



**GEZGİN SATICI PROBLEMİNİN ÇÖZÜMÜNE
YÖNELİK PARALEL ADA MODELLİ
HİBRİT KARADUL ÖRÜMCEK ALGORİTMASI**
Yüksek Lisans Tezi

Kardelen Bahar ERDEN

Eskişehir 2025

**GEZGİN SATICI PROBLEMİNİN ÇÖZÜMÜNE YÖNELİK PARALEL ADA
MODELLİ HİBRİT KARADUL ÖRÜMCEK ALGORİTMASI**

Kardelen Bahar ERDEN

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Gülçin DİNÇ YALÇIN

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Aralık 2025

JÜRİ VE ENSTİTÜ ONAYI

Kardelen Bahar ERDEN'in "GEZGİN SATICI PROBLEMİNİN ÇÖZÜMÜNE YÖNELİK PARALEL ADA MODELLİ HİBRİT KARADUL ÖRÜMCEK ALGORİTMASI" başlıklı tezi 19/12/2025 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği'nin ilgili maddeleri uyarınca, Endüstri Mühendisliği Anabilim dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Jüri Üyeleri

Unvanı Adı Soyadı

İmza

Üye

: Doç. Dr. Gülçin DİNÇ YALÇIN

Üye

: Prof. Dr. Refail KASIMBEYLİ

Üye

: Dr. Öğr. Üyesi İslam ALTIN

Prof. Dr. Harun BÖCÜK

Lisansüstü Eğitim Enstitüsü Müdürü

19/12/2025

DANIřMAN ONAYI

Daniřmanlıđını yrttđm Yksek Lisans đrencisi Kardelen Bahar ERDEN, GEZGİN SATICI PROBLEMİNİN ÇÖZÜMNE YNELİK PARALEL ADA MODELİ HİBRİT KARADUL RMCEK ALGORİTMASI bařlıklı tez alıřmasını tamamlamıřtır. Hazırlamıř olduđu tez tarafımca incelenmiř ve đrencinin tez savunma sınavına alınması bilimsel ve etik aıdan uygun grlmřtr.

Tez Daniřmanı

Do. Dr. Glin DİNÇ YALÇIN

ÖZET

GEZGİN SATICI PROBLEMİNİN ÇÖZÜMÜNE YÖNELİK PARALEL ADA MODELLİ HİBRİT KARADUL ÖRÜMCEK ALGORİTMASI

Kardelen Bahar ERDEN

Endüstri Mühendisliği Anabilim Dalı

Eskişehir Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü, Aralık 2025

Danışman: Doç. Dr. Gülçin DİNÇ YALÇIN

Kombinatoriyel eniyileme alanının temel problemlerinden biri olan Gezgin Satıcı Problemi (GSP), NP-Zor yapısı ve karmaşık çözüm uzayı nedeniyle araştırmacıların odak noktası olmaya devam etmektedir. Bu çalışmada, GSP'nin etkin çözümü amacıyla Karadul Örümcek Algoritması (BWO) temel alınarak geliştirilmiş Paralel Ada Modelli Karadul Örümcek Optimizasyonu (PH-BWO) önerilmektedir. Önerilen yöntem; genetik operatörler, ölçeğe duyarlı parametre sistemi ve dinamik yerel arama stratejileri ile desteklenerek temel sınırlılıkları aşmayı hedeflemektedir. Algoritmanın performansı, farklı ölçeklerdeki TSPLIB veri setleri üzerinde test edilmiştir. Elde edilen bulgular, PH-BWO'nun özellikle küçük ve orta ölçekli problemlerde bilinen en iyi değerlere istikrarlı bir şekilde ulaştığını ve büyük ölçekli problemlerde kararlı yapı sergilemiş ve bilinen en iyi değere düşük sapma oranlarıyla yaklaşmıştır. Yapılan analizlerde, yöntemin literatürdeki Hybrid-BWO ile Seçim Stratejisi ve Diferansiyel Mutasyon Uyarlamalı Kara Dul Örümcek Optimizasyonu (SDABWO) modellerinden hız ve çözüm kalitesi bakımından üstün olduğu doğrulanmıştır. Bu çalışma Türkçe literatürde GSP için BWO hibritini bütüncül bir çerçevede ele alan özgün bir yaklaşım sunmakta ve ileriki çalışmalar için teorik zemin oluşturmaktadır.

Anahtar Sözcükler: Gezgin Satıcı Problemi, Karadul Örümceği Optimizasyonu, Ada Modeli, Genetik Algoritma, Metasezgisel Yöntemler.

ABSTRACT

PARALLEL ISLAND MODEL BASED HYBRID BLACK WIDOW ALGORITHM FOR SOLVING THE TRAVELING SALESMAN PROBLEM

Kardelen Bahar ERDEN

Department of Industrial Engineering

Eskişehir Technical University, Institute of Graduate Programs, December, 2025

Supervisor: Assoc. Prof. Dr. Gülçin DİNÇ YALÇIN

The Traveling Salesman Problem (TSP), one of the fundamental problems in the field of combinatorial optimization, remains a focal point for researchers due to its NP-Hard structure and complex solution space. In this study, a Parallel Island Model Based Hybrid Black Widow Optimization (PH-BWO) algorithm, developed based on the Black Widow Optimization (BWO) algorithm, is proposed for the effective solution of the TSP. The proposed method aims to overcome fundamental limitations by being supported by genetic operators, a scale-sensitive parameter system, and dynamic local search strategies. Through the island architecture designed in the study, exploration is ensured, while the cannibalism mechanism has been refunctionalized as a directed selection pressure, thereby enhancing the algorithm's exploitation capability. The performance of the algorithm was tested on TSPLIB datasets of various scales. The findings indicate that PH-BWO consistently reaches the best-known values, particularly in small and medium-scale problems. In large-scale problems, it exhibited a stable structure and approached the best-known values with low deviation rates. Analyses confirmed that the method is superior in terms of speed and solution quality to the Hybrid-BWO and SDABWO models found in the literature. This study presents an original approach addressing the BWO hybrid for TSP within a holistic framework in the Turkish literature and establishes a theoretical foundation for future studies.

Keywords: Traveling Salesman Problem, Black Widow Optimization, Island Model, Genetic Algorithm, Metaheuristic Algorithms.

TEŞEKKÜR

Bu tez çalışması boyunca bana bilgileri ile yol gösteren saygıdeğer danışmanım Doç. Dr. Gülçin DİNÇ YALÇIN'a en içten teşekkürlerimi sunarım. Ayrıca, tez savunması sırasında değerli değerlendirmeleriyle çalışmamın kalitesini artıran saygıdeğer jüri üyelerine teşekkürlerimi sunarım. Bu süreçte benden desteklerini esirgemeyen sevgili aileme, özellikle kız kardeşim Hilal Nur ERDEN'e teşekkür ederim.

Kardelen Bahar ERDEN

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan “bilimsel intihal tespit programıyla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Kardelen Bahar ERDEN

İÇİNDEKİLER

BAŞLIK SAYFASI	i
JÜRİ VE ENSTİTÜ ONAYI.....	ii
DANIŞMAN ONAYI	iii
ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ.....	vii
İÇİNDEKİLER	viii
TABLolar DİZİNİ.....	xi
ŞEKİLLER DİZİNİ.....	xii
SİMGELER VE KISALTMALAR DİZİNİ	xiii
1. GİRİŞ	1
2. GEZGİN SATICI PROBLEMİ	7
2.1. Gezgin Satıcı Probleminin Tarihçesi.....	8
2.2. Gezgin Satıcı Probleminin Tanımı ve Matematiksel Modeli.....	11
2.3. Gezgin Satıcı Probleminin Sınıflandırılması.....	13
2.4. Gezgin Satıcı Problemi için Kullanılan Çözüm Yöntemleri	14
2.4.1. Kesin yöntemler.....	16
2.4.2. Sezgisel yöntemler	16
2.4.3. Metasezgisel yöntemler	17
2.4.3.1. Metasezgisel algoritmaların tasarımı ve uygulanması.....	20
2.4.3.2. Metasezgisel algoritmaların sınıflandırılmasına ilişkin literatür araştırması.....	21

3. GENETİK ALGORİTMA	25
3.1. Genetik Algoritmaların Kuramsal Temeli.....	25
3.2. Genetik Algoritmanın Temel Kavramları.....	27
3.3. Genetik Algoritma Operatörleri.....	29
3.4. Genetik Algoritmanın Çalışma Prensibi.....	30
3.5. Genetik Algoritmanın Gezgin Satıcı Problemine Uyarlanması.....	32
4. KARADUL ÖRÜMCEK ALGORİTMASI.....	36
4.1. Karadul Örümceği Optimizasyon Algoritması'nın Çalışma Prensibi.....	38
4.2. Parametrelerin Ayarlanması.....	41
5. PARALEL ADA MODELLİ HİBRİT KARADUL ÖRÜMCEK OPTİMİZASYONU (PH-BWO)	43
5.1. Amaç ve Kapsam	43
5.2. Algoritma Bileşenlerinin Seçilimi ve Nedenleri.....	45
5.3. Parametrelerin Belirlenmesi ve Etkileşimleri.....	49
5.4. Parametrelerin Etkileşimi.....	52
5.5. Önerilen Modelin Çalışma Prensibi.....	55
6. DENEYSEL ÇALIŞMALAR.....	59
6.1. Parametre Ayarlarının Belirlenmesi.....	60
6.1.1. Küçük Ölçekli Veri Seti Pilot Çalışma Sonuçları.....	62
6.1.2. Orta Ölçekli Veri Seti Pilot Çalışma Sonuçları	64
6.1.3. Büyük Ölçekli Veri Seti Pilot Çalışma Sonuçları	67
6.1.4. Ön Ayar Parametre Setlerinin Belirlenmesi	69
6.2. Deneylerde Kullanılan Tekrar Sayısı İçin İstatistiksel Analiz.....	75
6.3. Küçük Ölçekli Veri Seti Çalışma Sonuçlarının Değerlendirilmesi	80

6.4. Orta Ölçekli Veri Seti Çalışma Sonuçlarının Değerlendirilmesi	83
6.5. Büyük Ölçekli Veri Seti Çalışma Sonuçlarının Değerlendirilmesi	87
6.6. Tüm ölçeklerin genel karşılaştırması ve toplu değerlendirme.....	89
6.7. Karşılaştırmalı Performans Analizi ve Kıyaslama.....	91
7. SONUÇLAR VE ÖNERİLER	99
KAYNAKÇA.....	102
EKLER	
ÖZGEÇMİŞ	

TABLÖLAR DİZİNİ

Sayfa

Tablo 1.1 Tur Sayısının Hesaplanma Yöntemleri.....	3
Tablo 2.1 Tur Sayıları Ve Zaman Karmaşıklığı Hesapları	7
Tablo 2.2. GSP Kilometre Taşı Çalışmalar	10
Tablo 2.3 GSP Ölçeğe Göre Sınıflandırma	14
Tablo 3.1 GA'nın Çalışma Prensiplerinin Altında Yatan Evrimsel Kavramlar	28
Tablo 4.1 Karadul Örümcek Optimizasyonu Sözde Kodu.....	39
Tablo 6.1 Pilot Çalışma Esnasında Kullanılan Parametreler	61
Tablo 6.2 Küçük Ölçekli Veri Seti Pilot Çalışma Sonuçları.....	62
Tablo 6.3 Orta Ölçekli Veri Seti Pilot Çalışma Sonuçları	65
Tablo 6.4 Büyük Ölçekli Veri Seti Pilot Çalışma Sonuçları.....	67
Tablo 6.5 Küçük Ölçekli Problemler için Ön Ayarlı Parametre Seti.....	71
Tablo 6.6 Orta Ölçekli Problemler için Ön Ayarlı Parametre Seti	73
Tablo 6.7 Büyük Ölçekli Problemler için Ön Ayarlı Parametre Seti.....	74
Tablo 6.8 Küçük Ölçekli Problemler İçin PH-BWO Sonuçları	81
Tablo 6.9 Orta Ölçekli Problemler İçin PH-BWO Sonuçları	84
Tablo 6.10 Büyük Ölçekli Problemler İçin PH-BWO Sonuçları	87
Tablo 6.11 Küçük Ölçekli Problemler PH-BWO ile Hybrid BWO Karşılaştırması.....	92
Tablo 6.12 Orta ölçekli Problemler PH-BWO ile Hybrid BWO Karşılaştırması	93
Tablo 6.13 PH-BWO ile SDABWO Süre Karşılaştırılması.....	96

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1. Eniyileme Problemleri Çözüm Yöntemlerinin Sınıflandırması	15
Şekil 2.3. İlham Kaynağına Göre Metasezgisel Algoritmaların Sınıflandırılması	23
Şekil 5.1. PH-BWO'nun Akış Şeması	55
Şekil 6.1. Küçük Ölçekli Problemlerde Pilot Çalışma Süre Farkı Sonuçları	63
Şekil 6.2. Çözümün Bulunduğu İterasyon Sayısının Problem Bazında Gösterimi	66
Şekil 6.3. Büyük Ölçekli Problemlerde Bilinen En İyiden Ortalama Sapma Yüzdeleri	68
Şekil 6.4. Küçük Ölçekli Problemlerde Ön Ayar Setinde Süre Davranışları	82
Şekil 6.5. İlk Grup İçin Ön Ayarlı Set Analizinde Süre Farkları	84
Şekil 6.6. İkinci Grup İçin Ön Ayarlı Set Analizinde Nesil ve Süre Farkları	85
Şekil 6.7. Son Grup İçin Ön Ayarlı Setlerde Süre ve Nesil Analizi	86
Şekil 6.8. Büyük Ölçekli Problemler İçin Ön Ayar Setleri Süre Değerler	88
Şekil 6.9. Ön Ayar Setleri Bilinen En İyiden Sapma Yüzdeleri	88
Şekil 6.10. PH-BWO ile SDABWO Yakınsama Süresi Karşılaştırma	97

SİMGELER VE KISALTMALAR DİZİNİ

ABC	: Yapay Arı Kolonisi Optimizasyonu
ABC-ACS	: Yapay Arı Kolonisi/Karınca Kolonisi Hibrit Optimizasyonu Algoritması
ABO	: Afrika Bufalo Optimizasyonu
ACO	: Karınca Kolonisi Optimizasyonu
BBO	: Biyocoğrafya Tabanlı Optimizasyonu
BWO	: Karadul Örümceği Optimizasyonu Algoritması
DFJ	: Dantzig, Fulkerson ve Johnson
GA	: Genetik Algoritma
GSP	: Gezgin Satıcı Problemi
GWO	: Gri Kurt Optimizasyonu
HPSACO	: Hibrit Parçacık Sürüsü/Karınca Kolonisi Optimizasyonu Algoritması
LKH	: Lin-Kernighan Sezgiseli
MA	: Memetik Algoritmalar
NN	: En Yakın Komşu Sezgiseli
PH-BWO	: Paralel Ada Modelli Hibrit Karadul Örümcek Optimizasyonu
PSO	: Parçacık Sürü Optimizasyonu
SA	: Tavlama Benzetimi
SDABWO	: Seçim Stratejisi ve Diferansiyel Mutasyonlu Uyarlamalı Kara Dul Örümcek Optimizasyonu
VND	: Değişken Komşuluk İnişi

WOA : Balina Optimizasyonu Algoritması



1. GİRİŞ

Eniyileme¹, sınırlayıcı koşullar altında karar değişkenlerinin en iyi değer kombinasyonlarını belirlemeyi amaçlamakta ve bu yönüyle mühendislikten ekonomiye, yapay zekadan lojistiğe kadar birçok disiplinde temel bir çözüm yaklaşımı sunmaktadır. Mühendislik biliminde kullanılan eniyileme problemleri belirli bir amaç doğrultusunda, tanımlı bir çözüm uzayında en iyi çözümün elde edilmesini hedefleyen matematiksel bir yapıdır. “Bir eniyileme modeli, verilen kısıtları sağlayan karar değişkenlerinin tüm olası değerleri arasında, amaç fonksiyonunu en iyileyecek (maksimize veya minimize) karar değişkeni değerlerini bulmayı amaçlamaktadır” (Winston ve Goldberg, 2004, s. 2).

Eniyileme problemleri, kaynakların etkin kullanımı, maliyetin azaltılması ve verimliliğin artırılması gibi hedefler doğrultusunda ulaştırma, enerji, üretim planlaması, finansal portföy optimizasyonu, lojistik ve sağlık hizmetleri gibi gerçek hayat problemlerine uyarlanabilmektedir. Bu problemler farklı olasılıklar arasında en uygun yapılandırmayı veya değeri bulmaya odaklanmaktadır. Kesikli değişkenlere sahip eniyileme problemleri, kombinatoriyel eniyileme problemleri olarak tanımlanır ve sürekli değişkenli problemlere kıyasla çözüm yapıları bakımından farklılık göstermektedir (Peres ve Castelli, 2021, s. 1).

Eniyileme problemlerinde, değişken sayısının artması ile problemin daha da karmaşıklaşması durumunda çözümün bulunması giderek zorlaşmaktadır. Tüm çözüm uzayında en iyi sonucu sağlayan çözüm küresel en iyi çözüm (global optimum) iken, yerel en iyi (local optimum) çözüm ise çözüm uzayının yalnızca bir alt bölgesinde bulunan en iyi çözüm noktasıdır (Kayman, 2015, s. 5).² Amaç fonksiyonundaki çok sayıda yerel en iyi değer, küresel en iyiye ulaşmayı güçleştirmektedir. Küresel en iyiye ulaşmada başlıca zorluklar; yerel en iyiden kaçınma, daha iyi çözümler bulma ve yakınsama kriterlerinin belirlenmesidir.

¹ Çalışmada optimizasyon terimi yerine çalışmada eniyileme terimi kullanılmış olup, yalnızca terim olarak kullanıldığı alanlarda ‘optimizasyon’ olarak verilmiştir.

² Bu çalışmada "global optimum" ve "yerel optimum" terimleri, "en iyi" ifadesinin genel anlamından farklı olarak, belirli bir amaç fonksiyonu ve kısıtlar dahilinde matematiksel olarak tanımlanan en iyi çözümü belirtmek için kullanılmaktadır. Bu kullanım, ilgili literatürdeki terminolojiye uygunluk ve kavramsal hassasiyetin korunması amacıyla benimsenmiştir. Çalışmanın devamında, global optimum yerine küresel en iyi ve local optimum yerine yerel en iyi kavramları kullanılacaktır.

Küresel eniyileme, bir amaç fonksiyonunun tanımlı olduğu tüm çözüm uzayı içerisindeki küresel en iyi değeri bulma süreci olarak tanımlanmaktadır. Ancak, özellikle karmaşık problemler için küresel en iyinin kesin olarak bulunmasının her zaman mümkün olmaması, yaklaşık çözümler üreten yöntemlerin önemini artırmıştır. Bu doğrultuda geliştirilen algoritmalar, küresel en iyiyi garanti etmek yerine, makul bir hesaplama süresi içerisinde tatmin edici sonuçlar elde etmeyi hedeflemektedir (İbrahim, 2020, s. 1).

Gezgin Satıcı Problemi (GSP) en çok çalışılan kombinatoriyel eniyileme problemlerinden biridir. “Gezgin satıcı problemi en basit hali ile belirli bir şehirden (düğünden) yola çıkan bir satıcının, kendisine verilen bir şehirler grubunun her birini tam olarak bir kez ziyaret ettikten sonra başlangıç noktasına geri döneceği en kısa turu bulmak olarak tanımlanabilir” (Dantzig, Fulkerson ve Johnson, 1954, s. 393).³ Tanımlanması oldukça kolay bir problem olup uygulama alanı çok geniştir. “Günlük duraklarının tamamını kapsayacak en uygun güzergahı belirlemeye çalışan bir posta dağıtıcısından, yüzlerce bilgisayarı birbirine bağlayacak en verimli halka topolojisini tasarlamaya çalışan bir ağ mimarına kadar çeşitli örneklerde bu problem karşımıza çıkmaktadır” (Zambito, 2006, s. 1). Bunlar dışında birçok alanda kullanılmakta ve araştırılmaya devam edilmektedir.

Gezgin satıcı problemi hem teorik bilgisayar bilminde hem de pratik uygulamalarda büyük öneme sahiptir; çünkü problem kolayca formüle edilmesine rağmen, çözümü özellikle düğüm sayısı arttıkça büyük oranda karmaşıklaşmaktadır. GSP’nin en iyi çözümünün bulunması genellikle yüksek hesaplama gücü gerektirmektedir. Bu noktada, GSP’nin hesaplanabilirliğini daha iyi anlamak için öncelikle problemin ait olduğu karmaşıklık sınıfını tanımlamak gerekmektedir.

P (Polynomial time) sınıfı, deterministik Turing makineleri tarafından polinom zamanında çözülebilen problemlerin oluşturduğu ve genel olarak etkin bir şekilde çözülebilir problemler olarak kabul edilen bir problem sınıfıdır. P sınıfının üst kümesi olan NP (Nondeterministic Polynomial time) sınıfı ise, deterministik olmayan Turing makineleri tarafından polinom zamanda çözülebilen veya doğrulanabilen problemleri kapsamaktadır.

³ Bu çalışmada kullanılan kaynakların İngilizce’den Türkçe’ye çevirisi çalışmanın yazarı tarafından yapılmıştır.

Gezgin satıcı problemi, karar problemine indirgenebilir ve bu haliyle verilen bir turun maliyetinin belirli bir eşik değerden düşük olup olmadığı polinom zamanda doğrulanabilir. GSP, NP olarak sınıflandırılmakta ve NP problemlerinin çözülebilirliği, polinom zamanda çözülebilirliğin ötesinde doğrulanabilirlik üzerine temellenmektedir (Štencek, 2013, s. 8). Ayrıca, Hamilton turu problemi gibi NP-tam problemlere polinom zamanda indirgenebilmesi nedeniyle, GSP NP-Zor ve NP sınıfının kesişiminde yer alan NP-tam bir problem grubunda sayılmaktadır. Karar problemi olduğunda NP-Tam problem grubunda sayılan GSP, eniyileme problemi formunda ise genel olarak NP-Zor kabul edilmektedir (Zambito, 2006). GSP'nin eniyileme problemi olması durumunda, NP-Zor grubunda olması bu problemin en iyi bir çözümünü polinomsal zamanda bulabilen bir algoritmanın bilinmediği sonucuna ulaşılmaktadır. Gezgin satıcı probleminde en iyi turu bulmak için tüm turların taranmasının gerekli olup, problemin düğüm sayısının artması ile tur sayısının hızla artmaktadır. Tablo 1.1'de problemin tur sayısının hesaplanma yöntemleri kısaca özetlenmiştir.

Tablo 1.1 *Tur Sayısının Hesaplanma Yöntemleri*

Koşul	Formül	Açıklama
Başlangıç düğümü sabit, yönler arası mesafe aynı	$\frac{(n-1)!}{2}$	Başlanılan düğüm hesaptan çıkarılır. Kopya tur olmaması için yarıya bölünür.
Başlangıç düğümü sabit, yönler arası mesafe farklı	$(n-1)!$	Başlanılan düğüm hesaptan çıkarılır. Döngüsel olduğundan başlangıç düğümü hesaptan çıkarılır. Bu karar yönler arası mesafenin aynı olması nedenlidir.
Başlangıç düğümü serbest, yönler arası mesafe aynı	$\frac{(n)!}{2}$	Başlanılan düğüm hesaptan çıkarılmaz. Kopya tur olmaması için yarıya bölünür.
Başlangıç düğümü serbest, yönler arası mesafe farklı	$n!$	Başlanılan düğüm herhangi biri olabileceğinden düğüm sayısından düşülmez.

GSP'nin çözüm karmaşıklığını anlamak için tur sayısının nasıl hızla arttığını görmek ve algoritmaların performansını analiz etmek önemlidir. Bu noktada algoritmalar için zaman ve bellek karmaşıklığı devreye girecektir.

Zaman karmaşıklığı algoritmanın çalışma zamanını ifade ederken bellek karmaşıklığı bilgisayarın çözüm esnasındaki bellek yükünü ifade eder. Gezgin satıcı probleminin çözümünde zaman ve bellek karmaşıklığı çözüm için uygulanan yöntemlere göre değişiklik göstermektedir. Algoritma, geliştirme sürecinin doğası gereği yinelemeli olması nedeniyle kesin karmaşıklık analizlerinin her değişiklikte yeniden yapılmasının pratikte oldukça maliyetli ve zahmetli olmaktadır. Bu nedenle, kaynak tüketimini daha yönetilebilir ve genellenebilir bir biçimde modelleyebilmek amacıyla, küçük değişikliklere karşı duyarlı olmayan ve algoritmaların ölçeklenme davranışını esas alan bir yaklaşıma ihtiyaç duyulmaktadır. Bu bağlamda bilgisayar bilimciler, matematik disiplininin alından ve bu ihtiyaca uygun biçimde işlev gören bir sadeleştirme aracı olan Büyük-O Gösterimini (Big-O) benimseyerek kullanıma sunmuşlardır. (Rutanen vd., 2020).

Big-O notasyonunu anlamak için bu notasyon ile ilgili genel bir açıklama gerekmektedir. Big-O notasyonu, algoritmaların en kötü durumdaki performansını değerlendirmek için geliştirilmiş bir araçtır. Gerçek süre ölçümleri yerine, giriş verisi boyutu ile çalışma süresi arasındaki ölçeklenme ilişkisini temsil etmektedir. Bu yönüyle Big-O, algoritmaları farklı karmaşıklık sınıflarına ayırmak için kullanılır.

Bu sınıflandırma, algoritmaların kötü senaryolarda nasıl davrandığını anlamamıza yardımcı olmaktadır. Örneğin, büyük veri kümeleriyle çalışırken $O(n^2)$ veya $O(2^n)$ gibi karmaşıklığa sahip algoritmalar, ciddi performans sorunlarına yol açmaktadır. Sonuç olarak, Big-O notasyonu yalnızca teorik bir araç değil, aynı zamanda algoritma seçimi ve sistem tasarımı açısından uygulamalı bir rehberdir. Gerçek zaman değerlerini vermese de algoritmaların ölçeklenebilirliği ve verimliliği hakkında önemli içgörüler sağlamaktadır (Christian ve Griffiths, 2016, s. 84-87). Çözüm için seçilen algoritmaya göre temel büyüme hesabı değişeceğinden, GSP çözümünde seçilecek çözüm yöntemi kritiktir. Big-O notasyonunun dikkate aldığı temel büyüme türleri bulunmaktadır. Bilinen en iyi algoritma için GSP karmaşıklığı $O(n!)$ 'dir. Bu girdi boyutu arttıkça işlem sayısı çok hızlı arttığını, çoğu zaman hesaplaması imkansız olduğunu ifade etmektedir.

GSP'nin çözümüne yönelik çeşitli yöntemler geliştirilmiştir. Bu yöntemler temelde; kesin yöntemler, sezgisel yöntemler ve metasezgisel yöntemler olmak üzere üç grupta sınıflandırılmaktadır. Kesin yöntemler küçük problemlerin çözümünde kullanılırken daha büyük problemler için sezgisel ve metasezgisel yöntemler kullanılmaktadır. Sezgisel ve

metasezgisel yöntemler küresel en iyiği garantilememekte; ancak pratikte özellikle büyük ölçekli NP-Zor problemlerde makul sürede en iyiye oldukça yakın çözümler üretmeleri nedeniyle tercih edilmektedir. Problemin çözümü için daha farklı çözüm yöntemleri zaman içerisinde geliştirilmiştir. Son yıllarda popüler çözüm yöntemleri haline gelmiş metasezgisel yöntemler ise doğadaki canlıların karmaşık problemlere yaklaşımından ilham alınarak geliştirilmiştir. Bu algoritmalar, ilgili canlının doğada bir problemi çözerken izlediği süreç ve ortaklaşa çalışma tekniklerini esas almaktadır (Tonguç, 2024, s.107).

Gezgin satıcı problemlerinin çözümü için özel olarak geliştirilmiş metasezgisel algoritmalar bulunmaktadır. Bunlardan en çok kullanılanlara yasaklı arama (Glover,1986), genetik algoritma (Holland, 1975; Goldberg, 1989), karınca kolonisi optimizasyonu (Dorigo, 1992) ve parçacık sürü optimizasyonu (Kennedy ve Eberhart, 1995) örnek olarak verilebilir. Çözüm yöntemi seçileceği esnada problemin türüne uygun olanın algoritmanın belirlenmesi kritiktir.

Bu çalışmanın amacı GSP'nin çözümü için karadul örümceği optimizasyon algoritmasının temel mekanizmalarının kullanıldığı hibrit bir yaklaşımla algoritmanın problem üzerindeki performansının ölçülerek bir çözüm yöntemi olarak Türkçe literatüre kazandırılmasıdır. Önerilen yöntemde BWO'nun agresif seçim mekanizmalarını dengelemek amacıyla; popülasyondaki çeşitliliği artırmak ve buna bağlı erken yakınsamayı engellemek için ada modeli temelli bir paralel hesaplama yapısı kullanılmıştır. Ada modeli göç mekanizmasının, popülasyon çeşitliliğini koruyarak büyük ölçekli problemlerde keşif ve sömürü dengesini sağlayan bir arama davranışı sağlaması amaçlanmıştır. Ayrıca algoritmada dinamik 2-opt yerel arama, farklı ön ayar seti parametreleri ve küçük, orta ve büyük ölçekli TSPLIB problemlerine yönelik çoklu deney tasarımı kullanılmıştır. Böylece çalışma, yalnızca bir algoritma önermekle kalmamakta; aynı zamanda ölçek etkisi ve parametre duyarlılığı üzerine kapsamlı bir inceleme sunarak araştırmacılar için metodolojik bir çerçeve oluşturmaktadır.

İkinci bölümde, GSP ayrıntılı biçimde ele alınmakta; problemin tanımı, matematiksel modeli, tarihi gelişimi, sınıflandırılması ve çözüm yöntemleri kapsamlı bir biçimde sunulmaktadır. Ayrıca metasezgisel algoritmaların tasarımı ve GSP ile ilişkisi açıklanmaktadır. Üçüncü bölüm, Genetik Algoritmanın kuramsal temellerini, temel kavramlarını, operatörlerini ve çalışma prensibini içermekte; ayrıca GA'nın GSP'ye nasıl

uyarlandığı detaylandırılmaktadır. Dördüncü bölümde, BWO ele alınmakta; algoritmanın biyolojik esin kaynağı, temel mekanizmaları ve parametreleri açıklanmaktadır. Beşinci bölüm, bu çalışma kapsamında önerilen Paralel Ada Modeli Hibrit Karadul Örümcek Optimizasyonu (PH-BWO) yaklaşımını içermektedir. Bu bölümde hibrit yapının amacı, GA operatörleriyle zenginleştirilmiş BWO mimarisi, ada modeli paralel yapısı, kullanılan operatörler, parametre etkileşimleri ve algoritmanın bütünsel çalışma prensibi ayrıntılı biçimde sunulmaktadır. Altıncı bölümde, farklı ölçeklerdeki TSPLIB veri setleri üzerinde gerçekleştirilen deneysel çalışmalar aktarılmakta; parametre belirlenmesi sürecinde yapılan pilot çalışmalar, parametre ayarlarının belirlenmesi, ön ayar setlerinin değerlendirilmesi, elde edilen sonuçların istatistiksel analizi ve literatürde daha önce GSP için çalışılmış iki adet hibrit BWO modeli sonuçları ile karşılaştırma yapılmaktadır. Ayrıca küçük, orta ve büyük ölçekli veri setleri için ölçek ve parametre setleri baz alınarak ayrı ayrı performans değerlendirmeleri ve karşılaştırmalar bulunmaktadır. Yedinci bölüm, çalışmadan elde edilen bulguların genel bir değerlendirmesini yapmakta ve gelecekteki çalışmalara yönelik öneriler sunmaktadır.

2. GEZGİN SATICI PROBLEMİ

Gezgin satıcı probleminde, n adet şehir ile bu şehirler arasındaki mesafelerin verildiği durumda, her bir şehrin yalnızca bir kez ziyaret edilmesi koşuluyla toplam mesafeyi enküçükleyen en kısa turu belirlemek amaçlanmaktadır.

En kısa turu belirlemek için olası tüm turların değerlendirilmesi gerekmektedir. NP-Zor grubunda olan GSP için düğüm sayısı artması ile olası tur sayısı hızla çoğalacak ve problemin çözümü hem daha maliyetli hem de daha zaman alıcı olacaktır. Bu durumun anlaşılması için 13 ve 15 düğümden oluşan 2 adet problem üzerinden tur sayısı hesabı ve çözüm yöntemi olarak tam arama algoritması (Brute Force) kullanıldığı varsayılarak zaman karmaşıklığı hesaplanmıştır. Tablo 2.1 bu problemler için tur sayıları ile Big-O kullanılarak hesaplanmış ile yaklaşık gerekli işlem sayısını göstermektedir.

Tablo 2.1 Tur Sayıları Ve Zaman Karmaşıklığı Hesapları

Düğüm Sayısı (n)	Tur Sayısı ⁴	Yaklaşık Yapılacak İşlem Sayısı ($n \times \text{Tur}$)
13	239.500.800	3.113.510.400 işlem
15	43.589.145.600	653.837.184.000 işlem

Bu algoritma tüm olası turları deneyerek en kısa olanı bulmaya çalışacaktır. Simetrik yapılı GSP için olası tur sayısı $(n-1)!/2$ düzeyindedir. Bu nedenle algoritmanın zaman karmaşıklığı $O(n!)$ olarak ifade edilmektedir. Faktöriyel düzeyde artış, zaman karmaşıklığının çok hızlı artarak hesaplamaların aşırı büyüdüğünü ve pratikte kullanılamaz hale gelerek verimsizliğini ifade etmektedir. İki adet düğüm eklendiği durumda dahi tur sayısının artışı belirgin şekilde görülmektedir. Turların maliyetlerinin elle hesaplanması pratik değildir; bilgisayar ile hesaplama ise çok zaman alır. Aynı şekilde yapılacak işlem sayısı yani zaman karmaşıklığı için hesaplanan sayıların yüksekliği açıkça görülmektedir. Bu örneklendirme ile GSP'nin düğüm sayısı artışı ile kesin yöntemler ile çözülmesinin mümkün olmadığı sonucuna varılmaktadır. Bu nedenle, özellikle büyük ölçekli problemler için

⁴ Hesaplama $(n - 1)!/2$ bu formül kullanılarak yapılmıştır.

sezgisel ve metasezgisel yöntemler geliştirilerek hızlı ve tatmin edici çözümler elde edilmeye çalışılmaktadır.

GSP’de özellikle problem boyutu büyüdükçe hesaplama karmaşıklığı önemli bir engel haline gelmektedir. Bu nedenle pratik uygulamalarda, küresel en iyiden ziyade, hızlı çalışan ve iyi yakınsak çözümler üretebilen algoritmalar tercih edilmektedir. Çünkü problem büyüklüğündeki her artış, çözüm uzayını üstel olarak genişleterek hesaplama süresi ve kaynak gereksinimlerini önemli ölçüde artırmaktadır. Bu durum, hesaplama hızı ile çözüm kalitesi arasında dengeli bir yaklaşım geliştirilmesini zorunlu kılmaktadır. Dolayısıyla, GSP’e yönelik algoritma seçiminde hem hesaplama verimliliği hem de çözüm kalitesi birlikte gözetilerek, uygulanabilir ve pratik stratejiler oluşturulması gerekmektedir.

2.1. Gezgin Satıcı Probleminin Tarihçesi

Leonhard Euler, 18. yüzyılın başlarında Königsberg’deki Pregel Nehri üzerindeki yedi köprüyü her birinden yalnızca bir kez geçerek kapalı bir yürüyüş yapmanın mümkün olup olmadığını incelemiş ve bu probleme yönelik olarak, nehir kıyıları ve adaları dört düğümle temsil eden bir yönsüz grafik modeli geliştirmiştir. Euler, aranan yürüyüşün varlığı için grafin bağlı ve her düğümün çift dereceli olması gerektiğini göstermiş, ancak bu koşullar sağlanmadığından böyle bir yürüyüşün mümkün olmadığını sonucuna varmıştır. Bu problem Çizge Kuramının en çok bilinen problemidir (Gribovskaia, Halskau ve Laporte, 2007, s. 199).

GSP daha sonrasında Sir William Rowan Hamilton tarafından çalışılmıştır. “Hamilton turu, bir çizge üzerindeki her noktadan sadece bir kez geçen (dolayısıyla aynı yoldan da sadece bir kez geçen) ve başladığı noktada biten, 19. yüzyılda yaşamış matematikçi William Hamilton’ın adıyla anılan turdur” (Süral, 2003, s. 37). GSP’nin genel tanımına bakıldığında en az maliyetli Hamilton Turu’nun arandığı görülmektedir.

GSP’nin genel formu ise Karl Menger tarafından 1932’de tarafından çalışılmıştır. Menger, gezgin satıcı problemini şu şekilde ortaya koymuştur;

“Biz buna ‘Postacı Problemi’ diyoruz (çünkü bu soru pratikte her postacı tarafından, ayrıca birçok gezgin tarafından da çözülmesi gereken bir sorudur): Sonlu sayıda nokta verildiğinde ve bu noktalar arasındaki çiftler halinde mesafeler bilindiğinde, noktaları birbirine bağlayan en kısa yolu bulma görevidir. Bu problem, elbette sonlu sayıda denemeyle çözülebilir. Ancak, deneme sayısını verilen noktaların permütasyon sayısından daha az yapacak kurallar bilinmemektedir.

Başlangıç noktasından önce en yakın noktaya, sonra bu noktanın en yakınına vb. gidilmesi kuralı genellikle en kısa yolu vermez" (Schrijver, 2005, s. 41).

GSP, 1940'lerden itibaren Hassler Whitney, Merrill Flood ve Julia Robinson gibi isimlerin katkıları ile sistematik olarak araştırılan önemli bir eniyileme problemine dönüşmüştür. 1954 yılında George Dantzig, Ray Fulkerson ve Selmer Johnson *Solution of a Large Scale Traveling Problem* adlı çalışmalarında, döngü kısıtları adını verdikleri ve alt turların oluşumunu engelleyen bir yaklaşım ile ilk doğrusal programlama tabanlı çözümü sunmuşlardır. Günümüzde döngü kısıtları yerine, bu koşul 'alt tur eşitsizlikleri' adı ile anılmaya başlanmıştır (Applegate vd., 2003, s. 92). Bu çalışmada temelde simpleks algoritmasının grafiksel yorumu kullanılmıştır. Doğrusal programlama yaklaşımının kombinatoriyel problemlere uygulanması yeni olmamasına rağmen önerilen bu yöntem özellikle atama problemi alanında önemli başarılar elde edilmiştir (Dantzig vd., 1954, s. 2).

Kombinatoriyel eniyilemede kesme düzlemi yönteminin ilk büyük ölçekli uygulaması bu çalışma ile yapılmıştır. Günümüzde kullanılan birçok yöntem bu çalışmadaki yöntemin evrilmiş hallerinden oluşmaktadır. ABD'deki 48 eyalet ve başkent olan Washington D.C. şehirlerinin arasındaki karayolu mesafelerini kullanmışlardır (Dantzig vd., 1954, s. 24). Çalışmada doğrusal bir model kurmuş ve alt turları engelleyecek özel eşitsizlikler tanımlamışlardır. Başlangıçta alt tur kısıtları olmadan problem çözülmüş, daha sonra alt tur içeren bir çözüm bulunduğunda yeni kısıt eklenerek model yeniden çözülmüştür ve bu yinelemeli süreç kesme düzlemi olarak adlandırılmıştır. Çalışmada uygulanan yöntem 49 şehirli problem üzerinde başarı ile tamamlanmıştır. Bulunan en iyi çözüm 8.623 millik tur olmuştur (Dantzig vd., 1954).⁵

1962 yılında, Bellman tarafından dinamik programlama yaklaşımı ortaya konmuştur. Bu yaklaşım, GSP için açık ve sistematik bir çerçeve sunmuş, bu yalnızca bu özel problem için değil, aynı zamanda gelecekteki kombinatoriyel eniyileme araştırmaları için de genel bir yapı işlevi görmektedir (Bellman, 1962).

⁵ Çalışma boyunca ilgili çalışmanın genelinden yapılmış çıkarım olan alıntılarda sayfa numarası belirtilmemiştir.

1970’li yıllarda büyük ölçekli problemler için ilk sezgisel çözüm yöntemleri çalışılmaya başlanmıştır (Lin ve Kernighan, 1973). Bu algoritmalar polinom zaman içerisinde kaliteli çözümlere ulaşmayı başarmıştır. Genel olarak tatmin edici doğruluk, iyi performans ve düşük hesaplama yükü ile çalışmaktadır. Tüm olası çözümleri tek tek denemek yerine problem hakkında bilinen yapısal özellikleri, özel kuralları ve varsayımları kullanarak çözüm uzayını taramaktadır.

1991 yılında Gerhard Reinelt TSPLIB test örnekleri koleksiyonu oluşturulmuştur (Applegate vd., 2006, s. 499). Bu veri setindeki örnekler günümüzde hala çalışmalarda kullanılmaktadır. Concorde, William Cook, David Applegate, Robert Bixby ve Vašek Chvátal tarafından özellikle büyük ve zor GSP problemlerini çözmek için tasarlanmış güçlü bir yazılım olup yaklaşık 130.000 satır C ile kodlanmış ve paralel işlemeyi desteklemektedir. Çeşitli kesme düzlemi yöntemleri kullanarak, karmaşık ve büyük GSP örneklerini çözmek için kesme düzlemleri ve sütun üretimini etkin şekilde birleştiren, paralel çalışan, optimize edilmiş bir dal kesme algoritması kullanmaktadır (Applegate vd., 2006, s. 489). Bu yazılım kullanılarak 85.900 düğümlük problem çözülmüştür (Cook, 2011, s. 13).

1998 yılında Keld Helsgaun, Lin-Kernighan sezgiselinin etkin ve geliştirilmiş bir halini yayımlamış ve bu sezgisel LKH olarak adlandırılmıştır. Dünya turu olarak bilinen 1.904.711 düğümlü TSPLIB problemi 2003 yılında Concorde yazılımı ve bu sezgisel kullanılarak çözülmüştür (Helsgaun, 2006, s. 92). Bu problem için bulunan en kısa tur uzunluğu 7.512.218.268 metredir. Tablo 2.2’de gezgin satıcı probleminin çözümünde kilometre taşı olarak görülen çalışmalar listelenmiştir.

Tablo 2.2. *GSP Kilometre Taşı Çalışmalar*

Tarih	Araştırmacı	Düğüm	TSPLIB
1954	Dantzig, Fulkerson,Johnson	49	dantzig42
1971	Held, Karp	64	64 rastgele nokta
1975	Camerini, Fratta, Maffioli	67	67 rastgele nokta
1977	Grötschel	120	gr120
1980	Crowder, Padberg	318	lin318

Tablo 2.2. (Devam) *GSP Kilometre Taşı Çalışmalar*

1987	Padberg, Rinaldi	532	att532
1987	Grötschel, Holland	666	gr666
1987	Padberg, Rinaldi	2.392	pr2392
1994	Applegate, Bixby, Chvatal, Cook	7.397	pla7397
1998	Applegate, Bixby, Chvatal, Cook	13.509	usa13509
2001	Applegate, Bixby, Chvatal, Cook	15.112	d15112
2004	Applegate, Bixby, Chvatal, Cook, Helsgaun	24.978	sw24978
2004	Cook, Espinoza, Goycoolea	33.810	pla33810
2006	Applegate, Bixby, Chvatal, Cook	85.900	pla85900

2.2. Gezgin Satıcı Probleminin Tanımı ve Matematiksel Modeli

Gezgin satıcı problemi, kombinatoriyel eniyileme alanında en temel ve yaygın olarak çalışılan problemlerdendir. “Belirli bir şehir kümesi ve her bir şehir çifti arasındaki seyahat maliyetleri verildiğinde, gezgin satıcı problemi, tüm şehirleri ziyaret edip başlangıç noktasına geri dönen en düşük maliyetli güzergahın bulunması problemidir” (Applegate vd., 2006, s. 1). Gezgin satıcı problemini matematiksel olarak çizge teoremi ile ifade etmek gerekirse; simetrik olması durumunda tam, yönsüz bir çizge üzerinde $V = \{v_1 \dots v_n\}$ düğümler kümesi ve $E = \{(i, j): i, j \in V\}$ ayrıtlar(hatlar) kümesi iken $G = (V, E)$ şeklinde, $A = \{(i, j): i, j \in V\}$ ayrıtlar kümesi iken asimetric ise $G = (V, A)$ tam yönlü bir çizge üzerinde tanımlanmaktadır. Burada V , üzerinde çalışılan n adet düğümden oluşan bir küme ve E ya da A ise bu düğümler arasındaki ayrıtlar veya hatlardan oluşan bir kümedir. Bu yapıya bağlı olarak, $(C=c_{ij})$ ile temsil edilen bir mesafe veya maliyet matrisi tanımlanmaktadır. Bu matris E üzerinde tanımlı olup her bir c_{ij} , düğüm i 'den j 'e geçişin maliyetini belirtir. Maliyet matrisi C bazı durumlarda sadece mesafe değil, aynı zamanda seyahat süresi veya farklı maliyet ölçütleri olarak da yorumlanabilir. Bunun yanında, C matrisinin üçgen eşitsizliğini sağlaması önemli bir özellik olarak kabul edilir.

Eğer düğümler gerçek fiziksel noktalar olan konumlara sahip olarak düşünülüyorsa yani x, y koordinat düzlemi üzerinde yer alıyorsa, buna düzlemsel problem adı verilir

(Spielman, 2009). Bu durumda, düğümler $V=\{1,2,\dots,n\}$ şeklinde indislenen bir küme olarak tanımlanmakta ve her düğüm $i \in V$ (x_i, y_i) koordinatlarına sahiptir. C matrisi aşağıdaki (2.1)'deki eşitsizliği sağlıyorsa üçgen eşitsizliği sağlanmaktadır. Bu durum V 'nin R^2 uzayındaki noktalar kümesi olduğu ve c_{ij} 'nin i ve j arasında düz hat uzaklığı yani öklidyen mesafe olduğu problemlerde geçerlidir. k düğümü burada olası aracı düğümdür. Bu eşitsizliği sağlayan simetrik GSP, öklidyen GSP olarak adlandırılmaktadır (Rego ve Glover, 2002, s. 309).

$$c_{ij} \leq c_{ik} + c_{kj} \quad i, j, k \in V \quad (2.1)$$

Ziyaret edilecek düğüm sayısı n olmak üzere, tüm düğümleri kapsayan olası turların toplam sayısı GSP'nin uygun çözüm kümesini oluşturmaktadır. Uygun çözüm kümesi GSP'nin türüne göre değişiklik göstermektedir. “En erken formülasyonlardan biri Dantzig, Fulkerson ve Johnson (DFJ) tarafından 1954 yılında önerilmiştir. Bu model, her (i, j) yayı için bir ikili değişken x_{ij} tanımlar; bu değişken yalnızca (i, j) yayı en iyi çözümde kullanıldığında 1 değerini alır ($i \neq j$) koşulu ile). DFJ formülasyonu şu şekildedir” (Laporte, 1992, s. 233):

$$\min \sum_{i \neq j}^n c_{ij} * x_{ij} \quad (2.2)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad i = 1, \dots, n, \quad (2.3)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad j = 1, \dots, n, \quad (2.4)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1, S \subset V, 2 \leq |S| \leq n - 2 \quad (2.5)$$

$$x_{ij} \in \{0,1\} \quad i, j = 1 \dots n, i \neq j \quad (2.6)$$

Bu formülasyonu açıklamak gerekirse, n düğüm sayısı, c_{ij} düğüm i 'den j 'e geçişin maliyetini, x_{ij} düğüm i 'den düğüm j 'e geçilip geçilmediğini belirtmektedir. Denklem (2.2) yani amaç fonksiyonu, en iyi turun toplam maliyetini en küçükleme olarak tanımlamaktadır. Derece kısıtları olan denklem (2.3) ve (2.4), sırası ile her düğüme yalnızca bir kez giriş yapılmasını ve her düğümden yalnızca bir kez çıkış yapılmasını sağlamaktadır. Denklem (2.5) ise, alt tur engelleme kısıtı olmakta ve n 'den az sayıda düğüm içeren turların oluşmasını

engellemektedir. Eğer S kümesi üzerinde bir alt tur oluşursa, bu tur $|S|$ adet yayı ve yine $|S|$ adet düğümü içerir. Bu durumda, kısıt (2.5)'in sol tarafı $|S|$, sağ tarafı ise $|S| - 1$ olacağından, söz konusu kısıt ihlal edilmektedir. Ancak, derece kısıtları nedeniyle yalnızca tek düğümden oluşan ya da $n - 1$ düğüm içeren alt turların oluşması zaten mümkün değildir. Dolayısıyla, kısıt (2.5)'in yalnızca $2 \leq |S| \leq n - 2$ aralığında tanımlanması yeterlidir. Son olarak, denklem (2.6) ile modeldeki değişkenlerin ikili olduğunu belirtmektedir (Laporte, 1992, s. 233).

2.3. Gezgin Satıcı Probleminin Sınıflandırılması

Gezgin satıcı problemi temel olarak problemin üç ana grupta sınıflandırılmaktadır. Zaman içerisinde problem geliştirilmiş ve farklı varyasyonları türetilmiştir. “Genel olarak GSP, simetrik gezgin satıcı problemi (sGSP), asimetrik gezgin satıcı problemi (aGSP) ve çoklu gezgin satıcı problemi (mGSP) olmak üzere sınıflandırılmaktadır” (Matai, Singh ve Mittal, 2010, s. 2). Problem yapısına göre ayrıldığında simetrik ve asimetrik olmak üzere sınıflandırılmaktadır. Matematiksel olarak ifade etmek gerekirse ($C=c_{ij}$) ile temsil edilmekte olan bir mesafe ya da maliyet matrisi, $V = \{v_1 \dots v_n\}$ düğümler kümesi ve $E = \{(i, j): i, j \in V\}$ ayrıtlar kümesi olarak tanımlı olduğu durumda, problemin simetrik yapıda ise $c_{ij} = c_{ji}$ koşulu sağlanmaktadır. Bu ayrıtın iki düğüm arası gidiş ve dönüş maliyetinin eşit olduğunu ifade etmektedir. Aksi takdirde problem asimetrik olarak sınıflandırılır. Çoklu gezgin satıcı problemi ise GSP'nin bir türevi olarak 1970'li yıllarda Bellmore ve Hong tarafından literatürde matematiksel olarak tanımlanmıştır (Bellmore ve Hong, 1974). Çoklu ifadesi burada birden çok satıcı ya da ya da aracı ifade etmektedir. Bu problemin kendi içinde de farklı çeşitleri bulunmaktadır.

GSP, gerçek dünyadaki çeşitli uygulamalardan türetilmiş birçok varyasyona sahiptir ve bu varyasyonlar, problem yapısının farklı kısıtlar ve hedeflerle nasıl geliştirilebileceğini göstermektedir. Maksimum GSP, Darboğaz GSP, Çoklu GSP gibi temel varyasyonların yanı sıra, kümelenmiş, öncelik kısıtlı, zaman pencereli, kaynak kısıtlı ve hareketli hedef içeren versiyonlar da literatürde geniş yer bulmaktadır. Her bir varyasyon, klasik GSP'e özgü “her düğüm bir kez ziyaret edilip başlangıç noktasına dönülmesi” kuralını korurken; ek zaman,

kaynak, sıralama veya yapısal kısıtlar içerebilir. Bu sayede, GSP'nin farklı sektörlerde karşılaşılan karar verme problemlerine uyarlanabilirliği artırılmıştır (Gutin ve Punnen, 2002).

Problem boyutuna göre sınıflandırma açısından GSP küçük ölçekli, orta ölçekli ve büyük ölçekli olmak üzere 3 kategoriye ayrılmaktadır. Bu sınıflandırma resmi olarak yapılmamış olmakla birlikte, genellikle algoritma karşılaştırma çalışmaları ve metasezgisel performans analizlerinde kullanılan bir ayırım türü olarak kullanılmaktadır. Genel olarak Tablo 2.3'te görüldüğü gibi sınıflandırılmaktadır.

Tablo 2.3 *GSP Ölçeğe Göre Sınıflandırma*

Düğüm Sayısı(n)	Ölçek Türü
10-100	Küçük Ölçekli(Small-Scale)
100-1000	Orta Ölçekli(Medium-Scale)
1000 ve daha büyük	Büyük Ölçekli(Large-Scale)

Hossain ve Yılmaz Acar'ın (2024, s. 224) çalışmasında, TSPLIB kütüphanesinden alınmış problemler üzerinde en iyi çözüm için farklı boyuttaki örneklerde eski ve yeni algoritmalar arasında t-testi analizi tablosunda bu sınıflandırmanın bir örneği görülmektedir.⁶

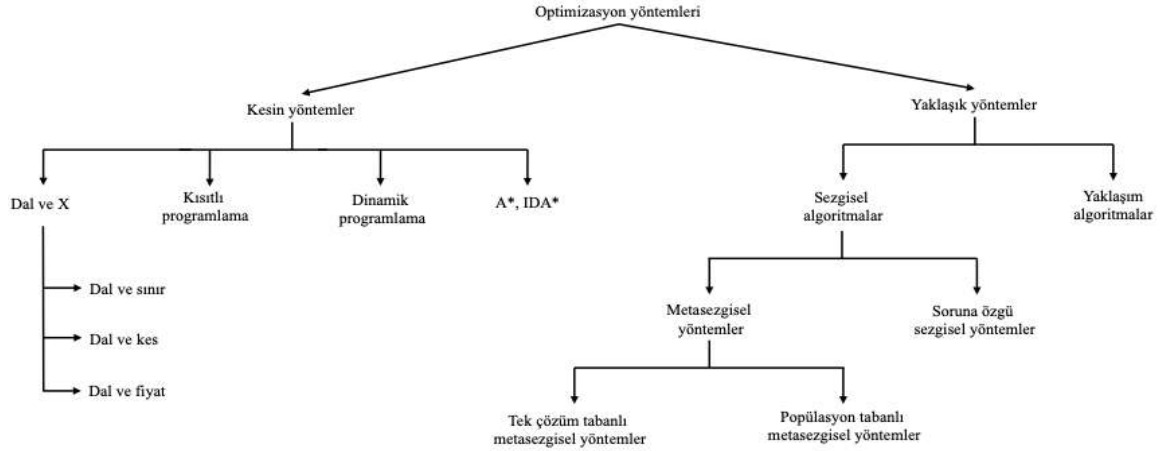
2.4. Gezgin Satıcı Problemi için Kullanılan Çözüm Yöntemleri

Eniyileme algoritmaları, başlangıç tahminlerinden yola çıkarak yinelemeli şekilde çözüme ulaşmayı amaçlar ve aralarındaki temel fark, nesiller arasındaki geçiş stratejisidir. Etkili algoritmalar; verimlilik, doğruluk ve dayanıklılık gibi özellikleri dengelerken, hız ve bellek kullanımı arasında genellikle bir ödünleşme söz konusudur.

Eniyileme problemlerinin çözüm yöntemleri genel olarak kesin ve yaklaşık yöntemler olarak ikiye ayrılmaktadır. Kesin yöntemler en iyi çözümü garanti ederken, büyük ölçekli problemlerde hesaplama maliyeti yüksektir. Yaklaşık yöntemler karmaşık, büyük ve çözümü zor eniyileme problemlerinde hızlı ve pratik çözümler üretmek için kullanılan ve tam

⁶ Birçok çalışmada ölçeğe göre sınıflandırma yapılmış olup, referans olması açısından bu çalışma örnek olarak gösterilmiştir. Çalışmada ölçeklendirme parametre hassasiyeti nedeniyle daha dar tutulmuştur.

anlamıyla kesin çözümü garanti etmese de iyi çözümler sunan yöntemlerdir. Şekil 2.1’de gösterilmiş olan sınıflandırma yapılan ilk çözüm yöntemleri sınıflandırmalarından biridir.



Şekil 2.1. Eniyileme Problemleri Çözüm Yöntemlerinin Sınıflandırması

Gezgin satıcı problemi ise karmaşık yapısı nedeniyle klasik eniyileme çözüm yöntemlerinin yetersiz kaldığı kombinatoriyel bir eniyileme problemleri grubunda sınıflandırılmaktadır.

1960’lı, 1970’li ve 1980’li yıllarda Gezgin satıcı problemini çözmeye yönelik çalışmalar, tur oluşturma yöntemleri ve tur iyileştirme yöntemleri üzerinde yoğunlaşmıştır. Son on beş yılda ise, tavlama benzetimi, yasaklı arama, genetik algoritmalar ve yapay sinir ağları gibi metasezgisel yaklaşımlar kullanılmaya başlanmıştır. Bu algoritmalar, yerel en iyilerden çıkabilme yeteneğine sahiptir. Sezgisel ve metasezgisel yöntemler, genel olarak zor problemlerin, özelde ise küresel ve kombinatoriyel problemlerin çözüm yaklaşımlarının giderek daha önemli bir bileşeni haline gelmiştir. Bir sezgisel yöntem tasarlandığında ortaya çıkan temel soru, elde edilen çözümün kalitesinin ne olduğudur. Bu soruya yanıt aramak için üç farklı yaklaşım bulunmaktadır (Floudas ve Pardalos, 2009, s. 1500).

Güncel literatürde ise gezgin satıcı probleminin sınıflandırması kesin yöntemler, sezgisel yöntemler ve metasezgisel yöntemler olarak üç gruba genişletildiği görülmektedir (Xu, Xia ve Chu, 2025). Söz konusu sınıflandırma bu çalışmada temel alınmıştır.

2.4.1. Kesin yöntemler

Kesin yöntemler, çözüme ulaşmak için problemi tam olarak çözen, yani en iyi sonucu garanti eden yöntemlerdir. Özellikle doğrusal programlama, tamsayılı programlama ve doğrusal olmayan problemler için kullanılmaktadır. GSP’de ise küçük ve orta ölçekli problemler için çözüm bulabilmektedir. Gezgin satıcı probleminin çözümünde kullanılan kesin yöntemler, en iyi turu garanti eden matematiksel programlama temelli yaklaşımlardır. Bu yöntemler genellikle GSP’nin tam sayılı doğrusal programlama formülasyonlarına dayanır ve çözüm uzayını sistematik olarak tarayarak en iyi çözümü elde etmeyi amaçlar (Matai vd., 2010).

2.4.2. Sezgisel yöntemler

Sezgisel yöntemler, NP-zor problemler gibi çözümü zor ve hesaplama maliyeti yüksek eniyileme problemlerinde, küresel en iyi çözüm garantisi sağlamaksızın kabul edilebilir ve uygulanabilir çözümler üretmeyi amaçlayan yaklaşık algoritmalarıdır. Özellikle büyük ve çözümü zor eniyileme problemlerinde hızlı ve pratik çözümler üretmek için kullanılmakta olup kesin çözümü garanti etmemekle birlikte iyi yakınsak çözümler sunmaktadır. Genellikle çözüm alanını tam taramak veya matematiksel olarak tüm mümkün çözümleri incelemek mümkün olmadığında tercih edilmektedir. “NP-zor kombinatoriyel eniyileme problemlerinin çoğu pratikte en iyi olarak çözülemez. Bu nedenle, yüksek kaliteli çözümler bulmak için sezgisel teknikler kullanılır” (Kobler, 2021, s. 950). Bu çözüm yöntemleri probleme özgü ya da genel, basit kurallara dayanan ve hızlı sonuç veren algoritmalarıdır.

Gezgin satıcı problemi için geliştirilen sezgisel yöntemler tur oluşturma, tur geliştirme ve hibrit yöntemler olarak 3 sınıfta incelenmektedir (Laporte, 1992). Tur oluşturma yöntemleri başlangıçta bir tur oluşturmak için, rastgele bir şehirden başlayıp her adımda ziyaret edilmemiş şehre giderek hızlı bir şekilde çalışarak kademeli bir tur oluşturmaktadır. Tur oluşturmada sıkça başvurulan sezgisel yöntemlerden biri de En Yakın Komşu (Nearest Neighbor – NN) sezgiselidir. Bu çalışmada da kullanılan bu yöntem, oldukça basit bir mantıkla çalışır: herhangi bir başlangıç noktasından hareket eder ve her adımda mevcut düğüme en yakın olan, henüz ziyaret edilmemiş düğümü seçerek ilerler. Tüm düğümler ziyaret edilene kadar bu şekilde devam eden süreç, hızlı çalışması ve düşük hesaplama

maliyeti sayesinde özellikle yapıcı algoritmalar içerisinde pratik bir avantaj sağlar. NN'nin hem kısa yol hem de GSP türü problemler için etkili bir başlangıç çözümü oluşturabildiği, ayrıca genetik algoritmalarla hibrit hale getirildiğinde çözüm kalitesini iyileştiren bir yapı sunduğu ifade edilmektedir. Yöntemin ağgözlü yapısı kimi zaman daha iyi alternatiflerin gözden kaçmasına yol açabilse de çoğu durumda hızlı bir şekilde makul kalitede turlar üretmesi nedeniyle tercih edilmektedir (Lima, Araújo ve Schimit, 2018, s. 7–8).

Tur geliştirme yöntemlerinde küçük bir alt turla başlayıp hangi düğümün tura katılacağına karar vererek o düğümü en uygun sıraya eklenmektedir. Amaç var olan turu iyileştirmektir. Gezgin satıcı problemi çözümü için en sık kullanılanlar yöntemler en yakın komşu sezgiseli, 2-opt, 3-opt ve Lin-Kernighan sezgiseli yöntemleridir (Floudas ve Pardalos, 2009).

Hibrit sezgisel yöntemler ise tur oluşturma ve tur geliştirme yöntemlerini bir arada kullanmaktadır. Bu yöntemde öncelikle bir tur oluşturulmakta ve bu tur iyileştirilmeye çalışılmaktadır.

2.4.3. Metasezgisel yöntemler

Metasezgisel yöntemler özellikle karmaşık ve büyük ölçekli eniyileme problemlerini çözmek için geliştirilmiş, genellikle problemten bağımsız stratejiler içeren arama yöntemi olan eniyileme algoritmalarıdır. Temelde sezgisel algoritmaların bir türü olarak ortaya çıkmış ancak zaman içerisinde bu kavramdan ayrılarak gelişmiştir. Sezgisel ve metasezgisel yaklaşımlar arasındaki temel ayrım, metodolojik kapsamı ve uygulama alanları bağlamında ortaya çıkmaktadır. Sezgisel yöntemler, sonlu adımlı algoritmalar veya belirli bir yakınsama kriterine sahip yinelemeli süreçler olarak yapılandırılan problem çözme stratejileridir. Bu yöntemlerin birincil amacı, özellikle büyük ölçekli ve karmaşık problemler için hesaplama süresi açısından verimli ve pratik çözümler üretmekte; ancak elde edilen çözümün küresel en iyi olacağına dair matematiksel bir garanti sunmamaktadırlar. Bu nedenle, küresel en iyinin hedeflenmediği, bunun yerine kabul edilebilir kalitede bir çözüme daha hızlı ulaşmanın önceliklendirildiği durumlarda etkin bir araç olarak öne çıkmaktadır. Bu yaklaşımlar, çözüm uzayını önemli ölçüde daraltarak problem çözüm sürecine kayda değer bir avantaj katmaktadır (Singh ve Choudhary, 2021).

Fred Glover 1986 yılında yasaklı arama algoritmasını tanıttığı çalışmasında metasezgisel kavramını ilk kez kullanmış ve ilgili çalışma alanının kuramsal temellerini şekillendirmiştir. “Yasaklı arama, başka bir sezgisel yöntemin üzerine yerleştirilmiş bir ‘metasezgisel’ olarak görülebilir. Bu yaklaşım, belirli hamleleri yasaklama (ya da daha genel anlamda, cezalandırma) stratejisiyle yerel en iyinin ötesine geçmeyi amaçlar“ (Glover, 1986, s. 541). Glover bu çalışması ile yerel arama gibi klasik sezgisel yöntemlerin özelleştirilmiş doğasını aşarak ve daha genel amaçlı, esnek bir yapı oluşturmuştur. Bir başka çalışmada metasezgisel terimi şu şekilde tanımlanmıştır: “Bir metasezgisel, arama alanını keşif ve sömürü süreçlerini akıllıca bir şekilde birleştiren farklı kavramlarla yönlendiren, alt bir sezgiseli yöneten yinelemeli bir üretim süreci olarak tanımlanır. Verimli bir şekilde en iyiye çözüme yakın sonuçlar bulabilmek amacıyla, bilgiyi yapılandırmak için öğrenme stratejileri kullanır” (Osman ve Laporte, 1996, s. 1).

Bilinen ilk metasezgisel çalışma 1983 yılında Scott Kirkpatrick, C. Daniel Gelatt Jr ve Mario P. Vecchi tarafından yayımlanan makalede tanıtılan tavlama benzetimi (SA)’dır (Kirkpatrick, Gelatt ve Vecchi, 1983). Bu çalışma teknik olarak metasezgisel bir yöntem izlemektedir. Ancak bu dönemde metasezgisel terimi henüz kullanılmadığından daha sonra bu gruba dahil edilmiştir.

Geleneksel eniyileme teknikleri, çözüm sürecinde belirli matematiksel varsayımlara ve problem yapısına ilişkin katı kısıtlamalara ihtiyaç duyarken, metasezgisel algoritmalar bu tür kısıtlamalardan bağımsız olarak çok çeşitli problem türlerine uygulanabilen genel çözüm çerçeveleri sunmaktadır. Özellikle kombinatoriyel eniyileme problemlerinde, bu algoritmalar küresel en iyi çözümü garanti etmemekle birlikte, kabul edilebilir kalitede çözümleri makul süreler içerisinde elde edebilme yetenekleri sayesinde kesin algoritmaların güçlü bir alternatif olmaktadır.

Metasezgisel algoritmalar mühendislik, lojistik, yapay zeka gibi birçok alanda yaygın olarak kullanılmaktadır. *Bedava Yemek Yok* (No Free Lunch-NFL) teoremi, hiçbir eniyileme algoritmasının evrensel olarak en iyisi olmadığını, çünkü bir problemdeki başarının başka bir problemdeki zayıf performansla dengelendiğini belirtir. Ancak, popülasyon tabanlı yapıları sayesinde metasezgisel algoritmalar bu genel kuraldan sıyrılarak belirli problem sınıfları için daha iyi performans gösterebilir. Bu durum, yani belirli problemler için üstün çözümler bulma imkanı, metasezgisel algoritmalar alanındaki araştırmaların devam etmesinin temel

nedenidir (Adam vd., 2019). Günümüzde metasezgisel algoritmalar bu nedenle hala aktif ve dinamik bir çalışma alanıdır. Hibrit yaklaşımlar ve derin öğrenme ile entegrasyon gibi teknikler araştırılmaktadır.

Metasezgisel yöntemlerin gelişimiyle birlikte, özellikle hibrit ve çok aşamalı arama stratejileri literatürde önemli bir araştırma alanı haline gelmiştir. Bu bağlamda Memetik Algoritmalar(MA), bir metasezgisel çerçevenin içerisine güçlü bir yerel arama mekanizmasının entegre edildiği hibrit modeller olarak öne çıkmaktadır.

Memetik algoritma, evrimsel hesaplamanın küresel arama kapasitesi ile yerel arama yöntemlerinin sömürü yeteneğini birleştiren hibrit bir metasezgisel eniyileme yaklaşımıdır. Bu yaklaşımı ilk olarak Pablo Moscato GA ve yerel arama hibriti olarak tanımlamıştır. Memetik algoritmanın temelinde, bireylerin yerel aramalarla iyileştirilerek öğrenmesi ve bu öğrenmenin evrim sürecine yansması yer almaktadır (Moscato,1989).

Memetik algoritmada her birey, kendi çözümünü yerel arama ile geliştirmektedir. Bu nedenle algoritmanın evrimsel süreci, yalnızca biyolojik değil kültürel evrime benzer bir yapıda kurgulanmaktadır. Memetik algoritmalar hızlı yakınsama, yüksek kararlılık ve elit bireylerin daha güçlü şekilde iyileştirilmesini sağlayarak problemi daha pürüzsüz hale getiren bir sömürü mekanizması sağlamaktadır (Moscato,1989).

Memetik algoritmalar, evrimsel süreçte üretilen bireylerin yalnızca popülasyon içi rekabete değil, aynı zamanda her bireyin kendi yerel düzeyinde iyileştirilmesine dayanmaktadır. Bu yapı, özellikle GSP gibi çok sayıda yerel en iyi içeren problemlerde hızlı yakınsama ve yüksek çözüm kalitesi elde edilmesini mümkün kılmaktadır. Literatürde 2-opt, 3-opt gibi yöntemlerin genetik algoritmalarla birleştirilerek kullanılan ilk örnekleri memetik yaklaşımın temellerini oluşturmuştur. Basit yapısı, düşük hesaplama maliyeti ve güçlü iyileştirme kapasitesi nedeniyle memetik yaklaşımda sıklıkla kullanılan 2-opt, metasezgisellerle birlikte hibritleştirilerek çözüm kalitesini artırmak amacıyla kullanılmaktadır (Hussain, Muhammad ve Sajid, 2018, s. 3).

Metasezgisel yöntemlerin performansını artıran bir diğer önemli kavram Ada Modeli olarak bilinen paralel evrimsel mimarilerdir. Ada modelinde popülasyon birden fazla alt popülasyona (adalara) bölünmekte, her ada kendi evrimsel sürecini bağımsız olarak gerçekleştirmekte ve belirli aralıklarla bireyler adalar arasında göç etmektedir. Bu yaklaşım hem çeşitliliğin korunmasını sağlamakta hem de çok modlu arama uzaylarında erken

yakınsamanın önüne geçmektedir. Özellikle büyük ölçekli problemlerde ada modellerinin avantajı belirginleşmekte; farklı adalarda farklı çözüm stratejilerinin paralel ilerlemesi, daha geniş bir arama kabiliyeti yaratmaktadır. Ada modeli ile memetik yaklaşımın birleşimi ise paralel memetik algoritmalar olarak adlandırılmakta ve literatürde GSP gibi zorlu problemlerde önemli başarılar elde etmektedir. Bu yapı, hem evrimsel operatörlerin hem de yerel aramanın birlikte kullanılmasını sağlayarak çeşitlendirme ve yoğunlaşma dengesini güçlü biçimde optimize eder (Alba ve Tomassini, 2002; Alba ve Luque, 2007).

2.4.3.1 Metasezgisel algoritmaların tasarımı ve uygulanması

Metasezgisel algoritmaların çalışma mantığı, temelde yinelemeli bir arama süreci üzerine kuruludur. Bu süreç, başlangıçta rastgele oluşturulan bir veya birden fazla çözümünden hareketle, yeni çözümler üretme ve mevcut çözümleri iyileştirme adımlarını içermektedir (Karimi-Mamaghan, Mohammadi, Meyer, Karimi-Mamaghan ve Talbi, 2022). Metasezgisel bir algoritmanın tasarımında, arama alanının yeterli düzeyde keşfedilmesini sağlayan keşif ve elde edilen yüksek kaliteli çözümler üzerinde çalışmayı hedefleyen sömürü sistemi olmak üzere, birbiriyiyle çelişen iki temel stratejik hedef temel alınmaktadır. “Çeşitlendirme (diversification) ile yoğunlaşma (intensification) arasındaki denge, bir yandan arama alanında yüksek kaliteli çözümlerin bulunduğu bölgelerin hızlı bir şekilde tespit edilmesi, diğer yandan ise daha önce keşfedilmiş veya yüksek kaliteli çözümler sunmayan bölgelerde gereksiz zaman harcanmaması açısından büyük önem taşımaktadır. Çeşitlendirme terimi genel olarak arama uzayının keşfine, yoğunlaştırma terimi ise birikmiş arama deneyiminin sömürülmesine karşılık gelir.” (Blum ve Roli, 2003, s. 271). Söz konusu dengeyi sağlamak için parametrelerin doğru biçimde belirlenmesi kritik öneme sahiptir. Bu parametreler, arama sürecinin farklı aşamalarında farklı davranışların sergilenmesini mümkün kılarak algoritmanın dinamik biçimde uyum sağlamasına katkıda bulunmaktadır (Karimi-Mamaghan vd., 2022). Metasezgiseller, yerel arama, popülasyon bazlı arama, rastgelelik ve uyarlanabilirlik gibi sezgisel teknikleri bir araya getirerek küresele en yakın çözümleri aramaktadır. Temel amaç yerel en iyiden kaçarak küresel en iyiye ulaşmaktır. (Blum ve Roli, 2003).

Bir metasezgisel algoritmanın etkin tasarımı ve uygulanması, parametrelerin belirlenmesi ve keşif ve sömürü dengesinin sağlanmasının yanı sıra çözümün uygun biçimde temsil edilmesine de bağlıdır. Temsil yöntemi, tüm olası çözümleri kapsayacak ve arama operatörleri tarafından kolayca işlenebilecek şekilde belirlenmelidir. Algoritmanın başarısı, çözüm kalitesini ölçen amaç fonksiyonunun doğru tanımlanmasına ve bu fonksiyonun arama yönünü etkin biçimde yönlendirmesine de dayanmaktadır. Erken yakınsama riskini azaltmak ve küresel en iyiye ulaşma olasılığını artırmak amacıyla operatörler ve parametreler dinamik olarak uyarlanmaktadır. Yeni çözümler, başlangıç çözümünden hareketle özel olarak tasarlanmış mekanizmalar aracılığıyla üretilmekte ve her yeni çözüm, amaç fonksiyonu değeri üzerinden değerlendirilmektedir. Bu süreçte, çözümler karşılaştırılarak hangisinin korunup korunmayacağına karar verilmekte ve önceden tanımlanmış durma kriteri sağlanana kadar yinelenmeli şekilde devam etmektedir. Süreç sonunda elde edilen en iyi çözüm raporlanarak algoritmanın performansı belirlenmektedir (Karimi-Mamaghan vd., 2022).

Bu noktada, kullanılan operatörler çözüm uzayında yeni bireylerin oluşturulmasını sağlayarak çeşitliliği güçlendirir. Bu çalışmada kullanılan GA operatörlerinin BWO yapısına entegre edilmesi hem keşif hacmini genişletmekte hem de yerel arama ile elde edilen iyileştirmelerin popülasyona yayılmasını kolaylaştırmaktadır.

2.4.3.2 Metasezgisel algoritmaların sınıflandırılmasına ilişkin literatür araştırması

Metasezgisel algoritmaların birçok farklı sınıflandırması literatürde bulunmaktadır. Sınıflandırma; bu yöntemlerin esnek, çok boyutlu ve birbirine geçen yapıları nedeniyle oldukça güçlüdür. Birçok algoritma hem tek çözüm tabanlı hem de popülasyon tabanlı özellikler gösterebilmekte; aynı zamanda biyolojik, fiziksel veya sosyal ilham kaynaklarını bir arada barındırabilmektedir. Hibrit ve çok stratejili yaklaşımların yaygınlaşması, sınıflar arasındaki sınırların belirginliğini azaltmakta; mevcut algoritmaların küçük değişikliklerle yeni varyantlar olarak sunulması ve zaman içinde evrimleşerek farklı kategorilere kayması ise bu süreci daha da karmaşık hale getirmektedir. Bu nedenlerle, katı hiyerarşik sınıflandırmalar yerine esnek ve bağlama duyarlı sınıflandırmaların tercih edilmesi daha uygun görülmektedir (Stegherr, Senger ve Müller, 2022; Stegherr, Heider ve Hähner, 2022).

En sık tercih edilen sınıflandırma ilham kaynağına göre sınıflandırmadır. Darwin'in evrim teorisine dayanan GA'nın geliştirilmesi ve uygulanmasının ardından, geliştirilen birçok metasezgisel çözüm yöntemi evrimsel süreçleri, hayvan davranışlarını ve fizik yasaları gibi biyolojik veya doğa olaylarını temel almaktadır. Zaman içerisinde algoritmaların sınıflandırılması her ne kadar zorlaşsa da literatür incelendiğinde metasezgisel algoritmalar içerisinde biyolojik davranışlardan esinlenenlerin sayıca önemli bir kısmı oluşturduğu görülmektedir. Bu algoritmaların çoğu, doğa yasaları, biyolojik süreçler ve içsel mekanizmalar gibi işlenebilir doğal olgulara dayalı olarak tasarlanmaktadır (Li, 2024).

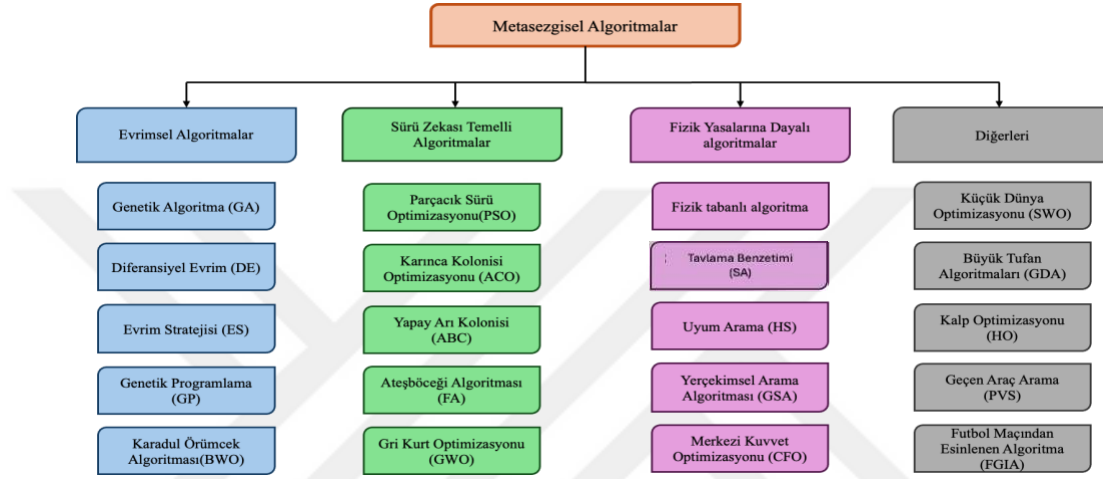
Fister vd. 'nin (2013) önermiş olduğu sınıflandırmaya göre çözüm yöntemlerini beş ana kategoriye ayırmaktadır: biyolojiden esinlenen, sürü zekasından esinlenen, fiziksel ve kimyasal yasalardan esinlenen, özel hayvan davranışlarından esinlenen ve hibrit yöntemler. Genel olarak, sınıflandırmaların büyük oranda odak noktası, vurgu ve ele alış biçimine bağlı olarak değişkenlik gösterdiği söylenebilir.

Molina, Herrera ve Gonzalez'e (2020) göre son yirmi yıl içerisinde, doğadan ve biyolojiden esinlenen eniyileme algoritmaları⁷, ilham kaynaklarından bağımsız olarak önemli bir popülerite ve akademik tanınırlık kazanmıştır. Bu artan ilgi, söz konusu algoritmaların öğrenme, uyum sağlama ve karmaşık problemlere etkin çözümler üretme kapasitesine dayanmaktadır. Bu algoritmalar, kombinatoriyel eniyileme ve büyük ölçekli eniyilemede somut faydalar sağlamaktadır. İncelemiş oldukları 300'den fazla çalışmada en çok çalışılan yöntemin doğadan ilham alan algoritmalar grubundaki sürü zekası tabanlı olduğu sonucuna ulaşmışlardır.

Rajwar, Deep ve Das'a (2023) göre literatürde en yaygın kullanılan sınıflandırma yöntemi ilham kaynağına göre sınıflandırmadır. Ayrıca, doğadan ilham alan algoritmalar veya metasezgiseller kavramı esasen doğal ya da biyolojik olgulara dayandığından, yararlı bir sınıflandırmadır. Bu çalışmada sınıflandırma türleri ilham kaynağına göre sınıflandırma, popülasyon boyutuna göre sınıflandırma, popülasyonun hareketine göre sınıflandırma ve parametre sayısına göre sınıflandırma olmak üzere dört ana başlık altında toplanmıştır. Bu

⁷ Doğadan esinlenen algoritmalar ve biyolojiden esinlenen algoritmalar literatürde sık sık birbiri yerine kullanılmaktadır. Bu nedenle çalışma için iki sınıflandırma da dikkate alınmıştır (Molina vd., 2020, s. 2).

sınıflandırmaların mutlak olmadığı ve bazı algoritmaların aynı anda birden fazla kategoriye dahil edilebilmesi mümkündür görülmektedir. Ancak sınıflandırmaların temelde doğadan ilham alan algoritmalara dayandığı sonucu çıkarılabilir. İlham kaynağına göre yapılan sınıflandırma Şekil 2.3'te verilmiştir.⁸



Şekil 2.2. İlham Kaynağına Göre Metasezgisel Algoritmaların Sınıflandırılması

Jakšić vd. 'nin (2023, s. 7) yapmış olduğu araştırmaya göre doğadan esinlenilmiş olan sürü zekası tabanlı çözüm yöntemlerinin incelemiş oldukları çalışmalarda %67,12 oranında kullanıldığı sonucuna varmışlardır.

Dökeroğlu, Cantürk ve KüçükYılmaz'ın (2024) yapmış olduğu çalışmada 2019-2024 arasında geliştirilen 150'den fazla metasezgisel algoritma incelenmiş olup en çok kullanılan veya esinlenen çalışmaların sürü zekası tabanlı algoritmalar olduğunu sonucuna varılmaktadır.

Li 'nin (2024) çalışmasına göre metasezgisel algoritmaların 1994'ten günümüze kadarki gelişimi, her aşamada doğadan esinlenen yöntemlerin hakimiyetiyle şekillenmiştir. Alanın temelleri, genetik algoritma gibi evrimsel ve simüle edilmiş tavlama gibi fiziksel süreçlerden ilham alan yöntemlerle atılmış, karınca kolonisi optimizasyonu gibi sürü zekası tabanlı algoritmalarıyla ana akım haline gelmiştir. Modern patlama döneminde ise Gri Kurt

⁸ Bu çalışmada Şekil 2.3'te verilmiş olan sınıflandırma temel alınacaktır.

Optimizasyonu ve Yapay Arı Kolonisi gibi çok çeşitli yeni doğa esinli algoritmanın ortaya çıkmasıyla bu baskınlık daha da pekişmiştir.

Sonuç olarak, metasezgisel algoritmaların esnek ve çok boyutlu yapıları nedeniyle sınıflandırılmasının zorluklar içerdiği literatürde kabul gören bir gerçektir. Buna karşın, incelenen çalışmalar, en yaygın ve kullanışlı sınıflandırma yönteminin algoritmaların ilham kaynağına dayandığı konusunda birleşmektedir. Bu bağlamda, metasezgisel algoritmaların gelişim sürecinin, temelden günümüze kadar büyük ölçüde doğadan esinlenen evrimsel süreçler ve biyolojik olgulardan esinlenen yöntemlerin hakimiyetiyle şekillendiği görülmektedir. Farklı zaman aralıklarını ve yüzlerce makaleyi analiz eden çok sayıda çalışma, doğadan ilham alan algoritmaların sık kullanılan ve araştırılan yaklaşım olduğunu istatistiksel olarak ortaya koymaktadır. Bu durum, doğadan esinlenen modellerin, karmaşık eniyileme problemlerinin çözümündeki etkinliğinin ve akademik alandaki popülaritesinin altını çizmektedir.

3. GENETİK ALGORİTMA

Bu bölümde genetik algoritmaların teorik temeli, temel kavramları, operatörleri, çalışma prensibi ve gezgin satıcı problemine uyarlanması bütüncül bir çerçevede ele alınmıştır. Holland'ın kuramsal yaklaşımından başlayarak, Darwinci seçim mantığının algoritmik yapıya nasıl aktarıldığı açıklanmış; ardından GA'nın yapı taşları ve arama davranışını belirleyen operatörler teknik ayrıntılarıyla incelenmiştir. GSP'ye özgü permütasyon temsili, özel çaprazlama mutasyon operatörleri ve uygunluk fonksiyonunun rolü ayrıntılı biçimde ortaya konmuş; böylece GA'nın bu özel probleme neden uyarlanabilir ve etkili bir yöntem olduğu temellendirilmiştir. Bölüm ayrıca GA'nın sınırlılıklarını, parametre duyarlılığını ve erken yakınsama risklerini değerlendirerek yöntemin pratik kullanım koşullarını gerçekçi bir biçimde yansıtmıştır. Son olarak, yerel arama ile zenginleştirilmiş memetik GA yapılarının GSP'deki belirgin üstünlüğü vurgulanmış ve GA'nın hibrit modellerle nasıl daha güçlü hale geldiği aktarılmıştır. Tüm bu içerik, GA'nın hem kuramsal hem uygulamalı yönleriyle GSP çözümünde kritik bir araç olduğunu akademik bütünlük içinde ortaya koymaktadır.

3.1. Genetik Algoritmaların Kuramsal Temeli

Doğal seçim, bir popülasyondaki bireylerin yaşam mücadelesi sırasında sürekli olarak elenen ve biriken özelliklerin oluşturduğu uzun vadeli bir uyum sürecidir. Her birey, genetik yapısının ona sağladığı özelliklerle çevresel koşullar karşısında sınanmakta; bu sınamada daha dayanıklı, daha uyumlu, temelde güçlü olan bireyler hayatta kalma ve üreme konusunda diğerlerine göre belirgin bir avantaj elde etmektedir. Böylece taşıdığı yararlı özellikler gelecekteki nesillere daha fazla aktarılırken, çevreye daha az uyum sağlayan bireylerin özellikleri zaman içinde azalmakta ya da tamamen kaybolmaktadır. Güçlü olanın hayatta kalması doğal seçilimin en belirgin özelliğidir ancak burada güç çevreyle en iyi uyumu sağlayan tüm özelliklerin toplamını ifade etmektedir. Uzun vadede popülasyonların yapısını şekillendiren ve türlerin bulunduğu ortamlara giderek daha iyi uyum sağlamasına yol açan temel kuvvet olarak tanımlanabilmektedir (Darwin, 2016).

Genetik algoritma karmaşık problemlere uyarlanabilir ve esnek çözümler üreten, metasezgisel bir eniyileme yaklaşımıdır. Doğal seçim, çeşitlilik ve yeniden kombinasyon

gibi biyolojik süreçlerden ilham alınarak geliştirilen bu yöntem, çözüm uzayı geniş ve yapısal olarak karmaşık olan problemlerde geleneksel deterministik arama tekniklerine göre daha etkili bir keşif ve sömürü dengesi sunmaktadır. Genetik algoritmaların temel gücü, çözümün yalnızca tek bir noktasını değil, çözümü oluşturan yapı taşlarını da değerlendirmesinden gelmektedir. Böylece algoritma, yüksek performans sergileyen yapıların popülasyon içinde korunmasını ve tekrar sınanmasını sağlayarak, giderek daha yüksek kalitede çözüm kümelerine ulaşmaktadır. Bu yönüyle genetik algoritmalar, özellikle evrimsel mantığın doğrudan modellenebildiği yapay sistemlerde hem küresel arama kapasitesi hem de uyarlanabilir öğrenme yeteneğiyle öne çıkan önemli bir metasezgisel yöntem olarak karşımıza çıkmaktadır.

GA yönteminin temelini 1975 yılında John H. Holland ortaya atmıştır (Holland,1975). Holland'ın yaklaşımı, Darwin'in *Türlerin Kökeni* adlı eserinin önsözünde vurguladığı biçimde, türlerin doğa yasaları çerçevesinde, küçük farklılıkların birikimiyle ve seçim baskıları altında ortaya çıktığı fikrine dayanmaktadır. Darwin'in kendi ifadesiyle, amaç "hayvanları ve bitkileri, Newton'un gezegenler için yaptığı gibi, mucizelere başvurmadan doğa yasalarıyla açıklamaktır" (Darwin, 2016). Genetik algoritmaların kuramsal temellerini atan Holland *Adaptation in Natural and Artificial Systems* adlı kitabında, kimi bölümlerde Alfred Russel Wallace'a referans vermiş olsa da algoritmalarının düşünsel zemini esasen Darwinci seçim anlayışıyla uyumlu olduğu görülmektedir. Wallace her ne kadar doğal seçim fikrine katkı sunmuş olsa da süreçleri açıklamada doğa yasalarının evrensel yeterliliğini kabul etmemiş; özellikle insan zihni söz konusu olduğunda doğal seçilimin tek başına açıklayıcı olamayacağını savunmuş ve doğal seçilimi fark etmiş olsa da onu Darwin gibi evrensel bir yasa olarak görmemiştir (Kamözüt, 2009, s. 206–207).

Holland'ın genetik algoritmalarında ise seçim, mutasyon, çaprazlama ve adaptasyon süreçleri tamamen doğaüstü güçlere veya teolojik bir yönetime başvurmadan, değişmez yasaların işleyişine dayanmaktadır: *Türlerin Kökeni* 'nin önsözünde Darwin açıkça şöyle belirtmektedir: "Bu ciltte sunulan görüşlerin, doğanın değişmez yasalarının etkileşimiyle, türlerin kökenini ve çeşitlenmesini açıklayabileceğini göstereceğimi umuyorum." (Darwin, 1876, önsöz). Darwin'in bu vurgusunun önemi, doğal seçilimi evrensel bir uyum yasası, yani çevre ile varyasyon arasındaki sürekli etkileşimden doğan bir matematiksel süreç olarak konumlandırmış olmasıdır. Holland'ın GA yaklaşımı tam olarak bu yasayı biçimsel bir

modele dönüştürmektedir. Dolayısıyla GA'ların temel adaptasyon mekanizması, Darwin'in küçük farkların kademeli birikimi anlayışının doğrudan algoritmik karşılığıdır.

Sonuç olarak Holland'ın modelinin dayandığı iki ayrı düzeyi ayırt etmek mümkündür. Holland'ın Wallace'tan yaptığı alıntılar biyolojik düzeyde ve teknik niteliktedir; kuramsal çerçeve yani doğal seçilimin evrenselliği, küçük varyasyonların birikimi, çevre ile organizma arasındaki sürekli geri besleme, uyumun kümülatif ve dinamik yapısı Darwin'i yansıtmaktadır. Holland'ın popülasyon temelli eniyileme şeması, her neslin bir öncekinden ufak değişiklikler ile türemesi ve türün değişerek çeşitlenmesi fikrini tam da Darwinci değişerek türeme yasasının bir bilgisayarlı modeline dönüştürür.

Genetik Algoritmaların Gezgin Satıcı Problemine uyarlanması ise Goldberg ve Lingle'in çalışmaları ile mümkün olmuştur. 1992 yılında Holland'ın kitabının 2. Baskısı çıkana dek birçok bilim insanı tarafından genetik algoritma çalışmaları yapılmıştır (Holland, 1992). 1985 yılında dünyadaki ilk uluslararası genetik algoritmalar konferansı Pennsylvania, ABD'de yapılmıştır. Burada farklı üniversitelerden bu konuda araştırma yapan bilim insanları bir araya gelmiştir ve Holland ekolünden çıkan öğrencilerin çoğu ilk kez burada sunum yapmıştır. Bu tarih evrimsel hesaplama alanının başlangıcı olarak kabul edilmekte ve genetik algoritmaların küçük bir araştırma alanı olmaktan çıkıp uluslararası bir bilim alanı haline gelmesi sağlanmıştır. Holland, De Jong ve Baker gibi araştırmacılar GA'nın seçim, şema işleme ve öğrenme yapısına yönelik temel katkılar sunarken; Goldberg, Grefenstette ve Davis gibi isimler GSP, iş çizelgeleme ve devre yerleşimi gibi problemlere yönelik ilk GA uygulamalarını tanıtmıştır. Böylece konferans, evrimsel algoritmaların hem bilimsel olarak olgunlaşmasına hem de farklı alanlara yayılmasına önemli ölçüde katkı sağlamıştır (Grefenstette, 1985).

3.2. Genetik Algoritmanın Temel Kavramları

Holland'ın Şema Teorisi, genetik algoritmaların neden etkili sonuçlar üretebildiğini açıklayan en temel kuramsal yaklaşımlardan biridir. Bu teoriye göre GA'lar, popülasyonda yüksek performans gösteren küçük kalıpları (yapı taşları) nesiller boyunca korur ve çaprazlama işlemiyle bu yapı taşlarını bir araya getirerek daha güçlü çözüm bileşenleri oluşturur (Holland, 1975, s. 179–196). GSP bağlamında bu yapı taşları, genellikle kısa ve

avantaj sağlayan alt turlar veya iyi kenarlardan oluşmaktadır. Algoritmanın güçlü performansı, bu küçük ama kaliteli yapıların popülasyon içinde korunması ve yeni bireylerde yeniden birleştirilmesiyle açıklanabilmektedir. Her ne kadar Şema Teorisi modern GA uygulamalarında doğrudan matematiksel bir araç olarak kullanılmasa da GA'nın evrimsel mantığını ve neden küresel en iyiye yakın çözümler üretebildiğini kavramsal düzeyde anlamamıza önemli katkı sağlamaktadır.

Doğal seçim ve varyasyon ilkeleri, popülasyon tabanlı arama süreçlerine uyarlanarak algoritmanın nesiller boyunca daha iyi çözümler üretmesi sağlanmaktadır. Genetik algoritmalar, çözüm üretme biçimlerini açıklayabilmek için biyolojik evrimdeki kavramlarla güçlü bir benzetim ilişkisi kurmaktadır. Tablo 3.1.'de biyolojik genetik yapıyla genetik algoritmalar arasındaki kavramsal eşleşmeler özetlenmekte; böylece GA'nın çalışma prensiplerinin altında yatan evrimsel metaforlar açık biçimde gösterilmektedir.

Tablo 3.1 *GA'nın Çalışma Prensiplerinin Altında Yatan Evrimsel Kavramlar*

Biyolojik Terim	Tanım (Biyoloji)	Genetik Algoritmadaki Karşılığı	Tanım (GA)
Kromozom (Chromosome)	Bir organizmanın genetik yapısını taşıyan yapı.	String / Yapı (Structure)	Çözümün kodlandığı bit dizisi veya sembolik temsil.
Gen (Gene)	Belirli bir özelliği belirleyen kalıtsal birim.	Özellik / Dedektör (Feature/Detector)	Çözümde belirli bir parametreyi temsil eden değer veya bit kümesi.
Allel (Allele)	Bir genin alabileceği farklı değerler.	Bit değeri / Parametre değeri	Genin temsil ettiği özelliğin GA içindeki değeri (0/1 veya başka kodlama).
Genotip (Genotype)	Organizmanın tüm genetik yapısı.	Bireyin kodlanmış hali	GA içinde bir çözümün tam kodlaması (tek string veya çoklu string).
Fenotip (Phenotype)	Genotip + çevre etkileşimi sonucu ortaya çıkan organizma.	Çözüm uzayındaki nokta / Parametre seti	Kodlamanın çözülmesiyle elde edilen gerçek çözüm (problem bağlamındaki anlamı).

Tablo 3.1 incelendiğinde kromozom, gen, allel ve fenotip gibi temel biyolojik terimlerin, yapay sistemlerde çözümün nasıl temsil edildiğini göstermek amacıyla yeniden tanımlandığı görülmektedir.

Genetik algoritmanın performansını etkileyen temel parametreler popülasyon büyüklüğü, çaprazlama oranı ve mutasyon oranıdır. Popülasyon büyüklüğü arttıkça iyi çözümlere ulaşma olasılığı yükselmekte, ancak hesaplama maliyeti de aynı oranda artmaktadır. Çok küçük popülasyonlarda örnekleme hataları belirginleşirken, aşırı büyük popülasyonlar da pratik olmamaktadır. Bu nedenle çözüm kalitesi ile işlem süresi arasında bir denge kurmak gerekmektedir. Mutasyon olasılığı çok yüksek olduğunda mevcut bireyler bozulur ve arama süreci rastgeleleşir. Bu durum nedeniyle genellikle düşük mutasyon değerleri tercih edilmektedir. Çaprazlama olasılığı genellikle orta ile yüksek aralığında seçilir; çünkü bu değerlerin yeni ve işe yarar kombinasyonlar üretme açısından daha etkili olduğu görülmüştür (Talbi, 2009 ,s. 221).

GA'nın evrimsel sürecinin temelini üç ana operatör oluşturmaktadır. Seçim operatörü, yüksek uygunluk değerine sahip bireylerin popülasyonda daha fazla temsil edilmesini sağlayarak iyi yapıların korunmasına yardımcı olur. Çaprazlama operatörü, klasik olarak tek-nokta veya iki-nokta çaprazlama teknikleridir. Bu operatör, iki ebeveynin kısmi yapılarını birleştirerek yeni bireyler üretmektedir. Mutasyon ise permütasyon dizisi üzerinde küçük, ancak etkili değişiklikler yaparak popülasyon çeşitliliğini sürdürmekte ve erken yakınsamayı engellemektedir (Goldberg, 1985).

3.3. Genetik Algoritma Operatörleri

Genetik algoritmalarda seçim operatörü, popülasyondaki bireyler arasından hangi bireylerin üreme sürecine katılacağını belirlediği için evrimsel aramanın temel yönlendiricisidir. Uygun şekilde tasarlanmış bir seçim mekanizması, yüksek uygunluk değerine sahip bireylerin genetik etkisini artırarak algoritmanın iyi çözüm bölgelerine daha dengeli ve hızlı bir biçimde yönelmesini sağlamaktadır.

Arama sürecinin ilk aşamalarında, popülasyondaki ortalama düzey bireyler arasında aniden çok düşük uygunluğa sahip bir birey belirdiğinde seçim mekanizması bu bireye aşırı ağırlık vermeye başlamaktadır. Bu durum, söz konusu bireyin birkaç nesil içinde popülasyonu hızla ele geçirmesine ve çeşitliliğin erken safhada kaybolarak algoritmanın erken yakınsama yaşamasına neden olabilmektedir. Benzer bir sorun sürecin ileri aşamalarında da ortaya çıkabilmektedir. Popülasyon gerçekten düşük uygunluk değerlerine

sahip bireylerden oluşsa bile, bu bireylerin uygunlukları normalize edildiğinde birbirine çok yakın çıkmakta ve seçim baskısı ortadan kalkmaktadır. Böylece, uygunluk değeri farklılıkları anlamlı olsa bile, bireyler neredeyse eşit üreme şansına sahip olmakta ve arama ilerlemesi yavaşlamaktadır (De Jong, 1985, s. 176).

Genetik algoritmalarda çaprazlama operatörü, popülasyona dahil edilecek yeni ve potansiyel olarak daha başarılı bireyleri üretmenin temel mekanizmasıdır. Basit uygulamalarda süreç, popülasyondan rastgele iki bireyin seçilmesiyle başlamaktadır. Popülasyon zaten yüksek uygunluk değerine sahip bireylere doğru eğilim gösterdiği için, bu rastgele seçim de doğal olarak benzer bir yanlılık içermektedir. Çaprazlama operatörü, seçilen ebeveynlerin belirli kısımlarını bir araya getirerek yeni bireyler oluşturmaktadır. Oluşturulan bu yavru birey popülasyona eklenmekte ve belirlenen kurallara göre başka bir bireyin yerine geçmektedir. Bu noktadan sonra yeni birey, sahip olduğu uygunluk düzeyine bağlı olarak ya popülasyonda güçlenme fırsatı bulmakta ya da süreç içerisinde elenmektedir (Ackley, 1985, s. 123).

Mutasyon operatörü genetik algoritmalarda, popülasyondaki genetik çeşitliliğin korunması ve aramanın tamamen durmasının önlenmesi için kullanılan temel bir mekanizmadır. Mutasyon, her gen üzerinde düşük bir olasılıkla yapılan rastlantısal değişiklikler yoluyla yeni allel değerlerinin popülasyona geri kazandırılmasını sağlamaktadır. Böylece, çaprazlama sırasında kaybolan kritik genetik bilginin sistemden tamamen silinmesi engellenmektedir. Holland'ın klasik yaklaşımında mutasyon, popülasyonun yerel bir en iyiye sıkışmasını önleyen arka plan bir operatör olarak değerlendirilmekte; düşük mutasyon oranları aramanın istikrarlı ilerlemesini sağlarken, aşırı mutasyon ise biriken genetik bilginin bozulmasına neden olarak algoritmanın yönlendirici gücünü zayıflatabilmektedir. Bu nedenle mutasyon hem çeşitliliği koruyan hem de adaptif sürecin sağlıklı bir şekilde devam etmesine imkan veren tamamlayıcı bir bileşen olarak kritik bir rol oynamaktadır (Ackley, 1985, s.123 ; Holland, 1992, s. 37-38).

3.4. Genetik Algoritmanın Çalışma Prensipleri

Genetik algoritmaların klasik akış yapısı, evrimsel hesaplama çerçevesi üzerine inşa edilmiştir ve sonraki çalışmalarda standart biçimini kazanmıştır (Holland, 1975; Goldberg,

1989; Mitchell, 1996). Bu yapı, bir eniyileme probleminin ardışık nesiller boyunca giderek daha iyi çözümlere doğru yönlendirilmesini sağlayan bir dizi temel adımdan oluşmaktadır. Genetik algoritmanın çalışma prensibi, yüzeyde basit görünmesine rağmen, aslında popülasyon dinamikleri ve evrimsel operatörlerin etkileşimiyle şekillenen oldukça zengin bir arama sürecidir. Aşağıda verilen adımlar hem genel GA çerçevesini hem de GSP'deki uygulama mantığını birlikte anlamayı kolaylaştıran, literatürde standart olarak kabul edilen süreci temsil etmektedir (Mitchell, 1996 , s.10):

- **Adım 1 :**Başlangıç popülasyonunun oluşturulmasıdır. Çözülecek problem için temsil edilen aday çözümler tanımlandıktan sonra, her biri sabit uzunlukta kromozomlardan oluşan rastgele bir başlangıç popülasyonu oluşturulmaktadır. Bu popülasyon, arama sürecinin ilk neslini temsil etmektedir.
- **Adım 2 :** Uygunluk fonksiyonu değerinin hesaplanmasıdır. Popülasyondaki her bir kromozom için, probleme özgü olarak tanımlanan uygunluk fonksiyonu hesaplanır. Böylece her bireyin çözüm kalitesine ilişkin nicel bir ölçüt elde edilmektedir.
- **Adım 3 :**Yeni neslin üretilmesi için popülasyonda istenen birey sayısına ulaşıncaya kadar aşağıdaki alt adımlar tekrarlanmaktadır:
 - **Adım 3a:** Seçim aşamasıdır. Mevcut popülasyondan, seçilme olasılığı uygunluk değeri ile artan şekilde iki ebeveyn kromozom seçilmektedir. Seçim işlemi yerine koymalı yapılmakta; dolayısıyla aynı birey birden fazla kez ebeveyn olarak seçilebilmektedir.
 - **Adım 3b :** Çaprazlama işlemidir. Belirlenen çaprazlama olasılığı altında, seçilen iki ebeveyn rastgele seçilen bir noktadan çaprazlanarak iki yeni kromozom (yavru) üretilir. Çaprazlama uygulanmadığı durumda ise yavrular, ebeveynlerinin doğrudan kopyası olarak oluşturulur.
 - **Adım 3c :** Mutasyon işlemidir. Oluşturulan yavru kromozomların her bir gen pozisyonu, mutasyon olasılığı altında rastlantısal olarak değiştirilmektedir. Bu işlem, genetik çeşitliliği korumak ve çözüm uzayında yeni bölgelerin keşfini sağlamak amacıyla gerçekleştirilmektedir. Mutasyona uğratılan yavrular yeni popülasyona eklenmektedir.

- **Adım 4 :** Popülasyonun güncellenmesi için yeni oluşturulan popülasyon, mevcut popülasyonun yerine geçirilmektedir. Böylece bir nesilden diğerine geçiş tamamlanmış olur.
- **Adım 5 :** Durdurma kriterinin kontrolü yapılır. Algoritma, belirlenen durdurma kriterlerinden (maksimum nesil sayısına ulaşılması, uygunluk değerinde iyileşmenin durması veya hedeflenen çözüm kalitesine erişilmesi) biri sağlanıncaya kadar Adım 2'den itibaren aynı döngüyü tekrar etmektedir.

3.5. Genetik Algoritmanın Gezgin Satıcı Problemine Uyarlanması

Genetik algoritmalar, kombinatoriyel eniyileme problemlerinin çözümünde sağladıkları esneklik, çeşitlilik üretme kapasitesi ve güçlü arama mekanizmaları nedeniyle GSP üzerinde sıklıkla uygulanan metasezgisel yöntemlerden biridir. GSP'nin doğası gereği çözüm uzayı faktöriyel büyüklükte olduğundan, kesin sonuç veren algoritmalar büyük ölçekli problemlerde hesaplama olarak yetersiz kalmakta; bu nedenle GA gibi popülasyona dayalı metasezgisel teknikler daha verimli bir arama yaklaşımı sunmaktadır. GA'nın temel gücü, çözümün yalnızca bir noktasını değil, çok sayıda alternatif turu aynı anda işleyerek çözüm uzayında çeşitliliği koruyabilmesidir. Bu bağlamda, GA'nın keşif ve sömürü dengesini doğal biçimde kurabilmesi, GSP gibi yoğun yerel en iyi içeren problemlerde özellikle avantajlıdır (Toaza vd., 2023).

GA'nın GSP'de uygulanmasındaki kritik aşamalardan biri uygunluk fonksiyonunun tanımlanmasıdır. Uygunluk genellikle turun toplam mesafesine bağlı olarak ters orantılı şekilde tanımlanmakta; daha kısa turlar daha yüksek uygunluk değerleri almaktadır. Bu yapı, doğal seçilimin mekanizmasını taklit ederek iyi turların popülasyonda daha fazla temsil edilmesini sağlamaktadır. Bununla birlikte GSP'nin çok sayıda yerel en iyi içeren karmaşık bir çözüm uzayı yapısına sahip olması, yalnızca uygunluk bazlı yönlendirmenin yeterli olmadığı durumlara yol açabilmektedir. Bu nedenle elitist seçim stratejileri, çeşitliliği koruyan yöntemler ve uyarlanabilir mutasyon oranları, erken yakınsamayı engellemek için sıkça tercih edilmektedir. Doğru tanımlanmamış bir uygunluk fonksiyonu tüm evrimsel sürecin yanlış yönlendirilmesine neden olabilmekte ve bu nedenle GSP'de uygunluk fonksiyonu değerlendirmesi, algoritmanın verimli çalışması açısından en kritik tasarım

unsurlarından biri olmaktadır. GSP’de GA kullanılırken ilk olarak problem bir kromozom yapısına dönüştürülmektedir. Bu kromozomlar başlangıç popülasyonunu oluşturmaktadır. Başlangıç popülasyonu tamamen rastgele oluşturulabileceği gibi, daha hızlı yakınsama elde etmek için en yakın komşu sezgiseli gibi sezgisel çözümlerle desteklenmiş hibrit başlangıç stratejileri de kullanılabilir (Grefenstette vd., 1985, s. 165). Bu seçim, algoritmanın başlangıçtaki çeşitliliğini ve ilk iyileşme hızını doğrudan etkiler. Bu noktada en yaygın yaklaşım, her bir şehir için benzersiz bir gen kullanarak kromozomu bir permütasyon dizisi şeklinde tanımlamaktır. Böyle bir temsil, hem çözümün geçerliliğini korumakta (her şehir bir kez ziyaret edilir) hem de operatörlerin uygulanmasını kolaylaştırmaktadır. Bu temsil, standart ikili kodlamadan farklı olarak özel operatörler gerektirmektedir; çünkü klasik GA operatörleri permütasyon yapısını kolayca bozabilmektedir. GSP gibi permütasyon tabanlı problemlerde seçim operatörünün etkisi daha da belirginleşmektedir. Çünkü permütasyon yapısı gereği popülasyonun genetik çeşitliliğinin hızla kaybolması, çözümlerin aynı kenar yapılarına doğru çökmesine ve tur çeşitliliğinin daralmasına neden olmaktadır. Bu nedenle klasik seçim operatörleri çoğu zaman agresif bir seçim baskısı yaratarak erken yakınsamayı tetikleyebilmektedir.

GSP’de sıralı yapı korunmak zorunda olduğundan özel çaprazlama yöntemleri kullanılmaktadır. Bu operatörlerin de ortak özelliği, çözümün permütasyon yapısını koruyarak ebeveynlerden anlamlı tur parçalarını yeni bireylere aktarabilmesidir. Bu çalışmada GSP için özel olarak geliştirilmiş olan Kenar Birleştirme Çaprazlaması (Edge Recombination Crossover-ERX) ve Sıralı Çaprazlama (Order Crossover- OX) kullanılmıştır.

- ERX, iki ebeveyn turdaki kenar ilişkilerini mümkün olduğunca koruyarak yeni bir tur üretmeyi amaçlayan bir çaprazlama yöntemidir. Operatör, her şehir için ebeveynlerden gelen tüm bağlantıları içeren bir kenar haritası oluşturmakta ve yeni turu bu haritaya dayanarak, bağlantı sayısı en az olan şehirleri öncelikli seçerek inşa etmektedir. Bu strateji, izolasyon riski yüksek şehirleri önce ekleyerek geçersiz yapıların oluşmasını engellemektedir. Ebeveynlerde ortak olmayan bağlantılar tükendiğinde ise algoritma geriye kalan şehirler arasından rastgele bir seçim yaparak tura devam etmektedir. Böylece ERX, ebeveynlerden devralınan kenar yapısını büyük ölçüde koruyan, düşük mutasyonlu ve yapısal açıdan tutarlı çözümler üretmektedir (Whitley, Starkweather ve Fuquay, 1989, s. 133-136).

- OX, iki ebeveyn turun belirli bir alt dizisini koruyarak ve diğer ebeveynlerden gelen şehirlerin görelî sırasını bozmadan yerleştirilerek yeni bir tur oluşturmayı amaçlayan bir permütasyon tabanlı çaprazlama yöntemidir. Operatör, ilk olarak ebeveynlerden birinde iki kesme noktası seçerek bu aralıktaki alt-turu doğrudan yavruya kopyalamakta ve diğer ebeveyn turu kesme noktasından itibaren dolaşmaktadır. Bu ebeveyndeki şehirler, tekrar edenler atılarak, kendi sıraları korunacak şekilde yavrudaki boş pozisyonlara yerleştirilmektedir. Böylece OX, hem ebeveynlerden birinin yapısal bir parçasını korur hem de diğer ebeveyndeki görelî ziyaret sırasını bozmadan bütünleyici bir yapı üretmektedir. Bu özellikleri sayesinde OX, özellikle GSP gibi permütasyon temelli problemlerde geçerli ve sıralı çözüm üretmek için sıklıkla tercih edilen bir çaprazlama operatörüdür (Hussain, Muhammad ve Sajid, 2018, s. 3–4).

GSP’de mutasyonun rolü diğer problemlere kıyasla daha merkezidir. Çünkü permütasyon temelli bir çözümde küçük bir mutasyon bile turun yapısal bütünlüğünü doğrudan etkilemektedir. Bu mutasyon türleri, aslında 2-opt benzeri küçük iyileştirmeler sağlayarak hem çeşitliliği korur hem de arama sürecinde mikro düzeyde bir yerel arama etkisi yaratmaktadır. Bu yapısı nedeniyle mutasyon, GSP’de yalnızca çeşitlilik artırıcı değil, aynı zamanda çözüm kalitesini doğrudan geliştiren bir bileşendir (Samanta, Nandi ve De 2021). Bu çalışmada GSP için uygun olan ters çevirme operatörü kullanılmıştır:

- Ters çevirme mutasyon operatörü, GSP gibi permütasyon temelli problemlerde sıklıkla kullanılan temel mutasyon operatörlerinden biridir. Operatör, kromozom üzerinde iki pozisyon seçerek bu iki nokta arasındaki şehir sırasının ters çevrilmesine dayanmakta; böylece turun genel yapısını bozmadan yerel bir yeniden düzenleme yapmakta ve yeni komşu çözümler üretmektedir. Bu tersine çevirme işlemi, genetik algoritmanın çeşitliliğini artırarak erken yakınsamayı azaltmakta ve özellikle GSP’de etkili bir iyileştirme mekanizması sağlamaktadır (Hassanat vd., 2016, s. 2).

Genetik algoritmaların GSP üzerindeki performansı, yerel arama teknikleri ile birleştirildiğinde belirgin şekilde artmaktadır. GA’yı sezgisel ile güçlendiren memetik algoritma yaklaşımının temel çalışma prensibi, çaprazlama ve mutasyon sonucu elde edilen

yeni bireylerin, popülasyona eklenmeden önce 2-opt, 3-opt veya Lin–Kernighan gibi güçlü yerel iyileştirme algoritmalarıyla optimize edilmesidir. Böylece GA'nın ürettiği ham çözümler yerel ölçekte iyileştirilerek daha kaliteli bireyler elde edilmektedir. Memetik GA'ların GSP'deki başarısının temel nedeni, çözüm uzayının çok geniş ve çok modlu yapısının yalnızca küresel arama ile etkili şekilde taranamayacak kadar karmaşık olmasıdır.

Yerel arama operatörleri özellikle permütasyon tabanlı problemler için yüksek başarı göstermekte; kısa kenar değişimlerinin turun maliyetine etkisi kolay hesaplanabilmekte ve iyileştirme süreci hızlı ilerlemektedir. Pek çok çalışma, 2-opt destekli GA'nın, klasik GA'ya göre hem yakınsama hızında hem de sonuç kalitesinde belirgin üstünlük sağladığını göstermektedir. Ayrıca memetik yaklaşım, GA'nın erken yakınsamaya yatkın doğasını da önemli ölçüde dengeler. Yerel arama ile iyileştirilen bireyler, popülasyon içindeki çeşitliliği artırır ve GA'nın yalnızca ebeveyn genetiğine bağlı kalmadan yeni arama bölgelerine ilerlemesine yardımcı olur. Dolayısıyla memetik GA yapıları, özellikle büyük ölçekli TSPLIB örneklerinde hem hesaplama verimliliği hem de çözüm kalitesi açısından en çok tercih edilen GA türlerinden biri haline gelmiştir (Merz ve Freisleben, 1999).

Genetik algoritmalar GSP gibi karmaşık ve çok modlu problemlerde güçlü bir arama mekanizması sunsa da belirli yapısal zorluklar nedeniyle bazı durumlarda sınırlı performans gösterebilmektedir. Bunların başında erken yakınsama sorunu gelmektedir. Popülasyondaki daha iyi uygunluk değerine sahip bireylerin kısa sürede baskın hale gelmesi, genetik çeşitliliğin azalmasına ve arama sürecinin geniş çözüm uzayının yalnızca dar bir bölgesine sıkışmasına yol açabilmektedir.

Bir diğer sınırlılık parametre hassasiyetinin yüksekliğidir. Popülasyon büyüklüğü, çaprazlama ve mutasyon oranı, seçim mekanizması ve elitizm düzeyi gibi unsurlar, algoritmanın arama davranışını doğrudan etkilemektedir. Küçük popülasyonlar çeşitliliği hızla kaybederken, çok büyük popülasyonlar hesaplama maliyetini artırarak özellikle büyük ölçekli GSP örneklerinde verimliliği azaltabilmektedir. Bunun yanı sıra, GSP'nin permütasyon yapısı nedeniyle GA'nın klasik bit temelli operatörlerinin doğrudan uygulanamaması GA'nın GSP'ye adaptasyonunu hem daha teknik hem de daha kırılgan hale getirir.

4. KARADUL ÖRÜMCEK ALGORİTMASI

Karadul örümceklerinin ilk olarak tanımlanması eski bir tarihe dayanmaktadır. *Latrodectus mactans* (Fabricius) adlı örümcek, günümüzde yaygın olarak karadul ismi ile bilinmekte ve ilk kez 1775 yılında Fabricius tarafından Amerika'dan tanımlanmış, 'Aranea mactans' olarak isimlendirilmiştir (Fabricius, 1775, s. 432). Temelde kanibalizm (yamyamlık) davranışı göstermesi ile bilinmektedir. Bu örümcek türü ismini çiftleşme sonrasında erkeği yemesi nedeni ile almaktadır. Buna cinsel yamyamlık denmektedir. Yamyamlık davranışının bir diğeri ise yönü ise yumurtaların çatlamasıyla birlikte kardeşler arasında yamyamlık süreci başlaması ve zayıf kardeşlerin tüketilmesidir. Bazı durumlarda, bu yavrular annelerini bile yiyebilmektedir. Bu tür yamyamlık bazı örümcek türlerinde görülmektedir. Bu nadir yamyamlık biçimi anneyi tüketme (matrphagy) olarak tanımlanmıştır (Evans, Wallis ve Elgar, 1995, s. 299).

Hayyolalam ve Kazem, karadul örümceğinin yaşam tarzını matematiksel bir model olarak ele alarak 2020 yılında Karadul Örümcek algoritmasını önermiştir. Bu algoritma, örümcek popülasyonunun üreme ve gelişim sürecinde hem makroskobik hem de mikroskobik düzeydeki kuralları simüle ederek eniyileme problemlerinin en iyi çözümünü arar. Makro düzeyde, karadul örümceklerin üreme ve yamyamlık davranışları Darwin'in evrim teorisindeki "en uygun olanın hayatta kalması" ve "güçlünün üstünlüğü" düşüncelerini yansıtır. Diğer yandan, örümcek popülasyonundaki mutasyon, mikroskobik düzeyde genetik manipülasyonu temsil eder. Bu üreme ve gelişim mekanizması sayesinde örümcek popülasyonu giderek daha rekabetçi hale gelir (Hayyolalam ve Kazem, 2020, s. 1).

Karadul örümceği eniyileme algoritmasının temeli de bu yamyamlık davranışına dayanmaktadır. Bu örümceklerin yamyamlık davranışı göstermesi aslında bir hayatta kalma davranışı olarak değerlendirilebilmektedir. Yamyamlık besin yetersizliğinden doğmaktadır. Bu bir doğal seçim sürecini temsil etmektedir. Darwin'in evrim teorisine göre doğal seçim işleyecek ve güçlü olan birey adaptasyon süreci ile hayatta kalacak, diğerleri ise elenecektir (Akgündüz, 2022, s. 279).

Darwin'in doğal seçilimi örnek olarak geliştirilmiş genetik algoritmanın ortaya çıkması ile geliştirilmeye başlanmış evrimsel tabanlı algoritmalar son yıllarda araştırma ilgisinin artmasıyla, sürekli gelişim gösteren güçlü metasezgisel yöntemler arasında yer almaktadır.

“Aslında, evrimsel algoritmaların ya da popülasyon tabanlı algoritmaların büyük bir kısmı genetik algoritmaya benzemektedir; bu nedenle bazı araştırmacılar bu algoritmaları ‘Genetik-tipi algoritmalar’ olarak adlandırmaktadır (Hayyolalam ve Kazem, 2020, s. 1)”. Her yıl bu algoritmalara yönelik çeşitli iyileştirme ve modifikasyon önerileri sunulmakta, bu da onların dinamik ve güncel kalmasını sağlamaktadır (Slowik ve Kwasnicka, 2020, s. 12375). Bu yöntemlerden biri de temelde evrimsel algoritmalar sınıfına giren karadul örümceği eniyileme algoritmasıdır.

BWO’nun çalışma prensibi temelde GA’nın çalışma prensibine benzemektedir. Çoğalma mekanizması ve üreme, GA’da çoğalma, klasik genetik operatörler olan seçim, çaprazlama ve mutasyon aracılığıyla gerçekleştirilmektedir (Çolak, 2010, s. 426). BWO’da ise çoğalma süreci, karadul örümceklerinin eşleşme ve yamyamlık davranışlarından ilham almaktadır. Bireyler rastgele eşleşmekte, çaprazlama ile yeni yavrular üretilmekte ve ardından yamyamlık aşaması ile zayıf bireylerin sistematik olarak elenmesi sağlanmaktadır. Bu yamyamlık mekanizması, popülasyonun genel kalitesini artırmaya ve arama sürecinin etkinliğini yükseltmeye yönelik benzersiz bir stratejidir.

Hayyolam ve Kazem (2020), algoritmanın en iyi çözüm bulma etkinliğini 51 farklı benchmark fonksiyonu ve 3 adet gerçek mühendislik problemlerinde test etmiştir. Elde edilen sonuçlar küresel en iyiye ulaşmada yüksek doğruluk ve zaman maliyeti açısından hızlı yakınsama ile dikkate değer bir yüksek performans göstermiştir. BWO algoritmasının orijinal çalışmasında GA, parçacık sürü optimizasyonu (PSO), yapay arı kolonisi algoritması (ABC) ve biyocoğrafya tabanlı optimizasyon (BBO) ile yapılan karşılaştırmalarda BWO’nun test fonksiyonlarının büyük çoğunluğunda ($\approx 85\%$) en iyi veya en iyiye yakın sonuçlar ürettiği görülmüştür. Özellikle yakınsama hızında belirgin üstünlük gösteren BWO, cinsel ve kardeş yamyamlığı sayesinde popülasyonu agresif biçimde iyileştirerek GA ve ABC’nin üzerinde, PSO ve BBO’dan ise genel olarak daha iyi bir performans ortaya koymuştur. Bu sonuçlar, BWO’nun hem keşif kapasitesi hem de sömürü gücü açısından güçlü bir metasezgisel olduğunu ve farklı problem tiplerinde rekabetçi sonuçlar ürettiğini göstermektedir. Çalışma sonucunda BWO, GA’a göre tüm boyutlarda daha hızlı yakınsama sağlamıştır. PSO bazı fonksiyonlarda hızlı başlangıç yapsa da BWO’nun ulaşabildiği değerin uzağında kalmıştır. ABC, PSO’dan daha iyi performans göstermiş ancak BWO’dan görünür şekilde daha aşağıda kalmıştır. Son olarak BBO, rekabetçi bir performans sergilemiş ancak yine de BWO’dan

daha düşük uygunluk değeri elde etmiştir. BWO büyük ölçekli problemlerde dahi performansını koruyan az sayıda algoritmadan biridir.

Problemin boyutu büyüdükçe GA, PSO ve ABC belirgin şekilde bozulurken, BWO'nun sonuçları kararlı kalmakta ve en düşük hata değerlerini üretmektedir. CEC 2005 kompozit fonksiyonları üzerinde yapılan deneylerde BWO, GA, PSO, ABC ve BBO dahil tüm rakip algoritmalara üstünlük sağlamıştır. Bu fonksiyonlar çok modlu, kaydırılmış ve döndürülmüş yapıda olduklarından metasezgisel algoritmalar için zorluk seviyesi yüksektir. Buna rağmen BWO'nun ulaştığı en küçük değerler, algoritmanın karmaşık arama yüzeylerinde bile kararlı ve etkili olduğunu göstermektedir (Hayyolam ve Kazem, 2020).

Sonuç olarak, BWO'nun hem keşif hem sömürü aşamalarında güçlü bir arama kapasitesine sahip olduğunu ve karmaşık problemlerde yüksek performans sunduğunu doğrulamaktadır. Çalışmada şu sonuçlar elde edilmiştir: Üreme sisteminde ebeveynlerin rastgele seçimi arama alanının tümünün keşfini garanti etmektedir. Üreme adımında çok sayıda yavru üretimi bunu desteklemektedir. Bu adım aynı zamanda yerel en iyi tuzağından kaçmayı sağlamaktadır. Kötü çözümleri ortadan kaldıran yamyamlık mekanizması algoritmanın en iyiye doğru çok hızlı yakınsamasını sağlamaktadır. Bu adım agresif sömürü sistemini garanti etmektedir. Mutasyon adımı ise keşfi desteklemekte olup, iki adımın birlikte kullanılması keşif ve sömürü dengesini sağlamaktadır (Hayyolam ve Kazem, 2020, s. 26). Algoritmanın çalışma prensibi Hayyolam ve Kazem'in algoritmayı sunduğu makale baz alınarak anlatılacaktır.

4.1. Karadul Örümceği Optimizasyon Algoritması'nın Çalışma Prensibi

Karadul örümceklerinin yaşam döngüsü üreme ve hayatta kalma süreçleri boyunca oldukça agresif ve seçici bir biyolojik mekanizmadır. BWO algoritması bu doğal davranışları matematiksel bir kurala oturtarak popülasyon temelli bir eniyileme süreci oluşturmaktadır. Temelde algoritma karadul örümceğinin çiftleşme ve yamyamlık davranışını modellemektedir. Sistemde üç farklı yamyamlık mekanizması bulunmaktadır. Erkek ve dişi örümceğin çiftleşmesi ile yeni bireyler üretilmektedir. Çiftleşme sırasında ya da sonrasında dişi erkek bireyi yemektedir. Buna cinsel yamyamlık adı verilir. Yumurtadan çıkan yavrular arasında ise kardeş yamyamlığı söz konusudur. Zayıflar elenmekte ve güçlüler hayatta

kalmaktadır. Bazı durumlarda yavruların anneyi yemesi bile söz konusu olmaktadır. Bu temelde daha önce de söz edildiği gibi Darwin'in güçlü olanın hayatta kalması prensibine dayanmaktadır. Algoritmada bu döngü her nesilde kötü olan çözümlerin agresif biçimde elenmesi ve iyilerin korunmasını sağlamaktadır. Böylece ana popülasyon her nesilde daha güçlü çözümlere doğru evrim geçirmektedir. Tablo 4.1.'de algoritmanın sözde kodu verilmiştir.

Tablo 4.1 Karadul Örümcek Optimizasyonu Sözde Kodu

Girdi: Maksimum nesil sayısı, popülasyon büyüklüğü, yamyamlık oranı, mutasyon oranı
Çıktı: Amaç fonksiyonu için yaklaşık-optimal çözüm
<pre>// Başlatma 1. Karadul örümceklerinden oluşan başlangıç popülasyonu oluşturulur Her birey, D boyutlu bir problemin aday çözümünü temsil eden D boyutlu bir dizidir // Sonlandırma koşuluna kadar döngü 1. Üreme oranına göre, üretilecek yavru sayısı "nr" hesaplanır 2. Üreme oranına göre, popülasyondaki en iyi nr çözüm seçilir ve pop1'e kaydedilir 3. i = 1'den nr'ye kadar döngü başlatılır 3.1. pop1 içinden rastgele iki çözüm seçilerek ebeveynler belirlenir 3.2. Mutasyon oranına göre bazı çocuklar (yeni aday çözümler) üretilir 3.3. Baba birey yok edilir 3.4. Yamyamlık oranına göre bazı çocuklar yok edilir 3.5. Kalan çözümler pop2'ye kaydedilir 4. Döngü sonu 5. Mutasyon işlemi Mutasyon oranına göre, mutasyona uğrayacak çocuk sayısı "nm" hesaplanır 6. i = 1'den nm'ye kadar döngü başlatılır 12. pop2 içinden rastgele bir çözüm seçilir 13. D boyutlu çözümün i'inci konumu için, mutasyon oranına göre yeni çözüm üretilir 14. Yeni çözüm pop3'e kaydedilir 7. Döngü sonu 8. Güncelleme 9. pop = pop1 + pop2 + pop3 olarak güncellenir 10. En iyi çözüm döndürülür</pre>

- **Adım 1:** Popülasyonun oluşturulmasıdır. Her aday çözüm algoritmada bir “widow” (birey) olarak tutulmaktadır. Sürekli bir problem için eşitlik 4.1’de her bir birey x_i karar değişkeni, N_{var} karar değişkenleri sayısı olmak üzere çözüm vektörü şu şekilde tanımlanmaktadır:

$$Widow = [x_1, x_2, \dots, x_{N_{var}}] \quad (4.1)$$

GSP türü bir probleme uyarlandığında bu dizi, şehir permütasyonu olarak düşünülebilmektedir. Ancak temel mekanizması aynı kalmaktadır. Belirlenen büyüklükte başlangıç popülasyonu üretilmekte ve her bir birey için karar değişkenleri problem tanımına göre rastgele seçilmektedir. Elde edilen her birey için uygunluk değeri hesaplanmaktadır. Bu popülasyon ilk nesil örümcekleri/bireyleri temsil etmektedir. Eşitlik 4.2’de f değeri amaç fonksiyonudur:

$$Fitness = f(Widow) = f(x_1, x_2, \dots, x_{N_{var}}) \quad (4.2)$$

Uygunluk değerlerinin hesaplanmasının ardından algoritmanın başlatılabilmesi için $N_{pop} * N_{var}$ ⁹ boyutlu bir matris oluşturulmaktadır.

- **Adım 2:** Çiftleşme ve yeniden üretimdir. Bu aşama karadul örümceğinin üreme davranışını taklit etmektedir. Popülasyondan üreme oranı parametre değeri kadar birey seçilmektedir. Bu bireyler rastgele eşleştirilerek ebeveyn çifti oluşturulmaktadır. Çiftler bağımsız ve paralel olarak üreme işlemi gerçekleştirirler. örümcek dizisi uzunluğunda bir α dizisi oluşturulmaktadır. Her çift için GA’da bulunan çaprazlama benzeri bir işlem uygulanarak yavrular oluşturulmaktadır. Yavrular şu denklem kullanılarak üretilmektedir:

$$y_1 = \alpha * x_1 + (1 - \alpha) * x_2 \quad (4.3)$$

$$y_2 = \alpha * x_2 + (1 - \alpha) * x_1 \quad (4.4)$$

Eşitlik 4.3’te ve 4.4’te y_1 ve y_2 yavruları, x_1 ve x_2 ebeveynleri temsil etmektedir. Bu işlem $N_{var}/2$ kez tekrarlanmaktadır ve rastgele seçilen sayılar tekrar kullanılmamaktadır. Baba bu süreçte cinsel yamyamlık mekanizması ile sistem dışına atılmaktadır. Anne ve yavrular yeni bir popülasyona eklenmekte ve uygunluk değerlerine göre sıralanmaktadır.

- **Adım 3:** Yamyamlık sürecidir. Algoritmada üç tip yamyamlık söz konusudur. İlki cinsel yamyamlık olup dişi (anne) yani daha iyi uygunluk değerine sahip bireyin diğer

⁹ (N_{pop} : popülasyon boyutu)

ebeveyn olan babayı yemesidir. Çiftleşme adımıda anne ve yavrulardan oluşturulan popülasyona yamyamlık oranı kullanılarak, orana göre hayatta kalacak bireylerin sayısı hesaplanmaktadır. Bazı durumlarda ise üçüncü tür yamyamlık gözlenebilmekte ve güçlü yavrular dişi tüketmektedir.

- **Adım 4:** Mutasyon işlemidir. Bu aşamada popülasyondan rastgele Mutepop(mutasyon yapılacak birey sayısı) kadar birey seçilmekte ve seçilen çözümdeki iki eleman rastgele yer değiştirmektedir. Mutepop değeri mutasyon oranına göre hesaplanmaktadır.
- **Adım 5:** Diğer metasezgisel algoritmalarda da olduğu gibi; önceden belirlenen nesil sayısına ulaşılması, belirlenen en iyiye ulaşması veya uzun nesil sayısı boyunca iyileşme olmaması durumunda algoritma sonlandırılır.

4.2. Parametrelerin Ayarlanması

Önerilen algoritmada temel olarak kullanılan parametreler üreme oranı, yamyamlık oranı ve mutasyon oranıdır. Bu üç parametrenin doğru şekilde ayarlanması algoritmanın başarısı için kritiktir. Yerel en iyiden çıkma ve çözüm uzayının tamamının keşfedilebilmesi için parametreler ayarlarken keşif ve sömürü dengesi göz önüne alınmaktadır.

- Üreme oranı (pp-procreating rate); üremeye katılacak bireylerin yüzdesini belirlemek için kullanılmaktadır. Bu parametre ile çeşitli yavruların üretilmesi sağlanarak çeşitlilik kontrol altında tutulmaya çalışılmaktadır. Böylece çözüm uzayı daha hassas olarak keşfedilmektedir.
- Yamyamlık oranı (cr-cannibalism rate); bu oran yamyamlık sisteminin kontrol oranıdır. Popülasyondan uygun olmayan bireyler çıkartılarak, algoritmanın sömürü aşamasında yüksek bir performans göstermesini sağlamaktadır.
- Mutasyon oranı (pm;-mutaton rate); mutasyon uygulanacak bireylerin yüzdesini belirlemektedir. Bu parametre keşif ve sömürü dengesini sağlamaktadır.

Bu üç parametre birlikte bir denge kuracak şekilde çalışmaktadır. Çalışma incelendiğinde üreme oranının çözüm uzayının yeni alanlarını keşfetmesini sağladığı, yamyamlık oranının daha iyi bireyleri seçerek sömürüyü desteklediği ve mutasyonun ise çeşitliliği tazelemek için kullanıldığı görülmüştür. Bu parametreler arasındaki dengenin

kurulması algoritmanın arama davranışını belirlemektedir. Doğru parametre ayarları ile çeşitliliği korurken en iyi çözüme hızla yaklaşılmakta ve yerel en iyilerde takılma riski azalmaktadır. Yanlış parametre ayarlanması ise algoritmanın rastgele aramalar yapmasına, popülasyon çeşitliliğinin erkenden yok olmasına ya da çözüm kalitesinin düşmesine neden olmaktadır.



5. PARALEL ADA MODELLİ HİBRİT KARADUL ÖRÜMCEK OPTİMİZASYONU (PH-BWO)

5.1. Amaç ve Kapsam

Gezgin Satıcı Problemi gibi NP-Zor yapıda olan karmaşık eniyileme problemlerinde, tek bir metasezgisel yöntemin hem keşif hem de sömür gereksinimlerini aynı anda karşılaması çoğu zaman mümkün olmadığından, hibrit yaklaşımlar önemli avantajlar sağlamaktadır. Bu bağlamda çalışmada Paralel Ada Modelli Karadul Örümcek Optimizasyonu olarak adlandırılan yöntem GSP'ye yönelik olarak geliştirilmiş olup ve çok katmanlı hibrit mimarisiyle farklı arama stratejilerini bütünleştirmektedir. Geliştirilen algoritmanın tasarımı, her bileşenin güçlü yönlerini bir araya getirirken zayıflıklarını diğer bileşenlerle dengeleyen, dolayısıyla kararlı bir keşif ve sömür dengesi sunan bir arama yapısı sağlamaktadır. Bu doğrultuda model; BWO'nun agresif seçim gücünü, Ada modelinin küresel çeşitlilik ve hız avantajlarını, GA operatörlerinin yapısal dönüşüm kabiliyetini ve 2-opt yerel aramanın iyileştirici etkisini birlikte kullanarak çok düzeyli bir arama mekanizması ortaya koymaktadır.

Temel yapı olarak BWO'nun seçilmesinin motivasyonu, algoritmanın doğal yapısındaki güçlü ve agresif seçim mekanizmalarının GSP üzerinde avantaj sağlamasıdır. BWO'da görülen cinsel ve aile içi yamyamlık süreçleri kalitesiz bireyleri hızlıca elerken, bu yüksek seçim baskısı tek bir yerel en iyi çevresinde erken yakınsamaya yol açabilmekte ve hibrit modelde ada modeli ve yerel arama yapıları denge için kullanılmaktadır. Böylece hem çeşitlilik korunmakta hem de çözüm uzayının farklı bölgeleri araştırılabilmektedir.

Ada Modeli, popülasyonu kısmen yalıtılmış alt popülasyonlara ayırarak her birinin bağımsız evrimsel süreç yürütmesine olanak tanımaktadır. Adalar belirli aralıklarla birey değişimi yaparak göç mekanizması aracılığıyla kontrollü bilgi paylaşımı sağlamakta; bu yapı hem erken yakınsamayı engellemekte hem de çok çekirdekli işlemcilerle doğal uyum sayesinde özellikle büyük ölçekli problemlerde önemli hızlanma sunmaktadır. Bu sayede geniş popülasyonların eşzamanlı işlenmesi ve daha derin arama yapılması mümkün hale gelmektedir.

Her ada kendi içinde bir yerel en iyiye yakınsayabilse de sistemdeki diğer adalar farklı bölgeleri araştırmayı sürdürmektedir. Periyodik göç, elit bireylerin yanı sıra rastlantısal

çeşitliliğin de adalar arasında dolaşımına imkan tanıyarak yöntemin dengeli bir evrimsel dinamik göstermesini sağlamaktadır. Böylece BWO'nun doğal olarak yüksek olan seçim baskısının yol açabileceği çeşitlilik kaybı telafi edilmekte, GA operatörleriyle bütünleşen hibrit yapı ise keşif ve sömürü dengesini daha kararlı bir şekilde yönetebilmektedir.

BWO ve GA operatörleri küresel aramayı destekleyen sıçrama niteliğinde hareketler üretse de oluşturulan bireyler çoğu zaman ham yapıda olmaktadır. Bu nedenle modele memetik bir adım olarak 2-opt yerel araması eklenmiş; bu adım bireyleri hızlı biçimde kendi yerel en iyilerine taşımakta ve en yakın komşu aday listelerinin kullanımıyla yalnızca umut vadeden kenar değişimlerinin değerlendirilmesi sağlanarak yöntem hız ve kalite dengesi güçlendirilmiştir.

Küçük, orta ve büyük ölçekli TSPLIB örnekleri üzerinde gerçekleştirilen deneysel analizlerde yöntemin performansı literatürdeki bilinen en iyi değerlere göre değerlendirilmiş; küçük ve orta ölçekli örnekler en iyiye ulaşma kapasitesini, büyük ölçekli örnekler ise hesaplama verimliliğini ve kararlılığı ortaya koymuştur. Bu deney tasarımı hibrit modelin çözüm kalitesini ve ölçeklenebilirliğini bütüncül biçimde incelemeye olanak tanımış, elde edilen sonuçlar GSP'ye uyarlanmış diğer hibrit BWO yöntemleriyle karşılaştırılarak modelin görece üstünlükleri belirlenmiştir.

Tüm bu bileşenler ışığında geliştirilen hibrit modelin temel amacı; BWO'nun agresif seçim gücünü, 2-opt yerel aramasının iyileştirme kapasitesini ve ada modelinin çeşitlilik ve hız avantajlarını bir araya getirerek TSPLIB problemlerinde bilinen en iyiye çok yakın veya doğrudan bilinen en iyi çözümleri üreten yüksek kaliteli ve ölçeklenebilir bir arama davranışı oluşturmaktır. Çalışma bu doğrultuda, GSP gibi zorlu eniyileme problemlerinin kabul edilebilir süreler içinde çözülmesini mümkün kılan etkili bir hibrit eniyileme çerçevesi sunmaktadır.

Bu çalışma iki temel özgün katkı sağlamaktadır. İlk olarak, literatürde sınırlı sayıda hibrit örneği bulunan BWO'nun yamyamlık temelli agresif seçim mekanizması, GSP'nin farklı ölçeklerdeki davranışı göz önüne alarak GA ve memetik adımla birlikte ada modeli uygulanarak keşif ve sömürü dengesini iyileştiren yeni bir hibrit yapı ortaya koymaktır. İkinci olarak, Karadul Örümcek Optimizasyonu tabanlı hibrit metasezgisel yaklaşımlar GSP için Türkçe literatürde temsil edilmediğinden, çalışma alana içerik ve terminoloji düzeyinde önemli bir katkı sunmaktadır.

5.2. Algoritma Bileşenlerinin Seçilimi ve Nedenleri

PH-BWO tasarımında seçilen bileşenler ve bu bileşenlerin neden kullanıldığı bu bölümde detaylıca incelenecektir.

Algoritmanın başlangıç popülasyonu, belirlenen oran kadar en yakın komşu sezgiseli ile, kalan kısmı ise rastgele turlar ile oluşturulmaktadır. Bu tercih, arama sürecini hızlandırırken çok erken körleşmeden çeşitliliği korumayı, kötü başlangıçlardan uzak durmayı ve yerel arama ile genetik operatörlerin daha verimli çalışmasını sağlamayı hedeflemektedir. GSP gibi zor kombinatoriyel problemlerde tamamen rastgele turların seçilmesi, çoğunlukla hem uygunluk değeri açısından küresel en iyiden çok uzak, hem de yapısal olarak dağınık çözümler üretmekte; bu da yerel arama operatörlerinin anlamlı bir iyileştirme yapabilmesi için fazladan zaman harcamasına neden olmaktadır. Buna karşın, en yakın komşu sezgiseli ile başlatılan turlar, her şehirde en yakın komşuya gideceğinden popülasyona çözüm uzayının görece iyi bölgelerini temsil eden makul kalitede çözümleri en baştan kazandırmaktadır. Ancak popülasyonun tamamını bu sezgisel ile oluşturmak başlangıç popülasyonunu tekdüze hale getirerek çeşitliliği azaltacağından, rastgele turlar çözüm uzayının farklı bölgelerine erişimi sağlarken, sezgisel turlar çaprazlama ve mutasyonun tüketebileceği yapı taşlarını sunmaktadır. Böylece keşif ve sömürü dengesi algoritmanın başlangıcında kurulmakta, yerel arama bu nispeten iyi başlangıçların etrafında hızlı iyileştirmeler yapabilmekte ve ada modeli bağlamında her ada kendi içinde hem görece daha iyi çözümler hem de rastgele çözümler barındırmaktadır.

Göç mekanizması da bu yapıyı desteklemekte; göç sırasında iyi çözümler adalar arasında taşınırken, farklı adalarda rastgele üretilen bireyler yeni arama alanları açmaktadır. Aynı yaklaşım, algoritmanın kopya temizliği aşamasında da kullanılmakta; kopya bireyler yerine aynı sistemle yeni bireyler eklenerek kalite ve çeşitlilik dengesi yalnızca başlangıçta değil, evrim süreci boyunca korunmaktadır.

Seçim mekanizması, uygunluk değerine göre sıralanmış popülasyondan, belirlenen parametre oranı kadar en iyi birey alarak daha zayıf kısmın üreme havuzuna katılmasını engellemektedir. Bu tercihin iki temel gerekçesi vardır: İlki, kaliteli yapı taşlarını koruyarak algoritmanın ilerleyen aşamalarında yoğun sömürü yapabilmektir. Zayıf bireyleri üreme havuzuna sokmamak, çaprazlama operatörlerinin hesaplama gücünü iyi çözümler üzerinde

yoğunlaştırarak daha verimli kullanım sağlamaktadır. İkincisi, özellikle büyük problemlerde yerel aramanın maliyetinin yüksek olması nedeniyle, bu maliyeti kontrol altında tutmaktır. Belirli bir elit alt popülasyon üzerinden yavru turları üretmek, yerel aramanın gücünü bu bireyler üzerinde kullanmayı ve yerel aramanın getirisini optimize etmeyi mümkün kılmaktadır. Böylece algoritmanın tasarımında amaçlanan, iyi çözümleri koruyan, popülasyonu kilitlemeyen ve yerel arama maliyetini gereksiz yere artırmayan denge mekanizması sağlanmış olmaktadır.

Çaprazlama operatörü, algoritmanın hem çeşitlilik üretme hem de iyi çözüm parçalarını bir sonraki nesle aktarma kapasitesini belirleyen temel mekanizmalardan biridir. GSP gibi permütasyon temelli problemlerde klasik GA çaprazlamalarının şehir tekrarına, eksik şehirlere veya yapısal bozulmalara yol açabilmesi nedeniyle, bu çalışmada özellikle GSP için geliştirilmiş iki operatör tercih edilmiştir. ERX ve OX'in birlikte kullanılmasıyla, iyi kenarların korunması ile sömürü ve yeni tur düzenlerinin oluşturulması ile keşif aynı anda gerçekleşmektedir. ERX iyi komşuluk ilişkilerini aktarırken, OX ebeveynlerde bulunmayan yeni şehir blok dizilimlerinin ortaya çıkmasını sağlamakta ve özellikle orta ve büyük ölçekli problemler için gerekli olan geniş varyasyon spektrumunu desteklemektedir. Bu iki operatörün birlikte kullanılması algoritmanın hem sömürü hem keşif mekanizmalarını dengeli biçimde beslemektedir. Çaprazlama operatörü seçimlerinin ada modeline uyumunun temel nedeni, üremenin elit bireyler üzerinden yürütülmesi ve her üreme döngüsünde anne ile çocuklar arasında aile içi bir seçim yapılmasıdır. Çaprazlamayla gelen iyi kenarlar çocuklara taşındığında, aile içi değerlendirme sırasında uygunluk değeri daha iyi olan çocuklar anneye karşı rekabet edebilmekte; bu da algoritmanın yalnızca rastgele mutasyonla değil, güçlü yapısal yeniden kombinasyonlarla gelişmesini sağlamaktadır. ERX ve OX'in oluşturduğu yapısal bütünlük, 2-opt tabanlı yerel arama ile desteklenmiş; 2-opt, çaprazlamanın ürettiği turları düşük maliyetli küçük düzeltmelerle hızla iyileştirmekte ve her yerel arama çalışmasında daha yüksek verimli sonuçların üretilmesine katkı sunmaktadır. Ada modeli içinde her adanın kendi yerel popülasyonunu iyileştirmesi, çaprazlamayla gelen bu iyi yapı taşlarının adalar arasında göç yoluyla aktarılmasıyla küresel düzeyde birleşerek çözüm kalitesini artırır.

Yamyamlık, temelde düşük kaliteli bireylerin sistematik olarak elendiği bir kontrol mekanizmasıdır. GSP gibi yüksek bağlantılı ve karmaşık problemlerde çaprazlama

operatörleri her zaman daha iyi bir tur üretmediğinden, aile içi seçim aşamasında, yamyamlık oranına göre ailenin kötü bireyleri elenmektedir. Bu işlem, rekabetçi seçim ve kalite güvencesi sağlayarak popülasyona zayıf bireylerin taşınmasını engellemekte, ada içi popülasyonun sürekli yenilenmesine yardımcı olmakta ve mutasyon ile yerel aramanın daha etkin çalışmasına imkan tanımaktadır. Hesaplama maliyeti açısından da kritik bir filtreleme mekanizması işlevi görmektedir; yalnızca elit bireylerin çocuk üretmesiyle yetinmeyip, bu çocukların ebeveynlerine kıyasla gerçekten kaliteli olup olmadığını da değerlendiren ek bir kontrol katmanı sunmaktadır.

Mutasyon işlemi de çaprazlama gibi yerel arama ile desteklenmiş ve ada modeli içindeki genetik çeşitliliği koruyan temel mekanizmalardan biri olarak yapılandırılmıştır. Çaprazlama operatörü iyi alt turları birleştirme konusunda güçlü olmakla birlikte, zamanla popülasyonda benzer bireylerin artmasına ve aynı alt döngülerin sürekli dolaşımında kalmasına yol açarak ada içi elitizmi aşırı sömürü yönüne çekebilmektedir. Bu durumda çaprazlama yeni kenarlar üretemez ve çözüm uzayının dar bir bölgesinde sıkışma meydana gelir. Bu sıkışma durumuna karşı tersine çevirme mutasyonu devreye girerek çözüm uzayının farklı bölgelerine geçişi mümkün kılmakta ve çaprazlamanın ulaşamadığı bölgelerin keşfini sağlamaktadır.

GSP için bu operatör; etkili olması, birçok iyi alt turu koruması ve 2-opt ile yapısal uyumu nedeniyle seçilmiştir. Mutasyonun yaptığı kaba değişiklikler, uygulanan 2-opt ile hızlı biçimde iyileştirilmekte; bu ikilinin birlikte kullanımı düşük kaliteli mutasyonların temizlenmesini, iyi mutasyonların ise daha iyi bireylere dönüşmesini sağlamaktadır. Ada modeli popülasyonu alt popülasyonlara ayırdığı için her ada zamanla benzerlik eğilimine girebilir ve göç aralıkları uzun olduğunda bilgi paylaşımı sınırlanabilir. Bu bağlamda mutasyon mekanizması, ada içi çeşitliliği koruyarak adaların farklı arama bölgelerini keşfetmesini, göç anında daha yüksek bilgi kazancı oluşmasını ve böylece küresel en iyiye ulaşma olasılığının artmasını sağlayan temel bir güvence işlevi görür.

2-opt tabanlı yerel arama, bu çalışmada hem teorik gerekçeler hem de algoritmik uyumluluk nedeniyle tercih edilmiştir. GSP için kullanılan klasik yerel arama operatörleri arasında özellikle Lin–Kernighan gibi daha karmaşık yapılar bulunsa da 2-opt, ada modeline uyarlanmış BWO’nun ihtiyaç duyduğu hız ve iyileştirme dengesi açısından en uygun yöntem olarak belirlenmiştir. Hem hızlı uygulanabilir olması hem de rastlantısal çeşitlilikle üretilen

turları anlamlı ölçüde iyileştirebilmesi, özellikle büyük popülasyonların işlendiği yapılar için avantajlı kılmaktadır. Algoritmada çaprazlama sonrası yamyamlık aşamasından kurtulan bireyler ve mutasyon için seçilen bireylerin olasılıksal olarak belirlenmiş bir kısmı yerel aramaya gönderildiğinde, bu bireyler daha kısa turlara yakın konumlara taşınmakta, popülasyondaki kötü gen dizilimleri hızla temizlenmektedir. 2-opt'un hesaplama maliyeti, 3-opt ve Lin-Kernighan gibi daha ileri tekniklere kıyasla oldukça düşük olduğundan, ada içinde her döngüde yüzlerce bireyin işlenmesi gereken senaryolarda ciddi bir performans avantajı sağlamaktadır.

2-opt'un segment tersleme mantığına sahip olması, daha önce belirtildiği üzere tersine çevirme mutasyonu ile doğal bir uyum oluşturarak mutasyonun yaptığı bozulmayı yönlendirilmiş biçimde iyileştirmeye imkan verir ve kontrollü bir çeşitlilik ve iyileştirme döngüsü meydana getirmektedir. Çaprazlamayla üretilen turlar genellikle iyi kenar parçaları taşımakla birlikte yapısal bozukluklar içerebilmekte; 2-opt bu bozuklukları düşük maliyetle düzelterek turu daha makul bir bölgeye çekmekte ve yerel aramayı yalnızca sömürü mekanizması değil, aynı zamanda popülasyon sağlığını koruyan dengeleyici bir unsur haline getirmektedir. BWO'nun biyolojik modelinde yavruların gelişme sürecinde güçlenmesi ne kadar önemliyse, yerel arama uygulanan bireylerin daha sağlam ve rekabetçi çözümlere dönüşmesi benzer şekilde önem arz etmektedir.

Ada sistemi, algoritmanın hem keşif yeteneğini hem de hesaplama yükünü doğrudan etkilemektedir. Popülasyon tek bir büyük kitle halinde çalıştırıldığında yamyamlık mekanizması nedeniyle genetik çeşitlilik hızla azalmakta, iyi bireyler erken baskın hale gelmekte ve algoritma birkaç nesil sonra aynı çözüm bölgesinde dönmeye başlamaktadır. Ada sayısının artırılması, bu problemi yapısal olarak çözerek popülasyonu farklı alt topluluklara bölmekte; her ada kendi içinde bağımsız bir evrim süreci yürütmekte ve çözüm uzayının farklı bölgelerini paralel olarak araştırmaktadır. Bu bağımsızlık, ada sayısı arttıkça aramanın daha geniş bir alana yayılmasını sağlamakta; bir ada erken yakınsamaya düşse dahi diğer adalarda farklı çözümler gelişmekte ve göç mekanizması sayesinde bu çözümler zaman içinde birbirine aktarılmaktadır.

Göç sistemi, ada modelinde yalnızca bir bilgi taşıma mekanizması değil, algoritmanın küresel arama davranışını şekillendiren temel itici güçlerinden biridir. Adalar kendi içlerinde bağımsız evrim süreçleri yürüttükleri için her ada farklı bir çözüm bölgesinde uzmanlaşma

eğilimindedir; kimisi kısa sürede kaliteli turlar üzerinde yoğunlaşırken, kimisi tamamen farklı bir bölgeyi keşfetmektedir. Göç, bu farklı uzmanlık alanlarını birbirine bağlayan ve her adanın öğrendiğini diğer adalara aktaran kolektif bir zeka mekanizması gibi çalışmaktadır. Göç gerçekleştiğinde en iyi bireylerin başka adalara taşınması, gelişmiş çözüm parçalarının adalar arasında dolaşmasını sağlayarak tek bir adanın başarısını tüm sistemin başarısına dönüştürmektedir. Aynı zamanda, rastgele seçilen bireylerin dolaşıma girmesi, zayıf adaların içinde sıkıştığı yerel en iyi senaryolarını kırmakta, popülasyonun tamamen homojenleşmesini engellemekte ve çeşitliliği kontrollü şekilde yeniden dağıtmaktadır. Hangi turun taşınacağı (en başarılı birey mi, rastgele birey mi) ve hangi adaya taşınacağı (komşu ada mı, uzak ada mı) gibi tasarım kararları, algoritmanın evrimsel dengesini belirlemektedir.

Sistem, iyi bireyleri diğer adalara göndererek sömürü mekanizmasını güçlendirirken, rastgele bireylerin dolaşımıyla keşif kapasitesini canlı tutmaktadır. Böylece göç, sadece bir aktarım aracı olmaktan çıkarak popülasyonların birbirini yeniden başlatmasını mümkün kılan bir yenilenme sürecine dönüşmektedir. Sonuç olarak göç sistemi, algoritmanın hem derinlemesine arama yapmasını hem de çözüm uzayının farklı bölgelerini eşzamanlı keşfetmesini sağlayan temel düzenleyicisidir; adalar arasında dolaşan bilgi, tek bir popülasyonla elde edilemeyecek bir çeşitlilik ve dayanıklılık üretmekte, algoritmanın yerel en iyilere takılmadan ilerlemesine ve küresel en iyiye yaklaşma kapasitesinin artmasına katkıda bulunmaktadır.

5.3. Parametrelerin Belirlenmesi ve Etkileşimleri

Popülasyon boyutu, algoritmanın her nesilde aynı anda değerlendirdiği çözüm sayısını ifade etmektedir. Daha büyük bir popülasyon, GSP’de çözüm uzayının daha geniş bir bölümünün aynı anda taranmasını sağlamakta ve çeşitliliği arttırmaktadır. Ancak popülasyonun büyümesi hesaplama maliyetini yükseltmektedir; bu nedenle popülasyon boyutu hem çeşitliliği koruyacak hem de aşırı maliyete neden olmayacak şekilde dengelenmelidir. Özellikle büyük ölçekli problemlerde popülasyon boyutunun doğru seçimi, erken yakınsamayı engelleyen temel bileşenlerden biridir.

Çalışmada popülasyon boyutu GSP’nin literatür araştırmasında kullanılan değerlerden görece olarak küçük belirlenmiştir. Bunun temel nedeni BWO’nun üreme ve yamyamlık mekanizmasının yarattığı doğal eleme baskısıdır. Popülasyonun büyük bir kısmı her nesilde

elenmektedir. Mutasyon ve yerel arama yapısının birleşimi hızlı sömürü sağlamak ve ada modeli sayesinde çeşitlilik zaten korunmaktadır. Bu bağlamda çeşitliliğin sağlanması popülasyon büyüklüğünden ziyade, popülasyon içi yüksek dönüşüm hızından gelmektedir. 2-opt'un sağlamış olduğu yoğun sömürü desteği ve yerel aramanın işlem yükü de göz önüne alınarak büyük popülasyonlarda gereksiz hesap yükünden kaçınılmıştır.

Maksimum nesil, algoritmanın durma koşullarından birini oluşturmada ve kaç evrimsel döngü yapılacağını belirlemektedir. Daha yüksek nesil sayıları, çözümlerin daha uzun süre iyileştirilebilmesine olanak tanımakta; ancak çalışma süresini doğrudan arttırmaktadır. Küçük problemlerde birkaç nesil bile yeterliyken, büyük ölçekli problemlerde yüksek nesil değerleri arama derinliği için kritiktir. Bu parametre, arama süresi ve çözüm kalitesi arasındaki temel denge noktalarından biridir.

Üreme oranı, popülasyondan kaç bireyin yeni neslin oluşumunda ebeveyn olarak seçileceğini belirlemektedir. Yüksek üreme oranı, popülasyonun büyük bir kısmının çaprazlama sürecine katılmasını sağlamakta ve güçlü çözümlerin daha fazla temsil edilmesine yol açmaktadır. Ancak aşırı yüksek bir üreme oranı, çeşitliliği azaltarak popülasyonun hızlı bir şekilde tek bir yapıya yakınsamasına neden olabilmektedir. Bu parametre, sömürü gücünün temel belirleyicilerindendir.

Yamyamlık oranı, BWO'ya özgü güçlü bir seçim mekanizmasını temsil etmekte ve üreme sonrasında zayıf bireylerin popülasyondan ne kadarının eleneceğini belirlemektedir. Bu mekanizma agresif bir sömürü uygulamakta; popülasyon hızla düşük kaliteli bireylerden arındırılmaktadır. Ancak yamyamlığın çok yüksek değerlerde verilmesi, çeşitliliği aşırı azaltarak erken yakınsamaya yol açabilmektedir. Bu nedenle bu parametre diğer parametrelerle birlikte dikkatle dengelenmelidir.

Mutasyon oranı, popülasyondaki bireylerin ne kadarının rastgele bozularak farklı bir çözüm yapısına dönüştürüleceğini belirlemektedir. Mutasyon keşif mekanizmasının temel kaynaklarından biridir ve özellikle yerel aramanın yoğun olduğu senaryolarda popülasyonun durağanlaşmasını engellemektedir. Mutasyon oranı düşük olduğunda popülasyon sıkışabilmekte, çok yüksek olduğunda ise çözümler aşırı rastgeleleşerek kalite kaybı yaşanabilmektedir. GSP literatüründe kullanılan düşük mutasyon oranlarının burada tercih edilmeme nedeni yamyamlık sisteminin kaybettiği çeşitliliğin desteklenmesi amacını taşımaktadır.

Yavru bireylere uygulanan yerel arama oranı, çaprazlama ile yeni üretilen çözümlerin ne kadarına 2-opt tabanlı yerel arama uygulanacağını ifade etmektedir. Yüksek değerlerde yavrular erken aşamada kendi yerel en iyilerine tırmanmakta ve popülasyonun sömürü kapasitesi artmaktadır. Düşük değerlerde ise yeterince sömürü yapılmayacak ve iyileştirme kısıtlı olacaktır. Büyük ölçekli problemlerde yerel arama maliyeti yükseldiği için bu oran dikkatle ayarlanmalıdır.

Mutant bireylere uygulanan yerel arama oranı, mutasyon sonucunda üretilen çözümlerin hangilerinin yerel arama ile iyileştirileceğini belirlemektedir. Mutant bireyler zaten keşfi artırmak için rastgele bozulduğundan, bu çözümlere yerel arama uygulamak hem keşif hem sömürü yönlerini birleştirmektedir. Ancak bu işlem maliyetlidir; özellikle büyük problemlerde mutantların tamamına yerel arama yapmak hesaplama süresini gereksiz yere uzatmaktadır. Bu nedenle bu oran genellikle yavrulara uygulanan yerel arama oranına kıyasla daha düşüktür.

En yakın komşu tabanlı başlangıç oranı, başlangıç popülasyonundaki bireylerin ne kadarının NN kullanılarak oluşturulacağını belirlemektedir. NN sezgiseli, GSP için yüksek kaliteli başlangıç turları üretme eğiliminde olduğu için başlangıç popülasyonunun kalitesini arttırmakta ancak aşırı kullanımında popülasyonu yapı olarak birbirine çok benzer hale getirmekte ve çeşitliliği düşürmektedir. Bu parametre başlangıç çeşitliliği ile kalite arasında kritik bir denge kurmaktadır. NN'nin başlangıçta uygulanması popülasyonun iyi ama aşırı benzer olmayan bir temel üzerinde başlamasını sağlamaktadır. Başlangıç kalitesi sonraki iyileştirme süreçlerinin verimliliğini doğrudan etkilemektedir.

Ada sayısı, popülasyonun kaç alt popülasyona bölüneceğini belirleyen parametredir. Ada sayısı arttıkça küresel keşif gücü artmakta; ancak her adadaki popülasyon küçüleceği için sömürü gücü zayıflayabilmektedir. Ada sayısının az olması ise bu algoritmada çeşitliliğin sağlanmasında ve hız açısından daha yavaş bir sistem olmasında etkili olacaktır. Bu nedenle ada sayısı problem ölçeğine göre hassas şekilde seçilmelidir.

Göç aralığı, adalar arasında birey alışverişinin kaç nesilde bir yapılacağını belirlemektedir. Çok sık göç yapılması adalar arasındaki çeşitliliği azaltmakta ve adaların birbirine benzemesine yol açmakta; çok seyrek göç ise kaliteli bilginin yayılmasını geciktirmektedir. Bu parametre, adalar arası bilgi akışının ritmini belirleyen keşif ve sömürü dengesine doğrudan etki eden bir bileşendir.

2-opt aday komşu listesi, yerel arama sırasında her düğüm için kaç komşunun değerlendirmeye alınacağını belirlemektedir. Tam 2-opt araması maliyetli olduğundan, aday komşuluk yapısı kullanılarak işlem hızlandırılmaktadır. Komşu (K) değerinin yükselmesi daha güçlü ama daha pahalı bir yerel arama sağlamakta; düşmesi ise maliyeti azaltmakta ancak iyileştirme kapasitesini kısıtlamaktadır. Büyük ölçekli problemlerde kritik bir hız ve kalite dengesi sağlamaktadır.

Çaprazlama oranı için ERX ve OX %50 olarak belirlenmesinde temel unsur GSP gibi karmaşık problemlerde yalnızca kenar koruyan veya yalnızca sıralama değiştiren bir çaprazlama operatörü kullanmak yeterli gelmemesinden kaynaklanmaktadır. Hibrit algoritmada bu iki operatörün olasılıksal olarak dönüşümlü kullanılması, hem yerel en iyilerden kaçma yeteneğini güçlendirir hem de yerel arama uygulandığında elde edilen ilerlemeyi yapısal olarak kalıcı hale getirir. Böylece keşif ve sömürü dengesi yalnızca ada modeliyle değil, çaprazlama mekanizmasının kendi iç dinamiğiyle de desteklenmiş olur.

5.4. Parametrelerin Etkileşimi

Önerilen PH-BWO modelinde kullanılan parametreler ve operatörler birbirinden bağımsız değildir; aksine arama sürecinin hem sömürü hem de keşif davranışını birlikte şekillendiren dinamik bir etkileşim ağı oluşturur. Popülasyon boyutu, üreme oranı, yamyamlık oranı ve mutasyon oranı gibi temel parametreler, çaprazlama ve yerel arama operatörleriyle doğrudan ilişkili çalışmaktadır. Bu nedenle her parametrenin etkisi, diğer bileşenlerle birlikte değerlendirildiğinde anlam kazanmaktadır.

Popülasyon boyutu, çeşitliliğin temel kaynağı olduğu için özellikle mutasyon ve çaprazlama operatörünün etkisini belirler. Daha büyük popülasyonlarda çaprazlamanın ürettiği çeşitlilik daha etkin yayılırken, küçük popülasyonlarda agresif yamyamlık mekanizması baskın hale gelmekte ve çeşitliliğin hızla tükenmesine yol açabilmektedir. Bu durum, ada yapısının önemini arttırmakta; adalar arasında yapılan göç, yetersiz kalan çeşitliliğin yeniden dengelenmesini sağlamaktadır.

Üreme oranı ile yamyamlık mekanizması birlikte çalışmaktadır. Üreme oranı yüksek olduğunda popülasyonda çok sayıda yeni birey üretilmekte, ancak aynı zamanda yamyamlık oranı düşükse bu yeni bireylerin önemli bir kısmı popülasyonda uzun süre kalabilmekte ve

sömürü gücünü artırırken keşif yeteneğini düşürebilmektedir. Yamyamlık oranı yükseltildiğinde ise zayıf bireyler hızla elenmekte, böylece popülasyon elit çözümlere doğru hızla yakınsamaktadır. Fakat bu agresif yapı, mutasyon ve çaprazlama operatörü ile desteklenmezse popülasyon tek bir yerel en iyi değerde kilitlenebilmektedir. Dolayısıyla mutasyon oranı bu noktada keşfi yeniden canlandıran kritik bir tamamlayıcı rol oynamaktadır.

Çaprazlama operatörlerinin etkisi de parametrelerle sıkı bir ilişki içindedir. ERX, turdaki iyi kenarları muhafaza ettiği için sömürü yönlü bir davranış sergiler ve yamyamlık mekanizmasıyla uyumludur; çünkü güçlü bireyleri daha da güçlendiren bir yapı oluşturmaktadır. Buna karşılık OX, şehir sıralarını köklü biçimde yeniden düzenlediği için mutasyonla benzer bir keşif etkisi yaratmaktadır. Bu nedenle OX'in etkisi, özellikle agresif sömürü yapan yamyamlık mekanizmasını dengeleyen bir çeşitlilik enjekte edici görev üstlenmektedir. ERX ve OX birlikte kullanıldığında biri kaliteyi(sömürü) , diğeri çeşitliliği(keşif) korumakta ve çift yönlü bir arama dengesi oluşturmaktadır.

Yerel arama (2-opt) ise üretilen bireylerin kalitesini doğrudan belirlediği için üreme, mutasyon ve çaprazlama operatörlerinin etkinliğiyle yakın ilişkilidir. Çaprazlama ya da mutasyon ile oldukça karışık bir tur üretildiğinde 2-opt bu bireyleri yerel en iyilere taşımakta ve yamyamlık mekanizmasının seçtiği bireylerin daha kaliteli olmasına katkı sağlamaktadır. Aşırı yerel arama uygulanması ise çeşitliliği zayıflatabileceğinden mutasyon ve çaprazlama ile dengelenmelidir.

Son olarak, ada modelinin göç aralıkları bu bütünlüğü destekleyen üst düzey bir kontrol mekanizmasıdır. Bir adada agresif yamyamlık ve 2-opt nedeniyle çeşitlilik azalsa bile, düzenli göç hem OX'in yarattığı keşfi hem de ERX'in koruduğu elit yapıyı diğer adalara taşımaktadır. Böylece her ada hem kendi içinde sömürü yapmakta hem de küresel düzeyde keşfe katkı sağlamaktadır.

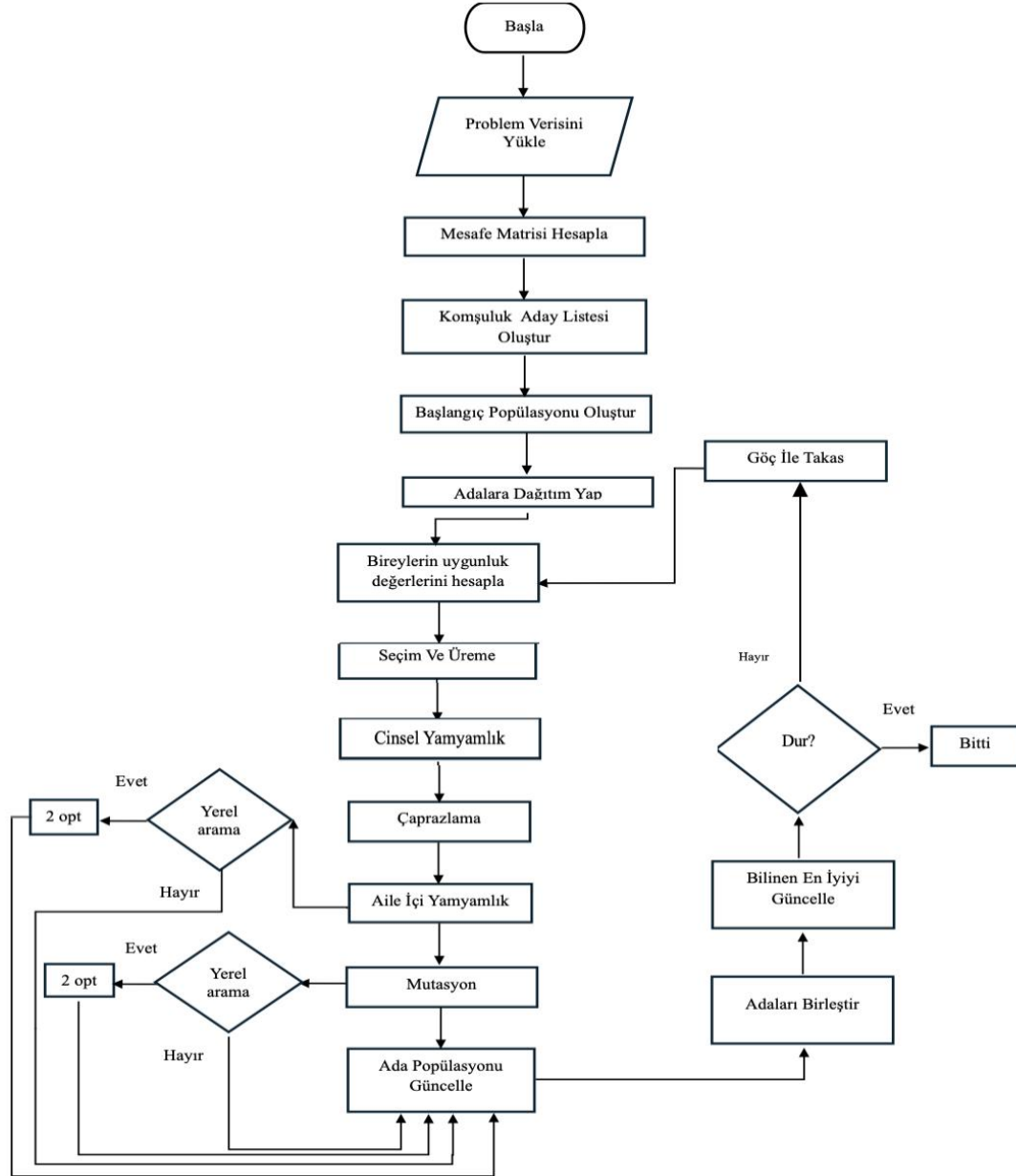
Bu çok katmanlı etkileşim, önerilen hibrit algoritmanın yalnızca tek bir parametreye bağlı olmadan, tüm yapı taşlarının birlikte çalışmasıyla dengeli bir arama performansı sergilemesini mümkün kılmaktadır. Bu etkileşim sisteminin bir diğer önemli boyutu, çözüm kalitesi ile hesaplama maliyeti arasındaki dengenin yönetilmesidir. Özellikle yerel arama ve mutasyon gibi işlem yükü yüksek operatörlerin, popülasyon büyüklüğü ve ada sayısı ile orantılı bir şekilde kurgulanması gerekmektedir.

NN Tohumlama Oranı, ada içi evrim sonunda popülasyonda eksik kalan bireylerin hangi oranının NN sezgiseliyle üretileceğini belirleyen parametredir. Üreme, yamyamlık ve mutasyon sonrası popülasyon küçüldüğünde, boşluklar bu oran kadar NN turlarıyla, kalan kısmı ise rastgele turlarla doldurulur. Böylece hem kalite korunur hem de çeşitlilik kaybolmaz. Algoritmada iki farklı NN parametresi olmasının nedeni, başlangıç popülasyonu kurulumu ile evrim sırasında yapılan yeniden tohumlama süreçlerinin birbirinden farklı gereksinimlere sahip olmasıdır. Bu ayrım, NN etkisini hem başlangıçta hem de süreç boyunca kontrollü şekilde ayarlayabilmeyi sağlar.

Parametrelerin oluşturduğu bu denge, problemin karmaşıklık düzeyi arttıkça algoritmanın dayanıklılığını belirleyen ana unsur haline gelmektedir. Küçük ölçekli problemlerde agresif sömürüye dayalı parametre kombinasyonları hızlı sonuç verirken, büyük ölçekli ve çok modlu uzaylarda etkileşimin yönü, çeşitliliği koruyan mekanizmalara kaydırılmaktadır. Bu bağlamda, her bir parametrenin tek başına optimize edilmesi yerine, diğer bileşenlerle olan ilişkisi üzerinden bütüncül bir yaklaşımla ele alınması, önerilen PH-BWO modelinin farklı problem yapılarında esnek, kararlı ve uyarlanabilir bir karakter sergilemesini sağlamaktadır.

5.5. Önerilen Modelin Çalışma Prensibi

Algoritmanın işleyişi, hazırlık, ada içi evrimsel süreç, küresel göç mekanizması ve sonlandırma aşamalarından oluşmaktadır. Şekil 5.1’de PH-BWO’nun en iyinin bulunduğu durumdaki şeması verilmiştir.



Şekil 5.1. PH-BWO'nun Akış Şeması

1. Hazırlık ve Başlatma: İlk olarak ilgili fonksiyon aracılığı ile TSPLIB dosyası okunmakta ve mesafe matrisi oluşturulmaktadır. Aynı zamanda problemin boyutuna ve tanımlanmış ön ayarlara göre tüm parametreler yüklenmektedir. Tüm mesafe matrisi hesaplandıktan sonra, yerel arama maliyetini düşürmek için her şehrin en yakın k adet komşusunu içeren bir aday komşuluk listesi oluşturulur.¹⁰ Hazırlık aşamasının son adımı ise başlangıç popülasyonunun oluşturulmasıdır. Bu popülasyon belirlenmiş oranda NN ve kalan kısmın rastgele turlar ile doldurulmasını sağlar. Popülasyon yine parametrelerce belirlenmiş ada sayısı kadar alt popülasyona bölünmekte ve paralel evrim sistemi için dağıtılmaktadır.

2. Ada İçi Evrimsel Döngü: Bu paralel yapıda her ada kendi yerel popülasyonunu, rastgele sayı üretim sürecini ve evrim döngüsünü tamamen bağımsız şekilde yürütmektedir. Adalar yalnızca göç aşamasında birbirleriyle bilgi paylaşarak etkileşime geçmektedir. Bu tasarım hem senkronizasyon maliyetlerini ortadan kaldırmakta hem de adaların çözüm uzayının farklı bölgelerinde bağımsız biçimde arama yapmalarını sağlayarak erken yakınsamaya karşı güçlü bir dayanıklılık oluşturmaktadır.

Her süreç bellek yoğun olan mesafe matrisi ve komşuluk listesini bir kez yükleyen kalıcı bir yapıya sahiptir. Evrimsel döngü tüm adalara çalıştırma komutunu vermekte ve sistem çalışmaya başlamaktadır. Her bir paralel ada, göç aralığı ile belirlenen sayı kadar nesil boyunca bağımsız evrim çalıştırmaktadır:

- **Sıralama ve Seçim:** Popülasyon uygunluk fonksiyonu değerine göre sıralanmaktadır. Üreme oranına göre en iyi bireyler üreme havuzu için seçilmektedir. Seçilen iki ebeveyn den daha iyi uygunluk fonksiyonu değerine sahip olan “anne”, diğeri ise “baba” olarak işaretlenir. Burada ilk popülasyon oluşturulmuştur (pop1).
- **Çaprazlama ve Cinsel Yamyamlık:** %50 olasılık ile ERX veya OX operatörleri kullanılarak çaprazlama işlemi ile 2 adet yavru birey üretilmektedir. “Baba” üreme

¹⁰ Büyük ölçekli TSPLIB örneklerinde tam 2-opt aramasının $O(n^2)$ karmaşıklığına sahip olması nedeniyle, bu tarama işlemi aday komşuluk listeleri ile sınırlandırılmış; böylece 2-opt yalnızca belirlenmiş en yakın K adet komşu üzerinde uygulanmış ve yerel arama maliyeti büyük ölçüde azaltılarak büyük ölçekli veri setlerinde işlem verimliliği önemli ölçüde artırılmıştır.

havuzundan atılmaktadır. Bu karadul örümceklerinin cinsel yamyamlık davranışının bir taklididir.

- **Aile İçi Yamyamlık:** BWO'nun yamyamlık mekanizması devreye girmekte ve “Anne” ve yavrular rekabet ederek yamyamlık oranı kadar güçsüz bireyler elenmektedir. İlk popülasyondan baba çıkarılır ve anne bu aşamada elendiyse o da popülasyondan atılarak yeni bir popülasyon oluşturulur ve hafızada tutulur (pop1_kept).
 - Aile içi seçim sonrasında hayatta kalan çocukların bir kısmına belirli bir olasılıkla 2-opt yerel araması uygulanmakta, yerel arama uygulanmayan ancak seçimden sağ çıkan çocuklar ise mevcut halleriyle ikinci popülasyona aktarılmaktadır (pop2). Buna karşılık, aile içi seçimde elenen çocuklar sonraki nesle taşınmamakta ve popülasyondan tamamen çıkarılmaktadır.
- **Mutasyon:** Mutasyon işlemi için kaynak olarak, üreme ve yamyamlık aşamalarından sağ çıkmayı başaran elit ebeveynler (pop1_kept) veya mevcut elit havuz (pop1) temel alınır. Bu havuzdan rastgele seçilen bireyler, mutasyon operatörüne tabi tutulur. Popülasyondan mutasyon oranı kadar seçilen bireye tersine çevirme mutasyonu uygulanmaktadır.
 - Parametrelerce belirlenen olasılıkla bireylere 2-opt yerel iyileştirmesi uygulanmaktadır. Bu noktada üçüncü popülasyon oluşturulur (pop3). Bu popülasyon mutasyona uğramış ve mutasyon sonrası yerel arama ile iyileştirilmiş bireylerden oluşmaktadır.
- **Birleştirme ve Güncelleme:** Hayatta kalan ebeveynler, yavrular ve mutant bireylerden oluşan bu üç popülasyon birleştirilmektedir. Kopya yavrular belirlenerek popülasyona yalnızca bir kez eklenmektedir. Bu noktada eğer popülasyon büyüklüğünden daha küçük bir popülasyon elde edilmiş ise NN ile veya rasgele olmak üzere üretilen yeni bireyler popülasyon büyüklüğüne ulaşınca kadar eklenir. Elitist seçim stratejisi uygulanarak, bireyler sıralanır ve popülasyon büyüklüğü kadar birey yeni popülasyon olarak güncellenir.

3. Küresel Göç ve Senkronizasyon: Ana süreç tüm adalardan gelen en iyi çözümleri toplamakta ve bilinen en iyiyi güncellemektedir. Her ada belirlenen göç aralığı kadar nesli tamamladığında, paralel süreçler duraklatılır ve adalar arası bilgi paylaşımı şu şekilde

gerçekleştirilir: Her ada sol komşusundan en iyi bireyi sağ komşusundan ise rastgele bir bireyi alır. Bu göç sistemi, iletişim maliyetini düşük tutarken elit bilginin kontrollü yayılmasını sağlamaktadır. Bu adadaki en kötü iki birey belirlenir ve popülasyondan atılır. Bu yenilenmiş popülasyonlar ile ana döngü bir sonraki paralel evrim aşamasına geri dönmektedir.

4. Sonlandırma ve Raporlama: Algoritma, bilinen en iyi değere ulaşana veya nesil sayısına kadar devam etmektedir. Böylece agresif sömürü ile küresel keşif dengelenmiş olur. Ana çözücü döngüsü sonlandığında en iyi uygunluk fonksiyonu değeri ve ilgili zaman istatistikleri döndürülür. Bu tek bir tekrarın sonucudur. İstatistiksel olarak anlamlılık ve sağlamlık elde etmek adına tüm bu süreç 10 defa bağımsız olarak tekrarlanmaktadır. Tüm çalışmalardan elde edilen sonuçlar toplanmakta ve her bir çalışma için medyan, ortalama, standart sapma, bulunan en kötü ve en iyi değerler ve %95 güven aralığı istatistikleri kaydedilmektedir.

6. DENEYSEL ÇALIŞMALAR

Tüm deneyler, Apple M4 çip (CPU) ile donatılmış bir MacBook Air bilgisayar üzerinde yürütülmüştür. Sistem, 10 çekirdekli M4 çipiyle bütünleşik çalışan 16 GB birleşik belleğe (Unified Memory/RAM) sahiptir. Algoritmanın paralel hesaplama yetenekleri, M4 çipinin çok çekirdekli mimarisinden faydalanmıştır. Algoritma, Python programlama dili (Sürüm 3.13.5) kullanılarak geliştirilmiştir. Gerekli olan bilimsel hesaplama kütüphaneleri (NumPy, Pandas, multiprocessing vb.) ve bu kütüphanelerin bağımlılıkları, Anaconda (Sürüm 25.7.0) paket ve ortam yönetim sistemi aracılığıyla yönetilmiştir. Kodun geliştirilmesi, hata ayıklanması ve çalıştırılması işlemleri, Anaconda dağıtımı içerisinde yer alan Spyder Bütünleşik Geliştirme Ortamı (IDE) (Sürüm 5.15.10) üzerinden yapılmıştır. Tüm bu yazılım çatısı, macOS (Sürüm 26.0.1, Tahoe) işletim sistemi üzerinde çalıştırılmıştır.

BLAS (Basic Linear Algebra Subprograms), temel doğrusal cebir işlemlerini yüksek hızda gerçekleştirmek için geliştirilmiş düşük seviyeli bir kütüphane standardıdır. NumPy ve benzeri bilimsel hesaplama kütüphaneleri, bu işlemleri daha verimli yapmak için BLAS rutinlerini kullanmaktadır. Zaman ölçümlerinde olası donanımsal varyasyonları önlemek amacıyla, BLAS tabanlı hesaplama kütüphanelerinin iş parçacığı sayıları bir ile sınırlandırılmıştır. Bu kütüphane çok çekirdekli çalışarak işlemleri hızlandırabilmektedir. Ancak, aynı işlemi her çalıştırmada farklı sayıda işlemci çekirdeği kullanılırsa çalışma sürelerinde dalgalanmalar olmaktadır. Bu nedenle deneysel çalışmalarda adil zaman karşılaştırması yapılabilmesi için, bu kütüphanelerin paralel iş parçacığı sayısı bire sabitlenmiştir. Yani, her çalışmada aynı işlemci çekirdeği kullanılarak deterministik süre ölçümü elde edilir. Deneylerin tekrarlanabilirliği için Python içerisinde bulunan `random.seed` ve `np.random.seed` çağrıları kullanılmıştır.

Gezgin Satıcı Problemi üzerine yapılan deneysel çalışmaların geçerliliği, kullanılan veri setlerinin niteliği ve sonuçların literatürle karşılaştırılabilirliği ile doğrudan ilişkilidir. Bu bağlamda Reinelt tarafından 1991 yılında oluşturulan TSPLIB, farklı şehir dağılımlarına, boyutlara ve mesafe türlerine sahip geniş bir problem çeşitliliği sunarak araştırmacılara standardize edilmiş bir test ortamı sağlamaktadır. TSPLIB örneklerinin en iyi değerleri bilinmektedir ve bu durum önerilen yeni algoritmaların performansının literatürdeki yerleşik çalışmalarla aynı zeminde değerlendirilebilmesine imkan tanımaktadır (Reinelt, 1991).

Kütüphanenin açık olması, sonuçların tekrarlanabilirliğini artırmakta ve bilimsel güvenilirliği güçlendirmektedir. TSPLIB veri setlerinin seçilmesi yalnızca literatürle uyumluluk sağlamakla kalmamış; aynı zamanda çözüm uzayının farklı ölçekteki problemlerde test edilmesine olanak tanıyarak, PH-BWO algoritmasının keşif ve sömürü dengesini düğüm sayısı değişiminde ne ölçüde sürdürebildiğini ortaya koymuştur.

6.1. Parametre Ayarlarının Belirlenmesi

Parametre belirleme süreci, yalnızca teorik bilgilerle değil, aynı zamanda sistematik biçimde yürütülen kapsamlı bir pilot çalışma ile şekillendirilmiştir. İlk aşamada, TSPLIB kütüphanesinden rastgele seçilen 32 problem üzerinde pilot testler yapılmış ve bu probleme özgü davranışları gözlemlemek amacıyla her biri 30 bağımsız tekrar ile çalıştırılmıştır. Bu testlerde en iyi değer, en kötü değer, ortalama tur uzunluğu, en iyiden sapma yüzdesi, çözüm süresi ve en iyiye ulaşma sayısı gibi metrikler raporlanmıştır. Amaç, parametrelerin problem boyutu ile ilişkili dinamiklerini incelemek ve hangi parametre aralıklarının farklı ölçeklerde daha kararlı davrandığını belirlemektir.

Hayyolam ve Kazem (2020) tarafından önerilen orijinal BWO parametreleri bu çalışmada doğrudan kullanılmamıştır. Bunun temel nedeni, orijinal BWO parametrelerinin sürekli fonksiyon problemlerine göre ayarlanmış olmasıdır. GSP ise kesikli yapıda bir problem olduğundan, mutasyon, yamyamlık ve üreme gibi operatörler sürekli uzaya kıyasla çok daha etkili ve keskin yapısal değişiklikler üretmektedir. Bu durum, orijinal parametrelerin GSP’de aşırı agresif davranarak erken yakınsamaya yol açmasına neden olmuştur. Ayrıca bu çalışma orijinal BWO’dan farklı olarak GA çaprazlamaları, ada modeli, NN-tabanlı başlangıç, 2-opt yerel arama ve mutant iyileştirme gibi ek bileşenlerle zenginleştirilmiş hibrit bir mimari sunduğundan, parametrelerin hibrit yapı içinde yeniden dengelenmesi gerekli hale gelmiştir. Pilot çalışma esnasında kullanılan parametreler Tablo 6.1’de sunulmuştur ve her ölçek sınıfı için en kararlı sonuçları üreten kombinasyonları temsil etmektedir.¹¹

¹¹ Belirlenen parametre değerleri literatürdeki GSP odaklı çalışmalar taranarak ve pilot çalışma esnasında yapılan deneylerin analiz edilmesi ile belirlenmiştir.

Tablo 6.1 Pilot Çalışma Esnasında Kullanılan Parametreler

Parametre / Set	Küçük Ölçek	Orta Ölçek	Büyük Ölçek
Popülasyon boyutu	512	256	256
Maksimum nesil sayısı	1000	1500	2000
Üreme oranı	0,70	0,60	0,55
Yamyamlık oranı	0,40	0,40	0,35
Mutasyon oranı	0,40	0,30	0,30
Çaprazlama yerel arama oranı	0,80	0,30	0,12
Mutasyon yerel arama oranı	0,40	0,15	0,05
En yakın komşu tabanlı başlangıç oranı	0,25	0,28	0,20
Ada sayısı	4	6	6
Göç aralığı	10	10	18
K-en yakın komşu sayısı	0	32	48

Elde edilen sonuçlar, üreme, yamyamlık ve mutasyon oranlarının orijinal çalışmadakinden¹² daha dengeli değerlere ayarlanmasının her ölçekteki problemler için daha istikrarlı performans sağladığını göstermiştir.

Bu aşamanın ardından, kütüphaneden temsil gücü yüksek olan 16 TSPLIB problemi seçilmiş ve yalnızca düğüm sayılarına göre küçük ($0 < n < 101$), orta ($100 < n < 500$) ve büyük ($499 < n$) ölçekli olmak üzere üç sınıfa ayrılmıştır. Her sınıf için ayrıca üç çalışma modu tasarlanmıştır. Hız odaklı ön ayar seti, düşük popülasyon boyutu ve sınırlı yerel arama ile kısa sürede kabul edilebilir çözümler üretmeyi hedeflerken; denge odaklı ön ayar seti süre ve kalite arasında makul bir denge sağlamaktadır. Kalite odaklı ön ayar seti ise yüksek popülasyon, güçlü yerel arama ve daha yoğun evrimsel baskı ile bilinen en iyiye yakın sonuçlar üretmek için tasarlanmıştır. Böylece PH-BWO, farklı problem boyutları ve performans beklentileri için esnek bir çalışma mimarisi sunacak şekilde yapılandırılmıştır.

¹² Hayyolam ve Kazem'in çalışmasında; üreme oranı = 0,60, yamyamlık oranı= 0,44, mutasyon oranı= 0,40 olarak belirlenmiştir.

Sonuç olarak ön ayar setlerinden her biri, kullanıcı gereksinimlerine ve problem ölçeğine uygun esnek çalışma modlarını test edecek şekilde, pilot deneylerin ve gözlemlenen algoritma davranışlarının sistematik bir analizi sonucunda belirlenmiştir.

6.1.1. Küçük Ölçekli Veri Seti Pilot Çalışma Sonuçları

Küçük ölçekli TSPLIB problemleri üzerinde yürütülen pilot çalışma, bu ölçek için belirlenen parametre setinin hem sömürü kapasitesi hem de yakınsama hızı açısından oldukça başarılı bir yapı sunduğunu göstermektedir. Bu sınıfta kullanılan parametreler, özellikle yüksek seçim baskısı ve yoğun yerel arama uygulamalarıyla agresif bir evrimsel süreç oluşturmak üzere tasarlanmıştır. Küçük ölçekli veri setlerinde düğüm sayısı düşük olduğundan, BWO'nun agresif sömürü mekanizması davranışının net biçimde gözlemlenebilmesi için ideal bir test ortamı sunmuştur. Tablo 6.2'de ilgili sonuçlar verilmiştir.

Tablo 6.2 incelendiğinde, tüm problemlerde algoritmanın tüm tekrarlarda bilinen en iyi değeri bulabildiği görülmektedir. En iyi, en kötü ve ortalama değerlerin tümünün bilinen en iyi değer ile tamamen eşleşmesi, küçük ölçekli problemlerde hiçbir tekrarda istatistiksel sapma oluşmadığını göstermektedir. Bu durum, kullanılan parametrelerin sömürü gücünün bu sınıf için fazlasıyla yeterli olduğunu ve özellikle yerel arama mekanizmasının çözüm kalitesini anında iyileştirerek en iyi değere hızla ulaşılmasını sağladığını ortaya koymaktadır.

Tablo 6.2 Küçük Ölçekli Veri Seti Pilot Çalışma Sonuçları

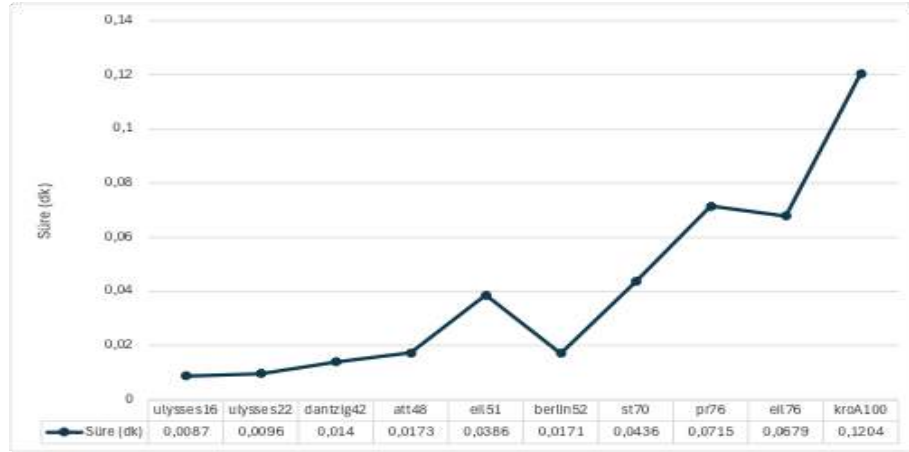
Veri Seti	Bilinen En İyi	Bulunan En İyi	Bulunan En Kötü	Bulunan Ortalama	Bulunan Ortalama İterasyon	En İyiden Ortalama Sapma(%)	Bulunan Süre (dk)	Copt ¹³
ulysses16	6859	6859	6859	6859	10	0	0,0087	30/30
ulysses22	7013	7013	7013	7013	10	0	0,0096	30/30
dantzig42	699	699	699	699	16	0	0,0140	30/30

¹³ Copt : Tüm çalışmalarda ulaşılan bilinen en iyi ölçeği.

Tablo 6.2. (Devam) *Küçük Ölçekli Veri Seti Pilot Çalışma Sonuçları*

att48	10628	10628	10628	10628	12	0	0,0173	30/30
eil51	426	426	426	426	25	0	0,0386	30/30
berlin52	7542	7542	7542	7542	10	0	0,0171	30/30
st70	675	675	675	675	17	0	0,0436	30/30
pr76	108159	108159	108159	108159	27	0	0,0715	30/30
eil76	538	538	538	538	24	0	0,0679	30/30
kroA100	21282	21282	21282	21282	30	0	0,1204	30/30

Elde edilen sonuçlar hesaplama verimliliği açısından değerlendirildiğinde ise, algoritmanın en iyi sonuca ulaşma hızının oldukça yüksek olduğu görülmektedir. Tabloda yer alan ortalama nesil sayılarının çok düşük seviyelerde seyretmesi, önerilen hibrit yapının başlangıç popülasyonundan itibaren doğru arama bölgesine hızla odaklandığını ve gereksiz döngülerden kaçınarak sonuca gittiğini kanıtlamaktadır. Şekil 6.1’de ilgili problemlere ait süre farkları grafiği verilmiştir.



Şekil 6.1. *Küçük Ölçekli Problemlerde Pilot Çalışma Süre Farkı Sonuçları*

Şekil 6.1 incelendiğinde problemler arası farkın yalnızca çözüm sürelerinde meydana geldiği görülmektedir. Problemde düğüm sayısının artışı ile çözüm süresinin de arttığı görülmüştür.

Ortalama nesil değerlerinin oldukça düşük seviyelerde yoğunlaşması da algoritmanın küçük ölçekli problemlerde bilinen en iyi değere çok hızlı ulaştığını göstermektedir. Ayrıca çalışma sürelerinin tüm problemler için saniyenin çok altında kalması, bu sınıf için kullanılan popülasyon boyutunun ve nesil limitlerinin hesaplama maliyeti açısından oldukça uygun olduğuna işaret etmektedir. Ada içi yoğun 2-opt uygulaması, güçlü yamyamlık mekanizması ve üreme sırasında ERX/OX operatörlerinin oluşturduğu yüksek sömürü ve keşif baskısı birlikte çalışarak çözüm kalitesinin hızlı biçimde yükselmesini sağlamıştır. Her adanın yalnızca 128 birey içermesi, agresif yamyamlık baskısıyla elitlerin dağılımını hızlandırırken, dört adanın paralel çalışması sayesinde çeşitliliğin tamamen tükenmesi engellenmiş; böylece hem hız hem kalite eşzamanlı olarak korunmuştur.

Genel olarak değerlendirildiğinde küçük ölçek için oluşturulan parametre seti, bu ölçekteki TSPLIB problemlerinin tamamında tam isabet ile en iyiye ulaşmış, hiçbir problemde erken yakınsama, çeşitlilik kaybı ya da performans dalgalanması gözlenmemiştir. Çocuk bireylere uygulanan yüksek yerel arama oranı ve orta düzey mutasyon oranı küçük ölçekli problemlerde hızlı iyileştirme ve yeterli çeşitlilik üretimi sağlayarak dengeli bir arama davranışı oluşturmuştur. Popülasyonun dört ada üzerinde bölünmesi, bir yandan çeşitliliğin korunmasını sağlarken, diğer yandan paralel yapı sayesinde hızlı bir evrimsel ilerlemeye imkan tanımıştır. Bu nedenle küçük ölçekli sınıf için daha agresif veya daha maliyetli ek parametre arayışına gerek duyulmamış; mevcut ayarlar, küçük ölçekli GSP problemlerinde yüksek kaliteyi garanti eden ve tüm çalışmalarda istikrarlı biçimde aynı performansı sergileyen kararlı bir çözüm çerçevesi olarak değerlendirilmektedir.

6.1.2. Orta Ölçekli Veri Seti Pilot Çalışma Sonuçları

Orta ölçekli TSPLIB problemleri üzerinde yürütülen pilot çalışma, bu parametre setinin genel olarak dengeli bir arama davranışı sergilediğini göstermektedir. Bu sınıf, küçük ölçekli problemlere kıyasla daha geniş ve daha karmaşık bir çözüm uzayı sunduğundan, kullanılan parametrelerin hem çeşitlilik üretimi hem de sömürü yoğunluğu açısından dikkatli bir şekilde ölçeklendirilmesi gerekmiştir. Bu nedenle popülasyon büyüklüğü düşürülmüş, ada sayısı artırılmış ve yerel arama oranları sınırlı tutulmuştur. Böylece hem hesaplama maliyeti kontrol

altında tutulmuş hem de paralel yapı sayesinde çözüm uzayının farklı bölgelerinin aynı anda keşfedilmesi sağlanmıştır. Pilot çalışmadan elde edilen sonuçlar Tablo 6.3'te sunulmuştur.

Tablo 6.3 Orta Ölçekli Veri Seti Pilot Çalışma Sonuçları

Veri Seti	Bilinen En İyi	Bulunan En İyi	Bulunan En Kötü	Bulunan Ort.	Bulunan Ortalama İterasyon	En İyiden Ortalama Sapma(%)	Bulunan Süre (dk)	Copt ¹⁴
ch130	6110	6110	6128	6117	128	0,29%	0,75	18/30
gr137	69853	69853	69853	69853	77	0,00%	0,10	30/30
brg180	1950	1950	1950	1950	16	0,00%	0,05	30/30
kroA200	29368	29368	29382	29371	334	0,00%	0,85	25/30
tsp225	3916	3916	3919	3917	76	0,08%	0,11	30/30
gr229	134602	134602	135849	135076	678	0,35%	2,92	1/30
pr264	49139	49135	49135	49135	129	0,00%	0,26	30/30
a280	2579	2579	2579	2579	224	0,00%	0,48	30/30
lin318	42029	42029	42564	42299	294	0,64%	3,45	3/30
rd400	15281	15281	15589	15488	1305	0,65%	5,04	0/30
fl417	11861	11861	12131	11897	1099	0,29%	4,04	1/30

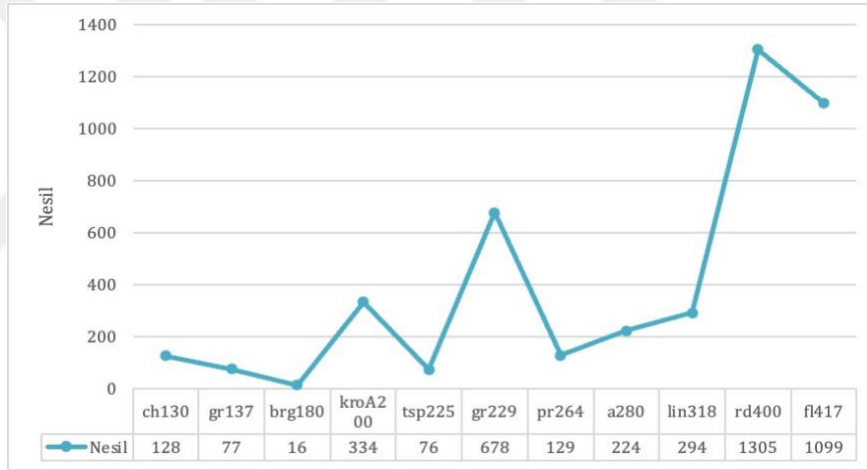
Tablo 6.3 değerlendirildiğinde, bazı orta ölçekli problemlerde algoritma, 30 bağımsız tekrarın tamamında bilinen en iyi değeri yeniden üretmeyi başarmıştır. Özellikle gr137, brg180, tsp225, a280 ve pr264 problemlerinde bulunan en iyi, en kötü ve ortalama değerlerin bilinen en iyi ile birebir örtüşmesi, bu parametre setinin bu düğüm aralığında güçlü bir sömürü kapasitesi sunduğunu göstermektedir.

Bununla birlikte orta ölçeğin sınırına yaklaşıldıkça, çözüm uzayının genişlemesiyle birlikte çözüm kalitesinde hafif dalgalanmalar görülmüştür. Özellikle gr229, lin318, rd400 ve fl417 gibi 229–417 düğüm aralığındaki problemlerde bilinen en iyiye ulaşma sayısının düştüğü gözlenmiştir. Ayrıca ch130 için ortalama sapma %0,29 olup 30 tekrardan yalnızca 18'inde en iyi elde edilmiştir. Ancak tüm bu problemlerde dahi ortalama sapmanın %1'in altında kalması, algoritmanın bilinen en iyiye çok yakın bir arama davranışı sergilediğini ve

¹⁴ Copt : Tüm çalışmalarda ulaşılan bilinen en iyi ölçeği.

yalnızca sömürü derinliğinin bazı tekrarlar da yetersiz kaldığını göstermektedir. Bu bulgu, özellikle orta ölçeğin geniş alt aralıklarında çocuklarda yerel arama oranının bir miktar artırılmasının kalite üzerinde olumlu etkiler yaratabileceğini işaret etmektedir.

Ortalama nesil sayılarının 300–1300 aralığında değişmesi, algoritmanın bu problemler üzerinde en iyiye yakın bir noktada daha uzun süre arama yaptığını göstermektedir. kroA200, gr229, rd400 ve fl417 problemlerinde nesil sayılarının belirgin şekilde artması, orta parametre setinin keşif ve sömürü dengesini genel olarak korusa da bazı problemler için yerel arama yoğunluğunun daha yüksek olması gerektiğini düşündürmektedir. Şekil 6.2’de problemin çözümünde en iyinin bulunduğu nesil farkları görülmektedir.



Şekil 6.2. Çözümün Bulunduğu İterasyon Sayısının Problem Bazında Gösterimi

Şekil 6.2 incelendiğinde çözümün bulunduğu neslin genel olarak problemin düğüm sayısı büyüdükçe hızla arttığı gözlemlenmektedir. Hesaplama süreleri bakımından ise parametre seti oldukça verimli bir yapı sunmaktadır. Süre değerlerinin 0,10–5,04 dakika aralığında kalması, ada sayısının artırılmasının paralel hesaplama kapasitesini etkin bir şekilde artırdığını doğrulamaktadır. Özellikle 180–280 düğümlü örneklerde süre ve performans dengesi oldukça tatmin edicidir. Ayrıca k-En Yakın Komşu parametresinin 2-opt maliyetini önemli ölçüde düşürerek büyük orta ölçekli problemler için arama hızını artırdığı gözlemlenmiştir.

Genel olarak değerlendirildiğinde orta parametre seti, orta ölçekli TSPLIB problemleri için hem yüksek çözüm kalitesi hem de makul hesaplama süreleri sunan dengeli bir mimari

ortaya koymaktadır. En iyiye ulaşamayan örneklerde bile sapmaların oldukça düşük kalması, parametre setinin doğru şekilde yapılandırıldığını ve arama davranışının istikrarlı olduğunu göstermektedir. Bununla birlikte özellikle 250 düğüm üzerindeki örneklerde görülen performans dalgalanmaları, bu problem sınıfı için daha yoğun yerel arama ve artırılmış sömürü kapasitesi sunan kalite odaklı bir ön ayar seti mimarisinin gerekli olduğuna işaret etmektedir. Bu nedenle parametre seti orta ölçek için güçlü bir temel niteliğinde olup, ön ayar setleri ile bu sınıfta daha yüksek arama derinliği sağlanabileceği değerlendirilmektedir.

6.1.3. Büyük Ölçekli Veri Seti Pilot Çalışma Sonuçları

Büyük ölçekli TSPLIB problemleri üzerinde yapılan pilot çalışma, bu parametre setinde popülasyon büyüklüğünün korunması, ada sayısının sınırlandırılması, yerel aramanın düşük oranlarda uygulanması ve k-En Yakın Komşu değerinin artırılması; özellikle geniş arama uzaylarında süreyi yönetilebilir tutmak, yerel yoğunlaşmanın aşırı artmasını engellemek ve keşif kapasitesini güçlendirmek amacıyla tercih edilmiştir. Elde edilen sonuçlar, yüksek ölçekli GSP örneklerinde keşif kapasitesinin süre kadar kritik bir rol oynadığını ortaya koymaktadır. Sonuçlar Tablo 6.4'te verilmiştir.

Tablo 6.4 *Büyük Ölçekli Veri Seti Pilot Çalışma Sonuçları*

Veri Seti	Bilinen En İyi	Bulunan En İyi	Bulunan En Kötü	Bulunan Ort.	Ort. Nesil	Ort. Sapma(%)	Ort. Süre (dk)	Copt
att532	27686	27960	29273	28474	1915	0,99%	8,47	0/30
pa561	2763	2807	2866	2840	1837	1,59%	6,36	0/30
d657	48912	49808	52169	50606	1917	1,83%	9,87	0/30
gr666	294358	300953	311039	306395	1958	2,24%	8,44	0/30
u724	41910	42888	43637	43246	1939	2,33%	10,80	0/30
rat783	8806	9075	9225	9156	1945	3,05%	10,87	0/30
dsj1000	18660188	19361503	19517519	19571835	1956	3,76%	18,01	0/30
pr1002	259045	267260	274198	269738	1907	3,17%	15,97	0/30
si1032	92650	92655	92938	92818	1335	0,01%	17,32	0/30
u1060	224094	233813	241175	236086	1951	4,34%	18,79	0/30

Tablo 6.4 incelendiğinde, büyük ölçekli problemlerden hiçbirinin 30 bağımsız tekrarda bilinen en iyi değere ulaşamadığı görülmektedir. Ancak bu durum, çözüm kalitesinin zayıf olduğu anlamına gelmemektedir; aksine 500–700 düğümlü örneklerde ortalama sapma değerlerinin %0,99 ile %2,24 aralığında kalması, algoritmanın geniş çözüm uzayına rağmen en iyiye oldukça yakın sonuçlar üretebildiğini göstermektedir. Bu performans yerel arama yoğunluğunun düşük tutulmasına rağmen agresif yamyamlık ve çaprazlama baskısının temel sömürü kapasitesini koruduğunu göstermektedir. Şekil 6.3'te problemlerin bilinen en iyiden ortalama sapma yüzdeleri verilmiştir.



Şekil 6.3. Büyük Ölçekli Problemlerde Bilinen En İyiden Ortalama Sapma Yüzdeleri

Şekil 6.3 incelendiğinde; düğüm sayısının 724 ve üzerine çıktığı problemler (u724, rat783, pr1002) için sapma yüzdesinin %2,33–%3,17 aralığına yükseldiği görülmektedir. Bu problemlerdeki sapma aralığı yüksekliği ve özellikle u1060 probleminde sapmanın %4,34'e kadar çıkması, 1000+ düğümlü örneklerde mevcut yerel arama yoğunluğunun sömürü derinliğini sınırladığını açıkça ortaya koymaktadır. Problemin düğüm sayısı arttıkça sapma yüzdesi de aynı oranda artmaktadır.¹⁵

¹⁵ Burada si1032 probleminin sapma yüzdesinin neredeyse sıfır olması problemin özel mesafe yapısından (MATRIX) kaynaklanmaktadır. Dolayısı ile genel sonucu yansıtmamakla birlikte topolojik farklılıkların da problemler üzerinde belirleyici rol oynayabildiği gözlemlenmiştir. Problem ilgili sonucun görülebilmesi adına özellikle seçilmiştir.

Bununla birlikte, büyük ölçekli problemlerin tümünde bulunan en iyi ve en kötü sonuçlar arasındaki farkın düşük kalması, algoritmanın geniş ölçeklerde dahi istikrarlı bir arama davranışı sergilediğini göstermektedir. Bu gözlem, arama sürecinin aşırı rastgele hale gelmediğini ve parametre setinin büyük ölçekte kontrollü bir keşif ve sömürü dengesi sunduğunu işaret etmektedir.

Ortalama nesil sayılarının tüm büyük ölçekli örneklerde 1.800–2.000 aralığında toplanması, algoritmanın çoğu tekrarda sınıra ulaştığını göstermektedir. Bu veri, büyük parametre setinin mevcut haliyle sömürü kapasitesinin sınırlı kaldığını ve problemin çözüm yapısının daha derin incelenmesi için ek nesile ihtiyaç duyulduğunu ortaya koymaktadır. Bu eğilim süre değerleriyle de uyumludur: 8–20 dakika arasında değişen çalışma süreleri, ada sayısının 6 ile sınırlandırılmasının paralel hesaplama verimliliğini belirli bir denge içerisinde tuttuğunu göstermektedir.

Genel olarak değerlendirildiğinde büyük ölçekler için oluşturulmuş parametre seti, bu problemler için hızlı en iyi değer arayışından ziyade, çözüm kalitesi ile hesaplanabilir süre arasında dengeli bir yapı sunmayı hedeflemektedir. En iyi çözüme ulaşamamasına rağmen sapmaların büyük çoğunlukla %1–3 seviyelerinde kalması, parametre setinin bu ölçek sınıfı için güçlü ve istikrarlı bir temel oluşturduğunu göstermektedir. Bununla birlikte 700+ düğüm üzeri problemlerde gözlenen kalite düşüşü, yerel arama yoğunluğunun ve K-En yakın komşu değerinin bu ölçekte sınırlayıcı olduğunu ve daha yüksek sömürü kapasitesi gerektirdiğini işaret etmektedir. Bulgular mevcut algoritmanın özellikle 700+ düğümlü problemlerde daha güçlü yerel arama, hiyerarşik tarama veya çeşitlilik artırıcı mekanizmalara ihtiyaç duyduğunu açık biçimde göstermektedir. Bu bulgular, büyük parametre set sınıfı için ilerleyen aşamalarda daha yoğun yerel arama içeren alternatif ön ayar seti tasarımlarının geliştirilmesinin gerekliliğine güçlü bir şekilde işaret etmektedir.

6.1.4. Ön Ayar Parametre Setlerinin Belirlenmesi

Küçük ölçekli TSPLIB problemleri üzerinde yürütülen pilot çalışma, bu problem sınıfının doğası gereği algoritmanın hem çözüm kalitesi hem de hız açısından son derece avantajlı bir performans sergilediğini göstermiştir. Tüm küçük veri setlerinde yöntem, 30 bağımsız tekrarın tamamında bilinen en iyi değeri elde etmiş; bulunan en iyi, ortalama ve en

kötü değerlerin tamamı bilinen en iyi ile birebir örtüşmüş; sapma sıfıra inmiş ve çalışma süreleri saniyeler düzeyinde kalmıştır. Küçük problemlerin görece düzenli çözüm uzayı ve düşük düğüm sayısı, agresif sömürü mekanizmalarının erken yakınsamaya yol açmadan hızlı ve kararlı sonuç üretmesini mümkün kılmıştır. Bu bulgular doğrultusunda farklı kullanım gereksinimlerine hitap edecek üç ayrı parametre yapısının tasarlanması uygun görülmüş ve hız, denge ve kalite odaklı ön ayar setleri oluşturulmuştur.

Üç ön ayar setin tasarımında pilot çalışmadan elde edilen bulgular doğrudan rehberlik etmiştir. Denge odaklı ön ayar seti oluşturulurken, en kararlı davranış sergileyen parametre düzeni temel alınmıştır. Popülasyon boyutunun görece geniş tutulması çözüm kalitesinin güçlü biçimde korunmasını sağladığından bu yapı korunmuş, nesil sayısı ise küçük problemlerde bilinen en iyinin çok erken bulunmasına rağmen güvenli bir arama payı bırakmak amacıyla yüksek tutulmuştur. Yerel arama düzeyinin pilot çalışmada çözüm kalitesine önemli katkı sağladığı gözlemlendiğinden dengeli ön ayar setinde güçlü fakat aşırı olmayan bir yerel arama yoğunluğu tercih edilmiştir. Üreme ve yamyamlık baskısının küçük ölçekli problemlerde erken yakınsamaya yol açmadığını gösteren bulgular doğrultusunda bu baskı seviyesi korunmuş; ada sayısının artırılmasının ek fayda sağlamadığı saptandığından ada sayısı sabit tutulmuştur. Böylece dengeli ön ayar seti, küçük boyutta hem hızlı hem kararlı sonuçlar sunan genel amaçlı bir yapı olarak belirlenmiştir.

Hız odaklı ön ayar seti ise pilot çalışmanın ortaya koyduğu kritik bir gözlem üzerinden şekillendirilmiştir: Küçük ölçekli problemler, popülasyon boyutu düşürüldüğünde, nesil sınırı azaltıldığında ve yerel arama yoğunluğu azaltılsa dahi en iyiye kolaylıkla ulaşabilmektedir. Bu nedenle hız odaklı ön ayar setinde popülasyon küçültülmüş, nesil sayısı azaltılmış ve yerel arama seviyeleri sınırlı tutulmuştur. Yerel aramanın azaltılması çalışma süresini çok belirgin biçimde düşürdüğünden, bu veri seti hızlı eniyileme gerektiren senaryolar için ideal hale gelmiştir. Çeşitliliğin aşırı azalmasını engellemek için mutasyon oranı yükseltilmiş, bilgi paylaşımını artırmak amacıyla göç aralıkları daha sık hale getirilmiştir. Bu yapı özellikle gerçek zamanlı uygulamalarda veya çoklu deneylerin kısa sürede tamamlanması gereken durumlarda en iyiye yakın çözümleri hızla üretmek için tasarlanmıştır.

Kalite odaklı ön ayar seti ise küçük problemler için en yüksek çözüm kalitesini hedefleyen durumlara yönelik olarak geliştirilmiştir. Pilot çalışma, küçük ölçekli

problemlerde yerel aramanın maliyetinin düşük, kazanımının ise yüksek olduğunu göstermiştir. Bu nedenle kalite odaklı ön ayar setinde popülasyon genişletilmiş, nesil sayısı artırılmış ve yerel arama yoğunluğu maksimum seviyeye çıkarılmıştır. Başlangıç çeşitliliğini artırmak için NN tabanlı başlangıç oranı yükseltilmiş, seçim baskısı güçlendirilerek yüksek kaliteli bireylerin korunmasına öncelik verilmiştir. Bu ön ayar seti maksimum kalite gerektiren araştırma senaryoları veya bilinen en iyiye mümkün olduğunca yakın çözüm elde etmek isteyen kullanıcılar için tasarlanmıştır.

Sonuç olarak küçük ölçekli ön ayar setlerinin oluşturulmasında pilot çalışma; agresif sömürü stratejilerinin güvenle uygulanabileceğini, yerel aramanın çözüm kalitesinde belirleyici etki yarattığını ve popülasyon ve nesil dengesinin hız ve kalite ayarına göre rahatlıkla ayarlanabileceğini göstermiştir. Bu üç ölçek seti, izleyen bölümlerde sunulan deneysel analizlerde küçük ölçekli TSPLIB problemleri için varsayılan parametre aileleri olarak kullanılmış ve tutarlı sonuçlar vermiştir. Pilot çalışma kapsamında elde edilen parametre hassasiyeti bulguları, orta ve büyük ölçekli ön ayar setlerinin tasarımında da başlangıç referansı işlevi görmüş; böylece ayarlama süreci rastlantısal denemelerden ziyade sistematik ve yeniden üretilebilir bir çerçeveye oturtulmuştur. Tablo 6.5'te küçük ölçek için belirlenen parametre değerleri sunulmaktadır.

Tablo 6.5 *Küçük Ölçekli Problemler için Ön Ayarlı Parametre Seti*

Parametre / Set	Hız Odaklı	Denge odaklı	Kalite Odaklı
Popülasyon boyutu	384	512	640
Maksimum nesil	700	1000	1500
Üreme oranı	0,60	0,66	0,75
Yamyamlık oranı	0,35	0,40	0,50
Mutasyon oranı	0,45	0,35	0,30
Yavru bireylere uygulanan yerel arama oranı	0,40	0,70	0,90
Mutant bireylere uygulanan yerel arama oranı	0,20	0,35	0,50
En yakın komşu tohumlama oranı	0,20	0,25	0,30
Ada sayısı	4	4	4
Göç aralığı	8	10	8
2-opt Aday Komşu Sayısı	Yok	Yok	Yok
En yakın komşu tabanlı başlangıç oranı	–	0,35	0,40

Orta ölçekli TSPLIB problemleri için parametrik yapılandırma, küçük ölçekli problemlere göre daha karmaşık bir denge gerektirmiştir. Pilot çalışma, orta ölçekli problemlerde çözüm uzayının genişlemesi nedeniyle çeşitliliğin kritik öneme sahip

olduğunu, buna karşın yerel aramanın maliyetinin küçük ölçeğe kıyasla çok daha yüksek olduğunu göstermiştir. Yerel aramanın güçlü biçimde uygulanması çözüm kalitesini artırmakla birlikte toplam hesaplama süresini belirgin ölçüde artırmakta, aşırı yerel arama ise arama sürecinin gereksiz yere uzamasına neden olmaktadır. Buna karşılık ada sayısının artırılmasının çözüm kalitesini ve arama stabilitesini önemli ölçüde güçlendirdiği pilot çalışmada açıkça görülmüştür. Bu üç kritik bulgu temel alınarak orta ölçek için yine hız, denge ve kalite odaklı ön ayar setleri oluşturulmuştur.

Denge odaklı ön ayar seti, orta ölçekli problemlerde elde edilen en tutarlı sonuçlar üzerine inşa edilmiştir. Çeşitliliği koruyabilmek için popülasyon boyutu uygun bir seviyede tutulmuş, nesil sayısı çözüm uzayının genişliği nedeniyle artırılmıştır. Maliyetin belirgin şekilde artması nedeniyle yerel arama yoğunluğu küçük problemlere göre daha düşük düzeyde bırakılmıştır. Buna karşın en yakın komşu tabanlı başlangıç oranı kaliteyi geliştirmek için korunmuştur. Ada sayısının orta düzeyde artırılması hem küresel çeşitliliği güçlendirmiş hem de aramanın kararlı şekilde ilerlemesine katkı sunmuştur. Sonuç olarak bu ön ayar seti genel amaçlı ve süre ile kalite açısından denge sağlayacak bir yapı olarak belirlenmiştir.

Hız odaklı ön ayar seti, orta ölçeğin gerektirdiği hız ihtiyacına yanıt verecek biçimde yerel arama oranlarını minimize eden, popülasyon boyutunu azaltan ve aday liste genişliğini düşüren bir yapıda tasarlanmıştır. Yerel aramanın azaltılması süreyi büyük oranda kısalttığından bu ön ayar seti özellikle büyük veri setlerinde çok sayıda tekrarın gerçekleştirilmesi gereken durumlar için idealdir. Mutasyon oranı çeşitliliği korumak adına artırılmış, göç aralığı sıklaştırılarak bilgi akışı hızlandırılmıştır. Bu yapı, hızlı fakat kabul edilebilir kalitede çözümler elde etmek için optimize edilmiştir.

Kalite odaklı ön ayar seti ise orta ölçekli problemlerde yüksek kaliteli çözüm arayışına yönelik olarak geliştirilmiştir. Popülasyon boyutu artırılmış, nesil sayısı yükseltilmiş ve yerel arama yoğunluğu dengeli ön ayar setine göre daha yüksek orana çıkarılmıştır. Aday komşuluk listesi genişletilerek iyileştirme kalitesi artırılmış; ada sayısı çeşitliliği maksimum düzeyde korumak amacıyla yüksek tutulmuştur. Üreme ve yamyamlık baskıları güçlü bireylerin korunmasını sağlamak üzere ayarlanmış; böylece arama süreci uzun fakat kaliteli bir iyileştirme döngüsüne dönüşmüştür. Tablo 6.6 orta ölçekli problemler için ön ayar seti değerlerini sunmaktadır.

Tablo 6.6 Orta Ölçekli Problemler için Ön Ayarlı Parametre Seti

Parametre / Set	Hız Odaklı	Denge odaklı	Kalite Odaklı
Popülasyon boyutu	224	256	320
Maksimum nesil	1200	1500	2000
Üreme oranı	0,58	0,66	0,72
Yamyamlık oranı	0,35	0,40	0,45
Mutasyon oranı	0,40	0,32	0,28
Yavru bireylere uygulanan yerel arama oranı	0,22	0,30	0,45
Mutant bireylere uygulanan yerel arama oranı	0,10	0,15	0,22
En yakın komşu tohumlama oranı	0,25	0,28	0,30
Ada sayısı	5	6	8
Göç aralığı	8	10	12
2-opt Aday Komşu Sayısı	28	32	40
En yakın komşu tabanlı başlangıç oranı	–	0,30	0,30

Büyük ölçekli TSPLIB problemleri için parametrelerin belirlenmesi, küçük ve orta ölçekli problemlere kıyasla çok daha hassas bir eniyileme gerektirmiştir. Pilot çalışma, düğüm sayısı arttıkça çözüm uzayının aşırı genişlediğini, yerel aramanın maliyetinin büyük oranda yükseldiğini ve çeşitliliğin kontrol altına alınmadığı durumda algoritmanın erken yakınsamaya eğilim gösterdiğini ortaya koymuştur. Bu nedenle büyük ölçekli veri setleri için ön ayar setleri belirlenirken çeşitlilik yönetimi, güçlü ada yapısı ve aday komşuluk listesinin genişliği temel belirleyiciler olmuştur.

Denge odaklı ön ayar seti, pilot çalışmada en güçlü genel performansı veren yapı üzerine kurulmuştur. Popülasyon boyutu hesaplama yükünü aşırı artırmadan çeşitliliği koruyacak şekilde orta seviyede tutulmuştur. Yerel arama oranları büyük ölçekli problemlerde aşırı maliyet yaratmaması için sınırlanmış; buna karşın aday komşuluk listesi geniş tutularak yerel iyileştirme kalitesi korunmuştur. Ada sayısının artırılmasının çözüm kalitesini belirgin biçimde iyileştirdiği gözlemlendiğinden ada sayısı orta ölçekli problem setine göre daha yüksek seçilmiştir. Göç aralığı ise hem bilgi paylaşımını geciktirmeyecek hem de maliyeti aşırı artırmayacak düzeyde ayarlanmıştır. Böylece denge odaklı ön ayar seti büyük ölçekli problemler için en güvenilir genel amaçlı yapı olarak belirlenmiştir.

Hız odaklı ön ayar seti, büyük ölçekli problemlerde süreyi azaltma amacıyla yerel arama yoğunluğunu en düşük düzeye indiren, popülasyon boyutunu düşüren ve aday komşuluk listesini daraltan bir yapıda tasarlanmıştır. Yerel arama maliyetinin aşırı yüksek

olduğu büyük ölçekli veri setlerinde bu yapı süreyi çok belirgin biçimde azaltmaktadır. Mutasyon oranı çeşitliliği korumak için artırılmış; göç aralığı sıklaştırılarak bilgi akışı hızlandırılmıştır. Bu ön ayar seti geniş veri setleri üzerinde çoklu tekrar gerektiren veya süresi kısıtlı uygulamalarda bilinen en iyiye yakın sonuçların hızlı şekilde elde edilmesi için uygundur.

Kalite odaklı ön ayar seti ise büyük ölçekli veri setlerinde çözüm kalitesini en üst seviyeye çıkaran yapı olarak tasarlanmıştır. Popülasyon genişletilmiş, nesil sayısı artırılmış ve ada sayısı daha da yükseltilmiştir. Yerel arama oranı maliyet göz önünde bulundurularak dengeli bir biçimde artırılmış; aday komşuluk listesi daha geniş tutulmuş ve elit birey baskısı güçlendirilmiştir. Tablo 6.7 büyük ölçekli problemler için ön ayar seti parametrelerini sunmaktadır.

Tablo 6.7 *Büyük Ölçekli Problemler için Ön Ayarlı Parametre Seti*

Parametre / Set	Hız Odaklı	Denge odaklı	Kalite Odaklı
Popülasyon boyutu	192	256	320
Maksimum nesil	1600	2000	2600
Üreme oranı	0,58	0,60	0,66
Yamyamlık oranı	0,35	0,35	0,40
Mutasyon oranı	0,32	0,30	0,28
Yavru bireylere uygulanan yerel arama oranı	0,10	0,12	0,18
Mutant bireylere uygulanan yerel arama oranı	0,04	0,06	0,10
En yakın komşu tohumlama oranı	0,20	0,22	0,25
Ada sayısı	5	6	8
Göç aralığı	12	18	20
2-opt Aday Komşu Sayısı	40	48	64
En yakın komşu tabanlı başlangıç oranı	—	0,25	0,25

Genel olarak büyük ölçekli ön ayar setlerinin belirlenmesinde üç kritik pilot çalışma bulgusu etkili olmuştur. İlk olarak çözüm uzayının büyümesi nedeniyle çeşitliliğin korunmasının zorunlu hale gelmesi, bir diğeri yerel arama maliyetinin problem boyutuyla birlikte hızla artması ve son olarak ada modelinin çözüm kalitesini küçük ve orta ölçekli problemlere kıyasla çok daha fazla iyileştirmesidir. Bu nedenle büyük ölçekli veri setleri için hem hesaplama maliyeti hem çözüm kalitesi açısından esnek ve ölçeklenebilir bir çerçeve oluşturulmuştur.

Sonuç olarak pilot çalışma bulguları ışığında oluşturulan hız, denge ve kalite odaklı ön ayar seti yapıları, önerilen PH-BWO algoritmasının farklı problem ölçeklerinde ölçeklenebilir ve esnek bir davranış sergilemesini mümkün kılmıştır. Küçük ölçekli problemler için agresif sömürü odaklı yapıların sorunsuz çalıştığı; orta ölçekli problemler için çeşitlilik yönetimi ve kontrollü yerel arama dengesinin kritik olduğu; büyük ölçekli örneklerde ise ada yapısının güçlendirilmesi, yerel arama maliyetinin dikkatle yönetilmesi ve çeşitliliğin korunmasının çözüm kalitesi açısından belirleyici olduğu deneysel olarak doğrulanmıştır. Her ölçek için hız, denge ve kalite odaklı üç ayrı parametre yapısı sunularak algoritmanın farklı kullanım senaryolarına uyarlanabilirliği artırılmıştır. Bu sistematik yapı, PH-BWO'nun pratik uygulamalarda daha kararlı, daha esnek ve problem ölçeğine daha duyarlı bir performans göstermesine olanak tanımıştır.

6.2. Deneylerde Kullanılan Tekrar Sayısı İçin İstatistiksel Analiz

Metasezgisel algoritmalar doğası gereği rastlantısal bileşenler içerdiğinden gerek çözüm kalitesi gerekse süre performansının güvenilir biçimde yorumlanabilmesi için örneklem büyüklüğünün belirlenmesi kritik öneme sahiptir. Çalışmada bağımsız on tekrar yapılmış ve sonuçlar değerlendirilmiştir. Bu çalışmada her problem için on bağımsız tekrar yapılmasının temel nedeni, metasezgisel algoritmaların değişken doğası gereği hem çözüm kalitesi hem süre performansında rastlantısal varyansı doğru şekilde gözlemleyebilecek en küçük örneklem değerinin bu aralıkta tanımlanmış olmasıdır. Yapılan literatür araştırmasında metasezgisel yöntemlerin deneysel analizinde 10–30 tekrarın standart uygulama olduğu görülmüş; varyans, standart sapma, medyan–ortalama ilişkisi ve güven aralığı gibi temel istatistiksel ölçümlerin on tekrar ile güvenilir biçimde hesaplanabildiği kabul edilmektedir.

Bu bölümde, önerilen PH-BWO hibrit algoritmasının küçük, orta ve büyük ölçekli TSPLIB problemlerindeki performansı nicel olarak değerlendirilmiştir. Analiz kapsamında üç farklı ön ayar seti (Hız Odaklı, Denge Odaklı, Kalite Odaklı) kullanılarak her problem için on bağımsız tekrar çalıştırılmış; bu tekrarlar üzerinden bulunan en iyi değer, ortalama değer, bilinen en iyi değerden ortalama sapma (%), en iyiye ulaşma oranı, çalışma süresi, standart sapmalar, bulunan en iyi-en kötü değer aralıkları, medyan değerler ve %95 güven aralıkları

hesaplanmıştır. Bu metrikler hem çözüm kalitesinin kararlılığını hem de algoritmanın arama dinamiklerini ön ayar set yapılandırmalarının sonuçlarına dayanarak yorumlanmıştır.

Özellikle küçük ölçekli problemler için varyansın sıfıra yakın olması, on tekrarın çözüm kalitesindeki kararlılığı doğrulamak için fazlasıyla yeterli olduğunu göstermektedir. Orta ve büyük ölçekli problemlerde ise sonuçlarının değişkenlik göstermeye başladığı, bu nedenle on tekrarın varyansı, çarpıklığı ve güven aralığı genişliğini ortaya koyarak algoritmanın davranışını istatistiksel olarak temsil edebildiği görülmüştür. Dolayısıyla tekrar sayısı hem literatür desteği hem de elde edilen ölçümler açısından bu çalışma için metodolojik olarak yeterli ve geçerli kabul edilmektedir. Bu kararı desteklemek amaçlı yapılan istatistiksel çalışma üç ölçek için de açıklanmıştır.

Küçük ölçekli problemler için analiz sonuçları EK-1’de verilmiştir. Bu ölçekteki problemler bulunan en iyi değer, en iyiden sapma yüzdesi ve en iyi bulma oranı metriklerinde tam kararlılık sergilemiştir. Bu problem sınıfı için gerçekleştirilen tüm deneyler, algoritmanın gerek çözüm kalitesi gerekse istatistiksel açıdan tamamen öngörülebilir bir davranış sunduğunu göstermektedir. Her bir küçük ölçekli veri setinde 10 tekrarın tamamında aynı en iyi tur uzunluğunun elde edilmesi, arama uzayının bu ölçekte hızlı biçimde taranabildiğini ve algoritmanın yönlendirilmiş keşif mekanizmalarının etkili çalıştığını göstermektedir.

Küçük ölçekli problemler üzerinde gerçekleştirilen deneyler, tüm ön ayar setlerinin on tekrarın tamamında küresel en iyiyi üretmesi nedeniyle algoritmanın bu problem sınıfında çözüm kalitesi açısından tam bir kararlılık sergilediğini göstermektedir. Bilinen en iyi değerlerin her tekrarda bulunması, standart sapmanın ve bilinen en iyiden sapma yüzdesinin sıfır olması ile %95 güven aralıklarının tek değere eşitlenmesi, değişken unsurların çözüm kalitesine etki etmediğini ve ön ayar setleri arasında kalite bakımından hiçbir ayrım oluşmadığını açık biçimde doğrulamaktadır.

Süre metrikleri değerlendirildiğinde; küçük ölçekte ön ayar setleri arasındaki temel farklılaşmanın çalışma süresi üzerinden ortaya çıktığı görülmektedir. Standart sapma farkları ile bulunan en iyi ve en kötü değer farklarının tüm setlerde sıfır olması ve medyan değerlerinin ortalamaya eşit olması, süre performansının istikrarlı ve öngörülebilir bir dağılıma sahip olduğunu kanıtlamaktadır. Ayrıca %95 güven aralıklarının çoğu problemde hız odaklı ve denge odaklı arasında kesişmemesi, süre farklarının istatistiksel olarak anlamlı

olduđuna işaret etmektedir. Bu durum, ön ayarlı set seçimlerinin küçük ölçekli problemlerde yalnızca işlem süresi üzerinde bir etki yarattığını; kalite metriklerinin ise ön ayarlı setten tamamen bağımsız kaldığını göstermektedir.

Genel olarak, küçük ölçekli TSPLIB problemlerinde algoritma çözüm kalitesi bakımından tamamen kararlı bir yapı sunmakta ve set seçimi yalnızca çalışma süresi üzerinde etkili olmaktadır. Tüm bu veriler göz önüne alındığında algoritmanın küçük ölçekli problemlerde bilinen en iyi değere istikrarlı şekilde ulaştığını göstermektedir. Bu bulgu aynı zamanda, küçük ölçekli GSP örneklerinde arama uzayının göreceli olarak sınırlı olması ve iyi bilinen en iyi yapılar etrafında kümelenmesi nedeniyle algoritmanın deterministik benzeri bir davranış sergileyebildiğine işaret etmektedir. Özellikle 2-opt tabanlı iyileştirmelerin hızlı yakınsama kabiliyeti nedeniyle algoritmanın bu problem sınıfında yüksek kararlılık ve istatistiksel istikrar sağlayabildiğini göstermektedir.

Orta ölçekli problemler için analiz sonuçları EK-2’de verilmiştir. TSPLIB problemleri üzerinde gerçekleştirilen deneyler, algoritmanın farklı ön ayar setleri altında nasıl davrandığını gözlemlemek ve özellikle çözüm kalitesi ile çalışma sürelerinin istatistiksel açıdan nasıl değiştiğini değerlendirmek amacıyla on bağımsız tekrar ile yürütülmüştür. Bu çalışmada orta ölçekli problemlere ilişkin elde edilen sonuçlar hem tam kararlılık gösteren hem de kısmi varyans içeren örnekler üzerinden on tekrarın istatistiksel açıdan yeterli olduğunu açık biçimde ortaya koymaktadır. Orta ölçekli problemler için elde edilen bulgular, küçük ölçekli sınıfta gözlenen tam kararlı yapıdan belirgin ölçüde farklılaşmaktadır. gr137, brg180 ve ts225 örneklerinde tüm ön ayar setleri on tekrar boyunca aynı en iyi tur uzunluğunu üretmiş ve %95 güven aralıkları tek noktaya indirgenmiştir. Bu problemlerde bilinen en iyiden sapma yüzdesinin tüm ön ayar setlerinde sıfır olması algoritmanın orta boyutun alt sınırında halen küçük problemlere benzer biçimde yüksek kararlılığa sahip olduğunu göstermektedir. Dolayısıyla bu örneklerde ön ayar seti seçimi çözüm kalitesi açısından bir ayrım yaratmamakta, yalnızca çalışma süresi üzerinde etkili olmaktadır.

Buna karşılık kroA200, orta ölçekli veri setlerinde çözüm kalitesinin ön ayar setlerine duyarlı hale geldiđi ilk kırılma noktası olarak ortaya çıkmaktadır. Hız ve denge odaklı ön ayar setlerinde en iyi değer artık farklılık göstermekte, kalite odaklı ön ayar setinde ise kararlı yapı korunmaktadır. Bilinen en iyiden ortalama sapma yüzdesi hız odaklı sette ortalama yaklaşık %0,02, denge odaklı sette yaklaşık %0,005 düzeyinde iken kalite odaklı set bilinen

en iyi değere her tekrarda ulaşmaktadır. Bilinen en iyi değerlere her tekrarda ulaşlamaması bu problemde en iyiye ulaşma güvenilirliğinin ön ayar setleri arasında ölçülebilir biçimde ayrıştığını göstermektedir. %95 güven aralıklarının hem en iyi değer hem de süre metrikleri için hız ve kalite odaklı setler arasında kısmen ayrılması, bu farklılıkların rastlantısal değil, istatistiksel olarak anlamlı olduğuna işaret etmektedir.

fl417 problemi ise orta ölçekli sınıfın en zorlayıcı örneği olarak belirgin biçimde diğer veri setlerinden ayrılmaktadır. Bu problemde hiçbir ön ayar seti bilinen en iyiye bulamamıştır. Bilinen en iyiden sapma yüzdesi hız, denge ve kalite odaklı setler için sırasıyla ortalama %1,51, %0,91 ve %0,65 düzeyindedir. Bulunan en iyi değer için standart sapmaların hız odaklı sette yaklaşık 108, denge odaklı sette 104 ve kalite odaklı sette 75 olması, çözüm kalitesinin kroA200'e kıyasla çok daha değişken bir yapıya sahip olduğunu göstermektedir. Çalışma süreleri de ön ayarlı setler arasında belirgin biçimde ayrılmaktadır; hız, denge ve kalite odaklı setler için ortalama süreler sırasıyla yaklaşık 2,66, 4,27 ve 9,34 dakikadır. %95 güven aralıklarının özellikle hız ve kalite odaklı setler arasında hemen hiç kesişmemesi, süre farklarının istatistiksel olarak güçlü bir ayrıma karşılık geldiğini doğrulamaktadır.

Bu sonuçlar bir arada değerlendirildiğinde, orta ölçekli problemlerde on bağımsız tekrarın varyansı, standart sapmayı, güven aralıklarını ve en iyiye ulaşma oranlarındaki değişimleri net biçimde ortaya çıkardığı görülmektedir. gr137, brg180 ve ts225 gibi tam kararlı örneklerde on tekrar, kararlılığı doğrulamak için fazlasıyla yeterlidir. kroA200 gibi kısmi varyans üreten problemlerde ise on tekrar, değişkenliğin boyutunu ve ön ayarlı setler arasındaki ayrımı istatistiksel olarak görünür hale getirmektedir. fl417 gibi belirgin biçimde değişken problemlerde dahi, on tekrar çözüm kalitesindeki farklılığı, sapma yüzdelerini ve süre dağılımlarını ortaya koymak açısından yeterli istatistiksel duyarlılığı sağlamaktadır.

Dolayısıyla orta ölçekli problemlere ilişkin bulgular hem tam kararlı hem de değişken örnekleri kapsayacak biçimde, on tekrarın algoritmanın davranışını temsil etmek için metodolojik olarak yeterli bir örneklem büyüklüğü olduğunu güçlü biçimde desteklemektedir.

Büyük ölçekli TSPLIB problemlerine ilişkin deneysel analizler, algoritmanın rastlantısal doğasının çözüm kalitesi ve süre performansı üzerindeki etkisini en belirgin biçimde gözlemleyebildiğimiz sınıfı oluşturmaktadır. EK-3'te istatistiksel sonuçlar verilmiştir. Bu ölçekte yapılan on bağımsız tekrar, hem bulunan en iyi değer, ortalamadan

sapma yüzdesi ve en iyiye ulaşma oranı gibi kalite metriklerindeki bozulmayı hem de çalışma sürelerindeki dalgalanmayı istatistiksel olarak anlamlandırmaya yetecek bir örneklem sunmaktadır. Raporlanan bulunan ortalama, standart sapma, bulunan en iyi–en kötü değer ve %95 güven aralığı değerleri; farklı ön ayar setlerinin büyük ölçekte nasıl davrandığını, hangi problemlerde kararsızlığın arttığını ve sapma oranlarının hangi aralıkta yoğunlaştığını açık biçimde göstermektedir. Bu yönüyle büyük ölçekli problemlerde, on tekrarın hem kararlılığı hem de varyansı yakalama gücünü en iyi gösteren ölçek düzeyi olarak değerlendirilebilir.

Algoritmanın küçük ve orta ölçekli sınıflarda gözlenen kararlı davranıştan belirgin ölçüde uzaklaştığını göstermektedir. att532, gr666, u724 ve rat783 problemlerinin tüm ön ayar setlerinde en iyi değer metriğinin standart sapmaları kayda değer seviyelere ulaşmıştır. Özellikle gr666 hız odaklı yapılandırmasındaki yaklaşık 3.128 birimlik varyans, çözüm kalitesinin başlatma koşulları, ada etkileşimleri ve yerel arama tetikleyicileri gibi değişken süreçlerden güçlü biçimde etkilendiğini açıkça ortaya koymaktadır. Benzer şekilde u724 ve rat783 gibi daha karmaşık örneklerde kalite odaklı sette dahi sıfır varyans elde edilememesi, ön ayar seti yoğunluğunun çözüm kalitesini stabilize etmekte büyük boyutta yetersiz kaldığını göstermektedir. Bununla birlikte kalite odaklı setin çoğu problemde hız ve denge odaklı sete kıyasla daha düşük varyans üretmesi, yapılandırma etkisinin kaybolmadığını ancak büyük ölçekte belirgin biçimde zayıfladığını işaret etmektedir.

Bilinen iyiden sapma yüzdesi de çözüm kalitesindeki bu bozulmayı desteklemektedir. att532 ve gr666 örneklerinde setler arasında sınırlı iyileşmeler görülse de fark oldukça küçüktür. Buna karşın u724 ve özellikle rat783 problemlerinde sapma değerleri belirgin biçimde artmıştır. rat783'te hız, denge ve kalite odaklı set için ortalama sapmalar sırasıyla yaklaşık %4,60, %4,37 ve %4,32 olup set yoğunluğunun yalnızca sınırlı bir iyileştirme sağlayabildiğini göstermektedir. Bu problemlerde sapmanın yüksek olması, algoritmanın tekrarlarda çözüm kalitesini kararlı biçimde sürdüremediğini doğrulamaktadır.

Süre metrikleri, büyük boyutta ön ayarlı setler arasında belirgin ayrımlar olduğunu göstermektedir. Hız odaklı set tüm problemler için en düşük süreyi üretirken, denge odaklı set genellikle hız odaklının yaklaşık 1,5–2 katı, kalite odaklı ise 2–3 katı süre gerektirmektedir. Süre varyansları ve dağılımsal özellikler de büyük boyutta belirgin biçimde bozulmaktadır. Ortalama değerlerin medyandan büyük olması, özellikle kalite odaklı sette sağa çarpık dağılımların oluştuğunu ve bazı çalıştırmalarda sürelerin aşırı biçimde uzadığını

işaret etmektedir. Güven aralıklarının genişlemesi ve setler arasında çoğu zaman kesişmeyen aralıklar üretmesi hem çözüm kalitesi hem de süre açısından istatistiksel belirsizliğin diğer ölçekli veri setlerine göre büyük boyutta önemli ölçüde arttığını göstermektedir.

Tüm bulgular bir arada değerlendirildiğinde, büyük ölçekli TSPLIB problemlerinde algoritmanın performansının küçük ve orta ölçekli problemlere kıyasla belirgin biçimde zayıfladığı hem en iyi değer hem en iyiden sapma metriklerinde yüksek varyansların ortaya çıktığı ve tüm setlerde bilinen en iyiye ulaşmadığı görülmektedir. Set değişimleri çözüm kalitesini yalnızca sınırlı ölçüde iyileştirebilmekte, buna karşılık hesaplama maliyetini ciddi biçimde artırmaktadır.

Bu ölçek sınıfında on bağımsız tekrar ile elde edilen ortalama, standart sapma ve %95 güven aralıkları hem çözüm kalitesindeki bozulmanın boyutunu hem de süre performansındaki dalgalanmayı açık biçimde görünür kılmaktadır. Tam kararlılığın kaybolduğu, bilinen en iyiye ulaşma yüzdesinin sıfıra indiği ve varyansların yükseldiği bu ortamda dahi on tekrar, algoritmanın büyük ölçekli problemlerdeki davranışını temsil edecek düzeyde açıklayıcı bir istatistiksel temel sunmaktadır. Böylece on tekrarın, büyük ölçekli sınıfta da hem kalite hem süre metrikleri açısından metodolojik olarak yeterli olduğu doğrulanmaktadır.

6.3. Küçük Ölçekli Veri Seti Çalışma Sonuçlarının Değerlendirilmesi

Küçük ölçekli TSPLIB problemlerinde yapılan deneyler, üç ön ayar setinin de çözüm kalitesi açısından tamamen aynı performansı sergilediğini göstermektedir. Tüm ön ayarlı setlerde bilinen en iyi değer bulunması, küçük ölçekli problemlerde çözüm uzayının görece daha sınırlı olması ve aramanın daha öngörülebilir biçimde yönlendirilebilmesiyle ilişkilendirilebilir. Bu sınıfta 2-opt operatörü, popülasyonda üretilen bireylerin başlangıç kalitesinden bağımsız olarak hızlı ve etkili bir iyileştirme sağlamış; yüksek seçim baskısı ile birleştiğinde set farklarını ortadan kaldıran bir yakınsama davranışı oluşturmuştur. Tablo 6.8’de çalışma sonuçları verilmiştir.

Tablo 6.8 *Küçük Ölçekli Problemler İçin PH-BWO Sonuçları*

Problem	P.Seti	Bilinen En iyi	Bulunan En İyi	Bulunan En Kötü	Bulunan Ort.	Nesil	Ort.Sap ma(%)	Ort.Sür e(dk)	Copt
ulysses22	Hız	7013	7013	7013	7013	8	0,00%	0,007	10
	Denge	7013	7013	7013	7013	8	0,00%	0,008	10
	Kalite	7013	7013	7013	7013	8	0,00%	0,007	10
dantzig42	Hız	699	699	699	699	8	0,00%	0,008	10
	Denge	699	699	699	699	10	0,00%	0,012	10
	Kalite	699	699	699	699	10	0,00%	0,008	10
att48	Hız	10628	10628	10628	10628	18	0,00%	0,012	10
	Denge	10628	10628	10628	10628	15	0,00%	0,017	10
	Kalite	10628	10628	10628	10628	20	0,00%	0,015	10
berlin52	Hız	7542	7542	7542	7542	9	0,00%	0,010	10
	Denge	7542	7542	7542	7542	10	0,00%	0,015	10
	Kalite	7542	7542	7542	7542	9	0,00%	0,010	10
eil76	Hız	538	538	538	538	30	0,00%	0,034	10
	Denge	538	538	538	538	25	0,00%	0,042	10
	Kalite	538	538	538	538	43	0,00%	0,055	10
kroA100	Hız	21282	21282	21282	21282	38	0,00%	0,060	10
	Denge	21282	21282	21282	21282	33	0,00%	0,094	10
	Kalite	21282	21282	21282	21282	53	0,00%	0,088	10

Tablo 6.8 incelendiğinde; örneklerde çözüm kalitesinin değişmemesi, bu problem grubunda düğüm sayılarının arama sürecini belirgin ölçüde zorlaştıracak bir karmaşıklık yaratmadığını göstermektedir. Bu durum, küçük ölçek sınıfında problem boyutunun, ön ayarlı set yapılandırmaları arasında kalite açısından fark yaratacak kadar ayırıştırıcı olmadığını düşündürmektedir. Küçük boyut nedeniyle arama yüzeyi daha sınırlı kalmakta, 2-opt'un iyileştirme kapasitesi hemen her koşulda en iyiye ulaşmayı mümkün kılmaktadır. Şekil 6.4'te küçük ölçekli problemlerde ön ayar seti davranışları verilmiştir.



Şekil 6.4. Küçük Ölçekli Problemlerde Ön Ayar Setinde Süre Davranışları

Şekil 6.4 incelendiğinde süre ve yakınsama davranışlarının ise ön ayar setlerinin değişimine göre anlamlı biçimde farklılaştığı görülmektedir. Hız Odaklı set, daha küçük popülasyon ve düşük nesil limiti nedeniyle tüm problemler için en kısa süreyi üretmiş; buna rağmen en iyi değeri kesintisiz şekilde bulmaya devam etmiştir. Bu bulgu, küçük ölçekli problemlerde popülasyon çeşitliliğinin düşük olmasının performans kaybına yol açmadığını göstermektedir. Denge Odaklı set, bazı problemler için biraz daha hızlı yakınsama sergilese de toplam süre açısından kayda değer bir üstünlük sunmamıştır. Kalite Odaklı set, büyük popülasyon ve yoğun yerel arama stratejileri kullanmasına rağmen bu problem sınıfında ek bir kalite kazanımı sağlamamış; yalnızca süre maliyetini artırmıştır.

Genel olarak deęerlendirildięinde kk lekli problemlerde zm kalitesini belirleyen temel unsur n ayarlı set yapısı deęil, problem boyutunun saęladığı sınırlı arama uzayıdır. Dęk dęm sayısı, arama srecinin hem daha hızlı hem daha kararlı biimde en iyiye ynelmesine olanak tanımaktadır. Bu nedenle en rasyonel n ayarlı set seeneęi, zm kalitesini korurken sreyi en dęk seviyede tutan hız odaklı set olarak belirlenmiştir. Denge odaklı set yalnızca yakınsama hızı aısından sınırlı farklılık yaratırken; kalite odaklı set kk lekli problemlerde anlamlı bir katkı sunmamakta ve yalnızca hesaplama maliyetini artırmaktadır.

6.4. Orta lekli Veri Seti alışma Sonularının Deęerlendirilmesi

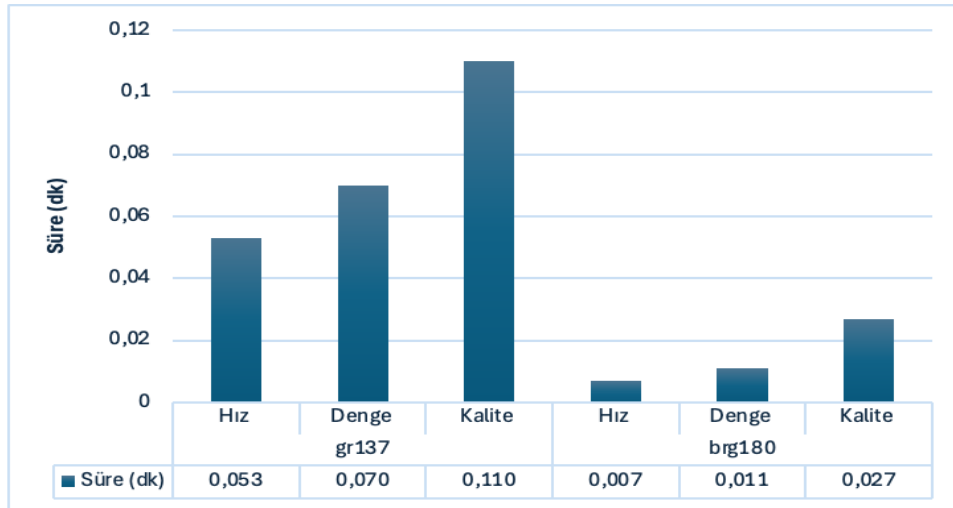
Orta lekli TSPLIB problemlerinde gerekleřtirilen deneyler, kk lekli sınıfta gzlenen yksek kararlılıęın byk lde devam ettięini, ancak bu lekte n ayar setleri arasındaki farklılıkların artık daha grnr hale geldięini gstermektedir. Bu problem grubunda  n ayar seti de oęu durumda literatrdeki bilinen en iyi deęere ulařabilmiş; zm kalitesi genel olarak yksek kararlılık sergilemiştir. Bununla birlikte n ayarlı set yapılandırmalarının yakınsama hızı ve alışma sresi zerindeki etkisi kk lekli problemlere kıyasla daha belirgin hale gelmiştir. zellikle poplasyon byklę, yerel arama yoęunluęu ve ada sayısı gibi parametreler orta lekli problemlerde sreyi anlamlı biimde artırırken, zm kalitesine katkıları problemin yapısına gre deęiřkenlik gsterebilmektedir.

Genel olarak orta lekli problem grubunda n ayarlı set farkları zm kalitesinden ziyade yakınsama nesli, sre performansı ve varyans dzeylerinde ortaya ıkmaktadır. zm kalitesinin byk blmnde kararlılık korunmakla birlikte, n ayarlı set yoęunluęu arttıka sre maliyetinin daha hızlı ykseldięi gzlenmiştir. Tablo 6.9’da alışmanın orta lekli veri seti sonuları sunulmuřtur.

Tablo 6.9 Orta Ölçekli Problemler İçin PH-BWO Sonuçları

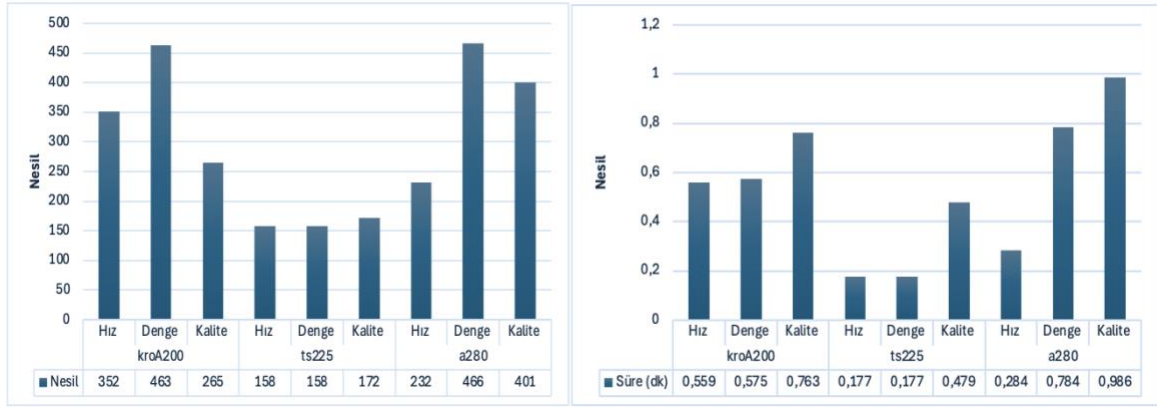
Problem	P.Seti	Bilinen En iyi	Bulunan En İyi	Bulunan En Kötü	Bulunan Ort.	Nesil	Ort. Sapma (%)	Ort. Süre (dk)	Copt
gr137	Hız	69853	69853	69853	69853	118	0,00%	0,053	10/10
	Denge	69853	69853	69853	69853	97	0,00%	0,070	10/10
	Kalite	69853	69853	69853	69853	56	0,00%	0,110	10/10
brg180	Hız	1950	1950	1950	1950	24	0,00%	0,007	10/10
	Denge	1950	1950	1950	1950	19	0,00%	0,011	10/10
	Kalite	1950	1950	1950	1950	14	0,00%	0,027	10/10
kroA200	Hız	29368	29368	29382	29368	352	0,00%	0,559	6/10
	Denge	29368	29368	29382	29368	463	0,00%	0,575	9/10
	Kalite	29368	29368	29368	29368	265	0,00%	0,763	10/10
ts225	Hız	126643	126643	126643	126643	158	0,00%	0,177	10/10
	Denge	126643	126643	126643	126643	158	0,00%	0,177	10/10
	Kalite	126643	126643	126643	126643	172	0,00%	0,479	10/10
a280	Hız	2579	2579	2579	2579	232	0,00%	0,284	10/10
	Denge	2579	2579	2579	2579	466	0,00%	0,784	10/10
	Kalite	2579	2579	2579	2579	401	0,00%	0,986	10/10
f1417	Hız	11861	11882	12182	11989	944	0,18%	2,689	0/10
	Denge	11861	11866	12160	11947	1.281	0,04%	4,305	0/10
	Kalite	11861	11878	12063	11903	1.318	0,14%	9,379	0/10

Orta ölçekli veri setlerine ilişkin tablo incelendiğinde problemlerin üç ayrı davranış kümesine ayrıldığı görülmektedir. Şekil 6.5'te orta ölçekli problemlerde ilk grup için ön ayarlı set analizinde süre farkları verilmiştir.



Şekil 6.5. İlk Grup İçin Ön Ayarlı Set Analizinde Süre Farkları

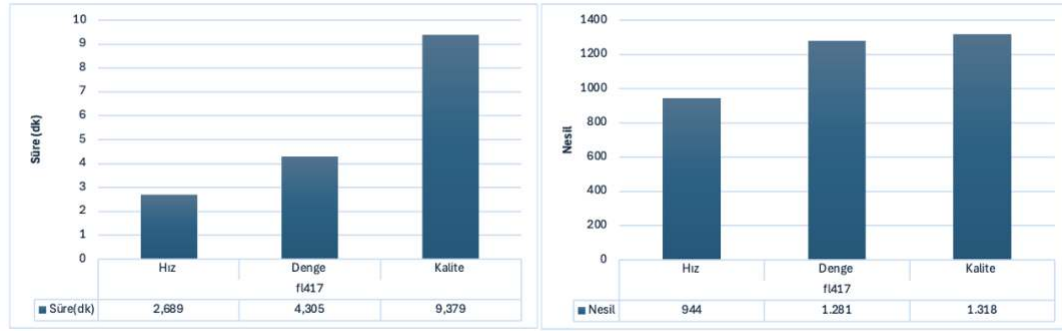
Şekil 6.5 incelendiğinde ilk grupta yer alan gr137 ve brg180 problemlerinde farklılığın yalnızca süre açısından ortaya çıktığı ve küçük ölçekli problemlere benzer biçimde tam bir kararlılık sergilediği görülmektedir. Her iki problemde de üç ön ayar seti tüm tekrarlar boyunca aynı bilinen en iyi değere ulaşmış, ortalama sapma yüzdesi sıfır olmuş ve varyans oluşmamıştır. Bu kümede ön ayarlı set farkları yalnızca süreye yansımakta; kalite açısından hiçbir farklılık oluşmamaktadır. İkinci grupta yer alan kroA200, ts225 ve a280 problemleri çözüm kalitesi bakımından yine yüksek kararlılık göstermiş; tüm ön ayarlı setlerde literatürdeki en iyi değere ulaşılmıştır. Buna karşın bu grup, ilk gruba göre daha duyarlı bir davranış sunmaktadır. Şekil 6.6’da orta ölçekli problemlerde ikinci grup için ön ayarlı set analizinde nesil ve süre farkları verilmiştir.



Şekil 6.6. İkinci Grup İçin Ön Ayarlı Set Analizinde Nesil ve Süre Farkları

Şekil 6.6 incelendiğinde; ön ayar setleri arasında bilinen en iyi değere yakınsama nesilleri arasında belirgin farklar olduğu, ön ayarlı set yoğunluğu arama hızını ve süre performansını daha fazla etkilemeye başladığı görülmüştür. Örneğin kroA200’de denge odaklı set en iyiye daha erken nesilde ulaşırken süre maliyeti artmış; hız ve denge odaklı setlerde ise yakınsama daha geç gerçekleşmiştir. ts225 ve a280 problemlerinde ön ayarlı set farkları daha çok süreye yansımış, kalite ise tüm setlerde sabit kalmıştır. Bu üç problem, çözüm uzayının tamamen basit olmadığı ancak en iyiye ulaşılabilirliğin korunduğu, parametre yoğunluğunun ise nesil ve süre üzerinde belirgin etkiye sahip olduğu ara grubu oluşturmaktadır.

Üçüncü kümede ise yalnızca fl417 bulunmaktadır. Bu problem, orta ölçekli sınıfın belirgin biçimde en zorlu örneğidir. Hiçbir ön ayarlı set en iyi değere ulaşamamış, ancak bilinen en iyiden ortalama sapma yüzdeleri düşük seviyelerde kalmıştır. Şekil 6.7’de orta ölçekli problemlerde son grup için ön ayarlı setlerde süre ve nesil analizi verilmiştir.



Şekil 6.7. Son Grup İçin Ön Ayarlı Setlerde Süre ve Nesil Analizi

Şekil 6.7 incelendiğinde; ön ayarlı set farklarının burada yalnızca süre ve nesil sayısında anlamlı fark yarattığı gözlemlenmiştir. Bu özellikleri dolayısıyla fl417, orta ölçekli problem grubunda ayrı bir grup oluşturmaktadır. Parametre karşılaştırmaları, dengeli ön ayar setinin %0,04 ile en düşük sapmayı ürettiğini, kalite ve hız odaklı setlerin ise daha yüksek sapmalar ve önemli süre maliyetleriyle çalıştığını göstermektedir.

Bu üç grup birlikte değerlendirildiğinde orta ölçekli problemlerde ön ayarlı set yapılandırmalarının etkisinin küçük ölçekli sınıfa kıyasla çok daha belirgin hale geldiği görülmektedir. İlk grupta problem kolaylıkla çözülebilirken ön ayarlı set farkları yalnızca süreye yansımakta; ikinci grupta ön ayarlı set yoğunluğu arama hızını doğrudan şekillendirmekte; üçüncü grupta ise ön ayarlı setler çözüm kalitesi açısından da ayrılmaktadır. Bu nedenle pratik uygulamalar açısından orta ölçekli problemlerde dengeli ön ayar seti genel olarak en istikrarlı yapılandırma olarak öne çıkarken, hız odaklı set süre avantajı ile zaman kısıtı olan durumlar için uygun bir seçenek sunmakta; kalite odaklı set ise yalnızca derin arama gerektiren senaryolar için tercih edilmelidir.

6.5. Büyük Ölçekli Veri Seti Çalışma Sonuçlarının Değerlendirilmesi

Büyük ölçekli TSPLIB problemleri üzerinde yapılan deneysel analizler, PH-BWO algoritmasının çözüm kalitesinin problem boyutu arttıkça belirgin biçimde zorlandığını, buna rağmen ön ayar setlerinin arama davranışını anlamada net bir ayırım sunduğunu göstermektedir. Bu ölçekte 500–800 düğümlü problemler yer almakta olup, bu problemler küçük ve orta ölçekli örneklerle kıyasla çok daha geniş bir çözüm uzayı oluşturmaktadır. Küçük ve orta ölçekte gözlenen -tüm ön ayarlı setler en iyi değere ulaşabilir- davranışının büyük ölçekte tamamen ortadan kalkması, arama yüzeyinin düğüm sayısına bağlı olarak daha zor bir yapıya geçmesinden kaynaklanmaktadır. Tablo 6.10’da çalışma sonuçları sunulmaktadır.

Tablo 6.10 Büyük Ölçekli Problemler İçin PH-BWO Sonuçları

Problem	P.Seti	Bilinen En iyi	Bulunan En İyi	Bulunan En Kötü	Bulunan Ort.	Nesil	Ort. Sapma (%)	Ort. Süre (dk)	Copt
att532	Hız	27686	28333	28977	28551	1444	2,34%	3,633	0/10
	Denge	27686	28310	28958	28658	1960	2,25%	5,875	0/10
	Kalite	27686	28259	28951	28483	2408	2,07%	9,136	0/10
gr666	Hız	294358	306107	316124	308893	1899	3,99%	6,540	0/10
	Denge	294358	303354	314196	309355	1429	3,06%	9,725	0/10
	Kalite	294358	305430	313880	308282	2440	3,76%	15,488	0/10
u724	Hız	41910	43184	44102	43621	1494	3,04%	6,314	0/10
	Denge	41910	43179	43779	43378	1951	3,03%	9,774	0/10
	Kalite	41910	43258	43588	43421	2380	3,22%	15,910	0/10
rat783	Hız	8806	9135	9259	9219	1465	3,74%	9,413	0/10
	Denge	8806	9115	9274	9198	1840	3,51%	12,077	0/10
	Kalite	8806	9099	9246	9195	2434	3,33%	19,325	0/10

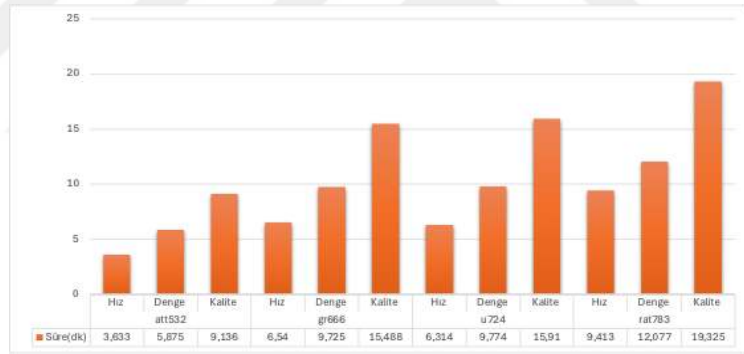
Tablo 6.10 değerlendirildiğinde, dört problemin üç farklı davranış grubu oluşturduğu görülmektedir. İlk grupta olan att532, büyük ölçekli sınıfta en istikrarlı sonuçların elde edildiği örnektir. Bu problemde üç ön ayar setinin de en iyiden sapma değerleri birbirine oldukça yakın olup %2,07–2,34 aralığında kalmıştır. Popülasyon veya yerel arama yoğunluğunun artırılması çözüm kalitesine sınırlı katkı sağladığından kalite odaklı ön ayarlı set, att532’de belirgin bir avantaj yaratmamıştır.

İkinci grup olan gr666, büyük ölçekli sınıf içinde başarıyı en çok zorlayan örneklerden biridir. Üç ön ayar setinin sapmaları %3,06–3,99 bandında seyretmiş, varyans değerleri ise gruptaki diğer problemlere göre daha yüksek gerçekleşmiştir. Bu durum, gr666’nın çözüm

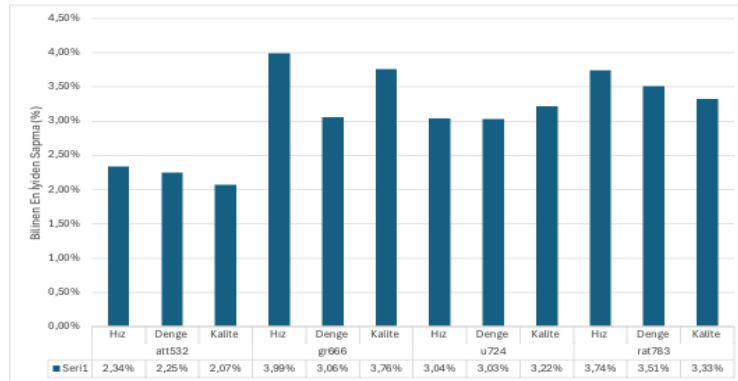
uzayında daha düzensiz bir yapılanma olduğuna işaret etmektedir. Özellikle hız odaklı ön ayarlı setin en yüksek sapmayı üretmiş olması parametre yoğunluğunun bu problemde kısmen olumlu etkisi olduğunu göstermekle birlikte, kalite odaklı ön ayar setinin süre maliyeti kalite kazanımını karşılamamaktadır.

Üçüncü grubu oluşturan u724 ve rat783, en iyiden sapma %3–3,8 aralığında olup üç ön ayarlı set arasında yalnızca sınırlı farklar oluşmuştur. u724’te dengeli ön ayarlı set daha düşük sapma üretmiş, rat783’te ise kalite ön ayarlı setinin bir miktar daha iyi performans göstermesine karşın süre maliyeti anlamlı biçimde yükselmiştir. Bu iki problemde ön ayarlı set farklarının kaliteye değil, daha çok süre ve kararlılığa yansıdığı görülmektedir

Büyük ölçekli problemlerde ön ayar setlerinin süre ve yakınsama davranışı daha belirgin bir ayrım göstermektedir. Şekil 6.8’de büyük ölçekli problemler için ön ayar setleri süre değerleri, Şekil 6.9’da ise bilinen en iyiden ortalama sapma yüzdeleri verilmiştir.



Şekil 6.8. Büyük Ölçekli Problemler İçin Ön Ayar Setleri Süre Değerleri



Şekil 6.9. Ön Ayar Setleri Bilinen En İyiden Sapma Yüzdeleeri

Şekil 6.8 ve 6.9 incelendiğinde; hız odaklı ön ayarlı set, küçük popülasyon ve düşük yerel arama maliyeti nedeniyle tüm problemler için en kısa süreyi üretmiştir; ancak bu avantaj daha yüksek sapma değerleri ile gelmektedir. Dengeli ön ayarlı set, hız ön ayarlı sete göre daha düşük sapma üreten bir yapı sunsa da süre maliyeti ciddi ölçüde artmıştır. Kalite odaklı ön ayarlı set, popülasyon büyüklüğü ve yerel arama oranlarını maksimum seviyeye çıkarmasına rağmen çözüm kalitesinde yalnızca marjinal iyileştirmeler sağlamış; özellikle süre açısından en maliyetli yapılandırma olmuştur.

Genel olarak değerlendirildiğinde büyük ölçekli problemlerde hiçbir ön ayar seti en iyi değere ulaşamamış ancak %2–4 bandında tutarlı ve pratik açıdan kabul edilebilir sapmalar üretilmiştir. Bu bulgu, PH-BWO’nun büyük ölçekli problemlerde tamamen başarısız olmadığını; aksine düğüm sayısı arttığında bile en iyiye oldukça yakın çözümler üretebildiğini göstermektedir. Bununla birlikte bu ölçeklerde algoritmanın keşif kapasitesinin sınırlı kaldığı, daha güçlü yerel arama operatörleri veya çoklu yeniden başlatma gibi mekanizmalara ihtiyaç duyulduğu anlaşılmaktadır.

Ön ayarlı setler karşılaştırıldığında hız odaklı ön ayarlı set, düşük süre ve kabul edilebilir sapma bileşimiyle pratik uygulamalar için en rasyonel yapılandırma olarak öne çıkmaktadır. Denge odaklı ön ayarlı set bazı örneklerde daha iyi sapma üretse de süre maliyeti bu kazanımları gölgelemiştir. Kalite odaklı ön ayarlı setinin sunduğu sınırlı kalite artışı ise büyük ölçekli problemlerde yüksek süre maliyeti nedeniyle gerek akademik gerek pratik kullanımda daha az tercih edilebilir görünmektedir.

Tüm sonuçlar değerlendirildiğinde; uygulama açısından büyük ölçekli problemler için hız odaklı ön ayarlı setin sunduğu süre–kalite dengesi en verimli çözüm yolu olarak öne çıkmaktadır.

6.6. Tüm ölçeklerin genel karşılaştırması ve toplu değerlendirme

Küçük, orta ve büyük ölçekli TSPLIB problemleri üzerinde yürütülen deneysel çalışmalar bütüncül olarak ele alındığında, PH-BWO hibritinin problem ölçeğine duyarlı, kademeli biçimde değişen bir arama dinamiği sergilediği görülmektedir. Problem boyutu büyüdükçe düğüm sayısının artışı ile ön ayar set yapılarının göreceli etkileri belirgin biçimde

farklılaşmış; parametre konfigürasyonlarının arama kalitesine ve süre performansına etkisi ölçek bazında giderek daha görünür hale gelmiştir.

Küçük ölçekli problemler, sınırlı düğüm sayıları sayesinde algoritma için en öngörülebilir ve en istikrarlı sonuçları üretmiştir. Bu sınıfta üç set de tüm örneklerde bilinen en iyi tura ulaşmış; parametrik farklılıklar yalnızca süre üzerinde etkili olmuştur. Bunun başlıca nedeni, küçük problemlerde yerel en iyi yapısının sınırlı olması, 2-opt operatörünün iyileştirme kapasitesinin neredeyse tam etki göstermesi ve popülasyondaki zayıf bireylerin BWO'nun seçim ve yamyamlık mekanizmasıyla hızla elenmesidir. Bu nedenle küçük ölçekte popülasyon büyüklüğü, yerel arama oranı veya nesil sayısı gibi parametrelerde yapılan artışlar kaliteyi yükseltmemekte; yalnızca çalışma süresini uzatmaktadır. Bu yapı, hız odaklı seti küçük ölçek kategorisinde belirgin biçimde en verimli seçenek haline getirmiştir.

Orta ölçekli problemlerde, çözüm uzayının karmaşıklığı yükselmekte, yerel en iyi sayısı artışı belirginleşmektedir. Bu ölçekte ön ayarlı setler arasındaki farklar daha görünür hale gelmiştir. Hız odaklı setin düşük popülasyon yapısı nedeni ile bazı örneklerde en iyiye ulaştığı; denge odaklı setin ise daha geniş popülasyonu ve orta düzey yerel arama oranı sayesinde hem çeşitliliği hem de iyileştirme hızını daha istikrarlı biçimde koruduğu görülmüştür. Kalite odaklı setin bazı örneklerde daha düşük sapma verdiği olsa da bu iyileştirme küçük ölçeklerde olduğu gibi belirgin değildir ve süre maliyeti hızla artmaktadır. Orta ölçekli problemler, algoritmanın keşif ve sömürü dengesinin ilk kez ciddi şekilde sınındığı ölçek aralığıdır.

Büyük ölçekli problemlerde, Hız ön ayarlı setin düşük maliyetli parametre yapısı süre açısından en avantajlı seçenektir, ancak arama uzayının karmaşıklığı karşısında en yüksek sapmayı veren yapı da yine bu set olmuştur. Dengeli set, çeşitlilik kaybı ve erken yakınsama riskini kontrol altında tuttuğu için bazı örneklerde daha stabil sonuçlar üretmiştir; diğer yandan süre maliyeti hızla yükselmiştir. Kalite ön ayarlı setinin geniş popülasyon, artırılmış nesil sayısı ve yoğun 2-opt kullanımı büyük ölçekli problemlerde belirli iyileştirmeler sağlamış olsa da bu kazanım süre maliyetiyle kıyaslandığında oldukça sınırlı kalmaktadır. Bu durum, bu ölçekteki problemler için parametre artırmanın her zaman doğrusal bir kalite kazancı yaratmadığını ortaya koymaktadır.

Tüm veriler birlikte değerlendirildiğinde PH-BWO'nun ölçek arttıkça parametre duyarlılığının yükseldiği, keşif ve sömürü dengesinin daha hassas biçimde ayarlanması gerektiği görülmektedir. Bu kapsamda hız odaklı ön ayar setinin küçük ve orta ölçeklerde sunduğu süre avantajı dikkat çekici bir pratiklik sağlarken, büyük ölçekli problemlerde sınırlı kalite iyileştirmesine rağmen süre maliyetinin çok ağır olması kalite odaklı ön ayar setinin kullanımını yalnızca araştırma amaçlı durumlarla sınırlamaktadır. Denge odaklı ön ayarlı set ise her iki uç arasında daha kontrollü ve istikrarlı bir seçenek sunmakla birlikte büyük ölçeklerde maliyet ve fayda dengesinin dikkatle gözden geçirilmesini gerektirmektedir.

Bu sonuçlar, algoritmanın farklı ölçeklerde neden bu şekilde davrandığını açıklayan tutarlı bir çerçeve sunmakta ve GSP'nin ölçekle birlikte değişen özelliklerinin arama stratejilerini doğrudan belirlediğini ortaya koymaktadır.

6.7. Karşılaştırmalı Performans Analizi ve Kıyaslama

Literatürde BWO tabanlı hibrit yaklaşımların GSP üzerindeki performansı, büyük ölçüde kullanılan yerel arama bileşeninin gücüne ve çözüm uzayında keşif-sömürü dengesinin ne ölçüde korunabildiğine bağlıdır. Bu kapsamda Jabbar ve Ku-Mahamud (2021) ile Cherrat ve Essaid'in (2024) çalışmaları, hibrit BWO ailesi içinde yöntemsel ve deneysel açıdan en önemli referans noktaları arasında yer almaktadır.

Jabbar ve Ku-Mahamud 2021 yılında Karadul Örümcek Algoritmasını GSP'e uyarlamıştır. Araştırmacılar BWO'nun güçlü bir popülasyon tabanı olmasına rağmen yerel aramada takılma eğilimi olduğuna dikkat çekmişlerdir. Popülasyon tabanlı algoritmaların tek başına yeterli olmadığını ve yerel arama yöntemleri ile desteklenmesinin dar alanları tarayarak daha kaliteli çözümlere ulaşabileceğini belirtmişlerdir. Yazarların ana motivasyonu klasik BWO'nun hibrit bir yapı kullanarak GSP gibi kesikli ve karmaşık bir problemi çözebileceğidir.

Temelde yavru üretiminin rastgele olması, yerel bölgelerin yeterince taranamaması ve küresel en iyinin etrafında yoğunlaşmanın zayıf olduğunu belirtmiş ve algoritmayı Değişken Komşuluk İnişi (VND) yerel araması ile birleştirmişlerdir. VND bir çözümün etrafında birden çok komşuluk yapısı kullanarak arama yapmakta ve her komşulukta iyileştirme gördükçe o bölge üzerinde yoğunlaşmaktadır. İyi bir çözüm elde edemediğinde komşuluk

yapısını değiştirerek yerel en iyide takılma ihtimalini ciddi bir oranda azaltmaktadır. Algoritma içerisinde bir yakınsama göstergesi tanımlanmış ve durgunluk durumlarında popülasyonu yeniden tazeleyerek ve yeni uzak bölgelerden tekrar arama yapmaya başlayarak yerelde takılı kalmayı engellemeye çalışılmıştır. Popülasyondaki en iyi bireyler geçici bir popülasyona kaydedilmekte ve yavrular ile yamyamlık sonrası çözümler başka bir geçici popülasyona kaydedilmektedir. Eğer ikinci popülasyondaki en iyi çözüm önceki nesil ile aynı ise ilk popülasyon güncellenerek arama yeni bir bölgeden başlatılmaktadır. Bu yamyamlık sisteminin çeşitliliği hızla azaltmasına karşı bir savunma mekanizmasıdır.

TSPLIB kütüphanesinden 16 adet problem üzerinde çalışmışlar ve sonuçları Afrika Bufalo Optimizasyonu (ABO), Karınca Kolonisi Optimizasyonu (ACO), Yapay Arı Kolonisi (ABC) ile karşılaştırmışlardır. Sekiz problemi kapsayan birinci senaryo testlerinde, problemlerin altısında en iyi çözüme ulaşmışlardır. İkinci çalışmada Afrika Bufalo Optimizasyonu, Parçacık Sürü Optimizasyonu, Karınca Kolonisi Optimizasyonu, Hibrit Parçacık Sürüsü/Karınca Kolonisi Optimizasyonu Algoritması (HPSACO) ile karşılaştırma yapmışlar ve on veri setinin sekiz adedinde yine en iyi çözümü bulmuşlardır.

Sonuç olarak Hybrid BWO, klasik BWO'nun hem keşif hem sömürü zayıflıklarını çözen bir yapı olmakla birlikte, GSP veri setleri üzerinde ABO, ACO, ABC, PSO ve HPSACO gibi güçlü algoritmalarından daha başarılı olduğu sonucuna ulaşmışlardır. Ancak yazarlar VND kullanımının zaman maliyetini artırdığını ve BWO'nun artan parametre sayısı nedeni ile parametre ayarlarının hassas olması gerektiğinin altını çizmişlerdir. Tablo 6.11'de küçük ölçekli problemler için ortak çalışılan problemler karşılaştırılmış ve sonuçlar verilmiştir.

Tablo 6.11 Küçük Ölçekli Problemler PH-BWO ile Hybrid BWO Karşılaştırması

Veri Seti	Bilinen En İyi	Metrik	Hybrid BWO	Sapma(%)	PH-BWO	Sapma(%)
berlin52	7542	En İyi	7542	0,00%	7542	0,00%
		Ortalama	7609	0,89%	7542	0,00%
eil76	538	En İyi	538	0,00%	538	0,00%
		Ortalama	541,23	0,60%	538	0,00%
kroA100	21282	En İyi	21298	0,08%	21282	0,00%
		Ortalama	21832,91	2,59%	21282	0,00%

Tabloya göre PH-BWO, veri setlerinde tüm tekrarlar da en iyi değeri bulmuştur. Hybrid BWO ise berlin52 ve eil76’da ortalama değer bazında en iyiyi yakalayamasa da çok düşük sapma göstermekte; ancak kroA100 veri setinde ortalama performansta %2,59 sapma göstermiştir. Bu durum, Hybrid BWO’nun bazı koşullarda en iyiye ulaşsa dahi tekrarlanabilirlik bakımından kararlı olmadığını ortaya koymaktadır. Buna karşılık PH-BWO’nun tüm veri setlerinde bilinen en iyiyi elde etmesi, yöntemin yüksek kararlılığını ve yerel en iyi tuzaklarından kaçınma kapasitesini açık biçimde göstermektedir. Ayrıca Hybrid BWO’nun bazı küçük ölçekli veri setlerinde en iyiye yakın sonuçlar üretmesine rağmen ortalama performansta yaşadığı sapmalar, algoritmanın yüksek keşif kapasitesi olsa dahi istikrar konusunda sınırlılıklar barındırdığını göstermektedir. PH-BWO ise ada modelinin sağladığı çeşitlilik yönetimi ve dinamik yerel arama tetiklemeleri sayesinde için her tekrarda bilinen en iyiye ulaşan tutarlı bir yakınsama davranışı sergilemiştir. Tablo 6.12’de orta ölçek için aynı ölçekte yer alan problemler karşılaştırılmış ve sonuçlar değerlendirilmiştir.

Tablo 6.12 Orta ölçekli Problemler PH-BWO ile Hybrid BWO Karşılaştırması

Veri Seti	Bilinen En iyi	Metrik	Hybrid BWO	Sapma(%)	PH-BWO	Sapma(%)
ch150	6528	En İyi	6531	0,05%	-	-
		Ortalama	6923,13	6,05%	-	-
tsp225	3916	En İyi	3920	0,10%	-	-
		Ortalama	3961,23	1,16%	-	-
gil262	2378	En İyi	2378	0,00%	-	-
		Ortalama	2379	0,04%	-	-
gr137	69853	En İyi	-	-	69853	0,00%
		Ortalama	-	-	69853	0,00%
ts225	126643	En İyi	-	-	126643	0,00%
		Ortalama	-	-	126643	0,00%
a280	2579	En İyi	-	-	2579	0,00%
		Ortalama	-	-	2579	0,00%

Tablo incelendiğinde; PH-BWO ise orta ölçekli veri setlerinin tümünde hem en iyi hem ortalama değerlerde %0 sapma ile çalışmış ve 10/10 en iyi bulma oranı sağladığı

görülmüştür. Dolayısıyla orta ölçekli problemlerde PH-BWO, ortalama sonuçların dalgali olduğu ve en iyi değerin bulunamadığı Hybrid BWO'ya kıyasla çok daha istikrarlı ve tutarlı bir performans profili sergilemiştir. Bunun temel nedeni, orta ölçekli problem sınıfının hem küçük problemlere kıyasla daha geniş bir arama uzayı sunması hem de büyük ölçeğe göre çok daha yönetilebilir yerel en iyi yapıları içermesidir. Bu orta düzey zorluk seviyesi, algoritmaların hem keşif hem sömürü bileşenlerinin etkinliğini daha belirgin biçimde ortaya çıkardığı bir alan oluşturmaktadır. Hybrid BWO'nun VND tabanlı yerel iyileştirme mekanizması, bazı problem örneklerinde yüksek kaliteli çözümler üretebilse de ortalama performansın dalgali olması algoritmanın arama sürecinde sürekliliği korumakta zorlandığını göstermektedir. Özellikle ch150 ve tsp225 gibi örneklerde ortalama değerlerin bilinen en iyi değerden anlamlı biçimde ayrılması, Hybrid BWO'nun yerel arama adımlarının bazı tekrarlar boyunca istenen etkiyi gösteremediğini ve aşırı çeşitlilik kaybı nedeniyle bazı koşullarda çözüm kalitesinin düştüğünü ortaya koymaktadır.

PH-BWO için durum bunun tersidir. Ada modeli yapısı sayesinde popülasyonun farklı alt bölümlere ayrılması, her adanın yarı bağımsız evrime sahip olması ve belirli aralıklarla göç mekanizmasının devreye girmesi, orta ölçekli problemlerde eniyileme sürecinin çeşitlilik kaybı olmadan ilerlemesini sağlamıştır. Bu nedenle PH-BWO yalnızca en iyi değerlerde değil, ortalama değerlerde de sıfır sapma elde etmiş; bu durum algoritmanın hem başlangıç koşullarına hem de rastlantısal varyasyonlara karşı dayanıklı bir yakınsama davranışı sergilediğini göstermektedir.

Tablo 6.12'de görüldüğü üzere Hybrid BWO bazı örneklerde en iyi değere çok yakın sonuçlar üretse de ortalama performansın sapması iki örnekte %1–6 aralığına kadar çıkmaktadır. Özellikle ch150'de ortalama sapmanın %6,05 gibi oldukça yüksek bir seviyeye ulaşması, tekrarlanabilirlik açısından orta ölçekli problemlerde zorlandığının açık bir göstergesidir. Buna karşın PH-BWO'nun aynı grupta yer alan gr137, ts225 ve a280 problemlerinde tüm tekrarlarda %0 sapma sağlaması, yöntemin orta ölçekli TSPLIB ailesi için daha yüksek kararlılık düzeyi sunduğunu kanıtlamaktadır.

Bir diğer önemli nokta, iki yaklaşım arasındaki keşif ve sömürü dengesinin orta ölçekte nasıl davrandığıdır. Hybrid BWO'nun VND tabanlı yerel iyileştirmesi sömürü gücünü artırmakla birlikte, bazı koşullarda keşif kapasitesini sınırlamakta; bu da özellikle çeşitliliğin erken tükendiği tekrarlarda ortalama çözüm kalitesinin düşmesine neden olmaktadır. PH-

BWO ise ada modelinin sağladığı doğal çeşitlilik yönetimi sayesinde hem keşif sürecini canlı tutmakta hem de yerel aramayı aşırı tetiklemeyerek arama sürecini dengeli biçimde ilerletmektedir.

Sonuç olarak orta ölçekli TSPLIB problemlerine ilişkin karşılaştırma, PH-BWO'nun istikrar ve tekrarlanabilirlik açısından Hybrid BWO'ya göre çok daha üstün bir performans sunduğunu, özellikle ortalama performans metriklerinde neredeyse kusursuz bir kararlılık ortaya koyduğunu göstermektedir. Hybrid BWO bazı durumlarda rekabetçi sonuçlar üretse bile, farklı tekrarlar arasında performansın dalgalanması ve ortalama sapmaların yükselmesi yöntemin orta ölçekli arama uzaylarında tutarlılık sorunları yaşadığını ortaya koymaktadır. Buna karşılık PH-BWO'nun ada modeli ve kontrollü yerel arama yaklaşımı, orta ölçek segmentinde her çalıştırmada aynı kaliteyi üretebilmesini sağlayarak bu özelliği bakımından daha güvenilir bir yöntem olduğunu kanıtlamaktadır.

Cherrat ve Essaid ise 2024 yılında Karadul Örümcek Algoritmasını GSP'e uyarlamıştır. Seçim Stratejisi ve Diferansiyel Mutasyon Uyarlamalı Kara Dul Örümcek Optimizasyonu (SDABWO) ismini verdikleri algoritmanın temel amacı, sürekli fonksiyonlar için geliştirilmiş bu algoritmanın kesikli bir probleme uyarlanması ve 2-opt kullanılarak çözüm kalitesinin artırılmasıdır. Çalışmada TSPLIB kütüphanesinden seçilen veri setleri kullanılmıştır. Araştırmacılar, klasik BWO'nun rastgele eş seçimi mekanizması nedeniyle çok sayıda zayıf çocuk ürettiğini ve bu durumun özellikle kesikli problemlerde yakınsama süresi ve doğruluğunu olumsuz etkilediğini belirtmişlerdir.

Yeni bir seçim stratejisi için Gri Kurt Optimizasyonu (GWO), Parçacık Sürü Optimizasyonu (PSO) ve Balina Optimizasyonu Algoritması (WOA) gibi yöntemlerde kullanılan “*en iyi bireyden öğrenme*” fikrinden esinlenmişlerdir. Bu yaklaşımla daha yapısal bir eş seçimi uygulanmış ve daha kaliteli ebeveynlerin bir arada eşleşmesi mümkün kılınmıştır. Buna ek olarak, diferansiyel evrim tabanlı bir mutasyon operatörü kullanılarak popülasyon bireyleri arasındaki fark vektörlerinden yararlanılmış ve rastgele sıçramalar yerine daha yönlendirilmiş atlamalar gerçekleştirilerek algoritmanın keşif gücü artırılmıştır. Üretilen yeni turların 2-opt ile iyileştirilmesi ise çözüm kalitesini belirgin şekilde yükseltmiştir.

TSPLIB kütüphanesinden alınmış 18 küçük ve orta ölçekli örnek için 30 bağımsız tekrar yapılmış ve 18 problemin 16'sında en iyi değerin bulunduğu rapor edilmiştir. a280 ve

lin318 problemlerinde dikkate değer sapmalar gözlenmiştir. Araştırmacılar ayrıca sonuçları Yapay Arı Kolonisi (ABC) ve ABC-ACS hibrit algoritmaları ile karşılaştırmış ve birçok veri setinde BWO'nun en iyi, en kötü ve ortalama değerler açısından rakip yöntemleri geride bıraktığını göstermiştir.

Sonuç olarak, BWO'nun biyolojik temelli yapısının GSP gibi kombinatoriyel bir problem için oldukça güçlü olduğu, hibritleştirildiğinde hem keşif hem sömürü kapasitesinin arttığı ve gelecekte daha karmaşık kısıtlar içeren problemlere de uyarlanabileceği vurgulanmıştır. Tablo 6.13'te ortak çalışılmış olan problemler karşılaştırılmış ve sonuçlar değerlendirilmiştir.

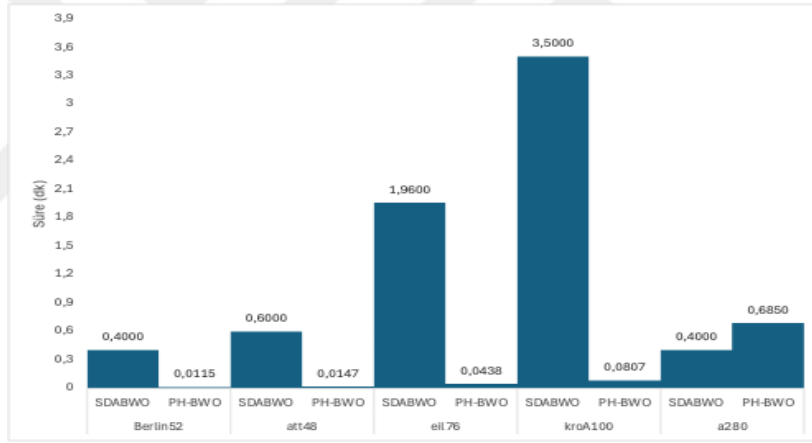
Tablo 6.13 PH-BWO ile SDABWO Süre Karşılaştırılması

Veri Seti	Algoritma	En İyi	Ortalama	Sapma(%)	Süre(dk)	Copt
Berlin52	SDABWO	7542	7542	0,00%	0,4000	30/30
	PH-BWO	7542	7542	0,00%	0,0115	10/10
att48	SDABWO	10628	10628	0,00%	0,6000	30/30
	PH-BWO	10628	10628	0,00%	0,01473	10/10
eil76	SDABWO	538	538	0,00%	1,96000	30/30
	PH-BWO	538	538	0,00%	0,04380	10/10
kroA100	SDABWO	21282	21282	0,00%	3,50000	30/30
	PH-BWO	21282	21282	0,00%	0,08067	10/10
a280	SDABWO	2579	2786	8,02%	0,40000	26/30
	PH-BWO	2579	2579	0,00%	0,68500	10/10

Tablo 6.13'te yalnızca Cherrat ve Essaid çalışması ile ortak kullanılan test problemleri yan yana gösterilmiştir. Bu nedenle tablo, çalışmanın tamamındaki 18 veri setini değil hem SDABWO hem PH-BWO'nun birlikte değerlendirildiği küçük ve orta ölçekli problemleri kapsamaktadır.

Tabloya göre PH-BWO, berlin52, eil76, att48 ve kroA100 veri setlerinde her tekrarda en iyi değeri bulmuş; tüm ortalama değerlerde %0 sapma elde etmiştir. SDABWO ise berlin52,

eil76 ve att48'de ortalama performans açısından da kararlı iken, kroA100 veri setinde ortalama sonuçlarda %2,59 sapma göstermiştir. Bu durum SDABWO'nun bazı koşullarda en iyi değere ulaşsa dahi tekrarlanabilirlik bakımından düşük kararlı olduğunu ortaya koymaktadır. a280 veri seti her iki çalışma tarafından da incelenen veri setleri arasında yer almaktadır. Hybrid BWO'nun bu problemde ortalama tur uzunluğu 2.786'dır ve bu değer, en iyi olan 2.579'a göre %8,02 sapma göstermektedir. PH-BWO ise 10/10 tekrarda en iyi değere ulaşarak ortalama performansta da %0 sapma elde etmiştir. Bu bulgu, özellikle orta ölçek geçiş bölgesinde PH-BWO'nun SDABWO'ya kıyasla çok daha kararlı ve tekrar edilebilir bir yakınsama davranışı sergilediğini göstermektedir. Şekil 6.10 bu iki algoritmanın yakınsama süreleri açısından karşılaştırılmasını göstermektedir.



Şekil 6.10. PH-BWO ile SDABWO Yakınsama Süresi Karşılaştırma

Genel olarak bakıldığında SDABWO, bazı küçük ölçekli veri setlerinde rekabetçi sonuçlar üretebilse de ortalama performansta yaşanan dalgalanmalar, yöntemin tekrarlanabilirlik açısından sınırlılık taşıdığını göstermektedir. Buna karşın PH-BWO'nun ada modeli yapısı, çeşitliliğin korunması ve kontrollü yerel arama tetiklemeleri sayesinde hem en iyi hem ortalama değerlerde tüm tekrarlarda tutarlılık sağlamış ve küçük–orta ölçekli segmentlerde belirgin biçimde üstün bir performans ortaya koymuştur.

Sonuç olarak; literatürde Jabbar ve Ku-Mahamud (2021) ile Cherrat ve Essaid (2024) tarafından sunulan iki farklı hibrit BWO modeli, BWO'nun doğal keşif kapasitesini güçlendirirken erken yakınsama ve çeşitlilik kaybı gibi temel sınırlılıklarını aşmayı hedefleyen önemli adımlar olarak değerlendirilmektedir.

Bu bağlamda geliştirilen PH-BWO modeli, ada temelli paralel nüfus yapısı, kontrollü göç mekanizması ve dinamik 2-opt iyileştiricisi sayesinde literatürdeki hibrit BWO örneklerinin ötesinde daha dengeli bir keşif-sömürü yapısı ortaya koymaktadır. Ortak veri setleri üzerinden yapılan karşılaştırmalar, PH-BWO'nun hem çözüm kalitesi hem kararlılık hem de çalışma süresi açısından belirgin bir üstünlük sağladığını göstermektedir.

PH-BWO'nun en çarpıcı avantajı, kararlılık, hız ve kalite üçlüsünü aynı anda sağlayabilmesidir. Karşılaştırma yapılan çalışmalardaki yüksek keşif kapasitesine rağmen ortalama sonuçlarda dalgalanmalar gösterirken, PH-BWO her veri setinde hem bulunan en iyi değer hem de ortalama değerlerde istikrarlı yakınsama sağlamış, çalışma süresinde ise 30–50 kata varan hızlanma elde edilmiştir. Bu sonuç, ada modeli ile yönlendirilmiş yerel aramanın birlikte kullanımının, BWO'nun stokastik doğasını dengeleyerek arama uzayında çok daha güvenilir bir ilerleyiş sunduğunu ortaya koymaktadır.

Genel olarak değerlendirildiğinde, PH-BWO literatürdeki hibrit BWO yaklaşımlarından farklı olarak yalnızca çözüm kalitesine değil, aynı zamanda yeniden üretilebilirlik, ölçeklenebilirlik ve hesaplama verimliliği gibi pratikte kritik önem taşıyan performans ölçütlerine de güçlü katkılar sunmaktadır. Bu yönüyle PH-BWO modeli hem küçük ve orta ölçekli TSPLIB örneklerinde hem de geniş arama uzayına sahip büyük ölçekli problemler için güvenilir bir çözüm mimarisi olarak öne çıkmakta; BWO tabanlı algoritmaların evriminde yeni bir referans noktası oluşturmaktadır.

7. SONUÇLAR VE ÖNERİLER

Bu çalışmada, Gezgin Satıcı Problemi için paralel ada temelli bir hibrit karadul örümcek optimizasyonu geliştirilmiş; yapıya genetik operatörler, ölçeğe duyarlı bir parametre sistemi ve dinamik 2-opt yerel araması entegre edilmiştir. Tasarlanan PH-BWO modeli, klasik BWO'nun yüksek varyans, erken yakınsama ve büyük ölçeklerde çözüm kalitesi kaybı gibi temel sınırlılıklarını gidermek amacıyla çoklu ada mimarisi, sistematik çeşitlilik yönetimi ve güçlü sömürü mekanizmaları ile yeniden yapılandırılmıştır. Arama sürecinin davranışı, yalnızca elde edilen en iyi çözüm üzerinden değil, aynı zamanda varyans, en iyiden sapma yüzdeleri, çalışma süreleri, bulunan en iyi–en kötü–ortalama performans değerleri ve bilinen en iyi değeri bulma oranları üzerinden istatistiksel olarak değerlendirilmiştir. Böylece algoritmanın farklı ölçeklerdeki kararlılığı, duyarlılığı ve davranış kalıpları nicel olarak ortaya konmuştur.

Elde edilen tüm bulgular, ada modelinin BWO çerçevesinde hem çeşitlilik koruma hem de kararlı yakınsama açısından kritik bir rol üstlendiğini göstermektedir. Önerilen yapıda yamyamlık mekanizmasının ada modeli ile kullanımı, literatürde çoğunlukla dezavantaj olarak nitelenen bu sürecin yönlendirilmiş bir seçim baskısına dönüşmesini sağlamıştır. Yamyamlık, düşük kaliteli bireyleri agresif şekilde eleyerek, ada tabanlı popülasyon dağılımı çeşitlilik kaybını engellemiş; böylece sömürü gücü artarken erken yakınsama riski belirgin biçimde azalmıştır. Bu mekanizmanın dinamik 2-opt ile birleşmesi, çözüm uzayının hem keşif hem sömürü dengesinin sağlanmasını mümkün kılmıştır. Bu yapı, özellikle orta ve büyük ölçeklerde varyansın düşmesi, en iyi çözümden sapma yüzdesinin sıfırlanması veya en aza indirilmesi ve bilinen en iyi değeri bulma oranlarının artması yoluyla istatistiksel olarak doğrulanmıştır.

Çalışma boyunca elde edilen sonuçlar, küçük, orta ve büyük ölçekli TSPLIB problemlerinde farklı ön ayar setlerinin arama davranışını belirgin şekilde etkilediğini göstermiştir. Küçük ölçekli tüm veri setlerinde üç ön ayar seti de bilinen en iyi değeri bulması; varyansın tüm çalışmalarda sıfırlanması, algoritmanın kararlı bir sömürü gücüne ulaştığını göstermiştir. Orta ölçekli veri setlerinde hız odaklı ön ayar setinin süre açısından avantaj sağlarken bazı problemler için varyansın arttığı görülmüş; denge odaklı ön ayar setinin hem yakınsama istikrarı hem ortalama performans hem de çalışma süresi bakımından

en dengeli sonuçları ürettiği belirlenmiştir. Büyük ölçekli veri setlerinde kalite odaklı ön ayar seti daha derin yerel arama sağlamasına rağmen süre maliyeti yüksek bulunmuş; hız odaklı set ise daha düşük maliyetle kabul edilebilir çözüm kalitesi sağlamıştır. Bu bulgular, ölçek duyarlı bir parametrizasyonun eniyileme başarısı için zorunlu olduğunu açıkça teyit etmektedir.

Karşılaştırmalı analizlerde önerilen PH-BWO modeli, literatürdeki iki güçlü hibrit yaklaşım olan Jabbar ve Ku-Mahamud (2021) ile Cherrat ve Essaid (2024) çalışmalarına göre daha istikrarlı bir performans sergilemiştir. Jabbar ve Ku-Mahamud'un VND ve popülasyon yenileme yaklaşımının büyük ölçeklerde varyans artışı ve yüksek süre maliyeti ürettiği görülürken; Cherrat ve Essaid'in 2-opt ve diferansiyel mutasyon tabanlı yapısı küçük veri setlerinde başarılı olsa da orta ölçeklerde performans dalgalanmaları göstermiştir. PH-BWO'nun tüm ortak veri setlerinde bilinen en iyi değeri üretmesi, her tekrarında en iyiye ulaşması ve çalışma süresinde 35–50 kata varan hızlanma sağlaması, önerilen yöntemin hem yakınsama kararlılığı hem çözüm kalitesi hem de hesaplama verimliliği bakımından literatürün mevcut BWO tabanlı hibrit modellerini önemli ölçüde geride bıraktığını göstermektedir. Ayrıca yamyamlık mekanizmasının ada modeli içinde yeniden işlevlendirilmesi ve Türkçe literatürde GSP için ilk kez bütüncül bir hibrit BWO model çerçevesinin sunulması bu çalışmayı yöntembilimsel açıdan özgün kılmaktadır.

Çalışma kapsamında gerçekleştirilen analizlere göre, her ne kadar PH-BWO çoklu ölçeklerde güçlü bir performans ortaya koysa da yöntemin bazı sınırlılıkları bulunmaktadır. Büyük ölçekli problemlerde yerel aramanın maliyeti hızla artmakta; en geniş komşuluk yapılarında aday sayısının büyümesi süre üzerinde doğrusal olmayan bir yük oluşturmaktadır. Ada sayısının artışı çeşitliliği güçlendirse de paralel iletişim maliyetini ve senkronizasyon gecikmelerini artırabilmektedir. Bu sınırlılıkların açıkça tanımlanması, çalışmanın metodolojik olgunluğunu artırmakta ve gelecekte yapılacak geliştirmelere yön göstermektedir.

Elde edilen en iyiden sapma yüzdelerinin büyük ölçekli problemlerde dahi çoğunlukla %3–5 bandında seyretmesi, PH-BWO'nun pratik lojistik ve rota planlama problemlerinde kullanılabilirliğini güçlü biçimde desteklemektedir. Gerçek dünya uygulamalarında, özellikle dağıtım, kargo optimizasyonu ve üretim içi taşımada %3–5'lik sapmalar operasyonel olarak kabul edilebilir sınırlar olarak değerlendirilmektedir. Algoritmanın makul

sürelerde sonuç üretebilmesi, çok tekrarlı senaryolarda veya gerçek zamanlı yeniden planlama gerektiren durumlarda dahi uygulanabilirliği artırmakta; PH-BWO'nun endüstriyel ölçekte kullanımını mümkün kılmaktadır.



KAYNAKÇA

- Ackley, D. H. (1985). A connectionist algorithm for genetic search. In J. J. Grefenstette (Ed.), *Proceedings of the 1st International Conference on Genetic Algorithms* (pp. 121–135). Lawrence Erlbaum Associates.
- Adam, S., Alexandropoulos, S.-A., Pardalos, P., and Vrahatis, M. (2019). No free lunch theorem: A review. In *Springer optimization and its applications* (pp. 57–82). Springer.
- Akgündüz, B. (2022). Darwin’ın doğal seçilimi: Bireysel seçim ve evrim. *Felsefe ve Sosyal Bilimler Dergisi*, 3(1), 1-15.
- Alba, E., and Luque, G. (2007). Parallel metaheuristics: A new class of algorithms. *Wiley*.
- Alba, E., and Tomassini, M. (2002). Parallelism and evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 6(5), 443–462.
- Applegate, D. L., Bixby, R. E., Chvátal, V. ve Cook, W. J. (2006). The traveling salesman problem: A computational study. *Princeton University Press*.
- Bellman, R. (1962). Dynamic Programming Treatment of the Travelling Salesman Problem. *Journal of the ACM*, 9(1), 61–63.
- Bellmore, M. and Hong, S. (1974). Transformation of multi-salesmen problem to the standard traveling salesman problem. *Journal of the ACM*, 21(3), 500–504.
- Blum, C., and Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3), 268–308.
- Cherrat, K., and Riffi, M. E. (2024). Black widow’s new approach to tackle the traveling salesman problem. *Journal of Theoretical and Applied Information Technology*, 102(1), 350–357.
- Christian, B., and Griffiths, T. (2016). *Algorithms to live by: The computer science of human decisions*. Henry Holt and Company.
- Cook, W. J. (2011). *In pursuit of the traveling salesman: Mathematics at the limits of computation*. Princeton University Press.
- Çolak, S. (2010). Genetik algoritmalar yardımı ile gezgin satıcı probleminin çözümü üzerine bir uygulama. *Çukurova Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 19(3), 423-438.
- Dantzig, G. B., Fulkerson, D. R. and Johnson, S. M. (1954). *Solution of a large scale traveling salesman problem*. *Operations Research*, 2(4), 393–410.

- Darwin, C. (2016). *Türlerin Kökeni*, Ankara: Yason Yayıncılık.
- De Jong, K. A. (1985). Genetic algorithms: A 10 year perspective. In J. J. Grefenstette (Ed.), *Proceedings of the 1st International Conference on Genetic Algorithms* (pp. 169–177). Lawrence Erlbaum Associates.
- Dorigo, M. (1992). Optimization, learning and natural algorithms [Yayımlanmamış doktora tezi]. Politecnico di Milano, İtalya.
- Dökeroğlu, T., Canturk, D., and Kucukyilmaz, T. (2024). A survey on pioneering metaheuristic algorithms between 2019 and 2024. *arXiv*.
- Evans, T., Wallis, E. and Elgar, M. (1995). Making a meal of mother. *Nature*, 376(6537), 299.
- Fabricius, J. C. (1775). Systema entomologiae, sistens insectorum classes, ordines, genera, species adiectis synonymis, locis, descriptionibus, observationibus. *Korte*.
- Fister, I., Fister, D., Yang, X.-S. and Brest, J. (2013). A comprehensive review of metaheuristics for optimization. *Information Sciences*, 231, 3–27.
- Floudas, C. A. and Pardalos, P. M. (Ed.). (2009). *Encyclopedia of optimization* (2. baskı). Springer Science+Business Media.
- Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5), 533–549.
- Goldberg, D. E. (1985). Genetic algorithms and rule learning in dynamic system control. In J. J. Grefenstette (Ed.), *Proceedings of the First International Conference on Genetic Algorithms and Their Applications* (pp. 8–19). Psychology Press.
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.
- Grefenstette, J. J. (Ed.). (1985). *Proceedings of the First International Conference on Genetic Algorithms and their Applications*. Psychology Press.
- Gribkovskaia, I., Halskau Sr., Ø. and Laporte, G. (2007). The bridges of Königsberg—A historical perspective. *Networks*, 50(3), 193–203.
- Gutin, G. and Punnen, A. P. (Ed.). (2002). The traveling salesman problem and its variations (Vol. 12). Springer Netherlands.
- Hassanat, A. B. A., Alkafaween, E., Al-Nawaiseh, N. A., Abbadi, M. A., Alkasassbeh, M., and Alhasanat, M. B. (2016). Enhancing Genetic Algorithms using Multi Mutations: Experimental Results on the Travelling Salesman Problem.

- Hayyolalam, V. and Pourhaji Kazem, A. A. (2020). Black widow optimization algorithm: A novel meta-heuristic approach for solving engineering optimization problems. *Engineering Applications of Artificial Intelligence*, 87, 103249.
- Helsgaun, K. (2006). An Effective Implementation of K-opt Moves for the Lin-Kernighan TSP Heuristic. Roskilde Universitet. Roskilde Universitet. Computer Science. Computer Science Research Report
- Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press.
- Holland, J. H. (1992). Genetic Algorithms. *Scientific American*, 267(1), 66–73.
- Hossain, M. A. A. and Yılmaz Acar, Z. (2024). Comparison of new and old optimization algorithms for Traveling Salesman Problem on small, medium, and large-scale b benchmark instances. *Bitlis Eren University Journal of Science*, 13(1), 216–231.
- Hussain, A., Muhammad, Y. S., and Sajid, M. N. (2018). An improved genetic algorithm crossover operator for traveling salesman problem. *Turkish Journal of Mathematics and Computer Science*, 9(1), 1–13.
- Ibrahim, S. A. I. (2020). New global optimization technique by using auxiliary function method in directional search with computer applications [Yayımlanmamış doktora tezi]. Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü.
- Jabbar, A. M., and Ku-Mahamud, K. R. (2021). Hybrid Black Widow Optimization and Variable Neighborhood Descent Algorithm for Traveling Salesman Problem. *International Journal of Systematic Innovation*, 6(5), 32–43.
- Jakšić, Z.; Devi, S.; Jakšić, O. and Guha (2023), K. A Comprehensive Review of Bio-Inspired Optimization Algorithms Including Applications in Microelectronics and Nanophotonics. *Biomimetics* 2023, 8, 278.
- Kamözüt, M. C. (2009). "Doğal seçim yasasının bulunuşu: Darwin ve Wallace'ın yaklaşımlarındaki farklar". *Uludağ Üniversitesi Fen-Edebiyat Fakültesi Felsefe Dergisi*, 12, 201-208.
- Karimi-Mamaghan, M., Mohammadi, M., Meyer, P., Karimi-Mamaghan, A. M. and Talbi, E.-G. (2022). Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: A state-of-the-art. *European Journal of Operational Research*, 296(2), 393–422.

- Kayman, A. Y. (2015). Gezgin satıcı probleminin çözümünde parçacık sürü optimizasyon algoritması performansının bulanık c-ortalamalar yöntemi ile iyileştirilmesi [Yayımlanmamış doktora tezi]. Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü.
- Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, 4, 1942–1948.
- Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680.
- Kobler, D. (2021). Evolutionary algorithms in combinatorial optimization. Y. Malik, A. Iqbal, P. Joshi, S. Agrawal ve F. I. Bakhsh (Ed.), *Metaheuristic and evolutionary computation: Algorithms and applications* (s. 159-179). Springer.
- Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(2), 231–247.
- Li, J. (2024). *Metaheuristic algorithms: Fundamentals and applications*. Springer.
- Lima, S. J. de A., Araújo, S. A. de, and Schimit, P. H. T. (2018). A hybrid approach based on genetic algorithm and nearest neighbour heuristic for CVRP. *Acta Scientiarum. Technology*, 40, e36334.
- Lin, S. and Kernighan, B. W. (1973). *An Effective Heuristic Algorithm for the Traveling Salesman Problem*. *Operations Research*, 21(2), 498–516.
- Matai, R., Singh, S. P. and Mittal, M. L. (2010). Traveling salesman problem: An overview of applications, formulations, and solution approaches. D. Davendra (Ed.), *Traveling salesman problem: Theory and applications* (p. 1–24). IntechOpen.
- Merz, P., and Freisleben, B. (1999). Memetic algorithms for the traveling salesman problem. *Complex Systems*, 13(4), 297–345.
- Mitchell, M. (1996). An introduction to genetic algorithms. *MIT Press*.
- Molina, D., Herrera, F. and González, J. (2020). Taxonomy and performance evaluation of metaheuristic algorithms for combinatorial optimization. *Applied Soft Computing*, 92, 106287.
- Moscato, P. (1989). *On evolution, search, optimization, genetic algorithms and martial arts: Toward memetic algorithms* (Technical Report No. 826). California Institute of Technology, Pasadena, CA.
- Osman, I. H. and Laporte, G. (1996). *Metaheuristics: A bibliography*. *Annals of Operations Research*, 63(1), 513–628.

- Peres, F. and Castelli, M. (2021). *Combinatorial optimization problems and metaheuristics: Review, challenges, design, and development*. Applied Sciences, 11(14), 6449.
- Rajwar, K., Deep, K., and Das, S. (2023). *An exhaustive review of the metaheuristic algorithms for search and optimization: Taxonomy, applications, and open challenges*. Artificial Intelligence Review, 56(12), 13187–13257.
- Rego, C., and Glover, F. (2002). Local search and metaheuristics. In G. Gutin and A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 309– 368). Kluwer Academic Publishers.
- Reinelt, G. (1991). *The Traveling Salesman: Computational Solutions for GSP Applications*. Springer.
- Rutanen, K., Gómez-Herrero, G., Eriksson, S.-L. and Egiazarian, K. (2020). A general definition of the big-oh notation for algorithm analysis. Tampere University of Technology.
- Samanta, S., Nandi, D., and De, A. (2021). *A comparison of GA crossover and mutation methods for the Traveling Salesman Problem*. Published manuscript.
- Schrijver, A. (2005). On the history of combinatorial optimization (till 1960). G. N. K. Aardal ve R. Weismantel (Ed.), *Discrete Optimization içinde* (s. 1–68). Elsevier.
- Singh, P. and Choudhary, S. K. (2021). Introduction: Optimization and metaheuristics algorithms. H. Malik, A. Iqbal, P. Joshi, S. Agrawal ve F. I. Bakhsh (Ed.), *Metaheuristic and evolutionary computation: Algorithms and applications* (s. 3–33). Springer.
- Slowik, A. and Kwasnicka, H. (2020). Evolutionary algorithms and their applications to engineering problems. *Neural Computing and Applications*, 32(16), 12363–12379.
- Spielman, D. A. (2009). Spectral graph theory lecture 25: Planar graphs, part 1 [Lecture notes]. Yale University.
<https://www.cs.yale.edu/homes/spielman/561/2009/lect25-09.pdf>
- Stegherr, A., Senger, F. and Müller, H. (2022). Flexible classifications of metaheuristics for combinatorial optimization problems. *Computers and Operations Research*, 142, 105689.
- Stegherr, H., Heider, M. and Hähner, J. (2022). Classifying metaheuristics: Towards a unified multi-level classification system. *Natural Computing*, 21(1), 155–171.
- Štencek, J. (2013). *Traveling salesman problem* [Yayımlanmamış lisans tezi]. JAMK University of Applied Sciences, Finlandiya.

- Süral, H. (2003). Gezgin satıcı problemi (Traveling Salesman Problem). *Matematik Dünyası*, 2(3), 37–40.
- Toaza, B., Caicedo Bravo, E., Bejarano, A., Roa, L. M., and Pérez, J. (2023). A review of metaheuristic algorithms for solving TSP-based optimization problems. *Applied Soft Computing*, 150, 111006.
- Tonguç, G. (2024). Optimizasyon algoritmalarına genel bakış: Klasik ve modern yöntemler. *Kuantum Teknolojileri ve Enformatik Araştırmaları*, 2(3), 105-113.
- Whitley, D., Starkweather, T., and Fuquay, D. (1989). Scheduling problems and travelling salesmen: *The genetic edge recombination operator*. *Proceedings of the 3rd International Conference on Genetic Algorithms*, 133–140.
- Winston, W. L. and Goldberg, J. B. (2004). *Operations research: Applications and algorithms*. Thomson Brooks/Cole.
- Xu, Q., Xia, K. and Chu, X. (2025). Discrete Differentiated Creative Search for traveling salesman problem. *Applied Soft Computing*, 174, 112998.
- Zambito, L. (2006). The Traveling Salesman Problem: A Comprehensive Survey (Unpublished course project, CSE 4080). York University.

EKLER

EK-1: Küçük Ölçekli Problemler İçin İstatistiksel Analiz Sonuçları

Veri Seti	P.Seti	Metrik	Ortalama	Std.Sapma	En İyi	Medyan	En Kötü	95% GA Düşük	95% GA Yüksek
ulysses22	Hız	En İyi	7013	0	7013	7013	7013	7013	7013
		Süre(dk)	0,0069	0,0005	0,0064	0,0066	0,0079	0,0066	0,0071
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	7013	0	7013	7013	7013	7013	7013
		Süre(dk)	0,0080	0,0008	0,0074	0,0077	0,0101	0,0075	0,0086
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	7013	0	7013	7013	7013	7013	7013
		Süre(dk)	0,0071	0,0003	0,0068	0,0069	0,0080	0,0068	0,0073
		Sapma (%)	0	0	0	0	0	0	0
dantzig42	Hız	En İyi	699	0	699	699	699	699	699
		Süre(dk)	0,0082	0,0003	0,0079	0,0081	0,0087	0,0080	0,0084
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	699	0	699	699	699	699	699
		Süre(dk)	0,0121	0,0006	0,0113	0,0118	0,0132	0,0117	0,0124
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	699	0	699	699	699	699	699
		Süre(dk)	0,0093	0,0016	0,0080	0,0084	0,0116	0,0083	0,0102
		Sapma (%)	0	0	0	0	0	0	0
att48	Hız	En İyi	10628	0	10628	10628	10628	10628	10628
		Süre(dk)	0,0128	0,0035	0,0089	0,0122	0,0187	0,0107	0,0150
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	10628	0	10628	10628	10628	10628	10628
		Süre(dk)	0,0175	0,0040	0,0129	0,0171	0,0222	0,0150	0,0200
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	10628	0	10628	10628	10628	10628	10628
		Süre(dk)	0,0151	0,0054	0,0086	0,0149	0,0221	0,0117	0,0184
		Sapma (%)	0	0	0	0	0	0	0

Veri Seti	P.Seti	Metrik	Ort	SS	En İyi	Medyan	En Kötü	95% GA Düşük	95% GA Yüksek
berlin52	Hız	En İyi	7542	0	7542	7542	7542	7542	7542
		Süre(dk)	0,0100	0,0012	0,0092	0,0097	0,0132	0,0093	0,0107
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	7542	0	7542	7542	7542	7542	7542
		Süre(dk)	0,0149	0,0006	0,0141	0,0148	0,0156	0,0146	0,0153
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	7542	0	7542	7542	7542	7542	7542
		Süre(dk)	0,0104	0,0018	0,0093	0,0099	0,0155	0,0093	0,0115
		Sapma (%)	0	0	0	0	0	0	0
eil76	Hız	En İyi	538	0	538	538	538	538	538
		Süre(dk)	0,0326	0,0105	0,0194	0,0341	0,0533	0,0260	0,0391
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	538	0	538	538	538	538	538
		Süre(dk)	0,0494	0,0131	0,0368	0,0421	0,0779	0,0413	0,0576
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	538	0	538	538	538	538	538
		Süre(dk)	0,0541	0,0181	0,0309	0,0551	0,0897	0,0429	0,0653
		Sapma (%)	0	0	0	0	0	0	0
kroA100	Hız	En İyi	21282	0	21282	21282	21282	21282	21282
		Süre(dk)	0,0576	0,0162	0,0248	0,0601	0,0832	0,0475	0,0676
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	21282	0	21282	21282	21282	21282	21282
		Süre(dk)	0,1021	0,0224	0,0583	0,0937	0,1282	0,0882	0,1160
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	21282	0	21282	21282	21282	21282	21282
		Süre(dk)	0,0862	0,0225	0,0507	0,0883	0,1155	0,0722	0,1001
		Sapma (%)	0	0	0	0	0	0	0

EK-2 : Orta Ölçekli Problemler İçin İstatistiksel Analiz Sonuçları

Veri Seti	P.S eti	Metrik	Ortalama	Std.Sapma	En İyi	Medyan	En Kötü	95% GA Düşük	95% GA Yüksek
gr137	Hız	En İyi	69853	0	69853	69853	69853	69853	69853
		Süre(dk)	0,0695	0,0376	0,0370	0,0532	0,1489	0,0462	0,0927
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	69853	0	69853	69853	69853	69853	69853
		Süre(dk)	0,0835	0,0343	0,0549	0,0698	0,1682	0,0622	0,1048
		Sapma (%)	0	0	0	0	0	0	0
	Kali te	En İyi	69853	0	69853	69853	69853	69853	69853
		Süre(dk)	0,1055	0,0195	0,0703	0,1098	0,1373	0,0934	0,1176
		Sapma (%)	0	0	0	0	0	0	0
brg180	Hız	En İyi	1950	0	1950	1950	1950	1950	1950
		Süre(dk)	0,0184	0,0163	0,0059	0,0070	0,0467	0,0083	0,0285
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	1950	0	1950	1950	1950	1950	1950
		Süre(dk)	0,0208	0,0179	0,0100	0,0109	0,0655	0,0097	0,0319
		Sapma (%)	0	0	0	0	0	0	0
	Kali te	En İyi	1950	0	1950	1950	1950	1950	1950
		Süre(dk)	0,0319	0,0115	0,0241	0,0267	0,0565	0,0248	0,0390
		Sapma (%)	0	0	0	0	0	0	0
kroA200	Hız	En İyi	29374	7,2296	29368	29368	29382	29369, 11907	29378, 08093
		Süre(dk)	0,6881	0,4495	0,0800	0,5590	1,2591	0,4095	0,9667
		Sapma (%)	0,0191	0,0246	0	0	0,0477	0,00381	0,03432
	Denge	En İyi	29369	4,4272	29368	29368	29382	29366, 6	29372, 1
		Süre(dk)	0,7778	0,6531	0,1047	0,5754	2,0772	0,3730	1,1827
		Sapma (%)	0,0048	0,0151	0	0	0,0477	-0,0045	0,0141
	Kali te	En İyi	29368	0	29368	29368	29368	29368	29368
		Süre(dk)	0,7034	0,2716	0,2655	0,7634	1,1679	0,5351	0,8718
		Sapma (%)	0	0	0	0	0	0	0

Veri Seti	P.S eti	Metrik	Ortalama	Std.Sapma	En İyi	Medyan	En Kötü	95% GA Düşük	95% GA Yüksek
ts225	Hız	En İyi	126643	0	126643	126643	126643	126643	126643
		Süre(dk)	0,1665	0,0430	0,1091	0,1773	0,2346	0,1398	0,1932
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	126643	0	126643	126643	126643	126643	126643
		Süre(dk)	0,1665	0,0430	0,1091	0,1773	0,2346	0,1398	0,1932
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	126643	0	126643	126643	126643	126643	126643
		Süre(dk)	0,4901	0,1249	0,2900	0,4789	0,6698	0,4127	0,5675
		Sapma (%)	0	0	0	0	0	0	0
a280	Hız	En İyi	2579	0	2579	2579	2579	2579	2579
		Süre(dk)	0,3077	0,1363	0,1509	0,2837	0,5198	0,2232	0,3922
		Sapma (%)	0	0	0	0	0	0	0
	Denge	En İyi	2579	0	2579	2579	2579	2579	2579
		Süre(dk)	0,8686	0,5958	0,2790	0,7840	2,2842	0,4993	1,2378
		Sapma (%)	0	0	0	0	0	0	0
	Kalite	En İyi	2579	0	2579	2579	2579	2579	2579
		Süre(dk)	1,3140	0,6524	0,6809	0,9863	2,3298	0,9097	1,7184
		Sapma (%)	0	0	0	0	0	0	0
fl417	Hız	En İyi	12040	107,96	11882	11988,5	12182	11973,184	12107,015
		Süre(dk)	2,6615	0,1482	2,4663	2,6886	2,8571	2,5696	2,7533
		Sapma (%)	1,51	0,9102	0,1771	1,07495	2,7063	0,9458242	2,0741571
	Denge	En İyi	11969	104	11866	11946,5	12160	11904,338	12033,261
		Süre(dk)	4,2667	0,1775	3,8434	4,3049	4,5068	4,1567	4,3767
		Sapma (%)	0,9089	0,8768	0,0422	0,72085	2,5209	0,365384	1,4523372
	Kalite	En İyi	11939	74,64	11878	11902,5	12063	11892,237	11984,762
		Süre(dk)	9,3390	0,2436	9,0456	9,3788	9,7189	9,1880	9,4900
		Sapma (%)	0,6534	0,6293	0,1433	0,34989	1,7031	0,2633630	1,0434407

EK-3 : Büyük Ölçekli Problemler İçin İstatistiksel Analiz Sonuçları

Veri Seti	P.Seti	Metrik	Ortalama	Std.Sapma	En İyi	Medyan	En Kötü	95% GA Düşük	95% GA Yüksek
att532	Hız	En İyi	28569,5	166,98	28333	28551	28977	28466,002	28672,997
		Süre(dk)	3,9681	0,9069	2,9893	3,6329	5,3287	3,4060	4,5303
		Sapma (%)	3,1911	0,6031	2,3369	3,1243	4,6630	2,8173	3,5649
	Denge	En İyi	28622,4	177,37	28310	28658	28958	28512,466	28732,333
		Süre(dk)	5,8039	0,2020	5,3637	5,8746	5,9882	5,6787	5,9292
		Sapma (%)	3,3822	0,6406	2,2538	3,5108	4,5943	2,9851	3,7792
	Kalite	En İyi	28537,4	217,33	28259	28483	28951	28402,695	28672,104
		Süre(dk)	9,3396	0,6614	8,7245	9,1362	10,6813	8,9297	9,7496
		Sapma (%)	3,0752	0,785	2,0696	2,8787	4,5690	2,5886	3,5617
gr666	Hız	En İyi	309740	3128	306107	308893	316124	307801,22	311678,77
		Süre(dk)	6,3354	0,7725	4,6483	6,5398	7,3655	5,8566	6,8142
		Sapma (%)	5,2256	1,0627	3,9913	4,9378	7,3943	4,5669	5,8842
	Denge	En İyi	309260,4	3250,1	303354	309355	314196	307245,98	311274,81
		Süre(dk)	9,6847	0,6613	8,2159	9,7250	10,7029	9,2748	10,0946
		Sapma (%)	5,0626	1,1041	3,0561	5,0948	6,7394	4,3783	5,7470
	Kalite	En İyi	308424,5	2568,3	305430	308282	313880	306832,64	310016,35
		Süre(dk)	15,6066	0,6268	14,5690	15,4879	16,6525	15,2181	15,9951
		Sapma (%)	4,7787	0,8725	3,7614	4,73029	6,6320	4,2379	5,3194

Veri Seti	P.Seti	Metrik	Ortalama	Std.Sapma	En İyi	Medyan	En Kötü	95% GA Düşük	95% GA Yüksek
u724	Hız	En İyi	43611,5	326,64	43184	43621	44102	43377,85	43845,15
		Süre(dk)	6,2604	0,2301	5,7027	6,3140	6,5870	6,0958	6,4251
		Sapma (%)	4,0650	0,7369	3,0938	4,0839	5,2320	3,5379	4,5921
	Denge	En İyi	43419,1	217,16	43179	43378	43779	43284,502	43553,697
		Süre(dk)	9,6401	0,4237	8,4863	9,7738	9,9215	9,3775	9,9027
		Sapma (%)	3,6008	0,5182	3,0279	3,5027	4,4595	3,2796	3,9219
	Kalite	En İyi	43427,47	176,6	43258	43421	43588	43336,9	43518,04
		Süre(dk)	15,7500	2,1700	13,7900	15,9100	20,6800	14,4100	17,0900
		Sapma (%)	3,99	0,61	2,36	4,01	4,7	3,63	4,35
rat783	Hız	En İyi	9211,4	37,212	9135	9219	9259	9188,3359	9234,4640
		Süre(dk)	9,1764	0,8876	7,3642	9,4135	10,0460	8,6263	9,7266
		Sapma (%)	4,6036	0,4226	3,7360	4,6899	5,1442	4,3417	4,8655
	Denge	En İyi	9190,9	46,035	9115	9198	9274	9157,9686	9223,8313
		Süre(dk)	12,3354	1,9728	9,8547	12,0768	15,0348	10,9241	13,7467
		Sapma (%)	4,3708	0,5228	3,5089	4,4515	5,3145	3,9969	4,7448
	Kalite	En İyi	9186,5	54,235	9099	9194,5	9246	9147,7029	9225,2970
		Süre(dk)	19,2381	2,5092	16,3580	19,3246	22,6358	17,4431	21,0330

ÖZGEÇMİŞ

Adı Soyadı : Kardelen Bahar Erden

Yabancı Dil : İngilizce

ORCID No. : 0009-0008-7766-3474

Eğitim ve Mesleki Geçmişi:

- 2016-2018, Akdeniz Üniversitesi, Teknik Bilimler MYO, Bilgisayar Programcılığı
- 2018-2022, Eskişehir Osmangazi Üniversitesi, Mühendislik-Mimarlık Fakültesi, Endüstri Mühendisliği
- 2022-Halen, Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Endüstri Mühendisliği, Endüstri Mühendisliği Anabilim Dalı