



T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**HAMMADDE TEDARİK AKTİVİTESİ İÇİN KESİN ZAMAN PENCERELİ
ARAÇ ROTALAMA OPTİMİZASYONU**

Kuntay AKCA

Doç. Dr. Âli Yurdun ORBAK
(Danışman)

YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

BURSA 2015

TEZ ONAYI

Kuntay AKCA tarafından hazırlanan “HAMMADDE TEDARİK AKTİVİTESİ İÇİN KESİN ZAMAN PENCERELİ ARAÇ ROTALAMA OPTİMİZASYONU” adlı tez çalışması aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Uludağ Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı’nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. Âli Yurdun ORBAK

Başkan : Doç. Dr. Âli Yurdun ORBAK İmza
Uludağ Üniversitesi Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı

Üye : Doç. Dr. Fatih ÇAVDUR İmza
Uludağ Üniversitesi Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı

Üye : Yrd. Doç. Dr. İlkün ORBAK İmza
Maltepe Üniversitesi Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı

Üye : İmza
Uludağ Üniversitesi Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı

Üye : İmza
Uludağ Üniversitesi Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım

Prof. Dr. Ali Osman DEMİR
Enstitü Müdürü
.././.....

U.Ü. Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

.././....

İmza

Kuntay AKCA

İÇİNDEKİLER

ÖZET.....	i
ABSTRACT.....	ii
TEŞEKKÜR.....	iii
ŞEKİLLER DİZİNİ.....	iv
TABLolar DİZİNİ.....	v
1. GİRİŞ.....	1
1.1 Problemin Tanımı ve Önemi.....	1
1.2 Çalışmanın Amacı.....	6
1.3 Çalışmanın Kapsamı.....	7
1.4 Araç Rotalama Problemlerinin Kavramsal Çerçevesi.....	9
2. KAYNAK ÖZETLERİ.....	10
2.1 Araç Rotalama Tanımı.....	10
2.2 Araç Rotalama Problem Prensipleri.....	11
2.3 Araç Rotalama Temel Bileşenleri.....	12
2.4 Araç Rotalama Çeşitleri.....	13
2.4.1 Dinamik ve Statik Çevre Durumuna Göre.....	13
2.4.2 Rotalama Durumuna Göre.....	13
2.4.3 Kısıtlarına Göre.....	14
2.5 Araç Rotalama Problemleri Çözüm Yöntemleri.....	15
2.5.1 Kesin Yöntemler.....	15
2.5.1.1 Dal ve Sınır Algoritması:.....	15
2.5.1.2 Dal ve Kesme Algoritması:.....	16
2.5.1.3 Küme Kapsama Algoritması.....	16
2.5.2 Klasik Sezgisel Yöntemler.....	17
2.5.2.1 Clarke ve Wright Tasarruf Algoritması.....	18
2.5.2.2 Clarke ve Wright Gelişmiş Tasarruf Algoritması.....	19
2.5.2.3 Sıralı Ekleme Sezgisel Algoritması.....	19
2.5.2.3.1 Cristofides, Mingozzi ve Toth Sıralı Ekleme Sezgiseli.....	19
2.5.2.3.2 Mole ve Jameson Sıralı Ekleme Sezgiseli.....	20
2.5.2.4 Eşleme Tabanlı Tasarruf Algoritması.....	20

2.5.2.5	Basit Kümeleme Yöntemi	21
2.5.2.5.1	Süpürge Algoritması	21
2.5.2.5.2	Fisher ve Jaikumar Algoritması	22
2.5.2.5.3	Bramel ve Simchi-Levi Algoritması	23
2.5.2.6	Budanmış Dal ve Sınır Algoritması	24
2.5.2.7	Taç Yaprağı Algoritması	25
2.5.2.8	Önce Rotala Sonra Grupla Algoritması	25
2.5.2.9	Tek ve Çok Rotalı İyileştirmeli Sezgisel Algoritması	25
2.5.3	Meta sezgisel Yöntemler	26
2.5.3.1	Benzetimli Tavlama	26
2.5.3.2	Deterministik Tavlama	27
2.5.3.3	Genetik Algoritmalar	27
2.5.3.4	Karınca Algoritması	28
2.5.3.5	Tabu Arama	30
2.5.3.6	Yapay Sinir Ağları	32
3.	MATERYAL ve YÖNTEM	38
3.1	Materyal	38
3.2	Yöntem	42
4.	BULGULAR ve TARTIŞMA	44
4.1	Durum analizi	44
4.1.1	Malzeme Siparişi ve Teslim planı	46
4.1.2	Malzeme Yükleme ve Boşaltma Operasyonu	47
4.1.3	Hedef, Amaç ve İstekler	47
4.2	Problemin tanımı ve içeriği	49
4.3	Problemin Varsayımları	51
4.4	Matematiksel Model	52
4.5	Sayısal Örnek / Deneysel Çalışma	58
4.5.1	Grup A ile Örnek Çözüm	59
4.5.2	Grup B, C ve D Çözümleri ve Grup A ile karşılaştırma	64
5.	SONUÇ	67
	KAYNAKLAR	69
	EKLER	72

ÖZGEÇMİŞ	124
----------------	-----

ÖZET

Yüksek Lisans Tezi

HAMMADDE TEDARİK AKTİVİTESİ İÇİN KESİN ZAMAN PENCERELİ ARAÇ ROOTALAMA OPTİMİZASYONU

Kuntay AKCA

Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Âli Yurdun ORBAK

Bu çalışma, klasik Araç Rotalama Probleminin (ARP) geliştirilmiş bir hali olan Zaman Pencereci Araç Rotalama Problemi (ZP_ARP) dikkate alınarak hazırlanmıştır. ZP_ARPde klasik ARPden farklı olarak araçların doluluklarını en yüksek değere ulaştırmakla birlikte, uğrayacakları noktaların çalışma veya yükleme saatlerine uygun olarak kullanılmasına izin verilmektedir. Pratikte sıklıkla karşılaşılan bir problem olmasına rağmen, ZP_ARP ile ilgili literatürde az sayıda çalışma bulunmaktadır. Bu çalışmada, ZP_ARP için bir karma tamsayılı doğrusal programlama modeli önerilmiştir. Tedarikçiler; 5, 10, 15 ve 20'lik gruplara ayrılarak, önerilen bu matematiksel modelin çalışma kapasitesi ve süresi test edilmiş ve sonuçları sunulmuştur.

Anahtar kelimeler: Karma Tamsayılı Doğrusal Programlama Modeli, Zaman Penceresi, Araç Rotalama Problemi, Araçlarda Yüksek Doluluk Oranı

2015, v + 124 sayfa.

ABSTRACT

MSc Thesis

SUPPLY CHAIN MATHEMATICAL MODEL FOR THE VEHICLE ROUTING PROBLEM WITH TIME WINDOWS

Kuntay AKCA

Uludağ University
Graduate School of Natural and Applied Sciences
Department of Industrial Engineering

Supervisor: Assoc. Prof. Dr. Âli Yurdun ORBAK

In this paper, the vehicle routing problem with time windows (VRP_TW) which is the generalized version of the classic vehicle routing problem, is considered. Unlike the classic Vehicle Routing Problem, vehicles are allowed to be used according to shipment calendar of the suppliers with maximum filling ratio in the VRP_TW. Although the VRP_TW is often encountered in practice, there are very few studies in literature. In this study, a mathematical model proposed for the VRP_TW. Suppliers are divided in some groups with 5, 10, 15 and 20 members and the efficiency of this proposed mathematical model is analyzed.

Keywords: Mixed Integer Linear Programming Model, Time Windows, Vehicle Routing Problem, High Filling Rate

2015, v + 124 pages.

TEŐEKKÜR

Tezimin konusunun belirlenmesinde, arařtırma ařamasında, yön tayininde ve tamamlanmasında, kısaca tüm çalıřma boyunca bilgi ve deneyimleri ile destek olan deęerli hocam ve tez danıřmanım Sayın Doç. Dr. Âli Yurdun ORBAK'a teőekkür ederim.

İř yařantımla birlikte yürütmek zorunda olduęum yüksek lisans fırsatını deęerlendirmeme izin veren, tezin güncel yařamla iliřkisini kurmak, gerçek veriler üzerinden hareket ederek yařamda kullanılabilir hale getirilmesinde desteęini esirgemeyen Lojistik Prosesleri, Organizasyonları ve Nakliye Departman Müdürüm Sayın Sibel BAYAT'a teőekkürlerimi bir borç bilirim.

Hedefe ulařmak için emek vermeyi ve bir iři sonuç alana kadar sürdürmeyi bana küçük yařta öğreten çok deęerli annem Bircan AKCA'ya en içten teőekkürlerimi sunarım.

Ve son olarak da gösterdięi sabır ve verdięi her türlü destek için hayat yoldařım, sevgili eřiim Gözde AKCA'ya teőekkürlerimi sunarım.

Kuntay AKCA

Eylül 2015

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1	Yurtdışı tedarikçiler için gerçekleştirilen taşıma tiplerinin kartografisi	7
Şekil 2	Araç Rotalama Örnek Şeması	10
Şekil 3	Araç Rotalama Probleminin Şebeke Gösterimi	11
Şekil 4	Süpürge Algoritması Adımları	22
Şekil 5	Fisher ve Jaikumar Algoritması İçin Kurulan Yapı	22
Şekil 6	Bramel ve Simichi-Levi Algoritması Örneği	23
Şekil 7	Karıncaların Davranışları	29
Şekil 8	Tabu arama algoritması	31
Şekil 9	ARP için Yapay Sınır Ağı	35
Şekil 10	Nakliye biçimleri	40
Şekil 11	İş akışına ait şema	41
Şekil 12	Üretici firma, taşıyıcı firma ve tedarikçi ilişkileri	44
Şekil 13	2014 yılı şirket doluluk oranları	49
Şekil 14	Tedarikçi uzaklıklarının harita üzerinde gösterimi	62
Şekil 15	Yüklü bir kamyonun 2 yan cepheden ve yukarıdan görünümü	63
Şekil 16	Sonuçların harita üzerinde gösterimi	65
Şekil 17	Çözüm raporu	67
Şekil 18	Çözümün harita üzerindeki görseli	67

TABLÖLAR DİZİNİ

	Sayfa
Tablo 1 Grup A tedarikçi listesi	61
Tablo 2 Tedarikçiler arası uzaklık matrisi	61
Tablo 3 5 tedarikçi arası uzaklıkların saat cinsinden ifadesi	62
Tablo 4 Tedarikçilerden alınacak günlük malzeme miktarı	63
Tablo 5 Tedarikçilerden alınacak günlük malzeme ağırlıkları	64
Tablo 6 Tedarikçi çalışma saatleri	64
Tablo 7 Senaryolar hakkında karşılaştırmalı bilgiler	68

1. GİRİŞ

1.1 Problemin Tanımı ve Önemi

Uluslararası ticaret, son çeyrek yüzyılda siyasi ve ekonomik gelişmeler ışığında hızlı bir gelişim sürecine girmiştir. Ülkeler arasındaki ticari ve ekonomik sınırların kalkması ve ulusal pazarlarda küresel sermayeye rahatça hareket edebilme olanağı tanınması, şirketlerin yaşam stratejilerini değiştirmiştir. İşletmelerin uluslararası alanda faaliyet göstermeye başlaması, pazar olanaklarını arttırmakla birlikte, rekabet unsurunun da yoğun biçimde hissedilmesine neden olmuştur.

Günümüz rekabetçi ortamında dağıtım maliyetleri ve zamanlaması işletmelerin karlılıklarını ve rekabet gücünü etkileyen önemli faktörler arasında yer almaktadır. Bu sebeple dağıtım rotalarının belirlenmesinde bilimsel ve teknolojik yöntemler kullanılması ile işletmeler lojistik maliyet tasarrufu sağlayarak önemli avantajlar elde edebilecek, kar marjlarını arttırabilecek ve aynı zamanda ülkemiz ekonomisindeki bazı kaynakların (petrol, iş gücü, zaman vb.) daha verimli kullanılması sağlanacaktır.

Tedarik zincir yönetimi kapsamında yer alan lojistik, genel olarak ürünün ya da hizmetin bir tedarik noktasından çeşitli talep noktalarına taşınması olarak tanımlanmaktadır. Tam bir lojistik sistemi hammaddelerin satıcılardan ya da tedarikçilerden alınarak taşınması, bu hammaddelerin üretim için imalat fabrikalarına dağıtılması, üretilen bu ürünlerin depolara ya da dağıtım merkezlerine aktarılması ve son olarak müşterilere dağıtılmasını kapsamaktadır.

Hem dağıtım hem de tedarik prosedürleri etkili bir taşıma yönetimine ihtiyaç duymaktadır. İyi bir dağıtım yönetimi işletmenin toplam dağıtım maliyetlerinde önemli ölçüde bir azalma sağlayabilmektedir (Tan 2000).

Lojistik kavramsal olarak ilk defa askeri ihtiyaçlardan doğmuş ve gelişmiştir. Eski Yunan, Roma ve Bizans İmparatorluklarında malzemelerin temininden ve ikmalinden sorumlu 'Logistikas' unvanı verilen askeri memurlar bulunmaktaydı (Anonim 2015).

Kelime olarak, “oran, hesaplama, neden” anlamlarına gelen, eski yunanca “logos(λ ó γ o ς)”dan türemiştir (Anonim 2012). Kelimenin sözlük tanımı "Askeri malzeme, teçhizat ve personelin; tedarik, tasıma ve idamesi ile uğrasan askeri bir bilim. Bir harekâtın detaylarının idaresi.” olarak yapılmıştır (Anonim 2012). Bir tanıma göre lojistik, malzeme, hizmet, bilgi ve sermaye akısı yönetimi için bir is planlama çerçevesidir. Günümüz is çevresinin gittikçe artan karmaşıklıkta bilgi, iletişim ve kontrol sistemlerini içermektedir (Anonim 1996).

Bir başka tanıma göre lojistik, müşteri gereksinimlerini karşılamak amacıyla bir merkezden tüketim noktalarına malların, hizmetlerin ve ilgili bilginin etkin ve etkili bir biçimde akısı ve depolanmasının planlama, uygulama ve kontrol etme sürecidir (Anonim 1998). Yine bir başka tanıma göre de lojistik, müşteri gereksinimlerini karşılamak amacıyla bir merkezden tüketim noktalarına hammadde, yarı mamul, bitmiş ürünler ve ilgili bilginin etkin ve maliyet etkili bir biçimde akısı ve depolanmasının planlama, uygulama ve kontrol etme sürecidir (Anonim 1998).

Lojistik kavramı ile yakından ilişkili kavramlardan birisi ise ulaştırma. Ulaştırma insanların veya malların bir yerden başka bir yere nakledilmesidir. Zaten İngilizce Transport kelimesinin kökenine bakılırsa, Latince trans ve portare kelimelerinin birleşiminden oluşmaktadır. Yani karsıdan karsıya tasıma anlamına gelmektedir (Anonim 2015). Ulaştırma kavramının birçok bileşeni vardır. Bunlar; altyapı, araçlar, operasyonlar olarak basitçe 3'e ayrılabilir. Altyapı, ulaşım ağları ve ulaşım terminallerinden oluşmaktadır; araçlar her türlü hareketliye verilen isimdir; operasyonlar ise ulaştırma sistemin kontrol edilmesini sağlarlar. Örneğin bunlar; trafik ışıkları, demiryolu makasları, hava trafik kontrolörleri ve/veya geçiş ücretleri, akaryakıt vergileri gibi sistemin nasıl finanse edileceği konusundaki kurallar olabilir.

Ulaştırma sistemlerinin ülkelerin ekonomisi, sektörler ve şirketler için önemi büyüktür ve bu konuda çarpıcı raporlar vardır. Örneğin Kearny'in 1984 yılında National Council of Physical Distribution Management (NCPDM) için hazırladığı raporda 1983 yılında Amerika Birleşik Devleti'nde ki yıllık dağıtım maliyetlerinin 650 milyon \$ (yaklaşık milli gelirin %21'i) olduğunu tahmin etmiştir. Ayrıca yine bu raporda taşıma

maliyetlerinin üretimdeki kontrol edilebilir maliyetlerin %22,5'ini oluşturduğundan da bahsedilmektedir (Kearny 1984). Ulaştırma Barosu İstatistikleri (Bureau of Transportation Statistics) internet sitesinin istatistiklerine göre Amerika Birleşik Devleti hükümetinin, tüm ulusal ulaştırma sistemlerinin inşa, bakım, işletme ve yönetim harcamalarının 2001 mali yılındaki tutarı 183,1 milyar \$'dır (Anonim 1998).

Ulaştırma, taşıma ve dağıtım konularında üzerinde durulan ilk problemlerden biri Gezgin Satıcı Problemi (GSP) dir. GSP'nin kökeni, 1880'lerde Sir William R. Hamilton tarafından bulunan, İkosyan Oyununa dayanmaktadır (Biggs ve ark. 1976). Bu oyunda amaç, 20 noktadan oluşan bir İcosahedron'un tüm noktalarını bir kez ziyaret edecek bir yol bulmaktır. Bulunan bu yola Hamilton Turu adı verilir. GSP probleminde 1 hareketli vardır, eğer birden fazla (m) hareketli varsa, bu problem m-GSP olarak tanımlanır. m-GSP probleminde hareketliler araçlar ise bu probleme özel olarak Araç Rotalama Problemi (ARP) adı verilir.

Araç Rotalama Problemi (ARP), ilk olarak Dantzig ve Ramser tarafından tanımlanmıştır (Dantzig ve Ramser 1959). Dantzig ve Ramser çalışmalarında, bir gerçek hayat uygulaması olan servis istasyonlarına benzin dağıtım problemini ele almışlar ve çözüm için bir matematiksel programlama modeli ve algoritmik yaklaşım ortaya koymuşlardır. İlk ortaya çıktığı yıllarda küçük bir grup matematikçinin ilgisini çeken ARP, giderek bir çok disiplindeki araştırmacının üzerinde çalıştığı alan haline gelmiştir.

ARP'de birden fazla ve genelde birbiriyle çelişen amaçlar dikkate alınabilmektedir. En bilinen amaçlar şunlardır:

- Araçların toplam katettiği mesafeye (ya da toplam seyahat süresine) ve kullanılan araçların sabit maliyetlerine (ve ilgili sürücüye) bağlı olan toplam taşıma maliyetlerini en küçüklemek
- Tüm müşterilere hizmet etmek için gerekli olan toplam araç sayısını (ya da sürücü sayısını) en küçüklemek
- Rotaları seyahat süresi ve araç yükü açısından dengelemek
- Müşterilere parçalı dağıtım yapılmasından kaynaklanan cezaları en küçüklemek

Yukarıda sayılan amaçların çeşitli kombinasyonları da problemin amaç fonksiyonunu oluşturabilmektedir. ARP sahip olduğu kısıtlara göre, geri toplamalı araç rotalama problemleri, dağıtım toplamalı araç rotalama problemleri, zaman pencereci araç rotalama problemleri gibi farklı türlere sahiptir (Toth ve Vigo 2002).

Kombinatoriyel en iyilemenin en önemli ve çok çalışılan problemlerinden biri olan ARP'de, belirli bir müşteri kümesine hizmet edecek olan bir araç filosunun izleyeceği rotaların en iyisinin belirlenmesine çalışılır. Klasik ARP problemlerinin çözümü, her rotanın depodan başlayıp depo ile bittiği ve her bir müşteriye bir kez uğranması kısıtının sağlandığı rotalar kümesidir. Bunun yanı sıra problemin türüne göre bazı yan kısıtların da sağlanması gerekebilmektedir. En yaygın olan yan kısıtlar; kapasite kısıtı, bir rotada olabilecek en fazla talep noktası kısıtı, bir rotada aracın toplam süre kısıtı, talep noktalarına hizmetin başlanabileceği zaman penceresi kısıtı, bir talep noktasının başka bir talep noktasından önce ziyaret edilmesinin gerektiği öncelik kısıtıdır (Laporte 1992).

Dağıtım toplamalı araç rotalama problemleri; her rotanın depoda başlayıp depoda bittiği, her müşteriye bir aracın gittiği; rotanın toplam talebinin araç kapasitesini aşmadığı; her müşterinin arz ve talebinin karşılandığı problemlerdir (Çetin ve Gencer 2010). Dağıtım ve toplama işleminin gerçekleştirilmesine göre:

- Önce dağıtım sonra toplama,
- Karışık dağıtım toplama,
- Eş zamanlı dağıtım ve toplama

problemleri olmak üzere üçe ayrılır.

Önce dağıtım sonra toplama araç rotalama problemlerinde; depodan müşterilere dağıtılacak malzemelerin tamamı dağıtıldıktan sonra müşterilerden depoya gönderilecek malzemelerin toplama işlemi yapılır. Müşterilere birden fazla kez uğranabilir.

Karışık dağıtım toplamalı araç rotalama problemlerinde; dağıtım ve toplama işlemi karışık olarak yapılmaktadır. Müşterilere birden fazla kez uğranabilir.

Eş zamanlı dağıtım-toplamalı araç rotalama problemleri (EZDTARP); dağıtım ve toplama işlemlerinin eş zamanlı gerçekleştirildiği problemlerdir. Eş zamanlı ifadesinden anlatılmak istenen müşteriye uğrandığında, dağıtılacağı bırakılması ve toplanacağı alınmasıdır. Dolayısıyla müşteriler herhangi bir ayrıma tabi tutulmazlar. Araçlar her müşteriye bir defa gider ve dağıtım ve toplama işlemini yaparak müşteriden ayrılır.

EZDTARP'leri NP-zor problemlerdir. EZDTARP'ne; içeceklerin marketlere dağıtılırken boş şişelerin veya günü geçen ürünlerin fabrikaya taşınması, kanların merkezlerden hastanelere dağıtılırken yeni kanların da merkeze taşınması, kargo firmalarında ana depodan kolilerin bayilere dağıtılması ve bayilerden gönderilecek kolilerin ana depoya taşınması örnek olarak verilebilir.

EZDTARP ilk defa Min tarafından ortaya atılmıştır (Min 1989). Min' in algoritması önce kümeleme sonra rotalama yöntemini temel almıştır. Dethloff eş zamanlı dağıtım-toplamalı araç rotalama problemleri için ekleme temelli yeni bir sezgisel algoritma geliştirmiştir (Dethloff 2001). Geliştirilen algoritma ile Min tarafından sunulan algoritmanın bulunduğu rotalardan daha iyi bir sonuca ulaşılmıştır. Nagy ve Salhi ise dağıtım-toplamalı araç rotalama problemleri için genel bir algoritma geliştirmiş ve bu algoritmayı hem karışık dağıtım-toplamalı araç rotalama problemlerinde hem de eş zamanlı dağıtım-toplamalı araç rotalama problemlerine uygulamışlardır (Nagy ve Salhi 2005).

Geliştirdikleri algoritmayı Min'in algoritması ile kıyaslamışlar ve daha iyi sonuçlar verdiğini belirtmişlerdir. Montane ve Galvao tabu arama algoritması geliştirmiştir (Montane ve Galvao 2006). Gencer ve Yaşa, Ulaştırma Komutanlığına gelen yurtiçi taşıma isteklerinin en düşük maliyetle karşılanabilmesi için, eş zamanlı dağıtım toplamalı araç rotalama modeli oluşturmuşlardır (Gencer ve Yaşa 2007). Taleplerin değişken olmasından dolayı rotaların dinamik olarak belirlenmesini kolaylaştırmak için karar destek yazılımı(VRP 2.0) hazırlamışlardır. Bianchessi ve Righini EZDTARP için sezgisel algoritmalar geliştirmiş ve bu algoritmaları karşılaştırmışlardır (Bianchessi ve Righini 2009). Gajpal ve Abad EZDTARP için karınca koloni algoritması geliştirmiş ve mevcut yöntemlerden hem bilgisayar zamanı hem de çözüm kalitesi açısından daha iyi sonuçlar verdiğini göstermiştir (Gajpal ve Abad 2009). Ai ve Kachitvchyanukul EZDTARP için

yeni bir matematiksel model geliřtirmiş ve problemin çözümlü için kuş sürüsü algoritması önermiştir. Geliřtirilen matematiksel model literatürdeki 3 matematiksel modelin genişletilmiş halidir (Kachitvichyanukul 2009). Zachariadis ise EZDTARP için tabu arama ve bölgesel arama yöntemlerini birleřtiren bir hibrit yaklaşım önermiştir (Zachariadis 2009). Çetin vd., heterojen araç filolu eş zamanlı dağıtım toplamalı araç rotalama problemlerinin çözümlü için yeni bir sezgisel algoritma önermişler ve önerilen algoritmaya dayalı bir karar destek sistemi geliřtirmişlerdir (Çetin ve ark. 2005).

1.2 Çalışmanın Amacı

Çalışmanın amacı, ülkemizde faaliyet gösteren lojistik firmalarının veya her türlü sektörde bünyesinde lojistik birimi barındıran şirketlerin kısa (günlük, haftalık) ve uzun vadede (yıllık), hesaplamanın yapıldığı gün itibariyle, tedarikçilerden gerçekleştirilecek optimum (en uygun) taşıma organizasyonunun belirlenmesine yardımcı olmaktır.

Lojistik sektörü, günümüz koşullarında hız ve kaliteyle birlikte maliyet unsurunu ön plana çıkaran çalışmalarla fark yaratmaktadır. Artık herhangi bir ürünün istenilen zamanda ve kalitede ilgili müşterisine yetiřmemesi yeterli olmamakta, aynı zamanda bu hizmetin en az maliyetle yapılması hedeflenmektedir. Burada müşteri nihai anlamda olmakla birlikte, tedarik zinciri yapısı içerisinde birbiriyle tedarikçi-müşteri ilişkisine sahip her türlü birimi de ifade edebilir.

Örneğin, bir ürünün şirket/fabrika girişinde idari kabulünün yapılmasından sorumlu kişilerin müşterisi aynı fabrikada üretim bandı kenarında görev yapan kişiler olarak tanımlanabilir. Böylelikle fabrikaya girişten sorumlu kişiler en kısa zamanda, en kaliteli ve en tasarruflu biçimde bant kenarında görev yapan kişilere hizmet sunmakla yükümlüdür.

Bu çalışmada, fabrikaların üretime kıyasla ön safhasında yer alan üretim planlama ve nakliye faaliyetleri dikkate alınmaktadır. Üretim veya ihracat yapacak fabrikaya hammadde, yarı mamul veya mamul tedarik eden firmalardan gelen nakliye organizasyonlarının tedarikçi ve müşteri çalışma saatlerine sadık kalınarak en uygun biçimde gerçekleştirilmesi çalışmanın varmak istediğı hedefi ifade etmektedir.

1.3 Çalışmanın Kapsamı

Ülkemizin başlıca otomotiv üretim ana sanayi şirketlerinden Oyak-Renault Otomobil Fabrikaları A.Ş.'nin lojistik planlama departmanında yer alan nakliye birimleri için mevcut dönemde gerçekleştirilen mühendislik faaliyetlerinin gözden geçirilmiş ve bu çalışmanın katkı sağlayacağı tespit edilmiştir. Bu çalışma daha küçük hacimlerde ihtiyaca cevap vermekle birlikte, Oyak-Renault Otomobil Fabrikaları gibi kitle üretimi yapan bir firmanın tedarikçi nakliye optimizasyonuna da katkı sağlayabilmektedir.

Oyak-Renault Otomobil Fabrikaları A.Ş. yurtdışı ve yurtiçi olmak üzere toplamda yüzlerce tedarikçiye sahiptir. Bu tedarikçilerin yaklaşık 40 kadar adedi Bursa şehri sınırlarında yer almaktadır. Çalışma, fabrika ihtiyaçları ve yönetim yapısı itibariyle Bursa tedarikçileriyle sınırlandırılmıştır. Öte yandan yurtdışı tedarikçiler için benzer bir organizasyon zaten kurgulanmakta ve Fransa merkez birimler tarafından yönetilmektedir. Yurtdışı tedarikçiler ile ilgili uygulamalara Şekil 1'de yer alan nakliye biçimleri örnek olarak gösterilebilir.



Şekil 1: Yurtdışı tedarikçiler için gerçekleştirilen taşıma tiplerinin kartografisi

Platformla Toplama: Yurtdışında bulunan tedarikçilerden ürün/hammaddelerin toplanarak bir outsourcing yöntemiyle hizmet alınan bir stoklama merkezinde (cross-dock) optimizasyona gidilerek Bursa'daki fabrikaya getirilme yöntemidir. Bu sistemde konunun yönetimi ve organizasyonu bir lojistik firması tarafından bedeli karşılığı sağlanmaktadır. Bursa'da bu konu ile ilgili herhangi bir platform (toplama merkezi) aktivitesi söz konusu olmadığından bu çalışma şekli konu dışında kalmaktadır.

Tedarikçilerden Toplama: Tedarikçilerden ürün/hammaddelerin toplanarak doğrudan Bursa'daki fabrikaya getirilmesi işlemidir. Bu nakliye organizasyonu Bursa içi çalışma için mümkün olabilecek bir yöntemdir. Dolayısıyla çalışmanın ilerleyen safhalarında matematiksel modelin kararlarından biri olarak yer alacaktır.

Direkt akış: Tedarikçilerden müşteri fabrikaya doğrudan, herhangi başka bir tedarikçiye uğramadan gerçekleştirilen nakliye organizasyonudur. Tedarikçiden alınması planlanan ürün/hammadde hacimleri bu kararın alınmasında etkin rol oynamaktadır. Bu yöntem de çalışma kapsamında kendisine yer bulacak ve yurtdışında gerçekleştirilen bu uygulamanın yurt içinde de kullanılması, sistemlerin bu duruma göre parametre edilmesi sağlanacaktır.

Karma görünüm: Toplama ve Direkt akışa çalışma içinde yer verileceğinden karma görünüme sahip bir sonuç elde etmek mümkün olacaktır. Bu kartografiyle mevcut Bursa tedarikçilerinin tek bir şekilde hangi nakliye tipine sahip olacağı ortaya çıkmış olacaktır.

1.4 Araç Rotalama Problemlerinin Kavramsal Çerçevesi

Araç Rotalama Problemi (ARP), coğrafi olarak dağınık müşterilere bir veya birden fazla depodan hizmet vermek üzere görevlendirilen araçların optimum dağıtım/toplama rotalarının tasarlanması problemidir (Laporte ve ark. 1987). ARP dağıtım yönetiminin kalbidir. ARP'nin en basit şekline genel ARP denir. Genel ARP'nde, birinci şehir depo olmak üzere n adet şehir ve m adet araç vardır. Her bir aracın kapasitesi Q ve i düğümünden j düğümüne olan mesafe C_{ij} olarak tanımlanır. ARP ile m adet aracın rotası belirlenirken ;

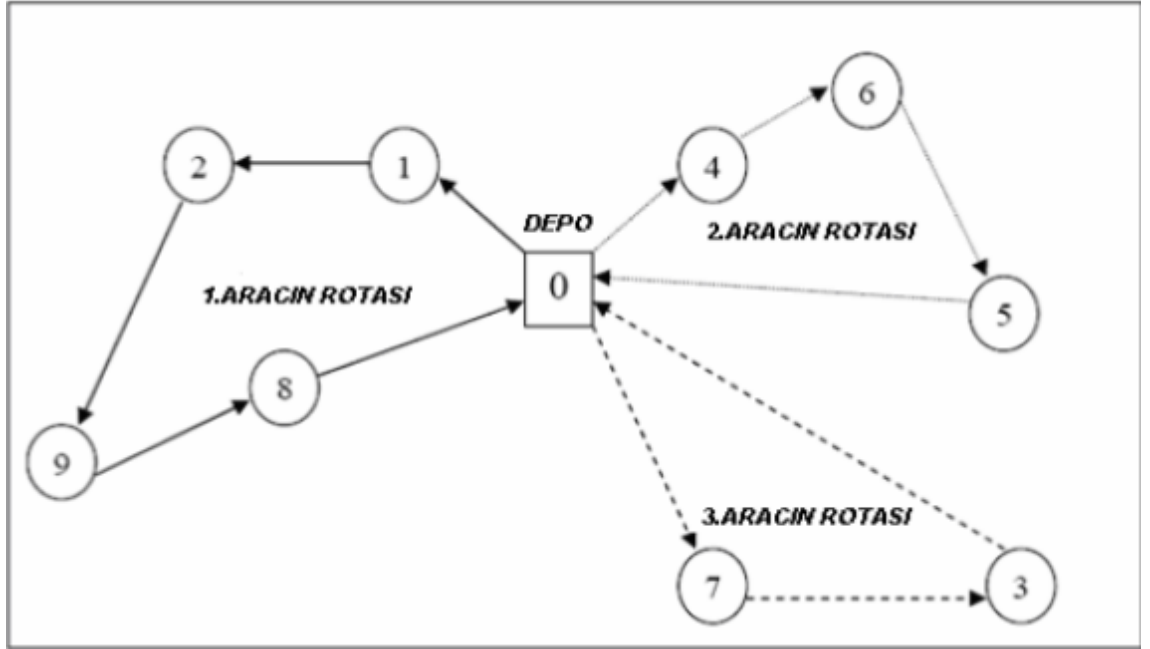
- Her bir şehir yalnız bir defa ziyaret edilir.
- Her bir araç rotasına aynı depoda başlar ve sonlandırır.
- Rota sayısı ve konfigürasyonu ile ilgili kısıtlar vardır.

Bu açıklanan temel kısıtlar haricindeki kısıtlar problemin özelliğine bağlı olarak değişmektedir.

2. KAYNAK ÖZETLERİ

2.1 Araç Rotalama Tanımı

Araç rotalama problemleri, bir müşteri serisi ne hizmet sağlayan araç grubunun, optimum rota serisine karar vermek için kullanılır. Bu çok önemli ve çalışılan bir kombinatoriyal optimizasyon problemlerindendir. Araç Rotalama Problemlerine 1959 yılında Dantzig ve Ramser tarafından bir giriş yapılmıştır. Benzinin benzin istasyonlarına teslimatı ile ilgili bir gerçek dünya uygulamasını gerçekleştirmiştir. Ayrıca bunun için ilk matematiksel programlama modelini ve algoritma yaklaşımını kurmuşlardır. Birkaç yıl sonra, 1964'te Clarke ve Wringht, Dantzig ve Ramser yaklaşımını geliştirerek probleme sezgisel bir çözüm önermişlerdir. Bu iki çalışmayı takiben, yüzlerce model ve algoritma, optimum ve optimuma yakın çözümlü araç rotalama probleminin farklı versiyonlarını sunmuşlardır.



Şekil 2: Araç Rotalama Örnek Şeması

ARP, merkezi bir depodan coğrafi olarak dağılmış çeşitli talep noktalarına dağıtım veya toplama rotalarının, araç filosunun kat ettiği toplam mesafeyi minimize edilecek şekilde dağıtım ve toplama rotalarının tasarlanması problemidir.

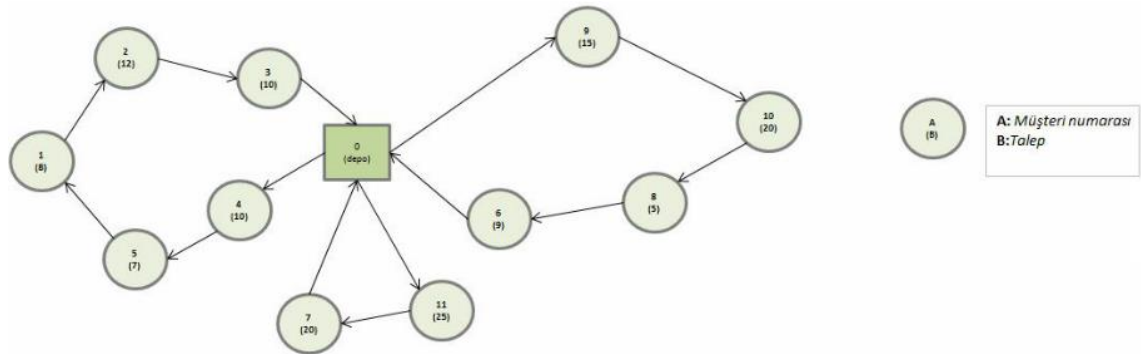
2.2 Araç Rotalama Problem Prensipleri

Literatürde çok sayıda araç rotalama problemleri ve bu problemlerin her biri için geliştirilmiş çok sayıda algoritma bulunmaktadır. Ancak bu algoritmaların hiç biri günlük yaşamda işletmeler için en optimum çözümü verememektedir.

Bundan dolayı araştırmacılar bugün hala araç rotalama problemlerinde en etkin ve iyi sonuçları verebilecek algoritmalar yaratmak için çalışmaktadırlar (Şeker 2007).

Araştırmacılar ve bu konunun uygulayıcıları daha başarılı bir rotalama için, aşağıdaki sekiz prensibin dikkate alınmasını önermişlerdir (Golden 2005).

- Noktalar içerisinde birbirine en yakın olanlar seçilmelidir. Bunun aracılığıyla toplam gidilen yol kısalmır.
- Farklı günlerdeki dağıtımlar birleştirilmelidir. Benzer noktadaki dağıtımlar birleştirilerek, aynı rotaların yakın tarihlerde tekrar gidilmesi engellenir.
- Rotalara, başlarken mümkün olan en uzak nokta seçilmelidir.
- Yapılacak olan rotalamaların şekli, gözyaşı şeklinde olmalıdır. Bu şekilde uzak noktalara ulaşmada kazanç elde edilebilecektir.
- Mümkün olan en yüksek kapasiteli araçlar kullanılmalıdır. Bu sayede toplam maliyetler azalacak ve avantaj elde edilecektir.
- Eğer yapılabiliyor ise dağıtım ve tedarik aynı araçlarla yapılmalıdır. Bu şekilde toplam maliyet ve bunun yanı sıra gereken zaman da azalacaktır.
- Rota dışındaki noktalara ulaşmada, küçük araçlar kullanılmalıdır.
- Eğer gerekiyor ise dağıtımların ve tedariklerin zamanları tekrardan kararlaştırılarak zaman tasarrufu sağlanmalıdır.



Şekil 3: Araç Rotalama Probleminin Şebeke Gösterimi

2.3 Araç Rotalama Temel Bileşenleri

ARP'yi oluşturan temel bileşenler; taşınacak malzemenin tipi, dağıtım/toplama noktaları, talep yapısı ve araç filosu olarak ifade edilebilir.

- Malzeme tipi: Araçlarla çok çeşitli malzemeler taşınır. Tehlikeli maddeler, gıda maddeleri, gazete dağıtımını bütün bunlar basit bölümlenmeler olarak adlandırılır ve problemde ek bir karışıklığa neden olmazlar (Bowerman, Hall ve Calomai 1995). Diğer taraftan öğrenci servisleri; güvenlik, etkinlik, eşitlik gibi ilave bazı amaçlardan ötürü daha karmaşık bir yapıya sahiptir. Tehlikeli maddeleri taşıyan araçların rotalarının belirlenmesinde ise coğrafi özellikler büyük önem kazanır.
- Dağıtım/toplama noktaları: Birçok ARP'nin de dağıtım noktaları müşterilerin bulunduğu yer, toplama noktaları ise depodur. Tüketim mallarının fabrikalardan toptancılara dağıtımını buna en iyi örnek olarak gösterilebilir.
Depo genelde aracın rotasına başladığı ve döndüğü noktadır. Depo sayısına göre problem, tek depolu ve çok depolu olarak adlandırılabilir. Çok depolu problemlerde, depoların her biri kendi araçlarıyla kendi işini yürütebilir, bu durumda problem birkaç bağımsız tek depolu ARP'ye dönüşür. Araç bir depondan çıkıp birden fazla depoda yükleme/boşaltma aktivitesi yapabilir. Bu durumda problem bir bütün olarak ele alınmalıdır.
- Araç filosu: Bütün ARP'lerin de araçların kapasitesinin bilindiği ve çoğunlukla araçların homojen olduğu varsayılır. Filo heterojen ise filodaki araçların taşıma kapasiteleri farklıdır. Bu durum hangi araç tipinin, hangi rotaya hizmet vereceğinin belirlenmesi ilave bir kararı gerektirir. Araçların diğer özellikleri arasında hız, yakıt tüketimi, taşınacak malzemeye uygunluğu sayılabilir. Bu özelliklerin rotalamaya doğrudan katkısı yoktur.
- Talep yapısı: ARP problemlerinde talep statik veya dinamik olabilir. Statik talep durumunda talep önceden bilinir. Dinamik durumda ise bazı düğümlerdeki talep bilinmekte bazıları ise araç rotasında devam ederken belirli olmaktadır (Savelsbergh 1995).

2.4 Araç Rotalama Çeşitleri

Araç rotalama problemleri değişik parametrelere ve bu parametrelerin etki alanlarına göre çeşitli gruplara ayrılmaktadır. Araç rotalama problemleri en genel hatlarla; müşteri taleplerinin stokastik veya deterministik olmasına, aracın başlangıç noktana dönüp dönmemesine, toplam zamana veya toplam mesafeye, zaman penceresine veya araç kapasitesine, gidiş-dönüş mesafelerinin eşit olup olmadığı gibi kısıtlara göre çeşitli gruplara ayrılır.

2.4.1 Dinamik ve Statik Çevre Durumuna Göre

Statik çevre durumuna göre olan araç rotalama problemlerinde, problem çözülmeden önce problem hakkındaki kısıtlar, talepler, kapasiteler, maliyet bilgileri gibi gerekli olan değişkenler bilinmektedir ve bu bilgiler problemin çözüm aşamasında da sabittirler. yapılan çalışmalarda da çoğunlukla Statik Araç Rotalama Problemi (SARP) üzerinde çalışmalar yapılmış olan bu problem, deterministik ARP olarak da karşımıza çıkmaktadır. Gerçek hayatta SARP çözüm yöntemleri, önceden miktarı ve zamanı bilinen talepler için rota planları oluşturmakta ve servis sistemlerinin genel olarak değerlendirilmesinde kullanılmaktadır.

2.4.2 Rotalama Durumuna Göre

Araç rotalama problemleri, bir aracın bir işletme birimi için çalışması diğer bir ifade ile rotaların bir işletme biriminden başlayarak aynı işletme biriminde sona ermesi veya aracın işletme birimlerinden bağımsız olarak seyir güzergahının en son müşteride bitirilebilmesi durumlarına göre açık ve kapalı uçlu olmak üzere iki farklı şekilde incelenir (Erol, 2006).

Açık uçlu araç rotalama problemleri, onlarca yıl önce öne sürülmüştür ama hala araştırmacılar tarafından tatmin edici bir çözüm için çok az bir ilgi gösterilmiştir. Açık uçlu araç rotalama problemi belirli bir talep ve bilinen coğrafi konumlar ile bir dizi müşteriye hizmet eden araç filosu için rotaların tanımlanmasıdır.

Her rota bir dizi müşteridir. Bu rota depo ile başlar ve müşteri ile biter. Bu tür problemler genellikle araçların kiralandığı sistemlerde uygulanır. Araç sadece gidiş için kiralanarak dönüş planlaması yapılmaz (Tüfekçier, 2008).

Kapalı uçlu araç rotalama problemlerinde, her rota bir işletme biriminden başlatılıp, aynı işletme biriminde bitirilmelidir. Literatürdeki çalışmalar genellikle kapalı uçlu araç rotalama problemleri ile yapılmakta olup, test problemleri kapalı uçlu araç rotalama problemi için mevcut olduğundan geliştirilen yöntemler kapalı uçlu araç rotalama problemi üzerinden birbirleriyle kıyaslanmaktadır (Erol, 2006).

2.4.3 Kısıtlarına Göre

Araç rotalama problemlerinde, gerçek hayatta kurulan lojistik sistemlerin tamamen modellenip çözümlenmesi zor olduğundan dolayı belirli bazı önemli kısıtlar seçilip diğer unsurlar göz ardı edilerek optimum sonuç bulunmaya çalışılmıştır. Bu kısıtlardan önemlileri araç yükleme, mesafe kısıdı, zaman pencereleri ve yükleme boşaltma durumlarıdır.

Bundan dolayı araç rotalama problemleri göz önüne alınan kısıtların isimleri ile beraber anılmakta, kısıtlarla araç rotalama problemleri türlere ayrılmaktadır (Erol, 2006).

Kapasite kısıtlı araç rotalama problemleri; bir serim üzerinde tüm araçlar tarafından kat edilen toplam mesafeyi minimum yapan ve her düğüme bir defa uğranılacak şekilde, merkez düğüme bağlı, kapasite kısıtlarını aşmayan m adet rotanın belirlenmesi problemidir (Tüfekçier, 2008). Kapasite kısıtlı araç rotalama probleminde, tüm müşteriler teslimatlar ve deterministik taleplere karşılık gelir, önceden bilinir ve parçalanamaz. Araçlar birbirleriyle aynıdır, tek bir merkez depoya bağlıdır ve yüklenen araçlar için sadece kapasiteler sınırlandırılır. Bunun amacı tüm müşterilere hizmet verebilmek için toplam maliyeti (ağırlıklı işlevi olan rota sayısı, rota uzunluğu ve seyahat süresi) en aza indirmektir.

2.5 Araç Rotalama Problemleri Çözüm Yöntemleri

İşletmelerin dağıtım kanallarında rotalama yapılması sırasında ortaya çıkan ve bu kapsamdaki maliyetleri önemli ölçüde belirleyen, sıralama ve gruplamanın yapıldığı NP-hard kombinatoriyal optimizasyon problemidir. Bu tür problemlerin çözüm yöntemleri, kesin ve yaklaşık yöntemler olmak üzere iki gruba ayrılmıştır.

Problemin NP-hard oluşu ve problem boyutunun artmasıyla birlikte çözüm süresinin üstel olarak artması sonucu, çözüm için yapılan çalışmaları daha çok yaklaşık yöntemler üzerinde yoğunlaşılmasına sebep olmuştur. Yaklaşık yöntemler, büyük boyutlu problemlerde daha az işlemle ve kısa hesaplama süresiyle optimuma yakın, iyi kalitede çözümler üretmektedir.

Yaklaşık yöntemler, klasik sezgisel ve meta sezgisel yöntemler olmak üzere ikiye ayrılmaktadır. Klasik sezgisel yöntemler orta kalitede çözümlere hızlı bir şekilde ulaşmaktadır ve bu yöntemlerin en önemlileri içerisinde Tasarruf, Süpürge ve Taç Yaprığı yöntemleri bulunmaktadır. Araç rotalama problemleri için en etkili sonuçları veren algoritmalar ise meta sezgisel algoritmalarıdır. Klasik sezgisel algoritmalara göre işlem süresi daha uzun olsa da hafıza tabanlı algoritmalarından Tabu Arama ve Adaptif Hafıza yöntemleri, ARP için en iyi çözümleri bulan tekniklerdir (Erol, 2006).

2.5.1 Kesin Yöntemler

Kesin yöntemler aracılığıyla en uygun sonuçlar bulunmaktadır. Ancak özellikle büyük ölçekli problemlerin çözümünde çözüm zamanı çok uzun olabilmektedir. Kesin yöntemler matematiksel programlama tabanlı yöntemler olup, ortak özellikleri en uygun sonucu vermeleridir.

2.5.1.1 Dal ve Sınır Algoritması:

Dal ve sınır algoritması çözüm uzayını alt problemlere ayırmak ve daha sonra her bir alt problemi ayrı ayrı optimize etmek için böl ve ele geçir stratejisini kullanır (Anonim 2015).

Dal ve sınır algoritması ile çözülecek problem bir minimizasyon problemi olarak ortaya atılır. Dal ve sınır algoritmaları, bir ağaç olarak canlandırılan durum uzayında en düşük maliyetli çözüm düğümünü arayan bir araştırma içine girer.

Dal ve sınır algoritmasını, iki temel fikirden yola çıkar. Bunlardan birincisi, arama alanı olarak bir arama ağacı formu düzenlemektir. Fikirleri düzeltmek için, ne kullanacağımızı gösteren bir ikili arama ağacına karar veririz. Ağacın her düğümü problemin bir değişkenine tekabül eder ve düğümün dışındaki iki dal değişkenin reddedilmesi veya kabul edilmesine tekabül eder.

2.5.1.2 Dal ve Kesme Algoritması:

Dal ve kesme algoritması tamsayı doğrusal programların çözümü için kombinatoriyal optimizasyon yöntemidir. Bu algoritma dal ve sınır algoritmasının bir türevidir (Anonim 2015).

Bu metot tam sayı kısıdı olmayan doğrusal programı, düzenli simpleks algoritma kullanarak çözer. Optimum bir çözüm elde edildiğinde ve bu çözüm tam sayı varsayılan bir değişken için tam sayı olamayan değer içerdiğinde, tüm uygun tamsayı noktalar tarafından tatmin edilen ama mevcut kesirli çözüm tarafından bozulan daha ilerdeki lineer kısıtları bulmak için kesme düzlemi algoritması kullanılır. Eğer böyle bir eşitsizlik bulunursa, buna lineer programa eklenir ve böylece “daha az kesirli” farklı bir sonuç bulunabilir.

Bu proses tam sayılı çözüm bulunana veya artık kesme düzlemi bulunmayana kadar tekrar eder (Anonim 2015).

2.5.1.3 Küme Kapsama Algoritması

Küme kapsama problemi, literatürde genellikle örtüleme problemi “covering problem”, “le probleme de recouvrement” gibi isimlerle yer almaktadır.

Küme örtüleme problemleri 0-1 tam sayılı programlama modelinin özel bir hâlidir. Bu problemler, gerçek dünya problemleri olarak oldukça fazla uygulama alanı bulmaktadır. Dağıtım ve rotalama problemleri, plânlama problemleri ve yerleştirme problemleri sık sık küme örtüleme yapısında karşımıza çıkmaktadır. Bu yapı sayesinde yerleşim, araç ya da insanlar tarafından her bir müşterinin ihtiyacını karşılayabilir (Gencer 2007).

2.5.2 Klasik Sezgisel Yöntemler

Araç rotalama problemleri için birkaç sezgisel yöntem önerilmiştir. Sezgisel yöntemler daha çok 1960 ve 1990 yılları arasında gelişmiştir. Günümüzde en çok kullanılan yapı ve geliştirilen prosedürdür. Bu yöntemler arama alanı oldukça sınırlı bir arama gerçekleştirirler ve genellikle az bir hesaplama süresi içinde iyi kalitede çözümler üretilir. Bununla birlikte, bu yöntemlerin hesaplamalarının çoğu, kolayca gerçek hayatta karşılaşılan çeşitli kısıtlar için uzatılabilir. Bu nedenle bu yöntemler hala ticari paketlerde yaygın olarak kullanılır (Toth ve Vigo 2002).

Sezgisel yöntemler üç kategori altında sınıflandırılır. Yapısal sezgisel yöntemler, çözüm maliyetleri göz önüne alınarak kademeli olarak uygun bir çözüm oluşturur fakat kendi başına bir gelişme aşaması içermez. İki aşamalı sezgisel yöntem de, iki doğal bileşene ayrılır. Bunlar, bu iki aşama arasındaki olası geri besleme döngüsü ile birlikte, uygun rotalardaki köşeleri kümelenme ve güncel rota yapısıdır. İki aşamalı sezgisel yöntem, önce grupla sonra rotala ve önce rotala sonra grupla olmak üzere iki sınıfa ayrılmıştır. İlk durum olan önce grupla sonra rotala yönteminde, uygun gruplamalarda ilk organize edilen köşelerdir ve bir araç rotası köşelerinin her biri için inşa edilir. İkinci durum olan önce rotala sonra grupla yönteminde, ilk olarak bütün araçlar için bir rota inşa edilir ve daha sonra uygun araç rotaları bölümlere ayrılır. Sonuncusu da iyileştirmeli sezgisel yöntemlerdir. Bu yöntemler, herhangi bir uygun çözümü araç rotaları içinde veya arasında bir dizi kenar, köşe değişikliği oluşturarak iyileştirmeye yöneliktir (Toth ve Vigo 2002).

Sezgisel yöntemler içerisindeki başlıca algoritmaları şu şekilde listeleyebiliriz (Toth ve Vigo 2002) ;

Yapısal sezgisel yöntemler

- Clarke ve Wright Tasarruf Algoritması
- Clarke ve Wright Gelişmiş Tasarruf Algoritması
- Eşleme Tabanlı Tasarruf Algoritması
- Sıralı Ekleme Sezgisel Algoritması
 - Mole ve Jameson Sıralı Ekleme Sezgiseli
 - Cristofides, Mingozzi ve Toth Sıralı Ekleme Sezgiseli

İki aşamalı sezgisel yöntemler

- Önce Grupla Sonra Rotala
 - Basit Kümeleme Yöntemi
 - Süpürge Algoritması
 - Fisher ve Jaikumar Algoritması
 - Bramel ve Simch-Levi Algoritması
 - Budanmış Dal ve Sınır Algoritması
 - Taç Yaprağı Algoritması
- Önce Rotala Sonra Grupla Algoritması

İyileştirmeli sezgisel yöntemler

- Tek Rota İyileştirmeli Sezgisel Algoritması
- Çok Rota İyileştirmeli Sezgisel Algoritması

2.5.2.1 Clarke ve Wright Tasarruf Algoritması

Clarke ve Wright tasarruf algoritması araç rotalama problemleri içinde en iyi bilinen sezgisel yöntemlerden biridir. Bu yöntem Clarke ve Wright tarafından 1964 yılında geliştirilmiştir. Araç sayısı sabit olmayan problemler için uygulandığı gibi rotası belli olan veya olmayan problemler içinde iyi sonuçlar verir. İki güzergah $(0, \dots, i, 0)$, $(0, j, \dots, 0)$ uygun bir şekilde tek bir rotada $(0, \dots, i, j, \dots, 0)$ birleştirildiği zaman bir mesafe kazancı $(s_{ij} = c_{i0} + c_{0j} - c_{ij})$ oluşturulur.

Bu algoritma ařağıdaki biçimde çalışır;

- Adım 0: Tasarrufları Hesaplama
- Adım 1: En Uygun Birleştirme (Paralel Sürüm)
- Adım 2: Rota Uzatma (Ardışık Versiyon)

2.5.2.2 Clarke ve Wright Gelişmiş Tasarruf Algoritması

Orijinal Clarke and Wright tasarruf algoritmasının dezavantajı, iyi rotalar üretmesine rağmen bazı çevresel yollarda dâhil olmak üzere rotaların sonlarına doğru bunu koruyamamasıdır. Bunu düzeltebilmek amacıyla Gaskell (1967) ve Yellow (1970) tarafından tasarruf hesaplama fonksiyonu $s_{ij} = c_{i0} + c_{0j} - \lambda c_{ij}$ olarak değiştirilmiş ve bu sayede köşeler arasındaki mesafe λ terimi ile ağırlandırılarak daha etkin bir tasarruf değeri elde edilmiştir

2.5.2.3 Sıralı Ekleme Sezgisel Algoritması

Sıralı Ekleme Sezgisel Algoritması, araç sayısının değişken olarak alındığı fakat yükleme kapasitesi bilinen problemler için geliştirilmiş bir yöntemdir. Burada süreç genel olarak başlangıç rotalarının oluşturulması ve bu rotalarla ilgili talep noktalarının rotaya en az maliyet artışına sebebiyet verecek şekilde eklenmesine dayanmaktadır. Bu konuyla ilgili iki çalışma yer alacaktır. Bunlardan 1976'da Mole ve Jameson tarafından geliştirilen model, bir iterasyonda sadece bir rota ele alınmaktadır. Christofides, Mingozzi ve Toth ise 1979'da bu yöntemi sıralı ve paralel rota oluşturan yordamlar kullanarak uygulamışlardır (Erol 2006).

2.5.2.3.1 Cristofides, Mingozzi ve Toth Sıralı Ekleme Sezgiseli

Cristofides, Mingozzi ve Toth kısmen daha karmaşık olan ve ayrıca kullanıcının kontrolü altında iki parametrelili (λ , μ), iki aşamalı ekleme sezgiselini geliştirmiştir. Bu algoritmanın genel anlamda aşamaları şöyledir;

- Adım 1: İlk Rotalama
- Adım 2: Ek Maliyetler

Herhangi bir rotalanmamış i_k noktası, k rotasına ilk değerini atamak için seçilir.
- Adım 3: Tepe Noktanın Eklenmesi

k noktasına eklenmesi uygun olan rotalanmamış noktaların grubu olarak S_k 'da $S_{i^*} = \min_{i \in S}$ bulunur. k rotasına i^* tepe noktası eklenir. 3-opt algoritması kullanılarak k rotası optimize edilir. k rotasına atanabilir noktalar kalmayana kadar bu adım tekrarlanır.
- Adım 4: Gelecek Rota

Eğer bütün noktalar rotaya eklendiyse durulur. Aksi takdirde $k_i = k+1$ olarak ayarlanır ve adım 2'ye geri dönülür.
- Adım 5: Rotayı Başlatma
- Adım 6: Maliyetleri Birleştirme
- Adım 7: Maliyetleri Ekleme
- Adım 8: Tepe Noktalarının Eklenmesi
- Adım 9: Sonucun Kontrolü

2.5.2.3.2 Mole ve Jameson Sıralı Ekleme Sezgiseli

Mole ve Jameson, bir rotanın alt yapısına kadar ulaşmak için λ ve μ parametrelerini kullanır. Algoritmanın genel anlamda aşamaları şöyledir;

- Adım 1: Rota Başlangıcını Oluşturma
- Adım 2: Gelecek Nokta
- Adım 3: Rota Optimizasyonu

2.5.2.4 Eşleme Tabanlı Tasarruf Algoritması

Desrochers ve Verhoog'un 1989 yılında ve Altinkemer ve Gavish'in 1991 yılında yaptıkları benzer açıklamalar, bu standart tasarruf algoritmasında yapılan ilginç bir değişikliktir. S_k , k rotasına ait nokta kümesi ve tS_k bu noktalara ait Gezgin Satıcı Probleminin (GSP) optimum çözümü ise her adımdaki tasarruf miktarı p ve q rotalarının

birleştirilmesi $S_{pq} = t(S_p) + t(S_q) - t(S_p \cup S_q)$ ile bulunmaktadır. Eşleme tabanlı tasarruf algoritmasına göre rotalar S_{pq} değerlerine göre büyükten küçüğe sıralanarak kapasite kısıdı da dikkate alınıp uygun bir şekilde birleştirilmektedir. Bu temel algoritmanın olası bir şekilde $t(S_k)$ değerleri yerine tam olarak onları hesaplama yaklaşımını da içermektedir (Anonim 2015).

Başka bir eşleme tabanlı tasarruf yaklaşımı Wark ve Holt tarafından açıklanmıştır. Buna göre eşleme ağırlıkları olarak s tasarruf miktarı değeri alınmakta fakat bazı durumlarda belirli bir olasılığa göre rotalar parçalara da bölünebilmekte veya birleştirilebilmektedir.

2.5.2.5 Basit Kümeleme Yöntemi

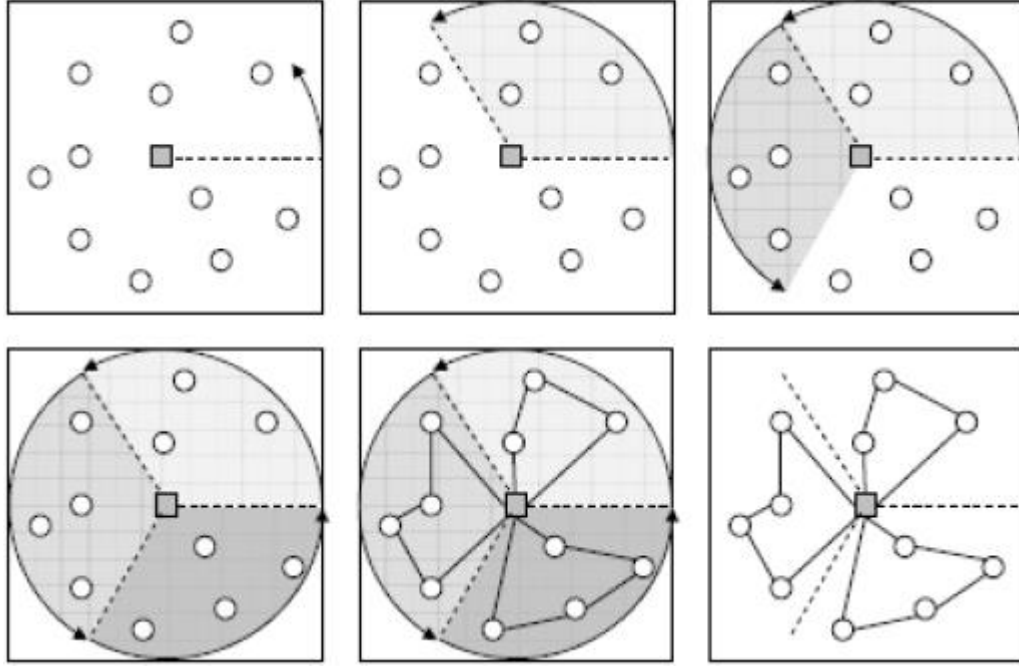
Burada üç basit kümeleme yöntemi mevcuttur. Bunlar süpürge algoritması, Fisher ve Jaikumar Algoritması ve Bramel ve Simchi-Levi algoritmasıdır (Toth ve Vigo 2002).

2.5.2.5.1 Süpürge Algoritması

Bu algoritma ARP'nin düzlemsel örneklerine uygulanır. Uygun kümeler ilk olarak depoyu merkez alan ışının ekseni üzerinde dönerek şekillenir. Bir araç rotası her küme için bir GSP çözümlenerek elde edilir. Bu algoritma ilk olarak 1971 yılında Wren'in kitabına geçmektedir fakat bu konuyu Gillett ve Miller (1974) popüler hale getirmiştir. Bu metodun basitleştirilmiş yürütme sistemi şöyledir:

Her i tepe noktası polar koordinat sisteminde (Q_i, r_i) , ile gösterilir. Burada Q_i açığı ve r_i ışın uzunluğunu göstermektedir. Keyfi olarak seçilen i^* noktasına $Q_{i^*} = 0$ değeri atanır ve merkezi $(0, i^*)$ ilk ışından 0 da konuşlanan geri kalan açığı hesaplar. Kenarları ve açıları Q_i ,nin artacağı şekilde sıralanır. Algoritmanın adımları şu şekildedir (Toth ve Vigo 2002);

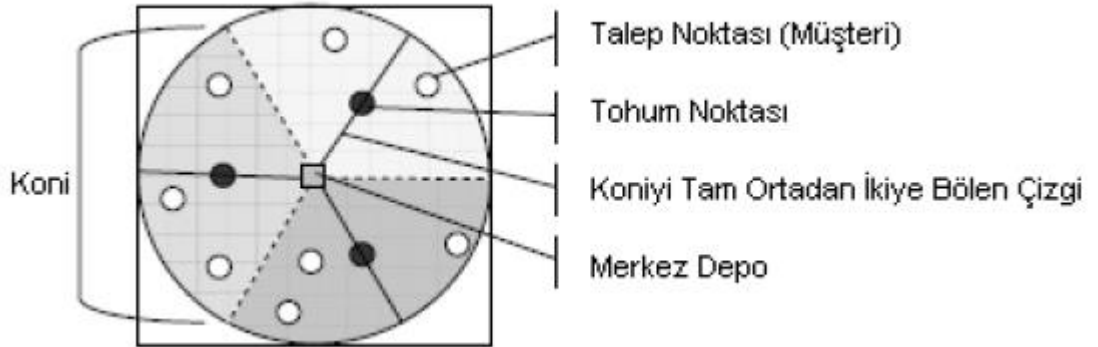
- Adım 1: Rota Sıfırlama
- Adım 2: Rota yapma
- Adım 3: Rota Optimizasyonu



Şekil 4: Süpürge Algoritması Adımları (Erol 2006).

2.5.2.5.2 Fisher ve Jaikumar Algoritması

Fisher ve Jaikumar algoritması (1981), iyi bilinen algoritmadır. Grupları oluşturmak için bir Genelleştirilmiş Atama Problemi (GAP) çözülür. k araç sayısı sabittir.



Şekil 5: Fisher ve Jaikumar Algoritması İçin Kurulan Yapı (Erol 2006).

Algoritmanın adımları genel olarak aşağıdaki gibi açıklanabilir (Toth ve Vigo, 2002);

- Adım 1: Çekirdeğin Seçilmesi

Problemin yer aldığı düzlemi, tepe noktaları depo olacak şekilde müşteri talep miktarları ve kapasite kısıtları dikkate alarak k adet koniye bölünür. Her bir koni için koniyi tam ortadan bölen bir doğru parçasının ortasında yer alan sanal j_k tohum noktası belirlenir ($k=1,\dots,k$) (j_k tohum noktası k konisi için maliyet hesaplamada kullanılacaktır).

- Adım 2: Müşterilerin Çekirdeklere Atanması

Her i müşterisi ve her k grubu için, grubun çekirdeği ile ilgili olarak bir yerleştirme maliyeti d_{ik} hesaplanır.

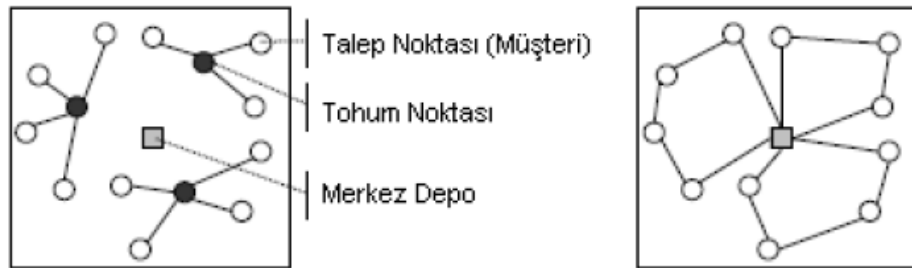
- Adım 3: Genelleştirilmiş Atama

Genel Atama Problemini d_{ij} maliyeti, q_i talep miktarı ve Q araç kapasite değerine göre çözülür ve gruplar tamamlanır

- Adım 4: GSP Çözümü

2.5.2.5.3 Bramel ve Simchi-Levi Algoritması

Bramel ve Simichi-Levi lokasyon iki aşamadan oluşan sezgisel olarak tanımlamıştır. Birinci aşamada, tohum noktalarının yeri, Kapasite Kısıtlı Çoklu Tesis Lokasyonu Belirleme Problemi ile çözülerek bulunmaktadır. İkinci aşamada kalan noktalar yavaş yavaş ayrılan rotaların içine dâhil edilir (Toth ve Vigo 2002).



Şekil 6: Bramel ve Simichi-Levi Algoritması Örneği (Erol 2008).

Algoritmanın işleyiş adımları genel anlamda aşağıda belirtilmektedir;

- Adım 1: Toplam taleplerin C araç kapasitesini geçmeyecek şekilde müşteriler gruplanarak n adet tohum noktasına bağlanırlar. Tohum noktalarının

koordinatları, oluşturdukları grup içinde yer alan müşterilere toplam mesafesinin minimum olması sağlanır.

- Adım 2: İlave olarak merkez depo da eklenir ve bir rota oluşturulur. Aracın müşterilere ziyaret sırası, ekleme mantığı ile belirlenir. Bir k grubunda yer alan tüm noktalar $T_k = \{0, i_1, \dots, i_l\}$ ise rastgele bir nokta seçilerek başlangıç rotası oluşturulur. Daha sonra, aşağıdaki maliyet fonksiyonlarından biri seçilerek rotaya eklenecek talep noktası i 'nin sırası daha önce rotalanmış noktalar ile belirlenir.

2.5.2.6 Budanmış Dal ve Sınır Algoritması

1976 yılında Chritofides, Mingozi ve Toth tarafından basit bir arama ağacı yapısı kullanan bu algoritmayı geliştirmiştir. Arama ağacının her seviyesinde tek bir dal bulunmakta, her iterasyonda rotalanmamış bir müşteri seçilerek onun içinde yer alması mümkün rotalar üzerinden sonuç bulunmaya çalışılmaktadır. Budanmış Dal-Sınır algoritmasının adımları genel olarak aşağıdaki gibidir (Erol 2006).

- Adım 1: Rota indeksi değişkeni $h = 1$ atanır. Rotalanmamış nokta kümesi $F_h = V \setminus \{0\}$ olarak düzenlenir.
- Adım 2: Eğer $F_h = \emptyset$ ise algoritma durdurulur. Aksi takdirde kapasite kısıdı da dikkate alınarak F_h kümesinden rotalanmamış bir i noktası seçilir. Tasarruf ve ekleme maliyeti fonksiyonlarının lineer kombinasyonu kullanılarak i müşterisinin içinde yer aldığı R_i rota kümesi oluşturulur.
- Adım 3: Bütün r rotaları için ki bu rotaların hepsi R_i 'de yer alır. $f(r) = t(S_r \cup \{0\}) + u(F_h \setminus S_r)$ değerleri hesaplanır. Burada S_r , r rotasında yer alan noktalar kümesi, $t(S_r \cup \{0\})$ ifadesi $S_r \cup \{0\}$ nokta grubundan oluşan GSP'nin iyi bir çözüm değerini vermektedir. Aynı zamanda $u(F_h \setminus S_r)$ ise rotalanmamış müşteriler için en kısa arama ağacı uzunluğunu vermektedir.
- Adım 4: R_i rota kümesi içerisinde $\min_{r \in R_i} \{ f(r) \}$ sağlayan r^* rotası seçilir. Rota indeksi h değeri bir artırılır ve rotalanmamış müşteri kümesi $F_h = F_{h-1} \setminus S_{r^*}$ olarak güncellenir ve adım 2'ye gidilir.

2.5.2.7 Taç Yaprığı Algoritması

Ryan, Hjorring ve Glover 1993 yılında süpürme algoritmasının doğal bir uzantısı olan bu yöntemi taç yaprağı olarak isimlendirdi. Burada taç yaprakları denilen birçok rota oluşturulur ve aşağıdaki alt problem kümesi çözülerek son bir seçim yapılır

2.5.2.8 Önce Rotala Sonra Grupla Algoritması

Bu grupta yer alan metotlar ilk aşamada çevre kısıtlarını dikkate almadan büyük bir Gezgin Satıcı Problemi turu oluştururlar ve ikinci bir aşamada bu turu uygun araç rotalarına ayırırlar. Bu yaklaşım araç sayısının sınırlı olmadığı problemlere uygulanır. Bu ilk olarak, ikinci aşamadaki problemin standart bir en kısa yol problemi olduğunu ve $O(n^2)$ zamanında çözülebileceğini araştırmış olan Beasley tarafından öne sürülmüştür. En kısa yol probleminde i ve j düğümleri arasındaki dolaşım maliyeti d_{ij}' nin, $c_{0i} + c_{0j} + l_{ij}$ ye eşittir (l_{ij} , gezgin satıcı problemi turunda i 'den j 'ye gitme maliyetidir.). Önce rotala sonra grupla sezgiselinin diğer yaklaşımlarla rekabet edebilir özellikte olduğunu kanıtlayan bulgulara mevcut değildir.

2.5.2.9 Tek ve Çok Rotalı İyileştirmeli Sezgisel Algoritması

Araç rotalama problemleri içinde en çok geliştirilen yöntemler Lin'in λ -opt metodu olarak açıklanabilir. Bu yöntemde λ adet yol bir rotadan çıkarılarak mümkün olan permütasyonlar da rotanın çeşitli noktalarına eklenmektedir. Mevcut çözümden daha iyi bir çözüm bulunması halinde yöntem, bu yeni rotayı çıktı olarak vermektedir. $O(n^\lambda)$ süresi içerisinde bir çözümün λ -opt kontrolü oluşturulabilir.

Genel olarak bir ARP'de λ -opt algoritması $O(n^2)$ sürede sonuç bulması beklenmektedir. λ -opt yöntemi üzerinde değişiklikler yapılarak örneğin art arda gelen belli sayıda nokta yer değiştirilerek bu süre kısaltılabilmektedir (Toth ve Vigo 2002).

İyileştirme algoritmaları uygun olan bir çözümün kenar ve düğümleri araç rotası içinde veya araç rotaları arasında birbirleri ile değiştirilerek uygun bir çözümün aranmasıdır. ARP

için çok rotalı iyileştirme sezgiselleri, çok rotanın bulunduğu problemlerde kullanılırken her bir rotada aynı anda işlem yaparlar.

2.5.3 Meta sezgisel Yöntemler

Son zamanlarda araç rotalama problemi için birçok meta sezgisel yöntem önerilmiştir. Araç rotalama problemlerine uygulanabilen meta sezgisel yöntemler altı şekildedir. Bunlar; Benzetimli Tavlama, Deterministik Tavlama, Genetik Algoritmalar, Karınca Algoritması, Yapay Sinir Ağları ve Tabu Aramadır.

2.5.3.1 Benzetimli Tavlama

Benzetimli Tavlama ismi demirin tavlanmasıyla gelmektedir. Demirin çok yüksek sıcaklıklarda çok yüksek enerjilere sahip olabilen bir metal olma özelliği ve aynı zamanda işlenen demirin sıcaklığı nispeten düşük olmasına karşın enerjisi yüksek durumlarla karşılaşılabilir. Benzetimli tavlama ise amaç fonksiyon değeri daha önce bulunanlardan yüksek olmasına karşın daha iyi sonuçlar veren çözümlerle karşılaşılabilir.

Benzetimli tavlama, çözüm kalitesizliğine neden olan lokal optimumlardan kaçınmayı sağlayan bir tırmanma işleminin temel dinamiklerine izin vermeye çalışan bir yöntem olarak algılanabilir. $D=f(x)-f(x_t)$ iken $D \leq 0$ ise S' çözümü yeni çözüm olarak kabul edilir. Aramanın lokal optimumdan uzaklaşması için, $D>0$ ise amaç değerini arttıran hareketler $e^{-d/t}$ olasılığıyla kabul edilirler.

Burada T sıcaklık parametresidir. T'nin değeri çok yüksek bir değerden 0'a yakın düşük bir değere kadar değişkenlik gösterir. Bu değerler soğutma çizelgesi ile kontrol edilirler. Tavlama benzetiminin t. iterasyonun da $N(x)$ arasından bir x çözümü rastsal olarak seçilir. $f(x)-f(x_t)$ ise x_{t+1} x'e eşit kabul edilir. Aksi halde; $x_{t+1} = x$, p_t olasılığıyla, x_t , $1 - p_t$ olasılığıyla p_t genellikle t'nin ve $f(x)-f(x_t)$ 'nin azalan bir fonksiyonudur. p_t genellikle $e^{-d/t}$ olarak tanımlanır.

2.5.3.2 Deterministik Tavlama

Deterministik Tavlama, bir hareketin kabulü için deterministik kuralın kullanımı dışında Benzetimli Tavlamaya benzer bir şekilde çalışır. Bu tekniğin iki standart uygulaması sınır kabulü ve seyahat kayıtlarının tutulmasıdır. Bir sınır kabul algoritmasının t iterasyonunda, eğer $f(x_{t+1}) < f(x_t) + Q_1$ ise çözüm x_{t+1} 'dir. Burada Q_1 kullanıcı kontrol parametresidir.

Seyahat kayıtlarının tutulmasında, bir kayıt arama boyunca x^* en iyi kayıttır. t iterasyonda, eğer $f(x_{t+1}) < Q_2 f(x_t)$ ise çözüm x_{t+1} 'dir. Burada Q_2 genellikle 1'den biraz büyük kullanıcı kontrol parametresidir (Toth ve Vigo 2002).

2.5.3.3 Genetik Algoritmalar

Sezgisel olarak kullanılacak arama tekniklerinden biride Genetik Algoritmalarıdır. Holland'ın 1960'lı yılların sonlarına doğru yaptığı çalışmalar bu algoritmanın temelini oluşturur. 1975 yılında öğrenebilen makinelerin tasarımını araştıran Holland öğrenme işleminin; tek bir organizmanın yanı sıra türlerin nesiller boyunca evrimsel uyumu ile gerçekleştiğini fark eder. Genetik algoritmaların gelişim süreci buna dayanmaktadır. Bu algoritmalar, hayatta kalabilen ve özelliklerini yeni nesillere aktarabilen organizmaların davranışlarını taklit etmektedirler. Bu sürecin taklit edilmesinin nedeni, iyi çözümleri çoğaltarak bireylerin güçlü özelliklerinin alınmasıyla daha iyi sonuçlar elde etmektir (Tüfekçiler 2008:19).

Genetik algoritmanın adımları şöyledir:

- Adım 1: Başlangıç
n adet kromozom içeren popülasyon oluşturulur
- Adım 2: Uyumluluk
Her x kromozomu için uyumluluğun $f(x)$ üzerinde değerlendirilmesi yapılır.
- Adım 3: Yeni popülasyon
Toplumdan uygunluklarına göre iki ata seçilir (daha iyi uyum, seçilme şansını artırır.). Yeni bir fert oluşturmak için ebeveynlerin bir çaprazlama olasılığına göre çaprazlanır. Eğer çaprazlama yapılmazsa yeni fert anne ve babanın aynısı

olacaktır. Yeni ferdin mutasyon olasılığına göre kromozom içindeki konumu değiştirilir. Yeni bireyin yeni popülasyona eklenir

- Adım 4: Değiştirme

Oluşan yeni popülasyon kullanılır

- Adım 5: Test

Eğer sonuç tatminkâr ise algoritma durdurulur. Aksi takdirde adım 6'ya gidilir

- Adım 6: Döngü

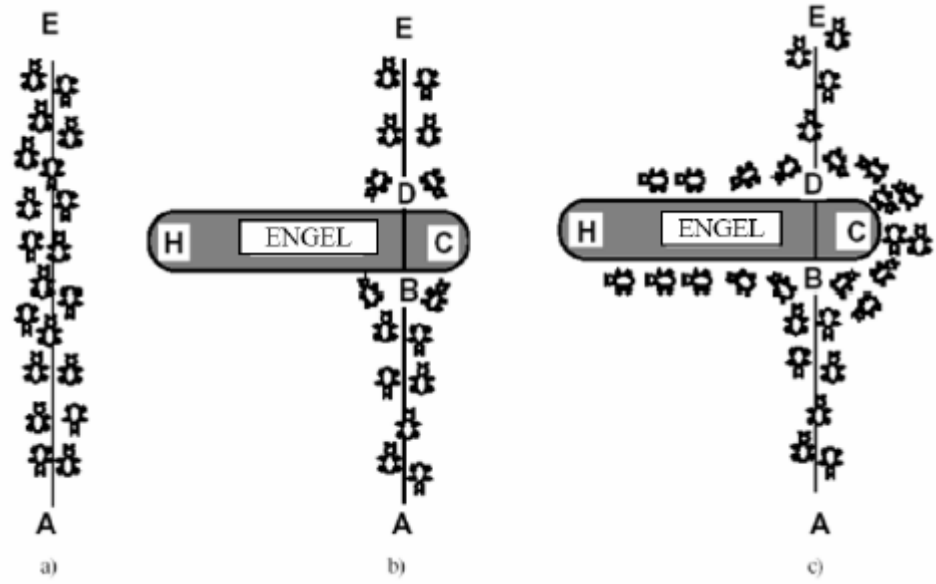
Adım 2'ye geri dönülür.

2.5.3.4 Karınca Algoritması

Temel ilkeleri ilk kez Marco Dorigo tarafından ortaya atılmış olan karınca kolonisi algoritmaları, karınca kolonilerinin feromon salgılayarak yiyecek kaynakları ile yuvaları arasındaki en kısa yolu bulma yöntemlerinden esinlenerek oluşturulmuş bir tekniktir.

Dorigo, karınca kolonilerinin davranışlarının matematiksel modelleri üzerine dayandığı karınca kolonisi algoritmalarını ilk kez gezgin satıcı problemi üzerinde kullanmış ve olumlu sonuçlar elde etmiştir. Bunun üzerine karınca kolonisi algoritmaları diğer araştırmacılar tarafından da kullanılmaya başlanmış ve günümüzde en iyileme problemlerinin çözümünde yaygın olarak kullanılan bir yapay zekâ tekniği haline gelmiştir (Dorigo 1999).

Karınca koloni optimizasyonlarının asıl kaynağı gerçek karıncaların yiyecek arama hareketidir. Karıncalar yiyecek ararken öncelikle kendi yuvalarına yakın çevreleri rastgele araştırırlar. Karıncalardan biri yiyecek kaynağı bulduğunda bu kaynağı kalite ve miktar açısından değerlendirir ve bir miktarını yuvasına taşır. Karınca yuvasına geri dönüş yolunda yol güzergâhına kimyasal feromon izleri adı verilen bir madde bırakır. Bırakılan feromon miktarı karıncanın bulunduğu yiyecek miktarı ve kalitesiyle ilişkilidir. Yola bırakılan bu feromon izleri diğer karıncaların bu yiyecek kaynağına ulaşabilmesi için yol gösterir. Bu feromon izleriyle karıncalar arasındaki dolaylı ilişki karıncaların yuvalarıyla yiyecek kaynağı arasındaki en kısa yolu bulmalarına yardımcı olur. Gerçek karıncalardaki bu karakteristik özellik tümleşik optimizasyon problemlerini çözmek için yapay karınca kolonilerinde kullanılmıştır (Serin 2009).



Şekil 7: Karıncaların Davranışları

- Karıncalar A-E arasındaki yolu izlemektedirler.
- Engel koyulan yolda karıncalar hangi yönü seçeceklerine rastgele karar verirler.
- Kısa olan yolda daha fazla feromon birikir.

Karıncı algoritması için kullanılan adımlar şöyledir;

- Adım 1: Tüm şehirlere belirli miktarlarda $b_i(t)$ karınca yerleştir. Buna göre karıncaların tabu listesi yenilenir. Her hattın koku miktarı sıfırlanır.
- Adım 2: Her şehirdeki tüm karıncalar için eğer $j \notin N_i$ ise $p_{ij}(t)=0$ fakat $j \in N_i$ ise $p_{ij}(t)=\{\tau_{ij}(t)\alpha[\eta_{ij}]\beta\}/\{\sum[\tau_{ij}(t)\alpha[\eta_{ij}]\beta]$ ile hesaplanan $p_{ij}(t)$ değerine bağlı olarak hareket etmek amacıyla j.şehir seçilir. Karınca k, j.şehre hareket ettirilir ve j. şehri k. karıncanın tabu listesine dâhil edilir. $\Delta\tau_{ij}(t,t+1)=\Delta\tau_{ij}(t,t+1)+Q/d_{ij}$ ile feromon miktarını yinelenir. Her kenar (i,j) için, $\tau_{ij}(t+1)=p.\tau_{ij}(t)+\Delta\tau_{ij}(t,t+1)$ hesaplanır. Bu adımdaki işlemler tabu listesi dolana kadar tekrarlanır
- Adım 3: Bulunan en kısa tur hafızaya alınır. Durdurma kriterini sağlanıyor ise Adım 4'e gidilir. Aksi takdirde, tüm tabu listesi boşaltılır. Tüm şehirlere belirli miktarda karınca yerleştirilir ve Adım 2'ye geri dönlür.
- Adım 4: En kısa tur yazılır ve algoritma durdurulur.

2.5.3.5 Tabu Arama

Tabu Arama Algoritması, optimizasyon problemlerinin çözümü için F.Glover tarafından geliştirilmiş iteratif bir araştırma algoritmasıdır. Temel yaklaşım, son çözüme götüren adımın dairesel hareketler yapmasını önlemek için bir sonraki döngüde tekrarın yasaklanması veya cezalandırılmasıdır. Böylece yeni çözümlerin incelenmesiyle Tabu Arama algoritması, bölgesel en iyi çözümün daha ilerisinde bulunan çözümlerin araştırılabilmesi için bölgesel-sezgisel araştırmaya kılavuzluk etmektedir.

Tabu Arama algoritmasının bölgesel optimalliği aşmak amacıyla kullandığı temel prensip, değerlendirme fonksiyonu tarafından her iterasyonda en yüksek değerlendirme değerine sahip hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesine dayanmaktadır. Bunu sağlamak amacıyla bir tabu listesi oluşturulur, tabu listesinin orijinal amacı önceden yapılmış bir hareketin tekrarından çok tersine dönmesini önlemektir. Tabu listesi kronolojik bir yapıya sahiptir ve esnek bir hafıza yapısı kullanır. Tabu arama algoritması her ne kadar istenmeyen noktaların işaretlenmesi olarak açıklanmış olsa da daha cazip noktaların işaretlenmesi olarak ta kullanılır.

TA'yı açıklamak için aşağıdaki gibi bir gösterimden yararlanılabilir;

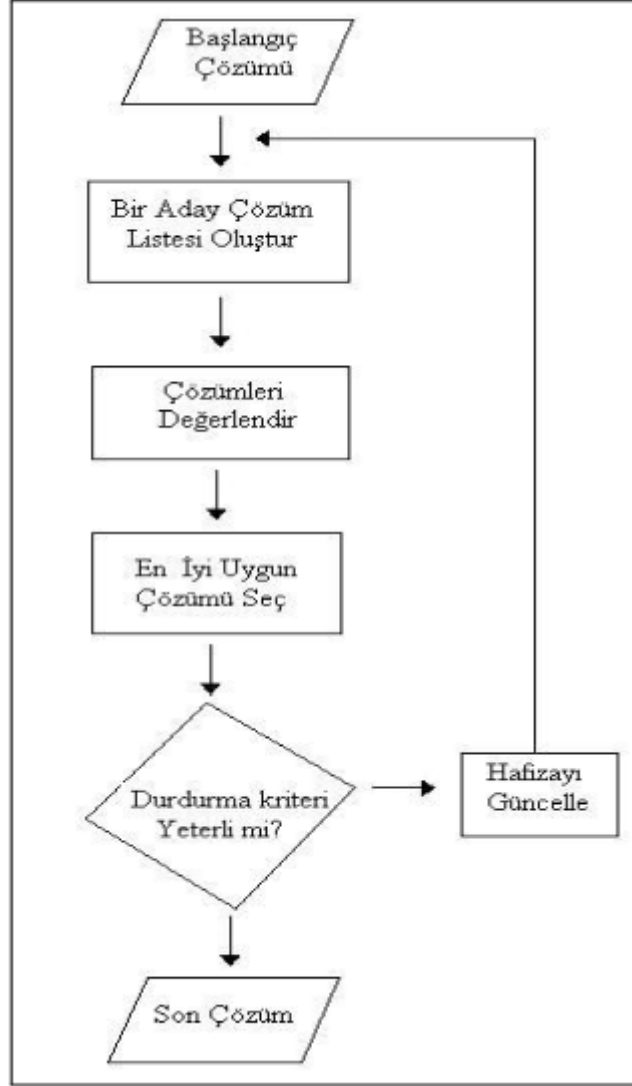
$$\text{Min } c(x) \quad x \in X$$

Yukarıdaki ifadeyi açıklarsak; amaç fonksiyonu $c(x)$ maliyet veya kar fonksiyonunun en küçük veya en büyük değerini aramaktadır fakat bu aramada x vektörü ile belirtilen kısıtlamalara uyularak çözüme ulaşılabacaktır.

Başka bir ifade ile her x elemanı bir hareketi temsil eder ve tüm hareketler X ile gösterilmektedir. Ancak daha doğru bir varsayım x vektörlerinin TA bellek yapısı olarak kullanıldığıdır. Böylece vektörde tutulan bellek değerine bağlı olarak çözüm aramada bazı hareketler tabu olarak kabul edilip engellenecek, bazılarına ise daha fazla odaklanacaktır.

X vektöründeki her bir hareket ise mevcut çözümün bir komşusunun seçimini temsil eder

Tabu Arama Algoritması Akış Diyagramı



Şekil 8: Tabu arama algoritması

TA algoritması, bir başlangıç çözümü ile aramaya başlar. Algoritmanın her iterasyonunda tabu olmayan bir hareket ile mevcut çözümün komşuları içerisinde bir tanesi seçilerek değerlendirilir. Eğer amaç fonksiyonunun değerinde bir iyileştirme sağlanmışsa komşu çözüm, mevcut çözüm olarak dikkate alınır. Seçilen bir hareket tabu olmasına rağmen tabu yıkma kriterlerini sağlıyorsa, mevcut çözümü oluşturmak için uygulanabilir. Geriye dönüşleri önlemek için, bir takım hareketler tabu listesine kaydedilerek tekrar yapılması belirli bir süre için yasaklanır. Belirlenen bir durdurma koşuluna göre algoritmanın çalışması sonlanmaktadır.

2.5.3.6 Yapay Sinir Ağları

Yapay Sinir Ağları (YSA), insan beyninde öğrenmeyi sağlayan biyolojik sinir sisteminin çalışma prensibinin bilgisayar programları ile simüle edilmesidir. YSA işlem birimi olarak da adlandırılan sinir hücreleri (nöronlar) içerirler. Bu işlem birimleri ağırlıklandırılmış olarak çeşitli şekillerde birbirlerine bağlanarak öğrenmeyi sağlayan ağı oluştururlar. Bir transfer fonksiyonu olarak görev yapan nöronlar diğer nöronlarla çeşitli sinyaller aracılığıyla haberleşerek sinyalleri birleştirme ve dönüştürme işlevleri ile sayısal bir sonuç ortaya çıkarırlar (Anonim 2015).

YSA veri madenciliği alanında sınıflandırma, kümeleme, örüntü tanıma gibi konularda kullanılmasının yanı sıra optimizasyon problemlerinin çözümünde de kullanılmaktadır. Bu amaçla YSA ilk kez Hopfield ve Tank (1985) tarafından NP-Zor problemler arasında yer alan gezgin satıcı problemine uygulanmıştır. Hopfield ve Tank'ın bu yaklaşımı daha sonra Foo ve Takefuji (1988) ve Sabuncuoğlu ve Gürgün (1996) tarafından atölye tipi çizelgeleme problemine uyarlanmıştır. Bu uygulamalar küçük boyutlu problemler için iyi sonuç verse de, büyük boyutlu problemlerin çözümünde çok etkin değillerdir. Bu sebeple Agarwal ve diğerleri (2003) genişletilmiş yapay sinir ağları (AugNN) adı verilen ve optimizasyon problemlerinin çözümünde kullanılan farklı bir yaklaşım ortaya koymuşlar ve bu yaklaşımı iş çizelgeleme problemi üzerinde test etmişlerdir (Touzet 1992).

AugNN sezgisel yöntemler ve YSA'nın birlikte kullanılması ile oluşmaktadır. Bu şekilde sezgisel yöntemlerin ve yinelemeli öğrenmenin avantajları bir araya getirilerek iyi sonuçlar elde edilebilmektedir. AugNN'de çözülmeye çalışılan problem; girdi katmanı, gizli katmanlar ve çıktı katmanı şeklinde tanımlanarak YSA'na dönüştürülür. Klasik YSA'ndan farklı olarak AugNN'de gizli katman sayısı probleme bağlıdır. Katmanlar arasındaki bağlantılara ağırlıklar verilir ve problemin kısıtları da göz önünde bulundurularak sezgisel yöntemler yardımı ile bir iterasyon sonucu elde edilir. Elde edilen bu sonuç hafızaya alınır. Daha sonra bağlantılar arası ağırlıklar bir öğrenme stratejisi kullanılarak değiştirilir ve bir sonraki iterasyonun sonucu bulunur. Bu şekilde iterasyonlar yapılarak daha iyi sonuçlar elde edilmeye çalışılır. AugNN'in avantajları arasında hızlı bir şekilde iyi sonuç bulması, sonuçlardaki iyileşmenin büyük bir kısmının arama sürecinin başında gerçekleşmesi ve yerel çözüm aramada etkili olması sayılabilir. Ayrıca AugNN'de diğer

meta sezgisel yöntemlere kıyasla yakınsama daha hızlı olmakta ve büyük boyutlu problemler için bile optimum veya optimuma yakın sonuçlar az sayıda iterasyon ile bulunabilmektedir.

Bu kısımda bir ARP'nin işlem birimleri yardımıyla YSA'na nasıl dönüştürüldüğünü göstermek amacıyla 3 araç ve n müşteriden oluşan bir örneği ele alalım. Şekil 9'da oluşturulmuş olan YSA, giriş katmanı, gizli katmanlar ve çıkış katmanı ile birlikte görülmektedir. Şekilden de görüleceği gibi n adet araç katmanı ve her bir araç katmanında üç adet araç düğüm noktası mevcuttur.

Her bir araç katmanını takip eden toplam n adet müşteri katmanı vardır. Her bir müşteri katmanı 3n adet müşteri düğüm noktasından oluşmaktadır. Birinci araç katmanı YSA'nın giriş katmanını oluşturmaktadır. Diğer araç katmanları ve müşteri katmanları gizli katmanlar olarak düşünülmekte ve en sonda bulunan düğüm noktası ağınc çıkış katmanı olarak görev yapmaktadır. Araç katmanları ve müşteri katmanları arasındaki bağlantılarda aracın bulunduğu nokta ile ilgili müşteri arasındaki mesafeler ağırlıklı olarak yer almaktadır.

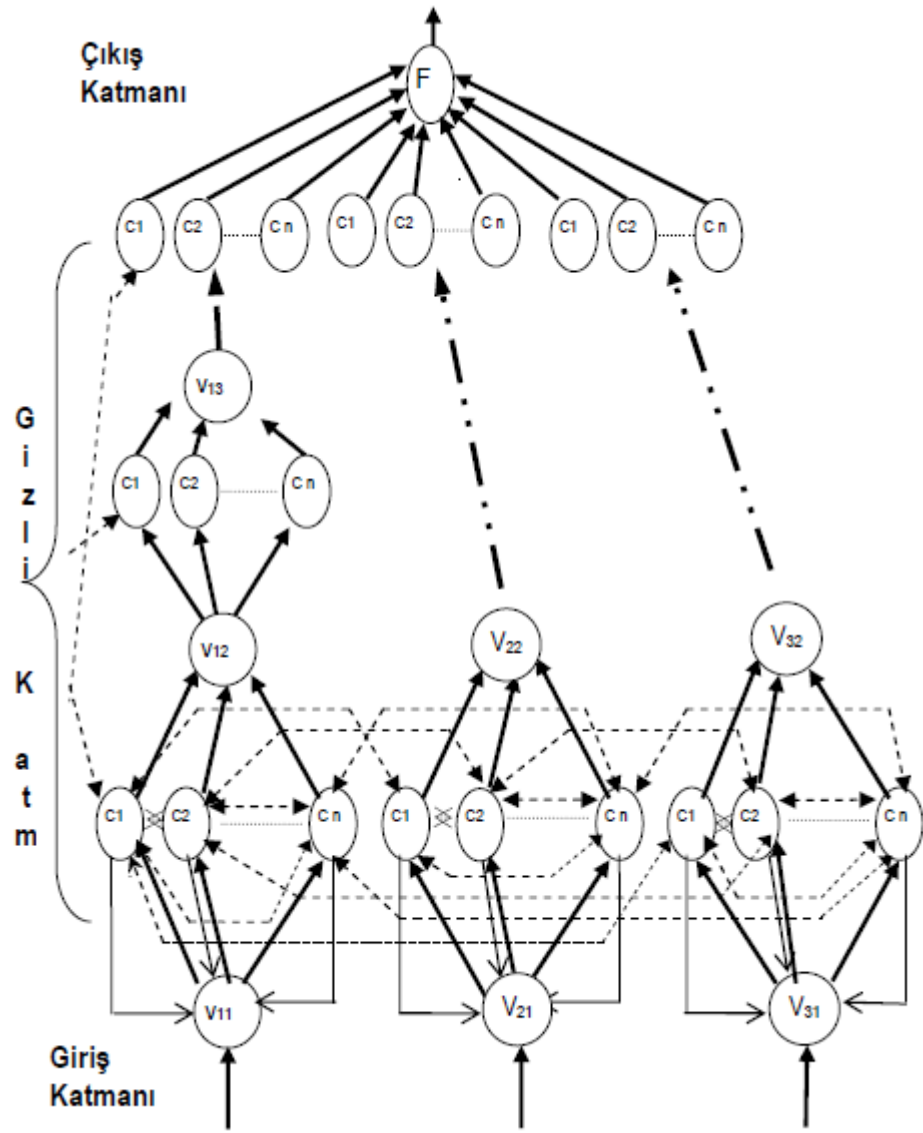
Başlangıç iterasyonunda bütün mesafeler için eşit ağırlıklar kullanılmakta, sonraki iterasyonlarda ise bu ağırlıklar değiştirilerek yeni sonuçlar elde edilmektedir. Benzer şekilde müşteri katmanları ve araç katmanları arasında çeşitli bağlantılar mevcuttur. Bu bağlantılar ağırlıksız olup bir katmandan diğer katmana bazı fonksiyonları tetiklemek için kullanılmaktadırlar. Müşteri katmanları arasında üç farklı bağlantı çeşidi mevcuttur. Birinci çeşitte, her bir müşteri katmanında aynı araçtan sinyal alan müşteriler arasında bağlantılar vardır. Bu bağlantıların amacı herhangi bir aracın aynı anda sadece bir müşteriye uğramasını sağlamaktır.

İkinci çeşit bağlantıda ise her bir müşteri düğümü, aynı katmanda bulunan ve diğer araçlardan sinyal alan aynı müşteri düğümlerine bağlıdır. Burada amaç aynı müşterinin aynı anda iki araç tarafından seçilmemesidir. Son olarak üçüncü çeşit bağlantıda ise her bir araç katmanında yer alan her bir müşteri düğümü, aynı araç ve diğer araçların aynı müşteri düğümlerine bağlıdır. Bu bağlantıların amacı ise her bir müşterinin sadece tek bir araç tarafından ziyaret edilmesini sağlamaktır. Ayrıca, müşteri düğümleri ters yönde ilgili araç düğümlerine bağlıdır. Bu sayede herhangi bir araç, herhangi bir müşteriye ziyaret

etmeye karar verir ise, o müşteri düğümünden araç düğümüne sinyal gönderilmekte ve aracın o aşamada o müşteriye ziyaret ettiği bildirilmektedir.

AugNN algoritmasının adımları Şekil 9’de görülmektedir. Bu algorithmada öncelikle problemin başlangıç ağırlıkları tespit edilmektedir. Sonraki aşamada ise ağırlıkların güncellenerek yeni çözümlerin elde edileceği iterasyonlar başlamaktadır. Elde edilen çözüm sayısı daha önceden belirlenmiş gerekli çözüm sayısına eşit oluncaya kadar iterasyonlar devam etmektedir. Her bir iterasyonda sezgisel yöntemlerle birlikte YSA çalıştırılır ve yeni bir sonuç elde edilir. Bu çalışmada kullanılan sezgisel yöntemler “en yakın komşuluk” ve “tasarruf algoritması” şeklindedir. Elde edilen sonuç daha önce bulunan sonuçlarla karşılaştırılır ve bulunan sonuçlar kümesine eklenir. Eğer bulunan bu sonuç daha önce bulunan sonuçların tamamından daha iyiye, en iyi sonuç olarak kaydedilir. Yeni sonuç bulunduktan sonra bir öğrenme stratejisi yardımıyla ağırlıklar güncellenir. Bu esnada ayrıca kuvvetlendirme ve en iyiye dönüş işlemleri de yapılmaktadır.

En yakın komşuluk yönteminde depodan çıkan araçlar, öncelikle depoya en yakın müşteriye gitmektedir. Daha sonra bu müşteriden araç kapasitesini aşmayacak şekilde en yakın olan diğer müşteriye veya depoya gidilerek rota tamamlamaktadır. Bu işlem tüm müşteriler ziyaret edilinceye kadar devam ettirilir.



Şekil 9: ARP için Yapay Sinir Ağı

AugNNE algoritmasının aşamalarının pseudo hali aşağıdaki gibidir;

Başlangıç ağırlıklarını belirle
Çözüm sayısı = gerekli çözüm sayısı olana kadar döngüye devam et
Yapay sinir ağını çalıştır
Sonucun tekrarlanıp tekrarlanmadığını kontrol et
Eğer sonuç tekrarlamalı bir sonuç değilse
Eğer sonuç en iyi sonuçtan daha iyiyse, en iyi sonuç olarak kaydet
Çözüm sayısını bir arttır
Bu sonucu bulunan sonuçlar kümesine ekle
Ağırlıkları güncelle
Bulunan sonuç, bir öncekine göre daha iyi ise, ağırlıkları bu yönde değiştir
Eğer daha önceden belirlenen bir iterasyon sayısı kadar denemede en iyi
sonuç geliştirememişse, ağırlıkları bir önceki en iyi sonucun ağırlıklarına
eşit al
Döngüyü sonlandır
En iyi sonucu göster

Tasarruf algoritmasında ise ilk aşamada her bir müşteriye bir araç atanır. Daha sonra $s_{ij} = c_{0i} + c_{0j} - c_{ij}$ tasarruf fonksiyonu yardımıyla iki müşterinin birleştirilmesi ile elde edilebilecek tasarruflar hesaplanır ve kapasite kısıdının aşılmadığı en büyük tasarruf bulunur. Bu müşteriler birleştirilerek yeni bir rota oluşturulur. Bu işlem birleştirilecek müşteri kalmayınca kadar sürdürülür.

YSA yukarıda bahsedilen en yakın komşuluk ve tasarruf algoritması sezgiselleri ile birlikte ağırlıklandırılmış uzaklıklar $w_{ij} = w_{ij} \times d_{ij}$ kullanılarak uygulanır. Bu şekilde her bir iterasyonda ağırlıklar değiştirilerek yeni bir çözüm elde edilmeye çalışılır. Elde edilen sonuç daha önce elde edilen sonuçlarla karşılaştırılır. Eğer sonuç yeni bir sonuç ise çözüm sayısı bir arttırılır ve bu çözüm mevcut sonuçlar kümesine eklenir. Aynı zamanda bu sonucun mevcut en iyi sonuçtan daha iyi olup olmadığına bakılır. Eğer yeni sonuç daha iyi ise en iyi sonuç olarak kaydedilir. Daha sonra ağırlıklar güncellenir.

Ağırlıklar güncellenirken bir öğrenme stratejisi uygulanır. Bu çalışmada öğrenme stratejisi olarak:

$$w_{ij} = \begin{cases} w_{ij} + LR \times r_2, & r_1 > 0,5 \text{ ise} \\ w_{ij} - LR \times r_2, & r_1 \leq 0,5 \text{ ise} \end{cases}$$

kullanılmıştır. Burada LR öğrenme katsayısını, r_1 ve r_2 ise (0,1) aralığında düzgün dağılımdan gelen rassal sayıları göstermektedir.

Ağırlıklar güncellendikten sonra, elde edilen çözümün bir önceki çözüme göre daha iyi olup olmadığı incelenir. Eğer mevcut çözüm bir öncekinden daha iyi ise, ağırlıklar bu yönde değiştirilir. Bu işleme kuvvetlendirme adı verilir.

Algoritmanın başlangıcında bir kuvvetlendirme faktörü seçilir ve ağırlıklar bu kuvvetlendirme faktörü oranında aşağıdaki gibi değiştirilir:

$$w_{ij} = w_{ij} + RF \times (w_{ij} - w_{ij}^{previous})$$

Burada w_{ij} i ve j müşterileri arasındaki ağırlığı, $w_{ij}^{previous}$ bir önceki iterasyondaki ağırlığı RF ise kuvvetlendirme faktörünü göstermektedir.

Ayrıca daha önceden belirlenen bir iterasyon sayısı kadar deneme yapılmış ve en iyi sonuçta herhangi bir gelişme görülmemişse, ağırlıklar bir önceki en iyi sonucun ağırlıklarına eşit olacak şekilde güncellenir. Bu işleme en iyiye dönüş adı verilir.

Bu işlemler tamamlandığında bir iterasyon yapılmış olur. İterasyonlar tamamlandığında en iyi sonuç belirlenir ve çözüm olarak sunulur.

3. MATERYAL ve YÖNTEM

3.1 Materyal

Gerçekleştirilen yüksek lisans tezi, bir ana sanayi kuruluşunun ihtiyacından yola çıkılarak üretilmiştir. İhtiyaca konu olan aktivite, tedarik zinciri kapsamı içinde önemli paya sahip olan hammadde tedarikinin nakliyesidir.

Çalışmanın temel materyali, Oyak-Renault Otomobil Fabrikaları A.Ş. nakliye departmanı tarafından organize edilen ve yurtiçi tedarikçilerinden fabrikaya kadarki parça tedarik sürecini kapsayan nakliye planlama aktivitesidir.

Oyak-Renault Otomobil Fabrikaları A.Ş., ülke sanayinde önemli rolü ve ağırlığı olan, kurulduğu 1969 senesinden bu yana yerli otomotiv sanayinde öncülük yapmış bir firmadır. Sürekli artan üretim hacmi, devreye aldığı yeni araç projeleriyle sadece faaliyet gösterdiği bölgede değil, tüm Dünya'daki Renault fabrikaları arasında da etkin ve güçlü bir konuma sahiptir.

Bu bağlamda söz konusu üretici firma Dünya otomotiv piyasasında önemli bir yere sahiptir. Türk otomotiv sanayinin öncülerinden olarak görülebilecek firma, 2014 yılı verilerine göre ortalama 360.000 adet araba ve 450.000 adet motor üretim kapasitesi ile Renault'nun Batı Avrupa dışında en yüksek kapasiteye sahip fabrikası konumundadır. 2014 yılında, Türkiye genelinde ihracatta 3. Konuma sahiptir.

Şirketin Türkiye fabrikasında 4 ana modelin üretimi yapılmaktadır. Firma, kurumsal yönetim yapısı olarak sınıflandırılabilir bir organizasyon yapısına sahiptir. Bu yapıda, yönetimin başında tek genel müdür ve genel müdüre bağlı 12 direktörlük bulunmaktadır. Bu direktörlüklerin her biri kuruluşun farklı kısımlarını yönetir ve onlarla ilişkili gerekli faaliyetleri ele alır.

Firmanın 535,000 m² kapsayan üretim tesisi ve depoları bulunmaktadır. Bu alan içerisinde malzeme boşaltma operasyonları için 13 indirme noktası (rampa) bulunmaktadır. Her rampa fabrikanın farklı bölümlerine ait belirli depolara hizmet vermektedir.

Üretim için kullanılan makineler, çalışanlar (6.000 civarında), teknolojiler, yazılım altyapısı firmanın kullandığı ana kaynaklardır. Firmanın taşıma işlemleri için ayırdığı cihazlar (kasalar, forkliftler ve elektrikli çekiciler), taşıeron şirket ve onun kamyonları sistemin diğer kaynakları olarak belirtilebilir.

Firma, karlılığına katkı sağlayacak optimizasyon aktiviteleri rekabet gücünü arttırmaya ve mevcut konumu güçlendirmeye katkı olarak şirket politikası gereği desteklemektedir..

Tedarik zincirinde meydana getirilecek iyileştirmeler bu doğrultuda firma tarafından ilgiyle takip edilmekte ve desteklenmektedir. Yönetimsel ve operasyonel aktivitelerin bir bölümü Fransa ve Romanya'da bulunan yetkili birimler tarafından karşılanmakta, büyük bir bölümü ise Türkiye'den yürütülmektedir.

Bu bağlamda yurtdışı nakliye aktiviteleri yukarıda bahsedilen merkezi birimler tarafından yürütülmekte olup, yerli taşımacılık faaliyetleri Oyak-Renault Otomobil Fabrikaları A.Ş. nakliye departmanı tarafından organize edilmektedir.

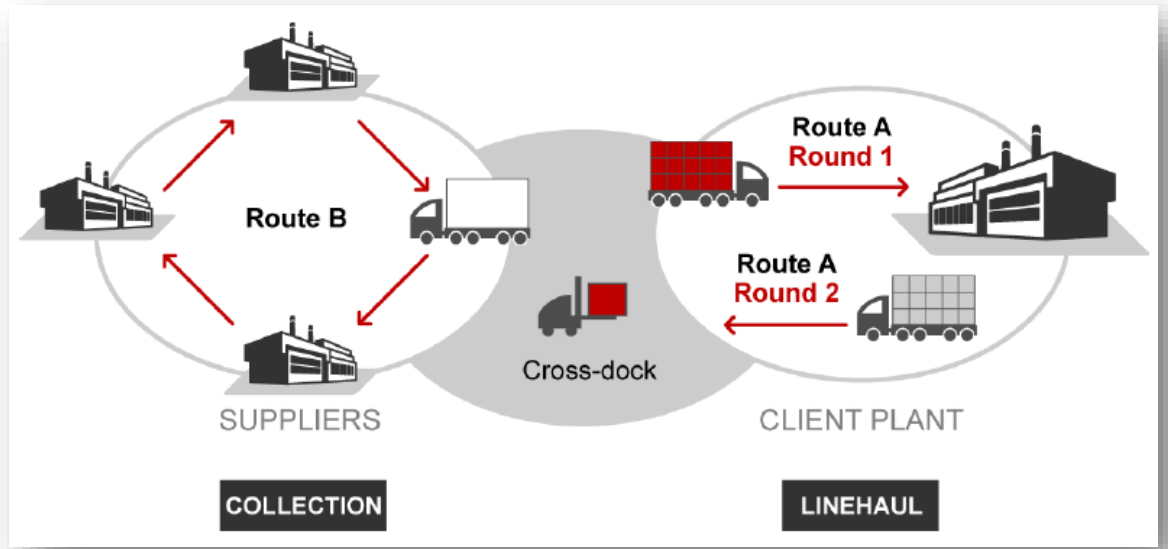
Çalışmanın iskeletini oluşturan yurtiçi parça tedarikinde nakliye optimizasyonu konusu tam da bu noktada lokal birimlerin amaçlarıyla kesişmektedir.

Yerli tedarikçilerden ikmal edilen parçaların nakliye bütçesi yıllık olarak 9 Trilyon TL civarındadır. Bu bütçenin aşağıya düşürülmesi söz konusu alanda yapılacak iyileştirmelerle mümkündür.

Yerli tedarikçilerin çoğunluğu Bursa'da bulunmakla birlikte, İstanbul, İzmit, Tekirdağ, Ankara ve İzmir'de yerleşmiş birçok imalatçı ve/veya tedarikçi ile de çalışılmaktadır.

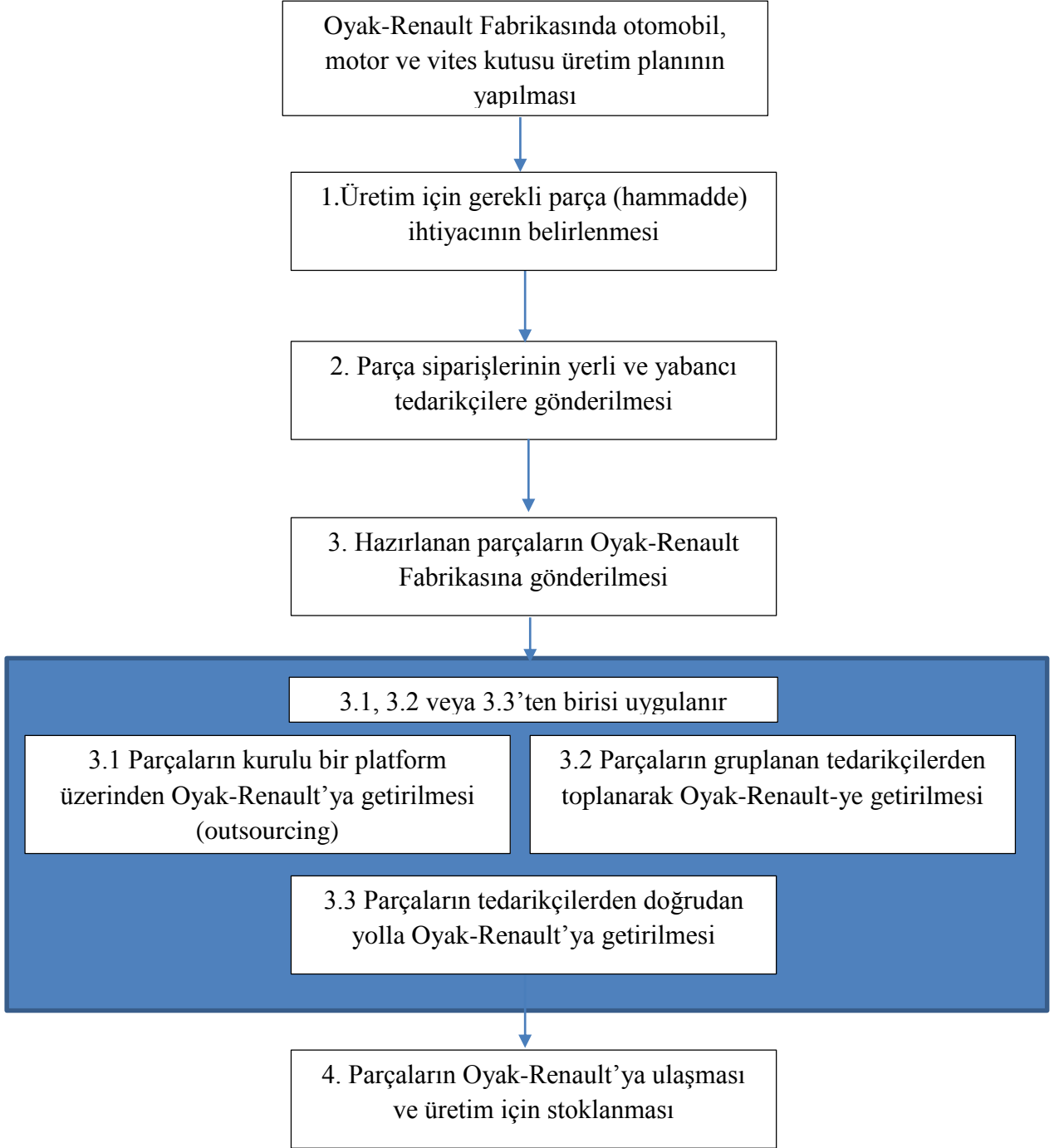
Tedarikçilerden toplanan hammadde, yarı mamul veya bitmiş mamullerin farikada üretime gelişinde toplama, direkt akış veya platform (cross-dock) biçimlerinden en uygun olanı, zamanın üretim ve maliyet koşullarına göre seçilmektedir.

Nakliye biçimlerini özetleyen bir şema Şekil 10 aşağıda paylaşılmaktadır.



Şekil 10: Nakliye biçimleri

Bununla birlikte baştan sona bütün hatlarıyla iş akışı 4 ana başlıkta aşağıdaki Şekil 11’de anlatılmıştır.



Şekil 11: İş akışına ait şema

3.2 Yöntem

Başından bu yana değinildiği gibi tez konusu NP Zor problemler kapsamındadır. Problem, bir merkezde bulunan araçların, talepleri bilinen müşteri kümesine hizmet edip tekrar merkeze dönmesini sağlayacak en kısa rotaların bulunması problemidir (Suna, Özkütük ve Gencer 2011). Her müşteriye sadece bir araç tahsis edilmeli ve bir araç ile ilişkilendirilen müşterilerin toplam talebi o aracın kapasitesini geçmemelidir (Balakrishnan 1993).

Diğer bir ifade ile teze konu olan problem, bir dizi yan kısıtlar eşliğinde dağıntık coğrafi yapıya sahip şehir veya tedarikçiler için bir veya birden çok depoda bulunan araç filosuna ait rotaların belirlenmesini içeren problem sınıfına girmektedir.

Oyak-Renault Otomobil Fabrikaları'na parça sağlayan yerli tedarikçilerinin yaklaşık 40 kadarı Bursa şehrinde ve çevresindeki ilçelerde faaliyet göstermektedir. Diğer şehirlerdeki tedarikçiler (İstanbul, İzmit, Tekirdağ, Ankara ve İzmir) için uygulanması gereken taşımacılık yöntemi fazla seçeneğe sahip değildir, çoğunlukla bu şehirlerdeki tedarikçiler talep, hacim ve kapasite nedeniyle aynı rotada bulunmaktadır.

Bir başka ifadeyle haftalık hacme göre kamyonlar hep aynı rotayı izleyerek Bursa dışı tedarikçilerden alınacak parçaları toplayarak fabrikaya ulaşmaktadır. Ayrıca, fabrika içerisinde yer alan boşaltma rampaları farklı olduğundan bu tedarikçilerden gelen kamyonların tam dolmamış olsalar bile Bursa'daki bir tedarikçiye uğramaları mümkün değildir.

Dolayısıyla, optimum sonucu elde etmek için bir matematiksel model kurulmuştur. Bu model MPL (Mathematical Programming Language) isimli bir yazılımda kodlanarak, Bursa'da faaliyet gösteren tedarikçiler için çalıştırılmıştır. Modelin hangi tedarikçi sayısına kadar optimum çözümü makul süreler içerisinde verebildiğini anlamak amacıyla tedarikçiler 5, 10, 15, 20, 25'lik gruplara ayrılmıştır.

Önerilen matematiksel modelde, ARP için kullanılan ve Miller, Tucker, Zemlin (MTZ) kısıtları olarak bilinen alt tur eleme ve kapasite kısıtları bu modele uyarlanmıştır.

Bu kısıtlar ilk olarak Gezgin Satıcı Problemi (GSP) için Miller vd. (Miller, Tucker ve Zemlin 1960) tarafından geliştirilmiş, Kulkarni ve Bhave tarafından ARP'ye uyarlanmış (Kulkarni ve Bhave 1985), Desrochers ve Laporte tarafından kuvvetlendirilmiş (Desrochers ve Laporte 1991) ve Kara vd. tarafından düzeltme yapılmıştır (Kara, Laporte ve Bektas 2004). Buna ek olarak Kara tarafından kullanılan yardımcı değişkenlerin sınırları üzerinde yeni kuvvetlendirmeler gerçekleştirilmiş ve bu değişkenlere kesin anlamlar yüklenmiştir.

Sanayi içerisinde tüm firmalar önceden belirledikleri takvimler ışığında planlamalarını yapmaktadırlar. Tez içerisinde ortaya konulan matematiksel model ise bu noktada, tedarikçiler ve ana sanayinin çalışma ve sevkiyat saatlerini dikkate alacak biçimde tasarlanmıştır. Bu nedenle matematiksel model, çok kullanımlı ve zaman pencereli araç rotalama problemi özelliğini taşımaktadır.

Matematiksel modelin uygun zamanda optimum çözüm verme sınırlarının belirlenmesinin yanı sıra çalışmanın yapıldığı ana sanayi firmasının beklentisini ölçülerek, kısa sürede kullanıma alınması mümkün olan bir uygulama hazırlanmaya çalışılmıştır.

Bu noktada firma yetkilileri 20 tedarikçi ve altı için bulunacak optimum bir çözümün ihtiyacın büyük bölümünü karşılayacağı yönünde görüş belirtmişlerdir.

4. BULGULAR ve TARTIŞMA

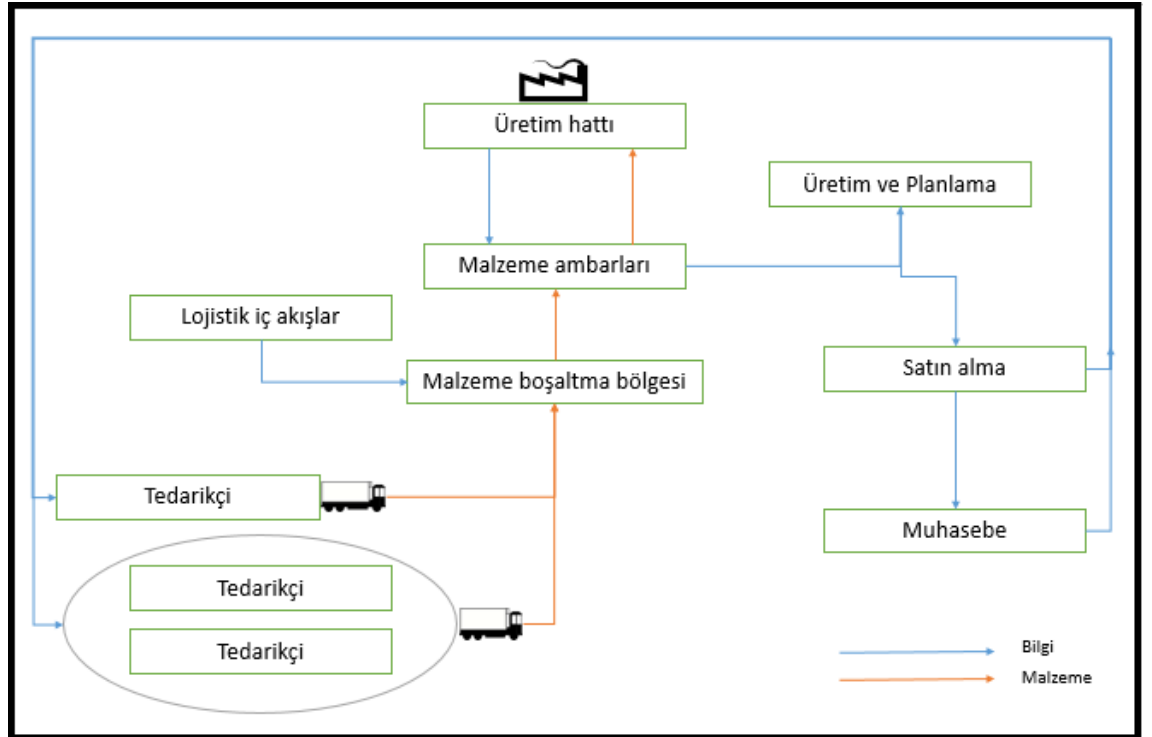
4.1 Durum analizi

Problem, yerli tedarikçilerden yapılan sevkiyatlar için belirli kısıtlar eşliğinde optimum nakliye rotasının belirlenmesidir. Üzerinde çalışılan model içeriğindeki algoritma, girdiler ve kısıtlar ışığında nakliye departmanı çalışanları için rota önerileri sunmaktadır.

Probleme ilişkin detaylara geçmeden önce, ihtiyacın nasıl ortaya çıktığını anlamak adına çalışmanın yapıldığı firmada bu faaliyetin nasıl gerçekleştirildiğine, güncel durumdaki verilere ilişkin genel anlamda bir bilgi vermek gerekmektedir.

Mevcut sistemde, üretici firmanın taşımacılık faaliyetleri taşıeron firmalar ile yapılmaktadır. Milk run sistemi ile yerli tedarikçilerden malzeme teminlerine ilişkin sistem dokümantasyonu üretici firma tarafından, operasyonel aktiviteler ve optimizasyon işleri de yine bu taşımacılık firmaları tarafından yürütülmektedir.

Bu sistemin ana bileşenleri üretici firma, taşıyıcı firma ve tedarikçilerdir. Bunlar arasında ilişkiler genel olarak Şekil 12 'de gösterilmiştir.

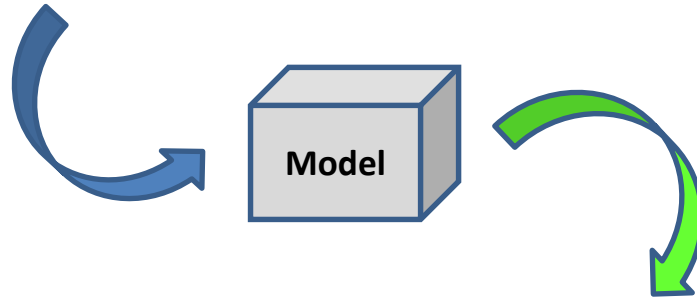


Şekil 12: Üretici firma, taşıyıcı firma ve tedarikçi ilişkileri

Yapılması hedeflenen çalışmaya genel bir bakış aşağıda ifade edilmiştir;

Girdiler:

- Tedarikçiler arası lokalizasyonlara bağlı olarak uzaklık hesapları
- Kamyon/konteynır yükleme doluluk hesapları
- Hammadde-tedarikçi hacim hesapları
- Bölgesel birim taşımacılık maliyetleri
- Tedarikçi ve müşteri çalışma/sevkiyat saatleri



Çıktılar:

- Her tedarikçi için optimum nakliye akış tipinin belirlenmesi (Toplama, direkt, ara platform vasıtasıyla)
- Taşımacılık maliyetlerinin minimizasyonu
- Üretici fabrika içi insan gücü optimizasyonu

Üretime konu olan mamullerin girdilerini sağlayan tedarikçi firmaların lokasyonları, sevkiyat hacimleri ve ulaşım masrafları dikkate alınarak en uygun taşıma yönteminin belirlenmesi, üretici firmanın sürdürülebilir gelişimine katkı sağlar. Nitekim bu çalışmada, üreticiye hammadde sağlayan tedarikçilerin taleplere bağlı olarak taşıma tipi ve planlamasının optimizasyonu hedeflenmektedir. Bu kapsamdaki mühendislik ve yükleme organizasyonu üretici sanayi kuruluşunun kontrolünde olacaktır.

4.1.1 Malzeme Siparişi ve Teslim planı

Malzeme ihtiyaç planlama yurt içi tedarikçilerle 1,2 veya 3 günlük dondurulmuş periyot ile çalıştırılmaktadır. Ayrıca, her hafta için toplam malzeme gereksinimi planlanmaktadır. Haftalık ihtiyaç içerisinde dondurulmuş dönem harici günlerdeki sipariş miktarlarında değişiklik yapılabilirken o hafta için toplam malzeme ihtiyacının değişmeme koşulu vardır. MİP içerisinde teslim süresi Bursa'da bulunan tedarikçiler için 2 gün, diğerleri için ise 3 gün olarak tanımlanmaktadır. Tedarikçiler için minimum sipariş miktarı yoktur. Firmada ana üretim planı ve MİP yürütülmesi için fabrika içi hazırlanmış MRP sistemi kullanılmaktadır.

Taşıyıcı firma ile yapılan sözleşme şartlarına göre üretici firmanın malzemeleri ile başka bir firmaya ait malzemeler aynı araç içerisinde taşınmazlar.

Taşıma işlemlerinde kapasitelerine göre 2 farklı araç kullanılmaktadır. Bunlar, 7,5 mL ve 13,4 mL uzunluğundadır. Nakliye planlamaları hacim üzerinden değil, mL birimi üzerinden yapılmaktadır. Çünkü parçaların konulduğu ambalajların yükleme esnasında en uygun eşleşmelerini hesaplayacak bir araç bulunmamaktadır. Bu nedenle pratikte işe yaradığı belirlenen, ambalaj taban uzunluğu ile kamyonun taban uzunluğu karşılaştırılması yöntem olarak kabul edilmiştir. Daha pratik anlatımla, bir ambalajın en uzun taban kenarı 2 m. ise toplam tabanı 13 m. olan bir kamyonun 2/13'ünü doldurduğu varsayılmaktadır.

Her tedarikçiye gidilebilecek belirli zaman pencereleri vardır, bunlar genellikle çalışma saatleri ve diğer yükleme işlemlerine bağlı olarak belirlenmektedir. Zaman pencerelerinin var oluşu planlamayı ve rotalamayı daha karmaşık yapmaktadır.

Sevkiyattan önceki gün, lojistik firma üretici firma ve tedarikçiler ile gerekli teyitleşmeleri yaparak hem siparişleri doğrular hem de tedarikçilerin malzemeleri teslim edebilecekleri zamanı belirlerler.

Bu doğrultuda günlük sefer planları oluşturulur ve Milk Run sistemi içerisindeki birimler ile paylaşılır. Günlük sefer planları içerisinde araçların yüklenme saatleri ile üretici firmaya varış saatlerini içermektedir. Bir örnek sefer planı Ek-A Tablo A.1 de verilmiştir.

4.1.2 Malzeme Yükleme ve Boşaltma Operasyonu

Tedarikçilerdeki yükleme operasyonları rampa planları, rampa- malzeme eşleşmesi ve malzeme-depo ilişkisi göz önüne alınmadan yapılmaktadır. Bu durum üretici firma içerisinde malzeme boşaltma işlemleri sırasında karmaşıklığa neden olmaktadır. Malzemelerin taşınma işlemlerinde kullanılan kasaların tamamı üretici firma tarafından temin edilmektedir. Kasalar belirli kasa tiplerine göre sabit hacimlere sahiptir. Bu kasa tiplerinin boyutları ve yapıldıkları malzemeler birbiriyle uyumlu olduğu için birbiri ile taşınamama kısıtı gibi bir kısıtlama bulunmamaktadır.

Malzemeler üretici firma içerisindeki 13 farklı indirme noktalarına boşaltılır. Rampaların en az biri boş ise araçlar yerleşke içerisine girebilir. Aksi takdirde, tesis içinde herhangi bir park alanı olmadığından tırlar giriş kapısı önünde beklemektedirler. Ancak, indirme noktalarının durumunu kontrol etmek için hiçbir sistem yoktur. Bir kamyon olağandışı bir durum olmadıkça birden çok boşaltma noktasına ait malzeme veya malzemeleri taşıyamaz.

Boşaltma işlemleri her gün 24 saat boyunca gerçekleştirilebilmektedir. Gereken indirme zamanı malzemenin hacmine, malzemenin tırdaki yerine ve mevcut işçi sayısına bağlı olarak değişmektedir ama ortalama olarak yarım saatlik zaman almaktadır.

4.1.3 Hedef, Amaç ve İstekler

Üretici firma etkili, entegre, verimli çalışan bir tedarik zinciri sistemi için milk run sistemi içerisindeki faaliyetlerinde iyileştirmeler hedeflemektedir. Buna ek olarak, üçüncü parti lojistik şirketi sorumluluğu altında olan milk run süreçlerinde daha fazla kontrol elde etmeyi amaçlamaktadır. Tüm bunlar ile birlikte daha az operasyonel maliyete ulaşmak üretici firmanın bu sistemdeki ana hedefidir.

Taşeron firma yüksek doluluk oranları ile çalışan bir filo ile maliyeti düşürmeyi ve karı maksimize etmeyi istemektedir. Ayrıca zaman penceresi kısıtı, üretici firmaya ait malzemelerin diğer firmaların malzemeleri ile taşınamama kısıtı gibi kısıtlamalardan kurtularak daha esnek bir sisteme sahip olmayı bu sayede de daha yüksek operasyonel maliyet ile çalışmayı hedeflemektedir.

Üretici firmanın yalın üretim politikası gereği minimum stok hedefi için malzemelerin tam zamanında fabrikaya getirilmesi isteği ile taşıyon firmanın yüksek doluluk oranları ile çalışma isteği bu sistem içerisindeki aktörlerin istekleri arasındaki zıtlıklardan biri olarak görülmektedir.

Üretici firmanın tedarikçilerinin performansını değerlendirmek için kullanılan bir tedarikçi değerlendirme sistemi bulunmaktadır. Bu nedenle, tedarikçiler yüksek hizmet seviyesine ulaşarak müşteri memnuniyetini arttırmayı ve mevcut sistemden elde ettikleri kar oranını arttırmayı hedeflemektedirler. Ayrıca, tedarikçi firmalar üretici firmadan gelen taleplerde düşük değişkenlik isteyerek kendi talep tahmin sistemlerinde daha doğru bir sonuca ulaşabilmeyi hedeflemektedirler.

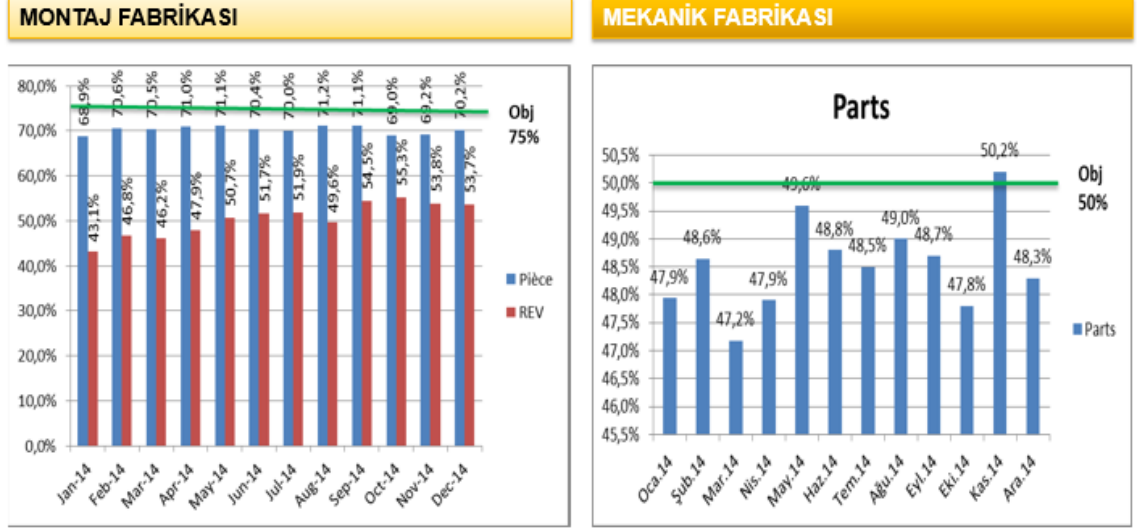
Milk Run sistemi faaliyetlerinin geliştirilmesinden sorumlu olan uygulayıcılar, düşük işletme maliyeti ve lojistik sistem operasyonları üzerinde daha fazla kontrol talep ederken Tedarik Zinciri Geliştirme yöneticisi, bir optimizasyon programı ile çalışan kolay bir arayüz ile iyi entegre olmuş ve maliyet etkili bir milk run faaliyet sistemi hedeflemektedir. Öte yandan, bu projede temel amaç, sistemik bir bakış açısı kullanarak mevcut durumun iyileştirilmesi için doğru bir eylem planı sunmaktır.

2014 yılında taşınan malzemeler için kullanılan rotalar incelendiğinde , milk run tedarik operasyonlarında bazı verimsiz rotalama aktivitelerinin gerçekleştirildiği görülmektedir. Özellikle, bazı uzak tedarikçi bölgelerine sıklıkla rotalar oluşturulduğu ve bu seferlerde, az miktarda malzeme temininin gerçekleştirildiği bulgusu göze çarpmaktadır.

Taşıyıcı firmanın, üretici firma tarafından günlük periyodlarla istenen tedarik malzemelerini taşımakla yükümlü olduğu gerçeği göz önüne alındığında, bahsedilen verimsiz rotaların üretici firmanın malzeme ihtiyaç planlamasının bir sonucu olduğu düşünülmektedir.

Mevcut durumda malzeme ihtiyaç planlaması gelecek 1 hafta için yapılmakta ve ilk 3 günlük periyot dondurulmaktadır. Bu duruma rağmen, özellikle uzak tedarikçilerden alınan malzemeler için dahi, hiçbir gruplama yapılmamakta ve az miktarlar olsa bile, taşıyıcı firma günlük bazda seferlere zorlanmaktadır. Bu durum da, bahsedilen uzak bölgeler için, oldukça düşük doluluk oranları ve verimsiz olarak gerçekleştirilen rotalar olarak sonuçlanmaktadır.

Bu durumu şekil 13 de gösterilen grafiklerle daha iyi izah etmek mümkündür.



Şekil 13: 2014 yılı şirket doluluk oranları

4.2 Problemin tanımı ve içeriği

Bahsedilen tüm analizler sonucu, çalışılacak olan problem şu şekilde ifade edilebilir:

“Üretici firmanın, malzeme ihtiyaç planlama sistemini, taşıma ve indirme aktivitelerini ve bunlar arasındaki etkileşimi göz önüne alarak yurtiçi milk run operasyonlarının verimliliğini arttırmayı hedefleyen etkili bir sistem tasarlamak.”

Başarı Ölçüleri

Mevcut sistemdeki aksaklıkları belirleyebilmek ve proje sonunda yapılan çalışmayı değerlendirebilmek amacıyla oluşturulmuştur. Farklı bakış açılarından değerlendirme yapabilmek amacıyla, çelişen başarı ölçüleri tercih edilmiş ve değerlendirmelerin değişik açılardan gerçekleştirilebilmesi amaçlanmıştır. Bu ölçüler;

Ortalama Tır Doluluk Oranları rotalama ve sipariş planlama verimliliği ile ilgili bir ölçüdür. Yurtiçi milk run tedarik sisteminin performansı açısından en önemli göstergelerden birini oluşturmaktadır.

İndirme Noktalarındaki Tırların Ortalama Bekleme Süresi üretici firma içerisindeki indirme operasyonlarının bir göstergesidir. Rampa Çizelgeleme, indirme prosesleri ve indirme noktalarındaki iş gücü dağılımı bu ölçüyü etkileyen başlıca bileşenlerdir.

Yurtiçi Milk Run Operasyonlarına Dahil Olan Malzemelerin Ortalama Envanter Seviyeleri, ortalama tır doluluk oranları ölçüsünün karşıtıdır. Proje kapsamında denenmesi düşünülen sipariş planlama toplasıklandırması sonucunda tır doluluk oranları artarken, söz konusu malzemelerin envanter seviyelerinin artacağı düşünülmektedir. Bu sebeple, tutulan envanter seviyesini gözlemlemek, önemli bir nokta olarak görülmektedir.

Yurtiçi Milk Run Taşınması Sürecinde Katedilen Toplam Rota Uzunluğu, taşıma maliyetini direkt olarak etkileyen bir ölçüdür. Hem rotalama hem de sipariş gruplama operasyonlarından etkileneyeceği düşünüldüğünde, önemli bir başarı ölçüsü olarak gözlenmesi gerekmektedir. Özellikle, geçmişte kullanılan ve proje sonunda önerilen sistemlerin karşılaştırılması sürecinde kullanılması düşünülmektedir.

Gerçekleşen Rotalar Sürecinde Geçen Zaman önerilen başarı ölçülerindedir. Kullanılan her bir araç ve iş gücü saatinin belirli bir fırsat maliyeti olduğu düşünüldüğünde, bu kaynakların kullanılma süresi önemli bir başarı ölçüsü olarak karşımıza çıkmaktadır.

Karar Ölçütleri

Proje kapsamındaki temel amaç “entegre ve etkili bir şekilde çalışan, düşük maliyetli bir yurtiçi milk run tedarik sistemine sahip olmak” olarak belirlenmiştir. Proje süresince, önerilen aksiyonların, belirtilen amaca ne kadar hizmet edip etmediğini görebilmek amacıyla, karar ölçütleri tanımlanmıştır. Önerilmesi planlanan aksiyonlar, belirtile ölçütlere göre değerlendirilecektir.

Milk Run Operasyonları Maliyetini Minimize Etmek, önerilen çözüm yaklaşımlarının sonucunda ulaşılması gereken en önemli ölçüt olarak görülmektedir.

Toplam Yurtiçi Milk Run Rota Uzunluğunu Minimize Etmek, tedarik sürecimaliyetini düşürmeyi sağlayacak bir ölçüttür. Katedilen her bir kilometrelik uzunluk, fazladan maliyet anlamına geldiği için, projenin amacına direkt etki etmektedir.

Milk Run Operasyonlarına Dahil Olan Malzemelerin Ortalama Envanter Seviyelerini Minimize Etmek, sistemdeki toplam maliyeti düşürmeyi amaçlayan diğer bir ölçüttür. Envanter taşıma maliyetleri de, projenin kapsamını oluşturan sistemin bir bileşeni olduğundan, bu maliyetleri azaltmak, proje amacına hizmet edecek bir ölçüt olarak düşünülmektedir.

Tırların Malzeme İndirme Sürelerini Minimize Etmek, kullanılan araç ve iş gücünü azaltmak anlamına geldiğinden, system maliyetini düşürme amacına direkt etki etmektedir.

4.3 Problemin Varsayımları

Problem çevresi ve elementleri belirlenirken bazı karmaşıklıklar ve kısıtlamalar ile karşılaşmıştır. Bu durumların aşılması amacıyla bazı varsayımlar yapılmış ve problem sınırları bu şekilde belirlenmiştir. Bunlar;

- i. Her müşteriye kesinlikle bir kez uğranmalıdır,
- ii. Tüm sefer rotaları, üretici firmada başlamakta ve yine üretici firmada sonlanmaktadır,
- iii. Rota üzerindeki müşterilerin talepleri toplamı araç kapasitesini geçmemelidir,
- iv. Her müşteriye zaman pencereleri içerisinde hizmet verilmelidir,
- v. Her bir rota, en fazla izin verilen süre içerisinde tamamlanmalıdır,
- vi. Sevkiyat sırasında kilometre başına geçen zaman sabittir ve dış faktörlerden etkilenmemektedir,
- vii. Malzeme ihtiyaç planlaması dışında meydana gelen ve olağanüstü durumlar sonucunda ortaya çıkan malzeme gereksinimleri proje kapsamında yer almamaktadır.
- viii. Tedarikçi, teyidini verdiği malzeme miktarını belirtilen zamanda hazır etmekte ve sevkiyatı yapan araca yüklemektedir,
- ix. Boş kasaların çevrimi, tırların geri dönüşleri sırasında yapılmakta ve taşıyıcı firma bu işlemleri yürütmektedir. Boş kasa çevrimi proje açısından bir sınırlama yaratmamaktadır,
- x. İndirme ve yükleme işlemleri 24 saat boyunca 3 vardiya şeklinde devam etmektedir,
- xi. Taşıyıcı firmada yeteri kadar araç olduğu varsayılmaktadır,
- xii. Ambar kapasiteleri yeterli ve kısıt oluşturmayacak seviyededir

4.4 Matematiksel Model

Notasyon ve parametreler

i, j : tedarikçiler (i ve $j=1$ ise fabrika)

J : tedarikçi noktaları kümesi

k : araçlar

K : araç kümesi

Q_k : araç kapasitesi (mL)

N_k : araç kapasitesi (kg)

D_j : j tedarikçisinden toplanacak ürün miktarı

E_j : j tedarikçisinden toplanacak ürünlerin ağırlıkları

a_i : i düğümünde (tedarikçisinde) servise en erken başlama zamanı

b_i : i düğümünde servise en geç başlama zamanı

$C_{i,j}$: i ve j noktaları arasındaki mesafe

$T_{i,j}$: i ve j noktaları arası ulaşım süresi

H_i : i düğümündeki hizmet süresi

$G_{i,j}$: i ve j noktaları arası ulaşım süresi ve servis süresi toplamı

M : Büyük bir sayı

Karar değişkenleri

l_{jk} : j noktasından sonra k aracında kalan ürün miktarı

Y_{jk} : alt tur oluşmasını engelleyen değişken

V_{jk} : j düğümüne k aracıyla varış zamanı

S_{ik} : i düğümünde k aracıyla servise başlama zamanı

W_i : i düğümünde bekleme zamanı

P_{jk} : j noktasından sonra k aracındaki yük ağırlığı

$X_{i,j,k}$: 1, eğer k aracı (i,j) ayrıtı için kullanılıyorsa

Amaç fonksiyonu

$$MIN \sum_{i \in J} \sum_{j \in J} \sum_{k \in K} C_{ij} X_{ijk} \quad (1)$$

Kısıtlar

$$\sum_{k \in K} X_{ijk} = 1 \quad i \neq j \quad j = 1, 2 \dots J \quad (2)$$

$$\sum_{j \in J} X_{ijk} = 0 \quad i = j \quad k = 1, 2 \dots K \quad (2.1)$$

$$\sum_{i \in J} X_{ijk} = \sum_{j \in J} X_{jik} \quad k = 1, 2 \dots K \quad (3)$$

$$\sum_{j \in J} X_{0jk} \leq 1 \quad k = 1, 2 \dots K \quad (4)$$

$$\sum_{j \in J} X_{i0k} \leq 1 \quad k = 1, 2 \dots K \quad (4.1)$$

$$l_{jk} \geq l_{ik} + D_j - M(1 - \sum_{k \in K} X_{ijk}) \quad k = 1, 2 \dots K \quad i = 2, 3 \dots J \quad j = 1, 2 \dots J \quad (5)$$

$$P_{jk} \geq P_{ik} + E_j - M(1 - \sum_{k \in K} X_{ijk}) \quad k = 1, 2 \dots K \quad i = 2, 3 \dots J \quad j = 1, 2 \dots J \quad (5.2)$$

$$l_{jk} \leq Q_k \quad k = 1, 2 \dots K \quad (6)$$

$$P_{jk} \leq N_k \quad k = 1, 2 \dots K \quad (6.1)$$

$$Y_{jk} \geq Y_{ik} + 1 - M(1 - \sum_{k \in K} X_{ijk}) \quad k = 1, 2 \dots K \quad i, j = 1, 2 \dots J \quad (7)$$

$$Y_{jk} \geq 0 \quad j = 1 \quad k = 1, 2 \dots K \quad (7.1)$$

$$S_{ik} - S_{jk} + M(\sum_{k \in K} X_{ijk}) \leq M - G_{ij} \quad k = 1, 2 \dots K \quad i, j = 1, 2 \dots J \quad (8)$$

$$S_{jk} \geq 7 \quad j = 1 \quad k = 1, 2 \dots K \quad (8.1)$$

$$a_j \sum_{k \in K} X_{ijk} \leq S_{jk} \leq b_j \sum_{i \in J} X_{ijk} \quad k = 1, 2 \dots K \quad i, j = 1, 2 \dots J \quad (9)$$

$$V_{jk} \leq S_{ik} + G_{ij} \sum_{i \in J} X_{ijk} + M(1 - \sum_{k \in K} X_{ijk}) \quad k = 1, 2 \dots K \quad i, j = 1, 2 \dots J \quad (10)$$

$$S_{ik} \geq G_{1i} - \left(1 - \sum_{i \in J} X_{1ik} \right) M \quad k = 1, 2 \dots K \quad j = 2 \dots J \quad (11)$$

$$V_{ik} \leq S_{ik} \quad i = 1, 2 \dots J \quad k = 1, 2 \dots K \quad (12)$$

$$X_{ijk} \in \{0, 1\} \quad (13)$$

$$j = 1, 2 \dots J \quad i = 1, 2 \dots J \quad k = 1, 2 \dots K \quad (14)$$

$$l_{jk} \geq 0 \quad V_{j,k} \geq 0 \quad l_{j,k} \quad W_i \geq 0 \quad S_{ik} \geq 0 \quad P_{jk} \geq 0 \quad Y_{jk} \geq 0 \quad j = 1, 2 \dots J \quad (15)$$

Modelde (1) nolu eşitlik; toplam mesafeyi en küçükleyen amaç fonksiyonudur. (2) nolu kısıt; bütün düğümlere bir defa gidilmesini sağlar. I ve j indislerinin aynı düğümleri tanımlıyor olması sebebiyle birinden yine aynı düğüme gidilmemesi gerektiğini anlatan kısıt, (2.1) nolu kısıttır. Varılan düğümü aynı araçla terk etmeyi sağlayan (3) nolu kısıttır.

İlk düğümden, yani fabrikadan çıkışı garanti alına alan (4) nolu kısıtın ardından tekrar fabrikaya dönüşü garanti altına alan (4.1) nolu kısıt gelmektedir. Rota boyunca düğümlerden sonraki araç yükleri (5) ve (5.2) nolu kısıtlarla sınırlandırılmıştır. İlk düğümden sonraki ve rota boyunca araç kapasitesi (6) ve (6.1) nolu kısıtla kontrol edilmektedir.

(7) nolu kısıt alt tur oluşmasını engeller. Hep fabrikadan başlamak gerektiğini, bu durumda ilk yerin Y değerinin hep 0 olması gerektiğini söyleyen kısıt (7.1) nolu kısıttır. (8) nolu kısıt rota üzerindeki tüm düğümlerde servise başlama zamanını belirler, (8.1) nolu kısıtsa, 1. noktadaki servise başlama saatinin sabit ve sabah 7 olmasını sağlayan kısıttır. (9) nolu kısıt servise başlama zamanının istenilen zaman penceresi arasında olmasını sağlar, (10) nolu kısıt rota üzerindeki herhangi bir düğüme varış zamanını belirler.

(11) nolu kısıt rota üzerindeki ilk düğümden servise başlama zamanını belirler, (12) nolu kısıt her düğümden servisin düğüme vardığından sonra başlamasını garanti eder, (13,14 ve 15) nolu kısıtlar ise işaret kısıtlarıdır.

Bu matematiksel modelin MPL (Mathematical Programming Language) programında yazılmış olan kodları ise aşağıdaki gibidir;

```
TITLE Zaman_Pencereli_ARP;
```

```
INDEX
```

```
    i := (1..6); ! Tedarikçi sayısı (1 numara fabrikayı temsil ediyor)
```

```
    k := (1..2); ! Araç sayısı
```

```
    j := i; ! i=j
```

```
DATA
```

```
! Araç kapasitesi (mL)
```

```
    Q[k] := SPARSEFILE("kapasite7.dat");
```

```
! Araç kapasitesi (kg)
```

```
    N[k] := SPARSEFILE("kapasitekg7.dat");
```

```
! i düğümünde en erken servise başlama zamanı
```

```
    A[i] := (0, 7, 8, 8, 7, 7);
```

```
! i düğümünde servise en geç başlama zamanı
```

```
    B[i] := (0, 14, 16, 18, 20, 20);
```

```
! Tedarikçilerden toplanacak miktar
```

```
    D[j] := SPARSEFILE("talepler7.dat");
```

```
! Tedarikçilerden toplanacak parçaların ağırlıkları
```

```
    E[j] := SPARSEFILE("agirlik7.dat");
```

```
! i ve j arası mesafe tablosu
```

```
    C[i,j,k]:= SPARSEFILE("mesafeler7.dat");
```

```
! i düğümünde hizmet süresi
```

```
    H[i]:= (0, 0.5, 0.5, 0.5, 0.5, 0.5);
```

```
! i,j arası ulaşım süresi
```

```
    T[i,j] := SPARSEFILE("sureler7.dat");
```

```
! i ve j arası ulaşım süresi ve servis süresi toplamı
```

```
    G[i,j] := T[i,j] + H[i];
```

INTEGER VARIABLES

$I[j,k]$; ! j noktasından sonra k aracındaki yük miktarı
 $Y[j,k]$; ! alt tur oluşmasını engelleyen değişken
 $V[j,k]$; ! j düğümüne k aracıyla varış zamanı
 $S[i,k]$; ! i düğümünde k aracıyla servise başlama zamanı
 $W[i]$; ! i düğümünde bekleme zamanı

VARIABLES

$P[j,k]$; ! j noktasından sonra k aracındaki yük ağırlığı

BINARY VARIABLES

$X[i,j,k]$; ! 1, eğer k aracı (i,j) ayrıtı için kullanılıyorsa

MODEL

MIN $Y_{ol} = \text{SUM}(X * C)$; ! ToplamTaşımaMesafesiMinimizasyonu

SUBJECT TO

! 2) Her tedarikçiye aynı rotada 1 kere gidilmesi
 $K40[j] \text{ WHERE } (j < > 1): \text{SUM}(i,k: X) = 1;$

! 2.1) 1 tedarikçiye 1 araçla gidiyorsa diğeriyle gitmesin;
 $K1[i,j,k]: X[i=j,k]=0;$

! 3) Varılan düğümü aynı araçla terk etmeyi sağlayan kısıt
 $K11[k]: \text{SUM}(i,j=2 : X) - \text{SUM}(i=2,j : X) = 0;$
 $K8[k]: \text{SUM}(i,j=3 : X) - \text{SUM}(i=3,j : X) = 0;$
 $K7[k]: \text{SUM}(i,j=4 : X) - \text{SUM}(i=4,j : X) = 0;$
 $K6[k]: \text{SUM}(i,j=5 : X) - \text{SUM}(i=5,j : X) = 0;$
 $K5[k]: \text{SUM}(i,j=6 : X) - \text{SUM}(i=6,j : X) = 0;$

! 4) Depodan her düğümüne mutlaka ama en fazla bir çıkış
 $K12[i=1,k]: \text{SUM}(j : X) = 1;$

! 4.2) Depoya mutlaka dönüş
 $K26[j=1,k]: \text{SUM}(i : X) = 1;$

! 5) j noktasından sonraki araçtaki parça miktarları
 $K13[i,j < > 1,k] \text{ WHERE } (i < > j): I[j,k] \geq I[i,k] + D[j] - (1000000000 * (1 - X[i,j,k]));$

! 5.1) 1. noktadan sonraki araç parça miktarı hep 0 olmalıdır
 $K41[j,k] \text{ WHERE } (j=1): I[j,k]=0;$

! 5.2) j noktasından sonraki araç parça ağırlıkları
!K52[i,j<>1,k] WHERE (i<>j): P[j,k] >= P[i,k] + E[j] - (1000000000 * (1- X[i,j,k]));

! 5.3) 1. noktadan sonraki araç parça ağırlıkları hep 0 olmalıdır
K53[j,k] WHERE (j=1): P[j,k]=0;

! 6) Araç yükü kısıtlaması
K14[j,k]: l[j<>1,k] <= Q[k];

! 6.1) Araç ağırlık kısıtlaması
K53[j,k]: P[j<>1,k] <= N[k];

! 7) Alt tur oluşmasını engelleyen kısıt
K15[i,j<>1,k] WHERE (i<>j) : Y[j,k] >= Y[i] + 1 - (1000000000 * (1- X[i,j,k]));

! 7.1) Hep fabrikadan başlamalı, bu durumda ilk yerin Y değeri hep 0 değeri olmalı
K42[j,k] WHERE (j=1): Y[j,k]=0;

! 8) ZPK-1 Rota üzerindeki tüm düğümlerde servise başlama zamanı
K16[i,j<>1,k] WHERE (i<>j): S[i,k] - S[j,k] + (1000000000 * X[i,j,k]) <= (1000000000 - G[i,j]);

! 8.1) 1. noktadaki servise başlama saati hep sabittir ve sabah 7'dir
K42[j,k] WHERE (j=1): S[j,k]=7;

! 9) ZPK-2 Servise başlama zamanının istenilen zaman penceresinde olması için
K17[j<>1,k]: A[j] * SUM (i : X) <= S[j,k];
K18[j<>1,k]: S[j,k] <= B[j] * SUM (i : X);

! 13) Her düğümde servisin düğüme vardıktan sonra başlaması için
K24[i,k]: V[i,k] <= S[i,k];

! 14) İşaret kısıtları
K25[j,k]: Y[j,k] >= 0;
K45[j,k]: S[j,k] >= 0;
K46[j,k]: l[j,k] >= 0;
K47[j,k]: V[j,k] >= 0;
K51[j,k]: P[j,k] >= 0;

END

4.5 Sayısal Örnek / Deneysel Çalışma

Küçük bir örnek kümede modelin çalışıp çalışmadığını anlamak adına 5 tedarikçilik bir gruplama yapılmıştır. Bunun ardından örneklem 10 ve 15 ve 20'lik gruplara dönüştürülerek modelin çalışıp çalışmadığı belirlenmiştir. Bu küme çalışmalarının ilerleyen safhalarında takibini sağlamak adına şu şekilde bir tanımlama yapılmıştır;

- Grup A: 5 tedarikçilik örneklem
- Grup B: 10 tedarikçilik örneklem
- Grup C: 15 tedarikçilik örneklem
- Grup D: 20 tedarikçilik örneklem

Matematiksel modele hesap olanağı yaratan girdiler aşağıda biçimde sıralanmıştır;

- a) Modele konu olan tedarikçi listesi
- b) Tedarikçiler arası uzaklıklar (kuşbakışı km)
- c) Tedarikçiler arası süreler (saat)
- d) Parçaların toplanacağı kamyonlar ve kapasiteleri
- e) Tedarikçilerden ilgili dönemde toplanacak parça miktarları (mL cinsinden) ve ağırlıkları (kg)
- f) Tedarikçilerin sevkiyat çalışma saatleri
- g) Bir kamyonun parça yüklenirken kaybettiği zaman

En küçük örneklem olan Grup A ile bu çalışmanın ne tür sonuçlar verdiğini takip eden bölümlerde görmek mümkündür. Devam eden bölümlerde ise diğer grupların detaylarına girmeden sonuçlarına ilişkin analizler ve en son bölümde de karşılaştırmalı analiz yer alacaktır.

4.5.1 Grup A ile Örnek Çözüm

Geçen sayfada bahsedilen girdilerin detaylı olarak ne anlama geldiğine ve değerlerine aşağıda yer verilmektedir.

a) Modele konu olan tedarikçi listesi

Grup A'yı oluşturan tedarikçi listesi aşağıdaki Tablo 1'de gösterildiği gibidir;

NO	TEDARİKÇİ LİSTESİ	TEDARİKÇİ KODU	ENLEM	BOYLAM
1	DELPHI	115238	40,28464	28,94268
2	ELRİNGKLİNGER	275057	40,18257	29,06687
3	ARVIN	115790	40,44032	29,34604
4	TKG-Çalı	111307	40,18257	29,06687
5	ORMETAL	268754	40,2255	28,37273

Tablo 1: Grup A tedarikçi listesi

b) Tedarikçiler arası uzaklıklar (kuşbakışı km)

Tablo 2'de detaylar bulunmaktadır. Öte yandan bu uzaklıklara ilişkin Google harita görüntüsü de Şekil 14'de gösterildiği gibidir.

5 TEDARİKÇİ MESAFE TABLOSU (km)						
	OYAK RENAULT	DELPHI (115238)	ELRİNGKLİNGER (275057)	ARVIN (115790)	TKG-Çalı (111307)	ORMETAL (268754)
OYAK RENAULT	0	31	27	21	20	68
DELPHI (115238)	31	0	4	15	26	49
ELRİNGKLİNGER (275057)	27	4	0	11	7	53
ARVIN (115790)	21	15	11	0	20	52
TKG-Çalı (111307)	20	26	7	20	0	66
ORMETAL (268754)	68	49	53	52	66	0

Tablo 2: Tedarikçiler arası uzaklık matrisi



Şekil 14: Tedarikçi uzaklıklarının harita üzerinde gösterimi

c) Tedarikçiler arası süreler (saat)

Bu 5 tedarikçi arası uzaklıklar yaklaşık olarak Tablo 3’de yer almaktadır.

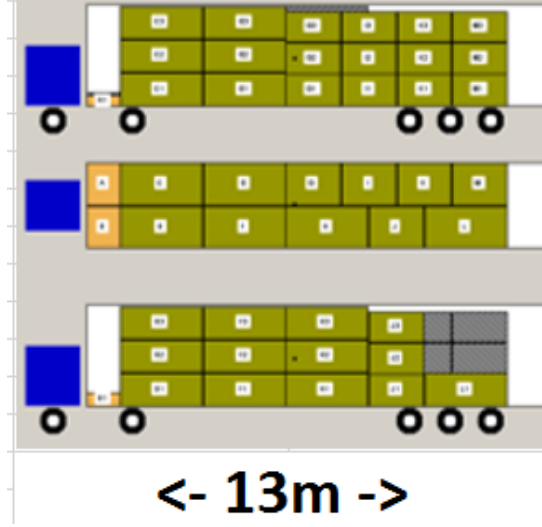
5 TEDARİKÇİ UZAKLIK TABLOSU (saat)						
	OYAK RENAULT	DELPHI (115238)	ELRINGKLINGER (275057)	ARVIN (115790)	TKG-Çalı (111307)	ORMETAL (268754)
OYAK RENAULT	0	0,44	0,39	0,30	0,29	0,97
DELPHI (115238)	0,44	0,00	0,06	0,21	0,37	0,70
ELRINGKLINGER (275057)	0,39	0,06	0,00	0,16	0,10	0,76
ARVIN (115790)	0,30	0,21	0,16	0,00	0,29	0,74
TKG-Çalı (111307)	0,29	0,37	0,10	0,29	0,00	0,94
ORMETAL (268754)	0,97	0,70	0,76	0,74	0,94	0,00

Tablo 3: 5 tedarikçi arası uzaklıkların saat cinsinden ifadesi

d) Parçaların toplanacağı kamyonlar ve kapasiteleri

Parçalar, kamyon (TIR) ve konteynır olmak üzere 2 çeşit nakliye aracıyla Oyak-Renault Otomobil Fabrikaları’na getirilmektedir. Bu çalışmada hacimler yeterli ve mesafeler kısa olduğundan taşıma aracı kamyonudur.

Kamyonun kapasite bilgisi olarak uzunluğu 13,2 m.dir Hesaplamalarda 13 m. olarak alınmıştır. Toplamda en fazla 23 tonluk malzeme taşıma kapasitesi vardır. Kamyonlar fabrikaya ait değildir, her sene satın alma departmanı tarafından imzalan protokollerle dış kaynak vasıtasıyla (nakliye firmalarından) bu faaliyet gerçekleştirilmektedir.



Şekil 15: Yüklü bir kamyonun 2 yan cepheden ve yukarıdan görünümü

e) Tedarikçilerden ilgili dönemde toplanacak parça miktarları (mL cinsinden) ve ağırlıkları (kg)

Bu iki girdi bilgisi, fabrika sistemlerine ait veri tabanlarından günlük veya haftalık çekilen bilgiler ışığında oluşturulmaktadır.

Veri tabanları SQL formatında çalışmakta ve bilgiyi XLS ve/veya CSV formatında kaydetmeye olanak sağlamaktadır.

Bu bağlamda, aşağıdaki Tablo 4 ve Tablo 5’de yer alan bilgiler bu şekilde elde edilmiştir.

NO	TEDARİKÇİ LİSTESİ	PARÇA MİKTARI (mL)
1	DELPHI	3
2	ELRINGKLINGER	2
3	ARVIN	2
4	TKG-Çalı	4
5	ORMETAL	8

Tablo 4: Tedarikçilerden alınacak günlük malzeme miktarı

NO	TEDARİKÇİ LİSTESİ	PARÇA AĞIRLIKLARI (kg)
1	DELPHI	1
2	ELRİNGKLİNGER	3
3	ARVIN	5
4	TKG-Çalı	8
5	ORMETAL	12

Tablo 5: Tedarikçilerden alınacak günlük malzeme ağırlıkları

f) Tedarikçilerin sevkiyat çalışma saatleri

Bir diğer konu fabrika ve tedarikçilerin çalışma saatleridir. Model bu kısıtları dikkate alarak çalışmakta ve çözüm üretmektedir. Tedarikçiler bu konu hakkındaki gerekli bilgileri ana sanayi Oyak-Renault Oto Fab. A.Ş.'ye mail yoluyla göndermektedir. Şartlar gereği bir değişiklik olması durumunda, yine mail yoluyla irtibat halinde oldukları nakliye servisi çalışanlarına bilgi vermektedirler.

Çalışmanın yapıldığı 2014 Kış, 2015 İlkbahar döneminde bu 5 tedarikçinin çalışma saatleri Tablo 6'da gösterilmektedir.

NO	TEDARİKÇİ LİSTESİ	EN ERKEN İŞE BAŞLAMA SAATİ	EN GEÇ İŞE BAŞLAMA SAATİ
1	DELPHI	07:00	14:00
2	ELRİNGKLİNGER	08:00	16:00
3	ARVIN	08:00	18:00
4	TKG-Çalı	07:00	18:00
5	ORMETAL	07:00	18:00

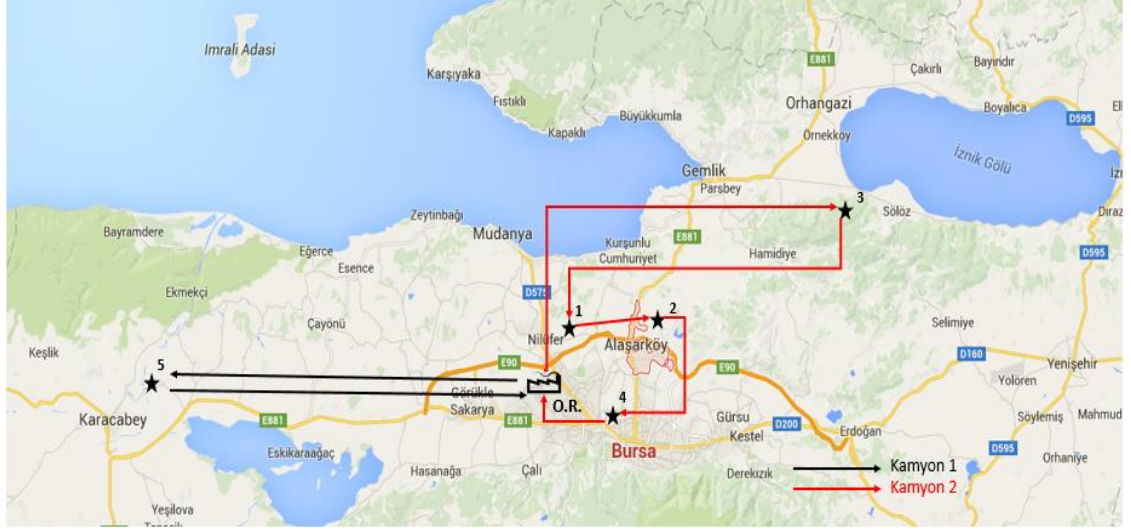
Tablo 6: Tedarikçi çalışma saatleri

g) Bir kamyonun parça yüklenirken kaybettiği zaman

Bir kamyonun fabrikaya girişinin ardından parçaları ilgili rampada yükleme süresi her firmada yaklaşık yarım saattir. Bu yarım saat sonrasında ilgili kamyon fabrikayı terk etmek üzere çıkış kapısına yönelir. Bu nedenle çalışma içerisinde hizmet süresi olarak adlandırılan parça/ambalaj yükleme aktivitesi yarım saat olarak ele alınmıştır.

Araç rotası önceden firmalara ve sürücüyeye iletildiğinden, sürücü elinde bulunan belgedeki toplama akışına uygun biçimde parkurunu tamamlar.

h) Grup A'da elde edilen sonuçlar



Şekil 16: Sonuçların harita üzerinde gösterimi

Modelin çalıştırıldığı bilgisayar, AMD 1,9 Ghz işlemcili ve 4 GB RAM' e sahiptir. Modelin çalışma ve optimum sonucu bulma süresi yaklaşık 1 dakika sürmüştür.

MPL programında Gurobi çözücüsü ile çalıştırılan model, 2 kamyon ve 5 tedarikçiye (T1, T2,..T5) ilişkin optimal çözüm bulmuştur ve çözümü şu şekildedir;

- 1. Kamyon: Oyak-Renault – T5 – Oyak-Renault => 136 km.
- 2. Kamyon: Oyak-Renault – T3 – T1 – T2 – T4 – Oyak-Renault => 67 km.

Toplam alınan mesafe 203 km.

Bu sonuç, harita üzerinde şekil 16 ile gösterilmiştir. Her tedarikçiye 1 kere uğranmış, kamyonlar Oyak-Renault'dan parkura başlamış ve yine Oyak-Renault'ya dönmüşlerdir. Sonuçlara göre kamyonların doluluk oranları mL cinsinden aşağıdaki gibidir:

- 1. Kamyon: $11/13 = \% 85$
- 2. Kamyon: $8/13 = \% 61,5$

Ortalama: % 73,25

Daha önceki sayfalarda paylaşılan 2014 verileriyle tekrar kıyaslamak gerekirse, 2 kamyonla yakalanan ortalama (% 73,25), 2014 yılı ortalamasının (% 70,25) üzerinde çıkmıştır. Bununla birlikte 2. kamyonun rotasında konumu olan bir tedarikçinin de modele eklenmesiyle bu kamyonun da doluluk oranı artırılmış olacaktır. Bu gibi

çözümlere yönelmek, modelin boyutunun artmasına neden olmakta ama aynı zamanda optimal çözümün de değişmesini sağlamaktadır.

Modelin kodlarının açılmış hali Ek B’de, ortaya koyduğu sonuçların detayları ise Ek C’de bulunmaktadır.

4.5.2 Grup B, C ve D Çözümleri ve Grup A ile karşılaştırma

Aynı bilgisayarda, bu defa XA adlı çözücü ile yaklaşık 45 dakikada sonuç elde edilebilmiştir. Bu sonuçlar Grup B olarak tanımladığımız, modelin 10 tedarikçi için kullanıldığı çalışmada ortaya çıkan sonuçları ifade etmektedir. Grup B’de yer alan tedarikçiler, sıraları ve diğer bilgiler aşağıdaki gibidir.

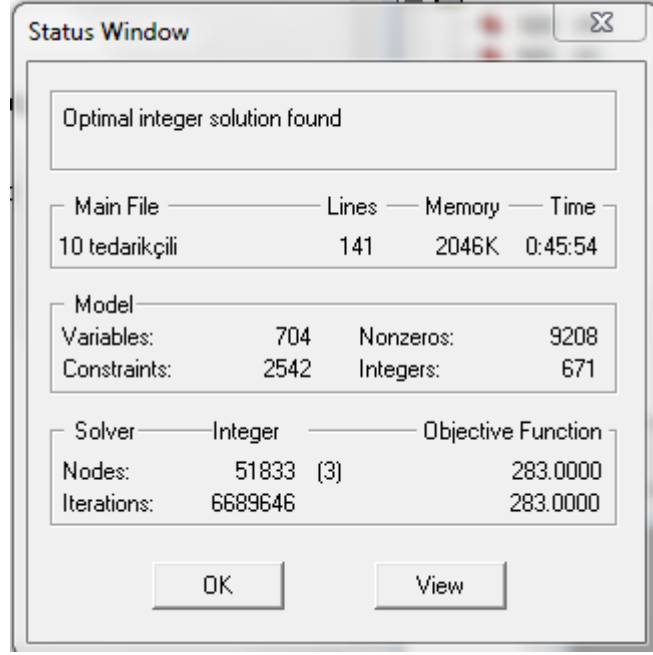
NO	TEDARİKÇİ LİSTESİ	TEDARİKÇİ KODU	ENLEM	BOYLAM
1	ORMETAL	268754	40,225498	28,37273
2	DELPHI	115238	40,284639	28,94268
3	ELRINGKLINGER	275057	40,18257	29,06687
4	ARVIN	115790	40,440319	29,34604
5	TKG-Çalı	111307	40,18257	29,06687
6	RIETER	115226	40,235517	28,93473
7	MATAY	111294-01	40,178763	28,38396
8	ERKURT-1	115121	40,18257	29,06687
9	SEGER	115771	40,232151	29,00927
10	TKG	111307	40,284639	28,94268

Buna göre ortaya çıkan optimum sonuç aşağıdaki gibidir.

- Kamyon 1: 1 – 5 – 10 – 1
- Kamyon 2: 1 – 9 – 1
- Kamyon 3: 1 – 4 – 3 – 2 – 11 - 1
- Kamyon 4: 1 – 7 – 6 – 8 – 1

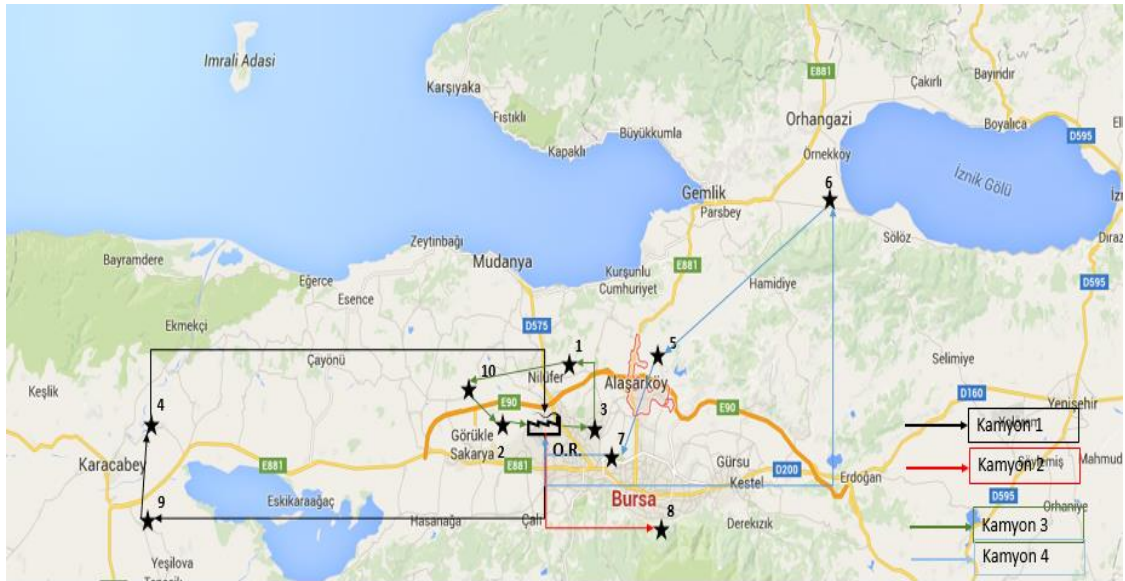
Toplam alınan mesafe 283 km.

Şekil 17 ise çözüm raporunun görsel halidir.



Şekil 17: Çözüm raporu

Şekil 18 ise çözümün harita üzerindeki görselini yansıtmaktadır.



Şekil 18: Çözümün harita üzerindeki görseli

Grup C ve Grup D senaryolarında ise veri yükü oldukça fazla ve gerekli olan iterasyon sayısı da yüksektir. Bu nedenle çalışma için kullanılan MPL öğrenci versiyonu limitleri gerekli hesapları yapmaya yeterli olmadığından herhangi bir sonuç elde edilememiştir.

Bununla birlikte Grup C senaryosunda sadece XA çözücüsü çalışmış olup, 15 saatlik süre zarfında da halen sonuç vermediği görülmüştür.

Senaryolar hakkında karşılaştırmalı bilgiler Tablo 7’de gösterilmektedir.

Grup İsmi	Tedarikçi Sayısı	Kullanılan Kamyon Sayısı	Çözüm Aracı	En iyi sonuç (km)	İterasyon Sayısı	Sonuç Bulma Zamanı (d)
Grup A	5	2	Gurobi	203	1478	1
Grup B	10	4	XA	283	6689646	45
Grup C	15	N/A	N/A	N/A	N/A	N/A
Grup D	20	N/A	N/A	N/A	N/A	N/A

Tablo 7: Senaryolar hakkında karşılaştırmalı bilgiler

5. SONUÇ

Yoğun rekabet ortamında; karmaşık tedarik zincirinin, lojistik ve taşımacılık süreçlerinin etkin bir şekilde yönetilebilmesi, süreçlerin tamamında katma değerli çıktılar yaratılabilmesi ve rekabet avantajı sağlanabilmesi ile gerçekleştirilebilir.

Lojistik faaliyetlerin gerçekleştirilmesinde dikkat edilmesi gereken nokta müşteri ihtiyaçlarını eksiksiz olarak karşılayacak en iyi dağıtım şeklinin planlanmasıdır. Bu dağıtım şeklinin planlanmasında, araç yükleme ve rotalama en önemli konulardır. Uygun kapasiteyle yüklenmiş araçların müşteri noktalarına en kısa mesafeyi kullanarak en kısa zamanda ulaşması; toplam dağıtım maliyetlerinin azaltılmasında, müşteri memnuniyetinin artırılmasında ve firmanın rekabet koşullarında daha avantajlı konuma geçmesinde büyük imkân sağlamaktadır. Bunun gerçekleştirilebilmesi de bazı tekniklerden faydalanılarak sağlanır.

Çalışmanın basından itibaren yapılanlar özetlenecek olursa, ilk olarak Araç Rotalama Problemlerinin ana hatlarıyla anlaşılması sağlanmıştır. Çözüm yöntemleri olarak kesin ve sezgisel yöntemlere ilişkin bilgiler de ayrıca verilmiş ve konu detaylı olarak incelenerek bir kaynak taraması yapılmıştır.

Araç rotalama problemlerinden olan Kesin Zaman Pencereli Araç Rotalama Problemi bu çalışmanın tanımında kullanılmış ve materyal bölümünde detaylı açıklama yapılmıştır. Problemin mevcut matematiksel modeli üzerinde durulmuş ve kesin çözümün nasıl elde edileceğine ilişkin teorik bir çalışma yapılmıştır.

Bu çalışmada tedarikçi sayısının 10 ve altında olduğu uygulamalar, çalışmayı talep eden şirketin tolere edebileceği sürelerde optimum sonuçlar vermiştir. Bu bağlamda şirket, aktiviteyi 10'arlı gruplara ayırabildiği ölçüde bu çalışmayı uygulabilmektedir.

Bununla birlikte, gerek literatürde incelenen çalışmalarda, problemin NP-hard oluşu ve problem boyutunun artmasıyla birlikte çözüm süresinin üstel olarak artması sonucu, gerekse kurulan modeldeki gözlemlerimizde, çözüm için yapılan çalışmaların daha çok meta sezgisel yöntemler üzerinden gerçekleştirilmesinin gerekliliği görülmüştür.

Bu konu üzerinde arařtırma yapılacak ve ileride kullanılacak meta sezgisel yöntem, son dönemlerde geniş bir arařtırma alanı bulan yapay sinir ağlarıdır.

Bu çalışma, literatürde az bulunan zaman pencereli araç rotalama problemleri ile ilgili olarak yeni bir bakış açısı geliřtirmektedir. Gerçek yaşamdan alınan verilerle çözüme ulařılmış ve kurulan model seri hayatta tatbik edilerek çalıştığı ispatlanmıştır.

KAYNAKLAR

- Anonim, 1996.** Logistix Partners Oy, Helsinki, FI.
- Anonim, 1998.** Canadian Association of Logistics Management, <http://www.calm.org/calm/>
- Anonim, 1998.** Council of Logistics Management, <http://www.clm1.org/mission.html>
- Anonim, 1998.** http://www.bts.gov/publications/national_transportation_statistics
- Anonim, 2012.** Merriam-Webster Online Dictionary
- Anonim, 2013.** Oxford English Dictionary
- Anonim, 2014.** www.yapay-zeka.org
- Anonim, 2015.** http://en.wikipedia.org/wiki/Branch_and_cut
- Anonim, 2015.** <http://en.wikipedia.org/wiki/Logistics>
- Anonim, 2015.** <http://en.wikipedia.org/wiki/Transport>
- Anonim, 2015.** <http://www.bernabe.dorronsoro.es/vrp/>
- Balakrishnan, N., 1993.** Simple Heuristics for the Vehicle Routing Problem with Soft Time Windows. *Journal of the Operational Research Society*, 44, 3, 279-287 (1993).
- Bianchessi, N., Righini, G., 2009.** “Heuristic Algorithms For The Vehicle Routing Problem With Simultaneous Pick-Up And Delivery”, *Computers & Operations Research*, 34, 578- 594.
- Biggs, N.L., Lloyd, E. K., and Wilson, R.J., 1976.** ”Graph Theory 1736–1936”, Clarendon Press, Oxford.
- Bowerman, R., Hall, B. and Calomai, P., 1995.** “ A multi-objective Optimization Approach to Urban School Bus Routing: Formulation and Solution Method”, *Transportation Research Part A: Policy and Practice*, 29(2), 107*123.
- Çetin, S., and Gencer, C., 2010.** Kesin Zaman Pencereli Eş Zamanlı Dağıtım Toplamalı Araç Rotalama Problemi: Matematiksel Model, *Gazi Ü.Müh.Fak. Dergisi*, Cilt.25, No:3, 575-585.
- Çetin, S., Özkütük, E., and Gencer, C., 2011.** “Heterojen Araç Filolu Eş Zamanlı Dağıtım Toplamalı Araç Rotalama Problemi İçin Bir Karar Destek Sistemi”, *Int.J.of.Research and Development*, Vol.3, No.1.
- Dantzig, G.B., Ramser, J.H., 1959.** The Truck Dispatching Problem, *Management Science*, 6, 80-91.

Demirciođlu M., 2009. Araç Rotalama Probleminin Sezgisel Bir Yaklaşım İle Çözömlenmesi Üzerine Bir Uygulama Çukurova Üniversitesi Sosyal Bilimler Enstitüsü, Doktora Tezi, 163s.

Desrochers, M., Laporte, G., 1991.“Improvements and extensions to the Miller–Tucker–Zemlin subtour elimination constraints”, *Operations Research Letters*, 10, 27-36,.

Dethloff, J. 2001. “Vehicle Routing And Reverse Logistics: The Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, *OR Spektrum*, 23, 79-96 (2001).

Dorigo M., DiCaro G., Gambardella L. M., 1999. Ant Algorithms for Discrete Optimization. *Artificial Life* 5, 137-172

Erol V., 2006. “Araç Rotalama Problemleri İçin Populasyon Ve Komşuluk Tabanlı Metasezgisel Bir Algoritmanın Tasarımı Ve Uygulaması”, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi.

Gajpal, Y., Abad, P., 2009. “An Ant Colony System (ACS) For Vehicle Routing Problem With Simultaneous Delivery And Pick Up”, *Computers & Operations Research*, 36, 3215- 223.

Gencer, C., Yaşa, Ö., 2007. ‘Ulaştırma Komutanlığı Ring Seferlerinin Eş Zamanlı Dağıtım Toplama Karar DestekSistemi”, *Gazi Ü.Müh.Fak. Dergisi*, Cilt.22, No:3, 437-449

Golden, Raghavan, Wasil, 2005. *The Next Wave in Computing, Optimization, and Decision Technologies*”, Springer Science Business Media, New York, sf108.

Kachitvichyanukul, V., 2009. “A Particle Swarm Optimization For The Vehicle Routing Problem With Simultaneous Pickup And Delivery”, *Computers & Operations Research*, 36, 1693-1702.

Kara, İ., Laporte, G., Bektas, T., 2004. “A note on the lifted Miller–Tucker–Zemlin subtour elimination constraints for the capacitated vehicle routing problem”, *European Journal of Operational Research*, 158, 793-795.

Kearny, A. T., 1984. “Measuring and Improving Productivity in Physical Distribution”, A report prepared for the National Council of Physical Distribution Management, Oak Brook, IL.

Kulkarni, R. V., Bhave, P. R., 1985. “Integer programming formulations of vehicle routing problems”, *European Journal of Operational Research*, 20, 58-67.

Laporte G., 1992. “The Vehicle Routing Problem: An overview of exact and approximate algorithms”, *European Journal of Operational Research*, 59, 345-358.

Laporte, G., Nobert, Y. and Tailleffer, S., 1987. “ Solving a family of Multi-Depot Vehicle Routing and Location Routing Problems”, *Cahier du Gerad G-87-10*, Ecole des Haltes Etudes Commercial de Montreal.

- Miller, C. E., Tucker, A. W., Zemlin, R. A., 1960.** “Integer programming formulations and traveling salesman problems”, *Journal of the ACM*, 7, 326-329.
- Min, H., 1989.** The Multiple Vehicle Routing Problem with Simultaneous Delivery and Pickup Points, *Transportation Research*, 23A, 377-386.
- Montane F.A.T, Galvao RD., 2006.** “A Tabu Search Algorithm For the Vehicle Routing Problem With Simultaneous Pick Up And Delivery Service”, *Computers & Operations Research*, 33, 595-619.
- Nagy, G. & Salhi, S., 2005.** “Heuristic Algorithms For Single And Multiple Depot Vehicle Routing Problems With Pickups And Deliveries”, *European Journal Of Operational Research*, 162, 1, 126-141.
- Savelsbergh, M., W.P. and Sol, M., 1995.** “The General Pick-up and Delivery Problem”, *Transportation Science* 29, 17-29.
- Serin S., 2009.** Karınca Kolonisi Yaklaşımıyla Karayolu Üstyapı Rutin Bakım Çalışmalarının Planlanması. Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 99s.
- Suna, Ç., Özkütük, E., Gencer, C., 2011.** ‘Heterojen Araç Filolu Eş Zamanlı Dağıtım Toplamalı Araç Rotalama Problemi İçin Bir Karar Destek Sistemi.’ *International Journal of Research and Development*, 3, 1.
- Şeker, Ş., 2007.** Araç Rotalama Problemleri ve Zaman Pencere Stokastik Araç Rotalama Problemine Genetik Algoritma Yaklaşımı”, *Yüksek Lisans Tezi*, Yıldız Teknik Üniversitesi.
- Tan, K.C., 2000.** A Framework of Supply Chain Management Literature, *European Journal of Purchasing Supply Management*.
- Toth P., and Vigo D., 2002.** “Models, Relaxations and Exact Approaches for The Capacitated Vehicle Routing Problem, *Discrete Applied Mathematics*, 123, 1-3, 487-512.
- Toth P., Vigo, D., 2002.** ”The Vehicle Routing Problem”, *Siam Monographs on Discrete Mathematics and Applications*, Philadelphia.
- Touzet, C., 1992.** *Les Reseaux de Neurones Artificiels, Cours, Exercices et Travaux Pratiques*
- Tüfekçiler, H., 2008.** ”İki Amaçlı Açık Araç Rotalama Problemi İçin Bir Çözüm Yaklaşımı”, *Yüksek Lisans Tezi*, Eskişehir Osmangazi Üniversitesi.
- Zachariadis, E.E., Tarantilis, C.D., Kiranoudis, C. T. 2009.** “Hybrid Metaheuristic Algorithm For The Vehicle Routing Problem With Simultaneous Delivery And Pick-Up Service”, *Expert System with Applications*, 36, 1070-1081.

EKLER

Ek A: Sefer/Rota Planı Örneđi

Bölge	Sefer_No	Araç tipi	Plaka	Tedarikçi	Planlanan Varış Zamanı	Planlanan Çıkış Zamanı
BURSA	4201	TIR	16 BYK 35	AUTOLIV	12:00	12:45
				ROZMAS	13:00	13:45
				KALE	14:15	15:15
				SOFITEC	15:30	16:15
BURSA	4222	TIR	16 KJN 99	MECAPLAST	13:00	13:45
				FERRO	14:15	15:15

Tablo A.1: Bursa'ya ait 2 bölgede 16.04.2015 tarihli sefer planı

Ek B: Modelin kodlarının açılmış hali

\ v7 gurobi ile çalışın küçük sample - Tezde yer alan.lp

\

\ Generated with the MPL Modeling System

\ Copyright (c) 1988-2014 Maximal Software

\

\ Date: June 1, 2015

\ Time: 23:07

\

\ Constraints: 401

\ Variables: 132

\ Integers: 126

\ Nonzeros: 1208

\ Density: 2 %

MINIMIZE

Yol: 31 X121 + 31 X122 + 27 X131 + 27 X132 + 21 X141 + 21 X142
+ 20 X151 + 20 X152 + 68 X161 + 68 X162 + 31 X211 + 31 X212
+ 4 X231 + 4 X232 + 15 X241 + 15 X242 + 26 X251 + 26 X252
+ 49 X261 + 49 X262 + 27 X311 + 27 X312 + 4 X321 + 4 X322
+ 11 X341 + 11 X342 + 7 X351 + 7 X352 + 53 X361 + 53 X362
+ 21 X411 + 21 X412 + 15 X421 + 15 X422 + 11 X431 + 11 X432
+ 20 X451 + 20 X452 + 52 X461 + 52 X462 + 20 X511 + 20 X512
+ 26 X521 + 26 X522 + 7 X531 + 7 X532 + 20 X541 + 20 X542
+ 66 X561 + 66 X562 + 68 X611 + 68 X612 + 49 X621 + 49 X622
+ 53 X631 + 53 X632 + 52 X641 + 52 X642 + 66 X651 + 66 X652

SUBJECT TO

$$\begin{aligned} \text{K402: } & X121 + X122 + X221 + X222 + X321 + X322 + X421 + X422 + X521 \\ & + X522 + X621 + X622 = 1 \end{aligned}$$

$$\begin{aligned} \text{K403: } & X131 + X132 + X231 + X232 + X331 + X332 + X431 + X432 + X531 \\ & + X532 + X631 + X632 = 1 \end{aligned}$$

$$\begin{aligned} \text{K404: } & X141 + X142 + X241 + X242 + X341 + X342 + X441 + X442 + X541 \\ & + X542 + X641 + X642 = 1 \end{aligned}$$

$$\begin{aligned} \text{K405: } & X151 + X152 + X251 + X252 + X351 + X352 + X451 + X452 + X551 \\ & + X552 + X651 + X652 = 1 \end{aligned}$$

$$\begin{aligned} \text{K406: } & X161 + X162 + X261 + X262 + X361 + X362 + X461 + X462 + X561 \\ & + X562 + X661 + X662 = 1 \end{aligned}$$

$$\text{K1111: } X111 = 0$$

$$\text{K1112: } X112 = 0$$

$$\text{K1221: } X221 = 0$$

$$\text{K1222: } X222 = 0$$

$$\text{K1331: } X331 = 0$$

$$\text{K1332: } X332 = 0$$

$$\text{K1441: } X441 = 0$$

$$\text{K1442: } X442 = 0$$

$$\text{K1551: } X551 = 0$$

$$\text{K1552: } X552 = 0$$

$$\text{K1661: } X661 = 0$$

$$\text{K1662: } X662 = 0$$

$$\begin{aligned} \text{K111: } & X121 - X211 - X231 - X241 - X251 - X261 + X321 + X421 + X521 \\ & + X621 = 0 \end{aligned}$$

$$\begin{aligned} \text{K112: } & X122 - X212 - X232 - X242 - X252 - X262 + X322 + X422 + X522 \\ & + X622 = 0 \end{aligned}$$

$$\text{K81: } X131 + X231 - X311 - X321 - X341 - X351 - X361 + X431 + X531$$

$$+ X631 = 0$$

$$\text{K82: } X132 + X232 - X312 - X322 - X342 - X352 - X362 + X432 + X532$$

$$+ X632 = 0$$

$$\text{K71: } X141 + X241 + X341 - X411 - X421 - X431 - X451 - X461 + X541$$

$$+ X641 = 0$$

$$\text{K72: } X142 + X242 + X342 - X412 - X422 - X432 - X452 - X462 + X542$$

$$+ X642 = 0$$

$$\text{K61: } X151 + X251 + X351 + X451 - X511 - X521 - X531 - X541 - X561$$

$$+ X651 = 0$$

$$\text{K62: } X152 + X252 + X352 + X452 - X512 - X522 - X532 - X542 - X562$$

$$+ X652 = 0$$

$$\text{K51: } X161 + X261 + X361 + X461 + X561 - X611 - X621 - X631 - X641$$

$$- X651 = 0$$

$$\text{K52: } X162 + X262 + X362 + X462 + X562 - X612 - X622 - X632 - X642$$

$$- X652 = 0$$

$$\text{K1211: } X111 + X121 + X131 + X141 + X151 + X161 = 1$$

$$\text{K1212: } X112 + X122 + X132 + X142 + X152 + X162 = 1$$

$$\text{K2611: } X111 + X211 + X311 + X411 + X511 + X611 = 1$$

$$\text{K2612: } X112 + X212 + X312 + X412 + X512 + X612 = 1$$

$$\text{K13121: } -111 + 121 - 1000000000 X121 \geq -999999997$$

$$\text{K13122: } -112 + 122 - 1000000000 X122 \geq -999999997$$

$$\text{K13131: } -111 + 131 - 1000000000 X131 \geq -999999998$$

$$\text{K13132: } -112 + 132 - 1000000000 X132 \geq -999999998$$

$$\text{K13141: } -111 + 141 - 1000000000 X141 \geq -999999998$$

$$\text{K13142: } -112 + 142 - 1000000000 X142 \geq -999999998$$

$$\text{K13151: } -111 + 151 - 1000000000 X151 \geq -999999996$$

$$\text{K13152: } -112 + 152 - 1000000000 X152 \geq -999999996$$

$$\text{K13161: } -111 + 161 - 1000000000 X161 \geq -999999992$$

K13162: - 112 + 162 - 1000000000 X162 >= - 999999992
K13231: - 121 + 131 - 1000000000 X231 >= - 999999998
K13232: - 122 + 132 - 1000000000 X232 >= - 999999998
K13241: - 121 + 141 - 1000000000 X241 >= - 999999998
K13242: - 122 + 142 - 1000000000 X242 >= - 999999998
K13251: - 121 + 151 - 1000000000 X251 >= - 999999996
K13252: - 122 + 152 - 1000000000 X252 >= - 999999996
K13261: - 121 + 161 - 1000000000 X261 >= - 999999992
K13262: - 122 + 162 - 1000000000 X262 >= - 999999992
K13321: 121 - 131 - 1000000000 X321 >= - 999999997
K13322: 122 - 132 - 1000000000 X322 >= - 999999997
K13341: - 131 + 141 - 1000000000 X341 >= - 999999998
K13342: - 132 + 142 - 1000000000 X342 >= - 999999998
K13351: - 131 + 151 - 1000000000 X351 >= - 999999996
K13352: - 132 + 152 - 1000000000 X352 >= - 999999996
K13361: - 131 + 161 - 1000000000 X361 >= - 999999992
K13362: - 132 + 162 - 1000000000 X362 >= - 999999992
K13421: 121 - 141 - 1000000000 X421 >= - 999999997
K13422: 122 - 142 - 1000000000 X422 >= - 999999997
K13431: 131 - 141 - 1000000000 X431 >= - 999999998
K13432: 132 - 142 - 1000000000 X432 >= - 999999998
K13451: - 141 + 151 - 1000000000 X451 >= - 999999996
K13452: - 142 + 152 - 1000000000 X452 >= - 999999996
K13461: - 141 + 161 - 1000000000 X461 >= - 999999992
K13462: - 142 + 162 - 1000000000 X462 >= - 999999992
K13521: 121 - 151 - 1000000000 X521 >= - 999999997
K13522: 122 - 152 - 1000000000 X522 >= - 999999997
K13531: 131 - 151 - 1000000000 X531 >= - 999999998

K13532: $132 - 152 - 1000000000 X532 \geq -999999998$
 K13541: $141 - 151 - 1000000000 X541 \geq -999999998$
 K13542: $142 - 152 - 1000000000 X542 \geq -999999998$
 K13561: $-151 + 161 - 1000000000 X561 \geq -999999992$
 K13562: $-152 + 162 - 1000000000 X562 \geq -999999992$
 K13621: $121 - 161 - 1000000000 X621 \geq -999999997$
 K13622: $122 - 162 - 1000000000 X622 \geq -999999997$
 K13631: $131 - 161 - 1000000000 X631 \geq -999999998$
 K13632: $132 - 162 - 1000000000 X632 \geq -999999998$
 K13641: $141 - 161 - 1000000000 X641 \geq -999999998$
 K13642: $142 - 162 - 1000000000 X642 \geq -999999998$
 K13651: $151 - 161 - 1000000000 X651 \geq -999999996$
 K13652: $152 - 162 - 1000000000 X652 \geq -999999996$
 K4111: $111 = 0$
 K4112: $112 = 0$
 K52121: $-P11 + P21 - 1000000000 X121 \geq -999999999$
 K52122: $-P12 + P22 - 1000000000 X122 \geq -999999999$
 K52131: $-P11 + P31 - 1000000000 X131 \geq -999999997$
 K52132: $-P12 + P32 - 1000000000 X132 \geq -999999997$
 K52141: $-P11 + P41 - 1000000000 X141 \geq -999999995$
 K52142: $-P12 + P42 - 1000000000 X142 \geq -999999995$
 K52151: $-P11 + P51 - 1000000000 X151 \geq -999999992$
 K52152: $-P12 + P52 - 1000000000 X152 \geq -999999992$
 K52161: $-P11 + P61 - 1000000000 X161 \geq -999999988$
 K52162: $-P12 + P62 - 1000000000 X162 \geq -999999988$
 K52231: $-P21 + P31 - 1000000000 X231 \geq -999999997$
 K52232: $-P22 + P32 - 1000000000 X232 \geq -999999997$
 K52241: $-P21 + P41 - 1000000000 X241 \geq -999999995$

K52242: - P22 + P42 - 1000000000 X242 >= - 999999995
K52251: - P21 + P51 - 1000000000 X251 >= - 999999992
K52252: - P22 + P52 - 1000000000 X252 >= - 999999992
K52261: - P21 + P61 - 1000000000 X261 >= - 999999988
K52262: - P22 + P62 - 1000000000 X262 >= - 999999988
K52321: P21 - P31 - 1000000000 X321 >= - 999999999
K52322: P22 - P32 - 1000000000 X322 >= - 999999999
K52341: - P31 + P41 - 1000000000 X341 >= - 999999995
K52342: - P32 + P42 - 1000000000 X342 >= - 999999995
K52351: - P31 + P51 - 1000000000 X351 >= - 999999992
K52352: - P32 + P52 - 1000000000 X352 >= - 999999992
K52361: - P31 + P61 - 1000000000 X361 >= - 999999988
K52362: - P32 + P62 - 1000000000 X362 >= - 999999988
K52421: P21 - P41 - 1000000000 X421 >= - 999999999
K52422: P22 - P42 - 1000000000 X422 >= - 999999999
K52431: P31 - P41 - 1000000000 X431 >= - 999999997
K52432: P32 - P42 - 1000000000 X432 >= - 999999997
K52451: - P41 + P51 - 1000000000 X451 >= - 999999992
K52452: - P42 + P52 - 1000000000 X452 >= - 999999992
K52461: - P41 + P61 - 1000000000 X461 >= - 999999988
K52462: - P42 + P62 - 1000000000 X462 >= - 999999988
K52521: P21 - P51 - 1000000000 X521 >= - 999999999
K52522: P22 - P52 - 1000000000 X522 >= - 999999999
K52531: P31 - P51 - 1000000000 X531 >= - 999999997
K52532: P32 - P52 - 1000000000 X532 >= - 999999997
K52541: P41 - P51 - 1000000000 X541 >= - 999999995
K52542: P42 - P52 - 1000000000 X542 >= - 999999995
K52561: - P51 + P61 - 1000000000 X561 >= - 999999988

K52562: - P52 + P62 - 1000000000 X562 >= - 999999988
K52621: P21 - P61 - 1000000000 X621 >= - 999999999
K52622: P22 - P62 - 1000000000 X622 >= - 999999999
K52631: P31 - P61 - 1000000000 X631 >= - 999999997
K52632: P32 - P62 - 1000000000 X632 >= - 999999997
K52641: P41 - P61 - 1000000000 X641 >= - 999999995
K52642: P42 - P62 - 1000000000 X642 >= - 999999995
K52651: P51 - P61 - 1000000000 X651 >= - 999999992
K52652: P52 - P62 - 1000000000 X652 >= - 999999992
K5311: P11 = 0
K5312: P12 = 0
K1421: l21 <= 13
K1422: l22 <= 13
K1431: l31 <= 13
K1432: l32 <= 13
K1441: l41 <= 13
K1442: l42 <= 13
K1451: l51 <= 13
K1452: l52 <= 13
K1461: l61 <= 13
K1462: l62 <= 13
K5321: P21 <= 23
K5322: P22 <= 23
K5331: P31 <= 23
K5332: P32 <= 23
K5341: P41 <= 23
K5342: P42 <= 23
K5351: P51 <= 23

K5352: $P52 \leq 23$
 K5361: $P61 \leq 23$
 K5362: $P62 \leq 23$
 K15121: $-Y11 + Y21 - 1000000000 X121 \geq -999999999$
 K15122: $-Y12 + Y22 - 1000000000 X122 \geq -999999999$
 K15131: $-Y11 + Y31 - 1000000000 X131 \geq -999999999$
 K15132: $-Y12 + Y32 - 1000000000 X132 \geq -999999999$
 K15141: $-Y11 + Y41 - 1000000000 X141 \geq -999999999$
 K15142: $-Y12 + Y42 - 1000000000 X142 \geq -999999999$
 K15151: $-Y11 + Y51 - 1000000000 X151 \geq -999999999$
 K15152: $-Y12 + Y52 - 1000000000 X152 \geq -999999999$
 K15161: $-Y11 + Y61 - 1000000000 X161 \geq -999999999$
 K15162: $-Y12 + Y62 - 1000000000 X162 \geq -999999999$
 K15231: $-Y21 + Y31 - 1000000000 X231 \geq -999999999$
 K15232: $-Y22 + Y32 - 1000000000 X232 \geq -999999999$
 K15241: $-Y21 + Y41 - 1000000000 X241 \geq -999999999$
 K15242: $-Y22 + Y42 - 1000000000 X242 \geq -999999999$
 K15251: $-Y21 + Y51 - 1000000000 X251 \geq -999999999$
 K15252: $-Y22 + Y52 - 1000000000 X252 \geq -999999999$
 K15261: $-Y21 + Y61 - 1000000000 X261 \geq -999999999$
 K15262: $-Y22 + Y62 - 1000000000 X262 \geq -999999999$
 K15321: $Y21 - Y31 - 1000000000 X321 \geq -999999999$
 K15322: $Y22 - Y32 - 1000000000 X322 \geq -999999999$
 K15341: $-Y31 + Y41 - 1000000000 X341 \geq -999999999$
 K15342: $-Y32 + Y42 - 1000000000 X342 \geq -999999999$
 K15351: $-Y31 + Y51 - 1000000000 X351 \geq -999999999$
 K15352: $-Y32 + Y52 - 1000000000 X352 \geq -999999999$
 K15361: $-Y31 + Y61 - 1000000000 X361 \geq -999999999$

K15362: $- Y32 + Y62 - 1000000000 X362 \geq - 999999999$
 K15421: $Y21 - Y41 - 1000000000 X421 \geq - 999999999$
 K15422: $Y22 - Y42 - 1000000000 X422 \geq - 999999999$
 K15431: $Y31 - Y41 - 1000000000 X431 \geq - 999999999$
 K15432: $Y32 - Y42 - 1000000000 X432 \geq - 999999999$
 K15451: $- Y41 + Y51 - 1000000000 X451 \geq - 999999999$
 K15452: $- Y42 + Y52 - 1000000000 X452 \geq - 999999999$
 K15461: $- Y41 + Y61 - 1000000000 X461 \geq - 999999999$
 K15462: $- Y42 + Y62 - 1000000000 X462 \geq - 999999999$
 K15521: $Y21 - Y51 - 1000000000 X521 \geq - 999999999$
 K15522: $Y22 - Y52 - 1000000000 X522 \geq - 999999999$
 K15531: $Y31 - Y51 - 1000000000 X531 \geq - 999999999$
 K15532: $Y32 - Y52 - 1000000000 X532 \geq - 999999999$
 K15541: $Y41 - Y51 - 1000000000 X541 \geq - 999999999$
 K15542: $Y42 - Y52 - 1000000000 X542 \geq - 999999999$
 K15561: $- Y51 + Y61 - 1000000000 X561 \geq - 999999999$
 K15562: $- Y52 + Y62 - 1000000000 X562 \geq - 999999999$
 K15621: $Y21 - Y61 - 1000000000 X621 \geq - 999999999$
 K15622: $Y22 - Y62 - 1000000000 X622 \geq - 999999999$
 K15631: $Y31 - Y61 - 1000000000 X631 \geq - 999999999$
 K15632: $Y32 - Y62 - 1000000000 X632 \geq - 999999999$
 K15641: $Y41 - Y61 - 1000000000 X641 \geq - 999999999$
 K15642: $Y42 - Y62 - 1000000000 X642 \geq - 999999999$
 K15651: $Y51 - Y61 - 1000000000 X651 \geq - 999999999$
 K15652: $Y52 - Y62 - 1000000000 X652 \geq - 999999999$
 K4211: $Y11 = 0$
 K4212: $Y12 = 0$
 K16121: $S11 - S21 + 1000000000 X121 \leq 999999999.60000002$

K16122: $S_{12} - S_{22} + 1000000000 X_{122} \leq 999999999.60000002$
 K16131: $S_{11} - S_{31} + 1000000000 X_{131} \leq 999999999.60000002$
 K16132: $S_{12} - S_{32} + 1000000000 X_{132} \leq 999999999.60000002$
 K16141: $S_{11} - S_{41} + 1000000000 X_{141} \leq 999999999.70000005$
 K16142: $S_{12} - S_{42} + 1000000000 X_{142} \leq 999999999.70000005$
 K16151: $S_{11} - S_{51} + 1000000000 X_{151} \leq 999999999.70000005$
 K16152: $S_{12} - S_{52} + 1000000000 X_{152} \leq 999999999.70000005$
 K16161: $S_{11} - S_{61} + 1000000000 X_{161} \leq 999999999$
 K16162: $S_{12} - S_{62} + 1000000000 X_{162} \leq 999999999$
 K16231: $S_{21} - S_{31} + 1000000000 X_{231} \leq 999999999.39999998$
 K16232: $S_{22} - S_{32} + 1000000000 X_{232} \leq 999999999.39999998$
 K16241: $S_{21} - S_{41} + 1000000000 X_{241} \leq 999999999.29999995$
 K16242: $S_{22} - S_{42} + 1000000000 X_{242} \leq 999999999.29999995$
 K16251: $S_{21} - S_{51} + 1000000000 X_{251} \leq 999999999.10000002$
 K16252: $S_{22} - S_{52} + 1000000000 X_{252} \leq 999999999.10000002$
 K16261: $S_{21} - S_{61} + 1000000000 X_{261} \leq 999999998.79999995$
 K16262: $S_{22} - S_{62} + 1000000000 X_{262} \leq 999999998.79999995$
 K16321: $-S_{21} + S_{31} + 1000000000 X_{321} \leq 999999999.39999998$
 K16322: $-S_{22} + S_{32} + 1000000000 X_{322} \leq 999999999.39999998$
 K16341: $S_{31} - S_{41} + 1000000000 X_{341} \leq 999999999.29999995$
 K16342: $S_{32} - S_{42} + 1000000000 X_{342} \leq 999999999.29999995$
 K16351: $S_{31} - S_{51} + 1000000000 X_{351} \leq 999999999.39999998$
 K16352: $S_{32} - S_{52} + 1000000000 X_{352} \leq 999999999.39999998$
 K16361: $S_{31} - S_{61} + 1000000000 X_{361} \leq 999999998.70000005$
 K16362: $S_{32} - S_{62} + 1000000000 X_{362} \leq 999999998.70000005$
 K16421: $-S_{21} + S_{41} + 1000000000 X_{421} \leq 999999999.29999995$
 K16422: $-S_{22} + S_{42} + 1000000000 X_{422} \leq 999999999.29999995$
 K16431: $-S_{31} + S_{41} + 1000000000 X_{431} \leq 999999999.29999995$

K16432: $- S32 + S42 + 1000000000 X432 \leq 999999999.29999995$
 K16451: $S41 - S51 + 1000000000 X451 \leq 999999999.20000005$
 K16452: $S42 - S52 + 1000000000 X452 \leq 999999999.20000005$
 K16461: $S41 - S61 + 1000000000 X461 \leq 999999998.79999995$
 K16462: $S42 - S62 + 1000000000 X462 \leq 999999998.79999995$
 K16521: $- S21 + S51 + 1000000000 X521 \leq 999999999.10000002$
 K16522: $- S22 + S52 + 1000000000 X522 \leq 999999999.10000002$
 K16531: $- S31 + S51 + 1000000000 X531 \leq 999999999.39999998$
 K16532: $- S32 + S52 + 1000000000 X532 \leq 999999999.39999998$
 K16541: $- S41 + S51 + 1000000000 X541 \leq 999999999.20000005$
 K16542: $- S42 + S52 + 1000000000 X542 \leq 999999999.20000005$
 K16561: $S51 - S61 + 1000000000 X561 \leq 999999998.5$
 K16562: $S52 - S62 + 1000000000 X562 \leq 999999998.5$
 K16621: $- S21 + S61 + 1000000000 X621 \leq 999999998.79999995$
 K16622: $- S22 + S62 + 1000000000 X622 \leq 999999998.79999995$
 K16631: $- S31 + S61 + 1000000000 X631 \leq 999999998.70000005$
 K16632: $- S32 + S62 + 1000000000 X632 \leq 999999998.70000005$
 K16641: $- S41 + S61 + 1000000000 X641 \leq 999999998.79999995$
 K16642: $- S42 + S62 + 1000000000 X642 \leq 999999998.79999995$
 K16651: $- S51 + S61 + 1000000000 X651 \leq 999999998.5$
 K16652: $- S52 + S62 + 1000000000 X652 \leq 999999998.5$
 K4211: $S11 = 7$
 K4212: $S12 = 7$
 K1721: $- S21 + 7 X121 + 7 X221 + 7 X321 + 7 X421 + 7 X521 + 7 X621$
 ≤ 0
 K1722: $- S22 + 7 X122 + 7 X222 + 7 X322 + 7 X422 + 7 X522 + 7 X622$
 ≤ 0
 K1731: $- S31 + 8 X131 + 8 X231 + 8 X331 + 8 X431 + 8 X531 + 8 X631$

≤ 0
 K1732: $-S_{32} + 8 X_{132} + 8 X_{232} + 8 X_{332} + 8 X_{432} + 8 X_{532} + 8 X_{632}$
 ≤ 0
 K1741: $-S_{41} + 8 X_{141} + 8 X_{241} + 8 X_{341} + 8 X_{441} + 8 X_{541} + 8 X_{641}$
 ≤ 0
 K1742: $-S_{42} + 8 X_{142} + 8 X_{242} + 8 X_{342} + 8 X_{442} + 8 X_{542} + 8 X_{642}$
 ≤ 0
 K1751: $-S_{51} + 7 X_{151} + 7 X_{251} + 7 X_{351} + 7 X_{451} + 7 X_{551} + 7 X_{651}$
 ≤ 0
 K1752: $-S_{52} + 7 X_{152} + 7 X_{252} + 7 X_{352} + 7 X_{452} + 7 X_{552} + 7 X_{652}$
 ≤ 0
 K1761: $-S_{61} + 7 X_{161} + 7 X_{261} + 7 X_{361} + 7 X_{461} + 7 X_{561} + 7 X_{661}$
 ≤ 0
 K1762: $-S_{62} + 7 X_{162} + 7 X_{262} + 7 X_{362} + 7 X_{462} + 7 X_{562} + 7 X_{662}$
 ≤ 0
 K1821: $S_{21} - 14 X_{121} - 14 X_{221} - 14 X_{321} - 14 X_{421} - 14 X_{521}$
 $- 14 X_{621} \leq 0$
 K1822: $S_{22} - 14 X_{122} - 14 X_{222} - 14 X_{322} - 14 X_{422} - 14 X_{522}$
 $- 14 X_{622} \leq 0$
 K1831: $S_{31} - 16 X_{131} - 16 X_{231} - 16 X_{331} - 16 X_{431} - 16 X_{531}$
 $- 16 X_{631} \leq 0$
 K1832: $S_{32} - 16 X_{132} - 16 X_{232} - 16 X_{332} - 16 X_{432} - 16 X_{532}$
 $- 16 X_{632} \leq 0$
 K1841: $S_{41} - 18 X_{141} - 18 X_{241} - 18 X_{341} - 18 X_{441} - 18 X_{541}$
 $- 18 X_{641} \leq 0$
 K1842: $S_{42} - 18 X_{142} - 18 X_{242} - 18 X_{342} - 18 X_{442} - 18 X_{542}$
 $- 18 X_{642} \leq 0$
 K1851: $S_{51} - 20 X_{151} - 20 X_{251} - 20 X_{351} - 20 X_{451} - 20 X_{551}$

- 20 X651 <= 0

K1852: S52 - 20 X152 - 20 X252 - 20 X352 - 20 X452 - 20 X552
- 20 X652 <= 0

K1861: S61 - 20 X161 - 20 X261 - 20 X361 - 20 X461 - 20 X561
- 20 X661 <= 0

K1862: S62 - 20 X162 - 20 X262 - 20 X362 - 20 X462 - 20 X562
- 20 X662 <= 0

K19121: V21 - S11 + 999999999.60000002 X121 <= 1000000000

K19122: V22 - S12 + 999999999.60000002 X122 <= 1000000000

K19131: V31 - S11 + 999999999.60000002 X131 <= 1000000000

K19132: V32 - S12 + 999999999.60000002 X132 <= 1000000000

K19141: V41 - S11 + 999999999.70000005 X141 <= 1000000000

K19142: V42 - S12 + 999999999.70000005 X142 <= 1000000000

K19151: V51 - S11 + 999999999.70000005 X151 <= 1000000000

K19152: V52 - S12 + 999999999.70000005 X152 <= 1000000000

K19161: V61 - S11 + 999999999 X161 <= 1000000000

K19162: V62 - S12 + 999999999 X162 <= 1000000000

K19231: V31 - S21 + 999999999.39999998 X231 <= 1000000000

K19232: V32 - S22 + 999999999.39999998 X232 <= 1000000000

K19241: V41 - S21 + 999999999.29999995 X241 <= 1000000000

K19242: V42 - S22 + 999999999.29999995 X242 <= 1000000000

K19251: V51 - S21 + 999999999.10000002 X251 <= 1000000000

K19252: V52 - S22 + 999999999.10000002 X252 <= 1000000000

K19261: V61 - S21 + 999999998.79999995 X261 <= 1000000000

K19262: V62 - S22 + 999999998.79999995 X262 <= 1000000000

K19321: V21 - S31 + 999999999.39999998 X321 <= 1000000000

K19322: V22 - S32 + 999999999.39999998 X322 <= 1000000000

K19341: V41 - S31 + 999999999.29999995 X341 <= 1000000000

K19342: $V42 - S32 + 999999999.29999995 X342 \leq 1000000000$
 K19351: $V51 - S31 + 999999999.39999998 X351 \leq 1000000000$
 K19352: $V52 - S32 + 999999999.39999998 X352 \leq 1000000000$
 K19361: $V61 - S31 + 999999998.70000005 X361 \leq 1000000000$
 K19362: $V62 - S32 + 999999998.70000005 X362 \leq 1000000000$
 K19421: $V21 - S41 + 999999999.29999995 X421 \leq 1000000000$
 K19422: $V22 - S42 + 999999999.29999995 X422 \leq 1000000000$
 K19431: $V31 - S41 + 999999999.29999995 X431 \leq 1000000000$
 K19432: $V32 - S42 + 999999999.29999995 X432 \leq 1000000000$
 K19451: $V51 - S41 + 999999999.20000005 X451 \leq 1000000000$
 K19452: $V52 - S42 + 999999999.20000005 X452 \leq 1000000000$
 K19461: $V61 - S41 + 999999998.79999995 X461 \leq 1000000000$
 K19462: $V62 - S42 + 999999998.79999995 X462 \leq 1000000000$
 K19521: $V21 - S51 + 999999999.10000002 X521 \leq 1000000000$
 K19522: $V22 - S52 + 999999999.10000002 X522 \leq 1000000000$
 K19531: $V31 - S51 + 999999999.39999998 X531 \leq 1000000000$
 K19532: $V32 - S52 + 999999999.39999998 X532 \leq 1000000000$
 K19541: $V41 - S51 + 999999999.20000005 X541 \leq 1000000000$
 K19542: $V42 - S52 + 999999999.20000005 X542 \leq 1000000000$
 K19561: $V61 - S51 + 999999998.5 X561 \leq 1000000000$
 K19562: $V62 - S52 + 999999998.5 X562 \leq 1000000000$
 K19621: $V21 - S61 + 999999998.79999995 X621 \leq 1000000000$
 K19622: $V22 - S62 + 999999998.79999995 X622 \leq 1000000000$
 K19631: $V31 - S61 + 999999998.70000005 X631 \leq 1000000000$
 K19632: $V32 - S62 + 999999998.70000005 X632 \leq 1000000000$
 K19641: $V41 - S61 + 999999998.79999995 X641 \leq 1000000000$
 K19642: $V42 - S62 + 999999998.79999995 X642 \leq 1000000000$
 K19651: $V51 - S61 + 999999998.5 X651 \leq 1000000000$

K19652: $V52 - S62 + 999999998.5 X652 \leq 1000000000$

K2411: $V11 - S11 \leq 0$

K2412: $V12 - S12 \leq 0$

K2421: $V21 - S21 \leq 0$

K2422: $V22 - S22 \leq 0$

K2431: $V31 - S31 \leq 0$

K2432: $V32 - S32 \leq 0$

K2441: $V41 - S41 \leq 0$

K2442: $V42 - S42 \leq 0$

K2451: $V51 - S51 \leq 0$

K2452: $V52 - S52 \leq 0$

K2461: $V61 - S61 \leq 0$

K2462: $V62 - S62 \leq 0$

K2511: $Y11 \geq 0$

K2512: $Y12 \geq 0$

K2521: $Y21 \geq 0$

K2522: $Y22 \geq 0$

K2531: $Y31 \geq 0$

K2532: $Y32 \geq 0$

K2541: $Y41 \geq 0$

K2542: $Y42 \geq 0$

K2551: $Y51 \geq 0$

K2552: $Y52 \geq 0$

K2561: $Y61 \geq 0$

K2562: $Y62 \geq 0$

K4511: $S11 \geq 0$

K4512: $S12 \geq 0$

K4521: $S21 \geq 0$

K4522: S22 ≥ 0
K4531: S31 ≥ 0
K4532: S32 ≥ 0
K4541: S41 ≥ 0
K4542: S42 ≥ 0
K4551: S51 ≥ 0
K4552: S52 ≥ 0
K4561: S61 ≥ 0
K4562: S62 ≥ 0
K4611: 111 ≥ 0
K4612: 112 ≥ 0
K4621: 121 ≥ 0
K4622: 122 ≥ 0
K4631: 131 ≥ 0
K4632: 132 ≥ 0
K4641: 141 ≥ 0
K4642: 142 ≥ 0
K4651: 151 ≥ 0
K4652: 152 ≥ 0
K4661: 161 ≥ 0
K4662: 162 ≥ 0
K4711: V11 ≥ 0
K4712: V12 ≥ 0
K4721: V21 ≥ 0
K4722: V22 ≥ 0
K4731: V31 ≥ 0
K4732: V32 ≥ 0
K4741: V41 ≥ 0

K4742: $V42 \geq 0$
K4751: $V51 \geq 0$
K4752: $V52 \geq 0$
K4761: $V61 \geq 0$
K4762: $V62 \geq 0$
K5111: $P11 \geq 0$
K5112: $P12 \geq 0$
K5121: $P21 \geq 0$
K5122: $P22 \geq 0$
K5131: $P31 \geq 0$
K5132: $P32 \geq 0$
K5141: $P41 \geq 0$
K5142: $P42 \geq 0$
K5151: $P51 \geq 0$
K5152: $P52 \geq 0$
K5161: $P61 \geq 0$
K5162: $P62 \geq 0$

BOUNDS

INTEGERS

111
112
121
122
131
132
141
142

I51

I52

I61

I62

Y11

Y12

Y21

Y22

Y31

Y32

Y41

Y42

Y51

Y52

Y61

Y62

V11

V12

V21

V22

V31

V32

V41

V42

V51

V52

V61

V62

S11

S12

S21

S22

S31

S32

S41

S42

S51

S52

S61

S62

X111

X112

X121

X122

X131

X132

X141

X142

X151

X152

X161

X162

X211

X212

X221

X222

X231
X232
X241
X242
X251
X252
X261
X262
X311
X312
X321
X322
X331
X332
X341
X342
X351
X352
X361
X362
X411
X412
X421
X422
X431
X432
X441
X442

X451
X452
X461
X462
X511
X512
X521
X522
X531
X532
X541
X542
X551
X552
X561
X562
X611
X612
X621
X622
X631
X632
X641
X642
X651
X652
X661
X662

END

Ek C: Sonuçların detayları

MPL Modeling System - Copyright (c) 1988-2014, Maximal Software, Inc.

MODEL STATISTICS

Problem name: Zaman_Pencereli_ARP

Filename: v7 gurobi ile çalışan küçük sample - Tezde yer alan.mpl

Date: June 1, 2015

Time: 14:14

Parsing time: 0.13 sec

MPL version: 4.2.14.106

Solver name: Gurobi (5.6.3)

Objective value: 203.000000000

Integer nodes: 38

Improving nodes: 3

Iterations: 1478

Solution status: Infeasible

Solution time: 0.41 sec

Solver result: Optimal solution found

Result code: 2

Constraints: 401

Variables: 132

Integers: 126

Nonzeros: 1208

Density: 2 %

SOLUTION RESULT

Optimal solution found

$$\text{MIN Yol} = 203.0000$$

DECISION VARIABLES

VARIABLE l[j,k] :

j	k	Activity	Reduced Cost
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	13.0000	0.0000
2	2	7.0000	0.0000
3	1	13.0000	0.0000
3	2	9.0000	0.0000
4	1	13.0000	0.0000
4	2	2.0000	0.0000
5	1	13.0000	0.0000
5	2	13.0000	0.0000
6	1	13.0000	0.0000
6	2	11.0000	0.0000

VARIABLE Y[j,k] :

j	k	Activity	Reduced Cost
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	999999998.0000	0.0000
2	2	2.0000	0.0000
3	1	1147483649.0000	0.0000
3	2	999999996.0000	0.0000
4	1	999999998.0000	0.0000
4	2	1.0000	0.0000
5	1	999999998.0000	0.0000
5	2	1000000000.0000	0.0000
6	1	1999999996.0000	0.0000
6	2	999999998.0000	0.0000

VARIABLE V[j,k] :

j	k	Activity	Reduced Cost
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	0.0000	0.0000
2	2	0.0000	0.0000
3	1	0.0000	0.0000

3 2	0.0000	0.0000
4 1	0.0000	0.0000
4 2	0.0000	0.0000
5 1	0.0000	0.0000
5 2	0.0000	0.0000
6 1	0.0000	0.0000
6 2	0.0000	0.0000

VARIABLE S[i,k] :

i k	Activity	Reduced Cost
1 1	7.0000	0.0000
1 2	7.0000	0.0000
2 1	0.0000	0.0000
2 2	14.0000	0.0000
3 1	0.0000	0.0000
3 2	15.0000	0.0000
4 1	0.0000	0.0000
4 2	8.0000	0.0000
5 1	0.0000	0.0000
5 2	20.0000	0.0000
6 1	20.0000	0.0000
6 2	0.0000	0.0000

VARIABLE W[i] :

i	Activity	Reduced Cost
1	0.0000	0.0000
2	0.0000	0.0000
3	0.0000	0.0000
4	0.0000	0.0000
5	0.0000	0.0000
6	0.0000	0.0000

VARIABLE P[j,k] :

j	k	Activity	Reduced Cost
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	23.0000	0.0000
2	2	6.0000	0.0000
3	1	23.0000	0.0000
3	2	15.0000	0.0000
4	1	23.0000	0.0000
4	2	5.0000	0.0000
5	1	23.0000	0.0000
5	2	23.0000	0.0000
6	1	23.0000	0.0000

6 2 23.0000 0.0000

VARIABLE X[i,j,k] :

i j k	Activity	Reduced Cost
1 1 1	0.0000	0.0000
1 1 2	0.0000	0.0000
1 2 1	0.0000	0.0000
1 2 2	0.0000	0.0000
1 3 1	0.0000	0.0000
1 3 2	0.0000	0.0000
1 4 1	0.0000	0.0000
1 4 2	1.0000	0.0000
1 5 1	0.0000	0.0000
1 5 2	0.0000	0.0000
1 6 1	1.0000	0.0000
1 6 2	0.0000	0.0000
2 1 1	0.0000	0.0000
2 1 2	0.0000	0.0000
2 2 1	0.0000	0.0000
2 2 2	0.0000	0.0000
2 3 1	0.0000	0.0000
2 3 2	1.0000	0.0000
2 4 1	0.0000	0.0000
2 4 2	0.0000	0.0000
2 5 1	0.0000	0.0000
2 5 2	0.0000	0.0000

2 6 1	0.0000	0.0000
2 6 2	0.0000	0.0000
3 1 1	0.0000	0.0000
3 1 2	0.0000	0.0000
3 2 1	0.0000	0.0000
3 2 2	0.0000	0.0000
3 3 1	0.0000	0.0000
3 3 2	0.0000	0.0000
3 4 1	0.0000	0.0000
3 4 2	0.0000	0.0000
3 5 1	0.0000	0.0000
3 5 2	1.0000	0.0000
3 6 1	0.0000	0.0000
3 6 2	0.0000	0.0000
4 1 1	0.0000	0.0000
4 1 2	0.0000	0.0000
4 2 1	0.0000	0.0000
4 2 2	1.0000	0.0000
4 3 1	0.0000	0.0000
4 3 2	0.0000	0.0000
4 4 1	0.0000	0.0000
4 4 2	0.0000	0.0000
4 5 1	0.0000	0.0000
4 5 2	0.0000	0.0000
4 6 1	0.0000	0.0000
4 6 2	0.0000	0.0000
5 1 1	0.0000	0.0000
5 1 2	1.0000	0.0000

5 2 1	0.0000	0.0000
5 2 2	0.0000	0.0000
5 3 1	0.0000	0.0000
5 3 2	0.0000	0.0000
5 4 1	0.0000	0.0000
5 4 2	0.0000	0.0000
5 5 1	0.0000	0.0000
5 5 2	0.0000	0.0000
5 6 1	0.0000	0.0000
5 6 2	0.0000	0.0000
6 1 1	1.0000	0.0000
6 1 2	0.0000	0.0000
6 2 1	0.0000	0.0000
6 2 2	0.0000	0.0000
6 3 1	0.0000	0.0000
6 3 2	0.0000	0.0000
6 4 1	0.0000	0.0000
6 4 2	0.0000	0.0000
6 5 1	0.0000	0.0000
6 5 2	0.0000	0.0000
6 6 1	0.0000	0.0000
6 6 2	0.0000	0.0000

CONSTRAINTS

CONSTRAINT K40[j] :

j	Slack	Shadow Price
---	-------	--------------

2	0.0000	0.0000
3	0.0000	0.0000
4	0.0000	0.0000
5	0.0000	0.0000
6	0.0000	0.0000

CONSTRAINT K1[i,j,k] :

i	j	k	Slack	Shadow Price
1	1	1	0.0000	0.0000
1	1	2	0.0000	0.0000
2	2	1	0.0000	0.0000
2	2	2	0.0000	0.0000
3	3	1	0.0000	0.0000
3	3	2	0.0000	0.0000
4	4	1	0.0000	0.0000
4	4	2	0.0000	0.0000
5	5	1	0.0000	0.0000
5	5	2	0.0000	0.0000
6	6	1	0.0000	0.0000
6	6	2	0.0000	0.0000

CONSTRAINT K11[k] :

k	Slack	Shadow Price
1	0.0000	0.0000

2	0.0000	0.0000
---	--------	--------

CONSTRAINT K8[k] :

k	Slack	Shadow Price
---	-------	--------------

1	0.0000	0.0000
---	--------	--------

2	0.0000	0.0000
---	--------	--------

CONSTRAINT K7[k] :

k	Slack	Shadow Price
---	-------	--------------

1	0.0000	0.0000
---	--------	--------

2	0.0000	0.0000
---	--------	--------

CONSTRAINT K6[k] :

k	Slack	Shadow Price
---	-------	--------------

1	0.0000	0.0000
---	--------	--------

2	0.0000	0.0000
---	--------	--------

CONSTRAINT K5[k] :

k	Slack	Shadow Price
1	0.0000	0.0000
2	0.0000	0.0000

CONSTRAINT K12[i=1,k] :

i k	Slack	Shadow Price
1 1	0.0000	0.0000
1 2	0.0000	0.0000

CONSTRAINT K26[j=1,k] :

j k	Slack	Shadow Price
1 1	0.0000	0.0000
1 2	0.0000	0.0000

CONSTRAINT K13[i,j,k] :

i	j	k	Slack	Shadow Price
1	2	1	-1000000010.0000	0.0000
1	2	2	-1000000004.0000	0.0000
1	3	1	-1000000011.0000	0.0000
1	3	2	-1000000007.0000	0.0000
1	4	1	-1000000011.0000	0.0000
1	4	2	0.0000	0.0000
1	5	1	-1000000009.0000	0.0000
1	5	2	-1000000009.0000	0.0000
1	6	1	-5.0000	0.0000
1	6	2	-1000000003.0000	0.0000
2	3	1	-999999998.0000	0.0000
2	3	2	0.0000	0.0000
2	4	1	-999999998.0000	0.0000
2	4	2	-999999993.0000	0.0000
2	5	1	-999999996.0000	0.0000
2	5	2	-1000000002.0000	0.0000
2	6	1	-999999992.0000	0.0000
2	6	2	-999999996.0000	0.0000
3	2	1	-999999997.0000	0.0000
3	2	2	-999999995.0000	0.0000
3	4	1	-999999998.0000	0.0000
3	4	2	-999999991.0000	0.0000
3	5	1	-999999996.0000	0.0000

3 5 2	0.0000	0.0000
3 6 1	-999999992.0000	0.0000
3 6 2	-999999994.0000	0.0000
4 2 1	-999999997.0000	0.0000
4 2 2	-2.0000	0.0000
4 3 1	-999999998.0000	0.0000
4 3 2	-1000000005.0000	0.0000
4 5 1	-999999996.0000	0.0000
4 5 2	-1000000007.0000	0.0000
4 6 1	-999999992.0000	0.0000
4 6 2	-1000000001.0000	0.0000
5 2 1	-999999997.0000	0.0000
5 2 2	-999999991.0000	0.0000
5 3 1	-999999998.0000	0.0000
5 3 2	-999999994.0000	0.0000
5 4 1	-999999998.0000	0.0000
5 4 2	-999999987.0000	0.0000
5 6 1	-999999992.0000	0.0000
5 6 2	-999999990.0000	0.0000
6 2 1	-999999997.0000	0.0000
6 2 2	-999999993.0000	0.0000
6 3 1	-999999998.0000	0.0000
6 3 2	-999999996.0000	0.0000
6 4 1	-999999998.0000	0.0000
6 4 2	-999999989.0000	0.0000
6 5 1	-999999996.0000	0.0000
6 5 2	-999999998.0000	0.0000

CONSTRAINT K41[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000

CONSTRAINT K52[i,j,k] :

i	j	k	Slack	Shadow Price
1	2	1	-1000000022.0000	0.0000
1	2	2	-1000000005.0000	0.0000
1	3	1	-1000000020.0000	0.0000
1	3	2	-1000000012.0000	0.0000
1	4	1	-1000000018.0000	0.0000
1	4	2	0.0000	0.0000
1	5	1	-1000000015.0000	0.0000
1	5	2	-1000000015.0000	0.0000
1	6	1	-11.0000	0.0000
1	6	2	-1000000011.0000	0.0000
2	3	1	-999999997.0000	0.0000

2 3 2	-6.0000	0.0000
2 4 1	-999999995.0000	0.0000
2 4 2	-999999994.0000	0.0000
2 5 1	-999999992.0000	0.0000
2 5 2	-1000000009.0000	0.0000
2 6 1	-999999988.0000	0.0000
2 6 2	-1000000005.0000	0.0000
3 2 1	-999999999.0000	0.0000
3 2 2	-999999990.0000	0.0000
3 4 1	-999999995.0000	0.0000
3 4 2	-999999985.0000	0.0000
3 5 1	-999999992.0000	0.0000
3 5 2	0.0000	0.0000
3 6 1	-999999988.0000	0.0000
3 6 2	-999999996.0000	0.0000
4 2 1	-999999999.0000	0.0000
4 2 2	0.0000	0.0000
4 3 1	-999999997.0000	0.0000
4 3 2	-1000000007.0000	0.0000
4 5 1	-999999992.0000	0.0000
4 5 2	-1000000010.0000	0.0000
4 6 1	-999999988.0000	0.0000
4 6 2	-1000000006.0000	0.0000
5 2 1	-999999999.0000	0.0000
5 2 2	-999999982.0000	0.0000
5 3 1	-999999997.0000	0.0000
5 3 2	-999999989.0000	0.0000
5 4 1	-999999995.0000	0.0000

5	4	2	-999999977.0000	0.0000
5	6	1	-999999988.0000	0.0000
5	6	2	-999999988.0000	0.0000
6	2	1	-999999999.0000	0.0000
6	2	2	-999999982.0000	0.0000
6	3	1	-999999997.0000	0.0000
6	3	2	-999999989.0000	0.0000
6	4	1	-999999995.0000	0.0000
6	4	2	-999999977.0000	0.0000
6	5	1	-999999992.0000	0.0000
6	5	2	-999999992.0000	0.0000

CONSTRAINT K53[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000

CONSTRAINT K14[j,k] :

j	k	Slack	Shadow Price
---	---	-------	--------------

2 1	0.0000	0.0000
2 2	6.0000	0.0000
3 1	0.0000	0.0000
3 2	4.0000	0.0000
4 1	0.0000	0.0000
4 2	11.0000	0.0000
5 1	0.0000	0.0000
5 2	0.0000	0.0000
6 1	0.0000	0.0000
6 2	2.0000	0.0000

CONSTRAINT K53[j,k] :

j k	Slack	Shadow Price
2 1	-0.0000	0.0000
2 2	17.0000	0.0000
3 1	-0.0000	0.0000
3 2	8.0000	0.0000
4 1	-0.0000	0.0000
4 2	18.0000	0.0000
5 1	-0.0000	0.0000
5 2	-0.0000	0.0000
6 1	-0.0000	0.0000

6 2 -0.0000 0.0000

CONSTRAINT K15[i,j,k] :

i	j	k	Slack	Shadow Price
1	2	1	-1999999997.0000	0.0000
1	2	2	-1000000001.0000	0.0000
1	3	1	-2147483648.0000	0.0000
1	3	2	-1999999995.0000	0.0000
1	4	1	-1999999997.0000	0.0000
1	4	2	0.0000	0.0000
1	5	1	-1999999997.0000	0.0000
1	5	2	-1999999999.0000	0.0000
1	6	1	-1999999995.0000	0.0000
1	6	2	-1999999997.0000	0.0000
2	3	1	-1147483650.0000	0.0000
2	3	2	-999999993.0000	0.0000
2	4	1	-999999999.0000	0.0000
2	4	2	-999999998.0000	0.0000
2	5	1	-999999999.0000	0.0000
2	5	2	-1999999997.0000	0.0000
2	6	1	-1999999997.0000	0.0000
2	6	2	-1999999995.0000	0.0000
3	2	1	-852516348.0000	0.0000

3 2 2	-5.0000	0.0000
3 4 1	-852516348.0000	0.0000
3 4 2	-4.0000	0.0000
3 5 1	-852516348.0000	0.0000
3 5 2	-3.0000	0.0000
3 6 1	-1852516346.0000	0.0000
3 6 2	-1000000001.0000	0.0000
4 2 1	-999999999.0000	0.0000
4 2 2	0.0000	0.0000
4 3 1	-1147483650.0000	0.0000
4 3 2	-1999999994.0000	0.0000
4 5 1	-999999999.0000	0.0000
4 5 2	-1999999998.0000	0.0000
4 6 1	-1999999997.0000	0.0000
4 6 2	-1999999996.0000	0.0000
5 2 1	-999999999.0000	0.0000
5 2 2	-1.0000	0.0000
5 3 1	-1147483650.0000	0.0000
5 3 2	-999999995.0000	0.0000
5 4 1	-999999999.0000	0.0000
5 4 2	0.0000	0.0000
5 6 1	-1999999997.0000	0.0000
5 6 2	-999999997.0000	0.0000
6 2 1	-1.0000	0.0000
6 2 2	-3.0000	0.0000
6 3 1	-147483652.0000	0.0000
6 3 2	-999999997.0000	0.0000
6 4 1	-1.0000	0.0000

6	4	2	-2.0000	0.0000
6	5	1	-1.0000	0.0000
6	5	2	-1000000001.0000	0.0000

CONSTRAINT K42[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000

CONSTRAINT K16[i,j,k] :

i	j	k	Slack	Shadow Price
1	2	1	999999992.6000	0.0000
1	2	2	1000000006.6000	0.0000
1	3	1	999999992.6000	0.0000
1	3	2	1000000007.6000	0.0000
1	4	1	999999992.7000	0.0000
1	4	2	0.7000	0.0000
1	5	1	999999992.7000	0.0000

1	5	2	10000000	12.7000	0.0000
1	6	1		12.0000	0.0000
1	6	2	99999999	2.0000	0.0000
2	3	1	99999999	9.4000	0.0000
2	3	2		0.4000	0.0000
2	4	1	99999999	9.3000	0.0000
2	4	2	99999999	3.3000	0.0000
2	5	1	99999999	9.1000	0.0000
2	5	2	10000000	05.1000	0.0000
2	6	1	10000000	18.8000	0.0000
2	6	2	99999999	84.8000	0.0000
3	2	1	99999999	9.4000	0.0000
3	2	2	99999999	8.4000	0.0000
3	4	1	99999999	9.3000	0.0000
3	4	2	99999999	2.3000	0.0000
3	5	1	99999999	9.4000	0.0000
3	5	2		4.4000	0.0000
3	6	1	10000000	18.7000	0.0000
3	6	2	99999999	83.7000	0.0000
4	2	1	99999999	9.3000	0.0000
4	2	2		5.3000	0.0000
4	3	1	99999999	9.3000	0.0000
4	3	2	10000000	06.3000	0.0000
4	5	1	99999999	9.2000	0.0000
4	5	2	10000000	11.2000	0.0000
4	6	1	10000000	18.8000	0.0000
4	6	2	99999999	90.8000	0.0000
5	2	1	99999999	9.1000	0.0000

5	2	2	999999993.1000	0.0000
5	3	1	999999999.4000	0.0000
5	3	2	999999994.4000	0.0000
5	4	1	999999999.2000	0.0000
5	4	2	999999987.2000	0.0000
5	6	1	1000000018.5000	0.0000
5	6	2	999999978.5000	0.0000
6	2	1	999999978.8000	0.0000
6	2	2	1000000012.8000	0.0000
6	3	1	999999978.7000	0.0000
6	3	2	1000000013.7000	0.0000
6	4	1	999999978.8000	0.0000
6	4	2	1000000006.8000	0.0000
6	5	1	999999978.5000	0.0000
6	5	2	1000000018.5000	0.0000

CONSTRAINT K42[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000

CONSTRAINT K17[j,k] :

j	k	Slack	Shadow Price
2	1	0.0000	0.0000
2	2	7.0000	0.0000
3	1	0.0000	0.0000
3	2	7.0000	0.0000
4	1	0.0000	0.0000
4	2	0.0000	0.0000
5	1	0.0000	0.0000
5	2	13.0000	0.0000
6	1	13.0000	0.0000
6	2	0.0000	0.0000

CONSTRAINT K18[j,k] :

j	k	Slack	Shadow Price
2	1	0.0000	0.0000
2	2	0.0000	0.0000
3	1	0.0000	0.0000
3	2	1.0000	0.0000
4	1	0.0000	0.0000

4	2	10.0000	0.0000
5	1	0.0000	0.0000
5	2	0.0000	0.0000
6	1	0.0000	0.0000
6	2	0.0000	0.0000

CONSTRAINT K19[i,j,k] :

i	j	k	Slack	Shadow Price
1	2	1	1000000007.0000	0.0000
1	2	2	1000000007.0000	0.0000
1	3	1	1000000007.0000	0.0000
1	3	2	1000000007.0000	0.0000
1	4	1	1000000007.0000	0.0000
1	4	2	7.3000	0.0000
1	5	1	1000000007.0000	0.0000
1	5	2	1000000007.0000	0.0000
1	6	1	8.0000	0.0000
1	6	2	1000000007.0000	0.0000
2	3	1	1000000000.0000	0.0000
2	3	2	14.6000	0.0000
2	4	1	1000000000.0000	0.0000
2	4	2	1000000014.0000	0.0000
2	5	1	1000000000.0000	0.0000

2 5 2	1000000014.0000	0.0000
2 6 1	1000000000.0000	0.0000
2 6 2	1000000014.0000	0.0000
3 2 1	1000000000.0000	0.0000
3 2 2	1000000015.0000	0.0000
3 4 1	1000000000.0000	0.0000
3 4 2	1000000015.0000	0.0000
3 5 1	1000000000.0000	0.0000
3 5 2	15.6000	0.0000
3 6 1	1000000000.0000	0.0000
3 6 2	1000000015.0000	0.0000
4 2 1	1000000000.0000	0.0000
4 2 2	8.7000	0.0000
4 3 1	1000000000.0000	0.0000
4 3 2	1000000008.0000	0.0000
4 5 1	1000000000.0000	0.0000
4 5 2	1000000008.0000	0.0000
4 6 1	1000000000.0000	0.0000
4 6 2	1000000008.0000	0.0000
5 2 1	1000000000.0000	0.0000
5 2 2	1000000020.0000	0.0000
5 3 1	1000000000.0000	0.0000
5 3 2	1000000020.0000	0.0000
5 4 1	1000000000.0000	0.0000
5 4 2	1000000020.0000	0.0000
5 6 1	1000000000.0000	0.0000
5 6 2	1000000020.0000	0.0000
6 2 1	1000000020.0000	0.0000

6	2	2	1000000000.0000	0.0000
6	3	1	1000000020.0000	0.0000
6	3	2	1000000000.0000	0.0000
6	4	1	1000000020.0000	0.0000
6	4	2	1000000000.0000	0.0000
6	5	1	1000000020.0000	0.0000
6	5	2	1000000000.0000	0.0000

CONSTRAINT K24[i,k] :

i	k	Slack	Shadow Price
1	1	7.0000	0.0000
1	2	7.0000	0.0000
2	1	0.0000	0.0000
2	2	14.0000	0.0000
3	1	0.0000	0.0000
3	2	15.0000	0.0000
4	1	0.0000	0.0000
4	2	8.0000	0.0000
5	1	0.0000	0.0000
5	2	20.0000	0.0000
6	1	20.0000	0.0000
6	2	0.0000	0.0000

CONSTRAINT K25[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	-999999998.0000	0.0000
2	2	-2.0000	0.0000
3	1	-1147483649.0000	0.0000
3	2	-999999996.0000	0.0000
4	1	-999999998.0000	0.0000
4	2	-1.0000	0.0000
5	1	-999999998.0000	0.0000
5	2	-1000000000.0000	0.0000
6	1	-1999999996.0000	0.0000
6	2	-999999998.0000	0.0000

CONSTRAINT K45[j,k] :

j	k	Slack	Shadow Price
1	1	-7.0000	0.0000

1	2	-7.0000	0.0000
2	1	0.0000	0.0000
2	2	-14.0000	0.0000
3	1	0.0000	0.0000
3	2	-15.0000	0.0000
4	1	0.0000	0.0000
4	2	-8.0000	0.0000
5	1	0.0000	0.0000
5	2	-20.0000	0.0000
6	1	-20.0000	0.0000
6	2	0.0000	0.0000

CONSTRAINT K46[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	-13.0000	0.0000
2	2	-7.0000	0.0000
3	1	-13.0000	0.0000
3	2	-9.0000	0.0000
4	1	-13.0000	0.0000
4	2	-2.0000	0.0000
5	1	-13.0000	0.0000

5	2	-13.0000	0.0000
6	1	-13.0000	0.0000
6	2	-11.0000	0.0000

CONSTRAINT K47[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	0.0000	0.0000
2	2	0.0000	0.0000
3	1	0.0000	0.0000
3	2	0.0000	0.0000
4	1	0.0000	0.0000
4	2	0.0000	0.0000
5	1	0.0000	0.0000
5	2	0.0000	0.0000
6	1	0.0000	0.0000
6	2	0.0000	0.0000

CONSTRAINT K51[j,k] :

j	k	Slack	Shadow Price
1	1	0.0000	0.0000
1	2	0.0000	0.0000
2	1	-23.0000	0.0000
2	2	-6.0000	0.0000
3	1	-23.0000	0.0000
3	2	-15.0000	0.0000
4	1	-23.0000	0.0000
4	2	-5.0000	0.0000
5	1	-23.0000	0.0000
5	2	-23.0000	0.0000
6	1	-23.0000	0.0000
6	2	-23.0000	0.0000

END

ÖZGEÇMİŞ

Adı Soyadı : Kuntay AKCA
Doğum Yeri ve Tarihi : Beyşehir / 23.07.1980
Yabancı Dili : Fransızca ve İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise : İzmir Saint Joseph Fransız Lisesi (1991-1998)

Lisans : Ege Üniversitesi
İngilizce Hazırlık (1999-2000)
İşletme Bölümü (2000-2004)

Yüksek Lisans : Uludağ Üniversitesi
Bilimsel Hazırlık (2012-2013)
Endüstri Mühendisliği (2013-2015)

Çalıştığı Kurum/Kurumlar ve Yıl : Oyak-Renault Otomobil Fabrikaları A.Ş. (2006-..)
İletişim (e-posta) : kuntay.akca@renault.com
akcakuntay@hotmail.com