



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

**DEPOLAMA VE GERİ ALMA PROBLEMİNİN META-SEZGİSEL
ALGORİTMALARLA ÇÖZÜMÜ**

**YÜKSEK LİSANS TEZİ
Hüseyin YILMAZ
185105002**

Bilgisayar Mühendisliği Anabilim Dalı Adı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Dr. Öğr. Üyesi Adem TUNCER

OCAK 2021



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

**DEPOLAMA VE GERİ ALMA PROBLEMİNİN META-SEZGİSEL
ALGORİTMALARLA ÇÖZÜMÜ**

**YÜKSEK LİSANS TEZİ
Hüseyin YILMAZ
185105002**

Bilgisayar Mühendisliği Anabilim Dalı Adı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Dr. Öğr. Üyesi Adem TUNCER

OCAK 2021

YALOVA Üniversitesi Fen Bilimleri Enstitüsü'nün **185105002** numaralı Yüksek Lisans Öğrencisi **Hüseyin YILMAZ**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**DEPOLAMA VE GERİ ALMA PROBLEMİNİN META-SEZGİSEL ALGORİTMALARLA ÇÖZÜMÜ**” başlıklı tezini aşağıda imzaları olan jüri önünde sunmuş ve oy birliği/oy çokluğu ile başarılı bulunmuştur.

İmza / Kanaati
(Kabul/Red)

Tez Danışmanı : **Dr. Öğr. Üyesi Adem TUNCER**/.....
Yalova Üniversitesi

Jüri Üyeleri : **Dr. Öğr. Üyesi Osman Hilmi KOÇAL**/.....
Yalova Üniversitesi

Prof. Dr. Mehmet YILDIRIM/.....
Kocaeli Üniversitesi

Teslim Tarihi : **11 Ocak 2021**
Savunma Tarihi : **03 Şubat 2021**



ÖNSÖZ

Lisans ve yüksek lisans eğitim sürecinde, hem ders hem de tez çalışma dönemlerinde bana yardımcı olan değerli danışmanım Dr. Öğr. Üyesi Adem TUNCER'e, yüksek lisans eğitimine başlamamda ve lisans döneminde desteğini esirgemeyen Dr. Öğr. Üyesi Yunus ÖZEN'e, eğitimim boyunca desteğini arkamda hissettiğim eşim Elçin YILMAZ'a ve aileme teşekkürlerimi sunarım.

Ocak 2021

Hüseyin YILMAZ
(Bilgisayar Mühendisi)





İÇİNDEKİLER

Sayfa No

ÖNSÖZ	iii
İÇİNDEKİLER	v
KISALTMALAR	ix
ÇİZELGE LİSTESİ.....	xi
ŞEKİL LİSTESİ.....	xiii
ÖZET.....	xv
ABSTRACT.....	xvii
1. GİRİŞ	1
1.1 Depolama ve Geri Alma Sistemi.....	2
1.1.1 Depolama Hareketi	2
1.1.2 Geri Alma Hareketi	4
1.2 Literatür Araştırması	5
1.3 Tezin Amacı ve Katkısı.....	10
2. META-SEZGİSEL ALGORİTMALAR.....	11
2.1 Yapay Arı Kolonisi Algoritması	11
2.1.1 Kontrol Parametreleri.....	13
2.1.2 Rastgele Yiyecek Kaynaklarının Oluşturulması	13
2.1.3 Amaç Fonksiyon Değeri	14
2.1.4 İşçi Arı Aşaması.....	14
2.1.5 Gözcü Arı Aşaması	15
2.1.5.1 Uygunluk değeri.....	15
2.1.5.2 Seçim.....	17
2.1.6 Kaşif Arı Aşaması.....	18
2.2 Genetik Algoritma.....	20
2.2.1 Çaprazlama.....	22
2.2.2 Mutasyon.....	23
2.2.3 Elitizm.....	24
3. DEPOLAMA/GERİ ALMA PROBLEMİNİN TANIMLANMASI.....	27
3.1 Maksimum Ürün Kısıtı.....	29
3.2 Minimum Ürün Kısıtı.....	29
3.3 Hacim Kısıtı	30
3.4 Ağırlık Kısıtı	30
3.5 Yakınlık Kısıtı	30
3.6 Ortam Kısıtı.....	31
3.7 Geri Alma Sıklık Kısıtı	31
3.8 Ürün Uyum Kısıtı.....	31
3.9 Son Kullanım Tarihi Kısıtı.....	31
3.10 Rota Kısıtı	32
4. DEPOLAMA/GERİ ALMA PROBLEMİNİN META-SEZGİSEL YÖNTEMLER KULLANILARAK ÇÖZÜLMESİ	33



4.1 Meta-Sezgisel Algoritmalaradaki Ortak Kullanılan Operatörler	33
4.1.1 Aday Çözüm Yapısı	33
4.1.2 Amaç Fonksiyon Değeri Hesaplama.....	33
4.1.2.1 Depolama siparişi amaç fonksiyon değeri hesaplama	34
4.1.2.2 Geri alma siparişi amaç fonksiyon değeri hesaplama	35
4.1.3 A Yıldız Algoritması ile Yol Maliyetinin Hesaplanması.....	36
4.2 Yapay Arı Kolonisi Algoritması	37
4.3 Genetik Algoritma.....	38
4.4 Hibrit Algoritma	39
5. DEPOLAMA/GERİ ALMA PROBLEMİNİN ÇÖZÜLMESİNE İLİŞKİN ELDE EDİLEN SONUÇLAR.....	43
6. SONUÇLAR ve ÖNERİLER.....	53
KAYNAKLAR	55
ÖZGEÇMİŞ	59



KISALTMALAR

A*	: A Yıldız Algoritması (A-Star Algorithm)
BS	: Besin Sayısı
HA	: Hibrit Algoritma (Hybrid Algorithm)
DGA	: Depolama Geri Alma Sistemi (Storage and Retrieval System)
PS	: Parametre Sayısı
YAKA	: Yapay Arı Kolonisi Algoritması (Artificial Bee Colony Algorithm)
GA	: Genetik Algoritma (Genetic Algorithm)





ÇİZELGE LİSTESİ

Sayfa No

Çizelge 5.1: Depolama siparişi sonuç değerleri (depo boş).....	44
Çizelge 5.2: Depolama siparişi sonuç değerleri (%25 dolu).....	44
Çizelge 5.3: Depolama siparişi sonuç değerleri (%50 dolu).....	45
Çizelge 5.4: Depolama siparişi sonuç değerleri (%75 dolu).....	46
Çizelge 5.5: Geri alma siparişi sonuç değerleri (%25 dolu)	47
Çizelge 5.6: Geri alma siparişi sonuç değerleri (%50 dolu)	48
Çizelge 5.7: Geri alma siparişi sonuç değerleri (%75 dolu)	49
Çizelge 5.8: Geri alma siparişi sonuç değerleri (%90 dolu)	50





ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: Yapay arı kolonisi algoritması akış diyagramı	12
Şekil 2.2: Sıra tabanlı uygunluk atama	16
Şekil 2.3: Orantısal uygunluk atama	16
Şekil 2.4: Rulet tekerleği yöntemi.....	17
Şekil 2.5: Genetik algoritma akış diyagramı.....	21
Şekil 2.6: Tek noktalı çaprazlama	22
Şekil 2.7: Çok noktalı çaprazlama	23
Şekil 2.8: Mutasyon işlemi.....	23
Şekil 2.9: Elitizm uygulanarak elde edilen yakınsama grafiği.....	24
Şekil 2.10: Elitizm uygulanmadan elde edilen yakınsama grafiği.....	25
Şekil 3.1: Tez çalışmasında kullanılan ürün listesi	27
Şekil 3.2: Tez çalışması için üretilen sanal depo ortamı.....	28
Şekil 4.1: Aday çözüm yapısı	33
Şekil 4.2: Yapay arı kolonisi algoritması örnek besin matrisi	37
Şekil 4.3: İşçi arı aşamasından işletilmeden önce besin matrisi	38
Şekil 4.4: İşçi arı aşaması bir kez işletildikten sonra besin matrisi.....	38
Şekil 4.5: Örnek bir çaprazlama işlemi	39
Şekil 4.6: Örnek bir mutasyon işlemi.....	39
Şekil 4.7: Hibrit algoritma akış diyagramı	40
Şekil 5.1: Depolama siparişi yakınsama grafikleri (depo boş)	43
Şekil 5.2: Depolama siparişi yakınsama grafikleri (%25 dolu)	44
Şekil 5.3: Depolama siparişi yakınsama grafikleri (%50 dolu)	45
Şekil 5.4: Depolama siparişi yakınsama grafikleri (%75 dolu)	46
Şekil 5.5: Geri alma siparişi yakınsama grafikleri (%25 dolu).....	47
Şekil 5.6: Geri alma siparişi yakınsama grafikleri (%50 dolu).....	48
Şekil 5.7: Geri alma siparişi yakınsama grafikleri (%75 dolu).....	49
Şekil 5.8: Geri alma siparişi yakınsama grafikleri (%90 dolu).....	50



DEPOLAMA GERİ ALMA PROBLEMİNİN META-SEZGİSEL ALGORİTMALARLA ÇÖZÜMÜ

ÖZET

Depo kaynaklarının verimli kullanılması, depoların daha yönetilebilir ve kullanışlı olmasını sağlayan önemli etkenlerden biridir. Ürün akışının daha hızlı gerçekleşmesini sağlayarak verimi arttırmaktadır. Ağırlık, hacim, ürün uyumluluğu gibi birçok kısıta sahip çok boyutlu depolarda, depolama ve geri alma süreci çözülmesi gereken karmaşık bir optimizasyon problemidir. Kısıtların sayısı düşünüldüğünde, depolama ve geri alma işlemlerinin geleneksel algoritmalarla çözümü yüksek efor gerektirmektedir. Meta-sezgisel algoritmalar, geleneksel algoritmalara göre daha kısa sürede kabul edilebilir çözümler sunmaktadır. Bu sebeple karmaşık optimizasyon problemlerinin çözülmesinde sıklıkla tercih edilmektedirler.

Bu tez çalışmasında, depolama ve geri alma problemini çözebilmek için popüler meta-sezgisel yöntemlerden olan genetik algoritma, yapay arı kolonisi algoritması ve genetik algortmada bulunan çaprazlama ve elitizm süreçlerinin yapay arı kolonisine eklenmesiyle oluşturulan hibrit bir yöntem kullanılmıştır. Raflar arası en optimum yolun tercih edilmesi için A-yıldız algoritması kullanılmış, depo ortamının simule edilebilmesi için operasyonel kısıtlara sahip üç boyutlu bir depo tasarlanmıştır. Depo içerisinde ürün akışını modellemek için farklı sayılarda ürün içeren siparişlerden bir kısmı depolama talebi, diğerleri ise geri alma talebi olarak belirlenmiştir.

Elde edilen sonuçlar meta-sezgisel yöntemlerin, operasyonel kısıtlara sahip çok boyutlu bir depoda, depolama ve geri alma problemine kısa sürede etkili çözümler üretebildiğini göstermektedir.

Anahtar Kelimeler: Depolama ve Geri Alma Sistemleri; Operasyonel Kısıtlar; Genetik Algoritma; Yapay Arı Kolonisi Algoritması; Hibrit Algoritma; A-Yıldız Algoritması.



SOLUTION OF STORAGE AND RETRIEVAL PROBLEM WITH META-HEURISTIC ALGORITHMS

ABSTRACT

Efficient use of warehouse resources is one of the important factors that makes them more manageable and useful. It increases efficiency by ensuring a faster flow of product. In multi-dimensional warehouses with many restrictions such as weight, volume, product compatibility, etc., the storage and retrieval processes are complex optimization problems that need to be solved. Considering the number of constraints, the solution of storage and retrieval operations with traditional algorithms requires high effort. Meta-heuristic algorithms provide acceptable solutions in less time than traditional algorithms. For this reason, they are often preferred in solving complex optimization problems.

In this thesis, the genetic algorithm, the artificial bee colony algorithm, which are the popular meta-heuristic methods, and a hybrid algorithm by adding the crossover and elitism processes in the genetic algorithm to the artificial bee colony were used in order to solve the storage and retrieval problem. The A-star algorithm was used to choose the optimum route between shelves, and a three-dimensional warehouse with operational constraints was designed to simulate the warehouse environment. In order to model the product flow in the warehouse, some of the orders containing different numbers of products were determined as storage demands and others as retrieval demands.

The results show that meta-heuristic methods can quickly produce effective solutions to the storage and retrieval problem in a multidimensional warehouse with operational constraints.

Keywords: Storage and Retrieval System; Operational Constraints; Genetic Algorithm; Artificial Bee Colony Algorithm; Hybrid Algorithm; A-Star Algorithm.



1. GİRİŞ

Son yıllarda büyük ivme kazanan bilişim teknolojileri alanında yapılan çalışmalar, dünyanın canlılar için daha konforlu ve yaşanabilir hale gelmesini sağlamaktadır. Ev elektroniği, robotik, veri analizi ve yazılım alanlarında yapılan çalışmalar günlük yaşantımızda etkisini her geçen gün daha fazla hissettirmektedir. Bilgisayar boyutlarının küçülerek taşınabilir hale gelmesi, yüz binlerce sayfalık bilginin küçük boyutlu belleklerde saklanabilmesi, yüzlerce kilometre uzaklıktan veri iletişiminin birkaç saniye içerisinde gerçekleştirilebilmesi ve internet teknolojisinin her zaman ulaşılabilir hale gelmesi teknolojik alanda katedilen mesafeyi gözler önüne sermektedir.

21. yüzyılda bilişim teknolojilerinde yaşanan atılımla birlikte, yapay zeka alanında da önemli çalışmalar yapılmaktadır. Yüz tanıma sistemleri [1], akıllı robot teknolojileri [2], optimizasyon yöntemleri kullanılarak kar marjının en üst seviyeye çıkarılmasına [3] yönelik araştırmalar bulunmaktadır.

Bu tez çalışmasında, çok katlı depoların verimli bir şekilde yönetilmesinde meta-sezgisel yöntemlerin kullanılmasına ilişkin yaklaşımlar sunulmuştur. Depolardaki ürünlerin depolanması ve geri alınması işlemlerinin en uygun bir şekilde gerçekleştirilebilmesi için Genetik Algoritma (GA), Yapay Arı Kolonisi Algoritması (YAKA) kullanılmış ve GA'da kullanılan çaprazlama ve elitizm süreçlerinin YAKA'ya eklenmesiyle oluşturulan hibrit algoritma (HA) tasarlanmıştır. Her üç algoritma farklı koşullarda çalıştırılarak deneysel sonuçları elde edilmiş ve performans değerlendirmeleri verilmiştir. Benzetim uygulamalarını gerçekleştirmek amacıyla örnek bir depo ortamı oluşturulmuş, siparişin depoya alınarak yerleştirilmesi ve istenilmesi durumunda hazırlanarak depodan gönderilmesi sağlanmaktadır. Uygulamanın amacı; kısıtlı olan arama uzayında herhangi bir kısıta takılmadan mümkün olan en uygun sonuca ulaşılmasıdır. Depo içerisindeki maliyetlerin en aza indirilmesi, sistematik bir depo altyapısının oluşturulması amaçlanmıştır. Bunun da doğal sonucu olarak hem yönetimi zor hem de maliyet açısından yönetimi zor olan insan gücünün en etkili biçimde kullanılması sağlanmaktadır.

Tez çalışması 6 bölümden oluşmaktadır. Birinci bölümde tez çalışmanın amacı, katkısı, depolama sistemleri ve benzer konularda literatürde yer alan çalışmalardan bahsedilmiştir. İkinci bölümde tez çalışması kapsamında kullanılan meta-sezgisel

algoritmalar açıklanmıştır. Üçüncü bölümde, benzetim uygulamalarını gerçekleştirmek amacıyla oluşturulan depo ortamı anlatılmıştır. Dördüncü bölümde, meta-sezgisel algoritmalar kullanılarak problemin nasıl çözüldüğü ile ilgili detaylara yer verilmiştir. Beşinci bölümde depolama ve geri alma problemine ilişkin benzetim uygulamalarında elde edilen sonuçlara yer verilmiştir. Altıncı bölümde ise sonuç ve önerilere yer verilerek tez çalışması sonlandırılmıştır.

1.1 Depolama ve Geri Alma Sistemi

DGA (Depolama Geri Alma Sistemi), teknolojinin gelişmesiyle birlikte üretim ve dağıtım ortamlarında yaygın olarak kullanılmaktadır. İnsan gücü kullanılarak depo süreçlerinin yönetildiği depolama yöntemleriyle DGA'nın kullanıldığı depolama yöntemi karşılaştırıldığında, DGA'nın tercih edildiği yöntemler öne çıkmaktadır. DGA'nın kullanıldığı depolama sistemlerinde daha az efor sarf ederek daha etkili sonuçlar alınmaktadır [7]. DGA, işletme içerisindeki depolama maliyetlerinin düşürülmesinde yardımcı olmaktadır. DGA kullanımının yaygınlaşmasıyla birlikte, insan gücünün sebep olduğu yanlış depolama operasyonları en aza indirilmiştir [8].

Otomatikleştirilmiş DGA, DGA makinelerinin koridorlar arasında dolaşabildiği sistemlerdir [9]. Bu tür sistemlerde DGA makinelerinin depodaki her rafa erişimi bulunmaktadır. DGA makineleri raflar arasında dolaşarak, ürün toplama ve ürün yerleştirme yapabilmektedir. Bu tür sistemlerde insan gücüne dayalı bir yöntem kullanılmadığı için tüm akış otomatize edilmiş sistemler tarafından DGA makinelerinin yönlendirilmesiyle yapılmaktadır.

Depolama süreçlerinde ürünler paletler halinde taşınmaktadır. Palet, ürünlerin belirli sayılarda gruplanarak oluşturulan bir depolama birimidir. DGA'da genel olarak iki farklı akış hareketi bulunmaktadır. Bunlar depolama ve geri alma hareketleridir.

1.1.1 Depolama Hareketi

Depolama sistemlerinde yerleştirilecek palet veya palet grubunun depolanması için yapılan işlemleri tanımlamaktadır. Depolama hareketleri ile işlenecek paletler, giriş noktalarından depoya alınmaktadır. Deponun çalışma şekline, fiziksel durumuna, ve taşıyıcının kapasitesine göre ürünler tek tek veya belirli gruplar halinde depo içerisinde taşınarak uygun lokasyona yerleştirilir.

Paletlerin taşıyıcılara rastgele yerleştirilmesi, yanlış lokasyona yerleştirilmesine sebep olabilmektedir. Paletlerin raflar arasındaki indirme/yükleme zorluğuna ek olarak insan tarafından manuel olarak atanmış lokasyonlardan farklı yerlere de yerleştirilmeleri mümkündür. Giriş noktasından alınacak paletlerin taşıyıcıya yüklenmeden önce yerleştirilecekleri lokasyonları belirlenmeli ve palet yüklemesinin yerleştirme sırasına uygun olarak yapılması gerekmektedir. Böylece, ürün yerleştirme sırasında zaman kaybına neden olan indirme/yükleme hareketlerinin önüne geçilebilmektedir. İnsan gücü ile yönetilen depolarda karşılaşılan bu tür durumlar depolama sürecini olduğundan daha karmaşık hale getirmektedir.

Giriş noktasında tek seferde taşıyıcının taşıyabileceğinden daha fazla palet bulunabilmektedir. Bu durumda hangi çevrimde hangi paletlerin işleneceği varsa önceliğine veya yerleşme sırasına göre belirlenmektedir. Tek çevrimde taşınabilecek palet listesi belirlendikten sonra taşıyıcının takip edeceği rota oluşturulmakta ve rota sırasına göre taşıyıcıya yüklenmektedir.

Paletlerin yerleştirilme rotası oluşturulurken depo kısıtlarına dikkat edilmelidir. Çok katlı depolarda üst raflara ulaşmanın maliyetinin daha yüksek olması nedeniyle alt rafların tercih edilmesi, işletmenin palet için belirlediği maksimum ve minimum depolama miktarı, birbirleri ile uyuşmayan ürünlerin yan yana yerleştirilmesinin engellenmesi veya deponun fiziksel yapısı gibi depo kısıtlarına dikkat edilmelidir.

Ürünlerin depo içi lokasyon atamalarının yapılması için birçok yöntem bulunmaktadır. En çok kullanılan depo içi atama yöntemleri aşağıda listelenmiştir [5]:

- Sabit depolama yöntemi (Dedicated storage method): Her bir ürünün belirli bir raf veya raf bloğuna sabit olarak atandığı depolama yöntemidir. Eğer x ürününe sahip bir palet için A bloğu sabit olarak belirlenmişse, x ürününün depo içerisinde başka bir lokasyona atanmasına izin verilmemektedir.
- Rastlantısal depolama yöntemi (Random storage method): Herhangi bir kısıt olmadan, ürünlerin hangi rafa yerleştirileceğinin rastgele belirlendiği yöntemdir. Palet, depo içerisinde herhangi bir lokasyona rastgele atanmaktadır.
- En yakın boş konum depolama yöntemi (Closest open storage method): Depo içerisinde bilinen boş lokasyonlardan en yakın olanından başlayarak paletlerin yerleştirildiği depolama yöntemidir. En yakın konum bilgisi referansı olarak çıkış noktası belirlenebilir. Bu durumda geri alma işlemlerinin hızlı

gerçekleştirilmesi beklenmektedir. En yakın konum bilgisi referansı olarak giriş noktası belirlenirse, ürünlerin yerleştirilme aşamasında en az eforla raflara yerleştirilmesi beklenmektedir.

- Tam devirli depolama yöntemi (Full turnover based storage method): En yüksek hareket frekansına sahip ürünlerin, giriş/çıkış noktalarına en yakın raflara yerleştirilmesinin önerildiği depolama yöntemidir. Yüksek frekansa sahip ürünler, giriş/çıkış noktalarına en yakın konuma yerleştirilerek depolama ve geri alma süreleri en aza indirilmektedir.
- Sınıf tabanlı depolama yöntemi (Class based storage method): Toplam depo alanının belirli alanlara bölündüğü bir depolama yöntemidir. Bu alanlar belirli ürün ya da ürün grubuna atanmaktadır. Ürünler belirlenen alanların dışında bir bölgeye yerleştirilmezler. En yüksek hareket frekansına sahip ürünler giriş/çıkış kapılarına daha yakın olan lokasyonlar ile eşleştirilmektedir. Ürünler belirlenen sabit alan içerisinde raslantısal depolama yöntemi kullanılarak yerleştirilmektedir.

Tez çalışmasında, depolama hareketlerini en az maliyetle yerine getirebilmek için sonraki bölümlerde detaylandırılmış olan depolama kısıtları dikkate alınarak bir çalışma gerçekleştirilmiştir.

1.1.2 Geri Alma Hareketi

Depolanmış paletlerin talep edilmesi durumunda depolandıkları lokasyonlardan alınarak çıkış noktasına iletilmesi, geri alma hareketi olarak tanımlanmaktadır. Deponun çalışma şekli, fiziksel durumu ve taşıyıcının kapasitesine göre ürünler tek tek veya belirli gruplar halinde toplanarak çıkış kapısına taşınmaktadır.

Geri alma işlemlerinde, taşıyıcının kapasitesine veya sevkiyat durumunun aciliyetine göre tek çevrimde toplanacak paletler belirlenmektedir. Deponun sahip olduğu kısıta göre paletlerin toplanacakları lokasyonlar belirlenmektedir. Lokasyon belirleme işi taşıyıcının inisiyatifine bırakıldığı durumlarda insan faktörü sebebiyle yanlış palet seçimi yapılarak sevkiyata dahil edilebilmektedir. Raf ömrü kısa süre sonra sona erecek bir ürünün yerine, raf ömrü daha uzun olan bir ürünün alınması, depoda kalan ürünün bozulmasına sebep olmaktadır. Bu nedenle taşıyıcı harekete geçmeden önce en uygun toplama lokasyonları bildirilmelidir.

Taşıyıcıya belirlenen paletlere uygun olarak depo içi toplama rotasının verilmesi, geri alma işlemini hızlandırmaktadır. Taşıyıcı tarafından toplama işleminin yapılacağı en hızlı rotanın belirlenmesi mümkün değildir. Depolama akışı yüksek olan depolarda, deponun fiziksel altyapısını çok iyi bilen bir çalışan olsa dahi, bazı koridorlarda anlık oluşan darboğazları tahmin edememektedir. Eğer mümkünse uygun koridor üzerinden rota belirlenerek taşıyıcı yönlendirilmelidir. Otomatikleştirilmiş DGA'da ise hem toplama lokasyonları hem de rota belirlenerek DGA makineleri yönlendirilmektedir.

Tez çalışmasında, geri alma hareketlerini en az maliyetle yerine getirebilmek için en uygun rotanın bulunması, raf ömrünün takip edilmesi gibi önceden belirlenen geri alma kısıtları dikkate alınarak bir çalışma gerçekleştirilmiştir.

1.2 Literatür Araştırması

DGA'nın doğru şekilde yönetilebilmesi literatürde bir çok araştırmaya konu olmuş karmaşık bir optimizasyon problemidir. Bu konuda literatürde farklı çalışmalar bulunmaktadır. DGA siparişlerinin yönetilmesi, otomatikleştirilmiş DGA bekleme noktalarının belirlenmesi, depo içi atama politikaları ve depo yerleşim yöntemleri alanlarında yapılan çalışmalar bu bölümde incelenmektedir.

Roodbergen ve Vis [5] tarafından gerçekleştirilen çalışmada, otomatikleştirilmiş DGA üzerine yapılan son 30 yıla ait çalışmalar incelenmiştir. Çalışma kapsamında çevrim sürelerinin tahmini olarak hesaplanmasını, depo içi lokasyon atamasını, boşta kalan veya işini tamamlayan DGA makinelerin en uygun bekleme noktasını bulmayı, otomatikleştirilmiş DGA tasarımını ve depoya gelen isteklerin doğru şekilde sıralanmasını konu alan çalışmalara yer verilmiştir.

Bektaş [9] tarafından gerçekleştirilen yüksek lisans tez çalışmasında, çalışmayı gerçekleştirmek amacıyla oluşturulan sanal depo ortamında, otomatikleştirilmiş DGA'nın benzetim ortamı aracılığıyla performans analizleri yapılmıştır. Çalışma kapsamında depo içi atama politikası ve DGA makinesinin bekleme noktası parametreleri değiştirilerek farklı senaryolar oluşturulmuştur. İlgili parametrelerin her birinin değiştirilmesi sonucu oluşturulan yeni sistemlerde parametrelerin etkileri incelenmiştir. Elde edilen sonuçlar sistem performansı açısından değerlendirilmiştir.

Balkan [10] tarafından gerçekleştirilen yüksek lisans tez çalışmasında, tek koridorlu bir çalışma ortamı için birim yük ile çalışan otomatikleştirilmiş DGA ele alınmıştır.

Tekli ve çoklu çevrimde, çevrim süresini hesaplayan alternatif bir yöntem geliştirilmiştir. Bu yöntem raftan rafa veya DGA makinesinin boşta iken beklediği noktadan rafa olan hareket süresini hesaplamaktadır. Geliştirilen yöntem, literatürde benzer konuda yapılan başka çalışmalarda kullanılan tekniklerle karşılaştırılmıştır. Microsoft Excel kullanılarak oluşturulan arayüzle, önerilen yöntem görselleştirilmiştir.

Trab ve diğ. [11] tarafından gerçekleştirilen çalışmada, ürün ve insan güvenliği açısından sürdürülebilir bir sistemin ortaya konması amaçlanmıştır. Depo içerisinde insandan kaynaklı plansız hareketlerin sebep olduğu hatalı durumları engellemek amacıyla bir ürün yerleşim algoritması önerilmiştir. Ürünlerin hatalı yerleştirmelerini önlemeye yönelik çalışmalar yapılmıştır. Yerleşim algoritması birbirleri ile uyumsuz olan ürünlerin yan yana olduğu çözümlerin yerine aynı türde ürünleri yan yana koyan çözümleri öne çıkarmaktadır. Çalışma NetLogo kullanılarak görselleştirilmiştir.

Kazemi ve diğ. [12] tarafından gerçekleştirilen çalışmada, çoklu mekik otomatikleştirilmiş DGA için iki farklı yöntem önerilmiştir. Yükleme yaparken veya boşta iken harcanan toplam seyahat süresini optimize etmeye yönelik çalışma yapılmıştır. Karınca kolonisi, benzetilmiş tavlama gibi meta-sezgisel yöntemler kullanılarak hibrit bir algoritma geliştirilmiştir. Ayrıca depo içi atama problemini çözmek amacıyla matematiksel bir yöntem önerilmiştir. Birden çok yükün aynı anda taşınabildiği DGA makineleri aracılığı ile depolama ve geri alma isteklerini tek çevrimde planlayarak seyahat sürelerinin optimize edilmesi amaçlanmıştır.

Yang ve diğ. [13] tarafından gerçekleştirilen çalışmada, çoklu mekik otomatikleştirilmiş DGA'da, geri alma işlemi sonucunda boşalacak konumun da dahil edildiği depolama ve geri alma planlaması amaçlanmıştır. Çok boyutlu optimizasyon problemini çözmek amacıyla değişken bir komşuluk arama yöntemi önerilmiştir. Önerilen yöntemin performansını değerlendirmek ve farklı parametrelerin algoritma üzerindeki etkisini için incelemek amacıyla benzetim uygulamaları gerçekleştirilmiştir.

Berg ve Gademan [14] tarafından yapılan çalışmada, her bir ürünün yerleşeceği alanın sabit olduğu bir depoda, algoritma başlangıcında belirlenen ve algoritma çalışırken herhangi bir değişikliğe uğramayan DGA isteklerinin sıralanması ele alınmıştır. Tüm isteklerin başarılı bir şekilde tamamlanması ve bu işlemlerin mümkün olan en kısa yol

ile yapılması amaçlanmıştır. Çalışmada elde edilen sonuçlar meta-sezgisel yöntemlerle elde edilen sonuçlarla karşılaştırılarak başarımlar değerlendirilmiştir.

Berg ve Gademann [15] tarafından yapılan başka bir çalışmada, otomatikleştirilmiş bir DGA'nın benzetim uygulamaları farklı parametreler kullanılarak oluşturulmuştur. Çeşitli depo içi atama durumları kullanılarak sonuçlar incelenmiştir. Sınıf tabanlı depo içi atama yönteminin kullanıldığı benzetim uygulamasında seyahat süresiyle, depolama alan gereksinimi parametreleri arasında belli oranlarda taviz veren yeni bir yöntem önerilmiştir. Depo içi atama politikası ve depolama yöntemleri dışında DGA isteklerinin sıralanması, DGA makinelerinin verimli kullanımı ile yanıt süresi parametreleri arasındaki en uygun noktanın seçilmesi üzerine de araştırmalar yapılmıştır.

Kasemset ve Sudphan [16] tarafından yapılan çalışmada, Tayland'daki bir plastik torba üreticisinin verileri kullanılarak depolama işlemleri gerçekleştirilmiştir. Benzetim uygulamasını gerçekleştirmek amacıyla oluşturulan depo ortamında matematiksel model ve sınıf tabanlı depolama politikası kullanılarak performans ölçümleri yapılmıştır. Performans ölçümleri sonucunda matematiksel modelin aylık depo içi seyahat mesafesi açısından sınıf tabanlı modele göre daha iyi sonuç verdiği gözlemlenmiştir.

Hackman ve diğ. [17] tarafından yapılan çalışmada, depolanacak tüm ürünlerin otomatikleştirilmiş DGA makineleri tarafından hareket ettirilmesinin bazı durumlarda mümkün olmadığından bahsedilmiştir. Bu tür durumlarda problemin çözülebilmesi için meta-sezgisel bir algoritma önerilmiştir. Geliştirilen meta-sezgisel algorithmada elde edilen sonuçlara göre hangi ürünlerin DGA makineleri tarafından hareket ettirileceğine dağıtım merkezi karar vermektedir. Deneysel çalışmalar için bir deniz ikmal merkezinin depo ortamı kullanılmıştır.

Heragu ve diğ. [18] tarafından yapılan çalışmada, ürün yerleşimini gerçekleştirmek amacıyla, hem matematiksel hem meta-sezgisel bir yöntem önerilmiştir. Çalışmada konu alınan depo iki fonksiyonel bölgeye ayrılmıştır. Bu bölgelerden biri ürünlerin geçici depolama alanı olarak kullanılması amacıyla ayrılmıştır. Diğer bölge ise müşterilerin bireysel siparişlerinin yerine getirilmesi, ürünlerin paketlenmesi gibi durumlar için kullanılmaktadır. Çalışmada konu alınan modeller kullanılarak fonksiyonel geri alma işlemi de yapılmıştır.

Guerriero ve diğ. [19] tarafından yapılan çalışmada çok katmanlı bir depo ortamında ürün yerleşimi ele alınmaktadır. Çalışmada teslimat sürelerini mümkün olduğunca azaltmak, lojistik maliyetlerini düşürmek ve yüksek müşteri memnuniyetini sağlamak amaçlanmıştır. Problemi matematiksel olarak ifade etmek için lineer bir model ortaya konmuştur. Büyük ölçekli senaryoları kabul edilebilir sürelerde çözümlemek amacıyla iteratif yerel arama tabanlı meta-sezgisel bir model tanımlanmıştır. Sayısal sonuçlar incelendiği zaman, meta-sezgisel yöntemin matematiksel yöntemle göre daha az efor sarfederek çözümler üretebildiği belirtilmiştir.

Sanei ve diğ. [20] tarafından yapılan çalışmada, çeşitli operasyonel kısıtlar dikkate alınarak ürünler için yerleşim problemi çözülmüştür. Bu kısıtlar envanter yönetiminin daha etkin bir şekilde gerçekleştirilmesi amacıyla belirlenmiştir. Ürünlerin depolama alanlarında dağınık olarak bulunmasını engellemek amaçlanmıştır. Problemi çözmek için iki farklı yöntem ortaya konmuştur. Bu yöntemlerden biri lineer bir tamsayı programlama modeli, diğeri meta-sezgisel bir yaklaşım içermektedir. Otomotiv endüstrisinden gelen gerçek verilere dayanan benzetim uygulamalarında, önerilen yöntemlerin makul sonuçlar üretebildiği açıklanmıştır.

Hu ve diğ. [21] tarafından yapılan çalışmada, kategoriye göre depolama yöntemi kullanılarak depolama ve geri alma maliyetlerinin azaltılması amaçlanmıştır. Bu çalışmada, parçacık sürü optimizasyonu algoritmasından esinlenilerek oluşturulmuş bir YAKA önerilmektedir. Ürünleri farklı kategorilere ayırmak için COI (sipariş başı küp endeksi) adı verilen bir değer kullanılmıştır. Yapılan deneysel çalışmaların verimliliği ve sonuçları incelendiği zaman önerilen yöntemin probleme etkili bir şekilde çözüm üretebildiği belirtilmiştir.

Hiri ve diğ. [22] tarafından yapılan çalışmada, çok tedarikçili bir ortamda sipariş miktarları ve farklı gereksinim parametrelerine göre ürün tahsis etmek için kullanılan bir model geliştirilmiştir. Geliştirilen karar destek modeli, GA tabanlı çalışmaktadır. Benzetim uygulamaları incelendiği zaman, geliştirilen modelin satın alma birimi yöneticilerinin benimsedikleri senaryolara kıyasla daha verimli kararlar verebildiğini göstermiştir.

Yang ve diğ. [23] tarafından yapılan çalışmada, otomatikleştirilmiş sistemlerin en deneyimli sevkiyat uzmanlarına oranla daha yüksek verimlilik sağlayabileceği belirtilmiştir. Birden çok fabrikanın bulunduğu bir ortamda, çeşitli ürünlerin üretilmesi

için bir sipariş tahsis şeması geliştirilmiştir. Bu şemanın oluşturulmasındaki amaç operasyon, kurulum ve nakliye aşamalarının sebep olduğu ceza maliyetiyle sipariş gecikmesinin sebep olduğu ceza maliyetinin en aza indirilmesidir. Bu problemin çözülmesi için tamsayılı bir programlama modeli önerilmiştir. Tamsayılı programlama modeliyle GA'dan elde edilen sonuçlar karşılaştırıldığında GA'nın problemin boyutu büyüdükçe daha iyi sonuçlar ortaya çıkardığı ortaya konulmuştur.

Zhou ve Mao [24] tarafından yapılan çalışmada otomatikleştirilmiş DGA'nın öneminden bahsedilerek, depolama konum optimizasyonu problemine matematiksel bir çözüm önerilmiştir. Flexsim ile depo lokasyon ataması üzerinde örnek benzetim ortamı tasarımı ve analizi gerçekleştirilmiştir. Çalışmada depolama konum atama modelinin, otomatikleştirilmiş DGA'daki depolama konumu atama sorununu etkili bir şekilde çözümleyebildiği sonucuna varılmıştır. Ayrıca bu yöntemle işletmenin depolama geri alma verimliliğinin artırıldığı vurgulanmıştır.

Gagliardi ve diğ. [25] tarafından yapılan çalışmada, otomatikleştirilmiş DGA'nın yönetimi ele alınmıştır. Literatürde yer alan çalışmaların teorikte olduğunu ve varsayımlara dayandığı belirtilerek, ilgili varsayımların endüstriyel ortama uygulanırken yeniden değerlendirilmesi gerektiği savunulmaktadır. Bu amaçla Kanada'nın Quebec şehrinde bulunan, gıda endüstrisindeki bir firmanın gerçek verileri kullanılarak bir benzetim uygulaması geliştirilmiştir. Bu uygulama, sistem performansı ve çeşitli depo atama politikalarını karşılaştırmak için kullanılmaktadır. Çalışma sonucunda en basit teorik beklentilerin dahi gerçek hayatta tekrar değerlendirilmeden kullanılamayacağı sonucuna ulaşılmaktadır.

Meneghetti ve Monti [26] tarafından yapılan çalışmada, otomatikleştirilmiş DGA sistemlerindeki DGA makinelerinin yapmış oldukları depolama ve geri alma hareketleri, enerji bakımından değerlendirilmiştir. Çalışmada sürdürülebilir bir DGA için otomatikleştirilmiş sistemlerin sadece zaman değil enerji bakımından da incelenmesi gerektiği belirtilmektedir. Çalışma kapsamında, toplama işleminin hızlanmasında önemli parametrelerden biri olan bekleme noktası politikaları enerji bakımından yeniden değerlendirilmiştir.

Koste ve diğ. [27] tarafından yapılan çalışmada, manuel sipariş toplama süreçlerinin tasarım ve kontrol karar problemlerine ilişkin literatür çalışmaları ele alınmıştır. Sipariş toplama işlemlerindeki düşük performansın yetersiz hizmete, tedarik zinciri

açısından yüksek operasyonel maliyete sebep olabileceğinden bahsedilmektedir. Bu tür olumsuzlukların önüne geçilebilmesi için rotalama, ürün yerleştirme, sipariş gruplama ve deponun bölgelere ayrılmasına ilişkin yöntemler incelenmiştir.

1.3 Tezin Amacı ve Katkısı

Tez çalışmasında, DGA’da sipariş yönetiminin doğru ve etkili bir şekilde gerçekleştirilmesi hedeflenmiştir. Meta-sezgisel yöntemlerle benzetim uygulaması gerçekleştirebilmek için sanal bir depo ortamı oluşturulmuştur. Depo ortamına, gerçek hayatta karşılaşılan kısıtlar eklenerek problem karmaşıklığı artırılmıştır. Kısıtların artmasına bağlı olarak depolanacak ürünlerin uygun raflara yerleştirilmesini ve geri alma siparişindeki ürünlerin yerlerinin belirlenmesini zorlaştırmaktadır.

Ürünlerin yerleştirilmesi için ayrıca uygun rotalar oluşturulurken, deponun sahip olduğu kısıtlar sebebiyle bir çok parametre kontrol edilmelidir. Kontrol edilmesi gereken parametrelerin fazla olması, problemin sonuç alma süresini artırmaktadır. Literatürde yer alan çalışmalarda gerçek hayatta karşılaşılabilecek kısıtların bir veya birkaçı dikkate alınarak çözümler geliştirilmiştir [4].

Meta-sezgisel algoritmalarda karmaşık problemler için kısa sürelerde kabul edilebilir sonuçlar alınabilmektedir [5]. Ayrıca meta-sezgisel yöntemlerin bir arada kullanılmasıyla daha etkili sonuçlar elde edilebilmektedir [6]. Bu sebeplerle problemin çözümünde meta-sezgisel algoritmaların kullanılması tercih edilmiştir.

Tez çalışmanın literatüre olan katkıları aşağıda özetlenmiştir:

- Hem depolama hem de geri alma siparişleri için literatürdeki farklı çalışmalarda kullanılan farklı kısıtlar bu tez çalışmasında toplanmış ve gerçek hayatta mevcut olan daha fazla kısıt dikkate alınarak çalışmanın gerçek dünya problemine daha yakın olması amaçlanmıştır.
- Standart meta-sezgisel algoritmaların kullanılmasına ek olarak hibrit bir model de kullanılmış ve problemin çözümü için daha başarılı sonuçlar elde edilmiştir.
- Optimizasyon kısıtlarının çoğaltılarak meta-sezgisel algoritmaların performanslarının da karşılaştırılması gerçekleştirilmiştir.

2. META-SEZGİSEL ALGORİTMALAR

Meta-sezgisel algoritmalar, karmaşık optimizasyon problemlerine çözüm üretmek için kullanılmaktadır. Geleneksel yöntemlerle çözüm bulunamayan veya uzun süre gerektiren problemlerde, kısa sürelerde kabul edilebilir sonuçlar üretebildikleri için tercih edilmektedir [4].

Geleneksel yöntemlerde, meta-sezgisel algoritmaların aksine algoritmaya verilen girdilere sabit kurallar uygulanarak sonuçlar elde edilmektedir. En iyi çözüm sabit kurallar çerçevesinde hesaplanabiliyorsa bulunmaktadır. Meta-sezgisel yöntemlerde en iyi çözüm garanti edilmez, ancak problem doğru ifade edilmişse en iyi çözüme yakın sonuçlar elde edilmektedir. Çözüm uzayının çok büyük olduğu ve en iyi çözüme yakın çözümlerin de kabul edilebilir olduğu problemlerde sıklıkla tercih edilmektedir.

Bu bölümde tez çalışmasında kullanılan meta-sezgisel algoritmalar açıklanmıştır.

2.1 Yapay Arı Kolonisi Algoritması

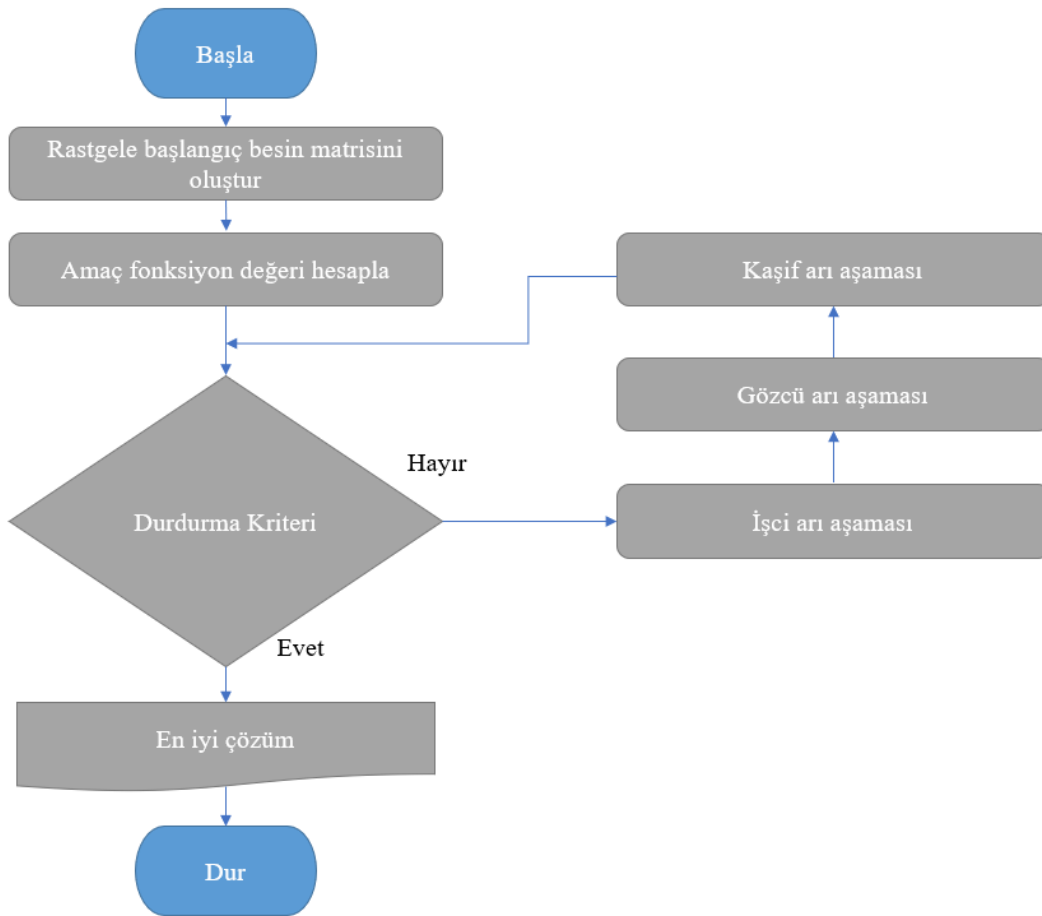
YAKA, arıların yiyecek arama davranışlarının incelenmesi sonucu ortaya çıkmış ve bu davranışların modellenmesine dayanan bir algoritmadır. 2005 yılında Derviş Karaboğa tarafından sunulmuştur [28]. Problemi ifade eden fonksiyonun minimize veya maksimize edilmesi için kullanılan meta-sezgisel bir yöntemdir.

YAKA, arı davranışlarının gözlemlenmesi sonucu oluşturulmuş bir yöntemdir. Aynı anda birden çok çözüm üzerinde çalışan bir yapıya sahiptir. YAKA'da her bir aday çözüm, yiyecek kaynağı olarak ifade edilmektedir. Yiyecek kaynaklarının oluşturduğu kümeye besin matrisi adı verilmektedir. Besin matrisinde yer alan yiyecek kaynakları, arıların yiyecek arama süreçlerinde elde ettikleri yeni kaynaklar ile güncellenmektedir. Her bir güncelleme işleminde daha iyi sonuca ulaştıran çözüm saklanarak, daha kötü sonuç veren çözüm besin matrisinden çıkarılmaktadır.

Geleneksel yöntemler kullanılırken, algoritmaya verilen girdiler işlenerek tek bir sonuç elde edilmektedir. YAKA sahip olduğu besin matrisi yapısı nedeniyle, aynı probleme birden çok çözüm önerisi sunabilmektedir. Aynı anda problemin çözüm uzayının farklı bölgelerinde arama yapabilmektedir. Algoritma sonuçlandığında elde edilen çözüm kümesi, birden çok çözüm yolu içermektedir. Elde edilen çözüm kümesindeki değerlerden biri en iyi sonucu veriyor olsa bile, diğer yiyecek kaynakları

da çözüm hakkında bilgi içermektedir. YAKA'nın rastgele sonuç arama yapısı sayesinde diğer yiyecek kaynakları da anlamlı hale gelebilmektedir.

YAKA, işçi arı, gözcü arı ve kaşif arı olmak üzere üç temel aşamadan oluşmaktadır. İşçi ve gözcü arı aşamalarında yiyecek kaynaklarının komşuluklarında yeni çözümler aranmaktadır. Komşuluk araması sonucu elde edilen yeni yiyecek kaynağının, komşuluğunda arama yapılan yiyecek kaynağından daha iyi sonuç vermesi durumunda besin matrisine dahil edilmekte ve iyi sonuç veren yiyecek kaynakları sonraki iterasyonlara aktarılmaktadır. Kötü olan yiyecek kaynağı ise yok edilmektedir.



Şekil 2.1: Yapay arı kolonisi algoritması akış diyagramı

YAKA başlatılmadan önce besin matrisinde yer alacak yiyecek kaynağı sayısı belirlenmektedir. Bu değer her bir iterasyonda aynı anda kaç farklı yiyecek kaynağı üzerinde çalışılacağını belirlemektedir. YAKA'da işçi ve gözcü arı aşamalarında kullanılacak arı sayısı, besin matrisindeki yiyecek kaynağı sayısı ile aynı olacak şekilde ayarlanmaktadır.

YAKA başlatıldığında besin matrisi rastgele değerler üretilerek veya farklı bir algoritmadan elde edilen değerler alınarak oluşturulmaktadır. Besin matrisindeki tüm yiyecek kaynakları için amaç fonksiyon değeri hesaplanmaktadır. En iyi amaç fonksiyon değerine ait kaynak, en iyi çözüm olarak saklanmaktadır. Besin matrisindeki herhangi bir yiyecek kaynağından daha iyi bir sonuç elde edilirse, ilgili yiyecek kaynağının en iyi çözümden daha iyi olup olmadığı kontrol edilmektedir.

YAKA'nın sonlandırılma kriteri olarak birçok farklı seçenek bulunmaktadır. Algoritma sonucunda elde edilebilecek en iyi sonuç değerine olan uzaklığın belirli bir değerin altına düşmesi, belirli bir iterasyon sayısı kadar daha iyi çözüm değeri elde edilememiş olması veya önceden belirlenmiş bir iterasyon sayısına ulaşılmış olması sonlandırma kriteri olarak kullanılabilir. YAKA'ya ait akış diyagramı Şekil 2.1'de görülmektedir.

2.1.1 Kontrol Parametreleri

YAKA'da üç farklı kontrol parametresi bulunmaktadır. Bunlar; her bir yiyecek kaynağının sahip olduğu parametre sayısı (PS), besin matrisindeki yiyecek kaynağı sayısı (BS) ve kaşif arı aşamasında kullanılan limit parametresidir (limit) [29]. Hem gözcü hem de işçi arı aşamalarında kullanılan gözcü arı sayısı BS değerine eşittir. Bu parametrelerin dışında algoritmanın sonlandırılması için kullanılan bir sonlandırma parametresi de belirlenmektedir.

2.1.2 Rastgele Yiyecek Kaynaklarının Oluşturulması

YAKA başlatıldığında başlangıç besin matrisinin oluşturulması için rastgele yiyecek kaynakları üretilmektedir. Yiyecek kaynaklarının oluşturulmasına ilişkin formül denklem 2.1'de verilmiştir [30]:

$$x_{i,j} = x_j^{min} + rand(0,1) (x_j^{min} - x_j^{max}) \quad (2.1)$$

Burada $i=1,2,...,BS$ olmak üzere besin matrisindeki yiyecek kaynak sayısını, $j=1,2,...,PS$ olmak üzere yiyecek kaynağındaki optimize edilecek olan parametre sayısını göstermektedir. x_j^{min} , ilgili parametrenin minimum alabileceği değeri, x_j^{max} ise

maksimum değeri ifade etmektedir. $rand(0,1)$ fonksiyonu 0-1 arasında rastgele üretilen bir değeri temsil etmektedir.

2.1.3 Amaç Fonksiyon Değeri

YAKA ve benzer tüm meta-sezgisel algoritmalarda, problem ile problem çözümü arasındaki bağlantı, amaç fonksiyon değerlerinin hesaplanma yöntemiyle belirlenmektedir. Bir yiyecek kaynağının, problemin çözümüne olan yakınlığı amaç fonksiyon değeri ile ifade edilmektedir. Amaç fonksiyon değerinin hesaplanma yöntemi problemi iyi ifade edebilirse, problemin en uygun çözümüne yakın sonuçlar elde edilebilmektedir.

2.1.4 İşçi Arı Aşaması

İşçi arı aşamasında, besin matrisinde yer alan yiyecek kaynağı sayısı kadar işçi arı çalışmaktadır. Her bir işçi arı, besin matrisindeki bir yiyecek kaynağının komşuluğunda yeni bir yiyecek kaynağı aramakla görevlendirilmiştir. Elde edilen yeni yiyecek kaynağı için amaç fonksiyon değeri hesaplanmaktadır. Yeni yiyecek kaynağının amaç fonksiyon değerinin mevcut yiyecek kaynağının amaç fonksiyonunun değerinden daha iyi olup olmadığı kontrol edilmektedir. Bu karşılaştırma işlemi genellikle açgözlü arama yöntemi ile gerçekleştirilmektedir. Elde edilen yeni kaynak, mevcut kaynaktan daha iyi ise mevcut yiyecek kaynağı besin matrisinden çıkarılarak yerine yeni yiyecek kaynağı eklenmektedir.

Yeni yiyecek kaynağının daha iyi amaç fonksiyon değerine sahip olmadığı durumda besin matrisinde değişiklik yapılmamaktadır. İlgili yiyecek kaynağı için kaşif arı aşamasında kullanılacak olan geliştirilememe sayacı bir artırılarak, arama yapılmasına rağmen daha iyi çözüm bulunamadığı bilgisi kaydedilmektedir. Kaynağın geliştirilememe sayacı belirlenen limit değerine ulaştığında artık o kaynak tükenmiştir ve bu kaynak için kaşif arı aşamasına geçilmektedir.

Denklem 2.2’de bir yiyecek kaynağının komşuluğunda yeni bir yiyecek kaynağı belirlenmesine ilişkin işlemler gösterilmektedir [30]:

$$x'_{i,j} = x_{i,j} + \varphi_{i,j}(x_{i,j} - x_{k,j}) \quad (2.2)$$

$x'_{i,j}$, mevcut yiyecek kaynağı içerisinde değiştirilecek olan parametreyi göstermektedir. Parametrenin değiştirilmesiyle yeni bir x'_i yiyecek kaynağı elde edilmektedir. φ , $[-1,1]$ arasında rastgele üretilecek bir sayısal değeri, k ise besin matrisindeki rastgele seçilen bir yiyecek kaynağını temsil etmektedir ($i \neq k$).

2.1.5 Gözcü Arı Aşaması

Gözcü arı aşamasında, işçi arı aşamasında olduğu gibi besin matrisinde yer alan yiyecek kaynağı sayısı kadar arı çalışmaktadır. Bu aşamada besin matrisinde yer alan tüm yiyecek kaynaklarına amaç fonksiyon değerine göre bir uygunluk değeri atanmaktadır.

Seçim işleminde algoritma başlangıcında belirlenen yiyecek kaynağı sayısı kadar aday çözüm seçilmektedir. Besin matrisinde olasılık durumuna bağlı olarak, bir yiyecek kaynağı birden çok kez seçilebilmektedir. Seçilen her bir yiyecek kaynağının komşuluğunda yeni bir yiyecek kaynağı elde edilmektedir. Yeni yiyecek kaynağına ait amaç fonksiyon değeri, komşuluğundan elde edilen yiyecek kaynağının amaç fonksiyon değerinden daha iyi ise besin matrisine komşuluğunda elde edilen yeni yiyecek kaynağı dahil edilir ve geliştirilememe sayacı sıfırlanır. Mevcut çözümün daha iyi olması durumunda ise geliştirilememe sayacı bir arttırılır.

2.1.5.1 Uygunluk değeri

Yiyecek kaynakları arasında seçim işleminin gerçekleştirilmesi için amaç fonksiyon değerleriyle orantılı olarak uygunluk değerleri belirlenmektedir. Uygunluk atama yöntemi olarak birçok seçenek bulunmaktadır. Atanan uygunluk değerlerine göre her bir yiyecek kaynağına yüzdesel bir seçilme oranı verilerek seçim işlemine geçilmektedir.

Sıra tabanlı uygunluk atama

Sıra tabanlı uygunluk atama yöntemi, uygunluk atama işleminde sık tercih edilen yöntemlerden biridir. Yiyecek kaynaklarının amaç fonksiyon değerleri arasında bir sıralama yapılarak, en iyi amaç fonksiyon değerine en yüksek puan verilmektedir. Bu yöntemde, en iyi değerden en kötü değere doğru giderken uygunluk değerleri

arasındaki farklar sabittir. Sıra tabanlı uygunluk atama yöntemi kullanılırken kullanılan formül denklem 2.3’de ifade edilmektedir:

$$U_i = U_{max} \frac{(N - P_i)}{(N - 1)} \quad (2.3)$$

U_{max} en iyi uygunluk değerini, N besin matrisindeki yiyecek kaynağı sayısını, P_i ise yiyecek kaynağına ait sıra numarasını ifade etmektedir. Sıra numarası değerleri; en uygun değer 1, en kötü amaç fonksiyon değerine sahip yiyecek kaynağı N olmak üzere atanmaktadır. Şekil 2.2’de örnek amaç fonksiyon değerlerine atanan uygunluk değerleri gösterilmektedir ($U_{max} = 2$):

Amaç Fonksiyon Değerleri	100	140	500	675	1500
Atanan Uygunluk Değerleri	2	1.5	1	0.5	0

Şekil 2.2: Sıra tabanlı uygunluk atama

Orantısal uygunluk atama

Orantısal uygunluk atama yönteminde sıra tabanlı uygunluk atama yönteminden farklı olarak her bir amaç fonksiyon değeri, toplam amaç fonksiyon değerleri içerisindeki ağırlığına göre değer almaktadır. Orantısal uygunluk atama işlemi sırasında kullanılan formül denklem 2.4’te ifade edilmektedir:

$$U_i = U_{max} - \frac{(f_i - f_{min})}{(f_{max} - f_{min})} \quad (2.4)$$

U_{max} belirlenen maksimum uygunluk değerini, f_i , i . yiyecek kaynağının amaç fonksiyon değerini, f_{min} en kötü amaç fonksiyon değerini ve f_{max} besin matrisindeki en iyi amaç fonksiyon değerini ifade etmektedir. Şekil 2.3’de örnek amaç fonksiyon değerlerine göre atanan uygunluk değerleri gösterilmektedir ($U_{max} = 2$):

Amaç Fonksiyon Değerleri	100	140	500	675	1500
Atanan Uygunluk Değerleri	2	1.97	1.71	0.58	1

Şekil 2.3: Orantısal uygunluk atama

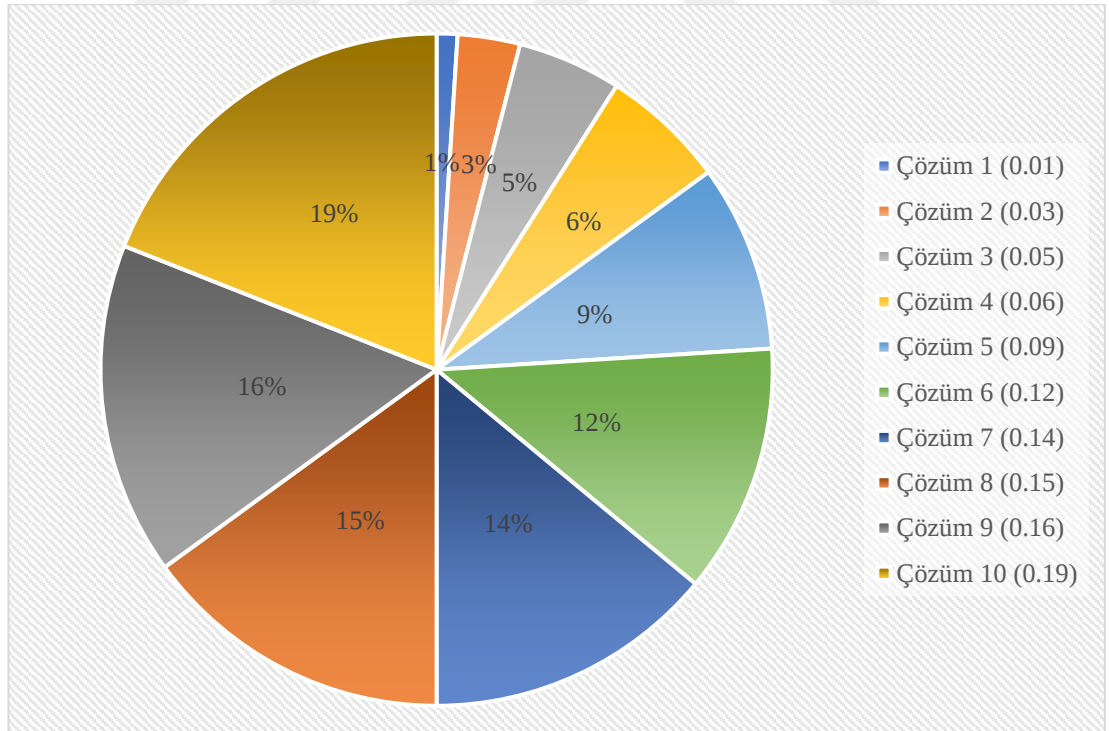
2.1.5.2 Seçim

Seçim aşamasında, tüm yiyecek kaynaklarına 0 ile 1 arasında seçilme olasılığı verilmektedir. Tüm bireylerin seçilme olasılıkları toplamı 1 olacak şekilde belirlenmekte, bu sayede 0 ile 1 arasında rastgele üretilecek bir sayı ile seçim işlemi yapılmaktadır. Bireylerin seçilme olasılığı hesaplanırken denklem 2.5'te ifade edilen formül kullanılmaktadır:

$$S_i = \frac{U_i}{\sum_{i=1}^{P_s} U_i} \quad (2.5)$$

Burada, U_i i . yiyecek kaynağına ait uygunluk değerini ifade etmektedir. Yiyecek kaynağına ait uygunluk değeri, tüm uygunluk değerleri toplamına bölünerek seçim olasılığı hesaplanmaktadır.

Şekil 2.4'de görüldüğü gibi, tüm yiyecek kaynaklarının seçim olasılıkları toplamı 1 olacak şekilde dairesel bir tekerlek üzerine yerleştirilir. Bu yönteme rulet tekerleği yöntemi adı verilmektedir. [0-1] aralığında rastgele üretilen sayı hangi aralığa denk geliyorsa, o yiyecek kaynağı besin matrisine dahil edilmektedir.



Şekil 2.4: Rulet tekerleği yöntemi

Tez çalışmasında bahsedilen uygunluk atama yöntemleri, daha iyi sonuç üreten yiyecek kaynaklarına daha yüksek uygunluk değeri vermektedir. Diğer yiyecek

kaynaklarına oranla daha kötü amaç fonksiyon değerlerine sahip olan yiyecek kaynaklarına da sonraki iterasyonlara aktarılma şansı verilmektedir. Bir yiyecek kaynağı besin matrisindeki diğer çözümlere oranla kötü bir amaç fonksiyon değerine sahip olsa dahi, komşuluğunda elde edilecek yeni yiyecek kaynağı iyi bir amaç fonksiyon değerine sahip olabilmektedir.

Kullanılacak seçim ve uygunluk atama yöntemlerinin hangisinin daha iyi sonuçlar üreteceği konusunda kesin bir yöntem bulunmamaktadır. Bir problemde sıra tabanlı uygunluk atama yöntemi kullanılan algoritma daha iyi performans gösterebileceği gibi, başka bir problemde orantısal uygunluk atama yöntemi kullanılan algoritma daha iyi performans gösterebilmektedir. Hangi yöntemin kullanılacağına karar verebilmek için deneme yanılma yolu ile elde edilecek sonuçlar incelenmelidir.

2.1.6 Kaşif Arı Aşaması

Gözcü ve işçi arı aşamalarında her bir kaynağın komşuluğunda daha iyi bir yiyecek kaynağı bulunamadığı zaman, ilgili yiyecek kaynağına ait geliştirilememe sayacı bir arttırılmaktadır. Geliştirilememe sayacı kaynağın tükenip tükenmediği konusunda bilgi vermektedir. Limit değeri kadar deneme yapılmasına rağmen daha iyi bir değer elde edilemiyorsa, yiyecek kaynağının komşuluğunda arama yapılmaktan vazgeçilmektedir. Algoritmanın bu bölümünde her bir yiyecek kaynağı için saklanan geliştirilememe sayacının, algoritmanın başında belirlenen limit değerine ulaşip ulaşmadığı kontrol edilmektedir. Tükenen kaynakta görevli olan arı, kaşif arı haline dönüşmekte ve kaşif arı denklem 2.1'i kullanarak rastgele yeni bir çözüm üretmektedir.

Kaşif arı aşamasında limit değerini aşan birden çok yiyecek kaynağının bulunması durumunda, en yüksek limit değerine sahip olan yiyecek kaynağı işleme alınmaktadır [30]. Kaşif arı aşaması, gözcü ve işçi arı aşamalarından farklı olarak her bir iterasyonda tek bir yiyecek kaynağı üzerine uygulanmaktadır.

YAKA'ya ait tüm işlemler tamamlandıktan sonra algoritma başlangıcında belirlenen optimizasyon kriterinin sağlanıp sağlanmadığı kontrol edilmektedir. Optimizasyon kriteri sağlanıyorsa, o ana kadar bulunmuş olan en iyi amaç fonksiyonuna sahip yiyecek kaynağı sonuç olarak kaydedilerek algoritma sonlandırılmaktadır.

YAKA'nın çalışma adımları aşağıda verilmiştir:

1. Tüm $x_{i,j}$ değerleri için denklem 2.1 kullanılarak rastgele yiyecek kaynakları üretilir ($i=1,2,...BS, j=1,2,...PS$).
2. Yiyecek kaynaklarının amaç fonksiyon değerleri hesaplanır (f_i).
3. Döngü başlangıç (durdurma kriteri sağlanana kadar)
4. Döngü başlangıç $i=1,2,...BS$ (İşçi arı aşaması)
5. Denklem 2.2 kullanılarak x_i komşuluğunda x_i' kaynağı üretilir.
6. Yeni x_i' kaynağının amaç fonksiyon değeri hesaplanır (f_i').
7. Eğer f_i' değeri f_i değerinden daha iyi ise
8. $x_i = x_i'$, yeni kaynak besin matrisine dahil edilir ve i . kaynağın geliştirilememe sayacı sıfırlanır ($l_i = 0$)
9. Değilse
10. x_i kaynağı tutulur ve geliştirilememe sayacı artırılır ($l_i = l_i + 1$)
11. Döngü sonlandırılır (i)
12. Döngü başlangıç $i=1,2,...BS$ (Uygunluk değeri atama)
13. Denklem 2.3 veya 2.4 kullanılarak uygunluk değeri atanır (u_i).
14. Döngü sonlandırılır (i)
15. Döngü başlangıç $i=1,2,...BS$ (Seçilme olasılığı belirleme)
16. Denklem 2.5 kullanılarak seçilme olasılığı belirlenir (s_i).
17. Döngü sonlandırılır (i)
18. Döngü başlangıç $i=1,2,...BS$ (Gözcü arı aşaması)
19. Eğer rastgele(0,1) < s_i
20. Denklem 2.2 kullanılarak seçilen yiyecek kaynağının komşuluğunda x_i' kaynağı üretilir.
21. Yeni kaynağın amaç fonksiyon değeri hesaplanır (f_i').
22. Eğer f_i' değeri f_i değerinden daha iyi ise
23. $x_i = x_i'$, yeni kaynak besin matrisine dahil edilir ve i . kaynağın geliştirilememe sayacı sıfırlanır ($l_i = 0$)
24. Değilse
25. x_i kaynağı tutulur ve geliştirilememe sayacı artırılır ($l_i = l_i + 1$)
26. Döngü sonlandırılır (i)
27. Eğer l^{max} değeri limit değerine ulaşmış ise
28. Denklem 2.1 kullanılarak rastgele yiyecek kaynağı oluşturulur.
29. Yeni kaynak besin matrisine dahil edilir ve yiyecek kaynağının geliştirilememe sayacı sıfırlanır.
30. Kaynağın amaç fonksiyon değeri hesaplanır.
31. Elde edilen en iyi yiyecek kaynağı saklanır.
32. Döngü sonlandırılır (sonlandırma kriteri)
33. En iyi yiyecek kaynağı kaydedilir.

2.2 Genetik Algoritma

GA, doğadaki canlılara ait kalıtım özelliklerinin sonraki nesillere aktarılmasının rol model alınması sonucu ortaya çıkmış bir algoritmadır. 1970’li yılların başlarında John Holland tarafından sunulmuştur [31]. Algoritmanın temel amacı; GA’ya ait kontrol parametreleri kullanılarak, problemi ifade eden fonksiyonun optimum değerlerin elde edilmesidir.

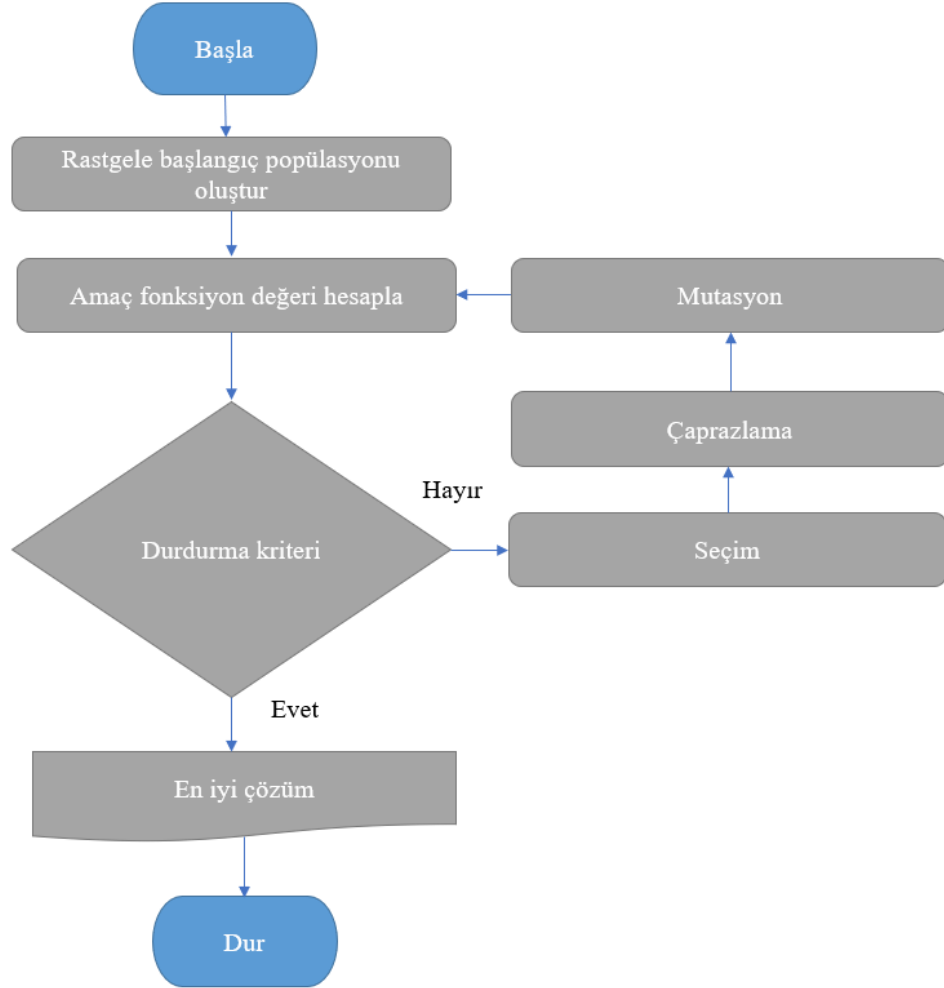
GA, evrimsel süreçlerden esinlenilerek oluşturulmuş bir yöntemdir. Rastgele veya evrimsel süreçler sonucu ortaya çıkan her bir aday çözüme kromozom, kromozomların her bir parametresine gen adı verilmektedir [32]. GA her çalıştırıldığında çıktı olarak bir çözüm kümesi elde edilmektedir. Kromozomların oluşturduğu çözüm kümesine ise popülasyon adı verilmektedir. Kromozomların kalıtsal süreçlerde işlenmesi sonucu, bir sonraki popülasyon oluşturulmaktadır. Algoritmanın işleyişindeki her bir iterasyona jenerasyon adı verilmektedir. Her jenerasyon sonrasında en iyi sonuca ulaştıran kromozomlar saklanarak, en kötü sonucu veren kromozomların olasılığa bağlı olarak popülasyondan çıkarılmasıyla en iyi çözüme ulaşılmaktadır.

Geleneksel yöntemlerde, algoritmaya verilen girdiler işlenerek tek bir sonuç elde edilmektedir. GA’nın sahip olduğu popülasyon yapısı nedeniyle, aynı probleme ait birden çok çözüm elde edilmektedir. Elde edilen çözüm kümesi birden çok çözüm yolu içermektedir. Bu çözümlerden biri, diğer çözümlere göre daha iyi sonuç vermektedir. Geriye kalan kromozomlar daha kötü amaç fonksiyon değerine sahip olsalar bile, problemin çözümü hakkında bilgi içermektedirler. Sahip olduğu popülasyon yapısı sebebiyle aynı anda problemin çözüm uzayının farklı bölgelerinde arama yapılabilmektedir.

GA çalıştırıldığında ilk olarak başlangıç popülasyonu oluşturulmaktadır. Başlangıç popülasyonu, rastgele değer üretilerek veya farklı bir algoritmanın çalıştırılması sonucu elde edilebilmektedir.

Tüm meta-sezgisel yöntemlerde olduğu gibi kromozomlar arasında hangisinin daha iyi sonuç verdiğini belirlemek için problemi ifade eden bir fonksiyon kullanılmaktadır. Bu fonksiyon problem ile algoritma arasındaki bağlantıyı sağlamaktadır. GA çalıştırıldığında her bir iterasyonda tüm kromozomlar için amaç fonksiyon değeri hesaplanmaktadır. Amaç fonksiyon değerleri için uygunluk atama işlemi yapılmaktadır. Belirlenen uygunluk değerine göre her kromozomun sonraki

jenerasyonda yer alabilmesi için bir olasılık değeri belirlenmektedir. Seçim işleminde olasılığa bağlı olarak rastgele seçilen kromozomlara sırasıyla çaprazlama ve mutasyon işlemleri uygulanarak yeni popülasyon oluşturulmaktadır.



Şekil 2.5: Genetik algoritma akış diyagramı

En iyi amaç fonksiyonuna sahip kromozom, seçim aşamasında yapılan işlemler sonucunda sonraki jenerasyona aktarılmamış olabilmektedir. Aktarılmış olsa bile, çaprazlama veya mutasyon işlemi sonucu daha kötü amaç fonksiyonuna sahip bir kromozoma dönüşmüş olma ihtimali de bulunmaktadır. Bu nedenle mutasyon işleminden sonra yeni popülasyondaki kromozomlardan en kötüsü, önceki iterasyonda en iyi amaç fonksiyon değerine sahip kromozom ile değiştirilmektedir.

GA'da da diğer algoritmalarda olduğu gibi sonlandırma işlemi belirlenen bir durdurma kriteri ile gerçekleştirilmektedir. Durdurma kriteri sağlanmıyorsa, sağlanana kadar işlemler aynı sıra ile devam ettirilmektedir. Algoritma sonucu en iyi amaç

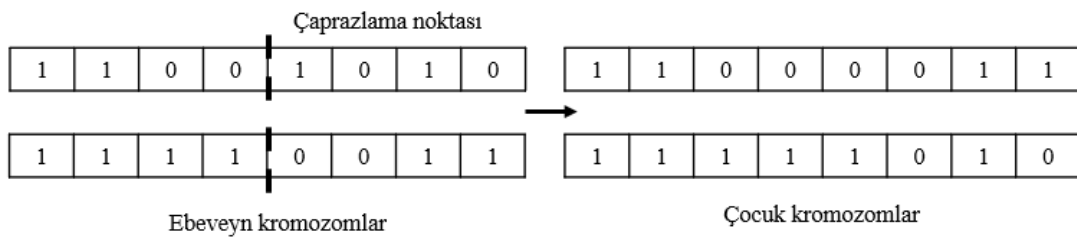
fonksiyon değerine sahip kromozom, algoritmanın çıkışı olarak kaydedilerek işlemler sonlandırılır. GA'nın işletilmesine ilişkin akış diyagramı şekil 2.5'te gösterilmektedir.

Algoritmanın sonlandırılma kriteri belirlenirken birçok seçenek bulunmaktadır. Algoritmanın elde ettiği en iyi kromozomun, eğer biliniyorsa mutlak çözüme olan hata payının belirli bir oranın altına düşürülmesi, algoritma başlangıcında belirlenen jenerasyon sayısının tamamlanması veya belirli bir jenerasyon sayısı kadar çalıştırıldığı halde daha iyi bir çözümün elde edilememiş olması algoritmanın sonlandırma kriteri olarak kullanılabilir.

2.2.1 Çaprazlama

Seçim işlemi ile birlikte seçilen kromozomlara çaprazlama işlemi uygulanmaktadır [33]. Çaprazlama işlemi, GA'da çeşitliliğin sağlanmasına katkı sağlamaktadır. Kromozomlar arasında gen alışverişi yapılarak yeni kromozomların elde edilmesi sağlanmaktadır.

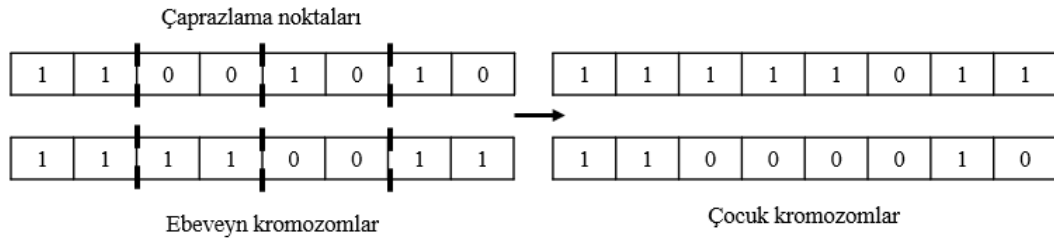
Çaprazlama işlemi uygulanırken bir çok farklı yöntem bulunmaktadır. Literatürde tek noktalı çaprazlama ve çok noktalı çaprazlama gibi yöntemler bulunmaktadır [34]. Şekil 2.6'da görüldüğü gibi, tek noktalı çaprazlama yönteminde kromozomlara ait genler rastgele belirlenmiş herhangi bir noktadan değiştirilerek yeni kromozomlar oluşturulmaktadır. Çok noktalı çaprazlama yönteminde ise rastgele üretilecek bir sayıya göre çaprazlama sayısı belirlenmektedir. Örnek bir çok noktalı çaprazlama işlemi şekil 2.7'de görülmektedir.



Şekil 2.6: Tek noktalı çaprazlama

Çaprazlama aşamasında farklı yöntemlerin kullanıldığı çalışmalar bulunmaktadır. Bazı çalışmalarda çaprazlama işlemi olasılığa bağlı olarak yapılmaktadır. Rastgele üretilen sayı belirli bir eşiğin üzerindeyse çaprazlama yapılır, diğer durumda ise bireyler oldukları gibi sonraki sürece aktarılmaktadır [35]. Çaprazlama yöntemine

seçim ve uygunluk atama işlemlerinde olduğu gibi deneme yanılma yöntemiyle elde edilen sonuçlara göre karar verilmektedir.

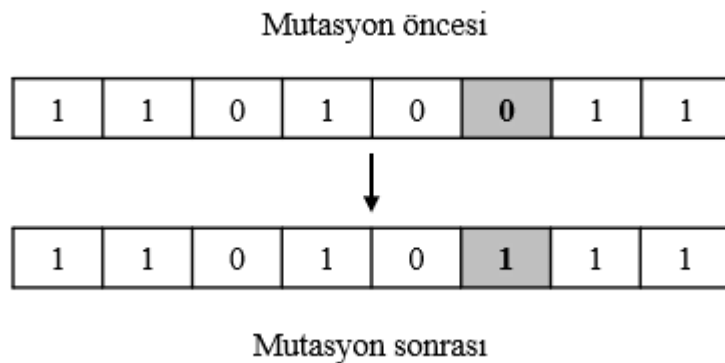


Şekil 2.7: Çok noktalı çaprazlama

2.2.2 Mutasyon

Mutasyon işlemi, çaprazlama sonucunda elde edilen kromozomların genlerinde rastgele değişimler gerçekleştirmek için kullanılan GA operatörüdür. Canlılarda çeşitliliğin sağlanması ve farklı kromozomların meydana gelmesinde çok önemli bir yeri bulunmaktadır. GA’da ise önceki popülasyondan farklı kromozomlar elde etmek ve daha iyi amaç fonksiyon değerlerine sahip kromozomların rastlantısal olarak oluşmasına yardımcı olmak için kullanılmaktadır. Mutasyon sırasında meydana gelen değişim, her zaman kromozomun daha iyi bir amaç fonksiyon değerine sahip olacağı anlamına gelmemektedir.

Birbirini takip eden jenerasyonlar sonucu her ne kadar çaprazlama, seçim gibi operasyonlar uygunlansa da bireylerin birbirlerine benzemeye başladıkları gözlemlenmiştir. Bu durumdan kaçınmak için mutasyon işlemi uygulanmaktadır.



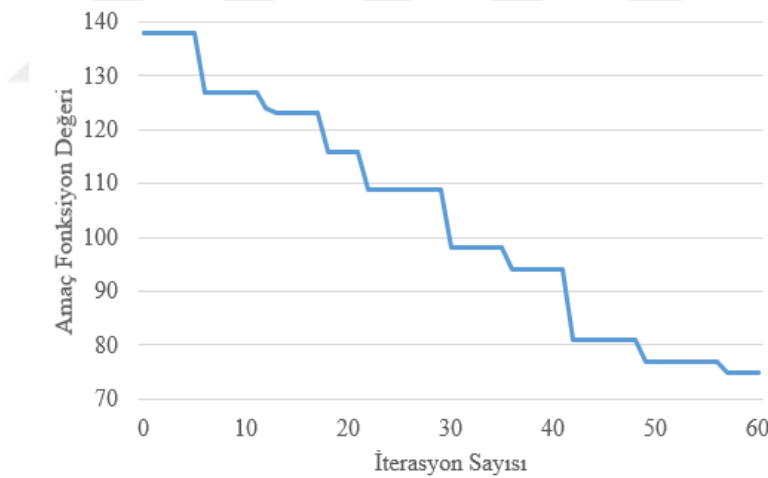
Şekil 2.8: Mutasyon işlemi

Algoritma başlatılmadan önce mutasyon oranı belirlenmektedir. Bu oran sabit kalabileceği gibi duruma göre artırılıp azaltılan çalışmalar da bulunmaktadır [36].

Mutasyon oranı, kromozom genlerindeki değişim oranını ifade etmektedir. Yüksek değerler seçilmesi durumunda çok fazla değişime sebep olmaktadır. Bu durum çözüm uzayının ilgili bölgesinde yeterince arama yapılmadan, başka bir bölgeye geçilmesi anlamına gelmektedir. Böylece daha iyi çözüm bulunabilme ihtimali varken çözüm bulunamamaktadır. Düşük değerler seçilmesi durumunda ise aynı bölgede çok fazla arama yapıldığı gözlemlenmiştir. Diğer operasyonlarda olduğu gibi denemeler yapılarak en iyi oran bulunmuştur. Örnek bir mutasyon işlemi Şekil 2.8’de görülmektedir.

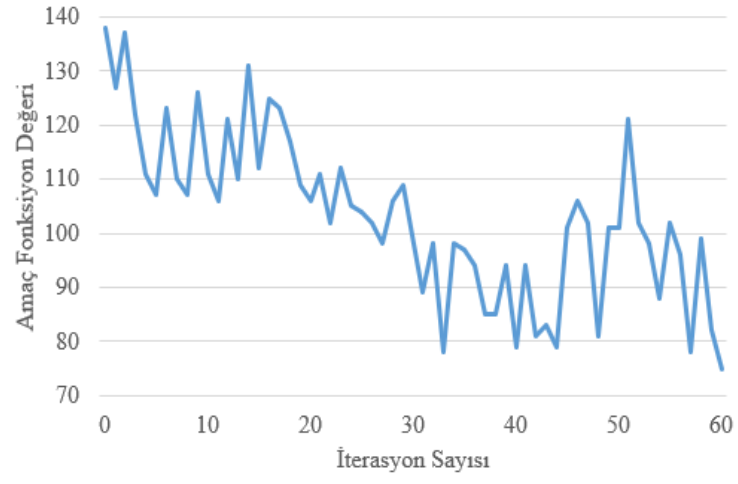
2.2.3 Elitizm

GA’da kullanılan seçim, çaprazlama ve mutasyon işlemleri nedeniyle popülasyon içerisindeki en iyi kromozomun sonraki jenerasyona aktarılma ihtimali bulunmaktadır. Elitizm adı verilen yöntemle en iyi amaç fonksiyonuna sahip kromozom veya kromozomların sonraki popülasyonda bulunmaları garanti altına alınmaktadır. Şekil 2.9 ve şekil 2.10 üzerinden örnek bir çalışma için elitizm uygulanarak ve elitizm uygulanmadan elde edilen yakınsama grafikleri görülmektedir.



Şekil 2.9: Elitizm uygulanarak elde edilen yakınsama grafiği

Tüm algoritma operasyonları, algoritmanın sonlandırılma koşulu sağlanana kadar devam ettirilmektedir. Her bir iterasyonda elde edilen en iyi değerler şekil 2.9 ve şekil 2.10 üzerinden de görülebileceği gibi yakınsama grafiklerini oluşturmaktadır. Yakınsama grafiğinde en optimum değeri alan sonuç, algoritmanın çıktısı olarak kaydedilir ve algoritma sonlandırılır.



Şekil 2.10: Elitizm uygulanmadan elde edilen yakınsama grafiği





3. DEPOLAMA/GERİ ALMA PROBLEMİNİN TANIMLANMASI

DGA’da birden çok kısıta sahip çok boyutlu depolarda, ürünlerin depolama ve geri alma lokasyonlarının belirlenmesi karmaşık bir optimizasyon problemidir. Bu tez çalışmasında kullanılan meta-sezgisel yöntemlerle, paletlerin mümkün olan en uygun yerlere depolanması sağlanmaktadır. İhtiyaç duyulması durumunda en uygun lokasyondaki paletler seçilerek geri alınmaktadır.

Benzetim uygulamalarında kullanılan depo ortamında, depolama birimi olarak palet belirlenmiştir. Bir ürün depolama veya geri alma işleminde paletler halinde yer alabilmektedir. Çalışmada, depolama ve geri alma siparişlerinde kullanmak üzere farklı özelliklere sahip 5 ürün belirlenmiştir.

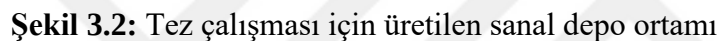
Ürün Numarası	Ürün Türü	Palet Genişlik (cm)	Palet Derinlik (cm)	Palet Yükseklik (cm)	Palet Ağırlığı (kg)	Ortam Kısıtı
1	Temizlik	80	120	80	400	Var
2	Yiyecek	100	120	100	600	Yok
3	Yiyecek	120	100	110	750	Yok
4	Yiyecek	120	100	90	800	Yok
5	Yiyecek	60	80	60	300	Yok

Şekil 3.1: Tez çalışmasında kullanılan ürün listesi

Şekil 3.1 üzerinde tez çalışmasında kullanılan ürün listesi gösterilmektedir. 1 numaralı ürün toz deterjandır. Temizlik maddesi olduğu için yiyecek türündeki diğer ürünlere komşu raflara yerleştirilememesi gerekmektedir. Raflar arasında en az bir raf boşluk bırakılmalıdır. Ayrıca, toz deterjan ürününün nemli ortamda saklanması ürünün çabuk bozulmasına sebep olmaktadır.

Depolama ve geri alma süreçlerinin işletilebilmesi için farklı siparişler benzetim uygulamalarında rastgele oluşturulmuştur. Bu siparişlerden bir kısmı depolama, bir kısmı ise geri alma siparişi olarak belirlenmiştir. Her bir siparişte 6 ile 13 arasında değişen sayılarda palet bulunmaktadır. Boş depoyu belirli bir oranda doldurmak için siparişlerden bir kısmı depolama siparişi olacak şekilde düzenlenmiştir. GA, YAKA ve HA farklı kontrol parametreleri altında çalıştırılarak en iyi amaç fonksiyon değerlerini veren kontrol parametreleri belirlenmiştir. Sonuçların eşit

Benzetim uygulamalarını gerçekleştirmek amacıyla çizilen çok boyutlu depo, şekil 3.2’de gösterilmektedir.



Depoda toplamda 38 raf bloğu ve 228 adet depolama alanı bulunmaktadır. Her bir depolama alanı dikeyde 5 farklı raf olmak üzere toplamda 1140 adet raftan oluşmaktadır. Rafların her birine sadece tek bir palet yerleştirilmektedir. Aynı raf bloğu içerisinde yer alan rafların genişlik, yükseklik, derinlik ve taşıma kapasiteleri birbirine eşittir. Şekil 3.2 üzerinde 'I' ile gösterilen alan depolama siparişlerinin kabul edildiği bölümü göstermektedir. 'O' ise geri alma siparişlerin sevkiyata gönderildiği bölümü temsil etmektedir.

28

Şekil 3.2 üzerinde görülen depo ortamında, sol üst köşede yer alan raf bloğunun (1, 5, 29, 43, 71) bulunduğu bölge fiziksel durumdan dolayı nemli bölge olarak işaretlenmiştir. Bu bölgeye ürünün depo içerisinde tahrip olmasına yol açacağı için toz deterjan ürünü yerleştirilmesine izin verilmemektedir.

Benzetim uygulamalarında kullanılmak için oluşturulan çok boyutlu depoda, gerçek hayatta karşılaşılan bazı kısıtların eklenmesi problemin karmaşıklığını arttırmıştır. Depolama ve geri alma işlemlerinde hiçbir kısıt bulunmasaydı, herhangi bir algoritma kullanılmadan en yakın lokasyondan başlayarak depolama ve geri alma işlemleri gerçekleştirilebilirdi. Gerçek hayatta karşılaşılan kısıtlar, depolama ve geri alma işlemlerini karar verilmesi zor bir hale getirdiği için meta-sezgisel yöntemler tercih edilmiştir.

Benzetim uygulamalarında depolama ve geri alma işlemleri sırasında belirlenen kısıtlara dikkat edilerek işlem yapılmaktadır. Meta-sezgisel algoritmaların çalışması sırasında dikkat edilen kısıtlara aşağıda yer verilmiştir.

3.1 Maksimum Ürün Kısıtı

Depo içerisinde yer alan her ürünün, raf ömrü ve son kullanım tarihi bulunmaktadır. Özellikle yiyecek ve içecek ürünlerinin kısa zamanda bozulması sebebiyle ihtiyaç duyulan miktardan fazla depolanması tercih edilmemektedir. Tez çalışmasında kullanılan yöntemlerde belirlenen sınırın üzerindeki depolama işlemlerine izin verilmemektedir.

Depolarda sipariş kabul aşamasında gelen her ürün bir için maksimum ürün kısıtı kontrol edilmektedir. Raflarda bulunan toplam ürün sayısı, depolama siparişinde yer alan ürün sayılarının toplamı maksimum ürün kısıt miktarının üzerindeyse sipariş kabul edilmemektedir.

3.2 Minimum Ürün Kısıtı

Depoda bir ürünün miktarının belirlenen minimum kısıt değerinin altında düşmesi durumunda, müşteri ihtiyaçları karşılanamayabilir. Müşterilerin bir ürüne acil olarak ihtiyaç duyması durumunda, talep edilen miktarın depodan karşılanması gerekmektedir. İhtiyaç karşılanamazsa gerçek hayatta müşteri kayıpları yaşanabilmektedir.

Minimum ürün kısıt değerin altına düşülen durumlarda, ilgili ürün için tedarikçiye satın alma siparişi gönderilmelidir. Tez çalışmasında kullanılan yöntemlerde, belirlenen minimum ürün kısıt değerin altına düşüleceğinin tespit edilmesi durumunda tedarikçiye en az minimum limitin üstüne çıkacak miktarda yeni ürün siparişi verilmektedir.

3.3 Hacim Kısıtı

Depolama işlemlerinde farklı türde palet birimleri bulunmaktadır. Dünya üzerinde farklı coğrafi bölgelerde farklı palet standartları kullanılmaktadır. Bir firma, ithalat ihracat yapıyorsa depo içerisinde farklı hacimlerde paletler bulunabilmektedir. Tez çalışmasında kullanılan yöntemlerde bir rafın genişlik, yükseklik ve derinlik değeri eşleştirilen palet ile uyumlu değilse raf ile paletin eşleşmesine izin verilmemektedir. Kısıtı ihlal eden her bir palet-raf eşleşmesi için amaç fonksiyon değerine ceza puanı eklenmektedir.

3.4 Ağırlık Kısıtı

Depo içerisinde kullanılan paletler farklı ağırlık değerlerine sahiptir. Depo raflarının, yapıldıkları malzemeye göre taşıyabilecekleri maksimum ağırlık değerleri bulunmaktadır. Tez çalışmasında, rafın maksimum taşıma kapasitesi paletin ağırlığından daha küçükse raf ile paletin eşleştirilmesine izin verilmemektedir. Kısıtı ihlal eden her bir palet-raf eşleşmesi için amaç fonksiyon değerine ceza puanı eklenmektedir.

3.5 Yakınlık Kısıtı

Depo süreçlerinde akış hareketlerinin hızlı gerçekleştirilebilmesi için ürünlerin birbirlerine yakın lokasyonlarda depolanması gerekmektedir. Hem depolama hem de geri alma işlemlerinde, tek bir siparişte aynı üründen birden fazla palet kullanılabilir. Ürünlere ait paletlerinin belirli bir bölgede gruplanmış olması sürecin hızlanması sağlanmaktadır. Ürünlerin bir arada depolanması, depoda daha düzenli bir yapı oluşmasına yardımcı olmaktadır. Bu sebeple ürünler mümkün olduğunca birbirlerine yakın olarak yerleştirilmelidir. Tez çalışmasında ürünlerin mümkün olan en yakın lokasyonlara yerleştirilmesi tercih edilmektedir.

3.6 Ortam Kısıtı

Depodaki bazı bölgeler, belirli ürünlerin daha hızlı bozulmasına sebep olmaktadır. Bu tür kısıta sahip ürünler, ilgili raflar için yasaklı olarak işaretlenmelidir. Örneğin; deponun belirli bir bölgesinde rutubet oluşuyorsa, kağıt ham maddesine sahip ürünlerin bu bölgelerde saklanması tercih edilmemelidir. Ürün kayıplarını en aza indirmek için, bu tür ortam kısıtlarına sahip ürünler yerleştirilirken dikkat edilmelidir. Kısıtı ihlal eden her bir palet-raf eşleşmesi için amaç fonksiyon değerine ceza puanı eklenmektedir.

Tez çalışmasında kullanılan yöntemlerde, ortam kısıtını ihlal eden çözümler tercih edilmemektedir.

3.7 Geri Alma Sıklık Kısıtı

Bir ürün geri alma siparişlerinde sıkça yer alıyorsa, ürüne ait paletler çıkış kapısına yakın olarak yerleştirilmelidir. Bu durum geri alma işlemlerinin daha hızlı yapılabilmesi açısından önemlidir. Bu tez çalışmasında kullanılan her ürün için bir frekans değeri bulunmaktadır. Frekans değeri; geri alma siparişlerinde yer alan ürün sayısının, aynı türdeki ürün sayısına bölünmesiyle hesaplanmaktadır. En yüksek frekans değerine sahip ürünler çıkış kapısına en yakın raflara yerleştirilmektedir.

3.8 Ürün Uyum Kısıtı

Bir ürün yerleştirilirken komşu raflarında yer alan ürünler kontrol edilmelidir. İki farklı ürün, birbirlerinin daha hızlı bozulmasına sebep olacak ürünler ise yan yana yerleştirilmemelidir. Depolama işlemi yapılırken, komşu raflarda bu türde bir ürün olup olmadığı kontrol edilmelidir. Tez çalışmasında kullanılan yöntemlerde, ürün uyum kısıtını ihlal eden çözümlere ceza puanı eklenmektedir.

3.9 Son Kullanım Tarihi Kısıtı

Her ürüne ait bir son kullanım tarihi bulunmaktadır. Depoda operasyonel hatalar nedeniyle ürünlerin son kullanım tarihleri geçebilmektedir. Son kullanım tarihi geçen bir ürün geri alma siparişlerinde kullanılmamalıdır. Bu tez çalışmasında, ürünlerin depoda bozulmasını engellemek için geri alma siparişlerinde ürünler seçilirken son kullanım tarihi en yakın olan ürünler tercih edilmektedir.

3.10 Rota Kısıtı

Depo içi dolaşım miktarının en aza indirilmesi, siparişlere daha hızlı cevap verilebilmesi için oldukça önemlidir. Herhangi bir ürünün depolanması veya geri alınması sırasında mümkün olan en optimal rota seçilerek taşıyıcıya bildirilmelidir. Taşıyıcı bu rotayı kullanarak ürünleri giriş veya çıkış noktasına teslim etmelidir.

Bu tez çalışmasında uygulanan meta-sezgisel yöntemlerde, belirtilen kısıtlara uymayan çözümlere ceza puanı eklenmektedir. Aday çözümün amaç fonksiyon değeri olduğundan daha yüksek bir değer almaktadır, böylece bu çözüm en iyi çözüm olmaktan uzaklaştırılmaktadır. Aday çözüm kümesi içerisindeki tüm çözümlerin ceza puanı alması durumunda ise algoritma tekrar başlatılarak çözüm aranmaktadır. Yakınsama grafiklerinde yer alan amaç fonksiyon değerleri, ceza puanı içermeyen çözümlerden oluşmaktadır.

4. DEPOLAMA/GERİ ALMA PROBLEMİNİN META-SEZGİSEL YÖNTEMLER KULLANILARAK ÇÖZÜLMESİ

Tez çalışmasının bu bölümünde, kullanılan meta-sezgisel yöntemlerin benzetim uygulamalarında kullanımına ilişkin detaylar yer almaktadır. Benzetim uygulamalarını gerçekleştirmek amacıyla 3 farklı yöntem kullanılarak sonuçlar elde edilmiştir. Optimizasyon problemlerinde sıklıkla tercih edilen GA ve YAKA yöntemlerinin yanı sıra, bir de hibrit yöntem kullanılmıştır. HA işçi arı aşaması, gözcü arı aşaması, elitizm ve çaprazlama operatörleri kullanılarak oluşturulmuştur.

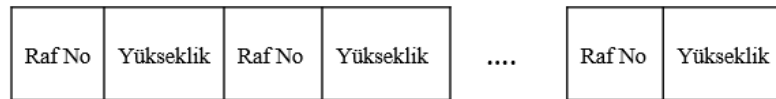
Meta-sezgisel yöntemlerin kullanımına ilişkin detaylara aşağıda yer verilmektedir.

4.1 Meta-Sezgisel Algoritmalarındaki Ortak Kullanılan Operatörler

GA, YAKA ve HA yöntemlerinin uygulanması sırasında ortak olarak kullanılan operatörlere bu bölümde yer verilmiştir.

4.1.1 Aday Çözüm Yapısı

GA’da her bir aday çözüme kromozom, bir iterasyondaki aday çözümlerin tamamına popülasyon adı verilmektedir. YAKA’da ise aday çözümlere yiyecek kaynağı, aday çözüm kümesine besin matrisi adı verilmektedir. Her iki yöntemde de aday çözüm yapısı benzer şekilde oluşturulmuştur. Aday çözümlerin ikili, tamsayı veya gerçel sayısı gibi farklı şekillerde kodlanması mümkündür [37]. Tez çalışmasında aday çözümler tam sayı şeklinde kodlanarak oluşturulmuştur. Aday çözüm yapısı şekil 4.1’de gösterilmektedir.



Şekil 4.1: Aday çözüm yapısı

4.1.2 Amaç Fonksiyon Değeri Hesaplama

Amaç fonksiyon değeri, kullanılan tüm meta-sezgisel yöntemlerde problem ile çözüm arasındaki ilişkiyi ifade etmektedir. Hem depolama hem de geri alma işlemlerinde en küçük amaç fonksiyon değerine sahip çözümler, en iyi aday çözümleri temsil etmektedir.

Herhangi bir kısıta takılan aday çözüme ceza puanı eklenmektedir. Eklenen ceza puanıyla birlikte aday çözümün amaç fonksiyon değeri artmaktadır. Böylece ceza puanı almayan bir aday çözüm, ceza alan aday çözüme göre daha iyi bir amaç fonksiyon değerine sahip olmaktadır.

4.1.2.1 Depolama siparişi amaç fonksiyon değeri hesaplama

Kullanılan meta-sezgisel yöntemlerde, tüm aday çözümler için amaç fonksiyon değeri hesaplanmaktadır. Her bir siparişte birden çok palet bulunmaktadır. Amaç fonksiyon değeri, tüm paletlerin atanmış olan raflara yerleştirilmesi sonucu elde edilen toplam puanı ifade etmektedir. Depolama siparişlerinde kullanılan amaç fonksiyon değeri hesaplama formülü, denklem 4.1’de verilmiştir:

$$F_S = f_c + f_r + f_p \quad (4.1)$$

F_S depolama siparişleri için amaç fonksiyon değerini ifade etmektedir. f_r değeri, depolama siparişlerinde yüksek geri alma frekansına sahip ürünleri ön plana çıkarmak için kullanılmaktadır. f_c aday çözüm içerisindeki paletlerin, kendi türlerindeki ürünlere olan toplam uzaklığını ifade etmektedir. Aday çözüm içerisindeki tüm paletler için ayrı ayrı hesaplanmaktadır. Ürünlerin birbirlerine mümkün olduğunca yakın yerleştirilebilmeleri için denkleme dahil edilmiştir. f_p , depolama siparişlerinde paletlerin ilgili rafa yerleştirilmesine engel olacak ceza puanlarının toplamını temsil etmektedir.

$$f_c = \left(\sum_{i=1}^n \sum_{j=1}^{m^t} d(s_i, s_j) \right) / \sum sp \quad (4.2)$$

Denklem 4.2’deki ilk toplama operatörünün üst limit değeri olan n değeri, siparişteki toplam ürün sayısını ifade etmektedir. İkinci toplama operatörü ise, ilgili ürünün depoda daha önce yerleştirilmiş olan tüm s_i türündeki ürünlere olan toplam uzaklıklarını göstermek için kullanılmaktadır. m^t , s_i ürününün depodaki toplam sayısını ifade etmektedir. $d(s_i, s_j)$, aday çözüm içerisindeki palet-raf eşleşmesinin, daha önce depolanmış olan aynı türdeki ürüne olan uzaklığını hesaplamak için kullanılmaktadır. $\sum sp$ ifadesi, depoda bulunan toplam ürün sayısını ifade etmektedir.

f_r değerinin hesaplanmasına ilişkin detaylar denklem 4.3’de ifade edilmiştir:

$$f_r = \sum_{i=1}^n d(s_i, O) \times (1 - rf_{s_i}) \quad (4.3)$$

Denklem 4.3'deki toplama operatörünün üst limit değeri olan n değeri, siparişteki toplam palet sayısını ifade etmektedir. Bu denklemdeki işlemler her bir palet-raf eşleşmesi için tekrarlanmaktadır. s_i , yerleştirilecek olan palete atanan raf lokasyonunu, $d(s_i, O)$ ise atanmış raf lokasyonunun çıkış kapısına olan uzaklığını ifade etmektedir. rf_{s_i} geri alma siparişlerinde ürünün kullanıma frekansını göstermektedir. Bu değer 1'den çıkarılarak en yüksek frekansa sahip ürünün, denkleme en az etki göstermesi amaçlanmıştır.

$$f_p = \sum_{i=1}^n (C_i^w + C_i^h + C_i^e + C_i^{pc}) \quad (4.4)$$

Denklem 4.4'deki toplama operatörünün üst limit değeri olan n değeri, siparişteki toplam ürün sayısını ifade etmektedir. Denklemdeki işlemler, aday çözümdeki her bir palet-raf eşleşmesi için tekrarlanmaktadır. C_i^w palete atanmış olan rafın maksimum taşıma kapasitesinin yeterli olmadığı durumda eklenecek ceza puanını ifade etmektedir. C_i^h ifadesi, palet-raf eşleşmesinde ürünün rafa sığmaması durumunda eklenecek ceza puanını temsil etmektedir. C_i^e ifadesi, palet-raf eşleşmesinde herhangi bir ortam kısıtına takılması durumunda eklenecek ceza puanını ifade etmektedir. C_i^{pc} ifadesi, paletin yerleşeceği rafın etrafında daha hızlı bozulmasına sebep olabilecek bir ürün bulunması durumunda eklenecek ceza puanını temsil etmektedir (ürün uyum kısıtı). Palet-raf eşleşmesinin kısıtlardan herhangi birine takılması durumunda eklenecek ceza puanı, her bir kısıt için sabit olarak belirlenmiştir. Bir palet-raf eşleşmesi birden çok ceza alabilmektedir.

4.1.2.2 Geri alma siparişi amaç fonksiyon değeri hesaplama

Geri alma süreçlerinde depolama siparişlerinde olduğu gibi tüm aday çözümler için amaç fonksiyon değeri hesaplanmaktadır. Her bir siparişte birden çok ürün bulunduğu için (6-13 palet), amaç fonksiyon değeri tüm paletlerin geri alınmasıyla ilgili toplam puanı ifade etmektedir. Geri alma siparişlerinde kullanılan amaç fonksiyon değeri, denklem 4.4'de ifade edilmiştir:

$$F_R = \left(\sum_{i=1}^{n-1} d(s_i, s_{i+1}) \right) + d(s_n, O) + \sum_{i=1}^n l_{s_i} / n \quad (4.4)$$

F_R geri alma siparişi için amaç fonksiyon değerini ifade etmektedir. $d(s_i, s_{i+1})$ ifadesi, aday çözüm içerisindeki s_i paletine atanan lokasyon ile ondan sonra gelen s_{i+1} paleti arasındaki uzaklığı ifade etmektedir. $d(s_n, O)$ ifadesi, siparişteki son ürünün (s_n) çıkış kapısına olan uzaklığını ifade etmektedir. Denklem 4.4'deki ilk toplama operatörü, ürünlere atanan rafları dolaşmak için kat edilmesi gereken mesafeyi hesaplamak için kullanılmaktadır. İkinci toplama operatörü ise son kullanım tarihi yaklaşan paletlerin geri alma siparişlerinde ön plana çıkarılması için kullanılmaktadır. Depolama siparişlerinde her ürün yerleştirildiğinde, sipariş sırasına göre bir sayısal değer verilmektedir. Bu sayısal değer (depolama sırası olarak da ifade edilebilir) depoya gelen ilk sipariş için en küçük değeri almaktadır. l_{s_i} ifadesi, ilgili paletteki ürüne ait depolanma sırasını ifade etmektedir. Bu değer ne kadar küçükse, depolama işlemi o kadar önce gerçekleşmiştir. Depoya ilk gelen ürünlere geri alma siparişlerinde yer verilmesi, son kullanım tarih kısıtı açısından önemlidir.

4.1.3 A Yıldız Algoritması ile Yol Maliyetinin Hesaplanması

A yıldız algoritması (A^*), aynı uzayda yer alan iki noktanın birbirine olan en kısa yolu bulmak için geliştirilmiş bir algoritmadır. A^* 'ın sahip olduğu liste yapısı nedeniyle düzlem üzerinde ziyaret edilen nokta hafızaya kaydedilmektedir. Tüm ziyaret edilen noktaların hafızaya alınması sebebiyle yüksek miktarda bellek kullanılmaktadır.

A^* 'da verilen iki nokta arasında en az bir yol mevcutsa, bunlardan en kısa olanı bulmayı garanti etmektedir. Kolay uygulanabilir yapısı nedeniyle birçok alanda tercih edilmektedir. Oyunlarda kısa yol problemlerinde [38], otonom araçlar için yol planlamada [39, 40], insansız hava araçları (İHA) için 3 boyutlu dünyada rota hesaplama [41] için kullanılmaktadır.

A^* 'da bir noktadan diğerine hareket etmek için komşuluk ilişkisi kullanılmaktadır. Komşuluk ilişkisi algoritmanın bir sonraki adımda gideceği noktanın seçeneklerini tanımlamaktadır.

A^* hem sezgisel hem de deterministik bir yapıya sahiptir. Bitiş noktasına olan uzaklık tahmin edilerek en kısa yol bulunmaktadır. A^* 'ın maliyet hesaplaması için denklem 4.5 kullanılmaktadır:

$$f(n) = g(n) + h(n) \quad (4.5)$$

$f(n)$ algoritmanın maliyet değerini ifade etmektedir. $g(n)$ ifadesi, başlangıç düğümünden maliyet değeri hesaplanan düğüme kadar kat edilen mesafeyi ifade etmek için kullanılmaktadır. $h(n)$, A^* 'daki sezgisel fonksiyonu tanımlamaktadır. Maliyet hesaplanan düğümden hedefe olan tahmini uzaklık değeridir. Bu tez çalışmasında $h(n)$ değeri hesaplanırken, düğümden hedefe olan Öklid mesafesi kullanılmaktadır.

Bu tez çalışmasında giriş kapısı, çıkış kapısı ve raflar arasında rota maliyeti hesaplanırken A^* kullanılmaktadır.

4.2 Yapay Arı Kolonisi Algoritması

Bu bölümde benzetim uygulamalarında kullanılan YAKA algoritma adımlarının kullanılmasına ilişkin detaylar yer almaktadır. YAKA'da kullanılan örnek bir besin matrisi şekil 4.2'de gösterilmektedir.

Aday Çözüm 1	128	1	132	5	110	1	26	3	33	5
Aday Çözüm 2	122	3	32	4	66	4	1	3	21	5
Aday Çözüm 3	18	1	101	3	10	4	68	2	226	2
Aday Çözüm 4	145	1	12	2	11	4	45	3	43	3
Aday Çözüm 5	58	2	71	5	25	2	5	3	19	5

Şekil 4.2: Yapay arı kolonisi algoritması örnek besin matrisi

İşçi arı aşamasında yeni bir çözüm arama işlemi için besin matrisinden rastgele bir yiyecek kaynağı seçilmektedir (Şekil 4.3 için 3 numaralı yiyecek kaynağı seçilmiştir). Bu yiyecek kaynağı, ilgili işçi arının çalıştığı yiyecek kaynağından (Şekil 4.3 için 1 numaralı yiyecek kaynağı) farklı bir çözüm olmalıdır. Yiyecek kaynağı belirlendikten sonra, değiştirilmek üzere rastgele bir parametre seçilir (Şekil 4.3'de parametre 2 olarak seçilmiştir). İşçi arının çalıştığı yiyecek kaynağında rastgele seçilen parametreye karşılık gelen değerler, seçilen yiyecek kaynağındaki parametreye karşılık gelen değerlerle değiştirilir. Şekil 4.4'de işçi arı aşaması bir kez işletildikten sonra oluşan yeni besin matrisi görülmektedir.

Aday Çözüm 1	128	1	132	5	110	1	26	3	33	5
Aday Çözüm 2	122	3	32	4	66	4	1	3	21	5
Aday Çözüm 3	18	1	101	3	10	4	68	2	226	2
Aday Çözüm 4	145	1	12	2	11	4	45	3	43	3
Aday Çözüm 5	58	2	71	5	25	2	5	3	19	5

Şekil 4.3: İşçi arı aşamasından işletilmeden önce besin matrisi

Aday Çözüm 1	128	1	132	5	10	4	26	3	33	5
Aday Çözüm 2	122	3	32	4	66	4	1	3	21	5
Aday Çözüm 3	18	1	101	3	10	4	68	2	226	2
Aday Çözüm 4	145	1	12	2	11	4	45	3	43	3
Aday Çözüm 5	58	2	71	5	25	2	5	3	19	5

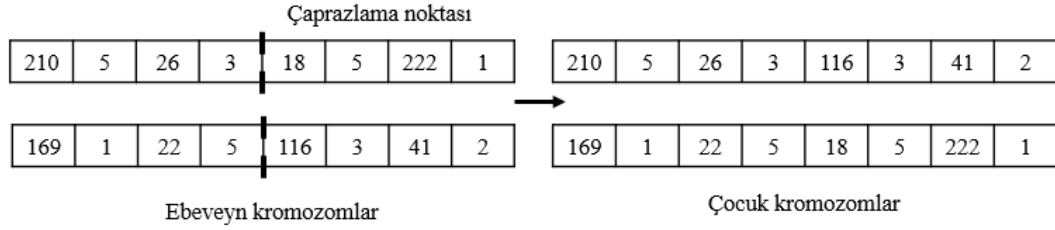
Şekil 4.4: İşçi arı aşaması bir kez işletildikten sonra besin matrisi

4.3 Genetik Algoritma

Bu tez çalışmasında kullanılan GA’da algoritma işletilirken kullanılan operatörler bu bölümde anlatılmaktadır.

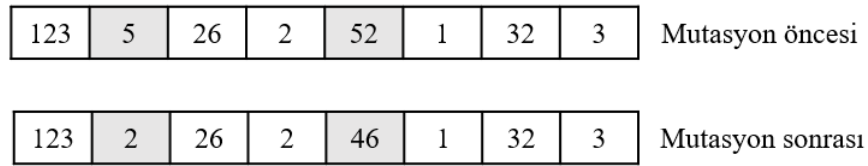
GA’da önceki bölümlerde bahsedilen orantısal uygunluk atama yöntemi, benzetim ortamında yapılan denemelere göre daha iyi sonuçlar verdiği için uygunluk atama yöntemi olarak tercih edilmiştir. Seçim yöntemi olarak rulet tekerleği yöntemi tercih edilmiştir.

Bu tez çalışmasında GA bölümünde bahsedilen tek noktalı çaprazlama yöntemi tercih edilmiştir. Çaprazlama noktası, rastgele seçilen bir sayı ile belirlenmektedir. Örnek bir çaprazlama işlemi şekil 4.5’de gösterilmektedir.



Şekil 4.5: Örnek bir çaprazlama işlemi

Bu tez çalışmasında mutasyon oranı yapılan denemeler sonucunda en iyi sonuçları verdiği için %4 olarak belirlenmiştir. Örnek bir mutasyon işlemi Şekil 4.6'da gösterilmektedir.



Şekil 4.6: Örnek bir mutasyon işlemi

Tez çalışmasında en iyi amaç fonksiyon değerine sahip kromozomun, sonraki iterasyona aktarılacak şekilde düzenleme yapılmıştır.

4.4 Hibrit Algoritma

Hibrit algoritma; GA ve YAKA'da elde edilen amaç fonksiyon değerlerinden daha iyi bir aday çözüm elde etmek amacıyla kullanılmaktadır. GA ve YAKA operatörleri bir arada kullanılarak daha iyi amaç fonksiyon değerlerine sahip aday çözümler elde edilmektedir.

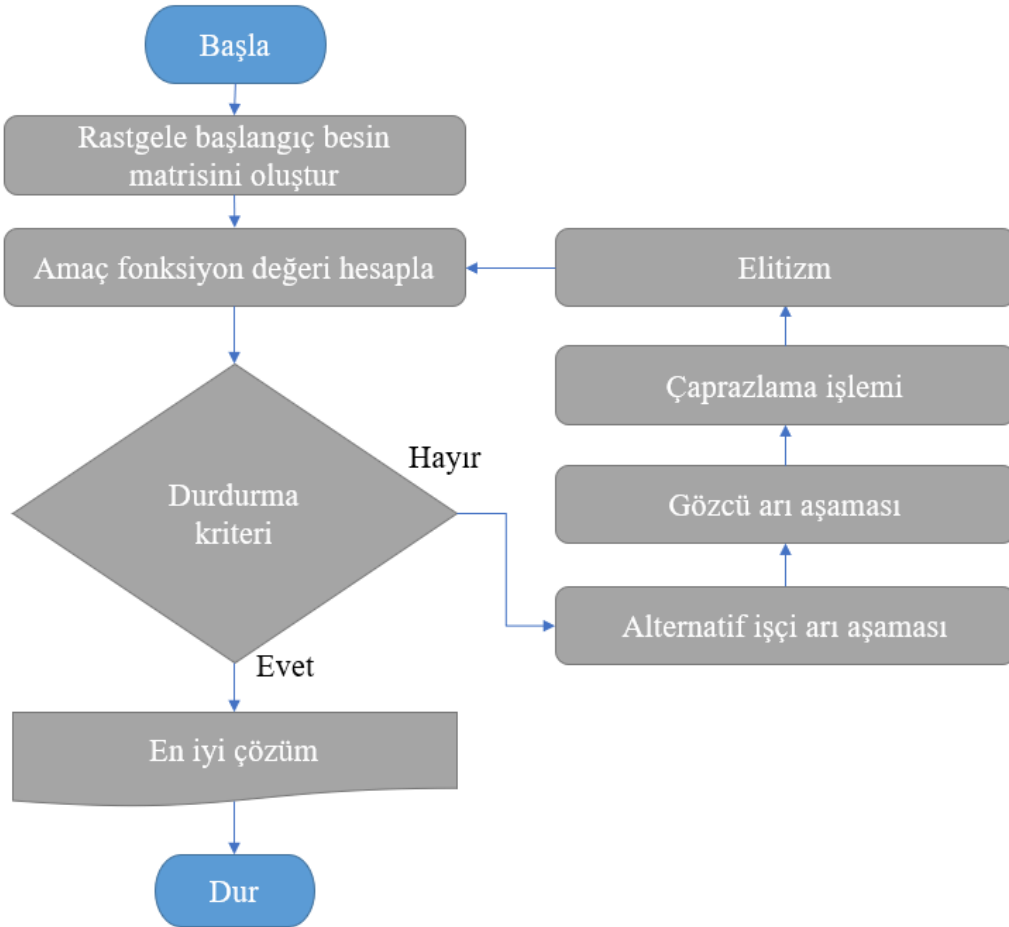
HA, standart YAKA algoritması üzerinde bazı değişiklikler yapılarak oluşturulmuştur. GA'dan ise çaprazlama ve elitizm operasyonları eklenerek, kaşif arı aşaması çıkarılmıştır. HA'ya standart YAKA yönteminden farklı olarak aşağıdaki operasyonlar eklenmiştir:

- İşçi arı aşamasında %40 ihtimalle alternatif komşuluk arama yöntemi kullanılmaktadır. Standart YAKA algoritmasında yiyecek kaynağı parametelerinden biri, besin matrisi içerisinde rastgele seçilen başka bir yiyecek kaynağındaki parametreyle değiştirilmektedir. Bu durum arama bölgesini çözüm uzayının belirli bir bölgesine daralmasına yol açmaktadır. Çözüm uzayının farklı bölgelerinde arama yapılabilmesi için alternatif

yöntemde bir yiyecek kaynağından değil rastgele yeni bir parametre üretilmektedir.

- Standart YAKA yönteminde gözcü arı aşamasında seçim işlemi yapılmaktadır. Bu durum en iyi amaç fonksiyon değerine sahip yiyecek kaynaklarının kaybedilmesine sebep olmaktadır. GA yönteminde kullanılan elitizm işlemi, eklenerek en iyi amaç fonksiyon değerine sahip yiyecek kaynağının sonraki iterasyonlarda kullanılması garanti altına alınmaktadır.
- Gözcü arı aşaması sona erdikten sonra GA'da kullanılan çaprazlama operatörü uygulanmaktadır. Her bir aday çözüm için belirli bir ihtimalle rastgele bir çaprazlama noktası belirlenerek işlem yapılmaktadır. Çaprazlama yapılan her bir kromozom için tekrar amaç fonksiyon hesaplama işlemi yapılmaktadır.

HA'ya ait akış diyagramı şekil 4.7'de gösterilmektedir.



Şekil 4.7: Hibrit algoritma akış diyagramı

HA'nın çalışma adımları aşağıda verilmiştir:

1. Tüm $x_{i,j}$ değerleri için denklem 2.1 kullanılarak rastgele yiyecek kaynakları üretilir ($i=1,2,...BS, j=1,2,...PS$).
2. Yiyecek kaynaklarının amaç fonksiyon değerleri hesaplanır (f_i).
3. Döngü başlangıç (sonlandırma kriteri sağlanana kadar)
4. Döngü başlangıç $i=1,2,...BS$ (İşçi arı aşaması)
5. Eğer $\text{rastgele}(0,1) < aep$ (Alternatif işçi arı aşaması)
6. Yiyecek kaynağı parametrelerinden biri rastgele değiştirilerek yeni yiyecek kaynağı üretilir (x_i').
7. Değilse (Standart işçi arı aşaması)
8. Denklem 2.2 kullanılarak x_i komşuluğunda x_i' kaynağı üretilir.
9. Yeni yiyecek kaynağının amaç fonksiyon değeri hesaplanır (f_i').
10. Eğer f_i' değeri f_i değerinden daha iyi ise
11. $x_i = x_i'$, yeni kaynak besin matrisine dahil edilir ve i . kaynağın geliştirilememe sayacı sıfırlanır ($l_i = 0$)
12. Değilse
13. x_i kaynağı tutulur ve geliştirilememe sayacı artırılır ($l_i = l_i + 1$)
14. Döngü sonlandırılır (i)
15. Döngü başlangıç $i=1,2,...BS$ (Uygunluk değeri atama)
16. Denklem 2.3 veya 2.4 kullanılarak uygunluk değeri atanır (u_i).
17. Döngü sonlandırılır (i)
18. Döngü başlangıç $i=1,2,...BS$ (Seçim oranı belirleme)
19. Denklem 2.5 kullanılarak seçim oranı belirlenir (s_i).
20. Döngü sonlandırılır (i)
21. Döngü başlangıç $i=1,2,...BS$ (Gözcü arı aşaması)
22. Eğer $\text{rastgele}(0,1) < s_i$
23. Denklem 2.2 kullanılarak seçilen yiyecek kaynağının komşuluğunda x_i' kaynağı üretilir.
24. Yeni kaynağın amaç fonksiyon değeri hesaplanır (f_i').
25. Eğer f_i' değeri f_i değerinden daha iyi ise
26. $x_i = x_i'$, yeni kaynak besin matrisine dahil edilir ve i . kaynağın geliştirilememe sayacı sıfırlanır ($l_i = 0$)
27. Değilse
28. x_i kaynağı tutulur ve geliştirilememe sayacı artırılır ($l_i = l_i + 1$)
29. Döngü sonlandırılır (i)
30. Döngü başlangıç $i=1,2,...BS/2$ (Çaprazlama operasyonu)
31. Eğer $\text{rastgele}(0,1) < cp$
32. Çaprazlama işlemi sonucu elde edilen iki yeni yiyecek kaynağı besin matrisine dahil edilir ($x_i', x_{(i + BS/2)}'$).

33. Yiyecek kaynaklarına ait amaç fonksiyon değerleri hesaplanır $(f'_i, f'_{(i+BS/2)})$.
34. Yiyecek kaynaklarına ait geliştirilememe sayaçları sıfırlanır $(l_i = l_{(i+BS/2)} = 0)$.
35. Döngü sonlandırılır (i)
36. Besin matrisindeki en iyi kötü yiyecek kaynağı ile elde edilen en iyi yiyecek kaynağı değiştirilir (Elitizm).
37. Elde edilen en iyi yiyecek kaynağı saklanır.
38. Döngü sonlandırılır (sonlandırma kriteri)
39. En iyi yiyecek kaynağı kaydedilir.

Not 1: aep değeri alternatif işçi aşamasının çalışma oranını ifade etmek için kullanılmaktadır.

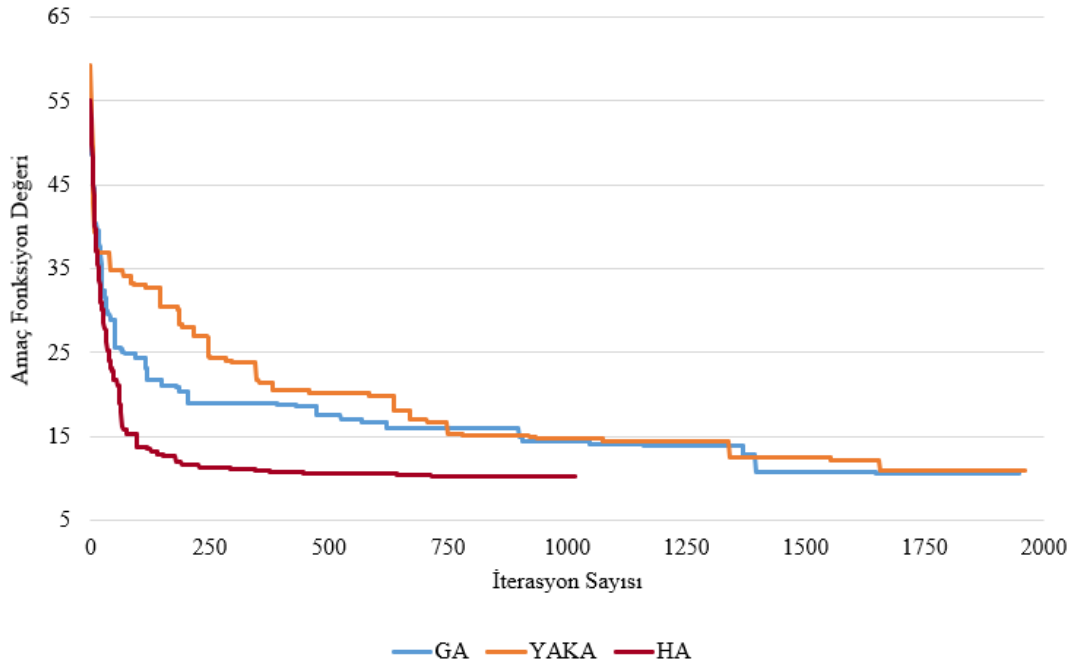
Not 2: cp değeri çaprazlama oranını ifade etmek için kullanılmaktadır.

5. DEPOLAMA/GERİ ALMA PROBLEMİNİN ÇÖZÜLMESİNE İLİŞKİN ELDE EDİLEN SONUÇLAR

DGA’da depolama ve geri alma siparişlerinin işlenmesi için oluşturulan depo ortamında deneysel çalışmalar gerçekleştirilmiş ve sonuçlar elde edilmiştir. Deponun farklı seviyelerde doluluk oranlarına göre depolama ve geri alma siparişleri işlenmiştir.

Yakınsama grafikleri, GA, YAKA ve HA’nın 100’er kez çalıştırılması sonucu elde edilen en iyi amaç fonksiyon değerlerine sahip olan sonuçları göstermektedir. Çizelgelerde yer alan değerler ise 100’er kez çalıştırma sonucunda elde edilen en iyi amaç fonksiyon değerlerini ve en iyi amaç fonksiyon değerlerinin ortalamasını göstermektedir. Çizelgelerde ayrıca ortalama iterasyon sayıları, ortalama çalışma ve ortalama iterasyon süreleri (bir iterasyon için) de verilmiştir.

Deneysel çalışmalarda, iterasyon sayıları sabit olarak verilmemiş bunun yerine belirli bir iterasyon sayısı kadar iyileşme olmaması durumu göz önüne alınmıştır. Önceden belirlenen sayı kadar iyileşme olmaması durumunda algoritma sonlandırılmaktadır. Bu bakımdan her algoritma için iterasyon sayıları farklı olmaktadır.



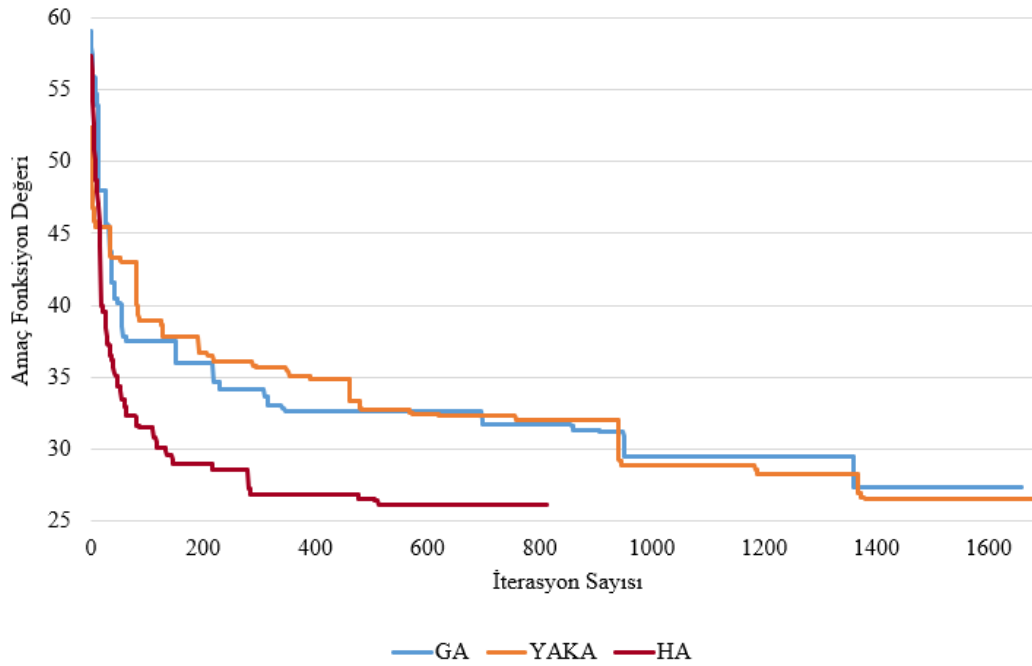
Şekil 5.1: Depolama siparişi yakınsama grafikleri (depo boş)

Şekil 5.1’de, depodaki tüm raflar boş olduğunda işlenen depolama siparişine ait yakınsama grafikleri görülmektedir. Depolama siparişi 13 paletten oluşmak üzere toplam 5 farklı ürün içermektedir.

Çizelge 5.1: Depolama siparişi sonuç değerleri (depo boş)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	13.05	10.55	1339	1.91	1.42×10^{-3}
YAKA	14.07	10.99	1626	4.86	2.98×10^{-3}
HA	10.57	10.25	740	2.39	3.22×10^{-3}

Çizelge 5.1’de görülen değerler, depodaki tüm raflar boş olduğunda işlenen depolama siparişi için elde edilmiştir.



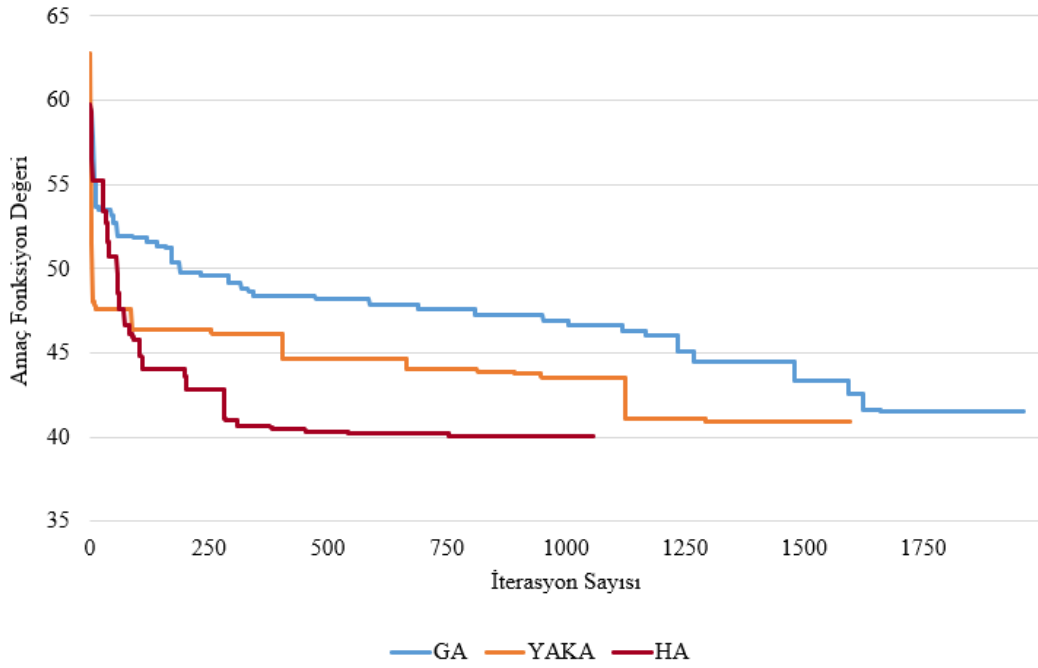
Şekil 5.2: Depolama siparişi yakınsama grafikleri (%25 dolu)

Şekil 5.2’de depo doluluk oranı %25 iken işlenen depolama siparişine ait yakınsama grafikleri görülmektedir. Depolama siparişi 9 paletten oluşmak üzere toplam 5 farklı ürün içermektedir.

Çizelge 5.2: Depolama siparişi sonuç değerleri (%25 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	31.84	27.36	1289	19.94	15.46×10^{-3}
YAKA	35.81	26.59	708	26.50	37.42×10^{-3}
HA	29.95	26.09	558	23.57	42.24×10^{-3}

Çizelge 5.2’de görülen değerler, depo doluluk oranı %25 olduğunda işlenen depolama siparişi için elde edilmiştir.



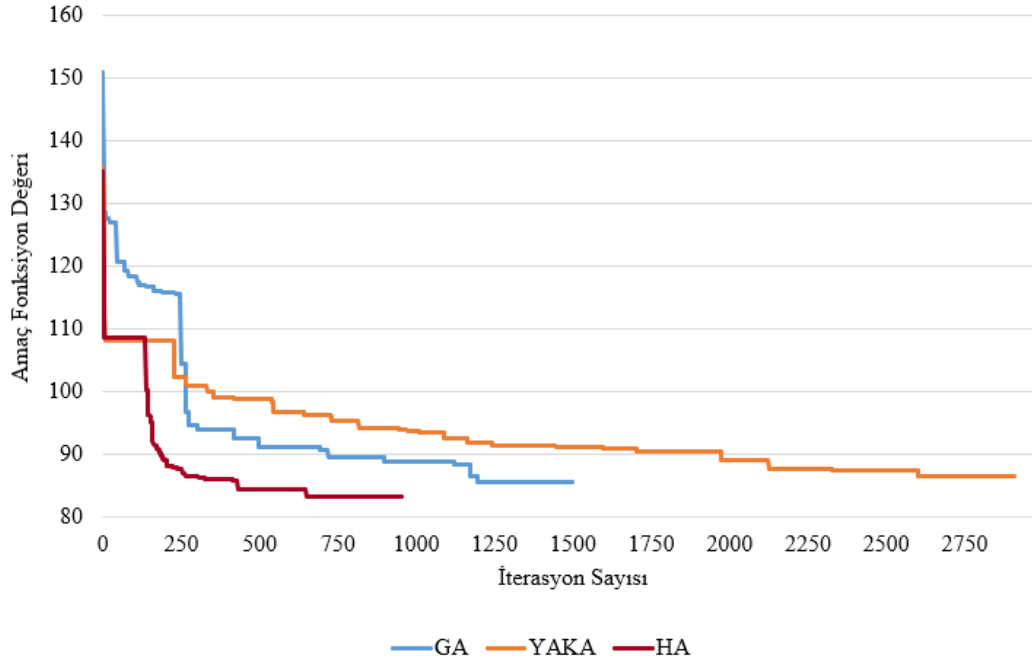
Şekil 5.3: Depolama siparişi yakınsama grafikleri (%50 dolu)

Şekil 5.3’de depo doluluk oranı %50 iken işlenen depolama siparişine ait yakınsama grafikleri görülmektedir. Depolama siparişi 7 paletten oluşmak üzere toplam 3 farklı ürün içermektedir.

Çizelge 5.3: Depolama siparişi sonuç değerleri (%50 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	45.47	41.52	1121	24.16	21.55×10^{-3}
YAKA	47.83	40.91	718	30.50	42.47×10^{-3}
HA	42.26	40.02	578	27.48	47.54×10^{-3}

Çizelge 5.3’de görülen değerler, depo doluluk oranı %50 iken işlenen depolama siparişi için elde edilmiştir.



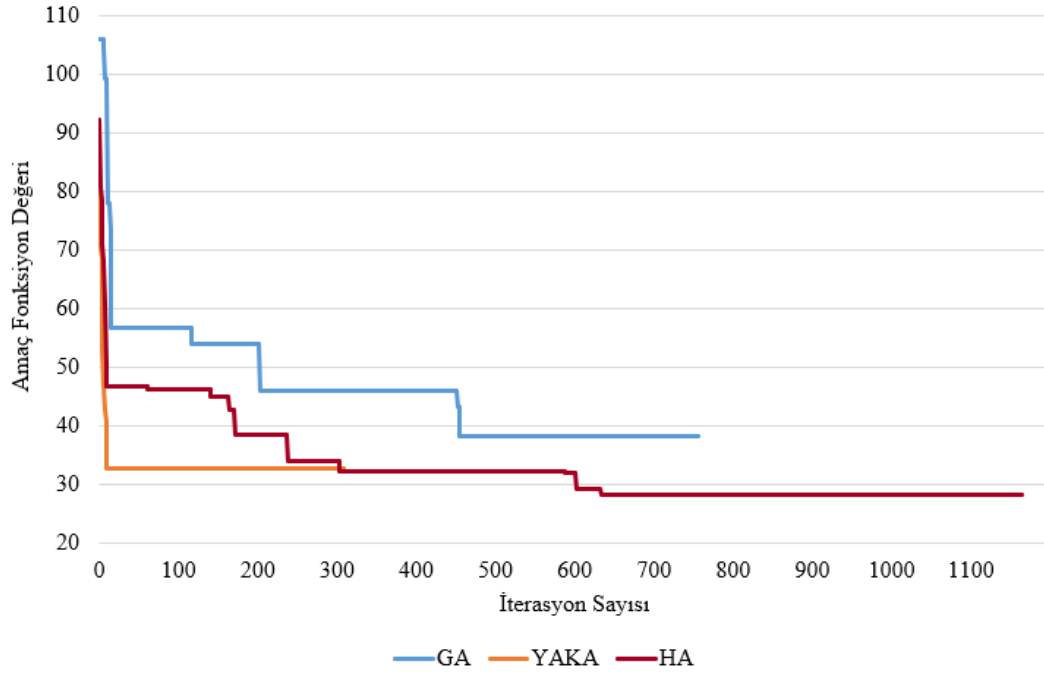
Şekil 5.4: Depolama siparişi yakınsama grafikleri (%75 dolu)

Şekil 5.4’de depo doluluk oranı %75 iken işlenen depolama siparişine ait yakınsama grafikleri görülmektedir. Depolama siparişi 11 paletten oluşmak üzere toplam 4 farklı ürün içermektedir.

Çizelge 5.4: Depolama siparişi sonuç değerleri (%75 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	91.43	85.69	1475	79.24	53.75×10^{-3}
YAKA	100.19	86.52	739	81.48	110.25×10^{-3}
HA	87.93	83.20	664	79.22	119.30×10^{-3}

Çizelge 5.4’de görülen değerler, depo doluluk oranı %75 iken işlenen depolama siparişi için elde edilmiştir.



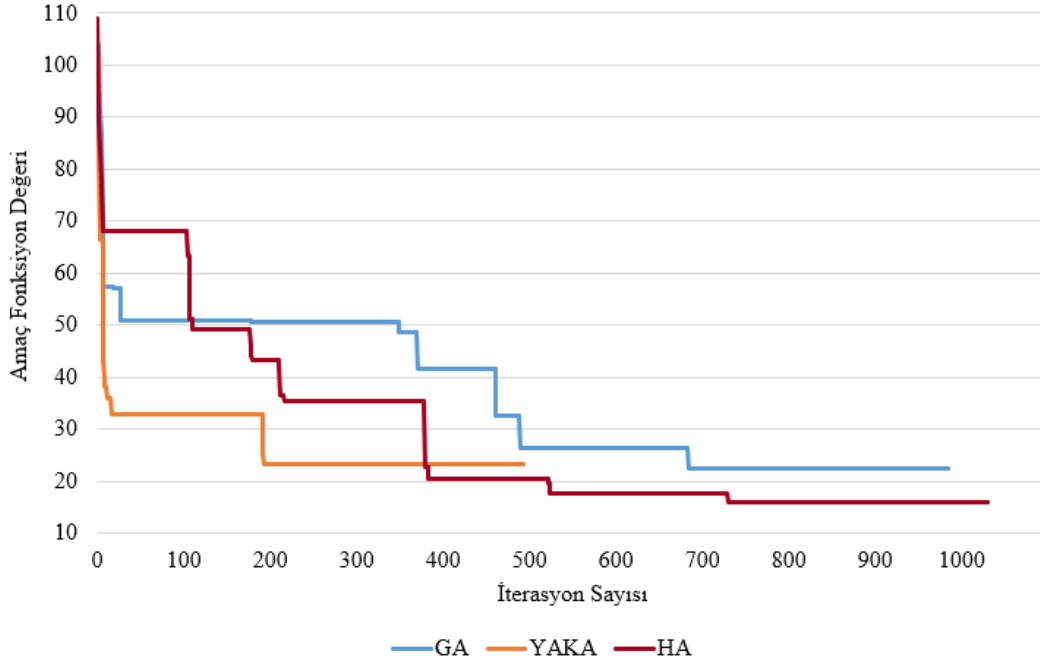
Şekil 5.5: Geri alma siparişi yakınsama grafikleri (%25 dolu)

Şekil 5.5’de depo doluluk oranı %25 iken işlenen geri alma siparişine ait yakınsama grafikleri görülmektedir. Geri alma siparişi 9 paletten oluşmak üzere toplam 5 farklı ürün içermektedir.

Çizelge 5.5: Geri alma siparişi sonuç değerleri (%25 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	61.37	38.33	705	0.41	0.58×10^{-3}
YAKA	64.85	32.77	334	0.38	1.13×10^{-3}
HA	58.15	28.22	452	0.53	1.17×10^{-3}

Çizelge 5.5’de görülen değerler, depo doluluk oranı %25 iken işlenen geri alma siparişi için elde edilmiştir.



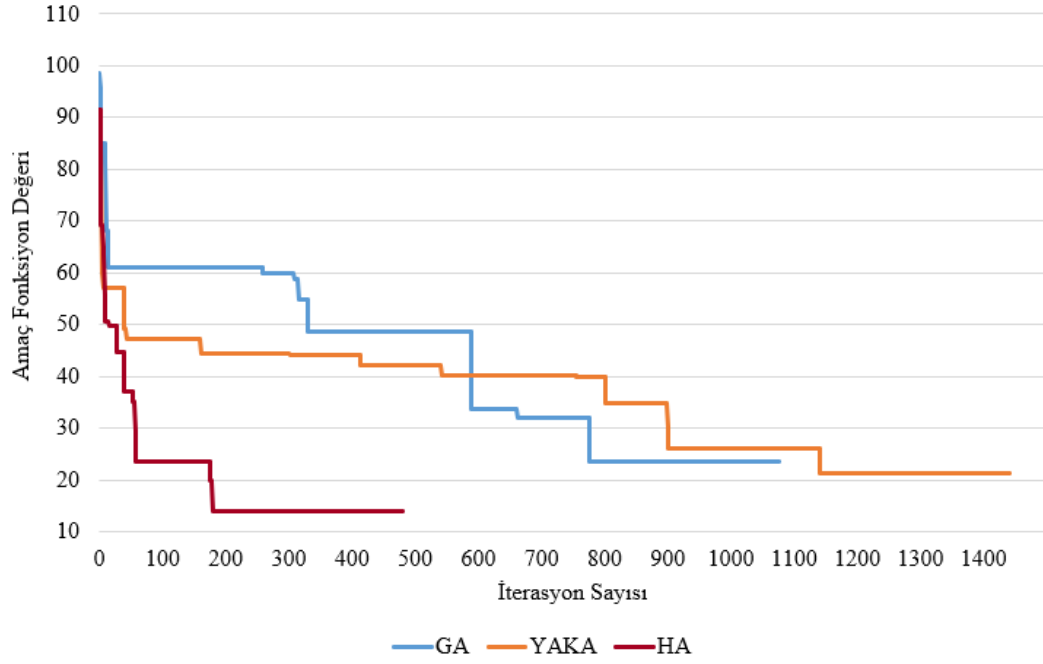
Şekil 5.6: Geri alma siparişi yakınsama grafikleri (%50 dolu)

Şekil 5.6’da depo doluluk oranı %50 iken işlenen geri alma siparişine ait yakınsama grafikleri görülmektedir. Geri alma siparişi 7 paletten oluşmak üzere toplam 3 farklı ürün içermektedir.

Çizelge 5.6: Geri alma siparişi sonuç değerleri (%50 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	48.76	22.42	787	0.37	0.47×10^{-3}
YAKA	50.23	23.14	378	0.34	0.89×10^{-3}
HA	33.55	16	710	0.67	0.94×10^{-3}

Çizelge 5.6’da görülen değerler, depo doluluk oranı %50 iken işlenen geri alma siparişi için elde edilmiştir.



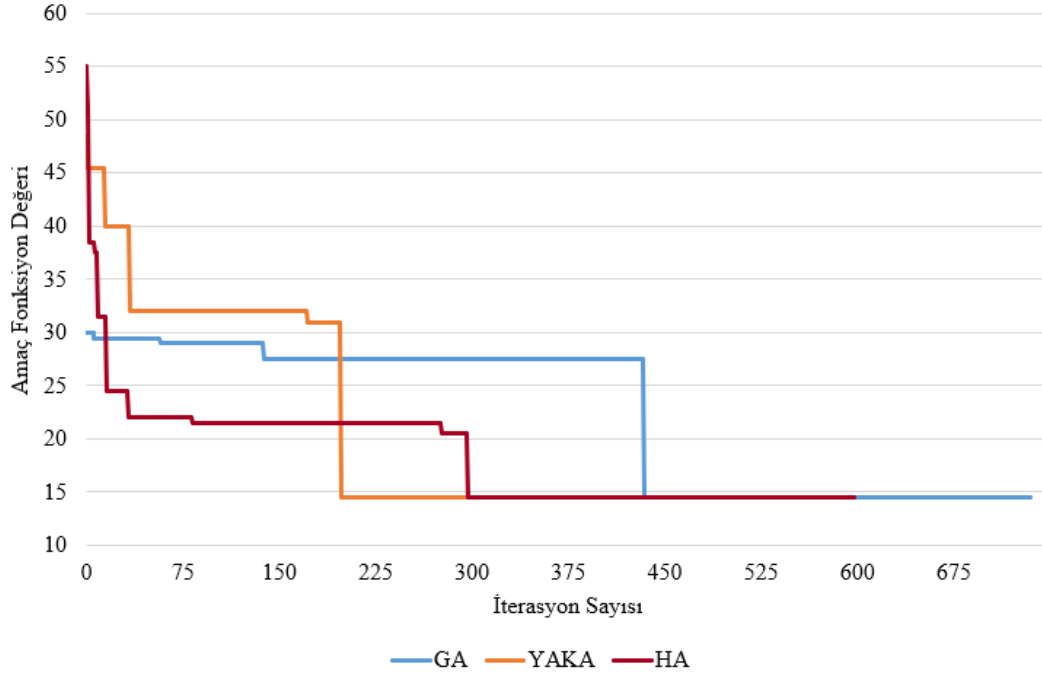
Şekil 5.7: Geri alma siparişi yakınsama grafikleri (%75 dolu)

Şekil 5.7’de depo doluluk oranı %75 iken işlenen geri alma siparişine ait yakınsama grafikleri görülmektedir. Geri alma siparişi 11 paletten oluşmak üzere toplam 4 farklı ürün içermektedir.

Çizelge 5.7: Geri alma siparişi sonuç değerleri (%75 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	45.98	23.40	684	0.24	0.35×10^{-3}
YAKA	47.99	21.40	410	0.28	0.68×10^{-3}
HA	30.09	14	578	0.43	0.74×10^{-3}

Çizelge 5.7’de görülen değerler, depo doluluk oranı %75 iken işlenen geri alma siparişi için elde edilmiştir.



Şekil 5.8: Geri alma siparişi yakınsama grafikleri (%90 dolu)

Şekil 5.8’de depo doluluk oranı %90 iken işlenen geri alma siparişine ait yakınsama grafikleri görülmektedir. Geri alma siparişi 8 paletten oluşmak üzere toplam 4 farklı ürün içermektedir.

Çizelge 5.8: Geri alma siparişi sonuç değerleri (%90 dolu)

Algoritma	Amaç Fonksiyon Değeri		İterasyon Sayısı	Süre (sn)	İterasyon Süresi (sn)
	Ortalama	En İyi			
GA	26.49	14.50	390	0.06	0.15×10^{-3}
YAKA	21.58	14.50	462	0.16	0.34×10^{-3}
HA	14.83	14.50	435	0.19	0.43×10^{-3}

Çizelge 5.8’de görülen değerler, depo doluluk oranı %90 iken işlenen geri alma siparişi için elde edilmiştir.

Farklı doluluk seviyelerinde işlenen depolama ve geri alma siparişleri incelendiğinde aşağıda verilen sonuçlar elde edilmektedir:

- Depolama amaç fonksiyon değeri hesaplama işlemi, geri alma amaç fonksiyon hesaplama işlemine göre daha karmaşık bir işlemidir. Bu sebeple depolama siparişlerinin işleme süresi, geri alma siparişlerinin işleme süresine göre daha fazladır.

- Depolama siparişlerinde depo doluluk oranı arttıkça boş raf bulmak zorlaşacağından dolayı hesaplama süresi artmaktadır. Ayrıca ürün uyum kısıtından dolayı depodaki tüm ürünlerin birbirlerine yakın yerleştirilmesi gerekmektedir. Depo doluluk oranı arttıkça ürün uyum kısıtından dolayı yine hesaplama süresi artmaktadır.
- Geri alma siparişlerinde depo doluluk oranı arttıkça, geri alınacak ürün seçeneği daha fazla olacağı için sonuca daha hızlı ulaşılmaktadır.
- Meta-sezgisel yöntemlerde yakınsama grafiklerinde ani düşüşler gözlenmemektedir. İterasyon sayısı arttıkça ufak iyileşmelerle en iyi sonuçlar elde edilmektedir. Ancak bu bölümde verilen yakınsama grafiklerinde amaç fonksiyon değerlerinde ani düşüşler gözlenmektedir. Bu durum verilen ceza puanlarının amaç fonksiyon değerlerini büyütmesinden kaynaklanmaktadır. Ceza puanlarına sahip aday çözümden, ceza puanlarından arınmış başka bir çözüm elde edildiği zaman ani düşüşlerin gözlenmesi durumuna eklenen ceza puanları sebep olmaktadır.

Sonuç olarak tüm meta-sezgisel yöntemlerin, operasyonel kısıtlara sahip depoda başarılı bir şekilde çalıştığı görülmektedir.



6. SONUÇLAR ve ÖNERİLER

21. yüzyılda gerçekleşen teknoloji atılımı ile birlikte sanayi, endüstri gibi bir çok alanda bilgisayar kullanımı yaygınlaşmıştır. Bilgisayarların işlem kapasitelerinin artması, insan gücü kullanılarak yapılamayacak işlemlerin milisaniyeler mertebesinde yapılabilmesine imkan sağlamıştır.

Günümüz teknolojiyle elde edilen işlem kapasitesinin, insan zekasının verebileceği kararların yazılımlar tarafından verilebilmesine yönelik araştırmalar yapılmaktadır. Bilgisayarlar insan beyninden daha hızlı çalışmaktadırlar. Tüm olasılıkları deneme yanılma yöntemiyle en doğru seçimleri yapabilirler, ancak problem karmaşıklığının artması durumunda en doğru seçimleri yapabilmek için işlem kapasiteleri yetersiz kalmaktadır. Bilgisayarlar, deneme yanılma yönteminin doğru alanlara yönlendirilebilmesi için insanlar tarafından konulan bazı kurallara ihtiyaç duyarlar.

Meta-sezgisel algoritmalar; belirli kurallar çerçevesinde çözüm uzayı geniş ve karmaşık olan problemlerde, bilgisayarların işlem kapasitesine doğru yönlendirmeler yaparak en iyi sonucu elde etmeye çalışmaktadır. En iyi sonucu üretmeyi garanti etmese dahi, doğru ifade edilmiş bir problem için en iyiye yakın sonuçlar üretmektedirler.

Tez çalışmasında, literatürde birçok çalışmaya konu olan DGA sistemleri için meta-sezgisel algoritmalar kullanılarak çözüm üretilmesi amaçlanmıştır. Literatürde farklı çalışmalarda ayrı ayrı kullanılmış olan DGA kısıtları tek bir çalışma altında toplanarak sanal bir depo ortamı tasarlanmıştır. Oluşturulan karmaşık optimizasyon problemi üzerinde, geleneksel metotlarla çözilemeyen ya da çözülmesi çok uzun süren problemler için sıkça tercih edilen meta-sezgisel algoritmalarından GA ve YAKA yöntemleri kullanılmıştır. GA ve YAKA yöntemleriyle elde edilen sonuçlardan daha iyi sonuçlar elde etmek amacıyla HA yaklaşımı da sunulmuş ve problemin çözümü için daha iyi sonuçlar elde edilmiştir. Farklı meta-sezgisel yöntemler, literatürde birçok çalışmada olduğu gibi, bir araya gelerek ayrı ayrı çalıştırıldıklarından daha iyi sonuçlar elde edebildikleri bir kez daha gösterilmiştir.

DGA sistemlerinde depolama ve geri alma siparişlerinin işlenmesi birçok kısıta sahip karmaşık bir optimizasyon problemi olması nedeniyle, farklı meta-sezgisel yöntemlerin karşılaştırılması amacıyla kullanılabilir depolama ve geri alma fonksiyonu ifade edilmiştir.

Çalışmanın odak noktası hem depolama hem de geri alma işlemleri için geliştirilmiş bir fonksiyon oluşturularak, bu fonksiyonun farklı meta-sezgisel yöntemlerle başarılı bir şekilde kullanılabileceğini göstermektir. Literatürde kullanılan kısıtlara yer verilerek tüm depolarda uygulanabilir hale gelmesi amaçlanmıştır. Uygulanmak istenen depoda yer almayan kısıtlar denklemden çıkarılarak farklı optimizasyon fonksiyonları elde edilebilir. Literatürde birçok çalışmada ayrı ayrı kullanılan depo kısıtları tek bir çalışmada toplanmıştır.



KAYNAKLAR

- [1] M. Taskiran, N. Kahraman, C. E. Erdem, Face recognition: Past, present and future (a review), *Digital Signal Processing*. 106 (2020).
- [2] A. Hicdurmaz: Gezin robotun görme tabanlı ve gerçek zamanlı olarak engelden sakınımı, Yüksek Lisans Tezi, Yalova Üniversitesi, Fen Bilimleri Enstitüsü, Yalova, 2020, 637269.
- [3] M. N. Zafar, J. C. Mohanta, Methodology for path planning and optimization of mobile robots: A review, *Procedia Computer Science*. 133 (2018) 141-152.
- [4] K. J. Roodbergen, I. F. A. Vis, A survey of literature on automated storage and retrieval systems, *European journal of Operational Research*. 194 (2009) 343-362.
- [5] E. A. Silver, An overview of heuristic solution methods, *Journal of the Operational Research Society*. 55 (2004) 936-956.
- [6] M. Mestria, New hybrid heuristic algorithm for the clustered traveling salesman problem, *Computers & Industrial Engineering*. 116 (2018) 1-12.
- [7] A. M. Atieh, H. Kaylani, Y. Al-abdallat, A. Qaderi, L. Ghoul, L. Jaradat, I. Hdairis, Performance improvement of inventory management system processes by an automated warehouse management system, *Procedia CIRP*. 41 (2016) 568-572.
- [8] S. Seifermann, J. Bollhoff, J. Metternich, A. Bellaghnach, Evaluation of work measurement concepts for a cellular manufacturing reference line to enable low cost automation for lean machining, *Procedia CIRP*. 17 (2014) 588-593.
- [9] S. D. Bektas, Otomatik depolama ve geri-alma sistemlerinin simülasyon ile performans analizi, Yüksek Lisans Tezi, Bursa Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Bursa, 2019, 543438.
- [10] E. Balkan, Otomatik depolama ve geri alma (AS/RS) sistemlerinde çevrim süresi hesabı için algoritma ve arayüz önerisi, Yüksek Lisans Tezi, Kara Harp Okulu Komutanlığı, Savunma Bilimleri Enstitüsü, Ankara, 2012, 346846.
- [11] S. Trab, E. Bajic, A. Zounkhi, M. N. Abdelkerim, H. Chekir, L. T. Ltaief, Product allocation planning with safety compatibility constraints in IoT-based warehouse, *Procedia Computer Science*. 73 (2015) 290-297.
- [12] M. Kazemi, A. Asef-vaziri, T. Shojaei, Concurrent optimization of shared location assignment and storage/retrieval scheduling in multi-shuttle automated storage and retrieval systems, *IFAC-PapersOnLine*. 52 (2019) 2531-2536.

- [13] P. Yang, L. Miao, Z. Xue, B. Ye, Variable neighborhood search heuristic for storage location assignment and storage/retrieval scheduling under shared storage in multi-shuttle automated storage/retrieval systems, *Transportation Research Part E: Logistics and Transportation Review*. 79 (2015) 164-177.
- [14] J. D. V. D. Berg, A. J. R. M. Gademann, Optimal routing in an automated storage/retrieval system with dedicated, *IIE Transactions*, 31 (1999) 407-415.
- [15] J. D. V. D. Berg, A. J. R. M. Gademann, Simulation study of an automated storage/retrieval system, *International Journal of Production Research*, 38 (2000) 1339-1356.
- [16] C. Kasemset, J. Sudphan, Warehouse storage assignment: The case study of a plastic bag manufacturer, 2014 IEEE International Conference on Industrial Engineering and Engineering Management, 2014, Bandar Sunway, Malaysia.
- [17] S. T. Hackman, M. J. Rosenblatt, J. M. Olin, Allocating items to an automated storage and retrieval system, *IIE Transactions*. 22, (1990) 7-14.
- [18] S. S. Heragu, L. Du, R. J. Mantel, P. C. Schuur, Mathematical model for warehouse design and product allocation, *International Journal of Production Research*. 43 (2005) 327-338.
- [19] F. Guerriero, R. Musmanno, O. Pisacane, F. Rende, A mathematical model for the multi-levels product allocation problem in a warehouse with compatibility constraints, *Applied Mathematical Modelling*. 37 (2013) 4385-4398.
- [20] O. Sanei, V. Nasiri, M. R. Marjani, S. M. Moattar Hussein, A heuristic algorithm for the warehouse space assignment problem considering operational constraints: with application in a case study, 2011 International Conference on Industrial Engineering and Operations Management, 2011, Kuala Lumpur, Malaysia.
- [21] W. Hu, Y. Wang, J. Zheng, Research on warehouse allocation problem based on the artificial bee colony inspired particle swarm optimization (ABC-PSO) algorithm, *Fifth International Symposium on Computational Intelligence and Design*. 1 (2012) 173-176.
- [22] M. E. Hiri, A. Ennadi, A. Chafi, Order allocation using genetic algorithm, *Journal of Engineering Science and Technology Review*. 13 (2020) 127-134.
- [23] F. C. Yang, K. Chen, M. T. Wang, P. Y. Chang, K. C. Sun, The International Journal of Advanced Manufacturing Technology. 51 (2010) 1251–1259.

- [24] G. Zhou, L. Mao, Design and simulation of storage location optimization module in AS/RS based on FLEXSIM, *International Journal of Intelligent Systems and Applications*. 2 (2010) 33-40.
- [25] J.-P. Gagliardi, J. Renaud, A. Ruiz, On storage assignment policies for unit-load automated storage and retrieval systems, *International Journal of Production Research*. 50 (2010) 879-892.
- [26] A. Meneghetti, L. Monti, Sustainable storage assignment and dwell-point policies for automated storage and retrieval systems, *Production Planning & Control*. 24 (2011) 511-520.
- [27] R. D. Koster, T. Le-Duc, K. J. Roodbergen, Design and control of warehouse order picking: A literature review, *European Journal of Operational Research*. 182 (2007) 481-501.
- [28] D. Karaboga, An idea based on honey bee swarm for numerical optimization, Technical report-tr06. 200 (2005) 1-10.
- [29] D. Karaboga, B. Akar, A comparative study of artificial bee colony algorithm, *Applied mathematics and computation*. 214 (2009) 108-132.
- [30] B. Akay, Nümerik optimizasyon problemlerinde yapay arı kolonisi (artificial bee colony) algoritmasının performans analizi, Doktora Tezi, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, 2009, 246078.
- [31] J. H. Holland, *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control and artificial intelligence*, MIT Press, Cambridge, 1992.
- [32] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed., Addison-Wesley Publishing Company Inc., Boston, MA, USA, 1989.
- [33] A. Tuncer, M. Yıldırım, K. Erkan, A hybrid implementation of genetic algorithm for path planning of mobile robots on FPGA, In *Computer and Information Sciences III*. (2013) 459-465.
- [34] P. Kora, P. Yadlapalli, Crossover operators in genetic algorithms: A review, *International Journal of Computer Applications*. 162 (2017) 34-36.
- [35] S. Katoch, S. S. Chauhan, V. Kumar, A review on genetic algorithm: past, present, and future, *Multimedia Tools and Applications*. (2020) 1-36.

- [36] A. Hassanat, K. Almohammadi, E. Alkafaween, E. Abunawas, A. Hammouri, V. B. Prasath, Choosing Mutation and Crossover Ratios for Genetic Algorithms-A Review with a New Dynamic Approach, *Information*. 10 (2019) 390.
- [37] A. Tuncer, M. Yildirim, Chromosome coding methods in genetic algorithm for path planning of mobile robots, In *Computer and Information Sciences II*. (2011) 377-383.
- [38] V. Benoit, T. Nakashima, A new approach to path-finding by possibilities search, 2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS), 2014, Kitakyushu, Japan.
- [39] C. Wang, L. Wang, J. Qin, Z. Wu, L. Duan, Z. Li, M. Cao, X. Ou, X. Su, W. Li, Z. Lu, M. Li, Y. Wang, J. Long, M. Huang, Y. Li, Q. Wang, Path planning of automated guided vehicles based on improved a-star algorithm, *International Conference on Information and Automation*, 2015, Lijiang, China.
- [40] J. Yim, M. Kim, S. Shin, J. Park, Robust path planning for autonomous vehicle in position uncertainty, *The 11th International Conference on Ubiquitous Robots and Ambient Intelligence*, 2014, Kuala Lumpur, Malaysia.
- [41] R. Tianzhu, Z. Rui, D. Zhuoning, Three-dimensional path planning of UAV based on an improved A* algorithm, *Proceedings of 2016 IEEE Chinese Guidance, Navigation and Control Conference*, 2016, Nanjing, China.

ÖZGEÇMİŞ

Adı Soyadı: Hüseyin YILMAZ

Lisans: Yalova Üniversitesi, Bilgisayar Mühendisliği

TEZDEN ÜRETİLEN YAYINLAR

- [1] H. Yılmaz, A. Tuncer, Genetic algorithm based storage and retrieval system optimization considering operational constraints in a multidimensional warehouse, International Journal of Applied Mathematics Electronics and Computers. 8 (2020) 148-153.