

PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

**İYİLEŞTİRİLMİŞ PEKİŞTİRMELİ ÖĞRENME YÖNTEMİ VE DİNAMİK
YÜKLEME İLE KENTİÇİ ULAŞIM AĞLARININ TASARIMI**

**DOKTORA TEZİ
Cenk OZAN**

Anabilim Dalı : İnşaat Mühendisliği

Programı : Ulaştırma

Tez Danışmanı: Doç. Dr. Soner HALDENBİLEN

TEMMUZ 2012

DOKTORA TEZ ONAY FORMU

Pamukkale Üniversitesi Fen Bilimleri Enstitüsü 091625005 nolu öğrencisi Cenk OZAN tarafından hazırlanan “İYİLEŞTİRİLMİŞ PEKİŞTİRMELİ ÖĞRENME YÖNTEMİ VE DİNAMİK YÜKLEME İLE KENTİÇİ ULAŞIM AĞLARININ TASARIMI” başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından bir Doktora tezi olarak kabul edilmiştir.

Tez Danışmanı : Doç. Dr. Soner HALDENBİLEN (PAÜ)

Jüri Üyesi : Doç. Dr. Halim CEYLAN (PAÜ)

Jüri Üyesi : Prof. Dr. Mehmet SALTAN (SDÜ)
(Jüri Başkanı)

Jüri Üyesi : Doç. Dr. Yetiş Şazi MURAT (PAÜ)

Jüri Üyesi : Doç. Dr. Serhan TANYEL (DEÜ)

Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 01.08.2012 tarih ve19119.... sayılı kararıyla onaylanmıştır.


Fen Bilimleri Enstitüsü Müdürü
Prof. Dr. Nuri KOLSUZ

Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu çalıřmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan çalıřmalara atfedildiđine beyan ederim.

İmza

: 

Öğrenci Adı Soyadı: Cenk OZAN

ÖNSÖZ

Tez çalışmam süresince destek ve görüşlerini benden esirgemeyen danışmanım değerli hocam Doç. Dr. Soner HALDENBİLEN'e şükranlarımı sunarım.

Tez çalışmam boyunca desteğini esirgemeyen ve bu konudaki bilgi ve tecrübesini bana aktararak her zaman yardımcı olan saygıdeğer hocam Doç. Dr. Halim CEYLAN'a bütün içtenliğimle teşekkür ederim.

Tez çalışması sürecinde desteğini esirgemeyen ve manevi olarak her zaman yanımda olan dostum Dr. Özgür BAŞKAN'a teşekkürlerimi sunarım.

Tez çalışmasında yazılım desteği veren PTV AG şirketine ve bu çalışmayı 2010-FBE-063 nolu proje ile destekleyen Pamukkale Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimine teşekkür ederim.

Hayatım boyunca maddi, manevi her türlü desteği veren sevgili annem Ecz. Güler OZAN'a, kardeşim Şehir Plancısı Çağlar OZAN'a teşekkürlerimi sunarım.

Doktora sürecinde karşılaştığım zorluklarda yaşadığı ağır bedellere rağmen desteği ve sabrı ile her zaman yanımda olan sevgili eşim ve hayat arkadaşım Dr. Ayşe YÜKSEL OZAN'a bütün kalbimle teşekkür ederim.

Temmuz 2012

Cenk OZAN

İnşaat Yüksek Mühendisi

Babamın anısına...

İÇİNDEKİLER

Sayfa

ÖZET.....	xvii
SUMMARY	xviii
1. GİRİŞ	1
1.1 Ulaşım Ağ Tasarımı	1
1.2 Problemin Tanımı.....	5
1.3 Amaç	7
1.4 Tez Düzeni	9
2. ÖNCEKİ ÇALIŞMALAR	11
2.1 Giriş.....	11
2.2 Ulaşım Ağ Tasarımı	12
2.2.1 İki Seviyeli Programlama.....	17
2.2.2 Karşılıklı İteratif Yaklaşım	19
2.3 Trafik Atama	20
2.4 Dinamik Trafik Atama	24
2.4.1 Dinamik Trafik Atama Yöntemleri.....	27
2.4.2 SO ve KD Altında Benzetim Bazlı DTA Modeli	37
2.4.3 Dynasmart-P Benzetim Bazlı Dinamik Trafik Atama Yazılımı	43
2.5 Trafik Sinyal Kontrolü	44
2.5.1 TRANSYT-7F Trafik Modeli	46
2.5.2 Sinyal Kontrolü İle İlgili Yapılan Çalışmalar	54
2.6 Pekiştirmeli Öğrenme Yöntemi.....	56
2.7 Tez Çalışmasının Genel Çerçevesi.....	58
2.8 Sonuçlar.....	59
3. PEKİŞTİRMELİ ÖĞRENME YÖNTEMİ	61
3.1 Giriş.....	61
3.2 Pekiştirmeli Öğrenme Yöntemi.....	61
3.3 İyileştirilmiş Pekiştirmeli Öğrenme Yöntemi	65
3.4 İyileştirilmiş PÖ Yönteminin Sinyal Kontrol Problemine Uyarlanması.....	77
3.5 Sonuçlar.....	91
4. PROBLEM FORMÜLASYONU VE MODEL GELİŞİMİ	93
4.1 Giriş.....	93
4.2 Dynasmart-P Benzetim Bazlı Dinamik Trafik Atama Yazılımı	93
4.2.1 Dynasmart-P Algoritması	95
4.2.2 Zamana Bağlı En Kısa Rota Algoritması.....	100
4.3 Dinamik Kullanıcı Dengesi Altında UAT Modeli	106
4.3.1 PÖTRAD Modeli	106
4.3.2 PÖTRAD Modeli Çözüm Algoritması	108
4.4 PÖTRAD Modeli Uygulaması	110

4.5 Sonular.....	117
5. SAYISAL UYGULAMA	119
5.1 Giriş.....	119
5.2 PÖTRAD Modeli Sayısal Uygulaması	119
5.2.1 Senaryo 1.....	125
5.2.2 Senaryo 2.....	127
5.2.3 Senaryo 3.....	129
5.2.4 Senaryo 4.....	131
5.3 Senaryoların Karşılaştırılması	132
5.4 Taşıt Emisyonlarının Hesaplanması.....	135
5.5 Sonular.....	139
6. SONULAR VE ÖNERİLER	141
6.1 Giriş.....	141
6.2 Sonular.....	141
6.3 Gelecek alıřmalar	146
KAYNAKLAR	147
EKLER.....	162

KISALTMALAR

AA	: Armoni Araştırması
AB	: Avrupa Birliği
ARRB	: Avustralya Yol Araştırma Dairesi
AOY	: Ardışık Ortalamalar Yöntemi
ATK	: Alansal Trafik Kontrolü
UAT	: Ulaşım Ağ Tasarımı
BO	: Başarı Oranı
B-V	: Başlangıç-Varış
CH_x	: Hidrokarbon
CO	: Karbon Monoksit
ÇU	: Çözüm Uzayı
DI	: Yararsızlık İndeksi
DKD	: Dinamik Kullanıcı Dengesi
DSO	: Dinamik Sistem Optimumu
DTA	: Dinamik Trafik Atama
Dynasmart-P	: Dynamic Network Assignment-Simulation Model for Advanced Roadway Telematics Planning version
FHWA	: Amerikan Federal Otoyol İdaresi
GA	: Genetik Algoritma
GAP	: Genetik Ağ Programlama
HC	: Hill-Climbing eniyileme modeli
HCM	: Highway Capacity Manual
İGİÇ	: İlk Giren İlk Çıkar
KD	: Kullanıcı Dengesi
KİY	: Karşılıklı İteratif Yöntem
KKO	: Karınca Kolonisi Optimizasyonu
KS	: Karınca Sistemi
KUAT	: Kesikli Ulaşım Ağ Tasarımı
MD	: Mevcut Değil
MKS	: Markov Karar Süreçleri
M-N	: Merchant ve Nemhauser modeli
İPÖ	: İyileştirilmiş Pekiştirmeli Öğrenme
NO_x	: Nitrojen Oksit
OSL	: Ordinal Sharing Learning
PFE	: Path Flow Estimator (Rota Akım Tahmin Edicisi)
PI	: Performans İndeksi
PÖ	: Pekiştirmeli Öğrenme
PÖTRAD	: İyileştirilmiş Pekiştirmeli Öğrenme TRANSYT-7F Dynasmart-P
PÖTRANS	: İyileştirilmiş Pekiştirmeli Öğrenme TRANSYT-7F
PSO	: Parçacık Sürüsü Optimizasyonu
SO	: Sistem Optimumu
SUAT	: Sürekli Ulaşım Ağ Tasarımı
TA	: Trafik Atama

TAR	: Tabu Arama
TB	: Tavlama Benzetimi
TDSP	: Zamana Baęlı En Kısa Rota Algoritması
TRANSYT	: TRAffic Network StudY Tool
TRANSYT-7F	: TRAffic Network StudY Tool, version 7F
TRB	: Transportation Research Board
TRRL	: İngiltere Ulaşım ve Yol Araştırma Laboratuvarı
UCTS	: Şehiriçi Trafik Kontrol Sistemi
VI	: Variational Inequality (Varyasyonel Eşitsizlik)
VT	: Varış Tipi

TABLO LİSTESİ

Tablolar

2.1 : Üniorm gecikme hesabı için koordine sinyal düzeltme faktörü (<i>PF</i>)	51
2.2 : Ölçüm düzeltme faktörleri.....	53
3.1 : <i>Q</i> tablosu örneği.....	64
3.2 : Pekiştirmeli öğrenme süreci (Kaelbling vd., 1996).....	64
3.3 : Tutarlılık analizi sonuçları.....	71
3.4 : İPÖ algoritması ile kıyaslanan algoritmalar.....	73
3.5 : İPÖ algoritmasının ve kıyaslanan diğer algoritmaların sonuçları	75
3.6 : Örnek ulaşım ağı sabit bağ akımları (Ceylan, 2002).....	84
3.7 : Örnek ulaşım ağı için en iyi <i>PI</i> değeri ve sinyal parametreleri.....	86
3.8 : Allsop ve Charlesworth ulaşım ağı sabit bağ akımları (Ceylan, 2006).....	87
3.9 : Allsop ve Charlesworth ulaşım ağı için en iyi <i>PI</i> değeri ve sinyal parametreleri.....	91
4.1 : Örnek ulaşım ağı verileri.....	110
4.2 : Örnek ulaşım ağında en iyi sinyal parametreleri ve <i>PI</i> değeri (yükleme profili 1).....	114
4.3 : Örnek ulaşım ağında yüklem profili 1'e göre dinamik denge bağ akımları (ta/sa).....	114
4.4 : Örnek ulaşım ağında en iyi sinyal parametreleri ve <i>PI</i> değeri (yüklem profili 2).....	115
4.5 : Örnek ulaşım ağında yüklem profili 2'ye göre dinamik denge bağ akımları (ta/sa).....	115
4.6 : Örnek ulaşım ağında PÖTRAD modelinin uygulanması sonucu elde edilen istatistikler.....	116
5.1 : Şekil 3.13 ile Şekil 5.1'de gösterilen bağların ilişkisi (Ceylan, 2002).....	120
5.2 : Allsop ve Charlesworth ulaşım ağı B-V talepleri (Ceylan, 2002).....	121
5.3 : Allsop ve Charlesworth ulaşım ağı karakteristik özellikleri (Ceylan, 2002).....	121
5.4 : Bir adımlı benzetim ile PÖTRAD modelinden elde edilen en iyi sinyal parametreleri ve <i>PI</i> değeri	124
5.5 : Bir adımlı benzetim ile PÖTRAD modelinden elde edilen bağ hacimleri (ta/sa).....	124
5.6 : Senaryo 1'de elde edilen en iyi sinyal parametreleri ve <i>PI</i> değeri.....	127
5.7 : Senaryo 1'de elde edilen dinamik denge bağ akımları (ta/sa).....	127
5.8 : Senaryo 2'de elde edilen en iyi sinyal parametreleri ve <i>PI</i> değeri.....	128
5.9 : Senaryo 2'de elde edilen dinamik denge bağ akımları (ta/sa).....	129
5.10 : Senaryo 3'de elde edilen en iyi sinyal parametreleri ve <i>PI</i> değeri.....	130
5.11 : Senaryo 3'de elde edilen dinamik denge bağ akımları (ta/sa).....	130
5.12 : Senaryo 4'de elde edilen en iyi sinyal parametreleri ve <i>PI</i> değeri.....	132
5.13 : Senaryo 4'de elde edilen dinamik denge bağ akımları (ta/sa).....	132

5.14 : Allsop ve Charlesworth ulaşım ağında PÖTRAD modelinin uygulanması sonucu elde edilen istatistikler.	134
5.15 : AB emisyon standartları (gr/ta-km) (Dieselnet.com, 2010).	137
5.16 : Taşıtların model yıllarına göre dağılımı.	137
5.17 : Allsop ve Charlesworth ulaşım ağında açığa çıkan taşıt emisyon değerleri.....	138

ŞEKİL LİSTESİ

Şekiller

1.1 : Trafik atama ve sinyal kontrolü arasındaki ilişki.	4
1.2 : Çalışmanın ana yapısı.....	8
2.1 : DTA'nın genel algoritma yapısı (TRB, 2010).....	27
2.2 : Taşıt kümeleri, sinyal ofsetleri ile bağ üzerindeki trafiğin davranışı (Ceylan, 2002).....	49
2.3 : PÖTRAD modeli genel çerçevesi.....	59
3.1 : PÖ'de ajan-çevre etkileşimi (Sutton ve Barto, 1998).....	63
3.2 : İPÖ algoritmasında orijinal ve alt çevrenin gösterimi.....	67
3.3 : İPÖ algoritması.....	69
3.4 : [-10,10] aralığında amaç fonksiyonunun grafiği.	70
3.5 : İPÖ ve PÖ algoritmalarının yakınsama davranışları.	70
3.6 : Farklı değişken sayıları için ortalama amaç fonksiyonu değeri.	74
3.7 : Farklı değişken sayıları için ortalama fonksiyon değerlendirme sayısının değişimi.....	77
3.8 : PÖ yönteminin temel çerçevesinin sinyal kontrol problemine uyarlanması.	78
3.9 : PÖTRANS modelinin genel çerçevesi.	80
3.10 : Sinyal kontrol problemine uyarlanmış İPÖ algoritması.....	80
3.11a : Örnek ulaşım ağı (Ceylan, 2002).....	84
3.11b : Örnek ulaşım ağı kavşak fazları (Ceylan, 2002).	84
3.12 : Örnek ulaşım ağına PÖTRANS modelinin yakınsaması.....	86
3.13 : Allsop ve Charlesworth ulaşım ağı (Ceylan, 2002).....	88
3.14 : Allsop ve Charlesworth ulaşım ağı kavşak fazları (Ceylan, 2002)	89
3.15 : Allsop ve Charlesworth ulaşım ağına PÖTRANS modelinin yakınsaması.	90
4.1 : Dynasmart-P (Bir adımlı benzetim-atama prosedürü) (Dynasmart-P, 2007).....	96
4.2 : Dynasmart-P için çözüm algoritması (İteratif KD Prosedürü) (Dynasmart-P, 2007).....	99
4.3 : Karşı karşıya kalınan seyahat süresi ile anlık seyahat süresinin belirlenmesi (TRB, 2010).....	101
4.4 : Zamana bağlı değişen bağ seyahat süreleri ile örnek ulaşım ağı (TRB, 2010).	103
4.5 : Birinci dakikada harekete geçiş için en kısa rotalar (TRB, 2010).....	104
4.6 : İkinci dakikada harekete geçiş için en kısa rotalar (TRB, 2010).....	105
4.7 : PÖTRAD modeli genel çerçevesi.....	107
4.8a : Örnek ulaşım ağı (Ceylan, 2002).....	110
4.8b : Örnek ulaşım ağı kavşak fazları (Ceylan, 2002).....	110
4.9 : Örnek ulaşım ağının trafik atama için düzenlenmiş hali (Ceylan, 2002)..	111
4.10 : Örnek ulaşım ağı için ulaşım talebinin ağa yükleme profili 1.....	112
4.11 : Örnek ulaşım ağı için ulaşım talebinin ağa yükleme profili 2.....	112

4.12 : Örnek ulaşım ağında PÖTRAD modelinin yakınsaması (yükleme profili 1).....	113
4.13 : Örnek ulaşım ağında PÖTRAD modelinin yakınsaması (yükleme profili 2).....	114
5.1 : Allsop ve Charlesworth ulaşım ağının TA için düzenlenmiş hali (Ceylan, 2002).....	122
5.2 : Bir adımlı benzetim ile PÖTRAD modelinin yakınsaması	124
5.3 : Senaryo 1’de ulaşım talebinin ağa yüklem profili.	126
5.4 : Allsop ve Charlesworth ulaşım ağında senaryo 1’e göre PÖTRAD modelinin yakınsaması.....	126
5.5 : Senaryo 2’de ulaşım talebinin ağa yüklem profili.	127
5.6 : Allsop ve Charlesworth ulaşım ağında senaryo 2’ye göre PÖTRAD modelinin yakınsaması.....	128
5.7 : Senaryo 3’de ulaşım talebinin ağa yüklem profili.	129
5.8 : Allsop ve Charlesworth ulaşım ağında senaryo 3’e göre PÖTRAD modelinin yakınsaması.....	130
5.9 : Senaryo 4’de ulaşım talebinin ağa yüklem profili.	131
5.10 : Allsop ve Charlesworth ulaşım ağında senaryo 4’e göre PÖTRAD modelinin yakınsaması.....	132
5.11 : Her senaryoda elde edilen ağ <i>PI</i> değerlerinin değişimi.	133
5.12 : Her senaryoda elde edilen devre süresinin değişimi.....	134

SEMBOL LİSTESİ

- A** kısıtlar kümesi
 $A(q)$ q düğüm noktasından çıkan bağların kümesi
 a_t t zamanında eylemler kümesi içinden seçilip uygulanan eylem
 $B(q)$ q düğüm noktasına giren bağların kümesi
 c ortak devre süresi
 c_w^r herhangi bir w B-V çifti arasındaki r rotasındaki seyahat maliyeti
 c_s Bağ seyahat maliyet fonksiyonu
 C şerit grubu kapasitesi
 d_{ij} i zaman diliminde j bağına giren taşıt sayısını
 d^{ta} t zaman aralığında a bağına giren toplam taşıt sayısı
 d_{ijk}^{na} i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atanarak giden, t zaman aralığında a bağına giren taşıtların sayısı
 d_a a bağı üzerindeki gecikme değeri
 d taşıt başına kontrol gecikmesi
 d_1 üniform kontrol gecikmesi
 d_2 rastgele varışların ve doygun üstü akımdaki kuyrukların etkisini açıklamak için eklenik gecikme değeri
 d_3 analiz süresi başlangıcındaki başlangıç kuyruğundan dolayı tüm taşıtlara etkiyen gecikme değeri
 E_j j emisyonunun değeri
 $e_{p,i}$ i 'nci dilimdeki model yıllık taşıtlar için benzinli motor emisyon değerleri
 $e_{d,i}$ i 'nci dilimdeki model yıllık taşıtlar için dizel motor emisyon değerleri
 F düzeltme faktörü
 f_{PA} yeşil süre sırasında gelen taşıtlar için tamamlayıcı düzeltme katsayısı
 $F_i(q)$ aynı zaman diliminde bağına giren fakat çıkamayan trafik miktarı
 g_w w B-V çifti arasındaki seyahat talebi
 g_j her bir bağına ait çıkış fonksiyonu
 h_w^r herhangi bir w B-V çifti arasındaki r rotasındaki akım
h rota akımları vektörü
 h_{ij} her bir bağına ait maliyet fonksiyonu
 h saat başına rasgele ve doygunüstü duruş sayısı
 I_n^t t zaman aralığında n düğüm noktasında üretilen taşıt sayısı
 I yeşillerarası süre
 K duruş ceza faktörü
 L bağ sayısı

m_{ijk}^{aa}	i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atılarak giden, t zaman aralığında a bağından çıkan taşıtların sayısı
m^{ta}	t zaman aralığında a bağından çıkan toplam taşıt sayısı
M	taşıt sayısı cinsinden kuyruk uzunluğu
m	çevrenin boyutu
N_o	rasgele ve doygünüstü kuyruk uzunluğu
n	karar değişkenlerinin sayısı
O_n^t	t zaman aralığında n düğüm noktası üzerinden ağdan çıkan taşıt sayısı
P	yeşilde geçen taşıtların oranı
PF	sinyal koordine düzeltme faktörü
\mathbf{q}	bağ akımları vektörü
q_a	bağ akım değeri
Q_b	analiz süresi başlangıcında kuyrukta olan taşıt sayısı
$Q(s, a)$	(s, a) durum eylem çifti için Q değeri
R_w	w B-V çifti arasındaki rotalar kümesi
r_{ij}^τ	τ harekete geçiş zaman aralığında i başlangıç noktasından j varış noktasına hareket eden taşıt sayısı
r_{ijk}^τ	τ harekete geçiş zaman aralığında i başlangıç noktasından j varış noktasına k rotası üzerine atılarak hareket eden taşıt sayısı
$r(s, a)$	s durumunda a eylemi tamamlandığında alınan ödül değeri
r_{ih}^τ	τ harekete geçiş zaman aralığı boyunca i başlangıç düğümünden başlayan h seyahat yapısına sahip seyahatlerin sayısı
r_{ijk}^τ	i başlangıç düğüm noktasından j varış düğüm noktasına hareket eden k rotasına τ harekete geçiş zaman aralığında atılarak seyahat edenlerin sayısı
S_a	bir saniyede a bağı üzerindeki taşıt-duruş sayısı
s_t	t zamanında çevrenin durumu
s	doygun akım değeri
t	analiz süresi içinde karşılanamamış talebin süresi
T'	atamaların yapıldığı zaman aralığı
T_{ijk}^τ	i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atılarak giden taşıtlar için karşı karşıya kalınan rota seyahat süresi
T	analiz periyodu süresi
t_{maks}	maksimum öğrenme evresi sayısı
T^{ta}	t zaman periyodunun başlangıcında a bağı üzerindeki seyahat süresi
TD	ulaşım talebi
$TP_{p,i}$	i 'nci dilimdeki model yıllık benzinli taşıtların ulaşım talebi içindeki yüzde cinsinden payı
$TP_{d,i}$	i 'nci dilimdeki model yıllık dizel taşıtların ulaşım talebi içindeki yüzde cinsinden payı
t_a^0	serbest akım seyahat süresi

u	gecikme parametresi
$VP_{p,i}$	i 'nci dilimdeki model yıllık benzinli taşıtların ortalama yüzdesi
$VP_{d,i}$	i 'nci dilimdeki model yıllık dizel taşıtların ortalama yüzdesi
v_s	s bağı üzerindeki akım
\mathbf{v}	bağ akımları vektörü
$v'_{(k+t)}$	k zaman aralığında bağına akım aşağısına gelen akım oranı
v_k	k zaman aralığında gelen (IN) akım yapısı
W	B-V çiftleri kümesi
w_{d_a}	gecikme değeri ağırlık katsayısı
w_{s_a}	duruş sayısı ağırlık katsayısı
$x_{ijk}^{\pi a}$	i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atanarak giden, t zaman aralığının başında a bağına üzerinde olan taşıtların sayısı
x^{ta}	t zaman aralığının başlangıcında a bağı üzerindeki toplam taşıt sayısı
X	doygunluk derecesi
X_u	akım yukarı kavşaktaki doygunluk derecesi
x^{ta}	t zaman periyodunun başlangıcında a bağı üzerinde seyahat edenlerin toplam sayısı
y_{ijk}^{τ}	i başlangıç düğüm noktasından j varış düğüm noktasına hareket eden k rotasına τ harekete geçiş zaman aralığında hep yada hiç atama algoritmasına göre atanarak seyahat edenlerin sayısı
z	yol ağındaki her bir sinyalizasyon kavşağına ait faz sayısı
τ	seyahat eden taşıtların harekete geçtiği zaman aralığını belirten altsimges, $\tau = 1, \dots, T1$
δ_w^{sr}	link-rota belirleme matrisi elemanı
$\delta_{ijk}^{\pi a}$	zamana bağlı bağ-rota belirleme elemanı
γ	gelecekteki ödüllere atanan ağırlığı temsil eden azaltma faktörü
α	algoritma öğrenme oranı
β	çözüm uzayını sınırlandırma vektörü
ψ	sinyal parametreleri vektörü
Ω_0	sinyal süreleri için uygun çözüm kümesi
$\theta_{ij}^{*\tau}$	τ zaman aralığında i başlangıç noktasından j varış noktasına en küçük karşı karşıya kalınan seyahat süresi
Δ	zaman aralığının uzunluğu
θ	ofset süresi
θ	ofset süresi vektörü
φ	yeşil süre
Φ	yeşil süre vektörü

ÖZET

İYİLEŞTİRİLMİŞ PEKİŞTİRMELİ ÖĞRENME YÖNTEMİ VE DİNAMİK YÜKLEME İLE KENTİÇİ ULAŞIM AĞLARININ TASARIMI

Bu çalışmada iyileştirilmiş pekiştirmeli öğrenme yöntemi ile dinamik kullanıcı dengesi altında ulaşım ağ tasarımı probleminin çözümü amaçlanmıştır. Problemin çözümü için iki seviyeli programlama tekniği kullanılmıştır. İki seviyeli programlama içinde alt seviye olan trafik atama probleminin çözümü için benzetim bazlı dinamik trafik atama modeli olan Dynasmart-P yazılımı kullanılmıştır. Üst seviye olarak adlandırılan sinyal kontrol aşamasında ise iyileştirilmiş pekiştirmeli öğrenme yöntemi ile atama sonunda elde edilen bağ akımlarına göre en uygun sinyal kontrol parametreleri bulunmuştur. Sinyal kontrol parametreleri, ortak devre süresi, kavşaklar arası ofset süreleri ve faz yeşil süreleri olarak tanımlanmıştır. Ulaşım ağı üzerindeki sinyal kontrol parametrelerine bağlı olarak trafik akımlarının birim zamanda duruş sayılarının ve gecikmelerinin ağırlıklandırılmış toplamı olarak ifade edilen ağ performans indeksi, TRANSYT-7F yazılımının trafik modeli ile değerlendirilmiştir.

İyileştirilmiş pekiştirmeli öğrenme yönteminin, trafik atama ve sinyal kontrol problemine entegre edilmesiyle oluşturulan iyileştirilmiş **Pekiştirmeli Öğrenme TRANSYT-7F Dynasmart-P (PÖTRAD)** modeli dinamik kullanıcı dengesi altında ulaşım ağ tasarımı probleminin çözümü için önerilmiştir. Geliştirilen PÖTRAD modelinin amaç fonksiyonu ağ performans indeksidir. Geliştirilen PÖTRAD modeli örnek ulaşım ağları üzerinde test edilmiştir. Hem küçük ölçekli hem de Allsop & Charlesworth örnek ulaşım ağında olumlu sonuçlar elde edilmiştir. PÖTRAD modelinin sayısal uygulaması için çeşitli senaryolar oluşturulmuştur. PÖTRAD modeli her senaryoda amaç fonksiyonu olan ağ performans indeksi değerini enküçükleyebilmiş ve her senaryo için oldukça kararlı bir yakınsama eğilimi göstermiştir. Elde edilen sonuçlar, önerilen PÖTRAD modelinin etkin şekilde sinyal kontrol parametrelerini ve ağ performans indeksini eniyilebildiğini göstermiştir. Bu nedenlerden dolayı geliştirilen PÖTRAD modelinin dinamik kullanıcı dengesi altında ulaşım ağ tasarımı probleminin çözümünde kullanılabileceği sonucuna varılmıştır.

Anahtar Kelimeler: Ulaşım Ağ Tasarımı, Pekiştirmeli Öğrenme Yöntemi, Dinamik Trafik Atama, Alansal Trafik Kontrolü

ABSTRACT

DYNAMIC USER EQUILIBRIUM URBAN NETWORK DESIGN BASED ON MODIFIED REINFORCEMENT LEARNING METHOD

This study aims to solve dynamic user equilibrium transportation networks using modified reinforcement learning approach. The bi-level programming technique is used to solve the problem. At the lower level of the problem, the equilibrium link flows are obtained by simulation based dynamic traffic assignment with Dynasmart-P and signal timings are obtained at the upper level by modified reinforcement learning method. Signal timings are defined by the common network cycle time, the green time for each signal stage, and the offsets between the junctions. The system performance index is defined as the sum of a weighted linear combination of delay and number of stops per unit time for all traffic streams, which is evaluated by the traffic model of TRANSYT-7F.

By integrating the modified reinforcement learning method, traffic assignment and traffic control, the modified **Reinforcement Learning TRANSYT-7F Dynasmart-P (POTRAD)** model is proposed to solve the dynamic equilibrium road network design problem. The objective function of the proposed POTRAD model is network performance index. The proposed POTRAD model is tested on the example road networks. The encouraging results are obtained for both small sized and Allsop & Charlesworth's network examples. Various scenarios are proposed for numerical applications of the POTRAD model. The proposed POTRAD model minimized the network performance index that is the objective function at each scenario and showed steady convergence for each scenario. Results showed that the proposed POTRAD model effectively optimizes the signal timings and values of the network performance index. Because of these reasons, it is obtained that the POTRAD model can effectively be used to solve network design problem under dynamic user equilibrium conditions.

Key Words: Network Design Problem, Reinforcement Learning Method, Dynamic Traffic Assignment, Area Traffic Control

1. GİRİŞ

1.1. Ulaşım Ağ Tasarımı

Ulaşım talebi, motorlu taşıt kullanımının ve hareketliliğin artışına bağlı olarak ekonomi ve teknolojinin gelişimi ile hızla artmaktadır. Artan ulaşım talebi, ulaşım yapılarının ve ulaşım sistemlerinin hizmet kapasitelerinin düşmesi ile başlayan problemleri ortaya çıkarmaktadır. Bu problemlere örnek olarak; ulaşım ağlarındaki tıkanıklıkların artması, kapasitelerin düşmesi, seyahat sürelerinin artması, taşıtların yakıt tüketimlerinin artması, artan egsoz emisyonları ile hava kirliliğinin artması, sürücülerin psikolojilerinin bozulması verilebilmektedir. Bu problemlerden, seyahat süresinin artması, kapasitenin düşmesi, enerji kaybı, yakıt tüketiminin artması gibi problemlerin etkileri kolaylıkla ölçülebilirken, sürücü psikolojilerinin bozulması gibi problemlerin etkileri zor ölçülebilmektedir.

Ulaşım ağlarında, karar vericiler ağın performansından en iyi şekilde yararlanılması için politikalar üretmeye çabalarken ağ kullanıcıları ise kendi bireysel seyahat maliyetlerini iyileştirecek şekilde hareket ederler. Bu yüzden, ulaşım ağlarında karar vericiler ile kullanıcılar sürekli etkileşim halindedirler ve ulaşım ağı tasarlanırken bu etkileşimin bir başka deyişle karar vericiler ile kullanıcıların davranışlarının karşılıklı olarak dikkate alınması oldukça önemlidir.

Ulaşım Ağ Tasarımı (UAT) problemi, ağ kullanıcıları ile karar vericilerin kararlarının karşılıklı olarak etkileşim içinde olduğu kabulü yapılarak ulaşım ağının tasarlanması olarak ifade edilebilmektedir. Ağ tasarımı sonucunda en uygun tasarım parametrelerinin bulunması ile kullanıcıların seyahatleri sırasında ağda kaybettiği zaman en az seviyelere çekilebilmektedir (Başkan, 2009).

Ulaşım ağı, kavşak olarak adlandırılan düğüm noktalarından ve bu düğüm noktalarının birbiri ile ilişkisini sağlayan bağlardan meydana gelmektedir. Kavşaklar, trafik akımlarının kesiştiği kısımlar olup ikiye ayrılmaktadır: Sinyalize ve sinyalize olmayan kavşaklar. Sinyalize kavşaklar, trafik kontrolünü sağlayarak taşıtlara geçiş önceliği vermektedir. Sinyalize bir kavşağın kapasitesi genellikle sadece trafik işaretleri ile kontrol edilen aynı geometrideki sinyalize olmayan kavşağın kapasitesinden daha yüksektir.

Sinyalizasyon sistemleri kontrol ettikleri kavşakların durumuna göre ayrık ve eşzamanlı sistemler olmak üzere ikiye ayrılmaktadır. Ayrık sinyalizasyon sistemleri, diğer kavşaklardaki sinyalizasyon sistemleri ile herhangi bir bağlantısı olmayan ve diğer sinyalize sistemlerin etkilemediği sistemlerdir. Ayrık sinyalizasyon sistemlerinden olan sabit zamanlı sinyalizasyon sistemleri özellikle gelişmekte olan ülkelerin çoğunda kullanılmaktadır. Bu tür sistemlerde, kavşaklarla ilgili trafik sayım ve modelleme çalışmalarının sürekli yapılması gerekmektedir. Trafik sayım ve modelleme çalışmalarının sürekli olarak yapılmaması bu tür sistemlerin hatalı olarak işletilmesine neden olmaktadır. Bu durum, kavşak kolları üzerindeki farklı trafik yüklerini dikkate almadan aynı tasarım parametrelerinin uygulanması sonucunu doğurmakta bu da gecikmelerin ve buna bağlı olarak trafikteki sıkışıkların artmasına neden olmaktadır. Ana yolların birbirine çok yakın iki veya daha fazla sayıdaki kavşaklarda, gecikmeleri azaltmak ve sık sık duruş-kalkışları engellemek amacıyla, kavşaklardaki sinyalizasyon sistemlerinin birbirine bağlanması eş zamanlı sinyalizasyon sistemi olarak tanımlanmaktadır. Eşzamanlı sistemler, genellikle anayol üzerindeki kavşaklardan, tali yol trafiğine de yeterli geçiş hakkı tanıyarak, birim zaman içinde mümkün olan en yüksek sayıda taşıtın durmadan geçirilmesi için düzenlenmektedir. Eşzamanlı sinyalizasyon sistemleri öncelikle anayol trafiği için uygulanmakla birlikte, bütün yönlerdeki toplam gecikmenin enküçüklenmesi için de uygulanabilmektedir. Bu şekilde sinyalizasyon sistemlerinin tümü arasında koordinasyonu sağlamak üzere en uygun sinyal parametrelerinin verilmesini düzenleyen eşzamanlı sinyalizasyon sistemlerine Alansal Trafik Kontrolü (ATK) adı verilmektedir (Murat, 1996).

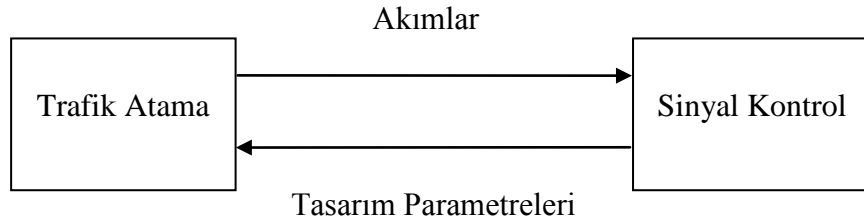
Başlangıç-Varış (B-V) noktaları arasındaki talebin ulaşım ağına yüklenmesi ve bağ akımlarının bulunması Trafik Atama (TA) problemi olarak adlandırılmaktadır.

Atama problemi sırasında çözümünü yapılacak ulaşım ağının özelliklerini en iyi şekilde yansıtan bağ maliyet fonksiyonlarının seçimi ve UAT probleminin çözümünde kullanılacak olan amaç fonksiyonu için gerekli yakınsamanın sağlanabilmesi çok önemlidir (Başkan, 2009).

Şehir içi trafik yönetiminde ulaşım ağının tasarımı, en uygun sinyal parametrelerinin belirlenmesi ve sinyalize kavşaklardaki gecikmelerin düşürülerek trafik sıkışıklığının azaltılması oldukça önemlidir. UAT probleminin çözümü, konveks olmayan yapısından dolayı oldukça zor ve karmaşıktır. UAT probleminin çözümü için literatürde farklı birçok yöntem bulunmaktadır. İki seviyeli programlama son yıllarda literatürde oldukça fazla yer bulan yöntemlerden biridir. Bu yöntemde sinyal kontrolü ve TA problemlerinin çözümü iki seviyeli olarak çözülmektedir. İki problemin çözümü sırasında aslında girdi olarak kullanılan parametreler diğer problemin çözümünden elde edilen çıktı değerleridir. Diğer bir çözüm yöntemi ise Karşılıklı İteratif Yaklaşımdır (KİY). Bu yöntemde sinyal kontrol ve TA problemleri ayrı ayrı çözümlenerek problem çıktıları diğer problemin girdisi olarak kullanılmaktadır.

Bu çalışmada UAT çözümü için son yıllarda sıkça kullanılan iki seviyeli programlama tekniği tercih edilmiştir. İki seviyeli programlama içinde üst seviye sinyal kontrol problemini temsil ederken alt seviye ise TA problemini temsil etmektedir. Üst seviyede sinyal süreleri bulunurken, alt seviyede üst seviyede bulunan sinyal süreleri ile TA problemi çözülerek bağ akımları bulunmaktadır. Daha sonra, bulunan bağ akımları üst seviye için girdi olarak kullanılmakta ve bu bağ akımlarına göre üst seviye olan sinyal kontrol problemi çözülerek sinyal süreleri hesaplanmaktadır. UAT probleminde, ağ kullanıcıları ile karar vericiler karşılıklı etkileşim içinde oldukları için, karar vericiler ulaşım ağı üzerindeki sinyal kontrolü ile ilgili bir karar aldığında bu karar seyahat maliyetlerini ya da sürelerini en küçükleme amacı içinde olan ağ kullanıcılarının davranışlarını etkileyerek rota seçimlerinin değişmesine ve bu da ağ üzerindeki bağ hacimlerinde farklılıklara neden olmaktadır. Sinyal kontrolü ile TA problemi karşılıklı olarak birbirlerini etkilemesine rağmen, bu iki problem genellikle ayrı ayrı çözülmektedir. Mevcut ulaşım ağının performansını artırma ihtiyacı rota seçimi ile sinyal kontrolü arasındaki etkileşimi önemli hale getirmektedir. Sinyalizasyonla kontrol edilen ulaşım ağlarında sinyal tasarım parametreleri ile bireysel yol kullanıcılarının seçtiği rotalar arasında bir

etkileşim mevcuttur. Trafik mühendisliği açısından, ulaşım ağı üzerindeki bağların trafik hacimleri genellikle sabit kabul edilmekte ve buna göre ulaşım ağının performansını artırmak için sinyal tasarım parametreleri eniyilenmektedir. Ulaşım planlaması açısından ise, sinyal tasarım parametreleri ile belirlenen bağ kapasitelerinin sabit olduğu kabulü yapılarak TA modelleri ile bağların hacimleri belirlenmektedir. Sinyal kontrol problemi ile TA problemi arasındaki karşılıklı etkileşim, birleşik kontrol atama problemi olarak adlandırılmaktadır. Birleşik kontrol atama problemi Allsop (1974) tarafından şöyle açıklanmıştır: “ulaşım ağı üzerindeki trafik sinyalizasyon sistemleri ile kontrol edilirken, ağ üzerindeki bağların trafik hacimleri ile bağların seyahat maliyeti arasındaki ilişki sinyal tasarım parametrelerine bağlıdır. Bu durum, ağ üzerinde yapılan seyahat sayısını ve seçilen rotaları etkilemektedir”. Sinyal kontrolü ile TA arasında karşılıklı etkileşim Şekil 1.1’de görülebilmektedir.



Şekil 1.1 Trafik atama ve sinyal kontrolü arasındaki ilişki

Ulaşım ağını kullanan yol kullanıcılarının oluşturduğu ulaşım talebinin yönetilmesi günümüzün en önemli sorunlarından birisidir. Hızla artan motorlu taşıt kullanımı da ulaşımında bir karmaşıklığa neden olmaktadır. Şehir içi ulaşım ağlarında seyahat edenlerin yarattığı talep artışının yönetilmeden karşılanması mümkün görünmemektedir. Artan talep, belli bir süre sonra ulaşım ağlarında trafik sıkışıklığına neden olmakta ve bu sıkışıklık toplum üzerinde çeşitli sosyolojik ve psikolojik problemler yaratmaktadır. Meydana gelen trafik sıkışıklığı toplumu oluşturan bireylerin günlük aktivitelerinin yapılarını değiştirmekte, özel motorlu taşıtlara bağımlılığı artırmakta ve çevresel sorunları içinden çıkılmaz hale getirmektedir. Artan talebe karşılık ağ kapasitelerinin artırılması yerine ulaşım talebinin yönetilerek kontrol altına alınması çok önem taşımaktadır.

1.2. Problemin Tanımı

UAT verilen tasarım amacına göre en uygun performansı sağlayacak şekilde ulaşım ağının işletilmesi olarak tanımlanmaktadır (Başkan, 2009). UAT, özellikle her geçen gün artan araç kullanımının beraberinde getirdiği sorunların çözümü ve kısıtlı kaynaklarla sürdürülebilir ağ kapasitesi yaratabilmek açısından çok önemli bir konudur. UAT'de öncelikle B-V noktaları arasındaki talebin ulaşım ağına yüklenerek bu ulaşım talebinin bağ maliyet fonksiyonuna göre dağıtılması ve güzergah akımlarının bulunması gerekmektedir. TA olarak adlandırılan bu problem çözüldükten sonra en uygun sinyal sürelerinin bulunması gerekmektedir. Güzergah akımlarının bulunmasında önerilen yöntem, rota üzerinde seyahat eden her bir ağ kullanıcısının kendi seyahat maliyetini veya kendi seyahat süresini en aza indirmeye yaklaşımıdır. Bu yaklaşım denge durumunda bütün kullanılan rotalardaki seyahat maliyetinin aynı olacağı ve bu maliyetin kullanılmayan rotalardaki seyahat maliyetinden daha az veya eşit olacağı kabulüne dayanmaktadır. Bu durum ağ kullanıcı dengesi altındadır şeklinde tanımlanmakta ve hiçbir kullanıcı kendi rotasını değiştirerek seyahat maliyetini değiştirememektedir (Wardrop, 1952). Bu yaklaşım deterministik Kullanıcı Dengesi (KD) olarak adlandırılmaktadır. Fakat gerçekte deterministik KD durumundan farklı olarak ağ kullanıcıları rotalar hakkında tam olarak bilgi sahibi değildirler ve ağ kullanıcılarının rota seçimleri sırasında algılama hataları meydana gelmektedir. Bu durum Stokastik KD durumunu doğurmaktadır. Stokastik KD'de bütün ağ kullanıcıları için tek bir maliyet tanımlanması yerine her bir kullanıcı için ayrı ayrı seyahat maliyeti tanımlandığı kabulü yapılmakta ve seyahat maliyetinin elde edilmesinde sürücü davranışlarındaki algılama hataları göz önüne alınmaktadır. Stokastik atama yöntemlerinde olasılık seçim modelleri kullanılarak B-V talebi rotalara atanmakta, bu sırada en küçük maliyetli rotaların denge noktasına kadar en fazla akımı çektiği kabul edilmektedir.

TA modelleri statik ve dinamik TA modelleri olmak üzere ikiye ayrılmaktadır. Statik TA modelleri analiz süresi boyunca ortalama ve sabit koşullar üzerine odaklanmıştır. Bir başka deyişle, analiz süresi boyunca rota ve bağların seyahat süreleri ve hacimleri sabit olarak düşünülmüş, zamanla bu değerlerde bir değişiklik olabileceği dikkate alınmamıştır. Oysa, ağ üzerindeki bağların kapasiteleri ağ üzerindeki trafik sıklığından, trafik kontrol sistemlerinden veya yol üzerindeki çalışma

alanlarından dolayı gün içinde deęişebilmektedir. Ayrıca, modern ulaşım talep modelleri ile gün içinde bir saatten daha az zaman dilimleri için B-V matrisleri elde edilebilmektedir. Zamana baęlı olarak deęişen ulaşım talebini ve ulaşım aęının zamana baęlı durumu dikkate alarak yapılan TA, Dinamik TA (DTA) olarak adlandırılmaktadır. Statik atamaya benzer şekilde dinamik atama, seyahat talebi sonucunda aęı oluřturan baęlardaki trafik hacimlerini belirlemek için uygulanmaktadır, fakat burada zamana baęlı seyahat talebi ve zamana baęlı ulaşım aęı durumu söz konusu olmaktadır. Zamanın da dahil olması problemin karmařıklıęını artırmaktadır.

UAT sürecinde ulaşım aęı karar vericileri ile ulaşım aęının kullanıcıları olmak üzere iki grubun birbiri ile etkileřimi söz konusu olmaktadır. Ulaşım aęı karar vericileri toplam sistem maliyetini enküçüklemeyi veya sosyal refahın ve konforun artırılmasını amaçlamaktadırlar. Herhangi bir ulaşım aęı kullanıcısı ise ulaşım aęı üzerindeki konfor veya sosyal refahı dikkate almayarak sadece kendi seyahat maliyetini veya süresini enküçüklemeyi amaçlamaktadır. Bu sebeple ulaşım aęı karar vericileri sistem maliyetini enküçükleyebilmek veya sosyal refahı ve konforu artırabilmek için çabalarırken aę kullanıcılarının davranıřlarını dikkate alması gerekmektedir. Dolayısıyla ulaşım aęında yapılan deęiřiklikler aę kullanıcılarının davranıřlarını etkilemektedir. Ulaşım aęı karar vericileri ve aę kullanıcıları arasındaki karřılıklı bu iliřkiden dolayı UAT problemi iki seviyedeki karar vericilerin oluřturduęu hiyerarřik bir problem olarak ifade edilmektedir (Bařkan, 2009).

UAT probleminin çözümleri matematiksel yöntemler ile yapılabilirken özellikle son yıllarda farklı yaklařımlar ile problemin çözümleri üzerine arařtırmalar yapılmaktadır. Ayrıca UAT probleminin çözümünde tasarım parametrelerinin çeřitlilięi sebebiyle yeni yaklařımların problemin çözümünde ne gibi sonuçlar verebileceęi üzerine çalıřmalar yapılması gerekmektedir. Literatürde UAT probleminin çözümleri için farklı yöntemler önerilirken özellikle son yıllarda Genetik Algoritma (GA), Tavlama Benzetimi (TB), Tabu ARama (TAR), Karınca Sistemi (KS), Karınca Kolonisi Optimizasyonu (KKO) gibi sezgisel yöntemlerin kullanımı oldukça artmaktadır. Yapay zeka yöntemlerinden olan Pekiřtirmeli Öęrenme (PÖ) yöntemi ile UAT literatürde mevcut deęildir. Ayrıca, UAT probleminin konveks olmayan yapısından

ve tasarım parametresi sayısının fazla olmasından dolayı çalışmada UAT çözümü için iyileştirilmiş PÖ yöntemi kullanılmıştır.

Bu çalışmada, dinamik olarak UAT probleminin çözümü yapılmıştır. Bu amaçla, iki seviyeli programlama tekniğinden faydalanılmıştır. Alt seviye olan TA probleminin çözümünde benzetim bazlı DTA modeli kullanılmış, üst seviye olan sinyal kontrol probleminde ise iyileştirilmiş PÖ ile TRANSYT-7F trafik modeli kullanılarak alt seviye olan TA sonunda elde edilen denge bağ hacimlerine bağlı olarak sinyal tasarım parametreleri hesaplanmıştır.

1.3. Amaç

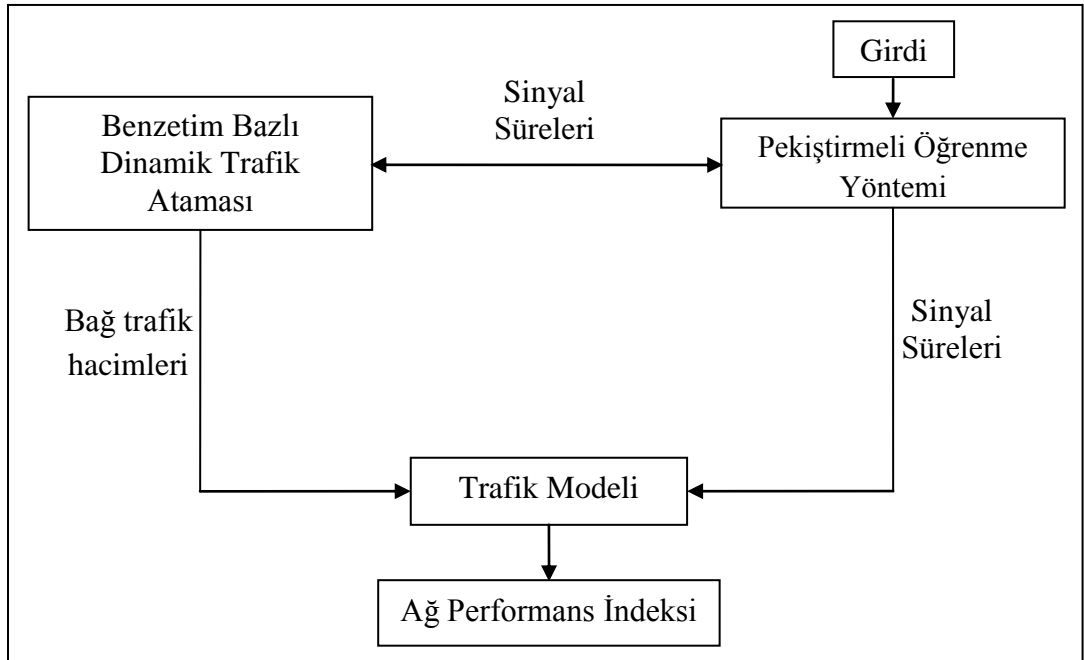
Çalışmanın temel amacı, kentiçi yol ağlarında zamana ait dinamikleri dikkate alarak sürekli ağ tasarımını gerçekleştirebilen model geliştirmektir. UAT ile, ağ kapasitesinin etkin, verimli kullanımı ve işletim koşullarının iyileştirilmesi probleminin verilen belirli bir bütçeyle nasıl çözülebileceği sorusu cevaplanmaya çalışılmaktadır.

UAT probleminin çözümü, konveks olmayan yapısından dolayı zor ve karmaşıktır. Bu yüzden, UAT probleminin çözümünde sezgisel yöntemlere ihtiyaç duyulmakta ve son yıllarda bu yöntemler sıkça kullanılmaktadır. Çalışmada, çeşitli eniyileme problemlerinin çözümü için son yıllarda gittikçe artarak kullanılan PÖ yönteminden faydalanılacaktır.

UAT problemi, sinyal kontrol ve TA problemlerinin bir arada çözülmesi gerekliliği sonucunda birleştirilmiş eniyileme problemi olarak tanımlanmıştır (Allsop 1974, Gartner 1974, Marcotte 1983). UAT probleminin çözümünde kullanılan tekniklerden biri olan iki seviyeli programlama tekniği literatürde sıkça kullanılmaktadır. İki seviyeli programlama içinde alt seviye TA problemini temsil ederken, üst seviye alt seviyenin çözümü ile elde edilen bağ akımlarına bağlı olarak ağ tasarım parametrelerinin en uygun değerlerinin belirlendiği bölüm olan sinyal kontrol problemini temsil etmektedir.

Üst seviye olarak adlandırılan sinyal kontrol aşamasında ise iyileştirilmiş PÖ yöntemi ile TRANSYT-7F yazılımının trafik modeli kullanılarak atama sonunda elde edilen bağ akımlarına göre sinyal süreleri eniyilenerek ulaşım ağından en iyi

performans elde edilmeye çalışılacaktır. Alt seviyede ise, benzetim bazlı DTA modeli olan Dynasmart-P yazılımı kullanılarak Dinamik Kullanıcı Dengesi (DKD) altında bağ akımları belirlenecektir. Bu amaçla, iyileştirilmiş Pekiştirmeli Öğrenme **TRANSYT-7F Dynasmart-P (PÖTRAD) UAT** modeli geliştirilmiştir. Kısaca belirtmek gerekirse bu çalışmada PÖTRAD modeli kullanılarak DKD altında ulaşım ağ tasarımı yapılacaktır. Geliştirilen PÖTRAD modelinde, UAT sürecini oluşturan benzetim bazlı DTA ve sinyal kontrol aşaması PÖ yöntemi ile birleşik olarak kullanılacaktır. PÖTRAD modelinin amacı, Ağ Performans İndeksi olarak adlandırılan ağ performansının eniyilenmesidir. Bir başka deyişle, ulaşım ağından en iyi performans alınacak şekilde DKD bağ hacimlerine göre sinyal kontrol parametrelerinin belirlenmesidir. Çalışma ile, UAT algoritmasının performansının iyileştirilmesi ve daha sonra bu algoritmanın örnek bir ulaşım ağına uygulanması ile ulaşım ağlarının işletilmesinde karşılaşılan sorunların en aza indirgenmesi ve bu sorunlardan dolayı meydana gelen kayıpların enküçüklenmesi gibi kazanımların olacağı düşünülmektedir. Geliştirilen PÖTRAD modelinin gerçek ulaşım ağına uygulanabilmesi ile ulaşım ağı üzerindeki sinyal sürelerinin en uygun değerlerinin bulunabilmesi, böylece gecikmelerin ve beraberinde getirdiği sorunların azaltılabilmesi bu çalışmanın ana amacını oluşturmaktadır. Şekil 1.2’de bu çalışmanın ana yapısı verilmektedir.



Şekil 1.2 Çalışmanın ana yapısı

1.4. Tez Düzeni

Tezin giriş bölümünde UAT problemi tanımlanmış, problem ve çalışmanın amacından bahsedilmiş ayrıca çalışmanın düzeni hakkında bilgi verilmiştir. Bölüm 2’de UAT problemi açıklanarak iki seviyeli programlama yöntemi ve KİY yöntemi hakkında bilgi verilmiştir. TA problemi, statik ve dinamik TA ile ilgili bilgi verilmiştir. KD kavramı açıklanmıştır. DTA ve çözüm yöntemleri ile ilgili bilgi verilerek DTA’da KD yaklaşımının önemli noktalarına değinilmiştir. Daha sonra benzetim bazlı DTA modelleri hakkında detaylı bilgi verilerek tüm DTA yöntemleri ile ilgili yapılan çalışmalardan bahsedilmiştir. Daha sonra, çalışmada kullanılan benzetim bazlı DTA modeli olan Dynasmart-P yazılımından söz edilmiştir. Ayrıca, bu bölümde trafik sinyal kontrolü hakkında bilgi verilerek çalışmada kullanılan TRANSYT-7F yazılımından bahsedilmiştir. Yine aynı bölümde, çalışmada kullanılan yapay zeka yöntemlerinden PÖ yöntemine giriş yapılarak tez çalışmasının genel çerçevesi hakkında bilgi verilmiştir. Bölüm 3’de PÖ yöntemi hakkında detaylı bilgi verilerek çalışmada kullanılan iyileştirilmiş PÖ yönteminin ayrıntılı algoritma adımları ve çalışma prensibinden bahsedilmiştir. İyileştirilmiş PÖ yönteminin test fonksiyonları üzerindeki çözümleri verilmiştir. Ayrıca, UAT probleminin üst seviyesi olan sinyal kontrol probleminin çözümünde kullanılan iyileştirilmiş **Pekiştirmeli Öğrenme TRANSYT-7F (PÖTRANS)** modelinin ayrıntılı çalışma prensibi ve çözüm algoritmasından söz edilerek PÖTRANS modelinin iki farklı ulaşım ağı üzerindeki sayısal uygulaması verilmiştir. Bir sonraki bölüm olan 4. Bölümde, benzetim bazlı DTA modeli olan Dynasmart-P yazılımından bahsedilerek algoritması hakkında bilgi verilmiştir. Benzetim bazlı dinamik TA modellerinde kullanılan zamana bağlı en kısa rota algoritmasından detaylı şekilde bahsedilerek örnekler verilmiştir. Daha sonra, DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modeli hakkında bilgi verilerek çözüm algoritmasından söz edilmiştir. Ayrıca, geliştirilen PÖTRAD modelinin örnek ulaşım ağı üzerindeki uygulaması verilmiştir. Bölüm 5’de DKD altında UAT modeli olan PÖTRAD modelinin sayısal uygulaması yapılmıştır. İlk sayısal uygulamada, 5000 ta/sa’lik ulaşım talebi altında ulaşım ağının performansını ve ağ üzerindeki trafiğin yapısını görebilmek amacıyla PÖTRAD modelinde Dynasmart-P modelinin bir adımlı benzetimi kullanılmıştır. Daha sonra, DKD altında UAT probleminin çözümü için

geliştirilen PÖTRAD modelinin sayısal uygulaması için dört tane senaryo oluşturulmuştur. Oluşturulan senaryolar B-V talebinin altı kavşaklı ulaşım ağı üzerine yüklenmesi ile ilgilidir. Oluşturulan senaryolarla yapılan sayısal uygulamalarda PÖTRAD modelinin altında TA probleminin çözümünde Dynasmart-P yazılımının iteratif benzetim bazlı atama prosedürü kullanılarak DKD altında UAT probleminin çözümü yapılmış, senaryoların sonuçları değerlendirilmiş ve sonuçların karşılaştırılması yapılmıştır. Ayrıca, oluşturulan senaryolara bağlı olarak ulaşım ağı üzerinde taşıtların oluşturacağı egzoz emisyon değerleri belirlenmiştir. Son bölümde ise yapılan çalışmanın genel bir değerlendirmesi yapılmış ve gelecekte yapılması planlanan çalışmalar hakkında bilgi verilmiştir.

2. ÖNCEKİ ÇALIŞMALAR

2.1. Giriş

Ulaşım ağının tasarım amacına göre en uygun performansı sağlayacak şekilde işletilmesi UAT olarak tanımlanmaktadır (Başkan, 2009). Mevcut karayolu ağ kapasitelerinin daha etkin kullanımı ya da bu kapasitelerin artırılması için yapılacak yatırımlardan daha etkili sonuçlar alınabilmesi için belirlenen amaç ya da amaçlar doğrultusunda geliştirilen UAT problemlerinin çözülmesi gerekmektedir. UAT, genellikle Kesikli Ulaşım Ağ Tasarımı (KUAT) ve Sürekli Ulaşım Ağ Tasarımı (SUAT) olmak üzere ikiye ayrılabilir. Her iki ağ tasarım yaklaşımında da temel amaç, yol kullanıcılarının güzergah seçim davranışlarını göz önünde bulundurarak sistem performansını artırmaktır (Bell ve Iida, 1997).

SUAT problemi, ulaşım ağına etki eden sürekli tasarım parametrelerinin en uygun ya da en uyguna yakın değerlerinin bulunması ile ağ performansının iyileştirilmesini ele almaktadır (Abdulaal ve LeBlanc, 1979; Davis, 1994). Diğer bir deyişle, yol ağının fiziksel özellikleri korunarak, ağın işletimi ile ilgili tasarım parametrelerinin eniyilenmesi olarak tanımlanabilmektedir. SUAT problemlerine kavşaklardaki sinyal sürelerinin eniyilenmesi ve sıkışıklık fiyatlandırması örnek olarak verilebilir. KUAT problemi ise, ulaşım ağının fiziksel yapısı ile ilgili olarak ulaşım ağından en uygun performans elde etmek için ağda yapılacak olan fiziksel ve yönetsel değişikliklerin belirlenmesidir. KUAT problemlerine, yol kapama planlarının hazırlanması, ulaşım ağına yeni bağların eklenmesi, ulaşım ağını oluşturan bağların şerit paylaşımlarının düzenlenmesi, güzergahlardaki tek-çift yön uygulamalarının yapılabirliğinin araştırılması örnek olarak verilebilir (LeBlanc, 1975; Poorzahedy ve Turnquist,1982).

SUAT problemi ile KUAT problemi arasındaki temel ayrım bir takım karar değişkenlerindeki farklılıktan kaynaklanmaktadır (Zixiao, 2004).

Trafik kontrol probleminin çözümü ile ilgili literatürde birçok matematiksel model bulunmaktadır. Trafik kontrol probleminin çözümünde Robertson (1969) tarafından önerilen TRANSYT modeli koordineli kavşaklarda en çok kullanılan trafik kontrol modellerinden birisidir. 1970'lerin başında Amerikan Karayolları Birliği tarafından geliştirilen Şehirçi Trafik Kontrol Sistemi (UCTS) ve MAXBAND (Little vd 1981) programları da trafik kontrol modelleri olarak kullanılmıştır. Ayrıca SCATS (Sims 1979, Lowrie 1982, Luk 1984), SCOOT (Hunt vd 1982, Robertson ve Bretherton 1991) ve UTOPIA (Mauro ve Di Taranto, 1989) trafik kontrol modelleri de farklı yaklaşımlarla sinyal kontrol parametrelerinin en iyilenmesine olanak sağlamaktadır.

Allsop (1974) ve Gartner (1974) sinyal kontrolü altında sürücü rota değişimlerini dikkate alan teorik yaklaşımlar geliştirmişlerdir. Seyahat dağılımından ve TA sonucu elde edilen trafik hacimleri trafik kontrol parametrelerinin fonksiyonu olarak nitelendirilmiştir. En uygun trafik sinyal sürelerinin bulunmasında seyahat dağılımı, TA ve trafik sinyal hesaplamaları için hangi metotların uygun olacağı belirlenmiştir. Sayısal bir örnek üzerinde uygulamalar yapılmış ve sonuçlar başarılı bulunmuştur. UAT problemi, sinyal kontrol ve TA problemlerinin bir arada çözülmesi gerekliliği sonucunda birleştirilmiş eniyileme problemi olarak tanımlanmıştır (Allsop 1974, Gartner 1974, Marcotte 1983).

2.2. Ulaşım Ağ Tasarımı

UAT sürecinde iki grup etkileşim halindedir ve her grubun birbirinden farklı amacı bulunmaktadır:

- Ulaşım ağı karar vericileri: Toplam sistem maliyetini enküçükmek veya ulaşım ağı üzerinde konfor ve sosyal refahı artırmak; ve
- Ulaşım ağı kullanıcıları: Ulaşım ağındaki konfor veya sosyal refahı dikkate almadan sadece kendi bireysel seyahat maliyetini veya süresini enküçükleme amacındadırlar.

Ulaşım ağı üzerindeki iki nokta arasında seyahat etmek isteyen kullanıcı iki noktayı birbirine bağlayan rotalar arasından birisini seçmek zorundadır. Kullanıcıların rota

seçimini etkileyen faktörler arasında rotaların özellikleri ve kullanıcıların sosyo ekonomik özellikleri sayılabilmektedir. Rota seçimindeki temel yaklaşım, kullanıcıların sürekli en küçük maliyetli rotayı seçeceği yönündedir. Bu yaklaşım denge durumunda bütün kullanılan rotalardaki seyahat maliyetinin aynı olacağı ve bu maliyetin kullanılmayan rotalardaki seyahat maliyetinden daha az veya eşit olacağı kabulüne dayanmaktadır. Bu durum ağ KD altındadır şeklinde tanımlanmakta ve hiçbir kullanıcı kendi rotasını değiştirerek seyahat maliyetini değiştirememektedir (Wardrop, 1952). Bu yaklaşım deterministik KD olarak adlandırılmaktadır. Fakat gerçekte durum böyle değildir. Çünkü her kullanıcı rotalar hakkında tam olarak bilgi sahibi değildir ve her kullanıcının rota seçimi sırasında algılama hataları meydana gelmektedir. Bu durum Stokastik KD durumunu doğurmaktadır (Başkan, 2009). Dolayısıyla, ulaşım ağı kullanıcıları kendi seyahat maliyetlerini veya sürelerini enküçükleme amacıyla olması deterministik KD ve stokastik KD durumlarını meydana getirmektedir. Bu yüzden, ulaşım ağı karar vericilerinin sistem maliyetini enküçüklemesi veya ulaşım ağı üzerindeki konfor ve sosyal refahı artırabilmesi için ağ kullanıcılarının rota seçim davranışlarını gözönünde bulundurması gerekmektedir. Bu arada, ulaşım ağındaki herhangi bir değişiklik ağ kullanıcılarının rota seçim davranışlarını etkilemektedir. Bu durum her iki grubun devamlı karşılıklı etkileşimi ile sonuçlanmakta ve birleşik atama kontrol problemi olarak adlandırılmaktadır.

Fisk (1984) UAT problemini, ulaşım ağı karar vericileri ile ağ kullanıcıları arasındaki karşılıklı etkileşim nedeniyle hiyerarşik bir problem olarak tanımlamış ve ağ karar vericilerinin kullanıcıların davranışları hakkında önceden bilgi sahibi olması durumunda UAT probleminin oyun teorisindeki lider-takipçi veya Stackelberg oyunu olarak tanımlanabileceğini belirtmiştir. Bu durumda ağ karar vericileri lider, kullanıcılar ise takipçi olarak nitelendirilebilmektedir. Bu sebeple Oppenheim (1995) UAT sürecini iki seviyeli programlama problemi olarak tanımlamıştır. Bu iki seviyeli programlama probleminde, liderin yani ulaşım ağı karar vericilerinin karar değişkenleri üst seviyede hesaplanırken takipçi yani ulaşım ağı kullanıcılarının davranışları ise alt seviyede hesaplanmaktadır.

B-V taleplerinin analiz yapılan süre içinde sabit olduğu kabul edilerek statik TA yöntemlerinin UAT probleminin alt seviyesinde kullanıldığı çeşitli çalışmalar literatürde mevcuttur. Davis (1994) UAT problemini verilen bir ulaşım ağında en

uygun kapasite genişletmelerinin nasıl uygulanacağını belirlenmesi şeklinde tanımlamıştır. Lim vd (2005) yaptıkları çalışmada UAT problemini konveks olmayan yapısından dolayı çözümü oldukça zor olarak tanımlamışlar ve UAT probleminin bazı türlerinin iki seviyeli programlama tekniği ile formüle edilebileceği belirtilmiştir.

UAT probleminin konveks olmayan yapısından dolayı çözüm için son yıllarda sezgisel metotların kullanımı oldukça artmıştır. Xu vd (2009) çalışmalarında UAT probleminin çözümü için TB ve GA metotlarını örnek bir ağ üzerinde denemişlerdir. Guogiang ve Jian (2007) tarafından yapılan çalışmada UAT probleminin çözümü için iki seviyeli programlama tekniğinin matematiksel ifadesi verilmiş ve GA tabanlı yeni bir algoritma problemin çözümü için önerilmiştir. UAT probleminin oldukça kompleks bir problem olduğu ve geleneksel eniyileme algoritmaları ile çözümünün oldukça zor olduğu vurgusu yapılmıştır. Poorzahedy ve Abulghasemi (2005) yaptıkları çalışmada Karınca Sistemi (KS) algoritmasını UAT probleminin çözümü için ilk olarak kullanmışlardır. Test ağı üzerinde performansı incelenmiş ve sonuçlar başarılı bulunmuştur.

Benzer şekilde Chiou (2005) UAT problemini ağı performansı en uygun yapan bağ kapasite genişletmelerinin belirlenmesi şeklinde tanımlamıştır. Çalışmada iki seviyeli programlama tekniğinin UAT probleminin formüle edilebilmesi için kullanılabileceği belirtilmiştir.

Poorzahedy ve Rouhani (2007) GA, TB, TAR ve KS'den oluşan hibrit algoritmaları UAT probleminin çözümü için uygulamışlardır. Algoritmaların test edilmesi için gerçek bir ulaşım ağı seçilmiştir. Sonuçlarda hibrit algoritmaların tek başına temel KS'ne göre UAT problemini çözmede başarılı olduğu vurgulanmıştır. Gallo vd (2010) UAT probleminin çözümü için yeni bir eniyileme modeli ve meta-sezgisel bir algoritma önermişlerdir. UAT problemi var olan yolların ve kavşaklardaki sinyal sürelerinin tasarlanarak yol ağının en uygun şekilde işletilmesi olarak tanımlanmıştır. Doğrusal olmayan kısıtlı eniyileme modeli problemin çözümü için formüle edilmiştir.

Jin vd'nin (2009) yapmış oldukları çalışmada SUAT probleminin çözümünde TB ve iki seviyeli programlama yöntemi en uygun parametrelerin bulunması için kullanılmıştır. Üst seviyede amaç fonksiyonu olarak bağ kapasite artırımları için

gerekli olan toplam yatırım maliyeti ve seyahat sürelerinin toplamı kullanılmıştır. Alt seviye ise eğitim tahmin algoritması kullanılarak çözümlenen kullanıcı dengesi olarak ifade edilmiştir.

Babazadeh vd (2011) yapmış oldukları çalışmada, Parçacık Sürüsü Optimizasyonu (PSO) yöntemini UAT probleminin çözümünde kullanmışlardır. UAT problemi iki seviyeli programlama problemi olarak tanımlanmıştır. Problemin amacı, ulaşım ağına bütçe sınırlarını aşmadan yeni eklenecek bağlar ile toplam seyahat süresinin enküçüklenmesidir. Problemin üst seviyesi bağlar üzerindeki kullanıcıların toplam seyahat süresini, alt seviye KD altındaki trafik atamasını temsil etmektedir. Geliştirilen modelin test ulaşım ağı üzerinde sayısal uygulaması yapılmıştır. Ayrıca, PSO yöntemi ile elde edilen sonuçlar KKO yöntemi ve hibrid KKO yöntemi ile elde edilen sonuçlar ile karşılaştırılmıştır. PSO yönteminin KKO yöntemine göre problemi daha kısa sürede çözdüğü, hibrid KKO yöntemi ile benzer sonuçlar verdiği belirtilmiştir.

Chen ve Xu (2012) yapmış oldukları çalışmada, iki seviyeli programlama ile stokastik UAT problemini çözmek için Amaç Programlama yöntemini kullanarak bir yaklaşım önermişlerdir. İki seviyeli programlamanın üst seviyesinde bağ kapasite genişletmeleri için toplam yatırım maliyeti ve seyahat süreleri dikkat alınırken alt seviyede ise kullanıcıların rota seçim kararları dikkate alınmıştır. Çalışmada amaç programlama ile ilgili üç farklı model oluşturulmuştur. Oluşturulan modelleri çözmek için benzetim bazlı GA yöntemi geliştirilmiştir. Geliştirilen modelin test ağı üzerinde sayısal uygulaması yapılmıştır. Çalışmanın sonuçlarında, amaç programlamanın ve benzetim bazlı GA yönteminin stokastik UAT problemleri için uygulanabileceği belirtilmiştir.

B-V taleplerinin analiz yapılan süre içinde düzgün dağılmayıp, taleplerin dağılımının zaman içinde değişiklik gösterdiği kabulü yapılan DTA yöntemlerinin UAT problemi içinde kullanılması son yıllarda artış göstermektedir ve bununla ilgili literatürdeki çalışmalar kısıtlıdır.

Abdelfatah ve Mahmassani (1998) yapmış oldukları çalışmada birleşik sinyal kontrolü ve DTA problemi için matematiksel bir formülasyon ve benzetim bazlı çözüm algoritması sunmuşlardır. Gerçek ulaşım ağları üzerinde sayısal uygulamalar yapmışlardır. Geliştirdikleri algoritmada sinyal parametrelerini eniyilemek için

Webster formülasyonu kullanılmıştır. Chen ve Ben-Akiva (1998) ağ kullanıcıları ile karar vericiler arasında işbiriksiz oyun olarak birleşik sinyal kontrol ve DTA problemini modellemek için oyun teorisi yöntemi geliştirmişlerdir. Cheng vd (2004) en uygun eşzamanlı sinyal süre planlarını belirlemek için oyun teorisi örneğini uygulamışlardır. Algoritmanın en önemli avantajı her iterasyon için sadece bir benzetim gerektirmesi ve büyük boyuttaki gerçek ağlar için uygulanabilir olmasıdır. Abdelfatah ve Mahmassani (2001), çalışmalarında Webster formülasyonu yerine benzetim bazlı sinyal eniyileme modeli kullanarak birleşik sinyal kontrol DTA problemi için çözüm algoritması sunmuşlardır.

Janson (1995) yapmış olduğu çalışmada, statik KD'sine göre farklı sonuçlar veren Dinamik Kullanıcı Dengesine (DKD) bağlı olarak yapılan UAT'nin etkilerini göstermiştir. Çalışmanın sonuçlarında DKD'ye göre yapılan UAT'nin daha makul sonuçlar verdiği belirtilmiştir. Karoosontawong ve Waller (2009) trafik sinyal eniyilenmesi ile DKD'sini eş zamanlı olarak çözen tamsayıli lineer iki seviyeli model önermiştir. Üst seviyede toplam sistem seyahat süresini enküçükleyen en uygun trafik sinyal parametreleri (devre süresi, yeşil süre, offset süresi ve faz sırası) bulunurken alt seviyede ise Daganzo'nun hücre aktarma modelinin yerleştirildiği lineer programla DKD aranmaktadır. Bu çalışmada modeli çözmek için iyileştirilmiş Reaktif TAR algoritması kullanılmıştır. Model iki test ağı üzerinde uygulanmış ve modelin performansı çözüm kalitesi, yakınsama hızı ve CPU açısından değerlendirilmiştir. Çalışmanın sonuçlarında geliştirilen modelin başarılı bulunduğu belirtilerek modelin GA, TB, KKO ile çözümlenerek elde edilen sonuçların karşılaştırılması gerektiği vurgulanmıştır.

Chen ve Hu (2009) sinyal parametreleri ile TA arasında DKD'sini bulmayı amaçlayan bir model önermişlerdir. Model iki seviyeli programlama ile çözülmüştür. Üst seviyede devre süresi ve yeşil sürelerini içeren sinyal parametreleri çözümlenirken alt seviyede ise ağda zamana bağlı KD akımları belirlenmektedir. Çalışmada sinyal parametreleri için sabit zamanlı ve uyarmalı sinyal kontrol yöntemi dikkate alınmıştır. DKD'si altındaki bağ akımları benzetim bazlı DTA yazılımı olan DynaTAIWAN ile elde edilmiştir. Önerilen model Kaohsiung şehri içindeki bir sinyalize ulaşım ağına uygulanmıştır. Sonuçlarda, hem sabit zamanlı hem de uyarmalı sinyal kontrolüne göre yapılan analizlerde bağ akım değerlerinin değişimi

iterasyon sayısına bağı olarak sabit duruma ulaştığı belirtilmiştir. Ayrıca, ağ üzerindeki ortalama seyahat süreleri sabit zamanlı sinyal kontrol yönteminde uyarımalı sinyal kontrol yöntemine göre daha iyi olduğu vurgulanmıştır.

Carey (2009) yapmış olduğu çalışmada DKD modeli geliştirmiştir. Bu model, İlk Giren İlk Çıkar (İGİÇ) kuralını sağlayarak KD'sini sağlamaktadır. Çalışmada modelin ulaşım ağları üzerinde test edildiği ve modelin hızlı şekilde yakınsadığı belirtilmiştir.

2.2.1. İki Seviyeli Programlama

İki seviyeli programlama bir çift eniyileme probleminden oluşan programlama tekniği olarak tanımlanmaktadır. Bu tür programlamada üst seviye ve alt seviyelerin parametreleri her iki seviye için girdi ve çıktı olarak kullanılmaktadır. İki seviyeli programlama tekniği genel olarak Denklem (2.1)-(2.2)'de verildiği gibi ifade edilmektedir (Başkan, 2009).

$$\min_x F(x, y) \quad (2.1)$$

Kısıtına bağı olarak;

$$G(x, y) \leq 0$$

Burada y aşağıda verilen eniyileme probleminin herhangi bir x için çözümüdür.

$$\min_y f(x, y) \quad (2.2)$$

Kısıtına bağı olarak;

$$g(x, y) \leq 0$$

İki seviyeli programlama metodunda Denklem (2.1)'de verilen eniyileme problemi üst seviye, Denklem (2.2)'de verilen problem ise alt seviye olarak adlandırılmaktadır. Yang ve Yagar (1995) yaptıkları çalışmada, sıkışıklık altındaki ulaşım ağlarında TA problemini ve sinyal kontrol problemini ele alan bir model önermişlerdir. Denge bağı akımlarının ve sinyal kontrol parametrelerinin belirlenmesinde kuyruklanma ve tıkanıklığı dikkate alan bu model iki seviyeli programlama yöntemi ile formüle

edilmiştir. Bu formülasyonda, alt seviye, sinyal kontrolündeki değişikliklere sürücülerin nasıl cevap verdiklerini yansıtan ve bağlar üzerindeki kuyruklanmayı dikkate alan KD altındaki TA problemini temsil etmektedir. Üst seviye ise, sinyal kontrolündeki değişikliklere göre sürücü rota seçim davranışlarını dikkate alarak amaç fonksiyonunu eniyileyen sinyal kontrol problemini temsil etmektedir. Sinyal kontrolüne göre kuyruk gecikmelerini ve denge bağ akımlarını elde etmek için duyarlılık analizi uygulanmıştır. Elde edilen kuyruk gecikmeleri ve denge bağ akımları eğim düşüm algortimasında kullanılarak önerilen iki seviyeli trafik sinyal kontrol problemi çözülmüş ve sayısal uygulama yapılmıştır. Elde edilen sonuçlara göre, geliştirilen algoritma ile global çözüm bulunamamış olmasına rağmen globale yakın çözümler elde edilmiştir. Önerilen duyarlılık analizi bazlı çözüm yönteminin en önemli noktası, sinyal kontrolüne göre kuyruk gecikmelerinin ve denge bağ akımlarının elde edilme yoludur. Ceylan ve Bell (2004) Stokastik KD bağ akımlarını kullanarak sinyal tasarım parametrelerinin eniyilenmesi için GA yaklaşımını önermişlerdir. Stokastik KD atama problemi eşdeğer enküçükleme problemi olarak ifade edilmiş ve Path Flow Estimator (PFE) (Bell vd, 1997) kullanılarak çözülmüştür. Sinyal tasarım parametreleri; devre süresi, her bir fazın sinyal süresi ve kavşaklar arasındaki koordinasyon olarak tanımlanmıştır. Amaç fonksiyonu olarak ağ performans indeksi kabul edilmiştir. Ağ performans indeksi, sistem performansını gösteren tüm trafik akımlarının durma sayıları ve gecikmelerin toplamı olarak ifade edilmiştir. TA ve trafik kontrol problemleri, GA entegrasyonu ile oluşturulan GATRANSPFE modeli ile çözülmüştür. GATRANSPFE'nin çözüm yeteneği örnek test ağı üzerinde gösterilmiş ve KİY ile karşılaştırılmıştır. Sonuçlara göre GA yaklaşımı oldukça başarılı olmuştur. Sonuçlarda, ağ performans indeksinde KİY çözümüne göre % 20 iyileşme olduğu belirtilmiştir.

Benzer şekilde Ceylan ve Bell (2005) çalışmalarında UAT probleminin çözümü için iki seviyeli programlama tekniğini kullanmışlardır. Sıkışıklık etkisi altındaki sinyalizasyon ulaşım ağlarında üst seviye probleminin çözümü için GA tekniği kullanılmıştır. Üst seviye probleminin çözümünde GA tekniği önceden belirlenen kısıtlara uygun sinyal sürelerinin bulunmasında kolaylık sağlamıştır. Alt seviye problemi olarak Stokastik KD ataması PFE (Bell vd, 1997) programı ile çözülmüş ve bağ akımları bulunmuştur. Amaç fonksiyonu olarak TRANSYT performans indeksi kullanılmıştır.

Varia ve Dhingra (2004) sıkışıklık etkisi altındaki sinyalize kavşaklardan oluşan ulaşım ağında sistem optimumu altında benzetim bazlı DTA modeli geliştirmişlerdir. Sabit sinyal süreli ulaşım ağındaki seyahat maliyetinin enküçüklenmesi ve sinyal sürelerinin eniyilenmesi için GA tekniği kullanılmıştır. Sonuçlarda, GA tekniğinin dinamik sistem optimumu altında benzetim bazlı DTA problemini çözebildiği fakat büyük sayıda jenerasyon sayısı ve hesaplama süresi gerektiği belirtilmiştir.

Chen vd (2010) yapmış oldukları çalışmada, ulaşım ağındaki bağların en uygun kapasite artırımlarını belirlemek için stokastik UAT problemi üzerine çalışmışlardır. Bu amaçla üç farklı stokastik UAT modeli geliştirmişler ve bu modeller iki seviyeli programlama yöntemi ile tanımlanmıştır. İki seviyeli programlamanın üst seviyesinde en uygun bağ kapasite artırımları belirlenirken, alt seviyede ise kullanıcı dengesi altındaki bağ trafik hacimleri belirlenmektedir. Oluşturulan üç farklı stokastik UAT modellerini çözmek için benzetim bazlı GA yöntemi geliştirilmiştir. Test ağı üzerinde sayısal uygulamalar yapılmış ve tutarlı sonuçlar alınmıştır.

2.2.2. Karşılıklı İteratif Yaklaşım

UAT probleminin çözümü için önerilen diğer bir yaklaşım KİY yöntemidir. KİY yönteminde sinyal kontrol ve TA problemleri ayrı ayrı çözümlenerek problem çıktıları diğer problemin girdisi olarak kullanılmaktadır. Bu iteratif yaklaşım ilk olarak Allsop (1974) ve Gartner (1974) tarafından önerilmiştir. KİY yaklaşımı aşağıdaki adımlar takip edilerek uygulanır (Ceylan, 2002):

Adım 0: Başlangıç ataması sonucu elde edilen denge bağ akımları kullanılarak sinyal kontrol programı ile en uygun sinyal sürelerinin bulunması.

Adım 1: Mevcut sinyal süreleri ve bağ maliyet fonksiyonu kullanılarak yeniden trafik ataması yapılması.

Adım 2: Elde edilen bağ akımlarına göre yeniden en uygun sinyal sürelerinin bulunması.

Adım 3: Tüm adımların sinyal sürelerinde ve akımlarda değişiklik olmayıncaya kadar devam ettirilmesi.

KİY yönteminde iterasyondan iterasyona denge akımlarının dalgalanma göstermemesi için denge bağ akımlarına Ardışık Ortalamalar Yöntemi (AOY)

yakınsama prosedürü uygulanmaktadır. Ağ tasarım probleminin çözümü için KİY yönteminin matematiksel ifadesi Denklem (2.3)-(2.4)'de verildiği gibi ifade eilmektedir (Ceylan, 2002).

$$\min_{\psi \in \Omega} F(\psi, \mathbf{q}) \quad (2.3)$$

Kısıtına bağlı olarak;

$$\mathbf{q} = \mathbf{q}^*$$

$$\min_{\mathbf{q} \in S} Z(\psi, \mathbf{q}) \quad (2.4)$$

Kısıtına bağlı olarak;

$$\psi = \psi^*$$

Burada \mathbf{q}^* ve ψ^* sırasıyla denge bağ akımları ve sinyal süreleri kümesidir. S ise sinyal süreleri ψ dikkate alındığı durumdaki kullanıcı dengesi akımları kümesidir. Allsop ve Charlesworth (1977) orta ölçekli ulaşım ağları için sinyal kontrol ve TA süreçlerini birleştiren UAT problemini çözmede KİY yöntemini kullanmışlardır. Çalışmada TA süreci Deterministik KD ataması olarak temsil edilmiştir. Sonuç olarak elde edilen sinyal süreleri ve denge bağ akımları Gershwin ve Tan (1979) ve Dickson (1981) çalışmalarında belirtildiği gibi en uygunu bulmada yeterli değildir. Ayrıca Suwansirikul vd (1987) yapmış olduğu çalışmada UAT probleminin çözümü için Hooke-Jeeves metodu tabanlı arama metodu kullanmışlardır. Sonuçlar küçük bir test ağı üzerinde KİY yöntemi ile kıyaslanmış ve daha başarılı bulunmuştur. Fakat arama metodunun hesaplama zorluğu nedeniyle sadece küçük ağlar için uygun olduğu belirtilmiştir.

2.3. Trafik Atama

TA problemi ile B-V noktaları arasındaki talep ulaşım ağına yüklenerek bağ akımları bulunmaktadır. TA yapılmasındaki amaçlar aşağıdaki gibi sıralanabilir;

- Ulaşım ağının mevcut durumunu belirlemek.
- Seyahatlerin ulaşım ağına yüklenmesi sonucu ağda meydana gelecek etkileri belirlemek.

- Gelecekteki seyahatleri mevcut ulaşım ağına yükledikten sonra ağda yapılması gereken düzenlemeleri ve bu düzenlemelerin yapım önceliklerini ortaya koymak (Başkan, 2009).

Seyahat maliyetini bağların geometrik özellikleri yanısıra bağlar üzerindeki trafik hacmi ve kavşaklardaki gecikmeler etkilemektedir. Bu nedenle, en düşük maliyetli rota her zaman en kısa mesafeli bağ olmayabilmektedir. Bir rota mesafe olarak kısa olmasına rağmen üzerindeki bağ ve düğümlerdeki gecikmelerin fazla olması nedeniyle o rotanın seyahat maliyeti artabilmektedir. Bu durumda ulaşım ağı kullanıcıları tarafından yeni rotalar aranmaktadır. Ağ kullanıcıları tarafından kısa mesafeli yola göre daha uzun mesafeli fakat daha düşük maliyetli rota tercih edilmektedir. TA probleminin amacı ulaşım ağı topolojisi, bağ maliyet fonksiyonu ve B-V seyahat miktarlarına göre bağ akımlarının bulunmasıdır. Problemin çözümü her bir ağ kullanıcısının B–V çiftleri arasındaki en düşük maliyetli rota üzerinde seyahat ettiği kabulüne dayanmaktadır (Ceylan 2002).

TA'nın çözümü için ağ üzerinde seyahat eden sürücülerin güzergah seçim kararlarını hangi ilkeler doğrultusunda verdiklerinin belirlenmesi önem taşımaktadır. TA problemi, yol ağındaki trafiğin güzergahlara dağılımı sonrasında ulaşılan *denge* durumu ile tanımlanarak matematiksel olarak çözümlenebilmektedir. Buna göre denge durumunda herhangi bir B-V çiftini bağlayan tüm kullanılan rotalardaki seyahat maliyetlerinin eşit olacağı ve ayrıca bu rotalardaki seyahat maliyetlerinin herhangi bir kullanılmayan rotadaki seyahat maliyetine eşit veya ondan daha az olacağı kabulü yapılmaktadır. Bu durumda ulaşım ağının kullanıcı dengesi altında hizmet verdiği kabulü yapılmakta ve hiçbir sürücü tek taraflı olarak rotasını değiştirerek kendi seyahat maliyetini azaltamamaktadır. Bu durum Wardrop (1952) tarafından öne sürülen ve Wardrop'un ilk prensibi olan Deterministik KD ataması durumudur. Bu prensibe göre tüm kullanıcıların ağ hakkında aynı derecede mükemmel bilgiye sahip oldukları kabulü yatmaktadır.

Herhangi bir ulaşım ağında rota seçim problemi için Deterministik KD koşulları varyasyonel eşitsizlik problemine (Smith 1979) eşdeğerdir. \mathbf{h}^* ile belirtilen rota akımları Denklem (2.5)'de verilen ifadeyi doğrulamaktadır (Başkan, 2009).

$$\sum_{w \in W} \sum_{r \in R_w} c_w^r(\mathbf{h}^*) \cdot (h_w^r - h_w^{r^*}) \geq 0 \quad \forall \mathbf{h} \in \Omega \quad (2.5)$$

Burada h_w^r ve c_w^r sırasıyla herhangi bir w B-V çifti arasındaki r rotasındaki akım ve seyahat maliyetidir. Ayrıca \mathbf{h} rota akımları vektörü, W olarak gösterilen B-V çiftleri kümesi, R_w w B-V çifti arasındaki rotalar kümesi ve Ω olası rotalar kümesidir. Deterministik KD yaklaşımında Denklem (2.6)-(2.8) ile verilen kısıtların sağlanması gerekmektedir.

$$\sum_{r \in R_w} h_w^r = g_w, \quad w \in W \quad (2.6)$$

$$h_w^r \geq 0, \quad r \in R_w, \quad w \in W \quad (2.7)$$

$$v_s = \sum_w \sum_{r \in R_w} \delta_w^{rs} h_w^r, \quad s \in S \quad (2.8)$$

Burada g_w , w B-V çifti arasındaki seyahat talebi, v_s , s bağı üzerindeki akım, δ_w^{sr} link-rota belirleme matrisi elemanı, S bağlar kümesidir. Bağ seyahat maliyet fonksiyonu (c_s) sürekli artan ve pozitif bir fonksiyon olması durumunda Deterministik KD yaklaşımı Beckman vd (1956) tarafından önerilen konveks eniyileme problemi şeklinde yazılabilir.

$$\min_{\mathbf{v}} \sum_{s \in S} \int_0^{v_s} c_s(x) dx \quad (2.9)$$

Denklem (2.9)'da verilen amaç fonksiyonunda \mathbf{v} bağ akımları vektörü olarak ifade edilmiştir. Verilen amaç fonksiyonu konveks bir fonksiyon olduğu için Deterministik KD sonucu elde edilen bağ akımları tekdir. Fakat rota akımları altında konveks olmadığı için rota akımlarının çözümü Deterministik KD koşulları altında tek değildir (Sheffi 1985).

Gerçekte tüm yol kullanıcılarının algılamalarının eşit ve kusursuz düzeyde olması beklenemez. Bu husustaki önemli etkenler, sürücülerin algılama seviyeleri, seyahat edilen ulaşım ağı hakkındaki bilgi düzeyleri ve alışkanlıklarıdır. Kullanıcıların seyahat süresi algılamalarının aynı kabul edilmesi ağ tasarımında hatalara neden olabilmektedir. UAT problemlerinin çözümünde Deterministik KD yaklaşımları geniş uygulama alanına sahiptir. Uygulama açısından Deterministik KD atamasının

içerdiği kolaylıklara rağmen, Stokastik KD denge yaklaşımının sürücü davranışlarını daha iyi temsil etmektedir (Zhang ve Gao 2007). Bu nedenle, kullanıcı algılamalarını yumuşatan ve buna rastgelelik ekleyen bir yaklaşım olan Stokastik KD kavramı son yıllarda ulaşım ağ tasarımında sıklıkla kullanılmaya başlanmıştır (Ceylan, 2009).

Stokastik KD kavramı, Deterministik KD'nin geliştirilmiş hali olarak kabul edilebilir. Eğer sürücüler tarafından algılanan güzergah seyahat süreleri tamamen doğruysa, Stokastik KD, Deterministik KD ile aynı hale gelmektedir (Sheffi 1985). Stokastik ve Deterministik KD arasındaki fark, Stokastik KD modelinde bir yol kullanıcısının, diğerlerinin de düşündüğü ortak yolculuk seyahat süresini dikkate almak yerine kendine özgü bir seyahat süresi tanımlamasıdır. Stokastik TA, kullanıcıların seyahat süreleri konusundaki algılama çeşitliliğini ele alır. Bu durum, belirli bir güzergah üzerindeki algılanan seyahat süresi, kullanıcıların arasında dağıtılmış rastgele bir değişken olarak dikkate alarak gerçekleşir ve her bir kullanıcı için farklı seyahat süreleri modellenebilir (Ceylan, 2009).

Ortuzar ve Willumsen (1994) Stokastik KD için “Her bir kullanıcı, en düşük “algılanan” seyahat süresini dikkate alarak güzergah seçimini yapar; başka bir deyişle stokastik KD altında her kullanıcı en düşük “algılanan” seyahat süreli güzergahı kullanır ve hiçbir kullanıcı kendi güzergahını değiştirmez” tarifini yapmışlardır. Stokastik KD için bir diğer tanım, “yol kullanıcıları, karşılıklı olarak güzergah değişimi yapıldığında kendi seyahat sürelerinin iyileşmediğine inanırlar ve algılama düzeyleri doğrultusunda Wardrop dengesini kurmaya çalışırlar” şeklindedir (Daganzo ve Sheffi 1977).

Stokastik KD ataması için yaygın olarak kullanılan atama modelleri logit ve probit tabanlı modellerdir. Bu modellerle ilgili detaylı bilgiler Dial (1971), Bell ve Iida (1997), Sheffi (1985), Ortuzar ve Willumsen (1994) ve Ceylan (2002)'de bulunabilir.

Stokastik KD yaklaşımında, deterministik yaklaşımdan farklı olarak bazı kullanıcıların seyahat esnasında yüksek seyahat süreli güzergahları seçtikleri kabul edilmektedir. Böylece yol ve trafik koşulları hakkındaki düşük bilgileri ya da alışkanlıkları nedeniyle yüksek seyahat süreli güzergahları seçen kullanıcıların varlığından söz edilebilmektedir. Dolayısıyla stokastik yaklaşım, deterministik yaklaşıma göre daha akılcıdır.

Stokastik KD, Deterministik KD'ne göre daha gerçekçi bir yaklaşım sağlamakla kalmayıp aynı zamanda bir takım avantajları da beraberinde getirmektedir. En belirgin avantajlardan birisi, Deterministik KD'nin aksine, denge durumunda güzergah akımları tekil olarak belirtilebilmektedir. Aynı B-V çiftini bağlayan güzergah akımlarının eşit olması durumunda, Deterministik KD'de yaşanan belirsizlik ile karşılaşılmaz (Ceylan, 2009).

2.4. Dinamik Trafik Atama

Dinamik Trafik Atama (DTA), B-V noktaları arasındaki talebin ulaşım ağına yüklenmesi ve zamana bağlı bağ ve rota akımlarının bulunması olarak tanımlanabilmektedir. Statik trafik atama yöntemleri analiz süresi boyunca ortalama ve sabit koşullar üzerine odaklanmıştır. Bir başka deyişle, analiz süresi boyunca rota ve bağların seyahat süreleri ve hacimleri sabit olarak düşünülmüş, zamanla bu değerlerde bir değişiklik olabileceği dikkate alınmamıştır. Bu durumda, bir ağ üzerindeki trafik ile seyahat süresi arasındaki ilişki matematiksel bir form olarak hacim-gecikme fonksiyonu ile açıklanmıştır. Statik atama yöntemleri ile kullanıcı dengesi altındaki trafik hacimleri kolaylıkla elde edilebilmektedir. Statik trafik atama yöntemleri uzun dönemli planlamalar için uygun olarak görülmektedir. Statik atama modellerinin gelişmesindeki ana amaç gelecek yıllar için ulaşım ağındaki kapasite artırma gibi yapılacak düzenlenmelerin planlanmasıdır. Buna rağmen, statik atama yöntemleri ulaşım ağı üzerinde zamanla değişen trafik akımlarını ve şartlarını yansıtamamaktadır. Bir başka deyişle, statik atama yöntemleri kısa süreli ve geçici seviyedeki tıkanıklık sorununu dikkate alacak ölçütlerin değerlendirilmesinde yetersiz kalmaktadır.

DTA, ulaşım ağlarındaki trafiğin zamana bağlı değişimini dikkate alır. DTA'da KD yaklaşımı iki şekilde düzenlenebilir. Birincisi, statik modellerdeki ağ kullanıcısının ağ hakkında mükemmel bilgiye sahip olma durumunda ve rota seçim kriterinde ağ üzerindeki bağların seyahat sürelerinin zamanla değiştiğinin göz önüne alınması; ikincisi ise dinamik yaklaşım gözönünde bulundurularak bir B-V çifti için farklı zamanlarda hareket eden kullanıcılar farklı seyahat süreleri ile karşı karşıya kalacakları için KD altında kullanılan rotalar üzerindeki seyahat süresinin eşitliği sadece aynı B-V çifti arasında ve aynı anda hareket ettiği kabulü yapılan kullanıcılar

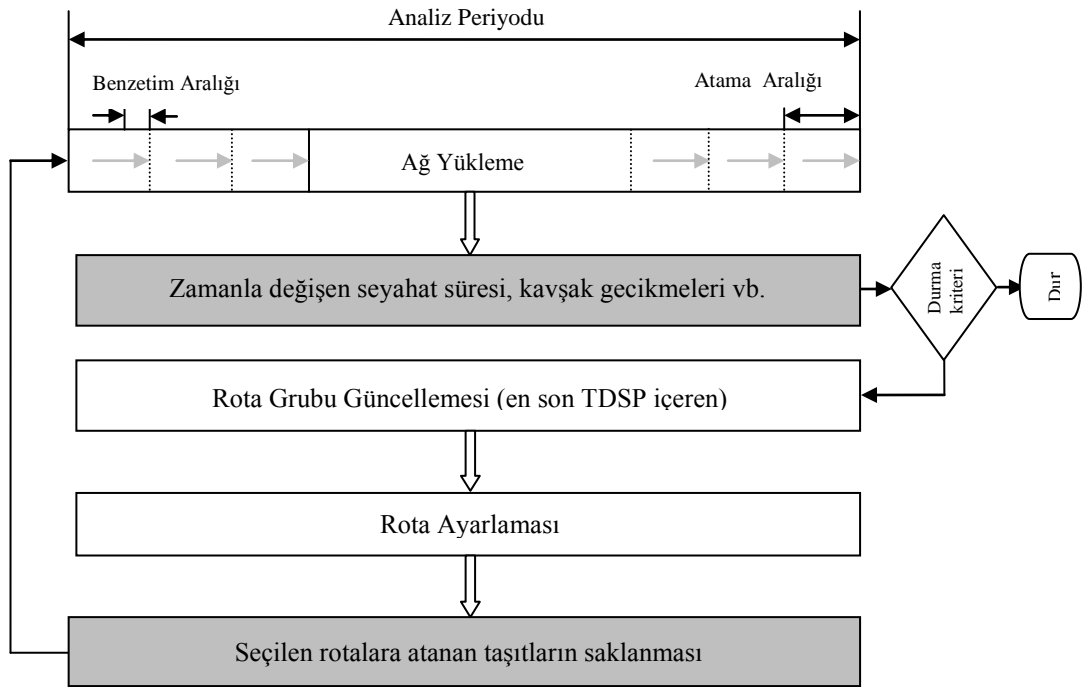
için geçerli olmasıdır. Bu iki kabul dinamik atama yöntemlerinin statik atama yöntemlerinden temel olarak ayrıldığı noktadır. Birinci kabul zamanla değişen ulaşım ağı trafik koşullarını tanımlamakta ve ulaşım ağı üzerinde zamanla değişen bağ seyahat sürelerini dikkate alarak en kısa rotayı bulmayı sağlayan etkin bir çözüm yolu gerektirmektedir. Bu yaklaşım; zamanla değişen ağ trafik koşullarının tanımlanması için ağ üzerinde trafik zaman içinde yayılırken, trafik akım özelliklerini, akım korunumunu ve trafik akım değişkenlerinin temel ilişkilerini geçici olarak korumayı sağlayan bir uygulama gerektirmektedir. Bu gereklilik hem analitik hem de benzetim bazlı yaklaşımları içeren çeşitli ağ yükleme modellerini gündeme getirmiştir. Seyahat süresini enküçükleyen en kısa rotanın hesaplanması Zamana Bağlı En Kısa Rota algoritması (Time Dependent Shortest Path-TDSP) yardımıyla yapılabilir. İkinci kabul ise KD yaklaşımını parçalara ayırır. Böylelikle, KD tüm analiz süresi için değil her bir kalkış zamanı için elde edilmiş olmaktadır. Bu durum Dinamik Kullanıcı Dengesi (DKD) olarak adlandırılmaktadır. Her ağ kullanıcısının en kısa rota seçimi, seyahat süresince ağ üzerindeki tıkanıklık seviyelerine, sırasıyla kendisinden önce, kendisiyle aynı zamanda ve kendisinden sonra hareket eden kullanıcıların ağ üzerindeki hareketlerine ve rota seçimlerine bağlı olmaktadır. Böylelikle, ilk rota seçim kümesinden başlayarak ve rota seçim kümeleri aşamalı olarak güncellenerek iteratif yaklaşım ile çözüm bulunur. Ulaşım ağlarında rota seçim kümelerinin güncellenmesi sonsuza kadar sürebilmekte ve tam bir KD durumunun bulunması zor olabilmektedir. DTA modellerinin çoğunda amaç, makul sürede elde edilerek uygulama için etkin şekilde yakınsayan yaklaşık bir KD durumunun bulunmasıdır. DTA'da KD bulunmasında en yaygın olarak kullanılan yöntem iteratif ardışık olarak uygulanan üç adımlı algoritma olup aşağıdaki sorulara cevap aranır:

- Ağ yükleme: Rota seçim kümesi, rotalar, rota akımları, rota seyahat süreleri nelerdir?;
- Rota kümesi güncellenmesi: Mevcut rota seyahat süresi, her B-V çifti ve harekete geçiş zaman aralığı için yeni en kısa rotalar hangileri?;
- Rota ayarlaması: Rota kümelerinin güncellenmesi ile daha iyi bir DKD elde edebilmek için taşıtlar rotalara nasıl yüklenmelidir?

DTA modellerinin yapısı birbirine benzer olmasına rağmen, birçok DTA modeli yukarıda bahsedilen bileşenlerin nasıl uygulanacağı konusunda birbirlerinden ayrılmaktadır. Rota değerlendirme aşamasında verilen rota seçim kümesini takip eden taşıtların etkisi ağ yükleme süreci ile belirlenmektedir. Analitik modeller, ağ içinde trafiğin nasıl yayılacağını tahmin etmek için tipik çıkış fonksiyonlarını kullanmakta, benzetim bazlı yaklaşımlar ise 5-10 sn zaman aralıklarında trafik akımlarındaki değişimi yansıtan mesoskopik benzetim yaklaşımını kullanmaktadır. İkinci adım olan rota kümesinin güncellenmesi, ağ yükleme aşamasının sonuçlarının analizini içermektedir. Ağ yükleme aşamasında belirlenen seyahat süreleri ve tıkanıklık durumuna bağlı olarak, her B-V çifti arasında harekete geçiş zaman periyodu için en düşük karşı karşıya kalınan seyahat süreli rotalar TDSP algoritması ile belirlenmektedir. Belirli bir B-V çifti ve harekete geçiş zaman periyodu için yeni bulunan zamana bağlı en kısa rota, rota kümesinin güncellenmesi için daha önceki iterasyonlarda bulunan tüm zamana bağlı en kısa rotalar ile birleştirilmektedir. Bu adımı takip eden adım üçüncü ve son adım olan rota atama ayarlamasıdır.

Kullanıcılar rota seçimlerini en düşük karşı karşıya kalınan seyahat süreli rotaları seçerek yaparlarsa, atama denge durumuna yaklaşabilir. KD'nin bulunmasında ana zorluk farklı kullanıcıların rota seçimleri ve seyahat süreleri arasındaki etkileşimin olmadığı kabulüdür. Tüm kullanıcılar bir önceki adımda bulunan en kısa rotalara yönlense, bu rotaların yoğunluğu artacağından artık bu rotalar en kısa rotalar olmayacaktır. Bunu önlemek için sadece bazı kullanıcıların rota seçimleri düzenlenmelidir. Genellikle bu adım, rota kümesi içindeki rotaların, atama akımları ile ne kadar artırılmasının veya ne kadar azaltılmasının belirlenmesini içermektedir. Normalde diğer en kısa seyahat süresine sahip rotalar ile birlikte yeni bulunan zamana bağlı en kısa rotaların seyahat süreleri, atama sonundaki akımlar ile artacaktır. Bunun sonucu olarak da, önceden daha uzun seyahat süresine sahip rotalar üzerindeki akımların azalması ile bu rotalar üzerindeki seyahat süreleri azalacaktır. Bu adımda dikkat edilmesi gereken nokta, taşıtların tümünün zamana bağlı en kısa yeni rotayı seçmeyecek olmalarıdır. Rota ayarlaması, mevcut rota kümesi içindeki tüm rotalar arasında eşit seyahat elde etmek için sadece gerektiğinde yapılmaktadır. Rota ayarlamasından sonra, algoritma yeni rota seçimleri sonucunda oluşacak trafik yapısını belirlemektedir. Sonuç olarak bu üç adım ardışık olarak çalışmakta ve durma kriteri sağlanıncaya kadar tekrarlanmaktadır. Bu üç adımlı algoritma Şekil 2.1'de

verilmektedir. Durma kriteri ağ yükleme adımının sonunda hesaplanmaktadır. Ağ yükleme terimi, B-V arasındaki belirli rotalar üzerindeki taşıtların hareketinin, bağ ve rota akımlarının hesaplarının ortaya çıkmasının ardından seyahat süresinin belirlenmesi olarak tanımlanmaktadır. Ağ yükleme, atama modelinin her iterasyonunda yer almaktadır. Atama terimi ise yukarıda bahsedilen üç adımı kapsamaktadır. Bir başka deyişle, atama terimi KD durumu süreci ile ilgilidir (TRB, 2010). DTA modelleri ile ilgili diğer bir önemli nokta daha bulunmaktadır. Statik atama yöntemlerinde tüm seyahatlerin atama süresi içinde hedeflerine ulaştığı kabulü yapılmasına karşın dinamik atama yöntemlerinde böyle bir kabul yapılmamaktadır. Örneğin, bir B-V seyahat süresi atama süresinden daha fazla olabilmektedir. Böylece bu B-V seyahatini yapan taşıtlar atama süresi sonunda varış noktalarına ulaşamayarak ağ içinde kalabilmektedirler.



Şekil 2.1 DTA'nın genel algoritma yapısı (TRB, 2010)

2.4.1. Dinamik Trafik Atama Yöntemleri

DTA konusunda literatüre bakıldığında zamanla probleme değişik açılardan yaklaşan birçok çalışma vardır. DTA problemini ilk olarak ele alan çalışma, Merchant ve Nemhauser'in (1978a, 1978b) yapmış oldukları çalışmalarıdır.

Merchant ve Nemhauser, DTA problemini ayrık zamanlı, doğrusal olmayan ve konveks olmayan program yaklaşımı ile ulaşım ağı üzerindeki toplam seyahat süresini enküçükleyen Dinamik Sistem Optimumu (DSO) olarak formüle etmişlerdir. Bu modelde, birden çok başlangıç ve tek bir varış noktası bulunmaktadır. M-N modelinin amaç fonksiyonu ve kısıtları Denklem (2.10)'da verildiği gibidir (Merchant ve Nemhauser, 1978a; 1978b).

$$\min \sum_{i=1}^I \sum_{j=1}^a h_{ij}(x_{ij}) \quad (2.10)$$

Kısıtlar:

$$x_{i+1,j} = x_{ij} - g_j(x_{ij}) + d_{ij} \quad i = 0, 1, \dots, I \text{ ve } j = 0, 1, \dots, I$$

$$\sum_{j \in A(q)} d_{ij} = F_i(q) + \sum_{i \in B(q)} g_j(x_{ij}) \quad i = 0, 1, \dots, I-1$$

$$x_{0j} = R_j \geq 0 \quad j = 1, 2, \dots, J$$

$$d_{ij} \geq 0 \quad i = 0, 1, \dots, I-1 \text{ ve } j = 1, 2, \dots, J$$

$$x_{ij} \geq 0 \quad i = 0, 1, \dots, I-1 \text{ ve } j = 1, 2, \dots, J$$

Burada i , analiz süresinin eşit ve küçük zaman dilimlerine ayrılmış kısımlarını ($i = 0, 1, \dots, I$), j ulaşım ağı üzerindeki bağları ($j = 1, 2, \dots, J$), h_{ij} her bir bağa ait maliyet fonksiyonunu, g_j her bir bağa ait çıkış fonksiyonunu, d_{ij} i zaman diliminde j bağına giren taşıt sayısını, $F_i(q)$ aynı zaman diliminde bağa giren fakat çıkamayan trafik miktarını, $A(q)$ q düğüm noktasından çıkan bağların kümesini, $B(q)$ q düğüm noktasına giren bağların kümesini temsil etmektedir. x i zaman diliminin başlangıcındaki j bağı üzerindeki trafik miktarı olduğunda, bağ üzerinde oluşan maliyet $h_{ij}(x)$ ve bağdan çıkan trafik miktarı $g_j(x)$ olduğu kabulü yapılmaktadır. Kısıtlardan birinci kısıt temel durum eşitliklerini, ikinci kısıt korunum kısıtlarını, üçüncü kısıt başlangıç koşullarını, dördüncü ve beşinci kısıt ise negatif olmama kısıtlarını temsil etmektedir.

Geçmişte DTA problemi ile ilgili yapılan çalışmalar dört genel gruba ayrılmaktadır: (1) Matematiksel Programlama yaklaşımları, (2) Optimum Kontrol Teorisi

yaklaşımı, (3) Varyasyonel Eşitsizlik (VI) yaklaşımı ve (4) Benzetim Bazlı yaklaşımlar. İlk üç grup analitik yaklaşım olarak isimlendirilmektedir.

Ayrıca, DTA problemi, statik TA problemlerinde olduğu gibi ikiye ayrılmaktadır: (1) ulaşım ağı üzerindeki toplam sistem seyahat süresinin enküçüklendiği Dinamik Sistem Optimumu (DSO) ataması, (2) sürücülerin rota seçiminde kendi seyahat maliyetlerini enküçükledikleri Dinamik Kullanıcı Dengesi (DKD) ataması (Peeta ve Ziliaskopoulos, 2001).

DTA problemi ile ilgili yapılan çalışmalar yapılan yaklaşımlara göre aşağıda verilmiştir. İlgili çalışmaların detayları literatürden bulunabilir.

- Matematiksel Programlama Yaklaşımları

Bu yaklaşımda DTA problemi ayırık zamanlarda formüle edilmektedir. Merchant ve Nemhauser (1978a, 1978b) yapmış oldukları çalışmalarda DTA problemini matematiksel programlama yaklaşımı ile ilk kez formüle etmişlerdir. Formülasyon deterministik, sabit talep, tek bir varış ve DSO ile sınırlandırılmıştır. Merchant ve Nemhauser (M-N) modeli, bağ hacim fonksiyonu olarak seyahat maliyetini göstermek için statik bir bağ performans fonksiyonunu ve trafiği dağıtmak için bir bağ çıkış fonksiyonunu kullanmaktadır. Model, akım bazlı, ayırık zamanlı, konveks ve lineer olmayan programlama formülasyonu ile sonuçlanmaktadır. Modelin, statik SO atama problemine uygun genelleştirme sağladığı gösterilmiştir. Global çözüm, modelin doğrusal versiyonunun parça parça çözülmesi ile elde edilmektedir. Ho (1980), M-N modeli için global çözümün en fazla $N + 1$ doğrusal programın ardışık çözülmesi ile elde edildiğini göstermiştir, burada N periyod sayısını belirtmektedir.

Carey (1986), M-N modelinde önerilen çıkış fonksiyonunun sürekli olarak değişmesi ve en uygunluk analizinin onaylanmasından dolayı M-N modelinin doğrusal bağımsız kısıt özelliğini sağladığını kanıtlamıştır. Carey (1987), çıkış fonksiyonlarının işlenmesi yoluyla daha iyi, konveks ve doğrusal olmayan program olarak M-N problemini yeniden formüle etmiştir. Bu formülasyon asıl formülasyona göre matematiksel ve algoritmik avantajlar sağlamaktadır. Bu model, M-N modeline çok benzemektedir. Formülasyonu konveks yapmak için, çıkış fonksiyonu, M-N modelinde olduğu gibi sadece çıkış akımını sınırlamaktan ziyade bağ çıkış akımlarını sınırlamak için kullanılmıştır. Model konveks olduğu için, standart matematiksel

programlama yazılımları modeli çözmek için kullanılabilir. Çalışmada, daha etkin algoritmalar geliştirilmesi için araştırma yapılması gerektiği belirtilmiştir. Genişletilen M-N modeli, çok varışlı durumlar için uygulanmış, fakat İGİÇ gereksinimi nedeniyle artan konveks olmayan sonuçlar sonuç formülasyonunu problemli olarak bırakmaktadır. Bu durum, tüm matematiksel programlama yaklaşımlarının hem KD hem de SO için zamana bağlı atama problemlerinin kusuru olarak ortaya çıkmaktadır. Trafığe gerçekçi bakış açısından, birden çok varış noktası modellerin açıkça İGİÇ şartını sağlamalarını gerektirmektedir. İGİÇ, fiziksel ve algoritmik olmak üzere iki boyuta sahiptir. Gerçek trafik ağlarında bireysel taşıtlar zaman zaman İGİÇ'i ihlal ettikleri için (geçme, sollama), trafiğin yayılma bakış açısından sadece ortalama bir anlamda İGİÇ gereksinimi sağlanmaktadır. İGİÇ gereksinimi, tek varışlı formülasyonlarda ve belirli özel ağ yapılarında kolaylıkla karşılanmaktadır. Genel ağlarda, bu gereksinim, konveks olmayan kısıt setleri gibi ilave kısıtlar meydana getirecektir (Carey, 1992).

DSO atamasında matematiksel programlama modellerinin bilinen diğer bir yanı ise, trafiğin gerçekliğini etkileyen bağ üzerindeki taşıtların kendini geri tutması olayıdır. Bağ çıkış kısıtları eşitsizlikler olarak sağlandığı zaman trafiğin kendini geri tutma olayı artmaktadır. Bir ağda toplam seyahat gecikmesinin en küçüklenmesi için bazı trafik akımları diğer akımlara göre avantaj sağlayabilmektedir. Buna örnek olarak bir ana yolla bir tali yolun birleştiği kavşakta, tali yol üzerindeki taşıtların kendini tutması gösterilebilir. Aksi belirtilmedikçe, SO atama formülasyonunun çözümünde, bir kavşakta diğer kollardaki trafiğin lehine olacak şekilde bir kol üzerindeki trafiğin geri tutulması gerekebilir. Bir başka deyişle, bir bağ üzerindeki taşıtlar makul olan gecikme değerinden daha fazla olacak şekilde yapay olarak geciktirilebilir. Böyle bir çözüm işletme ve sosyal açıdan kabul edilmemektedir. Modelleme açısından bu durumun engellenmesi için ilave kısıtlar gerekmektedir. Carey ve Subrahmanian (2000) yapmış oldukları çalışmada İGİÇ ve trafiğin geri tutulması konuları açısından problemi ele almışlardır.

Janson'un (1991a, 1991b) yapmış olduğu çalışmalar KD altında DTA problemini matematiksel programlama ile yapılan ilk çalışmalardır. Bu çalışmaların özelliği, daha önceki çalışmalarda kullanılan anlık seyahat süreleri yerine karşı karşıya kalınan seyahat süreleri açısından tanımlanmış bir dengeyi aramasıdır. Çözüm

formülasyonunda B-V akımlarının geçici sürekliliğini sağlamak için doğrusal olmayan tamsayı kısıtlar önerilmektedir. Bu çözüm yönteminin özellikleri yeterince başarılı değildir ve gerçekçi olmayan trafik davranışlarına neden olabilmektedir. Ayrıca, geliştirilen model trafiği modelleyebilmek için statik bağ performans fonksiyonlarını kullanmaktadır.

Birge ve Ho (1993), M-N problemini analiz süresince B-V taleplerini yumuşatan bir kabul yaparak stokastik duruma dönüştürmüşlerdir. Geliştirilen modelde stokastik matematiksel programlama yaklaşımı kullanılmıştır. Model, rastgele değişen senaryoların sonlu sayıda olduğunu kabul etmektedir ve her bir senaryo, her periyoddaki B-V taleplerinin bir kombinasyonu olarak tanımlanmaktadır. Buna karşın, model mevcut atama kararlarının gelecek B-V taleplerinden bağımsız olacağı kabulünü yapmaktadır. Model, ardışık doğrusal eniyileme yardımıyla birkaç zaman periyodu üzerinden parça parça doğrusal konveks olan seyahat maliyetlerini enküçükleyen B-V atama yapılarını belirlemektedir.

Ziliaskopoulos (2000), Daganzo (1994) tarafından geliştirilen hücre iletim modeline dayanan tek varışlı SO altında DTA için doğrusal bir programlama formülasyonu önermiştir. Bu formülasyonun, hücre iletim modeline göre trafik akımının yayılması sırasında olması gereken bağ performans fonksiyonu ihtiyacı gerektirmediği için trafiğin doğasına daha duyarlı olduğu belirtilmiştir. Geliştirilen modelin pratik uygulamalar için işlevsel olmadığı buna karşın DTA probleminin iç yüzünü anlama imkanı verdiği belirtilmiştir.

DTA probleminin çözümü için matematiksel programlama yaklaşımlarının kullanıldığı çalışmaların çoğunda, gerçek ulaşım ağlarına uygulanabilir bir modelin matematiksel programlama yaklaşımı ile geliştirilemeyeceği belirtilmektedir. Bunun nedeni, matematiksel işlenirlik ile trafik gerçeğinin ödünleşmesidir. Bunda, yetersiz modelleme yeteneği ve/veya trafik dinamiğinin gösterimi ile karşı karşıya kalındığında artan ve uygun olmayan kısıtlar etkilidir. İGİÇ özelliğinin tanımlanması için konveks olmayan kısıt ihtiyacı bununla ilgili bir konudur. Genel olarak, DTA probleminin matematiksel programlanması sırasında aşağıda verilen zorluklar bulunmaktadır;

1. Bağ performans ve/veya bağ çıkış fonksiyonu kullanımı;
2. Trafiğin geri tutulması;

3. Büyük ölçekli trafik ağlarında gerçek zamanlı yayılma için etkin çözüm;
 4. Gerçek problem senaryoları için anlaşılabilir, açık bir çözüm prosedürüdür.
- Optimum Kontrol Teorisi Yaklaşımları

DTA probleminin kısıtlı optimum kontrol teorisinde, B-V seyahat oranları zamanın sürekli fonksiyonu olarak kabul edilmekte ve bağ akımları zamanın sürekli fonksiyonu olarak araştırılmaktadır.

Friesz vd (1989) yapmış oldukları çalışmada bağ bazlı optimum kontrol formülasyonunu tek varışlı durum için hem SO hem de KD açısından incelemişlerdir. Modellerde, ulaşım ağının şartları değişirken sistem durumlarındaki değişikliklerin de aynı zamanda gerçekleşeceği kabulü yapılmaktadır. Bir başka deyişle, sürücüler rotalama kararlarını mevcut ağ şartlarına bağlı olarak vermekte fakat şartlar değiştikçe sürücüler rotalama kararlarında sürekli olarak değişiklik yapmaktadırlar. SO modeli, statik modelin genişletilmesidir. SO modeli, optimum çözümdeki bir B-V çifti için kullanılan rotalardaki anlık akım marjinal maliyetlerinin aynı olduğunu ve kullanılmayan rotalardakilere eşit veya onlardan daha az olduğunu kanıtlamaktadır. SO formülasyonunun ayrık zamanlı versiyonunun M-N modeli ile aynı olduğu belirtilmiştir. Modeller, trafiği dağıtmak için çıkış fonksiyonlarını, seyahat maliyetlerini belirlemek için bağ performans fonksiyonlarını kullanmaktadır. Anlamlı bağ performans ve çıkış fonksiyonlarının eksikliği etkin bir çözüm algoritmasının geliştirilmesindeki zorluk olarak göze çarpmaktadır. Çıkış akımları birer fonksiyon olmasına rağmen, formülasyonlar giriş akımlarını kontrol değişkeni olarak ele almaktadır. Birden fazla B-V çiftine sahip ulaşım ağında Wardrop'un birinci ilkesinin zamana bağlı genelleştirilmesinin oluşturulması lineer olmayan çıkış fonksiyonlarını zorlaştırdığından bu yaklaşım problemlilik olarak görülmektedir.

Ran ve Shimazaki (1989a), birden fazla B-V çiftine sahip bir ulaşım ağı için bağ bazlı SO modeli geliştirmek için optimum kontrol yaklaşımı kullanmışlardır. Çalışmada, şartların uygunluğu genelleştirmeyi önlediği için çıkış akımlarını bir fonksiyon olarak tanımlamışlardır. Çözüm yönteminde sayısal yükü azaltmak için lineer çıkış fonksiyonları ve ikinci dereceden bağ performans fonksiyonları kullanmışlardır. Gerçekçi olmayan tıkanıklık modellemesine ilave olarak, çoklu

varışlar için artan İGİÇ konusunu dikkate almamışlardır. Ran ve Shimazaki (1989b) yapmış oldukları çalışmada anlık KD altında DTA bazlı bir optimum kontrol teorisi sunmuşlardır. Bağ çıkışlarındaki akımlar, genelleştirme konusunu engelleyen fonksiyonlardan ziyade kontrol değişkenlerinin seti olarak ele alınmıştır. Bu formülasyonları çözebilecek etkin algoritmalar bulunmamaktadır.

Ran vd (1993) yapmış oldukları çalışmada bağ giriş ve çıkış akımlarını kontrol değişkeni olarak tanımlayarak anlık KD altında DTA probleminde konveks bir yapı elde etmek için optimum kontrol yaklaşımı kullanmışlardır. Dinamik kuyruklanma ve tıkanıklık maliyetlerini dikkate alan klasik maliyet fonksiyonlarının yetersizliğini belirtmişler ve bağ seyahat maliyetini hareket ve kuyruk bileşenlerine ayırmayı önermişlerdir. Bu fonksiyonların, negatif olmayan, artan ve türevlenebilir oldukları kabulü yapıldığı için gerçek trafiği yansıtamayabilir. Ayrıca, bu fonksiyonların örnekleri verilmemiş ve test edilmemiştir. Boyce vd (1995) yapmış oldukları çalışmada, Frank-Wolfe algoritmasını ve genişletilmiş zaman-mekan ağ gösterimi kullanarak problemin ayrık versiyonunu çözmek için bir yöntem önermişler, ancak yöntem detaylı olarak tanımlanmamış ve örnekler üzerinde test edilmemiştir.

DTA probleminin optimum kontrol teorisi yaklaşımı ile çözümü, İGİÇ şartını sağlayan ve düğüm noktalarında taşıtları geri tutmayı engelleyen kısıtların eksikliği, trafik tıkanıklığının yetersiz ve gerçekçi olmayan şekilde modellenmesi, genel ulaşım ağları için çözüm yöntemi eksikliği gibi bir çok nedenden dolayı problemlilik olarak görülmektedir.

- Varyasyonel Eşitsizlik (VI) yaklaşımı

Varyasyonel Eşitsizlik (VI), DTA bağlamında eniyileme, sabit nokta ve tamamlayıcılık gibi çeşitli tip problemler için genel bir formülasyon imkanı sağlamaktadır. VI, dengeyi ele alan ve eniyileme problemlerine eşdeğerleştirilmiş bir mekanizma sağlamaktadır. VI ile benzersizlik gibi matematiksel özellikler basit anlamda tanımlanabilmektedir. Dafermos (1980), statik trafik denge bağlamında VI yaklaşımını tanıtmıştır. Nagurney (1998), VI'nın kapsamlı bir özetini vermiş ve çeşitli denge problemlerini ele almıştır. Asimetrik bağ etkileşiminden dolayı, VI kısıtlandırılmış optimizasyon formülasyonlarından doğan analitik işlenirlik konularını engellemektedir. Bu anlamda, VI daha gerçekçi olarak trafiği yansıtabilmektedir. Genişletmeler ve duyarlılık analizleri uygun olarak

yapılabilmektedir. DTA problemlerinin çeşitli yönleri VI yaklaşımı ile diğer analitik yaklaşımlardan daha iyi olarak ele alınabilmektedir. Diğer analitik modellerin sınırları değişmeden kalmaktadır.

Friesz vd (1993) yapmış oldukları çalışmada gözlenmiş seyahat sürelerinin denklemleri ile varış zamanı/rota seçimini çözmek için sürekli bir VI modeli formüle etmişlerdir. Model, bağ performans fonksiyonlarını, erken/geç varışlar için ceza fonksiyonlarını, seyahat taleplerini, arzu edilen varış zamanlarını ve B-V noktaları arasındaki tüm muhtemel rotaları kullanmaktadır. Rota maliyeti, bağ performans fonksiyonu tarafından belirlenen seyahat maliyetinin ve rota boyunca seyahatin neden olduğu erken/geç varış cezasının bir kombinasyonudur. Formülasyon, seyahat edenin davranışını görece olarak daha gerçekçi temsil etmekte ancak hala çözülmemiş noktalar bulunmaktadır. Örnek olarak, çözümün bulunduğu veya çözümün tekliğinin kanıtı bulunmamaktadır. Formülasyon sürekli sonsuz boyutlu VI problemi olduğu için, problemin çözümü eş zamanlı integral eşitliklerinden oluşan karmaşık bir sistemin çözümünü gerektirmekte ve bunun için etkin bir çözüm yöntemi bulunmamaktadır.

Wie vd (1995), sayısal işlenebilirliği mümkün kılabilmek için eş zamanlı rota/varış denge problemi için ayrık bir VI formülasyonu önermişler ve bunu yaklaşık olarak çözebilmek için sezgisel bir algoritma geliştirmişlerdir. Belirli şartlar altında mevcut çözümün var olduğu kanıtlanmıştır. Çalışmada, Friesz vd'nin (1993) kullandıkları çıkış zaman fonksiyonlarının yerine çıkış akım fonksiyonları kullanmışlardır. Çıkış akım fonksiyonlarının kullanımı trafik akımının gösterim gerçekliğini artırmaktadır. Formülasyon rota bazlı olduğu için, rota numaralandırılması sayısal olarak zor olmaktadır. İlgili rotaların alt kümelerini belirlemek için etkin bir metoda ihtiyaç bulunmakta ve bazı kabullerin yapılması gerekmektedir.

Ran ve Boyce (1996), rota bazlı VI modelleri ile ilgili problemleri çözebilmek için sabit varış zamanlı bağ bazlı ayrık bir VI formülasyonu önermişlerdir. Friesz vd (1993) yapmış olduğu çalışmaya benzer şekilde gözlenmiş seyahat sürelerini de dengelemişlerdir. Çalışmada, analitik modeller ile artan trafik gerçekliğinin yansıtılması konusundaki eksiklikleri kısmi olarak azaltmak için bir kuyruk gecikme bileşeni hesaba katılmıştır. Buna rağmen, gerçek ağlarda kapasite ve doymuş akım kısıtları sayısal yükü önemli ölçüde artırmaktadır. Ayrıca, rota bazlı formülasyonlar,

rota kılavuzu kapsamında doğal olarak uygun olmaktadır. Ran vd (1996), eş zamanlı varış zamanı/rota seçimi problemi için önerilen bağ bazlı VI modelini genişletmişlerdir.

Chen ve Hsueh (1998) tarafından yapılan çalışmada, bağ üzerindeki seyahat süresinin sadece bağ giriş akımının fonksiyonu (bağ giriş, çıkış akımının fonksiyonu ve bağ üzerindeki taşıtların sayısı yerine) olarak ifade edilebileceği gösterilmiştir. Seyahat süresi fonksiyonunun jakobian matrisinin asimetrik olduğu ve seyahat süresi fonksiyonu karakteristikleri ile dinamik seyahat tercihi probleminin bir eşitlik eniyileme programına sahip olmadığı gösterilmiştir. İççe köşegenleştirme bazlı bir çözüm algoritması önerilmesine rağmen modelin gerçek ağlar üzerinde uygulanması önemli ölçüde zorluklar içermektedir.

VI yaklaşımı, çeşitli DTA problemlerinin ele alınmasında diğer analitik yaklaşımlara göre daha uygun ve daha fazla analitik esneklik sağlamasından dolayı daha geneldir. Eş zamanlı ve ideal KD altında DTA problemleri için karşı karşıya kalınan seyahat süresi kavramını görece olarak tanımlamak için VI yaklaşımı kullanılmaktadır. Seyahat maliyeti fonksiyonları için asimetrik jakobian matrisleri ile senaryoların ele alınmasında matematiksel programlama yaklaşımlarının yetersizliğini vurgulamaktadır. Bağ etkileşimlerini daha iyi temsil edebilme yeteneklerine rağmen analitik modellerin kapsamında artan trafik akımının gerçekçi olarak temsili konusundaki problemler bulunmaktadır.

- *Benzetim Bazlı Modeller*

Mahmassani ile Peeta (1992, 1993, 1995) ve Peeta ile Mahmassani (1995a) yapmış oldukları çalışmalarda sabit varış zamanlı B-V talebine SO ve KD altında DYNASMART (Jayakrishnan vd, 1994) mesoskopik trafik modeli geliştirmişlerdir. Mesoskopik benzetim modeli, bireysel olarak taşıtları temsil eden mikroskopik seviye ile trafiğin hidrodinamikte olduğu gibi sıkıştırılabilir akışkanlar mantığına uygun olarak ifade edilmeye çalışıldığı makroskopik seviyeyi birleştirmektedir.

Ghali ve Smith (1992a) yapmış oldukları çalışmada, tıkanıklığın sadece şişe boyunlarında arttığı deterministik SO altında DTA problemi için tek bir kullanıcı sınıflı model önermişlerdir. Bağ marjinal seyahat maliyetleri kullanarak belirlenen rotalar üzerinde bireysel olarak rotalanan taşıtlar vasıtasıyla benzetim bazlı bir

çözüm yöntemi önermişlerdir. Yapılan yaklaşım, SO garanti etmemesine ve kuyruklanmayla ilgili yapılan kabullerden dolayı kısıtlı olmasına rağmen, analitik yöntemlerde kısıtlanan trafik akımının modellenme konularını ele almaktadır. Ghali ve Smith (1992b) yapmış oldukları diğer bir çalışmada global marjinal seyahat sürelerinin hesaplanmasına ilave olarak yaklaşık marjinal seyahat sürelerinin hesaplanmasının farklı seviyelerini tartışmışlardır. Marjinal seyahat sürelerinin hesabında kullanılan çözüm yöntemi hem gerçek zamanlı uygulamalar için hem de büyük ölçekli ağlarda etkin olan alternatif senaryoların benzetimi için uygun değildir. Ghali ve Smith (1992c), Smith (1994), SO ve KD altında DTA problemlerini ele almışlar ve CONTRAM benzetim modelini (Leonard vd 1989; Taylor 1990, 1996) kullanarak çözüm yöntemleri uygulamışlardır. Ghali ve Smith (1993), ulaşım ağlarında global en uygun çözümleri elde etmeyi ve yakınsamayı engelleyen DTA problemlerindeki konveks olmama ve türevlenememe ihtimalini tanımlamak için örnekler oluşturmuşlardır.

Peeta ve Mahmassani (1995b), ağ şartlarındaki değişimleri açıkça içeren ve eşzamanlı olarak işlenebilirliği mümkün kılan planlama yaklaşımı DTA modelleri geliştirmişlerdir. Bu yaklaşım, gelecekteki talep bilgisini gerektiren problemleri ele alan DTA problemleri için pratik yöntem sağlamaktadır. Bu yaklaşımın avantajı, mevcut bilgiyi, yakın dönemli tahminleri kullanarak gerçek zamanlı olarak problemi çözüme yeteneğidir.

Ben-Akiva vd (1997a, 1997b) gerçek zamanlı mevcut ve gelecek trafik şartlarını tahmin eden dinamik trafik atama sistemi olarak DynaMIT modeli önermişlerdir. DynaMIT, KD altında rota kılavuzu üretmek için birbiriyle etkileşen bir talep ve arz simülatöründen oluşmaktadır. Talep simülatörü, Kalman Filtreleme yöntemi kullanarak B-V taleplerini tahmin etmektedir. Talep simülatörü hem eski bilgiyi hemde sürücünün bilgiye verdiği cevabı dikkate almaktadır. Arz simülatörü, talebe dayalı olarak akım yapısını belirlemek için kullanılmaktadır. Arz simülatörü, taşıtların desteler halinde hareket ettiği ve trafik akımını modellemek için bağları hareketli ve kuyruklanan kısım olarak parçalara ayrıldığı mesoskopik bir trafik simülatörüdür.

Ziliaskopoulos ve Waller (2000), veri ve modelleri internet bazlı bir Coğrafi Bilgi Sisteminde birleştiren bir sistem önermişlerdir. Bu sistemdeki benzetim bazlı DTA

modeli, trafik yayılımı için Daganzo (1994) tarafından önerilen hücre iletim modeline dayanan mesoskopik RouteSim trafik simülatörünü kullanmaktadır. Bu model, ağ üzerindeki sinyalizasyon kontrol sistemlerini, zamana bağlı hücre kapasitelerini kullanarak doymuş akım değerlerini dikkate almaktadır.

Benzetim bazlı DTA modelleri, analitik yaklaşımlarda zor olan modelleme konularını kolaylıkla ele almaktadır. Trafikğin teorik ilişkilerini, trafik akımını modellemek için biraraya getiren benzetim modelinin kullanımı, dinamik trafik olgusunun tekrarlanmasında analitik bağ performans ve çıkış fonksiyonlarının kısıtlamalarını engellemekte ve bu ilişkilerle tutarlı akım değerleri elde edilmektedir. Trafik simülatörü karmaşık taşıt etkileşimlerini doğru ve düzgün olarak yansıtmakta, böylece doğrusal olmayan amaç fonksiyonlarının değerlendirilmesi tatmin edici olarak maliyet fonksiyonları ile kıyaslanmaktadır. Ayrıca, benzetim bazlı modeller İGİÇ kısıtını tam olarak karşılamakta ve taşıtların geri tutulmasından dolayı oluşan problem engellenmektedir. Tüm bu faktörler DTA kapsamında benzetim bazlı yaklaşımların avantajlarını göstermektedir. Bu yüzden, analitik modeller öncelikli olarak konunun teorik içyüzünü araştırmaya odaklanırken, benzetim bazlı modeller gerçek ulaşım ağları için pratik uygulamaları mümkün kılmaktadır. Benzetim bazlı modeller, gerçek ulaşım ağlarındaki farklı kullanıcı sınıfları arasındaki karmaşık etkileşimleri gerçekçi olarak modellemek için de uygun seçenektir.

Benzetim bazlı DTA modellerinin dezavantajlarından birisi genellikle ilgili matematiksel özellikleri yansıtamamasıdır. Daha önce de belirtildiği gibi, bu durum gerçek ulaşım ağları için pratik uygulamalarda önemli değildir. Simülatör kullanımının diğer bir dezavantajı planlanma yaklaşımı kapsamındadır. Geleceği tahmin etmek için iteratif mekanizmanın bir parçası olarak simülatör kullanımı sonucu oluşan sayısal yük işletmesel olarak sınırlayıcı olabilmektedir. Bu yüzden, birçok benzetim bazlı modelde çözümlerin hatasızlığı ile sayısal yük arasında ödünleşme olmaktadır.

2.4.2. SO ve KD Altında Benzetim Bazlı DTA Modeli

Bu bölümde benzetim bazlı DTA modellerinin, bir başka deyişle, SO ve KD altında zamana bağlı TA probleminin formülasyonu açıklanmıştır.

Değişkenler ve notasyonlar:

- i = başlangıç noktalarının alt simgesi, $i \in I$,
- j = varış noktalarının alt simgesi, $j \in J$,
- n = ulaşım ağı üzerindeki düğüm noktası, $n \in N$,
- a = ulaşım ağı üzerindeki bir bağıın simgesi, $a \in A$,
- k = ağ içinde i başlangıç noktasından başlayan seyahatin rotası için alt simge,
- τ = seyahat eden taşıtların harekete geçtiği zaman aralığını belirten üst simge, $\tau = 1, \dots, T$,
- t = o anki zaman aralığını belirten üst simge, $t = 1, \dots, T$,
- T' = atamaların yapıldığı zaman aralığı,
- Δ = zaman aralığının uzunluğu ($= T'/T$),
- r_{ij}^τ = τ harekete geçiş zaman aralığında i başlangıç noktasından j varış noktasına hareket eden taşıt sayısı,
- r_{ijk}^τ = τ harekete geçiş zaman aralığında i başlangıç noktasından j varış noktasına k rotası üzerine atanarak hareket eden taşıt sayısı,
- δ_{ijk}^{aa} = zamana bağlı bağı-rotalı belirleme elemanı, taşıtlar i başlangıç noktasından j varış noktasına τ harekete geçiş zamanında k rotası üzerine atanarak giderlerken t zaman aralığı içinde a bağıının üzerinde iseler 1'e eşittir,
- [$\delta_{ijk}^{aa} = 1$, r_{ijk}^τ t zaman aralığı boyunca a bağıının üzerindeyse,
= 0, a bağı k rotası üzerinde değilse,
= 0, $\tau > t$ ise,
= 0, t zaman aralığı boyunca r_{ijk}^τ a bağıının üzerinde değilse],
- T_{ijk}^τ = i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atanarak giden taşıtlar için karşı karşıya kalınan rota seyahat süresi,

$x_{ijk}^{\pi a}$ = i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atanarak giden, t zaman aralığının başında a bağının üzerinde olan taşıtların sayısı,

$d_{ijk}^{\pi a}$ = i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atanarak giden, t zaman aralığında a bağına giren taşıtların sayısı,

$m_{ijk}^{\pi a}$ = i başlangıç noktasından j varış noktasına τ zaman aralığında harekete geçmiş k rotası üzerine atanarak giden, t zaman aralığında a bağından çıkan taşıtların sayısı,

x^{ta} = t zaman aralığının başlangıcında a bağı üzerindeki toplam taşıt sayısı,

d^{ta} = t zaman aralığında a bağına giren toplam taşıt sayısı,

m^{ta} = t zaman aralığında a bağından çıkan toplam taşıt sayısı,

I_n^t = t zaman aralığında n düğüm noktasında üretilen taşıt sayısı,

O_n^t = t zaman aralığında n düğüm noktası üzerinden ağdan çıkan taşıt sayısı.

SO altında DTA model formülasyonu, her bir rotaya ve bu rotaları oluşturan bağlar üzerine atanan taşıt sayısı ile ilgili $\delta_{ijk}^{\pi a}$ zamana bağlı bağ rota belirleme elemanlarını birleştiren doğrusal olmayan tamsayıli formülasyondur. Zamana bağlı TA problem çözümünün temel zorluğu, zamana bağlı bağ rota belirleme değişkenlerinin atamanın bir fonksiyonu olmasıdır. Statik TA probleminde, akımlar rota üzerindeki bağlara rota boyunca eşzamanlı olarak atanmaktadır. Bu durum, sabit bağ rota belirleme matrisi ile temsil edilmektedir. Dinamik TA probleminde, bir rota boyunca harekete geçen taşıtlar sadece rota üzerindeki seyahat süresine bağlı olarak verilen zamanda belirli bir bağın üzerinde olacaktır.

SO altında DTA probleminin amaç fonksiyonu ve kısıtları Denklem (2.11)'de verildiği gibidir (Peeta, 1994; Peeta ve Mahmassani, 1995a).

$$\min \sum_{\tau} \sum_i \sum_j \sum_k (r_{ijk}^{\tau} \cdot T_{ijk}^{\tau}) \quad (2.11a)$$

Kısıtlar;

$$r_{ij}^{\tau} = \sum_k r_{ijk}^{\tau}, \quad \forall i, j, \tau, \quad (2.11b)$$

$$\sum_b d^{tb} = \sum_c m^{tc} + I_n^t - O_n^t, \quad \forall t, n, b \in B(n), c \in C(n), \quad (2.11c)$$

$$x^{ta} = x^{t-1a} + d^{t-1a} - m^{t-1a}, \quad \forall t, a, \quad (2.11d)$$

$$x^{ta} = \sum_k \sum_{\tau} \sum_i \sum_j (r_{ijk}^{\tau} \cdot \delta_{ijk}^{a\tau}), \quad \forall t, a, \quad (2.11e)$$

$$T_{ijk}^{\tau} = \sum_t \sum_a [\delta_{ijk}^{a\tau} \cdot \Delta], \quad \forall i, j, k, \tau, \quad (2.11f)$$

$$\delta_{ijk}^{a\tau} = F(r_{ijk}^{\tau}, \forall i, j, k, \tau, t, a), \quad (2.11g)$$

$$d^{ta} = \sum_k \sum_{\tau} \sum_i \sum_j d_{ijk}^{a\tau}, \quad \forall t, a, \quad (2.11h)$$

$$m^{ta} = \sum_k \sum_{\tau} \sum_i \sum_j m_{ijk}^{a\tau}, \quad \forall t, a, \quad (2.11i)$$

$$I_n^t = \sum_j r_{nj}^t, \quad \forall t, n \in I, \quad (2.11j)$$

$$O_n^t = \sum_k \sum_{\tau} \sum_i \sum_c m_{ink}^{a\tau}, \quad \forall t, n \in J, c \in C(n), \quad (2.11k)$$

$$\tau \leq t, \quad (2.11l)$$

$$\delta_{ijk}^{a\tau} = 0 \text{ veya } 1, \quad \forall i, j, k, \tau, t, a, \quad (2.11m)$$

$$\text{Tüm deęişkenler} \geq 0. \quad (2.11n)$$

Denklem (2.11a)'daki amaç fonksiyonu, her bir B-V çifti ve her bir harekete geçiř zamanı için, rotalara atanan tařıt sayısı ile bu tařıtların karřı karřıya kaldığı seyahat sürelerinin çarpımlarının toplamıdır. Amaç fonksiyonunun anlaşılır şekilde yorumlanması sayısal olarak zor olmaktadır. Denklem (2.11f) ve (2.11g) amaç

fonksiyonunun doğrusal olmayan yapısını yansıtmakta ve T_{ijk}^{τ} rota seyahat süresinin, r_{ijk}^{τ} atanan taşıt sayılarının açık olmayan fonksiyonu olarak tanımlamaktadır. Amaç fonksiyonunun bu yapısından dolayı, amaç fonksiyonunu değerlendirmek için benzetim modelleri kullanılmaktadır. Benzetim modelleri, amaç fonksiyonunun değerlendirilmesinde ve trafiğin modellenmesinde kullanılmaktadır. Böylelikle, analitik açıdan zorluğun ve dinamik özelliklerin üstesinden simülatör kullanımı ile gelinebilmektedir.

Denklem (2.11b)'deki kısıt, başlangıç noktalarındaki, $i \in I$, B-V taleplerinin korunumunu sağlamaktadır. Denklem (2.11c)'deki kısıt, düğüm noktalarındaki akım korunumunu belirtmektedir. Bu kısıtlar, düğüm noktalarında taşıtların birikemeyeceğini ve n düğüm noktasında herhangi bir t zamanında, bu düğüm noktasından tüm bağlara giren taşıt sayısı ile tüm bağlardan bu düğüm noktasına çıkan taşıt sayısının eşit olması gerektiğini açıklamaktadır. Denklem (2.11d)'deki kısıt, bağlar üzerindeki akım korunumunu belirtmektedir. t zaman aralığının başında bir a bağının üzerindeki taşıt sayısı, bir önceki, $(t-1)$, zaman aralığının başında bağ üzerindeki taşıt sayısının, yine bir önceki zaman aralığında bağa giren taşıt sayısının ve yine bir önceki zaman aralığında bağdan çıkan taşıt sayısının cebrik toplamıdır.

Denklem (2.11e), (2.11f) ve (2.11g)'deki kısıtlar, zamana bağlı bağ rota belirleme elemanlarını δ_{ijk}^{aa} biraraya getirmektedir. Denklem (2.11e)'deki kısıt, bir bağ üzerindeki taşıt sayısını, x^{ta} , rota üzerine atanan taşıt sayısı, r_{ijk}^{τ} , ile açıklamaktadır.

Denklem (2.11g)'de gösterildiği gibi, rota üzerine atanan taşıt sayılarının bağ rota belirleme değişkenlerine bağımlılığından dolayı doğrusal değildirler. Denklem (2.11f)'deki kısıt, bağ rota belirtme değişkenlerini kullanarak rota üzerindeki seyahat sürelerini tanımlamaktadır.

Denklem (2.11g)'deki kısıt, trafik akımını fiziksel olarak tanımlamaktadır. Bu kısıt, zamana bağlı bağ rota belirleme değişkenlerinin analiz süresi boyunca yapılan tüm atamaların, r_{ijk}^{τ} , bir fonksiyonu, $F(\cdot)$, olduğunu belirtmektedir. Bu kısıtlar, İGİÇ şartını sağlamakta, trafiğin geri tutulma olayını engellemekte, bağ etkileşimlerini ve diğer dinamik trafik olgularını tamamiyle karşılamaktadır. Buna rağmen, $F(\cdot)$ fonksiyonunu yeteri kadar açıklayabilecek analitik formülasyon bulunmamaktadır.

Benzetim bazlı modeller, $F(r_{ijk}^\tau)$ fonksiyonunu değerlendirmek için trafik simülatörü kullanırlar. Trafik simülatörü, İGİÇ şartını sağlarken ve düğüm noktalarında trafiğin geri tutulma olayını engellerken, ulaşım ağındaki dinamik trafik etkileşimlerini tekrarlamaktadır.

Denklem (2.11h) ve (2.11i)'deki kısıtlar, sırasıyla çeşitli zaman aralıklarında bağlara giren ve bağlardan çıkan taşıt sayılarını tanımlamaktadır. Denklem (2.11j) ve (2.11k)'deki kısıtlar, sırasıyla t zaman aralığında n düğüm noktasından ağa giren ve çıkan taşıt sayılarını tanımlamaktadır. Denklem (2.11l)'deki kısıt, atanan taşıtların harekete geçiş zaman aralıklarının en fazla mevcut t zaman aralığı kadar olabileceğini belirtmektedir. Denklem (2.11m)'deki kısıt, zamana bağlı bağ rota belirleme değişkenlerinin 0 ya da 1 değeri alabileceğini, Denklem (2.11n)'deki kısıt ise, negatif olmama kısıtını göstermektedir.

Statik KD problemi için yapılan Wardrop'un (1952) yaklaşımı zamana bağlı durum için genelleştirilebilir. Ancak, bu durumda kullanıcıların karşı karşıya kaldıkları rota seyahat sürelerinin dengelenmesi gerekmektedir. Mevcut KD altındaki DTA formülasyonları optimum kontrol teorisi yaklaşımını kullanmakta ve anlık seyahat sürelerinin dengesini kabul etmektedirler. Özellikle zirve saatlerde karşı karşıya kalınan seyahat süresi anlık seyahat sürelerinden çok farklı olduğundan anlık seyahat süresi bazlı KD anlamsız hale gelmektedir.

SO altındaki DTA probleminde amaç, analiz süresi boyunca toplam sistem seyahat süresinin enküçüklenmesidir. KD altındaki DTA problemindeki amaç, zamana bağlı KD şartının sağlanmasıdır. Zamana bağlı KD, Wardrop'un (1952) birinci şartının genişletilmesiyle elde edilmektedir:

1. Herhangi bir B-V, $(i-j)$, çiftini bağlayan tüm rotalara, $k^* \in K_{ij}$, herhangi bir τ zaman aralığında atanan taşıtlar aynı karşı karşıya kalınan rota seyahat süresine, T_{ijk}^τ , sahiptir (τ zaman aralığında i başlangıç noktasından j varış noktasına en küçük karşı karşıya kalınan seyahat süresine, $\theta_{ij}^{*\tau}$, eşittir).

2. Herhangi bir B-V çiftini bağlayan tüm rotalara, herhangi bir τ zaman aralığında atanmayan taşıtların seyahat süresi, karşı karşıya kalınan rota seyahat süresinden, $\theta_{ij}^{*\tau}$, büyük veya eşittir.

Bu şartlar Denklem (2.12) ve (2.13)'deki gibi ifade edilebilir:

$$r_{ijk}^{\tau} (T_{ijk}^{\tau} - \theta_{ij}^{*\tau}) = 0, \quad \forall i, j, k, \tau, \quad (2.12)$$

$$(T_{ijk}^{\tau} - \theta_{ij}^{*\tau}) \geq 0, \quad \forall i, j, k, \tau, \quad (2.13)$$

Sonuç olarak, kullanılan herhangi bir rota için, rota akımı $r_{ijk}^{\tau} > 0$ ve $(T_{ijk}^{\tau} - \theta_{ij}^{*\tau}) = 0$ 'dır. Kullanılmayan bir rota için, $r_{ijk}^{\tau} = 0$ ve $(T_{ijk}^{\tau} - \theta_{ij}^{*\tau}) \geq 0$ 'dır. Kısıtlar, SO altındaki DTA formülasyonunun kısıtları ile aynıdır (2.11b)-(2.11n).

2.4.3. Dynasmart-P Benzetim Bazlı Dinamik Trafik Atama Yazılımı

Dynasmart-P (**D**ynamic **N**etwork **A**ssignment-**S**imulation **M**odel for **A**dvanced **R**oadway **T**elematics **P**lanning version) yazılımı ilk olarak Maryland Üniversitesi tarafından geliştirilen ve şu anda Amerika Federal Otoyol İdaresi (FHWA) tarafından desteklenen benzetim bazlı DTA yazılımıdır. Dynasmart-P içerdiği senaryo menüsü ile kontrol parametreleri, ücretlendirme sistemleri gibi ağ performansını etkileyici unsurların ATK üzerindeki olası etkilerini ortaya koyabilmektedir. Programın en önemli özelliklerinden birisi DTA yapabilmesidir (Dynasmart-P, 2007).

Dynasmart-P, benzetim bazlı DTA özelliğiyle ulaşım ağı planlaması ve trafik yönetim kararlarını desteklemektedir. Bu yazılım, planlama uygulamaları için talep tahmin prosedürleri ile birlikte kullanılan DTA modellerini ve trafik yönetim çalışmaları için kullanılan trafik benzetim modellerini bir araya getirmektedir.

Dynasmart-P ağ kullanıcılarının seyahatleri esnasında rota seçimi kararları sonucu ulaşım ağı üzerinde oluşan trafik akımlarının değişimini dinamik olarak temsil edebilmektedir. Bu durum sayesinde planlama çalışmalarında kullanılan statik TA yazılımlarına ait dezavantajların üstesinden gelinebilmektedir.

Dynasmart-P, dört aşamalı ulaşım planları ile kolayca koordine olabilen yeni ulaşım planlama yöntemlerini tanımlayabilmektedir. Ayrıca, yazılım trafiğin zamanla değişken olması durumunu dikkate almaktadır. Dynasmart-P ile hız, kuyruk uzunluğu, gecikme ve tıkanıklık etkisi gibi durumların belirlenmesi ile farklı ulaşım

planlama stratejilerinin çevresel ve fonksiyonel etkilerinin değerlendirilmesi yapılabilmektedir. Dynasmart-P benzetim bazlı DTA yazılımı ile ilgili detaylı bilgi ve yazılımın altında çalışan algoritma yapıları Bölüm 4’de verilmektedir.

2.5. Trafik Sinyal Kontrolü

Şehir içi ulaşım ağlarında trafik akımlarının kesiştiği kısımlar olan kavşaklar genel olarak sinyalize ve sinyalize olmayan kavşaklar olarak ikiye ayrılmaktadır. Sinyalize kavşaklar, kontrolsüz kavşaklarda kontrolü sağlamak, meydana gelebilecek kazaları önlemek ve kavşaklarda meydana gelen gecikmeleri azaltmak amacı ile kullanılmaktadır.

Sinyalizasyon sistemleri kontrol ettikleri kavşakların durumuna göre ayrık ve eşzamanlı sistemler olmak üzere iki başlıkta incelenmektedir. Ayrık sinyalizasyon sistemleri yakınındaki diğer kavşaklarda kurulmuş bulunan başka sinyalizasyon sistemleri ile herhangi bir bağlantısı olmayan ve diğer sinyalize tesislerin etkilemediği sistemdir. Bu tür sistemlerde kavşaklar herhangi bir trafik sayımı ve modelleme çalışması yapılmadan işletilmektedir. Bu durum, kavşak kolları üzerindeki farklı trafik yüklerini dikkate almadan aynı tasarım parametrelerinin uygulanması sonucunu doğurmakta bu da gecikmelerin ve buna bağlı olarak trafikteki sıkışıklıkların artmasına neden olmaktadır.

Ana yolların birbirine çok yakın iki veya daha fazla sayıdaki kavşaklarında, gecikmeleri azaltmak ve sık sık duruş-kalkışları engellemek amacıyla, kavşaklardaki sinyalizasyon sistemlerinin birbirine bağlanması eşzamanlı sinyalizasyon sistemi olarak tanımlanabilmektedir. Eşzamanlı sistemler, genellikle anayol üzerindeki kavşaklardan, tali yol trafiğine de yeterli geçiş hakkı tanıyarak, birim zaman içinde mümkün olan en yüksek sayıda taşıtın durmadan geçirilmesi için düzenlenmektedir. Eşzamanlı sinyalizasyon sistemleri öncelikle anayol trafiği için uygulanmakla birlikte, bütün yönlerdeki toplam gecikmenin enküçüklenmesi için de uygulanabilmektedir. Bu şekilde sinyalizasyon sistemlerinin tümü arasında koordinasyonu sağlamak üzere en uygun sinyal parametrelerinin verilmesini düzenleyen eşzamanlı sinyalizasyon sistemlerine Alansal Trafik Kontrolü (ATK) adı verilmektedir (Murat, 1996).

Sinyalizasyon uygulamaları için sıkça kullanılan bazı temel trafik ve sinyalizasyon terimleri ařařıdaki gibi özetlenebilir (Ceylan, 2002):

Trafik akımı: Bir kavřađın yaklařım kollarını ve bu yaklařım kolları üzerindeki řeritleri kullanan tařıt dizileri trafik akımı olarak adlandırılabilir. Dolayısıyla, bir kavřađın yaklařım kolu bir veya daha fazla trafik akımından oluřabilmektedir.

Yaklařım: Tařıtların bir kavřađa girmek için kullandıđı řerit grubudur.

Sinyal grubu: Her zaman aynı durumdaki sinyalleri gösteren sinyallerin grubudur.

Faz: Bir veya daha fazla trafik akımına geçiř önceliđi verilirken bunlarla çatıřan bazı akımların durdurulduđu zaman dilimine faz adı verilmektedir.

Devre süresi: Kavřaktaki her trafik akımına en az bir kez geçme hakkı verilen toplam zaman dilimine verilen addır. Bir bařka deyiřle, fazların bir devir yapmaları için gerekli toplam süreye verilen addır.

Faz düzeni: Devre süresi içindeki fazların sırasıdır.

Doğun akım: Kavřak yaklařım kolunda kırmızı ıřıkta kuyrukta bekleyen tařıtlar, yeřil ıřık yandıđı zaman hızla harekete geçerler ve sabit bir orana ulařıncaya kadar harekete devam ederler. Tařıtların kavřađa giriř yaptıđı bu sabit oran doğun akım olarak adlandırılır. Sinyalize bir kavřakta sürekli bir kuyruk bulunması halinde ve kesintisiz geçiř hakkı kořulları altında sabit bir hızla bořalan akım oranıdır. Bu oran, sinyalize bir kavřaktan bir saatte bir řeritten geçebilen en büyük tařıt sayısını yansıtmaktadır.

Bařlangıç kaybı: Kırmızı ıřıkta kuyrukta bekleyen tařıtların, ıřık yeřile döndüđu zaman bařlangıçta hareketleri yavařtır ve tařıtlar sabit bir oranda harekete geçinceye kadar ilk harekete geçiřlerinden dolayı bir kayıp söz konusu olmaktadır. Bu kayıp süre bařlangıç kaybıdır.

Bitiř kazancı: Sinyalize bir kavřakta yeřil ıřık alan akımda doğun akıma ulařıldıktan sonra yeřil ıřığın sönmesi ile sarı ıřığın yanması arasındaki zaman diliminde kavřakta hala hareket eden tařıtlar bulunmaktadır. Bu zaman dilimine bitiř kazancı denmektedir.

Etkin yeřil süre: Görünen yeřil süreden bařlangıç kaybının çıkarılıp bitiř kazancının eklenmesi ile elde yeřil süredir.

Kayıp zaman: Sinyalize bir kavşakta, kavşağın herhangi bir akım tarafından etkin şekilde kullanılmadığı süreye kayıp zaman adı verilir. Kayıp zaman, her fazın başındaki başlangıç kaybından, sarı süre ve tüm akımların kırmızı aldığı süreden oluşmaktadır.

Yeşillerarası süre: Bir fazdaki yeşil sürenin bitişi ile takip eden fazdaki yeşil sürenin başlangıcı arasındaki süreye yeşillerarası süre denir.

Yeşil süre oranı: Yeşil sürenin devre süresine oranıdır.

Kapasite: Sinyalize bir kavşakta mevcut sinyal süreleri için bir saatte bir şeritten geçebilen en büyük taşıt sayısıdır. Şerit kapasitesi; doymuş akım ve yeşil süre oranının çarpımıdır.

Doymuşluk derecesi: Trafik hacminin kapasiteye oranı doymuşluk derecesidir.

Akım oranı: Trafik hacminin doymuş akıma oranıdır.

Ofset: Eşzamanlı olarak işletilen iki sinyalize kavşakta aynı yöne gitmekte olan trafik akımına verilecek olan yeşil sürelerin başlangıçları arasındaki süredir.

Trafik sinyal kontrolü, gecikme, kuyruklanma, hava kirliliği, yakıt tüketimi ve trafiğin yayılımını içine alan çok amaçlı eniyileme problemi olup ve ağ performans indeksi (*Network Performance Index*) içinde birleştirilebilir (Robertson, 1969). Yeşil süre, devre süresi, faz sırası ve offset gibi değişkenler sinyal kontrol parametrelerini oluşturmaktadır. Şehir içi trafik yönetiminde en uygun sinyal kontrol parametrelerinin belirlenerek sinyalize kavşaklardaki gecikmelerin azaltılması oldukça önemlidir. Ayrık sinyalize kavşak için sinyal sürelerinin eniyilenmesi görece olarak basit olmasına karşın kavşaklar arası mesafelerin çok küçük olduğu yoğun ağlarda trafik kümelerinin dağılması için sinyal sürelerinin eniyilenmesi oldukça zor olmaktadır. Sinyalizasyonların koordinasyonu ile bu zorluğun üstesinden gelinebilmektedir. TRANSYT sinyal sürelerinin eniyilenmesi için kullanılan en yararlı yazılımlardan birisidir ve o alanda en yaygın kullanılan programdır.

2.5.1. TRANSYT-7F Trafik Modeli

TRANSYT (TRAffic Network StudY Tool) İngiltere Ulaşım ve Yol Araştırma Laboratuvarı (TRRL) tarafından geliştirilmiştir (Robertson, 1969). TRANSYT-7F faz bazlı en iyileme yazılımıdır. TRANSYT-7F'in başlıca özelliği; döngüsel akım

profili ve küme dağılım modelidir. TRANSYT-7F, trafik akım modeli ve sinyal süresi eniyileyici olmak üzere iki ana kısımdan oluşmaktadır. TRANSYT-7F'in içindeki trafik modeli, deterministik, makroskopik ve zaman taramalı benzetim modelidir. Sinyalize kavşaklı bir ağda girilen sinyal süre ve faz planlarına bağlı olarak Performans İndeksi (*PI*) değerini hesaplamak için trafiği simule etmektedir. TRANSYT-7F'deki *PI*, sinyalize kontrollü tüm akımların döngüsel akım profili ile değerlendirildiği, birim zamandaki duruş sayıları ve tahmin edilen gecikmenin ağırlıklandırılmış toplamıdır. TRANSYT-7F yazılımının amacı, koordine yol ağı için sinyal parametrelerinin tekrarlı olarak ayarlanması ve sadece sinyal parametrelerindeki değişimler ile *PI* değerini eniyileyen en uygun sinyal parametrelerinin bulunmasıdır.

TRANSYT-7F'de benzetim, bağ ve düğüm noktaları ile temsil edilen ağ içindeki trafik akımının gösterilmesi ile yapılmaktadır. Sinyalize kontrollü bir yol ağındaki her bir trafik akımı *PI* ile ilgili olarak, kendisine ait bağ ve düğüm noktası ile temsil edilmektedir.

TRANSYT-7F yazılımı, *PI* olarak adlandırılan amaç fonksiyonunu seçime bağlı olarak enbüyüklemekte yada enküçüklemektedir. TRANSYT-7F içinde birden fazla amaç fonksiyonu bulunmakta ve amaç fonksiyonu kullanıcı tarafından seçilmektedir. Amaç fonksiyonlarından biri standart TRANSYT-7F yararsızlık indeksi (*Disutility Index-DI*) değeridir. *DI* değeri ağın işletim dezavantajının bir ölçütü olup, ağdaki gecikmelerin, duruş sayılarının toplamıdır (McTrans, 2008). *DI* değerinin formülasyonu Denklem (2.14) verilmektedir.

$$DI = \sum_{a=1}^L (w_{d_a} d_a + K w_{s_a} S_a) \quad (2.14)$$

Burada, L bağ sayısını, d_a tş-sa/sa cinsinden a bağı üzerindeki gecikme değeri, K duruş ceza faktörü, S_a bir saniyede a bağı üzerindeki taşıt-duruş sayısı, w_{d_a} ve w_{s_a} sırasıyla gecikme değeri ve duruş sayısının ağırlık katsayılarıdır. TRANSYT-7F'de amaç fonksiyonu olarak standart *DI* seçilirse program amaç fonksiyonunu enküçükleyecektir.

DI değerinde, birim zamandaki gecikme ve duruş sayısı, trafik sinyallerinin döngüsel doğasının kabulü ile ilgili olarak iki bileşene ayrılmaktadır: (1) Yeşil sürenin

başındaki kuyruklanma ve trafik akımlarındaki değişimden dolayı oluşan gecikme ve duruş sayılarının rasgele ve doygun üstü bileşeni; (2) trafik sinyallerinin değişiminden dolayı oluşan gecikme ve duruş sayılarının üniform bileşeni.

TRANSYT-7F trafik modeli, taşıtlar akım aşağı doğru seyahat ederken kümelerin normal dağılımını simüle eden küme dağılım algoritmasından yararlanmaktadır. Ayrıca, gecikme, duruş sayısı, yakıt tüketimi, seyahat süresi ve diğer sistem etkinlik ölçütlerini de göz önüne almaktadır.

TRANSYT-7F modeli üç şekilde trafiği temsil eder:

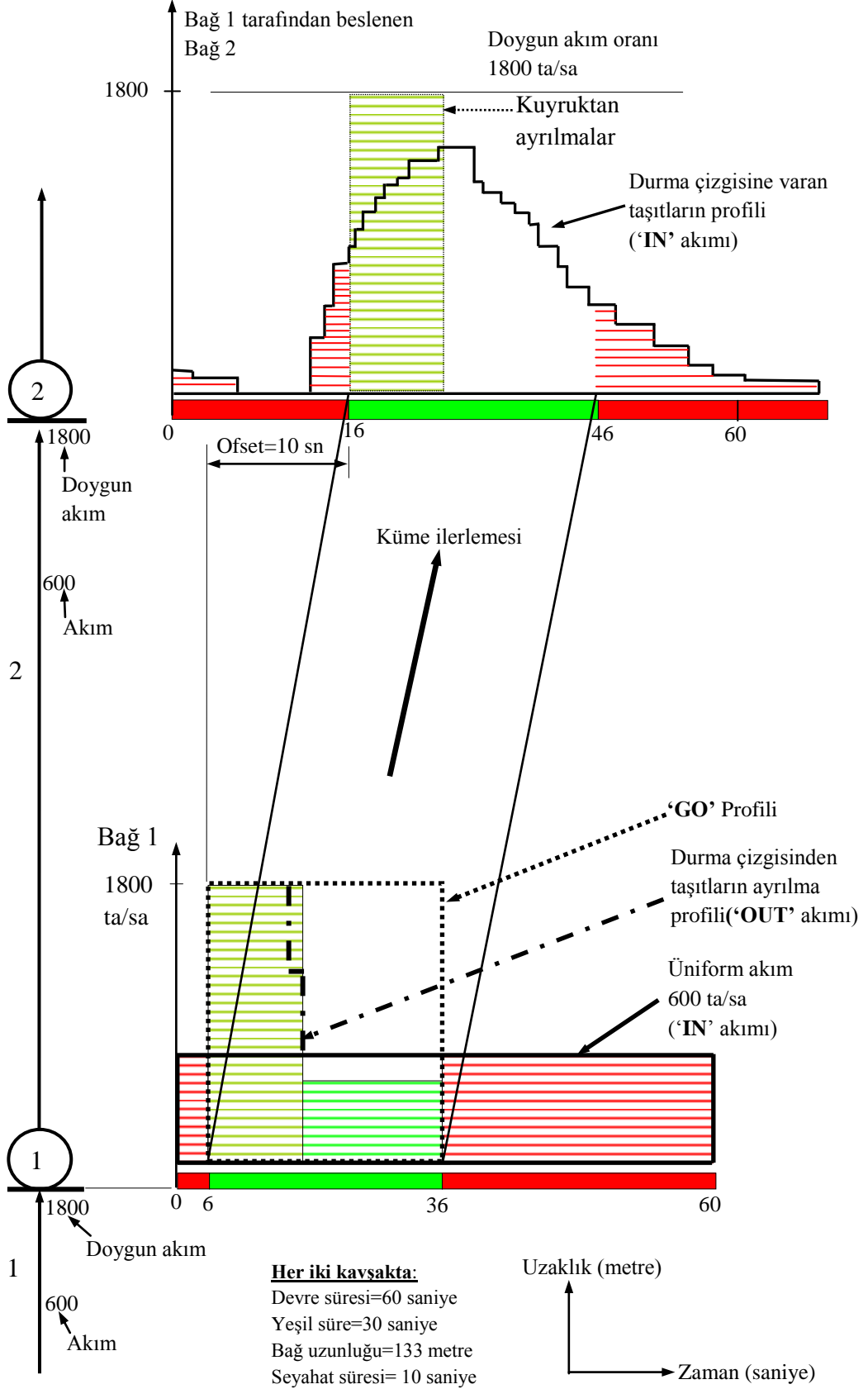
1. Gelen akım yapısı (IN yapısı): Sinyallerdeki durma çizgisinde trafik durmuyorsa, bağın akım aşağısı sonundaki durma çizgisine varan trafiğin yapısıdır.
2. Doygun akım yapısı (GO yapısı): Yeşil süreyi doyuracak kadar yeterli trafik olduğunda durma çizgisinden ayrılan trafiğin yapısıdır.
3. Ayrılan akım yapısı (OUT yapısı): Bağdan ayrılan trafiğin yapısıdır ve IN ve GO profillerinden türetilmektedir.

Şekil 2.2’de bir çift bağ üzerinde, gelen akım, doygun akım ve sinyal süreleri basitleştirilmiş olarak gösterilmektedir. Bağ 1 için gelen (IN) ve ayrılan (OUT) akım yapılarının düzenli olduğu kabulü yapılmaktadır. Şekil 2.2’de, bağ 2’ye giren taşıt kümeleri bağ 1’den boşalan taşıt kümeleridir. Bağ 2’nin durma çizgisine gelen akım yapısı, küme içindeki farklı hızlardaki taşıtlara uygulanan küme yayılım sürecine göre ayarlanmış olan bağ 1’in ayrılan akım yapısına göre belirlenmektedir. Şekil 2.2’de bağ 2’ye ait gelen taşıt kümesinin akım yapısı, bağ 1’e ait düzenli gelen (IN)/ayrılan (OUT) akım yapısından üretildiği için döngüsel karakterdedir. Bağın akım aşağısına gelen (IN) akım yapısı, bağın akım yukarısında ayrılan (OUT) akım yapıları ile belirlenmektedir (McTrans, 2008).

Her k zaman aralığı için bağın akım aşağısına gelen akım yapısı Denklem (2.15)’de verilen tekrarlamalı ifade ile elde edilmektedir.

$$v'_{(k+t)} = F \cdot v_k + [(1 - F) \cdot v'_{(k+t-1)}] \quad (2.15)$$

Burada, $v'_{(k+t)}$ k zaman aralığında bağın akım aşağısına gelen akım oranı, v_k k



Şekil 2.2 Taşıtların kümeleri, sinyal ofsetleri ile bağ üzerindeki trafiğin davranışı (Ceylan, 2002)

zaman aralığında gelen (IN) akım yapısı, t her zaman aralığında küme yayılım sürecinin dikkate alındığı süre olup $t = 0.8T$ 'dir. T her zaman aralığında bağ üzerindeki gecikme olmaksızın ortalama seyahat süresi, F düzeltme faktörüdür. F düzeltme faktörü Denklem (2.16)'da verilen formülasyon ile elde edilmektedir.

$$F = \frac{1}{1 + 0.35t} \quad (2.16)$$

Bir bağın akım yukarısındaki IN profili biliniyorsa, bağ boyunca küme yayılımı kabulü yapılarak akım aşağısındaki duruş çizgisinde OUT profili belirlenebilmektedir. GO profili, duruş çizgisindeki yeşil süre ve doygun akım oranı bilgisi ile elde edilebilmektedir. Dikey kümeleme kabulü ile, duruş çizgisinde kuyruklanmış taşıt sayısı ve buna bağlı olarak duruş sayıları ve ortalama gecikme değerleri başlangıç sinyal süreleri için belirlenebilmektedir. Daha sonra, elde edilen GO profili ile bağın akım aşağısındaki IN profili belirlenmekte ve süreç bu şekilde işlemektedir. Böylelikle, ağ üzerindeki her bir bağ için ortalama gecikme değerleri ve duruş sayılarının tahmini ile mevcut sinyal süreleri için PI değeri elde edilmektedir.

TRANSYT-7F gecikme hesabında, Highway Capacity Manual (TRB, 2000) gecikme modelini kullanmaktadır. TRANSYT-7F, gecikme hesabında, karmaşık trafik işleyişlerini dikkate almaya izin veren bu model ile makroskopik benzetim sonuçlarını kullanmaktadır. HCM (TRB, 2000) gecikme modeli, başlangıç yavaşlama gecikmesini, kuyruk hareket zamanını, durma gecikmesini ve son hızlanma gecikmesini içermektedir. HCM (TRB, 2000) gecikme modeli Denklem (2.17) ile verilmiştir.

$$d = d_1 PF + d_2 + d_3 \quad (2.17)$$

Burada, d taşıt başına kontrol gecikmesi (sn/ta), d_1 üniform kontrol gecikmesi (sn/ta), PF sinyal koordine düzeltme faktörü, d_2 rastgele varışların ve doygun üstü akımdaki kuyrukların etkisini açıklamak için eklenik gecikme değeri (sn/ta), d_3 analiz süresi başlangıcındaki başlangıç kuyruğundan dolayı tüm taşıtlara etkileyen gecikme değeridir (sn/ta). İyi bir koordine sinyal sistemi, yeşilde geçen taşıtların sayısını artırmaktadır. Sinyal koordine düzeltme faktörü, yarı etkileşimli kontrol sistemlerinde, etkileşimsiz şerit grupları ile sabit zaman kontrollü tüm koordine şerit gruplarına uygulanmaktadır. Etkileşimli şerit grupları için koordine kontrolün

sağlandığı durumlarda da, bu şerit gruplarına PF uygulanabilir. Koordine sistem, öncelikle üniform gecikmeyi etkileyeceğinden düzeltme sadece d_j 'e uygulanmaktadır. PF sinyal koordine düzeltme faktörü Denklem (2.18) ile verilmiştir.

$$PF = \frac{(1-P)f_{PA}}{1 - \frac{\varphi}{c}} \quad (2.18)$$

Burada, P yeşilde geçen taşıtların oranı, $\frac{\varphi}{c}$ yeşil sürenin devre süresine oranı, f_{PA} yeşil süre sırasında gelen taşıtlar için tamamlayıcı düzeltme katsayısıdır.

P 'nin değeri yerinde ölçümlerle veya geliş tipine bağlı olarak belirlenmektedir. Yerinde ölçümlerle bulunacaksa, yeşil fazda iken, durma çizgisine gelen ya da kuyruk varsa kuyruğa eklenen taşıtların oranı olarak alınmalıdır. P ve f_{PA} değerlerine bağlı olarak varış tipinin bir fonksiyonu olarak PF değerlerinin belirlenmesi için Tablo 2.1 kullanılabilir. Koordine olmayan şerit gruplarında varış tipi 3, koordine olan şerit gruplarında ise varış tipi 4'ün kullanılması tavsiye edilmektedir (TRB, 2000).

Tablo 2.1 Üniform gecikme hesabı için koordine sinyal düzeltme faktörü (PF)

Yeşil oran ($\frac{\varphi}{c}$)	Varış Tipi (VT)					
	VT 1	VT 2	VT 3	VT 4	VT 5	VT 6
0.20	1.167	1.007	1.000	1.000	0.833	0.750
0.30	1.286	1.063	1.000	0.986	0.714	0.571
0.40	1.445	1.136	1.000	0.895	0.555	0.333
0.50	1.667	1.240	1.000	0.767	0.333	0.000
0.60	2.001	1.395	1.000	0.576	0.000	0.000
0.70	2.556	1.653	1.000	0.256	0.000	0.000
f_{PA}	1.00	0.93	1.00	1.15	1.00	1.00
R_p	0.333	0.667	1.000	1.333	1.667	2.000

Üniform kontrol gecikmesi Denklem (2.19)'da verilen formülasyon ile hesaplanmaktadır. Denklem (2.19)'daki gecikme tahmin ifadesi, üniform gelişler, stabil akım ve başlangıçta kuyruk olmaması durumu içindir. Webster'in gecikme formülünün birinci terimine dayanır ve üniform varışların olduğu ideal durum için gecikmenin bulunuşunu ifade eder.

$$d_1 = \frac{0.5c \left(1 - \frac{\varphi}{c}\right)^2}{1 - \left[\min(1, X) \frac{\varphi}{c}\right]} \quad (2.19)$$

Burada, c devre süresi (sn), φ yeşil süre (sn), X doygunluk derecesidir.

d_3 ifadesi analiz süresi başlangıcındaki başlangıç kuyruğundan dolayı tüm taşıtlara etkileyen gecikme değerini belirtmektedir. Analiz süresi başlangıcında kuyruklanma yoksa d_3 ifadesi sıfır olmaktadır. Analiz süresi başlangıcında kuyruklanma varsa bu kuyruklanmanın neden olduğu gecikme Denklem (2.20) ile hesaplanmaktadır.

$$d_3 = \frac{1800Q_b(1+u)t}{CT} \quad (2.20)$$

Burada, Q_b analiz süresi başlangıcında kuyrukta olan taşıt sayısı (ta), C şerit grubu kapasitesi (ta/sa), T analiz süresi (sa), t analiz süresi içinde karşılanamamış talebin süresi (sa) ve u gecikme parametresidir. t ve u parametreleri sırasıyla Denklem (2.21) ve (2.22) ile bulunmaktadır (TRB, 2000).

$$Q_b = 0 \text{ ise } t = 0, \text{ aksi takdirde } t = \min \left\{ T, \frac{Q_b}{C[1 - \min(1, X)]} \right\} \quad (2.21)$$

$$t < T \text{ ise } u = 0, \text{ aksi takdirde } u = 1 - \frac{CT}{Q_b[1 - \min(1, X)]} \quad (2.22)$$

Burada, X doygunluk derecesidir.

TRANSYT-7F yazılımı üniform gecikme değerini simüle edilmiş kuyruk profili ile elde ettiği için PF düzeltme faktörünü uygulamasına gerek bulunmamaktadır. Ayrıca, d_3 gecikme değeri, TRANSYT-7F benzetim modelinin adım özelliği ile bir devre süresinden diğer devre süresine taşındığı için üniform gecikme değeri içinde ele alınmaktadır. Diğer bir deyişle, TRANSYT-7F üniform gecikme değeri, otomatik olarak PF düzeltme faktörü ve d_3 gecikme değeri etkilerini barındırmaktadır. TRANSYT-7F yazılımı, HCM (TRB, 2000) gecikme modeli ikinci gecikme değerinin hesabı için Denklem (2.23) formülasyonunu kullanmaktadır.

$$d_2 = 900T \left\{ (X - 1) + \sqrt{(X - 1)^2 + \frac{8kIX}{CT}} \right\} \quad (2.23)$$

Burada, T analiz periyodu süresi (sa), k eklenik gecikme faktörü, I ölçüm düzeltme faktörü, C şerit grubu kapasitesi (ta/sa), X doygunluk derecesidir. Eklenik gecikme faktörü, uyarmalı olmayan veya sabit zamanlı sinyalizasyon sistemlerinde 0.5 alınmaktadır. Ölçüm düzeltme faktörü ise izole sinyalizasyon sistemlerinde 1.0 olarak alınmakta izole olmayan sinyalizasyon sistemlerinde ise Denklem (2.24) formülasyonu ile belirlenmektedir.

$$I = 1.0 - 0.91X_u^{2.68} \quad (2.24)$$

Burada, X_u akım yukarı kavşaktaki doygunluk derecesini temsil etmekte ve bu değerlere göre ölçüm düzeltme faktörleri Tablo 2.2’de verilmektedir.

Tablo 2.2 Ölçüm düzeltme faktörleri

Akım yukarı kavşaktaki doygunluk derecesi, X_u							
	0.40	0.50	0.60	0.70	0.80	0.90	≥ 1.0
I	0.922	0.858	0.769	0.650	0.500	0.314	0.090

TRANSYT-7F modelinde duruş sayısı iki bileşene ayrılmaktadır: (1) üniform, (2) rasgele duruşlar. Kuyruk sonundaki taşıtlar harekete başlar başlamaz arkadan gelen taşıtlar tamamen durmamaktadırlar. Arkadan gelen taşıtlar durmadan sadece hızlarında değişikliklerle hareketlerine devam etmektedirler. TRANSYT-7F bu tip taşıtların kısmi olarak durduğunu kabul ederek duruş sayılarını hesaplamaktadır. TRANSYT-7F’de duruş çizgisinden ayrılan taşıt sayısı ile duran taşıt yüzdesinin çarpımı sonucu duruş sayısı hesaplanmaktadır. Bağ hacmi, bağ kapasitesini aştığında, devre süresinin yeşil süresinde gelen taşıtların hepsi ayrılamamaktadır. Dolayısıyla yukarıda bahsedilen hesap ile yeşilde geçemeyen taşıtlar dikkate alınmamaktadır. TRANSYT-7F’de bu taşıtların tamamen durduğu kabulü yapılmaktadır. Doymuş akım koşullarındaki toplam duruş sayısı, yukarıda bahsedilen hesap ile bulunan duruş sayısı ile tamamen durduğu kabulü yapılan yeşilde geçememiş taşıtların sayısının toplamı olmaktadır. TRANSYT-7F, saat başına rasgele ve doymuş duruş sayılarını Avustralya Yol Araştırma Dairesi (ARRB) tarafından geliştirilen Denklem (2.25) ile formülasyonu verilen model ile hesaplanmaktadır.

$$h = \frac{3240N_o}{c} \quad (2.25)$$

Burada, h saat başına rasgele ve doygunüstü duruş sayısı , c devre süresi (sn) ve N_o rasgele ve doygunüstü kuyruk uzunluğudur. Rasgele ve doygunüstü kuyruk uzunluğu, N_o , Denklem (2.26) ile hesaplanabilir.

$$N_o = 0.25cTX^2 \left\{ (X-1) + \sqrt{(X-1)^2 + \frac{MX}{CT}} \right\} \quad (2.26)$$

Burada, C şerit grubu kapasitesi (ta/sa), T analiz periyodu süresi (sa), X doygunluk derecesi, M taşıt sayısı cinsinden kuyruk uzunluğudur.

TRANSYT-7F yazılımı ile izole kavşak sinyal sürelerinin (yeşil sürelerin) eniyilenmesi ve tüm ağ sinyal sürelerinin (kavşaklar arasındaki ofsetler dahil olmak üzere) eniyilenmesi yapılabilmektedir. TRANSYT-7F yazılımı trafik sinyal parametrelerinin eniyilenmesinde Hill-Climbing (HC) veya Genetic Algorithms (GA) tekniklerini kullanmaktadır.

2.5.2. Sinyal Kontrolü ile İlgili Yapılan Çalışmalar

Tek bir kavşak ya da ağ için trafik sinyal parametreleri matematiksel programlama teknikleri ile değerlendirilebilmektedir. Bununla ilgili yöntemler ikiye ayrılmaktadır: Faz bazlı yaklaşım ve grup bazlı yaklaşım. Faz bazlı yaklaşımda fazların sırası, durumları başlangıçta sabit olduğu kabulü yapılmakta ve kavşaklar arasındaki ofset ve fazların yeşil süreleri PI değerini eniyileyecek şekilde hesaplanmaktadır. TRANSYT-7F yazılımı faz bazlı yaklaşımı kullanmaktadır. Grup bazlı yaklaşımda ise her bir sinyal grubunun yeşil süresinin ve birbirini izleyen grupların yeşil sürelerinin eniyilenmesi, doğrudan gruplar arasındaki uyumsuzluk bilgisi (Clearence matrix) kullanılarak yapılmaktadır.

Trafik sinyal sürelerinin eniyilenmesi iki seviyede yapılmaktadır: Tek bir kavşak sinyal süresi eniyilenmesi (yeşil süreler) ve ağ eniyilenmesi (birbirine komşu kavşaklar arasındaki ofset süreleri). Her iki seviyedeki eniyileme hem faz bazlı hem de grup bazlı yaklaşım ile yapılabilmektedir. Grup bazlı yaklaşım kullanılarak sinyal parametrelerinin eniyilenmesinde, sinyal gruplarının sinyal kontrol parametreleri ile ilgilenilmekte, sırasıyla sinyal gruplarının sinyal parametreleri eş zamanlı olarak değiştirilmektedir. Karşılıklı olarak birbiri ile uyumlu gruplar eş zamanlı olarak yeşil süre alabilmektedir. Diğer yandan, karşılıklı olarak birbiri ile uyumsuz olan gruplar

arasında enküçük yeşiller arası süre bulunmaktadır. Ayrık sinyal kontrollü kavşaklarda, grup bazlı yaklaşımın faz bazlı yaklaşıma göre sistem performansında oldukça büyük avantajlar elde edilebildiği gösterilmiştir (Silcock ve Sang, 1990; Silcock, 1992; Chiou, 1998). Bir yol ağı için grup bazlı yaklaşımda sinyal kontrol parametreleri: Fazların uzunluğu, fazların sırası, fazların yapıları, ofsetler ve ortak devre süresidir. Kontrol parametreleri eniyileme süreci içinde belirlenebilmektedir (Gallivan ve Heydecker, 1988; Allsop,1992).

Heydecker (1996) çalışmasında sinyal sürelerinin eniyilenmesini iki seviyede birleştirmiştir. Birinci seviyede, grup bazlı yaklaşım kullanarak ayrık kavşaklardaki sinyal süreleri eniyilenmiş, ikinci seviyede ise ağ için ofset ve ortak devre süresi gibi değişkenler ayrık kavşaklardan gelen sonuçlar birleştirilerek elde edilmiştir.

Wong (1996), TRANSYT trafik modeli kullanarak sinyal sürelerinin eniyilenmesi için grup bazlı yaklaşımı kullanan bir model önermiştir. Bu modelde, grup fazlı sinyal kontrol parametrelerini lineer olmayan matematiksel program olarak formüle edilmiştir. Matematiksel programın amaç fonksiyonu TRANSYT’de olduğu gibi birim zamanda duruş sayısı ve gecikmenin ağırlandırılmış lineer bir kombinasyonu olarak tanımlanmıştır. Amaç fonksiyonu, faz bazlı kontrol parametreleri veya grup bazlı kontrol parametrelerine göre enküçüklenebilmektedir.

ATK problemi faz bazlı eniyileme veya grup bazlı eniyileme yaklaşımının kullanımına göre formüle edilebilmektedir. Amaç fonksiyonu ve kısıtları Denklem (2.27)’deki gibidir (Ceylan, 2002).

$$\underset{\psi}{\text{Min}} PI(\psi) \quad (2.27)$$

Kısıtına bağlı olarak

$$A(\psi) \geq 0$$

Burada, **A** kısıtların kümesini göstermekte ve kısıtların formu sinyal sürelerinin özelliğine bağlıdır (Allsop, 1992). ψ uygun sinyal süreleri vektörüdür.

Farklı şartlar altında trafik sinyal parametrelerini eniyilemek için çeşitli matematiksel yöntemler önerilmesine rağmen özellikle son on yılda ATK probleminin çözümü için farklı sezgisel yöntemlerin uygulanmaya başlaması bu alanda önemli bir adım olarak görülmektedir. Lee (1998) çalışmasında, GA ve TB’nin iteratif ve yerel araştırma

algoritmaları ile karşılaştırmasını vermiş ve farklı arz - talep senaryoları için farklı algoritmaların daha iyi performans gösterdiğini sunmuştur. Chiou (1999), denge bağ akımları ile sınırlı ATK problemini çözmek için tamsayıli araştırma yöntemi önermiştir. Önerilen yöntemde, en uygun çözüm gradyan projeksiyon yöntemi ile etkin şekilde bulunabilmektedir.

Ağ bağ akımları ile ATK probleminin çözümü için birleşik bir eniyileme yöntemi Chiou (2005) tarafından araştırılmıştır. Bu birleşik eniyileme yöntemini formüle etmek için iki seviyeli programlama yöntemi kullanılmaktadır. Burada, Wardrop'un denge prensiplerini sağlayan ağ bağ akımları TA probleminin çözümü ile elde edilebilmektedir. KD ve SO altındaki bağ akımları ile birleşik eniyileme yöntemi sonucu elde edilen PI değerleri için karşılaştırmalar yapılmıştır. Ceylan (2006), GA ile TRANSYT HC eniyileme aracı ile sabit trafik hacimleri için GATHIC yöntemini geliştirmiş ve ATK probleminde sinyal sürelerini eniyilemek için çözüm uzayını daraltan bir yöntem önermiştir. Elde edilen sonuçların TRANSYT ile karşılaştırılması sonucunda, en uygun sinyal süreleri ve *PI* değerleri açısından GATHIC modelinin sinyal süre eniyilenmesinde daha iyi olduğu belirtilmiştir. Teklu vd (2007), kentsel bir yol ağındaki devre sürelerinin ve yeşil sürelerin eniyilenmesi üzerine çalışmışlardır. Çalışmalarında, problemi çözmek için GA yaklaşımı önermişlerdir. Çalışmanın sonuçlarında GA ile eniyilenen yöntemin daha iyi performans gösterdiği belirtilmiştir. Putha vd (2001), KKO ve GA yaklaşımlarını kullanarak doyunüstü durum için ağ sinyal koordinasyon problemini çözmüşlerdir. KKO yönteminin fazla sayıdaki denemeler için devamlı olarak daha etkin olduğu ve daha güvenilir sonuçlar sağladığı belirtilmiştir.

2.6. Pekiştirmeli Öğrenme Yöntemi

Bir çocuğun bisiklet sürmeyi öğrenmesi gibi doğal bazı süreçlerde öğrenme olgusu öğreticisiz öğrenme şeklindedir ve istenen değer için hâlihazırda bir bilgi yoktur. Geçmiş deneyimlere göre daha iyi veya daha kötü diye nitelendirilebilecek davranışların kusursuzlaştırılmasıyla istenen değerlerin elde edilmesi sürecinde ödül/ceza stratejisi uygulayan Pekiştirmeli Öğrenme (PÖ) yöntemleri pek çok problemin çözümüne doğal yöntemlerden esinlenen çözümler önerir.

PÖ, ortamda herhangi bir öğretici olmasını gerektirmeyen, sistemin dış dünya ile olan etkileşimini kullanan bir öğrenme yöntemidir. PÖ'de öğrenen ve aktif öğrenme süreci olmak üzere iki nesne vardır. Her bir zaman diliminde öğrenci o anki durumunu değerlendirerek bir eylem seçer ve uygular. Ortamdan aldığı geri beslemeyi sisteme uygular. Öğrenenin amacı herhangi bir durumda en iyi davranışı seçebilecek duruma gelmektir. Ortamdan alınan geri besleme sinyalleri herhangi bir amacın başarılması veya başarısızlık gibi bilgiler olup ödül olarak adlandırılmaktadır. Öğrencinin amacı daha önceden belirlenmiş olan bir performans ölçütünü eniyilemektir. Bu performans alınan ödüllerin bir fonksiyonudur. En iyi eylem dizisini belirlemek için her durumda yeterli sayıda güncelleme yapmak gerekir. İdeal olanı sonsuz sayıda güncelleme olduğu halde bu mümkün olmadığından daha önceden yeterli olacağı varsayılmış sayıda güncelleme yapma yoluna gidilmektedir.

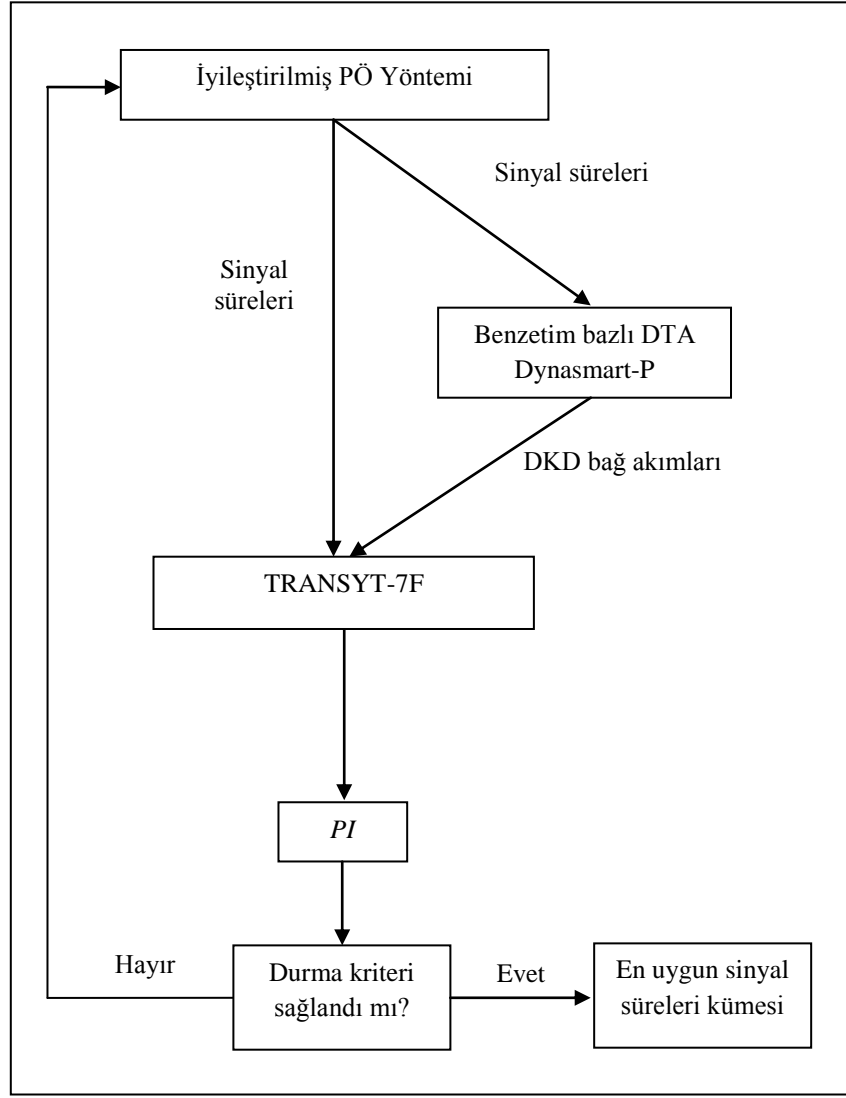
PÖ, deneyimlerle öğrenme yapısı üzerine kurulmuş ve durumlara veya durum-eylem çiftlerine değer ataması yapan bir yöntemdir (Sutton ve Barto, 1998). Bu yöntem, tecrübeye dayalı öğrenmede en çok tercih edilen yöntemlerden biridir çünkü, çoğunlukla insan davranışının söz konusu olduğu durumlarda doğru olarak nitelendirilebilecek eylem sayısı birden fazladır. Yapılan bir eylemin yararlığı pekiştirme ile ölçülebilir. Bu yöntem, en uygun eylemi seçebilmek için pekiştirme (yarar/zarar) tahminine dayanır. Anlık durum ve belirli bir eylemden sonraki durum karşılaştırılır ve eylemin uygulanmasının ne kadar yararlı olduğunu gösteren bir değer hesaplanır. Pekiştirme, sistemin hedeflerini, isteklerini ve durumu göz önünde bulunduran bir fonksiyonun sonucudur. En yüksek pekiştirmenin elde edildiği eylem tavsiyesi en iyi tavsiye olarak belirlenir ve seçilir. *Q*-öğrenme ise, PÖ yöntemleri içinde en çok tercih edilen yöntemlerden biridir (Watkins, 1989). *Q*-öğrenme algoritması, PÖ yöntemleri içinde modele ihtiyaç duymayan bir yaklaşım olup bu algorithmada çevrenin nasıl çalıştığı hakkında ajana herhangi bir bilgi verilmemektedir. Bunun yerine ajan eylemleri deneyerek en iyi ödülü veren eylemi seçmektedir. *Q*-öğrenme algoritması, durum-eylem çiftinin sahip olduğu değerlerin tahmin edilmesine bağlı olarak çalışmaktadır. *Q* değerleri olarak adlandırılan bu değerler verilen bir durum-eylem çifti için sayısal tahminler olarak nitelendirilmiştir.

Bu çalışmada iyileştirilmiş PÖ yönteminden faydalanılmıştır. İyileştirilmiş PÖ yöntemi, Q -öğrenme algoritmasına dayanmaktadır. Kullanılacak olan iyileştirilmiş PÖ yöntemi, literatürdeki PÖ yöntemlerinden farklı olup Çözüm Uzayı (ÇU) elde edilen en iyi çözüm değeri ve önceden verilen kısıt vektörüne bağlı olarak öğrenme süreci boyunca sınırlandırılmaktadır. Sınırlı ÇU tekniği sayesinde iyileştirilmiş PÖ algoritmasının performansı belirgin şekilde artmaktadır. Sonuç olarak, geliştirilen algoritma yerel en uygun çözüm noktalarına takılmaksızın global en uygun çözüme erişebilmektedir. PÖ, Q -öğrenme algoritması ve iyileştirilmiş PÖ algoritması ile ilgili detaylı bilgi Bölüm 3'de verilmektedir.

2.7. Tez Çalışmasının Genel Çerçevesi

Sinyal kontrol probleminin çözümü için trafik mühendisleri ve planlamacılar yıllardır etkin ve tutarlı yöntemler üzerine araştırmalar yapmaktadırlar. Problemin çözümü için göz önüne alınması gereken kontrol parametresi sayısının fazlalığı çözümün karmaşıklığını artırmaktadır. Ayrıca çözüm için probleme yeni kontrol parametrelerinin eklenmesi problemin konveks olmayan yapısından dolayı çözümü daha da zorlaştırmaktadır.

UAT probleminin çözümü için çalışmada önerilen PÖTRAD modelinin genel çerçevesi Şekil 2.3'de verilmektedir. PÖTRAD modelinde iyileştirilmiş PÖ yöntemi ile hem TA sürecinde hem de TRANSYT-7F yazılımının amaç fonksiyonu olan PI değerinin hesaplanmasında kullanılmak üzere sinyal süreleri kümesi oluşturulmaktadır. TA kısmında ulaşım ağı bilgileri ve B-V talebi Dynasmart-P benzetim bazlı dinamik trafik atama yazılımına verildikten sonra bağ akımları elde edilmektedir. TA süreci sonunda elde edilen bağ akımları ve iyileştirilmiş PÖ ile oluşturulan sinyal süreleri kümesinin TRANSYT-7F yazılımının trafik modelinde kullanılması ile amaç fonksiyonu olan PI değeri hesaplanmaktadır. PÖTRAD modelinde amaç fonksiyonu olan PI değeri önceden verilen yakınsama kriterini sağladığı anda en uygun sinyal süreleri kümesi elde edilmektedir. PÖTRAD modeli ile ilgili detaylı açıklamalar Bölüm 4'de verilmiştir.



Şekil 2.3 PÖTRAD modeli genel çerçevesi

2.8. Sonuçlar

Bu bölümde, UAT, iki seviyeli programlama yöntemi, KİY yöntemi, TA, Deterministik KD, stokastik KD, DTA hakkında detaylı bilgi ve formülasyonlar verilerek literatür çalışması yapılmıştır. Literatürden görüldüğü gibi DTA probleminin çözümü için çeşitli yaklaşımlar mevcuttur. Bu yaklaşımlardan analitik yaklaşımların trafiği modelleme konularında problemli oldukları ve analitik yaklaşımların pratik uygulamalarda rahatlıkla kullanılamayacağı konusunda ortak bir görüş vardır. Analitik yaklaşımların bu eksikliği benzetim bazlı DTA modellerinin gelişmesine sebep olmuştur. Literatürde çeşitli benzetim bazlı DTA modelleri bulunmaktadır. Bu çalışmada, benzetim bazlı DTA modeli olan Dynasmart-P yazılımı UAT probleminin alt seviyesi olan TA probleminin çözümünde

kullanılacağı için bu bölümde Dynasmart-P yazılımı ile ilgili kısa bir bilgi verilmiştir.

Bu bölümde sinyal kontrol problemi hakkında bilgi verilerek ve literatür çalışması yapılmıştır. Bu çalışmada UAT probleminin üst seviyesini oluşturan sinyal kontrol probleminde faz bazlı eniyileme yazılımı olan TRANSYT-7F yazılımı kullanılmıştır. TRANSYT-7F modeli, sinyalize kavşaklı bir ağda girilen sinyal süre ve faz planlarına bağlı olarak ağ PI değerini hesaplamak için trafiği simule etmektedir. TRANSYT-7F yazılımının amacı, koordine yol ağı için sinyal parametrelerinin tekrarlı olarak ayarlanması ve sadece sinyal parametrelerindeki değişimler ile ağ PI değerini düşüren en iyi sinyal parametrelerinin bulunmasıdır.

Bölüm 2.2'de verildiği gibi UAT problemi çözümü ile ilgili literatürde birçok çalışma mevcut olup birbirinden farklı birçok çözüm yöntemi sunulmuştur. Ancak son yıllarda yapılan çalışmaların çoğunda sezgisel metotların kullanımı oldukça artmıştır. Bu bölümde, PÖ yöntemi ve iyileştirilmiş PÖ yöntemi hakkında kısa bilgi verilmiştir. PÖ yöntemi özellikle son yıllarda sinyal kontrol probleminin eniyilenmesinde kullanılmaya başlanmış bir yöntemdir. Ancak iyileştirilmiş ve performansı artırılmış PÖ yöntemi ile UAT probleminin çözümü literatürde uygulanmamıştır. Bu nedenle çalışmada iyileştirilmiş PÖ yöntemi ile UAT problemi çözülmüştür. UAT probleminin çözümü için iki seviyeli programlama tekniği kullanılmıştır. Gelecek bölümde PÖ yöntemi ve iyileştirilmiş PÖ yöntemi ile ilgili detaylı bilgiler verilmiştir.

3. PEKİŞTİRMELİ ÖĞRENME YÖNTEMİ

3.1. Giriş

Pekiştirmeli öğrenme (PÖ) yönteminin temel felsefesi gerçekleşen bir olgunun ödüllendirilmesi veya cezalandırılmasını bir mantık içerisinde düzenleyerek istenen bir sonucun ortaya çıkmasını sağlamaktır. PÖ, bireyin çevre ile etkileşimi sonucunda öğrenmeyi vurgulayan yapay zeka yaklaşımıdır. PÖ yöntemi yöneylem araştırmasında, genetik algoritmalarda, sinir ağlarında, psikolojide ve kontrol mühendisliğinde kullanım alanı bulmaktadır.

Bu bölümde UAT probleminin üst seviyesi olarak belirtilen sinyal kontrol probleminin çözümü için kullanılacak olan PÖ yöntemi hakkında literatür çalışması verilmiştir. Bölüm 3.2’de PÖ yönteminin genel prensipleri, sonraki bölümde çalışmada kullanılan iyileştirilmiş PÖ yönteminin detaylı algoritma adımları verilmiştir. Bölüm 3.4’de iyileştirilmiş PÖ yönteminin sinyal kontrol problemine uyarlanması verilmiş ve son bölümde elde edilen sonuçlar irdelenmiştir.

3.2. Pekiştirmeli Öğrenme Yöntemi

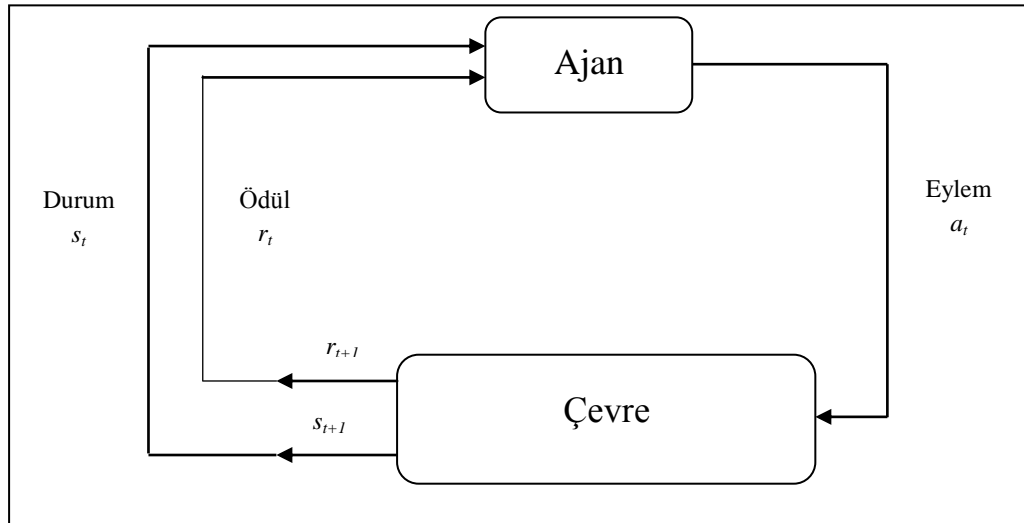
PÖ en basit tanımıyla deneme-yanılma ve ortamla etkileşim yoluyla öğrenme şeklinde tarif edilebilir. Öğrencinin etkileştiği, kendisi dışındaki her şey, çevre olarak adlandırılmaktadır. Öğrenci ve çevre sürekli olarak etkileşir; öğrenci bir eylem seçer, çevre bu eyleme bir tepki verir ve eylem sonucu çevre, yeni bir duruma geçer. Yeni durumun yanı sıra, ortam pekiştirme olarak adlandırabileceğimiz ödül veya cezayı öğrenciye iletir. Öğrencinin yapması gereken, alacağı ödülleri en yüksek seviyeye çıkarmak ve ceza getirecek hareketlerden uzak durmaktır.

PÖ, temel olarak belirli bir amaca erişmek için sosyal çevre ile etkileşim sonucunda öğrenme anlamına gelmektedir. PÖ, ilk olarak Markov Karar Süreçlerinin (MKS) çözümü için kullanılmıştır. Pekiştirmeli öğrenmede, sayısal bir ödül değerini enbüyüklemek için ne yapılacağı-eylemler ile durumları nasıl eşleştirileceği öğrenilmektedir. PÖ’de hangi eylemin yapılacağı söylenmemekte, bunun yerine eylemler denenerek en iyi ödülü sağlayan eylemlerin hangileri olduğu belirlenmek zorundadır. Çoğu zaman, yapılan eylemler sadece anlık ödülleri değil sonraki durum ve durumlarda elde edilen ödülleri etkileyebilmektedir. Bu iki özellik (deneme-hata araştırması ve gecikmiş ödül) PÖ’nün en ayırtedici iki özelliğidir (Sutton ve Barto, 1998).

PÖ’de karar verici ya da öğrenici ajan olarak adlandırılmakta ve ajanın çevresi ile etkileşim içinde olduğu kabulü yapılarak bu yaklaşım geliştirilmiştir. Bu etkileşim, ajana çevreyi algılamasını ve bu algılamaya bağlı olarak çevre içinde uygulayacağı eylemi seçmesini sağlamaktadır. Ajanın etrafındaki herşey sosyal çevreyi oluşturmaktadır. Çevre ile ilgili bilgiler, ajana bu etkileşim aracılığıyla iletilmektedir. Elde edilen bilgilere bağlı olarak, ajan çevre içinde uygulayacağı eylemi seçmektedir. Uygulanan eylem çevreyi değiştirebilmekte ve bu değişim ajana sayısal bir sinyal aracılığıyla iletilmektedir. Çevre ise yapılan eyleme karşı sayısal değerlerden oluşan ödülleri açığa çıkarmakta ve ajan zamanla bu ödülleri en büyükmeye çalışmaktadır. Ajan ve çevre ardışık zaman adımlarında, $t = (0,1,2,3,...)$, birbiriyle etkileşmektedir. Her bir t zaman adımında, ajan çevrenin muhtemel durum kümesinden, S , çevrenin durumu, $s_t \in S$, hakkında belirli bilgiler almakta ve s_t durumundaki muhtemel eylemler kümesinden, $A(s_t)$, bir eylem, $a_t \in A(s_t)$, seçmektedir. Sonrasında, ajan çevreden sayısal bir ödül, $r_{t+1} \in R$, almakta ve kendini çevrenin yeni bir durumunda, s_{t+1} , konumlandırmaktadır (Sutton ve Barto, 1998). Şekil 3.1’de ajan-çevre etkileşimi görülmektedir.

PÖ çözüm yöntemleri, Dinamik Programlama, Monte Carlo ve Geçici Fark Öğrenme yöntemleri olmak üzere üç ana kategoriden oluşmaktadır. Her kategorinin kendine göre avantajları ve dezavantajları bulunmaktadır. Bir Markov işlemi olarak ortamın eksiksiz bir modelinin verildiği durumda en uygun davranış biçimini hesaplayabilmek için kullanılacak algoritmalar kümesine, Dinamik Programlama

adı verilmektedir. Dinamik Programlamanın temel fikri, iyi davranış biçimlerini aramayı yapılandırmak ve organize etmek için değer fonksiyonlarının kullanılmasıdır. Dinamik Programlamanın tersine Monte Carlo Yöntemi, sistem modeline ihtiyaç duymaz. Tek ihtiyacı olan deneyimlerdir yani ortamla olan etkileşimler sonucu elde edilen durum, eylem ve ödül bilgileridir. Ortamla ilgili hiçbir ön bilgiye sahip olmayıp, yalnızca deneyimlerle en uygun davranışı yakalayabilmesi bu yöntemin dikkat çekici özelliğidir. Geçici Fark Öğrenme yöntemleri, PÖ yönteminin ana fikri olarak adlandırılmaktadır. Geçici Farklar Yöntemi Monte Carlo ve Dinamik Programlama yöntemlerinin bir bileşimi gibidir. Monte Carlo Yöntemi gibi doğrudan ham deneyimden öğrenebildiği gibi, dinamik programlama yöntemleri gibi bir bölümün tamamlanmasını beklemeye gerek kalmadan andan ana değer güncellemesi yapabilmektedir (Abdulhai ve Kattan, 2003).



Şekil 3.1 PÖ’de ajan-çevre etkileşimi (Sutton ve Barto, 1998)

Geçici fark öğrenme yöntemleri; Q -öğrenme, Sarsa ve Sarsa(λ) algoritması olmak üzere üçe ayrılmaktadır. Geçici fark öğrenme yöntemlerinden biri olan Q -öğrenme algoritması, PÖ yöntemleri içinde modele ihtiyaç duymayan bir yaklaşım olup bu algortmada çevrenin nasıl çalıştığı hakkında ajana herhangi bir bilgi verilmemektedir. Bunun yerine ajan eylemleri deneyerek en iyi ödülü veren eylemi seçmektedir. Q -öğrenme algoritması, durum-eylem çiftinin sahip olduğu değerlerin tahmin edilmesine bağlı olarak çalışmaktadır. Q değerleri olarak adlandırılan bu değerler verilen bir durum-eylem çifti için sayısal tahminler olarak nitelendirilmiştir

(Bazzan vd, 2010). Algoritmada, Q tablosu olarak adlandırılan tablonun elemanları her durum değişiminde güncellenmektedir (Sutton ve Button, 1998). Bu tabloda her bir durum, s , ve eylem, a , çifti için bir Q değeri, $Q(s,a)$, bulunmaktadır. Q tablosu örneği Tablo 3.1’de verilmektedir.

Tablo 3.1 Q tablosu örneği

	Eylem 1	Eylem 2	Eylem 3	Eylem n
Durum 1	0.002	0.342	0.188	0.862
Durum 2	0.221	0.541	0.358	0.452
Durum 3	0.479	0.323	0.655	0.224
...
...
Durum m	0.552	0.711	0.123			0.122

Ajan, çevrenin bir durumdan, s_t , başka bir duruma geçişinde, s_{t+1} , yapılan eylemin, a_t , ardından ödül, r_{t+1} , almaktadır. Q değerleri Denklem (3.1) ile tanımlanmaktadır.

$$Q(s,a) = r(s,a) + \gamma \times Q^*(s',a') \quad (3.1)$$

Burada, $Q(s,a)$, durum eylem çifti (s,a) için Q değeridir. $Q^*(s',a')$, s durumunda a eylemi tamamlanmasının ardından gelen s' durumunda seçilen a' eyleminin seçimi ile elde edilebilen en iyi Q değeridir. $r(s,a)$, s durumunda a eylemi tamamlandığında alınan ödül değeridir. γ , gelecekteki ödüllere atanan ağırlığı temsil eden azaltma faktörüdür (Vanhusel vd, 2009). Q tablosu öğrenme süreci boyunca Tablo 3.2’de gösterildiği gibi doldurulmaktadır.

Tablo 3.2 Pekiştirmeli öğrenme süreci (Kaelbling vd, 1996)

Başlangıç Q -değerlerini oluştur
N defa tekrarla (N =öğrenme evresi sayısı)
Rasgele bir s durumu seç
Öğrenme evresinin sonuna kadar tekrarla
Bir a eylemi seç ve uygula
Ödül değerini $r(s,a)$ al
Yeni s' durumunu gözlemler
Q tablosunu durum-eylem çiftleri (s,a) için güncelleme kuralına göre güncelle
$s = s'$ olarak ayarla

Öğrenme süreci, belirli sayıdaki öğrenme evresinden meydana gelmektedir. Her öğrenme evresi rasgele bir s durumunda başlamakta, ajan bir a eylemi seçip tamamlamakta, ödülü almakta ve yeni durumu gözlemlemektedir. Buna bağlı olarak, ajan Q değerlerini bu durum-eylem çiftine göre Denklem (3.2)'de verilen bağıntıyı kullanarak güncellemektedir.

$$Q_t(s, a) = (1 - \alpha) \times Q_{t-1}(s, a) + \alpha \times \left[r(s, a) + \gamma \times \max_{a'} Q_{t-1}(s', a') \right] \quad (3.2)$$

Burada, $Q_t(s, a)$ t 'inci öğrenme evresinde durum eylem çifti (s, a) için güncellenmiş Q değeri, $Q_{t-1}(s, a)$ t 'inci öğrenme evresinden bir önceki öğrenme evresinde durum eylem çifti (s, a) için Q tablosuna kaydedilmiş ve güncellenmesi gereken Q değeridir. α ise algoritmanın öğrenme oranı olup bir önceki öğrenme evresinde kaydedilen $Q_{t-1}(s, a)$ değeri ile bir önceki öğrenme evresinde s durumunda a eylemi tamamlanmasının ardından gelen s' durumunda seçilen a' eyleminin seçimiyle elde edilebilen en iyi Q değerini $Q_{t-1}(s', a')$ karşılaştırmak için yeni hesaplanan Q değerine $\left(r(s, a) + \gamma \times \max_{a'} Q_{t-1}(s', a') \right)$ atanan ağırlığı tanımlamaktadır ve γ azaltma faktörüdür (Vanhussel vd, 2009).

3.3. İyileştirilmiş Pekiştirmeli Öğrenme Yöntemi

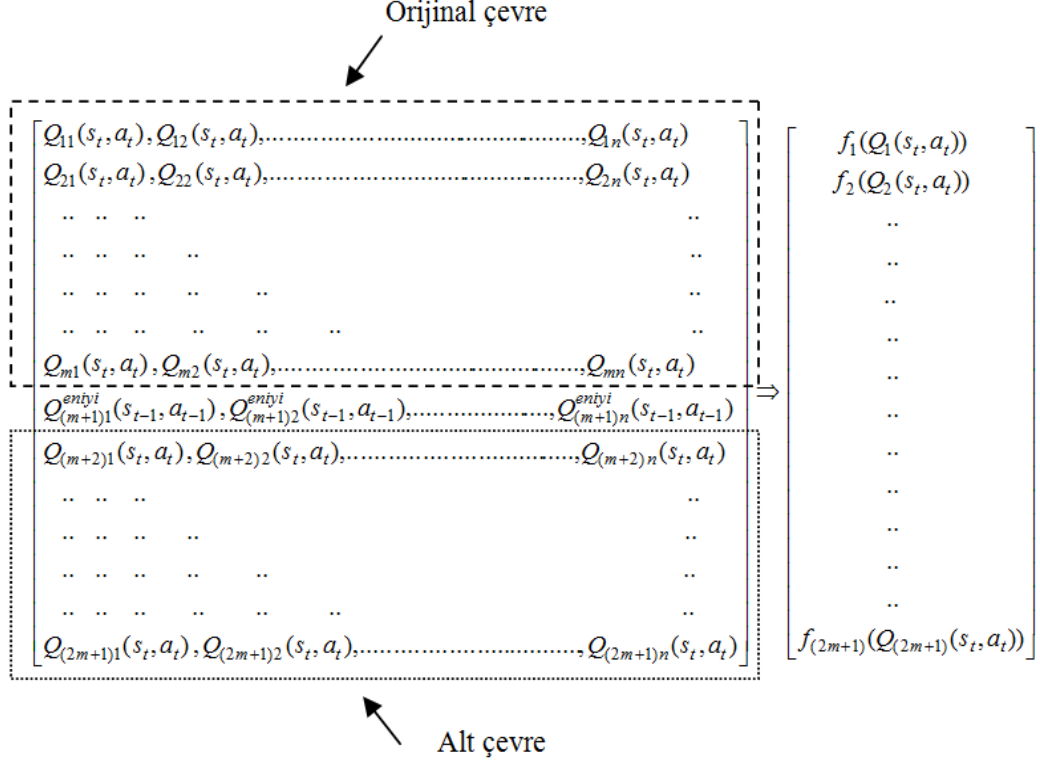
Bu çalışmada iyileştirilmiş PÖ yönteminden faydalanılmıştır. İyileştirilmiş PÖ yöntemi, Q -öğrenme algoritmasına dayanmaktadır. Kullanılacak olan iyileştirilmiş PÖ yöntemi, literatürdeki PÖ yöntemlerinden farklı olup Çözüm Uzayı (ÇU) elde edilen en iyi çözüm değeri ve önceden verilen kısıt vektörüne bağlı olarak öğrenme süreci boyunca sınırlandırılmaktadır. Sınırlı ÇU tekniği sayesinde iyileştirilmiş PÖ algoritmasının performansı belirgin şekilde artmaktadır.

Literatürde sezgisel yöntemlerin PÖ'ye entegre edilerek iyileştirilmiş çeşitli eniyileme problemlerini çözen çalışmalar mevcuttur. Liu ve Zheng (2009) yapmış oldukları çalışmada gezgin satıcı problemi için PÖ mutasyonu ile geliştirilmiş GA çözümü önermişlerdir. Geliştirilen hibrid algoritma ile makul zamanda en uygun çözümler elde edilebilmiştir. Maravall vd (2009), bilinmeyen engelleri içeren karışık çevredeki robotların otonom hareket kontrol problemini çözmek için PÖ ile evrimsel

algoritmaları entegre etmişlerdir. Bu yaklaşım ile gerçek zamanlı durumlarda evrimsel algoritmalarla PÖ algoritmasının birleşmesi sonucu elde edilen hibrid çözüm algoritmaların etkinliği gösterilmiştir. Deney sonuçları, gerçek zamanlı ve yüksek boyutlu robot hareket planlaması ve kontrolü problemlerini çözmek için hibrid yaklaşımların geçerliliğini doğrulamıştır. Chen vd (2009) yapmış oldukları çalışmada Genetik Ağ Programlama (GAP) ile Sarsa öğrenme yöntemi kullanılarak genişletilmiş hisse senedi alım satım modeli geliştirilmiştir. Çalışmada simulasyon modeli uygulanmış ve önerilen modelin etkinliğinin teyit edilmesi için modelin sonuçları GAP, GA gibi diğer metotlarla karşılaştırılmıştır. Sonuçlarda, GAP-Sarsa algoritması kullanılarak geliştirilen hisse senedi alım satım modelinin diğer tüm yöntemlerden daha iyi olduğu gösterilmiştir.

Wu vd (2011) yapmış oldukları çalışmada iş çizelgeleme problemleri için Ordinal Sharing Learning (OSL) olarak adlandırılan özgün çok ajanlı PÖ yöntemi önermişlerdir. Benzetim sonuçlarında, OSL metodunun etkin şekilde amaca ulaşabildiği ve geliştirilen yöntemin performansının bazı merkezi çizelgeleme algoritmaları ile kıyaslanabilir olduğu gösterilmiştir.

PÖ yöntemleri ile diğer sezgisel yöntemlerin birleştirilmesi ile elde edilen iyileştirilmiş PÖ algoritmaları farklı alanlardaki eniyileme problemlerinin çözümünde son yıllarda etkin olarak kullanılmaktadır. PÖ yöntemi eniyileme problemlerine başarılı şekilde uygulanan yaklaşım olmasına rağmen, bu yöntem en uygun çözüm elde etmek için daha da geliştirilebilir. Bu amaçla, herhangi bir matematiksel fonksiyon için global veya globale yakın en uygun çözümü bulmak için İyileştirilmiş PÖ (İPÖ) algoritması geliştirilmiştir. İPÖ algoritması, modele ihtiyaç duymayan PÖ tekniklerinden Q -öğrenme algoritmasına dayanmaktadır. Geliştirilen bu algorithmada, her öğrenme evresinde daha önceki en iyi çözüm bilgisinin yardımıyla orijinal çevre boyutunda bir alt çevre ACORSES modelindeki yöntem (Baskan vd, 2009) kullanılarak üretilmektedir (Şekil 3.2).



Şekil 3.2 İPÖ algoritmasında orijinal ve alt çevrenin gösterimi

Burada, m çevrenin boyutunu, n karar değişkenlerinin sayısını ve t öğrenme evresinin sayısını ifade etmektedir. Şekil 3.2’den görüldüğü üzere, önceki öğrenme evresinden elde edilen en iyi $Q(s, a)$ değeri, yerel en uygun çözüme takılmayı önlemek için çevrenin $(m+1)$ ’inci satırında saklanmaktadır. Ayrıca, alt çevre, orijinal çevre boyutunda $(m+2)$ ’nci satırdan $(2m+1)$ ’nci satıra kadar önceki öğrenme evresinde elde edilen en iyi değer ve β vektörü kullanılarak Denklem (3.3) yardımıyla oluşturulmaktadır.

$$rasgele(Q_{t-1}^{en\ iyi}(s, a) - \beta ; Q_{t-1}^{en\ iyi}(s, a) + \beta) \quad (3.3)$$

Böylelikle, global en uygun çözüm, algoritma süreci boyunca β değeri ile sınırlandırılan alt çevre yardımıyla aranmaktadır. Şekil 3.3’ün 7’nci satırında gösterildiği gibi β değeri alt çevre içinde çözüm uzayı boyutunu sınırlandırmak için azaltılmak zorundadır. β_j bir vektör ve $j=1,2,\dots,n$ kadar olup, n karar değişkenlerinin sayısıdır. β değerinin aralığı verilen probleme bağlı olarak seçilebilmektedir (Baskan vd., 2009).

İPÖ algoritmasında, Geem vd. (2001) tarafından önerilen Armoni Araştırmasında (AA) olduğu gibi t ’inci öğrenme evresinde amaç fonksiyonunun değerine göre daha

iyi $Q(s,a)$ elde etmek için orijinal çevre ve alt çevre iyiden kötüye doğru sıralanmaktadır. Bu sayede, bir önceki öğrenme evresinden elde edilen en iyi çözüm ve alt çevre, orijinal çevre ile karşılaştırılmaktadır. Çözümlerden biri kötü olanlardan daha iyi değer sağlıyorsa, yeni değerler orijinal çevrenin içine dahil edilirken kötü değerler çevreden çıkarılmaktadır. Sonuç olarak, geliştirilen algoritma yerel en uygun çözüm noktalarına takılmaksızın global en uygun çözüme erişebilmektedir. İPÖ algoritması, herhangi bir fonksiyon için en uygun çözümü sağlamaktadır. Çünkü global en uygun çözüm bir önceki öğrenme evresinden elde edilen en iyi çözüm etrafında hem alt çevre hem de orijinal çevre yardımıyla aranmaktadır. İPÖ algoritmasının çözüm prosedürünün adımları Şekil 3.3’de gösterilmiştir.

İPÖ algoritmasında ödül fonksiyonunun değeri, $r_t(s,a)$, s durumunda yapılan a eyleminin amaç fonksiyonunu nasıl etkilediğini belirlemektedir. Bu amaçla geliştirilen ödül fonksiyonu Denklem (3.4)’de verilmektedir.

$$\lim_{r_t(s,a) \rightarrow 0} \left(\frac{Q_t^{en\ iyi}(s,a) - Q_t(s,a)}{Q_t(s,a)} \right) = 0 \quad (3.4)$$

Burada, $r_t(s,a)$ ödül fonksiyonu, $Q_t(s,a)$ Q değeri ve $Q_t^{en\ iyi}(s,a)$ t 'inci öğrenme evresinde elde edilen en iyi Q değeridir. Geliştirilen algoritmada ödül fonksiyonu, en iyi Q değeri ile öğrenme evresindeki Q değeri arasındaki farkın Q değerine bölümüdür. İPÖ algoritmasında ödül değerleri, ödül fonksiyonunun yapısından dolayı “0” değerine yaklaşmaktadır. Öğrenme evrelerinin sonunda, amaç fonksiyonunun en uygun veya en uyguna yakın bir değeri elde edildiğinde, tüm ödül değerleri “0” değerine yaklaşmaktadır. Geliştirilen algoritmada toplam ödül değerinin, amaç fonksiyonunun en uygun veya en uyguna yakın bir değere ulaşması için enküçüklenmesi gerekmektedir.

İPÖ algoritmasının uygulaması literatürde yer alan matematiksel bir fonksiyon üzerinde test edilmiştir. Ayrıca, İPÖ algoritması ile Q -öğrenme algoritmasına dayalı PÖ algoritmasının karşılaştırılması yapılmıştır. PÖ ve İPÖ algoritmaları MATLAB R2009a yazılımı kullanılarak kodlanmış ve Intel Core 2 CPU 2.00 GHz, RAM 2 GB bilgisayarda çalıştırılmıştır. İPÖ algoritmasında çevre boyutu 20, γ azaltma faktörü 0.2 ve α öğrenme oranı 0.8 olarak seçilmiştir. β çözüm uzayı parametresi verilen problemin en küçük ve en büyük sınırları arasında alınmıştır. Çözüm süreci, durma

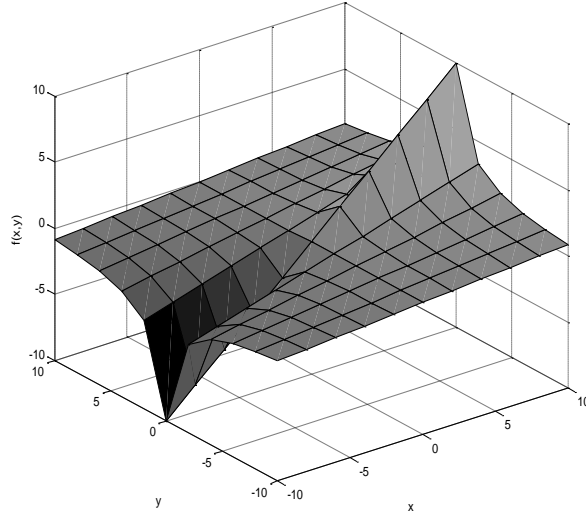
<i>Başlangıç $t=0$. β, α, γ ayarla</i>
<i>While $t \leq t_{max}$ (t_{max}=maksimum öğrenme evresi sayısı)</i>
<i>If $t=0$, başlangıç $Q(s, a)$ değerlerini oluştur</i>
<i>Çevrenin her boyutundaki $Q_t(s, a)$ değerleri için amaç fonksiyonunun değerini hesapla</i>
<i>Else</i>
<i>En iyi $Q_{t-1}^{en\ iyi}(s, a)$ sakla</i>
<i>$\beta_t = \beta_{t-1} * 0.99$</i>
<i>Alt çevrenin $Q_t(s, a)$ değerlerini Denklem (3.3)'ü kullanarak oluştur</i>
<i>Alt çevrenin her boyutundaki $Q_t(s, a)$ değerleri için amaç fonksiyonunun değerini hesapla</i>
<i>Çevre ve alt çevreyi iyiden kötüye doğru sırala</i>
<i>end if</i>
<i>En iyi $Q_t^{en\ iyi}(s, a)$ bul</i>
<i>$r_t(s, a)$ değerini Denklem (3.4)'ü kullanarak hesapla</i>
<i>$Q_t(s, a)$ değerlerini Denklem (3.2)'yi kullanarak güncelle</i>
<i>$t=t+1$</i>
<i>End while</i>

Şekil 3.3 İPÖ algoritması

kriteri sağlanıncaya kadar devam etmiştir. Durma kriteri her fonksiyon için global en uygun çözümü sağlayacak şekilde verilmiştir.

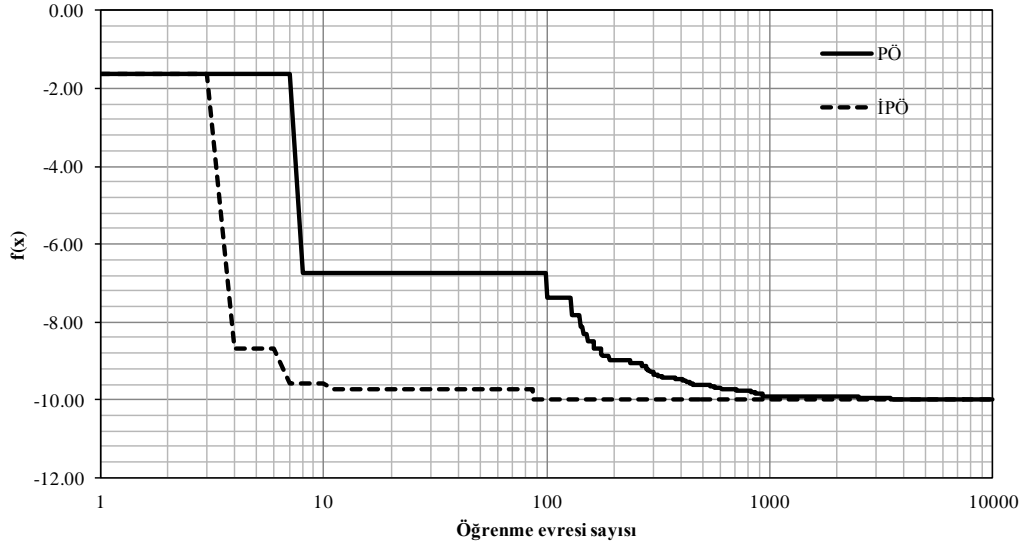
İPÖ ve PÖ algoritmasının karşılaştırılması için kullanılan test fonksiyonu Denklem (3.5)'de verilmektedir. Test fonksiyonu için $(x_1, x_2) = (-10, 0)$ olduğunda global en uygun çözümü $f(x_1, x_2) = -10$ olmaktadır. Amaç fonksiyonunun grafiği Şekil 3.4'de verilmektedir.

$$f(x_1, x_2) = \frac{x_1}{1 + |x_2|} \quad (3.5)$$



Şekil 3.4 [-10,10] aralığında amaç fonksiyonunun grafiği

İPÖ ve PÖ algoritmalarının yakınsama davranışları Şekil 3.5’de görülebilmektedir. Şekil 3.5’den görüldüğü üzere, her iki algoritma aynı başlangıç çözümlerinden başlamıştır. Bu test fonksiyonu için her iki algoritma en uygun çözümü bulmada yetenekli olmasına rağmen, İPÖ algoritması, PÖ algoritmasına göre daha az öğrenme evresinde en uygun çözüme ulaşmıştır. İPÖ algoritması, verilen test fonksiyonu için sadece 1176 öğrenme evresinde en uygun çözüme ulaşırken PÖ algoritması ise 7337 öğrenme evresinde en uygun çözüme ulaşmıştır.



Şekil 3.5 İPÖ ve PÖ algoritmalarının yakınsama davranışları

İPÖ algoritmasının tutarlılık analizi, Denklem (3.6)'da verilen Shelokar vd (2007) tarafından Başarı Oranı (BO) olarak adlandırılan fonksiyon ile yapılmıştır.

$$BO = \frac{100 * N_s}{N_T} \quad (3.6)$$

Burada, N_s algoritmanın gerekli hassasiyeti sağlayarak doğru sonuç verdiği çalışma sayısını, N_T ise algoritmanın toplam çalışma sayısını ifade etmektedir. Burada, amaç fonksiyonu global en uygun çözüme % 3.5 oranında yaklaştığında başarılı olarak değerlendirilmektedir. Geliştirilen algoritma, diğer algoritmalarla daha gerçekçi olarak kıyaslanabilmesi için toplam 50 defa çalıştırılmıştır. İPÖ algoritması, Particle Swarm Ant Colony Optimization-PSACO (Shelokar vd, 2007), Chaotic Particle Swarm Optimization-CPSO, PSO ve GA (Liu vd, 2005) algoritmalarının tutarlılık analizi sonuçları Tablo 3.3'de verilmektedir.

Tablo 3.3'den görülebildiği gibi, PSACO algoritması hariç diğer algoritmalarla İPÖ algoritması kıyaslandığında, İPÖ algoritmasının yüksek başarı oranında global en uygun çözümü bulabildiği görülmektedir. PSACO algoritması F12 ve F13 fonksiyonları için İPÖ algoritmasına göre daha yüksek başarı oranı vermesine rağmen, İPÖ ve PSACO algoritması F2, F7, F9 ve F16 fonksiyonları için aynı sonucu vermektedir. GA ise İPÖ ve PSACO'ya göre daha kötü sonuçlar vermektedir.

Tablo 3.3 Tutarlılık analizi sonuçları

Fonksiyon	BO değerleri				
	İPÖ	PSACO	CPSO	PSO	GA
F2	100	100	98	100	84
F7	100	100	100	98	98
F9	100	100	100	98	98
F12	98	100	90	96	16
F13	96	98	96	26	94
F16	100	100	100	94	92

İPÖ algoritmasının global veya globale en yakın en uygun çözümü bulmasındaki performansı onaltı fonksiyon üzerinde test edilmiştir. Test fonksiyonlarının tümü Ek 1'de verilmektedir. Fonksiyon 6, 7, 9, 13 ve 16 Shelokar vd'nin (2007) yaptığı çalışmadan alınmıştır. Fonksiyon 8 ve 14 sırasıyla Sun ve Dong'un (2011) yaptığı çalışmadan ve Chen vd'nin (2011) yapmış oldukları çalışmadan alınmıştır. Diğer fonksiyonlar Baskan vd'nin (2009) yapmış olduğu çalışmadan alınmıştır. Bu

fonksiyonların seçilme nedeni, bu fonksiyonlar için global en uygun çözümün bulunmasındaki zorluktur. Tablo 3.4’de, İPÖ algoritmasının performansının değerlendirilmesi için kullanılan fonksiyonların çözüldüğü farklı algoritmalar verilmektedir.

İPÖ algoritmasının yeteneğini test etmek için literatürde mevcut olan ve Tablo 3.4’de verilen 10 tane farklı algoritma ile kıyaslama yapılmıştır. Bu amaçla, onaltı test fonksiyonu İPÖ algoritması ile 100 defa çözülmüştür. Sonuçlar; en iyi fonksiyon değeri, fonksiyon değerlendirme sayısı, en iyi çözüm süresi, başarı oranı, ortalama fonksiyon değerlendirme sayısı ve ortalama hata olarak Tablo 3.5’de verilmektedir. En iyi fonksiyon değeri, algoritmanın çalıştırılması ile gereken hassasiyet değeri içinde global en uygun çözüme en yakın olarak bulunan değerdir. Gereken hassasiyet değeri, en iyi fonksiyon değeri ile teorik global en uygun çözüm değeri arasındaki farkın mutlak değeri olarak belirlenmiştir. Tüm test fonksiyonları için bu değer “0” olarak seçilmiştir. Fonksiyon değerlendirme sayısı, algoritmanın çalıştırılma sayısıdır. En iyi çözüm süresi ise en iyi çözüm için gereken zamandır. Ortalama fonksiyon değerlendirme sayısı, gereken hassasiyette en iyi çözümü sağlayan İPÖ algoritmasının ortalama çalışma sayısıdır. Ortalama hata ise en iyi fonksiyon değeri ile teorik global en uygun çözüm değerinin farkının ortalaması olarak tanımlanmıştır.

İPÖ algoritması F3 fonksiyonu haricinde tüm test fonksiyonları için global en uygun çözüm değerini bulabilmektedir. F3 fonksiyonu için gereken hassasiyette global en uygun çözüm değerini yakalayamamaktadır. Bu fonksiyon için İPÖ algoritması teorik global en uygun çözüm değerini yakalayamamasına rağmen IGARSET algoritmasına göre daha iyi değerler üretmektedir. İPÖ algoritması tüm test fonksiyonları için diğer algoritmalara göre daha az ortalama hata değeri vermektedir. Aynı zamanda, bu ortalama hata değerlerinin çoğu “0” değerine eşittir. Bu, İPÖ algoritmasının neredeyse her çalıştırılışında global veya globale yakın en uygun çözüm değerlerini bulabildiği anlamına gelmektedir. Sonuç olarak, herhangi verilen bir fonksiyonun en uygun çözüm değerlerinin bulunmasında İPÖ algoritmasının oldukça tutarlı ve etkin bir algoritma olduğu söylenebilmektedir. Tablo 3.5’den görülebildiği gibi, test fonksiyonlarının çoğunda gereken hassasiyet değerinden

Tablo 3.4 İPÖ algoritması ile kıyaslanan algoritmalar

Fonksiyonlar	Algoritma	Referans
F4-F7	SZGA	Successive zooming genetic algorithm (Kwon et al., 2003)
F1-F2-F3-F4	IGARSET	Improving GAs by random search technique (Hamzaçebi, 2008)
F7-F12-F13	ACO	Ant colony optimization (Toksarı, 2009)
F4-F5-F11-F12-F13-F15-F16	PSACO	Particle swarm and ant colony algorithm (Shelokar et al., 2007)
F5	ECTS	Enhanced continuous tabu search (Chelouah and Siarry, 2000)
F10	ACORSES	Ant colony optimization (Baskan et al., 2009)
F8	SA	Simulated annealing (Sun and Dong, 2011)
F14	RW-PSO+BOF	Random walking particle swarm optimization (Chen et al., 2011)
F5-F6-F7-F9-F11-F12-F13	GA-PSO	Genetic algorithm particle swarm optimization (Kao and Zahara, 2008)
F4-F7-F9	GAWLS	Genetic algorithm (Tutkun, 2009)
Tüm fonksiyonlar	İPÖ	İyileştirilmiş Pekiştirmeli Öğrenme algoritması

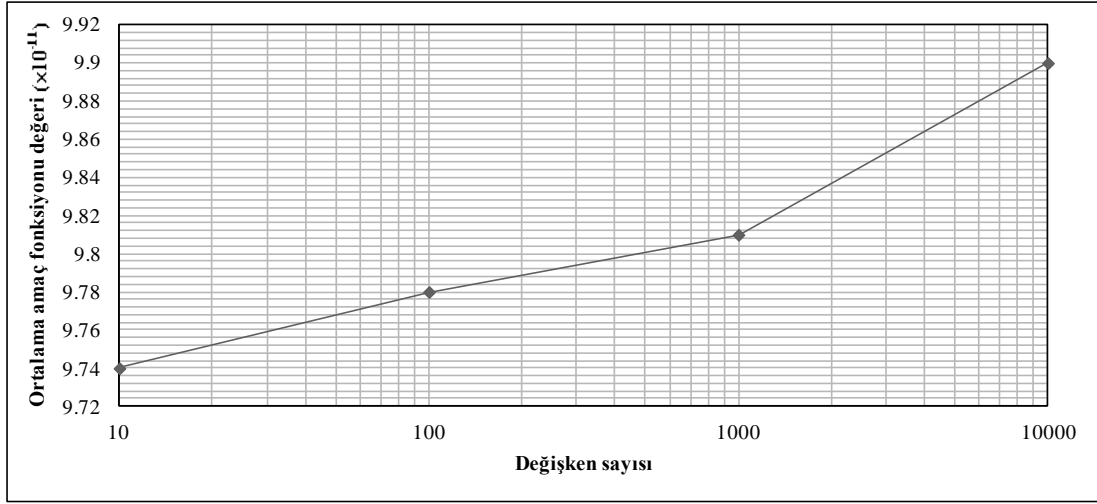
dolayı İPÖ algoritması diğer algoritmalarından daha fazla fonksiyon değerlendirme sayısı gerektirmektedir. Bunun nedeni tüm test fonksiyonları için gereken hassasiyet değerinin “0” olarak seçilmesidir.

İPÖ algoritmasının değişken sayısının çok olduğu durumlardaki etkinliğini göstermek için global en uygun çözüm değerine yakınsaması zor olan Ackley fonksiyonu seçilmiştir. Bu fonksiyonun global en küçük değeri $x=0$ 'da $f(x)=0$ 'dır. Shelokar vd'nin (2007) yapmış oldukları çalışmadan alınan bu fonksiyon Denklem (3.7)'de verilmektedir. Shelokar vd (2007) tarafından önerilen PSACO algoritması ile İPÖ algoritmasını daha gerçekçi olarak kıyaslayabilmek için bu fonksiyon değişken sayıları farklı olmak üzere her defasında 10 kez tekrarlanarak çözülmüştür. Ortalama fonksiyon değerlendirme sayıları ve ortalama amaç fonksiyonu değerleri değişken sayısının farklı olduğu her durum için kayıt altına alınmıştır. Bu fonksiyon için gereken hassasiyet değeri $1*10^{-10}$ birim olarak seçilmiştir.

$$AK_{(n)}(x) = -20 \exp(-0.2 * \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e \quad (3.7)$$

- Çözüm aralığı: $-32.768 \leq x_i \leq 32.768$; $i = 1, 2, \dots, n$; $n =$ değişken sayısı
- $x = (0, \dots, 0)$ $AK_n(x) = 0$

Şekil 3.6’da Ackley fonksiyonunun farklı değişken sayılarına göre 10 kez çalıştırılması sonunda elde edilen ortalama fonksiyon değerlerinin değişimi görülebilmektedir.



Şekil 3.6 Farklı değişken sayıları için ortalama amaç fonksiyonu değeri

Şekil 3.6’den görülebildiği gibi Ackley fonksiyonunun çok değişkenli durumları için İPÖ algoritmasının performansı oldukça iyidir. Ackley fonksiyonu için değişken sayısının aralığı 10 ile 10000 arasında seçilmiştir. Ortalama amaç fonksiyonu değeri 10 değişkenli durum için 9.74×10^{-11} ve 10000 değişkenli durum için 9.89×10^{-11} olarak bulunmuştur. Değişken sayısının farkı düşünüldüğünde elde edilen değerler birbirine çok yakındır. Şekil 3.6’dan 1000 değişkenli durum için ortalama amaç fonksiyonunun değeri 9.81×10^{-11} olduğu görülebilmektedir. Shelokar vd (2007) yapmış oldukları çalışmada önerdikleri PSACO algoritması ile 1000 değişkenli Ackley fonksiyonu için ortalama amaç fonksiyonunun değerinin yaklaşık “3” olduğunu belirtmişlerdir. Sonuçlardan görülebildiği gibi, her iki algoritmanın verdiği sonuçlar kıyaslandığında İPÖ algoritmasının çok değişken sayılı fonksiyonların global veya globale yakın en uygun çözüm değerinin bulunmasında oldukça iyi bir yöntem olduğu söylenebilmektedir. Şekil 3.7’den Ackley fonksiyonunun değişken sayısına göre ortalama fonksiyon değerlendirme sayısının değişimi görülebilmektedir.

Table 3.5. İPÖ algoritmasının ve kıyaslanan diğer algoritmaların sonuçları

Fonksiyon	Yöntem	En iyi fonksiyon değeri	Fonksiyon değerlendirme sayısı	En iyi çözüm süresi (sn)	Başarı Oranı	Ortalama fonksiyon değerlendirme sayısı	Ortalama hata
F1	IGARSET	0	2174	0.0568	MD	MD	MD
	İPÖ	0	68760	0.8688	100	71000	0
F2	IGARSET	-2	2400	0.0614	MD	MD	MD
	İPÖ	-2	34920	1.1790	97	36200	0
F3	IGARSET	$2.08*10^{-27}$	1821	0.0666	MD	MD	MD
	İPÖ	$9.71*10^{-34}$	31620	0.9584	98	31740	$9.33*10^{-34}$
F4	SZGA	$2.9*10^{-8}$	4000	MD	MD	MD	MD
	IGARSET	0	1004	0.0485	MD	MD	MD
	GAWLS	0	2572	MD	MD	MD	MD
	PSACO	MD	MD	MD	100	370	$5.55*10^{-17}$
	İPÖ	0	34340	0.5660	99	35700	$1.11*10^{-18}$
F5	ECTS	MD	MD	MD	MD	338	$3*10^{-08}$
	PSACO	MD	MD	MD	100	190	$7.69*10^{-29}$
	GA-PSO	MD	MD	MD	100	206	0.00004
	İPÖ	0	36955	21.1367	100	36980	0
F6	GA-PSO	MD	MD	MD	100	8254	0.00009
	İPÖ	-1	40680	0.6722	100	40960	0
F7	SZGA	3	9000	MD	MD	MD	MD
	ACO	MD	MD	0.11 ^a	MD	264 ^a	MD
	GAWLS	3	2573	MD	MD	MD	MD
	GA-PSO	MD	MD	MD	100	25706	0.00012
	İPÖ	3	33920	0.5344	100	37120	0
F8	SA	$-9.999994*10^{-01}$	16801	MD	MD	MD	MD
	İPÖ	-1	35640	0.5365	100	38380	0

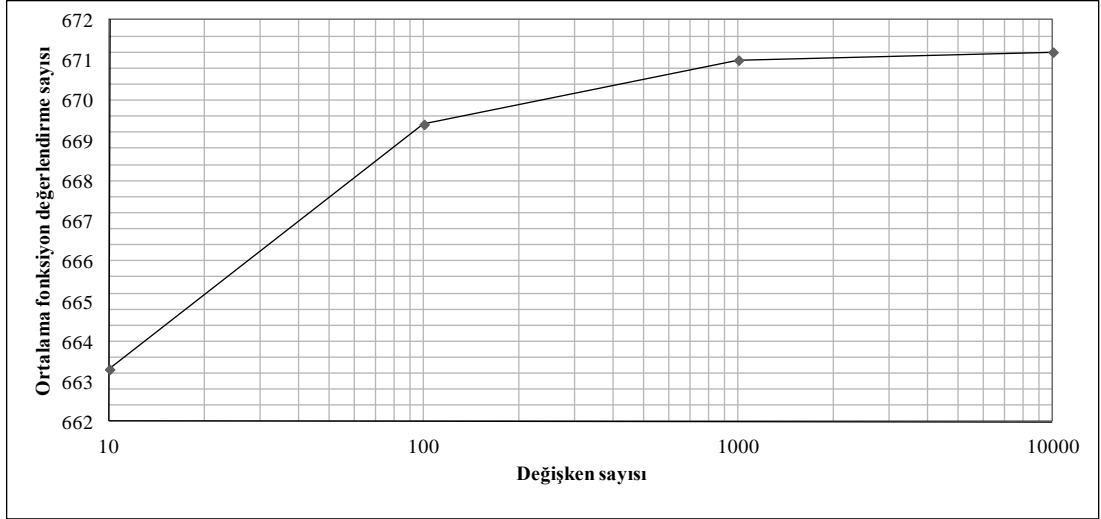
Table 3.5. İPÖ algoritmasının ve kıyaslanan diğer algoritmaların sonuçları (devam)

Fonksiyon	Yöntem	En iyi fonksiyon değeri	Fonksiyon değerlendirme sayısı	En iyi çözüm süresi (sn)	Başarı Oranı	Ortalama fonksiyon değerlendirme sayısı	Ortalama hata
F9	GAWLS	-186.7309	2568	MD	MD	MD	MD
	GA-PSO	MD	MD	MD	100	96211	0.00007
	İPÖ	-186.7309 ^b	16740	0.3516	100	17460	0
F10	ACORSES	-837.9658	1176	0.0690	MD	MD	MD
	İPÖ	-837.9658 ^b	14880	0.2722	100	17500	0
F11	PSACO	MD	MD	MD	100	167	5.706*10 ⁻²⁷
	GA-PSO	MD	MD	MD	100	95	0.00005
	İPÖ	0	37016	23.3041	100	37856	0
F12	PSACO	MD	MD	MD	100	592	2.075*10 ⁻¹¹
	ACO	MD	MD	0.74 ^a	MD	528 ^a	MD
	GA-PSO	MD	MD	MD	100	2117	0.00020
	İPÖ	-3.8628 ^b	13840	0.4053	96	15700	1.015*10 ⁻¹³
F13	PSACO	MD	MD	MD	96	529	4.478*10 ⁻¹¹
	GA-PSO	MD	MD	MD	100	12568	0.00024
	ACO	MD	MD	4.10 ^a	MD	1344 ^a	MD
	İPÖ	-3.32 ^c	30400	0.7804	96	32200	2.341*10 ⁻¹⁶
F14	RW- PSO+BOF	MD	MD	MD	MD	MD	0 ^d
	İPÖ	0	33500	0.5254	100	34440	0
F15	PSACO	MD	MD	MD	100	1081	6.23*10 ⁻²²
	İPÖ	0	37010	19.8856	100	38896	0
F16	PSACO	MD	MD	MD	100	209	2.618*10 ⁻¹³
	İPÖ	0.3979 ^b	7680	0.1413	100	7900	0

MD: Mevcut değil

^a Algoritmanın 4 kez çalıştırılması neticesindeki ortalama fonksiyon değerlendirme sayısı ve standart zaman biriminden çalışma süresi. ^b Teorik en küçük değer 4 hane olarak düşünülmüştür.^c Teorik en küçük değer 2 hane olarak düşünülmüştür. ^d Algoritmanın 30 kez çalıştırılması neticesindeki ortalama sonuçlar.

Şekil 3.7'den görülebildiği gibi, ortalama fonksiyon değerlendirme sayısının değişken sayısı 1000'e eşit oluncaya kadar artmasına rağmen, değişken sayısı 1000 ile 10000 arasında iken ortalama fonksiyon değerlendirme sayısının değişimi azdır. Bu, İPÖ algoritmasının çok değişkenli fonksiyonlar için ortalama fonksiyon değerlendirme sayısı açısından global veya globale yakın en uygun çözüm değerinin bulunmasında tutarlı ve etkin bir yöntem olduğunu göstermektedir.



Şekil 3.7 Farklı değişken sayıları için ortalama fonksiyon değerlendirme sayısının değişimi

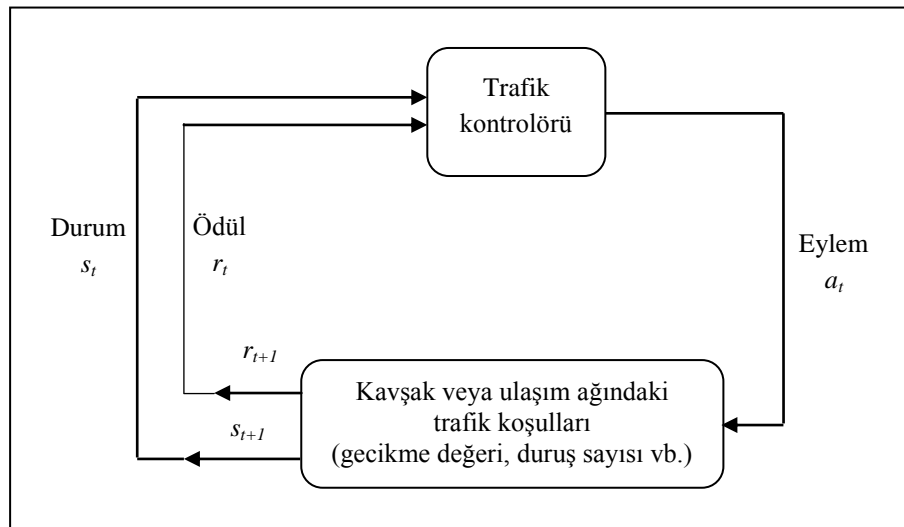
3.4. İyileştirilmiş PÖ Yönteminin Sinyal Kontrol Problemine Uyarlanması

Bu çalışmada, UAT probleminin üst seviyesi olan sinyal parametrelerinin eniyilenmesi geliştirilen İPÖ algoritması ile gerçekleştirilmiştir. Trafik sinyal parametrelerinin eniyilenmesinde farklı sezgisel yöntemler kullanılmasına rağmen PÖ yönteminin bu alanda uygulaması birkaç çalışma ile sınırlı kalmıştır. Thorpe (1997) yapmış olduğu çalışmada geçici fark öğrenme yöntemlerinden biri olan Sarsa algoritmasını simüle edilmiş trafik sinyal kontrol problemine uygulamıştır. Martin ve Brauer (2000) yapmış oldukları çalışmada PÖ yöntemi bazlı bulanık bir model önermişler ve bu modeli en uygun sinyal kontrol planı seçimi problemine uygulamışlardır. Wiering (2000) trafik sinyal kontrolünde çok ajanlı PÖ algoritmalarının kullanımı araştırmıştır. Bingham (2001) yapmış olduğu çalışmada trafik sinyal kontrol problemi için PÖ yöntemini sinirsel-bulanık yaklaşım içinde yer vererek uygulamıştır. Abdulhai vd (2003) iki fazlı ayırık sinyal kontrol sistemli bir yol ağı üzerinde basit bir Q -öğrenme algoritması uygulamışlardır.

Camponogara ve Kraus (2003) yapmış oldukları çalışmada PÖ yöntemi ile stokastik oyun teorisini kullanarak iki kavşak üzerinde trafik sinyal kontrol eniyilemesini araştırmışlardır. Sonuçlarda, önerilen algoritmanın Q -öğrenme, greedy ve rassal politika algoritmalarına göre daha iyi sonuç verdiği belirtilmiştir. Nunes ve Oliveria (2004) yapmış oldukları çalışmada, basit bir senaryo üzerinde PÖ yönteminde kullanılan ajanların öğrenme kabiliyetlerini geliştirmeye çalışan farklı teknikler uygulamışlardır. Cai vd (2009), gerçek zamanlı kendi kendini idare eden trafik uyarımalı sinyal kontrol sistemi geliştirmek amacıyla yaklaşık dinamik programlama yönteminin trafik sinyal kontrol alanına uygulamasını araştırmışlardır. Bazzan vd (2010) trafik sinyallerinin kontrolü için çok ajanlı bir PÖ yöntemi önermişler ve elde edilen sonuçları tek ajanlı PÖ yönteminin sonuçları ile karşılaştırmışlardır. Çalışmanın sonucunda, çok ajanlı PÖ yönteminin tek ajanlı PÖ yöntemine göre daha iyi sonuçlar verdiği belirtilmiştir.

Yukarıda bahsedilen çalışmalardan PÖ yönteminin ayrık sinyalizasyon sistemi ile kontrol edilen kavşaklarda veya ağlarda uygulandığı anlaşılmaktadır. Ayrıca, PÖ yönteminde çözüm uzayını daraltarak eniyileme yeteneğini iyileştiren herhangi bir çalışma bulunmamaktadır.

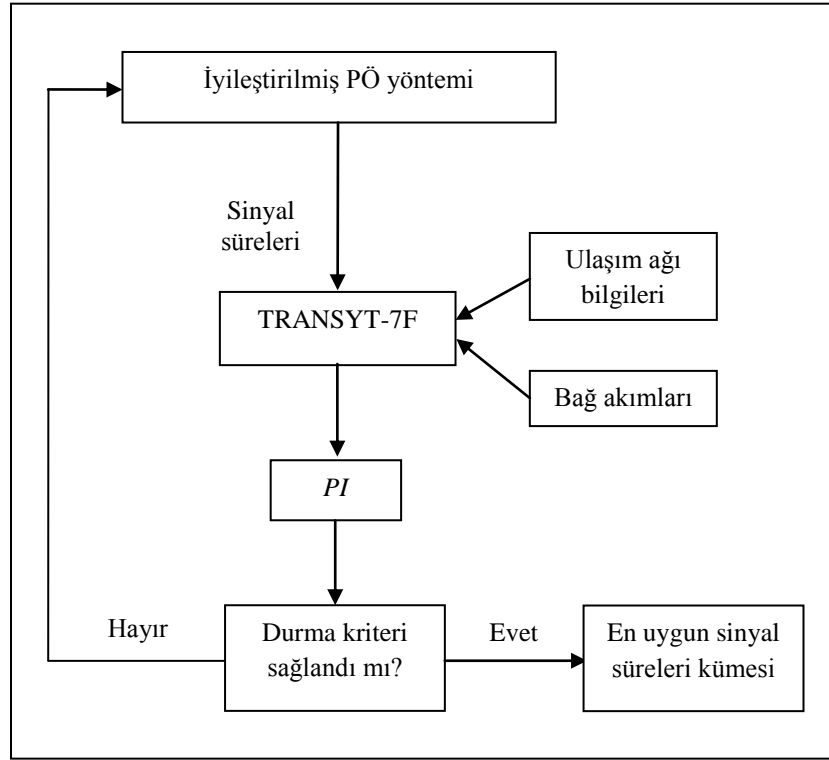
PÖ yönteminin temel çerçevesi olan ajan-çevre etkileşimi, yöntemin sinyal kontrol problemine uyarlanmasıyla Trafik Kontrolörü-Kavşak veya ulaşım ağındaki trafik koşulları olarak değiştirilmiştir. Sinyal kontrol problemine uyarlanmış PÖ yönteminin temel çerçevesi Şekil 3.8’de görülmektedir.



Şekil 3.8 PÖ yönteminin temel çerçevesinin sinyal kontrol problemine uyarlanması

PÖ yöntemi içinde bulunan ajan ya da karar verici, PÖ yönteminin temel çerçevesi sinyal kontrol probleminde uyarlanmasından sonra trafik kontrolörü olarak adlandırılmaktadır. PÖ yöntemi içinde bulunan çevre ise kavşak veya ulaşım ağındaki trafik koşullarını (gecikme değeri, duruş sayısı, performans indeksi vb.) temsil etmektedir. Trafik kontrolörü, çevredeki, bir başka deyişle kavşaktaki ya da ulaşım ağındaki trafik koşullarını gözlemleyerek kavşağın ya da ulaşım ağının durumu hakkında (kavşak veya ulaşım ağındaki gecikme değeri, duruş sayısı, performans indeksi değeri gibi) bilgi sahibi olmaktadır. Trafik kontrolörü daha sonra bu duruma göre muhtemel eylemler kümesinden bir eylem seçerek (birinci kavşağın birinci fazının yeşil süresini artır, ikinci kavşaktaki tüm fazların ofset sürelerini artır, beşinci kavşaktaki devre süresini azalt yada artır gibi) o eylemi gerçekleştirmektedir. Yapılan bu eylem kavşak ya da ulaşım ağındaki trafik koşullarını değiştirmekte, diğer bir ifadeyle kavşak veya ulaşım ağındaki gecikme değerini, duruş sayısını, ağ performans indeksi değerini azaltmakta ya da artırmaktadır. Sonuç olarak, yapılan eylem olumlu ya da olumsuz olarak kavşaktaki veya ulaşım ağındaki trafik koşullarını değiştirmektedir. Değişen trafik koşulları trafik kontrolörü tarafından sayısal sinyal olarak algılanmaktadır. Bu arada trafik kontrolörü, yapılan eylem, kavşak ya da ulaşım ağındaki trafik koşullarını olumlu yönde etkilemişse sayısal ödül değerini, olumsuz yönde etkilemişse sayısal ceza değerini almaktadır. Böylelikle, trafik kontrolörü, kavşak ya da ulaşım ağındaki trafik koşullarına göre yapılması gereken eylemi öğrenerek kavşak veya ulaşım ağından en iyi performansı alacak şekilde işletilmesini sağlamaktadır.

Sinyal kontrol probleminin çözümü için İPÖ yöntemi ile TRANSYT-7F yazılımının trafik modeli birleştirilerek iyileştirilmiş **Pekiştirmeli Öğrenme TRANSYT-7F (PÖTRANS)** modeli geliştirilmiştir. Bu modelde, İPÖ yöntemi sabit bağ akımlarını dikkate alarak trafik sinyal parametrelerinin eniyilenmesi aşamasında kullanılırken, TRANSYT-7F yazılımının trafik modeli mevcut sinyal sürelerini ve faz planlarını dikkate alarak amaç fonksiyonu olarak adlandırılan ulaşım ağının *PI* değerinin hesaplanmasında kullanılmıştır. PÖTRANS modelinin genel çerçevesi Şekil 3.9'da gösterilmektedir. İPÖ algortimasının sinyal kontrol probleminde uyarlanmış hali Şekil 3.10'da gösterilmektedir.



Şekil 3.9 PÖTRANS modelinin genel çerçevesi

<i>Başlangıç $t=0$. β, α, γ, değişken sayısı (n), çevre boyutunu (m) ayarla</i>
<i>While $t \leq t_{max}$ (t_{max}=maksimum öğrenme evresi sayısı)</i>
<i>If $t=0$, başlangıç $Q(s, a)$ değerlerini başlangıç sinyal süreleri olarak oluştur</i>
<i>Çevrenin her boyutundaki $Q_t(s, a)$ değerleri için amaç fonksiyonunun değerini, $f(Q(s, a))$, ağ PI değeri, $PI(Q(s, a))$, olarak hesapla</i>
<i>Else</i>
<i>En iyi ağ PI değerinin veren en iyi sinyal sürelerini, $Q_{t-1}^{en\ iyi}(s, a)$, sakla</i>
<i>$\beta_t = \beta_{t-1} * 0.99$</i>
<i>Alt çevrenin $Q_t(s, a)$ değerlerini, alt çevrenin sinyal süreleri olarak Denklem (3.3)'ü kullanarak oluştur</i>
<i>Alt çevrenin her boyutundaki $Q_t(s, a)$ değerleri için amaç fonksiyonunun değerini, $f(Q(s, a))$, ağ PI değeri, $PI(Q(s, a))$, olarak hesapla</i>
<i>Çevre ve alt çevreyi ağ PI değerine göre iyiden kötüye doğru sırala</i>
<i>end if</i>
<i>En iyi ağ PI değerini veren en iyi sinyal sürelerini, $Q_t^{en\ iyi}(s, a)$, bul</i>
<i>$r_t(s, a)$ değerini Denklem (3.4)'ü kullanarak hesapla</i>
<i>Sinyal süreleri olarak $Q_t(s, a)$ değerlerini Denklem (3.2)'yi kullanarak güncelle</i>
<i>$t=t+1$</i>
<i>End while</i>

Şekil 3.10 Sinyal kontrol problemine uyarlanmış İPÖ algoritması

İPÖ algoritmasının sinyal kontrol problemine uygulanabilmesi için her bir $Q(s, a)$ değeri sinyal parametreleri cinsinden karar değişkenlerini temsil edecek şekilde algoritma oluşturulmuştur (Şekil 3.2). Her öğrenme evresinde, çevrenin her boyutunda sinyal süre kısıtlarına bağlı olarak karar değişkenleri kadar sinyal süreleri $Q(s, a)$ olarak üretilir. Şekil 3.2'den görülebileceği gibi, her öğrenme evresinde, çevrenin her boyutunda üretilen sinyal süreleri dikkate alınarak amaç fonksiyonu olan ağ PI değeri, $f(Q(s, a))$, hesaplanmaktadır. Geliştirilen PÖTRANS modelinde, $Q_t^{en\ iyi}(s, a)$ t 'inci öğrenme evresinde çevre içinde en iyi ağ PI değerini veren en iyi sinyal süreleri olarak tanımlanmaktadır.

TRANSYT-7F yazılımının içinde PI değeri için birden fazla amaç fonksiyonu bulunmakta ve amaç fonksiyonu kullanıcı tarafından seçilebilmektedir. Amaç fonksiyonlarından biri standart TRANSYT-7F Disutility Index'dir (DI). DI değeri, ağın işletim dezavantajının bir ölçütü olup, ağıdaki gecikmelerin, duruş sayılarının toplamıdır (McTrans, 2008). PÖTRANS modelinde amaç fonksiyonu olarak TRANSYT-7F yazılımının DI değeri seçilmiştir. PÖTRANS modelinin amaç fonksiyonu ve kısıtları Denklem (3.8)'deki gibidir.

$$PI = \underset{\psi, \mathbf{q}}{\text{Min}} DI = \sum_{a \in L} [w_{d_a} \cdot d_a(\psi) + K \cdot w_{s_a} \cdot S_a(\psi)] \quad (3.8)$$

$$\text{Kısıtlar; } \quad \psi(c, \theta, \varphi) \in \Omega_0$$

$$c_{\min} \leq c \leq c_{\max} \quad \text{devre süresi kısıtı}$$

$$0 \leq \theta \leq c \quad \text{ofset değerleri için kısıt}$$

$$\varphi_{\min} \leq \varphi \leq c \quad \text{yeşil süre kısıtı}$$

$$\sum_{i=1}^z (\varphi + I)_i = c$$

Burada, L bağ sayısını, d_a tş-sa/sa cinsinden a bağı üzerindeki gecikme değerini, K taşıt duruş ceza faktörünü, S_a bir saniyede a bağı üzerindeki taşıt-duruş sayısını, w_{d_a} ve w_{s_a} sırasıyla gecikme değeri ve duruş sayısının ağırlık katsayılarını, \mathbf{q} bağ akımları vektörünü, ψ sinyal parametreleri vektörünü, c ortak devre süresini (sn), θ

ofset süresini (sn), φ yeşil süreyi (sn), Ω_0 sinyal süreleri için uygun çözüm kümesini, I yeşillerasarı süreyi (sn) ve z yol ağındaki her bir sinyalizasyon kavşağına ait faz sayısını göstermektedir.

Yeşil süreler, Ceylan (2002) tarafından geliştirilen Denklem (3.9) yardımıyla elde edilir.

$$\varphi_i = \varphi_{\min,i} + \frac{\varphi_i^*}{\sum_{k=1}^z \varphi_i^*} (c - \sum_{k=1}^z I_k - \sum_{k=1}^z \varphi_{\min,k}) \quad i=1,2,\dots,z \quad (3.9)$$

Burada, φ_i , i fazı için yeşil süreyi (sn), $\varphi_{\min,i}$, i fazı için en küçük yeşil süreyi, φ_i^* , i fazı için çevre içerisinde kısıtları sağlayan rasgele üretilen yeşil süreyi (sn), z faz sayısını ve I fazlar arasındaki yeşillerasarı süreyi, c ağ için ortak devre süresini göstermektedir.

PÖTRANS modelinin çözüm adımları şu şekildedir:

- **Adım 0:** Başlangıç $t=0$. Ulaşım ağı bilgilerini, bağ akımlarını, kullanıcıya özel PÖ parametrelerini; öğrenme oranı (α), azaltma faktörü (γ), en büyük öğrenme evresi sayısı (t_{maks}), çevre boyutunu (m), her değişken için çözüm uzayı parametresini (β) gir. Değişken sayısını (n) ve kısıtları sinyal parametrelerine (ψ) bağlı olarak tanımla.
- **Adım 1:** $t=1$ ise ($m \times n$) boyutundaki orijinal çevre içinde başlangıç sinyal parametrelerini, $\psi(c, \theta, \varphi)$, $Q(s, a)$ olarak üret.
- **Adım 2:** Adım 1'de orijinal çevre içinde üretilen başlangıç sinyal parametrelerinden yeşil süreleri Denklem (3.9)'a göre hesapla.
- **Adım 3:** Sinyal parametrelerini TRANSYT-7F trafik modeline gir.
- **Adım 4:** Orijinal çevre içindeki her boyut için TRANSYT-7F trafik modelini çalıştır.
- **Adım 5:** Orijinal çevre içindeki her boyut için TRANSYT-7F trafik modelinden ulaşım ağı PI değerlerini al.

- **Adım 6:** $t > 1$ ise $(m \times n)$ boyutunda yeni bir orijinal çevre içinde yeni başlangıç sinyal parametreleri üret, bir önceki öğrenme evresinde $(m \times n)$ boyutundaki orijinal çevre içinde en iyi PI değerini sağlayan sinyal parametrelerini $(m+1)$ 'inci satırda sakla, orijinal çevre boyutunda $(m+2)$ 'nci satırdan $(2m+1)$ 'nci satıra kadar önceki öğrenme evresinde elde edilen en iyi sinyal parametrelerini ve β vektörünü kullanarak Denklem (3.3) yardımıyla alt çevreyi oluştur. Aksi takdirde Adım 11'e git.

- **Adım 7:** Adım 6'da orijinal ve alt çevre içinde üretilen sinyal parametrelerinden yeşil süreleri Denklem (3.9)'u kullanarak hesapla.

- **Adım 8:** Sinyal parametrelerini TRANSYT-7F trafik modeline gir.

- **Adım 9:** Orijinal ve alt çevre içindeki her boyut için TRANSYT-7F trafik modelini çalıştır.

- **Adım 10:** Orijinal ve alt çevre içindeki her boyut için TRANSYT-7F trafik modelinden ulaşım ağı PI değerlerini al.

- **Adım 11:** Elde edilen ulaşım ağı PI değerlerine göre tüm çevreyi iyiden kötüye doğru sırala.

- **Adım 12:** Adım 11'deki sıralama sonunda en iyi ulaşım ağı PI değerini veren sinyal parametrelerini belirle.

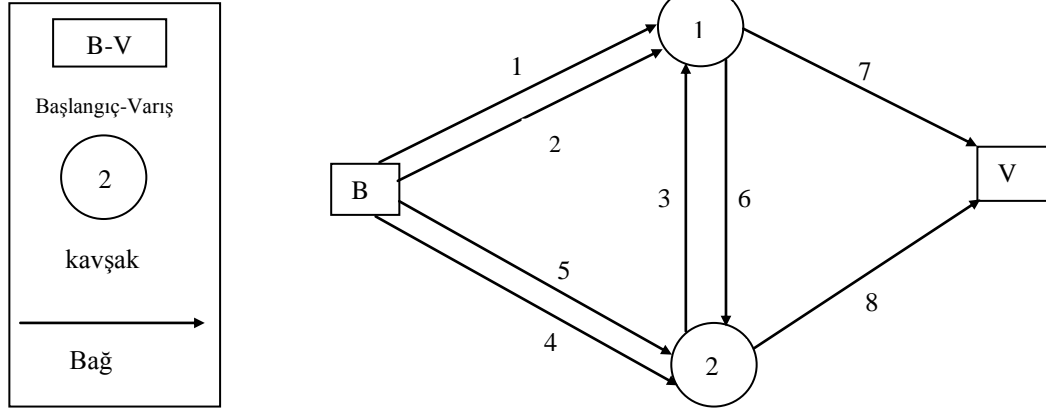
- **Adım 13:** Tüm çevre içinde Denklem (3.4)'ü kullanarak ödül fonksiyonu, $r_i(s, a)$, değerlerini hesapla.

- **Adım 14:** Sinyal parametreleri olarak $Q_i(s, a)$ değerlerini Denklem (3.2)'yi kullanarak güncelle.

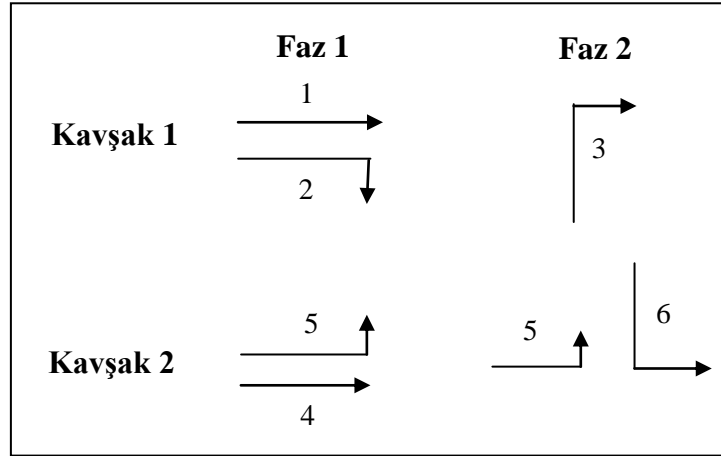
- **Adım 15:** Durma kriterini kontrol et. Eğer sağlanmışsa en iyi ψ sinyal parametrelerini yaz, aksi takdirde öğrenme evresi sayısını, t , bir arttırarak, $t = t + 1$, Adım 6'ya git.

Algoritma adımları verilen PÖTRANS modeli sabit bağ akımları ile örnek bir ulaşım ağı üzerinde uygulanmıştır. Örnek ulaşım ağında 1 adet B-V çifti, 2 adet kavşak ve 8 adet bağ bulunmaktadır. Ulaşım ağı ve kavşaklara ait fazlar sırasıyla Şekil 3.11a ve

3.11b’de görülebilmektedir. Örnek ulaşım ağına ait bağ akımları Ceylan (2002)’den alınmış ve Tablo 3.6’da verilmiştir.



Şekil 3.11a Örnek ulaşım ağı (Ceylan, 2002)



Şekil 3.11b Örnek ulaşım ağı kavşak fazları (Ceylan, 2002)

Tablo 3.6 Örnek ulaşım ağı sabit bağ akımları (Ceylan, 2002)

Bağ	q_a	s	t_a^0
1	615	1800	20
2	45	1800	20
3	225	1800	20
4	615	1800	20
5	225	1800	20
6	45	1800	20

Burada, q_a bağ akım değeri (ta/sa), s doymun akım değeri (ta/sa), t_a^0 serbest akım seyahat süresidir (sn).

Örnek ulaşım ağının çözümünde kullanılacak sinyal parametrelerinin alt ve üst limit değerleri aşağıda verilmiştir.

$$36 \leq c \leq 90$$

$$0 \leq \theta \leq c$$

$$7 \leq \varphi \leq c$$

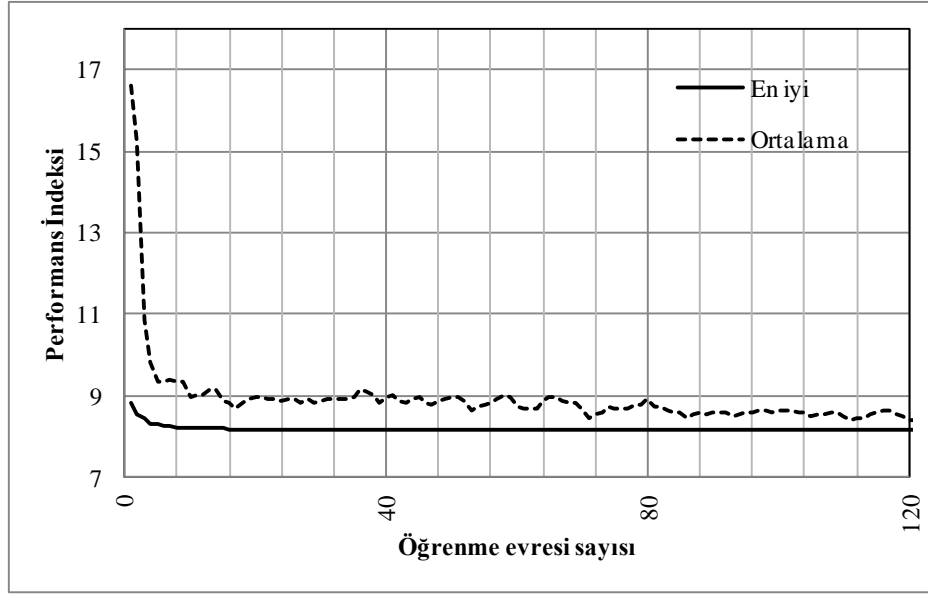
$$I_{1-2} = I_{2-1} = 5 \text{sn}$$

PÖTRANS modelinin ulaşım ağına uygulanması aşamasında İPÖ algoritması parametreleri; $\alpha = 0.8$, $\gamma = 0.2$, $m = 20$ ve $t_{\text{maks}} = 300$ olarak seçilmiştir.

Çözüm prosedürü durma kriteri sağlanıncaya kadar devam ettirilmiştir. PÖTRANS modelinde, durma kriteri t 'inci öğrenme evresinde elde edilen en iyi PI değeri ile t 'inci öğrenme evresinde elde edilen PI değerlerinin ortalaması arasındaki fark olarak tanımlanmıştır. t 'inci öğrenme evresinde ulaşım ağının en iyi PI değeri ile ortalama PI değeri arasındaki fark % 1'den az olduğunda PÖTRANS modeli sonlanmaktadır. Modelin yakınsama grafiği Şekil 3.12'de verilmiştir. Ulaşım ağı için PÖTRANS modeli 120'nci öğrenme evresinde sonlanmıştır. Bu öğrenme evresinde ulaşım ağı PI değeri 8.16'dır. PÖTRANS'da elde edilen devre süresi 76 sn'dir.

İki kavşaklı ulaşım ağı TRANSYT-7F-Genetic Algorithms ve TRANSYT-7F-Hill Climbing modelleri ile çözümlenerek PÖTRANS modelinin sonuçları ile karşılaştırılmıştır. PÖTRANS modelinin sonuçları ile TRANSYT-7F GA modelinin sonuçları karşılaştırmak için GA parametrelerinden populasyon büyüklüğü 20 ve en büyük jenerasyon sayısı da 300 olarak seçilmiştir.

TRANSYT-7F HC eniyileme modelinde yazılım enküçük devre süresi ile en büyük devre süresi arasındaki tüm tamsayı devre süreleri için benzetim yapmaktadır. İki kavşaklı ulaşım ağı için PÖTRANS modeli ve TRANSYT-7F yazılımının eniyileme modellerinin sonuçları Tablo 3.7'de verilmektedir.



Şekil 3.12 Örnek ulaşım ağında PÖTRANS modelinin yakınsaması

Tablo 3.7 Örnek ulaşım ağı için en iyi PI değeri ve sinyal parametreleri

	PI	Devre süresi c (sn)	Kavşak numarası i	Faz yeşil süreleri (sn)		Ofset süresi θ_i (sn)
				Faz 1 $\varphi_{i,1}$	Faz 2 $\varphi_{i,2}$	
PÖTRANS	8.16	76	1	50	16	0
			2	59	7	57
TRANSYT-7F HC	8.18	78	1	50	18	0
			2	61	7	0
TRANSYT-7F GA	8.17	79	1	53	16	0
			2	62	7	6

Tablo 3.7’den görülebildiği gibi, TRANSYT-7F yazılımının HC ve GA eniyileme modelleri ile elde edilen en iyi PI değerleri sırasıyla 8.18 ve 8.17 olup bu değerleri sağlayan devre süreleri de sırasıyla 78 ve 79 sn’dir. PÖTRANS modeli ile elde edilen en iyi PI değeri 8.16’dır. Elde edilen sonuçlardan, geliştirilen PÖTRANS modelinin trafik sinyal kontrol problemine uygulanabileceği gösterilmiştir.

PÖTRANS modelini test etmek için orta ölçekli bir ağ seçilip uygulanmıştır. Allsop ve Charlesworth (1977) ve Ceylan (2002) tarafından kullanılan ulaşım ağı 23 adet bağ ve 6 adet sinyal kontrollü kavşaktan oluşmaktadır. Ulaşım ağı ve kavşaklara ait fazlar sırasıyla Şekil 3.13 ve 3.14’de verilmiştir.

Ulaşım ağının çözümü için kullanılacak sinyal parametrelerinin alt ve üst limit değerleri aşağıda verilmiştir.

$$36 \leq c \leq 140$$

$$0 \leq \theta \leq c$$

$$7 \leq \varphi \leq c$$

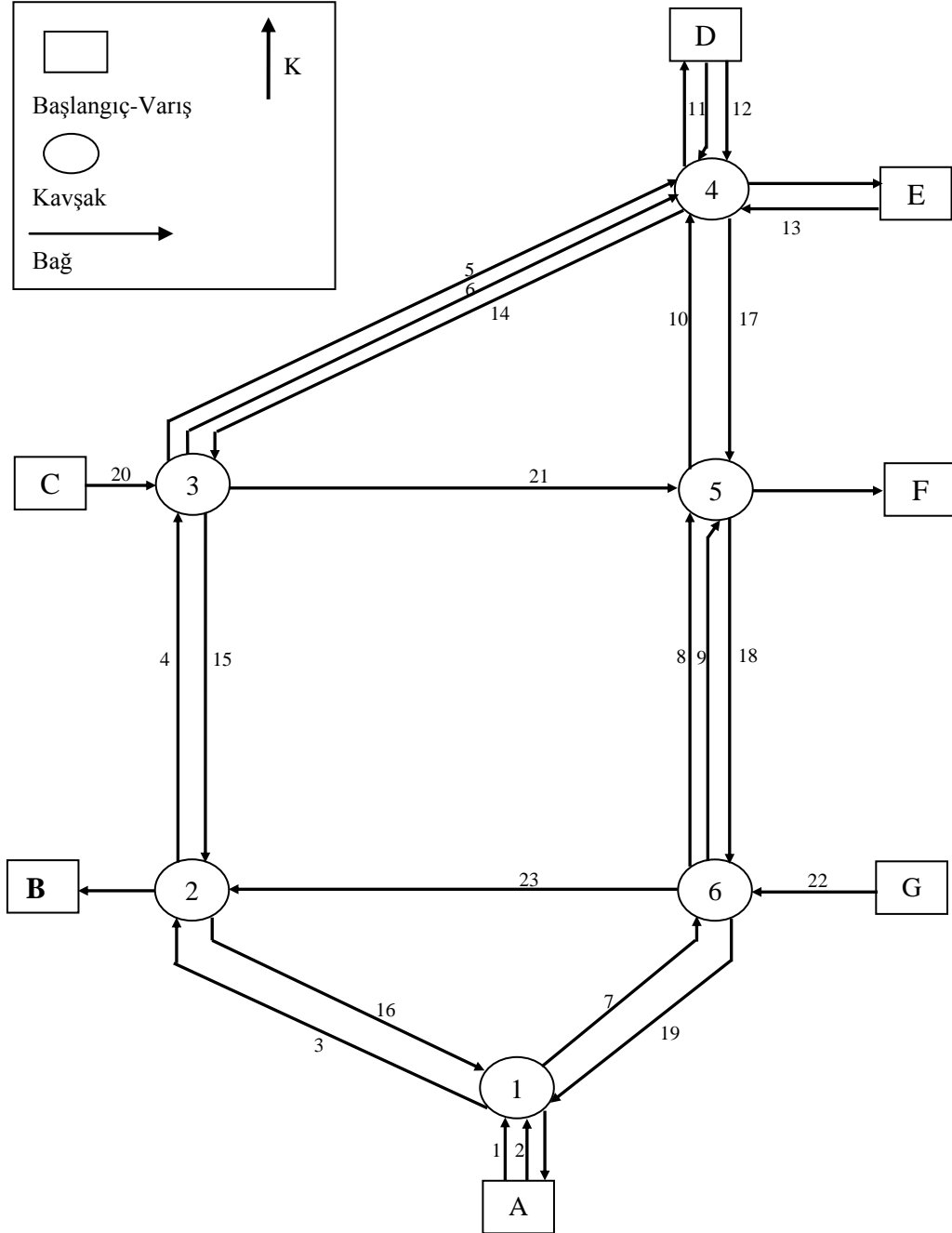
$$I_{1-2} = I_{2-1} = 5 \text{sn}$$

Bu ulaşım ağına ait sabit bağ akımları Ceylan (2006) tarafından yapılmış çalışmadan alınmış ve Tablo 3.8’de verilmiştir.

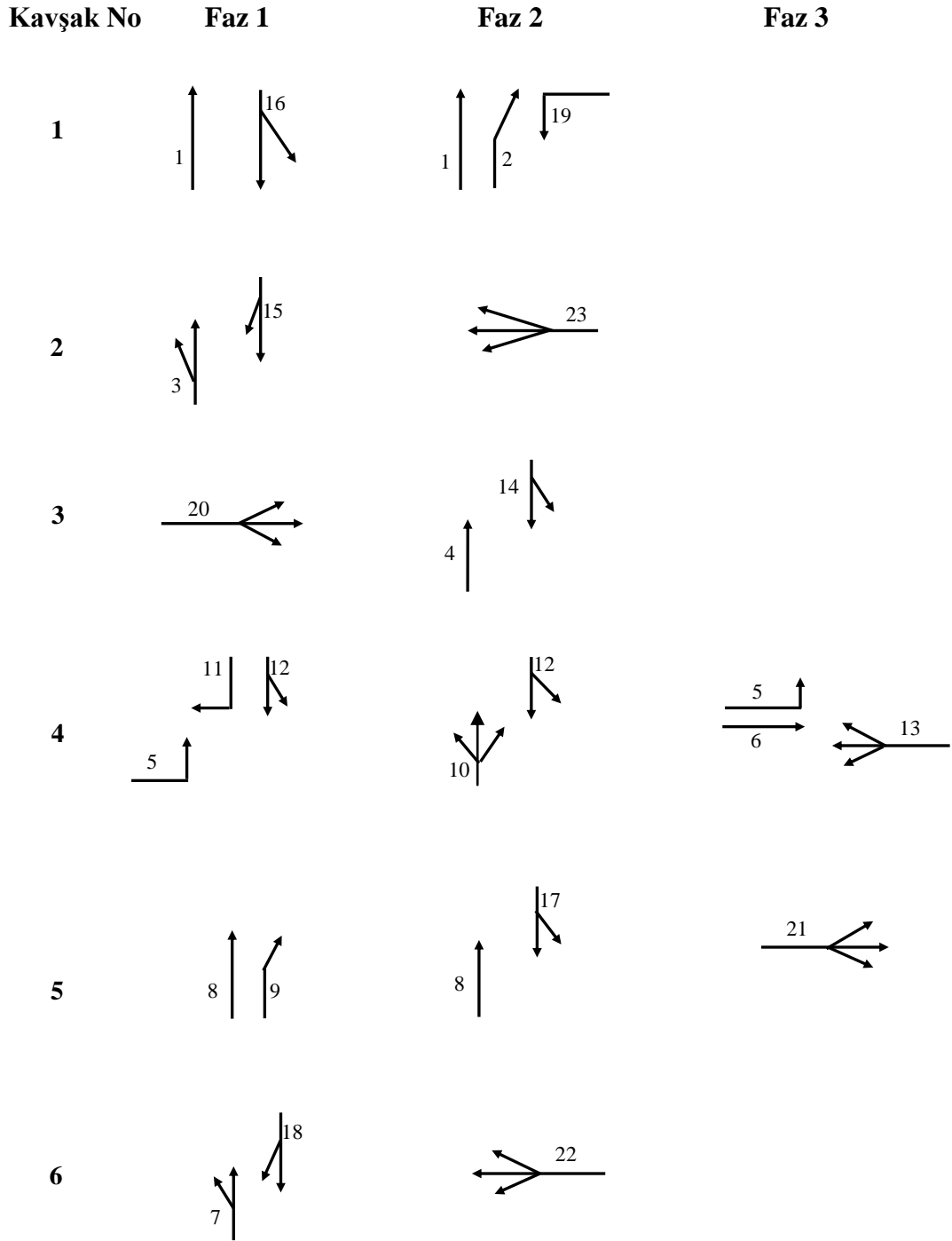
Tablo 3.8 Allsop ve Charlesworth ulaşım ağı sabit bağ akımları (Ceylan, 2006)

Bağ	q_a	s	t_a^0
1	716	2000	1
2	463	1600	1
3	716	3200	10
4	569	3200	15
5	636	1800	20
6	173	1850	20
7	462	1800	10
8	478	1850	15
9	120	1700	15
10	479	2200	10
11	499	2000	1
12	250	1800	1
13	450	2200	1
14	789	3200	20
15	790	2600	15
16	663	2900	10
17	409	1700	10
18	350	1700	15
19	625	1500	10
20	1290	2800	1
21	1057	3200	15
22	1250	3600	1
23	837	3200	15

Burada, q_a bağ akım değeri (ta/sa), s doygun akım değeri (ta/sa), t_a^0 serbest akım seyahat süresidir (sn).



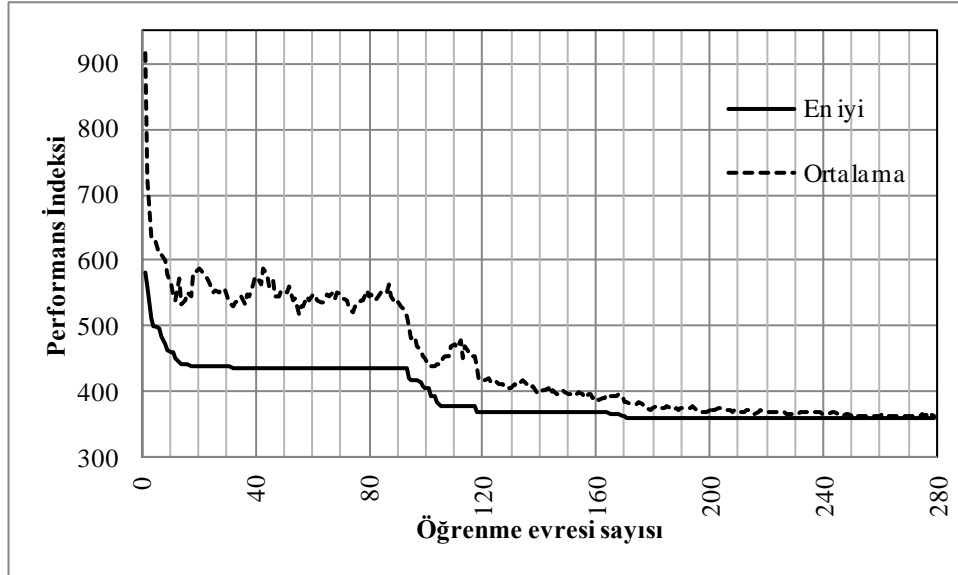
Şekil 3.13 Allsop ve Charlesworth ulaşım ağı (Ceylan, 2002)



Şekil 3.14 Allsop ve Charlesworth ulaşım ağı kavşak fazları (Ceylan, 2002)

Şekil 3.15’de PÖTRANS modelinin yakınsama grafiği verilmiştir. Şekilden görülebildiği gibi, ortalama *PI* değerleri eniyileme süreci boyunca dalgalanma göstermektedir. Bunun nedeni, algoritma sürecinde hesaplanan ortalama *PI* değerinin, hem orijinal hem de alt çevrede elde edilen *PI* değerlerinin ortalaması olmasıdır.

PÖTRANS modeli 280’inci öğrenme evresinde sonlanmış ve *PI* değeri 358.50 olarak elde edilmiştir. Bu öğrenme evresinde elde edilen devre süresi ise 109 sn’dir. PÖTRANS modelinin sonuçlarını kıyaslamak amacıyla ulaşım ağı TRANSYT-7F GA ve TRANSYT-7F HC ile çözümlenerek karşılaştırılmıştır. PÖTRANS modeli ve diğer modellerin sonuçları Tablo 3.9’da verilmektedir.



Şekil 3.15 Allsop ve Charlesworth ulaşım ağında PÖTRANS modelinin yakınsaması Tablo 3.9’dan görülebildiği gibi, TRANSYT-7F yazılımının HC ve GA eniyileme modelleri ile elde edilen en iyi *PI* değerleri sırasıyla 420.5 ve 410.0 olup bu değerleri sağlayan devre süreleri de sırasıyla 114 ve 120 sn’dir. PÖTRANS modeli ile elde edilen en iyi *PI* değeri 358.5 olup bu değeri sağlayan devre süresi de 109 sn’dir. TRANSYT-7F-GA, ulaşım ağının *PI* değerini TRANSYT-7F-HC’ye göre %2.4 iyileştirebilmiştir. Buna karşın, PÖTRANS modeli ulaşım ağı *PI* değerini, TRANSYT-7F-HC’ye göre %14.7, TRANSYT-7F-GA’ya göre %12.5 iyileştirebilmiştir. Elde edilen sonuçlardan, geliştirilen PÖTRANS modelinin, koordine sinyalizasyon sistemi ile kontrol edilen ulaşım ağlarında sabit bağ akımları ile sinyal parametrelerinin eniyilenmesinde iyi bir performans gösterdiği söylenebilir.

Tablo 3.9 Allsop ve Charlesworth ulaşım ağı için en iyi *PI* değeri ve sinyal parametreleri

	<i>PI</i>	Devre süresi c (sn)	Kav. No. i	Faz yeşil süreleri (sn)			Ofset süresi θ_i (sn)	İyileşme oranı (%)
				Faz 1 $\varphi_{i,1}$	Faz 2 $\varphi_{i,2}$	Faz 3 $\varphi_{i,3}$		
PÖTRANS	358.5	109	1	38	61	-	0	14.7
			2	63	36	-	73	
			3	58	41	-	70	
			4	35	32	27	42	
			5	10	30	54	43	
			6	31	68	-	73	
TRANSYT-7F HC	420.5	114	1	39	65	-	0	-
			2	51	53	-	98	
			3	64	40	-	98	
			4	38	31	30	98	
			5	10	31	58	98	
			6	34	70	-	98	
TRANSYT-7F GA	410.0	120	1	55	55	-	0	12.5
			2	69	41	-	89	
			3	66	44	-	37	
			4	39	33	33	106	
			5	10	33	62	75	
			6	55	55	-	55	

3.5. Sonuçlar

Bu bölümde son yıllarda farklı alanlarda uygulama alanı bulan PÖ yöntemi hakkında literatür çalışması verilmiştir. Eniyileme yeteneği çözüm uzayı daraltılarak geliştirilen iyileştirilmiş PÖ yönteminin algoritma adımları açıklanmış ve test fonksiyonları üzerinde elde edilen sonuçlar literatürdeki mevcut algoritmalarla karşılaştırılmıştır.

Literatüre bakıldığında PÖ yönteminin ayrık sinyalize sistem ile kontrol edilen kavşaklarda uygulandığı görülmüştür. PÖ yöntemi birçok alanda uygulama alanı bulmasına rağmen UAT probleminin çözümü için kullanılmasına literatürde rastlanmamıştır. Ayrıca, iyileştirilmiş PÖ yöntemi ile UAT probleminin çözümü literatürde bulunmamaktadır. Bu nedenle İPÖ yöntemi UAT probleminin üst seviyesini oluşturan sinyal kontrol probleminin çözümü için kullanılmış ve bu amaçla PÖTRANS modeli geliştirilmiştir. PÖTRANS sinyal kontrol modelinin

algoritma adımları açıklanmıştır. PÖTRANS modeli sabit trafik hacimleri ile iki kavşaklı örnek ulaşım ağı üzerinde test edilerek altı kavşaklı orta ölçekli bir ulaşım ağı üzerinde uygulanmıştır. TRANSYT-7F yazılımının HC ve GA eniyileme modelleri ile elde edilen en iyi *PI* değerleri sırasıyla 420.5 ve 410.0'dur. PÖTRANS modeli ile elde edilen en iyi *PI* değeri 358.5'dir. TRANSYT-7F-GA, ulaşım ağının *PI* değerini TRANSYT-7F-HC'ye göre %2.4 iyileştirebilmiştir. Buna karşın, PÖTRANS modeli ulaşım ağı *PI* değerini, TRANSYT-7F-HC'ye göre %14.7, TRANSYT-7F-GA'ya göre %12.5 iyileştirebilmiştir. Elde edilen sonuçlardan sabit bağ akımları altında PÖTRANS modelinin ulaşım ağları üzerindeki sinyal kontrol problemini çözebildiği anlaşılmıştır. 4. bölümde UAT modelinin alt seviyesini oluşturan TA probleminin çözümü için kullanılan benzetim bazlı DTA modeli Dynasmart-P yazılımı tanıtılarak DKD altında UAT problemini çözmek için oluşturulan PÖTRAD modeli ve çözüm algoritması hakkında detaylı bilgi verilmiştir.

4. PROBLEM FORMÜLASYONU VE MODEL GELİŞİMİ

4.1. Giriş

Bu bölümde Dinamik Kullanıcı Dengesi (DKD) altında UAT probleminin çözümü için geliştirilen model tanıtılmıştır. UAT probleminin alt seviyesini oluşturan TA aşamasında KD altında DTA yapılmıştır. Bu amaçla, UAT probleminin alt seviyesi olan TA aşamasında **Dynamic Network Assignment-Simulation Model for Advanced Roadway Telematics Planning version (Dynasmart-P)** yazılımı kullanılmıştır.

Bölüm 4.2’de Dynasmart-P benzetim bazlı mesoskopik dinamik trafik atama yazılımı hakkında bilgi verilerek yazılımın çözüm algoritması açıklanmıştır. Bölüm 4.3’de UAT probleminin üst seviyesi olan sinyal kontrol aşaması için geliştirilen PÖTRANS modeli ile UAT probleminin alt seviyesi olan TA aşaması için kullanılan Dynasmart-P yazılımının etkileşimli çalışması için önerilen iyileştirilmiş **Pekiştirmeli Öğrenme TRANSYT-7F Dynasmart-P (PÖTRAD)** modeli ve çözüm algoritması hakkında bilgi verilmiştir. Daha sonraki bölümde önerilen PÖTRAD modelinin örnek ulaşım ağı üzerinde uygulaması yapılmış ve elde edilen sonuçlar irdelenmiştir.

4.2. Dynasmart-P Benzetim Bazlı Dinamik Trafik Atama Yazılımı

Dynasmart-P yazılımı ilk olarak Maryland Üniversitesi tarafından geliştirilen ve şu anda Amerika Federal Otoyol İdaresi (FHWA) tarafından desteklenen benzetim bazlı DTA yazılımıdır. Son sürüm olan Dynasmart-P sürüm 1.3.0 2007 yılında yayınlanmıştır. Bu tez çalışmasında benzetim bazlı DTA yazılımı olan Dynasmart-P kullanılarak KD altında dinamik trafik ataması yapılarak ulaşım ağı tasarımı yapılacaktır.

Dynasmart-P ierdiđi senaryo menüsü ile kontrol parametreleri, ücretlendirme sistemleri gibi ađ performansını etkileyici unsurların ATK üzerindeki etkilerini ortaya koyabilmektedir. Yazılımın en önemli özelliklerinden birisi DTA yapabilmesidir (Dynasmart-P, 2007).

Dynasmart-P, benzetim bazlı DTA özelliđiyle ulaşım ađı planlaması ve trafik yönetim kararlarını desteklemektedir. Bu yazılım, planlama uygulamaları için talep tahmin prosedürleri ile birlikte kullanılan DTA modellerini ve trafik yönetim alıřmaları için kullanılan trafik benzetim modellerini bir araya getirmektedir.

Dynasmart-P ađ kullanıcılarının seyahatleri esnasında rota seimi kararları sonucu ulaşım ađı üzerinde oluşan trafik akımlarının deđişimini dinamik olarak temsil edebilmektedir. Bu durum sayesinde planlama alıřmalarında kullanılan statik TA yazılımlarına ait dezavantajların üstesinden gelinebilmektedir.

Dynasmart-P, dört aşamalı ulaşım planları ile kolayca koordine olabilen yeni ulaşım planlama yöntemlerini tanımlayabilmektedir. Ayrıca, yazılım trafiđin zamanla deđişken olması durumunu dikkate almaktadır. Dynasmart-P ile, hız, kuyruk uzunluđu, gecikme ve tıkanıklık etkisi gibi durumların belirlenmesi ile farklı ulaşım planlama stratejilerinin çevresel ve fonksiyonel etkilerinin deđerlendirilmesi yapılabilmektedir.

Dynasmart-P, özellikle ulaşım ađının temsil edilmesi ve ulaşım talebinin ađa yüklenmesi ile ilgili olarak geleneksel TA ve benzetim modelleri tarafından da yaygın şekilde kullanılan girdi verilerini gerektirmektedir. Girdi verisinin tipi ve çeşitliliđi, analiz edilen ađ ve kullanıcı tarafından belirlenen hassasiyet düzeyi ile deđişmektedir. Ađın karmaşıklığı; otoyollar, yüksek yoğunluklu řeritler, katılım kontrolü, arterler üzerindeki sinyal kontrollü kavşaklar ve muhtemel kaza olaylarının mevcut olup olmaması durumuna göre deđişiklik gösterebilmektedir.

Dynasmart-P trafik analizi yaparken kullanıcılara kolaylık sađlamak için oldukça geniş kapsamlı ıktı verisi üretmektedir. ıktı verisi; hacim, hız, toplam seyahat süresi, ortalama seyahat süresi, toplam seyahat mesafesi, ortalama seyahat mesafesi, ađda kalan taşıt sayısı (analiz periyodu içinde tıkanıklık ve yoğunluktan varış noktasına varamayan taşıt sayısı) ve gecikme gibi trafik mühendisleri tarafından sık şekilde kullanılan parametreleri iermektedir. Ayrıca, Dynasmart-P analizlerde oldukça faydalı olan her taşıtın rota dosyasını da oluřturmaktadır. İlave olarak, grafik

arayüzü vasıtasıyla Dynasmart-P, benzetim sonuçlarını ve diğer karakteristikleri çeşitli grafik formatları ile hem statik hem de dinamik olarak göstermek amacıyla kullanıcıya seçenekler sunmaktadır (Dynasmart-P, 2007).

Bu çalışmada DKD altında UAT probleminin çözümü amaçlanmıştır. Benzetim bazlı DTA modeli olan Dynasmart-P yazılımı KD altında DTA problemini çözebilmektedir. Bu nedenle, UAT probleminin alt seviyesi olan TA probleminin çözümü için benzetim bazlı DTA modeli olan Dynasmart-P yazılımı seçilip kullanılmıştır.

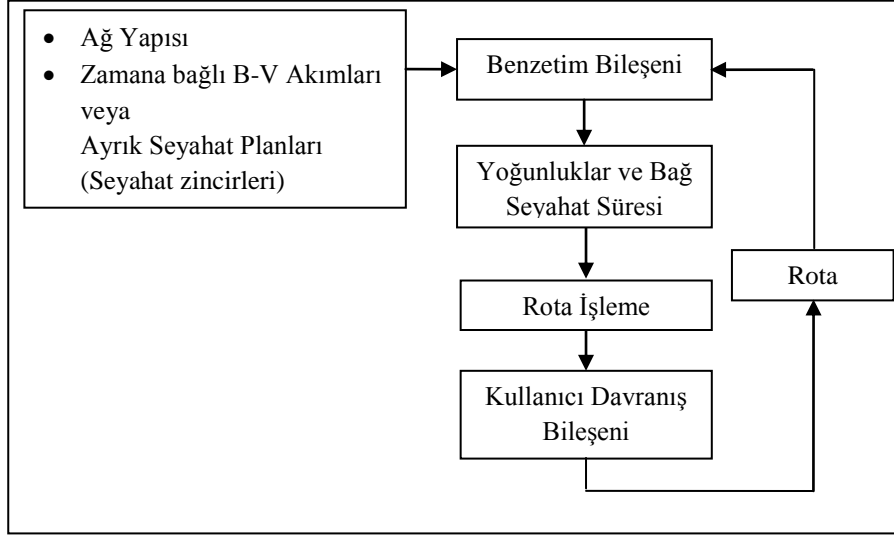
4.2.1. Dynasmart-P Algoritması

Dynasmart-P iki farklı modda kullanılabilir. Bu kullanım farklılığı, temel olarak uygulanan atama yönteminin bileşenlerinden kaynaklanmaktadır. Birinci modda, taşıtlar seçilen rotalama seçeneğine göre o anki en iyi rotalara veya önceden kararlaştırılmış rotalara atanmaktadır. İkinci modda, sürekli iteratif atama prosedürü ile KD ataması uygulanmaktadır. Birinci modda bir kere benzetim yapılarak atama tamamlanırken ikinci modda iteratif yaklaşım ile KD durumu sağlanıncaya kadar benzetim yapılmaktadır.

- **Mod 1 (Bir Adımlı Benzetim-Atama Prosedürü)**

Bu modda, Dynasmart-P sabit zaman adımlı simülasyon-atama modeli kullanılmaktadır. Trafik kontrol sistemi ile verilen ulaşım ağının zamana bağlı talepler altında performansının değerlendirilmesi ve ağ üzerindeki trafik yapılarının modellenmesi için tasarımılandırılmıştır. Modelleme yaklaşımında, trafik akım simülatörü, ağın rotalama bileşeni, kullanıcı davranışları birleştirilmektedir.

Mod 1'de, Dynasmart-P iki farklı talep konfigürasyonundan yararlanabilmektedir. Zamana bağlı B-V taşıt talebi ve her seyahat eden için ayrı seyahat planları. Birinci durumda, Dynasmart-P, zamana bağlı taşıt talebine bağlı olarak taşıtları yüklemekte ve taşıtlar varış noktalarına varıncaya kadar onları hareket ettirmektedir. İkinci durumda, Dynasmart-P, taşıtları, seyahat zincirindeki seyahat planlarına göre varış noktalarına ulaşıncaya kadar hareket ettirmektedir. Bir taşıt ağa dahil olduğunda, taşıt kullanıcı davranış kurallarına bağlı olarak ya o anki en iyi rotaya ya da seçilmiş rota kümesinin içinden rastgele bir rotaya atanmaktadır. Şekil 4.1 Dynasmart-P'nin mod 1'deki yapısını göstermektedir (Dynasmart-P, 2007).



Şekil 4.1 Dynasmart-P Bir adımlı benzetim-atama prosedürü (Dynasmart-P, 2007)

• **Mod 2 (İteratif Benzetim Bazlı Atama Prosedürü)**

Dynasmart-P, ulaşım ağındaki zamana bağlı akım yapısına göre KD şartlarını sağlayan çözümü bulabilmektedir. Bu algorithmada kullanılan değişken ve notasyonlar aşağıda verilmektedir (Dynasmart-P, 2007).

Değişken ve notasyonlar:

i = başlangıç düğümlerinin alt simgesi, $i \in I$,

j = varış düğümlerinin alt simgesi, $j \in J$,

t = o anki zaman aralığını belirten altsimge, $t = 1, \dots, T$,

h = seyahat edenlerin başlangıç noktasındaki seyahat yapısını (aynı varış düğümüne giden, aynı varış süresini tercih eden) belirten altsimge, $h \in H$,

τ = seyahat eden taşıtların harekete geçtiği zaman aralığını belirten altsimge, $\tau = 1, \dots, T1$,

k = ağ içinde i başlangıç düğümünden başlayan seyahatin rotası için altsimge,

r_{ih}^{τ} = τ harekete geçiş zaman aralığı boyunca i başlangıç düğümünden başlayan h seyahat yapısına sahip seyahatlerin sayısı,

r_{ijk}^{τ} = i başlangıç düğüm noktasından j varış düğüm noktasına hareket eden k rotasına τ harekete geçiş zaman aralığında atanarak seyahat edenlerin sayısı,

y_{ijk}^τ i başlangıç düğüm noktasından j varış düğüm noktasına hareket eden k rotasına τ harekete geçiş zaman aralığında hep yada hiç atama algoritmasına göre atanarak seyahat edenlerin sayısı,

$T^{ta} = t$ zaman periyodunun başlangıcında a bağı üzerindeki seyahat süresi,

$x^{ta} = t$ zaman periyodunun başlangıcında a bağı üzerinde seyahat edenlerin toplam sayısı.

Her i başlangıç düğüm noktasında analiz periyodu için aynı h seyahat yapısına sahip seyahat edenlerin sayısı verilmiştir, $r_{ih}^\tau \forall i \in I, \forall h \in H \vee \forall r$. Seyahat edenler, aynı başlangıç, aynı varış noktasına, ağ içinde harekete geçiş zamanına tanımlanmaktadır. Amaç, bireysel olarak seyahat edenlerin seyahat sürelerini veya genelleştirilmiş seyahat maliyetini enküçüklemek için ağ üzerinde farklı rotalardaki taşıtların zamana bağlı atamalarını yapmaktır. Diğer bir ifadeyle, amaç, τ harekete geçiş zaman aralıklarında, k rotaları üzerinde h seyahat yapısında seyahat eden taşıtların r_{ijk}^τ sayısını bulmaktır.

Şekil 4.2'de Dynasmart-P yazılımının iteratif benzetim bazlı atama prosedürünün akış diyagramı verilmektedir. Verilen çözüm algoritması ile ağdaki trafik etkileşimleri gösterilebilmekte ve sistem performansı değerlendirilebilmektedir. Algoritmanın adımları aşağıda tanımlanmıştır (Dynasmart-P, 2007).

Adım 0. Başla. İterasyon sayısını $iter = 0$ 'a ayarla. B-V talebini, $r_{ih}^\tau \forall i, \tau$ ve k , başlangıçtaki uygun rotalar kümesine $k \in k_{ij}$ ata. Burada j, h seyahat planındaki ilk varış düğümüdür. Bu nedenle, başlangıç çözümü $r_{ijk}^{\tau,0}, \forall i, h, \tau$ ve k ile verilmektedir.

Adım 1. Ağ üzerinde harekete geçiş zamanı ve rota atamaları, $r_{ihk}^{\tau,iter}$, altında, bağı seyahat süreleri de $T^{ta}, \forall t, a$ dahil olacak şekilde ilgili ağ performans ölçütlerini elde etmek için trafik ağ benzetimini uygula. Her düğüm noktasındaki yeni talebi hesapla $r_{ij}^{\tau,iter} = \sum_k r_{ijk}^{\tau,iter} \forall i, h, \tau$ ve k .

Adım 2. Ağ üzerinde her τ hareket ediş zaman aralığı için, her bir B-V çifti arasındaki en küçük seyahat süresine veya en küçük genelleştirilmiş seyahat maliyetine sahip rota kümesini hesapla.

Adım 3. Tüm seyahat talebi için hep ya da hiç atamasını uygula $r_{ij}^{\tau,iter}$. Bu, ağ üzerinde her bir τ harekete geçiş zaman aralığı için rotalar üzerinde hep yada hiç atama algoritmasına göre atanarak seyahat eden taşıtların sayısını vermektedir $y_{ijk}^{\tau,iter}$, $\forall i, j$ ve τ .

Adım 4. k^* 'in rota kümesinde, $k^* \in k_{ij}$, olup olmadığını kontrol ederek rotayı güncelle, eğer küme içinde değilse dahil et, $\forall i$ ve h . Yeni iterasyon, $r_{ijk}^{\tau,iter+1}$, için atamalar Denklem (4.1) ile formülasyonu verilen Ardışık Ortalamalar Yöntemi kullanılarak elde edilmektedir, $\forall i, h, \tau$ ve k .

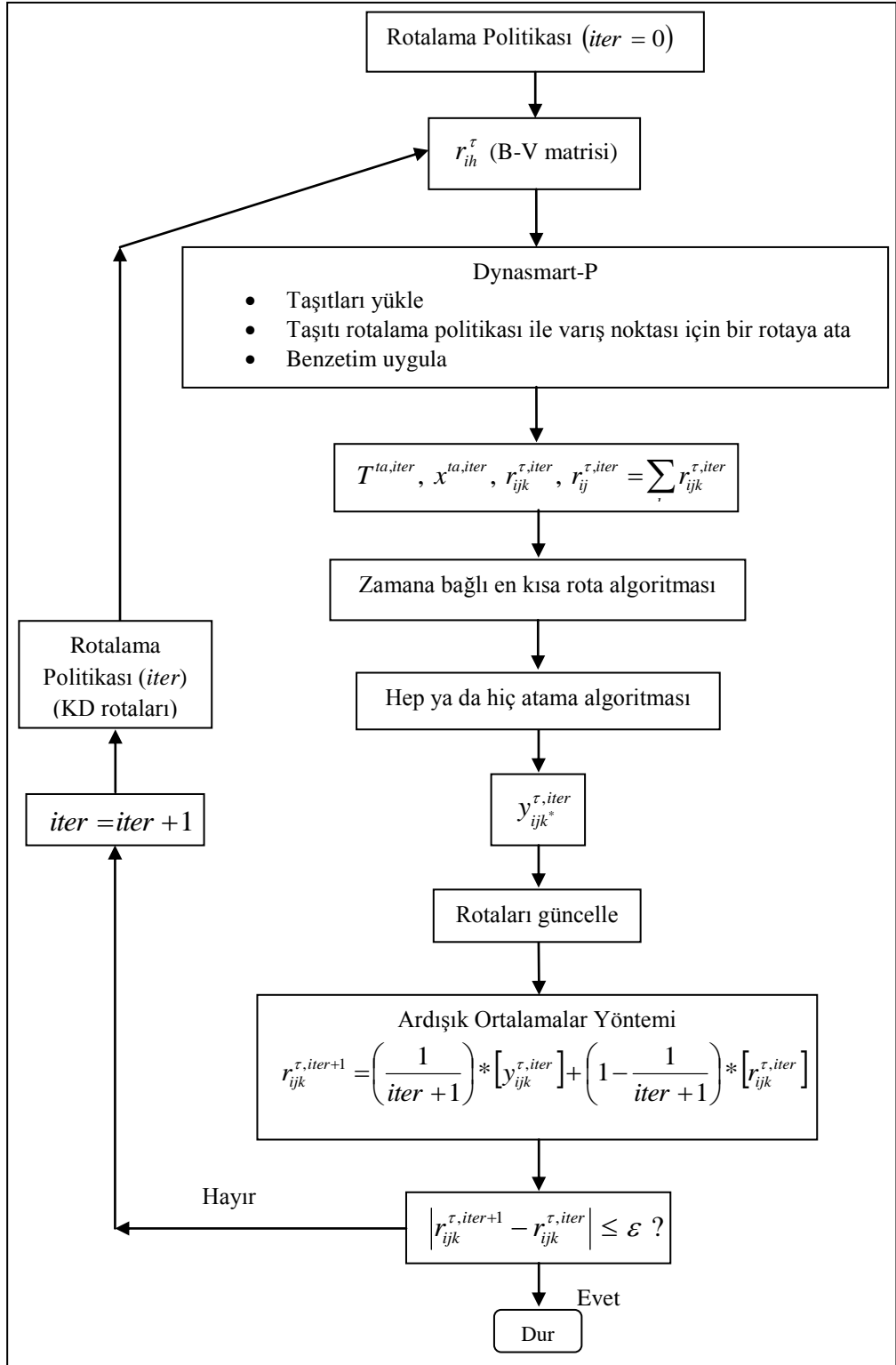
$$r_{ijk}^{\tau,iter+1} = \frac{1}{(iter+1)} * [y_{ijk}^{\tau,iter}] + \left(1 - \frac{1}{(iter+1)}\right) * [r_{ijk}^{\tau,iter}] \quad (4.1)$$

Adım 5. İki ardışık iterasyondan sonra, ağ üzerinde harekete geçiş zaman aralıklarına göre rotalara atanan taşıt sayılarındaki farklılıklara bağlı olarak Denklem (4.2)'de verilen yakınsama kriterini kontrol et. Bir sonraki iterasyon sonundaki rota atama sonuçları, $r_{ihk}^{\tau,iter+1}$, mevcut rota atamaları $r_{ijk}^{\tau,iter}$ $\forall i, j, \tau$ ve k ile karşılaştırılmaktadır:

$$\left| r_{ijk}^{\tau,iter+1} - r_{ijk}^{\tau,iter} \right| \leq \varepsilon \quad (4.2)$$

Burada, ε önceden tanımlanmış bir yakınsama sınırıdır.

Adım 7. Yakınsama kriteri sağlandıysa algoritmayı durdur. Atama probleminin çözümü $r_{ijk}^{\tau,iter}$ çıktısıdır. Yakınsama kriteri sağlanmadıysa, iterasyon sayısını artır ($iter = iter + 1$) ve yeni rota atamaları, $r_{ijk}^{\tau,iter+1}$, ile Adım 1'e git.



Şekil 4.2 Dynasmart-P için çözüm algoritması (İteratif KD Prosedürü)
(Dynasmart-P, 2007)

4.2.2. Zamana Bağlı En Kısa Rota Algoritması

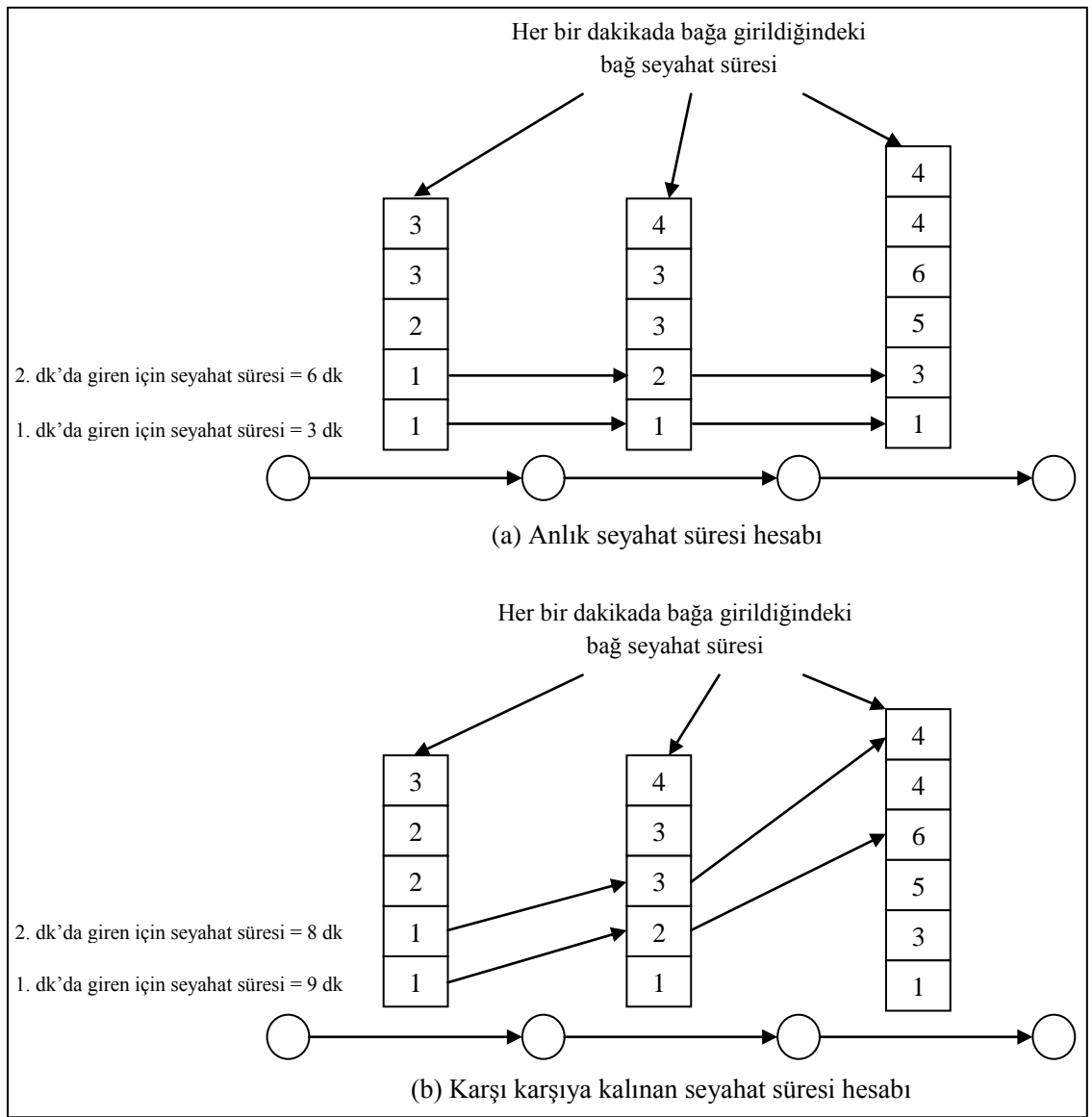
Benzetim bazlı dinamik trafik atama modelleri Zamana Bağlı En Kısa Rota (Time Dependent Shortest Path-TDSP algorithm) algoritmasını kullanmaktadırlar. Zamana bağlı en kısa rota algoritmasında *karşı karşıya kalınan seyahat süresi* ve *anlık seyahat süresi* olmak üzere iki kavram bulunmaktadır.

Karşı karşıya kalınan seyahat süresi, kullanıcıların rota seçim kararları ile tutarlı DKD kurulmasında anahtar rol oynamaktadır. Karşı karşıya kalınan seyahat süresi, anlık seyahat süresinden farklı bir kavramdır. Bu fark, Şekil 4.3'deki örnek ile açıklanmaktadır. Örnekteki ulaşım ağı dört adet düğüm noktası ve üç bağdan oluşmaktadır. Her bağın üzerinde istiflenmiş değerler, farklı dakikalarda bağın akım yukarı düğüm noktasından bağa giriş yapıldığında bağ katetmek için gereken süreleri göstermektedir. Zirve saatlerde bağlar üzerindeki yoğunluklar arttığından bu bağlar üzerindeki seyahat süreleri farklı olmaktadır. Örnek olarak, birinci dakikada bağ 1'in akım yukarı düğüm noktasından hareket edildiğinde bağ 1'i katetmek için gereken süre 1 dk'dır. Beşinci dakikada bağ 1'in akım yukarı düğüm noktasından hareket edildiğinde bağ 1'i katetmek için gereken süre 3 dk'dır. Benzer şekilde bağ 2 için seyahat süresi bağ 2'nin akım yukarı düğüm noktasından birinci ve ikinci dakikada hareket edildiğinde sırasıyla 1 ve 2 dk'dır. Bir rota için farklı harekete geçiş zamanlarındaki anlık seyahat süresi, rotayı oluşturan tüm bağların ilgili harekete geçiş zamanındaki bağ seyahat sürelerinin toplamı ile hesaplanmaktadır. Örnek olarak, birinci dakikada hareket eden taşıtlar için seyahat süresi $1+1+1=3$ dk olmaktadır; ikinci dakikada hareket eden taşıtlar için seyahat süresi $1+2+3=6$ dk olmaktadır (TRB, 2010).

Karşı karşıya kalınan seyahat süresi, bir bağ katetmek için gereken süreden oluşmakta ve bir bağın akım aşağı düğüm noktasına girildiği zamana bağlı olarak bağ seyahat süresi ile bulunmaktadır (düğüm noktalarından geçmenin zaman açısından bir maliyetinin olmadığı kabulü yapılmaktadır). Bu yaklaşıma bağlı olarak, birinci dakikada harekete geçildiğinde bağ 1 için seyahat süresi 1 dk, bağ 2'ye ikinci dakikada varıldığı için seyahat süresi 2 dk, bağ 3'e dördüncü dakikada varıldığı için seyahat süresi 6 dk olduğundan bu rota için toplam karşı karşıya kalınan seyahat süresi $1+2+6=9$ dk olmaktadır. Benzer şekilde ikinci dakikada harekete geçildiğinde bu rota için anlık seyahat süresi 6 dk olurken, karşı karşıya kalınan seyahat süresi 8

dk olmaktadır. Bu örnekten görüleceği üzere, iki yöntem farklı rota seyahat süreleri oluşturmakta ve benzer şekilde en kısa rota(lar) için sonuçlar farklılık göstermektedir (TRB, 2010).

Anlık seyahat süresi hesabı ile elde edilen en kısa rota, harekete geçildiğinde geçerli olan anlık bağ seyahat sürelerine bağlı olmaktadır. Fakat, yoğunluk veya tıkanıklıktan dolayı bağ seyahat süreleri dinamik olarak değişmekte ve rota ister istemez en küçük anlık seyahat süresi ile sonuçlanmamaktadır. Çünkü bu yöntemin yol üzerinde zaman içinde meydana gelebilecek tıkanıklığı öngörebilecek özelliği bulunmamaktadır.



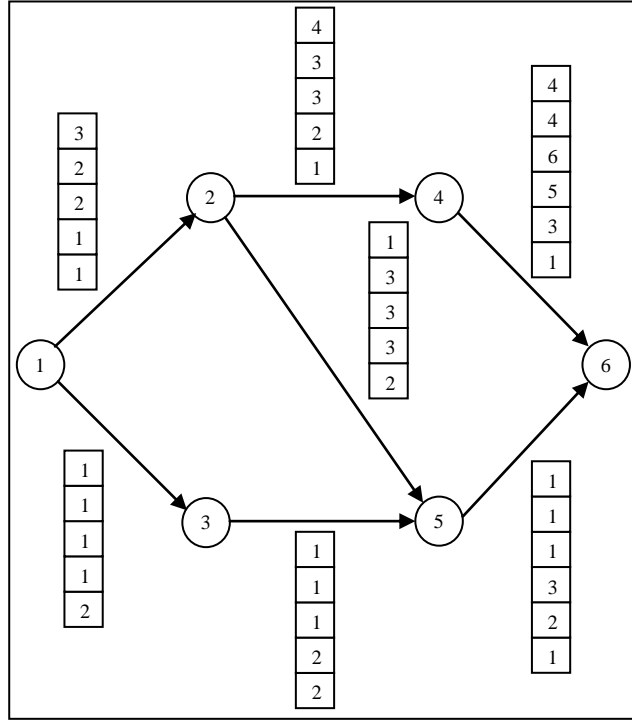
Şekil 4.3 Karşı karşıya kalınan seyahat süresi ile anlık seyahat süresinin belirlenmesi (TRB, 2010).

Anlık rota seyahat süresi ile TA yapılması tamamen yanlıştır denilememektedir, ancak anlık seyahat süresi ile ilgili kabullerin anlaşılması gerekmektedir. Anlık seyahat süresi ile rota seçimi yapılırken: (1) kullanıcıların seyahat öncesinde web sitesi, haberler aracılığıyla veya yol boyunca taşıt bilgi sistemleri ile bilgi edinebildiği ve böylelikle kullanıcı hareket ettiğinde veya yolda hareket halindeyken hangi rotanın en kısa rota olduğu bilgisine sahip olduğu; (2) günden güne kullanıcıların deneyimlerinden yola çıkarak rota seyahat süresini değerlendirmedikleri bunun yerine kullanıcıların kendi bilgilerine güvendikleri kabulleri yapılmaktadır.

Karşı karşıya kalınan seyahat süresi hesap yönteminden elde edilen en kısa rota, karşı karşıya kalınan seyahat süresi ile zamana bağlı en kısa rotayı vermektedir. Burada, kullanıcıların, hareket ediş anında sadece görünen en kısa rota yerine karşı karşıya kaldıkları seyahat sürelerini enküçükleyen rotaları aramaya gönüllü oldukları kabulü yapılmaktadır.

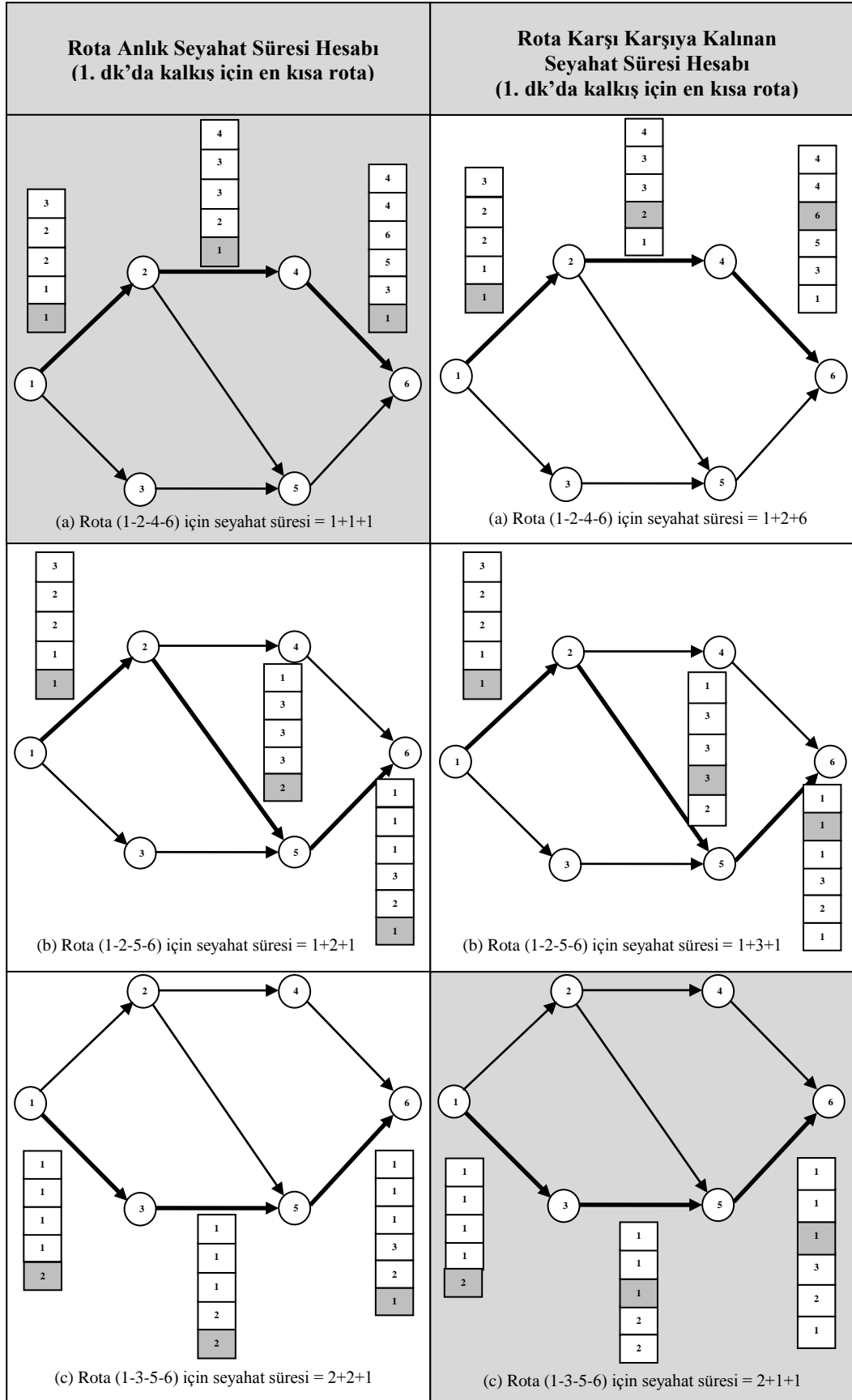
Şekil 4.4'deki örnek, anlık seyahat süresi hesabı ile karşı karşıya kalınan seyahat süresi hesabının nasıl farklı rotalar ürettiğini ispatlamaktadır. Şekil 4.4'deki ulaşım ağı, altı düğüm noktası, yedi bağ ve bir B-V çiftinden oluşmaktadır. Tüm taşıtların başlangıç düğüm noktası 1, varış düğüm noktası 6'dır. Her bağa ait zamanla değişen seyahat süreleri, bağa giriş zamanına göre dk cinsinden bağların üzerinde istiflenmiş olarak verilmektedir.

Şekil 4.4'den görülebildiği gibi 1 ile 6 nolu düğüm noktasını bağlayan üç adet rota bulunmaktadır: (1-2-4-6); (1-2-5-6); (1-3-5-6). Anlık seyahat süresi hesabı ile birinci dakikada harekete geçiş için seyahat süreleri rota (1-2-4-6) için 3 dk, rota (1-2-5-6) için 4 dk ve rota (1-3-5-6) için 5 dk olmaktadır. Sonuç olarak, rota (1-2-4-6) en kısa anlık seyahat süreli rota olmaktadır. Karşı karşıya kalınan seyahat süresi hesabı ile birinci dakikada harekete geçiş için seyahat süreleri rota (1-2-4-6) için 9 dk, rota (1-2-5-6) için 5 dk ve rota (1-3-5-6) için 4 dk olmaktadır. Sonuç olarak, rota (1-3-5-6) en kısa karşı karşıya kalınan seyahat süreli rota olmaktadır. Aynı şekilde diğer zamanlarda harekete geçiş için de hem anlık hem karşı karşıya kalınan seyahat süreli en kısa rotalar hesaplanabilmektedir. Birinci ve ikinci dakikada harekete geçiş için anlık ve karşı karşıya kalınan seyahat süreli en kısa rotalar sırasıyla Şekil 4.5 ve 4.6'da gösterilmektedir (TRB, 2010).

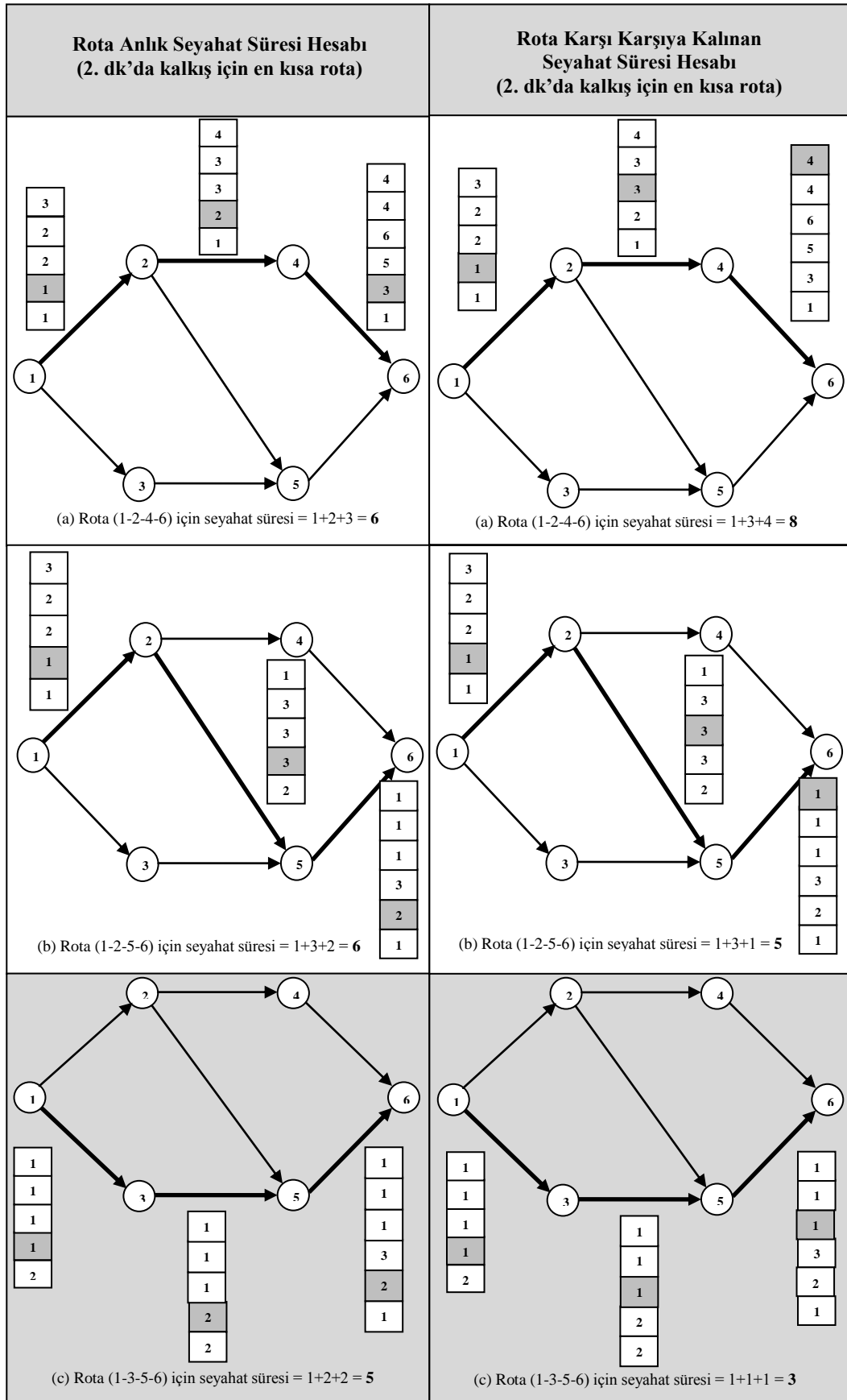


Şekil 4.4 Zamana bağlı değişen bağ seyahat süreleri ile örnek ulaşım ağı

(TRB, 2010).



Şekil 4.5 Birinci dakikada harekete geçiş için en kısa rotalar (TRB, 2010)



Şekil 4.6 İkinci dakikada harekete geçiş için en kısa rotalar (TRB, 2010)

4.3. Dinamik Kullanıcı Dengesi Altında UAT Modeli

Bu çalışmada DKD altında UAT problemini çözmek için iki seviyeli programlama tekniği kullanılmıştır. İki seviyeli programlama bir çift eniyileme probleminden oluşan programlama tekniği olarak tanımlanmaktadır. Bu tür programlamada üst seviye ve alt seviyelerin parametreleri her iki seviye için girdi ve çıktı olarak kullanılmaktadır. Bu çalışmada iki seviyeli programlama tekniğinin üst seviyesi olan sinyal kontrol probleminin çözümünde Bölüm 3.4’de anlatılan PÖTRANS modeli kullanılırken alt seviye olan TA probleminin çözümünde benzetim bazlı DTA yazılımı olan Dynasmart-P yazılımı kullanılmıştır. Bu amaçla, DKD altında UAT probleminin çözümünde kullanılmak üzere PÖTRANS sinyal kontrol modeli ile Dynasmart-P yazılımının etkileşimli çalışması için iyileştirilmiş Pekiştirmeli Öğrenme TRANSYT-7F Dynasmart-P (PÖTRAD) modeli oluşturulmuştur.

4.3.1. PÖTRAD Modeli

PÖTRANS modeli sabit bağ akımları ile trafik sinyal kontrol problemini çözerken, PÖTRAD modeli ise DKD bağ akımlarını dikkate alarak UAT problemini çözmektedir. Bir başka ifadeyle, UAT probleminin alt seviyesi olarak adlandırılan TA aşamasında Dynasmart-P yazılımını kullanarak elde edilen DKD bağ akımları, UAT probleminin üst seviyesi olan sinyal kontrol aşamasında kullanılan PÖTRANS modeline girdi olarak kullanılması ile UAT problemi çözülmektedir. PÖTRAD modelinin akış şeması Şekil 4.7’de verilmektedir.

PÖTRAD modelinin amacı, denge bağ akımlarının (\mathbf{q}^*) ve sinyal parametrelerinin ($\boldsymbol{\psi}$) bir fonksiyonu olan ağ DI değerini enküçükmektir. PÖTRAD modelinin amaç fonksiyonu ve kısıtları Denklem (4.3)’de verilmektedir.

$$PI = \underset{\boldsymbol{\psi}}{\text{Min}} DI(\mathbf{q}^*(\boldsymbol{\psi}), \boldsymbol{\psi}) \quad (4.3)$$

$$\text{Kısıtlar} \quad \boldsymbol{\psi}(c, \boldsymbol{\theta}, \boldsymbol{\varphi}) \in \boldsymbol{\Omega}_0$$

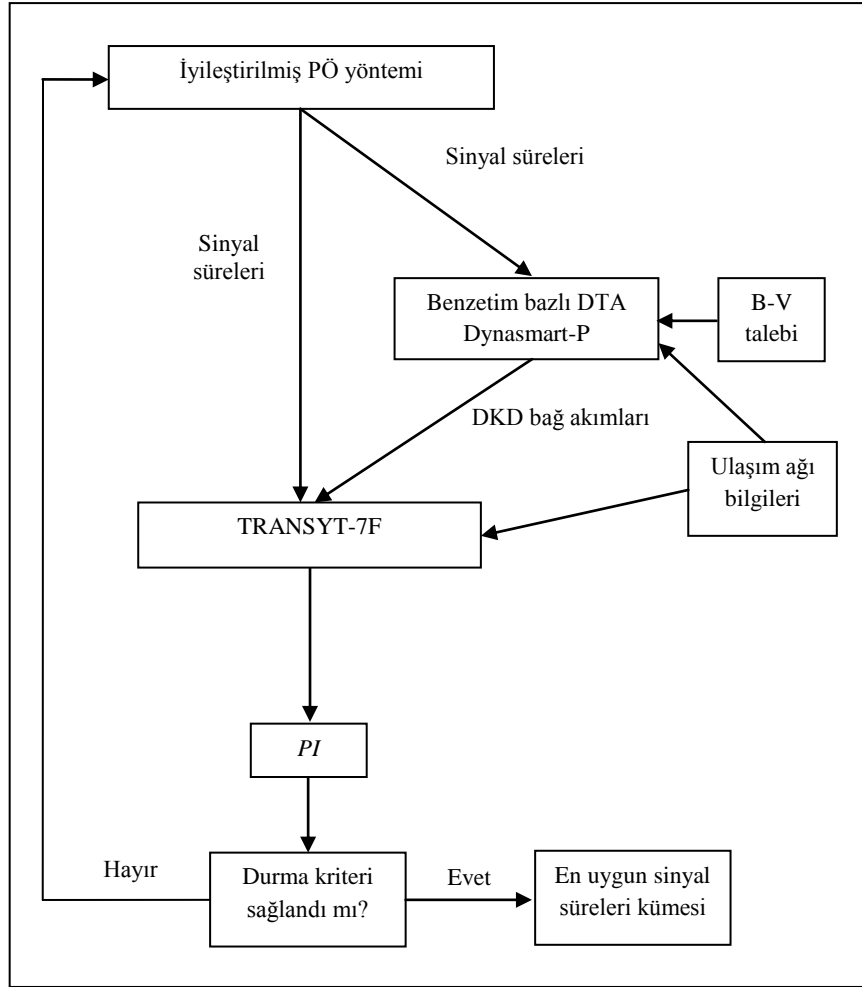
$$c_{\min} \leq c \leq c_{\max} \quad \text{devre süresi kısıtı}$$

$$0 \leq \theta \leq c \quad \text{offset süresi kısıtı}$$

$$\varphi_{\min} \leq \varphi \leq c \quad \text{yeşil süre kısıtı}$$

$$\sum_{i=1}^z (\varphi + I)_i = c$$

Burada, ψ sinyal kontrol parametrelerini (c devre süresi, θ her kavşak için ofset süresi, φ her fazın yeşil süresi), Ω_0 sinyal kontrol parametreleri için çözüm uzayını, $\mathbf{q}^*(\psi)$ DKD bağ trafik hacimlerini, I yeşillerarası süreyi (sn), z yol ağındaki her bir sinyalize kavşağa ait faz sayısını göstermektedir. DKD bağ trafik hacimleri PÖTRAD modelinin alt seviyesinde Dynasmart-P yazılımı ile elde edilmektedir. DKD altındaki bağ hacim değerleri ile sinyal kontrol parametreleri modelin üst seviyesinde TRANSYT-7F yazılımına girdi olmakta ve ağın PI değeri hesaplanmaktadır. Oluşturulan modelde İPÖ yöntemi ile bu işlemler her öğrenme evresinde tekrar edilerek PI değerini enküçükleyen en uygun sinyal kontrol parametrelerinin bulunması amaçlanmıştır.



Şekil 4.7 PÖTRAD modeli genel çerçevesi

4.3.2. PÖTRAD Modeli Çözüm Algoritması

DKD altında UAT problemi çözmek için oluşturulan PÖTRAD modelinde uygulanması gereken adımlar aşağıda verilmiştir:

- **Adım 0:** Başlangıç $t=0$. Ulaşım ağı bilgilerini, B-V talebini, kullanıcıya özel PÖ parametrelerini; öğrenme oranı (α), azaltma faktörü (γ), en büyük öğrenme evresi sayısı (t_{maks}), çevre boyutunu (m), her değişken için çözüm uzayı parametresini (β) gir. Değişken sayısını (n) ve kısıtları sinyal parametrelerine (ψ) bağlı olarak tanımla.

- **Adım 1:** $t=1$ ise ($m \times n$) boyutundaki orijinal çevre içinde başlangıç sinyal parametrelerini, $\psi(c, \theta, \phi)$, $Q(s, a)$ olarak üret.

- **Adım 2:** Adım 1'de orijinal çevre içinde üretilen başlangıç sinyal parametrelerinden yeşil süreleri Denklem (3.9)'a göre hesapla.

- **Adım 3:** Sinyal parametrelerinden devre süresi ile yeşil süreleri Dynasmart-P girdi dosyasına yaz.

- **Adım 4:** Orijinal çevre içindeki her boyut için Dynasmart-P yazılımını çalıştır.

- **Adım 5:** Orijinal çevre içindeki her boyut için Dynasmart-P çıktı dosyasından DKD bağ trafik hacimlerini belirle.

- **Adım 6:** Sinyal parametrelerini ve Dynasmart-P yazılımından elde edilen DKD bağ trafik hacimlerini TRANSYT-7F trafik modeline gir.

- **Adım 7:** Orijinal çevre içindeki her boyut için TRANSYT-7F trafik modelini çalıştır.

- **Adım 8:** Orijinal çevre içindeki her boyut için TRANSYT-7F trafik modelinden ulaşım ağı PI değerlerini al.

- **Adım 9:** $t > 1$ ise ($m \times n$) boyutunda yeni bir orijinal çevre içinde yeni başlangıç sinyal parametreleri üret, bir önceki öğrenme evresinde ($m \times n$) boyutundaki orijinal çevre içinde en iyi PI değerini sağlayan en iyi sinyal parametrelerini ($m+1$)'inci satırda sakla, orijinal çevre boyutunda ($m+2$)'nci satırdan ($2m+1$)'nci satıra kadar önceki öğrenme evresinde elde edilen en iyi sinyal parametrelerini ve β

vektörünü kullanarak Denklem (3.3) yardımıyla alt çevreyi oluştur. Aksi takdirde, Adım 17'ye git.

- **Adım 10:** Adım 9'da orijinal ve alt çevre içinde üretilen sinyal parametrelerinden yeşil süreleri Denklem (3.9)'u kullanarak hesapla.

- **Adım 11:** Sinyal parametrelerinden devre süresi ile yeşil süreleri Dynasmart-P girdi dosyasına gir.

- **Adım 12:** Orijinal ve alt çevre içindeki her boyut için Dynasmart-P yazılımını çalıştır.

- **Adım 13:** Orijinal ve alt çevre içindeki her boyut için Dynasmart-P çıktı dosyasından DKD bağ trafik hacimlerini belirle.

- **Adım 14:** Sinyal parametrelerini ve Dynasmart-P yazılımından elde edilen DKD bağ trafik hacimlerini TRANSYT-7F trafik modeline gir.

- **Adım 15:** Orijinal ve alt çevre içindeki her boyut için TRANSYT-7F trafik modelini çalıştır.

- **Adım 16:** Orijinal ve alt çevre içindeki her boyut için TRANSYT-7F trafik modelinden ulaşım ağı PI değerlerini al.

- **Adım 17:** Elde edilen ulaşım ağı PI değerlerine göre tüm çevreyi iyiden kötüye doğru sırala.

- **Adım 18:** Adım 17'deki sıralama sonunda en iyi ulaşım ağı PI değerini veren sinyal parametrelerini belirle.

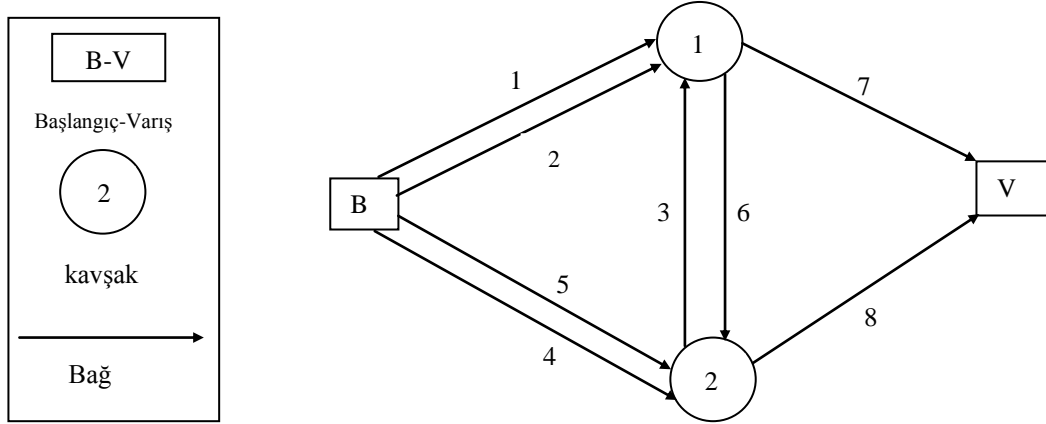
- **Adım 19:** Tüm çevre içinde Denklem (3.4)'ü kullanarak ödül fonksiyonu, $r_i(s, a)$, değerlerini belirle.

- **Adım 20:** Sinyal parametreleri olarak $Q_i(s, a)$ değerlerini Denklem (3.2)'yi kullanarak güncelle.

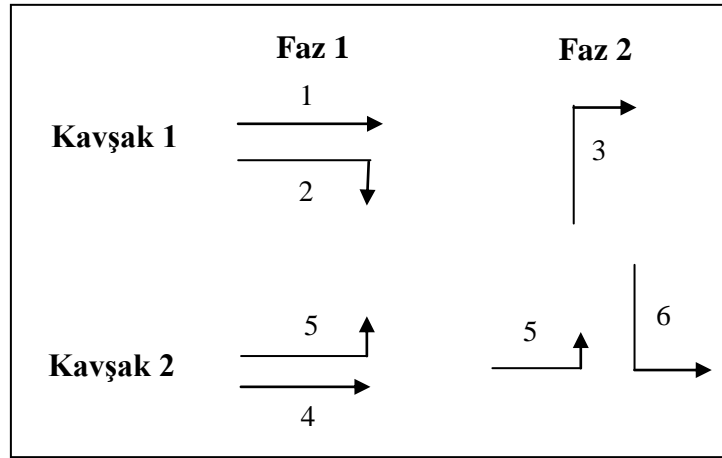
- **Adım 21:** Durma kriterini kontrol et. Eğer sağlanmamışsa Adım 9'a git ve $t = t + 1$ yap. Aksi takdirde en iyi ψ sinyal parametreleri ile Dynasmart-P yazılımı tekrar çalıştırılarak DKD bağ trafik hacimleri bulunur.

4.4. PÖTRAD Modeli Uygulaması

Geliştirilen PÖTRAD modeli örnek ulaşım ağına uygulanmıştır. Örnek ulaşım ağında 1 adet B-V çifti, 2 adet kavşak ve 8 adet bağ bulunmaktadır. Örnek ulaşım ağı ve kavşaklara ait sinyal faz düzenleri sırasıyla Şekil 4.8a ve 4.8b'de görülebilmektedir. Bu örnek ulaşım ağına ait diğer veriler Tablo 4.1'de görülebilmektedir (Ceylan, 2002).



Şekil 4.8a Örnek ulaşım ağı (Ceylan, 2002)



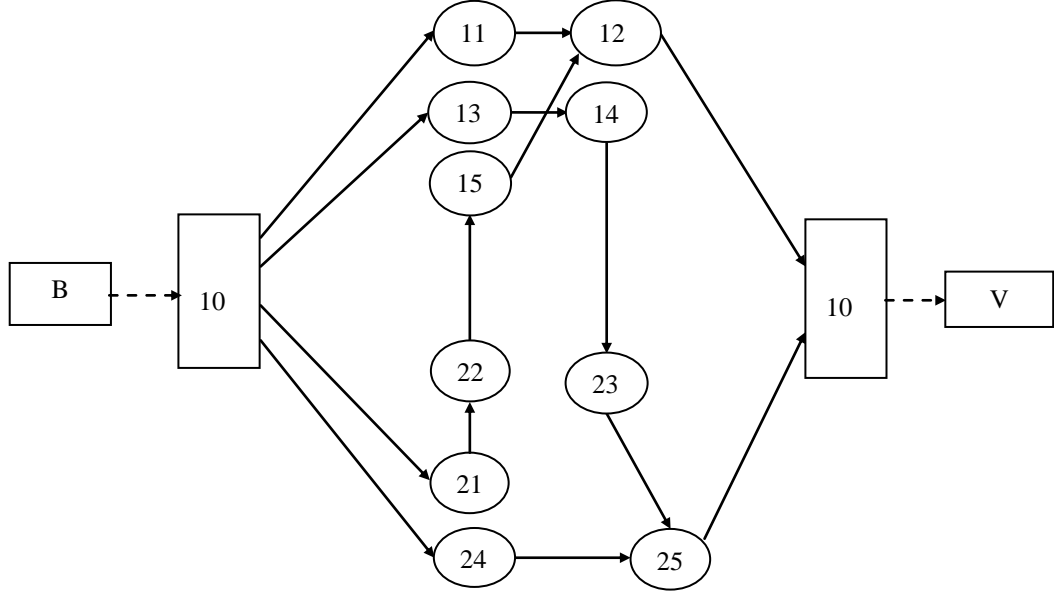
Şekil 4.8b Örnek ulaşım ağı kavşak fazları (Ceylan, 2002)

Tablo 4.1. Örnek ulaşım ağı verileri

Bağ	s	t_a^0	Bağ	s	t_a^0
1	1800	20	4	1800	20
2	1800	20	5	1800	20
3	1800	20	6	1800	20

Burada, s doygun akım değeri (ta/sa), t_a^0 serbest akım seyahat süresi (sn)., Bağ 7 ve 8 için serbest akım seyahat süresi 10 sn., B-V talebi 1500 ta/sa'dır.

Örnek ulaşım ağının trafik atama amacıyla yeniden düzenlenmiş hali Şekil 4.9'da görülebilmektedir. Yeniden düzenlenen örnek ulaşım ağı toplam 16 bağdan oluşmaktadır. Bu bağların 8 tanesinin (B-10, 11-12, 13-14, 15-12, 21-22, 24-25, 23-25, 10-V) seyahat süresi 1 sn'dir.



Şekil 4.9 Örnek ulaşım ağının trafik atama için düzenlenmiş hali (Ceylan, 2002)

Örnek ulaşım ağının çözümünde kullanılacak sinyal parametrelerinin alt ve üst limit değerleri aşağıda verilmiştir.

$$36 \leq c \leq 90$$

$$0 \leq \theta \leq c$$

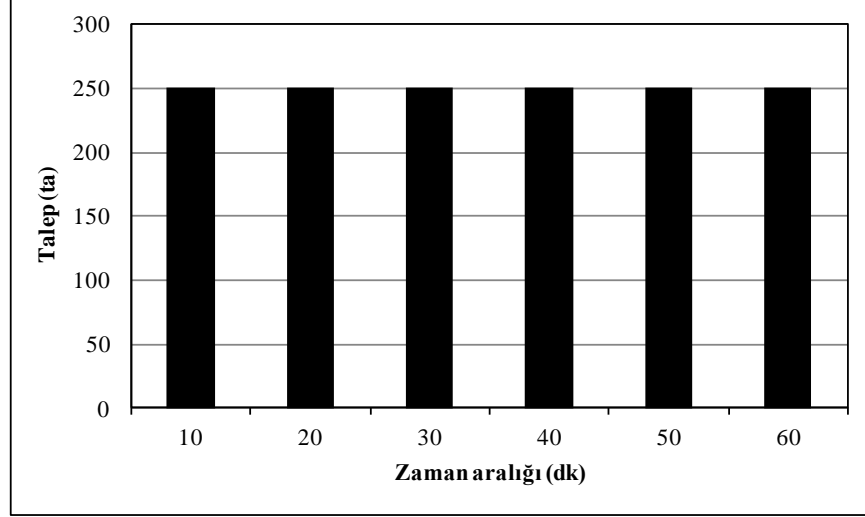
$$7 \leq \varphi \leq c$$

$$I_{1-2} = I_{2-1} = 5\text{sn}$$

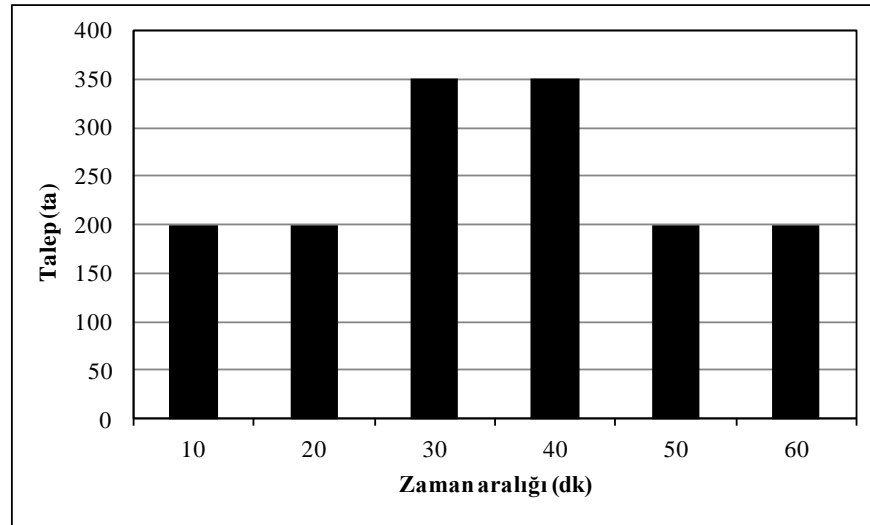
PÖTRAD modelinin örnek ulaşım ağına uygulanması aşamasında İPÖ algoritması parametreleri; $\alpha = 0.8$, $\gamma = 0.2$, $m = 20$ ve $t_{\text{maks}} = 300$ olarak seçilmiştir. Çözüm prosedürü durma kriteri sağlanıncaya kadar devam ettirilmiştir. t 'inci öğrenme evresinde ulaşım ağının en iyi PI değeri ile ortalama PI değeri arasındaki fark % 1'den az olduğunda PÖTRAD modeli sonlanmaktadır.

Örnek ulaşım ağı üzerinde iki sayısal uygulama yapılmıştır. Uygulamalardan birincisi B-V talebinin analiz periyodu içinde eşit olarak ağa yüklendiği durumu,

ikincisi ise B-V talebinin analiz periyodu içerisinde farklı ve dağınık olarak ağı yüklediği durumu göstermektedir. Birinci uygulamada kullanılacak talep yükleme profili Şekil 4.10'da, ikinci uygulamada kullanılacak talep yükleme profili Şekil 4.11'de gösterilmektedir.



Şekil 4.10 Örnek ulaşım ağı için ulaşım talebinin ağı yüklemesi profili 1

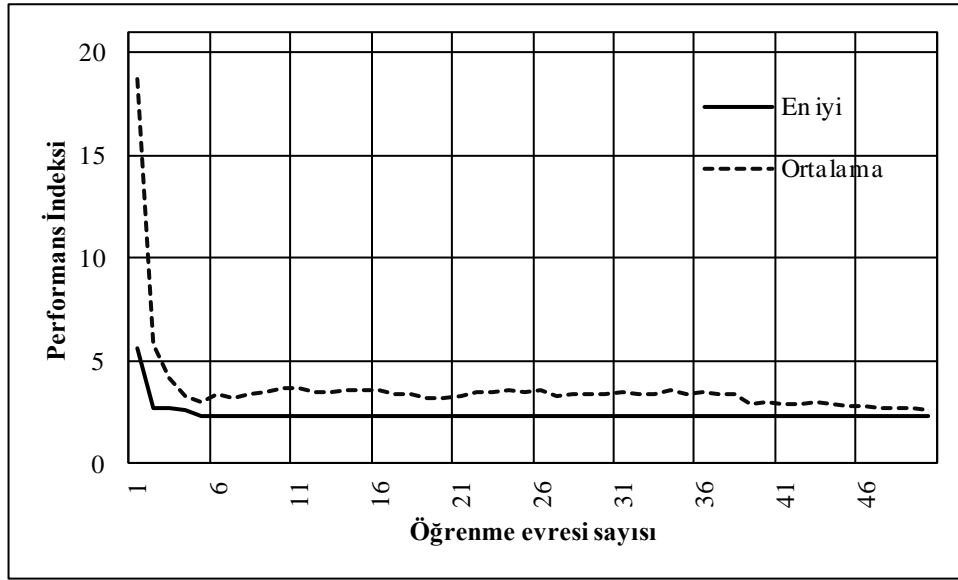


Şekil 4.11 Örnek ulaşım ağı için ulaşım talebinin ağı yüklemesi profili 2

Örnek ulaşım ağı üzerinde yapılan iki uygulama arasındaki fark, 1500 ta/sa'lik B-V talebinin analiz süresi içinde ağı yüklenme durumu ile ilgilidir. Birinci uygulamada 1500 ta/sa'lik B-V talebi analiz süresi içinde eşit ve düzgün olarak ağı yüklenmiştir. Bir başka deyişle, her 10 dakikalık zaman diliminde ağı 250 taşıt yüklenmiştir. İkinci uygulamada ise, 1500 ta/sa'lik B-V talebinin analiz süresi içinde ağı

yüklenme yapısı eşit ve düzgün dağılımlı değildir. İlk yirmi dakikanın her on dakikasında 200 taşıt, ikinci yirmi dakikanın her on dakikasında 350 taşıt ve son yirmi dakikanın her on dakikasında 200 taşıt ulaşım ağına yüklenmiştir. B-V talebinin farklı profillerle yüklenmesinin nedeni zamana bağlı olarak değişen talep altında ulaşım ağının nasıl çalıştığı sorusuna cevap bulunması içindir. Bir başka deyişle, DTA'nın ana özelliklerinden biri zamana bağlı talep kavramının UAT probleminin çözümü üzerindeki etkisini ortaya çıkarmaktır.

B-V talebinin, yükleme profili 1'e göre örnek ulaşım ağına yüklenmesi ve PÖTRAD modelinin çalıştırılması sonucu amaç fonksiyonu olan ağ *PI* değerinin yakınsama grafiği Şekil 4.12'de görülebilmektedir.



Şekil 4.12 Örnek ulaşım ağına PÖTRAD modelinin yakınsaması
(yükleme profili 1)

Şekil 4.12'den görülebildiği gibi PÖTRAD modeli en iyi *PI* değerine 5'inci öğrenme evresinde ulaşmıştır. 50'inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. Örnek ulaşım ağına yüklenme profili 1'e göre en iyi *PI* değeri 2.33 olarak bulunmuştur. En iyi *PI* değerini sağlayan sinyal parametreleri ve dinamik denge bağ akımları sırasıyla Tablo 4.2 ve 4.3'de verilmektedir.

Tablo 4.2 Örnek ulaşım ağında en iyi sinyal parametreleri ve PI değeri
(yükleme profili 1)

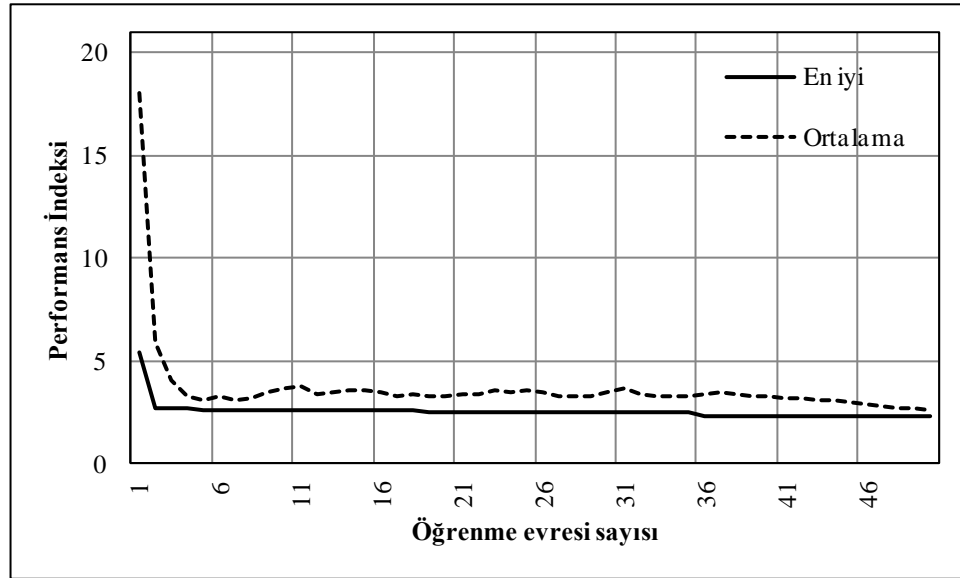
PI	Devre süresi c (sn)	Kavşak numarası i	Faz yeşil süreleri (sn)		Ofset süresi (sn) θ_i
			Faz 1 $\phi_{i,1}$	Faz 2 $\phi_{i,2}$	
PÖTRAD 2.33	90	1	70	10	26
		2	70	10	0

Tablo 4.3 Örnek ulaşım ağında yükleme profili 1'e göre dinamik denge bağ akımları (ta/sa)

q_1	q_2	q_3	q_4	q_5	q_6
365	382	54	10	54	367

Tablo 4.2 ve 4.3'den görülebildiği gibi, ortak devre süresi 90 sn olarak bulunurken her iki kavşaktaki her iki faz için yeşil süreler 70 ve 10 sn bulunmuştur. Ofset süresi birinci kavşak için 26 sn, ikinci kavşak için 0 sn'dir.

B-V talebinin, yükleme profili 2'ye göre örnek ulaşım ağına yüklenmesi ve PÖTRAD modelinin çalıştırılması sonucu amaç fonksiyonu olan ağ PI değerinin yakınsama grafiği Şekil 4.13'de görülebilmektedir.



Şekil 4.13 Örnek ulaşım ağında PÖTRAD modelinin yakınsaması
(yükleme profili 2)

Şekil 4.13'den görülebildiği gibi PÖTRAD modeli en iyi PI değerine 36'ncı öğrenme evresinde ulaşmıştır. 50'inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. Örnek ulaşım ağında yükleme profili 2'ye göre en iyi PI değeri 2.32 olarak bulunmuştur. En iyi PI değerini sağlayan sinyal parametreleri ve dinamik denge bağ akımları sırasıyla Tablo 4.4 ve 4.5'de verilmektedir.

Tablo 4.4 Örnek ulaşım ağında en iyi sinyal parametreleri ve PI değeri (yükleme profili 2)

PI	Devre süresi c (sn)	Kavşak numarası i	Faz yeşil süreleri (sn)		Ofset süresi (sn) θ_i
			Faz 1 $\phi_{i,1}$	Faz 2 $\phi_{i,2}$	
PÖTRAD 2.32	90	1	70	10	0
		2	69	11	49

Tablo 4.5 Örnek ulaşım ağında yükleme profili 2'ye göre dinamik denge bağ akımları (ta/sa)

q_1	q_2	q_3	q_4	q_5	q_6
380	401	47	10	47	373

Tablo 4.4 ve 4.5'den görülebildiği gibi, ortak devre süresi 90 sn olarak bulunurken yeşil süreler birinci kavşağın birinci fazı için 70 sn, ikinci faz için 10 sn, ikinci kavşağın birinci fazı için 69 sn, ikinci fazı için 11 sn bulunmuştur. Ofset süresi birinci kavşak için 0 sn, ikinci kavşak için 49 sn'dir.

1500 ta/sa'lık B-V talebinin iki farklı yükleme profili ile örnek ulaşım ağına yüklenmesi sonunda geliştirilen PÖTRAD modeli amaç fonksiyonu olan ağ PI değerini en küçükleyebilmektedir. Ayrıca model en iyi PI ile ortalama PI değerinin yakınsaması açısından iyi performans göstermiştir. Her iki yükleme profiline göre talebin ağ üzerine yüklenmesi sonunda elde edilen sinyal parametreleri birbirine benzerlik göstermektedir. Yalnızca ofset süreleri ve ağ PI değerleri farklılık göstermektedir.

Tablo 4.6'da B-V talebinin iki farklı yükleme profiline göre örnek ulaşım ağı üzerine yüklenmesi sonunda elde edilen ağ ile ilgili istatistikler (toplam seyahat süresi, ortalama seyahat süresi, toplam seyahat mesafesi, ortalama seyahat mesafesi ve analiz periyodu sonunda ağ içinde kalan taşıt sayısı) verilmektedir. Tablo 4.6'dan

görülebildiği gibi, B-V talebi yükleme profili 1'e göre ulaşım ağına yüklendiğinde tüm taşıtlar için toplam seyahat süresi 56.77 sa, yükleme profili 2'ye göre yüklendiğinde tüm taşıtlar için toplam seyahat süresi 62.85 sa olmaktadır.

Tablo 4.6 Örnek ulaşım ağına PÖTRAD modelinin uygulanması sonucu elde edilen istatistikler

Talep yükleme profili	Toplam seyahat süresi (sa)	Ortalama seyahat süresi (dk)	Toplam seyahat mesafesi (ta-km)	Ortalama seyahat mesafesi (ta-km)	Ağda kalan taşıt sayısı
1	56.77	2.29	378.32	0.25	337
2	62.85	2.52	383.33	0.26	344

Ayrıca, B-V talebi örnek ulaşım ağına yükleme profili 1'e göre yüklendiğinde ortalama seyahat süresi 2.29 dk iken yükleme profili 2'ye göre yüklendiğinde ortalama seyahat süresi 2.52 dk olmaktadır. Yükleme profili 1'e taşıtların kattettikleri toplam mesafe 378.32 km, ortalama mesafe 0.25 km iken yükleme profili 2'ye göre taşıtların kattettikleri toplam mesafe 383.33 km, ortalama mesafe 0.26 km olmaktadır.

Örnek ulaşım ağı üzerinde yapılan her iki uygulamada B-V talebi aynı olmasına rağmen, ağ üzerinde elde edilen seyahat süresi, katedilen mesafe gibi istatistikler açısından sonuçlar farklıdır. Bu farklılığın nedeni, B-V talebinin örnek ulaşım ağına yükleme profillerinden kaynaklanmaktadır. Özellikle yükleme profili 2'ye göre yüklenen 1500 ta/sa'lik B-V talebinin analiz süresi içinde zamanla değişiklik göstermesi ve buna bağlı olarak da ulaşım ağı üzerindeki rotaların ve rotaları meydana getiren bağların seyahat sürelerinin ve hacimlerinin zamanla değişmesidir. Ayrıca, Tablo 4.16'dan görülebildiği gibi analiz süresi sonunda B-V talebini oluşturan taşıtların tümünün seyahatlerinin varış noktalarına ulaşamadıkları görülmektedir. Yükleme profili 1'de 337 taşıt, yükleme profili 2'de 344 taşıt seyahatlerinin varış noktalarına ulaşamayıp ağ içinde kalmıştır. Bunun nedeni, benzetim bazlı DTA modellerinde kullanılan zamana bağlı en kısa rota algoritmasıdır. Statik TA modellerinde analiz süresi sonunda içinde tüm taşıtların varış noktalarına eriştikleri kabulü yapılırken DTA modellerinde analiz süresi sonunda talebi oluşturan tüm taşıtların hedeflerine varamayabilecekleri kabulü

yapılmaktadır. Bir başka deyişle, zamana bağlı en kısa rota algoritmasını kullanan benzetim bazlı DTA modellerinde analiz süresi sonunda ağ içinde kalan taşıtlar varış noktalarına ulaşamayıp seyahatlerini tamamlayamayabilirler.

4.5. Sonuçlar

Bu bölümde benzetim bazlı DTA modeli olan Dynasmart-P hakkında detaylı bilgi verilerek yazılımın altında çalışan çözüm algoritmalarından bahsedilmiştir. Ayrıca, benzetim bazlı DTA modellerinde kullanılan zamana bağlı en kısa rota algoritması, karşı karşıya kalınan seyahat süresi ve anlık seyahat süresi hakkında detaylı bilgi verilmiştir.

DKD altında UAT problemini çözmek için iki seviyeli programlama tekniğinden faydalanılarak PÖTRAD modeli oluşturulmuştur. Oluşturulan PÖTRAD modelinde alt seviyeyi temsil eden TA probleminin çözümünde benzetim bazlı DTA modeli olan Dynasmart-P yazılımı kullanılmış üst seviye olan sinyal kontrol probleminin çözümünde ise TRANSYT-7F yazılımının trafik modeli kullanılmıştır. İPÖ yöntemi ise her öğrenme evresinde sinyal parametrelerinin eniyilenmesi aşamasında kullanılmıştır.

DKD altında UAT probleminin çözümü için oluşturulan PÖTRAD modelinin algoritma adımları ve çalışma şekli detaylı biçimde açıklanmıştır. Oluşturulan PÖTRAD modeli iki kavşaklı örnek ulaşım ağı üzerinde test edilmiştir. Bu amaçla, 1500 ta/sa'lık B-V talebinin örnek ulaşım ağına yüklenmesi ile ilgili iki farklı uygulama yapılmıştır. Uygulamaların farklılıkları, zamana bağlı talep durumunu yansıtması açısından sadece B-V talebinin ulaşım ağına yükleme profillerinden kaynaklanmaktadır. PÖTRAD modeli her uygulama için çalıştırılmış ve elde edilen uygulama sonuçları değerlendirilmiştir.

Örnek ulaşım ağında en iyi PI değeri yükleme profili 1'e göre 2.33, yükleme profili 2'ye göre 2.32 olarak bulunmuştur. DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modeli, amaç fonksiyonu olan ağ PI değerini enküçükleyebilmiş ve model en iyi PI ile ortalama PI değerinin yakınsaması açısından iyi performans göstermiştir. Her iki yükleme profiline göre B-V talebinin ağ üzerine yüklenmesi sonunda elde edilen sinyal parametrelerinin birbirine benzerlik gösterdiği söylenebilir.

İki farklı yükleme profiline göre örnek ulaşım ağından elde edilen istatistikler verilmiştir. Tüm taşıtlar için toplam seyahat süresi, yükleme profili 1'e göre 56.77 sa, yükleme profili 2'ye göre 62.85 sa olmuştur. Ortalama seyahat süresi yükleme profili 1'e göre 2.29 dk iken yükleme profili 2'ye göre 2.52 dk bulunmuştur. Yükleme profili 1'e göre taşıtların katettikleri toplam mesafe 378.32 km, ortalama mesafe 0.25 km iken yükleme profili 2'ye göre taşıtların kattettikleri toplam mesafe 383.33 km, ortalama mesafe 0.26 km olarak bulunmuştur. Her iki uygulamada B-V talebi aynı olmasına rağmen, ağ üzerinde elde edilen seyahat süresi, katedilen mesafe gibi istatistikler açısından farklı sonuçlar elde edilmiştir. Bu farklılık, B-V talebinin örnek ulaşım ağına yükleme profillerinden kaynaklanmıştır.

Ayrıca, benzetim bazlı DTA modellerinde kullanılan zamana bağlı en kısa rota algortimasının sonucu olarak, analiz süresi sonunda B-V talebini oluşturan taşıtların tümünün seyahatlerinin varış noktalarına ulaşamadıkları görülmüştür. Yükleme profili 1'de 337 taşıt, yükleme profili 2'de 344 taşıt seyahatlerinin varış noktalarına ulaşamayıp ağ içinde kalmıştır.

Bir sonraki bölümde, PÖTRAD modelinin orta ölçekli bir ulaşım ağı üzerindeki sayısal uygulamaları yer almaktadır. Ayrıca, bu ulaşım ağı üzerinde taşıtların oluşturacağı egzoz emisyon değerlerinin hesapları bulunmaktadır.

5. SAYISAL UYGULAMA

5.1. Giriş

Bu bölümde, PÖTRAD modeli ile altı kavşaktan meydana gelen orta ölçekli ulaşım ağı üzerinde sayısal uygulamalar yapılmıştır. İlk sayısal uygulamada, 5000 ta/sa'lik ulaşım talebi altında ulaşım ağının performansını ve ağ üzerindeki trafiğin yapısını görebilmek amacıyla PÖTRAD modelinde Dynasmart-P modelinin bir adımlı benzetimi kullanılmıştır. Daha sonra, DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin sayısal uygulaması için dört tane senaryo oluşturulmuştur. Oluşturulan senaryolar B-V talebinin altı kavşaklı ulaşım ağı üzerine yüklenmesi ile ilgilidir. Oluşturulan senaryolarla yapılan sayısal uygulamalarda PÖTRAD modelinin altında TA probleminin çözümünde Dynasmart-P yazılımının iteratif benzetim bazlı atama prosedürü kullanılarak DKD altında UAT probleminin çözümü yapılmış ve elde edilen sonuçlar karşılaştırmalı olarak verilmiştir. Ayrıca, her senaryonun ulaşım ağı üzerinde uygulanması sonucu ağ üzerindeki taşıtların meydana getirdiği egzoz emisyon değerleri hesaplanmıştır.

5.2. PÖTRAD Modeli Sayısal Uygulaması

PÖTRAD modelinin sayısal uygulaması Allsop ve Charlesworth (1977) ve Ceylan (2002) tarafından kullanılan ulaşım ağı üzerinde yapılmıştır. Ulaşım ağı 20 adet B-V çifti, 6 adet sinyalize kavşak ve 23 adet bağdan oluşmaktadır. Ulaşım ağının yapısı ve faz planları üçüncü bölüm içinde verilmiştir (Şekil 3.13 ve 3.14). Şekil 5.1'de ulaşım ağının TA problemi için düzenlenmiş hali verilmektedir. Bir başka deyişle, Şekil 5.1'de ağda tüm düğüm noktalarını birleştiren bağlar gösterilmektedir.

Şekil 5.1’de düğüm noktalarını birleştiren bağlar xy olarak kodlanmıştır. Burada, x akım yukarı düğüm noktasını belirtirken y de akım aşağı düğüm noktasını belirtmektedir. Şekil 3.13’deki ulaşım ağının yapısı ile Şekil 5.1’deki ulaşım ağının düzenlenmiş yapısı arasındaki ilişki Tablo 5.1’den daha net olarak anlaşılmaktadır.

Tablo 5.1 Şekil 3.13 ile Şekil 5.1’de gösterilen bağların ilişkisi (Ceylan, 2002)

Şekil 3.14’deki bağ numarası	Şekil 5.1’deki bağ numarası	Şekil 5.1’de 1 sn seyahat süreli bağlar
1, B_A, B_D ^(*)		
2	1012	
3	1026	2621, 2622
4	2236	3632, 36321, 3634
5	3249	4948
6	32147	4742
7	1273	7374, 7375, 73751
8	7562	6264
9	75167	6766
10	6445	4546, 4542, 4548
11	40411	41146
12	4041	4142, 4144
13	5043	4344, 4346, 4348
14	4633	3334, 3335
15	3523	2325, 2321
16	2511	1112, 1113
17	4465	6566, 6561
18	6176	7672, 7674
19	7213	1310
20	3031	3132, 3135, 3134, 31321
21	3463	6364, 6361, 6366
22	7071	7172, 7174, 7175, 71751
23	7424	2422, 2421, 2425

^(*)1 numaralı bağ tüm devre süresi boyunca yeşil almaktadır böylelikle bu bağ üzerinde herhangi bir gecikme olmamaktadır. B_A ve B_D bağları kukla bağlardır ve 1 sn seyahat süresine sahiptirler. İkinci sütundaki bağlar birinci sütundaki bağların eşdeğeridir. Üçüncü sütundaki bağlar trafik atama amacından dolayı 1 sn seyahat süresine sahiptirler. Diğer bağlar ağdan çıkan bağlar olup 1 sn seyahat süresine sahiptirler (2120, 4840, 4250, 6660 gibi)

Her bir B-V çifti arasındaki seyahat talepleri Tablo 5.2’de verilmektedir. Tablo 5.3’de her bir sinyalize kavşağa giren bağ numaraları, bağlara ait serbest akım seyahat süreleri ve doymuş akım değerleri verilmektedir. Tablo 5.2’de görüldüğü gibi en çok seyahat üretilen zon C ile belirtilen zondur. Buradan diğer zonlara üretilen toplam seyahat talebi 1290’dır. En çok seyahat çeken ise A zonudur ve yine aynı şekilde toplam çekilen seyahat talebi 1290 olarak verilmiştir. Ulaşım ağında toplam çekilen ve üretilen seyahat talebi 5000 olarak verilmiştir.

Tablo 5.2 Allsop ve Charlesworth ulaşım ağı B-V talepleri (Ceylan, 2002)

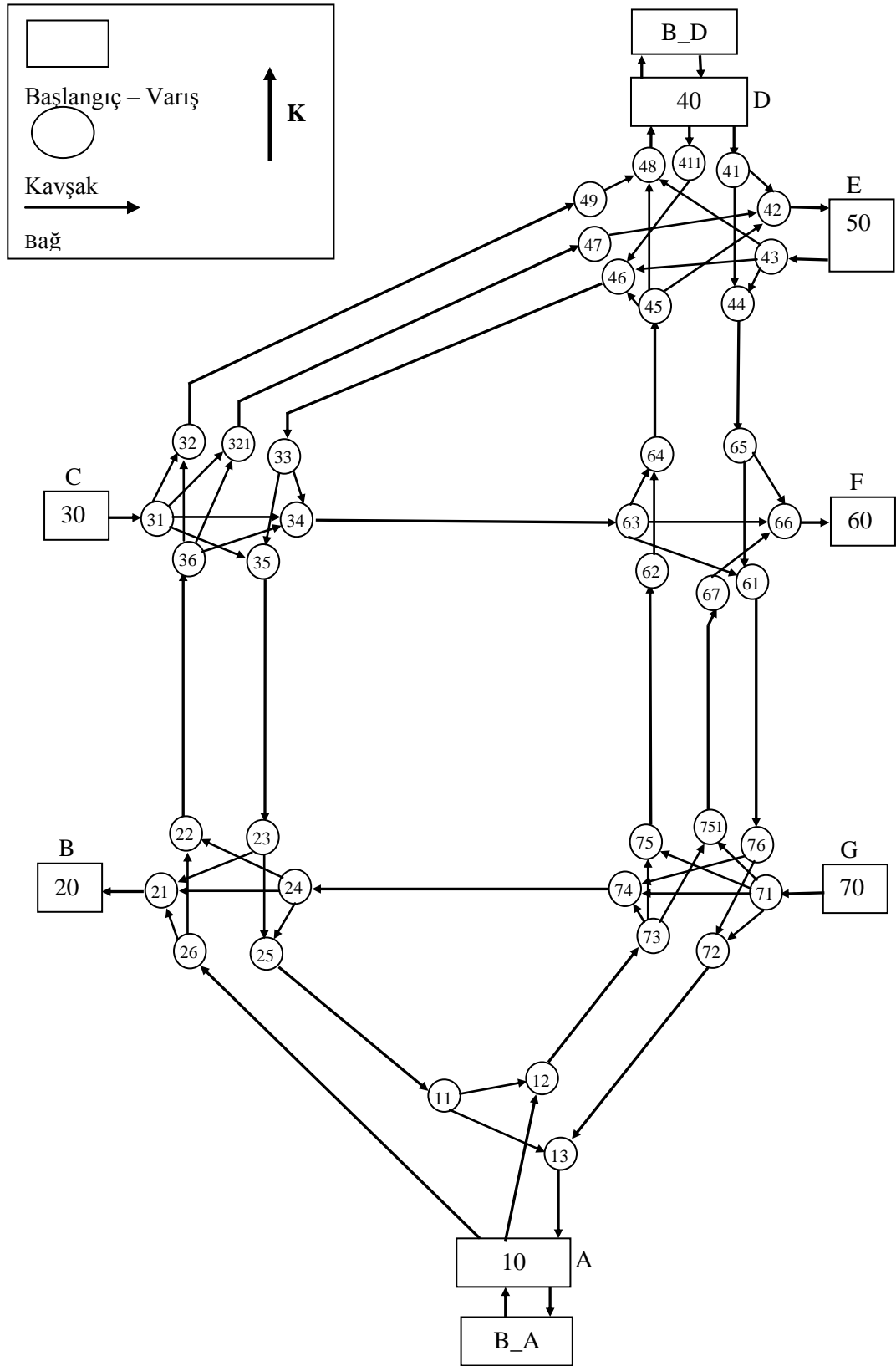
Başlangıç-Variş	A	B	D	E	F	Toplam
A	--	250	700	30	200	1180
C	40	20	200	130	900	1290
D	400	250	--	50*	100	800
E	300	130	30*	--	20	480
G	550	450	170	60	20	1250
Toplam	1290	1100	1100	270	1240	5000

* D ve E olarak belirtilen B-V çiftleri arasındaki ulaşım talebi sayısal uygulama içine dahil edilmemiştir.

Tablo 5.3 Allsop ve Charlesworth ulaşım ağı karakteristik özellikleri (Ceylan, 2002)

Kavşak	Bağ	t_a^0	s	Kavşak	Bağ	t_a^0	s
1	1	1	2000	4	5	20	1800
	2	1	1600		6	20	1850
	16	10	2900		10	10	2200
	19	10	1500		11	1	2000
					12	1	1800
					13	1	2200
2	3	10	3200	5	8	15	1850
	15	15	2600		9	15	1700
	23	15	3200		17	10	1700
					21	15	3200
3	4	15	3200	6	7	10	1800
	14	20	3200		18	15	1700
	20	1	2800		22	1	3600

Burada t_a^0 serbest akım seyahat süresi (sn), s ise doymuş akım (ta/sa) değeridir.



Şekil 5.1 Allsop ve Charlesworth ulaşım ağının TA için düzenlenmiş hali
(Ceylan, 2002)

PÖTRAD modelinde kullanılan İPÖ yöntemi global en uygun çözümü bulmada oldukça etkin bir algoritmadır. Geliştirilen algoritma ÇU içerisinde hem orijinal çevre hem de alt çevre ile en uygun çözümü aramaktadır. Alt çevre oluşturulması ile bir önceki öğrenme evresinde elde edilen en uygun çözüm kullanılarak çözüm kümesi daraltılmakta ve böylelikle algoritmanın lokal çözümlere takılması önlenmektedir. PÖTRAD modelinin sayısal uygulamasında kullanılacak sinyal parametrelerinin alt ve üst limit değerleri aşağıdaki gibidir.

$$36 \leq c \leq 140$$

$$0 \leq \theta \leq c$$

$$7 \leq \varphi \leq c$$

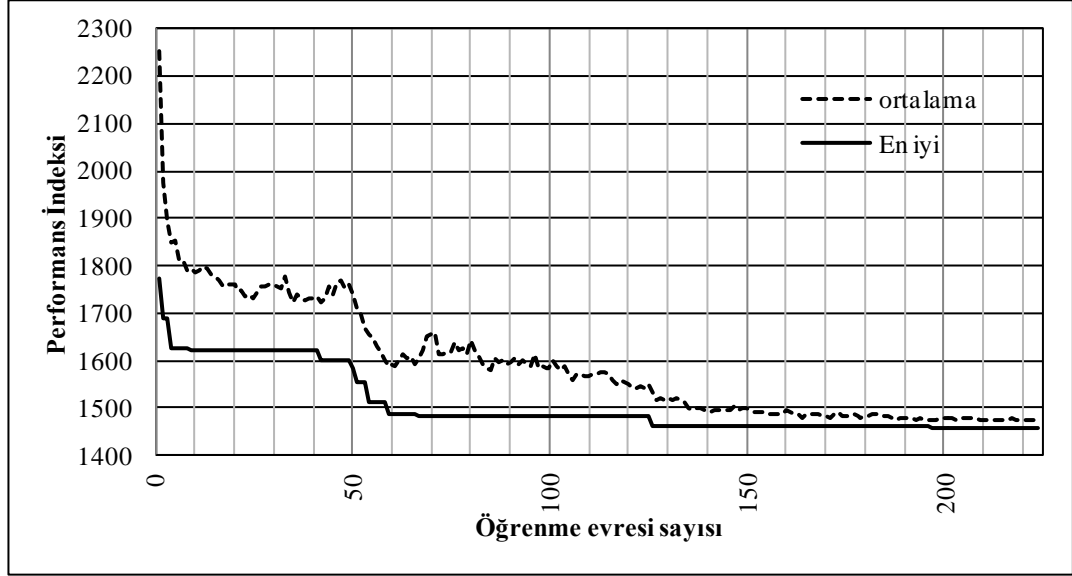
$$I_{1-2} = I_{2-1} = 5sn$$

Şekil 3.15'den görülebildiği gibi, ulaşım ağındaki 6 kavşakta toplam 14 tane faz bulunmaktadır. Dolayısıyla, bu ulaşım ağı için 14 tane yeşil süre, 6 tane ofset süresi ve 1 tane de ortak devre süresi olmak üzere toplam 21 tane karar değişkeni bulunmaktadır. İPÖ algoritmasında, sinyal parametrelerinden yeşil süreler, Ceylan (2002) tarafından geliştirilen ve üçüncü bölümde verilen Denklem (3.9) yardımıyla elde edilir.

5000 ta/sa'lik ulaşım talebi altında ulaşım ağının performansını ve ağ üzerindeki trafiğin yapısını görebilmek amacıyla PÖTRAD modelinin ağ üzerindeki ilk sayısal uygulamasında Dynasmart-P modelinin bir adımlı benzetimi kullanılmıştır. Bir başka deyişle, ilk sayısal uygulamada PÖTRAD modelinin altında TA probleminin çözümünde Dynasmart-P yazılımının bir adımlı benzetimi kullanılarak 5000 ta/sa'lik talebin ulaşım ağına yüklenmesi ve tüm taşıtların varış noktalarına ulaşmaya kadar hareket etmesi sağlanmıştır.

PÖTRAD modelinin bir adımlı benzetim ile ulaşım ağı üzerinde uygulanması sonucunda amaç fonksiyonu olan ağ PI değerinin yakınsama grafiği Şekil 5.2'de verilmektedir. Şekil 5.2'den görülebildiği gibi PÖTRAD modeli 199'uncu öğrenme evresinde en iyi PI değerine 1457'ye ulaşmıştır. 1'inci öğrenme evresinde elde edilen PI değeri 1773'dür. PÖTRAD modeli ÇU içinde her öğrenme evresi boyunca PI değerini en küçükleyen en uygun sinyal parametrelerini aramaya devam etmiş ve en küçük PI değerini 1457 olarak bulmuştur. PÖTRAD modeli ulaşım ağının

çözümünde oldukça kararlı bir yakınsama eğilimi göstermiştir. 225'inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. Bir adımlı benzetim sonunda diğer bir deyişle talebi oluşturan tüm taşıtların varış noktalarına ulaştığında ulaşım ağı üzerindeki bağlara ait hacim değerleri ve en iyi *PI* değerini sağlayan sinyal parametreleri sırasıyla Tablo 5.4 ve 5.5'de verilmektedir.



Şekil 5.2 Bir adımlı benzetim ile PÖTRAD modelinin yakınsaması

Tablo 5.4 Bir adımlı benzetim ile PÖTRAD modelinden elde edilen en iyi sinyal parametreleri ve *PI* değeri

<i>PI</i>	Devre süresi <i>c</i> (sn)	Kavşak numarası <i>i</i>	Faz yeşil süreleri (sn)			Ofset süresi (sn) θ_i
			Faz 1	Faz 2	Faz 3	
			$\varphi_{i,1}$	$\varphi_{i,2}$	$\varphi_{i,3}$	
PÖTRAD	1457	1	53	41		26
		2	43	51		10
		3	51	43		28
		4	12	40	37	92
		5	16	37	36	60
		6	60	34		87

Tablo 5.5 Bir adımlı benzetim ile PÖTRAD modelinden elde edilen bağ hacimleri (ta/sa)

q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
1180	512	732	650	356	428	875	623	401	992	295	490
q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}	q_{23}	
1236	773	895	858	928	755	854	1280	1024	1243	995	

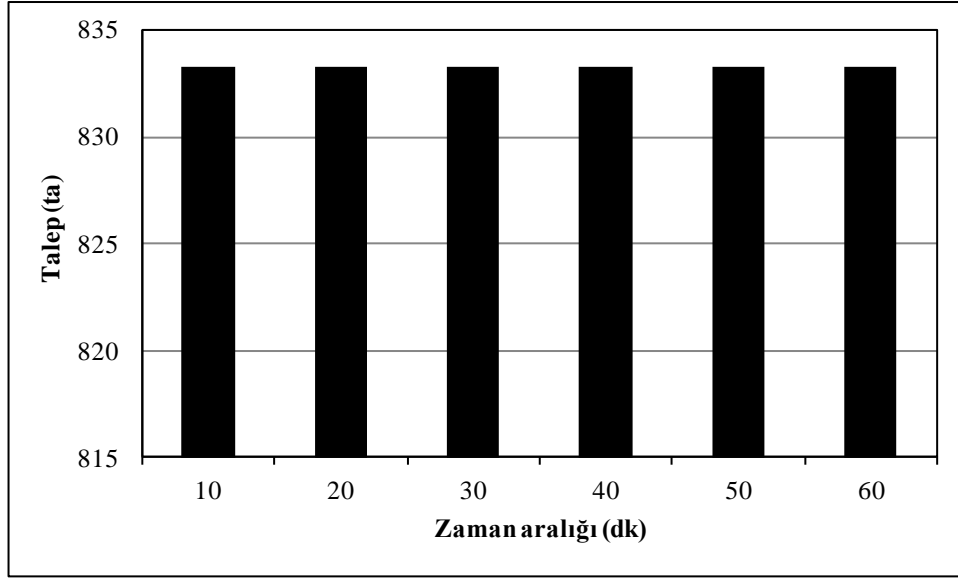
DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin sayısal uygulaması için dört tane senaryo oluşturulmuştur. Oluşturulan senaryolarla yapılan sayısal uygulamalarda PÖTRAD modelinin altında TA probleminin çözümünde Dynasmart-P yazılımının iteratif benzetim bazlı atama prosedürü kullanılarak DKD altında UAT yapılmıştır. Oluşturulan senaryolar, 5000 ta/sa'lik B-V talebinin analiz süresi içinde zamana bağlı olarak ulaşım ağına yükleme profilleri ile ilgilidir. Senaryolarla ilgili bilgiler aşağıda verilmektedir:

- **Senaryo 1:** Tablo 5.2'de verilen 5000 ta/sa'lik B-V talebinin 60 dakikalık analiz periyodu içinde eşit ve düzgün olarak ağa yüklendiği durum,
- **Senaryo 2:** Tablo 5.2'de verilen 5000 ta/sa'lik B-V talebinin %60'ının analiz periyodunun ilk 20 dakikasında, %40'ının diğer 40 dakikada eşit dağılımlı olarak ağa yüklendiği durum,
- **Senaryo 3:** Tablo 5.2'de verilen 5000 ta/sa'lik B-V talebinin %20'sinin ilk 20 dakikada, %60'ının 30 ile 40. dakikalar arasında, kalan %20'sinin de son 20 dakikada eşit dağılımlı olarak ağa yüklendiği durum,
- **Senaryo 4:** Tablo 5.2'de verilen 5000 ta/sa'lik B-V talebinin % 40'ının ilk 40 dakikada, %60'ının da son 20 dakikada eşit dağılımlı olarak ağa yüklendiği durum.

5.2.1. Senaryo 1

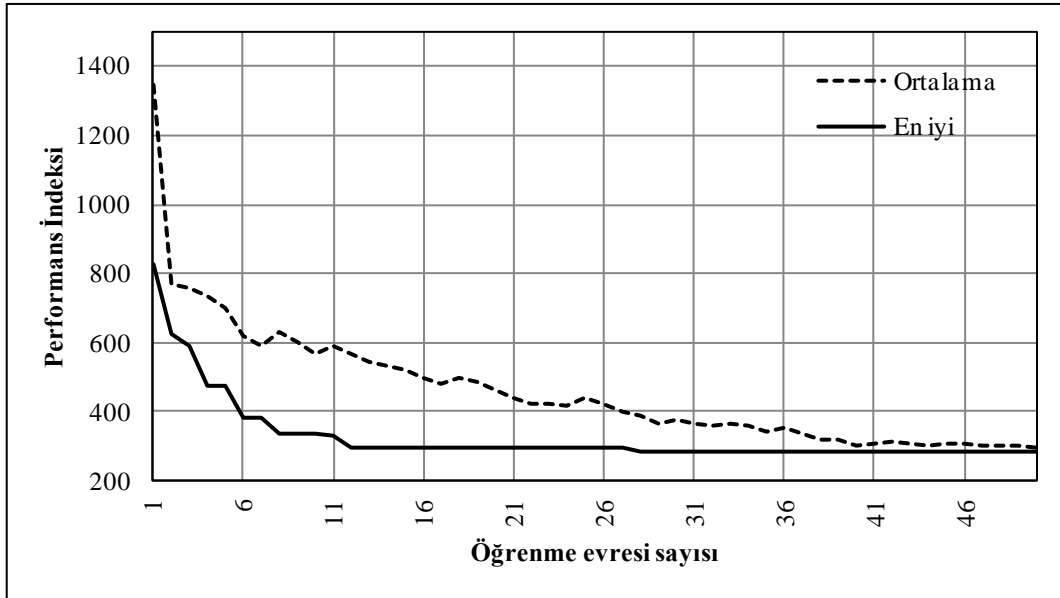
Bu senaryoda, Tablo 5.2'de verilen 5000 ta/sa'lik B-V talebi 60 dakikalık analiz periyodu içinde eşit ve düzgün olarak ulaşım ağına yüklendiği kabul edilmektedir. Bir başka deyişle, ulaşım ağına her 10 dk'da bir 833 taşıt yüklenmektedir. Şekil 5.3'den ulaşım talebinin ağa yükleme profili görülebilmektedir.

PÖTRAD modelinin ulaşım ağı üzerinde senaryo 1'e göre uygulanması sonucunda amaç fonksiyonu olan ağ PI değerinin yakınsama grafiği Şekil 5.4'de verilmektedir. Şekil 5.4'den görülebildiği gibi PÖTRAD modeli 28'inci öğrenme evresinde en iyi PI değerine 283.64'e ulaşmıştır. 1'inci öğrenme evresinde elde edilen PI değeri 829.40'dır. PÖTRAD modeli ÇU içinde her öğrenme evresi boyunca PI değerini en küçükleyen en uygun sinyal parametrelerini aramaya devam etmiş ve en küçük PI değerini 283.64 olarak bulmuştur.



Şekil 5.3 Senaryo 1’de ulaşım talebinin ağa yükleme profili

PÖTRAD modeli ulaşım ağının çözümünde oldukça kararlı bir yakınsama eğilimi göstermiştir. 50’inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. En iyi *PI* değerini sağlayan sinyal parametreleri ve dinamik denge bağ akımları sırasıyla Tablo 5.6 ve 5.7’de verilmektedir.



Şekil 5.4 Allsop ve Charlesworth ulaşım ağında senaryo 1’e göre PÖTRAD modelinin yakınsaması

Tablo 5.6 Senaryo 1’de elde edilen en iyi sinyal parametreleri ve *PI* değeri

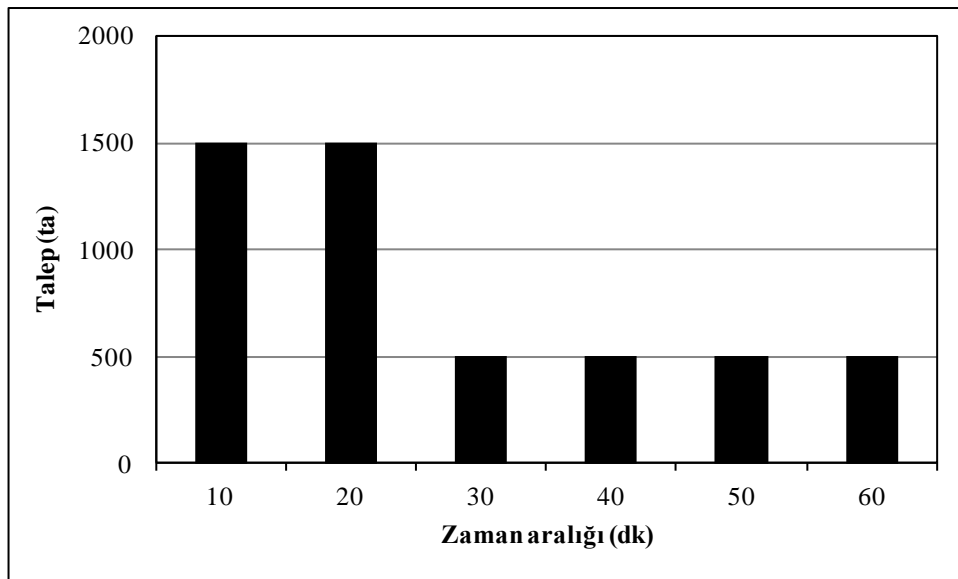
<i>PI</i>	Devre süresi <i>c</i> (sn)	Kavşak numarası <i>i</i>	Faz yeşil süreleri (sn)			Ofset süresi (sn) θ_i
			Faz 1	Faz 2	Faz 3	
			$\varphi_{i,1}$	$\varphi_{i,2}$	$\varphi_{i,3}$	
PÖTRAD	283.64	42	1	8	24	0
		2	23	9	0	
		3	23	9	0	
		4	13	7	7	0
		5	10	8	9	42
		6	21	11	0	

Tablo 5.7 Senaryo 1’de elde edilen dinamik denge bağ akımları (ta/sa)

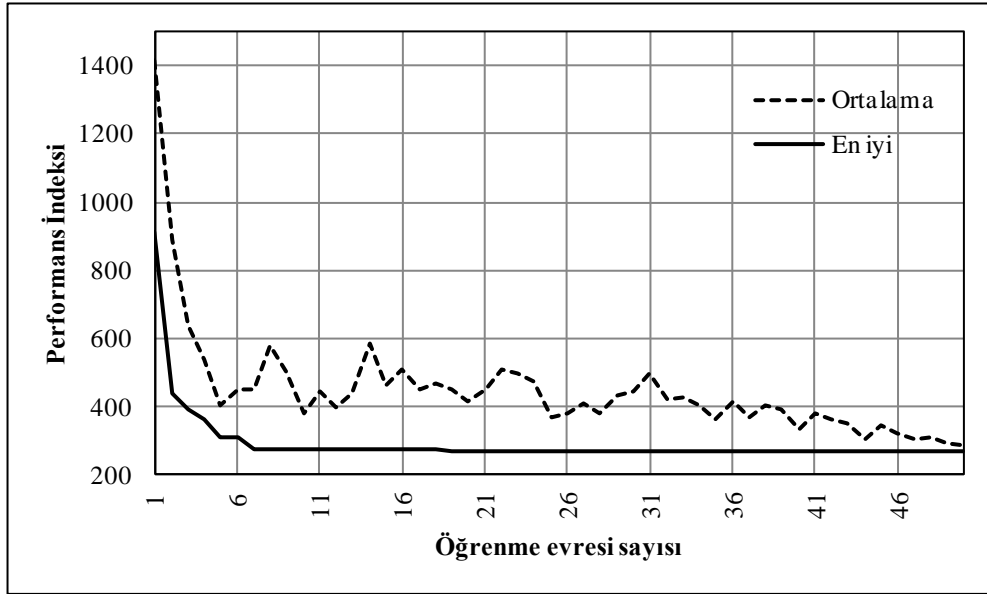
q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
1188	612	692	10	405	10	612	298	321	23	502	522
q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}	q_{23}	
31	10	177	10	10	137	137	1291	137	10	10	

5.2.2. Senaryo 2

Bu senaryoda, Tablo 5.2’de verilen 5000 ta/sa’lik ulaşım talebinin %60’lık bölümü ilk 20 dakikada, kalan talep diğer 40 dakikada eşit olarak ağa yüklenmektedir. Bir başka deyişle, talebin 3000 ta/sa’lik kısmı ilk 10 dakikada 1500 taşıt, ikinci 10 dakikada 1500 taşıt olarak kalan 2000 ta/sa’lik kısmı ise kalan 40 dakikada 500 taşıt olarak yüklenmektedir. Şekil 5.5’den ulaşım talebinin ağa yükleme profili görülebilmektedir.

**Şekil 5.5** Senaryo 2’de ulaşım talebinin ağa yükleme profili

PÖTRAD modelinin ulaşım ağı üzerinde senaryo 2'ye göre uygulanması sonucunda amaç fonksiyonu olan ağ PI değerinin yakınsama grafiği Şekil 5.6'da verilmektedir.



Şekil 5.6 Allsop ve Charlesworth ulaşım ağında senaryo 2'ye göre PÖTRAD modelinin yakınsaması

Şekil 5.6'dan görülebildiği gibi PÖTRAD modeli 19'uncu öğrenme evresinde en iyi PI değerine 270.84'e ulaşmıştır. 1'inci öğrenme evresinde elde edilen PI değeri 912.80'dir. PÖTRAD modeli ÇU içinde her öğrenme evresi boyunca PI değerini en küçükleyen en uygun sinyal parametrelerini aramaya devam etmiş ve en küçük PI değerini 270.84 olarak bulmuştur. 50'inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. En iyi PI değerini sağlayan sinyal parametreleri ve dinamik denge bağ akımları sırasıyla Tablo 5.8 ve 5.9'da verilmektedir.

Tablo 5.8 Senaryo 2'de elde edilen en iyi sinyal parametreleri ve PI değeri

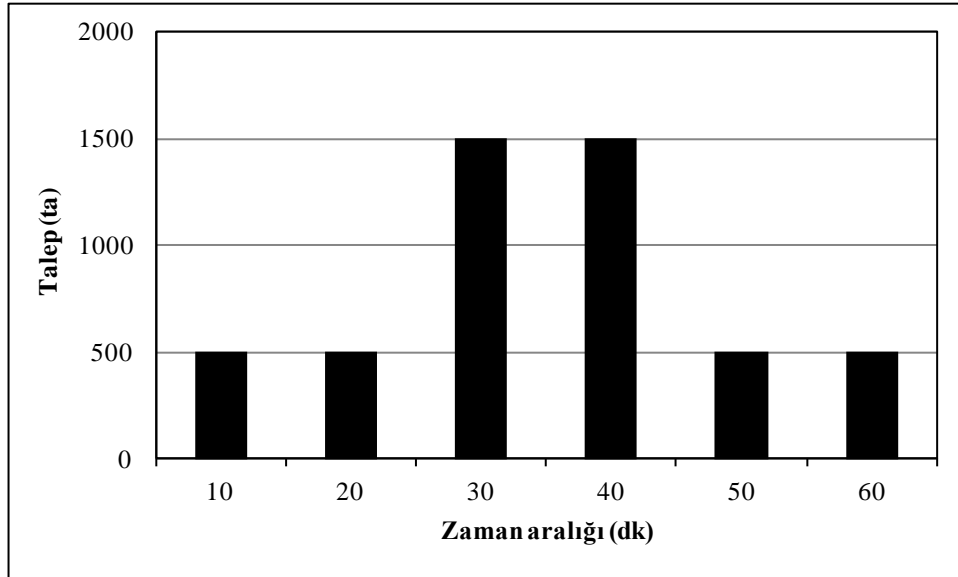
PI	Devre süresi c (sn)	Kavşak numarası i	Faz yeşil süreleri (sn)			Ofset süresi (sn) θ_i
			Faz 1	Faz 2	Faz 3	
			$\varphi_{i,1}$	$\varphi_{i,2}$	$\varphi_{i,3}$	
PÖTRAD 270.84	47	1	23	14		0
		2	21	16		45
		3	9	28		38
		4	18	7	7	45
		5	10	7	15	47
		6	30	7		47

Tablo 5.9 Senaryo 2’de elde edilen dinamik denge bađ akımları (ta/sa)

q ₁	q ₂	q ₃	q ₄	q ₅	q ₆	q ₇	q ₈	q ₉	q ₁₀	q ₁₁	q ₁₂
1179	401	778	714	655	10	401	10	182	10	615	214
q ₁₃	q ₁₄	q ₁₅	q ₁₆	q ₁₇	q ₁₈	q ₁₉	q ₂₀	q ₂₁	q ₂₂	q ₂₃	
62	615	316	10	10	327	327	102	435	22	188	

5.2.3. Senaryo 3

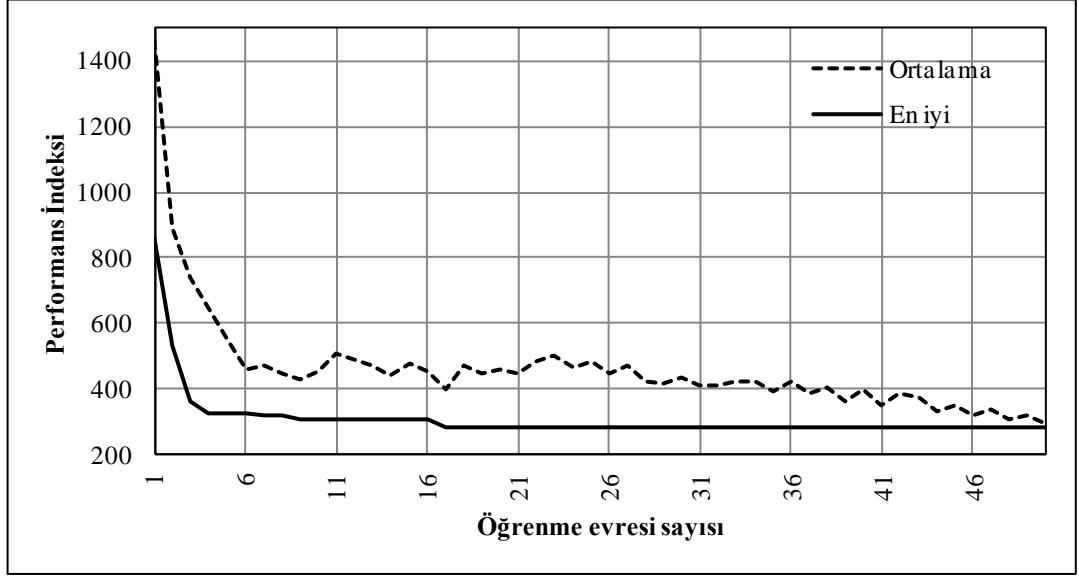
Bu senaryoda, Tablo 5.2’de verilen 5000 ta/sa’lik ulaşım talebinin %20’lik bölümü ilk 20 dakikada, %60’lık bölümü 30 ile 50. dakikalar arasında, kalan bölümde son 20 dakikada eşit olarak ulaşım ađına yüklenmektedir. Bir başka deyişle, talebin 1000 ta/sa’lik kısmı ilk 10 dakikada 500 taşıt, ikinci 10 dakikada 500 taşıt olarak, 3000 ta/sa’lik kısmı üçüncü 10 dakikada 1500 taşıt, dördüncü 10 dakikada 1500 taşıt olarak, kalan 1000 ta/sa’lik kısmı ise beşinci 10 dakikada 500 taşıt, altıncı 10 dakikada 500 taşıt olarak yüklenmektedir. Şekil 5.7’den ulaşım talebinin ađa yükleme profili görülebilmektedir.



Şekil 5.7 Senaryo 3’de ulaşım talebinin ađa yükleme profili

PÖTRAD modelinin ulaşım ađı üzerinde senaryo 3’e göre uygulanması sonucunda amaç fonksiyonu olan ađ *PI* deđeri yakınsama grafiđi Şekil 5.8’de verilmektedir. Şekil 5.8’den görülebildiđi gibi PÖTRAD modeli 17’nci öğrenme evresinde en iyi *PI* deđerine 283.37’ye ulaşmıştır. 1’inci öğrenme evresinde elde edilen *PI* deđerine 858.70’dir. PÖTRAD modeli ÇU içinde her öğrenme evresi boyunca *PI* deđerini en küçükleyen en uygun sinyal parametrelerini aramaya devam etmiş ve en küçük *PI*

değerini 283.37 olarak bulmuştur. 50'inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. En iyi PI değerini sağlayan sinyal parametreleri ve dinamik denge bağ akımları sırasıyla Tablo 5.10 ve 5.11'de verilmektedir.



Şekil 5.8 Allsop ve Charlesworth ulaşım ağında senaryo 3'e göre PÖTRAD modelinin yakınsaması

Tablo 5.10 Senaryo 3'de elde edilen en iyi sinyal parametreleri ve PI değeri

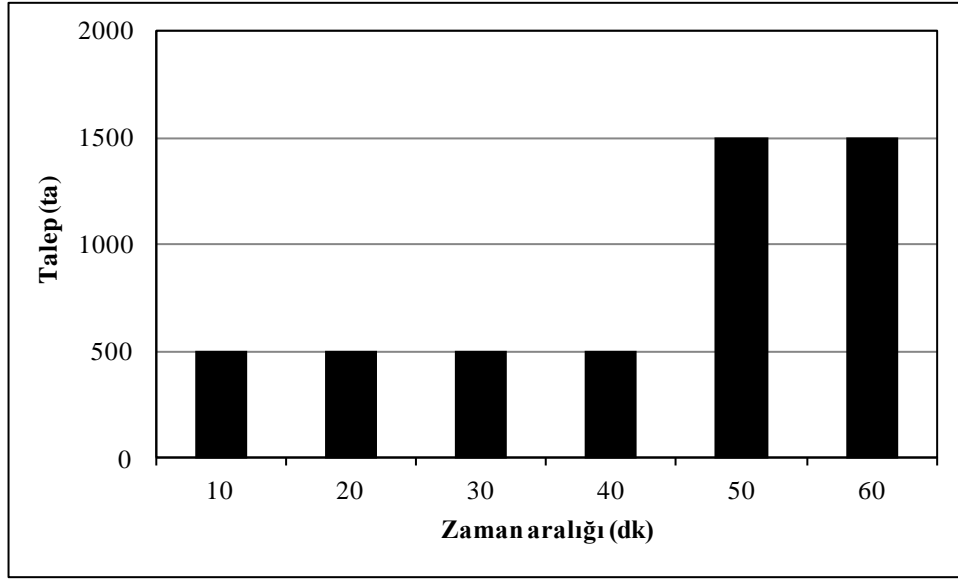
PI	Devre süresi c (sn)	Kavşak numarası i	Faz yeşil süreleri (sn)			Ofset süresi (sn) θ_i
			Faz 1	Faz 2	Faz 3	
			$\varphi_{i,1}$	$\varphi_{i,2}$	$\varphi_{i,3}$	
PÖTRAD 283.37	47	1	20	17		0
		2	23	14		0
		3	12	25		47
		4	9	13	10	47
		5	13	12	7	47
		6	17	20		12

Tablo 5.11 Senaryo 3'de elde edilen dinamik denge bağ akımları (ta/sa)

q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
1145	767	378	136	10	10	767	598	170	666	10	469
q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}	q_{23}	
96	10	10	10	10	10	10	155	128	11	10	

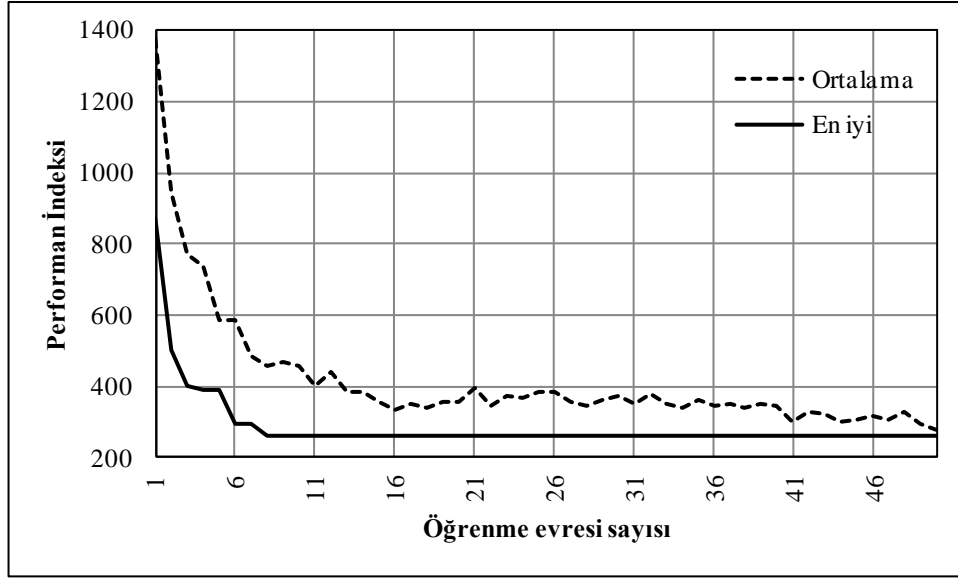
5.2.4. Senaryo 4

Bu senaryoda, Tablo 5.2’de verilen 5000 ta/sa’lik ulaşım talebinin %40’lik bölümü ilk 40 dakikada, kalan %60’lık bölümü son 20 dakikada eşit olarak ulaşım ağına yüklenmektedir. Bir başka deyişle, talebin 2000 ta/sa’lik kısmı ilk 10 dakikada 500 taşıt, ikinci 10 dakikada 500 taşıt, üçüncü 10 dakikada 500 taşıt, dördüncü 10 dakikada 500 taşıt olarak, kalan 3000 ta/sa’lik kısmı beşinci 10 dakikada 1500 taşıt, altıncı 10 dakikada 1500 taşıt olarak yüklenmektedir. Şekil 5.9’dan ulaşım talebinin ağa yükleme profili görülebilmektedir.



Şekil 5.9 Senaryo 4’de ulaşım talebinin ağa yükleme profili

PÖTRAD modelinin ulaşım ağı üzerinde senaryo 4’e göre uygulanması sonucunda amaç fonksiyonu olan ağ PI değerinin yakınsama grafiği Şekil 5.10’da verilmektedir. Şekil 5.10’dan görülebildiği gibi PÖTRAD modeli 8’inci öğrenme evresinde en iyi PI değerine 261.25’e ulaşmıştır. 1’inci öğrenme evresinde elde edilen PI değeri 872.50’dir. PÖTRAD modeli ÇU içinde her öğrenme evresi boyunca PI değerini en küçükleyen en uygun sinyal parametrelerini aramaya devam etmiş ve en küçük PI değerini 261.25 olarak bulmuştur. 50’inci öğrenme evresinde durma kriteri sağlandığı için model sonlanmıştır. En iyi PI değerini sağlayan sinyal parametreleri ve dinamik denge bağ akımları sırasıyla Tablo 5.12 ve 5.13’de verilmektedir.



Şekil 5.10 Allsop ve Charlesworth ulaşım ağında senaryo 4'e göre PÖTRAD modelinin yakınsaması

Tablo 5.12 Senaryo 4'de elde edilen en iyi sinyal parametreleri ve PI değeri

PI	Devre süresi c (sn)	Kavşak numarası i	Faz yeşil süreleri (sn)			Ofset süresi (sn) θ_i
			Faz 1	Faz 2	Faz 3	
			$\varphi_{i,1}$	$\varphi_{i,2}$	$\varphi_{i,3}$	
PÖTRAD	261.25	64	1	7	47	0
		2	40	14	18	
		3	7	47	58	
		4	35	7	7	59
		5	19	18	12	64
		6	9	45	64	

Tablo 5.13 Senaryo 4'de elde edilen dinamik denge bağ akımları (ta/sa)

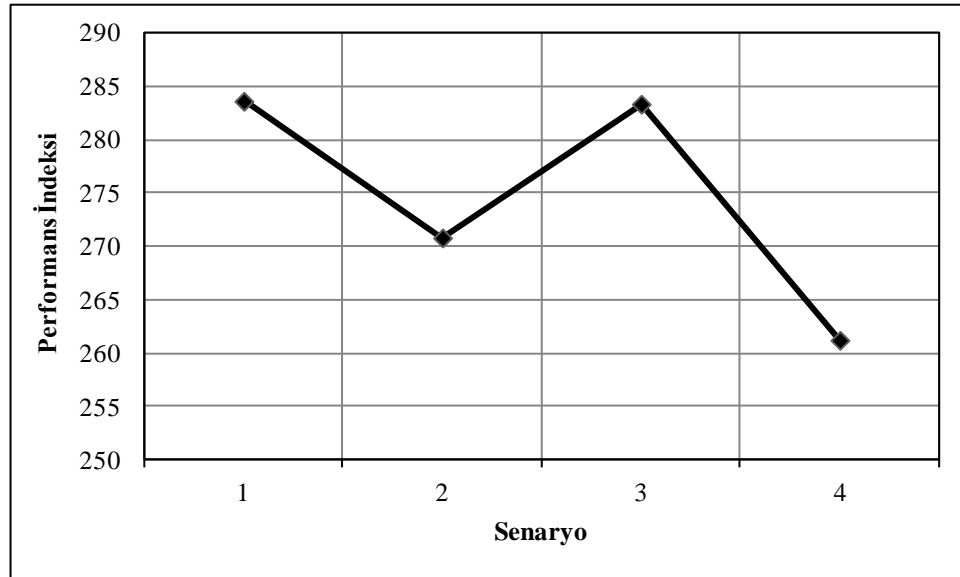
q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
1189	164	1025	908	703	10	10	117	17	65	446	387
q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}	q_{23}	
32	446	539	10	304	24	584	112	205	1264	570	

5.3. Senaryoların Karşılaştırılması

PÖTRAD modelinin sayısal uygulaması 6 kavşaklı ulaşım ağı üzerinde farklı senaryolar ile uygulanmıştır. Senaryolar, Tablo 5.2'de verilen 5000 ta/sa'lık ulaşım talebinin ulaşım ağına yükleme profilleri açısından farklılık göstermektedir. Her

senaryo sonunda elde edilen sonuçlara bakılırsa geliştirilen PÖTRAD modeli, bağ dinamik denge akımlarını bularak modelin amaç fonksiyonu olan ağ *PI* değerini enküçükleyebilmektedir. Şekil 5.11’de ulaşım ağı üzerinde her senaryonun uygulanması sonucunda elde edilen ağ *PI* değerlerinin değişimi görülebilmektedir.

Şekil 5.11’den görüldüğü gibi, ağ *PI* değerleri her senaryo için farklılık göstermektedir. Bunun nedeni, senaryolarda B-V talebinin ulaşım ağına yükleme profillerinin değişiklik göstermesinden kaynaklanmaktadır. Senaryo 1’de en yüksek *PI* değeri 283.64 olarak elde edilmiştir. Buna karşın, en düşük *PI* değeri Senaryo 4’de 261.25 olarak elde edilmiştir. Senaryo 2 ve Senaryo 3’de elde edilen *PI* değerleri sırasıyla 270.84 ve 283.37’dir. Senaryo 1 haricindeki tüm senaryolarda elde edilen ağ *PI* değerleri Senaryo 1’de elde edilen değerden azdır. Sadece Senaryo 3’de elde edilen ağ *PI* değeri Senaryo 1’de elde edilen değere diğer senaryolara göre daha



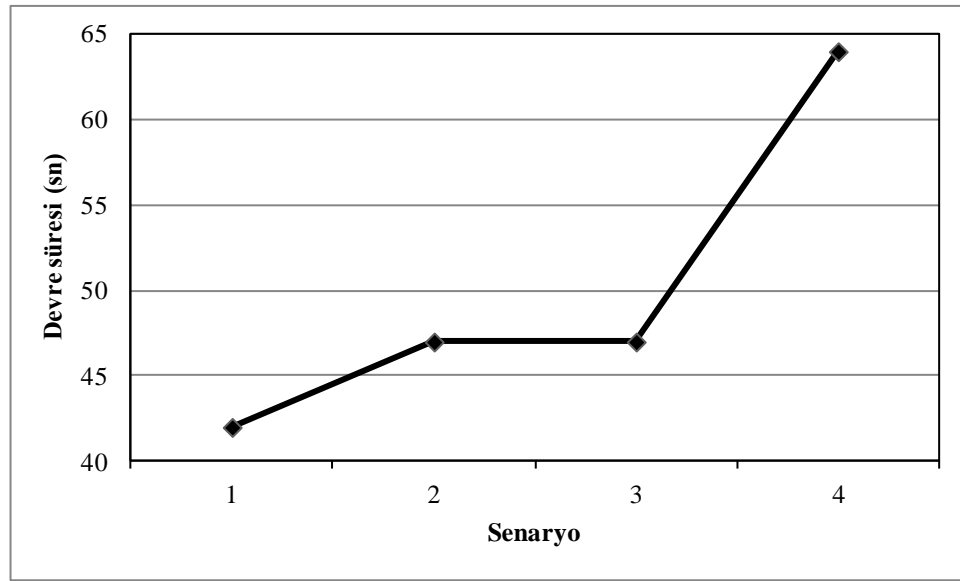
Şekil 5.11 Her senaryoda elde edilen ağ *PI* değerlerinin değişimi

yakındır. Senaryo 2’de B-V talebinin %60’lık kısmı analiz süresinin ilk 20 dakikasında, Senaryo 4’de ise B-V talebinin yine %60’lık kısmı analiz süresinin son 20 dakikasında ulaşım ağına yüklenmiştir. 5000 ta/sa’lik B-V talebinin analiz süresi boyunca düzgün dağılımlı olarak yüklenmesiyle elde edilen *PI* değeri, ulaşım talebinin analiz süresi içinde zamanla değişim göstererek ağına yüklenmesi sonucu elde edilen *PI* değerlerinden farklıdır. Elde edilen sonuçlar, ulaşım ağına yüklenen B-V talebinin analiz süresi içinde zamanla değişmesinin önemini göstermektedir. Ayrıca, statik TA yöntemlerinde analiz süresi boyunca ulaşım talebinin düzgün

dağılımlı olarak ağa yüklendiği gözönüne alınırsa DTA yöntemlerinin öneminin daha da arttığı düşünülmektedir.

Şekil 5.12’de ulaşım ağı üzerinde her senaryonun uygulanması sonucunda elde edilen devre sürelerinin değişimi görülebilmektedir. Senaryo 1’de 42 sn, Senaryo 2 ve 3’de 47 sn ve Senaryo 4’de 64 sn devre süresi elde edilmiştir.

Tablo 5.14’de ulaşım ağı üzerinde her senaryonun uygulanması sonunda elde edilen ağ ile ilgili istatistikler (toplam seyahat süresi, ortalama seyahat süresi, toplam seyahat mesafesi, ortalama seyahat mesafesi ve analiz periyodu sonunda ağ içinde kalan taşıt sayısı) verilmektedir.



Şekil 5.12 Her senaryoda elde edilen devre süresinin değişimi

Tablo 5.14 Allsop ve Charlesworth ulaşım ağına PÖTRAD modelinin uygulanması sonucu elde edilen istatistikler

Senaryo	Toplam seyahat süresi (sa)	Ortalama seyahat süresi (dk)	Toplam seyahat mesafesi (ta-km)	Ortalama seyahat mesafesi (ta-km)	Ağda kalan taşıt sayısı
1	924.90	11.09	10931.64	2.18	3964
2	850.04	10.21	11170.82	2.24	3200
3	938.77	11.39	9810.52	1.98	3744
4	586.85	7.04	11723.30	2.33	2429

Tablo 5.14’den görülebildiği gibi, toplam seyahat süresi Senaryo 1’de 924.90 sa, Senaryo 2’de 850.04 sa, Senaryo 3’de 938.77 sa ve Senaryo 4’de 586.85 sa olmuştur. Ortalama seyahat süresi ise Senaryo 1’de 11.09 dk, Senaryo 2’de 10.21 dk, Senaryo

3'de 11.39 dk ve Senaryo 4'de 7.04 dk olmuştur. Taşıtların katettikleri mesafe açısından karşılaştırma yapılacak olursa toplam seyahat mesafesi Senaryo 1'de 10931.64 ta-km, Senaryo 2'de 11170.82 ta-km, Senaryo 3'de 9810.52 ta-km ve Senaryo 4'de 11723.30 ta-km olmuştur. Ortalama seyahat mesafesi ise Senaryo 1'de 2.18 ta-km, Senaryo 2'de 2.24 ta-km, Senaryo 3'de 1.98 ta-km ve Senaryo 4'de 2.33 ta-km olmuştur. Her senaryoda ulaşım talebi aynı olmasına rağmen seyahat süresi, seyahat mesafesi gibi istatistikler açısından senaryolar arasında farklılıklar bulunmaktadır. Bu farkın nedeni, ulaşım ağına yüklenen 5000 ta/sa'lik B-V talebinin analiz süresi içinde zamanla değişiklik göstermesi ve buna bağlı olarak da arzın yani ulaşım ağı üzerindeki rotaların ve rotaları meydana getiren bağların seyahat sürelerinin ve hacimlerinin zamanla değişmesidir.

Tablo 5.14'den görülebildiği gibi, geliştirilen PÖTRAD modelinin ulaşım ağı üzerinde uygulanması sonunda 5000 ta/sa'lik B-V talebini oluşturan taşıtların tamamının seyahatlerini tamamlayamadıkları görülmektedir. Analiz süresi sonunda Senaryo 1'de 3964 taşıt, Senaryo 2'de 3200 taşıt, Senaryo 3'de 3744 taşıt ve Senaryo 4'de 2429 taşıt ağ içinde kalarak varış noktalarına ulaşamamışlardır. Bunun nedeni, benzetim bazlı DTA modellerinin özelliğidir. Statik TA modellerinde analiz süresi sonunda içinde tüm taşıtların varış noktalarına eriştikleri kabulü yapılırken DTA modellerinde analiz süresi sonunda talebi oluşturan tüm taşıtların hedeflerine varamayabilecekleri kabulü yapılmaktadır. Bu durum benzetim bazlı DTA yöntemlerinde zamana bağlı en kısa rota algoritması kullanılmasına bağlanabilmektedir. Benzetim bazlı DTA yöntemlerinde ulaşım ağına yüklenen talep zamanla değişebildiği gibi, ulaşım ağını oluşturan bağların seyahat süreleri dolayısıyla rotaların seyahat süreleri de zamanla değişebilmektedir. Böylelikle, ağ üzerindeki taşıtların bir kısmının kullandıkları rotalar üzerindeki seyahat süreleri analiz süresinden fazla olabileceğinden analiz süresi sonunda seyahatlerini henüz tamamlayamamış olabilmektedirler.

5.4. Taşıt Emisyonlarının Hesaplanması

Dünya genelinde birincil enerji kaynaklarının yaklaşık beşte biri, ulaştırma sektörü tarafından tüketilmektedir. Ayrıca bu sektör yaklaşık olarak aynı miktarlardaki sera gazları üretiminden de sorumludur. Gelecek yıllarda, gelişmiş ve gelişmekte olan

lkelerde hızla artan ulařım talebi karřısında sera gazlarının etkisinin artıřı daha kt durumlara gelebilecektir (Ceylan vd, 2007). Motorlu tařıtlardan kaynaklanan emisyonlar, sadece kresel ısınmayı tetiklediđi iin deđil, aynı zamanda ozon tabakasını deforme ettiđi ve birok canlı organizmaya zarar verdiđi iin, evrecilerin dikkatini ekmeye devam etmektedir. Dnya genelinde kullanımda olan tařıt sayısı arttıđından, motorlu tařıt emisyonlarının evre ve ekosistem zerindeki etkilerini azaltmak iin, motorların yeniden dizaynı dhil birok dzenleme yapılmaktadır.

lkelerin yođun yerleřim blgelerindeki zel tařıtlar zararlı tařıt egzoz emisyonlarının bařlıca ve en byk reticisidir. Egzoz emisyonları benzin ya da dizel gl iten yanmalı motorlarda yanan hava-yakıt karıřımı tarafından retilir. Emisyonlar aynı zamanda tařıtlar durduđunda yakıtın buharlařmasından ve sonrasında tekrar yakıt akıřı sađlandıđında da retilir. Tařıt emisyonlarının seimi, arabalarda (benzin ya da dizel) ve kamyonlarda (dizel) yakıt tipine ve alıřma karakterlerine gre eřitlilik gsterir. Tařıt egzoz emisyonlarının bařlıcaları CO, NO_x, CH_x'dir.

Tařıt trafiđinden kaynaklanan emisyonlar, ulařım talebine ve enerji tketime bađlı olarak hesaplanabilmektedir. Enerji tketime bađlı olarak yapılan hesaplamada 1 ton yakıtın tařıt tarafından yakılmasıyla elde edilen egzoz emisyonlarının en kk ve en byk deđerleri dikkate alınmaktadır. Ulařım talebine bađlı olarak yapılan hesaplamada ise trafik ile ilgili parametreler (ta-km cinsinden toplam ulařım talebi ve tařıtların model yıllarına gre yařları) dikkate alınmaktadır (Ozan vd, 2011). Bu alıřmada sayısal uygulama yapılan altı kavřaklı ulařım ađı zerinde senaryolara bađlı olarak elde edilen toplam ulařım talebi dikkate alınarak tařıt emisyon deđerleri belirlenmiřtir. Ayrıca, toplam ulařım talebinin sadece zel tařıtlardan oluřtuđu, tařıtların %50'sinin benzinli ve %50'sinin dizel olduđu kabul yapılmıřtır. Hesaplamalarda Avrupa Birliđi (AB) tarafından belirlenmiř Tablo 5.15'de verilen standart emisyon deđerleri kullanılmıřtır. Standart emisyon deđerleri AB tarafından belirlendiđi iin tařıt reticileri bu deđerlere uymak zorundadırlar. rnek olarak bir tařıt 1998 yılında retilmiře tařıt reticisi retim sırasında o tařıt iin 2000 yılında yrrlđe girecek olan Euro 3 standart emisyon deđerlerini gznne alarak retimi yapmalıdır.

Tablo 5.16’da taşıtların model yıllarına göre dağılımı verilmektedir. Taşıtların model yıllarına göre dağılımı AB emisyon standartları değerlerine göre altı dilime ayrılmıştır: model yılı 1992’den küçük olan, 1992 ile 1996 arasında olan, 1996 ile 2000 arasında olan, 2000 ile 2005 arasında, 2005 ile 2010 arasında olan ve 2010’dan büyük olan taşıtlar.

Taşıtların emisyonlarının hesaplanmasında Denklem (5.1)’de verilen formülasyon kullanılmıştır.

$$E_j = \sum_i (VP_{p,i} * e_{p,i,j} * TD * TP_{p,i}) + (VP_{d,i} * e_{d,i,j} * TD * TP_{d,i}) \quad (5.1)$$

Tablo 5.15 AB emisyon standartları (gr/ta-km) (Dieselnet.com, 2010)

Emisyon	CO	CH _x	NO _x
Benzinli taşıtlar			
Euro 1 (1992)	4.05	0.66	0.49
Euro 2 (1996)	3.28	0.34	0.25
Euro 3 (2000)	2.30	0.20	0.15
Euro 4 (2005)	1.00	0.10	0.08
Euro 5 (2010)	1.00	0.10	0.07
Euro 6 (2015)	1.00	0.10	0.06
Dizel taşıtlar			
Euro 1 (1992)	4.5	1.10	8.0
Euro 2 (1996)	4.0	1.10	7.0
Euro 3 (2000)	2.1	0.66	5.0
Euro 4 (2005)	1.5	0.46	3.5
Euro 5 (2010)	0.74	0.07	0.28
Euro 6 (2015)	0.74	0.07	0.125

Tablo 5.16 Taşıtların model yıllarına göre dağılımı

< 1992 (%)	1992< <1996 (%)	1996< <2000 (%)	2000< <2005 (%)	2005< <2010 (%)	>2010 (%)	Toplam ta-km içindeki payı (%)
Benzinli taşıtlar						
5	15	25	30	20	5	50
Dizel taşıtlar						
5	15	20	30	25	5	50

Burada, E_j j emisyonunun değerini, j emisyon numarasını (CO için $j = 1$, CH_x için $j = 2$ ve NO_x için $j = 3$), i taşıtların model yılına göre ayrımı sonunda hangi dilimde olduğunu, $VP_{p,i}$ i ’nci dilimdeki model yıllı benzinli taşıtların ortalama

yüzdesini, $e_{p,i}$ i 'nci dilimdeki model yıllık taşıtlar için benzinli motor emisyon değerlerini, TD ulaşım talebini (ta-km), $TP_{p,i}$ i 'nci dilimdeki model yıllık benzinli taşıtların ulaşım talebi içindeki yüzde cinsinden payını, $VP_{d,i}$ i 'nci dilimdeki model yıllık dizel taşıtların ortalama yüzdesini, $e_{d,i}$ i 'nci dilimdeki model yıllık taşıtlar için dizel motor emisyon değerlerini, $TP_{d,i}$ i 'nci dilimdeki model yıllık dizel taşıtların ulaşım talebi içindeki yüzde cinsinden payını göstermektedir.

Tablo 5.15'deki AB standart emisyon değerleri, Tablo 5.16'daki taşıtların model yıllarına dağılımları ve PÖTRAD modelinin dört farklı senaryo ile uygulanması sonunda elde edilen ta-km cinsinden toplam ulaşım talebi kullanılarak altı kavşaklı ulaşım ağı üzerindeki taşıt emisyon değerleri hesaplanmış ve Tablo 5.17'de verilmiştir. Elde edilen değerler analiz süresi sonunda ulaşım ağına çıkan taşıt emisyon değerleridir. Örnek olarak, Senaryo 1'den elde edilen toplam ulaşım talebi kullanılarak CO emisyon değeri Denklem (5.2)'de verildiği gibi hesaplanmıştır:

$$\begin{aligned}
E_{CO} = & (0.05 \times 4.05 \times 10931.64 \times 0.50) + (0.15 \times 3.28 \times 10931.64 \times 0.50) \\
& + (0.25 \times 2.30 \times 10931.64 \times 0.50) + (0.30 \times 1.00 \times 10931.64 \times 0.50) \\
& + (0.20 \times 1.00 \times 10931.64 \times 0.50) + (0.05 \times 1.00 \times 10931.64 \times 0.50) \\
& + (0.05 \times 4.5 \times 10931.64 \times 0.50) + (0.15 \times 4.0 \times 10931.64 \times 0.50) \\
& + (0.20 \times 2.1 \times 10931.64 \times 0.50) + (0.30 \times 1.5 \times 10931.64 \times 0.50) \\
& + (0.25 \times 0.74 \times 10931.64 \times 0.50) + (0.05 \times 0.74 \times 10931.64 \times 0.50) = 20.42 \text{ kg}
\end{aligned} \tag{5.2}$$

Tablo 5.17 Allsop ve Charlesworth ulaşım ağına açığa çıkan taşıt emisyon değerleri

Senaryo	Toplam ulaşım talebi (ta-km)	CO (kg)	CH _x (kg)	NO _x (kg)
1	10931.64	20.42	3.83	20.32
2	11170.82	20.87	3.91	20.76
3	9810.52	18.33	3.43	18.23
4	11723.30	21.90	4.10	21.79

Tablo 5.17'den görülebildiği gibi, her senaryo için elde edilen ta-km cinsinden toplam ulaşım talebi kullanılarak ağ üzerindeki taşıt emisyon değerleri hesaplanabilmiştir. Elde edilen emisyon değerlerinin farklı olmasının nedeni her senaryo sonunda elde edilen toplam ulaşım talebinin farklı olmasıdır. Elde edilen ta-km cinsinden toplam ulaşım talebinin farklı olmasının nedeni de daha önceki bölümde belirtildiği gibi 5000 ta/sa'lik B-V talebinin analiz süresi içinde zamanla

değişiklik göstermesidir. Sonuç olarak, DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin uygulanması sonucu ulaşım ağı üzerinde elde edilen ta-km cinsinden toplam ulaşım talebi kullanılarak ağ üzerinde oluşacak taşıt emisyon değerlerinin hesaplanabileceği ve buna bağlı olarak emisyon değerlerinin azaltımı için alınabilecek önlemlerin belirlenmesi konusunda yardımcı olacağı düşünülmektedir.

5.5. Sonuçlar

Bu bölümde DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin sayısal uygulaması altı kavşaklı bir ulaşım ağı üzerinde yapılmıştır. İlk sayısal uygulamada, 5000 ta/sa'lik ulaşım talebi altında ulaşım ağının performansını ve ağ üzerindeki trafiğin yapısını görebilmek amacıyla PÖTRAD modelinde Dynasmart-P modelinin bir adımlı benzetimi kullanılmıştır. Daha sonra, DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin sayısal uygulaması için dört tane senaryo oluşturulmuştur. Oluşturulan senaryolar, 5000 ta/sa'lik B-V talebinin analiz süresi içinde zamanla değişiklik göstermesi nedeniyle ortaya çıkan talep yükleme profilleri açısından farklılık göstermektedir. Oluşturulan senaryolarla yapılan sayısal uygulamalarda PÖTRAD modelinin altında TA probleminin çözümünde Dynasmart-P yazılımının iteratif benzetim bazlı atama prosedürü kullanılarak DKD altında UAT probleminin çözümü yapılmış ve elde edilen sonuçlar karşılaştırmalı olarak verilmiştir.

PÖTRAD modelinin bir adımlı benzetim ile ulaşım ağı üzerinde uygulanması sonucunda amaç fonksiyonu olan ağ PI değerini enküçükleyebilmiştir. Bir adımlı benzetim sonunda diğer bir deyişle talebi oluşturan tüm taşıtların varış noktalarına ulaştığında ulaşım ağı üzerindeki bağlara ait hacim değerleri ve en iyi PI değerini sağlayan sinyal parametreleri verilmiştir.

DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modeli her senaryoda amaç fonksiyonu olan ağ PI değerini enküçükleyebilmiştir. Model her senaryo için ulaşım ağının çözümünde oldukça kararlı bir yakınsama eğilimi göstermiştir. Her senaryoda 5000 ta/sa'lik B-V talebi aynı olmasına rağmen talebin ağa yüklenme profillerinin farklı yapıda olması farklı ağ PI değerleri ve sinyal parametreleri elde edilmesine neden olmuştur.

Her senaryo için modelin çalıştırılmasıyla ulaşım ağı üzerinden seyahat süresi ve seyahat mesafesi ile ilgili istatistikler elde edilmiştir. Her senaryoda elde edilen istatistiklerin birbirinden farklı olmasının nedeni talebin analiz süresi içinde ağa yüklenme durumu ile ilgilidir. Bir başka deyişle, yüklenen talebin analiz süresi içinde zamanla değişiklik göstermesi ve buna bağlı olarak da arzın yani ulaşım ağı üzerindeki rotaların ve rotaları meydana getiren bağların seyahat sürelerinin ve hacimlerinin zamanla değişmesi her senaryoda elde edilen istatistik değerlerini etkilemektedir. Ayrıca, her senaryonun analiz süresi sonunda talebi oluşturan taşıtların tümünün seyahatlerinin varış noktalarına ulaşamadıkları görülmektedir. Bu durum benzetim bazlı DTA yöntemlerinde zamana bağlı en kısa rota algoritması kullanılmasına bağlanabilmektedir. Benzetim bazlı DTA yöntemlerinde ulaşım ağına yüklenen talep zamanla değişebildiği gibi, ulaşım ağını oluşturan bağların seyahat süreleri dolayısıyla rotaların seyahat süreleri de zamanla değişebilmektedir. Böylelikle, ağ üzerindeki taşıtların bir kısmının kullandıkları rotalar üzerindeki seyahat süreleri analiz süresinden fazla olabileceğinden analiz süresi sonunda seyahatlerini henüz tamamlayamamış olabilmektedirler.

Ayrıca, ulaşım ağı üzerindeki taşıtların meydana getirdiği egzoz emisyon değerleri hesaplanmıştır. Bunun için ulaşım talebine bağlı bir yaklaşım kullanılmıştır. Bu yaklaşımda trafik ile ilgili parametreler (ta-km cinsinden toplam ulaşım talebi ve taşıtların model yıllarına göre yaşları) dikkate alınmaktadır. Sayısal uygulama yapılan altı kavşaklı ulaşım ağı üzerinde senaryolara bağlı olarak elde edilen ta-km cinsinden toplam ulaşım talebi dikkate alınarak taşıt emisyon değerleri belirlenmiştir. Elde edilen emisyon değerlerindeki farklılık her senaryo sonucu elde edilen ta-km cinsinden toplam ulaşım talebinin farklı olmasından kaynaklanmıştır. Gelecek bölümde tez çalışması boyunca elde edilen sonuçlar ve öneriler yer almaktadır.

6. SONUÇLAR VE ÖNERİLER

6.1. Giriş

Bu bölümde çalışmanın ana sonuçları ve gelecekte yapılması planlanan çalışmalar hakkında bilgi verilmiştir. Bölüm 6.2’de çalışmanın sonuçları, gelecek çalışmalarda yapılması planlananlar ise Bölüm 6.3’de bulunmaktadır.

6.2. Sonuçlar

Bu tez çalışmasında DKD altında UAT probleminin çözümü amaçlanmıştır. UAT probleminin çözümünde iki seviyeli programlama tekniği kullanılmıştır. İki seviyeli programlama içinde alt seviye TA problemini temsil ederken, üst seviye alt seviyenin çözümü ile elde edilen bağ akımlarına bağlı olarak ağ tasarım parametrelerinin en uygun değerlerinin belirlendiği bölüm olan sinyal kontrol problemini temsil etmektedir.

Çalışmanın 2. bölümünde, UAT, iki seviyeli programlama yöntemi, KİY yöntemi, TA, Deterministik KD, stokastik KD, DTA hakkında detaylı bilgi ve formülasyonlar verilerek literatür çalışması yapılmıştır. Literatürden görüldüğü üzere, DTA probleminin çözümü için çeşitli yaklaşımlar mevcuttur. Bu yaklaşımlardan analitik yaklaşımların trafiki modelleme konularında problemliler oldukları ve analitik yaklaşımların pratik uygulamalarda rahatlıkla kullanılamayacağı konusunda ortak bir görüş vardır. Analitik yaklaşımların bu eksikliği benzetim bazlı DTA modellerinin gelişmesine sebep olmuştur. Literatürde çeşitli benzetim bazlı DTA modelleri bulunmaktadır. Bu çalışmada, benzetim bazlı DTA modeli olan Dynasmart-P yazılımı UAT probleminin alt seviyesi olan TA probleminin çözümünde kullanılmıştır.

İkinci bölümde, sinyal kontrol problemi hakkında bilgi verilmiş ve literatür çalışması yapılmıştır. Bu çalışmada UAT probleminin üst seviyesini oluşturan sinyal kontrol probleminde faz bazlı eniyileme yazılımı olan TRANSYT-7F yazılımı kullanılmıştır. TRANSYT-7F modeli, sinyalize kavşaklı bir ağda girilen sinyal süre ve faz planlarına bağlı olarak ağ PI değerini hesaplamak için trafiği simule etmektedir. TRANSYT-7F yazılımının amacı, koordine yol ağı için sinyal parametrelerinin tekrarlı olarak ayarlanması ve sadece sinyal parametrelerindeki değişimler ile ağ PI değerini düşüren en iyi sinyal parametrelerinin bulunmasıdır.

UAT problemi çözümü ile ilgili literatürde birçok çalışma mevcut olup birbirinden farklı birçok çözüm yöntemi sunulmuştur. Ancak son yıllarda yapılan çalışmaların çoğunda sezgisel metotların kullanımı oldukça artmıştır. İkinci bölümde PÖ yöntemi ve iyileştirilmiş PÖ yöntemi hakkında kısa bilgi verilmiştir. PÖ yöntemi özellikle son yıllarda sinyal kontrol probleminin eniyilenmesinde kullanılmaya başlanmış bir yöntemdir. Ancak iyileştirilmiş ve performansı artırılmış PÖ yöntemi ile UAT probleminin çözümü literatürde uygulanmamıştır. Bu nedenle çalışmada iyileştirilmiş PÖ yöntemi ile UAT problemi çözülmüştür. UAT probleminin çözümü için iki seviyeli programlama tekniği kullanılmıştır.

Üçüncü bölümde son yıllarda farklı alanlarda uygulama alanı bulan PÖ yöntemi hakkında literatür çalışması verilmiştir. Eniyileme yeteneği çözüm uzayı daraltılarak geliştirilen iyileştirilmiş PÖ yönteminin algoritma adımları açıklanmış ve test fonksiyonları üzerinde elde edilen sonuçlar literatürdeki mevcut algoritmalarla karşılaştırılmıştır.

Literatüre bakıldığında PÖ yönteminin ayrık sinyalize sistem ile kontrol edilen kavşaklarda uygulandığı görülmüştür. PÖ yöntemi birçok alanda uygulama alanı bulmasına rağmen UAT probleminin çözümü için kullanılmasına literatürde rastlanmamıştır. Ayrıca, iyileştirilmiş PÖ yöntemi ile UAT probleminin çözümü literatürde bulunmamaktadır. Bu nedenle İPÖ yöntemi UAT probleminin üst seviyesini oluşturan sinyal kontrol probleminin çözümü için kullanılmış ve bu amaçla PÖTRANS modeli geliştirilmiştir. PÖTRANS sinyal kontrol modelinin algoritma adımları açıklanmıştır. PÖTRANS modeli sabit trafik hacimleri ile iki kavşaklı örnek ulaşım ağı üzerinde test edilerek altı kavşaklı orta ölçekli bir ulaşım ağı üzerinde uygulanmıştır. TRANSYT-7F yazılımının HC ve GA eniyileme

modelleri ile elde edilen en iyi *PI* deęerleri sırasıyla 420.5 ve 410.0'dur. PÖTRANS modeli ile elde edilen en iyi *PI* deęeri 358.5'dir. TRANSYT-7F-GA, ulaşım aęının *PI* deęerini TRANSYT-7F-HC'ye göre %2.4 iyileştirebilmiştir. Buna karşın, PÖTRANS modeli ulaşım aęı *PI* deęerini, TRANSYT-7F-HC'ye göre %14.7, TRANSYT-7F-GA'ya göre %12.5 iyileştirebilmiştir. Elde edilen sonuçlardan sabit baę akımları altında PÖTRANS modelinin ulaşım aęları üzerindeki sinyal kontrol problemini çözebildięi anlaşılmıştır. Dolayısıyla, PÖTRANS modelinin TRANSYT-7F yazılımındaki GA ve HC eniyileme modellerine bir alternatif olabileceęi düşünölmektedir.

Dördüncü bölümde, benzetim bazlı DTA modeli olan Dynasmart-P hakkında detaylı bilgi verilerek yazılımın altında çalıřan çözümlerinden bahsedilmiştir. Ayrıca, benzetim bazlı DTA modellerinde kullanılan zamana baęlı en kısa rota algoritması, karşı karşıya kalınan seyahat süresi ve anlık seyahat süresi hakkında detaylı bilgi verilmiştir.

DKD altında UAT problemini çözmek için iki seviyeli programlama teknięinden faydalanılarak PÖTRAD modeli oluşturulmuştur. Oluşturulan PÖTRAD modelinde alt seviyeyi temsil eden TA probleminin çözümünde benzetim bazlı DTA modeli olan Dynasmart-P yazılımı kullanılmış üst seviye olan sinyal kontrol probleminin çözümünde ise TRANSYT-7F yazılımının trafik modeli kullanılmıştır. İPÖ yöntemi ise her öğrenme evresinde sinyal parametrelerinin eniyilenmesi aşamasında kullanılmıştır.

DKD altında UAT probleminin çözümü için oluşturulan PÖTRAD modelinin algoritma adımları ve çalıřma şekli detaylı biçimde açıklanmıştır. Oluşturulan PÖTRAD modeli iki kavşaklı örnek ulaşım aęı üzerinde test edilmiştir. Bu amaçla, 1500 ta/sa'lık B-V talebinin örnek ulaşım aęına yüklenmesi ile ilgili iki farklı uygulama yapılmıştır. Uygulamaların farklılıkları, zamana baęlı talep durumunu yansıtmaması açısından sadece B-V talebinin ulaşım aęına yükleme profillerinden kaynaklanmaktadır. PÖTRAD modeli her uygulama için çalıştırılmış ve elde edilen uygulama sonuçları deęerlendirilmiştir.

Örnek ulaşım aęında en iyi *PI* deęeri yükleme profili 1'e göre 2.33, yükleme profili 2'ye göre 2.32 olarak bulunmuştur. DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modeli, amaç fonksiyonu olan aę *PI* deęerini

enküçükleyebilmiş ve model en iyi *PI* ile ortalama *PI* değerinin yakınsaması açısından iyi performans göstermiştir. Her iki yükleme profiline göre B-V talebinin ağ üzerine yüklenmesi sonunda elde edilen sinyal parametrelerinin birbirine benzerlik gösterdiği söylenebilir.

İki farklı yükleme profiline göre örnek ulaşım ağından elde edilen istatistikler verilmiştir. Tüm taşıtlar için toplam seyahat süresi, yükleme profili 1'e göre 56.77 sa, yükleme profili 2'ye göre 62.85 sa olmuştur. Ortalama seyahat süresi yükleme profili 1'e göre 2.29 dk iken yükleme profili 2'ye göre 2.52 dk bulunmuştur. Yükleme profili 1'e göre taşıtların katettikleri toplam mesafe 378.32 km, ortalama mesafe 0.25 km iken yükleme profili 2'ye göre taşıtların kattettikleri toplam mesafe 383.33 km, ortalama mesafe 0.26 km olarak bulunmuştur. Her iki uygulamada B-V talebi aynı olmasına rağmen, ağ üzerinde elde edilen seyahat süresi, katedilen mesafe gibi istatistikler açısından farklı sonuçlar elde edilmiştir. Bu farklılık, B-V talebinin örnek ulaşım ağına yükleme profillerinden bir başka deyişle B-V talebinin zamana bağlı olarak değişmesinden kaynaklanmıştır.

Ayrıca, benzetim bazlı DTA modellerinde kullanılan zamana bağlı en kısa rota algoritmasının sonucu olarak, analiz süresi sonunda B-V talebini oluşturan taşıtların tümünün seyahatlerinin varış noktalarına ulaşamadıkları görülmüştür. Yükleme profili 1'de 337 taşıt, yükleme profili 2'de 344 taşıt seyahatlerinin varış noktalarına ulaşamayıp ağ içinde kalmıştır.

Beşinci bölümde, DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin sayısal uygulamaları altı kavşaklı bir ulaşım ağı üzerinde yapılmıştır. İlk sayısal uygulamada, 5000 ta/sa'lik ulaşım talebi altında ulaşım ağının performansını ve ağ üzerindeki trafiğin yapısını görebilmek amacıyla PÖTRAD modelinde Dynasmart-P modelinin bir adımlı benzetimi kullanılmıştır. Daha sonra, DKD altında UAT probleminin çözümü için geliştirilen PÖTRAD modelinin sayısal uygulaması için dört tane senaryo oluşturulmuştur. Senaryolar, 5000 ta/sa'lik B-V talebinin analiz süresi içinde zamanla değişiklik göstermesi nedeniyle ortaya çıkan talep yükleme profilleri açısından farklılık göstermektedir. Oluşturulan senaryolarla yapılan sayısal uygulamalarda PÖTRAD modelinin altında TA probleminin çözümünde Dynasmart-P yazılımının iteratif benzetim bazlı atama prosedürü

kullanılarak DKD altında UAT probleminin çözümü yapılmış ve elde edilen sonuçlar karşılaştırmalı olarak verilmiştir.

PÖTRAD modeli her senaryoda amaç fonksiyonu olan ağ PI değerini enküçükleyebilmiştir. Model her senaryo için ulaşım ağının çözümünde oldukça kararlı bir yakınsama eğilimi göstermiştir. Her senaryoda 5000 ta/sa'lık B-V talebi aynı olmasına rağmen talebin ağa yüklenme profillerinin farklı yapıda olması farklı ağ PI değerleri ve sinyal parametreleri elde edilmesine neden olmuştur.

Her senaryo için modelin çalıştırılmasıyla ulaşım ağı üzerinden seyahat süresi ve seyahat mesafesi ile ilgili istatistikler elde edilmiştir. Her senaryoda elde edilen istatistiklerin birbirinden farklı olmasının nedeni talebin analiz süresi içinde ağa yüklenme durumu ile ilgilidir. Bir başka deyişle, yüklenen talebin analiz süresi içinde zamanla değişiklik göstermesi ve buna bağlı olarak da arzın yani ulaşım ağı üzerindeki rotaların ve rotaları meydana getiren bağların seyahat sürelerinin ve hacimlerinin zamanla değişmesi her senaryoda elde edilen istatistik değerlerini etkilemektedir. Ayrıca, her senaryonun analiz süresi sonunda talebi oluşturan taşıtların tümünün seyahatlerinin varış noktalarına ulaşamadıkları görülmektedir. Bu durum benzetim bazlı DTA yöntemlerinde zamana bağlı en kısa rota algortiması kullanılmasına bağlanabilmektedir. Benzetim bazlı DTA yöntemlerinde ulaşım ağına yüklenen talep zamanla değişebildiği gibi, ulaşım ağını oluşturan bağların seyahat süreleri dolayısıyla rotaların seyahat süreleri de zamanla değişebilmektedir. Böylelikle, ağ üzerindeki taşıtların bir kısmının kullandıkları rotalar üzerindeki seyahat süreleri analiz süresinden fazla olabileceğinden analiz süresi sonunda seyahatlerini henüz tamamlayamamış olabilmektedirler.

Ayrıca, ulaşım ağı üzerindeki taşıtların meydana getirdiği egzoz emsiyon değerleri hesaplanmıştır. Bunun için ulaşım talebine bağlı bir yaklaşım kullanılmıştır. Bu yaklaşımda trafik ile ilgili parametreler (ta-km cinsinden toplam ulaşım talebi ve taşıtların model yıllarına göre yaşları) dikkate alınmaktadır. Sayısal uygulama yapılan altı kavşaklı ulaşım ağı üzerinde senaryolara bağlı olarak elde edilen ta-km cinsinden toplam ulaşım talebi dikkate alınarak taşıt emisyon değerleri belirlenmiştir. Elde edilen emisyon değerlerindeki farklılık her senaryo sonucu elde edilen ta-km cinsinden toplam ulaşım talebinin farklı olmasından kaynaklanmıştır.

Statik TA yöntemleri analiz süresi boyunca ortalama ve sabit koşullar üzerine odaklandıkları düşünülürse ulaşım ağı üzerinde zamanla değişen trafik akımlarını ve şartlarını tam yansıtamamaktadırlar. Statik TA modelleri kısa süreli ve geçici seviyedeki tıkanıklık sorununun çözümünde yetersiz kalmakta ve bu durumun DTA modellerinin önemini açıkça ortaya çıkardığı düşünülmektedir. Ayrıca, analitik DTA modelleri konunun teorik içyüzü ile ilgilenirken benzetim bazlı DTA modelleri pratik uygulamalar için daha uygun olmaktadır. Bu nedenlerden dolayı geliştirilen PÖTRAD modelinin DKD altında UAT probleminin çözümünde kullanılabileceği sonucuna varılmıştır.

6.3. Gelecek Çalışmalar

Gelecek çalışmalarda UAT probleminin çözümü için yapılması planlananlar aşağıda verilmiştir:

- Çalışmada geliştirilen PÖTRAD modeli DKD altında UAT problemini çözmektedir. PÖTRAD modelinin DSO altında UAT probleminin çözümünde kullanılması gelecek çalışmalara bırakılmıştır.
- DKD altında UAT problemi çözümü için geliştirilen PÖTRAD modeli orta ölçekli bir ulaşım ağı üzerinde uygulanmıştır. Geliştirilen PÖTRAD modeli bağ sayısının ve B-V talebinin daha fazla olduğu büyük ölçekli gerçek ulaşım ağlarının üzerinde uygulanmalı ve elde edilen sonuçlar analiz edilmelidir.
- İPÖ yöntemi statik bir başka deyişle Stokastik ve Deterministik KD altındaki UAT probleminin çözümü için kullanılmalı ve elde edilen sonuçlar diğer sezgisel yöntemlerle karşılaştırılmalıdır.
- DKD altından UAT çözümü için geliştirilen PÖTRAD modeli sonuçları ile diğer benzetim bazlı DTA modellerinin kullanıldığı UAT modellerinin sonuçlarının karşılaştırılması çalışma kapsamı dışında olması nedeniyle gelecek çalışmalara bırakılmıştır.

KAYNAKLAR

- Abdelfatah, A. S., and Mahmassani, H. S.,** 1998: System Optimal Time-Dependent Path Assignment and Signal Timing in Traffic Network, *Transportation Research Record* 1645, pp. 185–93.
- Abdelfatah, A. S., and Mahmassani, H. S.,** 2001: A simulation based signal optimization algorithm within a dynamic traffic assignment framework, *Proceedings of 2001 IEEE Intelligent Transportation Systems Conference*, Oakland, CA, pp. 428–33.
- Abdulaal, M., and LeBlanc, L. J.,** 1979: Continuous equilibrium network design models. *Transportation Research Part B*, 13: 19-32.
- Abdulhai, B., and Kattan, L.,** 2003: Reinforcement learning: Introduction to theory and potential for transport applications, *Canadian Journal of Civil Engineering*, 30, 981-991.
- Abdulhai, B., Pringle, R., and Karakoulas, G.J.,** 2003: Reinforcement learning for true adaptive traffic signal control, *Journal of Transportation Engineering*, 129(3), 278-285.
- Allsop, R. E.,** 1974: Some possibilities for using traffic control to influence trip distribution and route choice, *Transportation Traffic Theory, Proceedings of the Sixth International Symposium on Transportation and Traffic Theory*, Amsterdam, 345-374.
- Allsop, R.E., and Charlesworth, J.A.,** 1977: Traffic in a signal-controlled road network: an example of different signal timings including different routings, *Traffic Engineering Control*, 18(5), 262-264.
- Allsop, R.E.,** 1992: Evolving application of mathematical optimisation in design and operation of individual signal-controlled road junctions, *Mathematics in Transport Planning and Control*, (Griffiths J.D. Ed), Oxford University Press. Oxford, 1-25.

- Babazadeh A., Poorzahedy, H., and Nikoosokhan, S.,** 2011: Application of particle swarm optimization to transportation network design problem, *Journal of King Saud University – Science*, 23, 293–300.
- Başkan, Ö.,** 2009: Karınca Kolonisi Optimizasyonu ile Ulaşım Ağ Tasarımı, Doktora Tezi, *Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü*, Denizli, 140s.
- Baskan, O., Haldenbilen, S., Ceylan, H., and Ceylan, H.,** 2009: A new solution algorithm for improving performance of ant colony optimization. *Applied Mathematics and Computation*, 211: (1), 75-84.
- Bazzan, A.L.C., Oliveira, D., and Silva, B.C.,** 2010: Learning in groups of traffic signals, *Engineering Applications of Artificial Intelligence*, 23, 560-568.
- Beckmann, M.J., McGuire, C.B., and Winsten, C.B.,** 1956: Studies in Economics of Transportation. *Yale University Press*, New Haven.
- Bell, M. G. H., and Iida, Y.,** 1997: Transportation Network Analysis. *John Wiley & Son Ltd*, England, pp. 216.
- Bell, M.G.H., Shield, C.M., Busch, F., and Kruse, G.,** 1997: Stochastic user equilibrium path flow estimator. *Transportation Research*, 5C, 197-210.
- Ben-Akiva, M.E., Koutsopoulos, H.N., and Mishalani, R.,** 1997a: Development of a Route Guidance Generation System for Real-Time Application, *Proceedings of the 8th International Federation of Automatic Control Symposium on Transportation Systems*, Chania, Greece, pp. 433-438.
- Ben-Akiva, M.E., Koutsopoulos, H.N., Mishalani, R., and Yang, Q.,** 1997b: Simulation Laboratory for Evaluating Dynamic Traffic Management Systems, *ASCE Journal of Transportation Engineering*, 123(4), 283-289.
- Ben-Akiva, M.E., Bierlaire, M., Koutsopoulos, H.N., and Mishalani, R.,** 1998: DynaMIT: A Simulation-based System for Traffic Prediction and

Guidance Generation, *Proceedings of the 3rd Triennial Symposium on Transportation Systems*, San Juan, Puerto Rico.

- Bingham, E.**, 2001: Reinforcement learning in neurofuzzy traffic signal control, *European Journal of Operation Research*, 131, 232-241.
- Birge, J.R., and Ho, J.K.**, 1993: Optimal Flows in Stochastic Dynamic Networks with Congestion, *Operations Research*, 41(1), 203-216.
- Boyce, D.E., Ran, B., and LeBlanc, L.J.**, 1995: Solving an Instantaneous Dynamic User-Optimal Route Choice Model, *Transportation Science* 29, 128-142.
- Cai, C., Wong, C.K., and Heydecker, B.G.**, 2009: Adaptive traffic signal control using approximate dynamic programming, *Transportation Research Part C*, 17, 456-474.
- Camponogara, E., and Kraus, W.Jr.**, 2003: Distributed learning agents in urban traffic control, Moura-Pires, F., and Abreu, S. (Eds.), *EPIA*, pp.324–335.
- Carey, M.**, 1986: A Constraint Qualification for a Dynamic Traffic Assignment Model, *Transportation Science*, 20, 55-58.
- Carey, M.**, 1987: Optimal Time Varying Flows On Congested Networks, *Operations Research*, 35(1), 58-69.
- Carey, M.**, 1992: Nonconvexity of the Dynamic Assignment Model, *Transportation Research*, 26B(2), 127-133.
- Carey, M.**, 2009: A framework for user equilibrium dynamic traffic assignment, *Journal of the Operational Research Society*, 60, 395-410.
- Carey, M., and Subrahmanian, E.**, 2000: An Approach to Modelling Time-Varying Flows on Congested Networks, *Transportation Research*, 34B(3), 157-183.
- Ceylan, H.**, 2002: A Genetic Algorithm Approach to the Equilibrium Network Design Problem, PhD Thesis, *University of Newcastle Upon Tyne*, England, pp.189.

- Ceylan, H., and Bell, M. G. H.,** 2004: Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing, *Transportation Research Part B*, 38, 329-342.
- Ceylan, H., and Bell, M. G. H.,** 2005: Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion, *Transportation Research Part B*, 39, 169-185.
- Ceylan, H.,** 2006: Developing Combined Genetic Algorithm—Hill-Climbing Optimization Method for Area Traffic Control, *Journal of Transportation Engineering*, 132(8), 663-671.
- Ceylan, H., Karavaşin, M., ve Haldenbilen, S.,** 2007: Gelişmiş ve Gelişmekte Olan Ülkelerde Ulaşım: Yenilenebilir Enerjiye Karşın Enerji Azal(t)ımı, 7. *Ulaştırma Kongresi*, İstanbul, s. 438-447.
- Ceylan, H.,** 2009: Şehir içi Ulaşım Ağlarının Armoni Araştırması Optimizasyon Tekniği İle Tasarımı, Doktora Tezi, *Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü*, Denizli, 130s.
- Chelouah, R., and Siarry, P.,** 2000: Tabu Search applied to global optimization, *European Journal of Operational Research*, 123, 256-270.
- Chen, H.-K., and Hsueh, C.F.,** 1998: A Model and An Algorithm for the Dynamic User-Optimal Route Choice Problem, *Transportation Research*, 32B(3), 21-234.
- Chen, A., Kim, J., Lee, S., and Kim, Y.,** 2010: Stochastic multi-objective models for network design problem, *Expert Systems with Applications*, 37, 1608–1619.
- Chen, L.-W., and Hu, T.-Y.,** 2009: Dynamic Equilibrium for Combined Signal Settings and Time-dependent Traffic Assignment, *Proceedings of the Eastern Asia Society for Transportation Studies*, Vol.7, 323-338.
- Chen, A., and Xu, X.,** 2012: Goal programming approach to solving network design problem with multiple objectives and demand uncertainty, *Expert Systems with Applications*, 39, 4160–4170.

- Chen, O. J., and Ben-Akiva, M. E.,** 1998: Game-theoretic formulations of the interactions between dynamic traffic control and dynamic traffic assignment, *Transportation Research Record*, 1617, 179–88.
- Chen, Y., Mabu, S., Shimada, K., and Hirasawa, K.,** 2009: A genetic network programming with learning approach for enhanced stock trading model, *Expert Systems with Applications*, 36, 12537-12546.
- Chen, C., Chang, K., and Ho, S.,** 2011: Improved framework for particle swarm optimization: Swarm intelligence with diversity-guided random walking, *Expert Systems and Applications*, 38, 10, pp. 12214-12220.
- Cheng, S. F., Epelman, M. A., and Smith, R. L.,** 2004: Cosign: A Fictitious Play Algorithm for Coordinated Traffic Signal Control, *IOE Technical Report 04-08*, University of Michigan, Ann Arbor, MI.
- Chiou , S.W.,** 1998: Optimization of area traffic control for equilibrium network flows, PhD Thesis, *University College London*, England.
- Chiou, S.W.,** 2005: Bi-level programming for the continuous transport network design problem, *Transportation Research Part B*, 39, 361-383.
- Chiou S.W.,** 1999: Optimization of area traffic control for equilibrium network flows, *Transportation Science*, 33(3), 279-289.
- Dafermos, S.,** 1980: Traffic Equilibrium and Variational Inequalities, *Transportation Science* 14, 42-54.
- Daganzo, C.,** 1994: Cell Transmission Model: A Dynamic Representation of Highway Traffic Consistent with the Hydrodynamic Theory, *Transportation Research* 28B(4), 269-287.
- Daganzo, C., and Sheffi Y.,** 1977: On Stochastic Models of Traffic Assignment, *Transportation Science*, 11: 253-274.
- Davis, G. A.,** 1994: Exact local solution of the continuous network design problem via stochastic user equilibrium assignment, *Transportation Research Part B*, 28, 61-75.
- Dial, R.B.,** 1971: A probabilistic multipath traffic assignment algorithm which obviates path enumeration, *Transportation Research* 5: (2), 81-111.

- Dial, R.B.**, 2006: A path-based user equilibrium traffic assignment algorithm that obviates path storage and enumeration, *Transportation Research Part B*, 40, 917-936.
- Dickson, T. J.**, 1981: A note on traffic assignment and signal timings in a signal controlled road network, *Transportation Research*, 15B: (4) 267-271.
- Dieselnet.com**, 2010: European Union emission standards, <<http://www.dieselnet.com/standards/eu/>>
- Dynasmart-P**, 2007: Intelligent Transportation Network Planning Tool User's Guide v1.3.0, *U.S. Department of Transportation, Federal Highway Administration*, Virginia, 22101.
- Fisk, C.S.**, 1984: Game theory and transportation systems modelling, *Transportation Research Part B*, 18: (4-5) 301-313.
- Friesz, T.L., Bernstein, D., Mehta, N.J., Tobin, R.L, and Ganjalizadeh, S.**, 1989: Dynamic Network Traffic Assignment Considered as a Continuous Time Optimal Control Problem, *Operations Research*, 37(6), 893-901.
- Friesz, T.L., Bernstein, D., Smith, T.E., Tobin, R.L, and Wie, B.W.**, 1993: Day-to-Day Dynamic Network Disequilibrium and Idealized Traveler Information Systems, *Operations Research*, 42(6), 1120-1136.
- Gallivan, S., and Heydecker, B.G.**, 1988:.) Optimising the control performance of traffic signals at a single junction, *Transportation Research*, 22B(5), 357-370.
- Gallo, M., D'Acierno, L., and Montella, B.**, 2010: A meta-heuristic approach for solving the urban network design problem, *European Journal of Operational Research*, Vol. 201, Issue.1, 144-157.
- Gartner, N.H.**, 1974: Area traffic control and network equilibrium methods, Traffic Equilibrium Methods, *Proceedings of the International Symposium*, Montreal, pp. 274-297.
- Geem, Z.W., Kim, J.H., and Loganathan, G.V.**, 2001: A new heuristic optimization algorithm: harmony search, *Simulation* 76, 2, 60-68.

- Gershwin, S. B., and Tan, H. N.,** 1979: Hybrid optimization: optimal static traffic control constrained by drivers' route choice behaviour, *Laboratory for Information and Decision System Report LIDS*, Massachusetts Institute of Technology, 870s.
- Ghali, M.O., and Smith, M.J.,** 1992a: A Dynamic Traffic Assignment Model, *The 71st TRB Annual Meeting*, Washington D.C.
- Ghali, M.O., and Smith, M.J.,** 1992b: Dynamic Road Pricing and Dynamic Traffic Assignment for Congested Large Networks, *The 39th North American Regional Science Association International Conference*, Chicago.
- Ghali, M.O., and Smith, M.J.,** 1992c: Optimal Dynamic Traffic Assignment of a Congested City Network, *Proceedings of the Second International Capri Seminar on Urban Traffic Networks*, Capri, Italy.
- Ghali, M.O., and Smith, M.J.,** 1993: Traffic Assignment, Traffic Control and Road Pricing, *Proceedings of the Twelfth International Symposium on Transportation and Traffic Theory*, Daganzo (ed.), University of California, Berkeley, California, pp. 147-170.
- Guogiang, Z., and Jian, L.,** 2007: Genetic algorithm for continuous network design problem, *Journal of Transportation Systems Engineering and Information Technology*, 7: (1), 101-105.
- Hamzacebi, C.,** 2008: Improving genetic algorithms' performance by local search for continuous function optimization, *Applied Mathematics and Computation*, 196, 309-317.
- Heydecker, B.G.,** 1996: A decomposed approach for signal optimisation in road networks, *Transportation Research*, 30B(2), 99-114.
- Ho, J.K.,** 1980: A Successive Linear Optimization Approach to the Dynamic Traffic Assignment Problem, *Transportation Science* 14, 295-305.
- Hunt, P.B., Robertson, D.I., Bretherton, R.D., and Royle, M.C.,** 1982: The SCOOT on-line traffic signal optimisation technique, *Traffic Engineering and Control*, 23: 190-192.

- Janson, B.N.**, 1991a: Convergent Algorithm for Dynamic Traffic Assignment, *Transportation Research Record* 1328, 69-80.
- Janson, B.N.**, 1991b: Dynamic Traffic Assignment for Urban Road Networks, *Transportation Research* 25B(2/3), 143-161.
- Janson, B.N.**, 1995: Network Design Effects of Dynamic Traffic Assignment, *ASCE Journal of Transportation Transportation Engineering*, Vol.121, No.1, 1-13.
- Jayakrishnan, R., Mahmassani, H.S., and Hu, T.Y.**, 1994: An Evaluation Tool for Advanced Traffic Information and Management Systems in Urban Networks, *Transportation Research* 2C(3), 12-147.
- Jin, Y., Meng, X., and Ziyou, G.**, 2009: Sensitivity analysis of simulated annealing for continuous network design problems, *Journal of Transportation Systems Engineering and Information Technology*, 9: (3), 64-70.
- Kaelbling, L.P., Littman, M.L., and Moore, A.W.**, 1996: Reinforcement Learning: A Survey, *Journal of Artificial Intelligence Research*, 4, 237-285.
- Kao, Y.-T., and Zahara, E.**, 2008: A hybrid genetic algorithm and particle swarm optimization for multimodal functions, *Applied Soft Computing*, 8, 849-857.
- Karoonsoontawong, A.K., and Waller, S.T.**, 2009: Application of Reactive Tabu Search for Combined Dynamic User Equilibrium and Traffic Signal Optimization Problem, *Transportation Research Record*, 2090, pp. 29-41.
- Kwon, Y.D., Kwon, S.B., and Kim, J.**, 2003: Convergence enhanced genetic algorithm with successive zooming method for solving continuous optimization problems, *Computers and structures*, 81, 1715-1725.
- Leblanc, L.J.**, 1975: An algorithm for the discrete network design problem, *Transportation Science*, 9: 183-199.

- Lee C.**, 1998: Combined traffic signal control and traffic assignment: Algorithms, implementation and numerical results, Ph.D. dissertation, *The University of Texas at Austin*, Texas.
- Leonard, D., Gower, P., and Taylor, N.B.**, 1989: CONTRAM: Structure of the Model, *TRRL Research Report RR 178*, Transport and Road Research Laboratory, Crowthorne, United Kingdom.
- Lim, Y., Heydecker, B. G., and Lee, S.**, 2005: A continuous network design model in stochastic user equilibrium based on sensitivity analysis, *Journal of Advanced Transportation*, 39: (1), 63-79.
- Little, J., Kelson, M., and Gartner, N.**, 1981: MAXBAND: a program for setting signals on arteries and triangular networks, *Transportation Research Record 795*, National Research Council, Washington DC, pp. 40-46.
- Liu, F., and Zeng, G.**, 2009: Study of genetic algorithm with reinforcement learning to solve the TSP, *Expert Systems with Applications*, 36, 6995-7001.
- Liu, B., Wang, L., Jin, Y.-H., Tang, F., and Huang, D.-X.**, 2005: Improved particle swarm optimization combined with chaos, *Chaos, Solitons and Fractals*, 25, 1261-1271.
- Lowrie, P.R.**, 1982: SCATS principles, methodology, algorithm, *IEE conference on road traffic signaling*, London, pp. 67-70.
- Luk, J.**, 1984: Two traffic-responsive area traffic control methods: SCATS and SCOOT, *Traffic Engineering and Control*, 25: (1), 17-18.
- Mahmassani, H.S., and Peeta, S.**, 1992: System Optimal Dynamic Assignment for Electronic Route Guidance in a Congested Traffic Network, *Proceedings of the Second International Capri Seminar on Urban Traffic Networks*, Capri, Italy.
- Mahmassani, H.S., and Peeta, S.**, 1993: Network Performance Under System Optimal and User Equilibrium Dynamic Assignments: Implications for ATIS, *Transportation Research Record 1408*, 83-93.
- Mahmassani, H.S., and Peeta, S.**, 1995: System Optimal Dynamic Assignment for Electronic Route Guidance in a Congested Traffic Network, In

Gartner and Improta (eds.), *Urban Traffic Networks: Dynamic Flow Modeling and Control*, Berlin: Springer-Verlag, pp. 2-27.

- Maravall, D., de Lope, J., and Martin H., J.A.**, 2009: Hybridizing evolutionary computation and reinforcement learning for the design of almost universal controllers for autonomous robots, *Neurocomputing*, 72, 887-894.
- Marcotte, P.**, 1983: Network optimisation with continuous control parameters, *Transportation Science*, 17: (2), 181-197.
- Martin, A., and Brauer, W.**, 2000: Fuzzy model-based reinforcement learning, *European Symposium on Intelligent Techniques (ESIT)*, Aachen Germany, pp. 14-15.
- Mauro V., and Di Taranto, C.**, 1989: UTOPIA, *Proceedings of the 6th IFAC/IFORS Conference on Control, Computers and Communications in Transport*, CCCT'89, Paris, France.
- McTrans**, 2008: TRANSYT-7F Users Guide, Release 11.3, McTrans Center, University of Florida, Gainesville, Florida.
- Merchant, D.K., and Nemhauser, G.L.**, 1978a: A model and an algorithm for the dynamic traffic assignment problem, *Transportation Science*, 12(3), 183-199.
- Merchant, D.K., and Nemhauser, G.L.**, 1978b: Optimality Conditions for a Dynamic Traffic Assignment Model, *Transportation Science*, 12(3), 200-207.
- Murat, Y.Ş.**, 1996: Denizli Şehiriçi Kavşaklarındaki Trafik Akımlarının Bilgisayarla İncelenmesi, Yüksek Lisans Tezi, *Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü*, Denizli, 156s.
- Nagurney, A.**, 1998: Network Economics: A Variational Inequality Approach, *Boston: Kluwer Academic Publishers*.
- Nunes, L., and Oliveira, E.C.**, 2004: Learning from multiple sources, in: Jennings, N., Sierra, C., Sonenberg, L., and Tambe, M. (Eds.), *Proceedings of the 3rd International Joint Conference on Autonomous Agents and*

Multi Agent Systems, AAMAS, Vol. 3. IEEE Computer Society, New York, USA, pp.1106–1113.

Oppenheim, N., 1995: Urban Travel Demand Modeling: From Individual Choices to General Equilibrium. *John Wiley & Sons, Inc.*, New York, pp. 480.

Ortuzar, J. D., and Willumsen, L. G., 1994: Modelling Transport (2nd Edition). *John Wiley and Sons*, Chichester, pp. 433.

Ozan, C., Haldenbilen, S., and Ceylan, H., 2011: Estimating emissions on vehicular traffic based on projected energy and transport demand on rural roads: Policies for reducing air pollutant emissions and energy consumption, *Energy Policy*, 39, 2542-2549.

Peeta S., 1994: System optimal dynamic traffic assignment in congested networks with advanced information systems. Doctoral dissertation, *The University of Texas at Austin*.

Peeta S., and Mahmassani, H.S., 1995a: System optimal and user equilibrium time-dependent traffic assignment in congested networks, *Annals of Operations Research* 60, 81-113.

Peeta S., and Mahmassani, H.S., 1995b: Multiple User Classes Real-Time Traffic Assignment For Online Operations: A Rolling Horizon Solution Framework, *Transportation Research C*, Vol. 3. No. 2. pp. 83-98.

Peeta S., and Ziliaskopoulos, A.K., 2001: Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future, *Networks and Spatial Economics*, 1, 233-265.

Poorzahedy, P., and Turnquist, M. A., 1982: Approximate algorithms for the discrete network design problem, *Transportation Research Part B*, 16: 45-56.

Poorzahedy, H., and Abulghasemi, F., 2005: Application of Ant system to network design problem, *Transportation*, 32: 251-273.

Poorzahedy, H., and Rouhani, O.M., 2007: Hybrid meta-heuristic algorithms for solving network design problem, *European Journal of Operational Research*, 182: (2), 578-596.

- Putha R., Quadrifoglio L., and Zechman E.,** 2001: Comparing ant colony optimization and genetic algorithm approaches for solving traffic signal coordination under oversaturation conditions, *Computer-Aided Civil and Infrastructure Engineering*, 26, 1-15.
- Ran, B., and Boyce, D.E.,** 1996: A link-based variational inequality formulation of ideal dynamic user-optimal route choice problem, *Transportation Research* 4C(1), 1-12.
- Ran, B., and Shimazaki, T.,** 1989a: A General Model and Algorithm for the Dynamic Traffic Assignment Problems, *Proceedings of the Fifth World Conference on Transport Research*, Yokohama, Japan.
- Ran, B., and Shimazaki, T.,** 1989b: Dynamic User Equilibrium Traffic Assignment for Congested Transportation Networks, *The Fifth World Conference on Transport Research*, Yokohama, Japan.
- Ran, B., Boyce, D.E., and LeBlanc, L.J.,** 1993: A new class of instantaneous dynamic user-optimal traffic assignment models, *Operations Research*, 41, 192-202.
- Ran, B., Hall, R.W., and Boyce, D.E.,** 1996: A link-based variational inequality model for dynamic departure time/route choice, *Transportation Research* 30, 31-46.
- Robertson, D.I.,** 1969: TRANSYT: a traffic network study tool, *RRL Report, LR 253*, Transport and Road Research Laboratory, Crowthorne.
- Robertson, D.I., and Bretherton, R.D.,** 1991: Optimizing networks of traffic signals in real-time: the SCOOT method, *IEEE Trans. on Vehicular Technology*, 40: (1), 11-15.
- Sheffi, Y.,** 1985: Urban Transportation networks: Equilibrium Analysis with Mathematical Programming Methods. *MIT. Prentice-Hall, Inc*, New Jersey.
- Shelokar, P.S., Siarry, P., Jayaraman, V.K., and Kulkarni, B.D.,** 2007: Particle swarm and ant colony algorithms hybridized for improved continuous optimization, *Applied Mathematics and Computation*, 188, 129-142.

- Silcock, J.P., and Sang, A.P.**, 1990: SIGSIGN: a phase-based optimisation program for individual signal-controlled junctions, *Traffic Engineering and Control*, 31(5), 291-298.
- Silcock, J.P.**, 1992: Phase-based optimisation of isolated signal-controlled junctions: sensitivity analysis and treatment of double green periods. Mathematics in Transport Planning and Control. (Griffiths J.F. Ed), *Oxford University Press*. Oxford, 45-57.
- Sims, A.**, 1979: The Sydney coordinated adaptive traffic system, *Proceedings of the ASCE engineering foundation conference on research priorities in computer control of urban traffic systems*, pp. 12-27.
- Smith, M.J.**, 1979: The existence, uniqueness and stability of traffic equilibria, *Transportation Research Part B*, 13: (4), 295-304.
- Smith, M.J.**, 1994: A New Dynamic Traffic Model and the Existence and Calculation of Dynamic User Equilibria on Congested Capacity-constrained Road Networks, *Transportation Research*, 27B, 49-63.
- Sun, W., and Dong, Y.**, 2011: Study of multiscale global optimization based on parameter space partition, *Journal of Global Optimization*, 49, 149-172.
- Sutton, R.S., and Barto, A.G.**, 1998: Reinforcement learning: An introduction. Cambridge, Massachusetts, USA/London, England: *The MIT Press*.
- Suwansirikul, C., Friesz, T. L., and Tobin, R. L.**, 1987: Equilibrium decomposed optimization: a heuristic for the continuous network design problem, *Transportation Science*, 21: (4), 254-263.
- Taylor, N.B.**, 1990: CONTRAM5: An Enhanced Traffic Assignment Model, *TRRL Research Report RR 249*, Transport and Road Research Laboratory, Crowthorne, UK.
- Taylor, N.B.**, 1996: CONTRAM Modeling in the Real-time Context – A Review and Prognosis, *TRRL Project Report PR/TT/088/96*, Transport and Road Research Laboratory, Crowthorne, UK.

- Teklu, F., Sumalee, A., and Watling, D.,** 2007: A genetic algorithm approach for optimizing traffic control signals considering routing, *Computer-Aided Civil and Infrastructure Engineering*, 22, 31-43.
- Thorpe, T.L.,** 1997: Vehicle traffic light control using SARSA, *Master's Project Report*, Computer Science Department, Colorado State University, Colo.
- Toksari, M.D.,** 2009: Minimizing the multimodal functions with Ant Colony Optimization approach, *Expert Systems and Applications*, 36(3), 6030-6035.
- TRB,** 2000: Highway Capacity Manual, Special Report 209, Transportation Research Board, National Research Council, Washington, USA.
- TRB,** 2010: A Primer for Dynamic Traffic Assignment, *ADB30 Transportation Network Modeling Committee*, Transportation Research Board.
- Tutkun, N.,** 2009: Optimization of multimodal continuous functions using a new crossover for the real-coded genetic algorithms, *Expert Systems and Applications*, 36, 8172-8177.
- Vanhulsel, M., Janssens, D., Wets, G., and Vanhoof, K.,** 2009: Simulation of sequential data: An enhanced reinforcement learning approach, *Expert Systems with Applications*, 36, 8032-8039.
- Varia, H. R., and Dhingra, S. L.,** 2004: Dynamic optimal traffic assignment and signal time optimization using genetic algorithms, *Computer-Aided Civil and Infrastructure Engineering*, 19, 260-273.
- Wardrop, J.,** 1952: Some theoretical aspect of road traffic research, *Proceedings Institute Civil Engineers Part II*, pp. 325-378.
- Watkins, C.,** 1989: Learning with Delayed Rewards, Ph.D. Thesis (unpublished), *Cambridge University*, Cambridge, UK, pp. 234.
- Wie, B.W., Tobin, R.L., Friesz, T.L., and Bernstein, D.,** 1995: A Discrete Time, Nested Cost Operator Approach to the Dynamic Network User Equilibrium Problem, *Transportation Science* 29(1), 79-92.

- Wiering, M.A.**, 2000: Learning to control traffic lights with multi-agent reinforcement learning, *First World Congress of the Game Theory Society Games*, Utrecht, Netherlands, Basque Country University and Foundation, Spain.
- Wong, S.C.**, 1996: Group-based optimisation of signal timings using the TRANSYT traffic model, *Transportation Research Part B*, 30(3), 217–244.
- Wu, J., Xu, X., Zhang, P., and Liu, C.**, 2011: A novel multi-agent reinforcement learning approach for job scheduling in Grid computing, *Future Generation Computer Systems*, 27, 430-439.
- Xu, T., Wei, H., and Hu, G.**, 2009: Study on continuous network design problem using simulated annealing and genetic algorithm, *Expert Systems and Applications*, 36, 1322-1328.
- Yang, H., and Yagar, S.**, 1995: Traffic assignment and signal control in saturated road networks, *Transportation research*, 29A(2), 125-139.
- Zhang, H., and Gao, Z.**, 2007: Two-Way Road Network Design Problem With Variable Lanes, *Journal of Systems Science and Systems Engineering*, 16: (1) 50–61.
- Ziliaskopoulos, A.K.**, 2000: A Linear Programming Model for the Single Destination System Optimum Dynamic Traffic Assignment Problem, *Transportation Science* 34(1), 37-49.
- Ziliaskopoulos, A.K., and Waller, S.T.**, 2000: An Internet-based Geographical Information System that Integrates Data, Models and Users for Transportation Applications, *Transportation Research*, 8C(1), 427-444.
- Zixiao, W.**, 2004: Motorized and non-motorized transport systems: Modeling and Optimisation, PhD Thesis, *The Hong Kong Polytechnic University*, Hong Kong.

EKLER

EK 1

F1:

Rosenbrock (2 deęişken)

$$f(x_1, x_2) = (100 * (x_1 - x_2^2)^2) + (1 - x_1)^2$$

- global minimum: $(x_1, x_2) = (1, 1)$, $f(x_1, x_2) = 0$

F2:

(2 deęişken)

$$f(x_1, x_2) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2)$$

- çözüm aralığı: $-1 \leq x_1, x_2 \leq 1$
- 50 yerel minimum
- global minimum: $(x_1, x_2) = (0, 0)$, $f(x_1, x_2) = -2$

F3:

(2 deęişken)

$$f(x_1, x_2) = \frac{(x_1 - 3)^8}{1 + (x_1 - 3)^8} + \frac{(x_2 - 3)^4}{1 + (x_2 - 3)^4}$$

- global minimum: $(x_1, x_2) = (3, 3)$, $f(x_1, x_2) = 0$

F4:

Bohachevsky (2 deęişken)

$$f(x_1, x_2) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$$

- çözüm aralığı: $-100 \leq x_1, x_2 \leq 100$
- birkaç noktada yerel minimum
- global minimum: $(x_1, x_2) = (0, 0)$, $f(x_1, x_2) = 0$

F5:

De Jong (3 değişken)

$$f(x) = \sum_{i=1}^n x_i^2$$

- çözüm aralığı: $-5.12 \leq x_i \leq 5.12; i = 1, 2, \dots, n$
- global minimum : $x = (0, 0, \dots, 0)$, $f(x) = 0$

F6:

Easom (2 değişken)

$$f(x_1, x_2) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$$

- çözüm aralığı: $-100 \leq x_1, x_2 \leq 100$
- birkaç noktada yerel minimum
- global minimum: $(x_1, x_2) = (\pi, \pi)$, $f(x_1, x_2) = -1$

F7:

Goldstein-Price (2 değişken)

$$f(x, y) = \left[1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2) \right] \\ * \left[(30 + (2x_1 - 3x_2)^2) (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2) \right]$$

- çözüm aralığı: $-2 \leq x_1, x_2 \leq 2$
- global minimum : $(x_1, x_2) = (0, -1)$, $f(x_1, x_2) = 3$

F8:

Drop wave (2 değişken)

$$f(x_1, x_2) = -\frac{1 + \cos(12\sqrt{x_1^2 + x_2^2})}{\frac{1}{2}(x_1^2 + x_2^2) + 2}$$

- çözüm aralığı: $-5.12 \leq x_1, x_2 \leq 5.12$
- global minimum : $(x_1, x_2) = (0,0)$, $f(x_1, x_2) = -1$

F9:

Shubert (2 değişken)

$$f(x_1, x_2) = \sum_{i=1}^5 i \cdot \cos((i+1)x_1 + 1) * \sum_{i=1}^5 i \cdot \cos((i+1)x_2 + 1)$$

- çözüm aralığı: $-10 \leq x_1, x_2 \leq 10$
- 760 noktada yerel minimum
- 18 global minimum $f(x_1, x_2) = -186.7309$

F10:

Schwefel (2 değişken)

$$f(x) = \sum_{i=1}^n -x_i * \sin \sqrt{|x_i|}$$

- çözüm aralığı: $-500 \leq x_1, x_2 \leq 500$
- global minimum: $(x_1, x_2) = (420.9687, 420.9687)$, $f(x_1, x_2) = -n * 418.9829$

F11:

Zakharov (2 değişken)

$$f(x) = \sum_{i=1}^n x_i^2 + \left(\sum_{i=1}^n 0.5i x_i \right)^2 + \left(\sum_{i=1}^n 0.5i x_i \right)^4$$

- çözüm aralığı: $-5 \leq x_i \leq 10$, $i = 1, 2, \dots, n$
- birkaç noktada yerel minimum
- global minimum: $x = (0, 0, \dots, 0)$, $f(x) = 0$

F12:

Hartman (3 değişken)

$$f(x) = -\sum_{i=1}^4 c_i \exp \left[-\sum_{j=1}^n \alpha_{ij} (x_j - p_{ij})^2 \right]$$

- çözüm aralığı: $0 \leq x_j \leq 1$, $j = 1, \dots, 3$

- dört noktada yerel minimum
- global minimum : $x = (0.11, 0.555, 0.855)$, $f(x) = -3.8628$

i	α_{i1}	α_{i2}	α_{i3}	c_i	p_{i1}	p_{i2}	p_{i3}
1	3	10	30	1	0.3689	0.1170	0.2673
2	0.1	10	35	1.2	0.4699	0.4387	0.7470
3	3	10	30	3	0.1091	0.8742	0.5547
4	0.1	10	35	3.2	0.03815	0.5743	0.8828

F13:

Hartman (6 değişken)

$$f(x) = -\sum_{i=1}^4 c_i \exp \left[-\sum_{j=1}^n \alpha_{ij} (x_j - p_{ij})^2 \right]$$

- çözüm aralığı: $0 \leq x_j \leq 1$, $j = 1, 2, \dots, 6$
- dört noktada yerel minimum
- global minimum: $x = (0.201, 0.150, 0.477, 0.275, 0.311, 0.657)$, $f(x) = -3.32$

i	α_{i1}	α_{i2}	α_{i3}	α_{i4}	α_{i5}	α_{i6}	c_i
1	10	3	17	3.5	1.7	8	1
2	0.05	10	17	0.1	8	14	1.2
3	3	3.5	1.7	10	17	8	3
4	17	8	0.05	10	0.1	14	3.2

i	p_{i1}	p_{i2}	p_{i3}	p_{i4}	p_{i5}	p_{i6}
1	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650
4	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

F14:

Rastrigin (2 değişken)

$$f(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)]$$

- çözüm aralığı: $-5.12 \leq x_i \leq 5.12$, $i = 1, 2, \dots, n$
- birkaç noktada yerel minimum
- global minimum: $(x_1, x_2) = (0, 0)$, $f(x) = 0$

F15:

Griewank (8 deęişken)

$$f(x) = \sum_{i=1}^n x_i^2 / 4000 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

- çözüm aralığı: $-300 \leq x_i \leq 600, i = 1, 2, \dots, n$
- birkaç noktada yerel minimum
- global minimum: $x = (0, \dots, 0), f(x) = 0$

F16:

Branin (2 deęişken)

$$f(x) = (x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6)^2 + 10(1 - \frac{1}{8\pi}) \cos(x_1) + 10$$

- çözüm aralığı: $-5 \leq x_1 \leq 10, 0 \leq x_2 \leq 15$
- yerel minimum noktası bulunmamaktadır

üç noktada global minimum: $(-\pi, 12.275), (\pi, 2.275), (3\pi, 2.475)$ $f(x) = \frac{5}{4\pi}$

ÖZGEÇMİŞ



Ad Soyad: Cenk OZAN

Doğum Yeri ve Tarihi: Aydın 04.11.1976

**Adres: Pamukkale Üniversitesi Mühendislik Fak. İnşaat Mühendisliği Bölümü
Ulaştırma Anabilim Dalı Kınıklı Yerleşkesi 20070 Denizli**

Lisans Üniversite: Pamukkale Üniversitesi

Yayın Listesi:

- **Ozan, C.**, Haldenbilen, S. and Ceylan, H., “Estimating emissions on vehicular traffic based on projected energy and transport demand on rural roads: Policies for reducing air pollutant emissions and energy consumption”, *Energy Policy*, 39, pp. 2542-2549, 2011.
- **Ozan, C.**, Başkan, Ö., Haldenbilen, S. ve Derici, E., “Trafik Kazalarının Tehlike İndeksi Metodu ile Analizi : Denizli Örneği”, Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi Cilt: 16, Sayı: 3, s. 325-333, 2010.
- **Ozan, C.**, Ceylan, H., Haldenbilen, S. ve Yaşar, A. B., “Kentiçi Otobüs Taşımacılığında Talep Tahmini ve Fiyat Analizleri: Denizli Örneği”, Dokuz Eylül Üniversitesi Mühendislik Bilimleri Dergisi Cilt: 12, Sayı: 1, s. 47-61, Ocak 2010.
- **Ozan, C.**, Başkan, Ö., Haldenbilen, S., Ceylan, H., “Pekiştirmeli Öğrenme Yöntemi İle Dinamik Kullanıcı Dengesi Altında Ulaşım Ağ Tasarımı”, Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu, ASYU 2012, Karadeniz Teknik Üniversitesi, Trabzon, s. 20-24, 3-4 Temmuz 2012.

- Başkan, Ö., **Ozan, C.**, Haldenbilen, S., Ceylan, H., “Modifiye Pekiştirmeli Öğrenme Yöntemi İle Fonksiyon Minimizasyonu”, Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu, ASYU 2012, Karadeniz Teknik Üniversitesi, Trabzon, s. 6-10, 3-4 Temmuz 2012.
- **Ozan, C.**, Başkan, Ö., Haldenbilen, S., Ceylan, H., “Reinforcement learning approach for optimising traffic signal timings at isolated intersections”, International Balkans Conference on Challenges of Civil Engineering, BCCCE 2011, Epoka University, Tirana, ALBANIA, 19-21 May 2011.
- Başkan, Ö., **Ozan, C.**, Haldenbilen, S., Ceylan, H. ve Ceylan, H., “Analyses of CO₂ Emissions Based on Estimated Transport Demand”, 9th International Congress on Advances in Civil Engineering, ACE 2010, Karadeniz Technical University, Trabzon, TURKEY, September 27-30, 2010.
- **Ozan, C.**, Başkan, Ö., Ceylan, H., Haldenbilen, S., “Pekiştirmeli Öğrenme Yöntemi İle Ayrık Sinyalize Kavşak Eniyilenmesi”, 6. Kentsel Altyapı Sempozyumu, TMMOB İnşaat Mühendisleri Odası Antalya Şubesi, Antalya, s. 129-140, 14-15 Ekim 2011.
- Ceylan, H., Ceylan H., Başkan, Ö., **Ozan, C.**, Haldenbilen, S., “Kentiçi Ulaşım Ağı Bağ Maliyet Fonksiyonlarının Geliştirilmesi”, 9. Ulaştırma Kongresi, TMMOB İnşaat Mühendisleri Odası İstanbul Şubesi, İstanbul, s. 127-139, 16-18 Mayıs 2011.
- **Ozan, C.**, “İzmir-Aydın Otoyolu Kapasite Analizi”, İzmir Ulaşım Sempozyumu, İUS, TMMOB İnşaat Mühendisleri Odası İzmir Şubesi, İzmir, s. 603-612, 8-9 Aralık 2009.
- **Ozan, C.**, Ceylan, H., Haldenbilen, S. ve Yaşar, A. B., “Kentiçi Otobüs Taşımacılığında Talep Tahmini ve Fiyat Analizleri: Denizli Örneği”, İzmir Ulaşım Sempozyumu, İUS, TMMOB İnşaat Mühendisleri Odası İzmir Şubesi, İzmir, s. 269-283, 8-9 Aralık 2009.
- Murat, Y. S., Başkan, O., **Ozan, C.**, “Trafik İşaretleme ve Yönetim Tekniklerinin Trafik Güvenliğine Etkilerinin Araştırılması”, 2. Trafik ve Yol Güvenliği Ulusal Kongresi, TRODSA, Ankara, 7-9 Mayıs 2003.