T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

İSTANBUL'DAKİ METROBÜS HATTI İÇİN GPS TABANLI KAZA VE ARIZA ALGILAMA ALGORİTMASI

Sadullah GÖNCÜ

YÜKSEK LİSANS TEZİ

İnşaat Mühendisliği Anabilim Dalı

Ulaştırma Programı

Danışman

Prof. Dr. İsmail ŞAHİN

Aralık, 2019

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

İSTANBUL'DAKİ METROBÜS HATTI İÇİN GPS TABANLI KAZA VE ARIZA ALGILAMA ALGORİTMASI

Sadullah Göncü tarafından hazırlanan tez çalışması çalışması 25.12.2019 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü İnşaat Mühendisliği Anabilim Dalı, Ulaştırma Mühendisliği Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. İsmail ŞAHİN
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Prof. Dr. İsmail ŞAHİN, Danışman
Yıldız Teknik Üniversitesi

Doç. Dr. Halit ÖZEN, Üye
Yıldız Teknik Üniversitesi

Prof. Dr. Hilmi Berk ÇELİKOĞLU, Üye
İstanbul Teknik Üniversitesi

Prof. Dr. İsmail ŞAHİN sorumluluğunda tarafımca hazırlanan “İSTANBUL’DAKİ METROBÜS HATTI İÇİN GPS TABANLI KAZA VE ARIZA ALGILAMA ALGORİTMASI” başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim

Sadullah GÖNCÜ

İmza

Aziz Milletimize

ve

Şehitlerimize

TEŞEKKÜR

Öncelikle bu tezin faydalı olmasını yürekten temenni ediyorum.

İstanbul'daki Metrobüs sistemi açıldığı ilk günden bu yana şehrin önemli bir yükünü taşımaktadır. Şüphesiz ki bu yükü hangi hizmet düzeyinde taşıdığı farklı bir tartışma konusu olmalıdır fakat son kertede, Metrobüs sisteminin bu hayati önemi, işletenleri sürekli tetikte ve acil durumlara karşı hazırlıklı olmaya zorlamaktadır. Bu konuda işletmeciler ciddi çaba sarf etmektedir. Fakat bu çaba insan gücüne dayalı bir şekilde sürdürülmektedir. Tez kapsamındaki çalışma bu çabayı insan gücünden bir nebze olsun çıkartarak, kolaylaştırmayı amaçlamaktadır. Tez kapsamındaki problem tanımının ortaya çıkmasından, tez çalışmasını tamamladığımız güne kadar birçok kişinin maddi ve manevi desteği geçmiştir. Burada bazı isimleri anmadan geçmemem gerektiği kanaatindeyim. En başta değerli danışmanım, tecrübelerini ve bilgisini hiçbir zaman benden esirgemeyen, benimle beraber bu yolculuğun her safhasında büyük emekleri olan Prof. Dr. İsmail ŞAHİN hocama teşekkürü bir borç bilirim. İlgili problem tanımının olgunlaşmasına verdikleri katkı ve sağladıkları veriler için başta İETT Bilgi İşlem Daire Başkanı Dr. Abdullah Önder TÜRKOĞLU Bey'e, Akıllı Sistemler Müdürü Yavuz YALÇIN Bey'e, Erkan LAMPİR Bey'e, İETT Coğrafi Bilgi Sistemleri ve Teknik Müdürlüğünden Güneş UYANIKER Hanım'a, Nagihan KALAYCI Hanım'a, İSBAK Coğrafi Bilgi Sistemleri Biriminden Elif Ceylan MENİZ Hanım'a şükranlarımı sunarım. Yüksek lisans eğitimim boyunca ders almış olduğum Doç. Dr. Mustafa GÜRSOY'a, Dr. Öğr. Üyesi Mustafa Sinan YARDIM'a, Ulaştırma Mühendisliğine yönelmemi sağlayan değerli Dr. Öğr. Üyesi Güzin AKYILDIZ-ALÇURA'ya müteşekkirim. Tez jürimdeki değerleri hocalarıma ise kıymetli önerileri ve gelecek çalışmalara yönelik verdikleri fikirler için bu vesile ile teşekkürü bir borç bilirim.

Lisans eğitimimi tamamladığım ve Araştırma Görevlisi olarak çalıştığım Fatih Sultan Mehmet Vakıf Üniversitesi, İnşaat Mühendisliği Bölüm Başkanı Prof. İbrahim EKİZ'e, üzerimde çok emeği olan değerli Prof. Dr. Zekai CELEP'e, çok kıymetli Dr. Öğr. Üyesi Cenk AKSOYLAR ve Dr. Öğr. Üyesi Nihan DOĞRAMACI

AKSOYLAR hocalarıma verdikleri destekler için canı gönülden minnet duymaktayım. Ayrıca Fatih Sultan Mehmet Vakıf Üniversitesi Bilgisayar Mühendisliği Bölümü Araştırma Görevlisi arkadaşlarım Muhammet ALKAN ve Samed KAYA'ya tez süresince verdikleri destekten ötürü ayrıca teşekkür ederim. Ayrıca mesai ve oda arkadaşlarım olan Araştırma Görevlileri Mustafa MERT, Sezer ÖZTÜRK, Haşim ÇAYIR, Muhammed Şükrü YAVAŞ'a, yüksek lisans yolcuğuna beraber çıktığım Yusuf İslam KARAKUŞ kardeşime teşekkürlerimi sunuyorum.

Ve son olarak, değerli babam Sadık GÖNCÜ'ye, annem Şerife GÖNCÜ'ye, ablam Dr. Beyza Servet GÖNCÜ'ye, ağabeyim Muhiddin GÖNCÜ, kuzenlerim Semih GÖNCÜ ve Ahmet Zeki YURTTAŞ'a, ev arkadaşım Said Halid YELEĞEN'e, bütün bu yolculuk boyunca maddi manevi sürekli yanımda oldukları için haklarını asla ödeyemem. Yukarıda saydığım veya sayamadığım bütün değerli isimlerin desteğine rağmen tezde yer alabilecek her türlü hata şahsıma aittir.

Sadullah GÖNCÜ

İÇİNDEKİLER

SİMGE LİSTESİ ix

KISALTMA LİSTESİ x

ŞEKİL LİSTESİ xi

TABLO LİSTESİ xii

ÖZET xiii

ABSTRACT xv

1 GİRİŞ 1

1.1 Literatür Özeti ... 1
1.2 Tezin Amacı ... 11
1.3 Hipotez ... 11

2 TEMEL KAVRAMLAR 12

2.1 BRT Sistemleri ve Metrobüs ... 12
2.2 Global Konumlama Sistemi (GPS) ... 15
2.3 Python Programlama Dili ve Kütüphaneleri ... 16
2.4 QGIS Programı ... 16
2.5 Yol-Zaman Diyagramları ... 16

3 YÖNTEM 19

3.1 Veri Ön İşleme ... 19
3.2 Mesafe Atama Algoritması ... 25
3.3 Kaza-Arıza Algılama Algoritması ... 28
3.4 Algoritmanın Sınanması ... 35

4 ANALİZ VE SONUÇLARI 37

4.1 Yol-Zaman Diyagramları ... 37
4.2 Hız Haritaları ... 45
4.3 Algoritma Performansı ... 51

5 SONUÇ VE ÖNERİLER 63

5.1 Tartışma ... 63

5.2 Sonuç ve Öneriler 64
5.3 Gelecek Çalışmalar 66
KAYNAKÇA 67
A YOL-ZAMAN DİYAGRAMLARI 70
B HIZ HARİTALARI 75
TEZDEN ÜRETİLMİŞ YAYINLAR 83

SİMGE LİSTESİ

t	Birim zaman
$\gamma_{1,2}$	Boylam
d	Eğri bir yüzey üzerindeki iki nokta arasındaki en kısa mesafe
$\phi_{1,2}$	Enlem
T	Gözlem süresi
L	Gözlem Uzunluğu
s	Mesafe cinsinden takip aralığı
NK	Toplam Kaza Sayısı
N	Trafik birimi
x	Uzunluk, Konum
h	Zaman cinsinden takip aralığı

KISALTMA LİSTESİ

BRT	Bus Rapid Transit, Hızlı Otobüs İşletmeciliği
DR	Detection Rate, Algılama Oranı
FAR	False Alarm Rate, Yanlış Algılama Oranı
GPS	Global Positioning System, Global Konumlama Sistemi
GIS	Geographic Information Systems, Coğrafi Bilgi Sistemleri
İETT	İstanbul Elektrik Tramvay ve Tünel İşletmeciliği
MTTD	Mean Time to Detect, Algılamaya Kadar Geçen Ortalama Süre
PR	Penetration Rate, Penetrasyon Oranı
TTI	Texas Transportation Institue, Teksas Ulaştırma Enstitüsü
UCB	University of California Berkeley, Kaliforniya Berkeley Üniversitesi

ŞEKİL LİSTESİ

Şekil 2.1 Metrobüs Güzergahı 14
Şekil 2.2 Yol – Zaman diyagramı 18
Şekil 3.1 QGIS programı ve referans poligon 27
Şekil 3.2 206 numaralı kesim için ortalama hızların dağılımı 31
Şekil 3.3 213 numaralı kesim için ortalama hızların dağılımı 32
Şekil 3.4 Algoritma akış diyagramı 35
Şekil 4.1 Yol–Zaman diyagramı 1 39
Şekil 4.2 Yol–Zaman diyagramı 2 40
Şekil 4.3 Yol-Zaman diyagramı 3 40
Şekil 4.4 Yol–Zaman diyagramı 4 41
Şekil 4.5 Yol-Zaman diyagramı 5 41
Şekil 4.6 Yol–Zaman diyagramı 6 42
Şekil 4.7 Yol–Zaman diyagramı–kaza kaydı 1 43
Şekil 4.8 Yol–Zaman diyagramı–kaza kaydı 2 44
Şekil 4.9 24.02.2019 Yol-Zaman diyagramı-kaza kaydı (yakınlaştırılmış) 1 45
Şekil 4.10 24.02.2019 Yol–Zaman diyagramı-kaza kaydı (yakınlaştırılmış) 2 46
Şekil 4.11 Darboğaz bölgesi hız haritası 1 47
Şekil 4.12 Darboğaz bölgesi hız haritası 2 48
Şekil 4.13 Darboğaz bölgesi hız haritası 3 48
Şekil 4.14 Anadolu Avrupa yönü hız haritası – Avrupa Anadolu yönünde 211 no'lu kesimde meydana gelen kazadan etkilenen kesimler (hızlar: km/sa) 49
Şekil 4.15 Avrupa Anadolu yönü hız haritası - 211'inci kesim kaza-arıza (hızlar: km/sa) 50
Şekil 4.16 Normal karayolu trafiğinde, kaza-arıza durumu 51
Şekil 4.17 Metrobüs hattında, kaza-arıza durumu 51

TABLO LİSTESİ

Tablo 2.1 İETT-Metrobüs Filo Bilgileri 14
Tablo 2.2 İETT-Metrobüs hat ve sefer bilgileri 15
Tablo 2.3 İETT veri setlerinin özellikleri 16
Tablo 3.1 Mesafe ataması için QGIS programından alınan referans matris örneği 28
Tablo 3.2 Örnek hız–zaman–kesim matrisi (hızlar, km/sa) 30
Tablo 3.3 Örnek karakteristik hız matrisi 32
Tablo 3.4 Örnek takip-uyarı matrisi 35
Tablo 4.1 Eldeki verilerin sınıflandırılması 54
Tablo 4.2 Avrupa anadolu yönü hız-zaman-kesim matrisi 56
Tablo 4.3 Anadolu avrupa yönü hız-zaman-kesim matrisi 57
Tablo 4.4 Avruupa anadolu yönü takip–uyarı matrisi 57
Tablo 4.5 Anadolu avrupa yönü takip–uyarı matrisi 58
Tablo 4.6 Avrupa anadolu yönü için ilgili komşu kesimlerin incelenmesi 59
Tablo 4.7 Anadolu avrupa yönü ilgili komşu kesimlerin incelenmesi 60
Tablo 4.8 Avrupa anadolu yönü, kaza kesimindeki araç listesi 60
Tablo 4.9 Veri kümelerine göre kaza-arıza algılama durumları 62

ÖZET

İstanbul'daki Metrobüs Hattı için GPS Tabanlı Kaza ve Arıza Algılama Algoritması

Sadullah GÖNCÜ

İnşaat Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Prof. Dr. İsmail ŞAHİN

Büyük veri çağında trafik mühendisleri artık verileri sahadan bizzat toplamak yerine, veri madenciliği yöntemlerini kullanarak veya veri depolayan kurum/kişilerden temin ederek çalışmalarını yürütmeye başlamıştır. Özellikle ulaştırma mühendisliği alanında trafiğin temel dinamikleri ve ilişkilerinin, bu veriler kullanılarak elde edilebilir olması, çeşitli akıllı ulaştırma sistem bileşenlerinin geliştirilmesine olanak sunmaktadır. İstanbul'daki Metrobüs hattı, şehrin en yoğun kullanılan ulaştırma sistemlerinden biridir. Kayıtlara göre (2017) sistemi günde yaklaşık 800.000 kişi kullanmaktadır. Metrobüs hattındaki tüm taşıtlara yerleştirilmiş GPS cihazları aracılığıyla hat üzerinden zaman ve konum verileri otomatik olarak toplanmaktadır. Bu bakımdan, her bir Metrobüs taşıtı akım içinde birer sonda aracı işlevi görmektedir. Bu tez çalışması kapsamında, İETT'den elde edilen GPS verileri kullanılarak İstanbul Metrobüs sistemi için sonda aracı tabanlı bir kaza-arıza algılama algoritması geliştirilmiştir. Bu algoritmanın geliştirilebilmesi için, Metrobüs sistemi GPS verilerinden yararlanılarak ayrıntılı şekilde incelenmiştir. GPS verilerinin kullanılmasıyla taşıt

yörüngeleri oluşturularak, yol-zaman diyagramları çizilmiştir. Sistemin hız haritaları çıkartılarak, tekrarlı darboğazlar tespit edilmiştir. Sistemin normal şartlar altındaki davranışlarının analizinden sonra ise kaza-arıza durumlarına ait veriler kullanılarak sistemin kaza-arıza durumunda nasıl davrandığına yönelik incelemeler yapılmıştır. Tüm analizlerden sonra kaza-arıza algılama algoritmasının tasarımı yapılmış ve algoritma sınanmıştır. Literatürün mevcut durumu ve algoritmanın performansı irdelenmiş ve gelecekte yapılabilecek çalışmalardan bahsedilmiştir. Çalışma kapsamında, ayrıca, büyük veri olan GPS kayıtlarının işlenmesi ve filtre edilmesine yönelik genelleştirilmiş ilkeler de önerilmektedir.

Anahtar Kelimeler : Metrobüs, kaza ve arıza , algılama, toplu taşıma

ABSTRACT

GPS Based Incident Detection Algorithm for Istanbul's Metrobus System

Sadullah GÖNCÜ

Department of Civil Engineering

Master's Thesis

Advisor: Prof. Dr. İsmail ŞAHİN

In the era of "Big Data", the need for traffic surveys is partially gone. Researchers can collect transportation related data with the use of data mining methods or directly from the transportation agencies. The ability to acquire information about the dynamics of a transportation system from the big data revealed new pathways for developing new intelligent transportation systems. Metrobus system in Istanbul is one of the most important transportation sub systems in the city. According to the records (2017) more than 800.000 passengers uses the system daily. The entire Metrobus fleet is equipped with onboard GPS devices, sending latitude and longitude information in real time to the IETT's (Istanbul electric tramway and tunnel establishments) database. In this regard, each Metrobus vehicle acts as a probe vehicle in the flow of the system. In this thesis, a probe-based incident detection algorithm is developed for Istanbul's Metrobus system, using only GPS data. To develop this algorithm, a detailed analysis of

the system's current performance has been deployed with the use of historical data. With the use of GPS data, trajectories of the vehicles are constructed and time-space diagrams are drawn. Speed maps of the system are generated for detecting recurrent bottlenecks and evaluating the speed changes throughout the system. Incident data are acquired and analyzed with the same manner to evaluate the system's behavior under incident conditions. After all the analysis, the algorithm is designed and tested. The current state of the literature is discussed and the performance of the algorithm is examined. Since the acquired GPS data is classified as "Big Data", generalized principles for GPS data processing and filtering are suggested.

Keywords: Metrobus, incident, detection, public transportation

1
GİRİŞ

1.1 Literatür Özeti

20.Yüzyıl'ın başında, ilk trafik akımı modellemelerini yapanlar şüphesiz ki bugün gelinen noktayı tahayyül edemezlerdi. Günümüzde akıllı ulaştırma sistemlerinin ortaya çıkması ve modelleme tecrübesinin çok gelişmesi ile beraber, günümüz ulaştırma problemlerine çok geniş perspektiflerde çözümler üretilmektedir. Özellikle büyük şehirlerde yaşantının zorlaşması ile beraber, toplu taşımaya yönelik eğilimler artmaktadır. Ulaşım planlamacıları da bu eğilimleri göz önünde bulundurarak toplu taşıma ekseninde bir planlama yapmaya çalışmaktadır. 'Büyük Veri' kavramının doğması ile beraber akıllı ulaştırma sistemleri de yeni bir ivme kazanmıştır. Çalışma kapsamında İstanbul'daki en önemli toplu taşıma sistemlerinden biri olan Metrobüs için GPS tabanlı bir kaza-arıza algılama algoritması önerilmiştir. Metrobüs filosunun tamamında mevcut olan GPS altyapısını kullanarak, trafik mühendisliğinin yol-zaman, hız–zaman gibi konvansiyonel diyagramları oluşturulmuştur. GPS verileri üzerinden araç yörüngelerini yol-zaman diyagramları üzerinden oluşturmak literatürde az da olsa mevcuttur fakat kullanılan yöntem ile ilgili kısıtlı bilgi verilmiştir Metrobüs koridor trafiğinin makroskopik analizlerinden sonra söz konusu algoritma oluşturulmuştur. Çalışma kapsamında bu konu detaylıca açıklanacaktır.

21.Yüzyıl'da insanın en kıymetli kaynağı olan zaman, trafik mühendislerinin de temel çalışma alanlarından biridir. Her kısıtlı kaynak gibi zaman da yönetilmeli ve israfının önüne geçilmelidir. Günümüzde şehirlerde yaşayan insanların önemli vakit kayıplarından birisi ulaşım fiili sırasında gerçekleşmektedir. Ulaşım ihtiyacı belirli amaç uğruna oluşmaktadır. Ulaşımın kendisi genelde amaç değildir. Kişiler aktivitelerini gerçekleştirmeden önce belirli bir mesafe katetmeleri gerekmektedir. Bu durum ulaşım fiilinin verimsizliğini ifade etmektedir. Bu verimsizliğe birde trafik kazaları vb. durumların gecikmeleri

eklenince ciddi bir vakit kaybı ortaya çıkmaktadır. Günümüz ulaştırma sistemleri planlama, tasarım, işletme gibi adımların her birinde vakit kaybını belirli şartlar altında en küçüklemek temel ilkelerden biridir.

Akıllı ulaştırma sistemleri ile seyahat süresi yönetimi yeni bir perspektif kazanmıştır. Günümüzde tekrarlı olabilecek trafik tıkanıklıklarını istatistiksel yöntemler ve veri tabanlarının kullanımıyla tahmin etmek mümkündür. Bu tahminler günümüzde yolcu bilgilendirme hizmetleri ile tüm vatandaşlara ilan edilebilmektedir. Fakat trafik kazalarını tahmin etmek, tekrarlı trafik tıkanıklıkları kadar kolay olmamaktadır. Trafik tıkanıklıkları genellikle sistem üzerindeki tekrarlı darboğazlardan kaynaklanırken, trafik kazaları sistemin herhangi bir noktasında yeni bir darboğaz oluşturma potansiyeline sahiptir.

Trafik akımındaki ani bir değişim, akabinde ciddi bir kullanım oranı kaybı ve ciddi tıkanıklık artışı trafik kazaları olarak tanımlanır. Bu kazaları tam bir doğrulukla meydana gelmeden önce tahmin etmek oldukça zordur. Bu durum ulaştırma sistemlerinin her an işlevsiz hale gelebilme ihtimali olduğunun ilanıdır. Bu durumda ulaştırma mühendisleri ise trafik kazası olduktan sonra mümkün olan en kısa sürede durumu saptayarak müdahale etme stratejileri gütmektedirler. Genel manada bu çalışmaların ana başlığı kısa zamanlı trafik tahmini olarak literatürde geçmektedir. Kısa zamanlı trafik tahmini konusu Vlahogianni ve ark (2014) tarafından ele alınarak, bu alanda mevcut teknolojinin avantajları ve dezavantajları üzerine bilgi verilmiştir. Fakat kısa dönemli trafik tahmini başlık olarak çok geniş bir literatürü ifade etmektedir. Bu literatür özeti kapsamında trafik kazalarının algılanması üzerinde durulacaktır.

Kaza-arıza algılama sistemleri, 1960'lardan itibaren kendisini literatürde göstermeye başlamıştır. Kaza-arıza algılama algoritmalarını kullandıkları yöntemlere göre sınıflandırmak mümkündür (Parkany ve Xie, 2005):

A) Yol Tabanlı Algoritmalar

i) **Karşılaştırmalı algoritmalar:** Temel olarak bir trafik parametresinin (akım, işgal, hız vb.) daha önceden belirlenmiş bir eşik değerine göre değerlendirilmesi ilkesine dayanır. En önemli örneği olan California

Algoritması, Payne (1976) tarafından geliştirilmiştir. Karşılaştırmalı Algoritma sınıfına giren California Algoritması, temel olarak bir kaza-arıza durumunda akım yukarısında işgal oranının artarken akım aşağı kesimde ise işgal oranının düşük olacağı kabulüne dayanır. Bu algoritmanın geliştirilmiş çeşitleri literatürde mevcuttur.

ii) **İstatistiksel Algoritmalar:** Temel istatistik yöntemlerle, toplanan sensör/detektör verilerindeki değişimlerin hesaplanması ya da tahmin edilmesine yönelik çalışmaktadır. Dudek ve ark. (1974) daha önce gözlem yoluyla toplanan trafik verilerini kullanarak bir kaza-arıza algılama algoritması geliştirmiştir. İlgili algoritma, şerit işgal oranı için hesaplanan standart normal sapma değerini kullanarak bir algılama gerçekleştirmektedir. Kısaca %92'lik bir algılama oranı (Detection Rate) ve %1 yanlış algılama oranı (False Alarm Rate) olarak sonuçlanmıştır.

iii) **Zaman Serisi Algoritmaları:** Trafik akımlarının belirli örüngeleri belirli zaman aralıklarında takip ettiği, ilgili örüngelerin analiz edilmesiyle trafik tahminleri yapılabileceği kabulüne dayanan algoritmalardır (Ahmed ve Cook, 1977, 1982). Temelde trafik verilerinin zaman serisi olarak modellenmesini amaçlar. İlgili modelden sapmalar ise algılama sistematiğinin temelini oluşturmaktadır.

iv) **Filtreleme ve Yumuşatma Algoritmaları:** Temel olarak gürültülü trafik verilerini eleyerek, yanlış algılama oranlarını iyileştirmeye yönelik algoritmalardır. Gürültülü veri ortadan kalktığında, kaza-arıza olaylarının daha kolay saptanabileceği kabulüne dayanır. Literatürde özellikle dalgacık dönüşüm yöntemi (Wavelet Transformation) ile oluşturulmuş birçok filtreleme ve yumuşatma algoritması mevcuttur (Jeong ve ark., 2011; Karim ve Adeli, 2002).

v) **Trafik Modellemesi Yapan Algoritmalar:** Trafik akım kuramının temel ilkelerini kullanarak kaza-arıza durumlarında trafik türetilmiş

değişkenlerinin (akım, hız, yoğunluk) tahmine dayanan algoritmalardır (Gall ve Fall, 1989; Wilsky ve ark., 1980).

vi) **Yapay Zekâ Algoritmaları:** Karmaşık durumlarda karar verebilen ve sürekli olarak kendini güncelleme mantığına dayalı çok çeşitli yöntemler kullanarak çalışan algoritmalardır (Ishak ve Al-Deek, 1998; Samant ve Adeli, 2000).

vii) **Görüntü İşleme Algoritmaları:** Trafik kameralarındaki veya kayıt altına alınmış trafik görüntülerin, görüntü işleme teknolojileri ile beraber veriye dönüştürülmesi ve temel trafik büyüklüklerinin elde edilmesi ilkesine dayanan algoritmalardır (Michalopoulos, 1991).

B) Sonda Aracı Tabanlı Algoritmalar

i) **MIT Algoritması:** Parkany ve Bernstein (1995) tarafından geliştirilen algoritma, temel olarak otoyollarda sıkça kullanılan elektronik ücret tahsil teknolojisinin, kaza-arıza algılama amacıyla geliştirdiği algoritmalardır. Algoritmanın geliştirilmesi sırasında MITSIM adlı simülasyon programı ile üç şeritli bir yol kesimi belirli bir mesafe için modellenmiştir. Simülasyondan alınan veriler, sahadan toplanan veriler ile karşılaştırılmıştır. Bir algılama oranı belirleyecek kadar deneme yapılmadığını fakat mevcut kaza verileri içerisinde algılamanın sorunsuz yapıldığı, çalışmayı yapan araştırmacılar tarafından belirtilmiştir (Parkany ve Bernstein, 1995).

ii) **ADVANCE Algoritması:** Sethi ve ark. (1995) tarafından ana arterlerdeki kaza-arızaları algılamak için geliştirilmiştir. Sistem analiz edilerek, kaza durumu ve kaza olmayan durum için trafik akımı modellenmiştir. Konum-zaman verileri ile oluşturulan bu modeller, algılama oranlarını yüksek tutmak, yanlış algılama oranlarını ise düşük tutmak için seyahat süresini de göz önüne almıştır. Genel olarak GPS verisi veya otomatik araç konum algılama (AVL – Automatic Vehicle Location) verisi kullanılmaktadır. Sonuç olarak, veri çözünürlüğü ve doğruluğu arttıkça algoritma performansının arttığı

gözlemlenmiştir. Sermons ve Koppelman, (1996) ise ADVANCE projesinin devamı olarak GPS verilerini kullanmış, yol kesimlerinin her biri için gerçekleşen seyahat süresi, toplam seyahat süresi ve anlık araç konum bilgilerini gerçek zamanlı izleyen bir algoritma geliştirmiştir. GPS cihazlarına sahip sonda araçlarının trafik içerisindeki kompozisyonlarına ve verinin güvenilirliği arttıkça algoritmaların performanslarının arttığı görülmüştür.

iii) **TTI Algoritmaları:** Teksas Ulaştırma Enstitüsü (Texas Institue of Transportation – TTI) tarafından geliştirilen algoritma, sonda araçları ile kaza-arıza algılama sistemlerinin uygulanabilirliğini ölçmek için deneysel bir çalışma yürütmüştür (Balke ve ark. 1996). Yaklaşık 200 civarı yolcu, kendilerine verilmiş olan cep telefonu benzeri iletişim araçları ile kendilerinin önemli noktalardan (kent meydanları, önemli sanat yapıları, tarihi binalar vb.) geçerken konumlarını bu iletişim araçları ile bildirmeleri istenmiştir. Oluşturulmuş olan çağrı merkezinden, her konum bildirisi sırasında ilgili görevli, yolcunun referans numarasını, bildiri zamanını ve konumunu ana sisteme kaydetmiştir. Bu şekilde birbirini takip eden iki referans noktası arasındaki seyahat süresi hesaplanmıştır. Yolcuları taşıyan sonda araçları yaklaşık beş dakikalık takip aralıkları ile işletilmiş, bir kaza-arıza durumunda seyahat süresinin artacağı kabulüne göre bir fonksiyon üretilmiştir. Bu fonksiyon Standart Normal Sapma değerleri kullanılarak oluşturulmuştur. Çalışmanın ön bulguları düşük penetrasyon oranlarında (sonda aracı/toplam trafik akım hacmi) düşük performans göstermektedir.

iv) **Kaliforniya Üniversitesi, UCB Algoritması:** California algoritması gibi literatürde sıkça karşımıza çıkan bir diğer algoritma, Lin ve Daganzo (1997) tarafından geliştirilen Kaliforniya Üniversitesi, Berkeley (UCB) algoritmasıdır. Sahadan toplanmış olan verileri kullanarak, işgal oranının zamana göre değişimini incelemiş ve belirli sınırların dışında kalan değerlerin kaza-arıza durumunu ifade ettiğini öne sürmüşlerdir.

UCB algoritması da daha sonraki yıllarda çeşitli değişimlere uğramıştır. Petty ve ark. (1997) araçların içerisine yerleştirilmiş olan radyo vericilerini kullanarak UCB algoritmasını kullanmışlardır. Hız değerlerindeki değişim üzerinden kaza-arıza algılaması mantığıyla çalışan algoritma, kaza-arıza durumlarında aracın hızının yavaşlayacağı veya duracağı kabulüne göre kaza-arıza durumlarını belirlenmesi ilkesine dayanmaktadır.

v) **TRANSMIT Algoritması:** TTI algoritmasına benzer olan TRANSMIT algoritması (Mouskos ve ark, 1999), araçların yolculuk sürelerinin kaza-arıza olmayan durumda normal dağılıma uyacağı kabulüne dayanmaktadır. Kaza-arıza durumu, birbirini takip eden araçların varması gereken noktalara, varması gereken süreden daha geç vardığı durumda tespit etmektedir. Diğer saha verilerini kullanan algoritmalara benzer bir performans gösteren TRANSMIT algoritmasının, yol kesimlerinin geometrik tasarımlarına göre zayıf performans gösterebildiği belirtilmektedir.

vi) **Çift Fazlı - Yol Segmenti Algoritması:** Kamran ve Haas (2007)'ın geliştirdiği bu algoritma, ilgili yol koridorunu kesimlere bölerek, gerçek zamanlı GPS verileri üzerinden, kesimlerdeki normal trafik akımı durumunda ortalama hızları belirlemiştir. Kesimler, normal trafik akımı dışında bir durum sergilediğinde, konum-zaman analizi yaparak kaza-arızanın olduğu yol kesimi belirlenmiştir.

vii) **MOSES Algoritması:** Cheu ve ark. (2002) tarafından geliştirilen algoritma, araçlara yerleştirilmiş mobil sensörler sayesinde araçlardan veri toplayarak çalışan bir algoritmadır. Kaza-arıza durumunun gerçekleştiği yol kesiminde, en az iki sonda aracının ilgili kesimdeki seyahat süresini kullanmaktadır. Algoritma, seyahat sürelerindeki farkları istatistiksel olarak değerlendirmeye alarak kaza-arıza durumunu otomatik bir şekilde belirlemeye yöneliktir. Algoritma performansının, sonda aracı sayısının artmasıyla yükseldiği

belirtilmiştir. Trafik kompozisyonun en az %50'si sonda aracı olduğu durumda, algoritma kullanılabilir bir performans göstermektedir.

viii) **BEAM Algoritması:** Li ve McDonald (2005) tarafından geliştirilen algoritma ile yol kesimlerinin seyahat süreleri ve birbirini takip eden kesimlerinin seyahat süresi farklarını, iki değişkenli analiz yöntemini kullanarak hem kaza-arıza olan durumda hem de normal koşullarda tespit yapılabilmektedir. Kaza – arıza durumunun, ilgili kesimlerdeki seyahat sürelerinin ciddi uzaması durumunda gerçekleştiği varsayımına göre algılama sistematiği oluşturulmuştur.

ix) **Aykırı Değer Algılama Algoritması:** Zhu ve ark. (2009) tarafından geliştirilen bu algoritma, ana arterlerdeki sonda araçlarından gelen verilere bağlı olarak konum-zaman içerisindeki hız değişimlerini izler ve birbirini takip eden yol kesimlerinin hız farklarını inceleyerek kaza-arıza durumunu tespit etmeye çalışmaktadır.

x) **Anomali Algılama Algoritması:** Güncel bir çalışma olan bu algoritma Kinoshita ve ark. (2015) tarafından geliştirilmiştir. Bu algoritma sonda araçlarından gelen veriler ile anormal olarak sınıflandırılabilecek araç hareketlerini inceleyerek ve seyahat süresi üzerinden kaza-arıza durumlarını algılamaya çalışmaktadır.

xi) **Şok Dalgası ve Seyahat Süresi Algoritması:** Asakura ve ark. (2017) tarafından yapılan çalışmada, Tokyo'daki bir otoyoldaki sonik detektörlerden okunan hız verileri ve trafik içerisine çeşitli penetrasyon oranları ile çeşitli takip aralıklarında işletilmiş olan GPS vericili sonda araçlarının topladığı veriler kullanılarak iki farklı algoritma geliştirilmiş ve literatürdeki diğer algoritmalarla karşılaştırılmıştır. Birinci algoritma seyahat süresine göre bir algılama yapmaya çalışırken, ikinci algoritma ise şok dalgası kuramını kullanarak bir algılama yapmaktadır. Sonuç olarak California ve UCB algoritmaları ile karşılaştırılabilecek bir performans elde edilmiş ve penetrasyon oranı arttıkça performansın iyileşeceği belirtilmiştir.

C) **Arter Bazlı Algoritmalar:** Ana arter olarak sınıflandırılan yollar, normal yollardan daha farklı davranış gösterebilmektedir. Ana arterler üzerindeki yaya geçitleri, parklanma alanları, otobüs durakları gibi çeşitli fiziki sistemler sebebiyle trafik akımında düzensizlikler oluşmaktadır. Bu tarz bir belirsizliğin var olduğu ana arterlerde, bunlara ek olarak kaza-arıza durumları ile birleşince trafik akımındaki düzensizlikler rahatsız edici boyutlara ulaşabilmektedir.

i) **Biçim Eşleme Algoritmaları:** Biçim eşleşme algoritmaları ya da biçim algılama algoritmaları, ana arterlerin trafik özelliklerini takip ederek trafik değişkenlerini hesaplamakta ve trafik içerisindeki tekrarlı örüngeleri keşfetmektedir. Sonrasında belirlenen parametrelerin ilgili değerleri karşılaştırılarak, kaza-arıza olan durum ile normal şartlardaki trafik değişkenleri belirlenmekte ve karşılaştırma yapılabilmektedir. Thancanamootoo ve Bell (1988) bu konudaki ilk çalışmaları yapanlar olarak, zaman serileri üzerinden bir biçim algılama algoritması geliştirmiştir. Detektör aracılığı ile gelen trafik akımı ve işgal oranı üzerinden bir zaman serisi tekniği kullanmışlardır. Akım yukarı ve akım aşağı trafik akımlarını aldıkları değerlere göre kaza-arıza algılaması yapmayı hedeflemişlerdir. Algoritma performansının genel manada kaza-arızanın olduğu konumun detektörlere olan uzaklığı ile değiştiği gözlenmiştir. Ayrıca tekrarlı darboğazların etkileri ile kaza-arıza durumunun ayrıştırılması mümkün olmamıştır. Bir diğer deyişle tekrarlı darboğazların etkileri de kaza-arıza durumu olarak algılanmıştır.

ii) **Kalman Filtresi Algoritmaları:** Kalman filtresi metodu, öğrenebilme yeteneğine sahip, değişken tahmini için kullanılan bir yöntemdir. Ayrıca verilerdeki gürültüyü silme işlevi de görmektedir. Arter bazlı Kalman filtreli kaza-arıza algılama algoritmaları için ilk çalışmalardan birini Lee ve Taylor (1999) yapmıştır. Detektör bazlı geçmiş dönem trafik verilerini kümülatif şekilde kullanarak kendini sürekli güncelleyen bir sisteme dönüştürmüşlerdir. Kısmi değiştirilmiş lineer

kalman filtresi metoduyla gerçek trafik akım ve hız değerlerini hesaplamaya çalışmışlardır. Bu hesaplanan değerler mevcut değerler ile sürekli olarak karşılaştırılarak, kaza-arıza algılanmaya çalışılmıştır. Genel manada Kalman filtresi ile çalışan algoritmalar, trafik akımının değerlerini sağlıklı geçmiş dönem verileri ile belirlemede başarılıdır, fakat uygulanması ve model kalibrasyonu oldukça karmaşık bir hal alabilmektedir.

iii) **Ayrıştırıcı Analiz Algoritmaları:** Ayrıştırma (diskriminant) analizi, bir çeşit matematiksel sınıflandırma metodu olup, Sethi ve ark. (1995) tarafından hem sonda aracı tabanlı hem de sabit detektör verilerini birbirinden bağımsız olarak kullanarak, ayrıştırma analizi ile çalışan bir algoritma geliştirmişlerdir. Algoritma, trafik akımındaki bozulmaların, ilgili kesimde a) İşgal oranında artış, hızda düşüş oluşturacağı, b) ilgili kesimde ve akım yukarıda, trafik akımı düşeceği kabulüne dayanır.

iv) **Bulanık Mantık Algoritmaları:** İlk bulanık mantık tabanlı arter kaza-arıza algılama algoritması ilk olarak Lee ve ark. (1998) tarafından geliştirilmiştir. Bulanık mantık gerçek zamanlı işleyecek olan, muhakeme yapması gereken sistemlerde verimli bir şekilde çalışmaktadır.

Yukarıda verilmiş olan bölümlendirme Parkany ve Xie (2005) tarafından verilen literatür sınıflandırmasının bu tezin yazarı tarafından kısmi olarak genişletilmiş bir versiyonu olup, tarihsel gelişim sürecine yönelik genel bir çerçeve çizmeye çalışmaktadır. Tezin yazarı, yukarıda zikredilmemiş olup geliştirilmiş olan algoritmalar olduğu gibi, yukarıda zikredilen algoritmaların başka araştırmacılar tarafından ileriye götürülmüş hallerinin de olduğunun unutulmaması kanaatindedir.

Yol-Zaman Diyagramlarının GPS Verilerinden Elde Edilmesine Yönelik Literatür

Çalışma kapsamında, yukarıda verilmiş olan sınıflandırmaya göre sonda aracı tabanlı, İstanbul'daki Metrobüs sistemi için bir kaza-arıza algılama sistemi

geliştirilmiştir. Fakat bu tarz algoritmalar geliştirilmeden öncede mutlaka sistem kendi mevcut durumu içerisinde analiz edilmeli ve karakteristikleri belirlenmelidir. Bu sistemin analizini yapmak için çeşitli yollar olabilir, fakat tez kapsamında konvansiyonel metotlar olan yol-zaman diyagramları ve hız haritaları kullanılmıştır. Tez kapsamında GPS verilerinden yol–zaman diyagramlarının nasıl çıkarılacağı ile ilgili literatür araştırması yapıldığında, ilgili literatürün oldukça sınırlı olduğunu görülmüştür. Li ve ark. (2013) tarafından yapılmış olan çalışmada Urumçi'deki BRT (Bus Rapid Transit – Yüksek Hızlı Otobüs İşletmeciliği) sistemi için GPS verilerinden yararlanarak bir demetlenme (bunching) analizi yapılmıştır. Demetlenme olgusu kent içi otobüs işletmeciliğinde sık karşılaşılan bir problem olup kullanım oranında düşüşlere sebep olmaktadır. Bu analizlerden önce ise sistemi yol–zaman diyagramı olarak analiz etmiş, tekrarlı darboğazları ve demetlenmeleri göz ile görülecek şekilde görselleştirmiştir. Fakat aynı çalışmada bu diyagramların nasıl oluşturulduğuna yönelik verilen bilgi oldukça kısıtlıdır. Benzer şekilde bahsedilebilecek diğer bir çalışma ise Anwar ve ark. (2013) tarafından yapılan çalışma olup, Los Angeles şehrindeki kentiçi otobüs hatlarının performansı analiz edilmiştir. Demetlenmeleri gerçek zamanlı şekilde gösteren bir bilgisayar programı oluşturulmuştur. Program gerçek zamanlı sistemi izlerken bir nevi üç boyutlu yol–zaman diyagramı çizmektedir. Bu şekilde araç yörüngeleri uç boyutlu sistemde ortaya çıkarılmaktadır. GPS verilerinin kullanılmasıyla yapılan çalışmaların çoğunluğu otoyollar ve kentiçi otobüslerin analizleri için yapılmaktadır. BRT'ler özelinde yapılmış çalışmalar yok denecek kadar azdır. Kentiçi otobüsler için yapılan çalışmalarda ise genelde harita eşleme yöntemleri kullanılarak araç yörüngeleri oluşturulmakta fakat bu yörüngeler yol-zaman diyagramı biçiminde gerçekleşmemektedir. Yukarıda zikredilen iki çalışma GPS verilerinden araç yörüngelerinin yol–zaman diyagramı formunda çıkarılmasına yönelik olan sınırlı çalışmalardır. Bu çalışmalar dışında GPS verilerinden yol-zaman diyagramları çıkarılmış olanlar mevcuttur fakat bu çalışmaların hiçbiri bunun nasıl yapıldığına yönelik bilgi içermemektedir. Bu tezin kapsamında ise bununla ilgili oluşturulan yöntemler kapsamlı şekilde anlatılmıştır.

1.2 Tezin Amacı

Çalışmanın temel amacı, İstanbul'daki Metrobüs sistemi için GPS tabanlı bir kaza – arıza algılama algoritması geliştirmektir. Bu şekilde bir sistemin geliştirilmesi için sistemin detaylı bir analizinin gerektiği açıktır. Sistemin mevcut durum performansının analizi ve değerlendirmesini yapmak tezin amaçları arasındadır. Çalışma, yapısı bakımından özgün olduğu için çeşitli analizler içinde özgün yöntemler geliştirme ihtiyacı ortaya çıkmıştır. Bu yöntemler çalışma kapsamında detaylı olarak açıklanmış olup, gelecek çalışmalar için araştırmacılara yol gösterici olması umulmaktadır.

1.3 Hipotez

Tez kapsamında, İstanbul'daki Metrobüs sistemi için kaza-arıza algılama stratejisinin, insan gücü ve gözlemine dayalı olan mevcut halini otomatik bir şekle dönüştürmenin, sistemin yönetilmesini kolaylaştıracağı düşünülmektedir. Metrobüs sistemi her yönde tek şerit olmak suretiyle işletildiği için, sistemde tıkayıcı bir etki oluşturan tüm olaylar eğer hızlı müdahale ile yönetilmezse ciddi kullanım oranı kayıplarına sebebiyet vermektedir. Geliştirilecek olan kaza-arıza algılama algoritması ile kaza-arıza durumunun hızlı tespiti mümkün olup, müdahalenin daha hızlı gerçekleştirileceği düşünülmektedir. Bu durum sistemin, kaza-arıza durumu ortadan kalktıktan sonra tekrar eski normal seyrine dönüş süresini de kısaltacaktır. Bir bakıma kaza-arıza sebebiyle sistemde oluşan kullanım oranı kayıplarını en küçüklemek mümkün olacaktır.

Literatür özeti, tezin amacı ve hipotezin belirtildiği birinci bölümün ardından tez, konu ile ilgili kavramların açıklandığı ikinci bölüm ile devam edecektir. Kullanılan yöntemler ve algoritmanın yapısı üçüncü bölümde anlatıldıktan sonra dördüncü bölümde yapılan sistem analizlerinin sonucu ve algoritma performansı incelenecektir. Beşinci bölümde ise literatür ve geliştirilen algoritma tartışılarak, gelecekte yapılabilecek çalışmalardan bahsedilecek ve son olarak sonuç ve öneriler kısmı ile tez tamamlanacaktır.

2
TEMEL KAVRAMLAR

2.1 BRT Sistemleri ve Metrobüs

İkinci bölümde, çalışma kapsamında sıkça karşımıza çıkan bazı temel kavramlardan bahsedilip, çalışmayı yaparken yararlandığımız bilgisayar programları ve programlama dilleri hakkında bilgiler verilecektir.

BRT, Hızlı Otobüs İşletmeciliği isminin İngilizce karşılığı olan "Bus Rapid Transit"' kavramının kısaltmasıdır. Raylı sisteme benzer şekilde yüksek yolcu kapasiteli, konforlu ve hızlı şekilde çalışan bir lastik tekerlekli sistemdir. Kendine ait şeridi veya alanı olan, karışık trafikten izole olarak işletilen bir sistemdir. Türkiye'deki tek BRT sistemi İstanbul'daki Metrobüs sistemi olup, şehrin ana ulaşım türlerindem biridir. İstanbul'da günde yaklaşık 800.000 kişi Metrobüs sistemini kullanmaktadır.

BRT sistemleri geleneksel otobüs taşımacılığı sistemlerinden farklı bir yerde değerlendirilebilir. İşletme şekli olarak raylı sistemler ile benzerlik gösterse bile, temel olarak bir lastik tekerlekli sistemdir. Bu nedenle karayolu trafiğinin ilkeleri büyük oranda geçerli olacaktır. Fakat yatırım ve işletme maliyetleri göz önüne alındığında ise performans/fiyat oranı yüksek bir ulaşım türü olduğu açıktır.

Metrobüs hattı ilk olarak, yapımı 2007 yılında biten 18,3 kilometrelik Topkapı – Avcılar hattında hizmet vermeye başlamıştır. İkinci hizmete alınan Topkapı-Zincirlikuyu kısmı ise 2008 yılında açılmıştır. Üçüncü etap Zincirlikuyu – Söğütlüçeşme 2009 yılında, dördüncü etap Beylikdüzü – Avcılar kısmı ise 2011 'de hizmete açılmıştır. Metrobüs koridorunun Boğaziçi Köprüsü kesimi hariç, tamamı karışık trafikten izole şekilde tasarlanmıştır. Toplam koridor uzunluğu

52 km'dir (Şekil 2.1). Metrobüs filosunda çeşitli araçlar bulunmaktadır. Metrobüs filosundaki araçlar ile ilgili bilgiler Tablo 2.1'de verilmiştir.

Tablo 2.1 İETT-Metrobüs Filo Bilgileri

Araç Tipleri	Araç Sayısı	Yolcu Kapasitesi
Capacity	250	165
Phileas	50	258
Citaro	100	160
Conecto	85	160
Karsan	50	155
Toplam	535	

Temel işlevi, hareketliliğin arttırılması ve özel araç kullanım oranını düşürerek emisyon düşüşü sağlamak olan Metrobüs sistemi, açıldığı ilk günden beri ciddi talep ile karşı karşıya kalmıştır. Metrobüs hattının işletmesi İETT (İstanbul Elektrik Tramvay ve Tünel İşletmeleri) tarafından yürütülmekte olup, İETT İstanbul Büyükşehir Belediyesine bağlıdır.



Şekil 2.1 Metrobüs güzergahı (https://metrobus.iett.istanbul sitesinden alınmıştır.)

Yapılan çalışmalar sonucunda zirve saat içerisinde ortalama olarak 420 ile 500 civarında aracın aktif işletildiği görülmüştür. Metrobüs koridorunun mevcut işletme durumu Tablo 2.2'de görülebilir.

Tablo 2.2 İETT-Metrobüs hat ve sefer bilgileri (https://metrobus.iett.istanbul/tr/ alınmıştır.)

Hat	Güzergah	Araç Sayısı	Hat Uzunluğu (Km)	Ortalama Araç Hızı (km/sa)
34	AVCILAR-ZİNCİRLİKUYU	92	30	34
34A	CEVİZLİBAĞ-SÖĞÜTLÜÇEŞME	22	22	30
34 AS	AVCILAR-SÖĞÜTLÜÇEŞME	100	41,5	31
34 BZ	BEYLİKDÜZÜ-ZİNCİRLİKUYU	128	40	31
34 C	BEYLİKDÜZÜ-CEVİZLİBAĞ	70	29	35
34 G	BEYLİKDÜZÜ-SÖĞÜTLÜÇEŞME	15	52	35
34 U	ZİNCİRLİKUYU - UZUNÇAYIR	38	11	38
34 Z	ZİNCİRLİKUYU-SÖĞÜTLÜÇEŞME	30	11,5	31

Metrobüs hattı içerisinde işletilmekte olan hatlardan 34G hariç hepsi gündüz saatlerinde bir arada işletilmektedir. 34G hattı ise gece boyu tüm koridor boyunca hizmet vermektedir. Tüm Metrobüs araçlarında GPS (Küresel Konumlama Sistemi) vericileri bulunmakta ve 10 ile 60 saniye arasında değişen periyotlarla anlık konum bilgilerini İETT veri tabanı ile paylaşmaktadırlar.

2.2 Küresel Konumlama Sistemi (GPS)

GPS sistemi, dünya yörüngesinde bulunan 30 uydu sayesinde anlık konum bilgisini öğrenebildiğimiz uydu şebekesine verilen isimdir. Tarihsel olarak bakıldığında Amerika Birleşik Devletleri'nin askeri amaçla geliştirdiği sistem bugün GPS özelliği olan herhangi bir elektronik alet sayesinde ilgili uydu şebekesinin sinyallerini algılayabilmekte ve kendi konumunu belirleyebilmektedir. GPS teknolojisi de kendi içerisinde çeşitli gelişimler göstermiştir. Teknoloji ile beraber dünya yüzeyini modelleme konusunda da yeni gelişmeler ortaya çıkınca, günümüzde hata oranı oldukça düşük "datum" olarak adlandırdığımız modeller çıkmıştır. Günümüzde en sık kullanılan datum WGS84 'tür. Çalışmamız kapsamındaki GPS sinyalleri de bu datuma göre depolanmıştır.

Tez kapsamındaki çalışma için İstanbul Elektrik Tramvay ve Tünel İşletmeleri Genel Müdürlüğü'nden (İETT) temin edilmiş veri kümeleri kullanılmıştır. Bu çalışmada kullanılan veriler 24-31 Ocak 2019 tarihleri arasını içermekte olup, her gün için yaklaşık 3,6 milyon satırlık veri içermektedir. Verilerin özellikleri ve içeriği Tablo 2.3'de verilmiştir.

Tablo 2.3 İETT veri setlerinin özellikleri

Başlık	**Açıklama**	**Başlık**	**Açıklama**
Konum ID	Her bir konum verisi için özel olan bir numara	***Boylam***	Konum verisine ait boylam bilgisi
Kapı Numarası	Her araç için tek olan bir kod	***Enlem***	Konum verisine ait enlem bilgisi
Gerçekleşme Zamanı	Sinyalin gönderildiği an (gg/aa/yy – ss/dd/sn)	***Hız***	GPS cihazı tarafından ölçülen anlık hız (km/sa)
Kayıt Zamanı	Sinyalin, merkezde algılanıp depolandığı an (gg/aa/yy – ss/dd/sn)	***Mesafe***	Sefer başlangıcından beri kat edilen mesafe (km)

2.3 Python Programlama Dili ve Kütüphaneleri

1990'lı yıllarda ortaya çıkan Python dili (Rossum, 1995) birçok programlama ekolüne uygun şekilde çalışabilen, oldukça popüler bir programlama dilidir. Özellikle son yıllarda "Büyük Veri" kavramının popülerleşmesiyle, Python gibi birçok, veri analizine uygun diller popüler hale gelmiştir. Python, dilinin okunabilirliğinin yüksek olması, kütüphanelerinin çeşitliliği ve zenginliği ile öne çıkmaktadır. Özellikle Pandas, Scipy, Numpy gibi kütüphaneler sayesinde veride ayıklama, değiştirme, filtreleme, eklemleme işlemleri yapılabilmektedir.

2.4 QGIS Programı

QGIS (QGIS Yapımcı Ekibi, 2009) bir coğrafi bilgi sistemi programıdır. Çeşitli harita veri tabanlarını kullanabilmektedir. Açık kaynak ve kullanımı ücretsiz olan bu program, özellikle haritalar üzerinde vektör işlemlerini, harita eşleme, planlama gibi birçok alanda kullanılabilmektedir.

2.5 Yol-Zaman Diyagramları

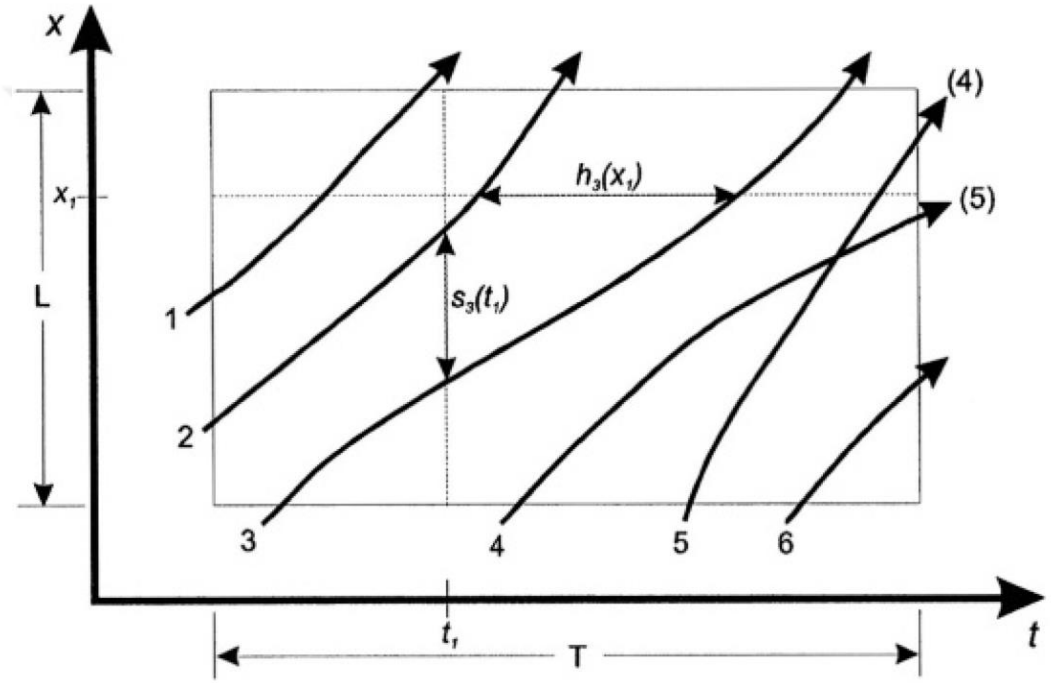
Ulaştırma mühendisliğinin temel amaçlarından biri de ilgili sistemin, topluma olan olumsuz tüm etkilerini en küçüklemektir. Bunu yapabilmek için ulaştırma mühendisleri, ilgilendikleri sistemlerin özelliklerini matematiksel olarak modelleme yoluna gitmişler; böylece, sistemin olumsuz yönleri daha belirgin bir biçimde saptanarak iyileştirme önerileri geliştirilebilmektedir. Karayolu trafiği özelinde ise trafik akımlarının modellenmesi, ancak trafik akımını doğru analiz etmek ile gerçekleştirilebilir. Konvansiyonel olarak trafik akımı çeşitli değişkenler aracılığıyla tanımlanır.

a) **Trafiğin Asal Değişkenleri**: Bu değişkenler, tek başlarına yeterli olmayan fakat ikili şekilde incelendiğinde, trafik hakkında bize bilgi veren değişkenlerdir. Bu değişkenler sırasıyla, Uzunluk (x) (m, km), Zaman (t) (sn, dk, sa), birim trafik elemanı (N) (Taşıt) olarak tanımlanabilir.

b) **Trafiğin Türetilmiş Değişkenleri:** Trafiğin asal değişkenlerinin her biri sırasıyla limit sıfıra yakın incelenirken diğer iki değişkenin beraber incelenmesi sonucu ortaya çıkan değişkenlerdir. Örneğin:

$$
\begin{aligned}
&N-t \rightarrow (\lim_{x\rightarrow\Delta x}) \rightarrow \frac{N}{t}(\frac{taşıt}{saat}) \rightarrow Trafik\ Hacmi \\
&N-x \rightarrow (\lim_{t\rightarrow\Delta t}) \rightarrow \frac{N}{x}(\frac{taşıt}{km}) \rightarrow Trafik\ Yoğunluğu \\
&x-t \rightarrow (N=1) \rightarrow \frac{x}{t}(\frac{km}{saat}) \rightarrow Trafik\ Hızı
\end{aligned}
\tag{1}
$$

Yol–zaman diyagramı ise denklem takımı 1'de verilen x-t ilişkisinin grafiğidir. Yani düşey eksende uzunluk, yatay eksende ise zaman olan bir grafiktir. Örnek bir yol-zaman grafiği Şekil 2.2'de görülebilir.



Şekil 2.2 Yol–Zaman diyagramı (Cassidy, 2006)

Herhangi bir yol kesiminde, örneğin bir otoyolun tek şeridi, herhangi bir yaya yolu, bir uçağın belirli bir rotayı takip etmesi vb. gibi durumlar için cisimlerin hareketleri zaman ve kartezyen koordinatlar üzerinden ifade edilebilir. Burada yatay eksene zamanı yerleştirerek, düşey eksen ise alınan mesafe olarak kullanılırsa cismin izlediği tüm yörünge görülebilir. Şekil 2.2'de görülebileceği gibi, her bir numaralandırılmış çizgi, bir aracın yörüngesini ifade etmektedir. Buradaki her bir eğrinin geometrik değişimleri, araçların hareketleri hakkında bilgi vermektedir. Örneğin eğrilerin eğimlerindeki düşüşler hareketin

yavaşladığını gösterirken, eğimdeki artışlar hareketin hızlandığını göstermektedir. Yukarıda bahsettiğimiz trafiğin türetilmiş değişkenlerinin hepsini yol-zaman diyagramlarından elde etmek mümkündür. Bir t_1 anında diyagramda alınacak düşey kesitte, kesitin üzerinden geçen eğrilerin sayısı, L gözlem uzunluğuna bölünürse, trafik yoğunluğu elde edilebilir. Benzer şekilde bir başlangıç noktası ile herhangi bir X_t konumunda yatayda alınacak kesitte, kesit çizgisini kesen tüm eğrilerin sayısı T gözlem süresine bölündüğünde hacim değerleri bulunabilmektedir. Her eğrinin eğimi ise, araca ait diyagramın içerdiği tüm kesimdeki ortalama hızları vermektedir. Şekil 2.2'de görülebileceği gibi birbirini takip eden iki eğri arasındaki düşey mesafe, uzunluk cinsinden takip aralığını verirken ($s_3(t_1)$), yataydaki mesafe zaman cinsinden takip aralığını ($h_3(x_1)$) vermektedir. Geleneksel trafik etütlerinde, yol–zaman diyagramlarının çizimi için tüm asal değişkenlere ait veriler sahadan toplanır. Daha sonra bu veriler işlenerek yol–zaman diyagramları oluşturulmaktadır. Günümüzde ise yeni veri tiplerinin oluşması sebebiyle, yol–zaman diyagramlarını oluşturmak için gereken verilere ancak farklı tipteki verilerin işlenmesi, dönüştürülmesiyle ulaşılmaktadır. Çalışma kapsamında kullanılan GPS verilerinden yol-zaman diyagramlarının oluşturulmasına yönelik yöntem Bölüm 3'te detaylı şekilde açıklanacaktır.

3
YÖNTEM

Çalışma kapsamında seçilen amaç doğrultusunda, İETT Bilgi İşlem Daire Başkanlığından her biri 24 saatlik olmak üzere toplamda 8 günlük veri alınmıştır. Bu verinin özellikleri Bölüm 2.2'deki Tablo 2.3'te verilmiştir. Bir kaza-arıza algoritması geliştirilmeden önce, ilgili ulaştırma sistemi ayrıntılı şekilde incelenmelidir. Bu inceleme doğrudan, İETT tarafından sağlanan veriler ile yapılmıştır. Sistemin incelenmesi sırasında izlenen yol bu bölümde sırasıyla açıklanacaktır.

Literatür özeti bölümünde incelendiği gibi, özellikle otoyollar üzerinde yapılmış olan çalışmalarda trafiğin temel dinamikleri veriye dayalı olarak rahatça çıkarılmaktadır. Fakat toplu taşımada kullanılan yöntemler kentiçi otobüs ile yapılan toplu taşımanın ötesine çok az geçmektedir, ayrıca araç yörüngeleri ise kentiçi otobüsler için genelde harita eşleme yöntemlerinin iyileştirilmesine yönelik kullanılmaktadır. Bu bölümde, İETT'den elde edilen GPS verileri kullanılarak İstanbul Metrobüs sistemindeki taşıtların yörüngelerinin oluşturulması ve sonrasında ise Metrobüs koridorunun hız haritalarının çıkarılmasına yönelik yöntem açıklanacaktır.

3.1 Veri Ön İşleme

GPS verileri kullanılarak BRT sisteminin taşıt yörüngelerinin bir yol–zaman diyagramı formunda çıkarılmasına yönelik çalışmalar yok denecek kadar azdır. Bu durumun özellikle GPS verilerine erişimdeki zorluktan kaynaklandığı düşünülmektedir. Ham GPS verilerini işlemenin de kendi içerisinde bazı zorlukları olduğu, kullanılan GPS datumlarının etkisi, hatalı verilerin filtre edilmesinin zorlukları göz önüne alınırsa, literatürün bu konuda sınırlı olması bir nebze anlaşılabilmektedir. Li ve ark. (2013), Çin'in Urumçi kentinde işletilmekte olan BRT sisteminin GPS verileri kullanılarak yol–zaman diyagramlarını

çıkarmışlardır. Çalışma genel olarak yapılan analizlere odaklandığı için, diyagramların çıkarılmasında kullanılan yöntem hakkında yeterli bilgi verilmemiştir. Urumçi'deki ilgili çalışmanın temel amacı, diyagramlar üzerinde çeşitli analizler yaparak BRT sistemi üzerinde zirve saatlerde meydana gelen demetlenme problemlerini analiz ederek çözüm önermektir. Bu çalışmada kullanılan veriler, karmaşık filtreleme adımlarına ihtiyaç duyulmadan işlenebilmiştir. Bu durum her veri için aynı olmamaktadır. Verilerin ham hallerinde yön bilgisi olup olmadığı, eğer farklı hatlar bir arada işletiliyor ise, hat bilgisinin veri içerisinde mevcut olup olmadığı, araçların sefer numaraları gibi bilgilerin varlığı ya da yokluğu verilerin işlenmesinin zorluğunu belirleyen ana etkenlerdir. Tablo 2.3 incelendiğinde elimizdeki veri kümlerinde bu bilgilerin olmadığı görülmektedir. Minimum veri kaybı ile uygun filtreleme adımlarına karar vermenin bu çalışmalardaki en kritik konulardan olduğu aşikârdır. Bu nedenle çalışma kapsamında, ayrıca, büyük veri sınıfına giren GPS kayıtlarının işlenmesi, filtre edilmesi ve eksik bilgilerin veriye işlenmesine yönelik genelleştirilmiş ilkeler önerilmeye çalışılacaktır.

Bölüm 2.2'deki Tablo 2.2'de verilen bilgiler incelendiğinde, mesafe sütunun kullanılamaz olduğu görülmüştür. Normal şartlar altında, Metrobüs araçlarının şoförleri her yeni sefere başladıklarında önlerindeki otomasyon sisteminde bunu manuel olarak işaretlemek zorundadırlar. Fakat bu işlem düzenli yapılmadığı için birçok veri kümesinde mesafe sütunundaki değerlerin sıfırlanmadan arttığı görülmüştür. Verilerde eksik bilgiler olan yön, hat, sefer numarası (incelenen aralık için), kesim numarası (bu konu detaylıca Bölüm 3.2 ve 3.3'te anlatılacaktır) bilgilerinin atanması yapılmalıdır.

Hat Ataması

Hatırlanacağı üzere İstanbul'daki Metrobüs hattında Tablo 2.2'de adları ve güzergâhları ile beraber görülebilen 8 ayrı hat işletim halindedir. Bu hatlar içerisinden 34G hattı sadece gece saatleri tüm Metrobüs kesimi boyunca hizmet vermektedir. Bunun dışında diğer 7 hat gün içinde birlikte hizmet vermektedir. Analizler ve diyagramlar zirve saatler için yapılacağı ve çıkarılacağı için, bir

aracın bu belirlenen zaman aralığında Tablo 2.2'deki hatlardan sadece birinde hizmet vereceği kabul edilmiştir. Gün içerisinde araç bazındaki hat değişimi sık rastlanan bir durum olmasa da gerçekleşebilmektedir. Fakat genel olarak incelemeler sabah zirve saatlerinde (07:00 – 10:00) yapıldığı için, bu süre zarfında hat değişimi yapılmadığı kabulü uygun gözükmektedir. Sabah zirve saatler hattın gün içinde en yoğun olduğu zaman iki zaman aralığından biridir. Geliştirilecek olan algoritma bu zaman aralığındaki işletim şartları ile uyumlu çalışabilmesi gereklidir. Bu nedenle incelencek zaman aralığı sabah zirve saatler olarak seçilmiştir.

Hat bilgisi olmayan veri kümelerinde, hat bilgisi atamak için başlangıç ve son duraklara varış şartları kullanılarak ilgili kapı numarasının incelenen zaman aralığında hangi güzergâhta hizmet verdiği anlaşılabilir ve ilgili atama veriye yapılabilir. Geliştirilen hat atama algoritmasının adımları şöyle sıralanabilir:

i) Olası tüm terminal bölgelerinin enlem boylam bilgileri kaydedilir.
ii) Tüm veriler araç kapı numaralarına göre ayrıştırıldıktan sonra, her bir kapı numarasına ait sinyaller toplu bir şekilde hat atama fonksiyonuna gönderilir.
iii) Hat atama fonksiyonu, ilgili kapı numarasının sinyallerinin başlangıç son duraklarını, mevcut tanımlı terminal bölgelerinin koordinatları ile karşılaştırarak inceler.
iv) İncelemeler sonucu uygun hat kodu ham veriye işlenir.

Yön Ataması

Yön bilgisi, ön işleme aşamaları arasında en kritik öneme sahip olan adımdır. Yön atamasının yanlış yapılması, veri satırlarının karışmasına ve diyagramların beklenen sonucu vermemesi ile sonuçlanacaktır. Yön ataması yaparken incelenen sistemin dinamikleri göz önüne alınmalı ve kullanılacak algoritmanın sistemin hangi dinamiklerinden etkilenebileceği iyi düşünülmelidir. Her bir veri satırına yapılacak yön ataması eğer bir araç için birbirini takip eden konum verilerinin karşılaştırılması ile yapılacaksa, araçların atanacak yönde beklenmeyen bir manevra yapması algoritmayı bozuma uğratacaktır.

İstanbul'daki Metrobüs hattı için iki yön mevcut olup bu yönler, Anadolu Avrupa yönü ve Avrupa Anadolu yönüdür. Çalışma kapsamında önerilen ve kullanılan yön ataması iki aşamada yapılmaktadır. Yön ataması yapılmadan önce;

- Hat bazında başlangıç son duraklarının olduğu satırlara yön ataması yapılabilir. Örneğin 34BZ hattı için aracın Zincirlikuyu durağına vardığı zaman, bu andan itibaren diğer yöne hareket edeceği göz önüne alınırsa, Anadolu Avrupa yönü için bir kod atamak uygun olacaktır.
- Başlangıç son durakların olduğu satırlara yön ataması yapıldıktan sonra, basit bir algoritma ile farklı iki yön kodu arası ilgili değerler sırasıyla olduğu gibi kopyalanabilir. Örneğin; Belirli bir yön ataması yapılmış satırdan, diğer yön atamasının olduğu satıra kadar ilk yön bilgisi kopyalanabilir.

Bu iki basit adım ile basit manevralardan etkilenmeyecek bir yön algoritması oluşturulabilir. Oluşturulacak algoritma ne olursa olsun, eldeki verinin yapısı göz önünde bulundurularak bir algoritma oluşturmak ise en doğrusu olacaktır. İlgili algoritmanın adımları şöyle sıralanabilir:

i) Her bir hat için, her iki yönde de başlangıç son duraklarının koordinatları enlem boylam çiftleri olarak bir fonksiyon içerisinde tanımlanır.

ii) Her kapı numarasına ait veriler sırasıyla yön atama fonksiyonuna gönderilir.

iii) Yön atama fonksiyonu, verideki hat koduna göre, belirli bir yönde son durağa ulaşmış olan veri satırlarına diğer yön (Anadolu Avrupa için 0 kodu, Avrupa Anadolu için 1 kodu) için bir kod ataması yapar. Bu şekilde tüm başlangıç son durak satırlarında diğer yön için bir kod ataması yapılmış olur.

iv) Fonksiyon daha sonra her yön kodunu, ilgili yöndeki son durağa kadar veri satırlarına kopyalar. Bu şekilde tüm veri satırlarına yön bilgisi işlenmiş olur.

Sefer Numarası Atanması

Yol-zaman diyagramları sefer bazında çıkarılması gerektiği için, incelenen zaman aralığında aracın seferleri numaralandırılmalıdır. Sefer numarası normal şartlar altında her yeni sefer başlangıcında şoför tarafından otomasyon sistemi üzerinden işaretlenmesi ile veriye atanmaktadır. Fakat eldeki ham veride bu bilgi yok ise başlangıç son duraklarına varış şartı kullanarak seferler numaralandırılabilir. Örneğin, işletilen yönlerin herhangi birinde seyrini tamamlayan araçların, sefer sonunda aksi yönde harekete başlaması beklenir. İlgili araçların verileri zamana göre sıralandığında, incelediğimiz yöndeki (örneğin sadece "1" yönü) bir önceki sefer ile bir sonraki sefer arasında bir zaman ve mekân farkı anında veri içerisinde kendini belli edecektir. Bu zaman ve mekân farkının ardışık veri satırlarında algılanması ile kolaylıkla seferlerin numaralandırılması yapılabilir. İlgili algoritmanın adımları şöyle sıralanabilir:

i) Her kapı numarası, hat numarası atanmış bir biçimde, yönlere göre ayrıştırılır. Her aracın belirli bir yöndeki tüm verileri bir arada tutulur.

ii) Araçların seferlerinin bitişi ile, aynı yöndeki diğer seferi arasında ciddi bir süre olacaktır (Örneğin Beylikdüzü – Zincirlikuyu yönündeki aracın, tekrar aynı yönde sefere başlaması için, öncelikle Zincirlikuyu – Beylikdüzü yönünde bir sefer tamamlaması gerekecektir). Bu zaman farklarının gerçekleştiği veri satırları belirlenir.

iii) Bu satırlar sefer numaralarının değişeceği satırlar olacağına göre, veri setleri ilk satırdan itibaren, adım ii'de belirtilen veri satırına kadar aynı sefer numarasına sahip olacak şekilde numaralandırılır. Değişimin gerçekleşeceği veri satırlarında ise sefer numarası bir birim arttırılarak numaralandırmaya devam edilir.

Verilerin Filtre Edilmesi

Yol-zaman diyagramlarının, incelenen yol kesimini kullanan araçlar için bir grafik oluşturması beklenir. Bu nedenle elimizdeki veriler içerisinden bazı filtrelemeler yapmak uygun olacaktır. Bu filtrelemeler basitçe şu şekilde sıralanabilir:

- Araçlar sefer halinde değilken, terminallerdeki beklemelerinin grafiklere dâhil edilmemesi.
- Garajlardaki beklemelerin grafiklerde gözükmemesi için garajların bulunduğu enlem ve boylamlardan gelen verilerin filtre edilmesi.
- Araçların terminallerden sefere başlarken bazı servis yollarını kullanması ve bu durumun yol zaman diyagramlarında fazladan mesafe kat edilmiş gibi gözükmemesi için, terminallerden çıkışından itibaren mesafe bilgilerinin atanması ve grafiklerin çizdirilmeye başlanması uygun olacaktır. Aksi takdirde terminal içi dönüş manevrası veya servis yollarını kullanım sebebiyle fazladan mesafe kat edilmiş görüntüsü oluşacaktır.
- Ayrıca çalışma kapsamında geliştirilen algoritma, Metrobüs sisteminin sadece Beylikdüzü-Zincirlikuyu kısmı için oluşturulmuş olup, Boğaziçi köprüsü ve Anadolu yakası tarafındaki kesimler dahil edilmemiştir. Bunun temel sebebi, Boğaziçi Köprüsü üzerinde Metrobüs sistemi karışık trafikte seyir etmekte ve koridorun dinamiklerinden farklı bir davranış göstermektedir. Bu nedenle çalışma kapsamındaki algoritma sadece karışık trafikten izole bölgeler için uygun çalışabilmektedir. Beylikdüzü-Zincirlikuyu arası için oluşturulan algoritma, kolaylıkla Boğaziçi Köprüsü-Söğütlüçeşme durakları arasındaki izole kesim içinde kolaylıkla oluşturulabilir.

Kesim Numarası Atama

Yol–zaman diyagramlarının çıkarılması için araçların sefer bazında kat ettikleri mesafelerin zamana göre değişimlerinin gösterilmesi gerekmektedir. Bu sebeple her bir GPS sinyalinin, yol koridoru üzerindeki konumunun, bir referans başlangıç noktasına göre, kaçıncı kilometrede olduğu bilgisi veriye işlenmelidir. Bunu yapabilmek için geliştirilen yöntem Bölüm 3.2'de anlatılacaktır. Her bir GPS sinyalina kesim numarası atanması, karakteristik hızların belirlenmesi içinde veriyi ayıklama imkânı verecektir. İlgili yöntemi uygulayabilmek için tüm yol koridoru yaklaşık 100-150 metre arasında uzunluklara sahip kesimlere ayrılmıştır ve kesimler numaralandırılmıştır. Kesim uzunluklarının seçilmesi için

geçmiş literatürdeki kesimlere ayırma işlemleri incelenmiştir. Bu incelemeler sonucu başlangıç olarak 100 metrelik kesim uzunlukları seçilmiştir. Fakat bazı istasyonların boyutları 100 metreden daha uzun bir yapıya sahip olduğundan bu bu bölgeleri iki kesimde incelememek için peronlar uzunluklarından bağımsız olarak tek kesim olarak incelenmiştir. Sonrasında ise istasyon bölgesinin girişinde ve çıkışında 100 metreden daha küçük kesimler ortaya çıkmış ise bu kesimleri ise bir önceki veya bir sonraki kesimle birleştirerek tekrar bir kesimleme işlemi yapılmıştır. Kesim uzunluklarının en iyilenmesine yönelik bir çalışmaya Bölüm 5'te değinilecektir. İlgili algoritmanın adımları şöyle sıralanabilir:

i) Bölüm 3.2'de anlatılan yöntem ile elde edilen referans matrisi kullanarak yol kesimi yaklaşık 100-150 metre uzunluğunda kesimlere ayrılmıştır. Her bir kesimin enlem boylam bilgileri ayrı bir matrise işlenmiştir.

ii) Ön işlemeden geçirilecek olan verilerin, bu matris içinde hangi kesime düşmekte olduğu belirlenir.

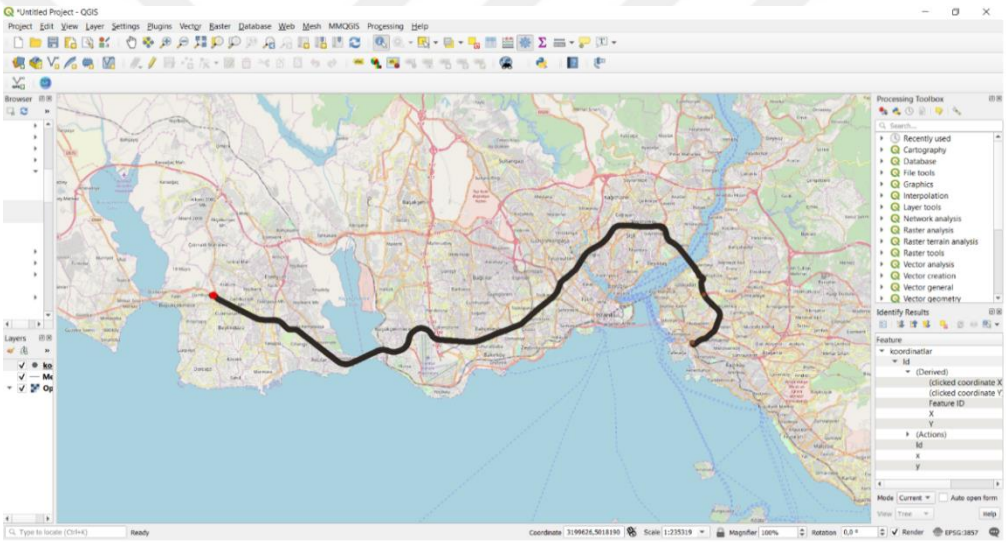
iii) Belirlenen kesim numarası, ham GPS verisine her bir satır için işlenir.

3.2 Mesafe Atama Algoritması

Bölüm 3.1'de kesim numarası atanması kısmında bahsedildiği üzere, ilgili kesim numaralarının verilere atanmasının temel sebebi, kesim bazında bazı karakteristik değerlerin belirlenmesi gerekliliğidir. Bu kesim numarasının atanması için yapılmış olan işlemleri şu şekilde sıralayabiliriz:

- QGIS ya da herhangi bir GIS (Coğrafi Bilgi Sistemleri) programı üzerinden, ilgili koridorun harita dosyası 'Şekil Dosyası' (Shapefile) formatında çekilir.
- İlgili dosya üzerinden, incelenecek olan koridorun, ilgili şeritlerinden bir çizgi geçirilmek suretiyle, tüm yol boyunca sürekli bir poligon oluşturulur.

- Birçok GIS programında olan ‘Vektör Araç Kutusu’ kullanılarak, çizilmiş olan poligonun üzerine, seçilmiş olan belirli bir mesafe aralığına göre düğüm noktaları atanır. Bu çalışma kapsamında 10 metre aralıkla, yaklaşık 50 km’lik (servis yolları ve terminallerin iç yolları hariç) metrobüs hattı için 5049 düğüm noktası elde edilmiştir. Buradaki 10 metrelik aralık seçimi deneme yanılma şeklinde oluşturulmuş olup daha düşük aralıkla daha fazla referans noktası elde edilebilir. Bu durumun grafiklerde iyileştirme yapması fakat verinin işlenme süresini arttıracağı unutulmamalıdır.
- Tüm atanmış olan düğüm noktalarının enlem boylam bilgileri, düğüm noktası numaraları ile beraber bir excel dosyası olarak kaydedilir.



Şekil 3.1 QGIS programı ve referans poligon

- Tüm düğüm noktaları belirli bir başlangıç noktasına göre sıralanır. Düğüm noktalarının arasındaki mesafe biliniyorsa, kümülatif olarak tüm düğüm noktalarının kaçıncı kilometrede olduğu bilgisi bulunur.
- Daha sonra kullanılan programlama diline göre bir arama fonksiyonu yazılarak, mevcut yollanmış olan sinyalin, oluşturulmuş olan referans koordinat matrisindeki hangi enlem boylam çiftine yakın olduğu bulunarak, ilgili satırdaki mesafe bilgisi ham veri satırına çekilebilir.

Tablo 3.1 Mesafe ataması için QGIS programından alınan referans matris örneği

ID	Boylam	Enlem	Mesafe_1 (metre)	Mesafe_0 (metre)
5	28,62666	41,02219328	270	50230
6	28,62677	41,02215853	280	50220
7	28,62688	41,02212377	290	50210
8	28,62698	41,02207802	300	50200
9	28,62709	41,02203106	310	50190
10	28,62719	41,0219841	320	50180
11	28,62729	41,02193714	330	50170
12	28,62739	41,02189018	340	50160
13	28,62749	41,02184322	350	50150
14	28,62759	41,02179433	360	50140
15	28,62769	41,02174399	370	50130
16	28,62779	41,02169077	380	50120
17	28,62788	41,02163644	390	50110

Bu arama fonksiyonunun çalışma adımları aşağıdaki gibidir:

i) Tablo 3.1'de örnek olarak verilmiş olan referans matrisi incelenirse, her bir düğüm noktasının enlem boylam bilgisi ilgili sütunlarda verilmiştir.

ii) Her bir GPS verisindeki enlem boylam çifti kendisine en yakın 3 enlem boylam çiftini referans matrisinde bulur.

iii) Daha sonra Haversine adı verilen bir jeodezi formülasyonu kullanarak, GPS verisindeki enlem boylam çifti ile adım ii'de bulunan her bir enlem boylam çifti ile arasındaki kuş uçuşu mesafe hesaplanır.

iv) Kuş uçuşu mesafenin en küçük çıktığı referans değerlerin satırlarındaki mesafe bilgisi GPS verisine işlenir.

İlgili adımlar takip edilirse, her bir GPS sinyali Metrobüs koridoru üzerindeki, başlangıç durağına göre rölatif konumu metre cinsinden kolaylıkla bulunabilir. Adım 3'de bahsedilen haversine formülasyonu Chopde ve Nichat (2013) yaptığı çalışmadan alınmıştır. İlgili çalışmada belirli yarıçapa sahip bir küre üzerinde bulunan iki nokta arasındaki en yakın mesafe ilgili formülasyon ile hesaplanmıştır. Benzer bir kabul ile iki GPS verisi arasındaki mesafe de bu formülasyon ile yüksek doğrulukla hesaplanabilmektedir. Haversine formülasyonu denklem 2'deki gibidir. Denklem 2'deki d, mesafeyi temsil etmektedir ve bir fonksiyon olarak ifade edilmektedir. φ_1, γ_1 ise ilk enlem, boylam çiftini ifade ederken, φ_2, γ_2 ise ikinci enlem boylam çiftini ifade eder. Denklemdeki r ifadesi ise Dünya'nın yarıçapını temsil etmektedir.

$$d(\varphi_1,\gamma_1,\varphi_2,\gamma_2) = 2r.\arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2-\varphi_1}{2}\right)+\cos(\varphi_1).\cos(\varphi_2).\sin^2\left(\frac{\gamma_2-\gamma_1}{2}\right)}\right) \quad (2)$$

3.3 Kaza-Arıza Algılama Algoritması

Çalışma kapsamında geliştirilen kaza-arıza algoritması, GPS tabanlı bir algoritmadır. Tüm Metrobüs filosunda hali hazırda bulunan GPS vericileri, 10-60 saniye arasında değişen periyotlarla İETT veri tabanına konum bilgileri yollamaktadır. Bu veriler İETT tarafından temin edilmiştir. İlgili veri kümeleri kullanılarak, Metrobüs sistemi analiz edilmiş ve algoritma geliştirilmiştir. Analiz sonuçları Bölüm 4'te incelenecektir. Bu analiz sonuçları algoritmayı şekillendirmiş olup, temel olarak sisteme ek detektör maliyeti getirmeyecek, eldeki veri altyapısı üzerinden, kaza-arızaları yüksek doğrulukta ve kısa sürede algılayabilecek bir algoritma geliştirilmesi amaçlanmıştır. Algoritma ile ilgili temel bazı kavramlar aşağıda açıklanacaktır.

Hız–Zaman–Kesim Matrisleri ve Hız Haritaları

Algoritma kapsamında önemli bir yeri olan bu matrisler, Bölüm 3.2'de anlatılan kesimlere ayırma işlemindeki kesim numaralarını kullanarak çalışmaktadır. Numaralandırılan her yol kesimi için, önceden seçilmiş ölçüm aralığında hız ortalamaları alınarak matris doldurulur. Bu matrisler daha sonra bir kontür grafiği olarak görselleştirildiğinde ortaya hız haritaları çıkarılmış olur. Hız haritaları detaylı şekilde 4. Bölümde analiz edilecektir. Tablo 3.2'de ise örnek bir hız–zaman–kesim matrisi görülebilir.

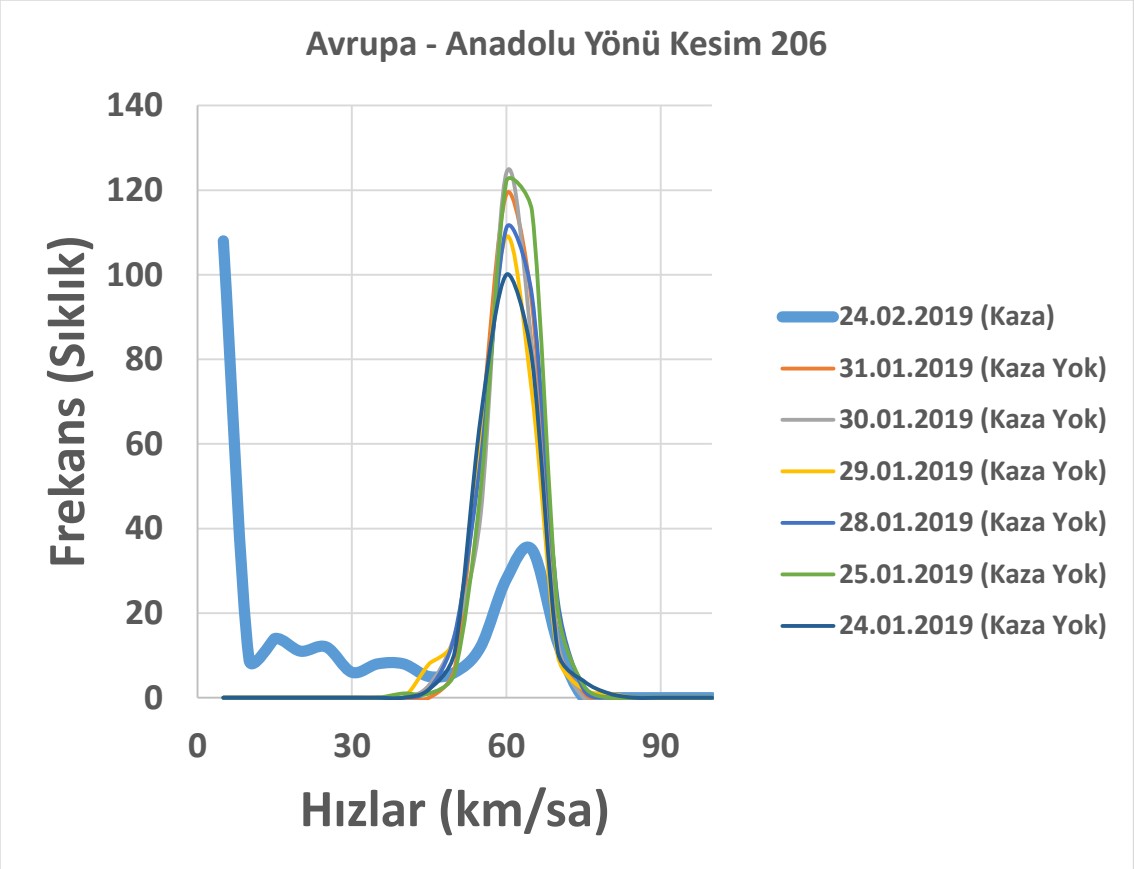
Tablo 3.2 Örnek hız–zaman–kesim matrisi (hızlar, km/sa)

	Kesim Numaraları						
Zaman (saniye)	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0-900	42,90	59,25	59,75	62,00	61,33	67,00	64,60
900-1800	39,50	57,50	61,33	Na	67,57	69,80	69,29
1800-2700	44,71	54,14	61,67	64,00	67,40	66,57	68,83
2700-3600	43,50	56,38	63,00	68,00	67,00	67,40	69,14
3600-4500	45,75	52,25	61,50	Na	68,50	69,50	70,00
4500-5400	42,50	56,50	62,43	Na	68,40	69,17	70,00
5400-6300	43,63	53,75	61,17	61,33	68,75	63,00	67,38

Tablo 3.2'de Zaman sütunu saniye cinsinden olup, belirli bir başlangıç zamanından (örneğin 07:00) itibaren geçen süreyi saniye cinsinden ifade etmektedir. Yatayda verilen en üst satır ise kesim numaralarını ifade etmektedir. Bu matrisler çeşitli ölçüm aralıklarına göre oluşturulabilir. Çalışma kapsamında birbirinden farklı ölçüm aralığında matrisler kullanılmıştır. Tablo 3.2'deki matris örnek olarak verilmiş olup, kullanılan ölçüm aralığı 900 saniyedir (15 dakika).

Karakteristik Hız Matrisleri

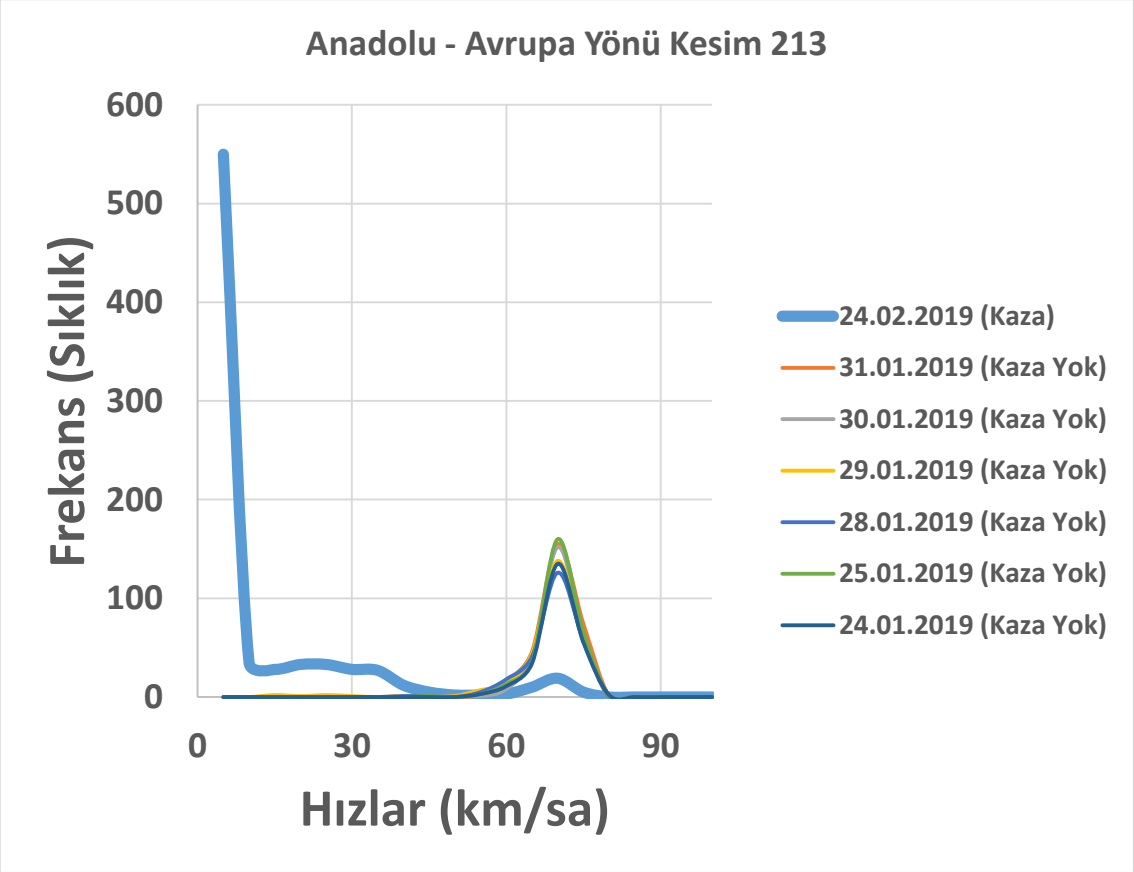
Algoritmanın algılama sistematiği içerisinde kullanılmak üzere, daha önce numaralandırılmış olan kesimler için bir karakteristik hız değeri tayin edilir. Bu değerler geçmişteki verilerin incelenmesi ile oluşturulabilir. Verilere göre belirli genişlikte (5 km/sa) hız sınıfları oluşturulur. Tüm kesimler için zirve saatlerde ilgili hız sınıflarında yollanmış olan her bir hız verisi için, ilgili sınıfın frekans değeri bir arttırılır. Bu sayede hızların kesim bazında istatistiki dağılımları incelenerek, gerçekleşmiş olan hız dağılımlarının alt uçlarındaki hız değerleri, ilgili kesimin normal şartlardaki en düşük karakteristik hız değeri kabul edilmiştir.



Şekil 3.2 206 numaralı kesim için ortalama hızların dağılımı

Şekil 3.2'de görüldüğü gibi, 206 numaralı kesim (24 Şubat 2019 tarihli GPS verisine göre ilgili kaza bu kesimde gerçekleşmiştir) için normal şartlar altında

hız değerleri 40 – 80 km/saat arasında bir davranış göstermektedir. İlgili kesimdeki taşıt akışı kaza günü ise, durma noktasına gelmiştir.



Şekil 3.3 213 numaralı kesim için ortalama hızların dağılımı

Benzer şekilde diğer yön için, 213 numaralı kesim benzeri bir davranış göstermektedir. Bu iki grafik bizlere kaza durumlarında ve normal şartlarda kesimlerin nasıl davranacağına birer örnek teşkil etmektedir.

Tablo 3.3 Örnek karakteristik hız matrisi

Kesim	Hız_1 (km/sa)	Hız_0 (km/sa)
3	32,9	36,1
4	38,6	45,0
5	42,8	41,1

Bu grafikler, bizlere bir kesim için seçilecek karakteristik hız matrislerinin oluşturulmasında kullanılacak yaklaşımın nasıl olması gerektiği konusunda fikir vermektedir. Kaza-arıza olmayan günlerde, hızların dağıldığı aralığın alt ucundaki (sol kuyruk) hız değerleri, bu kesim için gerçekleşebilecek hızların en küçüğünü temsil etmektedir. Algoritma içerisindeki kontrolü bu hız değerine göre yapmak uygun olacaktır.

Elde edilen kaza günlerine ait olmayan tüm hafta içi günü veri setleri için, sabah 07.00–10.00 arasında tüm kesimler için hız dağılımları çıkartılmıştır. Daha sonra hız dağılımlarının alt ucundaki hız değeri karakteristik değer olarak seçilmiştir. Bu ortalama değerler ilgili kesimin karakteristik hızları olarak seçilmiştir. Tablo 3.3'de örnek karakteristik hız matrisi verilmiştir. Sütun isminde "1" karakterini barındıran değerler, Avrupa Anadolu yönünü temsil ederken, "0" karakterini barındıran değerler Anadolu – Avrupa yönünü temsil etmektedir.

Takip-Uyarı Matrisleri

Bu matris karakteristik hız değerleri ile GPS sinyallerindeki hız değerlerinin kesim bazında karşılaştırılması sonucu, belirli bir kesimde karakteristik hızdan önemli ölçüde aşağı yönlü sapma gösteren kesimlerin zaman içerisindeki takipleri için oluşturulan bir matristir. Karakteristik değerlerden aşağı yönlü ciddi sapma olduğunda, ilgili kesim için, ilgili zaman adımında "1" değeri matriste ilgili hücreye işlenmektedir. Aksi durumda ise "0" değeri matrise işlenmektedir. Bu şekilde "1" ve "0" değerleri aynı kesim içerisinde, zamana göre takip edilebilmektedir. Karakteristik değerlerden sapmanın ölçüsü olarak çalışma kapsamında deneme yanılma şeklinde bir yöntem izlenmiştir. Karakteristik hızdan küçük eşit şeklinde bir şart kullanmak, kaza-arıza olmayan durumlarda da algoritmayı yanlış algılamaya yöneltme durumu oluşturabileceği öngörülmüştür. Bu nedenle sapma miktarı %50 olarak seçilmiştir. Örneğin; Belirli bir zaman aralığında herhangi bir kesimde karakteristik hızın %50'sinden daha küçük bir ortalama hız gözlendiyse, takip-uyarı matrisinde ilgili hücreye "1" ifadesi işlenebilir. Tablo 3.4'te örnek bir Takip – Uyarı matrisi verilmiştir. Bu takip uyarı matrisinde, ilgili zaman adımı için ilgili kesimdeki ortalama hız,

karakteristik değerine eşit veya büyük ise 0, karakteristik değerden önemli ölçüde (En az %50'si kadar) küçük ise 1 kodu işlenmiştir. Eğer ilgili zaman adımında, ilgili kesimde hiç GPS verisi yollanmamış ise "Na" (Not Available – Bulunamıyor) ifadesi olarak işlenmiştir. Tüm bu matrisler kullanılarak algoritmamız üç aşamalı olacak şekilde kaza–arıza algılamaya çalışmaktadır.

Tablo 3.4 Örnek takip-uyarı matrisi

	Kesim Numaraları			
	2	**3**	**4**	**5**
t1	0	Na	0	Na
t2	Na	Na	Na	Na
t3	0	Na	Na	Na
t4	0	0	0	0

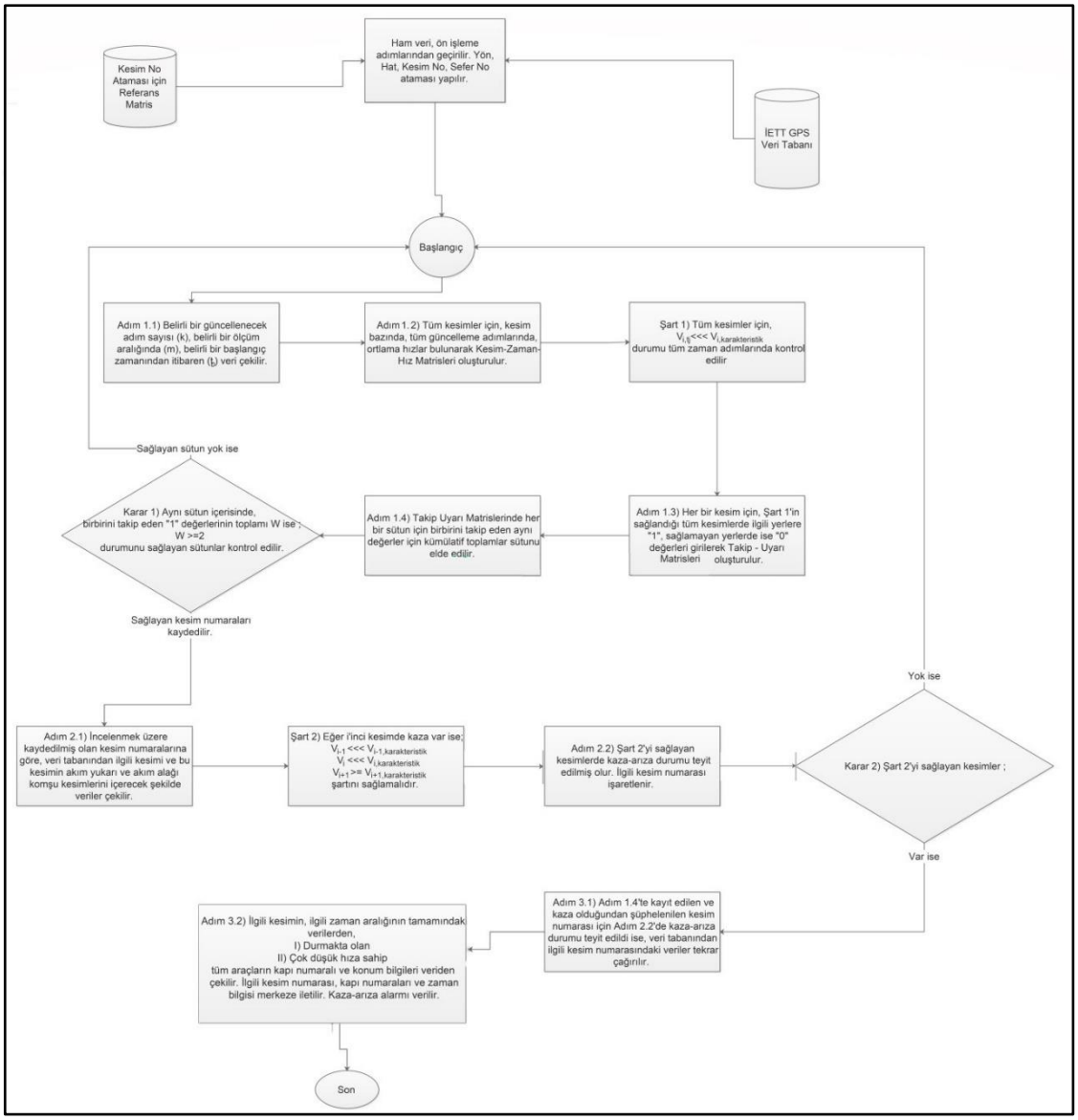
Adım 1.1) Seçilen belirli bir ölçüm aralığı için (interval), seçilen belli bir güncelleme adımı sayısına göre, incelenen zaman aralığı boyunca Bölüm 3.1'de bahsedilmiş olan ön işleme adımlarından geçirilmiş olan ham veri kümeleri veritabanından çekilir.

Adım 1.2) Bu adımda Hız–Zaman–Kesim matrisleri oluşturulur. Bu matrisler tüm güncelleme adımları boyunca, tüm kesimler için ayrı ayrı olacak şekilde ortalama hız değerlerini içerir. Bir kesim içerisinde birden fazla araç var ise, GPS sinyallerindeki hız değerlerinin ortalaması alınır ve ilgili matris hücresine işlenir.

Adım 1.3) Daha öncesinde oluşturulmuş olan Karakteristik Hız Matrisi kullanılarak, oluşturduğumuz Hız – Zaman – Kesim matrislerindeki her bir değer, ilgili karakteristik değeri ile karşılaştırılır. Eğer hız değerleri, karakteristik değerlerinden önemli ölçüde küçük ise, Takip–Uyarı matrislerinde ilgili hücreye 1 değeri girilir. Eğer hız değerleri, karakteristik hız değerlerine eşit veya büyük ise 0 değeri girilir.

Adım 1.4) Takip–Uyarı matrislerinde, kesim bazında birbirini takip eden 1 değerleri oluşursa, ilgili kesim numarası kaydedilerek adım 2.1'e geçilir. Aksi durumda Adım 1.1'e dönülür.

Adım 2.1) İncelemek üzere kaydedilmiş olan kesim numaralarına göre, veri tabanından problemlerin başladığı düşünülen zaman aralığında veri çekilir. Çekilen veriler özellikle ilgili kesimi ve bu kesimin akım yukarı ve akım aşağıdaki komşu kesimlerini içerir.



Şekil 3.4 Algoritma akış diyagramı

Adım 2.2) Herhangi bir kaza-arıza durumunda, kaza olan kesime göre akım yukarı kesimde tıkalı bir trafik gözlenirken, akım aşağı kesimde akıcı bir trafik gözlenir. Bu şart, ilgili kesim ve komşu kesimler için kontrol edilir ve kaza-arıza durumu teyit edilir. Eğer şüphelenilen kesimde kaza-arıza teyit edilemez ise Adım 1.1'e dönülür.

Adım 3.1) Adım 1.4'te kayıt edilen ve kaza olduğundan şüphelenilen kesim numarası için Adım 2.2'de kaza-arıza durumu teyit edildikten sonra, veri tabanından bu kesimlerin problemli tüm zaman aralığı boyunca verileri tekrar çekilir.

Adım 3.2) İlgili kesimdeki tüm araçlar, akım yönüne göre sıralanarak, bu kesim içerisindeki durmakta olan veya durmaya yakın olan araçların kapı numaraları, kesim numarası ile sıralarına uygun olacak şekilde merkeze bildirilir ve kaza-arıza bildirimi yapılır.

Algoritma her iterasyonundan sonra başlangıç zamanı değiştirilerek kullanılabilir. Algoritma ile ilgili akış diyagramı Şekil 3.4'deki gibidir.

3.4 Algoritmanın Sınanması

Geliştirilmiş olan algoritma, temel olarak Metrobüs koridorunda, şerit kapanmalı kazaları algılama amacıyla geliştirilmiştir. Metrobüs hattının işletmecileri, kaza-arıza durumlarına hızlı şekilde müdahale etmektedir. Fakat bu müdahale sürecinin başlaması için, öncelikle kaza-arıza durumunun bildirilmesi gerekmektedir. Algoritma geliştirildikten sonra gerçek kaza-arıza verilerinin kullanılması ile sınanmıştır. Öncelikle eldeki veriler sınıflandırılmıştır. Bu sınıflandırma temel olarak;

- Şerit kapanmalı kaza-arıza veri kümeleri.
- Kaza – arıza olmayan veri kümeleri.
- Şerit kapanmalı olmayan kaza-arıza veri kümeleri.

Olarak belirlenmiştir. 24 Şubat 2019 tarihine ait 2 adet şerit kapanmalı kaza kaydı, 4-7 Şubat 2019 tarihleri arasındaki şerit kapanmalı olmayan ufak çaptaki kaza kayıtları ve ilgili günlerin 24 saatlik GPS verileri İETT tarafından

sağlanmıştır. Doğrudan gerçek veriler üzerinden algoritma sınanmıştır. Bunun dışında kaza-arıza kaydı bulunmayan 24-31 Ocak 2019 tarihleri arasındaki günler içinde algoritma denenerek, herhangi bir yanlış algılama durumunun varlığı da test edilmiştir. Farklı özellikteki bu üç veri grubu içerisinde algoritmanın denenmesi sonucu algoritmanın farklı şiddetteki kazalara olan duyarlılığı ile kaza-arıza olmayan durumlardaki davranışı ölçülmüştür. Algoritmanın performansı Bölüm 4'te ayrıntılı olarak değerlendirilecektir.

4
ANALİZ VE SONUÇLAR

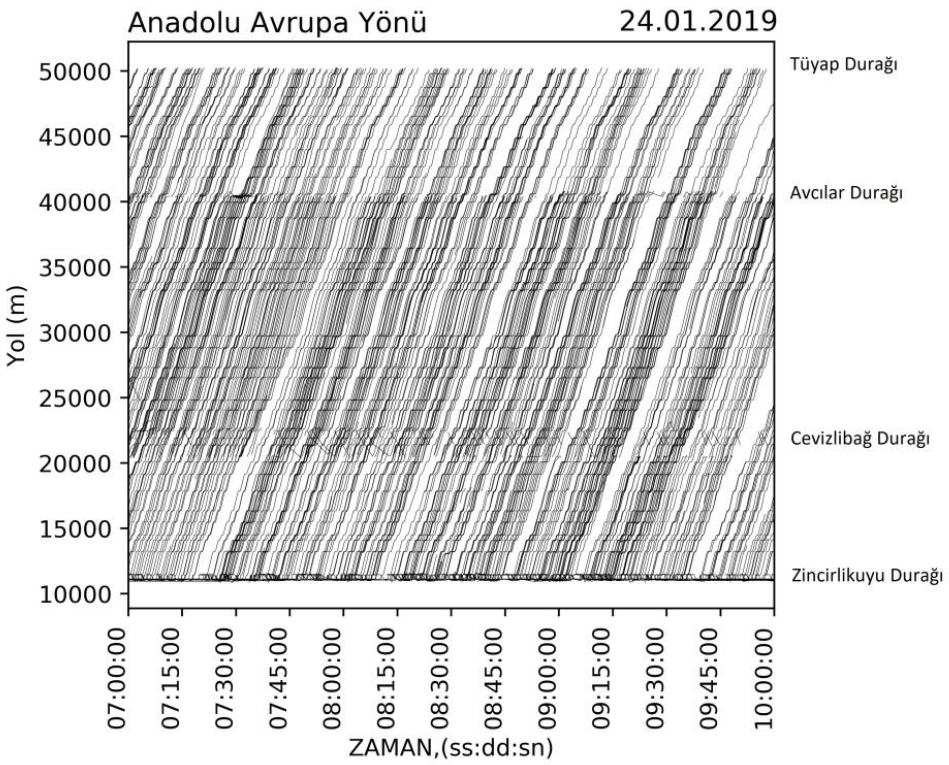
Bu bölümde Metrobüs sisteminin analizi için kullanılmış olan temel diyagramlar incelenecektir. Ayrıca sistem üzerindeki tekrarlı darboğazların varlıkları araştırılacaktır. Sistemin hafta içi ve hafta sonları için yol–zaman ve hız–zaman diyagramları çıkartılarak, doğrudan sabah zirve saatleri (07:00-10:00) için analiz edilmiştir. Bölüm, algoritmanın sınanmasını örnek içerisinde açıkladıktan sonra tüm sınamaların sonuçlarının değerlendirilmesiyle son bulacaktır.

4.1 Yol-Zaman Diyagramları

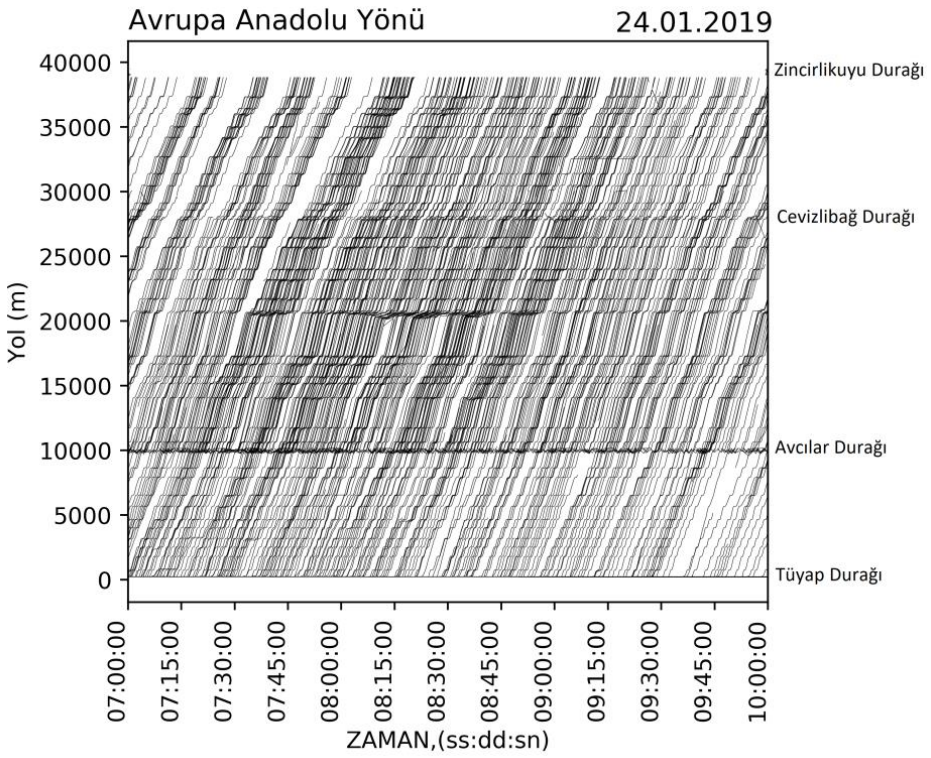
Metrobüs hattının GPS verileri, İETT tarafından “csv” (comma separated value, virgül ile ayrılmış değerler) formatında sağlanmıştır. İlgili veriler 2019 Ocak ayının son 8 gününü kapsamaktadır. Bu tarih aralığı dışında, algoritmanın sınanması için 24 Şubat 2019 tarihli kaza-arıza kayıtlarına sahip günün GPS kayıtları da temin edilmiştir. İlgili kayıtlar, öncelikle Bölüm 3.2 ve 3.3’te bahsedilen ön işleme, mesafe atama adımlarından geçirilmiştir. Daha sonrasında ise sabah zirve saatleri Beylikdüzü – Zincirlikuyu kesimi için her iki yönde grafikler çizdirilmiştir. Şekil 4.1’de ilgili tarihlerdeki yol-zaman grafiği verilmiştir. Şekil 4.1, Şekil 4.2, Şekil 4.3, Şekil 4.4, Şekil 4.5 ve Şekil 4.6’daki grafikler sadece 24-26 Ocak 2019 tarihlerine ait olup diğer günlerde birbirinin tekrarı şeklinde grafikler çıkmıştır. EK-A bölümünde diğer günlerin grafikleri verilecektir.

Grafiklerde ilk olarak göze çarpan tekrarlı bir darboğaz görülmemektedir. Fakat tüm diyagramlar, hız haritaları ile beraber incelendiğinde, hafta içi zirve saatlerde yoğunluk sebebiyle aktif olan tekrarlı bir darboğaz oluşumu gözlenmiştir. Bu darboğaza Bölüm 4.2’de değinilecektir. Grafikler oluşturulurken, Avrupa Anadolu yönü için düşey eksende 0 noktası Beylikdüzü durağını, 40000’inci metre ise Zincirlikuyu durağını temsil etmektedir. Anadolu

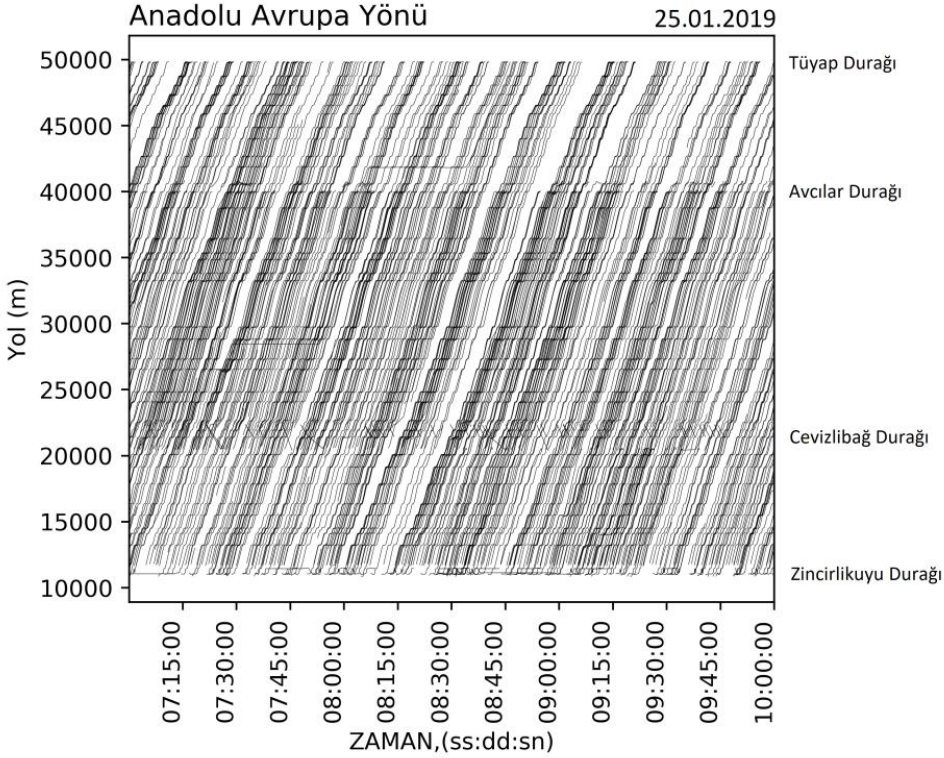
Avrupa yönünde ise 10000'inci metre Zincirlikuyu durağını ifade ederken, 50000'inci metre ise Beylikdüzü durağını ifade eder. Örnek olarak verilmiş olan yol–zaman diyagramlarının tarihleri incelendiğinde, 24–25 Ocak tarihleri hafta içi günlerine denk gelirken 26 Ocak tarihinin ise hafta sonuna denk geldiği görülecektir. Bu durum, hafta içi günü zirve saatlerde daha yoğun ve sık işletilen bir yol–zaman diyagramları görmemizi sağlarken, aynı saatlerde hafta sonu günleri için daha seyrek ve daha az yoğunluk görülmektedir. Yol-zaman diyagramlarında özellikle ara terminal (Avcılar, Cevizlibağ vd.) bölgelerinde, aykırı çizgiler görülebilmektedir. Bu aykırı ufak yörüngeler, ilgili bölgelerde araçların diğer yöne sefere başlarken istasyon etrafındaki genişlik sayesinde doğrudan dönüş manevrası yaparak diğer yönün şeridine geçiş yaptığını göstermektedir. Bu kısımlar grafiklerden çıkarılmamıştır. Bunun temel sebebi ise, bu bölgedeki verilerin yok sayılması ilgili peron bölgelerindeki beklemelerin grafiklerde gözükmemesi ile sonuçlanacaktır.



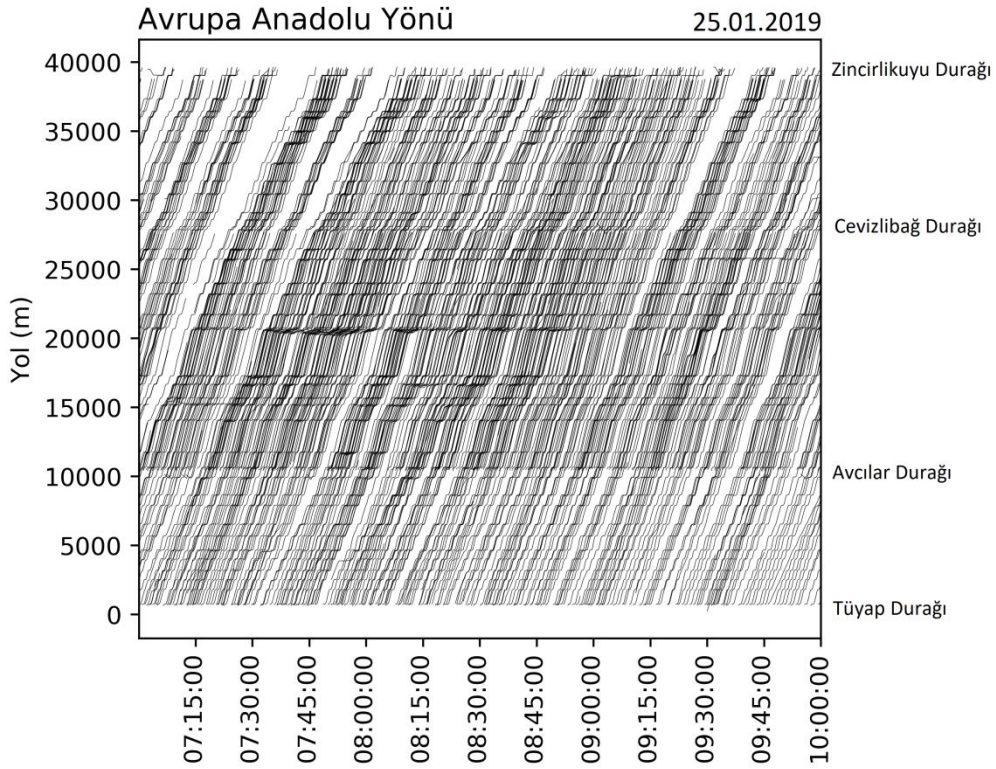
Şekil 4.1 Yol–Zaman diyagramı 1



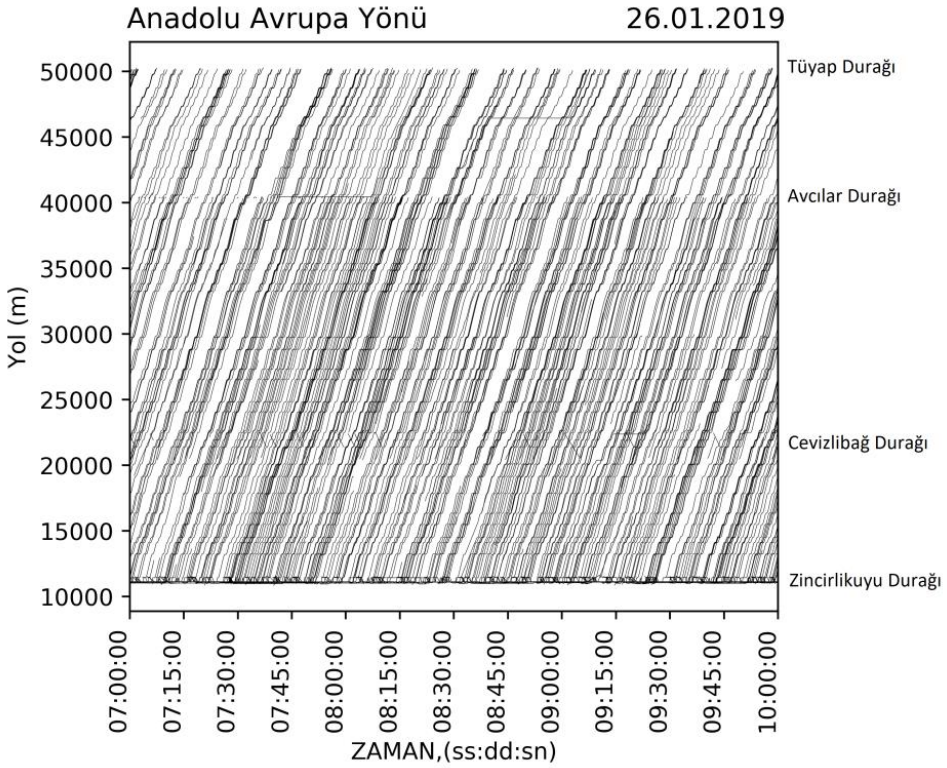
Şekil 4.2 Yol-Zaman diyagramı 2



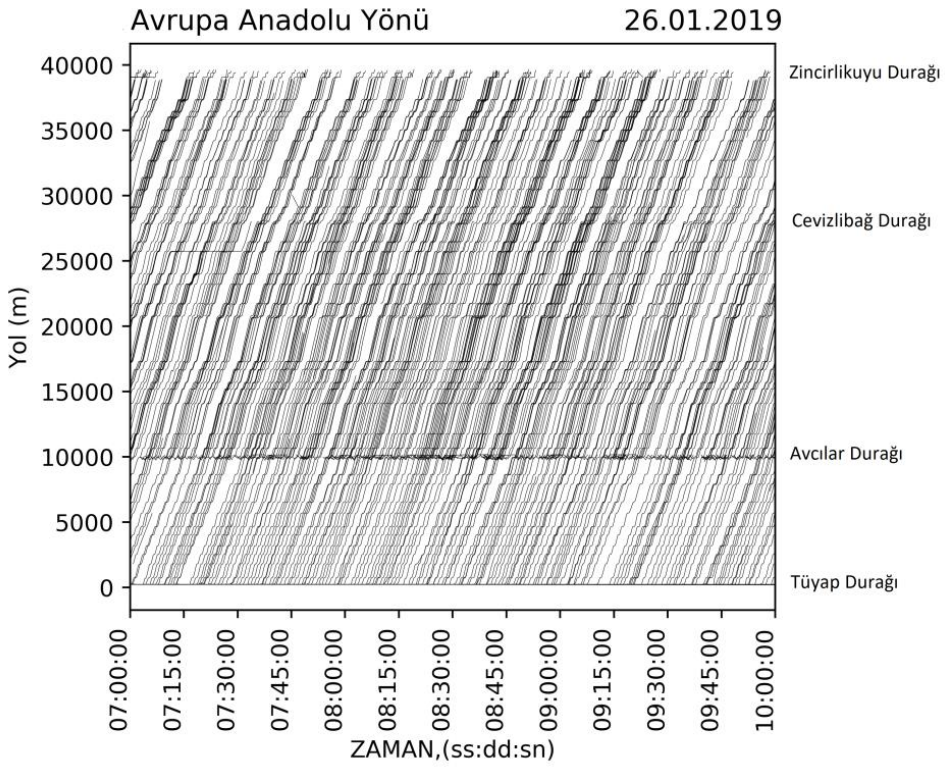
Şekil 4.3 Yol–Zaman diyagramı 3



Şekil 4.4 Yol–Zaman diyagramı 4



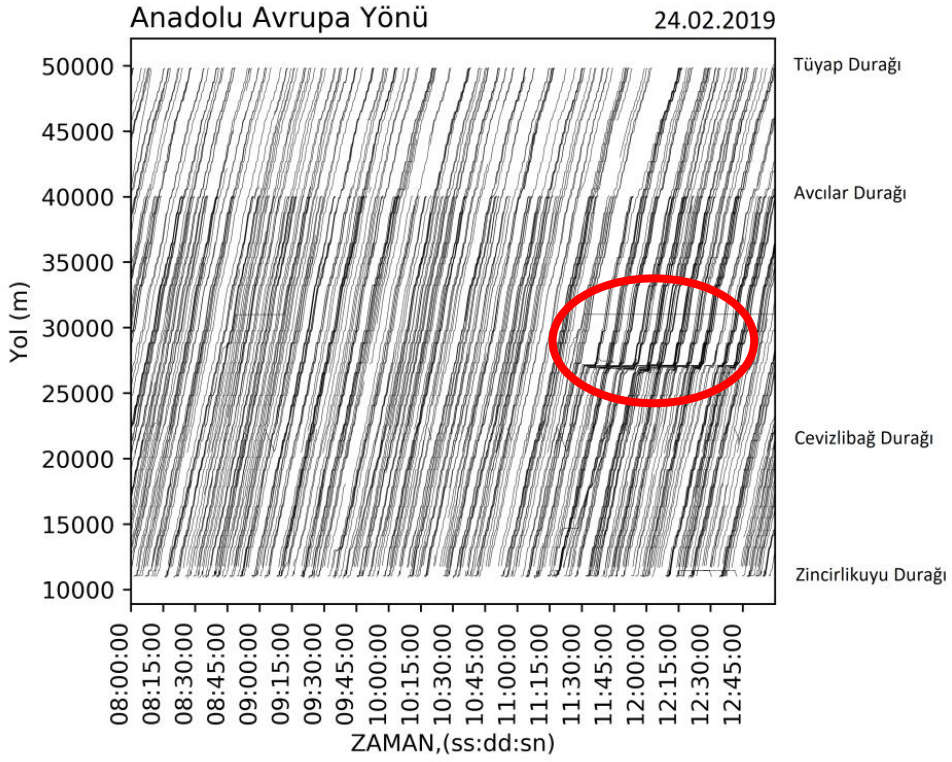
Şekil 4.5 Yol–Zaman diyagramı 5



Şekil 4.6 Yol–Zaman diyagramı 6

Bölüm 3'te anlatılan yöntemlerle oluşturulan yol–zaman diyagramlarında araç yörüngeleri büyük oranda takip edilebilmiştir. Özellikle hafta içi günlerde yoğunluk sebebiyle duraklarda gerçekleşen demetlenme (bunching) hadiseleri yol–zaman diyagramlarından görülebilmiştir. Diğer yörünge çizgilerine göre daha kalın olan yörünge çizgileri demetlenme hadiselerini ifade etmektedir. Araçların belirli bir takip aralığı (headway) ile işletildiğini düşünürsek, seyir sırasında bu takip aralığının aşıldığı ve araçların tampon tampona seyir durumuna geldiği görülmektedir. Bu tampon tampona seyir eden araçların sayısının artması ve dolayısıyla kuyruklanmanın artması demetlenme hadisesi olarak kendini göstermektedir. Hafta sonu ve hafta içi arasındaki işletim farkı ve koridor yoğunluğu yol-zaman diyagramlarından gözlendikten sonra, Bölüm 3.4'te bahsedilen karakteristik hız matrisini oluşturmak için hafta içi günlerin verilerinin kullanılması gerektiği düşünülmektedir. Zira yoğun işletim şartları ve demetlenme hadiselerinin en sık görüldüğü zaman hafta içi günleridir. Gerçekleşen hızların ortalamalarının dağılımları incelendiğinde, karakteristik hızlar için kabul edilen, dağılımın alt ucundaki hız değerlerinin, hafta için

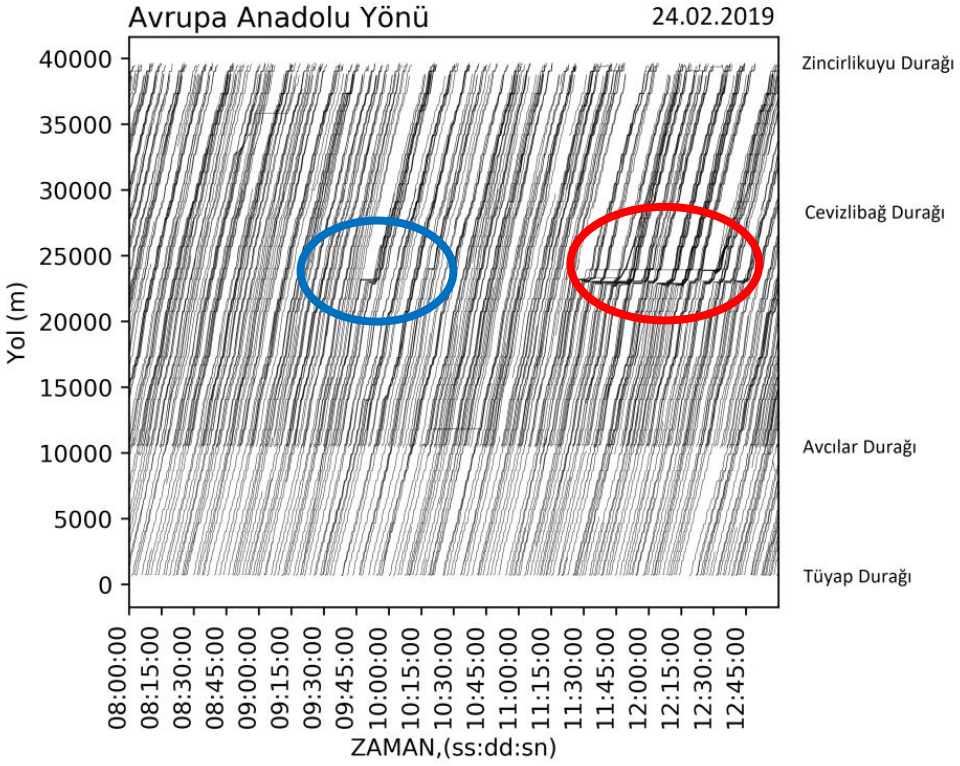
günlerde oluştuğu saptanmıştır. Genel olarak grafiklerde şok dalgaları görülmektedir, fakat bu şok dalgaların hiçbiri yüksek hıza sahip değildir ve ciddi bir yayılma göstermemektedir. Görülebilen şok dalgaları genelde demetlenme kaynaklı, peronlardaki kuyruklanma hadisesinden kaynaklanmaktadır. Kaza-arıza algoritmasının geliştirilebilmesi için sistemin kaza durumunda nasıl davrandığının incelenmesi gerekmektedir. Bunun için İETT tarafından 24 Şubat 2019 tarihli kaza kaydına binaen, o güne ait GPS verileri temin edilmiştir. İlgili güne ait yol–zaman diyagramları çıkarıldığında, sistemin kaza durumunda nasıl davrandığına dair genel bir fikir edinilmiştir (Şekil 4.7 ve Şekil 4.8).



Şekil 4.7 Yol–Zaman diyagramı–kaza kaydı 1

24 Şubat 2019 tarihli kaza kaydında, iki adet kaza olayından bahsedilmiştir. Bu kazalardan birincisi saat 09:46 civarında, Bahçelievler durağı peron içerisinde Avrupa Anadolu yönünde maddi hasarlı kaza, ikincisi ise saat 11:30 civarında yine Bahçelievler durağında, Avrupa Anadolu yönünde yaralanmalı ve maddi hasarlı bir kaza şeklindedir. Bu iki kaza-arıza durumundan ilkinde uzun şerit kapanması olmamıştır fakat etkisi kısıtlı da olsa yol–zaman diyagramlarından

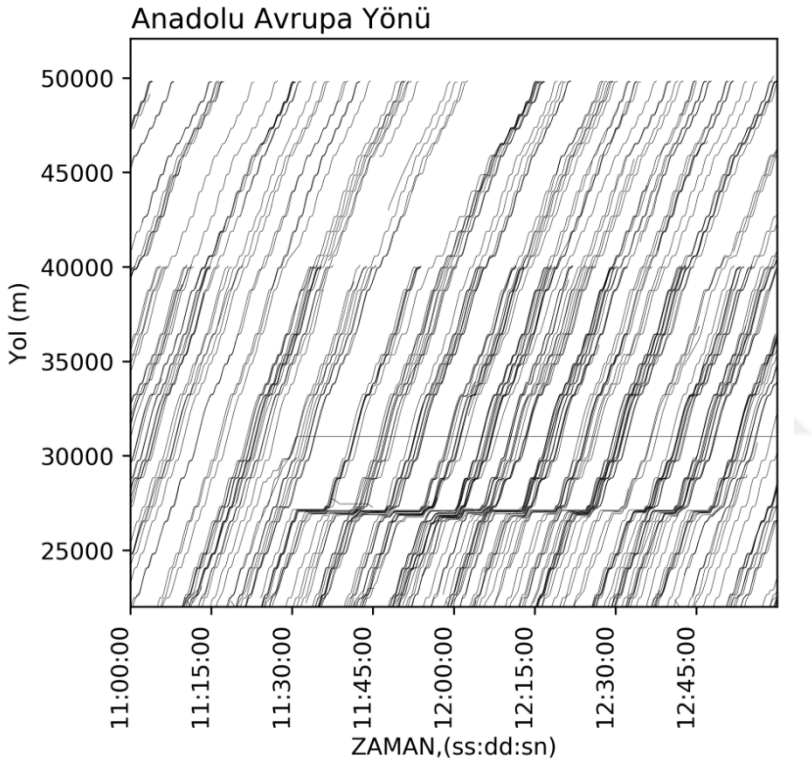
okunabilmiştir. Şekil 4.7'de (mavi elips) görüldüğü gibi Avrupa Anadolu yönünde kısa süreli bir şerit kapanması olmuş ve akım kaza yönünde yaklaşık 3-4 dakika durmuştur. Şeridin tekrar kullanıma açılması ile beraber akım tekrar devam etmiştir. Bu kazanın şerit kapatma süresi çok uzun olmaması ve ayrıca olayın Pazar günü sabah saatlerde meydana gelmesi yani sistemin çok yoğun olmadığı bir zaman aralığında gerçekleşmesi, sistemi ciddi şekilde etkilememiştir. Fakat aynı gün meydana gelen bir diğer kazanın etkileri ise kazadan uzun süre sonra bile sistemde düzensizlikler oluşturmuştur.



Şekil 4.8 Yol–Zaman diyagramı–kaza kaydı 2

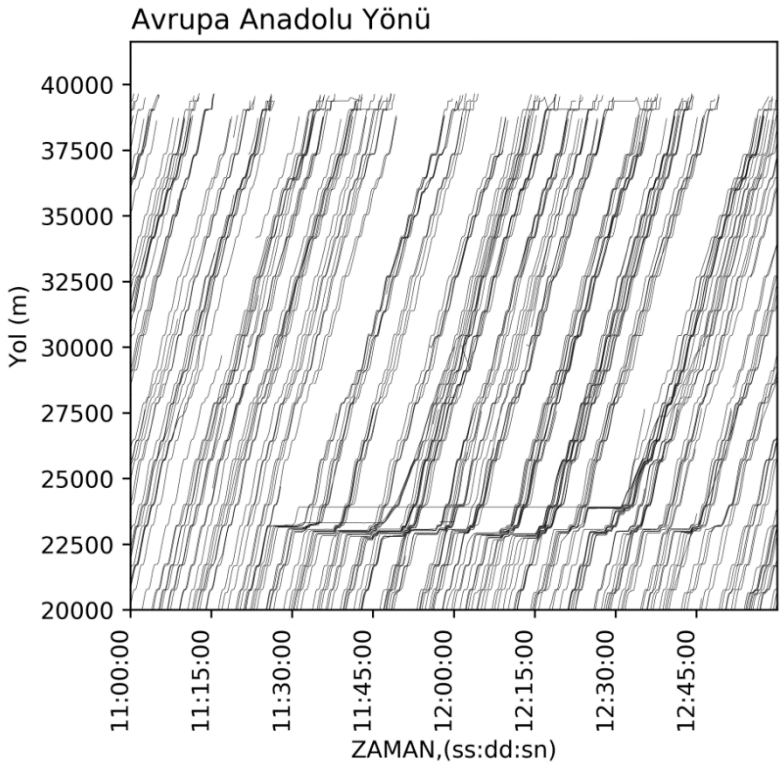
Şekil 4.6 ve Şekil 4.7'de (kırmızı elipsler) görüldüğü üzere saat 11:30 civarı olan kaza anından itibaren yaklaşık 10 dakika boyunca sistem her iki yönde durmuştur. Şekil 4.9 ve 4.10'da daha net görüleceği üzere, İETT ekiplerinin müdahalesi ile müsait olan diğer şerit üzerinden çift yönlü kontrollü geçiş yaptırılmak suretiyle trafik akımı tekrar başlatılmıştır. Burada İETT ekiplerinin ciddi bir çaba sarf ettiği açıkça görülmektedir. Şehir için büyük hayati önemi olan bir ulaştırma sisteminin benzeri bir şekilde durma noktasına gelmesinin

ciddi gecikmelere sebep olacağı açıktır. Bu kazanın etkileri her iki yönde yaklaşık 12:30-12:45 saatlerine kadar gözlenebilmektedir. Kazanın gerçekleştiği zamanı 11:30 kabul edersek bu ciddi bir yayılma süresidir. Bu nedenle kazaların hızlı algılanıp, hızlı müdahalesinin çok önemli olduğu buradan anlaşılmaktadır. Kapanan bir şeridin yeniden kullanıma açılması ne kadar hızlı gerçekleştirilebilirse, etkinin yayılma ve toparlanma süreleri kısalacaktır. Bölüm 5'te ayrıntılı olarak değinileceği üzere geliştirilecek olan algoritma ile beraber çalışacak bir hız kontrol sistemi, kaza-arıza sürecini daha iyi yönetilebilir hale getirecektir.



Şekil 4.9 24.02.2019 tarihli kaza kaydı için yol–zaman diyagramı (yakınlaştırılmış) 1

Tüm bu incelemelere göre, kaza-arıza nedeniyle ilgili yol kesimlerinde ortalama hızların ciddi düşüşlere uğrayacağı açıkça tahmin edilebilir. Bu hız düşüşleri, algoritmanın çalışmasındaki temel etkenlerden biri olarak hızların göz önüne alınmasının, sağlıklı bir karar olduğunu gösterir. İlgili yol kesimlerindeki hız haritalarının incelenip, hızlar ile ilgili daha net analizler yapılabilir.



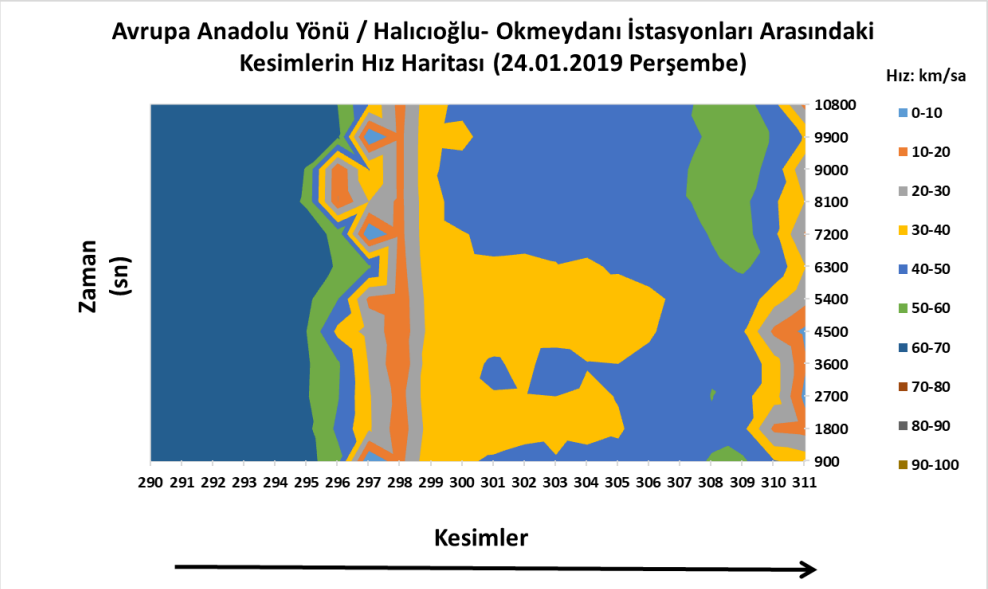
Şekil 4.10 24.02.2019 tarihli kaza kaydı için yol–zaman diyagramı (yakınlaştırılmış) 2

4.2 Hız Haritaları

Hız haritaları, trafik mühendisliğinde oldukça kullanışlı olan bir gösterim şeklidir. 3 boyutlu olarak toplanan trafik verisini 2 boyut üzerinde temsil etme imkânı sunar. Çalışma kapsamında, İETT tarafından temin edilen GPS veri kümelerinde, her bir GPS sinyali için aracın anlık hız değerini kilometre/saat cinsinden veren bir sütun da bulunmaktadır.

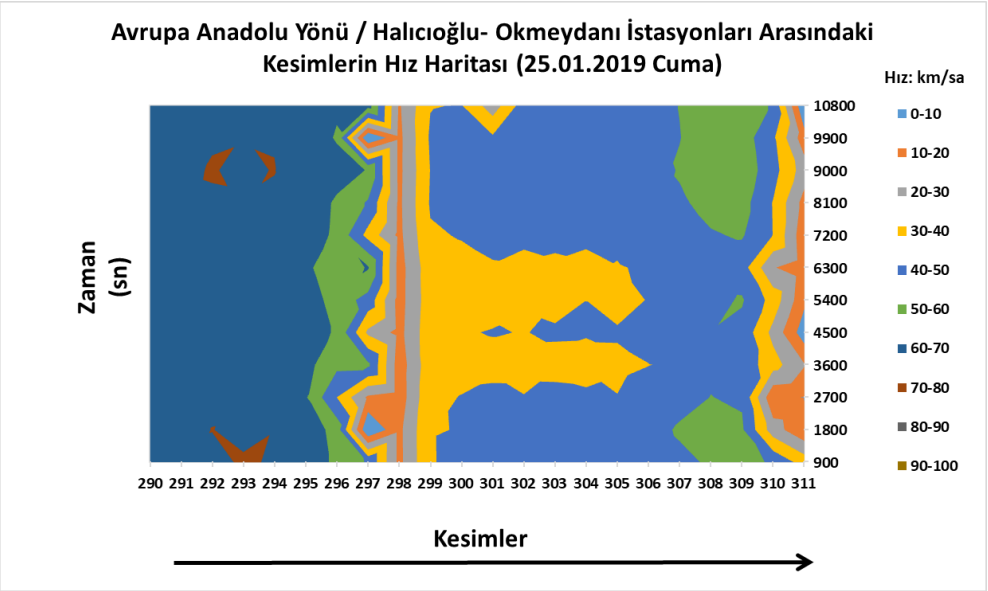
Bölüm 3'te anlatılan ön işleme adımlarında, hız haritalarının çıkarılabilmesi için şerit parçalarına kesim numarası atanması işlemi detaylıca anlatılmıştır. 100-150 metre arasında değişen uzunluklarda Metrobüs koridoru kesimlere ayrılarak, bu kesimler sırasıyla numaralandırılmıştır. Sonrasında ise her bir kesim için belirli bir başlangıç zamanından itibaren (07:00) çeşitli ölçüm aralıklarında ortalama hızlar belirlenerek, hız–zaman–kesim matrislerinde ilgili yerlere işlenmiştir. Bu matrislerin kontür grafiği olarak işlenmesiyle hız haritaları çıkarılmış olur. Metrobüs koridorunun çalışma kapsamında incelenen Beylikdüzü – Zincirlikuyu

arası için yaklaşık 353 kesim belirlenmiştir. Kesim sayısının fazla olması sebebiyle oldukça büyük hız haritaları ortaya çıkmıştır. Hız haritaları EK-B'de ayrıntılı şekilde gösterilecektir. Bölüm 4 kapsamında, sadece analizler sonucu dikkat çekici olan kısımlar paylaşılacaktır.

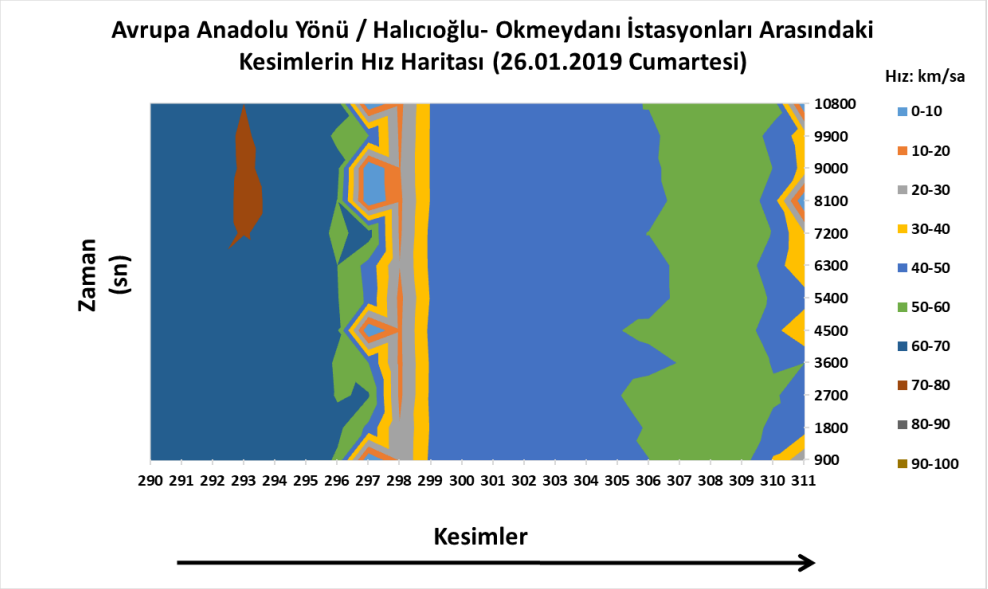


Şekil 4.11 Darboğaz bölgesi hız haritası 1

Hız haritaları temel olarak tekrarlı darboğazlar ile kaza-arıza nedeniyle oluşan tıkanıklıkları tespit etmek üzere kullanılmıştır. Bu analizler sonucunda, hafta içi günlerde 298 ve 306 no'lu kesimlerden oluşan Halıcıoğlu-Okmeydanı istasyonları arasından sabah zirve saatlerde (07:00–10:00 arasında), Avrupa Anadolu yönünde bir tekrarlı darboğaz gözlemlenmiştir. Bu darboğaz hafta sonu ortaya çıkmamaktadır (Şekil 4.13). Fakat hafta içi günlerde sabah zirve saati yoğunluğu sebebiyle, sadece zirve saatleri içerisinde kendisini göstermektedir.



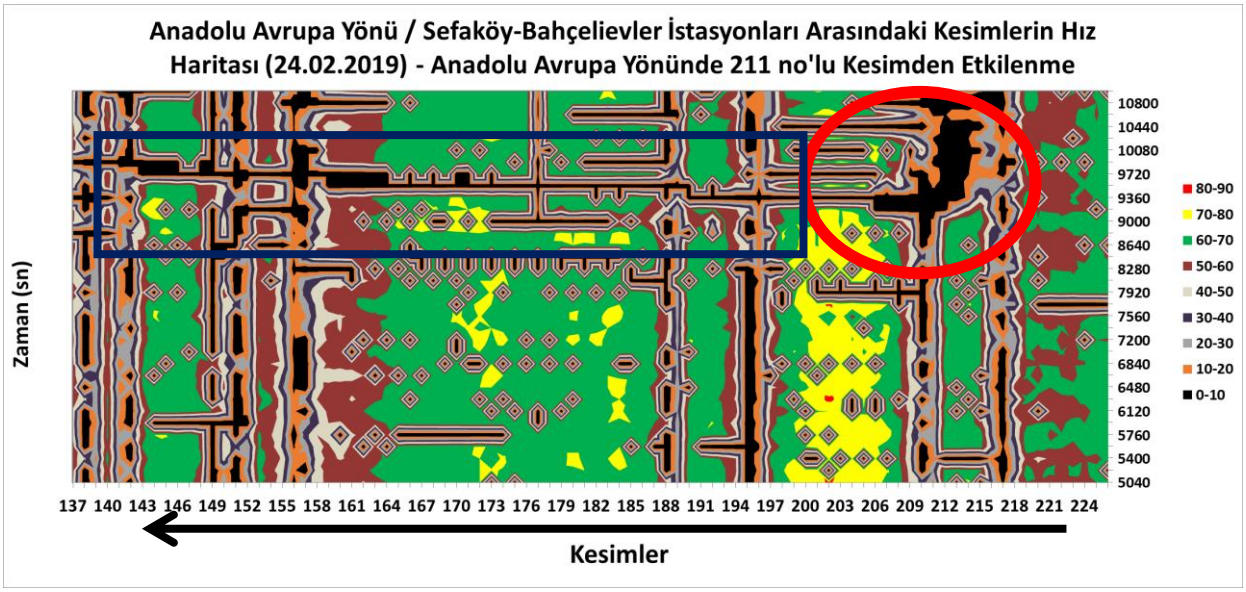
Şekil 4.12 Darboğaz bölgesi hız haritası 2



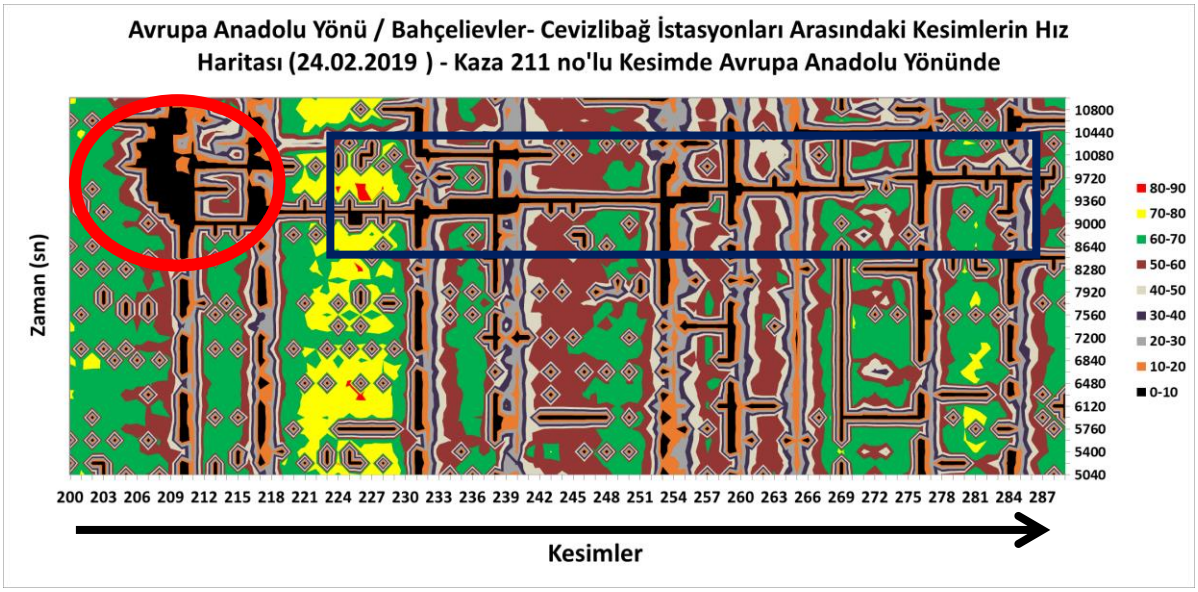
Şekil 4.13 Darboğaz bölgesi hız haritası 3

Şekil 4.11, Şekil 4.12, Şekil 4.13'te görüleceği üzere 24.01.2019 Perşembe günü için zirve saatlerde darboğaz aktiftir ve şok dalgaları biçiminde kendini göstermektedir. Aynı bölge 25.01.2019 Cuma gününde aktif olarak görülmektedir. 26.01.2019 Cumartesi ise darboğaz, seferdeki otobüs sayısı/sıklığı ve istasyonlardaki yolcu sayısı azaldığı için aktif değildir ve Halıcıoğlu-Okmeydanı istasyonları arasındaki kesimlerde ortalama hızlar yüksektir. Eldeki veri setleri içerisinde bu durum hafta içi günlerin tamamında

kendini tekrar etmektedir. EK B'de çeşitli günler için verilmiş olan hız haritaları detaylı incelendiğinde, hafta içi yoğun saatlerde kendini gösteren irili ufaklı başka darboğazlara rastlamak mümkündür. Fakat bu darboğazlar sebebiyle oluşan şok dalgaları ciddi bir yayılma göstermemekte ve ciddi hız düşüşlerine sebep olmamaktadır. Şekil 4.11 ve Şekil 4.12'de gösterilen darboğaz en belirgin olanıdır. Yol–zaman diyagramlarındaki analizlerden vardığımız ana sonuç, geliştirilecek olan kaza-arıza algılama algoritmasının temel parametresinin hız değerleri olması gerektiğidir. Hız haritaları ise bu hız değişimlerin algılanmasını sağlayacak olan, Bölüm 3'te de anlatılan karakteristik hız matrislerinin oluşturulmasına öncülük etmiştir. Sistem üzerinde bir darboğazın varlığı, hızlarda aşırı yoğunluk sebebiyle ani düşüşlerin yaşanabileceğini göstermektedir. Bu nedenle karakteristik hız matrisleri oluşturulurken, hafta içi günlerin hız dağılım frekanslarına bakılmalıdır. İlgili dağılımda, alt uçtaki hızların karakteristik hız olarak seçilmesinin, tekrarlı darboğazlar sebebiyle, kaza-arıza algılama algoritmasının hatalı çalışmasının önüne geçebileceği (ya da algılama hatalarına en küçükleyeceği) düşünülmüştür. Şekil 4.14 ve Şekil 4.15'te bazı hız haritaları verilmiş olup, bir kaza-arıza durumunda sistemin nasıl davranacağı ile ilgili önemli bulgulara rastlanmıştır.

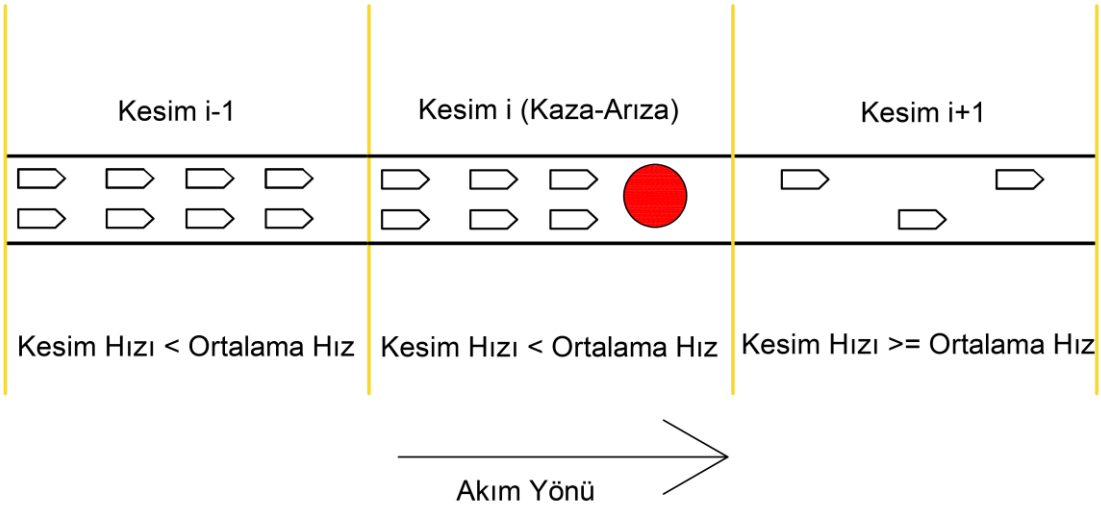


Şekil 4.14 Anadolu Avrupa yönü hız haritası – Avrupa Anadolu yönünde 211 no'lu kesimde meydana gelen kazadan etkilenen kesimler (hızlar: km/sa)

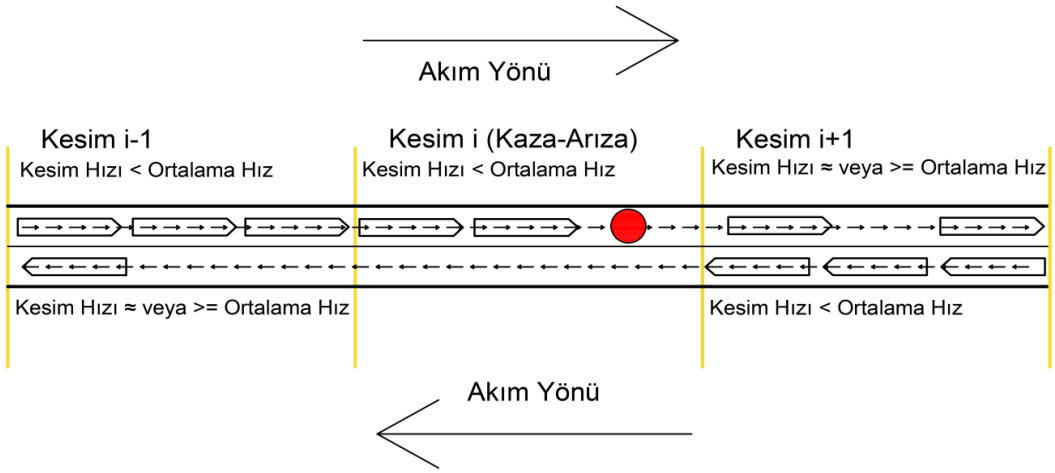


Şekil 4.15 Avrupa Anadolu yönü hız haritası - 211'inci kesim kaza-arıza (hızlar: km/sa)

Şekil 4.9 ve 4.10'daki kaza-arıza olayı için, kesimlerin hız haritaları incelenmiştir. Anımsanacağı gibi, kaza Bahçelievler istasyonunda (211'inci kesim) Avrupa Anadolu yönünde saat 11:30 sıralarında meydana gelmişti. Normal bir karayolu trafiğinde, Şekil 4.16'da görülebileceği gibi herhangi bir kaza-arıza durumu gerçekleştiğinde, kazanın gerçekleştiği kesimden itibaren akım yukarıya doğru kuyruklanma görülmesi beklenir. Bu kuyruklanma, kaza-arıza durumu ortadan kalkıp, ilgili yol kesiminin tekrar kullanıma açılmasına kadar sürer, akım tekrar sağlandıktan sonra ise kuyruğun zaman içerisinde erimesi beklenir.



Şekil 4.16 Normal karayolu trafiğinde, kaza-arıza durumu



Şekil 4.17 Metrobüs hattında, kaza-arıza durumu

Fakat Metrobüs sistemindeki kaza-arıza durumlarında sistemin farklı davrandığı görülmektedir (Şekil 4.17). Öncelikle herhangi bir kaza-arıza durumunda, sistemde belirli bir süre (yaklaşık 5-10 dakika) akış durmaktadır. Daha sonra İETT yetkilileri olay yerine hızlıca müdahale ederek, en azından 1 şeridi kullanıma açmaktadırlar. Bu kullanıma açılan şerit üzerinden çift yönlü araç akışı kontrollü olarak sağlanmaktadır. Şekil 4.9 ve Şekil 4.10 incelendiğinde bu durum açıkça görülebilmektedir, fakat Şekil 4.14 ve 4.15 ile beraber incelendiğinde, Metrobüs koridorundaki trafik akımının başka bir özelliği ortaya çıkmaktadır. Bu özellik ise, tek şerit üzerinden kontrollü akış sebebiyle, akım

aşağı yönde Metrobüs araçlarının konvoy şeklinde devam etmesi ve ortalama hızların hat sonuna kadar düşük bir şekilde gerçekleşmesidir. Bu durum Şekil 4.14 ve Şekil 4.15'te, kazanın olduğu 211'inci kesimden (kırmızı çemberler ve mavi dikdörtgenler) akım aşağı yöne doğru araçların bir konvoy şeklinde devam ettiği ve seferlerinin sonuna kadar düşük hızlarda (0-10 km/sa – koyu siyah kontür rengi) seyir ettikleri gözlenmektedir. Kaza olduktan sonra, İETT/Metrobüs ekiplerinin müdahalesine kadar geçen sürede akım yukarı yönde bir kuyruklanma mevcut olup, süreç yönetimi sayesinde kuyruklanmanın olay yerinin ardında kontrolsüz şekilde uzaması engellenmiştir. Burada Metrobüs ekiplerinin ciddi bir gayret içerisinde olduğu aşikârdır. Fakat sistemin akım aşağı yöndeki aşırı hız düşüşü ise ciddi bir problemdir. Tez kapsamında geliştirilen algoritma, kaza-arıza durumlarının algılanması sürecini hızlandırmayı amaçlamaktadır. Bu algılama sistemine bağlı çalışacak çeşitli kriz yönetim sistemleri geliştirilerek, Metrobüs koridorundaki trafik akışının kaza-arıza olaylarından mümkün olan en az şekilde etkilenmesi sağlanabilir. Bu konuya 5. Bölümde gelecekte yapılabilecek çalışmalar kısmında ayrıntılarıyla değinilecektir.

Tüm bu analizler, ortalama hızların olası bir kaza-arıza durumunda, kesim bazında ciddi düşüşlere uğrayacağını ve bunun belirli bir zaman boyunca devam edeceğini söylemektedir. Ayrıca bu ortalama hız düşüşlerinin, kazanın başlangıcından itibaren akım yukarı kesimlere de yayılacağını, akım aşağı kesimlerde ise kaza anından itibaren ortalama hızların karakteristik hızlara eşit veya yüksek olmaya devam edeceğini söylemektedir. Akım aşağı kesimlerdeki ortalama hızlar, kontrollü geçişlerin başlamasıyla düşmektedir.

Algoritma oluşturulurken, kesim–zaman–hız matrisleri ve takip–uyarı matrisleri, akım yukarı ve akım aşağı kesimlerdeki hız düşüşlerini ve bu düşüşlerin zaman içerisinde takip edilmesine olanak sağlayacaktır.

4.3 Algoritma Performansı

Literatür incelendiğinde, kaza-arıza algılama algoritmalarının performansları üç gösterge ile incelenmektedir. Bu göstergeler algılama oranı (detection rate, DR),

yanlış algılama oranı (false alarm rate, FAR), algılamaya kadar geçen ortalama süre (mean time to detect, MTTD) olarak tanımlanmaktadır. Bu göstergeler Denklem 3'te verilmiştir.

$$DR = \frac{[Doğru\ Algılamarın\ Sayısı]}{[NK]} * 100$$
$$FAR = \frac{[Yanlış\ Algılamaların\ Sayısı]}{[Tüm\ Algılamaların\ Sayısı]} * 100 \quad (3)$$
$$MTTD = \sum_i \frac{(t_i^{algılanan} - t_i^{gerçekleşen})}{NK}$$

Denklem 3'teki $t_i^{algılanan}$ ifadesi, kazanın algoritma tarafından algılandığı zamanı ifade ederken, $t_i^{gerçekleşen}$ ifadesi kazanın gerçekleştiği zamanı ifade etmektedir. NK değeri ise toplam kaza sayısını ifade etmektedir. Literatürde bahsi geçen kaza-arıza algılama algoritmaları, normal karayolu trafiği için geliştirildiğinden, otoyollar özelinde kaza-arıza verisi bulmak kolay olmamaktadır. Bu nedenle kaza-arıza durumunda ilgili yol kesimlerinin nasıl davrandığı ile ilgili veri ancak özel bir çalışma ile belirlenebilmektedir. Bu durum araştırmacıları simülasyon çalışmalarına itmektedir. Simülasyon ortamında, uygun şekilde ve kalibre edilmiş modeller, sanal bir kaza-arıza üreterek, sistemin olası davranışını tahmin etmektedir. Sonrasında ise bu tahmin sonucu oluşan sanal veriler üzerinden algoritmalar şekillenmekte ve sınanmaktadır.

Bir diğer husus ise, karayolu trafiğine yönelik GPS tabanlı kaza-arıza algılama algoritmalarının veri ihtiyacı, genelde trafik hacminin belli bir yüzdesi olarak (penetrasyon oranı) sayıları belirlenen çeşitli takip aralıklarında trafik akımı içerisine yollanan sonda araçları ile sağlanmaktadır. Penetrasyon oranı (Denklem 4) ise çalışmadan çalışmaya değişmektedir. Tez kapsamındaki çalışma için tüm Metrobüs araçları birer sonda aracı vazifesi görmektedir, diğer bir deyişle PR değeri %100'dür. Bu durum, GPS tabanlı kaza-arıza algılama sistemlerinin, BRT gibi izole çalışan ulaştırma türleri için ne kadar uyumlu olduğunun da bir göstergesidir. Bu duruma beşinci bölümdeki tartışma kısmında ayrıntılı şekilde değinilecektir.

$$PR = \frac{Sonda\ Aracı\ Sayısı}{Toplam\ Trafik\ Hacmi} * 100 \quad (4)$$

Çalışma kapsamında ise doğrudan Metrobüs sisteminde gerçekleşmiş olan kaza-arıza verileri kullanılmıştır. Eğer veriler sağlıklı ve yeterli çözünürlükte ise gerçek veri kullanarak yapılan sınamalar algoritmaların performanslarını değerlendirme için en uygun yöntemdir. Fakat normal karayolu trafiğinde bu durum her zaman mümkün olmadığı için simülasyon uygulaması yapılmaktadır. Çalışma kapsamında gerçek kaza-arıza verilerinin kullanılabiliyor olması önemli bir avantajdır. Eldeki kaza-arıza verilerinin sayıları arttırılarak daha ayrıntılı inceleme yapmak her zaman mümkündür fakat çalışma kapsamında İETT tarafından sağlanan sınırlı sayıdaki veriler üzerinden bir deneme yapılabilmiştir.

Bölüm 3.5'te algoritmanın denenmesi için kullanılan yöntem açıklanmıştır. Eldeki veriler sınıflandırılarak algoritma beslenmiştir. Bu veriler ile ilgili ayrıntılı bilgi Tablo 4.1'deki gibidir.

Tablo 4.1 Eldeki verilerin sınıflandırılması

Olay No	Veri Tarihi	Açıklama
1	24.01.2019	Kaza-Arıza Kaydı Bulunmuyor
2	25.01.2019	Kaza-Arıza Kaydı Bulunmuyor
3	26.01.2019	Kaza-Arıza Kaydı Bulunmuyor
4	27.01.2019	Kaza-Arıza Kaydı Bulunmuyor
5	28.01.2019	Kaza-Arıza Kaydı Bulunmuyor
6	29.01.2019	Kaza-Arıza Kaydı Bulunmuyor
7	30.01.2019	Kaza-Arıza Kaydı Bulunmuyor

Tablo 4.1 Eldeki verilerin sınıflandırılması (devamı)

Olay No	Veri Tarihi	Açıklama
8	31.01.2019	Kaza-Arıza Kaydı Bulunmuyor
9	24.02.2019	Saat 11:30'da M4884 Nolu Araç ile M3159 Nolu Aracın Kazası-Yaralanmalı ve Hasarlı Kaza-10 dk Şerit Kapanması-Kaza Yeri Avrupa Anadolu Yönü, Bahçelievler Durağı
10	24.02.2019	Saat 09:46'da M4608 Nolu Araç ile M3045 Nolu Araç-Yaralanmalı ve Hasarlı–3 dk Şerit Kapanması - Avrupa Anadolu Yönü, Bahçelievler Durağı
11	4.02.2019	Kaza-Arıza Kaydı Bulunmuyor
12	5.02.2019	Kaza-Arıza Kaydı Bulunmuyor
13	6.02.2019	Saat 12:22'de M4703 Nolu Araç ile M4504 Nolu Aracın Kazası-Ufak Hasarlı Kaza-Avrupa Anadolu Saadetdere Durağı
14	7.02.2019	Saat 12:10'da M4859 Nolu Araç Şirinevler İstasyonu Üst Geçit Ayağına Çarpma-Az Hasarlı-Avrupa Anadolu Yönü Şirinevler Durağı

Tablo 4.1'de görülebileceği üzere, eldeki veriler ve kaza-arıza kayıtları eşleştirilmiş ve algoritma tüm ilgili verilerle denenmiştir. Kaza-arıza kaydı bulunmayan günler için, algoritma ilgili gün için farklı zaman aralıklarında denenmiştir. Kaza-arıza kaydı bulunan günler ise doğrudan kaza kayıt zamanı üzerinden algoritmaya beslenmiş ve algılamanın gerçekleşip, gerçekleşmediği belirlenmiştir. Algoritmanın akış şeması ve adımları Bölüm 3.5'te açıklanmıştır. Bu bölümde ise doğrudan bir veri kaydı üzerinden sayısal örnek şeklinde açıklanacaktır. Bu sayısal örnek için Tablo 4.1'de olay no dokuz seçilmiştir. Olay no 9'da bahsedilen kaza Bahçelievler durağı içinde gerçekleşmiştir. Bahçelievler durağının kesim numarası 211'dir. Algoritma için Bölüm 3.5'te de bahsedildiği üzere belirli bir başlangıç zamanına, seçilen bir ölçüm aralığı, güncelleme sıklığı, güncellenecek adım sayısı belirlenmelidir. Olay no dokuz için algoritmaya beslenecek verinin başlangıç zamanı 11:29 (kaza-arıza olayından 1 dakika önce),

ölçüm aralığı 1 dakika, güncelleme sıklığı 1 dakika, güncellenecek adım sayısı 3 seçilmiştir. Güncellenecek adım sayısının 3 seçilmesi, algoritmanın yapısı gereği minimum alabileceği değer olarak seçilmiştir. Başlangıç zamanı 11:29 olarak algoritmaya verilmesine rağmen, bu örnek kapsamında hız düşüşlerinin daha iyi takip edilebilmesi için tablolara süre aralığı olarak öncül değerler içinde sonuçlar yazılmıştır. Tüm bu başlangıç parametrelerinden sonra ilk olarak hız-zaman-kesim matrisleri oluşturulmuştur.

Tablo 4.2 Avrupa Anadolu yönü hız-zaman-kesim matrisi (hızlar km/sa birimindedir)

Zaman- Kesim ($V_{karakteristik}$)	208 (48)	209 (33)	210 (14)	211 (14)	212 (53)	213 (58)	214 (60)	215 (55)
11:25-11:26	57	52,5	Na	14,5	59	Na	66	Na
11:26-11:27	65	51	Na	5,7	60	Na	67	Na
11:27-11:28	62	52	Na	0,4	Na	Na	Na	Na
11:28-11:29	59	Na	Na	0,0	Na	Na	Na	Na
11:29-11:30	57	9	Na	0,1	Na	Na	Na	Na
11:30-11:31	7	0	Na	2,0	48	Na	59	55
11:31-11:32	0,18	2	0	2,3	Na	Na	Na	55

Tablo 4.2 ve Tablo 4.3 incelenirse, kaza sonucu hız düşüşleri her iki yönde de akım yukarı yönde hızlıca yayılmıştır. Her iki tabloda “Na” ile gösterilen hücreler ise, ilgili kesimde ilgili zaman aralığında herhangi bir GPS sinyalinin gönderilmediği, yani araç bulunmadığı anlamına gelmektedir. Tablo 4.2 ve Tablo 4.3’te kesim numaralarının altına, ilgili yöndeki karakteristik ortalama hızlar parantez içerisinde yazılmıştır. Hız-zaman-kesim matrislerinin oluşturulmasından sonra, her bir hücredeki değer ilgili karakteristik hız değerleri ile karşılaştırılarak takip-uyarı matrisleri oluşturulmaktadır. Tablo 4.4 ve 4.5

incelenirse, kazanın olduğu Avrupa Anadolu yönü 211'inci kesimde ilgili zaman aralığındaki ortalama hızlar, ilgili kesimin karakteristik hız değerlerinden %50'den fazla bir aşağı yönlü sapma göstermiştir (karakteristik hızdan sapma miktarı ile ilgili açıklama Bölüm 3.3'te verilmiştir). Takip uyarı matrisleri bu tarz sapma durumlarının zaman içerisinde takip edilebilmesini sağlamaktadır. Kaza-arıza durumu Avrupa Anadolu yönünde gerçekleştiği için akım yukarı kesimlerde hız azalmalarındaki yayılma Tablo 4.4'te görülmektedir.

Tablo 4.3 Anadolu Avrupa yönü hız-zaman-kesim matrisi

Zaman- Kesim ($V_{karakteristik}$)	**208 (56)**	**209 (40)**	**210 (25)**	**211 (13)**	**212 (58)**	**213 (62)**	**214 (63)**	**215 (56)**
11:25-11:26	56,0	31,8	Na	11,2	25,5	27,3	29	0
11:26-11:27	55,3	40,8	Na	9,1	31,0	Na	61	49
11:27-11:28	52,5	35,5	Na	7,0	9,1	44	Na	57
11:28-11:29	Na	Na	Na	Na	0,0	Na	Na	51
11:29-11:30	56	35	Na	5,4	Na	67	Na	67
11:30-11:31	Na	Na	Na	0,4	29,5	67	65	63
11:31-11:32	Na	Na	Na	0	1,6	Na	Na	Na

Tablo 4.4 Avrupa Anadolu yönü takip–uyarı matrisi

Zaman- Kesim ($V_{karakteristik}$)	**208 (48)**	**209 (33)**	**210 (14)**	**211 (14)**	**212 (53)**	**213 (58)**	**214 (60)**	**215 (55)**
11:25-11:26	0 (57)	0 (52,5)	Na	0 (14,5)	0 (59)	Na	0 (66)	Na
11:26-11:27	0 (65)	0 (51)	Na	1 (5,7)	0 (60)	Na	0 (67)	Na
11:27-11:28	0 (62)	0 (52)	Na	1 (0,4)	Na	Na	Na	Na

Tablo 4.4 Avrupa Anadolu yönü takip–uyarı matrisi (devamı)

Zaman- Kesim ($V_{karakteristik}$)	**208 (48)**	**209 (33)**	**210 (14)**	**211 (14)**	**212 (53)**	**213 (58)**	**214 (60)**	**215 (55)**
11:28-11:29	0 (59)	Na	Na	1 (0)	Na	Na	Na	Na
11:29-11:30	0 (57)	1 (9)	Na	1 (0,1)	Na	Na	Na	Na
11:30-11:31	1 (7)	1 (0)	Na	1 (2,0)	0 (48)	Na	0 (59)	0 (55)
11:31-11:32	1 (0,18)	1 (2)	1 (0)	1 (2,5)	Na	Na	Na	0 (55)

Tablo 4.5 Anadolu Avrupa yönü takip–uyarı matrisi

Zaman- Kesim	**208 (56)**	**209 (40)**	**210 (25)**	**211 (13)**	**212 (58)**	**213 (62)**	**214 (63)**	**215 (56)**
11:25-11:26	0 (56)	0 (31,8)	Na	0 (11,2)	1 (25,5)	1 (27,3)	1 (29)	1 (0)
11:26-11:27	0 (55,3)	0 (40,8)	Na	0 (9,1)	0 (31,0)	Na	0 (61)	0 (49)
11:27-11:28	0 (52,5)	0 (35,5)	Na	0 (7,0)	1 (9,1)	0 (44)	Na	0 (47)
11:28-11:29	Na	Na	Na	Na	1 (0,0)	Na	Na	0 (51)
11:29-11:30	0 (56)	0 (35)	Na	1 (5,4)	Na	0 (67)	Na	0 (67)
11:30-11:31	Na	Na	Na	1 (0,4)	0 (29,5)	0 (67)	0 (65)	0 (63)
11:31-11:32	Na	Na	Na	1 (0)	1 (1,6)	Na	Na	Na

Takip uyarı matrislerinde, belirli bir kesim için birbirini zaman içerisinde takip eden “1” değerleri var ise, ilgili kesim numarası kayıt edilir ve daha ayrıntılı incelemeye geçilir. Bir önceki adımda kaydedilen kesim numarası kullanılarak,

komşu kesimlerin ortalama hızları incelemeye alınır. Herhangi bir i'inci kesimde Avrupa Anadolu yönünde kaza gerçekleştiyse, akım aşağı yöndeki(i+1'inci) kesimlerde akımın hızı yüksek seyretmeye devam ederken, akım yukarı kesimde ise (i-1'inci) akım hızının düşmesi beklenir. Bu durum Şekil 4.16 ve Şekil 4.17'de gösterilmiştir. Şekil 4.16'daki şartın her iki yön için sağlandığı Tablo 4.6 ve Tablo 4.7'de görülmektedir. Bu durum 211'inci kesimdeki kaza-arıza durumunu doğrulamaktadır. Tablo 4.4'te 11:26 ile 11:29 arasında da birbirini takip eden 1 değerleri oluşmuştur fakat bu değerlerin kaza-arıza durumuna ait olup olmadığı bilinmemektedir ve ayrıca komşu kesimlerde bir hız düşüşü de görülmemektedir. Yazılı kayıtlarda, kaza-arıza durumu ile ilgili bilgiler, kazanın saat 11:30'da olduğunu ifade etmektedir. Bu nedenle kaza-arıza ile ilgili hız düşüşlerinin 11:29-11:30 aralığı itibari ile gerçekleştiğini kabul ederek analizler yapılmıştır.

Son olarak kaza-arıza durumunun doğrulandığı 211'inci kesimde, ilgili zaman aralığında, durmakta ya da çok yavaş hızlarda seyreden araçlar listelenerek, kaza-arıza yaşamış olan şüpheli araçların listesi çıkarılır. Elimizdeki kaza-arıza kaydındaki araçlar, ilgili listede bulunuyorsa başarılı şekilde kaza-arıza algılanmış demektir. Şekil 4.16'daki şartın her iki yön için sağlandığı Tablo 4.6 ve Tablo 4.7'de görülmektedir. Tablo 4.6 ve 4.7'deki ortalama hızlar, problemin başladığı düşünülen zaman aralığı (11:30'dan itibaren) için bulunmuştur. Bu durum 211'inci kesimdeki kaza-arıza durumunu doğrulamaktadır. Örneğin Tablo 4.6 için, 211'inci kesimin akım yukarı kesiminde (210) ortalama kesim hızı 0 km/sa (kuyruklanma), akım aşağı yöndeki 212'inci kesimde ise ortalama kesim hızı 48 km/saat (karakteristik hıza çok yakın) hesaplanmıştır.

Tablo 4.6 Avrupa Anadolu yönü için ilgili komşu kesimlerin incelenmesi

Kesim Numaraları	**210 (i-1)**	**211-Kaza Yeri (i)**	**212 (i+1)**
Hız (km/sa)	0	1,4	48,0

Tablo 4.7 Anadolu avrupa yönü ilgili komşu kesimlerin incelenmesi

Kesim Numaraları	210 (i+1)	211-Kazadan Etkilenme (i)	212 (i-1)
Hız (km/sa)	Na	1,9	7,8

Bu durum, 211'inci kesimdeki kazayı doğrulamaktadır. Tablo 4.7 için ise, Avrupa Anadolu yönü 211'inci kesimdeki kaza diğer yöndeki trafik akımının da durmasına sebep olmuştur. Tablo 4.7'ye göre akım yukarı kesim olan 212'de çok düşük hız değerleri gözlenmiş, akım yukarı kesim olan 210'da ise herhangi bir araç verisi olmadığı için "Na" bilgisi işlenmiştir. Tablo 4.7'de 210'uncu kesim için ortalama hız değeri olarak "Na" çıkması, ilgili kesimde herhangi bir aracın bulunmadığını göstermektedir. Bu durum kesimde serbest akım koşullarının varlığı olarak kabul edilmiştir. Bu durum Şekil 4.17'deki şartın tekrar sağlandığını ve kaza-arıza durumunun doğrulandığını ifade etmektedir.

Algoritmanın son aşaması olarak kaza-arıza durumunun doğrulandığı 211'inci kesimde, ilgili zaman aralığında, durmakta ya da çok yavaş hızlarda seyreden araçlar listelenerek, kaza-arıza yaşamış olan şüpheli araçların listesi çıkarılır. Elimizdeki kaza-arıza kaydındaki araçlar, ilgili listede bulunuyorsa başarılı şekilde kaza-arıza algılanmış demektir.

Tablo 4.8 Avrupa Anadolu yönü, kaza kesimindeki araç listesi

KAPI NUMARASI	GERÇEKLESME ZAMANI	BOYLAM	ENLEM	HIZ	HAT	Kesim
M4884	11:30:04	28,86349	40,99505	0	34AS	211
M3138	11:30:06	28,86328	40,99503	0	34C	211
M3195	11:30:07	28,86287	40,99487	0	34BZ	211
M3159	11:30:07	28,86392	40,9952	0	34BZ	211

Tablo 4.8 Avrupa Anadolu yönü, kaza kesimindeki araç listesi (devamı)

KAPI NUMARASI	GERÇEKLESME ZAMANI	BOYLAM	ENLEM	HIZ	HAT	Kesim
M4865	11:30:10	28,8631	40,99495	0	34AS	211
M3220	11:32:42	28,86353	40,99509	0	34BZ	211
M3192	11:32:44	28,86338	40,99493	0	34BZ	211
M4835	11:32:50	28,86328	40,99495	0	34AS	211
M4808	11:32:51	28,86302	40,99491	0	34AS	211

Tablo 4.8'de verilmiş olan araçların listesi içerisinde, Tablo 4.1 olay no dokuz için verilmiş olan açıklamada kapı numaraları "M4884" ve "M3159" olan araçların bulunduğu görülmektedir (sarı renge boyalı satırlar). Ayrıca ilgili kaza, Metrobüs koridorundaki trafik akışını her iki yönde de etkilemiştir. Algoritma, 211'inci kesimdeki kazayı, diğer yön (Anadolu Avrupa) için de algılamıştır. Anadolu Avrupa yönü, bu kaza kaydı için Avrupa Anadolu yönündeki kazadan etkilenme sebebiyle bir algılama yapmıştır. Bu kaza-arıza algılama algoritması sahada uygulanıyor olduğunu farz edelim. İlgili algoritma her iki yön için sürekli algılamaya çalışacaktır. Olay no 9 özelinde test ettiğimiz durum gerçekleştiğinde ise her iki yönde de aynı kesim için bir algılama yapacaktır. Her iki yön için aynı kesim numarası algılanması, ilgili kazanın şerit kapanmalı ve şiddetli olduğunu söyler. Ayrıca kazanın tekrar doğrulanması anlamına da gelmektedir. Kazanın 11:30'da gerçekleştiğini kabul edersek, 11:29'dan 11:32'ye kadar olan veriler ile kaza-arıza durumu algılanabilmiştir. Literatür incelendiğinde, bu çeşit algılama algoritmaları, kaza-arıza olmayan durumlarda da algılama potansiyeline sahiptir. Bu tarz hatalı algılamalar, FAR (yanlış algılama oranı) değerini arttırmaktadır. Tablo 4.1'de verilmiş olan tüm olaylar için kaza-arıza algılama durumu Tablo 4.9'da gösterilmiştir.

Tablo 4.9 Veri kümelerine göre kaza-arıza algılama durumları

Olay No	Kaza-Arıza Kaydı	Algılama Durumu	Kaza Zamanı	Algılama Zamanı
1	Yok	0	-	-
2	Yok	0	-	-
3	Yok	0	-	-
4	Yok	0	-	-
5	Yok	0	-	-
6	Yok	0	-	-
7	Yok	0	-	-
8	Yok	0	-	-
9	VAR-Şerit Kapanmalı	1	11:29	11:32
10	VAR-Şerit Kapanmalı	1	09:46	09:49
11	VAR-Şerit Kapanması Yok	0	-	-
12	VAR-Şerit Kapanması Yok	0	-	-
13	VAR-Şerit Kapanması Yok	0	-	-
14	VAR-Şerit Kapanması Yok	0	-	-

Tablo 4.9 incelendiğinde, geliştirilen algoritmanın performansına dair genel bir değerlendirme yapmak mümkündür. Algoritma, algılama sistematiği olarak, kesim bazındaki ortalama hızların düşüşü, kuyruklanma veya ortalama hızların sıfır değerine yaklaşmasını tespit etmeye çalışmaktadır. Bu sebepledir ki, şerit kapanması olmayan kaza kayıtları için herhangi bir algılama söz konusu olmamıştır. Zira şerit kapanması olmayan kaza-arıza hadiseleri, kesim bazında ortalama hızları fazla etkilememiştir veya karakteristik hızların yeterince altına

inmemiştir. Bu durum algoritmanın FAR değerinin düşük olduğuna işarettir. Fakat algoritma performansı sadece FAR değeri ile ölçülmemektedir. Burada DR (algılama oranı) hesaplanmadan önce şu sorunun cevabı verilmelidir, Algoritma kapsamında algılanması hedeflenen kazaların türü nedir? Eğer bu soruya verilen cevap, şerit kapanmalı kazalar ise DR'nin yüksek çıkacağı görülmektedir. Ancak algılanması beklenen kaza tipleri hem şerit kapanmalı hem de şerit kapanmasına sebep olmayan küçük çaplı kazalar ise Tablo 4.9'daki durum bizlere DR değerinin düşük çıkabileceğini göstermektedir. Algoritma performansının bir diğer göstergesi olan MTTD (algılamaya kadar geçen sürelerin ortalaması) ise, Tablo 4.8'e göre 2-3 dakika arasında olduğunu göstermektedir. Tablo 4.9, bizlere genel bir değerlendirme imkânı sunabilmektedir. Daha kesin sonuçlara ulaşmak için farklı kaza-arıza durumları ile algoritma sınanmaya devam edilmelidir. Eğer farklı kaza-arıza durumlarına ait veri kümelerine erişim yok ise sistemin simülasyon ortamında modellenmesi ile algoritmanın sınanması gerçekleştirilebilir. Eğer algoritma algılamasını beklediğimiz kazaların şerit kapanmalı kazalar olacağını kabul edersek, Tablo 4.8'e göre DR, FAR, MTTD değerleri sırasıyla %100, %0, 2,5 dakika olarak hesaplanmıştır. Bu değerler algoritmanın tasarımına uygun davrandığını ve daha ayrıntılı bir çalışma ile kullanılabilir bir algoritma olacağına işarettir. Fakat tezin yazarı bu değerlerin mevcut kısıtlı kaza-arıza verileriyle elde edilmiş ön bulgular olduğunun unutulmaması kanaatindedir.

5
SONUÇ VE ÖNERİLER

Bu bölümde kaza-arıza algoritmaları ile ilgili literatür hakkında değerlendirmelere ve gelecekte yapılabilecek çalışmalara yer verilecektir. Son olarak ise sonuç kısmı ile tez tamamlanacaktır.

5.1 Tartışma

Bölüm 1'de detaylı şekilde bir literatür özeti vermiştik. Bu literatür sınıflandırmasına göre, tez kapsamında geliştirilmiş olan algoritma, sonda aracı tabanlı algoritma sınıfına girmektedir. Sonda aracı tabanlı algoritmalar genel olarak normal otoyollara yönelik kaza-arıza algılama sistemleri olarak geliştirilmektedir. BRT sistemleri için bir sisteme literatür içerisinde rastlanmamıştır. Hali hazırda yayınlanmış bir akademik çalışma bulunamadığı gibi benzer sistemlerin varlığı ile ilgili bir bilgi de bulunamamıştır. Bu bağlamda tez çalışması kapsamında geliştirilen algoritma akademik manada bir ilki gerçekleştiriyor gibi gözükmektedir. Akademik manadaki her ilk gibi birçok araştırma fırsatı da bu çalışma ile beraber doğmaktadır. Bu konuya gelecek çalışmalar kısmında değinilecektir.

Literatür incelendiğinde, benzer algoritmaların performanslarında bazı kısıtlar bulunmaktadır. Özellikle sonda aracı tabanlı algoritmalarda, penetrasyon oranı algoritmaların performanslarını belirleyen temel etkendir. Penetrasyon oranları %1, %5, %10 gibi değerler alsa bile, bu değerlerin artması bu sistemlerin sahaya uygulanabilirliğini zorlaştırmakta ve işletme maliyetleri doğurmaktadır. Filosunun tamamında GPS altyapısı bulunan Metrobüs sistemleri ise sonda aracı tabanlı algoritmalar için en uygun çalışma ortamıdır. Sistemin karışık trafikten izole olması, normal karayolu trafiğinin düzensizliklerini bünyesinde barındırmaması, sonda aracı tabanlı algoritmaların BRT sistemlerine olan uygulanabilirliğini arttırmaktadır.

Diğer önemli bir husus ise normal karayolu akımlarındaki tekrarlı darboğazlardır. Literatürdeki temel problemlerden biri de ilgili algoritmaların tekrarlı darboğazlarla, kaza-arıza olaylarını birbirinden ayıramamasıdır. Bu problemin ortaya çıkmasında, penetrasyon oranlarının etkisi olduğu gibi, karayolu trafiğinin düzensizliğinin de önemli bir payı vardır. Bu manada, otoyollar üzerinde aktif biçimde kullanılan kaza-arıza algılama sistemlerinin genelde detektör tabanlı olma sebebi daha anlaşılır olmaktadır. Otoyollar için geliştirilen sonda aracı tabanlı algoritmalar için penetrasyon oranları belirli bir değere kadar arttırılabilmektedir. Bu kısıt, araştırmacıları sonda aracı verilerini kullanarak çok çeşitli yöntemleri denemeye itmiştir. Bölüm 1'de verilen literatür özetinde sonda aracı tabanlı algoritmaların çeşitliliği üzerinden bu durum kolaylıkla görülmektedir. Birbirinden farklı çalışan onlarca algoritma bulunduğu gibi tam anlamıyla alt bir sınıflandırma yapılamamaktadır.

Şu herkes tarafından fark edilebilir ki her algoritmanın tasarımı, aslında algoritmanın çalışma ortamını da belirlemektedir. Geliştirilen algoritmalar farklı çalışma ortamlarında aynı performansı vermeyecektir. Zira algoritmalar tasarım aşamasında, çalışma ortamının geometrisi, eğer kullanılıyorsa detektör veya sensörlerin yapısı, verilerin tipi, verilerin işlenme yöntemi, ilgili yol kesimi veya sistemin analiz ederken kullanılan diğer yöntemler vb. gibi etkenlere göre şekil almaktadır.

Son olarak ise algoritma performanslarının karşılaştırılması için kullanılan DR, FAR, MTTD göstergelerin her biri farklı karakteristiklere sahiptir. Özellikle MTTD değeri algoritma yapısından çok etkilenirken, FAR değerleri veri doğruluğu ve çözünürlüğünden etkilenmektedir. DR değerlerinin en çok etkilendiği etken penetrasyon oranı olarak gerçekleşmektedir. Tüm bunlar bizlere algoritmaların geliştirilmesi aşamasında çalışma ortamının tüm dinamiklerinin göz önünde bulundurulması gerektiğini açıkça göstermektedir.

5.2 Sonuç ve Öneriler

Tez çalışması kapsamında, sonda aracı tabanlı bir kaza arıza algılama algoritması geliştirilmiştir. İlgili algoritma İstanbul'daki Metrobüs Sistemi için

geliştirilmiş olup, Metrobüs filosunun tamamında var olan GPS altyapısını kullanmaktadır.

Bu algoritmanın geliştirilmesi için öncelikle Metrobüs sisteminin detaylı bir analizi yapılmıştır. Bu analiz, Metrobüs hattını işletmekte olan İETT tarafından sağlanan GPS veri kümeleri ile yapılmıştır. İlgili veri kümeleri 24-31 Ocak 2019, 24 Şubat 2019 (kaza-arıza veri kümesi) ve algoritmanın duyarlılığını ölçmek için 4-7 Şubat 2019 (küçük çaplı kaza-arıza veri kümesi) tarihlerini kapsamaktadır. İlgili analizin yapılması için Metrobüs sisteminin yol-zaman, hız-zaman diyagramları çıkartılmıştır. Bu diyagramların ve grafiklerin çıkarılması, çalışma için geliştirilmiş yöntemlerle gerçekleştirilmiştir. İlgili yöntemler ise tez çalışması kapsamında detaylı şekilde anlatılmıştır. İlgili diyagramlar sayesinde sistemin normal şartlar altında ve kaza-arıza durumundaki davranışı belirlenmiş ve tekrarlı darboğazların varlığı gözlenmiştir. Kaza-arıza günlerine ait diyagramlar, olası bir kaza durumunda Metrobüs sisteminde trafik akımının nasıl davrandığı belirlemede kullanılmış ve algılama sistematiği için strateji belirlenerek algoritma oluşturulmuştur.

Algoritma geliştirildikten sonra gerçek kaza-arıza verileri ile algoritmanın sınanması gerçekleştirilmiştir. Yeterli kaza-arıza verisi olmadığı için algoritma sınaması kısıtlı veri ile yapılmıştır. Algoritmanın kullanılabilir olması için daha farklı kaza-arıza verilerine erişilmesi gerekmektedir. Mevcut veriler ışığında algoritmanın performansı yüksek çıkmıştır. Literatürde ortak olarak kullanılan DR, FAR, MTTD gibi göstergeler, mevcut kaza-arıza verileri ışığında hesaplandığında ilgili değerler sırasıyla %100, %0, 3 dakika olarak gerçekleşmiştir. Fakat tezin yazarı bu değerlerin ön bulgu olduğunun unutulmaması gerektiğini vurgulamaktadır. Daha net sonuç elde etmek için daha fazla veriye erişim ve daha fazla sınama yapılması gerekmektedir.

Algoritma yapısı bakımından literatürdeki bazı algoritmalar ile benzerlik gösterse (Kamran ve Haas, 2007) de çalışma ortamı bakımından özgündür. Diğer bir özgün yanı ise algoritmanın tasarımı öncesi Metrobüs sisteminin analizi için kullanılan yöntemlerdir. Literatürde kısıtlı bilgi olan GPS verilerinden yol-

zaman diyagramlarının çıkarılmasına yönelik genel bir yöntem geliştirilmiş ve tez kapsamında detaylıca açıklanmıştır.

5.3 Gelecek Çalışmalar

BRT sistemlerinin karakteristik özellikleri, sonda aracı tabanlı kaza-arıza algoritmalarını uygulanabilir kılmaktadır. Çalışma kapsamında geliştirilen algoritma akademik manada bir ilki oluşturmaktadır. Bu ilk olma özelliği birçok araştırma fırsatını da beraberinde getirmektedir. Özelikle kaza-arıza algılama algoritmasına entegre olarak birçok kaza-arıza durumu yönetim (incident management) stratejisi ve sistemi çalışabilir. Örneğin belirli bir kesimde kaza algılandıktan sonra, ilgili kesime doğru seyir halinde olan araçlara hız kısıtlaması verecek veya algılama sonrası terminal bölgelerinde geçici olarak sefer sıklığına müdahalede edecek bir sistem geliştirilebilir. Aynı zamanda mevcut kaza-arıza algılama algoritması geliştirilmeye devam edilebilir ve algılama sistematiği açısından çeşitlendirilebilir. Bu çalışma kapsamındaki algoritmanın parametreleri, genel olarak deneme yanılma ve iterasyonlara dayanmaktadır. Nihayetinde algoritmanın uygulanabilir hale gelmesinden önce, ilgili parametrelerin her biri optimize edilmelidir. Bu optimizasyon ancak daha kapsamlı veri kümelerine erişim ile gerçekleşebilir. Eğer veri kümelerine erişim kısıtlı ise sistemin iyi kalibre edilmiş bir simülasyon modeli oluşturularak, elde edilelecek veri ile optimizasyonun yapılması gerekmektedir.

Yol-zaman diyagramlarının çıkarılması için tez kapsamında önerilen yöntemlerin iyileştirilmesi de gelecekte yapılabilecek çalışmalar arasındadır. Yol-zaman diyagramları özellikle BRT sistemleri için gerçek zamanlı izleme (real time monitoring) sistemlerine dönüştürülerek anlık takip imkânı sunabilecektir. Yol-zaman diyagramları şok dalgası saptama çalışmalarında da kullanılabilir. Otobüs taşımacılığı yapan tüm ulaştırma sistemleri için ortak bir problem olan demetlenme (bunching) hadisesi, BRT sistemlerinde de gerçekleşebilmektedir. Bu konu GPS verileri ile daha detaylı şekilde analiz edilebilir ve sistem için demetlenme önleyici sefer düzenlemesi veya kontrol mekanizmaları geliştirilebilir.

Kaynakça

Anwar, A., Zeng, W., & Arisona, S. M. (2014). Time–Space Diagram Revisited. *Transportation Research Record: Journal of the Transportation Research Board*, *2442*(1), 1–7. https://doi.org/10.3141/2442-01

Asakura, Y., Kusakabe, T., Xuan, L., & Ushiki, T. (2017). Incident detection methods using probe vehicles with on-board GPS equipment. *Transportation Research Part C*, *81*, 330–341. https://doi.org/10.1016/j.trc.2016.11.023

Balke, K., Dudek, C. L., & Mountain, C. E. (1996). Using Probe-Measured Travel Times to Detect Major Freeway Incidents in Houston, Texas. *Transportation Research Record: Journal of the Transportation Research Board*, *1554*(1), 213–220. https://doi.org/10.1177/0361198196155400125

Cassidy, M. J. (2006). Traffic Flow and Capacity. *Handbook of Transportation Science*, 155–191. https://doi.org/10.1007/0-306-48058-1_6

Cheu, R. L., Qi, H., & Lee, D. H. (2002). Mobile sensor and sample-based algorithm for freeway incident detection. *Transportation Research Record*, (1811), 12–20.

Chopde, N., & Nichat, M. (2013). Landmark Based Shortest Path Detection by Using A* and Haversine Formula. *International Journal of Innovative Research in Computer and Communication Engineering* , 1(2), 298–302.

Gall, Ana I. Hall, Fred L., (1989).Distinguishing Between Incident Congestion and Recurrent Congestion: A Proposed Logic, *Transportation Research Record,* 1232(1), 1-8

G. van Rossum (1995), Python tutorial. Technical Report CS-R9526, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, May 1995.

Ishak, S. S., & Al-Deek, H. M. (1998). Fuzzy ART Neural Network Model for Automated Detection of Freeway Incidents. *Transportation Research Record: Journal of the Transportation Research Board*, *1634*(1), 56–63. https://doi.org/10.3141/1634-07

Jeong, Y. S., Castro-Neto, M., Jeong, M. K., & Han, L. D. (2011). A wavelet-based freeway incident detection algorithm with adapting threshold parameters. *Transportation Research Part C: Emerging Technologies*, *19*(1), 1–19. https://doi.org/10.1016/j.trc.2009.10.005

Kamran, S., & Haas, O. (2007). A multilevel traffic incidents detection approach: Identifying traffic patterns and vehicle behaviours using real-time GPS data. *IEEE Intelligent Vehicles Symposium, Proceedings*, 912–917.

Karim, A., & Adeli, H. (2002). Incident Detection Algorithm using Wavelet Energy Representation of Traffic Patterns,*Journal of Transportation Engineering,*128 (3), https://doi.org/10.1061/(ASCE)0733-947X(2002)128

Kinoshita, A., Takasu, A., & Adachi, J. (2015). Real-time traffic incident detection using a probabilistic topic model. *Information Systems*, *54*, 169–188. https://doi.org/10.1016/j.is.2015.07.002

Lee, J.-T., & Taylor, W. C. (1999). Application of a Dynamic Model for Arterial Street Incident Detection. *ITS Journal - Intelligent Transportation Systems Journal*, *5*(1), 53–70. https://doi.org/10.1080/10248079908903757

Lee, S., Krammes, R. A., & Yen, J. (1998). Fuzzy-logic-based incident detection for signalized diamond interchanges. *Transportation Research Part C: Emerging Technologies*, *6*(5–6), 359–377. https://doi.org/10.1016/S0968-090X(99)00004-2

Li, F., Yang, D., & Ma, K. (2013). Bus Rapid Transit (BRT) Bunching Analysis with Massive GPS Data. *ICTE 2013*, 959–965 https://doi.org/10.1061/9780784413159.141

Li, Y., & McDonald, M. (2005). Motorway incident detection using probe vehicles. *Proceedings of the Institution of Civil Engineers: Transport*, *158*(1), 11–15. https://doi.org/10.1680/tran.2005.158.1.11

Lin, W. H., & Daganzo, C. F. (1997). A simple detection scheme for delay-inducing freeway incidents. *Transportation Research Part A: Policy and Practice*, *31*(2), 141–155. https://doi.org/10.1016/S0965-8564(96)00009-2

Michalopoulos, P. G. (1991). Vehicle detection video through image processing: the Autoscope system. *IEEE Transactions on Vehicular Technology*, *40*(1), 21–29. https://doi.org/10.1109/25.69968

M. S. Ahmed and A. R. Cook (1979), Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques, *Transportation Research Record*, 722(1), pp. 1-9

M. S. Ahmed and A. R. Cook (1982), Application of Time-Series Analysis Techniques to Freeway Incident Detection, *Transportation Research Record*, 841 (3), 19–21.

Mouskos, K. C., Niver, E., Lee, S., Batz, T., & Dwyer, P. (1999). Transportation Operations Coordinating Committee System for Managing Incidents and Traffic: Evaluation of the Incident Detection System. *Transportation Research Record: Journal of the Transportation Research Board*, *1679*(1), 50–57. https://doi.org/10.3141/1679-07

Nuckles, N. B., Messer, C. J., & Dudek, C. L. (1974). Incident detection on urban freeways. *Transportation Research Record*, *495*, 12–24.

Thancanamootoo, S. &Bell, M.G.H. (1988). Automatic detection of traffic incidents on a signal-controlled road network. Research Report No. 76, Transport Operations Research Group, University of Newcastle, Tyne, UK

Url - Istanbul Elektrik Tramvay ve Tünel İşletmeleri (İETT), 16 Kasım 2019 tarihinde https://metrobus.iett.istanbul adresinden alınmıştır.

Payne, H.J. (1976). Development and testing of incident detection algorithms, Volume 1: summary of results., Report No. FHWA-RD-76-19, FHWA, Washington D.C., April 1976

Parkany, E., & Bernstein, D. (1995). Design of incident detection algorithms using vehicle-to-roadside communication sensors. *Transportation Research Record*, (1494), 67–74.

Parkany E. & Xie, P.C.(2005), A Complete Review of Incident Detection Algorithms & their deployment: What Works and What Doesn't, Tech. Rep., Fall River, MA:

New England Transp. Consortium.

Petty, K. F., Skabardonis, A., & Varaiya, P. P. (1997). Incident Detection with Probe Vehicles: Performance, Infrastructure Requirements, and Feasibility. *IFAC Proceedings Volumes*, *30*(8), 125–130. https://doi.org/10.1016/s1474-6670(17)43812-2

QGIS Development Team (2019). QGIS Geographic Information System. Open Source Geospatial Foundation Project. http://qgis.osgeo.org.

Samant, A., & Adeli, H. (2000). Feature Extraction for Traffic Incident Detection Using Wavelet Transform and Linear Discriminant Analysis. *Computer-Aided Civil and Infrastructure Engineering*, *15*(4), 241–250. https://doi.org/10.1111/0885-9507.00188

Sermons, M. W., & Koppelman, F. S. (1996). Use of vehicle positioning data for arterial incident detection. *Transportation Research Part C: Emerging Technologies*, *4*(2), 87–96. https://doi.org/10.1016/0968-090X(96)00003-4

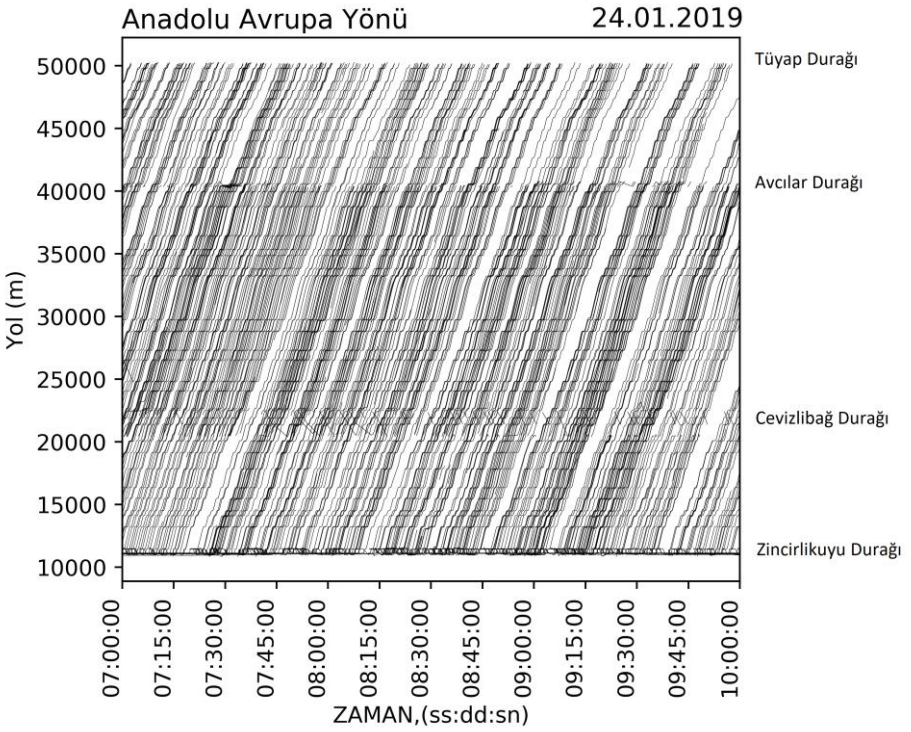
Sethi, V., Bhandari, N., S. Koppelman, F., & L. Schofer, J. (1995). Arterial incident detection using fixed detector and probe vehicle data. *Transportation Research Part C*, *3*(2), 99–112. https://doi.org/10.1016/0968-090X(94)00017-Y

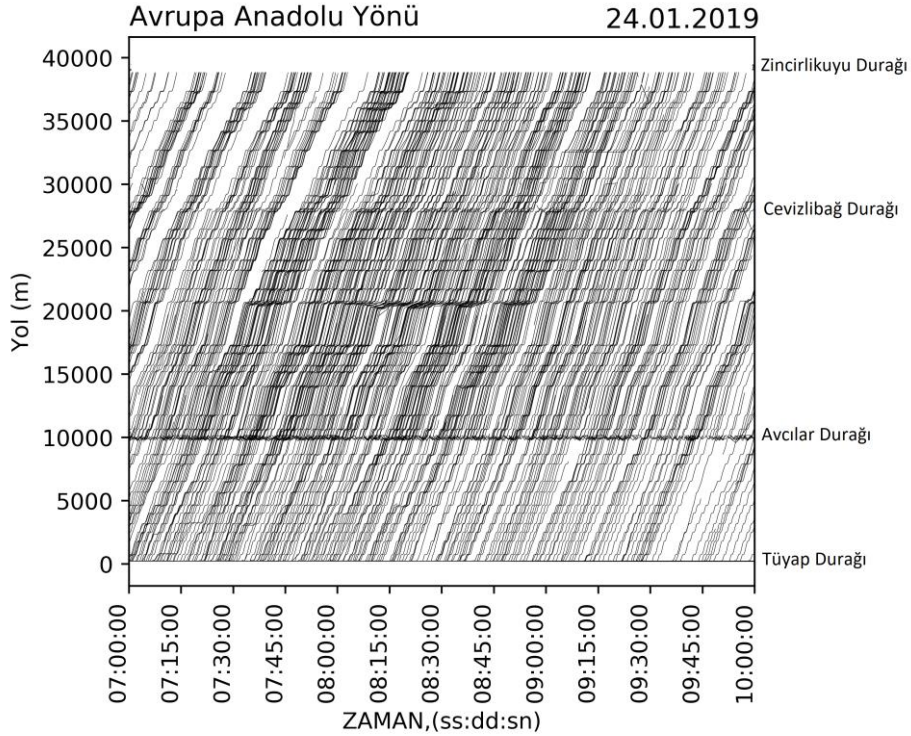
Vlahogianni, E. I., Karlaftis, M. G., & Golias, J. C. (2014). Short-term traffic forecasting: Where we are and where we're going. *Transportation Research Part C: Emerging Technologies*, *43*, 3–19. https://doi.org/10.1016/j.trc.2014.01.005

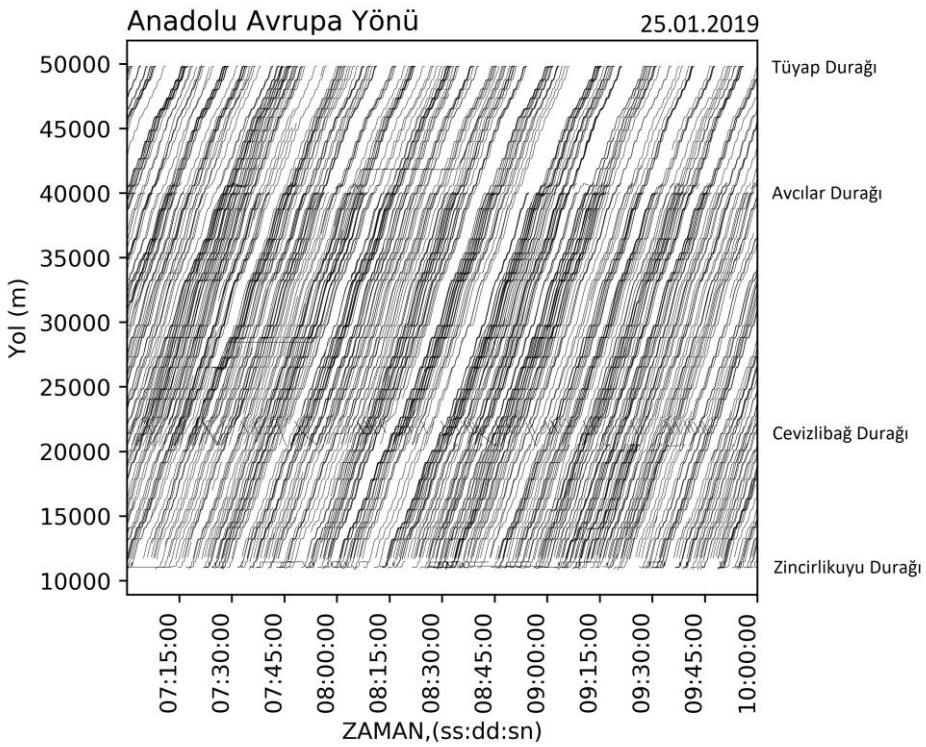
Willsky AS, Chow E, Gershwin S, Greene C, Houpt P, Kurkjian A (1980),Dynamic model-based techniques for the detection of incidents on freeways, *Automatic Control IEEE Transactions on*, 25(3), 347–360, https://doi.org/10.1109/TAC.1980.1102392

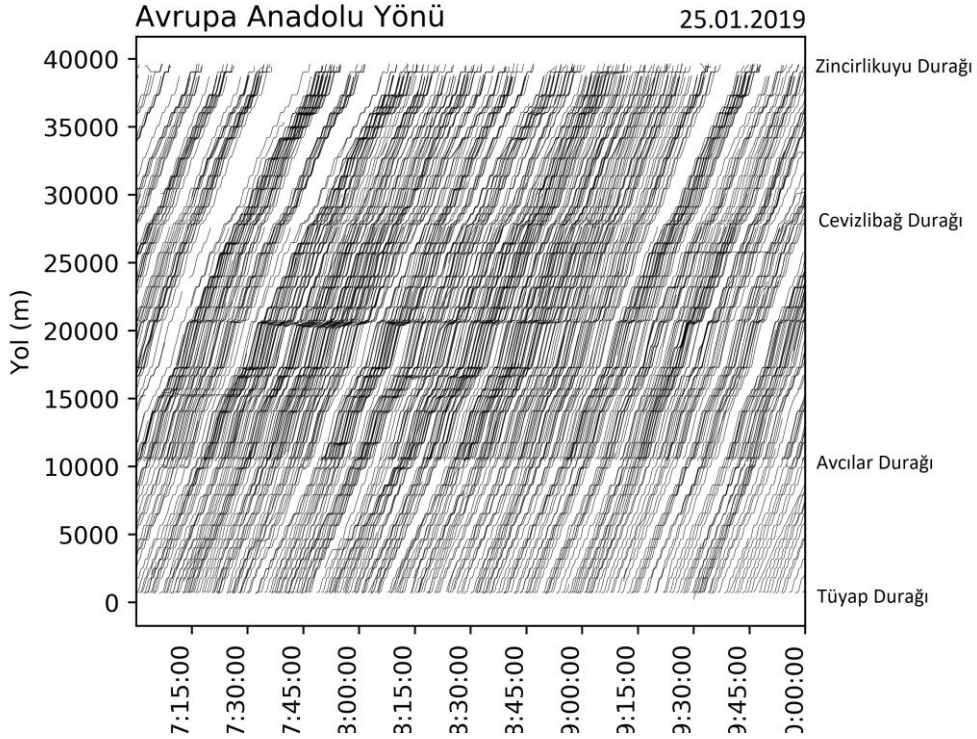
Zhu, T., Wang, J., & Lv, W. (2009). Outlier mining based automatic incident detection on urban arterial road. *Proceedings of the 6th International Conference on Mobile Technology, Application and Systems, Mobility '09*. https://doi.org/10.1145/1710035.1710064

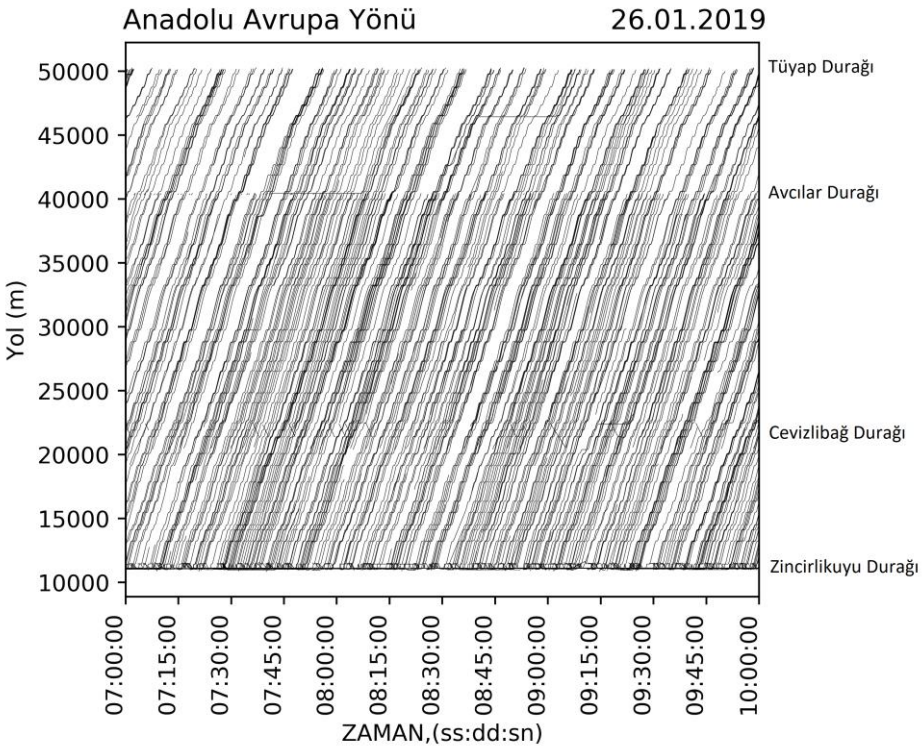
A
Yol – Zaman Diyagramları

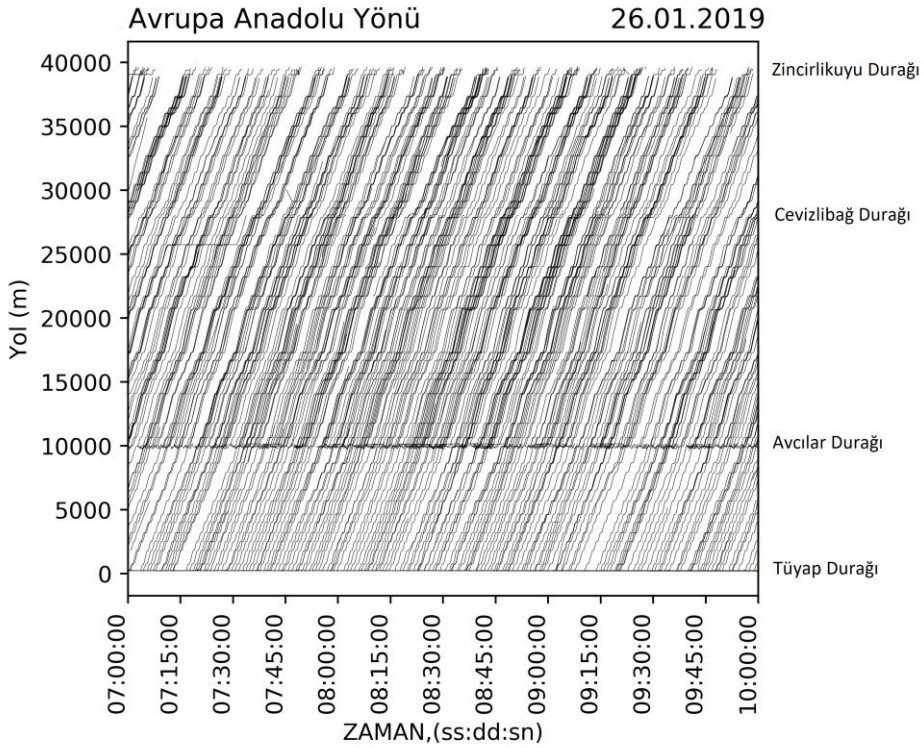


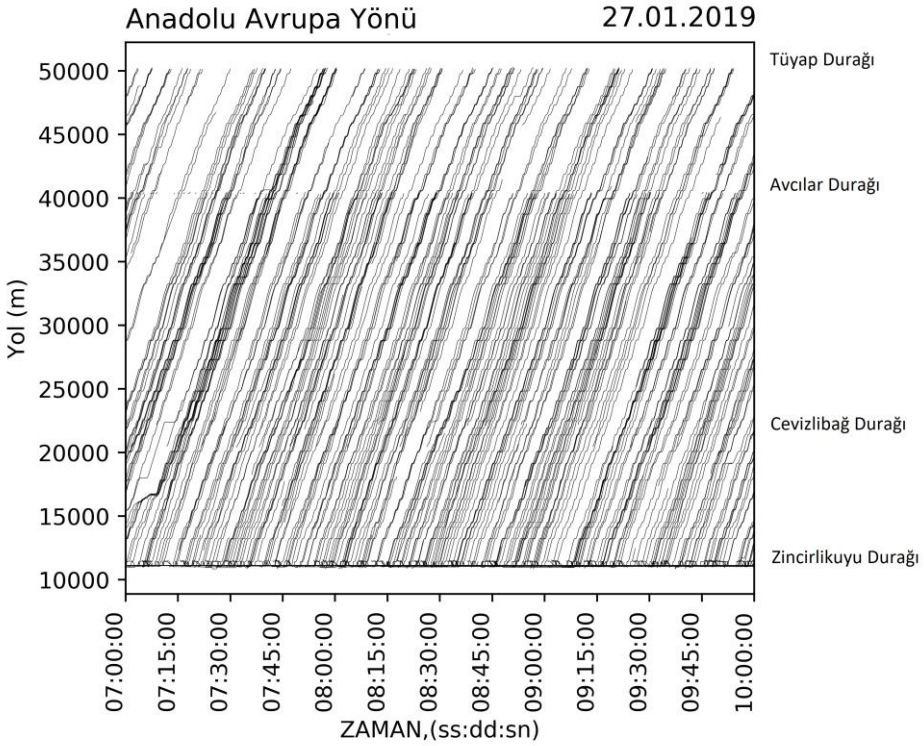


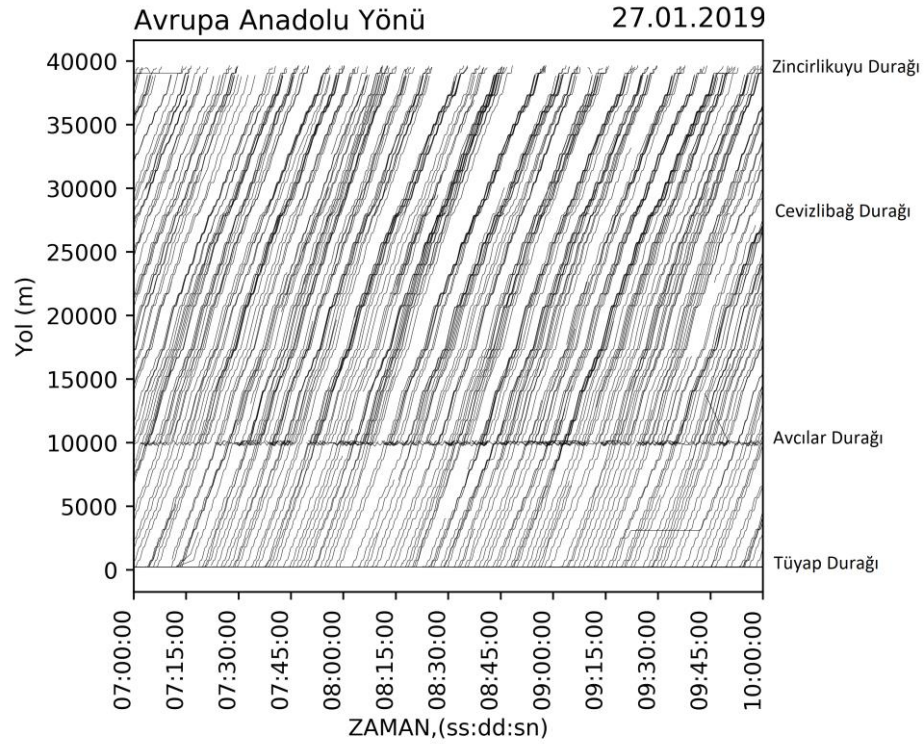

Anadolu Avrupa Yönü
25.01.2019
Yol (m)
50000
45000
40000
35000
30000
25000
20000
15000
10000
Tüyap Durağı
Avcılar Durağı
Cevizlibağ Durağı
Zincirlikuyu Durağı
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)

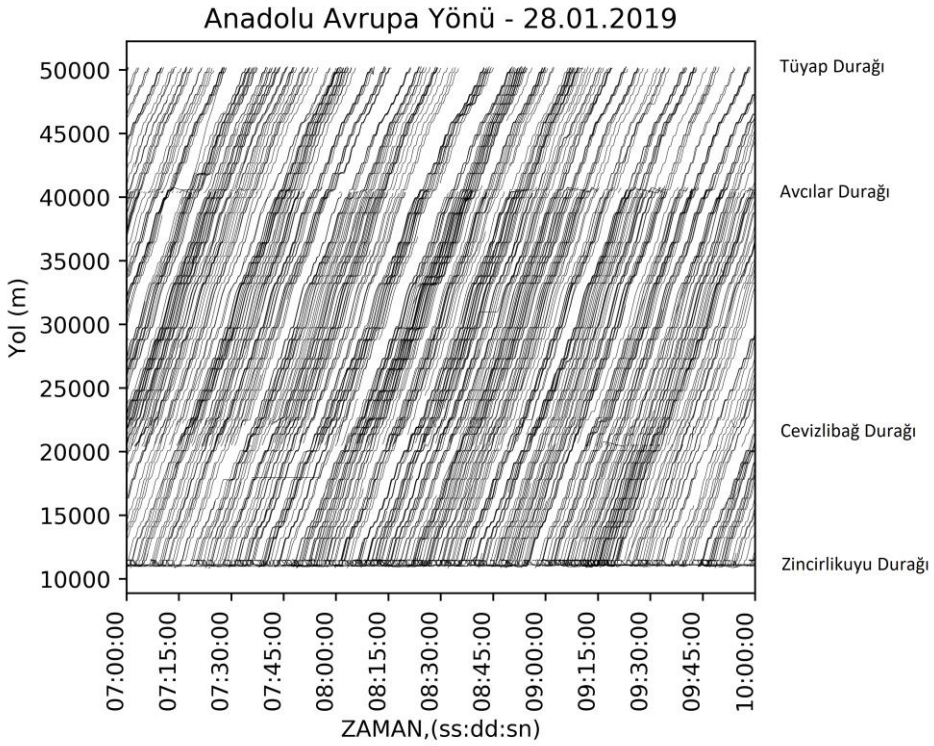

Avrupa Anadolu Yönü
25.01.2019
Yol (m)
40000
35000
30000
25000
20000
15000
10000
5000
0
Zincirlikuyu Durağı
Cevizlibağ Durağı
Avcılar Durağı
Tüyap Durağı

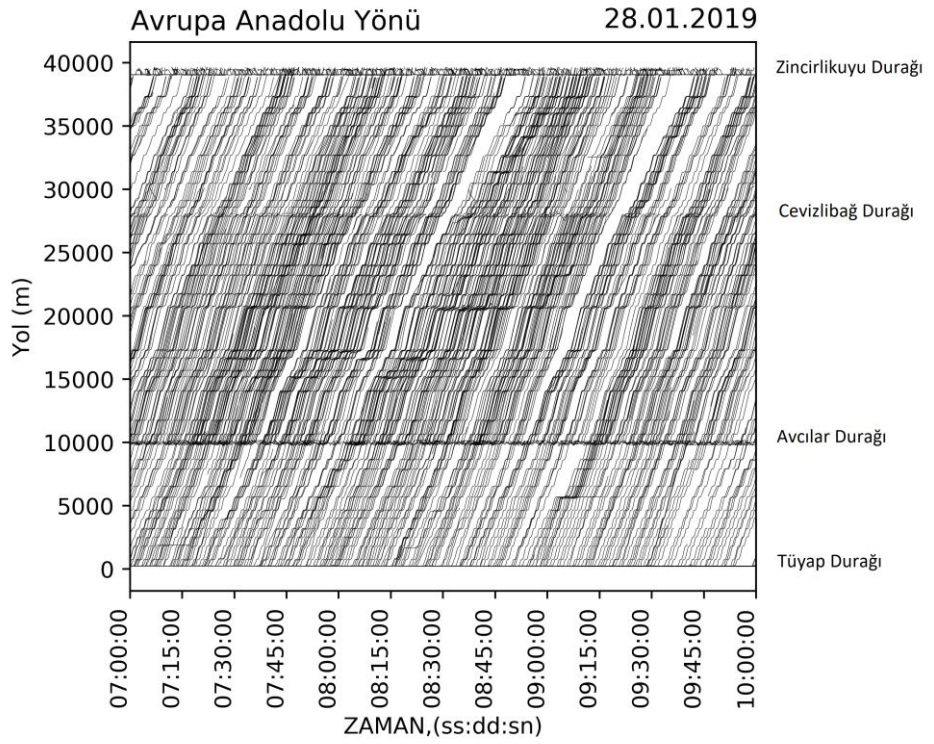

Anadolu Avrupa Yönü
26.01.2019
Yol (m)
50000
45000
40000
35000
30000
25000
20000
15000
10000
Tüyap Durağı
Avcılar Durağı
Cevizlibağ Durağı
Zincirlikuyu Durağı
07:00:00
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)


Avrupa Anadolu Yönü
26.01.2019
Yol (m)
40000
35000
30000
25000
20000
15000
10000
5000
0
Zincirlikuyu Durağı
Cevizlibağ Durağı
Avcılar Durağı
Tüyap Durağı
07:00:00
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)

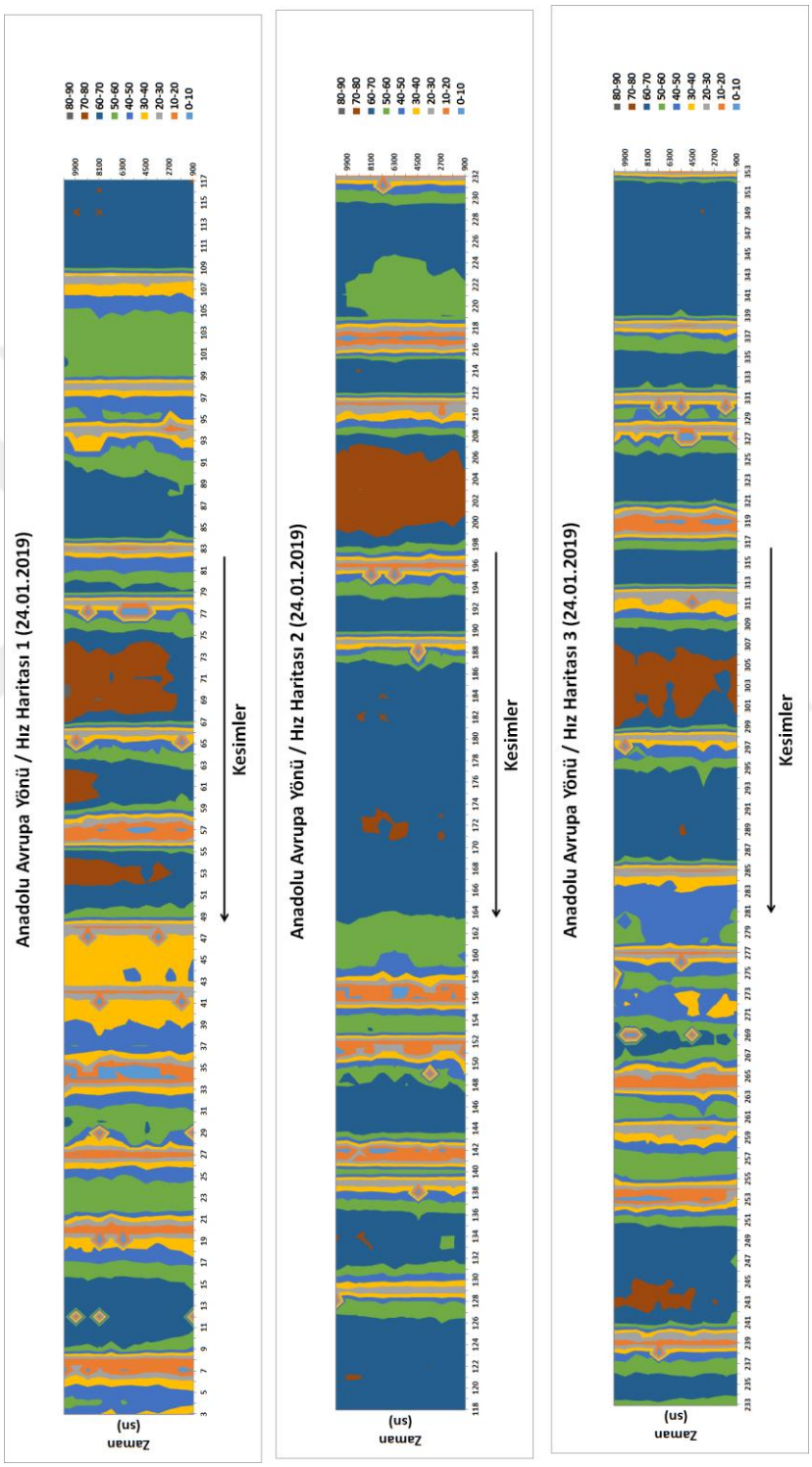

Anadolu Avrupa Yönü
27.01.2019
Tüyap Durağı
Avcılar Durağı
Cevizlibağ Durağı
Zincirlikuyu Durağı
Yol (m)
50000
45000
40000
35000
30000
25000
20000
15000
10000
07:00:00
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)

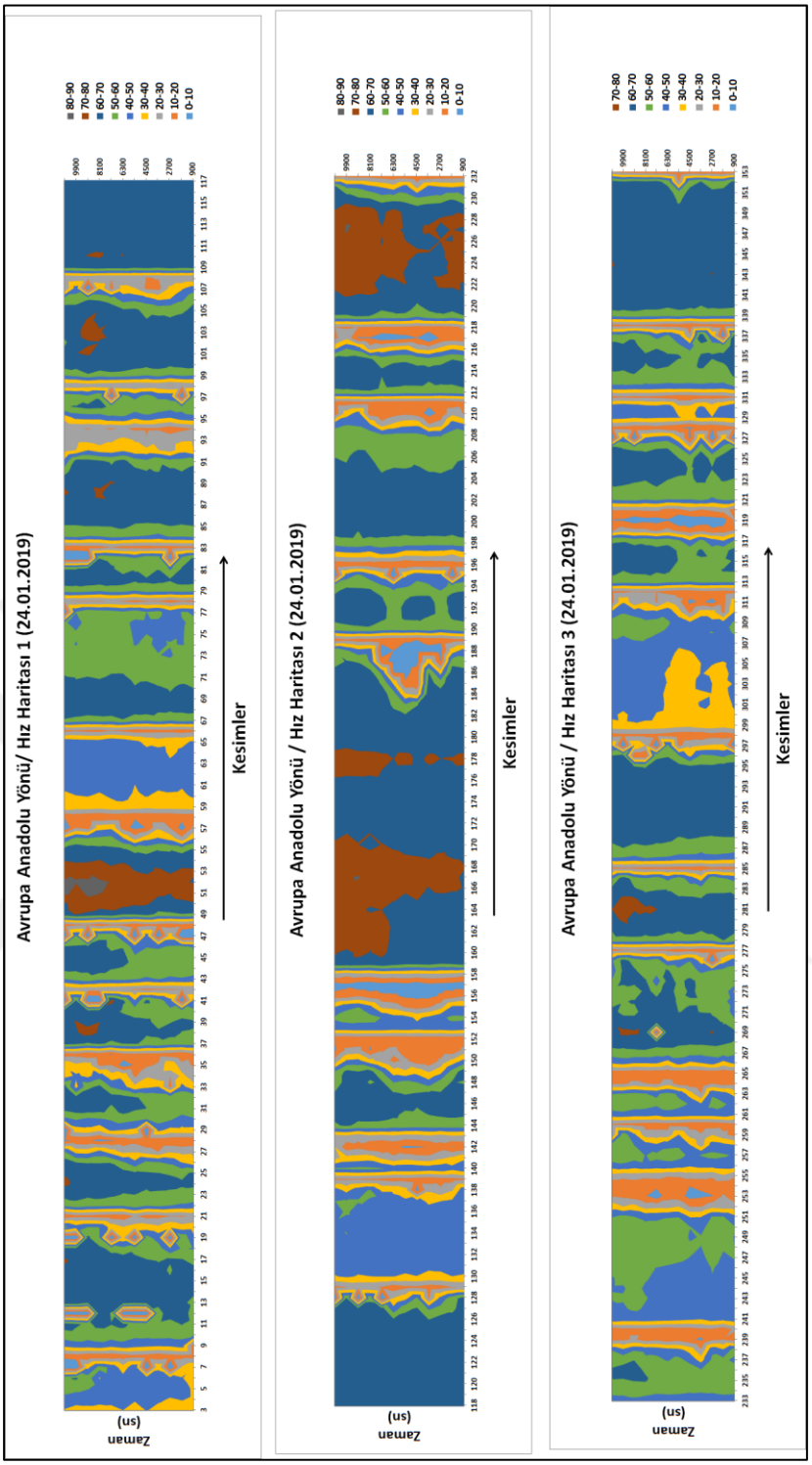

Avrupa Anadolu Yönü
27.01.2019
Zincirlikuyu Durağı
Cevizlibağ Durağı
Avcılar Durağı
Tüyap Durağı
Yol (m)
40000
35000
30000
25000
20000
15000
10000
5000
0
07:00:00
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)

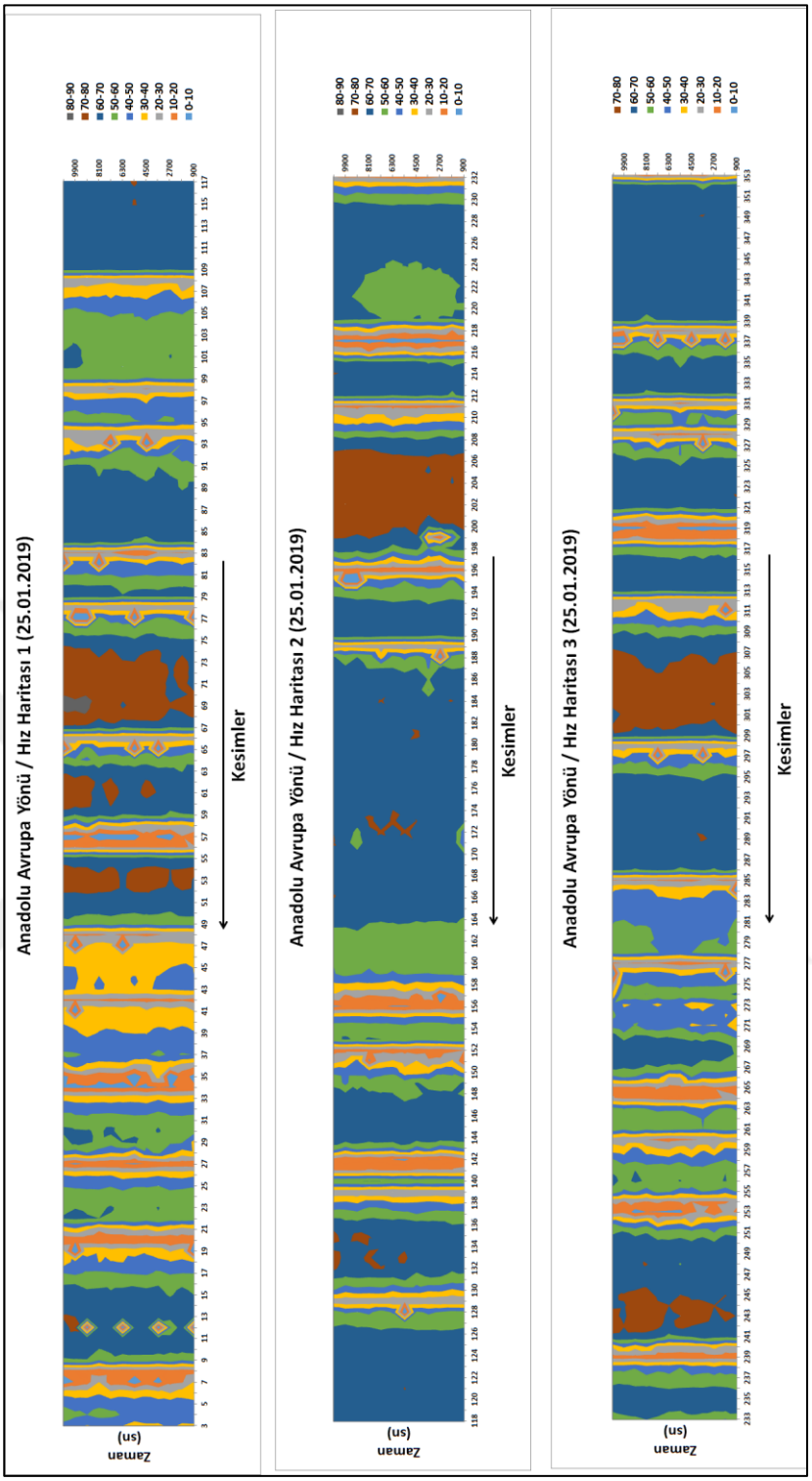

Anadolu Avrupa Yönü - 28.01.2019
Yol (m)
50000
45000
40000
35000
30000
25000
20000
15000
10000
Tüyap Durağı
Avcılar Durağı
Cevizlibağ Durağı
Zincirlikuyu Durağı
07:00:00
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)

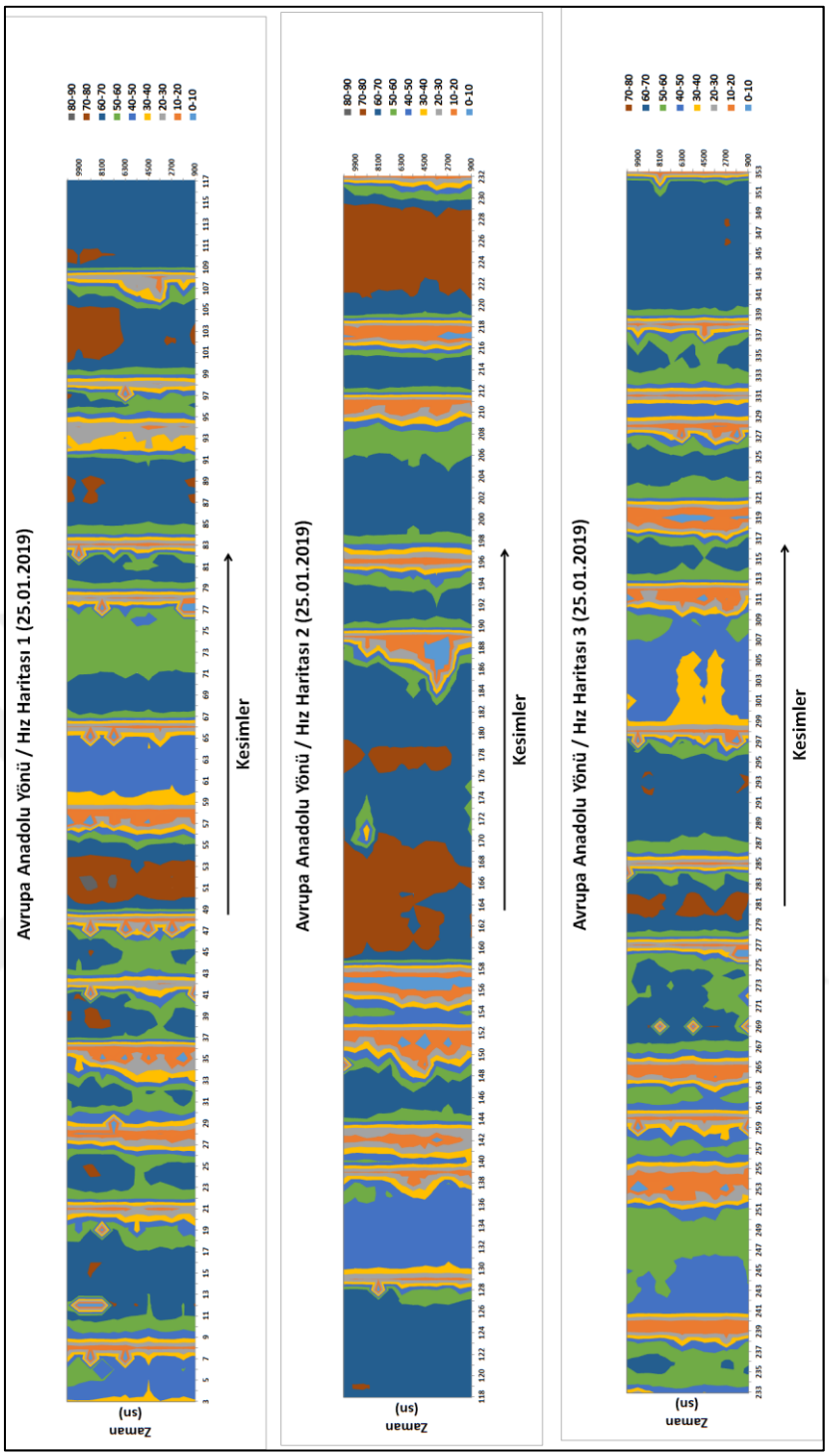

Avrupa Anadolu Yönü
28.01.2019
Yol (m)
40000
35000
30000
25000
20000
15000
10000
5000
0
Zincirlikuyu Durağı
Cevizlibağ Durağı
Avcılar Durağı
Tüyap Durağı
07:00:00
07:15:00
07:30:00
07:45:00
08:00:00
08:15:00
08:30:00
08:45:00
09:00:00
09:15:00
09:30:00
09:45:00
10:00:00
ZAMAN,(ss:dd:sn)

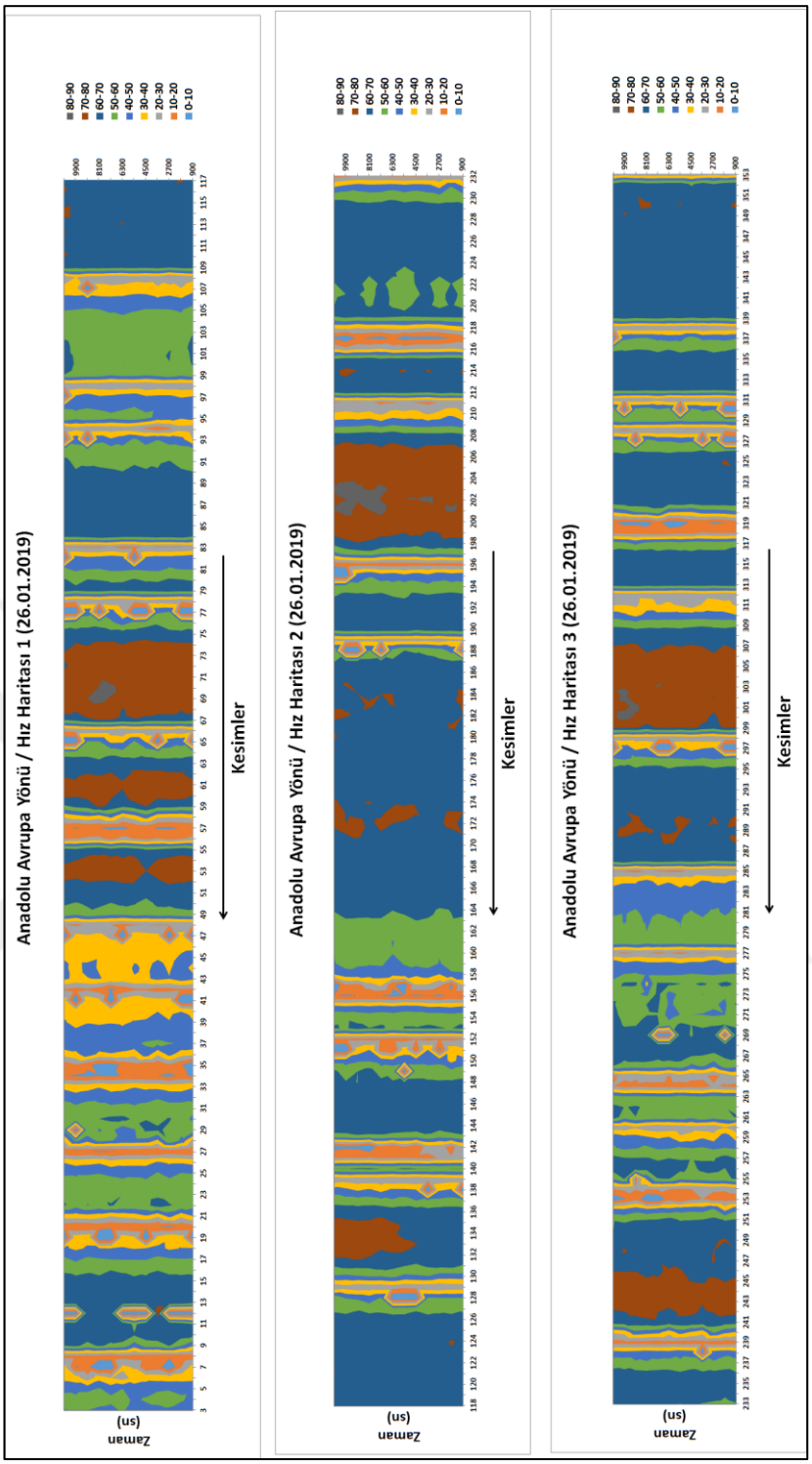
B
Hız Haritaları

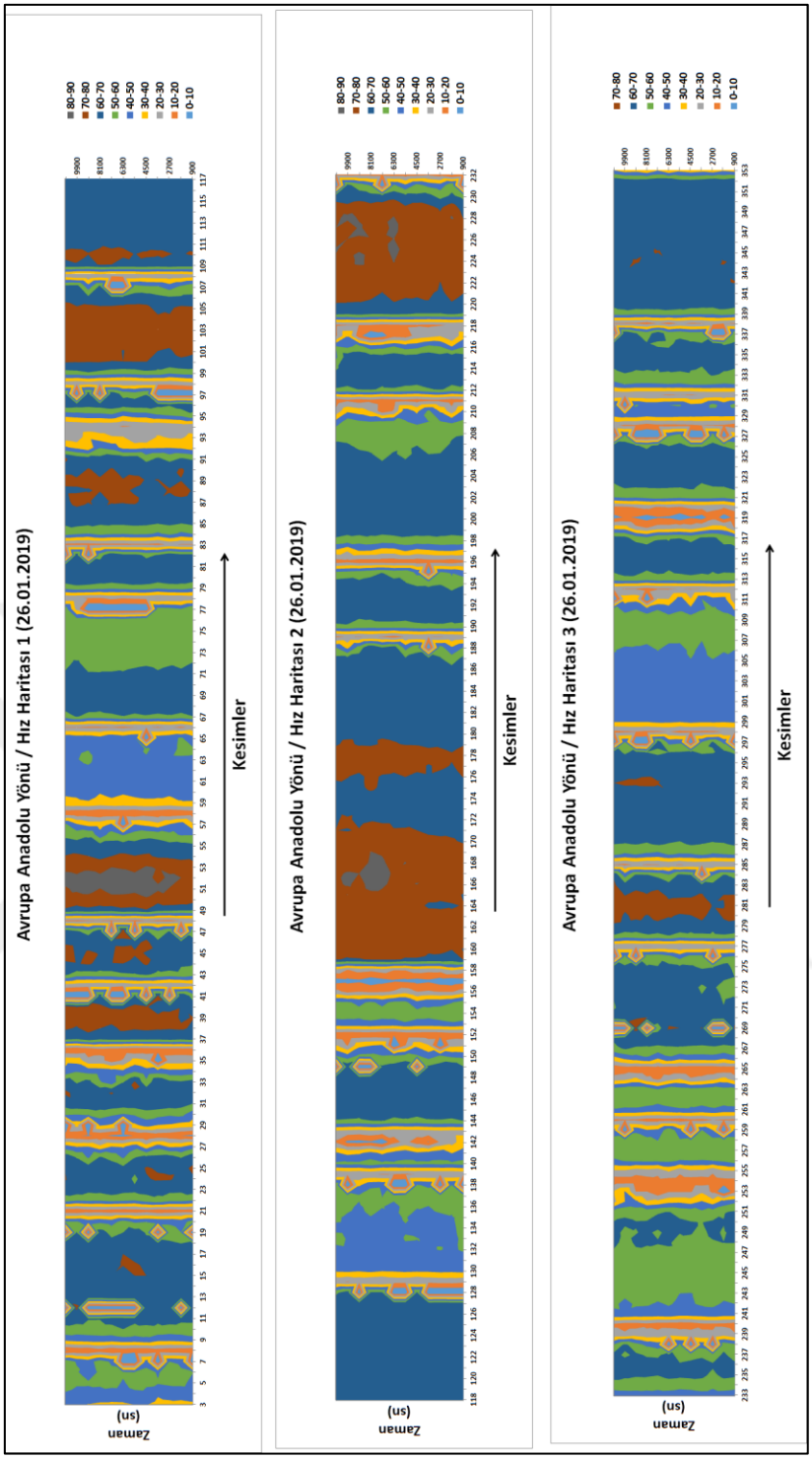



Avrupa Anadolu Yönü/ Hız Haritası 1 (24.01.2019)
Avrupa Anadolu Yönü / Hız Haritası 2 (24.01.2019)
Avrupa Anadolu Yönü / Hız Haritası 3 (24.01.2019)
Zaman (sn)
Kesimler
80-90
70-80
60-70
50-60
40-50
30-40
20-30
10-20
0-10

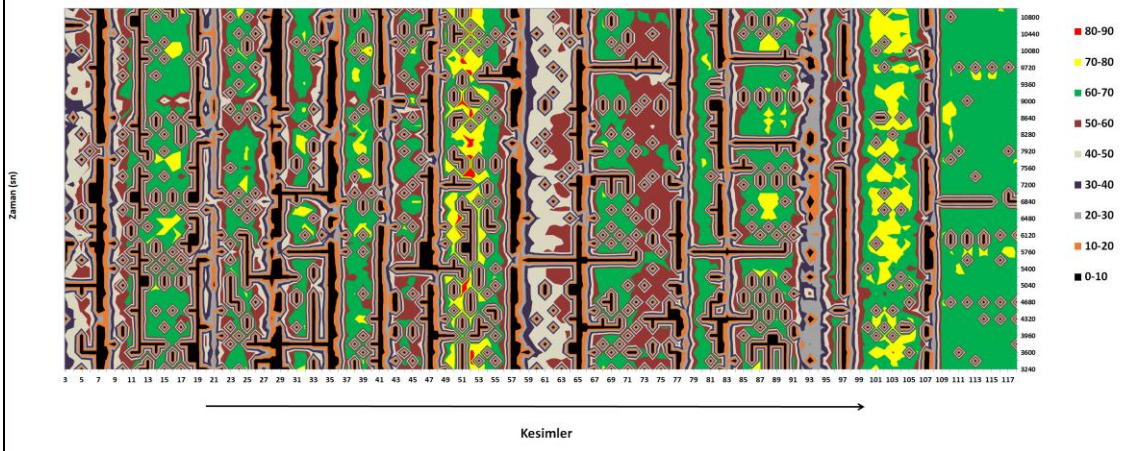

Anadolu Avrupa Yönü / Hız Haritası 1 (25.01.2019)
Anadolu Avrupa Yönü / Hız Haritası 2 (25.01.2019)
Anadolu Avrupa Yönü / Hız Haritası 3 (25.01.2019)
Kesimler
Zaman (sn)
80-90
70-80
60-70
50-60
40-50
30-40
20-30
10-20
0-10


Avrupa Anadolu Yönü / Hız Haritası 1 (25.01.2019)
Avrupa Anadolu Yönü / Hız Haritası 2 (25.01.2019)
Avrupa Anadolu Yönü / Hız Haritası 3 (25.01.2019)
80-90
70-80
60-70
50-60
40-50
30-40
20-30
10-20
0-10
Kesimler
Zaman (sn)

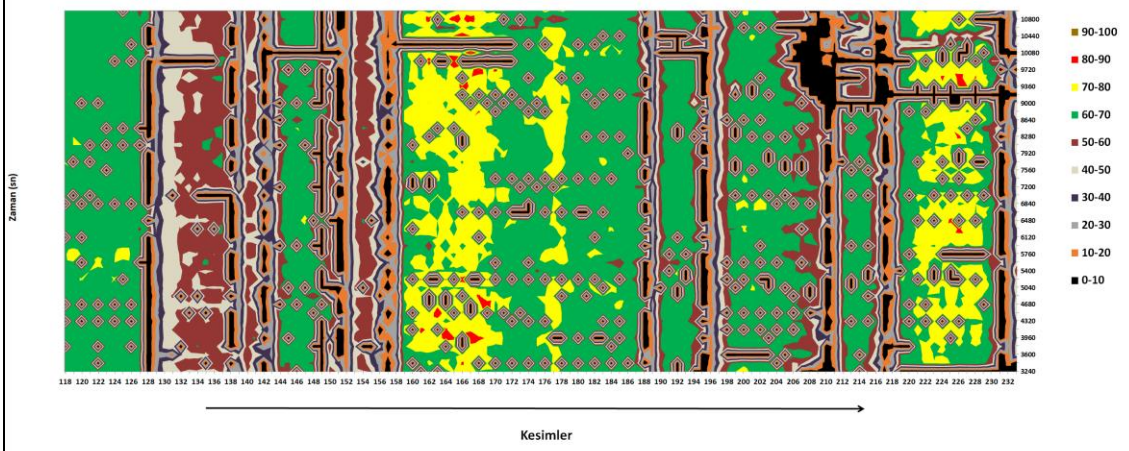

Anadolu Avrupa Yönü / Hız Haritası 1 (26.01.2019)
Anadolu Avrupa Yönü / Hız Haritası 2 (26.01.2019)
Anadolu Avrupa Yönü / Hız Haritası 3 (26.01.2019)
80-90
70-80
60-70
50-60
40-50
30-40
20-30
10-20
0-10
9900
8100
6300
4500
2700
900
Kesimler
Zaman (sn)


Avrupa Anadolu Yönü / Hız Haritası 1 (26.01.2019)
Avrupa Anadolu Yönü / Hız Haritası 2 (26.01.2019)
Avrupa Anadolu Yönü / Hız Haritası 3 (26.01.2019)
Kesimler
Zaman (sn)
80-90
70-80
60-70
50-60
40-50
30-40
20-30
10-20
0-10

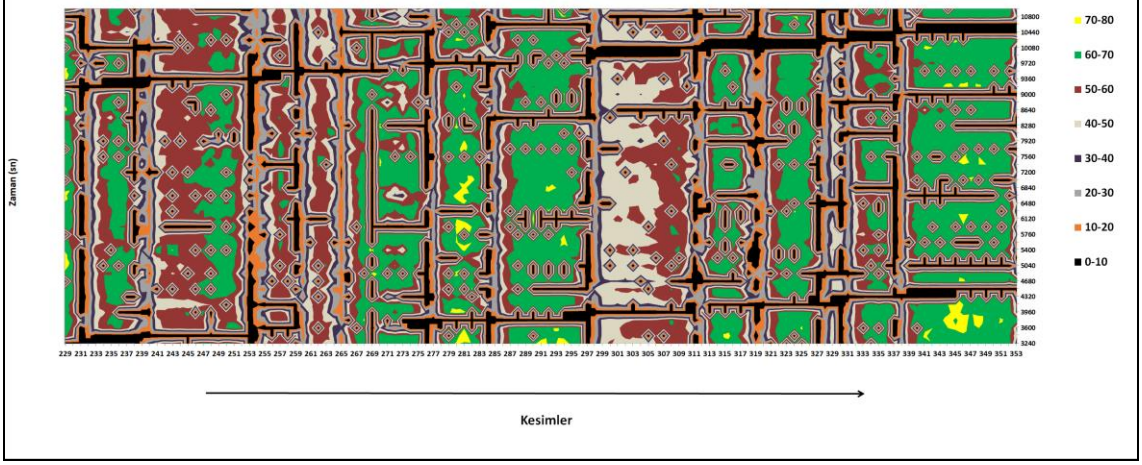
Avrupa Anadolu Yönü / 24.02.2019 Tarihli Kaza-Arıza Kaydı Hız Haritası 1



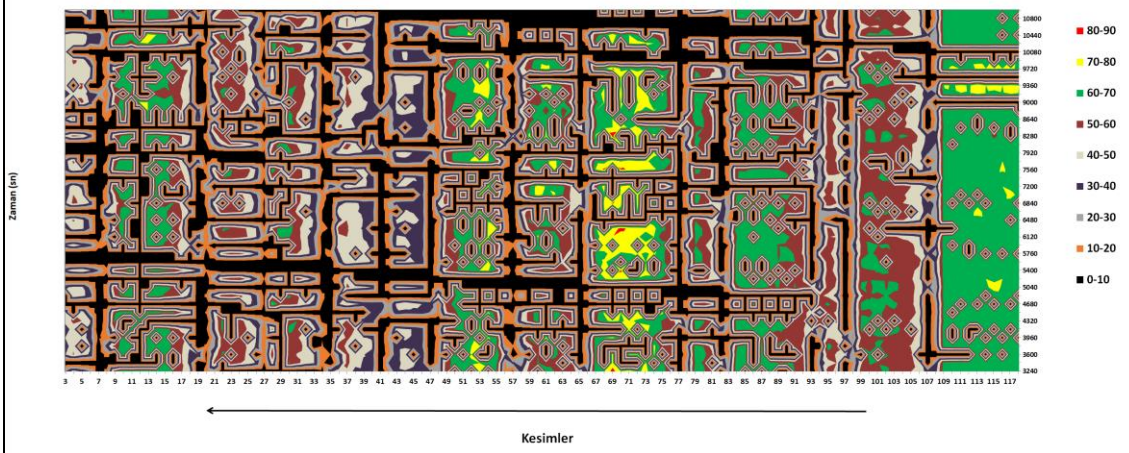
Avrupa Anadolu Yönü / 24.02.2019 Tarihli Kaza-Arıza Kaydı Hız Haritası 2



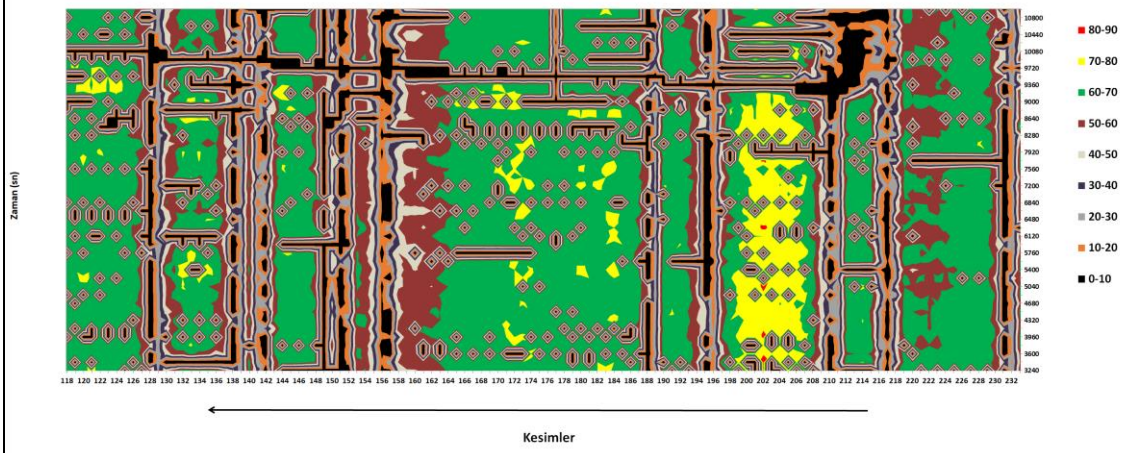
Avrupa Anadolu Yönü / 24.02.2019 Tarihli Kaza-Arıza Kaydı Hız Haritası 3



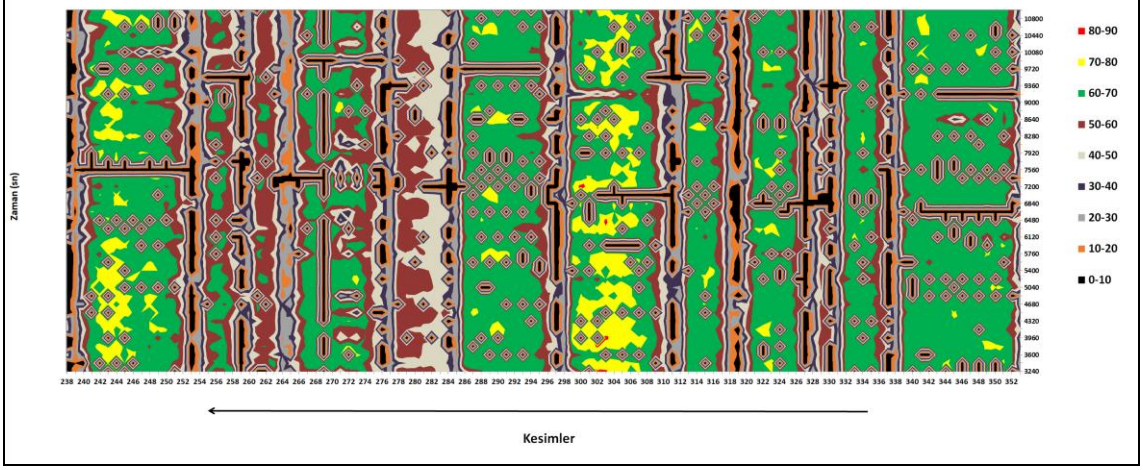
Anadolu Avrupa Yönü / 24.02.2019 Tarihli Kaza-Arıza Kaydı Hız Haritası 1



Anadolu Avrupa Yönü / 24.02.2019 Tarihli Kaza-Arıza Kaydı Hız Haritası 2



Anadolu Avrupa Yönü / 24.02.2019 Tarihli Kaza-Arıza Kaydı Hız Haritası 3



TEZDEN ÜRETİLMİŞ YAYINLAR

İletişim Bilgisi: sadullahgoncu@gmail.com

Konferans Bildirileri

1. Göncü S., Şahin İ. (2019).İstanbul Metrobüs Koridorundaki Trafik Akımlarının GPS Verilerinden Yararlanılarak İncelenmesi. *TMMOB İnşaat Mühendisleri Odası 13. Ulaştırma Kongresi*, 10-12 Ekim 2019, Erzurum, Türkiye