



TÜBİTAK

PERİYODİK ARAÇ ROTALAMA PROBLEMİNİN
ÜÇ FARKLI TÜREVI ÜZERİNE

PROGRAM KODU : 1001
PROJE NO : 214M195
PROJE YÜRÜTÜCÜSÜ : Yrd. Doç Dr. Ahmet CAMCI

Araştırmacı:

Doç. Dr. Özgür ÖZPEYNİRCİ

Bursiyerler:

Okan ALTINKÖK

Onur ÇAĞIRICI

Cansu KÜRKÜTLÜ

Arya SEVGİN

Temmuz 2017

İZMİR

Önsöz

Bu proje gerçek pek çok farklı firmanın lojistik operasyonlarından esinlenilerek tasarlanmış üç farklı problem üzerine gerçekleştirilmiştir. Projede ele alınan problemler için sezgisel çözüm yöntemleri geliştirilmiş ve bu sezgisel yöntemlerin performansları sayısal deneylerle test edilmiştir. Proje TÜBİTAK 1001 Bilimsel ve Teknolojik Araştırma Projelerini Destekleme Programı kapsamında desteklenmiştir.

İçindekiler

Önsöz	i
Proje Özeti	vii
Abstract	ix
1 Giriş	1
2 Literatür Taraması	3
3 Problem Tanımı	8
3.1 Araç Rotalama Problemi	8
3.2 Periyodik Araç Rotalama Problemi	12
4 Periyodik Araç Rotalama Problemi - 2 Ziyaret Tipi	16
4.1 Problem tanımı	16
4.1.1 Hesaplama Karmaşıklığı	20
4.2 Alt Sınır Algoritmaları	22
4.2.1 Kurucu Alt Sınır Algoritması	22
4.2.2 Dal-Kesi Algoritması	24
4.3 Sezgisel Çözüm Yöntemleri	29
4.3.1 Başlangıç Çözümü Elde Etme	32

4.3.2	Komşuluk Arama	33
4.3.3	Yerel Arama	38
4.3.4	Benzetimli Tavlama	38
4.4	Deney Sonuçları	39
4.4.1	Örnek Problem Oluşturma	39
4.4.2	Kurucu Alt Sınır Sonuçları	40
4.4.3	Sayısal Sonuçları	43
5	Periyodik Araç Rotalama Problemi - Ziyaret Sıklığı	46
5.1	Problem Tanımı	46
5.1.1	Hesaplama Karmaşıklığı	50
5.2	Üst Sınır	50
5.2.1	Örnek Problemlere Eklemeler	51
5.2.2	Dal-Kesi Algoritması	52
5.3	Sezgisel Yöntem	55
5.4	Deney Sonuçları	58
6	Periyodik Araç Rotalama Problemi - Birleşik	60
6.1	Problem Tanımı	60
6.1.1	Hesaplama Karmaşıklığı	62
6.2	Sezgisel Yöntem	62
6.3	Deney Sonuçları	63
7	Sonuç	65

Tablo Listesi

4.1	Küçük problemler - 5 müşterili	41
4.2	Küçük Problemler - 20 müşterili	41
4.3	Örnek Problem Bilgileri ve Alt Sınır Performansları	42
4.4	Eşitsizliklerin incelenmesi	44
4.5	PARP-2ZT Alt ve Üst Sınır Algoritmaları	45
5.1	Alt Sınır Algoritmaları Sayısal Sonuçları	59
6.1	PARP-B Deney Sonuçları	64

Şekil Listesi

4.1	Swap Operatörü	35
4.2	Move Operatörü	37
5.1	Ziyaret sıklığının talebe getirisi	53

Algoritmalar Listesi

1	Alt Sınır Algoritması	23
2	Dal Kesi Algoritması	26
3	DKA Algoritması	31
4	Başlangıç Çözümü Elde Etme	33
5	Swap Operatörü	34
6	Move operatörü	36
7	Dal Kesi Algoritması	54

Proje Özeti

Araç rotalama problemi güncel bir problem olup ellili yıllardan beri çalışılmaktadır. Önceleri araç rotalama problemi karmaşık oluşu ve teknolojinin yeterli olmaması sebebiyle literatürde yaygın olarak ele alınmamıştır. Doksanlı yıllarda gelişen teknoloji ile araç rotalama problemleri çalışılabilir hale gelmiş ve araç rotalama problemi üzerine yapılan çalışmaların sayısı artmıştır Araç rotalama problemi bir veya daha fazla depodan çıkan araçların tüm müşterileri ziyaret ederek tekrar depoya dönmesi üzerine kurulu bir problemdir. Araç rotalama problemi kat edilen mesafeyi en küçükleyerek ulaşım giderlerini ve harcanan zamanı azalttığı gibi aynı zamanda araç sayısını en aza indirerek maliyetlerin düşmesine yardımcı olur. Buna ek olarak karbon salınımını da düşürerek çevreye verilen zararı azaltır. Son yıllarda tedarik zinciri ve dağıtım kanallarının rekabetçi piyasadaki önemi oldukça artmıştır. Dolayısıyla firmaların araç rotalama problemine olan ilgisi de artmıştır.

Araç rotalama probleminin gerçek hayat ihtiyaçları üzerine birçok türevi oluşmuştur. Bunun sebebi farklı tipteki şirketlerin işleyişlerinin değişkenlik göstermesidir. Örneğin müşterilerin sadece belirli saat aralıklarında ziyaret edilebilmesi, kargo şirketinin aynı anda hem dağıtım hem de toplama işlemi yapması, bazı müşterilerin aynı araçla ve belirli bir sıra ile ziyaret edilme zorunluluğu olması gibi sebeplerden dolayı problemin birçok türevi literatürde tanımlanmış ve çalışılmıştır.

Araç rotalama probleminin literatürde sıklıkla çalışılan türevlerinden birisi periyodik araç rotalama problemidir. Bu problem belirlenen bir periyot içerisinde müşterilerin bir veya daha fazla ziyaret edilmesi üzerine kuruludur. Her periyotta ziyaret edilecek müşterilere ve her periyot için araç rotalarına karar verilmektedir. Amaç genellikle toplam rota

uzunluklarını en aza indirmektir. Ziyaret günlerinin uygun şekilde düzenlenmesi maliyetler ve araç sayısı açısından büyük avantaj sağlar.

Bu projede periyodik araç rotalama problemi üzerine gerçek hayattan esinlenen üç yeni problem tanımlanmıştır ve bu problemler için matematiksel modeller, alt ve üst sınırlar ile sezgisel çözüm yöntemleri geliştirilmiştir. Geliştirilen yöntemlerin sonuç kalitesi ve süre performansları rassal olarak oluşturulan örnekler üzerinde test edilmiştir.

İlk problem bir müşteriye yapılan her ziyareti takip eden günde farklı amaçlı bir ziyaret yapılmasını gerektirmektedir. İlk ziyaret talep toplama, ikinci ziyaret ise ürünlerin teslimi olarak düşünülebilir. İlk tip ziyaret küçük, hızlı ve zaman kapasiteli araçlarla, ikinci tip ziyaret ise görece büyük, yavaş ve fiziksel kapasiteye sahip araçlarla yapılmaktadır. Bu sebeple birbirini takip eden günlerde aynı rotaların kullanılması mümkün olamamaktadır.

Gerçek hayatta müşterilerin daha sık ziyaret edilmesi satışları artırmakta ancak toplam rota uzunluğunun da artmasına sebep olmaktadır. Proje kapsamında çalışılan ikinci problemde, periyodik araç rotalama problemi kararlarına ek olarak müşterilerin ziyaret sıklığının müşteri talebini değiştirmesi durumu incelenmiştir.

Üçüncü problem ilk iki problemin birleşmesi ile elde edilmiştir. Bu problemde iki tip araç vardır ve müşterilerin ziyaret sıklıkları değiştirilebilmektedir.

Proje kapsamında ele alınan problemler, detaylı olarak tanımlanmış, matematik modeli geliştirilmiş ve çözüm karmaşıklığı incelenmiştir. Problemler için alt ve üst sınır algoritmaları ve olurlu çözüm bulan sezgisel yöntemler geliştirilmiştir. Geliştirilen alt ve üst sınır algoritmaları ile sezgisel yöntemlerin performansları rassal örnekler üzerinde test edilmiştir.

Anahtar Kelimeler: Araç rotalama problemi, periyodik araç rotalama problemi, sezgisel yöntemler, matematik programlama

Abstract

Vehicle routing problem is a contemporary topic and it has been studied since 1950s. In early times, vehicle routing problem has not been studied widely because it's a complex problem and at that time the level of technology was not sufficient enough. In 1990s, with developing technology vehicle routing problem became available to study and number of studies on vehicle routing problem began to increase. Vehicle routing problem is the problem in which vehicles departure from one or more depots, visit customers and finally return to the depot(s). Generally, vehicle routing problem aims to minimize the distance of the vehicles' routes and decrease transportation costs. However, the vehicle routing problem may aim to minimize number of vehicles. Moreover, vehicle routing problem protects the environment by decreasing the carbon emission. In last decades, the importance of supply chain management and distribution channels has increased for today's competitive environment. In consequence, firms started to give more importance to vehicle routing problems.

There are many variants of vehicle routing problem derived in order to cope with the requirements of the real life applications. The reason behind this is the operational differences of the companies. The necessity of some customers to be visited in certain time windows, the joint pickup and delivery operation of cargo firms, the requirement of visiting some customer with the same vehicle and with predefined order precedence relations between customers are some of the reasons for defining and studying different variants of the problem.

Periodic vehicle routing problem is a derivative of vehicle routing problem which is well studied in the literature. This problem is based on visiting customers once or more in a given

period. The problem decides the customers to be visited and the routes for each period. In general, the goal is to minimize the total length of the routes. Organizing the visit days properly provides an important advantage to decreasing costs and the number of vehicles.

In this project, two new vehicle routing problems are defined based on real life industrial operations. Mathematical models, lower and upper bounds and heuristic methods for these problems are developed. Solution quality and time performance of the developed methods are tested on randomly generated instances.

The first problem requires a different aimed customer visit in the following day of a regular customer visit day. One can consider the first customer visit for collecting demand information and the second customer visit for delivering goods. The first type visit is realized with small scale, fast and time capacitated vehicles, whereas the second type visit is realized with relatively large scale, slow and physical capacitated vehicles. Hence using the same routes for the consecutive days may not be possible.

It can be seen from real life examples that as the number of visits to a customer increases the demand increases but the total distance traveled in a period also increases. In the second problem it is assumed that the demand is dependent on the frequency of the visits in a periodic vehicle routing problem setting.

The third problem is a combination of the first and the second problems. In this problem, there are two types of visits and the frequency of the visits can be changed.

Each problem is defined in detail, mathematical models for the problems are developed and the computational complexity of the problems is analyzed. Lower and upper bound algorithms and heuristic methods for obtaining feasible solutions are developed for each problem. The performances of the lower and upper bound algorithms and heuristic methods are tested with the randomly generated instances.

Keywords: Vehicle routing problem, periodic vehicle routing problem, heuristic methods, mathematical programming

Bölüm 1

Giriş

Araç Rotalama Problemi (ARP) bir veya daha fazla depodan çıkan araçların tüm müşterileri ziyaret ederek tekrar depoya dönmesi üzerine kurulu bir problemdir. ARP kat edilen mesafeyi en küçükleyerek ulaşım giderlerini ve harcanan zamanı azaltır. Aynı zamanda araç sayısını en aza indirerek maliyetlerin düşmesine yardımcı olur.

ARP'nin literatürde sıklıkla çalışılan türevlerinden birisi Periyodik Araç Rotalama Problemi(PARP)'dir. Bu problem belirlenen bir periyot içerisinde müşterilerin bir veya daha fazla ziyaret edilmesi üzerine kuruludur. Her periyotta ziyaret edilecek müşterilere ve her periyot için araç rotalarına karar verilmektedir. Amaç genellikle toplam rota uzunluklarını en aza indirmektir. Ziyaret günlerinin uygun şekilde düzenlenmesi maliyetler ve araç sayısı açısından avantaj sağlar.

Projenin birinci iş paketi olan Periyodik Araç Rotalama Problemi - 2 Ziyaret Tipi (PARP-2ZT), iki tip aracın bulunduğu bir PARP türevidir. Önerilen problem bir müşteriye yapılan her ziyareti takip eden günde farklı amaçlı bir ziyaret yapılmasını gerektirmektedir. İlk ziyaret talep toplama, ikinci ziyaret ise ürünlerin teslimi olarak düşünülebilir. İlk tip ziyaret küçük, hızlı ve zaman kapasiteli araçlarla, ikinci tip ziyaret ise görece büyük, yavaş ve fiziksel kapasiteye sahip araçlarla yapılmaktadır.

Projenin ikinci iş paketi olan Periyodik Araç Rotalama Problemi - Ziyaret Sıklığı (PARP-

ZS), PARP kararlarına ek olarak müşterilerin ziyaret sıklığını müşteri talebini deęiřtirmesi durumu incelenmektedir. PARP-ZS ziyaret sıklığını artırmannın müşteri talebini artırdığı gerçeğini temel alan bir problemdir. Önerilen problemde kapasite kısıtı olan tek tip araç bulunmakta ve taleplerin telefonla toplandığı varsayılmaktadır.

Projenin üçüncü iş paketi olan Periyodik Araç Rotalama Problemi - Birleşik (PARP-B), ilk iki iş paketinde tanımlanan problemlerin (PARP-2ZT ve PARP-ZS) birleşimidir. Bu problem hem iki tip ziyaret içermektedir hem de ziyaret sıklığı deęiřtirilebilmektedir.

Proje ekibi ve çalışma süreleri řu şekilde olmuřtur:

- Ahmet Camcı (Yürütücü, 06/09/2014-15/09/2016)
- Özgür Özpeynirci (Arařtırmacı, 06/09/2014-15/09/2016)
- Okan Altınkök (Bursiyer, 15/03/2014-15/09/2015)
- Cansu Kürkütü (Bursiyer, 15/09/2014-15/10/2015)
- Arya Sevgen (Bursiyer, 01/10/2015-15/09/2016)
- Onur Çaęırıcı (Bursiyer, 01/10/2015-15/09/2016)

Sonuç raporu řu şekilde planlanmıřtır. Bölüm 2 literatür taraması sunmaktadır. Araç rotalama ve periyodik araç rotalama problemleri Bölüm 3'de tanımlanmıřtır. Bölüm 4 PARP-2ZT problemi kapsamında yapılan çalışmalarını açıklamıřtır. PARP-ZS problemi için yapılan çalışmalar Bölüm 5 altında sunulmuřtur. Bölüm 6 PARP-B problemi ile ilgilidir. Bölüm 7 ile rapor sonuçlandırılmıřtır.

Bölüm 2

Literatür Taraması

Araç rotalama probleminden (ARP) literatürde ilk kez Dantzig vd. (1954) bahsetmiştir. Birden fazla araç içeren bir probleme ilk çözümü Clarke ve Wright (1964) sunmuştur. Literatürde çokça çalışılmış olan gezgin satıcı problemi, araç rotalama probleminin bir özel halidir. Laporte (2009) gezgin satıcı probleminin 50 yıllık geçmişini ve son dönemde gerçekleşen gelişmeleri özetlemiştir.

Araç rotalama probleminin gerçek hayat ihtiyaçları doğrultusunda birçok türevi oluşmuştur. Bunlardan bazıları; müşteri ziyaretlerinin belirli zaman penceresi içerisinde yapılması (Solomon, 1987), birden fazla araç tipinin olması (Gendreau vd., 1999), müşterilerin yükleme ve boşaltma tipi taleplerinin olması (Toth ve Vigo, 1999), müşterilerin ziyaretlerinde öncelik kısıtlarının olması (Dumas vd., 1991), birden fazla deponun olması (Vidal vd., 2012), birden fazla periyot için planlama yapılması (Francis ve Smilowitz, 2006), talep bilgisinin dinamik olması (Wen vd., 2010) şeklinde sıralanabilir. Problemin amacı toplam mesafeyi azaltmak yerine kullanılan araç sayısını azaltmak olarak da ele alınmıştır. Eksioğlu vd. (2009) araç rotalama problemleri literatürü için bir sınıflandırma yöntemi önermişlerdir.

Periyodik araç rotalama problemi (PARP) ilk olarak Beltrami ve Bodin (1974) tarafından tanımlanmıştır. Ayrıca literatürde sıklıkla bahsedilen Russell ve Igo (1979) ve Christofides ve Beasley (1984) çalışmaları da önemli ilk çalışmalar arasında yer almaktadır.

Campbell ve Wilson (2014) PARP ile ilgili bir literatür taraması yapmış ve 40 yılı aşkın süredir bu problem üzerine yapılan çalışmaları derlemiştir. Literatür taraması kapsamında PARP için geliştirilen kesin ve sezgisel yöntemleri incelemiş ve sıkça kullanılan örnek problemlerin bilinen en iyi sonuçlarının zaman içerisinde nasıl geliştiğini raporlanmıştır. Ayrıca PARP için önerilen türevler ve gelecek çalışma alanlarına dair açıklamalar yapmışlardır.

Çöp toplama ve geri dönüşüm (Coene vd., 2010), gıda ve içecek dağıtımı (Golden ve Wasil, 1987) ve yedek parça dağıtımı (Alegre vd., 2007) gibi çeşitli gerçek hayat problemleri periyodik araç rotalama problemi olarak modellenmiştir. Mourgaya ve Vanderbeck (2006) gerçek hayata daha uygun sonuçlar elde edebilmek için sürücülerin tecrübe sahibi olduğu rotalara atanması ve müşterilerin bölgeselleştirilmesi odaklı bir yöntem geliştirmişlerdir. Bu yöntemde bölgeselleşme ve iş yükü dağılımı değerlerinde büyük iyileştirmeler gözlemlenirken, toplam mesafenin uzadığı görülmüştür. Baldacci vd. (2011) bu problem için dal-fiyat tabanlı bir kesin çözüm yöntemi önermiştir ve Cordeau vd. (1997) tarafından derlenmiş olan 32 örneğin 5 tanesinde bilinen en iyi sonuçları geliştirmiş. Ayrıca, 14 tanesinde ise optimal sonuçlara ulaşmıştır.

Lysgaard vd. (2004) araç rotalama problemi için bir dal-kesi yöntemi önermiştir. Çok farklı geçerli eşitsizliğin dikkate alındığı bir algoritma geliştirmişler ve bu algoritma ile üç örneğin bilinen en iyi sonucunu geliştirmeyi başarmışlardır. Bildiğimiz kadarı ile dal-kesi algoritması periyodik araç rotalama problemi üzerinde kullanılmamıştır.

Cordeau vd. (1997) araç rotalama probleminin üç farklı türevi (PARP, Periyodik Gezin Satıcı Problemi ve Çok Depolu Araç Rotalama Problemi) için tabu arama yöntemi geliştirmişlerdir. Literatürde bulunan örnekleri derlemişler ve 32 örnek problem üzerinde geliştirmiş oldukları tabu arama yöntemini test etmişlerdir. Yapılan testler sonucunda 32 örnek problemden 24 tanesinde bilinen en iyi sonuçlardan daha iyi sonuçlar elde etmişlerdir.

Değişken komşuluk arama (DKA) yöntemi literatürde daha önce araç rotalama probleminin çeşitli türevleri için uygulanmıştır. Ancak bilindiği kadarıyla literatürde periyodik araç rotalama problemi için ilk defa Hemmelmayr vd. (2009) tarafından uygulanmıştır.

Hemmelmayr vd. (2009) periyodik araç rotalama problemi için deęişken komşuluk arama ve benzetim tavlama ileri-sezgisel yöntemlerinin karması olan bir çözüm yöntemi önermiş ve yöntemlerini literatürde yer alan başka çalışmalar ile karşılaştırarak test etmişlerdir. Geliştirmiş oldukları sezgisel yöntem, yapılan testler sonucunda büyük problemlerde ve ziyaret sıklığı sayısının yüksek olduğu problemlerde yüksek performans göstermiştir. Geliştirmiş oldukları sezgisel yöntem Cordeau vd. (1997)'de derlemiş olan 42 örnek problemin 24'ünde bilinen en iyi sonuçlardan daha iyi sonuç vermiştir. Aynı zamanda büyük problemler için Cordeau vd. (1997) tarafından önerilmiş olan yönteme kıyasla daha hızlı sonuç almaktadır.

Periyodik araç rotalama problemleri çözümü için önerilen bir başka sezgisel yöntem ise Mourgaya ve Vanderbeck (2006) tarafından geliştirilen sütun türetimi bazlı bir sezgisel yöntemdir. Çözüm yöntemlerinin ilk adımı, aynı araç tarafından ziyaret edilecek müşteriler arasındaki Öklid mesafeyi en azlarken araçlar arasındaki iş yükünü dengeleyen bir taktiksel planlama modelidir. Periyodik Araç Rotalama Problemi örnek problemleri arasından Cordeau vd. (1997) çalışmasında sunulan p01-02-03-17-25-26 üzerinde yaptıkları testler sonucunda literatürde bilinen en iyi sonuçlar ile karşılaştırıldığında toplam maliyet ortalama %20 artarken, sonuçlar bölgelere ayırma kriteri açısından %16, iş yükü dengesi kriteri açısından %70 iyileştirilmiştir.

Cacchiani vd. (2014) tamsayı karışık doğrusal programlama problemleri için geliştirdikleri hibrid en iyileme algoritmasını PARP ile geçerliliğini kanıtlamışlardır. Çalışmada LP gevşetme ve kolon üretimi teknikleri kullanılmıştır. Elde edilen sonuçlarda 32 örnekte bilinen en iyi sonuçlar için 2 adet iyileştirme görülmüştür.

Periyodik Araç Rotalama Probleminde müşteri ziyaret sıklığı bir parametredir. Ancak bu parametreyi bir karar deęişkeni olarak tanımlamak mümkündür. Francis vd. (2006) tarafından ele alınan problem kütüphaneler arası kitap siparişleri ile ilgilidir ve periyodik araç rotalama problemine ek olarak ziyaret sıklığına da karar vermektedir. Çözdükleri kütüphane sistemi örneğinde toplama ve dağıtım bir aradadır. Problemde 50 müşteri 4 araç ile kitap vb. ürünlerin toplama ve teslimatı için ziyaret edilmektedir. Diğer kütüphaneler tarafından

talep edilen ürünler toplanıp, yapılacak bir sonraki ziyarette teslim edilmek üzere depoda sıralanmaktadır. Bu kütüphaneler arası sistem müşterilerine mümkün olan en yüksek ziyaret sıklığını sağlamaya çalışırken bütçe kısıtı nedeniyle en iyi ziyaret sayısını bulmayı hedeflemektedir. Problemden her bir müşteri düğümü için minimum ziyaret sıklığı tanımlanmış ve bu seviyenin üzerinde yapılan ziyaretler için amaç fonksiyonunda bir servis seviyesi getirisi olduğu varsayılmıştır. Model ziyaret sıklığı ve günlük araç rotalarına karar verirken, toplam kat edilen mesafe, ziyaret sebebi ile oluşan ek bekleme süreleri ve servis seviyesi arasında bir denge kurmaktadır. Francis vd. (2006) çözüm yöntemlerinin ilk adımında müşterilerin ziyaret programını belirleyen kararlardan, rotaları oluşturanları ayırmışlar ve 2 alt problem oluşturmuşlardır. İlk alt problemde her müşteriye en uygun ziyaret programı ve araç atanmıştır. İkinci alt problemde ise rotalama kararları her araç-ziyaret programı ikilisi için verilmelidir. Ortaya çıkan problemin NP-Zor olduğu kanıtlanmış Kar Getiren Gezgin Satıcı Problemi'ne benzer olduğu için alt ve üst sınırlar geliştirilmiştir. İlk alt problem için en iyi sonuç elde edilmiştir fakat bu ikinci alt problem için alt tur engelleme kısıtlarının fazlalığından dolayı mümkün olmamıştır. Bu kısıtların sadece bazıları gerekli olduğu için hepsi gevşetilmiş, elde edilen sonuçta bir alt tur varsa gerekli olan kısıtlar eklenip olurlu bir sonuç elde edilene ya da durdurma kuralı devreye girene kadar tekrar çözülmüştür. Elde edilen sonucu rotalama alt problemi için alt sınır olarak kabul etmişler, ziyaret programı atama alt problemi optimal sonucu ile toplayarak amaç fonksiyonu için bir alt sınır elde etmişlerdir. Üst sınır ise ARP için tasarruf sezgisel yöntemi uygulayarak elde edilmiştir. Alt ve üst sınırlar eşit değil ise dal ve sınır algoritması uygulayarak, aralığın kapanmasını sağlamışlardır. Çözüm yöntemlerinin performansını ölçmek için standart PARP ile ziyaret sıklığının karar değişkeni olduğu modelleri karşılaştırılmış rotalama maliyetindeki %32 artışa rağmen daha sık gerçekleştirilen ziyaretlerin servis getirisi ile toplam amaç fonksiyonunda %55 gelişme elde etmişlerdir. Francis vd. (2006) tarafından geliştirilen model, talep büyüklüğü ziyaret sıklığı ilişkisi yönünden bu projede ele alınacak problem ile farklılık göstermektedir.

Literatürdeki bir diğer örnek ise Francis ve Smilowitz (2006) çalışmasıdır. Motivasyon

problemleri çöp toplama, asansör bakımı ve tamiri, kurye servisleri gibi talep bilgisinin dinamik olduğu problemlerdir. Ziyaret sıklığının karar değişkeni olduğu bir sürekli yaklaşım modeli geliştirilmiş, kesikli değişken ve parametrelere sürekli fonksiyonlar ile yaklaşmıştır. Bu model hızlı çözülebilir ve büyük problemler için dağıtım servis sağlayıcılarına servis seçenekleri tasarımı kararlarında yol göstericidir.

Ziyaret sıklığının karar değişkeni olması durumunda problem literatürde bulunan Envanter Rotalama Problemi'ne benzemektedir. Envanter Rotalama Problemi'nde ziyaret sıklığı ve rotaların yanı sıra ziyaret edilen müşteri noktasına teslim edilecek ürün miktarına da karar verilir. Envanter Rotalama Problemi, müşteriye her ziyarette farklı miktarda ürün verilmesi yönünden bizim problemimizle farklılık göstermektedir. Örneğin, ele aldığımız Coca-Cola İçecek firması için müşterilere her ziyarette eşit miktarda ürün teslim edilmesi müşteri tarafından tercih edilen bir durumdur. Literatürde bu alanda yapılan güncel çalışmalardan biri Aksen vd. (2014) tarafından tanımlanan, atık bitkisel yağların biyodizel üretiminde kullanılmak üzere toplanmasını modelleyen seçici ve devirli bir envanter rotalama problemidir. Aksen vd. (2014) tesisin hammadde ihtiyacını karşılarken, toplam toplama, envanter tutma ve satın alma maliyetlerini en azlamayı hedeflemektedirler. Geliştirdikleri uyarlanabilir geniş komşuluk arama yöntemi ile elde ettikleri sonuçların çıktılarında biri de restoran, otel vb. gibi atık bitkisel yağların biriktiği müşteri noktalarından hangilerinin ziyaret programlarına dahil edileceği ve ne kadar yağ toplanacağıdır.

Bölüm 3

Problem Tanımı

Bu bölümde araç rotalama problemi ve periyodik araç rotalama problemleri tanımlanacaktır. Problemler için gerekli notasyon sunulacak ve her iki problemin de matematiksel programlama modelleri açıklanacaktır.

3.1 Araç Rotalama Problemi

Araç rotalama problemleri (ARP) genel olarak bir veya daha fazla depodan ayrılan araçların önceden belirlenmiş noktaları kat edilen mesafeyi en aza indirecek şekilde ziyaret etmesi üzerine kurulu problemlerdir.

Araç rotalama probleminin matematiksel programlama modeli şu şekilde sunulabilir:

Kümeler

N : müşteri düğümleri kümesi, $(i, j \in N)$

N_0 : depo dahil düğümler kümesi, $N_0 = N \cup \{0\}$

Parametreler

c_{ij} : i müşterisinden j müşterisine gitme maliyeti

d_i : i müşterisinin talebi

q : aracın fiziksel kapasitesi

m : araç sayısı

Karar Değişkenleri

x_{ij} : İkili karar değişkenidir, eğer i müşterisinden j müşterisine gidildiyse 1 değerini, diğer durumda ise 0 değerini almaktadır.

u_i : i müşterisi ziyaret edildikten sonra araçtaki toplam yük miktarı

ARP(1)

$$\min \sum_{i \in N_0} \sum_{j \in N_0} c_{ij} x_{ij} \quad (3.1)$$

Öyle ki;

$$\sum_{i \in N_0} x_{ij} = 1 \quad j \in N \quad (3.2)$$

$$\sum_{j \in N_0} x_{ij} = 1 \quad i \in N \quad (3.3)$$

$$\sum_{i \in N_0} x_{i0} \leq m \quad (3.4)$$

$$\sum_{j \in N_0} x_{0j} \leq m \quad (3.5)$$

$$u_i - u_j + x_{ij}q \leq q - d_j \quad i \in N, j \in N \quad (3.6)$$

$$u_i \leq q \quad i \in N_0 \quad (3.7)$$

$$u_i \geq d_i \quad i \in N_0 \quad (3.8)$$

$$x_{ij} \in \{0, 1\} \quad i, j \in N_0 \quad (3.9)$$

Amaç fonksiyonu (3.1) kat edilen mesafeyi en aza indirmeyi hedefler. Kısıt kümeleri (3.2) ve (3.3) her müşteriye tek bir aracın gelip tek bir aracın ayrılmasını garanti altına almaktadır. Kısıt (3.4) ve (3.5) ise depodan ayrılan araç sayısının m kadar ve ya m den az olması gerektiğini belirtmektedir. Kısıt kümeleri (3.6), (3.7), ve (3.8) araçların alt tur (sub-tour) yapmasını, yani belirli müşteriler arasında bir döngü içerisine girmesini engellemekte,

ayrıca araca atanan yüklerin araç kapasitesini aşmamasını sağlamaktadır.

Proje kapsamında, literatürde de sıkça kabul edildiği üzere, müşteri düğümleri arasında mesafelerin simetrik olduğu varsayılmıştır, $c_{ij} = c_{ji} \forall i, j \in N_0$. Bu varsayım altında problem şu şekilde modellenebilir.

Kümeler

N : müşteri düğümleri kümesi, $(i, j \in N)$ ve $i = 1, \dots, n$ öyle ki $|N| = n$

N_0 : depo dahil düğümler kümesi, $N_0 = N \cup \{0\}$

E : ayrıtlar kümesi, $e = (i, j) \in E$ ve $i, j \in N_0$

Parametreler

c_e : e ayrıtlarını kullanma maliyeti

d_i : i müşterisinin talebi

q : aracın fiziksel kapasitesi

Karar Değişkenleri

x_e : e ayrıtlarının kaç defa kullanıldığını gösteren tam sayılı karar değişkenidir.

Verilen bir müşteri alt kümesi, $S \subseteq N$ için $q(S)$ fonksiyonu $\sum_{i \in S} d_i$ değerini, $\delta(S)$ bağlı olduğu iki düğümden sadece birisi S kümesinde olan ayrıtlar kümesini, $E(S)$ bağlı olduğu her iki düğüm de S kümesinde olan ayrıtlar kümesini, $r(S)$ S kümesindeki müşteri taleplerini taşımak için gereken en az araç sayısını gösterebilir. Dikkat edileceği üzere $r(S)$ değerinin hesaplanması için q kapasitesi olan kutu paketleme problemi çözümlenmelidir. Ayrıca bir ayrıtlar alt kümesi $F \subseteq E$ için $x(F)$ fonksiyonu $\sum_{e \in F} x_e$ değerini gösterebilir.

ARP(2)

$$\min \sum_{e \in E} c_e x_e \quad (3.10)$$

Öyle ki;

$$x(\delta(\{i\})) = 2 \quad i \in N \quad (3.11)$$

$$x(\delta(S)) \geq r(S) \quad S \subseteq N, |S| \geq 2 \quad (3.12)$$

$$x_{ij} \in \{0, 1\} \quad 1 \leq i < j \leq n \quad (3.13)$$

$$x_{ij} \in \{0, 1, 2\} \quad i = 0, j = 1, \dots, n \quad (3.14)$$

ARP(2) modelinin amaç fonksiyonu (3.10) kullanılan ayrıtların maliyetlerinin toplamını (kat edilen mesafeyi) en aza indirmeyi hedefler. Kısıt kümesi (3.11) her bir müşteri düğümünün derecesinin iki olmasını, böylece her düğüme bir defa araç gelmesini ve bir defa araç ayrılmasını garantiler. Kısıt kümesi (3.12) verilen bir müşteri düğümleri alt kümesi S için, bir ucu bu kümede diğer ucu bu küme dışında olan ayrıtların değerlerinin toplamının bu alt kümedeki talepleri karşılayacak araç sayısının iki katından daha az olmamasını garantiler. Kapasite eşitsizliği olarak anılan bu kısıt ile hem araç kapasitesi dikkate alınır hem de alt tur oluşumu önlenir. Kısıt kümesi (3.13) müşteri düğümlerini bağlayan ayrıtlarının 0 veya 1 değeri alabileceğini, Kısıt kümesi (3.14) ise depo ile bağlantılı ayrıtların 0, 1 veya 2 değeri alabileceğini ifade eder. Şayet bir araç depodan çıkıp sadece bir i müşterisi ziyaret edip tekrar depoya dönüyor ise x_{0i} ayrıtı 2 değerini alır. Araç sayısının m adetten az olması ile ilgili kısıt $x(\delta(\{0\})) \leq 2m$ şeklinde modele eklenebilir.

Bu modelde önemli bir zorluk $r(S)$ değerinin optimal olarak belirlenmesidir. Verilen bir S kümesi için bu değeri belirlemek için NP-Zor olan Kutu Paketleme Probleminin çözülmesi gerekir. Ayrıca S alt kümelerinin sayısı düğüm sayısı arttıkça üstel olarak artmaktadır. Literatürde genellikle $r(S)$ değeri yerine buna bir alt sınır olan $k(S) = \lceil q(S)/q \rceil$ değeri kullanılmaktadır ve bu kısıtlar yuvarlanmış kapasite eşitsizliği (rounded capacity inequality) olarak anılmaktadır.

3.2 Periyodik Araç Rotalama Problemi

Periyodik araç rotalama problemi (PARP) araç rotalama probleminin literatürde sıkça çalışılan türevlerinden bir tanesidir. PARP'ın ARP'den farkı araç rotalamanın ve müşteri ziyaretlerinin belirlenen bir periyot için planlanmasıdır. Müşterilerin planlama periyodu içerisindeki ziyaret sayılarını belirleyen frekansları vardır ve ziyaret günleri bu frekanslara uygun olarak belirlenen olurlu çizelgelerinden birisi seçilerek gerçekleştirilir.

Örneğin pazartesi cumartesiye, 6 günlük bir planlama periyodu olan problemi ele alalım. Bir müşterinin haftada iki defa ziyaret edilmesi gerekiyor ise bu ziyaretler {pazartesi, perşembe}, {salı, cuma} veya {çarşamba, cumartesi} olurlu çizelgelerinden birisini seçerek yapılabilir.

Genel olarak müşterilerin taleplerine göre periyot içerisinde hem müşterilerin talebini zamanında karşılayacak hem de ziyaretleri en az araç ve en az maliyetle tamamlamayacak bir sonuç hedeflenir. PARP için gereken ek notasyon ve PARP matematisel programlama modeli şu şekildedir:

Kümeler

T : Planlama periyodu içerisindeki gün sayısı

S : Olurlu çizelgeler kümesi

S_i : i müşterisinin ziyaret sıklığı f_i olan olurlu çizelgeler kümesi

Parametreler

a_{st} : t günü s çizelgesinde ise 1, diğer durumda 0

d_i : i müşterisinin her bir ziyaretteki talebi

f_i : i müşterisinin periyot içerisinde gerçekleştirilmesi gereken ziyaret sayısı

Karar Değişkenleri

x_{ijt} : İkili karar değişkenidir, eğer i müşterisinden j müşterisine t gününde gidildiyse 1 değerini, diğer durumda ise 0 değerini almaktadır.

y_{is} : İkili karar değişkenidir, s çizelgesi i müşterisine atanmış ise 1, diğer durumda 0

u_{it} : i müşterisi t gününde ziyaret edildikten sonra araçtaki toplam yük miktarı

PARP(1)

$$\min \sum_{i \in N_0} \sum_{j \in N_0} \sum_{t=1}^T c_{ij} x_{ijt} \quad (3.15)$$

Öyle ki;

$$\sum_{j \in N_0} x_{ijt} = \sum_{s \in S_i} a_{st} y_{is} \quad i \in N, t = 1, \dots, T \quad (3.16)$$

$$\sum_{i \in N_0} x_{ijt} = \sum_{s \in S_i} a_{st} y_{is} \quad j \in N, t = 1, \dots, T \quad (3.17)$$

$$\sum_{i \in N} x_{oit} \leq m \quad t = 1, \dots, T \quad (3.18)$$

$$\sum_{s \in S_i} y_{is} = 1 \quad i \in N \quad (3.19)$$

$$u_{it} - u_{jt} + q x_{ij} < q - d_j \quad i \in N, j \in N, t = 1, \dots, T \quad (3.20)$$

$$d_i \leq u_i \leq q \quad i \in N, t = 1, \dots, T \quad (3.21)$$

$$x_{ijt} \in \{0, 1\} \quad i, j \in N_0, t = 1, \dots, T \quad (3.22)$$

$$y_{is} \in \{0, 1\} \quad i \in N, s \in S_i \quad (3.23)$$

Amaç fonksiyonu (3.15) (aynı araç rotalama probleminde olduğu gibi) kat edilen mesafeyi en küçüklemeyi hedefler. Kısıt kümeleri (3.16) ve (3.17) her müşteriye t gününde eğer ziyaret edilecekse tek bir aracın gelmesini ve tek bir aracın ayrılmasını garanti altına almaktadır. Kısıt kümesi (3.18) her gün depodan en fazla m sayıda araç çıkmasını garanti altına almaktadır. Kısıt kümesi (3.19) her i müşterisi için 1 ziyaret çizelgesinin seçilmesini garantiler. Kısıt kümeleri (3.20) ve (3.21) araçların alt tur (subtour) yapmasını, yani belirli müşteriler arasında bir döngü içerisine girmesini engellemekte ve aracın kapasitesinden fazla yük almamasını sağlamaktadır.

ARP için önerilen ikinci modele benzer şekilde PARP için de benzer bir model önerilebilir. Bu model için x_{ijt} değişkeni yerine x_{et} değişkeni kullanılacaktır. Tam sayılı karar değişkeni x_{et} e ayrıtının t periyodunda kaç defa kullanıldığını gösterir.

Ayrıca ARP(2)'de tanımlanan $x(F)$ fonksiyonuna t indeksinin eklenmesi gerekmektedir. Bu durumda tanım şu şekilde güncellenebilir: Bir ayrıt alt kümesi $F \subseteq E$ ve $t = 1, \dots, T$ için $x(F, t)$ fonksiyonu $\sum_{e \in F} x_{et}$ değerini gösterebilir.

$$\begin{aligned} & \mathbf{PARP(2)} \\ & \min \sum_{t=1}^T \sum_{e \in E} c_e x_{et} \end{aligned} \quad (3.24)$$

Öyle ki;

$$x(\delta(\{i\}), t) = \sum_{s \in S_i} a_{st} y_{is} \quad i \in N, t = 1, \dots, T \quad (3.25)$$

$$x(E(S), t) \leq |S| - r(S) \quad S \subseteq N, |S| \geq 2, t = 1, \dots, T \quad (3.26)$$

$$\sum_{s \in S_i} y_{is} = 1 \quad i \in N \quad (3.27)$$

$$x_{ijt} \in \{0, 1\} \quad 1 \leq i < j \leq n, t = 1, \dots, T \quad (3.28)$$

$$x_{ijt} \in \{0, 1, 2\} \quad i = 0, j = 1, \dots, n, t = 1, \dots, T \quad (3.29)$$

$$y_{is} \in \{0, 1\} \quad i \in N, s \in S_i \quad (3.30)$$

PARP(2) modelinin amaç fonksiyonu (3.24) kullanılan ayrıtların maliyetlerinin toplamını (kat edilen mesafeyi) en aza indirmeyi hedefler. Kısıt kümesi (3.25) bir müşteri için t gününde ziyaret etmeyi gerektiren bir çizelge seçilmiş ise bu müşteri düğümünün derecesinin t günü için iki olmasını, aksi durumda ise 0 olmasını garantiler. Kısıt kümesi (3.26) verilen bir müşteri düğümleri alt kümesi S ve bir periyot t için, her iki ucu bu kümede olan ayrıtların değerlerinin toplamının bu alt kümedeki düğüm sayısı ile talepleri karşılayacak araç sayısının farkından az olmasını garantiler. Kısıt kümesi (3.27) her müşteri için bir olurlu çizelge seçilmesini sağlar. Kısıt kümesi (3.28) müşteri düğümlerini bağlayan ayrıtlarının 0 veya 1 değeri alabileceğini, Kısıt kümesi (3.29) ise depo ile bağlantılı ayrıtların 0, 1 veya 2

deęeri alabileceęini ifade eder. Kısıt kümesi (3.30) çizelge seçme kararına karşılık gelen ikili deęişkenlerini tanımlar.

Bölüm 4

Periyodik Araç Rotalama Problemi - 2 Ziyaret Tipi

Bu bölümde Periyodik Araç Rotalama Problemi - 2 Ziyaret Tipi (PARP-2ZT) üzerinde çalışılmıştır. Alt bölüm 4.1'de problemin tanımı yapılarak matematiksel programlama modeli geliştirilmiş ve hesaplama karmaşıklığı sunulmuştur. Alt bölüm 4.2'de problem için geliştirilen alt sınır algoritmaları açıklanmıştır. Probleme özgü geliştirilen sezgisel yöntem Alt bölüm 4.3 altında sunulmuştur. Alt bölüm 4.4'de PARP-2ZT örneklerini oluşturma yöntemini ve sayısal deney sonuçlarını raporlamıştır.

4.1 Problem tanımı

Bu projenin birinci bölümünde bilinen periyodik araç rotalama problemine farklı bir açıdan bakılmıştır. Projenin esin kaynağı olan Coca Cola İçecek işleyiş sistemi göz önünde bulundurularak, iki tip aracın bulunduğu bir periyodik araç rotalama problemi üzerine çalışılmıştır. Bu problemde iki farklı tipteki araçların kapasiteleri birbirinden hem miktar hem de cins olarak farklıdır. Bir aracın fiziksel kapasitesi bulunurken diğer aracın zamansal olarak kapasitesi bulunmaktadır. Buna ek olarak, problemin kısıtlarından bir tanesi de bir

aracın ziyaret ettiği müşterinin, bir gün sonrasında diğer tip araçla ziyaret edilmek zorunda olmasıdır. Gerçek hayatta ilk araç sipariş alan ve mağazada raf düzenlemesi (teşhir) yapan kişileri temsil etmektedir. İkinci araç ise bir gün önce alınan siparişi mağazaya getiren nakliye aracıdır. Bildiğimiz kadarıyla, literatürde Periyodik Araç Rotalama Problemi'nin böyle bir türevi üzerine çalışılmamıştır. PARP notasyonuna ek olarak PARP-2ZT'ye özgü notasyon ve matematiksel model şu şekildedir:

Kümeler

V : Araç tipi kümesi $V = \{1, 2\}$ (1: sipariş toplama, 2: Nakliye)

Parametreler

a_{stv} : v tip araç için t günü s çizelgesinde ise 1, diğer durumda 0

d_{iv} : i müşterisinin v tipi araç için talebi (1. araç için zaman, 2. araç için fiziksel)

q_v : v tipi aracın kapasitesi (1. araç için zaman, 2. araç için fiziksel)

m_v : v tipi araç için kullanılabilir araç sayısı

r_{ij} : i müşterisinden j müşterisine gitmek için geçen süre

Karar Değişkenleri

x_{ijtv} : İkili karar değişkenidir, eğer i müşterisinden j müşterisine t gününde v tipi araç ile gidildiyse 1 değerini, diğer durumda ise 0 değerini almaktadır.

u_{itv} : i müşterisi t gününde v tipi araç ile ziyaret edildikten sonra araçtaki toplam yük\aracın geçirmiş olduğu toplam zaman miktarı

PARP-2ZT(1)

$$\min \sum_{i \in N_0} \sum_{j \in N_0} \sum_{v=1}^2 \sum_{t=1}^T c_{ij} x_{ijtv} \quad (4.1)$$

Öyle ki;

$$\sum_{j \in N_0 \setminus \{i\}} x_{ijtv} = \sum_{s \in S_i} y_{is} a_{stv} \quad i \in N, t = 1, \dots, T, v \in \{1, 2\} \quad (4.2)$$

$$\sum_{j \in N_0 \setminus \{i\}} x_{jivt} = \sum_{s \in S_i} y_{is} a_{stv} \quad i \in N, t = 1, \dots, T, v \in \{1, 2\} \quad (4.3)$$

$$\sum_{i \in N} x_{0itv} \leq m_v \quad t = 1, \dots, T, v \in \{1, 2\} \quad (4.4)$$

$$\sum_{s \in S_i} y_{is} = 1 \quad i \in N \quad (4.5)$$

$$u_{itv} - u_{jtv} + q_v x_{ijtv} \leq q_v - d_{jv} \quad i, j \in N, t = 1, \dots, T, v \in \{2\} \quad (4.6)$$

$$d_{iv} \leq u_{itv} \leq q_v \quad i \in N, t = 1, \dots, T, v \in \{2\} \quad (4.7)$$

$$u_{itv} - u_{jtv} + (q_v + r_{ij}) x_{ijtv} \leq q_v - d_{jv} \quad i, j \in N, t = 1, \dots, T, v \in \{1\} \quad (4.8)$$

$$d_{iv} \leq u_{itv} \leq q_v \quad i \in N, t = 1, \dots, T, v \in \{1\} \quad (4.9)$$

$$x_{ijtv} \in \{0, 1\} \quad i, j \in N_0, \quad t = 1, \dots, T, v \in \{1, 2\} \quad (4.10)$$

$$y_{is} \in \{0, 1\} \quad i, j \in N, s \in S_i \quad (4.11)$$

Proje önerisinde verilmiş olan modelde müşteri ziyaret günleri ile ilgili herhangi bir kısıt bulunmamaktadır. Örneğin; Periyot süresi 4 gün olan bir problem için ziyaret sıklığı 2 gün olan bir müşteri, birinci ve ikinci günlerde ziyaret edilebilmektedir. Ancak gerçek hayat uygulamasına uygun bir yöntem olmayışı ve yapılan literatür taraması sonucunda model olurlu çizelge yaklaşımı ile modellenmiştir (Baldacci vd., 2011). Buna göre periyot uzunluğu ve ziyaret sıklığı göz önünde bulundurularak, müşteriler önceden belirlenmiş çizelgelere göre ziyaret edilmektedir. Örneğin; periyot süresi 4 gün olan bir problemde ziyaret sıklığı 2 gün olan bir müşteriyi ele aldığımızda, müşteri ziyareti için birinci ve üçüncü gün ziyaret içeren çizelge ya da ikinci ve dördüncü gün ziyaret içeren çizelgeyi seçebilir. PARP-2ZT

probleminde belirtildiği üzere 2 farklı tipte araç tarafından ziyaret bulunmaktadır. Bu ziyaretler ise birbirlerini takip eden günlerde gerçekleştirilmektedir. Eğer S_i çizelgesi 1. araç tipi için 1. ve 3. günleri içeriyorsa 2. araç tipi için 2. ve 4. günleri içerir. Böylece gerçek hayat uygulamalarına daha uygun bir ziyaret sistemi kullanılmış olmaktadır.

PARP-2ZT problemi de, PARP(2)'ye benzer şekilde modellenenbilir. Bu model için ihtiyaç duyulan ek tanımlar açıklanmıştır. Tam sayılı karar değişkeni x_{et}^v e ayrıtının t periyodunda v tipi araçlar ile kaç defa kullanıldığını gösterir. Çizelgelerin ziyaret günleri gösteren a_{st} parametresi de a_{st}^v şeklinde güncellenir ve v tipi araç s çizelgesinde t günü ziyaret yapıyor ise 1 aksi durumda 0 değerini alır.

Ayrıca PARP(2)'de tanımlanan $x(F, t)$ fonksiyonuna v indeksinin eklenmesi gerekmektedir. Bu durumda tanım şu şekilde güncellenebilir: Bir ayrıt alt kümesi $F \subseteq E$, $t = 1, \dots, T$ ve $v \in \{1, 2\}$ için $x(F, t, v)$ fonksiyonu $\sum_{e \in F} x_{et}^v$ değerini gösterebilir.

Yük kapasitesini gözetilen ve alt turları önleyen kısıt kümeleri (3.12) ve (3.26) için $r(S)$ fonksiyonu gereklidir. Bu fonksiyon verilen bir S ayrıt alt kümesindeki düğümlerin taleplerini karşılamak için gereken araç sayısını bulur. Bu fonksiyonuna benzer şekilde her araç için tur süresi limitini de dikkate alan şekilde güncellenmesi gerekmektedir. Bu amaçla $r(S, v)$ fonksiyonu tanımlanmıştır. Verilen bir S ayrıt alt kümesi ve v tipi araç için gerekli olan en az araç sayısını veren fonksiyon $r(S, v)$ olsun.

PARP-2ZT(2)

$$\min \sum_{v \in \{1,2\}} \sum_{t=1}^T \sum_{e \in E} c_e x_{et}^v \quad (4.12)$$

Öyle ki;

$$x(\delta(\{i\}), t, v) = \sum_{s \in S_i} a_{st}^v y_{is} \quad i \in N, t = 1, \dots, T, v \in \{1, 2\} \quad (4.13)$$

$$x(E(S), t, v) \leq |S| - r(S, v) \quad S \subseteq N, |S| \geq 2, t = 1, \dots, T, v \in \{1, 2\} \quad (4.14)$$

$$\sum_{s \in S_i} y_{is} = 1 \quad i \in N \quad (4.15)$$

$$x_{ijt}^v \in \{0, 1\} \quad 1 \leq i < j \leq n, t = 1, \dots, T, v \in \{1, 2\} \quad (4.16)$$

$$x_{0jt}^v \in \{0, 1, 2\} \quad 1 \leq j \leq n, t = 1, \dots, T, v \in \{1, 2\} \quad (4.17)$$

$$y_{is} \in \{0, 1\} \quad i \in N, s \in S_i \quad (4.18)$$

4.1.1 Hesaplama Karmaşıklığı

Bu bölümde PARP-2ZT probleminin hesaplama karmaşıklığı incelenmiş ve problemin NP-Zor olduğu Teorem 1 ile ispatlanmıştır. İspat şöyledir;

Teorem 1. *PARP-2ZT NP-Zordur.*

İspat.] Problemin hesaplama karmaşıklığının analizi için problemin özel bir hali PARP'ye benzetilmiştir. Bu özel halinde aşağıda belirtilmiş olan varsayımlar göz önüne alınarak iki farklı tipteki araç için araç kapasitelerini ve müşteri taleplerini birbirlerine eşitlemek amaçlanmıştır. Bunu yaparken fiziksel kapasiteye sahip 2. tip araç için geçerli olan fiziksel talep ile, 1. tip araç için geçerli olan yolda harcanan süre ve müşteride harcanan süre arasında ilişki kurulmuştur. Böylece 1. tip araç için farklı bir karar verilmeyecek ve problem bir PARP örneğine dönüşmüş olacaktır.

- Satış elemanları tarafından kullanılan birinci tip araç ve fiziksel kapasiteye sahip ikinci

tip araçların kapasiteleri birbirine eşittir.

$$q_1 = q_2 = q$$

- Satış elemanları tarafından kullanılan birinci tip araçların seyahat süreleri (r_{ij}) i ve j müşterilerinin taleplerinin toplamının yarısından j müşterisinin servis süresi çıkarılarak hesaplanmaktadır.

$$r_{ij} = \frac{d_{i2} + d_{j2}}{2} - d_{j1} \quad i, j \in N_0 \quad (4.19)$$

$$d_{02} = d_{01} = 0$$

- (4.19) denkleminde hiç bir seyahat süresinin (r_{ij}) negatif çıkmamasını sağlamak üzere, birinci tip araç için geçerli olan servis süreleri j müşterilerinin ikinci araç için taleplerinin yarısından küçüktür.

$$0 < d_{j1} < \frac{d_{j2}}{2} \quad j \in N$$

Yukarıdaki varsayımlar göz önünde bulundurulduğunda, herhangi bir rota için rota üzerindeki bir noktada ikinci tip aracın üzerindeki yük miktarı ile birinci aracın o noktaya kadar geçirmiş olduğu toplam süre birbirine eşit olacaktır. Araçların yük ve süre açısından kapasiteleri birbirine eşit olduğu ($q_1 = q_2$) düşünüldüğünde fiziksel kapasiteye sahip araçlar ile satış elemanlarının araçları için kapasite açısından olurlu rota setleri birbirine eşit olacaktır. Rotaların uzunluklarının da birbirine eşit olduğu dikkate alındığında 2. araç için optimal olan rotaların birinci araç için de optimal olacağı görülmektedir. Çünkü bir tip araç için verilen her olurlu rota diğer araç tipi için de olurlu olacaktır. Bir başka deyişle, satış elemanları için herhangi bir rota kararı verilmesi gerekli değildir. Bu durumda PARP-2ZT(1) modelinde bulunan x_{ijtv} karar değişkenindeki v dizinin bir önemi kalmayarak x_{ijt} karar değişkenine dönüştürülebilir. Böylece gerekli düzenlemeler yapıldığında PARP-2ZT için geliştirilmiş PARP-2ZT(1) modeli PARP için literatürde verilmiş olan PARP(1) modeline dönüştürülebilir. PARP'ın NP-Zor olduğu bilinmektedir. Buradan yola çıkarak

PARP-2ZT matematiksel modelinin özel bir durumunun NP-Zor olduđu, yani PARP-2ZT probleminin de NP-Zor olduđu kanıtlanmaktadır. \square

4.2 Alt Sınır Algoritmaları

PARP-2ZT problemi için iki farklı alt sınır algoritması geliştirilmiştir. İlk algoritma kurucu bir sezgiseldir. İkinci algoritma ise dal-kesi yöntemini kullanan bir kesin çözüm algoritmasıdır. Belirli bir düğüm seçme politikası ile bu algoritma da alt sınır algoritması olarak kullanılmıştır.

4.2.1 Kurucu Alt Sınır Algoritması

Alt sınır bulmak üzere bir kurucu sezgisel yöntem geliştirilmiş ve Algoritma 1 olarak sunulmuştur. Bu yöntemde alt sınır hesaplanırken müşterilerin ziyaret sıklığı ve en yakın düğümler gözetilmiştir. İlk aşamada bir müşteri seçilmekte ve bu müşterinin periyot içerisindeki ziyaret sıklığına bakılmaktadır. Bu müşterinin ziyaret sıklığı ikiyle çarpılarak periyot içerisinde bu müşterinin dahil olduđu kaç tane yol kullanılacağı hesaplanmaktadır. Örneğin bir müşterinin ziyaret sıklığı üç ise bu müşteriye periyot içerisinde 3 kere gidilip gelineceği için müşterinin etrafındaki 6 bağlantı kullanılacaktır. Bir müşteri düğümü için kaç kere ziyaret edileceği hesaplandıktan sonra kendisine en yakın müşteri düğümü tespit edilmektedir. Tespit edilen bu müşteri düğümünün periyot içerisindeki ziyaret sıklığına bakılmaktadır ve iki müşteri arasına daha az frekansa sahip müşteri belirlenmekte ve bu frekans sayısı kadar yol bağlanmaktadır. Daha sonra müşteri frekansı bu sayı kadar azaltılmakta ve güncellenen frekans 0'dan büyük ise kendisine en yakın ikinci müşteri düğümüne geçilmekte ve aynı işlem tekrar edilmektedir. Bu şekilde devam eden algoritma sonucunda alt sınır elde edilmektedir. Elde edilecek tüm olurlu sonuçlarda müşteriler iki farklı araç tarafından ziyaret sıklığı parametresinin gereğince, ziyaret sıklığı değerinin iki katı kadar ziyaret edilmek zorundadır. Ancak kapasite ve olurlu çizelge kısıtları dolayısıyla bu ziyaretler her zaman kendisine en yakın müşterilere yapılamayacaktır. Bu durumda kendilerine en yakın müşterilerin ziyaret

edildiği varsayımıyla elde edilen çözüm olası çözümlerden daha yüksek bir maliyete sahip olamaz ve alt sınır görevi görecektir.

Girdi:

N düğümlük müşteri kümesi

Müşterilerin koordinatları

Müşterilerin ziyaret sıklığı

Çıktı:

Toplam tur maliyeti

Başla

Her $i \in N$ için yap

$a_i = 2f_i$

$a_i > 0$ sürece yap

En yakın müşteriyi (j) bul

Eğer $j = depo$ ise

i müşterisi ile depo arasında a_i kadar yol bağla

$a_i = 0$

Aksi takdirde

Müşteriler arasında $\min\{f_i, f_j\}$ kadar yol bağla

Müşteri i ve j arasındaki mesafeyi M olarak belirle

$a_i = a_i - \min\{f_i, f_j\}$

Seçilen yolların maliyetlerini topla

Algoritma 1: Alt Sınır Algoritması

Ayrıca literatür taraması sonucunda LP gevşetme tekniğinin de ARP için basit bir alt sınır belirleme tekniği olarak kullanıldığı görülmüştür. Bu sebeple geliştirilmiş olan alt sınır algoritması ile LP gevşetme sonuçlarını karşılaştırıldı. Sonuç olarak LP gevşetme ve geliştirmiş olduğumuz alt sınır algoritmasının birbirlerine karşı kesin bir üstünlük sağlayamadığı görülmektedir. Deney sonuçları Alt bölüm 4.4 içinde raporlanmıştır.

4.2.2 Dal-Kesi Algoritması

PARP-2ZT probleminin çözümü için dal-kesi (branch-and-cut) tabanlı bir kesin algoritma geliştirilmiştir. Bu algoritma PARP(2) tabanlı bir model kullanmaktadır. PARP(2) modelinin kapasite eşitsizliği olarak anılan (3.26) kısıt kümesinde düğüm sayısının üstel fonksiyonu kadar kısıt vardır ve optimal sonuçta bu kısıtların çoğu bağlayıcı değildir. Bu sebeple dal-kesi algoritması kapsamında kapasite eşitsizliklerinin olmadığı PARP(2) modeli, $PARP'$, ele alınmış ve bu modelin doğrusal gevşetimi çözülmüştür. Elde edilen sonuç genellikle oldukça zayıf bir alt sınıra karşılık gelmektedir ve bu sonuç çeşitli kesen düzlemler eklenerek geliştirilebilir. Verilen bir doğrusal gevşetim optimal sonucu için belirli bir geçerli eşitsizlik için tasarlanmış ayırma algoritmaları ile (şayet varsa) sağlanmayan geçerli eşitsizlikler tespit edilebilir ve modele eklenerek model tekrar çözülür. Ayırma algoritmaları kesin veya sezgisel algoritmalar olabilirler.

Lysgaard vd. (2004) ARP için birçok farklı geçerli eşitsizlikleri kullanan bir algoritma geliştirmiş ve bu geçerli eşitsizliklerin performanslarını süre ve çözüm kalitesi açısından incelemiştir. Biz proje kapsamında PARP-2ZT için bir dal-kesi algoritması geliştirdik. Algoritmamız çeşitli adımlarında Lysgaard vd. (2004) tarafından hazırlanmış olan kütüphanelerden de yararlanmışır. Bildiğimiz kadarı ile PARP için dal-kesi tabanlı başka bir çalışma yoktur.

Algoritma 2 PARP için geliştirilen Dal-Kesi algoritmasının adımlarını göstermektedir. Algoritma öncelikle $PARP'$ modelinin kök düğümü ile başlar. Her dış iterasyonda düğüm kümesinden en düşük alt sınıra sahip düğümü seçerek devam eder. Seçilen düğümü düğüm kümesinden çıkartır ve bu düğümüne karşılık gelen modeli çözer. Optimal sonucun sağlamadığı geçerli eşitsizlikler bulur ise bunları modele ekleyerek tekrar çözer. Şayet bir geçerli eşitsizlik bulunamıyor ise eldeki çözümün tam sayılı olup olmadığını kontrol eder ve gerekiyorsa bilinen en iyi sonucu (üst sınırı) günceller. Modele yeni bir eşitsizlik eklenemiyor ve eldeki sonuç tamsayı değil ise bir karar değişkeni belirlenerek bu karar değişkeni üzerinden yeni alt düğümler oluşturulur ve bu düğümler düğüm kümesine eklenir.

Algoritmanın durma koşulu olarak tüm düğümlerin ziyaret edilmesi tanımlanmıştır. Ancak, süre, alt ve üst sınır arasındaki fark gibi çeşitli durma koşulları da algoritmaya eklenebilir. Bu algoritma bir kesin çözüm yöntemidir. Ancak probleme bir alt sınır belirleme amacı ile bu algoritmayı geliştirdiğimiz için düğüm seçme politikası her seferinde en düşük alt sınıra sahip düğümü seçmektedir.

Girdi:

$PARP'$ modeli

Kök düğüm

Çıktı:

Alt ve üst sınır değerleri

Başla

Düğüm kümesi boş olmadığı sürece yap

Alt sınır değeri olan düğümü seç

Eğer *Alt sınır üst sınırdan küçük ise ise*

Alt sınır değerini güncelle

Seçilen düğümü, düğüm kümesinden çıkart

Seçilen düğümde $PARP'$ modelini çöz

Sağlanmayan eşitsizlik bulunduğun sürece yap

Eşitsizliği $PARP'$ modeline ekle

$PARP'$ modelini çöz

Eğer *tüm karar değişkenleri tamsayılı ise*

Eğer *sonuç üst sınırdan iyi ise*

Üst sınırı güncelle

Aksi takdirde

Tam sayılı olmayan bir karar değişkeni seç

Yeni alt düğümler oluştur

Her düğümün alt sınırı belirle

Yeni düğümleri düğümler kümesine ekle

Aksi takdirde

Düğüm kümesini boşalt

Alt sınır değerini raporla

Varsa üst sınırı raporla

Algoritma 2: Dal Kesi Algoritması

Kullanılan Eşitsizlikler

Dal-kesi algoritması kapsamında beş farklı eşitsizlik tipi kullanılmıştır. Bunlardan ilki ARP için kullanılan yuvarlanmış kapasite eşitsizliği kesisidir. Diğer kesiler ise PARP için proje kapsamında geliştirilmiştir. Her bir eşitsizlik detaylı olarak açıklanmıştır.

Dal-kesi algoritmasının herhangi bir düğümünde çözülen modelin optimal sonucunda karar değişkenlerinin değerlerinin x_{et}^{*v} ile gösterildiğini varsayalım.

- *Kapasite eşitsizlikleri*

Bu eşitsizlik ARP için kullanılmaktadır ve verilen bir S alt kümesi için farklı şekillerde yazılabilir (Lysgaard vd., 2004).

- $x(\delta(S)) \geq 2r(S)$
- $x(E(S)) \leq |S| - r(S)$

Her iki eşitsizlik de ARP için uygundur, çünkü problem tek periyot için çözülmektedir. Ancak PARP için ilk kısıt uygun olmayabilir, çünkü verilen bir günde ayrıt değerlerinin toplamının belirli bir değer üzerinde olmaya zorlanması, müşterilerin o günde ziyaret edilme zorunluluğu doğurur. Bu sebeple ikinci tipteki kısıt dikkate alınarak eşitsizlikler eklenmiştir. Bu eşitsizliklerde kullanılan kapasite fiziksel kapasitedir.

Lysgaard vd. (2004) tarafından hazırlanmış olan kesi kütüphanesi, C++ programı içerisinde Mevcut optimal sonuç çağrılarla verilerek aşağıdaki eşitsizliği sağlamayan S alt kümeleri tüm t periyotları ve $v = 2$ için aranmıştır.

$$x(E(S), t, 2) \leq |S| - k(S, 2) \quad (4.20)$$

- *Tek ziyaret eşitsizlikleri*

Bu eşitsizlik planlama periyodu içerisinde bir defa ziyaret edilecek olan müşteriler için yazılmaktadır, $i \in N | f_i = 1$. PARP-2ZT probleminin optimal çözümünde $f_i = 1$ olan bir i müşterisinden çıkan bir araç t_1 gününde j_1 müşterisine ve $t_2 \neq t_1$ gününde j_2

müşterisine gidemez. Bu gözleme aykırı i, j_1, j_2 müşterileri, t_1, t_2 günleri ve v tipi araç bulunur ise aşağıdaki eşitsizlik eklenir.

$$x_{i,j_1,t_1}^v + x_{i,j_2,t_2}^v \leq 1 \quad (4.21)$$

Eğer $T > 2$ ise Eşitsizlik (4.21) t_1 ve t_2 'den farklı t günleri için i müşterisi ile j_t müşterisi arasındaki ayrıtlar eklenerek güçlendirilebilir. Buradaki ek j_t müşterileri en yüksek değerli $x_{i,j_t,t}^v$ değişkeni olacak şekilde seçilir.

$$x_{i,j_1,t_1}^v + x_{i,j_2,t_2}^v + \sum_{t=1|t \neq t_1, t \neq t_2}^T x_{i,j_t,t}^v \leq 1 \quad (4.22)$$

- *Çizelge eşitsizlikleri*

Bu eşitsizlik tek ziyaret eşitsizliğinin genel hali olarak düşünülebilir. Bir i müşterisi hiçbir olurlu çizelgesinde hem t_1 hem de t_2 gününde ziyaret edilemiyor ise optimal sonuçta bu iki günde birden kullanılan ve bir ucu i müşterisi olan bir ayrıt olamaz. Optimal sonuçta bu gözleme aykırı i, j_1, j_2 müşterileri, t_1, t_2 günleri ve v tipi araç bulunur ise aşağıdaki eşitsizlik eklenir.

$$x_{i,j_1,t_1}^v + x_{i,j_2,t_2}^v \leq 1 \quad (4.23)$$

- *Çizelgeli alt tur eşitsizlikleri*

Bu eşitsizlik belirli bir S müşteri alt kümesi içindeki ayrıtların toplam değerinin düğüm sayısının bir eksiginden fazla olması durumunda çözümün alt tur içereceği gözlemine dayanır, her S, t ve v için

$$x(E(S), t, v) \leq |S| - 1 \quad (4.24)$$

olmalıdır. Şayet S kümesi içerisinde yer alan bir i müşterisi t gününde v tipi araç ile ziyaret ediliyor ise $a_{st}^v y_{is}$ çarpımı bir değerini alır. Bu durumda $|S|$ yerine $\sum_{i \in S} \sum_{s \in S_i} a_{st}^v y_{is}$ ifadesi kullanılarak daha sıkı bir kısıt yazılması mümkündür.

Bu eşitsizlik için bir t gününde birbirine pozitif ayrıt değerli ile bağlı bir S müşteri alt kümesi bulunur ve bu kümede yer alan her müşterinin ilgili gün ziyaret edilmesine olanak veren çizelgelere karşılık gelen y değerlerinin toplamına bakılır. Şayet aşağıdaki eşitsizlik sağlanmıyor ise ilgili kısıt modele eklenir.

$$x(E(S), t, v) \leq \sum_{i \in S} \sum_{s \in S_i} a_{st}^v y_{is} - 1 \quad (4.25)$$

- *Çizelgeli kapasite eşitsizlikleri*

Bu eşitsizlik kapasite eşitsizliğinde yer alan $|S|$ sabiti yerine y_{is} değişkenlerinin getirilmesi ile elde edilen bir eşitsizliktir.

$$x(E(S), t, 2) \leq \sum_{i \in S} \sum_{s \in S_i | a_{st}^v = 1} y_{is} - k(S, 2) \quad (4.26)$$

- *Hypotour eşitsizlikleri ve Gomory Kesileri*

Bu iki eşitsizlik belirlemesi diğerlerine göre göreceli daha uzun süren eşitsizliklerdir. Lysgaard vd. (2004) bu eşitsizlikleri sadece kök düğümde ve diğer eşitsizliklerin hiçbirinden sonuç alınamazsa kullanmıştır. Biz de benzer şekilde sadece kök düğümde ve bir kere bu iki eşitsizlik ile ilgili kütüphaneleri çağırdık. Eğer modele yeni kısıt eklenir ise diğer eşitsizlikler aranmaya devam etmektedir. Şayet bu iki eşitsizlik ile ilgili bir kısıt bulunamaz ise, dal-kesi algoritmasının kök düğümdeki çalışması sona ermekte ve düğüm oluşturmaya başlamaktadır.

4.3 Sezgisel Çözüm Yöntemleri

Bu alt bölümde, geliştirilmiş olan DKA algoritması detaylı bir şekilde sunulmuştur. Önerilen DKA algoritması 3 ana adım içermektedir. İlk adım başlangıç çözümünün elde edilmesidir. Başlangıç çözümü elde etmek için olurlu bir çözüm üreten bir algoritma tasarlanmıştır. Komşuluk arama ikinci ana adımdır. Önerilen DKA algoritması için 3 farklı operatör uygulanmaktadır. Bu operatörler; Swap operatörü, move operatörü, ve ziyaret

kombinasyonu deęişikliği operatörüdür. Bu operatörler çözümden rastgele seçilen olurlu bir komşu çözüme ulaşmak için kullanılmaktadır. Son adımda ise, komşuluk arama ile elde edilen her bir sonucu geliştirmek amaçlı bir yerel arama metodu uygulanmıştır. Bu işlemler sonucunda elde edilen sonuç bu ana kadar elde edilen en iyi sonuçtan daha iyi ise elde edilen yeni sonuç kabul edilir ve en yakın komşuluk arama ile işleme devam edilir. Aksi durumda, benzetimli tavlama yaklaşımından gelen ihtimalle bu kötü sonuca geçilir veya yeni sonuca geçilmeden bir sonraki operatör kullanılarak başka bir komşu çözüme ulaşmaya çalışılır. Başlangıç çözümü oluşturulduktan sonra komşuluk arama ve yerel arama uygulamaları her bir problem için 15 dakika süresince devam eder. Bu DKA algoritması Mladenović ve Hansen (1997) tarafından önerilmiştir. DKA algoritmasının adımları Algoritma 3’de gösterilmiştir.

Başla

Başlangıç çözümü elde etme ($f(x)$)

Süre \leq *TimeLimit* **sürece yap**

$k = 1$

$k \leq 12$ **sürece yap**

Eğer $k \leq 3$ **ise**

| Swap operatörü ($f(x')$)

bitir

Eğer $k > 3$ *and* $k \leq 6$ **ise**

| Move Operatörü ($f(x')$)

bitir

Aksi takdirde

| Ziyaret Kombinasyonu Değişimi Operatörü ($f(x')$)

bitir

Yerel Arama($f(x'')$)

Eğer *Yeni sonuç daha düşük ise* ($f(x'') < f(x)$) **ise**

| Yeni sonucu kabul et

bitir

Aksi takdirde

| Yeni sonuç daha yüksek ise $e^{\frac{-(f(x'')-f(x))}{temperature}}$ ihtimalle sonucu kabul et

| $k = k + 1$

bitir

bitir

bitir

Bitir

Algoritma 3: DKA Algoritması

4.3.1 Bařlangıç Çözümü Elde Etme

Başlangıç çözümü elde etmek için Algoritma 4'de gösterilen basit bir sezgisel yöntem kullanılmıştır. Müşteri kümesi, müşterilerin atanabilecekleri olurlu çizelgeler kümesi, ziyaret sıklıkları, talep bilgileri ve araçların kapasite bilgisi bu sezgisel yöntem için girdi olarak kullanılmıştır.

Başlangıç çözümü elde etmek için ilk önce tüm müşteriler mümkün olan olurlu çizelgelere atanır. Bu atama gerçekleştirilirken en büyük talebe sahip müşteriden başlanır ve müşteri bir önceki müşterinin çizelgesinden bir sonraki çizelgeye atanır. Burada amaç çizelgelere ve dolayısıyla günlere yüklerin eşit bir şekilde dağılmasını sağlamaktır. Bu işlem tüm müşteriler olurlu bir çizelgeye atanıncaya kadar devam eder.

Sonraki adımda müşteriler çizelgelerindeki her bir gün için o güne ait en düşük toplam yükü olan büyük araca ve en düşük toplam rota süresi olan küçük araca atanmaktadır. Bu iki adım olurlu bir çözüm elde etme ihtimalini arttırmak için uygulanmaktadır ve sayısal deneylerde büyük ölçüde işlerlik göstermiştir. Son olarak elde edilen ilk sonucu iyileştirmek için her bir aracın rotası üzerinde yerel arama algoritması çalıştırılmıştır.

Girdi:

Müşteri Seti

Müşterilerin atanabileceği çizelge kümesi

Müşterilerin ziyaret sıklığı

Müşterilere ait talep ve zaman bilgisi

Büyük ve Küçük Araç kapasiteleri

Çıktı:

Başlangıç Çözümü

Başla

Müşterileri çizelgelere ata

Müşterileri araçlara ata

Rotaları oluştur

Yerel arama algoritması uygula

Toplam maliyeti hesapla

Algoritma 4: Başlangıç Çözümü Elde Etme

4.3.2 Komşuluk Arama

Bu bölümde, komşuluk arama için kullanılan operatörler açıklanmıştır. Bu operatörler sırasıyla; Swap operatörü, move operatörü ve ziyaret kombinasyonu değişikliği operatörüdür.

Swap Operatörü Swap operatörü aynı günde aynı tip araçta ancak farklı rotalarda bulunan iki müşterinin rotalarının kendi aralarında değiştirilmesini sağlamaktadır. Örneğin, 6 müşteriye ve 2 araca sahip bir örnek ele alalım. 1. aracın rotası 0-1-2-3-0, 2. aracın rotası ise 0-4-5-6-0 olsun. Swap operatörü 2. ve 5. müşteri için çalıştığında yeni oluşacak rotalar, 0-1-5-3-0 ve 0-4-2-6-0 olarak belirlenir. Swap operatörü sezgisel yöntemde her aşamada rassal sayıda (n) müşteri için uygulanmaktadır. Aynı komşuluk adımında birden fazla müşteri çifti aynı anda yer değiştirmekte ve en sonundaki durum dikkate alınıp komşu çözüme geçilip

geçilmeyeceğine karar verilmektedir. Bu şekilde daha uzak komşulara hızlı geçilebileceği düşünülmektedir. Sayısal deneylerde n için en yüksek değer 6 olarak belirlenmiştir. DKA algoritmasında swap operatörünün uygulanma şekli Algoritma 5’da gösterilmiştir.

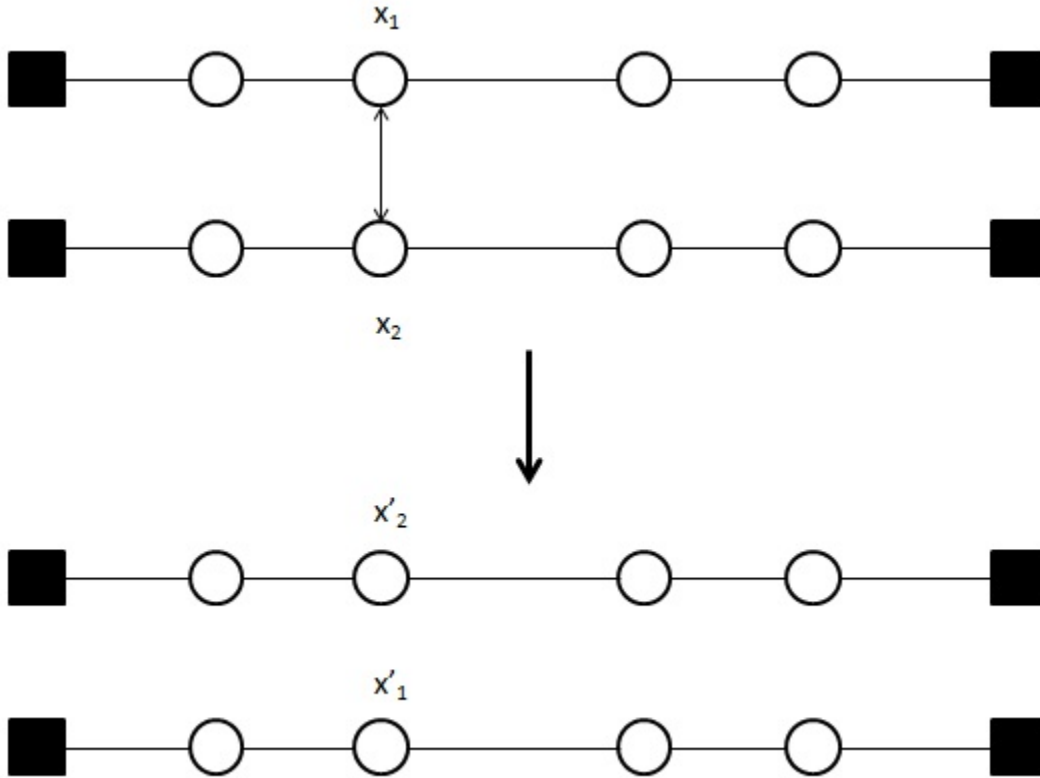
Başla

- Rastgele n seç (1 ile 6 arasında)
- Rastgele n adet gün seç
- Rastgele bir araç tipi seç
- Seçilen günlerin her biri için rastgele iki farklı araç seç
- Seçilen her araç çifti için rastgele bir müşteri çifti seç
- Müşteri çiftlerinin araçlarını birbirleri ile değiştir

Algoritma 5: Swap Operatörü

Bu operatörün 3 kere üstüste uygulanması sonucunda yeni bir komşu çözüme geçilememesi durumunda move operatörüne geçilmektedir. Şekil 4.1’de bu operatör için bir örnek verilmiştir.

Move Operatörü Move operatörü literatürde sıkça DKA algoritmasına uygulanan bir komşuluk arama operatörüdür (Algoritma 6). Bu operatör seçilen bir müşteriyi aynı gün içerisinde bulunan bir rotadan çıkarılıp, farklı bir rotaya yerleştirilmesini sağlamaktadır. Örneğin; 5 müşterili ve 2 araca sahip bir örnek ele alalım. 1. aracın rotası 0-1-2-3-0 olsun ikinci tip aracın rotası ise 0-4-5-0. Move operatörü 2. müşteri için çalıştığında, yeni rotalar 0-1-3-0, 0-2-4-5-0 şeklinde olacaktır. Bu işlem yapılırken araç kapasitelerinin aşılmamasına dikkat etmek büyük önem taşımaktadır. Move operatörü sezgisel yöntemde her aşamada rassal sayıda (n) müşteri için uygulanmaktadır. Aynı komşuluk adımında birden fazla müşteri aynı anda yer değiştirmekte ve en sonundaki durum dikkate alınıp komşu çözüme geçilip geçilmeyeceğine karar verilmektedir. Bu şekilde daha uzak komşulara hızlı geçilebileceği düşünülmektedir. Sayısal deneylerde n için en yüksek değer 6 olarak belirlenmiştir.



Şekil 4.1: Swap Operatörü

Başla

Rastgele n seç (1 ile 6 arasında)

Rastgele n adet gün seç

Rastgele bir araç tipi seç

Seçilen her gün için seçilen araç tipinde rastgele iki araç seç

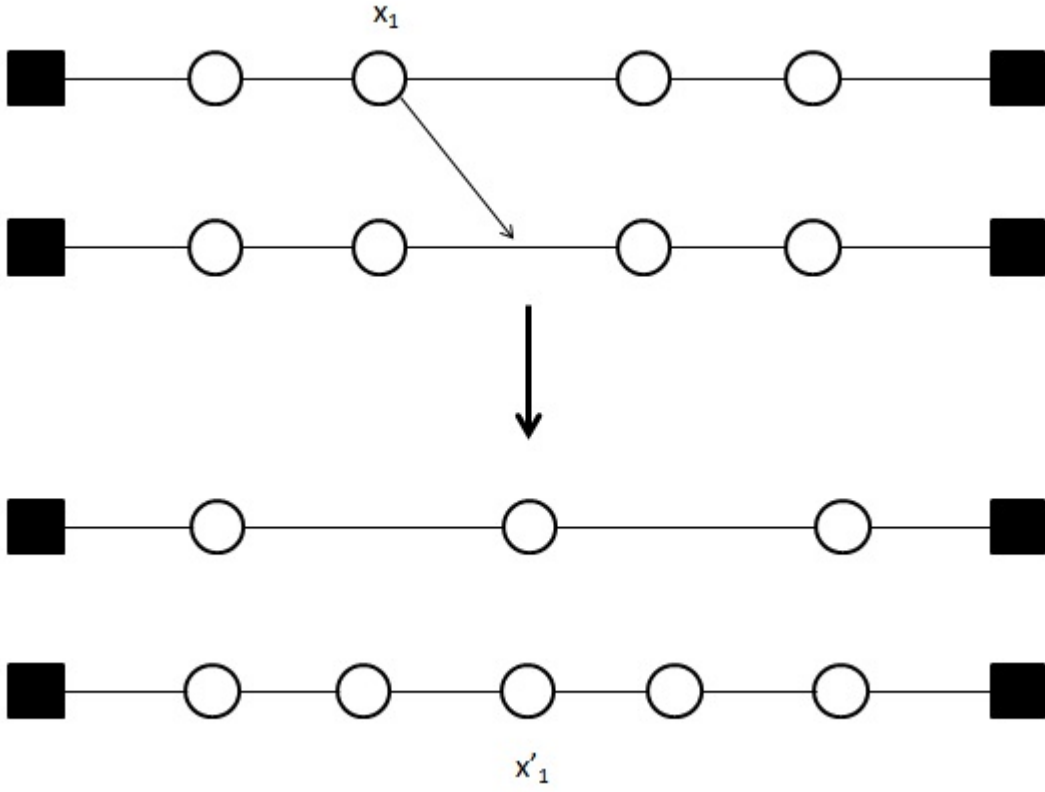
Seçilen araçlardan ilkinden rastgele bir müşteri seç Müşteriyi ilk araçtan çıkartıp

ikinci araca ekle

Algoritma 6: Move operatörü

Bu operatörün 3 kere üstüste uygulanması sonucunda yeni bir komşu çözüme geçilememesi durumunda ziyaret kombinasyonu değişimi operatörüne geçilmektedir. Şekil 4.2’de bu operatör için bir örnek sunulmuştur.

Ziyaret Kombinasyonu Değişimi Operatörü Ziyaret kombinasyonu değişimi, DKA algoritmasında kullandığımız son komşuluk arama operatörüdür. Bu operatör seçilen müşterinin atanmış olduğu çizelgenin mümkün olan diğer çizelgelerinden birisi ile değiştirilmesini sağlar. Bu değişiklik kapsamında aynı anda en fazla 6 müşterinin ziyaret çizelgesi değiştirilir. Bu operatör uygulanırken, diğer operatörlerden en önemli farkı ise, diğer operatörler tarafından uygulanan değişikliğin tek tip aracı etkilemesi, ziyaret kombinasyonu değişimi operatörünün ise yapılan bir değişiklikte her iki tip aracı da etkilemesidir. Bunun sebebi, PARP-2ZT probleminin özelliklerinden biri olan bir müşterinin farklı araçlar tarafından birbirini takip eden günlerde ziyaret edilmesi zorunluluğudur. Örneğin; Bir müşteri 3. günde 2. tip araç tarafından ziyaret ediliyorsa, 2. günde 1. tip araç tarafından ziyaret edildiği bilinmektedir. Eğer 2. tip araç tarafından 3. gün değil de 2. gün ziyaret gerçekleştirilirse, 1. tip araç tarafından gerçekleştirilen ziyaret de 2. günden 1. güne aktarılmak zorundadır. Bu operatörün 6 kere üstüste uygulanması sonucunda yeni bir komşu çözüme geçilememesi durumunda swap operatörüne geri dönülmektedir.



Şekil 4.2: Move Operatörü

4.3.3 Yerel Arama

Yerel arama algoritması olarak 2-opt algoritması kullanılmıştır. 2-opt algoritması ilk olarak Croes (1958) tarafından gezgin satıcı problemi çözmek için önerilmiştir. Bu algoritma 2 ayrıt seçip bu ayrıtları çaprazlayarak yeni rota kurma üzerine kuruludur. Çaprazlama işlemi yapıldıktan sonra rotanın gerekli olan kısmı ters yönde çevrilir ancak kullandığımız örnekler simetrik örnekler olduğu için çaprazlama işlemi sonrasında rotanın hangi kısmının ters yöne çevrildiğinin bir önemi yoktur. Buna ek olarak, 2-opt algoritması bir rota içerisinde gerçekleştirilebilecek tüm değişiklikleri dener ve en fazla iyileştirme sağlayan değişikliği kabul eder. Bu uygulama değişiklik içeren her bir rota için hiçbir olası çaprazlama daha iyi bir sonuç vermeyinceye kadar devam ederek en düşük maliyetli rotalar elde edilmeye çalışılır.

4.3.4 Benzetimli Tavlama

Daha önce belirtilen komşuluklar ile yapılan ilk deneylerde görülmüştür ki kötü sonuçlara giden bir sezgisel yaklaşım daha iyi sonuçlar üretmektedir. Bu bağlamda sezgisel yönteme literatürde sıkça kullanılan benzetimli tavlama yaklaşımı eklenmiştir. Buna göre komşu çözüm mevcut çözümden daha iyi bir amaç fonksiyonu değerine sahipse komşu çözüme geçilir. Bunun yanında komşu çözümün amaç fonksiyonu değeri daha kötü ise $e^{-\frac{-(f(x')-f(x))}{temp}}$ ihtimalle bu yeni sonuca geçilir. Bilinen en iyi sonuç ayrı bir yerde saklanır ve gerektiğinde güncellenir. Burada *temp* değeri sıcaklık sabitidir ve yüksek olması durumunda daha kötü sonuçlara geçme ihtimali artar. Ayrıca fonksiyonun yapısı gereği daha yakın kötü sonuçlara geçme ihtimali daha uzak kötü sonuçlara göre fazladır. Algoritmada sıcaklık sabiti başta 0.01 olarak belirlenmiştir. Üst üste 50 adımda daha iyi bir sonuca geçilememesi durumunda sıcaklık sabiti 4 değerine yükseltilir ve bu değer 300 adım boyunca korunur. 300 adım sonra değer yeniden 0.01'e düşürülür. Burada amaç lokal bir en iyi noktada sezgiselin takılması durumunda bu noktadan daha kötü sonuçlara geçme ihtimalini ciddi oranda artırarak kaçılmasının sağlanmasıdır.

4.4 Deney Sonuçları

Bu aşamada ilk olarak sayısal deneylerde kullanılacak örneklerin oluşturulma yöntemi anlatılacak ardından sayısal deneylerle ilgili sonuçlar sunulacaktır.

4.4.1 Örnek Problem Oluşturma

Oluşturmuş olduğumuz matematiksel model ve alt sınır algoritmaları test etmek için Cordeau vd. (1997) tarafından derlenmiş olan ve bir çok ARP ve PARP makalelerinde kullanılmış olan örnek problemleri kullanmaktayız. 1-10 arasındaki örnekler Eilon vd. (1971) tarafından ARP için sunulmuş ve Christofides ve Beasley (1984) tarafından PARP için uyarlanmıştır, 11. örnek Russell ve Igo (1979), 12. örnek Cook ve Russell (1978), 13. örnek Russell ve Gribbin (1991) den alınmıştır. 14-32 arasındaki örnekler Chao vd. (1995) makalesinden alınmıştır. Cordeau vd. (2001), Angelelli ve Speranza (2002), Baptista vd. (2002), Cacchiani vd. (2014), ve Baldacci vd. (2011) örneklerin kullanıldığı diğer makalelere örnek olarak gösterilebilir. Örnek kümemizde 42 tane örnek problem bulunmaktadır. Bu problemlerde en az müşteri sayısı 20, en çok müşteri sayısı ise 417 dir. Müşterilerin koordinat bilgileri, talep bilgileri, ziyaret sıklığı, olası ziyaret kombinasyon sayıları ve bu kombinasyonlar veri setinin içerisinde bulunmaktadır.

Proje kapsamında önerilen problem literatürde daha önce çalışılmamış bir problem olduğu için Cordeau vd. (1997) tarafından derlenmiş olan örnekler olduğu gibi kullanılmamaktadır. Bu sebeple, bu örnek problemler üzerine bazı eklemeler yapılmıştır. 2. tip araç için kullanılan veriler Cordeau vd. (1997) tarafından derlenen örneklerden olduğu gibi alınmıştır. 1. araç için ise yeni parametreler oluşturulmuştur. Birinci araç için gereken müşterilerde geçirilecek zaman parametreleri müşteri taleplerine küçük bir rassal değer (beklenen değeri 0) eklenerek hesaplanmıştır. Her bir müşteri ikilisi için aralarındaki yolda harcanan zaman ise müşteriler arasındaki mesafeye eşitlenmiştir. Son olarak, 1. tip araç kapasitesi hesaplanırken öncelikle araç sayısı ile gün sayısı çarpımına müşterilerin frekansları toplamı eklenip tüm problem için kullanılacak toplam ayrıt sayısı hesaplanmış ve bu değer ortalama ayrıt

süresi ile çarpılarak araçların yolda geçirecekleri toplam zaman tahminlenmiştir. Bu değer araç sayısı-gün çarpımına bölünerek tek bir araç için değer hesaplanmış ve bu değere yine rota başına düşen ortalama müşteri süresi eklenmiştir. Bu değer yüzde 30 oranında artırılarak 1. tip araç kapasitesi belirlenmiştir.

4.4.2 Kurucu Alt Sınır Sonuçları

Kurucu alt sınır algoritmasının performansı rassal olarak hazırlanmış örnekler üzerinde test edilmiştir. Tablo 4.1 de 5 müşterilik 10 adet küçük örnek ele alınmıştır. Bu örnekler proje başlangıcında tarafımızca PARP-2ZT için geliştirilmiş olan matematiksel model PARP-2ZT(1) ile GAMS versiyon 22.5'te çözülmüş ve elde edilen sonuçlar listelenmiştir. 10 adet örneğin 8 tanesinde saniye cinsinden verilmiş olan süreler içinde en iyi sonuca ulaşılmıştır. Diğer 2 örnekte ise deneylerimiz için üst süre sınırı olan 1800 saniye içerisinde en iyi sonuca ulaşamamıştır ve bu sonuçlar için alt sınır ile ulaşılan sonuç arasındaki farklar listelenmiştir.

Tablo 4.2 de 20 müşterilik 10 adet küçük örnek üzerinde yapılan testler sunulmuştur. Matematiksel model belirtilen zaman içerisinde hiç bir örnek için en iyi sonuca ulaşamamıştır. GAMS kesin çözücü ile bulunan alt sınır ile aradaki farklar listelenmiş ve farkların oldukça yüksek olduğu gözlemlenmiştir. PARP-2ZT için geliştirilmiş olan matematiksel model 20 müşterilik problemler için dahi en iyi sonuçlara ulaşmakta başarılı olamamıştır. Yani projenin esin kaynağı olan Coca Cola İçecek firmasının 1000'den fazla müşterili problemini çözmek için kesin bir çözüm yöntemi kullanılamayacağı çok açıktır. Bu da problemimizin çözümü için sezgisel yöntem geliştirme ihtiyacını kanıtlamaktadır.

Geliştirilen yöntemlerin kalitesini anlamak için üst ve alt sınırlar arasındaki farka bakmak hedeflenmiştir. PVRP-2ZT üzerinde bu yöntemleri test etmeden önce, literatürde en iyi sonuçları bilinen PVRP örnek kümesi üzerinde çalışılmıştır. Tablo 4.3, Cordeau vd. (1997) tarafından derlenmiş olan örnekler üzerinde geliştirmiş olduğumuz alt sınır algoritmasının,

Tablo 4.1: Küçük problemler - 5 müşterili

Deney Numarası	Müşteri Sayısı	CPU Süresi (saniye)	Alt Sınır	Üst Sınır	Fark* (%)
1	5	1800	2747	2850	4
2	5	38	1749	1749	0
3	5	180	1482	1482	0
4	5	8	1917	1917	0
5	5	8	1342	1342	0
6	5	11	2772	2772	0
7	5	134	1621	1621	0
8	5	1800	1864	2070	10
9	5	130	1498	1498	0
10	5	769	1797	1797	0

$$\text{Fark*} = 100 \times \frac{\text{Üst Sınır} - \text{Alt Sınır}}{\text{Alt Sınır}}$$

Tablo 4.2: Küçük Problemler - 20 müşterili

Deney Numarası	Müşteri Sayısı	CPU Süresi (saniye)	Alt Sınır	Üst Sınır	Fark (%)
1	20	1800	3471	8089	58
2	20	1800	2508	6708	60
3	20	1800	2536	7466	66
4	20	1800	5915	9787	42
5	20	1800	2742	6443	59
6	20	1800	3170	5745	46
7	20	1800	3167	6992	55
8	20	1800	4088	7900	47
9	20	1800	3101	7209	55
10	20	1800	2340	5910	59

Tablo 4.3: Örnek Problem Bilgileri ve Alt Sınır Performansları

Ornek	Musteri	Periyor	Arac	Alt Snr		En İyi	iyi sonuc		Fark	
				LB1	LP G.		LB1	LP G.	LB1	LP G.
p01	51	2	3	390	374	524	1	-	25,57	28,63
p02	50	5	3	1025	874	1322	1	-	22,47	33,89
p03	50	5	1	389	379	524	1	-	25,76	27,67
p04	75	2	5	490,5	479	835	1	-	41,26	42,63
p05	75	5	6	1233,5	1110	2027	1	-	39,15	45,24
p06	75	10	1	490,5	479	836	1	-	41,33	42,70
p07	100	2	4	573	570	826	1	-	30,63	30,99
p08	100	5	5	1405	1285	2034	1	-	30,92	36,82
p09	100	8	1	573	570	826	1	-	30,63	30,99
p10	100	5	4	1144	1054,5	1595	1	-	28,28	33,89
p11	139	5	4	385,5	-	779	1	-	50,51	-
p12	163	5	3	442	421	1195	1	-	63,01	64,77
p13	417	7	9	787	-	3511	1	-	77,58	-
p14	20	4	2	620	672	954	-	1	35,01	29,56
p15	38	4	2	1406	1496	1862	-	1	24,49	19,66
p16	56	4	2	2350	2388	2875	-	1	18,26	16,94
p17	40	4	4	976	960	1597	1	-	38,89	39,89
p18	76	4	4	2084	2144	3147	-	1	33,78	31,87
p19	112	4	4	3608	3680	4834	-	1	25,36	23,87
p20	184	4	4	6736	6936	8367	-	1	19,49	17,10
p21	60	4	6	1356	1152	2184	1	-	37,91	47,25
p22	114	4	6	2724	2544	4271	-	1	36,22	40,44
p23	168	4	6	4434	4368	6602	1	-	32,84	33,84
p24	51	6	3	2163	2496	3687	-	1	41,33	32,30
p25	51	6	3	2163	2496	3777	-	1	42,73	33,92
p26	51	6	3	2163	2496	3795	-	1	43,00	34,23
p27	102	6	6	12837	14994	21956	-	1	41,53	31,71
p28	102	6	6	12837	14994	22934	-	1	44,03	34,62
p29	102	6	6	12837	14994	22909	-	1	43,97	34,55
p30	153	6	9	42792	50860	75016	-	1	42,96	32,20
p31	153	6	9	42792	50860	78179	-	1	45,26	34,94
p32	153	6	9	42792	50860	80479	-	1	46,83	36,80
pr01	48	4	2	1750	1380	2209	1	-	20,78	37,53
pr02	96	4	4	2788	2212	3799	1	-	26,61	41,77
pr03	144	4	6	3640	3028	5218	1	-	30,24	41,97
pr04	192	4	8	3919,5	3355	6012	1	-	34,81	44,19
pr05	240	4	10	4243,5	3479	6769	1	-	37,31	48,60
pr06	288	4	12	4843,5	4196	8422	1	-	42,49	50,18
pr07	72	6	3	3759,5	3399	4997	1	-	24,76	31,98
pr08	144	6	6	4677	3864,5	7094	1	-	34,07	45,52
pr09	216	6	9	6743,5	5488,5	10370	1	-	34,97	47,07
pr10	288	6	12	7450	5976	13370	1	-	44,28	55,30
							26	16	36,46	36,70

$$*Fark = 100 \times \frac{En\ iyi - alt\ sınır}{En\ iyi}$$

uygulanan LP gevşetme yönteminin sonuçları ve bu örnekler için literatürde raporlanmış en iyi sonuçlar gösterilmiştir. Elde edilen değerlere göre geliştirmiş olduğumuz algoritma ile LP gevşetme yöntemi birbirlerine karşı üstünlük sağlayamamıştır. Bir sonraki alt bölümde, geliştirdiğimiz kurucu alt sınır, dal-kesi algoritması ve üst sınır algoritmalarının PARP-2ZT üzerindeki sonuçları raporlanmıştır.

4.4.3 Sayısal Sonuçları

Bu bölümde yapılan testler sonucunda elde edilen sonuçlar sunulacaktır. Algoritmalar C++ programlama dili ile geliştirilmiştir. Alt sınır algoritmaları matematiksel model çözümü için IBM ILOG CPLEX çağrılabilir kütüphaneleri ve sağlanmayan eşitsizliklerin bazılarını bulmak için Lysgaard vd. (2004) kapsamında geliştirilen CVRPSEP kütüphanelerini kullanmaktadır. Testler, Intel(2) Core(TM) i5-52000 2.20 GHz işlemci ve 8,00 GB Ram içeren bir bilgisayarda gerçekleştirilmiştir.

Öncelikli olarak kullanılan eşitsizliklerin kök düğümdeki etkileri incelenmiştir. Tablo 4.4 dört farklı örnek üzerinde her eşitsizliğin tek başına başarımını ve birlikte kullanıldıkları durumda kaç kısıt eklendiğini raporlamaktadır. Her örnek ve her eşitsizlik için A B sütunları yüzde değerleri raporlamaktadır. A sütunları eklenen kesilerin eşitsizlik çeşitleri arasındaki dağılımını göstermektedir ve toplamları %100 değerini verir. B sütunları ise her çeşit eşitsizliğin başarımını raporlamaktadır. Kök düğümde, hiçbir eşitsizlik kullanılmadan elde edilen doğrusal gevşetme değeri LP_0 , beş eşitsizlik birlikte kullanıldığı durumda elde edilen doğrusal gevşetme sonucu LP_5 ve eşitsizlik $i \in \{Kapasite, \text{Çizelgeli kapasite}, \text{Çizelgeli alt tur}, \text{Çizelge}, \text{Tek ziyaret}\}$ tek başına kullanıldığında elde edilen sonuç ise LP_i olsun. Eşitsizlik i için başarım $(LP_i - LP_0)/(LP_6 - LP_0)$ olarak hesaplanmıştır. Eşitsizlikler birlikte kullanıldıkları zaman daha sıkı alt sınır değeri vermektedirler ve başarım değerlerinin toplamı tanımları gereği %100 etmek zorunda değildir, toplam %100'ün altında veya üstünde olabilir.

Tüm örnekler için alt ve üst sınır algoritmaları ile elde edilen sonuçlar Tablo 4.5'de

Tablo 4.4: Eşitsizliklerin incelenmesi

Örnek	p14		p06		p07		p10	
Müşteri Sayısı	20		75		100		100	
Araç Sayısı	2		1		4		4	
Gün Sayısı	4		10		2		5	
	A	B	A	B	A	B	A	B
Kapasite	0,0	0,0	0,0	0,0	7,7	0,0	0,3	0,0
Çizelgeli kapasite	0,0	0,0	0,0	0,0	5,7	0,0	0,2	7,4
Çizelgeli alt tur	6,2	0,0	3,0	0,0	1,5	0,0	0,3	0,0
Çizelge	83,0	70,4	59,0	0,0	58,1	70,4	87,1	37,6
Tek ziyaret	10,8	15,9	38,0	100,0	27,0	15,9	12,1	46,7

gösterilmektedir. Tabloda kurucu alt sınır Alt sınır 1, dal-kesi algoritması ise Alt Sınır 2 sütunlarında gösterilmiştir. Dal-kesi algoritması, kurucu alt sınırdan ortalama olarak %12 daha iyi sonuç vermektedir. Aynı tabloda üç farklı üst sınır vardır. İlki literatürde bilinen en iyi PARP sonucu kullanılarak türetilmiştir. PARP sonucunda her gün için ziyaret edilecek müşteriler belirlenmiştir. Sipariş toplama aracı için girdi olan bu bilgi ile gün sayısı kadar süre sınırlı ARP çözümü sezgisel olarak bulunmuştur. Üst Sınır 1 sütunları altında iki araç tipi için oluşan maliyetler ve toplam maliyet raporlanmıştır. Elde edilen başlangıç çözümü üst Sınır 2 sütununda raporlanmıştır. Sezgisel yaklaşımın sonucu ise Üst Sınır 3 sütununda yer almaktadır. Görüleceği üzere tüm problemler için Üst Sınır 3 en düşük değerleri vermektedir. Alt sınır üst sınır arasındaki fark ortalama olarak %76'dır. Ancak proje kapsamında geliştirilen sezgisel yaklaşım, PARP için bilinen en iyi sonuç kullanılarak elde edilen üst sınırdan ortalama olarak %33 daha iyi sonuç vermektedir.

Tablo 4.5: PARP-2ZT Alt ve Üst Sınır Algoritmaları

Örnek	Müşteri Sayısı	Araç Sayısı	Gün Sayısı	Alt Sınır 1	Alt Sınır 2	Alt Sınır	Üst Sınır 1		Üst Sınır 2	Üst Sınır 3	Üst Sınır	Alt Sınırlar Arası Fark (%)	Üst Sınır ve Alt Sınır Arası Fark (%)
							Sipariş Toplama	Nakliye					
p01	51	3	2	794,3	1031,0	1031,0	1093,3	524,6	1617,9	1320,9	1108,9	29,8	7,6
p02	50	3	5	1896,1	2180,7	2189,7	2718,0	1322,9	4040,9	3175,9	2835,1	15,5	29,5
p03	50	1	5	817,1	866,4	866,4	1093,0	524,6	1617,6	1354,5	1085,2	6,0	25,2
p04	75	5	2	1077,7	1284,4	1284,4	1701,5	835,4	2537,0	2126,8	1834,6	19,2	42,8
p05	75	6	5	2474,6	2783,1	2783,1	4143,6	2028,0	6171,6	4624,0	4306,8	12,5	54,7
p06	75	1	10	1033,3	1135,4	1135,4	1680,8	836,4	2517,2	2072,6	2061,4	9,9	81,6
p07	100	4	2	1246,4	1452,1	1452,1	1808,9	826,1	2635,0	2401,8	1952,4	16,5	34,4
p08	100	5	5	2739,4	3144,1	3144,1	4517,2	2034,2	6551,4	5507,0	5254,8	14,8	67,1
p09	100	1	8	-	1317,4	1317,4	1663,5	826,1	2489,6	2381,5	1810,4	-	37,4
p10	100	4	5	2241,6	2517,9	2517,9	3773,1	1595,8	5369,0	4801,3	4382,9	12,3	74,1
p11	139	4	5	-	963,2	963,2	1798,3	779,3	2577,6	-	2240,5	-	132,6
p12	163	3	5	962,4	1267,5	1267,5	1799,3	779,3	2577,6	3224,4	3054,5	31,7	595,9
p13	417	9	7	1612,4	-	1612,4	-	3511,6	-	11308,4	11221,3	-	141,0
p14	20	2	4	1552,9	2047,2	2047,2	1923,9	954,8	2878,7	2916,6	2722,7	31,8	33,0
p15	38	2	4	3240,7	3397,4	3397,4	3722,5	1862,6	5585,2	4099,2	3727,6	4,8	9,7
p16	56	2	4	5206,5	5233,1	5233,1	5923,6	2875,2	8798,8	7168,4	6153,1	0,5	17,6
p17	40	4	4	2226,1	2470,6	2470,6	3211,4	1597,8	4809,2	3397,5	3284,3	11,0	32,9
p18	76	4	4	4564,1	4811,5	4811,5	6488,3	3147,2	9635,6	7623,0	7120,6	5,4	48,0
p19	112	4	4	7542,0	7838,0	7838,0	11853,8	4834,3	16688,1	14122,6	12055,9	3,9	53,8
p20	184	4	4	14495,4	14530,0	14530,0	25999,1	8367,4	34366,5	32949,3	30433,8	0,2	109,5
p21	60	6	4	2704,9	3346,2	3346,2	4333,7	2184,0	6517,7	4569,5	4454,2	23,7	33,1
p22	114	6	4	5506,5	6054,7	6054,7	9398,6	4271,1	13669,7	11002,0	10443,2	10,0	72,5
p23	168	6	4	9027,7	9478,6	9478,6	16651,3	6602,6	23253,9	21427,2	18859,8	5,0	99,0
p24	51	3	6	5266,1	6186,2	6186,2	7208,3	3687,5	10895,7	7877,6	7474,2	17,5	20,8
p25	51	3	6	5275,4	6302,1	6302,1	7340,2	3777,2	11117,3	8069,4	7544,2	19,5	19,7
p26	51	3	6	5286,2	6337,6	6337,6	7457,4	3795,3	11252,7	8601,5	7539,9	19,9	19,0
p27	102	6	6	31585,8	33254,7	33254,7	43915,6	21956,5	65872,1	53181,8	47120,1	5,3	41,7
p28	102	6	6	31684,2	33338,5	33338,5	44420,4	22934,7	67355,1	51920,4	47145,8	5,2	41,4
p29	102	6	6	31684,2	33363,0	33363,0	44082,2	22909,4	66991,6	54307,0	48567,0	5,9	44,7
p30	153	9	6	108286,1	109669,3	109669,3	150170,5	75016,6	225187,0	179875,8	169404,0	1,3	54,5
p31	153	9	6	113941,7	122624,2	122624,2	-	78179,9	-	177040,3	177040,3	7,6	44,4
p32	153	9	6	107087,4	110110,3	110110,3	156792,3	80479,2	237271,5	186477,2	172114,7	2,8	56,3
pr01	48	2	4	3423,6	3696,3	3696,3	4461,7	2209,0	6670,7	5012,6	4597,7	8,0	24,4
pr02	96	4	4	5722,1	6229,4	6229,4	8484,5	3799,3	12283,7	9698,3	8800,8	8,9	41,3
pr03	144	6	4	6908,3	7537,0	7537,0	12437,4	5218,1	17655,5	15789,8	14103,0	9,1	87,0
pr04	192	8	4	7884,7	8445,5	8445,5	15227,9	6012,8	21240,7	19629,7	18251,1	7,1	116,1
pr05	240	10	4	7851,7	8754,6	8754,6	17560,8	6769,8	24330,6	22845,0	20051,9	11,5	129,0
pr06	288	12	4	9558,2	10796,7	10796,7	22419,9	8422,6	30842,5	29956,0	26164,3	13,0	142,3
pr07	72	3	6	7298,4	7673,8	7673,8	10887,8	4997,4	15885,2	12025,9	11449,7	5,1	49,2
pr08	144	6	6	8496,5	9490,7	9490,7	17195,7	7094,5	24290,2	-	19886,9	11,7	109,5
pr09	216	9	6	12100,2	13939,7	13939,7	27709,6	10370,5	38080,1	-	33815,7	15,2	142,6
pr10	288	12	6	13388,5	15472,1	15472,1	35077,5	13370,0	48447,5	180050,0	42948,1	15,6	177,6
											Ortalama	11,7	76,1

Bölüm 5

Periyodik Araç Rotalama Problemi - Ziyaret Sıklığı

Bu bölümde Periyodik Araç Rotalama Problemi - Ziyaret Sıklığı (PARP-ZS) üzerinde çalışılmıştır. Alt bölüm 5.1'de problemin tanımı yapılarak matematiksel programlama modeli geliştirilmiş ve hesaplama karmaşıklığı sunulmuştur. Alt bölüm 5.2'de problem için geliştirilen üst sınır algoritması açıklanmış ve veri kümesi için gerekli güncellemelerden bahsedilmiştir. Alt bölüm 5.3 geliştirilen sezgisel yöntemi açıklamaktadır. Geliştirilen üst sınır ve sezgisel algoritmalarının performansları Alt bölüm 5.4 içerisinde raporlanmıştır.

5.1 Problem Tanımı

Bu problemde, Periyodik Araç Rotalama Problemi kararlarına ek olarak müşterilerin ziyaret sıklığının müşteri talebini değiştirmesi durumunu incelemekteyiz. PARP-ZS matematiksel modelinde müşteri ziyaret sıklığı (frekans) bir karar değişkeni olup, ziyaret sıklığını artırmanın müşteri talebini azalan bir hızda (konkav) artırdığı varsayımı altında çalışmaktayız. Müşteriye yapılan ziyaretlerin sıklığını artırmanın talebi ve dolayısıyla geliri artırmasının nedenleri başlıca;

- iki ziyaret arasında olası stok tükenme ihtimalini azaltmak ve tüketicinin farklı bir satış noktasına gitmek istemediği için alımdan vazgeçmesi ya da farklı markalara yönelmesi durumunu engellemek
- rafların sürekli dolu olması sayesinde ürünün tüketicinin gözüne çarpması ve tercih edilme ihtimalinin artmasını sağlamak
- rakiplerden daha önce müşteri ile buluşulacağından, müşterinin elindeki nakit ile daha fazla sipariş verme ihtimalini (cash catch-up) artırmaktır.

Ziyaret sıklığını artırmanın müşteri talebini artırdığı varsayımı, problemin hem özgün olmasını sağlamakta hem de daha gerçekçi bir yapıya sokmaktadır. PARP-ZS'nin gerçek hayatta benzerlerinden biri, Coca Cola içecek firması operasyonlarında görülmektedir. Önerilen problemin gerçekçi yapısını incelemek adına, firmadan talep edilen gerçekleşen planlı ziyaret ve satış verisi incelenmiş, 2013 yılında haftada 1 defa ziyaret edilirken, 2014'te ziyaret sıklıkları haftada 1'den 2 ye çıkarılan müşterilerin taleplerinde ortalama %25 artış gözlemlenmiştir. Bu artış firmaya ait meyve suyu, su, enerji içeceği gibi pazar payı düşük ürün kategorisinde daha belirgindir.

Coca Cola İçecek firması küçük market ve bakkallardan oluşan müşteri grubu için dağıtımını planlarken bu müşterilerin her biri için talep seviyelerini temel alarak bir ziyaret sıklığı belirlemektedir. Müşteriler önce siparişleri alan küçük araçlar (preseller) tarafından ziyaret edilmekte, ertesi gün sipariş edilen ürünlerin dağıtımı kamyonlar kullanılarak gerçekleştirilmektedir. Bu ziyaretler her bir müşteri için belirlenen sıklıkta ve günlerde gerçekleşmektedir. Bu modelde temel amaç, müşteri ziyaret sıklığının müşteri talebini değiştirmesi durumu üzerine yoğunlaşmak olduğu için, yalnızca fiziksel kapasite kısıtı olan tek tip araç bulunmakta ve siparişlerin telefonla alındığı, dağıtımı gerçekleştiren çalışanların müşterilerin raflarında ürünün görünürlüğünü artıracak çalışmalar yaptıkları varsayılmaktadır. Bu bölümde PARP üzerine eklenen parametreler, karar değişkenleri ve geliştirilen matematiksel model sunulmaktadır.

Kümeler

S_i : i müşterisinin (ziyaret sıklığı en az f_i olan) olurlu çizelgeler kümesi

Parametreler

dd_i : i müşterisinin haftada bir defa ziyaret edilmesi durumundaki günlük talebi

f_i : i müşterisinin ziyaret sıklığı için belirlenen alt limit

β_s : s çizelgesinin ziyaret sıklığı dolayısıyla toplam talebe getirisi

γ_s : ziyaret çizelgesi s için toplam ziyaret günü sayısı

α : birim fiyat

b_{is} : i müşterisinin s ziyaret çizelgesine atanma durumunda, yapılan ziyaret başına talebi

$b_{is} = \lceil T \times dd_i \times \beta_s / \gamma_s \rceil$ şeklinde hesaplanır.

R_{is} : i müşterisinin s ziyaret çizelgesi kullanıldığı durumda haftalık geliri

$R_{is} = b_{is} \times \alpha \times \gamma_s$ şeklinde hesaplanır.

Karar Değişkenleri

d_{it} : i müşterisine t gününde teslim edilen miktar

Geliştirilen model şu şekildedir;

PARP-ZS

$$Max \sum_{i \in N} \sum_{s \in S_i} R_{is} y_{is} - \sum_{i \in N_0} \sum_{j \in N_0} \sum_{t=1}^T c_{ij} x_{ijt} \quad (5.1)$$

Öyle ki;

$$\sum_{s \in S_i} y_{is} = 1 \quad i \in N \quad (5.2)$$

$$\sum_{j \in N_0 \setminus \{i\}} x_{ijt} = \sum_{s \in S_i} a_{st} y_{is} \quad i \in N, t = 1, \dots, T \quad (5.3)$$

$$\sum_{j \in N_0 \setminus \{i\}} x_{jit} = \sum_{s \in S_i} a_{st} y_{is} \quad i \in N, t = 1, \dots, T \quad (5.4)$$

$$\sum_{i \in N} x_{0it} \leq m \quad t = 1, \dots, T \quad (5.5)$$

$$d_{it} = \sum_{s \in S_i} b_{is} a_{st} y_{is} \quad i \in N, t = 1, \dots, T \quad (5.6)$$

$$d_{it} \leq u_{it} \leq q \quad i \in N, t = 1, \dots, T \quad (5.7)$$

$$u_{it} - u_{jt} + q x_{ijt} \leq q - d_{jt} \quad i, j \in N, t = 1, \dots, T \quad (5.8)$$

$$u_{it} \geq 0 \quad (5.9)$$

$$x_{ijt} \in \{0, 1\} \quad (5.10)$$

$$y_{is} \in \{0, 1\} \quad (5.11)$$

Amaç fonksiyonunun (5.1) ilk kısmı toplam geliri, ikinci kısmı ise toplam taşıma maliyetlerini hesaplamakta, yani (5.1) toplam karı en çoklamayı hedeflemektedir. Kısıt kümesi (5.2) her müşteriye bir ziyaret kombinasyonu atanmasını sağlamaktadır. Kısıt kümeleri (5.3) ve (5.4) her müşteri için t gününde eğer ziyaret edilecekse tek bir aracın gelmesini ve tek bir aracın ayrılmasını garanti altına almaktadır. Kısıt kümesi (5.5) her gün için depodan çıkan araç sayısının toplam araç sayısını aşmasını önlemektedir. Kısıt kümesi (5.6) i müşterisinin ziyaret edileceği günlerdeki talebini belirlemektedir. Kısıt kümeleri (5.7) ve (5.8) araçların

alt tur (subtour) yapmasını, yani belirli müşteriler arasında bir döngü içerisine girmesini engellemekte, ayrıca araca yüklenen yüklerin araç kapasitesini aşmamasını sağlamaktadır.

5.1.1 Hesaplama Karmaşıklığı

Teorem 2. *PARP-ZS NP-Zordur.*

Kanıt. Modelde bir parametre olan f_i , i müşterisinin ziyaret sıklığı için belirlenen alt limittir. Müşteri ziyaret sıklığı alt limiti, belirlenen periyottaki toplam gün sayısına (T) eşitlenerek bir örnek problem oluşturulabilir. Bu durumda PARP-ZS, ayrı ayrı çözüldüğünde aynı en iyi sonucu verecek, T adet tek günlük Araç Rotalama Problemi'ne dönüştürülebilir. ARP matematiksel modelinin NP-Zor olduğu bilinmektedir. Buradan yola çıkılarak PARP-ZS matematiksel modelinin özel bir durumunun NP-Zor olduğu, bu sebeple PARP-ZS probleminin de NP-Zor olduğu kanıtlanmaktadır. \square

5.2 Üst Sınır

PARP-ZS problemi için üst sınır çalışmalar birinci bölümde belirtildiği gibi PARP-2ZT için de aynı şekilde uygulanmış, 4 farklı yöntem sonuçları karşılaştırılmıştır. Tüm üst sınır çalışmalarının test sonuçları karşılaştırıldığında PARP-ZS için de en iyi üst sınırın kesi ekleme algoritması ile elde edildiği görülmüştür.

Bu bölümde kullanılan örnek problemlere yapılan eklemeler ve sayısal sonuçlar sunulacaktır.

5.2.1 Örnek Problemlere Eklemeler

Literatürdeki problemler genelde maliyeti en azlamayı amaçlamaktadır ve test sonuçları toplam maliyeti gösterir. PARP-ZS'de amaç karı en büyükmektir. Üst sınır algoritması sonuçlarını literatürde bilinen en iyi sonuçlarla karşılaştırabilmek için Cordeau vd. (1997) tarafından derlenen örnek problemlere eklemeler yapılarak kullanılmıştır.

- Her müşteri için toplam talep;

$$\text{Toplam talep} = \text{Ziyaret başına düşen talep} \times \text{Frekans}$$

şeklinde hesaplanmıştır.

- Negatif amaç fonksiyonu değerlerini ortadan kaldırmak için her örnek problem için farklı birim fiyatlar;

$$\text{Birim fiyat} = \lceil \text{Bilinen en iyi sonuç (toplam mesafe)} / \text{Toplam talep} \rceil$$

şeklinde hesaplanmıştır.

- Bilinen en iyi sonuçlar için toplam gelir her gün ziyaret edilen müşteri listesinden müşteri talepleri ve birim fiyatlar kullanılarak hesaplanmıştır.

- PARP-ZS de kullanılmak üzere ana talep;

$$\text{Ana talep} = \text{Toplam talep} / \text{Örnek problemde müşteriye atanmış frekansın etkisi}$$

şeklinde hesaplanmıştır.

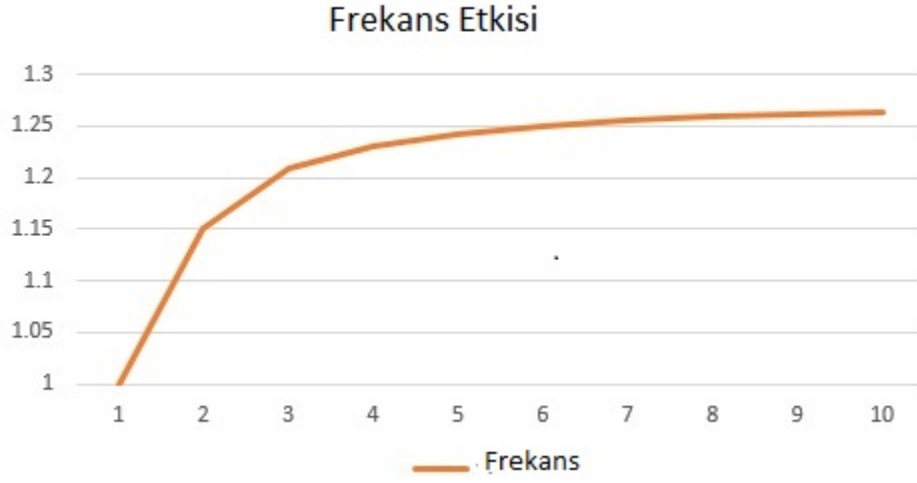
Ana talep örnek problemlerdeki atanmış frekansın etkisine bölünerek daha sonra

PARP-ZS'de karar verilen frekansın etkisiyle çarpılmaktadır. Örnek problemdeki frekanslar minimum frekans olarak kabul edildiği için ana talep en az toplam talebe eşit olacaktır. Örneğin, problemde ziyaret sıklığı 2 olan toplam talebi 100 olan bir müşteri için ana talep, Şekil 5.1'de belirtildiği gibi 2 gün ziyaret'e karşılık gelen 1,15' e bölünerek hesaplanır. Örneğin bu müşterinin model tarafından 6 kere ziyaret edilmesine karar verilirse, ana talep 1,25 ile çarpılıp en yakın tamsayıya yuvarlanarak toplam talep (satış miktarı) 108 olarak kabul edilir.

- Ziyaret çizelgesi seçenekleri 1 gün ile sınırlı olan örnek problemler alternatif ziyaret çizelgesi seçenekleriyle zenginleştirilmiştir.
- Örnek problemlerde planlama periyotları en az 2 en fazla 10 gündür. Bu aralıkta tüm ziyaret sıklığı sayıları için talebe getiri katsayıları, birinci gelişme raporunda sunulan Coca Cola planlı ziyaret sayısı ve toplam satış verisinden yola çıkılarak belirlenmiştir. Bu veri ziyaret sıklığı 1 den 2 ye çıkarılan müşteriler için satış miktarında %25 artışı içermektedir. Bu projede marketteki büyüme etkisi de göz önünde bulundurularak bu katsayı 1,15 olarak kabul edilmiş ve frekansa göre azalan şekilde artan biçimde belirlenmiştir. Belirlenen frekanslara karşılık gelen talebe getiri katsayıları Şekil 5.1 olarak verilmiştir.

5.2.2 Dal-Kesi Algoritması

PARP-ZS probleminde üst sınır belirlemek için dal-kesi (branch-and-cut) tabanlı bir algoritma geliştirilmiştir. Bu algoritma içerisinde PARP-2ZT problemi alt sınır algoritmasında da ele alınan Lysgaard vd. (2004) tarafından hazırlanmış kütüphaneden kapasite eşitsizlikleri kullanılmıştır. İki model arasındaki eşitsizlik kullanımındaki tek fark PARP-ZS'de, müşteriye ait olası en düşük talebe göre bir eşitsizlik yazılmasıdır. Bu şekilde yapılmasının sebebi, önceden hangi çizelgenin seçileceği ve dolayısıyla müşterinin talebinin de bilinmesidir.



Şekil 5.1: Ziyaret sıklığının talebe getirisi

PARP-ZS için geliştirilen dal-kesi algoritması Algoritma 7’de sunulmuştur. Bu algoritmada kapasite eşitsizliklerinin olmadığı model ele alınıp doğrusal gevşetimi çözülerek bir kök düğüm elde edilmiştir. Algoritma bu düğüm ile başlamaktadır ve üst sınır, kök düğümün değeri olarak belirlenir. Bu algoritmada derinlik öncelik (depth first) yaklaşımı kullanılmaktadır. Her iterasyonda yeni düğümler oluşturulmaktadır. Her yeni düğüm eklendikten sonra sağlanmayan kapasite eşitsizlikleri olup olmadığı Lysgaard vd. (2004) tarafından hazırlanan sezgisel yardımı ile kontrol edilir. Eğer sağlanmayan eşitsizlik bulunursa modele eklenir ve model tekrar çözülür. Daha sonra bu düğüm, ziyaret edilecek düğüm kümesinden çıkartılır. Her yeni düğüm sonrasında eldeki çözümün tam sayılı olup olmadığı kontrol edilir. Çözüm tam sayılı ve bilinen en iyi alt sınırdan daha düşük ise alt sınır güncellenir. Algoritma, tüm düğümler gezildikten sonra veya verilen süre limitine ulaşıldığında sona ermektedir ve

böylece alt sınır üst sınıra eşitlenerek üst sınır elde edilmektedir.

Girdi:

PARP – ZS modeli

Kök düğüm

Çıktı:

Alt ve üst sınır değerleri

Başla

┌ Kök düğüm değerini üst sınıra eşitle

└ *Düğüm kümesi boş olmadığı veya süre limiti aşılmadığında sürece yap*

┌ Düğüm kümesindeki en son oluşturulan düğümü seç

└ Seçilen düğümü, düğüm kümesinden çıkart

└ Seçilen düğümde *PARP – ZS* modelini çöz

└ *Sağlanmayan eşitsizlik bulunduğu sürece yap*

┌ Eşitsizliği *PARP – ZS* modeline ekle

└ *PARP – ZS* modelini çöz

└ **Eğer tüm karar değişkenleri tamsayılı ise**

┌ **Eğer sonuç alt sınırdan iyi ise**

└ alt sınırı güncelle

└ **Aksi takdirde**

┌ Tam sayılı olmayan bir karar değişkeni seç

└ Yeni alt düğümler oluştur

└ Yeni düğümleri düğümler kümesine ekle

└ **Aksi takdirde**

┌ Düğüm kümesini boşalt

└ Alt sınırı üst sınıra eşitle

└ Üst sınır değerini raporla

└ Varsa alt sınır değerini raporla

Algoritma 7: Dal Kesi Algoritması

5.3 Sezgisel Yöntem

Alt bölüm 5.1.1'de belirtildiği gibi PARP-ZS NP-Zor bir problemidir. PARP-2ZT'de olduğu gibi bu problem için de bir sezgisel yöntem geliştirilmiştir. Bu bölümde literatürdeki benzer problemleri çözmek için kullanılan sezgisel yöntemlerden ve PARP-ZS için geliştirilen sezgisel yöntemden bahsedilecektir.

Periyodik Araç Rotalama Problemi türevlerini çözmek için literatürde bir çok sezgisel yöntem geliştirilmiştir. (Cordeau vd., 1997) başlangıç çözümünü oluşturmak için, müşteriler depo ile yaptıkları açığa göre artan şekilde sıralanmış, her müşteriye bir ziyaret kombinasyonu atanmış, her gün için rastgele bir müşteri seçilerek GENI sezgisel yöntemi ile rotaya eklenmiştir. Bu işlem araç kapasitesi ve sayısı aşılmadan yapılmıştır. GENI bir müşteri düğümünü diğer ikisinin arasına 4-opt kullanılarak en az maliyetli sonucu elde edecek şekilde ekleyerek bir tur oluşturur. Maksimum araç sayısına eşit olan sıradaki tur olursuz bir tur olabilir. Uygulanan Tabu Arama algoritması bu olursuz turlar için de çalışır. Sonuçlar, rotalama, fazla kapasite ve fazla süreyi içeren bir maliyet fonksiyonu üzerinden değerlendirilir. Komşuluk arama bir müşteriyi aynı gün içinde farklı bir rotaya yerleştirme ya da ziyaret kombinasyonu değişikliği ile yapılmaktadır. Eski rotalarından kaldırılan müşteriler tabu olarak adlandırılmış ve yeniden aynı rotalara yerleştirilmesi en iyi sonuçtan daha iyi bir olurlu sonuç elde edilene kadar yasaklanmıştır. Bulunan her olurlu sonuçta fazla kapasite ve fazla süre ceza faktörleri bir sayıya bölünerek azaltılmış, aksi durumda çarpılarak artırılmıştır. Uygulanan tabu algoritması ile elde edilen sonuçlar literatürde bilinen en iyi sonuçlarla karşılaştırıldığında p01-p32 örnek problemleri için 24 sonuç iyileştirilmiştir.

Periyodik araç rotalama problemleri çözümü için önerilen bir başka sezgisel yöntem ise Mourgaya ve Vanderbeck (2006) tarafından geliştirilen sütun türetimi bazlı bir sezgisel yöntemdir. Çözüm yöntemlerinin ilk adımı, aynı araç tarafından ziyaret edilecek müşteriler arasındaki Öklid mesafeyi en azlarken araçlar arasındaki iş yükünü dengeleyen bir taktiksel planlama modelidir.

Müşterileri araçlara atarken bölgelere ayırma yöntemini kullanmışlardır. Gerçek hayat

problemlerinde araç sürücülerini tecrübeli oldukları bölgelere atamak şirketler için faydalı olacağından, araç rotaları belirli bölgeler içinde sınırlanmıştır. Sonraki adımda bir Araç Rotalama Problemi çözülür. Başlangıç adımında iş yükü için bir üst sınır belirlenmiş, en iyileme için iyileştirme sezgisel yöntemi olarak LP bazlı yuvarlama sezgiseli kullanılmıştır. Yuvarlama algoritması uygulanırken, sütun türetimi sürecini dengede tutmak için olurlu bir çözümün varlığını garanti altına alan yapay sütunlar eklenmiştir. Çözüm yönteminin sonraki adımlarında bölgelere ayırma kararlarından sapma en azlanırken, ziyaret sıklığına bağlı tüm operasyonel kısıtları içeren kararlar en iyilenmiştir. Bu yöntemle elde edilen sonuçlar literatürdeki çalışmalarla karşılaştırıldığında bölgelere ayırma ve iş yükü dağılımını dengesini iyileştirmişlerdir. Periyodik Araç Rotalama Problemi örnek problemleri arasından Cordeau vd. (1997) çalışmasında sunulan p01-02-03-17-25-26 üzerinde yaptıkları testler sonucunda literatürde bilinen en iyi sonuçlar ile karşılaştırıldığında toplam maliyet ortalama %20 daha kötü sonuç verirken, sonuçlar bölgelere ayırma kriteri açısından %16, iş yükü dengesi kriteri açısından %70 iyileştirilmiştir.

Cacchiani vd. (2014) tarafından karışık tamsayı doğrusal programlama problemleri için melez bir sezgisel yöntemi önerilmiş, çözüm yöntemi Periyodik Araç Rotalama örnek problemleri üzerinde test edilmiştir. Çözüm yöntemleri sütun türetimi ile çözülen LP gevşetme bazlı küme kapsama benzeri tamsayı doğrusal programlama formülasyonuna dayanmaktadır. LP çözümü bir tamsayı çözümü elde edilmek için kullanılmıştır. Kaynak kısıtlı bir en kısa yol problemi (ESPRC) olan alt problem, yinelemeli yerel arama (ILS) algoritması ile çözülmektedir. ESPRC 2 müşteri düğümü arasındaki en kısa yolu bulmayı hedeflerken, başlangıç ve bitiş düğümleri deponun 2 kopyasıdır. Komşuluk yapıları; ekleme, çıkarma, taşıma, yerleştirme ve yer değiştirme operatörleridir.

Daha önce PARP-ZS problemine ziyaret sıklığının karar değişkeni olması açısından benzerliğinden bahsettiğimiz Francis vd. (2006) büyük problemlerin çözümünde kullanılmak üzere kesin çözüm yönteminin bir sezgisel varyasyonu önermişlerdir. Çözüm yöntemlerinin ilk adımında müşterilerin ziyaret programını belirleyen kararlardan, rotaları oluşturanları

ayırmışlar ve 2 alt problem oluşturmuşlardır. İlk alt problemde her müşteriye en uygun ziyaret programı ve araç atanmıştır. İkinci alt problemde ise rotalama kararları her araç-ziyaret programı ikilisi için verilmelidir. Ortaya çıkan problemin NP-Zor olduğu kanıtlanmış Kar Getiren Gezgin Satıcı Problemi'ne benzer olduğu için alt ve üst sınırlar geliştirilmiştir. İlk alt problem için en iyi sonuç elde edilmiştir fakat bu ikinci alt problem için alt tur engelleme kısıtlarının fazlalığından dolayı mümkün olmamıştır. Bu kısıtların sadece bazıları gerekli olduğu için hepsi gevşetilmiş, elde edilen sonuçta bir alt tur varsa gerekli olan kısıtlar eklenip olurlu bir sonuç elde edilene ya da durdurma kuralı devreye girene kadar tekrar çözülmüştür. Elde edilen sonucu rotalama alt problemi için alt sınır olarak kabul etmişler, ziyaret programı atama alt problemi optimal sonucu ile toplayarak amaç fonksiyonu için bir alt sınır elde etmişlerdir. Üst sınır ise ARP için tasarruf sezgisel yöntemi uygulayarak elde edilmiştir. Alt ve üst sınırlar eşit değil ise dal ve sınır algoritması uygulayarak, aralığın kapanmasını sağlamışlardır. Çözüm yöntemlerinin performansını ölçmek için standart PARP ile ziyaret sıklığının karar değişkeni olduğu modelleri karşılaştırılmış rotalama maliyetindeki %32 artışa rağmen daha sık gerçekleştirilen ziyaretlerin servis getirisi ile toplam amaç fonksiyonunda %55 gelişme elde etmişlerdir. Francis vd. (2006) tarafından geliştirilen model, talep büyüklüğü ziyaret sıklığı ilişkisi yönünden bu projede ele alınacak PARP-ZS problemi ile farklılık göstermektedir. Çözüm yönteminin yeterli büyüklükteki problemler için yavaş kalması nedeniyle de bu projede doğrudan faydalanılamayacaktır. Hemmelmayr vd. (2009) ve Vidal vd. (2013) tarafından uygulanan Değişken Komşuluk Arama sezgisel yöntemi Periyodik Araç Rotalama Problemleri için önerilen çözüm yöntemleri arasında bilinen en iyi sonuçları veren sezgisel yöntemdir.

PARP-ZS için geliştirilen DKA sezgiseli, PARP-2ZT sezgiseli ile büyük oranda benzerlik göstermektedir. Bu sebeple algoritmanın tamamı yerine PARP-2ZT'den olan farklılıkları raporlanmıştır. Bu farklılıklar şu şekilde sıralanabilir:

- PARP-2ZT sezgiseli için move, swap ve ziyaret kombinasyonu değişikliği komşuluk operatörleri vardır. PARP-ZS için de move ve swap operatörleri kullanılmıştır.

- PARP-2ZT’de iki çeşit araç vardır. PARP-ZS ise tek çeşit araç içerir. Bu sebeple sadece kamyonlar üzerinde komşuluk aramaları yapılmıştır.
- Ziyaret kombinasyonu değişikliği operatörü PARP-2ZT’de sadece aynı frekanslı ziyaret kombinasyonları arasında değişiklik yapmaktadır. Ancak PARP-ZS için bu operatör farklı frekansa sahip ziyaret kombinasyonları arasında da değişiklik yapabilir. Bu sayede bir müşterilerin ziyaret sıklıkları değişmektedir.

5.4 Deney Sonuçları

PARP-ZS için yapılan deneylerin sonuçları Tablo 5.1’de verilmiştir. PARP için bilinen en iyi sonucun geliri hesaplanmış ve bu değer PARP-ZS için geliştirilen sezgiselin önerdiği sonuç ile karşılaştırılmıştır. Nakliye sütunu PARP için bilinen en iyi sonuç değerini, Birim Fiyat sütunu teslim edilen her ürün için elde edilen getiriyi, Talep sütunu toplam talebi göstermektedir. Talep ve birim maliyet çarpılarak Gelir sütunu hesaplanmıştır. Alt Sınır 1 sütunu gelir ve nakliye sütunları arasındaki farkı göstermektedir. Alt Sınır 2 sütunu ise sezgisel sonucu gösterir. Fark sütunu ise iki alt sınır arasındaki farkı göstermektedir ve proje kapsamında geliştirilen sezgisel PARP için bilinen en iyi sonuçtan ortalama olarak %0,8 daha iyi bir gelir elde edilmesini sağlamaktadır. PARP-ZS için geliştirilen üst sınır sonuçlarında gözlemlenen çeşitli sıkıntılar sebebi ile üst sınır değerleri raporlanmamıştır ve bu konudaki çalışmaların proje sonrasında da devam etmesi planlanmaktadır.

Tablo 5.1: Alt Sınır Algoritmaları Sayısal Sonuçları

Örnek	Nakliye	Birim Fiyat	Talep	Gelir	Alt Sınır 1	Alt Sınır 2	Fark (%)
p01	524,61	9,49	937	8892,1	8367,5	8303,2	-0,8
p02	1322,87	9,2	2133	19623,6	18300,7	18156,0	-0,8
p03	524,61	9,62	777	7474,7	6950,1	6799,9	-2,2
p04	835,43	8,12	1364	11075,7	10240,3	10146,2	-0,9
p05	2027,99	6,99	3555	24849,5	22821,5	22714,9	-0,5
p06	836,37	7,18	1364	9793,5	8957,2	8361,2	-6,7
p07	826,14	8,59	1458	12524,2	11698,1	11380,2	-2,7
p08	2034,15	6,89	4094	28207,7	26173,5	25434,4	-2,8
p09	826,14	7,79	1458	11357,8	10531,7	10057,2	-4,5
p10	1595,84	7,03	3216	22608,5	21012,6	20346,5	-3,2
p11	779,29	2,46	3459	8509,1	7729,9	7718,8	-0,1
p12	1195,88	6,6	1694	11180,4	9984,5	10167,1	1,8
p13	3511,62	0,12	116052	13926,2	10414,6	7623,4	-26,8
p14	954,81	147,16	152	22368,3	21413,5	22395,4	4,6
p15	1862,63	146,54	256	37514,2	35651,6	39588,9	11,0
p16	2875,24	161,22	360	58039,2	55164,0	56166,6	1,8
p17	1597,75	88,8	304	26995,2	25397,5	26482,8	4,3
p18	3147,24	103,77	512	53130,2	49983,0	54127,7	8,3
p19	4834,34	120,74	720	86932,8	82098,5	82817,0	0,9
p20	8367,4	142,44	1136	161811,8	153444,4	147617,3	-3,8
p21	2184,04	80,19	456	36566,6	34382,6	35617,1	3,6
p22	4271,11	87,05	768	66854,4	62583,3	66916,5	6,9
p23	6602,59	97,35	1080	105138,0	98535,4	97645,4	-0,9
p24	3687,46	197,66	324	64041,8	60354,4	65542,1	8,6
p25	3777,15	188,78	306	57766,7	53989,5	60958,3	12,9
p26	3795,33	178,68	324	57892,3	54097,0	58343,9	7,9
p27	21956,46	531,28	648	344269,4	322313,0	348105,0	8,0
p28	22934,71	499,33	612	305590,0	282655,3	321006,0	13,6
p29	22909,36	473,14	648	306594,7	283685,4	304586,5	7,4
p30	75016,58	1168,1	972	1135393,2	1060376,6	1121179,8	5,7
p31	78179,89	1075,13	1038	1115984,9	1037805,1	1045510,7	0,7
p32	80479,2	1034,82	972	1005845,0	925365,8	997057,7	7,7
pr01	2209,02	25,95	1301	33761,0	31551,9	33663,3	6,7
pr02	3799,28	21,4	2651	56731,4	52932,1	55058,4	4,0
pr03	5218,13	17,4	3952	68764,8	63546,7	62535,7	-1,6
pr04	6012,79	14,66	5259	77096,9	71084,2	68981,5	-3,0
pr05	6769,8	10,99	7225	79402,8	72633,0	69004,2	-5,0
pr06	8422,64	13,17	7502	98801,3	90378,7	86012,0	-4,8
pr07	4997,41	24,16	2890	69822,4	64825,0	65169,6	0,5
pr08	7094,52	13,72	6314	86628,1	79533,6	75225,5	-5,4
pr09	10370,45	14,85	8632	128185,2	117814,8	109689,2	-6,9
pr10	13370,04	11,56	12283	141991,5	128621,4	118112,9	-8,2
Ortalama							0,8

Bölüm 6

Periyodik Araç Rotalama Problemi - Birleşik

Bu bölümde Periyodik Araç Rotalama Problemi - Birleşik (PARP-B) üzerinde çalışılmıştır. Bu problem önceki iki bölümde tanımlanmış olan PARP-2ZT ve PARP-ZS problemlerin birleşimi olarak değerlendirilebilir. PARP-B her müşteri için ziyaret sıklığına karar vermekte ve her müşteriyi iki farklı tip araç ile birbirini takip eden periyotlarda ziyaret etmektedir.

Problemin tanımı Alt bölüm 6.1 içerisinde sunulmuştur. Alt bölüm 6.2'de problem için geliştirilen sezgisel açıklanmıştır. Geliştirilen sezgisel algoritmanın sonuçları Alt bölüm 6.3 içerisinde raporlanmıştır.

6.1 Problem Tanımı

Bu bölümde PARP-B'nin matematiksel programlama modeli sunulmuştur. Bu problem PARP-2ZT ve PARP-ZS kapsamında verilen kararların tamamını içermektedir. Toplam geliri en çoklamayı amaçlayan bu problemde hem iki tip araç yer almakta hem de müşterilerin ziyaret sıklığı değiştirilebilmektedir. Ziyaret sıklığının artırılması toplam talebi (azalan bir hızda olmak üzere) artırmaktadır. Ancak ziyaret süresinin her ziyaret için aynı kaldığı var-

sayılmıştır ve dur_i parametresi i müşterisine 1. tip araç ile yapılan her ziyaretin süresini göstermektedir. PARP-B matematiksel programlama modelinde şu şekildedir:

$$\begin{aligned} & \textbf{PARP-B} \\ \text{Max} \sum_{i \in N} \sum_{s \in S_i} R_{is} y_{is} - \sum_{i \in N_0} \sum_{j \in N_0} \sum_{v=1}^2 \sum_{t=1}^T c_{ij} x_{ijtv} \end{aligned} \quad (6.1)$$

Öyle ki;

$$\sum_{j \in N_0 \setminus \{i\}} x_{ijtv} = \sum_{s \in S_i} a_{stv} y_{is} \quad i \in N, t = 1, \dots, T, v \in \{1, 2\} \quad (6.2)$$

$$\sum_{j \in N_0 \setminus \{i\}} x_{jitiv} = \sum_{s \in S_i} a_{stv} y_{is} \quad i \in N, t = 1, \dots, T, v \in \{1, 2\} \quad (6.3)$$

$$\sum_{i \in N} x_{0itv} \leq m_v \quad t = 1, \dots, T, v \in \{1, 2\} \quad (6.4)$$

$$\sum_{s \in S_i} y_{is} = 1 \quad i \in N \quad (6.5)$$

$$d_{itv} = \sum_{s \in S_i} b_{is} a_{stv} y_{is} \quad i \in N, t = 1, \dots, T, v \in \{2\} \quad (6.6)$$

$$u_{itv} - u_{jtv} + q_v x_{ijtv} \leq q_v - d_{jtv} \quad i, j \in N, t = 1, \dots, T, v \in \{2\} \quad (6.7)$$

$$d_{itv} \leq u_{itv} \leq q_v \quad i \in N, t = 1, \dots, T, v \in \{2\} \quad (6.8)$$

$$d_{itv} = dur_i \quad i \in N, t = 1, \dots, T, v \in \{1\} \quad (6.9)$$

$$u_{itv} - u_{jtv} + (q_v + r_{ij}) x_{ijtv} \leq q_v - d_{jtv} \quad i, j \in N, t = 1, \dots, T, v \in \{1\} \quad (6.10)$$

$$d_{itv} \leq u_{itv} \leq q_v \quad i \in N, t = 1, \dots, T, v \in \{1\} \quad (6.11)$$

$$u_{itv}, d_{itv} \geq 0 \quad i \in N_0, \quad t = 1, \dots, T, v \in \{1, 2\} \quad (6.12)$$

$$x_{ijtv} \in \{0, 1\} \quad i, j \in N_0, \quad t = 1, \dots, T, v \in \{1, 2\} \quad (6.13)$$

$$y_{is} \in \{0, 1\} \quad i, j \in N, s \in S_i \quad (6.14)$$

Amaç fonksiyonu (6.1) toplam geliri (satışların toplam değeri ile rotalama maliyetleri arasındaki fark) ençoklamayı amaçlar. Kısıt kümeleri (6.2- 6.3) her müşteriye, seçilen çizelgesinin zorunlu kıldığı günlerde bir kez uğramayı, diğer günlerde ise uğramamayı garantiler.

Kısıt kümesi (6.4) her tip araç için üst limit tanımlar. Kısıt kümesi (6.5) ile her müşteriye bir çizelge atanır. Kısıt kümesi (6.6) müşteriye atanan çizelgeye göre her ziyarette teslim edilmesi gereken sipariş miktarını hesaplar. Kısıt kümeleri (6.7-6.8) alt turları ve fiziksel kapasitenin aşılmamasını 2. tip araç için garantiler. Kısıt kümesi (6.9) 1. tip araç için müşteride geçirilecek zamanı tanımlar. Kısıt kümeleri (6.10-6.11) 1. tip araçlar için alt turları engeller ve zaman kapasitenin aşılmamasını sağlar. Kısıt kümeleri (6.12-6.14) karar değişkenlerini tanımlar.

6.1.1 Hesaplama Karmaşıklığı

Teorem 3. *PARP-B NP-Zordur.*

Kanıt. Modelde S_i kümesi i müşterisini ziyaret etmek için kullanılacak çizelge kümesini göstermektedir. Bir müşterinin her gün ziyaret edilmesini gerektiren çizelgenin $cizelge_1$ olduğunu ve her müşteri için $S_i = \{cizelge_1\}$ olduğunu varsayalım. Bu durumda her müşteri her gün ziyaret edilmek durumundadır. Bu örnek durumda PARP-B, ayrı ayrı çözüldüğünde aynı en iyi sonucu verecek, $2T$ adet tek günlük Araç Rotalama Problemi'ne dönüştürülebilir, bu problemlerin yarısı süre yarısı zaman kapasiteli araçlar içerecektir. ARP matematiksel modelinin NP-Zor olduğu bilinmektedir. Buradan yola çıkılarak PARP-B matematiksel modelinin özel bir durumunun NP-Zor olduğu, bu sebeple PARP-B probleminin de NP-Zor olduğu kanıtlanmaktadır. \square

6.2 Sezgisel Yöntem

Projede ele aldığımız üçüncü problem PARP-B konusunda ilk iki problemin örneklerini harmanlayarak örnek problem setleri oluşturduk. Buna göre ürünün satış fiyatı ve ziyaret sıklığının talebe etkisi ikinci problemden gelirken ikinci araç tipi ile ilgili parametreleri ilk problemden edindik. Geliştirdiğimiz sezgisel yöntem de yine iki problemin sezgisel yön-

temlerinin harmanlanmasına dayanmaktadır. Buna göre yine iki tip araç için gün içindeki rotalarda müşterilerin yerlerini değiştiren yer değiştirme ve ekleme komşuluk yapıları ve yine frekans değişikliğini de içerecek şekilde çizelge değiştirme komşuluk yapısı bulunmaktadır. Yine değişen çizelgede müşterinin ziyaret sıklığı değişiyorsa talebi ve her seferde müşteriye götürülen yük miktarı da ona göre değiştirilir. Son olarak yerel arama için yine 2-opt yaklaşımı kullanılmaktadır.

6.3 Deney Sonuçları

PARP-B kapsamında yapılan deneyler Tablo 6.1’de sunulmuştur. Her örnek için bilinen en iyi sonuçtan bir alt sınır değeri türetilmiştir. Uygulanan yaklaşım PARP-2ZT ve PARP-B için kullanılan yaklaşımın aynısıdır. Öncelikle her gün için ayrı ayrı zaman kapasiteli PARP çözülerek sipariş toplama aracı için maliyet belirlenmiştir. Toplam gelir ve toplam maliyet arasındaki fark Alt Sınır 1 sütununda raporlanmıştır. Alt Sınır 2 sütunu ise geliştirilen sezgiselin sonucunu göstermektedir. Sezgisel kullanımı ile ortalama olarak %6,6 oranında bir gelir artışı olmuştur.

Tablo 6.1: PARP-B Deneş Sonuđları

Örnek	Sipariş Toplama	Nakliye	Birim Fiyat	Talep	Gelir	Alt Sınır 1	Alt Sınır 2	Park
p01	1093,25	524,61	9,49	937	8892,13	7274,27	7615,49	4,7
p02	2718,02	1322,87	9,2	2133	19623,6	15582,71	16483,1	5,8
p03	1093	524,61	9,62	777	7474,74	5857,13	6082,11	3,8
p04	1701,54	835,43	8,12	1364	11075,68	8538,71	8967,5	5,0
p05	4143,56	2027,99	6,99	3555	24849,45	18677,9	20088,54	7,6
p06	1680,8	836,37	7,18	1364	9793,52	7276,35	6715	-7,7
p07	1808,88	826,14	8,59	1458	12524,22	9889,2	10306,73	4,2
p08	4517,22	2034,15	6,89	4094	28207,66	21656,29	22462,14	3,7
p09	1663,48	826,14	7,79	1458	11357,82	8868,2	8724,08	-1,6
p10	3773,12	1595,84	7,03	3216	22608,48	17239,52	17832,69	3,4
p11	1798,28	779,29	2,46	3459	8509,14	5931,57	6302,48	6,3
p12	2779,41	1195,88	6,6	1694	11180,4	7205,11	7835,09	8,7
p13	-	3511,62	0,12	116052	13926,24	-	35,14	-
p14	1923,88	954,81	147,16	152	22368,32	19489,63	21299,36	9,3
p15	3722,52	1862,63	146,54	256	37514,24	31929,09	36967,24	15,8
p16	5923,6	2875,24	161,22	360	58039,2	49240,36	52303,77	6,2
p17	3211,44	1597,75	88,8	304	26995,2	22186,01	24396,04	10,0
p18	6488,34	3147,24	103,77	512	53130,24	43494,66	49060,92	12,8
p19	11853,76	4834,34	120,74	720	86932,8	70244,7	73427,34	4,5
p20	25999,14	8367,4	142,44	1136	161811,84	127445,3	128890,63	1,1
p21	4333,7	2184,04	80,19	456	36566,64	30048,9	32839,15	9,3
p22	9398,61	4271,11	87,05	768	66854,4	53184,68	58901,59	10,7
p23	16651,3	6602,59	97,35	1080	105138	81884,11	85623,06	4,6
p24	7208,26	3687,46	197,66	324	64041,84	53146,12	59218,08	11,4
p25	7340,18	3777,15	188,78	306	57766,68	46649,35	55574,26	19,1
p26	7457,35	3795,33	178,68	324	57892,32	46639,64	51958,15	11,4
p27	43915,64	21956,46	531,28	648	344269,44	278397,34	316804,28	13,8
p28	44420,42	22934,71	499,33	612	305589,96	238234,83	286477,03	20,2
p29	44082,24	22909,36	473,14	648	306594,72	239603,12	274151,56	14,4
p30	150170,46	75016,58	1168,1	972	1135393,2	910206,16	1016433,75	11,7
p31	-	78179,89	1075,13	1038	1115984,94	-	939716,81	-
p32	156792,32	80479,2	1034,82	972	1005845,04	768573,52	884277,25	15,1
pr01	4461,65	2209,02	25,95	1301	33760,95	27090,28	29852,26	10,2
pr02	8484,45	3799,28	21,4	2651	56731,4	44447,67	48776,25	9,7
pr03	12437,39	5218,13	17,4	3952	68764,8	51109,28	51116,02	0,0
pr04	15227,87	6012,79	14,66	5259	77096,94	55856,28	55395,16	-0,8
pr05	17560,82	6769,8	10,99	7225	79402,75	55072,13	55368,08	0,5
pr06	22419,9	8422,64	13,17	7502	98801,34	67958,8	66960,66	-1,5
pr07	10887,78	4997,41	24,16	2890	69822,4	53937,21	57589,54	6,8
pr08	17195,7	7094,52	13,72	6314	86628,08	62337,86	62114,89	-0,4
pr09	27709,62	10370,45	14,85	8632	128185,2	90105,13	87120,68	-3,3
pr10	35077,49	13370,04	11,56	12283	141991,48	93543,95	90790,44	-2,9
							Ortalama	6,6

Bölüm 7

Sonuç

Proje kapsamında periyodik araç rotalama problemi üzerinde çalışılmıştır. İlgili literatür detaylı şekilde taranmış, problem tanımları, kesin ve sezgisel çözüm yöntemleri ve örnek problem kümeleri incelenmiş ve raporlanmıştır.

Klasik periyodik araç rotalama probleminin varyantı yeni problemler tanımlanmış ve bu problemler için alt ve üst sınır algoritmaları geliştirilmiştir. Geliştirilen algoritmaların performanslarını test etmek üzere literatürde sıkça kullanılan örnek problem kümeleri kullanılmıştır. Bu kümelerdeki örnekler klasik problem için geçerlidir ve tanımlanan varyantlar için çeşitli özelleştirmeler proje kapsamında yapılmıştır. Bu örnekler üzerinde performans testleri yapılmıştır.

Sonuçlar özellikle literatürde periyodik araç rotalama problemi için bulunan en iyi sonuçların projede ele alınan problemlere uyarlanmış halleri ile karşılaştırılmıştır. Bu şekilde gösterilmiştir ki iki araç tipinin olduğu veya ziyaret sıklığının talebi etkileyen bir karar değişkeni olduğu durumlarda periyodik araç rotalama problemi yaklaşımı yeterli gelmemektedir.

Proje kapsamında, farklı dönemlerde olmak üzere, dört bursiyer görev yapmıştır. İki bursiyer, Okan Altinkök ve Arya Sevgen, proje esnasında yüksek lisans derecelerini almıştır. Projede görev yapan bursiyerlerden, Arya Sevgen ve Onur Çağırıcı, doktora programlarına devam etmektedir. Proje kapsamında yapılan çalışmalar çeşitli ulusal ve uluslararası kong-

relerde sunulmuştur. Proje ekibi proje sonuçlarını uluslararası hakemli dergilerde yayınlama konusu üzerinde halen çalışmaktadır.

Kaynakça

- Aksen, D., Kaya, O., Sibel Salman, F., ve Tüncel, O. (2014). An adaptive large neighborhood search algorithm for a selective and periodic inventory routing problem. *European Journal of Operational Research*, 239(2):413–426.
- Alegre, J., Laguna, M., ve Pacheco, J. (2007). Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts. *European Journal of Operational Research*, 179(3):736 – 746.
- Angelelli, E. ve Speranza, M. G. (2002). The application of a vehicle routing model to a waste-collection problem: two case studies. *Journal of the Operational Research Society*, 53:944–952.
- Baldacci, R., Bartolini, E., Mingozzi, A., ve Valletta, A. (2011). An exact algorithm for the period routing problem. *Operations Research*, 59(1):228–241.
- Baptista, S., Oliveira, R. C., ve Zúquete, E. (2002). A period vehicle routing case study. *European Journal of Operational Research*, 139:220–229.
- Beltrami, E. ve Bodin, L. (1974). Networks and vehicle routing for municipal waste collection. *Networks*, 4(1):65–94.
- Cacchiani, V., Hemmelmayr, V. C., ve Tricoire, F. (2014). A set-covering based heuristic algorithm for the periodic vehicle routing problem. *Discrete Applied Mathematics*, 163:53–64.

- Campbell, A. M. ve Wilson, J. H. (2014). Forty years of periodic vehicle routing. *Networks*, 63(1):2–15.
- Chao, I.-M., Golden, B. L., ve Wasil, E. (1995). An improved heuristic for the period vehicle routing problem. *Networks*, 26(6):25–44.
- Christofides, N. ve Beasley, J. E. (1984). The period routing problem. *Networks*, 14:237–256.
- Clarke, G. ve Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581.
- Coene, S., Arnout, A., ve Spieksma, F. C. R. (2010). On a periodic vehicle routing problem. *Journal of the Operational Research Society*, 61(12):1719–1728.
- Cook, T. M. ve Russell, R. A. (1978). A Simulation and Statistical Analysis of Stochastic Vehicle Routing With Timing Constraints. *Decision Sciences*, 9:673–687.
- Cordeau, J.-F., Gendreau, M., ve Laporte, G. (1997). A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30(2):105–119.
- Cordeau, J.-F., Laporte, G., ve Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *The Journal of the Operational Research Society*, 52(8):928–936.
- Croes, G. A. (1958). A method for solving traveling-salesman problems. *Operations Research*, 6(6):791–812.
- Dantzig, G., Fulkerson, R., ve Johnson, S. (1954). Solution of a large-scale traveling-salesman problem. *Operations Research*, 2:393–410.
- Dumas, Y., Desrosiers, J., ve Soumis, F. (1991). The pickup and delivery problem with time windows. *European Journal of Operational Research*, 54(1):7 – 22.

- Eilon, S., Watson-Gandy, C. D. T., ve Christofides, N. (1971). *Distribution management: mathematical modelling and practical analysis by Samuel Eilon, C. D. T. Watson-Gandy and Nicos Christofides*. Griffin London.
- Eksioglu, B., Vural, A. V., ve Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers and Industrial Engineering*, 57(4):1472 – 1483.
- Francis, P. ve Smilowitz, K. (2006). Modeling techniques for periodic vehicle routing problems. *Transportation Research Part B: Methodological*, 40(10):872 – 884.
- Francis, P., Smilowitz, K., ve Tzur, M. (2006). The period vehicle routing problem with service choice. *Transportation Science*, 40(4):439–454.
- Gendreau, M., Laporte, G., Musaraganyi, C., ve Taillard, E. D. (1999). A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Computers and Operations Research*, 26(12):1153 – 1173.
- Golden, B. L. ve Wasil, E. A. (1987). Computerized vehicle routing in the soft drink industry. *Operational Research*, 35(1):6–17.
- Hemmelmayr, V. C., Doerner, K. F., ve Hartl, R. F. (2009). A variable neighborhood search heuristic for periodic routing problems. *European Journal of Operational Research*, 195(3):791 – 802.
- Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science*, 43(4):408–416.
- Lysgaard, J., Letchford, A. N., ve Eglese, R. W. (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100(2):423–445.
- Mladenović, N. ve Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100.
- Mourgaya, M. ve Vanderbeck, F. (2006). The periodic vehicle routing problem: classification and heuristic. *RAIRO - Operations Research*, 40:169–194.

- Russell, R. ve Igo, W. (1979). An assignment routing problem. *Networks*, 9(1):1–17.
- Russell, R. A. ve Gribbin, D. (1991). A multiphase approach to the period routing problem. *Networks*, 21:747–765.
- Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research, Vol. 35, No. 2.*, 35:254–265.
- Toth, P. ve Vigo, D. (1999). A heuristic algorithm for the symmetric and asymmetric vehicle routing problems with backhauls. *European Journal of Operational Research*, 113(3):528 – 543.
- Vidal, T., Crainic, T. G., Gendreau, M., Lahrichi, N., ve Rei, W. (2012). A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operational Research*, 60(3):611–624.
- Vidal, T., Crainic, T. G., Gendreau, M., ve Prins, C. (2013). Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research*, 231(1):1–21.
- Wen, M., Cordeau, J.-F., Laporte, G., ve Larsen, J. (2010). The dynamic multi-period vehicle routing problem. *Computers and Operations Research.*, 37(9):1615–1623.

TÜBİTAK
PROJE ÖZET BİLGİ FORMU

Proje Yürütücüsü:	Yrd. Doç. Dr. AHMET CAMCI
Proje No:	213M425
Proje Başlığı:	Periyodik Araç Rotalama Probleminin Üç Farklı Türevi Üzerine
Proje Türü:	1001 - Araştırma
Proje Süresi:	30
Araştırmacılar:	NAİL ÖZGÜR ÖZPEYNİRCİ
Danışmanlar:	
Projenin Yürütüldüğü Kuruluş ve Adresi:	YAŞAR Ü. İKTİSADİ VE İDARİ BİLİMLER F. İŞLETME B.
Projenin Başlangıç ve Bitiş Tarihleri:	15/03/2014 - 15/09/2016
Onaylanan Bütçe:	142200.0
Harcanan Bütçe:	103129.68

TÜBİTAK

<p>Öz:</p>	<p>Araç rotalama problemi güncel bir problem olup ellili yıllardan beri çalışılmaktadır. Önceleri araç rotalama problemi karmaşık oluşu ve teknolojinin yeterli olmaması sebebiyle literatürde yaygın olarak ele alınamamıştır. Doksanlı yıllarda gelişen teknoloji ile araç rotalama problemleri çalışılabilir hale gelmiş ve araç rotalama problemi üzerine yapılan çalışmaların sayısı artmıştır Araç rotalama problemi bir veya daha fazla depodan çıkan araçların tüm müşterileri ziyaret ederek tekrar depoya dönmesi üzerine kurulu bir problemdir. Araç rotalama problemi kat edilen mesafeyi en küçükleyerek ulaşım giderlerini ve harcanan zamanı azalttığı gibi aynı zamanda araç sayısını en aza indirerek maliyetlerin düşmesine yardımcı olur. Buna ek olarak karbon salınımını da düşürerek çevreye verilen zararı azaltır. Son yıllarda tedarik zinciri ve dağıtım kanallarının rekabetçi piyasadaki önemi oldukça artmıştır. Dolayısıyla firmaların araç rotalama problemine olan ilgisi de artmıştır.</p> <p>Araç rotalama probleminin gerçek hayat ihtiyaçları üzerine birçok türevi oluşmuştur. Bunun sebebi farklı tipteki şirketlerin işleyişlerinin değişkenlik göstermesidir. Örneğin müşterilerin sadece belirli saat aralıklarında ziyaret edilebilmesi, kargo şirketinin aynı anda hem dağıtım hem de toplama işlemi yapması, bazı müşterilerin aynı araçla ve belirli bir sıra ile ziyaret edilme zorunluluğu olması gibi sebeplerden dolayı problemin birçok türevi literatürde tanımlanmış ve çalışılmıştır.</p> <p>Araç rotalama probleminin literatürde sıklıkla çalışılan türevlerinden birisi periyodik araç rotalama problemidir. Bu problem belirlenen bir periyot içerisinde müşterilerin bir veya daha fazla ziyaret edilmesi üzerine kuruludur. Her periyotta ziyaret edilecek müşterilere ve her periyot için araç rotalarına karar verilmektedir. Amaç genellikle toplam rota uzunluklarını en aza indirmektir. Ziyaret günlerinin uygun şekilde düzenlenmesi maliyetler ve araç sayısı açısından büyük avantaj sağlar.</p> <p>Bu projede periyodik araç rotalama problemi üzerine gerçek hayattan esinlenen iki yeni problem tanımlanmıştır ve bu problemler için matematiksel modeller, alt sınırlar ve sezgisel çözüm yöntemleri geliştirilmiştir. Geliştirilen yöntemlerin sonuç kalitesi ve süre performansları rassal olarak oluşturulan örnekler üzerinde test edilmiştir.</p> <p>İlk problem bir müşteriye yapılan her ziyareti takip eden günde farklı amaçlı bir ziyaret yapılmasını gerektirmektedir. İlk ziyaret talep toplama, ikinci ziyaret ise ürünlerin teslimi olarak düşünülebilir. İlk tip ziyaret küçük, hızlı ve zaman kapasiteli araçlarla, ikinci tip ziyaret ise görece büyük, yavaş ve fiziksel kapasiteye sahip araçlarla yapılmaktadır. Bu sebeple birbirini takip eden günlerde aynı rotaların kullanılması mümkün olmamaktadır.</p> <p>Gerçek hayatta müşterilerin daha sık ziyaret edilmesi satışları artırmakta ancak toplam rota uzunluğunun da artmasına sebep olmaktadır. Proje kapsamında çalışılan ikinci problemde, periyodik araç rotalama problemi kararlarına ek olarak müşterilerin ziyaret sıklığının müşteri talebini değiştirmesi durumu incelenmiştir.</p> <p>Proje kapsamında ele alınan problemler, detaylı olarak tanımlanmış, matematik modeli geliştirilmiş ve çözüm karmaşıklığı incelenmiştir. Problemler için alt sınır algoritmaları ve sezgisel yöntem geliştirilmiştir. Geliştirilen alt sınır algoritmaları ve sezgisel yöntemin performansları rassal örnekler üzerinde test edilmiştir.</p>
<p>Anahtar Kelimeler:</p>	<p>Araç rotalama problemi, periyodik araç rotalama problemi, sezgisel yöntemler, matematik programlama</p>
<p>Fikri Ürün Bildirim Formu Sunuldu Mu?:</p>	<p>Hayır</p>
<p>Projeden Yapılan Yayınlar:</p>	<p>1- Periyodik Araç Rotalama Probleminde Ziyaret Sıklığının Belirlenmesi (Bildiri - Ulusal Bildiri - Sözlü Sunum), 2- Periyodik Araç Rotalama Probleminin Bir Türevi Üzerine (Bildiri - Ulusal Bildiri - Sözlü Sunum), 3- Periodic Vehicle Routing Problem with Two Types of Visits (Bildiri - Uluslararası Bildiri - Sözlü Sunum), 4- Periyodik Araç Rotalama Probleminin Bir Türevi Üzerine (Bildiri - Ulusal Bildiri - Sözlü Sunum),</p>