

**T.C.  
PAMUKKALE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**ELEMAN SAYISI KISITLI PORTFÖY OPTİMİZASYONU  
İÇİN DEĞİŞKEN KOMŞULUK ARAMA ALGORİTMASI  
TEMELLİ BİR ÇÖZÜM YAKLAŞIMI**

**YÜKSEK LİSANS TEZİ**

**MEHMET ANIL AKBAY**

**DENİZLİ, ARALIK - 2019**

**T.C.  
PAMUKKALE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**



**ELEMAN SAYISI KISITLI PORTFÖY OPTİMİZASYONU  
İÇİN DEĞİŞKEN KOMŞULUK ARAMA ALGORİTMASI  
TEMELLİ BİR ÇÖZÜM YAKLAŞIMI**

**YÜKSEK LİSANS TEZİ**

**MEHMET ANIL AKBAY**

**DENİZLİ, ARALIK - 2019**

## KABUL VE ONAY SAYFASI

Mehmet Anıl Akbay tarafından hazırlanan "Eleman Sayısı Kısıtlı Portföy Optimizasyonu İçin Değişken Komşuluk Arama Algoritması Temelli Bir Çözüm Yaklaşımı" adlı tez çalışmasının savunma sınavı 27.12.2019 tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği ile Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Üyeleri

İmza

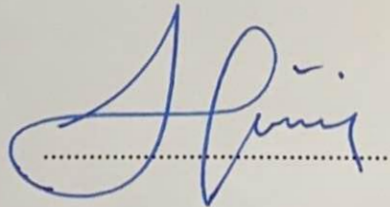
Danışman  
Doç. Dr. Can Berk KALAYCI

Üye  
Doç. Dr. Olcay POLAT

Üye  
Dr. Öğretim Üyesi Erdal AYDEMİR



Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 02/01/2020 tarih ve 2020/08.... sayılı kararıyla onaylanmıştır.

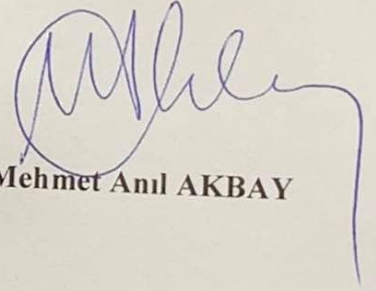


Prof. Dr. Uğur YÜCEL

Fen Bilimleri Enstitüsü Müdürü

**Bu tez çalışması TÜBİTAK tarafından 214M224 nolu proje ile desteklenmiştir.**

Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu alıřmanın dođrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan alıřmalara atfedildiđine beyan ederim.



Mehmet Anıl AKBAY

## ÖZET

**ELEMAN SAYISI KISITLI PORTFÖY OPTİMİZASYONU İÇİN  
DEĞİŞKEN KOMŞULUK ARAMA ALGORİTMASI TEMELLİ BİR  
ÇÖZÜM YAKLAŞIMI  
YÜKSEK LİSANS TEZİ  
MEHMET ANIL AKBAY  
PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ  
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**(TEZ DANIŞMANI:DOÇ. DR. CAN BERK KALAYCI)**

**DENİZLİ, ARALIK - 2019**

Yıllardır portföy optimizasyonu gerek yatırımcılar gerekse araştırmacılar için yatırım faaliyetlerinde temel karar verme stratejilerinden birisi olarak kullanılmaya devam etmektedir. Bu alanda en bilindik ve en yaygın olarak kullanılan yöntemlerden birisi de Harry Markowitz tarafından önerilen ortalama varyans yaklaşımıdır. Bu öncü çalışmanın ardından, birçok araştırmacı modelin daha pratik ve gerçek hayat problemlerine daha gerçekçi çözümler üretebilmesi için çeşitli varyasyonlarını geliştirmiştir. Bu çalışma kapsamında, bu varyasyonlardan birisi olan eleman sayısı kısıtlı portföy optimizasyonu problemi ele alınmıştır. Eleman sayısı kısıtı, orijinal kuadratik optimizasyon modelini NP-Zor sınıfında olduğu kanıtlanmış karma tam sayılı kuadratik programlama modeline dönüştürmekte böylelikle klasik kesin çözüm metodolojileri kullanılarak kabul edilebilir zaman dilimlerinde optimal çözümün bulunabilmesini zorlaştırmaktadır. Bu nedenle, araştırmacıların büyük çoğunluğu bahsedilen hesaplama zorluklarının üstesinden gelebilmek için makul sürelerde optimale yakın çözümler üretebilen yakınsama temelli algoritmalarından yararlanmaktadırlar. Bu çalışmada, eleman sayısı kısıtlı portföy optimizasyonu probleminin çözümü için kuadratik programlama ile hibritlenmiş paralel değişken komşuluk arama algoritması önerilmiştir. Önerilen bu iki aşamalı çözüm yaklaşımında değişken komşuluk arama algoritması portföye seçilecek varlık kombinasyonlarını belirlerken, varlıkların ağırlıkları ise kuadratik programlama aracılığıyla hesaplanmıştır. Literatürde sıklıkla kullanılan beş farklı veri seti üzerinde yapılan testler ve literatürdeki diğer çözüm yaklaşımları ile karşılaştırmalı analizler neticesinde önerilen çözüm yaklaşımının son derece rekabetçi sonuçlar verdiği ve özellikle düşük riskli portföylerde daha etkili olduğu tespit edilmiştir.

**ANAHTAR KELİMELEER:** portföy optimizasyonu, eleman sayısı kısıtı, meta-sezgiseller, değişken komşuluk arama, asenkron paralelleştirme, kuadratik programlama

## **ABSTRACT**

### **A VARIABLE NEIGHBORHOOD SEARCH BASED SOLUTION APPROACH FOR CARDINALITY CONSTRAINT PORTFOLIO OPTIMIZATION**

**MSC THESIS**

**MEHMET ANIL AKBAY**

**PAMUKKALE UNIVERSITY INSTITUTE OF SCIENCE  
INDUSTRIAL ENGINEERING**

**(SUPERVISOR:ASSOC. PROF. DR. CAN BERK KALAYCI)**

**DENİZLİ, DECEMBER 2019**

Over the years, portfolio optimization remains as an important decision-making strategy for investment. The most familiar and widely used approach in the field of portfolio optimization is the mean-variance framework introduced by Markowitz. Following this pioneering work, many researchers have extended this model to make it more practical and adapt to real-life problems. In this study, one of these extensions, cardinality constrained portfolio optimization problem, is considered. Cardinality constraints transform the quadratic optimization model into the mixed-integer quadratic programming problem, which is proved to be NP-Hard, making it harder to obtain an optimal solution within a reasonable time by using exact solution methodologies. Hence, the vast majority of the researchers have taken advantage of approximate algorithms in order to overcome arising computational difficulties. In order to develop an efficient solution approach for cardinality constrained portfolio optimization, in this study, a parallel variable neighborhood search algorithm combined with quadratic programming is proposed. While the variable neighborhood search algorithm decides the combination of assets to be held in the portfolio, quadratic programming quickly calculates the proportions of assets. The performance of the proposed algorithm is tested on five well-known datasets and compared with other solution approaches in the literature. Obtained results reveal that the proposed solution approach is competitive with the state-of-the-art algorithms and very efficient especially on the portfolios with low risk.

**KEYWORDS:** portfolio optimization, cardinality constraints, metaheuristics, variable neighborhood search, asynchronous parallelization, quadratic programming

# İÇİNDEKİLER

Sayfa

<b>ÖZET</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>İÇİNDEKİLER</b> .....	<b>iii</b>
<b>ŞEKİL LİSTESİ</b> .....	<b>v</b>
<b>TABLO LİSTESİ</b> .....	<b>vi</b>
<b>ÖNSÖZ</b> .....	<b>vii</b>
<b>1. GİRİŞ</b> .....	<b>1</b>
<b>2. LİTERATÜR TARAMASI</b> .....	<b>4</b>
2.1 Araştırma Metodolojisi.....	5
2.1.1 Materyal Toplama .....	5
2.1.2 Betimsel Analiz.....	6
2.1.3 Sınıflandırma .....	8
2.1.4 Materyal Değerlendirme .....	9
2.2 OVPO Modeli.....	9
2.2.1 Tek-amaçlı OVPO modeli .....	9
2.2.2 Çok-amaçlı OVPO modeli.....	10
2.2.2.1 Çok-amaçlı OVPO modeli için yaklaşımlar .....	12
2.2.3 Tek-amaçlı ve Çok-amaçlı OV modellerinin karşılaştırılması ....	15
2.2.4 Gerçek hayat kısıtları .....	17
2.3 Veri .....	19
2.4 Performans Ölçütleri .....	23
2.5 OVPO Modeli Üzerine Uygulamalar .....	28
2.5.1 Kesin Çözüm Teknikleri.....	29
2.5.2 Kesin Olmayan Çözüm Teknikleri .....	30
2.5.2.1 Meta-sezgisel yöntemler .....	30
2.5.2.1.1 Popülasyon tabanlı algoritmalar .....	31
2.5.2.1.2 Tek-çözüm tabanlı algoritmalar.....	40
2.5.2.2 Makine öğrenmesi algoritmaları .....	41
2.5.3 OVPO problemine uygulanan çözüm yaklaşımlarına genel bir bakış	42
2.5.4 Hibrit Çözüm Teknikleri.....	44
2.5.5 Kısıt işleme teknikleri.....	45
<b>3. YÖNTEM</b> .....	<b>48</b>
3.1 Eleman Sayısı Kısıtlı Portföy Optimizasyonu.....	48
3.2 Değişken Komşuluk Arama Algoritması ve Paralleleştirme Stratejileri	49
3.3 Önerilen Çözüm Yaklaşımı .....	51
3.3.1 Varlık Seçimi .....	53
3.3.1.1 Başlangıç çözümünün ve Arama havuzunun elde edilmesi....	53
3.3.1.2 Çalkalama.....	55
3.3.1.3 Yerel Arama .....	56
3.3.1.4 Paralleleştirme .....	57
3.3.1.5 Varlık Oranlarının Belirlenmesi.....	58
3.4 Hesaplamalı Sonuçlar .....	59
3.4.1 Uygulama.....	59
3.4.2 Test Problemleri.....	59



3.4.3	Parametre Ayarlama .....	59
3.4.4	Performans Ölçütleri.....	62
3.4.5	Hesaplamalı Sonuçlar .....	64
<b>4.</b>	<b>SONUÇLAR VE TARTIŞMA .....</b>	<b>71</b>
<b>5.</b>	<b>KAYNAKLAR.....</b>	<b>73</b>
<b>6.</b>	<b>ÖZGEÇMİŞ.....</b>	<b>101</b>

## ŞEKİL LİSTESİ

### Sayfa

Şekil 2.1: Yayınların yıllara göre dağılımları .....	7
Şekil 2.2: Yayınların, yayınlandığı dergilere göre dağılımı.....	7
Şekil 2.3: Bildirilerin yayın yerine göre dağılımları .....	8
Şekil 2.4: Kitap bölümlerin yayın yerine göre dağılımı.....	8
Şekil 2.5: OVPO model ve uygulamalarının sınıflandırılması .....	8
Şekil 2.6: OVPO için tek ve çok-amaçlı modellerin yıllara göre dağılımı .....	15
Şekil 2.7: OV modelinde kullanılan gerçek hayat kısıtlarının dağılımı.....	18
Şekil 2.8: Veri setlerinin dağılımı .....	23
Şekil 2.9: Performans göstergelerinin dağılımı.....	26
Şekil 2.10: OVPO için uygulanan çözüm tekniklerinin sınıflandırılması.....	1
Şekil 2.11: OVPO problemine uygulanan çözüm tekniklerinin dağılımı .....	43
Şekil 2.12: Kısıt işleme tekniklerinin dağılımı .....	45
Şekil 3.1: Değişken komşuluk arama algoritması (Hansen ve Mladenovic 2001)	50
Şekil 3.2: Önerilen çözüm yaklaşımına ait sözde kod .....	52
Şekil 3.3: FTSE 100 veri seti için örnek bir başlangıç çözümü ve havuz büyüklüğü gösterimi.....	54
Şekil 3.4: Sıralı komşuluk değişim mekanizması .....	56
Şekil 3.5: Çalkalama aşamasına ait sözde kod.....	56
Şekil 3.6: Yerel arama sözde kodu.....	57
Şekil 3.7: Değişken komşuluk arama algoritması için paralelleştirme stratejisi temsili gösterimi.....	58
Şekil 3.8: Çalkalama ve yerel arama aşamaları için havuz büyüklüklerine ait etkileşim grafiği.....	61
Şekil 3.9: Çalkalama ve yerel arama aşamaları için havuz büyüklüklerine ait ana faktör etkileri (main effect) grafiği.....	61
Şekil 3.10: Hang Seng veri seti için etkin sınır.....	65
Şekil 3.11: DAX 100 veri seti için etkin sınır.....	65
Şekil 3.12: FTSE 100 veri seti için etkin sınır.....	66
Şekil 3.13: S&P 100 veri seti için etkin sınır.....	66
Şekil 3.14: NIKKEI veri seti için etkin sınır.....	67

## TABLO LİSTESİ

### Sayfa

Tablo 2.1: Portföy optimizasyonu üzerine yayınlanan derleme makaleler .....	5
Tablo 2.2: Önerilen anahtar kelime yapısı .....	6
Tablo 2.3: Yayınların problem amaç fonksiyonu tipine göre sınıflandırılması	16
Tablo 2.4: Portföy optimizasyonu için kullanılan gerçek hayat kısıtları .....	17
Tablo 2.5: Kullanılan kısıt tipine göre yayınların sınıflandırılması .....	19
Tablo 2.6: OVPO yayınlarının kullanılan veri kaynağına göre sınıflandırılması	22
Tablo 2.7: OVPO literatüründeki yakınsama tabanlı performans göstergeleri.	27
Tablo 2.8: OVPO literatüründeki çeşitlilik tabanlı performans göstergeleri ....	27
Tablo 2.9: OVPO literatüründeki hibrit performans göstergeleri .....	27
Tablo 2.10: OVPO literatüründeki risk uyarlı performans göstergeleri .....	27
Tablo 2.11: OVPO için kesin çözüm tekniklerinin kullanıldığı yayınlar .....	29
Tablo 2.12: OVPO için uygulanan evrimsel tabanlı algoritmalar.....	31
Tablo 2.13: OVPO için evrimsel tabanlı algoritma kullanılan yayınlar .....	34
Tablo 2.14: OVPO problemine uygulanan sürü temelli algoritmalar .....	35
Tablo 2.15: OVPO için sürü temelli algoritma kullanılan yayınlar .....	39
Tablo 2.16: OVPO problemine uygulanan tek çözüm tabanlı algoritmalar.....	40
Tablo 2.17: OVPO için tek çözüm tabanlı algoritmaların uygulandığı yayınlar	41
Tablo 2.18 OVPO problemine uygulanan makine öğrenmesi algoritmaları.....	42
Tablo 2.19: OVPO için makine öğrenmesi algoritmasının kullanıldığı yayınlar	42
Tablo 2.20: OVPO için hibrit çözüm yöntemlerinin kullanıldığı yayınlar .....	44
Tablo 2.21: OVPO için kullanılan farklı kısıt işleme tekniklerini bazında yayınların sınıflandırılması .....	47
Tablo 3.1: Önerilen çözüm yaklaşımı için parametre seviyeleri.....	60
Tablo 3.2: FTSE 100 veri seti için temsili bir havuz büyüklüğü gösterimi .....	60
Tablo 3.3: Hesaplamalı sonuçların ortalama yüzde hata ölçütüne göre Deng, ve diğ. (2012) ile performans karşılaştırması ( $\lambda$ 'nın 50 farklı değeri için) .....	68
Tablo 3.4: Hesaplamalı sonuçların Chang, ve diğ. (2000), Deng, ve diğ. (2012), Lwin ve Qu (2013) ile performans karşılaştırması ( $\lambda$ 'nın 50 farklı değeri için).....	69
Tablo 3.5: Hesaplamalı sonuçların Cura (2009), Baykasoglu ve diğ. (2015) and Kalayci ve diğ. (2017) ile performans karşılaştırması ( $\lambda$ 'nın 51 farklı değeri için).....	70

## ÖNSÖZ

Yüksek lisans öğrenimim boyunca gösterdiği her türlü destekleri ve yardımları için, akademisyenlik yolunda değerli önerileri ile bana ışık tutan çok kıymetli hocam Doç. Dr. Can Berk Kalaycı'ya,

Hayatım boyunca maddi manevi desteklerini daima arkamda hissettiğim sevgili aileme,

Bu yoğun dönemde, yeterince zaman ayıramadığım sevgili eşim ve hayatımızı enerjisi ile renklendiren canım kızıma sabırları ve destekleri için teşekkür ve minnetlerimi sunarım.

# 1. GİRİŞ

Bireysel yatırımcılar, aracılar ve fon yöneticileri her yıl çeşitli sektörlere milyarlarca dolar yatırım yapmaktadırlar. Yapılacak yatırımlar dikkate alındığında, kar elde etmek ve piyasalardaki muhtemel gerilemeler nedeniyle oluşabilecek kayıpları minimize etmek istendiğinden, finansal yatırım için en iyi yatırım alternatiflerinin seçimi ön plana çıkmaktadır. Bu kapsamda kullanılan en yaygın yatırım stratejilerinden birisi de riski dağıtmak için farklı menkul kıymetlerden oluşan bir portföy oluşturmaktır. Portföy optimizasyonu (PO) problemi ise birbiri ile çelişen iki amaç olan risk minimizasyonunu ve getiri maksimizasyonu dikkate alınarak yatırım için en uygun portföyün oluşturulması olarak tanımlanabilmektedir. Geleneksel portföy analizi, bireysel menkul kıymetlerin getiri ve risk koşullarının değerlendirilmesini gerektirmekte ve öznel niteliği nedeniyle başarı sağlayamamaktadır. 1950'li yılların başlarında Markowitz (1952) riskin yatırım üzerine etkisini analiz ederek Ortalama-Varyans (OV) modelini geliştirmiş ve portföy teorisi alanında devrimsel nitelikte bir çalışmaya imza atmıştır (Markowitz 1952). Yaptığı bu çalışma ile modern portföy teorisi olarak da nitelendirilen yeni bir çağ başlatmıştır. Kovaryansın yatırım alternatiflerinin değerlendirilmesinde bir risk ölçütü olarak kullanılması kantitatif finans yaklaşımını tetikleyen en önemli unsurlardan birisi olmuş, bu dönüm noktasından sonra OV modeli birçok araştırmacı ve yatırımcı tarafından portföy oluşturmak ve performansını ölçebilmek için kullanılan standart bir karar verme yaklaşımı haline gelmiştir (Rubinstein 2002). Nitekim Markowitz'in bu öncü çalışması 1991 yılında ona Nobel ekonomi ödülünü kazandırmıştır. PO konusu bilişim dünyasındaki gelişmeler ve karmaşık problemlerin hesaplayabilen tekniklerin geliştirilmesi ile birlikte sadece finans alanından değil, bilgisayar bilimleri ve matematik gibi alanlardan da birçok araştırmacının dikkatini çekmiştir. OVPO problemi ile ilgili akademik dergilerde yayınlanan yayın sayısı da bunu doğrular niteliktedir.

Markowitz'in önermiş olduğu OV yaklaşımı risk ve getiri arasındaki ödünleşmeyi basit ve etkin bir şekilde açıklasa da birtakım eksiklikleri nedeniyle gerçek hayat problemleri karşısında yetersiz kalmaktadır. Bu doğrultuda birçok

arařtırmacı yeni kısıtlar, ama fonksiyonu yapıları ve özüm yaklařımları geliřtirerek orijinal modele eklemeler yapmıřlar ve yeni modeller önermiřlerdir (Konno ve Yamazaki 1991; Rockafellar ve Uryasev 2000; Young 1998). Orijinal OVPO modelinin eksikliklerinin bir tanesi de modelde gerek hayat kısıtlarının bulunmamasıdır. Bu sebeple önerilen model gerek hayat problemleri iin kabul edilebilir sonuçlar üretmekte eksik kalmaktadır. Örneėin, belirli bir portföyde ok fazla sayıda varlık tutmak beraberinde ek maliyetler getireceėinden gerek hayat senaryoları iin ok da mantıklı olmayacaktır. Bu sebeple, portföye dahil edilecek varlık sayısının sınırlandırılması iin eleman sayısı kısıtının (cardinality constraint) orijinal modele eklenmesi gerekmektedir (Chang ve diė. 2000).

Markowitz'in standart OV modeli konveks kuadratik programlama ile özülebilmekte iken, eleman sayısı kısıtının modele dahil edilmesi kuadratik modeli NP-Zor sınıfındaki karma tam sayılı kuadratik modele evirmektedir (Karp 1972). Bu durum ESK PO probleminin kesin özüm algoritmaları kullanılarak polinom zamanda özülebilmesini zorlařtırmaktadır. Dolayısıyla, oėu arařtırmacı, kısa sürede optimal özümüne yakın sonuç elde edilmesine olanak saėlayan meta-sezgisel yöntemlere bařvurmaktadır

Bu alıřma kapsamında, ESK PO probleminin özümü iin kuadratik programlama (KP) ve deėiřken komřuluk arama (DKA) algoritmasına dayanan iki ařamalı bir özüm yaklařımı geliřtirilmiřtir. PO literatürü incelendiėinde arařtırmacıların büyük bir oėunluėu portföye dahil edilecek varlıkların seimi ve seilen varlıkların yatırım oranlarının belirlenmesi iřlemlerini aynı anda deėerlendirirken, bu alıřmada bu iki süreç ayrı ayrı ele alınmıřtır. Böylelikle algoritma performansı basit ama etkili bir Őekilde arttırılmıřtır. Öncelikle, portföyde tutulacak varlıklar DKA algoritması kullanılarak seilmiřtir. Portföye dahil edilecek varlıklar belirlendikten sonra ek bir kısıt iřleme prosedürüne gerek kalmadan karma-tam sayılı kuadratik programlama modeli tekrar kuadratik programlama modeline dönüřtürülmüř olur. Daha sonra, önceden seilmiř varlıklar iin en uygun aėırlıklar KP ile belirlenir. Bahsedilen bu iki ařamalı algoritma yapısına ek olarak, bařlangı özümü oluřturma prosedürü, sıralı komřuluk deėiřimi, optimal havuz büyüklüėünün seimi mekanizmaları ve asenkron paralelleřtirme stratejisi, önerilen özüm yaklařımını literatürdeki diėer algoritmalarından ayırarak üstünlük kazandıran kritik

bileşenler olarak sayılabilir. Son olarak, önerilen çözüm yaklaşımı literatürde yaygın olarak kullanılan veri setleri üzerinde test edilerek diğer algoritmalarla performans karşılaştırması yapılmıştır. Deneysel sonuçlarda, bu çözüm yaklaşımının diğerleriyle rekabet edebilir bir performans sergilediği görülmüştür.

Bu çalışmanın ikinci bölümünde, OVPO problemi ile ilgili literatürde yapılan çalışmalar amaç fonksiyonu tipi, kısıtlar, kısıt işleme teknikleri, kullanılan çözüm yöntemleri, veri seti ve performans ölçütleri dikkate alınarak kapsamlı bir şekilde sınıflandırılmış ve analiz edilmiştir. Üçüncü bölümde ise eleman sayısı kısıtlı PO probleminin matematiksel modelinin yanı sıra, geliştirilen iki aşamalı çözüm yönteminin detaylarına yer verilmiştir. Son olarak bölüm dördte sonuçlar ve gelecekteki muhtemel araştırma yönelimleri yer almaktadır.

## 2. LİTERATÜR TARAMASI

Markowitz'in devrim niteliğindeki çalışmasının ardından, PO problemi birçok araştırmacı tarafından büyük ilgi görmüştür. Son on yılda, araştırmacılar PO'daki mevcut eğilimleri ve gelecekteki araştırma yönelimlerini analiz etmişlerdir. Bu kapsamda yayınlanan derleme yayınlar Tablo 2.1'de sunulmuştur. Kolm ve diğ. (2014), modele eklenen gerçek hayat kısıtlarının problemin yapısı ile ilgili getirmiş olduğu zorlukları dikkate alarak portföy optimizasyonu problemini pratik perspektiften ele almışlardır. Metaxiotis ve Liagkouras (2012) ise, çok amaçlı evrimsel algoritmaların (ÇAEA) tasarımını ve uygulamalarını incelemişlerdir. Aouni ve diğ. (2018), portföy optimizasyonu problemine uygulanan kesin çözüm yöntemlerini çeşitli risk ölçütleri ile birlikte ele almışlardır. Ertenlice ve Kalayci (2018), portföy optimizasyonunun çözümünde kullanılan sürü zekâsı algoritmalarını gözden geçirmişlerdir.

Yakın zamanda ise Kalaycı ve diğ. (2019) portföy optimizasyonu üzerine yapılan uygulamaları model tipi, kullanılan kısıtlar, kısıt işleme teknikleri, geliştirilen çözüm yaklaşımları ve kullanılan performans göstergelerini temel alarak geniş bir sınıflandırma altında incelemişler ve portföy optimizasyonu problemine ait mevcut trendleri ve muhtemel gelecek çalışma konularını tartışmışlardır.



**Tablo 2.1:** Portföy optimizasyonu üzerine yayınlanan derleme makaleler.

Yıl	Yayın	Perspektif	Odaklanılan konu
2010	Azmi ve Tamiz (2010)	Hedef programlama	Lexicographic/ağırlıklı/minmaks/bulanık hedef programlama modelleri
2012	Metaxiotis ve Liagkouras (2012)	Çok-amaçlı optimizasyon	Çok-amaçlı Evrimsel Algoritmalar
2013	Ponsich ve diğ. (2013)	Çok-amaçlı optimizasyon	Çok-amaçlı Evrimsel Algoritmalar
2014	Mansini ve diğ. (2014)	Doğrusal programlama	Doğrusal programlama ile çözülebilir PO modelleri
2014	Kolm ve diğ. (2014)	Uygulama zorlukları	OV PO'da uygulamalarındaki gelişmeler
2014	Aouni ve diğ. (2014)	Hedef Programlama	Lexicographic/ağırlıklı/minmaks/bulanık hedef programlama modelleri
2016	Doering ve diğ. (2016)	Hesaplamalı Analiz	Meta-sezgiseller
2017	Masmoudi ve Abdelaziz (2018)	Programlama modelleri	Deterministik ve stokastik çok-amaçlı programlama modelleri
2017	Zhang ve diğ. (2018b)	Belirsizlik	Dinamik/gürbüz/uygulamalı faktörlerle birlikte bulanık portföy optimizasyonu
2018	Aouni ve diğ. (2018)	Çok kriterli karar destek metotları	Çoklu kriterleri temel alan portföy seçimi için kesin çözüm yöntemleri
2019	Kalaycı ve diğ. (2019)	Hesaplamalı Analiz	OV PO'nun deterministik modelleri için kesin ve yakınsama temelli yaklaşımlar

## 2.1 Araştırma Metodolojisi

Bu çalışmaya ait içerik analizi ve literatür araştırmasının nasıl gerçekleştirildiği şu dört başlık altında özetlenmiştir: materyal toplama, betimsel analiz, kategori seçimi ve materyal değerlendirme (Govindan ve diğ. 2015; Özceylan ve diğ. 2018).

### 2.1.1 Materyal Toplama

Bu bölümde, literatür taraması için kullanılan materyaller detaylı olarak anlatılmıştır.

Bu çalışma kapsamında incelenen yayınlar, 1998'den 2019 yılına kadar İngilizce dilinde yayınlanmış olan makale, kitap bölümü ve konferans bildirimlerini kapsamaktadır. Araştırma için kullanılan arama terimleri, daha önceki yazarların tecrübeleri ile birlikte farklı yayınlarda kullanılan anahtar kelimelerden de yararlanılarak birçok deneme yanılma girişimi sonrasında belirlenmiştir. Bu kapsamda, OVPO problemi ile ilgili yayınları kapsayacak şekilde geniş bir arama

imkânı sunması amaçlanan anahtar kelime yapısı Tablo 2’de gösterilmiştir. Söz konusu anahtar kelime yapısının birinci seviyesi temel arama içeriğini tanımlarken, ikinci seviye, yazarların kullanması muhtemel terim alternatiflerini çeşitlendirmektedir. Üçüncü seviye araştırmayı ortalama-varyans yaklaşımı ile sınırlandırırken son olarak “bulanık” ve “gürbüz” anahtar kelimelerinin aramadan hariç tutulmasıyla bulanık mantık ve gürbüz çözüm yaklaşımları bu araştırma kapsamından çıkarılmıştır.

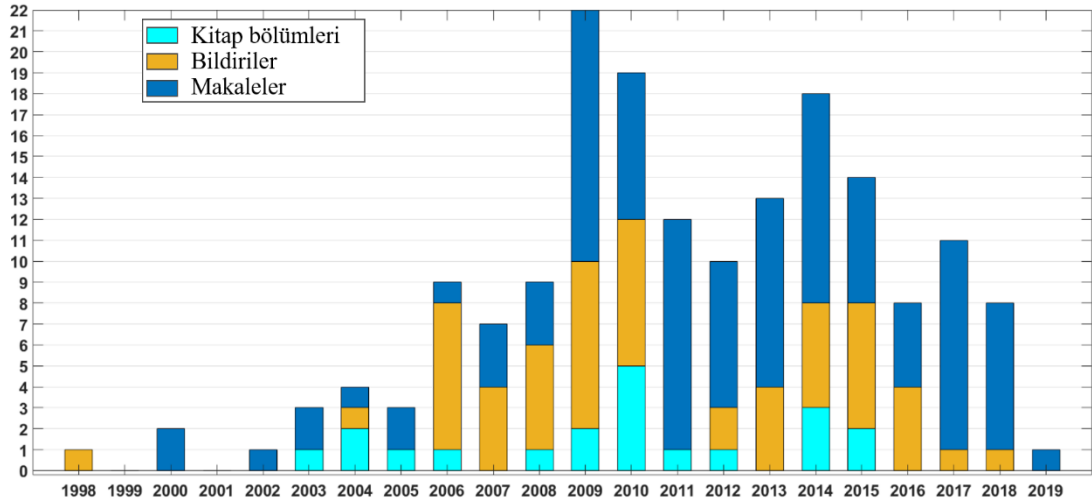
**Tablo 2.2:** Önerilen anahtar kelime yapısı.

Seviye	Arama terimleri
1	Portfolio AND
2	Selection            OR            Management            OR            Optimization AND
3	Mean-variance AND NOT
4	Fuzzy            OR            Robust

Başlangıç olarak, Web of Science ve Scopus veri tabanlarından çok daha fazla yayın barındıran Google Scholar veri tabanında “başlık, özet, anahtar kelimeler” arama yapısını kullanılarak yaklaşık olarak 34.100 yayına ulaşılmıştır. Yöneylem araştırması perspektifinden bakılarak OVPO problemine ait deterministik model ve uygulamaları ile ilgili yayınların elde edilmesi amacıyla, toplam 175 adet yayın teker teker seçilmiş, incelenmiş ve sınıflandırılarak daha detaylı analizlerin yapılabilmesi için bir Excel sayfasında derlenmiştir.

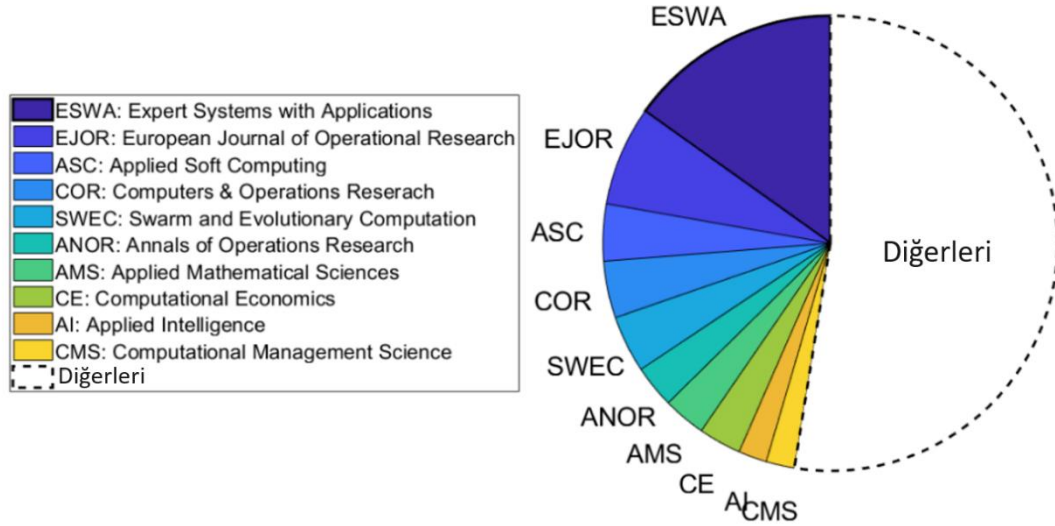
### 2.1.2 Betimsel Analiz

1998-2019 yılları arasındaki yayınların yıl bazında dağılımları Şekil 2.1’de gösterilmektedir. 20 kitap bölümü, 56 konferans bildirisi ve 99 makaleyi kapsayan toplamda 175 adet yayın incelenmiştir. Genel olarak değerlendirildiğinde, 2009’dan sonra bu alandaki yayın sayısındaki ciddi artış göze çarpmaktadır.

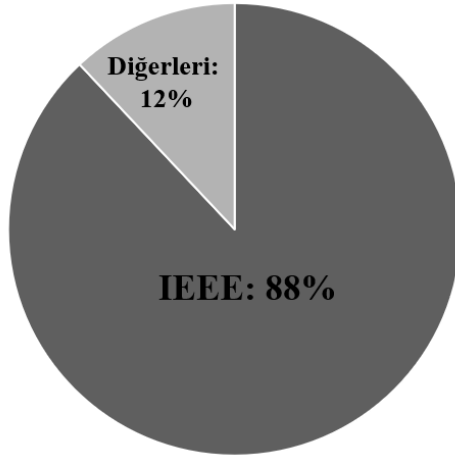


Şekil 2.1: Yayınların yıllara göre dağılımları.

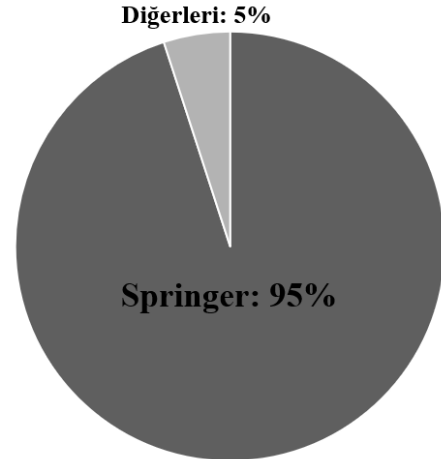
Şekil 2.2, 2.3 ve 2.4 sırasıyla makalelerin, konferans bildirilerinin ve kitap bölümlerinin yayın yerlerine göre dağılımlarını göstermektedir. Şekil 2.2’de görüldüğü üzere makalelerin %15’lik ile çoğunlukta olan kısmı Expert Systems with Applications dergisinde yayınlanmıştır. Şekil 2.3 ve Şekil 2.4’e göre ise konferans bildirilerinin %88’i IEEE’de yayınlanırken kitap bölümlerinin %95’inin ise Springer tarafından yayınlandığı görülmektedir. 1998-2018 tarihleri arasında toplam 46 adet bildiri ve 36 adet kitap bölümü dikkate alınmıştır.



Şekil 2.2: Yayınların, yayınlandığı dergilere göre dağılımı.



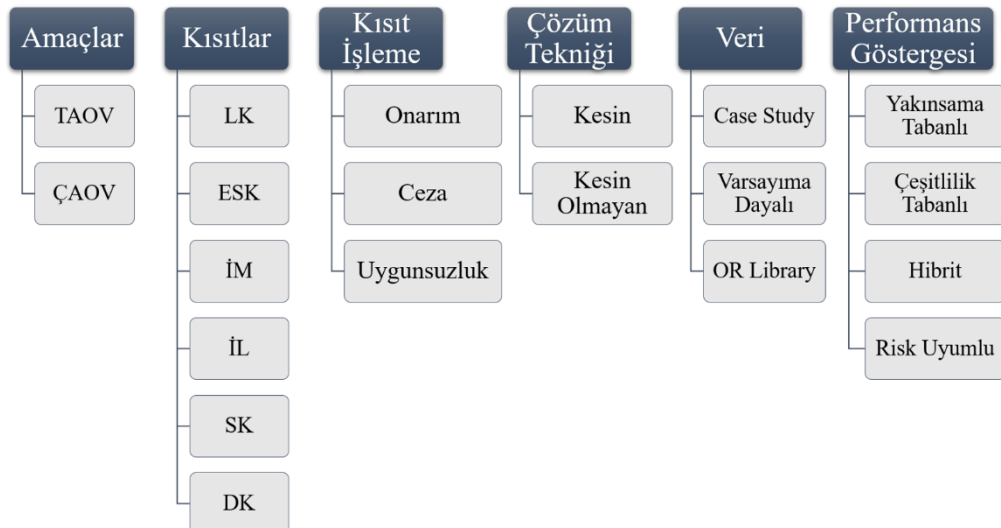
Şekil 2.3: Bildirilerin yayın yerine göre dağılımları.



Şekil 2.4: Kitap bölümlerin yayın yerine göre dağılımı.

### 2.1.3 Sınıflandırma

Literatür incelemesinde uygulanan sınıflandırma işlemi OVPO problemi model ve uygulamalarının farklı özellikleri temel alınarak tasarlanmıştır. Bu doğrultuda geliştirilen sınıflandırma şu kriterler çerçevesinde yapılmıştır: modeller, kullanılan kısıtlar, kısıt işleme teknikleri, çözüm teknikleri, performans ölçütleri ve kullanılan veri seti. İlgili sınıflandırma yapısı Şekil 2.5'te gösterilmiştir.



Tek-amaçlı OV (TAOV); Çok-amaçlı OV (ÇAOV); Limit Kısıtı (LK); Eleman Sayısı Kısıtı (ESK); İşlem Maliyeti (İM); İşlem Lotu Kısıtı (İL); Sektör Kapitalizasyonu Kısıtı (SK); Devir Kısıtı (DK); OVPO problemi ile ilgili çeşitli kısıt işleme metodolojileri (Kısıt İşleme); Önerilen algoritmaların etkinliğinin test edilmesi için kullanılan çeşitli performans (Performans Göstergeleri);

Şekil 2.5: OVPO model ve uygulamalarının sınıflandırılması.

## 2.1.4 Materyal Değerlendirme

Doğrulama sırasında araştırmanın güvenilirliğini artırmak amacıyla, muhtemel hataları değerlendirmek analiz etmek ve en aza indirmek için Microsoft Excel ve Endnote referans yöneticisi yazılımlarından yararlanılmıştır. Başlangıç aşamasında bulunamayan yayınları çalışmaya dahil ederek ve inceleyerek çalışmayı zenginleştirmek adına Web of Science ve Scopus gibi diğer veri tabanlarından da faydalanılmıştır.

## 2.2 OVPO Modeli

### 2.2.1 Tek-amaçlı OVPO modeli

İstenen bir getiri seviyesi için risk (varyans) minimizasyonunu amaçlayan orijinal OV modeli aşağıda sunulmaktadır.

Parametreler:

$N$  Toplam varlık sayısı

$\mu_i$   $i$ . varlığın beklenen getirisi

$\sigma_{ij}$   $i$ . ve  $j$ . varlıkları arasındaki kovaryans değeri

$R^*$  İstenen getiri seviyesi

Karar değişkenleri:

$w_i$   $i$ . varlığın yatırım oranı

$$\min \sum_{i=1}^N \sum_{j=1}^N w_i w_j \sigma_{ij} \quad (2.1)$$

*Kısıtlar:*

$$\sum_{i=1}^N w_i \mu_i = R^* \quad (2.2)$$

$$\sum_{i=1}^N w_i = 1 \quad (2.3)$$

$$0 \leq w_i \leq 1, \quad i = 1, \dots, N \quad (2.4)$$

Denklem (2.1) de verilen amaç fonksiyonu risk minimizasyonunu hedeflemektedir, Denklem (2.2) getirinin istenen seviyede olmasını garantilemektedir. Toplam bütçenin tamamının yatırım alternatiflerine dağıtılması Denklem (2.3) ile sağlanırken Denklem (2.4) bir varlığın ağırlığının 0 ile 1 arasında olmasını sağlamaktadır. Riskin minimizasyonu dikkate alınarak verilmiş yukarıdaki tek amaçlı OV modeli aynı zamanda belirli bir risk seviyesi altında getirinin maksimizasyonunu dikkate alacak şekilde tekrar düzenlenebilmektedir. Bahsedilen bu modeller doğrultusunda, belirli bir getiri seviyesinde minimum risk ya da kabul edilebilir bir risk seviyesi için maksimum getiri dikkate alınarak elde edilen portföye etkin portföy (efficient portfolio) denilmektedir. Fakat problem bu doğrultuda modellendiğinde etkin portföyün belirlenebilmesi için yatırımcının kabul edebileceği maksimum risk seviyesini ya da elde etmek istediği minimum getiri seviyesini belirlemesi gerekmektedir. Bu durum gerçek hayat senaryoları için çoğu zaman mümkün olmamaktadır. Bu doğrultuda, birçok farklı varlık kombinasyonu arasından etkin portföyün bulunabilmesi için araştırmacıların ve yatırımcıların tek bir amaç fonksiyonu yerine tüm amaç fonksiyonlarını birlikte değerlendirmesi gerekmektedir. Bu sebeple araştırmacılar zamanla tek amaçlı modeller yerine çok amaçlı modellere ihtiyaç duymuştur.

## 2.2.2 Çok-amaçlı OVPO modeli

Zitzler (1999) e göre çok amaçlı matematiksel model şu şekilde ifade edilmektedir:

$$\min \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}) \dots, f_p(\mathbf{x})) \quad (2.5)$$

$$\begin{aligned} \text{kısıtlar: } \mathbf{e}(\mathbf{x}) &= (e_1(\mathbf{x}), e_2(\mathbf{x}) \dots, e_m(\mathbf{x})) \leq \mathbf{0} \\ \mathbf{x} &= (x_1, x_2, \dots, x_n) \in \mathbf{X} \end{aligned} \quad (2.6)$$

Kısıt  $\mathbf{e}(\mathbf{x}) \leq \mathbf{0}$  olurlu çözüm kümesini,  $\mathbf{x} = (x_1, \dots, x_n)$  karar değişkenleri vektörünü (parametreler) ve  $\mathbf{X}$  ise karar uzayını temsil etmektedir.  $p > 1$ 'e eşit olduğunda model tek amaçlı optimizasyon modelini ifade etmekte iken  $p = 1$ 'den büyük değerleri için model çok amaçlı optimizasyon modeline dönüşmektedir.

Tek amaçlı optimizasyon problemlerinin aksine, çok amaçlı optimizasyon problemlerinde çözüm bir tanımdan ziyade daha çok kavramlar bütünü ile açıklanabilmektedir (Marler ve Arora 2004). Bu sebeple, çok amaçlı optimizasyon probleminin yapısının anlaşılabilmesi için bu temel kavramların üzerinde durmak daha yerinde olacaktır. Bu doğrultuda, Zitzler (1999) tarafından tanımlanan yapıya göre, olurlu küme (feasible set), pareto üstünlüğü (pareto dominance), pareto optimalliği (pareto optimality) ve baskın olmayan küme (nondominated set) gibi kavramlar aşağıda açıklanmaktadır:

### **Olurlu Küme**

$\mathbf{X}_f$  olurlu kümesi  $\mathbf{x}$  karar değişkeni vektörünün  $\mathbf{e}(\mathbf{x})$  kısıtını sağlayan elemanlarının oluşturduğu küme olarak tanımlanmaktadır.

$$\mathbf{X}_f = \{\mathbf{x} \in \mathbf{X} \mid \mathbf{e}_i(\mathbf{x}) \leq 0, i = 1, \dots, m\} \quad (2.7)$$

$\mathbf{X}_f$  olurlu kümesinin görüntüsü, diğer bir deyişle amaç uzayındaki olurlu bölge,  $f(\mathbf{X}_f) = \cup_{\mathbf{x} \in \mathbf{X}_f} \{f(\mathbf{x})\}$  ile ifade edilmektedir.

### **Pareto Üstünlüğü**

Pareto üstünlüğü kavramı aşağıdaki şekilde açıklanmaktadır (Deb 2001):

Herhangi iki karar değişkeni vektörü  $\mathbf{u}$  ve  $\mathbf{v}$  için,

$\mathbf{u} < \mathbf{v}$  ( $\mathbf{u}, \mathbf{v}'$  den üstündür) sadece ve sadece  $\mathbf{f}(\mathbf{u}) < \mathbf{f}(\mathbf{v})$  ise  
 $\mathbf{u} \preceq \mathbf{v}$  ( $\mathbf{u}, \mathbf{v}'$  den zayıf üstündür) sadece ve sadece  $\mathbf{f}(\mathbf{u}) \preceq \mathbf{f}(\mathbf{v})$  ise  
 $\mathbf{u} \sim \mathbf{v}$  ( $\mathbf{u}, \mathbf{v}'$  den farksızdır) sadece ve sadece  $\mathbf{f}(\mathbf{u}) \not\preceq \mathbf{f}(\mathbf{v})$  ve  $\mathbf{f}(\mathbf{v}) \not\preceq \mathbf{f}(\mathbf{u})$  ise

### Pareto Optimallığı

Eğer herhangi bir  $\mathbf{u} (\mathbf{u} \in \mathbf{A}: \mathbf{u} < \mathbf{x})$  vektörü mevcut değilse  $\mathbf{x} (\mathbf{x} \in \mathbf{X}_f)$  karar vektörü  $\mathbf{A} \subseteq \mathbf{X}_f$  kümesi ile ilgili olarak üstünlük kurulamayan (nondominated) olarak tanımlanmaktadır. Eğer  $\mathbf{x}$  vektörüne  $\mathbf{X}_f$  olurlu kümesinde üstünlük kurulamamışsa,  $\mathbf{x}$  vektörüne pareto optimal adı verilir. Bütün pareto optimal çözümleri içeren kümeye ise Pareto-optimal çözüm kümesi adı verilir ve buna karşılık gelen amaç vektörleri Pareto-optimal sınırı oluşturmaktadır (Zitzler 1999).

### Üstünlük Kurulamamış Küme

$\mathbf{A}$  ile ifade edilen bir çözüm kümesinde, çözüm kümesinin geriye kalan elemanları tarafından üstünlük kurulamayan elemanları üstünlük kurulamayan çözüm kümesi olarak adlandırılır.

OVPO için optimal çözümlerin tek amaçlı problem çözme yapıları kullanarak elde edilmesi gerçek hayat koşulları nedeniyle zordur. Bu sebeple, çok amaçlı optimizasyon yöntemlerinin uygulanmasına ihtiyaç duyulmaktadır. Tek amaçlı optimizasyon modeli ile çözüm elde edildiğinde global ve tek bir optimal çözüm bulunabilirken çok amaçlı optimizasyon modelleri ile pareto optimal çözüm olarak da adlandırılan bir dizi optimal çözüm elde edilir. Ayrıca, tek amaçlı modeller risk minimizasyonunu ve getiri maksimizasyonunu temel alarak iki farklı şekilde tanımlanabilirken, çok amaçlı optimizasyon modellerinde birbiriyle çelişen amaçlar aynı anda optimize edilmektedir.

#### 2.2.2.1 Çok-amaçlı OVPO modeli için yaklaşımlar

Çok amaçlı modellerde, tüm amaç fonksiyonlarını aynı anda optimize etmek mümkün olmayabilmektedir. Bu nedenle, ya ağırlıklı toplam metodu (weighted sum method) gibi yöntemler kullanılarak amaçlar bir parametre aracılığıyla



önceliklendirilir ya da üstünlük temelli yaklaşımlar aracılığıyla her iki amacı aynı anda değerlendirerek birbirine üstün gelen çözümler tespit edilir.

### Ağırlıklı Toplam Yöntemi (Weighted Sum Approach)

Ağırlıklı toplam yönteminde, birden fazla amaç (objective) birbiri üzerinde önceliklendirme yapılacak şekilde her biri üzerine birleşik bir ağırlık atanarak tek bir amaç fonksiyonu olarak birleştirilir. Böylece tek amaçlı modellerdeki kısıt olarak ifade edilen amaç fonksiyonu gevşetilmiş olur. Basit yapısı ve uygulama kolaylığı nedeniyle, çok amaçlı optimizasyon problemleri için en yaygın kullanılan klasik yaklaşım olmasına yanı sıra (Deb 2005), bu yaklaşım OVPO için de en popüler metotlar arasındadır. Fakat, ağırlıklı toplam yöntemi basit yapısına karşın, konveks olmayan pareto optimal sınıra sahip çok amaçlı optimizasyon problemleri için tüm pareto optimal çözümleri üretememektedir (Zitzler 1999).

$$\min \sum_{i=1}^p \lambda_i f_i(x) \quad (2.8)$$

$$\text{kısıtlar: } x \in X_f \quad (2.9)$$

Burada  $\lambda_i$ ,  $f_i$  amaç fonksiyonunun ağırlığını ifade etmektedir.

Bu doğrultuda, çok amaçlı OV (ÇAOV) modeli ağırlıklı toplam yaklaşımı kullanılarak aşağıdaki şekilde yeniden ifade edilmiştir (Chang ve diğ. 2000):

$$\min \lambda \left[ \sum_{i=1}^N \sum_{j=1}^N w_i w_j \sigma_{ij} \right] - (1 - \lambda) \left[ \sum_{i=1}^N w_i \mu_i \right] \quad (2.10)$$

*Kısıtlar:*

$$\sum_{i=1}^N w_i = 1 \quad (2.11)$$

$$0 \leq w_i \leq 1 \quad i = 1, \dots, N \quad (2.12)$$

Denklem (2.10) da görüldüğü gibi, iki çelişen amaç (risk minimizasyonu, getiri maksimizasyonu) bir  $\lambda$  parametresi aracılığıyla ağırlıklandırılmıştır. Bu  $\lambda$  ağırlık parametresi 0 ile 1 arasında değişen değerlerden oluşmaktadır.  $\lambda = 0$  olarak belirlendiğinde amaç fonksiyonu maksimum getiriye araştırırken,  $\lambda = 1$  olarak belirlendiğinde ise amaç fonksiyonu risk minimizasyonuna dönüşmektedir. Özetle

$0 < \lambda < 1$  aralığındaki değerleriyle  $\lambda$  risk ve getiri arasındaki ödünleşmeyi ifade etmektedir.

### Pareto-temelli yaklaşımlar

Pareto temelli yaklaşımlar, tek bir optimizasyon işleminde, geniş arama uzaylarının ve çoklu alternatifler arası ödünleşmenin üstesinden gelebilmektedir. (Zitzler 1999). Fakat, ağırlıklı toplam yönteminin aksine çok amaçlı yapının tek bir amaç fonksiyonu şeklinde ifade edilmesi için etkin sınırın kalitesini belirleyecek belirli bir kriter bulunmamaktadır; kalite ölçütlerinin belirlenmesi de nispeten zordur.

Pareto temelli yaklaşımlarda, tipik olarak, pareto optimalite konseptine dayanan bir çözüm sıralama stratejisi kullanılmaktadır (Horn ve diğ. 1994). Çok-amaçlı algoritmaların büyük bir çoğunluğu pareto sıralamasını temel almaktadır. Ancak, baskınlık derinliği (dominance depth) (Deb ve diğ. 2002) ve baskınlık sayımı (dominance count) (Zitzler ve diğ. 2001) gibi varyasyonları da bulunmaktadır.

Lwin ve diğ. (2014)'a göre ÇAOV modeli aşağıdaki şekilde ifade edilebilmektedir:

$$\min \left[ \sum_{i=1}^N \sum_{j=1}^N w_i w_j \sigma_{ij} \right] \text{ and } \max \left[ \sum_{i=1}^N w_i \mu_i \right] \quad (2.13)$$

*Kısırlar:*

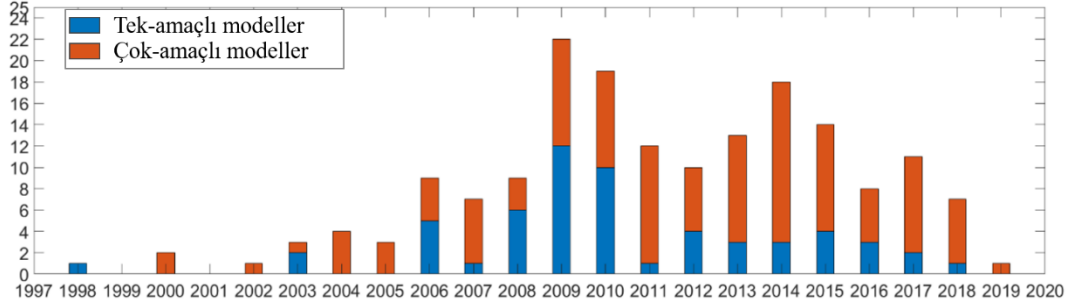
$$\sum_{i=1}^N w_i = 1 \quad (2.14)$$

$$0 \leq w_i \leq 1 \quad i = 1, \dots, N \quad (2.15)$$

Denklem (2.13) de görüldüğü gibi iki çelişen amaç (risk minimizasyonu, getiri maksimizasyonu), pareto optimal sınırın belirlenmesi için bağımsız olarak değerlendirilmektedir.

### 2.2.3 Tek-amaçlı ve Çok-amaçlı OV modellerinin karşılaştırılması

Tek amaçlı modellerde yatırımcıların arzu ettikleri risk veya getiri seviyelerini biliyor olduğu varsayılmaktadır. Ancak gerçek hayatta, bu varsayım her zaman geçerli olmamaktadır. Bu yüzden, çok amaçlı modellerin gerçek hayat problemleri için daha uygun olduğu bilinmektedir. Literatür incelendiğinde de çok amaçlı OVPO modelleri, tek amaçlı modellere göre daha fazla kullanıldığı açıkça görülmektedir (Şekil 2.6).



Şekil 2.6: OVPO için tek ve çok-amaçlı modellerin yıllara göre dağılımı.

OVPO literatüründeki yayınlar amaç fonksiyonu türlerine (çok amaçlı modeller ve tek amaçlı modeller) ve model yaklaşımlarına (ağırlıklı toplam yaklaşımı ve Pareto-temelli yaklaşımlar) göre Tablo 2.3'te sınıflandırılmıştır.

**Tablo 2.3:** Yayınların problem amaç fonksiyonu tipine göre sınıflandırılması.

Amaç fonksiyonu tipi	Yaklaşım metodu	Yayınlar
Çok Amaçlı	Ağırlıklı Toplam Metodu	(Ackora-Prah ve diğ. 2014 <sup>a</sup> ; Ackora-Prah ve diğ. 2014 <sup>b</sup> ; Anagnostopoulos ve diğ. 2010; Bacanin & Tuba, 2014, 2015; Bacanin ve diğ. 2014; Baykasoglu ve diğ. 2015; Buseti, 2006; Cesarone ve diğ. 2013; Chang ve diğ. 2000; Chang ve diğ. 2009; Chen ve diğ. 2013; Chen ve diğ. 2012; Chen ve diğ. 2008; ChiangLin, 2006; Cura, 2009; Deng & Lin, 2010 <sup>a</sup> , 2010 <sup>b</sup> ; Deng ve diğ. 2012; Farzi ve diğ. 2013; Fernandez & Gomez, 2007; Gaspero ve diğ. 2011; Golmakani & Alishah, 2008; Hadi ve diğ. 2016; Hao & Liu, 2009; Kalayci ve diğ. 2017; Kamili & Riffi, 2015, 2016; Kao & Cheng, 2013; Koshino ve diğ. 2007; Li ve diğ. 2010; Lin ve diğ. 2005; Lu & Wang, 2013; Lwin & Qu, 2013; Maringer & Kellerer, 2003; Moral-Escudero ve diğ. 2006; Mozafari ve diğ. 2011; Ni ve diğ. 2017; Niu ve diğ. 2010; Niu ve diğ. 2009; Pai & Michel, 2009; Peng ve diğ. 2011; Pouya ve diğ. 2016; Qu ve diğ. 2017; Rong ve diğ. 2009; Sabar & Song, 2014; Sadigh ve diğ. 2012; Schaerf, 2002; Suganya & Vijayalakshmi Pai 2009; Sun ve diğ. 2011; Suthiwong & Sodanil, 2016; Tan ve diğ. 2013; Tan ve diğ. 2014; Tuba & Bacanin, 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Tuba ve diğ. 2014; Wang ve diğ. 2015; Wang ve diğ. 2011, 2012; Woodside-Oriakhi ve diğ. 2011; Xia ve diğ. 2000; Xu ve diğ. 2010; Yaakob & Watada, 2010; Yin ve diğ. 2015 <sup>a</sup> ; Yin ve diğ. 2015 <sup>b</sup> ; Zhu ve diğ. 2010; Zhu ve diğ. 2011)
	Pareto Temelli Metot	(Anagnostopoulos & Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos & Mamanis, 2011 <sup>b</sup> ; Arkeman ve diğ. 2013; Bevilacqua ve diğ. 2011; Branke ve diğ. 2009; Chen ve diğ. 2017; Chen & Zhou, 2018; Chiam ve diğ. 2007; Chiam ve diğ. 2008; Dreżewski & Doroz, 2017; Duran ve diğ. 2009; Eftekharian ve diğ. 2017; Ehgott ve diğ. 2004; Fieldsend ve diğ. 2004; Garcia ve diğ. 2012; Jalota & Thakur, 2018; Kumar & Mishra, 2017; Liagkouras & Metaxiotis, 2014, 2017, 2018; Liang & Qu, 2013; Lwin ve diğ. 2014; Lwin ve diğ. 2013; Lwin ve diğ. 2017; Macedo ve diğ. 2017; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Mishra ve diğ. 2009; Ong ve diğ. 2005; Ruiz-Torrubiano & Suárez, 2007; Sen ve diğ. 2015; Skolpadungket ve diğ. 2007; Streichen & Tanaka-Yamawaki, 2006; Streichert ve diğ. 2004 <sup>a</sup> ; Streichert ve diğ. 2004 <sup>b</sup> ; Zhou & Li, 2014)
Tek Amaçlı		(Abbas & Haider, 2009; Aranha & Iba, 2009; Ban ve diğ. 2018; Bonami & Lejeune, 2009; Cao & Tao, 2010; Cesarone ve diğ. 2015; Chang & Chen, 2008; Chang & Hsu, 2007; Chen & Cai, 2008; Chen ve diğ. 2006; Chen & Zhang, 2010; Corazza ve diğ. 2012 <sup>a</sup> ; Corazza ve diğ. 2013; Coutino-Gomez ve diğ. 2003; Crama & Schyns, 2003; Cui ve diğ. 2014; Cui ve diğ. 2013; Dehghan Hardoroudi ve diğ. 2017; Fasheng & Wei, 2006; Freitas ve diğ. 2009; Gao & Chu, 2009; Golmakani & Fazel, 2011; He & Qu, 2016; Hoklie & Zuhail, 2010; Hong-mei ve diğ. 2010; Hu & Zhangy, 2010; Huang & Shen, 2010; Huang, 2012; Jiang ve diğ. 2014; Jiang ve diğ. 2008; Kumar & Bhattacharya, 2012; Lai ve diğ. 2006; Lean ve diğ. 2008; Li ve diğ. 2006; Lin & Liu, 2008; Loukeris ve diğ. 2009; Mayambala ve diğ. 2015; Reid & Malan, 2015; Ruiz-Torrubiano & Suarez, 2010, 2015; Sadjadi ve diğ. 2012; Shaikh & Abbas, 2009; Shaw ve diğ. 2008; Shoaf & Foster, 1998; Soleimani ve diğ. 2009; Strumberger ve diğ. 2017; Talebi ve diğ. 2010; Tang ve diğ. 2009; Thomaidis, 2010; Tian ve diğ. 2016; Tuba ve diğ. 2013; Wang ve diğ. 2009; Xu & Chen, 2006; Xu ve diğ. 2007; Yu ve diğ. 2009; Zaheer & Pant, 2016; Zhang ve diğ. 2010)

Tablo 2.3 açık bir şekilde literatürde çok amaçlı modellerin, yöneylem araştırması topluluğu tarafından tek hedefli modellere nazaran daha sıklıkla tercih edildiğini göstermektedir. Ayrıca ağırlıklı toplam yaklaşımının daha popüler bir yaklaşım olduğu görünse de pareto optimalliği temel alan yaklaşımların son yıllarda artan bir ilgi gördüğü de göze çarpmaktadır.

## 2.2.4 Gerçek hayat kısıtları

Orijinal OV modeli tüm avantajlarına rağmen, gerçek hayat problemlerinin çözümü için yetersiz kalmaktadır. Orijinal OV modelinde, varlık ağırlıkları toplamının bire eşit olmasını sağlayan böylece eldeki toplam bütçenin tamamının muhtemel tüm yatırım alternatiflerine dağıtılmasını garantileyen sadece bir kısıt dikkate alınmaktadır. Bu sebeple, OV modelinin gerçekçi PO problemlerine gerçekli sonuçlar üretebilmesi için ek kısıtlara ihtiyacı vardır. Literatürde, OV modeline eklenerek daha gerçekçi modeller geliştirmek için kullanılan kısıtlar Tablo 2.4’te özetlenmiştir.

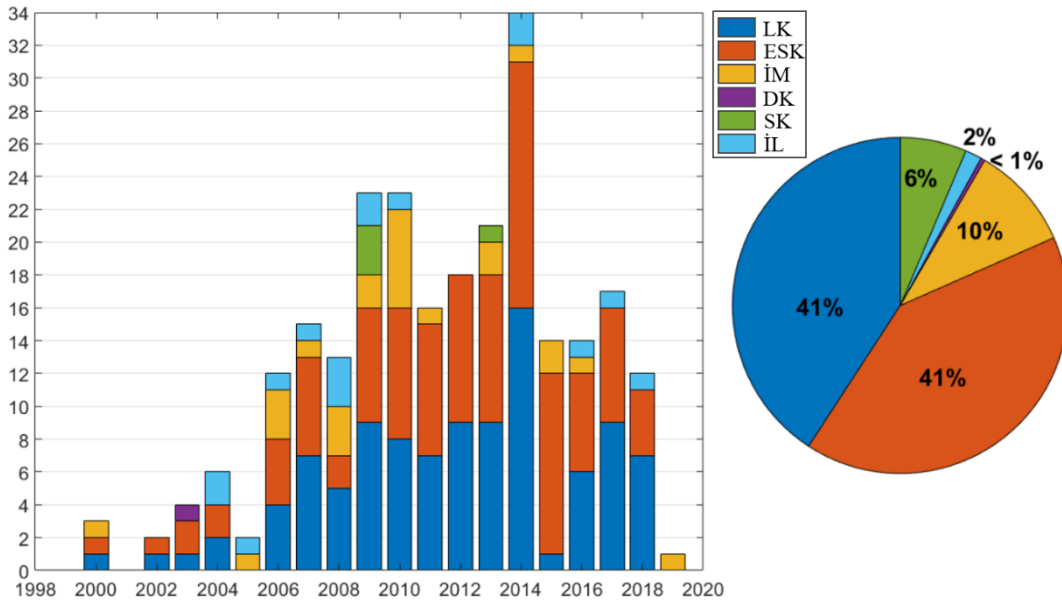
**Tablo 2.4:** Portföy optimizasyonu için kullanılan gerçek hayat kısıtları.

<i>Limit Kısıtları (LK-Boundary Constraints)</i>	Her bir varlığa ait ağırlık için alt ve üst sınır uygulamaktadır.
<i>Eleman Sayısı Kısıtı (ESK-Cardinality Constraints)</i>	Portföye dahil edilecek toplam varlık sayısını kısıtlamaktadır
<i>İşlem Maliyeti (İM-Transaction Cost)</i>	Yatırımcılar herhangi bir varlık alımı ya da satımı gerçekleştirdiğinde “işlem maliyeti” adı verilen bir ücret ödemek zorundadır. Dolayısıyla toplam kar bu maliyetten etkilenmektedir.
<i>İşlem Lotları Kısıtı (İLK)</i>	Herhangi bir varlığa yatırım yapılacak miktarı minimum işlem lotunun katları olmasını sağlar.
<i>Sektör Kapitalizasyonu Kısıtı (SKK-Sector Capitalization)</i>	Sektörde daha fazla kapitalizasyona sahip varlığı portföy içerisinde de daha fazla paya sahip olmaya zorlar
<i>Devir Kısıtı (DK-Turnover constraint)</i>	Çok periyotlu PO modellerinde bir varlığa ait mevcut periyot ve bir sonraki periyot arasındaki devir oranını belirlemektedir

Sırasıyla, Tablo 2.5 ve Şekil 2.7, literatürdeki yayınların kullanılan kısıt türlerine göre sınıflandırılmasını ve OV modelinde kullanılan gerçek hayat kısıtlarının yıllara göre ve kısıt tipine göre dağılımını göstermektedir. Buna göre, limit kısıtı ve eleman sayısı kısıtının diğerlerine göre daha fazla yayında kullanılmış olduğu açıkça görülmektedir. Her ne kadar işlem maliyetleri kısıtı çok-periyotlu portföy optimizasyonu problemi için daha anlamlı olsa da araştırmacılar bu kısıtı modelleme amacıyla tek-periyotlu portföy optimizasyonu modellerinde de yaygın olarak kullanmışlardır. Seçilen portföyün daha önceden belirlenmiş yatırım ufkuna göre

birkaç kez yeniden güncellendiği durumlarda, yalnızca çok dönemli portföy optimizasyon problemleri için geçerli olan devir kısıtının modele eklendiği görülmektedir.

Orijinal OV modeli, kuadratik amaç fonksiyonu ve doğrusal kısıtları içerirken, bazı araştırmacılar, OVPO modeline non-lineerlik ve non-konvekslik kazandıracak gerçek hayat kısıtlarını dahil etmişlerdir. Örneğin doğrusal veya kuadratik olmayan bir fonksiyonla ifade edilebilen İK modele eklendiğinde problemin karmaşıklığını artırabilmektedir. Fakat doğrusal bir fonksiyon olarak ifade edilen DK ise problemin karmaşıklığına herhangi bir etki yapmadığından problem hala KP ile çözülebilmektedir (Yoshimoto 1996). Bunun yanı sıra, LK ile birlikte modele dahil edilen ESK arama uzayının non-konveks hale dönüşmesine sebep olmaktadır (Xidonas ve Mavrotas 2014). Bu sebeple, LK ve ESK'yı içeren problemlerin çözümü için KP yetersiz kaldığından, çoğu araştırmacı gerçekçi senaryolar için kesin olmayan çözüm yaklaşımlarına başvurmaktadır.



Şekil 2.7: OV modelinde kullanılan gerçek hayat kısıtlarının dağılımı.

**Tablo 2.5:** Kullanılan kısıt tipine göre yayınların sınıflandırılması.

Kısıt Tipi	Yayınlar
LK- ESK	(Ackora-Prah ve diğ. 2014 <sup>a</sup> ; Ackora-Prah ve diğ. 2014 <sup>b</sup> ; Anagnostopoulos & Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos & Mamanis, 2011 <sup>b</sup> ; Anagnostopoulos ve diğ. 2010; Bacanin & Tuba, 2014, 2015; Bacanin ve diğ. 2014; Baykasoglu ve diğ. 2015; Buseti, 2006; Cesarone ve diğ. 2013, 2015; Chang ve diğ. 2000; Chang ve diğ. 2009; Chen ve diğ. 2013; Chen ve diğ. 2012; Chen ve diğ. 2017; Chiam ve diğ. 2007; Corazza ve diğ. 2012 <sup>a</sup> ; Corazza ve diğ. 2013; Cui ve diğ. 2014; Cui ve diğ. 2013; Cura, 2009; Deng & Lin, 2010 <sup>a</sup> , 2010 <sup>b</sup> ; Deng ve diğ. 2012; Eftekharian ve diğ. 2017; Farzi ve diğ. 2013; Fernandez & Gomez, 2007; Garcia ve diğ. 2012; Gaspero ve diğ. 2011; Golmakani & Alishah, 2008; Golmakani & Fazel, 2011; Hadi ve diğ. 2016; Jalota & Thakur, 2018; Jiang ve diğ. 2014; Jin ve diğ. 2015; Kalayci ve diğ. 2017; Kamili & Riffi, 2015, 2016; Kao & Cheng, 2013; Kırış & Ustun, 2012; Koshino ve diğ. 2007; Kumar & Mishra, 2017; Kumar & Bhattacharya, 2012; Liagkouras, 2018; Liagkouras & Metaxiotis, 2014, 2018; Lwin & Qu, 2013; Lwin ve diğ. 2013; Mansour ve diğ. 2007; Mayambala ve diğ. 2015; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Moral-Escudero ve diğ. 2006; Mozafari ve diğ. 2011; Ni ve diğ. 2017; Ruiz-Torrubiano & Suarez, 2010; Ruiz-Torrubiano & Suárez, 2007; Sabar & Song, 2014; Sadigh ve diğ. 2012; Sadjadi ve diğ. 2012; Schaerf, 2002; Streichen & Tanaka-Yamawaki, 2006; Suthiwong & Sodanil 2016; Tang ve diğ. 2009; Tian ve diğ. 2016; Tuba & Bacanin, 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Tuba ve diğ. 2014; Wang ve diğ. 2015; Wang ve diğ. 2011, 2012; Woodside-Oriakhi ve diğ. 2011; Xu ve diğ. 2010; Yaakob & Watada, 2010; Yin ve diğ. 2015 <sup>a</sup> ; Yin ve diğ. 2015 <sup>b</sup> )
İM	(Chen & Zhang, 2010; Huang & Shen, 2010; Li ve diğ. 2010; Lu & Wang, 2013; Paiva ve diğ. 2019; Peng ve diğ. 2011; Tan ve diğ. 2013; Xia ve diğ. 2000; Zhang ve diğ. 2010)
LK- ESK- İLK	(Chiam ve diğ. 2008; Liagkouras & Metaxiotis, 2017; Lwin ve diğ. 2014; Lwin ve diğ. 2017; Skolpadungket ve diğ. 2007; Streichert ve diğ. 2004 <sup>a</sup> ; Streichert ve diğ. 2004 <sup>b</sup> )
İM-İLK	(Chen ve diğ. 2008; ChiangLin, 2006; Lin & Liu, 2008; Lin ve diğ. 2005; Niu ve diğ. 2010)
LK-ESK-İM	(Brito & Vicente, 2014 <sup>a</sup> ; Gao & Chu, 2009; Hu & Zhangy, 2010; Ruiz-Torrubiano & Suarez, 2015)
ESK	(Dehghan Hardoroudi ve diğ. 2017; Maringer & Kellerer, 2003; Xu ve diğ. 2011)
LK	(Abbas & Haider, 2009; Chang & Hsu, 2007; Jiang ve diğ. 2008)
LK-İM	(Chen & Cai, 2008; Chen ve diğ. 2006; Xu ve diğ. 2007)
LK-ESK-SKK	(Pai & Michel, 2009; Tuba ve diğ. 2013)
LK-İLK	(Bonami & Lejeune, 2009; Zhou & Li, 2014)
LK-ESK-SKK-İLK	(Soleimani ve diğ. 2009)
LK-ESK-İM-İLK	(He & Qu, 2016)
LK-ESK-İM-SKK	(Suganya & Vijayalakshmi Pai, 2009)
LK-İM-DK	(Crama & Schyns, 2003)

LK: Limit Kısıtı; ESK: Eleman Sayısı Kısıtı; İM: İşlem Maliyeti; İLK: İşlem Lotları Kısıtı; SKK: Sektör Kapitalizasyonu Kısıtı; DK: Devir Kısıtı

### 2.3 Veri

Araştırmacılar OVPO problemi için geliştirdikleri çözüm yaklaşımlarının performansını literatürde yer alan farklı veri setleri üzerinde test etmektedirler. Bu

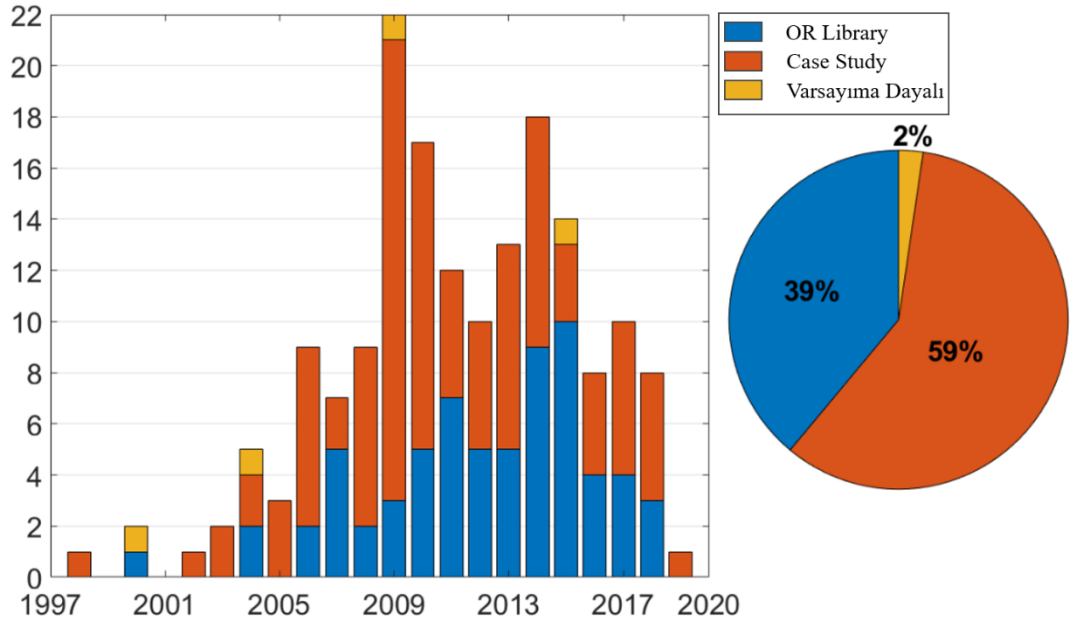
doğrultuda, OVPO literatüründe kullanılan veri setlerinin dağılımı Tablo 2.6'da gösterilmektedir. Yapılan araştırmaya göre, çoğu araştırmacı geliştirdiği algoritmanın test edilmesi için gerçek veri setlerini kullandığı görülmektedir. OR-Library (Beasley 1990) de yer alan portföy optimizasyonu test verileri OVPO literatüründe yaygın olarak kullanılan popüler karşılaştırma verileri olarak karşımıza çıkmaktadır. Bu veri setleri, sırasıyla 31,85, 89, 98 ve 225 farklı varlık bulunan Hang Seng, Dax 100, FTSE 100, S&P 100 ve Nikkei indekslerine ait Mart 1992-Eylül 1997 yılları arasındaki hisse senedi fiyatlarını içermektedir. Bu veri setlerine ek olarak Kalayci ve diğ. (2017) XU030 ve XU100 indeksleri için Mayıs 2013 ve Nisan 2016 tarihleri arasındaki günlük hisse senedi fiyatlarını kapsayan yeni veri setlerini literatüre kazandırmıştır. İlgili veri setleri <http://www.pau.edu.tr/portfolio/en> internet adresinde araştırmacıların kullanımına sunulmuştur. OR-Library'deki kullanıma açık veri setlerinin yanı sıra birçok araştırmacı geliştirilen çözüm yaklaşımlarını test etmek ve performans karşılaştırmaları için farklı indekslere ait veri setlerini kullanmışlardır (Case study). Bu durum literatürdeki veri çeşitliliğinin artırılması açısından çok kıymetli olsa da kullanılan çoğu veri setinin erişime açık olmaması sebebiyle bu setleri kullanan algoritmalarla birebir karşılaştırma yapılabilmesi çoğu zaman mümkün olmamaktadır. Literatürdeki birçok çalışmada Çin (Chen ve Zhou 2018; Chen ve Cai 2008; Chen ve diğ. 2008; Chen ve diğ. 2006; Jiang ve diğ. 2008; Lai ve diğ. 2006; Lean ve diğ. 2008; Liang ve Qu 2013; Lu ve Wang 2013; Niu ve diğ. 2010; Qu ve diğ. 2017; Rong ve diğ. 2009; Xu ve diğ. 2011; Yu ve diğ. 2009; Zhou ve Li 2014), Birleşik Krallık (Brito ve Vicente 2014<sup>a</sup>; Brito ve Vicente 2014<sup>b</sup>; Corazza ve diğ. 2012<sup>a</sup>; Corazza ve diğ. 2012<sup>b</sup>; Eftekharian ve diğ. 2017; Garcia ve diğ. 2012; Jalota ve Thakur 2018; Kumar ve Bhattacharya 2012; Liagkouras ve Metaxiotis 2018; Loukeris ve diğ. 2009; Shaw ve diğ. 2008), ABD (Aranha ve Iba 2009; Bonami ve Lejeune 2009; Corazza ve diğ. 2013; Eftekharian ve diğ. 2017; Ehrgott ve diğ. 2004; Fieldsend ve diğ. 2004; Lwin ve diğ. 2017; Sun ve diğ. 2011; Xu ve diğ. 2007), Japonya (Aranha ve Iba 2009; Eftekharian ve diğ. 2017; Jalota ve diğ. 2018; Kumar ve Bhattacharya 2012; Pai ve Michel 2009; Suganya ve Vijayalakshmi Pai 2009), Hong Kong (Eftekharian ve diğ. 2017; Jalota ve Thakur 2018; Jiang ve diğ. 2014; Li ve diğ. 2006), India (Suganya ve Vijayalakshmi Pai 2009; Zaheer ve Pant 2016), Venezuela (Duran ve diğ. 2009), Gana (Ackora-Prah ve diğ. 2014<sup>a</sup>; Ackora-Prah ve diğ. 2014<sup>b</sup>), Portekiz (Freitas ve diğ. 2009; Paiva ve diğ. 2019), Türkiye (Kırış ve Ustun 2012; Kocadağlı ve Keskin 2015) Endonezya (Hoklie ve Zuhail 2010), Pakistan (Abbas ve Haider 2009; Shaikh ve Abbas



2009), Tayland (Suthiwong ve Sodanil 2016), Tayvan (Chang ve Chen 2008; ChiangLin 2006; Lin ve Liu 2008), İnan (Pouya ve diğ. 2016; Talebi ve diğ. 2010), Tunus (Aouni ve diğ. 2005; Mansour ve diğ. 2007) ve Polonya (Dreżewski ve Doroz 2017) pazarlarına ait gnlk, aylık veya yıllık hisse senedi fiyatlarını ieren veri setleri kullanılmıřtır. Son olarak, ok kısıtlı sayıda alıřmada ise nerilen zm yaklařımlarının karřılařtırmaları varsayıma dayalı (hypotethical) veri setleri ve karřılařtırma fonksiyonları kullanılarak elde edilen test kmeleri zerinden gerekleřtirilmiřtir.

**Tablo 2.6:** OVPO yayınlarının kullanılan veri kaynağına göre sınıflandırılması.

Veri Setleri	Yayınlar
OR-Library	(Anagnostopoulos ve Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos ve Mamanis, 2011 <sup>b</sup> ; Bacanin ve Tuba, 2014, 2015; Baykasoglu ve diğ. 2015; Cesarone ve diğ. 2013, 2015; Chang ve diğ. 2000; Chang ve diğ. 2009; Chen ve diğ. 2013; Chen ve diğ. 2012; Chen ve diğ. 2017; Chiam ve diğ. 2007; Chiam ve diğ. 2008; Cui ve diğ. 2014; Cura, 2009; Deng ve Lin, 2010 <sup>a</sup> , 2010 <sup>b</sup> ; Deng ve diğ. 2012; Fernandez ve Gomez, 2007; Gaspero ve diğ. 2011; Golmakani ve Alishah, 2008; He ve Qu, 2016; Jin ve diğ. 2015; Kalayci ve diğ. 2017; Kamili ve Riffi, 2015, 2016; Kao ve Cheng, 2013; Koshino ve diğ. 2007; Kumar ve Mishra, 2017; Liagkouras, 2018; Liagkouras ve Metaxiotis, 2014, 2017; Lwin ve Qu, 2013; Lwin ve diğ. 2014; Lwin ve diğ. 2013; Mayambala ve diğ. 2015; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Mishra ve diğ. 2009; Moral-Escudero ve diğ. 2006; Mozafari ve diğ. 2011; Ni ve diğ. 2017; Ruiz-Torrubiano ve Suarez, 2010; Ruiz-Torrubiano ve Suárez, 2007; Sabar ve Song, 2014; Sadigh ve diğ. 2012; Sadjadi ve diğ. 2012; Sen ve diğ. 2015; Skolpadungket ve diğ. 2007; Streichen ve Tanaka-Yamawaki, 2006; Streichert ve diğ. 2004 <sup>a</sup> ; Streichert ve diğ. 2004 <sup>b</sup> ; Tian ve diğ. 2016; Tuba ve Bacanin, 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Wang ve diğ. 2015; Wang ve diğ. 2011, 2012; Woodside-Oriakhi ve diğ. 2011; Xu ve diğ. 2011; Xu ve diğ. 2010; Yin ve diğ. 2015 <sup>a</sup> ; Yin ve diğ. 2015 <sup>b</sup> ; Zhang ve diğ. 2018 <sup>a</sup> )
Case study	(Abbas ve Haider, 2009; Ackora-Prah ve diğ. 2014 <sup>a</sup> ; Ackora-Prah ve diğ. 2014 <sup>b</sup> ; Aouni ve diğ. 2005; Aranha ve Iba, 2009; Arkeman ve diğ. 2013; Bacanin ve diğ. 2014; Bevilacqua ve diğ. 2011; Bonami ve Lejeune, 2009; Branke ve diğ. 2009; Brito ve Vicente, 2014 <sup>a</sup> ; Buseti, 2006; Cao ve Tao, 2010; Chang ve Chen, 2008; Chang ve Hsu, 2007; Chen ve Zhou, 2018; Chen ve Cai, 2008; Chen ve diğ. 2008; Chen ve diğ. 2006; Chen ve Zhang, 2010; ChiangLin, 2006; Corazza ve diğ. 2012 <sup>a</sup> ; Corazza ve diğ. 2013; Crama ve Schyns, 2003; Cui ve diğ. 2013; Dehghan Hardoroudi ve diğ. 2017; Dreżewski ve Doroz, 2017; Duran ve diğ. 2009; Eftekharian ve diğ. 2017; Ehr Gott ve diğ. 2004; Farzi ve diğ. 2013; Fasheng ve Wei, 2006; Fieldsend ve diğ. 2004; Freitas ve diğ. 2009; Gao ve Chu, 2009; García ve diğ. 2018; Garcia ve diğ. 2012; Golmakani ve Fazel, 2011; Hadi ve diğ. 2016; Hao ve Liu, 2009; Hoklie ve Zuhail, 2010; Hong-mei ve diğ. 2010; Hu ve Zhangy, 2010; Huang ve Shen, 2010; Huang, 2012; Jalota ve Thakur, 2018; Jiang ve diğ. 2014; Jiang ve diğ. 2008; Kamali, 2014; Kıriş ve Ustun, 2012; Kocadağlı ve Keskin, 2015; Kumar ve Bhattacharya, 2012; Lai ve diğ. 2006; Lean ve diğ. 2008; Li ve diğ. 2006; Li ve diğ. 2010; Liagkouras, 2018; Liagkouras ve Metaxiotis, 2018; Liang ve Qu, 2013; Lin ve Liu, 2008; Lin ve diğ. 2005; Loukeris ve diğ. 2009; Lu ve Wang, 2013; Lwin ve diğ. 2017; Mansour ve diğ. 2007; Maringer ve Kellerer, 2003; Niu ve diğ. 2010; Niu ve diğ. 2009; Ong ve diğ. 2005; Pai ve Michel, 2009; Paiva ve diğ. 2019; Peng ve diğ. 2011; Pouya ve diğ. 2016; Qu ve diğ. 2017; Reid ve Malan, 2015; Rong ve diğ. 2009; Ruiz-Torrubiano ve Suarez, 2015; Schaerf, 2002; Shaikh ve Abbas, 2009; Shaw ve diğ. 2008; Shoaf ve Foster, 1998; Strumberger ve diğ. 2017; Suganya ve Vijayalakshmi Pai, 2009; Sun ve diğ. 2011; Suthiwong ve Sodanil, 2016; Talebi ve diğ. 2010; Tan ve diğ. 2013; Tan ve diğ. 2014; Tang ve diğ. 2009; Thomaidis, 2010; Tuba ve diğ. 2013, 2014; Wang ve diğ. 2009; Xu ve Chen, 2006; Xu ve diğ. 2007; Yaakob ve Watada, 2010; Yu ve diğ. 2009; Zaheer ve Pant, 2016; Zhang ve diğ. 2010; Zhou ve Li, 2014; Zhu ve diğ. 2011)
Varsayıma Dayalı	(Bacanin ve Tuba, 2015; Ehr Gott ve diğ. 2004; Soleimani ve diğ. 2009; Xia ve diğ. 2000)



Şekil 2.8: Veri setlerinin dağılımı.

## 2.4 Performans Ölçütleri

Literatürde OVPO probleminin çözümü için geliştirilmiş algoritmaların karşılaştırılması ve performans analizi için farklı performans ölçütleri kullanılmaktadır. Bu performans ölçütleri, dört ana başlık altında sınıflandırılmıştır: yakınsama tabanlı (convergence based), çeşitlilik tabanlı (diversity based), hibrit ve riske uyarlanmış (risk-adjusted) göstergeler.

*Yakınsama tabanlı göstergeler (Convergence based indicators):* Geliştirilen algoritma ile elde edilen çözüm setinin teorik pareto optimal sınırına olan yakınsaklığını temsil etmektedir.

- Ortalama Öklid Uzaklığı (Mean Euclidean Distance): Teorik etkin sınır ve algoritma tarafından elde edilen etkin sınır üzerindeki her bir portföy alternatifi arasındaki ortalama doğrusal uzaklıktır (Cura 2009).
- Getiri Hatasının Varyansı (Variance of Return Error): Belirli bir getiri seviyesine göre her iki etkin sınır üzerindeki portföy alternatiflerinin her birisi arasındaki yatay mesafelerin ortalamasıdır (Fernandez ve Gomez 2007).

- Getiri Hatasının Ortalaması (Mean Return Error): Belirli bir risk seviyesine göre her iki etkin sınır üzerindeki portföy alternatiflerinin her birisi arasındaki dikey mesafelerin ortalamasıdır (Fernandez ve Gomez, 2007).
- Yüzde Hatasının Ortalaması (Mean Percentage Error): Yüzde hatası değerlerinin ortalamasını ifade etmektedir (Chang ve diğ. 2000).
- Yüzde Hatasının Medyanı (Median Percentage Error): Tüm yüzde hata değerlerinin medyanını ifade etmektedir (Chang ve diğ. 2000).
- Kuşak Mesafesi (KM) (Generational Distance): Önerilen çözüm algoritması ile elde edilen etkin sınırla kesin pareto sınırı arasındaki ortalama mesafeyi ölçmektedir (Van Veldhuizen ve Lamontv 2000). Bu değerin sıfıra eşit olması, önerilen çözüm algoritması tarafından üretilen tüm çözümlerin kesin Pareto sınırının üzerinde olduğu anlamına gelmektedir. Bir başka ifadeyle, küçük KM değeri daha iyi yakınsama performansı sağlandığını göstermektedir.
- Epsilon Göstergesi: Tüm amaçlar dikkate alındığında bir çözüm setinin diğerinden daha kötü olup olmadığını gösteren ikili bir göstergedir (Zitzler ve diğ. 2003). Bu bağlamda her iki pareto sınırının ayrı ayrı dağılımlarının birbirinden pozitif veya negatif üstünlük durumuna göre ne kadar farklı olduğunu ifade etmektedir.
- Yakınsaklık Ölçütü (Convergence metric): Teorik etkin sınırı dikkate almadan, iki üstünlük kurulmamış (non-dominated) çözüm kümesinin yakınsama kalitesini karşılaştırmak için kullanılmaktadır (Khare ve diğ. 2003).

*Çeşitlilik Tabanlı Göstergeler (Diversity based indicators):* Elde edilen çözümlerin pareto sınırı boyunca dağılımının homojenliğini temsil etmektedir.

- Aralık Ölçütü (Spacing Metric): Üstünlük kurulmamış pareto sınırı üzerindeki çözümlerin dağılım karakteristiğini tanımlamaktadır (Van Veldhuizen ve Lamont 2000). Diğer bir ifadeyle, üstünlük kurulmamış çözümlerin ne kadar iyi dağılım gösterdiğini ifade etmektedir.
- Yayılma (diversity) Ölçütü (Spread Metric): Yayılmanın ölçüsünü ifade etmektedir. Çözüm uzayındaki noktaların sezgisel etkin sınır üzerinde ne ölçüde eşit dağıldığını göstermektedir (Deb ve diğ. 2002).

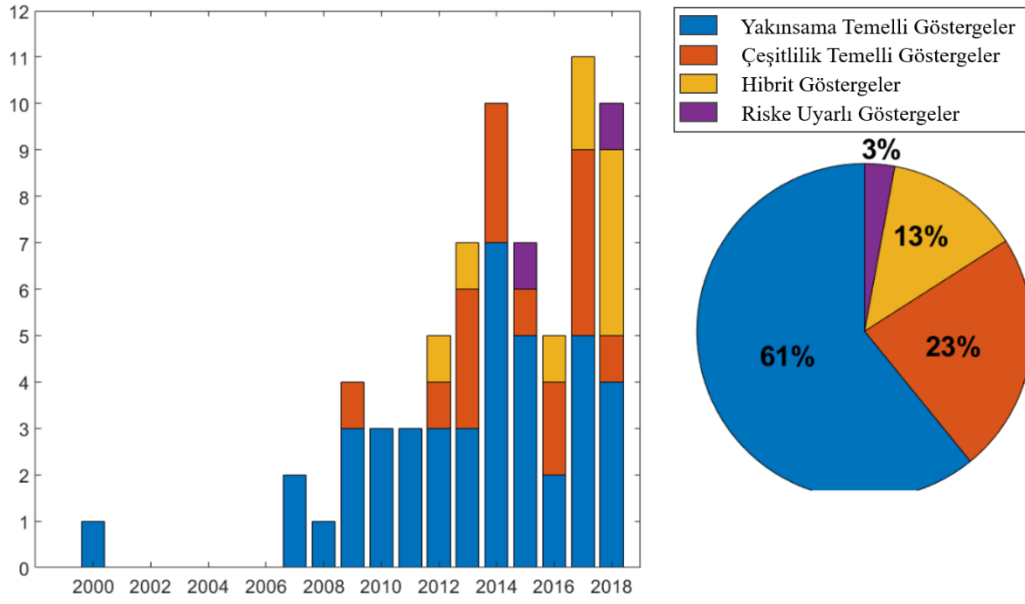
*Hibrit Performans Göstergeleri:* Yakınsama ve çeşitlilik tabanlı göstergelerin hibritlenmesi ile elde edilen göstergeleri ifade etmektedir.

- Hipervolüm Göstergesi (Hypervolume indicator): Bir çok-amaçlı çözüm algoritması tarafından elde edilen üstünlük kurulmamış çözüm seti tarafından üstünlük kurulan iki boyutlu alanın hacmini ifade etmektedir (Zitzler ve Thiele 1999). Yüksek hipervolüm değeri çözüm setinin teorik etkin sınıra daha iyi yakınsama performansı gösterdiğini ifade etmektedir.
- $D1_R$ : Optimal pareto sınırına yakınsamanın yanı sıra en yakın çözümün etkin sınıra olan ortalama mesafesini ifade eder (Czyżżak ve Jaszkiwicz 1998).
- Tersinir Kuşak Mesafesi (TKM) (Inverted Generational Distance): KM performans ölçütünden farklı olarak etkin sınırlar arasındaki mesafe ölçümü tam aksi yönde yapılmaktadır. Yani KM’de ölçüm sezgisel etkin sınırdan teorik etkin sınıra yapılırken TKM’de bunun tam tersi geçerlidir. Bu durum hızlı hesaplama avantajı sunmaktadır (Van Veldhuizen ve Lamont 2000).

*Riske Uyarlanmış Göstergeler:* Muhtemel getirilerin faydalı etkilerini ve aynı zamanda kayıpların zararlı etkilerini içeren bir getiri dağıtımının risk-ödül değerlendirmesini temsil eder.

- Sharpe oranı: Portföyün riske göre getirisini ölçmek için kullanılmaktadır. (Sharpe 1966). Diğer bir ifadeyle sharpe oranı portföyün getirisinin yatırımcı tarafından alınan riski ne kadar iyi karşıladığını ifade etmektedir. Yüksek sharpe oranı daha iyi performansı göstermektedir.
- Omega oranı: Sharpe oranı ortalama varyans yapısı ve normal dağılmış bir girdi verisini varsayımına dayanmaktayken, omega oranı dağılıma ait tüm momentleri performans ölçümünde kullanır. Böylelikle kayıp-kazanç oranı ölçülür (Keating ve Shadwick 2002).

OVPO literatüründe kullanılan performans göstergelerinin dağılımı Şekil 2.9’da sunulmuştur.



Şekil 2.9: Performans göstergelerinin dağılımı.

Algoritma tarafından üretilen kısıtlı etkin sınırla, referans olarak kabul edilen kısıtsız etkin sınır arasındaki mesafeyi hesaplayan yakınsama tabanlı performans göstergeleri Tablo 2.7’de ilgili yayınlarla birlikte gösterilmiştir. Çeşitlilik, hibrit tabanlı ve riske uyarlanmış performans göstergeleri ise sırasıyla Tablo 2.8, Tablo 2.9 ve Tablo 2.10’da sunulmuştur.

**Tablo 2.7:** OVPO literatüründeki yakınsama tabanlı performans göstergeleri.

Performans Ölçütü	Yayın
Ortalama Öklid Uzaklığı	(Bacanin ve Tuba, 2014; Baykasoglu ve diğ. 2015; Chen ve diğ. 2017; Cura, 2009; Jin ve diğ. 2015; Kalayci ve diğ. 2017; Kamili ve Riffi, 2015, 2016; Kao ve Cheng, 2013; Kumar ve Mishra, 2017; Lwin ve diğ. 2013; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Mozafari ve diğ. 2011; Ni ve diğ. 2017; Sadigh ve diğ. 2012; Sen ve diğ. 2015; Tuba ve Bacanin, 2014 <sup>a</sup> ; Wang ve diğ. 2011, 2012; Yin ve diğ. 2015 <sup>a</sup> )
Getiri Hatasının Varyansı ve Ortalaması	(Bacanin ve Tuba, 2014; Baykasoglu ve diğ. 2015; Cura, 2009; Fernandez ve Gomez, 2007; Golmakani ve Alishah, 2008; Kalayci ve diğ. 2017; Kamili ve Riffi, 2015, 2016; Kao ve Cheng, 2013; Mishra ve diğ. 2014 <sup>a</sup> ; Mozafari ve diğ. 2011; Ni ve diğ. 2017; Sadigh ve diğ. 2012; Tuba ve Bacanin, 2014 <sup>a</sup> ; Wang ve diğ. 2011, 2012; Yin ve diğ. 2015 <sup>a</sup> )
Yüzde Hatasının Ortalaması	(Baykasoglu ve diğ. 2015; Chang ve diğ. 2000; Cui ve diğ. 2014; Deng ve Lin, 2010 <sup>a</sup> , 2010 <sup>b</sup> ; Deng ve diğ. 2012; Jalota ve Thakur, 2018; Jin ve diğ. 2015; Kalayci ve diğ. 2017; Lwin ve Qu, 2013; Pai ve Michel, 2009; Sabar ve Song, 2014; Tuba ve Bacanin, 2014 <sup>b</sup> ; Woodside-Oriakhi ve diğ. 2011; Xu ve diğ. 2010)
Yüzde Hatasının Medyanı	(Chang ve diğ. 2000)
Kuşak Mesafesi Epsilon Göstergesi	(Lwin ve diğ. 2017; Skolpadungket ve diğ. 2007)
Yakınsaklık Ölçütü	(Liagkouras, 2018; Liagkouras ve Metaxiotis, 2017, 2018)
	(Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>b</sup> ; Mishra ve diğ. 2009; Sen ve diğ. 2015)

**Tablo 2.8:** OVPO literatüründeki çeşitlilik tabanlı performans göstergeleri.

Performans Ölçütü	Yayın
Aralık Ölçütü	(Chen ve Zhou, 2018; Eftekharian ve diğ. 2017; Kumar ve Mishra, 2017; Mishra ve diğ. 2016; Mishra ve diğ. 2009)
Yayımla Ölçütü	(Chen ve diğ. 2013; Chen ve diğ. 2012; Chen ve diğ. 2017; Liagkouras ve Metaxiotis, 2014; Liang ve Qu, 2013; Lwin ve diğ. 2014; Lwin ve diğ. 2013; Lwin ve diğ. 2017; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>b</sup> ; Mishra ve diğ. 2009; Sen ve diğ. 2015; Suthiwong ve Sodanil, 2016)

**Tablo 2.9:** OVPO literatüründeki hibrit performans göstergeleri.

Performans Ölçütü	Yayın
D1 <sub>R</sub>	(Chen ve diğ. 2013; Chen ve diğ. 2012; Suthiwong ve Sodanil, 2016)
Tersinir Kuşak Mesafesi	(Liagkouras, 2018; Liagkouras ve Metaxiotis, 2017, 2018; Lwin ve diğ. 2017)
Hipervolüm	(Dreżewski ve Doroz, 2017; Liagkouras, 2018; Liagkouras ve Metaxiotis, 2017, 2018; Lwin ve diğ. 2017; Zhang ve diğ. 2018 <sup>a</sup> )

**Tablo 2.10:** OVPO literatüründeki risk uyarlı performans göstergeleri.

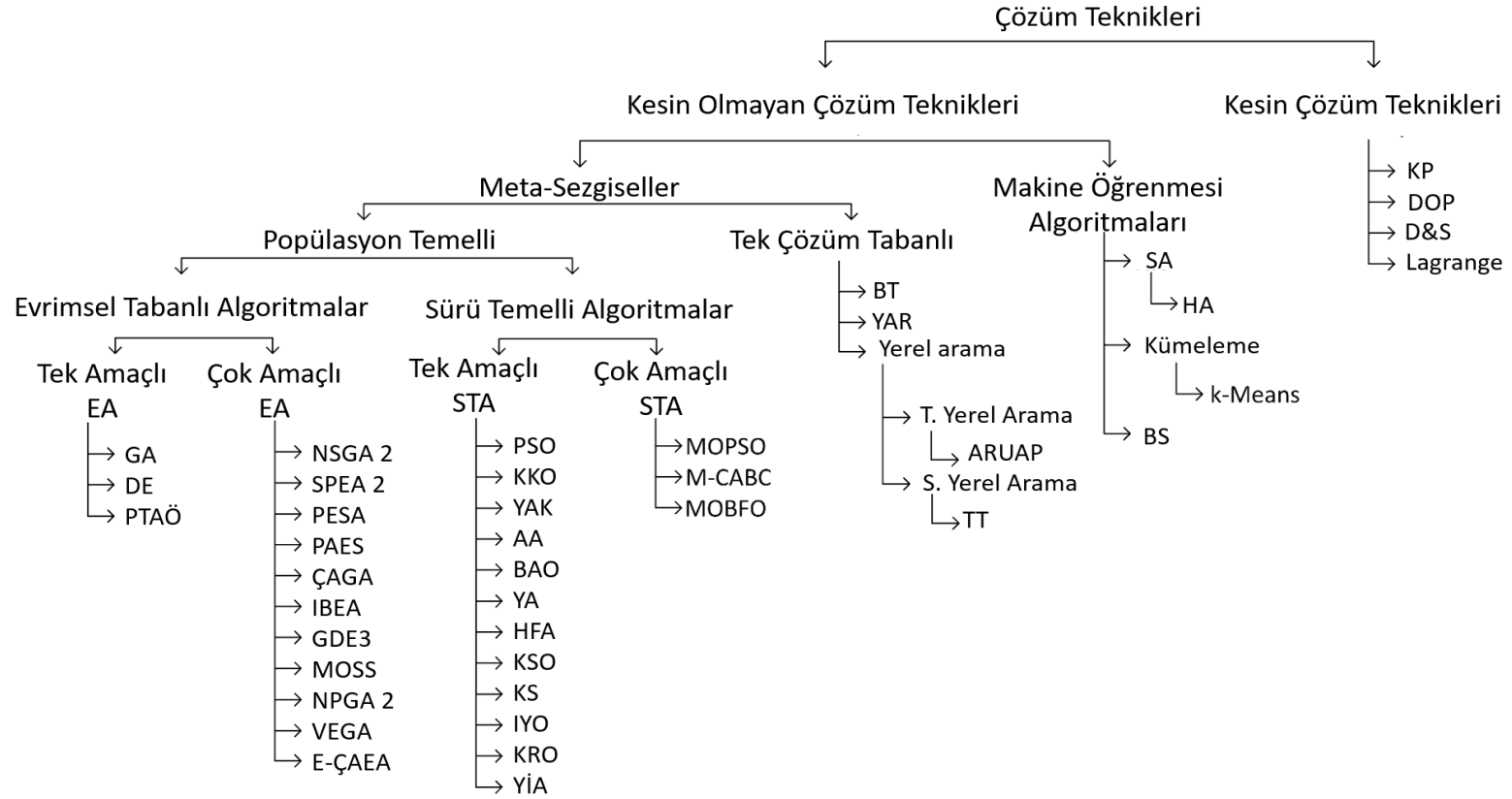
Performans Ölçütü	Yayın
Sharpe Oranı	(Ban ve diğ. 2018; Ruiz-Torrubiano ve Suarez, 2015)

## 2.5 OVPO Modeli Üzerine Uygulamalar

Orijinal OVPO problemi kuadratik yapıya sahiptir, bu sebeple kuadratik programlama gibi kesin çözüm algoritmaları kullanılarak çözülebilmektedir. Fakat, modele eklenen ek kısıtlar ve problem boyutunun büyümesi OVPO problemine kesin çözüm algoritmaları ile kabul edilebilir zaman dilimlerinde çözüm üretilmesini zorlaştırmaktadır. Bu sebeple birçok araştırmacı, matematiksel modelinde konveks olmayan kısıtlar da içeren OVPO problemi için kesin olmayan çözüm yaklaşımlarına başvurmuştur. Şekil 2.10, OV modelini temel alarak PO problemi için kullanılan çözüm yaklaşımlarının geniş bir sınıflandırmasını göstermektedir.

Sonraki bölümlerde, OVPO probleminin çözümü için geliştirilen veya uyarlanan çözüm tekniklerinin ayrıntılı bir literatür incelemesi sunulmuştur.





**Şekil2.10:** OVPO için uygulanan çözüm tekniklerinin sınıflandırılması.

Artificial Cooperative Search (ACS), Yapay Arı kolonisi (YAK), Karınca Kolonisi Optimizasyonu (KKO), Yarasa Algoritması (YA), Bakteriyel Avlanma Optimizasyonu (BAO), Dal & Sınır (D&S), Kimyasal Reaksiyon Optimizasyonu (KRO), Kedi Sürüsü Optimizasyonu (KSO), Diferansiyel Evrim (DE), Ateşböceği Algoritması (AA), Havai Fişek Algoritması (HFA), Genetik Algoritma (GA), Generalized DE3 (GDE3), Açgözlü Rastgele Uyarlanmış Arama Prosedürü (ARUAP), Guided LS (GLS), Tepe Tırmanma (TT), Hopfield Ağları (HA), Indicator-Based Evolutionary Algorithm (IBEA), Invasive Weed Optimization (IWO), Iterated Local Search (ILS), Krill Sürüsü (KS), Yerel Arama (YA), Multi objective co-variance based ABC (M-CABC), Çok Amaçlı GA (ÇAGA), Çok Amaçlı PSO (ÇAPSO), Multi Objective Scatter Search (MOSS), Doğrusal Olmayan Programlama (DOP), Sınır Ağları (SA), Niche Pareto GA 2 (NPGA-II), Non-dominated Sorting GA 2 (NSGA-II), Pareto Archived Evolution Strategy (PAES), Popülasyon Tabanlı Artan Öğrenme (PTAÖ), Pareto Envelope-based Selection Algorithm (PESA), Parçacık Sürü Optimizasyonu (PSO), Kuadratik Programlama (KP), Benzitilmiş Tavlama (BT), Stokastik LS (SLS), Strength Pareto Evolutionary Algorithm 2 (SPEA-II), Yasaklı Arama (TS), Vector Evaluated GA (VEGA), Değişken Komşuluk Arama (DKA)

### 2.5.1 Kesin Çözüm Teknikleri

OVPO kesin çözüm algoritmaları kullanılarak çözülebilmektedir. En yaygın kullanılan kesin çözüm yöntemi ise doğrusal olmayan programlamanın bir çeşidi olan KP'dır. Aouni ve diğ. (2005) ve Mansour ve diğ. (2007) KP kullanarak OVPO için bir hedef programlama modeli geliştirmişlerdir. Hu ve Zhangy (2010) lineer programlama kullanarak OV, Ortalama mutlak sapma, Koşullu riske maruz değer (Conditional Value-at-Risk) ve Minimaks gibi farklı yaklaşımları dikkate alan PO modellerinin performans karşılaştırmalarını yapmışlardır. Peng ve diğ. (2011) kuadratik yapıdaki işlem maliyeti kısıtının da yer aldığı ve KP ile çözülebilen bir OV modeli önermişlerdir. Brito ve Vicente (2014<sup>b</sup>) ilk amaç fonksiyonunun eleman sayısı minimizasyonunu, ikincisinin ise getiri maksimizasyonu ile birlikte risk minimizasyonunu ifade ettiği iki amaçlı bir model önermişlerdir. Mayambala ve diğ. (2015) kovaryans matrisinin öz ayrıştırılmasına dayalı yeni bir OV modeli geliştirmişler ve KP kullanarak performans analizi gerçekleştirmişlerdir. Qin (2015) getirinin eş zamanlı rastsallığı ve belirsizliğini dikkate alarak hibrit portföy seçim problemi için bir OV modeli önermiştir. Elde edilen hesaplamalı sonuçlar önerilen modelin anlamlı ve uygulanabilir olduğunu göstermiştir.

Araştırmacılar KP ve doğrusal olmayan programlama aracılığıyla OVPO probleminin çözümü için çoğunlukla CPLEX ya da MATLAB kuadratik programlama çözücüsü kullanmışlardır. KP ve doğrusal olmayan programlamanın yanı sıra, Lagrange yöntemi (Liv d.2006; Shaw ve diğ. 2008) ve Dal ve Sınır algoritması (Bonami ve Lejeune 2009) da OVPO çözümünde kullanılan yöntemler arasındadır.

**Tablo 2.11:** OVPO için kesin çözüm tekniklerinin kullanıldığı yayınlar.

<b>Teknik</b>	<b>Yayınlar</b>
<b>KP</b>	(Abbas ve Haider, 2009; Aouni ve diğ. 2005; Baykasoglu ve diğ. 2015; Brito ve Vicente, 2014 <sup>a</sup> ; Cesarone ve diğ. 2013, 2015; Cui ve diğ. 2014; Cui ve diğ. 2013; Gaspero ve diğ. 2011; Hu ve Zhangy, 2010; Jiang ve diğ. 2014; Jin ve diğ. 2015; Liagkouras ve Metaxiotis, 2014; Mansour ve diğ. 2007; Mayambala ve diğ. 2015; Moral-Escudero ve diğ. 2006; Peng ve diğ. 2011; Ruiz-Torrubiano ve Suarez, 2015)
<b>Doğrusal Olmayan Programlama</b>	(Kırış ve Ustun, 2012; Kocadağlı ve Keskin, 2015)
<b>Dal ve Sınır</b>	(Bonami ve Lejeune, 2009; He ve Qu, 2016)
<b>Lagranj</b>	(Li ve diğ. 2006; Shaw ve diğ. 2008)

Literatür analizi sonucunda OVPO probleminin çözümü için kesin çözüm yöntemlerini kullanan yayınların sınırlı sayıda olduğu görülmektedir.

## **2.5.2 Kesin Olmayan Çözüm Teknikleri**

OVPO modeline eklenen bazı gerçek hayat kısıtlarının orijinal problemi NP-zor sınıfına soktuğu bilinmektedir. Dolayısıyla kabul edilebilir zaman dilimlerinde optimale en yakın sonucun elde edilebilmesine olanak sağlayan yakınsama teknikleri OVPO probleminin çözümü için literatürde yoğun olarak kullanılmaktadır. Bu doğrultuda literatürde kullanılan çözüm yöntemleri iki başlık altında toplanmıştır: meta-sezgisel yöntemler ve makine öğrenmesi algoritmaları.

### **2.5.2.1 Meta-sezgisel yöntemler**

Meta-sezgiseller geniş bir çözüm uzayında en uygun çözüme ulaşmak için çeşitli arama stratejilerini uygulayarak mevcut optimizasyon problemlerini çözmek için geliştirilmiş alt-seviye sezgiselleri yönlendiren iteratif üst-seviye çözüm prosedürleridir. Her iterasyonda meta-sezgiseller tek bir çözüm ya da bir popülasyon kullanabilmektedir. Söz konusu alt-seviye sezgisel basit bir yerel arama algoritmasından ya da özel olarak geliştirilmiş bir çözüm metodolojisinden oluşabilmektedir.

Genel olarak meta-sezgisel yöntemlerin iki temel arama mekanizmaları bulunmaktadır. Bunlar keşif (exploration) ve sömürü (exploitation) mekanizmalarıdır. Keşif mekanizması, muhtemel daha iyi çözümlerin bulunması için yeni çözüm alanlarının keşfedilmesini böylelikle algoritmanın yerel optimum noktalarda takılı kalmasının önüne geçilmesi için ana algoritmaya rehberlik ederken, sömürü mekanizması yeni keşfedilen çözüm alanının daha detaylı incelenerek daha iyi bir komşu çözümün aranmasına vesile olmaktadır. Bu iki mekanizma arasındaki denge ve zamanlama meta-sezgisel algoritmaların performansı ve gürbüz (robust) yapısı için hayati öneme sahiptir. Meta-sezgisel algoritmalar iki başlık altına sınıflandırılabilir: popülasyon tabanlı algoritmalar ve tek çözüm tabanlı algoritmalar.

### 2.5.2.1.1 Popülasyon tabanlı algoritmalar

Popülasyon tabanlı algoritmalar, oldukça geniş bir çözüm alanında gerçekleştirdikleri arama sırasında popülasyonun gücünü kullanmaktadırlar. Popülasyon ne kadar çeşitlendirilirse bununla bağlantılı olarak o kadar geniş bir çözüm alanı taranabilmektedir. Rastgele ya da bir sezgisel aracılığıyla elde edilen başlangıç çözümünün akabinde her bir iterasyonda popülasyon iyileştirme stratejilerinin uygulanmasıyla arama işlemi devam eder. Popülasyon tabanlı algoritmalar iki kategoride incelenebilir: evrimsel tabanlı algoritmalar ve sürü tabanlı algoritmalar.

#### *Evrimsel tabanlı algoritmalar:*

Evrimsel tabanlı algoritmalar doğal evrimsel süreçten esinlenilerek geliştirilmiştir. Bu algoritmaların temel çalışma mekanizması bir başlangıç çözüm popülasyonundan doğal seçim, çaprazlama ve mutasyon stratejilerini kullanarak kromozomlarla temsil edilen yeni çözümler bir başka ifadeyle yeni nesiller oluşturmak üzerine kurulmuştur. Evrimsel algoritmalar arasında en yaygın olarak kullanılan yöntem Holland (1975) tarafından geliştirilen genetik algoritmadır. OVPO literatüründe kullanılan evrimsel tabanlı algoritmalar Tablo 2.12’de özetlenmiştir.

**Tablo 2.12:** OVPO için uygulanan evrimsel tabanlı algoritmalar.

<b>Evrimsel Tabanlı Algoritmalar</b>	<b>Geliştiren</b>	<b>Amaç Tipi</b>
Genetik Algoritma (GA)	Holland (1975)	Tek Amaçlı
Diferansiyel Evrim (DE)	Storn and Price (1997)	Tek Amaçlı
Popülasyon tabanlı artan öğrenme (PTAÖ)	Baluja (1994)	Tek Amaçlı
Non-dominated sorting genetic algorithm 2 (NSGA-II)	Deb ve diğ. (2002)	Çok Amaçlı
Strength Pareto evolutionary algorithm 2 (SPEA2)	Zitzler ve diğ. (2001)	Çok Amaçlı
Pareto envelope-based selection algorithm (PESA)	Corne ve diğ. (2000)	Çok Amaçlı
Pareto archived evolution strategy (PAES)	Knowles ve diğ. (2006)	Çok Amaçlı
Indicator based evolutionary algorithm (IBEA)	Zitzler ve Künzli (2004)	Çok Amaçlı
Çok amaçlı GA (MOGA)	Fonseca ve Fleming (1993)	Çok Amaçlı
Niched Pareto genetic algorithm 2 (NPGA-II)	Erickson ve diğ. (2001)	Çok Amaçlı
Generalized differential evolution 3 (GDE3)	Kukkonen ve Lampinen (2005)	Çok Amaçlı
Multi objective scatter search (MOSS)	Glover ve diğ. (2000)	Çok Amaçlı
Vector evaluated genetic algorithm (VEGA)	Schaffer (1985)	Çok Amaçlı
e-çok amaçlı evrimsel algoritma (e-ÇOEA)	Hanne (2007)	Çok Amaçlı

### ***OVPO için Tek amaçlı evrimsel algoritmalar:***

Shoaf ve Foster (1998) ilk olarak genetik algoritmayı MVPO probleminde uygulamış sonrasında Chang ve diğ. (2000) eleman sayısı kısıtlı PO probleminin çözümü için genetik algoritmayı uygulayarak diğer iki sezgisel algoritmayla karşılaştırmışlardır. Ehr Gott ve diğ. (2004) OVPO için önerdikleri matematiksel modelin çözümü için genetik algoritma, yasaklı arama ve benzetilmiş tavlama sezgisellerini uygulamışlardır. Getiri ve varyans dikkate alınarak yaptıkları performans değerlendirmesine göre genetik algoritmanın diğerlerine göre daha iyi ve daha stabil sonuçlar ürettiğini rapor etmişlerdir. Benzer şekilde Busetti (2006) de yasaklı arama ve genetik algoritmayı OVPO probleminde uygulamışlar ve genetik algoritmanın daha hızlı ve etkin sonuçlar verdiğini ifade etmişlerdir. Lin ve Liu (2008), minimum işlem lotu (minimum transaction lot) kısıtını da modele eklemişler ve kısıtlı modelin çözümü için genetik algortmadan yararlanmışlardır. Hesaplamalı sonuçlar, genetik algoritmanın oldukça kısa bir süre içerisinde optimale yakının çözümler elde edebileceğini göstermiştir. Chang ve diğ. (2009), OVPO probleminin çözümünde genetik algoritmanın yatırımcılar için etkin bir çözüm yöntemi olduğunu söylemişler ve yaptıkları analiz sonucunda indeksteki toplam varlık sayısının üçte birinden daha fazla sayıda varlığın portföyde tutulmamasının daha iyi olacağını tespit etmişlerdir. Loukeris ve diğ. (2009) genetik algoritmayı, diferansiyel evrim ve parçacık sürü optimizasyonunu algoritmalarıyla karşılaştırmış, diferansiyel evrim algoritmasının farklı senaryolar için en yüksek ve en güvenilir sonuçları elde ettiğini bildirmişlerdir. Xu ve diğ. (2010) popülasyon tabanlı bir artan öğrenme algoritması önermiş ve algoritmanın performansını Chang ve diğ. (2000) tarafından önerilen genetik algoritma ve Cura (2009) tarafından önerilen parçacık sürüsü optimizasyonu ile karşılaştırmışlardır. Popülasyona dayalı artan öğrenme algoritmasının diğer algoritmalarla benzer sonuçlar elde ettiğini bildirmişlerdir. Woodside-Oriakhi ve diğ. (2011), Chang ve diğ. (2000) tarafından tasarlanan algoritmalarından tüm veri setlerinde ortalama yüzde hatası ve hesaplama süresi gibi performans göstergelerinde daha iyi performans sergileyen üç algoritma (genetik algoritma, tabu arama ve benzetme tavlama) geliştirmişlerdir. Sadjadi ve diğ. (2012), eleman sayısı kısıtlı OVPO probleminin etkin olarak çözümü için genetik algoritma temelli bir çözüm stratejisi geliştirmişlerdir. Geliştirdikleri bu çözüm yönteminde PO probleminin belirsiz yapısını da dikkate almışlardır. Liu ve Zhang (2015), çok dönemli bulanık bir PO

modeli önermiş ve önerilen modelin çözümü için genetik algorithmadan yararlanmışlardır

### ***OVPO için Çok amaçlı evrimsel algoritmalar:***

Çok amaçlı evrimsel algoritma ilk defa Fieldsend ve diğ. (2004) tarafından MVPO probleminde uygulanmıştır. Ong ve diğ. (2005), beklenen getirinin maksimizasyonu, belirsizlik ve ilişki riskinin ise minimizasyonunu dikkate alan üç amaçlı bir PO modeli önermişlerdir. Önerdikleri bu modelin çözümü için NSGA-II algoritmasından yararlanmışlar ve NSGA-II'nin esnek ve doğru sonuçlar verebildiğini göstermişlerdir. Skolpadungket ve diğ. (2007) vektör değerlendirmeli genetik algoritma, çok amaçlı genetik algoritma, SPEA2 ve NSGA-II gibi çok amaçlı evrimsel algoritmaları eleman sayısı ve limit kısıtlarını içeren OVPO probleminin çözümü için uygulamışlardır. Elde edilen test sonuçlarına göre, çok amaçlı genetik algoritmalar ve SPEA2 daha karmaşık yapılar gerektirse de diğerlerinden daha iyi performans göstermiştir. Anagnostopoulos ve Mamanis (2010) beklenen getiriyi maksimize eden ederken riski ve portföyde tutulan varlık sayısını minimize etmeyi hedefleyen üç amaçlı bir PO modeli önermişlerdir. Optimale yakın çözümler NSGA-II, Pareto zarflama temelli seçim algoritması ve SPEA2 algoritmaları kullanılarak elde edilmiş ve test sonuçlarına göre SPEA-II algoritmasının diğerlerine göre üstünlük gösterdiği bildirilmiştir. Anagnostopoulos ve Mamanis (2011<sup>a</sup>), MVPO problemini çözmek için beş farklı çok amaçlı ve bir de tek amaçlı evrimsel algoritma kullanmışlardır. Hesaplamalı sonuçlara göre, tüm çok amaçlı evrimsel algoritmaların, epsilon performans göstergesi dikkate alındığında tüm veri setlerinde tek amaçlı evrimsel algoritmalarından daha iyi ve daha hızlı performans gösterdiği tespit edilmiştir. Ayrıca, NSGA-II ve SPEA2'nin algoritmalar arasında en iyi performans gösteren algoritmalar olduğunu raporlanmıştır. Liagkouras ve Metaxiotis (2014) yeni bir mutasyon (probe guided mutation) operatörü geliştirmişler ve geliştirdikleri bu operatörü NSGA-II ve SPEA2 gibi algoritmalarda klasik olarak kullanılan operatörlerle karşılaştırmışlardır. Performans karşılaştırılmasında hipervolum, epsilon ve yayılma performans göstergelerinden yararlanılmıştır. Test sonuçlarına göre yeni geliştirilen operatörün diğerlerine göre üstünlük gösterdiği ifade edilmiştir. Lwin ve diğ. (2014), OVPO problemi için öğrenme rehberli çok amaçlı evrimsel bir algoritma önermiş ve çeşitlilik ve aşırı hacimli performans göstergelerine göre karşılaştırmıştır. Önerilen çözüm

algoritmasının özellikle büyük veri setlerinde iyi performans gösterdiği ifade edilmiştir.

MVPO problemi için uygulanan evrimsel algoritmalar Tablo 2.13'te gösterilmektedir.

**Tablo 2.13:** OVPO için evrimsel tabanlı algoritma kullanılan yayınlar.

<b>Kategori</b>	<b>Evrimsel Algoritma</b>	<b>Yayınlar</b>
<b>TAEA</b>	<b>GA</b>	(Ackora-Prah ve diğ. 2014 <sup>a</sup> ; Aranha ve Iba, 2009; Buseti, 2006; Chang ve diğ. 2000; Chang ve diğ. 2009; Chen ve diğ. 2008; ChiangLin, 2006; Corazza ve diğ. 2012 <sup>a</sup> ; Coutino-Gomez ve diğ. 2003; Dreżewski ve Doroz, 2017; Ehrgott ve diğ. 2004; García ve diğ. 2018; Hadi ve diğ. 2016; Hao ve Liu, 2009; Hoklie ve Zuhail, 2010; Huang ve Shen, 2010; Huang, 2012; Kamali, 2014; Lai ve diğ. 2006; Lean ve diğ. 2008; Lin ve Liu, 2008; Lin ve diğ. 2005; Loukeris ve diğ. 2009; Lu ve Wang, 2013; Moral-Escudero ve diğ. 2006; Pai ve Michel, 2009; Rong ve diğ. 2009; Ruiz-Torrubiano ve Suarez, 2010, 2015; Ruiz-Torrubiano ve Suárez, 2007; Sabar ve Song, 2014; Sadjadi ve diğ. 2012; Shaikh ve Abbas, 2009; Shoaf ve Foster, 1998; Soleimani ve diğ. 2009; Talebi ve diğ. 2010; Thomaidis, 2010; Woodside-Oriakhi ve diğ. 2011; Xia ve diğ. 2000; Yu ve diğ. 2009)
	<b>DE</b>	(Loukeris ve diğ. 2009; Lwin ve Qu, 2013; Zaheer ve Pant, 2016)
	<b>PBIL</b>	(Jin ve diğ. 2015; Lwin ve Qu, 2013; Xu ve diğ. 2010)
<b>ÇAEA</b>	<b>NSGA ve NSGA-II</b>	(Anagnostopoulos ve Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos ve Mamanis, 2011 <sup>b</sup> ; Arkeman ve diğ. 2013; Branke ve diğ. 2009; Chen ve Zhou, 2018; Chiam ve diğ. 2007; Chiam ve diğ. 2008; Duran ve diğ. 2009; Fieldsend ve diğ. 2004; Garcia ve diğ. 2012; Liagkouras, 2018; Liagkouras ve Metaxiotis, 2014; Lwin ve diğ. 2014; Macedo ve diğ. 2017; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> ; Mishra ve diğ. 2009; Ong ve diğ. 2005; Qu ve diğ. 2017; Sen ve diğ. 2015; Skolpadungket ve diğ. 2007; Streichen ve Tanaka-Yamawaki, 2006; Streichert ve diğ. 2004 <sup>a</sup> ; Streichert ve diğ. 2004 <sup>b</sup> )
	<b>SPEA2</b>	(Anagnostopoulos ve Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos ve Mamanis, 2011 <sup>b</sup> ; Chen ve Zhou, 2018; Duran ve diğ. 2009; Garcia ve diğ. 2012; Liagkouras, 2018; Liagkouras ve Metaxiotis, 2014; Lwin ve diğ. 2014; Macedo ve diğ. 2017; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> ; Mishra ve diğ. 2009; Skolpadungket ve diğ. 2007)
	<b>PESA ve PESA2</b>	(Anagnostopoulos ve Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos ve Mamanis, 2011 <sup>b</sup> ; Lwin ve diğ. 2014; Mishra ve diğ. 2014 <sup>a</sup> )
	<b>PAES</b>	(Lwin ve diğ. 2014)
	<b>MOGA</b>	(Bevilacqua ve diğ. 2011; Skolpadungket ve diğ. 2007)
	<b>IBEA</b>	(Duran ve diğ. 2009)
	<b>GDE3</b>	(Garcia ve diğ. 2012)
	<b>MOSS</b>	(Lwin ve diğ. 2013)
	<b>NPGA-II</b>	(Anagnostopoulos ve Mamanis, 2011 <sup>a</sup> )
	<b>VEGA</b>	(Skolpadungket ve diğ. 2007)
	<b>e-MOEA</b>	(Anagnostopoulos ve Mamanis, 2011 <sup>a</sup> )
	<b>Co-EMOA</b>	(Dreżewski ve Doroz, 2017)
	<b>MODE-GL</b>	(Lwin ve diğ. 2017)
	<b>MOEA/D-C</b>	(Zhang ve diğ. 2018 <sup>a</sup> )

DE: Differential Evolutionary, GA: Genetik Algoritma, GDE3: Generalized DE3, MOGA: Çok Amaçlı GA, MOPSO: Çok Amaçlı PSO, MOSS: Multi Objective Scatter Search, NPGA-II: Niched Pareto GA 2, NSGA-II: Non-dominated Sorting Genetic Algorithm II, PAES: Pareto Archived Evolution Strategy, PBIL: Population Based Incremental Learning, PESA: Pareto Envelope-based Selection Algorithm, SPEA-II: Strength Pareto Evolutionary Algorithm 2, VEGA: Vector Evaluated GA

GA, şu ana kadarki OVPO literatüründe en çok tercih edilen evrimsel algoritmalarından birisiyken, son yıllarda ÇAEA'ların kullanıldığı bu alandaki yayın sayısındaki artış dikkat çekmektedir.

### ***Sürü tabanlı algoritmalar:***

Sürü tabanlı algoritmalar, sürüler halinde yaşayan canlıların doğal yaşamlarındaki davranışlarından ilham alarak geliştirilen çözüm yaklaşımlarıdır. Balık ve kuş sürüleri, karınca kolonileri ve arılar gibi canlı toplulukları doğal yaşamlarındaki birtakım aktiviteleri için birlikte hareket ederler (çoğunlukla avlanma). Sürü tabanlı algoritmalar bu iş birliğinin hesaplama algoritmaları şeklinde uyarlanmasıyla ortaya çıkmıştır. En popüler sürü temelli algoritma, Kennedy ve Eberhart (1995) tarafından geliştirilen parçacık sürü optimizasyonu (PSO) algoritmasıdır.

Bu çalışma kapsamında, sürü tabanlı algoritmalar iki kategoride incelenmiştir: tek amaçlı sürgü tabanlı algoritmalar ve çok amaçlı sürü tabanlı algoritmalar. OVPO problemine uygulanan sürü tabanlı algoritmalar Tablo 2.14'te gösterilmektedir.

**Tablo 2.14:** OVPO problemine uygulanan sürü temelli algoritmalar.

<b>Sürü Tabanlı Algoritmalar</b>	<b>Geliştiren</b>	<b>Amaç Tipi</b>
Parçacık Sürü Optimizasyonu (PSO)	Kennedy ve Eberhart (1995)	Tek Amaçlı
Karınca Kolonisi Optimizasyonu (KKO)	Dorigo ve diğ. (1996)	Tek Amaçlı
Yapay Arı Kolonisi (YAK)	Karaboga (2005)	Tek Amaçlı
Ateşböceği Algoritması (AA)	Yang (2010 <sup>b</sup> )	Tek Amaçlı
Bakteriyel Avlanma Algoritması (BAA)	Passino (2002)	Tek Amaçlı
Yarasa Algoritması (YA)	Yang (2010 <sup>a</sup> )	Tek Amaçlı
Havai Fişek Algoritması (HFA)	Tan ve Zhu (2010)	Tek Amaçlı
Kedi Sürüsü Optimizasyonu (KSO)	Chu ve Tsai (2007)	Tek Amaçlı
Kril Sürüsü (KS)	Gandomi ve Alavi (2012)	Tek Amaçlı
İstilacı Yabancı Ot Optimizasyonu (IYO)	Mehrabian ve Lucas (2006)	Tek Amaçlı
Kimyasal Reaksiyon Optimizasyonu (KRO)	Lam ve Li (2010)	Tek Amaçlı
Yapay İşbirlikçi Arama (YİA)	Civicioglu (2013)	Tek Amaçlı
Çok Amaçlı Parçacık Sürü Optimizasyonu (ÇAPSO)	Coello ve diğ. (2004)	Çok Amaçlı
Çok Amaçlı Kovaryans Temelli Yapay Arı Kolonisi (ÇA-KYAK)	Kumar ve Mishra (2017)	Çok Amaçlı
Çok Amaçlı Bakteriyel Avlanma Algoritması (ÇABAA)	Panigrahi ve diğ. (2011)	Çok Amaçlı

### ***OVPO için Tek amaçlı sürü temelli algoritmalar:***

Chen ve diğ. (2006) OVPO problemine parçacık sürü optimizasyonu algoritmasını ilk defa uygulamışlardır. Cura (2009), limit ve eleman sayısı kısıtlı



OVPO probleminin çözümü için parçacık sürü optimizasyon algoritmasını uyarlamış ve genetik algoritma, yasaklı arama ve benzetilmiş tavlama algoritmaları ile karşılaştırmıştır. Chang ve diğ. (2000), dört farklı algoritma için eleman sayısı kısıtlı PO problemi üzerinde yapmış oldukları performans karşılaştırması sonucunda genel anlamda hiçbir algoritmanın tam anlamıyla birbirine üstünlük sağlamadığını fakat özellikle düşük risk seviyelerinde öklid uzaklığı, yüzde katkı, getiri hatasının varyansı ve getiri hatasının ortalaması gibi performans göstergelerine göre parçacık sürü optimizasyonu algoritmasının daha iyi performans sergilediğini ifade etmişlerdir. Deng ve Lin (2010<sup>a</sup>) ilk olarak OVPO problemi için ilk defa karınca kolonisi optimizasyonu algoritmasını uygulamışlar parçacık sürü optimizasyonu algoritması ile yapmış oldukları performans karşılaştırması sonrasında ve özellikle düşük riskli yatırım alternatiflerinde gürbüz ve etkin sonuçlar verdiğini raporlamışlardır. Hong-mei ve diğ. (2010) OVPO problemi için ilk defa yapay arı kolonisi algoritmasını uyarlamış ve algoritmanın bu problem tipi için başarılı sonuçlar verdiğini belirtmişlerdir. Zhu ve diğ. (2010) yine bu problem için iki farklı sürü tabanlı algoritma önermişlerdir: karınca kolonisi optimizasyonu ve parçacık sürüsü optimizasyonu algoritmaları. Hesaplamalı sonuçlar, karınca kolonisi optimizasyonunun küçük ve büyük ölçekli portföylerde parçacık sürüsü optimizasyonu algoritmasından daha iyi performans gösterdiğini, ancak orta ölçekli portföylerde Sharpe oranı performans göstergesine göre bunun tam tersi bir sonucun elde edildiğini ifade etmişlerdir. Golmakani ve Fazel (2011) parçacık sürü optimizasyonunu, genetik algoritma ile karşılaştırmışlar ve PSO'nun daha az sürede daha iyi çözümler bulma başarısını gösterdiğini bildirmişlerdir. Xu ve diğ. (2011), evrimsel sürece bir süper molekül ekleyerek kimyasal reaksiyon optimizasyon algoritmasını PO problemine uyarlamış ve Sharpe oranı performans ölçütüne göre orijinal algoritmanın geliştirildiğini göstermişlerdir. Deng ve diğ. (2012), mutasyon operatörüyle iyileştirilmiş bir parçacık sürü optimizasyonu algoritması tasarlamışlardır. Algoritmayı Chang ve diğ. (2000)'nin önerdiği genetik algoritma, yasaklı arama ve benzetilmiş tavlama algoritmaları ile ortalama yüzde hata performans göstergesine göre karşılaştırmışlar ve önerilen algoritmanın diğerlerine göre üstünlük gösterdiğini belirtmişlerdir. Chen ve diğ. (2013), yapay arı kolonisi algoritması temelli iki aşamalı bir çözüm yaklaşımı geliştirmişlerdir. Geliştirilen algoritmanın ilk aşamasında varlık seçimi işlemi gerçekleştirilirken ikinci aşamasında ise seçilen varlıklar için yatırım oranları belirlenmektedir. Kao ve Cheng (2013) OVPO problemine bakteriyel avlanma optimizasyonu algoritmasını uygulamışlar ve genetik

algoritma, tabu arama ve benzetilmiş tavlama algoritmaları (Chang ve diğ. 2000) ve parçacık sürüsü optimizasyonu algoritması (Cura 2009) ile karşılaştırmışlardır. Karşılaştırmalı sonuçlara göre önerilen çözüm algoritmasının çözüm kalitesi ve zamanı açısından üstünlük gösterdiğini ifade etmişlerdir. Tuba ve diğ. (2013) ilk defa ateş böceği algoritmasını OVPO problemine uygulamışlar ve elde ettikleri sonuçlara göre algoritmanın bu problemi etkin olarak çözme potansiyeli olduğunu bildirmişlerdir. Bacanin ve diğ. (2014) uyguladıkları yapay arı kolonisi algoritmasını genetik algoritma ve ateş böceği algoritması ile karşılaştırmış ve yapay arı kolonisi algoritmasının varyans ve getiri hatası ölçütlerine göre en iyi sonuçları verdiğini ifade etmişlerdir. Tuba ve Bacanin (2014<sup>b</sup>), özellikle algoritmanın son aşmalarında daha yoğun bir sömürü mekanizması uygulayan ateşböceği algoritması temelli bir çözüm yaklaşımı önermişlerdir. Ortalama ve medyan yüzde hata performans ölçütlerine göre önerilen çözüm yaklaşımının orijinal algoritmadan daha iyi performans gösterdiğini bildirmişlerdir. Chen (2015), OVPO için kaotik başlangıç stratejisi ile beraber yapay arı kolonisi algoritması ve parçacık sürü optimizasyonu algoritmalarından oluşan hibrit bir çözüm yaklaşımı geliştirmiştir. Kamili ve Riffi (2015) eleman sayısı ve limit kısıtlı MVPO için kedi sürüsü optimizasyonunu algoritmasını uygulamışlardır. Kamili ve Riffi (2016), parçacık sürü optimizasyonu, kedi sürüsü optimizasyonu ve yarasa algoritmaları kullanarak OVPO problemi üzerine karşılaştırmalı bir çalışma yayınlamışlardır. Elde edilen sonuçlara göre, üç algoritmanın da benzer sonuçlar ürettiği ifade edilmiştir. Ni ve diğ. (2017), dinamik rasgele popülasyon topolojisi stratejileriyle bir parçacık sürü optimizasyonu algoritması önermiş ve elde edilen sonuçlara göre, önerilen bu stratejinin, orijinal algoritmanın performansını arttırdığını göstermişlerdir. Kalayci ve diğ. (2017), limit ve eleman sayısı kısıtlı OVPO için yapay arı kolonisi algoritması temelli bir çözüm yaklaşımı geliştirmişlerdir. Bu çözüm yaklaşımında modele dahil edilen kısıtların işlenmesi için uygunluğa zorlayan ve uygunsuzluğa izin veren bir kısıt işleme prosedürü entegre etmişlerdir. Yapılan performans analizi sonrasında önerilen çözüm yaklaşımının orijinal algoritmadan ve genetik algoritma, yasaklı arama ve benzetilmiş tavlama (Chang ve diğ. 2000), (Deng ve diğ. 2012), popülasyon tabanlı artan öğrenme yaklaşımı (Lwin ve Qu 2013) ve ARUAP-KP (Baykasoğlu ve diğ. 2016) gibi literatürdeki diğer çözüm yaklaşımlarından daha iyi sonuçlar verdiğini ifade etmişlerdir.

### ***OVPO için Çok amaçlı sürü temelli algoritmalar:***

Mishra ve diğ. (2009), OVPO problemine çok amaçlı parçacık sürü optimizasyonu algoritmasını uygulamışlar ve NSGA-II, SPEA2 ve paralel tek sınırlı genetik algoritma ile karşılaştırmışlardır. Deneysel sonuçlar, önerilen çözüm yaklaşımının, tüm diğer çözüm yaklaşımlarından daha iyi performans gösterdiğini belirtmişlerdir. Garcia ve diğ. (2012) OVPO için gürbüz bir yaklaşım ortaya koymuş ve önerilen çözüm yaklaşımını dört farklı çok amaçlı algoritma (NSGA-II, SPEA-II, çok amaçlı parçacık sürüsü optimizasyonu ve genelleştirilmiş diferansiyel evrim algoritmaları) ile karşılaştırmışlardır. Mishra ve diğ. (2014<sup>b</sup>), limit ve eleman sayısı kısıtlı OVPO problemine çok amaçlı bakteriyel avlanma optimizasyonu önermiş ve üç çok amaçlı evrimsel algoritma ile karşılaştırmıştır. Mishra ve diğ. (2016), tahmin tabanlı bir OV modeli önermişler ve önerilen modelin çözümü için üç çok amaçlı algoritma (çok amaçlı parçacık sürüsü optimizasyonu, NSGA-II ve SPEA2) uygulamışlardır. Test sonuçlarına göre çok amaçlı parçacık sürüsü optimizasyonunun en iyi Pareto optimal çözümlerini ürettiği ifade edilmiştir. Kumar ve Mishra (2017), OVPO problemi için birbiriyle çelişen iki farklı amacı eş zamanlı dikkate alan ve çözüm üreten kovaryans temelli ve çok amaçlı yapay arı kolonisi algoritması önermişlerdir.

OVPO problemi için uygulanan sürü tabanlı algoritmalar Tablo 2.15'te özetlenmiştir.

**Tablo 2.15:** OVPO için sürü temelli algoritma kullanılan yayınlar.

<b>Kategori</b>	<b>Sürü Tabanlı Algoritmalar</b>	<b>Yayınlar</b>	
<b>TASTA</b>	<b>PSO</b>	(Abbas ve Haider, 2009; Cao ve Tao, 2010; Chang ve Chen, 2008; Chang ve Hsu, 2007; Chen ve Cai, 2008; Chen ve diğ. 2006; Chen ve Zhang, 2010; Corazza ve diğ. 2012 <sup>a</sup> ; Corazza ve diğ. 2013; Cui ve diğ. 2014; Cura, 2009; Deng ve Lin, 2010 <sup>b</sup> ; Deng ve diğ. 2012; Farzi ve diğ. 2013; Fasheng ve Wei, 2006; Gao ve Chu, 2009; Golmakani ve Fazel, 2011; Kamali, 2014; Kamili ve Riffi, 2016; Koshino ve diğ. 2007; Li ve diğ. 2010; Loukeris ve diğ. 2009; Mozafari ve diğ. 2011; Ni ve diğ. 2017; Niu ve diğ. 2010; Niu ve diğ. 2009; Pouya ve diğ. 2016; Reid ve Malan, 2015; Sadigh ve diğ. 2012; Sun ve diğ. 2011; Talebi ve diğ. 2010; Tang ve diğ. 2009; Thomaidis, 2010; Wang ve diğ. 2015; Wang ve diğ. 2009; Xu ve Chen, 2006; Xu ve diğ. 2007; Yaakob ve Watada, 2010; Yin ve diğ. 2015 <sup>a</sup> ; Yin ve diğ. 2015 <sup>b</sup> ; Zhang ve diğ. 2010; Zhu ve diğ. 2010; Zhu ve diğ. 2011)	
	<b>YAK</b>	(Bacanin ve diğ. 2014; Chen ve diğ. 2013; Chen ve diğ. 2012; Hongmei ve diğ. 2010; Kalayci ve diğ. 2017; Suthiwong ve Sodanil, 2016; Tuba ve Bacanin, 2014 <sup>a</sup> ; Wang ve diğ. 2011, 2012)	
	<b>KKO</b>	(Deng ve Lin, 2010 <sup>a</sup> ; Zhu ve diğ. 2010)	
	<b>AA</b>	(Bacanin ve Tuba, 2014; Tuba ve Bacanin, 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Tuba ve diğ. 2013)	
	<b>BAO</b>	(Kao ve Cheng, 2013; Tan ve diğ. 2013; Tan ve diğ. 2014)	
	<b>YA</b>	(Kamili ve Riffi, 2016; Strumberger ve diğ. 2017)	
	<b>HA</b>	(Bacanin ve Tuba, 2015)	
	<b>KSO</b>	(Kamili ve Riffi, 2015, 2016)	
	<b>KS</b>	(Tuba ve diğ. 2014)	
	<b>IYO</b>	(Pouya ve diğ. 2016)	
	<b>KRO</b>	(Xu ve diğ. 2011)	
	<b>YİA</b>	(Kumar ve Bhattacharya, 2012)	
	<b>ÇASTA</b>	<b>ÇAPSO</b>	(Chen ve Zhou, 2018; Garcia ve diğ. 2012; Lwin ve Qu, 2013; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>a</sup> ; Mishra ve diğ. 2009; Qu ve diğ. 2017; Sen ve diğ. 2015; Zhou ve Li, 2014)
		<b>ÇAYAK</b>	(Kumar ve Mishra, 2017)
<b>ÇABAO</b>		(Mishra ve diğ. 2014 <sup>b</sup> )	

Tablo 2.15'te gösterildiği gibi, diğer sürü temelli algoritmaların arasında en eskilerden olması ve OVPO problemi için herhangi bir dönüşüme ihtiyaç duymaması gibi nedenlerden dolayı sürü temelli algoritmaların arasında parçacık sürü optimizasyonunun bu problem tipi için kullanılan en yaygın çözüm yöntemi olduğu görülmektedir. Algoritmanın orijinal yapısı süreklidir dolayısıyla bu problem tipine doğrudan uyum göstermektedir. Parçacık sürü optimizasyonu algoritmasının yanı sıra, yapay arı kolonisi algoritmasının problem üzerindeki üstün performansı son yıllarda araştırmacılar tarafından dikkat çekmektedir. Bu durum yapay arı kolonisi algoritmasının gelecekte OVPO problemi için daha çok kullanılabilme olasılığını artırmaktadır.

### 2.5.2.1.2 Tek-çözüm tabanlı algoritmalar

Tek çözümü dikkate alarak çalışan algoritmalara aynı zamanda yörünge temelli (trajectory) algoritmalar da denilmektedir (Blum ve Roli 2003). Tek çözüm tabanlı algoritmalar, her bir iterasyonda tek bir çözümü değiştirmeye ve geliştirmeye odaklanmaktadırlar. Her ne kadar popülasyonun gücünden faydalanmıyor olsalar da her bir iterasyonda popülasyonun tamamı yerine tek bir çözümü geliştirdikleri için çok hızlı bir şekilde sonuç alabilmektedirler. OVPO problemine uygulanan tek çözüm tabanlı algoritmalar Tablo 2.16'da özetlenmiştir.

**Tablo 2.16:** OVPO problemine uygulanan tek çözüm tabanlı algoritmalar.

<b>Tek Çözüm Tabanlı Algoritmalar</b>	<b>Geliştiren</b>	<b>Amaç Tipi</b>
Benzetilmiş tavlama (BT)	Kirkpatrick, ve diğ.. (1983)	Tek Amaçlı
Yasaklı Arama (YA)	Glover (1986)	Tek Amaçlı
Açgözlü Rastgele Uyarlanabilir Arama Prosedürü (ARUAP)	Feo and Resende (1995)	Tek Amaçlı
Tepe Tırmanma (TT)	Forrest and Mitchell (1993)	Tek Amaçlı
Arşivlenmiş Çok Amaçlı Benzetilmiş tavlama (AÇABT)	Bandyopadhyay ve diğ. (2008)	Çok Amaçlı

#### ***OVPO için Tek çözüm tabanlı algoritmalar:***

Chang ve diğ. (2000) ve Ehrgott ve diğ. (2004) genetik algoritma, yasaklı arama ve benzetilmiş tavlama algoritmalarını OVPO üzerinde uygulamış ve herhangi bir çözüm yönteminin diğerleri üzerinde belirli bir üstünlük göstermediğini ifade etmişlerdir. Anagnostopoulos ve diğ. (2010) OVPO problemini çözmek için Açgözlü Rastgele Uyarlanabilir Arama Prosedürü (ARUAP) tasarlamışlardır. Woodside-Oriakhi ve diğ. (2011) OVPO problemine üç meta-sezgisel algoritma (genetik algoritma, tabu arama ve benzetme tavlama) önermişlerdir. Algoritma sonuçlarını Chang ve diğ. (2000)'nin elde ettiği sonuçlarla karşılaştırmışlar ve daha iyi sonuçlara daha hızlı ulaşıldığını raporlamışlardır. Fastrich ve Winker (2012), benzetilmiş tavlama algoritmasını bir modifikasyon mekanizması olarak ana çözüm yaklaşımına entegre eden bir hibrit algoritma tasarlamışlardır. Ackora-Prah ve diğ. (2014<sup>b</sup>), Hooke ve Jeeves (1961) tarafından geliştirilen ve patern arama algoritmasını OVPO için uygulamışlar ve algoritmanın etkin sonuçlar ürettiğini ifade etmişlerdir. Sen ve diğ. (2015) benzetilmiş tavlama algoritması tabanlı yeni bir çözüm yaklaşımı geliştirmişler ve çeşitlilik performans göstergelerini dikkate alarak NSGA-II ve çok amaçlı parçacık

sürü optimizasyonu algoritması ile karşılaştırıldılar. Karşılaştırmalı sonuçlara göre önerdikleri çözüm yaklaşımının diğerlerine göre daha iyi performans sergilediğini belirtmişlerdir. Chen ve diğ. (2017), OVPO problemini çözmek için yerel arama tabanlı çok amaçlı bir optimizasyon algoritması önermişlerdir. OVPO problemi için tek çözüm tabanlı algoritmalar uygulayan yayınlar Tablo 2.17'de özetlenmiştir.

**Tablo 2.17:** OVPO için tek çözüm tabanlı algoritmaların uygulandığı yayınlar.

<b>Tek Çözüm Tabanlı Algoritmalar</b>	<b>Yayımlar</b>
<b>BT</b>	(Buseti, 2006); Chang ve diğ. (2000); Coutino-Gomez ve diğ. (2003); Ehrgott ve diğ. (2004); Fastrich and Winker (2012); (García ve diğ. 2018); Li ve diğ. (2009); Schaerf (2002); Thomaidis (2010); (Woodside-Oriakhi ve diğ. 2011)
<b>YA</b>	(Chang ve diğ. 2000; Coutino-Gomez ve diğ. 2003; Crama ve Schyns, 2003; Ehrgott ve diğ. 2004; Maringer ve Kellerer, 2003; Schaerf, 2002; Thomaidis, 2010; Woodside-Oriakhi ve diğ. 2011)
<b>ARUAP</b>	(Anagnostopoulos ve diğ. 2010; Baykasoglu ve diğ. 2015)
<b>TT</b>	(Coutino-Gomez ve diğ. 2003; Gaspero ve diğ. 2011; Schaerf, 2002)
<b>AÇATB</b>	(Sen ve diğ. 2015)
<b>LS Temelli ÇAEA</b>	Chen ve diğ. (2017)
<b>PA</b>	Ackora-Prah ve diğ. (2014 <sup>b</sup> )

Benzetilmiş Tavlama (BT); Yasaklı Arama (YA); Açgözlü Rastgele Uyarlanabilir Arama Prosedürü (ARUAP); Tepe Tırmanma (TT); Arşivlenmiş Çok Amaçlı Benzetilmiş tavlama (AÇABT); Patern Arama (PA)

Tablo 2.17'de gösterildiği gibi, popülasyon temelli çözüm yaklaşımlarına kıyasla, tek çözüm temelli algoritmalar OVPO problemini çözümü için yaygın olarak kullanılmamaktadır. MVPO probleminde, kesikli yerine sürekli yapıya sahip olması tek çözüm tabanlı algoritmalar için çözüm uzayını etkin bir şekilde keşfedilmesini zor kılmaktadır.

### 2.5.2.2 Makine öğrenmesi algoritmaları

Makine öğrenmesi, yapay zekada alanındaki örüntü tanıma çalışmaları ve hesaplamalı öğrenme kuramı üzerinden geliştirilmiş bir çözüm yaklaşımıdır. Makine öğrenme algoritmaları, sabit program komutlarını takip etmek yerine, çıktı olarak ifade edilen kararları almak için, problem verilerini keşfeder ve girdi gözlemleri kümesinden veri temelli tahminleme yapar.

MVPO problemine uygulanan makine öğrenme algoritmaları Tablo 2.18'de gösterilmektedir.

**Tablo 2.18** OVPO problemine uygulanan makine öğrenmesi algoritmaları.

<b>Makine Öğrenmesi Algoritmaları</b>	<b>Geliştiren</b>
Hopfield Ağları (HA)	Hopfield (1984)
Yapay Bağışıklık Sistemi (YBS)	De Castro and Timmis (2002)
K-means Kümeleme	Lloyd (1982)

Fernandez ve Gomez (2007) OVPO problemine için Hopfield ağını (HA) uygulamış ve GA, BT ve YA (Chang ve diğ. 2000) ile yapılan performans karşılaştırması sonucunda HA'nın daha büyük örneklerde üstün performans gösterdiğini rapor etmişlerdir. Golmakani ve Alishah (2008) bir yapay bağışıklık sistemi algoritması önermişler ve genetik algoritma, tabu arama ve benzetme tavlama (Chang ve diğ. 2000) ve Hopfield ağı (Fernandez ve Gomez 2007) ile karşılaştırmışlardır. Yapay bağışıklık sistemi yaklaşımının diğerlerinden daha iyi sonuçlar üretebildiği fakat diğerleri kadar hızlı olmadığı belirtilmiştir. Suganya ve Vijayalakshmi Pai (2009), Hopfield ağını temel alan bir algoritma tasarlamış ve bu algoritmadaki ek kısıtlamaları aşmak için k-means kümeleme analizinden yararlanmışlardır. OVPO problemi için makine öğrenme algoritmaları uygulayan yayınlar Tablo 2.19'da özetlenmiştir.

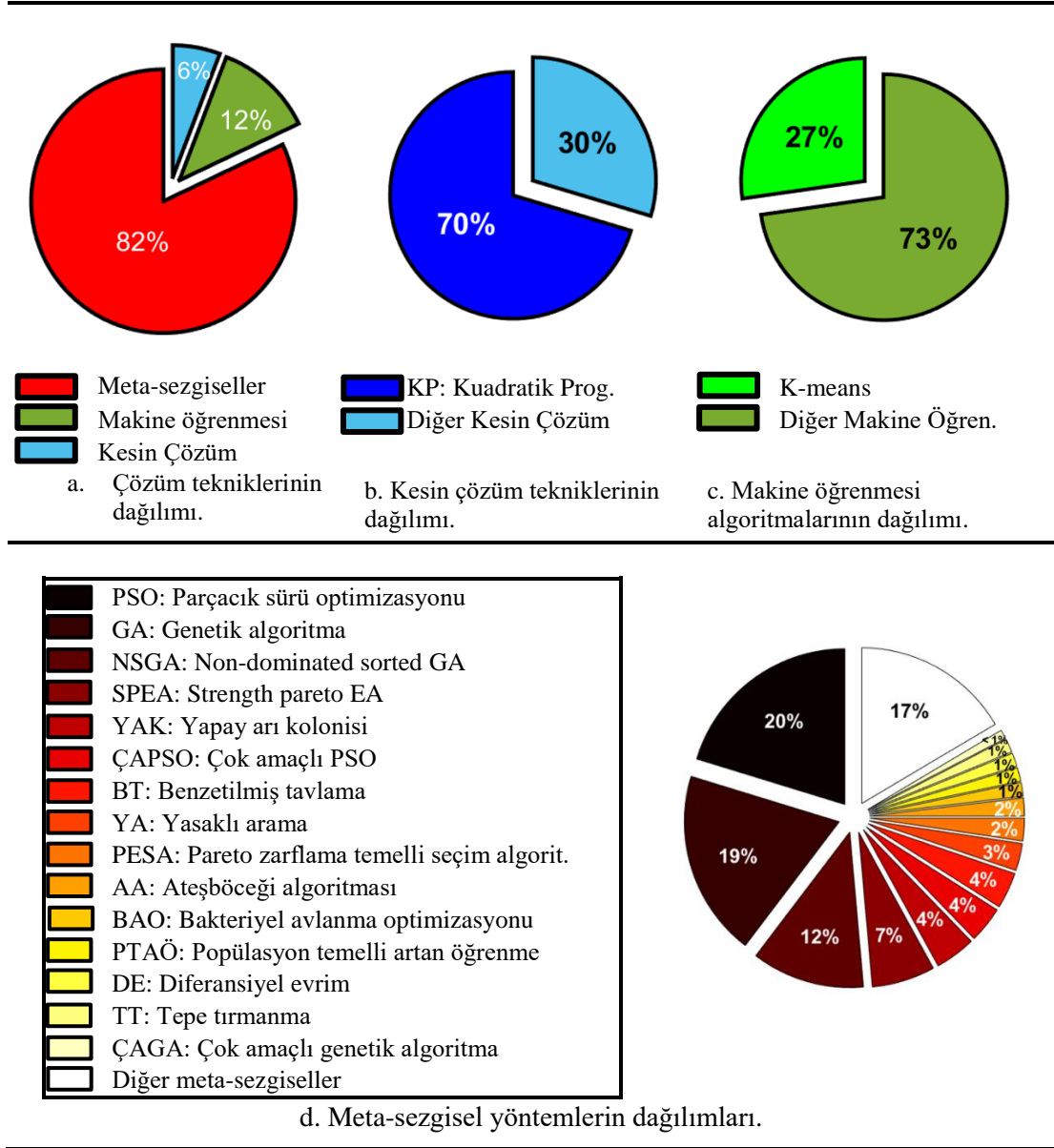
**Tablo 2.19:** OVPO için makine öğrenmesi algoritmasının kullanıldığı yayınlar.

<b>Makine Öğrenmesi Algoritmaları</b>	<b>Yayınlar</b>
<b>Sinir Ağları</b>	(Fernandez ve Gomez, 2007; Freitas ve diğ. 2009; Suganya ve Vijayalakshmi Pai, 2009)
<b>Yapay Bağışıklık Sistemi</b>	(Abbas ve Haider, 2009; Golmakani ve Alishah, 2008)
<b>K-means Kümeleme</b>	(Jiang ve diğ. 2014; Pai ve Michel, 2009; Suganya ve Vijayalakshmi Pai, 2009)

### **2.5.3 OVPO problemine uygulanan çözüm yaklaşımlarına genel bir bakış**

OVPO problemine uygulanan kesin ve kesin olmayan çözüm teknikleri, sırasıyla bölüm 3.1 ve bölüm 3.2'de açıklanmıştır. Bu çözüm teknikleri arasında meta-sezgisel yöntemlerin OVPO problemine için yaygın olarak kullanıldığı görülmektedir (hepsinin yaklaşık %82'sini meta-sezgiseller oluşturmaktadır). Bunun öncelikli nedenlerinden birisi de tam sayılı kısıtların modele eklenmesiyle birlikte problemin karmaşıklaşarak NP-Zor sınıfına girmesi olarak açıklanabilmektedir.

OVPO problemine uygulanan çözüm tekniklerinin literatürdeki dağılımı Şekil 2.11'de verilmiştir.



Şekil 2.11: OVPO problemine uygulanan çözüm tekniklerinin dağılımı.

Yukarıdaki grafiklerden de anlaşılacağı üzere, genetik algoritmalar ve parçacık sürü optimizasyonu gibi iyi bilinen meta-sezgisellerin, OVPO literatüründe birçok araştırmacı tarafından büyük ilgi görmüştür.

Bununla birlikte, GA ve PSO ile ilgili araştırmaların yeterli olgunluğa ulaşması ve bu alanlarda belki de çok sınırlı bir gelişme alanı kalması nedeniyle, son yıllarda araştırmacıların bu iki algoritma üzerine olan ilgisinin giderek azaldığı dikkat



çekmektedir. Buna karşın, nispeten daha yeni bir algoritma olan yapay arı kolonisi algoritması, diğer algoritmalar karşısında daha iyi performans sergilemesi nedeniyle ileride muhtemelen OVPO için kullanılan meta-sezgisel yöntemler arasında daha yüksek bir pay elde edebilecektir (Kalaycı ve diğ. 2017).

#### 2.5.4 Hibrit Çözüm Teknikleri

Hibritleme stratejisi, farklı algoritmaların güçlü özelliklerini bir araya getirerek daha iyi bir algoritma tasarlamak olarak tanımlanabilir. Araştırmacılar, OVPO için daha verimli algoritmalar tasarlamak amacıyla, bu stratejiyi, algoritmaların avantajlarından yararlanmak ve zayıf yanlarını elimine etmek için kullanmışlardır. OVPO problemini çözmek için hibrit çözüm stratejilerini kullanan yayınlar Tablo 2.20'deki gibi sınıflandırılmıştır.

**Tablo 2.20:** OVPO için hibrit çözüm yöntemlerinin kullanıldığı yayınlar.

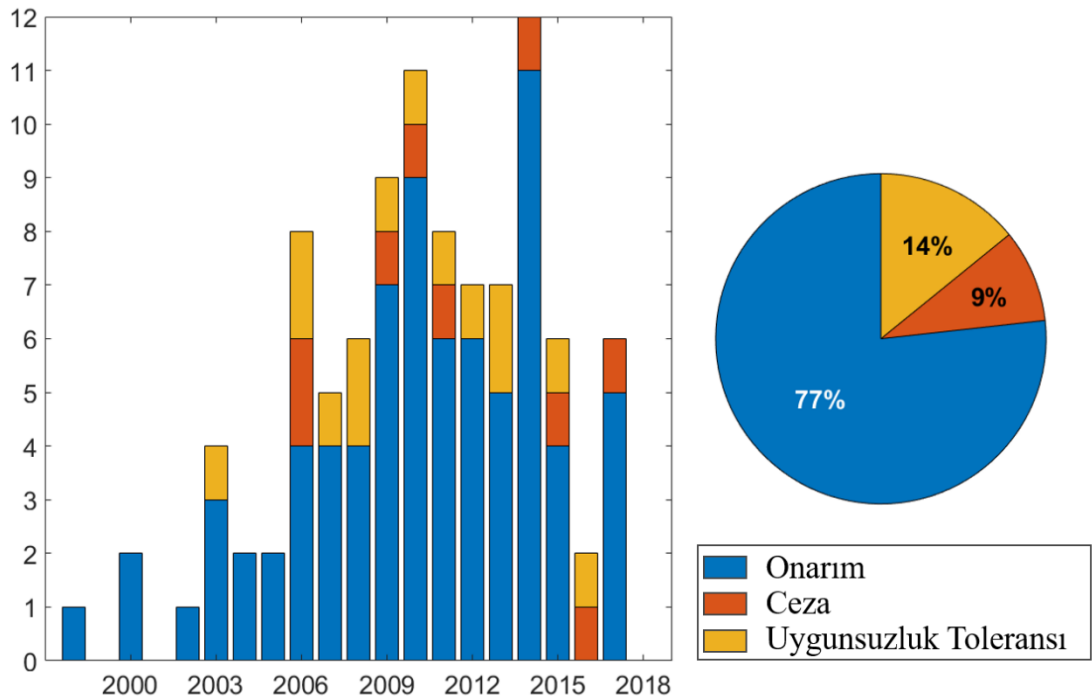
	Meta-Sezgiseller	Kesin Çözüm Yaklaşımları	Makine Öğrenmesi
<b>Meta-Sezgiseller</b>	(Lwin ve Qu, 2013; Pai ve Michel, 2009; Sadigh ve diğ. 2012; Tuba ve Bacanin, 2014 <sup>a</sup> ; Yaakob ve Watada, 2010)	(Baykasoglu ve diğ. 2015; Gaspero ve diğ. 2011; He ve Qu, 2016; Jin ve diğ. 2015; Moral-Escudero ve diğ. 2006; Ruiz-Torrubiano ve Suarez, 2015)	(Pai ve Michel, 2009; Sadigh ve diğ. 2012)
<b>Kesin Çözüm Yaklaşımları</b>		N/A	N/A
<b>Makine Öğrenmesi</b>			(Suganya ve Vijayalakshmi Pai, 2009)

Moral-Escudero ve diğ. (2006), evrimsel teknikleri, kuadratik programlamayla birleştiren bir hibrit çözüm stratejisi tasarlamışlardır. Gaspero ve diğ. (2011), efendi-köle (master-slave) yapısında bir hibrit çözüm yöntemi önermişler; efendi çözücü olarak yerel arama meta-sezgiseli kullanırken köle çözücü olarak kuadratik programlamadan yararlanmışlardır. Lwin ve Qu (2013), popülasyona dayalı artan öğrenme ve diferansiyel evrim algoritmasını bir araya getiren bir hibrit çözüm yaklaşımı geliştirmişlerdir. Arama uzayında aramanın etkinliğini artırabilmek için mutasyon ve elitizm stratejilerinden yararlanmışlardır. Tuba ve Bacanin (2014<sup>a</sup>), OVPO probleminin çözümü için yapay arı kolonisi ve ateşböceği algoritmalarından oluşan hibrit bir çözüm stratejisi geliştirmişlerdir. Böylelikle algoritmanın sömürü ve

keşif mekanizmalarının arasındaki optimum ödünleşmenin sağlanması amaçlanmıştır. Geliştirilen hibrit çözüm yaklaşımının performansını genetik algoritma, yasaklı arama ve benzetilmiş tavlama (Chang ve diğ. 2000) ve parçacık sürü optimizasyonu (Cura 2009) ile karşılaştırmışlardır. Ruiz-Torrubiano ve Suarez (2015), genetik algoritmayı ve KP'yi birleştiren hibrit bir çözüm yaklaşımı önermişlerdir. Baykasoglu ve diğ. (2015) önceden belirlenmiş sayıda varlığın ARUAP algoritmasıyla seçildiği ve seçilen varlıklara dağıtılacak sermaye oranlarının ise KP ile bulunduğu hibrit bir çözüm algoritması geliştirmişlerdir.

### 2.5.5 Kısıt işleme teknikleri

OVPO literatürü incelendiğinde, problem kısıtlarının yerine getirilmesi için farklı yöntemlerin uygulandığı görülmektedir. Bu doğrultuda, kısıt işleme tekniği olarak bir onarım prosedürü, ceza fonksiyonu veya uygunsuzluk toleransı gibi yöntemlerin sıklıkla kullanılmaktadır. Şekil 2.12, OVPO literatüründe kullanılan kısıt işleme tekniklerinin dağılımını göstermektedir.



Şekil 2.12: Kısıt işleme tekniklerinin dağılımı.

Modele dahilindeki kısıtların yerine getirilmesi için kullanılan en yaygın yöntemlerden birisi olurlu olmayan bir çözümü tamir prosedürü uygulanarak olurlu bölgeye sokma işlemidir. Birçok araştırmacı eleman sayısı ve limit kısıtlarını, Chang ve diğ. (2000) tarafından önerilen tamir prosedürünü uygulayarak işlemişlerdir. (Babaei ve diğ. 2015; Chen ve diğ. 2013; Chen ve diğ. 2017; Deng ve Lin 2010<sup>a</sup>, 2010<sup>b</sup>; Lwin ve Qu 2013; Lwin ve diğ. 2013; Mishra ve diğ. 2014<sup>a</sup>; Pai ve Michel 2009; Schaerf 2002; Vijayalakshmi Pai ve Michel 2013; Woodside-Oriakhi ve diğ. 2011). Crama ve Schyns (2003) her bir kısıt için hem portföylerin olurlu bölgede kalabilmesine zorlayan hem de olurlu olmayan portföylere ceza puanı uygulayan spesifik bir kısıt işleme prosedürü geliştirmişlerdir. Golmakani ve Fazel (2011) ve Corazza, ve diğ. (2013) ise kısıt işleme için ceza fonksiyonlarını kullanmışlardır. Bazı araştırmacılar, olurlu olmayan sonuçları düzeltebilmek için Deb (2000) tarafından önerilen seçim kurallarını uygulamışlardır. Bu seçim kuralları aşağıdaki gibi özetlenebilir: (i) Çözümlerden birisi olurlu ve diğeri olursuz ise, olurlu olan seçilir, (ii) her iki çözüm de olurlu değilse, olurluluğu en az ihlal eden çözüm seçilir, (iii) Her iki çözüm de olurlu ise; amaç fonksiyonu değeri daha iyi olan çözüm seçilir. Kalayci ve ark. (2017), kısıtlı OVPO modeli için önerdikleri yapay arı kolonisi algoritmasında, ikili (binary) kısıtı için uygunsuzluğa izin veren, eleman sayısı kısıtı için ise onarım prosedürü içeren bir kısıt işlem prosedürü geliştirmişlerdir. Yapılan karşılaştırmalı analiz sonucunda önerilen prosedürlerin Chang ve diğ. (2000) önerdiği onarım prosedüründen daha iyi sonuç verdiğini ifade etmişlerdir. Tablo 2.21, OVPO literatüründe kullanılan kısıt işleme tekniklerine göre çalışmaların sınıflandırmasını göstermektedir.

**Tablo 2.21:** OVPO için kullanılan farklı kısıt işleme tekniklerini bazında yayınların sınıflandırılması.

Kısıt İşleme Teknikleri	Yayınlar
Onarım Prosedürü	(Anagnostopoulos ve Mamanis, 2010, 2011 <sup>a</sup> ; Anagnostopoulos ve Mamanis, 2011 <sup>b</sup> ; Anagnostopoulos ve diğ. 2010; Bacanin ve Tuba, 2014, 2015; Bacanin ve diğ. 2014; Branke ve diğ. 2009; Chang ve diğ. 2000; Chang ve diğ. 2009; Chen ve diğ. 2013; Chen ve diğ. 2012; Chen ve diğ. 2017; Chiam ve diğ. 2007; Chiam ve diğ. 2008; ChiangLin, 2006; Coutino-Gomez ve diğ. 2003; Crama ve Schyns, 2003; Cui ve diğ. 2014; Cura, 2009; Deng ve Lin, 2010 <sup>a</sup> , 2010 <sup>b</sup> ; Deng ve diğ. 2012; Eftekharian ve diğ. 2017; Fernandez ve Gomez, 2007; Garcia ve diğ. 2012; Golmakani ve Alishah, 2008; Hao ve Liu, 2009; Huang ve Shen, 2010; Jiang ve diğ. 2014; Jiang ve diğ. 2008; Kalayci ve diğ. 2017; Kamili ve Riffi, 2015; Kao ve Cheng, 2013; Koshino ve diğ. 2007; Kumar ve Mishra, 2017; Kumar ve Bhattacharya, 2012; Lai ve diğ. 2006; Lean ve diğ. 2008; Liagkouras ve Metaxiotis, 2014; Lin ve diğ. 2005; Lwin ve Qu, 2013; Lwin ve diğ. 2014; Lwin ve diğ. 2013; Lwin ve diğ. 2017; Maringer ve Kellerer, 2003; Mishra ve diğ. 2014 <sup>a</sup> ; Moral-Escudero ve diğ. 2006; Mozafari ve diğ. 2011; Ong ve diğ. 2005; Pai ve Michel, 2009; Reid ve Malan, 2015; Ruiz-Torrubiano ve Suarez, 2010; Sabar ve Song, 2014; Sadigh ve diğ. 2012; Schaerf, 2002; Shoaf ve Foster, 1998; Skolpadungket ve diğ. 2007; Streichen ve Tanaka-Yamawaki, 2006; Streichert ve diğ. 2004 <sup>a</sup> ; Streichert ve diğ. 2004 <sup>b</sup> ; Talebi ve diğ. 2010; Tang ve diğ. 2009; Thomaidis, 2010; Tuba ve Bacanin, 2014 <sup>a</sup> , 2014 <sup>b</sup> ; Tuba ve diğ. 2013; Wang ve diğ. 2012; Woodside-Oriakhi ve diğ. 2011; Xia ve diğ. 2000; Xu ve diğ. 2011; Xu ve diğ. 2010; Yin ve diğ. 2015 <sup>a</sup> ; Yu ve diğ. 2009; Zhou ve Li, 2014; Zhu ve diğ. 2011)
Ceza Fonksiyonu	(Chen ve Cai, 2008; Chen ve diğ. 2006; Chen ve Zhang, 2010; Corazza ve diğ. 2012 <sup>a</sup> ; Corazza ve diğ. 2013; Crama ve Schyns, 2003; Golmakani ve Fazel, 2011; Lean ve diğ. 2008; Lu ve Wang, 2013; Moral-Escudero ve diğ. 2006; Pouya ve diğ. 2016; Reid ve Malan, 2015; Rong ve diğ. 2009; Xu ve diğ. 2007)
Uygunsuzluk Toleransı	(Anagnostopoulos ve Mamanis, 2010; Fasheng ve Wei, 2006; Kalayci ve diğ. 2017; Mishra ve diğ. 2016; Mishra ve diğ. 2014 <sup>b</sup> ; Reid ve Malan, 2015; Soleimani ve diğ. 2009; Wang ve diğ. 2011; Xu ve Chen, 2006)

### 3. YÖNTEM

Bu bölümde eleman sayısı kısıtlı portföy optimizasyonu problemine ait matematiksel modelle birlikte eleman sayısı kısıtının problemin karmaşıklığına olan katkısı, problemin çözümü için geliştirilmiş iki aşamalı hibrit çözüm yaklaşımı ve önemli algoritma bileşenleri, önerilen algoritmanın performans değerlendirmesi ve literatürdeki diğer çözüm yaklaşımları ile karşılaştırılmasında kullanılan performans ölçütleri ve veri setlerine yer verilmiştir.

#### 3.1 Eleman Sayısı Kısıtlı Portföy Optimizasyonu

Chang ve diğ. (2000) tarafından eleman sayısı kısıtlı portföy optimizasyonu problem için önerilen matematiksel model aşağıda sunulmuştur:

##### **Parametreler**

$N$	Mevcut varlık sayısı
$\mu_i$	$i$ . varlığın beklenen getirisi
$\sigma_{ij}$	$i$ . ve $j$ . varlıklar arasındaki kovaryans değeri
$\lambda$	Varyans ve getirinin ağırlık parametresi
$K$	Portföye tutulmak istenen varlık sayısı
$\varepsilon_i$	$i$ . varlığın minimum oranı
$\delta_i$	$i$ . varlığın maksimum oranı

##### **Değişkenler**

$w_i$	$i$ . varlığın portföydeki oranı
$z_i = \begin{cases} 1 \\ 0 \end{cases}$	$i$ . varlık portföye dahil edildiyse; aksi takdirde 0.

$$\text{Amaç Fonksiyonu: } \min \lambda \left[ \sum_{i=1}^N \sum_{j=1}^N w_i w_j \sigma_{ij} \right] - (1 - \lambda) \left[ \sum_{i=1}^N w_i \mu_i \right] \quad (3.1)$$

Kısıtlar:

$$\sum_{i=1}^N w_i = 1 \quad (3.2)$$

$$\sum_{i=1}^N z_i = K \quad (3.3)$$

$$\varepsilon_i z_i \leq w_i \leq \delta_i z_i \quad i = 1, \dots, N \quad (3.4)$$

$$z_i \in \{0,1\} \quad i = 1, \dots, N \quad (3.5)$$

$$0 \leq w_i \leq 1 \quad i = 1, \dots, N \quad (3.6)$$

$$0 \leq \varepsilon_i \leq \delta_i \leq 1 \quad i = 1, \dots, N \quad (3.7)$$

Denklem (3.1) de verilen amaç fonksiyonu çakışan iki amaç olan getiri maksimizasyonu ve risk minimizasyonunu içermektedir.  $\lambda$  ile ifade edilen parametre 0 ve 1 arasında alacağı değerle bu iki amaç arasındaki ödünleşme (tradeoff) belirlemektedir.  $\lambda = 0$  olduğunda, amaç fonksiyonu getirinin maksimizasyonunu ifade ederken,  $\lambda = 1$  olduğunda ise riskin minimizasyonuna dönüşmektedir. Denklem (3.2) portföye dahil edilecek varlıkların toplam oranının 1'e eşit olmasını sağlamaktadır. Eleman sayısı kısıtı olarak adlandırılan Denklem (3.3) ise portföye dahil edilecek varlık sayısına sınırlama getirmekte ve yalnızca  $K$  sayıda varlığın portföyde tutulmasını garanti etmektedir. Eleman sayısı kısıtının modele eklenmesiyle birlikte sürekli yapıdaki etkin sınır kesikli hale dönüşmektedir (Chang ve diğ. 2000). Bu dönüşüm, hesaplama karmaşıklığını (computational complexity) ve problemin zorluğunu artırmaktadır. Denklem (3.4) portföydeki varlık oranlarının önceden belirlenen minimum ve maksimum limitlerin arasında olmasını sağlamaktadır. Son olarak Denklem (3.5), (3.6) ve (3.7) karar değişkenlerini ve alabileceği değer aralıklarını göstermektedir.

### 3.2 Değişken Komşuluk Arama Algoritması ve Paralleleştirme Stratejileri

Değişken komşuluk arama (DKA), Mladenovic ve Hansen (1997) tarafından kombinatorik optimizasyon problemlerinin çözümü için geliştirilen tek çözüm (single solution) ve yerel arama (local search) tabanlı meta-sezgisel bir çözüm algoritmasıdır. Klasik yerel arama tabanlı algoritmaların aksine DKA, tek bir komşuluk yapısı yerine arama esnasında birden fazla komşuluk yapısını kullanmaktadır. Herhangi bir komşuluktaki yerel minimum nokta diğer komşuluklar için geçerli olmayabileceğinden, çoklu komşuluk yapısının kullanımı arama uzayının farklı

bölgelerindeki minimum noktalara erişim sağlanması açısından avantaj sağlamaktadır. Ayrıca, arama esnasında bu komşuluk yapıları sistematik olarak değiştirilmektedir. Böylelikle arama uzayında çeşitlilik (diversification) sağlanarak, çoğu yerel arama tabanlı sezgisellerin karşılaştığı yerel optimum noktalara takılma sorununun üstesinden gelinebilmektedir.

$G_k$ , ( $k = 1, \dots, k_{max}$ ) ve  $G_k(x)$  sırasıyla komşuluk yapıları kümesini ve  $x$  çözümünün  $k$ . komşuluktaki çözüm kümesini ifade etsin. DKA arama algoritmasının temel adımları Şekil 3.1’de gösterilmiştir.

---

**Algoritma:** Değişken Komşuluk Arama

**Başla**

Komşuluk yapılarını belirle  $G_k$ , ( $k = 1, \dots, k_{maks}$ )

Başlangıç çözümünü belirle

Durdurma koşulunu belirle

Durdurma koşulu gerçekleşinceye kadar aşağıdaki adımları tekrarla:

1.  $k \leftarrow 1$  olarak belirle
2.  $k = k_{maks}$  oluncaya kadar aşağıdaki adımları tekrarla
  - a. Çalkalama:  $x$  çözümünün  $k$ .komşuluğuna ait bir  $x'$  çözümü oluştur ( $x' : G_k(x)$ )
  - b. Yerel arama: Oluşturulan  $x'$  çözümüne yerel arama uygula ve bulunan yerel en iyi çözümü  $x''$  olarak belirle.
  - c. Eğer bulunan yerel en iyi çözüm mevcut çözümden daha iyiyse,  $x = x''$  yap ve aramaya başlangıç komşuluk yapısı ile devam et. Aksi takdirde, komşuluk yapısını bir arttır  $k \leftarrow k + 1$

**Bitir**

---

**Şekil 3.1:** Değişken komşuluk arama algoritması (Hansen ve Mladenovic 2001).

Diğer sezgisel algoritmalara kıyasla DKA, basit yapısı, farklı çözüm teknikleriyle entegre edilebilmesi ve çok az parametre gerektirmesi nedeniyle diğer algoritmalara göre önemli avantajlar sunmaktadır (Hansen ve Mladenovic 2001). Ayrıca, popülasyon tabanlı algoritmalarda olduğu gibi, uzun algoritma çalışma sürelerine ve nispeten büyük hafıza kaynaklarına ihtiyaç duymamaktadır. Bu özellikler DKA’yı lokasyon problemleri (Hansen ve diğ. 2001), kümeleme problemleri (Hansen ve Mladenović 2001), grafik(graph) problemleri (Canuto ve diğ. 2001), sırt çantası ve paketleme (knapsack and packing) problemleri (Puchinger ve diğ. 2006), zaman çizelgeleme (timetableing) problemleri (Côté ve diğ. 2005), çeşitli planlama (scheduling) problemleri (Hansen ve Mladenović 1999; Hansen ve diğ. 2006) ve araç rotalama problemleri (Kalayci ve Kaya 2016; Polat ve diğ. 2015<sup>a</sup>), montaj hattı dengeleme (Kalayci ve diğ. 2015, 2016; Polat ve diğ. 2015<sup>b</sup>) ve portföy optimizasyonu

problemi (Bačević ve diğ. 2019) gibi birçok kombinatorik ve global optimizasyon problemlerinin çözümü için başarılı bir şekilde uygulanmasına olanak sağlamaktadır.

Her ne kadar temel DKA (basic VNS) birçok farklı kombinatorik optimizasyon problemi için iyi performans gösterse de probleme özgü ortaya çıkan hesaplama karmaşıklıkları ve zorluklarının üstesinden gelmek ve algoritmanın performansını arttırmak için çeşitli varyantlar geliştirilmiştir (Hansen ve diğ. 2009). Ayrıca, bilgisayar donanım teknolojisinin gelişmesiyle birlikte, birden fazla işlemci kullanılmak suretiyle aynı anda çeşitli algoritma yapılarını eş zamanlı çalıştırmak mümkün olmuştur. Dolayısıyla algoritmanın çeşitli bileşenlerini mevcut işlemciler arasında dağıtarak algoritma çalışma zamanını azaltmak ve geliştirilen çözüm yaklaşımlarının arama (exploration) kabiliyetlerini arttırmak için paralelleştirme stratejileri geliştirilmiştir. Bu doğrultuda, García-López ve diğ. (2002) DKA algoritmasını temel alarak geliştirdikleri senkronize paralel DKA (synchronous parallel VNS) (yerel arama aşamasının paralelleştirilmesi), çoğaltılmış paralel DKA (replicated parallel VNS) (birden çok başlangıç stratejisini uygulayarak aynı anda birkaç DKA'yı eş zamanlı çalıştırmak) ve çoğaltılmış çalkalama DKA (replicated shaking VNS) (çalkalama ve yerel arama aşamaları köle işlemciler için kontrol edilirken, ana algoritma efendi (master) işlemci vasıtasıyla çalıştırılmaktadır). Crainic ve diğ. (2004) ise işbirlikçi komşuluk DKA (cooperative neighborhood VNS) (çeşitli arama kontrol mekanizmalarını, bilgi alışverişi yöntemlerini ve arama stratejilerini dikkate alan asenkron efendi-köle ilişkisi) stratejisini geliştirmişlerdir.

### 3.3 Önerilen Çözüm Yaklaşımı

Önerilen metodolojinin genel yapısını gösteren sözde kod Şekil 3.2'de sunulmuştur. Temel olarak, geliştirilen hibrit çözüm yaklaşımı iki bölümden oluşmaktadır. Bunlar sırasıyla, portföydeki varlıkların seçilmesi ve seçilen varlıkların ağırlıklarının belirlenmesidir. Bu iki aşamalı çözüm prosedürüne göre, portföye dahil edilecek varlıklar DKA algoritması kullanılarak belirlenirken, varlıkların ağırlıkları ise kuadratik programlama aracılığıyla hesaplanmaktadır.

İlk olarak algoritma başlangıç çözümünün ve varlık seçim havuzunun oluşturulması ile başlamaktadır. Sonrasında ise algoritmanın ana bölümü tekrarlayan



bir şekilde çalkalama (shaking), yerel arama ve KP prosedürlerini uygulayarak optimum portföy için arama gerçekleştirilir. Oluşturulan döngü yapısı maksimum iterasyon koşulu sağlanana kadar tekrar etmektedir. Eğer çalkalama prosedürü ve peşi sıra uygulanan kuadratik programla sonrası mevcut çözüm iyileştirilirse algoritma bulunan çözüme yerel arama uygulamadan çalkalama adımıyla devam eder. Aksi taktirde, çalkalama adımında bulunan çözümle yerel arama aşamasına geçilir. Eğer çalkalama ve yerel arama aşamalarının herhangi birinde mevcut çözüm iyileştirilirse, algoritma başlangıç komşuluk yapısı ile devam eder. Bunun aksine, çalkalama ve yerel arama aşamalarının her ikisinde de mevcut çözüm iyileştirilemezse, komşuluk yapısı bir artırılarak arama bir sonraki çalkalama prosedürü ile devam eder.

Önerilen yaklaşımın ana bileşenleri olarak; başlangıç çözümü oluşturma prosedürü, komşuluk seçimi, çalkalama ve yerel arama prosedürleri ilerleyen bölümlerde ayrıntılı olarak sunulmuştur.

---

```

1: Algoritma: DKA
2:  $W$ : işleme gelen çözüm
3:  $W_S$ : çalkalama ve KP sonrasında elde edilen çözüm
4:  $W_L$ : yerel arama ve KP sonrasında elde edilen çözüm
5:  $W_{GB}$ : global en iyi çözüm
6: Başla
7: Komşuluk yapılarını belirle  $G_k, k = 1, \dots, k_{maks}$ 
8: Bir başlangıç çözümü ( $W$ ) ve arama havuzu oluştur
9: Maksimum iterasyon sayısına ( $MaxIt$ ) ulaşana kadar aşağıdaki adımları TEKRARLA
10:    $k \leftarrow 1$  olarak belirle
12:    $k = k_{maks}$  oluncaya kadar aşağıdaki adımları TEKRARLA
13:     Çalkalama:  $W$ 'nin  $k$ . Komşuluğundan bir  $W_S$  çözümü oluştur( $W_S: G_k(W)$ )
14:     Kuadratik Programlama:  $W_S$  çözümüne ait varlık ağırlıklarını KP ile hesapla.
15:     eğer  $W_S$  çözümü  $W$ 'den daha iyi değil ise
16:       Yerel Arama:  $W_S$  çözümüne yerel arama uygula ve  $W_L$  çözümünü elde et ( $W_L: G_k(W_S)$ )
17:       Kuadratik Programlama:  $W_L$  çözümüne ait varlık ağırlıklarını KP ile hesapla.
18:     Bitir
19:     eğer yerel arama VEYA çalkalama adımlarında bir iyileşme kaydedildi ise
20:        $W \leftarrow W_S$  VEYA  $W \leftarrow W_L$ 
21:        $k \leftarrow 1$ 
22:     aksi halde
23:        $k \leftarrow k + 1$ 
24:     Bitir
25:   bitir
26:   eğer  $W_{GB}$  iyileştirildi ise
27:      $W_{GB} \leftarrow W$ 
28:   Bitir
29: Bitir
30: Bitir

```

---

**Şekil 3.2:** Önerilen çözüm yaklaşımına ait sözde kod.

### 3.3.1 Varlık Seçimi

Önerilen çözüm yaklaşımının varlık seçim adımı üç aşamadan oluşmaktadır: başlangıç çözümü ve arama havuzunun oluşturulması (*i*), çalkalama (*ii*) ve yerel arama (*iii*).

#### 3.3.1.1 Başlangıç çözümünün ve Arama havuzunun elde edilmesi

Başlangıç çözümü rastgele olarak üretilebileceği gibi (Chang ve diğ. 2000; Deng ve diğ. 2012; Kalayci ve diğ. 2017), basit sezgisel algoritmalar ya da çeşitli inşa yöntemleri kullanılarak da elde edilebilmektedir (Baykasoglu ve diğ. 2015; Cura 2009; Sadigh ve diğ. 2012). Fakat, kullanılacak olan başlangıç çözümü algoritmanın etkin sınıra yakınsama performansını ve optimale en yakın çözüme ulaşma süresini doğrudan etkilediği için, bu çalışma kapsamında rastgele stratejiden ziyade (Cura 2009) tarafından önerilen metodoloji kullanılarak algoritma için gerekli başlangıç çözümü ve varlık seçim havuzu oluşturulmuştur.

Bu metodolojiye göre, her bir varlık, getiri ve risk arasındaki ödünleşme parametresine ( $\lambda$ ) bağlı olarak ortalama getiri ve ortalama risk arasındaki oranı gösteren  $c$  değerlerine göre sıralanır. Bazı durumlarda, veri setleri ayrıca negatif getirili ve risk seviyesine sahip varlıkları da içerebilmektedir. Denklem (3.10) ve Denklem (3.11) bahsedilen bu negatif değerlerin yol açabileceği bazı hesaplama hatalarının engellenmesi amacıyla kullanılmaktadır.  $\Omega$  ve  $\psi$  değerlerinin de ilgili denklemler kullanılarak hesaplanması sonrasında, her bir varlık için  $c$  değerleri Denklem (3.12) kullanılarak hesaplanır ve son olarak, tüm varlıklar hesaplanan  $c$  değerlerine göre büyükten küçüğe doğru sıralanır. Sıralama işlemine müteakip, en yüksek  $c$  değerine sahip varlıklar en çok tercih edilen varlıkları temsil ederken, düşük  $c$  değerine sahip varlıklar nispeten daha az tercih edilebilir olarak sınıflandırılır.

$$\theta_i = 1 + (1 - \lambda)\mu_i \quad i = 1, \dots, N \quad (3.8)$$

$$\rho_i = 1 + \lambda \frac{\sum_{j=1}^N \sigma_{ij}}{N} \quad i = 1, \dots, N \quad (3.9)$$

$$\Omega = -1 \times \min(0, \theta_1, \dots, \theta_N) \quad (3.10)$$

$$\psi = -1 \times \min(0, \rho_1, \dots, \rho_N) \quad (3.11)$$

$$c_i = \frac{\theta_i + \Omega}{\rho_i + \psi} \quad i = 1, \dots, N \quad (3.12)$$

$a(i)$ , veri setindeki her bir varlığı temsil etsin. FTSE 100 veri setine ait örnek bir başlangıç çözümü ve arama havuzu Şekil 3.3'te gösterilmektedir.  $c$  değerlerine göre sıralanmış indeks listesindeki, portföye dahil edilmek istenen varlık sayısını ifade eden ilk  $K$  adet varlık KP ile ağırlıklarının bulunmasından sonra başlangıç çözümü olarak kabul edilirken, geriye kalan varlıklar kümesi daha sonraki çalkalama ve yerel arama aşamalarında kullanılacak olan varlık seçim havuzunu oluşturacaktır.

$i$	1	...	10	...	28	...	32	...	40	...	55	...	89
$a(i)$	18	...	9	...	30	...	48	...	80	...	49	...	45
$c(i)$	1,0039	...	1,0022	...	1,0015	...	1,0014	...	1,0012	...	1,0009	...	0,9993
	Başlangıç çözümü $K$												
	Havuz büyüklüğü $K + N/5$												
	Havuz büyüklüğü $K + N/4$												
	Havuz büyüklüğü $K + N/3$												
	Havuz büyüklüğü $K + N/2$												
	Havuz büyüklüğü $N$												

Şekil 3.3: FTSE 100 veri seti için örnek bir başlangıç çözümü ve havuz büyüklüğü gösterimi.

Bazı durumlarda, arama havuzunun belli bir oranda daraltılması arama performansını arttırmakta ve böylelikle optimal sonuca daha kısa sürede ulaşmaya yardımcı olabilmektedir. Bu sebeple, veri setindeki tüm varlıkları arama havuzuna dahil etmek yerine, arama havuzu belli oranda sınırlandırılmalıdır. Veri setindeki toplam varlık sayısı ( $N$ ) ve portföye dahil edilmesi istenen varlık sayısı ( $K = 10$ ) parametrelerine bağlı olarak belirlenmiş olan 5 seviye için örnek havuz büyüklükleri Şekil 3.3'te gösterilmiştir. Arama havuzunun daraltılması ve optimum havuz büyüklüğünün bulunması ile ilgili detaylar ilerleyen bölümlerde açıklanmaktadır.

Başlangıç çözümünün ve arama havuzunun oluşturulmasına müteakip, çalkalama ve yerel arama adımları sistematik bir şekilde uygulanır.

### 3.3.1.2 Çalkalama

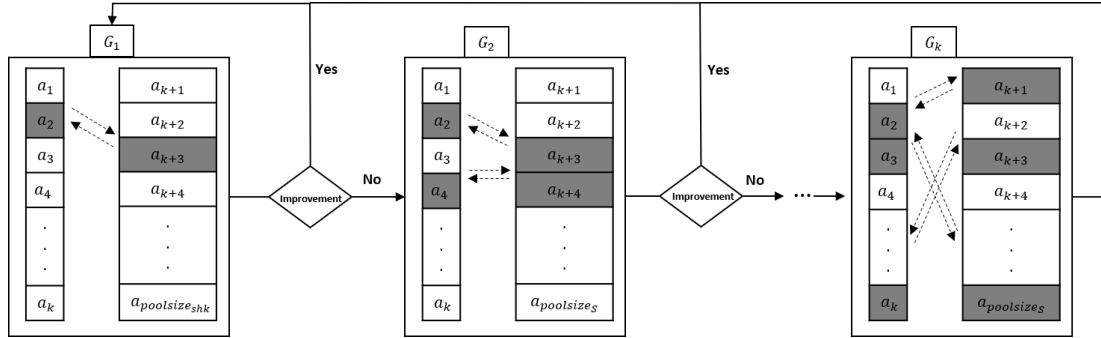
DKA algoritmasının en kritik kısımlarından birisi de arama performansı için hayat öneme sahip olan komşuluk yapılarının sistematik olarak değiştirmesi mekanizmasıdır. Algoritmanın hem keşif (exploration) hem de sömürü (exploitation) kabiliyetini göz önünde bulundurarak etkili bir arama prosedürü tasarlamak bu çözüm yaklaşımının en önemli amaçlarından birini oluşturmaktadır. Bu amaçlar doğrultusunda; çalkalama prosedürünün iki ana bileşeni olan: komşuluk yapılarının ve komşuluk değişim mekanizmalarının belirlenmesi çok önemlidir. Bir yandan, seçilen komşuluk yapısı ve değişim stratejisi, yerel optimum noktalara sıkışıp kalmamak için aramayı etkili bir şekilde çeşitlendirmeli, diğer yandan, mümkün olan en hızlı şekilde global optimumun bulunmasına olanak sağlamalıdır.

Geliştirilen bu çözüm algoritmasında, her komşuluk yapısı mevcut çözümde yer alan varlıkların kaç tanesinin mevcut çözümden kaldırılacağını ve kaç tane yeni varlığın çözüme ekleneceğini temsil etmektedir. Bu bilgi algoritmada, Şekil 3.4 ve Şekil 3.5' de gösterildiği üzere  $k = (1, \dots, k_{max})$  değişkeni ile çalkalama prosedürüne aktarılır ve çalkalama adımında bu bilgi çerçevesinde bazı varlıklar mevcut portföyden çıkartılır ve ilgili çözüm yeni varlıkların eklenmesiyle güncellenir. Başlangıçta, çalkalama aşaması yalnızca bir varlığın rastgele seçilmesi suretiyle kaldırılması / eklenmesiyle başlar ve kaldırılacak / eklenecek varlığın sayısı her bir adımda arttırılır.  $K$ , portföye dahil edilmek istenen varlık sayısını ifade ettiği üzere, mevcut çözümden kaldırılacak ve yerleştirilebilecek maksimum varlık sayısı  $K$  ( $k_{max} = K$ ) olacaktır. Her iterasyonda,  $W_S$  çözümü, çalkalama prosedürüne gelen  $W$  çözümünden  $k$  sayıda varlığın değiştirilmesi suretiyle üretilir. Başka bir deyişle,  $W_S$  çözümü  $G_k(W)$ 'nin komşuluğundadır denebilir.

Kullanılacak komşuluk yapılarının yanı sıra, komşuluk değişim mekanizmaları da aramanın çeşitlendirilmesinde büyük öneme sahiptir. Önceden belirlenmiş bir komşuluk kümesinde, rastgele, stokastik veya sıralı değişim gibi birçok farklı değişim stratejileri uygulanabilmektedir. Bu çalışma kapsamında komşuluk yapılarının sıralı olarak değiştirildiği bir komşuluk değişim stratejisi kullanılmıştır. Bu stratejiye göre, değiştirme işlemi ilk komşuluk ile başlamakta ve her iterasyonda komşuluk yapısı bir artırarak devam edilmektedir. Mevcut bir komşuluk yapısı ( $G_k$ ) dahilindeki herhangi

bir çalkalama adımında, eğer mevcut çözümde bir iyileştirme gerçekleştirilemezse, çalkalama işlemi bir sonraki komşuluk yapısı ( $G_{k+1}$ ) ile devam eder; aksi takdirde, bir sonraki iterasyonda uygulanacak prosedürler ilk komşuluk yapısı ( $G_1$ ) ile devam eder.

Sıralı komşuluk değişim mekanizması ve çalkalama prosedürünün yapısı Şekil 3.4 'teki akış şemasında ve Şekil 3.2 ile Şekil 3.5 'de verilen sözde kodlarda gösterilmiştir.



Şekil 3.4: Sıralı komşuluk değişim mekanizması.

- 
- 1: **Prosedür:** Çalkalama
  - 2: **Girdi:**  $W, poolsize_S$
  - 3: **Çıktı:**  $W$
  - 4:  $W$ : prosedüre gelen çözüm
  - 5:  $poolsize_S$ : çalkalama aşaması için kullanılacak varlık seçim havuzu büyüklüğü
  - 6:  $P$ : portföydeki varlıklar kümesi
  - 7:  $A$ :  $poolsize_S$  parametresi ile sınırlandırılmış varlık seçim havuzu içerisindeki varlıklar kümesi
  - 8:  $k$ : çıkartılacak/eklenecek varlık sayısı ( $k \leq K$ )
  - 9: **Başla**
  - 10:  $REMOVE \leftarrow P$  kümesinden rastgele  $k$  adet varlık seç /\*çözümde kaldırılacak varlıklar\*/
  - 11:  $ADD \leftarrow A \setminus P$  kümesinden rastgele  $k$  adet varlık seç /\*çözümde eklenecek varlıklar\*/
  - 12:  $W$  çözümünden  $REMOVE$  içerisindeki varlıkları  $ADD$  ile değiştir
  - 13: **Bitir**
- 

Şekil 3.5: Çalkalama aşamasına ait sözde kod.

### 3.3.1.3 Yerel Arama

Önerilen çözüm yaklaşımında, yerel arama adımı için, algoritmanın tüm olası hareketler yapıncaya kadar çalışmasına izin veren en iyi gelişme (best improvement) stratejisi kullanılmıştır. Her ne kadar en iyi gelişme stratejisinin ilk iyileşme (first improvement) stratejisine göre çok daha fazla zaman alan bir yaklaşım olduğu bilinse de, rastgele başlangıç çözümü yerine, iyi bir başlangıç çözümü inşa stratejisi ya da

sezgisel algoritma aracılığıyla üretilmiş iyi başlangıç çözümlerinde en iyi gelişme stratejisinin algoritmayı daha iyi çözüme daha kısa sürede ulaştırdığı tespit edilmiştir (Hansen ve Mladenović 2006).

Bu aşamada, mevcut portföydeki her bir varlık teker teker portföyden çıkartılıp yerine arama havuzundan sırasıyla seçilen varlıklar yerleştirilir. Önerilen algoritma için kullanılan yerel arama prosedürünün detayları Şekil 3.6' daki sözde kodla gösterilmiştir.

---

```

1  Prosedür: Yerel arama
2:  Girdi:  $W_S, K$ 
3:  Çıktı:  $W_L$ 
4:   $W_S$ : prosedüre gelen çözüm
5:   $W_L$ : prosedürden çıkan çözüm
6:   $W_B$ : yerel en iyi çözüm
8:   $K$ : portföyde tutulmak istenen varlık sayısı
10:  $poolsize_L$ : yerel arama aşamasında kullanılan varlık seçim havuzu büyüklüğü
11:  $P$ : portföydeki varlıklar kümesi
12:  $A$ :  $Poolsize_L$  parametresi ile sınırlandırılmış varlık seçim havuzu içerisindeki varlıklar kümesi
13: Başla
14:  $W_L \leftarrow W_S$  ve  $W_B \leftarrow W_S$ 
15:  $i = 0$ 
16: for  $i = 1$  to  $K$ 
17:     for  $j = 1$  to  $poolsize_L$ 
20:          $W_L \leftarrow W_S$ 'nin  $i$ . varlığını  $A \setminus P$  kümesinin  $j$ . varlığı ile değiştir
23:          $W_L$  çözümüne ait varlık ağırlıklarını KP ile hesapla
24:         eğer  $W_B$  iyileştirildi ise
25:              $W_B \leftarrow W_L$ 
27:              $i = 1$ 
28:         aksi halde
29:              $i = i + 1$ 
30:         bitir
32:     bitir
33: Bitir

```

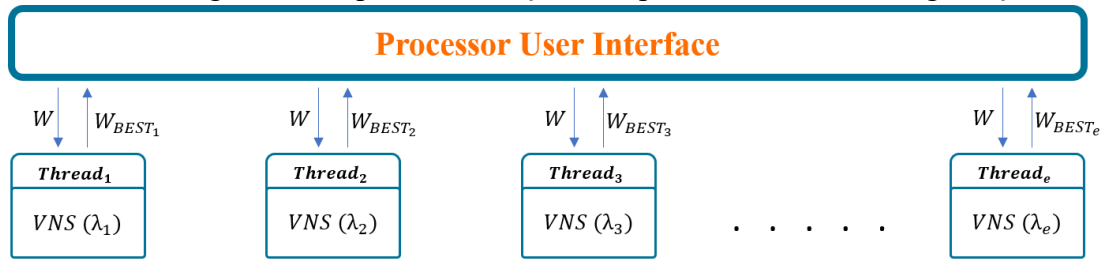
---

Şekil 3.6: Yerel arama sözde kodu.

### 3.3.1.4 Paralleleştirme

Bu çalışmada, hesaplama süresini kısaltmak amacıyla basit bir paralelleştirme stratejisi uygulanmıştır. Crainic ve diğ. (1997) tarafından önerilen üç boyutlu taksonomiye dayanarak uygulanan strateji  $pC/C/MPSS$  olarak sınıflandırılmaktadır. Bu sınıflandırmaya göre birinci boyut arama kontrol kardinalitesini temsil ederken, ikinci ve üçüncü boyutlar sırasıyla iletişim kontrolü ve arama stratejisini

tanımlamaktadır. Şekil 3.7 'de görüldüğü gibi, arama kontrolü, bir efendi-köle yapısı kullanmak yerine, algoritmayı bağımsız ve asenkron olarak her bir  $\lambda$  değeri için farklı işlemcilerde çalıştırmaktadır. İşlemciler arasında iletişim ve bilgi aktarımı gerçekleştirilmemektedir. Bunun yanı sıra, işlemcilerde çalıştırılan her bir DKA algoritmasında aynı arama stratejisi uygulanmaktadır. Ayrıca, farklı  $\lambda$ 'nın farklı değerleri için her bir işlemcide dikkate alınan amaç fonksiyonları farklı sonuçlar vereceğinden, her bir işlemcideki çalışan algoritmalar için farklı başlangıç çözümleri kullanılmış olacaktır. Sonuç olarak, işlemcinin her bir çekirdeğinden aynı anda farklı  $\lambda$  değerlerinin optimum sonuçlarını toplamak mümkün hale gelmiştir.



Şekil 3.7: Değişken komşuluk arama algoritması için paralelleştirme stratejisi temsili gösterimi.

Sıralı DKA algoritmasının en çok zaman alan bileşenlerinden biri olan yerel arama aşaması dikkate alınarak performans iyileştirmesi amacıyla ya da arama uzayının farklı noktalarında eş zamanlı arama yaparak algoritmanın keşif ve sömürü kabiliyetlerini arttırmaya yönelik olarak farklı paralelleştirme stratejilerinin geliştirilmesi her zaman mümkündür. Bu doğrultuda çeşitli meta-sezgisel yöntemler için geliştirilmiş paralelleştirme stratejileri ve detaylı incelemeleri için bkz. Crainic ve Hail (2005)

### 3.3.1.5 Varlık Oranlarının Belirlenmesi

Optimum portföyde yer alacak varlıkların seçilmesinden sonra, eleman sayısı kısıtı elimine edilmiş ve probleme ait model basit bir kuadratik programlama modeline dönüştürülmüş olur. Dolayısıyla eleman sayısı kısıtının kaldırılmasıyla sorunun karmaşıklığı önemli ölçüde azalmıştır. Ayrıca, değerlendirilecek varlık sayısı  $N$ 'den  $K$ 'ya düştüğünden, arama uzayı da ciddi ölçüde daraltılmıştır. Bu aşamadan sonra varlıkların ağırlıkları basit bir sezgisel algoritma kullanılarak hesaplanabilir. Bununla birlikte, problem kuadratik programlama modeline indirildiği optimum çözümlere ulaşmak için sezgisel bir algoritma kullanmak yerine, kesin çözüm teknikleri de

uygulanabilmektedir. Bu nedenle, ikinci aşamada kesin optimal sonuçları elde etmek için KP kullanılmıştır (Baykasoglu ve diğ. 2015). Bu doğrultuda ise MATLAB kuadratik programlama çözücüsü (MATLAB R2019a, Optimization Toolbox, Version 8.3) kullanılmıştır. DKA algoritmasının da aynı ortamda geliştirilmesi dolayısıyla ayrıca bir entegrasyona gereklilik duyulmamıştır.

### **3.4 Hesaplamalı Sonuçlar**

#### **3.4.1 Uygulama**

Önerilen çözüm yaklaşımı MATLAB R2019a kullanılarak modellenmiş ve test edilmiştir. Parametre testleri için yapılan istatistiksel analizler için ise MINITAB 18'den yararlanılmıştır. Deneysel çalışmalar 32 GB RAM ve Intel Xeon E5-2650 2.0 GHz işlemci bulunan bir bilgisayar ile gerçekleştirilmiştir.

#### **3.4.2 Test Problemleri**

Önerilen algoritmanın performans analizi ve literatürdeki diğer çözüm yaklaşımlarıyla ((Chang ve diğ. 2000), (Cura 2009), (Deng ve diğ. 2012), (Lwin ve Qu 2013), (Baykasoglu ve diğ. 2015), (Kalayci ve diğ. 2017)) karşılaştırma yapılabilmesi için OVPO literatüründe yaygın olarak kullanılan beş veri setinden (Beasley 1990) yararlanılmıştır. Kullanılan veri setleri sırasıyla Hang Seng, DAX 100, FTSE 100, S&P 100, ve Nikkei 225 endekslerinin Mart 1992 ve Eylül 1997 yılları arasındaki haftalık fiyatları içermektedir.

#### **3.4.3 Parametre Ayarlama**

En iyi algoritma performansının elde edilebilmesi için önerilen çözüm yaklaşımı için en uygun parametre değerlerinin belirlenmesi gerekmektedir. Bu sebeple, kullanılan her bir veri seti bazında parametre testleri gerçekleştirilmiştir. Gerçekleştirilen parametre testleri şu kabuller çerçevesinde yapılmıştır: Portföye dahil



edilecek varlık sayısı ( $K = 10$ ),  $\varepsilon_i = 0.01$  ( $i = 1, \dots, N$ ) ve toplam iterasyon sayısı = ( $N \times 1000$ ). Toplam tekrar sayısı = 5.

**Tablo 3.1:** Önerilen çözüm yaklaşımı için parametre seviyeleri.

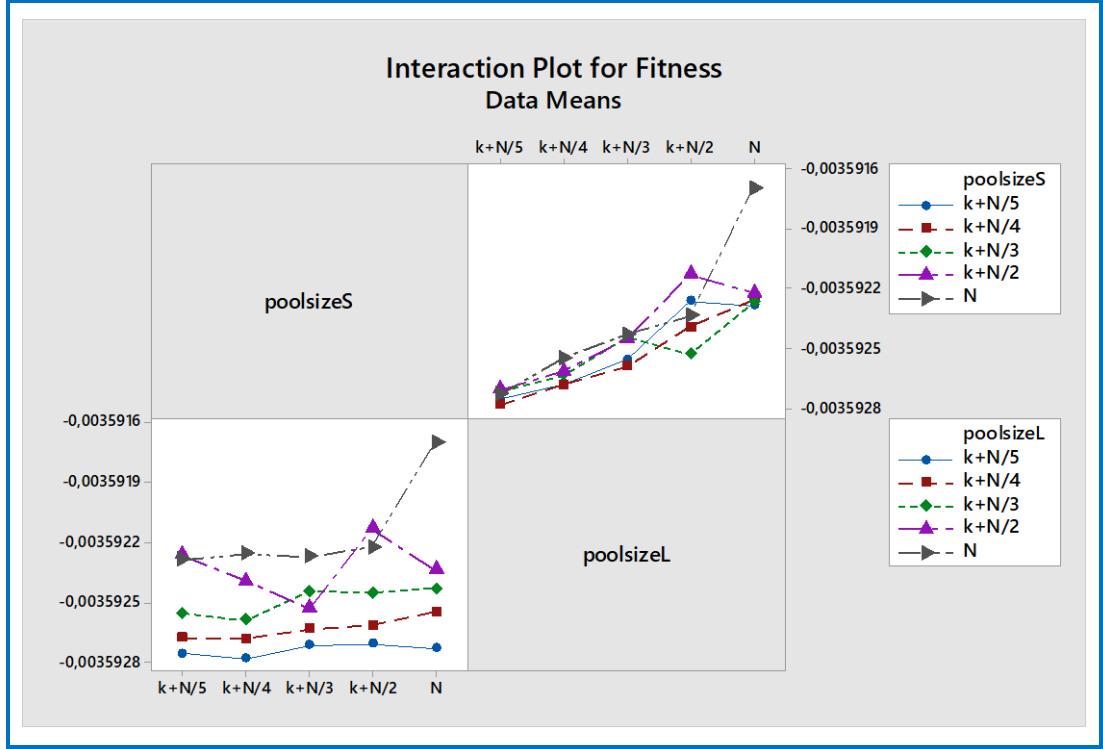
Parametre	Tanım	Test edilen seviye				
$\Delta$	İyileştirme sınırı	Sabit: $1e - 16$				
$MaxIt$	Maximum iterasyon sayısı	Sabit: $N \times 1000$				
$poolsize_S$	Çalkalama için havuz büyüklüğü	$K + N/5$	$K + N/4$	$K + N/3$	$K + N/2$	$N$
$poolsize_L$	Yerel arama için havuz büyüklüğü	$K + N/5$	$K + N/4$	$K + N/3$	$K + N/2$	$N$

Arama performansını ve algoritmanın hızını etkileyebilecek en kritik parametrelerden biri, portföyde yer alacak varlıkların seçildiği havuzun büyüklüğüdür. Büyük bir arama havuzu kullanmak, aramadaki çeşitliliğin artması sebebiyle iyi olsa da algoritma, optimum çözüme ulaşmak için fazladan zaman kaybedebilir. Öte yandan, arama havuzunu daraltmak, yerel optimum noktalarda takılı kalma sorununu ortaya çıkarırken, algoritmanın sömürü yeteneğini dolayısıyla algoritma performansını ve algoritma hızını artırabilir. Bu nedenle, geliştirilen algoritma için en uygun havuz boyutunun belirlenmesi büyük önem arz etmektedir. Bununla birlikte, çalkalama ve yerel arama aşamalarında farklı varlık seçim mekanizmaları uygulandığı için, bu aşamalarda kullanılacak optimum havuz büyüklükleri de farklı olabilir. Dolayısıyla, havuz büyüklüğü parametresi hem çalkalama hem de yerel arama adımları için ayrı olarak belirlenmiştir. Bu doğrultuda, her parametre için, ilgili veri setindeki varlık sayısına ( $N$ ) ve portföyde bulundurulması istenen varlık sayısına  $K$  bağlı olarak 5 farklı parametre seviyesi belirlenmiştir. FTSE 100 veri kümesi için 5 farklı parametre seviyesine göre örnek bir havuz boyutu gösterimi Tablo 3.2'de sunulmaktadır.

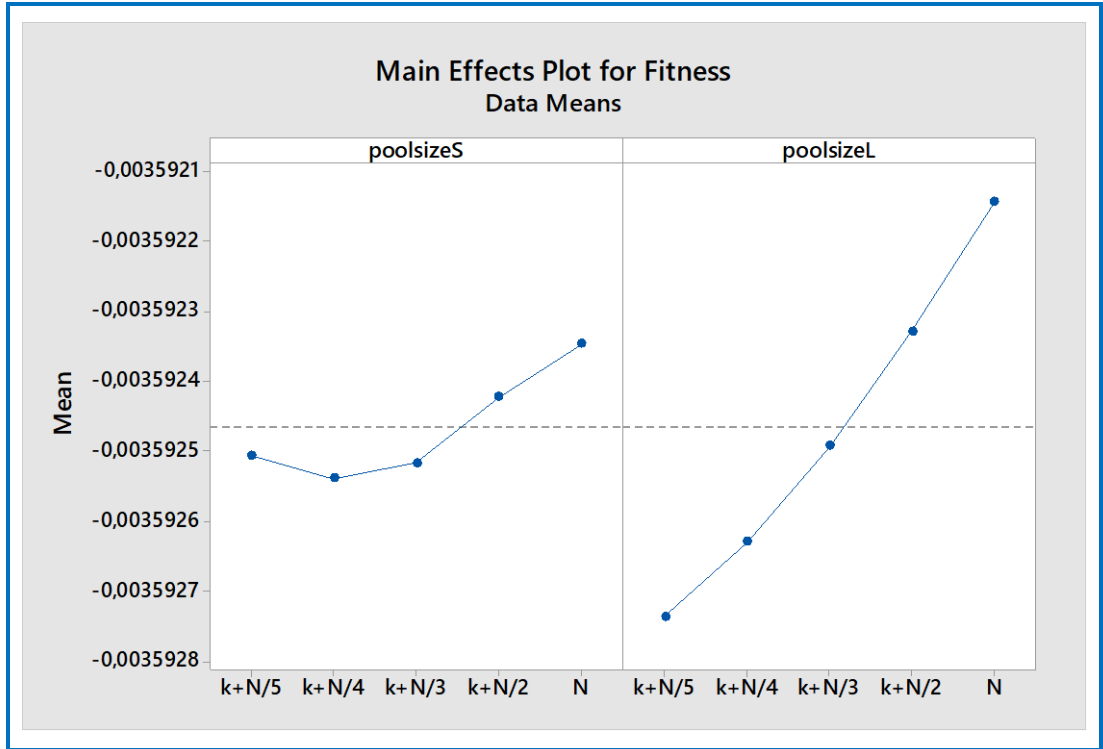
**Tablo 3.2:** FTSE 100 veri seti için temsili bir havuz büyüklüğü gösterimi.

Parametre	Tanım	Test seviyeleri				
		$K + N/5$	$K + N/4$	$K + N/3$	$K + N/2$	$N$
$poolsize_S$	Çalkalama için havuz büyüklüğü	28	32	40	55	89
$poolsize_L$	Yerel arama için havuz büyüklüğü	28	32	40	55	89

Belirlenen seviyelere göre çalkalama aşamasında kullanılan havuz büyüklüğü ile yerel arama aşamasında kullanılan havuz büyüklüğü arasındaki etkileşim analiz edilerek optimal havuz büyüklüğü araştırıldı. Elde edilen test sonuçları Şekil 3.8 ve Şekil 3.9'de gösterilmektedir.



Şekil 3.8: Çalkalama ve yerel arama aşamaları için havuz büyüklüklerine ait etkileşim grafiği.



Şekil 3.9: Çalkalama ve yerel arama aşamaları için havuz büyüklüklerine ait ana faktör etkileri (main effect) grafiği.

Şekil 3.8 Parametreler arası etkileşim grafiğini gösterirken, Şekil 3.9 ise optimal çözümlerin ortalamasına bağlı olarak çalkalama ve yerel aramalar için kullanılan havuz büyüklüklerine ait ana faktör etkileri grafiğini göstermektedir. Yapılan istatistiksel test sonuçlarına göre yerel arama adımında daha dar havuz kullanılması

daha iyi sonuçlar getirdiği açıkça görülmektedir. Diğer yandan, çalkalama aşamasında kullanılacak olan havuz büyüklüğünün ortalama sonuçları yerel arama aşamasındaki kadar etkilemediği görülse de özellikle ilk üç test seviyesinde büyük havuz seçiminin sonuçları bir miktar kötüleştirdiği göze çarpmaktadır (Bkz. Şekil 3.8)

### 3.4.4 Performans Ölçütleri

Önerilen çözüm yaklaşımının performansını literatürdeki diğer algoritmalarla karşılaştırmak için, Chang ve diğ. (2000) tarafından önerilen yüzde hatasının ortalaması (YHO), yüzde hatasının medyanı (YHMED) minimum yüzde hatası (YHMIN) ve maksimum yüzde hatası (YHMAKS) performans ölçütleri ve Cura (2009) tarafından önerilen ortalama öklid uzaklığı (OÖU), ortalama getiri hatası (OGH) ve getiri hatasının varyansı (GHV) performans ölçütleri kullanılmıştır.

$(x_i, y_i)$  kısıtsız etkin sınır üzerindeki her bir ayrık noktayı ifade ederken,  $(x^*, y^*)$  önerilen çözüm algoritması tarafından üretilen ve sırasıyla varyans ve getiri değerlerini içeren her bir optimal çözümü ifade etsin.

İlk olarak, geliştirilen algoritma tarafından üretilen her bir çözüm noktasının teorik etkin sınır üzerindeki yatay ve dikey izdüşümleri olan  $x^{**}$  ve  $y^{**}$  noktaları aşağıdaki formülasyonlarda gösterildiği gibi hesaplanmaktadır. Hesaplama ara değerlerin bulunması için basit interpolasyon yöntemi kullanılabilir.

$$x^{**} = x_k + (x_j - x_k) \left[ \frac{(y^* - y_k)}{(y_j - y_k)} \right] \quad (3.13)$$

$$y^{**} = y_k + (y_j - y_k) \left[ \frac{(x^* - x_k)}{(x_j - x_k)} \right] \quad (3.14)$$

Burada  $x'_k$  ve  $x'_j$  sırasıyla  $y_k = \max[y_i | y_i \leq y^*]$  ve  $y_j = \min[y_i | y_i \geq y^*]$  ile ilgili olarak varyans değerlerini ve  $y'_k$  ve  $y'_j$  ise  $x_k = \max[x_i | x_i \leq x^*]$  and  $x_j = \min[x_i | x_i \geq x^*]$  ile ilgili olarak getiri değerlerini temsil etmektedir. Bu değerler doğrultusunda  $x^{**}$  ve  $y^{**}$  değerlerinin hesaplanmasının ardından, her iki doğrultu için de yüzde sapma değerleri hesaplanmıştır:

$$\varphi_j = 100 \left| \frac{y^* - y^{**}}{y^{**}} \right| \quad (3.15)$$

$$\omega_j = 100 \left| \frac{x^* - x^{**}}{x^{**}} \right| \quad (3.16)$$

Denklem (3.15) getirinin sapma yüzdesini ifade ederken, Denklem (3.16) ise varyansın sapma yüzdesini göstermektedir.

$$MEAPE = \frac{\sum_j^E \min(\varphi_i, \psi_i)}{E} \quad (3.17)$$

$$\text{Where } \psi_i = 100 \left| \frac{\sqrt{X^{**}} - \sqrt{X^*}}{X^{**}} \right| \quad (3.18)$$

$$MEDPE = \min(\text{median}\{\varphi_1, \varphi_2, \dots, \varphi_E\}, \text{median}\{\psi_1, \psi_2, \dots, \psi_E\}) \quad (3.19)$$

$$MINPE = \min(\min\{\varphi_1, \varphi_2, \dots, \varphi_E\}, \min\{\psi_1, \psi_2, \dots, \psi_E\}) \quad (3.20)$$

$$MAXPE = \max(\max\{\varphi_1, \varphi_2, \dots, \varphi_E\}, \max\{\psi_1, \psi_2, \dots, \psi_E\}) \quad (3.21)$$

$$MEUCD = \frac{\sum_{j=1}^E \sqrt{(x^{***} - x^*) + (y^{***} - y^*)}}{E} \quad (3.22)$$

$$VRE = \frac{\sum_{j=1}^E \frac{100|(x^{***} - x^*)|}{x^*}}{E} \quad (3.23)$$

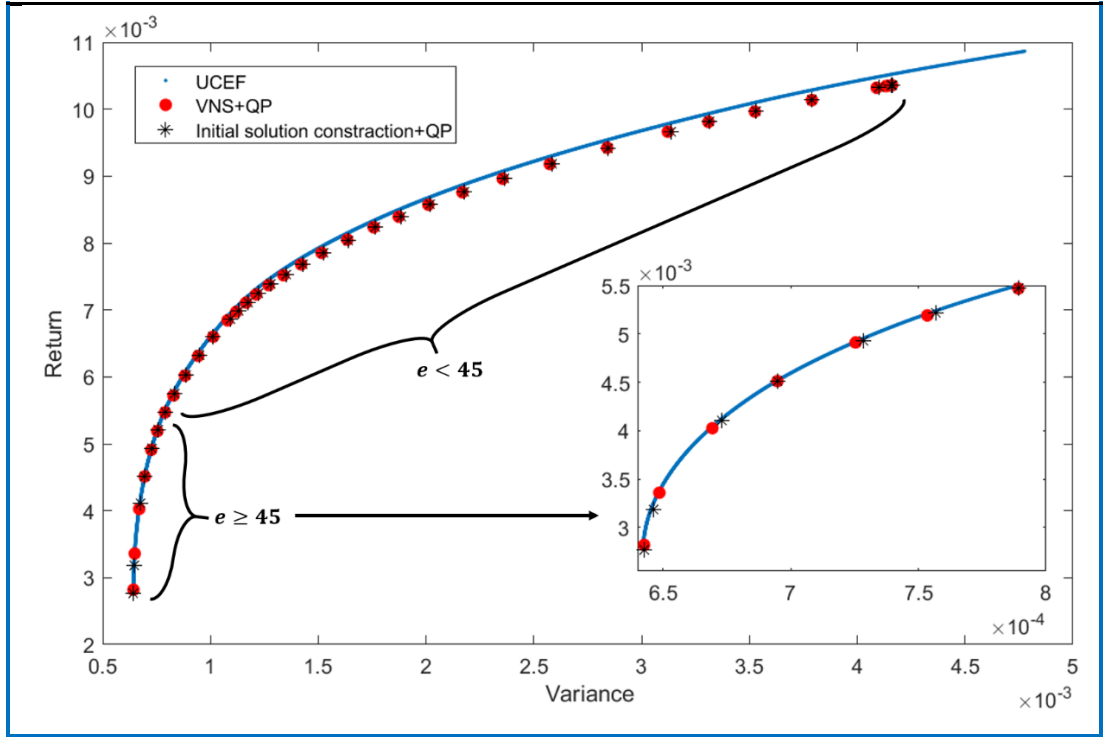
$$MRE = \frac{\sum_{j=1}^E \frac{100|(y^{***} - y^*)|}{y^*}}{E} \quad (3.24)$$

### 3.4.5 Hesaplamalı Sonuçlar

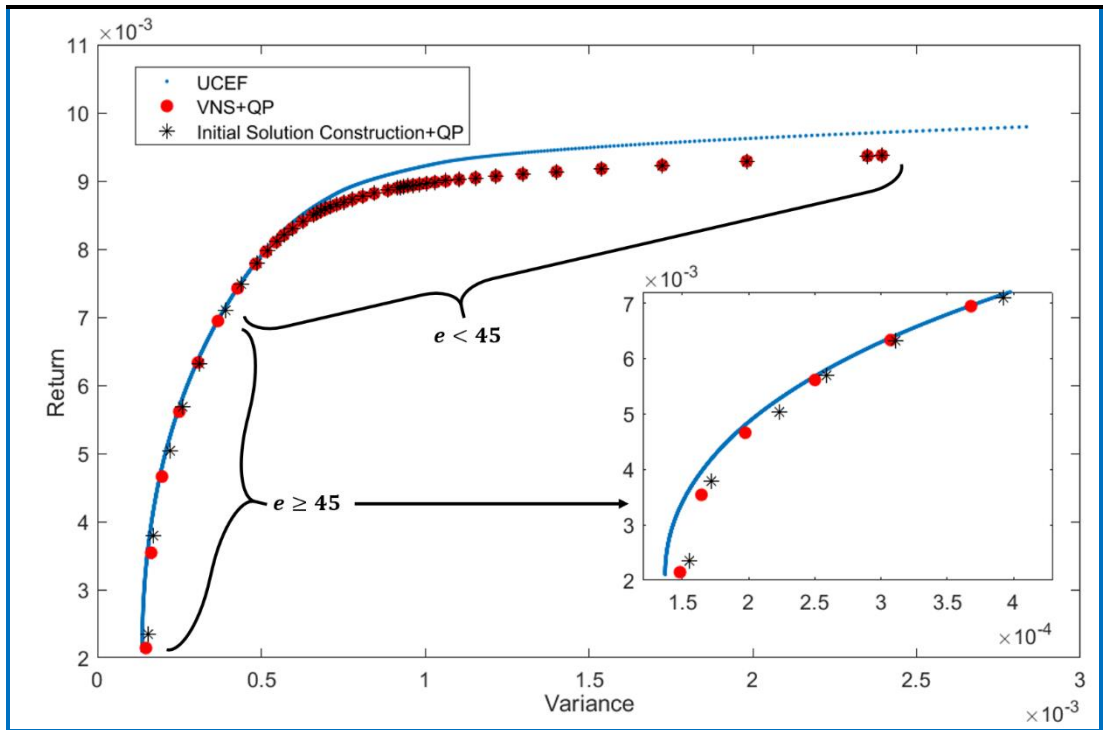
Bu bölümde, önerilen çözüm algoritmasının literatürdeki beş farklı veri seti üzerinde test edilmesine müteakip elde edilen sonuçlar, literatürdeki diğer çözüm yaklaşımları ile karşılaştırılmalı olarak sunulmuştur. Yapılan bütün testler  $K = 10$ ,  $\varepsilon_i = 0.01$  ( $i = 1, \dots, N$ ) ve  $\delta_i = 1$  ( $i = 1, \dots, N$ ) kabulleri üzerinde gerçekleştirilmiştir.

$e = (1, \dots, 50)$  her bir  $\lambda$  değerine bağlı olarak üretilen etkin sınır üzerindeki çözüm noktalarını temsil etsin. İlk deneysel test önerilen çözüm yaklaşımının düşük lamda değerleri  $\lambda(e < 45)$  ve yüksek lamda değerlerinde  $\lambda(e \geq 45)$  performansının değerlendirilmesi için gerçekleştirilmiştir. Amaç fonksiyonunun lineer/non-lineer yapısı mevcut lamda değerine göre değişmektedir. Örneğin,  $\lambda = 0$  iken, amaç fonksiyonu neredeyse lineer olmaktadır,  $\lambda$  değerinin yükselmesi amaç fonksiyonunun da non-lineerliğini arttırmaktadır. Dolayısıyla önerilen çözüm yaklaşımı ile üretilen çözümlerin etkin sınıra yakınsaklığı  $\lambda$ 'nın farklı değerleri için değişkenlik gösterebilmektedir. Bu sebeple algoritma performansı  $\lambda$ 'nın hem düşük hem de yüksek değerlerine göre test edilmesi daha isabetli olacaktır.

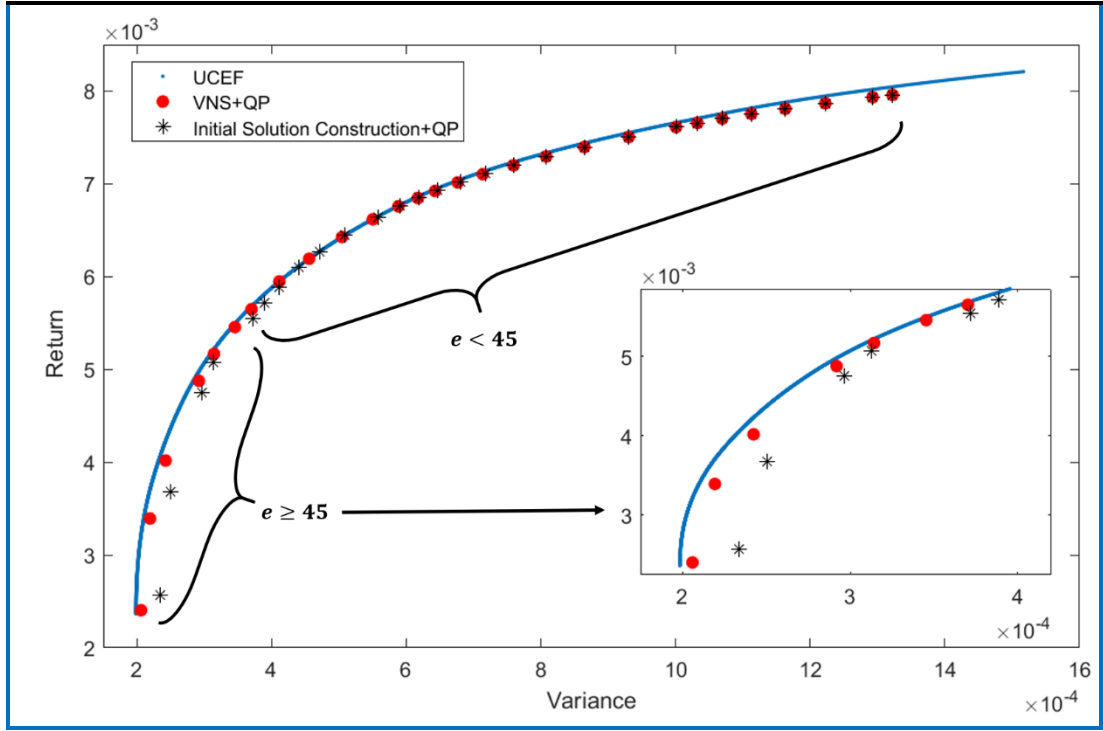
Şekil 3.10, 3.11, 3.12, 3.13 ve 3.14 sırasıyla XU030, Hang Seng, DAX 100, FTSE 100, S&P 100 ve NIKKEI veri setleri için başlangıç çözümü oluşturma prosedürü ve KP dan oluşan çözüm stratejisi (INIQuad) ve önerilen çözüm algoritması (DKAQuad) tarafından oluşturulan etkin sınırları göstermektedir. Genel olarak değerlendirildiğinde, DKAQuad çözüm yaklaşımının teorik etkin sınıra INIQuad'dan daha iyi bir yakınsama performansı sergilediği açıkça görülmektedir. Ayrıca  $\lambda$ 'nın daha düşük değerlerinde  $\lambda(e < 45)$  etkin sınırlar arası mesafeler daha yakınken, özellikle geniş veri setlerinde ve  $\lambda$ 'nın daha büyük değerlerinde  $\lambda(e \geq 45)$  bu farkın daha belirgin olduğu göze çarpmaktadır.



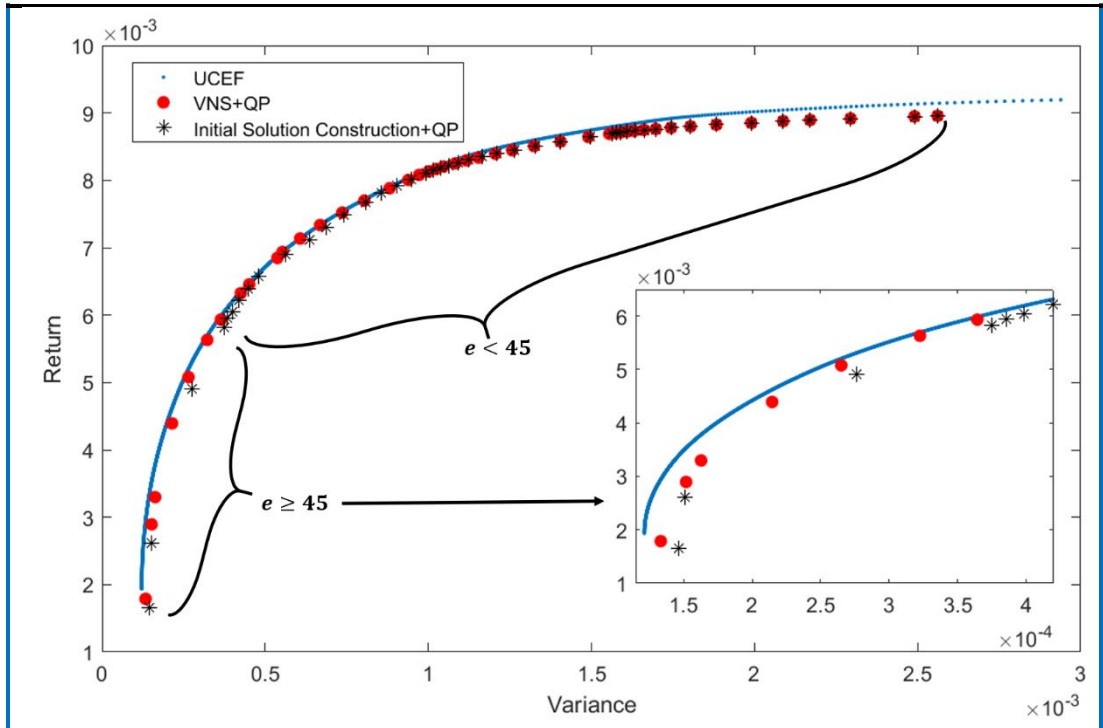
Şekil 3.10: Hang Seng veri seti için etkin sınır.



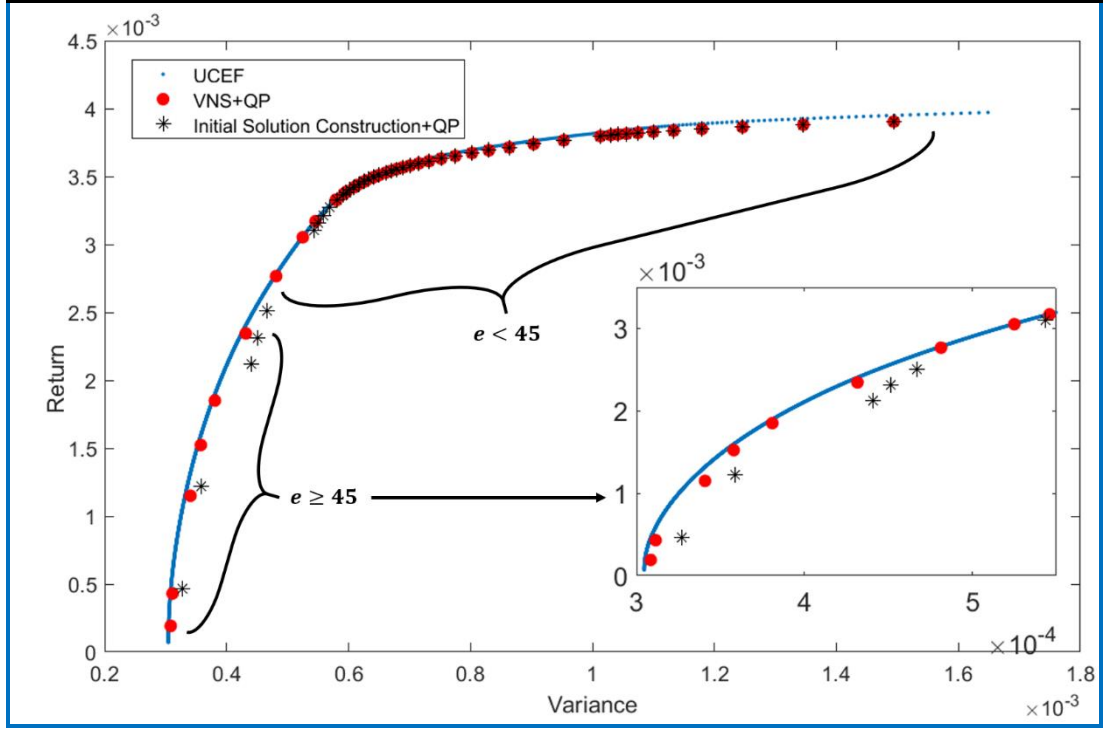
Şekil 3.11: DAX 100 veri seti için etkin sınır.



Şekil 3.12: FTSE 100 veri seti için etkin sınır.



Şekil 3.13: S&P 100 veri seti için etkin sınır.



Şekil 3.14: NIKKEI veri seti için etkin sınır.

Önerilen çözüm yöntemi ile elde edilen hesaplamalı sonuçlar Bölüm 3.4.4'te bahsedilen performans göstergeleri çerçevesinde literatürdeki diğer yöntemlerle karşılaştırılmalı olarak Tablo 3.3, Tablo 3.4 ve Tablo 3.5'de sunulmuştur. Kullanılan performans göstergeleri ve  $\lambda$  değeri sayısı karşılaştırma yapılacak çalışmalarda değişiklik gösterdiği için, sonuçlar üç grup altında toplanmıştır. Ayrıca, adil bir karşılaştırma yapabilmek adına, önerilen çözüm yaklaşımı için maksimum iterasyon sayısı  $1000 \times N$  olarak belirlenmiştir.

İlk olarak, önerilen çözüm yaklaşımının performansı hem düşük  $\lambda$  değerleri ( $e < 45$ ) hem de yüksek  $\lambda$  değerleri ( $e \geq 45$ ) için Deng, ve diğ. (2012) tarafından önerilen PSO algoritması ve INIQuad ile karşılaştırılmıştır. Tablo 3.3 DKAQuad, INIQuad ve Deng ve diğ. (2012) tarafından önerilen PSO algoritması tarafından elde edilen sonuçların ortalama yüzde hata değerlerini göstermektedir. Elde edilen bu sonuçlara göre, önerilen çözüm yaklaşımı özellikle yüksek  $\lambda$  değerlerinde ( $e \geq 45$ ) diğerlerine göre gözle görülür bir üstünlük gösterirken  $\lambda$ 'nın düşük değerlerinde ( $e < 45$ ) ise karşılaştırılabilir sonuçlar alındığı gözlemlenmektedir. Veri seti bazında değerlendirme yapıldığında ise, DAX 100 ve S&P 100 veri setleri için INIQuad çözüm yaklaşımının dahi daha riskli portföy seviyelerinde daha iyi sonuçlar ürettiği görülmektedir.



**Tablo 3.3:** Hesaplamalı sonuçların ortalama yüzde hata ölçütüne göre Deng, ve diğ. (2012) ile performans karşılaştırması ( $\lambda$ 'nın 50 farklı değeri için).

İndeks	$N$	$e$	(Deng, ve diğ., 2012)	INIQuad	DKAQuad
Hang Seng	31	$e < 45$	<b>1.2398</b>	1.3119	1.2507
		$e > 45$	<b>0.0357</b>	0.0809	0.0516
DAX100	85	$e < 45$	<b>2.4189</b>	2.4475	2.4220
		$e > 45$	3.4420	2.9226	<b>1.8707</b>
FTSE100	89	$e < 45$	<b>0.7808</b>	0.8115	0.7854
		$e > 45$	3.1310	4.284	<b>1.5426</b>
S&P100	98	$e < 45$	<b>1.0702</b>	1.2582	1.0744
		$e > 45$	6.2273	4.2827	<b>3.5534</b>
Nikkei	225	$e < 45$	<b>0.6405</b>	0.7132	0.6696
		$e > 45$	1.0281	3.0434	<b>0.7451</b>
Ortalama		$e < 45$	<b>1.2300</b>	1.3085	1.2404
		$e > 45$	2.7728	2.9227	<b>1.5527</b>

Chang ve diğ. (2000), Deng ve diğ. (2012), Lwin ve Qu (2013) tarafından önerilen çözüm yaklaşımları ile birlikte, INIQuad ve DKAQuad'a ait ortalama medyan, minimum ve maksimum yüzde hata değerlerini içeren ikinci bir performans karşılaştırmasına ait sonuçlar Tablo 3.4'te sunulmuştur. Tutarlı bir karşılaştırma yapılabilmesi için her bir hata değeri  $\lambda$ 'nın 50 farklı değeri dikkate alınarak hesaplanmıştır. Elde edilen sonuçlara göre önerilen çözüm yaklaşımının hemen hemen bütün rakiplerinden daha iyi bir performans gösterdiği açıkça görülmektedir.

**Tablo 3.4:** Hesaplamalı sonuçların Chang, ve diğ. (2000), Deng, ve diğ. (2012), Lwin ve Qu (2013) ile performans karşılaştırması ( $\lambda$ 'nın 50 farklı değeri için).

İndeks	Hata N Ölçütü	GA (Chang, ve diğ. 2000)	YA (Chang, ve diğ. 2000)	TB (Chang, ve diğ. 2000)	PSO (Deng, ve diğ. 2012)	PTAÖDE (Lwin ve Qu 2013)	INIQuad	DKAQuad	
Hang									
Seng	31	<i>MEAPE</i>	1.0974	1.1217	<b>1.0957</b>	1.0953	1.1431	1.1640	1.0964
		<i>MEDPE</i>	1.2181	1.2181	1.2181	–	1.2390	1.3090	<b>1.2155</b>
		<i>MINPE</i>	–	–	–	–	–	<b>0.0000</b>	<b>0.0000</b>
		<i>MAXPE</i>	–	–	–	–	–	1.5540	<b>1.5538</b>
DAX									
100	85	<i>MEAPE</i>	2.5424	3.3049	2.9297	2.5417	2.4251	2.5050	<b>2.3125</b>
		<i>MEDPE</i>	2.5466	2.6380	2.5661	–	2.5866	2.6120	<b>2.5630</b>
		<i>MINPE</i>	–	–	–	–	–	0.3619	<b>0.0059</b>
		<i>MAXPE</i>	–	–	–	–	–	6.3090	<b>4.0275</b>
FTSE									
100	89	<i>MEAPE</i>	1.1076	1.6080	1.4623	1.0628	0.9706	1.2280	<b>0.8453</b>
		<i>MEDPE</i>	1.0841	1.0841	1.0841	–	1.0840	1.0840	<b>1.0840</b>
		<i>MINPE</i>	–	–	–	–	–	0.0108	<b>0.0045</b>
		<i>MAXPE</i>	–	–	–	–	–	8.4190	<b>2.0669</b>
S&P									
100	98	<i>MEAPE</i>	1.9328	3.3092	3.0696	1.6890	1.6386	1.6210	<b>1.2649</b>
		<i>MEDPE</i>	1.2244	1.2882	1.1823	–	1.1692	1.2150	<b>1.1323</b>
		<i>MINPE</i>	–	–	–	–	–	0.0000	<b>0.0000</b>
		<i>MAXPE</i>	–	–	–	–	–	9.0600	<b>5.4551</b>
Nikkei									
225		<i>MEAPE</i>	0.7961	0.8975	0.6066	0.6870	0.5972	0.9929	<b>0.5904</b>
		<i>MEDPE</i>	0.6133	0.6093	0.6732	–	0.5896	0.6514	<b>0.5857</b>
		<i>MINPE</i>	–	–	–	–	–	0.2836	<b>0.0000</b>
		<i>MAXPE</i>	–	–	–	–	–	4.8720	<b>1.1606</b>
Ortalama									
		<i>MEAPE</i>	1,4953	2,0483	1,8328	1,4152	1,3549	1,5022	<b>1,2219</b>
		<i>MEDPE</i>	1,2146	1,2457	1,2101	–	1,2158	1,2440	<b>1,1990</b>
		<i>MINPE</i>	–	–	–	–	–	0,2048	<b>0,1192</b>
		<i>MAXPE</i>	–	–	–	–	–	5,1251	<b>2,6207</b>

Son olarak Tablo 3.5 Cura (2009) tarafından önerilen MEUCD, VRE ve MRE performans göstergeleri doğrultusunda önerilen çözüm algoritması ile Cura (2009), Baykasoglu ve diğ. (2015) ve Kalayci ve diğ. (2017) tarafından geliştirilen algoritmaların ve INIQuad çözüm prosedürünün performans karşılaştırmalarını göstermektedir. Karşılaştırmalı sonuçlara göre VRE performans ölçütü baz alındığında ve özellikle Hang Seng, DAX 100, FTSE 100 ve S&P 100 veri setlerinde önerilen çözüm algoritması diğerlerine göre standart etkin sınıra daha iyi bir yakınsama performansı gösterirken, NIKKEI veri setinde ise diğerleriyle karşılaştırılabilir sonuçlar elde edildiği görülmektedir.

**Tablo 3.5:** Hesaplamalı sonuçların Cura (2009), Baykasoglu ve diğ. (2015) and Kalayci ve diğ. (2017) ile performans karşılaştırması ( $\lambda$ 'nın 51 farklı değeri için).

Index	N	Hata Ölçütleri	PSO (Cura 2009)	ARUAP-KP (Baykasoglu ve diğ. 2015)	YAK (Kalayci ve diğ. 2017)	INIQP	DKAQuad
Hang Seng	31	<i>MEUCD</i>	0.0049	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>
		<i>VRE</i>	2.2421	1.6400	1.6432	1.7545	<b>1.6397</b>
		<i>MRE</i>	0.7427	0.6060	<b>0.6047</b>	0.6341	0.6058
DAX 100	85	<i>MEUCD</i>	0.0090	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>
		<i>VRE</i>	6.8588	6.7593	6.7925	7.1446	<b>6.7583</b>
		<i>MRE</i>	1.5885	1.2769	<b>1.2761</b>	1.2798	1,2767
FTSE 100	89	<i>MEUCD</i>	0.0022	<b>0.0000</b>	<b>0.0000</b>	<b>0.0000</b>	<b>0.0000</b>
		<i>VRE</i>	3.0596	2.4350	2.4397	3.1653	<b>2.4349</b>
		<i>MRE</i>	0.3640	<b>0.3245</b>	0.3255	0.3282	0.3252
S&P 100	98	<i>MEUCD</i>	0.0052	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>
		<i>VRE</i>	3.9136	2.5211	2.5260	3.6934	<b>2.5105</b>
		<i>MRE</i>	1.4040	0.9063	<b>0.8885</b>	1.0484	0.9072
Nikkei	225	<i>MEUCD</i>	0.0019	<b>0.0000</b>	<b>0.0000</b>	<b>0.0000</b>	<b>0.0000</b>
		<i>VRE</i>	2.4274	<b>0.8359</b>	0.8396	1.6031	0.8561
		<i>MRE</i>	0.7997	0.4184	<b>0.4127</b>	0.4259	0,4217
Ortalama		<i>MEUCD</i>	0.0046	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>	<b>0.0001</b>
		<i>VRE</i>	3.7003	<b>2.8383</b>	2.8482	3.4722	2.8399
		<i>MRE</i>	0.9798	0.7064	<b>0.7015</b>	0.7433	0.7073

Genel olarak sonuçlar, önerilen çözüm algoritmasının (DKAQuad), eleman sayısı kısıtlı portföy optimizasyon literatüründeki karşılaştırma yapılan diğer algoritmalara karşı çok rekabetçi olduğunu ortaya koymaktadır. Ayrıca, başlangıç çözüm oluşturma metodu ve kuadratik programlamanın bir arada kullanıldığı (INIQuad) çözüm yöntemi de diğerlerine kıyasla oldukça rekabetçi sonuçlar üretmiş olduğu görülmektedir. Hatta bazı veri setlerinde karşılaştırıldığı bazı çözüm yöntemlerinden daha iyi performans sonuçları elde edilmiştir. Bu durum, geliştirilen çözüm yaklaşımında kullanılan başlangıç çözüm inşa stratejisinin önerilen çözüm yaklaşımı için önemli bir avantaj sağladığını göstermektedir.

## 4. SONUÇLAR VE TARTIŞMA

Bu çalışmada, eleman sayısı kısıtlı portföy optimizasyonu probleminin çözümü için asenkron paralelleştirme stratejisi ile iyileştirilmiş değişken komşuluk arama algoritmasını kuadratik programlama ile hibritleyen iki aşamalı bir çözüm yaklaşımı geliştirilmiştir. Geliştirilen algoritmanın ilk aşamada, portföyde yer alan varlıklar değişken komşuluk arama algoritması kullanılarak seçilmektedir. Algoritmanın başlangıç çözümünün ve arama havuzunun belirlenmesi için  $\lambda$  parametresine bağlı olarak hesaplanan ve indeksteki her bir varlığa ait ortalama risk ve ortalama getiri arasındaki oranı temel alan bir sıralama stratejisi kullanılmıştır. Portföye dahil edilecek varlıkların belirlenmesi sonrası, problem, mevcut varlıklar için sermayenin hangi oranlarda dağıtılması gerektiğini arayan basit bir kuadratik programlama modeline dönüşmektedir. Dolayısıyla elde edilen yeni modelin çözümünün kesin çözüm yöntemleri kullanılarak elde edilebilmesi mümkün hale gelmiştir. Bu sebeple çözüm yaklaşımının ikinci aşamasında kuadratik programlamayla portföye seçilen varlıkların ağırlıkları hesaplanmıştır. Önerilen bu çözüm metodolojisi, literatürde sıklıkla kullanılan beş farklı veri setinde test edilmiş ve literatürdeki diğer çözüm yaklaşımları ile karşılaştırılmıştır. Elde edilen sonuçlar, önerilen çözüm yaklaşımının gelişmiş algoritmalarla rekabet edebilir seviyede olduğunu göstermiştir.

Konu ile alakalı gelecekteki muhtemel araştırma yönelimlerinden bazıları aşağıda özetlenmiştir:

- Geliştirilen çözüm yaklaşımları aracılığıyla elde edilen sonuçlar her zaman gelişime açık olduğundan probleme özgü zorluklara ve gerçek hayat koşullarına etkin bir şekilde cevap verebilecek gürbüz çözüm yaklaşımlarının ve algoritma bileşenlerinin geliştirilmesi gelecek çalışmalar için önemli olacaktır.
- Geliştirilen algoritmaların farklı pazarlara ait veri kümeleri üzerinde test edilmesi, pazara özgü değişkenliğe ne kadar iyi yanıt verdiğini görmek açısından büyük önem arz edecektir.
- Ortalama varyans yaklaşımı portföy optimizasyonu probleminin modellenmesinde en yaygın kullanılan yaklaşımlardan birisi olmasına rağmen, gerçek hayat kısıtlarını da içeren ve riske maruz değer (value at risk), koşullu

riske maruz deęer (conditional value at risk), ortalama mutlak sapma (mean absolute deviation) gibi farklı risk ölçütlerini de dikkate alan yeni yaklaşımlarının geliştirilmesi ele alınması gereken önemli hususlardan biri olarak gelecek çalışmalardaki yerini koruyacaktır.

- Önerilen çözüm yaklaşımının çoklu yatırım ufuklarında test edilmesi de araştırmaya deęer konuların başında gelmektedir.

## 5. KAYNAKLAR

Abbas, A., ve Haider, S., “Comparison of AIS and PSO for constrained portfolio optimization”, *In Proceedings - 2009 International Conference on Information and Financial Engineering, ICIFE 2009*, 50-54, (2009).

Ackora-Prah, J., Gyamerah, S. A., ve Andam, P. S., “A heuristic crossover for portfolio selection”, *Applied Mathematical Sciences*, 8, 3215-3227, (2014<sup>a</sup>).

Ackora-Prah, J., Gyamerah, S. A., Andam, P. S., ve Gyamfi, D., “Pattern search for portfolio selection”, *Applied Mathematical Sciences*, 8, 7137-7147, (2014<sup>b</sup>).

Anagnostopoulos, K. P., ve Mamanis, G., “A portfolio optimization model with three objectives and discrete variables”, *Computers and Operations Research*, 37, 1285-1297, (2010).

Anagnostopoulos, K. P., ve Mamanis, G., “The mean–variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms”, *Expert Systems with Applications*, 38, 14208-14217, (2011<sup>a</sup>).

Anagnostopoulos, K. P., ve Mamanis, G., “Multiobjective evolutionary algorithms for complex portfolio optimization problems”, *Computational Management Science*, 8, 259-279, (2011<sup>b</sup>).

Anagnostopoulos, K. P., Šević, Ž., Chatzoglou, P. D., ve Katsavounis, S., “A reactive greedy randomized adaptive search procedure for a mixed integer portfolio optimization problem”, *Managerial Finance*, 36, 1057-1065, (2010).

Aouni, B., Colapinto, C., ve La Torre, D., “Financial portfolio management through the goal programming model: Current state-of-the-art”, *European Journal of Operational Research*, 234, 536-545, (2014).

Aouni, B., Doumpos, M., Pérez-Gladish, B., ve Steuer, R. E., “On the increasing importance of multiple criteria decision aid methods for portfolio selection”, *Journal of the Operational Research Society*, 1-18, (2018).

Aouni, B. F., Ben Abdelaziz, F., ve Martel, J. M., “Decision-maker's preferences modeling in the stochastic goal programming”, *European Journal of Operational Research*, 162, 610-618, (2005).

Aranha, C., ve Iba, H., “The Memetic Tree-based Genetic Algorithm and its application to Portfolio Optimization”, *Memetic Computing*, 1, 139-151, (2009).

Arkeman, Y., Yusuf, A., Mushthofa, Laxmi, G. F., ve Seminar, K. B., “The formation of optimal portfolio of mutual shares funds using multi-objective genetic algorithm”, *Telkomnika*, 11, 625-636, (2013).

Azmi, R., ve Tamiz, M. “A review of goal programming for portfolio selection”, In *Lecture Notes in Economics and Mathematical Systems* (Vol. 638, pp. 15-33), (2010).

Babaei, S., Sepehri, M. M., ve Babaei, E. “Multi-objective portfolio optimization considering the dependence structure of asset returns”, *European Journal of Operational Research*, 244, 525-539, (2015).

Bacanin, N., ve Tuba, M., “Firefly algorithm for cardinality constrained mean-variance portfolio optimization problem with entropy diversity constraint”, *Scientific World Journal*, 2014, (2014).

Bacanin, N., ve Tuba, M., “Fireworks algorithm applied to constrained portfolio optimization problem”, In *2015 IEEE Congress on Evolutionary Computation*, CEC 2015 - Proceedings (pp. 1242-1249), (2015).

Bacanin, N., Tuba, M., ve Pelevic, B., “Constrained portfolio selection using artificial bee colony (ABC) algorithm”, *International Journal of Mathematical Models and Methods in Applied Sciences*, 8, 190-198, (2014).

Bačević, A., Vilimonović, N., Dabić, I., Petrović, J., Damnjanović, D., ve Džamić, D., “Variable neighborhood search heuristic for nonconvex portfolio optimization”, *The Engineering Economist*, 64, 254-274, (2019).

Baluja, S., “Population-based incremental learning. a method for integrating genetic search based function optimization and competitive learning”, In: *DTIC Document*, (1994).

Ban, G. Y., El Karoui, N., ve Lim, A. E. B., “Machine Learning and Portfolio Optimization”, *Management Science*, 64, 1136-1154, (2018).

Bandyopadhyay, S., Saha, S., Maulik, U., ve Deb, K., “A simulated annealing-based multiobjective optimization algorithm: AMOSA”, *IEEE Transactions on Evolutionary Computation*, 12, 269-283, (2008).

Baykasoğlu, A., Avci, M. G., ve Burcin Özsoydan, F., “Erratum to “A GRASP based solution approach to solve cardinality constrained portfolio optimization problems” [Comput. Indus. Eng. 90 (2015) 339–351]”. *Computers and Industrial Engineering*, 96, 249-250, (2016).

Baykasoğlu, A., Yunusoglu, M. G., ve Ozsoydan, F. B., “A GRASP based solution approach to solve cardinality constrained portfolio optimization problems”, *Computers and Industrial Engineering*, 90, 339-351, (2015).

Beasley, J. E., “Or-Library - Distributing Test Problems by Electronic Mail”, *Journal of the Operational Research Society*, 41, 1069-1072, (1990).

Bevilacqua, V., Pacelli, V., ve Saladino, S., “A Novel Multi Objective Genetic Algorithm for the Portfolio Optimization”, In D.-S. Huang, Y. Gan, V. Bevilacqua ve J. C. Figueroa (Eds.), *Advanced Intelligent Computing* (pp. 186-193). Berlin, Heidelberg: Springer Berlin Heidelberg, (2011).

Blum, C., ve Roli, A., “Metaheuristics in combinatorial optimization: Overview and conceptual comparison”, *Acm Computing Surveys*, 35, 268-308, (2003).

Bonami, P., ve Lejeune, M. A., “An Exact Solution Approach for Portfolio Optimization Problems Under Stochastic and Integer Constraints”, *Operations Research*, 57, 650-670, (2009).

Branke, J., Scheckenbach, B., Stein, M., Deb, K., ve Schmeck, H., “Portfolio optimization with an envelope-based multi-objective evolutionary algorithm”, *European Journal of Operational Research*, 199, 684-693, (2009).

Brito, R. P., ve Vicente, L. N., “Efficient Cardinality/Mean-Variance Portfolios”, In (pp. 52-73). Berlin, Heidelberg: Springer Berlin Heidelberg, (2014<sup>a</sup>).

Brito, R. P., ve Vicente, L. N., “Efficient cardinality/mean-variance portfolios”, In *IFIP Advances in Information and Communication Technology* (Vol. 443, pp. 52-73), (2014<sup>b</sup>).

Busetti, F., “Heuristic approaches to realistic portfolio optimisation”, In *WIT Transactions on Modelling and Simulation* (Vol. 43, pp. 361-370), (2006).

Cao, J., ve Tao, L., “Improved particle swarm algorithm for portfolio optimization problem”, In *ICIMA 2010 - 2010 2nd International Conference on Industrial Mechatronics and Automation* (Vol. 2, pp. 561-564), (2010).



Canuto, S. A., Resende, M. G. C., ve Ribeiro, C. C., “Local search with perturbations for the prize-collecting Steiner tree problem in graphs”, *Networks*, 38, 50-58, (2001).

Cesarone, F., Scozzari, A., ve Tardella, F., “A new method for mean-variance portfolio optimization with cardinality constraints”, *Annals of Operations Research*, 205, 213-234, (2013).

Cesarone, F., Scozzari, A., ve Tardella, F., “Linear vs. quadratic portfolio selection models with hard real-world constraints”, *Computational Management Science*, 12, 345-370, (2015).

Chang, J. F., ve Chen, K. L., “Applying new investment satisfied capability index and particle swarm optimization to stock portfolio selection”, *ICIC Express Letters*, 3, 349-354, (2008).

Chang, J. F., ve Hsu, S. W., “The construction of stock's portfolios by using particle swarm optimization”, *In Second International Conference on Innovative Computing, Information and Control, ICICIC 2007*, (2007).

Chang, T. J., Meade, N., Beasley, J. E., ve Sharaiha, Y. M., “Heuristics for cardinality constrained portfolio optimisation”, *Computers and Operations Research*, 27, 1271-1302, (2000).

Chang, T. J., Yang, S. C., ve Chang, K. J., “Portfolio optimization problems in different risk measures using genetic algorithm”, *Expert Systems with Applications*, 36, 10529-10537, (2009).

Chen, A. H. L., Liang, Y. C., ve Liu, C. C., “Portfolio optimization using improved artificial bee colony Approach”, *In Proceedings of the 2013 IEEE Conference on Computational Intelligence for Financial Engineering and Economics, CIFE 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013* (pp. 60-67), (2013).

Chen, A. H. L., Yun-Chia, L., ve Chia-Chien, L., “An artificial bee colony algorithm for the cardinality-constrained portfolio optimization problems”, *In Evolutionary Computation (CEC), 2012 IEEE Congress on* (pp. 1-8), (2012).

Chen, B. L., Lin, Y. B., Zeng, W. H., Xu, H., ve Zhang, D. F., “The mean-variance cardinality constrained portfolio optimization problem using a local search-based multi-objective evolutionary algorithm”, *Applied Intelligence*, 47, 505-525, (2017).

Chen, C., ve Zhou, Y. S., “Robust multiobjective portfolio with higher moments”, *Expert Systems with Applications*, 100, 165-181, (2018).

Chen, W., “Artificial bee colony algorithm for constrained possibilistic portfolio optimization problem”, *Physica a-Statistical Mechanics and Its Applications*, 429, 125-139, (2015).

Chen, W., ve Cai, Y. M., “Study on the efficient frontier in portfolio selection by using particle swarm optimization”, *In Chinese Control and Decision Conference, 2008, CCDC 2008* (pp. 269-272), (2008).

Chen, W., Xu, W. J., Yang, L., ve Cai, Y. M., “Genetic algorithm with an application to complex portfolio selection”, *In Proceedings - 4th International Conference on Natural Computation, ICNC 2008* (Vol. 5, pp. 333-337): IEEE, (2008).

Chen, W., Zhang, R. T., Cai, Y. M., ve Xu, F. S., “Particle swarm optimization for constrained portfolio selection problems”, *In Proceedings of the 2006 International Conference on Machine Learning and Cybernetics* (Vol. 2006, pp. 2425-2429): IEEE, (2006).

Chen, W., ve Zhang, W. G., “The admissible portfolio selection problem with transaction costs and an improved PSO algorithm”, *Physica a-Statistical Mechanics and Its Applications*, 389, 2070-2076, (2010).

Chiam, S. C., Al Mamun, A., ve Low, Y. L., “A realistic approach to evolutionary multiobjective portfolio optimization”, *In 2007 IEEE Congress on Evolutionary Computation, CEC 2007* (pp. 204-211), (2007).

Chiam, S. C., Tan, K. C., ve Mamun, A., “Evolutionary multi-objective portfolio optimization in practical context”, *International Journal of Automation and Computing*, 5, 67-80, (2008).

ChiangLin, C. Y., “Applications of genetic algorithm to portfolio optimization with practical transaction constraints”, *In Proceedings of the 9th Joint Conference on Information Sciences, JCIS 2006* (Vol. 2006), (2006).

Chu, S. C., ve Tsai, P. W., “Computational intelligence based on the behavior of cats”, *International Journal of Innovative Computing Information and Control*, 3, 163-173, (2007).

Civicioglu, P., “Artificial cooperative search algorithm for numerical optimization problems”, *Information Sciences*, 229, 58-76, (2013).

Coello, C. A. C., Pulido, G. T., ve Lechuga, M. S., “Handling multiple objectives with particle swarm optimization”, *IEEE Transactions on Evolutionary Computation*, 8, 256-279, (2004).

Corazza, M., Fasano, G., ve Gusso, R., “Portfolio selection with an alternative measure of risk: Computational performances of particle swarm optimization and genetic algorithms”, In C. Perna ve M. Sibillo (Eds.), *Mathematical and Statistical Methods for Actuarial Sciences and Finance* (pp. 123-130). Milano: Springer Milan, (2012<sup>a</sup>).

Corazza, M., Fasano, G., ve Gusso, R., “Portfolio selection with an alternative measure of risk: Computational performances of Particle Swarm Optimization and Genetic Algorithms”, In *Mathematical and Statistical Methods for Actuarial Sciences and Finance* (pp. 123-130), (2012<sup>b</sup>).

Corazza, M., Fasano, G., ve Gusso, R., “Particle Swarm Optimization with non-smooth penalty reformulation, for a complex portfolio selection problem”, *Applied Mathematics and Computation*, 224, 611-624, (2013).

Corne, D., Knowles, J., ve Oates, M., “The Pareto envelope-based selection algorithm for multiobjective optimization”, In *Parallel problem solving from nature PPSN VI* (pp. 839-848): Springer, (2000).

Côté, P., Wong, T., ve Sabourin, R., “A Hybrid Multi-objective Evolutionary Algorithm for the Uncapacitated Exam Proximity Problem”, In (pp. 294-312). Berlin, Heidelberg: Springer Berlin Heidelberg, (2005).

Coutino-Gomez, C. A., Torres-Jimenez, J., ve Villarreal-Antelo, B. M., “Heuristic Methods for Portfolio Selection at the Mexican Stock Exchange”, In J. Liu, Y.-m. Cheung ve H. Yin (Eds.), *Intelligent Data Engineering and Automated Learning: 4th International Conference, IDEAL 2003, Hong Kong, China, March 21-23, 2003. Revised Papers* (pp. 919-923). Berlin, Heidelberg: Springer Berlin Heidelberg, (2003).

Crainic, T. G., Gendreau, M., Hansen, P., ve Mladenović, N., “Cooperative Parallel Variable Neighborhood Search for the p-Median”, *Journal of Heuristics*, 10, 293-314, (2004).

Crainic, T. G., ve Hail, N., “Parallel Metaheuristics Applications”, In E. Alba (Ed.), *Parallel metaheuristics: A new class of algorithms*. Hoboken, New Jersey: John Wiley ve Sons, (2005).

Crainic, T. G., Toulouse, M., & Gendreau, M., “Toward a Taxonomy of Parallel Tabu Search Heuristics”, *INFORMS Journal on Computing*, 9, 61-72, (1997).

Crama, Y., ve Schyns, M., “Simulated annealing for complex portfolio selection problems”, *European Journal of Operational Research*, 150, 546-571, (2003).

Cui, T., Cheng, S., ve Bai, R., “A combinatorial algorithm for the cardinality constrained portfolio optimization problem”, *In Proceedings of the 2014 IEEE Congress on Evolutionary Computation*, CEC 2014 (pp. 491-498), (2014).

Cui, X. T., Zheng, X. J., Zhu, S. S., ve Sun, X. L., “Convex relaxations and MIQCQP reformulations for a class of cardinality-constrained portfolio selection problems”, *Journal of Global Optimization*, 56, 1409-1423, (2013).

Cura, T., “Particle swarm optimization approach to portfolio optimization”, *Nonlinear Analysis-Real World Applications*, 10, 2396-2406, (2009).

Czyżżak, P., ve Jaskiewicz, A., “Pareto simulated annealing—a metaheuristic technique for multiple-objective combinatorial optimization”, *Journal of Multi-Criteria Decision Analysis*, 7, 34-47, (1998).

De Castro, L. N., ve Timmis, J., “Artificial immune systems: a new computational intelligence Approach”, *Springer Science ve Business Media*, (2002).

Deb, K., “An efficient constraint handling method for genetic algorithms”, *Computer Methods in Applied Mechanics and Engineering*, 186, 311-338, (2000).

Deb, K., “Multi-objective optimization using evolutionary algorithms”, (*Vol. 16*): *John Wiley ve Sons*, (2001).

Deb, K., “Multi-Objective Optimization”, *In E. K. Burke ve G. Kendall (Eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques* (pp. 273-316). Boston, MA: Springer US, (2005).

Deb, K., Pratap, A., Agarwal, S., ve Meyarivan, T., “A fast and elitist multiobjective genetic algorithm: NSGA-II”, *IEEE Transactions on Evolutionary Computation*, 6, 182-197, (2002).

Dehghan Hardoroudi, N., Keshvari, A., Kallio, M., ve Korhonen, P., “Solving cardinality constrained mean-variance portfolio problems via MILP”, *Annals of Operations Research*, 254, 47-59, (2017).

Deng, G.-F., ve Lin, W.-T., “Ant Colony Optimization for Markowitz Mean-Variance Portfolio Model”, *In B. K. Panigrahi, S. Das, P. N. Suganthan ve S. S. Dash (Eds.), Swarm, Evolutionary, and Memetic Computing: First International Conference on Swarm, Evolutionary, and Memetic Computing, SEMCCO 2010, Chennai, India, December 16-18, 2010. Proceedings* (pp. 238-245). Berlin, Heidelberg: Springer Berlin Heidelberg, (2010<sup>a</sup>).

Deng, G.-F., ve Lin, W.-T., “Swarm intelligence for cardinality-constrained portfolio problems”, *In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 6423 LNAI, pp. 406-415), (2010<sup>b</sup>).

Deng, G. F., Lin, W. T., ve Lo, C. C., “Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization” *Expert Systems with Applications*, 39, 4558-4566, (2012).

Doering, J., Juan, A. A., Kizys, R., Fito, A., ve Calvet, L., “Solving realistic portfolio optimization problems via metaheuristics: a survey and an example”, *In Modeling and Simulation in Engineering, Economics and Management* (Vol. 254, pp. 22-30): Springer, (2016).

Dorigo, M., Maniezzo, V., ve Colorni, A., “Ant system: optimization by a colony of cooperating agents”, *IEEE Trans Syst Man Cybern B Cybern*, 26, 29-41, (1996).

Dreżewski, R., ve Doroz, K., “An Agent-Based Co-Evolutionary Multi-Objective Algorithm for Portfolio Optimization”, *Symmetry*, 9, 168-198, (2017).

Duran, F. C., Cotta, C., ve Fernández, A. J., “Evolutionary Optimization for Multiobjective Portfolio Selection under Markowitz’s Model with Application to the Caracas Stock Exchange”, *In R. Chiong (Ed.), Nature-Inspired Algorithms for Optimisation* (Vol. 193, pp. 489-509). Berlin, Heidelberg: Springer Berlin Heidelberg, (2009).

Eftekharian, S., Shojafar, M., ve Shamshirband, S., “2-Phase NSGA II: An Optimized Reward and Risk Measurements Algorithm in Portfolio Optimization”, *Algorithms*, 10, 130-143, (2017).

Ehrgott, M., Klamroth, K., ve Schwehm, C., “An MCDM approach to portfolio optimization”, *European Journal of Operational Research*, 155, 752-770, (2004).

Erickson, M., Mayer, A., ve Horn, J., “The niched pareto genetic algorithm 2 applied to the design of groundwater remediation systems”, *In International Conference on Evolutionary Multi-Criterion Optimization* (Vol. 1993, pp. 681-695): Springer, (2001).

Ertenlice, O., ve Kalayci, C. B., “A survey of swarm intelligence for portfolio optimization: Algorithms and applications”, *Swarm and Evolutionary Computation*, (2018).

Farzi, S., Shavazi, A. R., ve Pandari, A., “Using Quantum-Behaved Particle Swarm Optimization for Portfolio Selection Problem”, *International Arab Journal of Information Technology*, 10, 111-119, (2013).

Fasheng, X., ve Wei, C., “Stochastic portfolio selection based on velocity limited particle swarm optimization”, *In Proceedings of the World Congress on Intelligent Control and Automation (WCICA)* (Vol. 1, pp. 3599-3603): IEEE, (2006).

Fastrich, B., ve Winker, P., “Robust portfolio optimization with a hybrid heuristic algorithm”, *Computational Management Science*, 9, 63-88, (2012).

Feo, T. A., ve Resende, M. G. C., “Greedy Randomized Adaptive Search Procedures”, *Journal of Global Optimization*, 6, 109-133, (1995).

Fernandez, A., ve Gomez, S., “Portfolio selection using neural networks”, *Computers and Operations Research*, 34, 1177-1191, (2007).

Fieldsend, J. E., Matatko, J., ve Peng, M., “Cardinality Constrained Portfolio Optimisation”, *In Z. R. Yang, H. Yin ve R. M. Everson (Eds.), Intelligent Data Engineering and Automated Learning – IDEAL 2004: 5th International Conference, Exeter, UK. August 25-27, 2004. Proceedings* (pp. 788-793). Berlin, Heidelberg: Springer Berlin Heidelberg, (2004).

Fonseca, C. M., ve Fleming, P. J., “Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization”, *In Icga* (Vol. 93, pp. 416-423): Citeseer, (1993).

Forrest, S., ve Mitchell, M., “Relative Building-Block Fitness and the Building-Block Hypothesis”, *Foundations of Genetic Algorithms 2*, 2, 109-126, (1993).

Freitas, F. D., De Souza, A. F., ve de Almeida, A. R., “Prediction-based portfolio optimization model using neural networks”, *Neurocomputing*, 72, 2155-2170, (2009).

Gandomi, A. H., ve Alavi, A. H., “Krill herd: A new bio-inspired optimization algorithm”, *Communications in Nonlinear Science and Numerical Simulation*, 17, 4831-4845, (2012).

Gao, J., ve Chu, Z., “An improved particle swarm optimization for the constrained portfolio selection problem”, *In Computational Intelligence and Natural Computing*, 2009. CINC'09. International Conference on (Vol. 1, pp. 518-522): IEEE, (2009).

García, F., Guijarro, F., ve Oliver, J., “Index tracking optimization with cardinality constraint: a performance comparison of genetic algorithms and tabu search heuristics”, *Neural Computing and Applications*, 30, 2625-2641, (2018).

Garcia, S., Quintana, D., Galvan, I. M., ve Isasi, P., “Time-stamped resampling for robust evolutionary portfolio optimization”, *Expert Systems with Applications*, 39, 10722-10730, (2012).

García-López, F., Melián-Batista, B., Moreno-Pérez, J. A., ve Moreno-Vega, J. M., “The Parallel Variable Neighborhood Search for the p-Median Problem”, *Journal of Heuristics*, 8, 375-388, (2002).

Gaspero, L. D., Tollo, G. D., Roli, A., ve Schaerf, A., “Hybrid metaheuristics for constrained portfolio selection problems”, *Quantitative Finance*, 11, 1473-1487, (2011).

Glover, F., “Future Paths for Integer Programming and Links to Artificial-Intelligence”, *Computers and Operations Research*, 13, 533-549, (1986).

Glover, F., Lagunai, M., ve Marti, R., “Fundamentals of scatter search and path relinking”, *Control and cybernetics*, 29, 653-684, (2000).

Golmakani, H. R., ve Alishah, E. J., “Portfolio selection using an artificial immune system”, *In 2008 IEEE International Conference on Information Reuse and Integration, IEEE IRI-2008* (pp. 28-33), (2008).

Golmakani, H. R., ve Fazel, M., “Constrained Portfolio Selection using Particle Swarm Optimization”, *Expert Systems with Applications*, 38, 8327-8335, (2011).

Govindan, K., Soleimani, H., ve Kannan, D., “Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future”, *European Journal of Operational Research*, 240, 603-626, (2015).

Hadi, A. S., El Nagggar, A. A., ve Bary, M. N. A., “New model and method for portfolios selection”, *Applied Mathematical Sciences*, 10, 263-288, (2016).

Hanne, T., “A multiobjective evolutionary algorithm for approximating the efficient set”, *European Journal of Operational Research*, 176, 1723-1734, (2007).

Hansen, P., ve Mladenovic, N., “Variable neighborhood search: Principles and applications”, *European Journal of Operational Research*, 130, 449-467, (2001).

Hansen, P., ve Mladenović, N., “An Introduction to Variable Neighborhood Search”, In S. Voß, S. Martello, I. H. Osman ve C. Roucairol (Eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization* (pp. 433-458). Boston, MA: Springer US, (1999).

Hansen, P., ve Mladenović, N., “J-Means: a new local search heuristic for minimum sum of squares clustering”, *Pattern Recognition*, 34, 405-413, (2001).

Hansen, P., ve Mladenović, N., “First vs. best improvement: An empirical study”, *Discrete Applied Mathematics*, 154, 802-817, (2006).

Hansen, P., Mladenović, N., ve Moreno Pérez, J. A., “Variable neighbourhood search: methods and applications”, *Annals of Operations Research*, 175, 367-407, (2009).

Hansen, P., Mladenović, N., ve Perez-Britos, D., “Variable Neighborhood Decomposition Search”, *Journal of Heuristics*, 7, 335-350, (2001).

Hansen, P., Mladenović, N., ve Urošević, D., “Variable neighborhood search and local branching”, *Computers and Operations Research*, 33, 3034-3045, (2006).

Hao, F. F., ve Liu, Y. K., “Mean-variance models for portfolio selection with fuzzy random returns”, *Journal of Applied Mathematics and Computing*, 30, 9-38, (2009).



He, F., ve Qu, R., “Hybridising Local Search with Branch-and-Bound for Constrained Portfolio Selection Problems”, *In Proceedings - 30th European Conference on Modelling and Simulation Ecms 2016* (pp. 446-452), (2016).

Hoklie, ve Zuhail, L. R., “Resolving multi objective stock portfolio optimization problem using genetic algorithm”, *In 2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)* (Vol. 2, pp. 40-44): IEEE, (2010).

Holland, J. H., “Adaptation in natural and artificial systems”, *Ann Arbor, MI: The University of Michigan Press*, (1975).

Hong-mei, L., Zhuo-fu, W., ve Hui-min, L., “Artificial bee colony algorithm for real estate portfolio optimization based on risk preference coefficient”, *In Management Science and Engineering (ICMSE), 2010 International Conference on* (pp. 1682-1687): IEEE, (2010).

Hooke, R., ve Jeeves, T. A., “Direct Search Solution of Numerical and Statistical Problems”, *Journal of the ACM (JACM)*, 8, 212-229, (1961).

Hopfield, J. J., “Neurons with graded response have collective computational properties like those of two-state neurons”, *Proceedings of the national academy of sciences*, 81, 3088-3092, (1984).

Horn, J., Nafpliotis, N., ve Goldberg, D. E., “A niched Pareto genetic algorithm for multiobjective optimization”, *In Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on* (pp. 82-87): Ieee, (1994).

Hu, J., ve Zhangy, G., “Comparison of portfolio optimization models with real features: An empirical study based on Chinese stock market”, *Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications and Algorithms*, 17, 83-100, (2010).

Huang, X., ve Shen, W., “Multi-period mean-variance model with transaction cost for fuzzy portfolio selection”, *In Proceedings - 2010 7th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2010* (Vol. 2, pp. 894-898): IEEE, (2010).

Huang, X. X., “Mean-variance models for portfolio selection subject to experts' estimations”, *Expert Systems with Applications*, 39, 5887-5893, (2012).

Jalota, H., ve Thakur, M., “Genetic algorithm designed for solving portfolio optimization problems subjected to cardinality constraint”, *International*

*Journal of System Assurance Engineering and Management*, 9, 294-305, (2018).

Jiang, K., Li, D., Gao, J., ve Yu, J. X., “Factor model based clustering approach for cardinality constrained portfolio selection”, *In IFAC Proceedings Volumes (IFAC-PapersOnline)* (Vol. 19, pp. 10713-10718): Elsevier, (2014).

Jiang, W., Zhang, Y., ve Xie, J., “A particle swarm optimization algorithm based on diffusion-repulsion and application to portfolio selection”, *In 2008 International Symposium on Information Science and Engineering, ISISE 2008* (Vol. 2, pp. 498-501): IEEE, (2008).

Jin, Y., Qu, R., ve Atkin, J., “A population-based incremental learning method for constrained portfolio optimisation”, *In Proceedings - 16th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2014* (pp. 212-219): IEEE, (2015).

Kalayci, C. B., Ertenlice, O., ve Akbay, M. A., “A comprehensive review of deterministic models and applications for mean-variance portfolio optimization”, *Expert Systems with Applications*, 125, 345-368, (2019).

Kalayci, C. B., Ertenlice, O., Akyer, H., ve Aygoren, H., “An artificial bee colony algorithm with feasibility enforcement and infeasibility toleration procedures for cardinality constrained portfolio optimization”, *Expert Systems with Applications*, 85, 61-75, (2017).

Kalayci, C. B., ve Kaya, C., “An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery”, *Expert Systems with Applications*, 66, 163-175, (2016).

Kalayci, C. B., Polat, O., ve Gupta, S. M., “A variable neighbourhood search algorithm for disassembly lines”, *Journal of Manufacturing Technology Management*, 26, 182-194, (2015).

Kalayci, C. B., Polat, O., ve Gupta, S. M., “A hybrid genetic algorithm for sequence-dependent disassembly line balancing problem”, *Annals of Operations Research*, 242, 321-354, (2016).

Kamali, S., “Portfolio optimization using particle swarm optimization and genetic algorithm”, *Journal of mathematics and computer science*, 10, 85-90, (2014).

Kamili, H., ve Riffi, M. E., “Portfolio selection using the cat swarm optimization”, *Journal of Theoretical and Applied Information Technology*, 74, 374-380, (2015).

Kamili, H., ve Riffi, M. E., “A comparative study on portfolio optimization problem”, *In Proceedings - 2016 International Conference on Engineering and MIS, ICEMIS 2016: IEEE*, (2016).

Kao, Y. C., ve Cheng, H. T., “Bacterial Foraging Optimization Approach to Portfolio Optimization”, *Computational Economics*, 42, 453-470, (2013).

Karaboga, D., “An idea based on Honey Bee Swarm for Numerical Optimization”, *In (Vol. 200). Kayseri: Erciyes University, Engineering Faculty, Computer Engineering Department*, (2005).

Karp, R. M., “Reducibility among Combinatorial Problems”, In R. E. Miller, J. W. Thatcher ve J. D. Bohlinger (Eds.), *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20–22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, and sponsored by the Office of Naval Research, Mathematics Program, IBM World Trade Corporation, and the IBM Research Mathematical Sciences Department* (pp. 85-103). Boston, MA: Springer US, (1972).

Keating, C., ve Shadwick, W. F., “A universal performance measure”, *Journal of performance measurement*, 6, 59-84, (2002).

Kennedy, J., ve Eberhart, R., “Particle swarm optimization”, *In Neural Networks, 1995. Proceedings., IEEE International Conference on* (Vol. 4, pp. 1942-1948 vol.1944), (1995).

Khare, V., Yao, X., ve Deb, K., “Performance Scaling of Multi-objective Evolutionary Algorithms”, *In (pp. 376-390). Berlin, Heidelberg: Springer Berlin Heidelberg*, (2003).

Kırış, Ş., ve Ustun, O., “An integrated approach for stock evaluation and portfolio optimization”, *Optimization*, 61, 423-441, (2012).

Kirkpatrick, S., Gelatt, C. D., Jr., ve Vecchi, M. P., “Optimization by simulated annealing”, *Science*, 220, 671-680, (1983).

Knowles, J., Thiele, L., ve Zitzler, E., “A tutorial on the performance assessment of stochastic multiobjective optimizers”, *Tik report*, 214, 327-332, (2006).

Kocadağlı, O., ve Keskin, R., “A novel portfolio selection model based on fuzzy goal programming with different importance and priorities”, *Expert Systems with Applications*, 42, 6898-6912, (2015).

Kolm, P. N., Tutuncu, R., ve Fabozzi, F. J., “60 Years of portfolio optimization: Practical challenges and current trends”, *European Journal of Operational Research*, 234, 356-371, (2014).

Konno, H., ve Yamazaki, H., “Mean-Absolute Deviation Portfolio Optimization Model and Its Applications to Tokyo Stock-Market”, *Management Science*, 37, 519-531, (1991).

Koshino, M., Murata, H., ve Kimura, H., “Improved particle swarm optimization and application to portfolio selection”, *Electronics and Communications in Japan Part III-Fundamental Electronic Science*, 90, 13-25, (2007).

Kukkonen, S., ve Lampinen, J. GDE3: “The third evolution step of generalized differential evolution”, *In Evolutionary Computation, 2005. The 2005 IEEE Congress on* (Vol. 1, pp. 443-450): IEEE, (2005).

Kumar, D., ve Mishra, K. K., “Portfolio optimization using novel co-variance guided Artificial Bee Colony algorithm”, *Swarm and Evolutionary Computation*, 33, 119-130, (2017).

Kumar, R., ve Bhattacharya, S., “Cooperative Search Using Agents for Cardinality Constrained Portfolio Selection Problem”, *IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews*, 42, 1510-1518, (2012).

Lai, K. K., Yu, L., Wang, S., ve Zhou, C., “A Double-Stage Genetic Optimization Algorithm for Portfolio Selection”, *In I. King, J. Wang, L.-W. Chan ve D. Wang (Eds.), Neural Information Processing: 13th International Conference, ICONIP 2006, Hong Kong, China, October 3-6, 2006. Proceedings, Part III* (pp. 928-937). Berlin, Heidelberg: Springer Berlin Heidelberg, (2006).

Lam, A. Y. S., ve Li, V. O. K., “Chemical-Reaction-Inspired Metaheuristic for Optimization”, *IEEE Transactions on Evolutionary Computation*, 14, 381-399, (2010).

Lean, Y., Wang, S., ve Lai, K. K., “Portfolio optimization using evolutionary algorithms”, In *Reflexing Interfaces: The Complex Coevolution of Information Technology Ecosystems* (pp. 235-245), (2008).

Li, D., Sun, X. L., ve Wang, J., “Optimal lot solution to cardinality constrained mean-variance formulation for portfolio selection”, *Mathematical Finance*, 16, 83-101, (2006).

Li, L., Xue, B., Tan, L., ve Niu, B., “Improved particle swarm optimizers with application on constrained portfolio selection”, In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 6215 LNCS, pp. 579-586), (2010).

Li, X., Zhang, Y., Wong, H. S., ve Qin, Z. F., “A hybrid intelligent algorithm for portfolio selection problem with fuzzy returns”, *Journal of Computational and Applied Mathematics*, 233, 264-278, (2009).

Liagkouras, K., “A new three-dimensional encoding multiobjective evolutionary algorithm with application to the portfolio optimization problem”, *Knowledge-Based Systems*, (2018).

Liagkouras, K., ve Metaxiotis, K., “A new Probe Guided Mutation operator and its application for solving the cardinality constrained portfolio optimization problem”, *Expert Systems with Applications*, 41, 6274-6290, (2014).

Liagkouras, K., ve Metaxiotis, K., “Handling the complexities of the multi-constrained portfolio optimization problem with the support of a novel MOEA”, *Journal of the Operational Research Society*, 1-19, (2017).

Liagkouras, K., ve Metaxiotis, K., “A new efficiently encoded multiobjective algorithm for the solution of the cardinality constrained portfolio optimization problem”, *Annals of Operations Research*, 267, 281-319, (2018).

Liang, J. J., ve Qu, B. Y., “Large-scale portfolio optimization using multiobjective dynamic multi-swarm particle swarm optimizer”, In *Proceedings of the 2013 IEEE Symposium on Swarm Intelligence, SIS 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013* (pp. 1-6), (2013).

Lin, C. C., ve Liu, Y. T., “Genetic algorithms for portfolio selection problems with minimum transaction lots”, *European Journal of Operational Research*, 185, 393-404, (2008).

Lin, D., Li, X., ve Li, M., “A Genetic Algorithm for Solving Portfolio Optimization Problems with Transaction Costs and Minimum Transaction Lots”, In (Vol. 3612, pp. 808-811). Berlin, Heidelberg: Springer Berlin Heidelberg, (2005).

Liu, Y. J., ve Zhang, W. G., “A multi-period fuzzy portfolio optimization model with minimum transaction lots”, *European Journal of Operational Research*, 242, 933-941, (2015).

Lloyd, S. P., “Least-Squares Quantization in Pcm”, *IEEE transactions on information theory*, 28, 129-137, (1982).

Loukeris, N., Donnelly, D., Khuman, A., ve Peng, Y., “A numerical evaluation of meta-heuristic techniques in portfolio optimisation”, *Operational Research*, 9, 81-103, (2009).

Lu, Z., ve Wang, X., “Improved portfolio optimization with non-convex and non-concave cost using genetic algorithms”, In *Proceedings - 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer, MEC 2013* (pp. 2567-2570): IEEE, (2013).

Lwin, K., ve Qu, R., “A hybrid algorithm for constrained portfolio selection problems”, *Applied Intelligence*, 39, 251-266, (2013).

Lwin, K., Qu, R., ve Kendall, G., “A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization”, *Applied Soft Computing*, 24, 757-772, (2014).

Lwin, K., Qu, R., ve Zheng, J., “Multi-objective scatter search with external archive for portfolio optimization”, In *IJCCI 2013 - Proceedings of the 5th International Joint Conference on Computational Intelligence* (pp. 111-119), (2013).

Lwin, K. T., Qu, R., ve MacCarthy, B. L., “Mean-VaR portfolio optimization: A nonparametric Approach”, *European Journal of Operational Research*, 260, 751-766, (2017).

Macedo, L. L., Godinho, P., ve Alves, M. J., “Mean-semivariance portfolio optimization with multiobjective evolutionary algorithms and technical analysis rules”, *Expert Systems with Applications*, 79, 33-43, (2017).

Mansini, R., Ogryczak, W., ve Speranza, M. G., “Twenty years of linear programming based portfolio optimization”, *European Journal of Operational Research*, 234, 518-535, (2014).

Mansour, N., Rebai, A., ve Aouni, B., “Portfolio selection through the imprecise goal programming model: Integration of manager’s preferences”, *Journal of Industrial Engineering International*, 3, 1-8, (2007).

Maringer, D., ve Kellerer, H., “Optimization of cardinality constrained portfolios with a hybrid local search algorithm”, *OR Spectrum*, 25, 481-495, (2003).

Markowitz, H., “Portfolio Selection”, *The Journal of Finance*, 7, 77-91, (1952).

Marler, R. T., ve Arora, J. S., “Survey of multi-objective optimization methods for engineering”, *Structural and Multidisciplinary Optimization*, 26, 369-395, (2004).

Masmoudi, M., ve Abdelaziz, F. B., “Portfolio selection problem: a review of deterministic and stochastic multiple objective programming models”, *Annals of Operations Research*, 267, 335-352, (2018).

Mayambala, F., Rönnberg, E., ve Larsson, T., “Eigendecomposition of the mean-variance portfolio optimization model”, *In Springer Proceedings in Mathematics and Statistics* (Vol. 130, pp. 209-232), (2015).

Mehrabian, A. R., ve Lucas, C., “A novel numerical optimization algorithm inspired from weed colonization”, *Ecological informatics*, 1, 355-366, (2006).

Metaxiotis, K., ve Liagkouras, K., “Multiobjective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review”, *Expert Systems with Applications*, 39, 11685-11698, (2012).

Mishra, S. K., Panda, G., ve Majhi, B., “Prediction based mean-variance model for constrained portfolio assets selection using multiobjective evolutionary algorithms”, *Swarm and Evolutionary Computation*, 28, 117-130, (2016).

Mishra, S. K., Panda, G., ve Majhi, R., “A comparative performance assessment of a set of multiobjective algorithms for constrained portfolio assets selection”, *Swarm and Evolutionary Computation*, 16, 38-51, (2014<sup>a</sup>).

Mishra, S. K., Panda, G., ve Majhi, R., “Constrained portfolio asset selection using multiobjective bacteria foraging optimization”, *Operational Research*, 14, 113-145, (2014<sup>b</sup>).

Mishra, S. K., Panda, G., ve Meher, S., “Multi-objective particle swarm optimization approach to portfolio optimization”, *In 2009 World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings* (pp. 1612-1615): IEEE, (2009).

Mladenovic, N., ve Hansen, P., “Variable neighborhood search”, *Computers and Operations Research*, 24, 1097-1100, (1997).

Moral-Escudero, R., Ruiz-Torrubiano, R., ve Suarez, A., “Selection of Optimal Investment Portfolios with Cardinality Constraints”, *In Evolutionary Computation, 2006. CEC 2006. IEEE Congress on* (pp. 2382-2388): IEEE, (2006).

Mozafari, M., Jolai, F., ve Tafazzoli, S., “A new IPSO-SA approach for cardinality constrained portfolio optimization”, *International Journal of Industrial Engineering Computations*, 2, 249-262, (2011).

Ni, Q. J., Yin, X. S., Tian, K. W., ve Zhai, Y. Q., “Particle swarm optimization with dynamic random population topology strategies for a generalized portfolio selection problem”, *Natural Computing*, 16, 31-44, (2017).

Niu, B., Tan, L., Xue, B., Li, L., ve Chai, Y., “Constrained portfolio selection using multiple swarms”, *In IEEE Congress on Evolutionary Computation* (pp. 1-7): IEEE, (2010).

Niu, B., Xue, B., Li, L., ve Chai, Y., “Symbiotic Multi-swarm PSO for Portfolio Optimization”, *In D.-S. Huang, K.-H. Jo, H.-H. Lee, H.-J. Kang ve V. Bevilacqua (Eds.), Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence: 5th International Conference on Intelligent Computing, ICIC 2009 Ulsan, South Korea, September 16-19, 2009 Proceedings* (pp. 776-784). Berlin, Heidelberg: Springer Berlin Heidelberg, (2009).

Ong, C. S., Huang, H. J., ve Tzeng, G. H., “A novel hybrid model for portfolio selection”, *Applied Mathematics and Computation*, 169, 1195-1210, (2005).

Özceylan, E., Kalayci, C. B., Güngör, A., ve Gupta, S. M., “Disassembly line balancing problem: a review of the state of the art and future directions”, *International Journal of Production Research*, 1-23, (2018).

Pai, G. A. V., ve Michel, T., “Evolutionary Optimization of Constrained k-means Clustered Assets for Diversification in Small Portfolios”, *IEEE Transactions on Evolutionary Computation*, 13, 1030-1053, (2009).



Paiva, F. D., Cardoso, R. T. N., Hanaoka, G. P., ve Duarte, W. M., “Decision-making for financial trading: A fusion approach of machine learning and portfolio selection”, *Expert Systems with Applications*, 115, 635-655, (2019).

Panigrahi, B. K., Pandi, V. R., Sharma, R., Das, S., ve Das, S., “Multiobjective bacteria foraging algorithm for electrical load dispatch problem”, *Energy Conversion and Management*, 52, 1334-1342, (2011).

Passino, K. M., “Biomimicry of bacterial foraging for distributed optimization and control”, *IEEE Control Systems Magazine*, 22, 52-67, (2002).

Peng, H., Kitagawa, G., Gan, M., ve Chen, X. H., “A new optimal portfolio selection strategy based on a quadratic form mean-variance model with transaction costs”, *Optimal Control Applications ve Methods*, 32, 127-138, (2011).

Polat, O., Kalayci, C. B., Kulak, O., ve Gunther, H. O., “A perturbation based variable neighborhood search heuristic for solving the Vehicle Routing Problem with Simultaneous Pickup and Delivery with Time Limit”, *European Journal of Operational Research*, 242, 369-382, (2015<sup>a</sup>).

Polat, O., Kalayci, C. B., Mutlu, Ö., ve Gupta, S. M., “A two-phase variable neighbourhood search algorithm for assembly line worker assignment and balancing problem type-II: an industrial case study”, *International Journal of Production Research*, 54, 722-741, (2015<sup>b</sup>).

Ponsich, A., Jaimes, A. L., ve Coello, C. A. C., “A Survey on Multiobjective Evolutionary Algorithms for the Solution of the Portfolio Optimization Problem and Other Finance and Economics Applications”, *IEEE Transactions on Evolutionary Computation*, 17, 321-344, (2013).

Pouya, A. R., Solimanpur, M., ve Rezaee, M. J., “Solving multi-objective portfolio optimization problem using invasive weed optimization”, *Swarm and Evolutionary Computation*, 28, 42-57, (2016).

Puchinger, J., Raidl, G. R., ve Pferschy, U., “The Core Concept for the Multidimensional Knapsack Problem”, *In (pp. 195-208). Berlin, Heidelberg: Springer Berlin Heidelberg*, (2006).

Qin, Z. F., “Mean-variance model for portfolio optimization problem in the simultaneous presence of random and uncertain returns”, *European Journal of Operational Research*, 245, 480-488, (2015).

Qu, B. Y., Zhou, Q., Xiao, J. M., Liang, J. J., ve Suganthan, P. N., “Large-Scale Portfolio Optimization Using Multiobjective Evolutionary Algorithms and Preselection Methods”, *Mathematical Problems in Engineering*, 2017, 1-14, (2017).

Reid, S. G., ve Malan, K. M., “Constraint handling methods for portfolio optimization using particle swarm optimization”, *In Proceedings - 2015 IEEE Symposium Series on Computational Intelligence, SSCI 2015* (pp. 1766-1773): IEEE, (2015).

Rockafellar, R. T., ve Uryasev, S., “Optimization of conditional value-at-risk”, *Journal of risk*, 2, 21-42, (2000).

Rong, X., Lu, M., ve Deng, L., “Multi-period model of portfolio investment and adjustment based on hybrid genetic algorithm”, *Transactions of Tianjin University*, 15, 415-422, (2009).

Rubinstein, M., “Markowitz’s “Portfolio Selection”: A Fifty-Year Retrospective”, *The Journal of Finance*, 57, 1041-1045, (2002).

Ruiz-Torrobiano, R., ve Suarez, A., “Hybrid Approaches and Dimensionality Reduction for Portfolio Selection with Cardinality Constraints”, *IEEE Computational Intelligence Magazine*, 5, 92-107, (2010).

Ruiz-Torrobiano, R., ve Suarez, A., “A memetic algorithm for cardinality-constrained portfolio optimization with transaction costs”, *Applied Soft Computing*, 36, 125-142, (2015).

Ruiz-Torrobiano, R., ve Suárez, A., “Use of heuristic rules in evolutionary methods for the selection of optimal investment portfolios”, *In 2007 IEEE Congress on Evolutionary Computation, CEC 2007* (pp. 212-219): IEEE, (2007).

Sabar, N. R., ve Song, A., “Dual Population Genetic Algorithm for the Cardinality Constrained Portfolio Selection Problem”, *In (pp. 703-712). Cham: Springer International Publishing*, (2014).

Sadigh, A. N., Mokhtari, H., Iranpoor, M., ve Ghomi, S. M. T. F., “Cardinality Constrained Portfolio Optimization Using a Hybrid Approach Based on Particle Swarm Optimization and Hopfield Neural Network”, *Advanced Science Letters*, 17, 11-20, (2012).

Sadjadi, S. J., Gharakhani, M., ve Safari, E., “Robust optimization framework for cardinality constrained portfolio problem”, *Applied Soft Computing*, 12, 91-99, (2012).

Schaerf, A., “Local Search Techniques for Constrained Portfolio Selection Problems”, *Computational Economics*, 20, 177-190, (2002).

Schaffer, J. D., “Multiple objective optimization with vector evaluated genetic algorithms”, In *Proceedings of the 1st international Conference on Genetic Algorithms* (pp. 93-100): L. Erlbaum Associates Inc, (1985).

Sen, T., Saha, S., Ekbal, A., ve Laha, A. K., “Bi-objective portfolio optimization using archive multi-objective simulated annealing”, In *2014 International Conference on High Performance Computing and Applications, ICHPCA 2014: IEEE*, (2015).

Shaikh, R. A., ve Abbas, A., “Genetic algorithm and MS solver for portfolio optimization under exogenous influence”, In *2009 International Conference on Computer and Electrical Engineering, ICCEE 2009* (Vol. 1, pp. 555-558), (2009).

Sharpe, W. F., “Mutual Fund Performance”, *The Journal of Business*, 39, 119-138, (1966).

Shaw, D. X., Liu, S. C., ve Kopman, L., “Lagrangian relaxation procedure for cardinality-constrained portfolio optimization”, *Optimization Methods ve Software*, 23, 411-420, (2008).

Shoaf, J., ve Foster, J. A., “Efficient set GA for stock portfolios”, In *Proceedings of the IEEE Conference on Evolutionary Computation, ICEC* (pp. 354-359): IEEE, (1998).

Skolpadungket, P., Dahal, K., ve Harnpornchai, N., “Portfolio optimization using multi-objective genetic algorithms”, In *2007 IEEE Congress on Evolutionary Computation, CEC 2007* (pp. 516-523): IEEE, (2007).

Soleimani, H., Golmakani, H. R., ve Salimi, M. H., “Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm”, *Expert Systems with Applications*, 36, 5058-5063, (2009).

Storn, R., ve Price, K., “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces”, *Journal of Global Optimization*, 11, 341-359, (1997).

Streichen, F., ve Tanaka-Yamawaki, M., “The effect of local search on the constrained portfolio selection problem”, *In 2006 IEEE Congress on Evolutionary Computation, CEC 2006* (pp. 2368-2374): IEEE, (2006).

Streichert, F., Ulmer, H., ve Zell, A., “Evaluating a hybrid encoding and three crossover operators on the constrained portfolio selection problem”, *In Evolutionary Computation, 2004. CEC2004. Congress on* (Vol. 1, pp. 932-939 Vol.931), (2004<sup>a</sup>).

Streichert, F., Ulmer, H., ve Zell, A., “Evolutionary algorithms and the cardinality constrained portfolio optimization problem”, *In Operations Research Proceedings 2003* (Vol. 2003, pp. 253-260): Springer, (2004<sup>b</sup>).

Strumberger, I., Bacanin, N., ve Tuba, M., “Constrained Portfolio Optimization by Hybridized Bat Algorithm”, *In 7th International Conference on Intelligent Systems, Modelling and Simulation* (pp. 83-88): IEEE, (2017).

Suganya, N. C., ve Vijayalakshmi Pai, G. A., “Constrained portfolio rebalancing with transaction costs using evolutionary wavelet hopfield network strategy”, *In 2009 World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings* (pp. 451-456), (2009).

Sun, J., Fang, W., Wu, X. J., Lai, C. H., ve Xu, W. B., “Solving the multi-stage portfolio optimization problem with a novel particle swarm optimization”, *Expert Systems with Applications*, 38, 6727-6735, (2011).

Suthiwong, D., ve Sodanil, M., “Cardinality-constrained Portfolio optimization using an improved quick Artificial Bee Colony Algorithm”, *In Computer Science and Engineering Conference (ICSEC), 2016 International* (pp. 1-4): IEEE, (2016).

Talebi, A., Molaei, M. A., ve Sheikh, M. J., “Performance investigation and comparison of two evolutionary algorithms in portfolio optimization: Genetic and particle swarm optimization”, *In Proceedings - 2010 2nd IEEE International Conference on Information and Financial Engineering, ICIFE 2010* (pp. 430-437), (2010).

Tan, L., Niu, B., Lin, F., Duan, Q., ve Li, L., “Modified Bacterial Foraging Optimization for Constrained Portfolio Optimization”, *Information Technology Journal*, 12, 7918-7921, (2013).

Tan, L., Niu, B., Wang, H., Huang, H., ve Duan, Q., “Bacterial Foraging Optimization with Neighborhood Learning for Dynamic Portfolio Selection”, *In (Vol. 8590, pp. 413-423). Cham: Springer International Publishing, (2014).*

Tan, Y., ve Zhu, Y., “Fireworks Algorithm for Optimization”, *In Y. Tan, Y. Shi ve K. C. Tan (Eds.), Advances in Swarm Intelligence: First International Conference, ICSI 2010, Beijing, China, June 12-15, 2010, Proceedings, Part I (pp. 355-364). Berlin, Heidelberg: Springer Berlin Heidelberg, (2010).*

Tang, J., Wang, K., ve Zhao, X., “A novel constraint handling technique for complex portfolio selection”, *In 6th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2009 (Vol. 2, pp. 156-160): IEEE, (2009).*

Thomaidis, N. S., “Active Portfolio Management from a Fuzzy Multi-objective Programming Perspective”, *In C. Chio, A. Brabazon, G. A. Caro, M. Ebner, M. Farooq, A. Fink, J. Grahl, G. Greenfield, P. Machado, M. O’Neill, E. Tarantino ve N. Urquhart (Eds.), Applications of Evolutionary Computation: EvoApplications 2010: EvoCOMNET, EvoENVIRONMENT, EvoFIN, EvoMUSART, and EvoTRANSLOG, Istanbul, Turkey, April 7-9, 2010, Proceedings, Part II (pp. 222-231). Berlin, Heidelberg: Springer Berlin Heidelberg, (2010).*

Tian, Y., Fang, S., Deng, Z. B., ve Jin, Q. W., “Cardinality Constrained Portfolio Selection Problem: A Completely Positive Programming Approach”, *Journal of Industrial and Management Optimization*, 12, 1041-1056, (2016).

Tuba, M., ve Bacanin, N., “Artificial Bee Colony Algorithm Hybridized with Firefly Algorithm for Cardinality Constrained Mean-Variance Portfolio Selection Problem”, *Applied Mathematics ve Information Sciences*, 8, 2831-2844, (2014<sup>a</sup>).

Tuba, M., ve Bacanin, N., “Upgraded firefly algorithm for portfolio optimization problem”, *In Proceedings - UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, UKSim 2014 (pp. 113-118): IEEE, (2014b).*

Tuba, M., Bacanin, N., ve Pelevic, B., “Framework for constrained portfolio selection by the firefly algorithm”, *International Journal of Mathematical Models and Methods in Applied Sciences*, 7, 888-896, (2013).

Tuba, M., Bacanin, N., ve Pelevic, B., “Krill Herd (KH) algorithm applied to the constrained portfolio selection problem”, *International Journal of Mathematics and Computers in Simulation*, 8, 94-102, (2014).

Van Veldhuizen, D. A., ve Lamont, G. B., “Multiobjective evolutionary algorithms: Analyzing the state-of-the-art”, *Evolutionary computation*, 8, 125-147, (2000).

Vijayalakshmi Pai, G. A., ve Michel, T., “Fuzzy decision theory based optimization of constrained portfolios using metaheuristics”, In *IEEE International Conference on Fuzzy Systems: IEEE*, (2013).

Wang, J.-b., Chen, W.-N., Zhang, J., ve Lin, Y., “A Dimension-Decreasing Particle Swarm Optimization Method for Portfolio Optimization”, In *Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference - GECCO Companion '15* (pp. 1515-1516). Madrid, Spain: ACM, (2015).

Wang, W., Wang, H., Wu, Z., ve Dai, H., “A Simple and Fast Particle Swarm Optimization and Its Application on Portfolio Selection”, In *Intelligent Systems and Applications, 2009. ISA 2009. International Workshop on* (pp. 1-4): IEEE, (2009).

Wang, Z., Liu, S., ve Kong, X., “An Improved Artificial Bee Colony Algorithm for Portfolio Optimization Problem”, *International Journal of Advancements in Computing Technology*, 3, 67-74, (2011).

Wang, Z., Liu, S., ve Kong, X., “Artificial Bee Colony Algorithm for Portfolio Optimization Problems”, *International Journal of Advancements in Computing Technology*, 4, 8-16, (2012).

Woodside-Oriakhi, M., Lucas, C., ve Beasley, J. E., “Heuristic algorithms for the cardinality constrained efficient frontier”, *European Journal of Operational Research*, 213, 538-550, (2011).

Xia, Y. S., Liu, B. D., Wang, S. Y., ve Lai, K. K., “A model for portfolio selection with order of expected returns”, *Computers and Operations Research*, 27, 409-422, (2000).

Xidonas, P., ve Mavrotas, G., “Multiobjective portfolio optimization with non-convex policy constraints: Evidence from the Eurostoxx 50”, *European Journal of Finance*, 20, 957-977, (2014).

Xu, F., ve Chen, W., “Stochastic portfolio selection based on velocity limited particle swarm optimization”, In *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on* (Vol. 1, pp. 3599-3603): IEEE, (2006).

Xu, F., Chen, W., ve Yang, L., “Improved particle swarm optimization for realistic portfolio selection”, *In Proceedings - SNPD 2007: Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing* (Vol. 1, pp. 185-190): IEEE, (2007).

Xu, J., Lam, A. Y. S., ve Li, V. O. K., “Stock portfolio selection using chemical reaction optimization”, *World Academy of Science, Engineering and Technology*, 5, 458-463, (2011).

Xu, R. T., Zhang, J., Liu, O., ve Huang, R. Z., “An estimation of distribution algorithm based portfolio selection Approach”, *In Proceedings - International Conference on Technologies and Applications of Artificial Intelligence*, TAAI 2010 (pp. 305-313): IEEE, (2010).

Yaakob, S. B., ve Watada, J., “A hybrid particle swarm optimization approach to mixed integer quadratic programming for portfolio selection problems”, *International Journal of Simulation: Systems, Science and Technology*, 11, 68-74, (2010).

Yang, X.-S., “A New Metaheuristic Bat-Inspired Algorithm”, *In J. R. González, D. A. Pelta, C. Cruz, G. Terrazas ve N. Krasnogor (Eds.), Nature inspired cooperative strategies for optimization (NICSO 2010)* (Vol. 284, pp. 65-74). Berlin, Heidelberg: Springer Berlin Heidelberg, (2010<sup>a</sup>).

Yang, X. S., “Engineering optimization: an introduction with metaheuristic applications”, *John Wiley ve Sons*, (2010<sup>b</sup>).

Yin, X., Ni, Q., ve Zhai, Y., “A Novel Particle Swarm Optimization for Portfolio Optimization Based on Random Population Topology Strategies”, *In* (pp. 164-175). Cham: Springer International Publishing, (2015<sup>a</sup>).

Yin, X., Ni, Q., ve Zhai, Y., “A novel PSO for portfolio optimization based on heterogeneous multiple population strategy”, *In 2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings* (pp. 1196-1203): IEEE, (2015<sup>b</sup>).

Yoshimoto, A., “The Mean-Variance Approach To Portfolio Optimization Subject To Transaction Costs”, *Journal of the Operations Research Society of Japan*, 39, 99-117, (1996).

Young, M. R., “A minimax portfolio selection rule with linear programming solution”, *Management Science*, 44, 673-683, (1998).

Yu, L., Wang, S. Y., ve Lai, K. K., “Multi-Attribute Portfolio Selection with Genetic Optimization Algorithms”, *INFOR*, 47, 23-30, (2009).

Zaheer, H., ve Pant, M., “Solving portfolio optimization problem through differential evolution”, *In International Conference on Electrical, Electronics, and Optimization Techniques, ICEEOT 2016* (pp. 3982-3987): IEEE, (2016).

Zhang, H., Zhao, Y., Wang, F., Zhang, A., Yang, P., ve Shen, X., “A new evolutionary algorithm based on MOEA/D for portfolio optimization”, *In Advanced Computational Intelligence (ICACI), 2018 Tenth International Conference on* (pp. 831-836): IEEE, (2018<sup>a</sup>).

Zhang, X. L., Zhang, W. G., Xu, W. J., ve Xiao, W. L., “Possibilistic Approaches to Portfolio Selection Problem with General Transaction Costs and a CLPSO Algorithm”, *Computational Economics*, 36, 191-200, (2010).

Zhang, Y., Li, X., ve Guo, S., “Portfolio selection problems with Markowitz’s mean–variance framework: a review of literature”, *Fuzzy Optimization and Decision Making*, 17, 125-158, (2018<sup>b</sup>).

Zhou, J., ve Li, J., “An improved multi-objective particle swarm optimization for constrained portfolio selection model”, *In 11th International Conference on Service Systems and Service Management, ICSSSM 2014 - Proceeding: IEEE*, (2014).

Zhou, Z., Liu, X., Xiao, H., Wu, S., ve Liu, Y., “A DEA-based MOEA/D algorithm for portfolio optimization”, *Cluster Computing*, (2018).

Zhu, H., Chen, Y., ve Wang, K., “Swarm Intelligence Algorithms for Portfolio Optimization”, *In Y. Tan, Y. Shi ve K. C. Tan (Eds.), Advances in Swarm Intelligence: First International Conference, ICSI 2010, Beijing, China, June 12-15, 2010, Proceedings, Part I* (Vol. 6145, pp. 306-313). Berlin, Heidelberg: Springer Berlin Heidelberg, (2010).

Zhu, H. H., Wang, Y., Wang, K. S., ve Chen, Y., “Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem”, *Expert Systems with Applications*, 38, 10161-10169, (2011).

Zitzler, E., “Evolutionary algorithms for multiobjective optimization: Methods and applications”, (Vol. 63): Citeseer, (1999).



Zitzler, E., ve Künzli, S., “Indicator-based selection in multiobjective search”, *In International Conference on Parallel Problem Solving from Nature* (Vol. 3242, pp. 832-842): Springer, (2004).

Zitzler, E., Laumanns, M., Thiele, L., Zitzler, E., Zitzler, E., Thiele, L., ve Thiele, L., “SPEA2: Improving the strength Pareto evolutionary algorithm”, *In: Eidgenössische Technische Hochschule Zürich (ETH), Institut für Technische Informatik und Kommunikationsnetze (TIK)*, (2001).

Zitzler, E., ve Thiele, L., “Multiobjective evolutionary algorithms: A comparative case study and the Strength Pareto Approach”, *IEEE Transactions on Evolutionary Computation*, 3, 257-271, (1999).

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., ve Da Fonseca, V. G., “Performance assessment of multiobjective optimizers: An analysis and review”, *IEEE Transactions on Evolutionary Computation*, 7, 117-132, (2003).

## 6. ÖZGEÇMİŞ

Adı Soyadı : Mehmet Anıl AKBAY

Doğum Yeri ve Tarihi : AFYONKARAHİSAR 18/07/1991

Lisans Üniversite : KOCAELİ ÜNİVERSİTESİ

Elektronik posta : m.anil.akbay@gmail.com

İletişim Adresi : Mimar Sinan Mah. Çömezler Cad. No:6  
Başmakçı/Afyonkarahisar

Yayın Listesi :

- Kalayci, C. B., Ertenlice, O., ve Akbay, M. A. (2019). A comprehensive review of deterministic models and applications for mean-variance portfolio optimization. *Expert Systems with Applications*, 125, 345-368.