

**BAŐKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ÇOK GEZGİNLİ EN KÜÇÜK GECİKME PROBLEMİ İÇİN  
YENİ KARAR MODELLERİ**

**GÖZDE ÖNDER**

**YÜKSEK LİSANS TEZİ  
2015**

**ÇOK GEZGİNLİ EN KÜÇÜK GECİKME PROBLEMİ İÇİN  
YENİ KARAR MODELLERİ**

**NEW FORMULATIONS FOR MULTIPLE TRAVELER  
MINIMUM LATENCY PROBLEM**

**GÖZDE ÖNDER**

Başkent Üniversitesi  
Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin  
ENDÜSTRİ Mühendisliği Anabilim Dalı İçin Öngördüğü  
YÜKSEK LİSANS TEZİ  
olarak hazırlanmıştır.

2015

“Çok Gezinli En Küçük Gecikme Problemi için Yeni Karar Modelleri” başlıklı bu çalışma, jürimiz tarafından, 31/07/2015 tarihinde, **ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI 'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Başkan

Prof. Dr. Fulya ALTIPARMAK

Tez Danışmanı

Prof. Dr. İmdat KARA

Üye

Yrd. Doç. Dr. Tusan DERYA

**ONAY**

..../08/2015

Prof. Dr. Emin AKATA  
Fen Bilimleri Enstitüsü Müdürü

## **TEŐEKKÜR**

Bu alıőmanın gerekleőmesi sũresince bilgi ve deneyimleri ile bana her zaman destek olan ve yardımlarını esirgemeyen tez danıőmanım deęerli hocam Sayın Prof. Dr. İmdat KARA'ya,

Bu sũrete yaptıęı katkılardan ve desteęinden tũrũ Sayın hocam Yrd. Do. Dr. Tusan Derya'ya,

Tez alıőmam sırasında saęladıęı yardımlardan dolayı Sayın hocam ęr. Gr. Dr. Barıő Keeci'ye teőekkũrlerimi sunarım.

## ÖZ

### **ÇOK GEZGİNLİ EN KÜÇÜK GECİKME PROBLEMİ İÇİN YENİ KARAR MODELLERİ**

Gözde ÖNDER

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Endüstri Mühendisliği Anabilim Dalı

En küçük Gecikme Problemi (EGP) rotalama problemlerinin temelini oluşturan Gezgin Satıcı Probleminin (GSP) bir türü olmaktadır. EGP, bir başlangıç düğümünden başlayarak, tüm düğümlere uğradıktan sonra başlangıç noktasında veya verilen bir düğümde sona eren Hamilton turunu veya yolunu araştırmaktadır. GSP bütün müşterilere uğramak için gerekli olan toplam zamanı en küçük yapmayı amaçlamakta, EGP ise tüm müşterilerin toplam gecikme zamanını en küçük yapmayı amaçlamaktadır. EGP'nin en önemli özel durumu olarak görülen çok gezginli uzantısı için kaynaklardaki yapılan çalışmalar incelendiğinde polinom sayıda ve üstel sayıda kısıta sahip iki farklı matematiksel model olduğu ancak bu modellere bakıldığında kısa sürede çözüme ulaşma açısından verimli olmadığı belirlenmiştir. Bu nedenle yeni karar modellerine ihtiyaç olduğu görülmüştür. Bu çalışma kapsamında ise, temel konu olarak ele alınan çok gezginli EGP için yapılacak çalışmanın altyapısını oluşturması amacıyla öncelikle EGP modelleri kaynaklarda bulunan kıyaslama problemi verileri kullanılarak sayısal analizlere tabi tutulmuştur. İşlem süresi (CPU) ve doğrusal programlama (LP) gevşetme değerleri yönüyle en iyi performans gösteren model belirlenmiştir. Çok gezginli EGP için üç tanesi yeni model bir tanesi kaynaklarda yer alan bir model olmak üzere toplam dört model ele alınıp kaynaklardaki farklı düğüm sayısına sahip kıyaslama problemleri ve gezgin sayıları için çözdürülerek en iyi performans gösteren model önerilmiştir. Yapılan bu karşılaştırmalı analizler sonucunda problem boyutu ve CPU süresi arasındaki ilişki ve gezgin sayısı ile CPU süresi arasındaki ilişki ile ilgili çıkarımlar da elde edilmiştir. Bu çalışmanın en önemli sonucu çok gezginli EGP için yeni bir modelin bilime katkı olarak sunulmasıdır.

**ANAHTAR SÖZCÜKLER:** En Küçük Gecikme Problemi, Gezgin Satıcı Problemi.

**Danışman:** Prof. Dr. İmdat KARA, Başkent Üniversitesi, Endüstri Mühendisliği Bölümü.

## **ABSTRACT**

### **NEW FORMULATIONS FOR MULTIPLE TRAVELER MINIMUM LATENCY PROBLEM**

Gozde ONDER

Baskent University Institute of Science and Engineering

Department of Industrial Engineering

The Minimum Latency Problem (MLP) is a kind of Traveling Salesman Problem (TSP) which is the basis of the routing problems. MLP investigates the Hamilton tour or path, which starts from an initial node, after visiting all nodes, it ends at the starting or any given node. While TSP aims to make the smallest total time required to visit all customers, MLP aims to minimize total delay time of customers. When we review the literature, MLP with multiple traveler which is regarded as the most important exception of MLP, is modeled in two different ways: one with polynomial and the other with exponential number of constraints. However, both models are not efficient in terms of reaching a solution in a short time. Therefore the need for a new decision model is obvious. In this study, firstly MLP models are subjected to several quantitative analysis using available benchmarking problems in the literature. The purpose of this analysis is to create an infrastructure for the multiple traveling MLP. The best performing model is determined in terms of processing time (CPU) and linear programming (LP) relaxation values. Four models, one from literature and three new ones, are solved for benchmark problems with different number of nodes and different number of travelers and the best performing model is selected accordingly. As a result of this comparative analysis, we observe the relationship between problem size and CPU time and also the relationship between the number of travelers and CPU time. The most important result of this study is presented as a contribution to science, a new model for multiple traveling MLP.

**KEY WORDS:** Minimum Latency Problem, Traveling Salesman Problem.

**Supervisor:** Professor İmdat KARA, Baskent University, Department of Industrial Engineering.

# İÇİNDEKİLER LİSTESİ

	<u>Sayfa</u>
ÖZ.....	i
ABSTRACT .....	ii
İÇİNDEKİLER LİSTESİ .....	iii
ŞEKİLLER LİSTESİ.....	v
ÇİZELGELER LİSTESİ.....	vi
EKLER LİSTESİ .....	vii
SİMGELER VE KISALTMALAR LİSTESİ .....	viii
<b>1. GİRİŞ.....</b>	<b>1</b>
<b>2. EN KÜÇÜK GECİKME PROBLEMİ .....</b>	<b>3</b>
2.1. En Küçük Gecikme Probleminin Tanımı ve Çözüm Yaklaşımları .....	3
2.1.1. Problemin tanımı.....	3
2.1.2. Çözüm yaklaşımları .....	5
2.2. En Küçük Gecikme Probleminin Karar Modelleri.....	6
2.2.1. Modeller için ortak gösterimler .....	7
2.2.2. Fischetti et al. modeli .....	7
2.2.3. Kara et al. modeli.....	9
2.2.4. Angel-Bello et al. modeli .....	10
2.3. Modellerin Sayısal Analizleri .....	11
2.3.1. Çözülen problemler.....	11
2.3.2. Yazılım ve donanım .....	12
2.3.3. Sayısal sonuçlar .....	12
2.4. Değerlendirme.....	15
<b>3. ÇOK GEZGİNLİ EN KÜÇÜK GECİKME PROBLEMİ (ÇEGP) .....</b>	<b>16</b>
3.1. ÇEGP'nin Çözüm Yaklaşımları .....	16
3.2. Kaynaklarda Var Olan Modeller .....	17
3.2.1. Kara et al. modeli.....	18
3.2.2. Luo et al. modeli .....	19
3.3. Yeni Karar Modelleri.....	21
3.3.1. Modeller için ortak gösterimler .....	21
3.3.2. Fischetti et al. modelinin uyarlanması.....	22
3.3.3. Angel-Bello et al. modelinin uyarlanması .....	23
3.3.4. Yeni bir model.....	25

3.4. Modellerin Sayısal Analizleri .....	27
3.4.1. Çözülen problemler.....	27
3.4.2. Sayısal sonuçlar .....	27
<b>4. SONUÇ VE ÖNERİLER.....</b>	<b>38</b>
EKLER .....	39
KAYNAKLAR LİSTESİ .....	47



## ŞEKİLLER LİSTESİ

	<u>Sayfa</u>
Şekil 1.1 Örnek bir serim.....	4
Şekil 3.1 Angel-Bello et al. modelinin EGP için örnek çözümü.....	25
Şekil 3.2 YM2 modelinin ÇEGP için örnek çözümü.....	25
Şekil 3.3 29 düğümlü problemler için CPU ve m grafiği.....	35
Şekil 3.4 39 düğümlü problemler için CPU ve m grafiği.....	36
Şekil 3.5 49 düğümlü problemler için CPU ve m grafiği.....	37

## ÇİZELGELER LİSTESİ

	<u>Sayfa</u>
Çizelge 2.1 10 düğümlü problemlerin en iyi değerleri ve CPU süreleri.....	12
Çizelge 2.2 10 düğümlü problemlerin LPR (% sapma) değerleri.....	13
Çizelge 2.3 20 düğümlü problemlerin en iyi değerleri ve CPU süreleri.....	14
Çizelge 3.1 10 düğümlü problemlerin m=1 için en iyi değerleri ve CPU süreleri...	28
Çizelge 3.2 10 düğümlü problemlerin m=1 için LPR (% sapma) değerleri.....	29
Çizelge 3.3 10 düğümlü problemlerin m=2 için en iyi değerleri ve CPU süreleri...	30
Çizelge 3.4 10 düğümlü problemlerin m=2 için LPR (% sapma) değerleri.....	31
Çizelge 3.5 20 düğümlü problemlerin m=1 için en iyi değerler ve CPU süreleri....	32
Çizelge 3.6 20 düğümlü problemlerin m=2 için en iyi değerleri ve CPU süreleri...	33
Çizelge 3.7 29 düğümlü problemler için ortalama CPU süreleri.....	34
Çizelge 3.8 39 düğümlü problemler için ortalama CPU süreleri.....	35
Çizelge 3.9 49 düğümlü problemler için ortalama CPU süreleri.....	36

## EKLER LİSTESİ

### Sayfa

Ek 1	29 düğümlü problemlerin $m=1$ ve $m=2$ alındığında en iyi değerleri ve CPU süreleri.....	39
Ek 2	29 düğümlü problemlerin $m=4$ ve $m=6$ alındığında en iyi değerleri ve CPU süreleri.....	41
Ek 3	39 düğümlü problemlerin $m=4$ , $m=6$ ve $m=8$ alındığında en iyi değerleri ve CPU süreleri.....	43
Ek 4	49 düğümlü problemlerin $m=6$ , $m=8$ ve $m=10$ alındığında en iyi değerleri ve CPU süreleri.....	45

## SİMGELER VE KISALTMALAR LİSTESİ

$G=(V,A)$	tam bir serim
$V$	düğüm kümesi
$A$	ayrıt kümesi
$V_c=\{1,\dots,n\}$	müşteri kümesi
$V_i^+$	i. düğümün öncül kümesi
$V_i^-$	i. düğümün ardıl kümesi
$d_{ij}$	i. düğümden j. düğüme geçiş süresi
$x_{ij}$	(i,j) ayrıtı tur üzerinde ise 1, değilse 0 değerini alan karar değişkeni
$y_{ij}$	(i,j) ayrıtı herhangi bir tur üzerinde ise sondan itibaren turdaki sırasını belirten karar değişkeni
$z_{ij}$	(i,j) ayrıtı k. pozisyonda ise $n-k+1$ , eğer (i,j) ayrıtı kullanılmamış ise 0 değerini alan karar değişkeni
$x_i^{(k)}$	i. düğüm k. pozisyonda ise 1, değilse 0 değerini alan karar değişkeni
$y_{ij}^{(k)}$	i. düğüm k.pozisyonda ve j.düğüm $k+1$ . pozisyonda ise 1, değilse 0 değerini alan karar değişkeni
$x_{ijk}$	k aracı i. düğümden j. düğüme geçmişse 1, diğer durumda 0 değerini alan karar değişkeni
$u_{ik}$	k aracının i. düğüme varış zamanını gösteren pozitif karar değişkeni
$t_{ij}$	i. düğümden j. düğüme geçildiğinde j. düğümün turdaki ziyareti gerçekleşene kadar geçen süre (j. düğüm gecikmesi)
$n$	düğüm sayısı
$m$	gezgin (araç) sayısı
$F$	K araç kümesi
$L$	her aracın i. düğüme varış zamanı üst sınırı

M	yeterince büyük pozitif bir sayı
GSP	Gezgin satıcı problemi
EGP	En küçük gecikme problemi
ÇEGP	Çok gezginli en küçük gecikme problemi
CPU	İşlem süresi
LPR	Doğrusal programlama gevşetme değeri
M1	Çok gezginli en küçük gecikme problemi için Kara et al. modeli
YM1	Fischetti et al. modelinin çok gezginli en küçük gecikme problemi için uyarlaması
YM2	Angel-Bello et al. modelinin çok gezginli en küçük gecikme problemi için uyarlaması
YM3	Çok gezginli en küçük gecikme problemi için yeni bir model

## 1. GİRİŞ

Gezgin satıcı problemi (GSP) rotalama problemlerinin temelini oluşturmaktadır. GSP'nin NP-zor olmasına bağlı olarak rotalama problemleri üzerinde çalışan araştırmacılar çalışmalarını yalın problemlerin özel algoritmalar veya sezgisellerle çözümü üzerinde yoğunlaştırmışlardır. Bu yaklaşımların problemlerin özgün yapılarına ek olarak ortaya çıkabilecek yeni kısıtları da göz önüne almaları her zaman mümkün değildir.

Rotalama problemlerinin kolay kullanımlı matematiksel modellerinin geliştirilmiş olması hem ek kısıtların modele kolay eklenmesine imkan verecek ve en iyi çözüm sonrası analizler yapma imkanı ortaya çıkacak, hem de gerçek hayatta karşılaşılan çoğu problemler paket programlarla doğrudan çözülebilecektir. Bu ihtiyaca cevap verme açısından son yıllarda bilgisayar teknolojilerindeki gelişmeler, donanım ve yazılımdaki ucuzlamalar, uygun matematiksel modellerin geliştirilmesi halinde büyük boyutlu problemlerin bile en iyi çözümlerinin doğrudan bulunabileceği örneklenmiştir. Sözelimi Kara ve Derya 2015 yılında yaptıkları çalışmada 400 düğümlü zaman pencereli gezgin satıcı probleminin en kısa tur süresini, CPLEX 12.5 ile saniye mertebesinde bulmuşlardır [26].

Yukarıda belirtilen ihtiyaçlar ve bilişim teknolojilerinde ortaya çıkan gelişmeler, özel bir rotalama problemi olan en küçük gecikme problemine yeni karar modelleri geliştirmenin yerinde olacağı düşüncesi bu araştırmanın temel motivasyonunu oluşturmuştur.

Bu çerçevede araştırmanın amaçları şöyle sıralanmıştır;

- a) Gecikme probleminin tanımı, gelişimi, uygulamaları ve matematiksel modellerini incelemek.
- b) Gecikme probleminin en önemli uzantısı olarak görülen “çok gezginli en küçük gecikme problemi”ni ele alarak, bu durum için paket programlarla doğrudan kullanılabilir yeni karar modelleri geliştirmek.

İkinci bölümde en küçük gecikme problemi ele alınarak bunun için önerilen karar modelleri incelenmiş, karşılaştırmalı performans analizleri yapılarak sayısal sonuçlara yer verilmiştir. EGP'nin uzantılarına da bu bölümde değinilmiştir. Üçüncü bölümde, önce çok gezginli gecikme problemi tanımlanarak çözüm

yaklaşımları ve kaynaklarda yer alan matematiksel modelleri verilmiştir. Kaynaklardaki modellerin özellikleri ve ikinci bölümdeki sayısal analizler göz önüne alınarak çok gezginli en küçük gecikme problemi için üç yeni karar modeli geliştirilmiştir. Önerilen modellerin karşılaştırmalı sayısal analizleri yapılmış ve sonuçlara da bu bölümde yer verilmiştir. Son bölümde sonuç ve öneriler kapsamında yapılan çalışmaların genel olarak özetlenmesine ve elde edilen sonuçlarla bundan sonra yapılabilecek çalışmalara yer verilmiştir.

## 2. EN KÜÇÜK GECİKME PROBLEMİ

### 2.1. En Küçük Gecikme Probleminin Tanımı ve Çözüm Yaklaşımları

Bu kesimde en küçük gecikme problemi tanımı verilerek kaynaklardaki çalışmalarda geliştirilen çözüm yaklaşımlarına değinilecektir.

#### 2.1.1. Problemin tanımı

En küçük gecikme problemi (minimum latency problem) (EGP) ilk olarak 1967 yılında Conway et al. tarafından tanımlanmıştır [15].

Tek makine için yapılan çizelgeleme problemlerinde işlerin yapılış sırasına göre değişiklik gösteren hazırlık süreleri olduğu durumda yani sıra bağımlı hazırlık süresi olduğunda bu problemler işlerin toplam tamamlanma zamanı veya işlerin toplam akış zamanını en küçüklemeyle ilgili olmaktadır. Bu durumda  $i$ . işten sonra gelen  $j$ . işi yapmak için makinenin gerekli hazırlık zamanı, bir gezginin (aracın) iki düğüm (müşteri) arasındaki seyahat zamanlarına karşılık gelmektedir. İşlerin yapılış zamanları ise müşterilerin taleplerini karşılamak için gereken servis zamanlarına karşılık gelmektedir ve amaç işlerin toplam akış zamanını veya işlerin toplam tamamlanma zamanını en küçükleyecek yapılış sırasını bulmaktır. En küçük gecikme problemi bu tür problemlerden ortaya çıkmıştır [15,25,34].

EGP gezgin satıcı probleminin (GSP) uzantılarından biri olup kaynaklarda tamirci problemi, dağıtıcı problemi ve kümülatif gezgin satıcı problemi olarak da tanımlanmaktadır [38].

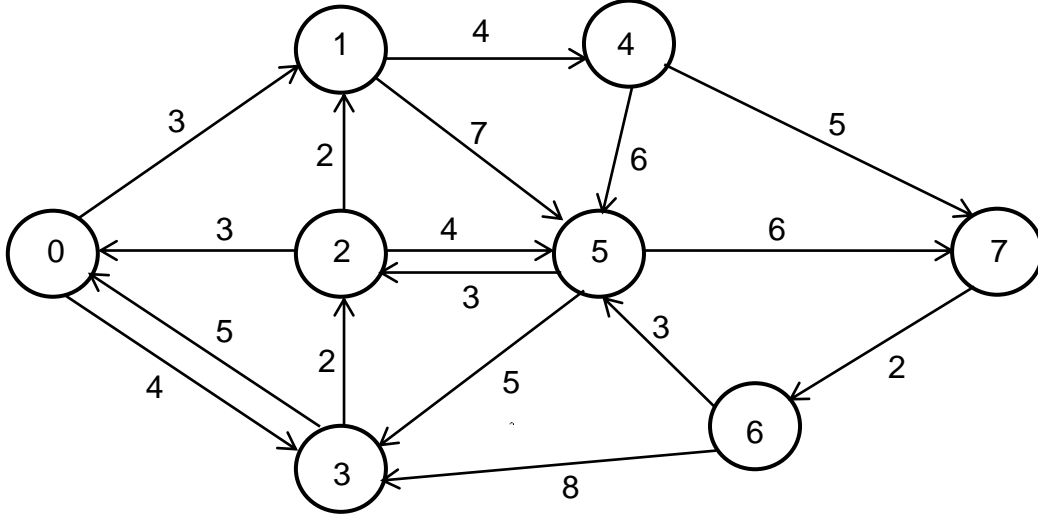
EGP, bir serimde başlangıç düğümünden hareket edip, diğer tüm düğümlere uğrayarak başlangıç düğümünde sona eren ve bütün müşterilerin gecikme zamanları toplamını enküçükleyen Hamilton turunu veya verilen bir düğümde son bulan Hamilton yolunu bulmaktır.

$G=(V,A)$  tam bir serim olmak üzere;

$V=\{0,1,2,\dots,n\}$ ,  $\{0\}$  başlangıç noktası (depo) olmak üzere serimdeki tüm müşterileri ifade eden düğümler kümesi,  $A=\{(i,j)|i,j\in V,i\neq j\}$  ayrıtlar kümesi olsun.

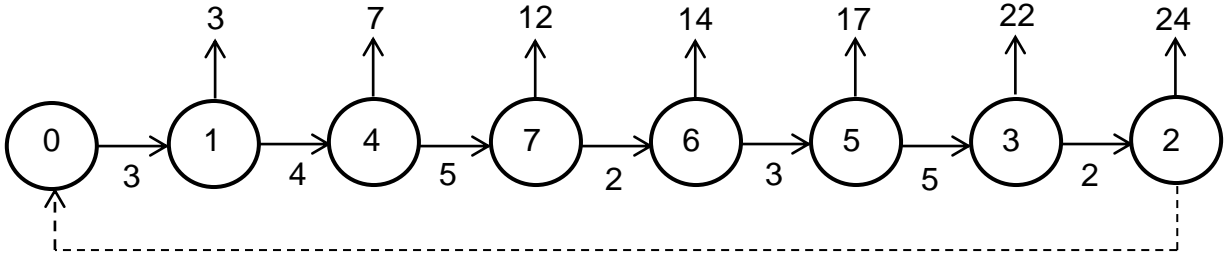


Gecikme, bir müşteri talebinin karşılanıncaya kadar geçen süreyi ifade etmektedir. Gecikme ve toplam gecikme, 0.düğüm başlangıç noktasını göstermek üzere Şekil 1.1'de örneklenmiştir.



Şekil 1.1 Örnek bir serim

Örnek serimde, tüm müşterilerin ziyaret edildiği bir tur şu şekilde olsun;



Bu uygun çözüme göre; 1.müşteri için gecikme= 3 birim zaman, 4.müşteri için gecikme= 3+4=7 birim zaman, 7.müşteri için gecikme= 3+4+5=12 birim zaman, 6.müşteri için gecikme= 3+4+5+2=14 birim zaman, 5.müşteri için gecikme= 3+4+5+2+3=17 birim zamandır. Böylece, müşterilerin toplam gecikme süresi EGP'nin çözümüne göre değeri, gecikmeler toplamı olan 99 birim zaman olacaktır.

Gecikme probleminin birçok uygulama alanı bulunmaktadır. Evlere yapılan her türlü dağıtım servisi uygulamasında [31], acil yardım lojistiğinde [13], esnek üretim sistemlerinde otomatik araçların rotalarının belirlenmesinde [39], bilgisayar ağlarında bilgi toplama işlemlerinde [19], okul servisleriyle öğrencileri toplama ve

dağıtımında [14] ve işlerin ortalama akış zamanını en küçük yapacak şekilde çözeltilmesinde [21] en küçük gecikme probleminin uygulandığı görülmektedir.

EGP bekleme zamanlarını (gecikmeleri) göz önüne alır ve amaç müşterilerin bekleme zamanı toplamını en küçük yapmaktır. GSP'de amaç bütün müşterileri ziyaret etmek için gereken toplam zamanı en küçük yapmaktır. Bu açıdan bakıldığında, EGP müşteri odaklı olup GSP hizmet veren kişi odaklıdır [5]. Bundan dolayı gecikme problemi farklı hizmet sistemi tiplerinin modellenmesinde kullanılmaktadır. Bu iki problem tipi arasındaki temel farklılık amaç fonksiyondan kaynaklanmaktadır.

### **2.1.2. Çözüm yaklaşımları**

EGP, GSP'nin özel bir uzantısı olduğundan, EGP'de NP-zor bir problemdir [40]. GSP ile benzerliklerine rağmen, EGP'nin çözümünün GSP'den daha zor olduğu ifade edilmektedir [22]. Bu nedenle, EGP'nin çözüm yaklaşımları en iyi çözümü garanti eden özel algoritmalar ve sezgiseller üzerinde yoğunlaşmıştır.

Problemin özel algoritmalarla çözümüne ilişkin başlıca yayınlar şunlardır;

Picard and Queyranne tamsayılı doğrusal karar modeli geliştirerek 20 düğümlü problemlere kadar çözüm bulmuşlardır [34]. Lucena alt sınırları belirlemek için lagrange gevşetmesi kullanarak doğrusal olmayan tamsayılı programlama ile 30 düğüme kadar olan problemleri çözmüştür. [29]. Simchi-Levi and Berman dal sınır algoritması kullanarak 20 düğümlü problemlere çözüm bulmuşlardır [39]. Bianco et al. lagrange gevşetmesi ile alt sınırları belirleyip özel iki algoritma geliştirmişlerdir [10]. Fischetti et al. alt sınırları belirlemek için dal sınır algoritması geliştirip tamsayılı programlama kullanarak 60 düğüme kadar problemlere çözüm bulmuşlardır [21]. Eijl karma tamsayılı karar modeli [18], Gouveia and Voss tamsayılı karar modeli geliştirmişlerdir [23]. Wu et al. dinamik programlama ile kısaltma kullanarak 23 düğüme kadar çözüm bulmuşlardır [42]. Sarubbi et al. Picard and Queyranne tarafından zaman bağımlı gezgin satıcı problemi için geliştirilen modeli EGP'ye uyarlamışlardır [37]. Kara et al. tamsayılı doğrusal karar modeli geliştirmişlerdir [27]. Bigras et al. dal sınır algoritmasının yanı sıra tamsayılı programlama önermiştir [11]. Mendez-Diaz et al. karma tamsayılı programlama kullanmıştır [31]. Ezzine et al. iki yeni tamsayılı programlama modeli önermişlerdir

[19]. Abeledo et al. dal kesme ve fiyat yaklaşımı kullanmışlar ve 107 müşteriye kadar çözüm bulmuşlardır [1]; [2]. Angel-Bello et al. iki farklı tamsayılı programlama modeli geliştirmişlerdir [3].

EGP için önerilen belli başlı sezgiseller ise şöyle özetlenebilir;

Blum et al. 1994 yılında ilk sezgisel yaklaşım olan sabit faktör yaklaşım algoritmasını kullanmışlardır [12]. Daha sonra Goemans and Kleinberg yaklaşım oranı için bir algoritma geliştirmişlerdir [22]. Ausiello et al. [7], Arora and Karakostas [6], Archer and Williamson [5], Chaudhuri [18], Nagarajan and Ravi [32] ve Archer and Blasiak, yaklaşım algoritması kullanmışlardır [4]. Salehipour et al. açgözlü rassallaştırılmış uyarlamalı arama prosedürü ile değişken komşuluk arama ve değişken komşuluk azalması sezgisellerini kullanmışlardır [36]. Dewilde et al. kar tabanlı EGP için tabu arama algoritması geliştirmişlerdir [17]. Ngeveu et al. memetik algoritma geliştirmişlerdir [33]. Ribeiro and Laporte kapasite kısıtlı araç rotalama problemi çözümünde iyi performans gösteren komşuluk araması geliştirmişlerdir ancak EGP için denememişlerdir [35].

## **2.2. En Küçük Gecikme Probleminin Karar Modelleri**

Rotalama problemleri için geliştirilen karar modelleri iki grupta toplanabilir. İlk grup, Dantzig-Fulkerson-Johnson alt tur engelleme kısıtlarını esas alan modellerdir [16]. Bu modellerin ortak özelliği, serimdeki düğüm sayısı  $n$  iken, modelin kısıt sayısının üstel ( $a^n$  gibi) artış göstermesidir. Kısıt sayısının problemin boyutuna bağlı olarak üstel artması nedeniyle, böyle modeller bir paket program kullanarak doğrudan çözüm bulmaya elverişli değildir.

İkinci grupta modelin kısıt sayısının,  $n$ 'nin polinom fonksiyonu ( $an^b$  vb.) olarak artış gösteren modeller yer almaktadır. Bu tür modeller polinom büyüklükte karar modelleridir. Yakın bir geçmişte, Angel-Bello et al. [3] EGP'nin polinom büyüklükte karar modellerini karşılaştırmalı sayısal analize tabi tutmuşlar, yanısıra da iki yeni model önermişlerdir. Angel-Bello et al. tarafından önerilen ve model A olarak isimlendirilen modelin, EGP için önerilen önceki tüm modellerden hem işlem süresi (CPU) hem de doğrusal programlama gevşetme değeri olarak daha iyi sonuç verdiğini sayısal olarak göstermişlerdir.

İzleyen sayfalarda Angel-Bello et al. tarafından karşılaştırmalarda ele alınmayan iki polinom büyüklükte model ile, araştırmacıların model A olarak isimlendirdikleri model ele alınıp, sayısal analize tabi tutularak, bu modellerin özel durumlara uyarlanabilirliği tartışılacaktır.

Aşağıda, önce EGP'nin tez kapsamında ele alınan 3 farklı karar modeline değinilecek, daha sonra sayısal analizlere yer verilecektir.

### 2.2.1. Modeller için ortak gösterimler

EGP için yapılan birçok modelleme çalışmasında problem başlangıç düğüme dönen hamilton turu [21; 27] veya bilinen bir düğüme son bulan hamilton yolu [3] şeklinde ele alınmıştır. Bu çalışmada hamilton yolu esas alınacaktır.

Ele alınan karar modelleri için yazarlar farklı tanımlamalar ve gösterimler kullanmışlardır. Bu modeller için ortak bir gösterim yapılabilmesi ve bu sayede daha kolay bir anlatım sağlanabilmesi amacıyla dizin kümeleri, parametreler ve ortak karar değişkenleri için aşağıdaki tanımlamalara yer verilmiştir. Modellerde tanımlanan özel karar değişkenleri ayrıca belirtilecektir.

$G=(V,A)$  tam bir serimi ifade edip;

$V=\{0,1,2,\dots,n\}$  düğüm (müşteri) kümesi,

$A=\{(i,j) \mid i,j \in V, i \neq j\}$  ayrıt kümesi,

$\{0\}$  başlangıç düğüümü,  $\{n\}$  bitiş düğüümü göstermektedir.

Problemlerde ortak olan parametre;

$d_{ij} = i.$  düğümden  $j.$  düğüme geçiş süresi olarak ifade edilmektedir.

Ortak kullanılan karar değişkeni ise;

$$x_{ij} = \begin{cases} 1, & \text{eğer } (i,j) \text{ ayrıtı tur üzerinde ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

olarak gösterilmektedir.

### 2.2.2. Fischetti et al. modeli

Fischetti et al. modelinde özel olarak tanımlanan karar değişkeni;

$$z_{ij} = \begin{cases} n-k+1, & \text{eğer } (i,j) \text{ ayrıtı } k. \text{ pozisyonda ise} \\ 0, & \text{eğer } (i,j) \text{ ayrıtı kullanılmamış ise} \end{cases}$$

olup, model aşağıdaki gibidir.

$$\sum_{j=0}^n x_{ij} = 1 \quad i=0,1,2,\dots,n \quad (2.1)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad j=0,1,2,\dots,n \quad (2.2)$$

$$\sum_{i=0}^n z_{ik} - \sum_{j=0}^n z_{kj} = \begin{cases} -n, & (k=0) \\ 1, & (k=1,2,\dots,n) \end{cases} \quad (2.3)$$

$$z_{ij} \leq r_{ij} x_{ij} \quad i=0,1,2,\dots,n, \quad j=0,1,2,\dots,n \quad (2.4)$$

$$r_{ij} = \begin{cases} 1, & \text{eğer } j=0 \text{ ise} \\ n, & \text{eğer } i=0 \text{ ise} \\ n-1, & \text{d.d.} \end{cases} \quad (2.5)$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in A \quad (2.6)$$

$$z_{ij} \geq 0 \text{ ve tamsayı} \quad (2.7)$$

kısıtları altında

$$Enkx_0 = \sum_{i=0}^n \sum_{j=0}^n d_{ij} z_{ij} \quad (2.8)$$

2.1 kısıtı ve 2.2 kısıtı her düğüme bir defa uğranmasını sağlayan atama kısıtlarıdır. 2.3 kısıtı düğümler arası akışı sağlayan ve yol üzerindeki düğümlere sırasına göre sondan başa doğru değer vererek, alt turları da engelleyen kısıttır. 2.4 numaralı kısıt eğer  $(i,j)$  ayrıtı kullanılmışsa  $z_{ij}$  değişkeninin pozitif değer almasını, kullanılmamış ise bu değişkenin sıfır olmasını sağlayan kısıttır. 2.5 kısıtı ise  $i$  ve  $j$  değerlerine göre  $r_{ij}$  değişkenine değer atamayı sağlamaktadır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı,  $n^2$  kadar negatif olamayan karar değişkenleri bulunmaktadır. Modeldeki kısıt sayısı ise  $n^2+4n$  kadardır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir.

### 2.2.3. Kara et al. modeli

Kara et al. modelinde özel olarak tanımlanan karar değişkeni;

$y_{ij} = (i,j)$  ayrıtı herhangi bir tur üzerinde ise sondan itibaren turdaki sırası

olup, model aşağıdaki gibidir.

$$\sum_{i=1}^n x_{0i} = 1 \quad (2.9)$$

$$\sum_{i=1}^n x_{i0} = 1 \quad (2.10)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad j=1,2,\dots,n \quad (2.11)$$

$$\sum_{j=0}^n x_{ij} = 1 \quad i=1,2,\dots,n \quad (2.12)$$

$$\sum_{j=0}^n y_{ji} - \sum_{j=0}^n y_{ij} = 1 \quad i=1,2,\dots,n \quad (2.13)$$

$$y_{ij} \leq nx_{ij} \quad \forall (i, j) \in A \quad (2.14)$$

$$y_{ij} \geq x_{ij} \quad j \neq 0 \quad (2.15)$$

$$x_{ij} \in \{0,1\} \quad \forall (i, j) \in A \quad (2.16)$$

kısıtları altında

$$Enkx_0 = \sum_{i=0}^n \sum_{j=0}^n d_{ij} y_{ij} \quad (2.17)$$

2.9 kısıtı başlangıç düğümünden çıkışın olmasını sağlayan ve 2.10 kısıtı başlangıç noktasına girişin olmasını sağlayan kısıtlardır. 2.11 ve 2.12 numaralı kısıtlar her düğüme bir defa uğranmasını sağlayan atama kısıtlarıdır. 2.13 kısıtı düğümler arası akışı sağlayan ve yol üzerindeki düğümlere sırasına göre sondan başa doğru

değer vererek, alt turları da engelleyen kısıttır. 2.14 ve 2.15 numaralı kısıtlar ise eğer  $(i,j)$  ayrıtı kullanılmışsa  $y_{ij}$  değişkeninin pozitif değer almasını, kullanılmamış ise bu değişkenin sıfır olmasını sağlamaktadır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı,  $n^2$  kadar negatif olamayan karar değişkenleri bulunmaktadır. Modeldeki kısıt sayısı ise  $2n^2+4n+2$  kadardır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir.

#### 2.2.4. Angel-Bello et al. modeli

Angel-Bello et al. yaptıkları çalışmada iki yeni model önermiş ve bunlardan model A olarak isimlendirdiklerinin çok daha üstün olduğunu göstermişlerdir. Bu nedenle aşağıda model A'ya yer verilecektir.

Angel-Bello et al. modelinde diğer iki model ile ortak tanımlanan karar değişkeni bulunmamaktadır. Bu modelde özel olarak tanımlanan karar değişkenleri;

$$x_i^{(k)} = \begin{cases} 1, & \text{eğer } i.\text{düğüm } k.\text{pozisyonda ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

$$y_{ij}^{(k)} = \begin{cases} 1, & \text{eğer } i.\text{düğüm } k.\text{pozisyonda ve } j.\text{düğüm } k+1. \text{ pozisyonda ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

olup, model aşağıdaki gibidir.

$$\sum_{k=1}^n x_i^{(k)} = 1 \quad i=1,2,\dots,n \quad (2.18)$$

$$\sum_{i=1}^n x_i^{(k)} = 1 \quad k=1,2,\dots,n \quad (2.19)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^n y_{ij}^{(k)} = x_i^{(k)} \quad i=1,2,\dots,n, \quad k=1,2,\dots,n-1 \quad (2.20)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^n y_{ji}^{(k)} = x_i^{(k+1)} \quad i=1,2,\dots,n, \quad k=1,2,\dots,n-1 \quad (2.21)$$

$$x_i^{(k)} \in \{0,1\} \quad i=1,2,\dots,n, \quad k=1,2,\dots,n \quad (2.22)$$

$$y_{ij}^{(k)} \geq 0 \quad i=1,2,\dots,n, \quad j=1,2,\dots,n, \quad k=1,2,\dots,n-1, \quad j \neq i \quad (2.23)$$

kısıtları altında

$$Enkx_0 = n \sum_{i=1}^n d_{0i} x_i^{(1)} + \sum_{k=1}^{n-1} \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n (n-k) d_{ij} y_{ij}^{(k)} \quad (2.24)$$

2.18 kısıtı her düğümün bir pozisyonda bulunmasını, 2.19 kısıtı ise her pozisyonda bir düğümün bulunmasını sağlayan atama kısıtlarıdır. 2.20 numaralı kısıt k. pozisyondan sadece bir çıkışın olmasını sağlayan kısıttır. 2.21 kısıtı ise k+1. pozisyona yalnızca bir girişin olmasını sağlamaktadır. Burada  $x_i^{(k)}$  değişkeninin  $\{0,1\}$  tamsayı değer alması,  $y_{ij}^{(k)}$  değişkeninin de  $\{0,1\}$  tamsayı değer almasını sağlamaktadır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı,  $n^3$  kadar negatif olmayan karar değişkenleri bulunmaktadır. Modeldeki kısıt sayısı ise  $2n^2$  kadardır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir.

### 2.3. Modellerin Sayısal Analizleri

Ele alınan bu 3 model üzerinde deneysel analizler yapılarak CPU süreleri ve doğrusal programlama gevşetme (LPR) değerleri izleyen kesimlerde verilmiştir.

#### 2.3.1. Çözülen problemler

Modelleri karşılaştırmak için kaynaklarda yer alan kıyaslama problemleri verileri araştırılmış ve Salehipour et al. tarafından oluşturulmuş kıyaslama problemi verileri seçilmiştir [31]. Bu veriler 10, 20, 50, 100, 150, 200 ve 500 düğüm olmak üzere 7 farklı problem boyutu için oluşturulmuş ve her biri için 20 adet farklı simetrik problem verisi hazırlanmıştır. Veriler oluşturulurken 0 ile 100 aralığında düzgün (uniform) dağılım kullanılarak her düğüm için x ve y koordinatları oluşturulmuştur. (500 düğümlü problemler için 0 ile 500 aralığından veriler üretilmiştir.) Tüm düğümler arası uzaklıklar Öklid uzaklığı temel alınarak oluşturulmuştur. Hesaplanan uzaklık değerleri en yakın alt sınıra yuvarlanarak uzaklık matrisleri oluşturulmuştur. Üst süre sınırı 7200 saniye olarak alınmıştır. Ortalama CPU süreleri hesaplanırken üst sınır değerleri hesaba katılmamıştır.



### 2.3.2. Yazılım ve donanım

Ele alınan problem boyutlarına göre oluşturulan tüm modellerin CPLEX 12.6.0.0 paket programı kullanılarak Intel Core Quad CPU 2.66 GHz ve 2 GB RAM özellikli bilgisayar ortamında çözümü araştırılmıştır.

Oluşturulan modeller kıyaslama problemleri için çözdürülmekte, modellerin performansları, çözüm süreleri ve LPR değerleri kaydedilmektedir. Modellerin çözüm süresinden dolayı kıyaslama problemlerinden bu aşamada yalnızca 10 düğümlü ve 20 düğümlü olan problemler ele alınmıştır. Ele alınan 3 karar modeli için çözdürülmüş ve istenen veriler elde edilmiştir.

### 2.3.3. Sayısal sonuçlar

Bu kesimde ele alınan üç model için 10 düğümlü ve 20 düğümlü problemlerin çözümü sonucunda elde edilen CPU süresi ve hesaplanan LPR değerleri (yüzde olarak sapma) çizelge 2.1, çizelge 2.2, çizelge 2.3 ve çizelge 2.4'te verilmiştir.

10 düğümlü problemlerin çözümü sonucunda elde edilen en iyi çözümler ile CPU süreleri ve ortalamaları çizelge 2.1'de gösterilmiştir.

**Çizelge 2.1** 10 düğümlü problemlerin en iyi değerleri ve CPU süreleri

Problem	En iyi Değer	Fischetti et al.	Kara et al.	Angel-Bello et al.
		CPU (sn)	CPU (sn)	CPU (sn)
1	1303	0.65	1.97	0.19
2	1517	0.71	1.48	0.20
3	1233	0.69	0.53	0.14
4	1386	0.88	2.34	0.20
5	978	0.66	0.51	0.16
6	1477	0.94	0.96	0.03
7	1163	0.79	0.64	0.03
8	1234	0.90	3.81	0.16
9	1402	0.63	1.31	0.22
10	1388	6.47	8.28	0.14
11	1405	0.93	1.87	0.27
12	1150	0.44	0.54	0.17
13	1531	4.15	5.08	0.19
14	1219	0.66	2.04	0.16
15	1087	0.77	2.71	0.16
16	1264	8.19	8.17	0.21
17	1058	0.39	0.26	0.22

**Çizelge 2.1** 10 düğümlü problemlerin en iyi değerleri ve CPU süreleri (devamı)

18	1083	0.70	1.15	0.25
19	1394	0.88	5.84	0.18
20	951	0.44	1.36	0.10
<b>ortalama</b>	<b>-</b>	<b>1.54</b>	<b>2.54</b>	<b>0.17</b>
<b>std. sapma</b>		<b>2.15</b>	<b>2.45</b>	<b>0.06</b>

10 düğüm için ortalama CPU değerlerine bakıldığında Angel-Bello et al. tarafından geliştirilen modelin diğerlerine göre daha hızlı çözüme ulaştığı ve bundan dolayı performansının daha iyi olduğu görülmüştür. Ayrıca değişkenliği ifade eden standart sapma değerleri yönüyle de karşılaştırıldığında bu modelin standart sapmasının daha düşük olduğu görülmektedir.

10 düğümlü problemlerin çözümü sonucunda hesaplanan LPR değerleri (yüzde olarak sapma) çizelge 2.2’de verilmiştir.

**Çizelge 2.2** 10 düğümlü problemlerin LPR (% sapma) değerleri

Problem	Fischetti et al.	Kara et al.	Angel-Bello et al.
	LPR (% sapma)	LPR (% sapma)	LPR (% sapma)
1	0,464	0,351	0,048
2	0,605	0,462	0,050
3	0,512	0,367	0,054
4	0,617	0,473	0,125
5	0,558	0,419	0,083
6	0,572	0,423	0,000
7	0,537	0,370	0,000
8	0,701	0,549	0,086
9	0,668	0,517	0,101
10	0,717	0,584	0,172
11	0,586	0,458	0,088
12	0,509	0,366	0,076
13	0,707	0,577	0,179
14	0,600	0,464	0,108
15	0,506	0,417	0,136
16	0,699	0,547	0,060
17	0,478	0,348	0,098
18	0,604	0,497	0,237
19	0,555	0,433	0,107
20	0,421	0,263	0,002
<b>ortalama</b>	<b>0,581</b>	<b>0,444</b>	<b>0,090</b>

LPR % sapma deęerleri hesaplanırken (Eniyi deęer - LPR) / (Eniyi deęer) formülasyonu kullanılmıřtır.

10 düęüm için hesaplanan LPR % sapma deęerlerine bakıldıęında Angel-Bello et al. tarafından geliřtirilen modelin dięerlerine göre sapması daha düřük olarak bulunmuřtur. Bu aęıdan da dięer modellere üstünlük saęladıęı görölmektedir.

Ele alınan bu üç model 20 düęümlü 20 adet problem için çözdürüldüęünde Fischetti et al. ve Kara et al. tarafından geliřtirilen modellerin üst sınır olarak verilen 7200 saniye ierisinde en iyi çözüme ulaşamadıęı ancak Angel-Bello et al. tarafından geliřtirilen modelin bu sürenin oldukça altında ortalama bir süre ierisinde 20 düęümlü ele alınan 20 problem için en iyi çözüme ulařtıęı görölmüřtür. En iyi deęerler ve çözümler Çizelge 2.3'te verilmiřtir.

**Çizelge 2.3** 20 düęümlü problemlerin en iyi deęerleri ve CPU süreleri

Problem	En iyi Deęer	Angel-Bello et al.
		CPU (sn)
1	3175	4.51
2	3248	8.77
3	3570	2.66
4	2983	13.83
5	3248	28.22
6	3328	12.86
7	2809	8.50
8	3461	22.20
9	3475	22.19
10	3359	7.11
11	2916	6.48
12	3314	4.16
13	3412	5.97
14	3297	4.84
15	2862	3.92
16	3433	2.89
17	2913	2.63
18	3124	11.96
19	3299	8.69
20	2796	1.81
<b>ortalama</b>	-	<b>9.21</b>
<b>std. sapma</b>		<b>7.40</b>

20 düğümlü problemlerin çözümü sonucunda belirlenen üst süre sınırı içerisinde yalnızca bir model çözüme ulaştığından 20 düğümlü problemler için LPR değerleri (yüzde olarak sapma) karşılaştırması yapılmamıştır.

#### **2.4. Değerlendirme**

CPU ve LP gevşetme değerleri açısından Angel-Bello et al. tarafından geliştirilen model A üstünlük göstermiştir. Bu modelin gerçek hayatta karşılaşılan küçük boyutlu problemlerin çözümünde doğrudan kullanılabileceği görülmektedir.

Gerçek hayat problemlerinde zaman penceresi, öncelikli müşteriler, çok gezgin, gezginlerin zaman veya kapasite kısıtı vb. özel kısıtlarla karşılaştırılması kaçınılmazdır ve böyle durumlardaki performans incelemeye değer görülmektedir.

Gerçek hayatta karşılaşılan özel kısıtlar açısından çok gezginli durum daha gerçekçi olmakta ve daha yaygın görülmektedir. Bu nedenle bu aşamaya kadar yapılan çalışmalar ve sayısal analizler tez kapsamında temel konu olarak ele alınacak olan çok gezginli en küçük gecikme problemi için altyapıyı oluşturmaktadır. İzleyen bölümde çok gezginli EGP ele alınarak var olan karar modelleri, uygulama yerleri, çözüm yaklaşımları incelenecek ve yeni karar modelleri geliştirilecektir.

### 3. ÇOK GEZGİNLİ EN KÜÇÜK GECİKME PROBLEMİ (ÇEGP)

Çok gezginli gecikme problemi her biri aynı depodan başlayan, her bir düğümün yalnız bir tur üzerinde olması koşuluyla ve tüm düğümleri (müşterileri) ziyaret ederek müşterilerin toplam gecikmesini en küçük yapan  $m$  adet turdan veya  $m$  adet yoldan oluşan en küçük gecikme problemi olarak bilinmektedir. Amaç müşterilerin toplamda en küçük gecikmesini sağlayan  $m$  adet gezginin (turun) bulunmasıdır [24]. Bu problemde  $m=1$  olarak alındığında yani tek gezgin için çözüm araştırıldığı zaman ise en küçük gecikme problemi haline dönüşmektedir. Eğer her gezgin başladığı noktaya geri dönüyorsa bu durumda tur oluşturmakta, eğer her gezgin herhangi bilinen bir düğümde son buluyorsa bu durumda yol oluşturmaktadır. ÇEGP için yapılan modelleme çalışmalarında problem turlar veya yollar şeklinde ele alınmıştır. Bu çalışmada yolları araştıran modellemeler esas alınacaktır. ÇEGP probleminin uygulama alanlarına bakıldığında EGP probleminin uygulama alanlarıyla çok benzer olduğu görülmektedir. ÇEGP problemi pizza dağıtımı, otomatik araçların esnek üretim sisteminde rotalanması, makinelerin işlerin tamamlanması için geçen ortalama akış zamanını en küçükleyecek şekilde çizelgelenmesi gibi alanlarda kullanılabilir [21].

#### 3.1. ÇEGP'nin Çözüm Yaklaşımları

ÇEGP çok gezginli gezgin satıcı probleminin bir türü olarak ortaya çıkmıştır [41; 8]. ÇEGP üzerine çalışılan makaleler oldukça sınırlıdır. İlk çalışmanın 2007 yılında Fakcharoenphol et al. tarafından yapılmış olduğu görülmektedir. Bu makalede ÇEGP için  $k$  en küçük yayılan ağaç problemi için polinom zamanlı  $8.497\gamma$  yaklaşım algoritması kullanılmıştır [20].

Bennett and Gazis her otobüsün sabit kapasitesinin olduğu ve her toplama noktası için öğrenci sayısını gösteren talebin olduğu, öğrencileri toplama noktasından alıp okula götüren bir okul servis filosunun sekinin yapıldığı okul servisi dağıtım problemini uygulamışlardır. Amaç toplam otobüs seyahat zamanını ve toplam öğrencilerin seyahat zamanını en küçük yapmaktır [9].

Li and Fu ise Hong Kong anaokulu okul servis rotasının belirlenmesi üzerine bir çalışma yapmışlardır. Bu çalışmada 4 farklı amaçtan oluşan kombinatoriyal eniyileme yapılmıştır. 1. öncelikli amaç gerekli toplam otobüs sayısını en küçük

yapmak, 2. öncelikli amaç öğrencilerin toplam seyahat zamanını en küçük yapmak, 3. öncelikli amaç toplam otobüs seyahat zamanını en küçük yapmak ve 4. öncelikli amaç ise otobüsler arası seyahat zamanını ve yükü dengelemektir [28].

Kara et al. yaptıkları çalışmada EGP için geliştirdikleri tamsayılı karar modelini çok gezginli durum için de düzenlemişlerdir [27]. Tez kapsamında bu model kaynaklarda var olan bir model olarak izleyen aşamalarda kıyaslaması yapılacak olan diğer yeni modellerle birlikte ele alınacaktır.

Luo et al. yeni bir dal-fiyat-kesme algoritması geliştirmişler ve iki farklı sütun oluşturma prosedürü kullanmışlardır. Birincisi için tabu arama ve azalan durum uzayı gevşetmesi ile sınırlı iki yönlü etiket düzenleme algoritması (BBL) kullanarak ikincisi içinse durum uzayı gevşetmesi ile yalnızca BBL kullanarak sütun oluşturmalarını gerçekleştirmişlerdir. Problem çözümünde veri olarak *brd14051*, *d15112*, *d18512*, *fnl4461*, *nrv1379* ve *pr1002* olarak isimlendirilen altı TSP örneğinden türetilmiş verileri kullanmışlardır. Her bir örnek için rassal olarak seçilen, düğüm sayısı  $n=29$ ,  $39$  ve  $49$  olan ve bir düğüm depo olan  $n+1$  düğümlü 10 adet alt küme bulunmaktadır.  $n=29$  için  $m=6$ ,  $n=39$  için  $m=8$  ve  $n=49$  için  $m=10$  alınarak çözüm elde edilmiştir [30].

Kaynaklar incelendiğinde matematiksel model olarak;

- Kara et al. EGP için önerdikleri modelin ÇEGP için de uygulanabileceği,
- Luo et al. tarafından önerilen karar modeli

olduğu anlaşılmıştır.

### **3.2. Kaynaklarda Var Olan Modeller**

Bu bölümde ÇEGP'nin kaynaklarda bulunan iki farklı modeline yer verilecektir ve bu modeller incelenecektir.

Ele alınan karar modelleri için yazarlar farklı tanımlamalar ve gösterimler kullanmışlardır. Dizin kümeleri, parametreler ve karar değişkenleri incelendiğinde bu modeller için ortak bir gösterimin olmadığı, bu nedenle kullanılan tüm dizin kümeleri, parametreler ve karar değişkenleri modeller kapsamında ayrı ayrı belirtilecektir.

### 3.2.1. Kara et al. modeli

$G=(V,A)$  tam bir serimi olup,

$V=\{0,1,2,\dots,n\}$  düğüm (müşteri) kümesini,

$A=\{(i,j) \mid i,j \in V, i \neq j\}$  ayrıt kümesini,

$m$ = gezgin (araç) sayısını,

$\{0\}$  başlangıç düğümünü,  $\{n\}$  bitiş düğümünü göstermektedir.

$d_{ij}$ =  $i$ inci düğümden  $j$ inci düğüme geçiş süresi olarak ifade edilmektedir.

$$x_{ij} = \begin{cases} 1, & \text{eğer } (i,j) \text{ ayrıtı tur üzerinde ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

$y_{ij} = (i,j)$  ayrıtı herhangi bir tur üzerinde ise sondan itibaren turdaki sırası

olup, model aşağıdaki gibidir.

$$\sum_{i=1}^n x_{0i} = m \quad (3.1)$$

$$\sum_{i=1}^n x_{i0} = m \quad (3.2)$$

$$\sum_{j=0}^n x_{ij} = 1 \quad i=1,2,\dots,n \quad (3.3)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad j=1,2,\dots,n \quad (3.4)$$

$$\sum_{j=0}^n y_{ji} - \sum_{j=0}^n y_{ij} = 1 \quad i=1,2,\dots,n \quad (3.5)$$

$$y_{ij} \leq (n-m)x_{ij} \quad \forall (i,j) \in A \quad (3.6)$$

$$y_{ij} \geq x_{ij} \quad j \neq 0 \quad (3.7)$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in A \quad (3.8)$$

kısıtları altında

$$Enkx_0 = \sum_{i=0}^n \sum_{j=0}^n d_{ij} y_{ij} \quad (3.9)$$

3.1 kısıtı başlangıç düğümünden  $m$  sayıda çıkışın olmasını ve 3.2 kısıtı başlangıç noktasına  $m$  sayıda girişin olmasını sağlamaktadır. 3.3 ve 3.4 numaralı kısıtlar düğümler arasındaki bağlantıyı ve geçişi sağlayan atama kısıtlarıdır. 3.5 kısıtı düğümler arası akışı sağlayan ve düğümlerin buldukları sıraya göre sondan itibaren değer almasını sağlayan kısıttır. 3.6 ve 3.7 numaralı kısıtlar değişkenler arasındaki ilişkiyi sağlamaktadır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı ve  $n^2$  kadar negatif olmayan karar değişkenleri ve  $2n^2+4n+2$  kadar kısıt bulunmaktadır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir. Bundan dolayı izleyen sayfalarda yer alacak sayısal karşılaştırmalarda kullanılacak ve  $M1$  olarak isimlendirilecektir.

### 3.2.2. Luo et al. modeli

$G=(V,A)$  tam bir serimi olup,

$V=\{0,1,\dots,n,n+1\}$  düğümler kümesini,

$A=\{(i,j): i,j \in V, i \neq j, i \neq n+1, j \neq 0\}$  ayrıt kümesini,

0.düğüm ve  $n+1$ . düğüm başlangıç noktasını ifade etmektedir.

$V_c=\{1,\dots,n\}$   $n$  adet müşteri kümesini göstermektedir.

$d_{ij}$   $i$ . düğümden  $j$ . düğüme geçiş süresini ifade etmektedir.

$V^+(i)=\{j \in V | (i,j) \in E\}$  ve  $V^-(i)=\{j \in V | (j,i) \in E\}$  kümesi  $i$ . düğümün öncül ve ardıl kümesini ifade etmektedir.

$F=K$  araç kümesi,  $L$ =her aracın  $i$ . düğüme varış zamanı üst sınırı ve  $M$ = yeterince büyük pozitif bir sayı olmaktadır.

$$x_{ijk} = \begin{cases} 1, & \text{eğer } k \text{ aracı } i. \text{ düğümden } j. \text{ düğüme geçmişse} \\ 0, & \text{diğer durumlarda} \end{cases}$$

$u_{ik}$  =  $k$  aracının  $i$ . düğüme varış zamanını gösteren pozitif değişken olup, model aşağıdaki gibidir.



$$\sum_{k \in F} \sum_{j \in V^+(i)} x_{i,j,k} = 1 \quad \forall i \in V_c \quad (3.10)$$

$$\sum_{j \in V^+(0)} x_{0,j,k} = 1 \quad \forall k \in F \quad (3.11)$$

$$\sum_{j \in V^+(i)} x_{i,j,k} = \sum_{j \in V^-(i)} x_{j,i,k} \quad \forall k \in F, \quad i \in V_c \quad (3.12)$$

$$\sum_{i \in V^-(n+1)} x_{i,n+1,k} = 1 \quad \forall k \in F \quad (3.13)$$

$$u_{j,k} \geq u_{i,k} + d_{i,j} + M(x_{i,j,k} - 1) \quad \forall k \in F, \quad (i, j) \in E \quad (3.14)$$

$$\sum_{k \in F} \sum_{(i,j) \in E: i \in V/S, j \in S} x_{i,j,k} \geq 1 \quad \forall S \subset V_c \quad (3.15)$$

$$0 \leq u_{i,k} \leq L \quad \forall k \in F, \quad i \in V \quad (3.16)$$

$$x_{i,j,k} \in \{0,1\} \quad \forall k \in F, \quad (i, j) \in E \quad (3.17)$$

kısıtları altında

$$z = \min \sum_{k \in F} \sum_{i \in V_c} u_{i,k} \quad (3.18)$$

3.10 kısıtı her düğümün (müşterinin) bir kere ziyaret edilmesini sağlamaktadır. 3.11 ve 3.13 numaralı kısıtlar her aracın 0.düğümünden başlayıp  $n+1$ . düğümde bitmesini sağlayan kısıtlardır. 3.12 kısıtı akış korunumunu sağlamaktadır. 3.14 kısıtı aynı araç tarafından ziyaret edilen ardışık iki düğüm arası varış zamanlarıyla ilişkili kısıttır. 3.15 kısıtı alt turların oluşumunu engellemektedir. 3.16 numaralı kısıt ise her varış zamanını  $[0,L]$  aralığıyla sınırlandırmaktadır.

Bu modelde  $Kn^2+2Kn+K$  kadar karar değişkenleri bulunmaktadır. Kısıt sayısına bakıldığında 3.15 numaralı kısıtın içeriğinden dolayı alt küme sayısına bağlı olarak modelin üstel sayıda kısıtının olmasına neden olmaktadır.

Kaynaklarda var olan bu iki model incelendiğinde bu modellerden ilkinin polinom sayıda; ikincisinin üstel sayıda kısıtları vardır. İkinci modelin doğrudan

kodlanabilirlik ve çözüme ulaşma açısından kullanılabilirliği yetersizdir. O halde çok gezginli EGP için yeni karar modellerine ihtiyaç vardır. Bu çerçevede tezimiz kapsamında yeni modeller önerilmiştir.

### 3.3. Yeni Karar Modelleri

Bu bölümde kaynaklarda var olan modellerin yanı sıra ÇEGP için üç farklı model geliştirilmiştir. İlk model Fischetti et al. tarafından EGP için geliştirilen modelin çok gezginli EGP'ye uyarlanmasıyla elde edilmiştir. İkinci model ise Angel-Bello et al. tarafından EGP için geliştirilen modelin uyarlanmasıyla elde edilmiştir. Ancak bu modelin uyarlanması diğer modele göre oldukça zor olmuştur. Üçüncü model ise yeni geliştirilen bir modeldir.

#### 3.3.1. Modeller için ortak gösterimler

ÇEGP için yapılan bu modelleme çalışmasında problem bilinen bir düğümde son bulan hamilton yolu esas alınacaktır.

Ele alınan karar modelleri için yazarlar farklı tanımlamalar ve gösterimler kullanmışlardır. Bu modeller için ortak bir gösterim yapılabilmesi ve bu sayede daha kolay bir anlatım sağlanabilmesi amacıyla dizin kümeleri, parametreler ve ortak karar değişkenleri için aşağıdaki tanımlamalara yer verilmiştir. Modellerde tanımlanan özel karar değişkenleri ayrıca belirtilecektir.

$G=(V,A)$  tam bir serimi ifade edip;

$V=\{0,1,2,\dots,n\}$  düğüm (müşteri) kümesi,

$A=\{(i,j) \mid i,j \in V, i \neq j\}$  ayrıt kümesi,

$\{0\}$  başlangıç düğümü,  $\{n\}$  bitiş düğümü göstermektedir.

Problemlerde ortak olan parametreler;

$m$ = gezgin (araç) sayısı,

$d_{ij}$ =  $i$ . düğümden  $j$ . düğüme geçiş süresi olarak ifade edilmektedir.

Ortak kullanılan karar değişkeni ise;

$$x_{ij} = \begin{cases} 1, & \text{eğer } (i,j) \text{ ayrıtı tur üzerinde ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

olarak gösterilmektedir.

### 3.3.2. Fischetti et al. modelinin uyarlanması

Fischetti et al. modelinde özel olarak tanımlanan karar değişkeni;

$$z_{ij} = \begin{cases} n-k+1, & \text{eğer } (i,j) \text{ ayrıtı } k. \text{ pozisyonda ise} \\ 0, & \text{eğer } (i,j) \text{ ayrıtı kullanılmamış ise} \end{cases}$$

olup, model aşağıdaki gibidir.

$$\sum_{j=1}^n x_{0j} = m \quad (3.19)$$

$$\sum_{j=0}^n x_{ij} = 1 \quad i=0,1,2,\dots,n \quad (3.20)$$

$$\sum_{i=1}^n x_{i0} = m \quad (3.21)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad j=0,1,2,\dots,n \quad (3.22)$$

$$\sum_{i=0}^n z_{ik} - \sum_{j=0}^n z_{kj} = \begin{cases} -n, & (k=0) \\ 1, & (k=1,2,\dots,n) \end{cases} \quad (3.23)$$

$$z_{ij} \leq r_{ij} x_{ij} \quad i=0,1,2,\dots,n, \quad j=0,1,2,\dots,n \quad (3.24)$$

$$r_{ij} = \begin{cases} 1, & \text{eğer } j=0 \text{ ise} \\ n, & \text{eğer } i=0 \text{ ise} \\ n-1, & \text{d.d.} \end{cases} \quad (3.25)$$

$$x_{ij} \in \{0,1\} \quad (3.26)$$

$$z_{ij} \geq 0 \quad \text{ve tamsayı} \quad (3.27)$$

kısıtları altında

$$Enkx_0 = \sum_{i=0}^n \sum_{j=0}^n d_{ij} z_{ij} \quad (3.28)$$

Bu modelin uyarlaması yapılırken EGP karar modelleri kapsamında ele alınan Fischetti et al. modelinin 2.1 ve 2.2 numaralı kısıtlarında değişiklik yapılarak  $m$  adet gezginin başlangıç noktasından çıkışı ve tekrar başlangıç noktasına dönüşü sağlanmış olmaktadır. Böylece EGP modelinin çok gezginli EGP haline dönüşümü sağlanmıştır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı ve  $n^2$  kadar negatif olmayan karar değişkenleri ve  $n^2+4n$  kadar kısıt bulunmaktadır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir.

Bu model izleyen sayfalarda yer alacak sayısal karşılaştırmalarda kullanılacak ve birinci yeni model anlamında *YM1* olarak yer alacaktır.

### 3.3.3. Angel-Bello et al. modelinin uyarlanması

Angel-Bello et al. modelinde diğer iki model ile ortak tanımlanan karar değişkeni bulunmamaktadır. Bu modelde özel olarak tanımlanan karar değişkenleri;

$$x_i^{(k)} = \begin{cases} 1, & \text{eğer } i.\text{düğüm } k.\text{pozisyonda ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

$$y_{ij}^{(k)} = \begin{cases} 1, & \text{eğer } i.\text{düğüm } k.\text{pozisyonda ve } j.\text{düğüm } k+1. \text{ pozisyonda ise} \\ 0, & \text{diğer durumlarda} \end{cases}$$

olmaktadır.

Ayrıca  $n+1$ . pozisyon başlangıç noktasının bulunduğu pozisyonu ifade edip, model aşağıdaki gibidir.

$$\sum_{k=1}^n x_i^{(k)} = 1 \quad i=1,2,\dots,n \quad (3.29)$$

$$\sum_{i=1}^n x_i^{(k)} \leq m \quad k=1,2,\dots,n \quad (3.30)$$

$$\sum_{i=1}^n x_i^1 = m \quad (3.31)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n+1} y_{ij}^{(k)} = x_i^{(k)} \quad i=1,2,\dots,n, \quad k=1,2,\dots,n-1 \quad (3.32)$$

$$y_{in+1}^{(n)} = x_i^{(n)} \quad i=1,2,\dots,n \quad (3.33)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^n y_{ji}^{(k)} = x_i^{(k+1)} \quad i=1,2,\dots,n, \quad k=1,2,\dots,n-1 \quad (3.34)$$

$$x_i^{(k)} \in \{0,1\} \quad i=1,2,\dots,n, \quad k=1,2,\dots,n \quad (3.35)$$

$$y_{ij}^{(k)} \geq 0 \quad i=1,\dots,n, \quad j=1,\dots,n, \quad k=1,2,\dots,n-1, \quad j \neq i \quad (3.36)$$

kısıtları altında

$$Enkx_0 = \sum_{k=1}^n \sum_{i=1}^n kd_{0i} y_{in+1}^k + \sum_{k=1}^{n-1} \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n (k-1)d_{ij} y_{ij}^{(k)} \quad (3.37)$$

Bu uyarılama elde edilirken Angel-Bello et al. tarafından en küçük gecikme problemi için geliştirilen modeldeki karar değişkeni tanımlamalarına sadık kalınmıştır. 3.29 kısıtı her düğümün bir pozisyonda olmasını sağlamaktadır. 2.19 numaralı kısıt gezgin (araç) sayısı olan  $m$  değerinden küçük eşit yapılarak 3.30 numaralı kısıt haline dönüştürülmüş ve böylece her pozisyonda en fazla gezgin sayısı kadar düğüm olması sağlanmıştır. 3.31 kısıtı ve 3.33 kısıtı bu dönüşüm yapılırken modele yeni eklenmiştir. 3.31 numaralı kısıt depodan  $m$  sayıda gezginin çıkış yapmasını sağlayan kısıttır. 3.33 numaralı kısıt ise son uğranacak düğüm belirlenirken  $y_{ij}^{(k)}$  değişkeninin aldığı değere göre  $x_i^{(k)}$  değişkeninin de değer almasını sağlamaktadır. 3.32 numaralı kısıt  $k$ . pozisyondan yalnızca bir çıkış olmasını sağlayan kısıttır. 3.34 numaralı kısıt  $k+1$ . pozisyona yalnızca bir girişin olmasını sağlamaktadır.

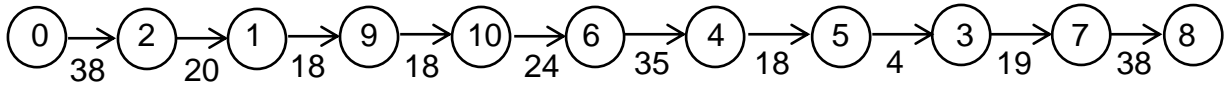
Amaç fonksiyonu yapısal olarak ve anlam yönü ile tamamen yeniden yazılmıştır. Bu modelde  $m=1$  olarak alındığında tek gezginli duruma da cevap verebilme özelliği bulunmaktadır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı ve  $n^3$  kadar negatif olamayan karar değişkenleri ve  $2n^2+n+1$  kadar kısıt bulunmaktadır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir.

Bu model izleyen sayfalarda yer alacak sayısal karşılaştırmalarda kullanılacak ve YM2 olarak isimlendirilecektir.

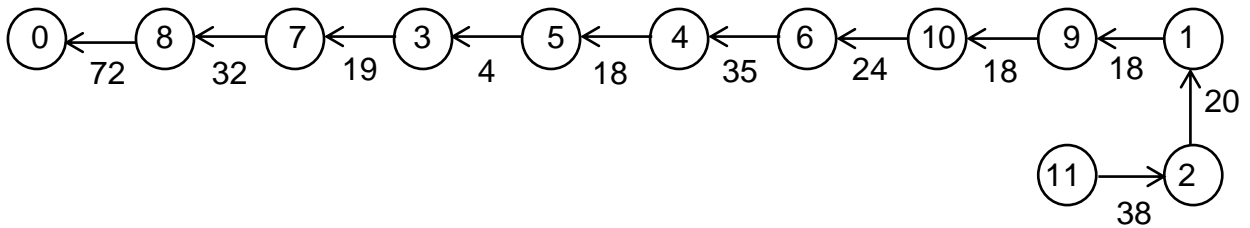
Şekil 3.1’de ele alınan 10 düğümlü örnek bir problemin Angel-Bello et al. tarafından EGP için geliştirilen model ile çözüldüğünde bulunan yol verilmiştir. Bu yola bakıldığında gezgin başlangıç düğümünü ifade eden 0. düğümden yola çıkmakta tüm düğümlere uğradıktan sonra 8. düğümden yolu tamamlamaktadır. Bu yol üzerinden toplam gecikme zamanı şu şekilde hesaplanmaktadır;

$$10x(38) + 9x(20) + 8x(18) + 7x(18) + 6x(24) + 5x(35) + 4x(18) + 3x(4) + 2x(19) + 1x(38) = 1303$$



**Şekil 3.1** Angel-Bello et al. modelinin EGP için örnek çözümü

Aynı problemin Angel-Bello et al. modelinden yola çıkılarak çok gezginli EGP için yeniden düzenlenmesiyle elde edilen model (YM2) ile çözüldüğünde bulunan yol verilmiştir. Bu yola bakıldığında gezgin başlangıç düğümünü ifade eden 11. düğümden yola çıkmakta ve yine tüm düğümlere uğradıktan sonra 8. düğümden yolu tamamlamaktadır. 8. düğümden sonra 0. düğüme yani tekrar depoya dönüş bulunan yol üzerinde görülmektedir ancak amaç fonksiyonuna dahil edilmediği için hesaba katılmamaktadır.



**Şekil 3.2** YM2 modelinin ÇEGP için örnek çözümü

### 3.3.4. Yeni bir model

Bu modelde özel olarak tanımlanan karar değişkenleri;

$t_{ij} = i$ . düğümden  $j$ . düğüme geçildiğinde  $j$ . düğümün turdaki ziyareti gerçekleşene kadar geçen süre ( $j$ . düğüm gecikmesi)

olmaktadır.

Ayrıca  $\{n+1\}$  yapay bitiş düğümünü,  $M$  ise yeterince büyük pozitif bir sayıyı ifade etmekte olup, model aşağıdaki gibidir.

$$\sum_{i=1}^n x_{0i} = m \quad (3.38)$$

$$\sum_{i=1}^n x_{in+1} = m \quad (3.39)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad j=1, \dots, n \quad (3.40)$$

$$\sum_{j=1}^{n+1} x_{ij} = 1 \quad i=1, \dots, n \quad (3.41)$$

$$t_{0i} = d_{0i} x_{0i} \quad i=1, \dots, n \quad (3.42)$$

$$t_{ij} \leq Mx_{ij} \quad i=1, \dots, n, \quad j=1, \dots, n+1 \quad (3.43)$$

$$\sum_{j=1}^{n+1} t_{ij} - \sum_{j=0}^n t_{ji} - \sum_{j=1}^n d_{ij} x_{ij} = 0 \quad i=1, \dots, n \quad (3.44)$$

$$t_{ij} \geq 0 \quad \forall (i, j) \in A \quad (3.45)$$

$$x_{ij} \in \{0,1\} \text{ ve tamsayı} \quad (3.46)$$

kısıtları altında

$$Enk \sum_{i=0}^n \sum_{j=1}^n t_{ij} \quad (3.47)$$

3.38 kısıtı başlangıç noktasından  $m$  adet aracın çıkmasını sağlamaktadır. 3.39 numaralı kısıt başlangıç noktasına  $m$  adet aracın girmesini sağlamaktadır. 3.40 kısıtı her düğüme yalnızca bir düğümden gelinmesini sağlayan ve 3.41 kısıtı her düğümden yalnızca bir düğüme geçilmesini sağlayan kısıtlardır. 3.42 ve 3.43

numaralı kısıtları değişkenler arasındaki ilişkiyi sağlamaktadır. 3.44 kısıtı ise alt tur engelleme kısıtı olmaktadır.

Bu modelde  $n^2$  kadar  $\{0,1\}$  tamsayı ve  $n^2$  kadar negatif olmayan karar değişkenleri ve  $n^2+5n+2$  kadar kısıt bulunmaktadır. Bu durum, modelin polinom büyüklükte olduğunu göstermektedir.

Bu model izleyen sayfalarda yer alacak sayısal karşılaştırmalarda kullanılacak ve YM3 olarak isimlendirilecektir.

### **3.4. Modellerin Sayısal Analizleri**

M1, YM1, YM2 ve YM3 olarak isimlendirilen dört modelin performans analizi yapılarak elde edilen CPU zamanı değerleri izleyen bölümlerde yer almaktadır.

#### **3.4.1. Çözülen problemler**

Modellerin karşılaştırmalı analizi yapılırken kaynaklarda yer alan Salehipour et al. tarafından oluşturulmuş kıyaslama problemi verilerinden 10 ve 20 düğüm için elde edilen uzaklık matrisleri kullanılmıştır [26] ve EGP için izlenen yöntemle aynı yöntem, yazılım ve donanım kullanılarak yapılmıştır. Üst süre sınırı 7200 saniye olarak alınmıştır. Ortalama ve standart sapma değerleri hesaplanırken süre sınırı içerisinde çözüm elde edilen değerler alınmıştır.

Modellerde gezgin (araç) sayısı olan  $m=1$  olarak alındığında, EGP için elde edilen sonuçlarla aynı değerlerin elde edildiği yani teorik olarak dört modelin de  $m=1$  için tek gezgine cevap verdiği görülmüştür.

Bunun yanı sıra  $m=2$  alınarak çözüm yapılmış ve dört modelin her biri için aynı sonuçlara ulaşıldığı görülmüştür. Ayrıca bu modeller arasında üstünlük gösteren model belirlenip ÇEGP için kaynaklarda bulunan tek veri seti olan Luo et al. tarafından geliştirilmiş 6 farklı problem türü kıyaslama verileri olarak kullanılmıştır.

#### **3.4.2. Sayısal sonuçlar**

Bu kesimde ele alınan dört model için 10 düğümlü ve 20 düğümlü problemlerin gezgin sayısı olan  $m$  değeri 1 ve 2 alınarak yapılan çözümler sonucunda elde edilen CPU süreleri ve hesaplanan LPR değerleri (yüzde olarak sapma) çizelge 3.1, çizelge 3.2, çizelge 3.3 ve çizelge 3.4'te verilmiştir.



10 düğümlü problemlerin  $m=1$  için çözümü sonucunda elde edilen en iyi değerler ile CPU süreleri ve ortalamaları çizelge 3.1'de gösterilmiştir.

**Çizelge 3.1** 10 düğümlü problemlerin  $m=1$  için en iyi değerleri ve CPU süreleri

Problem	En iyi Değer	M1	YM1	YM2	YM3
		CPU (sn)	CPU (sn)	CPU (sn)	CPU (sn)
1	1303	1.97	0.65	0.28	72.69
2	1517	1.48	0.71	0.15	43.61
3	1233	0.53	0.69	0.11	25.12
4	1386	2.34	0.88	0.17	45.75
5	978	0.51	0.66	0.13	24.41
6	1477	0.96	0.94	0.03	39.80
7	1163	0.64	0.79	0.02	14.77
8	1234	3.81	0.90	0.17	29.45
9	1402	1.31	0.63	0.15	23.43
10	1388	8.28	6.47	0.17	37.45
11	1405	1.87	0.93	0.09	37.63
12	1150	0.54	0.44	0.15	33.31
13	1531	5.08	4.15	0.25	29.89
14	1219	2.04	0.66	0.19	15.57
15	1087	2.71	0.77	0.19	10.03
16	1264	8.17	8.19	0.17	26.89
17	1058	0.26	0.39	0.25	10.31
18	1083	1.15	0.70	0.21	34.87
19	1394	5.84	0.88	0.18	44.90
20	951	1.36	0.44	0.11	8.90
<b>ortalama</b>	-	<b>2.54</b>	<b>1.54</b>	<b>0.16</b>	<b>30.44</b>
<b>std.sapma</b>	-	<b>2.45</b>	<b>2.15</b>	<b>0.07</b>	<b>15.39</b>

10 düğüm ve  $m=1$  için elde edilen ortalama CPU sonuçlarına bakıldığında YM2 modelinin diğerlerine göre daha hızlı çözüme ulaştığı ve bundan dolayı performansının daha iyi olduğu görülmüştür. Ayrıca standart sapma değerleri yönüyle de karşılaştırıldığında bu modelin sapmasının daha düşük olduğu görülmektedir.

10 düğümlü problemlerin  $m=1$  için çözümü sonucunda hesaplanan LPR değerleri (yüzde olarak sapma) dört model için çizelge 3.2'de verilmiştir.

**Çizelge 3.2** 10 düğümlü problemlerin  $m=1$  için LPR (% sapma) değerleri

Problem	M1	YM1	YM2	YM3
	LPR (% sapma)	LPR (% sapma)	LPR (% sapma)	LPR (% sapma)
1	0,351	0,464	0,048	0,972
2	0,462	0,605	0,050	0,991
3	0,367	0,512	0,054	0,975
4	0,473	0,617	0,125	0,988
5	0,419	0,558	0,083	0,992
6	0,423	0,572	0,000	0,978
7	0,370	0,537	0,000	0,992
8	0,549	0,701	0,086	0,982
9	0,517	0,668	0,101	0,978
10	0,584	0,717	0,172	0,986
11	0,458	0,586	0,088	0,989
12	0,366	0,509	0,076	0,993
13	0,577	0,707	0,179	0,990
14	0,464	0,600	0,108	0,994
15	0,417	0,506	0,136	0,981
16	0,547	0,699	0,060	0,989
17	0,348	0,478	0,098	0,987
18	0,497	0,604	0,237	0,984
19	0,433	0,555	0,107	0,978
20	0,263	0,421	0,002	0,993
<b>ortalama</b>	<b>0,444</b>	<b>0,581</b>	<b>0,101</b>	<b>0,986</b>

10 düğüm ve  $m=1$  için hesaplanan LPR % sapma değerlerine bakıldığında *YM2* modelinin diğerlerine göre sapması daha düşük olarak bulunmuştur. Bu açıdan da diğer modellere üstünlük sağladığı görülmektedir.

10 düğümlü problemlerin  $m=2$  için çözümü sonucunda elde edilen en iyi değerler ile CPU süreleri ve ortalamaları çizelge 3.3'te gösterilmiştir.

**Çizelge 3.3** 10 düğümlü problemlerin  $m=2$  için en iyi değerleri ve CPU süreleri

Problem	En iyi Değer	M1	YM1	YM2	YM3
		CPU (sn)	CPU (sn)	CPU (sn)	CPU (sn)
1	828	0.13	0.16	0.11	31.34
2	872	0.17	0.22	0.02	7.77
3	846	0.21	0.44	0.22	16.69
4	821	0.28	0.28	0.22	8.90
5	596	0.20	0.24	0.02	6.55
6	939	0.31	0.28	0.19	10.37
7	741	0.24	0.29	0.02	3.82
8	681	0.28	0.30	0.14	6.50
9	820	0.24	0.35	0.20	6.72
10	756	0.27	0.60	0.18	5.83
11	767	0.10	0.28	0.02	5.12
12	712	0.14	0.24	0.07	7.17
13	812	0.34	0.63	0.11	3.52
14	738	0.27	0.31	0.18	6.30
15	690	0.21	0.20	0.25	6.17
16	676	0.47	0.35	0.09	4.11
17	691	0.11	0.22	0.23	3.64
18	661	0.27	0.20	0.18	22.52
19	829	0.21	0.32	0.15	8.65
20	687	0.30	0.27	0.13	16.16
<b>ortalama</b>	-	<b>0.24</b>	<b>0.31</b>	<b>0.14</b>	<b>9.39</b>
<b>std.sapma</b>	-	<b>0.09</b>	<b>0.12</b>	<b>0.08</b>	<b>7.12</b>

10 düğüm ve  $m=2$  için elde edilen ortalama CPU sonuçlarına bakıldığında YM2 modelinin diğerlerine göre daha hızlı çözüme ulaştığı ve bundan dolayı performansının daha iyi olduğu görülmüştür. Ayrıca standart sapma değerleri yönüyle de karşılaştırıldığında bu modelin sapmasının daha düşük olduğu görülmektedir.

10 düğümlü problemlerin  $m=2$  için çözümü sonucunda hesaplanan LPR değerleri (yüzde olarak sapma) dört model için çizelge 3.4'te verilmiştir.

**Çizelge 3.4** 10 düğümlü problemlerin  $m=2$  için LPR (% sapma) değerleri

Problem	M1	YM1	YM2	YM3
	LPR (% sapma)	LPR (% sapma)	LPR (% sapma)	LPR (% sapma)
1	0,061	0,157	0,011	0,909
2	0,166	0,313	0,000	0,952
3	0,157	0,288	0,058	0,924
4	0,205	0,353	0,077	0,935
5	0,137	0,275	0,000	0,963
6	0,187	0,327	0,038	0,928
7	0,120	0,273	0,000	0,957
8	0,256	0,458	0,014	0,928
9	0,238	0,432	0,003	0,918
10	0,314	0,480	0,087	0,947
11	0,130	0,243	0,000	0,957
12	0,113	0,206	0,006	0,944
13	0,298	0,448	0,087	0,953
14	0,192	0,339	0,070	0,962
15	0,149	0,222	0,025	0,919
16	0,259	0,436	0,018	0,948
17	0,111	0,201	0,010	0,935
18	0,231	0,351	0,126	0,927
19	0,139	0,251	0,038	0,922
20	0,096	0,198	0,021	0,955
<b>ortalama</b>	<b>0,178</b>	<b>0,313</b>	<b>0,043</b>	<b>0,939</b>

10 düğüm ve  $m=2$  için hesaplanan LPR % sapma değerlerine bakıldığında  $YM2$  modelinin diğerlerine göre sapması daha düşük olarak bulunmuştur. Bu açıdan da diğer modellere üstünlük sağladığı görülmektedir.

Ele alınan bu dört model 20 düğümlü 20 adet problemde  $m=1$  için çözdürüldüğünde  $M1$ ,  $YM1$  ve  $YM3$  modellerinin üst sınır olarak verilen 7200 saniye içerisinde en iyi çözüme ulaşamadığı ancak  $YM2$  modelinin bu sürenin oldukça altında ortalama bir süre içerisinde 20 düğümlü ele alınan 20 problem için  $m=1$  değeri ile en iyi çözüme ulaştığı görülmüştür. En iyi değerler ve çözüm süreleri Çizelge 3.5'te verilmiştir.

**Çizelge 3.5** 20 düğümlü problemlerin  $m=1$  için en iyi değerler ve CPU süreleri

Problem	En iyi Değer	YM2
		CPU (sn)
1	3175	4.47
2	3248	18.07
3	3570	12.45
4	2983	9.51
5	3248	31.45
6	3328	20.22
7	2809	15.95
8	3461	22.23
9	3475	21.56
10	3359	10.25
11	2916	13.35
12	3314	2.46
13	3412	9.45
14	3297	17.88
15	2862	6.75
16	3433	8.64
17	2913	17.18
18	3124	17.73
19	3299	14.65
20	2796	3.60
<b>ortalama</b>	-	<b>13.89</b>
<b>std.sapma</b>	-	<b>7.25</b>

20 düğümlü problemlerin  $m=1$  için çözümü sonucunda belirlenen üst süre sınırı içerisinde yalnızca bir model çözüme ulaştığından 20 düğümlü problemlerin  $m=1$  için LPR değerleri (yüzde olarak sapma) karşılaştırması yapılmamıştır.

Ele alınan bu dört model 20 düğümlü 20 adet problemde  $m=2$  için çözdürüldüğünde  $M1$ ,  $YM1$  ve  $YM3$  modellerinin üst sınır olarak verilen 7200 saniye içerisinde bazı problemlerde en iyi çözüme ulaştığı ancak bazılarında ulaşamadığı görülmüştür.  $YM2$  modelinin ise bu sürenin oldukça altında ortalama bir süre içerisinde 20 düğümlü ele alınan 20 problem için  $m=2$  değeri ile en iyi çözüme ulaştığı görülmüştür. En iyi değerler ve çözüm süreleri Çizelge 3.6'da verilmiştir.

**Çizelge 3.6** 20 düğümlü problemlerin  $m=2$  için en iyi değerleri ve CPU süreleri

Problem	En iyi Değer	M1	YM1	YM2	YM3
		CPU (sn)	CPU (sn)	CPU (sn)	CPU (sn)
1	1950	379.54	456.47	7.47	7200
2	1655	169.15	1369.75	1.23	1297.77
3	1926	401.82	814.67	1.50	7200
4	1871	7200	7200	11.30	7200
5	1724	2153.24	7200	9.50	7200
6	1887	7200	7200	15.97	7200
7	1560	7200	7200	8.46	7200
8	1962	7200	7200	6.87	7200
9	1989	7200	7200	20.90	7200
10	1948	1489.03	7200	2.66	7200
11	1697	1990.11	7200	15.35	7200
12	2000	479.47	1589.03	1.73	7200
13	1879	1032.91	2576.46	3.37	7200
14	1723	423.04	2131.60	2.00	7200
15	1795	172.51	383.68	2.93	7200
16	1989	3879.36	7200	4.15	7200
17	1570	1687.05	7200	2.22	7200
18	1696	7200	7200	14.97	7200
19	1859	162.84	146.14	2.38	7200
20	1674	109.04	275.11	1.89	7200
<b>ortalama</b>	-	<b>1037.79</b>	<b>1082.55</b>	<b>6.84</b>	<b>1297.77</b>
<b>std.sapma</b>	-	<b>1092.27</b>	<b>876.94</b>	<b>5.97</b>	-

20 düğümlü problemlerin  $m=2$  için çözümü sonucunda belirlenen üst süre sınırı içerisinde yalnızca bir model tüm problemler için çözüme ulaştığından 20 düğümlü problemlerin  $m=2$  için LPR değerleri (yüzde olarak sapma) karşılaştırması yapılmamıştır.

Tüm bu sonuçlara bakıldığında 10 düğüm ve 20 düğüm için  $m=1$  ve  $m=2$  alındığında YM2 modelinin diğer modellere göre daha üstün olduğu tespit edilmiştir. Bu nedenden dolayı bundan sonraki düğüm sayısı artırılarak ve m değeri değiştirilerek yapılacak analizler YM2 modeli için yapılmış ve sonuçlar elde edilmiştir.

Bu aşamada Luo et al. tarafından oluşturulan kıyaslama problemi verileri kullanılmıştır. Bu veriler 29 düğüm için  $m=6$ , 39 düğüm için  $m=8$  ve 49 düğüm için  $m=10$  alınarak oluşturulmuş problemlerdir. YM2 modeli öncelikle bu üç durum için

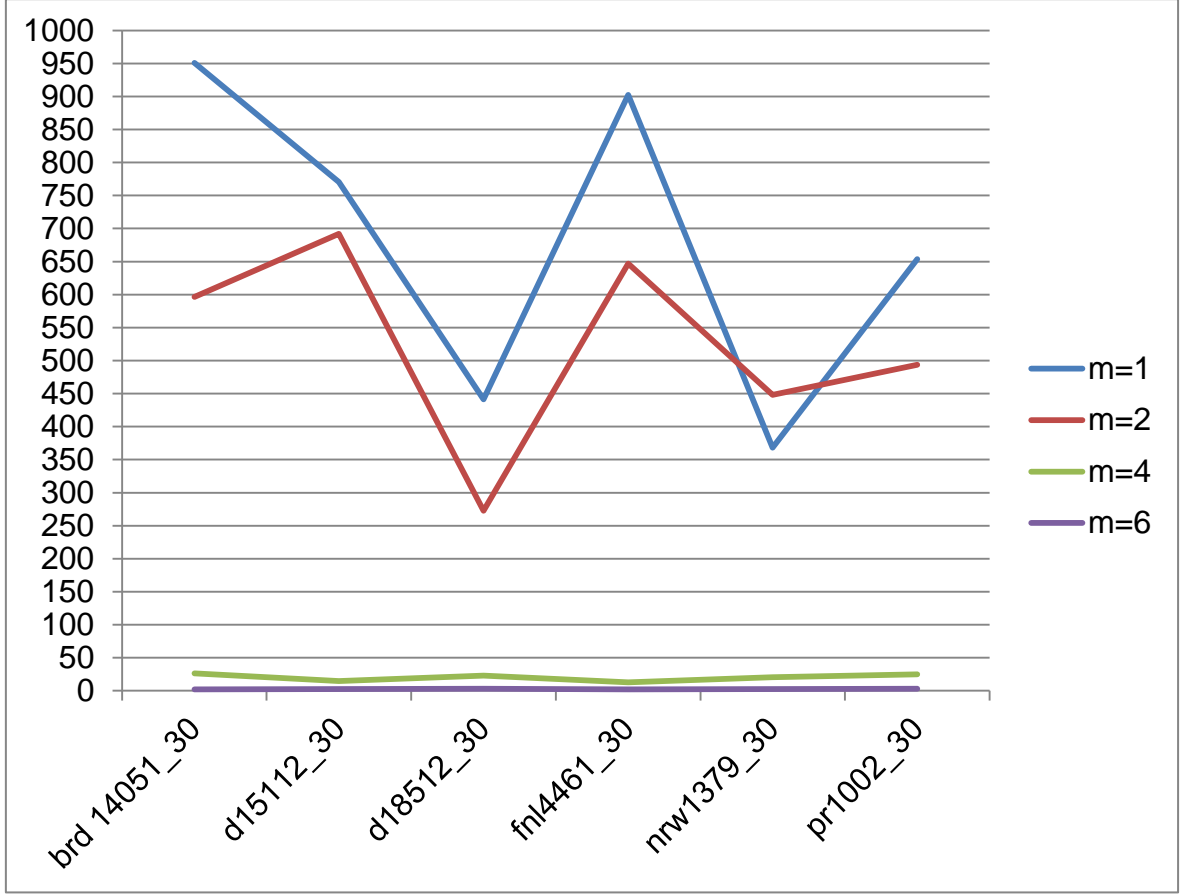
çözdürülmüş ve CPU sonuçları elde edilmiştir. Daha sonra  $m$  değerleri için de bir analiz yapılabilmesi için 29 düğüm için  $m=6$  değerinin yanı sıra  $m=1$ ,  $m=2$  ve  $m=4$  durumları için de  $YM2$  modeli çözdürülerek en iyi değerler ve CPU süreleri elde edilmiştir. 29 düğüm için  $m=1$  ve  $m=2$  alınarak bulunan sonuçlar Ek 1’de, 29 düğüm için  $m=4$  ve  $m=6$  alınarak bulunan sonuçlar ise Ek 2’de verilmiştir.

Bulunan bu değerler sonucunda 29 düğüm için dört farklı  $m$  değeri alınarak altı farklı problem tipine göre hesaplanan ortalama CPU değerleri aşağıdaki tabloda gösterilmiştir.

**Çizelge 3.7** 29 düğümlü problemler için ortalama CPU süreleri

Problem	Ortalama CPU (sn)			
	$m=1$	$m=2$	$m=4$	$m=6$
brd14051	951.02	596.32	26.48	2.22
d15112	770.64	691.86	14.78	2.74
d18512	441.39	272.80	23.00	2.86
fnl4461	902.51	647.07	12.67	2.00
nrv1379	368.30	447.98	20.38	2.56
pr1002	653.71	493.83	24.80	3.05

Ortalama CPU değerleri ile  $m$  değerleri arasındaki ilişkiye bakıldığında  $m$  değeri arttıkça CPU zamanının azaldığı görülmüştür. Bu ilişki Şekil 3.3’teki grafik yardımı ile de görülebilmektedir.



**Şekil 3.3** 29 düğümlü problemler için CPU ve  $m$  grafiği

Daha sonra  $YM2$  modeli 39 düğüm için  $m=8$  değerinin yanı sıra  $m=6$  ve  $m=10$  değerleri ile de çözdürülerek en iyi değerler ve CPU süreleri elde edilmiştir. Bulunan bu değerler Ek 3'te verilmiştir.

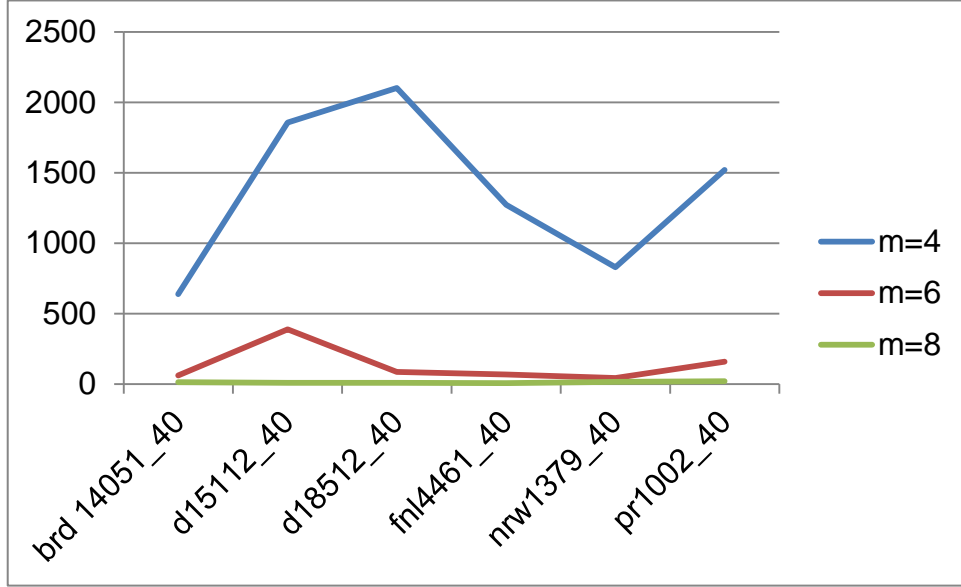
Bulunan bu değerler sonucunda 39 düğüm için üç farklı  $m$  değeri alınarak altı farklı problem tipine göre hesaplanan ortalama CPU değerleri aşağıdaki tabloda gösterilmiştir.

**Çizelge 3.8** 39 düğümlü problemler için ortalama CPU süreleri

Problem	Ortalama CPU (sn)		
	m=4	m=6	m=8
brd14051	636.98	61.00	13.16
d15112	1855.41	388.28	9.44
d18512	2102.10	86.68	7.77
fnl4461	1271.12	66.74	7.52
nrw1379	829.32	41.81	15.18
pr1002	1520.57	158.88	20.44



Ortalama CPU değerleri ile  $m$  değerleri arasındaki ilişkiye bakıldığında  $m$  değeri arttıkça CPU zamanının azaldığı görülmüştür. Bu ilişki Şekil 3.4'teki grafik yardımı ile de görülebilmektedir.



**Şekil 3.4** 39 düğümlü problemler için CPU ve  $m$  grafiği

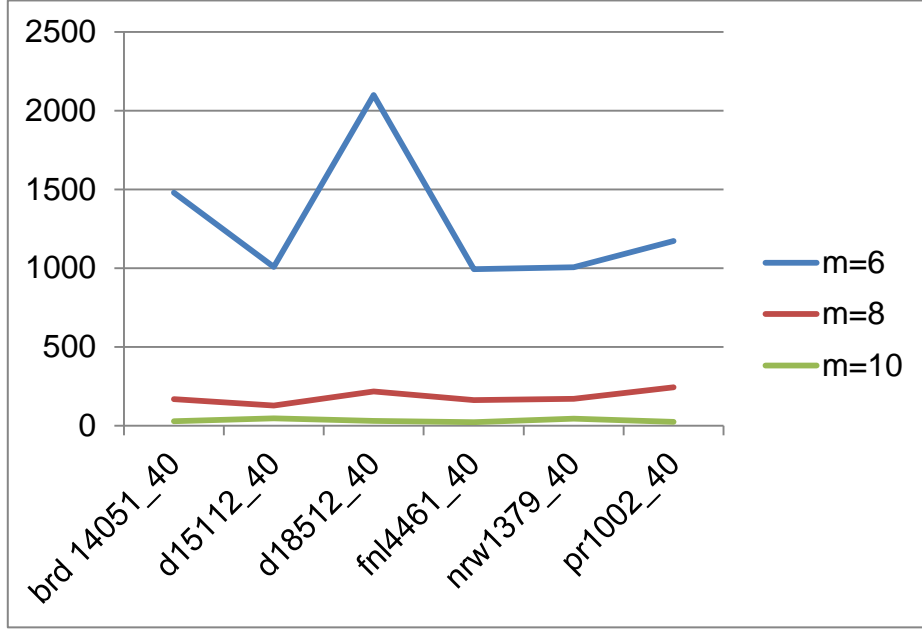
YM2 modeli 49 düğüm için  $m=10$  değerinin yanı sıra  $m=6$  ve  $m=8$  değerleri ile de çözdürülerek en iyi değerler ve CPU süreleri elde edilmiştir. Bulunan bu değerler Ek 4'te verilmiştir.

Bulunan bu değerler sonucunda 49 düğüm için üç farklı  $m$  değeri alınarak altı farklı problem tipine göre hesaplanan ortalama CPU değerleri aşağıdaki tabloda gösterilmiştir.

**Çizelge 3.9** 49 düğümlü problemler için ortalama CPU süreleri

Problem	Ortalama CPU (sn)		
	m=6	m=8	m=10
brd14051	1480.20	168.71	29.10
d15112	1008.61	129.34	46.75
d18512	2099.08	217.37	31.47
fnl4461	992.95	163.94	23.30
nrw1379	1005.57	171.75	44.59
pr1002	1172.71	244.34	25.27

Ortalama CPU deęerleri ile m deęerleri arasındaki iliřkiye bakıldıęında m deęeri arttıka CPU zamanının yine azaldıęı grlmřtr. Bu iliřki Őekil 3.5'teki grafik yardımı ile de grlebilmektedir.



**Őekil 3.5** 49 dęml problemler iin CPU ve m grafięi

Yapılan analizler sonucunda belirlenen st sre sınırı dahilinde YM2 modelinin tez kapsamında ele alınan dięer modellere gre hem CPU hem de LPR deęerleri ynyle stnlk saęladıęı tespit edilmiřtir. YM2 modelinin ele alınan tm problemler iin zm bulabildięi grlmřtr.

#### 4. SONUÇ VE ÖNERİLER

EGP problemi müşterilerin toplam gecikme zamanını en küçükmeyi amaçlayan bir problem türü olduğu için uygulama alanları ve önemi oldukça büyük olmaktadır. Kaynaklarda bu konuda yapılan çalışmalar incelenmiş ve henüz karşılaştırılması yapılmamış matematiksel modeller tespit edilmiştir. Bu modeller kaynaklarda yer alan 10 ve 20 düğümlü kıyaslama problemi verileri için çözdürülmüştür. Ele alınan modeller için karşılaştırmalı analizler yapılarak ortalama CPU süreleri ve LP gevşetme değerleri yönüyle bakıldığında EGP modelleri arasında Angel-Bello et al. tarafından geliştirilen modelin üstünlük gösterdiği belirlenmiştir.

EGP probleminin uzantıları arasında uygulama alanı yönüyle gerçek hayatta daha çok karşılaşılan ve bundan dolayı en önemli özel durumu olarak görülen çok gezginli EGP problemi de tez kapsamında ele alınmıştır. Kaynaklar incelendiğinde bu konuda yapılan çalışmaların oldukça az ve kullanım kolaylığı açısından çok verimli olmadığı görülmüştür. ÇEGP için, EGP problemi için oluşturulmuş iki modelin uyarlaması yapılmış ve yeni bir model de geliştirilmiştir. ÇEGP için geliştirilmiş olan bir model ile birlikte toplamda dört model için sayısal analizler yapılmıştır. Bu analizler için yine iki farklı kıyaslama problemi verileri kullanılmıştır. Bu modeller arasından *YM2* olarak isimlendirilen modelin diğerlerine göre üstün olduğu belirlenmiştir. Tez kapsamında *YM2* modeli için farklı düğüm sayıları ve farklı *m* değerleri için birçok analiz yapılmıştır.

Yapılan bu analizler ile ulaşılan sonuçlar şu şekilde özetlenebilmektedir:

- a) *YM2* modeli her durumda üstünlük göstermiştir.
- b) Problem boyutu arttıkça CPU süresinin de arttığı görülmüştür.
- c) *m* değeri (gezgin/araç sayısı) arttıkça CPU süresinin azaldığı tespit edilmiştir.

Ele alınan bu modeller EGP'nin diğer yaygın kullanım alanına sahip uzantısı olan zaman pencereli durum içinde uyarlaması yapılabilir. Ayrıca EGP ve ÇEGP matematiksel modellerinin düğüm sayısı artırılarak yapılacak çözümlerde performansları incelenebilir. ÇEGP için farklı *m* değerleri kullanılarak çeşitli analizler gelecek çalışmalarda yapılabilir. Bu konuda yapılacak çalışmaların gerçek hayatta kullanılabilirliği açısından ve bilime yapacağı katkı açısından önemli bir yere sahip olduğu düşünülmektedir.

## EKLER

Ek 1 29 düğümlü problemlerin m=1 ve m=2 alındığında en iyi değerleri ve CPU süreleri

Problem	m=1		m=2	
	En iyi değer	CPU (sn)	En iyi değer	CPU (sn)
brd14051_0	318413.2	148.42	177167.1	215.86
brd14051_1	285349.8	256.89	171783.0	208.68
brd14051_2	238224.5	219.08	123569.9	91.64
brd14051_3	279345.6	450.16	150255.0	80.81
brd14051_4	318978.0	1123.56	162412.4	123.17
brd14051_5	271537.2	849.38	141151.0	319.63
brd14051_6	318498.2	163.07	181769.4	184.74
brd14051_7	324248.4	4264.81	164271.9	2380.01
brd14051_8	284057.9	1098.58	148950.2	644.51
brd14051_9	372686.4	936.20	193193.1	1714.17
<b>ortalama</b>		<b>951.02</b>		<b>596.32</b>
<b>std.sapma</b>		<b>1228.22</b>		<b>797.13</b>
d15112_0	935165.6	651.25	494055.1	272.53
d15112_1	867970.4	361.56	457614.4	1015.68
d15112_2	916587.6	587.67	469243.6	159.46
d15112_3	1028681.8	1398.74	526278.0	1217.02
d15112_4	826278.9	243.16	436594.3	131.48
d15112_5	931363.9	2299.83	495723.7	2390.72
d15112_6	831304.2	882.27	449825.0	1280.53
d15112_7	887818.5	112.44	506834.3	86.05
d15112_8	904957.0	380.73	449942.8	63.91
d15112_9	994429.6	788.79	487803.4	301.21
<b>ortalama</b>		<b>770.64</b>		<b>691.86</b>
<b>std.sapma</b>		<b>651.86</b>		<b>767.43</b>
d18512_0	356145.0	787.63	174603.5	101.61
d18512_1	341292.6	344.29	170515.1	238.48
d18512_2	364768.6	538.74	188838.3	285.62
d18512_3	359866.9	415.05	187039.3	448.46
d18512_4	341525.8	396.09	175926.2	221.13
d18512_5	309811.4	692.11	163012.4	115.10
d18512_6	342095.3	235.89	184028.6	780.21
d18512_7	356019.5	326.45	171255.7	54.30
d18512_8	298263.4	389.86	152933.4	175.70
d18512_9	300696.9	287.77	165503.1	307.33
<b>ortalama</b>		<b>441.39</b>		<b>272.80</b>
<b>std.sapma</b>		<b>178.29</b>		<b>212.10</b>
fnl4461_0	195104.3	121.89	95497.2	83.50
fnl4461_1	174727.3	2529.41	89121.0	631.47
fnl4461_2	165199.0	209.21	92495.0	518.73

fnl4461_3	186473.8	787.61	101690.5	2319.29
fnl4461_4	208051.0	228.13	117037.3	277.45
fnl4461_5	199313.7	640.04	100348.8	395.72
fnl4461_6	210710.0	653.79	106585.6	167.30
fnl4461_7	190518.0	450.75	100183.5	860.70
fnl4461_8	209941.3	1760.28	109296.8	649.46
fnl4461_9	183903.9	1643.94	95893.8	567.03
<b>ortalama</b>		<b>902.51</b>		<b>647.07</b>
<b>std.sapma</b>		<b>804.51</b>		<b>633.57</b>
nrv1379_0	111717.1	218.10	59098.2	60.15
nrv1379_1	125148.5	713.47	65748.6	724.77
nrv1379_2	124320.2	257.27	64383.4	168.15
nrv1379_3	112532.6	135.97	61748.9	179.96
nrv1379_4	96082.7	306.47	46178.3	19.16
nrv1379_5	112880.6	411.50	57028.5	133.93
nrv1379_6	124159.7	480.29	65896.4	1311.73
nrv1379_7	118671.0	664.11	61282.8	1222.80
nrv1379_8	112143.5	161.64	62303.1	356.22
nrv1379_9	104150.4	334.13	55462.8	302.89
<b>ortalama</b>		<b>368.30</b>		<b>447.98</b>
<b>std.sapma</b>		<b>199.34</b>		<b>475.58</b>
pr1002_0	784784.0	879.19	403661.5	399.89
pr1002_1	670514.1	332.69	374110.6	1012.57
pr1002_2	705804.9	741.72	352708.9	232.58
pr1002_3	588090.6	420.37	305594.0	263.19
pr1002_4	624604.1	145.66	316325.5	121.19
pr1002_5	653295.3	614.73	343099.0	124.90
pr1002_6	595339.7	343.32	324379.4	209.16
pr1002_7	677389.3	513.94	364235.8	1144.57
pr1002_8	656951.3	240.38	354802.8	358.62
pr1002_9	655314.8	2305.13	330445.1	1071.66
<b>ortalama</b>		<b>653.71</b>		<b>493.83</b>
<b>std.sapma</b>		<b>622.69</b>		<b>412.45</b>

**Ek 2** 29 düğümlü problemlerin m=4 ve m=6 alındığında en iyi değerleri ve CPU süreleri

Problem	m=4		m=6	
	En iyi değer	CPU (sn)	En iyi değer	CPU (sn)
brd14051_0	113182.5	31.06	97380.3	1.79
brd14051_1	112580.6	39.01	96322.5	3.96
brd14051_2	77945.9	5.53	64109.4	1.08
brd14051_3	101636.4	9.95	89582.5	2.42
brd14051_4	101502.0	5.86	87615.7	2.69
brd14051_5	88048.7	27.56	75079.5	2.67
brd14051_6	111963.8	5.23	94540.8	1.28
brd14051_7	99772.9	110.73	81515.8	1.67
brd14051_8	85302.5	5.87	74160.8	2.57
brd14051_9	110811.4	23.98	90628.1	2.11
<b>ortalama</b>		<b>26.48</b>		<b>2.22</b>
<b>std.sapma</b>		<b>32.14</b>		<b>0.84</b>
d15112_0	286768.2	18.83	225070.2	1.56
d15112_1	259294.8	24.28	213332.3	2.12
d15112_2	260350.6	7.67	208323.6	1.70
d15112_3	301754.9	55.11	222870.4	1.70
d15112_4	262634.7	7.80	216056.0	4.41
d15112_5	271367.1	8.12	235215.8	5.48
d15112_6	239689.3	3.77	207139.0	2.26
d15112_7	325720.3	2.82	280309.0	1.10
d15112_8	286738.6	6.63	244015.4	3.15
d15112_9	284514.8	12.79	238976.2	3.88
<b>ortalama</b>		<b>14.78</b>		<b>2.74</b>
<b>std.sapma</b>		<b>15.68</b>		<b>1.44</b>
d18512_0	111386.4	24.79	87971.3	3.97
d18512_1	105366.1	4.79	91563.1	3.42
d18512_2	99541.2	5.93	81268.9	3.63
d18512_3	111446.2	21.75	88322.4	2.37
d18512_4	114838.5	65.25	94606.6	4.56
d18512_5	107804.9	61.72	87036.6	3.94
d18512_6	103909.4	5.20	84771.6	1.46
d18512_7	101078.2	4.12	80245.1	1.06
d18512_8	90613.2	28.93	72064.4	1.77
d18512_9	99903.5	7.47	80339.5	2.42
<b>ortalama</b>		<b>23.00</b>		<b>2.86</b>
<b>std.sapma</b>		<b>23.24</b>		<b>1.20</b>
fnl4461_0	60756.0	2.65	51192.2	1.24
fnl4461_1	51778.9	3.21	44154.7	2.36
fnl4461_2	54666.1	3.57	46571.2	1.67
fnl4461_3	57211.5	15.87	48591.4	2.78
fnl4461_4	70448.6	33.56	54485.9	1.31

fnl4461_5	59522.9	7.91	47907.3	2.76
fnl4461_6	58028.5	12.51	45882.1	2.88
fnl4461_7	54755.0	9.38	44545.3	1.56
fnl4461_8	63946.5	31.46	50365.3	1.21
fnl4461_9	58332.6	6.58	49179.6	2.20
<b>ortalama</b>		<b>12.67</b>		<b>2.00</b>
<b>std.sapma</b>		<b>11.27</b>		<b>0.68</b>
nrv1379_0	37886.4	43.97	31249.4	3.35
nrv1379_1	38658.0	2.86	33138.5	1.92
nrv1379_2	38935.3	6.30	31872.0	1.26
nrv1379_3	38723.6	42.59	31777.1	3.72
nrv1379_4	31941.3	44.17	26671.2	2.29
nrv1379_5	34376.0	4.57	29010.3	2.28
nrv1379_6	36675.6	5.13	30398.1	2.71
nrv1379_7	36619.9	38.13	30765.5	3.65
nrv1379_8	36314.4	13.21	28796.4	3.02
nrv1379_9	31066.6	2.90	26271.2	1.39
<b>ortalama</b>		<b>20.38</b>		<b>2.56</b>
<b>std.sapma</b>		<b>19.08</b>		<b>0.88</b>
pr1002_0	223557.0	27.36	168188.8	1.66
pr1002_1	231908.2	4.92	195805.8	2.71
pr1002_2	228201.7	121.98	182635.0	8.83
pr1002_3	174915.0	6.52	139784.7	2.17
pr1002_4	200543.6	11.09	164916.7	2.22
pr1002_5	201043.3	10.62	163642.3	1.87
pr1002_6	200958.0	36.19	160585.9	4.27
pr1002_7	202142.3	9.61	166887.0	3.56
pr1002_8	205085.0	5.23	161025.8	1.38
pr1002_9	180775.9	14.52	144167.0	1.83
<b>ortalama</b>		<b>24.80</b>		<b>3.05</b>
<b>std.sapma</b>		<b>35.62</b>		<b>2.22</b>

**Ek 3** 39 düğümlü problemlerin m=4, m=6 ve m=8 alındığında en iyi değerleri ve CPU süreleri

Problem	m=4		m=6		m=8	
	En iyi değer	CPU (sn)	En iyi değer	CPU (sn)	En iyi değer	CPU (sn)
brd14051_0	133032.8	536.15	106431.5	11.10	97630.7	7.43
brd14051_1	142663.5	901.94	119134.3	22.32	110671.1	16.96
brd14051_2	163326.3	1574.27	139738.0	236.49	127629.7	29.79
brd14051_3	131803.0	349.33	109281.2	77.66	99527.6	8.05
brd14051_4	161351.2	250.99	135428.7	81.23	123881.8	20.62
brd14051_5	133817.3	699.29	108098.3	17.56	98329.4	12.12
brd14051_6	142790.6	407.03	120788.4	73.60	110676.6	20.15
brd14051_7	129647.9	83.23	111836.2	22.07	103775.5	4.41
brd14051_8	142571.3	1280.54	113550.4	54.80	101387.3	6.22
brd14051_9	113887.9	287.04	95093.0	13.12	87945.1	5.86
<b>ortalama</b>		<b>636.98</b>		<b>61.00</b>		<b>13.16</b>
<b>std.sapma</b>		<b>481.97</b>		<b>67.79</b>		<b>8.39</b>
d15112_0	417244.6	4115.62	323430.9	60.05	287734.8	5.07
d15112_1	388764.4	6622.03	297434.7	65.20	256987.1	5.21
d15112_2	439763.1	7200	340963.5	139.37	307407.1	11.01
d15112_3	400913.9	783.54	329180.8	275.65	292602.7	8.45
d15112_4	417856.1	186.80	338260.4	75.83	299259.0	4.89
d15112_5	382488.8	311.17	301543.6	10.13	269559.4	3.57
d15112_6	451067.8	463.21	362320.1	74.50	320989.4	5.82
d15112_7	425672.4	7200	330482.4	3152.08	287270.7	38.16
d15112_8	431213.1	1243.51	338853.9	16.06	303263.9	6.42
d15112_9	392696.3	1117.39	312756.2	13.88	282412.3	5.77
<b>ortalama</b>		<b>1855.41</b>		<b>388.28</b>		<b>9.44</b>
<b>std.sapma</b>		<b>2299.20</b>		<b>974.29</b>		<b>10.31</b>
d18512_0	153195.3	1188.52	120719.5	9.53	108982.1	11.62
d18512_1	141861.3	1870.24	110483.1	5.60	102490.3	5.95
d18512_2	172469.2	4217.77	136757.4	159.70	121904.8	6.85
d18512_3	159840.8	1305.04	126363.4	170.48	111537.3	8.30
d18512_4	154638.2	2248.39	125366.1	170.03	109343.7	12.75
d18512_5	156778.4	222.11	129657.2	6.58	119095.8	7.32
d18512_6	152726.2	1508.82	119134.7	48.78	106544.9	7.54
d18512_7	146619.6	3358.54	115106.9	116.88	102674.3	5.33
d18512_8	140531.6	3550.11	114382.1	169.01	102622.9	7.29
d18512_9	133596.4	1551.41	103209.9	10.24	92937.5	4.74
<b>ortalama</b>		<b>2102.10</b>		<b>86.68</b>		<b>7.77</b>
<b>std.sapma</b>		<b>1242.14</b>		<b>76.89</b>		<b>2.58</b>
fnl4461_0	90145.8	478.67	71847.7	158.39	63096.0	6.34
fnl4461_1	96904.4	264.89	76277.1	20.10	66882.6	6.23
fnl4461_2	96339.6	1446.61	76893.7	116.05	70151.4	14.64
fnl4461_3	86265.9	525.26	67168.3	21.15	58843.9	3.70
fnl4461_4	86313.6	319.53	69222.4	11.69	61654.9	6.45



fnl4461_5	82532.7	5020.12	63157.3	41.92	56144.5	5.87
fnl4461_6	87627.1	331.17	69729.5	34.17	61274.9	9.06
fnl4461_7	97340.4	1618.55	74151.4	43.03	65698.3	6.11
fnl4461_8	90911.6	793.51	73250.9	208.79	64260.9	5.40
fnl4461_9	83489.2	1912.88	65083.1	12.14	58717.5	11.37
<b>ortalama</b>		<b>1271.12</b>		<b>66.74</b>		<b>7.52</b>
<b>std.sapma</b>		<b>1446.54</b>		<b>69.53</b>		<b>3.26</b>
nrv1379_0	47398.4	877.60	39035.2	12.56	35655.2	9.37
nrv1379_1	47005.8	419.46	37436.9	13.46	33000.7	2.76
nrv1379_2	54628.5	308.88	45057.3	34.77	39928.1	5.14
nrv1379_3	49841.9	286.47	39980.4	18.38	36685.4	10.50
nrv1379_4	52495.2	1479.35	40952.9	23.67	36168.6	4.36
nrv1379_5	52871.6	419.30	42416.5	31.72	38005.4	4.29
nrv1379_6	45026.9	342.11	35976.9	88.55	31837.3	6.95
nrv1379_7	54634.9	2637.52	43489.7	31.44	39394.8	93.27
nrv1379_8	51732.2	7200	40615.2	98.48	36674.5	9.49
nrv1379_9	52008.6	693.23	41614.3	65.03	36447.7	5.62
<b>ortalama</b>		<b>829.32</b>		<b>41.81</b>		<b>15.18</b>
<b>std.sapma</b>		<b>778.78</b>		<b>31.15</b>		<b>27.56</b>
pr1002_0	340145.4	7200	265755.7	243.14	233387.2	6.39
pr1002_1	315118.9	7200	250793.6	521.5	218781.4	22.75
pr1002_2	303097.1	6640.25	229389.8	8.13	211241.7	4.18
pr1002_3	271245.4	828.96	215667.5	13.42	196120.4	8.77
pr1002_4	321016.7	7200	253277.1	152.01	227450.1	55.18
pr1002_5	278686.9	255.51	221198.1	8.96	194802.2	3.43
pr1002_6	330617.4	248.03	259316.6	12.91	229730.1	5.33
pr1002_7	329195.4	7200	262591.8	546.37	236896.1	78.28
pr1002_8	317840.5	882.09	253923.4	72.00	230126.6	16.37
pr1002_9	274494.0	268.56	215094.5	10.35	192179.9	3.74
<b>ortalama</b>		<b>1520.57</b>		<b>158.88</b>		<b>20.44</b>
<b>std.sapma</b>		<b>2525.45</b>		<b>212.41</b>		<b>25.75</b>

**Ek 4** 49 düğümlü problemlerin m=6, m=8 ve m=10 alındığında en iyi değerleri ve CPU süreleri

Problem	m=6		m=8		m=10	
	En iyi değer	CPU (sn)	En iyi değer	CPU (sn)	En iyi değer	CPU (sn)
brd14051_0	157071.4	7200	140731.4	176.42	133642.3	24.07
brd14051_1	146133.7	1643.45	129995.1	180.70	123212.7	41.43
brd14051_2	160074.0	3966.93	142908.7	18.81	137175.1	17.49
brd14051_3	177233.5	1171.75	158730.9	123.16	150209.8	93.64
brd14051_4	137322.0	424.26	122096.3	20.11	116278.7	23.25
brd14051_5	143357.5	84.38	129708.8	17.45	124648.2	19.53
brd14051_6	149261.2	3100.98	131766.2	559.00	121190.4	21.52
brd14051_7	151265.8	7200	132583.5	131.37	124077.6	17.47
brd14051_8	142578.9	45.93	131269.3	26.74	125446.0	19.17
brd14051_9	142655.5	1403.91	127581.5	433.29	118925.0	13.45
<b>ortalama</b>		<b>1480.20</b>		<b>168.71</b>		<b>29.10</b>
<b>std.sapma</b>		<b>1417.18</b>		<b>186.66</b>		<b>23.90</b>
d15112_0	423416.6	1394.41	378026.2	426.45	353657.8	201.36
d15112_1	446757.1	3558.01	383107.9	42.97	355115.2	13.37
d15112_2	459137.1	116.07	415357.3	85.14	392196.1	14.90
d15112_3	450782.3	7200	382947.9	82.44	350821.8	9.52
d15112_4	415068.3	229.71	371642.9	109.65	341493.6	9.12
d15112_5	429675.1	1472.87	384852.6	67.00	360717.4	21.97
d15112_6	473951.8	333.12	421875.1	40.75	390251.4	8.82
d15112_7	389732.7	255.30	349028.8	264.90	327701.9	161.10
d15112_8	411010.1	908.92	364711.9	138.82	344600.5	17.36
d15112_9	425304.6	809.07	377421.8	35.23	347783.6	9.99
<b>ortalama</b>		<b>1008.61</b>		<b>129.34</b>		<b>46.75</b>
<b>std.sapma</b>		<b>1080.55</b>		<b>124.42</b>		<b>71.63</b>
d18512_0	159920.5	1318.81	141644.0	396.83	131512.3	19.27
d18512_1	146896.5	599.70	129685.5	22.66	121078.4	25.58
d18512_2	187473.1	4245.29	163941.6	512.66	152941.3	82.16
d18512_3	161019.8	822.65	139750.1	29.12	130662.6	14.25
d18512_4	171342.6	7200	147363.3	83.25	138082.5	17.22
d18512_5	154224.9	<b>7200</b>	<b>135636.1</b>	<b>389.64</b>	<b>126707.5</b>	<b>41.91</b>
d18512_6	169845.8	<b>398.14</b>	<b>147812.3</b>	<b>18.94</b>	<b>140764.3</b>	<b>25.64</b>
d18512_7	146677.3	2403.09	125371.5	37.00	115801.8	14.75
d18512_8	163910.5	3973.55	140853.3	470.89	129326.8	53.79
d18512_9	162475.2	3031.42	143021.5	212.66	134276.4	20.11
<b>ortalama</b>		<b>2099.08</b>		<b>217.37</b>		<b>31.47</b>
<b>std.sapma</b>		<b>1533.48</b>		<b>204.56</b>		<b>21.86</b>
fnl4461_0	101110.6	1163.24	87290.0	19.44	81562.2	13.87
fnl4461_1	96052.1	509.03	85707.9	167.93	79804.8	42.66
fnl4461_2	96283.8	3175.40	80224.0	19.25	73309.1	8.90
fnl4461_3	96928.1	510.86	84662.1	21.68	79335.1	20.88
fnl4461_4	92873.1	7200	81812.3	829.50	75052.0	12.68

fnl4461_5	90424.5	717.46	80852.3	25.56	76738.1	38.75
fnl4461_6	94202.3	1027.82	81955.1	51.00	75268.9	14.59
fnl4461_7	88238.5	806.60	77958.9	40.79	72956.3	32.04
fnl4461_8	86894.5	616.89	75863.8	68.16	70244.0	13.52
fnl4461_9	96143.4	409.29	87269.3	396.11	82157.0	35.07
<b>ortalama</b>		<b>992.95</b>		<b>163.94</b>		<b>23.30</b>
<b>std.sapma</b>		<b>855.32</b>		<b>261.45</b>		<b>12.54</b>
nrv1379_0	48506.1	1795.31	41836.2	31.95	39206.1	22.21
nrv1379_1	55006.6	7200	47478.8	275.75	44741.6	260.75
nrv1379_2	53321.8	754.85	48386.3	63.64	45914.2	14.72
nrv1379_3	55003.5	1563.52	48198.8	17.95	46208.0	15.00
nrv1379_4	53851.2	1103.92	46809.9	339.71	43557.9	12.19
nrv1379_5	56014.8	1574.24	49817.5	370.45	46718.4	34.04
nrv1379_6	59742.7	793.18	52708.8	63.74	49421.1	18.65
nrv1379_7	58280.7	119.73	52898.1	456.62	49960.1	38.10
nrv1379_8	50894.4	677.65	44940.5	53.54	41560.9	15.04
nrv1379_9	51853.3	667.77	46819.1	44.10	44404.0	15.18
<b>ortalama</b>		<b>1005.57</b>		<b>171.75</b>		<b>44.59</b>
<b>std.sapma</b>		<b>545.71</b>		<b>168.81</b>		<b>76.45</b>
pr1002_0	350126.9	7200	295210.5	64.85	272908.4	15.00
pr1002_1	309339.5	709.30	270966.5	59.47	249275.4	12.30
pr1002_2	343659.1	7200	298848.6	468.25	277959.1	80.45
pr1002_3	355072.6	4372.25	315074.6	39.01	298846.1	17.73
pr1002_4	317477.3	625.09	281393.3	41.89	262518.4	20.04
pr1002_5	341467.9	581.03	298434.7	57.41	273318.8	10.99
pr1002_6	352719.2	7200	306753.0	494.46	280317.3	26.00
pr1002_7	295503.8	194.42	264471.6	366.67	246341.0	30.32
pr1002_8	313398.2	554.15	279473.4	243.47	256971.4	15.59
pr1002_9	322275.1	7200	289970.3	607.92	267596.7	24.26
<b>ortalama</b>		<b>1172.71</b>		<b>244.34</b>		<b>25.27</b>
<b>std.sapma</b>		<b>1577.43</b>		<b>222.19</b>		<b>20.35</b>

## KAYNAKLAR LİSTESİ

- [1] ABELEDO, H., FUKASAWA R., PESSOA, A., UCHOA E., The time dependent traveling salesman problem: Polyhedra and algorithm. Tech. Rep. RPEP, Universidade Federal Fluminense Brasil, vol.10, no.15, 2010a.
- [2] ABELEDO, H., FUKASAWA R., PESSOA A., UCHOA E., The time dependent traveling salesman problem: Polyhedra and branch-cut-and-price algorithm, In: Proceedings of the 9th International Symposium on Experimental Algorithms, SEA 2010, s.202-213, 2010b.
- [3] ANGEL-BELLO, F., ALVAREZ, A., AND GARCIA, I., Two improved formulations for the minimum latency problem, Applied Mathematical Modelling, vol.37, no.4, s.2257-2266, 2013.
- [4] ARCHER, A., BLASIÁK, A., Improved approximation algorithms for the minimum latency problem via prize-collecting strolls, In: Proceedings of the 21th Annual ACM-SIAM Symposium on Discrete Algorithms, s.429-447, 2010.
- [5] ARCHER, A., WILLIAMSON, D.P., Faster approximation algorithms for the minimum latency problem In: Proceedings of the 40th Annual ACM-SIAM Symposium on discrete algorithms, s.88-96, 2003.
- [6] ARORA, S., KARAKOSTAS, G., Approximation schemes for minimum latency problems. SIAM Journal on Computing, vol.32, no.5, s.1317-1337, 2003.
- [7] AUSIELLO, G., LEONARDI, S., MARCHETTI-SPACCAMELA, A., On salesmen, repairmen, spiders, and other traveling agents. In: Proceedings of the 4th Italian Conference on Algorithms and Complexity, s.1-16, 2000.
- [8] BEKTAS, T., The multiple traveling salesman problem: An overview of formulations and solution procedures. Omega, vol.34, no.3, s.209-219, 2006.
- [9] BENNETT, B.T., and GAZIS, D.C., School bus routing by computer, Transportation Research/UK/, vol.6, s.317-325, 1972.
- [10] BIANCO, L., MINGOZZI, A., RICCIARDELLI, S., The traveling salesman problem with cumulative costs. Networks, vol.23, no.2, s.81-91, 1993.
- [11] BİGRAS, L.-P., GAMACHE, M., SAVARD, G., The time dependent traveling salesman problem and single machine scheduling problems with sequence dependent setup times. Discrete Optimization, vol.5, no.4, s.685-699, 2008.
- [12] BLUM, A., CHALASANIT, P., PULLEYBLANKT, B., RAGHAVAN, P., SUDAN, M., The minimum latency problem. In: Proceedings of the 26th Annual ACM Symposium on Theory of Computing, s.163-171, 1994.

- [13] CAMPBELL, A., VANDENBUSSCHE, D., HERMANN, V., Routing for relief efforts, *Transportation Science*, vol.42, no.2, s.127-145, 2008.
- [14] CHAUDHURI, K., GODFREY, B., RAO, S., TALWAR, K., Paths, trees and minimum latency tours. In: *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, FOCS*, s.36-45, 2003.
- [15] CONWAY, R., MAXWELL, W., and MILLER, L., *Theory of scheduling*. Addison-Wealey, 1967.
- [16] DANTZIG, G.B., FULKERSON, D.R., JOHNSON, S.M., Solution of a large-scale traveling salesman problem, *Operations Research*, vol.2, no.4, s.393-410, 1954.
- [17] DEWILDE, T., CATTERYSSSE, D., COENE, S., SPIEKSMAN, F.C.R., VANSTEENWEGEN, P., Heuristics for the traveling repairman problem with profits. In: *Proceedings of the 10th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization and Systems, ATMOS 2010*, s.34-44, 2010.
- [18] EIJL, C.A., A polyhedral approach to the delivery man problem. Tech. Rep. COSOR 95-19. Eindhoven University of Technology, 1995.
- [19] EZZINE, I.O., SEMET, F., CHABCHOUB, H., New formulations for the traveling repairman problem. In: *Proceedings of the 8th International conference of modeling and simulation*, 2010.
- [20] FAKCHAROENPHOL, J., HARRELSON, C., & RAO, S., The k-traveling repairmen problem. *ACM Transactions on Algorithms*, vol.3, 2007.
- [21] FISCHETTI, M., LAPORTE, G., MARTELLO, S., The deliveryman problem and cumulative matroids, *Oper. Res.*, vol.41, no.6, s.1055-1064, 1993.
- [22] GOEMANS, M.X., KLEINBERG, J.M., An improved approximation ratio for the minimum latency problem. *Mathematical Programming*, vol.82, no.1, s.111-124, 1998.
- [23] GOUVEIA, L., VOSS, S., A classification of formulations for the (time-dependent) traveling salesman problem. *European Journal of Operational Research*, vol.83, no.1, s.69-82, 1995.
- [24] JOTHI, R., RAGHAVACHARI, B., Approximating the k-traveling repairman problem with repair times. *Journal of Discrete Algorithms*, vol.5, no.2, s.293-303, 2007.
- [25] KAN, R., A.H.G. *Machine scheduling problems*, Martinus Nijhoff, The Hague, 1976.

- [26] KARA, I., DERYA T., Formulations for Minimizing tour duration of the traveling salesman problem with time Windows, 4th World Conference on Business, Economics and Management, 2015.
- [27] KARA, I., KARA, B. Y., YETIS, M. K., Cumulative vehicle routing problems. In T.Caric, & H. Gold (Eds.), Vehicle routing problem, I-Tech, Vienna Austria, s.85-98, 2008.
- [28] LÌ, L.Y.O., and FU, Z., The school bus routing problem: A case study, The Journal of the Operational Research Society, vol.53, s.552-558, 2002.
- [29] LUCENA, A., Time dependent traveling salesman problem-the deliveryman case, Networks, vol.20, s.753-763, 1990.
- [30] LUO, Z., QÌN, H., LÌM, A., Branch-and-price-and-cut for the multiple traveling repairman problem with distance constraints, European Journal of Operational Research, vol.234, s.49-60, 2014.
- [31] MENDEZ-DÌAZ, I., ZABALA, P., LUCENA, A., A new formulation for the traveling deliveryman problem, Discrete applied mathematics, vol.156, s.3223-3237, 2008.
- [32] NAGARAJAN, V., RAVÌ, R., The directed minimum latency problem, In: Proceedings of the 11th International Workshop, APPROX 2008, and 12th International Workshop, RANDOM 2008 on Approximation, Randomization and Combinatorial Optimization: Algorithms and Techniques, s.193-206, 2008.
- [33] NGUEVEU, S., PRÌNS, C., WOLFLER CALVO, R., An effective memetic algorithm for the cumulative capacitated vehicle routing problem, Computers and Operations Research, vol.37, no.11, s.1877-1885, 2010.
- [34] PÌCARD, J., QUEYRANNE, M., The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling, Operations Research, vol.26 no.1, s.86-110, 1978.
- [35] RÌBEÌRO, G., LAPORTE, G., An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem, Computers and Operations Research, vol.39, no.3, s.728-735, 2012.
- [36] SALEHÌPOUR, A., SÖRENSEN, K., GOOS, P., BRAYSY, O., Efficient GRASP +VND and GRASP + VNS metaheuristics for the traveling repairman problem. 40R: A Quarterly Journal of Operations Research, vol.9, no.2, s.189-209, 2011.

- [37] SARUBBÌ, J., LUNA, H., MÍRANDA, G., Minimum latency problem as a shortest path problem with side constraints, in: Book of Extended Abstracts of the XIV Latin Ibero-American Congress on Operations Research (CLAIO 2008), 2008.
- [38] SÍLVA, M. M., SUBRAMANIAN, A., VÍDAL, T., and OCHÍ, L. S., A simple and effective metaheuristic for the minimum latency problem, European Journal of Operational Research, v.221, s.513-520, 2012.
- [39] SÍMCHL-LEVI, D., BERMAN, O., Minimizing the total flow time of n jobs on a network, IIE Trans, vol.23, no.3, s.236-244, 1991.
- [40] SÍTTERS, R., The minimum latency problem is NP-hard for weighted trees. In: proceedings of the 9th international conference on integer programming and combinatorial optimization, IPCO 2002, s.230-239, 2002.
- [41] SVESTKA, J. A., and HUCKFELDT, V. E., Computational experience with an m-salesman traveling salesman algorithm, Management Science, vol.19, s.790-799, 1973.
- [42] WU, B.Y., HUANG, Z.N, ZHAN, F.J., Exact algorithms for the minimum latency problem, Information Processing Letters, vol.92, no.6, s.303-309, 2004.