

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**YERLEŐTİRME ROTALAMA PROBLEMİNİN YENİ
BİR MATEMATİKSEL MODEL VE PARÇACIK
SÜRÜ ALGORİTMASI İLE ÇÖZÜLMESİ**

YÜKSEK LİSANS TEZİ

Vildan AMİL

Enstitü Anabilim Dalı : ENDÜSTRİ MÜHENDİSLİĐİ

Tez Danışmanı : Prof. Dr. Harun Reşit YAZGAN

Haziran 2019

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YERLEŐTİRME ROTALAMA PROBLEMİNİN YENİ
BİR MATEMATİKSEL MODEL VE PARÇACIK
SÜRÜ ALGORTİMASI İLE ÇÖZÜLMESİ

YÜKSEK LİSANS TEZİ


Vildan AMİL

Enstitü Anabilim Dalı : ENDÜSTRİ MÜHENDİSLİĐİ

Bu tez 14.06.2019 tarihinde aŐağıdaki jüri tarafından oybirliĐi / oyçokluĐu ile kabul edilmiŐtir.


Jüri BaŐkanı


Üye


Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Vildan AMİL

13.06.2019

TEŐEKKÜR

Yüksek lisans eğitimim boyunca bilgi ve deneyimleri konusunda desteğini esirgemeyen, teşvik eden ve titizlikle yönlendiren çok değerli hocam Prof. Dr. Harun Reşit YAZGAN'a teşekkür ederim.

Matematiksel modelin geliştirilmesi ve uygulamasında desteğini esirgemeyen Ünsal Ozan KAHRAMAN ve Selim BALCI'ya teşekkür ederim.

Ayrıca manevi desteğini ve sevgisini hissettiren canım aileme ve ekip arkadaşlarım Burcu KIR SARI ve Beril AĞYOL'a çok teşekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR	i
İÇİNDEKİLER	ii
SİMGELER VE KISALTMALAR LİSTESİ	iv
ŞEKİLLER LİSTESİ	v
TABLOLAR LİSTESİ	vi
ÖZET	vii
SUMMARY	viii
BÖLÜM 1.	
GİRİŞ.....	1
BÖLÜM 2.	
YERLEŞTİRME ROTALAMA PROBLEMLERİ İLE İLGİLİ LİTERATÜR	3
ARAŞTIRMASI.....	
2.1. Yerleştirme Problemi.....	3
2.2. Rotalama Problemi.....	5
2.3. Yerleştirme Rotalama Problemi.....	7
2.3.1. Yerleştirme rotalama problemleri sınıflandırılması.....	9
2.3.2. Yerleştirme rotalama problemlerinde çözüm yaklaşımları.....	10
2.3.2.1. Kesin çözüm yöntemleri.....	10
2.3.2.2. Metasezgisel çözüm yöntemleri.....	12
BÖLÜM 3.	
TEZ ÇALIŞMASINDA ELE ALINAN YERLEŞTİRME ROTALAMA	13
PROBLEMİ	
3.1. Problemin Tanımlanması.....	13

3.2. Varsayımlar.....	13
3.3. Amaç ve Kısıtlar.....	14
3.4. Matematiksel Model.....	15
3.5. Matematiksel Modelin Kompleksliği.....	17
3.6 Matematiksel Yöntemin Kesin Yöntem ile Çözülmesi.....	19
BÖLÜM 4	
PARÇACIK SÜRÜ ALGORİTMASI.....	20
4.1. Tanım ve Özellikler.....	20
4.2. Standart Parçacık Sürü Algoritması.....	22
4.3. Ayrık Parçacık Sürü Algoritması.....	25
4.4. Parçacık Sürü Algoritması Parametreleri.....	26
4.5. Parçacık Sürü Algoritması Sosya Ağ Yapıları.....	28
4.6. Parçacık Sürü Algoritması Avantajları ve Dezavantajları.....	29
4.7. Parçacık Sürü Algoritması Uygulama Alanları.....	30
BÖLÜM 5	
UYGULAMA.....	31
5.1. Cathering Firmasında Yerleştirme Rotalama Problemi.....	31
5.1.1. Kullanılan parametreler ve veri seti.....	31
BÖLÜM 6	
SONUÇ.....	38
KAYNAKLAR	39
EKLER.....	44
ÖZGEÇMİŞ	48

SİMGELER VE KISALTMALAR LİSTESİ

DFA	: Dal ve fiyat algoritması
DKA	: Dal ve kesme algoritması
DSA	: Dal ve sınır algoritması
İDS	: İkili dallandırma stratejisi
KTP	: Karma tamsayılı programlama
LRP	: Yerleştirme rotalama problemleri (Location routing problem)
MILP	: Mixed Integer linear programming
NP-Hard	: Non polynomial-time hardness
PSO	: Parçacık Sürü Algoritması (Particle Swarm Optimization)
TP	: Tamsayılı programlama
TS	: Tabu arama (Tabu searching)

ŞEKİLLER LİSTESİ

Şekil 2.1. Yerleştirme problemleri işleyişi.....	4
Şekil 2.2. Yerleştirme rotalama problemleri işleyişi.....	7
Şekil 2.4. Rotalama problemleri gelişimi.....	8
Şekil 4.1. Parçacığın pozisyon değiştirmesi.....	24
Şekil 4.2. Merkez-tabanlı parçacık hareketi.....	26
Şekil 4.3. PSO Ağ Topolojileri.....	29
Şekil 5.1. 100 müşteri ve 8 tesis sayısına göre Google map üzerindeki noktalar....	33
Şekil 5.2. 100 müşteri ve 8 tesisin çözümü.....	34

TABLolar LİSTESİ

Tablo 2.1. Yerleştirme rotalama problemlerinin sınıflandırılması.....	9
Tablo 2.2. Yerleştirme rotalama probleminde kullanılan kesin yöntemler.....	10
Tablo 2.3 Yerleştirme rotalama probleminde kullanılan metasezgisel yöntemler..	12
Tablo 3.1. n tane değişkene karşılık gelen kısıt sayısı.....	18
Tablo 3.2. Müşteri koordinat ve talepleri.....	19
Tablo 3.3. Tesis koordinat ve talepleri.....	19
Tablo 3.4. LINGO sonucu özet tablo.....	19
Tablo 4.1. PSO algoritmasının işleyişi.....	25
Tablo 4.2. PSO ile ilgili yapılan çalışmalar.....	30
Tablo 5.1. Müşteriye ait koordinatlar ve talep miktarları.....	32
Tablo 5.2. Tesislere ait koordinatlar ve talep miktarları.....	34
Tablo 5.3. 10 parçacık sayısına göre rotalama ve toplam maliyet.....	35
Tablo 5.4. 20 parçacık sayısına göre rotalama ve toplam maliyet	35
Tablo 5.5. 30 parçacık sayısına göre rotalama ve toplam maliyet	35
Tablo 5.6. Parçacık sayısına göre toplam maliyet.....	36
Tablo 5.7. Parçacık sayısı ve iterasyona göre toplam maliyet.....	36

ÖZET

Anahtar kelimeler: Parçacık sürü algoritması (PSO), metasezgisel yöntemler, yerleştirme rotalama problemleri (LRP), karışık tamsayı programlama(MILP)

Çok boyutlu optimizasyon problemi olan yerleştirme rotalama problemi, toplam maliyeti düşürmek amacıyla, birbiri ile etkileşim halinde olan üç temel karar sürecinden oluşmuştur. Bu karar süreçleri, potansiyel tesis yerlerinden hangisinin açılacağı, hangi müşterinin hangi tesisten hizmet alacağı ve araçların hangi rotayı izleyeceğini ele alır.

Bu tez çalışmasında, yerleştirme rotalama problemi ile ilgili Daskin ve Perl (1985)'ün geliştirdikleri modele yeni kısıtlar dahil edilerek bir matematiksel model elde edilmiştir. Elde edilen modelde her bir kısıt denklemi ve optimizasyon denkleminin tek tek birbiri ile nasıl etkileşim halinde olduğu incelenmiş, modelin çalışma mantığı ortaya konmuştur. Ancak modeldeki değişken sayısı arttıkça kısıt sayısı üssel olarak arttığı için metasezgisel çözüm yöntemlerinden olan Parçacık Sürü Algoritması ile çözülmüştür. Bu çalışmada önerilen çözüm yaklaşımı 100 müşteri ve 8 yer için gösterilmiştir. Elde edilen sonuçlar geliştirilen ve önerilen yaklaşımın oldukça etkin olduğunu göstermektedir

A NEW MATHEMATICAL AND META-HEURISTIC APPROACH FOR THE SOLUTION OF THE LOCATION ROUTING PROBLEM

SUMMARY

Keywords: Particle swarm optimization, metaheuristics methods, location routing problems, mixed integer programming

Location routing problem, which is a multidimensional problem, consists of three decision processes interacting with one another in order to reduce the total cost. These decision processes address which of the potential plant locations will be opened, which customer will receive service from which plant and which vehicles will follow which route.

In this thesis, a mathematical model is obtained by adding new constraints to the model developed by Daskin and Perl (1985) on location routing problem. In the obtained model, how each constraint equation and optimization equation interact with each other is examined and the logic of the model has been put forward. However, as the number of variables in the model increased, the number of constraints increased exponentially, so the Particle Swarm Algorithm, which is one of the metaheuristic solution methods is referred. In this study the proposed approach has been carried out on the problem with 100 customers and 8 facility. The results imply that the proposed approach produce much more effective results.

BÖLÜM 1. GİRİŞ

Küresel rekabetin artması, müşteri taleplerinin hızlı değişimi ve maliyet üzerindeki artan baskı, işletmeleri rekabette üstünlük sağlamak için stratejik kararlar almaya zorlamaktadır. Bu kararlar; tesis yeri seçiminden, müşterilerin hangi depo/fabrika vb'den hizmet alacağına ve verilecek hizmetin hangi rotalama ile yapılacağına dair lojistik yönetimini kapsayan operasyonel ve stratejik kararları içerir.

Yerleştirme rotalama problemi bu noktada literatürde yer alan tesis yerleştirme ve araç rotalama problemlerini bütünleşik olarak ele alır ve problemi ayrı ayrı çözmek yerine birbiri üzerindeki etkisini bütünleşik olarak çözmeye çalışır. Hiyerarşik bir bakış açısı sunar ve potansiyel tesislerden hangi tesislerin açılması gerektiğine karar verilirken açılacak tesislere atanacak müşteriler belirlenir ve talep noktalarına hizmet götüreceği araç rotası belirlenir. Bu problemin genel amacı, sabit maliyet olan tesis yerleşim maliyeti ve rotalama maliyetinin toplamının minimize edilmesidir.

Tezin ikinci bölümünde bu noktadan hareket edilerek yerleştirme problemleri ve rotalama problemleri tanımlanarak işleyişlerinden bahsedilmiş akabinde yerleştirme rotalama problemleri detaylıca açıklanmıştır. Bu iki problemin birbirleri ile olan etkileşimi bütünleşik olarak ele alınmıştır. Literatürdeki çalışmalar incelendiğinde problemin çözümündeki örnekleme tesis ve müşteri sayısı az ise kesin çözüm yöntemleri aksi taktirde metasezgisel çözüm ile ilgili çalışmalar yapılmıştır ve kullanılan yöntemlerin bir kısmına değinilmiştir.

Tezin üçüncü bölümünde Daskin ve Perl (1985)'in modeline kısıtlar eklenerek yeni bir model elde edilmiştir ve bu yapılırken her bir denklem seti, kısıtlar ve optimizasyon denklemleri tek tek birbiri ile nasıl ilişkili olduğu belirtilerek katkı

yapılan kısıtlara değinilmiştir. Modeldeki veri sayısının artması durumunda oluşabilecek kısıt sayıları ve dolayısıyla neden metasezgisel yöntemlere başvurulduğunun temel mantığı üzerinde durulmuştur. LINGO programında küçük bir veri seti için modelin çözümü incelenmiştir.

Dördüncü bölümde sürü zekası ile hareket eden, hayvanların davranışından esinlenerek bir dizi iterasyon üzerine kurulu metasezgisel bir yaklaşım olan parçacık sürü algoritmasına dair tanımlamalar ve özellikleri, parametrelerine yer verilmiştir. Parçacığın komşuları ile arasındaki bilgi akışına göre oluşturulan sosyal ağ yapılarının ne anlama geldikleri, ayırık parçacık sürü algoritmasının işleyişi, algoritmanın avantajlı ve dezavantajlı kısımları ve algoritma ile ilgili olan literatürdeki bir kısım çalışmalar incelenmiştir.

Beşinci bölümde ise İstanbul Anadolu yakasında faaliyet gösteren bir catering firması, sanayi bölgesindeki 100 müşteriye hizmet verebilmek için potansiyel 8 tesisten maliyet optimizasyonu yapabilmek adına hem rotalama hem de yerleştirmesi metasezgisel yöntemlerden biri olan PSO algoritması ile çözüm bulunmuştur.

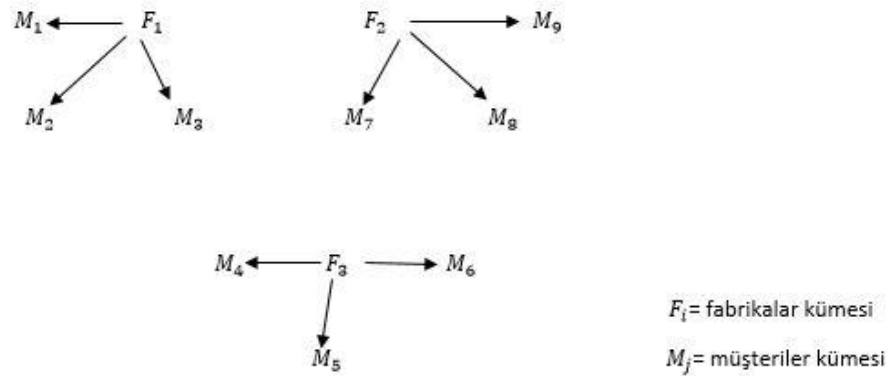
BÖLÜM 2. YERLEŐTİRME, ROTALAMA VE YERLEŐTİRME ROTALAMA PROBLEMLERİ

2.1. Yerleőtirme Problemi

Yerleőtirme problemi, birbirine belirli ve tanımlı mesafede olan müşterilerin/tüketicilerin taleplerini karşılamak için potansiyel tesis kümelerinden maliyet minimizasyonu sağlayarak nasıl konumlandırılacağına karar verilmesidir. Burada tesis yerinin kararı ve ayrıca tesislere atanacak müşterilerin hangilerinin olacağı kararı veriliyor (Şekil 2.1.). Amaç, sabit maliyet olan tesis açma maliyeti ve talep ve hizmet noktasındaki lojistik maliyeti minimizasyonunun sağlanmasıdır. Yerleőtirme problemleri önemli bir stratejik karardır ve birçok parametre ve kısıt içermesinden dolayı çözümlenmesi zordur. Gerek özel sektör gerek kamu sektöründe yerleşim problemlerine sıklıkla yer verilmektedir (Yiğit ve Türkbey, 2003; Eiselt, 2004; Küçükdeniz, 2009).

Literatürde ilk çalışma yirminci yüzyılın başlarında Alfred Weber tarafından yapılmıştır. Müşteri taleplerini karşılayacak ve mesafeden dolayı oluşacak maliyeti minimum yapmak için bir model ortaya koymuştur. Aynı zamanda bu model, Fermat problemi, Fermat-Weber problemi, Steiner problemi, Steiner-Toriçelli problemi, tekli medyan problemi ve merkez medyan problemi gibi adlarla da bilinmektedir. Modelde 3 müşterinin talebini tek bir tesisten karşılanması amaç edinmiştir. Hakimi (1964)'de p medyan çalışması olarak adlandırılan birden fazla tesisin yerleőtirilmesi üzerine çalışmalar yapmıştır. Çalışmasında polis merkezlerinin otoyollara yerleőtirilmesi ve telekomünikasyon şebekesinin ağ bağlantı noktalarının en uygun yere konumlanmasını ele almıştır (Kısakol, 2015).

Genel olarak tesis yerleştirme problemleri; tesisin yerleştirileceği alan, yerleştirilecek tesisler, belli bir talebi olan müşteriler, müşteri ve tesis arasındaki etkileşim, müşteriler ve tesisler arasındaki mesafeyi belirleyecek ölçümler ve modeldeki kısıtlamaları ifade eden ana unsurlar ile tanımlanır. Alan; tesisin yerleştirileceği coğrafi bir bölgeye karşılık gelir. Herhangi bir nokta tesisin yerleştirilmesi için uygunsa sürekli alan, aksi taktirde bir dizi noktalardan herhangi biri ise ayrık olarak adlandırılır. Tesisler; sayısı, kapasitesi, tedarik ve servis hizmeti ile ayırt edilir. Bazı tesisler sınırsız kapasiteye sahiptirler dolayısıyla sınırsız müşteri talebini karşılar; bazı tesisler ise sınırlı sayıda müşteri talebini karşılar. Müşteri taleplerinin tesislerden karşılanması daha çok en yakın tesise atanma kısıtı ile gerçekleşir. Hizmet sektöründe hastane, postane vb alanlarda daha çok hizmete olan yakınlık ön plandadır ve oluşacak maliyet hizmetten faydalanacak kişilerden elde edilir. Fabrika, depo ve dağıtım yeri için yerleşim yeri seçiminde, ulaştırma maliyeti ve dağıtım maliyetine etki eder ve bu doğrudan karı etkilediği için en aza indirmek istenir. Her iki durumda da tesis ile hizmet noktalarının konumu stratejik bir karar olmaktadır ve stratejik planlamada kritik bir unsurdur (Barbati, 2013).



Şekil 2.1. Yerleştirme problemleri işleyişi

Yerleştirme problemleri literatürde 5 kategoriye ayrılmıştır: p medyan problemi, p-merkez problemi, kapasite kısıtsız tesis yerleştirme problemi, kapasite kısıtlı tesis yerleştirme problemi ve karesel atama problemidir. p medyan problemi, p adet tesis için en uygun yerlerin belirlenmesi ve talep noktalarının bu tesislere atanarak maliyet minimizasyonu sağlanmasını amaç edinir. p merkez problemi, p adet tesise atanacak

olan talep noktalarının maksimum mesafesinin en küçüklenmesidir. Kapasite kısıtsız yerleştirme problemi, açılacak tesis sayısı belli değildir ve kapasite kısıtı olmadığı için birden fazla tesisten hizmet alma durumu olmayacaktır. Kapasite kısıtlı tesis yerleştirme problemi tesisler sınırlı kapasitedir ve birden fazla tesisten hizmet alma durumu olacaktır. Karesel atama problemi, her yerleşim yeri için farklı tesis maliyeti olan tesislerin minimum toplam maliyeti ile eldeki tesislere atanmasıdır. Kurulum ve lojistik maliyeti ile tesisler arası akış miktarının azalması hedeflenmektedir (Basti ve Özçakar, 2012; Küçükdeniz, 2009; Basti, 2012; Dökeroğlu, 2017).

2.2. Rotalama Problemi

Araç rotalama problemi (ARP) belli bir kapasitesi olan araç kümesi ile farklı yerleşim yerlerinde olan talep noktalarına olan hizmetin taşıma maliyeti ve mesafesini en aza indirgeyerek en uygun rotalamanın yapılmasıdır (Çeyrekoğlu, 2017). Yani ARP’de x tane müşteriye hizmet vermek için y tane araç rotası ile başlangıç noktasındaki araçlar, toplam maliyeti minimize edecek şekilde müşteri talebini karşılamak zorundadır (Dursun, 2009).

Modelde talebi karşılanacak olan müşteriler yalnızca bir kere ziyaret edilir ve araç rotası başlangıç noktasındaki dağıtım noktasından başlayarak tekrar başladığı noktaya gelmek suretiyle rotalamanın tamamlanması problemin temel kısıtlarını oluşturur (Atasagun, 2015). Bu problemlerin temel hedefi, farklı lokasyonlarda bulunan müşterilerin talebinin karşılanması için minimum zaman, mesafe ve rotalama ile talebini karşılamaktır. Dikkat alınan temel ususlar aşağıdaki gibidir (Alparslan, 2015; Çeyrekoğlu, 2017):

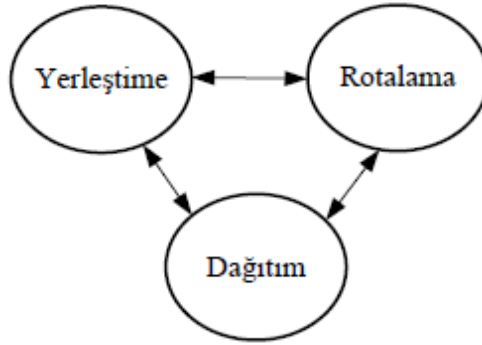
- Minimum rotalama maliyetini sağlamak,
- Sabit veya değişken minimum araç maliyetini sağlamak,
- Müşterilerin tamamının hizmet talebinin karşılanmasını sağlamak,
- Minimum rota süresini sağlayacak rotalama oluşturmak,
- Minimum araç sayısı sağlamak.

ARP'nin temel bileşeni talep yapısı, talep noktalarına taşınacak malzemenin tipi, dağıtım noktaları ve araç filosudur. Talep önceden biliniyorsa statik talep olarak adlandırılırken, bazı düğüm noktalarında bilinmiyor ve rotalama esnasında belli oluyorsa dinamik talep olarak adlandırılır. Malzemeler tipi, taşınacak malzemenin tipine göre modelin daha kompleks hale getirebilir. Gazete dağıtımı, postacı gibi taşınan malzemeler problemi karmaşık hale getirmez. Cathering firmasında dağıtım yapılırken yemek ısının korunması için belli bir mesafeye kadar rotalama yapılması kısıtı ve benzeri kısıtlamalar getirilmesi modeli karmaşıklaştırır. Depolar, genellikle araç rotasının başladığı ve başladığı noktaya geri döndüğü yerdir. Depo sayısı tekli veya çoklu olabilir. Dağıtım noktaları sabit veya önceden biliniyorsa hizmet verilecek noktalara hangi rota ile gideceği belirlenir. Araç filosu homojen veya heterojen olabilir. Homojen araç filusunda araçların kapasitesi aynıdır, heterojen filoda araç kapasitesi farklıdır (Çalışkan, 2011).

Literatürde birçok araç rotalama problemi çeşiti bulunmaktadır, problemin sahip olduğu kısıtlar ile birbirinden farklılaşır. Kapasite kısıtlı araç rotalama probleminde araçların belli bir kapasitesi vardır ve müşteri talepleri doğrultusunda minimum yol maliyeti göz önünde bulundurularak rotalama yapılır. Mesafe kısıtlı rotalama problemi, taşınacak malzeme cinsi veya sürücünden dolayı belli bir süreden fazla mesafe kısıtının eklenmesi durumunda kullanılır. Önce dağıtım sonra toplama araç rotalama problemi, önce talep noktasına ait ihtiyaçlar dağıtılacak daha sonra yine bu araçla müşterilerden toplama yapılacaktır. Buna sütçü örneğindeki gibi sütün tüm talep noktalara verildikten sonra boşları toplamak için talep noktalarına gitmesi örneği verilebilir. Eş zamanlı toplama-dağıtım araç rotalama problemi, eş zamanlı olarak müşterilerden malzemelerin alınıp müşteri ihtiyacının eş zamanlı verildiği bir modeldir. Bölünmüş dağıtımlı araç rotalama problemi, müşterinin talebi sağlamak için birden fazla araçtan karşılanması durumudur. Çok depolu araç depolama problemi, müşterilere hizmet verilebilecek birden fazla deponun olması durumudur. Periyodik araç rotalama problemi, müşterilerin taleplerini karşılamak için periyodik olarak servis edilebilme sıklığı belirlenir. Zaman pencereci araç rotalama problemi, müşteriye hizmet verilecek belli bir zaman aralığında hizmet verilmesi gerekir (Dursun, 2009; Demirtaş, 2009).

2.3. Yerleştirme Rotalama Problemi

Bruns (1998)'de yerleştirme rotalama problemlerini (LRP) “rota planlaması dikkate alınarak yerleşke optimizasyonu” olarak tanımlamıştır. Tanımlama hiyerarşik bir bakış açısına dayanmaktadır; sınırsız veya belirli bir potansiyel gruptan hangi tesislerin (fabrikalar, depolar, ambarlar, limanlar vb.) açılması gerektiğine karar verilmelidir, ancak bunu yapabilmesi için eşzamanlı olarak belli bir araç grubu ile açılmış tesislerden talep noktalarına hizmet götüreceği araç rotasının nasıl olacağını kararı verilmelidir (Nagy ve Salhi, 2007; Schneider ve Drexl, 2017).



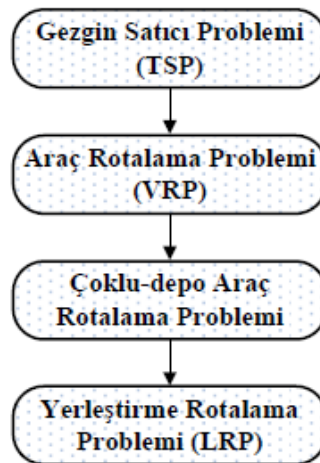
Şekil 2.2. Yerleştirme rotalama işlevişi (Hassanzadeh ve ark.,2009)

Yerleştirme rotalama probleminde toplam maliyeti oluşturan rotalama ve yerleşim yeri maliyetinin minimize edilmesi amaçlanır. Her iki problemin birbiri ile olan ilişkileri incelenerek bütünleşik karar verilmesi beklenir. Perl ve Daskin'a göre LRP birbiriyle ilişkili 3 temel karardan oluşmuştur: tesisin nereye konumlandırılacağı, müşterilerin tesislere nasıl tahsil edileceği ve müşterilere hizmet vermek için araçların nasıl rotalama olacağıdır (Şekil 2.2.). Problem çözme becerisinin artmasının nedenlerinden biri model tarafından yapılması gereken çok daha fazla kararın var olmasıdır. Bu kararlar şunları içermektedir (Marikanis, 2009):

- Yerleştirilecek kaç tesis olacak,
- Tesisler nerede olacak,
- Hangi müşteriler hangi tesisten hizmet alacak,
- Hangi rotalara hangi müşteriler atanacak,

- Müşteriler hangi rotada hangi sırada olacak.

Literatürde yapılan çalışmalar incelendiğinde rotalama problemlerinin birbiri ile olan ilişkileri ve gelişim detayları Şekil 2.4.'deki gibi açıklanabilir. Araç rotalama problemlerinin temelini oluşturan gezgin satıcı problemde tesisler arasında en kısa yol mesafesi göz önünde bulundurulur. Rotalama içerisinde tek bir araç ve tesis olması, tüm talep noktalarına hizmet verilmesi modelin temel kısıtlarıdır. Modeldeki tek rota ve tek tesis kısıtı genişletilerek birden fazla rotalama olması durumunda gezgin satıcı modeli araç rotalama problemine dönüşür. Araç rotalama ile istenen yol maliyetini minimize edilerek rotalamaların oluşturulması istenir (Kızılateş ve Nuriyeva, 2016). Bir sonraki aşamada, çok depolu araç rotalamadır. Birden fazla depo ve müşterilerin olduğu bir örnekte müşteriler depolara atanır ve her bir araç bulunduğu depodan müşterilere minimum maliyet oluşturacak rotalamalara izin verilir ve yine başladığı depoya geri döner (Düzakın ve Demircioğlu, 2009). Her rotalama hem de yerleştirme modeli ile yakında ilişkili olan yerleştirme rotalama problemde ise potansiyel tesis yerlerinden hangisinin açılacağı, açılan tesislere hangi müşterilerin atanacağı ve tesislere atanan müşterilerin hangi rotalama ile atanacağı sorularına cevap aranır (Ahn, 2018).



Şekil 2.4. Rotalama problemlerinin gelişimi (Ahn, 2008)

Eğer tüm müşteriler bir tesise atanırsa yerleştirme rotalama problemi standart yerleştirme modeline dönüşür. Diğer taraftan yerleşim yerleri belirlenirse problem

araç rotalama problemine dönüşür. Geniş bir bakış açısıyla, yerleştirme rotalama problemi dağıtım probleminin temelini oluştururken birleşimsel bir matematiksel model olduğu görülüyor (Nagy ve Salhi, 2007).

2.3.1. Yerleştirme rotalama problemleri sınıflandırılması

Min ve arkadaşları (1998) yerleştirme rotalama problemlerini tesislerin bulunduğu yere, araç yollarının düzenine ve sorunun tümüne ilişkin olarak literatürde yapılan çalışmaları referans alarak Tablo 2.1.'deki gibi sınıflandırmıştır (Nagy ve Salhi, 2006; Akpınar, 2009).

Tablo 2.1. Yerleştirme rotalama problemlerinin sınıflandırılması

Hiyerarşik Seviye	Tek aşamalı, Çift aşamalı
Talep Yapısı	Deterministik, Stokastik
Tesis Sayısı	Tek tesis, Çok tesis
Araç Filosu Büyüklüğü	Tek araç, Çok araç
Araç Kapasitesi	Kapasite sınırlı, Kapasite sınırsız
Fabrika Kapasitesi	Kapasite sınırlı, Kapasite sınırsız
Tesis Katmanı	Birinci katman, İkinci/ara katman
Planlama Ufku	Tek periyot, Çoklu periyot
Zaman Kısıtı	Zaman kısıtı verilmemiş, Gevşek son teslim tarihi, Kesin son teslim tarihi
Amaç Fonksiyonu	Tek amaç fonksiyonu, Çok amaç fonksiyonu
Model Veri Tipi	Hipotetik Veri, Gerçek veri
Çözüm metodu	Kesin algoritmalar, Sezgisel algoritmalar

Hiyerarşik seviye müşterilerin belli bir araç rotası ile doğrudan hizmet alıp almadığı ile ilgilidir. Doğrudan merkez depodan hizmet alıyor ise tek aşamalıdır değilse k- aşamalı problem sınıfına girecektir. Problemi çözümlerken kullanılan veriler biliniyor ise deterministiktir aksi taktirde stokastik veriler ışığında problem çözülür. Araç sayısı ve tesis sayısı bir veya birden fazla olabilir, tesis ve araçlar belli bir kapasite kısıtı altında veya kapasite kısıtsız her talebi karşılıyor olması ile ayrışacaktır. Amaç fonksiyonu problemde istenilen hedefler doğrultusunda tek veya birden fazla değişkenlerin optimizasyonu ile sağlanacaktır. Hedefe ulaşmak için kullanılan parametreler arttıkça çözüme kesin yöntemler ile ulaşmak zor olacağından metasezgisel çözüm yöntemleri için çözüm aranacaktır (Oost, 2015).

2.3.2. Yerleştirme rotalama problemlerinde çözüm yaklaşımları

Yerleştirme rotalama problemlerinde çözüm yöntemleri kesin çözüm yöntemleri ve metasezgisel çözüm yöntemleri olmak üzere iki gruba ayrılır.

2.3.2.1. Kesin çözüm yaklaşımları

Kesin yöntemler matematiksel programlama temellidir. Oluşturulan modellerde; her bir turda tek bir tesis olması, tesislerin rotalara bağlanmaması ve değişkenlerin tamsayılı veya binary değişken olması kısıtları bulunmaktadır. Literatürde kullanılan kesin çözüm yöntemleri aşağıdaki gibidir (Hassanzadeh ve ark, 2009; Akpınar, 2009):

- Dal sınır algoritması
- Dinamik programlama
- Karma tamsayılı programlama
- Doğrusal olmayan programlama

Tablo 2.2. Yerleştirme rotalama problemlerinin çözümünde kullanılan kesin yöntemler

Yazar	Çözüm Metodu	Problem Boyutu
Laporte G., Nobert Y (1981)	İDS	15-50
Laporte, G. ve ark.(1986)	TP	47
Laporte, G.Dejax, P. J.(1989)	TP	28
Zografos, K. G. Samara, S.(1989)	KTP	16-32
Laporte, G. ve ark. (1989)	DSA	83
Daskin, M. S. ve ark. (2007)	DFA	40
Baldacci, R. ve ark.(2011)	DFA	14-199
Belenguer, J.M ve ark.(2010)	DKA	5-40
Benavent ve ark.(2011)	DKA	40-80
Gonzalez ve ark.(2014)	DKA	25-50

İDS: İkili dallandırma stratejisi, TP: Tamsayılı programlama, KTP:Karma tamsayılı programlama, DSA:Dal ve sınır algoritması, DFA: Dal ve fiyat algoritması, DKA:Dal ve kesme Algoritması

Laporte ve Nobert (1981) yaptığı çalışmasında çoklu kapasite kısıtsız araçlar ve kapasite kısıtsız tesis ile deterministik bir modelde ikili dallandırma strateji ile probleme çözüm bulmuştur. En iyi rotalama ve en iyi depo yerini seçerek minimum maliyetle kesin çözüm yöntemini kullanmıştır. Laporte ve ark.(1986) çoklu kapasite

kısıtlı araçlar ve kapasite kısıtlı tesis ile kesin yöntemlerden olan tamsayı programlama ile çözerken Laporte ve Dejax (1989) kapasite kısıtsız modelinde bu kesin çözüm yöntemini kullanmıştır. Zografos ve Samara (1989) kapasite kısıtsız araç ve kapasite kısıtlı tesisi karma tamsayı programlama ile tehlikeli atıkların taşınması ve bertaraf edilmesinde kullanmıştır. Daskin ve ark. (2007) kapasite kısıtsız tesis ve çoklu araç probleminde dal ve fiyat algoritmasını kullanırken Baldacci ve ark.(2011) kapasite kısıtlı çoklu araç ve tesis modelinde kesin çözüm yöntemlerini kullanmıştır. Belenguer ve ark.(2010) ve Benavent ve ark.(2011) kapasite kısıtlı tesis ve araç problemlerinde dal ve kesme algoritmasını kullanmıştır. Gonzalez ve ark. (2014) hangi tesislerin açılacağı, müşterilerin hangi tesislerden hizmet alınacağını hesap ederek kesin çözüm yöntemlerden olan tamsayı lineer programlardan dal ve kesme algoritmasını kullanmıştır (Tablo 2.2.)

Kesin algoritmalarında sıklıkla gevşeme kısıtları bulunur ve kısıtlara dahil edilir. Alt tur eliminasyonu ile tüm turlarda tek bir tesis yeri olması sağlanır, rotadaki müşteri talepleri tek bir tesisten alınır. Bir depoyu diğer depoya bağlayan rotalamalara izin verilmez, bir depodan başlayan araç talep noktalarına hizmet verdikten sonra başladığı noktaya geri döner. Değişkenleri tamsayı olması istenir.

2.3.2.2. Metasezgisel çözüm yöntemleri

Yerleştirme rotalama problemlerinde uygulanan birçok metasezgisel çözüm yöntemleri vardır. Literatürde kullanılan yöntemlerden bir kısmı Tablo 2.3.'deki gibidir (Marikanis, 2009).

- Tabu arama (TS)
- Benzetimli Tavlama
- Açgözlü Randomize Adaptif Arama Prosedürü
- Genetik Algoritma
- Değişken Komşuluk Araması
- Karınca Kolonisi Algoritması
- Parçacık Sürü Algoritması (PSO)

Tablo 2.3. Yeleştirme rotalama problemlerinin çözümünde kullanılan metasezgisel yöntemler

Yazar	Çözüm Metodu	Problemin Boyutu
Burke, L ve Tuzun, D.,1999	Tabu arama	110-220
Derbel, H. ve ark, 2012	Genetik Algoritma	20-60
Burketove, A. ve ark, 2016	Genetik Algoritma	50-80
Bouhaf, L ve ark, 2006	Genetik Algoritma	Benchmark örnekleri
Bouhaf, L ve ark, 2006	Karınca Kolonisi	Benchmark örnekleri
Chen, C ve Ting, C.,2013	Karınca Kolonisi	Benchmark örnekleri
Nadizadeh, A. ve ark, 2011	Karınca Kolonisi	Benchmark örnekleri
Nadizadeh, A. ve ark, 2011	Açgözlü Algoritması	Benchmark örnekleri
Yu, F. ve ark, 2010	Benzetimli Tavlama	25-220
Kazemi, A. ve ark., 2015	PSO Algoritması	100
Dou, F ve ark., 2017	PSO Algoritması	21-111
Aydın, İ. ve ark, 2018	PSO Algoritması	479-2000
Saplioglu, T. ve ark., 2018	PSO Algoritması	13

Yerleştirme rotalama problemlerinde kullanılan metasezgisel yöntemler çözümlenirken birçok kategoride kullanılır. Sıralı yöntemlerde sorunlar kademeli olarak çözümlenir. Önce tesislerin yerleşim yeri belirlenir, müşteri ve potansiyel tesis yerlerinin mesafeleri hesaba katılarak minimum maliyet oluşturacak atamalar yapılır akabinde açılan tesislere atanacak müşterilere odaklanılır. Kümeleme temelli yöntemlerde, her bir müşteri kümesine ayrılarak her küme için bir depo yeri belirlenir ve belirlenen kümelemede araç rotalaması yapılır veya tesislerin yerleştirilmesinden önce gezgin satıcı modeli çözülür. İteratif yöntemler, sıralı yöntemin birkaç yinemesinin gerçekleştiği her bir adımdan bir önceki adımı kullanıldığı yöntemlerdir. Hiyerarşik yöntemlerde problemler bölünmeden hiyerarşik olarak ele alınır. Çözümlenecek yerleştirme ve rotalama kararı eşit önemde görülmemekle birlikte önce tesislerin yerleşimi yapılır daha sonra araç rotalama problemi çözülür (Sambola, 2015)

BÖLÜM 3. TEZ ÇALIŞMASINDA ELE ALINAN YERLEŞTİRME ROTALAMA PROBLEMİ

Yerleştirme rotalama problemi, farklı çalışma alanlarında farklı kısıtlar altında ele alınarak birçok şekilde tanımlanmıştır. Tesis sayısı, hiyerarşik seviye, araç kapasiteleri gibi kısıtlardan dolayı matematiksel modeli farklılık göstermektedir.

Bu bölümde, ele alınan modelin matematiksel modeli sunulacaktır. Yapılan varsayımlar açıklandıktan sonra problemde kullanılan değişkenler ve kısıtlar verilecek; ardından her bir kısıtın ne anlam ifade edilerek küçük bir veri seti için LINGO programında matematiksel model çözülecektir.

3.1. Problemin Tanımlanması

Bu tez kapsamında problemin tanımı şu şekildedir. Yerleştirme rotalama problemi çözülerek potansiyel tesis yerlerinden hangi tesislerin açılacağı ve müşterilerin açılan tesislerden hangisinden hizmet alacağı ve araç rotaları belirlenecektir. Aday tesisler ve müşterilerin yerleri bilinmektedir. Müşterilerin talepleri ve dağıtım araçlarının kapasiteleri ve tesis kapasiteleri de bilinmektedir.

3.2. Varsayımlar

Problemde amaç olarak; hangi tesisin açılacağı ve müşterilerin açılacak tesislere dağılımının nasıl yapılacağı göz önünde bulundurularak toplam maliyet minimize edilmeye çalışılır.

Problemin kısıtlarını tanımlamak için belirli varsayımlar üzerinden çözüm aranmıştır. Bu matematiksel modeldeki varsayımlar şu şekildedir:

Problemin hangi seviyede ele alınacağıının belirlenmesi gerekir. Bu çalışmada tek seviyeli olarak ele alınmıştır; tesisler ve müşteriler aynı sistemde yer almalıdır.

- Problemin amacı maliyeti minimize etmektir.
- Bütün müşterilerin yeri biliniyor ve müşterilerin tamamına hizmet verilmelidir.
- Müşterilerin talebi biliniyor.
- Kurulacak tesis sayısında herhangi bir kısıt bulunmamaktadır.
- Tesislerde kapasite kısıtı bilinmektedir.
- Araçların kapasite kısıtları bilinmektedir.
- Her araç müşterilere hizmet verdikten sonra ayrıldığı depoya dönmek zorundadır.
- Bir müşteri talebi tek bir tesisten karşılanmalıdır.

3.3. Amaç ve Kısıtlar

Amaç: Tesis yerleşimi ve araç rotalama için katlanılacak maliyeti en küçükleme.

Kısıtlar:

- Her müşteri tek bir rotadan hizmet alır.
- Araç kapasitesi aşılamaz.
- Sadece müşterilerden oluşan rota olamaz, her rotada mutlaka bir tesis olmalı.
- Her araç geldiği noktadan ayrılmalıdır.
- Bir rota sadece bir tesis içerir.
- Bir rotada i. müşterisi ve j. tesisinden geçiyorsa i müşterisi j tesisinden hizmet alır.
- Bir tesis, bu tesisi kullanan müşterilerin talebi kadar üretim yapar.
- Bir tesis t_j üretim kısıtını aşamaz.

Parametreler:

- Aday tesis yerleri,

- Müşterilerin bulunduğu yer,
- Müşteri talep miktarı,
- Tesis kuruluş maliyetleri
- Tesisler için birim üretim maliyeti,
- Araçlar için birim taşıma maliyeti,
- Araç kapasitesi,
- Tesislerdeki üretim.

Kararlar:

- Tesisler nereye kurulacak,
- Hangi müşteri hangi tesisten hizmet alacak,
- Her aracın hizmet vereceği rota.

3.4. Matematiksel Model

Problemin matematiksel modellemesinde Perl ve Daskin (1985) modeli temel alınmıştır (Akpınar, 2009; Daskin, 2005).

Kümeler:

I: müşteriler kümesi

J: Aday tesisler kümesi

P: tüm noktalar kümesi (I∪J)

k: Araçlar kümesi

Parametreler:

$f_j = j \in J$ fabrikanın kuruluş maliyeti

$d_{ij} = i \in P$ ile $j \in P$ arasındaki mesafe

$v_j = j \in J$ fabrikasında üretimin birim maliyeti

$h_i = i \in I$ müşterisinin talep miktarı

$t_j = j \in J$ aday fabrikanın üretim kısıtı

$r_k = k$. aracın kapasitesi

$l_k = k$. aracın birim mesafede harcadığı yakıt maliyeti

$w_j = j$. Tesisin üretimi

Karar değişkenleri:

$$Z_{ijk} = \begin{cases} 1, & i. \text{ müşteri } j. \text{ tesisten } k \text{ rotası ile alır.} \\ 0, & i. \text{ müşteri } j. \text{ tesisten } k \text{ rotası ile almaz.} \end{cases}$$

(3.1)

$$Y_{ij} = \begin{cases} 1, & i. \text{ müşteri } j. \text{ tesisten hizmet alır.} \\ 0, & i. \text{ müşteri } j. \text{ tesisten hizmet almaz.} \end{cases}$$

(3.2)

$$X_j = \begin{cases} 1, & j. \text{ tesisi açılır.} \\ 0, & j. \text{ tesisi açılmaz.} \end{cases}$$

(3.3)

Model:

$$\text{enaz} \quad \sum_{j \in J} f_j X_j + \sum_{j \in J} v_j \cdot (\sum_{i \in I} h_i Y_{ij}) + \sum_{k \in K} l_k (\sum_{j \in P} \sum_{i \in P} d_{ij} Z_{ijk})$$

(3.4)

$$\sum_{k \in K} \sum_{j \in P} Z_{ijk} = 1 \quad ; \quad \forall i \in I$$

(3.5)

$$\sum_{i \in I} h_i \sum_{j \in P} Z_{ijk} \leq r_k \quad ; \quad \forall k \in K$$

(3.6)

$$\sum_{i \in V} \sum_{j \in V} \sum_{k \in K} Z_{ijk} \geq 1; \quad \forall V \subset P \text{ öyle ki } J \subset V$$

(3.7)

$$\sum_{j \in P} Z_{ijk} - \sum_{j \in P} Z_{jik} = 0; \quad \forall i \in P, \quad \forall k \in K$$

(3.8)

$$\sum_{j \in J} \sum_{i \in I} Z_{ijk} \leq 1; \quad \forall k \in K$$

(3.9)

$$\sum_{m \in P} Z_{imk} + \sum_{h \in P} Z_{jhk} - Y_{ij} \leq 1 ; \quad \forall_j \in J; \quad \forall_i \in I ; \quad \forall_k \in k$$

(3.10)

$$\sum_{j \in J} w_j - \sum_{i \in I} h_i \sum_{j \in J} Y_{ij} = 0$$

(3.11)

$$w_j - t_j X_j \leq 0$$

(3.12)

$$X_j \in \{0,1\} ; \quad \forall_j \in J$$

(3.13)

$$Y_{ij} \in \{0,1\}; \quad \forall_i \in I , \quad \forall_j \in J$$

(3.14)

$$Z_{ijk} \in \{0,1\}; \quad \forall_i \in P, \forall_j \in P, \forall_k \in K$$

(3.15)

$$w_j \geq 0; \quad \forall_j \in J$$

(3.16)

Matematiksel modeli şöyle açıklayabiliriz:

- Tesis kurma maliyeti, her birim için üretim maliyeti ve ulaştırma maliyetinden oluşan toplam maliyet en küçükmeye çalışılacaktır (3.4).

Problemin kısıtları için ise şu şekilde sıralanabilir:

- Her müşteri bir rotadan hizmet alır (3.5).
- Bir araç rotası hesaplanırken araç kapasitesi aşılamaz (3.6).
- Her rota mutlaka bir tesisten geçmelidir, sadece müşterilerden oluşan bir rota olması engellenmiştir (3.7).
- Bir rotanın tanımlanabilmesi için her araç geldiği noktadan ayrılmalıdır (3.8).
- Bir rota sadece bir tesisten geçmelidir (3.9).
- Bir rota i. müşterisi ve j tesisinden geçiyorsa i müşterisi j. tesisinden hizmet alır (3.10).
- Bir tesis, bu tesisi kullanan müşterilerin talebi kadar üretim yapar (3.11).
- Bir tesis üretim kısıtını aşamaz (3.12).
- Modelde 0-1 tamsayılı değişken olma kısıtı verilmiştir (3.13, 3.14,3.15).

Matematiksel modele yapılan katkılar aşağıdaki gibidir:

- Daskin modelinde tesis kapasitesinin sınırsız olduğu varsayılmıştır bu modelde her bir tesis için üretim kapasite kısıtı eklenmiştir.
- Müşteri taleplerinin tesis kapasitesinin aşmaması kısıtı dahil edilmiştir.
- Her bir rota için farklı kapasitede araç kullanılabilir.

3.5. Matematiksel Modelin Kompleksliği

Yerleştirme rotalama NP-hard problemlerindendir. Bu tür problemlerde modeldeki değişken sayısı arttıkça optimum çözüme ulaşmak zordur. Dolayısıyla çözüme ulaşmak için birçok alanda kullanılan metasezgisel algoritmalar geliştirilmiştir. Modeldeki değişken sayısına bağlı olarak oluşabilecek kısıt sayısını irdelediğinde modelin kompleksliği daha iyi anlaşılacaktır. i , j ve k değişkenlerin notasyonu olduğundan bunların sayısını belirten sabitler için farklı notasyonlar kullanılmaktadır.

i_n = müşteri sayısı

j_n = aday tesis sayısı

k_n = araç sayısı

İki gruba böldüğümüz kısıtları her gruptan şu şekilde kısıt denklem çıkacaktır.

1. i_n kadar kısıt
2. k_n tane kısıt
3. $2^{i_n}-1$ tane kısıt
4. $i_n \cdot k_n$ tane kısıt
5. k_n tane kısıt
6. $i_n \cdot k_n \cdot j_n$ tane kısıt
7. j_n tane kısıt
8. j_n tane kısıt

9. j_n tane kısıt
10. $i_n \cdot j_n$ tane kısıt
11. $i_n \cdot j_n k_n$ tane kısıt
12. j_n tane kısıt

Modeldeki her bir kısıtın (toplamda 12 kısıt) değişken sayısına bağlı olarak oluşturacağı kısıt sayısı i_n , j_n ve k_n bir fonksiyon olarak yazılmıştır.

Tablo 3.1. n tane değişkene karşılık gelen kısıt sayısı

n	Kısıt sayısı
2	2^2-1
3	2^3-1
4	2^4-1
⋮	⋮
10	$2^{10}-1$

Tablo 3.1.'de görüleceği gibi n değişkeni sayısı arttıkça modeldeki kısıt sayısı artacaktır ve optimum çözüme ulaşmak mümkün olamayacağından metasezgisel yöntemlere başvurulacaktır.

3.6. Matematiksel Modelin Kesin Yöntem ile Çözülmesi

Matematiksel model, 3 müşteri ve 2 tesis yeri için LINGO 18.0 programı ile çözülmüştür. Açık formdan da anlaşılacağı üzere değişken sayısı arttıkça kısıt sayısının artacağı dolayısıyla kesin yöntemler ile çözülebilmesinin zor olacağı görülmektedir (EK 1).

Tablo 3.2.'de müşteri koordinatları ve müşterilerin talebi verilmiştir. Tablo 3.3.'de tesis yerine ait koordinatlar, tesis kurulum maliyeti ve değişken maliyeti verilmiştir. Ayrıca araç kapasitesinin 100 kişilik talebi, tesis kapasitesinin ise 150 ve 100 kişilik talebi karşılayacağı bilinmektedir.

Tablo 3.2. Müşteri koordinatları ve talepleri

Müşteri No	X koordinatı	Y koordinatı	Talep miktarı
1	5	8	50
2	10	8	50
3	5	2	40

Tablo 3.3. Tesis koordinatları ve diğer detaylar

Tesis no	x koordinatı	y koordinatı	Tesis kapasitesi (kişi sayısı)	Tesis kurulum maliyeti (TL)
1	1	5	150	30000
2	9	5	100	20000

Sonuç olarak model çözüldüğünde a ve b tesislerinin açıldığı ve toplam maliyetin ortalama 55592,98 TL görülmüştür. Rotalama da a-1-3-a ve b-2-b şeklindedir.

Tablo 3.4. LINGO sonucu özet tablo

Müşteri sayısı	Açılan tesis sayısı	Rotalama	Toplam maliyet (TL)
3	2	a-1-3-a b-2-b	55592,98

BÖLÜM 4. PARÇACIK SÜRÜ ALGORİTMASI (PSO)

Bu bölümde Parçacık Sürü Optimizasyonu Algoritması tanım ve özellikleri, standart PSO algoritmasının kurgusu, algoritmanın temel parametreleri ve algoritmanın avantajlı ve dezavantajlı noktaları detaylı bir şekilde anlatılacaktır.

4.1. Tanım ve Özellikleri

Parçacık Sürü Optimizasyonu (PSO) 1995 yılında Russel Eberhant ve James Kennedy tarafından önerilmiş kuş ve/veya balık sürülerinin sosyal davranışından esinlenen popülasyon temelli optimizasyon yöntemidir. İlk kez sürekli lineer olmayan fonksiyonları optimize etmek için kullanılmıştır (Eberhant ve Kennedy, 1995; Chandrasekaran ve ark., 2009).

Parçacık sürü optimizasyonu genel olarak kuş, balık ve hayvan sürülerinin besin kaynağı ararken ortaya koyduğu davranış üzerine kuruludur ve bu davranış biçimine sürü zekası ismi verilmektedir. Sürü zekası hiçbir kontrol mekanizması olmadan sürüdeki bireylerin birbirinin davranışından etkilenerek zekice hareket etmeleri ve kendi karşılaştıkları problemlere çözüm getirdikleri kolektif davranış biçimidir. Diğer evrimsel algoritmalarla karşılaştırıldığında sürü ile hareket eden algoritmaların bireysel hareket eden algoritmalara göre çözüm uzayına daha uygun bir şekilde yayıldığı gözlenmiştir (Gen ve Yu, 2010; Demirtaş, 2015; Alataş ve Varol, 2016).

PSO'nun ana mantığı bu konuda yapılmış iki çalışma ile yakından ilişkilidir: evrimsel algoritmalar ve yapay yaşam çalışmaları. Evrimsel algoritmalarda olduğu gibi optimize edilen amaç fonksiyonu geniş çözüm aralığında incelenir. Yapay yaşam çalışmalarında, yaşamın karakteristiği yapay sistemlerden tarafından incelenir. Millonas, yapay yaşam teorisi yardımıyla sosyal hayvanların davranışını incelerken

bilgisayarda toplumsal davranış içeren yaşam sistemlerinin nasıl oluşturulabileceğine dair 5 temel prensip ortaya koyar (Liu ve ark, 2017):

- Yakınlık: Ölçülebilirlik, yakınsama. Sürü küçük zaman ve yer aralıklarını başarıyla ölçebilmeli.
- Kalite: Sürü çevredeki kalite değişimini sezebilmeli ve buna tepki vermeli.
- Tepkide çeşitlilik: Sürü dar bir alandaki kaynakları elde etmek için yolunu sınırlandırmamalı.
- Kararlılık: Sürü davranış modunu her çevre değişikliğinde değiştirmemelidir.
- Uyarlanabilirlik: Sürü değişikliğe değer olduğunda davranış modunu değiştirmelidir.

PSO algoritmasında çözüm uzayındaki popülasyon “sürü”, her bir sürü üyesi “parçacık” olarak adlandırılır ve her bir parçacığın pozisyon bilgisi bir çözümü temsil eder. Parçacığın pozisyon değiştirme miktarı “parçacık hızı” olarak adlandırılır. Her bir parçacık kendi en iyi pozisyonu ve sürünün en iyi pozisyonuna sahip parçacığı referans alarak çözüme ulaşmaya çalışır (Shi ve ark., 2004; Ortakçı, 2011).

Her bir parçacık aşağıdaki özelliklere sahiptir (Ponnambalam,2009):

- Bir pozisyonu ve hızı vardır.
- Komşularını, en iyi pozisyonunu ve objektif fonksiyonun değerini bilir.
- Önceki en iyi konumunu hatırlar.

Sürü tipik olarak konumu ve hızı olan çok boyutlu uzayda parçacıklar tarafından modellenir. Bu parçacıklar iki temel akıl yürütme yeteneğine sahiptirler. Kendi en iyi konumları ve komşusunun en iyi konumunu bilirler. Bu sürüdeki üyeler, iyi pozisyonları birbirlerine aktarır ve bu iyi pozisyonlara dayanarak kendi konumlarını ve hızlarını ayarlar.

Her bir parçacığın hareket edebileceği olası üç yön vardır (Goldbarg ve ark., 2008; Ponnambalam ve ark, 2009) :

- Kendi yolunu takip eder.
- İterasyon sırasından sahip olduğu en iyi konuma (pbest) doğru hareket eder.
- En iyi parçacığın konumuna (gbest) hareket eder.

4.2. Standart PSO Algoritması

PSO algoritması rastgele üretilmiş parçacıklarla başlatılır. Her bir iterasyonda parçacıklar konumunu ve hızını güncelleyerek optimum çözüm noktasına ulaşmayı hedeflemektedir. Parçacığın şimdiye kadar ki en iyi değeri pbest (kişisel en iyi) olarak adlandırılır. Popülasyondaki tüm parçacıkların şimdiye kadar aldıkları en iyi değer ise global en iyi değerdir ve gbest olarak adlandırılır. Şekil 4.1.'de ki gibi sürüdeki her bir parçacık belirlenen iterasyon boyunca hızını ve konumunu sürekli güncelleyerek optimum çözüm noktasını veya çözüm kümesini bulmayı amaçlar (Ortakçı, 2011; Özkaya, 2011; Çayıroğlu, 2019).

M adet parametreden ve n adet parçacıktan oluşan optimizasyon probleminde her bir parçacığın çözüm belirtebilmesi için M boyutlu bir vektör ile gösterilmesi gerekmektedir. Pozisyon matrisi aşağıdaki gibi ifade edilir (Demirtaş, 2015):

$$X = \begin{bmatrix} X_{11} & X_{21} & \dots & X_{2M} \\ X_{21} & X_{22} & \dots & X_{2M} \\ \vdots & \vdots & & \vdots \\ X_{N1} & X_{N2} & \dots & X_{NM} \end{bmatrix} \quad (4.1)$$

Pozisyon matrisindeki i. satır i. parçacığın pozisyonunu temsil eder ve $X_i = [x_{i1}, x_{i2}, \dots, x_{im}]$ olarak ifade edilir. Aynı parçacığın iterasyonlar boyunca elde ettiği iyi değerini temsil eden p_{best} değeri (kişisel en iyi değeri) $p_i = [p_{i1}, p_{i2}, \dots, p_{im}]$ ile ifade edilir. Güncelleme işlemi eşitlik (4.2)'de görülmektedir (Özkaya, 2011).

$$p_{i(t+1)} = \begin{cases} p_i(t) & f(x_i(t+1)) \geq f(p_i(t)) \\ x_i(t+1) & f(x_i(t+1)) < f(p_i(t)) \end{cases} \quad (4.2)$$

Burada f enaz yapılan amaç fonksiyonunu temsil etmektedir. Enaz hedefi olan fonksiyonlar için daha küçük değere sahip olanlar en yüksek uygunluk değeri olanlardır.

Sürü içerisindeki parçacıkların en iyisine karşılık gelen değeri (G_{gbest}) ile gösterilmektedir. Her iterasyonda bu değer (4.3) eşitliği ile bulunmaktadır.

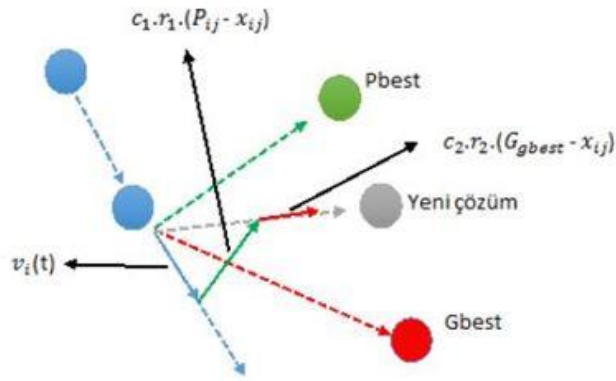
$$G(t) \in \{ p_1(t), p_2(t), \dots, p_m(t) \} \quad \text{I} \quad f(G(t)) = \min \{ f(p_1(t)), f(p_2(t)), \dots, f(p_m(t)) \} \quad (4.3)$$

Her bir parçacığın değişim vektörü yani hız vektörü ise $V_i = [V_{i1}, V_{i2}, \dots, V_{im}]$ olarak gösterilir. Belirtilen iterasyonlar boyunca güncellenen i . parçacığının hız vektörü ve buna bağlı olarak konum vektörü aşağıdaki gibidir (Eberhant ve ark, 2004):

$$V_{ij}^{t+1} = V_{ij}^t + c_1 \cdot r_1 \cdot (P_{ij} - x_{ij}) + c_2 \cdot r_2 \cdot (G_{gbest} - x_{ij}) \quad (4.4)$$

$$X_{ij}^{t+1} = X_{ij}^t + V_{ij}^t \quad (4.5)$$

Formülasyon (4.4)'de, t . iterasyonda i . parçacığın $(t+1)$. iterasyondaki hız vektörü bulunmuş olur. (4.5)' de ise V_{ij}^t hız vektörüne i . parçacığın t . zamanındaki konumu eklenerek $(t+1)$. iterasyondaki konumu bulunmuş olur. Algoritmadaki r_1 ve r_2 parametresi $[0,1]$ aralığında düzgün dağılıma ait rasgele sayılardır. c_1 ve c_2 öğrenme katsayısıdır ve $[0,2]$ arasında değer alır. Eberhant ve Kennedy 2 olarak alınmasını önermiştir. c_1 , parçacığın kendi hızına göre hareket etmesini, c_2 ise sürüdeki diğer parçacıkların tecrübelerine göre hareket etmesini sağlar. Bu parametreler Parçacık sürü algoritmasının rastgeleliğini ortaya koymaktadır (Ortakçı, 2011; Özkaya, 2011; Demirtaş, 2015).



Şekil 4.1. Parçacığın pozisyon deęiřtirmesi (Kavel,2009)

X_i^t = i. Parçacığının t. iterasyondaki konumu,

V_i^t = i. Parçacığının t. iterasyondaki hızı,

P_{best} = i.parçacığın sahip olduęu en iyi konumu,

G_{best} = sürüdeki en iyi konuma sahip olan parçacığın en iyi konumu

Çözüm uzayında parçacığın nasıl ilerleyeceęi konusu parçacığın kendi en iyi pozisyonu ve komşu en iyi pozisyonu arasındaki fark dikkate alınarak hız ve yön açıklığı kavuşturulacaktır. Bu durumda parçacığın komşuluęu tanımlanmalıdır. Literatürde sıklıkla kullanılan iki komşuluk yöntemi bulunmaktadır: gbest yöntemi ve lbest yöntemidir (Demirtaş, 2015; Talugder, 2011).

- Gbest yöntemi (global best): Her bir parçacık pozisyonunun tüm sürü içerisinde en iyi uygun parçacıktan etkilendięi bir yöntemdir. Sürüdeki tüm parçacıklardan elde edilen sosyal bilgilerin bulunduęu yıldız sosyal aę topolojisidir. Bu modelde parçacığın hız ve konum güncellemesi (4.4) ve (4.5) ile ifade edilir.
- lbest yöntemi (local best): Her bir parçacığın yalnızca izin verilen komşu parçacıklardan etkilenmesine izin verilir. Her bir parçacık için alt küme parçacıkları oluşturulur ve bunlar arasında en iyi performansa sahip olan seçilir. Sosyal bilgiler halka sosyal topolojisini yansıtır (Özkaya, 2011).

PSO algoritmasının genel işleyiři Tablo 4.1.'deki gibidir:

Tablo 4.1. PSO algoritmasının genel işleyişi (Demirtaş,2015)

```

PSO parametrelerinin belirlenmesi;
For Her Bir Parçacık İçin{
    Pozisyonlarını ve hızlarını belirle
}
End For
Do {
    For Her Bir Parçacık İçin{
        Uygunluk değerini hesapla;
        Kişisel en iyi ve global en iyi değerlerini güncelle;
    }
    End For
    For Her Bir Parçacık İçin{
        Parçacığın hızını hesapla
        Parçacığın pozisyonunu güncelle
    }
    End For
}
While {
    Maksimum iterasyon sayısına veya minimum hata koşulu sağlanana kadar
    devam et
}

```

4.3. Ayrık PSO Algoritması

Parçacık sürü optimizasyonu sürekli alanlardaki optimizasyonları çözmek için tasarlanmıştır. Algoritmanın başarısı ve sadeliğinden esinlenerek problemin dinamiği ayrık problemlerde uyarlamak için çeşitli yaklaşımlar önerilmiştir. Clerck (2011) PSO'nun (4.4) ve (4.5) hareket denklemlerini ayrık problemlere uyarlamak için aşağıdaki terimlerin veya yorumlamaların önermiştir (Hoffman ve ark, 2011).

- Parçacıkların konumunu şehirlerin π permütasyonundan oluşur. $\pi = (c_{i1}, c_{i2}, c_{i3} \dots c_{in})$ karşılık gelen tur $c_{i1} \rightarrow c_{i2} \rightarrow c_{i3} \dots \rightarrow c_{i1}$
- x ve y iki parçacık arasındaki fark transpozisyonun en kısa dizisi tarafından $T = (t_1, t_2, t_3, \dots t_n)$, ($yoT = x$) gibi temsil edilir. T kümesindeki her t değeri iki şehir arasındaki mesafeyi verir. Tur başlanılan noktaya gelinerek tamamlanır.
- Farkın uzunluğu transpozisyon T dizisinin uzunluğudur.(toplam rotalamanın uzunluğudur)
- Her bir tranpozisyon için öğrenme katsayısı ile çarpılır ve katsayımı sıfır ve 1 aralığında olması ona göre pozisyon alması beklenir.

- $T_1 = (t_1^1 + t_2^1 + t_3^1 + \dots + t_k^1)$ ve $T_2 = (t_1^2 + t_2^2 + t_3^2 + \dots + t_k^2)$ farkın toplamı $T_1 + T_2 = t_1^1 + t_2^1 + t_3^1 + \dots + t_k^1 + t_1^2 + t_2^2 + t_3^2 + \dots + t_k^2$ olarak ifade edilir.
- Farklılığın ve bir pozisyonun eklenmesi farklılığın transpozisyonlarına pozisyon uygulamaktır.

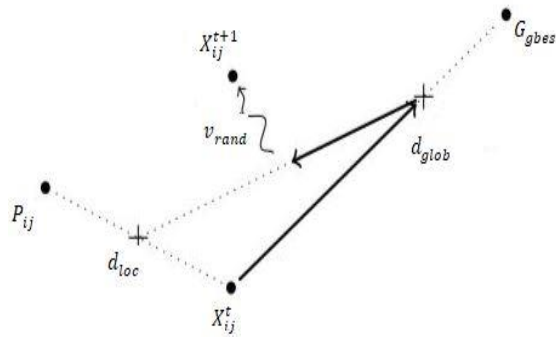
Hofmann ve ark (2011) çalışmasında atalet ağırlığına yer verilmiyor ancak (4.3) formülasyonuna öğrenme katsayıları ile hız vektörü ağırlıklandırılıyor. Lokal ve global iki çekici noktaya odaklandığından merkezler arasındaki farkı hesaplayarak ve onları yarıya kadar ölçeklendirerek ilk hedef kolayca hesaplanacaktır (Şekil 4.2.). Rasgele hareket ile sürünün çok hızlı bir şekilde birleşmesinin önüne geçiyor.

$$d_{loc} = X_{ij}^t + c_1 \cdot r_1 \cdot (P_{ij} - x_{ij})$$

$$d_{glob} = X_{ij}^t + c_2 \cdot r_2 \cdot (G_{gbest} - x_{ij})$$

$$v_{rand} = r_1 \cdot r_2 \cdot (P_{rand} - x_{ij})$$

$$X_{ij}^{t+1} = d_{glob} + \frac{1}{2} (d_{loc} - d_{glob}) + v_{rand}$$



Şekil 4.2. Merkez-tabanlı parçacık hareketi (Hoffman ve ark., 2011)

4.4. Parçacık Sürü Algoritması Parametreleri

PSO algoritması, performansı üzerinde etkisi olan çeşitli parametrelerden oluşur. Sürünün büyüklüğü, iterasyon sayısı, eylemsizlik katsayısı, öğrenme sabitleri ve hız vektörü algoritmanın temel parametrelerindedir (Eberhant ve Shi, 2004; Engelbrecht, 2007; Talukder, 2011; Demirtaş, 2015).

Sürü Büyüklüğü: Popülasyon büyüklüğü veya sürü büyüklüğü sürüdeki parçacık sayısıdır. Parçacık sayısı ne kadar fazla ise çözüm uzayı o kadar farklı koldan taranmaktadır ve her bir parçacık için hesaplamalar yapıldığından algoritmanın işleyişi süresi uzamaktadır. Bir dizi ampirik çalışmadan, PSO uygulamalarında sürü büyüklüğü için $n \in [20,60]$ aralığında kullanıldığı görülmüştür.

İterasyon Sayısı: İyi bir sonuç elde etmek için iterasyon sayısı probleme bağlıdır. Az sayıda iterasyon sayısı arama işlemini zamanından önce sonlandırırken, fazla sayıda iterasyon sayısı gereksiz hesaplama karmaşıklığına ve daha fazla zamana ihtiyaç duymaktadır.

Eylemsizlik Katsayısı (w): Eylemsizlik katsayısı Shi ve Eberhant tarafından literatüre kazandırılmıştır. Standart PSO'da eylemsizlik katsayısı, parçacığın bir önceki hızının mevcut hız üzerindeki etkisini kontrol edilmesi ve global olarak yayılmanın ve yerel bir noktada toplanmanın kontrolünün daha iyi yapılması amaçlamaktadır.

w değerinin yüksek seçilmesi parçacığın hız vektöründeki değişim miktarını arttırarak çözüm uzayının global arama yapmasını sağlamaktadır. Küçük seçilmesi ise hız vektöründeki değişim miktarını azaltarak yerel arama yapmasını sağlamaktadır. Xin ve diğerleri eylemsizlik katsayısının 0.9 ile 0.4 arasında doğrusal azalan değerler aldığı daha etkin olduğunu test etmişlerdir (Xin ve ark, 2009).

Hızlandırma (öğrenme) Sabitleri (c_1 ve c_2): Parçacıkların hız vektörlerinin güncellenmesinde kullanılan sabitlerdir. r_1 ve r_2 değeriyle beraber kullanılarak parçacık hızının bilişsel ve sosyal bileşenlerin stokastik etkisini korurlar. Hızlandırma sabitlerinden c_1 , parçacığın kendi geçmişinden öğrendiği bilgiyi c_2 ise komşularının geçmişinden öğrendiği bilgiyi göstermektedir. Genellikler $c_1 = c_2 = 2$ olarak seçilmektedir ve algoritma sonuna kadar değişmemektedir.

Hız vektörü: Parçacığın bir sonraki konumunun belirlenmesinde kullanılır. Hız vektörünün adımlarının büyük olması, parçacığın iki konumu arasındaki mesafeyi

artırmaktadır ve bu da o aralıkta iyi bir çözüm noktası varsa gözden kaçırılmasına sebep olur. Diğer taraftan yavaş ilerlemesi bir noktanın etrafında çok fazla zaman harcamasına sebep olur.

Her bir hız vektörünün bileşenleri için $[-v_{max}, v_{max}]$ sınırlarının olması gerekmektedir. Eğer hız vektörü güncelleme esnasında belirlenen maksimum hızı aşarsa parçacığın hızı maksimum hıza ayarlanır. Veya belirlenen alt sınırın altında kalıyorsa alt sınır değerine ayarlanır.

$$v_{ij}^{t+1} = \begin{cases} v_{ij}^{t+1} & \text{if } v_{ij}^{t+1} < v_{max} \\ v_{max} & \text{if } v_{ij}^{t+1} \geq v_{max} \end{cases}$$

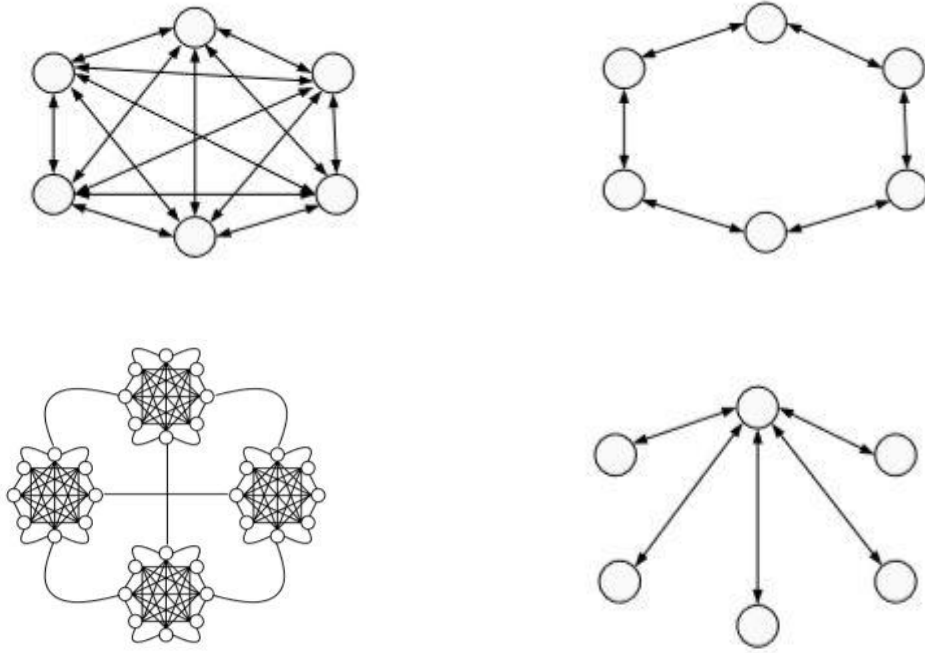
(4.6)

4.5. Parçacık Sürü Algoritmasının Sosyal Ağ Yapıları

Parçacık sürü algoritması sosyal davranışlardan esinlenerek oluşturulmuştur. Bu yapıda parçacıklar komşuluğunda olan diğer parçacıkların en iyi durumu hakkında bilgi alışverişinde bulunarak iletişim kullanırlar. Daha sonra en iyi pozisyonda olan parçacığa doğru hareket eder. Genel olarak PSO performansı sosyal yapı ile ilişkilidir. Bu sosyal ağ arasındaki ilişki genel olarak düğümler arasındaki bağlanma derecesi, kümelenme miktarı ve düğümler arasındaki mesafe ile ilişkilidir (Engelbrecht, 2007; Onwunalu, 2010; Talugder, 2010).

- Yıldız topolojisi: Sadece bir komşuluğu vardır ve her parçacık tüm parçacıklar ilişkilidir. Gbest algoritması da denir ve hızlı yakınsama durumundan dolayı yerel minimum noktasında takılması durumu meydana gelmektedir.
- Halka topolojisi: Her bir parçacık iki parçacık ile bağlantılıdır. Komşuluğunda toplamda üç parçacık olmaktadır. Lokal (yerel) en iyi problemleri de denir.

- Dört Küme topolojisi: Kümeler arasında iki bağlantı ile 4 kümeden oluşmaktadır. Bu tür problemlere lokal (yerel) en iyi problemleri de denir. Bir küme içerisindeki parçacıklar beş küme ile bağlantılıdır.
- Tekerlek topolojisi: Bir parçağın tüm parçalara bağlandığı ve tüm bilgilerin bu parçacık yolu ile iletildiği topolojidir. Tüm parçacıkların en iyi pozisyonları değerlendirilip konumunu en iyi olana doğru ayarlar.



Şekil 4.3. PSO ağ topolojileri (Talugder, 2011)

4.6. Parçacık Sürü Algoritmasının Avantajları ve Dezavantajları

Parçacık sürü algoritmasının avantajları şunlardır (Talugder, 2011):

- Uygulaması kolaydır, hem akademik çalışmalarda hem de mühendislik çalışmalarında kullanılmaktadır.
- Sınırlı sayıda parametreleri vardır ve parametrelerin çözümlere olan etkisi diğer optimizasyon tekniklerine kıyasla daha küçüktür.
- PSO algoritmasındaki hesaplama çok basittir.
- Problemin optimum değerini ve yakınsama değerini kısa sürede hesaplayan bazı teknikler vardır.
- Diğer optimizasyon tekniklerine göre başlangıç noktasına daha az bağımlıdır.

- Kavramsal olarak basittir.

Parçacık sürü algoritmasının dezavantajları şunlardır:

- Diğer buluşsal optimizasyon teknikleri gibi parçacık sürü algoritmasının en büyük dezavantajı, ilgili teorilerin gelecekteki gelişiminde üstesinden gelinmesi gereken analiz için sağlam bir matematik temeli olmamasıdır.
- Parçacık sürü algoritması, matematiksel yaklaşımlara nispeten daha uzun hesaplama zamanı gerektir.

4.7. Parçacık Sürü Algoritmasının Uygulama Alanları

Parçacık sürü algoritmasının kompleks modellerde uygulama kolaylığı, basit olması ve maliyet avantajı nedeniyle sıkça karşımıza çıkmaktadır. Literatürde PSO tabanlı algoritmalar ile birçok alanda problemleri çözüm getirilmiştir. Bu çalışmalardan bir kısmı tablo Tablo 4.2.'de verilmiştir.

Tablo 4.2. PSO ile ilgili yapılan çalışmalar

Yazarlar	Yapılan Çalışma
Sakri ve ark (2018)	Meme kanserinin nüksetmesi tahmininde PSO kullanılmıştır.
Cansız ve Göçmen (2018)	Gezgin satıcı problemlerinde PSO algoritması kullanılmıştır.
Hannan ve ark (2018)	Katı atıklarını toplama ve rotalama optimizasyonu için PSO algoritması kullanılmıştır.
Saplıoğlu ve ark. (2018)	Çatılarda biriken yağmur suyunu depolayacak tankın kapasitesinin belirlenmesinde PSO kullanılmıştır.
Aydın ve ark (2018)	Duygu analizinde çoklu popülasyon temelli PSO kullanılmıştır.
Dereli ve Köker (2018)	Robot kontrolünün temelinde olan ters kinematik çözümün bulunması için PSO kullanılmıştır.
Dou ve ark. (2017)	Montaj hattı dengelemede PSO algoritması kullanılmıştır.
Kazemi ve ark. (2016)	İran'da konut ve ticari sektörlerinde enerji talebi tahmininde en iyi senaryoyu seçmek için PSO kullanılmıştır.
Gomes,2015	Çeyrek araç pasif süspansiyon tasarım sistemlerinin çok amaçlı optimizasyonu için PSO tabanlı algoritma kullanılmıştır.
Liu ve ark(2011)	Tersine lojistik modelinde rotalama ve yerleştirme probleminin çözümünde PSO kullanılmıştır.

BÖLÜM 5. UYGULAMA

Bu bölümde, catering firması için yerleştirme rotalama problemi parametre değerleri değiştirilerek elde edilen sonuçlar analiz edilecek ve parçacık sürü algoritması (PSO) ile çözüm bulunacaktır.

5.1. Catering Firmasında Yerleştirme Rotalama Problemi

İstanbul Anadolu yakasında faaliyet gösteren bir catering firması sanayi bölgesindeki farklı kapasitede talepleri olan 100 müşteri için daha iyi hizmet verebilmek ve maliyet minimizasyonu sağlayabilmek adına hem rotalama hem de lokasyon anlamında iyileştirme yapmak istiyor. Alternatif 8 iş yeri bulunmaktadır.

Problemi çözerken Matlab R2014a kullanılarak parçacık sürü algoritması ile çözüm aranmıştır. Çözüm kümesi 1'den müşteri sayısı+ tesis sayısı kadar permütasyon sıralamasından oluşuyor. Parçacıklar pozisyon değiştirirken, lokal ve global en iyi çözüm kümesine yönelmesi mevcut çözümün lokal ve global çözüme ne kadar benzediğine göre hesaplanır. Yani fonksiyon girdi olarak mevcut çözümü, global en iyi ve lokal en iyi çözümü alır, global ve yerel en iyi çözümler mevcut çözüm üzerine etki eder ve permütasyon sırasını değiştirir ve yeni çözüm hesaplanır.

5.1.1. Kullanılan parametreler ve veri seti

Parçacık sürü algoritmasında kullanılan parametreler için literatürde önerilen değer aralıkları kullanılarak çözüme aranmıştır.

- Çözümlerin kodlanması permütasyon PSO kodlama ile yapılmıştır.
- Başlangıç hızı rassal olarak verilmiştir.

- $c_1=c_2=0,2$ değeri verilmiştir.
- $r_1=r_2=[0,1]$ arasında rassal üretilmiştir.
- $w=0$ olarak tanımlanmıştır.
- Parçacık sayısı 10, 20 ve 30 olmak üzere 3 şekilde de çözüm aranmıştır.
- Programı sonlandırma kriteri olan iterasyon sayısı 20 olarak belirlenmiştir. (10 iterasyonda değer değişmeyecek ise durdurma komutu da dahil edilmiştir.)

100 müşteri ve tesis lokasyonları, tesis kapasiteleri, değişken maliyetler ve diğer parametre değerleri Tablo 5.1. ve Tablo 5.2.'de verilmiştir.

Tablo 5.1. Müşterilere ait koordinatlar ve talep miktarları

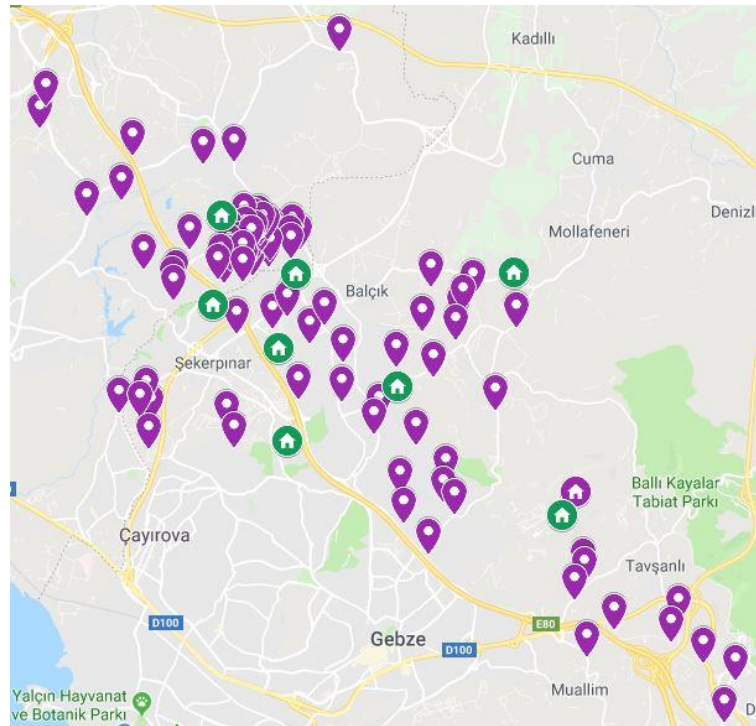
Müşteri No	Müşteri No	x koordinatı	y koordinatı	Talep miktarı	x koordinatı	y koordinatı	Talep miktarı
1	51	40,88462	29,40837	85	40,84185	29,37074	45
2	52	40,885	29,41009	95	40,8422	29,39421	47
3	53	40,8838	29,41241	45	40,85167	29,36992	80
4	54	40,88571	29,41033	44	40,84876	29,36833	33
5	55	40,88478	29,40511	20	40,86391	29,41535	36
6	56	40,88352	29,40582	40	40,86705	29,40507	69
7	57	40,88346	29,4096	50	40,86981	29,40911	74
8	58	40,88187	29,41017	60	40,8662	29,39485	40
9	59	40,88127	29,4027	85	40,86799	29,41934	19
10	60	40,8809	29,4019	100	40,86023	29,42452	24
11	61	40,8802	29,40376	120	40,85245	29,41225	26
12	62	40,8784	29,3996	42	40,85189	29,42425	71
13	63	40,88072	29,39933	56	40,84822	29,43443	150
14	64	40,88136	29,40435	95	40,84262	29,44471	150
15	65	40,88377	29,40201	28	40,84513	29,43302	140
16	66	40,88474	29,39832	33	40,83261	29,44018	160
17	67	40,88482	29,4015	36	40,82634	29,44129	240
18	68	40,88314	29,39819	90	40,81976	29,4482	120
19	69	40,88571	29,4037	57	40,83528	29,45311	146
20	70	40,88666	29,399	64	40,83081	29,4523	135
21	71	40,88771	29,40102	48	40,82823	29,45529	140
22	72	40,88721	29,39992	74	40,90219	29,39424	120
23	73	40,88625	29,40204	66	40,90162	29,38556	140
24	74	40,88432	29,39529	65	40,89401	29,36304	170
25	75	40,88466	29,39976	40	40,89078	29,35344	156
26	76	40,88608	29,39611	30	40,90339	29,36633	157
27	77	40,88514	29,39461	45	40,90911	29,34029	150
28	78	40,88808	29,39702	30	40,91375	29,34218	164

29	79	40,88441	29,40031	35	40,92531	29,42352	154
----	----	----------	----------	----	----------	----------	-----

Tablo 5.1. (Devamı)

30	80	40,88422	29,39796	20	40,85922	29,43935	136
31	81	40,88536	29,39756	24	40,86887	29,45684	114
32	82	40,8847	29,3931	74	40,85702	29,44956	110
33	83	40,88227	29,39599	69	40,87402	29,46055	180
34	84	40,88351	29,39905	85	40,8666	29,44653	146
35	85	40,8817	29,39287	40	40,87101	29,45775	56
36	86	40,87866	29,39752	12	40,87607	29,44905	59
37	87	40,87663	29,39169	14	40,86738	29,47253	42
38	88	40,87942	29,39335	100	40,86484	29,45583	40
39	89	40,88029	29,39656	120	40,85013	29,46672	41
40	90	40,87954	29,39057	140	40,83335	29,48919	36
41	91	40,87758	29,38986	250	40,81562	29,49117	78
42	92	40,8769	29,39686	60	40,81377	29,49151	55
43	93	40,87633	29,37825	80	40,81027	29,48873	20
44	94	40,87485	29,37706	90	40,80391	29,49958	90
45	95	40,87967	29,36934	105	40,579842	29,49203	125
46	96	40,88378	29,38207	60	40,80588	29,51758	140
47	97	40,87298	29,37754	47	40,80133	29,51557	60
48	98	40,84948	29,36227	36	40,78461	29,53039	55
49	99	40,84785	29,37116	39	40,79701	29,52433	60
50	100	40,84657	29,3924	44	40,79325	29,53345	87

Müşteri ve tesislerin bulunduğu yer Google map üzerinde Şekil 5.1.'de gösterilmiştir



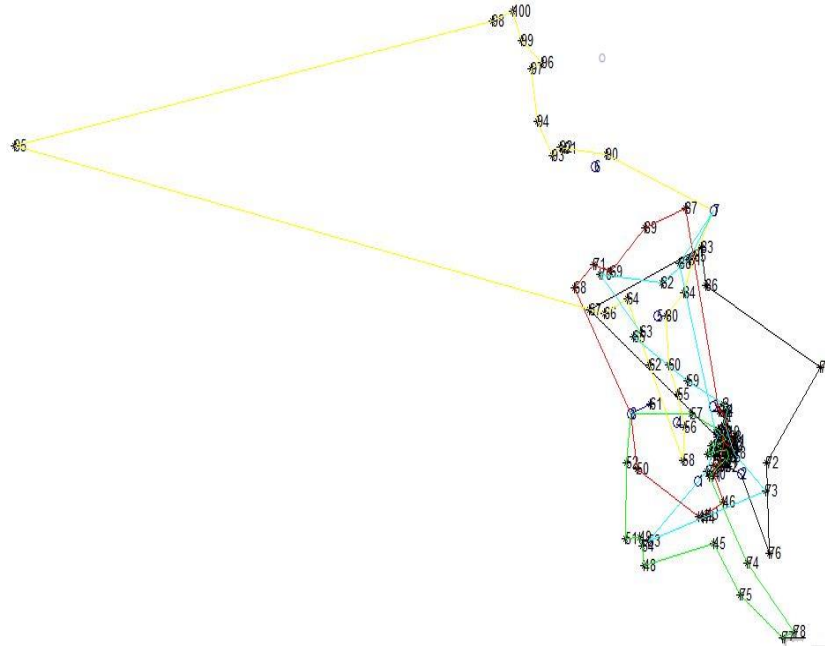
Şekil 5.1. 100 müşteri ve 8 tesisin Google map üzerindeki noktaları

Tesislerin koordinatları, kurulum maliyetleri, tesis kapasitesi ve her bir tesisteki hizmetin birim maliyeti Tablo 5.2.'de verilmiştir.

Tablo 5.2. Tesislere ait koordinatlar ve talep miktarları

Tesis no	x koordinatı	y koordinatı	Tesis kurulum maliyeti (TL)	Tesis kapasitesi (kişi sayısı)	Birim maliyet(TL/kişi)
1	40,8727	29,38865	128000	1200	5,75
2	40,89135	29,39084	120000	1800	5,4
3	40,87915	29,41161	135000	1250	5,7
4	40,8635	29,40674	140000	2400	5,42
5	40,8554	29,4395	125000	1500	5,4
6	40,82837	29,48538	130000	2350	5,4
7	40,87937	29,47198	115000	3200	5,1
8	40,84392	29,40907	118000	3600	5

Araç kapasitesinin 1800 kişiye yemek verebilecek kapasite olduğu biliniyor ve 10, 20 ve 30 parçacık için PSO algoritması kullanılarak çözüm aranmıştır. 10 parçacıkla çözülen modelde 3 tesis açıldığı görülmüştür. 8., 2. ve 7. tesisin açıldığı toplam maliyetin ise 394204,5496 TL olduğu görülüyor (Tablo 5.3.). Rotalama ise Şekil 5.2. ve Tablo 5.3. verilmiştir.



Şekil 5.2. 100 müşteri ve 8 tesisin çözümü

Tablo 5.3. 10 parçacık sayısına göre rotalama ve toplam maliyet

Tesis	Rotalama	Toplam Maliyet (TL)
8	8-68-71-69-89-87-8-7-19-23-21-22-20-25-29-15-34-18-35-38-40-46-43-44-47-50-52-51-49-54-48-45-75-77-78-74-41-24-27-26-31-16-30-36-42-12-13-17-57-61-8	394204,5496
2	2-32-33-39-10-67-83-86-79-72-76-2	
7	7-90-91-92-93-94-97-96-99-100-98-95-66-64-62-58-56-55-60-80-84-85-81-82-70-63-65-59-3-4-2-1-6-5-28-73-53-37-9-11-88-7	

PSO algoritması 20 iterasyon için 20 parçacık ile problem çözüldüğünde Tablo 5.4.'deki gibi sonuçlar bulunmuştur. 3 tesis açılmıştır ve toplam maliyet olarak 394205,9802 bulunmuştur.

Tablo 5.4. 20 parçacık sayısına göre rotalama ve toplam maliyet

Tesis	Rotalama	Toplam Maliyet (TL)
8	95-98-100-99-94-92-93-71-70-66-64-88-86-60-59-3-7-14-21-73-42-56-57-11-8-2-4-79-6-5-19-20-30-17-15-10-36-27-32-38-40-43-45-48-53-49-62-65	394205,9802
2	76-78-77-75-54-51-50-52-90-89-81-84-29-25-24	
7	72-74-44-47-46-33-22-28-26-31-16-34-18-39-41-37-58-80-82-85-87-83-96-97-91-68-69-63-67-61-55-13-2-9-23-1	

PSO algoritması 20 iterasyon için 30 parçacık ile problem çözüldüğünde Tablo 5.5.'deki gibi sonuçlar bulunmuştur. 3 tesis açılmıştır ve toplam maliyet olarak 394205,1357 bulunmuştur.

Tablo 5.5. 30 parçacık sayısına göre rotalama ve toplam maliyet

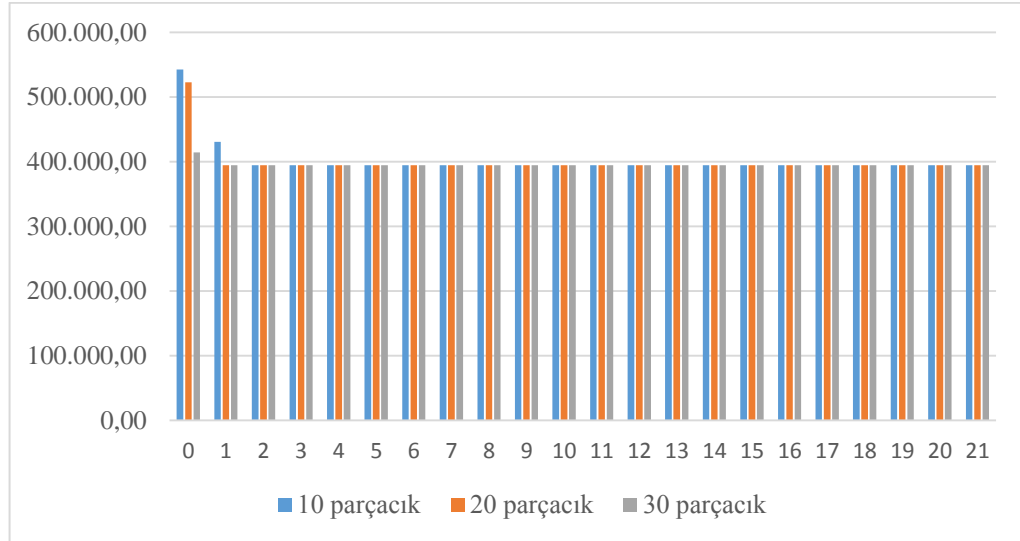
Tesis	Rotalama	Toplam Maliyet (TL)
8	48-75-77-78-76-73-45-47-44-43-46-41-37-39-18-34-13-15-5-19-21-25-38-35-24-27-26-28-72-74-53-54-49-67-71-70-69	394205,1357
2	79-4-1-7-2-3-57-56-6-17-20-16-32-40-42-58-51-52-50-55-59-60-62-65-80-86-85-83-81-88-84-82-89-95-98-100-96-97-94-93-92-91-90-87	
7	30-12-36-33-10-99-66-64-63-61-8-14-11-9-29-23-31	

Her bir parçacık sayısına göre toplam maliyetin iterasyonlara göre sonuçları Tablo 5.6.'da ve Tablo 5.7.'de verilmiştir.

Tablo 5.6. Parçacık sayısına göre toplam maliyet

İterasyon	10 parçacık	20 parçacık	30 parçacık
0	542348,20	522488,80	414304,80
1	430281,77	394236,17	394227,54
2	394217,49	394215,01	394216,92
3	394212,38	394212,24	394208,67
4	394212,38	394211,96	394208,67
5	394209,26	394211,47	394208,67
6	394209,01	394211,42	394207,69
7	394208,39	394209,00	394207,28
8	394208,28	394208,63	394207,28
9	394208,28	394208,45	394206,72
10	394208,28	394208,44	394206,69
11	394208,28	394206,59	394205,64
12	394208,03	394206,38	394205,45
13	394208,03	394206,33	394205,36
14	394206,91	394206,27	394205,36
15	394206,91	394206,05	394205,17
16	394206,91	394206,05	394205,17
17	394204,92	394206,01	394205,14
18	394204,92	394206,01	394205,14
19	394204,77	394206,01	394205,14
20	394204,61	394206,01	394205,14
21	394204,55	394205,98	394205,14

Tablo 5.7. Parçacık sayısı ve iterasyona göre toplam maliyet



Parçacık sayısı artınca ilk iterasyonda toplam maliyetin daha az olduğu görülmüştür ve Tablo 5.7.'de iterasyonlar boyunca elde edilen sonuçlar incelenmiştir. Sonuç olarak 100 müşteri ve 8 potansiyel tesis için 21. iterasyon sonunda toplam maliyetin birbirine benzer çıktığı görülmüştür.

BÖLÜM 6. SONUÇ

Yerleşim rotalama problemi ile belirli müşteri kümesi ve potansiyel tesisler baz alınarak, hem potansiyel tesislerden hangisinin açılacağına kararını verir hem de açılacak tesislere atanan müşterinin hangi rotalama ile hizmet alması gerektiğinin kararını minimum maliyet oluşturacak şekilde optimize edilmeye çalışılır. Aynı anda yerleşim ve rotalama problemine tek bir modelde çözüm bulması problemi önemli kılamaktadır.

Bu tez çalışmasında, Daskin ve Perl(1985)'in modeline istenen amaç doğrultusunda kısıtlar eklenerek, yeni bir matematiksel model elde edilmiş ve her bir denklem seti, kısıtlar ve optimizasyon denklemleri tek tek birbiri ile nasıl ilişkili olduğu incelenmiştir. Ancak modeldeki değişken sayısı arttıkça kısıt sayısı üssel olarak arttığı için kesin çözüm yöntemleri ile sonuca ulaşmak mümkün olmamaktadır. Bu nedenle çalışmada metasezgisel yöntemlerden biri olan Parçacık Sürü Algortimasına değinilerek İstanbul'da faaliyet gösteren bir cathering firması üzerinden 100 müşteri grubuna karşılık 8 potansiyel tesis yeri için çözüm bulunmuştur. Bu metasezgisel yöntemin uygulama kolaylığının olması, az parametre değerine sahip olması ve yakınsama değerine ve optimum değere hızlı ulaşması sebebi ile bu uygulamada yer verilmiştir. Bu çalışmada vurgulanmak istenen, hem rotalama hem de yerleştirme probleminin bir matematiksel modelde çözüm bulması ve aynı zamanda rotalama faaliyetini de kapsayarak maliyet minimizasyonunun sağlanmasıdır.

Çalışmanın ilerleyen süreçlerinde diğer metasezgisel yöntemler kullanılarak her bir algoritmanın etkinliklerin daha büyük model ve iterasyon ile çözümlerin karşılaştırılması öngörülmektedir.

KAYNAKLAR

- Ahn, J. 2018. The generalized location routing problem with profits for planetary surface exploration and terrestrial applications. Massachusetts Institute of Technology, PhD Thesis.
- Akpınar, F. 2009. Yerleştirme ve rotalama problemi için bir genetik algoritma. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Aydın, İ. ve ark. 2018. Duygu analizi için çoklu popülasyon tabanlı parçacık sürü optimizasyonu. Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, 11(1), 52-64.
- Atasagun, G. C. 2015. Zaman Bağımlı Eş Zamanlı Topla Dağıt Araç Rotalama Problemi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi.
- Alparslan, M. 2015. Araç Rotalama Problemi için Matematiksel Modeller ve Çözüm Yaklaşımı. Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi.
- Alataş, B., Varol, E. 2016. Sürü zekasında yeni yaklaşım: Kuş sürüsü algoritması. Dicle Üniversitesi, Mühendislik Fakültesi Dergisi, 8(1):133-146.
- Barbati, M. 2013. Model and Algorithms for Facility Location Problems with Equity Considerations. University of Naples Federico II, Department of Industrial Engineering, Phd Program in Science and Technology Managements XXV Cycle, PhD Thesis.
- Basti, M. ve Özçakar, N. 2012. p-medyan Kuruluş Yeri Seçimi Probleminin çözümünde Parçacık Sürü Algoritması. İstanbul Üniversitesi İşletme Dergisi Fakültesi, 41(2): 241-257.
- Basti, M. 2012. P-medyan Tesis Yeri Seçim Problemi ve Çözüm Yaklaşımlar. AJIT-e: Online Academic Journal of Information Technology, 3(7):48-75.
- Berger, R. T., Daskin, M. S., Snyder L.V. 2005. Logistics systems design and optimization, Facility location in supply chain design. Springer Science+Business Media, Inc., 39-65.

- Bouhafs, L., Hijjam, A., Koukam, A. 2008. A tabu search and ant colony system approach for the capacitated location-routing problem. Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, IEEE computer society, 47-50.
- Cansız, Ö. F., Göçmen, S. 2018. Distance analysis of multimodal transportation based on traveling salesman problem with particle swarm optimization method. International Journal of Advanced Engineering Research and Science, 5(6),1-6.
- Chandrasekaran, C. ve ark. 2009. Particle Swarm Optimization. Discrete Particle Swarm Optimization Algorithm for Flowshop Scheduling. InTech, Austria, 397-422
- Chen, C., Ting, C. 2013. A multiple ant colony optimization algorithm for the capacitated location routing problem. International Journal of Production Economics, 141(1), 34-44.
- Çalışkan, K. 2011. Karınca Kolonisi Optimizasyonu ile Araç Rotalama Probleminin Maliyetlerinin Kümeleme Tekniği ile İncelenmesi, TOBB Ekonomi ve Teknoloji Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayara Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi.
- Çeyrekoğlu, S. 2017. Araç Rotalama Problemine Genetik Algoritma Yaklaşımı ve Örnek Bir Uygulama, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi.
- Dökeroğlu, T. 2017. Karesel Atama Problemi için Yeni Bir Özuyarlamalı Paralel Güçlü Tabu-arama Algoritması, Pamukkale Üniversitesi Mühendislik Bilim Dergisi, 23(5):559-565.
- Dursun, P.2009. Zaman Pencere Araç Rotalama Probleminin Genetik Algoritma ile Modelenmesi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri MÜhendisliği Anabilim Dalı, Yüksek Lisans Tezi.
- Derbel ve ark., 2012. Genetic algorithm with iterated local search for solving a location- routing problem. Expert systems with applications, 39, 2865-2871.
- Dereli, S., Köker,R. 2018. IQ-PSO approach to the inverse kinematics problem solution of a 7-dof serial robot manipulator, Sigma Journal of Engineering and Natural Sciences, 36(1), 77-85.
- Dou, J. ve ark. 2017. A novel discrete particle swarm algorithm for assembly line balancing problems. Assembly Automation, 37(4), 452-463.
- Demirtaş, Y.E. 2015. Dinamik araç programlama problemine parçacık sürü optimizasyonu algoritması çözüm önerisi. İstanbul Üniversitesi, Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı, Doktora Tezi.

- Düzakın, E ve Demircioğlu M. 2009. Araç rotalama problemleri ve çözüm yöntemleri, Çukurova Üniversitesi İİBF Dergisi, 13(1):68-87.
- Eiselt, H. A. 2004. Decision Analysis, Location Models, and Scheduling Problems. Fundamentals of Location and Layout Problems. Springer-Verlag Berlin Heidelberg, 153-167.
- Eberhart, R., Kennedy, J. 1995. Particle Swarm Optimization. IEEE International Conference on Neural Network, Washington DC, 1942-1948.
- Engelbrecht, A. P. 2007. Computational Intelligence, Second Edition, 1-581.
- Golbarg, E. ve ark. 2008. Travelling salesman problem. Particle swarm optimization algorithm for the travelling salesman problem, IntechOpen, 75-96
- Gomes, H. M. 2015. Multi-objective optimization of quarter car passive suspension design in the frequency domain based on PSO. International Journal for Computer Aided Engineering and Software, 33(5), 1422-1434.
- Gen, M., Yu, X. 2010. Introduction to Evolutionary Algorithm, Springer-Verlag London Limited, 1-433.
- Hannan, Ö.A. ve ark. 2018. Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm. Waste Management, 71, 31-41.
- Hassanzadeh ve ark. 2009. Facility Location: Concepts, Models, Algorithms and Case Studies, Contributions to Management Science. Location-Routing Problem, Physica-Verlag Heidelberg, 395-317.
- Kaveh, A., 2017. Advances in Metaheuristic Algorithms for Optimal Design of Structure, Springer International Publishing, 1-426.
- Kazemi, A. ve ark., 2015. Evaluating the performance of genetic and particle swarm optimization algorithms to select an appropriate scenario for forecasting energy demands using economic indicators: residential and commercial sectors of Iran. 6(4), 345-355.
- Kısakol, B. E. 2015. Acil Sağlık İstasyonlarının Yerleşim Probleminin Küme Kapsama ve Medyan Modeli Yaklaşımı ile Değerlendirilmesi: Manisa ili Merkezi için Bir Uygulama. Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi.
- Kızılateş, G. ve Nuriyeva, F. 2016. Gezgin Satıcı Problemi için Merkezden Kenarlara Hipersezgisel Yöntem, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 20(2):319-323.

- Küçükdeniz, T. 2009. Sürü zekası optimizasyon tekniği ve tedarik zinciri yönetiminde bir uygulama. İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Liu ve ark. 2011. Research on location-routing problem of reverse logistics with grey recycling demands based on PSO. *Grey Systems: Theory and application*, 1(1), 97-104.
- Lee, V.F., Lin, S., Ting, C. 2010. A simulated annealing heuristics for the capacitated location routing problem. *Computers & Industrial Engineering*, 58(2), 288-299.
- Marikakis, Y. 2009. *Encyclopedia of Optimization. Location Routing Problem. Second Edition*, Springer Science + Business Media, USA, 1919-1925.
- Nagy, G., Salhi, S. 2007. Location-routing: Issues, models and methods. *European Journal of Operation Research*. 177:649-672.
- Nadizadeh ve ark., 2011. Using greedy clustering method to solve capacitated location-routing problem. *African journal of business management*, 5(17), 7499-7506.
- Ortakçı, Y. 2011. Parçacık sürü optimizasyonu yöntemlerinin uygulamalarla karşılaştırılması. Karabük Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği, Yüksek Lisans Tezi.
- Oost, V. H. 2015. *Cooperative Game Theory in Location-Routing*, Université Catholique de Louvain, Louvain School of Management and Norwegian School of Economics, Master Business Engineering, Master Thesis.
- Özkaya, U. 2011. Parçacık sürü algoritmalarının mikrodalga kuvvetlendirici uygulamaları. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı Haberleşme Programı, Doktora Tezi.
- Ponboon, S. 2015. Exact solutions for location-routing problems with the time window using branch and price method. Kyoto University, PhD Thesis.
- Ponnambalam, S.G. ve ark. 2009. Discrete particle swarm optimization algorithm for flowshop scheduling. *Particle swarm optimization. InTech*, 397-422.
- Saplıoğlu, K. ve ark. 2018. Determining rainwater harvesting storage capacity with particle swarm optimization. *Preprints*, 1(2), 1-17.
- Sakri ve ark. 2018. Particle swarm optimization feature selection for breast cancer recurrence prediction. *Special Section on Big Data Learning and Discovery, IEEE Access*, 29637-29647.
- Schneider, M., Drexl, M. 2017. A survey of the standard location-routing problem. *M. Ann Operation Research. Volume 259*: 389-414.

Shi, Y. ve ark. 2004. Recent Advances in Particle Swarm. Proceeding of the 2004 congress on evolutionary computation. 1:90-97.

Talukder S. 2011. Mathematical modelling and applications of particle swarm optimization. School of Engineering Blekinge Institute of Technology, Department of Mathematics and Science, Master's Thesis.

Yiğit, V. ve Türkbey O. 2003. Tesis yerleşim problemlerine sezgisel metotlarla yaklaşım. Gazi Üniversitesi Mimarlık Fakültesi Dergisi, 18(4):45-56.

Xin, J. ve ark. 2009. A particle swarm optimizer with multi-stage linearly-decreasing inertia weight. 2009 International Joint Conference on Computational Sciences and Optimization, CSO 2009, China, 505-508.

<http://www.ibrahimcayiroglu.com/Dokumanlar/IleriAlgoritmaAnalizi/IleriAlgoritmaAnalizi-10.Hafta-ParcacikSuruAlgoritmasi.pdf>, Erişim Tarihi: 29.04.2019.

EKLER

EK 1: Matematiksel Modelin Açılımı

3 müşteri grubu ve 2 potansiyel tesis kümesi için matematiksel modelin LINGO'da açık formda yazımını aşağıdaki gibidir.

$$\begin{aligned} \text{min} &= 30000 * x_a + 20000 * x_b \\ &+ 5.2 * (50 * y_{1a} + 50 * y_{2a} + 40 * y_{3a}) \\ &+ 5.8 * (50 * y_{1b} + 50 * y_{2b} + 40 * y_{3b}) \\ &+ 5.5 * (0 * z_{111} + 5 * z_{121} + 6 * z_{131} + 5 * z_{1a1} + 5 * z_{1b1} \\ &+ 5 * z_{211} + 0 * z_{221} + 7.8102496 * z_{231} + 9.4868329 * z_{2a1} + 3.162277 * z_{2b1} \\ &+ 6 * z_{311} + 7.8102496 * z_{321} + 0 * z_{331} + 5 * z_{3a1} + 5 * z_{3b1} \\ &+ 5 * z_{a11} + 9.4868329 * z_{a21} + 5 * z_{a31} + 0 * z_{aa1} + 8 * z_{ab1} \\ &+ 5 * z_{b11} + 3.162277 * z_{b21} + 5 * z_{b31} + 8 * z_{ba1} + 0 * z_{bb1}); \\ \\ &+ 5.5 * (0 * z_{112} + 5 * z_{122} + 6 * z_{132} + 5 * z_{1a2} + 5 * z_{1b2} \\ &+ 5 * z_{212} + 0 * z_{222} + 7.8102496 * z_{232} + 9.4868329 * z_{2a2} + 3.162277 * z_{2b2} \\ &+ 6 * z_{312} + 7.8102496 * z_{322} + 0 * z_{332} + 5 * z_{3a2} + 5 * z_{3b2} \\ &+ 5 * z_{a12} + 9.4868329 * z_{a22} + 5 * z_{a32} + 0 * z_{aa2} + 8 * z_{ab2} \\ &+ 5 * z_{b12} + 3.162277 * z_{b22} + 5 * z_{b32} + 8 * z_{ba2} + 0 * z_{bb2}); \\ \\ &z_{111} + z_{121} + z_{131} + z_{1a1} + z_{1b1} + z_{112} + z_{122} + z_{132} + z_{1a2} + z_{1b2} = 1; \\ &z_{211} + z_{221} + z_{231} + z_{2a1} + z_{2b1} + z_{212} + z_{222} + z_{232} + z_{2a2} + z_{2b2} = 1; \\ &z_{311} + z_{321} + z_{331} + z_{3a1} + z_{3b1} + z_{312} + z_{322} + z_{332} + z_{3a2} + z_{3b2} = 1; \\ \\ &50 * (z_{111} + z_{121} + z_{131} + z_{1a1} + z_{1b1}) + 50 * (z_{211} + z_{221} + z_{231} + z_{2a1} + z_{2b1}) \\ &+ 40 * (z_{311} + z_{321} + z_{331} + z_{3a1} + z_{3b1}) \leq 100; \end{aligned}$$

$$50*(z112+z122+z132+z1a2+z1b2)+50*(z212+z222+z232+z2a2+z2b2) \\ +40*(z312+z322+z332+z3a2+z3b2)\leq 100;$$

$$za11+za21+za31+zb11+zb21+zb31+za12+za22+za32+zb12+zb22+zb32\geq 1; \\ za21+za31+zb21+zb31+z121+z131+za22+za32+zb22+zb32+z122+z132\geq 1; \\ za11+za31+zb11+zb31+z211+z231+za12+za32+zb12+zb32+z212+z232\geq 1; \\ za11+za21+zb11+zb21+z311+z321+za12+za22+zb12+zb22+z312+z322\geq 1; \\ za31+zb31+z131+z231+za32+zb32+z132+z232\geq 1; \\ za21+zb21+z121+z321+za22+zb22+z122+z322\geq 1; \\ za11+zb11+z211+z311+za12+zb12+z212+z312\geq 1;$$

$$(z111+z121+z131+z1a1+z1b1)-(z111+z211+z311+za11+zb11)=0; \\ (z211+z221+z231+z2a1+z2b1)-(z121+z221+z321+za21+zb21)=0; \\ (z311+z321+z331+z3a1+z3b1)-(z131+z231+z331+za31+zb31)=0; \\ (za11+za21+za31+zaa1+zab1)-(z1a1+z2a1+z3a1+zaa1+zba1)=0; \\ (zb11+zb21+zb31+zba1+zbb1)-(z1b1+z2b1+z3b1+zab1+zbb1)=0; \\ (z112+z122+z132+z1a2+z1b2)-(z112+z212+z312+za12+zb12)=0; \\ (z212+z222+z232+z2a2+z2b2)-(z122+z222+z322+za22+zb22)=0; \\ (z312+z322+z332+z3a2+z3b2)-(z132+z232+z332+za32+zb32)=0; \\ (za12+za22+za32+zaa2+zab2)-(z1a2+z2a2+z3a2+zaa2+zba2)=0; \\ (zb12+zb22+zb32+zba2+zbb2)-(z1b2+z2b2+z3b2+zab2+zbb2)=0;$$

$$z1a1+z1b1+z2a1+z2b1+z3a1+z3b1\leq 1;; \\ z1a2+z1b2+z2a2+z2b2+z3a2+z3b2\leq 1;$$

$$z111+z121+z131+z1a1+z1b1+za11+za21+za31+zaa1+zab1-y1a\leq 1; \\ z112+z122+z132+z1a2+z1b2+za12+za22+za32+zaa2+zab2-y1a\leq 1; \\ z211+z221+z231+z2a1+z2b1+za11+za21+za31+zaa1+zab1-y2a\leq 1; \\ z212+z222+z232+z2a2+z2b2+za12+za22+za32+zaa2+zab2-y2a\leq 1; \\ z311+z321+z331+z3a1+z3b1+za11+za21+za31+zaa1+zab1-y3a\leq 1; \\ z312+z322+z332+z3a2+z3b2+za12+za22+za32+zaa2+zab2-y3a\leq 1;$$

$$\begin{aligned}z_{111}+z_{121}+z_{131}+z_{1a1}+z_{1b1}+z_{b11}+z_{b21}+z_{b31}+z_{ba1}+z_{bb1}-y_{1b}&\leq 1; \\z_{112}+z_{122}+z_{132}+z_{1a2}+z_{1b2}+z_{b12}+z_{b22}+z_{b32}+z_{ba2}+z_{bb2}-y_{1b}&\leq 1; \\z_{211}+z_{221}+z_{231}+z_{2a1}+z_{2b1}+z_{b11}+z_{b21}+z_{b31}+z_{ba1}+z_{bb1}-y_{2b}&\leq 1; \\z_{212}+z_{222}+z_{232}+z_{2a2}+z_{2b2}+z_{b12}+z_{b22}+z_{b32}+z_{ba2}+z_{bb2}-y_{2b}&\leq 1; \\z_{311}+z_{321}+z_{331}+z_{3a1}+z_{3b1}+z_{b11}+z_{b21}+z_{b31}+z_{ba1}+z_{bb1}-y_{3b}&\leq 1; \\z_{312}+z_{322}+z_{332}+z_{3a2}+z_{3b2}+z_{b12}+z_{b22}+z_{b32}+z_{ba2}+z_{bb2}-y_{3b}&\leq 1;\end{aligned}$$

$$W_a - 50 \cdot y_{1a} - 50 \cdot y_{2a} - 40 \cdot y_{3a} = 0;$$

$$W_b - 50 \cdot y_{1b} - 50 \cdot y_{2b} - 40 \cdot y_{3b} = 0;$$

$$W_a - 150 \cdot x_a \leq 0;$$

$$W_b - 100 \cdot x_b \leq 0;$$

$$W_a \geq 0;$$

$$W_b \geq 0;$$

Tesis açma, rotalama ve müşterilerin tesislerden hizmet alması için tanımlanan değişkenler için binary kısıtları da denklem setine eklenmiştir.

ÖZGEÇMİŞ

Vildan AMİL, 28.09.1991’de İstanbul’da doğdu. İlk ve orta eğitimini İstanbul’da tamamladı. Lise eğitimini Yalova Anadolu Öğretmen Lisesi’nde, üniversite eğitimini ise Sakarya Üniversitesi Endüstri Mühendisliği’nde tamamladı.2016 yılında Sakarya Üniversitesi Endüstri Mühendisliği Bölümü’nde yüksek lisans eğitimine başladı.Meslek hayatında kısa bir süre Netlog Lojistikte “Tedarik Zinciri Sorumlusu” olarak başlayıp akabinde Konveyör A.Ş’de “Hammadde Planlama Mühendisi” olarak görev yapmaktadır.