Sinyal zamanlama planlarını hazırlamak için gereken veriler tanımlayıcı ve talep veriler olmak üzere iki kategoriye ayrılabilir. Tanımlayıcı veriler elde edilmesi kolay olanıdır ve çoğunlukla işletmecilerden veya belediyelerden temin edilebilir. Bu veriler aşağıdakileri bilgileri içerir:

- **Tüm linklerdeki şerit sayısını ve şeritlerin genişliğini gösteren kavşağın durum diyagramı:** Diyagram cadde ve sokak adlarını içermelidir
- **Mevcut kontrol cihazlarıyla elde edilen kavşak noktalarına ait faz diyagramı:** Faz diyagramının, her faz hareketi için NEMA faz numarasını içermesi önemlidir.
- **Mevcut detektör yeri, tipi ve faz tahsis bilgisi:** Bu veriler, minimum yeşil ve uzatma gibi faz aralığı ayarlarını belirlemek için gereklidir.
- **Mevcut trafik sayımı verileri:** En kullanışlı veriler zaman serisi verileridir. Eski sayım verilerini kullanırken ve sayım yapıldıktan sonra trafik talebinde büyük bir değişiklik olup olmadığını belirlemek gerekir. Talepte önemli bir değişiklik olmadıysa, yıllık trafik büyümesi için sayımlar ayarlanabilmektedir. Büyük bir değişiklik olması durumunda sayımlar yeterli olmayacaktır. Saatlik sayımlar ve hatta bir yıla ait günlük ortalama trafiğin 24 saat sayımı da oldukça faydalıdır. Bu sayımlar trafik büyümesini ve hatta hareket sayımını tahmin etmek için kullanılabilir. Manuel sayım, pahalı ve çok çaba gerektiren bir teknik olsa da elde edilen çözümler uzun süreli olduğundan her halükârda kârlı olmaktadır.
- **Zamanlama planının çalışacağı koşullar için kavşaklar ve serbest akışlı hareket hızı arasındaki mesafe:** Açıklanan bilgiler yolları ve işaretlenmiş kavşakları gösteren alanın haritasında tanımlanmalıdır. Haritanın ölçeklendirilmesi

gerekli değildir. Haritadaki her bağlantının; bağlantı uzunluğu, hız ve ses gibi çeşitli verileri gösterebilecek kadar ayrıntılı olması önemlidir.

1.5.4. Saha Araştırması

Saha Araştırması adımı çalışma yapılacak bölgeyi yerinde incelemek manasına gelmektedir. Çalışma yapılacak kavşakları görmeden de hem yerel hem de koordineli sinyal zamanlama parametreleri üretmek mümkün olsa da bu durum çok problemli olabilmektedir. Bir plan sayfasında belirtilebilecek veya belirtilmeyecek, ancak trafik akışı üzerinde bariz bir etkiye sahip olabilecek fiziksel kısıtlamalar bu adımda net biçimde ortaya çıkarılmalıdır. Örneğin bitki örtüsünden dolayı oluşan görüş kısıtlılığı, kavşak giriş ve çıkışlarını engelleyebilecek olumsuzluklar, yollardaki eğrilik ve kötü kaldırım gibi etkenler saha araştırması sırasında ortaya çıkan ve trafik akışını etkileyen faktörlere örnek olarak verilebilir.

Saha araştırması, mevcut verilerin tümü toplandıktan ve organize edildikten sonra yapıldığında oldukça etkilidir. Saha ziyaretinin amacı, önceki adımda toplanan mevcut bilgilerin doğru olduğunu teyit etmek ve gerekli olabilecek ek verileri toplamaktır. Bunun için özellikle her kavşağa ait bir kontrol listesi hazırlanmalıdır. Temel kavşak kontrol listesi aşağıdakileri içerir:

Durum Diyagramı: Bu, önceki adımda elde edilen kavşak taslağının doğrulanması veya mevcut bir çizim yoksa yeni bir şema hazırlanması şeklinde olabilmektedir. Bu şema aşağıdakileri içermelidir:

- Kavşaklara ilişkin taslaklar, karayolu kavşakları, kaldırımlar, yaya geçitleri, Kuzey oku, sokak isimleri.
- Genişlikler ve hareket atamaları dahil olmak üzere yaklaşma şeridi yapılandırmaları.
- Görüş mesafesi kısıtlamaları, düşey veya yatay eğrilik, bitki örtüsü.

Faz Diyagramı: Durum diyagramı gibi, faz diyagramında da mevcut bilgilerin doğrulanması gereklidir. Aksi taktirde yeni bir faz diyagramı hazırlanabilir.

Detektör Konumları: Faz aralığı ayarlarını belirlemek için mevcut detektör konumu, türü ve faz atama verileri gereklidir. Saha ziyaretinin amacı, detektörlerin mevcut belgelerde gösterildiği gibi konuşlandırıldığını doğrulamak olmalıdır. En önemli amaç detektörlerin olması gerektiği gibi çalıştığını doğrulamaktır.

Trafik Akışı Gözlemleri: Bütün kavşak noktalarını incelerken, linklerde ayrı ayrı gözlemlenen tipik serbest akış hızının bilgisidir.

Sinyal zamanlamasının etkili olması için bu minimum maliyet yaklaşımında mevcut tüm verilerin tam olarak kullanılması hayati öneme sahiptir. Sürecin bu noktasında uygulayıcı mevcut parametreler ve detektör operasyonu hakkında tam bilgi sayesinde, ilgilenilen süre boyunca kavşağın çalışmasını gözlemleyebilmektedir. Kavşak performansını değerlendirmek için gerçek bir araç kullanabilmek önemli olsa da düşük bütçeli yaklaşım, performans açısından yetersizdir. Bunun yerine trafik uzmanının gözlemleri ve deneyimleri göz önünde bulundurulmalıdır.

1.5.5. Günlük senaryo sayısının belirlenmesi

Günlük senaryo sayısının belirlenmesi adımı, geliştirilecek zamanlama planında her bir birincil kavşak için döngü verilerinin hazırlanmasını içerir. Bu senaryo örneğin; sabah, öğle ve akşam gibi günlük trafik talebi davranışının değiştiği dönemlerde farklı senaryolar uygulanmasını öngörmektedir.

1.5.6. Yerel zaman parametrelerinin hesaplanması

Bu adım saha ziyaretinden sonra yerel denetleyici operasyon parametrelerini gözden geçirilmesi durumunun tekrar düzenlenmesini kapsamaktadır.

Yerel operasyon parametreleri faz minimum ve maksimum değerleri, faz değişimi ve ofset gibi ayarlardır. Bu ayarlar temel olarak trafik talebi, kavşağın geometrik tasarımı, detektörlerin türü ve konumu gibi verilerin sonucunda ortaya çıkmaktadır.

1.5.7. Sinyal gruplarının tanımlanması

Sinyal zamanlama işlemindeki en ideal durum tüm kavşakların izole kavşaklarda olduğu gibi verimli çalışmasıdır. Başka bir ifade ile her kavşak yerel talepleri karşılayabilir olmalıdır. Doğal olarak izole edilmiş bir kavşakta tek başına verimli

çalışmak ve bir sistem olarak verimli çalışmak iki farklı durumdur. Adımın amacı, birlikte koordine edilmiş bir birim olarak çalışması gereken sinyalle kavşak gruplarını belirlemektir. Gruplandırma sinyallerinin kısıtı, bir gruptaki tüm kontrol cihazlarının aynı döngü uzunluğunda çalışması gerektiğidir. Farklı kavşaklar için döngü uzunluğu gereksinimlerinin her zaman aynı olmaması muhtemeldir. Koordine işleminin dezavantajı, gruptaki kesişmelerin bazılarının verimsiz bir döngü uzunluğunda çalışmasıdır. Ortaya çıkabilecek bu dezavantaj, koordineli işlemde elde edilen faydalarla dengelenmek zorunda değildir.

1.5.8. Koordinasyon parametrelerinin hesaplanması

Her bir faza ait parametreler hesaba katıldığında toplam parametre sayısı yüzün üzerinde gibi görünse de aslında koordinasyon parametrelerinin sayısı, döngü uzunluğu, ofset ve faz uzunluğu ile sınırlıdır. Her zamanlama planı için bu parametrelerin bir kombinasyonu bulunmaktadır.

Bu parametreler kullanıcı tarafından manuel olarak ve bir modele bağlı olmak üzere iki şekilde belirlenmektedir. Burada modelden kasıt optimizasyon algoritmasıdır.

Minimum maliyet yaklaşımı, bazı model giriş parametrelerinin tahmin edilebileceğini varsaymaktadır. Buradaki parametrelerin her şart altında ölçülmesi veya izlenmesi önemlidir. Çoğu yazılım bazı zamanlama parametrelerini sabit tutarken, diğer bir parametrenin optimize edilmesini sağlamaktadır. Örneğin küçük kavşak noktalarında sola dönüş fazı gibi kısa fazların süreleri manuel olarak belirlenebilir ve daha sonra modele girilebilir.

Bilgisayar destekli bir kavşak optimizasyon tekniği kullanmanın avantajları şu şekilde sıralanabilir:

1. Hem manuel hem de bilgisayarlı yöntemler için gereken girişlerin çoğu ağın doğru tanımlanması ile ilişkilidir. Bu tanımlamalar sinyal fazlama, link mesafeleri, araç hızları ve kavşakların geometrisi gibi parametreleri içermektedir. Bilgisayarlı yaklaşımla bu bilgiler, girdide nispeten az bir değişiklik yapılarak yeni zamanlama planları oluşturmak için kolayca kullanılabilir.
2. Veriler birçok senaryo için simüle edilebilir.

3. Veri dosyaları hem girdi hem de çıktı için belgeler sağlamaktadır.

1.5.9. Yeni planın uygulaması ve değerlendirilmesi

Sürecin son aşaması, üretilen yeni zamanlama planlarını kurmak ve değerlendirmektir. Yeni zamanlama planlarının değerlendirilmesinde, durma zamanının geciktirilmesi ve seyahat süresinin azaltılması olmak üzere iki temel analitik prosedür bulunmaktadır. Bu iki analitik prosedür simülasyon ortamında değerlendirilerek faz süreleri belirlenmektedir. Tez dahilinde geliştirilen uygulamada bu aşamalar kullanılmış ve yeni zamanlama planları oluşturulmuştur.

1.6. Trafik Simulasyon Modelleri

Bölüm kapsamında trafik simülasyon modelleri hakkında genel bir bilgilendirme yapılmıştır. Doğal olarak mevcut tüm trafik simülasyon modellerinin tam bir sentezinin yapılması tez kapsamında olmadığından, trafik ataması ile ilgili noktalarda ve bu modeller içindeki kullanıcı dengelerinin hesaplanması konusunda daha ayrıntılı bilgiler verilmiştir.

Trafik ataması açısından trafik simülasyon modelleri, tek tek araçları ayırt etmeyen akış bazlı (makroskobik) modeller (trafiğin bir tür 'akışkan' olarak tanımlandığı modeller), tek tek araçların ağ üzerinden hareketini tanımlayan araç odaklı (mikroskobik) modeller ve araçları tek tek tanımlayan fakat toplu genel sonuç üreten (mezoskopik) modeller olarak üç ana sınıfa ayrılabilir.

1.6.1. Makroskobik Modeller

Trafik ataması, güzergâh seçiminin bir modelde tanımlanabileceği ilkesine dayanır. Bu bakımdan makroskobik modelde, modeldeki her bir değişken için hız alanı $v(x, t)$ ve $\rho(x, t)$ yoğunluğu gibi her yol için karşılık gelen değişkenlerin eklenmesi gerekir. Bu nedenle analitik dinamik trafik ataması modellerinde genellikle oldukça basit bir trafik akışı modeli kullanılmaktadır.

Trafik akışının, akışkanlar dinamiği ile açıklanmasını sağlayan ve süreklilik denkleminden oluşan Lighthill-Whitham modeli Denklem 24 ile ifade edilmektedir (Lighthill ve Whitham, 1955).

$$\frac{\partial}{\partial t} \rho + \frac{\partial}{\partial x} \rho v = 0 \quad (24)$$

KW denklemini veren $v = V(\rho)$ hız yoğunluk ilişkisi ile birlikte oluşan yapı Denklem 25 ile ifade edilmektedir.

$$\frac{\partial}{\partial t} \rho + c(\rho) \frac{\partial}{\partial x} \rho v = 0 \quad (25)$$

Burada KW'ların hızı olan c ρ değeri Denklem 26 ile ifade edilmektedir:

$$c \rho = \frac{d}{d\rho} \rho V(\rho) \quad (26)$$

Lighthill-Whitham modeliyle ilgili en büyük sorun trafiğin her zaman dengede olması değil, sürücülerin hızlanmalarını destekleyerek tepki vermeleri gerektiğidir. Problemde yerel hızın zamana göre türevi, yani ivme Denklem 27 ile ifade edilebilir.

$$\frac{d}{dt} v \text{ } x, t \equiv \frac{\partial}{\partial t} + v \frac{\partial}{\partial x} = \frac{v}{\tau} \rho - \frac{c^2}{\rho} \frac{\partial \rho}{\partial x} \quad (27)$$

Denklemin sağ tarafında, birinci terim denge hızına bir azalma tarif ederken, beklenti terimi olarak adlandırılan ikinci terim, sürücünün önündeki yoğunluk artarsa yavaşladığını açıklamaktadır. Denklem 26'nın çözümleri süresiz şok dalgaları geliştirme eğilimindedir. Bu nedenle Kühne, süresizliğin oluşumunu önlemek için akışkan bir terim eklemeyi önermiştir (Kühne, 1984). Önerilen terim, Denklem 28'de ifade edilmiştir:

$$\frac{\partial}{\partial t} + v \frac{\partial}{\partial x} = \frac{v}{\tau} \rho - \frac{c^2}{\rho} \frac{\partial \rho}{\partial x} + \frac{1}{\rho} \frac{\partial}{\partial x} \mu \frac{\partial v}{\partial x} \quad (28)$$

Benzer modeller Kerner ve Konhäuser (Kerner ve Konhäuser, 1994) ile Helbing (Helbing, 1995) tarafından da önerilmiştir.

1.6.2. Mikroskobik Modeller

Mikroskobik yani araç yönelimli modellerde diğer bir ifade ile Dinamik Trafik Ataması (DTA) algoritmasında başlangıç, varış, güzergâh ve kalkış saatleri, modelin veri yapılarına kolayca eklenebilir.

Araç takibi yapan modeller ile mezoskopik modeller arasındaki fark doğrudan DTA ile ilgili olmasa da birçok çalışmada ilişkili kabul edildiği için bundan sonraki bölümde bu kabul sürdürülecektir.

1.6.2.1. Araç Takip Modelleri

Birçok kişinin araç sürmeyi bilmesine rağmen, hala benzersiz bir araç takip davranışı modeli türetebilecek genel kabul görmüş temel bir prensip bulunmamaktadır. Günlük deneyimlerden elde edilen bazı davranışlar şu şekilde sıralanabilir:

- 1- Çoğu zaman çarpışma yaşanmaz.
- 2- Maksimum hız, hızlanma ve yavaşlama otomobillerin ve sürücülerin fiziksel sınırlamaları ile sınırlıdır.
- 3- Sürücüler çoğu zaman sabırsız davranır ve 'birinci dereceden yaklaşımda' sadece önündeki araca tepki verir.
- 4- Sürücülerin harekete geçmesi için zaman gereklidir.

Ancak deneyimlere rağmen birçok farklı modelleme yaklaşımı da bulunmaktadır. Genellikle bu yaklaşımlarda tek şeritli bir yolda, $i + 1$. aracın i . aracın önündeki araç olduğu şeklinde numaralandırılmış N adet aracın olduğu varsayılmaktadır. Araç takip modellerinden bazıları Gecikmeli Diferansiyel Denklemler, Hücresel Otomatlar ve Bağlantılı Haritalardır.

1.6.2.1.1. Gecikmeli Diferansiyel Denklemler

Sürücülerin tepkime sürelerinin reaksiyon süreleri olan Δt ile ertelenmesini, gecikmeli diferansiyel denklem olarak sürüş davranışının modellenmesi Denklem 29'da olduğu gibi Gawron tarafından önerilmektedir (Gawron, 1998):

$$\frac{dv_i}{dt} = f(v_i, v_{i+1}, \Delta x_i) \big|_{t-\Delta t} \quad (29)$$

Burada $\Delta x_i = x_{i+1} - x_i$ olarak ifade edilmektedir. Aslında, bu tür birçok model 1950'li yıllarda önerilmiştir. Gazis, Herman ve Rothery formun Denklem 30'daki modelini önermişlerdir (Gazis vd., 1961).

$$\frac{dv_i}{dt} = \alpha v_i^m t \frac{v_{i+1}-v_i}{(x_{i+1}-x_i)^l} \quad (30)$$

Burada α , l , m değerleri modelin ölçekleme parametreleridir. Modellerin avantajı, model denklemlerinin hız-yoğunluk ilişkisini belirlemek için doğrudan sisteme entegre edilebilir olmasıdır.

Bununla birlikte, bu modellerin insan davranışı çalışmalarında herhangi bir temeli yoktur. Wiedemann algılama eşiklerine dayanan gecikmeli bir diferansiyel denklem modelinde mesafelerin ve hız farklarının algılanmasındaki fizyolojik sınırlamaları önermektedir (Wiedemann, 1974). Bu yaklaşım PELOPS gibi sürücü-araç sisteminin karmaşık ve oldukça ayrıntılı modelleri için daha da geliştirilmiştir (Ludmann vd., 1996).

DTA uygulamaları için kullanılan bu modeller hesaplama açısından diğer modellere göre daha maliyetlidir.

1.6.2.1.2. Hücresel Otomatlar ve Bağlantılı Haritalar

Gecikmeli diferansiyel denklem modellerinin en büyük dezavantajı düşük hesaplama performansıdır. Sorunun üstesinden gelmek için yaygın bir yaklaşım, bu tür modellerin genellikle reaksiyon zamanı (bir saniye ara ile) olarak seçilen sabit bir zaman aşaması ile ayrıklaştırılmasıdır. Bağlantılı haritalar seti ise, tüm araçlara aynı anda uygulanan formun bir güncelleme kuralını vermektedir. Bununla birlikte, verilen bir diferansiyel denklemin ayrıklaştırılması yerine, bu tür eşlenmiş haritalar Denklem 31’de gösterildiği gibi doğrudan tasarlanabilir.

$$v_i(t + \Delta t) = f(v_i(t), x_i(t), v_{i+1}(t), x_{i+1}(t)) \quad (31)$$

Gecikmeli diferansiyel denklem modellerinin arkasındaki fikir, araç dinamiğinin mümkün olduğunca ayrıntılı bir şekilde tanımlanmasını sağlamaktır. Bağlantılı harita modelleri için ilginç olan sorun trafik akışının temel özelliklerini korurken, bir modelin ne kadar minimalist olabileceğidir (Gawron, 1998). Bu tür modellerin ilkleri Cremer ve Ludwig (Cremer ve Ludwig, 1986) ve Schütt (Schütt, 1991) tarafından önerilmiştir. Ancak en göze çarpan hücresel otomat modeli, birçok yazar tarafından ayrıntılı olarak incelenen Nagel-Schreckenberg modelidir (Nagel ve Schreckenberg, 1992), (Nagel,

1995; Nagel ve Paczuski, 1995; Ponzlet ve Wagner, 1996; Schadschneider ve Schreckenberg, 1993).

Nagel-Schreckenberg modelinde, linkler Δx uzunluktaki hücrelere ayrılır. Yoğun sıkışmalarda (kilometrede yaklaşık 130 araba) araç yoğunluğunun ölçülmesi, $\Delta x = 7.5 m$ 'nin makul bir değer olduğunu göstermektedir. Sonrasında, “doğal” birimler olan Δx ve Δt değerleri sisteme dahil edilmez. Burada aynı zamanda $\Delta x/\Delta t$ nin değeri olan hız da kullanılmamaktadır. Kullanılan değişkenler ise şu şekildedir:

$$\textbf{İvmelenme (Acceleration): } v_i(t + \Delta t) := \min\{v_{max}, v_i(t) + 1\} \quad (32)$$

$$\textbf{Yavaşlama (Deceleration): } v_i(t + \Delta t) := \min\{v_i(t + \Delta t), x_{i+1}(t) - x_i(t)\} \quad (33)$$

Umarsız Sürüş (Dawdling): $p_{frenleme}$ olasılığı ile

$$v_i(t + \Delta t) := \max\{v_i(t + \Delta t), 0\} \quad (34)$$

$$\textbf{Yer Değiştirme (Translation): } x_i(t + \Delta t) := x_i(t) + \Delta t v_i(t + 1) \quad (35)$$

Güncelleme kuralları her araç için paralel olarak uygulanır. Bu model tam sayılı işlemler kullandığından ve tek bitli kodlama kullanılarak bile gerçekleştirilebildiğinden hesaplama çok verimli olmaktadır (Nagel, 1995).

Oluşan yapı basit bir sürüş davranışı tanımlamasına rağmen, modelin trafik akışının temel özelliklerini yeniden net şekilde ürettiği görülmektedir. Örneğin:

- Belli bir yoğunluğun üzerinde, trafik sıkışıklığı kendiliğinden meydana gelmektedir.
- Yoğunluk-akış ilişkisi niteliksel olarak doğrudur.
- Trafik ışıklarından geçen iki araç arasındaki zaman aralığı ve sinyalize kavşakların kapasitesinin kritik parametresi arasındaki bağıntı doğru şekilde modellenmektedir.

Nagel-Schreckenberg modelinde iki önemli eksiklik vardır. İlki, modelin ayrık doğası gereği kirletici emisyonlarını kolayca modelleyememesidir. İkincisi, Nagel-

Schreckenberg modelinin ölçümlerde gözlemlenen yarı kararlı durumları ve diğer karmaşık davranış türlerini üretememesidir (Kerner ve Rehborn, 1997, 1996a, 1996b).

Açıklanan nedenlerden dolayı Krauß (Krauß, 1998, 1997) bölümün başında belirtilen dört şartı sağlayan minimal bir model olarak görülebilecek bir modeli tanımlamış ve sürüş davranışındaki kusurların hızın stokastik dalgalanmaları olarak modellenebileceği varsayımını önermiştir. Maksimum yavaşlama b dikkate alındığında, emniyetli hız (v_{eh}) Denklem 36'daki gibi ifade edilebilir:

$$v_{eh} = v_{i+1} + \frac{x_{i+1} - x_i - v_{i+1} \Delta t}{\frac{v_{i+1} + v_i}{2b} \Delta t} \quad (36)$$

Krauß bu güvenli hız kavramını kullanarak Nagel-Schreckenberg modelinin güncelleme kurallarını genişletmiştir. Yeni sistem değişkenleri ise Denklem 37, 38 ve 39'teki şekliyle güncellenmiştir:

$$\textbf{Hız Güncellemesi (Speed Update): } v_i(t + \Delta t) := \min\{v_{max}, v_i(t) + a, v_{eh}\} \quad (37)$$

$$\textbf{Umarsız Sürüş (Dawdling): } v_i(t + \Delta t) := \max\{v_i(t + \Delta t) - a\varepsilon\xi, 0\} \quad (38)$$

$$\textbf{Yer Değiştirme (Translation): } x_i(t + \Delta t) := x_i(t) + \Delta t v_i(t + 1) \quad (39)$$

Burada $\xi \in [0, 1]$ aralığındaki üniform dağılımlı rastgele değişken ve ε hızlanmadaki 'gürültünün' genliğini kontrol eden bir parametredir. Nagel-Schreckenberg modelinde olduğu gibi Δt zaman adımı genellikle 1. adım olarak alınır.

Krauß, azami ivmelenmeye ve azami yavaşlamaya bağlı olarak modelin, farklı yapı oluşum tipleri ile üç alt sınıfa ayrıldığını göstermiştir (Janz, 1998; Krauß, 1998): Birinci sınıf Nagel-Schreckenberg modeline benzer davranışlar gösterirken, ikinci sınıf gerçek dünya verilerinde gözlemlenen değişken durumları göstermiş, üçüncü sınıf ise hiçbir yapı oluşumu göstermeyip, gerçekçi parametrelere sahip bir model olarak yarı kararlı durumları gösteren sınıfa aittir.

1.6.3. Mezoskopik Modeller

Sürücü odaklı modellerin DTA için ana avantajı, bireysel sürücüleri takip etme ve yönlendirme bilgilerini onlarla ilişkilendirme yeteneğidir. Araç takip davranışının tanımı teorik bakış açısından ve emisyonların modellenmesi gibi bazı uygulamalar için ilginç olsa da DTA için gerekli olmamaktadır. Sürücü yönelimli olan ancak davranış sonrasında araç modellemeyen mezoskopik modeller bu nedenle DTA uygulamalarında kullanım için ideal modellerdir (Gawron, 1998).

Mezoskopik modeller, makroskopik modellerin toplam seviye yaklaşımı ile mikroskopik modellerin bireysel etkileşimleri arasındaki boşluğu doldurmaktadır. Mezoskopik modellerde genellikle trafiğe ilişkin varlıklar yüksek bir ayrıntı düzeyinde tanımlanıp, trafik varlıklarının davranışları ve etkileşimleri daha düşük bir ayrıntı düzeyinde açıklanmaktadır.

Mezoskopik modeller çeşitli biçimlerde olabilmektedir. Örneğin, araçlar ağ üzerinde toplu hareket eden yapılar olarak nitelendirilebilir (Leonard vd., 1989). Araç grupları tek bir varlık gibi davranır ve her linkin hızı, o link için tanımlanan hız yoğunluğu fonksiyonundan ve giriş anındaki linkin yoğunluğundan türetilmektedir. Bir linkteki yoğunluk, her şerit için kilometre başına düşen araç sayısı olarak tanımlanır. Hız-yoğunluk fonksiyonu ise yoğunluğu oluşturan linkteki araçların hızını ilişkilendirir. Linkte çok fazla trafik varsa (yani yoğunluk çok yüksek ise), hız-yoğunluk fonksiyonu araçlarda düşük hıza, düşük yoğunluk ise yüksek hızlara neden olmaktadır. Mezoskopik modellerde şerit değişiklikleri ve araçların hızlanması/yavaşlaması modellenmemektedir.

Bir diğer mezoskopik modelde, linkler hücrelere bölünür ve araçlar kendi davranışlarını kontrol eden hücreler halinde hareket eder. Araç olarak nitelenen hücreler linklerden geçer ve araçlar gerektiğinde hücrelere girip çıkabilir, ancak sollama yapamaz. Araçların hızı bireysel sürücülerin kararları tarafından değil hücrenin durumu ile belirlenir. Örneğin, aracın bulunduğu hücrenin önünde bulunan hücre dolu ise araç yavaşlar veya durur (Ben-Akiva, 1996).

Alternatif olarak yolun kuyruklanmış ve akan bölümler olarak modellendiği, kuyruk sunucusu (queue-server) yaklaşımı adı verilen mezoskopik modeller bulunmaktadır. Kuyruk sunucusu yaklaşımında şeritler ayrı ayrı modellenenirse de genellikle tercih edilmemektedir. Araçlar ayrı ayrı temsil edilmelerine ve bireysel hızlarını korumalarına rağmen, davranışları ayrıntılı olarak modellenmemektedir. Araçlar linkin akan kısmında makroskopik bir hız yoğunluğu fonksiyonu ile belirlenen bir hızda hareket eder ve linkin sonunda bir kuyruk sunucusu araçları bir sonraki linke yönlendirir (Gawron, 1998; Jayakrishnan vd., 1994). Kuyruk sunucusu yaklaşımı, mikroskopik trafik akışı modellemesinin avantajlarını (araçlar ayrı ayrı modellendiği için), kalibrasyon kolaylığı ve makroskopik hız/yoğunluk ilişkilerinin kullanımı ile birleştirir. Kavşak sunucularındaki kapasiteler doygunluk akışlarından ve varyanslarından (ölçülen veya hesaplanan) kaynaklanır. Sinyal kontrollü kavşaklar, kuyruk sunucularını sinyal kontrolünün durumlarına (yeşil/sarı/kırmızı) göre açılan ve kapanan kapılarla değiştirerek modellenenmektedir. (Burghout, 2004)

Adaptif sinyal kontrolünün modellenmesi daha zordur, çünkü linklerdeki araçların konumları bilinmemektedir. Bu nedenle sinyal kontrolüne bağlı detektörleri ne zaman geçtiklerini bilmek zordur. Bireysel araçların temsil edilmesinin bir başka avantajı, ayrıştırılmış güzergâh seçiminin modellenmesi olasılığıdır. Bu durum güzergâh değişikliklerinin modellenmesi gerektiğinde, örneğin sürücülerin güzergâhlarına karar vermelerine yardımcı olan akıllı ulaşım sistemleri değerlendirilirken önemli yer teşkil etmektedir. Belirtilen sebeplerden dolayı tez kapsamında mikroskopik simülasyonlar kullanılmıştır.

BÖLÜM 2

YÖNTEM ve MATERİYAL

Materyal bölümünde teze konu olan problemin çözümü için kullanılan VISSIM trafik simülasyon programı ile ilgili temel ve teknik bilgilerden bahsedilecektir.

2.1. Kullanılan Yazılımlar

Bu tez kapsamında simülasyonlar için PTV VISSIM 11.00-11 programından yararlanılmıştır. VISSIM çok modelli taşıma işlemlerinin modellenmesinde önde gelen mikroskobik simülasyon programıdır. Tüm yol kullanıcılarını ve etkileşimlerini tek bir modelde görüntüleyebilmektedir. Ayrıca bilimsel olarak başarılı olan hareket modelleri ile tüm yol kullanıcılarının gerçekçi bir modellemesini sağlamaktadır.

VISSIM programına birçok programlama dili ile müdahale etmek mümkündür. Bunun için yine VISSIM modülü olan COM arayüzü kullanılmaktadır (Bölüm 5.1.1.2.'de COM arayüzü ile ilgili ayrıntılı bilgi verilmektedir.). Programa gereken müdahaleyi yapmak için ise MathWorks tarafından geliştirilen MATLAB 2017b programlama dilinden yararlanılmıştır. MATLAB matris işlemleri ve genel olarak sayısal hesaplamalar üzerinde durulan gelişmiş bir programlama dilidir.

Bu programların veri alışverişinde bulunması için Microsoft Windows 10 işletim sistemi kullanılmaktadır.

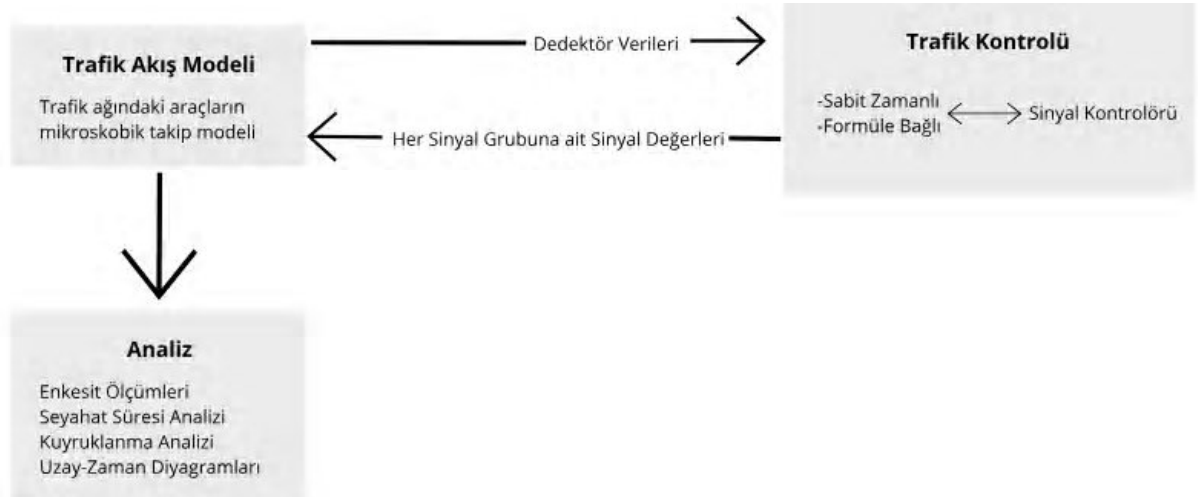
2.1.1. VISSIM

PTV VISSIM, 1992'den bu yana Almanya'nın Karlsruhe kentinde PTV (Planung Transport Verkehr AG) firması tarafından geliştirilen mikroskobik çok modüllü trafik akışı simülasyon yazılım paketidir. Adı "Verkehr In Städten - SIMulationsmodell" ("Şehirlerde Trafik Simülasyon Modeli") kelimelerinden türetilmiştir (PTV AG, 2018).

VISSIM, şerit konfigürasyonu, araç çeşitleri, trafik sinyalleri vb. kısıtlamalar altında özel ve toplu taşıma operasyonlarını analiz ederek, ulaştırma mühendisliği ve trafik planlama önlemlerine dayalı çeşitli alternatiflerin değerlendirilmesini mümkün kılmaktadır. Ayrıca yaya akışları, özel araçlar ve toplu taşıma bir bütün olarak modellemeye olanak sağlamaktadır.

2.1.1.1. Trafik Simülasyon Modeli

VISSIM simülasyon paketi, dahili olarak detektör verisi ve sinyal durumunu bir arayüz üzerinden ayarlayan iki farklı parçadan oluşmaktadır. Simülasyon, trafik işlemlerinin çevrimiçi olarak görselleştirilmesini sağlayıp; seyahat süreleri ve sıra uzunlukları gibi istatistiksel verileri toplayan çıktı dosyalarının oluşturulmasını çevrimdışı hale getirmektedir. Trafik simülatörü, aracı takip eden ve şerit değiştirme mantığını içeren mikroskobik bir trafik akışı simülasyon modelidir. Sinyal durumu üretici ise, trafik simülatöründen belirli bir zaman adımı temelinde (saniyenin 1 / 10'una kadar) detektör bilgisini sorgulayan bir sinyal kontrol yazılımıdır. Şekil 8'de görüldüğü gibi sinyal durumu üretici her zaman adımı için sinyal durumunu belirler ve bu bilgiyi trafik simülatörüne geri iletir.



Şekil 8. Trafik Simülatörü ve Sinyal Durumu Üretici Arasındaki İletişim

VISSIM, araçların bireysel davranışlarına dayanan mikroskobik bir yol trafik simülatörüdür. Mikroskobik modelleme yaklaşımının amacı trafik dinamiklerinin doğru tanımlanmasıdır. Böylece, simüle edilmiş trafik ağı ayrıntılı olarak analiz edilebilir hale

gelmektedir. Simülatör, aslında Wiedemann (1974) tarafından geliştirilen sözde psiko-fiziksel sürücü davranış modelini kullanmaktadır. Psikolojik yönlerin ve sürücünün algısının fizyolojik kısıtlamalarının birleşimi nedeniyle, modele psiko-fiziksel araç takip modeli adı verilmektedir.

VISSIM, herhangi bir yol ağı tipinin geometrisinin tasarlanabildiği ve simülasyonların basit bir şekilde kurulabildiği bir grafik arayüzü sunmaktadır. Bununla birlikte, birkaç problem için grafik arayüzü tatmin edici değildir. Örneğin kullanıcı simülasyon sırasında dinamik olarak VISSIM nesnelere erişmeyi ve bunları manipüle etmeyi hedeflediğinde söz konusu durum oluşmaktadır. Bu amaçla, yazılımlar arasında süreçlerin iletişimini sağlayan bir teknoloji olan COM'a dayanan ek bir arayüz sunulmaktadır (Bölüm 2.1.1.3.'de COM arayüzü ile ilgili ayrıntılı bilgi verilmektedir.). VISSIM COM arayüzü, başlangıçta grafik arayüzü tarafından sağlanan simülatörün işlev ve parametrelerinin programlama ile manipüle edilebildiği hiyerarşik bir model tanımlamaktadır. COM, hiyerarşik nesnelerini (Bknz. Bölüm 2.1.1.3.) işleyebilecek birçok dilde (örneğin, MATLAB, C ++, Python, Java, vb.) programlanabilmektedir. VISSIM COM aracılığıyla kullanıcının, dahili nesnelerin çoğunun niteliklerini dinamik olarak değiştirebilmesine imkân sağlanmaktadır.

2.1.1.2. VISSIM ile ilgili Teknik Bilgiler

VISSIM programı üzerinde simülasyon yapabilmek için bazı ön koşul parametrelerinin programa yüklenmesi gereklidir. Bu bölümde kısaca ön koşul parametrelerinden bahsedilecektir.

2.1.1.2.1. Rastgele Tohumlama (Random Seed)

VISSIM'da üretilen ağ modellerinde simülasyon yapılırken üretilen araçlar belirli bir matematiksel fonksiyona göre üretilmektedir. Bu fonksiyonlar ise rastgele sayıları temel alırlar. Buna rastgele tohum adı verilmektedir. Fakat en iyi sinyal sürelerinin bulunması gibi uygulamalarda testlerin doğru olması için, her simülasyonda araçların aynı formül ile üretilmesi gereklidir. VISSIM'da üretilen her rastgele tohum için bir fonksiyon numarası bulunmaktadır. Bu fonksiyon numaralarının tümü doğal sayıdır ve en fazla 4.294.967.295 olabilmektedir. Fakat MATLAB üzerinde tanımlanabilen en büyük doğal

sayı 2.147.483.647 olduğundan en büyük rastgele tohum fonksiyonu bununla sınırlı olmaktadır.

2.1.1.2.2. Eriyod ve Simülasyon Çözünürlüğü

Tez sırasında kullanılan VISSIM lisansı akademik lisans olarak erişime açılmıştır. Bu durum simülasyonların en fazla 1 saatlik dilimde yapılabilmesine imkân vermektedir. VISSIM ise periyod süresini saniye cinsinden ölçeklemektedir. Dolayısıyla bu süre Akademik Lisans için 3600 saniye olarak sınırlandırılmıştır.

Simülasyon çözünürlüğü saniyedeki adım miktarıdır. Simülasyon çözünürlüğü zaman adımlarının sayısını belirtmektedir. Bir aracın hareketi saniyede 1 ile 20 defa arasında örneklenebilmektedir. Değerin 1-5 aralığında olması durumunda simülasyon düzensizleşmektedir. Bu değer aralığı sadece tasarım esnasında sistemin çalışıp çalışmadığını kontrol etmek üzere kullanılmaktadır. Çözünürlük değerinin 5 ile 10 arasında olması simülasyonun gerçeğe yakın olmasını sağlamaktadır. Bu değer aralığı ise genellikle trafik sinyal optimizasyonu projelerindeki sonuçların üretilmesi için kullanılmaktadır. Çözünürlüğün 10 ile 20 arasında olması sürücülerin uzman olması şeklinde yorumlanabilir.

Simülasyon çözünürlüğü araç davranışlarına doğrudan etki etmektedir. Bu yüzden farklı çözünürlüklerde farklı sonuçlar elde edilmektedir.

VISSIM, sadece araçlar için hızlanma ve yavaşlama değeri kullanmaz, bunun yanında sürücünün davranışındaki farklılıkları temsil etmek için farklı parametreler de kullanır. Hızlanma ve yavaşlama mevcut hızın işlevleridir. Örneğin içten yanmalı motorlar maksimum ivmelenmeye düşük hızlarda ulaşırken üç fazlı motorlar, tramvaylar veya trenler, geniş bir hız aralığında sabit bir hızlanma gösterir.

Simülasyon ortamında her araç tipi için iki hızlanma fonksiyonu ve iki yavaşlama fonksiyonunun atanması gerekir. Bunlar:

Maksimum İvmelenme (Max. Acceleration): Teknik olarak uygulanabilir maksimum ivmelenmeyi ifade eder. Araçlar ancak, iniş eğimli bölgeler gibi hızı arttıran kısımlarda beklenen hızı aşip maksimum ivmelenmeye ulaşabilir.

Beklenen ivme (Desired Acceleration): Link üzerinde ortalama beklenen hıza göre araçların gösterdiği davranış olarak belirtilebilir. Simülasyon üzerindeki her araç bulunduğu linkin ortalama beklenen hızına göre hızlanır veya yavaşlar.

Maksimum Yavaşlama (Max. Deceleration): Teknik olarak uygulanabilir maksimum yavaşlamayı ifade eder. Eğimler üzerinde, her bir pozitif gradyan yüzdesi için $0,1 \text{ m} / \text{s}^2$ ve her bir negatif gradyan için $-0,1 \text{ m} / \text{s}^2$ ayarlanmıştır.

Beklenen Yavaşlama (Desired Deceleration): Yavaşlama maksimum yavaşlama değerinden düşükse, beklenen yavaşlama anlık olarak maksimum yavaşlama değeri yerine kullanılır. Beklenen yavaşlama durumu aşağıda belirtilen koşullar için geçerlidir:

- Beklenen hız kuralının değişmesi durumu,
- Dur-Kalk trafik durumlarında ve aracın öndeki araca fazla yaklaşması durumu,
- Şerit içinde sollama için yetersiz bir yanıl mesafe olması durumu
- Acil durmanın gerektirdiği durumlar, (Kaza gibi)
- Kooperatif frenlemesi Durumu: Bu durumda, aracın %50'sinin istediği yavaşlama, sinyal veren bir aracın komşu şeritten aracın şeridine geçip geçmeyeceği konusunda karar için azami makul yavaşlama olarak kullanılır.

Bunun dışındaki her durumda araç takip modeli esas alınmaktadır.

2.1.1.2.3. VISSIM'da Varsayılan Maksimum İvmelenme-Yavaşlama Eğrileri

Araçlarda maksimum ivmelenme eğrileri için, Wiedemann 74 modeli baz alınmaktadır. Otomobiller için, 1974'ten önce Almanya'da Wiedemann ölçümleri, kısa süreli simülasyon adımları için jerk sınırlaması ve kullanıcı tarafından gelen (min-max) değer farkı kullanılarak geliştirilmiştir. Jerk, hızlanma değişiminin oranı olarak ifade edilir ve ivmenin zamana göre türevidir. Buna ek olarak ölçümler sürecinde, saniyede 2'den fazla olan her zaman adımı, bu zaman adım uzunluğunun iki katına karşılık gelen bir pay değeri ile sınırlandırılmaktadır. Bu sınırlandırmaya jerk sınırlandırması adı verilir. Örneğin; Saniyede 10 zaman adımı (zaman adımı = 0.1 s) ile sınırlama, hızlanmada

beklenen deęişimin %20'sidir (0.2). Her araç için maksimum hızlanma, gradyandan řu řekilde etkilenir:

- %1 hızlanmalarda her gradyan başına $0.1 \text{ m} / \text{s}^2$ azalır.
- Yavaşlamalarda %1 gradyan başına $0.1 \text{ m} / \text{s}^2$ artar.

Kamyonlar için gerçek hızlanma, istenen hızlandırma fonksiyonu ile de sınırlandırılmıştır; bu nedenle kamyonlar için aşırı derecede yüksek hızlanma deęerleri, dik yokuřlar için sadece çok düşük hızlarda geçerlidir.

Bu süreçte araçların konumlanmaları belirlenirken rastgele sayılardan yararlanılır. Bu rastgele deęerler normal dağılıma sahiptir. Normal dağılım [0,1] arasında deęer alır ve ortalaması 0.5 ve Standart Sapması (SS) 0.15 olarak kullanılmaktadır. Bu nedenle ortalama eğri ile minimum/ maksimum eğri arasındaki mesafe SS'nin 3.333 katı olmaktadır. Bu da VISSIM'da doğrusal enterpolasyonla, araçların %68'inin rastgele deęerlerin [-1 SS, + 1 SS] aralığında, %95'inin ise [-2 SS, +2 SS] aralığında oluşmasını sağlamaktadır.

VISSIM, 0.5 deęerinin altındaki rastgele sayılar için minimum (0.0) ile ortalama eğri (0.5) aralığında ve 0.5 deęerinin üstündeki rastgele sayılar için ortalama eğri ile maksimum eğri (1.0) arasındaki deęerleri enterpole etmektedir.

2.1.1.2.3.1. Wiedemann Yaklaşımı

VISSIM'deki trafik akış modeli, sürücü, araç ve dięer birimleri tek bir varlık olarak kabul eden ayrık, stokastik ve zaman adımına dayalı olan mikroskobik bir modeldir. Model, psiko-fiziksel davranış gösteren bir aracın ileriye doğru hareketi için belirli bir fonksiyonel modeli takip ettięi, yanal hareketler için kural tabanlı modeli takip ettięi bir algoritma içermektedir. Model Wiedemann'ın çalışmalarına dayanmaktadır (Wiedemann, 1974).

Wiedemann modelinin temel fikri, ařağıdaki sürücünün dört sürüş modundan herhangi birinde olabileceęi varsayımdır:

Serbest Sürüş (Free Driving): Öndeki araçların etkisi gözlemlenemez. Bu modda sürücü, bireysel olarak istenen hız olarak belirledięi belirli bir hıza ulaşmayı ve

korumayı amaçlamaktadır. Gerçekte serbest sürüş sırasındaki hız, mükemmel olmayan gaza basma kontrolü nedeniyle sabit tutulmayıp, istenen hız etrafında salınım göstermektedir.

Yaklaşma (Approaching): Sürücünün kendi hızını, önceki bir aracın düşük hızına uyarlama işlemi olarak tanımlanmaktadır. Yaklaşma durumunda sürücü, iki aracın hız farkının gerektirdiği güvenlik mesafesine ulaştığı anda aradaki mesafe sabit olacak şekilde yavaşlama davranışı gösterir.

Takip (Following): Sürücü önceki aracı herhangi bir bilinçli hızlanma veya yavaşlama olmadan izler. Güvenlik mesafesini az ya da çok sabit tutma eğilimi göstermektedir. Bununla birlikte kusursuz gaza basamama ve yanlış tahmin nedeniyle hız farkı sıfır civarında salınım yapmaktadır.

Frenleme (Braking): Aradaki mesafe istenen güvenlik mesafesinin altına düşerse, orta ile yüksek yavaşlama oranları uygulaması olarak tanımlanmaktadır. Öndeki araba ani bir şekilde hızını değiştirirse veya üçüncü bir araba gözlemlenen sürücünün önünde şerit değiştirirse gerçekleşebilmektedir.

Her mod için hızlanma; hız farkı, mesafe ve sürücü ile aracın bireysel özelliklerinin bir sonucu olarak açıklanmaktadır. Sürücü, hız farkı ve mesafenin bir kombinasyonu olarak ifade edilebilecek belli bir eşiğe ulaştığı anda bir moddan diğerine geçiş yapar. Örneğin küçük bir hız farkı sadece küçük mesafelerde gerçekleştirilebilir, aksi durumda büyük hız farkları yaklaşan sürücüler için çok daha erken tepki vermeye zorlamaktadır. Hız farklarını algılama ve mesafeleri tahmin etme kabiliyeti sürücü popülasyonunun yanı sıra, beklenen hızlar ve emniyet mesafeleri arasında değişim göstermektedir.

Wiedemann araç takip modeli Wiedemann 74 ve Wiedemann 99 modelleri olmak üzere ikiye ayrılmaktadır. Wiedemann 74 modeli şehir içi trafiği için, Wiedemann 99 modeli ise şehirlerarası trafik modellemeleri için geliştirilmiştir. Bu tezdeki çalışma alanları şehir kavşakları olduğundan Wiedemann 74 modeli üzerinde durulacaktır.

2.1.1.2.3.2. Wiedemann 74 Modeli Parametreleri

Ortalama durma mesafesi (ax) durdurulan araçlar arasında olması istenen ortalama mesafeyi tanımlar. -1.0 m ile +1.0 m arasında değişen, 0.0 m ortalama ve 0.3 m standart

sapma ile normal dağılıma uymaktadır. İstenen emniyet mesafesinin ilave kısmı (bx_{ilave}) ve istenen emniyet mesafesinin çarpım kısmı ($bx_{çarpım}$) emniyet mesafesinin hesaplanmasına etki eden faktörlerdir. İki araç arasındaki d ortalama mesafesi Denklem 40'taki formülle kullanarak hesaplanır:

$$d = ax + bx \quad (40)$$

Burada ax durma mesafesidir. b_x ise Denklem 41'te gösterildiği gibi tanımlanmaktadır:

$$bx = (bx_{ilave} + bx_{çarpım} \times z) \times \frac{1}{v} \quad (41)$$

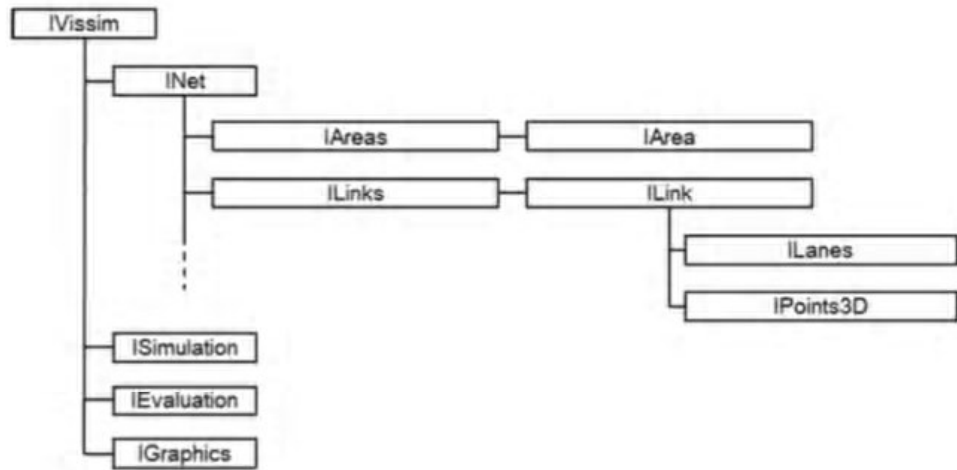
Burada v (m/s) aracın hızı olarak tanımlanmaktadır. z ise 0.5 ortalama ve 0.15 standart sapmalı $[0, 1]$ arasında değişen değer aralığıdır.

2.1.1.3. COM Arayüzü

Bileşen Nesne Modeli (The Component Object Model (COM)), kısaca yazılım bileşenleri için ikili bir arabirim standardıdır. Farklı programların ikili bileşenlerinin birlikte nasıl çalıştığını açıklamaktadır. COM, diğer programlarda bulunan verilere ve işlemlere erişim sağlamaktadır. VISSIM 4.0 sürümünden bu yana, VISSIM'deki verilere, otomasyon sunucusu yerine VISSIM kullanılarak, COM arayüzü üzerinden erişilebilir. VISSIM Sürüm 4.30'dan beri, COM komut dosyaları doğrudan VISSIM ana menüsünden çağrılabilir.

COM, belirli bir programlama diline bağlı değildir. COM Nesneleri, VBA, VBS, Python, C, C ++, C #, Delphi ve MATLAB dahil olmak üzere çok çeşitli programlama ve komut dosyası dilleri ile kullanılabilir.

VISSIM-COM Modeli Şekil 9'da gösterildiği gibi bir nesne hiyerarşisine bağlıdır. IVISSIM en yüksek rütbeli nesnedir. Örneğin ağdaki bir bağlantı, alt nesneye erişmek için bu hiyerarşiye uymak zorundadır.



Şekil 9. VISSIM-COM Modeli Nesne Hiyerarşisi (PTV AG, 2018)

VISSIM-COM ile çalışırken, her şeyden önce, VISSIM'da bir model oluşturulmalı ve kaydedilmelidir. Tez kapsamında çalışma alanı olarak belirlenen trafik ağı VISSIM'da modellenmiştir.

Yol tasarımları COM aracılığı ile MATLAB ve desteklenen diğer programlama dilleri ile yapılabilir. Fakat VISSIM üzerinde grafik arayüzü olduğu için bu tasarımları gerçekleştirmek daha kolaydır (Skrůžný, 2017).

2.1.1.4. VISSIM ile Otomatik Aşama Tabanlı Sabit Zamanlı Kontrolün Yeşil Sinyal Optimizasyonu (Vissig)

VISSIM programının içerisinde Vissig eklentisi ile birlikte sabit zamanlı sinyal programlarında otomatik olarak yeşil sinyal süresi optimizasyonu yapmak mümkündür. Bu optimizasyonun yapılabilmesi için bazı girdiler ve kısıtlamalar bulunmaktadır.

Girdiler şu şekildedir:

Sinyal grupları, yeşiller arası bekleme (intergreen) matrisi, fazlar ve faz temelli sinyal programı.

Kısıtlar ise şöyledir:

Minimum yeşil zamanlar ve ilgili yeşil alanlar kavşak noktalarına dahil edilmelidir, çünkü optimizasyon aşama uzunluğunu sıfıra indirebilir. Tüm aşamalar sıfır uzunluk gösterse bile sinyal programının tutarlı olması gerekir.

Aşama bazlı sinyal programı keyfi aşama uzunluklarına sahip olabilen kullanıcı tanımlı bir döngü süresine sahiptir. Sinyal programı oluştururken varsayılan değerler olarak faz uzunluklarının kullanılması Vissig yönteminin temelini oluşturmaktadır. Algoritma kısaca şu adımları izlemektedir:

1. Otomatik olarak üretilen bir düğüm değerlendirmesini kullanarak, VISSIM her bir sinyal grubu ve tam simülasyon için, sinyal grubunun belirli şeritlerinde kavşaktan geçen tüm araçların, belirli dönüş ilişkileri çerçevesinde, ortalama gecikme süresini belirler.
2. Optimizasyon için araçların en büyük ortalama gecikmeyi gösterdiği sinyal grubu her aşama için ayrı ayrı belirlenir.
3. Daha sonra en düşük maksimum ortalama gecikmeli durum en iyi faz olarak seçilirken, en büyük maksimum ortalama gecikmeli durum en kötü faz olarak seçilir. En iyi fazın yeşil süresinden bir saniye eksiltiilip en kötü fazın yeşil süresine bir saniye eklenir. En iyi fazdan kaldırılacak hiçbir saniye kalmazsa en iyi ikinci faz kullanılacaktır. Bu faz bile azaltılamaz duruma geldiğinde, yinelemeyle bir sonraki en kötü faz olan kullanılır. Daha fazla azaltma için herhangi bir fazın kalmadığı durumda optimizasyon tamamlanır.
4. Akış (simülasyon çalışırken düğümü geçen toplam araç sayısı) önemli ölçüde artmamışsa (en az 25 araç veya %10) anlık sinyal programı her zaman daha iyi kabul edilir. Önemli ölçüde artan akış için tüm araçların ortalama gecikmesi azalır. Bir sinyal programı öncekinden daha yüksek derecelendirilmişse, önceki en iyi sinyal programı tercih edilen programla değiştirilir ve optimizasyon başka bir adımla devam eder.
5. Optimizasyon, sinyal programının 10 simülasyon içerisinde iyileşmediği veya akışın o ana kadarki en iyi sinyal programına kıyasla %25'ten daha fazla azalması veya ortalama gecikmenin %25'ten fazla artması durumunda sona erer.

2.2. Sezgisel Arama Algoritmaları

Uzman Sistemler, İstatistiksel öğrenme, Derin Öğrenme ve Yapay Sinir Ağları, Bulanık Sistemler, Sezgisel Hesaplama gibi Yapay Zekâ (YZ) teknolojileri, yakın gelecekte geliştirilerek insan toplumunun tüm sosyal ve ekonomik yapısını yeni bir formata dönüştürme potansiyeline sahiptir. Gelişmiş YZ uygulamalarının çoğu hala yeni ve oldukça gelişmiş taşınabilir, giyilebilir nano ölçekli bilgisayar, sensör, enerji teknolojileri ve bilgi işlem algoritmaları gerektirmektedir. Bu nedenle araştırmacıların sezgisel hesaplama teknolojilerine en küçük katkıları bile teknolojik gelişim için kritik öneme sahiptir. Son yıllarda, araştırmacılar daha verimli genetik operatörler ile hızlı Sezgisel Arama Algoritması (SAA) yöntemleri geliştirmeye odaklanmıştır.

Standart Diferansiyel Gelişim Algoritması (sDE), sezgisel kompozisyonda en yaygın kullanılan stokastik arama yöntemlerinden birisidir. Kullanılan mutasyon, çaprazlama gibi genetik operatörler ve bu operatörlerin içsel parametrelerinin başlangıç değerleri, sDE'nin problem çözme başarısını doğrudan etkilemektedir (Brest vd., 2006; Das vd., 2016; Price vd., 2005). sDE'de genetik operatörlerin seçilmesi ve parametrelerinin başlangıç değerlerinin ayarlanması, deneme yanılma sürecine dayanmakta ve zaman alıcı olmaktadır (Mlakar vd., 2016; Zhang ve Sanderson, 2009). SAA'ların parametrelerinin başlangıç değerlerinin seçimi ve ilgili algoritmaların problem çözme başarıları arasındaki karmaşa ilişkisi hala aktif bir araştırma alanıdır. Çünkü problem için en iyi Sezgisel Arama Algoritması parametrelerini ayarlayan analitik bir parametre belirleme yöntemi bulunmamaktadır. SAA'lara ait iç parametrelerin başlangıç değerlerine karşı çok hassas olması ve ilgili iç parametreler için sabit başlangıç değerlerinin kullanılması, SAA'ların problem çözme başarısını sınırlayabilir. Bu nedenle, SAA'lar için çeşitli uyarlamalı parametre ayarlama yöntemleri geliştirilmiştir. Adaptif parametre ayarlama yöntemleri de, sabit parametre ayarlama yöntemleri gibi problem tipine bağlı olarak SAA'nın problem çözme başarısında etkili olmaktadır (Chen vd., 2017; Clerc ve Kennedy, 2002; Karaboğa ve Basturk, 2007; Liu vd., 2018; Lynn ve Suganthan, 2017; Ozsoydan ve Baykasoğlu, 2019; Qin vd., 2014; Zhang ve Zhu, 2011).

sDE için geliştirilmiş birçok mutasyon ve çaprazlama operatörü bulunmaktadır. Mutasyon operatörleri genellikle iki farklı yöntemden birini kullanarak yeni bir çözüm vektörü üretmektedir. Birinci tür çözüm-vektör üretme metotları basitçe parametre-uzay

bazlı vektör karışımını temel almaktadır (Das vd., 2016; Opara ve Arabas, 2018). Bu yöntemler ana vektörlerden çok farklı bir yeni vektör üretmesine izin vermesine rağmen, başarısı ebeveyn vektörlerdeki gen çeşitliliğine bağlıdır (Das vd., 2016; Opara ve Arabas, 2018; Zhang vd., 2019). İkinci tipte çözelti vektörünü üretme yöntemleri de sahte olmayan rastgele bir sayı üretici kullanarak yeni çözüm vektörünün üretilmesine dayanır. Bu yöntemler, çeşitlilik probleminden etkilenmese de, arama süreci uzun ve çoğu zaman verimsiz hale gelmektedir (Das vd., 2016; Zhang vd., 2019) sDE için binom ve üstel olarak adlandırılan iki tür geçiş operatörü bulunmaktadır (Das vd., 2016). Günümüzde henüz sDE kullanarak nümerik bir problemi en verimli şekilde çözmek için hangi mutasyon ve geçit operatörlerinin kullanılması gerektiğini gösteren analitik bir yöntem bulunmamaktadır. Bu nedenle, ilgili arama işlemini en verimli şekilde yapmak için gereken mutasyon ve çaprazlama operatörlerinin kombinasyonunu belirlemek zor ve zaman alıcıdır. Daha verimli mutasyon ve çaprazlama işlemlerinin geliştirilmesi, ilgili arama sürecini radikal biçimde verimli hale getirebilmektedir (Das vd., 2016; Zhang vd., 2019).

SAA'ların parametreleri ortak parametreler ve yapısal parametreler olarak iki gruba ayrılabilir. Ortak parametreler yinelemelerin sayısı ve örüntü matrisinin elemanlarının sayısı (ham genetik yöntemlerdeki popülasyon) gibi doğrudan algoritma ile yapısal olarak ilişkilendirilemeyen parametrelerdir. Bu parametrelerin değerleri, tüm SAA'larda problem çözme başarısını etkilemektedir. İlgili algoritmayı doğrudan tanımlamak için gerekli olan yapısal parametreler algoritmanın problem çözme başarısını ve yakınsama yeteneğini etkilemektedir. Ortak parametrelerin yanı sıra SAA'lar, kullanımlarını zorlaştıran ve zaman alan parametre ayarlama işlemlerini gerektiren başka alt parametrelere de sahiptir (Çivicioğlu, 2013a, 2013b, 2012; Karaboğa ve Basturk, 2007; Price vd., 2005).

SAA'lar için geliştirilen parametre adaptasyon yöntemleri , deterministik yöntemler, uyarlamalı (adaptive) yöntemler ve kendine uyarlamalı (self-adaptive) yöntemler olarak 3 gruba ayrılabilir. Deterministik yöntemler, algoritma tarafından sağlanan bilgileri kullanmadan analitik yöntemleri kullanarak parametreyi değiştirir. Deterministik yöntemler, sorunun türüne bağlı olarak davranışı değiştirme yeteneğine sahip değildir. Örneğin; Cuckoo Arama Algoritması (CAA) çok başarılı ve çok hızlı bir diferansiyel evrim algoritmasıdır (Çivicioğlu ve Beşdok, 2013; Yang ve Deb, 2009). CAA'nın

parametreleri sabit değerlere sahiptir. Bu nedenle parametreleri ayarlamak için deterministik bir işlem kullanmaktadır. Uyarlanabilir yöntemler ilgili kontrol parametrelerinin değerlerini algoritmanın başarısına göre ayarlamaktadır. Örneğin Uyarlamalı Diferansiyel Evrim Algoritması (JADE) (Zhang ve Sanderson, 2009) ve kendine uyarlamalı diferansiyel gelişim algoritması (self-adaptive differential evolution algortihm (SADE)) (Brest vd., 2006) uyarlamalı parametreleri ayarlama yeteneğine sahip DE algoritmalarıdır. Uyarlanabilir yöntemler, elde edilen başarıya bağlı olarak davranışı değiştirebilir. Kendinden uyarlamalı yöntemler her bir model vektörü için bir parametrenin değerini ayrı ayrı belirlemektedir. Yapay Arı Koloni Algoritması (ABC) böyle algoritmalara örnek olarak verilebilir (Çivicioğlu, 2013a, 2013b; Karaboğa ve Basturk, 2007). Uyarlamalı parametre belirleme yönteminde iyi parametre değerlerine sahip örüntü vektörleri daha iyi gelişme potansiyeline sahiptir. Bir SAA'nın ilgili parametre değerlerini belirlemek için uyarlanabilir veya kendinden uyarlamalı bir yöntem kullanmak, ilgili SAA'nın problem çözme başarısını artırabilmektedir. Bu durum parametre değerlerini belirlemek için gereken deneme-yanılma işlemini tamamen ortadan kaldıramasa da oldukça kısaltılabilmektedir. Uyarlanabilir ve kendinden uyarlamalı DE'ler, sayısal optimizasyon problemlerini çözmede klasik DE'den daha başarılıdır. Parametre değerleri uyarlanabilir olsa bile sorunu çözmek için kullanılması gereken en iyi mutasyon ve çaprazlama stratejisini bulmak zaman alıcı bir deneme yanılma sürecini içermektedir (Çivicioğlu, 2013a, 2013b, 2012). Bu durum mutasyon ve çaprazlama süreçleri nispeten basit olduğu için parametreleri rastgele belirlenebilen yeni DE tekniklerinin geliştirilmesini teşvik etmektedir.

sDE'nin genetik operatörlerinin neden olduğu eksikliklerden arınmış evrensel bir DE (uDE), arama alanını daha verimli bir şekilde arayabilmektedir. Bir uDE'nin problem çözme başarısı, kullanılan genetik operatörlere ve ilgili operatörlerin parametrelerinin başlangıç değerlerine duyarlı değildir. uDE geliştirme çabaları parametresiz, çok verimli, hızlı ve kontrol edilmesi kolay ve son derece gelişmiş DE sürümleri geliştirmeyi amaçlamaktadır. Kolayca kontrol edilebilir, basit yapılandırılmış, özyinelemeli olmayan, yüksek verimli, hızlı ve pratik bir parametre içermeyen uDE'ye duyulan ihtiyaç, önerilen BSD algoritması için önemli bir yere sahiptir.

Bu tez kapsamında en iyi trafik sinyal zamanlarının belirlenmesi için Bernstein Arama Diferansiyel Gelişim Algoritması (BSD) kullanılmıştır. Sonuçların karşılaştırılabilmesi

ve ölçeklenebilmesi için yine SAA olan GA ve sDE tekniklerinin yanında istatistik temelli olan Vissig algoritması ile en iyi trafik sinyal zaman belirlenmesi işlemleri yapılmıştır. GA ve sDE algoritmaları SAA'nın temel teknikleri sayılabileceğinden ilerleyen bölümlerde öncelikle bu algoritmalara değinilip ve sonrasında BSD algoritmasının ayrıntılarından bahsedilecektir.

2.2.1. Genetik Algoritma (GA)

Genetik Algoritma (GA) fikri ve temelleri John Holland tarafından 1975 yılında “Adaptation in Natural and Artificial Systems” adlı kitapta ortaya atılmış ve 1989’da öğrencisi Goldberg tarafından genişletilmiştir (Whitley, 1994). GA, optimizasyon problemlerinin çözümü için doğada gözlenen evrimsel süreci temel almaktadır. Kavramsal olarak çok boyutlu çözüm uzayında en iyi çözümün hayatta kalması prensibine dayanmaktadır.

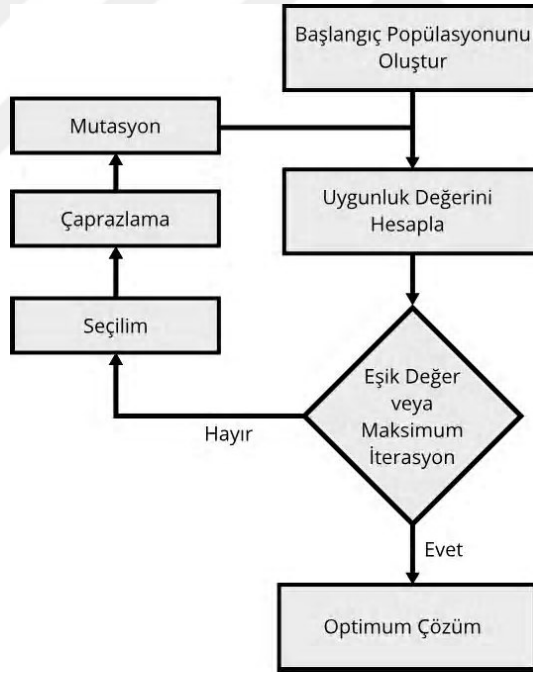
GA, mutasyon (mutation), çaprazlama(crossover) ve seçilim(selection) gibi doğadan ilham alarak operatörlere dayalı optimizasyon ve arama gibi problemlere yüksek kaliteli çözümler üretmek için yaygın olarak kullanılmaktadır.

GA’da, bir optimizasyon problemine aday çözümler (bireyler, yaratıklar veya fenotipler olarak ta kullanılmaktadır) daha iyi çözümlere doğru yakınsama yapılıır. Her aday çözümün mutasyona uğratılabilen ve değiştirilebilen bir dizi özelliği (kromozomları veya genotipi) bulunmaktadır; geleneksel olarak, çözümler 0 ve 1’lerin dizeleri şeklinde ikili olarak temsil edilmektedir. Ancak başka kodlamalar da mümkündür. Örneğin; bu tez kapsamındaki problemde çözüm kümesi trafik ağından çıkan araçların toplamı olduğundan tam sayılı bir çözüm üretilmektedir.

2.2.1.1. Genetik Algoritmanın Adımları

GA yönteminde, çözüm genellikle rastgele üretilen bireylerin popülasyonu ile başlar (initialization) ve her bir yinelemedeki popülasyonun bir nesil olarak adlandırıldığı yinelemeli bir süreçte ilerler. Her nesilde, popülasyondaki her bireyin uygunluğu (fitness) değerlendirilmektedir. Uygunluk genellikle çözülmekte olan optimizasyon problemindeki amaç fonksiyonun değeridir. Daha uygun bireyler, mevcut popülasyondan stokastik olarak seçilir ve her bireyin genomu, yeni bir nesil oluşturmak

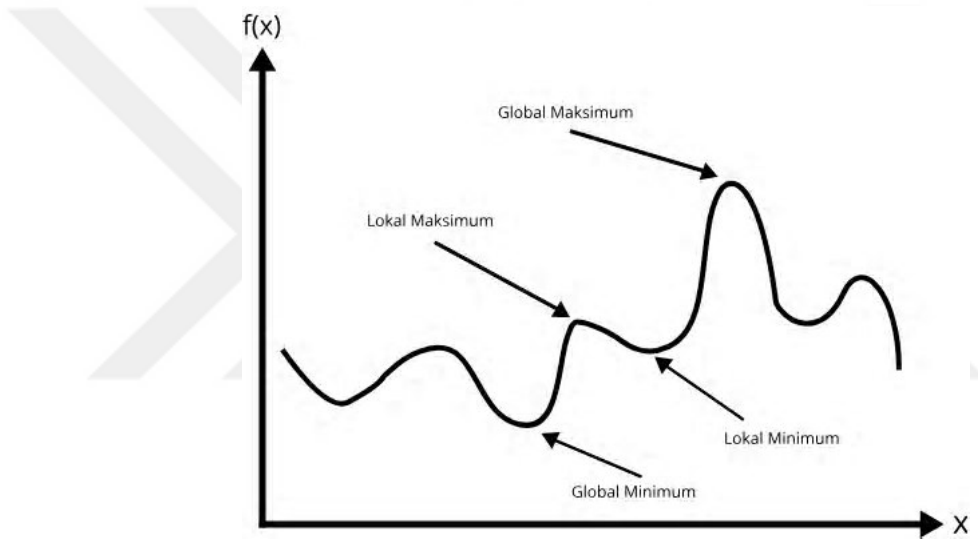
için değiştirilmektedir (çaprazlanır ve rastgele mutasyona uğratılır). Yeni nesil aday çözümler daha sonra algoritmanın bir sonraki yinelemesinde kullanılmaktadır. GA yönteminin akış diyagramı Şekil 10'da gösterilmiştir. Genel olarak algoritma, maksimum sayıda nesil üretildiğinde veya popülasyon için yeterli (eşik) uygunluk seviyesine ulaşıldığında sona ermektedir. GA genel olarak iki temel bilgiye ihtiyaç duymaktadır. Bunlardan birincisi sistemi temsil eden değişkenler ve sonuçlardır. İkincisi ise sistemin uygunluk yani amaç fonksiyonudur. Bu tezdeki problem için sistemi temsil eden değişkenler trafik sinyal lambalarına ait değerler, çıkış ise ağı terk eden araç sayısı olacaktır. Sistemin uygunluk fonksiyonu ise araç sayım detektörlerinden elde edilen giren ve çıkan toplam araç sayılarının birbirine olan oranının maksimize edilmesi olarak özetlenebilir.



Şekil 10. Genetik Algoritmaya Ait Akış Diyagramı (Whitley, 1994)

Başlangıç Popülasyonun Oluşturulması (Initialization): Popülasyon, sistem ile hesaplanmış girdilerin ve bu girdilere ait oluşan çıktının oluşturduğu küme olarak adlandırılabilir. Kümenin her üyesine kromozom, kromozomun her alt üyesine (girdiler ve çıktı) birey adı verilmektedir. Popülasyon, girdilerin alt ve üst sınırları içerisinde rastgele üretilen değerler ile bu girdi değerlerine karşılık gelen çıktı değerlerinden oluşturan kromozomlardan meydana gelmektedir. Üretilen bu kromozomların çıktıları

optimize edilmeye çalışılmaktadır. Popülasyon sayısının seçimi her problemin çözümü için farklı olabilmektedir. Fakat popülasyonun büyük seçilmesi çözüm sürecinin uzamasına, çok küçük seçilmesi ise çözümün yeterince optimize edilemeyip lokal çözümler üretmesine neden olabilmektedir. Problemin optimum olabilmesi için global çözümler üretmesi beklenmektedir. Optimizasyon problemleri minimizasyon ve maksimizasyon problemi olabilmektedir. İki boyutlu bir uzayda yerel ve global noktalar Şekil 11’de gösterilmiştir. Şekil 11’de x ’ler giriş değişkenini, $f(x)$ ise çıkış değişkenini temsil etmektedir. Tez kapsamında işlenen problemin 94 parametre girişi ve 1 çıkış olduğu için sistem 95 boyutludur.



Şekil 11. İki Boyutlu Uzayda Yerel ve Global Çözümler

Seçilim (Selection): Üretilen kromozomların kaliteli olanları, bazen de ek olarak kaliteli olmayanların bir kısmı optimizasyon için kullanılmak amacı ile seçilmektedir. Seçilim için en yaygın kullanılan algoritmalar Rulet Tekerleği, Rank, Kararlı Durum, Turnuva ve Elitizm Algoritmalarıdır (Whitley, 1994). Seçilim sırasında kromozomlar üzerinde bir değişiklik yapılmayıp kaliteli olanlar filtrelenip, kaliteli olmayanlar ise yok edilmektedir.

Çaprazlama (Crossover): Seçilim sonrası kalan kromozomlar arasındaki girdi değerlerinin karşılıklı olarak değiştirilmesi işlemidir. Çaprazlama işleminin amacı en iyi kromozomlar arasındaki bireylerin değiştirilip daha kaliteli kromozomlar üretilme ihtimalinin artırılmasıdır. Literatürde en çok karşılaşılan çaprazlama teknikleri tek-noktalı, iki veya k noktalı ve üniform çaprazlama algoritmalarıdır (Whitley, 1994).

Mutasyon (Mutation): Çaprazlama sonucunda oluşan kromozomlar kendi içinde çok tutarlı olup çözüme hızlı yakınsamaya sebep olmaktadır. Fakat yakınsanan sonucun lokal en iyi çözüm olması ihtimali daha yüksektir. Bunun sebebi tek bir global çözüme karşılık birden çok lokal çözümün olmasıdır. Global çözüm arama ihtimalini arttırmak amacı ile oluşan kromozomların bazı bireyleri değiştirilmektedir. Bu değiştirme işlemi mutasyon olarak adlandırılmaktadır.

2.2.2. Standart Diferansiyel Gelişim Algoritması (sDE)

sDE, 1995 yılında Prince ve Storn tarafından, GA temeline dayanan, kolay kodlanabilme ve hızlı sonuçlar verme avantajına sahip sezgisel bir algoritmadır (Karaboğa, 2004). sDE tıpkı GA gibi popülasyon temelli; seçim, çaprazlama ve mutasyon gibi genetik operatörleri kullanmaktadır. Fakat GA'dan farklı olmak üzere genetik operatörler tüm kromozomlara değil her kromozomun yanında seçilen üç farklı kromozoma uygulanmaktadır. Ayrıca çözüm uzayında aynı anda pek çok noktada arama yaparak sonuca daha hızlı yakınsanmaktadır.

2.2.2.1. sDE'nin adımları

Başlangıç Popülasyonunun Oluşturulması: Tıpkı GA'da olduğu gibi sistem için alt ve üst sınırları arasında üretilmiş girdi ve çıktılardan oluşan bir başlangıç popülasyonu oluşturulmaktadır. Oluşan popülasyon 1 hedef kromozomu, 2 rastgele seçilen kromozom ve 1 mutasyon uygulanacak kromozom olmak üzere en az 4 kromozom oluşturulmalıdır.

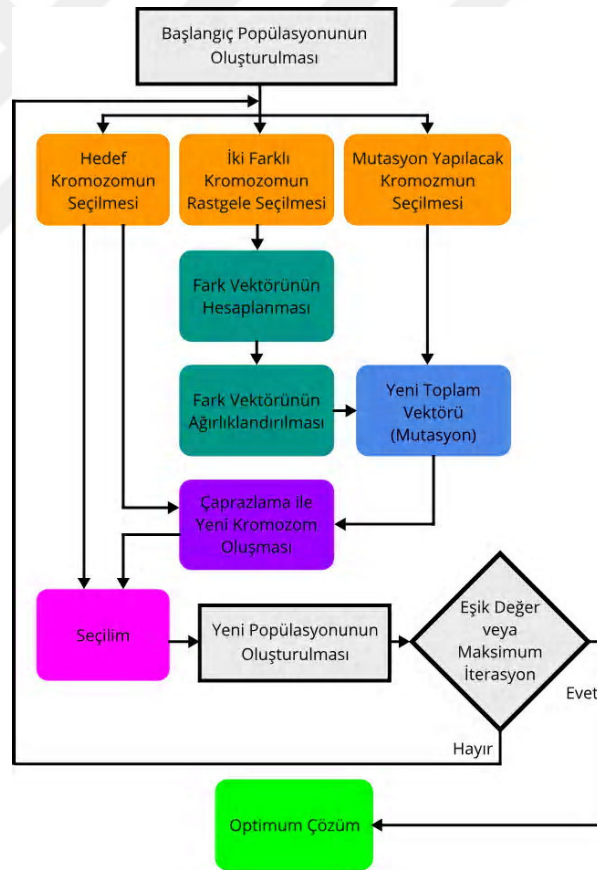
Mutasyon: Rastgele seçilen 2 kromozomun farkı alınıp F parametresi ile çarpılarak genişletilir. F parametresi genellikle $[0,2]$ aralığında seçilmektedir. Elde edilen değer ise üçüncü kromozom ile toplanır. Bu toplama işlemi ile 3. kromozom mutasyona uğramış olur. GA yönteminde kullanılan rastgele mutasyondan farklı olarak sDE'de kullanılan mutasyon kromozomların genişletilmiş haldeki çevrelerinde çözümü aramaya odaklanmaktadır. Böylelikle çok daha kısa adımda sonuca yaklaşma ihtimali GA'ya göre daha yüksektir.

Çaprazlama: GA çaprazlamasında da olduğu gibi, çaprazlamadaki amaç kromozomlar arasındaki bireylerin değiştirilerek başarı ihtimalinin artmasıdır. sDE'de

çaprazlama başlangıç popülasyonundan seçilen hedef kromozom ile mutasyona uğramış kromozom arasında yapılmaktadır. Çaprazlama için GA için kullanılan tekniklerin aynısı kullanılmaktadır. Çaprazlama sonunda elde edilen yeni bireyler ile uygunluk fonksiyonunun çıktısı üretilerek yeni kromozomlar elde edilmektedir.

Seçilim: Çaprazlama sonucunda uygunluk fonksiyonundan elde edilen kromozomlardan daha başarılı olan, bir sonraki yeni popülasyona aktarılır. Başarısız olan kromozom ise yok edilir. Bu işleme seçim adı verilmektedir.

Seçilim işleminden sonra elde edilen kromozomlar eşik değerinin altında başarı göstermişlerse (çözüm belli bir oranda iyileşmiyorsa) veya iterasyon sayısını tamamlamış ise algoritma durur. sDE'ye ait akış diyagramı Şekil 12'de verilmiştir.



Şekil 12. sDE'ye Ait Akış Diyagramı

2.2.3. Bernstein Arama Diferansiyel Gelişim Algoritması (Bernstrain-Search Differential Evolution Algorithm – BSD)

BSD algoritması Çivicioğlu ve Beşdok tarafından 2019 yılında önerilmiştir. BSD temelde DE'ye çok benzemektedir.

BSD yönteminin de yapısal parametre değerleri rastgele belirlenmektedir ve BSD'de, örüntü matrisindeki her örüntü vektörü ayrı ayrı geliştirilmektedir. Her model vektörünün evrimi diğerlerinden bağımsız olduğu için BSD doğal olarak, paralel bir arama algoritmasıdır. BSD'de, çaprazlama işlemi Bernstein polinomsalları kullanılarak kontrol edilmektedir. Bu nedenle BSD çaprazlama süreci için bir parametreye sahip değildir. Diğer EA'lar gibi BSD'nin problem çözme başarısı ortak parametre değerlerine duyarlıdır.

BSD tarafından sağlanan yenilikler şu şekildedir;

- BSD benzersiz bir bijektif (birebir örten) mutasyon stratejisi kullanmaktadır.
- BSD, sDE'nin çaprazlama operatörlerinininkilerden daha verimli bir çaprazlama operatörü kullanmaktadır.
- BSD'nin çaprazlama adımı Bernstein polinomsalları kullanılarak rastgele kontrol edilmektedir.
- BSD, sDE'nin aksine önceden belirlenmiş bir mutasyona ve çaprazlama değerine sahip değildir.
- BSD özyinelemeli olmayan bir yöntem olduğundan, BSD'yi paralel hesaplama yöntemleriyle kullanmak kolay olmaktadır.
- BSD, sDE ve sDE'nin modern uyarlamalarının aksine, mutasyon ve çaprazlama operatörlerinin iç parametreleri için parametre ayarlama aşamaları gerektirmemektedir.
- BSD, ABC, JADE ve CUCKOO elitist yöntemlerin aksine gibi kısmî elitist bir yöntemdir.

2.2.3.1. BSD Algoritmasının Adımları

BSD algoritmasında kromozomlara örüntü vektörü, popülasyona ise örüntü matrisi adı verilmektedir. Tıpkı GA ve sDE’de bahsedildiği gibi popülasyon yani örüntü matrisi rastgele çözümle N adet örüntü vektörü, örüntü matrisi olan P ’yi oluşturmaktadır. Her örüntü vektörü ise D adet bireyden oluşmaktadır.

EA’lar sınırlı ve/veya sınırsız arama yapabilmektedir. Sınırlı arama, bireylerin alt ve üst sınırları arasında çalışmaktadır (Çivicioğlu, 2013b, 2013a, 2012; Çivicioğlu vd., 2018). BSD yöntemi de, sınırlı arama gerçekleştiren küresel minimumlama algoritması olarak tasarlanmıştır. BSD’nin adımları şu şekildedir:

- 1- BSD’de bireyler alt (low_j) ve üst (up_j) sınırları kullanılarak Denklem 42 kullanılarak belirlenmektedir:

$$P_{i,j} \sim U_{low_j, up_j} \mid i = 1:N, j = 1:D, i, j \in \mathbb{Z}^+ \quad (42)$$

- 2- Örüntü vektörlerinin amaç fonksiyonun ($\mathcal{F}$), uygunluk ($fitP_i$) değerleri Denklem 43 kullanılarak hesaplanmaktadır:

$$fitP_i = \mathcal{F}(P_i) \quad (43)$$

- 3- Global minimumlama örüntü vektöründe, soruna en iyi çözümü sağlayan $bestP$ ve global minimumlama örüntü vektörünün ($solP$) amaç fonksiyon değeri Denklem 44 ile elde edilmektedir:

$$solP, bestP = fitP_{\gamma}, P_{\gamma} \quad fitP_{\gamma} = \min_{\gamma \in [1:N]} fitP_{\gamma} \quad (44)$$

- 4- BSD Denklem 45 ve Denklem 46’ı kullanarak M ile çaprazlama oranını kontrol etmektedir. M başlangıç değeri Denklem 45 ile belirlenmektedir:(Çivicioğlu ve Beşdok, 2019)

$$M_{(i=1:N, j=1:D)} = 0 \quad (45)$$

$$M_{(i, u(1: \rho, D))} = 1 \quad (46)$$

Buradaki ρ değeri Bernstein Polinomsalları kullanılarak Denklem 47 ile belirlenmektedir (Azhari vd., 2018). İkinci derece Bernstein Polinomsalları Bölüm 5.2.1.1’de anlatılacaktır.

switch κ_0

$$\text{case } 1 \quad \rho = (1 - \beta)^2$$

$$\text{case } 2 \quad \rho = 2 \cdot \beta \cdot (1 - \beta) \quad (47)$$

$$\text{case } 3 \quad \rho = \beta^2$$

endsw

Burada $\beta \sim \mathbf{U}(0,1)$ ve $\kappa_0 = 3 \cdot \kappa_1^3, \kappa_0 \sim \mathbf{U} 0 \ 1, \kappa_0 \in \mathbf{U} 1:3$ olarak tanımlanmaktadır. Denklem 40’ta bulunan u vektörü Denklem 48 kullanılarak tanımlanmaktadır:

$$u = \text{permute}(1:D) \quad (48)$$

Burada $\text{permute}(a)$ fonksiyonu a ’nın aldığı değere göre rastgele olarak değişmektedir.

5- Evrimsel adım olan seçim (F) ise Denklem 49 kullanılarak tanımlanmaktadır:

$$\begin{aligned} \text{if } \kappa_2 < \kappa_3, \quad F &= [\eta_{(1,1:D)}^3 \circ \lambda_{(1,1:D)}^3]' \times Q_{(1,1:N)}' \\ \text{else,} \quad F &= \lambda_{(N,1)}^3 \times Q_{(1,D)} \end{aligned} \quad (49)$$

Burada κ_2, κ_3, η ve λ değerleri $\kappa_{2:3}, \eta \sim \mathbf{U} 0,1, \lambda \sim \mathbf{N}(0,1)$ ve $Q_{(m,n)} = 1$ olan $m \times n$ boyutundaki birler matrisinin her bir adımındaki elde edilen rastgele sayılardır.

6- BSD’nin deneme örüntü vektörünün (T_i gibi) üretim süreçleri rastgele çaprazlama süreçlerinden oluşmaktadır. BSD’de deneme örüntü vektörleri Denklem 50 kullanılarak üretilmektedir.

$$T = P + F \circ M \circ w^{*3} \circ E + 1 - w^{*3} \circ \text{best}P - P \quad w_{1:N,1}^* \sim \mathbf{U}(0,1) \quad (50)$$

Burada $E = w.P_{L_1} + 1 - w .P_{L_2}$ $w_{(1:N,1:D)} \sim U(0,1)$ ve L_1 ve L_2 Denklem 51 ile tanımlanmaktadır:

$$L_1 = \text{permute } 1:N, L_2 = \text{permute}(1:N) \quad L_1 \neq 1:N, L_1 \neq L_2 \quad (51)$$

7- Herhangi bir deneme örüntü vektörüne ait bireyin arama uzayının boyutunu aşması durumunda birey Denklem 52 kullanılarak güncellenmektedir:

$$\begin{aligned} & \text{If } T_{i,j} < low_j \text{ or } T_{i,j} > up_j \\ & \text{then } T_{i,j} = low_j + \delta.(up_j - low_j) \end{aligned} \quad (52)$$

Burada, $\delta \sim U(0,1)$ olarak tanımlanmaktadır.

8- Deneme örüntü vektörlerine ait amaç fonksiyonu, $\mathcal{F}(\cdot)$ ve $fitT$, Denklem 53 kullanılarak hesaplanmaktadır:

$$fitT = \mathcal{F}(T) \quad (53)$$

9- İlgili örüntü vektörünü güncellemek için karşılık gelen örüntü vektöründen daha iyi bir hedef fonksiyon değeri sağlayan deneme örüntü vektörü kullanılmaktadır. Model de vektörünün nesnel işlev değeri ile güncellenmektedir. Bu işlem Denklem 54 kullanılarak elde edilmektedir:

$$\text{If } fitT_{(i^*)} < fitP_{(i^*)}, P_{(i^*)}, fitP_{(i^*)} = T_{i^*}, fitT_{i^*} \quad | \quad i^* \in [1:N] \quad (54)$$

10- Mevcut yineleme adımında, en iyi çözümü sağlayan model vektörü ($bestP$) ve amaç fonksiyon değeri ($solP$), Denklem 55 kullanılarak elde edilmektedir:

$$solP, bestP = fitP_{\gamma}, P_{\gamma} \quad | \quad fitP_{\gamma} = \min(fitP) \quad (55)$$

BSD algoritmasına ait kod mimarisi Şekil 10'da verilmiştir. BSD'nin yapısal özellikleri şu şekilde listelenebilir:

- BSD'nin rastgele çapraz geçiş süreci test edilen yöntemlere karşılık gelen çaprazlama işlemlerinden farklıdır.

- BSD'nin çaprazlama süreci Bernstein polinomsallarının kullanımına dayanan stokastik bir süreçtir ve bu süreci kontrol eden herhangi bir parametre bulunmamaktadır.
- BSD, küresel minimumlama örüntü vektörü vektörünü sistem denkleminde kullandığı için (Denklemler 50), ABC ve CUCKOO gibi elitist algoritmalara nazaran kısmî elitist bir davranış sergilemektedir.
- BSD, amaç fonksiyonu değerlerini hesaplamak için paralel olarak çalışabilir ve herhangi bir değişiklik yapmadan sınırlı veya sınırsız arama yapabilmektedir.
- BSD, ABC ve CUCKOO'dan farklı olarak tek adımlı bir arama işlemine sahiptir.

```

Input: Objective Function:  $\mathcal{F}$ , Search-Space Limits:  $(low, up)$ , Size of Pattern Matrix:  $N$ ,
Dimension of problem:  $D$ , Maximum Number of Iterations:  $MaxCycle$ 
Output:  $solP$ : Global Minimum,  $bestP$ : Global Minimizer
// Initialization
1  $P_{i,j} \sim \mathcal{U}(low_j, up_j) \mid i = [1 : N], j = [1 : D], \text{ where } i, j \in \mathbb{Z}^+$ 
2  $fitP_i = \mathcal{F}(P_i)$ 
3  $[solP, bestP] = [fitP_{(\gamma)}, P_{(\gamma)}] \mid fitP_{(\gamma)} = \min(fitP) \mid \gamma \in [1 : N]$ 
4 for Iteration=1 to  $MaxCycle$  do
    // Generation of Mutation Control Matrix ; M
     $M_{(i=1:N, j=1:D)} = 0$ 
    for  $i=1$  to  $N$  do
         $u = \text{permute}(1 : D)$ 
        Generate  $\beta$ , where  $\beta \sim \mathcal{U}(0,1)$ 
        Generate  $\kappa_0$ , where  $\kappa_0 = \lceil 3 \cdot \kappa_1^3 \rceil, \kappa_1 \sim \mathcal{U}[0, 1], \kappa_0 \in \mathcal{U}\{1 : 3\}$ 
        switch  $\kappa_0$  do
            case 1,  $\rho = (1 - \beta)^2$ 
            case 2,  $\rho = 2 \cdot \beta \cdot (1 - \beta)$ 
            case 3,  $\rho = \beta^2$ 
        endsw
         $M_{(i, u(1:\rho \cdot D))} = 1$ 
    end
    // Generation of Evolutionary Step Size: F
     $\kappa_{2,3}, \eta$ , and  $\lambda$  are random numbers, where  $\kappa_{2,3} \sim \mathcal{U}(0, 1), \eta \sim \mathcal{U}(0, 1), \lambda \sim \mathcal{N}(0, 1)$ , and all-ones matrix  $Q_{(\cdot, \cdot)} = 1$ 
    if  $\kappa_2 < \kappa_3$  then
         $F = \left( \left[ \eta_{(1,1:D)}^3 \circ \lambda_{(1,1:D)}^3 \right]' \times Q_{(1,1:N)} \right)'$ 
    else
         $F = \lambda_{(N,1)}^3 \times Q_{(1,D)}$ 
    end
    // Generation of Trial Pattern Vectors; T
     $L_1 = \text{permute}(1 : N), L_2 = \text{permute}(1 : N) \mid L_1 \neq [1 : N], L_1 \neq L_2$ 
     $E = w \cdot P_{L_1} + (1 - w) \cdot P_{L_2} \mid w_{(1:N,1:D)} \sim \mathcal{U}(0, 1)$ 
     $T = P + F \circ M \circ ((w^*)^3 \circ E + (1 - (w^*)^3) \circ bestP - P) \mid w_{(1:N,1)}^* \sim \mathcal{U}(0, 1)$ 
    // Boundary Control Mechanism
    if  $(T_{i,j} < low_j) \text{ or } (T_{i,j} > up_j)$  then  $T_{i,j} = low_j + \delta \cdot (up_j - low_j) \mid \delta \sim \mathcal{U}(0, 1)$ 
    // Update
     $fitT = \mathcal{F}(T)$ 
    if  $fitT_{(i^*)} < fitP_{(i^*)}$  then  $[P_{(i^*)}, fitP_{(i^*)}] = [T_{(i^*)}, fitT_{(i^*)}] \mid i^* \in [1 : N]$ 
    // Get the solutions
     $[solP, bestP] = [fitP_{(\gamma)}, P_{(\gamma)}] \mid fitP_{(\gamma)} = \min(fitP)$ 
30 end

```

Şekil 13. BSD'ye ait Kod Mimarisi (Çivicioğlu ve Beşdok, 2019)

2.2.3.2. Bernstein Polinomsalları

Bernstein polinomsalları Sergei Natanovich Bernstein tarafından nümerik analiz alanında eğri uydurma (Curve Fitting) problemleri için geliştirilmiştir. Bernstein polinomsalları başlangıçta Stone–Weierstrass yaklaşımının kanıtı için geliştirilmiş olup, zamanla $[0, 1]$ uzayında Bezier eğrilerinden daha önemli bir hale gelmiştir.

İkinci dereceden Bernstein polinomsalları Denklem 56 ve 57 ile tanımlanmaktadır (Azhari vd., 2018):

$$B_{s,n} t = \binom{n}{s} t^s (1-t)^{n-s} \quad (56)$$

Burada $s = 0$: $\binom{n}{s} = \frac{n!}{s! (n-s)!}$ olarak ifade edilmektedir.

Denklem 57, $n + 1$ boyutlu n . Dereceden Bernstein polinomsallarını üretmektedir. $s < 0$ ve $s > n$ için $B_{s,n} = 0$ değerini almaktadır.

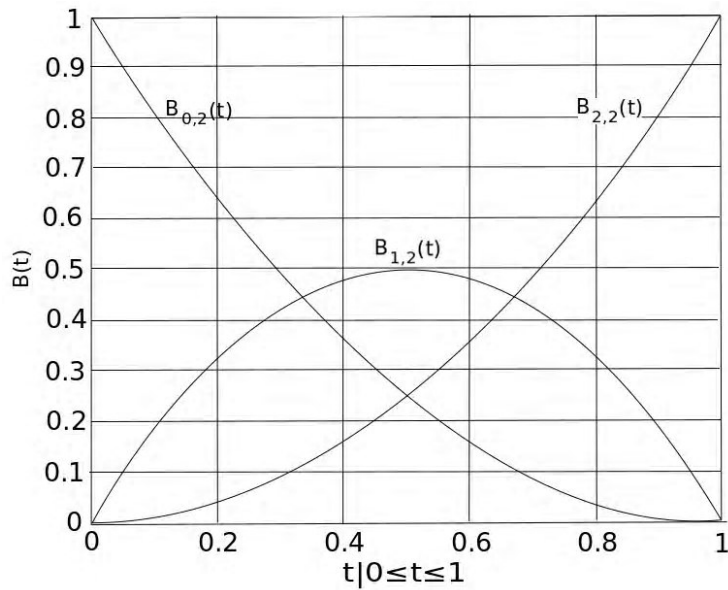
$$B_{0,2} t = (1-t)^2$$

$$B_{1,2} t = 2t(1-t)$$

$$B_{2,2} t = t^2 \quad (57)$$

Burada $B_{m,n}$ Bernstein polinomsalları ve t çözüm uzayının aralığını ifade etmektedir.

Şekil 11'de $0 \leq t \leq 1$ aralığındaki 2. dereceden Bernstein polinomsalları gösterilmektedir.



Şekil 14. 2. dereceden Bernstein Polinomsalları (HandWiki, 2020)

BÖLÜM 3

UYGULAMA ve BULGULAR

Çalışma bölgesi, halihazırda Kayseri Büyükşehir Belediyesi'nin sinyalizasyon çalışması yaptığı bölge olarak seçilmiştir. Bölge hakkındaki bilgiler Kayseri Büyükşehir Belediyesi tarafından hazırlanan rapordan elde edilmiştir.

3.1. Çalışma Alanı ve Mevcut Durum

Kayseri Büyükşehir Belediyesi sorumluluk alanında bulunan karayolu ağının hizmet düzeyinin belirlenerek öneri projelerin geliştirilmesi işi kapsamında; 4 adet kavşak sayımı yapılarak trafik hacimleri ve kavşaklardaki dönüş oranları ölçülmüştür.

Ölçümü yapılan 4 adet kavşağın yerleri Şekil 12'de verilmiştir. Şekil 12'de görüldüğü üzere K1, K2, K3 ve K4 kavşakları Kartal Bulvarı üzerinde yer almaktadır. Ölçümler sabah 07.00-09.00 arasında yapılmıştır.



Şekil 15. Ölçüm yapılan 4 adet kavşak

3.1.1. Kavşak Sayımları ve Mevcut Durum

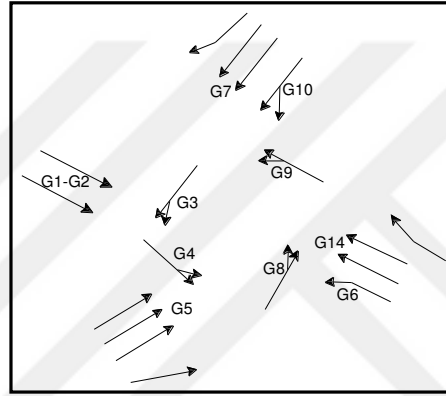
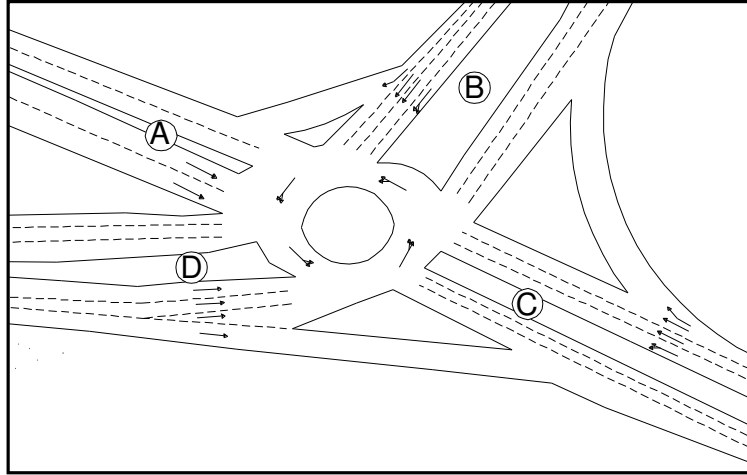
3.1.1.1. K1 Kavşağı (30 Nolu Sanat Okulu Kavşağı)

K1 Kavşağı dört kollu rotary adalı kavşak tipindedir. Kayseri'nin sabah ve akşam en işlek kavşaklarından olan Sanat Okulu Kavşağı; Mustafa Kemal Paşa Blv., Talas Blv., Seyyid Burhanettin Blv. ve Kartal Blv. kesişiminde bulunmaktadır.

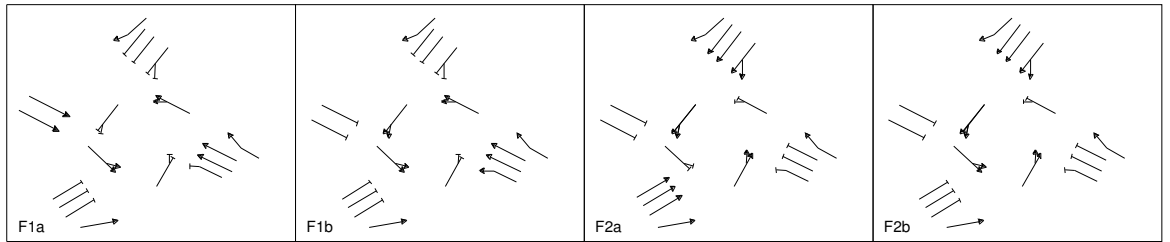
Şekil 16'te K1 kavşağının cadde görünümü, Şekil 17'de sayım yapılan kollardaki şeritler ve trafik akım yönleri, Şekil 18'te mevcut faz diyagramı ve Şekil 19'da mevcut sinyal grup diyagramı verilmiştir. Tablo 2'de bu kavşağa ait trafik hacim sayımları verilmiştir.



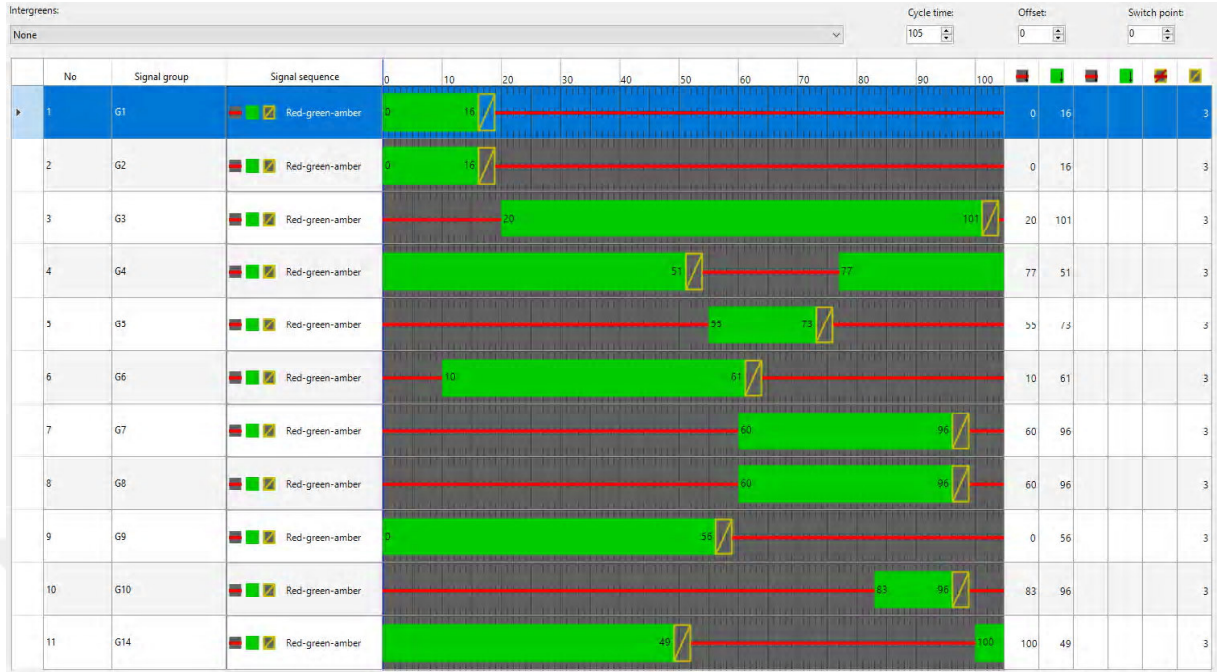
Şekil 16. K1 kavşağının cadde görünümü



Şekil 17. K1 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri



Şekil 18. K1 kavşağına ait faz diyagramı



Şekil 19. K1 kavşağına ait sinyal grup diyagramı

Şekil 19 VISSIM ortamında oluşturulmuş sinyal grup diyagramının ekran görüntüsüdür. Burada:

Cycle time: Döngü süresi (105)

Offset: Ofset, İki kavşak arasındaki arasındaki sinyal programlarının yeşil faza dönmesi arasındaki zaman farkı olarak tanımlanabilir. Genellikle eşit döngü uzunluğundaki kavşaklarda kullanılır. Bu çalışmada ofset değeri sıfır (0) olarak alınmıştır.

Swich Point: Gerekli durumlarda bir başka sinyal grup diyagramına geçilecek süre (Bu bilgi genellikle dışardan verilmektedir.)

No: Sinyal grup numaraları

Signal Group: Sinyal Grubu adları

Signal Sequence: Diyagramdaki sinyal tiplerini (Red-green-amber sırasıyla kırmızı-yeşil-sarı sinyallerini göstermektedir.)

Diyagram boyunca yatay uzanan yeşil kalın bar: Yeşil sinyalin yanma süresini

Yeşil kalın barın sağında kalan ve hemen ardından gelen içi köşegen çizgili sarı bar: Sarı sinyalin

Kırmızı ince çizgi: Kırmızı sinyali temsil etmektedir.

Diyagramın sağında bulunan ilk sütun kırmızı sinyalin bitiş zamanını (yeşil sinyalin başlangıç zamanını), ikinci sütun ise yeşilin bitiş zamanını göstermektedir. En sağdaki sütun ise sarı sinyalin süresini göstermektedir.

Tablo 2’de K1 kavşağına ait sabah (07:00-09:00) trafik hacim sayıları verilmiştir. Kavşakta A,B,C, D ile isimlendirilen 4 link (kol) bulunmaktadır. Trafik hacimleri tablosunda dikey sütunlar linklerin girişlerini ve yatay sütunlar linklerin çıkışlarını otomobil, otobüs, kamyonet ve minibüs bazında göstermektedir (Tablo 2). Tablo 2’de görüleceği üzere sabah saatlerinde kavşaktaki trafik yükü 13000 taşıt olup ortalama saatlik trafik hacmi 6500 taşıt/saat civarındadır. 08:00-09:00 arası bir saatlik dilimdeki trafik hacmi ise yaklaşık 7600 taşıt/saat mertebesinde.

Tablo 2. K1 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları

	A				B				C				D				TOPLAM
	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	
A	0	0	0	0	79	0	0	9	475	221	6	85	0	0	0	0	875
B	169	0	0	34	0	0	0	0	1302	124	20	139	1937	119	35	396	4275
C	2510	272	14	160	76	0	0	16	3	0	0	0	1666	82	20	138	4957
D	154	60	1	45	1464	148	25	436	476	23	0	108	0	0	0	0	2940
TOPLAM	2833	332	15	239	1619	148	25	461	2256	368	26	332	3603	201	55	534	13047

3.1.1.2. K2 Kavşağı (74 Nolu Şehitlik Kavşağı)

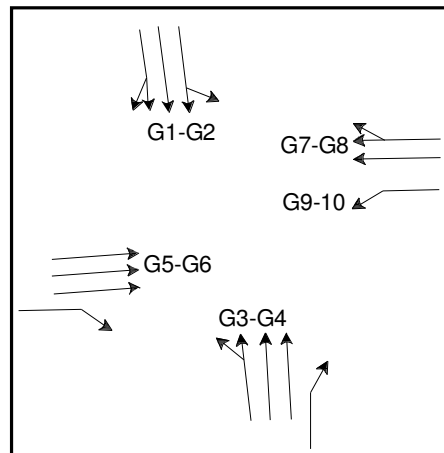
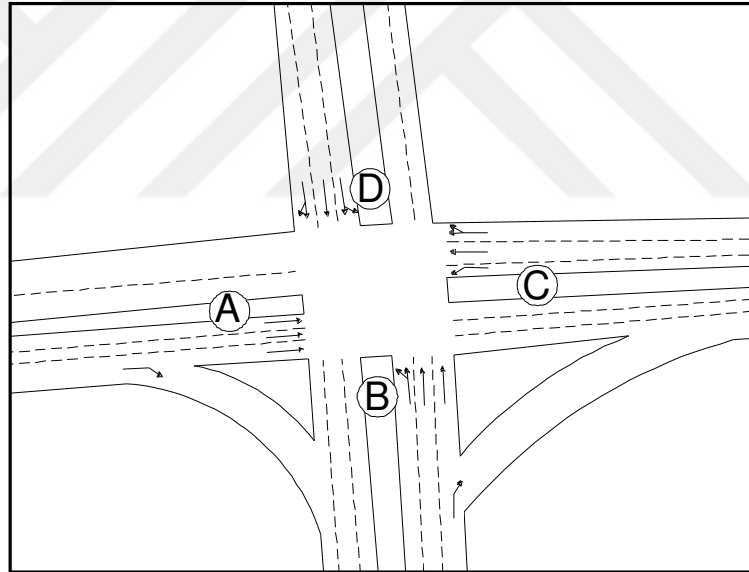
K2 Kavşağı dört kollu sığınmasız kavşak tipindedir. Yine çok yoğun bir noktada bulunan Şehitlik Kavşağı; Kartal Blv. ve Erciyes Blv. kesişim noktasında bulunmaktadır.

Şekil 20’de K2 kavşağının cadde görünümü, Şekil 18’de sayım yapılan kollardaki şeritler ve trafik akım yönleri, Şekil 19’da mevcut faz diyagramı ve Şekil 20’de mevcut

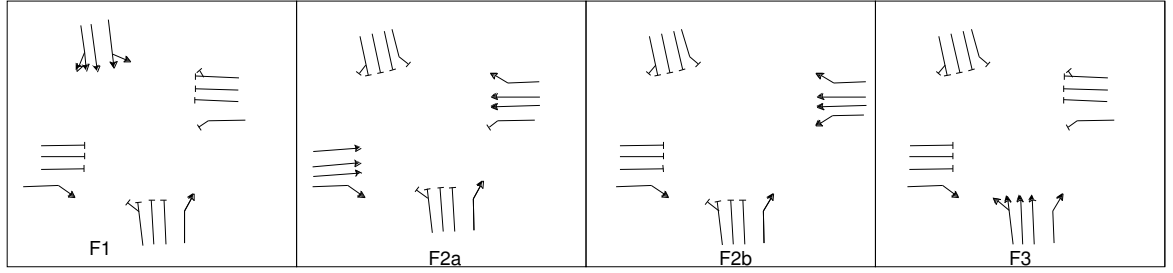
sinyal grup diyagramları verilmiştir. Tablo 3’te bu kavşağa ait trafik hacim sayımları verilmiştir.



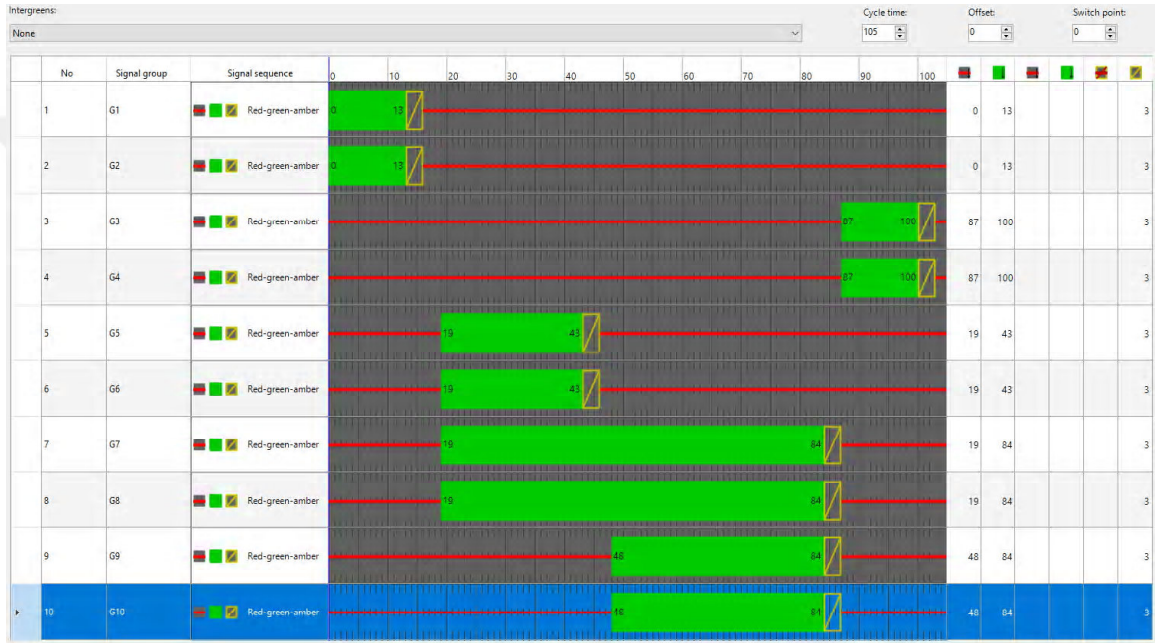
Şekil 20. K2 kavşağının cadde görünümü



Şekil 21. K2 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri



Şekil 22. K2 kavşağına ait faz diyagramı



Şekil 23. K2 kavşağına ait sinyal grup diyagramı

Tablo 3'te görüleceği üzere, sabah saatlerinde kavşaktaki trafik yükü 9000 taşıt olup saatlik trafik hacmi 4500 taşıt civarındadır.

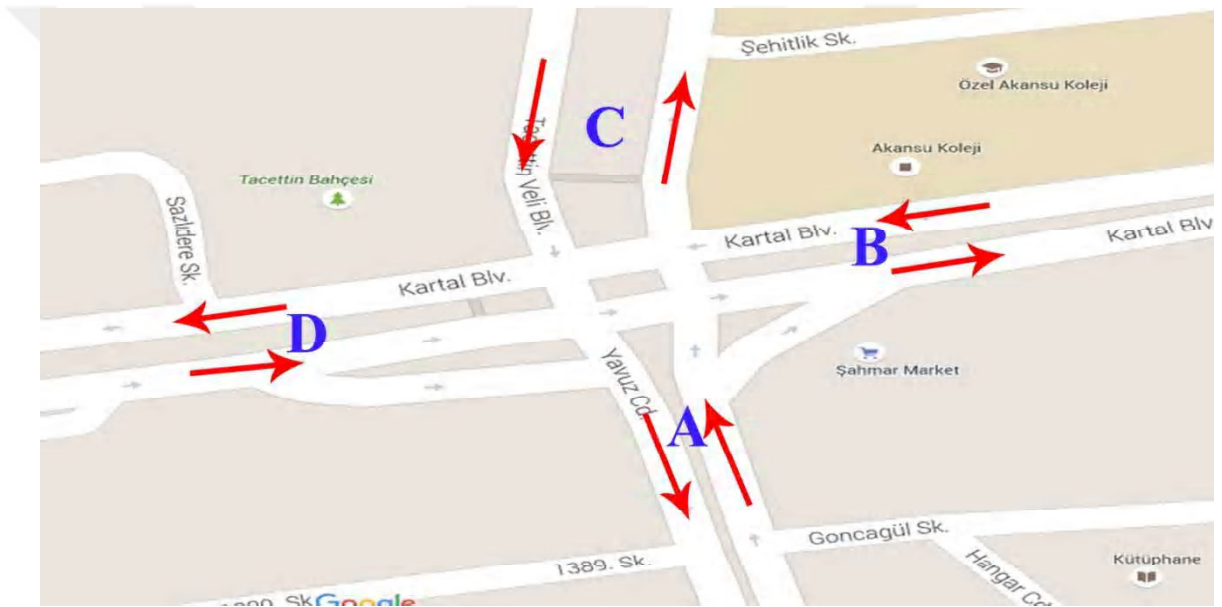
Tablo 3. K2 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları

	A				B				C				D				TOPLAM
	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	
A	0	0	0	0	61	3	1	5	1593	127	29	165	0	0	0	0	1985
B	203	5	5	13	0	0	0	0	1093	5	12	111	424	19	1	27	1919
C	3288	105	71	272	676	24	25	119	0	0	0	0	0	0	0	0	4580
D	63	21	0	13	191	29	5	45	159	1	0	16	0	0	0	0	544
TOPLAM	3553	132	76	299	928	56	32	169	2845	133	41	292	424	19	1	27	9028

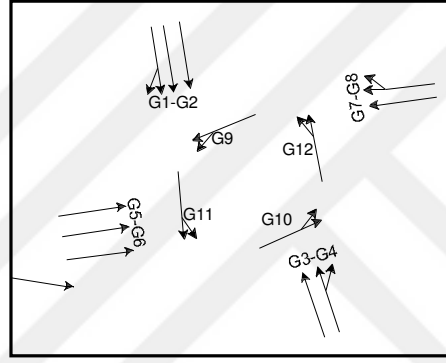
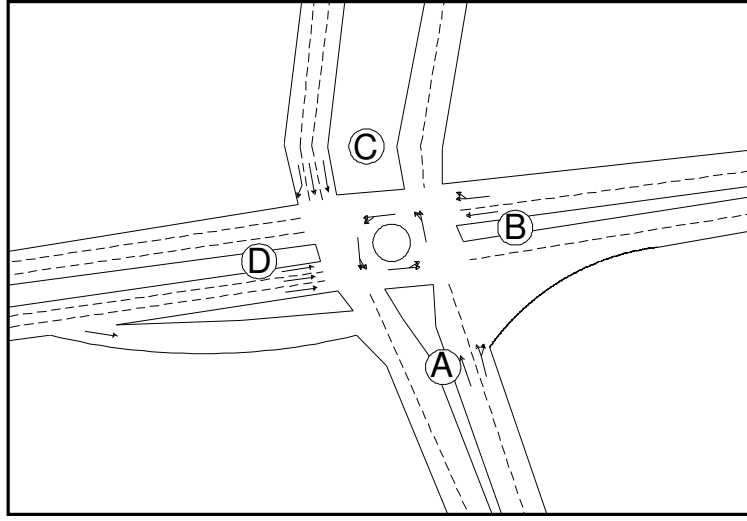
3.1.1.3. K3 Kavşağı (80 Nolu Akansu Kavşağı)

K3 Kavşağı dört kollu rotary adalı kavşak tipindedir. Akasu kavşağı; Kartal Blv.'nin devamında Tacettin Veli Blv.'nin son noktası olup kavşağın güney yaklaşım kolunda Yavuz Cad. bulunmaktadır.

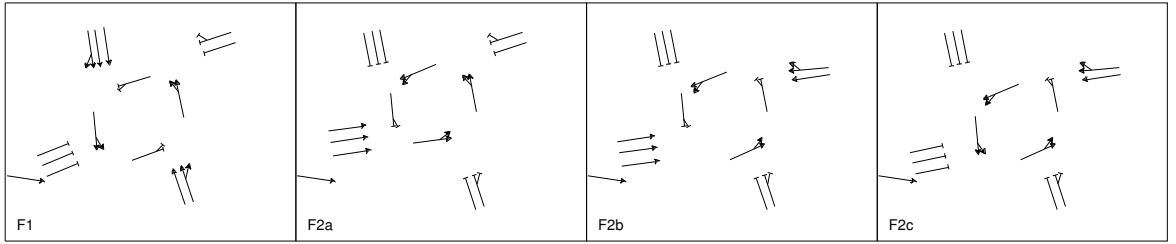
Şekil 21'de K3 kavşağının cadde görünümü, Şekil 22'de sayım yapılan kollardaki şeritler ve trafik akım yönleri, Şekil 23'te mevcut faz diyagramı ve Şekil 24'te mevcut sinyal grup diyagramları verilmiştir. Tablo 4'te bu kavşağa ait trafik hacim sayımları verilmiştir.



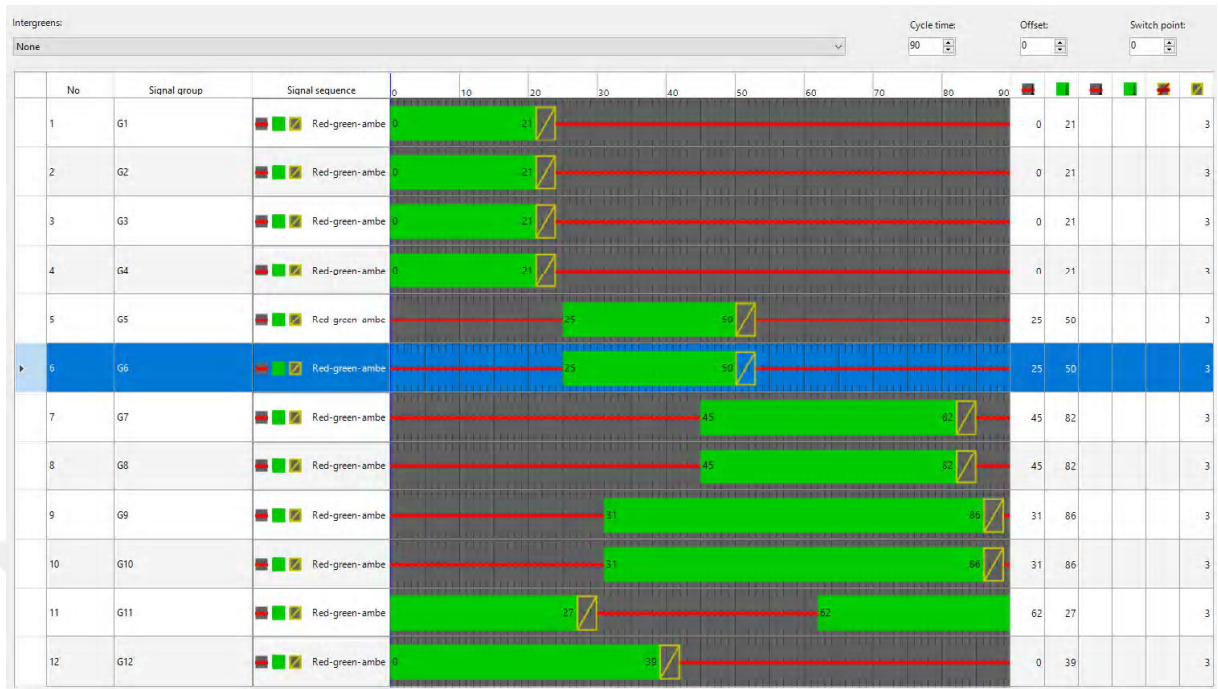
Şekil 24. K3 kavşağının cadde görünümü



Şekil 25. K3 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri



Şekil 26. K3 kavşağına ait faz diyagramı



Şekil 27. K3 kavşağına ait sinyal grup diyagramı

Tablo 4'te görüleceği üzere, sabah saatlerinde kavşaktaki trafik yükü 8100 taşıt olup saatlik trafik hacmi 4000 taşıt/saat değerinin üzerindedir.

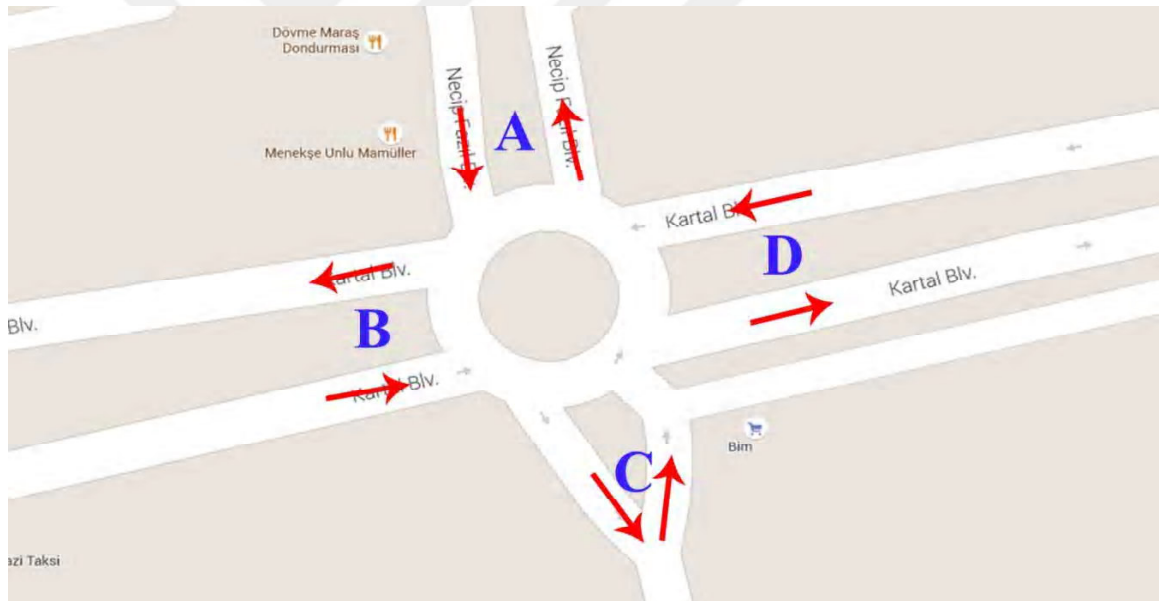
Tablo 4 K3 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları

	A				B				C				D				TOPLAM
	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	
A	0	0	0	0	148	12	11	36	820	35	7	91	297	5	3	48	1512
B	241	16	12	21	7	1	0	4	276	20	4	25	3043	75	51	185	3981
C	155	13	1	32	109	17	4	9	7	0	0	0	157	13	11	43	572
D	128	11	0	45	1240	104	36	103	264	41	4	37	17	4	3	3	2040
TOPLAM	524	40	13	99	1504	135	51	152	1367	96	15	153	3515	97	67	279	8105

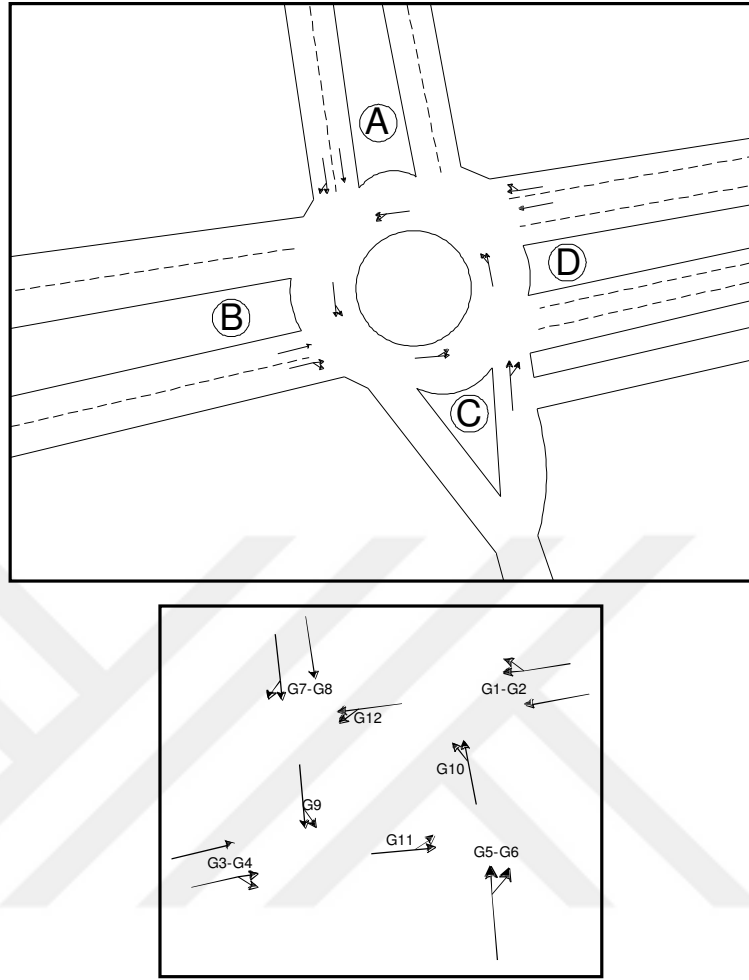
3.1.1.4. K4 Kavşağı (66 Nolu Battalgazi Kavşağı)

K4 Kavşağı dört kollu rotary adalı kavşak tipindedir. Özellikle sabah çok yoğun olan Battalgazi Kavşağı; Kartal Blv.'nin devamında Necip Fazıl Blv.'nin son noktası olup güney yaklaşım kolunda Haksınaz Sk. bulunmaktadır. Haksınaz Sk. sokak olmasına rağmen ara arterler arasındaki bağlantı konumunda bulunduğundan trafiğin çok aktif olduğu bir konumdur.

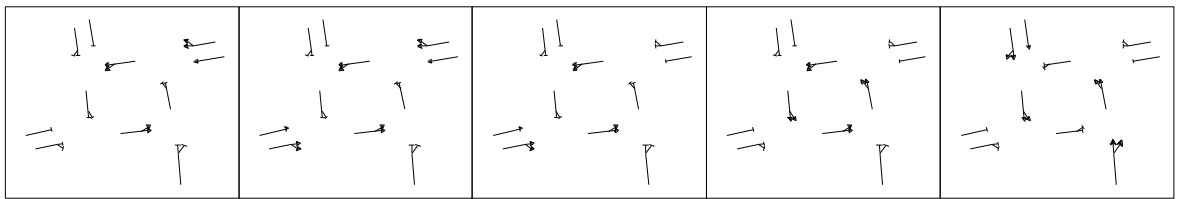
Şekil 25'te K4 kavşağının cadde görünümü, Şekil 26'da sayım yapılan kollardaki şeritler ve trafik akım yönleri, Şekil 27'de mevcut faz diyagramı ve Şekil 28'de mevcut sinyal grup diyagramı verilmiştir. Tablo 5'te bu kavşağa ait trafik hacim sayımları verilmiştir.



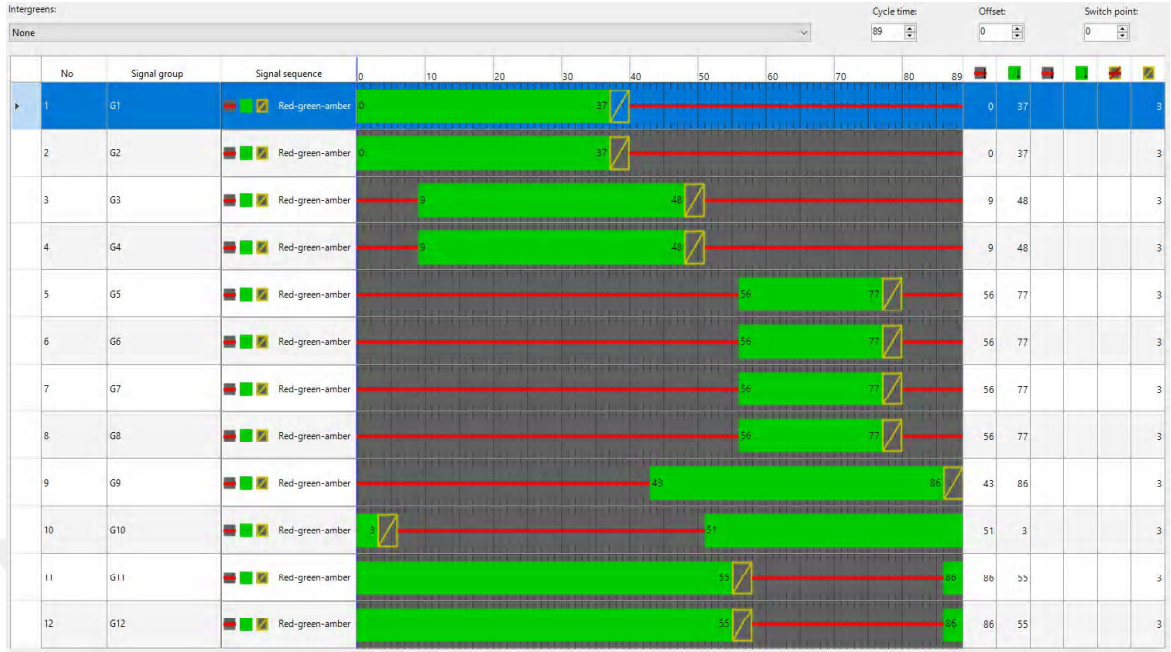
Şekil 28. K4 kavşağının cadde görünümü



Şekil 29. K4 kavşağının yaklaşım kollarındaki şeritler ve trafik akım yönleri



Şekil 30. K4 kavşağına ait faz diyagramı



Şekil 31. K4 kavşağına ait sinyal grup diyagramı

Tablo 5’te görüleceği üzere, sabah saatlerinde kavşaktaki trafik yükü 8200 taşıt olup saatlik trafik hacmi 4100 taşıt/saat civarındadır.

Tablo 5. K4 kavşağına ait sabah (07:00-09:00) trafik hacim sayımları

	A				B				C				D				TOPLAM
	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	Otomobil	Otobüs	Kamyonet	Minibüs	
A	31	0	0	8	151	15	0	24	215	3	21	41	341	16	24	59	948
B	179	16	27	41	7	0	0	3	109	17	13	39	1328	83	11	145	2017
C	541	17	11	85	548	4	16	39	3	0	1	0	125	0	0	12	1403
D	623	45	5	77	2627	29	9	272	85	8	28	20	0	0	0	0	3829
TOPLAM	1373	79	43	212	3332	48	25	337	412	28	64	100	1795	99	35	216	8197

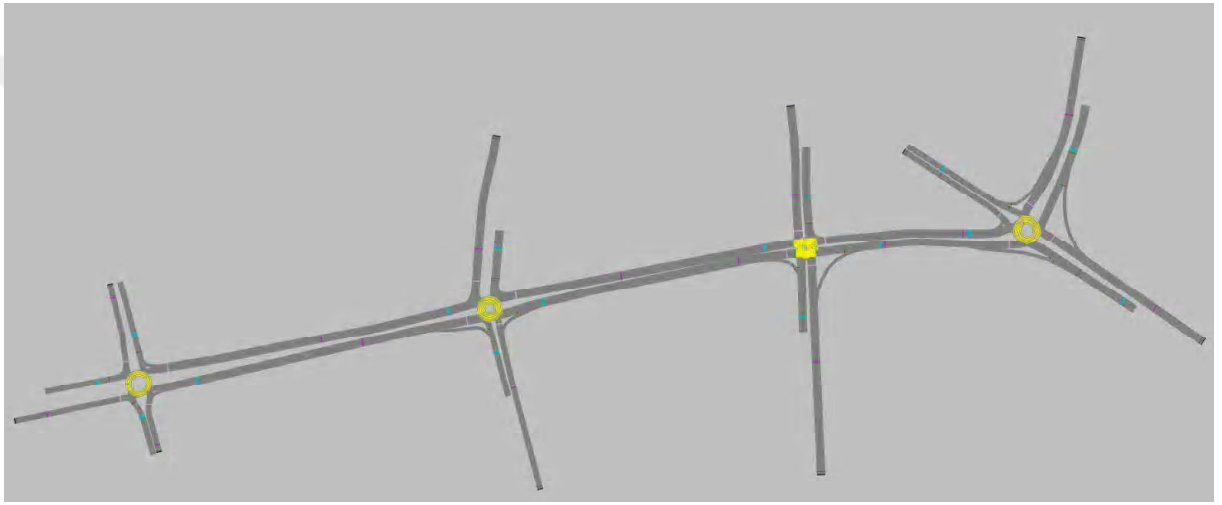
3.2. Algoritmalar ile Performans Değerleme

Tezin uygulama kısmında Bernstein Arama Diferansiyel Gelişim Algoritması (BSD) ile Standart Diferansiyel Gelişim Algoritması (sDE), Genetik Algoritma (GA) ve optimize

edilen sinyal sürelerinin halihazırda çalışma bölgelerinde kullanılan Vissig algoritmasının performansları karşılaştırılmıştır. Bu sürecin aşamaları bu bölümde ele alınacaktır.

3.2.1. Çalışma Alanının VISSIM Üzerinde Oluşturulması

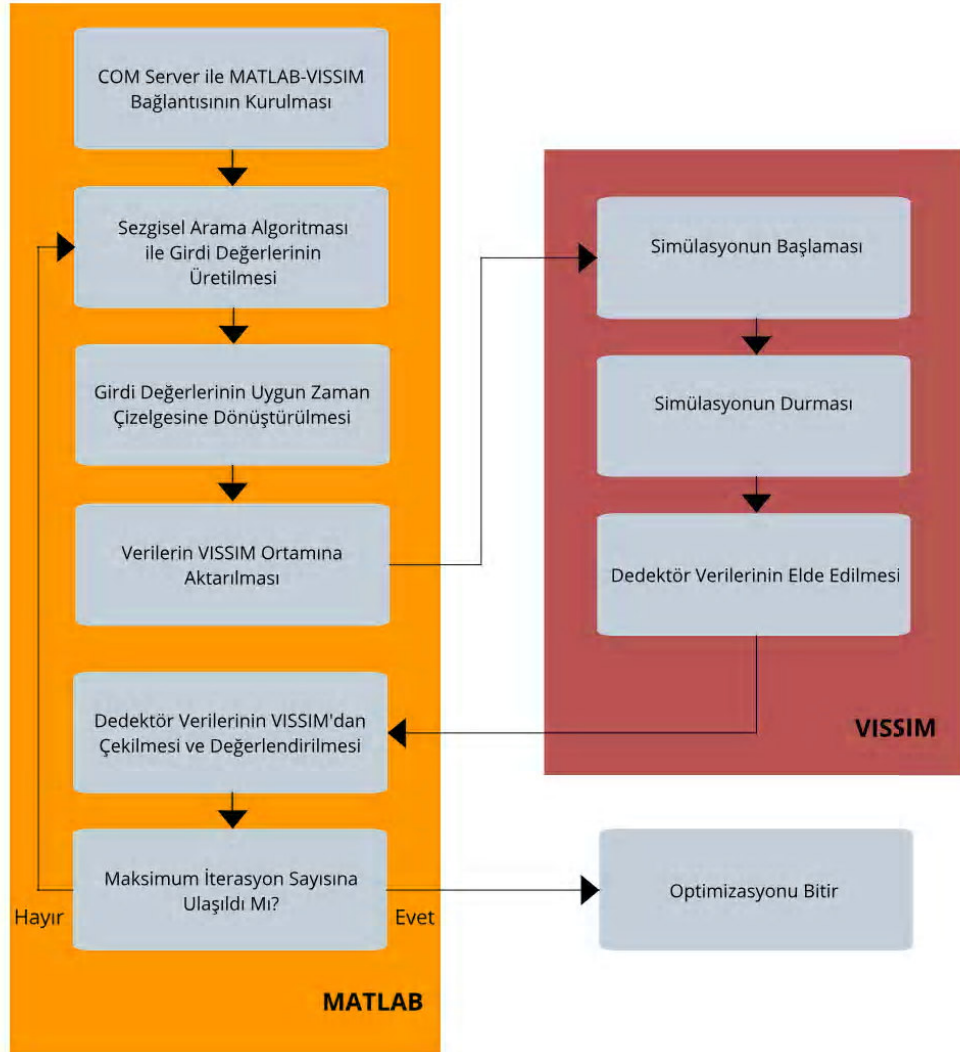
Kayseri Büyükşehir Belediyesi'nden alınan yol bilgileri dahilinde, gerekli harita altlıkları da kullanılarak VISSIM ortamında çalışma alanının Şekil 29'daki modeli oluşturulmuştur.



Şekil 32. Çalışma alanına ait VISSIM modeli

Çalışma alanında 4 adet kavşak ve 10 adet giriş ve 10 adet çıkış bulunmaktadır. Ağ üzerinde ana linkler ve bağlantı linklerinden oluşan 79 adet link bulunmaktadır. Her linke ait uzunluk, genişlik, şerit sayıları ve beklenen maksimum hızlar gibi değerler modellenmiştir. Ağa giriş olan link başlarına ölçümlerden elde edilen araç giriş değerleri eklenmiştir. Araç girişleri bir saatlik zaman dilimine göre sabit olarak belirlenebileceği gibi, çeşitli istatistik dağılımları ile de belirlenebilir. Ağda 4 kavşağa ait 1'er sinyal kontrolörü ve toplamda 45 adet düzenlenebilir sinyal grubu bulunmaktadır. Giriş ve çıkış linklerine toplamda 20 adet araç sayaç detektörü yerleştirilmiştir. Güzergâhların birbirine göre rölatif akış değerleri el ile düzenlenmiştir. Bu sayede gerçekte olduğu gibi araçların her linke eşit şekilde değil, gerçekte yoğun olan bölgelere doğru akması sağlanmıştır. Rastgele tohumlama sabit tutulduğu için her simülasyonda ağa saatte 10309 araç gitmektedir. Her linkin ağırlığı eşit olmadığından ve araçlar istatistiki dağılıma göre üretildiği 5 dakikada ağa giren araç sayısı 880'dir. Şekil 29'da kavşak bölgelerinde görülen sarı renkli şeritler hız düşürme bölgeleri olarak

planlanmıştır. Böylelikle araçlar bölgeye geldiklerinde sinyalleri yeşil bile olsa hızlarını düşürmektedir. Ayrıca sağa dönüşlerde karşıdan gelen araca göre geçiş üstünlüğü gözetilmiştir. Şekil 33'te uygulamanın akış diyagramı verilmiştir. Tüm bu adımlar ilerleyen bölümlerde ayrıntılı olarak incelenecektir.



Şekil 33. Uygulamanın akış diyagramı

3.2.2. Verilerin COM yardımı ile MATLAB Ortamına Aktarılması ve Simülasyon Yönetimi

VISSIM programına herhangi bir programlama dili ile uygun müdahalelerin yapılabilmesi için öncelikle COM Server kaydının yapılması gereklidir. Bu kaydın

yapılmasının ardından MATLAB ortamında VISSIM programını başlatıp bağlantı kurmak için basitçe aşağıdaki kod kullanılır:

```
VISSIM=actxserver('VISSIM.VISSIM');
```

Program açıldıktan sonra ilgili çalışma dosyasının çağrılması için şu kod dizisi kullanılabilir:

```
VISSIM.LoadNet('C:\Users\Public\Documents\PTV Vision\PTV VISSIM 11\myProjects\tezalani.inp0')
VISSIM.LoadLayout('C:\Users\Public\Documents\PTV Vision\PTV VISSIM 11\myProjects\tezalani.layx')
```

İlgili dosya çağrıldıktan sonra simülasyon MATLAB üzerinden çalıştırılabilir hale gelir. Fakat simülasyondan daha doğru bir optimizasyon değeri elde edilmesi için rastgele tohumlama değeri sabit tutulmuştur. Böylelikle her simülasyon başladığında aynı trafik senaryosu oluşmaktadır. Rastgele tohumlamayı sabitlemek için şu kod kullanılmaktadır:

```
Random_Seed = 42;
set(VISSIM.Simulation, 'AttValue', 'RandSeed', Random_Seed);
```

Simülasyon süresinin belirlenmesi için ise şu kod kullanılmaktadır.

```
Sim_break_at = 300;
set(VISSIM.Simulation, 'AttValue', 'SimBreakAt', Sim_break_at);
VISSIM.Simulation.RunContinuous;
```

Burada 300 ile belirtilen değer simülasyon adımdır. Bu adım değeri minimum saniyenin 1/10'u kadar olabilir. Fakat bunu 1 saniye olarak ayarlamak mümkündür. Dolayısı ile 300 adım, 5 dakikaya tekabül etmektedir. Bu nedenle tezdeki simülasyon örneklemeleri 5 dakikalık periyotlar ile ölçülmüştür.

Simülasyonun başlatılması, gerektiğinde adım adım ilerletilmesi ve durdurulması için sırayla aşağıdaki kodlar kullanılmaktadır:

```
VISSIM.Simulation.RunContinuous;
VISSIM.Simulation.RunSingleStep;
VISSIM.Simulation.Stop;
```

Simülasyon esnasında trafik sinyal lambalarına müdahale edebilmek için şu kod dizisi kullanılabilir:

```
SC_number = 1; % SC = Sinyal Kontrolörü Numarası
SG_number = 1; % SG = Sinyal Grup Numarası
SignalController = VISSIM.Net.SignalControllers.ItemByKey(SC_number);
SignalGroup = SignalController.SGs.ItemByKey(SG_number);
new_state = 'GREEN'; %Bu değer 'GREEN', 'RED', 'AMBER', 'REDAMBER' değerlerini almabilmektedir.
set(SignalGroup, 'AttValue', 'SigState', new_state);
```

Kod yapısından da anlaşılacağı üzere yeni durumun alacağı ‘GREEN’, ‘RED’, ‘AMBER’ ve ‘REDAMBER’ değerleri sinyallerin renkleridir. Bu kod ‘GREEN’ olarak çalıştırıldığında ilgili sinyal grubu simülasyon sonuna kadar yeşil renkte olacaktır. Dolayısı ile yeşil süresinin sınırlandırılması için gerektiğinde kırmızıya dönüştürecek bir çizelgeleme algoritmasına ihtiyaç bulunmaktadır. Çizelgeleme algoritmasının yapım aşamaları Bölüm 3.2.4’te anlatılacaktır.

Son olarak araç sayım verilerinin elde edilmesi gerekmektedir. Bunun elde edilebileceği kod dizini ise şu şekilde olabilir:

```
All_Measurement = VISSIM.Net.DataCollectionMeasurement.GetAll;
for count = 1 : length(All_Measurement)
    DC_measurement = VISSIM.Net.DataCollectionMeasurements.ItemByKey(count);
    No_Veh        = get(DC_measurement, 'AttValue', 'Vehs    (Avg,1,All)');
end
```

Fakat detektör verileri ağı a ait bir değer olduğu için bu veriler simülasyonun çalıştığı esnada elde edilemez. Verilerin elde edilebilmesi için belirli aşamalarla simülasyonun durdurulması gereklidir.

3.2.3. Sezgisel Arama Algoritmaları ile Sinyal Değerlerinin Optimize Edilmesi

Tez için belirlenen çalışma alanında giriş değerleri olarak 4 adet sinyal kontrolörüne ait döngü süreleri, 45 sinyal grubuna ait yeşil yanma başlangıç zamanları ve 45 yeşil sinyal süresi zamanları olmak üzere toplam 94 parametrenin optimize edilmesi gerekmektedir. Yeşil sinyal başlangıç süresinin, yeşil sinyal süresi ve 3 saniyelik sarı sinyal yanma süresi ile toplamı kırmızı sinyal başlangıç zamanını vermektedir.

Çıkış performans değeri olarak ağı giren ve ağıdan çıkan toplam araç sayılarının oranı dikkate alınmaktadır. Ayrıca farklı çıkış değerleri istenirse, örneğin; linklerdeki ortalama kuyruklanma sayıları ve ortalama seyahat süreleri gibi bilgiler ek olarak alınabilmektedir.

Sezgisel arama algoritmaları önceki bölümlerde de bahsedildiği üzere başlangıç değeri olarak rastgele sayılar alabilir. Fakat optimumum çözüme daha hızlı yakınsamak ve algoritmanın fazla aracı çıkışlara ulaştırabilmesi için, görece daha az yoğun olan

linklerdeki aracı simülasyon süresi sonuna kadar geciktirmemesi adına giriş parametreleri olan döngü süresi, yeşil faz başlangıç süreleri ve yeşil faz süreleri değerlerine bazı kısıtlamaların eklenmesi gereklidir. Bunlardan en önemlisi üretilen tüm değerlerin pozitif tamsayı değer almasıdır. Değerlerin alt ve üst sınırları tecrübeye dayanan bilgilere göre belirlenmektedir. Dört kavşakta döngü süreleri için belirlenen alt ve üst sınırları Tablo 6’da gösterilmektedir.

Tablo 6. Döngü süreleri için belirlenen alt ve üst sınırlar

	K1	K2	K3	K4
Alt Sınır (sn.)	90	90	80	80
Üst Sınır (sn.)	120	120	100	100

Yeşil Faz Başlangıç Süreleri: Burada herhangi bir sınır bulunmamaktadır. Tüm yeşil fazlar 0. saniyeden itibaren başlayabilir.

Yeşil Faz Süreleri: Her sinyal lambası için en az 10 saniye ve en fazla döngü süresi kadardır.

3.2.4. Değerlerin VISSIM Ortamına Aktarılması ve Simülasyon

Üretilen tüm değerler MATLAB üzerinde geliştirilen algoritma sayesinde VISSIM için kullanılabilir hale gelmektedir. Önceki bölümde bahsedildiği üzere VISSIM programı üzerinde sinyallerin yeşil, sarı veya kırmızıya çevrilmesi, ilgili sinyal grubunun simülasyon sona erene kadar aynı fazda kalmasına neden olmaktadır. Sinyal grubunun belirli bir süre boyunca tercih edilen herhangi bir sinyale dönüşmesi için sürenin başında ve sonunda simülasyona müdahale edilmesi gerekmektedir. Simülasyona her adımda müdahale etmek mümkün olsa da bu durum simülasyon süresini çok uzatmaktadır. Süreyi hızlandırmak ve gerektiğinde doğru müdahalelerin yapılması adına bir çizelgeleme algoritması geliştirilmiştir.

Algoritma temelde, çizelgelemeyi oluşturma aşamaları ve oluşturulan çizelgedeki ilgili zamanda VISSIM ortamına müdahale etme aşamaları olmak üzere iki kısımda incelenebilir.

Çizelgelemeyi oluşturma aşamaları şu şekildedir:

- 1- Üretilen döngü uzunlukları, yeşil faza dönme başlangıç zamanları ve yeşil faz yanma süreleri elde edilir.
- 2- Önce tüm lambalar kırmızı faza çevrilir.
- 3- Yeşil faza dönme başlangıç zamanları ile yeşil faz yanma sürelerinin toplamında elde edilen zaman, kırmızı faza dönme başlangıç zamanı olarak elde edilir.
- 4- Elde edilen kırmızı faza dönme başlangıç zamanı döngü uzunluğunu aşıyorsa fazlalık olan süre döngünün başına kaydırılır.
- 5- Simülasyon süresince 4 kavşakta toplam kaç döngü oluşacağı hesaplanır.
- 6- Oluşacak döngülerdeki tüm sinyal gruplarının yeşil ve kırmızı faza dönme zamanları hesaplanır.
- 7- İlgili sinyal kontrolörü ve sinyal grubuna ait faz bilgisi yeşil ise 1, kırmızı ise 2 olarak indekslenir.
- 8- Tüm sinyallere ait zaman bilgisi matris haline getirilir
- 9- Bu matris, zaman bilgisine göre küçükten büyüğe satırlar halinde sıralanır.
- 10- Sonuç olarak zaman ve faz bilgilerini gösteren bir çizelgeleme matrisi elde edilir. Çizelgeleme matrisine ait bir örnek Tablo 7’de verilmiştir.

Tablo 7. Çizelgeleme matrisi

Müdahale Zamanı	Sinyal Kontrolör N.	Sinyal Grup N.	Faz
21	2	2	1
53	1	9	2
60	1	11	2
72	1	5	2
77	3	6	2
82	1	8	1
88	4	9	1
91	1	5	1
91	2	3	2
98	2	1	2
104	1	11	1
104	2	3	1

105	1	8	2
105	2	1	1
110	1	3	1
111	1	2	1
117	1	3	2
118	1	7	2
118	2	2	1
123	1	10	2
132	1	10	1
134	3	1	1
139	1	4	1
139	1	7	1
145	1	6	1
147	1	4	2
154	4	2	2
157	1	1	2
161	1	2	2
163	2	2	2
164	1	2	2
173	1	2	2
178	1	2	2
183	1	8	1
189	3	9	1
192	1	5	1
205	4	11	1
206	1	2	2
211	2	3	1
212	1	2	1
218	1	2	2
219	1	2	2
224	1	2	2
233	1	10	1
233	2	2	2
235	1	1	1
240	4	4	1
240	1	7	1
240	2	2	2
246	1	6	1
246	2	3	1
247	2	1	1
248	1	2	2
258	3	2	2
260	2	2	1
265	1	2	2

Tablo 7’de, Matrisin ilk sütunu müdahale zamanını, ikinci sütun sinyal kontrolör numarasını, üçüncü sütun sinyal grup numarasını, dördüncü sütun ise faz türünü belirtmektedir. Matristeki dördüncü sütun değeri 1 ise faz yeşile, 2 ise faz kırmızıya döndürülür. Örneğin Tablo 7’de görülen ilk satır için 21. simülasyon adımı (saniyede) 2. sinyal kontrolöründeki (K2 kavşağı) 2 numaralı sinyal grubu (B kolu) sinyaller yeşile döndürülecektir (1 kodu yeşil olduğu için). Bu simülasyonlarda sarı fazı kırmızı faz başlamadan 3 saniye boyunca yanmaktadır.

Üretilen çizelge matrisinin VISSIM ortamında kullanılabilmesi için şu adımlar izlenmektedir:

- 1- Simülasyon başlatılır.
 - 2- Simülasyon, matrisin öncelikle ilk satırındaki ve ilk sütunundaki süreye kadar devam ettirilir.
 - 3- İlgili süreye geldiğinde simülasyon duraklamaya alınır.
 - 4- Duraklamanın olduğu satırda ikinci, üçüncü ve dördüncü sütunlar kontrol edilir.
 - 5- İkinci sütunda bulunan sinyal kontrolörüne ait üçüncü sütunun sinyal grubu, dördüncü sütundaki faz değerine dönüştürülür.
 - 6- Simülasyon devam ettirilir.
 - 7- Simülasyon süresine ulaşıldı ise simülasyon durdurulur. Aksi durumda 2. adıma gidilir ve bir sonraki satır için aynı işlemler uygulanır.
 - 8- Simülasyon durduğu zaman detektör bilgileri toplanır. (Detektör bilgileri simülasyon esnasında elde edilememektedir. Bu durum gerçek zamanlı adaptif optimizasyon yapılmasına kısıt oluşturmaktadır.)
 - 9- Detektör verilerinden ağa giren ve çıkışa ulaşabilen araçların oranı hesaplanır.
- Tez kapsamında oluşturulan simülasyona ait görsel Şekil 34’te görülmektedir.



Şekil 34. VISSIM Simülasyon Ortamından bir Görüntü

3.3. Bulgular

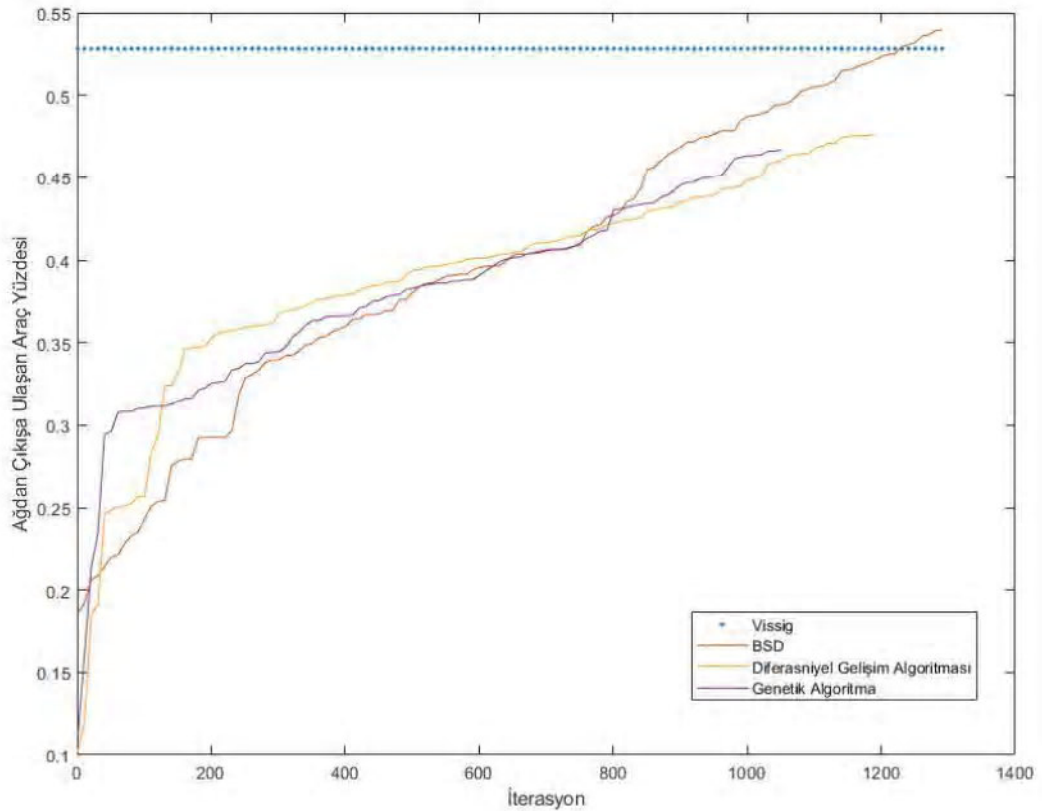
Bu çalışma ile literatüre yeni kazandırılan BSD Algoritmasının trafik sinyal optimizasyonu performansı hesaplanmıştır. Performansın ölçeklenmesi ve doğrulanması adına literatürde en iyi bilinen sezgisel arama algoritmalarından olan Standart Diferansiyel Gelişim Algoritması (sDE) ve Genetik Algoritmanın (GA) yanında istatistik temelli olan Vissig yöntemi kıyaslanmıştır. Yapılan uygulamalar ortalama 100 defa 5000 iterasyon ile koşturulmuştur.

Sonuçların doğru değerlendirilebilmesi adına her simülasyon 5 dakika üzerinden hesaplanmış ve bu süre içerisindeki rastgele tohumlama değeri sabit tutularak her iterasyonda aynı senaryo uygulatılmıştır. Bu senaryo çerçevesinde 5 dakika içinde ağa 880 araç girmektedir. Performans olarak ise çıkışa ulaşabilen araçların oranlarına bakılmaktadır. BSD, Vissig, sDE ve GA'nın en iyi performans değerleri ve çıkışa ulaşan araç sayıları Tablo 8'de verilmiştir.

Tablo 8. Algoritmalarla ait en iyi performans deęerleri ve ıkıřa ulařan ara sayıları

	Giriřteki Ara Sayısı	Performans (%)	ıkıřa Ulařan Ara Sayısı
BSD	880	53.9205	475
Vissig	880	52.8409	469
sDE	880	47.6342	420
GA	880	46.7040	411

Algoritmaların iterasyon sayılarına gre bařarı oranları ise řekil 31’de verilmiřtir. Buradan da anlařılabileceęi zere ortalama 1000-1300 iterasyonda performansların eřik deęere ulařtıęı grlmektedir.



řekil 35. Algoritmaların iterasyon sayılarına gre bařarı oranları

Tablo 9’ da BSD, sDE ve GA’nın 5000 iterasyon boyunca deęiřim yařanan performans deęerleri verilmiřtir. Deęiřim yařanan deęerlerin verilmesinin sebebi algoritmaların

bazen yüzlerce adım iyileşme sağlayamamasıdır. Tablo 9'dan görüldüğü üzere sDE ve GA, BSD'ye göre daha az veriye sahiptir. Bunun sebebi sDE ve GA'nın sonrasında daha iyi değerlere ulaşamamasıdır.

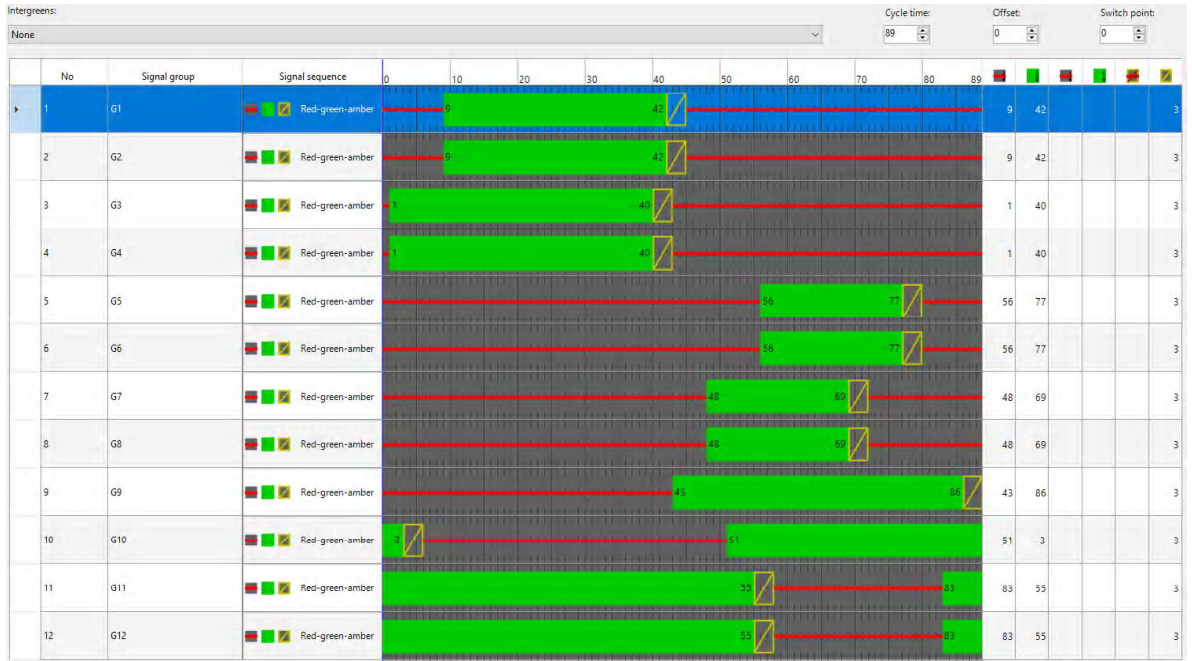
Tablo 9. Algoritmaların kırılma yaşanan performans değerleri

BSD	sDE	GA
0,186246	0,102040816	0,11143695
0,192308	0,119496855	0,160660661
0,206044	0,184636119	0,213204952
0,208791	0,191196699	0,234417344
0,214286	0,246298789	0,294486216
0,21981	0,248655914	0,296105695
0,221167	0,250336474	0,308126636
0,227951	0,250673854	0,308322646
0,232653	0,252702703	0,308436501
0,234735	0,256868132	0,310664049
0,243537	0,256868132	0,310927291
0,251018	0,283983849	0,31181448
0,253731	0,293405114	0,312038804
0,254098	0,324213406	0,312077427
0,275441	0,324213406	0,313309144
0,278155	0,331536388	0,314746662
0,279512	0,346361186	0,316232307
0,279512	0,34682861	0,31634245
0,292617	0,347297297	0,321405151
0,292617	0,348171849	0,322816064
0,292617	0,352882005	0,325691709
0,292617	0,356010996	0,32619216
0,292617	0,356722064	0,326903227
0,296644	0,35750331	0,333630239
0,318244	0,357686689	0,334370598
0,328748	0,359458448	0,337304808
0,330827	0,359495975	0,337491933
0,333333	0,360302082	0,338530819
0,338346	0,360767867	0,343683293
0,33947	0,361679824	0,344316633
0,339585	0,367274104	0,344442918
0,34224	0,368689448	0,347139827
0,342326	0,36961511	0,352884166
0,344842	0,370679855	0,356292067
0,348455	0,372392576	0,360437943
0,349595	0,374071877	0,36322596
0,353027	0,376038457	0,36333958
0,353806	0,376250403	0,365445888
0,356523	0,377925095	0,365893258
0,358261	0,378209021	0,366028306
0,359887	0,37862903	0,366121174
0,363842	0,379085535	0,366466593
0,364454	0,382117778	0,370833072
0,366907	0,383609054	0,371806402
0,366911	0,384593699	0,374940037
0,367459	0,385071044	0,375332922
0,369178	0,386555888	0,377543663

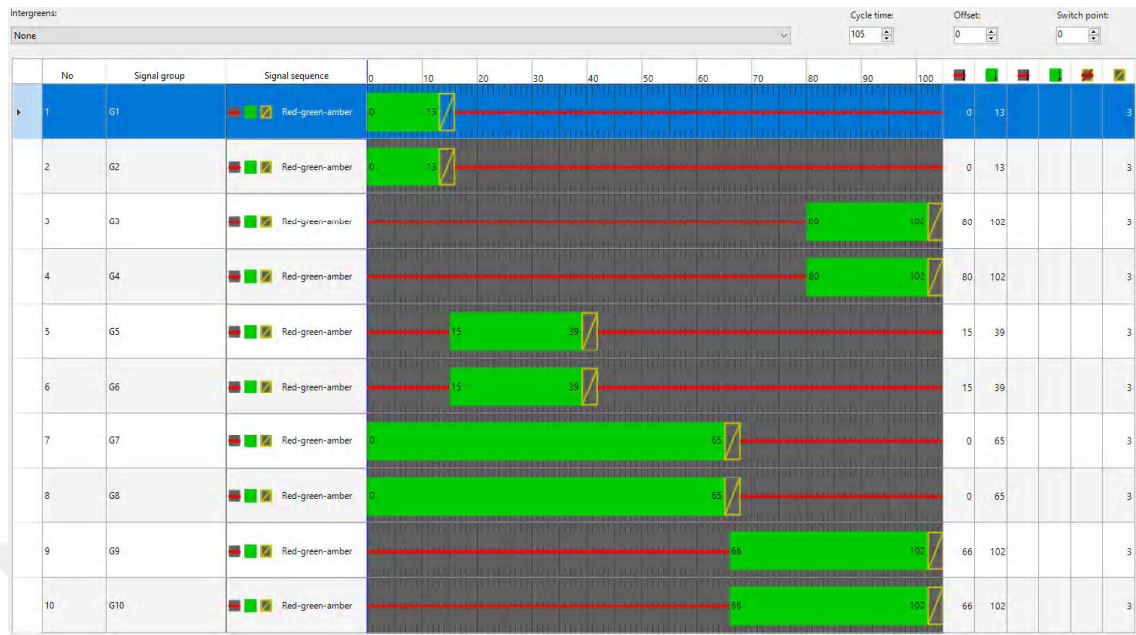
0,369328	0,386687552	0,37850236
0,375701	0,387365537	0,378911418
0,375889	0,391492346	0,382387851
0,380398	0,393805021	0,383052573
0,384092	0,394170336	0,384280823
0,386037	0,396021909	0,385072496
0,386347	0,39625589	0,385855085
0,388185	0,396565648	0,38626763
0,390461	0,397335955	0,38634649
0,390697	0,398312272	0,387480032
0,391517	0,398671696	0,387495107
0,391517	0,400225833	0,388299579
0,394215	0,401041158	0,388379288
0,395609	0,401133253	0,391088273
0,396324	0,401256564	0,393924191
0,396595	0,402550751	0,396303925
0,3968	0,403350038	0,399204438
0,399778	0,403596444	0,400504084
0,403023	0,404535943	0,401488786
0,40362	0,40488723	0,401783816
0,403659	0,407042566	0,403603858
0,404468	0,410071413	0,403983894
0,405529	0,410351885	0,404542623
0,406302	0,410633261	0,405722062
0,406441	0,411241416	0,406224819
0,406441	0,412247832	0,406344469
0,407372	0,414014949	0,406606433
0,408303	0,414092112	0,408205803
0,408905	0,414806807	0,41024578
0,417403	0,417633372	0,413133787
0,420342	0,418276474	0,41510284
0,42124	0,418931752	0,417453892
0,425841	0,421656221	0,418022863
0,426959	0,422003524	0,430394026
0,429162	0,423217776	0,430626066
0,43549	0,424196566	0,431652937
0,436769	0,424721621	0,432833152
0,443053	0,425358144	0,433550332
0,45522	0,429688406	0,434245604
0,456388	0,429939418	0,434505119
0,460268	0,43140511	0,437811727
0,463886	0,431735866	0,439138641
0,466481	0,432442392	0,442171597
0,468891	0,434736794	0,44550719
0,471847	0,436245986	0,446765784
0,472193	0,437840136	0,447245095
0,474728	0,43790705	0,449451407
0,475068	0,438545625	0,449694853
0,476557	0,439127651	0,450764783
0,478611	0,442456941	0,451259874
0,478611	0,443283714	0,456138281
0,478629	0,443374333	0,461487357
0,48526	0,445091779	0,462841928
0,48724	0,448097224	0,463416993
0,487955	0,449240246	0,463612862
0,488775	0,45124197	0,464082287
0,490137	0,45831923	0,466286145

0,49406	0,459534242	0,466426513
0,494241	0,460868998	0,467040212
0,49548	0,463119275	
0,498382	0,464303385	
0,502421	0,464527813	
0,503643	0,464777364	
0,50533	0,468260522	
0,505546	0,469362522	
0,506842	0,471025421	
0,508873	0,471198557	
0,514847	0,474765008	
0,51515	0,475179948	
0,516491	0,475649093	
0,518235	0,475711069	
0,519686	0,475958305	
0,52137	0,476342512	
0,523889		
0,524751		
0,525464		
0,529617		
0,530623		
0,532077		
0,53606		
0,536663		
0,539205		

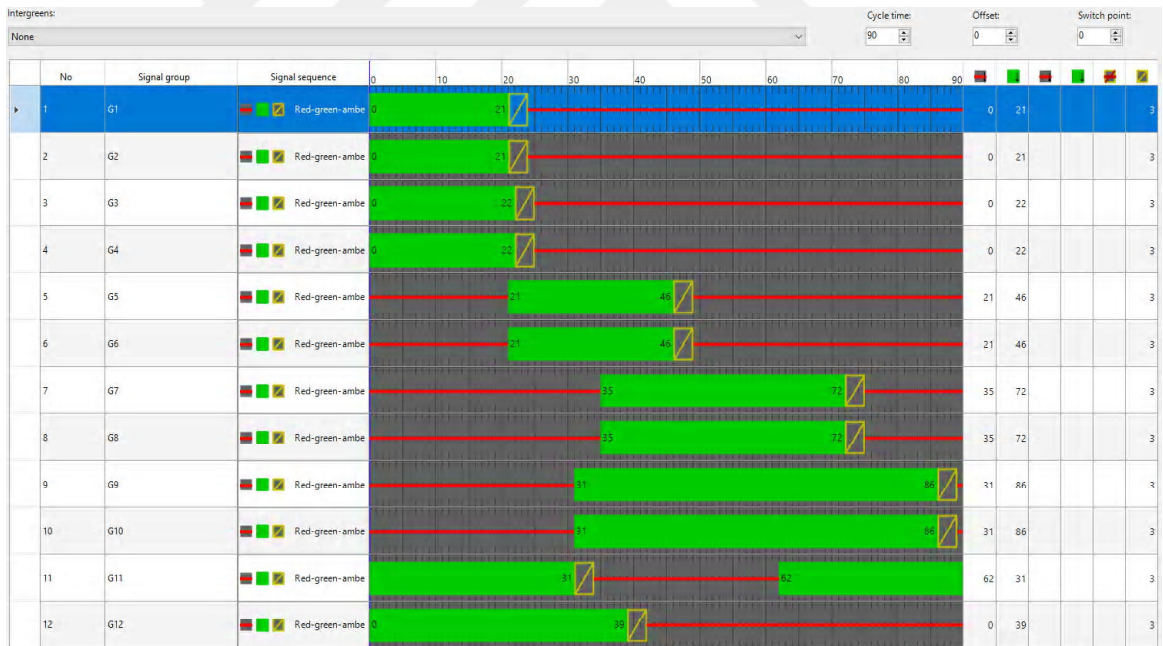
BSD algoritmasının mevcut durumdaki en iyi performansında verdiği faz değerleri Şekil 32-35'te verilmiştir.



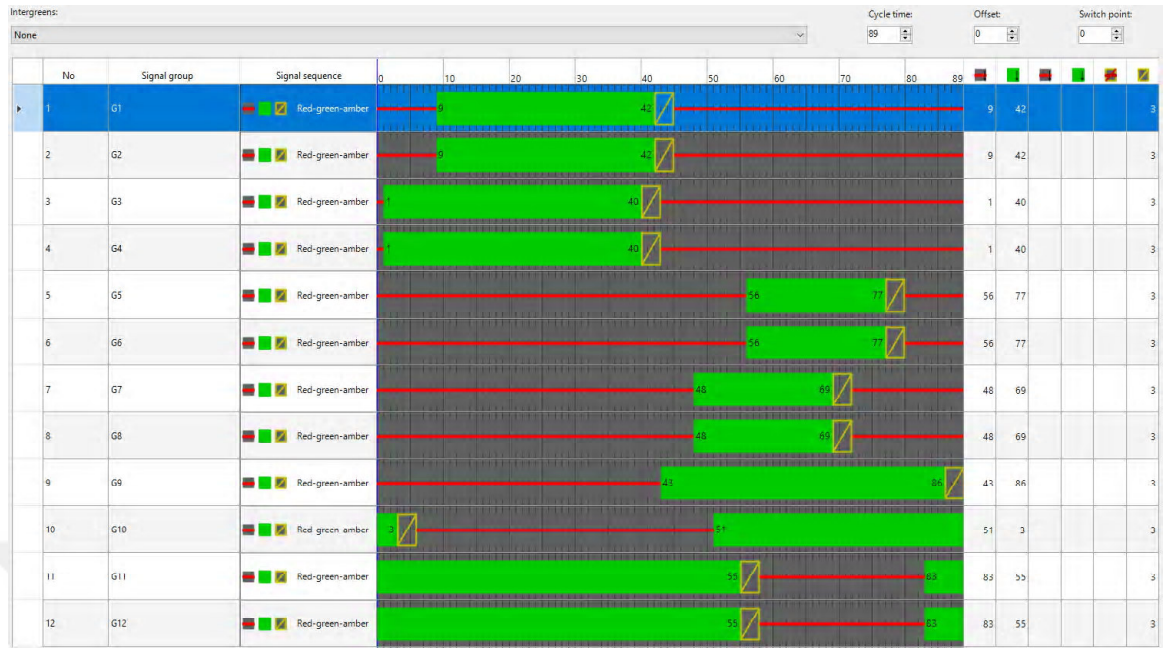
Şekil 36. BSD algoritması ile üretilen K1 kavşağına ait yeni sinyal grup diyagramı



Şekil 37. BSD algoritması ile üretilen K2 kavşağına ait yeni sinyal grup diyagramı



Şekil 38. BSD algoritması ile üretilen K3 kavşağına ait yeni sinyal grup diyagramı

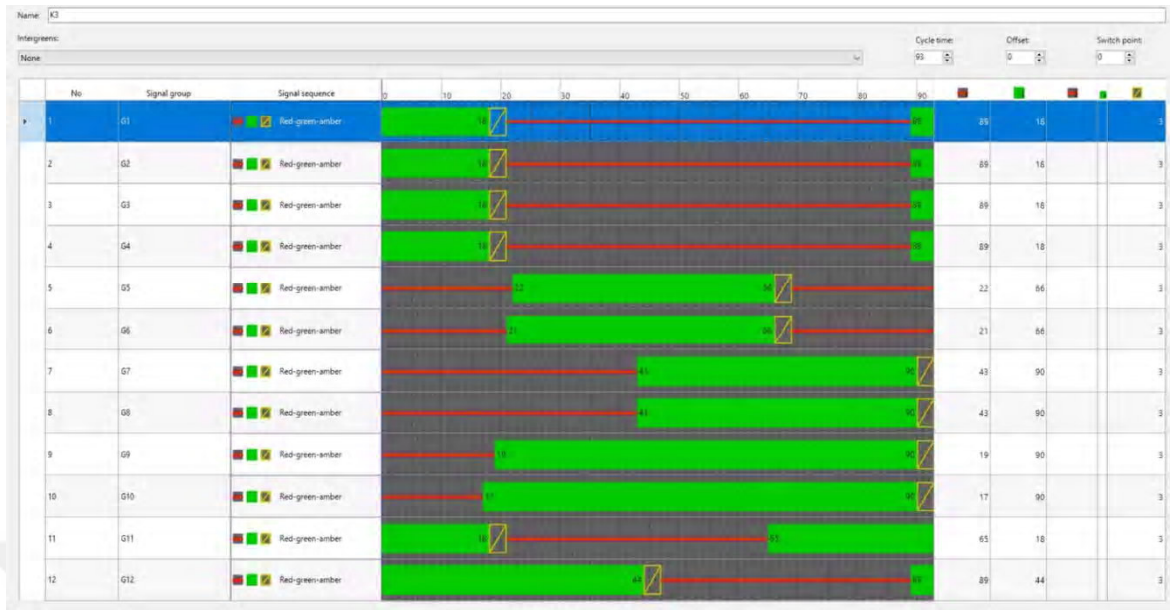


Şekil 39. BSD algoritması ile üretilen K4 kavşağına ait yeni sinyal grup diyagramı

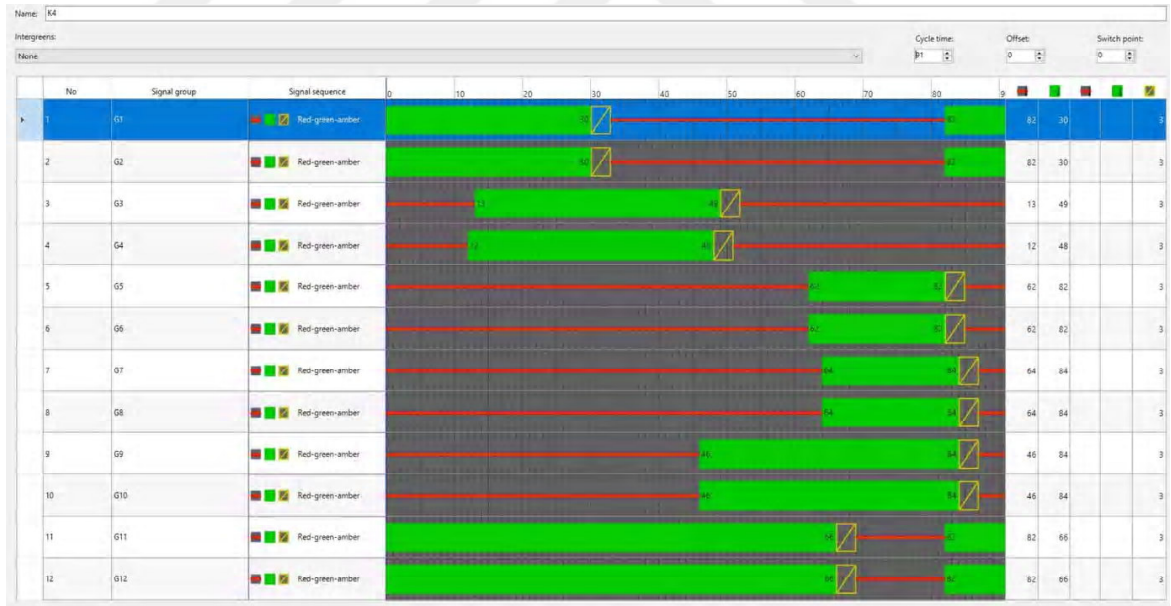
İkinci bir uygulamada ise yine aynı model üzerinde algoritmaların linklerin aşırı doygun olduğu durumlarda verdiği tepkiler gözlenmiştir. Aşırı doygun durum çalışma alanında henüz gerçekleşmediğinden gözleme dayalı değil tahmine dayalı simülasyon ile gerçekleştirilmiştir. Aşırı doygun durumun gerçekleşmesi için rölatif akış değerleri sabit tutularak ağa giren saatlik araç sayısı 10309'dan 19510'a çıkartılmıştır. Yapılan uygulamalar yine ortalama 100 defa 5000 iterasyon ile koşturulmuştur. Bu süreçte 5 dakikada ağa girebilen maksimum araç sayısı 1521 olarak ölçülmüştür. Aşırı doygun durumda BSD, Vissig, sDE ve GA'nın en iyi performans değerleri ve çıkışa ulaşan araç sayıları Tablo 10'da verilmiştir.

Tablo 10. Aşırı Doygun Durumda Algoritmaların Gösterdiği En İyi Performans Değerleri

	Girişteki Araç Sayısı	Performans (%)	Çıkışa Ulaşan Araç Sayısı
BSD	1521	49.0467	746
Vissig	1521	46.2196	703
sDE	1521	42.2748	643
GA	1521	40.6312	618



Şekil 42. BSD algoritmasının aşırı doygun durumda ürettiği K3 kavşağına ait yeni sinyal grup diyagramı



Şekil 43. BSD algoritmasının aşırı doygun durumda ürettiği K4 kavşağına ait yeni sinyal grup diyagramı

BÖLÜM 4

SONUÇLAR VE TARTIŞMA

Tez çalışması kapsamında Kayseri şehir merkezinde bulunan 4 kavşaktan oluşan bir trafik ağında, sezgisel arama algoritmaları ile trafik sinyal optimizasyonu başarılı bir şekilde uygulanmış ve sonuçlandırılmıştır. Buna göre BSD algoritmasının tez kapsamında öngörülen koşullar çerçevesinde en başarılı çıkış performans değerini verdiği görülmüştür.

Sezgisel süreç ile optimizasyon iteratif yapıda olduğu için gerçek trafik üzerinde yapılması olanaksızdır. Bu sebeple gerçek trafik sayım değerleri simüle edilerek simülasyon sonunda ağı terk eden araç sayısının ağa giren araç sayısına oranının maksimize edilmesine çalışılarak trafik sinyallerinin optimum değerlerinin bulunması uygulama kapsamında gerçekleştirilmiştir.

Optimizasyon sürecinde trafik simülasyonları için VISSIM programı kullanılmıştır. VISSIM kullanılmasının en önemli sebeplerinden biri halihazırda Kayseri Büyükşehir Belediyesi tarafından kullanılıyor olması ve çalışma alanındaki sinyallerin VISSIM programının Vissig algoritması ile optimize edilmiş olmasıdır. Bunun yanında VISSIM araç takibi, şerit değiştirme gibi mikroskobik simülasyon mimarilerini çok başarılı olarak temsil etmektedir.

VISSIM programı COM arayüzü sayesinde pek çok programlama dili ile manipüle edilebilmektedir. Fakat manipülasyon kodlarının çok sade yapıda olmasından dolayı, trafik sinyallerinin başlangıç sürelerini gösteren bir çizelgeleme programına ihtiyaç duyulmaktadır. Bu tez çalışması kapsamında, bahsedilen optimizasyonların yapılabilmesi adına, bir çizelgeleme programı da geliştirilmiştir. Ayrıca VISSIM simülasyon ortamında kullanılan detektör verileri anlık olarak manipüle edilememektedir. Dolayısıyla detektör verileri üzerinden anlık öz yinelemeli trafik

sinyal optimizasyonu yapılamamaktadır. Bu sebeple simülasyon sürecindeki iterasyon süreleri 5 dakika ile sınırlandırılmıştır. Bunun yanında VISSIM mikroskobik olarak tüm araçların anlık değişken değerlerine ulaşma imkânı vermektedir. Bu veriler kullanılarak anlık öz yinelemeli trafik sinyal optimizasyonu yapılabilir. Halihazırda gerçek araç verileri anlık olarak toplanamadığından bu uygulama yalnızca teorik bir çalışmanın ürünü olabilir. Ek olarak 5 dakikadan daha kısa sürede öz yineleme yapan bir sistemde sıklıkla histerezis görülmesi muhtemeldir.

Optimizasyon sürecinde 4 kavşakta bulunan toplam 45 sinyal grubuna ait trafik sinyallerinin döngü süresi, yeşil sinyalin başlangıç zamanı ve yeşil sinyal süresi olmak üzere toplam 94 parametre optimize edilmiştir. Fakat trafik sinyal optimizasyonu için gerekli olan asıl bilgi yeşil ve kırmızı sinyallerin başlangıç zamanıdır. Yeşil sinyalin başlangıç zamanı, yeşil sinyal süresi ve 3 saniyelik sarı sinyal süresinin toplamı, kırmızı sinyallerin başlangıç zamanını vermektedir. Fakat elde optimizasyon iterasyonları sırasında üretilen sinyal değerlerinin simülasyonda uygulanması için standart bir teknik bulunmamaktadır. Ayrıca VISSIM programı dahilinde sezgisel optimizasyon yapmaya sağlayan hazır bir ek paket bulunmamaktadır. Tez kapsamında getirilen diğer bir yenilik ise GA, sDE ve BSD gibi sezgisel algoritmaların sisteme entegre edilerek optimizasyon iterasyonları sırasında üretilen sinyal değerlerinin simülasyonda uygulanması için MATLAB ortamında bir sinyal çizelgeleme algoritmasının geliştirilmesidir. Geliştirilen çizelgeleme algoritması iterasyona dayalı tüm algoritmaların VISSIM ortamında kullanılabilmesine imkân sağlamaktadır.

Optimizasyon işlemi için ise GA, sDE ve BSD olmak üzere 3 farklı sezgisel arama algoritması kullanılmış ve sonuçları Vissig algoritması ile kıyaslanmıştır. Bulgulardan da görüleceği üzere kısıtlamaların doğru verilmesi halinde mevcut durumda BSD algoritmasının performansı ile Vissig algoritmasının performansı kıyaslandığında, araçların çıkışa ulaşma performansları arasında yaklaşık %1.1 seviyesinde artış görülse de kendi içlerinde bağıl olarak değerlendirildiklerinde bu oran yaklaşık olarak %1.3 seviyelerindedir. sDE ve GA araçların çıkışa ulaşma performanslarının BSD'ye göre yaklaşık %6.3 ve %7.2 seviyelerinde düşük olduğu buna karşılık bağıl performanslarındaki düşüşün %11.6 ve %13.5 seviyelerinde olduğu görülmektedir. Aşırı doygun durumda ise BSD'nin duruma daha başarılı şekilde adapte olduğu görülmüştür. Buna göre BSD'nin Vissig'a göre yaklaşık %2.8, bağıl olarak ise

yaklaşık %5.7 daha başarılı performans gösterdiği görülmektedir. Buna karşın sDE ve GA'nın performans değerlerinin BSD'ye göre sırasıyla yaklaşık %6.7 ve %8.42, bağlı olarak ise yaklaşık %13.8 ve %17.2 daha düşük performans gösterdiği görülmüştür.

Vissig algoritmasının en büyük dezavantajı girdi parametreleri olan döngü uzunluğu, yeşiller arası bekleme matrisi, fazlar ve faz temelli sinyal programına ait verilerin el ile girilmesi ve tecrübeye dayalı olmasıdır. Genellikle bu değerler deneme yanılma yolu ile elde edilmektedir. Tez kapsamında getirilen en önemli yenilik sezgisel arama algoritmalarında tüm parametrelere doğru bir aralık verilerek doğru optimizasyonun yapılabilmesini göstermektedir. Bunun kanıtı ise BSD algoritmasının trafik sinyal optimizasyonunda kullanılıp uygulanan diğer algoritmalarından başarılı sonuç elde edilmiş olmasıdır.

Sezgisel arama algoritmaların temel eksikliklerinden biri parametre kısıtlarının tecrübeye dayalı olarak el ile verilmesidir. Kısıtlamaların çok geniş tutulması halinde BSD, sDE ve GA algoritmaları genellikle lokal maksimuma yakınsamış diğer bir ifade ile verilen 5000 iterasyonda global maksimuma yakınsama performansı gösterememiştir. Fakat kısıtların farklı şekillerde değiştirilmesi durumunda, algoritma başarısını olumlu yönde etkileyebilme ihtimali bulunmaktadır. Bu ihtimalin bulunması için ise sınırların kaldırılıp sonsuz iterasyon yapılması gerekmektedir. Ayrıca kısıtların çok geniş tutulması veya tamamen kaldırılmasında, algoritmalar sezgisel olarak sürekli en kalabalık linklerin akışına izin verip, diğer linklerdeki yoğunluğu göz ardı etmektedir. Sonuç olarak bu daha fazla aracın çıkışa ulaşmasına sebep olsa da aslında kullanışlı bir model değildir. Bunun yanında çözümün durması için de eşik değer belirtilebilir. Fakat tez kapsamındaki problemde geçen araç sayısı tam sayılı değerlerdir. Bu sayısının tek bir araç dahi artması çözümün iyileşmesi manasına geldiği için çözümün durması için eşik değer bulunmamaktadır. Böylelikle çözüm maksimum iterasyon sayısına kadar sürmektedir.

Sezgisel arama algoritmalarının temeli GA'ya dayansa da seçim, mutasyon ve çaprazlama operatörlerinin prensipleri algoritmaların başarısını doğrudan etkilemektedir. Örneğin sDE'nin başlangıç popülasyonundan seçilen üç kromozomun biri seçim için ayrılırken diğer ikisi mutasyon için kullanılıp arama uzayındaki global sonuca çok hızlı yakınsamayı kolaylaştırmaktadır. BSD yönteminin ise sDE'nin çaprazlama stratejisinde

Bernstein polinomsallarını kullanması global uzaylardaki çözümleri daha da hızlandırmaktadır. Bunun sebebi ise sDE ve GA'da kullanılan çaprazlama parametrelerin üniform dağılırken, sistemin homojenitesinin korunmasıdır. Bernstein polinomsalları ise bu homojeniteyi ortadan kaldırıp arama uzayındaki çeşitliliğin artmasına sebep olmaktadır. Sonuç olarak BSD algoritması incelenen çalışma alanında istatistiksel temelli olan Vissig'ten daha başarılı bir performans göstermiştir.

Trafik sinyal optimizasyonunun özellikle otoban şeklinde yapılamayan şehir içi ana arterlerde önümüzdeki yıllarda daha da artacaktır. Halen adaptif trafik sinyal optimizasyonu çalışmaları pek çok grup tarafından sürdürülmektedir. Fakat bunun başarılabilmesi için gerçek trafiğin araç bazında takibi gerekmektedir. Bu takibin başarılması içinse yeni mobil teknolojiler geliştirilmektedir. Günümüzde trafik akışının tahmin edilmesi için çoğunlukla kişisel cep telefonları verileri kullanılıyor olsa da araçlar üzerinde bulunan CAN bus destekli yapıların daha verimli olacağı düşünülmektedir. Böylelikle oluşması muhtemel trafik sıkışıklığı önceden yüksek doğrulukla tahmin edilip, trafik sinyallerine anlık ve adaptif olarak müdahale kolaylaşacaktır.

Günümüz çalışmalarının bir diğer konusu ise otonom araçlardır. Günümüzde sadece şehir trafiğine kapalı bölgelerde test edilen ve kullanılan otonom araçların uzun vadede günlük hayatımızın vazgeçilmezi olacağı beklenmektedir. Otonom araçlar aktif olarak hayatımıza girdiğinde ise trafik kontrol merkezlerinin olacağı tahmin edilmektedir. Böylelikle otonom araçlar birbiri ile doğrudan haberleşebilirken, trafiğin akış optimizasyonu için trafik kontrol merkezi ile irtibatta olacağı düşünülmektedir. Eğer bu sistem yeterli hızda ve doğrulukta çalışırsa trafik sinyal lambalarına ihtiyaç kalmayıp, araçlar trafikte yüksek hızda ilerleyebileceklerdir. Böylelikle bireysel araç sahibi olma sayısının yıllara göre azalacağı ön görülmektedir. Bunun yerine anlık ihtiyaçlara göre adapte olabilen toplu taşıma araçları ile ilgili çalışmalar yapıldığı görülmektedir.

KAYNAKLAR

- Abu-Lebdeh, G., Benekohal, R.F., 2003. Design and evaluation of dynamic traffic management strategies for congested conditions. **Transp. Res. Part A** **37**, 109–127.
- Abu-Lebdeh, G., Benekohal, R.F., 2000. Genetic algorithms for traffic signal control and queue management of oversaturated two-way arterials. **Transp. Res. Rec. J. Transp. Res. Board** 1727.
- Alizadeh, H., Bealulieu, L.-D., Bouchard, E., Fabre, L., Jeudy, Y., Laviolette, J., Lafebvre-Ropars, G., Morency, C., Strauss, J., Wielinski, G., 2018. Smart Cities And Integrated Mobility A White Paper.
- Azhari, F., Heidarpour, A., Zhao, X.-L., 2018. On the use of Bernstein-Bézier functions for modelling the post-fire stress-strain relationship of ultra-high strength steel (Grade 1200). Elsevier 175, 605–616. <https://doi.org/10.1016/j.engstruct.2018.08.088>
- Bad, A., Fahmy, A., 2004. A proof of convergence for Ant algorithms. **Inf. Sci.** **160**, 267–279.
- Bando, M., Hasebe, K., Nakanishi, A., Shibata, A., Sugiyama, Y., 1995. Phenomenological Study of Dynamical Model of Traffic Flow. **J. Phys. I** 1389–1399.
- Başkan, Ö., 2009. Karınca Kolonisi Optimizasyonu ile Ulaşım Ağ Tasarımı (Doktora). Pamukkale Üniversitesi, Denizli.
- Ben-Akiva, M., 1996. Development of a deployable real-time dynamic traffic assignment system, Task D Interim report: analytical developments for DTA system. MIT ITS Program, Cambridge (MA).
- Brest, J., Grenier, S., Bošković, B., Mernik, M., Zumer, V., 2006. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. **IEEE Trans. Evol. Comput.** **10**, 646–657.
- Buisson, C., Ladier, C., 2009. Exploring the impact of homogeneity of traffic measurements on the existence of macroscopic fundamental diagrams. **Transp. Res. Rec. J. Transp. Res. Board** **2124**, 127–136.

- Burghout, W., 2004. Hybrid microscopic-mesoscopic traffic simulation (Doktora). Royal Institute of Technology, Stockholm, İsveç.
- Cai, C., Wong, C.K., Heydecker, B.G., 2009. Adaptive traffic signal control using approximate dynamic programming. **Transp. Res. Part C** **17**, 456–474.
- Cassidy, M.J., Jang, K., Daganzo, C.F., 2011. Macroscopic Fundamental Diagrams for Freeway Networks: Theory and Observation. **Transp. Res. Rec. J. Transp. Res. Board** **2260**, 8–15.
- Chang, T., Lin, J., 2000. Optimal signal timing for an oversaturated intersection. **Transp. Res. Part B** **34**, 471–491.
- Chang, T., Sun, J., 2004. Modeling and optimization of an oversaturated signalized network. **Transp. Res. Part B** **38**, 687–707.
- Chen, J., Zheng, J., Wu, P., Zhang, L., Wu, Q., 2017. Dynamic particle swarm optimizer with escaping prey for solving constrained non-convex and piecewise optimization problems. **Expert Syst. Appl.** **86**, 208–223.
- Çivicioğlu, P., 2013a. Backtracking Search Optimization Algorithm for numerical optimization problems. Elsevier 219, 8121–8144. <https://doi.org/10.1016/j.amc.2013.02.017>
- Çivicioğlu, P., 2013b. Artificial cooperative search algorithm for numerical optimization problems. Elsevier 229, 58–76. <https://doi.org/10.1016/j.ins.2012.11.013>
- Çivicioğlu, P., 2012. Transforming geocentric cartesian coordinates to geodetic coordinates by using differential search algorithm. Elsevier 46, 229–247. <https://doi.org/10.1016/j.cageo.2011.12.011>
- Çivicioğlu, P., Beşdok, E., 2019. Bernstein-search differential evolution algorithm for numerical function optimization. **Expert Syst. Appl.** 138.
- Çivicioğlu, P., Beşdok, E., 2013. A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. **Artif. Intell. Rev.** **39**, 315–346.
- Çivicioğlu, P., Beşdok, E., Günen, M.A., Atasever, U.H., 2018. Weighted differential evolution algorithm for numerical function optimization: a comparative study

with cuckoo search, artificial bee colony, adaptive differential evolution, and backtracking search optimization algorithms. *Neural Comput. Appl.*

Clerc, M., Kennedy, J., 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. **IEEE Trans. Evol. Comput.** **6**, 58–73.

Cremer, M., Ludwig, J., 1986. A Fast Simulation Model for Traffic Flow on the Basis of Boolean Operations. **Math. Comput. Simul.** **28**, 297–303.

Daganzo, C.F., Gayah, V., Gonzales, E., 2011. Macroscopic relations of urban traffic variables: bifurcations, multivaluedness and instability. **Transp. Res. Part B** **41**, 278–288.

Das, S., Mullick, S., Suganthan, P.N., 2016. Recent advances in differential evolution – An updated survey. **Swarm Evol. Comput.** **27**, 1–30.

Davarynejad, M., Hegyi, A., Vrancken, J., Van den Berg, J., 2011. Motorway ramp-metering control with queuing consideration using Q-learning. Presented at the **IEEE Conference Proceedings on Intelligent Transportation Systems**, pp. 1652–1658.

Duret, A., Ahn, S., Buisson, C., 2012. Lane flow distribution on a three-lane freeway: general features and the effects of traffic controls. **Transp. Res. Part C** **24**, 157–167.

Edie, L.C., 1961. Car Following and Steady-State Theory for Non-Congested Traffic. **Oper. Res.** **9**, 66–76.

Fares, A., Gomaa, W., 2015. Multi-agent reinforcement learning control for ramp metering. Presented at the Proceedings of the 23rd International Conference on Systems Engineering, pp. 1590–1595.

Fares, A., Gomaa, W., 2014. Freeway ramp-metering control based on reinforcement learning. Presented at the 11th **IEEE International Conference on Control & Automation**, pp. 167–173.

Gardner, N.H., 1983. OPAC: a demand-responsive strategy for traffic signal control. **Transp. Res. Rec. J. Transp. Res. Board** **906**, 75–81.

Gawron, C., 1998. Simulation-Based Traffic Assignment. Universität zu Köln, Köln/Germany.

- Gazis, D.C., Herman, R., Rothery, R.W., 1961. Nonlinear Follow-the-Leader Models of **Traffic Flow. Oper. Res. 9**, 545–567.
- Gazis, D.C., Herman, R., Weiss, G.H., 1962. Density oscillations between lanes of a multilane highway. **Oper. Res. 10**, 658–667.
- Geroliminis, N., Daganzo, C.F., 2008. Existence of urban-scale macroscopic fundamental diagrams: some experimental findings. **Transp. Res. Part B 43**, 759–770.
- Geroliminis, N., Sun, J., 2011. Hysteresis Phenomena of a Macroscopic Fundamental Diagram in Freeway Networks, in: Proceedings of the 10th International Symposium on Transportation and Traffic Theory. Elsevier Science Publishing, pp. 213–228.
- Gu, H., Wang, W., 1998. A global optimization simulated annealing algorithm for intersection signal timing. **J. Southeast Univ. 28**, 68–72.
- Haijema, R., Hendrix, E.M.T., 2014. Traffic responsive control of intersections with predicted arrival times: a Markovian approach. **Comput. Aided Civ. Infrastructure Eng. 29**, 123–139.
- HandWiki, 2020. Bernstein polynomial. handwiki.org.
- He, J., Hou, Z., 2012. Ant colony algorithm for traffic signal timing optimization. **Adv. Eng. Softw. 43**, 14–18.
- Helbing, D., 1997. Modeling multi-lane traffic flow with queuing effects. **Phys. A 242**, 175–194.
- Helbing, D., 1996. Derivation and empirical validation of a refined traffic flow model. **Phys. A 233**, 253–282.
- Helbing, D., 1995. Improved fluid-dynamic model for vehicular traffic. **Phys. Rev. E 51**, 3164–3169.
- Helbing, D., Treiber, M., 1998. Gas-Kinetic-Based Traffic Model Explaining Observed Hysteretic Phase Transition. **Phys. Rev. Lett. 81**, 3042–3045.

- Henry, J.J., Farges, J.L., Tuffal, J., 1983. The Prodyn real time traffic algorithm. Presented at the Proceedings of the Fourth IFAC-IFIP-IFORS Conference on Control in Transportation Systems, Baden-Baden, pp. 307–311.
- Henry, R.D., 2005. Signal Timing on A Shoestring. Continuing Education and Development, Inc., Sabra, Wang & Associates, Inc. 1504 Joh Avenue, Suite 160 Baltimore, MD 21227.
- Herman, R., Potts, R., 1959. Single Lane Traffic Theory and Experiment. Presented at the Proceedings of the Symposium on Theory of Traffic Flow, Elsevier Publications Co., pp. 120–146.
- Herman, R., Prigogine, I., 1979. A two-fluid approach to town traffic. **Science** **204**, 148–151.
- Hoogendoorn, S., Bovy, P., 1999. Gas-kinetic model for multilane heterogeneous traffic flow. **Transp. Res. Rec. J. Transp. Res. Board** **1678**, 150–159.
- Hu, H., Liu, H.X., 2013. Arterial offset optimization using archived high-resolution traffic signal data. **Transp. Res. Part C** **37**, 131–144.
- Hu, H., Wu, X., Liu, H.X., 2013. Managing oversaturated signalized arterials: a maximum flow based approach. **Transp. Res. Part C** **36**, 196–211.
- Hunt, P.B., Robertson, D.I., Bretherton, R.D., 1982. The SCOOT on-line traffic signal optimisation technique. **Traffic Eng. Control** **25**, 190–192.
- Janz, S., 1998. Mikroskopische Minimalmodelle des Straßenverkehrs (Doktora). University of Cologne, Almany.
- Jayakrishnan, R., Mahmassani, H.S., Hu, T.Y., 1994. An Evaluation Tool fo Advanced Traffic Information and Management Systems in Urban Networks. **Transp. Res. C** **2C**, 129–147.
- Jin, W.L., 2013. A Multi-commodity Lighthill-Whitham-Richards Model of Lane-changing Traffic Flow. **Procedia - Soc. Behav. Sci.** **80**, 658–677. <https://doi.org/10.1016/j.sbspro.2013.05.035>
- Karaboğa, D., 2004. Yapay Zeka Optimizasyonu Algoritmaları. Atlas Yayın Dağıtım, İstanbul.

- Karaboğa, D., Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. **J. Glob. Optim.** **39**, 459–471.
- Kerner, B.S., Konhäuser, P., 1994. Structure and parameters of clusters in traffic flow. **Phys. Rev. E** **50**, 54–83.
- Kerner, B.S., Rehborn, H., 1997. Experimental Properties of Phase Transitions in Traffic Flow. **Phys. Rev. Lett.** **79**, 4030–4033.
- Kerner, B.S., Rehborn, H., 1996a. Experimental properties of complexity in traffic flow. **Phys. Rev. E** **53**, R4275–R4278.
- Kerner, B.S., Rehborn, H., 1996b. Experimental Features and Characteristics of Traffic Jams. **Phys. Rev. E** **53**, 1297–1300.
- Knoop, V., Duret, A., Buisson, C., van Arem, B., 2010. Lane distribution of traffic near merging zones influence of variable speed limits. Presented at the 13th International IEEE Conference on Intelligent Transportation Systems, pp. 485–490.
- Knoop, V.L., Hoogendoorn, S., Shiomi, Y., Buisson, C., 2012. Quantifying the number of lane changes in traffic. **Transp. Res. Rec. J. Transp. Res. Board** **2278**, 31–41.
- Krauß, S., 1998. Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics (Doktora). University of Cologne, Almany.
- Krauß, S., 1997. Towards a Unified View of Microscopic Traffic Flow Theories. Presented at the Proceedings of the 8th IFAC/IFIP/IFORS Symposium, Elsevier Science, Yunanistan.
- Kühne, R.D., 1984. Macroscopic Freeway Model for Dense Traffic-Stop-Start Waves and Incident Detection. Presented at the Proceedings of the Ninth International Symposium on Transportation and Traffic Theory, Hollanda.
- Kuyer, L., Whiteson, S., Bakker, B., Vlassis, N., 2008. Multiagent reinforcement learning for urban traffic control using coordination graphs, in: Machine Learning and Knowledge Discovery in Databases. Springer, Berlin Heidelberg, pp. 656–671.

- Kwon, E., Stephanedes, Y.J., 1994. Comparative evaluation of adaptive and neural-network exit demand prediction for freeway control. **Transp. Res. Rec. J. Transp. Res. Board** 1446.
- Laval, J.A., Daganzo, C.F., 2006. Lane-changing in traffic streams. **Transp. Res. Part B** 40, 251–264.
- Lee, H.Y., Lee, H.W., Kim, D., 1998. Origin of Synchronized Traffic Flow on Highways and Its Dynamic Phase Transitions. **Phys. Rev. Lett.** 81, 1130–1133.
- Leonard, D.R., Power, P., Taylor, N.B., 1989. Contram: structure of the model. *Transp. Res. Lab.*
- Li, H., Prevedouros, P.D., 2004. Traffic adaptive control for oversaturated isolated intersections: model development and simulation testing. **J. Transp. Eng.** 130, 594–601.
- Li, R., 2009. Urban Road Traffic Management. Beijing People's Traffic Press.
- Li, R., Jin, P.J., 2017. Transit Signal Priority Optimization for Urban Traffic Network Considering Arterial Coordinated Signal Control. **Adv. Mech. Eng.** 9, 1–12.
- Lieberman, E., Chang, J., 2005. Optimizing traffic signal timing through network decomposition. **Transp. Res. Rec. J. Transp. Res. Board** 1925, 167–175.
- Lieberman, E., Chang, J., Prasas, E., 2000. Formulation of real-time control policy for oversaturated arterials. **Transp. Res. Rec. J. Transp. Res. Board** 1727, 77–88.
- Lighthill, M.J., Whitham, G.B., 1955. On kinematic waves: II. A theory of traffic flow on long crowded roads. **Proceedings of the Royal Society of London. Ser. Math. Phys. Sci.** 229, 317–345.
- Lin, S., Schutter, B.D., Xi, Y., Hellendoorn, H., 2012. Efficient network-wide model-based predictive control for urban traffic networks. **Transp. Res. Part C** 24, 122–140.
- Liu, H., Balke, K.N., Lin, W., 2008. A reverse causal-effect modeling approach for signal control of an oversaturated intersection. **Transp. Res. Part C** 16, 742–754.

- Liu, J., Zhang, H., He, K., Jiang, S., 2018. Multi-objective particle swarm optimization algorithm based on objective space division for the unequal-area facility layout problem. **Expert Syst. Appl.** **102**, 179–192.
- Liu, Y., Chang, G.-L., 2011. An arterial signal optimization model for intersections experiencing queue spillback and lane blockage. **Transp. Res. Part C** **19**, 130–144.
- Lo, H.K., Chow, A.H.F., 2004. Control strategies for oversaturated traffic. **J. Transp. Eng.** **130**, 466–478.
- Ludmann, J., Neunzig, D., Weilkes, M., 1996. Traffic Simulation with Consideration of Driver Models- Theory and Examples. Presented at the roceedings of the Euromotor- Seminar on Telematics, Vehicle and Environment, Almany.
- Luk, J.Y.K., 1984. Two traffic-responsive area traffic control methods: SCATS and SCOOT. *Traffic Eng. Control* **25**, 14–22.
- Lynn, N., Suganthan, P.N., 2017. Ensemble particle swarm optimizer. **Appl. Soft Comput.** **55**, 533–548.
- Mahmassani, H.S., Peeta, S., 1993. Network Performance under System Optimal and User Equilibrium Dynamic Assignments: Implications for ATIS. **Transp. Res. Rec. J. Transp. Res. Board** **1408**, 83–93.
- Mahmassani, H.S., Williams, J.C., Herman, R., 1987. Performance of urban traffic networks. Presented at the Proceedings of the 10th International Symposium on Transportation and Traffic Theory, Elsevier Science Publishing, pp. 1–20.
- Mahmassani, H.S., Williams, J.C., Herman, R., 1984. Investigation of network-level traffic flow relationships: some simulation results. **Transp. Res. Rec. J. Transp. Res. Board** **971**, 121–130.
- Marca, J., Rindt, C., Jayakrishnan, R., 2006. Method for creating a real-time distributed travel history database: persistent traffic cookies project. **Transp. Res. Rec. J. Transp. Res. Board** 69–77.
- Mauro, V., DiTaranto, D., 1990. UTOPIA. Presented at the Proceedings of the 6th IFAC-IFIP-IFORS Symposium on Control Computers and Communication in Transportation, pp. 245–252.

- Michalopoulos, P.G., Beskos, D.E., Yamauchi, Y., 1984. Multilane traffic flow dynamics: some macroscopic considerations. **Transp. Res. Part B** **18**, 377–395.
- Mlakar, U., Potočník, B., Brest, J., 2016. A hybrid differential evolution for optimal multilevel image thresholding. **Expert Syst. Appl.** **64**, 221–232.
- Nagel, K., 1995. High-speed microsimulations of traffic flow (Doktora). Universität zu Köln, Almanya.
- Nagel, K., Paczuski, M., 1995. Emergent Traffic Jams. *Phys. Rev. E* **51**.
- Nagel, K., Schreckenberg, M., 1992. A cellular automaton model for freeway traffic. **J. Phys. Fr.** **2**, 2221–2229.
- Opara, K., Arabas, J., 2018. Comparison of mutation strategies in Differential Evolution – A probabilistic perspective. **Swarm Evol. Comput.** **39**, 53–69.
- Ozsoydan, F.B., Baykasoğlu, A., 2019. Quantum firefly swarms for multimodal dynamic optimization problems. **Expert Syst. Appl.** **115**, 189–199.
- Park, B., Messer, C., Urbanik, T., 2000. Enhanced genetic algorithm for signal-timing optimization of oversaturated intersections. **Transp. Res. Rec. J. Transp. Res. Board** **1727**, 32–41.
- Park, B., Messer, C., Urbanik, T., 1999. Traffic signal optimization program for oversaturated conditions: a genetic algorithms approach. Presented at the Proceedings of the 78th Annual TRB Meeting, Washinton, Amerika.
- Park, J.-Y., 2009. Network-wide Signal Control with Distributed Real-time Travel Data (Doktora). University of California, Irvine, Amerika.
- Ponzlet, M., Wagner, P., 1996. Validation of a CA-Model for Traffic Simulation of the Northrhine-Westphalia Motorway Network. Presented at the 24th European Transport Forum, Proceedings of Seminar D&E, p. 1.
- Price, K.V., Storn, R.M., Lampien, J.A., 2005. Differential Evolution- A Practical Approach to Global Optimization. Springer.
- Prigogine, I., 1961. A Boltzmann-like Approach to the Statistical Theory of Traffic Flow. *Theory Traffic Flow*.

- Prigogine, I., Herman, R., 1971. Kinetic Theory of Vehicular Traffic. Am. Elsevier Publ. Co.
- PTV AG, 2018. PTV VISSIM 11 User Manual. PTV Group, Karlsruhe, Almanya.
- Qin, F., Li, L., Gao, S., Yang, X., Xiang, C., 2014. A deep learning approach to the classification of 3D CAD models. **J. Zhejiang Univ. Sci. C** **15**, 91–106.
- Rapoport, A., Kugler, T., Dugar, S., Gisches, E.J., 2009. Choice of routes in congested traffic networks: Experimental tests of the Braess Paradox. **Games Econ. Behav.** **65**, 538–571.
- Rezaae, K., Abdulhai, H., Abdelgawad, H., 2012. Application of reinforcement learning with continuous state space to ramp metering in real-world conditions. Presented at the 15th International IEEE Conference on Intelligent Transportation Systems, pp. 1590–1595.
- Richards, P.I., 1956. Shock waves on the highway. **Oper. Res.** **4**, 43–51.
- Robertson, D.I., Bretherton, R.D., 1974. Optimum control of an intersection for any known sequence of vehicular arrivals. Presented at the Proceedings of the Second IFAC- IFIP-IFORS Symposium on Traffic Control and Transportation system, Monte Carlo.
- Robetson, D.I., 1969. TRANSYT: A Traffic Network Study Tool (No. LR253). Transport and Road Research Laboratory, İngiltere.
- Saberi, M., 2013. Network Traffic Science: Theory, Characteristics, and Dynamics (Doktora). Northwestern University, Evanston, Illinois.
- Saberi, M., Mahmassani, H.S., 2012. Exploring the properties of network-wide flow-density relations in freeway networks. **Transp. Res. Rec. J. Transp. Res. Board** **2315**, 153–163.
- Schadschneider, A., Schreckenberg, M., 1993. Cellular Automaton Models and Traffic Flow. *J. Phys. A* **26**.
- Schreckenberg, M., Wolf, D.E., 1998. Traffic and Granular Flow '97, in: Springer. Singapur.

- Schütt, H., 1991. Entwicklung eines sehr schnellen, bitorientierten Verkehrssimulationssystems für Straßennetze (Doktora). Schriftenreihe der ag automatisierungstechnik, TU Hamburg, Almany.
- Skružný, F., 2017. Control System Development of Traffic Signal Control in MATLAB for PTV VISSIM (Yüksek Lisans). Czech Technical University of Prague Faculty of Transportation Sciences, Çek Cumhuriyeti.
- Spall, J., Chin, D., 1994. A model-free approach to optimal signal light timing for system-wide traffic control. Presented at the Proceedings of the 33rd Conference on Decision and Control, pp. 1868–1875.
- Stevanovic, A., Stevanovic, J., So, J., Ostojic, M., 2015. Optimization of traffic signal timings based on surrogate measures of safety. **Transp. Res. Part C** **55**, 46–68.
- Sun, W., Wang, Y., Yu, G., Liu, H.X., 2015. Quasi-optimal feedback control for a system of oversaturated intersections. **Transp. Res. Part C** 57.
- Teklu, F., Sumalee, A., Watling, D., 2007. A genetic algorithm approach for optimizing traffic control signals considering routing. **Comput. Aided Civ. Infrastructure Eng.** **29**, 123–139.
- Tong, Y., Zhao, L., Li, L., Zhang, Y., 2015. Stochastic programming model for oversaturated intersection signal timing. **Transp. Res. Part C** **58**, 474–486.
- Treiber, M., Kesting, 2013 *Traffic Flow Dynamics*.
- Treiber, M., Kesting, A., 2013. *Traffic Flow Dynamics, Data, Models and Simulation*. Springer, Dresden/Germany.
- Wardrop, J.G., 1952. Some Theoretical Aspects of Road Traffic Research, in: *Proceedings of the Institute of Civil Engineers*. pp. 325–378.
- Wei, W., Zhang, Y., 2002. FL-FN based traffic signal control. Presented at the Proceedings of the 2002 IEEE International Conference on Fuzzy Systems, pp. 296–300.
- Wey, W.-M., Jayakrishnan, R., 1997. Network traffic signal optimization formulation with embedded platoon dispersion simulation. *Transp. Res. Rec. J. Transp. Res. Board* 150–159.

- Whitley, D., 1994. A genetic algorithm tutorial. *Stat. Comput.* 65–85.
- Wiedemann, R., 1974. Simulation des Straßenverkehrsflusses (Doktora). Universität Karlsruhe, Schriftenreihe des Instituts für Verkehrswesen Heft 8.
- Williams, J.C., Mahmassani, H.S., Herman, R., 1995. Sampling strategies for two-fluid model parameter estimation in urban networks. **Transp. Res. Part A** **29**, 229–244.
- Williams, J.C., Mahmassani, H.S., Herman, R., 1987. Urban traffic network flow models. **Transp. Res. Rec. J. Transp. Res. Board** **1112**, 78–88.
- Yagar, S., Han, B., 1994. A procedure for real-time signal control that considers transit interference and priority. **Transp. Res. Part B** **28**, 315–331.
- Yan, Y., Li, W., 2006. Ant colony optimization for signalized intersection. *J Highway Transport Res Develop.* **J. Highw. Transp. Res. Dev.** **23**, 116–119.
- Yang, J., Yang, D., 2001. Optimized signal time model in signaled intersection. *J. Tongji Univ.* **29**, 789–794.
- Yang, X.S., Deb, S., 2009. Cuckoo search via levy flights. Presented at the World congress on nature and biologically inspired computing, Hindistan, pp. 210–214.
- Zhang, J., Sanderson, A.C., 2009. JADE: Adaptive differential evolution with optional external archive. **IEEE Trans. Evol. Comput.** **13**, 945–958.
- Zhang, Q., Zou, D., Duan, N., Shen, X., 2019. An adaptive differential evolutionary algorithm incorporating multiple mutation strategies for the economic load dispatch problem. **Appl. Soft Comput.** **78**, 641–669.
- Zhang, W., Zhu, G., 2011. Comparison and application of four versions of particle swarm optimization algorithms in the sequence optimization. **Expert Syst. Appl.** **38**, 8858–8864.
- Zheng, X., Recker, W., 2013. An adaptive control algorithm for traffic-actuated signals. **Transp. Res. Part C** **30**, 93–115.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı:	Abdullah KARAAĞAÇ
Uyruğu:	Türkiye (T.C)
Doğum Tarihi ve Yeri:	Mayıs 1990 - İstanbul
Medeni Durum:	Evli
e-mail:	akaraagac@erciyes.edu.tr
Yazışma Adresi:	Erciyes Üniversitesi Mühendislik Fakültesi Harita Mühendisliği Bölümü Talas/Kayseri

EĞİTİM

Derece	Kurum	Mezuniyet Tarihi
Yüksek Lisans	Erciyes Ü. Harita Müh. ABD, Kayseri	2015
Lisans	Erciyes Ü. Jeodezi ve Fotogrametri Müh., Kayseri	2013
Lise	Derya Öncü Koleji, İstanbul	2008

İŞ DENEYİMLERİ

Yıl	Kurum	Görev
2015-Halen	Usturlab Mühendislik, Kayseri	Genel Müdür
2013-Halen	ERÜ Harita Müh. Bölümü, Kayseri	Arş. Gör.
2013-2013	Talas Anadolu Meslek Lisesi, Kayseri	Ücretli Öğretmen
2013-2013	TR COM Bilişim, Ankara	Yazılım Geliştirici
2013-2013	RAN Mühendislik ve Güvenlik, İstanbul	Kayseri Bölge Satış Müdürü
2010-2013	RCE Elektronik, İstanbul	Stajyer/Mühendis
2010-2010	Erciyes Harita, Kayseri	Stajyer

YABANCI DİL

İngilizce, Arapça, Japonca

YAYINLAR

1. Bostancı B., Karaağaç A., Investigating the Shortest Survey Route in a GNSS Traverse Network, TEHNICKI VJESNIK-TECHNICAL GAZETTE, cilt.26, no.2, ss.355-362, 2019 (SCI İndekslerine Giren Dergi)
2. Taşkanat T., Karaağaç A., Beşdok E., Bostancı B., Kentsel Sorunların Yönetimi için Bir Gönüllü Coğrafi Bilgi Mobil Uygulaması Geliştirilmesi, Geomatik, cilt.3, ss.84-91, 2018 (Diğer Kurumların Hakemli Dergileri)
3. BOSTANCI B., ZEYDAN M., Çete M., DEMİR H., KARAAĞAÇ A., Decision making for site selection using fuzzy modeling, Journal of Urban Planning and Development, cilt.143, 2017 (SCI İndekslerine Giren Dergi)

HAKEMLİ KONGRE / SEMPOZYUM BİLDİRİ KİTAPLARINDA YER ALAN YAYINLAR

1. Karaağaç A., Bostancı B., Deep Reinforcement Learning Approach with Hybrid Policy for Traffic Signal Timing Optimization, 26th Intelligent Transportation Systems World Congress (ITSWC'19), Singapore, Singapur, 21- 25 October 2019, ss.1-5
2. Kılıç S., Karaağaç A., Konum Servisleri Kullanılarak Mobil Haber Uygulaması Geliştirilmesi: NEWSONMAP, 10. TÜRKİYE ULUSAL FOTOGRAMETRİ VE UZAKTAN ALGILAMA BİRLİĞİ TEKNİK SEMPOZYUMU, Aksaray, Türkiye, 25-27 April 2019, ss.287-290
3. Aktaş Ş., Yeşilyurt K., Korkmaz H. N., Yaşar İ., Akduman P., Akalın S., Karaağaç A., Konuma Bağlı Hatırlatıcı Mobil Uygulamanın Geliştirilmesi: WHISPERER, 10. TÜRKİYE ULUSAL FOTOGRAMETRİ VE UZAKTAN ALGILAMA BİRLİĞİ TEKNİK SEMPOZYUMU, Aksaray, Türkiye, 25 - 27 April 2019, ss.279-284
4. Altuntaş B., Emek Ş., Yılmaz M., Karaağaç A., Konum Tabanlı Mobil Uygulama Örneği: ASTRA TRACKER, 10. TÜRKİYE ULUSAL FOTOGRAMETRİ VE UZAKTAN ALGILAMA BİRLİĞİ TEKNİK SEMPOZYUMU, Aksaray, Türkiye, 25-27 April 2019, ss.285-286
5. KARAAĞAÇ A., BOSTANCI B., The Impact of Astrolabes in the Development of Contemporary Mapping, II. International Scientific and Vocational Studies Congress, Nevşehir, Türkiye, 5-08 July 2018, ss.1
6. KARAAĞAÇ A., BOSTANCI B., Mobile Application Development for Traversing, II. International Scientific and Vocational Studies Congress, Nevşehir, Türkiye, 5-08 July 2018, ss.1
7. KARAAĞAÇ Z., KARAAĞAÇ A., Counting the Nanoparticles of Gold(Au), Imaged via SEM Device, 2nd International Vocational Science Symposium, Antalya, Türkiye, 26-29 April 2018, ss.1
8. KARAAĞAÇ A., KARAAĞAÇ Z., BOSTANCI B. Comparison The Positioning Methods of Android and iOS Devices 2nd International Vocational Science Symposium, Antalya, Türkiye, 26 - 29 April 2018, ss.1-2
9. TAŞKANAT T., KARAAĞAÇ A., BEŞDOK E., BOSTANCI B., Kentsel Sorunların Yönetimi İçin Bir Gönüllü Coğrafi Bilgi Mobil Uygulaması Geliştirilmesi, TMMOB ULUSLARARASI COĞRAFİ BİLGİ SİSTEMLERİ KONGRESİ 2017, Adana, Türkiye, 15 - 18 November 2017, ss.1
10. TAŞKANAT T., KARAAĞAÇ A., BEŞDOK E., BOSTANCI B., Development of a Volunteered Geographical Information Mobile Application for the Management of Urban Affairs, UCTEA International Geographical Information Systems Congress 2017, Adana, Türkiye, 15 - 18 November 2017, ss.62-66
11. KARAAĞAÇ A., TAŞKANAT T., BOSTANCI B., Determination of Optimum Route with Time Weighted Dijkstra Algorithm, International Advance Researches in Engineering Congress / IAREC'17, Osmaniye, Türkiye, 16 - 18 November 2017, cilt.1, no.1, ss.1

12. KARAAĞAÇ A. , BOSTANCI B., Development of an iOS application: Altitude Estimator in a Local Area, The 8th International Conference on Information Technology ICIT 2017 Internet of Things, Amman, Ürdün, 17 - 18 May 2017, ss.1-5
13. KARAAĞAÇ A. , BOSTANCI B., Konum Servislerini Kullanan Mobil Uygulamaların Geliştirmesi: Taksi GO Uygulaması, 6. Uzaktan Algılama ve CBS Sempozyumu, Adana, Türkiye, 5-07 October 2016, ss.1-5
14. KARAAĞAÇ A. , BOSTANCI B., PROCESSABLE DATA MAKING IN THE REMOTE SERVER SENT BY ANDROID PHONE AS A GIS DATA COLLECTİNG TOOLEUROPEAN GEOSCIENCES UNION GENERAL ASSEMBLY 2016, Viyana, Avusturya, 17-22 April 2016, no.14947, ss.1
15. BOSTANCI B. , DEMİR H., KARAAĞAÇ A., DETERMINATION OF NOMINAL VALUE WITH FUZZY ANALYTICAL HIERARCHY PROCESS WEIGHTS, The World Cadastre Summit, Congress& Exhibition, İstanbul, Türkiye, 20-24 April 2015, ss.1-12

KİTAP BÖLÜMÜ

1. BOSTANCI B., Demir H., KARAAĞAÇ A., Determination of Nominal Value by Fuzzy Analitic Hierarchic Process Weights , The World Cadastre, Yomralıoğlu T., Editör, Tapu Kadastro Döner Sermaye İşletmesi Müdürlüğü, Ankara, ss.1-10, 2016