

YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

757677

151671

KESİN OLMAYAN TALEPLER İLE ARAÇ ATAMA PROBLEMİNE DİNAMİK BİR YAKLAŞIM

Endüstri Müh. Özhan Nuri ÖZESENLİ

FBE Endüstri Mühendisliği Anabilim Dalı Endüstri Mühendisliği Programında
Hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Yrd. Doç. Dr. Tufan DEMİREL

Yrd. Doç. Dr. Tufan DEMİREL

Prof. Dr. Hüseyin TOAŞLIL

Prof. Dr. Oya KALEPSİZ

İSTANBUL, 2004

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iv
KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
ÇİZELGE LİSTESİ	vii
ÖNSÖZ	viii
ÖZET	ix
ABSTRACT	x
1. GİRİŞ	1
2. ARAÇ ÇİZELGELEME PROBLEMİ	3
2.1 Statik (Geleneksel) Araç Çizelgeleme Problemi (SAÇP)	4
2.2 Dinamik Araç Çizelgeleme Problemi (DAÇP)	4
2.3 Statik (Geleneksel) ve Dinamik Araç Rotalama Yaklaşımlarının Kıyaslanması	9
2.4 Dinamik Araç Rotalama Probleminin Günlük Hayatta Uygulanabileceği Alanlar	12
2.4.1 Ambulans, İtfaiye ve Polis Hizmetleri	13
2.4.2 Altyapı Onarımı Problemi	13
2.4.3 Kurye - Posta Hizmetleri	14
2.4.4 Merkezi Bir Terminalden Mal Teslimatı	14
2.4.5 Al ve Teslim Et Problemleri	15
2.4.6 Ticari Taksi Hizmetleri	15
3. AKADEMİK ve TEKNİK ALTYAPI	17
3.1 Gerçek Zamanlı Optimizasyon Metotları	17
3.1.1 Dinamik Gezgin Satıcı Problemi (DGSP)	17
3.1.1.1 Performans Ölçütleri	18
3.1.1.2 Hafif Trafik Durumu	19
3.1.1.3 Yoğun Trafik Durumu	19
3.1.1.4 Yeniden Konumlandırma	20
3.1.2 Dinamik Gezgin Tamirci Problemi (DGTP)	20
3.1.2.1 Tek ve Kapasite Sınırı Olmayan Araç ile Dinamik Gezgin Tamirci Problemi	22
3.1.2.2 Kapasite Sınırı Olan Çok Araçlı Sistemler İçin Genellemeler	26
3.2 Dinamiklik Derecesi	28
3.3 Teknolojik Gereksinimler	31
3.3.1 Pozisyonlama Teknolojisi	31
3.3.1.1 Küresel Pozisyonlama Sistemi (GPS) ve Bileşenleri	33
3.3.1.2 GPS Çalışma Prensipleri	34
3.3.2 İletişim Teknolojisi	35

3.3.3	Coğrafi Bilgi Sistemleri (CBS).....	37
4.	BEKLENEN YOLCULUK MESAFESİ ve ZAMANLARININ HESAPLANMASI	39
4.1	Acil Servis Modeli.....	39
4.1.1	Acil Servis Modelinin Basitleştirilmiş Uygulaması	41
4.2	Optimal Alan Seçimi ve Optimal Olmayan Alanlar için Sonuçlar.....	45
4.3	Daha Gerçekçi Yolculuk Zamanı Modeli.....	52
4.4	Literatürde Yer Alan Ortalama Hız ve İvme Değerleri	55
4.5	Yolculuk Zaman Verisi için Doğrudan Ölçüm ve Tahmin	58
4.5.1	Yolculuk Zamanı Verisi için Doğrudan Ölçüm ve İlgili Teknikler	59
4.5.2	Modelimiz için v_c ve a Değerlerinin Tespit Edilmesi.....	60
5.	DİNAMİK TALEPLER İÇİN ARAÇ ATAMA YAKLAŞIMI ve SİMÜLASYON UYGULAMASI	68
5.1	Sistem Kavramı ve Simülasyon.....	68
5.1.1	Sistem Kavramı	68
5.1.2	Sistem Simülasyon Modeli	69
5.2	Dinamik Bir Gezgin Tamirci Probleminin Sistem Perspektifinden İncelenmesi..	70
5.2.1	Problemin Tanımı	72
5.2.2	Probleme Uygun Soyut Model Seçimi	72
5.2.3	Simülasyon Dilinin Seçimi	74
5.3	Simülasyon Modeli	76
5.3.1	Simülasyon Modelinin Bileşenleri	76
5.3.2	Simülasyon Modelinin Varsayımlarının Belirlenmesi	78
5.3.3	Simülasyon Modeli Yardımı ile İrdelenen Dinamik Araç Atama Yaklaşımı.....	79
5.3.4	Simülasyon Modelinin Formülasyonu.....	80
5.3.4.1	Olayların Akışı.....	81
5.3.4.2	Simülasyon Modelinde Yer Alan Değişken ve Parametreler ile Diğer Elemanlarının Sembolleri	81
5.3.5	Simülasyon Algoritmasının Açıklanması	84
5.3.5.1	SAT Politikası için Simülasyon Algoritması.....	85
5.3.5.2	NN ve SQM Politikaları için Simülasyon Algoritması	91
5.3.6	Akış Diyagramı.....	93
5.3.7	El ile Simülasyon.....	95
5.4	Simülasyon Modelinin Doğrulanması ve Gerçeklenmesi	95
5.4.1	Simülasyon Modelinin Doğrulanması	95
5.4.2	Simülasyon Modelinin Gerçeklenmesi.....	96
5.5	Simülasyon Sonuçları	98
5.5.1	Sabit-Durum Simülasyonlarında Çıktı Analizleri	98
5.5.2	Sabit-Durum Simülasyonlarında Aralık Tahmini için Dizi Ortalamaları.....	99
5.5.3	Çeşitli Müşteri Yoğunlukları için Alternatif Atama Politikalarının Kıyaslanması	100
5.6	Bekleme Zamanı ve Kapasite Arasında Tercih	109
5.6.1	Kapasite ve Optimal Kapasite Kavramı	110
5.6.2	PCServis Şirketi için Amaç Fonksiyonunun Tespit Edilmesi.....	112
5.6.3	PCServis Şirketi için İşletme Maliyet Merkezlerinin Tespit Edilmesi.....	112
5.6.3.1	PCServis Şirketinin Araç Tercihi ve İlgili Maliyetin Hesaplanması.....	113
5.6.3.2	PCServis Şirketinin Günlük Operasyonel Maliyetinin Belirlenmesi	115
5.6.4	PCServis Şirketi İçin Optimal Kapasitenin Tespit Edilmesi	117

6.	SONUÇ ve ÖNERİLER	120
	KAYNAKLAR	123
	EKLER	128
Ek 1	İstanbul Kenti İçin Ortama Hız Ölçümü.....	129
Ek 2	SAT Yaklaşımına Göre Simülasyon Programı Kodları.....	130
Ek 3	NN ve SQM Yaklaşımlarına Göre Simülasyon Programı Kodları.....	139
Ek 4	El ile Simülasyonu Yapılan Programın Bilgisayar Çıktısı	147
Ek 5	El ile Simülasyon Sonuçları.....	149
Ek 6	Hafif Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	150
Ek 7	Hafif Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	155
Ek 8	Hafif Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	160
Ek 9	Az Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	165
Ek 10	Az Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	170
Ek 11	Az Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	175
Ek 12	Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	180
Ek 13	Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	185
Ek 14	Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	190
Ek 15	Çok Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	195
Ek 16	Çok Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	200
Ek 17	Çok Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)	205
Ek 18	Yoğun Trafik Altında SAT Politikası Doğrultusunda 3 Araçlık Filo İçin Koşturulan Simülasyonun Raporu.....	210
Ek 19	Yoğun Trafik Altında SAT Politikası Doğrultusunda 4 Araçlık Filo İçin Koşturulan Simülasyonun Raporu.....	215
Ek 20	Yoğun Trafik Altında SAT Politikası Doğrultusunda 5 Araçlık Filo İçin Koşturulan Simülasyonun Raporu.....	220
Ek 21	Yoğun Trafik Altında SAT Politikası Doğrultusunda 6 Araçlık Filo İçin Koşturulan Simülasyonun Raporu.....	225
Ek 22	Yoğun Trafik Altında SAT Politikası Doğrultusunda 7 Araçlık Filo İçin Koşturulan Simülasyonun Raporu.....	230
Ek 23	Yoğun Trafik Altında SAT Politikası Doğrultusunda 8 Araçlık Filo İçin Koşturulan Simülasyonun Raporu.....	235
	ÖZGEÇMİŞ.....	240

SİMGE LİSTESİ

a	Karar vericinin alternatiflerinin sayısı
a	Servis aracının ivmesi
A	Servis Alanı
c	Karar vericinin maliyet fonksiyonu
c	Beklenen yolculuk mesafesi katsayısı
D	Yolculuk mesafesi
d'	Akış hızına ulaşamayan servis aracının maksimum hızına ulaşana dek kat ettiği mesafe
d_c	Servis aracının akış hızına ulaşana dek kat ettiği yol
G	n adet talep ve terminallerin oluşturduğu servis ağı
K	Toplam tamirci sayısı
l	Bir araç tarafından yanıtlanacak tur parçası
m	Toplam araç sayısı
n	Karar vericinin kararını etkileyecek değişkenlerin sayısı
n	Servis ağındaki taleplerin sayısı
N_k	n talepten oluşan k . GSP turu
p	trafik yoğunluğu (λ / μ)
p_t	Eldeki atama problemi
q	Servis aracının kapasite kısıtı
r	Herhangi bir müşterinin en yakın terminale uzaklığı
s	Ortalama servis süresi
s_i	Talep i için servis süresi
s^2	Servis süresinin ikinci momenti
SD	Terminal kümesi
T	Ortalama sistem zamanı
T^*	Optimal sistem zamanı
T_i	Talep i için sistem zamanı
t_{ij}	i ve j noktaları arasındaki yolculuk zamanı
t'	Akış hızına ulaşamayan servis aracının maksimum hızına ulaşma süresi
v	Karar vericinin değer fonksiyonu
v	Servis aracının sabit hızı
v_c	Servis aracının akış hızı
v'	Akış hızına ulaşamayan servis aracının ulaştığı maksimum hız
W	Ortalama bekleme süresi
W_i	Talep i için bekleme süresi
x	Servis alanı içinde herhangi bir nokta
X^*	Servis alanının medyanı
X_a	Servis alanının uzun kenarı (boyu)
x_t	Eldeki atama probleminin sonucu
Y_a	Servis alanının kısa kenarı (eni)
α	Servis alanının boy/en oranı
γ	Yoğun trafik durumu için Bertsimas sabiti
Δ	Herhangi bir alan ile optimal alan için beklenen yolculuk mesafeleri arasındaki fark
λ	Poisson dağılımı ile oluşturulan müşteri taleplerinin sisteme geliş oranı
μ	Exponential dağılıma göre birim zamanda yanıt verilen müşterilerin sayısı

KISALTMA LİSTESİ

A.B.D	Amerika Birleşik Devletleri
AÇP	Araç Çizelgeleme Problemi
bd	Bugünkü Değer (Present Value)
CWAAS	Canada Wide Area Surveillance System
CBS	Coğrafi Bilgi Sistemi
cdf	Cummulative Density Function
DAV	Danimarka Yol ve Adres Veritabanı
DAÇP	Dinamik Araç Çizelgeleme Problemi
dd	Dinamiklik Derecesi
DGTP	Dinamik Gezgin Tamirci Problemi
EGNOS	European Geostationary Navigation Overlay Satellite System
GLONASS	GLObal NAVigation Satellite System
GNSS	Global Navigation Satellite System (Galileo)
GPS	Küresel Pozisyonlama Sistemi
GSM	Global System for Mobile Communication
GSP	Gezgin Satıcı Problemi
GTP	Gezgin Tamirci Problemi
HCM	Highway Capacity Manual
İĞİÇ	İlk Giren İlk Çıkar
kbp	kilo bits per second
LORAN	Long Ranger Navigation
MSAS	Multifunctional Transport Satellite-based Augmentation System
MOD-GSP	Modifiye Gezgin Satıcı Problem
NAVSTAR	Navigation Satellite Timing and Ranging
NN	Nearest Neighbour (En Yakın Komşu)
NP	Non-Polynomial
qRP	Kapasite Kısıtlı Alan Ayırma
qTP	Kapasite Kısıtlı Tur Ayırma
SAÇP	Statik Araç Çizelgeleme Problemi
SAT	Shortest Arrival Time (En Kısa Ulaşım Zamanı)
SatNav	Satellite Navigation
SDHPT	State Department of Highways and Public Transportation
SFC	Space Filling Curve (Alan Doldurma Eğrisi)
SMS	Short Message Service
SQM	Stochastic Queue Median (Rassal Kuyruk Medyanı)
TCK	Türkiye Cumhuriyeti Karayolları
TDOA	Time Difference of Arrival
TEM	Trans European Motorway
UÖC	Uzaklık Ölçüm Cihazı
WAAS	Wide Area Surveillance System
ZPAÇP	Zaman Pencereleeri ile Araç Çizelgeleme Problemi
ZUS	Zeki Ulaşım Sistemi

ŞEKİL LİSTESİ

Şekil 2.1 İki araç için, 9 adet önceden belirlenmiş ve 2 adet de ani müşteri talebini göre hazırlanmış dinamik bir araç rotalama senaryosu (Larsen, 2001)	7
Şekil 3.1 Dinamik taleplerin iki farklı senaryo için sisteme giriş zamanları (Lund vd., 1996)	30
Şekil 4.1 Talep ve ambulansımızın konumları ile ambulansın izleyeceği yol	40
Şekil 4.2 X_1 ve X_2 için birleşik örnek uzay (1 x 1) lik kare	43
Şekil 4.3 $2d_c$ den daha kısa mesafeler için hız-zaman grafiği	54
Şekil 4.4 $2d_c$ den daha uzun mesafeler için hız-zaman grafiği	55
Şekil 4.5 Ulaşım analizlerinde yararlanılan doğrudan ölçüm ve tahmin teknikleri arasındaki ilişki (Turner vd., 1998)	59
Şekil 5.1 Bir sistemi incelemenin değişik yolları	69
Şekil 5.2 PCServis Şirketi araç atama sistemindeki bilgi akışı	71
Şekil 5.3 Atama politikalarının (SAT, NN ve SQM) algoritmalarının akış diyagramı ile gösterimi	94
Şekil 5.4 Hafif trafik altında atama algoritmalarının performansı	103
Şekil 5.5 Az yoğun trafik altında atama algoritmalarının performansı	105
Şekil 5.6 Yoğun trafik altında atama algoritmalarının performansı	107
Şekil 5.7 Çok yoğun trafik altında atama algoritmalarının performansı	109
Şekil 5.8 Servis seviyesi ile servis şirketinin işletme maliyeti arasındaki ilişki (Heizer ve Render, 2001)	111

ÇİZELGE LİSTESİ

Çizelge 4.1 Ortalama yolculuk zamanlarının tespitinde kullanılacak katsayılar (Larson ve Odoni, 1981)	48
Çizelge 4.2 Herhangi bir kompakt alan için ortalama yolculuk zamanlarının tespitinde kullanılacak katsayılar	49
Çizelge 4.3 Farklı α değerleri için beklenen yolculuk mesafesinin optimal durumdan sapması ve beklenen yolculuk mesafesi	51
Çizelge 4.4 A.B.D'deki bazı itfaiye departmanlarına ait seyir hızı ve ivmeleri (Walker, 1979)	57
Çizelge 4.5 Yol karakteristiklerine göre ortalama hız değerleri (Hoban vd.,1994; Brabyn, 2002)	58
Çizelge 4.6 Belirli mesafeler için modelimizdeki parametrelere göre hesaplanan toplam yolculuk zamanları	64
Çizelge 4.7 A.B.D'nin en büyük üç kenti için işyerine ulaşım karakteristikleri [11]	65
Çizelge 4.8 A.B.D için işyerine ulaşım karakteristikleri [11]	66
Çizelge 5.1 Simülasyon paketleri (Maria, 1997)	75
Çizelge 5.2 Sistem elemanları ve açıklamaları	82
Çizelge 5.3 Hafif trafik ($\lambda=1$) altında atama politikalarının kıyaslanması	102
Çizelge 5.4 Az yoğun trafik ($\lambda=2$) altında atama politikalarının kıyaslanması	104
Çizelge 5.5 Yoğun trafik ($\lambda=3$) altında atama politikalarının kıyaslanması	106
Çizelge 5.6 Çok yoğun trafik ($\lambda=4$) altında atama politikalarının kıyaslanması	108
Çizelge 5.7 Alternatif araç modelleri ve herbir modelin şirkete üç yıllık maliyeti	114
Çizelge 5.8 PCServis Şirketinin sabit operasyonel maliyetleri	116
Çizelge 5.9 PCServis Şirketinin değişken operasyonel maliyetleri	117
Çizelge 5.10 PCServis Şirketinin işlettiği farklı sayıda araç için günlük toplam maliyetleri	118

ÖNSÖZ

Bu çalışmayı seçmemde ve tamamlamamda beni yönlendiren yardımını ve deseteğini esirgemeyen değerli Hocam Yrd. Doç. Dr. Tufan Demirel'e teşekkür ederim.

Bilkent Üniversitesi Bilgisayar Mühendisliği öğrencilerinden Eyüp Serdar Ayaz arkadaşşıma vermiş olduđu destekten dolayı teşekkür ederim.

Bütün çalışmalarımnda olduđu gibi bu çalışmada da beni cesaretlendiren ve desteklerini eksik etmeyen aileme teşekkür ederim.



ÖZET

Günlük hayatımızın gittikçe önemli bir parçası haline gelen dağıtım ve ulaştırma ağlarının önemli bir bileşeni olan araçların atanmasına yanıt verebilecek bir model ortaya atılmış ve tartışılmıştır. Araçların değişken hızlar ile öklitsel bir coğrafyada oluş zamanı, yeri ve gereksinim duyduğu servis zamanı rassal olan, olay meydana gelmeden önce kesin olarak bilinemeyen taleplere yanıt vermek amacı ile seyrettiği bir sistem tasarlanmıştır. Amacımız, zaman ekseninde müşterilerin bekleme zamanlarını minimize edecek bir politika bulmaktır. Çalışmamıza araç atama ve çizelgeleme problemleri ele alınarak ve dinamik atama probleminin geleneksel (statik) probleminden farkları ortaya konularak başlanmıştır. Dinamik problemler için bir ölçüt olabilecek dinamiklik derecesi kavramı irdelenmiştir. Dinamik atama yaklaşımımızı mümkün ve fizibil kılan teknik ve teknolojiler, bazı maliyet analizleri ile birlikte değerlendirilmiştir. Takip eden bölümde İstanbul Kenti için mesafenin bir fonksiyonu olan temel bir yolculuk modeli kabul ve yöntemleri ile birlikte kurulmuştur. Çalışmamızın son bölümünde dinamik müşteri taleplerine etkin yanıt vermek üzere tasarlanan atama yaklaşımımızın ve literatürde kendine yer bulmuş alternatif atama politikalarının davranışları simülasyon modeli aracılığı ile gözlemlenmiştir. Yapılan simülasyon koşumları ile farklı yoğunluktaki müşteri talepleri altında yaklaşımımızın performansı tecrübe edilmiştir. Yaklaşımımızın hafif, az yoğun, yoğun ve çok yoğun müşteri talepleri altında alternatif politikalara göre daha etkin performans gösterdiği gözlenmiştir. Çalışmamız, temel maliyet yapıları dahilinde, yoğun trafik altında ve önerdiğimiz yaklaşım için optimal filo büyüklüğü tespit edilerek sonlandırılmıştır.

Anahtar kelimeler: Dinamik Araç Atama ve Rotalama, Küresel Pozisyonlama Sistemi, Yolculuk Zamanı Modeli, Simülasyon, Optimal Kapasite.

ABSTRACT

Distribution and transportation networks become an important part of our daily life gradually. We introduced and discussed a model, which can answer vehicle allocation problem, which is an important component of these networks. A system has been built in which vehicles traveling at a variable velocity in a Euclidean region to satisfy customer requests, whose time of arrival, location and on-site service time are stochastic and definitely unknown before the happening of the events. The objective is to find a policy to service demands over time period that minimizes the expected waiting time of customers. We began by dealing with vehicle allocation and programming problems also exposing the differences between conventional (static) and dynamic vehicle routing problem. Degree of dynamism concept, a possible performance measure for the dynamic problems, has been discussed. Technical and technological structures that enable and make feasible our dynamic allocation approach, with cost analysis have been analyzed. In the following chapter, a basic transportation model, which is a function of distance, has been formed with its assumptions and methods. In the final chapter of our study, behavior of alternative allocation policies and our allocation policy that has been projected to answer dynamic customer requests have been analyzed. Performance of the alternative allocation approach has been tested with several simulation runs under various customer demand levels. We also observed that our approach performs better under light, less high, high and very high customer demands than alternative policies. Our study has been finalized while determination of optimal fleet size for proposed policy under high traffic with general cost structures.

Keywords: Dynamic Vehicle Allocation and Routing, Global Positioning System, Traveling Time Model, Simulation, Optimal Capacity.

1. GİRİŞ

Yeryüzündeki gelişmiş ülkelerde yaşayan insanların %75'i, toplam dünya nüfusunun da %48'i, günümüzde kentlerde ya da büyük şehirlerde yaşamaktadır [6]. Bahsi geçen bu yüksek yoğunluğa sahip kümelerde gözlemlenen ekonomik ve sosyal yapıdaki karmaşa, bireyin kendi iyiliğinin diğer insanların aktivitelerine bağlı olmasının doğal bir sonucudur. Birbirlerine sıkı sıkıya bağlılık ve bağımlılık insanın tüm ihtiyaç alanlarında baş gösterir: beslenme, barınma, güvenlik, giyinme, eğlenme, bakım, enerji tedarigi gibi. Yukarıda değinilen birbirinden farklı kentsel ihtiyaçları gidermek, ciddi anlamda organize olmuş ve yapılanmış taşıma ve iletişim ağlarını gerektirir: polis, itfaiye, acil servis, posta toplama ve teslimatı, yol bakımı, şehir temizleme, kar kürüme, çöp toplama, otobüs ve metro taşımacılığı ve taksi hizmetleri önemli kentsel ihtiyaçlara örnek gösterilebilir.

Müşteri ve yurttaşlar, tip, miktar ve kalite olarak daha fazla hizmeti artan bir hız ile talep etmektedirler. Özellikle ülkemizde yaşanan ekonomik problem ve krizlerden sonra ek hizmetler için belediye ve firmalar bütçelerinden aktarabilecekleri kaynaklar konusunda daha tutucu ve tutumlu olma yolunu seçmişlerdir. Daha fazla ve daha iyi hizmet için talep ile bütçelerindeki kısıntılar arasında ciddi bir uçurum oluşmuştur. Bu uçurum, servis hizmetlerinin başarı ile yürütülmesi için gelişmiş bir yönetim ve karar alma süreçlerine güçlü bir ihtiyaç doğurmuştur. Çalışmamızın ana gayesi bu kararların alınmasında yardımcı olmak, kıt kaynakları daha etkin kullanılmasını sağlamak olarak özetlenebilir.

Yöneylem araştırmalarının kentsel problemlere eğilmesi yeni bir araştırma sahası değildir. 1736 yılında, ünlü matematikçi Leonhard Euler, Königsberg (Şimdiki Kaliningrad) kentinin yedi köprüsü üzerinden geçen ancak bir köprü'nün bir kereden fazla kullanılmadığı fizibil bir rota oluşturmaya çalıştığında da benzer bir problem ile karşılaşmıştı. Daha yakın bir zamanda (1937) George Dantzig ve arkadaşları, okul otobüslerinin rotalanması üzerine "Gezgin Satıcı Probleminin" (GSP) özel uygulamaları sonucunda Columbia Üniversitesi özel ödülü ile mükafatlandırılmışlardır. (Larson ve Odoni, 1981)

Şüphesiz ki alınan yol, Euler'den bu yana oldukça fazladır. Ancak bütün yaklaşımların ortak noktası, kaynakların geridönüşümsüz atanmasıdır. Howard (1966)'ya göre geridönüşümsüzlük, karar verildikten sonra, bir önceki duruma geri dönüşün imkansız ya da son derece maliyetli olduğu durumları tasvir etmek için kullanılır. Bu görüşün devamı olarak karar, bir dizi faaliyeti izlemek için zihinsel bir taahhüt olarak değil, bir dizi faaliyetin gerçek bir takibi olarak tanımlanabilir. Howard (1966)'ın yaklaşımının bir uzantısı olarak, insanı ve

çevresini etkileyen hizmetlerin yerine getirilmesinde etkili olan kaynaklar; personel, ekipman, araçlar ve değişik servis-geliştirme teknolojileri bu bağlamda yayılabilir ve atanabilir.

Çalışmamızda hayatımızın önemli bir parçası olan araç atama problemine, dinamik bir yaklaşım getirmeye gayret gösterilmiştir. Araştırmamızın amacı, geliştirdiğimiz atama politikasının uygulanabilirliği mümkün kılan teknolojik altyapıları ve bu altyapıların fizibilitesinin tartışılması, ortaya atılan yaklaşımın performansını literatürde kendine yer bulmuş ve performansları sınanmış alternatif atama politikaları ile kesin olmayan dinamik talepleri yanıtlamadaki başarısına göre karşılaştırmaktır. Çalışmamızda doğrudan ölçüm metodu ve literatürde yer alan çeşitli yöntemler ile tespit edilmiş sonuçlar kullanılarak, İstanbul Kenti için yolculuk mesafesine bağlı bir yolculuk zamanı modeli önerilmiş, bahsi geçen yolculuk modeli üzerine dinamik araç atama yaklaşımımızın alternatif sistem girdileri doğrultusunda sınanabileceği bir simülasyon modeli kurulmuştur.



2. ARAÇ ÇİZELGELEME PROBLEMİ

Çalışmamızın ilk kısmında, insanı ve onun ayrılmaz bir parçası olan çevresini etkileyen hizmetlerin yerine getirilmesinde en etkili kaynaklardan biri olan araçların çizelgelenmesi ve sonrasında bahsi geçen çizelgeler dahilinde ilgili görevlere atanması üzerinde durulmuştur. Araç Çizelgeleme Problemleri (AÇP), özel olarak klasik gezgin satıcı problemi (GSP), yöneylem araştırması literatüründe en fazla irdelenmiş ve tartışılmış problemlerden biridir. Kazandıkları popüleritenin sebebi problemin doğasında olan çekicilik ve içeriğinin zenginliğidir. Özellikle GSP, pek çok açıdan, prototip olarak ele alınmış zor bir kombinasyon problemidir.

Bu özel problem tipi (GSP) başta olmak üzere, AÇP ile pratik dağıtım sistemlerinde hem doğrudan hem de dolaylı olarak karşılaşmaktadır. Konumuzun gündelik hayat ile bu denli iç içe olması araştırma için güçlü bir uyarıcı durumundadır.

Araç çizelgeleme problemleri özellikle son 30-40 yıl içerisinde akademik çevrelerden ciddi ilgi görmüştür. Geleneksel olarak problem, statik ve deterministik olarak irdelenmiştir. Geleneksel yaklaşımın sonuçları, kombinatoral optimizasyon, sezgisel ve kompleks teori analizlerine ciddi katkılar yapmıştır. Bahsi geçen problemlerde çizelgeleme ile ilgili bütün veriler rotaların oluşturulmaya başlandığında belirlidir. Belirli müşteri noktalarını bir küme haline getirildiği takdirde, amacımız bu kümenin elemanlarını toplam yolculuk maliyetini minimize edecek şekilde, problem kısıtlarını da dikkate alarak ziyaret etmek olarak tanımlanabilir. (Bertsimas ve Van Ryzin, 1991)

Konumuzun pratik uygulamalarının pek çoğunda, rassal ve / veya dinamik veri işletilen rotaya paralel olarak oluşur. Öyle ki, AÇP'nin yer aldığı pratik uygulamaların önemli bir bölümünde (mobilize tamir ve dağıtım, acil servis, stoğun yeniden tedariği, kurye hizmetleri, özel taşımacılık, akaryakıt taşımacılığı gibi) problemin ciddi bir dinamik yüzü olduğunu gözlemlenebilir. Bu tür örneklerde, müşteri profili (servisin başlayacağı zaman, coğrafi konumu, gerçekleşecek talep vb.) planlama esnasında hatta servis başlamadan önce bilinemeyebilir. Yukarıda tanımlamaya çalıştığımız durumlarda müşteriye tatmin edecek nitelik ve karakteristikteki rotaların planlanması statik durumdakine kıyasla çok daha zordur. Bu zorluğun nedenleri sabitlerin değişken haline gelmesi ve zaman ufğunun darlığıdır. Tüm bu zorluklara rağmen, dinamik modelleri ele almamızın sebebi, pratik hayatta karşılaştığımız problemleri temsil etmeye daha uygun olmasıdır. Günümüzde, pek çok şirket müşterilerine, taleplerinden birkaç saat sonra hizmet sunmayı teklif eder. Müşteri, dolayısı ile pazar

tarafından belirlenen, karmaşık taleplerin çözümlenmesi ve karşılanması ilgili şirketlerin sürekliliği ve karlılığı için zorunludur. Şüphesiz ki, bu tarz müşteri odaklı hizmet politikaları, sistemin dinamikliğini ve dolayısı ile karmaşıklığını da arttırır. Takip eden kısımda çalışmamızın özünü teşkil eden geleneksel ve dinamik programlama problemlerine sırası ile değinilmiş ve kıyaslanmaya çalışılmıştır.

2.1 Statik (Geleneksel) Araç Çizelgeleme Problemi (SAÇP)

Larsen (2001)'e göre araç çizelgeleme konusunda, en temel ve en çok irdelenmiş problem, gezgin satıcı problemidir (GSP). Bu problemde satıcı bir şehirler kümesini dolaşır ve en sonunda turuna başladığı şehre geri döner, amacı dolaştığı toplam mesafeyi minimize etmektir. AÇP, GSP'nin genelleştirilmiş şeklidir denebilir. Araç çizelgeleme, her biri belli bir terminalde başlayan, daha önce tespit edilmiş sıradaki müşterileri ziyaret eden ve sonunda tekrar turuna başladığı terminaline geri dönen, m adet araç rotasının tespitinden oluşmaktadır. Çizelge her bir müşterinin sadece bir defa ziyaret edileceğini ve bir rotada karşılanan müşteri talebinin araç kapasitesini aşamayacağını göz önüne alarak oluştururuz. Burada amaç, toplam dağıtım maliyetini minimize etmeye çalışmaktır.

Araç kapasitesinin dışında günlük hayatta modelimizin kompleksliğini arttıracak ek kısıtlar da bulunabilir. Bu kısıtların en sık gözlemlenenleri, kaynaklar ile ilgili kapasite sınırları ve toplam rota zamanı üzerinde zaman kısıtları ile servisin başlaması gereken zamana işaret eden zaman pencereleridir. İkinci kısıta sahip problemler, Zaman Pencereleri ile Araç Çizelgeleme Problemi (ZPAÇP) olarak adlandırılabilir. Bununla birlikte birden fazla terminal ve ticari malın yer aldığı vakalar da modelin karmaşıklığını arttıracaktır. Bahsettiğimiz problemlerin çözümünde matematik programlama gibi kesin metodlara yer verilebileceği gibi özel olarak geliştirilmiş sezgisel yaklaşımlar ve çeşitli simülasyonlar da araç çizelgeleme problemine uygulanabilir.

2.2 Dinamik Araç Çizelgeleme Problemi (DAÇP)

Gerçek dağıtım sistemlerinde talep, zaman içinde rasgele oluşurken, ilgili taleplerin toplanması, bu veri doğrultusunda tur ve rotaların oluşturulması ve araçların salınması, sürekli bir proses olabilir. Bu tarz değişken bir atmosferde, teslim için bekleme (servis seviyesi) de en az yolculuk maliyeti kadar önemlidir. Bazı özel durumlarda, örneğin acil servis uygulamalarında, bekleme süreleri sistem performansını üzerindeki en önemli ölçüt olacaktır. Bekleme süresine ilişkin tartışmamızı çalışmamızın ileriki kısımlarına bırakarak statik ve

dinamik çizelgeleme problemleri arasındaki ayrımı daha net yapabilmek için Psaraftis (1988)'in tanımlarını referans alacağız. Bu doğrultuda:

Psaraftis (1988), statik çizelgeleme problemini: “Kesin formülasyonun çıktısı, yeniden optimize edilmeyen ve gerçek zamanlı olarak değişmeyen verilerden hesaplanan, planlama öncesi rotalar kümesi” olarak tanımlarken, dinamik problemi: “Çıktının bir rotalar kümesini oluşturmadığı, fakat gerçek zamanlı olarak değişen girdilerin bir fonksiyonu olarak rotaların nasıl değişmesi gerektiğine salık veren bir politika” olarak tasvir eder.

Yukarıdaki ifade ve ibareleri yeniden organize etmek ve statik çizelgeleme problemi olarak kastettiğimiz kavramı daha net ifade edebilmek için tanım 2.1’den yararlanılabilir.

Tanım 2.1 Statik Araç Çizelgeleme Problemi. (SAÇP)

- Rotaların planlanması için gereken her türlü ilgili bilgi, rotalama prosesi başlamadan önce planlayıcı tarafından biliniyor kabul edilir.
- Rotalama ile ilgili bilgi, rotalar kurulduktan sonra değişmez.

Gerekli olduğu kabul edilen bilgi, müşterinin coğrafi konumu, servis zamanı ve her müşterinin talebi gibi müşterilerin ilgili bütün niteliklerini içerir. Ek olarak, müşteriler arasında araçların yolculuk zamanı gibi sistem verisi de planlayıcı tarafından bilinmelidir.

Statik probleme benzer şekilde, dinamik problemimiz için de tanımımız aşağıda incelenebileceği üzere yapılabilir.

Tanım 2.2 Dinamik Araç Çizelgeleme Problemi. (DAÇP)

- Rotaların planlanması için gereken bütün bilginin, rotalama prosesi başlamadan önce planlayıcı tarafından bilinmediği kabul edilir.
- İlk rotaların oluşturulmasının ardından veriler değişebilir.

Tanım 2.1 ve 2.2’den çıkarabileceğimiz yorum: Şüphesiz ki, DAÇP’i, geleneksel SAÇP’ye kıyasla çok daha zengindir ve problem kümesi statik probleminkini kapsar.

Dinamik çizelgeleme, mümkün olduğu kadar fazla acil taleplere de cevap vermek isteyeceğinden, gerçek zamanlı olarak işletilebilecek, bağlantılı algoritmalar talep eder. Geleneksel-statik araç çizelgeleme problemleri, NP-zor (NP-hard) problem sınıfına girer. Öyleki pratik problemlerin boyutlarında, optimal çözümlerin belirli bir zaman dilimlerinde hesaplanması her zaman mümkün değildir. Bu yorumdan, dinamik araç çizelgeleme

probleminin de NP-zor problem sınıfına dahil edilebileceğini söyleyebiliriz. Zira, her yeni talebin sisteme girişinde bir statik problemin çözülmesi gerekecektir.

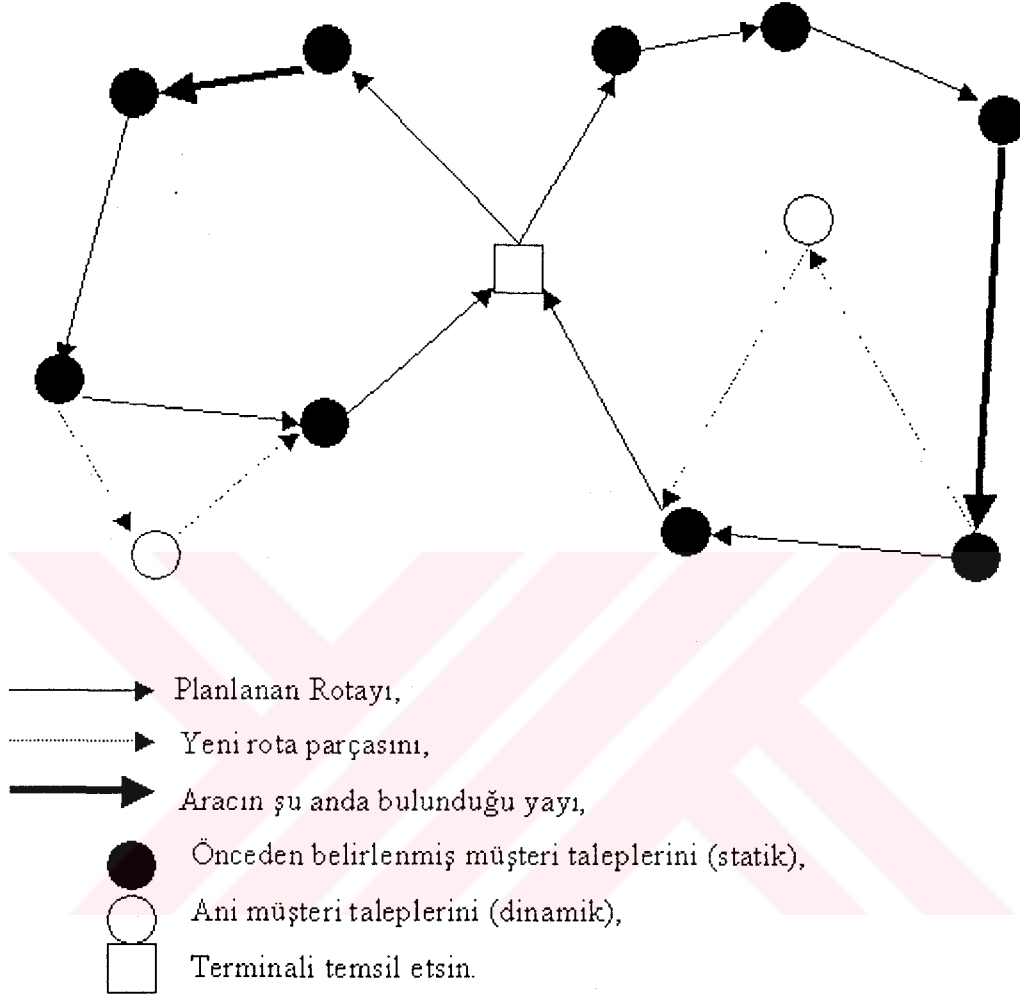
Bu noktada Psaraftis (1980)'in probleme yaklaşımı dikkat çekicidir. Çalışmasında, eldeki çizelgeleme problemini p_t olarak isimlendirir ve p_t problemi, x_t sonucuna işaret eder ve bu sonuç, 'kesin olmayan sonuç' adını alır. Kesin olmayan sonuçlar yalnız ve ancak eldeki veriler kümesine dayanır. Bu sonucun uygulanma prosesinde sisteme hiç yeni müşteri talebi alınmaz ise, 'kesin olmayan rota optimaldir.' denilir.

Şekil 2.1'de dinamik araç rotalama durumunu gösteren basit bir örnek gösterilmeye çalışılmıştır. Örneğimizde, önceden belirlenmiş ve ani müşteri taleplerini sağlamak ile yükümlü iki adet kapasite kısıtı olmayan aracı ($m=2$) irdeleyeceğiz. Örneğimizde, rotamıza ek kısıtlar getirecek olan zaman pencerelerine yer verilmeyecektir. İlgili şekilde, siyah noktalar ile önceden belirlenmiş müşteri taleplerini, beyaz noktalar ile de ani müşteri taleplerini göstereceğiz. Kare ise rotaların başladığı ve sonuçlandığı terminalimiz olsun. Düz çizgiler, araçlar terminalden ayrılmadan önce planlayıcı tarafından organize edilen iki rotayı (her araç için bir rota) temsil etsin. İdeal olarak, yeni gelen müşteri talepleri (ani talepler) daha önce planlanmış olan rotalara eklenirken, henüz ziyaret edilmemiş müşterilerin sırası değiştirilmeden müşteri gecikmelerinin toplamının minimum tutulmasına çalışılır. Saat-yönünün tersine işleyen rota, özellikle yeni talebin sisteme dahil olduğunda, servis aracının bulunduğu yay göz önüne alındığında, bahsi geçen ideal yaklaşıma güzel bir örnektir. Ne var ki, gündelik hayatta, yeni müşteri taleplerinin sisteme eklenmesi çok daha karmaşık bir prosesi de beraberinde getirir ve rotanın henüz ziyaret edilmemiş kısmının yeniden planlanması anlamına gelir. Bu da, saat-yönünde işleyen rotada izlenilebileceği gibi yeni müşteriye servis verebilmek dolambaçlı ve karmaşık bir yol ile mümkün olabilir.

Şunu belirtmemiz gerekir ki, rotalama problemimiz daha karmaşık ve kısıtlarımız da daha çok olduğunda, yeni dinamik müşterilerin sistem ile bütünleştirilmesi o denli kompleks olacaktır. Örnek vermemiz gerekirse, zaman pencereleri ile kısıtlanmış rotalama problemine yeni müşterilerin ilave edilmesi, zaman kısıtlarının olmadığı (şekil 2.1) probleme kıyasla genellikle daha zor olacaktır.

Yeni müşterilerin eklenmesi için fizibil bir boşluğun yakalanamadığı şartlarda, bağlantılı rotalama sistemi müşteriye hizmet vermeyi red bile edilebilir. Çoğunlukla müşteriye reddetme politikası, müşterinin talebini bir sonraki rotalama periyodunda (mesela ertesi gün) karşılama önerisi ile birlikte gelir. Buna rağmen, günümüzde yeni bir müşteri kazanmanın maliyetininin

yüzlerce dolar olduğu yoğun rekabetin yaşandığı endüstrilerde, özellikle şehiriçi kurye hizmeti, şehirlerarası kargo taşımacılığı gibi bazı sistemlerde, servis sağlayıcı (kurye, kargo şirketi) talebinini karşılayamadığı müşterisini kendi elleri ile rakibine teslim etmiş olacaktır.



Şekil 2.1 İki araç için, 9 adet önceden belirlenmiş ve 2 adet de ani müşteri talebini göre hazırlanmış dinamik bir araç rotalama senaryosu. (Larsen, 2001)

Larsen (2001)'e göre; dinamik araç rotalama problemini incelerken, gerçek veriye kıyasla rasgele oluşturulmuş veri daha sık kullanılmaktadır. Bu uygulamayı tercih etmenin iki ana sebebi vardır. İlk olarak, rasgele oluşturulmuş veri, verikümelerinin diğer konular ile de ilişkilendirecek şekilde yaratılabileceğinden, bize daha derinlemesine araştırma yapma şansını verir. İkinci olarak da, gerçek dinamik rotalama problemlerinin çok ciddi bir bölümü, özel bir araç rotalama problemi üzerinde derinlemesine analiz edilmesini mümkün kılan veriyi, günümüzde kapsayacak ve paylaşacak durumda değildir. Her yeni veri talebin sisteme ulaştığında, her bir aracın coğrafi konum bilgisi, pratik durumları irdeleyen araştırmalarda en

sık ulaşılamayan verilerden birisidir. Çalışmamızın ileriki bölümlerinde, bahsi geçen bu verinin nasıl toplanabileceği, bunu sağlayabilecek teknolojik donanım ve bu teknolojinin araç rotalama probleme uygulanmasının getirebileceği yararlar bazı maliyet analizleri ile birlikte tartışılmıştır.

Çalışmamızda irdelemiş olduğumuz vakalarda veriyi modelimizde öngördüğümüz şekilde rasgele oluşturduk. Rasgele oluşturulan verinin yaratılmasında ve sayısal modelimize dahil edilmesinde Microsoft Visual C++ 6.0 programlama dilinden yararlanılmıştır. Aşağıda, rasgele yaratılacak verilerden üçü kısaca tartışılmıştır. İncelediğimiz üç tip veri, diğer verileri de doğrudan etkileyeceği için araştırmamızda izleyeceğimiz yolu doğru bir şekilde seçmemiz adına bize yol gösterir. Çalışmamızın beşinci kısmında ilgili girdi değişken ve parametreleri ile bunların modelimiz dahilindeki konum ve etkileri detaylı olarak irdelenmiştir.

Öncelikle, ani müşteri taleplerini oluşturmak için seçeceğimiz metodolojiye karar vermemiz gerekir. Sistemi kurarken bu bilginin sıklıkla kullanıldığı ve çözüm metodları geliştirilirken yararlanıldığı için ani müşteri taleplerinin oluşturulması çok önemli bir meseledir. Geleneksel olarak, Poisson proses bu amaç için kullanılır. Sisteme yeni müşteri gelme oranı, genellikle λ ile gösterilir, sistemdeki 'tikanıklığı' tasvir eder. (Winston, 1994) Daha karmaşık sistemleri tasvir eden senaryolar irdelenilmek istendiğinde, sisteme giriş prosesi, birden fazla farklı rassal prosesin karışımı olarak da ele alınabilir.

Diğer önemli bir modelleme parametresi de, müşteri taleplerinin karakteristiklerinin ve müşteri noktalarındaki servis sürelerinin dağılımıdır. Düzgün ve Gaussian dağılımlar, sabit değerler gibi sisteme adapte edilebilir. Sabit değerler yardımı ile çözüme ulaşmak çoğu zaman daha kolay olmakla birlikte pek çok durum için gerçekçilikten çok uzaktır. (Winston, 1994) Örnek bir vaka teşkil edebilecek, bir yerden bir malın alınıp istenilen noktaya teslimatının yapıldığı kurye hizmetlerini ele alalım. Bu durumda, her müşteri için alandaki servis zamanı, pek çok değişkene sıkı sıkıya bağlıdır. Park yeri durumu, servisin yapılacağı binanın fiziksel yerleşimi ve ilgili malın teslimi ile ilgili verimlilik (teslimat için kuryenin beklemesinin gerekip gerekmediği, elden bir sevkiyatın istenilip istenilmediği vb.) bahsi geçen değişkenlerden bazılarıdır ve değişken sayısı arttıkça sabit değerler yardımıyla çözüme ulaşmak güçleşir ve modelin gerçek durumları temsil yeteneğini de ciddi oranda azaltır.

Öncül olarak irdeleneceğimiz son konu da, sistemin reaksiyon süresidir. Sistem reaksiyon süresi, talebin sisteme ulaşması ile servisin başlaması gereken en geç fizibil süre arasında geçen zamandır. Her müşteri için reaksiyon süresi, bu müşteriye hizmet götürme aciliyetinin

bir ölçüsü olarak kabul edilebilir. Sistemin toplam reaksiyon zamanı, sistem içinde sunulan dinamiklik seviyesi ile sıkı sıkıya bağlıdır. Genellikle, sistemin daha dinamik olması, hızlı bir reaksiyon oluşturulmasını daha maliyetli ya da zor kılar. Bu nedenle, sistemin dinamik yapısı, değişik senaryolar için karar alma mekanizmasını oluşturacak doğru model ve algoritma seçme konusunda en önemli rolü oynayacaktır. İleriki bölümlerde, uygulanabilir politikalar arasından yer alabilecek ve performans kriterlerimiz doğrultusunda bizi optimal bir çözüme götürmesini beklediğimiz bir politika kurup ve oluşturulan bu politikayı literatürde önceden kendine yer bulmuş alternatifleri ile kıyaslamaya çalıştık.

2.3 Statik (Geleneksel) ve Dinamik Araç Rotalama Yaklaşımlarının Kıyaslanması

Önceki bölümlerde araştırma konumuzun bel kemiğini teşkil eden geleneksek (statik) ve dinamik araç rotalama / çizelgeleme problemini tanımladık. Çalışmamızın bu kısmında ise, statik ve dinamik araç rotalama problemleri arasındaki farklar irdelenmiştir. Psaraftis (1988 ve 1995)'te dinamik araç rotalama problemi ile statik rotalama problemi arasındaki farklar 12 noktada listelenmiş ve ayrıntılı olarak tanımlanmıştır. Aşağıda bahsi geçen 12 fark ve bu farkların kısa özetlerine değinilecektir.

1.) Problemin zaman boyutu önemlidir.

Statik çizelgeleme ve atama probleminde, problemin zaman boyutunun önemi olabilir ya da olmayabilir. Dinamik çizelgelemede ise zaman daima önemlidir. Planlayıcı, tüm araçların pozisyonlarını mümkün olduğu kadar geniş bir zaman ekseninde, özellikle de servis için talebin kendisine ulaştığı anlarda bilmelidir. Daha önce de değinildiği üzere, gerçek rotalama problemlerinin pek çoğunda planlayıcı herhangi bir zamanda araçlarının tam pozisyonlarının teknolojik ve / veya ekonomik yetersizliklerden dolayı bilememektedir.

2.) Problem açık uçlu (sınırlandırılmamış) olabilir.

Statik problemlerde proses genellikle geçici olarak sınırlandırılmıştır. Bu tip problemlerde rotalar terminalde başlar ve terminalde sona erer. Dinamik bir ortamda ise, problem açık uçlu olabilir ve bu tip problemlerde sınırlardan söz etmek pek de mümkün olmayabilir. Planlayıcı tarafından çizilmiş rotalar yerine araçların izleyeceği yollardan bahsedilebilir.

3.) Geleceğe ilişkin bilgi kesin olmayabilir ya da bilinemeyebilir.

Statik ve dinamik rotalamalarda, değerlendirmeye tabi tutulacak verinin kalitesi de farklılık gösterir. Girdiler, kesinlikle bilinen (deterministik), tahminler ile kesin olmaksızın bilinen veya istatistiki dağılımları izliyor (probabilistik) olabilir. Statik bir rotalama probleminde, problem ile ilgili bütün bilgi, anlık bilgi ile aynı kalitede biliniyor (deterministik) kabul edilir.

İlgili araçların rotası, programlayıcı tarafından bu bilgiler ışığında oluşturulmaya çalışılır.

Gerçek bir dinamik rotalama probleminde ise; geleceğin kesin bir şekilde bilinmesi hemen hemen imkansızdır. En iyi ihtimal ile gelecek ile ilgili olasılıksal veri biliniyor olabilir. Şüphesiz gelecek ile ilgili olasılıksal veri bile sağlıklı bir analiz yapabilmek için çok değerli olacaktır.

4.) Kısa vadeli olaylar çok daha önemlidir.

Statik araç rotalamada, veri kalitesinin hep belli seviyede olması ve belli bir çizelgeleme periyodu için verilerde değişikliğin söz konusu olmamasından dolayı tüm olaylar aynı önemde ele alınır. Dinamik bir ortamda ise araç kaynaklarımızı uzun vadeli ihtiyaçlar için atamak ve bu nedenle harcamak mantıklı bir hareket olmayacaktır. Çünkü uzun vadeli ihtiyaçlarda, dinamik rotalamanın doğası gereği değişiklikler gözlenmesi muhtemeldir. Bu nedenle dinamik bir ortamda planlayıcının odaklanması gereken noktanın kısa vadeli olaylar olacağı açıktır.

5.) Veri ve bilgilerin güncellenmesini sağlayan mekanizmalar gereklidir.

Dinamik rotalama problemi ile alakalı neredeyse bütün veriler gün içerisinde operasyonların akışına göre değişime mahkumdur. Bu nedenle veri güncelleme mekanizmalarının çözüm metoduna entegre edilmesi ve güçlü bir veri tabanı yönetim sistemi gereklidir. İşin doğal akışının bir sonucu olarak, bahsi geçen güncelleme mekanizmaları statik rotalama literatürü ile ilgili değildir.

6.) Yeniden sıralama ve yeniden atama kararları yetkilendirilebilmelidir.

Dinamik rotalamada, yeni verinin sisteme girişi, planlayıcı tarafından alınmış kararları optimal olmaktan çıkarabilir. Bu da, sisteme yeni talepler ulaştığında, bu yeni duruma yanıt verebilmek adına planlayıcının araçları yeniden rotalamaya veya en azından tekrar atamaya zorlar. Bu nedenle dinamik rota programlayıcısı yeniden atama ve rotalama yetkileri ile donatılmalıdır.

7.) Hızlı hesaplama süreleri bir zorunluluktur.

Statik bir ortamda, planlayıcının yüksek kalitede bir çözüm elde etmesi hatta optimal bir çözüme ulaşabilmesi için bir kaç saat bekleme lüksüne sahiptir. Örneğin altı aylık bir otobüs planlaması sırasında araç atama problemini çözerken 5-6 saatte sonuçlanan çözümler dahi kabul görebilmektedir. Oysa ki, dinamik bir ortamda bu kadar uzun hesaplama süreleri kesinlikle kabul edilemez. Planlayıcı, eldeki problemin çözümüne bir an önce ulaşmak ister. Bir an önceden kastımız, dakikalar hatta tercihen saniyelerdir. Bahsi geçen programın

‘çalışma süresi’ ile ilgili kısıtları, yeniden rotalamanın ve buna bağlı olarak yeniden atamaların genellikle ekleme ve k-değişim (k-interchange) gibi yerel geliştirme sezgisel yöntemleri ile yapılmasını tercih edilir.

8.) Sınırsız erteleme mekanizmaları gereklidir.

Bazı taleplerin diğer taleplere göre bağıl olarak tercih edilmeyen karakteristiklerinden (örneğin coğrafi konum) söz etmek mümkündür. Bahsi geçen bu taleplere hizmet götürmenin süresiz olarak ileri bir tarihe tecil ettirilmesine “sınırsız erteleme” adı verilir. Bu mekanizma müşterilerin çok uzun süre beklemesi sonucunu doğurduğu için şirketin prestij, müşteri ve para kaybetmesine neden olabilir. Tercih edilmeyen bu durum, zaman pencereleri kısıtları veya aşırı beklemeceze ceza yöntemi getiren liner olamayan bir amaç fonksiyonu ile hafifletilmeye çalışılabilir.

9.) Amaç fonksiyonu farklı olabilir.

Daha önce değinildiği üzere proses sınırlandırılmamış (açıkuçlu) olabilir. Bu nedenle, toplam mesafenin veya toplam çizelgeleme zamanının minimize edilmesi gibi geleneksel statik amaçlar dinamik bir ortam için anlamsız olabilir. Gelecek ile ilgili girdiler elde değilse, sadece bilinen verileri kullanarak optimizasyon yapmak mantıklı olacaktır. Bazı sistemlerde yukarıda bahsi geçen süresiz erteleme gibi istenmeyen durumlardan elden geldiğince sakınmak için liner olmayan amaç fonksiyonları da kullanılır.

10.) Zaman kısıtları farklı olabilir.

Daha önce tanımladığımız kurye prosesine yeniden dönersek: İşin alınması veya teslim edilmesi gereken en geç zaman gibi zaman kısıtları dinamik bir ortamda statik bir ortama göre daha esnektir. Zira dinamik ortamda yeni ve ani talepler söz konusudur. Bahsi geçen taleplerin reddedilmesi, zaman kısıtları tatmin edilmediğinde, (her yeni talebin sisteme ilave edilmesi ile eldeki kısıtların çığneneceği ve çözümün optimal olmaktan çıkacağı anlamına gelmez.) zaman kısıtlarını çığnemekten çok daha az tercih edilen bir durum olacaktır.

11.) Toplam kapasitede değişiklik esnekliği daha azdır.

Statik bir ortamda algoritmanın çalışması ve çözüm üretmesi ile bu çözümün rotalar haline getirilerek yürürlüğe sokulması arasındaki zaman toplam kapasitede (filo büyüklüğünde ve araç sayısında) değişimlere gidilmeyi mümkün kılar. Buna karşılık dinamik bir ortamda, planlamacının filo büyüklüğü üzerinde değişikliğe gitme, bazı araçları yedekleme yetkisine sahip değildir. Bu uygulamanın neticesinde bazı müşterilerin daha düşük kalitede hizmet alabileceği sonucuna ulaşılabilir.

12.) Kuyruk Teorisi ile ilgili etmenler önemli olabilir.

Eğer müşteri taleplerinin sıklığı belirli bir eşiği aşarsa, sistem tıkalı bir hale gelir ve anlamsız sonuçlar üretmeye başlayabilir. Yöneylem araştırmasında, hem araç programlama hem de kuyruk teorisi bağımsız olarak çok fazla çalışılmış ve literatürde fazlasıyla yer bulmuş olmasına karşın bu iki disiplinin bütünleştirilmesi ne yazık ki yetersiz kalmıştır.

Çalışmamızda araç programlama ve araç atama problemini dinamik bir ortamda çözmeye çalışırken, kuyruk teorisinin açtığı kapılardan ve ulaşılmış olduğu sonuçlardan da azami ölçüde yararlanmaya çalıştık. Bu amaç doğrultusunda ortaya koyacağımız sonuçlar, nicel olarak irdelenebilecek pek çok problem tipine rahatlıkla uygulanabilir. Ek olarak, bu sonuçlar ile rassal ve dinamik araç rotalama sistemlerinin davranışlarını irdeleme şansı aranmış ve uygulanabilir ve tercih edilebilir bir yaklaşım üretmeye odaklanılmıştır.

Psaraftis'ten sonra Powell, vd. (1995)'nin dinamik problem üzerine yaptığı yorumları kısaca irdelemek yararlı olacaktır. Powell vd. (1995)'te problem, model ve modelin uygulanması konularının ortamın dinamikliği ile ilişkilendirilmesi başarılmıştır. Bu doğrultuda:

- Problemin, bir ya da daha fazla parametresi zamanın bir fonksiyonu ise problem dinamiktir denilebilir. Bu tanıma göre, dinamik datanın sabit olarak zaman içinde değiştiği ve zamana bağlı değişkenin peşinen bilindiği problemler de dinamik problem olarak adlandırılabilir.
- Eğer model, zaman içerisinde aktivitelerin eklenmesi veya çıkarılmasını da içerecek şekilde bütünleşmiş ise modelimizin dinamik olarak tanımlanması mümkündür.

Bahsi geçen model uygulamaya geçildiğinde ise, her yeni bilginin sisteme ulaştığında yeniden çözülüyor ise uygulamamız dinamiktir. Modelin uygulaması konusunda Psaraftis (1988) ile neredeyse tamamen örtüşen Powell vd. (1995)'e göre dinamik uygulamaların en büyük dezavantajının çok ciddi hesaplama kaynağına gereksinim duyması olduğu belirtilmiştir. Analizimizin pratik uygulamalarını daha mümkün kılan dinamik yaklaşım, gereksinim duyduğu ek kaynaklara rağmen çalışmamızda tercih edilmiştir. Bu tercih ile ilgili detayları, modelimizin dinamik yönü ve dinamiklik derecesini takip eden kısımda ayrıntıları ile irdelenmiştir.

2.4 Dinamik Araç Rotalama Probleminin Günlük Hayatta Uygulanabileceği Alanlar

Yazılı ve görsel basında, ambulans ya da itfaiyenin olay mahalline geç gelmesi sonucu oluşan maddi ve manevi kayıplarımız duyarlı her Türk Vatandaşının yüreğini burkmaktadır. Bu iki örnek çok yaygın ve önemli olsa da kentsel hizmetler konusunda şikayet ve

memnuniyetsizlikler bu iki örnek ile kısıtlı kalmaz. Kurye ve işyeri servislerinin geç gelmesi, beklenen tamircinin evimize geç gelmesi, caddelerimizden karın oldukça yavaş kürünebilmesi, yolda kaldığımızda bizi soğuk ve yağışta uzun süre bekleten çekici veya kasko tutanağı için beklenen ve genellikle de trafiğin keşme keşliğini daha arttıran polis otomobilleri hayatımızda oldukça önemli yer tutmaktadır. Çalışmamızın bu bölümünde, günlük hayatımıza uygulanabilecek ve geliştirdiğimiz yaklaşımdan yararlanabilecek, dolayısı ile yaşam kalitemize olumlu etki edebilecek örneklerle kısaca değinilmiştir.

2.4.1 Ambulans, İtfaiye ve Polis Hizmetleri

Acil durumlarımızda imdadımıza yetişen acil servis (ambulans, itfaiye ve polis) hizmetlerine dinamik rotalama problemi öğretileri, hizmet için talebin gerçek zamanlı olarak ve coğrafi olarak irdelenen kuyruk sistemi ağı üzerinde oluşturduğu için rahatlıkla uygulanabilir. Pek çok zaman, talepler yeni bir talebin sisteme girişinden önce karşılandığı için (hafif trafik durumu), rotaların oluşturulmasına gerek kalmaz. Bu sebeple problemimiz, yeni talep için atanabilecek en iyi aracın (pek çok zaman en yakın araç) talebe atanması haline gelmiştir. Acil servis araçlarını gönderimi için kullanılan metodlar genellikle araç ve ekiplerin nerede koğuşlandırılacağını da içeren yer seçim analizlerini de içerir. Bahsi geçen acil servis yerleşkelerinin seçimi ile ilgili detaylı bir çalışma Landolfi (1997)'de irdelenmiştir.

2.4.2 Altyapı Onarımı Problemi

Belediye hizmeti veren bir şirket (elektrik, doğalgaz, su-kanalizasyon, yol vs) geniş ve coğrafi olarak dağılmış bir ağı kapsamak ve ilgili ağın bakım ve onarımını yapmakla yükümlüdür. Bahsi geçen ağ üzerinde, hem zaman ekseninde hem de coğrafi nokta olarak rasgele yetersizlikler ve / veya problemler oluşur. Şirket, yetersizliklere cevap vermek üzere terminalden ayrılacak ve olay yerine intikal edecek bir tamir filosu işletir. Rotalama kararları, o anki yetersizliklere ve bazen de gelecek yetersizlik prosesinin karakteristiklerine göre verilir. Araçta görev yapan ekip, her yetersizliği düzeltmek için rasgele bir servis zamanı harcar ve bir sonraki yetersizliği çözmek üzere harekete geçene kadar serbest kalır. Genellikle şirket, yetersizliklerden dolayı, ağın servis dışı kalmış kısmının, servis dışı kalma zamanını minimize edecek şekilde ellerindeki tamir filosunu yönetmeye çalışır.

Bu tip problemlerde, filonun bir parçası olan teknisyenin izleyeceği yol, kat edilen mesafeyi ya da zamanı minimize edecek bir amaç fonksiyonuna ya da talebin aciliyetine bağlı olarak seçilebilir. Bahsi geçen problem tipi literatüre Bertsimas ve Van Ryzin (1991) tarafından kazandırılmış olup 'Dinamik Gezgın Tamirci Problemi (DGTP) olarak geçer. Çalışmamızda

ele aldığımız model, bu problem tipi üzerine bina edilmeye çalışılmıştır. Bu bağlamda DGTP çalışmamızın bir sonraki bölümünde detaylı olarak irdelenmiştir.

2.4.3 Kurye - Posta Hizmetleri

Kurye-Posta hizmetlerini veren şirketler, bir noktadan posta ve / veya paketleri alarak, diğer bir noktaya bahsi geçen malları güvenli bir şekilde, dünya çapında ve belirli bir zaman dilimi içinde teslim etmeyi teklif eder. Posta / paketlerin sevkinde sadece yerel değil, şehirler hatta ülkeler arası ulaşım ihtiyacı duyulabilir. Teslimatlar önce bir toplama merkezine götürülür ve buradan teslim edileceği noktaya doğru yüklenir. Tanım 2.1 ve 2.2'yi referans almaya devam edersek, teslimatı bekleyen alıcılar yüklemeyen önce şoför ve planlayıcı tarafından bilindiği için teslimatlar statik rotalama problemi oluşturur. Bunun yanında, teslimat sırasında ortaya çıkan alma ve teslim etme talepleri, problemin doğasını değiştirir ve statik problemi dinamik hale getirir. Çünkü bu ani talebin nereye ve ne zaman teslim edileceği ile ilgili gerekli bütün bilgiye planlayıcı ve sürücü sahip değildir.

2.4.4 Merkezi Bir Terminalden Mal Teslimatı

Merkezi bir terminalden ellerindeki malları, müşterilerin gerçek zamanlı talepleri doğrultusunda teslim eden bir şirket ele alalım. Siparişler bir dosyaya kaydedilir ve teslim araçları ortalama teslim zamanı ile teslim maliyetinin bir kombinasyonunu içeren bir amaç fonksiyonu dahilinde atanır. Bu ve benzeri hizmeti veren şirketler kalabalık bir müşteri popülasyonuna (potansiyel müşterilere) servis vermesi beklenebilir. Bahsi geçen hizmete verilebilecek en güzel örnek ev ve iş yerlerine akaryakıt (fuel-oil) veya kömür sevkiyatıdır. Şimdi bu prosesi bir model haline getirip yakından irdelemeye çalışalım.

Ev ve şirketlere yakıt sevkiyatı sıcaklık-gün ilişkisi ile yürütülebilir. Bu bağlamda sevkiyatları müşterinin yakıt stoklama kapasitesine ve dış ortam sıcaklığına yürütebilecek bir mekanizma kurmak mümkündür. Yakıt istasyonları, sıcaklık-gün ölçüsünü hangi müşterinin ısınmak için ne kadar yakıt kullandığını hesaplayabilmek için kayıt altında tutar. Veritabanı rotalama sistemine, müşterinin az miktarda yakıtı kaldığı uyarısını verdiğinde, bu müşteri servis bekleyen müşteriler listesine eklenebilir. Müşterilerin yakacak stoğu ile günlük hava durumuna bağlı olarak harcamalarını bütünleştirebilen bir yaklaşım ile işletilen bir mekanizma ile sistemi dinamik olarak yürütmek dahi mümkün olacaktır. Zira geleceğe ilişkin talep zamanının (mevsimsel günlük sıcaklığın) bir fonksiyonudur. Problemimizin karmaşıklığı, müşterinin stoğundaki yakıtın veritabanında takip edilenden daha önce bitmesi ile artacaktır. Bu durumun oluşmasında en büyük etmen, hava sıcaklığında görülen

öngörülemez bir düşüş ya da hanenin yakıt harcama politikasındaki değişme olabilir (eve gelen yatılı misafirler sebebi ile yeterince ısıtılmayan kısımların yeniden ısıtılması gibi). Yukarıda bahsettiğimiz detaylar ile ortaya attığımız modelimize çözüm getirmek, geleneksel statik ve deterministik rotalama probleminin çözümüne kıyasla çok daha zordur. Çünkü kullanılan sıcaklık-gün ölçüsü planlayıcıya gerçek yakıt kullanımı kesin olarak veremez, bu sadece bir tahmindir ve eldeki bu tahmin problemi talebe göre rassal hale getirir. Bununla birlikte müşterinin depo kapasitesi ve depolama stratejisi de modelimize dahil edilebilir, hatta müşteri deposunda ne kadar yakıt bulunduğu gerçek zamanlı olarak kontrol altında tutulduğu bağlantılı yaklaşımlar dahi geliştirilebilir.

2.4.5 Al ve Teslim Et Problemleri

Dinamik araç rotalama ve çizelgeleme probleminin bir uzantısı da 'Al ve Teslim Et' problemleridir. En basit hali ile bu problem, bir araç filosu ve müşteri talepleri kümesinden oluşur. Her müşteri talebi, en az bir başlangıç noktası (alındığı orijin) ve teslim noktası (hedef) ile belirlenir. Araçlar hedeflenen noktalar ulaşmadan önce ilgili hedeflerin orijinlerine uğramak durumundadır. Bu basit yapı her türlü karmaşık 'Al ve Teslim Et' problemine uygulanabilir. Bu problem tipi, Gezgin Satıcı Probleminin (GSP) sınırlandırılmış bir versiyonudur. Bahsi geçen ek kısıt orijinlerin hedeflerden önce gelmesidir. Bu problemin özel bir uygulaması da çocuk, yaşlı ya da özürlü vatandaşların taşınmasıdır. Larsen (2001)'e göre, Kopenhag'da (Danimarka), yerel otobüs işletmeleri yaşlı ve özürlüler için benzer bir uygulamayı gerçekleştirmektedirler. Şu anki mekanizmaya göre, servis talep eden vatandaş, ilgili işletmeyi ihtiyacı olan hizmetin gerçekleşeceği günden bir gün önceden arar ve gerekli bilgileri (orijin- hedef) verir. Şu anki mekanizma problemimizi statik hale getirir de, bahsi geçen işletme internet üzerinden online servis vermeye hazırlanmaktadır. Şüphesiz ki bu uygulamaya geçişi ile problem dinamik bir hal alacaktır.

2.4.6 Ticari Taksi Hizmetleri

İstanbul Kentinde, İstanbul Büyükşehir Belediyesi verilerine göre, Ağustos 2000 itibari ile 16.985 taksi ve 572 adet dolmuş taksi olmak üzere toplam 17.557 adet ticari taksi mevcuttur [17]. Ticari taksi hizmetini veren bu kadar büyük bir araç filosunun bulunması bile taksi hizmetlerinin yönetimini dinamik rotalama ve çizelgeleme probleminin konusu olabilecek potansiyel uygulamalarından biri saymak için yeterlidir. Ticari taksi sistemlerinin pek çoğunda, dinamik müşterilerin oranı çok yüksektir, zira taksici taksi durağını terk etmeden önce, pek az müşteri planlayıcı tarafından belirlenebilir. Ticari taksi rotalama probleminde,

taksi boş olabilir ya da bir veya daha fazla müşteriye hizmet veriyor olabilir. Eğer taksi boş ise, taksi sürücüsü genellikle aracını daha rahat müşteri bulabileceği bir konuma doğru hareket ettirir. Yeniden pozisyonlanmanın görüldüğü bu noktada, taksicin son hizmet verdiği müşterisini bıraktığı noktanın yeni müşteri bulma açısından karakteristiği de önemlidir. Konum seçimi çeşitli ampirik veri yardımıyla ya da sürücünün hisleri ve tecrübesi doğrultusunda gerçekleşir. İstanbul Şehiriçi vapur iskelelerinde, vapurların programlı olarak iskele verdiği saatlerde ve mesai saatleri sonrası büyük alışveriş merkezlerinde gözlemlenen ticari taksi yığılmaları bahsi geçen konum seçiminin doğal bir sonucudur. Taksinin müşteriye atanması, ülkeden ülkeye hatta duraktan durağa farklılık gösterir. Türkiye’de uygulanan pratik ise, bir ya da daha çok ticari taksiye sahip girişimcileri bir araya gelir. Bu girişimciler bir çağrı ünitesini (ortak telefon ve / veya telsiz) ve bir ayrılma noktasını (taksi durağını) paylaşır. Müşteriler ticari taksilere daha önce kararlaştırılmış atama politikasına göre (genellikle ilk gelen ilk müşteriyi alır) atanır. Ek olarak müşterilerin taksiye atanmasında, her girişimci tarafından sahip olunan ticari taksi sayısı da göz önüne alınır. Bunun bir sonucu olarak, ticari taksi rotalama sistemi, müşterilerin taksilere atanmasında bir yük dengeleme stratejisi oluşturmalıdır. Bu politika ile, girişimciler taksi durağından en iyi verimi alacaklarından emin olabilirler.

3. AKADEMİK ve TEKNİK ALTYAPI

Çalışmamızın bu bölümünde, dinamik araç rotalama ve çizelgeleme problemini kapsayan akademik literatür taraması ile probleme ait yaklaşımımızı mümkün kılan teknik ve teknolojik altyapıya yer verilmiştir. 1980'lerden günümüze dek rassal ve / veya dinamik araç çizelgeleme problemini ele alan yayımlanmış çalışmaların sayısında çok ciddi bir artış olmuştur. Bahsi geçen çalışmalar hem uygulama alanı hem de yöntemsel yaklaşımlar olarak geniş bir alana hitap etmektedir. Özellikle problem içindeki değişkenlerin ve kısıtların fazlalığı ile kapsadıkları bütçelerin büyüklüğü göz önüne alındığında toplu taşıma şirketlerinin araç ve ekip programlamalarına yönelik çalışmaları dikkat çekicidir. Şüphesiz ki, bu kısımda araç programlama üzerine yazılmış tüm çalışmaları irdelemek mümkün değildir. Bu kısımda amacımız çalışmamızın kökenini oluşturan konu başlıkları hakkında özet bilgi vermektir. Literatür taramamız, 1 Mart 2003 günü sona ermiştir. Bu nedenle o tarihten sonra yayımlanmış ve literatüre kazandırılmış çalışmalar, incelemimize dahil edilmemiştir.

Bu kısımda gerçek zamanlı optimizasyon metotlarını özetlendikten sonra, dinamik gezgin tamirci problemini ortaya atan Bertsimas ve Van Ryzin (1991,1993)'in çalışmaları gözden geçirilmiştir. Bu çalışmalarda, dinamik araç programlama ile kuyruk teorisini bütünleştiren kapsamlı bir matematiksel çalışma yapılmış olup, bahsi geçen çalışmada yer alan politika ve kısıtlar incelememizde geliştirilmeye çalışılmıştır. Akabinde Lund vd. (1996) tarafından ortaya atılan 'dinamiklik derecesi' kavramı ele alınarak, 'kesin olmayan' talepler ile araç atama problemini çözmenin pratik zorlukları değerlendirilmiştir. Son olarak da, çalışmamızı mümkün kılacak teknik gereksinimlerimiz olan; pozisyonlama ve iletişim ekipmanları ile ilgili teknolojiler gözden geçirilmiştir.

3.1 Gerçek Zamanlı Optimizasyon Metotları

Çalışmamızın bir önceki bölümümüzde, statik ve dinamik araç atama ve çizelgeleme problemleri ile rotaların oluşturulmalarında izlenen farklı yöntemler tanımlanmıştı. Bu kısımda, günlük operasyon esnasında, gerçek zamanlı olarak ve araçlar bir rota üzerinde yol alırken, rotalama çalışmasının gerçekleştirilmesi irdelenmiştir. İlk olarak dinamik gezgin satıcı problemine (DGSP) değinilmiş olup, daha sonra da dinamik gezgin tamirci problemi (DGTP) ele alınmıştır.

3.1.1 Dinamik Gezgin Satıcı Problemi (DGSP)

Daha önce de belirttiğimiz gibi, GSP yöneylem araştırmasında en fazla çalışılmış

problemlerden biridir. Bu çalışmaların çok ciddi bir kısmı, GSP'nin deterministik ve statik uygulamalarına yoğunlaşmıştır. Gezgin satıcı probleminin dinamik bir versiyonu (DGSP) ilk defa Psaraftis tarafından 1988 yılında literatüre kazandırılmıştır. Psaraftis (1988), DGSP'yi şu şekilde tanımlar: “ G , n adet noktadan oluşan ağımız olsun. Hizmet için talepler her nokta için birbirinden bağımsız olarak oluşturulur. Genellikle veri yaratma prosesinde, λ parametresi ile Poisson Dağılımı tercih edilir. Oluşturulan talepler, nokta i 'den nokta j 'ye sabit bir hız ile yolculuk eden bir satıcı tarafından karşılanmaya çalışılır. Yolculuk sırasında geçen zaman t_{ij} ile gösterilirken, her bir noktada talebin gereksinim duyduğu servis zamanı t_0 ile gösterilir. Bu bilgiler ışığında dinamik problem için optimal bir rotalama politikası ve dolayısı ile optimal bir çözüm aranmaktadır.”

Şüphesiz ki, dinamik gezgin satıcı probleminin pek çok değişik versiyonu bulunabilir. İlgilenilen ağ, tamamlanmamış, öklitsel ya da simetrik olabilir. Ek olarak, talep yaratma prosesinde Poisson'dan farklı dağılımlara başvurulabilir, her nokta için talep oranları farklı olabilir ve taleplerin karşılanmasında birden fazla satıcı kullanılabilir. Bununla birlikte Psaraftis (1988)'de bazı konuları ilginç ve araştırmaya değer olarak belirtmiştir ve ileride yapılacak çalışmalar için irdelenebilecek konular olarak göstermiştir. Bu konular, performans ölçütleri, hafif ve yoğun trafik durumu ile yeniden konumlandırma. Aşağıda, bahsi geçen konular irdelenmeye çalışılmıştır.

3.1.1.1 Performans Ölçütleri

Gregory ve Platts (1995)'a göre; performans ölçüsü, verimlilik ve / veya etkinliği ölçmek için bir metrik olarak tanımlanabilir. Bu tanımlı araç çizelgelemeye uygularsak, atamanın etkinliği, istenilen iletim ve servis hizmetlerinin ne derecede başarabildiğini, verimliliği ise (kaynakların değerlendirilmesi cinsinden) bu görevlerin ne derece ekonomik olarak yerine getirildiğini tasvir eder. Bu nedenle, etkin bir sistemin verimsiz ya da verimli bir sistemin etkinlikten uzak olması mümkündür.

Performans ölçütleri, kalite, zaman, maliyet, kaynak değerlendirilmesi, işletme etkinliği ve esneklik ölçütleri esasına göre kategorize edilebilir. Buna rağmen, taşımacılık ağlarında kullanılan performans ölçütlerinin büyük çoğunluğu, çıktı veya gecikme ölçütleri üzerinedir. Araç atama ve çizelgeleme çerçevesinde bu ölçütlerin anlamı optimal politikanın anlamını da belirler. Bu bağlamda optimal politika birim zamanda yanıt verilmesi beklenen talep sayısını maksimize etmek ya da aynı talepler için ortalama bekleme zamanını minimize etmek olabilir.

Doğru ve ilgili performans ölçütünü seçmek, herhangi bir sistemi hatasız olarak analiz etmek

adına kritiktir. Globerson (1985), performans ölçülerini seçme adına bazı öneriler getirmiştir.

Bu öneriler:

- Performans kriterinin hesaplama metodu net olarak tanımlanmalıdır.
- Nesnel bir performans kriteri, öznel olana tercih edilmelidir.
- Oransal performans kriteri, mutlak sayılara tercih edilmelidir, zira oranlar, iki ya da daha fazla faktörün kıyaslanmasına imkanı verir.

Çalışmamızın ileriki kısımlarında modelimizi kurulması, performans ölçüsü ile amaç fonksiyonlarının belirlenmesi ve sistemi yerinde analiz edilebilmesi için Globerson (1985)'in önerileri doğrultusunda performans ölçüleri seçilmiştir.

3.1.1.2 Hafif Trafik Durumu

Talep oranının (λ) bağıl olarak düşük olduğu, kuyruk teorisindeki gösterime atıfta bulunarak ($p = \lambda / \mu$) $p \rightarrow 0$ olduğu durumlarda, izlenen politika ne olursa olsun araç ya da araçlar talebi karşılamak için yeterli olacaktır. Aynı zamanda yine politikadan bağımsız olarak çıktı da, $n \times \lambda$ olacaktır. Buna rağmen bekleme zamanları için benzer bir yorum yapmamız mümkün değildir; zira bekleme zamanları tercih edilen politikaya bağlı olacaktır. Hafif trafiğin gözlemlendiği durumlarda 'Ortaya çıkan talebi en kısa zamanda yanıtla' politikası 'Daha önce belirlenmiş noktalar kümesindeki talebi, belirlenmiş sıra ile cevapla' politikasına kıyasla daha az bekleme süresi doğuracaktır. Bu tarz trafiğin gözlemlendiği pratik uygulamalar acil servis, itfaiye, polis hizmetleri gibi kapasitenin yetersiz tutulduğu yani talebe yanıt verilemediği durumların maliyetinin çok yüksek olabileceği vakaları içermektedir.

3.1.1.3 Yoğun Trafik Durumu

Günlük hayatta hafif durumlar kadar yoğun trafik durumlarına da rastlanılır. Kapasite tutmanın çok masraflı olduğu ya da talebi reddetmenin veya bekleme zamanının çok yüksek gerçekleşmesinin bağıl maliyetinin az olduğu durumlar yoğun trafik durumuna örnek olarak gösterilebilir. Eğer talep oranlarımız daha yüksek değerler almaya başlarsa [$(p \rightarrow 1)$ olduğu durumda], problemimiz daha karmaşık bir şekle bürünecektir. Bahsi geçen durumlarda, araç ya da araçlar talebi karşılamakta yetersiz kalabilir. Hafif trafik durumunda, hangi politika seçilirse seçilsin, talebi karşılamada hiç bir sorun yaşanmazken, yoğun trafik durumunda talebin karşılaşıp karşılanamaması büyük ölçüde seçilen politikaya bağlı olacaktır.

3.1.1.4 Yeniden Konumlandırma

Yeniden konumlandırmayı doğru bir şekilde analiz etmek istiyorsak, bu yeni problemimizin dinamik araç atama problemi ile terminal konumlandırmanın kombinasyonu olarak irdelenmesi doğru bir yaklaşım olabilir. Bu yaklaşım kendisi ile birlikte yeni bir politikayı da gündeme getirir, araçlar herhangi bir talebi karşıladıktan sonra, planlayıcı tarafından daha önce belirlenen bir noktada yeniden konumlandırılarak yeni talepleri beklemeye başlarlar. Bahsi geçen belirlenmiş noktanın, stratejik açıdan beklenen taleplerin medyanı olması öne sürülebilir. Yeniden konumlandırma yaklaşımı için Kısım 2.4.6'da irdediğimiz taksi hizmetleri örneği de tekrar gözden geçirilebilir. Bu yaklaşım, talep oranlarının zayıf olduğu, hafif trafik durumlarında planlayıcı açısından değerli sonuçlar ortaya koyabilir.

Yukarıda kısaca özetlediğimiz geçen dört konu başlığından da, yaklaşımımızın performansını değerlendirme amacıyla uygulama kısmında yararlanılmıştır.

3.1.2 Dinamik Gezgin Tamirci Problemi (DGTP)

Bu kısımda araştırmamızın matematiksel ve fikirsel temellerini atan dinamik gezgin tamirci problemi (DGTP) incelenmeye çalışılmıştır. Dinamik gezgin tamirci problemi, Bertsimas ve Van Ryzin (1991) tarafından ortaya atılmıştır. Bu çalışmada, araç rotalama üzerine kurulan matematiksel alt ve üst sınırlar özellikle dikkat çekicidir. Matematiksel denklemler rassal ve kuyruk teorisinden gelen alt yapı ile desteklenmiştir. Bertsimas ve Van Ryzin (1991)'e göre DGTP şu şekilde tanımlanabilir:

- Bütün talepler dinamiktir ve λ parametresi ile gösterilen poisson proses ile zaman içerisinde oluşturulur.
- Taleplerin olduğu noktalar, bir A alanı içinde, bağımsız ve düzgün olarak dağılmaktadır.
- Taleplere yanıt, belirlenmiş A alanının dışbükey bir kısmında hizmet veren bir tamirci tarafından verilir.
- Taleplere K adet tamirciden oluşan bir küme ile yanıt verilmeye çalışılır. Tamirci bahsedilen sınırlı alanda bir sabit bir hız ile yol alır, şüphesiz ki bahsi geçen sabit hız yolculuk zamanını etkileyecektir.
- Her talep, birbirinden bağımsız ve eş olarak dağıtılmış bir olay yerinde servis zamanı gerektirir. Talep i için servis zamanı, s_i ile gösterilebilir. Bu servis zamanının aritmetik

ortalaması $\bar{s}$ ile ve ikinci momenti de s^2 ile gösterilir. Bununla beraber, tamircinin olay mahallinde geçirdiği zamanın toplam zamana oranı p ile gösterilecektir ve sabit duruma erişebilmiş bir sistem için $p = \lambda \cdot s$ eşitliği geçerlidir.

- Talep i için bekleme zamanı, W_i ile gösterilebilir ve talebin oluşmasından, servisin başlamasına kadar geçen zamana işaret eder. Aynı talep için sistem zamanı (T_i) ise, talebin olduğu andan, tamircinin servisi tamamlayarak olay yerinden ayrılmasına dek geçen toplam zamana işaret eder.

Yukarıda temel karakteristiklerini tanımlamaya çalıştığımız dinamik problemin amaç fonksiyonu taleplerin sistemde harcadığı toplam zamanı minimize etmektir. Bununla birlikte bekleme zamanını (W_i) veya toplam sistem zamanı ile bekleme zamanının belirli ağırlıklar ile kombinasyonunu minimize edilen amaç fonksiyonları da tercih edilebilir. Her bir talebin sistemde harcadığı toplam zamanın (bekleme zamanı ile servis zamanının toplamı) formülize edilmiş şeklini eşitlik (3.1)'de irdelenebilir.

$$T_i = W_i + s_i \quad (3.1)$$

Sistem denge durumuna eriştiğinde şu eşitliklere ulaşılabilir:

$$T_i = \lim_{i \rightarrow \infty} E[T_i] \quad (3.2)$$

ve

$$W = T - \bar{s} \quad (3.3)$$

Yukarıda kısaca açıklanan teorik özelliklerin pek çok pratik çerçeveye adapte edilmesi mümkündür. Taleplerin gerçek zamanlı alındığı ve teslimatın bu talepler baz alınarak yapıldığı tüm dağıtım sistemleri (kurye servisleri, pizza ve çiçek dağıtımı gibi) bu uyarlama için birer adaydır. Dinamik gezgin tamirci problemi başlığı altında incelenebilecek diğer uygulamalara, elektrik ağının onarımı, bilgisayar ve telekomünikasyon ekipmanlarının bakımı ve onarımı, doğalgaz ve su tesisatlarının onarımı ile yol çalışmaları örnek olarak gösterilebilir.

Çalışmamızda ele aldığımız araç atama probleminde, klasik amaç fonksiyonumuz olan katedilen mesafenin minimize edilmesinden çok daha önemli bir amaç fonksiyonu; sistem zamanının ve bekleme zamanının minimizasyonunu ele alınmıştır. Bununla birlikte araç başına kat edilen toplam mesafe de tartışmalarımızda göz önüne alınacaktır. Bu tercihi

yapmamızın başlıca nedenleri şunlardır:

- Şirket politikası teslimat maliyetinden çok servis kalitesi ve müşteri tatmini üzerine kurulmuş olabilir.
- Teslimatları hızlandırmak adına taleplere olduğu anda en hızlı cevabı verebilmek için tam zamanında yanıt politikası uygulanıyor olabilir.
- Operasyonel maliyetlerin çok büyük bir kısmı (işçi, amortisman, kira) sabittir, bu nedenle mesafenin minimize edilmesinin maliyetleri azaltmak üzerinde sınırlı etkisi olacaktır.

Gezgin tamirci problemi, T 'yi minimize edecek bir rotalama politikası bulmak üzerine tasarlanmış ve optimal T değeri T^* ile gösterilmiştir. Sistem zamanının bileşenlerinden biri yukarıda bahsettiğimiz üzere yolculuk zamanıdır. Yolculuk zamanı, hem servis için bekleme süresini hem de toplam servis süresini etkileyecektir. Problemimiz klasik bir kuyruk problemini andırsa dahi yolculuk zamanları bağımsız değişken olarak ele alınamayacağı için kuyruk teorisi doğrudan problemimize uygulanamayacaktır. Bu nedenle Bertsimas ve Van Ryzin (1991)'in önerdiği yaklaşım bütün politikalar için, ortalama sistem zamanı (T) için alt sınırlar elde etmeye çalışmak olmuştur. Bu sonuçları elde etmek için kombinasyonel optimizasyon, kuyruk teorisi, geometrik olasılık ve simülasyondan yararlanılmıştır. Takip eden bölümde DGTP iki temel kısımda irdelenmiştir. Bu kısımlardan ilki, kapasite sınırı olmayan tek bir araç ile taleplere yanıt verilmeye çalışılan sistem olurken, diğeri birden fazla araç için kapasite kısıtlarının olduğu sistemler olacaktır.

3.1.2.1 Tek ve Kapasite Sınırı Olmayan Araç ile Dinamik Gezgin Tamirci Problemi

Tek ve kapasite sınırı olmayan araç için dinamik gezgin tamirci problemi için yapılan çalışma ve analizleri ana hatları ile iki kısımda inceleyebiliriz. Bunlardan birincisi, kısım 3.1.1.2'de de değindiğimiz hafif trafik (örneğin, $\lambda \rightarrow 0$) durumu diğeri de, kısım 3.1.1.3'te değindiğimiz yoğun trafik (örneğin, $p \rightarrow 1$) durumudur.

Hafif trafik durumunda, T için alt sınır, sistem zamanını üç bileşene ayırarak oluşturulmuştur. Bu bileşenler, i talebinden önceki taleplere yolculuk sebebiyle oluşan bekleme zamanı, i talebinden önceki taleplere yanıt verme sebebiyle oluşan bekleme zamanı ve talep i 'nin kendi servis zamanıdır.

Bu bağlamda Bertsimas ve Van Ryzin (1991)'e göre alt sınır:

$$T^* \geq \left[\frac{E[\|X - x^*\|]}{1-p} \right] + \left[\frac{\lambda \bar{s}^2}{2(1-p)} \right] + \bar{s} \quad (3.4)$$

Yoğun trafik durumu için, Bertsimas ve Van Ryzin (1991) bir γ sabiti olduğunu ve bu sabitin:

$\gamma = \frac{2}{3\sqrt{2\pi}} \approx 0,266$ olduğunu kanıtlamıştır. Yoğun trafik durumunda, T için, alt sınır:

$$T^* \geq \gamma^2 \left[\frac{\lambda A}{(1-p)^2} \right] - \left[\frac{1-2p}{2\lambda} \right] \quad (3.5)$$

Eşitlik (3.5)'ten çıkarabileceğimiz önemli bir sonuç sistem zamanının $[(1-p)^{-2}]$ bir çarpanı olarak artacağıdır. Oysa ki geleneksel kuyruk sistemlerinde bu artış $[(1-p)^{-1}]$ 'in bir çarpanı olarak gerçekleşir. Aradaki fark, sistem geometrisinden kaynaklanmaktadır. Sistem geometrisi kavramı ve sistem geometrisinin yolculuk zamanları üzerine etkisi takip eden bölümde modelin kurulumu safhasında detaylı olarak irdelenmiştir.

Servis için bekleme hayatın en önemli gerçeklerinden biridir. Özellikle yoğun trafik durumlarında, bekleme zamanının toplam zaman içinde önemi büyüktür. Ortalama bir Amerika Birleşik Devletleri (A.B.D) vatandaşı günde ortalama 30 dakika coğrafi ya da gözüken bir kuyrukta beklemektedir. Yine A.B.D'deki ortalama boş vaktin 1973'ten bu yana %35 azaldığını düşündüğümüzde, hem toplam yaşam kalitesinin artırılması hem de rakip şirketlere kıyasla müşterilerin gözünde daha avantajlı bir noktaya ulaşmak için bekleme zamanlarını azaltmanın önemi büyüktür. [3] Bekleme zamanı aynı kaynak için diğer eş zamanlı taleplerin sayısına, servis zamanına ve taleplerin nasıl programlandığına bağlıdır. Bu doğrultuda bekleme zamanlarını geliştirmenin yolu pek çok kişi için kapasitenin artırımı olmakla birlikte, bu tek ve en etkili yol değildir. Tercih edilen atama politikasının iyileştirilmesi de bekleme zamanını dolayısı ile sistemde geçirilen toplam zamanı etkileyecektir. Atama politikası değişikliği ya da iyileştirilmesi ile bekleme süresinde elde edilebilecek kazancın diğer bir avantajı da kapasite artırımının aksine planlamacıya ek mali yük getirmeyecektir.

Bertsimas ve Van Ryzin (1991), tek araç için, alt sınırlarının performanslarını değişik politikalar için sınanmış ve karşılaştırmıştır. Bu politikaların kısa özetleri aşağıda bulunabilir:

İlk Giren İlk Çıkar Politikası: (İGİÇ)

Gezgin tamirci problemi için en basit politikadır. Bununla birlikte eğer fiziksel olarak oluşturulmuş bir sırada işletilen bir servis politikası ise, müşterilerde eşitlik ve adalet

duygularını pekiştireceğinden kullanımı yaygındır. Bu atama politikasında, talepler, programlayıcıya geldiği sıra ile yanıtlanır. Talepler oluşur oluşmaz, en uygun durumdaki araç, en son yanıtlanılan servisten bir sonraki talep noktasına İGİÇ sırası ile yolculuk eder. Herhangi bir servis tamamlandıktan sonra, servis bekleyen hiç talep kalmaz ise; yeni talepler planlayıcıya ulaşana dek araçlar anlık pozisyonunu korurlar ve planlayıcı tarafından atanacak olan görevlerini beklerler.

Stochastic Queue Median (Rassal Kuyruk Medyanı) Politikası (SQM)

Rassal kuyruk medyanı politikası İGİÇ politikasının bir modifikasyonudur. X ; A alanı içindeki herhangi bir noktayı temsil ederken, x^* , $E[||X-x||]$ 'i minimize eden nokta olsun. Bu politikaya göre, araç A alanın medyanı olan x^* noktasında konumlandırılır. Araç her talebi İGİÇ politikası doğrultusunda yanıtladıktan sonra tekrar x^* noktasına geri döner. Sistemde hiç talep yoksa, araç, yeni talepler planlayıcıya ulaşana dek medyanda bekler.

Gezgin Satıcı Problemi (GSP) Politikası

Planlamacıya ulaşan servis talepleri n adetlik kümeler şeklinde gruplandırılır, burada n politika tarafından daha önceden belirlenmiş bir parametre olsun. N_k , n talepten oluşan k . grup olmak kaydıyla, her grup oluşturulduğunda, bir gezgin satıcı problemi çözülür ve talepler optimal GSP turuna göre yanıtlanır. Eğer herhangi bir anda birden fazla grup yer alırsa, gruplar İGİÇ politikası ile yanıtlanacaktır.

Space Filling Curve (Alan Doldurma Eğrisi) Politikası (SFC)

Alan doldurma politikası ilk defa Bartholdi ve Platzman (1988) tarafından ortaya atılmıştır. Bu politika ile taleplerin ön görüntüleri sistemde tutulur ve sisteme giren talepler, servis alanını kapsayan yinelenen, saat yönündeki çember parçaları ile yanıtlanır. Bu politika ile terminal sürekli bir "talep" olarak ele alınabilir ve her çember parçası için bir kez ziyaret edilebilir.

Ayırma Politikası (PART)

Kare şeklindeki A alanını m^2 adet, $m > 1$ olmak üzere, altbölgelere ayırılır. Burada m politikanın belirlenmiş parametresi olsun. Her altbölge içinde, talepler İGİÇ disiplini dahilinde yanıtlanır ve araç bahsi geçen altbölgede hiç talep kalmayacak şekilde altbölgedeki tüm taleplere yanıt verir. Daha sonra sıradaki altbölgeye yönelerek aynı mantıkta o kısımdaki talepler de karşılanır.

Modifiye Gezgin Satıcı Problemi (MOD GSP) Politikası

Bu politikanın özü, herhangi bir servis alanını altbölgelere bölerek, daha küçük alan parçaları için atama problemine çözüm aramaktır. k , sabit bir pozitif tam sayı olmak üzere ve A

alanının orta noktasından, servis alanını k parçaya bölünür ve n , daha önce belirlenmesi gereken bir parametre olmak üzere, her altbölge için n/k büyüklüğünde kümeler oluşturulur. Kümeler oluşturulduktan sonra, ilgili kümeler kuyruğa eklenir ve İĞİÇ disiplini dahilinde GSP turuna göre optimize edilir.

Nearest Neighbour (En Yakın Komşu Politikası) (NN)

Son olarak ele alınan politika da en yakın komşu politikasıdır. Araç(lar) atandıkları görevi tamamladıktan sonra, en yakın komşusunun talebini karşılamak için harekete geçer. Bu politikayı tercih etmek için bir kaç motivasyonumuz vardır:

- En yakın komşu politikası Bertsimas ve Van Ryzin (1991) tarafından yoğun trafik altında alt sınır olarak kullanılmıştır.
- Klasik bir M/G/1 kuyruğu için ‘En kısa işleme zamanı (SPT) Kuralı’ optimal olarak bilinmektedir. Her ne kadar daha önce de belirttiğimiz üzere problemimizde servis zamanları servis sırasına bağımlılık gösterdiği için SPT kuralının her zaman optimal sonuçları vermesi garanti edilemese de en yakın komşu politikası ve oluşturacağı sonuçların irdelenmesi yararlı olacaktır.

Yukarıda kısaca tanımlanan politikalar içerisinde, Rassal Kuyruk Medyanı (SQM) politikası, hafif trafik durumunda asimtotik olarak optimal sonuç verdiği gözlenmiştir. Bununla birlikte Alan Doldurma Eğrisi (SFC) ve En Yakın Komşu (NN) Politikasını değerlendiren simülasyonlar sonuçları bu iki politikanın da hem hafif hem de yoğun trafik altında etkin çalıştığını gösterir. Özellikle En Yakın Komşu (NN) Politikasının tekrar altını çizmek önemlidir. Analitik zorluklara rağmen simüle edildiğinde, ortalama sistem zamanı, en iyi GSP politikasından yaklaşık olarak % 60 daha fazladır. Bağlı olarak zayıf performans garantisine rağmen, bu politika etkin hesaplama imkanının yanında, uygulanması da kolaydır. Bahsedilen avantajları göz önüne alındığında, bu politikayı pratiğe geçirmek oldukça caziptir. Ek olarak, SFC ve NN politikaları parametrik değildir, yani; bu politikalar trafik durumunda büyük varyasyonların görüldüğü sistemlerde daha kararlı bir karakter izlerler. Bahsi geçen politikalardan çeşitli trafik durumları için daha avantajlı olduğu gösterilen politikalar (SQM ve NN), modelimiz dahilinde oluşturulan ve detayları çalışmamızın beşinci kısmında verilen atama politikamız ile (SAT) çeşitli yoğunluktaki müşteri talepleri altında karşılaştırılmış ve atama algoritmamız için nesnel kıyaslama ölçüleri olarak seçilmiştir.

3.1.2.2 Kapasite Sınırı Olan Çok Araçlı Sistemler İçin Genellemeler

Bertsimas ve Van Ryzin (1993), bir üst bölümde özetlediğimiz analizlerini ve sonuçlarını, ilk olarak kapasite sınırı olmayan m adet aynı özellikteki aracın yer aldığı sistem için genişletme yolunu seçmiştir. Bu kısımda, araçlar bir önceki kısımda olduğu gibi sabit v hızıyla yol almaya devam edecektir.

Bu bağlamda hafif trafik için bulduğu alt sınır:

$$T^* \geq \frac{1}{v} E \left[\min_{x_n \in D} \|X - x^*\| \right] + \bar{s} \quad (3.6)$$

Eşitlik (3.6)'da, talebin olduğu noktaya en yakın aracın ulaşması ile servis süresinin toplamından oluşan beklenen toplam sistem zamanı ile ilgili kısıt verilmiştir.

Yoğun trafik durumu için alt sınır ise aşağıda irdelenebilir.

$$T^* \geq \gamma^2 \frac{\lambda A}{m^2 v^2 (1-p)^2} - \frac{\bar{s}(1-2p)}{2p} \quad (3.7)$$

Eşitlik (3.7)'de görülebileceği üzere sistem zamanı, aynı kapasite kısıtının olmadığı tek araçlı durumda görüldüğü gibi $(1-p)^{-2}$ 'nin bir çarpanı olarak artacaktır. Aynı şekilde, araç sayısını n katına çıkarmak ile araçların hızını n katına çıkarmanın, sistem zaman üzerine etkisinin aynı olacağı yorumunu yapabiliriz.

Tek araç için irdelediğimiz SQM politikası, m adet aracın, A alanının m medyan noktasında konumlandırılması suretiyle m araçlık senaryomuza uygulanabilir. Yeni talepler sisteme girdiğinde, bu talepler en yakın medyan noktasındaki araç tarafından yanıtlanmaya çalışılır. Talep yanıtlandıktan sonra araç medyan noktasına geri dönmek durumundadır. Bertsimas ve Van Ryzin (1993)'in yaptığı simülasyon testleri sonucunda m araç için yukarıda tasvir edilen SQM politikasının, hafif trafik için optimal olduğu gösterilmiştir. Yoğun trafik durumu için birden fazla politika önerilmiştir. Bunlardan en kayda değeri GSP politikasının $G/G/m$ versiyonu olarak gösterilmiştir. Bu politikada talepler n adetlik kümelerde gruplanır ve her küme oluşturulduğunda optimal GSP turu rotalanır. Bertsimas ve Van Ryzin (1993)'e göre, bu politika araç sayısından bağımsız olarak uygulanabilir. Uygulanabilecek diğer bir politika da, rasgele atama politikasıdır. Bu politika ile, Poisson dağılımı, m adet alt poisson dağılımına bölünür ve her araç bir alt prosese atanır, talepler tek araçlı politakada olduğu gibi yanıtlanır. Bir diğer politikada ise, servis alanı A , m adet alt kısma bölünür ve her araç bir alana atanır.

Kendi sorumluluk alanında gerçekleşen her talep o araç tarafından herhangi bir tek araçlık politikanın izlenmesi suretiyle yanıtlanır.

Dinamik gezgin tamirci probleminin araçlarının herbirinin kapasite sınırı olduğu versiyonunda, her aracı, A alanında yer alan belirli bir terminal noktasına atanması ve araçların sadece atandıkları terminali kullanması gerekmektedir. Bu bağlamda, terminallerimizin yer aldığı kümeyi SD ile gösterebilir ve birden fazla aracın aynı terminali kullanabileceği gerçeğinden de yola çıkarak, terminal sayısının 1 ile m arasında bir tam sayı olması gerektiği yorumunu yapabiliriz. Her aracın kapasitesi de, aracın terminali terk ettiği andan terminala tekrar dönene kadar yanıtlayabileceği müşteri sayısıdır. Bu kısıt q ile gösterilebilir ve her araç için aynı tamsayı ile temsil edilir.

Bu kısıtlar ve yoğun trafik altında Bertsimas ve Van Ryzin (1993), tarafından bulunan alt sınır aşağıda bulunabilir:

$$T^* \geq \frac{\gamma^2}{9} \frac{\lambda A(1+1/q)^2}{m^2 v^2 (1-p-2\lambda\bar{r}/mqv)^2} - \frac{\bar{s}(1-2p)}{2p} \quad (3.8)$$

Bu alt sınırdaki r ile temsil ettiğimiz, herhangi bir müşterinin en yakınındaki terminale uzaklığıdır. Ek olarak daha önce de belirttiğimiz üzere $\gamma = \frac{2}{3\sqrt{2\pi}} \approx 0,266$ idi. Eşitlik (3.7) ile (3.8)'de ele aldığımız alt sınırları incelediğimizde, sezgisel olarak da yorumlayabileceğimiz gibi kapasite kısıtı (q) azaldıkça, yani aracın terminali terk ettiği andan terminala tekrar dönene kadar yanıtlayabileceği müşteri sayısı azaldıkça, sistem zamanımız artacaktır. Ek olarak kapasite kısıtımız arttıkça ($q \rightarrow \infty$) kapasite kısıtlarının yer almadığı ve formül (3.7)'de verdiğimiz alt sınırı da kullanabiliriz.

Kapasite kısıtlarının yer aldığı vakalar için ortaya atılan politikaları hafif ve yoğun trafik durumlarında başarılarına göre incelenmiştir. Bertsimas ve Van Ryzin (1993), daha önce irdelediğimiz rassal kuyruk medyanı (SQM) politikasını, m aracın yer aldığı hafif trafik durumu için optimal politika olarak önermiştir. Bu yorumu geliştirirsek, SQM politikası, herhangi bir kapasite için ($q > 0$) hafif trafikte optimal sonuçlar verecektir.

Yoğun trafik altında kapasite kısıtlarının bulunduğu durumlar için ise, üç politika önermiştir. Bahsi geçen çalışmada yoğun trafik, $p + (2\lambda r) / (m v q) \rightarrow 1$ limitinde gösterilmiştir. Yoğun trafik durumu için ilk ele alınan 'Kapasite Kısıtlı Alan Ayırma' (qRP) politikasıdır. Bu politika dahilinde, A alanı, k adet eşit alanlı altkısımlara bölünür. Her altkısımda ortaya çıkan

q adet birbirini takip eden talepler kendi başına bir küme oluşturur ve kümeler oluşum sırasına göre yanıtlanır. qRP politikasının izlendiği sistemlerde, ilk müsait durumdaki araç, küme içindeki q talep için GSP turu oluşturur. Küme içindeki taleplerden biri rasgele seçilir ve tur bu talepten başlar, tur tamamlandıktan sonra araç terminaline geri döner. Bu atama politikasının dezavantajı, sistemin bir parametresi olan k sayısının da optimize edilmesi gerekmesidir.

Araç atama için ikinci önerilen ‘Kapasite Kısıtlı Tur Ayırma’ (qTP) politikasıdır. G/G/m politikasında olduğu gibi, talepler n adetlik kümeler halinde toplanır ve optimal turlar oluşturulur. Herhangi bir noktadan başlamak üzere, turlar herbiri bir araç tarafından yanıtlanmak amacıyla $l = \lceil n / q \rceil$ parçaya ayrılır ve qRP politikasında olduğu gibi oluşum sırasına göre yanıtlanır. Yoğun trafik için önerilen son politika ise ‘Kapasite Kısıtlı Tur Ayırma’ (qTP) politikasının değişikliğe tabi tutulmuş şeklidir. Değişikliğe uğramış qTP’ye göre, A alanımız terminalden çıkan ışınlar ile, sabit bir k sayısı için, k adet eşit alanlı altkısma ayrılır. Her alan için, n / k büyüklüğünde kümeler oluşturulur, bahsi geçen kümelerdeki taleplerin yanıtlanması aynı qTP politikasında irdedeğimiz gibidir. Kümeler oluştuktan sonra, küme kuyruğu oluşturulur ve kümeler ilk müsait araç tarafından oluşum sırasına göre yanıtlanır. (Bertsimas ve Van Ryzin, 1993)

Yukarıda irdedeğimiz Bertsimas ve Van Ryzin (1991 ve 1993) tarafından ortaya atılan alt sınırlara en ciddi katkı, Xu (1994) tarafından yapılmıştır. Xu, çalışmasında ortaya koyduğu yeni alt sınırlara da yer vermiştir. Ancak ilgili alt sınırlar sadece tek araçlı sistem için ve yoğun trafik altında ispatlanabilmiştir. Ek olarak ayırma politikası için (PART) sistem zamanının varyansı üzerine çalışmaları da dikkate değerdir.

Kapasite kısıtları olan birden çok aracın yer aldığı sistemler için irdelenen politikalardan, performansı sınanan ve tescih edilenler (NN ve SQM), modelimizin ve atama politikamızın doğru değerlendirilmesinde yararlanılacaktır.

3.2 Dinamiklik Derecesi

Dinamik bir araç rotalama sisteminin performansının tespit edilmesi ve ölçülmesi, o sistemin objektif olarak değerlendirilmesi adına çok önemlidir. Deterministik ve statik araç rotalama probleminde sistem performansı sadece talep sayısına ve bu taleplerin olduğu noktanın coğrafi konumuna bağlı kalmaktayken, aynı problemin dinamik ayağında dinamik olayların sayısı ve planlama ufku içinde ne zaman olduğu da sistem performansını etkilemektedir. Bu

nedenle, sistemin dinamikliğini, bir algoritma altında ancak farklı durumlar için tasvir edebilecek tek bir ölçütün kurulması sistemin performansı ile ilgilenen herkes için çok değerli olacağı açıktır.

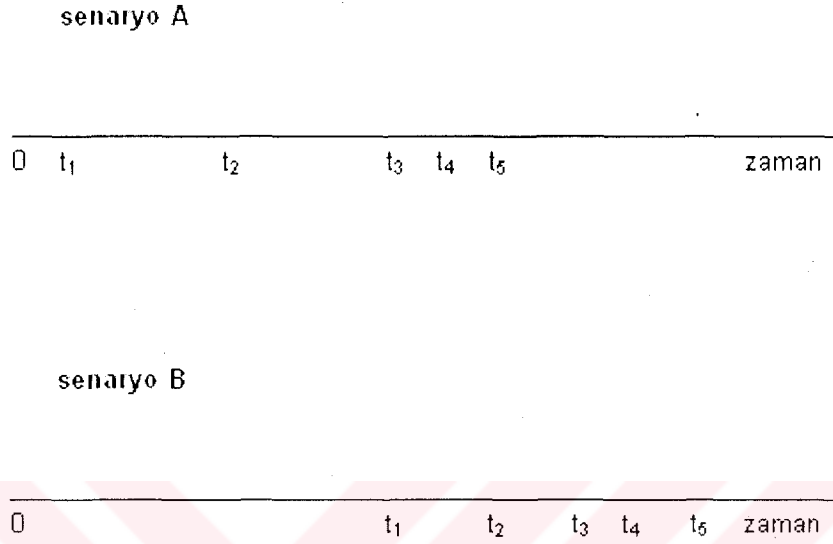
Bir sistemin dinamiklik derecesini tasvir etmek adına umut verici gözükten ölçütler, bir başka sistem için yetersiz olabilir. Bu kısımda, zaman pencerelerinin bulunmadığı dinamik araç rotalama sistemi için dinamiklik derecesini ele alınmaya çalışılmıştır. Zaman penceresiz sistemde, analizimizi etkileyecek üç farklı parametre yer alır, bu parametreler statik müşterilerin sayısı, dinamik müşterilerin sayısı ve dinamik taleplerin sisteme giriş zamanlarıdır.

Pek çok dinamik rotalama sisteminde, ilgili bilgi sadece sistem bağlantılı ve atamaların çoktan başlamış olduğu zaman periyodunda alınmaz. Genellikle, sistem ile ilgili pekçok bilgi operasyon başlamadan da önce toplanmaya başlanır. Bu doğrultuda sistem elindeki kaynakları atamaya başladığı anda, talep havuzunda bir ya da birden çok talep bekliyor olabilir. Sistem operasyonel faza geçtiği andan itibaren sisteme giren yeni bilgilerin miktarı, sistemin dinamiklik derecesini ölçmek adına kullanılabilir. Lund vd. (1996), bu ölçümün en kolay rotalama başlığı altında dinamik talep sayısının toplam sayıya bölünmesi ile çıkarılabileceğini önermiştir. Lund vd. (1996), tarafından ortaya atılan ve *dd* olarak isimlendireceğimiz sistemin dinamikliği aşağıda incelenebilir. Buna göre:

$$dd = \text{Dinamik Taleplerin Sayısı} / \text{Toplam Taleplerin Sayısı} \quad (3.9)$$

(3.9) eşitliği yardımıyla gösterdiğimiz oran ciddi bir ölçüt olmasına rağmen, sistem parametrelerimizden sadece statik ve dinamik müşteri sayısına bağlı bir ölçüt olup sisteme giren dinamik taleplerin geliş zamanlarını dikkate almaz. Durumu netleştirmek için iki örnek durum ele alalım. Her iki durumda da 5 adet talebimiz olsun ve bu taleplerin hepsi de planlama periyodu içinde sisteme girsin. Ancak ilk örneğimizde, dinamik taleplerimiz planlama periyodunun başlangıç safhalarında gelirken, ikinci örneğimizde dinamik taleplerimiz planlama periyodunun daha geç safhalarında sisteme girsin. Her iki örnek için de *dd* değerinin 1 (5/5) olduğu halde, iki senaryonun gerçek dünya uygulamalarında oldukça farklı sonuçlar doğuracağı açıktır. Şekil 3.1’de dinamik araç rotalama problemi dahilinde iki farklı senaryo için planlama periyodunda taleplerin sisteme giriş zamanını gösterilmeye çalışılmıştır. Senaryo A’da talepler, planlama periyodunun bağıl olarak erken döneminde sisteme girerken, senaryo B’de taleplerin planlama periyoduna yayıldığını, hatta geç döneminde daha fazla talebin sisteme girdiğini gözlemlenebilir. Birinci senaryonun

planlayıcıya, dinamik taleplere çözüm üretecek ve reaksiyon gösterecek zamanı sağlama açısından ikinci senaryoya göre daha fazla avantajlı olduğu için planlayıcı tarafından tercih edilmesi önerilebilir. Zira ikinci senaryoda (B) planlayıcı dinamik taleplere uygun aksiyonu alamayabilir. (Lund vd., 1996)



Şekil 3.1 Dinamik taleplerin iki farklı senaryo için sisteme giriş zamanları (Lund vd., 1996)

Analizimizi zaman ekseninden, performans eksenine taşımamız mümkündür. Amaç fonksiyonumuzun katedilen toplam mesafenin minimize edilmesi olduğu durum olarak ele aldığımızda, elde yanıt bekleyen taleplerin sayısını maksimize edecek durumun ve vakanın çözüm kalitemiz açısından en olumlu sonucu verecektir. Şekil 3.1'deki A ve B senaryomuzu tekrar ele aldığımızda, t_5 her iki senaryo için de gerekli olan tüm bilgilerin ancak elde edilebildiği andır. Bu bağlamda t_5 anında planlayıcı bekleyen talepler kümesinden optimal bir GSP turu çıkartabileceği için senaryo A'dan elde edilebilecek turun beklenen uzunluğu, senaryo B'den daha kısa olabilecektir.

Şekil 3.1'de incelediğimiz her iki senaryo için de planlama periyodunun başlangıcında hali hazırda sisteme girilmiş taleplerin varlığından söz etmemiştik. Her A ve B senaryosu için $t=0$ anında n adet hazır talebin yer aldığı A_2 ve B_2 senaryosunu ele alalım. Hazır taleplerin $t=0$ anında yer aldığı ve bunların planlama periyodu içinde sisteme giren taleplere göre öncelikli olarak yanıtlamaya çalışıldığı durum için, senaryo A_2 yönetilmesi daha zor olan senaryodur diyebiliriz. Zira planlamanın erken safhalarında, hazır talepleri yanıtlamaya çalışırken ani talepler de yağmaya başlayacaktır. Yukarıdaki açıklama Senaryo A_2 yi incelenen iki senaryo

içinde en dinamik olan olarak adlandırılmasına neden olabilir. Buna rağmen Lund vd. (1996), göre ilk analizin değerlendirilmesi hem zaman hem de amaç fonksiyonu bazında sezgisel olarak da en doğru seçenek olacaktır.

Çalışmamızda incelediğimiz vakalarda taleplerin sisteme girilmesi sadece planlama periyodu içinde gerçekleşmektedir. Daha açık bir ifade ile, planlama periyodunun başlangıcında ($t=0$ anında) sistemde talep yer almayacaktır. Lund vd. (1996)'nın gösterimini ve notasyonunu kullandığımızda dd oranımızın her vakamız için 1'e eşit olacağı açıktır. Bu gerçek, analizimizin tamamiyle dinamik ortamda gerçekleştirilmeye çalışıldığını yani araç atama probleminin en karmaşık örneğini irdelediğimizin altını çizer. Senaryo A ve B'de de irdelediğimiz, dinamik taleplerin dağılışını tezimiz açısından gündeme getirirsek; her vaka için planlama zamanının başlangıcında belli sayıda kapasitede kaynağımız (aracımız) bulunacağından, sezgisel olarak taleplerin homojen olarak dağıldığı vakalarda araçların kullanım oranı artacak ve beklenen yanıt zamanı azalacaktır. Şüphesiz araç atama algoritmamız her dağılıma maksimum derecede yanıt verecek ve her duruma kolaylıkla adapte edilebilecek şekilde kurulmaya çalışılmıştır. Ancak planlama ufkunun son anlarında yoğunlaşacak olan taleplerin tamamına yanıt vermek, beklenen zaman aralıklarında mümkün olamayabilir, bu ve benzeri durumlarda programcıya taleplerin ertelenmesi, başka bir atama çizelgesi ile yanıtlanması ya da taleplere yanıtın verilebileceği zaman aralığının uzatılması önerilebilir.

3.3 Teknolojik Gereksinimler

Çalışmamızın bu bölümünde, dinamiklik derecesinin çok yüksek olduğu, araç rotalama problemlerinin teorik öğretilerinin gündelik hayatta karşılaşılabilecek daha önce sıraladığımız vakalara uyarlanması mümkün kılacak teknolojiler irdelenmeye çalışılmıştır. Gerekli teknolojileri üç grupta toplayabiliriz: İlk grubu pozisyonlama, ikinci grubu iletişim ekipmanları oluştururken, üçüncü ve tamamlayıcı teknoloji grubunu Coğrafi Bilgi Sistemleri (CBS) teşkil edecektir.

3.3.1 Pozisyonlama Teknolojisi

Nerede olduğunu ve nereye gittiğini göstermeye çalışmak insanoğlunun en eski uğraşlarından biri olmuştur. Pozisyonlama, pek çok aktivite için hayatidir; buna rağmen kullanılan yöntemler çoğu zaman pek de kullanışlı olamamıştır. İnsanlık tarihi boyunca pek çok teknoloji ele alınmış ve probleme çözüm üretilmeye çalışılmıştır. Bu teknolojileri kısaca özetlersek:

- **İşaret Taşları:** Sadece yerel alanlarda yararlı olabilir. En büyük dezavantajı ise çevrel faktörler tarafından konumunun değişmesine veya yıkıma açık olmasıdır.
- **Mevki Tahmini:** Bu teknolojinin uygulanması, çok karmaşık olabilmektedir. Hesapların doğruluğu genellikle çok da hassas olmayan ekipmana bağlıdır ve hatalar yığın şeklinde hızla büyüyebilir.
- **Göksel Hesaplar:** Yüzyıllardır uygulanmasına rağmen karmaşıktır. Sadece havanın ve görüş alanının iyi olduğu gün ve gecelerde işe yarar ve kesinliği, doğruluğu kısıtlıdır.
- **OMEGA:** 1950'li yıllarda geliştirilmeye başlanan pozisyonlama teknolojilerinden biridir. Bu pozisyonlama teknolojisi, bağıl olarak radyo yön işaretlerine dayanarak işler ve denizcilikte oldukça geniş bir kullanım alanı bulmuştur. Bu yöntemi kullanarak elde edilecek sonuçların güvenilirliği kısıtlıdır ve güvenilirlik, radyo dalgalarındaki parazitten etkilendiği için engembeli arazilerde hata payı oldukça yüksektir [23].
- **LORAN (Long Ranger Navigation) :** 1942 yılında denizcilik amaçlı kullanılmak üzere geliştirilmiş bir teknolojidir. Çalışma prensibi, bir ağ içerisinde yer alan merkezlerden belli frekansta, belli zaman aralıklarında yayınlanan sinyallere dayanır. Sinyalleri alan gemi ya da uçak sinyalleri izler ve değişik merkezlerden gelen farklı sinyalleri kayıt edip işleyerek konumunu tespit eder. LORAN'ın en büyük dezavantajı, OMEGA da olduğu gibi küresel bir kapsama alanından yoksun olmasıdır. Dolayısı ile karasal alanlar için bir çözüm üretmekten uzaktır. Uygulama alanının dar oluşuna ek olarak güvenilebilirliği düşüktür, coğrafi durumdan etkilenir. Sistemin tıkanıklığa ulaşma veya bozulma indisi oldukça yüksektir [31].
- **SatNav (Satellite Navigation) :** Düşük-frekanslı Doppler ölçülerine dayanır bu nedenle alıcıda gözlemlenen küçük hareketlere duyarlıdır. Bunun yanında yörüngede az sayıda uydu bulunduğu için güncellenme frekansı çoğu zaman yetersizdir [7].

Sonuç olarak Amerika Birleşik Devletleri Savunma Bakanlığı mümkün bütün pozisyonlama teknolojilerini göz önüne alarak, ordularının dünya çapında pozisyonlama sistemlerine sahip olması gerektiğine karar vermiş, bu sistemin inşası için 12 milyar dolar harcamış ve pozisyonlama teknolojilerinde bir devrim sayılacak olan “Küresel Pozisyonlama Sistemi” (GPS) yaratmıştır. Küresel Pozisyonlama Sistemi (GPS), yürüyüzüne sürekli kodlanmış veri ileten ve bu veriler ışığında uydulardan mesafeyi ölçmek sureti ile yeryüzündeki konumları tam olarak belirlemeyi mümkün kılan bir uydular ağıdır. Uydu sisteminin kurulum maliyeti

sadece 12 milyar dolarla kısıtlı kalmamış, süregelen bakım, onarım ve yenileme uydularının gönderilme masrafları, sistemin maliyetini arttırmıştır. Şüphesiz ki, kuruluş prensibi doğrultusunda tasarımcılarının öncelikli gayesini askeri uygulamalar oluşturmaktaydı. GPS alıcıları, askeri seferlere, birliklerin yayılmasına ve topçu ve füze bataryalarının ateşine yardımcı olabilecek şekilde tasarlanmıştır. İlk GPS uydusunun fırlatıldığı ve sistemin “işler” sayıldığı 1978 yılından 5 yıl sonra GPS teknolojisi, sivil kullanım için kullanılabilir kılındı. 1989 yılında ilk Blok II uyduları fırlatıldı ve ilk defa dünya etrafında 24 saatlik kapsama alanına ulaşıldı [40].

3.3.1.1 Küresel Pozisyonlama Sistemi (GPS) ve Bileşenleri

NAVSTAR sistemi (Navigation Satellite Timing and Ranging, Amerika Birleşik Devletleri Savunma Bakanlığı'nın GPS için kullandığı orjinal isimdir.), uzay kısmından (uydular), kontrol kısmından (yer istasyonları) ve kullanıcı kısmından (kullanıcı ve GPS alıcısı) oluşur. Çalışmamızın bu kısmında GPS sisteminin üç bileşeni incelenmeye çalışılmıştır [15].

Uzay Kısım

Sistemin uzay kısmı, 24 (21 aktif ve 3 adet operasyonel yedek) uydudan oluşur ve bu uydu ağı tüm sistemin kalbini oluşturur. Uydular yeryüzünden 12.000 mil yukarıda, ‘yüksek yörünge’ denilen kısımda konuşlandırılır. Uyduların bu yörüngeye yerleştirilmesinin amacı, yeryüzündeki herhangi bir GPS alıcısının bahsi geçen uydulardan en az dördünü her an görebilmesini sağlamaktır. Uydular saatte 7.000 mil hız ile dünyanın etrafında dönerler ve bu hız, uyduların her 12 saatte bir dünya etrafındaki daireyi tamamlamasını mümkün kılar. Uydular enerjilerini, güneşten alır ve uyduların yaklaşık 10 sene beklenen ömrü vardır. İlk GPS uyduları 1978 yılında yörüngeye sokulmuş olup, 24 uydunun tamamlanması 1994 yılında gerçekleşmiştir.

Her bir uydu, birkaç değişik frekansta düşük radyo dalgaları (L1,L2 vb.) yayar. Sivil GPS alıcıları sadece L1 frekansını dinler. Bu dalgalar, cam, plastik gibi nesnelerden geçerken, dağ ve bina gibi pek çok katı cisimden geçemez. Bu da sistemin, sinyalin ulaşmasının imkansız olduğu, bina ve mağaraların içi, garajlar ve diğer yeraltı konumları ile su altı dışında her yerde çalışmasını mümkün kılar.

Kontrol Kısım

Kontrol kısmı adından da anlaşılacağı gibi, GPS uydularını izleyerek kontrol eder ve bu uydulara doğrulanmış yörünge ve zaman bilgisini temin eder. Yeryüzünde farklı beş noktaya konumlandırılmış, beş adet kontrol istasyonu vardır. Bu istasyonların dördü insansız olup,

beşincisi ‘ana’ kontrol istasyonudur. Dört insansız kontrol istasyonu, uydulardan sürekli veri alır ve bu veriyi ana kontrol istasyonuna gönderir. Ana istasyon, uydu verisini ‘düzeltir’ ve iki adet anten konumu ile GPS uydularına bu düzeltilmiş veriyi gönderir.

Kullanıcı Kısmı

Kullanıcı kısmı en basit anlatım ile kullanıcı ve GPS alıcısından oluşur. GPS alıcı fiyatları 10 yıl içinde birkaç bin dolardan, 50 dolara kadar düşmüştür ve bu trend hala da sürmektedir [8]. Şüphesiz bu maliyet avantajı, sistemi kullananların ve potansiyel kullanıcılarının sayısını arttırmaktadır. Sistemin kullanıcıları arasında, pilotlar, dağcılar, avcılar, askeri görevliler ile şu an nerede olduğunu veya nereye gitmek istediğini bilmek herkes yer alır. Şüphesiz ki, araç takip sistemleri kullanıcıları da GPS teknolojisinin avantajlarından yararlananlar arasındadır.

3.3.1.2 GPS Çalışma Prensibi

GPS’in çalışma prensibi iki esasa dayanır. GPS alıcısı, uyduların nerede olduğunu (konumunu) ve ne kadar uzakta olduğunu (uzaklık) bilmesi şarttır. GPS alıcısına, uydulardan iki kodlanmış veri ulaşır. Birinci tip veri (almanac), uyduların yaklaşık pozisyonunu, hızını, yörüngesini ve yüksekliğini içerir, sürekli yayınlanır ve GPS alıcısının hafızasında saklanır. Birinci tip veri, yer istasyonlarında düzeltilir ve tekrar uydulara kesin veri olarak iletilir. Bahsi geçen ikinci veri ile GPS alıcısı, uyduların konumunu tam olarak bilmektedir [39].

Yer tesbiti için, GPS alıcısı TDOA (Time Difference of Arrival) adı verilen bir teknik kullanmaktadır: Her uydudan gelen sinyal, o uydunun belli bir andaki mutlak konum bilgisini içermekte ve üç ya da daha çok uydudan sinyal alabilmesi halinde, GPS alıcısı sinyallerin kendisine ulaşma sürelerinin yardımıyla kendi bulunduğu konumu enlem ve boylam olarak hesaplayabilmektedir. Uyduların konumlandırıldığı konum ve irtifa sebebi ile, dünya üzerindeki alan nere olursa olsun, alıcı her an üç veya daha fazla GPS uydusu görebilir. Dolayısı ile, GPS alıcısının, bulunduğu konumu tesbit etmesi hiçbir zaman sorun olmamaktadır [37].

TDOA tekniği kullanılarak tesbit edilen konum ile alıcısının gerçek konumu arasındaki mesafe (başka bir ifade ile hata payı) istatistiksel olarak 10 metreyi aşmamaktadır. Bu hassasiyet 1 Mayıs 2000 tarihinde ABD hükümetinin yeni uygulamaya giren kararı ile gerçekleşmiştir. Bu tarihten önce askeri uygulamalarda 10 metre, sivil uygulamalarda 100 metre hassaslık söz konusu idi [37]. Bununla birlikte Langley (1997)’e göre, GPS (NAVSTAR) tek uydu bazlı pozisyonlama sistemi değildir. 1 Ocak 1996’da 24 adet operasyonel uydunun yörüngeye sokularak sistemin tamamlandığı Rusya Federasyonu

tarafından işletilen GLONASS ile tamamiyle sivil Avrupalı organizasyonlar ve kuruluşlar tarafından işletilecek olan GNSS (Galileo) diğer küresel pozisyonlama sistemleri olarak sayılabilir. Bu üç küresel sisteme ek olarak geliştirilmiş ve işler olan bazı yerel pozisyonlama sistemleri mevcuttur; WAAS (A.B.D), CWAAS (Kanada), MSAS (Japonya) ve EGNOS (Avrupa). 2005 yılında sinyal göndermesi ve 2008 yılında tamamen işlerlik kazanması planlanan GNSS, getireceği farklı hizmet ve kalite ile GPS'e en ciddi rakip olacaktır [5].

3.3.2 İletişim Teknolojisi

Araç ile araç atama merkezi arasındaki iletişimi sağlayan iletişim teknolojisi, atama sistemi içinde çok önemli rol üstlenir. Günümüzde pek çok uygulamada standard olarak Mobil İletişim için Küresel Sistem (GSM) şebekeleri kullanılsa da, Mobitex ve telsiz sistemleri de bu veri akışını transfer etmeye muktedir teknolojilerdir. Özellikle Mobitex altının çizilmesi gereken, mobil üniteler arasında veri iletişimi için tasarlanan bir teknolojidir ve son yıllarda sayısı katlanarak artan 1,2 milyon profesyonel kullanıcısı vardır [36]. GSM ve Mobitex sistemler gibi baz istasyonlarından oluşan yeni nesil teknolojiler ile monolog ve analog telsiz sistemlerini farklı iki teknoloji grubu olarak ele alabiliriz. Veri transferinin fizibilitesi açısından bu teknolojiler arasındaki asıl fark, altyapı ve operasyonel maliyetlerinden kaynaklanır. Mobil telefon iletişim sisteminin işletme maliyeti bağıl olarak daha yüksektir, buna karşın altyapı maliyetleri, gerekli temel teknolojinin GSM operatörleri tarafından tedarik edilmesi sebebi ile çok düşüktür. Ek olarak günümüzde, gelişmiş ülkelerin tamamında GSM şirketleri, müşterilerine tam kapsama sunmaktadır. Radyo bazlı, telsiz sistemlerinin kurulum maliyeti ise çok yüksektir. Zira radyo yayını mümkün kılan antenlerinin dikilmesi ve hayli maliyetli radyo ekipmanının her araca monte edilmesi gerekir. Radyo bazlı telsiz sistemlerinin, yüksek kurulum maliyetine nazaran, işletme maliyeti neredeyse ihmal edilebilecek kadar azdır. Bunun yanında, telsiz sistemi, GSM ve Mobitex sistemlerinin sunduğu esnekliği sunamamaktadır.

GSM teknolojisi kullanılan sistemlerde, işletme maliyetinin esasını, konum ve hız bilgilerinin, GSM şebekesinin sunduğu SMS (Kısa Mesaj Servisi) veya GPRS (General Packet Radio Service) kullanılarak araç atama merkezine gönderilmesi oluşturmaktadır. Özellikle GPRS'in 171 kbps'a ulaşabilen asimetric veri transver hızı, açık teknoloji oluşu ve dünya üzerinde 49 ülkedeki 121 GPRS ağının bulunması (Mart 2003) [16] ile Mobitex'in 8 kbps'da kalan simetrik veri transver hızı ve tescilli bir standard oluşu GSM'i birkaç adım öne çıkarmaktadır [36]. 2000'li yılların başında mobil iletişimin en yaygın yararlanıldığı ve en geniş halk kitleleri tarafından benimsendiği ülkelerden biri olan Danimarka'da, pazar payının yaklaşık

%25'üne sahip, Sonofon Şirketi tarifelerine göre, bir adet SMS gönderme maliyeti 0,50 dkr, \$ 0,07 iken [35]; Mart 2003, itibari ile Turkcell© FiloCELL tarifi içinde atılan her SMS 60.000TL (~ 0,03 \$) olarak ücretlendirilmiştir [41]. Yani, 24 saat boyunca saat başı atama merkezine bilgi gönderen bir aracın takibinin SMS ile yapılmasının günlük işletme maliyeti yaklaşık \$ 0,75 civarındadır. SMS teknolojinden yararlanılarak 10 adet aracın takip edilmesinin aylık maliyeti Turkcell© FiloCELL tarifi dahilinde yaklaşık olarak \$ 225'dir. Tarihsel veriler de, iletişim ücretlerinin yıllar boyunca düştüğünü gösterir. Bu şüphesiz, alternatif iletişim teknolojileri arasından telekomünikasyon bazlı olanı seçmek için yeterli motivasyonu sağlayacaktır.

İdeal olarak, atama merkezi, herhangi bir zamanda aracın ve sürücüsünün hangi durumda olduğunu bilir. Yukarıdaki ifadenin bir sonucu olarak, sistemin operasyonel maliyetleri göz önüne alındığında bu uygulama fizibil olmayan sonuçlar doğurabilir. Bunu önlemek adına, pratik uygulamalarda belirli aralıklarla bilgi transferi ve araçların yaklaşık yerini tahmin etmede kullanacağımız enterpolasyon kullanılabilir.

Alternatif olarak, sürücü, her müşteri talebini yerine getirdikten sonra, o anki durumunu ve pozisyonunu atama merkezine SMS ile bildirebilir. Bu yaklaşım şüphesiz ki, bir sonraki müşteriye kimin servis vereceğini noktasındaki atama kararı açısından, ideal duruma kıyasla içerdiği bilgi düzeyi daha sığdır. Yeni durum bilgisi atama merkezine ulaştıktan sonra, araç programlayıcı, planlamış olduğu rotada veya atama prosesinde değişikliğe gidebilir, bu durumda diğer araçlara SMS ile yeni rotadaki rollerini bildirmesi gerekecektir.

Diğer bir iletişim ve veri toplama seçeneği de, hesaplı GPS cihazları ile kablosuz veri iletimi için GPRS sisteminin tercih edilmesidir. GSM Ağlarından ve mobil telefonlardan yararlanılan sistem ile birlikte büyük ölçekli veri toplamak için en çok tercih edilen teknolojilerden biridir. Örneğin, son yıllarda kullanımı hızla artan GPRS teknolojinden yararlanılarak aracın takip edilmesi ilgili kurumlara ciddi maliyet avantajı sağlamaktadır. Bahsi geçen cazip maliyete ulaşmamızı sağlayan iki faktörden ilki GPRS teknolojisinin iletilen veri bazında ücretlenmeyi mümkün kılması (1024 kilobyte için yaklaşık maliyetin \$ 0,95), diğeri ise aracın yerinin (enlem, boylam ve yerden yüksekliği) ile hızının yaklaşık 50 byte'lık veri dosyaları ile iletilmesinin olanaklı oluşudur. GPRS teknolojinden yararlanılarak her bir aracın konumunun gün içerisinde 100 defa sorgulandığı ve 10 adet aracın takip edilmesinin aylık maliyeti Turkcell© GPRSCELL tarifi dahilinde yaklaşık olarak \$ 1,38'dir [40]. GPRS teknolojisinin kullanılması karşılaşılan maliyet, SMS teknolojisi kullanılarak ulaşılan maliyetin yaklaşık

olarak 163'te biridir. GPRS teknolojisinin, yine GSM kaynaklı SMS teknolojisine üstün maliyet avantajının bulunmasına rağmen en büyük yetersizliği her baz istasyonun GPRS desteği verememesinden kaynaklanmaktadır. Ülkemizde ve dünyada pek çok ülkede operatörler büyük şehirlerdeki baz istasyonlarını GPRS desteği verecek şekilde yatırımlarını tamamlamış olsa da, küçük şehirlerde ve kırsal alanlarda yatırım eksikleri mevcuttur. Ek yatırımlar tamamlandıktan sonra araç takibini GSM kaynaklı GPRS teknolojisine ile gerçekleştirmenin tercih edilirliliği artacaktır.

3.3.3 Coğrafi Bilgi Sistemleri (CBS)

Toplumlar, gün geçtikçe, küresel ekonomi ve toplum anlayışının da etkisi ile, bilginin coğrafi açıdan yönetiminin mutlak bir ihtiyaç olduğunun farkına varmaktadır. Bununla birlikte, teknolojinin sık sık görülen negatif etkileri, eldeki kaynakların daha akıllıca yönlendirilmesini gerektirmektedir. Coğrafi Bilgi Sistemleri (CBS) bu sorunları çözmek için yeni teknikler ve araçlar sağlar; yani pek çok çeşitli işi daha etkin ve verimli bir şekilde yapılmasına yardımcı olur. CBS, mekanik olarak yeryüzüne referans verilmiş olan verinin elde edilmesine, saklanmasına, kontrol edilmesine, manipülasyonuna, analizine ve gösterimine yarayan bir sistemdir. (Goodchild vd., 1992)

Coğrafi bilgi teknolojileri 1980'lerde olgunlaşmaya başlamış ve dijital haritalar ile CBS, 1990'lı yıllarda ciddi ilerleme kaydetmiştir. Sanayileşmiş ülkelerin pek çoğu, detaylı yol ağları veritabanlarına sahiptir. Mapinfo© [24] ve Multimap© [27] Şirketleri, pek çok ülkedeki yüzlerce şehrin 100 metre duyarlılıkta detaylı haritalarını kullanıcılarına sunar. Brande-Lavridsen (2002)'ye göre, Avrupa çapında ise dijital ortama taşınan en etkileyici coğrafi veritabanlarından biri Danimarka'ya aittir. Danimarka'da DAV (Danimarka Yol ve Adres Veritabanı), ülkedeki her adresin detaylı bilgisi ile ilişkili dijital yol veritabanı sunar. DAV veritabanı, posta kodlarını, cadde adlarını, rota numaralarını, yolların sınıflandırılmasını, caddenin başında ve sonunda yer alan mekanların sokak numaralarına ilişkin bilgiyi sağlar. Ülkemizde de İstanbul Belediyesi, İstanbul İli için detaylı bir yol haritası hazırlamış ve bu haritayı, internet adresinde ücretsiz olarak kullanıma açmıştır [18]. Bununla birlikte İkiNokta [19] ve Gürbüz Yayınları [14] Şirketleri ellerinde hali hazırda olan basılı verinin bir yazılım yardımı ile bilgisayar ortamına aktarımını tamamlamışlardır.

CBS, dünya çapında pek çok kurum ve kuruluş tarafından başarılı, güvenilir ve üretken bir şekilde kullanılmaktadır. Bu kullanım yazılım ve donanım fiyatlarının azalmasına, bilgisayar teknolojilerinin ilerlemesine ve bilgisayar çevre elemanlarındaki (sayısallaştırıcı, tarayıcı,

yazıcı vb.) gelişmelere bağlı olarak hızla artmaktadır. İlgili teknolojinin araç atama problemine uygulanabilmesinde, çözüm algoritmasının, problemi kısa zamanda çözebilmesi için büyük miktardaki coğrafi veriyi hızla işleyebilecek kapasitede olması bir koşuldur [32]. Bununla beraber, pratik rotalama uygulamalarında son derece önemli bir rol alan, yol ağlarında en kısa yolların hesaplanması ile ilgili tartışmalara bu tezin kapsamı içerisinde yer verilmeyecektir. Yine de Mercedes©'in resmi internet adresinde ücretsiz olarak kullanılabilen rota uygulaması Türkiye içinde seçilen iki yerleşim merkezi arasında, hangi yol güzergahından gidilebileceğini tespit etmeden kullanılabilecek başarılı bir uygulamadır. Uygulama ile seçilen iki nokta arasındaki en uygun güzergahı “en kısa mesafe” ya da “en kısa zaman” kriterlerine göre belirlenebilmekte ve bu güzergaha ilişkin bir harita irdelenebilmektedir. Yakıt bedeli ve benzin tüketimi değerlerinin girilmesiyle planlanan seyahatin yaklaşık maliyeti de öngörülebilmektedir [26].

Yukarıda detayları ile temel karaktersitiklerini, bazı maliyet analizleri ile tartışmaya çalıştığımız pozisyonlama sistemleri ile herhangi bir araç filosunun herhangi bir üyesinin tam pozisyonunun yüksek duyarlılık ile tespit edilebildiği, araçların konumlarının istenilen detayda bir harita üzerinde coğrafi bilgi sistemleri yardımıyla gösterilebildiği ve güncel verilerin son yıllarda geliştirilmiş olan iletişim teknolojileri ile hem kullanıcılara hem de planlayıcılara çift taraflı iletilmesinin mümkün ve fizibil olduğu gösterilmeye çalışılmıştır.

4. BEKLENEN YOLCULUK MESAFESİ ve ZAMANLARININ HESAPLANMASI

Yolculuk zamanı, ilgili iki nokta arasındaki rotayı kat etmek için gereken zaman olup, ulaşım için temel bir ölçüttür. Yolculuk zamanı, içinde ulaşım mühendisleri ve planlayıcıları, iş adamları, tüketiciler ve idarecilerin de yer aldığı geniş bir kesim tarafından ele alınan ve irdelenen yaygın bir kavramdır. Çalışmamızın bu kısmının ana gayesi, İstanbul Kenti için kuracağımız model ve devamında seçeceğimiz politikanın belirlenmesinde çok önemli bir rol yüklenecek olan yolculuk mesafesini ve dolayısı ile yolculuk zamanını gerçekçi bir şekilde temsil edebilecek analiz, tecrübe ve bilgi dermesi oluşturmaktır. Bu analizleri gerçekleştirirken literatürde daha önce yerini almış temel kabul ve eşitliklerden yararlanılmıştır. İstanbul Kenti veri ve karakterisitikleri doğrultusunda formülasyonda değişiklikler ve eklemeler yapılarak, modelin geçerliliği ve güvenilirliği arttırılmaya çalışılmıştır. Sonuç olarak karmaşık formülasyonlardan ziyade günlük uygulamalarda da kolaylıkla kullanılabilir yaklaşımlar ortaya konulmuştur.

4.1 Acil Servis Modeli

Yukarıda ifade edilmeye çalışıldığı üzere, bu çalışmamızın amacı gerçek dünyayı mümkün olduğu kadar temsil edebilecek bununla birlikte hesaplama tekniği ve zamanı olarak karmaşık olmayan bir model ortaya koymaya çalışmaktır. Öncelikle çalışmamızın ileriki kısımlarında ortaya koymuş olduğumuz kapsamlı modelin hesaplama ve atama mantığını daha rahat kurmak adına sistem parametreleri başlangıç itibarı ile en basit olarak ele alınmış ve modelin zamanla arzu edilen detay, gerçekçilik ve yönetilebilirlik seviyesine getirilmesine gayret gösterilmiştir.

Öncelikle belirli bir alan ve bu alanda meydana gelen acil durumlara müdahale etmeye çalışan, kapasite kısıtlarının olmadığı tek bir servis aracını ele alalım. Acil servis modeli adını verdiğimiz nispeten basit olan modelde, acil durumlara müdahale edebilecek bir aracın talepler doğrultusunda hareket mekanizması ve bunun sonucunda beklenen yolculuk mesafesini ile buna bağlı olarak da yolculuk zamanı irdelenmiştir.

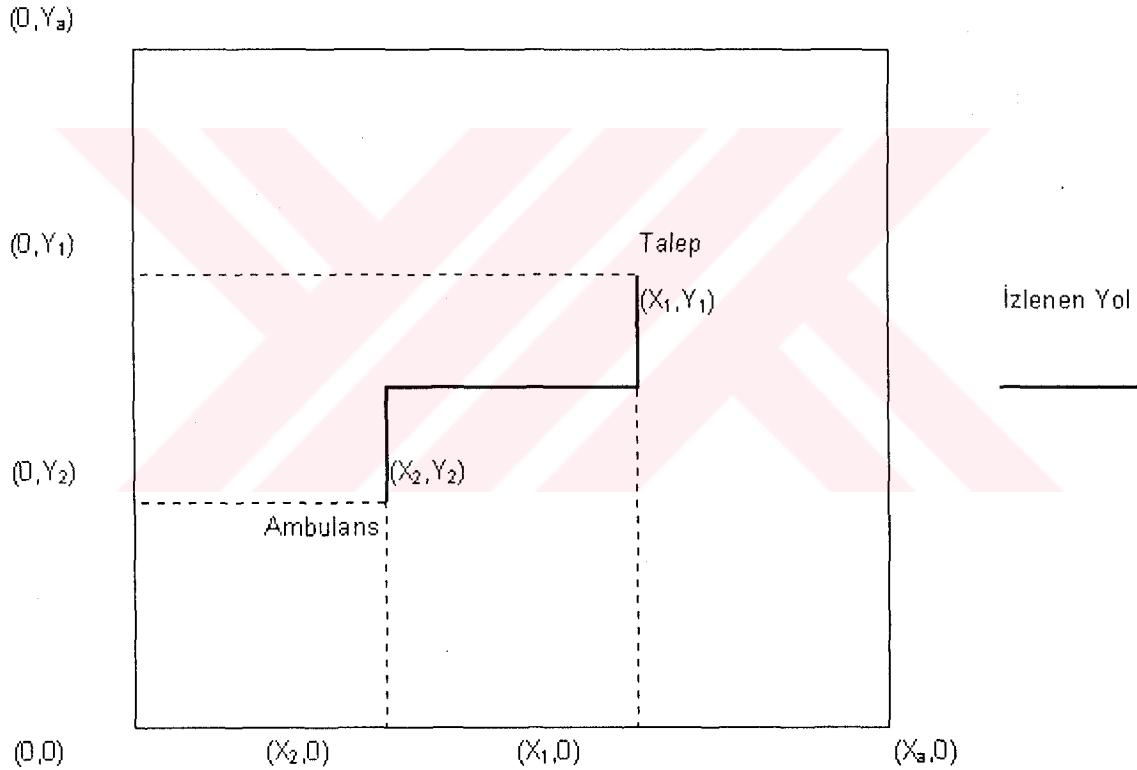
Servis için talebin geldiği noktanın konumunun (X_1, Y_1) ve servis aracın (ambulans, trafik otosu vs.) konumunun (X_2, Y_2) olduğunu ve bu iki noktanın dikdörtgen şeklindeki servis alanında düzgün olarak dağıldığını ve birbirinden bağımsız olduğunu kabul edelim.

İki nokta arasında yolculuk mesafesinin hesaplanmasında birkaç farklı yol kullanılabilir. Bunlardan birincisi kuş uçuşu mesafedir (öklitsel mesafe). Bu yaklaşıma göre yolculuk

mesafesi (D) Pisagor Bağntısından şu şekilde hesaplanabilir:

$$D = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (4.1)$$

Öklitsel mesafe, gerçekten iki nokta arasında kuş uçuşu (doğrusal) ulaşım yapılabildiğinde kullanılır. Öklitsel mesafenin, bir diğer versiyonu da öklitsel mesafenin karesinin alındığı durumdur. Öklitsel mesafenin karesini, kuş uçuşu (doğrusal) ulaşımın mümkün olduğu ancak aşırı mesafeleri kat etmeyi önlemek istediğimiz durumlarda tercih edebiliriz. (Herhangi bir uzun mesafeyi temsil eden sayının karesinin alınması, şüphesiz ki daha büyük bir sayının oluşumu ile sonuçlanacaktır. Elde edilen bu daha büyük sayıyı, minimize etmeyi planladığımız bir amaç fonksiyonunda kullanabiliriz.)



Şekil 4.1 Talep ve ambulansın konumları ile ambulansın izleyeceği yol

Diğer bir alternatif de doğrusal (rectilinear) mesafe yaklaşımıdır. Buna göre iki nokta arasındaki mesafe Şekil 4.1’de irdelenebileceği üzere eksenlere paralel ve dik açılar ile kat edilir. Bu yaklaşıma göre ise yolculuk mesafesi (D) şu şekilde hesaplanabilir:

$$D = |X_2 - X_1| + |Y_2 - Y_1| \quad (4.2)$$

Doğrulu mesafe yaklaşımı özellikle fabrikalar, Amerikan Şehirleri gibi dikdörtgensel ağlar için oldukça sık kullanılır. Bu sebeptendir ki bu yaklaşım literatürde zaman zaman “Manhattan Mesafesi” olarak da adlandırılmaktadır. Özellikle şehir içerisinde yolculuk mesafelerini hesaplamakta ikinci yaklaşım ülkemizdeki şehir planlama ve yolların inşa mantalitesi göz önüne alındığında daha gerçekçi olacaktır. Çalışmamızda ve kurduğumuz modelde şehriçi mesafelerinin hesaplanmasında doğrulu (Manhattan) yaklaşımı göz önüne alınmıştır.

4.1.1 Acil Servis Modelinin Basitleştirilmiş Uygulaması

Beklenen yolculuk mesafesi hesaplamalarına, yukarıda irdelediğimiz modelin daha basit bir uygulamasını ele alarak başlanılabilir. Öncelikle sadece bir ekseninde (tek boyutta) hareket eden, kapasite kısıtlarının olmadığı tek bir servis aracını ele alalım. Bu uygulamamızda servis aracın bir otoyol üzerinde yardıma ihtiyacı olan insanlar için ileri-geri hareket edeceğini kabul edelim. Uygulamamızı olasılık hesaplarından ve onun uygulamalarından yararlanarak modellemeye ve incelemeye çalışalım.

Öncelikle bu modelimizi kurmak için sırası ile:

İlgilendiğimiz rasgele değişkenlerin tanımlanmış, birleşik örnek uzayın oluşturulmuş ve birleşik olasılık dağılımı örnek uzay üzerinde tanımlanmış olması gerekir.

Rasgele değişkenler (Random Variables)

Otoyolun uzunluğunu bir birim mesafede kabul edelim. Öyle ise iki kilit rastgele değişkenimiz:

X_1 = Acil Servis ihtiyacının oluştuğu nokta olsun, $0 \leq X_1 \leq 1$ ve,

X_2 = Acil Servis aracının konumu olsun, $0 \leq X_2 \leq 1$

Eşitlik 4.2’yi modelimize uygularsak, U dönüşlerinin her noktada yapılabildiğini kabul ederek, sadece X ekseninde yolculuk mesafesi (D) şu şekilde hesaplanabilir:

$$D = |X_2 - X_1| \quad (4.3)$$

Birleşik Örnek Uzay

Birleşik örnek uzay, analitik düzlemdeki birinci bölgede yer alan birim karedir. ($0 \leq X_1 \leq 1$, $0 \leq X_2 \leq 1$)

Birleşik Olasılık Dağılımı

Ele aldığımız ilk model ile paralel şekilde servis aracının konumunu ve acil durumun oluştuğu

noktaları otoyol üzerinde bağımsız ve düzgün kabul edelim. X_1 ve X_2 sürekli rasgele değişkenler olduğundan birleşik olasılık yoğunluk fonksiyonu (joint probability density function) üzerinden tartışmamıza devam edebiliriz.

$$f_{x_1, x_2}(x_1, x_2) = f_{x_1}(x_1)f_{x_2}(x_2) \quad (4.4)$$

$$f_{x_1, x_2}(x_1, x_2) = \begin{cases} 1.1 = 1 \dots\dots 0 \leq X_1, X_2 \leq 1 \\ 0 \dots\dots\dots 0 > X_1 \vee X_2 > 1 \end{cases}$$

Eşitlik (4.3) ile ilişkili olasılık hesaplarını gerçekleştirebilmek için Şekil 4.2’de izleyebileceğimiz örnek uzaydan yararlanılabilir. Bu doğrultuda D ’nin kümülatif yoğunluk fonksiyonunu ($F_D(y)$) (cdf) hesaplamaya ve olayın ortaya çıkacağı ($D < y$) noktalar kümesini temsil etmeye çalışılmıştır.

Horald (1982)’e göre kümülatif yoğunluk fonksiyonunu aşağıdaki gibi bulabiliriz:

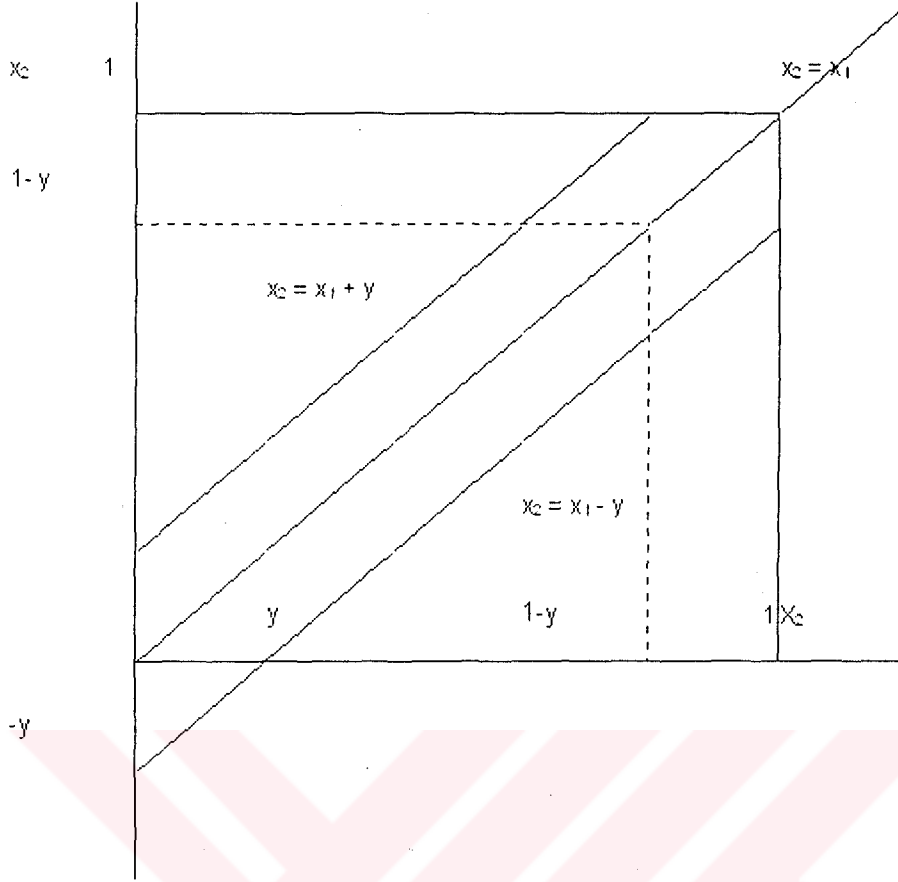
$$F_D(y) = P\{D \leq y\} \quad (4.5)$$

Eşitlik 4.3’te gösterdiğimiz yolculuk mesafesi fonksiyonumuzu eşitlik (4.5)’te yerine koyarsak eşitlik (4.6)’da yer alan formüle ulaşılabilir.

$$F_D(y) = P\{|X_1 - X_2| \leq y\} \quad (4.6)$$

Mutlak değer işaretini ortadan kaldırarak analizimizi sürdürmek için problemimizi iki kısımda incelememiz doğru olacaktır. Buna göre 1. durumda $X_1 \geq X_2$ ve ikinci durumda $X_1 < X_2$ kısmını irdeleyeceğiz. $X_1 \geq X_2$ durumu için X_1 ve X_2 ’ye ait değerler $x_2 = x_1$ ile $x_2 \geq x_1 - y$ doğruları arasında yer alacaktır. 2. durumda ise ilgili değerler $x_2 = x_1$ ile $x_2 \geq x_1 + y$ doğruları arasında yer alacaktır. (Şekil 4.2)

$x_2 \geq x_1 - y$ ve $x_2 \geq x_1 + y$ doğruları arasında yer alan alan ile ilgilendiğimiz olayın ortaya çıkacağı ($D < y$) noktalar kümesini temsil edecektir. Eşitlik (4.4)’te irdelediğimiz: $f_{x_1, x_2}(x_1, x_2)$ fonksiyonunun bahsi geçen noktalar kümeler doğrultusunda integralini alındığında $F_D(y)$ ’ye ulaşılabilir.



Şekil 4.2 X_1 ve X_2 için birleşik örnek uzay (1 x 1) lik kare

Çalışmamızın bu kısmında, integral olarak $F_D(y)$ 'ye ulaşmak yerine, integral hesabı ile aynı sonuca götürecek olan alan hesaplamaları ile $F_D(y)$ 'ye ulaşmak tercih edilmiştir. $x_2 \geq x_1 - y$ ve $x_2 \leq x_1 + y$ doğruları arasında kalan ve analitik düzlemin birinci bölgesinde yer alan iki yamuğun oluşturduğu toplam alan bize alacağımız integral ile aynı sonucu verecektir. Bu doğrultuda 1 br²'lik karenin alanından, her bir kenarının (1-y) birim olan 2 adet ikizkenar dik üçgenin alanını çıkararak eşitlik (4.7)'de irdelenebileceği üzere $F_D(y)$ 'ye ulaşılmıştır.

$$F_D(y) = 1 - 2\left(\frac{1}{2}\right)(1-y)^2 \quad 0 \leq y \leq 1 \quad (4.7)$$

Olasılık yoğunluk fonksiyonuna (pdf) ulaşmak istenildiğinde (4.7)'de yer alan eşitliğin türevini almamız yeterli olacaktır.

$$f_D(y) = \frac{dF_D(y)}{dy} = 2(1-y) \quad 0 \leq y \leq 1 \quad (4.8)$$

$f_D(y)$ fonksiyonumuz, eşitlik (4.8)'e göre, $0 \leq y \leq 1$ aralığı dışında 0 değeri alacaktır.

Analizimizde araştırdığımız beklenen yolculuk mesafesini ($E[D]$) hesaplayabilmek için gerekli olan fonksiyon eşitlik (4.8) elde edilerek sağlanmıştır.

$$E[D] = \int_{-\infty}^{\infty} y f_D(y) dy \quad (4.9)$$

Eşitlik (4.9)'da verilen beklenen değer fonksiyonuna, eşitlik (4.8)'de elde edilen yoğunluk fonksiyonu uygulandığında ise:

$$E[D] = \int_0^1 y 2(1-y) dy = 1/3 \quad (4.10)$$

$E[D] = 1/3$ 'e ulaşılabilir.

Benzer şekilde D 'nin varyansını da bulmaya çalışırsak:

$$\sigma^2 = E[(D - E[D])^2] = E[D^2] - (E[D])^2 \quad (4.11)$$

Eşitlik (4.11)'den görülebileceği üzere D 'nin varyansını bulmak için $E[D]$ ve $E[D^2]$ değerlerine ihtiyaç duymaktayız. $E[D]$ değerini eşitlik (4.10)'dan çekebilir ve $E[D^2]$ için ise aşağıda bulunabilecek olan, eşitlik (4.12) kullanılabilir.

$$E[D^2] = \int_0^1 y^2 2(1-y) dy = \int_0^1 2y^2 - 2y^3 dy = 1/6 \quad (4.12)$$

Eşitlik (4.12)'den elde ettiğimiz sonucu (4.11)'de verdiğimiz eşitlikte yerine koyduğumuzda, D 'nin varyansı olarak ulaşılan sonuç eşitlik (4.13)'te görebileceğimiz üzere $1/18$ 'dir.

$$\sigma^2 = E[D^2] - (E[D])^2 = 1/6 - (1/3)^2 = 1/18 \quad (4.13)$$

Tek boyut ve eksen için elde edilen bu yararlı sonuçlardan sonra kısım 4.1'de irdelemeye başlanılan acil servis modeline geri dönülebilir. Amacımız tek boyut için elde ettiğimiz formül ve sonuçları iki boyut ve iki eksen üzerindeki hareketleri de kapar hale getirmektir. Doğrulu (rectilinear) mesafe yaklaşımdan elde edilen yolculuk mesafesi (D), 4. bölümün başında eşitlik (4.2)'de verilmişti. Beklenen yolculuk zamanına ulaşmak için eşitlik (4.10)'da elde edilmiş olan sonuca başvurulmuştur. Buna göre:

$$E[D] = 1/3 [X_a + Y_a] \quad (4.14)$$

Eşitlik (4.14)'te yer alan X_a ve Y_a Şekil 4.1'de tanımladığımız dikdörtgenin kenarları

olacaktır. Diğer bir ifade ile, kare şeklinde 9 km²'lik bir arazilik bir servis alanına hizmet vermeye çalışan bir sağlık ocağında konuşlandırılmış bir ambulans için herhangi bir zamanda herhangi bir olay için beklenen yolculuk mesafesi 2 km [1/3 x (3 +3)] olacaktır.

4.2 Optimal Alan Seçimi ve Optimal Olmayan Alanlar için Sonuçlar

Kısım 4.1'de ele aldığımız servis alanımızı bir kenarı X_a diğer kenarı Y_a olan dikdörtgen şeklinde bir arazi olarak kabul etmiş ve sonuçlarımıza da bu kabule göre ulaşmıştık. Çalışmamızın bu kısmında öncelikle beklenen yolculuk mesafesini ($E[D]$ 'yi) minimize etmek için bölgenin optimal boyutlarını bulmaya çalışılmıştır. Bu sonuca ulaşmak için bölgenin alanı $A_a = X_a \times Y_a$ 'yı sabit tutularak ve eşitlik (3.14)'te verilmiş olan $E[D]$ fonksiyonu $Y_a = A_a / X_a$ eşitliğine göre minimize edilmiştir.

$$\text{Min}E[D] = \min 1/3 [X_a + A_a / X_a] \quad (4.15)$$

Eşitlik (4.15)'in X_a 'ya göre türevi alınıp ve elde edilen sonuç eşitlik (4.16)'da 0'a eşitlendiğinde: (4.17) ve (4.18)'de incelenebilecek olan X_a ve Y_a eşitliklerine ulaşılmıştır.

$$1/3 (1 - A_a / X_a^2) = 0 \quad (4.16)$$

$$X_a = \sqrt{A_a} \quad (4.17)$$

$$Y_a = \sqrt{A_a} \quad (4.18)$$

Bu durumda 4.17'de elde ettiğimiz sonuçları eşitlik (4.14)'e uyguladığımızda beklenen yolculuk mesafesini aşağıdaki gibi sonuçlandırabiliriz:

$$E[D] = 2/3 \sqrt{A_a} \quad (4.19)$$

Bu doğrultuda bölgenin optimal şeklinin kare olduğunu ve c katsayısının kare şeklindeki bir alan için 2/3 olduğunu söyleyebiliriz. Beklenen yolculuk uzunluğunu tespit edildikten sonra, beklenen yolculuk zamanının hesaplanmasına geçilebilir.

Hız zaman formülümüzü devreye sokarak mesafeden zamana geçilebilir:

$$D(\text{Uzunluk}) = v(\text{Hız}) \times t(\text{Zaman}) \quad (4.20)$$

Bu formülasyonun kullanılabilmesi için tanımlanması gereken etkin yolculuk zamanları önemlidir. x ve y doğrultusunda araçların farklı eksenlerde farklı hızda ilerleyebileceğini de

göz önüne alarak (sırası ile v_x ve v_y) beklenen yolculuk zamanı şu şekilde elde edilebilir.

$$E[T] = 1/3(X_a/v_x + Y_a/v_y) \quad (4.21)$$

$E[D]$ 'ye benzer şekilde $E[T]$ 'de eşitlik (4.21)'de belirttiğimiz durumda minimize edilir:

$$X_a/v_x = Y_a/v_y = \sqrt{A_a/(v_x \times v_y)} \quad (4.22)$$

Bu analizlerimiz optimal alan (kare) için bize yeterli bilgi verecektir. Analizimizi optimal olmayan alanları da kapsar duruma getirmek için $X_a = \alpha Y_a$ durumunun ele alınmaya çalışılmıştır. (Clapp ve Rodriguez, 1988) Bu durumda ilgilendiğimiz alan herhangi bir dikdörtgensel alanı temsil edebilecektir. ($\alpha = 1$, optimal alanı yani kare şeklindeki servis alanını temsil eder.)

$$A_a = X_a \times Y_a = \alpha Y_a \times Y_a \quad (4.23)$$

Eşitlik (4.22)'den Y_a ve X_a 'yı çektüğümüzde, ilgili değerler aşağıdaki gibi bulunabilir:

$$Y_a = \sqrt{A_a/\alpha} \quad (4.24)$$

$$X_a = \sqrt{\alpha A_a} \quad (4.25)$$

Optimal olmayan alanlar için eşitlik (4.23)'ten elde ettiğimiz sonuçları kullanır ve eşitlik (4.14)'te yerine koyarsak: Eşitlik (4.26)'ya ulaşabiliriz.

$$E[D] = 1/3(\sqrt{A_a/\alpha} + \sqrt{\alpha A_a}) \quad (4.26)$$

Eşitliğimizi düzenlersek:

$$E[D] = (\sqrt{A_a} + \alpha\sqrt{A_a}) / 3\sqrt{\alpha} = \sqrt{A_a}(1 + \alpha) / 3\sqrt{\alpha} \quad (4.27)$$

Eşitlik (4.19)'da $E[D]$ 'nin minimum değerini, optimal alan (kare için) $2/3 \sqrt{A_a}$ olarak bulmuştuk. Genel ifademizde optimal durumdan sapmanın değerini bulmak için (4.27)'den (4.19)'u çıkarırsak:

$$\Delta = \sqrt{A_a}(1 + \alpha) / 3\sqrt{\alpha} - 2/3\sqrt{A_a} = (\sqrt{\alpha} - 1)^2 \sqrt{A_a} / 3\sqrt{\alpha} \quad (4.28)$$

$\alpha = 1$ iken optimal alana (kareye) ulaştığımızın altını tekrar çizdikten sonra, optimal alandan

sapmada örneğin alanımızın boyunun eninin iki katı olduğu $\alpha=2$ durum için:

$$\Delta = (0,414)^2 / (3 \times 1,414) \sqrt{A_a} = 0,0404 \sqrt{A_a} \text{ olduğu sonucuna ulaşabiliriz.}$$

Bu durumda ise beklenen yolculuk mesafesi, ideal durumdan sadece % 6.06 daha fazla olacaktır. $[(0,0404 + 0,6666) / 0,6666]$

Yukarıda sadece dikdörtgen şeklindeki alanlar için ve doğrulu mesafe yaklaşımına göre beklenen yolculuk mesafesini göstermeye çalıştık. Çizelge 4.1’de özel birkaç şekil için (Kare, Eşkenar Dörtgen ve Daire) c değerleri incelenebilir. İlgili değerler, her bir şekil için kuş uçuşu mesafe (öklitsel) ile doğrulu (rectilinear) ulaşım yaklaşımı için ayrı ayrı hesaplanmış olup, bu iki farklı yaklaşım için servis aracının şeklin merkezinde konumlandırıldığı ve alan içinde rasgele bir konumda bulunduğu durum için ele alınmıştır. (Larson ve Odoni, 1981)

Bu doğrultuda her bir özel alan için irdelleyeceğimiz dört farklı durum vardır:

- Öklitsel yolculuk ve servis aracının rasgele konumlandırıldığı.
- Birbirine dik yollarda yolculuk (doğrulu yaklaşım) ve servis aracının rasgele konumlandırıldığı.
- Öklitsel yolculuk ve servis aracının şeklin merkezinde konumlandırıldığı.
- Birbirine dik yollarda yolculuk (doğrulu yaklaşım) ve servis aracının şeklin merkezinde konumlandırıldığı durum.

Her dört vakada da servis talebinin olduğu noktaların servis alanı üzerinde düzgün olarak dağıldığı kabul edilmiştir.

Çizelge 4.1 Ortalama yolculuk zamanlarının tespitinde kullanılabilecek katsayılar (Larson ve Odoni, 1981)

Ortalama Yolculuk Uzunluğu ($\sqrt{A_a}$ cinsinden)	Alanın Şekli	Kare	Mükemmel Eşkenar Dörtgen	Daire
	Metrikse Uzunluk			
Servis Aracı Alan üzerinde rasgele yerleşecek	Öklitsel	0,520	0,520	0,511
	Doğrulu	0,667	0,660	0,650
Servis Aracı Alanın Merkezinde yer aldığı	Öklitsel	0,383	0,383	0,376
	Doğrulu	0,500	0,471	0,479

Çizelge 4.1’den de görülebileceği üzere, $E[D]$ değerlerimiz alanın geometrik şekli ile pek de ilintili değildir. Çizelgemiz incelendiğinde optimal alandan sapmanın ne kadar fazla gerçekleşirse, ortalama yolculuk zamanının o denli arttığına ve servis aracının alanın merkezinde yer aldığı her durumda (kare, eşkenar dörtgen ve daire için) alanın içinde herhangi bir noktada bulunduğu durumdan daha kısa ortalama yolculuk zamanlarına tanık olunmuştur. Açıkçası bu sonuç da zaten beklenmekteydi. Zira, bir önceki kısımda Rassal Kuyruk Medyanı olarak irdelenen konumlandırma ve atama yaklaşımı araçların taleplerin medyanında, bir başka ifade ile taleplerin düzgün dağıldığı bir ortamda servis alanının merkezinde konuşlandırılması hafif trafik durumu için optimal sonucu verdiğini irdelemiştik. Bununla birlikte ilgilendiğimiz şekillerde servis alanımızın kompaktlık (derli topluluk) kıyaslamasını “Daire > Mükemmel Eşkenar Dörtgen > Kare” şeklinde yapabiliriz. Bu doğrultuda araç ister rasgele konumlandırılsın isterse servis alanının merkezinde yer alsın servis alanının kompaktlığı arttıkça c sabiti bir diğer ifade ile beklenen yolculuk zamanı azalacaktır.

Pratik uygulamalarda özel bir geometrik şekle sahip alanlar ile pek de sık karşılaşılmayacağı düşünüldüğünde, yapılan analizleri daha geniş kapsamlı kılmak yararlı olacaktır. Çizelge

4.1’de irdelenen farklı vakalar için katsayılar, yaklaşık sonuçlar olarak şu şartları sağlayan her şekil için uygulanabilir:

- Alanın bir boyutu diğerinden çok büyük olmamalıdır. (4.25’teki 2 kat büyüklükte bile $E[D]$ ’nin sadece %6 artması bize bu analizleri gönül rahatlığı ile pratik hayata aktarmamız konusunda bize cesaret verecektir.)
- Büyük engel ve sınırlamalar alanın belli bir kesiminde yoğunlaşmamalıdır.

Çalışmamızda yukarıdaki iki özelliği sağlayan alanlara ‘oldukça kompakt alanlar’ adı verilmiştir.

Bu bağlamda oldukça kompakt ve dışbükey alanlar ve bağımsız, coğrafyada düzgün dağılmış hizmet talepleri için aşağıdaki eşitlik yazılabilir:

$$E[D] = c \times \sqrt{A_a} \quad (4.29)$$

Bu eşitlikte c sadece alandaki yolculuk mesafesini ele alan metriksel yaklaşım ile servis aracının kabul edilen konumuna bağlı olarak değişen bir değişken olurken, A_a da irdelediğimiz coğrafyanın alanı olacaktır.

Çizelge 4.2 Herhangi bir kompakt alan için ortalama yolculuk zamanlarının tespitinde kullanılacak katsayılar

Ortalama Yolculuk Uzunluğu ($\sqrt{A_a}$ cinsinden)	Metriksel Uzunluk	Yaklaşımızda kullanacağımız c değerleri
Servis Aracı Alan üzerinde rasgele yerleşecek	Öklitsel	0,520
	Dik Yollar	0,667
Servis Aracı Alanın Merkezinde yer aldığına	Öklitsel	0,383
	Dik Yollar	0,500

Eşitlik (4.29)'da yerine koyulabilecek c sabit değerlerinin yer aldığı çizelge yukarıda incelenebilir. Bu çizelge daha önce belirttiğimiz dört ayrı vaka için, her satırın en büyük değerinin Çizelge 4.1'den çekilmesi ile oluşturulmuştur.

Eşitlik (4.28)'in uygulanması için gereken şartlardan biri de yukarıda belirtildiği üzere; alanın bir boyutunun diğerinden çok büyük olmaması idi. Çizelge 4.3'te incelenebilecek olan farklı α değerleri için optimal (ideal) durumdan sapmalar bunun en açık ispatıdır. Dikdörtgen şeklindeki alanımızın uzun kenarının kısa kenarının 1,25 katı olduğu durumda ideal durumdan sapma sadece % 0,62 ve 1,5 katı olduğunda sadece % 2,06 iken, α oranımız 2,5 olduğunda %10,68'lere kadar çıkmakta ve modelimizin güvenilirliğini azaltmaya başlamaktadır. α oranının 7 olduğu durumda ise sapma oranı %50 seviyesini geçmekte ve alanımızın kompaktlıktan daha da uzaklaştığı durumlarda ($\alpha > 7$) optimal durumdan uzaklaşma katlanarak büyümektedir. Çizelge 4.3'te, farklı α değerleri için beklenen yolculuk mesafesinin optimal durumdan yüzde sapmasının yanı sıra, beklenen yolculuk mesafesinin de A alanı cinsinden değeri incelenebilir.

Çizelge 4.3 Farklı α değerleri için beklenen yolculuk mesafesinin optimal durumdan sapması ve beklenen yolculuk mesafesi

A Alanımızın boy / en oranı (α)	Optimal Durumdan Sapma (%)	Beklenen Yolculuk Mesafesi ($\sqrt{A_a}$ cinsinden)
1	0,00	0,667
1,25	0,62	0,671
1,50	2,06	0,680
2,00	6,07	0,707
2,50	10,68	0,738
5	34,16	0,894
7	51,19	1,008
10	73,93	1,160
15	106,56	1,377

Oldukça kompakt, dışbükey alanlar ve bağımsız, coğrafyada düzgün dağılmış hizmet talepleri için yaptığımız tartışmaya geri dönersek, eşitlik (4.28)'den yararlanarak (Çizelge 4.2'deki c değerlerini kullanarak) ve etkin yolculuk hızının kat edilen mesafeden bağımsız olduğu durum altında aşağıda verilen beklenen yolculuk zamanı için yaklaşık formüle ulaşılabilir. Bu formülde x ve y ekseninde aracın sabit v hızı ile ilerlediğini kabul edersek, eşitlik (4.21)'de ele aldığımız her eksen için farklı hız (v_x ve v_y) parametrelerini alınmasına gerek duyulmaz ve eşitlik (4.30)'da incelenebilecek olan beklenen yolculuk zamanına ulaşılabilir.

$$E[T] = (c/v) \times \sqrt{A_a} \quad (4.30)$$

Buraya kadar gerçekleştirdiğimiz analiz ve hesaplamaları sadece tek servis aracı için yaptık.

Analizimizi benzer şekilde birden fazla servis aracı tarafından kontrol edilen alanlar için $E[D]$ ve $E[T]$ 'yi hesaplayarak da genişletebiliriz. (Alanımızın dışbükey ve kompakt olduğu kabulünü çok araçlı durum için de korumamız şartı ile)

Daha önce tek aracın sorumluluk alanı A_a idi. Aynı bölge m adet araç tarafından kapsansa her bir aracın sorumluluk alanı A_a/m olacaktır. Sorumluluk alanı mantığından yola çıkarak:

$$E[D] = c \times \sqrt{A_a/m} \quad (4.31)$$

Sonuç olarak, eşitlik (4.31) yardımı ile beklenen yolculuk mesafesi, eşitlik (4.32) kullanarak da beklenen yolculuk zamanı hesaplanabilir.

$$E[T] = (c/v) \times \sqrt{A_a/m} \quad (4.32)$$

4.3 Daha Gerçekçi Yolculuk Zamanı Modeli

Kısım 4.2'de yaptığımız analizde aracın etkin hızı (en azından belli doğrultularda) sabit olarak alınmıştı. Oysa ki pratik pek çok vakada araçların kat edilmesi arzu edilen mesafeyi sabit bir hızla almadığı, hatta araçların etkin hızlarının kat edilmesi beklenen mesafeye bağlı olduğu, diğer bir ifade ile araçların etkin hızlarının yolculuk mesafesinin bir fonksiyonu olduğu gözlemlenebilir. Genellikle, uzun mesafelerde ortalama hız, kısa mesafeye kıyasla daha yüksektir. Tekrar İstanbul Kenti örneğine döndüğümüzde kısa mesafeler için ara ve tali yollar tercih edilirken, bağıl olarak daha uzun mesafeler için çevre yolları (D100 ve E80 otoyollarının İstanbul Kenti sınırları içerisinde kalan kısımları) tercih edilmektedir. Bu yollar şehrin trafik yükünü dağıtırken, belirli iki nokta arasındaki ortalama yolculuk zamanını da azaltmaktadır. Bu bilgi ve kabuller ışığında $E[T]$ 'yi yolculuk mesafesinin bir fonksiyonu olarak göstermeye devam ederek yolculuk mesafesine bağlı daha gerçekçi etkin hız değerlerini kullanarak daha anlamlı ve pratik hayatta kolaylıkla uygulanabilecek bir model geliştirmeye çalıştık.

Öncelikle, aracın toplam yolculuk mesafesini kat ettiği toplam yolculuk süresi, aracın farklı etkin hız fonksiyonuna istinaden üç kısma ayrılmıştır. Servis aracının bir talebe atanmasından sonra ilk olarak bir ivmelenme periyodu geçireceğini kabul edelim. Araç bu zaman diliminde sabit ivmeli hareket yapar, yani aracın etkin hızı sabit bir şekilde artar. İvmelenme periyodu olarak da adlandırabileceğimiz ilk zaman periyodunda gerekli manevraların yapılmasını, dönüş ve kavşakların kullanılmasını ve ara sokaklarda aracın ilerlemesini göz önüne alabiliriz. İvmelenme periyodunun aracın belirli bir akış hızına ulaşınca dek sürdüğü kabul edilir.

İkinci zaman periyodunu aracın normal şartlar altında, belli bir tempoda (ortalama bir etkin hız ile) akış hızında yolculuk ettiği kısım oluşturur. Aracın akış hızı, ivmelenme periyodundan sonra ulaştığı, aracın hareketinin ikinci kısmını kapsayan, göreceli olarak birim zamanda daha fazla araca hizmet vermek için inşa edilmiş otoyol kısımları, çok şeritli yollar ve çevreyollarını kat ettiğini kabul ettiğimiz sabit bir hızdır. Aracın seyrinin üçüncü ve son kısmı ise, akış hızından yavaşlayarak aracın durmasına kadar geçen kısımdır. Aracın hareketinin son kısmında ilk kısımda yaptığı hareketin simetriğini yapacaktır. Ulaşmak istediği servis noktasına dek ara sokaklarda aracın ilerlemesi, gerekli manevraların yapılmasını bu kısımda gerçekleştirilmektedir.

Yukarıda tanımlamaya çalıştığımız modelimizde, hareketin birinci ve üçüncü kısımlarında aracın hızlanma (a) ve yavaşlama ivmesinin ($-a$) sabit olduğu (kilometre / dakika²) ve ikinci kısımda da aracın akış hızının sabit bir hız olduğu (kilometre / dakika) kabul edilmiştir. Farklı kısımlardaki farklı fonksiyonlar ile gösterilebilecek, araçların kat ettiği mesafeleri zamana bağlı hesaplamak için temel fizik formüllerinden yararlanılmıştır.

İvmeli hareket formülünden:

$$v_c = a \times t \quad (4.33)$$

$$d_c = 1/2 a \times t^2 \quad (4.34)$$

Sabit hızlı hareket formülünden:

$$D = v \times t \quad (4.35)$$

Formül 4.33 ve 4.34'ten, formül (4.36)'ya ulaşılabilir.

$$d_c = v_c^2 / 2a \quad (4.36)$$

d_c , aracın sabit akış hızına ulaşmaya kadar kat ettiği mesafe ve v_c de aracın ulaştığı akış hızı olsun. Akış hızına ulaşmak için d_c ve tekrar aynı ivme ile ($-a$) yavaşlamak ve durmak için yine d_c mesafesine ihtiyacımız olacaktır. Bu doğrultuda, $2d_c$ den daha kısa yolculuk mesafelerinde araç asla akış hızına ulaşamayacaktır.

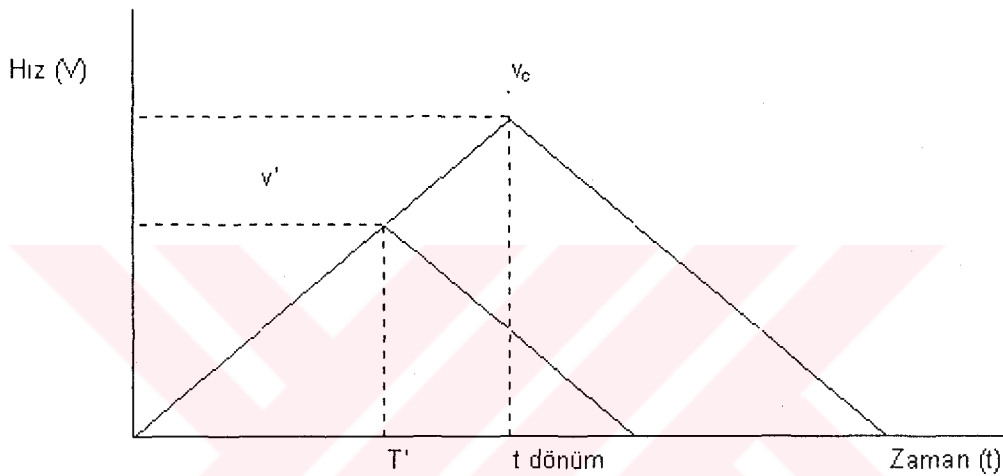
Şekil 4.3'te $2d_c$ den daha kısa mesafeler için hız-zaman grafiği incelenebilir. Bu şekilde iç içe çizilmiş iki grafiğin birinde aracın $2d_c$ lik bir yolu aldığını ele aldık. Buna göre araç 't dönüm' zamanına dek sabit a ivmesi ile hızlanır ve t dönüm anında v_c hızına ulaşır. Araç aynı ivme ile yavaşlaması gerektiğinden bu hızını koruyamaz ve $-a$ ivmesi ile durana dek yavaşlar. Akış

hızına ulaştığı andan durana dek tekrar d_c lik bir yol alacaktır.

Kat edilmesi gereken toplam mesafe $2d_c$ 'den daha kısa bir mesafe olduğunda yukarıda da bahsettiğimiz üzere araç v_c hızına asla ulaşamayacaktır. Bu durumda aracın ulaşabileceği en yüksek hız v' olsun. Eşitlik (4.33)'ten $v' = a \times t'$ olacaktır.

Sadece t' anına dek gerçekleşen ivmeli hareketi şekil 4.3 yardımıyla irdediğimizde eşitlik (4.37) ile gösterilebilecek olan formülasyona ulaşabiliriz.

$$d/2 = 1/2 a \times t'^2 \quad (4.37)$$



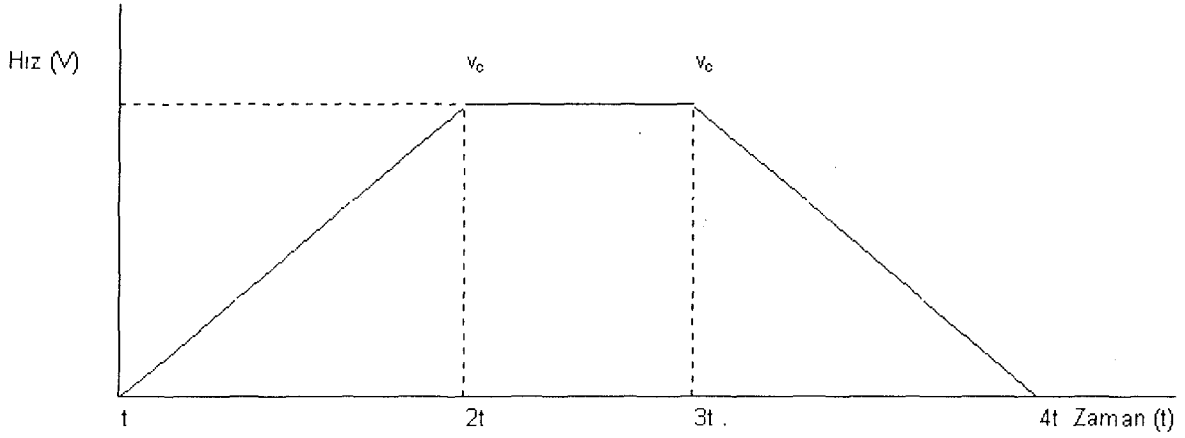
Şekil 4.3 $2d_c$ den daha kısa mesafeler için hız-zaman grafiği

Eşitlik (4.37)'ü düzenler isek t' anını, aşağıdaki eşitlik ile de gösterebiliriz.

$$t' = \sqrt{d/a} \quad (4.38)$$

Şekil 4.4'te $2d_c$ den uzun mesafeler için hız-zaman grafiği incelenebilir. Bu şekilde, t anında hareketine başlar, hareketinin ilk kısmını sabit ivmeli hareket yaparak tamamlayan araç $2t$ anında v_c hızına ulaşır ve herekete başladığı andan akış hızına ulaşana dek d_c lik bir yol kateder. Böylelikle hareketinin ilk kısmını d_c lik bir yol katetmiş olarak ve v_c hızına ulaşmış olarak hareketinin ikinci kısmına geçecektir. Hareketinin ikinci kısmında (toplam yolculuk mesafesinden $(d) 2d_c$ 'lik bir fark kadar) akış hızında ilerler ve şekil 4.4'ten de görülebileceği üzere $2t$ ile $3t$ arasındaki zaman periyodunda sabit hızlı hareket yapar. Araç $3t$ anında aynı ivme ile yavaşlamaya başlar ve durana dek tekrar d_c 'lik bir yol kat eder. Hareketinin üçüncü ve son kısmını böylelikle sonlandırır. Şekilden de gözlenlenebileceği üzere aracın t ve $2t$ zaman diliminde çizdiği yol zaman grafiği ile $3t$ ve $4t$ zaman dilimleri arasında çizdiği yol

zaman grafikleri y eksenine paralel , $x = 5t/2$ doğrusuna göre simetriktir.



Şekil 4.4 $2d_c$ den daha uzun mesafeler için hız-zaman grafiği

Şekil 4.3 ve 4.4 doğrultusunda yaptığımız analizler ışığında; $E[T]$ 'yi yolculuk mesafesinin bir fonksiyonu olarak parçalı şekilde yazabiliriz. Buna göre:

$$E[T|D = d] = 2\sqrt{d/a} \quad d \leq 2d_c \text{ için} \quad (4.39)$$

$$E[T|D = d] = (d - 2d_c)/v_c + 2v_c/a \quad d \geq 2d_c \text{ için} \quad (4.40)$$

Eşitlik (4.40)'ı düzenleyerek aşağıdaki eşitliğe ulaşabiliriz: ($v_c^2 = 2d_c \times a$)

$$E[T|D = d] = d/v_c + v_c/a \quad d \geq 2d_c \text{ için} \quad (4.41)$$

Temel fizik formüllerinden elde ettiğimiz eşitlik (4.39) ve (4.40)'ı pratik hayata taşıyabilmemiz için şüphesiz ki, v_c ve a değerlerine ulaşmamız gerekir. Ne yazık ki, İstanbul Kenti ya da Türkiye'nin herhangi bir kenti için hesaplanmış bir yolculuk zaman verisine ulaşamamıştır. Çalışmamızın bundan sonraki bölümünde, doğrudan ölçüm, tahmin ve literatürde yer alan ve çeşitli yöntemler ile tespit edilmiş sonuçlar kullanılarak, İstanbul Kenti için v_c ve a değerlerine ulaşılmaya çalışılmıştır.

4.4 Literatürde Yer Alan Ortalama Hız ve İvme Değerleri

Ulaşım analizleri için uygun bir tahminde bulunmada, literatürde yer alan bulgulardan doğru bir şekilde yararlanmanın önemi büyüktür. Literatürde özellikle acil servis araçları için (ambulans, arazöz, polis otomobilleri) kaza yerine ulaşma zamanı ile ortaya çıkması muhtemel zarar ve kayıp ile ilgili yaklaşımlar oldukça sık irdelenmiştir. Bu yaklaşımlarda

kaza yerine intikal zamanı, ölçüm ve hesaplamanın yapıldığı bölge, o bölgenin yerleşim alanı olarak temel karakteristiklerine (kent, şehir merkezi ya da daha küçük yerleşim birimi olması) bağlı olacağından yaklaşımımız için birebir yararlanmanın doğru olmadığını bununlar birlikte ancak temel bir kıyas kabul edilebileceği sonucuna ulaşılmıştır.

Yolculuk zamanının özellikle de yanıt zamanının ne denli önemli olduğunu araştırmamızın önceki kısımlarında irdelenmişti, bu kısımda literatür taramamız yolculuk zamanının yolculuk mesafesi ile ne denli ilintili olduğu incelenmeye kaydırılmıştır. Bulduğumuz sonuçlar özetle şu şekildedir:

A.B.D İskan ve Kent Gelişim Departmanının (The Department of Housing and Urban Development) raporuna göre (1977): “Yolculuk zamanlarının (yolculuk mesafelerinden) tahmin edilmesinde pek çok metot kullanılabilir. Tavsiye edilen ve örnek gösterilen metoda göre, itfaiye istasyonu veya hastaneler ile ilgili alanlar içinde yer alan olayın meydana geldiği noktalar arasındaki mesafe odometre (araba ile kat edilen mesafeyi ölçen alet) ile ölçülebilir. Ölçülen bu mesafeler, gerçek yolculuk zamanları ölçülerek ya da ortalama bir yolculuk zamanı kabul edilerek yolculuk zamanlarına dönüştürülebilir.”

Landolfi (1997)’ye göre ise yolculuk zamanı ve mesafesi için yararlanabilecek diğer bir metot, talebin simüle edilmesi ve ilgili zamanın ve mesafenin ölçülmesidir.

Literatürde yer alan yolculuk hızları için bulunan bazı tahmin ve kabuller aşağıdaki gibidir:

- Seul için Başkent Ulaşım Planı Komitesinin bir üyesi olan Keeyeon Hwang’a göre 2000 yılı için ortalama hız 22.9 km/ saat’tir. [2]
- Delhi, Hindistan için Kokaz vd., (2001)’e göre ortalama hız 21,2 km/ saat’tir.
- Auckland, Yeni Zellanda için Hudges (2002)’ye göre tıkalı şehir içi yollar için ortalama hız 7,7 km/saat; maksimum hız 49,3 km/ saat olarak; tıkalı otoyollar için ise ortalama hız 32,9 km/saat; maksimum hız 74,1 km/ saat olarak ölçülmüştür.
- Chicago, A.B.D için Fitzpatrick (1989)’a göre ortalama hız, kısa-ara sokaklar için 30,57 km/ saat (19 mil/ saat), çok şeritli yollar için 57,92 km/ saat (36 mil/ saat)’tir.
- Thompson (1998)’e göre Emmitsburg için ortalama hız 48,28 km/ saat (30 mil/ saat)’tir.
- Washington D.C için Granito ve Dionne (1988)’e göre ortalama hız 32,18 km/ saat (20

mil/ saat)'tir.

- Wilson (1994)'e göre çeşitli araç tipleri için (ambulans, otomobil, kamyon vs.) ortalama hız 27,35 ila 38,62 km/ saat (17 ila 24 mil/saat) arasında değişmektedir.
- Lodi Kenti İtfaiye Teşkilatı (1989)'a göre için ortalama yolculuk hızı 41,83 km/ saat (26 mil/ saat) olmuştur.
- Sybesma (1995)'e göre ortalama yolculuk hızı 48,27 km/ saat (30 mil /saat)'tir.

Walker (1979), çalışmasında yukarıda alıntı yaptığımız araştırmacılardan farklı olarak, A.B.D'de yer alan bazı kentler için v_c değerlerine ek olarak ve a (ivme) değerlerini de irdelenmiştir, bahsi geçen değerleri Çizelge 4.4'te incelenebilir.

Çizelge 4.4 A.B.D'deki bazı itfaiye departmanlarına ait seyir hızı ve ivmeleri (Walker, 1979)

Şehir	v_c (km / saat)	a (km / dakika ²)
New York, N.Y.	62,91	0,77
Trenton, N.Y.	55,99	1,75
Yonkers, N.Y.	54,55	1,38
Denver, Colo	63,07	1,69

Yukarıda verdiğimiz bağımsız ve genel ortalama hız bulgu ve kabullerinin yanı sıra çeşitli yol parça ve karakteristiklerine göre hız modellemeleri yapmak da mümkündür. Çizelge 4.5'te Hoban vd. (1994)'in yol genişliğine göre yolları sınıflandırmasını ve bahsi geçen her sınıf yol için Brabyn (2002)'in önerdiği ortalama hız değerleri bulunabilir. Çizelge 4.5'e göre yolun genişliği arttıkça, km/saat cinsinden ortalama hızın da arttığını gözlemlemek mümkündür.

Çizelge 4.5 Yol karakteristiklerine göre ortalama hız değerleri (Hoban vd., 1994; Brabyn, 2002)

Yol Tipi	Yol Geniřlięi (m)	Ortalama Hız (km/ saat)
Tek řeritli Yol	< 4	30
Orta Seviye Yol	4-5,5	40
Çift řeritli Yol	5,5-9	50
Çift řeritli Geniř Yol	9-12	60
Dört řeritli Yol	>12	80

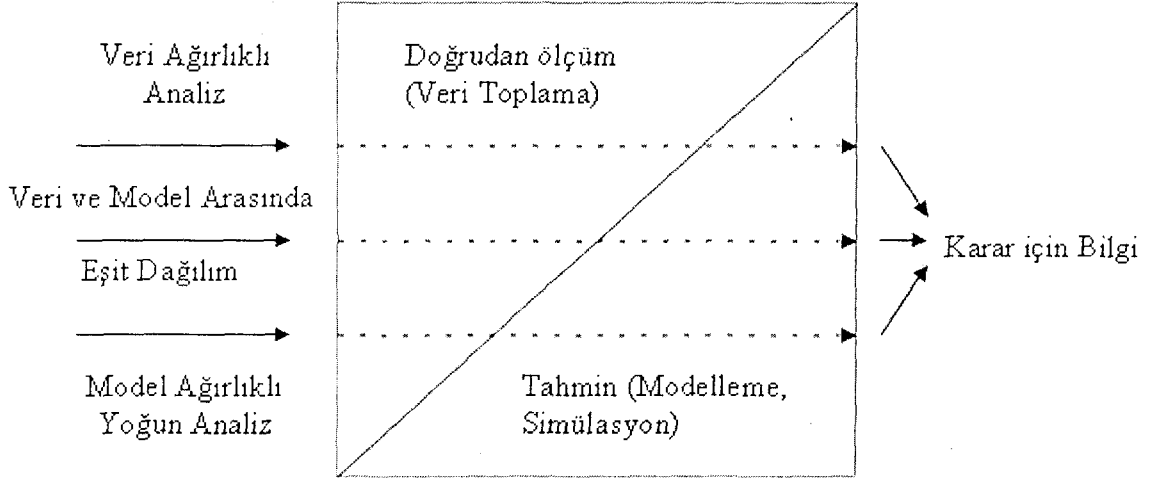
4.5 Yolculuk Zaman Verisi için Doğrudan Ölçüm ve Tahmin

Amerika Birleşik Devletleri İskan ve Kent Gelişim Departmanı raporuna (1977) ve Landolfi (1997)'ye göre ulaşım analizleri, iki temel kaynaktan beslenen veriyi kullanır diyebiliriz. Bu veriler:

- Gerçek trafik ortamında doğrudan toplanan veya ölçülen veri ve
- Bilgisayar simülasyon modelleri veya tahmin teknikleri ile üretilen veridir.

Kaynaktaki farklılıklar, her iki verinin de farklı karakteristiklere sahip olmasına, uygulamaya göre farklı avantaj ve dezavantajları olmasına neden olabilir. Şekil 4.5'te, ulaşım analizlerinin bahsi geçen kaynakları arasındaki ilişki incelenebilir. Şekil 4.5'te de gösterildiği üzere, planlamacı doğrudan ölçüm ile tahminin herhangi bir kombinasyonunu tercih edebilir. (Şekil 4.5'te kalın çapraz çizgi ile gösterilmiştir.) Sadece veri toplamaya itibar eden analizler karenin sağ üst tarafında yer alırken, sadece tahmine itibar eden analizler ise karenin sol alt kısmında yer alır. Veri toplama ve tahminin herhangi bir analizdeki kombinasyonu, analizin amacına, analiz bütçesine, elde edilebilir personel, ekipman ve diğer kısıtlara bağlı olarak belirlenebilir. Bununla beraber iki kaynağın belli bir kombinasyonda kullanımı araştırmanın ihtiyaçlarına göre her bir kaynağın dezavantajlarının minimize edilmesi bakımından değerlidir. Çalışmamızın takip eden kısımlarında yolculuk zamanı verisi için doğrudan

ölçümü ve ilgili teknikler irdelenmiş, araç atama modeli için gerekli olan v_c ve a değerleri tespit edilmeye çalışılmış, ilgili modeli kurmak için ihtiyaç duyduğumuz yolculuk zamanlarını belirleyecek olan parametrelerin tamamlanmasına gayret edilmiştir.



Şekil 4.5 Ulaşım analizlerinde yararlanılan doğrudan ölçüm ve tahmin teknikleri arasındaki ilişki (Turner vd., 1998)

4.5.1 Yolculuk Zamanı Verisi için Doğrudan Ölçüm ve İlgili Teknikler

Birçok veri toplama tekniği, yolculuk zamanlarını ölçmek ya da toplamak adına tercih edilebilir. Bu teknikler, belirlenmiş yol parçaları ve bağlantıları için yolculuk zamanını ve ortalama hızları toplamak için tasarlanmıştır. Bahsi geçen teknikler nokta endeksli hız ölçümlerinden farklı olduğu için elde edilen yolculuk zamanı ve hız verileri anlık hızlardan farklıdır. Şimdi bu teknikleri kısaca gözden geçirirsek:

- Test aracı teknikleri, en sık kullanılan yolculuk zamanı toplama metodudur ve özel olarak veri toplamak amacıyla, trafik akışında yol alan araç ya da araçlardan oluşur. Test aracı içinde yer alan veri toplama sorumlusu aracın hızını daha önce tespit edilmiş sürüş tarzına göre kontrol eder. Test aracındaki bir yolcu, daha önceden tespit edilmiş kontrol noktalarında aracın hızını, yolculuk zamanını ve mesafesini manuel kaydedebilir. Elektronik bir UÖC (Uzaklık Ölçüm Cihazı), cihaza eklenebilir ve bilgisayar ile eşleştirilerek kat edilen mesafe ve hız belli aralıklarla kaydedilebilir. Bir GPS alıcısının bilgisayara bağlanması ile aracın konumu ve hızı bir saniye sıklığında otomatik olarak kayıt edilebilir. (Turner vd., 1998)
- Plaka eşleme teknikleri, belirli noktalarda plaka karakterlerini ve geliş zamanını toplamayı, ardaşık kontrol noktalarında plakaları eşlemeyi ve gelme zamanları

arasındaki farklardan yolculuk zamanlarını hesaplamayı içerir. Bu teknik birkaç farklı yol ile uygulanabilir. Manuel metot, araçların plakasını ses kayıt cihazları ile kaydetmeyi ve daha sonra eldeki veriyi daha sonraki eşleştirmeler için bilgisayar ortamına kopyalanmasını içerir. Plakada imajlarını kameralar ile toplayıp, karakter tanımlama yazılımları ile otomatik olarak bilgisayar ortamına transfer eden metodlar da mevcuttur. (Turner vd., 1998)

- ZUS (Zeki Ulaşım Sistemi) araç teknikleri, trafik akışındaki pasif aygıtlı taşıtları ve uzaktan algılama kapasitesi olan ekipmanları kullanarak yolculuk zamanlarını toplarlar. ZUS araçları, kişisel, ticari ya da toplu taşımacılığa ait araçlar olabilir ve genellikle yolculuk zamanları toplamaktan başka amaçlar için yol alırlar. Bu araçlar, gerçek zamanlı olarak yolculuk zaman verilerini ulaşım idare merkezine raporlarlar. GPS vericileri, mobil telefon cihazları ve araç takip sistemleri bu tekniklerin oldukça sık yararlandığı aparatlardır. (Turner vd., 1998)

Bu üç geleneksel yöntemin yanısıra, gelişmekte olan teknikler de mevcuttur. Bu teknikler indüktans ilmekleri, hareket ağırlıklı istasyonlar veya havaya ilişkin kamera yöntemleri gibi pek çok aparatı, yolculuk zamanlarını ölçmek için kullanır. Bahsi geçen gelişmekte olan teknikler şu an test aşamasındadır ve henüz sahada geniş bir şekilde test edilmemiş ya da kullanılmamıştır. Geleneksel ve gelişmekte olan teknikleri kısaca gözden geçirdikten sonra çalışmamızda yararlandığımız doğrudan ölçüm metodu detayları ile kısım 4.5.2’de bulunabilir.

4.5.2 Modelimiz için v_c ve a Değerlerinin Tespit Edilmesi

Modelimizde baz alacağımız ortalama hız ve ivmenin belirlenmesinde literatürde başka şehirler için tespit edilmiş hız ve yolculuk süresi verilerinden, doğrudan ölçümden ve tahminden yararlanılmıştır. Doğrudan ölçüm için öncelikle araştırmanın yapılacağı coğrafi alan, ilgili yollar ve yol tipleri ile zaman elemanları (ay, haftaiçi günleri ve zaman periyodları) tespit edilmiştir. Bu 3 temel kriterin tespit ederken aslında üç önemli sorunun cevabını almaya çalıştık. (Berry ve Green, 1949; Berry, 1952)

- Yolculuk zamanı verilerini nerede toplayacağız? (Coğrafi Alan)
- Hangi yollar üzerinde yolculuk zaman verisine ulaşmayı hedefliyoruz? (İlgili Yol ve Yol Tipleri)
- Yolculuk zaman verisini ne zaman toplayacağız? (Zaman elemanları)

Eğer çalışmanın hedefi belli sayıda koridor ve yol için yolculuk zaman verisi toplamak ise, yolculuk verisi her bir yol için toplanması en uygun olacakken, araştırmanın coğrafi genişliği tüm şehirs el alanı veya bölgeyi kapsıyor ise örnekleme yöntemi, elimizdeki kaynakların en etkin bir şekilde kullanımı açısından en doğru hamle olacaktır. Modelimizi İstanbul Kentinin tüm şehirs el alanı için kurmayı hedeflediğimiz için yolculuk zaman verilerini de İstanbul Kentini doğru temsil edebilecek olan 35 nokta arasında toplamaya, bir örnekleme yapmaya karar verdik. Bu noktaların kentimizin her iki yakasında olmasına dikkat ettik, seçtiğimiz noktalardan 17'si Avrupa Yakasında iken (% 49), 18 adeti Anadolu Yakasında (%51) yer almaktadır.

İlk soru ile birlikte ikinci sorumuzun da cevabını aramaya başlarsak, aynı coğrafi alanda durumunda değindiğimiz üzere gibi ilgili yol ve tipleri de çalışma hedeflerimiz ile uyumlu olmalıdır. İstanbul il sınırları içerisinde 38 ana arter, bağlantı yolu ve köprü lü kavşak yer almaktadır ve bu yollar TCK 1. Bölge ve 17. Bölge Müdürlüklerinin sorumluluğ u altındadır. [22] Verilerimizin toplanacağı ilgili yolları seçerken İstanbul Kentinin trafik yoğunluğ unda en çok payı alan, en çok kullanılan yolları almayı hedefledik. Bu yollara örnek vermemiz gerekirse, Barbaros Bulvarı, Bağdat Caddesi, her iki yakadaki sahil yolları, D-100 ve TEM çevreyolları ve şehrimizin sembolü Boğaziçi Köprüsü doğrudan ölçümümüzde yer alan kapasiteler (yollar) olarak gösterilebilir [22]. İlgili 35 nokta arasında yer alan ve belli başlı yolları da yukarıda verdiğimiz yaklaşık 3.203 kilometrelik yol üzerinde analizimizi gerçekleştirmeye çalıştık. Bununla birlikte ilgili yolların HCM (1994)'te yer alan şehri içi yolların fonksiyonel sınıflandırmalarına da sadık kalınmasına da gayret ettik.

Yolculuk zaman verilerinin toplanılmasında zaman elemanları da (yılın ayları, haftanın günleri ve gün içindeki zaman dilimleri) ciddi anlamda araştırmamızın sonuçlarını etkileyebilecek parametrelendir. Aşağıda ilgili zaman parametrelerin, araştırmamız için nasıl düzenlendiğ i açıklanmaya çalışılmıştır.

- Yılın ayları, zaman elemanları içerisinde ilk önce karar verilmesi gereken parametredir. Yolculuk zaman verisi, “tipik” veya “ortalama” yıllık koşulları temsil edeceğ i için, tipik ya da ortalama trafik hacminin gözlemlendiğ i aylarda toplanması önemlidir. SDHPT (1989)'a göre Houston, Texas için ölçülen değ erler doğ rultusunda, ortalama trafik yoğunluğ unun %2'lik aralığ ı içinde bulunan aylar mart, nisan, mayıs, haziran, eylül, kasım ve aralık olarak belirtilmiş ve veri toplanılması için aday aylar olarak önerilmiştir. Genel olarak ise güz (eylül, ekim, kasım) ve bahar ayları (mart, nisan, mayıs) veri toplamak için en uygun zamanlar kabul edilir. İstanbul Kentimiz

için ölçümümüzü, bu iki verinin ışığında nisan ayının ilk haftasında gerçekleştirdik. Nisan ayında hava koşulları tüm yılı temsil edebilecek düzeyde olması, okulların ve üniversitelerin açık olması, yaz tatilinde yer almaması ve trafik koşullarını etkileyecek bölgesel festival ve özel olayların olmaması sebebi ile bu ayın ilk haftasında topladığımız verilerin oldukça sağlıklı olduğu varsayılabilir.

- Haftanın günleri, ele aldığımız bir diğer zaman parametresiydi. Yolculuk zaman verisinin ortalama bir günü temsil etmesi gerektiği için genellikle sadece haftaiçi günlerinin analizlerde yer alması önerilir. Çeşitli sebeplerden ötürü, beş haftaiçi günü için ölçümün yapılamadığı durumlarda ise orta haftaiçi günlerine (Salı, Çarşamba, Perşembe) konsantre olunması tavsiye edilebilir (SDHPT, 1989). İstanbul Kenti için ilgili ölçümlerimizi, 31 Mart 2003 Pazartesi gününden 4 Nisan 2003 Cuma gününe kadar, beş haftaiçi gün için topladık. Bu günleri seçerken, resmi tatillere (milli ve dini bayramlar), özel günlere (sevgililer günü, İstanbul'un kurtuluşu gibi), zaman değişikliklerinin ertesi günlere (saatlerin ileri-geri alınmasını takip eden günler, mesai saatlerinde değişiklik yapıldığı günler) ve özel olayların olduğu (milli maç, önemli spor olayları, festivaller) günlere denk gelmemesine özen gösterildi.
- Gün içerisinde trafik yoğunluğunda değişim küçük-büyük bütün kentlerde gözlemlenen bir olgudur. Özellikle İstanbul gibi metropollerde gün içinde gözlemlenebilecek bu fark daha açıktır. Trafik yoğunluğu açısından genellikle 3 zaman dilimi kabul edilir: Sabah Yoğunluğu (6:00 -9:00), Sakin Dilim (10:00-11:00, 13:00-15:00, ve 19:00'dan sonra) Akşam Yoğunluğu (16:00-19:00). İstanbul Kenti için akşam yoğunluğu saatini 20:00'ye kadar uzatmak mantıklı olacaktır, ayrıca 11:00 ile 13:00 arasını öğlen tatili ve yemeği yoğunluğuna da ayırabiliriz. Kentimiz için veri toplama sabah ve akşam yoğunluğu sırasında gerçekleştirilmiştir. Modelimizi tüm iş saatlerini kapsamaları için ilgili ek verileri Teşvikiye Merkez Taksi çalışanlarından elde edilen sezgisel ve tecrübeye dayanan bilgiler ile şahsi tecrübe ve ölçümlerimizden elde etmeye çalıştık.

Ölçümlerimiz sonrasında, günde 3.203 kilometre, beş gün sonrasında toplam 16.015 kilometre yol, toplam 353 saat 45 dakikada kat edilmiştir. Buna göre ortalama hız, eşitlik (4.20)'den 45,27 km / saat bulunmuştur. Ek 1'de ölçümün yapıldığı herbir rotanın adını, kat edilen günlük toplam mesafeyi (gidiş ve dönüş olmak üzere), toplam taşıma zamanını ve rota başına ortalama hız ölçümleri verilmiştir.

Bununla birlikte elde ettiğimiz ortalama hız verisini, kısım 4.3'te tanımladığımız modele şahsi tecrübeler ve profesyonel sürücülerden edindiğimiz sonuçlar ile harmanlanarak uygulamaya çalışılmıştır. Kısa mesafeler için örnek rota olarak Maçka ile Levent semtleri arası, uzun mesafeler için örnek rota olarak ise Maçka ile Maltepe arası seçilmiştir. Böylelikle kısa mesafeler için aynı yakada yer alan iki nokta ele alınırken, uzun mesafeler için Boğaz Köprülerinden birinin kullanılması gereken, İstanbul Kentinin farklı iki yakasında yer alan iki nokta ele alınmıştır. Tercihlerimizin sebebi gündelik hayatımızda sıklıkla kat ettiğimiz iki mesafe üzerine analizimizin daha doğru ve sağlıklı yapılabileceğinin öngörülmesidir. Bu bağlamda kısa mesafeler için örnek mesafe 6,66 kilometre, uzun mesafeler için ise 23,33 kilometre olarak ele alınmıştır. Şahsi bir araç ile yine Nisan 2003'te kat edilen mesafeler sonrası 6,66 kilometre için ortalama zaman yaklaşık 20 dakika iken, 23,33 kilometre için ortalama zaman yaklaşık olarak 45 dakika olarak ölçülmüştür. Bahsi geçen verileri Şekil 4.3'e uyguladığımızda $t_{dönüm} = 10$ dakika ($20 \text{ dakika} / 2 = 10 \text{ dakika}$) olarak bulunabilir. Bununla birlikte $t_{dönüm}$ anında kat edilmiş olan toplam mesafe ise $6,66/2 = 3,33$ kilometre olacaktır. Modelimiz için ivme değerinin hesaplanmasında eşitlik (4.38)'den yararlanılmıştır. Eşitlik (4.38)'e zaman verisini (10 dakika), mesafe verisi olarak 3,33 kilometreyi koyup ivmeyi ilgili eşitlikten çektiğimizde, modelimiz için ivme değeri, $240 \text{ kilometre} / \text{saat}^2$ ($0,067 \text{ kilometre} / \text{dakika}^2$) olarak bulunmuştur.

Modelimiz için diğer bir kritik parametre v_c değeridir. Yine Şekil 4.3'e döndüğümüzde, araç ilk 10 dakikada, sabit ivmeli hareket etmiş ve bu esnada 3,33 km yol almıştır. Daha sonraki zaman aralığında ($2t$ ile $3t$ arasında) sabit hızlı (v_c hızı ile) hareket etmiştir. Son zaman aralığında ise araç sabit ivme ile yavaşmış ve yine 10 dakika içerisinde 3,33 km yol almıştır. Eşitlik (4.41)'i göz önüne alarak, yukarıda ele aldığımız örnek uzak mesafeyi (23,33 km) kullanarak v_c değeri elde edilmeye çalışılmıştır.

$$T_{yolculuk} = T_{hizlanma} + T_{sabit} + T_{yavasama} \quad (4.42)$$

$$45 = 10 + \left(\frac{23,33 - 6,66}{v_c} \right) \times 60 + 10 \quad (4.43)$$

Eşitlik (4.42)'den v_c değerini çektiğimizde ilgilendiğimiz parametremiz $40 \text{ km} / \text{saat}$ olarak bulunabilir. Yine ilk 10 dakikadaki ivmeli hareketten aracın ivmesi $240 \text{ km} / \text{saat}^2$ ($0,067 \text{ km} / \text{dakika}^2$) olarak bulunmuştur. Elimizde ilgili tüm parametreler bulunduğu için rahatlıkla bu parametreleri (4.39) ve (4.41) numaralı formüllere dakika cinsinden uyguladığımızda eşitlik (4.44) ve (4.45)'e ulaşabiliriz.

$$E[T|D = d] = 7,746\sqrt{d} \quad d \leq 6,66 \text{ için} \quad (4.44)$$

$$E[T|D = d] = 1,5d + 10,1 \quad d \geq 6,66 \text{ için} \quad (4.45)$$

Eşitlik (4.44) ve (4.45) kullanılarak farklı mesafeler için toplam yolculuk zamanlarının hesaplanmış örnekleri Çizelge 4.6’da irdelenebilir.

Çizelge 4.6 Belirli mesafeler için modelimizdeki parametrelere göre hesaplanan toplam yolculuk zamanları

Mesafe (Kilometre)	Toplam Yolculuk Zamanı (Dakika)
3	13,42
5	17,32
6,66	20,00
10	25,10
15	32,60
20	40,10
30	50,10
40	71,10

Modelimizde yer alan sayısal verilerin büyük çoğunluğu sabah ve akşam trafik yoğunluğunda toplanılmış, modelimizi bu veriler ışığında günlük tecrübeler ve tahminler ile biçimlendirilerek kurulmuştur. Modelimizin tutarlılığını ve uygulanabilirliğini irdellemek için Çizelge 4.6’te verdiğimiz sonuçlarımızı Amerika Birleşik Devletleri’nin (A.B.D) ulaşım karakteristikleri ile kıyaslamayı uygun bulduk. Çizelge 4.7’de A.B.D’nin en büyük üç kentinde işyerine ulaşım karakteristikleri (toplam yolculuk eden çalışan sayısı, ulaşım şekli ve süresi) ve Çizelge 4.8’de de tüm A.B.D için ilgili karakteristikleri incelenebilir. İstanbul

Büyükşehir Belediyesi kayıtlarına göre, İstanbul Kentinde 2000 yılı temmuz ayı itibari ile 2.854.425 sürücü belgesi ve 2.166.070 adet motorlu araç bulunduğu göz önüne alındığında özellikle Çizelge 4.7’de yer alan veriler kıyas için oldukça anlamlı olacaktır.

Çizelge 4.7 A.B.D’nin en büyük üç kenti için işyerine ulaşım karakteristikleri [11]

Şehir, Eyalet	16 yaşın üstünde toplam çalışan sayısı	Ulaşım Şekli (%)				İşyerine Ortalama Ulaşım Süresi (Dakika)
		Kişisel Araç	Şirket Araç	Toplu Taşımacılık	Diğerleri	
New York, NY	3,183,088	24,0	8,5	53,4	14,0	36,5
Los Angeles, CA	1,629,096	65,2	15,4	10,5	8,9	26,5
Chicago, IL	1,181,677	46,3	14,8	29,7	9,2	31,5

A.B.D’de 1990 itibari ile 16 yaş ve üzerinde toplam 115,070,274 çalışan vardır. Toplam çalışanların yaklaşık olarak % 3’ü (3,406,025 kişi) evinde çalışırken toplam 111,664,249 kişi, çalışan nüfusun %97’si, işyerine ulaşmak için zamanının belli bir kısmını ulaşımaya ayırmaktadır. Çizelge 4.8’de işyerine ulaşım zamanı dağılımına göre A.B.D için işyerine ulaşım zamanının dağılımı incelenebilir.

Çizelge 4.8 A.B.D için işyerine ulaşım karakteristikleri [11]

İşyerine Ortalama Ulaşım Süresi (Dakika)	Yüzde (%)
5 Dakikadan Daha Az	3,9
5 ila 9 dakika arası	12,5
10 ila 14 dakika arası	16,1
15 ila 19 dakika arası	17,0
20 ila 24 dakika arası	14,5
25 ila 29 dakika arası	5,5
30 ila 34 dakika arası	12,8
35 ila 39 dakika arası	2,4
40 ila 44 dakika arası	2,8
45 ila 59 dakika arası	6,4
60 ila 89 dakika arası	4,5
90 ve daha uzun	1,6

Çizelge 4.8'e göre A.B.D vatandaşlarının % 69,5'i iş yerlerine 30 dakikadan daha kısa bir zamanda ulaşmaktadır. A.B.D için işyerine ortalama ulaşım zamanının 22.4 dakika olduğunu göz önüne aldığımızda ve İstanbul Kentimiz için yaklaşımımızın A.B.D ortalamasında daha çok en büyük üç kentinin ortalamasına yakın olduğunu söylemek güç değildir. Çizelge 4.6'te elde ettiğimiz sonuçları tekrar göz önüne aldığımızda ve Çizelge 4.7 ve 4.8 ile karşılaştırdığımızda modelimizin gerçek hayat ile uyumlu olduğunu bir kere daha

gözlemlemiş olduk.

İstanbul Kentine özgü olarak ulaşmaya çalıştığımız, yolculuk modelimizin en önemli iki parametresi olan akış hızı ile ivme formülasyon ve değerlerinden çalışmamızın takip eden kısmında yararlanılmış, komple bir atama sistemi ele alınmış ve ilgili atama sistemi, simülasyon modeli ile irdelenmiştir.



5. DİNAMİK TALEPLER İÇİN ARAÇ ATAMA YAKLAŞIMI ve SİMÜLASYON UYGULAMASI

Çalışmamızın bu bölümünde, dinamik araç rotalama problemi için gerçek hayatta karşımıza çıkabilecek bir vaka ele alınıp, vakamızı literatürde yer alan farklı yaklaşımlar ve dinamik talepler için geliştirdiğimiz araç atama yaklaşımı ile tek tek analiz edip, geliştirdiğimiz alternatif yaklaşımın performansını, alternatif politikaların performansı ile kıyaslanmaya çalışılmıştır. Öncelikle sistem kavramına ve model ile çalışmanın avantajlarına değinilip, önceki bölümlerde yapılan analiz ve ulaşılan bulgular ışığında simülasyon modelini sağlam temeller üzerine oturtmaya gayret gösterilmiştir. Bu bölümde amacımız problemin doğru formüle edilip, amaç ve proje planı yapıldıktan sonra bir simülasyon modeli yardımı ile farklı yaklaşımların simülasyon sonuçlarını kıyaslamak, elde edilen simülasyon çıktılarından da yararlanarak seçilen araç atama yaklaşımı ile optimal büyüklükte bir araç filosu ile müşteri taleplerine yanıt verilebilecek komple bir sistem oluşturmaktır.

5.1 Sistem Kavramı ve Simülasyon

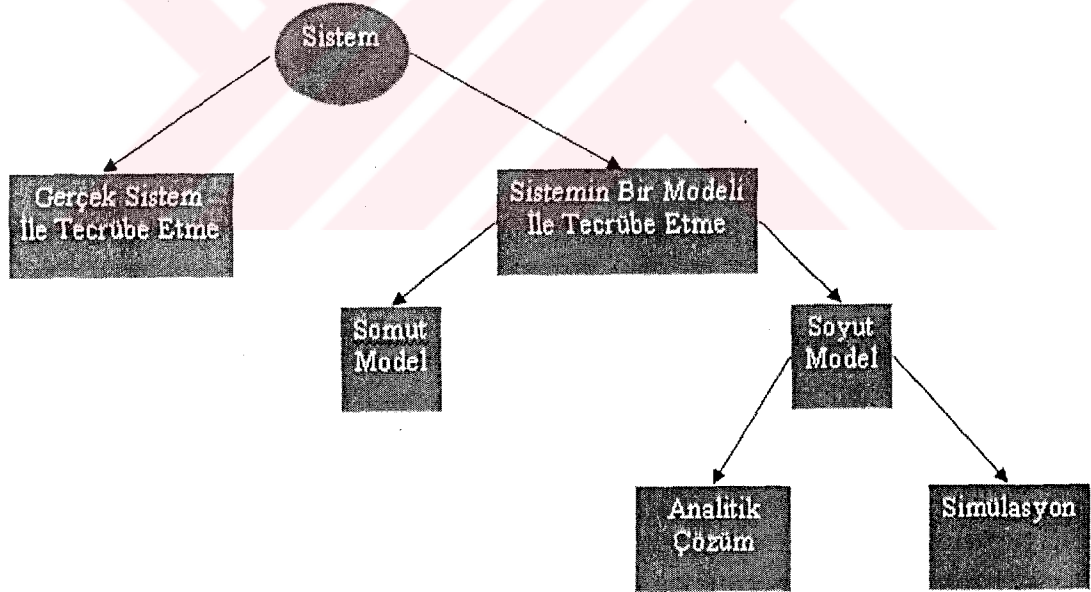
Erkut (1995a)'e göre sistem kavramı, günümüzde hem akademik alanda hem gündelik alanda oldukça sık kullanılan bir sözcüktür. Çalışmamızın bu kısmında, öncelikli olarak sistem kavramı incelenip, sistem üzerine çalışmayı mümkün kılan değişik yollardan biri olan sistemin simülasyon modeli ile irdelenmesine değinilmiştir.

5.1.1 Sistem Kavramı

Simülasyon kavramını, öncelikle sistem kavramını tanımlayarak netleştirmek çok daha sağlıklı bir yaklaşım olacaktır. Erkut (1995a)'e göre sistem, fiziksel ya da fiziksel olmayan, kendi aralarında ilişki ve etkileşim bulunan, bir veya daha fazla amaca yönelik öğelerin bütünüdür. Bu bağlamda İstanbul Kentimiz de, kentimizdeki taşımacılık ağı da, taşımacılığın gerçekleştirildiği araç da, hatta birey olarak kendimiz de bir sistemdir.

Günümüzde, artan ve farklılaşan müşteri talepleri, sürekli değişen teknoloji ve gittikçe sertleşen rekabet, firmaları ürettikleri ürünü veya verdikleri hizmeti daha hızlı, daha etkili, daha kaliteli ve daha az maliyet ile nasıl gerçekleştirilebileceğini araştırmaya zorlamaktadır. Bütün bu sorulara cevap verebilmenin yolu, sistemde hızlı ve yerinde değişiklikler yapmaktan geçmektedir. Sistemin kendisini ve sistemin öğeleri arasındaki ilişkiyi anlamak, alternatif politikalar ile sistemin yeniden nasıl tanımlanabileceğini tartışmak, bu politikalar altında sistemin nasıl davranacağını tahmin etmek için sistemi öğeleri ile birlikte incelemek şarttır.

Şekil 5.1’de bir sistemi incelemenin değişik yolları irdelenebilir. Şüphesiz ki, bir sistemi incelemek için sistemin kendisi ile deney yapmak mümkündür. Ancak bu, her zaman izlenebilecek bir yol değildir. Gerçek sistem ile deneyler yapmak yerine, sistemin bir temsili (modeli) ile çalışmak; gerçek sistemin fiziksel olarak var olmaması (örneğin tasarım aşamasında olması) veya sistemin kendisi ile deney yapmanın uygulanamaz ya da bu uygulamanın çok yüksek maliyetli olması sebebi ile tercih edilir. Bahsi geçen nedenlerden ötürü, sistemin incelenmesinin sistemin fiziksel ya da soyut bir modeli ile yapılması tercih edilebilir. İster fiziksel olsun, isterse soyut olsun; iyi bir model, gerçekçilik ile basitlik arasında doğru noktada yer alan, gerçek sistemi temsil edebilirken üzerinde çalışılabilecek kadar da yalın karakteristikte olmalıdır. Soyut (Matematiksel) modeller, sistemi temsil etmek için sembolik notasyon ve matematiksel denklemler kullanır. Soyut modeller, çözücü (prescriptive)-tasvir edici (descriptive), analitik-nümerik, stokastik-deterministik, statik-dinamik, kesikli-sürekli olarak sınıflandırılabilir. Bir soyut model olan simülasyon modelinin karakteristikleri, güçlü ve zayıf yönlerinin değerlendirilmesi ve sonuç olarak sistem analizi için neden simülasyon modelini seçildiği takip eden kısımda ele alınmıştır [28].



Şekil 5.1 Bir sistemi incelemenin değişik yolları

5.1.2 Sistem Simülasyon Modeli

Şekil 5.1’de de görebileceğimiz üzere, sistem üzerinde çalışmanın bir yolu da sistemin simüle edilmesidir. Sistem simülasyonu, Schriber (1987) tarafından gerçek bir sistemin mantıksal veya matematiksel modelini tasarlama ve daha sonra model kullanılarak, bilgisayar tabanlı

denemeler ile gerçek sistemin davranışını tanımlama, açıklama ve tahmin etme prosesi olarak tanımlanmıştır. Simülasyonun kökeni, istatistiksel örnekleme teorisine (örneğin rasgele sayılar ve rasgele örnekleme) dek uzanır, İkinci Dünya Savaşı sırasında Monte Carlo Simülasyonu ortaya çıkmış olup, simülasyonun modern uygulamaları İkinci Dünya Savaşını takip eden yıllarda gözlenmeye başlanmıştır. Sistem davranışlarını tanımlamada ve tahmin etmede kullanılan yöntemler arasında simülasyon, 1945-1970 yılları arasında en son baş vurulan teknik olarak değerlendirilirken, Rasmussen ve George (1978)'e göre 5. sırada, Thomas ve Decosta (1979)'a göre 2. sırada, Shannon vd. (1980)'e göre 2. sırada ve Harpell vd. (1989)'a göre 2. sırada yer almıştır. Simülasyon, son yıllarda eğitimden eğlenceye, animasyondan taşımacılık ve üretime kadar pek çok dalda oldukça yaygın bir kullanım alanı bulmuştur. Önceleri sadece büyük bütçeli projelerde kendine yer bulabilen simülasyon tekniği, teknolojinin gelişmesi, ucuzlaması ve yaygınlaşması ile gittikçe daha fazla tercih edilen bir sistem inceleme tekniği olmuştur. Bu trendin devam edeceği göz önüne alınarak, simülasyonun kullanım alanlarının ve tercih edildiği proje sayısının artacağı ön görülebilir [28].

Simülasyonun karakteristikleri, soyut (matematiksel), nümerik, tasvir edici, deterministik / stokastik, statik / dinamik ve kesikli / sürekli olarak sıralanabilir. Simülasyon kullanılarak, önce değişik koşullar ya da değişik karar politikaları altında sistemin davranışı gözlemlenebilmekte, daha sonra simülasyondan elde edilen bilgiler ışığında sistemin tasarımı veya en uygun karar kuralının seçimi mümkün olmaktadır. Ayrıca simülasyon yardımıyla yapısal değişiklikler karşısında sistemin tepkisini de değerlendirmek mümkündür.

5.2 Dinamik Bir Gezgin Tamirci Probleminin Sistem Perspektifinden İncelenmesi

Bu kısımda, dinamik gezgin tamirci problemine örnek teşkil edebilecek bir vaka ele alınmış ve ilgili vaka problem olarak tanımlanmış, ilgili kavramsal ve soyut model yardımı ile problem üzerinde yaklaşımlarımız geliştirilmeye çalışılmıştır. Kurmuş olduğumuz model ile, gerçek hayatta karşımıza çıkabilecek temsili bir sistemi tasvir etmeye çalışılmış olup, yararlandığımız değişkenler ile modelimizin kolaylıkla başka gerçek sistemleri temsil edebilir hale gelmesi öncelikli hedefimiz olmuştur.

Özel olarak isimlendirmemiz gerekirse problemimizi, evlerimiz ve ofislerimizin ayrılmaz parçaları haline gelen bilgisayar sistemleri ve ilgili bileşenlerinin tamirlerini gerçekleştiren bir tamir ve bakım şirketinin (PCServis) verdiği hizmet oluşturmaktadır. Sistemin amacı, eldeki kapasiteden en uygun şekilde yararlanarak, müşteri taleplerine en iyi hizmeti vermektir.

5.2.1 Problemin Tanımı

Çeşitli coğrafi, fiziksel ve kavramsal ağların bakımı ve onarımı gündeme getiren ve kısım 3.1.2’de detayları ile irdelemiş olduğumuz DGTP dahilinde, coğrafi bir ağ üzerinde hizmet verme çalışan PCServis Şirketi, kurulduğu nokta olan Maçka-Şişli Senti ve çevresindeki müşterilere odaklandıktan sonra, zamanla büyüyerek İstanbul Kenti’nin her iki yakasına da hizmet verir hale gelmiştir. Müşteri portföyü ve müşterilerinin konumu göz önüne alındığında ise PCServis Şirketinin konumlandığı nokta merkez olmak üzere 1,600 km²’lik yaklaşık olarak kare şeklinde bir alana hizmet verdiği gözlemlenmiştir. Servis için gelen taleplerin mesai saatleri içine homojen olarak dağıldığı, öğleden önce ile öğleden sonra arasında taleplerin dağılımında herhangi bir fark olmadığı ve sisteme saatte ortalama 3 talep girdiği gözlenmiştir. Taleplerin geldiği noktalar, servis alanı içinde özellikle müşteri portföyünün daha yoğun bir şekilde yer aldığı şirket merkezinin etrafında yoğunlaştığı gözlemlenmiştir. Talepler birbirinin aynısı olan 5 araçtan oluşan bir tamirci filosu ile yanıtlanmaya çalışılmaktadır. Araçlar, şirket sekreteri tarafından içgüdüsel olarak atanmakta ve araçların konumu zaman zaman sürücüsü telefon ile aranarak zaman zaman da CBS sisteminden öğrenilmektedir. Müşteri talepleri sabah 9’dan akşam 17’ye kadar 8 saatlik bir süre dahilinde sisteme alınmakta, 17’den sonra sisteme talep kabul edilmemektedir. Mobil tamirciler, İstanbul şehir trafiği içinde ilerlemekte ve sabah 9’dan sisteme girmiş son talebe cevap verilene dek çalışmakta, akşam 17’den sonraki hizmetleri için fazla mesai ücreti almaktadırlar. Yapılan çalışmalarda müşterinin sorununu çözmek için ortalama 30 dakikalık bir servis zamanına ihtiyaç olduğu ancak bu servis zamanının standart sapmasının da 10 dakika olduğu görülmüştür. PCServis Şirketi idarecileri ve ortakları, şirketin karlılığını maksimize edecek, müşteri beklentileri cevap verecek bir tamirci (araç) atama sistemi kurulmasına ve bu doğrultuda şirketin araçlarını hangi politika ile atayacağına, müşterileri en iyi hizmeti verirken malitayetlerini de minimumda tutacak toplam tamirci sayısına ulaşmayı hedeflemektedirler.

5.2.2 Probleme Uygun Soyut Model Seçimi

Kısım 5.1’de “Sistem ve Simülasyon” kavramları üzerinde tartışmalar gerçekleştirmiş ve bir sistemi irdelemenin pek çok yolu olduğu ve simülasyonun, her ne kadar tercih edilme oranı en çok artan teknik olsa da, sistemi inceleme yollardan biri olduğunun altını çizmiştik. Problem tanımımızda verilen detaylar göz önüne alındığında, problemimiz için komple bir matematiksel formülasyonun yer almadığını ve analitik çözümün geliştirilemeyeceğini söyleyebiliriz. Literatürde yer alan ve çalışmamızın ikinci kısmında değindiğimiz bazı araştırmalarda, gezgin tamirci problemindeki detayları basitleştirerek ya da bir kısım kısıtları

yok kabul ederek bazı analitik sonuçlara ve formülasyonlara ulaşıldığı irdelenmişti. Bu bağlamda, problemimizdeki detay ve kısıt seviyesini tanımlandığı şekilde kompleks tutmaya gayret ettiğimizde soyut model olarak simülasyonu seçmemizin bizi daha kolay bir sonuca götüreceğini öngörebiliriz. Soyut model olarak simülasyon modelini seçmemizin sebepleri, simülasyon tekniğinin avantajları da göz önüne alınarak aşağıdaki gibi sıralanabilir: (Pedgen vd., 1995)

- “Neden?” sorusuna yanıt verebildiği,
- Alternatifleri değerlendirmeye ve koşul analizi (what if?) yapmaya olanak sağladığı,
- Problemle ilgili teşhiste bulunmaya ve kısıtları belirlemeye yardımcı olduğu,
- Problem çözmeye sistem yaklaşımı getirilebildiği,
- Daha az kabule gereksinim duyduğu,
- Rastgeleliğe ve belirsizliğe cevap verebildiği,
- Dinamik bir davranış sergileyebildiği,
- Esnek ve kolaylıkla değiştirilebilir olduğu,
- Güvenilir ve sonuçlarını izah etmenin daha kolay olduğu.

Bütün bu sebeplere ek olarak simülasyon, sistem değişkenleri arasındaki çok sayıda karmaşık etkileşimi, analiz etme ve değerlendirme olanağı verir ve sistemin başarısını etkileyen çevre koşullarını göz önünde bulundurur. Sistem başarısının bu şekilde değerlendirilmesi, başka herhangi bir yöntemle gösterilemeyen değişkenler arası etkileşimlerin hangilerinin sistem başarısını tayin etmede etkili olduğunu belirlememizi sağlar. Simülasyon ile, olabilecek yararlı yapısal değişikliklerin ortaya çıkarılması ve bunların mevcut sisteme dahil edilmesinin yanında, değişen sistemin işleminde çıkabilecek problemlerin tahmin edilmesi ve bunların önlenmesi sağlanabilir. Tezimizin amacının dinamik araç atama problemine dinamik bir yaklaşım getirmek olduğu göz önüne alındığında simülasyon analizi ile önerilen politika ve kararların doğruluğunu test etmenin, değerlendirmenin mümkün olduğu, alternatif yaklaşımlar ile sağlıklı kıyaslamalar yapılmasına olanak verdiği öngörülmüştür.

Simülasyon tekniğinin yukarıda saydığımız avantajları ve sunduğu olanaklarının yanında, Pegden vd. (1995)’te sıralanan dezavantajları arasında modelimiz açısından en büyük dezavantajı, çözüm üretmesi yerine sistemin işleyişini vermesi ve kendi başına optimal bir

sonuç vermemesi olarak gösterilebilir. Yukarıda saydığımız bütün avantajları ve olanakları doğrultusunda simülasyon tekniği araç (tamirci) atama problemimizin incelenmesinde tercih edilmiştir. Tekniğin eksikliklerini ise simülasyon çalışmaları sonrası ve simülasyon sonucu elde ettiğimiz veriler ile yapacağımız maliyet-fayda analizi ile kapatılmaya çalışılmış ve maliyet fayda analizi ile şirketin karlılığını maksimize edecek seçilen araç atama yaklaşımın en başarılı uygulandığı araç sayısına ulaşmak, bekleme zamanı ile kapasite arasında doğru bir tercih yapmak hedeflenmiştir.

5.2.3 Simülasyon Dilinin Seçimi

Simülasyon çalışması, sistemin yapay bir performansının oluşturulması ile gerçekleştirilir. Şüphesiz ki, ilgili performansın geliştirilmesinde el ile simülasyon yapmak yerine, bilgisayarlar tercih edilen platformlardır. Banks vd. (1996)'ya göre bilgisayar programlama dilleri kompleks gerçek dünya sistemlerinin simülasyonlarının geliştirilmesini ve çalıştırılmasını büyük ölçüde kolaylaştıran unsurlardır. Kompleks bir simülasyon modeli için bilgisayar kodları yazmak genellikle zor ve emek gerektiren bir iştir. Bu zorluğa göğüs gerebilmek ve programlamayı daha kolay bir hale getirmek için pek çok özel amaçlı bilgisayar simülasyon dilleri (SIMAN V, SLAM II, GPSS/H, SIMSCRIPT II.5, GASP gibi) ve simülasyon paketleri (MODSIM III, SIMPLE++) geliştirilmiştir.

Simülasyon dilleri, olay programlama (event-scheduling) veya proses etkileşimi (process interaction) olarak adlandırılan iki farklı modelleme yaklaşımından birini kullanır. Winston (1994)'e göre, olay programlama yaklaşımında sistem, karakteristik olayları tanımlayarak ve her olayın gerçekleşmesinden sonra sistemdeki ilgili durum değişikliklerini gerçekleştirecek usulleri oluşturarak modellenir. Simülasyon zaman ekseninde, bir sonraki programlanmış olaya dek simülasyon saatini güncelleyerek ve bu esnada sistemdeki gerekli değişiklikleri ve istatistikleri gerçekleştirerek akar. Proses etkileşimi yaklaşımında, bir gezen birimin (varlık) sisteme girip sistemden ayrılana dek üstlendiği aktiviteler serisini ele alınarak sistem modellenmeye çalışılır. Programlama safhasında, genel amaçlı programlama dilleri, C++, FORTRAN veya BASIC gibi, genellikle olay programlama yaklaşımı kullanılırken, bir kısım özel amaçlı programlama dilleri olay programlama yaklaşımını, bir kısmı proses programlama yaklaşımını (örneğin GPSS) kullanırken; bir kısmı ise herhangi bir yaklaşımın, hatta iki yaklaşımın karışımının kullanılmasını, (örneğin SLAM) mümkün kılar. Erkut (1995b)'e göre de simülasyon modeli tasarlanırken ve simülasyon dili seçilirken, en önemli tercih sebeplerinden biri de simülasyonda zaman tutma yöntemidir. (sabit zaman artışlı ve değişken zaman artışlı yöntem)

Simülasyon dillerinin yanı sıra kullanıma yönelik simülatörler (application-oriented simulators) de pratik hayatta kendileri oldukça yaygın bir kullanım alanı bulmaktadırlar. Çizelge 5.1’de simülasyon paketleri tiplerini ve örneklerini irdelenebilir. Maria (1997)’ye göre, genel amaçlı programlama dilleri, özel amaçlı simülasyon dillerine kıyasla, özel amaçlı simülasyon dilleri de simülatörlere kıyasla daha fazla esneklik sunarken, daha fazla programlama tecrübesine gereksinim duyar. Simülatörlerin kullanımının öğrenilmesi oldukça kolaydır ve modelin kurulması uygulama ile yakın ilişki gösterir.

Çizelge 5.1 Simülasyon paketleri (Maria, 1997)

Simülasyon Paketinin Türü	Örnekler
Simülasyon Dilleri	Arena (önceden SIMAN), Awesim (önceden SLAM II), Extend, GPSS, Micro Saint, SIMSCRIPT, SLX <i>Nesne-yönelik yazılım:</i> MODSIM III, SIMPLE++ <i>Animasyon yazılımı:</i> Proof Animation
Kullanıma Yönelik Simülatörler	İmalat: AutoMod, Extend+MFG, FACTOR/AIM, ManSim/X, MPSIM, ProModel, QUEST, Taylor II, WITNESS İletişim/bilgisayar: COMNET III, NETWORK II.5, OPNET Modeler, OPNET Planner, SES/Strategizer, SES/workbench Ticaret: BPSIM, Extemd+BPR, ProcessModel, ServiceModel, SIMPROCESS, Time machine Sağlık Hizmetleri: MedModel

Tezimizde, sistemimizi modellerken olay programlama yaklaşımı tercih edilmiştir. Özel simülasyon dillerinin sunduğu çeşitli avantajlara, sunduğu standart çatı ve bloklar ile simülasyon modelleme, animasyon ve çıktı istatistikleri alma kolaylığına rağmen, güçlü bir genel amaçlı programlama dili olan Microsoft Visual C++ 6.0’ı karmaşık sistem modelimizde, değişken zaman artışı zaman ilerletme tekniğinin işletilmesi ile simülasyon

zamanının ilerletilmesinde, sistem durumunun güncellenmesinde sistemdeki bileşenlerin ve olayların senkronizasyonundaki performansı ve modellemede ihtiyacımız olan esneklik göz önüne alarak simülasyon dili olarak kullanılmış ve modelimiz tercih edilen programlama dilinin sunduğu platformlarda kodlanarak oluşturulmuştur.

5.3 Simülasyon Modeli

Simülasyon modelimizin doğru bir şekilde kurulması için öncelikle ele aldığımız sistemin bileşenlerini, değişkenlerini, parametrelerini ve bunlar arasındaki ilişkiler ile birlikte varsayımlarımızın belirlenmesi önemlidir. Bu kısımda yürüteceğimiz çalışmalar kesikli olay simülasyonuna dayanmaktadır. Banks vd. (1996)'ya göre kesikli olay simülasyonu, durum değişkenlerinin zaman ekseninde sadece kesikli noktalarda değiştiği sistemlerin modellenmesidir. Çalışmamızın bu kısımda, kesikli olay modellemiz ile ilgili gerekli bileşenler belirlendikten sonra, simülasyon modelimizin formülasyonu ve algoritmanın açıklanması gerçekleştirilmiş olup ve ilgili algoritma ve akış diyagramı doğrultusunda, Microsoft Visual C++ 6.0 kullanarak simülasyon modeli kurulmaya çalışılmıştır.

5.3.1 Simülasyon Modelinin Bileşenleri

Bir sistemi anlamak ve analiz etmek için, bir dizi terimin tanımlanması gerekmektedir. Banks vd. (1996)'ya göre, gezen birim (Varlık), model içerisinde açık bir şekilde temsil edilmesi gereken obje ya da bileşendir. Nitelik, verilen bir gezen birim (varlığın) özelliğidir. Olay, bir sistemin durumunda aniden meydana gelen değişimdir. Faaliyet ise, ne zaman başladığı bilinen, belirli bir parçanın zaman periyodunu temsil etmektedir. Simülasyon modelimizi oluşturmamızın ana amacının ele aldığımız gerçek sistemin davranışını tanımlama, açıklama ve tahmin etme olduğunu belirtmiştik. Gerçek sistemin davranışını tanımlamadan, ilgili sistemin davranışlarını açıklamak ve tahmin etmek mümkün değildir. Sistem durumu bir sistemi herhangi bir anda tanımlamak için gerekli olan tüm bilgileri içeren değişkenlerin birarada verilmesi olarak tasvir edilebilir. Sistem durumunda değişikliklere olay adı verilen anlık etmenler neden olmaktadır.

Sistem değişkenlerini sistem performansı ile ilgili ve ilgisiz değişkenler olarak ikiye ayırabiliriz. Çalıştığımız sistemin tüm ilgili değişkenleri, simülasyon koşusu sırasında etkilenip etkilenmediği göz önüne alınarak iki grupta organize edilmiştir. Bir değişkenin değeri simülasyon koşusu sırasında etkilenmemekte ise dış değişken, değişkenin değeri simülasyon koşusu sırasında diğer değişkenler tarafından tayin ediliyor ise iç değişken olarak tanımlanmıştır. Simülasyon modelimiz tarafından verilen ve manipüle edilemez olarak

kabul ettiğimiz dış değişkenler, denetlenemeyen; bir sonuca ulaşmak için manipüle ettiğimiz dış değişkenler ise denetlenebilir değişkenler olarak sınıflandırılmıştır [12].

Dinamik gezgin tamirci problemimizi ele aldığımız modelimizde yukarıda tanımlanan bileşenler aşağıda tek tek ele alınmıştır.

Gezen Birim (Varlık); müşteriler, tamirciler ve araçlar,

Nitelik; müşterinin konumu (servis verdiğimiz coğrafi alan üzerindeki konumu), müşterinin ihtiyaç duyduğu servis zamanı,

Değişkenler; tamirci ve araçların sayısı (dışsal, girdi değişkeni ve denetlebilir), ardışık müşteri gelişleri arasındaki zaman (dışsal, girdi değişkeni ve denetlenemeyen), müşterinin konumu (dışsal, girdi değişkeni ve denetlenemeyen), müşterilerin ihtiyaç duyduğu servis zamanı (dışsal, girdi değişkeni ve denetlenemeyen), müşterinin tamirciyi bekleme süresi (içsel, çıktı değişkeni ve denetlenemeyen), ortalama bekleme süresi (içsel, çıktı değişkeni ve denetlenemeyen), araçların yolda geçirdiği toplam süre (içsel, çıktı değişkeni ve denetlenemeyen), araçların yolda kat ettiği toplam mesafe (içsel, çıktı değişkeni ve denetlenemeyen), araçların atandığı olay sayısı (içsel, çıktı değişkeni ve denetlenemeyen),

Parametreler; hizmet verdiğimiz coğrafyanın eni ve boyu, ardışık müşteri gelişleri arasındaki zaman dağılımı, gün içindeki toplam hizmet zamanı, tamir araçlarının hızı ve ivmesi, müşteri talebinin olduğu coğrafyanın x ve y koordinatlarındaki dağılımı ve tamir araçlarının kapasitesi,

Olaylar; yeni müşteri talebinin oluşması, tamircinin ilgili müşteri talebine atanması, tamircinin atandığı müşteri talebine doğru yola çıkması, tamircinin olay yerine ulaşması ve servise başlaması, tamircinin servisini bitirmesi ve olay yerinden ayrılması,

Faaliyetler; yolculuk zamanı, servis zamanı, iki ardışık olay arasında gözlemlenen zamandır.

Şüphesiz ki, bileşenlerimizin sınıflandırılmasını çalışmamızın amacına, problemimizde ele aldığımız detay seviyemize ve elimizdeki kaynaklar doğrultusunda gerçekleştirdik. Erkut (1995b)'e göre, tespit edilen farklı amaç, detay seviyeleri ve kaynaklar ile bileşenlerin sayısı, türü ve sınıflandırılması farklı olabilmektedir. Bununla birlikte, tamirci ve araç sayısı hem değişken (dışsal, girdi değişkeni ve denetlebilir) hem de parametre olarak yer almaktadır. Karar verici olarak, simülasyon modelimizi kurarken, simülasyon iterasyonunun herhangi bir kümesi boyunca tamirci ve araç sayısını denetleyebilmekteyiz. Böylece, tamirci ve araç sayısı

simülasyonun her bir iterasyon kümesi için parametre olurken, simülasyon çalışmamızın tamamı için ise dışsal, denetlenebilir bir girdi değişkenidir.

5.3.2 Simülasyon Modelinin Varsayımlarının Belirlenmesi

İncelediğimiz sisteme ilişkin olarak simülasyon modelimizdeki varsayımlar aşağıdaki gibi sıralanabilir:

- Müşteri talepleri sisteme mesai saatinden önce girilmemektedir. Servis talebi için vakanın açılması ancak müşteri taleplerini toplayan operatörün çalışma saatleri içerisinde mümkündür.
- Müşteri talepleri, sadece ilgilenilen coğrafi alan içerisinde toplanır. Bu alanın dışındaki taleplere sistem yanıt vermez.
- İlgilenilen coğrafi alandaki müşterilerin konumları bellidir ve ilgili servis talebinin yapılmasından hemen sonra araç programlayıcısı ile bu bilgi paylaşılabilir.
- Servis talebinde bulunan müşteri, problemini ve mümkün ise gereksinim duyduğu servisin detaylarını operatöre bildirir, operatörden vaka bilgisi ise ilgili müşterinin gereksinim duyduğu servis süresi programlayıcı tarafından tespit edilebilir.
- Bütün servis elemanlarının bilgi, beceri ve yetenekleri eşittir. Müşterilerin herhangi bir servis elemanı tercihi söz konusu değildir.
- Servis talebinin olduğu nokta ile gereksinim duyulacak servis zamanı birbirinden bağımsızdır.
- Mesai saatleri içerisinde hizmet taleplerinin sistemde tutulması için sonsuz yeterlilikte alan vardır. Bu bağlamda mesai saatleri içerisinde gelen hiçbir talep reddedilmez.
- Müşteri taleplerinin geliş oranı, sistemde yer alan talep sayısına ve kuyruk uzunluğuna bağlı olmaksızın sabittir. Müşterilerin kuyruk uzunluğundan dolayı servis hizmetinden vazgeçmeleri söz konusu değildir.
- İlgili servis zamanı, kuyruk uzunluğundan bağımsızdır. Tamir elemanı elindeki işi, kendisini atanan işleri bilse dahi daha hızlı bitiremez.
- Bütün işlerin önceliği aynıdır ve sadece ilgili atama politikasına göre atanırlar.
- Herhangi bir işe atanan araç / tamirci, o işi tamamlamadan (henüz yolda dahi olsa)

başka bir işe yönlendirilemez.

- Araçların kapasite ve kısıtları eşittir.
- Araçlar trafikte aynı akış modeline göre hareket ederler.
- Araçların trafikteki akış modeli, gün içerisindeki zaman diliminden ve araçların kat ettiği yolun yer aldığı coğrafi bölgeden etkilenmez.
- İlgili taleplerin toplanmasına mesai saati bittiği anda son verilir. Mesai saatlerinden sonra gelen müşteri taleplerine yanıt verilmez.
- Araç-Tamirciler, ya işsiz, ya herhangi bir işe ulaşmak üzere yolda, ya da müşteriye servis vermekle meşguldürler.
- Mesai saati bitiminden önce sistemi girmiş müşteri taleplerine, mutlaka aynı gün içerisinde yanıt verilir. Kendilerine atanan işleri mesai saati dahilinde bitiremeyen araçlar fazla mesai yapar.
- Atama politikası aksini belirtmedikçe, servis araçlarımızın hepsi, “sabırlı sunucu” dur. Atandıkları bir iş yok ise en son bulundukları konumda bir sonraki vazifelerini beklerler.
- Mesaisi biten ve elinde yapması gereken işi olmayan araçlar merkeze döner.
- Bir sonraki çalışma günü başladığında, tam kapasite ve hiçbir işe atanmamış olarak merkezde hazır bulunurlar.

5.3.3 Simülasyon Modeli Yardımı ile İrdelenen Dinamik Araç Atama Yaklaşımı

Yukarıda betimlenen vaka ve ilgili kabuller doğrultusunda problemin çözümünü etkileyecek en önemli etken PCServis Şirketinin araçlarının (tamircilerinin) ilgili müşteri taleplerine atanmasını gerçekleştiren politikadır. Atama politikası, sistemin hedeflediği amaçlara ulaşmasını kolaylaştırabilecek, belirlenen performans kriterlerine ulaşmasını sağlayabilecek bunu yaparken de kapasite artırımı gibi karar vericileri ek mali yükümlülükler altına sokmayan kaynak programlayıcılarının ciddi bir silahıdır. Bu doğrultuda uygun atama politikasının seçilmesi veya halihazırdaki politikanın terk edilerek daha yüksek performans göstereceğine inanılan atama politikasına geçilmesinin önemi ihmal edilemez.

Çalışmamızda önerdiğimiz araç atama politikası dinamik bir yaklaşımın ürünüdür ve önceki

kısımlarda etkinliğini ve fizibilitesini ispatlamaya çalıştığımız teknolojik bazı altyapılardan yararlanır. Araç atama problemini, “dinamik bir zaman ekseninde bir ya da birden fazla kapasitenin (araç, tamirci sayısı), bir ya da birden fazla işe (müşteri talepleri) sistemin hedefleri dahilinde en etkin yanıtı vererek çözme” olarak ele alır. Başlıgil (1988)’e göre sıralamanın temel fikri bir işin diğerlerine göre olan durumunu saptamaktır. Sıralama, ilgili işlerin bir kaynak tarafından belli bir sıra ile yanıtlanması olarak düşünülürse, atama için de herhangi bir işin yanıtlanmasında hangi kaynağın en uygun olduğunun tespit edilerek, görevlendirilmesi olarak kabul edilebilir. Bu doğrultuda, atamanın temel fikrini de bir kaynağın (aracın) diğer kaynaklara göre durumunun saptanması olarak ele alınabilir.

Bütün araçlar için her işin yanıt verilmesi için geçen zaman ekseninin üç bileşeni vardır. Bunlardan ilki aracın daha önce atandığı işleri tamamlayarak boş durumu geçene dek kaydedilen zaman, ikincisi aracın bulunduğu konumdan olay yerine ulaşana dek geçen zaman ve üçüncüsü de olay yerinde müşteri talebinin yanıtlanmasına dek geçen zamandır. Problemimiz ile ilgili kabullerimizde, müşterinin gereksinim duyduğu servis zamanını atanan tamirciden bağımsız olduğu ve programcı tarafından müşteri talebinin sisteme girilmesi sırasında tespit edilebildiği belirtilmişti. Problemimizi tezgah çizelgelemeye benzetmeye çalışırsak; ilk iki zaman periyodunu tezgahın ayarlarının yapılması, üçüncü zaman periyodunu ise işleme zamanı olarak kabul edersek, her tezgah için ayar zamanını ne kadar minimize edebilirsek o tezgahtan o kadar yüksek verim alırız diyebiliriz.

Bu bilgiler ve kabuller ışığında, araç filomuzdan maksimum verimi almayı ve müşteri bekleme zamanını oluşturan ilk iki zaman bileşenini minimize etmeye çalışan dinamik araç atama politikamıza En Kısa Ulaşım Zamanı (Shortest Arrival Time- SAT) adını verilmiş ve ilgili politikanın NN ve SQM politikaları ile simülasyon modeli oluşturularak performansı kıyaslanmaya çalışılmıştır.

5.3.4 Simülasyon Modelinin Formülasyonu

SAT, NN ve SQM atama politikaları doğrultusunda oluşturulan simülasyon modelinin formülasyonunda, Erkut (1995b)’ten yararlanılmış ve aşağıdaki adımlar izlenilmiştir.

- Olayların genel akışının tanımlanması,
- Daha önce tanımladığımız sistemin değişken ve parametreler ile diğer elemanlarının semboller ve açıklamalar ile ifade edilmesi,
- Simülasyon algoritmasının açıklanması,

- Akış diyagramının çizilmesi,
- El ile simülasyon denemesi yapılarak, el ile simülasyon tablosunun kurulması.

5.3.4.1 Olayların Akışı

Sistemimizde olayların genel akışı aşağıdaki gibi tanımlanabilir.

- Hizmet talebinin sisteme ulaşması.
- Hizmet talebinin bir olay olarak kabul edilmesi, olduğu coğrafi konumun ve gereksinim duyacağı servis zamanının tespit edilmesi.
- Tüm araç kümesinin bulunduğu coğrafi yer ve araçların kullanılabilirlik durumlarının irdelenmesi.
- İlgili araç-tamircinin, atama politikasına göre olaya atanması.
- Araç-tamircinin müşteriye doğru yola çıkması.
- Araç-tamircinin müşteriye ulaşması ve servis vermeye başlaması.
- Araç-tamircinin müşterideki görevini tamamlaması.
- Atama politikasına göre, olay yerinde bir sonraki işini beklemesi, merkeze dönmesi ya da bir sonraki atandığı iş için yola çıkması.

5.3.4.2 Simülasyon Modelinde Yer Alan Değişken ve Parametreler ile Diğer Elemanlarının Sembolleri

Sistemin elemanlarının, değişken ve parametrelerinin sembolleri ve açıklamaları Çizelge 5.2'de verilmektedir.

Çizelge 5.2 Sistem elemanları ve açıklamaları

Eleman Sembolü	Tanımı
MAX_OLAY	Sistemin bir çalıştırılması sırasında cevap verebileceği maksimum olay sayısı
MAX_ARAÇ	Sistemde yer alan araçların üst sınırı
IDLE	Herhangi bir işe atanmamış, beklemede olan aracın durumu
YOLDA	Herhangi bir işe atanmış, yolda olan aracın durumu
BUSY	Herhangi bir işe atanmış, müşteriye servis vermekte olan aracın durumu
ATANMAMIS	Herhangi bir araç atanmamış olan müşteri talebinin durumu
ATANMIS	Herhangi bir araç atanmış olan müşteri talebinin durumu
BITMIS	Herhangi bir araç tarafından servis verilmiş olan müşteri talebinin durumu
EN	Servis alanının eni
BOY	Servis alanının boyu
LAMBDA	Bir saatte sisteme girmesi beklenen talep sayısı (Poisson Değişkeni)
TOPLAM_SÜRE	Sistemin toplam çalışma süresi
ARAÇ_SAYISI	Sistemde yer alan araç (tamirci) sayısı
HIZ	Servis aracının akış hızı

IVME	Servis aracının hızlanma ve yavaşlama ivmesi
YER_MEAN_X	Normal dağılıma göre X ekseninde müşteri taleplerinin oluşmasını beklediğimiz noktaların ortalaması
YER_STANDARDDEV_X	Normal dağılıma göre X ekseninde müşteri taleplerinin oluşmasını beklediğimiz noktaların standart sapması
YER_MEAN_Y	Normal dağılıma göre Y ekseninde müşteri taleplerinin oluşmasını beklediğimiz noktaların ortalaması
YER_STANDARDDEV_Y	Normal dağılıma göre Y ekseninde müşteri taleplerinin oluşmasını beklediğimiz noktaların standart sapması
SERVIS_MEAN	Normal dağılıma göre müşterinin gereksinim duyduğu servis zamanının ortalaması
SERVIS_STANDARDDEV	Normal dağılıma göre müşterinin gereksinim duyduğu servis zamanının standart sapması
CAPACITY	Servis aracının merkeze uğramadan cevap verebileceği talep sayısı (Eğer kapasitesini tamamlamış veya geçmişse ve yeni bir işe de atanmamış ise merkeze gönderilir.)
zaman	Ana zaman, simüle edilen sistem zamanı
toplam_yolda	Servis araçlarının yolda geçirdiği toplam süre
toplam_busy	Servis araçlarının çalışarak geçirdiği toplam süre
toplam_olay	Servis araçlarının yanıt verdiği olay sayısı
olay_sayısı	Sistemde yanıt verilen toplam olay sayısı
toplam_bekleme	Hizmet taleplerin gelmesinden, servis aracının müşteriye ulaşmasına dek geçen toplam sistem zamanı
olay_var	Mesai bittiği halde sisteme girilmiş yanıt beklemekte olan

	olayların olduğunu gösterir.
yeni_olay	Simülasyon döngüsünde her dakika için kaç adet yeni olay olduğunu tutar.
olusturuldugu_zaman	Simülasyon döngüsünde her olayın oluşturulduğu zaman
son_olay	Her aracın merkezden ayrıldıktan sonra yanıt verdiği olayların sayısı
uzaklik	Her aracın olay yerine uzaklığı
sure	Araçların bir sonraki olay yerine ulaşabileceği zaman dilimi[bos_kalma_zamani + yolculuk zamanı (uzaklık'a bağlı 2 farklı fonksiyon ile hesaplanır.)]
min	Atanan aracın bir sonraki olay yerine ulaşabileceği en kısa zaman (bos_kalma_zamani + minsure)
bos_kalma_zamani	Her aracın idle duruma dönmesine ne kadar zamanı kaldığını gösterir.
minsure	Atanan aracın olay yerine ulaşabileceği en kısa yolculuk zamanı
minyol	Atanan aracın olay yerine ulaşabileceği en kısa uzaklık
atandigi_olay	Tüm simülasyon zamanında araca atanan olayların listesi
olay_bas	Aracın ilgilenmekte olduğu olay (atandigi_olay listesinin bir indeksi)
olay_son	Araca atanmış olayların sonuncusu (atandigi_olay listesinin bir indeksi)

5.3.5 Simülasyon Algoritmasının Açıklanması

Bilgisayar destekli problem çözümünde üç temel adım vardır [4]. Bu adımlar:

1. Algoritma Geliştirme: Problemin analizi ve algoritmanın geliştirilmesi

2. Programın Kodlanması: Programın algoritma kullanılarak, programlama dilinin kurallarına uygun olarak kodlanması.
3. Program Çalıştırma ve Test: Programın çalıştırılarak bilinen sonuçlar için test edilmesi.

Yukarıda belirtilen adımlar doğrultusunda, öncelikle algoritmanın geliştirilmesine çalışılmıştır. Algoritma kısaca program tasarımıdır. [9]

Algoritmayı oluşturarak problemin dikkatlice analiz edilerek, çözüm için neler gerektiği belirlenilmiş, gerekli formüllerin geliştirilmesi tamamlanmış, sonuçların hangi yapıda olması gerektiği ve nelerin istendiği kararlaştırılmıştır. Algoritmada, açıklamalı parçalar arasında bağlantıyı korumaya, programın başından sonuna kadar tam bir özetini vermeye ve mümkün olduğu kadar açık tutmaya gayret gösterilmiştir. Geliştirdiğimiz simülasyon algoritmasını, programı kodlayacağımız C++ programlama diline uygun bir şekilde yazılmıştır, bu yaklaşımımız ile algoritmadan programı kodlamanın daha etkin bir şekilde tamamlanması hedeflenilmiştir. Simülasyon algoritmasının tamamlanmasından sonra, akış diyagramı ile tanımlanan işlemlerin şematik gösteriminden yararlanılmıştır. Aşağıda sistem simülasyonunu çalışmamızda ortaya koymaya çalıştığımız yaklaşımımıza (SAT) göre gerçekleştirdiğimiz programının algoritması açıklamaları ile incelenebilir.

5.3.5.1 SAT Politikası için Simülasyon Algoritması

ADIM 1 Gerekli değişken ve parametrelerin okutulması ve tanımlanmasını gerçekleştir.

OKU: EN, BOY, LAMBDA, TOPLAM_SURE, ARAC_SAYISI, HIZ, IVME, YER_MEAN_X, YER_STANDARDDEV_X, YER_MEAN_Y, YER_STANDARDDEV_Y, SERVIS_MEAN, SERVIS_STANDARDDEV, CAPACITY, RAPOR_ARALIGI.

ADIM 2 Sistemi simülasyona hazır hale getir.

Arac.i_x = EN/2 // Tüm araçları merkeze getir.

Arac.i_y = BOY/2 // Tüm araçları merkeze getir.

Arac.i_state = IDLE // Tüm araçların durumunu kullanılmaya hazır yap.

zaman = 0 // Simülasyon zamanını 0 olarak belirle.

ADIM 3 Simülasyon döngüsünü başlat.

```
for ( zaman = 0; zaman < TOPLAM_SÜRE veya olay_var; zaman++)
```

```
{
```

ADIM 4 Poisson dağılımını kullanarak yeni_olay sayısını oluştur.

ADIM 5 Yeni_olay'lar için olayların koordinatlarını ve sürelerini normal dağılım ile belirle:

```
{
```

```
x = normal (YER_MEAN_X, YER_MEAN_STANDARDDEV_X)
```

```
y = normal (YER_MEAN_Y, YER_MEAN_STANDARDDEV_Y)
```

```
sure = normal (SERVIS_MEAN, SERVIS_STANDARDDEV)
```

```
olay_ekle (x,y, sure)
```

```
}
```

ADIM 6 Olayların Eklenmesi.

```
Void olay_ekle (x, y, sure)
```

Olayın koordinatları ve hizmet süresi için normal dağılımın oluşturabileceği sınır dışı değerleri sınır değerlere indirilmesini gerçekleştir.

```
olay_i.x = x, // oluşturulan olayın apsisini x yap.
```

```
olay_i.y = y, // oluşturulan olayın ordinatını y yap.
```

```
olay_i.state = ATANMAMIS, // oluşturulan olayın durumunu ATANMAMIS yap.
```

```
olay_i.olusturuldugu_zaman = zaman, //olayın oluşturulduğu zamanı işaretle.
```

```
olay_sayisi++; //olay sayısını bir arttır.
```

ADIM 7 Olayların durumlarını kontrol et, atanmamış olan olaylara araç ata.

```
for (j =0; j<olay_sayisi; j++)
```

```
if (olay_j.state == ATANMAMIS)
```

```
arac_ata.
```

ADIM 8 Bütün araçların olay yerine uzaklığını tespit et.

```
for(i=0;i<ARAC_SAYISI;i++)
```

```
uzaklik = | arac_i.x-olay_i.x | + | arac_i.y-olay_i.y |
```

ADIM 9 Bütün araçların olay yerine ulaşma zamanını tespit et. (bos_kalma_zamani + yolculuk zamanı)

```
for(i=0;i<ARAC_SAYISI;i++)
```

```
{
```

```
if(arac_i.bos_kalma_zamani<zaman)
```

arac_i.bos_kalma_zamani=zaman; //aracın durumu idle ise, aracın bos_kalma_zamani'nı simülasyon zamanına eşitle.

```
if(uzaklik<6.67)
```

```
sure= arac_i.bos_kalma_zamani + 2 * (sqrt (uzaklik / IVME))*60
```

else // aracın durumu idle değil ise, aracın bos_kalma_zamani ile yolculuk zamanını toplayarak olay yerine ulaşabileceği en kısa zamanı tespit et.

```
sure= arac_i.bos_kalma_zamani + (uzaklik / HIZ + HIZ / IVME)*60
```

```
if (sure < min)
```

```
{atanan=i;
```

```
min=sure; //aracın olay yerine ulaşabileceği en kısa zaman (bos_kalma_zamani + minsure)
```

```
minsure= min- arac_i.bos_kalma_zamani} //aracın olay yerine ulaşabileceği en kısa yolculuk zamanı
```

```
}
```

ADIM 10 Olay yerine en kısa zamanda ulaşabilecek aracı ilgili olaya ata. min(bos_kalma_zamani + yolculuk zamanı)

```
olay_j.state= ATANMIS; // Atanan olayın durumunu ATANMIS yap.
```

```
arac_i[atanan].toplam_olay++; // Atanan aracın yanıt verdiği toplam_olay sayısını bir arttır.
```

```
arac_i[atanan].son_olay++; // Atanan aracın merkeze dönmeden önce yanıt verdiği toplam olay sayısını (son_olay) bir arttır.
```

```

arac_i[atanan].state= YOLDA; // Atanan aracın durumunu YOLDA yap.

arac_i[atanan].x= olay_j.x; // Atanan aracın apsisini atandığı olayın apsisini yap.

arac_i[atanan].y= olay_j.y; // Atanan aracın ordinatını atandığı olayın ordinatı yap.

arac_i[atanan].atandigi_olay= olay; // Atanan aracın atandığı olayı, olay yap.

arac_i[atanan].bos_kalma_zamani= min; // Atanan aracın bos_kalma_zamani'nı, min yap.

arac_i[atanan].kalan_busy= olay_j.sure; // Atanan aracın kalan_busy zamanını, atandığı
olay'ın süresine eşitle.

arac_i[atanan].kalan_yolda= minsure; // Atanan aracın kalan_yolda zamanını, minsure'ye
eşitle.

arac_i[atanan].olay_son++; // Atanan aracın olay_son'unu bir arttır.

ADIM 11 Araçların durumlarını değerlendir. Durumu son bulan araçları bir sonraki
durumlarına geçir. (Adım 2'de tüm araçların durumlarını IDLE yapmış, adım 9'da ise bir
olaya atadığımız aracın durumunu YOLDA olarak revize etmiştik.)

for(i=0;i<ARAC_SAYISI;i++)

if(arac_i.state==YOLDA) //Durumu YOLDA olan araçlar için,

{

{ arac_i.kalan_yolda--; //Atanan aracın kalan_yolda süresini bir azalt.

arac_i.toplam_yolda++; //Atanan aracın toplam_yolda süresini bir arttır.}

if (arac_i.kalan_yolda==0)

{ if(arac_i.atandigi_olay[arac_i.olay_bas]!=-1

toplam_bekleme += zaman-olay_j.olusturuldugu_zaman; // Müşterinin toplam bekleme
zamanını, simülasyon zamanından, olayın oluşturulduğu zamanı çıkararak elde et.

arac_i.state==BUSY //Aracımız olay yerine ulaşmış demektir. Aracın durumunu BUSY yap.

}

}

```

```

}

else if(arac_i.state==BUSY) //Durumu BUSY olan araçlar için,

{ arac_i.kalan_busy--; //Atanan aracın kalan_busy süresini bir azalt.

arac_i.toplam_busy++; //Atanan aracın toplam_busy süresini bir arttır.

if (arac_i.kalan_busy==0)

{

olay_j.state=BITMIS; //Atanan olayın durumunu BITMIS yap.

arac_i.olay_bas++;

if(arac_i.olay_bas== arac_j.olay_son) //araca yeni görev atanıp atanmadığını kontrol et.

arac_i.state=IDLE; //Servisini bitiren ve yeni görevini bekleyen aracın durumunu IDLE yap.

else

arac_i.state=YOLDA; // Servisini bitiren ve yeni göreve atanan aracın durumunu YOLDA yap.

}

}

```

ADIM 12 Kapasite kısıtının olduğu durumlarda, araç kapasite kısıtını doldurduğu işten sonra aracın merkeze gönderilmesini sağla.

```

{

if(arac_i.son_olay>=CAPACITY) // Aracın merkezinden ayrıldığı son zamandan bu yana servis verdiği müşteri talebi sayısı CAPACITY'sine eşit ya da büyük olup olmadığı kontrol et.

{

min = | arac_i.x-EN/2 | + | arac_i.y- BOY/2 | //CAPACITY'sine ulaşmış aracın merkezden olan uzaklığı tespit et.

arac_i.state=YOLDA; //Merkeze gönderilen aracın durumunu YOLDA yap.

```

```
arac_i.x= EN/2; // Atanan aracın apsisini merkezin apsisi yap.
```

```
arac_i.y= BOY/2; // Atanan aracın ordinatını merkezin ordinatı yap.
```

```
arac_i.atandigi_olay[arac_i.olay_bas]= -1 // Araç merkeze gönderildiği ve istatistiklerimizi bozmamak için olay_bas'ı -1 yap.
```

```
arac_olay_son ++ // Olay_son'u bir arttır.
```

```
if(min<6.67)
```

```
arac_i.kalan_yolda= 2 * (sqrt (min / IVME))*60
```

```
else
```

```
arac_i.kalan_yolda= (min / HIZ + HIZ / IVME)*60
```

```
arac_i.son_olay=0; //Araç merkeze ulaştığı için kapasite kısıtını sıfırla.
```

```
}
```

```
}
```

ADIM 13 Simülasyon zamanı (TOPLAM_SURE) bitmiş olsa bile, simülasyon döngüsüne devam etmek için sistemde servis bekleyen müşteri talebi ve görevini bitirmemiş araç olup olmadığını kontrol et.

```
olay_var()
```

```
{
```

```
for(j=0;j<olay_sayisi;j++)
```

```
if(olay_j.state!=BITMIS) // Sistemde servis bekleyen müşteri talebi olup olmadığını kontrol et.
```

```
return 1;
```

```
for(i=0;i<olay_sayisi;i++)
```

```
if(olay_i.state!=IDLE) //Sistemde görevini bitirmemiş araç olup olmadığını kontrol et.
```

```
return 1;
```

```
return 0 // Her iki kontrol şartı da izin veriyor ise sistemin simülasyon döngüsüne son
```

vermesine izin ver.

}

ADIM 14 İstatistikleri topla ve sonuçları yaz.

Yukarıda verilen adımlar dahilinde yazılan simülasyon programının kodları açıklamaları ile birlikte Ek 2’de bulunabilir.

5.3.5.2 NN ve SQM Politikaları için Simülasyon Algoritması

Çalışmamızda ortaya koymaya çalıştığımız atama yaklaşımımızı literatürde kendine yer bulmuş atama yaklaşımları ile kıyaslayarak performansını test etmek hedeflenmiştir. Bu doğrultuda sisteme ulaşan müşteri taleplerine, olay yerine en yakın konumda bulunan ve herhangi bir işe atanmamış aracın (IDLE), IDLE araç yok ise en kısa zamanda IDLE duruma geçecek aracın atandığı (NN) yaklaşım ile herhangi bir müşteri talebine yanıt verdikten sonra, eğer araç programlayıcı tarafından henüz başka bir işe atanmayan araçların müşteri taleplerinin medyanına konumlandırma (SQM) yaklaşımının performansları kıyaslanmıştır. Her iki atama politikasının algoritması ve program kodları kısım 5.4.4.1’de verdiğimiz SAT yaklaşımı ile benzerlik gösterir. Bu doğrultuda program algoritması verilirken, sadece farklı adımlar incelenecektir. NN ve SQM yaklaşımlarının kodlanması sırasında aynı kod akışından yararlanılmış, iki farklı yaklaşımın üreteceği farklı sonuçlar, sadece CAPACITY değişkeninin atanacağı değere bağımlı kılınmıştır. NN yaklaşımı için CAPACITY’nin çok büyük bir değer alması (örneğin 1000), SQM yaklaşımı için ise 1 alınması sağlanmıştır. Aşağıda NN ve SQM yaklaşımlarına göre gerçekleştirdiğimiz programların algoritması açıklamaları ile incelenebilir.

Simülasyon Algoritması:

ADIM 1,2,3,4,5,6,7 // SAT Yaklaşımındaki adımların aynısı.

ADIM 8 IDLE araçların olay yerine ulaşma zamanlarını tespit et. (yolculuk zamanı)

```
for(i=0;i<ARAC_SAYISI;i++)
```

```
if(arac_i.state==IDLE) // IDLE durumdaki araçları tek tek tespit et.
```

```
if ( ( |arac_i.x-olay_i.x| + |arac_i.y-olay_i.y| ) < min ) // olay yerine en yakın IDLE durumdaki aracı tespit et.
```

```
{atanan=i; //Tespit edilen bu aracı olaya ata.
```

$\min = | \text{arac_i.x} - \text{olay_i.x} | + | \text{arac_i.y} - \text{olay_i.y} |$ //Aracın yolda geçirmesi gereken mesefeyi min'e eşitle.

}

if(min<6.67) //Aracın olay yerine uzaklığı 6.67 km'den küçükse aşağıdaki formüle göre yolculuk zamanını tespit et.

$\text{arac}[\text{atanan}].\text{kalan_yolda} = 2 * (\text{sqrt}(\min / \text{IVME})) * 60$

else // //Aracın olay yerine uzaklığı 6.67 km'den büyükse aşağıdaki formüle göre yolculuk zamanını tespit et.

$\text{arac}[\text{atanan}].\text{kalan_yolda} = (\min / \text{HIZ} + \text{HIZ} / \text{IVME}) * 60$

ADIM 9 Olay yerine en kısa zamanda ulaşabilecek aracı ilgili olaya atadıktan sonra ilgili güncellemeleri gerçekleştir.

$\text{olay_j.state} = \text{ATANMIS}$; // Atanan olayın durumunu ATANMIS yap.

$\text{arac_i}[\text{atanan}].\text{toplam_olay}++$; // Atanan aracın yanıt verdiği toplam_olay sayısını bir arttır.

$\text{arac_i}[\text{atanan}].\text{son_olay}++$; // Atanan aracın merkeze dönmeden önce yanıt verdiği toplam olay sayısını (son_olay) bir arttır.

$\text{arac_i}[\text{atanan}].\text{state} = \text{YOLDA}$; // Atanan aracın durumunu YOLDA yap.

$\text{arac_i}[\text{atanan}].\text{x} = \text{olay_j.x}$; // Atanan aracın apsisini atandığı olayın apsisini yap.

$\text{arac_i}[\text{atanan}].\text{y} = \text{olay_j.y}$; // Atanan aracın ordinatını atandığı olayın ordinatını yap.

$\text{arac_i}[\text{atanan}].\text{atandigi_olay} = \text{olay}$; // Atanan aracın atandığı olayı, olay yap.

$\text{arac_i}[\text{atanan}].\text{kalan_busy} = \text{olay_j.sure}$; // Atanan aracın kalan_busy zamanını, atandığı olay'ın süresine eşitle.

$\text{arac_i}[\text{atanan}].\text{olay_son}++$; // Atanan aracın olay_son'unu bir arttır.

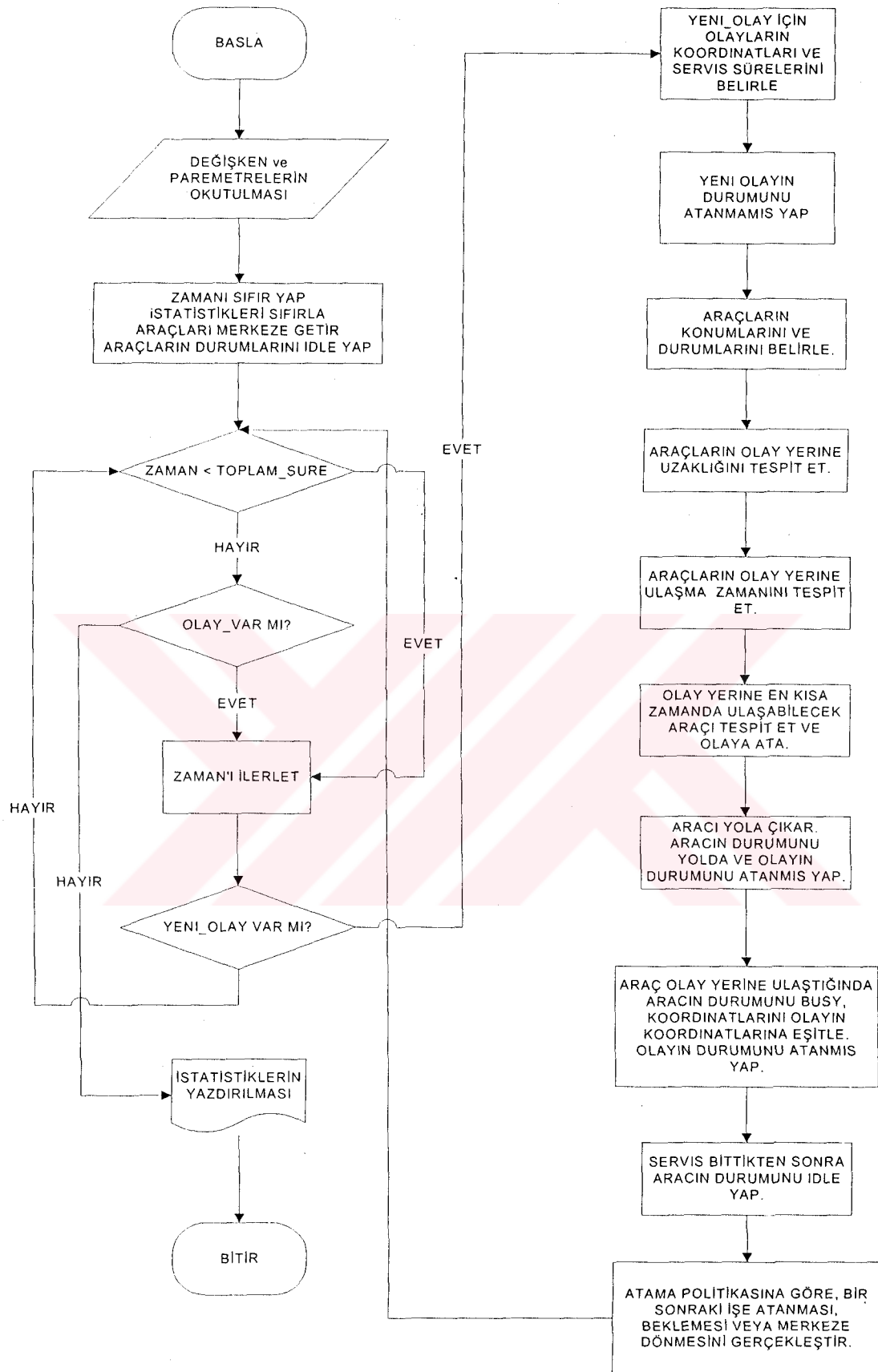
ADIM 10,11,12,13 ve 14 // SAT Yaklaşımındaki adımların aynısı.

Yukarıda verilen adımlar dahilinde yazılan simülasyon programının kodları açıklamaları ile birlikte Ek 3'te bulunabilir.

5.3.6 Akış Diyagramı

Algoritma tasarlandıktan sonra akış diyagramı oluşturulmuştur. Akış diyagramı, algorithmadaki işlemlerin şematik olarak gösterilmesidir. Bu şekilsel gösterimde oklar yardımı ile akış diyagramının yönü işaret edilmeye, sabit şekiller kullanılarak da program akışının evrensel bir dil ile açıklanmaya çalışılmıştır. Şekil 5.3'te bir önceki kısımda algoritmaları verilen üç farklı yaklaşım için simülasyon modeli, akış diyagramı ile tanımlanarak ilgili işlemlerin şematik gösterimi gerçekleştirilmiştir.





Şekil 5.3 Atama politikalarının (SAT, NN ve SQM) algoritmalarının akış diyagramı ile gösterimi

5.3.7 El ile Simülasyon

İlgili algoritmaların açıklanmasından ve akış diyagramının çizilmesinden sonra, algoritma ve akış diyagramı ile ifade edilen işlemler dizisi el ile çalıştırılması sonucu ulaşılan sonuçlar paylaşılacaktır. Erkut (1995b)'e göre el ile simülasyonun amacı, algoritmanın sistemdeki olayların simülasyonunu gerçek anlamda ifadelendirip ifade edilip edilmediğinin anlaşılmasıdır. El ile simülasyon yolu ile, modelin kuramsal yapısı gözden geçirilmeye, modelin çalışması denenmeye ve bilgisayar modeline geçmeden önce uygulamada çıkması muhtemel eksiklikler ve aksaklıklar öngörülme ve önüne geçilmeye çalışılmıştır. El ile simülasyonun bu faydalarının yanı sıra, gerçek verilerle çalıştırılmış bilgisayar modelinin doğrulanması ve gerçekleşmesi sırasında da el ile simülasyondan yararlanılmıştır. Bilgisayar modelinin kurumsal model ile karşılaştırılması adım adım izlenilmiştir.

Ek 4'te bilgisayar yardımı ile simüle edilen sistem çıktıları, Ek 5'te ise aynı sistemin el ile simülasyonu incelenebilir. Her iki yöntem ile elde edilen sonuçlar kıyaslandığı bilgisayar yardımıyla elde ettiğimiz çıktının kuramsal sistemimizle ve gerçek sistemimizle uyumlu olduğunu görülmüştür.

5.4 Simülasyon Modelinin Doğrulanması ve Gerçeklenmesi

Simülasyon modelinin doğrulanması ve gerçekleşmesi, simülasyon modelinin en önemli ve en zorlu uğraşlarından biridir. Banks vd. (1996)'ya göre doğrulama (verification), modeli doğru kurmak ile ilgilenir ve şu sorulara cevaplandırmaya çalışır: Model, bilgisayar ortamında doğru olarak kuruldu mu? Modelin parametreleri ve mantıksal yapısı doğru olarak gösterildi mi? Modelin gerçekleşmesi (validation) ise doğru modeli kurmak ile ilgilenir. Modelin gerçek sistemin, doğru bir temsili olup olmadığını sorgular.

5.4.1 Simülasyon Modelinin Doğrulanması

Modelin doğrulanmasının amacı, çalışmamızın ilk dört bölümünde oluşturduğumuz kavramsal modelin, bilgisayar ortamında doğru olarak temsil edildiğini garanti altına almaktır. Banks vd. (1996)'da yer alan doğrulama önerilerinden modelimizin doğrulanması sırasında yararlandıklarımız aşağıda irdelenebilir:

- 1.) Kısım 5.3.5'te oluşturduğumuz akış diyagramı ile herhangi bir olay meydana geldiğinde, sistemin alacağı mantıksal olarak mümkün aksiyonlar ve her aksiyon ve olay için sistemin model mantığı ve olayların akışı toplu olarak takip edilebilir.
- 2.) Modelin bilgisayar temsili, kurucusundan farklı kişiler tarafından kontrol edilmiştir.

- 3.) Modelde kullanılan girdi parametreleri, parametre.txt dosyasından okutturulmuş ve kontrol değişkenleri hariç değiştirilmediğinden emin olunmuştur.
- 4.) Kullanılan her değişkenin tam bir tanımı verilmiş ve kod kısmının her ana kısmı için genel tanımlama ve açıklamalar yapılmıştır. Böylelikle modelin kurucusunun ya da başka birinin daha sonra simülasyon modelini doğrulaması kolaylaştırılmıştır.
- 5.) Simülasyon çıktılarının, modelin kendiliğinden alması sağlanmıştır. Böylelikle, ayrı bir hesaplama yöntemi ile ortaya çıkabilecek hataların önüne geçilmeye gayret edilmiştir.
- 6.) Microsof Visual C++ Debugger'den modelin kurulması sırasında yüzleşilen hataların bulunması ve düzeltilmesi konusunda yararlanılmıştır.
- 7.) Çeşitli girdi değişkenleri ile modelin çıktı değişkenleri, sebep sonuç ilişkisinin tutarlılığı açısından takip edilmiş, seçilmiş bazı çıktı değerleri için (örneğin simülasyon süresinde gerçekleşen olay sayısı) model çalıştırılmadan önce belli bir değer öngörülmüş ve yapılan kıyaslamalar sonucu beklenmedik bir çıktı görülmesi halinde simülasyon kodlarının tekrar gözden geçirilmiştir.

5.4.2 Simülasyon Modelinin Gerçeklenmesi

Banks vd. (1996)'ya göre simülasyon modelinin gerçekleşmesi, modelin ve modelin davranışlarının, gerçek sistem ve gerçek sistemin davranışları ile kıyaslandığı tüm prosestir. Simülasyon modelinin doğrulanması ve gerçekleşmesi, çalışmamızın pek çok aşamasında eş zamanlı olarak gerçekleştirilmiş olup hiçbir modelin, çalışılan sistemin tam bir temsili olamayacağı göz önüne alınarak modelin tutarlılığı ile gereksinim duyduğu zaman ve çaba arasında doğru bir denge oturtulmaya çalışılmıştır.

Aşağıda verilen, Naylor ve Finger (1967) tarafından formülize edilen üç adımlı gerçekleştirme prosesi, simülasyon modelimizin gerçekleşmesi için kullanılmıştır.

- 1.) Görünüş olarak gerçekleşmesi yüksek bir model kurulur,
- 2.) Modelde ele aldığımız kabuller gerçekleşir,
- 3.) Model içinde gerçekleşen girdi-çıkı dönüşümleri, gerçek sistemdeki girdi-çıkı değişkenleri ile karşılaştırılır.

Görünüş Gerçekliği:

Sistemin modellenmesi sırasında öncelikli hedefiniz, modelin kullanıcıları ve simüle edilen gerçek sistem hakkında bilgi sahibi olan kullanıcı ve şahıslara görünüş olarak makul bir model oluşturmak olmuştur. Duyarlılık analizlerinden yararlanılarak, modelin geçerliliği sınanmaya çalışılmıştır. Modelin birden fazla girdi değişkeninde değişikliğe giderek, modelin beklenen şekilde davranıp davranmadığı incelenmiştir. Modelimizde yer alan değişkenlerin çokluğu göz önüne alındığında, irdelenebilecek pek çok duyarlılık analizinin mevcut olduğu görülse de, en kritik olduğunu öngördüğümüz, araç / tamirci sayısının ve müşteri talepleri arasındaki zamanın ortalama bekleme zamanı üzerine etkisi gözlenmiştir. Müşterilerin geliş oranı (müşterilerin servis için talepleri) sıklaştıkça, tamircilerimizin meşguliyet oranlarının, müşterilerin bekleme zamanlarının artması beklenmiş ve modelin de beklendiği doğrultuda davrandığı gözlemlenmiştir. Aynı şekilde, modelde yer alan araç sayısı arttırıldıkça, tamircilerimizin meşguliyet oranlarının, müşterilerin bekleme zamanlarının azalması beklenmiş ve modelin de beklenen doğrultuda davrandığı gözlemlenmiştir.

Model Varsayımlarının Gerçeklenmesi:

Modelimizde yer alan varsayımlarımızı kısım 5.3.2’de detayları ile irdlemiştik. Ele aldığımız varsayımlarımızı iki ana grupta organize edebiliriz. Yapısal varsayımlar ile sistemin işleyişi üzerine varsayımlarda bulunup, özellikle araçların ve müşterilerin konumlarının tespiti gibi birkaç yıl öncesine kadar teknolojik ve ekonomik olarak fizible olmayan yaklaşım ve tekniklerin, çalışmamızın üçüncü bölümünde detayları ile irdelemeye çalıştığımız teknolojiler sayesinde, gerçek hayatta da uygulanmasının mümkün olduğu ispat edilmeye çalışılmıştır.

Veri varsayımları ise güvenilir verinin toplanması ve verinin istatistiksel analizinin doğru olarak yapılması üzerine kurulması yerine, ilgili varsayımlara uygun olasılık dağılımların seçilmesi ve parametrelerin tahmin edilmesi sureti ile bir servis şirketini temsil eder şekilde kurulmaya çalışılmıştır.

Girdi-Çıktı Dönüşümlerinin Gerçeklenmesi:

Simülasyon modelinin gerçekleşmesinde en önemli ve en güçlü gerçekleştirme noktası girdi-çıktı dönüşümlerinin gerçekleşmesidir [12]. Bununla birlikte Banks vd. (1996)’ya göre, çalışmamızda olduğu gibi, irdelenen sisteme ait tam bir operasyonel veri mevcut olmadığında tam bir girdi-çıktı dönüşümlerinin gerçekleşmesi mümkün olamamaktadır. Modellenmiş olan sistemin, mevcut alt sistemleri irdelenmiş ve kısmi girdi-çıktı gerçekleşmesi yapılmaya

çalışılmıştır.

Modellemiş olduğumuz sistemin, en önemli alt sistemini detayları ile çalışmamızın dördüncü bölümünde verdiğimiz trafik akışı alt sistem oluşturmaktadır. Kısım 4.4'te incelediğimiz literatürde yer alan ortalama hız ve ivme değerleri yardımıyla şekillendirdiğimiz ve İstanbul Kent'inin trafik akışını modellemek için uygun gördüğümüz ortalama akış hızı ve ivme değerleri ile oluşturduğumuz verilen bir mesafe için beklenen ulaşım zamanını Çizelge 4.6'da paylaşmıştık. Modellemeye çalıştığımız sisteme ait gerçek veriler elimizde olmadığı için, simülasyon modelimizin koşturulması ile elde ettiğimiz sonuçlar, Amerika Birleşik Devletleri'nin (A.B.D) ulaşım karakteristikleri hakkında oldukça detaylı veri içeren Çizelge 4.7 ve 4.8 ile ve Ek 1'de verilen İstanbul Kentine ait 35 nokta arasında sabah ve akşam trafik yoğunluğunda ölçümlenen veriler ile karşılaştırılmıştır. İki verinin uyumlu olduğu, modelin gerçek sistemin davranışını temsil etmeye muktedir olduğu söylenebilir.

5.5 Simülasyon Sonuçları

Çalışmamızın bu kısmında simülasyon programımız tarafından oluşturulan veri irdenmeye ve analiz edilmeye çalışılmıştır. Amacımız sistemimizin performansını tahmin etmeye çalışırken, üç alternatif atama yaklaşımının performanslarını kıyaslamaktır. Sistemin performansı ortalama bekleme zamanı ile ölçülmüş olup, toplam kat edilen mesafenin de ölçümü yapılmış ve yaklaşımların kıyaslanması sırasında yararlanılmıştır.

Öncelikle sonuçları incelenecek olan simülasyon modelinin bitirilerek mi ya da sabit durumunun mu analiz edileceğine karar verilmiş, bu doğrultuda simülasyon sonuçları ilgili istatistiksel düzeltmeler yapılarak ve güven aralıkları verilerek gösterilmiştir.

5.5.1 Sabit-Durum Simülasyonlar İçin Çıktı Analizleri

Sistemimizin doğası gereği, biten (terminating) simülasyon (sistemin sabah 8:00'dan akşam 16:00'ya kadar çalışması ve daha sonra simülasyonun sonlandırılması) ile temsil edilmesi uygun gözükürken; atama algoritmalarının arasındaki farkların daha net gözlenlenebilmesi ve daha da önemlisi 480 dakikalık koşumlar ile simülasyonun geçiş safhasından çıkıp, sabit-durum safhasına geçememesi üzere, sistemin uzun bir koşum ile sabit-durumdaki davranışının gözlenmesi ve performansının ölçülmesi tercih edilmiştir.

Çalışmamızda olduğu gibi pek çok simülasyon analizlerinde, sistem boş iken koşturulmaya başlanır ve sistemin doğasına göre belli bir süre kararsız durumda kalır. Genellikle sistemin sadece sabit-durumdaki performansı ile ilgilenildiğinden, sistemin sabit-durumdaki

performansı incelenmek için, kararlı halini tanımlanıp bu durumdan simülasyonun başlatılması, ilk kısım verinin silinmesi, bağımsız koşumların hareketli ortalamalarının alınması veya uzun bir simülasyon koşumu yaptırılıp dizilerin ortalamalarının kıyaslanması veya kararsız kısmın çıkartılması yöntemleri tercih edilebilir. Analizimizde dizi ortalamalarının kıyaslanmasından yararlanılmıştır. (Banks vd., 1996)

5.5.2 Sabit-Durum Simülasyonlarında Aralık Tahmini için Dizi Ortalamaları

Sabit-durum simülasyon modellerinin çıktı analizlerini gerçekleştirmek için var olan alternatiflerin arasından dizi ortalamalarının kıyaslanmasının tercih edilmesinin sebebi, tek ve uzun bir koşuma dayanan deney tasarımı kullanılmasıdır. Bu yöntem, uzun simülasyon koşuturulup, simülasyon birkaç parçaya bölünüp, her parçanın bir dizi ve alt örnek oluşturduğu kabul edilip, dizilerin ortalamalarının kıyaslanmasına dayanır. Bu yöntemin dezavantajı ise verinin tek koşum ile üretilmesi, bir önceki dizinin sonucunun, bir sonraki dizinin başlangıcını etkilemesi nedeni ile veri bağımlıdır.

Dizi ortalamaları yöntemi bağımlı veri nedeniyle ortaya çıkan problemi, tek koşumdan oluşturulan veriyi, birkaç diziye bölerek ve her bir dizinin ortalamasına sanki bağımsızmış gibi davranarak çözmeye çalışır. (Banks vd., 1996)

Kesikli-zaman prosesinden elde edilen çıktı verisi, ortalama bekleme zamanı, $\{Y_i, i = n_0, n_0 + 1, \dots, n_0 + n\}$ toplam n 'lik koşumda Y_i olsun. her biri m büyüklüğünde olan ($m = n/k$) k dizi oluşturulduğunda, $j = 1, 2, 3, \dots, k$ için, her dizinin ortalaması:

$$\bar{Y}_j = \frac{1}{m} \sum_{i=(j-1)m+1}^{jm} Y_i \quad (5.1)$$

Dizi ortalamalarının ortalaması ise $\bar{Y}$ olup, eşitlik (5.2) yardımı ile hesaplanabilir.

$$\bar{Y} = \frac{1}{k} \sum_{j=1}^k \bar{Y}_j \quad (5.2)$$

Örneğin ortalamasının varyansı ise eşitlik (5.3) 'te yer alan s^2/k ile tahmin edilebilir.

$$s^2/k = \frac{1}{k} \sum_{j=1}^k \frac{(\bar{Y}_j - \bar{Y})^2}{k-1} \quad (5.3)$$

Bu noktada eldeki ham veriden doğru yararlanabilmek için dizi sayısının tespit edilmesi önemlidir. Law ve Carson (1979)'a göre çok sayıda dizi ortalamasından ve daha küçük dizi

büyükliğünden yararlanarak ($100 \leq k \leq 400$) otokorelasyonu hesaplamayı önerirken, Schmeiser (1982), 10'dan az sayıda dizi ile yapılan analizlerin zayıf sonuçlar doğurduğunun altını çizerken, dizi sayısının 30'un üzerine çıkarmanın çok az olumlu katkısı olduğu belirtmiştir. Analizimizde her bir dizinin 480 dakikadan oluştuğu ($m=480$), 50 dizi oluşturulmuş, her bir alternatif için toplam 24.000 dakikalık uzun tek bir simülasyon koşumundan yararlanılmıştır.

Simülasyon koşumunun ortalamasının hesaplanmasında, güven limitlerinden yararlanılmıştır. Aralık tahminlerinin tercih edilmesinin sebebi, ortalamanın örnekten örneğe farklılıklar göstermesidir. Güven aralıkları ile ortalama için tahmin yerine, ortalama için alt ve üst sınırlar oluşturulmuştur. Aralık tahmini, gerçek ortalamanın tahmininde ne kadarlık belirsizlik olduğunu belirler. Aralık daraldıkça, tahminimizin daha tutarlı olduğunu söyleyebiliriz.

Güven aralıkları şu şekilde tanımlanabilir [21]:

$$\bar{Y} \pm t_{(\alpha/2, k-1)} \frac{s}{\sqrt{k}}$$

Burada s , $\bar{Y}$ 'in Standard hatası, α tercih edilen değer seviyesi, $t_{(\alpha/2, k-1)}$ ise t dağılımının $k-1$ özgürlük derecesi ile üst kritik değeridir. Çalışmamızda %95 güven aralığı tercih edilmiş olup, 50 dizi için ilgili değer, $t_{(0.025, 49)}$, 2,010'dir [21].

5.5.3 Çeşitli Müşteri Yoğunlukları için Alternatif Atama Politikalarının Kıyaslanması

PCServis Şirketi ile ilgili vakanın betimlenmesi sırasında şirketin veri tabanına saatte ortalama 3 müşteri talebinin ulaştığı ($\lambda=3$) belirtilmişti. Alternatif atama politikalarının (SAT, NN ve SQM) performanslarının sağlıklı olarak kıyaslanabilmesi için betimlenen $\lambda=3$ müşteri yoğunluğu ile yetinilmeyip 4 farklı müşteri yoğunluğu (müşteri trafiği) tespit edilmiştir. Bu seviyeler:

- Hafif trafik ($\lambda=1$)
- Az yoğun trafik ($\lambda=2$)
- Yoğun trafik ($\lambda=3$)
- Çok yoğun trafik ($\lambda=4$)'dir.

Dört müşteri seviyesinin herbiri için atama algoritmalarının performansları kıyaslanması için yapılan simülasyon koşumlarında araç sayısı 5 olarak tutulmuş, her trafik durumunda

algoritmaların çok yakın sayıda müşteri talebine cevap vermesine özellikle dikkat edilmiş böylelikle ilgilenilen ortalama bekleme zamanı ve kat edilen mesafenin sadece atama politikalarının performansından kaynaklanmasına gayret edilmiştir. Programlama algoritmamız gereği simülasyon koşumları simülasyon süresi dolduğunda ve sistemde yanıtlanmamış müşteri talebi kalmadığında ancak sonuçlanabilmekte idi. Bu kabul ışığında, her simülasyon koşumunun ilk 24.000 dakikası içindeki herbiri 480 dakikadan oluşan ilk 50 dizisi çıktı analizine dahil edilmiştir.

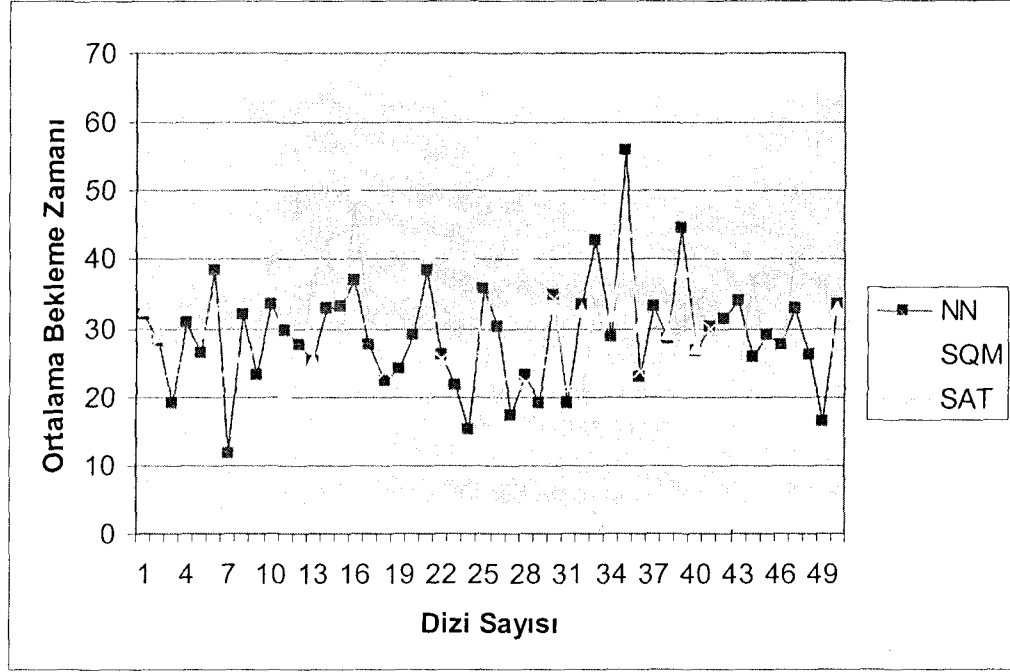
Hafif trafik altında, NN politikası için koşturululan simülasyonun raporu Ek 6'da, SQM politikası için koşturululan simülasyonun raporu Ek 7'de ve SAT politikası için koşturululan simülasyonun raporu Ek 8'de incelenebilir. İlgili simülasyon raporlarına ve bu raporların analizi sonucunda hafif trafik durumu altında alternatif atama politikalarının kıyaslanması Çizelge 5.3'te incelenebilir.



Çizelge 5.3 Hafif trafik ($\lambda=1$) altında atama politikalarının kıyaslanması

İlgili Karar Değişkenleri	NN Atama Algoritması	SQM Atama Algoritması	SAT Atama Algoritması
Günlük Ortalama Müşteri Sayısı	7,82	7,84	7,96
Müşteri Başına Ortalama Bekle Zamanı (W) (dk)	28,92	33,32	28,61
W'nin Alt Sınırı	26,69	30,51	26,72
W'nin Üst Sınırı	31,15	34,13	30,49
Müşteri Başına Ortalama Bekle Zamanının Standart Sapması (s.h (W))	1,11	0,90	0,94
Araç Başına Kat Edilen Günlük Mesafe (km)	20,64	46,06	19,16

Hafif trafik durumu ($\lambda=1$) için SAT atama politikası müşteri başına ortalama bekle zamanına göre NN atama politikasından %1, SQM atama politikasından %14 daha iyi performans göstermiştir. Araç başına kat edilen günlük mesafeye göre, SAT atama politikası NN atama politikasından %7, SQM atama politikasından %58 daha iyi performans göstermiştir. Şekil 5.4'te hafif trafik altında atama algoritmalarının performansı incelenebilir.



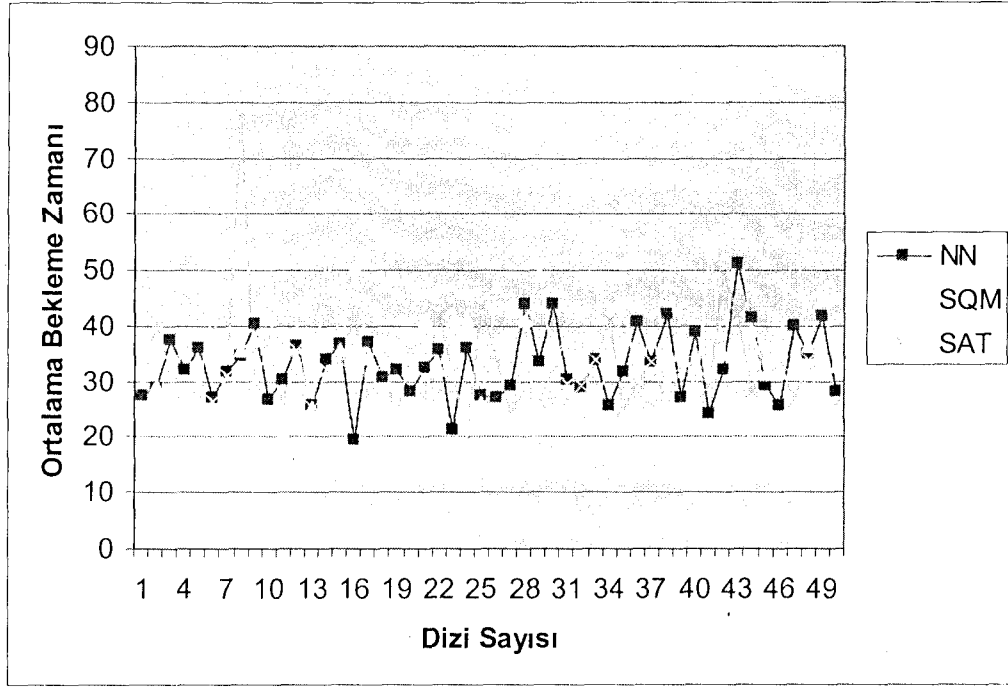
Şekil 5.4 Hafif trafik altında atama algoritmalarının performansı

Az yoğun trafik altında, NN politikası için koşturululan simülasyonun raporu Ek 9'da, SQM politikası için koşturululan simülasyonun raporu Ek 10'da ve SAT politikası için koşturululan simülasyonun raporu Ek 11'de incelenebilir. İlgili simülasyon raporlarına ve bu raporların analizi sonucunda az yoğun trafik durumu altında alternatif atama politikalarının kıyaslanması Çizelge 5.4'te incelenebilir.

Çizelge 5.4 Az yoğun trafik ($\lambda=2$) için atama politikalarının kıyaslanması

İlgili Karar Değişkenleri	NN Atama Algoritması	SQM Atama Algoritması	SAT Atama Algoritması
Günlük Ortalama Müşteri Sayısı	15,82	15,92	15,96
Müşteri Başına Ortalama Bekle Zamanı (W) (dk)	32,84	37,27	32,49
W'nin Alt Sınırı	31,01	34,88	29,98
W'nin Üst Sınırı	34,66	39,67	35,00
Müşteri Başına Ortalama Bekle Zamanının Standart Sapması (s.h (W))	0,91	1,19	1,25
Araç Başına Kat Edilen Günlük Mesafe (km)	47,76	90,96	42,29

Az yoğun trafik durumu ($\lambda=2$) için SAT atama politikası müşteri başına ortalama bekle zamanına göre NN atama politikasından %1, SQM atama politikasından %13 daha iyi performans göstermiştir. Araç başına kat edilen günlük mesafeye göre, SAT atama politikası NN atama politikasından %11, SQM atama politikasından %54 daha iyi performans göstermiştir. Şekil 5.5’da az yoğun trafik altında atama algoritmalarının performansı incelenebilir.



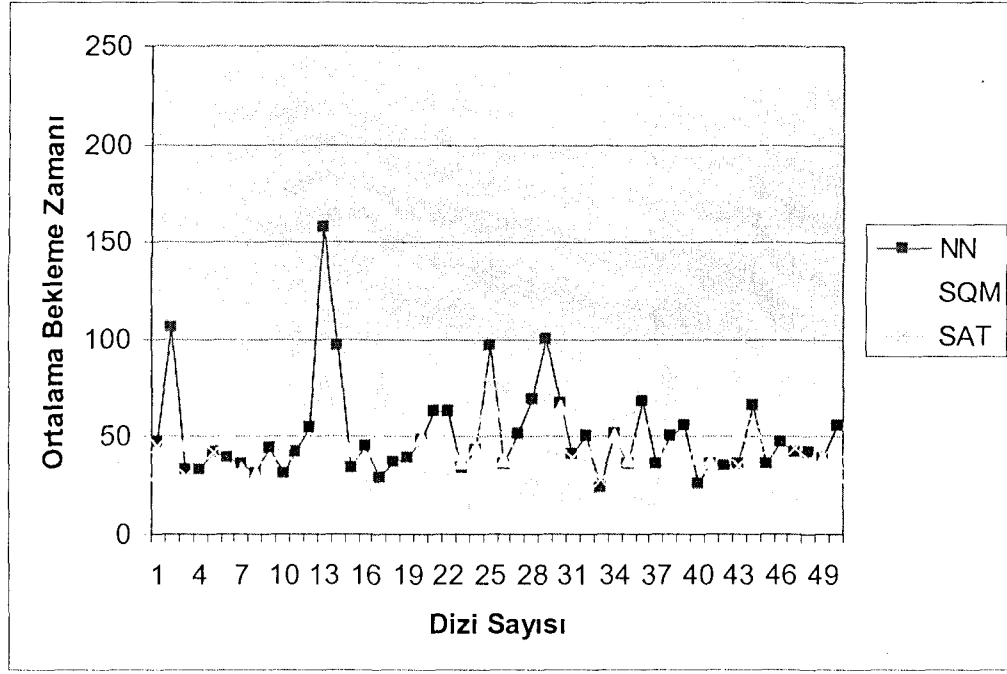
Şekil 5.5 Az yoğun trafik altında atama algoritmalarının performansı

Yoğun trafik altında, NN politikası için koşturululan simülasyonun raporu Ek 12’de, SQM politikası için koşturululan simülasyonun raporu Ek 13’de ve SAT politikası için koşturululan simülasyonun raporu Ek 14’te incelenebilir. İlgili simülasyon raporlarına ve bu raporların analizi sonucunda yoğun trafik durumu altında alternatif atama politikalarının kıyaslanması Çizelge 5.5’te incelenebilir.

Çizelge 5.5 Yoğun trafik ($\lambda=3$) altında atama politikalarının kıyaslanması

İlgili Karar Değişkenleri	NN Atama Algoritması	SQM Atama Algoritması	SAT Atama Algoritması
Günlük Ortalama Müşteri Sayısı	23,94	23,84	23,90
Müşteri Başına Ortalama Bekle Zamanı (W) (dk)	50,43	53,72	45,04
W'nin Alt Sınırı	43,50	43,81	39,66
W'nin Üst Sınırı	57,36	63,64	50,43
Müşteri Başına Ortalama Bekle Zamanının Standart Sapması (s.h (W))	3,45	4,93	2,68
Araç Başına Kat Edilen Günlük Mesafe (km)	89,13	126,85	73,34

Yoğun trafik durumu ($\lambda=3$) için SAT atama politikası müşteri başına ortalama bekle zamanına göre NN atama politikasından %11, SQM atama politikasından %16 daha iyi performans göstermiştir. Araç başına kat edilen günlük mesafeye göre, SAT atama politikası NN atama politikasından %8, SQM atama politikasından %42 daha iyi performans göstermiştir. Şekil 5.6'da yoğun trafik altında atama algoritmalarının performansı incelenebilir.



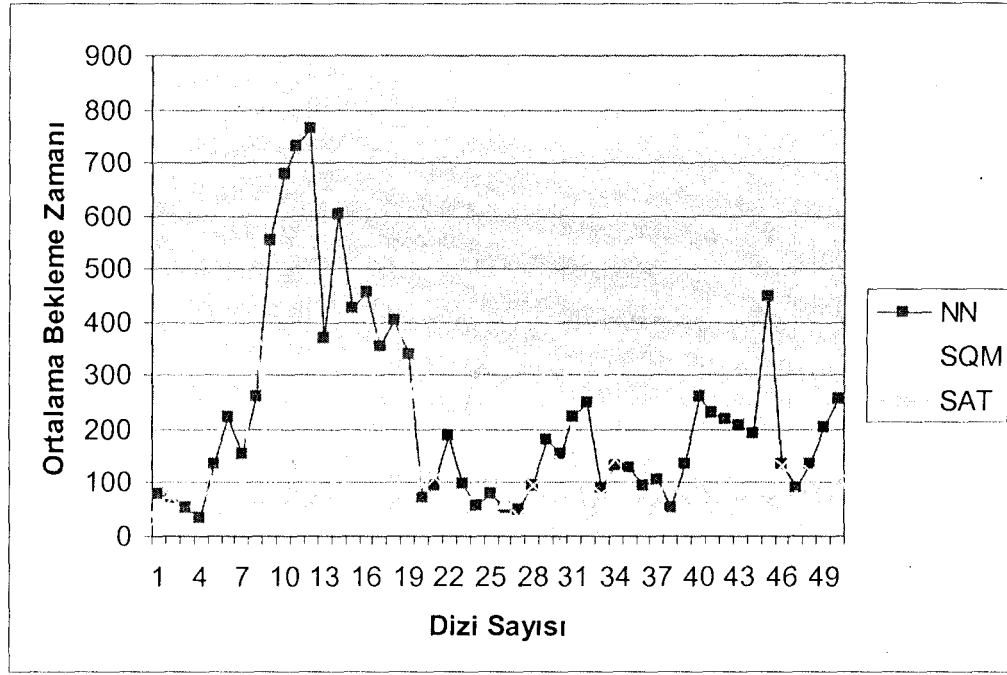
Şekil 5.6 Yoğun trafik altında atama algoritmalarının performansı

Çok yoğun trafik altında, NN politikası için koşturululan simülasyonun raporu Ek 15'te, SQM politikası için koşturululan simülasyonun raporu Ek 16'da ve SAT politikası için koşturululan simülasyonun raporu Ek 17'de incelenebilir. İlgili simülasyon raporlarına ve bu raporların analizi sonucunda çok yoğun trafik durumu altında alternatif atama politikalarının kıyaslanması Çizelge 5.6'da incelenebilir.

Çizelge 5.6 Çok yoğun trafik ($\lambda=4$) altında atama politikalarının kıyaslanması

İlgili Karar Değişkenleri	NN Atama Algoritması	SQM Atama Algoritması	SAT Atama Algoritması
Günlük Ortalama Müşteri Sayısı	31,44	30,78	30,92
Müşteri Başına Ortalama Bekle Zamanı (W) (dk)	227,82	156,99	100,37
W'nin Alt Sınırı	175,01	123,31	82,34
W'nin Üst Sınırı	280,62	190,66	118,41
Müşteri Başına Ortalama Bekle Zamanının Standart Sapması (s.h (W))	26,27	16,75	8,97
Araç Başına Kat Edilen Günlük Mesafe (km)	139,85	141,83	103,84

Çok yoğun trafik durumu ($\lambda=4$) için SAT atama politikası müşteri başına ortalama bekle zamanına göre NN atama politikasından %56, SQM atama politikasından %36 daha iyi performans göstermiştir. Araç başına kat edilen günlük mesafeye göre, SAT atama politikası NN atama politikasından %26, SQM atama politikasından %27 daha iyi performans göstermiştir. Şekil 5.7'da çok yoğun trafik altında atama algoritmalarının performansı incelenebilir.



Şekil 5.7 Çok yoğun trafik altında atama algoritmalarının performansı

Gerçekleştirilen simülasyon koşulları ve yapılan çıktı analizleri sonucunda, SAT atama yaklaşımının NN ve SQM atama yaklaşımlarından hafif, az yoğun, yoğun ve çok yoğun müşteri talepleri altında, 5 araçtan oluşan tamir filosu ile daha başarılı performans gösterdiği ortaya konulmuştur. Dinamik atama politikasının performansı müşteri taleplerinin yoğunluğu arttıkça arttığı gözlenmiştir.

5.6 Bekleme Zamanı ve Kapasite Arasında Tercih

Simülasyon çalışmalarımızı tamamladıktan sonra ulaştığımız istatistikler kısmında, kullandığımız kaynakların miktarına göre değişen farklı ortalama bekleme zamanları ile karşılaştığımıza değinmiştik. Tüm şartlar aynı kalmak üzere kaynaklarımızı (araç sayımızı) arttırdıkça ortalama bekleme zamanlarımızda bir düşüş bekleyebiliriz. Ortalama bekleme zamanı, sistemde servis bekleyen ortalama müşteri sayısı gibi performans ölçüleri ile kapasite arasında doğru dengeyi sağlamak, planlayıcıların en zor tercihlerinden biridir. Özellikle bu dengeyi sağlıklı bir şekilde oturtmak, yatırımlarındaki geri dönüşümün hızla sağlanmasını bekleyen yatırımcılar açısından hayatidir [28].

Bir taraftan, müşteriler fiziksel ya da coğrafi bir kuyrukta beklemekten hoşlanmazlar. Müşteriler eğer çok fazla beklediklerini düşünürler ise, kuyruktan hizmet almadan ayrılabilir ya da bir bir sonraki hizmet ihtiyaçlarında tedarikçilerini değiştirme yolunu seçebilirler.

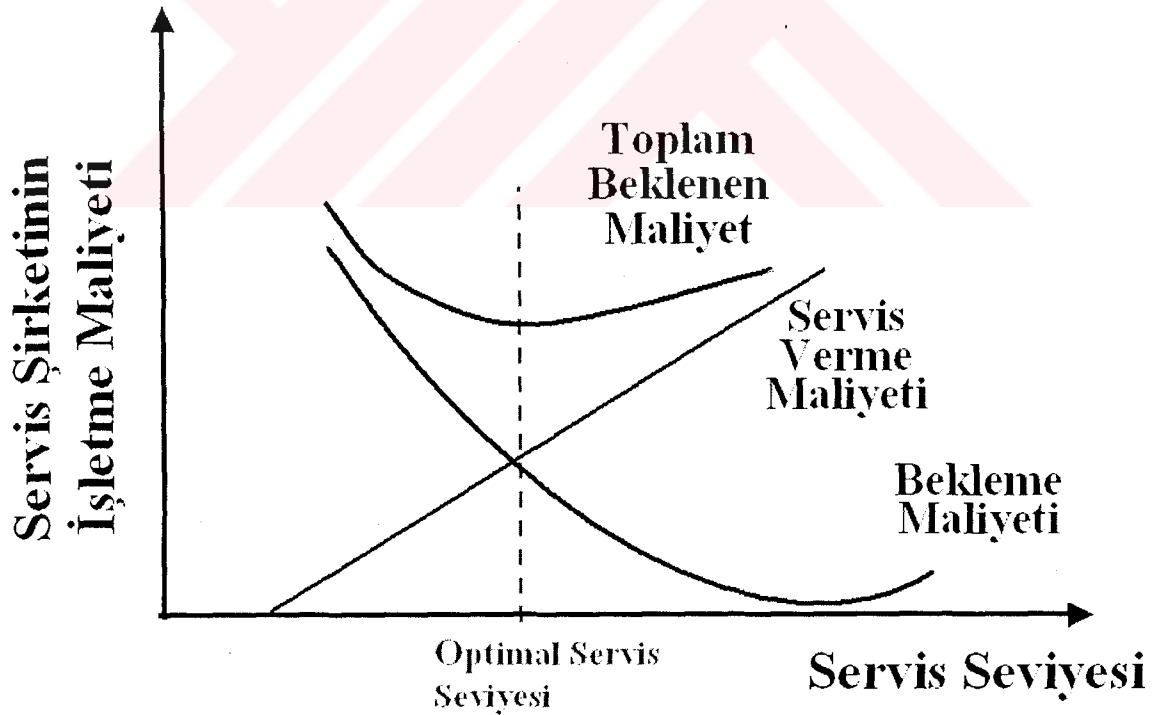
Müşterinin tercihi hangisi olursa olsun, seçimleri müşteri taleplerinin azalmasına dolayısı ile kazanç ve karlılıkta azalmaya neden olur. Diğer bir taraftan, yukarıda değindiğimiz üzere yönetici ve planlayıcılar öncelikle kapasiteyi arttırarak bekleme zamanlarını azaltırlar. Bu yaklaşımın kendisi de oldukça pahalı ve sonuç olarak karlılığı azaltacak bir aksiyondur. Kullanımın göreceli olarak yüksek tutulmasına devam ederken, müşterilerin kabul edilebilir bulunduğu bir bekleme zamanı bulmak, kritik bir operasyonel önem teşkil etmektedir. (Chane vd., 2000)

Servis için talep, bu servis için ayrılmış olan kapasiteyi aşıyorsa, bekleme kaçınılmazdır. Bununla birlikte, proses kullanımı %100'ün altında olduğunda bile ortalama bir bekleme zamanını gözlemleyebiliriz. Bunun birden fazla sebebi olabilir, kapasitenin değişiklik gösterdiği durumda, sistemimiz bağıl olarak yoğun bir talebe, yine bağıl olarak düşük bir kaynak seviyesi ile yakalanmış olabilir. Daha yaygın bir sebep ise, kapasitenin sabit kaldığı durumda talebin dalgalanmasıdır. Kapasitenin ortalama müşteri gelme oranından çok büyük olduğu durumlarda, proses henüz sisteme girmemiş, belki de oluşmamış talepleri “önceden” karşılayamaz. Otomobil, elektronik cihaz tamirleri, acil servis hizmetleri gibi hizmet sektöründe yer alan sistemler, kapasitenin kullanılmadığı zamanlarda, müstakbel müşteri talepleri için “hizmet envanterleri” oluşturamazlar. Geçici olarak servis talebinin kapasiteyi aştığı durumlarda, kuyrukta bekleyen müşteri sayısında ve ortalama bekleme zamanında artış oluşur. Bu bağlamda, ortalama kullanımın % 100'ün altında olduğu sistemlerde, ortalama müşteri için, sıfırdan büyük bir bekleme zamanı ile karşılaşılabilir. Çalışmamızın takip eden kısmında, kapasite kullanımının göreceli olarak yüksel tutulduğu bir sistemde; kapasite tutmanın getirdiği maliyet ile müşterileri bekletmenin maliyeti birlikte değerlendirilip örnek vakamız için “optimal kapasitenin” tespit edilmesine çalışılmıştır.

5.6.1 Kapasite ve Optimal Kapasite Kavramı

Hill (2001)'e göre kapasite, herhangi bir proses için, birim zamandaki adetler çıktısı olarak ölçülen, çıktıların maksimum oranı olarak tanımlanmıştır. Bahsi geçen birim zaman, herhangi bir zaman birimi; gün, vardiya veya dakika olabilir. Kapasitenin tanımı gereği, “maksimum kapasite” ibaresinin kullanımı gereksizdir, zira kapasitenin kendisi zaten maksimumdur. Bazı kaynaklar, nominal kapasite, gözüken kapasite ve teorik kapasite arasında ayrım yaparken, çalışmamızda ele almayı hedeflediğimiz unsur optimal kapasitenin seçimi olduğunun altını çizerek, yine Hill (2001)'e göre optimal kapasiteyi, ilgili toplam maliyeti minimize eden kapasite olarak tanımlayabiliriz.

Herhangi bir sistemde yer alan toplam maliyetin iki temel bileşeni kapasite maliyeti ve bekleme maliyetidir. Optimal kapasiteyi tespit ederken amacımız, toplam maliyetin iki ayrı bileşenini ayrı ayrı minimize çalışmak değil, toplam maliyeti minimize etmektir. Heizer ve Render (2001)'e göre bütün sistemlerin, kapasite kullanımı ile bekleme zamanı arasında bir tercih yapması gerekir. Bu iki değişken arasındaki liner olmayan ilişki ile farklı servis seviyelerinde gözlemlenen toplam işletme maliyeti Şekil 5.4'te irdelenebilir. Kapasite kullanımı % 100'e gittikçe, bekleme zamanı sonsuza gitme eğilimindedir. Bununla beraber kapasite kullanım oranlarını düşürecek önlemler aldıkça, bekleme zamanlarında bir azalma gözlemleyebiliriz. Toplam maliyet penceresinden bakıldığında, daha fazla kapasite kullanıldıkça, servis verme maliyeti artacak, bununla beraber müşterilerin servis bekleme maliyeti azalacaktır. Sistemimizin amacı, kullanım oranını maksimize etmek değildir. Sistemimizin amacı iki ilgili maliyetin toplamını minimize etmektir. Örneğin, Lodi Kenti İtfaiye Teşkilatı (1989)'da irdelenen sistemde yer alan bir itfaiye aracının kapasitesi %100 değildir, %1'e yakındır. Ardışık araştırma prosedürü ile, Şekil 5.8'de gösterilen optimal kapasite ile yürütülebilen optimal servis seviyesine ulaşılmaya çalışılmıştır. (Tanchoco vd., 1987)



Şekil 5.8 Servis seviyesi ile servis şirketinin işletme maliyeti arasındaki ilişki (Heizer ve Render, 2001)

5.6.2 PCServis Şirketi için Amaç Fonksiyonunun Tespit Edilmesi

PCServis Şirketi için optimal servis seviyesinin tespit edilmesi için öncelikli olarak analizimizi bir karar problemi olarak ele alıp, öncül olarak problemimizin amaç fonksiyonunu belirlemeliyiz. Winston (1994)'e göre, karar problemleri temelde iki kısma ayrılır. İlkinde karar verici, tercihini tek bir değişkenden etkilenen muhtemel aksiyonları doğrultusunda yapar. Bir şirketin yatırım yapıp yapmamasına sadece on yıllık şirket karlılığına bakarak yapılması bu tür karar vermeye örnek gösterilebilir. Bununla beraber, pek çok durumda karar verme aksiyonu, birden fazla değişkene bağlıdır. Örneğin, aracımızı değiştirmek istediğimizde, kararımızı etkileyebilecek muhtemel değişkenler; aracın fiyatı, yakıt tüketimi, tamir maliyeti, servis ağı yaygınlığı, stili ve büyüklüğü olabilir.

Birden fazla değişkenin varlığında tercihler arasında karar verilmesini esnasında, öncelikle kararımızı etkileyecek değişkenlerin sayısına karar verilmelidir. Elimizdeki her bir alternatif senaryo için karar verici her bir değişkene bir değer ya da bir maliyet atayabilir.

a ile karar vericinin alternatiflerinin sayısını, n ile karar vericinin kararını etkileyen değişkenler sayısını ve x_i ile her bir i . değişkenin değeri / maliyetini sembolize edildiğinde, her bir alternatif için yazılabilecek olan katkı değer fonksiyonu eşitlik (5.4)'te ve katkı maliyet fonksiyonu da eşitlik (5.5)'te irdenebilir.

$$v(x_1, x_2, \dots, x_n) = \sum_{i=1}^{i=n} v_i(x_i) \quad (5.4)$$

$$c(x_1, x_2, \dots, x_n) = \sum_{i=1}^{i=n} c_i(x_i) \quad (5.5)$$

Karar verici elindeki alternatiflerinden, değeri maksimize eden alternatif ya da maliyeti minimize edecek alternatif seçilebilir. Çalışmamızda PCServis Şirketinin tercih etmesi önerilen optimal servis seviyesi toplam maliyetlerini minimize edecek alternatifin bulunması sureti ile gerçekleştirilmesine çalışılmıştır. Bu doğrultuda amaç fonksiyonumuz eşitlik (5.6)'da incelenebilir.

$$\min_{a \in A} c(x_1(a), x_2(a), \dots, x_n(a)) = c(x_1^*, x_2^*, \dots, x_n^*) \quad (5.6)$$

5.6.3 PCServis Şirketi için İşletme Maliyet Merkezlerinin Tespit Edilmesi

PCServis şirketi için optimal kapasiteyi, mevcut alternatifler arasında minimum toplam maliyet ile sonuçlandırabileceğimiz alternatifi belirlemek için öncelikle karar vericinin

kararını etkileyen değişkenlerin yani maliyet merkezlerinin tespit edilmesi gerekmektedir. PCServis Şirketi için Şekil 5.1’de değindiğimiz, PCServis Şirketinin öğeleri arasındaki ilişkinin ışığında, şirketin maliyet kalemlerini, ofis kira, ısınma, elektrik, telefon giderleri, memurların, araç programlayıcısının ve tamircilerin maaşları, araçların alım, onarım, yakıt, kasko maliyetleri ve yararlanılan teknolojilerin sabit ve kullanım maliyetleri olarak sınıflandırılabilir. Yapılan sınıflandırma dahilinde yukarıdaki maliyet kalemlerinin bazıları karar vericinin tercih ettiği kapasite alternatifinden yani sistemin işleyeceği toplam tamirci sayısına bağımlı olmadığı (örneğin, ofis kira, ısınma, elektrik, telefon giderleri, memurların, araç programlayıcısının maaşları gibi) görülebilir. Sistemde yer verilen tamirci sayısından etkilenebilecek maliyetler ise tamircilere ödenen maaş, tamircilerin kullandığı araçların alım, onarım, yakıt, kasko maliyetleri ve yararlanılan teknolojilerin lisans ve kullanım maliyetleri olarak sıralanabilir. Çalışmamızın dördüncü kısmında yararlanılan teknolojilerin sabit ve kullanım maliyetleri tamircilere ödenecek maaş ve araçların toplam maliyetinin yanında ihmal edilebilecek düzeyde kaldığından maliyet fonksiyonu araç ve tamircilerin sabit ve değişken maliyetler üzerine kurulmuştur. Tamircilere verilen maaş sabit bir maliyettir ve tamircilerin yanıt verdiği müşteri sayısında, kat ettikleri kilometreden bağımsızdır. Ek olarak, tamircilerin kullandığı araçların bir satın alma maliyeti vardır ve bu maliyet sabittir. Bununla birlikte araçların kat edilen mesafenin bir fonksiyonu olabilecek yakıt, lastik ve bakım maliyetleri ile kullanıldığı zaman periyodu ile ilişki yerel vergileri ve kasko maliyetleri bulunmaktadır. Takip eden kısımda öncelikli olarak tamircilerin kullanacağı araç modeli seçilmiş olup, seçilen aracın maliyet fonksiyonunda konumlandırılması gerçekleştirilmiştir.

5.6.3.1 PCServis Şirketinin Araç Tercihi ve İlgili Maliyetin Hesaplanması

PCServis Şirketinin önemli maliyet kalemlerinden birini de tamircilerin kullandığı araçlar teşkil eder. Aracın sabit ve değişken maliyetinin öngörülmesinde tercih edilen aracın spesifikasyonlarının önemi büyüktür. Bununla birlikte tercih edilen aracın tipi, sahip olduğu özellikler ile maliyet fonksiyonunu ciddi ölçüde etkilemesi beklenebilir. PCServis Şirketi için “doğru” araç tipini seçmek de kendi içinde bir karar problemi olarak ele alınabilir. Aracın satın alma fiyatı, yakıt tercihi, yakıt tüketim seviyesi araç seçimi sırasında kararı etkileyebilecek değişkenler olarak sıraladığımızda, farklı araç modelleri için aynı kullanım oranı ve zamanı ile araçların şirkete maliyetlerini bugünkü değer prensibine göre hesaplamaya çalıştık. Kıyaslama, yerel üretim araçların içerisinde benzin ve dizel yakıt kullanan en hesaplı alternatifleri arasında gerçekleştirilmiştir. İlgili kıyaslama Çizelge 5.7’de incelenebilir.

Çizelge 5.7 Alternatif araç modelleri ve herbir modelin şirkete üç yıllık maliyeti

	Alternatif Araç Modelleri			
Maliyet ve Özellikleri	Albea 1.3 Multijet SL/EL [38]	Albea 1.2 SL [38]	Clio Symbol Authentique 1.5 dCi [30]	Clio HB Authentique 1.2 16V [30]
Araçın satın alma fiyatı (bd) (A)	\$19.304	\$17.516	\$19.511	\$17.873
Kat edilen km (günlük)	200	200	200	200
Kat edilen km (yıllık)	40000	40000	40000	40000
Yıllık yakıt maliyeti	\$2.524	\$4.923	\$2.524	\$4.220
3 yıllık yakıt masrafının bd'si (B)	\$7.208	\$14.061	\$7.208	\$12.052
toplam maliyet (bd) (A+B)	\$26.512	\$31.577	\$26.719	\$29.925
Şehiriçi yakıt harcama lt / 100km [13] ve [30]	6	9,1	6	7,8
Yakıt harcama lt / km	0,06	0,091	0,06	0,078
Yakıt maliyeti usd / km	\$0,063	\$0,123	\$0,063	\$0,105

Çizelge 5.7’de alternatif araç modellerinin tavsiye edilen satış fiyatlarından, araçların bugünkü değeri (bd) 1 Amerikan Doları, 1.329.357 TL alınarak hesaplanmıştır. Araçların

yakıt harcama seviyelerinin kıyaslanmasında, araçların şehriçi yakıt harcama indislerinden yararlanılmış, Albea 1.3 Multijet SL/EL ve Clio Symbol Authentique 1.5 dCi araçları için motorin (1 lt motorin bedeli 1.398.000 TL) ; Albea 1.2 SL ve Clio HB Authentique 1.2 16V için ise benzinin (1 lt benzin bedeli 1.798.000 TL) pompa fiyatları ele alınmış ve ilgili kurdan Amerikan Dolarına çevrilmiş, böylelikle her aracın yakıt maliyetine (usd / km) ulaşılmıştır [34].

PCServis Şirketinin tamircilerinin bir iş gününde 200 kilometre kat edeceği öngörülmüş olup bir yılda 40.000 kilometre kat edeceği göz önüne alınmıştır. Kat edilen yıllık kilometre ve yakıt maliyeti (usd/km) bilgilerinden her aracın yıllık yakıt maliyetine ulaşılmış ve eşitlik (5.7)'den yararlanarak üç yıllık yakıt masrafının bugünkü değeri (bd), Amerikan Dolarına %3 yıllık faiz oranı (% 0.25 aylık) göz önüne alınarak hesaplanmıştır [20].

$$\text{Bugünkü Değer} = (\text{Dönemlik Faiz Oranı; Dönem Sayısı; Yıllık Yakıt Maliyeti; 0; 0}) \quad (5.7)$$

Araçların hesaplanan üç yıllık yakıt maliyetinin bugünkü değeri ile aracın satın alma bedelinin bugünkü değeri toplanarak toplam maliyet hesaplanmıştır. Çizelge 5.7'den de görüldüğü üzere toplam \$ 26.512 minumum maliyet, \$ 19.304 satın alma bedeli ve \$ 0,063 kilometre başına yakıt bedeli ile Albea 1.3 Multijet SL/EL PCServis şirketinin tamircileri için seçtiği araç olmuştur.

5.6.3.2 PCServis Şirketinin Günlük Operasyonel Maliyetinin Belirlenmesi

PCServis şirketinin operasyonel maliyetlerini, bir önceki kısımda tercihini gerçekleştirdiğimiz şirket aracının maliyetleri ve tamircilere ödenen ücret oluşturmaktadır. Araç ile ilgi maliyetler; kullanım miktarından bağımsız, aracın kullanıldığı zaman periyoduna bağlı sabit maliyetler (aracın amortismanı), kullanım miktarından bağımsız aracın yıllık maliyetler (trafik pul bedeli, zorunlu trafik sigortası ve kasko) ve araç ile kat edilen kilometreye bağlı değişken maliyetler (akaryakıt, bakım ve lastik) olarak sınıflandırabiliriz. Şirketin tamircilerine ödediği maaşı asgari ücret kabul edildiği takdirde, PCServis Şirketinin bahsedilen maliyet kalemleri, Çizelge 5.8 (sabit maliyetler) ve Çizelge 5.9'te (değişken maliyetler) detayları ile birlikte incelenebilir.

Çizelge 5.8 PCServis Şirketinin sabit operasyonel maliyetleri

Maliyet ve Özellikleri	Yıllık Sabit Maliyetler (usd)	Aylık Sabit Maliyetler (usd)	Günlük Sabit Maliyetler (usd)
Tamircinin Maaşı [10]	\$5.076	\$423	\$21,15
Aracın Amortismanı	\$5.460	\$455	\$22,75
Aracın Kasko Bedeli [33]	\$562	\$47	\$2,34
Aracın Zorunlu Trafik Sigortası Bedeli [33]	\$135	\$11,25	\$0,56
Aracın Vergisi	\$300	\$25	\$1,25
Toplam Sabit Maliyet (usd)	\$11.533	\$961,25	\$48,06

Aracın amortismanının hesaplanmasında aylık yaklaşım esas alınmıştır. Aracın kullanılacağı zaman periyodu 3 yıl olarak planlanmış olup, 3 sene sonunda aracın hurdası \$ 4.000'a satılabileceği öngörülmüştür. Aracın bugünkü değeri (bd), \$19.304, kullanılarak aracın aylık amortisman bedeline eşitlik (5.8) ve (5.9)'dan yararlanılarak ulaşılmıştır.

$$\text{Devresel_Ödeme} = (\text{Dönemlik Faiz Oranı; Dönem Sayısı; Aracın Bugünkü Değeri; Aracın Gelecekteki Değeri; 0}) \quad (5.8)$$

$$\text{Devresel_Ödeme} (0,0025; 36; \$19.304; \$4,000; 0) = \$ 455 \quad (5.9)$$

Çizelge 5.8'de yer alan değerlerden, aracın aylık amortisman bedeli maliyeti ve tamircinin aylık maaşı 12 ile çarpılarak yıllık sabit maliyet, şirketin operasyonel olarak aktif olduğu 20 gün baz alınıp, 20'ye bölünerek de günlük sabit maliyet bulunmuştur. Aracın kasko, zorunlu

trafik sigortası ve vergi bedelleri ise yıllık sabit maliyetler olarak ele alınıp, bu değerler 12'ye bölünerek aylık maliyetlere, aylık sabit maliyetler de 20'ye bölünerek günlük sabit maliyetlere ulaşılmıştır.

Çizelge 5.9 PCServis Şirketinin değişken operasyonel maliyetleri

Maliyet ve Özellikleri	Toplam Maliyet (usd)	Maliyetin Gerçekleştiği Toplam Mesafe (km)	Değişken Maliyet (usd /km)
Akaryakıt Maliyeti	\$ 0,063	1	\$ 0,063
Bakım Maliyeti	\$ 120	15.000	\$ 0,008
Lastik Değiştirme Maliyeti [29]	\$ 300	50.000	\$ 0,006
Toplam Değişken Maliyet (usd/km)			\$ 0,077

Çizelge 5.9'da tercih edilen aracın kullanım miktarından etkileneceği öngörülen değişken operasyonel maliyetler verilmiştir. Aracın değişken maliyetleri kat edilen kilometreye bağlı olarak ele alınmış olup, akaryakıt maliyeti Çizelge 5.6'da yer verilen ilgili değer olup, toplam bakım ve lastik değiştirme maliyetleri her iki maliyetin gerçekleşeceği öngörülen mesafeye bölünerek kilometre başına bakım ve lastik değiştirme maliyetlerine ulaşılmıştır.

Çizelge 5.8 ve çizelge 5.9 ile PCServis Şirketinin operasyonel maliyetleri günlük sabit maliyetler ve kilometre başına değişken maliyetler olarak incelenmişti. Bu doğrultuda, aracın bir günlük sabit maliyeti \$ 48,06 ve kilometre başına değişken maliyeti ise \$ 0,077 olarak hesaplanmıştı. Hesaplanan operasyonel maliyetlerden, PCServis Şirketi için optimal kapasitenin tespit edilmesinde yararlanılmıştır.

5.6.4 PCServis Şirketi İçin Optimal Kapasitenin Tespit Edilmesi

Çalışmamızda çeşitli müşteri talepleri seviyeleri için tartışılan simülasyon sonuçlarının sonucunda, SAT yaklaşımının alternatif atama politikalarına daha performanslı sonuçlar

verdiğini incelemiştik. Bu kısımda ise $\lambda=3$ durumu için, SAT politikası altında sistemimiz farklı araç kapasiteleri ile performansını daha önceki kısımlardaki hesaplamalarımız doğrultusunda incelenmiştir.

Öncelikle toplam günlük maliyeti etkileyen bileşenlerden sabit operasyonel maliyetleri için Çizelge 5.7’de hesaplanan \$ 45,06 /araç-gün, değişken operasyonel maliyetleri için Çizelge 5.8’de hesaplanan \$ 0,077/km ve müşteriye bekletme maliyeti olarak da \$ 10/saat [1] kullanılmıştır.

Her bir farklı kapasite durumu için yoğun trafik altında, SAT politikası doğrultusunda koşturulan simülasyonların raporları Ek 18’de 3, Ek 19’da 4, Ek 20’de 5, Ek 21’de 6, Ek 22’de 7 ve Ek 23’de 8 araçtan oluşan filolar için incelenebilir. İlgili raporlar doğrultusunda her bir kapasite durumu için PCServis Şirketinin günlük toplam maliyetleri Çizelge 5.10’da gösterilmiştir.

Çizelge 5.10 PCServis Şirketinin işlettiği farklı sayıda araç için günlük toplam maliyetleri

Maliyet ve Özellikleri	Toplam Araç / Tamirci Sayısı					
	3	4	5	6	7	8
Günlük Ortalama Müşteri Sayısı (A)	23,82	23,94	23,9	23,76	23,92	23,88
Müşteri Başına Ortalama Bekleme Zamanı (dakika) (B)	1615,35	94,27	45,82	32,24	29,32	26,04
Toplam Müşteri Bekleme Zamanı (saat)						
(C = AxB/60)	641,29	37,61	18,25	12,77	11,69	10,36
Bekleme Zamanı Maliyeti (usd / saat) (D)	10	10	10	10	10	10
Toplam Bekleme Maliyeti (usd) (E= CxD)	6.413	376	183	128	117	104

Bir aracın ve Tamircinin Sabit Maliyeti (usd) (F)	48,06	48,06	48,06	48,06	48,06	48,06
Toplam Sabit Maliyetler (usd) (G)	144	192	240	288	336	384
Kat edilen Toplam Günlük Mesafe (km) (H)	461,64	416,56	366,68	310,78	275,84	246,29
Birim Mesafe İçin Değişken Maliyet (usd/km) (I)	0,077	0,077	0,077	0,077	0,077	0,077
Toplam Değişken Maliyet (usd) (J=HxI)	36	32	28	24	21	19
Toplam Günlük Maliyet (usd) (K=E+G+J)	6593	600	451	440	475	507

Çizelge 5.10’da verilen maliyet analizi sonucunda, PCServis Şirketinin SAT yaklaşımı altında günlük toplam maliyetini minimize edecek optimal araç (kapasite) sayısının, günlük \$ 440 maliyet ile, 6 olduğu tespit edilmiştir. Bu doğrultuda şirketin mevcut araç filosuna bir araç daha kazandırarak operasyonlarını 6 araç ile yürütmesi önerilebilir.

6. SONUÇ VE ÖNERİLER

Çalışmamızda araç atama problemlerinin pek çok yönü tartışılmıştır. Öncelikle geleneksel ve dinamik araç atama problemlerinin tanımları ve birbirlerinden farkları ele alınmış takip eden kısımda da dinamik araç atama ve rotalama problemlerinin günlük hayatta uygulanabileceği alanlar sıralanmıştır. İlgili alanların sayı olarak çokluğu, kapsam olarak genişliği ve önem olarak önceliği bize çalışmamız süresince ek motivasyon sağlamıştır.

DAÇP üzerine gerçekleştirilen literatür taraması ile sadece matematik programlama ve kuyruk teorisi öğretileri ile gerçek yaşama dair hayati kısıtları kaldırmadan probleme yanıt vermenin pek de mümkün olmadığını altı çizilmiştir. Bertsimas ve Van Ryzin (1991) tarafından servis alanında düzgün dağılmış talepler ile kapasite kısıtı olmayan ve sabit bir hızla hareket eden tek bir araç için ortaya konan sonuçlar, Bertsimas ve Van Ryzin (1993) tarafından geliştirilmiş; servis alanında düzgün dağılmış taleplere yanıt vermek için sabit hız ile seyir eden çok araçlı ve kapasite kısıtlarının olduğu duruma cevap verebilir hale getirilmiştir. DGTP’de yolculuk zamanları bağımsız değişken olarak ele alınamadığından, elde edilen sonuçlar, bekleme zamanları için alt sınırların tespit edilmesi şeklinde gerçekleştirilebilmiştir. Bahsi geçen alt sınırlara Xu (1994) tarafından katkı yapılmış ancak sadece tek araçlı sistem için ve yoğun trafik altında ispatlanabilmiştir.

Çalışmamızda olay oluşmadan önce tam olarak kestirilemeyen (kesin olmayan), servis alanında düzgün dağılmayan, dinamiklik derecesi yüksek, müşteri taleplerine yanıt verebilecek bir araç atama politikası ortaya konulmaya gayret edilmiştir. Araç filomuzdan maksimum verimi almayı ve müşteri bekleme zamanını minimize etmeyi hedefleyen yaklaşımımıza En Kısa Ulaşım Zamanı (Shortest Arrival Time- SAT) adı verilmiştir. Bu yaklaşım müşteri taleplerinin olduğu noktalar ile sistemde görevli araçların anlık konumlarını, iş yüklerini tespit edebilecek ve araçlar ile merkez arasında bilgi alışverişini sağlayabilecek teknik ve teknolojik altyapılara gereksinim duymaktadır. Bahsi geçen altyapılar sırası ile pozisyonlama teknolojisi (GPS), iletişim teknolojisi (SMS ve GPRS) ve CBS’dir. Geliştirdiğimiz atama politikasının uygulanabilirliğini mümkün kılan teknolojik altyapılarının kısaca tanıtılması ve ilgili maliyet analizleri ile bu altyapıların fizibilitesinin tartışılması çalışmamızın üçüncü kısmında gerçekleştirilmiştir.

Literatürde sadece sabit hızlarda seyreden araçlar için ele alınan DGTP’ye ilk ciddi katkımızı İstanbul Kenti için bir yolculuk zamanı modeli ortaya atarak gerçekleştirdik ve tartışmalarımızı gerçek hayat ile uyumlu ve anlamlı sonuçlar vermesini beklediğimiz gerçekçi

bir model ile yürütmeye odaklandık. Bu çalışmalar sırasında, doğrudan ölçüm metodu ve literatürde yer alan çeşitli yöntemler ile tespit edilmiş sonuçlar kullanılmış olup yolculuk mesafesine bağlı bir yolculuk zamanı modeli önerilmiş ve önerilen model, performansı kıyaslanan politikalar dahilinde ilk defa uygulanmıştır.

Çalışmamızın son kısmında, atama yaklaşımının performansı, literatürde kendilerine yer bulmuş iki adet alternatif politika ile (NN ve SQM) simülasyon modeli oluşturularak farklı trafik yoğunlukları (hafif, az yoğun, yoğun ve çok yoğun) altında kıyaslanmıştır.

Her dört trafik yoğunluğunda da hem ortalama bekleme zamanına göre, hem de kat edilen mesafeye göre en yüksek performansı SAT politikası sağlamıştır. Hafif ve az yoğun trafikte SAT politikası ortalama bekleme zamanına göre NN politikasından sadece %1 daha iyi performans sağlarken; SQM politikasından %13-14 daha iyi performans göstermiştir. Yine aynı trafik yoğunluklarında SAT politikası kat edilen mesafeye göre NN politikasından sadece % 7-11 daha iyi performans sağlarken; SQM politikasından %42-54 daha iyi sonuç vermiştir. Yoğun trafik altında bekleme zamanına göre SAT politikasının, NN politikasına göre daha belirgin bir performans vermiş olup, SQM politikasının sonuçları da NN politikasının sonuçlarına yaklaşmıştır. Bununla beraber kat edilen mesafeye göre SQM politikasının performansı hala SAT ve NN politikasından zayıftır. Çok yoğun trafik altında SAT politikasının performansı bağıl olarak zirveye çıkmış, bekleme zamanına göre NN'den %56, SQM'den ise %36 daha iyi sonuçlar vermiştir. Kat edilen mesafeye göre ise NN'den %26, SQM'den ise %27 daha iyi sonuçlar vermiştir.

Yapılan simülasyon koşumları doğrultusunda yukarıda özetlenen sonuçlara dayanarak, her trafik yoğunluğunda SAT algoritmasının hem bekleme zamanını hem de kat edilen mesafeyi minimize etmek için kullanılması önerilebilir. Bununla beraber trafik yoğunluğu arttıkça yaklaşımımızın bağıl performansında ilerleme gözlemlenmiştir. Alternatif yaklaşım olarak hafif, az yoğun ve yoğun trafik altında NN, çok yoğun trafik altında ise SQM politikaları göz önüne alınabilir.

Çalışmamızın takip eden bölümünde, işleyen bir sistem için optimal kapasite (araç sayısı) tartışmaları yürütülmüş, belirlenen maliyet merkezleri doğrultusunda, yoğun trafik altında ve SAT araç atama politikası dahilinde, ele aldığımız vaka için günlük toplam maliyeti minimize etmesi öngörülen araç sayısı 6 olarak tespit edilmiştir.

Araştırmamız dahilinde simülasyon modelleri oluşturulurken, ilgili modellerin, araçların kapasite kısıtına sahip olabilecek, servis alanının boyutları revize edilebilecek ve servis

alanındaki taleplerin dağılımı değiştirilebilecek yapıda kurulmasına önem verilmiştir. Farklı sistemlerin de incelenmesini mümkün kılan simülasyon modelimizde bahsi geçen özellikler parametre olarak tanımlanmış olup; ilgili parametreler, değişken haline getirilip başka bir çalışmanın konusunu teşkil edebilir.

DGTP'ini literatüre kazandıran araştırmalarda sadece tek performans ölçütü (ortalama bekleme zamanı) ele alınmıştır. Yürüttüğümüz çalışmalarda, ortalama bekleme zamanı performans ölçütü ile birlikte kat edilen mesafeyi de minimize etmek hedeflenmiştir. Takip eden çalışmalarda sonuçları kıyaslanan politikaların, araçların iş yüklerinin dağılımına etkileri de incelenebilir. Minimum ortalama bekleme zamanı, kat edilen minimum mesafe ve araçların iş yüklerinin en adil dağılımı amaç programlama dahilinde tartışılabilir ve elde edilecek bulgular ile çalışma hem araç hem de ekip çizelgeleme problemine çözüm üretebilecek bir noktaya getirilebilir.

Kendisi başlı başına bir çalışma olarak ele alınabilecek olan İstanbul Kenti için yolculuk zaman modeli daha detaylı bir şekilde irdelendiğinde, elde edilebilecek sonuçlar doğrultusunda ilgili simülasyon modellerinin tekrar koşturulması gerçek hayat ile daha tutarlı sonuçlar elde etmek adına önerilebilir.

KAYNAKLAR

- Banks, J., Carson, J.S. ve Nelson, B.L. (1996), Discrete-Event System Simulation, 2nd edition, Englewood Cliffs, N. J.: Prentice Hall.
- Başılgil, H., (1988), Sıralama ve Programlama, Eğitim Yayınları, İstanbul.
- Bartholdi, J.J. ve Platzman L.K., (1988), "Heuristics Based on Spacefilling Curves for Combinatorial Problems in Euclidean Space", Management Science, 34 (3): 291-305.
- Berry, D.S. ve Gren, F.H., (1949), "Techniques for Measuring Over-All Speeds in Urban Areas.", In Proceedings. Highway Research Board, National Research Council, Volume 28.
- Berry, D.S., (1952), "Evaluation of Techniques for Determining Over-All Travel Time", In Proceedings. Highway Research Board, National Research Council, Volume 31.
- Bertsimas, D. ve Van Ryzin, G., (1991), "A Stochastic and Dynamic Vehicle Routing Problem in the Euclidean Plane", Operations Research, 39 : 601-615.
- Bertsimas, D. ve Van Ryzin, G., (1993), "A Stochastic and Dynamic Vehicle Routing Problem in the Euclidean Plane with Multiple Capacitated Vehicles", Operations Research, 41 : 60-76.
- Brabyn, L. ve Skelly, C., (2002), "Geographical Access to Services, Health (GASH): Modelling Population Access to New Zealand Public Hospitals", International Journal of Health Geographics 2002, 1:3.
- Brande-Lavridsen, H., (2002), "Spatial Management in Denmark", The Danish Way, April 2002, Publication no:9.
- BRW Inc., (2000), "Travel Time Data Collection Using Global Positioning System Technology", Puget Sound Regional Council, August.
- Chane, R., Aquilano, N. ve Jacobs, R., (2000), "Operations Management For Competitive Advantage", 9th edition, Technical Note 6: Waiting Line Management. The McGraw-Hill Higher Education.
- Clapp, J. ve Rodriguez, M., (1998), "Using a GIS for Real Estate Market Analysis: The Problem of Spatially Aggregated Data", Journal of Real Estate Research, Volume 16, Number 1.
- Erkut, H., (1995a), "Sistem Yönetimi", Yönetim Bilimleri Dizisi:4, İrfan Yayınları, İstanbul.
- Erkut, H., (1995b), "Yönetimde Sistem Yaklaşımı", İrfan Yayınları, İstanbul.
- Fitzpatrick, R., (1989), Chicago Fire Department Management Study: A Case Study of Distance Versus Travel Time, National Fire Academy, Executive Development Research Paper, Emmitsburg, MD.
- Goodchild, M.F., Haining, R. ve Wise, S. (1992), "Integrating GIS and spatial data analysis: problems and possibilities", International Journal of Geographical Information Systems, 6(5): 407-423.
- Granito, J. ve Dionne, J., (1998), Evaluating Community Fire Protection, Managing Fire Services, International City Management Association, Washington, D.C.
- Globerson, S., (1985), "Issues in Developing a Performance Criteria System for an

Organization", *International Journal of Production Research*, 23 (4).

Gregory, N. ve Platts, K., (1995), "Performance Measurement System Design: A Literature Review and Research Agenda", *International Journal of Operations and Production Management*, 15(4): 80-116.

Harpell, J.L., Lane, M.S. ve Mansoura, A.H. (1989), "Operations Research in Practice", *Interfaces* Volume 19, Number 3, May/Jun: 65-74.

HCM, (1994), Special Report 209, Transportation Research Board, Washington, D.C.

Heizer, J.H. ve Render, B., (2001), *Principles of Operations Management*, 4th Edition, Prantice-Hall Inc., Upper Saddle River, N.J.

Hoban, C., Reilly, W. ve Archondo-Callao, R., (1994), *Economic Analysis of Road Projects with Congested Traffic*, World Bank Publications, Washington D.C.

Horald, J.L., (1982), *Introduction to Probability Theory and Statistical Inference*, 3rd Edition, Wiley, New York.

Howard, R.A., (1966), "Decision Analysis: An Applied Decision Theory", in *Proceedings of Fourth International Conference on Operations Research*, Wiley, New York.

Hudges, J., (2002), "AIMSUN2 Simulation of Congested Auckland Freeway", 6th EURO Working Group on Transportation Göteborg, Sweden.

Kokaz, K., Harshadeep, N.R., Rogers, P. ve Srinivasan, S., (2001), "A Simulation Analysis of Transportation Policies on Health and Environment in Delhi, India", *Computers, Environment and Urban Systems* 2002 Fortcoming.

Landolfi, D., (1997), *Determining Station Location for a Combined Arff/structural Response*, National Fire Academy, Applied Research Project, Emmitsburg, MD.

Langley, R.B., (1997), "GLONASS: review and update" *GPSWorld* 8(7):46-55.

Larsen, A., (2001), "The Dynamic Vehicle Routing Problem", Ph.D-thesis, IMM-PHD-73, IMM 2000.

Larson, R.C. ve Odoni, A.R., (1981), *Urban Operations Research*, Prentice-Hall, Englewood Cliffs.

Law, A.M. ve Carson, J.S., (1982), "A Sequential Procedure for Determining the Length of a Steady-State Simulation.", *Operations Research*, Vol.27:1011-1025.

Lund, K., Madsen, O.B.G. ve Rygaard, J.M., (1996), "Vehicle Routing Problems with Varying Degrees of Dynamism", Technical Report, IMM, The Department of Mathematical Modelling, Technical University of Denmark.

Maria, A., (1997), "Introduction to Modeling and Simulation", WSC'97, Proceedings of 1997 Winter Simulation Conference, December 7-10, 1997, Atlanta, GA, USA.

Naylor, T.H. ve Finger, J.M., (1967), "Verification of Computer Simulation Models", *Management Science*, Vol. 2: B92-B101.

Pegden, C.D., Shannon R.E. ve Sadowski R.P., (1995), "Introduction to Simulation Using SIMAN", 2nd edition, McGraw-Hill, Inc., New York, NY.

- Powell, W.B., Jaillet, P. ve Odoni, A., (1995), "Network Routing", Volume 8, Chapter Stochastic and Dynamic Networks and Routing: 141-295, Elsevier Science, Amsterdam.
- Psaraftis, H.N., (1980), "A Dynamic Programming Solution to the single Many-to-many Immediate Request Dial-a-Ride Problem", *Transportation Science*, 14:130-154.
- Psaraftis, H.N., (1988), "Vehicle Routing: Methods and Studies", 223-248, Elsevier Science Publishers B.V. (North Holland).
- Psaraftis, H.N., (1995), "Dynamic Vehicle Routing: Status and Prospects", *Ann. Of Operations Research*, 61:143-164.
- Rasmussen, J.J. ve George, T., (1978), "After 25 years: A Survey of Operations Research Alumni", *Interfaces* Volume 8, Number 3, May/Jun: 48-52.
- Schmeiser, B., (1982), "Batch Size Effects in the Analysis of Simulation Output.", *Operations Research*, Vol.30: 556-568.
- Schriber, T.J., (1987), "The Nature and Role of Simulation in the Design of Manufacturing Systems" in *Simulation in CIM and Artificial Intelligence Techniques*, J. Retti and K.E. Wichmann (eds.), Society for Computer Simulation, 5-18.
- Shannon, R.E., Long, S.S. ve Buckles, B.P., (1980), "Operations Research Methodologies in Industrial Engineering", *AIIE Transactions*, Vol:12, No:4: 364-367 .
- SDHPT, (1989), "1988 Annual Report: Permanent Automatic Traffic Recorders", Texas Department of Transportation.
- Sybesma, P., (1995), "An Equation for Station Location", *Fire Chief*, 39.
- Tanchoco, J.M.A., Egbelu, P.J. ve Taghaboni, F., (1987), "Determination of the Total Number of Vehicles in a AGV-Based Material Transport System", *Material Flow*, 4: 33-51.
- Taylor, R.G., (1998), "Impact of a Second Firehouse on Respond Time", (Executive Development Research Paper) Emmitsburg, MD: National Fire Academy.
- Thomas, G. ve Decosta, J., (1979), "A Sample Survey of Corporate Operations Research", *Interfaces* Volume 9, Number 4, Jul/Aug: 102-111.
- The Department of Housing and Urban Development, (1977), *Deployment Methodology for Fire Departments*, US Government Printing Office, Washington D.C.
- Thompson, T., (1998), "Fire Station Relocation", (Executive Development Research Paper) Emmitsburg, MD: National Fire Academy.
- Turner, S.T., Eisele W.L., Benz R.J. ve Holdener D.J., (1998) "Travel Time Data Collection Handbook", Texas Transportation Institute Research Report 070470-1F, FHWA-PL-98-035.
- Walker, W., (1979), *Fire Department Deployment Analysis*, Elsevire North Holland Inc., New York.
- Wilson, R., (1994), *Nine Steps From Ignition to Extinguishment*, Firepro Institute Ltd. Putney, Vermont.
- Winston, W.L., (1994), *Operations Research: Applications and Algorithms*, 3rd edition, Wadsworth Publishing Company, Belmont, CA.

Xu, H., (1994), "Optimal Policies for Stochastic and Dynamic Vehicle Routing Problems", Massachusetts Institute of Technology, MIT_THESSES//1994-54, Department of Civil and Engineering.

INTERNET KAYNAKLARI

- [1] 66.102.9.104/search?q=cache:AqPUpl-40jDj:
www.infoday.org/business/operman/tech01.pdf
- [2] 210.90.46.47/html_service/eng/seoul%20town%20meeting/sess3_kh_1.htm
- [3] business.kent.edu/courses/fall02/m&is/34060/Chapter-16-Waiting%20Line.ppt
- [4] fef.sdu.edu.tr/~osuat/
- [5] mss03.isunet.edu/tp2/literature.htm
- [6] unstats.un.org/unsd/default.htm
- [7] gps.faa.gov/Library
- [8] gpsinformation.net/ot-20.htm
- [9] www21.blinkster.com/adil1985/
- [10] www.calisma.gov.tr/istatistik/cgm/yillik_asgari_ucret.htm
- [11] www.census.gov/population/socdemo/journey
- [12] www.csc.ncsu.edu/faculty/perros/simulation.pdf
- [13] www.fiat.com.tr/modeller/albea7.htm
- [14] www.gurbuz.com.tr
- [15] www.garmin.com/aboutGPS/
- [16] www.gsmworld.com
- [17] www.ibb.gov.tr/istanbultr/380/38002/2001/ulasim/
- [18] www.ibb.gov.tr/kentharitasi/kent_haritasi.cfm
- [19] www.ikinokta.com
- [20] www.isbank.com.tr
- [21] www.itl.nist.gov/div898/handbook/eda/section3/eda352.htm
- [22] www.kgm.gov.tr
- [23] www.law.nyu.edu/journals/envtllaw/issues/vol6/1/6nyuelj93.html
- [24] www.mapinfo.com
- [25] www.mcombs.utexas.edu/~andersone/courses/ops_core_course/q_note.pdf

- [26] www.mercedes.com.tr/Internet/TR/Route/
- [27] www.miltimap.com
- [28] www.njcmr.org/mpids/deep-eng/IE/Simulation/Chapter1%20-%20Final.htm
- [29] www.pirelli.com.tr/tr_TR/tyres/car_suv/prz/images/oto.pdf
- [30] www.renault.com.tr/2004/fiyat.php?model=clio_symbol_2
- [31] www.retroarchive.org/cpm/cdrom/ZSYS/ZNODE-12/A/CCPM92.NOV
- [32] www.rsgis.com/cbs1.htm
- [33] www.sigortam.net
- [34] www.shell.com/home/tr-tr/html/iwgen/app_profile/pompa.html
- [35] www.sonofon.dk/english/key_figures/market_shares/index.shtml
- [36] www.stmobiledata.com/more_on_mobitex_hm
- [37] www.teknobil.com.tr
- [38] www.tofas.com.tr/arama/arama.php?model_kod=MA&aksgrp_kod=KA
- [39] www.trimble.com/gps/
- [40] www.turkcell.com.tr/index/0,1027,683,00.html
- [41] www.turkcell.com.tr/kurumsal_hizmetler/tarifeler/filocellucet.html

EKLER

- Ek 1 İstanbul Kenti İçin Ortama Hız Ölçümü
- Ek 2 SAT Yaklaşımına Göre Simülasyon Programı Kodları
- Ek 3 NN ve SQM Yaklaşımlarına Göre Simülasyon Programı Kodları
- Ek 4 El ile Simülasyonu Yapılan Programın Bilgisayar Çıktısı
- Ek 5 El ile Simülasyon Sonuçları
- Ek 6 Hafif Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 7 Hafif Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 8 Hafif Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 9 Az Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 10 Az Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 11 Az Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 12 Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 13 Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 14 Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 15 Çok Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 16 Çok Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 17 Çok Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)
- Ek 18 Yoğun Trafik Altında SAT Politikası Doğrultusunda 3 Araçlık Filo İçin Koşturulan Simülasyonun Raporu
- Ek 19 Yoğun Trafik Altında a SAT Politikası Doğrultusunda 4 Araçlık Filo İçin Koşturulan Simülasyonun Raporu
- Ek 20 Yoğun Trafik Altında SAT Politikası Doğrultusunda 5 Araçlık Filo İçin Koşturulan Simülasyonun Raporu
- Ek 21 Yoğun Trafik Altında SAT Politikası Doğrultusunda 6 Araçlık Filo İçin Koşturulan Simülasyonun Raporu
- Ek 22 Yoğun Trafik Altında SAT Politikası Doğrultusunda 7 Araçlık Filo İçin Koşturulan Simülasyonun Raporu
- Ek 23 Yoğun Trafik Altında SAT Politikası Doğrultusunda 8 Araçlık Filo İçin Koşturulan Simülasyonun Raporu

Ek 1 İstanbul Kenti İçin Ortama Hız Ölçümü

Güneşli Memur / ÖNGÖRÜLEN ROTA		KM	YOLCULUK ZAMANI (SAAT)	ORTALAMA HIZ (km/saat)
1	Şenesenevler-Tüccarbaşı-Merdivenköy-S.Cedid-Atatürk cd.-İnönü Cd.-2.Köp.-	155	3,5	44,29
2	Fikirtepe - Güneşli	125	3	41,67
3	Pendik-Cevizli-Maltepe-Bostancı Köp.-K.Yatağı-Bayar Cd.-2.Köp.-E6-Güneşli	160	3,5	45,71
4	Namazgah-Belediye-Nato Yolu-İlke 2 Sitesi-Rasathane-Küçüksu-Kavacık-Güneşli	145	3	48,33
5	Ataşehir-Kayışdağı-Kayışdağı-Ataşehir-İçerenköy-Güneşli	160	3,5	45,71
6	Bağdat Cd.-Güneşli	130	3	43,33
7	Bağdat cd.-Selami Çeşme-K.Toprak-Z.Kamil-A.Zade-1.Köp.-Güneşli	120	3	40,00
8	Acıbadem - Güneşli	120	3	40,00
9	Acıbadem-Koşu Yolu-Bağlarbaşı-Altunizade-Beylerbeyi-1.Köp.-Güneşli	120	3	40,00
10	Beşiktaş-Güneşli	75	1,45	51,72
11	Beşiktaş-Balmumcu-Esentepe-M.Köy-Şişli-O.Meydanı-Haliç Köp Güneşli	100	2	50,00
12	M.Köy Taksim	100	2	50,00
13	M.Köy-Şişli-Taksim-Aksaray-Şehremini-Topkapı-Vatan Cd.-Güneşli	100	3	33,33
14	Halkalı Bahçeşehir	100	3	33,33
15	Bahçeşehir-Halkalı-Güneşli	80	2,5	32,00
16	Avcılar-Beylikdüzü	40	1,45	27,59
17	B.Düzü-İhlas Marmara Evler-Avcılar-Güneşli	40	1,45	27,59
18	Bakırköy	138	3	46,00
19	Ataköy-Bakırköy-Yenimah-İncirli-B.evler-Soğanlı-Şirinevler-Güneşli	138	3	46,00
20	Tarabya (İstinye) - Güneşli	90	2	45,00
21	Tarabya-İstinye-Hisarüstü- Uçaksavar- Akmerkez-4.Levent-Güneşli	90	2	45,00
22	Bostancı (tek çıkış)	115	3	38,33
23	Güneşli-Aksaray-Taksim-Bostancı-	115	3	38,33
24	Seyrantepe-Güneşli	110	3	36,67
25	S.Tepe-Levent-G.O.Paşa-Güneşli	110	3	36,67
26	Beylerbeyi - Bağlarbaşı	110	3	36,67

MAÇKA Memur / ÖNGÖRÜLEN ROTA		KM	YOLCULUK ZAMANI (SAAT)	ORTALAMA HIZ (km/saat)
1	Çamlıca - Maçka	83	1,10	75,45
2	Çamlıca-Libadiye- Acıbadem- Koşuyolu-Maçka	98	1,30	75,38
3	Çengelköy - Üsküdar	65	1,20	54,17
4	Ata 2- Nato Yolu- Sahil Yolu- Üsküdar- Altunizade	80	1,45	55,17
5	İçerenköy- Maçka	100	3,5	28,57
6	İçerenköy-Ataşehir-Atakent-K.Yatağı-Libadiye-Maçka	72	1,45	49,66
7	Bağdat Cd. - Maçka(tek çıkış)	70	1,30	53,85
8	A.Zade-Acıbadem-Fenerbahçe Stadı-Bostancı- Maçka	70	1,30	53,85
9	Pendik-Kartal-Drags-Maltepe-K.Yalı-Erenköy-Suadiye-Sahil Yolu-Bağdat Cd.-K.Yolu-Maçka	60	1	60,00
10	Maltepe - Maçka	50	1,50	33,33
11	Maltepe-Kazasker-M.Caddesi-Ş.Evler-Adatepe-Bostancı-K.Yalı-Maçka	70	1	70,00
12	Göztepe-Maçka	110	2	55,00
13	Göztepe-Ziverbey-Hasanpaşa-Maçka	42	1,20	35,00
14	Akatlar - Nispetiye-Akmerkez-Korukent-Ortaköy-Bebek-H.Üstü-Arnautköy-Maçka	70	1,20	58,33
15	Sultançiftliği - Maçka	120	3,5	34,29
16	G.O.Paşa-Vatan cad. -Taksim-Maçka	60	1,15	52,17
17	Bahçelievler - Maçka	132	2	66,00
18	B.Evler-İncirli-Ş.Evler-Maçka	78	1	78,00
19	Avcılar-Sefaköy-Maçka	78	1	78,00
20	K.Çekmece-Sefaköy-Şenlikköy-Yeşilköy-Bakırköy-Ataköy-Maçka	78	1	78,00
21	Zeytinburnu - Maçka	78	1	78,00
22	Z.Burnu - K.M.Paşa-Aksaray- Taksim- Maçka	78	1	78,00
23	Oyak Sitesi -Maçka	78	1	78,00
24	Maslak-4.Levent-Levent-Göztepe-M.Köy-Fulya-Maçka	78	1	78,00
25	Mimaroba-B.Çekmece-Avcılar-Bahçeşehir-Başak Şehir-Halkalı-Sefaköy-Maçka	78	1	78,00
26	Moda - Maçka	78	1	78,00
27	Moda-Maçka	78	1	78,00
28	Alemdağ-Ataşehir-Maçka	78	1	78,00
29	Alemdağ-Çekmeköy- Yenişehir- Ataşehir-Maçka	78	1	78,00
30	İstinye-Maçka	78	1	78,00
31	Maslak-İstinye- Tarabya - Bebek- Ortaköy- Maçka	78	1	78,00
Toplam Rota Uzunluğu (gidiş / geliş)		3203	70,75	45,27

Ek 2 SAT Yaklaşımına Göre Simülasyon Programı Kodları

```
//sat.cpp dosyasi baslar.
/*SAT Atama Politikasi Icin Simulasyon Programi
Istanbul 2004
*/
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>

#define MAX_OLAY 15000 //Programin bilgisayar belleğinde tutacağı maksimum olay sayısı
#define MAX_ARAC 100 //Programin bilgisayar belleğinde tutacağı maksimum arac sayısı

#define IDLE 0 //Araclarin durumlari
#define YOLDA 1
#define BUSY 2

#define ATANMAMIS 0 //Olaylarin durumlari
#define ATANMIS 1
#define BITMIS 2

int EN; //Servis alaninin eni
int BOY; //Servis alaninin boyu

int LAMBDA; //Poisson degiskeni, bir saatte olusmasi beklenen olay sayısı.
int TOPLAM_SURE; //Simulasyon suresi
int ARAC_SAYISI; //Toplam arac-tamirci sayısı
int HIZ; //Her bir aracın akis hizi
int IVME; //Her bir aracın hızlanma ve yavaşlama ivmesi

int YER_MEAN_X; //Olayın olustugu noktanın apsisinin ortalamasi
int YER_STANDARDDEV_X; //Olayın olustugu noktanın apsisinin standart sapmasi

int YER_MEAN_Y; //Olayın olustugu noktanın ordinatinin ortalamasi
int YER_STANDARDDEV_Y; //Olayın olustugu noktanın ordinatinin standart sapmasi

int CAPACITY; //Servis aracının merkeze ugramadan cevap verebilecegi talep sayısı

int SERVIS_MEAN; //Servis zamanlarının ortalamasi
int SERVIS_STANDARDDEV; //Servis zamanlarının standart sapmasi

int rapor_araligi; //Tek bir simulasyonun herbir altdizilerinin uzunlugu

int olay_sayisi; //Simulasyon kosumundaki olay sayısı
double table[401];
int zaman;
int toplam_bekleme; //Simulasyon kosumundaki toplam bekleme zamanı
int toplam_kilometre; //Simulasyon kosumundaki kat edilen toplam mesafe
FILE *output, *raporlar; //output ve raporlar isimli txt dosyalari acilir.
```

```

struct olay                                //olay sinifi
{
// Koordinatlar
    int x,y;
// Servis suresi
    int sure;
// 0'sa atanmamis 1'se atanmis 2 ise tamamlanmis
    int state;

    int olusturuldugu_zaman;

}olaylar[MAX_OLAY];

struct arac                                //arac sinifi
{
// Aracin koordinatlari belirtilir.
    int x,y;

// Durumu belirtir.
    int state;

// Istatistikler
    int toplam_yolda;
    int toplam_busy;
    int toplam_idle;

    int son_olay;

// Durumun bitmesine ne kadar sure var
    int kalan_yolda[MAX_OLAY];
    int kalan_busy[MAX_OLAY];
    int toplam_olay;

    int atandigi_olay[MAX_OLAY];
    int olay_bas; //olay dizisindeki ilk olay
    int olay_son; //olay dizisindeki son olay
    int bos_kalma_zamani;    //aracin bos kalacagi zaman

    int kilometre;

}araclar[MAX_ARAC];

void istatistikler();//Istatistikler fonksiyonu

void onislemeler()
//Simulasyonun calismaya baslamasindan onceki gerekli onislemeler yapilir.
{

    FILE *f=fopen("parametre.txt","r");
//parametre.txt dosyasi acilir ve ilgili parametreler okutulur.

```

```

fscanf(f, "%d",&EN);
fscanf(f, "%d",&BOY);
fscanf(f, "%d",&LAMBDA);
fscanf(f, "%d",&TOPLAM_SURE);
fscanf(f, "%d",&ARAC_SAYISI);
fscanf(f, "%d",&HIZ);
fscanf(f, "%d",&IVME);
fscanf(f, "%d",&YER_MEAN_X);
fscanf(f, "%d",&YER_STANDARDDEV_X);
fscanf(f, "%d",&YER_MEAN_Y);
fscanf(f, "%d",&YER_STANDARDDEV_Y);
fscanf(f, "%d",&SERVIS_MEAN);
fscanf(f, "%d",&SERVIS_STANDARDDEV);
fscanf(f, "%d",&CAPACITY);
fscanf(f, "%d",&rapor_araligi);
fclose(f);
//parametre.txt dosyasi kapatilir.

for(int i=0;i<ARAC_SAYISI;i++)
{
    //Tum araclari sirketin merkezinde konumlandiriyoruz.
    araclar[i].x=EN/2;
    araclar[i].y=BOY/2;

    //Araclarin hepsini IDLE durumuna getiriyoruz
    araclar[i].state=IDLE;
}

srand((unsigned)time(NULL)); //rastgele sayilarimizi olusturmamizi sagliyoruz.

// Normal dagilim tablosunu daha once olusturdugumuz normal.dat dosyasindan okutuyoruz,
f=fopen("normal.dat","r");
for(i=0;i<=400;i++)
    fscanf(f, "%lf",table+i);
fclose(f);
output=fopen("output.txt","w");
raporlar=fopen("raporlar.txt","w");
}

void arac_ata(int olay) //atanmamis olaylara arac atanir.
{
    int min=3000000,atanan,minsure,minyol;
    //300000000 cok buyuk bir sayi olarak secildi.
    for(int i=0;i<ARAC_SAYISI;i++)
    {
        if(araclar[i].bos_kalma_zamani<zaman)
            //arac daha once bosta kalmis ise simulasyon zamanini kullanilir.
            araclar[i].bos_kalma_zamani=zaman;

        int uzaklik=abs(araclar[i].x-olaylar[olay].x)+abs(araclar[i].y-olaylar[olay].y);
    }
}

```

```

//araclarin olay yerine uzakligini tespit edilir.
    int sure;
    //araclarin olay yerine ulasma zamani tespit edilir.
    if(uzaklik<(((double)HIZ)*HIZ/IVME)

    sure=araclar[i].bos_kalma_zamani+2*sqrt(((double)uzaklik)/IVME)*60+1;
    else

    sure=araclar[i].bos_kalma_zamani+(((double)uzaklik)/HIZ+(((double)HIZ)/IVME)*60
+1;

    if(sure<min || (sure==min && rand()%2==0))
    //olay yerine en kısa zamanda ulasabilecek arac, olaya atanir ve kat edeceği
mesafe tespit edilir..
    {
        atanan=i;
        minyol=uzaklik;
        min=sure; //olay yerine en kısa zamanda ulasabilecek aracın süresi
        minsure=min-araclar[i].bos_kalma_zamani;
    }
}

if(min!=3000000)
{
    olaylar[olay].state=ATANMIS; //olayın durumunu ATANMAMIS'tan ATANMIS'a getir.
    araclar[atanan].toplam_olay++; //toplam olay sayısını bir arttır.
    araclar[atanan].son_olay++; //son_olay sayısını bir arttır.
    if(araclar[atanan].state==IDLE)
        araclar[atanan].state=YOLDA; //aracın durumunu IDLE'dan YOLDA'ya getir.
    araclar[atanan].x=olaylar[olay].x; //atanan aracın yeni apsisini atandığı olayın apsisine
gotur.
    araclar[atanan].y=olaylar[olay].y; //atanan aracın yeni ordinatını atandığı olayın ordinatına
gotur.
    araclar[atanan].atandigi_olay[araclar[atanan].olay_son]=olay; //aracın atandığı olayı
olay kil.
    araclar[atanan].bos_kalma_zamani=min+olaylar[olay].sure; //aracın bos kalma
zamanını tespit et,
    araclar[atanan].kalan_busy[araclar[atanan].olay_son]=olaylar[olay].sure; //aracın
kalan mesgul zamanını olayın servis süresine eşitle.
    araclar[atanan].kalan_yolda[araclar[atanan].olay_son]=minsure;
    araclar[atanan].kilometre+=minyol; //atanan aracın kat ettiği mesafeye, atanan aracın olay
yerinden uzaklığını ekle.
    toplam_kilometre+=minyol; //toplam kat edilem mesafeye, atanan aracın olay yerinden
uzaklığını ekle.

    araclar[atanan].olay_son++; //aracın olay_son'unu bir arttır.

    fprintf(output, "Zaman=%d iken %d nolu arac %d,%d,%d olayına atandı. Ulasma
zamani:%d\n", zaman, atanan+1, olaylar[olay].x, olaylar[olay].y, olaylar[olay].sure, min);
}

```

```

}

void olay_ekle(int x,int y,int sure)//simulasyon dongusune olay ekleme fonksiyonu.
{
// Normal dagilimdan olusabilecek sinir disi degerleri sinirlara indiriyoruz.
if(x<0)
    x=0;
if(x>=EN)
    x=EN-1;

if(y<0)
    y=0;
if(x>=BOY)
    x=BOY-1;

if(sure<1)    //surenin negatif cikmasini engelliyoruz.
    sure=1;

olaylar[olay_sayisi].x=x,    //olayin olustugu noktanin apsisi
olaylar[olay_sayisi].y=y,    //olayin olustugu noktanin ordinati
olaylar[olay_sayisi].sure=sure,    //olayin gereksinim duyacagi servis zamani
olaylar[olay_sayisi].state=ATANMAMIS, //olayin durumunu ATANMAMIS'a getiriyoruz.
olaylar[olay_sayisi].olusturuldugu_zaman=zaman,    //olayin olusturuldugu zamani
belirliyoruz.
olay_sayisi++;    //olay sayisini bir arttir.

fprintf(output,"Zaman=%d iken yeni olay:%d,%d,%d\n",zaman,x,y,sure);
}

int normal(int mean,int variance)//Normal dagilima gore rastgele sayi olusturulmasi
{
double f=((double)(rand()%1001))/1000;

int i;
for(i=0;i<400&&table[i]<f;i++);

double fl=((double)(i-200))/100;

return fl*variance+mean;
}

int faktoriyel(int x)//Poisson dagilimda kullanacagimiz faktoriyel fonksiyonu
{
if(x<2)
    return 1;
return x*faktoriyel(x-1);
}

int poisson(double lambda)//Poisson dagilima gore rastgele sayi olusturulmasi

```

```

{
    double f,toplam=0;
    f=((double)(rand()%1000))/1000;

    int i;

    for(i=0;toplam<f;i++)
        toplam+=exp(- lambda)*pow(lambda,i)/faktoriyel(i);

    return i-1;
}

int olay_var()//Mesai bittiđi halde sisteme girilmis yanıt beklemekte olan olayların olduđunu
gosterir.
{
    for(int i=0;i<olay_sayisi;i++)
        if(olaylar[i].state!=BITMIS) //sistemde durumu BITMIS olmayan olay vardır.
            return 1;

    for(i=0;i<ARAC_SAYISI;i++)
        if(araclar[i].state!=IDLE) //sistemde durumu IDLE olmayan olay vardır.
            return 1;

    return 0;
}

void simulasyon() //Simulasyon fonksiyonu
{
    //Kac tane yeni olay olduđunu tutar
    int yeni_olay;

    //Yeni olayın koordinatları ve servis süresi
    int x,y,sure;

    // Simulasyon dongusu. Buraya her dakika girer.
    for(zaman=0;zaman<TOPLAM_SURE || olay_var();zaman++)
    {
        yeni_olay=0;

        if(zaman%rapor_araligi==0 && zaman!= 0)
            istatistikler();

        // Poissonla yeni olay sayisini belirtilir.
        if(zaman<TOPLAM_SURE)
            yeni_olay=poisson(((double)LAMBDA)/60);

        // Yeni olayların koordinatlarını ve sürelerini normal distributionla belirliyor.
        for(int j=0;j<yeni_olay;j++)
        {
            x=normal(YER_MEAN_X,YER_STANDARDDEV_X);

```

```

y=normal(YER_MEAN_Y,YER_STANDARDDEV_Y);
sure=normal(SERVIS_MEAN,SERVIS_STANDARDDEV);
olay_ekle(x,y,sure);
}

// Su ana kadar arac atanmamis olan olaylara ilgili aracların atanması gerçekleştirilir.
for(j=0;j<olay_sayisi;j++)
    if(olaylar[j].state==ATANMAMIS)
        arac_ata(j);

// Araclarin durumlarını değerlendiriyor. Durumu bitmiş araçların bir sonraki duruma
gecmelerini sağlar.
for(j=0;j<ARAC_SAYISI;j++)
    if(araclar[j].state==YOLDA)
    {
        araclar[j].kalan_yolda[araclar[j].olay_bas]--; //Yolda olan aracın
yolda geçireceği zamanı bir azalt.
        araclar[j].toplam_yolda++; //Yolda olan araçların yolda geçirdikleri
toplam zamanı bir artırılır.
        if(araclar[j].kalan_yolda[araclar[j].olay_bas]==0) //Yolda olan arac
olay yerine ulaştığında,
        {
            if(araclar[j].atandigi_olay[araclar[j].olay_bas]!=-1) //Eğer arac
merkeze gönderilmemiş ise,
            {
                toplam_bekleme+=zaman-
olaylar[araclar[j].atandigi_olay[araclar[j].olay_bas]].olusturuldugu_zaman;
                //musterinin toplam bekleme zamanını tespit et.
                araclar[j].state=BUSY; //Aracın durumunu
YOLDA'dan BUSY'e getir.
                fprintf(output,"Zaman=%d iken %d nolu arac olay yerine
ulasti\n",zaman,j+1);
            }
            else //Eğer arac merkeze gönderilmiş ise,
            {
                araclar[j].olay_bas++; //olay_bas indeksini bir
arttır.
                if(araclar[j].olay_bas==araclar[j].olay_son) //olay_bas
indeksi olay_son indeksine eşit ise,
                    araclar[j].state=IDLE; //Aracın durumunu
IDLE yap.
                    else
                        araclar[j].state=YOLDA; //Aracın durumunu
YOLDA yap.
                fprintf(output,"Zaman=%d iken %d nolu arac merkeze
donda\n",zaman,j+1);
            }
        }
    }
}

```

```

else if(araclar[j].state==BUSY) //Aracin durumu BUSY ise,
{
    araclar[j].kalan_busy[araclar[j].olay_bas]--; //olay_bas indeksini bir
azaltilir.
    araclar[j].toplam_busy++; //Aracin mesgul oldugu toplam zaman bir
arttirilir.
    if(araclar[j].kalan_busy[araclar[j].olay_bas]==0) //Aracin kalan
mesgul zamani 0 ise,
    {
        olaylar[araclar[j].atandigi_olay[araclar[j].olay_bas]].state=BITMIS; //Aracin
atandigi olayin durumu BITMIS'e getirilir.
        fprintf(output, "Zaman=%d iken %d nolu arac gorevini
tamamladi\n", zaman, j+1);

        araclar[j].olay_bas++; //olay_bas indeksini bir arttir.
        if(araclar[j].olay_bas==araclar[j].olay_son) //olay_bas indeksi
olay_son indeksine esit ise,
            araclar[j].state=IDLE; //Aracin durumunu IDLE yap.
        else
            araclar[j].state=YOLDA; //Aracin durumunu YOLDA
yap.
    }
}

// SQM arac atama politikasinin isletilmesinde kullanilmistir.
// NN ve SAT politikasinda CAPACITY=1000 (buyuk bir sayi) olarak parametrelerde
kisminda tanimlanir.
/* Arac eger kapasitesini tamamlamis
veya gecmisse yeni bir ise de atanmamissa merkeze gonderilmesi saglanmistir.
Bunun yapmak icin araclara son_olay diye bir degisken eklenmistir.
son_olay en son merkeze gonderilisinden sonra kac kez olaya gonderildigini tutmaktadır.
*/
else
{
    if(araclar[j].son_olay >= CAPACITY)
    {
        fprintf(output, "Zaman=%d iken %d nolu arac merkeze
gonderildi\n", zaman, j+1);
        int min=abs(araclar[j].x-EN/2)+abs(araclar[j].y-BOY/2);
        araclar[j].state=YOLDA;
        araclar[j].x=EN/2;
        araclar[j].y=BOY/2;
        araclar[j].atandigi_olay[araclar[j].olay_bas]=-1;
        araclar[j].olay_son++;
        araclar[j].kilometre+=min;
        toplam_kilometre+=min;
        if(min<((double)HIZ)*HIZ/IVME)

        araclar[j].kalan_yolda[araclar[j].olay_bas]=2*sqrt(((double)min)/IVME)*60+1;
        else

```

```

        araclar[j].kalan_yolda[araclar[j].olay_bas]=(((double)min)/HIZ+((double)HIZ)/IVME)
        *60+1;

        araclar[j].bos_kalma_zamani= zaman+
araclar[j].kalan_yolda[araclar[j].olay_bas];
        araclar[j].son_olay=0;

    }

}

}

void istatistikler()    //simulasyon istatistikleri olusturulur.
{
    fprintf(raporlar,"Rapor(0-%d)\n",zaman);
    for(int i=0;i<ARAC_SAYISI;i++) //her aracın istatistigi tutulur.
    {
        fprintf(raporlar,"%d. arac istatistikleri\n",(i+1));
        fprintf(raporlar,"Yolda gecirdigi toplam sure: %d\n",araclar[i].toplam_yolda);
        fprintf(raporlar,"Calisarak gecirdigi toplam sure: %d\n",araclar[i].toplam_busy);
        fprintf(raporlar,"Atandigi toplam olay sayisi %d\n",araclar[i].toplam_olay);
        fprintf(raporlar,"Aldigi toplam yol: %d\n\n",araclar[i].kilometre);
    }
    fprintf(raporlar,"Olay sayisi=%d\n",olay_sayisi);
    fprintf(raporlar,"Ortalama bekleme zamanı=%lf\n",((double)toplam_bekleme)/olay_sayisi);
    fprintf(raporlar,"Arac basına alınan yol=%lf\n",((double)toplam_kilometre/ARAC_SAYISI);
}

int main() //Main Fonksiyonu
{
    onislemeler();
    simulasyon();
    istatistikler();
    fclose(output);
    return 0;
}

// sat.cpp dosyasi biter.

```

Ek 3 NN ve SQM Yaklaşımlarına Göre Simülasyon Programı Kodları

```
//sqm.cpp dosyasi baslar.
/*Calismamizda SAT Atama Politikasinin performansini kiyaslayacagimiz
   SQM ve NN Atama Politikaları Icin Simulasyon Programi
   Istanbul 2004
*/

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>

#define MAX_OLAY 15000 //Programin bilgisayar belleğinde tutacağı maksimum olay sayısı
#define MAX_ARAC 100 //Programin bilgisayar belleğinde tutacağı maksimum arac sayısı

#define IDLE 0 //Aracın durumları
#define YOLDA 1
#define BUSY 2

#define ATANMAMIS 0 //Olayların durumları
#define ATANMIS 1
#define BITMIS 2

int EN; //Servis alanının eni
int BOY; //Servis alanının boyu

int LAMBDA; //Poisson değışkeni, saatte karşılasılması
beklenen olay (musteri talebi) sayısı.
int TOPLAM_SURE; //Simulasyon süresi
int ARAC_SAYISI; //Toplam arac-tamirci sayısı
int HIZ; //Her bir aracın akis hızı
int IVME; //Her bir aracın hızlanma ve yavaşlama ivmesi

int YER_MEAN_X; //Olayın oluştuğu noktanın apsisinin ortalaması
int YER_STANDARDDEV_X; //Olayın oluştuğu noktanın apsisinin standart
sapması

int YER_MEAN_Y; //Olayın oluştuğu noktanın ordinatinin ortalaması
int YER_STANDARDDEV_Y; //Olayın oluştuğu noktanın ordinatinin standart
sapması

int CAPACITY; //Servis aracının merkeze uğramadan cevap verebileceği
talep sayısı

int SERVIS_MEAN; //Servis zamanlarının ortalaması
int SERVIS_STANDARDDEV; //Servis zamanlarının standart sapması

int rapor_araligi; //Tek bir simulasyonun Herbir altdizilerinin uzunluğu

int olay_sayisi; //Simulasyon kosumundaki olay sayısı
double table[401];
```

```

int zaman;
int toplam_bekleme;           //Simulasyon kosumundaki toplam bekleme zamani
int toplam_kilometre; //Simulasyon kosumundaki kat edilen toplam mesafe
FILE *output, *raporlar; //output ve raporlar isimli txt dosyalari acilir.

struct olay                    //olay sinifi
{
// Koordinatlar
    int x,y;
// Servis suresi
    int sure;
// 0'sa atanmamis 1'se atanmis 2 ise tamamlanmis
    int state;

    int olusturuldugu_zaman;

} olaylar[MAX_OLAY];

struct arac                    //arac sinifi
{
// Aracin koordinatlari belirtilir.
    int x,y;

// Aracin durumu belirtir
    int state;

// Istatistikler
    int toplam_yolda;
    int toplam_busy;
    int toplam_idle;

    int son_olay;

// Durumun bitmesine ne kadar sure var
    int kalan_yolda;
    int kalan_busy;
    int toplam_olay;
    int kilometre;

    int atandigi_olay;
} aracler[MAX_ARAC];

void istatistikler();//Istatistikler fonksiyonu

void onislemler()           //Simulasyonun calismaya baslarmasindan onceki gerekli onislemler
yapilir.
{

    FILE *f=fopen("parametre.txt","r");
    //parametre.txt dosyasi acilir ve ilgili parametreler okutulur.

```

```

fscanf(f, "%d", &EN);
fscanf(f, "%d", &BOY);
fscanf(f, "%d", &LAMBDA);
fscanf(f, "%d", &TOPLAM_SURE);
fscanf(f, "%d", &ARAC_SAYISI);
fscanf(f, "%d", &HIZ);
fscanf(f, "%d", &IVME);
fscanf(f, "%d", &YER_MEAN_X);
fscanf(f, "%d", &YER_STANDARDDEV_X);
fscanf(f, "%d", &YER_MEAN_Y);
fscanf(f, "%d", &YER_STANDARDDEV_Y);
fscanf(f, "%d", &SERVIS_MEAN);
fscanf(f, "%d", &SERVIS_STANDARDDEV);
fscanf(f, "%d", &CAPACITY);
fscanf(f, "%d", &rapor_araligi);

```

```

fclose(f); //parametre.txt dosyasi kapatilir.

```

```

for(int i=0; i<ARAC_SAYISI; i++)
{
    //Tum araclari sirketin merkezinde konumlandiriyoruz.
    araclar[i].x=EN/2;
    araclar[i].y=BOY/2;

    //Araclarin hepsini IDLE durumuna getiriyoruz
    araclar[i].state=IDLE;
}

```

```

srand((unsigned)time(NULL)); //rastgele sayilarimizi olusturmamizi sagliyoruz.

```

```

// Normal dagilim tablosunu daha once olusturdugumuz normal.dat dosyasindan okutuyoruz.
f=fopen("normal.dat", "r");
for(i=0; i<=400; i++)
    fscanf(f, "%lf", table+i);
fclose(f);
output=fopen("output.txt", "w");
raporlar=fopen("raporlar.txt", "w");
}

```

```

void arac_ata(int olay) //atanmamis olaylara arac atanir.
{
    int min=3000000, atanan, yol; //30000000 cok buyuk bir sayi olarak secildi.

    for(int i=0; i<ARAC_SAYISI; i++)
        if(araclar[i].state==IDLE) //durumu idle olan araclarin olay yerine uzakligini tespit et.
        {
            yol=abs(araclar[i].x-olaylar[olay].x)+abs(araclar[i].y-olaylar[olay].y);
            if(yol<min || (yol==min && rand()%2==0))
            {

```

```

        atanan=i;        //atanmasına karar verilen aracı tespit et.
        min=yol;        //atanmasına karar verilen aracın kat edeceği mesafeyi
tespit et.
    }
}

if(min!=3000000)
{
    olaylar[olay].state=ATANMIS; //olayın durumunu ATANMAMIS'tan ATANMIS'a getir.
    aralar[atanan].toplam_olay++; //toplam olay sayısını bir arttır.
    aralar[atanan].son_olay++; //son_olay sayısını bir arttır.
    aralar[atanan].state=YOLDA; //aracın durumunu IDLE'dan YOLDA'ya getir.
    aralar[atanan].x=olaylar[olay].x; //atanan aracın yeni apsisini atandığı olayın
apsisine gotur.
    aralar[atanan].y=olaylar[olay].y; //atanan aracın yeni ordinatini atandığı olayın
ordinatına gotur.
    aralar[atanan].kalan_busy=olaylar[olay].sure; //aracın kalan mesgul zamanını olayın
servis suresine esitle.
    aralar[atanan].atandigi_olay=olay; //aracın atandığı olayı olay kil.
    aralar[atanan].kilometre+=min; //atanan aracın kat ettiği mesafeye, atanan aracın
olay yerinden uzaklığını ekle.
    toplam_kilometre+=min; //toplam kat edilem mesafeye,
atanan aracın olay yerinden uzaklığını ekle.

    if(min<(((double)HIZ)*HIZ/IVME) //aracın olay yerinden uzaklığına göre olay yerine
ulasma zamanını tespit et.
        aralar[atanan].kalan_yolda=2*sqrt(((double)min)/IVME)*60+1;
    else
        aralar[atanan].kalan_yolda=(((double)min)/HIZ+((double)HIZ)/IVME)*60+1;

    fprintf(output,"Zaman=%d iken %d nolu arac %d,%d,%d olayına atandı. Ulasma
zamanı:%d\n",zaman,atanan+1,olaylar[olay].x,olaylar[olay].y,olaylar[olay].sure,zaman+aralar[atanan].kalan_yolda-1);
}
}

void olay_ekle(int x,int y,int sure) //simulasyon dongusune olay ekleme fonksiyonu.
{
// Normal dagilimdan olusabilecek sinir disi degerleri sinirlara indiriyoruz.
if(x<0)
    x=0;
if(x>=EN)
    x=EN-1;

if(y<0)
    y=0;
if(y>=BOY)
    y=BOY-1;

if(sure<1) //surenin negatif cikmasini engelliyoruz.

```

```

sure=1;

olaylar[olay_sayisi].x=x,    //olayin olustugu noktanin apsisi
olaylar[olay_sayisi].y=y,    //olayin olustugu noktanin ordinati
olaylar[olay_sayisi].sure=sure,    //olayin gereksinim duyacagi servis zamani
olaylar[olay_sayisi].state=ATANMAMIS, //olayin durumunu ATANMAMIS'a getiriyoruz.
olaylar[olay_sayisi].olusturuldugu_zaman=zaman,    //olayin olusturuldugu zamani
belirliyoruz.
olay_sayisi++;    //olay sayisini bir arttir.

fprintf(output, "Zaman=%d iken yeni olay:%d,%d,%d\n", zaman, x, y, sure);
}

int normal(int mean, int variance)    //Normal dagilima gore rastgele sayi olusturulmasi
{
    double f=((double)(rand()%1001))/1000;

    int i;
    for(i=0; i<400 && table[i]<f; i++);

    double fl=((double)(i-200))/100;

    return fl*variance+mean;
}

int faktoriyel(int x)    //Poisson dagilimda kullanacagimiz faktoriyel fonksiyonu
{
    if(x<2)
        return 1;
    return x*faktoriyel(x-1);
}

int poisson(double lambda)    //Poisson dagilima gore rastgele sayi olusturulmasi
{
    double f, toplam=0;
    f=((double)(rand()%1000))/1000;

    int i;

    for(i=0; toplam<f; i++)
        toplam+=exp(-lambda)*pow(lambda, i)/faktoriyel(i);

    return i-1;
}

int olay_var() //Mesai bittigi halde sisteme girilmis yanıt beklemekte olan olayların olduğunu
gosterir.
{
    for(int i=0; i<olay_sayisi; i++)

```

```

        if(olaylar[i].state!=BITMIS)           //sistemde durumu BITMIS olmayan olay vardır.
            return 1;

    for(i=0;i<ARAC_SAYISI;i++)
        if(araclar[i].state!=IDLE) //sistemde durumu IDLE olmayan olay vardır.
            return 1;

    return 0;
}

void simulasyon() //Simulasyon fonksiyonu
{
    //Kac tane yeni olay oldugunu tutar
    int yeni_olay;

    //Yeni olayin koordinatlari ve servis suresi
    int x,y,sure;

    // Simulasyon dongusu. Buraya her dakika girer.
    for(zaman=0;zaman<TOPLAM_SURE || olay_var();zaman++) //Toplam sure dolmamissa
        veya olay var ise simulasyon dongusu devam eder.
        {

            yeni_olay=0;

            if(zaman%rapor_araligi==0 && zaman!=0)
                istatistikler();

            // Poissonla yeni olay sayisini belirtilir.
            if(zaman<TOPLAM_SURE)
                yeni_olay=poisson(((double)LAMBDA)/60);
            // Yeni olaylarin koordinatlarini ve surelerini normal distributionla belirliyor.
            for(int j=0;j<yeni_olay;j++)
            {
                x=normal(YER_MEAN_X,YER_STANDARDDEV_X);
                y=normal(YER_MEAN_Y,YER_STANDARDDEV_Y);
                sure=normal(SERVIS_MEAN,SERVIS_STANDARDDEV);
                olay_ekle(x,y,sure);
            }
            // Su ana kadar arac atanmamis olan olaylara ilgili aracarin atanmasi gerceklestirilir.
            for(j=0;j<olay_sayisi;j++)
                if(olaylar[j].state==ATANMAMIS)
                    arac_ata(j);

            // Aracarin durumlarini degerlendiriyor. Durumu bitmis aracarin bir sonraki duruma
            gecmelerini saglanir.
            for(j=0;j<ARAC_SAYISI;j++)
                if(araclar[j].state==YOLDA)
                {

```

```

        araclar[j].kalan_yolda--; //Yolda olan aracların yolda geçirecekleri
zaman bir azaltılır.
        araclar[j].toplam_yolda++; //Yolda olan aracların yolda geçirdikleri
toplam zaman bir arttırılır.
        if(araclar[j].kalan_yolda==0)//Aracın yolda geçireceği zaman yoksa
olay yerine ulaşmıştır.
        {
            if(araclar[j].atandigi_olay!=-1)
            {
                toplam_bekleme+=zaman-
olaylar[araclar[j].atandigi_olay].olusturuldugu_zaman; //Musterinin bekleme zamanı
hesaplanır.
                araclar[j].state=BUSY; //Aracın durumu BUSY'ye
getirilir.
                fprintf(output,"Zaman=%d iken %d nolu arac olay yerine
ulaştı\n",zaman,j+1);
            }
            else
            {
                araclar[j].state=IDLE; //Arac merkeze gönderilmiş ise
durumu IDLE'a getirilir.
                fprintf(output,"Zaman=%d iken %d nolu arac merkeze
döndü\n",zaman,j+1);
            }
        }
    }
    else if(araclar[j].state==BUSY) //Durumu BUSY olan araçlar için,
    {
        araclar[j].kalan_busy--; //Kalan mesgul oldukları zaman bir
azaltılır.
        araclar[j].toplam_busy++; //Toplam mesgul oldukları zaman bir
arttırılır.
        if(araclar[j].kalan_busy==0) //Kalan mesgul oldukları zaman 0 ise,
        {
            araclar[j].state=IDLE; //Aracın durumu IDLE'a getirilir.
            olaylar[araclar[j].atandigi_olay].state=BITMIS; //Aracın
atanmış olduğu olayın durumu BITMIS yapılır.
            fprintf(output,"Zaman=%d iken %d nolu arac görevini
tamamladı\n",zaman,j+1);
        }
    }
}

// SQM arac atama politikasının işletilmesinde kullanılmıştır.
//SQM politikasında CAPACITY=1 olarak parametrelerde kısmında tanımlanır.
/*Arac eğer kapasitesini tamamlamış
veya geçmişse yeni bir işe de atanmamışsa merkeze gönderilmesi sağlanmıştır.
Bunun yapmak için araçlara son_olay diye bir değişken eklenmiştir.
son_olay en son merkeze gönderilisinden sonra kaç kez olaya gönderildiğini tutmaktadır.
*/
    else
    {

```

```

        if(araclar[j].son_olay >= CAPACITY)
        {
            fprintf(output,"Zaman=%d iken %d nolu arac merkeze
gonderildi\n",zaman,j+1);
            int min=abs(araclar[j].x-EN/2)+abs(araclar[j].y-BOY/2);
            araclar[j].state=YOLDA;
            araclar[j].x=EN/2;
            araclar[j].y=BOY/2;
            araclar[j].atandigi_olay=-1;
            araclar[j].kilometre+=min;
            toplam_kilometre+=min;
            if(min<((double)HIZ)*HIZ/IVME)

araclar[j].kalan_yolda=2*sqrt(((double)min)/IVME)*60+1;
            else

araclar[j].kalan_yolda=(((double)min)/HIZ+((double)HIZ)/IVME)*60+1;
            araclar[j].son_olay=0;
        }
    }
}

void istatistikler()    //simulasyon istatistikleri olusturulur.
{
    fprintf(raporlar,"Rapor(0-%d)\n",zaman);
    for(int i=0;i<ARAC_SAYISI;i++) //her aracın istatistigi tutulur.
    {
        fprintf(raporlar,"%d. arac istatistikleri\n",(i+1));
        fprintf(raporlar,"Yolda gecirdigi toplam sure: %d\n",araclar[i].toplam_yolda);
        fprintf(raporlar,"Calisarak gecirdigi toplam sure: %d\n",araclar[i].toplam_busy);
        fprintf(raporlar,"Atandigi toplam olay sayisi %d\n",araclar[i].toplam_olay);
        fprintf(raporlar,"Aldigi toplam yol:%d\n\n",araclar[i].kilometre);
    }
    fprintf(raporlar,"Olay sayisi=%d\n",olay_sayisi);
    fprintf(raporlar,"Ortalama bekleme zamanı=%lf\n",((double)toplam_bekleme)/olay_sayisi);
    fprintf(raporlar,"Arac basına alınan yol=%lf\n",(double)toplam_kilometre/ARAC_SAYISI);
}

int main()    //Main Fonksiyonu
{
    onislemeler();
    simulasyon();
    istatistikler();
    fclose(output);
    fclose(raporlar);
    return 0;
}
// sqm.cpp dosyası biter.

```

Ek 4 El ile Simülasyonu Yapılan Programın Bilgisayar Çıktısı

Zaman=22 iken yeni olay:16,37,27
 Zaman=22 iken 1 nolu arac 16,37,27 olayına atandı. Ulasma suresi:40
 Zaman=23 iken yeni olay:23,17,30
 Zaman=23 iken 2 nolu arac 23,17,30 olayına atandı. Ulasma suresi:17
 Zaman=28 iken yeni olay:20,33,27
 Zaman=28 iken 3 nolu arac 20,33,27 olayına atandı. Ulasma suresi:28
 Zaman=40 iken 2 nolu arac olay yerine ulasti
 Zaman=55 iken yeni olay:25,31,54
 Zaman=55 iken 4 nolu arac 25,31,54 olayına atandı. Ulasma suresi:33
 Zaman=56 iken 3 nolu arac olay yerine ulasti
 Zaman=62 iken 1 nolu arac olay yerine ulasti
 Zaman=70 iken 2 nolu arac gorevini tamamladi
 Zaman=83 iken 3 nolu arac gorevini tamamladi
 Zaman=88 iken 4 nolu arac olay yerine ulasti
 Zaman=89 iken 1 nolu arac gorevini tamamladi
 Zaman=90 iken yeni olay:33,6,63
 Zaman=90 iken 2 nolu arac 33,6,63 olayına atandı. Ulasma suresi:40
 Zaman=96 iken yeni olay:14,18,1
 Zaman=96 iken 5 nolu arac 14,18,1 olayına atandı. Ulasma suresi:21
 Zaman=117 iken 5 nolu arac olay yerine ulasti
 Zaman=118 iken 5 nolu arac gorevini tamamladi
 Zaman=120 iken yeni olay:10,8,1
 Zaman=120 iken 5 nolu arac 10,8,1 olayına atandı. Ulasma suresi:29
 Zaman=127 iken yeni olay:19,26,30
 Zaman=127 iken 3 nolu arac 19,26,30 olayına atandı. Ulasma suresi:21
 Zaman=130 iken 2 nolu arac olay yerine ulasti
 Zaman=142 iken 4 nolu arac gorevini tamamladi
 Zaman=148 iken 3 nolu arac olay yerine ulasti
 Zaman=149 iken 5 nolu arac olay yerine ulasti
 Zaman=150 iken 5 nolu arac gorevini tamamladi
 Zaman=169 iken yeni olay:2,10,9
 Zaman=169 iken 5 nolu arac 2,10,9 olayına atandı. Ulasma suresi:23
 Zaman=178 iken 3 nolu arac gorevini tamamladi
 Zaman=192 iken 5 nolu arac olay yerine ulasti
 Zaman=193 iken 2 nolu arac gorevini tamamladi
 Zaman=197 iken yeni olay:30,29,27
 Zaman=197 iken 4 nolu arac 30,29,27 olayına atandı. Ulasma suresi:19
 Zaman=201 iken 5 nolu arac gorevini tamamladi
 Zaman=205 iken yeni olay:37,14,66
 Zaman=205 iken 2 nolu arac 37,14,66 olayına atandı. Ulasma suresi:27
 Zaman=216 iken 4 nolu arac olay yerine ulasti
 Zaman=232 iken 2 nolu arac olay yerine ulasti
 Zaman=243 iken 4 nolu arac gorevini tamamladi
 Zaman=261 iken yeni olay:23,13,39
 Zaman=261 iken 3 nolu arac 23,13,39 olayına atandı. Ulasma suresi:34
 Zaman=274 iken yeni olay:27,10,9
 Zaman=274 iken 4 nolu arac 27,10,9 olayına atandı. Ulasma suresi:42
 Zaman=278 iken yeni olay:21,19,45
 Zaman=278 iken 1 nolu arac 21,19,45 olayına atandı. Ulasma suresi:43
 Zaman=295 iken 3 nolu arac olay yerine ulasti
 Zaman=298 iken 2 nolu arac gorevini tamamladi
 Zaman=310 iken yeni olay:39,24,1
 Zaman=310 iken 2 nolu arac 39,24,1 olayına atandı. Ulasma suresi:27
 Zaman=316 iken 4 nolu arac olay yerine ulasti
 Zaman=317 iken yeni olay:25,32,70
 Zaman=317 iken 5 nolu arac 25,32,70 olayına atandı. Ulasma suresi:76
 Zaman=320 iken yeni olay:13,25,87
 Zaman=321 iken 1 nolu arac olay yerine ulasti
 Zaman=325 iken 4 nolu arac gorevini tamamladi
 Zaman=326 iken 4 nolu arac 13,25,87 olayına atandı. Ulasma suresi:52
 Zaman=329 iken yeni olay:15,8,27
 Zaman=330 iken yeni olay:14,9,37
 Zaman=334 iken 3 nolu arac gorevini tamamladi
 Zaman=335 iken 3 nolu arac 15,8,27 olayına atandı. Ulasma suresi:28
 Zaman=336 iken yeni olay:24,0,50
 Zaman=337 iken 2 nolu arac olay yerine ulasti
 Zaman=338 iken 2 nolu arac gorevini tamamladi
 Zaman=339 iken 2 nolu arac 14,9,37 olayına atandı. Ulasma suresi:69
 Zaman=349 iken yeni olay:37,25,37
 Zaman=357 iken yeni olay:33,29,21
 Zaman=359 iken yeni olay:39,24,55

Zaman=363 iken 3 nolu arac olay yerine ulasti
 Zaman=366 iken 1 nolu arac gorevini tamamladi
 Zaman=367 iken 1 nolu arac 24,0,50 olayina atandi. Ulasma suresi:42
 Zaman=368 iken yeni olay:19,29,1
 Zaman=378 iken 4 nolu arac olay yerine ulasti
 Zaman=390 iken 3 nolu arac gorevini tamamladi
 Zaman=391 iken 3 nolu arac 37,25,37 olayina atandi. Ulasma suresi:67
 Zaman=392 iken yeni olay:31,28,66
 Zaman=393 iken 5 nolu arac olay yerine ulasti
 Zaman=408 iken 2 nolu arac olay yerine ulasti
 Zaman=409 iken 1 nolu arac olay yerine ulasti
 Zaman=445 iken 2 nolu arac gorevini tamamladi
 Zaman=446 iken 2 nolu arac 33,29,21 olayina atandi. Ulasma suresi:67
 Zaman=451 iken yeni olay:11,26,15
 Zaman=458 iken 3 nolu arac olay yerine ulasti
 Zaman=459 iken 1 nolu arac gorevini tamamladi
 Zaman=460 iken 1 nolu arac 39,24,55 olayina atandi. Ulasma suresi:67
 Zaman=463 iken 5 nolu arac gorevini tamamladi
 Zaman=464 iken 5 nolu arac 19,29,1 olayina atandi. Ulasma suresi:22
 Zaman=465 iken 4 nolu arac gorevini tamamladi
 Zaman=466 iken 4 nolu arac 31,28,66 olayina atandi. Ulasma suresi:40
 Zaman=473 iken yeni olay:37,17,78
 Zaman=486 iken 5 nolu arac olay yerine ulasti
 Zaman=487 iken 5 nolu arac gorevini tamamladi
 Zaman=488 iken 5 nolu arac 11,26,15 olayina atandi. Ulasma suresi:25
 Zaman=495 iken 3 nolu arac gorevini tamamladi
 Zaman=496 iken 3 nolu arac 37,17,78 olayina atandi. Ulasma suresi:21
 Zaman=506 iken 4 nolu arac olay yerine ulasti
 Zaman=513 iken 2 nolu arac olay yerine ulasti
 Zaman=513 iken 5 nolu arac olay yerine ulasti
 Zaman=517 iken 3 nolu arac olay yerine ulasti
 Zaman=527 iken 1 nolu arac olay yerine ulasti
 Zaman=528 iken 5 nolu arac gorevini tamamladi
 Zaman=534 iken 2 nolu arac gorevini tamamladi
 Zaman=572 iken 4 nolu arac gorevini tamamladi
 Zaman=582 iken 1 nolu arac gorevini tamamladi
 Zaman=595 iken 3 nolu arac gorevini tamamladi

1. arac istatistikleri

Yolda gecirdigi toplam sure: 196
 Calisarak gecirdigi toplam sure: 177
 Atandigi toplam olay sayisi 4

2. arac istatistikleri

Yolda gecirdigi toplam sure: 253
 Calisarak gecirdigi toplam sure: 218
 Atandigi toplam olay sayisi 6

3. arac istatistikleri

Yolda gecirdigi toplam sure: 205
 Calisarak gecirdigi toplam sure: 238
 Atandigi toplam olay sayisi 6

4. arac istatistikleri

Yolda gecirdigi toplam sure: 191
 Calisarak gecirdigi toplam sure: 243
 Atandigi toplam olay sayisi 5

5. arac istatistikleri

Yolda gecirdigi toplam sure: 202
 Calisarak gecirdigi toplam sure: 97
 Atandigi toplam olay sayisi 6

Olay sayisi=27

Ortalama bekleme zamani=56.814815

Ek 5 El ile Simülasyonu Yapılan Programın Bilgisayar Çıktısı

örün x	37	hız	40 km/saat	yolculuk mesafesi	8 km
örün y	25	vmg.	240 km/saat2	yolculuksuresi	22.00 dakika
hedef x	37	if d<5.66	21.91 dakika		
hedef y	17	if d>5.66	22.00 dakika		

										LARIN				
olay no.	olay tipi	oluşma zamanı	olay apsisi	olay ordinatı	servis süresi	atanan araç	sonraki olay	araç 1	araç 2	araç 3	araç 4	araç 5		
1	yeni olay	22	16	37	27	1	63	41	41	41	41	41		
2	yeni olay	23	23	17	30	2	42	yolda	19	19	19	19		
3	yeni olay	28	20	33	27	3	57	yolda	yolda	29	29	29		
4	ulaşma	42	23	17	30	2	72	yolda	busy	yolda	idle	idle		
5	yeni olay	55	23	31	54	4	86	yolda	busy	yolda	31	31		
6	ulaşma	57	20	33	27	3	84	yolda	busy	busy	yolda	idle		
7	ulaşma	63	16	37	27	1	90	busy	busy	busy	yolda	idle		
8	tamamlama	72	23	17	30	2	ok	busy	idle	idle	yolda	idle		
9	tamamlama	84	20	33	27	3	ok	busy	idle	idle	yolda	idle		
10	ulaşma	86	23	31	54	4	140	busy	idle	idle	busy	idle		
11	tamamlama	90	16	37	27	1	ok	idle	idle	idle	busy	idle		
12	yeni olay	90	33	6	63	2	131	82	41	70	busy	50		
13	yeni olay	96	14	18	1	5	118	41	yolda	41	busy	22		
14	ulaşma	118	14	18	1	5	119	idle	yolda	idle	busy	busy		
15	tamamlama	119	14	18	1	5	ok	idle	yolda	idle	busy	idle		
16	yeni olay	120	10	8	1	5	151	62	yolda	62	busy	31		
17	yeni olay	127	19	26	30	3	149	31	yolda	22	busy	yolda		
18	ulaşma	131	33	6	63	2	194	idle	busy	yolda	busy	yolda		
19	tamamlama	140	23	31	54	4	ok	idle	busy	yolda	idle	yolda		
20	ulaşma	149	19	26	30	3	179	idle	busy	busy	idle	yolda		
21	ulaşma	151	10	8	1	5	152	idle	busy	busy	idle	busy		
22	tamamlama	152	10	8	1	5	ok	idle	busy	busy	idle	idle		
23	yeni olay	169	2	10	9	5	194	71	busy	busy	73	25		
24	tamamlama	179	19	26	30	3	ok	idle	busy	idle	idle	yolda		
25	ulaşma	194	2	10	9	5	203	idle	busy	idle	idle	busy		
26	tamamlama	194	33	6	63	2	ok	idle	idle	idle	idle	busy		
27	tamamlama	203	2	10	9	5	ok	idle	idle	idle	idle	idle		
28	yeni olay	205	37	14	66	2	233	50	28	55	56	68		
29	ulaşma	233	37	14	66	2	299	idle	busy	idle	idle	idle		
30	yeni olay	261	23	13	39	3	296	56	busy	35	37	46		
31	yeni olay	274	27	10	9	4	321	67	busy	yolda	47	47		
32	yeni olay	278	21	19	45	1	322	44	busy	yolda	yolda	71		
33	ulaşma	296	23	13	39	3	335	yolda	busy	busy	yolda	idle		
34	tamamlama	299	37	14	66	2	ok	yolda	idle	busy	yolda	idle		
35	yeni olay	310	39	24	1	2	338	yolda	28	busy	yolda	66		
36	yeni olay	317	25	32	70	5	394	yolda	yolda	busy	yolda	77		
37	yeni olay	320	13	25	87	4	335	yolda	yolda	busy	yolda	yolda		
38	ulaşma	321	27	10	9	4	330	yolda	yolda	busy	busy	yolda		
39	ulaşma	322	21	19	45	1	367	busy	yolda	busy	busy	yolda		
40	yeni olay	329	15	8	27	3	339	busy	yolda	busy	busy	yolda		
41	tamamlama	330	27	10	9	4	ok	busy	yolda	busy	idle	yolda		
42	bekleyen atama	330	13	25	87	4	383	busy	yolda	busy	yolda	yolda		
43	yeni olay	330	14	9	37	2	339	busy	yolda	busy	yolda	yolda		
44	tamamlama	335	23	13	39	3	ok	busy	yolda	idle	yolda	yolda		
45	bekleyen atama	335	15	8	27	3	364	busy	yolda	yolda	yolda	yolda		
46	yeni olay	336	24	0	50	1	367	busy	yolda	yolda	yolda	yolda		
47	ulaşma	338	39	24	1	2	339	busy	busy	yolda	yolda	yolda		
48	tamamlama	339	39	24	1	2	ok	busy	idle	yolda	yolda	yolda		
49	bekleyen atama	339	14	9	37	2	409	busy	yolda	yolda	yolda	yolda		
50	yeni olay	349	37	25	37	3	391	busy	yolda	yolda	yolda	yolda		
51	yeni olay	357	33	29	21	2	446	busy	yolda	yolda	yolda	yolda		
52	yeni olay	359	39	24	55	1	460	busy	yolda	yolda	yolda	yolda		
53	ulaşma	364	15	8	27	3	391	busy	yolda	busy	yolda	yolda		
54	tamamlama	367	21	19	45	1	ok	idle	yolda	busy	yolda	yolda		
55	bekleyen atama	367	24	0	50	1	410	yolda	yolda	busy	yolda	yolda		
56	yeni olay	368	19	29	1	5	464	yolda	yolda	busy	yolda	yolda		
57	ulaşma	383	13	25	87	4	470	yolda	yolda	busy	busy	yolda		
58	tamamlama	391	23	13	39	3	ok	yolda	yolda	idle	busy	yolda		
59	bekleyen atama	391	37	25	37	3	458	yolda	yolda	yolda	busy	yolda		
60	yeni olay	392	31	26	66	4	470	yolda	yolda	yolda	busy	yolda		
61	ulaşma	394	25	32	70	5	464	yolda	yolda	yolda	busy	busy		
62	bekleyen atama	403	37	25	37	3	452	yolda	yolda	yolda	busy	busy		
63	ulaşma	409	14	9	37	2	446	yolda	busy	yolda	busy	busy		
64	ulaşma	410	24	0	50	1	450	busy	busy	yolda	busy	busy		
65	yeni olay	451	11	26	15	5	488	busy	busy	yolda	busy	busy		
66	ulaşma	458	37	25	37	3	495	busy	busy	busy	busy	busy		
67	tamamlama	446	14	9	37	2	ok	busy	idle	busy	busy	busy		
68	bekleyen atama	446	33	29	21	2	514	busy	yolda	busy	busy	busy		
69	tamamlama	460	24	0	50	1	ok	idle	yolda	busy	busy	busy		
70	bekleyen atama	460	39	24	55	1	528	yolda	yolda	busy	busy	busy		
71	tamamlama	464	25	32	70	5	ok	yolda	yolda	busy	busy	idle		
72	bekleyen atama	464	19	29	1	5	487	yolda	yolda	busy	busy	yolda		
73	tamamlama	470	13	25	87	4	ok	yolda	yolda	busy	idle	yolda		
74	bekleyen atama	470	31	28	66	4	505	yolda	yolda	busy	yolda	yolda		
75	yeni olay	473	37	17	78	3	495	yolda	yolda	busy	yolda	yolda		
76	ulaşma	487	19	29	1	5	488	yolda	yolda	busy	yolda	busy		
77	tamamlama	488	19	29	1	5	ok	yolda	yolda	busy	yolda	idle		
78	bekleyen atama	488	11	26	15	5	514	yolda	yolda	busy	yolda	yolda		
79	tamamlama	495	37	25	37	3	ok	yolda	yolda	idle	yolda	yolda		
80	bekleyen atama	495	37	17	78	3	517	yolda	yolda	yolda	yolda	yolda		
81	ulaşma	505	31	28	66	4	571	yolda	yolda	yolda	busy	yolda		
82	ulaşma	514	33	29	21	2	535	yolda	busy	yolda	busy	busy		
83	ulaşma	514	11	26	15	5	529	yolda	busy	yolda	busy	busy		
84	ulaşma	517	37	17	78	3	595	yolda	busy	busy	busy	busy		
85	ulaşma	528	39	24	55	1	583	busy	busy	busy	busy	busy		
86	tamamlama	529	11	26	15	5	ok	busy	busy	busy	busy	busy		
87	tamamlama	535	33	29	21	2	ok	busy	busy	busy	busy	idle		
88	tamamlama	571	31	28	66	4	ok	busy	idle	busy	idle	idle		
89	tamamlama	583	39	24	55	1	ok	idle	idle	busy	idle	idle		
90	tamamlama	595	37	17	78	3	ok	idle	idle	idle	idle	idle		

**Ek 6 Hafif Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu
(m=5)**

Rapor(0-480)

Olay sayisi=5

Ortalama bekleme zamanı=31.800000

Arac basına alınan yol=17.800000

Rapor(0-960)

Olay sayisi=9

Ortalama bekleme zamanı=30.222222

Arac basına alınan yol=24.800000

Rapor(0-1440)

Olay sayisi=21

Ortalama bekleme zamanı=23.809524

Arac basına alınan yol=49.600000

Rapor(0-1920)

Olay sayisi=27

Ortalama bekleme zamanı=25.333333

Arac basına alınan yol=61.800000

Rapor(0-2400)

Olay sayisi=32

Ortalama bekleme zamanı=25.468750

Arac basına alınan yol=69.600000

Rapor(0-2880)

Olay sayisi=39

Ortalama bekleme zamanı=24.769231

Arac basına alınan yol=81.000000

Rapor(0-3360)

Olay sayisi=47

Ortalama bekleme zamanı=25.063830

Arac basına alınan yol=99.800000

Rapor(0-3840)

Olay sayisi=50

Ortalama bekleme zamanı=25.480000

Arac basına alınan yol=108.800000

Rapor(0-4320)

Olay sayisi=57

Ortalama bekleme zamanı=25.210526

Arac basına alınan yol=121.600000

Rapor(0-4800)

Olay sayisi=66

Ortalama bekleme zamanı=26.318182

Arac basına alınan yol=156.200000

Rapor(0-5280)
Olay sayisi=78
Ortalama bekleme zamani=26.807692
Arac basina alinan yol=185.800000

Rapor(0-5760)
Olay sayisi=90
Ortalama bekleme zamani=26.911111
Arac basina alinan yol=213.800000

Rapor(0-6240)
Olay sayisi=97
Ortalama bekleme zamani=26.793814
Arac basina alinan yol=228.800000

Rapor(0-6720)
Olay sayisi=105
Ortalama bekleme zamani=27.247619
Arac basina alinan yol=250.600000

Rapor(0-7200)
Olay sayisi=114
Ortalama bekleme zamani=27.701754
Arac basina alinan yol=278.600000

Rapor(0-7680)
Olay sayisi=123
Ortalama bekleme zamani=28.373984
Arac basina alinan yol=311.200000

Rapor(0-8160)
Olay sayisi=128
Ortalama bekleme zamani=28.343750
Arac basina alinan yol=323.200000

Rapor(0-8640)
Olay sayisi=131
Ortalama bekleme zamani=28.198473
Arac basina alinan yol=328.400000

Rapor(0-9120)
Olay sayisi=140
Ortalama bekleme zamani=27.942857
Arac basina alinan yol=346.000000

Rapor(0-9600)
Olay sayisi=153
Ortalama bekleme zamani=28.032680
Arac basina alinan yol=389.000000

Rapor(0-10080)
Olay sayisi=157
Ortalama bekleme zamani=28.292994
Arac basina alinan yol=395.400000

Rapor(0-10560)
Olay sayisi=163
Ortalama bekleme zamani=28.214724
Arac basina alinan yol=408.800000

Rapor(0-11040)
Olay sayisi=173
Ortalama bekleme zamani=27.843931
Arac basina alinan yol=425.800000

Rapor(0-11520)
Olay sayisi=181
Ortalama bekleme zamani=27.281768
Arac basina alinan yol=435.800000

Rapor(0-12000)
Olay sayisi=185
Ortalama bekleme zamani=27.464865
Arac basina alinan yol=446.400000

Rapor(0-12480)
Olay sayisi=189
Ortalama bekleme zamani=27.529101
Arac basina alinan yol=457.600000

Rapor(0-12960)
Olay sayisi=192
Ortalama bekleme zamani=27.359375
Arac basina alinan yol=460.800000

Rapor(0-13440)
Olay sayisi=199
Ortalama bekleme zamani=27.211055
Arac basina alinan yol=473.600000

Rapor(0-13920)
Olay sayisi=207
Ortalama bekleme zamani=26.893720
Arac basina alinan yol=486.400000

Rapor(0-14400)
Olay sayisi=217
Ortalama bekleme zamani=27.262673
Arac basina alinan yol=517.600000

Rapor(0-14880)
Olay sayisi=223
Ortalama bekleme zamani=27.040359
Arac basina alinan yol=531.000000

Rapor(0-15360)
Olay sayisi=237
Ortalama bekleme zamani=27.409283
Arac basina alinan yol=574.200000

Rapor(0-15840)
Olay sayisi=244
Ortalama bekleme zamani=27.852459
Arac basina alinan yol=600.400000

Rapor(0-16320)
Olay sayisi=256
Ortalama bekleme zamani=27.886719
Arac basina alinan yol=655.800000

Rapor(0-16800)
Olay sayisi=266
Ortalama bekleme zamani=28.951128
Arac basina alinan yol=693.000000

Rapor(0-17280)
Olay sayisi=275
Ortalama bekleme zamani=28.745455
Arac basina alinan yol=714.600000

Rapor(0-17760)
Olay sayisi=281
Ortalama bekleme zamani=28.836299
Arac basina alinan yol=727.800000

Rapor(0-18240)
Olay sayisi=287
Ortalama bekleme zamani=28.825784
Arac basina alinan yol=747.400000

Rapor(0-18720)
Olay sayisi=293
Ortalama bekleme zamani=29.150171
Arac basina alinan yol=770.800000

Rapor(0-19200)
Olay sayisi=304
Ortalama bekleme zamani=29.059211
Arac basina alinan yol=796.200000

Rapor(0-19680)
Olay sayisi=315
Ortalama bekleme zamani=29.098413
Arac basina alinan yol=826.800000

Rapor(0-20160)
Olay sayisi=331
Ortalama bekleme zamani=29.211480
Arac basina alinan yol=879.000000

Rapor(0-20640)
Olay sayisi=339
Ortalama bekleme zamani=29.321534
Arac basina alinan yol=899.800000

Rapor(0-21120)
Olay sayisi=344
Ortalama bekleme zamani=29.267442
Arac basina alinan yol=910.600000

Rapor(0-21600)
Olay sayisi=353
Ortalama bekleme zamani=29.260623
Arac basina alinan yol=937.200000

Rapor(0-22080)
Olay sayisi=361
Ortalama bekleme zamani=29.216066
Arac basina alinan yol=953.400000

Rapor(0-22560)
Olay sayisi=370
Ortalama bekleme zamani=29.310811
Arac basina alinan yol=981.600000

Rapor(0-23040)
Olay sayisi=376
Ortalama bekleme zamani=29.263298
Arac basina alinan yol=995.400000

Rapor(0-23520)
Olay sayisi=381
Ortalama bekleme zamani=29.091864
Arac basina alinan yol=1004.800000

Rapor(0-24000)
Olay sayisi=391
Ortalama bekleme zamani=29.199488
Arac basina alinan yol=1031.800000

**Ek 7 Hafif Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu
(m=5)**

Rapor(0-480)

Olay sayisi=10

Ortalama bekleme zamani=30.100000

Arac basina alinan yol=64.000000

Rapor(0-960)

Olay sayisi=23

Ortalama bekleme zamani=36.304348

Arac basina alinan yol=151.200000

Rapor(0-1440)

Olay sayisi=28

Ortalama bekleme zamani=35.714286

Arac basina alinan yol=191.200000

Rapor(0-1920)

Olay sayisi=40

Ortalama bekleme zamani=34.125000

Arac basina alinan yol=264.600000

Rapor(0-2400)

Olay sayisi=49

Ortalama bekleme zamani=34.755102

Arac basina alinan yol=325.600000

Rapor(0-2880)

Olay sayisi=58

Ortalama bekleme zamani=33.724138

Arac basina alinan yol=367.000000

Rapor(0-3360)

Olay sayisi=65

Ortalama bekleme zamani=32.953846

Arac basina alinan yol=410.800000

Rapor(0-3840)

Olay sayisi=76

Ortalama bekleme zamani=33.894737

Arac basina alinan yol=492.400000

Rapor(0-4320)

Olay sayisi=80

Ortalama bekleme zamani=34.237500

Arac basina alinan yol=518.800000

Rapor(0-4800)

Olay sayisi=90

Ortalama bekleme zamani=33.755556

Arac basina alinan yol=584.800000

Rapor(0-5280)
Olay sayisi=96
Ortalama bekleme zamani=34.239583
Arac basina alinan yol=627.200000

Rapor(0-5760)
Olay sayisi=103
Ortalama bekleme zamani=34.174757
Arac basina alinan yol=671.400000

Rapor(0-6240)
Olay sayisi=109
Ortalama bekleme zamani=34.394495
Arac basina alinan yol=715.200000

Rapor(0-6720)
Olay sayisi=120
Ortalama bekleme zamani=34.541667
Arac basina alinan yol=796.000000

Rapor(0-7200)
Olay sayisi=131
Ortalama bekleme zamani=33.740458
Arac basina alinan yol=839.200000

Rapor(0-7680)
Olay sayisi=138
Ortalama bekleme zamani=33.427536
Arac basina alinan yol=883.600000

Rapor(0-8160)
Olay sayisi=143
Ortalama bekleme zamani=33.727273
Arac basina alinan yol=921.800000

Rapor(0-8640)
Olay sayisi=153
Ortalama bekleme zamani=33.830065
Arac basina alinan yol=983.000000

Rapor(0-9120)
Olay sayisi=157
Ortalama bekleme zamani=33.764331
Arac basina alinan yol=1012.200000

Rapor(0-9600)
Olay sayisi=166
Ortalama bekleme zamani=33.795181
Arac basina alinan yol=1067.600000

Rapor(0-10080)
Olay sayisi=174
Ortalama bekleme zamani=33.908046
Arac basina alinan yol=1124.400000

Rapor(0-10560)
Olay sayisi=183
Ortalama bekleme zamani=33.994536
Arac basina alinan yol=1192.400000

Rapor(0-11040)
Olay sayisi=189
Ortalama bekleme zamani=33.846561
Arac basina alinan yol=1217.200000

Rapor(0-11520)
Olay sayisi=196
Ortalama bekleme zamani=33.816327
Arac basina alinan yol=1260.800000

Rapor(0-12000)
Olay sayisi=207
Ortalama bekleme zamani=33.347826
Arac basina alinan yol=1316.800000

Rapor(0-12480)
Olay sayisi=213
Ortalama bekleme zamani=33.431925
Arac basina alinan yol=1350.400000

Rapor(0-12960)
Olay sayisi=219
Ortalama bekleme zamani=33.479452
Arac basina alinan yol=1391.200000

Rapor(0-13440)
Olay sayisi=223
Ortalama bekleme zamani=33.363229
Arac basina alinan yol=1410.000000

Rapor(0-13920)
Olay sayisi=228
Ortalama bekleme zamani=33.307018
Arac basina alinan yol=1438.000000

Rapor(0-14400)
Olay sayisi=235
Ortalama bekleme zamani=33.268085
Arac basina alinan yol=1477.600000

Rapor(0-14880)
 Olay sayisi=248
 Ortalama bekleme zamani=32.915323
 Arac basina alinan yol=1552.600000

Rapor(0-15360)
 Olay sayisi=255
 Ortalama bekleme zamani=33.003922
 Arac basina alinan yol=1588.800000

Rapor(0-15840)
 Olay sayisi=262
 Ortalama bekleme zamani=32.702290
 Arac basina alinan yol=1625.600000

Rapor(0-16320)
 Olay sayisi=268
 Ortalama bekleme zamani=32.958955
 Arac basina alinan yol=1661.800000

Rapor(0-16800)
 Olay sayisi=279
 Ortalama bekleme zamani=32.670251
 Arac basina alinan yol=1725.800000

Rapor(0-17280)
 Olay sayisi=286
 Ortalama bekleme zamani=32.797203
 Arac basina alinan yol=1771.200000

Rapor(0-17760)
 Olay sayisi=290
 Ortalama bekleme zamani=32.917241
 Arac basina alinan yol=1801.800000

Rapor(0-18240)
 Olay sayisi=294
 Ortalama bekleme zamani=33.129252
 Arac basina alinan yol=1830.800000

Rapor(0-18720)
 Olay sayisi=300
 Ortalama bekleme zamani=32.953333
 Arac basina alinan yol=1867.400000

Rapor(0-19200)
 Olay sayisi=305
 Ortalama bekleme zamani=32.911475
 Arac basina alinan yol=1899.400000

Rapor(0-19680)
Olay sayisi=311
Ortalama bekleme zamani=32.810289
Arac basina alinan yol=1929.600000

Rapor(0-20160)
Olay sayisi=315
Ortalama bekleme zamani=32.971429
Arac basina alinan yol=1959.600000

Rapor(0-20640)
Olay sayisi=323
Ortalama bekleme zamani=33.040248
Arac basina alinan yol=2015.200000

Rapor(0-21120)
Olay sayisi=331
Ortalama bekleme zamani=32.836858
Arac basina alinan yol=2050.600000

Rapor(0-21600)
Olay sayisi=334
Ortalama bekleme zamani=32.898204
Arac basina alinan yol=2071.600000

Rapor(0-22080)
Olay sayisi=343
Ortalama bekleme zamani=32.842566
Arac basina alinan yol=2127.400000

Rapor(0-22560)
Olay sayisi=353
Ortalama bekleme zamani=33.079320
Arac basina alinan yol=2206.800000

Rapor(0-23040)
Olay sayisi=363
Ortalama bekleme zamani=32.909091
Arac basina alinan yol=2263.000000

Rapor(0-23520)
Olay sayisi=368
Ortalama bekleme zamani=33.024457
Arac basina alinan yol=2294.800000

Rapor(0-24000)
Olay sayisi=376
Ortalama bekleme zamani=33.002660
Arac basina alinan yol=2350.600000

**Ek 8 Hafif Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu
(m=5)**

Rapor(0-480)

Olay sayisi=6

Ortalama bekleme zamanı=30.000000

Arac basına alınan yol=19.400000

Rapor(0-960)

Olay sayisi=13

Ortalama bekleme zamanı=29.153846

Arac basına alınan yol=33.800000

Rapor(0-1440)

Olay sayisi=21

Ortalama bekleme zamanı=27.142857

Arac basına alınan yol=55.600000

Rapor(0-1920)

Olay sayisi=29

Ortalama bekleme zamanı=30.724138

Arac basına alınan yol=81.400000

Rapor(0-2400)

Olay sayisi=34

Ortalama bekleme zamanı=30.764706

Arac basına alınan yol=93.400000

Rapor(0-2880)

Olay sayisi=50

Ortalama bekleme zamanı=31.480000

Arac basına alınan yol=143.400000

Rapor(0-3360)

Olay sayisi=56

Ortalama bekleme zamanı=31.517857

Arac basına alınan yol=156.200000

Rapor(0-3840)

Olay sayisi=68

Ortalama bekleme zamanı=29.852941

Arac basına alınan yol=175.800000

Rapor(0-4320)

Olay sayisi=81

Ortalama bekleme zamanı=29.691358

Arac basına alınan yol=213.000000

Rapor(0-4800)

Olay sayisi=88

Ortalama bekleme zamanı=30.511364

Arac basına alınan yol=232.600000

Rapor(0-5280)
Olay sayisi=95
Ortalama bekleme zamani=30.221053
Arac basina alinan yol=248.800000

Rapor(0-5760)
Olay sayisi=105
Ortalama bekleme zamani=29.457143
Arac basina alinan yol=278.000000

Rapor(0-6240)
Olay sayisi=117
Ortalama bekleme zamani=30.239316
Arac basina alinan yol=306.400000

Rapor(0-6720)
Olay sayisi=121
Ortalama bekleme zamani=30.644628
Arac basina alinan yol=317.400000

Rapor(0-7200)
Olay sayisi=131
Ortalama bekleme zamani=30.389313
Arac basina alinan yol=349.000000

Rapor(0-7680)
Olay sayisi=134
Ortalama bekleme zamani=30.783582
Arac basina alinan yol=356.200000

Rapor(0-8160)
Olay sayisi=140
Ortalama bekleme zamani=30.814286
Arac basina alinan yol=373.600000

Rapor(0-8640)
Olay sayisi=146
Ortalama bekleme zamani=30.520548
Arac basina alinan yol=385.200000

Rapor(0-9120)
Olay sayisi=154
Ortalama bekleme zamani=30.058442
Arac basina alinan yol=398.400000

Rapor(0-9600)
Olay sayisi=160
Ortalama bekleme zamani=29.837500
Arac basina alinan yol=410.200000

Rapor(0-10080)
Olay sayisi=167
Ortalama bekleme zamani=29.365269
Arac basina alinan yol=421.600000

Rapor(0-10560)
Olay sayisi=180
Ortalama bekleme zamani=29.077778
Arac basina alinan yol=455.000000

Rapor(0-11040)
Olay sayisi=189
Ortalama bekleme zamani=29.089947
Arac basina alinan yol=471.400000

Rapor(0-11520)
Olay sayisi=200
Ortalama bekleme zamani=28.940000
Arac basina alinan yol=496.400000

Rapor(0-12000)
Olay sayisi=208
Ortalama bekleme zamani=28.812500
Arac basina alinan yol=510.600000

Rapor(0-12480)
Olay sayisi=217
Ortalama bekleme zamani=28.400922
Arac basina alinan yol=526.400000

Rapor(0-12960)
Olay sayisi=225
Ortalama bekleme zamani=28.288889
Arac basina alinan yol=542.200000

Rapor(0-13440)
Olay sayisi=233
Ortalama bekleme zamani=28.068670
Arac basina alinan yol=554.000000

Rapor(0-13920)
Olay sayisi=242
Ortalama bekleme zamani=27.958678
Arac basina alinan yol=575.000000

Rapor(0-14400)
Olay sayisi=248
Ortalama bekleme zamani=28.112903
Arac basina alinan yol=589.800000

Rapor(0-14880)
Olay sayisi=256
Ortalama bekleme zamani=27.890625
Arac basina alinan yol=602.200000

Rapor(0-15360)
Olay sayisi=265
Ortalama bekleme zamani=27.615094
Arac basina alinan yol=620.000000

Rapor(0-15840)
Olay sayisi=272
Ortalama bekleme zamani=27.783088
Arac basina alinan yol=637.000000

Rapor(0-16320)
Olay sayisi=282
Ortalama bekleme zamani=27.670213
Arac basina alinan yol=662.200000

Rapor(0-16800)
Olay sayisi=289
Ortalama bekleme zamani=27.674740
Arac basina alinan yol=676.400000

Rapor(0-17280)
Olay sayisi=297
Ortalama bekleme zamani=27.558923
Arac basina alinan yol=692.400000

Rapor(0-17760)
Olay sayisi=304
Ortalama bekleme zamani=27.480263
Arac basina alinan yol=703.000000

Rapor(0-18240)
Olay sayisi=311
Ortalama bekleme zamani=27.369775
Arac basina alinan yol=718.600000

Rapor(0-18720)
Olay sayisi=315
Ortalama bekleme zamani=27.400000
Arac basina alinan yol=728.200000

Rapor(0-19200)
Olay sayisi=317
Ortalama bekleme zamani=27.444795
Arac basina alinan yol=731.000000

Rapor(0-19680)
Olay sayisi=326
Ortalama bekleme zamani=27.506135
Arac basina alinan yol=754.400000

Rapor(0-20160)
Olay sayisi=330
Ortalama bekleme zamani=27.490909
Arac basina alinan yol=763.200000

Rapor(0-20640)
Olay sayisi=333
Ortalama bekleme zamani=27.447447
Arac basina alinan yol=769.000000

Rapor(0-21120)
Olay sayisi=343
Ortalama bekleme zamani=27.731778
Arac basina alinan yol=803.000000

Rapor(0-21600)
Olay sayisi=348
Ortalama bekleme zamani=27.681034
Arac basina alinan yol=812.800000

Rapor(0-22080)
Olay sayisi=363
Ortalama bekleme zamani=27.589532
Arac basina alinan yol=855.800000

Rapor(0-22560)
Olay sayisi=370
Ortalama bekleme zamani=27.821622
Arac basina alinan yol=874.600000

Rapor(0-23040)
Olay sayisi=381
Ortalama bekleme zamani=28.091864
Arac basina alinan yol=910.800000

Rapor(0-23520)
Olay sayisi=387
Ortalama bekleme zamani=28.219638
Arac basina alinan yol=930.000000

Rapor(0-24000)
Olay sayisi=398
Ortalama bekleme zamani=28.158291
Arac basina alinan yol=958.000000

Ek 9 Az Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)

Rapor(0-480)

Olay sayisi=12

Ortalama bekleme zamani=27.333333

Arac basina alinan yol=32.000000

Rapor(0-960)

Olay sayisi=28

Ortalama bekleme zamani=28.071429

Arac basina alinan yol=73.800000

Rapor(0-1440)

Olay sayisi=46

Ortalama bekleme zamani=31.695652

Arac basina alinan yol=131.800000

Rapor(0-1920)

Olay sayisi=61

Ortalama bekleme zamani=31.737705

Arac basina alinan yol=174.600000

Rapor(0-2400)

Olay sayisi=75

Ortalama bekleme zamani=32.546667

Arac basina alinan yol=226.800000

Rapor(0-2880)

Olay sayisi=91

Ortalama bekleme zamani=31.549451

Arac basina alinan yol=260.000000

Rapor(0-3360)

Olay sayisi=110

Ortalama bekleme zamani=31.563636

Arac basina alinan yol=328.400000

Rapor(0-3840)

Olay sayisi=125

Ortalama bekleme zamani=31.912000

Arac basina alinan yol=368.000000

Rapor(0-4320)

Olay sayisi=148

Ortalama bekleme zamani=33.216216

Arac basina alinan yol=447.000000

Rapor(0-4800)

Olay sayisi=162

Ortalama bekleme zamani=32.654321

Arac basina alinan yol=475.600000

Rapor(0-5280)
Olay sayisi=175
Ortalama bekleme zamani=32.462857
Arac basina alinan yol=521.000000

Rapor(0-5760)
Olay sayisi=192
Ortalama bekleme zamani=32.843750
Arac basina alinan yol=577.600000

Rapor(0-6240)
Olay sayisi=211
Ortalama bekleme zamani=32.203791
Arac basina alinan yol=625.000000

Rapor(0-6720)
Olay sayisi=226
Ortalama bekleme zamani=32.314159
Arac basina alinan yol=682.000000

Rapor(0-7200)
Olay sayisi=240
Ortalama bekleme zamani=32.562500
Arac basina alinan yol=713.000000

Rapor(0-7680)
Olay sayisi=259
Ortalama bekleme zamani=31.613900
Arac basina alinan yol=753.400000

Rapor(0-8160)
Olay sayisi=274
Ortalama bekleme zamani=31.908759
Arac basina alinan yol=800.000000

Rapor(0-8640)
Olay sayisi=290
Ortalama bekleme zamani=31.844828
Arac basina alinan yol=840.000000

Rapor(0-9120)
Olay sayisi=306
Ortalama bekleme zamani=31.852941
Arac basina alinan yol=894.400000

Rapor(0-9600)
Olay sayisi=323
Ortalama bekleme zamani=31.653251
Arac basina alinan yol=934.000000

Rapor(0-10080)
Olay sayisi=336
Ortalama bekleme zamani=31.681548
Arac basina alinan yol=978.000000

Rapor(0-10560)
Olay sayisi=355
Ortalama bekleme zamani=31.904225
Arac basina alinan yol=1036.600000

Rapor(0-11040)
Olay sayisi=369
Ortalama bekleme zamani=31.485095
Arac basina alinan yol=1069.400000

Rapor(0-11520)
Olay sayisi=386
Ortalama bekleme zamani=31.686528
Arac basina alinan yol=1119.800000

Rapor(0-12000)
Olay sayisi=401
Ortalama bekleme zamani=31.533666
Arac basina alinan yol=1153.600000

Rapor(0-12480)
Olay sayisi=417
Ortalama bekleme zamani=31.352518
Arac basina alinan yol=1191.200000

Rapor(0-12960)
Olay sayisi=435
Ortalama bekleme zamani=31.259770
Arac basina alinan yol=1253.600000

Rapor(0-13440)
Olay sayisi=446
Ortalama bekleme zamani=31.576233
Arac basina alinan yol=1286.200000

Rapor(0-13920)
Olay sayisi=466
Ortalama bekleme zamani=31.652361
Arac basina alinan yol=1363.000000

Rapor(0-14400)
Olay sayisi=479
Ortalama bekleme zamani=31.983299
Arac basina alinan yol=1406.600000

Rapor(0-14880)
 Olay sayisi=488
 Ortalama bekleme zamani=31.946721
 Arac basina alinan yol=1430.800000

Rapor(0-15360)
 Olay sayisi=505
 Ortalama bekleme zamani=31.843564
 Arac basina alinan yol=1485.400000

Rapor(0-15840)
 Olay sayisi=519
 Ortalama bekleme zamani=31.886320
 Arac basina alinan yol=1520.600000

Rapor(0-16320)
 Olay sayisi=534
 Ortalama bekleme zamani=31.711610
 Arac basina alinan yol=1564.600000

Rapor(0-16800)
 Olay sayisi=551
 Ortalama bekleme zamani=31.705989
 Arac basina alinan yol=1610.400000

Rapor(0-17280)
 Olay sayisi=566
 Ortalama bekleme zamani=31.952297
 Arac basina alinan yol=1660.000000

Rapor(0-17760)
 Olay sayisi=578
 Ortalama bekleme zamani=31.977509
 Arac basina alinan yol=1697.800000

Rapor(0-18240)
 Olay sayisi=601
 Ortalama bekleme zamani=32.371048
 Arac basina alinan yol=1782.600000

Rapor(0-18720)
 Olay sayisi=611
 Ortalama bekleme zamani=32.279869
 Arac basina alinan yol=1803.400000

Rapor(0-19200)
 Olay sayisi=624
 Ortalama bekleme zamani=32.423077
 Arac basina alinan yol=1851.000000

Rapor(0-19680)
Olay sayisi=637
Ortalama bekleme zamani=32.251177
Arac basina alinan yol=1876.400000

Rapor(0-20160)
Olay sayisi=656
Ortalama bekleme zamani=32.243902
Arac basina alinan yol=1949.800000

Rapor(0-20640)
Olay sayisi=679
Ortalama bekleme zamani=32.883652
Arac basina alinan yol=2035.400000

Rapor(0-21120)
Olay sayisi=692
Ortalama bekleme zamani=33.036127
Arac basina alinan yol=2076.800000

Rapor(0-21600)
Olay sayisi=708
Ortalama bekleme zamani=32.953390
Arac basina alinan yol=2119.400000

Rapor(0-22080)
Olay sayisi=724
Ortalama bekleme zamani=32.792818
Arac basina alinan yol=2158.600000

Rapor(0-22560)
Olay sayisi=747
Ortalama bekleme zamani=33.009371
Arac basina alinan yol=2254.600000

Rapor(0-23040)
Olay sayisi=758
Ortalama bekleme zamani=33.036939
Arac basina alinan yol=2287.800000

Rapor(0-23520)
Olay sayisi=774
Ortalama bekleme zamani=33.220930
Arac basina alinan yol=2344.600000

Rapor(0-24000)
Olay sayisi=791
Ortalama bekleme zamani=33.109987
Arac basina alinan yol=2388.000000

Ek 10 Az Yoğun Trafik Durumunda SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)

Rapor(0-480)

Olay sayısı=18

Ortalama bekleme zamanı=32.888889

Arac basına alınan yol=84.400000

Rapor(0-960)

Olay sayısı=31

Ortalama bekleme zamanı=31.258065

Arac basına alınan yol=157.400000

Rapor(0-1440)

Olay sayısı=46

Ortalama bekleme zamanı=31.434783

Arac basına alınan yol=247.000000

Rapor(0-1920)

Olay sayısı=65

Ortalama bekleme zamanı=32.630769

Arac basına alınan yol=353.400000

Rapor(0-2400)

Olay sayısı=78

Ortalama bekleme zamanı=32.102564

Arac basına alınan yol=421.600000

Rapor(0-2880)

Olay sayısı=92

Ortalama bekleme zamanı=33.728261

Arac basına alınan yol=511.600000

Rapor(0-3360)

Olay sayısı=106

Ortalama bekleme zamanı=32.462264

Arac basına alınan yol=576.200000

Rapor(0-3840)

Olay sayısı=123

Ortalama bekleme zamanı=32.886179

Arac basına alınan yol=672.200000

Rapor(0-4320)

Olay sayısı=134

Ortalama bekleme zamanı=33.246269

Arac basına alınan yol=754.400000

Rapor(0-4800)

Olay sayısı=150

Ortalama bekleme zamanı=34.573333

Arac basına alınan yol=848.400000

Rapor(0-5280)
Olay sayisi=160
Ortalama bekleme zamani=33.693750
Arac basina alinan yol=888.200000

Rapor(0-5760)
Olay sayisi=172
Ortalama bekleme zamani=34.052326
Arac basina alinan yol=972.600000

Rapor(0-6240)
Olay sayisi=193
Ortalama bekleme zamani=33.082902
Arac basina alinan yol=1072.200000

Rapor(0-6720)
Olay sayisi=214
Ortalama bekleme zamani=35.144860
Arac basina alinan yol=1187.800000

Rapor(0-7200)
Olay sayisi=233
Ortalama bekleme zamani=35.180258
Arac basina alinan yol=1284.000000

Rapor(0-7680)
Olay sayisi=246
Ortalama bekleme zamani=35.979675
Arac basina alinan yol=1362.800000

Rapor(0-8160)
Olay sayisi=257
Ortalama bekleme zamani=35.910506
Arac basina alinan yol=1427.800000

Rapor(0-8640)
Olay sayisi=272
Ortalama bekleme zamani=35.952206
Arac basina alinan yol=1526.000000

Rapor(0-9120)
Olay sayisi=285
Ortalama bekleme zamani=35.905263
Arac basina alinan yol=1611.200000

Rapor(0-9600)
Olay sayisi=303
Ortalama bekleme zamani=35.805281
Arac basina alinan yol=1692.000000

Rapor(0-10080)
Olay sayisi=323
Ortalama bekleme zamani=36.126935
Arac basina alinan yol=1814.200000

Rapor(0-10560)
Olay sayisi=336
Ortalama bekleme zamani=35.967262
Arac basina alinan yol=1900.800000

Rapor(0-11040)
Olay sayisi=348
Ortalama bekleme zamani=36.143678
Arac basina alinan yol=1989.400000

Rapor(0-11520)
Olay sayisi=370
Ortalama bekleme zamani=36.400000
Arac basina alinan yol=2094.400000

Rapor(0-12000)
Olay sayisi=386
Ortalama bekleme zamani=35.961140
Arac basina alinan yol=2181.000000

Rapor(0-12480)
Olay sayisi=397
Ortalama bekleme zamani=36.267003
Arac basina alinan yol=2276.400000

Rapor(0-12960)
Olay sayisi=415
Ortalama bekleme zamani=36.551807
Arac basina alinan yol=2386.400000

Rapor(0-13440)
Olay sayisi=437
Ortalama bekleme zamani=36.768879
Arac basina alinan yol=2506.000000

Rapor(0-13920)
Olay sayisi=448
Ortalama bekleme zamani=37.080357
Arac basina alinan yol=2575.000000

Rapor(0-14400)
Olay sayisi=468
Ortalama bekleme zamani=36.912393
Arac basina alinan yol=2679.000000

Rapor(0-14880)
Olay sayisi=484
Ortalama bekleme zamani=36.925620
Arac basina alinan yol=2785.200000

Rapor(0-15360)
Olay sayisi=504
Ortalama bekleme zamani=36.775794
Arac basina alinan yol=2898.400000

Rapor(0-15840)
Olay sayisi=517
Ortalama bekleme zamani=36.512573
Arac basina alinan yol=2953.200000

Rapor(0-16320)
Olay sayisi=536
Ortalama bekleme zamani=36.354478
Arac basina alinan yol=3061.600000

Rapor(0-16800)
Olay sayisi=547
Ortalama bekleme zamani=36.510055
Arac basina alinan yol=3137.000000

Rapor(0-17280)
Olay sayisi=559
Ortalama bekleme zamani=36.375671
Arac basina alinan yol=3197.600000

Rapor(0-17760)
Olay sayisi=573
Ortalama bekleme zamani=36.425829
Arac basina alinan yol=3299.600000

Rapor(0-18240)
Olay sayisi=596
Ortalama bekleme zamani=36.159396
Arac basina alinan yol=3412.400000

Rapor(0-18720)
Olay sayisi=609
Ortalama bekleme zamani=36.210181
Arac basina alinan yol=3480.800000

Rapor(0-19200)
Olay sayisi=628
Ortalama bekleme zamani=35.944268
Arac basina alinan yol=3573.400000

Rapor(0-19680)
Olay sayisi=647
Ortalama bekleme zamani=36.445131
Arac basina alinan yol=3708.000000

Rapor(0-20160)
Olay sayisi=669
Ortalama bekleme zamani=36.399103
Arac basina alinan yol=3803.200000

Rapor(0-20640)
Olay sayisi=692
Ortalama bekleme zamani=37.096821
Arac basina alinan yol=3927.800000

Rapor(0-21120)
Olay sayisi=708
Ortalama bekleme zamani=37.152542
Arac basina alinan yol=4045.200000

Rapor(0-21600)
Olay sayisi=718
Ortalama bekleme zamani=37.072423
Arac basina alinan yol=4090.600000

Rapor(0-22080)
Olay sayisi=737
Ortalama bekleme zamani=37.051560
Arac basina alinan yol=4219.400000

Rapor(0-22560)
Olay sayisi=761
Ortalama bekleme zamani=37.687254
Arac basina alinan yol=4329.600000

Rapor(0-23040)
Olay sayisi=772
Ortalama bekleme zamani=37.669689
Arac basina alinan yol=4394.600000

Rapor(0-23520)
Olay sayisi=785
Ortalama bekleme zamani=37.789809
Arac basina alinan yol=4481.400000

Rapor(0-24000)
Olay sayisi=796
Ortalama bekleme zamani=37.721106
Arac basina alinan yol=4548.200000

Ek 11 Az Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu (m=5)

Rapor(0-480)

Olay sayisi=21

Ortalama bekleme zamani=29.666667

Arac basina alinan yol=61.200000

Rapor(0-960)

Olay sayisi=40

Ortalama bekleme zamani=34.825000

Arac basina alinan yol=118.800000

Rapor(0-1440)

Olay sayisi=55

Ortalama bekleme zamani=33.890909

Arac basina alinan yol=154.400000

Rapor(0-1920)

Olay sayisi=76

Ortalama bekleme zamani=31.052632

Arac basina alinan yol=194.000000

Rapor(0-2400)

Olay sayisi=91

Ortalama bekleme zamani=30.846154

Arac basina alinan yol=236.200000

Rapor(0-2880)

Olay sayisi=110

Ortalama bekleme zamani=30.081818

Arac basina alinan yol=289.400000

Rapor(0-3360)

Olay sayisi=131

Ortalama bekleme zamani=30.229008

Arac basina alinan yol=350.800000

Rapor(0-3840)

Olay sayisi=144

Ortalama bekleme zamani=34.486111

Arac basina alinan yol=386.000000

Rapor(0-4320)

Olay sayisi=160

Ortalama bekleme zamani=34.000000

Arac basina alinan yol=433.800000

Rapor(0-4800)

Olay sayisi=173

Ortalama bekleme zamani=34.213873

Arac basina alinan yol=475.800000

Rapor(0-5280)
Olay sayisi=187
Ortalama bekleme zamani=33.598930
Arac basina alinan yol=514.200000

Rapor(0-5760)
Olay sayisi=198
Ortalama bekleme zamani=33.722222
Arac basina alinan yol=542.200000

Rapor(0-6240)
Olay sayisi=215
Ortalama bekleme zamani=33.051163
Arac basina alinan yol=583.000000

Rapor(0-6720)
Olay sayisi=227
Ortalama bekleme zamani=32.995595
Arac basina alinan yol=607.800000

Rapor(0-7200)
Olay sayisi=248
Ortalama bekleme zamani=33.032258
Arac basina alinan yol=679.200000

Rapor(0-7680)
Olay sayisi=261
Ortalama bekleme zamani=33.478927
Arac basina alinan yol=713.200000

Rapor(0-8160)
Olay sayisi=274
Ortalama bekleme zamani=33.164234
Arac basina alinan yol=746.600000

Rapor(0-8640)
Olay sayisi=292
Ortalama bekleme zamani=33.171233
Arac basina alinan yol=784.800000

Rapor(0-9120)
Olay sayisi=307
Ortalama bekleme zamani=32.690554
Arac basina alinan yol=813.600000

Rapor(0-9600)
Olay sayisi=320
Ortalama bekleme zamani=32.812500
Arac basina alinan yol=845.200000

Rapor(0-10080)
Olay sayisi=339
Ortalama bekleme zamani=32.675516
Arac basina alinan yol=912.600000

Rapor(0-10560)
Olay sayisi=358
Ortalama bekleme zamani=33.159218
Arac basina alinan yol=964.800000

Rapor(0-11040)
Olay sayisi=375
Ortalama bekleme zamani=33.096000
Arac basina alinan yol=1006.600000

Rapor(0-11520)
Olay sayisi=390
Ortalama bekleme zamani=32.597436
Arac basina alinan yol=1042.000000

Rapor(0-12000)
Olay sayisi=405
Ortalama bekleme zamani=32.997531
Arac basina alinan yol=1085.600000

Rapor(0-12480)
Olay sayisi=419
Ortalama bekleme zamani=33.019093
Arac basina alinan yol=1126.200000

Rapor(0-12960)
Olay sayisi=437
Ortalama bekleme zamani=32.649886
Arac basina alinan yol=1167.600000

Rapor(0-13440)
Olay sayisi=454
Ortalama bekleme zamani=32.504405
Arac basina alinan yol=1202.400000

Rapor(0-13920)
Olay sayisi=471
Ortalama bekleme zamani=32.282378
Arac basina alinan yol=1242.000000

Rapor(0-14400)
Olay sayisi=489
Ortalama bekleme zamani=32.274029
Arac basina alinan yol=1289.200000

Rapor(0-14880)
Olay sayisi=502
Ortalama bekleme zamani=32.199203
Arac basina alinan yol=1325.000000

Rapor(0-15360)
Olay sayisi=512
Ortalama bekleme zamani=32.136719
Arac basina alinan yol=1348.800000

Rapor(0-15840)
Olay sayisi=531
Ortalama bekleme zamani=32.199623
Arac basina alinan yol=1415.200000

Rapor(0-16320)
Olay sayisi=540
Ortalama bekleme zamani=32.346296
Arac basina alinan yol=1438.800000

Rapor(0-16800)
Olay sayisi=551
Ortalama bekleme zamani=32.197822
Arac basina alinan yol=1463.400000

Rapor(0-17280)
Olay sayisi=573
Ortalama bekleme zamani=32.094241
Arac basina alinan yol=1521.800000

Rapor(0-17760)
Olay sayisi=595
Ortalama bekleme zamani=32.151261
Arac basina alinan yol=1582.600000

Rapor(0-18240)
Olay sayisi=607
Ortalama bekleme zamani=32.021417
Arac basina alinan yol=1607.800000

Rapor(0-18720)
Olay sayisi=628
Ortalama bekleme zamani=32.052548
Arac basina alinan yol=1672.200000

Rapor(0-19200)
Olay sayisi=638
Ortalama bekleme zamani=32.067398
Arac basina alinan yol=1687.600000

Rapor(0-19680)
 Olay sayisi=653
 Ortalama bekleme zamani=31.995406
 Arac basina alinan yol=1723.200000

Rapor(0-20160)
 Olay sayisi=675
 Ortalama bekleme zamani=32.562963
 Arac basina alinan yol=1808.800000

Rapor(0-20640)
 Olay sayisi=685
 Ortalama bekleme zamani=32.696350
 Arac basina alinan yol=1835.000000

Rapor(0-21120)
 Olay sayisi=699
 Ortalama bekleme zamani=32.608011
 Arac basina alinan yol=1874.000000

Rapor(0-21600)
 Olay sayisi=717
 Ortalama bekleme zamani=32.695955
 Arac basina alinan yol=1924.800000

Rapor(0-22080)
 Olay sayisi=736
 Ortalama bekleme zamani=32.709239
 Arac basina alinan yol=1974.800000

Rapor(0-22560)
 Olay sayisi=751
 Ortalama bekleme zamani=32.584554
 Arac basina alinan yol=2007.200000

Rapor(0-23040)
 Olay sayisi=767
 Ortalama bekleme zamani=32.608866
 Arac basina alinan yol=2051.200000

Rapor(0-23520)
 Olay sayisi=783
 Ortalama bekleme zamani=32.454662
 Arac basina alinan yol=2086.000000

Rapor(0-24000)
 Olay sayisi=798
 Ortalama bekleme zamani=32.329574
 Arac basina alinan yol=2114.400000

**Ek 12 Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu
(m=5)**

Rapor(0-480)

Olay sayisi=30

Ortalama bekleme zamanı=46.966667

Arac basina alinan yol=98.400000

Rapor(0-960)

Olay sayisi=53

Ortalama bekleme zamanı=72.433962

Arac basina alinan yol=237.600000

Rapor(0-1440)

Olay sayisi=74

Ortalama bekleme zamanı=61.162162

Arac basina alinan yol=289.600000

Rapor(0-1920)

Olay sayisi=90

Ortalama bekleme zamanı=56.111111

Arac basina alinan yol=332.600000

Rapor(0-2400)

Olay sayisi=115

Ortalama bekleme zamanı=53.139130

Arac basina alinan yol=422.600000

Rapor(0-2880)

Olay sayisi=137

Ortalama bekleme zamanı=50.948905

Arac basina alinan yol=498.600000

Rapor(0-3360)

Olay sayisi=160

Ortalama bekleme zamanı=48.825000

Arac basina alinan yol=566.400000

Rapor(0-3840)

Olay sayisi=176

Ortalama bekleme zamanı=47.181818

Arac basina alinan yol=620.800000

Rapor(0-4320)

Olay sayisi=205

Ortalama bekleme zamanı=46.478049

Arac basina alinan yol=724.400000

Rapor(0-4800)

Olay sayisi=225

Ortalama bekleme zamanı=45.320000

Arac basina alinan yol=778.600000

Rapor(0-5280)
 Olay sayisi=250
 Ortalama bekleme zamani=45.020000
 Arac basina alinan yol=860.000000

Rapor(0-5760)
 Olay sayisi=286
 Ortalama bekleme zamani=46.220280
 Arac basina alinan yol=977.000000

Rapor(0-6240)
 Olay sayisi=315
 Ortalama bekleme zamani=56.460317
 Arac basina alinan yol=1124.200000

Rapor(0-6720)
 Olay sayisi=337
 Ortalama bekleme zamani=59.091988
 Arac basina alinan yol=1231.200000

Rapor(0-7200)
 Olay sayisi=363
 Ortalama bekleme zamani=57.286501
 Arac basina alinan yol=1312.600000

Rapor(0-7680)
 Olay sayisi=380
 Ortalama bekleme zamani=56.734211
 Arac basina alinan yol=1379.600000

Rapor(0-8160)
 Olay sayisi=400
 Ortalama bekleme zamani=55.332500
 Arac basina alinan yol=1436.600000

Rapor(0-8640)
 Olay sayisi=425
 Ortalama bekleme zamani=54.258824
 Arac basina alinan yol=1518.600000

Rapor(0-9120)
 Olay sayisi=452
 Ortalama bekleme zamani=53.384956
 Arac basina alinan yol=1629.600000

Rapor(0-9600)
 Olay sayisi=470
 Ortalama bekleme zamani=53.165957
 Arac basina alinan yol=1691.400000

Rapor(0-10080)
Olay sayisi=500
Ortalama bekleme zamani=53.742000
Arac basina alinan yol=1818.000000

Rapor(0-10560)
Olay sayisi=525
Ortalama bekleme zamani=54.182857
Arac basina alinan yol=1927.800000

Rapor(0-11040)
Olay sayisi=542
Ortalama bekleme zamani=53.560886
Arac basina alinan yol=1979.600000

Rapor(0-11520)
Olay sayisi=579
Ortalama bekleme zamani=52.887737
Arac basina alinan yol=2107.800000

Rapor(0-12000)
Olay sayisi=609
Ortalama bekleme zamani=55.037767
Arac basina alinan yol=2241.600000

Rapor(0-12480)
Olay sayisi=634
Ortalama bekleme zamani=54.277603
Arac basina alinan yol=2324.000000

Rapor(0-12960)
Olay sayisi=658
Ortalama bekleme zamani=54.177812
Arac basina alinan yol=2393.000000

Rapor(0-13440)
Olay sayisi=692
Ortalama bekleme zamani=54.888728
Arac basina alinan yol=2523.800000

Rapor(0-13920)
Olay sayisi=723
Ortalama bekleme zamani=56.822960
Arac basina alinan yol=2655.200000

Rapor(0-14400)
Olay sayisi=748
Ortalama bekleme zamani=57.153743
Arac basina alinan yol=2750.000000

Rapor(0-14880)
Olay sayisi=766
Ortalama bekleme zamani=56.579634
Arac basina alinan yol=2820.200000

Rapor(0-15360)
Olay sayisi=789
Ortalama bekleme zamani=56.604563
Arac basina alinan yol=2912.600000

Rapor(0-15840)
Olay sayisi=806
Ortalama bekleme zamani=55.910670
Arac basina alinan yol=2950.400000

Rapor(0-16320)
Olay sayisi=828
Ortalama bekleme zamani=55.783816
Arac basina alinan yol=3057.800000

Rapor(0-16800)
Olay sayisi=856
Ortalama bekleme zamani=55.149533
Arac basina alinan yol=3177.400000

Rapor(0-17280)
Olay sayisi=883
Ortalama bekleme zamani=55.525481
Arac basina alinan yol=3282.600000

Rapor(0-17760)
Olay sayisi=909
Ortalama bekleme zamani=54.965897
Arac basina alinan yol=3380.800000

Rapor(0-18240)
Olay sayisi=933
Ortalama bekleme zamani=54.857449
Arac basina alinan yol=3481.600000

Rapor(0-18720)
Olay sayisi=957
Ortalama bekleme zamani=54.569488
Arac basina alinan yol=3554.800000

Rapor(0-19200)
Olay sayisi=977
Ortalama bekleme zamani=54.275333
Arac basina alinan yol=3622.400000

Rapor(0-19680)
 Olay sayisi=997
 Ortalama bekleme zamani=53.917753
 Arac basina alinan yol=3697.400000

Rapor(0-20160)
 Olay sayisi=1010
 Ortalama bekleme zamani=53.684158
 Arac basina alinan yol=3738.000000

Rapor(0-20640)
 Olay sayisi=1040
 Ortalama bekleme zamani=53.178846
 Arac basina alinan yol=3836.400000

Rapor(0-21120)
 Olay sayisi=1059
 Ortalama bekleme zamani=53.396601
 Arac basina alinan yol=3918.000000

Rapor(0-21600)
 Olay sayisi=1089
 Ortalama bekleme zamani=52.917355
 Arac basina alinan yol=4014.400000

Rapor(0-22080)
 Olay sayisi=1108
 Ortalama bekleme zamani=52.820397
 Arac basina alinan yol=4105.000000

Rapor(0-22560)
 Olay sayisi=1131
 Ortalama bekleme zamani=52.609195
 Arac basina alinan yol=4187.200000

Rapor(0-23040)
 Olay sayisi=1158
 Ortalama bekleme zamani=52.356649
 Arac basina alinan yol=4296.400000

Rapor(0-23520)
 Olay sayisi=1181
 Ortalama bekleme zamani=52.105843
 Arac basina alinan yol=4402.200000

Rapor(0-24000)
 Olay sayisi=1197
 Ortalama bekleme zamani=52.151211
 Arac basina alinan yol=4456.600000

**Ek 13 Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu
(m=5)**

Rapor(0-480)

Olay sayisi=29

Ortalama bekleme zamanı=31.275862

Arac basına alınan yol=124.200000

Rapor(0-960)

Olay sayisi=47

Ortalama bekleme zamanı=35.425532

Arac basına alınan yol=234.800000

Rapor(0-1440)

Olay sayisi=69

Ortalama bekleme zamanı=39.000000

Arac basına alınan yol=360.800000

Rapor(0-1920)

Olay sayisi=95

Ortalama bekleme zamanı=46.863158

Arac basına alınan yol=498.600000

Rapor(0-2400)

Olay sayisi=113

Ortalama bekleme zamanı=45.230088

Arac basına alınan yol=602.800000

Rapor(0-2880)

Olay sayisi=137

Ortalama bekleme zamanı=46.678832

Arac basına alınan yol=732.400000

Rapor(0-3360)

Olay sayisi=161

Ortalama bekleme zamanı=44.670807

Arac basına alınan yol=832.400000

Rapor(0-3840)

Olay sayisi=187

Ortalama bekleme zamanı=42.679144

Arac basına alınan yol=935.200000

Rapor(0-4320)

Olay sayisi=213

Ortalama bekleme zamanı=47.032864

Arac basına alınan yol=1076.400000

Rapor(0-4800)

Olay sayisi=236

Ortalama bekleme zamanı=47.055085

Arac basına alınan yol=1220.400000

Rapor(0-5280)
Olay sayisi=258
Ortalama bekleme zamani=45.899225
Arac basina alinan yol=1342.800000

Rapor(0-5760)
Olay sayisi=273
Ortalama bekleme zamani=45.663004
Arac basina alinan yol=1424.200000

Rapor(0-6240)
Olay sayisi=298
Ortalama bekleme zamani=47.493289
Arac basina alinan yol=1567.000000

Rapor(0-6720)
Olay sayisi=314
Ortalama bekleme zamani=46.872611
Arac basina alinan yol=1673.600000

Rapor(0-7200)
Olay sayisi=336
Ortalama bekleme zamani=46.619048
Arac basina alinan yol=1793.800000

Rapor(0-7680)
Olay sayisi=365
Ortalama bekleme zamani=45.679452
Arac basina alinan yol=1917.400000

Rapor(0-8160)
Olay sayisi=383
Ortalama bekleme zamani=45.872063
Arac basina alinan yol=2031.000000

Rapor(0-8640)
Olay sayisi=412
Ortalama bekleme zamani=47.060680
Arac basina alinan yol=2169.400000

Rapor(0-9120)
Olay sayisi=438
Ortalama bekleme zamani=49.289954
Arac basina alinan yol=2325.200000

Rapor(0-9600)
Olay sayisi=464
Ortalama bekleme zamani=49.278017
Arac basina alinan yol=2463.400000

Rapor(0-10080)
Olay sayisi=486
Ortalama bekleme zamani=49.345679
Arac basina alinan yol=2576.200000

Rapor(0-10560)
Olay sayisi=508
Ortalama bekleme zamani=49.330709
Arac basina alinan yol=2700.600000

Rapor(0-11040)
Olay sayisi=522
Ortalama bekleme zamani=49.005747
Arac basina alinan yol=2791.600000

Rapor(0-11520)
Olay sayisi=546
Ortalama bekleme zamani=48.811355
Arac basina alinan yol=2935.400000

Rapor(0-12000)
Olay sayisi=565
Ortalama bekleme zamani=48.449558
Arac basina alinan yol=3021.600000

Rapor(0-12480)
Olay sayisi=583
Ortalama bekleme zamani=48.144082
Arac basina alinan yol=3148.000000

Rapor(0-12960)
Olay sayisi=610
Ortalama bekleme zamani=47.878689
Arac basina alinan yol=3297.000000

Rapor(0-13440)
Olay sayisi=630
Ortalama bekleme zamani=47.314286
Arac basina alinan yol=3398.800000

Rapor(0-13920)
Olay sayisi=654
Ortalama bekleme zamani=47.212538
Arac basina alinan yol=3531.000000

Rapor(0-14400)
Olay sayisi=683
Ortalama bekleme zamani=47.972182
Arac basina alinan yol=3675.200000

Rapor(0-14880)
Olay sayisi=710
Ortalama bekleme zamani=47.445070
Arac basina alinan yol=3790.400000

Rapor(0-15360)
Olay sayisi=736
Ortalama bekleme zamani=47.288043
Arac basina alinan yol=3943.800000

Rapor(0-15840)
Olay sayisi=758
Ortalama bekleme zamani=47.284960
Arac basina alinan yol=4077.400000

Rapor(0-16320)
Olay sayisi=784
Ortalama bekleme zamani=47.404337
Arac basina alinan yol=4227.000000

Rapor(0-16800)
Olay sayisi=806
Ortalama bekleme zamani=47.153846
Arac basina alinan yol=4350.400000

Rapor(0-17280)
Olay sayisi=838
Ortalama bekleme zamani=47.312649
Arac basina alinan yol=4504.800000

Rapor(0-17760)
Olay sayisi=861
Ortalama bekleme zamani=47.725900
Arac basina alinan yol=4622.400000

Rapor(0-18240)
Olay sayisi=882
Ortalama bekleme zamani=47.395692
Arac basina alinan yol=4718.200000

Rapor(0-18720)
Olay sayisi=901
Ortalama bekleme zamani=47.143174
Arac basina alinan yol=4810.800000

Rapor(0-19200)
Olay sayisi=926
Ortalama bekleme zamani=47.173866
Arac basina alinan yol=4958.400000

Rapor(0-19680)
Olay sayisi=951
Ortalama bekleme zamani=46.909569
Arac basina alinan yol=5077.600000

Rapor(0-20160)
Olay sayisi=976
Ortalama bekleme zamani=47.258197
Arac basina alinan yol=5230.400000

Rapor(0-20640)
Olay sayisi=1014
Ortalama bekleme zamani=47.409270
Arac basina alinan yol=5386.600000

Rapor(0-21120)
Olay sayisi=1045
Ortalama bekleme zamani=51.586603
Arac basina alinan yol=5522.600000

Rapor(0-21600)
Olay sayisi=1068
Ortalama bekleme zamani=55.432584
Arac basina alinan yol=5667.600000

Rapor(0-22080)
Olay sayisi=1098
Ortalama bekleme zamani=55.693989
Arac basina alinan yol=5805.800000

Rapor(0-22560)
Olay sayisi=1127
Ortalama bekleme zamani=55.802130
Arac basina alinan yol=5964.400000

Rapor(0-23040)
Olay sayisi=1148
Ortalama bekleme zamani=56.078397
Arac basina alinan yol=6111.000000

Rapor(0-23520)
Olay sayisi=1168
Ortalama bekleme zamani=55.785103
Arac basina alinan yol=6219.400000

Rapor(0-24000)
Olay sayisi=1192
Ortalama bekleme zamani=55.270134
Arac basina alinan yol=6342.400000

**Ek 14 Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu
(m=5)**

Rapor(0-480)

Olay sayisi=23

Ortalama bekleme zamanı=45.347826

Arac basına alınan yol=83.200000

Rapor(0-960)

Olay sayisi=45

Ortalama bekleme zamanı=38.133333

Arac basına alınan yol=136.200000

Rapor(0-1440)

Olay sayisi=68

Ortalama bekleme zamanı=35.352941

Arac basına alınan yol=202.200000

Rapor(0-1920)

Olay sayisi=94

Ortalama bekleme zamanı=37.106383

Arac basına alınan yol=285.600000

Rapor(0-2400)

Olay sayisi=119

Ortalama bekleme zamanı=38.100840

Arac basına alınan yol=347.400000

Rapor(0-2880)

Olay sayisi=141

Ortalama bekleme zamanı=35.872340

Arac basına alınan yol=412.200000

Rapor(0-3360)

Olay sayisi=170

Ortalama bekleme zamanı=46.500000

Arac basına alınan yol=514.800000

Rapor(0-3840)

Olay sayisi=192

Ortalama bekleme zamanı=46.125000

Arac basına alınan yol=578.200000

Rapor(0-4320)

Olay sayisi=208

Ortalama bekleme zamanı=44.725962

Arac basına alınan yol=617.400000

Rapor(0-4800)

Olay sayisi=239

Ortalama bekleme zamanı=45.589958

Arac basına alınan yol=723.200000

Rapor(0-5280)
Olay sayisi=258
Ortalama bekleme zamani=49.926357
Arac basina alinan yol=798.800000

Rapor(0-5760)
Olay sayisi=284
Ortalama bekleme zamani=49.151408
Arac basina alinan yol=886.200000

Rapor(0-6240)
Olay sayisi=311
Ortalama bekleme zamani=51.257235
Arac basina alinan yol=971.600000

Rapor(0-6720)
Olay sayisi=338
Ortalama bekleme zamani=52.467456
Arac basina alinan yol=1055.800000

Rapor(0-7200)
Olay sayisi=365
Ortalama bekleme zamani=51.865753
Arac basina alinan yol=1139.600000

Rapor(0-7680)
Olay sayisi=392
Ortalama bekleme zamani=50.010204
Arac basina alinan yol=1241.000000

Rapor(0-8160)
Olay sayisi=416
Ortalama bekleme zamani=51.786058
Arac basina alinan yol=1333.800000

Rapor(0-8640)
Olay sayisi=440
Ortalama bekleme zamani=51.479545
Arac basina alinan yol=1419.600000

Rapor(0-9120)
Olay sayisi=460
Ortalama bekleme zamani=51.817391
Arac basina alinan yol=1501.400000

Rapor(0-9600)
Olay sayisi=489
Ortalama bekleme zamani=50.963190
Arac basina alinan yol=1594.000000

Rapor(0-10080)
Olay sayisi=507
Ortalama bekleme zamani=50.258383
Arac basina alinan yol=1629.600000

Rapor(0-10560)
Olay sayisi=526
Ortalama bekleme zamani=49.509506
Arac basina alinan yol=1675.800000

Rapor(0-11040)
Olay sayisi=552
Ortalama bekleme zamani=48.469203
Arac basina alinan yol=1744.200000

Rapor(0-11520)
Olay sayisi=575
Ortalama bekleme zamani=48.572174
Arac basina alinan yol=1828.200000

Rapor(0-12000)
Olay sayisi=613
Ortalama bekleme zamani=50.344209
Arac basina alinan yol=1938.000000

Rapor(0-12480)
Olay sayisi=638
Ortalama bekleme zamani=50.341693
Arac basina alinan yol=2025.400000

Rapor(0-12960)
Olay sayisi=661
Ortalama bekleme zamani=50.057489
Arac basina alinan yol=2090.200000

Rapor(0-13440)
Olay sayisi=681
Ortalama bekleme zamani=49.443465
Arac basina alinan yol=2162.800000

Rapor(0-13920)
Olay sayisi=700
Ortalama bekleme zamani=49.215714
Arac basina alinan yol=2214.000000

Rapor(0-14400)
Olay sayisi=732
Ortalama bekleme zamani=48.814208
Arac basina alinan yol=2290.600000

Rapor(0-14880)
 Olay sayisi=755
 Ortalama bekleme zamani=48.512583
 Arac basina alinan yol=2362.200000

Rapor(0-15360)
 Olay sayisi=774
 Ortalama bekleme zamani=48.025840
 Arac basina alinan yol=2413.600000

Rapor(0-15840)
 Olay sayisi=799
 Ortalama bekleme zamani=47.619524
 Arac basina alinan yol=2487.400000

Rapor(0-16320)
 Olay sayisi=827
 Ortalama bekleme zamani=47.977025
 Arac basina alinan yol=2571.200000

Rapor(0-16800)
 Olay sayisi=851
 Ortalama bekleme zamani=48.321974
 Arac basina alinan yol=2662.600000

Rapor(0-17280)
 Olay sayisi=873
 Ortalama bekleme zamani=47.957617
 Arac basina alinan yol=2713.800000

Rapor(0-17760)
 Olay sayisi=900
 Ortalama bekleme zamani=47.226667
 Arac basina alinan yol=2786.600000

Rapor(0-18240)
 Olay sayisi=921
 Ortalama bekleme zamani=48.059718
 Arac basina alinan yol=2875.400000

Rapor(0-18720)
 Olay sayisi=947
 Ortalama bekleme zamani=47.709609
 Arac basina alinan yol=2932.800000

Rapor(0-19200)
 Olay sayisi=965
 Ortalama bekleme zamani=47.538860
 Arac basina alinan yol=2985.600000

Rapor(0-19680)
 Olay sayisi=982
 Ortalama bekleme zamani=47.308554
 Arac basina alinan yol=3029.800000

Rapor(0-20160)
 Olay sayisi=1000
 Ortalama bekleme zamani=46.946000
 Arac basina alinan yol=3075.800000

Rapor(0-20640)
 Olay sayisi=1027
 Ortalama bekleme zamani=46.627069
 Arac basina alinan yol=3162.200000

Rapor(0-21120)
 Olay sayisi=1056
 Ortalama bekleme zamani=46.920455
 Arac basina alinan yol=3259.600000

Rapor(0-21600)
 Olay sayisi=1079
 Ortalama bekleme zamani=46.825765
 Arac basina alinan yol=3342.200000

Rapor(0-22080)
 Olay sayisi=1102
 Ortalama bekleme zamani=46.411978
 Arac basina alinan yol=3409.800000

Rapor(0-22560)
 Olay sayisi=1114
 Ortalama bekleme zamani=46.386894
 Arac basina alinan yol=3441.200000

Rapor(0-23040)
 Olay sayisi=1144
 Ortalama bekleme zamani=46.170455
 Arac basina alinan yol=3526.800000

Rapor(0-23520)
 Olay sayisi=1164
 Ortalama bekleme zamani=46.062715
 Arac basina alinan yol=3585.400000

Rapor(0-24000)
 Olay sayisi=1195
 Ortalama bekleme zamani=45.820921
 Arac basina alinan yol=3666.800000

Ek 15 Çok Yoğun Trafik Altında NN Politikası İçin Koşturulan Simülasyonun Raporu (m=5)

Rapor(0-480)

Olay sayisi=32

Ortalama bekleme zamani=80.750000

Arac basina alinan yol=152.400000

Rapor(0-960)

Olay sayisi=61

Ortalama bekleme zamani=77.196721

Arac basina alinan yol=279.000000

Rapor(0-1440)

Olay sayisi=84

Ortalama bekleme zamani=70.428571

Arac basina alinan yol=394.200000

Rapor(0-1920)

Olay sayisi=113

Ortalama bekleme zamani=61.123894

Arac basina alinan yol=489.800000

Rapor(0-2400)

Olay sayisi=151

Ortalama bekleme zamani=80.192053

Arac basina alinan yol=632.000000

Rapor(0-2880)

Olay sayisi=181

Ortalama bekleme zamani=103.784530

Arac basina alinan yol=777.800000

Rapor(0-3360)

Olay sayisi=231

Ortalama bekleme zamani=114.961039

Arac basina alinan yol=921.600000

Rapor(0-3840)

Olay sayisi=277

Ortalama bekleme zamani=139.158845

Arac basina alinan yol=1070.000000

Rapor(0-4320)

Olay sayisi=307

Ortalama bekleme zamani=179.680782

Arac basina alinan yol=1204.600000

Rapor(0-4800)

Olay sayisi=338

Ortalama bekleme zamani=225.414201

Arac basina alinan yol=1350.200000

Rapor(0-5280)
Olay sayisi=363
Ortalama bekleme zamani=260.236915
Arac basina alinan yol=1504.200000

Rapor(0-5760)
Olay sayisi=390
Ortalama bekleme zamani=295.028205
Arac basina alinan yol=1645.600000

Rapor(0-6240)
Olay sayisi=428
Ortalama bekleme zamani=301.570093
Arac basina alinan yol=1773.200000

Rapor(0-6720)
Olay sayisi=455
Ortalama bekleme zamani=313.123077
Arac basina alinan yol=1918.800000

Rapor(0-7200)
Olay sayisi=485
Ortalama bekleme zamani=322.364948
Arac basina alinan yol=2056.600000

Rapor(0-7680)
Olay sayisi=516
Ortalama bekleme zamani=330.397287
Arac basina alinan yol=2184.600000

Rapor(0-8160)
Olay sayisi=548
Ortalama bekleme zamani=331.830292
Arac basina alinan yol=2352.800000

Rapor(0-8640)
Olay sayisi=574
Ortalama bekleme zamani=335.102787
Arac basina alinan yol=2477.600000

Rapor(0-9120)
Olay sayisi=594
Ortalama bekleme zamani=335.286195
Arac basina alinan yol=2628.200000

Rapor(0-9600)
Olay sayisi=622
Ortalama bekleme zamani=323.495177
Arac basina alinan yol=2759.400000

Rapor(0-10080)
 Olay sayisi=657
 Ortalama bekleme zamani=311.289193
 Arac basina alinan yol=2899.800000

Rapor(0-10560)
 Olay sayisi=685
 Ortalama bekleme zamani=306.251095
 Arac basina alinan yol=3059.200000

Rapor(0-11040)
 Olay sayisi=716
 Ortalama bekleme zamani=297.255587
 Arac basina alinan yol=3183.000000

Rapor(0-11520)
 Olay sayisi=747
 Ortalama bekleme zamani=287.210174
 Arac basina alinan yol=3308.000000

Rapor(0-12000)
 Olay sayisi=777
 Ortalama bekleme zamani=279.109395
 Arac basina alinan yol=3455.800000

Rapor(0-12480)
 Olay sayisi=802
 Ortalama bekleme zamani=272.072319
 Arac basina alinan yol=3558.200000

Rapor(0-12960)
 Olay sayisi=833
 Ortalama bekleme zamani=263.721489
 Arac basina alinan yol=3691.600000

Rapor(0-13440)
 Olay sayisi=874
 Ortalama bekleme zamani=255.779176
 Arac basina alinan yol=3833.400000

Rapor(0-13920)
 Olay sayisi=905
 Ortalama bekleme zamani=253.209945
 Arac basina alinan yol=3973.200000

Rapor(0-14400)
 Olay sayisi=942
 Ortalama bekleme zamani=249.360934
 Arac basina alinan yol=4120.400000

Rapor(0-14880)
Olay sayisi=973
Ortalama bekleme zamani=248.436793
Arac basina alinan yol=4253.800000

Rapor(0-15360)
Olay sayisi=995
Ortalama bekleme zamani=248.423116
Arac basina alinan yol=4395.400000

Rapor(0-15840)
Olay sayisi=1028
Ortalama bekleme zamani=243.372568
Arac basina alinan yol=4551.600000

Rapor(0-16320)
Olay sayisi=1063
Ortalama bekleme zamani=239.643462
Arac basina alinan yol=4708.400000

Rapor(0-16800)
Olay sayisi=1093
Ortalama bekleme zamani=236.566331
Arac basina alinan yol=4851.600000

Rapor(0-17280)
Olay sayisi=1118
Ortalama bekleme zamani=233.417710
Arac basina alinan yol=4979.600000

Rapor(0-17760)
Olay sayisi=1147
Ortalama bekleme zamani=230.154316
Arac basina alinan yol=5108.000000

Rapor(0-18240)
Olay sayisi=1186
Ortalama bekleme zamani=224.261383
Arac basina alinan yol=5225.200000

Rapor(0-18720)
Olay sayisi=1224
Ortalama bekleme zamani=221.526961
Arac basina alinan yol=5384.800000

Rapor(0-19200)
Olay sayisi=1253
Ortalama bekleme zamani=222.378292
Arac basina alinan yol=5510.200000

Rapor(0-19680)
Olay sayisi=1284
Ortalama bekleme zamani=222.546729
Arac basina alinan yol=5658.800000

Rapor(0-20160)
Olay sayisi=1316
Ortalama bekleme zamani=222.489362
Arac basina alinan yol=5804.800000

Rapor(0-20640)
Olay sayisi=1348
Ortalama bekleme zamani=222.162463
Arac basina alinan yol=5953.200000

Rapor(0-21120)
Olay sayisi=1383
Ortalama bekleme zamani=221.359364
Arac basina alinan yol=6113.600000

Rapor(0-21600)
Olay sayisi=1405
Ortalama bekleme zamani=224.911032
Arac basina alinan yol=6263.600000

Rapor(0-22080)
Olay sayisi=1437
Ortalama bekleme zamani=222.900487
Arac basina alinan yol=6402.400000

Rapor(0-22560)
Olay sayisi=1474
Ortalama bekleme zamani=219.531208
Arac basina alinan yol=6539.000000

Rapor(0-23040)
Olay sayisi=1513
Ortalama bekleme zamani=217.320555
Arac basina alinan yol=6688.800000

Rapor(0-23520)
Olay sayisi=1545
Ortalama bekleme zamani=217.015534
Arac basina alinan yol=6844.800000

Rapor(0-24000)
Olay sayisi=1572
Ortalama bekleme zamani=217.716285
Arac basina alinan yol=6992.400000

Ek 16 Çok Yoğun Trafik Altında SQM Politikası İçin Koşturulan Simülasyonun Raporu (m=5)

Rapor(0-480)

Olay sayisi=34

Ortalama bekleme zamanı=47.647059

Arac basına alınan yol=148.600000

Rapor(0-960)

Olay sayisi=72

Ortalama bekleme zamanı=65.555556

Arac basına alınan yol=297.400000

Rapor(0-1440)

Olay sayisi=106

Ortalama bekleme zamanı=94.952830

Arac basına alınan yol=443.800000

Rapor(0-1920)

Olay sayisi=123

Ortalama bekleme zamanı=120.040650

Arac basına alınan yol=579.800000

Rapor(0-2400)

Olay sayisi=155

Ortalama bekleme zamanı=104.277419

Arac basına alınan yol=733.400000

Rapor(0-2880)

Olay sayisi=188

Ortalama bekleme zamanı=95.079787

Arac basına alınan yol=877.400000

Rapor(0-3360)

Olay sayisi=227

Ortalama bekleme zamanı=95.691630

Arac basına alınan yol=1001.200000

Rapor(0-3840)

Olay sayisi=244

Ortalama bekleme zamanı=114.782787

Arac basına alınan yol=1165.600000

Rapor(0-4320)

Olay sayisi=272

Ortalama bekleme zamanı=111.819853

Arac basına alınan yol=1314.200000

Rapor(0-4800)

Olay sayisi=308

Ortalama bekleme zamanı=107.915584

Arac basına alınan yol=1473.800000

Rapor(0-5280)
 Olay sayisi=337
 Ortalama bekleme zamani=104.513353
 Arac basina alinan yol=1603.000000

Rapor(0-5760)
 Olay sayisi=366
 Ortalama bekleme zamani=101.158470
 Arac basina alinan yol=1748.800000

Rapor(0-6240)
 Olay sayisi=396
 Ortalama bekleme zamani=103.782828
 Arac basina alinan yol=1888.600000

Rapor(0-6720)
 Olay sayisi=433
 Ortalama bekleme zamani=99.822171
 Arac basina alinan yol=2030.800000

Rapor(0-7200)
 Olay sayisi=470
 Ortalama bekleme zamani=103.431915
 Arac basina alinan yol=2174.400000

Rapor(0-7680)
 Olay sayisi=514
 Ortalama bekleme zamani=108.832685
 Arac basina alinan yol=2325.600000

Rapor(0-8160)
 Olay sayisi=541
 Ortalama bekleme zamani=124.663586
 Arac basina alinan yol=2474.200000

Rapor(0-8640)
 Olay sayisi=570
 Ortalama bekleme zamani=141.201754
 Arac basina alinan yol=2606.600000

Rapor(0-9120)
 Olay sayisi=603
 Ortalama bekleme zamani=148.316750
 Arac basina alinan yol=2752.800000

Rapor(0-9600)
 Olay sayisi=635
 Ortalama bekleme zamani=157.083465
 Arac basina alinan yol=2884.000000

Rapor(0-10080)
Olay sayisi=666
Ortalama bekleme zamani=161.993994
Arac basina alinan yol=3013.800000

Rapor(0-10560)
Olay sayisi=691
Ortalama bekleme zamani=165.057887
Arac basina alinan yol=3144.200000

Rapor(0-11040)
Olay sayisi=718
Ortalama bekleme zamani=165.527855
Arac basina alinan yol=3282.600000

Rapor(0-11520)
Olay sayisi=753
Ortalama bekleme zamani=161.665339
Arac basina alinan yol=3430.400000

Rapor(0-12000)
Olay sayisi=784
Ortalama bekleme zamani=161.071429
Arac basina alinan yol=3575.400000

Rapor(0-12480)
Olay sayisi=812
Ortalama bekleme zamani=157.673645
Arac basina alinan yol=3717.600000

Rapor(0-12960)
Olay sayisi=851
Ortalama bekleme zamani=154.361927
Arac basina alinan yol=3866.200000

Rapor(0-13440)
Olay sayisi=895
Ortalama bekleme zamani=154.892737
Arac basina alinan yol=4004.200000

Rapor(0-13920)
Olay sayisi=919
Ortalama bekleme zamani=160.051143
Arac basina alinan yol=4161.800000

Rapor(0-14400)
Olay sayisi=944
Ortalama bekleme zamani=165.574153
Arac basina alinan yol=4314.600000

Rapor(0-14880)
Olay sayisi=983
Ortalama bekleme zamani=166.225839
Arac basina alinan yol=4467.600000

Rapor(0-15360)
Olay sayisi=1012
Ortalama bekleme zamani=172.118577
Arac basina alinan yol=4603.600000

Rapor(0-15840)
Olay sayisi=1044
Ortalama bekleme zamani=175.702107
Arac basina alinan yol=4752.000000

Rapor(0-16320)
Olay sayisi=1069
Ortalama bekleme zamani=180.509822
Arac basina alinan yol=4867.800000

Rapor(0-16800)
Olay sayisi=1097
Ortalama bekleme zamani=178.260711
Arac basina alinan yol=5016.600000

Rapor(0-17280)
Olay sayisi=1131
Ortalama bekleme zamani=174.580902
Arac basina alinan yol=5160.600000

Rapor(0-17760)
Olay sayisi=1164
Ortalama bekleme zamani=171.671821
Arac basina alinan yol=5290.200000

Rapor(0-18240)
Olay sayisi=1181
Ortalama bekleme zamani=171.002540
Arac basina alinan yol=5406.800000

Rapor(0-18720)
Olay sayisi=1210
Ortalama bekleme zamani=167.648760
Arac basina alinan yol=5525.600000

Rapor(0-19200)
Olay sayisi=1243
Ortalama bekleme zamani=164.843121
Arac basina alinan yol=5680.600000

Rapor(0-19680)
Olay sayisi=1277
Ortalama bekleme zamani=162.141738
Arac basina alinan yol=5826.600000

Rapor(0-20160)
Olay sayisi=1302
Ortalama bekleme zamani=161.426267
Arac basina alinan yol=5965.800000

Rapor(0-20640)
Olay sayisi=1331
Ortalama bekleme zamani=159.048835
Arac basina alinan yol=6099.400000

Rapor(0-21120)
Olay sayisi=1358
Ortalama bekleme zamani=157.432990
Arac basina alinan yol=6248.400000

Rapor(0-21600)
Olay sayisi=1387
Ortalama bekleme zamani=155.312185
Arac basina alinan yol=6397.600000

Rapor(0-22080)
Olay sayisi=1416
Ortalama bekleme zamani=153.552260
Arac basina alinan yol=6531.200000

Rapor(0-22560)
Olay sayisi=1441
Ortalama bekleme zamani=152.120056
Arac basina alinan yol=6651.200000

Rapor(0-23040)
Olay sayisi=1471
Ortalama bekleme zamani=150.445955
Arac basina alinan yol=6812.000000

Rapor(0-23520)
Olay sayisi=1505
Ortalama bekleme zamani=149.848505
Arac basina alinan yol=6951.400000

Rapor(0-24000)
Olay sayisi=1539
Ortalama bekleme zamani=148.870045
Arac basina alinan yol=7091.400000

Ek 17 Çok Yoğun Trafik Altında SAT Politikası İçin Koşturulan Simülasyonun Raporu ($m=5$)

Rapor(0-480)

Olay sayisi=30

Ortalama bekleme zamanı=34.433333

Arac basına alınan yol=92.600000

Rapor(0-960)

Olay sayisi=64

Ortalama bekleme zamanı=53.687500

Arac basına alınan yol=214.000000

Rapor(0-1440)

Olay sayisi=99

Ortalama bekleme zamanı=72.646465

Arac basına alınan yol=336.400000

Rapor(0-1920)

Olay sayisi=131

Ortalama bekleme zamanı=76.503817

Arac basına alınan yol=431.200000

Rapor(0-2400)

Olay sayisi=170

Ortalama bekleme zamanı=75.941176

Arac basına alınan yol=550.400000

Rapor(0-2880)

Olay sayisi=196

Ortalama bekleme zamanı=74.392857

Arac basına alınan yol=612.600000

Rapor(0-3360)

Olay sayisi=217

Ortalama bekleme zamanı=70.949309

Arac basına alınan yol=686.200000

Rapor(0-3840)

Olay sayisi=254

Ortalama bekleme zamanı=65.771654

Arac basına alınan yol=821.800000

Rapor(0-4320)

Olay sayisi=281

Ortalama bekleme zamanı=68.124555

Arac basına alınan yol=915.200000

Rapor(0-4800)

Olay sayisi=314

Ortalama bekleme zamanı=67.388535

Arac basına alınan yol=1061.000000

Rapor(0-5280)
Olay sayisi=343
Ortalama bekleme zamani=73.087464
Arac basina alinan yol=1156.800000

Rapor(0-5760)
Olay sayisi=372
Ortalama bekleme zamani=72.419355
Arac basina alinan yol=1238.000000

Rapor(0-6240)
Olay sayisi=407
Ortalama bekleme zamani=68.769042
Arac basina alinan yol=1357.800000

Rapor(0-6720)
Olay sayisi=437
Ortalama bekleme zamani=72.457666
Arac basina alinan yol=1491.800000

Rapor(0-7200)
Olay sayisi=470
Ortalama bekleme zamani=77.202128
Arac basina alinan yol=1638.000000

Rapor(0-7680)
Olay sayisi=503
Ortalama bekleme zamani=82.477137
Arac basina alinan yol=1766.000000

Rapor(0-8160)
Olay sayisi=530
Ortalama bekleme zamani=91.339623
Arac basina alinan yol=1869.000000

Rapor(0-8640)
Olay sayisi=574
Ortalama bekleme zamani=89.047038
Arac basina alinan yol=2043.000000

Rapor(0-9120)
Olay sayisi=595
Ortalama bekleme zamani=93.914286
Arac basina alinan yol=2122.000000

Rapor(0-9600)
Olay sayisi=638
Ortalama bekleme zamani=93.949843
Arac basina alinan yol=2247.600000

Rapor(0-10080)
Olay sayisi=675
Ortalama bekleme zamani=94.845926
Arac basina alinan yol=2397.800000

Rapor(0-10560)
Olay sayisi=712
Ortalama bekleme zamani=101.544944
Arac basina alinan yol=2520.200000

Rapor(0-11040)
Olay sayisi=745
Ortalama bekleme zamani=106.510067
Arac basina alinan yol=2648.400000

Rapor(0-11520)
Olay sayisi=770
Ortalama bekleme zamani=113.074026
Arac basina alinan yol=2728.400000

Rapor(0-12000)
Olay sayisi=792
Ortalama bekleme zamani=113.178030
Arac basina alinan yol=2806.800000

Rapor(0-12480)
Olay sayisi=813
Ortalama bekleme zamani=111.093481
Arac basina alinan yol=2860.600000

Rapor(0-12960)
Olay sayisi=840
Ortalama bekleme zamani=108.744048
Arac basina alinan yol=2945.800000

Rapor(0-13440)
Olay sayisi=880
Ortalama bekleme zamani=108.078409
Arac basina alinan yol=3081.400000

Rapor(0-13920)
Olay sayisi=910
Ortalama bekleme zamani=108.771429
Arac basina alinan yol=3190.600000

Rapor(0-14400)
Olay sayisi=947
Ortalama bekleme zamani=109.931362
Arac basina alinan yol=3323.800000

Rapor(0-14880)
Olay sayisi=963
Ortalama bekleme zamani=110.740395
Arac basina alinan yol=3361.200000

Rapor(0-15360)
Olay sayisi=992
Ortalama bekleme zamani=108.450605
Arac basina alinan yol=3429.000000

Rapor(0-15840)
Olay sayisi=1039
Ortalama bekleme zamani=107.202117
Arac basina alinan yol=3581.800000

Rapor(0-16320)
Olay sayisi=1074
Ortalama bekleme zamani=108.400372
Arac basina alinan yol=3686.600000

Rapor(0-16800)
Olay sayisi=1107
Ortalama bekleme zamani=109.705510
Arac basina alinan yol=3798.400000

Rapor(0-17280)
Olay sayisi=1131
Ortalama bekleme zamani=110.163572
Arac basina alinan yol=3883.800000

Rapor(0-17760)
Olay sayisi=1154
Ortalama bekleme zamani=109.297227
Arac basina alinan yol=3973.400000

Rapor(0-18240)
Olay sayisi=1190
Ortalama bekleme zamani=107.039496
Arac basina alinan yol=4083.600000

Rapor(0-18720)
Olay sayisi=1224
Ortalama bekleme zamani=105.959150
Arac basina alinan yol=4192.400000

Rapor(0-19200)
Olay sayisi=1250
Ortalama bekleme zamani=105.827200
Arac basina alinan yol=4286.000000

Rapor(0-19680)
Olay sayisi=1277
Ortalama bekleme zamani=105.129209
Arac basina alinan yol=4368.000000

Rapor(0-20160)
Olay sayisi=1309
Ortalama bekleme zamani=103.330787
Arac basina alinan yol=4450.000000

Rapor(0-20640)
Olay sayisi=1333
Ortalama bekleme zamani=102.640660
Arac basina alinan yol=4525.200000

Rapor(0-21120)
Olay sayisi=1364
Ortalama bekleme zamani=101.145894
Arac basina alinan yol=4610.800000

Rapor(0-21600)
Olay sayisi=1399
Ortalama bekleme zamani=100.238027
Arac basina alinan yol=4735.600000

Rapor(0-22080)
Olay sayisi=1418
Ortalama bekleme zamani=100.672779
Arac basina alinan yol=4788.000000

Rapor(0-22560)
Olay sayisi=1451
Ortalama bekleme zamani=99.061337
Arac basina alinan yol=4900.000000

Rapor(0-23040)
Olay sayisi=1477
Ortalama bekleme zamani=99.424509
Arac basina alinan yol=4981.600000

Rapor(0-23520)
Olay sayisi=1510
Ortalama bekleme zamani=98.041060
Arac basina alinan yol=5069.200000

Rapor(0-24000)
Olay sayisi=1546
Ortalama bekleme zamani=98.135188
Arac basina alinan yol=5192.000000

**Ek 18 Yoğun Trafik Altında SAT Politikası Doğrultusunda 3 Araçlık Filo İçin
Koşturulan Simülasyonun Raporu**

Rapor(0-480)

Olay sayisi=22

Ortalama bekleme zamani=47.636364

Arac basina alinan yol=126.333333

Rapor(0-960)

Olay sayisi=46

Ortalama bekleme zamani=66.369565

Arac basina alinan yol=290.666667

Rapor(0-1440)

Olay sayisi=74

Ortalama bekleme zamani=100.554054

Arac basina alinan yol=464.000000

Rapor(0-1920)

Olay sayisi=92

Ortalama bekleme zamani=152.793478

Arac basina alinan yol=584.333333

Rapor(0-2400)

Olay sayisi=110

Ortalama bekleme zamani=217.118182

Arac basina alinan yol=700.666667

Rapor(0-2880)

Olay sayisi=139

Ortalama bekleme zamani=220.546763

Arac basina alinan yol=852.333333

Rapor(0-3360)

Olay sayisi=158

Ortalama bekleme zamani=253.734177

Arac basina alinan yol=946.000000

Rapor(0-3840)

Olay sayisi=184

Ortalama bekleme zamani=254.114130

Arac basina alinan yol=1098.333333

Rapor(0-4320)

Olay sayisi=210

Ortalama bekleme zamani=265.166667

Arac basina alinan yol=1248.333333

Rapor(0-4800)

Olay sayisi=237

Ortalama bekleme zamani=277.447257

Arac basina alinan yol=1405.000000

Rapor(0-5280)
Olay sayisi=258
Ortalama bekleme zamani=307.182171
Arac basina alinan yol=1551.666667

Rapor(0-5760)
Olay sayisi=283
Ortalama bekleme zamani=331.487633
Arac basina alinan yol=1725.666667

Rapor(0-6240)
Olay sayisi=309
Ortalama bekleme zamani=363.055016
Arac basina alinan yol=1909.333333

Rapor(0-6720)
Olay sayisi=326
Ortalama bekleme zamani=396.177914
Arac basina alinan yol=2011.666667

Rapor(0-7200)
Olay sayisi=346
Ortalama bekleme zamani=421.271676
Arac basina alinan yol=2144.333333

Rapor(0-7680)
Olay sayisi=372
Ortalama bekleme zamani=448.752688
Arac basina alinan yol=2346.000000

Rapor(0-8160)
Olay sayisi=394
Ortalama bekleme zamani=480.723350
Arac basina alinan yol=2513.333333

Rapor(0-8640)
Olay sayisi=423
Ortalama bekleme zamani=501.174941
Arac basina alinan yol=2707.666667

Rapor(0-9120)
Olay sayisi=451
Ortalama bekleme zamani=519.148559
Arac basina alinan yol=2874.666667

Rapor(0-9600)
Olay sayisi=478
Ortalama bekleme zamani=549.382845
Arac basina alinan yol=3045.333333

Rapor(0-10080)
Olay sayisi=498
Ortalama bekleme zamani=581.124498
Arac basina alinan yol=3166.000000

Rapor(0-10560)
Olay sayisi=525
Ortalama bekleme zamani=609.089524
Arac basina alinan yol=3334.333333

Rapor(0-11040)
Olay sayisi=554
Ortalama bekleme zamani=627.167870
Arac basina alinan yol=3548.000000

Rapor(0-11520)
Olay sayisi=577
Ortalama bekleme zamani=666.162912
Arac basina alinan yol=3700.333333

Rapor(0-12000)
Olay sayisi=603
Ortalama bekleme zamani=689.439469
Arac basina alinan yol=3877.333333

Rapor(0-12480)
Olay sayisi=621
Ortalama bekleme zamani=732.283414
Arac basina alinan yol=3987.666667

Rapor(0-12960)
Olay sayisi=644
Ortalama bekleme zamani=761.830745
Arac basina alinan yol=4181.666667

Rapor(0-13440)
Olay sayisi=666
Ortalama bekleme zamani=800.345345
Arac basina alinan yol=4316.333333

Rapor(0-13920)
Olay sayisi=685
Ortalama bekleme zamani=839.951825
Arac basina alinan yol=4422.333333

Rapor(0-14400)
Olay sayisi=708
Ortalama bekleme zamani=885.668079
Arac basina alinan yol=4564.666667

Rapor(0-14880)
Olay sayisi=730
Ortalama bekleme zamani=915.290411
Arac basina alinan yol=4694.000000

Rapor(0-15360)
Olay sayisi=751
Ortalama bekleme zamani=956.151798
Arac basina alinan yol=4801.333333

Rapor(0-15840)
Olay sayisi=782
Ortalama bekleme zamani=987.970588
Arac basina alinan yol=4981.666667

Rapor(0-16320)
Olay sayisi=803
Ortalama bekleme zamani=1032.990037
Arac basina alinan yol=5130.666667

Rapor(0-16800)
Olay sayisi=831
Ortalama bekleme zamani=1071.241877
Arac basina alinan yol=5304.666667

Rapor(0-17280)
Olay sayisi=856
Ortalama bekleme zamani=1120.146028
Arac basina alinan yol=5454.666667

Rapor(0-17760)
Olay sayisi=876
Ortalama bekleme zamani=1160.452055
Arac basina alinan yol=5586.000000

Rapor(0-18240)
Olay sayisi=900
Ortalama bekleme zamani=1204.761111
Arac basina alinan yol=5737.666667

Rapor(0-18720)
Olay sayisi=923
Ortalama bekleme zamani=1250.793066
Arac basina alinan yol=5881.666667

Rapor(0-19200)
Olay sayisi=945
Ortalama bekleme zamani=1285.552381
Arac basina alinan yol=6021.000000

Rapor(0-19680)
 Olay sayisi=964
 Ortalama bekleme zamani=1324.627593
 Arac basina alinan yol=6152.000000

Rapor(0-20160)
 Olay sayisi=996
 Ortalama bekleme zamani=1352.977912
 Arac basina alinan yol=6376.333333

Rapor(0-20640)
 Olay sayisi=1025
 Ortalama bekleme zamani=1388.908293
 Arac basina alinan yol=6573.666667

Rapor(0-21120)
 Olay sayisi=1043
 Ortalama bekleme zamani=1426.251198
 Arac basina alinan yol=6710.000000

Rapor(0-21600)
 Olay sayisi=1065
 Ortalama bekleme zamani=1454.692958
 Arac basina alinan yol=6862.000000

Rapor(0-22080)
 Olay sayisi=1088
 Ortalama bekleme zamani=1486.030331
 Arac basina alinan yol=7019.000000

Rapor(0-22560)
 Olay sayisi=1117
 Ortalama bekleme zamani=1527.279320
 Arac basina alinan yol=7194.666667

Rapor(0-23040)
 Olay sayisi=1146
 Ortalama bekleme zamani=1557.641361
 Arac basina alinan yol=7393.333333

Rapor(0-23520)
 Olay sayisi=1171
 Ortalama bekleme zamani=1583.021349
 Arac basina alinan yol=7554.666667

Rapor(0-24000)
 Olay sayisi=1191
 Ortalama bekleme zamani=1615.353484
 Arac basina alinan yol=7694.000000

**Ek 19 Yoğun Trafik Altında SAT Politikası Doğrultusunda 4 Araçlık Filo İçin
Koşturulan Simülasyonun Raporu**

Rapor(0-480)

Olay sayısı=18

Ortalama bekleme zamanı=42.944444

Arac basına alınan yol=86.500000

Rapor(0-960)

Olay sayısı=42

Ortalama bekleme zamanı=47.666667

Arac basına alınan yol=173.000000

Rapor(0-1440)

Olay sayısı=59

Ortalama bekleme zamanı=43.813559

Arac basına alınan yol=240.250000

Rapor(0-1920)

Olay sayısı=88

Ortalama bekleme zamanı=54.568182

Arac basına alınan yol=358.250000

Rapor(0-2400)

Olay sayısı=119

Ortalama bekleme zamanı=65.495798

Arac basına alınan yol=504.000000

Rapor(0-2880)

Olay sayısı=142

Ortalama bekleme zamanı=83.500000

Arac basına alınan yol=621.750000

Rapor(0-3360)

Olay sayısı=161

Ortalama bekleme zamanı=93.248447

Arac basına alınan yol=691.500000

Rapor(0-3840)

Olay sayısı=193

Ortalama bekleme zamanı=93.989637

Arac basına alınan yol=846.750000

Rapor(0-4320)

Olay sayısı=217

Ortalama bekleme zamanı=92.566820

Arac basına alınan yol=932.000000

Rapor(0-4800)

Olay sayısı=232

Ortalama bekleme zamanı=91.853448

Arac basına alınan yol=991.500000

Rapor(0-5280)
Olay sayisi=255
Ortalama bekleme zamani=88.254902
Arac basina alinan yol=1096.250000

Rapor(0-5760)
Olay sayisi=275
Ortalama bekleme zamani=84.454545
Arac basina alinan yol=1163.000000

Rapor(0-6240)
Olay sayisi=296
Ortalama bekleme zamani=81.692568
Arac basina alinan yol=1250.500000

Rapor(0-6720)
Olay sayisi=324
Ortalama bekleme zamani=79.842593
Arac basina alinan yol=1359.500000

Rapor(0-7200)
Olay sayisi=355
Ortalama bekleme zamani=80.439437
Arac basina alinan yol=1506.750000

Rapor(0-7680)
Olay sayisi=373
Ortalama bekleme zamani=90.423592
Arac basina alinan yol=1601.500000

Rapor(0-8160)
Olay sayisi=396
Ortalama bekleme zamani=95.237374
Arac basina alinan yol=1707.500000

Rapor(0-8640)
Olay sayisi=416
Ortalama bekleme zamani=93.012019
Arac basina alinan yol=1774.250000

Rapor(0-9120)
Olay sayisi=438
Ortalama bekleme zamani=90.721461
Arac basina alinan yol=1863.000000

Rapor(0-9600)
Olay sayisi=470
Ortalama bekleme zamani=87.412766
Arac basina alinan yol=2005.500000

Rapor(0-10080)
Olay sayisi=492
Ortalama bekleme zamani=89.871951
Arac basina alinan yol=2110.500000

Rapor(0-10560)
Olay sayisi=507
Ortalama bekleme zamani=88.863905
Arac basina alinan yol=2181.250000

Rapor(0-11040)
Olay sayisi=533
Ortalama bekleme zamani=87.440901
Arac basina alinan yol=2295.500000

Rapor(0-11520)
Olay sayisi=558
Ortalama bekleme zamani=86.569892
Arac basina alinan yol=2380.250000

Rapor(0-12000)
Olay sayisi=581
Ortalama bekleme zamani=84.882960
Arac basina alinan yol=2477.250000

Rapor(0-12480)
Olay sayisi=605
Ortalama bekleme zamani=83.852893
Arac basina alinan yol=2582.000000

Rapor(0-12960)
Olay sayisi=625
Ortalama bekleme zamani=82.851200
Arac basina alinan yol=2660.000000

Rapor(0-13440)
Olay sayisi=647
Ortalama bekleme zamani=81.871716
Arac basina alinan yol=2735.500000

Rapor(0-13920)
Olay sayisi=668
Ortalama bekleme zamani=80.658683
Arac basina alinan yol=2825.000000

Rapor(0-14400)
Olay sayisi=691
Ortalama bekleme zamani=79.406657
Arac basina alinan yol=2940.500000

Rapor(0-14880)
 Olay sayisi=723
 Ortalama bekleme zamani=80.517289
 Arac basina alinan yol=3079.750000

Rapor(0-15360)
 Olay sayisi=741
 Ortalama bekleme zamani=81.716599
 Arac basina alinan yol=3142.000000

Rapor(0-15840)
 Olay sayisi=767
 Ortalama bekleme zamani=82.061278
 Arac basina alinan yol=3250.250000

Rapor(0-16320)
 Olay sayisi=790
 Ortalama bekleme zamani=81.356962
 Arac basina alinan yol=3338.750000

Rapor(0-16800)
 Olay sayisi=815
 Ortalama bekleme zamani=80.758282
 Arac basina alinan yol=3432.250000

Rapor(0-17280)
 Olay sayisi=840
 Ortalama bekleme zamani=79.869048
 Arac basina alinan yol=3518.500000

Rapor(0-17760)
 Olay sayisi=857
 Ortalama bekleme zamani=79.074679
 Arac basina alinan yol=3585.500000

Rapor(0-18240)
 Olay sayisi=885
 Ortalama bekleme zamani=78.471186
 Arac basina alinan yol=3744.250000

Rapor(0-18720)
 Olay sayisi=915
 Ortalama bekleme zamani=78.946448
 Arac basina alinan yol=3902.250000

Rapor(0-19200)
 Olay sayisi=939
 Ortalama bekleme zamani=81.290735
 Arac basina alinan yol=4031.250000

Rapor(0-19680)
 Olay sayisi=964
 Ortalama bekleme zamani=81.891079
 Arac basina alinan yol=4143.500000

Rapor(0-20160)
 Olay sayisi=987
 Ortalama bekleme zamani=81.770010
 Arac basina alinan yol=4258.750000

Rapor(0-20640)
 Olay sayisi=1016
 Ortalama bekleme zamani=81.794291
 Arac basina alinan yol=4399.250000

Rapor(0-21120)
 Olay sayisi=1041
 Ortalama bekleme zamani=84.188280
 Arac basina alinan yol=4514.250000

Rapor(0-21600)
 Olay sayisi=1060
 Ortalama bekleme zamani=84.274528
 Arac basina alinan yol=4584.000000

Rapor(0-22080)
 Olay sayisi=1079
 Ortalama bekleme zamani=83.658017
 Arac basina alinan yol=4657.000000

Rapor(0-22560)
 Olay sayisi=1113
 Ortalama bekleme zamani=82.891285
 Arac basina alinan yol=4828.250000

Rapor(0-23040)
 Olay sayisi=1139
 Ortalama bekleme zamani=85.770852
 Arac basina alinan yol=4950.500000

Rapor(0-23520)
 Olay sayisi=1170
 Ortalama bekleme zamani=89.605128
 Arac basina alinan yol=5074.000000

Rapor(0-24000)
 Olay sayisi=1197
 Ortalama bekleme zamani=94.226399
 Arac basina alinan yol=5207.000000

**Ek 20 Yoğun Trafik Altında SAT Politikası Doğrultusunda 5 Araçlık Filo İçin
Koşturulan Simülasyonun Raporu**

Rapor(0-480)

Olay sayısı=23

Ortalama bekleme zamanı=45.347826

Arac basına alınan yol=83.200000

Rapor(0-960)

Olay sayısı=45

Ortalama bekleme zamanı=38.133333

Arac basına alınan yol=136.200000

Rapor(0-1440)

Olay sayısı=68

Ortalama bekleme zamanı=35.352941

Arac basına alınan yol=202.200000

Rapor(0-1920)

Olay sayısı=94

Ortalama bekleme zamanı=37.106383

Arac basına alınan yol=285.600000

Rapor(0-2400)

Olay sayısı=119

Ortalama bekleme zamanı=38.100840

Arac basına alınan yol=347.400000

Rapor(0-2880)

Olay sayısı=141

Ortalama bekleme zamanı=35.872340

Arac basına alınan yol=412.200000

Rapor(0-3360)

Olay sayısı=170

Ortalama bekleme zamanı=46.500000

Arac basına alınan yol=514.800000

Rapor(0-3840)

Olay sayısı=192

Ortalama bekleme zamanı=46.125000

Arac basına alınan yol=578.200000

Rapor(0-4320)

Olay sayısı=208

Ortalama bekleme zamanı=44.725962

Arac basına alınan yol=617.400000

Rapor(0-4800)

Olay sayısı=239

Ortalama bekleme zamanı=45.589958

Arac basına alınan yol=723.200000

Rapor(0-5280)
Olay sayisi=258
Ortalama bekleme zamani=49.926357
Arac basina alinan yol=798.800000

Rapor(0-5760)
Olay sayisi=284
Ortalama bekleme zamani=49.151408
Arac basina alinan yol=886.200000

Rapor(0-6240)
Olay sayisi=311
Ortalama bekleme zamani=51.257235
Arac basina alinan yol=971.600000

Rapor(0-6720)
Olay sayisi=338
Ortalama bekleme zamani=52.467456
Arac basina alinan yol=1055.800000

Rapor(0-7200)
Olay sayisi=365
Ortalama bekleme zamani=51.865753
Arac basina alinan yol=1139.600000

Rapor(0-7680)
Olay sayisi=392
Ortalama bekleme zamani=50.010204
Arac basina alinan yol=1241.000000

Rapor(0-8160)
Olay sayisi=416
Ortalama bekleme zamani=51.786058
Arac basina alinan yol=1333.800000

Rapor(0-8640)
Olay sayisi=440
Ortalama bekleme zamani=51.479545
Arac basina alinan yol=1419.600000

Rapor(0-9120)
Olay sayisi=460
Ortalama bekleme zamani=51.817391
Arac basina alinan yol=1501.400000

Rapor(0-9600)
Olay sayisi=489
Ortalama bekleme zamani=50.963190
Arac basina alinan yol=1594.000000

Rapor(0-10080)
Olay sayisi=507
Ortalama bekleme zamani=50.258383
Arac basina alinan yol=1629.600000

Rapor(0-10560)
Olay sayisi=526
Ortalama bekleme zamani=49.509506
Arac basina alinan yol=1675.800000

Rapor(0-11040)
Olay sayisi=552
Ortalama bekleme zamani=48.469203
Arac basina alinan yol=1744.200000

Rapor(0-11520)
Olay sayisi=575
Ortalama bekleme zamani=48.572174
Arac basina alinan yol=1828.200000

Rapor(0-12000)
Olay sayisi=613
Ortalama bekleme zamani=50.344209
Arac basina alinan yol=1938.000000

Rapor(0-12480)
Olay sayisi=638
Ortalama bekleme zamani=50.341693
Arac basina alinan yol=2025.400000

Rapor(0-12960)
Olay sayisi=661
Ortalama bekleme zamani=50.057489
Arac basina alinan yol=2090.200000

Rapor(0-13440)
Olay sayisi=681
Ortalama bekleme zamani=49.443465
Arac basina alinan yol=2162.800000

Rapor(0-13920)
Olay sayisi=700
Ortalama bekleme zamani=49.215714
Arac basina alinan yol=2214.000000

Rapor(0-14400)
Olay sayisi=732
Ortalama bekleme zamani=48.814208
Arac basina alinan yol=2290.600000

Rapor(0-14880)
Olay sayisi=755
Ortalama bekleme zamani=48.512583
Arac basina alinan yol=2362.200000

Rapor(0-15360)
Olay sayisi=774
Ortalama bekleme zamani=48.025840
Arac basina alinan yol=2413.600000

Rapor(0-15840)
Olay sayisi=799
Ortalama bekleme zamani=47.619524
Arac basina alinan yol=2487.400000

Rapor(0-16320)
Olay sayisi=827
Ortalama bekleme zamani=47.977025
Arac basina alinan yol=2571.200000

Rapor(0-16800)
Olay sayisi=851
Ortalama bekleme zamani=48.321974
Arac basina alinan yol=2662.600000

Rapor(0-17280)
Olay sayisi=873
Ortalama bekleme zamani=47.957617
Arac basina alinan yol=2713.800000

Rapor(0-17760)
Olay sayisi=900
Ortalama bekleme zamani=47.226667
Arac basina alinan yol=2786.600000

Rapor(0-18240)
Olay sayisi=921
Ortalama bekleme zamani=48.059718
Arac basina alinan yol=2875.400000

Rapor(0-18720)
Olay sayisi=947
Ortalama bekleme zamani=47.709609
Arac basina alinan yol=2932.800000

Rapor(0-19200)
Olay sayisi=965
Ortalama bekleme zamani=47.538860
Arac basina alinan yol=2985.600000

Rapor(0-19680)
 Olay sayisi=982
 Ortalama bekleme zamani=47.308554
 Arac basina alinan yol=3029.800000

Rapor(0-20160)
 Olay sayisi=1000
 Ortalama bekleme zamani=46.946000
 Arac basina alinan yol=3075.800000

Rapor(0-20640)
 Olay sayisi=1027
 Ortalama bekleme zamani=46.627069
 Arac basina alinan yol=3162.200000

Rapor(0-21120)
 Olay sayisi=1056
 Ortalama bekleme zamani=46.920455
 Arac basina alinan yol=3259.600000

Rapor(0-21600)
 Olay sayisi=1079
 Ortalama bekleme zamani=46.825765
 Arac basina alinan yol=3342.200000

Rapor(0-22080)
 Olay sayisi=1102
 Ortalama bekleme zamani=46.411978
 Arac basina alinan yol=3409.800000

Rapor(0-22560)
 Olay sayisi=1114
 Ortalama bekleme zamani=46.386894
 Arac basina alinan yol=3441.200000

Rapor(0-23040)
 Olay sayisi=1144
 Ortalama bekleme zamani=46.170455
 Arac basina alinan yol=3526.800000

Rapor(0-23520)
 Olay sayisi=1164
 Ortalama bekleme zamani=46.062715
 Arac basina alinan yol=3585.400000

Rapor(0-24000)
 Olay sayisi=1195
 Ortalama bekleme zamani=45.820921
 Arac basina alinan yol=3666.800000

**Ek 21 Yoğun Trafik Altında SAT Politikası Doğrultusunda 6 Araçlık Filo İçin
Koşturulan Simülasyonun Raporu**

Rapor(0-480)

Olay sayisi=27

Ortalama bekleme zamanı=30.222222

Arac basına alınan yol=59.833333

Rapor(0-960)

Olay sayisi=42

Ortalama bekleme zamanı=29.285714

Arac basına alınan yol=85.333333

Rapor(0-1440)

Olay sayisi=64

Ortalama bekleme zamanı=29.234375

Arac basına alınan yol=127.666667

Rapor(0-1920)

Olay sayisi=87

Ortalama bekleme zamanı=28.264368

Arac basına alınan yol=167.000000

Rapor(0-2400)

Olay sayisi=112

Ortalama bekleme zamanı=30.267857

Arac basına alınan yol=234.166667

Rapor(0-2880)

Olay sayisi=145

Ortalama bekleme zamanı=31.027586

Arac basına alınan yol=300.333333

Rapor(0-3360)

Olay sayisi=174

Ortalama bekleme zamanı=31.247126

Arac basına alınan yol=358.833333

Rapor(0-3840)

Olay sayisi=196

Ortalama bekleme zamanı=31.632653

Arac basına alınan yol=415.833333

Rapor(0-4320)

Olay sayisi=230

Ortalama bekleme zamanı=31.265217

Arac basına alınan yol=480.000000

Rapor(0-4800)

Olay sayisi=249

Ortalama bekleme zamanı=31.425703

Arac basına alınan yol=525.333333

Rapor(0-5280)
Olay sayisi=280
Ortalama bekleme zamani=30.882143
Arac basina alinan yol=585.833333

Rapor(0-5760)
Olay sayisi=305
Ortalama bekleme zamani=32.262295
Arac basina alinan yol=651.666667

Rapor(0-6240)
Olay sayisi=329
Ortalama bekleme zamani=31.930091
Arac basina alinan yol=696.500000

Rapor(0-6720)
Olay sayisi=346
Ortalama bekleme zamani=31.696532
Arac basina alinan yol=729.166667

Rapor(0-7200)
Olay sayisi=366
Ortalama bekleme zamani=31.303279
Arac basina alinan yol=765.500000

Rapor(0-7680)
Olay sayisi=389
Ortalama bekleme zamani=30.987147
Arac basina alinan yol=810.333333

Rapor(0-8160)
Olay sayisi=421
Ortalama bekleme zamani=31.282660
Arac basina alinan yol=885.500000

Rapor(0-8640)
Olay sayisi=441
Ortalama bekleme zamani=31.292517
Arac basina alinan yol=932.166667

Rapor(0-9120)
Olay sayisi=470
Ortalama bekleme zamani=31.648936
Arac basina alinan yol=997.000000

Rapor(0-9600)
Olay sayisi=490
Ortalama bekleme zamani=31.604082
Arac basina alinan yol=1039.000000

Rapor(0-10080)
Olay sayisi=525
Ortalama bekleme zamani=31.297143
Arac basina alinan yol=1113.833333

Rapor(0-10560)
Olay sayisi=548
Ortalama bekleme zamani=31.507299
Arac basina alinan yol=1158.333333

Rapor(0-11040)
Olay sayisi=575
Ortalama bekleme zamani=31.377391
Arac basina alinan yol=1213.166667

Rapor(0-11520)
Olay sayisi=598
Ortalama bekleme zamani=31.603679
Arac basina alinan yol=1265.333333

Rapor(0-12000)
Olay sayisi=624
Ortalama bekleme zamani=31.649038
Arac basina alinan yol=1324.666667

Rapor(0-12480)
Olay sayisi=648
Ortalama bekleme zamani=31.547840
Arac basina alinan yol=1376.666667

Rapor(0-12960)
Olay sayisi=677
Ortalama bekleme zamani=31.775480
Arac basina alinan yol=1447.500000

Rapor(0-13440)
Olay sayisi=696
Ortalama bekleme zamani=31.734195
Arac basina alinan yol=1479.833333

Rapor(0-13920)
Olay sayisi=716
Ortalama bekleme zamani=31.484637
Arac basina alinan yol=1522.000000

Rapor(0-14400)
Olay sayisi=733
Ortalama bekleme zamani=31.503411
Arac basina alinan yol=1558.333333

Rapor(0-14880)
Olay sayisi=754
Ortalama bekleme zamani=31.376658
Arac basina alinan yol=1593.500000

Rapor(0-15360)
Olay sayisi=770
Ortalama bekleme zamani=31.376623
Arac basina alinan yol=1631.333333

Rapor(0-15840)
Olay sayisi=799
Ortalama bekleme zamani=31.678348
Arac basina alinan yol=1711.166667

Rapor(0-16320)
Olay sayisi=818
Ortalama bekleme zamani=32.121027
Arac basina alinan yol=1754.166667

Rapor(0-16800)
Olay sayisi=842
Ortalama bekleme zamani=31.945368
Arac basina alinan yol=1799.500000

Rapor(0-17280)
Olay sayisi=861
Ortalama bekleme zamani=31.904762
Arac basina alinan yol=1834.000000

Rapor(0-17760)
Olay sayisi=879
Ortalama bekleme zamani=32.038680
Arac basina alinan yol=1891.666667

Rapor(0-18240)
Olay sayisi=900
Ortalama bekleme zamani=32.236667
Arac basina alinan yol=1945.666667

Rapor(0-18720)
Olay sayisi=928
Ortalama bekleme zamani=32.237069
Arac basina alinan yol=2012.333333

Rapor(0-19200)
Olay sayisi=953
Ortalama bekleme zamani=32.295908
Arac basina alinan yol=2065.666667

Rapor(0-19680)
 Olay sayisi=979
 Ortalama bekleme zamani=32.188968
 Arac basina alinan yol=2130.000000

Rapor(0-20160)
 Olay sayisi=1005
 Ortalama bekleme zamani=32.207960
 Arac basina alinan yol=2181.500000

Rapor(0-20640)
 Olay sayisi=1026
 Ortalama bekleme zamani=32.205653
 Arac basina alinan yol=2218.333333

Rapor(0-21120)
 Olay sayisi=1049
 Ortalama bekleme zamani=32.116301
 Arac basina alinan yol=2262.500000

Rapor(0-21600)
 Olay sayisi=1067
 Ortalama bekleme zamani=31.984067
 Arac basina alinan yol=2304.833333

Rapor(0-22080)
 Olay sayisi=1090
 Ortalama bekleme zamani=32.152294
 Arac basina alinan yol=2367.500000

Rapor(0-22560)
 Olay sayisi=1111
 Ortalama bekleme zamani=32.106211
 Arac basina alinan yol=2414.333333

Rapor(0-23040)
 Olay sayisi=1140
 Ortalama bekleme zamani=32.003509
 Arac basina alinan yol=2478.833333

Rapor(0-23520)
 Olay sayisi=1166
 Ortalama bekleme zamani=32.099485
 Arac basina alinan yol=2542.500000

Rapor(0-24000)
 Olay sayisi=1188
 Ortalama bekleme zamani=32.239057
 Arac basina alinan yol=2589.833333

**Ek 22 Yoğun Trafik Altında SAT Politikası Doğrultusunda 7 Araçlık Filo İçin
Koşturulan Simülasyonun Raporu**

Rapor(0-480)

Olay sayisi=23

Ortalama bekleme zamani=32.913043

Arac basina alinan yol=49.000000

Rapor(0-960)

Olay sayisi=44

Ortalama bekleme zamani=27.409091

Arac basina alinan yol=72.142857

Rapor(0-1440)

Olay sayisi=68

Ortalama bekleme zamani=27.029412

Arac basina alinan yol=118.428571

Rapor(0-1920)

Olay sayisi=87

Ortalama bekleme zamani=28.022989

Arac basina alinan yol=149.857143

Rapor(0-2400)

Olay sayisi=109

Ortalama bekleme zamani=28.495413

Arac basina alinan yol=188.571429

Rapor(0-2880)

Olay sayisi=142

Ortalama bekleme zamani=29.246479

Arac basina alinan yol=256.142857

Rapor(0-3360)

Olay sayisi=158

Ortalama bekleme zamani=29.170886

Arac basina alinan yol=277.000000

Rapor(0-3840)

Olay sayisi=183

Ortalama bekleme zamani=28.786885

Arac basina alinan yol=317.714286

Rapor(0-4320)

Olay sayisi=202

Ortalama bekleme zamani=29.009901

Arac basina alinan yol=350.000000

Rapor(0-4800)

Olay sayisi=230

Ortalama bekleme zamani=28.869565

Arac basina alinan yol=388.857143

Rapor(0-5280)
Olay sayisi=250
Ortalama bekleme zamani=28.796000
Arac basina alinan yol=422.571429

Rapor(0-5760)
Olay sayisi=269
Ortalama bekleme zamani=28.520446
Arac basina alinan yol=449.285714

Rapor(0-6240)
Olay sayisi=289
Ortalama bekleme zamani=28.484429
Arac basina alinan yol=485.571429

Rapor(0-6720)
Olay sayisi=311
Ortalama bekleme zamani=28.633441
Arac basina alinan yol=527.714286

Rapor(0-7200)
Olay sayisi=337
Ortalama bekleme zamani=28.910979
Arac basina alinan yol=573.285714

Rapor(0-7680)
Olay sayisi=358
Ortalama bekleme zamani=28.675978
Arac basina alinan yol=602.428571

Rapor(0-8160)
Olay sayisi=388
Ortalama bekleme zamani=29.072165
Arac basina alinan yol=657.428571

Rapor(0-8640)
Olay sayisi=421
Ortalama bekleme zamani=29.548694
Arac basina alinan yol=723.571429

Rapor(0-9120)
Olay sayisi=449
Ortalama bekleme zamani=29.612472
Arac basina alinan yol=773.714286

Rapor(0-9600)
Olay sayisi=472
Ortalama bekleme zamani=29.487288
Arac basina alinan yol=806.714286

Rapor(0-10080)
 Olay sayisi=492
 Ortalama bekleme zamani=29.530488
 Arac basina alinan yol=838.142857

Rapor(0-10560)
 Olay sayisi=510
 Ortalama bekleme zamani=29.535294
 Arac basina alinan yol=868.000000

Rapor(0-11040)
 Olay sayisi=526
 Ortalama bekleme zamani=29.340304
 Arac basina alinan yol=892.285714

Rapor(0-11520)
 Olay sayisi=551
 Ortalama bekleme zamani=29.960073
 Arac basina alinan yol=950.000000

Rapor(0-12000)
 Olay sayisi=572
 Ortalama bekleme zamani=29.833916
 Arac basina alinan yol=982.142857

Rapor(0-12480)
 Olay sayisi=598
 Ortalama bekleme zamani=30.021739
 Arac basina alinan yol=1029.428571

Rapor(0-12960)
 Olay sayisi=624
 Ortalama bekleme zamani=29.979167
 Arac basina alinan yol=1075.428571

Rapor(0-13440)
 Olay sayisi=658
 Ortalama bekleme zamani=29.951368
 Arac basina alinan yol=1132.285714

Rapor(0-13920)
 Olay sayisi=684
 Ortalama bekleme zamani=30.157895
 Arac basina alinan yol=1174.857143

Rapor(0-14400)
 Olay sayisi=705
 Ortalama bekleme zamani=29.967376
 Arac basina alinan yol=1202.857143

Rapor(0-14880)
Olay sayisi=732
Ortalama bekleme zamani=29.831967
Arac basina alinan yol=1244.000000

Rapor(0-15360)
Olay sayisi=751
Ortalama bekleme zamani=29.934754
Arac basina alinan yol=1280.571429

Rapor(0-15840)
Olay sayisi=779
Ortalama bekleme zamani=29.866496
Arac basina alinan yol=1336.714286

Rapor(0-16320)
Olay sayisi=803
Ortalama bekleme zamani=30.557908
Arac basina alinan yol=1381.428571

Rapor(0-16800)
Olay sayisi=824
Ortalama bekleme zamani=30.468447
Arac basina alinan yol=1411.857143

Rapor(0-17280)
Olay sayisi=851
Ortalama bekleme zamani=30.226792
Arac basina alinan yol=1446.428571

Rapor(0-17760)
Olay sayisi=875
Ortalama bekleme zamani=30.300571
Arac basina alinan yol=1491.000000

Rapor(0-18240)
Olay sayisi=898
Ortalama bekleme zamani=30.153675
Arac basina alinan yol=1524.571429

Rapor(0-18720)
Olay sayisi=923
Ortalama bekleme zamani=30.032503
Arac basina alinan yol=1559.428571

Rapor(0-19200)
Olay sayisi=949
Ortalama bekleme zamani=29.838778
Arac basina alinan yol=1593.000000

Rapor(0-19680)
Olay sayisi=969
Ortalama bekleme zamani=29.691434
Arac basina alinan yol=1616.000000

Rapor(0-20160)
Olay sayisi=987
Ortalama bekleme zamani=29.639311
Arac basina alinan yol=1640.000000

Rapor(0-20640)
Olay sayisi=1014
Ortalama bekleme zamani=29.493097
Arac basina alinan yol=1679.857143

Rapor(0-21120)
Olay sayisi=1043
Ortalama bekleme zamani=29.508150
Arac basina alinan yol=1726.857143

Rapor(0-21600)
Olay sayisi=1069
Ortalama bekleme zamani=29.405987
Arac basina alinan yol=1761.714286

Rapor(0-22080)
Olay sayisi=1098
Ortalama bekleme zamani=29.270492
Arac basina alinan yol=1807.285714

Rapor(0-22560)
Olay sayisi=1124
Ortalama bekleme zamani=29.276690
Arac basina alinan yol=1845.571429

Rapor(0-23040)
Olay sayisi=1150
Ortalama bekleme zamani=29.234783
Arac basina alinan yol=1892.285714

Rapor(0-23520)
Olay sayisi=1177
Ortalama bekleme zamani=29.344095
Arac basina alinan yol=1944.285714

Rapor(0-24000)
Olay sayisi=1196
Ortalama bekleme zamani=29.316890
Arac basina alinan yol=1970.285714

**Ek 23 Yoğun Trafik Altında SAT Politikası Doğrultusunda 8 Araçlık Filo İçin
Koşturulan Simülasyonun Raporu**

Rapor(0-480)

Olay sayısı=24

Ortalama bekleme zamanı=26.791667

Arac basına alınan yol=33.375000

Rapor(0-960)

Olay sayısı=50

Ortalama bekleme zamanı=25.960000

Arac basına alınan yol=66.375000

Rapor(0-1440)

Olay sayısı=91

Ortalama bekleme zamanı=25.824176

Arac basına alınan yol=127.250000

Rapor(0-1920)

Olay sayısı=114

Ortalama bekleme zamanı=26.938596

Arac basına alınan yol=156.250000

Rapor(0-2400)

Olay sayısı=137

Ortalama bekleme zamanı=25.554745

Arac basına alınan yol=178.500000

Rapor(0-2880)

Olay sayısı=163

Ortalama bekleme zamanı=25.625767

Arac basına alınan yol=208.625000

Rapor(0-3360)

Olay sayısı=185

Ortalama bekleme zamanı=25.729730

Arac basına alınan yol=243.750000

Rapor(0-3840)

Olay sayısı=201

Ortalama bekleme zamanı=25.805970

Arac basına alınan yol=260.125000

Rapor(0-4320)

Olay sayısı=231

Ortalama bekleme zamanı=25.727273

Arac basına alınan yol=297.000000

Rapor(0-4800)

Olay sayısı=261

Ortalama bekleme zamanı=25.455939

Arac basına alınan yol=332.000000

Rapor(0-5280)
 Olay sayisi=292
 Ortalama bekleme zamani=26.410959
 Arac basina alinan yol=378.125000

Rapor(0-5760)
 Olay sayisi=317
 Ortalama bekleme zamani=26.394322
 Arac basina alinan yol=412.500000

Rapor(0-6240)
 Olay sayisi=340
 Ortalama bekleme zamani=26.467647
 Arac basina alinan yol=444.250000

Rapor(0-6720)
 Olay sayisi=358
 Ortalama bekleme zamani=26.215084
 Arac basina alinan yol=465.000000

Rapor(0-7200)
 Olay sayisi=392
 Ortalama bekleme zamani=27.244898
 Arac basina alinan yol=531.875000

Rapor(0-7680)
 Olay sayisi=415
 Ortalama bekleme zamani=27.448193
 Arac basina alinan yol=563.875000

Rapor(0-8160)
 Olay sayisi=432
 Ortalama bekleme zamani=27.347222
 Arac basina alinan yol=582.875000

Rapor(0-8640)
 Olay sayisi=456
 Ortalama bekleme zamani=27.107456
 Arac basina alinan yol=610.625000

Rapor(0-9120)
 Olay sayisi=471
 Ortalama bekleme zamani=26.881104
 Arac basina alinan yol=624.000000

Rapor(0-9600)
 Olay sayisi=500
 Ortalama bekleme zamani=26.516000
 Arac basina alinan yol=651.375000

Rapor(0-10080)
Olay sayisi=511
Ortalama bekleme zamani=26.430528
Arac basina alinan yol=663.250000

Rapor(0-10560)
Olay sayisi=531
Ortalama bekleme zamani=26.218456
Arac basina alinan yol=684.750000

Rapor(0-11040)
Olay sayisi=554
Ortalama bekleme zamani=26.095668
Arac basina alinan yol=710.125000

Rapor(0-11520)
Olay sayisi=568
Ortalama bekleme zamani=26.059859
Arac basina alinan yol=724.000000

Rapor(0-12000)
Olay sayisi=595
Ortalama bekleme zamani=25.978151
Arac basina alinan yol=758.000000

Rapor(0-12480)
Olay sayisi=624
Ortalama bekleme zamani=26.241987
Arac basina alinan yol=799.125000

Rapor(0-12960)
Olay sayisi=643
Ortalama bekleme zamani=26.172628
Arac basina alinan yol=821.750000

Rapor(0-13440)
Olay sayisi=670
Ortalama bekleme zamani=26.180597
Arac basina alinan yol=859.375000

Rapor(0-13920)
Olay sayisi=702
Ortalama bekleme zamani=26.243590
Arac basina alinan yol=902.125000

Rapor(0-14400)
Olay sayisi=729
Ortalama bekleme zamani=26.325103
Arac basina alinan yol=938.375000

Rapor(0-14880)
Olay sayisi=761
Ortalama bekleme zamani=26.160315
Arac basina alinan yol=973.500000

Rapor(0-15360)
Olay sayisi=783
Ortalama bekleme zamani=26.185185
Arac basina alinan yol=1002.875000

Rapor(0-15840)
Olay sayisi=808
Ortalama bekleme zamani=26.252475
Arac basina alinan yol=1042.375000

Rapor(0-16320)
Olay sayisi=825
Ortalama bekleme zamani=26.191515
Arac basina alinan yol=1063.375000

Rapor(0-16800)
Olay sayisi=852
Ortalama bekleme zamani=26.223005
Arac basina alinan yol=1107.750000

Rapor(0-17280)
Olay sayisi=868
Ortalama bekleme zamani=26.299539
Arac basina alinan yol=1126.500000

Rapor(0-17760)
Olay sayisi=893
Ortalama bekleme zamani=26.138858
Arac basina alinan yol=1158.375000

Rapor(0-18240)
Olay sayisi=925
Ortalama bekleme zamani=26.255135
Arac basina alinan yol=1208.000000

Rapor(0-18720)
Olay sayisi=951
Ortalama bekleme zamani=26.249211
Arac basina alinan yol=1238.500000

Rapor(0-19200)
Olay sayisi=979
Ortalama bekleme zamani=26.307457
Arac basina alinan yol=1276.875000

Rapor(0-19680)
Olay sayisi=1009
Ortalama bekleme zamani=26.306244
Arac basina alinan yol=1312.625000

Rapor(0-20160)
Olay sayisi=1028
Ortalama bekleme zamani=26.181907
Arac basina alinan yol=1330.875000

Rapor(0-20640)
Olay sayisi=1044
Ortalama bekleme zamani=26.167625
Arac basina alinan yol=1347.500000

Rapor(0-21120)
Olay sayisi=1065
Ortalama bekleme zamani=26.197183
Arac basina alinan yol=1378.500000

Rapor(0-21600)
Olay sayisi=1079
Ortalama bekleme zamani=26.083411
Arac basina alinan yol=1389.250000

Rapor(0-22080)
Olay sayisi=1097
Ortalama bekleme zamani=26.051960
Arac basina alinan yol=1411.000000

Rapor(0-22560)
Olay sayisi=1116
Ortalama bekleme zamani=25.948925
Arac basina alinan yol=1429.875000

Rapor(0-23040)
Olay sayisi=1136
Ortalama bekleme zamani=25.870599
Arac basina alinan yol=1456.250000

Rapor(0-23520)
Olay sayisi=1168
Ortalama bekleme zamani=26.086473
Arac basina alinan yol=1508.625000

Rapor(0-24000)
Olay sayisi=1194
Ortalama bekleme zamani=26.041876
Arac basina alinan yol=1539.250000

ÖZGEÇMİŞ

Doğum tarihi 26.08.1979

Doğum yeri Kastamonu

Lise 1994-1997 İzmir Fen Lisesi

Lisans 1997-2001 Bilkent Üniversitesi Mühendislik Fak.
Endüstri Mühendisliği Bölümü (Burslu)

Yüksek Lisans 2001-2004 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Endüstri Müh. Anabilim Dalı, Endüstri Mühendisliği
Programı

Çalıştığı kurum(lar)

2001-2004 İŞ-TİM Telekomünikasyon Hizmetleri A.Ş.
2004-Devam ediyor TT&TİM İletişim Hizmetleri A.Ş.

