T.C.
KAYSERİ ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ
ANABİLİM DALI

META-SEZGİSEL ALGORİTMALAR İLE TRAFİK SİNYAL OPTİMİZASYONU

Hazırlayan
Olcay ADIYAMAN

Danışman
Doç. Dr. Ali DURMUŞ

Yüksek Lisans Tezi

Temmuz 2023, KAYSERİ

T.C.
KAYSERİ ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ
ANABİLİM DALI

META-SEZGİSEL ALGORİTMALAR İLE TRAFİK SİNYAL OPTİMİZASYONU

(Yüksek Lisans Tezi)

Hazırlayan
Olcay ADIYAMAN

Danışman
Doç. Dr. Ali DURMUŞ

Temmuz 2023, KAYSERİ

BİLİMSEL ETİĞE UYGUNLUK

Bu tezin kendi çalışmam olduğunu, planlanmasından yazımına kadar hiçbir aşamasında etik dışı davranışımın olmadığını, tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları kaynaklar listesine aldığımı, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

Olcay ADIYAMAN

İmza

TEZ İNTİHAL RAPORU TESLİM FORMU

T.C.

KAYSERİ ÜNİVERSİTESİ

Lisansüstü Eğitim Enstitüsü Müdürlüğü

Anabilim Dalı : Elektrik Elektronik Mühendisliği ABD

Program Adı : Tezli Yüksek Lisans

Tez Başlığı : Meta-Sezgisel Algoritmalar ile Trafik Sinyal Optimizasyonu

Yukarıda bilgileri verilen tez çalışmasının a) Giriş, b) Ana bölümler ve c) Sonuç kısımlarından oluşan (Kapak, Önsöz, Özet, İçindekiler ve Kaynakça hariç) toplam 56 sayfalık kısmına ilişkin 26/07/2023 tarihinde ***Turnitin*** intihal programından aşağıda belirtilen filtreleme uygulanarak alınmış olan özgünlük raporuna göre tezin benzerlik oranı: % 8' dur.

Uygulanan filtrelemeler:

1- Giriş dahil
2- Ana Bölümler dahil
3- Sonuç dahil
4- Alıntılar dahil
5- Kapak hariç
6- Önsöz ve Teşekkür hariç
7- İçindekiler hariç
8- Kaynakça hariç
9- Özet hariç
10- Yedi (7) kelimeden daha az örtüşme içeren metin kısımları hariç

Kayseri Üniversitesi Lisansüstü Eğitim Enstitüsü Tez Önerisi Hazırlama, Tez Yazma ve Teslim Yönergesini inceledim ve bu yönergede belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini, aksinin tespit edileceği muhtemel durumlarda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini bilgilerinize arz ederim.

Danışman: Doç. Dr. Ali DURMUŞ

Öğrenci: Olcay ADIYAMAN

YÖNERGEYE UYGUNLUK

“Meta-Sezgisel Algoritmalar ile Trafik Sinyal Optimizasyonu**”** adlı Yüksek Lisans Tezi, Kayseri Üniversitesi Lisansüstü Tez Önerisi ve Tez Yazma Yönergesi’ ne uygun olarak hazırlanmıştır.

Hazırlayan
Olcay ADIYAMAN
İmza

Danışman
Doç. Dr. Ali DURMUŞ
İmza

Elektrik Elektronik Mühendisliği ABD Başkanı

Doç. Dr. Engin Cemal MENGÜÇ
İmza

KABUL VE ONAY

Doç. Dr. Ali DURMUŞ danışmanlığında **Olcay ADIYAMAN** tarafından hazırlanan **"Meta-Sezgisel Algoritmalar ile Trafik Sinyal Optimizasyonu"** adlı bu çalışma jürimiz tarafından Kayseri Üniversitesi Lisansüstü Eğitim Enstitüsü **Elektrik-Elektronik Mühendisliği** Anabilim Dalı'nda **Yüksek Lisans Tezi** olarak kabul edilmiştir.

… /…/2023

JÜRİ:

Danışman : Doç. Dr. Ali DURMUŞ

Üye : Prof. Dr. Ercan KARAKÖSE

Üye : Dr. Öğr. Üyesi Rıfat KURBAN

ONAY:

Bu tezin kabulü Enstitü Yönetim Kurulunun ……...… tarih ve ……..……sayılı kararı ile onaylanmıştır.

……. /……../ ……

Doç. Dr. Ahmet TERZİ
Enstitü Müdürü

TEŞEKKÜR

Çalışmam süresince bilgisi, tecrübesi, uzmanlığı ve bilimsel katkılarıyla yardımlarını esirgemeyen ve bu çalışmanın bilimsel bir tabanda nihayete ermesinde en büyük katkı sahibi değerli danışman hocam **Sayın Doç. Dr. Ali DURMUŞ**' a sonsuz teşekkürlerimi sunarım.

Araştırmalarım ve çalışmalarım sırasında vermiş olduğu desteklerden dolayı aileme, eşim ve çalıştığım kurum olan Kayseri Ulaşım A.Ş deki tüm yöneticilerime teşekkür ederim.

Olcay ADIYAMAN

Kayseri, Temmuz 2023

META-SEZGİSEL ALGORİTMALAR İLE TRAFİK SİNYAL OPTİMİZASYONU

Olcay ADIYAMAN

Kayseri Üniversitesi, Lisansüstü Eğitim Enstitüsü

Yüksek Lisans Tezi, Temmuz 2023

Danışman: Doç. Dr. Ali DURMUŞ

ÖZET

Küresel çapta kentleşmenin artmasıyla birlikte hayatımızı olumsuz etkileyen en önemli unsurlardan birisi de trafiktir. Artan insan ve araç sayısına bağlı olarak gerçekleşen trafik insan hayatını ciddi şekilde olumsuz yönde etkilemektedir. Yakıt maliyetleri, zaman, stres, karbon emisyonunun artması gibi nedenler hem ekonomik hem de iklim değişikliği anlamında ciddi etkileri bulunmaktadır. Günümüzün bu probleminin çözümü hususunda birçok çalışma yapılmaktadır. Yolların genişletilmesi, kavşaklardaki fiziki değişiklikler ve kavşak sinyalizasyonunda yapılacak olan çalışmalarla sürekli trafikteki bekleme sürelerinin azaltılması amaçlanmaktadır. Kavşak sinyalizasyon sisteminin performansını gösteren güvenlik sonrasındaki ilk unsur gecikme olarak karşımıza çıkmaktadır. Bu çalışmada meta-sezgisel optimizasyon yöntemleri kullanılarak kavşaklardaki bekleme süresinin azaltılmasına yönelik bir çalışma yapılmıştır.

Yapılan çalışmada Amerikan modeli kullanılarak gecikme süresi optimizasyonu MATLAB yazılımı ile gerçekleştirilmiştir. Bu yazılıma işlenecek olan kavşak sayım verileri ile farklı meta-sezgisel algoritmalarla gecikme süresini en aza indirgenecek şekilde yeşil süreleri elde edilmesi amaçlanmıştır. Optimizasyon metodu olarak; Genetik Algoritma (Genetic Algorithm - GA), Cıvık Mantar Algoritması (Slime Mould Algorithm -SMA), Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO), Salp-Sürü Algoritması (Sap-Swarm Algorithm - SSA), Gri Kurt Algoritması (Grey Wolf Algorithm - GWO), Ağaç-Tohum Algoritması (Tree Seed Algorithm - TSA), Harris Hawks Algoritması (Harris Hawks Optimization - HHO) ve Bal Porsuğu Algoritması (Honey Badger Algorithm - HBA) algoritmaları kullanılmıştır. Farklı meta-sezgisel metotlarla elde edilen yeşil süreler karşılaştırılmış ve ayrıca kavşakta tramvay öncelik sistemi incelenerek kavşağa etkileri analiz edilmiştir.

Anahtar Kelimeler: Sinyalizasyon, Meta-sezgisel, Optimizasyon Algoritmaları, Kavşak, Trafik

TRAFFIC SIGNAL OPTIMIZATION WITH META-HEURISTIC ALGORITHMS

OLCAY ADIYAMAN

KAYSERİ UNİVERSİTY, INSTİTUTE OF GRADUATE EDUCATİON

MASTER THESİS, JANUARY 2023

ABSTRACT

Traffic has become one of the most important factors that negatively affect our lives with the increase in urbanization on a global scale. The traffic congestion resulting from the growing number of people and vehicles has a serious impact on human life. The reasons such as fuel costs of traffic, time, stress, increase in carbon emissions have serious effects both in terms of economy and climate change. Many studies are being carried out to solve this problem today. In order to reduce waiting times in traffic, studies are carried out on the continuous widening of roads, physical changes at junctions and junction signaling. The first element after safety that shows the performance of the intersection signaling system is the delay. In this study, a study has been carried out to reduce the waiting time at intersections by using meta-heuristic optimization methods. The aim of the study is to reduce the waiting time at intersections by using meta-heuristic optimization methods.

In the study, the delay time optimization is carry out with MATLAB software using the American model. With the intersection count data to be processed into this software, it is aimed to obtain green times by minimizing the delay time with different meta-heuristic algorithms. The intersection count data is being processed into this software with the aim of obtaining green times by minimizing the delay time using different meta-heuristic algorithms. In our optimization, Genetic Algorithm (GA), Slime Mould Algorithm (SMA), Particle Swarm Optimization (PSO), Salp-Swarm Algorithm (SSA), Gray Wolf Algorithm (GWO), Tree-Seed Algorithm (TSA), Harris Hawks Algorithm (HHO) and Honey Badger Algorithm (HBA) algorithms are used. The green times obtained by different meta-heuristic methods are compared and also the tram priority system at the intersection is examined and its effects on the intersection are analyzed.

Keywords: Signaling, Meta-heuristic, Optimization Algorithm, Junction, Traffic

İÇİNDEKİLER

META-SEZGİSEL ALGORİTMALAR İLE TRAFİK SİNYAL OPTİMİZASYONU

BİLİMSEL ETİĞE UYGUNLUK i
TEZ İNTİHAL RAPORU TESLİM FORMU ii
YÖNERGEYE UYGUNLUK iii
KABUL VE ONAY iv
TEŞEKKÜR v
ÖZET vi
ABSTRACT vii
İÇİNDEKİLER viii
SİMGELER VE KISALTMALAR x
TABLOLAR LİSTESİ xi
ŞEKİLLER LİSTESİ xii

BİRİNCİ BÖLÜM

GİRİŞ

1.1. Giriş **1**
1.2. Literatür Değerlendirmesi **3**
1.3. Tezin Amacı ve Önemi **6**
1.4. Yöntem **6**

İKİNCİ BÖLÜM

KAVŞAK VE SİNYALİZASYON

2.1. Kavşak **8**
2.2. Kavşak Sinyalizasyonu **8**
2.2.1. İzole Kavşak Sinyalizasyon Sistemleri **10**
2.2.1.1. Sabit Zamanlı Sinyalizasyon Sistemi **10**
2.2.1.2. Trafik Uyarmalı Sinyalizasyon Sistemi **10**
2.2.1.3. Yaya Uyarmalı Sinyalizasyon Sistemi **10**
2.2.1.4. El ile Uyarmalı Sinyalizasyon Sistemi **11**
2.2.2. Koordine Sinyalizasyon Sistemleri **11**
2.3. Sinyalizasyon Kavramları **11**

2.4. İngiliz Yöntemi ile Sinyalizasyon 12
2.4.1 Doygun Akım Hesaplanması 13
2.4.2 Kavşakta Gecikme Hesaplanması 17
2.5. Amerikan Yöntemi ile Sinyalizasyon 17
2.5.1 Doygun Akım Hesaplanması 20
2.5.2 Kavşakta Gecikme Hesaplanması 23
2.5.3 Kavşakta Hizmet Düzeyi Belirlenmesi 23

ÜÇÜNCÜ BÖLÜM

META-SEZGİSEL ALGORİTMALAR

3.1. Genetik Algoritma (GA) 25
3.2. Cıvık Mantar Algoritması (SMA) 27
3.3. Parçacık Sürü Optimizasyonu (PSO) 31
3.4. Gri Kurt Algoritması (GWO) 31
3.5. Ağaç-Tohum Algoritması (TSA) 33
3.6. Salp-Sürü Algoritması (SSA) 34
3.7. Harris Hawks Algoritması (HHO) 35
3.8. Bal Porsuğu Algoritması (HBA) 37

DÖRDÜNCÜ BÖLÜM

ARAŞTIRMA ve BULGULAR

4.1. Giriş 39
4.2. Tek Kavşak Üzerinde Analiz ve Ölçümler 39
4.3. Tek Kavşak Üzerinde Optimizasyon 46
4.4. Tramvayın Kavşak Üzerinde Etkisi 47
4.5. Çoklu Üzerinde Analiz ve Ölçümler 48
4.6. Çoklu Kavşakların Eşzamanlı Çalıştırılması 52
4.7. Çoklu Kavşak Üzerinde Optimizasyon 53

BEŞİNCİ BÖLÜM

SONUÇ VE TARTIŞMALAR

5.1. Sonuçlar 55

KAYNAKÇA 57
ÖZGEÇMİŞ 61

SİMGELER VE KISALTMALAR

KISALTMALAR

BM	: Bulanık Mantık
EEMD	: Ensemble Ampirical Mode Decomposition
FCMNN	: Fuzzy C-mean Neural Network
GA	: Genetic Algorithm
GWO	: Grey Wolf Optimization
HBA	: Honey Badger Algorithm
HCM	: Highway Capacity Manual
HHO	: Harris Hawks Optimization
NN	: Neural Network
PA	: Petri Ağları
PHF	: Peak Hour Factor
PSO	: Particle Swarm Optimization
SMA	: Slime Mould Algorithm
SSA	: Salp Swarm Algorithm
SUMO	: Simulation of Urban Mobility
TSA	: Tree Seed Algorithm
YSA	: Yapay Sinir Ağları

TABLOLAR LİSTESİ

Tablo 2.1. Kavşak Akım Kolu Genişliğine Göre Doygun Akım Değerleri 14
Tablo 2.2. Farklı Türlerdeki Taşıtlar Birim Otomobil Eşdeğerileri 15
Tablo 2.3. Şerit Genişliği Düzetlme Faktörü 20
Tablo 2.4. Sinyalize Kavşak için Hizmet Düzeyi Otomobil Modu 24
Tablo 4.1. Organize Kavşak Şerit Sayı ve Genişlikleri 42
Tablo 4.2. TS6407'ye Göre Birim Otomobil Eşdeğer Katsayıları 43
Tablo 4.3. Organize Kavşak Araç Sayım Değerleri 43
Tablo 4.4. Organize Kavşak 7.30-8.30 Arası Birim Otomobil Cinsinden Araç Sayım Değerleri 45
Tablo 4.5. Organize Kavşak Optimizasyon Sonuçları 46
Tablo 4.6. Tınaztepe Kavşak Şerit Sayı ve Genişlikleri 50
Tablo 4.7. Tınaztepe Kavşak 7.30-8.30 Arası Birim Otomobil Cinsinden Araç Sayım Değerleri 51
Tablo 4.8. Organize Kavşak ile Tınaztepe Kavşak Birlikte Çalışma Optimizasyon Sonuçları 54

ŞEKİLLER LİSTESİ

Şekil 1.1. Yıllara Göre Taşıt Sayıları Değişimi 2
Şekil 2.1. İngiliz Yöntemine Göre Doygun Akım Grafiği 13
Şekil 2.2. Amerikan Yöntemi İşleyiş Şeması 18
Şekil 3.1. Genetik Algoritma Akış Şeması 26
Şekil 3.2. Cıvık Mantarların Yiyecek Arama Algoritması 28
Şekil 3.3. Cıvık Mantar Algoritması (SMA) Süreç Sıralaması 29
Şekil 3.4. Gri Kurtların Hiyerarşi Piramidi 32
Şekil 3.5. HHO Algoritması Kademeleri 36
Şekil 3.6. Ters Kare Yasası 37
Şekil 4.1. Metro Kavşak Google Earth Görüntüsü 40
Şekil 4.2. Organize Kavşak Google Earth Görüntüsü 41
Şekil 4.3. Organize Kavşak Sinyal Krokisi 44
Şekil 4.4. Organize Kavşak Mevcut Sinyal Planı 45
Şekil 4.5. Tınaztepe Kavşak Google Earth Görüntüsü 49
Şekil 4.6. Tınaztepe Kavşak Sinyal Krokisi 51
Şekil 4.7. Tınaztepe Kavşak Mevcut Sinyal Planı 52
Şekil 4.8. Organize Kavşak-Tınaztepe Kavşak Ana Arterler 53
Şekil 4.9. Yeşil Dalga Çalışması Sinyal Planı 53

BİRİNCİ BÖLÜM

Giriş

1.1. Giriş

Günümüzde şehirlerde yaşarken karşılaşılan en büyük sorunlardan bir tanesi trafik sıkışıklıkları ve trafikte oluşan bekleme süreleridir. Artan nüfus, kentleşme, araç sayılarının artışı ve insanların özel araçlara yöneliminin artması kavşaklardaki kapasite kullanımını artırmıştır. Bu artış bekleme sürelerinin artmasına, üretim kayıplarına, trafikteki insanlar için strese, enerji kayıplarına ve karbon emisyonunun artmasına neden olmaktadır [1].

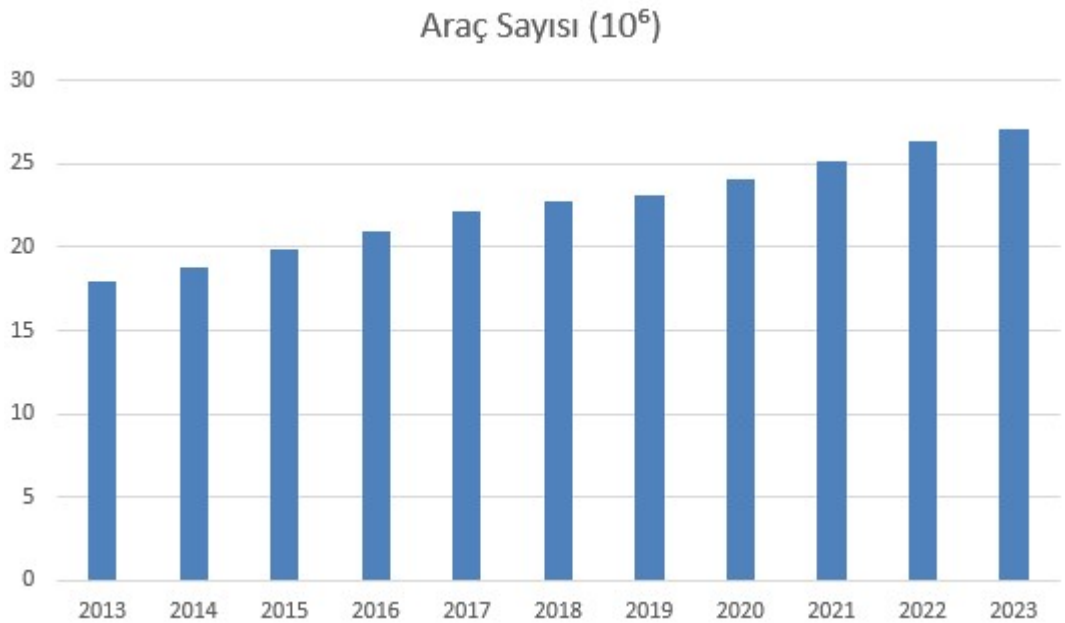
Günümüzde şehirlerdeki trafik sorunu, hızla büyüyen nüfus, artan araç sayısı, kentleşme ve toplu taşıma yerine özel araçlara yönelimin artmasıyla birlikte giderek daha büyük bir sorun haline gelmektedir. Şehir trafiğinin yoğunluğu, zaman kaybı, hava kirliliği, karbon emisyonunun artması, enerji kayıpları ve stres gibi bir dizi olumsuz etkiye neden olmaktadır. Şehir trafiğinin en önemli sorunlarından biri, yolların yetersiz kapasitesidir. Mevcut yolların artan araç trafiğini karşılamakta zorlanması, trafik sıkışıklığına ve yolculuk sürelerinde uzamalara neden olmaktadır. Bu sorunu çözmek için, altyapı geliştirmeleri ve yeni yol ağlarının inşası önemli adımlardır. İyi planlanmış yollar, trafik akışını düzenleyerek sürücülere daha hızlı ve verimli bir ulaşım sağlar. Bununla birlikte, yolların genişletilmesi tek başına yeterli değildir. Araç sahiplerinin çoğunluğu genellikle tek başlarına seyahat ederken, araç paylaşımı ve toplu taşıma sistemlerinin teşvik edilmesi önemlidir. Toplu taşıma, bireysel araç kullanımının azalmasına ve trafik yoğunluğunun azaltılmasına yardımcı olabilir. Hızlı ve etkili toplu taşıma ağları, insanları kendi araçlarından vazgeçmeye teşvik ederken çevreye vermiş olduğu olumsuz etkileri de azaltabilir [1].

Akıllı ulaşım sistemleri, şehir trafiğini yönetmek için kullanılan bir diğer çözüm olarak öne çıkmaktadır. Bu sistemler, trafik akışını izlemek ve yönlendirmek için teknoloji ve veri analizi kullanır. Trafik ışıklarının senkronizasyonu, trafik sıkışıklığını azaltmaya

yardımcı olmakla birlikte gerçek zamanlı trafik bilgilerini sürücülere ileterek alternatif rotalar sunmak gibi uygulamalar, trafik yoğunluğunu dağıtmaya yardımcı olabilir. Yaya ve bisiklet yollarının iyileştirilmesi de şehir trafiğinin azaltılmasında etkili bir yöntemdir. Yaya dostu kaldırımlar ve bisiklet yolları, insanları araç yerine yürümeye veya bisiklete binmeye teşvik eder. Bu, hem trafiği rahatlatır hem de sağlıklı yaşam tarzını teşvik eder. Şehir trafiğini azaltmak için uzaktan çalışma ve esnek çalışma saatlerinin teşvik edilmesi gibi politikalar da oldukça önemlidir. İşletmelerin çalışanlarına esnek çalışma seçenekleri sunması, iş saatlerindeki yoğunluğu azaltarak trafik sıkışıklığını azaltabilir [2].

Şehir trafiği, karmaşık, çok boyutlu ve Şekil 1.1'de de görüldüğü üzere sürekli araç sahipliğinin artması nedeniyle büyüyen bir sorundur. Ancak, altyapı geliştirmeleri, toplu taşıma teşvikleri, akıllı ulaşım sistemleri ve bisiklet/yaya yollarının iyileştirilmesi gibi çözümlerle bu soruna etkin bir şekilde müdahale edilebilir. Ayrıca, işbirliği ve politika değişiklikleri de şehir trafiğini azaltmakta önemli bir rol oynar. Bu çözümler bir araya geldiğinde, daha sürdürülebilir ve verimli bir şehir ulaşımı sağlanabilir.

Şekil 1.1. *Yıllara Göre Taşıt Sayıları Değişimi*



Kaynak: TÜİK.(Türkiye İstatistik Kurumu) (2022, Mart 18). Yıllara göre motorlu kara taşıtları sayısı,2022. https://data.tuik.gov.tr/Bulten/Index?p=Motorlu-Kara-Tasitlari-Aralik-2022-49436

Bu çalışma kapsamında kavşaklarda oluşan bekleme sürelerinin azaltılması amacıyla sinyalizasyon sisteminde yeşil ışık ve gecikme süreleri meta-sezgisel optimizasyon algoritmaları kullanılarak optimize edilmiştir. Çalışmada tek kavşak ve çoklu kavşak

optimizasyonları yapılmıştır. Çoklu kavşak uygulamasında Kayseri'de birbirini takip eden iki kavşak (Organize kavşağı ve Tınaztepe kavşağı) üzerinde çalışma gerçekleştirilmiştir. Gerçek araç sayımları yapılmış ve optimizasyonda iki kavşak arasında birbirini takip eden gruplarda araçlar bekleme yapmayacak şekilde tasarım yapılmıştır. Ayrıca incelediğimiz iki kavşakta tramvay güzergâhındadır ve tramvayların sinyal sürelerine ve bekleme sürelerine olumsuz etkileri vardır. Tramvay geçmesi esnasında kavşakta paralel çalışabilecek ana arterlerde yeşil yanmakta ve geçişi tamamladıktan sonra kavşak sinyal planında çalışmasına kaldığı yerden devam etmektedir. Kayseri raylı sistemde tramvay öncelikli sistem uygulaması vardır. Uygulanan tramvay öncelikli sistemde tramvay geldiğinde kavşak kontrol cihazına bir talep sinyali gider ve kavşak kontrol cihazı tramvay geçişine uygun faza geçer. Tramvay kavşağı terk ettiğinde yine kavşak kontrol cihazına bir iptal sinyali gider. Kavşak sinyal sistemi tekrar eski çalışma moduna (sinyal planına) geri döner. Yapılan optimizasyonda tramvaya yer verilmemiş ancak etkileri araştırılmıştır. Optimizasyonda 8 farklı meta-sezgisel optimizasyon algoritması kullanılmıştır. Kavşaklarda optimizasyon işlemi Genetik Algoritma (Genetic Algorithm - GA), Cıvık Mantar Algoritması (Slime Mould Algorithm -SMA), Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO), Salp-Sürü Algoritması (Sap-Swarm Algorithm - SSA), Gri Kurt Algoritması (Grey Wolf Algorithm - GWO), Ağaç-Tohum Algoritması (Tree Seed Algorithm - TSA), Harris Hawks Algoritması (Harris Hawks Optimization - HHO) ve Bal Porsuğu Algoritması (Honey Badger Algorithm - HBA) algoritmalarıyla gerçekleştirilmiştir.

1.2. Literatür Değerlendirmesi

Kavşaklarda bekleme sürelerinin optimizasyonu ile ilgili birçok çalışma yapılmıştır. Ancak sorunun güncelliğini koruması ve gün geçtikçe boyutlarının da büyümesi nedeniyle şehirlerde trafik yönetimi önemli bir konudur.

Harb ve arkadaşları çalışmalarında gerçek verilerle sinyalizasyon sisteminin analizi için Bulanık Mantık (BM) metodunu kullanılmışlardır. Çalışılan kavşak sabit zamanlı yönetilmektedir. MATLAB yazılımı ile kavşak analiz edilmiştir. Kavşağa gelen araçların sayıları giriş parametreleri olarak alınmış, BM ile işlenmiş ve sonrasında da BM'nin yeşil sürelerine karar vermesi sağlanmıştır. Klasik sabit zamanlı sistem mantığına göre BM'nin verdiği yeşil süreleri daha verimli olmuştur [3].

Gençosman kavşaktaki araç sayımlarının sensörler aracılığı ile yapıldığı bu çalışmasında her bir şerit için ayrı veri toplamıştır. Şerit bazlı trafik verileri ışığında tam adaptif uyarmalı bir sinyal sistemi tasarlamıştır. Buradaki trafik verileri veri tabanına kaydedilmiş ve bu veriler saatlere göre sınıflandırılmıştır. Toplanan veriler simülasyon yazılımına giriş dataları olarak verilmiştir. Oluşan bu veri tabanı ile Radyal tabanlı yapay sinir ağı eğitilmiş ve daha sonra en uygun yeşil süreleri tahmin etmeye çalışılmıştır. Radyal Tabanlı Ağın (Radial Basis Function - RBF) tahminleri simülasyon sistemi ile karşılaştırılmış ve daha sonrasında başarılı olduğu görülmüştür [4].

Doğan ve arkadaşları BM temelli çalışmalarında iki amaç hedeflemişlerdir. İlki kırmızı sinyaldeki duruş sürelerini BM tabanlı sinyal sistemi ile iyileştirmektir. Diğer ikinci amaçları ise iyileştirme çalışmasının çıktılarını tam uyarmalı ve ön zamanlı sinyal denetim sistemleri ile kıyaslanmasıdır. Bu çalışma sonucunda BM ile yapılan düzenlemenin araç sayılarına göre %7 ila %65 arasında performans artışı sağladığı ve bekleme sürelerinin azaldığı gözlemlenmiştir [5].

Mahmood uygulama içeren çalışmasında, farklı koşullardaki trafik akışlarında ortalama duruş sürelerini azaltmayı amaçlamıştır. Bunun için sinyalizasyon sistemi denetiminde gerçek zamanlı kavşağı çalıştırmak için BM tabanlı 2 kademeli ve 3 bölümden oluşan bir optimizasyon metodu önermiştir. Burada önerilen sistem bir sonraki fazın hangisi olacağına karar verme, faz süresinde yeşil uzatma veya durdurma ve son olarak yeşil yanan fazın devam etmesi ya da yeni başka bir faza yeşil yakma kararını verecek çıktıların üretilmesidir. Çalışma Simulation of Urban Mobility (SUMO) simülasyon yazılımıyla test edilmiştir. Yapılan sistem dört farklı kavşağa uygulanmıştır. BM denetleyicisi ile klasik mantık denetleyicisi mukayese edilmiş performans artışı ve bekleme sürelerinde azalmalar gözlemlenmiştir [6].

Harb ise diğer bir çalışmasında dört akım kolu olan bir kavşakta gerçek verilerle çalışmak için araç sayım verileri toplamıştır. Bu veriler ile BM yöntemini ve Petri Ağlarını (PA) sinyalizasyon sistemine uygulamıştır. Çalışmanın sonunda bu iki yöntemde yapılan sistemler ile klasik sistem karşılaştırılmıştır. Oluşan çıktılar değerlendirilmiş, sonuçta en iyi performansın sırasıyla PA sonrasında BM ve en son klasik mantıkta meydana geldiği görülmüştür [7].

Tunç ve arkadaşı yaptıkları çalışmada trafik sinyalizasyon sistemini BM ve PI tipi optimizasyon sistemleri ile kontrol etmeyi amaçlanmıştır. SUMO programı aracılığı ile

çalışmalarında doğru sonuçlar elde edilmeye çalışmışlardır. Yapılan çalışma neticesinde elde edilen çıktılar klasik tabanlı sinyalizasyon sistemi ile kıyaslanmıştır [8].

Tang ve arkadaşları yaptıkları tahminleme çalışmasında akıllı ulaşım sistemlerinde anahtar parametrenin trafik tahmini olduğunu ve bu doğruluğun artırılmasının öneminden bahsetmişlerdir. Çalışmalarında trafik tahminini oluşan gürültü, çevresel faktörüler gibi tahmin doğruluğunu azaltan unsurlardan en az etkilenecek model tasarlamayı hedeflermişler. Çalışmaları, tahmin doğruluğunu iyileştirmek için Ensemble Ampirical Mode Decomposition (EEMD) gürültü arındırma şemalarına ve Fuzzy C-mean Neural Network (FCMNN)'e dayalı öğrenme algoritmasını sınıflandıran hibrit bir tahminleme modeli tasarımıdır. Model eğitim sürecinde, Minneapolis şehrindeki otoyol ağında tespit edilen trafik hacmine dayalı tahmin hatalarına göre EEMD ve FCMNN'deki birkaç temel parametre belirlemişlerdir. Model doğrulamasında, Artificial Neural Network (ANN), EEMD + ANN, FCMNN ve EEMD+FCMNN dahil olmak üzere tek ve çok aşamalı dört aday modelin tahmin doğruluğunu tahmin etmek için yaygın olarak kullanılan üç hata değerlendirme göstergesi uygulanmıştır. Örnek olay incelemesinde gösterilen sonuçlar, gürültüden arındırma yöntemleriyle birleştirilen tahmin modellerinin, gürültüden arındırma algoritması benimsemeyen modellerden daha üstün olduğunu göstermektedir. Ayrıca, FCMNN öğrenme yöntemini kullanan model, geleneksel ANN modelinden daha yüksek tahmin doğruluğu üretebilir. Ek olarak, FCMNN'nin uzun vadeli tahmin performansı da ANN'ninkinden çok daha iyidir. Çünkü bu alt NN sistemi, daha iyi optimizasyon etkisi elde etmek için her bir sınıflandırma modeline göre eğitilmiştir. Bu çalışmada özetlenen sonuçlar, sinyalleri yönetirken yüksek tahmin doğruluğuna göre yönetme ve kontrol stratejileri tasarlamasına yardımcı olmaktadır [9].

Kavşakta sürelerin optimizasyonu ile ilgili yapılan çalışmalardan bir diğerinde ise Karaağaç'ın yapmış olduğu sezgisel arama algoritmaları ile trafik sinyal optimizasyonudur. Çalışmada literatürde sıkça kullanılan GA ve Standart Diferansiyel Gelişim Algoritmasının yanında literatüre daha sonradan kazandırılan Bernstein Arama Diferansiyel Gelişim Algoritması ile birkaç kavşaktan oluşan trafik ağlarındaki sinyal sürelerinin aynı anda optimizasyonu hedeflenmiştir. Trafik ağı olarak Kayseri merkezde bulunan trafik ağı seçilmiş, gerçek zamanlı araç sayım sonuçları alınmıştır. Optimizasyon süreci iteratif yaklaşımlı olduğundan trafik ağında denenmesi mümkün değildir. Bu

yüzden gerçek zamanlı bir simülasyon programı üzerinden araç verileri simüle edilmiştir. Ayrıca sezgisel metotların sonuçları da karşılaştırılmalı olarak verilmiştir [10].

Bir diğer çalışmada Korkmaz sinyalizasyon sistemi ile çalışan kavşaklar için yeşil süreleri, faz planı ve gecikme sürelerini optimize etmiştir. Optimizasyonda BM, Çiçek Tozlaşma Algoritması ve Karga Arama Algoritmaları kullanmıştır. Çiçek Tozlaşma Algoritması faz planında, Karga Arama Algoritması ve BM ise yeşil süreleri belirleme konusunda optimizasyonu sağlanmıştır. Farklı trafik hacimlerinde ve farklı kavşak sistemlerinde KU-Trsim simülasyon programı ile çalışma simüle edilmiştir [11].

1.3. Tezin Amacı ve Önemi

Günümüzde şehirlerdeki nüfus sürekli artmaktadır. Şehirlerin artan nüfusları, şehirlerde yaşamı ciddi derecede zorlaştırmakta ve şehir yönetimlerini çözüm bulmaya zorlamaktadır. Varlıkları ve kaynakları akıllı bir şekilde yönetmek için verileri toplamak ve verileri etkin kullanma zarureti ortaya çıkmıştır. Bu çalışmaların bütünü akıllı şehir uygulamaları olarak adlandırılır. Akıllı şehir uygulamaları aydınlatma, ulaşım, eğitim, sağlık ve trafik gibi birçok farklı alana uygulanabilmektedir. Özellikle büyük şehirlerde artan araçlarla birlikte bir noktada diğerine ulaşım oldukça zaman almakta ve maddi anlamda maliyetli olmaktadır. Dolayısıyla akıllı şehir uygulamalarının temel uygulama alanlarından birisi de trafiktir. Trafik büyük kentlerin gün geçtikçe büyüyen sorunlarının başında gelmektedir. Bireysel araç sahiplenmesinin artması kavşak noktalarında bekleme sürelerinin artmasına, yolculuk maliyetlerinin ve karbon emisyonunun artmasına bununla birlikte insanlarda stres kaynağı oluşturmaya devam etmektedir [1].

Günümüzde kavşaklarda sabit zamanlı ve dinamik (uyarmalı) tip olmak üzere iki tür kontrol cihazı bulunmaktadır [12]. Kavşaklarda bekleme sürelerinin optimizasyonu ile ilgili birçok çalışma yapılmıştır. Bu çalışmada seçilen bir kavşak yapısına meta-sezgisel optimizasyon metotları uygulanarak kavşakta bekleme süresini en aza indirgemek hedeflenmektedir. Böylelikle hem karbon salınımını azaltarak daha çevreci hem de daha az yakıt harcayarak daha verimli ve insanlara zaman tasarrufu sağlayacak bir sistemin tasarlanması amaçlanmaktadır. Ayrıca çalışmada iki kavşak birlikte optimize edilmiştir. Birbirini takip eden kavşağın birbirini izleyen sinyal grupları arasında senkronizasyon sağlanarak kavşaklarda takip eden sinyal gruplarında hiç beklemeden geçen gruplar oluşturulmuştur.

1.4. Yöntem

Daha önceden belirlenen iki kavşakta gerçek zamanlı araç sayımları ile kavşaktaki sinyal sürelerinin meta-sezgisel algoritmalar kullanılarak optimizasyonu gerçekleştirilmiştir. Optimizasyon işlemleri için MATLAB programlama dili kullanılmıştır. Optimizasyonda tramvaya yer verilmemiş ancak etkileri araştırılmıştır. Optimizasyonda 8 farklı meta-sezgisel optimizasyon algoritması kullanılmıştır. GA, SMA, PSO, SSA, GWO, TSA, HHO ve HBA optimizasyon algoritmaları kullanılarak kavşaklardaki gecikme süreleri optimize edilmiştir. Optimizasyon algoritmalarında uygulanan model Amerikan yöntemi esas alınarak oluşturulmuş, gecikme sürelerinin yanı sıra dışarıdan girilen yeşil sürelerine göre de gecikme süresi hesaplayabilmektedir. Mevcut sinyal planı zamanlarına göre algoritmalar gecikme süresini bu şekilde hesaplayacaktır. Sonrasında optimizasyon algoritmaları kavşakların girilen araç sayım değerleri ve kavşak özelliklerine göre en az gecikmeye neden olacak yeşil sürelerini hesaplamıştır. Bulduğu en iyi sonuçlar ile mevcuttaki gecikme süreleri karşılaştırılmıştır.

İKİNCİ BÖLÜM

Kavşak Ve Sinyalizasyonu

2.1. Kavşak

Kavşak, farklı yolların kesiştiği veya birleştiği noktadır. Yolların kavşaklarda kesiştiği yerlerde trafik akışının yönlendirilmesi ve düzenlenmesi için çeşitli trafik düzenlemeleri ve işaretlemeler bulunur. Kavşaklar, karayolu ağında trafik akışının güvenli ve düzenli bir şekilde gerçekleşmesini sağlar. Kavşakları eşdüzey kavşaklar ve katlı kavşaklar olmak üzere iki tür olarak tanımlayabiliriz [12].

- Eşdüzey Kavşaklar
    - Denetimli Eşdüzey Kavşaklar (Sinyalize)
    - Denetimsiz Eşdüzey Kavşaklar
- Katlı Kavşaklar

2.2. Kavşak Sinyalizasyonu

Trafik yönetimi günümüzde giderek önem kazanan bir konudur. Yoğun araç trafiği ve kavşaklardaki karmaşa, trafik akışını düzenlemek ve kazaları önlemek için çeşitli çözümler gerektirmektedir. Bu çözümlerden biri de kavşak sinyalizasyonudur. Sinyalizasyon, araç, bisiklet ve yaya trafiğini düzenlemek için kullanılan bir sistemdir. Bu sistem, trafik işaretleri, trafik ışıkları ve yol işaretlerinden oluşur. Temel amacı, trafik akışını kontrol etmek, düzenlemek ve trafik kazalarını önlemektir. Sinyalizasyon sistemi sayesinde, farklı yönlerden gelen araçlar ve yayalar belirli bir düzende hareket eder ve çarpışmalar en aza indirgenir.

Kavşak sinyalizasyonu, sinyalizasyonun en önemli uygulama alanlarından birisidir. Kavşaklar, trafik akışının en yoğun olduğu ve karmaşıklaştığı noktalardır. Kavşak sinyalizasyonu, bu karmaşıklığı yönetmek ve trafik akışını düzenlemek amacıyla kullanılır. Kavşaklarda yer alan trafik ışıkları, araçların hareketini kontrol eder. Belirli bir

sıra ve zamanlama ile yanıp sönen trafik ışıkları, araçlara ve yayalara geçiş hakkı verir ve trafiğin düzenli ve güvenli bir şekilde ilerlemesini sağlar.

Sinyalizasyon sistemi zaman içinde gelişmiş ve teknolojinin ilerlemesiyle birlikte yeniliklerle geliştirilmiştir. Günümüzde geleneksel trafik ışıklarının yanı sıra, akıllı trafik yönetim sistemleri de kullanılmaktadır. Bu sistemler, trafik yoğunluğunu gerçek zamanlı olarak izler ve trafik ışıklarının zamanlamasını buna göre ayarlar. Böylece, trafik akışı daha verimli hale getirilir, kavşaklardaki bekleme süreleri azalır ve trafik sıkışıklığı önlenir.

Kavşak sinyalizasyonu, trafiğin yönetimi açısından önemli avantajlar sunar. Trafik akışının düzenlenmesi ve trafik sıkışıklığının azaltılması, daha hızlı ve güvenli bir ulaşım sağlar. Ayrıca, trafik kazalarını da önler. Trafik ışıkları, araçların ve yayaların geçiş hakkını belirler ve çarpışmaların önüne geçer. Bu da trafiğin daha güvenli bir şekilde yönetilmesini sağlar.

Sonuç olarak, kavşak sinyalizasyonu, trafik yönetimi, kapasitenin kullanımı, bekleme zamanlarının azaltılması, ekonomiklik, yakıt tasarrufu, karbon emisyonunun azaltılması ve trafik güvenliği için önemli bir araçtır. Gelişen teknolojiyle birlikte akıllı trafik yönetim sistemleri kullanımı artmakta ve artırılması da gerekmektedir. Günümüzde global dünyada en büyük problem enerjidir. Enerji tasarrufu yapacak bütün çalışmalara ülkeler ciddi destek sağlamaktadırlar. Trafiği daha verimli hale getirmek, yakıt, zaman ve enerji tasarrufu sağlamak için kavşaklardaki fazladan bekleme süreleri azaltılmalıdır. Ayrıca kavşaklara uygulanan tek başına çözümlerin yeteri derecede problemi çözemediği trafikte birbirini takip eden kavşakların birlikte değerlendirilmesi gerektiği ve yeni dönemde çalışmaların bu yönde yoğunlaşmış olduğu bilinmektedir. Bir kavşakta bireysel yapılan iyileştirmenin bir sonraki kavşakta ciddi derecede sıkışıklığa neden olabildiği görülmüştür. Hatta son dönemlerde büyük yazılım firmaları ve sinyalizasyon firmaları birkaç kavşak değil bir şehri bütün olarak ele alan çalışmalar yürütmektedir. Tüm bu çalışmalar trafik güvenliğini artırmak, daha düzenli ve hızlı bir trafik akışı sağlamak içindir.

Kavşak Sinyalizasyon Sistemlerini iki ana başlıkta inceleyebiliriz [12];

- İzole Kavşak Sinyalizasyon Sistemleri
- Koordine Kavşak Sinyalizasyon Sistemleri

2.2.1. İzole Kavşak Sinyalizasyon Sistemleri

İzole Kavşak Sinyalizasyon Sistemi kavşağın tek başına hareket ettiği, etrafındaki kavşaklardan herhangi bir bilgi alışverişi ve etkileşimde bulunmadığı sistemlerdir. Bu tip sinyalizasyon sistemlerini aşağıdaki gibi 4 farklı sistem olarak sınıflandırabiliriz [13]. Bunlar;

- Sabit zamanlı sinyalizasyon sistemi,
- Trafik uyarmalı sinyalizasyon sistemi,
- Yaya Uyarmalı Sinyalizasyon Sistemi,
- El ile Uyarmalı Sinyalizasyon Sistemidir.

2.2.1.1.Sabit Zamanlı Sinyalizasyon Sistemi

Sabit zamanlı sinyalizasyon sistemleri, trafik sinyallerinin belirli bir zaman aralığına göre değişmesini sağlar. Bu sistemde, kavşaktaki trafik sinyalleri belirli bir döngü süresi boyunca aynı sırayla çalışır [13]. Örneğin, kırmızı ışık 25 saniye yanar, ardından yeşil ışık 70 saniye yanar ve bu sıra tekrar eder. Bu süreler, trafik yoğunluğu, saat dilimi ve kavşağın özellikleri gibi faktörlere bağlı olarak belirlenir. Sabit zamanlı sinyalizasyon sistemleri daha basit ve düşük maliyetli olduğu için yaygın olarak kullanılır. Ancak, trafik akışındaki değişkenliklere veya öncelikli yolların varlığına uyum sağlayamazlar. Bu nedenle, sabit zamanlı sistemlerde bazı yönlerde beklemeler olabilir veya boş yollarda ciddi zaman kayıpları yaşanabilir.

2.2.1.2.Trafik Uyarmalı Sinyalizasyon Sistemi

Trafik uyarmalı sinyalizasyon sistemlerinde trafik sinyalleri sabit zamanlıdaki gibi belirli bir sıra ve süre ile değil sahadan dedektörler vasıtasıyla alınan araç sayıları, geçiş talepleri ve yoğunluk bilgileri doğrultusunda sistemin yeşil ışık sıra ve sürelerini değiştirebildiği sistemlerdir [13]. Kavşaklarda tali yollar araç olmadığı sürece kırmızı yanacak şekilde tasarlanan sistemler olduğu gibi bütün kavşak kollarında geçiş talepleri, araç sayıları ve yoğunluk bilgilerini alarak tam adaptif çalışan sistemler de mevcuttur.

2.2.1.3.Yaya Uyarmalı Sinyalizasyon Sistemi

Genelde sürekli yaya geçitlerinin olmadığı, araç trafiğinin yoğun olduğu yerlerde kullanılır [13]. Trafik akışını kesmeme adına yaya ışığı sürekli kırmızı, oto ışıkları sürekli yeşil olacak şekilde çalışır. Yayaların geçebilmesi için kavşaklara konulan yaya

butonlarına yayaların basarak geçiş hakkı talep etmeleri sonrasında yayalara yeşil ışık yanar. Kayseri'de genellikle raylı sistem istasyonlarının yaya geçitlerinde bu sinyalizasyon sistemi modeli uygulanmaktadır.

2.2.1.4.El ile Uyarmalı Sinyalizasyon Sistemi

Bu tip sinyal sisteminde kumanda ile kavşakta belirli gruplara geçiş verilmesi ya da geçiş sürelerinin uzatılması yöntemiyle çalışmaktadır. Sabit zamanlı sinyalizasyon sistemlerinde çalışan belirli saatlerde ve zamanlarda trafik yoğunluklarının aşırı derecede değiştiği kavşaklarda sıklıkla kullanılmaktadır [13].

2.2.2. Koordine Sinyalizasyon Sistemi

Birden fazla kavşağın birbirleriyle iletişim kurarak trafik akışını optimize etmeyi amaçlayan bir sistemdir. Bu sistemde, kavşaklardaki trafik sinyalleri birbirleriyle senkronize edilir ve bir koordinasyon içinde çalışır [13]. Genellikle trafiğin yoğun olduğu ana yollar veya birbirine yakın kavşaklar arasında kullanılır. Sistem, kavşaklardaki trafik akışını daha verimli hale getirmek ve trafik sıkışıklığını azaltmak için tasarlanmıştır.

Bu sistemlerde, her kavşakta birbiriyle haberleşen kavşak kontrol cihazları, gps sistemleri ve araç algılama sensörleri gibi teknolojiler kullanır. Sensörler, kavşaktaki trafik yoğunluğunu veya araç hareketlerini izler ve bu bilgileri merkezi bir kontrol sistemiyle paylaşır. Merkezi kontrol sistemi, kavşaklardaki trafik durumunu analiz eder ve trafik akışını optimize etmek için en uygun zamanlamayı belirler. Bu sistem Trafik sinyallerinin senkronize edilmesiyle çalışır. Bu senkronizasyon, trafik akışını düzenlemek ve araçların kavşaklara daha düzenli bir şekilde ulaşmasını sağlamak için önemlidir. Trafik sinyalleri, belirli bir zamanlama ve sıralama ile çalışır ve sürücülere doğru zamanda ve uygun sürelerde yeşil ışık verilir. Bu sistemin diğer bir uygulama yöntemi de ana yolların belirli bir hızda sürekli yeşil olacak şekilde çalışmasının sağlandığı sistemleri de gösterebiliriz. Günümüzde bu tip sistemleri desteklemek amacıyla büyük şehir trafik koordinasyon yazılımları geliştirilmiş ve kullanılmaktadır.

2.3. Sinyalizasyon Kavramları

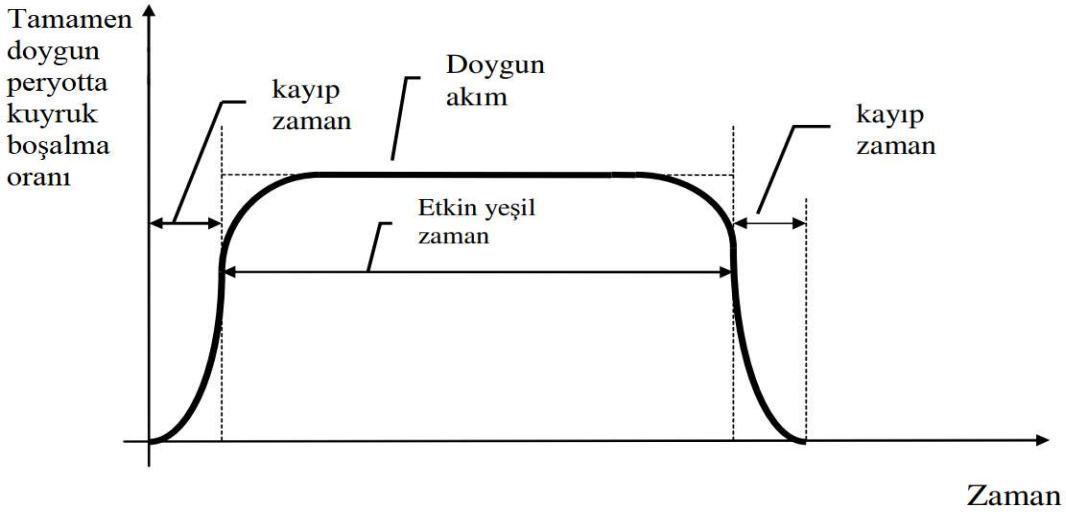
Kavşak sinyalizasyon sistemi trafik akışını düzenleyen**,** araçların kavşaktan güvenli bir şekilde geçişini sağlayan sistemin genel adıdır. Sinyalizasyon özellikle trafik kazalarının önlenmesinde önemli rol oynar. Sinyalizasyon sisteminde kullanılan temel kavramlar aşağıdaki gibidir [12];

- **Faz :** Kavşakta en az bir guruba (akıma) geçiş verilen süredir.
- **Devre :** Tüm grupların geçişini tamamladığı ve sürekli tekrarlan döngüdür.
- **Devre Süresi (Periyot) :** Her akıma en az bir geçişin verildiği toplam süredir.
- **Kırmızı Süre :** Bir fazdaki kırmızı yanma süresi
- **Yeşil Süre :** Bir fazdaki yeşil yanma süresi
- **Sarı Süre :** Bir fazdaki sarı yanma süresi
- **Kayıp zaman :** Kavşakta tüm grupların (akımların) kırmızı olduğu kullanılamayan zaman
- **Kapasite :** Kavşaktan veya şeritten belirli bir sürede geçebilecek olan maksimum araç sayısı.
- **Gecikme :** Bir aracın kavşak bölgesini geçiş esnasında çeşitli nedenlerle kaybettiği zaman,

2.4. İngiliz Yöntemi ile Sinyalizasyon

1958 yılında Webster tarafından İngiltere'de önerilmiş analitik bir sistemdir. Bu yöntem, trafik akışını ve kavşaktaki trafik durumunu analiz ederek, sinyal programlarını belirlemeyi hedefler. Kavşak kapasitesi, belirli bir zaman diliminde kavşaktan geçebilecek araç sayısı olarak tanımlanır. Kapasite hesaplamaları, trafik akış hızı, yoğunluk, sinyalizasyon planı, kavşak geometrisi ve diğer faktörleri dikkate alarak yapılır. Bu hesaplamalar, genellikle matematiksel modeller, saha çalışmaları ve trafik simülasyonları gibi yöntemler kullanılarak yapılır. Kavşak Sinyalizasyon sisteminde trafiğin akış hızı sabit değildir. Yani bir kola yeşil yandığı anda araçların kalkışları ve belirli bir hıza ulaşana kadar belirli bir zaman geçecektir. Bu zamandan sonraki akan akım doygun akım, doygun akımın aktığı sürede etkin yeşil süre olarak tanımlanmaktadır. Şekil 2.1' deki grafikte, İngiliz yöntemine göre doygun akım grafiği gösterilmiştir. [14].

Şekil 2.1. *İngiliz Yöntemine Göre Doygun Akım Grafiği*



Kaynak: F.V. Webster, ve B.M. Cobbe, *Traffic Signals,* H.M.S.O., Londra, 111, 1966.

Burada kayıp zamanlar incelendiğinde araçların kırmızı bekleme sonrasında kalkış ve hızlanma esnasında bir süre kaybettiği bir kayıp zaman görünmektedir. Belirli bir süre sonrasında akış hızı normale dönmektedir [13].

Etkin yeşil süre akım koluna yanan yeşil süreden kayıp sürelerin çıkarılması ile elde edilir.

$$g = G - I \tag{2.1}$$

Kapasite ise etkin yeşil sürenin devre süresine oranının doygun akım ile çarpılması ile bulunur.

$$\boldsymbol{Q} = \left(\frac{g}{c}\right) S \tag{2.2}$$

Burada G yeşil süreyi, g etkin yeşil süreyi, C devre süresini, I kayıp süreyi ve S ise doygun akımı ifade etmektedir.

2.4.1. Doygun Akım Hesaplanması

Doygun akım hesaplanırken İngiliz yönteminde birçok faktör etkili olmaktadır. Bunları sırayla açıklayacak olursak;

- **Kavşak Kol Genişliği Etkisi**

Doygun akım ve kayıp zaman belirlenmesinde çevresel faktörlerin etkisinin çok fazla olmasından dolayı en sağlıklı yöntem sahadan ölçümdür [13].

Doygun Akım ;

$$S = 160w \tag{2.3}$$

Denklemiyle hesaplanır. S burada doygun akım, w ise akım kolu genişliğini ifade etmektedir. Tablo 2.1'de 10ft ile 17ft arasındaki kavşak akım kolu genişliğine göre doygun akım değerleri yer almaktadır [1].

Tablo 2.1. *Kavşak Akım Kolu Genişliğine Göre Doygun Akım Değerleri*

W (ft)	10	11	12	13	14	15	16	17
S (bo/sa)	1850	1875	1900	1950	2075	2250	2475	2700

K**aynak:** Z. A. Polat, A. Memduhoğlu, M. Acar, H. Duman, "Kentsel Büyüme ile Motorlu Araç Trafiği Yoğunluğu Arasındaki İlişkinin Belirlenmesi", *Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi,* 6(2), 442-451, 2017.

- **Trafik Kompozisyon Etkisi**

Hesaplamayı yaparken araç türlerinin trafiğe olan etkileri farklı farklıdır. Bu etki ifade edilirken her bir aracın otomobil birim eşdeğeri alınarak değerlendirilmektedir. Buna göre araçların birim otomobil eşdeğerleri Tablo 2.2'de verilmiştir [14].

Tablo 2.2. *Farklı Türlerdeki Taşıtlar Birim Otomobil Eşdeğerleri*

1 ağıt taşıt	1.75 b.o.
1 otobüs	2.25 b.o.
1 tramvay	2.50 b.o.
1 hafif taşıt	1 b.o.
1 motosiklet	0.33 b.o.
1 bisiklet	0.20 b.o.

K**aynak:** F.V. Webster, ve B.M. Cobbe, *Traffic Signals,* H.M.S.O., Londra, 111, 1966.

- **Eğim Etkisi**

Doygun akım değerlerine eğiminde ciddi derecede etkisi bulunmaktadır. Akım kolu için her %1'lik çıkış eğitimi %3 doygun akım azalışına her %1'lik iniş eğitimi %3 doygun akım artışına neden olmaktadır.

- **Sola Dönen Araçların Etkisi**

Kavşakta ters yönde sola dönen araçlar kavşakta kilitlenmelere neden oluyor ise kavşağın kapasitesini ciddi derecede kötü etkileyecektir. Sola dönen araçların etkisini dört grupta inceleyebiliriz [13].

❖ Kesişen akım kolu olmadığı ve özel sola dönüş şeridi olmadığı durumlarda sola dönen araçların doygun akıma etkisi dikkate alınmaz.

❖ Kesişen akım kolu olmadığı ve özel sola dönüş şeridi varsa bu durumlarda sola dönen araçların doygun akıma etkisi aşağıdaki gibi bulunur.

Burada r dönüş yarıçapıdır.

Sola dönüş tek şerit ise :

$$S = \frac{1800}{1+\frac{5}{r}} \quad \text{b.o./sa} \qquad (2.4)$$

Sola dönüş iki şerit ise :

$$S = \frac{3000}{1+\frac{5}{r}} \quad \text{b.o./sa} \tag{2.5}$$

- ❖ Kesişen akım kolu varsa ve özel sola dönüş şeridi olmadığı durumlarda sola dönen araçlar doğru gidecek araçların gecikmesi yönünde doygun akıma olumsuz bir etkisi olur. Ayrıca yeşil sürenin uzamasına ve diğer akım kollarının bekleme süresinin artmasına sebep olacaktır. Burada her sola dönen taşıtı 1,75 doğru giden taşıt cinsinden hesaplayarak etkiyi bulabiliriz [13].

- ❖ Kesişen akım kolu varsa ve özel sola dönüş şeridi olduğu durumlarda aynı koldaki doğru giden araçlara etkisi olmayacaktır ancak diğer akım kollarının gecikmesine neden olacağı için yine yukarıdaki gibi her sola dönen taşıtı 1,75 doğru giden taşıt cinsinden hesaplayarak etkiyi bulabiliriz [13].

- **Sağa Dönen Araçların Etkisi**

Sağa dönen araçların etkisi kavşaktaki dönüşün geometrik yapısına, dönüş sayısına ve burayı kullanan yaya sayılarına bağlıdır. Eğer sağa dönen taşıtların kavşaktaki araç sayısına oranı %10'un altında ise etki dikkate alınmaz ancak %10'dan fazla bir araç var ise her sağa dönen taşıtı 1,75 doğru giden taşıt cinsinden hesaplayarak etkiyi bulabiliriz [13].

- **Yaya Etkisi**

Kavşağı kullanan yaya sayısı az ise dikkate etki alınmaz, yaya sayısı fazla ise etkinin büyüklüğü süre uzatma yapılması veya yeni faz eklenmesi durumuna göre ayrıca incelenmelidir [13].

- **Dur Çizgisine Park Eden Araçların Etkisi**

Kavşakta dur çizgisine yakın bir şekilde park etmiş araçların kavşak boşaltma şeridinde oluşturdukları azalma nedeniyle doygun akıma olumsuz bir etkisi bulunmaktadır. Bu etki aşağıdaki gibi hesaplanır [13].

$$W_{kayıp} = 55 - \frac{0.9(z-25)}{9} \tag{2.6}$$

Burada *z* park edilen konumdaki araçların dur çizgisine uzaklığını temsil etmektedir.

2.4.2. Kavşakta Gecikme Hesaplanması

Sinyalizasyon sisteminde meydana gelen gecikme hesabı birçok araştırmaya konu olmuştur. Gecikme hesap sonuçları kavşağın ve sinyalizasyon sisteminin performansını göstergelerinden kabul edilmektedir. Sinyalizasyon sistemine sahip bir kavşakta gecikme kavşakta araçların beklemeden geçtiği süre ile yavaşlama, durma gibi nedenler sonrasında kavşağı geçme süreleri arasındaki fark olarak tanımlanabilir [13].

Burada d her akım kolundaki her bir araç için gecikme ortalamasını, λ etkin yeşil süresinin devre süresine oranını ve x doygunluk derecesini göstermektedir.

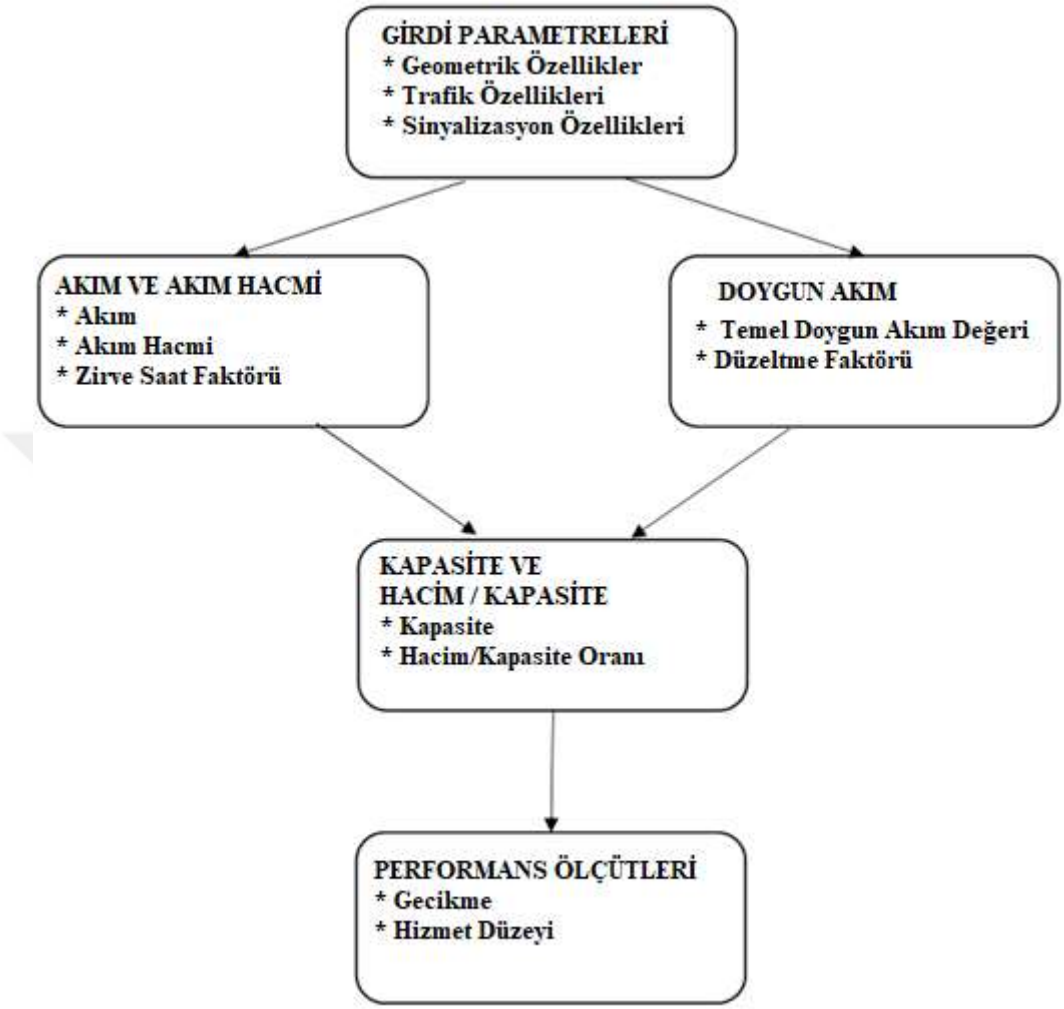
Doygunluk derecesi $x = q/\lambda S$ formülü ile hesaplanmaktadır.

$$d = \frac{c(1-\lambda)^2}{2(1-\lambda\ )} + \frac{x^2}{2q(1-x)} - 0.65\left(\frac{c}{q^2}\right)^{1/3}.x^{(2+5\lambda)} \qquad (2.7)$$

2.5. Amerikan Yöntemi ile Sinyalizasyon

Highway Capacity Manual (HCM) Amerika Birleşik Devletleri'nde ulaşım mühendisliği alanında kullanılan referans bir kitaptır [15].

Şekil 2.2. *Amerikan (HCM) Yöntemi İşleyiş Şeması*



Kaynak: O. Eraslan, "Işıklı Kavşaklarda Amerikan ve Avusturalya Yöntemleri ile Gecikme Analizi ve Örnek Bir Kavşak çözümü", Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2008.

Amerikan yöntemi işleyiş şeması Şekil 2.2'de gösterildiği gibidir. Amerikan yöntemi ile analiz için gereken temel kavramlar, tanımlamalar ve hesaplamalar aşağıdaki gibidir [16].

- **Akım hacmi**

Hesap yapılacak belirli bir süre içinde kavşağa gelen taşıt miktarı olarak tanımlanmaktadır. Burada yapılacak analiz için 1 saat yada 15 dakikalık periyotlar belirlenebilir. Trafik akışındaki dalgalanmanın derecesini ve zirve yoğunluğun ne kadar sürekli olduğu belirlemek için kullanılır.

$$\vartheta = \frac{V}{\mathrm{PHF}} \tag{2.8}$$

Burada ϑ en yoğun saat için 15dk. trafik akım hacmini, *V* bir saatlik trafik hacmini ve *PHF* (peak-hour factor) ise en yoğun saat faktörünü ifade etmektedir. Eğer elinizde saatli akım hacmi yok ise 15dk. akım hacmi alınarak saatlik akım hacmi tahmini bu formül kullanılmaktadır [16].

- **Kapasite**

Sinyalize kavşaklarda kapasite doygun akım, yeşil süre ve devre süresine bağlıdır. Bir devresi süresinde geçen maksimum araç sayısı olarak da tanımlanabilir.

$$c_i = s_i \frac{g_i}{\mathrm{C}} \tag{2.9}$$

Bu denklemde c_i, *i.* şerit grubu kapasitesini, s_i ise *i.* şerit grubu doygun akımı, g_i *i.* şerit grubu etkin yeşil süresi ve *C* devre süresine göstermektedir [16].

- **Hacim/Kapasite Oranı**

Sinyalizasyon sistemi çalışan kavşakların analizinde hacminin kapasiteye oranı X sembolü ile gösterilir. Doygunluk derecesi olarak adlandırılır.

$$X = \frac{\vartheta}{\mathrm{c}} \tag{2.10}$$

Burada *X* doygunluk derecesi, *c* kapasite ve ϑ trafik akım hacmini araç/saat cinsinden ifade etmektedir [16].

- **Kritik Akım**

Doygunluk derecesi kavşaktaki yeşil sürelerin belirlenmesinde önemli rol oynar. Hangi akım kolunun kaç saniye yeşil yanması gerektiği aslında doygunluk derecesi ile belirlenir. Bir kavşakta en yoğun araçların olduğu en yüksek doygunluk derecesine sahip akım kritik akım olarak adlandırılır [16].

$$Xc = \left(\frac{C}{C-L}\right) \sum_{i \in ci} y_{c,i} \tag{2.11}$$

Bu denklemde Xc kavşak kritik doygunluk derecesi, *c* döngü süresi, $y_{c,i}$ kritik faz akım oranı (ϑ_i / S_i) , *L* iste toplam kayıp süresidir.

2.5.1. Doygun Akım Hesaplaması

Doygun akım her bir şerit için ayrı ayrı hesaplanır. Temel doygun akım değeri HCM'de bir şeritte 1900 otomobil/saat olarak önerilmektedir. Temel doygun akımın dışında kavşaktaki bir şeridin doygun akımını etkileyen birçok faktör bulunmaktadır. Doygun akım değerini aşağıdaki formül ile hesaplayabiliriz.

$$s = s_0 \; f_w \; f_{HV} \; f_g \; f_p \; f_{bb} \; f_a \; f_{LU} \; f_{LT} \; f_{RT} \; f_{Lpb} \; f_{Rpb} \qquad (2.12)$$

Bu formülde s doygun akımı, s_0 temel doygun akımı, f_w şerit genişliği düzeltme faktörünü, f_{HV} trafik akışı ağır araç düzeltme faktörünü, f_g yaklaşım kolu eğim düzeltme faktörünü, f_p park şeridi ve parklanma düzeltme faktörünü, f_{bb} kavşak alanı içerisinde duran otobüs etkisi düzeltme faktörünü, f_a alan (kavşağın bulunduğu bölge) tipi için düzeltme faktörünü, f_{LU} şerit kullanım için düzeltme faktörünü, f_{LT} sola dönen araç düzeltme faktörünü, f_{RT} sağa dönen araç düzeltme faktörünü, f_{Lpb} sola dönen araçlar için yaya düzeltme faktörünü, f_{Rpb} sağa dönen araçlar için yaya ve bisikletli düzeltme faktörünü ifade etmektedir [15].

Yukarıdaki formülde kullandığımız faktörler aşağıda açıklanmıştır.

- **Şerit genişliği düzeltme faktörü**

Şerit genişliği ayarlama faktörü f_w , dar şeritlerin doygunluk akış hızı üzerindeki olumsuz etkisini hesaba katar ve geniş şeritlerde artan akış hızına izin verir.

Tablo 2.3. *Şerit Genişliği Düzeltme Faktörü*

Ortalama Şerit Genişliği (ft)	Düzeltme Faktörü (f_w)
$< 10.0^a$	0.96
≥10.0-12.9	1.00
>12.9	1.04

Kaynak: Ulaşım Araştırma Kurulu, Ulaşım Araştırma Konseyi, *Highway Capacity Manual 2000 (HCM, (2000),* TRB publication, Washington, 2000.

Şerit genişliği düzelme faktörüne ait açıklama Tablo 2.3'de verilmiştir. Bu tabloda a faktörü ortalama şerit genişliği 8ft ve üzerinde uygulanır. Standart şeritler 12ft. genişliğindedir [15].

- **Trafik akışı ağır araç düzeltme faktörü**

Aşağıdaki bağıntı ile hesaplanır.

$$f_{HV} = \frac{100}{100+P_{HV}(E_T-1)} \quad (2.13)$$

Burada P_{HV} akım kolu üzerindeki ağır araç oranı ve E_T ağır araçların eşdeğer hafif araç sayısını göstermektedir [15].

- **Trafik yaklaşım kolu eğim düzeltme faktörü**

Eğim düzeltme faktörü, yaklaşma eğiminin araç performansı üzerindeki etkilerini hesaba katar. Bu faktörün değerleri şu şekilde hesaplanır:

$$f_g = 1 - \frac{P_g}{200} \quad (2.14)$$

Burada P_g araçlar için yaklaşım eğimidir. Bu faktör -%6,0 ile +%10,0 arasında değişen dereceler için geçerlidir. Yokuş yukarı eğimin pozitif bir değeri ve yokuş aşağı eğimin negatif bir değeri vardır [15].

- **Park şeridi ve parklanma düzeltme faktörü**

Park etme yok ise bu faktörün değeri 1.00'dir. Park varsa, bu faktörün değeri (2.15) ile hesaplanır.

$$f_p = \frac{\left(N-0.1-\left(\frac{18N_m}{3600}\right)\right)}{N} \geq 0.050 \quad (2.15)$$

Bu denklemde N_m şeridi kullanan araçlarda park eden araç oranı, *N* ise şerit sayısını göstermektedir [15].

- **Kavşak alanı içerisinde duran otobüs etkisi düzeltme faktörü**

Kavşak bölgesi ile otobüs durağı arasındaki uzaklık 250ft.(76m)'den az ise bu faktör dikkate alınır. Bunun ile ilgili bağıntı:

$$f_{bb} = \frac{\left(N-\left(\frac{14.4N_b}{3600}\right)\right)}{N} \geq 0.050 \quad (2.16)$$

Burada N_b şeridi kullanan saatteki otobüs sayısı, N ise şerit sayısıdır [15].

- **Alan (kavşağın bulunduğu bölge) tipi için düzeltme faktörü**

Alan tipi düzeltme faktörü kavşağı benzer konumlardaki kavşaklara göre verimsizliğini açıklar. Merkezi iş bölgesi olan yerlerde 0.90 alınır [15].

- **Şerit kullanım için düzeltme faktörü**

Bu faktör birden fazla özel şeride sahip bir şerit grubu için doygunluk akış hızını tahmin etmek için kullanılır. Şerit grubunun bir ortak şeride veya bir özel şeride sahip olması durumunda, bu faktör 1.0'dır [15].

- **Sola Dönen Araçlar için düzeltme faktörü**

Sola dönüş ayar faktörü, öncelikle sola dönüş yolu geometrisinin doygunluk akış hızı üzerindeki etkisini yansıtmak için tasarlanmıştır. Bu ayarlama faktörünün değeri şu şekilde hesaplanır.

$$f_{LT} = \frac{1}{E_L} \tag{2.17}$$

E_R sola dönüş için ayrılmış şeritten doğru giden araç sayısıdır [13].

- **Sağa Dönen Araçlar için düzeltme faktörü**

Sağa dönüş ayar faktörü, öncelikle sağa dönüş yolu geometrisinin doygunluk akış hızı üzerindeki etkisini yansıtmak için tasarlanmıştır. Bu ayarlama faktörünün değeri şu şekilde hesaplanır.

$$f_{RT} = \frac{1}{E_R} \tag{2.18}$$

E_R sağa dönüş için ayrılmış şeritten doğru giden araç sayısıdır [15].

- **Sağa, Sola dönen araçlar için yaya ve bisikletli düzeltme faktörü**

Sağa, sola dönen araçlar için yaya ve bisikletli düzeltme faktörünün temeli, dönen araçlar, yayalar ve bisikletler arasındaki kesişmeyi açıklayan kesişim bölgesi doluluk kavramına dayanmaktadır. İlgili kesişim bölgesi işgali, karşı araç akışının da sola dönüş hareketiyle çelişkili olup olmadığını dikkate alır. Kesişim bölgesinin işgal edildiği yeşil sürenin oranı, ilgili doluluk ve dönüş yapan araçlar için alış şerit sayısının bir fonksiyonu olarak belirlenir [15].

2.5.2. Kavşakta Gecikme Hesaplanması

Kavşaktaki gecikme kavşağın en önemli performans göstergelerindendir. Bu alanda bir çok çalışma yapılmıştır. Kavşağın fiziksel yapısıyla beraber sinyalizasyon sistemi de gecikmeyi etkileyen faktörlerdendir. Gecikme hesabı, analiz süresi boyunca gelen tüm araçların yaşadığı ortalama kontrol gecikmesini temsil eder. Analiz süresi sona erdikten sonra hala kuyrukta olan bu araçların neden olduğu gecikmeleri içerir. Belirli bir şerit grubu için kontrol gecikmesi, toplam gecikme, aşağıdaki denklem kullanılarak hesaplanır [12].

$$d = d1(PF) + d2 + d3 \tag{2.19}$$

Burada d her bir araç için kontrol gecikmesi, *d1* uniform kontrol gecikmesi, *d2* kademeli gecikme, *d3* ilk kuyruk gecikmesi ve *PF* ise ilerleme düzeltme faktörünü göstermektedir. Uniform kontrol gecikmesi (*d1*) Döngü boyunca rastgele olduğu varsayılan gecikmeyi hesaplamaktadır. Ayrıca döngü sırasında bir etkin yeşil periyodu ve bu periyot boyunca derecelendirilen bir doygunluk akışını varsayar. Uniform kontrol gecikmesi denklem (2.20) ile hesaplanır [16].

$$d1 = \frac{0.5C(1-\frac{g}{C})^2}{1-[\frac{\min(1,X)g}{C}]} \tag{2.20}$$

Kademeli gecikme terimi döngü bazında varış sayısındaki rastgele değişimden kaynaklanan gecikmeyi açıklar. Aynı zamanda, analiz süresi boyunca kapasiteyi aşan talebin neden olduğu gecikmeyi de hesaba katar. Analiz süresi boyunca talebin kapasiteyi aşma miktarı burada karşılanmayan talep olarak anılır. Kademeli gecikme denklemi, önceki analiz periyodunda karşılanmayan talep nedeniyle ilk kuyruğun olmadığı varsayımı kullanılarak türetilmiştir.

$$d2 = 900T[(X_A - 1) + \sqrt{(X_A - 1)^2 + \frac{8kIX_A}{c_A T}}] \tag{2.21}$$

$$X_A = v/c_A \tag{2.22}$$

Burada X_A ortalama hacim kapasite oranıdır.

Yukarıdaki denklemler ile hesaplanan şerit bazlı kavşak gecikmeleridir. Buradan akım kolu için hesaplamalar yapılarak denklem (2.23)'de gösterildiği gibi bu gecikmelerin toplanması ile toplam kavşak gecikmesi hesaplanır [16].

$$d_i = \frac{\sum d_i v_i}{\sum v_i} \tag{2.23}$$

2.5.3. Kavşakta Hizmet Düzeyi Belirlenmesi

Bu yöntem ile hesaplamalar yapıldığında iki ana sonuç ortaya çıkmaktadır. Kavşak içerisindeki kritik akımlar ve her bir akım kolu için hacim/ kapasite oranı. Bu iki sonuçtan da kavşak hizmet düzeyi değeri elde edilir. Hizmet düzeyi kavşak için bir değerlendirme yapar. Gecikme verisi burada artan seyahat süresini gösterir. Tablo 2.4'de otomobiller için sinyalize kavşaklarda araç başına gecikmeye göre hizmet düzeyi sınıflandırması gösterilmektedir [12].

Tablo 2.4. *Sinyalize Kavşak için Hizmet Düzeyi Otomobil Modu*

LOS (Hizmet Düzeyi)	Araç Başı Kontrol Gecikmesi
A	**≤ 10**
B	**>10-20**
C	**>20-35**
D	**>35-55**
E	**>55-80**
F	**>80**

Kaynak: S. Yavuz, "Çiçek Tozlaşma Algoritması ile Devre Süresi Modellerinin Gerçekleştirilmesi", Yüksek Lisans Tezi, Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü, Kırıkkale, 2019.

ÜÇÜNCÜ BÖLÜM

Meta-Sezgisel Algoritmalar

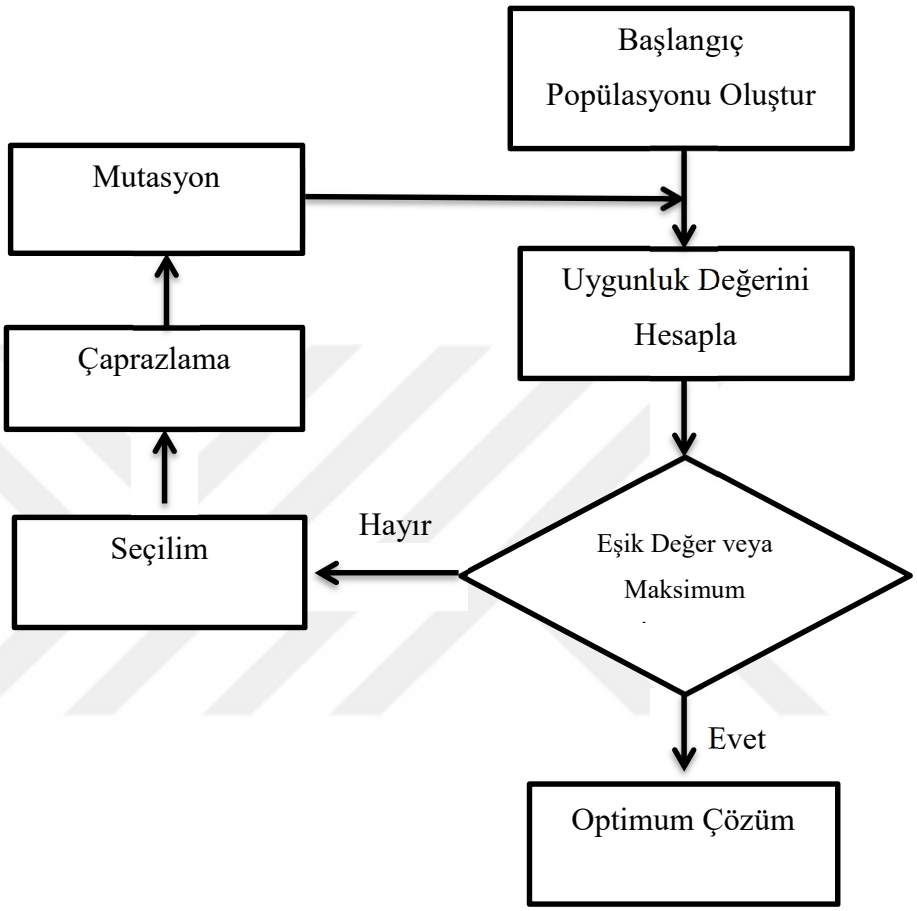
3.1.Genetik Algoritma (GA)

GA, biyolojik evrim prensiplerinden esinlenerek, eniyileme ve arama problemlerini çözmek için kullanılan bir optimizasyon yöntemidir. GA, bir başlangıç popülasyonundan başlayarak aday çözümleri evrimleştirir ve en iyi çözümü bulmayı hedefler. Bir optimizasyon problemine en iyi çözümü (en optimum) bulmaya çalışan meta-sezgisel bir optimizasyon tekniğidir [17]. Bu algoritma, birçok optimizasyon problemine uygulanmakta, karmaşık gerçek dünya problemlerinde kullanılmakta ve başarılı sonuçlar elde edilmektedir.

GA, bir sorunu çözmek için bir dizi aday çözümü temsil eden "bireyler" içeren bir "popülasyon" kullanır. Her birey, genler olarak adlandırılan parametrelerin bir kombinasyonunu temsil eder. Genellikle genler, problem bağlamına bağlı olarak binary (0/1) veya gerçel sayılar olabilir.

GA, popülasyondaki bireylerin doğal seçilim, çaprazlama (çiftleşme) ve mutasyon gibi genetik operasyonlara tabi tutulduğu bir iteratif süreç izler. Bu süreç, doğal seçilim yoluyla daha iyi uyum sağlayan bireylerin seçilerek bir sonraki nesile aktarılmasını içerir. Çaprazlama, seçilen bireylerin genetik materyallerini birleştirerek yeni bireyler oluştururken, mutasyon ise bazı genlerin rasgele değiştirilmesini sağlar. Şekil 3.1'de GA'nın genel bir akış diyagramı gösterilmiştir [18].

Şekil 3.1. *Genetik Algoritma Akış Şeması*



Kaynak: D. Whitley, "A genetic algorithm tutorial", *Statistics and Computing,* . Technical Report CS-93-103, Mart 1993.

Her bir nesil, bireylerin uyum seviyeleri veya performanslarına göre değerlendirilir. Bu değerlendirme, bir "uygunluk fonksiyonu" kullanılarak gerçekleştirilir. Uygunluk fonksiyonu, problem bağlamına göre tanımlanan bir metriği temsil eder. GA, belirli bir sonlandırma kriterine ulaşana veya belirli bir iterasyon sayısına ulaşana kadar nesilleri oluşturup işlemi tekrarlar.

Bu şekilde, GA, popülasyonun genetik operasyonlarla evrimleşerek daha iyi çözümler üretmesini sağlar. Başlangıçta rastgele olan bireyler, iterasyonlar ilerledikçe daha iyi çözümlere yaklaşır. GA, özellikle karmaşık, çok boyutlu ve optimize edilmesi zor problemlerde etkili bir çözüm sunabilir.

GA'nın çalışma şeklini yani akışını özetleyecek olursak;

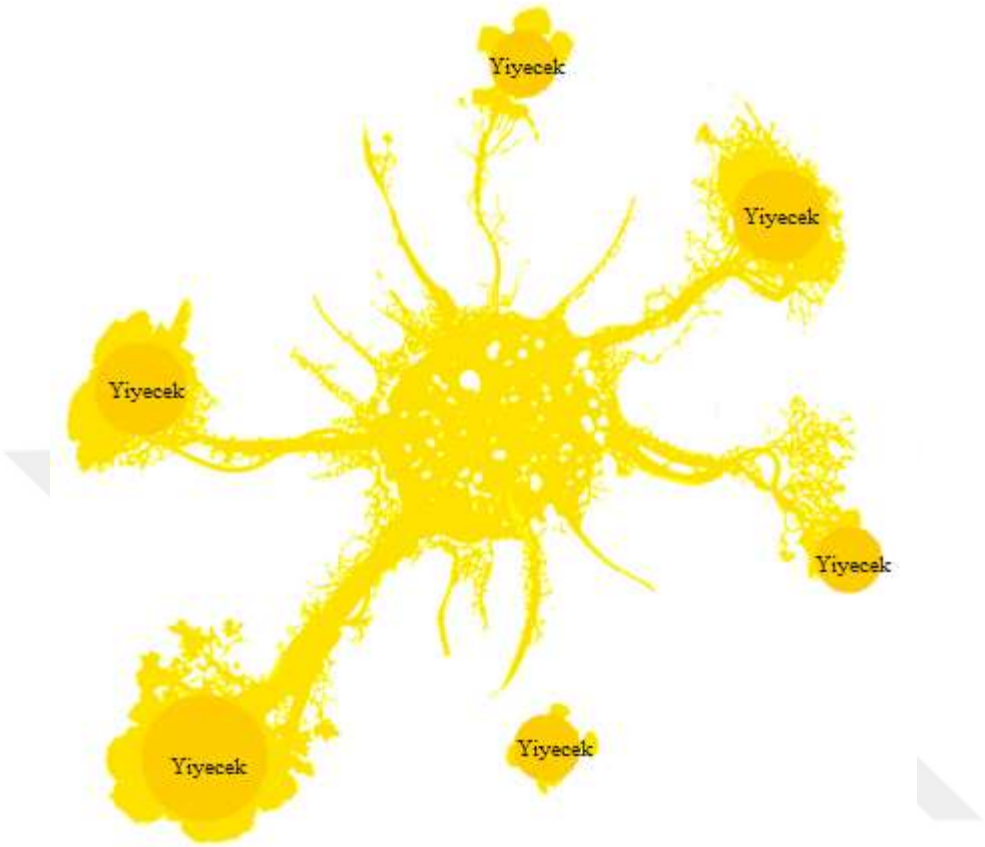
- İlk olarak popülasyon belirlenir, başlangıç popülasyonunu rastgele oluştur ve popülasyonun boyutu belirlenir. Uygunluk fonksiyonu tanımlanır.
- İkinci adım olarak uygunluk değerlendirmesi yapılır, her bireyin uygunluk değeri hesaplanır ve her birey uygunluk değerine göre sıralanır.
- Sonlandırma kriterini kontrol edilir. Yani maksimum iterasyon sayısına veya kabul edilebilir bir uygunluk değerine ulaşılıp ulaşılmadığını kontrol edilir.
- Seçilim stratejisini belirlenir. Yüksek uygunluk değerine sahip bireyleri seçilir.
- Seçilen bireyler arasında çaprazlama gerçekleştir. Çaprazlama oranını belirlenir ve yeni bireyler oluşturmak için çaprazlama yapılır.
- Yeni oluşturulan bireyler üzerinde mutasyon gerçekleştir. Mutasyon oranını belirlenir ve belirli genlerin değerini rasgele değiştirerek çeşitliliği artırılır.
- Çaprazlama ve mutasyon sonucunda oluşan bireyleri yeni nesile eklenir ve ilk popülasyon ile birleştirilir.
- Sonrasında ikinci adıma geri dönülerek yeni nesil üzerinde uygunluk değerlendirmesi ve sonlandırma kontrolü yapılır.
- Sonuç olarak sonlandırma kriteri sağlandığında veya maksimum iterasyon sayısına ulaşıldığında en iyi uyum sağlayan çözüm raporlanır.

Bu adımlar, GA'nın temel akışını gösterir. Akış diyagramında döngü, uygunluk değerlendirmesi ve sonlandırma kontrolü ile birlikte seçim, çaprazlama ve mutasyon adımları arasında gerçekleşir. Bu döngü, bireylerin nesiller boyunca evrimleşerek daha iyi çözümler üretmesini sağlar.

3.2.Cıvık Mantar Algoritması (SMA)

Li ve arkadaşları geliştirdikleri bu algoritma ile Physarum Polycephalum mantarının yaşadıkları ortamlardaki yiyecek bulma karakterinden ilham alarak önerdikleri bir optimizasyon algoritmasıdır. Mantar besinleri koku yoluyla arar ve onlara ulaşmak için Şekil 3.2'deki gibi uzantılar oluşturur daha sonrasında da kaynağın etrafını sararak sindirir [19].

Şekil 3.2. Cıvık *Mantarların Yiyecek Arama Algoritması*



Kaynak: S. Li, H. Chen, M.Wang, A.A. Heidari, S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization", *Future Generation Computer Systems*, 111, 300-323, 2020.

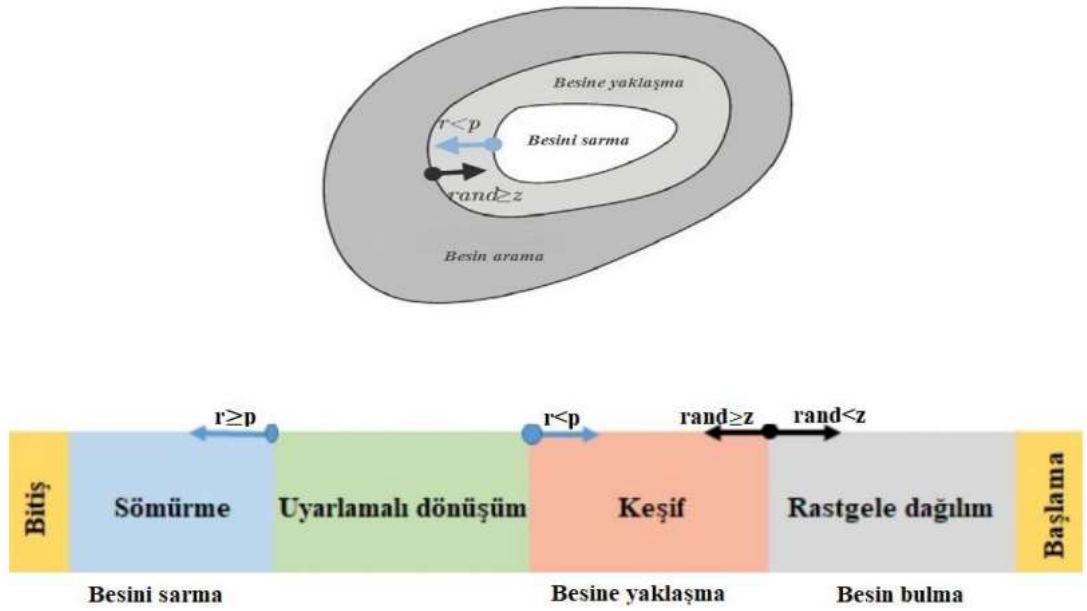
Cıvık mantar algoritmasının çalışma şeklini yani akışını özetleyecek olursak;

- Optimizasyon yapılacak probleme uygun bir model ve uygunluk fonksiyonu belirlenir. Başlangıç için belirli sayıda mantar parçacığı popülasyonu oluşturulur. Her mantar parçacığı bir çözümü temsil eder.
- Mantar parçacıkları mevcut konumunda hareket ederek potansiyel çözüm alanında dolaşır. Hareket ederken çevresel bilgileri kullanır ve her zaman en iyi çözüm bölgesini bulmaya eğilimlidir.
- Mantar parçacıkları yakın çevrelerindeki diğer mantar parçacıkları ile etkileşim ve halindedir. Bu etkileşim ile bilgi alışverişi sağlanarak parçacıkların hareketlerinde, keşiflerinde ve karar vermesinde etkilidir.

- Her mantar parçacığı kendi konumunu ve hedefini sürekli günceller. Uygunluk fonksiyonuna göre en iyi çözümü ulaşmaya çalışır. Yeni konumlar her zaman daha iyi çözümler için bir potansiyeldir.
- Arama ve konum güncellemesi iterasyon sayısına yada sonlandırma kriterlerine ulaşana kadar devam eder. Her iterasyonda en iyi çözüm arayışı devam eder.
- Algoritma sonlandığında elde edilen en iyi uygun çözüm optimizasyon probleminin çözümü olarak raporlanır.

Şekil 3.3'de cıvık mantarların süreç sıralaması tarif edilmektedir [20].

Şekil 3.3. *Cıvık Mantar Algoritması (SMA) Süreç Sıralaması*



Kaynak: I.B. Uğur, "Büyük Ölçekli Çelik Yapıların Sezgisel Optimizasyon yöntemleriyle Optimum Tasarımı", Doktora Tezi, Dicle Üniversitesi, Fen Bilimleri Enstitüsü, 2021.

$$\vec{X}(t+1) = \begin{cases} r_1 . (UB - LB) + LB & eğer\ r_2 < z \quad (1) \\ \vec{X}_B(t) + \overrightarrow{vb}.\left(\overrightarrow{W}.\vec{X}_A(t) - \vec{X}_B(t)\right) & eğer\ r_2 < p \quad (2) \\ \overrightarrow{vc}.\vec{X}(t) & eğer\ r_3 \geq p \quad (3) \end{cases} \qquad (3.1)$$

$$p = \tanh|S(i) - DF| \qquad (3.2)$$

$$a = \operatorname{arctanh}(-\left(\frac{t}{\max_t}\right) + 1) \tag{3.3}$$

Matematiksel modelin formülü yukarıdaki gibidir. Eşitlik (3.1), (3.2) ve (3.3)'te kullanılan değişkenlerde *UB* arama alanında üst sınırını, *LB* arama alanında alt sınırını, *z* mantarın en iyi kaynak arama olasılık sabitini, $\vec{X}(t)$ mantarın konumunu (t aramasındaki), $\vec{X}_A(t)$ rastgele belirlenen parçacık konum vektörünü, $\vec{X}_B(t)$ rastgele belirlenen parçacık konumu vektörünü, $r_{1,2,3}$ rastgele oluşturulmuş 0 ile 1 arasında oluşturulmuş sayıları, *W* cıvık mantarın ağırlığını, p parametre değerini, $\overrightarrow{vb}$ [a , -a] arasında değişkeni, $\overrightarrow{vc}$ 1 den 0'a arasında lineer vektörünü, t iterasyon sayısını, *max_t* maksimum iterasyon sayısını, *S(i)* popülasyonun ilk %50'lik kısmı, *DF* algoritma sonrasında elde edilen en iyi sonucu (konum) ifade etmektedir [21].

Cıvık mantar optimizasyon algoritması, keşif ve işbirliği yeteneklerini kullanarak potansiyel çözüm alanında etkili bir arama yapabilir. Bu algoritma, doğadan ilham alarak çeşitli optimizasyon problemlerini çözmek için kullanılabilir, özellikle karmaşık ve çok-boyutlu alanlarda etkili sonuçlar elde edebilir.

$$\overrightarrow{W}\left(koku_{indeksi(i)}\right) = \begin{Bmatrix} 1 + r.\log\left(\frac{bF - S(i)}{bF - wF} + 1\right) \\ 1 - r.\log\left(\frac{bF - S(i)}{bF - w} + 1\right) \end{Bmatrix} \tag{3.4}$$

Koku_indeksi = sırala (s)

Bu kısımda *r* mantarın kılcallarının arama belirsizliğini benzeştirir. Besin yoğunluğu fazlalaştığında orantılı olarak ağırlık büyür aksi durumda ise ağırlık küçülür. Küçülme durumunda yeni kaynak arayışına geçer. Burada *bF* en iyi amaç fonksiyonu, *wF* en kötüyü gösterir [19].

SMA metodunda ilk olarak popülasyon sayısı, iterasyon sayısı ve konuma başlangıç değerlerini atar. Her bir arama ajanının uygunluk değerini hesaplar ve buna göre en iyi arama ajanı konumunu belirler. Daha iyi bir arama ajanını bulduğunda en iyi değeri buna göre günceller. İterasyon sayısı tamamlanana kadar yada herhangi bir sonlandırma kısıtı gelene kadar döngü devam eder. Döngünün sonlanması ile en uygun değer elde edilmiş olur [21].

3.3.Parçacık Sürü Optimizasyonu (PSO)

Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO), doğadaki bir sürünün (kuş, balık v.b.) sosyal davranışlarından yola çıkarak önerilmiş bir optimizasyon algoritmasıdır. PSO, bir problemin en iyi çözümünü bulmak için parçacıkların (sürüdeki kuşları yada balıkları temsil eder) zekasını kullanarak belirli bir arama uzayında hareket etmesini ve sürü bireylerinin birbirleriyle bilgi paylaşmasını sağlar [22].

PSO'nun algoritması aşağıdaki adımlarla açıklanabilir:

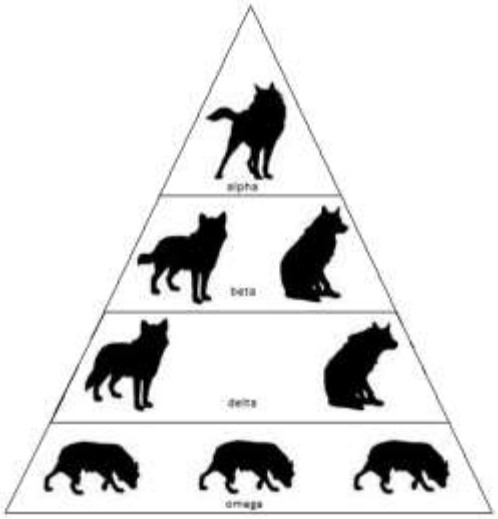
Optimizasyon yapılacak probleme uygun bir uygunluk fonksiyonu belirlenmesinin ardından başlangıçta, belirli sayıda parçacık (particle) oluşturulur. Her parçacık, potansiyel çözümleri temsil eden bir konumu ve hızı (velocity) sahiptir. Sonrasında her parçacık, hızı ve konumu güncellemek için belli bir formülü kullanır. Bu hız güncellemesi, parçacığın kendi en iyi konumuna ve sürüdeki en iyi konuma doğru hareket etme eğilimini içerir. Her bireyin yeni bir potansiyel çözümü temsil eden konuma doğru hareket etmesiyle konumlar güncellenir. Her parçacık kendisinin (pbest) ve sürünün en iyi (gbest) konumlarını takip eder. En iyi konum güncellemesi, parçacığın daha iyi çözüm bölgelerine doğru yönlendirilmesini sağlar. Konum ve hız araması algoritma sonlanana kadar devam eder. Algoritma ise maksimum iterasyon sayısına ulaşınca veya herhangi bir sonlama kriteri varsa ona ulaşınca çözümü oluşturur. Optimizasyon tamamlandığında en uygun çözüm bulunmuş olur. PSO'nun en önemli avantajlarından birisi de parçacıklarının bilgi paylaşımları yapmalarıdır. Zor problemlerde rahatlıkla kullanılabilen hızlı ve kolay bir sonuç ortaya çıkaran bir optimizasyon yöntemidir.

3.4.Gri Kurt Algoritması (GWO)

Kurtlar besin zincirinde en üst kademelerde yer almaktadırlar. Vahşi doğada kurtların kendilerine ait avlanma ve hayatta kalma mücadeleleri, alışkanlıkları ve sitilleri vardır. Gri kurt algoritması da bu mücadeleden faydalanarak oluşturulmuş bir meta-sezgisel algoritmadır. Kurtlar 10 üyeyi geçmeyecek hareket ederler ve dört gri kurt grubu bulunur. Gri kurtlar, liderlik hiyerarşisini taklit eden bir optimizasyon algoritması olan Gri Kurt Optimizasyonu (Grey Wolf Optimization - GWO) kullanılarak modellemektedir [23].

Gri kurt hiyerarşisi Şekil 3.4'de gösterilmektedir. Şekilden de anlaşılacağı üzere hiyerarşinin en tepesinde alfa sonra sırasıyla beta, delta ve omega olmak üzere piramitte dört kurt yer alır [24].

Şekil 3.4. *Gri Kurtların Hiyerarşi Piramidi*



Kaynak: L. Doğan, U. Yüzgeç (11-13 Mayıs 2018), *Robot Path Planning using Gray Wolf Optimizer in Proceedings* - International Conference on Advanced Technologies Computer Engineering and Science (ICATCES'18), 2018.

Gri kurtlar, avlanma davranışını simüle etmek için av arama, avı kuşatma ve ava saldırma aşamalarını yerine getirirler. Alfa kurt koordinatörlüğünde, gri kurtlar avın yerini tespit eder ve avın etrafını sararlar. Ancak GWO'da omega kurtlar hiyerarşinin en altındadır ve en son beslenecek gruptur. Bu nedenle, GWO algoritması hiyerarşiye göre omega haricinde diğer üç grubun avın tahmini yeri hususunda kanaate sahip olduğunu kabul eder ve üç çözümü en iyi olarak raporlar. Kurtların konumu, bu çözümlere göre tespit edilir ve lider kurtları izleyerek hareket ederler.

GWO algoritması liderlik hiyerarşisine dayalı bir arama algoritması ve optimizasyonu oluşturur. Kurtlar arasındaki iletişim ve birlikte hareket etme stratejileri, muhtemel çözümleri keşfetmelerini ve optimize etmelerini sağlar. Bu şekilde, algoritma problemlerin optimize edilmesi için etkili bir yol sunar.

GWO' nun algoritması aşağıdaki adımlarla açıklanabilir:

Optimize edilecek problemle ilgili fonksiyon tespiti sonrasında gri kurt kaç tane ise her biri muhtemel çözümü simgeler. İkinci adımda her gri kurt konumunu ve konumunun uygunluğunu kontrol eder, en iyi konumda olan gri kurt alfa kurt olarak yani lider seçilmiştir. Sonrasında lider kurt hareketine göre diğer kurtlar pozisyon alırlar. Bu işlemi icra ederken bir

yandan da yeni çözüm alanları araştırırlar (keşif) ve alfa kurt çevresinde çoğalarak patlama en iyi çözüm optimizasyonu yapmaya çalışırlar. Sürekli kurtlar alfa kurdun(en iyi çözümün) konumunda göre konumlarını yenilerler. Bu konum seçme işlemi en iyi çözüm bulunana kadar devam eder. İterasyon sayısına ulaşınca yada sonlandırma kriterleri yerine geldiğinde optimizasyon tamamlanır.

3.5.Ağaç-Tohum Algoritması (TSA)

TSA algoritması doğadan ilham alan zeki optimizasyon tekniklerindendir. Bu algoritma çözüm için doğadaki ağaçlar ile tohumlar arasındaki ilişkilerden faydalanır. Bu ilişkiyi taklit ederek problemin optimizasyonunu amaçlar. Ağaçlar tohumlama yöntemiyle yani tohumların doğaya yayılması ile çoğalırlar. Tohumların çoğalması ağaçların konumlarına göre gerçekleştirilir [25].

$$Ti,j = Lj,min + rand(0,1)(Hj,max - Lj,min) \quad (3.4)$$

Denklem(3.4) konum gerçekleştirme denklemidir. Burada Ti,j ağacın konumu, Hj,max arama alanı üst limit değerini, Lj,min arama alanı alt limit değerini gösterir. Tohum meydana gelmesi esnasında arama eğilimi Denklem(3.5) ve Denklem(3.6) ile hesaplanır.

$$Si,j = Ti,j + rand(-1,1) \times (Bj - Tr,j) \quad (3.5)$$

$$Si,j = Ti,j + rand(-1,1) \times (Ti,j - Tr,j) \quad (3.6)$$

Bu denklemde ise Si,j *i.* tohumun *j.* boyutu, Ti,j *i.* ağacın *j.* boyutu ve Bj meydana gelen en iyi ağacın *j.* boyutudur.

TSA' nın algoritması aşağıdaki adımlarla açıklanabilir:

Optimize edilecek problemle ilgili fonksiyon tespiti sonrasında belirli bir adet tohum meydana getirilir ve her bir tohum bir muhtemel bir çözümü simgeler. Bir sonraki adımda her tohum bir ağacın korumasındadır. Her tohum diğer tohumlarla iletişim halindedir. Bu etkileşim ile tohumlar yerlerini güncelleyerek yeni çözüm arayışlarını devam ettirmiş olurlar. Bu tohum yayılma taktiği ile farklı alanlara yayılarak çözüm uzayını genişletirler. Tohumların her biri ağaçların koruması altında kendi konumlarını yeniler. Ağaçların verdiği bilgiler ışığında tohumlar daha iyi çözümü bulur. Besin kaynaklarının yakınında konumlanır. Bu en iyi konum bulma çalışması algoritma sonlanana kadar devam eder [25].

3.6. Salp-Sürü Algoritması (SSA)

Salp denizde yaşayan bir canlı türüdür. Okyanusun diplerinde sürü şeklinde davranış gösterirler. Salp zinciri şeklinde birlikte hareket ederler. Jöle balığına benzerliği ile dikkat çekmektedirler. Salp zincirinde önde giden salp lider olarak tanımlanır. Zincirin geri kalan kısmındaki Salplar ise lideri takip eder. Bu zincir hareketleri yiyecek aramak için kullanılır [26].

SSA' nın algoritması aşağıdaki adımlarla açıklanabilir:

Optimize edilecek problemle ilgili fonksiyon tespiti sonrasında belirli bir adet salp meydana getirilir ve her bir tohum bir muhtemel bir çözümü simgeler. Salplara kendine özgü konum ve hız bilgilerine sahiptir. Salplara başlangıç hız ve konumları verilir. Salpların konumları ve uygunluk değerleri onların ne kadar iyi beslendiğinin göstergesidir. Beslenme durumlarına göre konumlarını güncellerler. Salp'lar hep en iyi beslenme konumuna doğru hareket etmeye çalışırlar ve bunu beslenme ve hızları ile yaparlar. Her konum güncellemesi muhtemel bir çözümdür ve optimizasyon sonlandırılana kadar bu hareketlerine devam ederler. Adımlar halinde anlatacak olursak;

İlk olarak başlangıç popülasyonunun oluşturulur. Yani SSA, bir başlangıç popülasyonu oluşturarak başlar ve popülasyon, belirli bir problemin çözüm uzayında rastgele dağıtılan salplardan oluşur. Her salp, parametrelerle temsil edilen bir çözüm adayını temsil eder.

İkinci adımda hedef fonksiyonun değerinin hesaplanmasıdır. Her salp, mevcut konumundaki parametre değerleri kullanılarak hedef fonksiyonun değerini hesaplar. Hedef fonksiyon, optimizasyon probleminin amaçlandığı kriteri temsil eder ve salpların hedef fonksiyon değerleri, çözümlerin kalitesini yansıtır.

Üçüncü adımda en iyi çözümün belirlenmesidir. Salplar arasında en iyi çözüm adayını belirlemek için hedef fonksiyon değerleri karşılaştırılır. En iyi çözüm, hedef fonksiyonu en iyi değeri veren çözüm adayıdır ve bu çözüm adayının değerleri tüm salplar arasında paylaşılır ve güncel tutulur.

Dördüncü adımda salpların hareketinin hesaplanmasıdır. Salpların hareketini hesaplamak için yüzme ve çekim kuvvetleri kullanılır. Yüzme kuvveti, salpların mevcut hızlarına ve en iyi çözüm adaylarına bağlı olarak belirlenir. Salplar, yüzme kuvvetini kullanarak daha iyi çözümler aramak için hareket ederler. Çekim kuvveti ise salpların en iyi çözümlerin etrafında yoğunlaşmasını sağlar ve birbirlerine yaklaşmalarını teşvik eder.

Beşinci adımda yeni konumların hesaplanmasıdır. Salpların yeni konumları, mevcut konumlarına ve hesaplanan hareket yönlerine bağlı olarak hesaplanır. Hareket yönleri, yüzme ve çekim kuvvetleri kullanılarak belirlenen yönlere ve mesafelere göre belirlenir. Yeni konumlar, mevcut konumlar ve hareket yönleriyle belirlenir. Eğer salplar çözüm uzayı sınırları dışına çıkarsa konum tekrar düzenlenir.

Altıncı adım ile yeni hedef fonksiyonun değerlerinin hesaplanması ve yeni konum performansının hesaplanarak hedef fonksiyon değeri güncellenir.

Son adım ise önceden belirlenmiş bir iterasyon sayısı, hedef fonksiyonun belirli bir değeri aşması veya bir zaman sınırı gibi bir durdurma kriterleri var ise bu şartlar sağlanıncaya kadar döngü devam eder. Şart sağlanması durumunda algoritma en iyi çözüm adayı en iyi hedef fonksiyonuna sahip Salp olarak belirlenir [26].

3.7.Harris Hawks Algoritması (HHO)

Harris bir şahin türüdür. Bu algoritma da bu kuşun doğada avlanma taktiklerinden ilham alarak geliştirilmiş bir meta-sezgisel bir algoritmadır. Harris şahinleri bir araya gelir ve işbirliği içerisinde birlikte avlanırlar. Bu algoritma popülasyon tabanlıdır. İki, üç şahin bir avı tespit eder ve avın kaçış yönlerine farklı yönlerden saldırarak kısa zamanda avlarını elde etmeye çalışırlar. Sürü zekası ile çalışan bir yapıya sahiptir [27].

HHO'nın çalışma prensibi aşağıdaki adımları içerir:

İlk olarak HHO, bir başlangıç popülasyonu oluşturarak başlar. Popülasyon, çözüm adaylarını temsil eden bir dizi bireyden oluşur. Her birey, belirli bir problemin çözüm uzayında bir noktayı temsil eden parametrelerle tanımlanır.

İkinci adımda hedef fonksiyonun değeri hesaplanır. Yani her birey, mevcut değişken değerleri kullanılarak hedef fonksiyonun, optimizasyonun amaç kriteri, değerini hesaplar. Bireylerin hedef fonksiyon değerleri, çözümlerin kalitesini yansıtır.

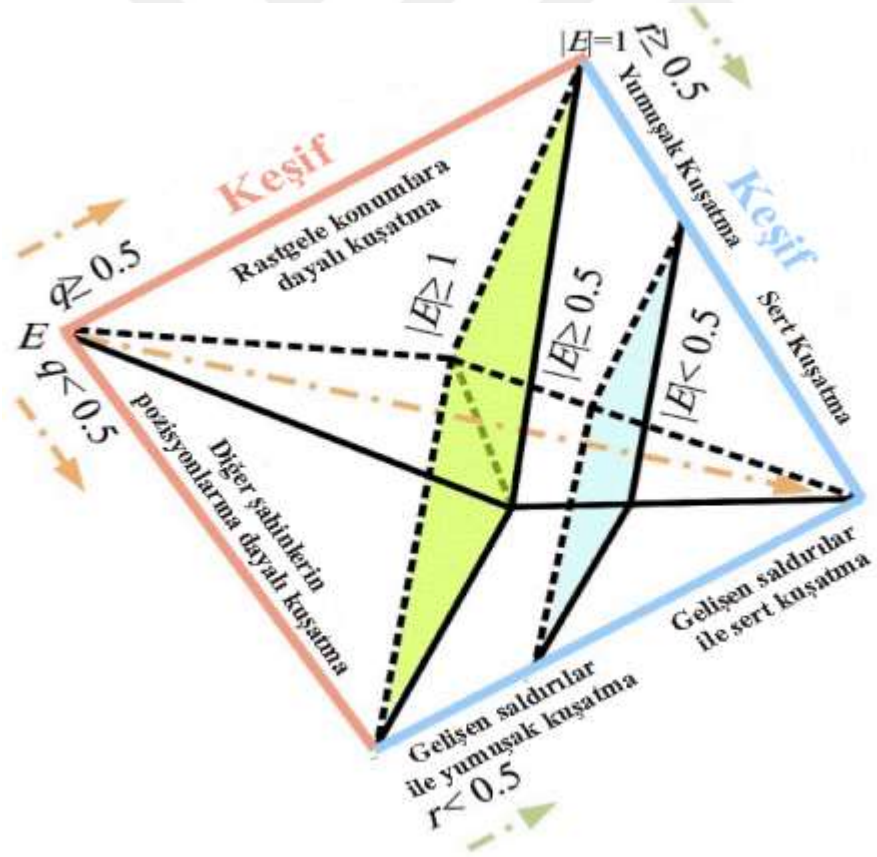
Üçüncü adımda en iyi bireyler belirlenir. Bireyler arasında en iyi çözüm adaylarını belirlemek için hedef fonksiyon değerleri karşılaştırılır. Burada hedef fonksiyonunu en iyi değeri veren bireyler çözüm adayıdır. Bu çözümler, birbirleriyle iletişim halinde olacak olan Harris şahinlerini temsil eder ve hedefe ulaşmaya en yakın çözüm adayı hedef fonksiyon değeri en iyi çıkan adaydır. Harris şahinleri işbirliği yaparlar ve iletişimleri süreklidir. Her iterasyonda da bu avlanmadaki iletişim gibi çözüm adayları birlikte hareket eder ve yeni konum belirlerler. Şahinler burada avlanma stratejilerini kullanırlar. Burada takip, sürekli iletişim, keşif ve

saldırı gibi yöntemlerle avı yakalamak için en iyi konumu bulmaya çalışırlar. Her şahinin hareketi diğer şahinleri de etkiler.

Dördüncü adım yeni konumlar belirlemedir. Harris şahinleri aralarındaki iletişim sayesinde en iyi konumdaki şahinin etrafında yoğunlaşırlar. En iyi çözüm adayı en iyi konumdaki şahindir. HHO da bunu taklit eder ve sürekli konum güncelleme devam eder.

Son adımda ise önceden belirlenmiş bir iterasyon sayısı, hedef fonksiyonun belirli bir değeri aşması veya bir zaman sınırı gibi bir durdurma kriterleri var ise bu şartlar sağlanıncaya kadar döngü devam eder. Şart sağlanması durumunda algoritma en iyi çözüm adayı en iyi hedef fonksiyonuna sahip, hedef fonksiyonun en iyi değerini veren şahin tarafından temsil edilir. HHO algoritmasının kademeleri Şekil 3.5'de gösterilmektedir [27].

Şekil 3.5. *HHO Algoritması Kademeleri*

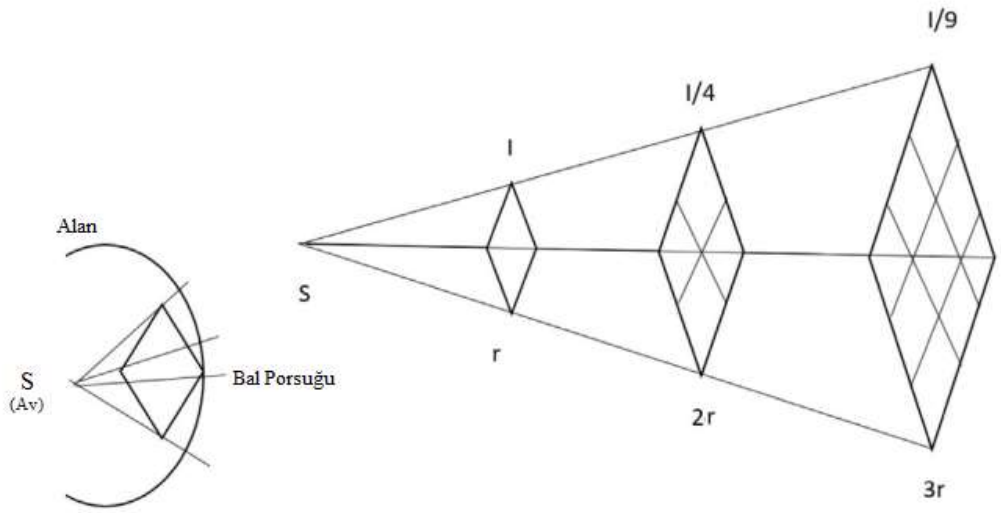


Kaynak: A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, "Harris hawks optimization: Algorithm and applications", *Future Generation Computer Systems*, 97, 849–872, 2019.

3.8.Bal Porsuğu Algoritması (HBA)

Benzer algoritmalarda olduğu gibi bu algoritma da bal porsuklarının doğada besin arama stratejilerinden esinlenerek oluşturulmuş meta-sezgisel bir algoritmadır. Bu algoritmada bal porsuğunun balı araması, kazması ve bulmasının taklit edilmesi üzerine kurulmuştur. Bal porsukları av ararken koku alma yetisini kullanır ya da kılavuz kuşu takip eder. Bal porsuğu avı ararken avın kokusu fazla ise porsuğun avı bulması hızlı olacaktır. Ters kare yasasına göre Şekil 3.6'da görüldüğü gibi uzaklaşıldıkça karesel olarak koku yoğunluğu azalacaktır.Şekli incelediğimizde *I* koku yoğunluğu, *S* avın konumu ve *r* ise 0 ile 1 arasında rastgele bir sayıdır [28].

Şekil 3.6. *Ters Kare Yasası*



Kaynak: F.A.Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, W. Al-Atabany, "Honey badger algorithm: New metaheuristic algorithm for solving optimization problems", *Math. Comput. Simul.* 192, 84–110. doi:10.1016/j.matcom.2021.08.013, 2022.

Her bal porsuğu pozisyonunun uygunluğunu değerlendirir. Kazı aşamasına geçtiğinde konum değiştirebilir. Bal porsuğu algoritmasında iki kademe vardır. İlk kademe "kazma" yani avı arama kademesidir. Bal porsuğunun oku alma yetisini kullanarak avı bulur. Sonrasında "bal evresi" avını bulduktan sonra avın etrafında yakalamak için tam yerinin tahminleme evresidir.

HBA'nın çalışma şeklini yani akışını özetleyecek olursak;

- Optimizasyon yapılacak probleme uygun bir model ve uygunluk fonksiyonu belirlenir. Başlangıç için belirli sayıda bal porsuğu popülasyonu oluşturulur. Her bal porsuğu bir çözümü temsil eder.
- Bal porsukları rasgele konumlarda başlar. Bal porsukları mevcut konumlarından hareket ederek potansiyel çözüm alanında dolaşır. Hareket ederken koku alma fonksiyonlarını ve bal kılavuz kuşu kullanır ve her zaman en iyi çözüm bölgesini bulmaya eğilimlidir.
- Her bal porsuğu kendi konumunu ve hedefini sürekli günceller. Uygunluk fonksiyonuna göre en iyi çözümü ulaşmaya çalışır. Yeni konumlar her zaman daha iyi çözümler için bir potansiyeldir.
- Arama ve konum güncellemesi iterasyon sayısına ya da sonlandırma kriterlerine ulaşana kadar devam eder. Her iterasyonda en iyi çözüm arayışı devam eder [28].

DÖRDÜNCÜ BÖLÜM

Araştırma Ve Bulgular

4.1.Giriş

Bu tez çalışmasında Amerikan Yöntemi uygulanmıştır. Kavşağı belirli zaman dilimlerinde kullanan araç verileri toplanmıştır. Kavşak gecikmesi hesaplanabilmesi için önce kavşağın akım hacmi, sonrasında kapasitesi, devamında Hacim/ Kapsite Oranı, doygun akım hesaplanmıştır. Doygun akımın hesabında birçok faktör göz önünde bulundurulur. Yukarıdaki hesapları yapan ve bu hesaplar ışığında kavşaktaki uniform gecikmeyi ve kontrol gecikmesini hesaplayan bu gecikmelerin birleşimiyle akım kolu gecikmesi ve kavşak gecikmesini hesaplayan bir matematiksel model geliştirilmiştir. Bu modellemede MATLAB yazılımı kullanılmıştır. Yazılımda farklı meta-sezgisel algoritmalarla optimizasyon teknikleri kullanarak yeşil süreleri optimize ederek gecikmeyi minimize edilmiştir. Burada amaç fonksiyonumuz kavşaktaki gecikme olarak belirlenmiştir.

Modellenecek kavşak olarak ilk etapta tek kavşak ile çalışılmış daha sonrasında optimizasyon yazılımlarının birçoğu aynı değerleri sağlıklı bir şekilde verince problem zorlaştırılarak iki kavşak birlikte eş zamanlı çalıştırılarak modellenmiştir.

4.2.Tek Kavşak Üzerinde Analiz ve Ölçümler

Kavşak olarak ile etapta Metro Market yakınlarındaki kavşak modellenmiştir. Hafta içi yoğun günlerde araç sayım değerleri alınmıştır. Seçtiğimiz bu kavşak Kayseri'de eski çevreyol üzerinde Organize Sanayi bölgesi girişi için kullanılan yoğun bir kavşaktır. Şekil 4.1'de Metro Market kavşağının genel görünümü verilmiştir.

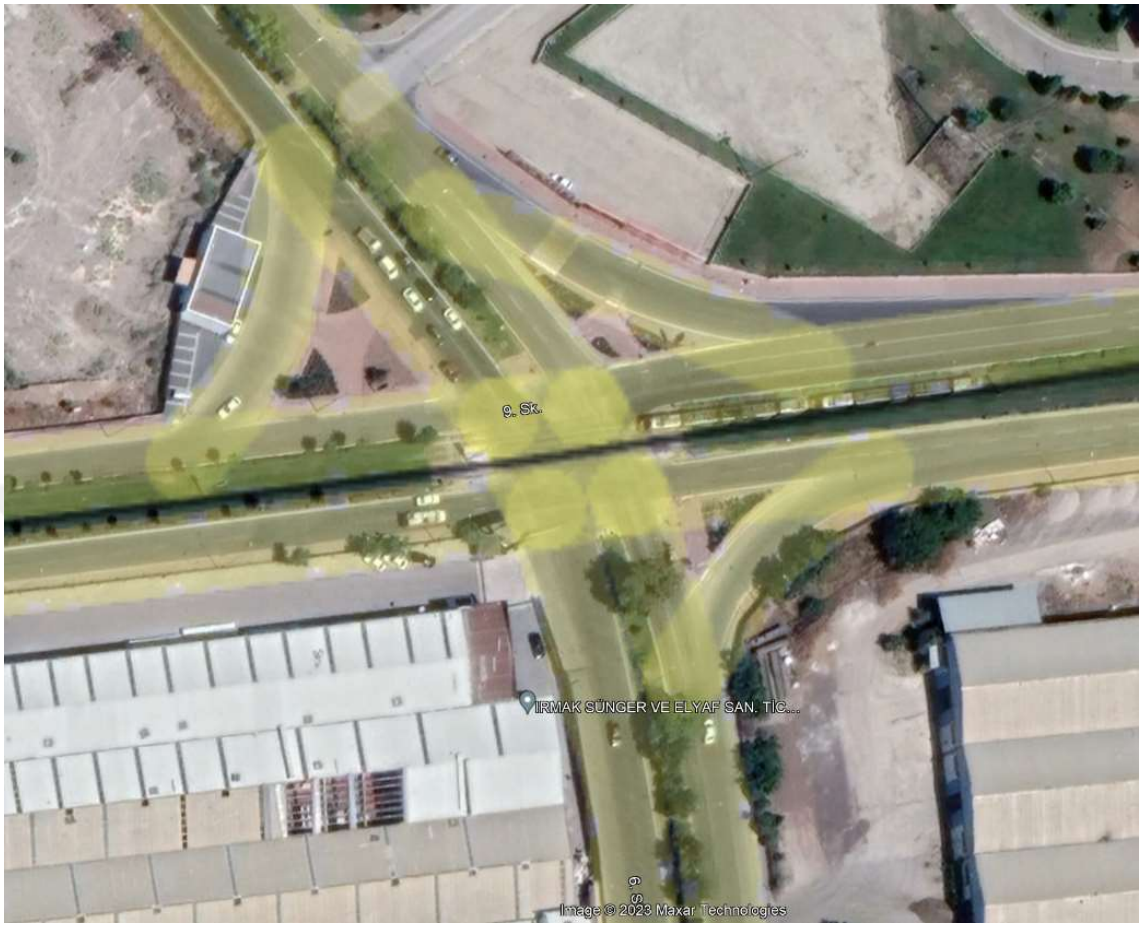
Şekil 4.1. *Metro Kavşak Google Earth Görüntüsü*



Kavşak MATLAB üzerinde optimize edilirken kavşağın en yoğun olduğu akşam pik saat (17:30 – 18:30) araç sayımlarına göre optimize edilmiştir. Faz 1 yönünde 1650 araç/saat, Faz2 yönünde 1364 araç/saat ve Faz3 yönünde 1256 araç/saat sayılmıştır. SMA, PSO, GA, GWO ve TSA algoritmaları kullanılarak optimizasyon yapılmıştır. Optimizasyon sonuçları kavşağın 3 kollu olması ve problemin basit olması nedeniyle bütün algoritmalarda çok yakın çıkmıştır. Bu sonuçların sonrasında optimizasyon için çalışılan kavşak değiştirilmeye karar verilmiştir.

Sonraki süreçte dört kollu Organize Kavşağı üzerinde çalışma yapılmıştır. Kameralar vasıtasıyla araç sayımları yapılmıştır. Dört kollu ve ayrıca tramvay geçişleri olan özellikle iş giriş ve çıkış saatlerinde çok yoğun bir kavşaktır. Şekil 4.2'de Organize Kavşağı genel görünümü verilmiştir.

Şekil 4.2. *Organize Kavşak Google Earth Görüntüsü*



Kavşak doyum akım değerleri hesaplanırken kavşağın geometrik özellikleri de dikkate alınmaktadır.

Tablo 4.1. *Organize Kavşak Şerit Sayı Ve Genişlikleri*

Organize tarafı çıkış şerit sayısı (1 gruplar)	1	
Organize tarafı çıkış şerit genişliği	3,25	metre
Organize tarafı çıkış tek şerit genişliği	3,25	metre
Şehir merkezi geliş şerit sayısı (2 gruplar)	1	
Şehir merkezi geliş şerit genişliği	3,17	metre
Şehir merkezi geliş tek şerit genişliği	3,17	metre
Çevreyol geliş yolu geliş şerit sayısı (3 grup)	2	
Çevreyol geliş şerit genişliği	7,5	metre
Çevreyol geliş tek şerit genişliği	3,75	metre
Org. Cami geliş yolu geliş şerit sayısı (4 grup)	2	
Org. Cami geliş şerit genişliği	7,8	metre
Org. Cami geliş tek şerit genişliği	3,9	metre
Organize tarafı çıkış şerit sayısı (5 gruplar)	1	
Organize tarafı çıkış şerit genişliği	3,25	metre
Organize tarafı çıkış tek şerit genişliği	3,25	metre
Şehir merkezi geliş şerit sayısı (6 gruplar)	2	
Şehir merkezi geliş şerit genişliği	6,34	metre
Şehir merkezi geliş tek şerit genişliği	3,17	metre

Organize Kavşağın şerit sayıları ve şerit genişlikleri Tablo 4.1'de gösterilmektedir. Akım kollarındaki akım yükü hesaplanırken bütün araç türleri birim otomobil eşdeğer katsayıları ile çarpılarak birim otomobil türüne dönüştürülür. Tablo 4.2'de taşıtların birim otomobil eşdeğerleri gösterilmektedir [27].

Tablo 4.2. *TS6407'e Göre Birim Otomobil Eşdeğer Katsayıları*

Araç Cinsi	Şehir İçi Yollarda	Dönel Kavşaklarda	Işıklı Kontrollü Kavşaklarda
Özel otomobil, taksi, sepetli motosiklet, triptör, yüksüz 1500kg'a kadar olan pikap	1,00	1,00	1,00
Minibüs, Dolmuş, Taksi	1,15	1,30	1,27
Yüksüz 1500kg'a kadar ağır ticari kamyon, at arabası	2,00	2,80	1,75
Kent içi ve dışı yolcu otobüsü (körüklü dahil), servis otobüsü, troleybus, tramvay	3,00	2,80	2,25
Motosiklet, mopet	0,75	0,75	0,33
Bisiklet	0,33	0,50	0,20

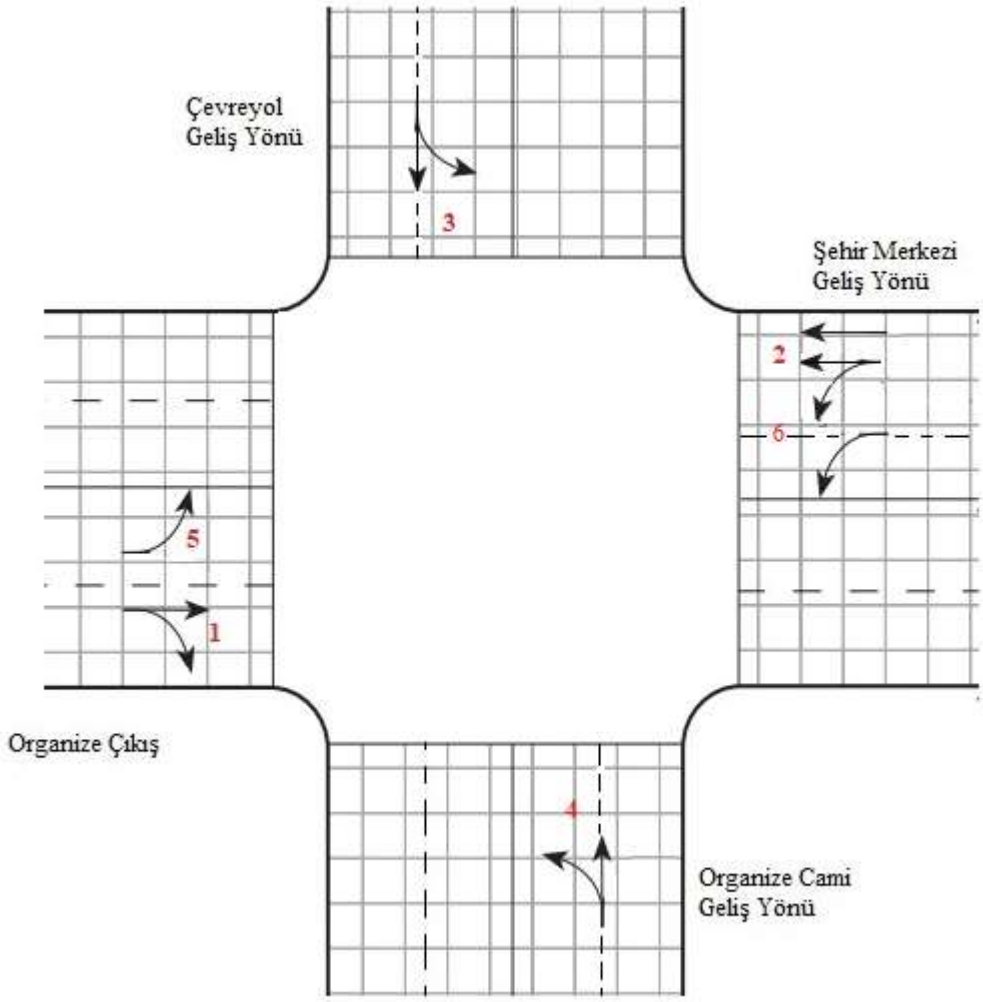
Kaynak: A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, "Harris hawks optimization: Algorithm and applications", *Future Generation Computer Systems*, 97, 849–872, 2019.

Tablo 4.3. *Organize Kavşak Araç Sayım Değerleri*

Grup No:			7.30/7.45	7.45/8.00	8.00/8.15	8.15/8.30	17.45/18.00	18.00/18.15	18.30/18.45	18.45/19.00
	1. Yol Düz	Servis	5	13	9	14	31	47	26	23
		Otobüs	0	1	1	0	1	0	0	1
		Araç	15	15	18	10	51	86	55	35
1		Kamyon	0	0	0	0	0	5	2	0
	1. Yol Sağ	Servis	0	2	1	1	0	21	0	1
		Otobüs	0	0	0	0	0	0	0	1
		Araç	3	0	2	2	0	52	2	0
1		Kamyon	0	0	0	0	0	3	0	0
	1. Yol Sol	Servis	0	6	2	3	1	13	0	2
		Otobüs	0	0	0	0	0	0	0	0
		Araç	3	2	5	2	9	16	12	7
5		Kamyon	0	2	0	0	0	0	2	1
	2. Yol Düz	Servis	79	30	20	5	10	4	0	1
		Otobüs	0	2	3	1	1	2	0	2
		Araç	43	102	64	84	23	10	12	10
2		Kamyon	0	3	3	1	0	0	1	1
	2. Yol Sağ	Servis	121	63	41	29	10		2	3
		Otobüs	0	0	0	0	0	0	0	0
		Araç	26	96	120	127	48		5	7
0		Kamyon	8	10	19	10	1		1	2
	2. Yol Sol	Servis	134	30	57	32	18	2	8	4
		Otobüs	0	0	0	0	0	0	0	0
		Araç	64	220	141	110	25	30	23	28
6		Kamyon	4	35	7	4	0	4	0	0
	3. Yol Düz	Servis	40	8	14	2	28	5	4	0
		Otobüs	0	0	0	0	0	0	0	1
		Araç	23	82	49	43	47	29	29	25
3		Kamyon	15	12	5	10	3	10	2	6
	3. Yol Sol	Servis	3	2	5	4	7	5	12	5
		Otobüs	0	0	0	0	0	0	0	0
		Araç	7	7	12	7	21	69	68	51
3		Kamyon	0	0	0	0	2	8	2	0
	4. Yol Düz	Servis	10	19	4	15	12	9	14	7
		Otobüs	0	0	0	0	0	0	0	0
		Araç	13	26	31	31	50	46	32	33
4		Kamyon	0	4	3	3	12	11	11	6
	4. Yol Sağ	Servis	22	37	24	40	62	53	103	46
		Otobüs	0	0	0	0	0	0	0	0
		Araç	15	21	40	35	120	101	165	83
0		Kamyon	0	0	2	3	2	0	2	2
	4. Yol Sol	Servis	1	0	0	2	0	5	0	0
		Otobüs	0	0	0	0	0	0	0	0
		Araç	1	1	4	1	0	9	3	1
4		Kamyon	0	1	0	0	0	0	1	0

Organize Kavşaktaki sabah ve akşam pik saatlerdeki araç sayı değerleri Tablo 4.3'de gösterilmektedir. Sayım değerlerinden de anlaşılacağı üzere en yoğun saatler sabah işe gidiş saatleridir. (7.30 – 8.30) Bizde optimizasyonu en yoğun saate göre yapılmasında karar kıldık. Şekil 4.3'de kavşakların her bir akım kollarına numara verilmiş bunlarda grup olarak isimlendirilmiştir. Hangi grubun hangi yöne gidecek araçları ve akım kollarını temsil ettiği aşağıdaki gibidir.

Şekil 4.3. *Organize Kavşak Sinyal Krokisi*



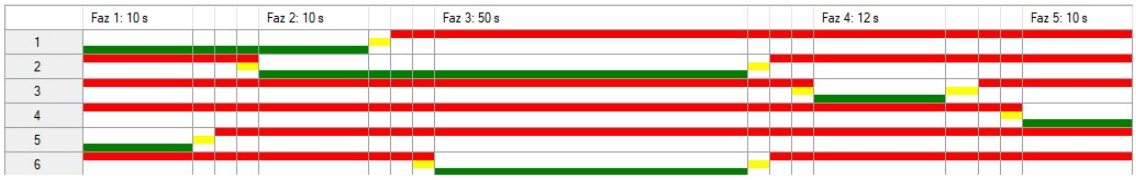
Mevcutta kavşak sayım değerlerinin birim otomobil cinsine dönüştürülerek elde edilen değerler Tablo 4.4'de gösterilmiştir. Burada kavşakta sabah yoğunluğunu en fazla olduğu gruplar 2, 6 ve 3 numaralı gruplardır.

Tablo 4.4. *Organize Kavşak 7.30-8.30 Arası Birim Otomobil Cinsinden Araç Sayım Değerleri*

	Organize Çıkış Yönü			Şehir Merkezi Geliş Yönü		Çevreyol Geliş Yönü		Org. Cami Geliş Yönü	
Grup Numarası	**1**	**1**	**5**	**2**	**6**	**3**	**3**	**4**	**4**
Hareket Yönü	**Sağa Dönüş**	**Doğru Gidiş**	**Sola Dönüş**	**Doğru Gidiş**	**Sola Dönüş**	**Doğru Gidiş**	**Sola Dönüş**	**Doğru Gidiş**	**Sola Dönüş**
Araç Sayıları	12	115	30	492	969	373	51	184	13
Toplam Araç Sayıları	127		30	492	969	424		197	

Organize kavşak mevcutta çalışma sinyal planı Şekil 4.4'deki gibidir. Burada ortak çalışabilecek gruplar birlikte aynı anda yeşil yanacak şekilde ayarlanmıştır. Faz 1 de 1 ile 5'nolu grup, Faz 2 de 1 ile 2'nolu grup, Faz 3 de 2 ile 5'nolu grup, Faz 4 de 3'nolu grup ve Faz 5 de 4'nolu grup çalıştırılmaktadır. Kavşağa tramvay gelmesi durumundaki etkisini inceleyecek olursa sayımlarda 1 saatlik dilim içerisinde 12 tramvay gelmiş ve kavşağı terk etme süresi ortalama 13,8s olarak ölçülmüştür. Önerilen sistemde sinyal planında değişiklik yerine burada uygulanan yeşil sinyal süreleri optimize edilerek gecikme süresini en aza indirmektir.

Şekil 4.4. *Organize Kavşak Mevcut Sinyal Planı*



Mevcut kavşakta grupların geçişi için verilen süreler Grup1 26 s, Grup2 66s, Grup 3 12s, Grup 4 10s, Grup 5 10s, Grup 6 50s şeklindedir.

4.3.Tek Kavşak Üzerinde Optimizasyon

Kavşakta optimizasyon yapılırken her bir grubun yeşil süre ihtiyacı ayrı ayrı hesaplanmış, birlikte çalışması düşünülen gruplardan kritik olan (daha uzun yeşil süreye ihtiyacı olan grup) tercih edilir. Bu şekilde hesap yapılması durumunda fazlar içerisindeki grup seçimleri de daha sağlıklı olacaktır.

Doyum akım oranı hesaplanırken temel doygun akım 1900 araç/saat alınmıştır. Ağır araç oranı etkisinin hesaplanırken ağır araç oranı %25 olarak alınmıştır. Sağa dönen araç etkisi olmadığı için 1, sola dönen araçların etkileri de kavşaktaki şerit durumuna göre hesaplanmıştır. Faz geçişlerinde 6s kayıp süre oluşmaktadır. 4 faz geçişi olması nedeniyle toplam kayıp süre L = 24s olarak alınmıştır. Yaya geçişleri için optimizasyon da ihmal edilmiş, minimum süre olarak yayaların kavşaktan geçecekleri süre dikkate alınarak her bir grubun 10s altına düşmesine izin verilmemiştir.

Burada Amerikan yöntemine göre hesaplama yapıldığında kavşak gecikmesi 326 s olarak hesaplanmıştır.

Optimizasyon işlemi sonrasında sonuçlar Tablo 4.5'de verilmiştir.

Tablo 4.5. *Organize Kavşak Optimizasyon Sonuçları*

Organize Kavşak Optimizasyon Sonuçları	SMA	PSO	GA	GWO	TTA
İterasyon sayısı (max)	100	100	100	100	100
Populasyon Size	50	50	50	50	50
Toplam gecikme	205,6	205,6	205,6	205,6	205,7
Grup 1	14,9	14,6	14,8	14,7	14
Grup 2	59	58,4	59,2	58,5	58,3
Grup 3	28,7	28,3	28,75	28,5	27,9
Grup 4	14,9	14,6	14,8	14,7	14,8
Grup 5	10	10	10	10	10
Grup 6	80,52	79,9	81	80	79

Burada görüldüğü üzere kavşaktaki problemin kolay olması nedeniyle optimizasyon programları hiç zorlanmamıştır. Sonuçlar çok yakın çıkmıştır. Yeşil süreleri ve gecikme süreleri arasında da çok az fark vardır. Ancak burada dikkat çeken husus optimizasyonla ortaya çıkan sonuçlarda kavşağın toplam gecikmesinde %36,9'luk bir iyileşme söz konusudur.

Mevcuttaki faz sistemi ile kavşağı çalıştırmak yerine sabah yoğun saatlerde Faz 2'nin kaldırıldığı bir sistemde çalışmanın daha uygun olacağı gözlemlenmiştir.

4.4.Tramvayın Kavşak Sinyalizasyon Sistemine Etkisi

Kavşakta tramvay etkisi analiz edecek olursak tramvay ile birlikte çalışan gruplara baktığımızda 1 ve 2. gruplar olduğunu görürüz. Bu gruplar 6'nolu grup dışında diğer iki grup kavşağın ek yoğun kullanılan gruplarıdır. Yapılan ölçümlerde 1 saatlik dilim içerisinde 12 tramvay gelmiş ve kavşağı terk etme süresi ortalama 13.8 s olarak ölçülmüştür. Kavşakta tramvayların hiçbirisi karşılıklı denk gelmemiştir. Kavşakta tramvay öncelikli sistem çalışmaktadır. Tramvayı algılayan bir talep sistemi ve kavşağı boşalttığını algılayan bir iptal sistemi kavşakta çalışmaktadır. Kavşağa yaklaşırken tramvay otomatik talep vermekte kavşağı boşaltınca da otomatik iptal sistemi ile kavşağın eski haline dönmesi için iptal bilgisi vermektedir. Tramvay geldiği anda talep verince kavşak bulunduğu fazdan Tramvay fazına geçmektedir. Tramvay fazı mevcut döngüde çalışan Faz 2 ile aynı grupları aktif etmektedir. Tramvay geçişi sonrası tramvay fazı ile Faz 2 aynı olması nedeniyle kavşak Faz 2'yi işlendi bilgisini 1 yaparak, Faz 2 yi bu döngü tamamlanana kadar bir daha işlemeyecektir. Tramvay fazı sonrası kavşak döngüye kaldığı yerden devam etmektedir.

Tramvayın kavşağa etkileri;

- Kavşakta Faz 2 çalışırken tramvay gelirse kavşaktaki kayıp az olmaktadır. Tramvay fazı ile Faz 2 aynı olduğu için yeşil uzatma yapılmış olacak ve sadece Faz 2 en fazla tramvay geçiş süresi kadar uzayacaktır. Tramvay eğer Faz 2'nin başında gelmiş ise;

  Kavşaktaki Faz 2 uzaması = Tramvay geçiş süresi –Faz 2 süresi kalan süre kadar olacaktır.

- Kavşakta Faz 2 ‘ ye geçerken tramvay gelirse (tek tramvay için en istenen durum budur) kavşaktaki kayıp en aza inecektir. Faz 2’nin başında gelmiş ise Faz 2 işlenmediği için tramvay fazı işlenecektir. Faz 2 işlenmiş kabul edilecektir.
- Kavşakta başka bir faz çalışırken tramvay gelmesi durumunda kavşak tramvay fazına geçecektir. Faz 2’yi işlemiş olsa dahi tramvay fazı da aynı faz olduğu için bir döngü içerisinde iki kez işlemiş olacaktır. Bu süre de kavşak için bir kayıptır. Bunun yanında tramvayın fazı yarıda kesip kesmemesi de çok önemlidir. Yani örneğin Faz 3 çalışırken tramvay gelmiş ise eğer tramvay talebinde faz işletmesini beklemeden Faz 3 minimum süresi işlendikten sonra direk tramvay fazına geçiş yapılırsa Faz 3 döngü içerisinde tüm süresini kullanamamış olacak ve yoğun bir akım var ise bir sonraki döngüye kadar kuyruklanma oluşabilecektir.
- Karşılıklı tramvay geçişleri aynı ana denk gelmesi veya getirilmesi durumu en ideal bir durumdur. Yukarıda bahsettiğimiz durumlarda karşılıklı tramvay gelmesi demek kayıp etkisinin etkinin ikiye bölünmesi anlamına gelmektedir.

4.5.Çoklu Kavşak Üzerinde Analiz ve Ölçümler

İki kavşağın senkron yönetilmesi ile ilgili çalışma yapılmıştır. Art arda birbirini takip eden iki kavşak birlikte optimize edilmiştir. Burada ana faktör iki kavşağın ana arterlerinde yeşil dalga uygulaması yapılmasıdır. Yeşil dalga uygulaması olan ana arterde art arda kavşaklarda araçların her bir kavşakta durmasını engellemek için yapılan bir sistemdir. Bizde yaptığımız optimizasyon ile yeşil dalgada kavşaklara verilebilecek en uygun yeşil sürelerini ve en az gecikmeyi bulmayı amaçladık.

Burada kavşak olarak tek kavşak olarak modellediğimiz Organize kavşak ile bir öncesinde bulunan Tınaztepe kavşaklarını birlikte optimize edilmiştir. Bu iki kavşakta da tramvay geçmekte olup, modellememizde tramvay öncelik sistemini dahil etmedik. Tramvayı vereceği talepler, kavşaktaki senkronizasyonu bozacaktır. Organize kavşak ile ilgili detay bilgiler yukarıda anlatıldığı için Tınaztepe Kavşağı ile ilgili ölçüm ve analizlerle başlayalım.

Tınaztepe kavşağında kameralar üzerinden araç sayımları yapılmış. Tınaztepe kavşağı dört kollu bir kavşaktır ve Organize kavşağı gibi tramvay geçişleri olan bir kavşaktır. Şekil 4.5’de Tınaztepe kavşağın genel görünümü verilmiştir.

Şekil 4.5. *Tınaztepe Kavşak Google Earth Görüntüsü*

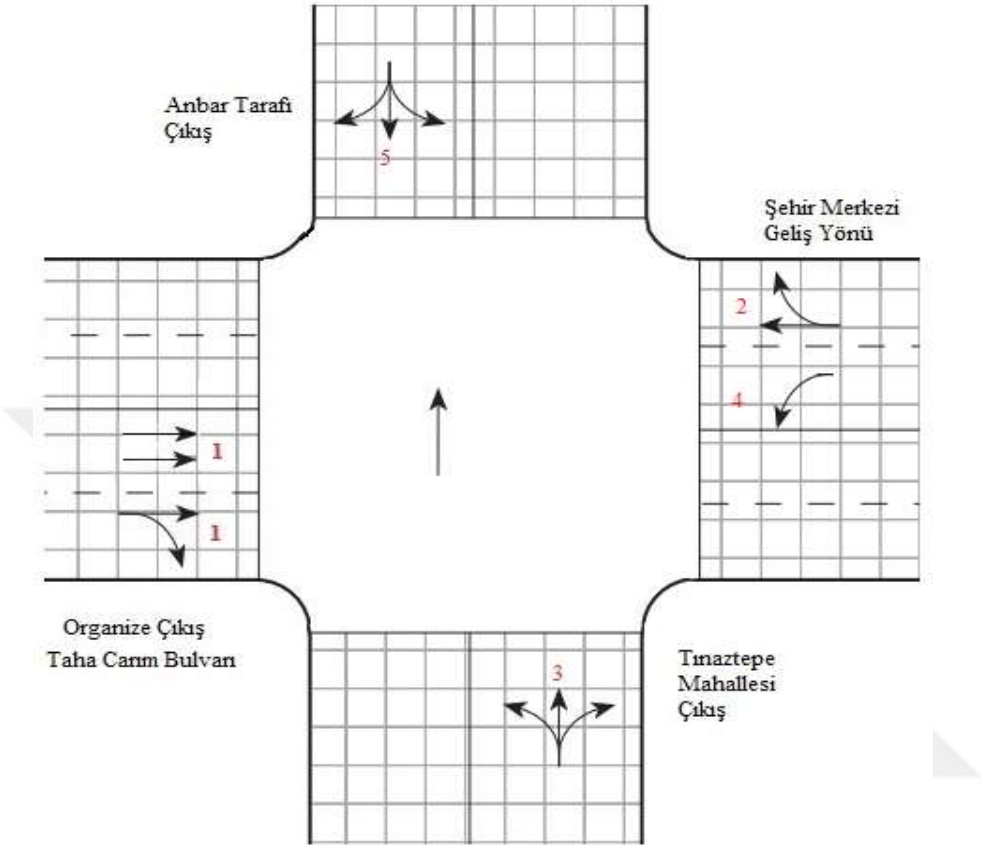


Bu kavşakta aynı şekilde sayım değerlerinde anlaşılacağı üzere en yoğun saatler sabah işe gidiş saatleridir (7.30 – 8.30). Tınaztepe kavşağın şerit sayıları ve şerit genişlikleri Tablo 4.6'de gösterilmektedir. Kavşağın her bir akım kollarına numara verilmiş bunlardan grup olarak isimlendirilmiştir. Hangi grubun hangi yöne gidecek akım kollarını temsil ettiği Şekil 4.6'daki krokide gösterilmiştir.

Tablo 4.6. *Tınaztepe Kavşak Şerit Sayı Ve Genişlikleri*

Organize tarafı geliş şerit sayısı	3	
Organize tarafı geliş şerit genişliği	11,4	metre
Organize tarafı çıkış tek şerit genişliği	3,8	metre
Şehir merkezi geliş direk gidiş şerit sayısı	2	
Şehir merkezi geliş şerit genişliği	6,7	metre
Şehir merkezi geliş tek şerit genişliği	3,35	metre
Anbar tarafı geliş yolu geliş şerit sayısı	2	
Anbar geliş şerit genişliği	6	metre
Anbar geliş tek şerit genişliği	3	metre
Tınaztepe çıkış yolu şerit sayısı	2	
Tınaztepe çıkış yolu şerit genişliği	6,7	metre
Tınaztepe çıkış yolu tek şerit genişliği	3,35	metre
Şehir merkezi geliş sola dönüş şerit sayısı	1	
Şehir merkezi geliş şerit genişliği	3,35	metre
Şehir merkezi geliş tek şerit genişliği	3,35	metre

Şekil 4.6. *Tınaztepe Kavşak Sinyal Krokisi*



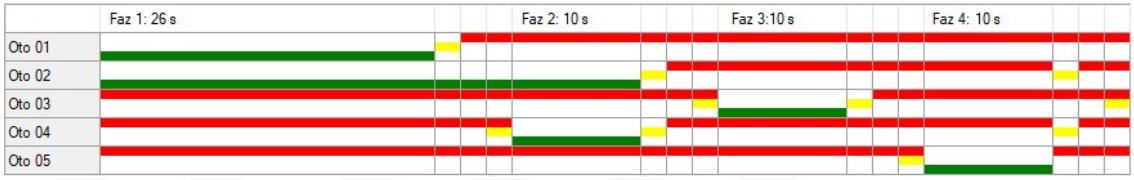
Mevcutta kavşak sayım değerlerinin birim otomobil cinsine dönüştürülerek tablo haline getirilmiştir. (Tablo 4.7) Burada kavşakta sabah yoğunluğunu en fazla olduğu grup 2 numaralı gruptur.

Tablo 4.7. *Tınaztepe Kavşak 7.30-8.30 Arası Birim Otomobil Cinsinden Araç Sayım Değerleri*

	Organize Çıkış Yönü		**Şehir Merkezi Geliş Yönü**			**Tınaztepe Mah. Çıkış Yönü**			**Anbar Geliş Yönü**		
Grup Numarası	**1**	**1**	**2**	**2**	**4**	**3**	**3**	**3**	**5**	**5**	**5**
Hareket Yönü	**Sağa Dönüş**	**Doğru Gidiş**	**Sağa Dönüş**	**Doğru Gidiş**	**Sola Dönüş**	**Sağa Dönüş**	**Doğru Gidiş**	**Sola Dönüş**	**Sağa Dönüş**	**Doğru Gidiş**	**Sola Dönüş**
Araç Sayıları	11	219	9	1187	90	134	23	193	72	21	97
Toplam Araç Sayıları	230		1196		90	350			190		

Tınaztepe kavşak mevcutta çalışma sinyal planı Şekil 4.7'de gösterilmiştir. Burada ortak çalışabilecek gruplar birlikte çalıştırılmıştır. Faz 1 de 1 ile 2'nolu Grup, Faz 2 de 2 ile 4'nolu Grup, Faz 3 de 3'nolu Grup ve Faz 4 de 5'nolu Grup çalıştırılmaktadır. Kavşağa tramvay gelmesi durumundaki etkisini inceleyecek olursa sayımlarda 1 saatlik dilim içerisinde 12 tramvay gelmiş ve kavşağı terk etme süresi ortalama 15 s olarak ölçülmüştür.

Şekil 4.7. *Tınaztepe Kavşak Mevcut Sinyal Planı*



Mevcut kavşakta grupların geçişi için verilen süreler grup1 26s, grup2 36s, grup 3 10s, grup 4 10s, grup 5 10s şeklindedir.

4.6.Çoklu Kavşakların Eşzamanlı Çalıştırılması

Çoklu kavşakların eş zamanlı çalıştırılması için en uygun çözüm yeşil dalga uygulamasıdır. Yeşil dalga uygulamalarında ana arterin akışında süreklilik sağlanarak kavşak gecikme en aza indirilmektedir. Yeşil dalga uygulaması için mesafenin hıza bölünmesi ile offset süresi bulunur. Bu süre iki kavşakta ana arterlerin açması arasındaki süre farkıdır. Burada hızı aradaki yolun hız limitine göre belirlemek gerekmektedir [28].

$$OS = \frac{\Delta L}{v} \tag{4.1}$$

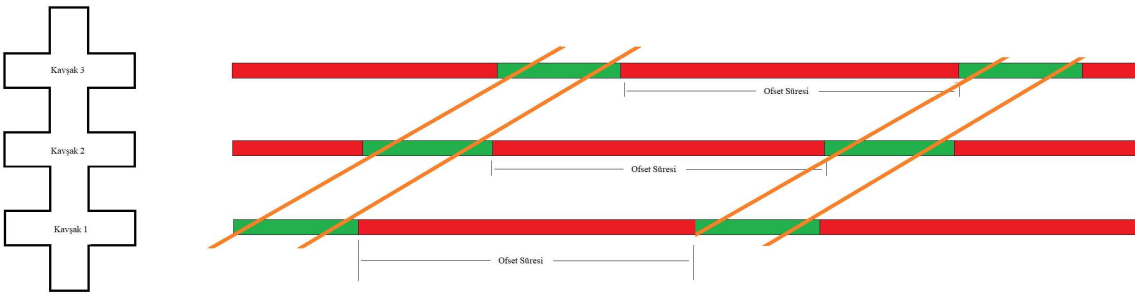
Burada OS ofset zamanını (s), ΔL iki kavşak arası uzaklığı (m), v ise hızı (m) ifade etmektedir. Mesafe 750 m, şehir içi hız sınırı 50 km/sa olduğu için OS= 54s'dir. Organize ve Tınaztepe kavşağın birlikte genel görünümü Şekil 4.8'de gösterilmiştir.

Şekil 4.8. *Organize Kavşak-Tınaztepe Kavşak Ana Arterler*



Yeşil dalga uygulamalarında GPS saati kullanılması gerekmektedir. Saatlerin hassas olması sistemin sağlıklı çalışması için önem arz etmektedir. Yeşil dala uygulanırken çalışma sinyal planı ve aradaki ofset süresinin genel gösterimi Şekil 4.9'da verilmiştir.

Şekil 4.9. *Yeşil Dalga Çalışması Sinyal Planı*



4.7.Çoklu Kavşak Üzerinde Optimizasyon

Optimizasyonda doyum akım oranı hesaplanırken temel doygun akım 1900 araç/saat alınmıştır. Ağır araç oranı etkisinin hesaplanırken ağır araç oranı %25 olarak alınmıştır. Sağa dönen araç etkisi 1, sola dönen araçların etkileri de kavşaktaki şerit durumuna göre hesaplanmıştır. Faz geçişlerinde 6s kayıp süre oluşmaktadır. Organize kavşakta 4 faz geçişi olması nedeniyle toplam kayıp süre L = 24s, Tınaztepe kavşakta 3 faz geçişi olması nedeniyle toplam kayıp süre L = 18s, olarak alınmıştır. Yaya geçişleri için optimizasyon

da ihmal edilmiş, minimum süre olarak yayaların kavşaktan geçecekleri süre dikkate alınarak her bir grubun 10s altına düşmesine izin verilmemiştir.

Burada Amerikan yöntemine göre hesaplama yapıldığında Organize kavşak toplam gecikmesi 326 s, Tınaztepe kavşakta ise 167 olarak hesaplanmıştır. Organize ve Tınaztepe kavşaklarının birlikte çalıştırılması durumundaki optimizasyon sonuçları Tablo 4.8'de gösterilmiştir.

Tablo 4.8. *Organize Kavşak – Tınaztepe Kavşak Birlikte Çalışma Optimizasyon Sonuçları*

Organize Tınaztepe Kavşakları Optimizasyon Sonuçları		SMA	PSO	GWO	HBA	HHO	SSA	TSA	GA
iterasyon sayısı (max)		1000	1000	1000	1000	1000	1000	1000	1000
Popülasyon Size		50	50	50	50	50	50	50	50
Gecikme Organize Kavşak		260	361	245	261	324	298	525	311
Gecikme Tınaztepe Kavşak		142	302	167	176	318	298	393	237
Döngü Süresi Organize Kavşak		205	281	229	193	172	231	290	254
Döngü Süresi Tınaztepe Kavşak		205	282	229	193	172	231	280	254
Toplam Gecikme		402	663	412	437	642	596	918	548
G1-G7 yeşil süre farkı		0	0	0	0	0	0	0	0
G2-G8 yeşil süre farkı		0	0	0	0	0	0	0	0
Döngü Süre farkı		0	-1	0	0	0	0	10	0
Organize K.	G1	10	59	15	11	13	35	45	24
Organize K.	G2	75	72	79	64	39	57	64	71
Organize K.	G3	20	30	26	13	15	20	33	23
Organize K.	G4	10	24	14	10	11	12	16	41
Organize K.	G5	10	13	10	10	28	20	63	10
Organize K.	G6	55	62	60	60	43	64	46	60
Tınaztepe K.	G1	10	59	16	12	12	35	48	23
Tınaztepe K.	G2	75	75	80	64	39	57	60	75
Tınaztepe K.	G3	24	80	43	15	27	38	18	23
Tınaztepe K.	G4	63	33	62	24	43	51	80	35
Tınaztepe K.	G5	14	19	10	61	33	32	56	80

BEŞİNCİ BÖLÜM

Sonuç ve Tartışmalar

5.1. Sonuçlar

Bu çalışmada 5 farklı optimizasyon algoritması ile tek kavşak, 8 farklı optimizasyon algoritması ile iki kavşak birlikte MATLAB yazılımı aracılığı ile gecikme süresi optimizasyonu yapılmıştır. Çalışmada tek kavşak optimizasyonunda klasik modeldeki yeşil süreleri ile elde edilen gecikme değeri ile optimizasyon algoritmalarının verdiği yeşil sürelerinden elde edilen kavşak gecikme sürelerinde %36.9'luk bir iyileşme görülmüştür. Bu demek oluyor ki bu tip optimizasyon sistemleri kavşaklarda araç sayıları dinamik olarak alınması ve bu verinin algoritmalarla işlenmesi durumda kavşak kontrol cihazı ile bu verilerin paylaşılması sonrası kavşakta dinamik bir yeşil süresi güncellemesi kavşaklardaki sıkışıklığı gidermede ve bekleme süresinin azaltılmasında faydalı olacağı görülmektedir. Eğer kavşakta dinamik bir araç sayım sistemi teşkil etmek durumu yok ise periyodik olarak günün belirli saatlerinde yapılacak olan araç sayımlarında optimizasyon teknikleri ile belirlenen süreler cihaza girilmesi de ciddi derecede bekleme sürelerinde iyileştirme sağlayacaktır.

İki kavşak birlikte optimize edilmesi durumunda amaç fonksiyonuna kısıt olarak iki kavşağın döngü sürelerinin eşit olması ve iki kavşağında ana arterdeki gruplarının (g1-g2) yeşil sürelerinin eşit olması da diğer kısıt olmuştur. İki kavşak yeşil dalga metoduyla birlikte çalıştırılmasının sonucunda 8 farklı optimizasyon tekniğinde en başarılı SMA olmuştur. Optimizasyon kısıtlarını sağlamış ve en düşük gecikme değerlerini vermiştir. Sonrasında ikinci sırada GWO ve üçüncü sırada HBA algoritmaları en düşük gecikme değerlerini veren metotlar olmuşlardır. Ancak bu iki algoritma iyi sonuçlar vermelerine rağmen SMA algoritması kadar kararlı sonuç üretmemişler bazı sonuçlarda kısıtları sağlayamadan sonuç üretmişlerdir. Yaklaşık ürettikleri sonuçların %65'inde sonuçları kısıtları aşmış uygulanamaz görünmektedir. Sonra gecikme süresine göre sırasıyla GA,

HHO, PSO gelmektedir. Bunlarda da aynı şekilde kararlılık görünmemektedir. TSA algoritması burada en başarısız sonuç üretmiştir. SMA algoritması kullanılarak üretilen bütün çözümlerde iki kavşağın döngü süreleri eşit, aynı yöndeki yolların Organize Kavşak g1-g2 ile Tınaztepe Kavşak g1-g2 yeşil sinyalleri arasındaki süre farkı da Tablo 4.7'de de görüleceği üzere 0 yani yeşil sürelerinde eşit sonuçlar elde etmiştir. Kavşakların mevcuttaki durumda gecikme süreleri Organize kavşak 326s, Tınaztepe Kavşak 167s'dir. Optimize ettiğimizde aldığımız değerlere bakıldığında kavşaklar birlikte çalıştırılırsa Organize kavşaktaki toplam gecikmede % 20.2 Tınaztepe Kavşaktaki toplam gecikmede %14.9 iyileştirme sağlanmıştır. Bu gecikme iyileşmesine yeşil dalgadan dolayı Organize Kavşak grup 2 ve Tınaztepe Kavşak grup 1 gecikmeleri dahildir. Yeşil dalganın tam kullanıldığını düşünürsek bu iki kolda da araçlar beklemeden geçecekleri için doğru olan bu iki akım kolunun gecikme sürelerine dahil edilmemesidir. Yeşil dalga tam kullanma durumunda Organize Kavşakta gecikme süresi 235s, Tınaztepe Kavşakta gecikme süresi 123s'dir. Yeşil dalga ile Organize Kavşakta kavşaklardaki gecikme iyileşmesi %27.9 ve Tınaztepe Kavşakta kavşaklardaki gecikme iyileşmesi %26.3'tür. Bu durumda iki kavşakta toplam gecikme iyileşmesi %27.4'tür. Yeşil dalga ile optimizasyon modelini ikiden fazla kavşağa da uygulamak mümkündür. En başarılı optimizasyon modeli ile dinamik araç sayılarının periyodik olarak değerlendirilip optimize edilerek yeşil dalga ile kavşakların çalıştırılmasının ortalama araç gecikmesinde başarılı olacağı yapılan simülasyonlarda görünmektedir. Gelecekteki yapılacak çalışmalarda yeşil süre kararlarında SMA optimizasyon metodu kullanılabilir ve anlık araç sayılarının alınması durumunda gün içerisindeki değişikliklere hızlı bir şekilde yeni yeşil süreleri üretilebilir. Kavşak kontrol cihazlarına bu anlık verilerin işlenmesi ile dinamik bir akıllı kavşak modeli oluşturulabilir. Ayrıca bu karar mekanizmasına art arda gelen birden fazla kavşağın araç sayım değerleri verilmesi durumunda da dinamik bir akıllı kavşak bölgesi oluşturulması mümkündür.

KAYNAKÇA

[1] Z. A. Polat, A. Memduhoğlu, M. Acar, H. Duman, "Kentsel Büyüme ile Motorlu Araç Trafiği Yoğunluğu Arasındaki İlişkinin Belirlenmesi", *Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi,* 6(2), 442-451, 2017.

[2] Ö. Mutlu, Z. Durak, H. Akyer, "Kademeli Mesai Saati ile Trafik Sıkışıklığının Azaltılması", *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 2*6(4), 730-736, 2020.

[3] A. A. M. Harb, A. Durdu, H. Terzioğlu, "Kavşak Trafik Sinyalizasyon Kontrolü için Bulanık Mantık Yöntemi ile Gerçek Zamanlı Sistemin Tasarımı ve Uygulaması", *Avrupa Bilim ve Teknoloji Dergisi Özel Sayı, Özel Sayı*, 490-497, 2019.

[4] B.Ç. Gençosman, "Radyal Tabanlı Fonksiyon Ağları ile Adaptif Trafik Sinyal Kontrolü", *Journal of Current Researches on Engineering, Science and Technology (JoCREST)*, 7(4), 1-18, 2017.

[5] E. Doğan, A.P. Akgüngör, "Bulanık Mantık ile Diğer Sinyal Denetim Sistemlerinin Karşılaştırılması: Üç Kollu Sinyalize Kavşak Örneği". *El-Cezeri Fen ve Mühendislik Dergisi,* 3(1), 110-117, 2016.

[6] T.A.M. Mahmood, "Akıllı Şehirler için Adaptif Trafik Sinyalizasyon Kontrolü", Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2019.

[7] A.A.M. Harb, "Akıllı Kontrol Yöntemleri ile Trafik Sinyalizasyon Kontrolü", Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 2019.

[8] I. Tunç, M.T. Söylemez, (2019 Eylül 11-14) *Kavşak Yönetiminde Bulanık Mantık ve PI Kontrol Yöntemlerinin Benzetimi*, TOK'2019 Otomatik Kontrol Ulusal Toplantısı, Muğla.

[9] I. Tang, F. Gao, F. Lıu, X. Chen, "A Denoising Scheme-Based Traffic Flow Prediction Model: Combination of Ensemble Empirical Mode

Decomposition and Fuzzy C-Means Neural Network". *IEEE Access*, Vol.8, p. 11546-11559, 2020.

[10] A. Karaağaç, "Sezgisel Arama Algoritmaları ile Trafik Sinyal Optimizasyonu", Doktora Tezi, Erciyes Üniversitesi Fen Bilimleri Enstitüsü, Kayseri, 2020.

[11] E. Korkmaz, "İzole Sinyalize Kavşaklarda Yapay Zekâ Teknikleri ile Trafik Sinyal Kontrolü ve Optimizasyonu", Doktora Tezi, Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü, Kırıkkale, 2019.

[12] S. Yavuz, "Çiçek Tozlaşma Algoritması ile Devre Süresi Modellerinin Gerçekleştirilmesi", Yüksek Lisans Tezi, Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü, Kırıkkale, 2019.

[13] Y.Ş. Murat, "Denizli Şehiriçi Kavşaklardaki Trafik Akımlarının Bigisayarla İncelenmesi", Yüksek Lisans Tezi, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Denizli, 1996.

[14] F.V. Webster, ve B.M. Cobbe, *Traffic Signals,* H.M.S.O., Londra, 111, 1966.

[15] Ulaşım Araştırma Kurulu, Ulaşım Araştırma Konseyi, *Highway Capacity Manual 2000 (HCM, (2000),* TRB publication, Washington, 2000.

[16] O. Eraslan, "Işıklı Kavşaklarda Amerikan ve Avusturalya Yöntemleri ile Gecikme Analizi ve Örnek Bir Kavşak çözümü", Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2008.

[17] J. Holland, *Adaptation in natural and artificial systems,* MIT Press, 1975.

[18] D. Whitley, "A genetic algorithm tutorial", *Statistics and Computing,* . Technical Report CS-93-103, Mart 1993.

[19] S. Li, H. Chen, M.Wang, A.A. Heidari, S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization", *Future Generation Computer Systems*, 111, 300-323, 2020.

[20] I.B. Uğur, "Büyük Ölçekli Çelik Yapıların Sezgisel Optimizasyon yöntemleriyle Optimum Tasarımı", Doktora Tezi, Dicle Üniversitesi, Fen Bilimleri Enstitüsü, 2021.

[21] E.V. Altay, O. Altay, "Güncel meta-sezgisel optimizasyon algoritmalarının CEC2020 test fonksiyonları ile karşılaştırılması", *Dicle University Journal of Engineering,* 12(5), 729-741, 2021.

[22] R. Eberhart, J. Kennedy, (04-06 Ekim 1995) *A new optimizer using particle swarm theory. In MHS'95*, Proceedings of the Sixth International Symposium on Micro Machine and Human Science (pp. 39-43), Ieee, 1995.

[23] S. Mirjalili, S.M. Mirjalili, A. Lewis, "Grey Wolf Optimizer", *Advances in English Software*, 69, 46-61, 2014.

[24] L. Doğan, U. Yüzgeç (11-13 Mayıs 2018), *Robot Path Planning using Gray Wolf Optimizer in Proceedings* - International Conference on Advanced Technologies Computer Engineering and Science (ICATCES'18), 2018.

[25] M.S. Kıran, "TSA: Tree-seed algorithm for continuous optimization", *Expert Systems with Applications*, 42(19), 6686-6698, 2015.

[26] S. Mirjalili, A. H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, S. M., "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems", *Advances in Engineering Software*, 114, 163-191,2017.

[27] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, "Harris hawks optimization: Algorithm and applications", *Future Generation Computer Systems*, 97, 849–872, 2019.

[28] F.A.Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, W. Al-Atabany, "Honey badger algorithm: New metaheuristic algorithm for solving optimization problems", *Math. Comput. Simul.* 192, 84–110. doi:10.1016/j.matcom.2021.08.013, 2022.

[29] S.P. Çalışkanelli, S.Tanyel, "Sinyalize Kavşaklarda Doygun Akım Değerinin İrdelenmesi", *İMO Teknik Dergi*, Yazı 496, 8225-8248, 2018.

[30] H. Kara, G. Sarısoy, Y. Delice "Yeşil dalga uygulamalarında ışıklı kılavuz sistem kullanımının değerlendirilmesi", *Akıllı Ulaşım Sistemleri ve Uygulamaları Dergisi*, Cilt:5 – Sayı:2, 2022

ÖZGEÇMİŞ

1. İLETİŞİM BİLGİLERİ

Adı Soyadı: Olcay ADIYAMAN

Uyruğu: Türkiye (T.C)

2. ÖĞRENİM DURUMU

Derece	**Alan**	**Üniversite**	**Yıl**
Yüksek Lisans	Elektrik-Elektronik Mühendisliği	Kayseri Üniversitesi	2019-2023
Lisans	Elektrik-Elektronik Mühendisliği	Erciyes Üniversitesi	1998 -2003

3. YABANCI DİL : İngilizce

4. İŞ DENEYİMLERİ

Yıl	**Kurum**	**Görev**
2006-2008	TSE-Sojuztest Met. Ve Kal.Ltd.Şti –KAYSERİ	Planlama Sorumlu Mühendisi
2008-2009	Kayseri Raylı Sistem Projesi Eser Müşavirlik	Kontrol Mühendisi
2009-2017	Kayseri Ulaşım A.Ş. – KAYSERİ	**Elektronik Sistemler Şefi**
2017-2020	Kayseri Ulaşım A.Ş. – KAYSERİ	**Atölye Müdürü**
2020 -	Kayseri Ulaşım A.Ş. – KAYSERİ	**Elektrik-Elektronik Tesisler Müdürü**

TEZ İNİTİHAL RAPORU

META-SEZGİSEL ALGORİTMALAR İLE TRAFİK SİNYAL OPTİMİZASYONU

Yazar Olcay Adıyaman

Gönderim Tarihi: 26-Tem-2023 01:57PM (UTC+0300)
Gönderim Numarası: 2137061241
Dosya adı: Olcay_ADIYAMAN_tez_r7_25072023.docx (5.18M)
Kelime sayısı: 13284
Karakter sayısı: 86366

ORİJİNALLİK RAPORU

%8	%7	%1	%4
BENZERLİK ENDEKSİ	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCİ ÖDEVLERİ

BİRİNCİL KAYNAKLAR

1	acikbilim.yok.gov.tr İnternet Kaynağı	%2
2	Submitted to The Scientific & Technological Research Council of Turkey (TUBITAK) Öğrenci Ödevi	%1
3	acikerisim.pau.edu.tr:8080 İnternet Kaynağı	%1
4	polen.itu.edu.tr İnternet Kaynağı	%1
5	www.researchgate.net İnternet Kaynağı	%1
6	enstitu.kayseri.edu.tr İnternet Kaynağı	%1
7	Submitted to Bahcesehir University Öğrenci Ödevi	<%1
8	manasbis.manas.edu.kg İnternet Kaynağı	<%1
9	Submitted to Middle East Technical University Öğrenci Ödevi	<%1

10	gcris.ktun.edu.tr İnternet Kaynağı	<%1
11	acikerisimarsiv.selcuk.edu.tr:8080 İnternet Kaynağı	<%1
12	katalog.istanbul.edu.tr İnternet Kaynağı	<%1
13	tez.sdu.edu.tr İnternet Kaynağı	<%1
14	Submitted to Eskisehir Osmangazi University Öğrenci Ödevi	<%1
15	acikerisim.dicle.edu.tr:8080 İnternet Kaynağı	<%1
16	Submitted to Erciyes Ãniversitesi Öğrenci Ödevi	<%1
17	jag.journalagent.com İnternet Kaynağı	<%1
18	Dana Marsetiya Utama, Meri Dines Primayesti. "A novel hybrid Aquila optimizer for energy-efficient hybrid flow shop scheduling", Results in Control and Optimization, 2022 Yayın	<%1
19	Dipu Sarkar, Sagar Kudkelwar, Debasree Saha. "Optimal Coordination of Overcurrent	<%1

Relay Using Crow Search Algorithm", Smart Science, 2019
Yayın

20	acikerisim.btu.edu.tr İnternet Kaynağı	<%1
21	ouci.dntb.gov.ua İnternet Kaynağı	<%1
22	acikerisim.nku.edu.tr:8080 İnternet Kaynağı	<%1
23	hit.alljournals.cn İnternet Kaynağı	<%1
24	teses.usp.br İnternet Kaynağı	<%1
25	abis-files.kayseri.edu.tr İnternet Kaynağı	<%1
26	dergipark.org.tr İnternet Kaynağı	<%1
27	"Innovations in Smart Cities Applications Volume 5", Springer Science and Business Media LLC, 2022 Yayın	<%1

Alıntıları çıkart Kapat
Bibliyografyayı Çıkart üzerinde
Eşleşmeleri çıkar Kapat