



T.C.
NECMETTİN ERBAKAN
ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



**İKİLİ GRİ KURT OPTİMİZASYON
ALGORİTMASININ İKİLİ OPTİMİZASYON
PROBLEMLERİNE UYGULANMASI**

Feyza ERDOĞAN

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

**Haziran-2023
KONYA
Her Hakkı Saklıdır**

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Feyza ERDOĞAN

Tarih: 06.06.2023

ÖZET**YÜKSEK LİSANS TEZİ****İKİLİ GRİ KURT OPTİMİZASYON ALGORİTMASININ İKİLİ
OPTİMİZASYON PROBLEMLERİNE UYGULANMASI****Feyza ERDOĞAN****Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı****Danışman: Doç. Dr. Şaban GÜLCÜ****2023, 91 Sayfa****Jüri****Doç. Dr. Şaban GÜLCÜ****Prof. Dr. Halife KODAZ****Dr. Öğr. Üyesi Murat KARAKOYUN**

Metasezgisel algoritmalar, optimizasyon problemlerinin çözümü için önerilen ve sıklıkla kullanılan yöntemlerdir. Günümüzde birçok zorlu probleme uyarlanmış ve başarıları tespit edilmiştir. Gri Kurt Optimizasyon Algoritması (GWO), yeni nesil metasezgisel yöntemlerden biridir. GWO, avantajlarından dolayı ortaya çıktığı günden bu yana pek çok farklı sorunun çözümü için önerilmiş ve başarısını kanıtlamıştır. Bu tez çalışmasında, ikili optimizasyon problemlerinin çözümü için GWO'nun yeni bir varyantı olan İkili Dinamik Gri Kurt Optimizasyon Algoritması (IDGKO) önerilmiştir. IDGKO'nun diğer ikili GWO yöntemlerinden farkı, sürekli arama uzayını ikili arama uzayına dönüştürmek için XOR lojik-kapı tabanlı operatörü kullanması ve orijinal GWO'daki üç baskın bireyin (alfa, beta ve delta) çözüm kalitesi üzerindeki etkisini belirlemek için geliştirilen dinamik katsayı yöntemine dayalı olmasıdır. IDGKO'nun, ikili optimizasyon problemlerini çözmede yerel arama ile küresel arama arasında bir denge kuran basit, uygulanabilir ve başarılı bir yöntem olması amaçlanmıştır. Önerilen IDGKO'nun başarısını ve doğruluğunu belirlemek için literatürde adından sıkça söz ettiren özellik seçimi (FS) ve NP-Hard problemlerin bir üyesi olan 0-1 sırt çantası problemi (0-1 KP) üzerinde testler gerçekleştirilmiştir. FS, veri madenciliğinde temel ön işleme adımlarından biri olup zorlu ikili optimizasyon problemleri arasında yer almaktadır. FS, performansı ve doğruluğu etkilemeden belirli bir veri kümesinden çok az etkisi olan özellikleri kaldırarak veri kümesini en iyi temsil edebilecek alt kümeyi belirleme işlemidir. 0-1 KP birçok farklı alanda etkin bir role sahip olup temel amacı, yüksek kar ve düşük maliyetle kaynakların verimli bir şekilde kullanılmasıdır. Önerilen yöntemin performansı, ilgili problem için literatüre kazandırılmış yakın tarihli ikili GWO algoritmaları ve farklı metasezgisel yöntemler dahil olmak üzere birçok algoritmanın performansı ile karşılaştırılmıştır. Deneylerde algoritmanın etkinliği ve tutarlılığını tespit etmek adına, farklı boyutta ve özellikte birçok veri seti kullanılmıştır. Yapılan kıyaslamalar ve istatistiksel testler sonucunda, önerilen IDGKO'nun amacına uygun olarak etkili ve başarılı bir yöntem olduğu kanıtlanmıştır.

Anahtar Kelimeler: 0-1 Sırt Çantası Problemi, Gri Kurt Optimizasyon Algoritması, İkili Optimizasyon, Lojik-Kapı Tabanlı Operatörler, Metasezgisel Algoritmalar, Optimizasyon, Özellik Seçimi, XOR Operatörü

ABSTRACT**MS THESIS****APPLICATION OF BINARY GREY WOLF OPTIMIZATION ALGORITHM
TO BINARY OPTIMIZATION PROBLEMS****Feyza ERDOĞAN****THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF
NECMETTİN ERBAKAN UNIVERSITY
THE DEGREE OF MASTER OF SCIENCE IN COMPUTER ENGINEERING****Advisor: Assoc. Prof. Dr. Şaban GÜLCÜ****2023, 91 Pages****Jury****Assoc. Prof. Dr. Şaban GÜLCÜ****Prof. Dr. Halife KODAZ****Asst. Prof. Dr. Murat KARAKOYUN**

Metaheuristic algorithms are recommended and frequently used methods for solving optimization problems. Today, it has been adapted to many challenging problems and its successes have been identified. Grey Wolf Optimization Algorithm (GWO) is one of the next generation metaheuristic methods. Due to its advantages, GWO has been proposed for the solution of many different problems since the day it emerged and has proven its success. In this thesis, a new variant of GWO, Binary Dynamic Grey Wolf Optimization Algorithm (IDGKO), is proposed for the solution of binary optimization problems. The difference of IDGKO from other binary GWO methods is that it uses the XOR logic-gate-based operator to convert the continuous search space to binary search space and is based on the dynamic coefficient method developed to determine the effect of the three dominant individuals (alpha, beta and delta) in the original GWO on the solution quality. that is. IDGKO is intended to be a simple, feasible and successful method that strikes a balance between local search and global search in solving binary optimization problems. In order to determine the success and accuracy of the proposed IDGKO, tests were carried out on the 0-1 backpack problem (0-1 KP), which is a member of the feature selection (FS) and NP-Hard problems, which are frequently mentioned in the literature. FS is one of the basic preprocessing steps in data mining and is among the difficult binary optimization problems. FS is the process of determining the subset that can best represent the dataset by removing features that have little impact from a given dataset without affecting performance and accuracy. 0-1 KP has an active role in many different areas and its main purpose is to use resources efficiently with high profit and low cost. The performance of the proposed method has been compared with the performance of many algorithms, including recent binary GWO algorithms and different metaheuristic methods for the related problem. In the experiments, many data sets of different sizes and features were used to determine the efficiency and consistency of the algorithm. As a result of the comparisons and statistical tests, the proposed IDGKO has proven to be an effective and successful method in line with its purpose

Keywords: 0-1 Knapsack Problem, Binary Optimization, Feature Selection, Grey Wolf Optimizer, Logic-Gate Based Operator, Metaheuristic Algorithm, Optimization, XOR Operator

ÖNSÖZ

Yüksek lisans sürecim boyunca gösterdiği sabır ve değerli bilgi paylaşımları için danışman hocam Doç. Dr. Şaban GÜLCÜ'ye,

Tez konumu belirleme sürecinde ve tez yazım aşamasında desteğini ve yardımlarını bir kez olsun esirgemeyerek bana her daim yol gösteren kıymetli hocam Dr. Öğr. Üyesi Murat KARAKOYUN'a,

Yüksek lisans ders sürecimde ve akademik anlamda bilgi ve tecrübeleriyle bana ilham kaynağı olan Doç. Dr. Mehmet HACIBEYOĞLU, Doç. Dr. Hüseyin HAKLI, Dr. Öğr. Üyesi Ayşe Merve ACILAR ve Arş. Gör. Ferda Nur ARICI hocalarıma,

Tez sürecindeki destekleri ve anlayışı için kıymetli hocam Dr. Öğr. Üyesi Vahit TONGUR'a teşekkürü bir borç bilirim.

Eğitim hayatım boyunca yanımda olan, motivasyon kaynağım ve dayanak noktam annem, babam ve kardeşlerime en kalbi şükranlarımı sunarım.

Feyza ERDOĞAN
KONYA-2023

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR	viii
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMASI	10
2.1. Özellik Seçimi Üzerine Yapılan Çalışmalar	10
2.2. 0-1 Sırt Çantası Problemi Üzerine Yapılan Çalışmalar	20
3. MATERYAL VE YÖNTEM.....	26
3.1. Özellik Seçimi Problemi	27
3.2. Özellik Seçimi Problemi için Deneylerde Kullanılan Veri Setleri	31
3.3. Özellik Seçimi için Uygunluk Fonksiyonu.....	32
3.4. Özellik Seçimi Problemi için Performans Belirleme Ölçütleri	33
3.4.1. Sınıflandırma Doğruluğu	33
3.4.2. Uygunluk Fonksiyonu Değerleri	33
3.5. K- En Yakın Komşu Algoritması	34
3.6. 0-1 Sırt Çantası Problemi.....	35
3.6.1. Onarım Fonksiyonu	37
3.7. Gri Kurt Optimizasyon Algoritması	38
4. ÖNERİLEN YÖNTEM	44
4.1. İkili Dinamik Gri Kurt Optimizasyon Algoritması.....	44
5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....	52
5.1. FS için IDGKO'nun Araştırma Sonuçları	52
5.1.1. Deneysel Çalışma 1	52
5.1.2. Deneysel Çalışma 2	55
5.2. 0-1 KP için IDGKO'nun Araştırma Sonuçları	58
5.2.1. Deneysel Çalışma 1	58
5.2.2. Deneysel Çalışma 2	63
5.2.3. Deneysel Çalışma 3	72
6. SONUÇLAR VE ÖNERİLER	76
7. KAYNAKLAR	79

SİMGELER VE KISALTMALAR

Simgeler

α	: GWO algoritmasındaki en iyi çözüme sahip birey (alpha wolf)
β	: GWO algoritmasındaki ikinci en iyi çözüme sahip birey (beta wolf)
δ	: GWO algoritmasındaki üçüncü en iyi çözüme sahip birey (delta wolf)
ω	: GWO algoritmasında baskın bireyler dışındaki diğer bireyler (omega wolf)
\oplus	: XOR lojik-kapı tabanlı operatörü
$ECR_{\alpha}, ECR_{\beta}, ECR_{\delta}$: Sırasıyla alfa, beta ve delta bireyin çözüm kalitesi üzerindeki etki katsayı oranı
$NDU_{\alpha}, NDU_{\beta}, NDU_{\delta}$: Sırasıyla alfa, beta ve delta bireyin yeni aday çözüm için güncelleyeceği boyutların sayısı
w	: 0-1 sırt çantası problemindeki nesnelerin ağırlık değerlerini temsil eden parametre
p	: 0-1 sırt çantası problemindeki nesnelerin kâr (fayda) değerlerini temsil eden parametre
W	: 0-1 sırt çantası problemindeki sırt çantası kapasitesini temsil eden parametre

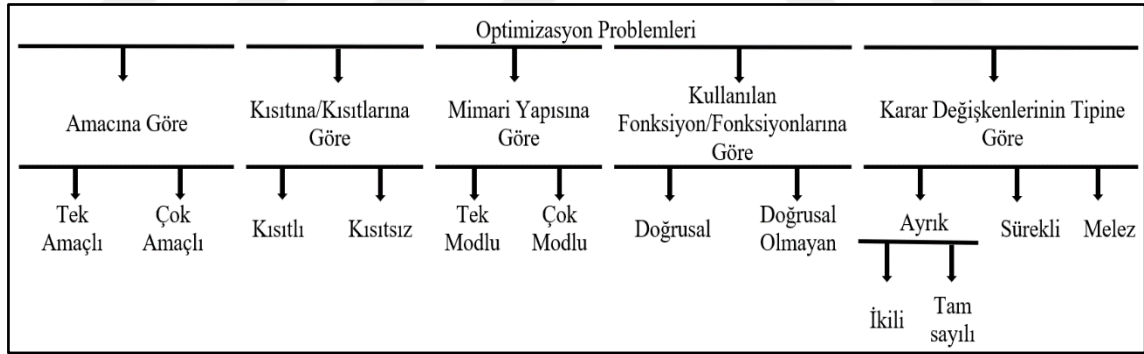
Kısaltmalar

GWO	: Gri Kurt Optimizasyonu (<i>Grey Wolf Optimizer</i>)
IDGKO	: İkili Dinamik Gri Kurt Optimizasyonu (<i>Binary Dynamic Grey Wolf Optimizer</i>)
FS	: Özellik Seçimi (<i>Feature Selection</i>)
0-1 KP	: 0-1 Sırt Çantası Problemi (<i>0-1 Knapsack Problem</i>)
RF	: Onarım Fonksiyonu (<i>Repair Function</i>)
NDCA	: Yeni Dinamik Katsayı Yöntemi
KNN	: K- En Yakın Komşu Algoritması (<i>K-Nearest Neighbors Algorithm</i>)
PSO	: Parçacık Sürü Optimizasyonu (<i>Particle Swarm Optimization</i>)
GA	: Genetik Algoritma (<i>Genetic Algorithm</i>)

1. GİRİŞ

Optimizasyon; bir problemin, belirlenen kısıtlara sadık kalarak, çözümünde kullanılabilir tüm olası çözümler arasından en iyi çözümü elde etme sürecidir. Günümüzde birçok farklı disiplin optimizasyon ile yakından ilgilidir. Çünkü disiplinlerde yer alan birçok problem optimizasyon problemi kapsamına girmektedir ve çözümü için optimizasyon tekniklerine başvurulmaktadır. Biyoloji (Huang, 2012; Shortt ve ark., 2022), finans (Firouzi ve Meshkani, 2021; Bae ve ark., 2022; Chang ve ark., 2022), fizik (Abedinpourshotorban ve ark., 2016; Hashim ve ark., 2019), kimya (Bezerra ve ark., 2008; Naveenkumar ve Baskar, 2021), mühendislik (Bekdaş ve ark., 2019; Agushaka ve Ezugwu, 2021; Yong ve ark., 2021; Subbaih ve Ramasamy, 2022), tıp (Han ve ark., 2019; Shang ve ark., 2022; Zhou ve ark., 2022), üretim ve pazarlama (Krogh ve ark., 2021; Wang ve ark., 2021a) gibi birçok disiplin alanına özgü problemlerin çözümü için optimizasyon teknikleri başarılı bir şekilde kullanılmaktadır.

Optimizasyon problemleri farklı karakteristik özelliklere sahip olabilmektedir. Bu sebeple problemler kendi içerisinde alt sınıflara ayrılmaktadır (Yang, 2010b; Korkmaz, 2019). Şekil 1.1’de optimizasyon problemlerine ait sınıflandırma şeması verilmiştir.



Şekil 1.1. Optimizasyon problemlerine ait sınıflandırma şeması (Korkmaz, 2019)

Şekil 1.1’de görüldüğü üzere optimizasyon problemleri beş ana kategoride değerlendirilmektedir. Optimizasyon problemleri amacına göre tek amaçlı ve çok amaçlı olmak üzere iki sınıfa ayrılmaktadır. Tek amaçlı optimizasyon problemlerinde yalnızca tek bir amaca hizmet edilirken; çok amaçlı optimizasyon problemlerinde birden fazla amaca hizmet edilmektedir. Mühendislik problemleri gibi zorlu birçok problemde birden fazla amaç fonksiyonu yer almakta ve çok amaçlı optimizasyon problemleri kapsamına girmektedir (Özkış, 2018; Korkmaz, 2019).

Optimizasyon problemleri kısıtına göre değerlendirildiğinde iki sınıfa ayrılmaktadır. Probleme ait karar değişkenleri için herhangi bir kısıtlama bulunmuyorsa kısıtsız optimizasyon problemi; bir veya birden çok kısıtlama bulunursa kısıtlı optimizasyon problemi olarak adlandırılmaktadır (Yang, 2010b).

Optimizasyon problemleri mimari yapısına göre tek modlu ve çok modlu olmak üzere iki ayrı sınıfa ayrılmaktadır. Tek modlu optimizasyon problemlerinin grafiksel mimarisine bakıldığında benzersiz global optimum için tek bir tepe veya çukur noktası bulunmaktadır. Çok modlu optimizasyon problemleri ise adından anlaşılacağı üzere birden fazla tepe ve çukur noktası barındırmaktadır. Bu tür problemlerin grafik olarak yorumlanması şu şekildedir: en yüksek tepe veya en alçak çukur noktası global optimum değerini; bunlar dışında kalan tepe veya çukur noktaları ise yerel optimum değerlerini göstermektedir (Çelik, 2013; Korkmaz, 2019).

Kullanılan fonksiyon/fonksiyonların yapısına göre optimizasyon problemleri iki ayrı başlıkta incelenmektedir. Doğrusal optimizasyon problemlerinde, problem için kullanılan amaç fonksiyonları ve kısıt fonksiyonlarının tümü doğrusal yapıdadır. Bunun aksine bahsi geçen kısıt ve amaç fonksiyonlarından herhangi biri bile doğrusal yapıda değil ise doğrusal olmayan optimizasyon problemi olarak adlandırılmaktadır (Karaboğa, 2014; Korkmaz, 2019).

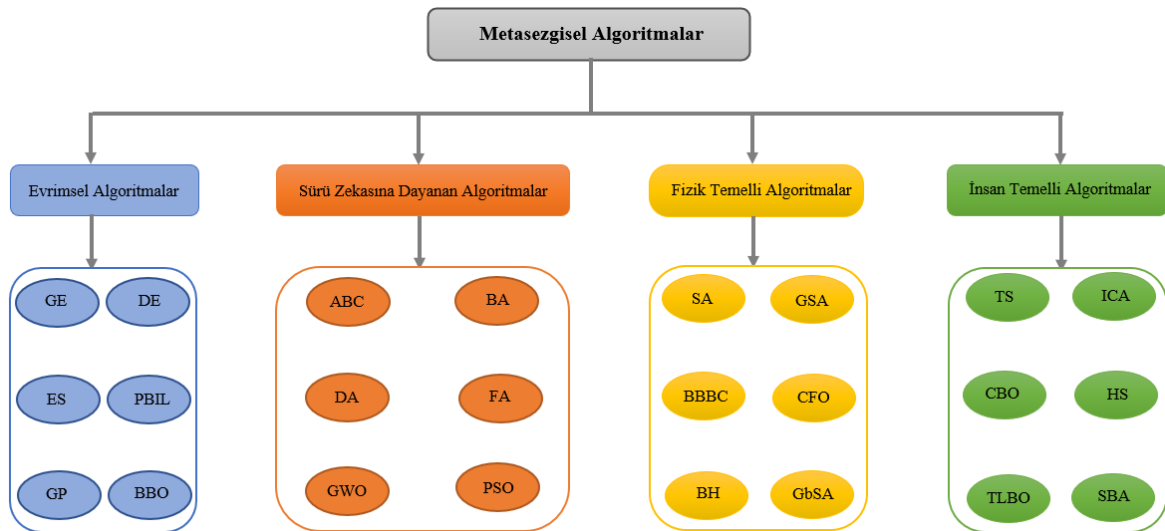
Optimizasyon problemleri karar değişkenleri tipine göre değerlendirildiğinde sürekli, ayrık ve melez olmak üzere üç ayrı sınıfa ayrılmaktadır. Sürekli optimizasyon problemlerinde, probleme ait karar değişkenleri gerçek değerlerden oluşmaktadır. Ayrık optimizasyon problemlerinde karar değişkenleri belirlenen bir küme içerisindeki belirli değerlerden oluşmaktadır. Melez optimizasyon problemleri ise karar değişkenleri hem sürekli hem ayrık değerlerden oluşabilen optimizasyon problemleridir (Akay, 2009; Karaboğa, 2014; Korkmaz, 2019).

Ayrık optimizasyon problemleri kendi içerisinde de iki başlıkta incelenmektedir. Tam sayılı optimizasyon problemlerinde karar değişkenleri tam sayı değerlerinden oluşurken; ikili optimizasyon problemlerinde karar değişkenleri yalnızca 0 veya 1 değerlerinden oluşmaktadır (Cinar ve Kiran, 2018). İkili optimizasyon, ayrık optimizasyon sınıfına önemli bir alt çalışma alanıdır. Günümüzde birçok problem yapısı gereği ikili optimizasyon problemleri sınıfına girmektedir. Kapasitesiz tesis yerleştirme problemleri, rüzgâr tribün yerleşimi problemleri, özellik seçimi, sırt çantası problemleri, birim yükleme problemleri ve çizelgeleme problemleri önemli bazı ikili optimizasyon problemlerindedir (Banitalebi ve ark., 2016; Hakli, 2020).

Gerçek dünya problemleri ve optimizasyon problemleri; uzun hesaplama süresi ihtiyacı, doğrusal olmayan kısıtlar, yüksek boyutlu arama uzayı, karmaşık ve zorlu yapıya sahip olma gibi temel nedenlerden dolayı çözümü zor problemlerdendir (Mirjalili ve Lewis, 2016; Abualigah ve ark., 2021a). Bu sebeple etkili çözümü en kısa sürede bulmayı hedefleyen, güvenilir ve yeni optimizasyon tekniklerine ihtiyaç duyulmaktadır. Bu zamana kadar birçok geleneksel yöntem önerilmiş ve problem çözümünde kullanılmıştır. Son dönemlerde adından sıkça söz ettiren ve her geçen gün yeni yöntemler önerilmeye devam eden metasezgisel yöntemler de zorlu optimizasyon problemlerinin çözümü için önerilen yöntemlerdendir (Mirjalili ve Lewis, 2016; Abualigah ve ark., 2021a). Özellikle mühendislik problemlerinde sıklıkla başvurulan metasezgisel yöntemler; basit kavram ve formüllerden oluşması, problemlere kolaylıkla uygulanabilmesi, gradyan bilgisi gerektirmemesi, yerel optimumdan kaçınmada başarılı olması ve farklı disiplinleri kapsayan çok çeşitli problemlerde bile etkili bir şekilde kullanılabilmesi gibi sebeplerden dolayı zorlu optimizasyon problemlerinin çözümünde önerilmektedir (Mirjalili ve Lewis, 2016).

Metasezgisel algoritmaları önemli ve başarılı kılan bir diğer etken, kullandığı arama stratejisidir. Bu algoritmalar keşif/çeşitlendirme (exploration/diversification) ve sömürü/yoğunlaştırmaya (exploitation/intensification) dayalı önemli bir arama stratejisine sahiptir (Abualigah ve Diabat, 2020; Abualigah ve ark., 2020a). Arama uzayını global keşfetme yeteneği keşif olarak adlandırılmaktadır ve bu sayede yerel optimumdan kaçınma noktasında önem teşkil etmektedir (Abualigah ve ark., 2021a). Sömürü ise çözüm kalitesine iyileştirmek amacıyla yakınlarda bulunan umut vadeden çözümleri yerel olarak keşfetme yeteneğidir (Abualigah, 2021). Bir metasezgisel algoritmanın başarısı için keşif ve sömürü arasında uyumlu ve etkili denge ayarına bağlıdır (Sadollah ve ark., 2018; Faramarzi ve ark., 2020; Gholizadeh ve ark., 2020).

Metasezgisel algoritmalar, oluşumunda ilham aldığı prensibe göre farklı sınıflardan oluşmaktadır (Mirjalili ve Lewis, 2016). Şekil 1.2'de metasezgisel algoritmaların sınıflandırılma şeması verilmiştir.



Şekil 1.2. Metasezgisel algoritmaların sınıflandırılması (Mirjalili ve Lewis, 2016)

Metasezgisel algoritmalar; evrimsel algoritmalar, sürü zekasına dayanan algoritmalar, fizik temelli algoritmalar ve insan temelli algoritmalar şeklinde dört başlık altında sınıflandırılabilir (Mirjalili ve Lewis, 2016; Kallioras ve ark., 2018; Abualigah ve ark., 2020b; Abualigah ve ark., 2021a).

Evrimsel algoritmalar, doğal evrim yasalarından ilham alan ve en iyi bireylerin bir araya gelerek yeni nesilleri oluşturmasına dayanan algoritmalarlardır. Bu güçlü yanı sayesinde nesiller boyunca popülasyondaki bireylerin sürekli optimize edimesini sağlamaktadır (Mirjalili ve Lewis, 2016). Ayrıca mutasyon, çaprazlama, elitizm ve doğal seçim gibi biyolojik davranışları benimseyen operatörler kullanılmaktadır. Evrimsel algoritmalar deyince akla ilk gelen Darwin Evrimi'ni temel alarak geliştirilen Genetik Algoritmalar (GA)'dır (Holland, 1992). Diferansiyel Evrim (Differential Evolution, DE) (Storn ve Price, 1997), Evrim Stratejisi (Evolution Strategy, ES) (Rechenberg, 1978), Olasılığa Dayalı Artan Zihinsel Öğrenme (Probability-Based Incremental Learning, PBIL) (Dasgupta ve Michalewicz, 2013), Genetik Programlama (Genetic Programming, GP) (Koza, 1992) ve Biyocoğrafya Tabanlı Optimize Edici (Biogeography-Based Optimizer, BBO) (Simon, 2008) evrimsel algoritmalar sınıfına ait birkaç önemli yöntemdir.

Fizik temelli algoritmalar, doğadaki fizik yasalarını ve kurallarını ilham alarak geliştirilmiş yöntemlerdir (Mirjalili ve Lewis, 2016). Simüle Edilmiş Tavlama (Simulated Annealing, SA) (Kirkpatrick ve ark., 1983; Černý, 1985), Yerçekimi Yerel Arama (Gravitational Local Search, GLSA) (Webster ve Bernhard, 2003), Big-Bang Big-Crunch (BBBC) (Erol ve Eksin, 2006), Yerçekimi Arama Algoritması (Gravitational Search

Algorithm, GSA) (Rashedi ve ark., 2009), Ücretli Sistem Arama (Charged System Search, CSS) (Kaveh ve Talatahari, 2010), Merkezi Kuvvet Optimizasyonu (Central Force Optimization, CFO) (Formato, 2007), Yapay Kimyasal Reaksiyon Optimizasyon Algoritması (Artificial Chemical Reaction Optimization Algorithm, ACROA) (Alatas, 2011), Kara Delik Algoritması (Black Hole, BH) (Hatamlou, 2013), Işın Optimizasyonu Algoritması (Ray Optimization, RO) (Kaveh ve Khayatazad, 2012), Küçük Dünya Optimizasyon Algoritması (Small-World Optimization Algorithm, SWOA) (Du ve ark., 2006), Galaksi Tabanlı Arama Algoritması (Galaxy-based Search Algorithm, GbSA) (Shah-Hosseini, 2011) ve Eğri Uzay Optimizasyonu (Curved Space Optimization, CSO) (Moghaddam ve ark., 2012) fizik temelli metasezgisel yöntemlerin bazı örnekleridir.

Sürü zekasına dayanan algoritmalar, doğadaki hayvanların sosyal davranışlarından, hareketlerinden ve avlanma tekniklerinden ilham alarak geliştirilmiş yöntemlerdir (Gandomi ve Alavi, 2012; Abualigah ve ark., 2021b). Sürü zekasına dayanan algoritmalarda, sürüdeki tüm bireyler bir aday çözümü temsil etmektedir ve genellikle boyut adı verilen problem boyutu kadar pozisyon uzunluğuna sahiptir. Bu algoritmaların temel özelliği, optimizasyon süreci boyunca sürüdeki tüm bireylerin sahip oldukları bilgileri sürünün geri kalanı ile sürekli paylaşıyor olmalarıdır (Abualigah ve ark., 2021a). Böylece global optimum değerine en yakın aday çözüme göre diğer bireylerde boyut değerlerinde güncellemeye giderek çözümü iyileştirebilecektir. Sürü zekasına dayanan algoritmalar, son dönemde oldukça revaçta olup her geçen gün sürü zekasına dayanan yeni metasezgisel algoritmalar önerilmektedir. Parçacık Sürü Optimizasyonu (Particle Swarm Optimization, PSO), Kennedy ve Eberhart tarafından önerilmiş oldukça popüler sürü zekasına dayanan bir metasezgisel algoritmadır (Kennedy ve Eberhart, 1995). PSO'nun evrimsel algoritmalara ve fizik tabanlı algoritmalara göre daha rekabetçi ve verimli olduğu kanıtlanmış ve bu sebeple sürü zekasına dayanan yöntemlere odaklanılmıştır (Mirjalili ve Lewis, 2016). Evrimsel algoritmalarda kullanılan operatörler (mutasyon, çaprazlama, doğal seçim, elitizm, vb.) sebebiyle sürü zekasına dayanan yöntemlere göre uygulanabilirliği daha zordur. Sürü zekasına dayanan yöntemler optimizasyon sürecinde ilerleyen iterasyonlar için popülasyon bilgilerini korumaktadır. Ancak evrimsel algoritmalarda yeni popülasyon oluşturulduğu an geçmişe yönelik bilgiler kaybolmaktadır (Mirjalili ve Lewis, 2016). Bunlar gibi temel farkların sonucunda optimizasyon problemlerinde sürü zekasına dayanan yöntemler diğerlerine oranla daha fazla tercih edilmektedir. Literatürde optimizasyon problemlerinin çözümü için önerilen birçok sürü zekasına dayanan metasezgisel algoritma bulunmaktadır. Yapay Arı Kolonisi

(Artificial Bee Colony, ABC) (Karaboga, 2005), Yarasa Optimizasyon Algoritması (Bat Algorithm, BA) (Yang, 2010a), Yusufçuk Optimizasyon Algoritması (Dragonfly Algorithm, DA) (Mirjalili, 2016), Ateş Böceği Optimizasyon Algoritması (Firefly Algorithm, FA) (Yang, 2009) ve Gri Kurt Optimizasyonu (Grey Wolf Optimizer, GWO) (Mirjalili ve ark., 2014b) sürü zekasına dayanan popüler bazı metasezgisel optimizasyon algoritmalarıdır.

İnsan temelli metasezgisel algoritmalar, toplumlardaki insan davranışlarından ve ilişkilerinden esinlenilerek tasarlanmış yöntemlerdir (Abualigah ve ark., 2021a). Bu sınıfa ait bazı örnek algoritmalar şunlardır: Öğretme Tabanlı Optimizasyon Algoritması (Teaching Learning Based Optimization, TLBO) (Rao ve ark., 2011; 2012), Uyum Arama (Harmony Search, HS) (Geem ve ark., 2001), Emperyalist Rekabet Algoritması (Imperialist Competitive Algorithm, ICA) (Atashpaz-Gargari ve Lucas, 2007), Tabu Arama (Tabu Search, TS) (Glover, 1989; 1990; Fogel, 1998), Borsa Piyasası Algoritması (Exchange Market Algorithm, EMA) (Ghorbani ve Babaei, 2014), Sosyal Tabanlı Algoritma (Social Based Algorithm, SBA) (Ramezani ve Lotfi, 2013), Çarpışan Gövdeler Optimizasyonu (Colliding Bodies Optimization, CBO) (Kaveh, 2014; Kaveh ve Mahdavi, 2014).

Literatüre kazandırılmış birçok optimizasyon algoritması temelinde sürekli optimizasyon problemlerinin çözümü için önerilmiştir. Sürekli değerler kabul eden optimizasyon problemlerinde karar değişkenleri, kısıt/kısıtlar olmadığı sürece belirlenen aralıkta sonsuz değerler almaktadır (Karaboğa, 2014). Ancak ayırık optimizasyon problemlerine bakıldığında karar değişkenleri, belirlenen aralıkta yalnızca belli değerleri ve hatta ikili optimizasyon problemlerinde yalnızca 0 veya 1 değerlerini alabilmektedir (Cinar ve Kiran, 2018). Dolayısıyla problemin karakteristiğine göre kullanılması muhtemel optimizasyon algoritmasında birtakım değişikliğe gidilmelidir (Hakli, 2020). Sürekli optimizasyon problemleri için önerilen optimizasyon algoritmalarını ikili optimizasyon problemlerine uyarlayabilmek için literatürde bazı yöntemler önerilmiştir. Bu yöntemlerin bir kısmı yalnızca probleme ait amaç fonksiyonuna odaklanmakta ve fonksiyon sonucu üretilirken etkin olmaktadır. Örneğin transfer fonksiyonu gibi yöntemler sürekli karar değişkenlerine sahip problemin amaç fonksiyonu çıktısını almadan devreye girerek ikili değerlere dönüşmesini sağlar ve böylece ikili optimizasyon problemlerinde kullanılabilir hale getirir. Bazı yöntemler ise doğrudan yöntemin çalışma prensibinde birtakım değişikliğe/değişikliklere gitme esasına dayanmaktadır (Korkmaz, 2019). Açık modülasyonu, benzerlik tabanlı yaklaşımlar, genetik operatörler, kuantum

ilhamlı bitler, lojik-kapı tabanlı ikili operatörler ve transfer fonksiyonları literatürde ikilileştirme adına sıklıkla başvurulan yöntemlerdendir (Banitalebi ve ark., 2016).

Açı modülasyonu (angle modulation), birleşik sinüs ve kosinüs fonksiyonlarını kullanarak ikili dizi oluşturma yöntemidir. Bu yöntem özellikle telekomünikasyon endüstrisinde sinyal işleme işlemlerinde kullanılmaktadır. Bu yöntemde kullanılan fonksiyon, sürekli değerlerden oluşan dört parametreye sahiptir. Bu parametre setlerinin her biri yalnızca bir adet ikili diziyi temsil etmektedir (Pampará ve Engelbrecht, 2011; Banitalebi ve ark., 2016).

Benzerlik tabanlı yaklaşımlar, farklı benzerlik indeksleri ile ikili dizilerin benzerliğini hesaplayarak bu benzerlik oranına göre yeni aday çözüm üretmeyi temel alan yöntemlerdir (Cinar ve Kiran, 2018; Korkmaz, 2019).

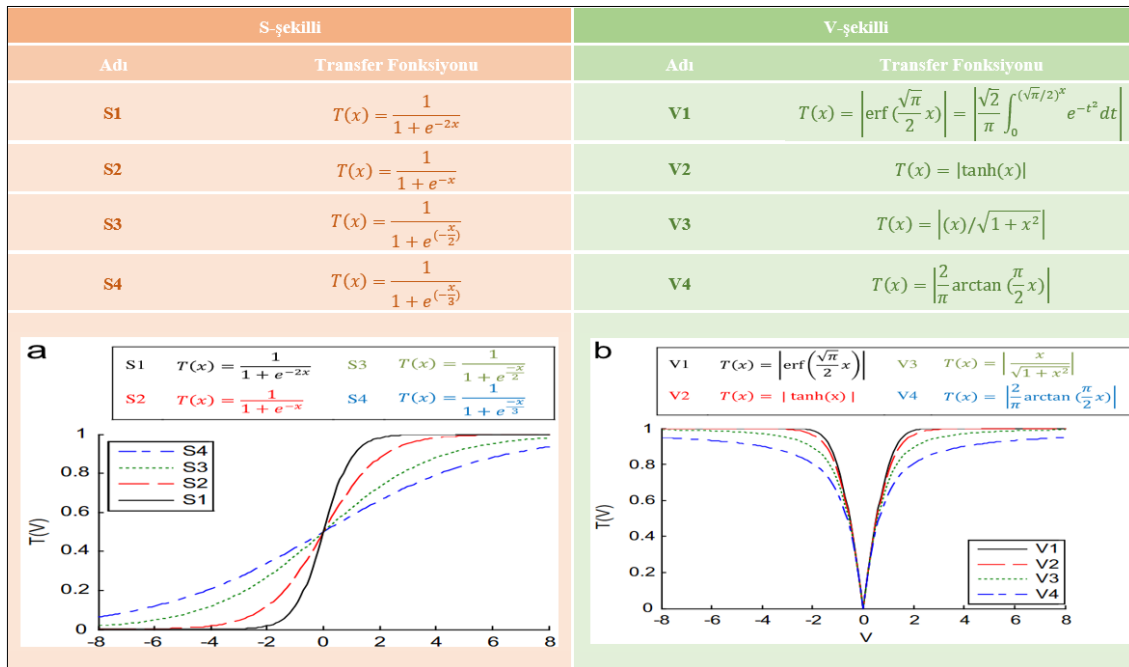
Evrimsel algoritmaların sahip olduğu çaprazlama, mutasyon, takas gibi genetik operatörler ikili bir formda arama uzayı ikili değerlerden oluşan ikili optimizasyon problemlerinde kullanılabilir (Pampará ve Engelbrecht, 2011; Banitalebi ve ark., 2016). Öztürk ve arkadaşları, Derviş Karaboğa tarafından önerilen ve literatürde başarıyla kullanılmakta olan Yapay Arı Kolonisi Algoritmasının etkin bir ikili varyantını oluştururken özel arama stratejilerini oluşturmada genetik operatörlerden faydalanmışlardır (Ozturk ve ark., 2015). Huang ve arkadaşları, 0-1 Sırt Çantası Probleminin çözümü için genetik operatörler kullanılarak Çok Ölçekli Kuantum Harmonik Osilatör Algoritmasının (Multi-Scale Quantum Harmonic Oscillator Algorithm, BMQHOA) ikili varyantını önermiştir.

Kuantum ilhamlı bitler, arama uzayındaki her bir ajana bir çift sayı değeri tanımlanmasına ve dönme açısının sürekli güncellenmesiyle oluşan rotasyon matrisi kullanarak aday çözümler üretilmesine dayanan kuantum hesaplama faaliyetlerinden esinlenerek geliştirilen bir tekniktir (Banitalebi ve ark., 2016; Dönmez, 2020).

Lojik kapı tabanlı ikili operatörler, ikili optimizasyon problemlerinin çözümü için ve (AND), veya (OR), değil (NOT), özel veya (XOR) gibi ikili değerler ile çalışmakta olan lojik kapı tabanlı yöntemlerdir (Dönmez, 2020). Yapılan çeşitli çalışmalarda lojik kapı tabanlı operatörlerin biri (Aslan ve ark., 2019) veya birkaçı (Cinar ve Kiran, 2018) kullanılarak ikili varyantlar oluşturulmuştur (Korkmaz, 2019).

Transfer fonksiyonları, sürekli değerleri ikili değerlere dönüştürmek için sıklıkla kullanılan bir ikilileştirme yöntemidir. İlk olarak PSO'yu ikilileştirmek için Kennedy ve Eberhart tarafından sigmoid transfer fonksiyonu kullanılmıştır (Kennedy ve Eberhart, 1997; Korkmaz, 2019). Bu çalışmanın ışığında yeni transfer fonksiyonları geliştirilmiş ve

literatüre kazandırılmıştır. Bazı transfer fonksiyon örnekleri Şekil 1.3’de verilmiştir (Mirjalili ve Lewis, 2013). Bu fonksiyonlar, grafiksel olarak aldığı şekle benzerliğinden S-şekilli ve V-şekilli transfer fonksiyonları olarak adlandırılmaktadır.



Şekil 1.3. S-şekilli (a) ve V-şekilli (b) transfer fonksiyon örnekleri (Mirjalili ve Lewis, 2013) (S2 Fonksiyonu (Geem ve ark., 2001), V2 Fonksiyonu (Ling ve ark., 2010))

İkilileştirme yöntemleri arasında en çok tercih edilen yöntem transfer fonksiyonlarıdır. Ancak transfer fonksiyonlarının kullanılması, uzaysal bağlantı kesilmesi (spatial disconnect), yalpalamalar (hamming cliffs), olası bir hassasiyet kaybı (a possible loss of precision) ve boyut laneti (curse of dimension) gibi bazı sorunlara neden olabilmekte ve her zaman verimli ve etkili sonuçlar garanti etmemektedir (Lanza-Gutierrez ve ark., 2017; García ve ark., 2019). Literatürdeki metasezgisel yöntemlerin çoğu sürekli optimizasyon problemleri için önerilmiştir ve başarıları büyük ölçüde parametrelerine bağlıdır (Lanza-Gutierrez ve ark., 2017). Bu parametreler, transfer fonksiyonlarıyla ikilileştirme sırasında değişerek yetersiz keşifle sonuçlanabilmektedir (García ve ark., 2019; Ozsoydan, 2019).

GWO, sürekli optimizasyon problemlerinin çözümü için doğadaki gri kurtların liderlik hiyerarşisinden ve sosyal davranışlarından esinlenerek tasarlanmış sürü zekasına dayanan metasezgisel bir optimizasyon algoritmasıdır (Mirjalili ve ark., 2014b). Günümüze kadar birçok problemin çözümünde önerilmiş ve başarısı test edilmiştir (Jayabarathi ve ark., 2016; Mittal ve ark., 2016; Gupta ve Deep, 2020; Yu ve ark., 2021). Bu tez kapsamında, sürüdeki etkin bireylerin çözüm kalitesine etkisini belirlemek adına

dinamik katsayı yöntemine dayanan ve lojik kapı tabanlı ikili XOR operatörü kullanan yeni bir ikili GWO varyantı önerilmektedir. Önerilen yöntem ile keşif ve sömürü arasındaki dengeyi koruyarak doğrudan ikili arama uzayında uygulanabilir çözümler elde edilmesi hedeflenmektedir. Önerilen yöntemin başarı tespiti için popüler ikili optimizasyon problemleri arasında yer alan özellik seçimi ve 0-1 sırt çantası problemleri üzerinde çeşitli deneyler gerçekleştirilmiştir.

Bu tez çalışması aşağıdaki gibi organize edilmiştir:

İlk bölüm olan Giriş başlığı altında, optimizasyon ve metasezgisel algoritmalar hakkında genel bilgilendirme yapılmış ve tezin içeriğine dair genel hatlar verilmiştir. İkinci bölümde önerilen yöntemin başarı tespitinde ele alınan ikili optimizasyon problemleri hakkında bir kaynak araştırması sunulmuştur. Üçüncü bölümde önerilen yöntemin başarı tespitinde ele alınan özellik seçimi ve 0-1 sırt çantası problemleri, önerilen yöntemin başarı tespiti için kullanılan veri setleri, çalışma ortamı, parametre ayarları, ikili varyantı önerilen orijinal GWO algoritması hakkında ayrıntılı bilgi verilmiştir. Dördüncü bölümde tez kapsamında önerilen İkili Dinamik Gri Kurt Optimizasyon Algoritması (IDGKO) sunulmuştur. Beşinci bölümde, ilgili problemler üzerinde gerçekleştirilen deneyler sunularak analizler yapılmıştır. Altıncı ve son bölüm olan Sonuç bölümünde ise tezin genel bir özeti, deneysel analizlerin kısa bir özeti ve elde edilen genel sonuçlar sunulmuş ve gelecekte yapılacak çalışmalar için önerilerde bulunulmuştur.

2. KAYNAK ARAŞTIRMASI

Tez kapsamında önerilen IDGKO algoritması, özellik seçimi ve 0-1 sırt çantası problemleri üzerinde test edilmiştir. Bu başlık altında, ilgili problemler hakkında literatüre kazandırılan çalışmalara dair literatür taraması sunulmuştur.

2.1. Özellik Seçimi Üzerine Yapılan Çalışmalar

Bu bölümde, sürü zekasına dayanan algoritmaları kullanarak özellik seçim probleminin çözümü için literatüre kazandırılan çalışmalara dair kısa bir tarama özeti sunulmuştur.

PSO, sürü zekasına dayanan popüler metasezgisel bir optimizasyon algoritmasıdır. Özellik seçimi probleminde de başarılı bir kullanılmış ve literatüre kazandırılmıştır. Too ve arkadaşları, elektromiyografi (EMG) özniteliklerinin sayısındaki artış sebebiyle sınıflandırma performansının düşürmemek ve sınıflandırıcı karmaşıklığını artırmamak için özellik seçimini önermektedir. Özellik seçimi problem sürecinde popülasyondaki bireylerin sahip olduğu yeni bir şahsi en iyi (pBest) yöntemine dayanan ikili PSO (Pbest-Guide Binary Particle Swarm Optimization, PBPSO) algoritması önerilmiştir. Önerilen yöntemin başarısını belirlemek adına ikili parçacık sürü optimizasyonu (BPSO), genetik algoritma (GA), modifiye ikili ağaç büyüme algoritması (MBTGA) ve ikili diferansiyel evrim (BDE) algoritması ile performans kıyaslaması yapılmıştır. Yapılan deneyler sonucunda önerilen yöntemin özellikle öznitelik indirgemedi, diğer yöntemlere göre üstün olduğunu ve yüksek bir sınıflandırma doğruluğunu korurken özelliklerin %90'ından fazlasını azaltabildiğini göstermektedir (Too ve ark., 2019b). Song ve arkadaşları, PSO'daki geleneksel parçacık güncelleme mekanizmaları ve sürü başlatma stratejilerinin, yüksek boyutlu özellik seçimi problemleriyle başa çıkma performansını engellemesi sebebiyle, karşılıklı bilgi (mutual information) ile yalın kemik (bare bone) yöntemlerine dayalı yeni bir ikili özellik seçim algoritması (Bare-Bones Particle Swarm Optimization, BBPSO) önermektedir. Algoritmada yakınsamayı hızlandırmak öznitelikler ve sınıf etiketleri arasındaki korelasyondan yararlanarak, etiket korelasyonuna dayalı etkili bir popülasyon başlatma stratejisi önerilmektedir. Algoritma performansını daha da artırabilmek için öznitelik ilgili/ilgisiz durumuna dayalı iki yerel arama operatörü geliştirilmiştir. Bunların yanında popülasyondaki parçacıkların yerel optimum çözümlerden dışarı sıçramasını engellemek

amacıyla uyarlanabilir bir mutasyon operatörü tasarlanmıştır. Önerilen algoritma için KNN sınıflandırıcısı kullanılmıştır. Önerilen yöntemin başarı tespiti için birçok veri kümesi üzerinde test edilmiş ve performans sonuçları son teknoloji 11 algoritma ile kıyaslanmıştır. Deneysel sonuçlar, önerilen algoritmanın, diğer algoritmalarından daha iyi performansa sahip bir özellik alt kümesi elde edebileceğini ve oldukça rekabetçi bir FS algoritması olduğunu göstermektedir (Song ve ark., 2021).

BA, yarasaların ekolojik davranışlarından esinlenerek tasarlanmış sürü zekasına dayanan metasezgisel bir optimizasyon algoritmasıdır. BA, önerildiği günden itibaren birçok farklı çalışmada başarılı bir şekilde kullanılmıştır. Rodrigues ve arkadaşları, özellik seçimi problemi için BA ve Optimum-Path Forest (OPF) tabanlı yeni bir sarmalayıcı özellik seçimi yaklaşımı önermektedir. Bununla birlikte, seçilecek özneliklerin kalitesini daha iyi tahmin etmek için bir metodoloji sunulmaktadır. 6 adet veri seti üzerinde gerçekleştirilen deneyler sonucunda, önerilen yaklaşımın istatistiksel olarak daha anlamlı indirgenmiş öznelik alt kümesi sağladığı ve bazı durumlarda sınıflandırma etkinliğini iyileştirebileceği tespit edilmiştir (Rodrigues ve ark., 2014a). Yang ve arkadaşları, destek vektör makinaları (Support Machine Machine, SVM) kullanarak özellik seçimi problemi için yeni bir ikili BA algoritması (MBAFS) önermiştir. Önerilen yöntemin özellik seçimi probleminde başarısını artırmak için erken yakınsamadan kaçınma ve popülasyon çeşitliliğini koruma amaçlı yeni mekanizmalar geliştirilmiştir. Önerilen yöntemin başarı tespiti için, performansı 12 kıyaslama veri seti üzerinde test edilmiş ve diğer BA tabanlı algoritmalar, iyi bilinen bazı BPSO tabanlı algoritmalar ile karşılaştırılmıştır. Yapılan deneyler sonucunda, önerilen algoritmanın diğer yöntemlerden daha iyi performans gösterdiğini ve hesaplama süresi açısından rekabetçi olduğunu göstermiştir (Yang ve ark., 2017).

DA doğadaki yusufçukların gezinme, yiyecek arama ve dinamik veya statik olarak sürü halindeyken düşmanlardan kaçınma esnasındaki sosyal etkileşimini modelleyerek tasarlanmış sürü zekasına dayanan metasezgisel bir optimizasyon algoritmasıdır. Sürekli optimizasyon problemlerinin yanı sıra ikili optimizasyon problemlerinin çözümü için de başarılı bir şekilde kullanılmaktadır. Mafarja ve arkadaşları, ikili DA'ya dayalı yeni bir sarmalayıcı özellik seçim algoritması (BDA) önermiştir. Önerilen yöntemin performansını değerlendirmek için 18 kıyaslama veri seti kullanılmış ve PSO, GA'lar ile sınıflandırma doğruluğu ve seçilen öznelik sayısı açısından karşılaştırılmıştır. Yapılan deneyler sonucunda BDA'nın özellik uzayını arama ve sınıflandırma için en gerekli özellikleri seçme konusunda yeteneğini kanıtlamaktadır (Mafarja ve ark., 2017b).

Sawhney ve Jain, Human Papillomavirus (HPV)'nin hızlı aracılığı ile oluşan hastalıkların son yıllarda artış göstermesi ve çok sayıda tedavi yöntemi arasından uygun yöntemin seçimini zorlaştırması sebebiyle yeni bir ikili sarmalayıcı özellik seçimi algoritması (Binary Dragonfly Algorithm, BDFA) önermektedir. Önerilen yöntemde DA'na optimum özellik seçimi için bir ceza fonksiyonu eklenmiş ve yeni bir evrimsel hesaplama tekniği önerilmiştir. Random Forest (RF) sınıflandırıcısını kullanan bu sarmalayıcı tabanlı yöntem hem immünoterapi hem de kriyoterapi tedavi yöntemlerinde kullanılmıştır. Önerilen yöntem bulanık kural tabanlı sistemler, genetik algoritmalar ve OPF sınıflandırıcılarına kıyasla hem sınıflandırma doğruluğunda hem de özellik seçiminde daha başarılı sonuçlar elde etmiştir (Sawhney ve Jain, 2018). Karizaki ve Tavassori, yüksek boyutlu verilerde öznitelik seçimi için hem filtre hem de sarmalayıcı yöntemleri içeren hibrit bir yöntem önermektedir. Önerilen algoritmada, özellikleri sıralamak için bir filtre yöntemi olan ReliefF algoritması kullanılmıştır. Sarmalayıcı yöntem olarak ise yeni bir ikili DA (BDA) kullanılmıştır. BDA algoritması, aşamalı ve yinelemeli olarak en uygun özellik alt kümesini bulmak için sıralanmış özellikleri kullanmaktadır. Elde edilen özellik alt kümelerini hiyerarşik olarak değerlendirmek için çapraz doğrulama kaybının en aza indirilmesi ve özellik sayısının azaltılması fikriyle hareket edilmektedir. Önerilen yöntem birçok farklı ikili yöntem ile 5 veri seti üzerinde kıyaslanmıştır. Deneyler sonucunda, önerilen algoritmanın yalnızca veri kümesinin boyutunu azaltmakla kalmayıp, aynı zamanda sınıflandırıcıların test verileri üzerindeki performansını da iyileştirdiğini gözlemlenmiştir (Karizaki ve Tavassoli, 2019). Cui ve arkadaşları, özellik seçimi probleminde optimum aday alt kümeleri oluşturmak ve daha yüksek sınıflandırma doğruluğu elde etmek amacıyla Maximum Relevance Minimum Redundancy Algorithm (mRMR) ve DA'ya dayalı yeni bir özellik seçim algoritması Hybrid Improved Dragonfly Algorithm (HIDA) önermektedir. Küçük ağırlıklı özelliklerin mRMR'de küçük bir olasılıkla aday alt küme seçilme şansı vardır. HIDA'nın sömürü ve keşif yeteneklerini, küresel ve yerel kapasiteyi dengelemek için dinamik kaynaşma faktörleri önerilmiştir. Önerilen yöntemin performansı birçok diğer yöntem ile çok sayıda veri seti üzerinde test edilmiştir. Elde edilen sonuçlar, önerilen yöntemin kıyaslama yapılan diğer yöntemlerden daha üstün olduğunu göstermektedir (Cui ve ark., 2020).

Balina Optimizasyon Algoritması (Whale Optimization Algorithm, WOA) kambur balinaların sosyal davranışlarını modelleyerek geliştirilen sürü zekasına dayalı metasezgisel bir optimizasyon algoritmasıdır (Mirjalili ve Lewis, 2016). Günümüze kadar

birçok farklı problemin çözümü için önerilmiştir. Özellik seçimi problemi gibi birçok ikili optimizasyon probleminin çözümünde faydalanılan bir yöntemdir. Tawhid ve Ibrahim, özellik seçim probleminin çözümü için yeni bir ikili WOA (BWOA) önermiştir. Yapılan ilk deneyde özellik seçimi probleminde etkili bir algoritma olan Kaba Küme Teorisi (Rough Set Theory, RST) BWOA ile kullanılmıştır. İkinci deneyde, özellik seçimi için üç farklı sınıflandırıcı kullanılarak BWOA ile yeni sarmalayıcı bir yaklaşım kullanılmıştır. Önerilen yöntemin performansı literatürdeki güçlü bazı algoritmalar ile 32 veri seti üzerinde kıyaslanmıştır. Bunun yanında parametrik olmayan iki istatistiksel test ile sonuçlar istatistiksel bakımından değerlendirilmiştir. Yapılan deneyler sonucunda önerilen algoritmanın performansı ve etkinliği doğrulanmış, özelliklerin optimum alt kümesini bulmak için verimli bir araç sağlayabileceği tespit edilmiştir (Tawhid ve Ibrahim, 2020). Hussien ve arkadaşları, boyut azaltma ve sınıflandırma problemlerinde en iyi özellik alt kümesini bulmak için yeni bir ikili WOA (bWOA) önermiştir. Sürekli arama uzayına sahip algoritmayı ikili hale getirirken S-şekilli adı verilen transfer fonksiyonu kullanılmıştır. Sınıflandırıcı olarak KNN yöntemini kullanan bWOA, yapılan deneyler sonucunda optimum özellikleri bulmada kayda değer bir performansa sahip olduğu gözlemlenmiştir (Hussien ve ark., 2019). Mafarja ve Mirjalili, WOA'na dayalı yeni bir sarmalayıcı özellik seçimi yaklaşımı önermiştir. Çalışmada özellik seçimi problemi için iki adet ikili WOA varyantı sunulmuştur. İlk varyantta, arama sürecinde rastgele bir operatör kullanmak yerine Turnuva ve Rulet Çarkı seçim mekanizmalarını kullanılarak çözüme etkisi incelenmiştir. İkinci varyantta ise WOA algoritmasının kullanımını geliştirmek için çaprazlama ve mutasyon operatörleri kullanılmıştır. Önerilen her iki varyant standart bazı veri setleri üzerinde test edilmiş ve performans sonuçları PSO, GA, ALO, beş standart filtre özellik seçim yöntemi algoritması ile karşılaştırılmıştır. Çalışmada ayrıca önerilen varyantlar için kapsamlı bir parametre ayarı çalışması yer almaktadır. Yapılan deneyler sonucunda, önerilen varyantların optimum özellik kümesini aramada etkili bir yaklaşım oldukları görülmektedir (Mafarja ve Mirjalili, 2018).

Salp Sürü Algoritması (Salp Swarm Algorithm, SSA), okyanuslardaki salpların gezinme ve yiyecek arama davranışlarından esinlenerek tasarlanmış sürü zekasına dayalı metasezgisel bir optimizasyon algoritmasıdır (Mirjalili ve ark., 2017). SSA, ikili optimizasyon problemlerinin çözümünde geliştirilen yeni teknikleri ile başarılı bir kullanılmaktadır. Ibrahim ve arkadaşları, özellik seçimi probleminde SSA'nın kullanımını önererek bu alandaki ilk çalışmayı (SSA-FS) yapmışlardır. Önerilen yöntem

Irak Hastanelerindeki meme, mesane ve kolon kanseri hasta verilerinden elde edilen gerçek veri setleri üzerinde ve sentetik veri setleri üzerinde özellik seçimi için kullanılmıştır. Yöntem doğruluk ve çalışma süresi bakımından PSO ve DE algoritmaları ile karşılaştırılmıştır. Yapılan deneyler sonucunda SSA-FS'nin hem gerçek hem de sentetik veri setlerinde diğer algoritmalara kıyasla daha az çalışma süresi ile en yüksek doğruluğu elde ettiği kanıtlanmıştır (İbrahim ve ark., 2017). Hegazy ve arkadaşları, orijinal SSA'nın çözüm doğruluğunu, güvenilirliğini ve yakınsama hızını artırmak için yeni bir geliştirilmiş SSA (ISSA) önermiştir. Atalet ağırlığı adı verilen bir kontrol parametresi en iyi çözümü ayarlamak için kullanılmıştır. Önerilen yöntemin başarı tespiti için özellik seçim probleminde testler gerçekleştirilmiştir. Sınıflandırıcı olarak KNN yöntemi tercih edilen ISSA algoritması, 23 veri seti üzerinde test edilmiş ve 4 sürü zekasına dayalı metasezgisel algoritma ile karşılaştırılmıştır. Yapılan deneyler sonucunda, önerilen yöntemin sınıflandırma doğruluğu ve en iyi özellik alt kümesi seçimi açısından diğer yöntemlerden daha üstün sonuçlar verdiği kanıtlanmıştır (Hegazy ve ark., 2020). Faris ve arkadaşları, sürü zekasına dayalı algoritmaların özellik seçimi probleminde gösterdikleri üstün başarılarından yola çıkarak, SSA algoritmasını özellik seçimi probleminin çözümü için önermiştir. Çalışmalarında SSA için iki yeni sarmalayıcı özellik seçimi yaklaşımı yer almaktadır. İlk yaklaşımda, SSA'nın sürekli arama uzayını ikili hale dönüştürmek için sekiz transfer fonksiyonu kullanılmıştır. İkinci yaklaşımda, algoritmanın keşif davranışını geliştirmek için transfer fonksiyonlarına ek olarak çaprazlama operatörü kullanılmıştır. Önerilen yaklaşımlar, 22 adet veri seti üzerinde literatüre kazandırılmış İkili Gri Kurt Optimize Edici (BGWO), İkili Yerçekimi Arama Algoritmaları (BGSA), İkili Yarasa Algoritması (BBA), İkili Parçacık Sürü Optimizasyonu (BPSO) ve Genetik Algoritma (GA) 5 adet ikili yöntem ile performans bakımından kıyaslanmıştır. Yapılan deneyler sonucunda, önerilen yaklaşımın veri setlerinin yaklaşık %90'ında diğer yöntemlerden önemli ölçüde daha iyi performans gösterdiği kanıtlanmıştır (Faris ve ark., 2018).

Karınca Aslanı Optimize Edici (The Ant Lion Optimizer, ALO), doğadaki karınca aslanlarının rastgele yürüyüşü, tuzak kurma, karıncaların tuzaklara yakalanması, av yakalama ve tuzakları yeniden kurma gibi avlanma mekanizmasından esinlenerek modellenmiş sürü zekasına dayalı metasezgisel bir optimizasyon algoritmasıdır (Mirjalili, 2015). Özellik seçimi problemi için literatüre kazandırılan sürü zekasına dayalı popüler algoritmalar arasında yer almaktadır. Zawbaa ve arkadaşları, etkili arama kabiliyetinden yola çıkarak sarmalayıcı tabanlı özellik seçimi probleminde ALO algoritmasını

önermiştir. Önerilen yöntem, 18 farklı veri seti üzerinde farklı değerlendirme kriterleri kullanılarak değerlendirilmiş ve iki yaygın arama yöntemi olan PSO ve GA ile karşılaştırılmıştır. Elde edilen sonuçlar, önerilen yöntemin sınıflandırma performansı ve seçilen özellik alt kümesi bakımından etkisini kanıtlamaktadır (Zawbaa ve ark., 2015). Emary ve arkadaşları, sarmalayıcı tabanlı özellik seçimi için ALO'nun yeni ikili varyantını önermiştir. İlk varyantta, orijinal ALO operatörlerinden ilham alarak geliştirilen ikili operatörler kullanılmıştır. İkinci varyantta, uygun eşik fonksiyonu kullanılarak sürekli adımlar ikili formata dönüştürülmüştür. Önerilen yöntemler 21 farklı veri seti üzerinde PSO, GA, BA ve orijinal ALO ile kıyaslanmıştır. Sonuçlar, önerilen ikili yöntemlerin başlatma ve kullanılan stokastik operatörlerden bağımsız olarak optimum özellik alt kümesini belirlemek için özellik uzayını arama kabiliyetini kanıtlamaktadır (Emary ve ark., 2016b). Mafarja ve arkadaşları, sarmalayıcı özellik seçim yöntemi olarak ALO'nun varyantlarını önermiştir. Çalışmada 6 adet ALO varyantı kullanılmış ve her biri sürekli arama uzayını ikili hale getirmek için geliştirilmiş S-şekilli ve V-şekilli transfer fonksiyonu içermektedir. Önerilen varyantların performansı, 18 veri seti üzerinde test edilmiş ve literatürde mevcut PSO, GSO ve önerilen diğer ikili ALO yöntemleriyle karşılaştırılmıştır. Deneyler sonucunda, önerilen yaklaşımların özellik uzayını verimli bir şekilde aradığı ve sınıflandırma doğruluğunu geliştirmeye yardımcı olan en önemli özellikleri seçtiği gözlemlenmiştir (Mafarja ve ark., 2017a).

Tez kapsamında ikili optimizasyon problemlerinin çözümü için önerilen GWO, özellik seçimi probleminde sıklıkla başvurulmuş bir yöntem olup çeşitli ikili varyantları literatüre kazandırılmıştır. Emary ve arkadaşları, özellik seçimi problemi için iki farklı yöntem kullanan yeni ikili GWO algoritmasını (bGWO) önermiştir. İlk yöntemde, en iyi çözüme sahip ilk üç bireye yönelik bireysel adımlar üzerinde ikilileştirme yaparak güncel birey konumunu tespit etmek için orijinal algoritmaya ait üç temel hareket arasında stokastik çaprazlama işlemi yapılmaktadır. İkinci yöntemde ise, konum güncelleme işleminde sigmoidal fonksiyon kullanılarak hesaplanan güncel konum değerlerini stokastik olarak eşikleme işlemi önerilmektedir. Önerilen iki yeni yöntem bu alanda kullanımı oldukça yaygın optimize edicilerden ikisi olan PSO ve GA ile 18 farklı veri seti üzerinde karşılaştırılmıştır. Deneyler sonucunda önerilen iki yeni yaklaşımın, başlatma ve kullanılan stokastik operatörlerden bağımsız olarak optimum özellik kombinasyonları için özellik uzayını arama kabiliyeti kanıtlanmıştır (Emary ve ark., 2016a). Li ve arkadaşları, tıbbi teşhis kapsamında tıbbi veriler için en iyi özellikleri tespit etmek amacıyla geliştirilmiş bir GWO algoritması (IGWO) ve IGWO-KELM olarak

adlandırılan Çekirdek Aşırı Öğrenme Makinesi (Kernel Extreme Learning Machine, KELM) kullanarak yeni bir yöntem önermiştir. IGWO, ilk olarak çeşitlendirilmiş başlangıç konumlarını oluşturmak için GA kullanmakta ve daha sonra popülasyonun mevcut konumlarını ayrık arama uzayında güncellemek için GWO algoritmasını kullanmaktadır. Önerilen yaklaşım, sınıflandırma doğruluğu (classification accuracy), duyarlılık (sensitivity), özgüllük (specificity), kesinlik (precision), G-ortalama (G-mean), F-ölçüm (F-measure) ve seçilen özellik alt küme boyutu da dahil olmak üzere birçok performans ölçütü kullanarak iki yaygın hastalık tanı probleminde testler gerçekleştirilmiştir. IGWO performans kıyaslaması bakımından orijinal GA ve GWO ile karşılaştırılmıştır. Yapılan deneyler sonucunda, önerilen yöntemin diğer iki algoritmaya göre üstünlüğü kanıtlanmıştır (Li ve ark., 2017b). Too ve arkadaşları, elektromiyografi sinyalinden çıkarılan özelliklerin alakasız ve gereksiz niteliği taşıyabilmesi nedeniyle özellik kümesini bu tarz özelliklerden arındırmak amacıyla yeni bir ikili GWO algoritması (CBGWO) önermiştir. İlk aşamada kısa zamanlı Fourier dönüşümü (STFT) ile EMG sinyalleri zaman-frekans gösterimine dönüştürülmektedir. STFT katsayısından 10 adet zaman-frekans özelliği çıkarılmaktadır. Ardından CBGWO, orijinal özellik kümesinden en uygun özellik alt kümesini değerlendirmek için kullanılmaktadır. Önerilen yöntemin performansı ikili gri kurt optimizasyonu (BGWO1 ve BGWO2), ikili parçacık sürü optimizasyonu (BPSO) ve genetik algoritma (GA) ile karşılaştırılmıştır. Deneysel sonuçlar, CBGWO'nun sadece sınıflandırma performansında değil, aynı zamanda özellik sayısını azaltmada da üstünlüğünü göstermektedir. Ayrıca CBGWO'nun gerçek dünya problemleri için çok düşük bir hesaplama maliyetine sahip olduğu vurgulanmıştır (Too ve ark., 2018). Devanathan ve arkadaşları, Bag-of-Keypoint Features (BoKF) modeline ve özellik seçimi konusunda ikili GWO algoritmasına (BGWO) dayalı nükleolar ve centromere boyama modellerini ayırt etmek için bir girişimde bulunmuştur. Floresan boyama desenleri, Dolaylı İmmünofloresan (Indirect Immunofluorescence, IIF) görüntüleme ile üretilmektedir. IIF görüntüleri, kenara duyarlı yerel kontrast geliştirme yöntemi kullanılarak önceden işlenmektedir. Kontrast artırılmış görüntüler BoKF çerçevesine tabi tutulmuş ve Hızlandırılmış Sağlam Özellik (Speeded up Robust Feature) anahtar noktaları çıkarılmıştır. En önemli özellik alt kümesini belirlemek için BGWO ve sınıflandırma için KNN yöntemi kullanılmıştır. Yapılan deney sonuçları, BGWO'nun nükleolar ve centromere modellerini ortalama %91.6 doğrulukla sınıflandırabildiğini göstermektedir. Sonuçlar ayrıca BGWO kullanılarak elde edilen belirgin özelliklerin IIF boyama modellerini ayırt etme performansını iyileştirebileceğini göstermektedir. Bundan

yola çıkararak, BGWO tabanlı özellik seçiminin otoimmün hastalıkların bilgisayar destekli teşhisini mümkün kıldığı kanıtlanmıştır (Devanathan ve ark., 2019). Al-Tashi ve arkadaşları, sürekli ve çok amaçlı optimizasyon problemlerini çözmek için yakın zamanda önerilen Çok Amaçlı Gri Kurt Optimize Edici (Multi-objective Grey Wolf Optimizer, MOGWO) algoritmasını ayrik ve çok amaçlı özellik seçimi problemlerine uyarlamak amacıyla sigmoid transfer fonksiyonuna dayalı ikili bir versiyonunu (BMOGW-S) geliştirmişlerdir. Seçilen özellik alt kümesinin sınıflandırma performansını değerlendirmek için sarmalayıcı tabanlı bir Yapay Sinir Ağı (Artificial Neural Network, ANN) kullanılmıştır. Önerilen yöntemin performansını doğrulamak için 15 adet veri seti üzerinde; tanh transfer fonksiyonu ve Domine Edilmeyen Sıralama Genetik Algoritması (Non-dominated Sorting Genetic Algorithm, NSGA-II), Çok Amaçlı Parçacık Sürü Optimizasyonu (MOPSO) ve orijinal MOGWO ile karşılaştırma yapılmıştır. Sonuçlar, önerilen yöntemin oldukça düşük bir hesaplama maliyetine sahip olduğunu ve çoğu durumda özellik azaltma ve sınıflandırma hata oranı açısından mevcut çok amaçlı yaklaşımlardan daha iyi performansa sahip olduğunu göstermektedir (Al-Tashi ve ark., 2020). Hu ve arkadaşları, ikili optimizasyon problemlerinin çözümü için yeni bir geliştirilmiş ikili GWO (BGWO) önermiştir. Önerilen yöntemde ikili koşul altında değer aralığını analiz etmek ve parametre için global arama ve yerel arama yeteneklerini dengelemek için yeni bir güncelleme denklemi önerilmektedir. BGWO, orijinal GWO'daki sürekli değerleri ikili değerlere dönüştürmek için transfer fonksiyonu kullanmaktadır. Burada 5 farklı transfer fonksiyonu kullanılmakta ve bu fonksiyonların çözüm kalitesini iyileştirmeye odaklanılmaktadır. Yapılan deneyler sonucunda, geliştirilmiş ikili GWO'nun optimallik, zaman tüketimi ve yakınsama hızında orijinal ikili GWO'dan (BGWO) daha üstün olduğu kanıtlanmıştır. Ayrıca, önerilen yöntemin veri kümelerinde özellik seçimini başarıyla uyguladığı ve az özellekle düşük sınıflandırma hataları elde ettiği vurgulanmıştır (Hu ve ark., 2020). Abasi ve arkadaşları, metin özellik seçimi problemini çözmek için yeni bir ikili GWO (BGWO) algoritması önermektedir. Bu yöntem, metinden bilgilendirici özellikler seçerek GWO algoritmasının yeni bir uygulamasını sunmaktadır. BGWO'nun performansı Tr41, Tr12, Wap, Classic4, 20Newsgroups ve CSTR dahil olmak üzere altı veri kümesi üzerinde ikili PSO (BPSO) ve GA ile kıyaslanmıştır. Yapılan deneylerin sonucunda, BGWO'nun değerlendirme kriterlerine dayalı olarak GA ve BPSO gibi karşılaştırılan diğer algoritmalarından daha iyi performansa sahip olduğu kanıtlanmıştır (Abasi ve ark., 2021). Sathiyabhama ve arkadaşları, kaba küme teorisi (rough set theory) ile GWO'ya dayalı mamogram görüntü

analizi için yeni bir CAD sistemi önermiştir. Doku, yoğunluk ve şekil tabanlı özellikler, kitle segmentli mamogram görüntülerinden çıkarılmaktadır. Çıkarılan mamogram görüntülerinden önemli özellikleri bulmak için GWO ve Rough Set (GWORS) yöntemlerinin hibridizasyonu kullanılmıştır. Önerilen GWORS'un performans tespiti için PSO, GA, Hızlı İndirgeme (Quick Reduct) ve Görelî İndirgeme (Relative Reduct) dahil olmak üzere iyi bilinen diğer kaba küme ve biyo-ilhamlı özellik seçim algoritmaları ile kıyaslama yapılmıştır. Deneyler sonucunda, önerilen GWORS'un doğruluk (accuracy), F-Ölçüleri (F-Measures) ve alıcı çalışma karakteristik eğrisi (receiver operating characteristic curve) açısından diğer tekniklerden daha iyi performans gösterdiği kanıtlanmıştır (Sathiyabhama ve ark., 2021). Deep, kronik hastalıklarda hastalığın erken teşhisi için ilgili özelliklerin belirlenmesi oldukça önem taşıdığından kronik hastalık verilerinde özellik seçimi için GWO'ya dayalı yeni bir algoritma önermektedir. Önerilen algoritma, GWO'nun yerel arama kabiliyetini daha da geliştirmek ve keşif ile sömürü arasında bir denge kurabilmek için dağılım faktörü yaklaşımına dayalı Rastgele Yürüyen Gri Kurt Optimize Edici (Random Walk Grey Wolf Optimizer, RWGWO) algoritmasıdır. Önerilen yöntem 18 farklı kronik hastalık verisi üzerinde birkaç son teknoloji yöntemle karşılaştırılmıştır. Sonuçlar, RWGWO yönteminin diğer yöntemlere göre en iyi sırada olduğunu ve her bir kronik hastalık verisindeki özellik boyutunu büyük ölçüde azaltabildiğini göstermektedir (Deep, 2022). Wang ve arkadaşları, yüksek boyutlu sınıflandırma için optimum özellik alt kümesini belirlemede Uyarlanabilir Dengeli Gri Kurt Optimizasyonu (Adaptively Balanced Grey Wolf Optimization, ABGWO) algoritmasını önermiştir. ABGWO'da, keşif yeteneğini artırmak ve iş birliği yapmak için rastgele bir kurt tanımlanır. Rastgele kurdu seçmek için seviye tabanlı yeni bir strateji önerilmiştir. Ayrıca, farklı optimizasyon aşamalarında keşif ve sömürü yeteneğini dinamik olarak modüle etmek için GWO'daki lider üç birey ve rastgele seçilen kurdun liderliğini düzenleyici uyarlanabilir bir katsayı geliştirildi. Önerilen yöntem, 12 adet yüksek boyutlu veri kümesi üzerinde en gelişmiş yedi özellik seçim yaklaşımıyla karşılaştırılmıştır. Sonuçlar, ABGWO'nun sınıflandırma doğruluğu, özellik alt kümesi boyutu ve hesaplama süresi açısından diğer yöntemlere göre üstün olduğunu göstermektedir (Wang ve ark., 2022).

Literatüre kazandırılan birçok sürü zekasına dayalı metasezgisel optimizasyon algoritması özellik seçimi problemi için önerilmiştir. Bunlardan bazıları aşağıda listelenmiştir:

- Yapay Arı Kolonisi Algoritması (Artificial Bee Colony Optimization, ABC) (Ghanem ve Jantan, 2016; Özger ve ark., 2016; Kiliç ve Keleş, 2018; Rao ve ark., 2019; Wang ve ark., 2020; Rani, 2021)
- Guguk Kuşu Arama Algoritması (Cuckoo Search Algorithm, CS) (Rodrigues ve ark., 2013; Gunavathi ve Premalatha, 2015; Sudha ve Selvarajan, 2016; Pandey ve ark., 2020; Alzaqebah ve ark., 2021; Mehedi ve ark., 2021)
- Çiçek Tozlaşması Algoritması (Flower Pollination Algorithm, FPA) (Sayed ve ark., 2016; Rajamohana ve ark., 2017; Majidpour ve Soleimani Gharehchopogh, 2018; Zawbaa ve Emary, 2018; Yan ve ark., 2019; Yan ve ark., 2022)
- Kril Sürüsü Algoritması (Krill Herd Algorithm, KHA) (Rodrigues ve ark., 2014b; Rani ve Ramyachitra, 2017; Zhang ve ark., 2020)
- İmparator Penguenler Optimize Edici (Emperor Penguins Optimizer, EPO) (Baliarsingh ve ark., 2019; Baliarsingh ve Vipsita, 2020; Dhiman ve ark., 2021; Kalra ve ark., 2022)

Farklı metasezgisel algoritmaların, erken yakınsamadan ve yerel optimuma takılmaktan kaçınmak, arama uzayını etkili ve verimli bir şekilde kullanıp keşfetmek, çözüm kalitesini artırmak vb. gibi sebeplerle bireysel verimli operatörlerini ve arama stratejilerini birleştirerek geliştirdikleri yöntemler hibrit yöntemler olarak adlandırılmaktadır. Son yıllarda popüler hale gelen ve problem çözümlerinde başarıyla kullanılan birçok hibrit metasezgisel algoritma bulunmaktadır. Bu algoritmalar özellik seçimi problemlerinde de sıklıkla kullanılmaktadır. Bunlardan bazıları aşağıda örneklendirilmiştir:

- BA ve PSO algoritmaları kullanılarak geliştirilen Hybrid Binary Bat Enhanced Particle Swarm Optimization Algorithm (HBBEPSO) (Tawhid ve Dsouza, 2018)
- Maymun Algoritması (Monkey Algorithm, MA) ve KHA algoritmaları kullanılarak geliştirilen Hybrid Monkey Algorithm and Krill Herd Algorithm (MAKHA) (Hafez ve ark., 2015)
- GWO ve Karga Arama Algoritması (Crow Search Algorithm, CSA) kullanılarak geliştirilen Hybrid GWO with CSA (GWOCSA) (Arora ve ark., 2019)
- GWO ve Çekirge Optimizasyon Algoritması (Grasshopper Optimization Algorithm, GOA) kullanılarak geliştirilen Hybrid GWO–GOA (Purushothaman ve ark., 2020)

- GWO ve PSO algoritmaları kullanılarak geliştirilen Hybrid Binary GWO-PSO Algorithm (BGWOPSO) (Al-Tashi ve ark., 2019)
- WOA ve GWO kullanılarak geliştirilen hibrit algoritma (Mafarja ve ark., 2020)

2.2. 0-1 Sırt Çantası Problemi Üzerine Yapılan Çalışmalar

Bu bölümde, sürü zekasına dayanan algoritmaları kullanarak 0-1 sırt çantası probleminin çözümü için literatüre kazandırılan çalışmalara dair kısa bir tarama özeti sunulmuştur.

0-1 KP'nin çözümü için sıklıkla başvuru alan yöntemlerin arasında metasezgisel optimizasyon algoritmaları yer almaktadır. Bu bağlamda literatüre birçok çalışma kazandırılmıştır.

Karınca Kolonisi Optimizasyonu (Ant Colony Optimization, ACO) bazı karınca türlerinin yiyecek arama davranışlarından ilham alarak geliştirilmiş sürü zekasına dayalı bir optimizasyon algoritmasıdır (Dorigo ve ark., 2006). Shi, gezgin satıcı problemi (Traveling Salesman Problem, TSP) ve 0-1 KP için ACO'daki parametreler ve algoritma modeli üzerinde değişiklik yaparak ikili geliştirilmiş ACO yöntemini önermiştir. Geliştirilmiş ACO'ya dayalı deneyler, algoritmanın sağlamlığını ve güçlü performansa sahip olduğunu göstermiştir (Shi, 2006).

Bansal ve Deep, literatüre kazandırıldığı günden itibaren birçok problemin çözümünde başarılı bir şekilde kullanılan PSO algoritmasının, özellikle 0-1 KP ve Çok Boyutlu KP olmak üzere KP'leri çözmek için geliştirmiş ikili PSO (MBPSO) versiyonunu önermişlerdir. MBPSO'nun, sürüdeki çeşitliliği korumak ve KP'lerin çözümünde daha keşfedici, etkili ve verimli olması için yeni bir olasılık işlevi önerilmektedir. Bu yönüyle önerilen yöntem, temel ikili PSO (BPSO) ile farklılık göstermektedir. MBPSO'nun performansı, kıyaslama problemleri (benchmark problems) üzerinde test edilmiş ve sonuçlar BPSO, Genotip-Fenotip Modifiye Edilmiş İkili PSO (Genotype-Phenotype Modified Binary Particle Swarm Optimization, GPMBPSO) ile karşılaştırılmıştır. 0-1 KP ve Çok Boyutlu KP'nin çeşitli veri örnekleriyle yapılan ve performans bakımından BPSO, GPMBPSO ile karşılaştırılan MBPSO yönteminin başarılı bir yöntem olduğu gözlemlenmiştir (Bansal ve Deep, 2012).

Guguk Kuşu Arama Algoritması (CS), guguk kuşlarının nüfuslarını artırmak için kuluçka üreme stratejisine dayanan bir optimizasyon algoritmasıdır (Bansal ve Deep, 2012). Feng ve arkadaşları, 0-1 KP'nin çözümü için açgözlü stratejiye sahip geliştirilmiş

bir hibrit kodlamalı CS (Improved Hybrid Encoding Cuckoo Search Algorithm, ICS) önermişlerdir. İkili optimizasyon problemlerini ICS ile çözmek için, bireysel hibrit kodlama fikrine dayalı olarak, guguk kuşu araması ayırık arama uzayında senkron evrim aramaya dönüştürülmüş ve ardından güven aralığı (confidence interval, CI) kavramı algoritmaya entegre edilmiştir. Bu sayede sürekli optimizasyon problemlerinin çözümü için üretilen CS'nin konum güncelleme denklemi, ikili optimizasyon problemleri için uygun hale getirilmiştir. Bununla beraber ICS'nin, hızla global optimum çözüme doğru ilerleyebilmesini ve yerel optimuma hapsolmesini etkili bir şekilde önleyebilmek için küçük değerli olasılıkla genetik mutasyon kullanılmıştır. Problem için uygun olmayan çözümü onarmak ve uygun çözümü optimize etmek için açgözlü dönüşüm yöntemi kullanılmıştır. Çok sayıda 0-1 KP örneğiyle yapılan deneyler, önerilen algoritmanın etkinliğini ve iyi kalitede çözümler elde etme yeteneğini göstermiştir (Feng ve ark., 2014).

Yapay Balık Sürüsü Algoritması (Artificial Fish Swarm Algorithm, AFSA), beslenme sürecindeki balıkların davranışlarını ve aralarındaki bilgi etkileşimlerini taklit eden sürü zekâsı dayalı optimizasyon algoritmasıdır (Li, 2002). Azad ve arkadaşları, 0-1 KP'nin çözümü için yeni ikili AFSA (S-bAFSA) yöntemi önermiştir. S-bAFSA'da çaprazlama ve mutasyon kullanılarak deneme noktaları oluşturulmakta ve bu noktalar için rastgele bir buluşsal öge (drop_item) prosedürü kullanılmaktadır. Prosedürün temel amacı, çözümlerin kalitesini artırmak ve optimum olmayan çözümlere yakınsamayı önlemektir. Önerilen yöntemin başarısı birçok 0-1 KP verileri ile test edilmiştir. Test sonuçları algoritmanın başarısını kanıtlamaktadır (Azad ve ark., 2014).

Maymun Algoritması (Monkey Algorithm, MA), maymunların dağa tırmanma süreçlerinden etkilenerik tasarlanmış sürü zekasına dayalı metasezgisel bir algoritmadır (Zhang ve Liu, 2008). Zhou ve arkadaşları, MA'nın yerel arama yeteneğini güçlendirmek için açgözlü bir algoritmanın kullanıldığı, tırmanıştaki takla işleminin yerel optimum çözümlere düşmekten kaçınmak için değiştirildiği ve iş birliği sürecinin benimsendiği yeni ikili MA algoritması önermiştir. Önerilen yöntemin başarı tespiti için, çeşitli 0-1 KP veri örnekleriyle deneyler yapılmış ve sonuçlar beş metasezgisel algoritmanın sonuçlarıyla karşılaştırılmıştır. Elde edilen sonuçlar, önerilen yöntemin başarısını kanıtlamaktadır (Zhou ve ark., 2016).

Hükümdar Kelebek Optimizasyonu (Monarch Butterfly Optimization, MBO), yaz sonunda/sonbaharda kuzey ABD ve güney Kanada'dan Meksika'ya binlerce mil kat ederek güneye doğru göç etmesiyle bilinen hükümdar kelekelerin göç eylemini

basitleştirerek ve idealleştirerek geliştirilen metasezgisel bir algoritmadır (Wang ve ark., 2019). Feng ve arkadaşları, yaptıkları bir çalışmada 0-1 KP'nin çözümü için yeni ikili MBO'yu (BMBO) önermektedir. BMBO'da gerçek değerli vektörler ve ikili vektörlerden oluşan iki vektör kullanılmıştır. Gerçek değerli vektörler arama uzayını, ikili vektörler ise çözüm uzayını temsil etmektedir. BMBO'da bireylerin daha iyi performans elde etmesi için üç tür bireysel tahsis planı üzerinde testler gerçekleştirme ve uygun olmayan çözümleri revize eden; uygun olanları optimize eden ağgözlü stratejiye dayalı yeni bir onarım operatörü şeklinde yeni iki mekanizma bulunmaktadır. Önerilen yöntemin performansı üç adet 0-1 KP örneği üzerinde kapsamlı sayısal deneyler ile test edilmiş ve sonuçlar son teknoloji dört algoritma ile karşılaştırılmıştır. Sonuçlar, BMBO'nun özellikle yüksek boyutlu 0-1 KP örneklerini çözmeye arama doğruluğu, yakınsama yeteneği ve kararlılık açısından üstünlüğünü kanıtlamaktadır (Feng ve ark., 2017). Feng ve arkadaşları, karşıtlık temelli öğrenme (opposition-based learning, OBL) stratejisinin evrimin son aşamasında popülasyonun yarısı üzerinde kullanıldığı ve Gauss pertürbasyonunun kısmen etki ettiği Gauss Pertürbasyonu ile Genelleştirilmiş Karşıtlık Temelli Öğrenme Hükümdar Kelebek Optimizasyonu 'nu (OMBO) önermiştir. Gauss pertürbasyonu yerel optimumda takılmayı önleyen bir yöntemdir. Önerilen yöntemin performansını test etmek için 800 ila 2000 boyutlu 15 büyük ölçekli 0-1 KP örneğinden oluşan üç adet veriseti kullanılmıştır. Deneysel sonuçlar, OMBO'nun yüksek kaliteli çözümler bulabileceğini göstermektedir (Feng ve ark., 2018).

Wang ve arkadaşları, DA (Yusufçuk Algoritması)'nın yeni bir ikili versiyonunu önermektedir. Önerilen yöntemde DA'yı ikili arama uzayında etkinleştirebilmek için açı modülasyon mekanizması (angle modulation mechanism, AMDA) kullanılmış ve böylece 0-1 düzeyinde bit dizileri üretilmiştir. Buradaki trigonometrik fonksiyonun katsayılarını optimize etmek için DA kullanılmaktadır. Ayrıca, algoritma kararlılığını ve yakınsama hızını iyileştirmek için orijinal üretici fonksiyonun kosinüs kısmının dikey yer değiştirmesini ayarlamak için bir katsayı daha eklenerek IAMDA adı verilen geliştirilmiş bir AMDA önerilmiştir. IAMDA ve AMDA'nın performansını test etmek için 13 adet klasik kıyaslama problemi ve 12 adet 0-1 KP verileri kullanılmıştır. Yapılan deneyler sonucunda, IAMDA'nın diğer algoritmalara kıyasla üstün bir yakınsama hızına ve çözüm kalitesine sahip olduğunu kanıtlanmıştır (Wang ve ark., 2021b). 0-1 KP'nin çözümü için DA'yı öneren diğer bir yöntem Abdel-Basset ve arkadaşları tarafından önerilen İkili Yusufçuk Algoritması (BDA)'dır. 0-1 KP veri örnekleriyle yapılan ve literatürdeki diğer

yöntemler ile kıyaslanan BDA, literatürdeki diğer algoritmalara kıyasla üstün performansını kanıtlamıştır (Abdel-Basset ve ark., 2017).

Yassien ve arkadaşları, ağlardaki 0-1 KP'nin çözümü için GWO ve K-Anlamalı Kümeleme Algoritması (K-means Clustering Algorithm) kullanan yeni bir algoritma önermektedir. Önerilen algoritmanın problem çözümünde etkili ve verimli olduğu gözlemlenmiştir (Yassien ve ark., 2017). Wang ve arkadaşı, özellikle yüksek boyutlu 0-1 KP için çeşitlilik ve yakınsama oranını artırmak amacıyla Kuantum İlhamlı Diferansiyel Evrim Algoritması (Quantum-Inspired Differential Evolution Algorithm) ile GWO'nun yeni bir hibrit yöntemini (QDGWO) önermektedir. Önerilen algoritma, kuantum süperpozisyon durumları ve kuantum kapıları gibi kuantum hesaplama ilkelerini benimsemektedir. Ayrıca, deneme bireyler ile yeni çözümler üretmek için adaptif mutasyon ve çaprazlama yöntemleri ve kuantum gözlemi kullanılmaktadır. Algoritma yer alan seçim işlemleri, mevcut bireyler ile mutasyon ve çaprazlama işlemleriyle oluşturulan deneme bireyler arasından daha iyi çözümleri belirlemek için kullanılır. Deneme bireylerinin mevcut bireylerden daha kötü olması durumunda, popülasyonun çeşitliliğini korumak ve global optimum çözüm arayışını hızlandırmak için uyarlanabilir gri kurt iyileştirici ve kuantum döndürme kapısı kullanılmıştır. Yapılan deneyler sonucunda QDGWO'nun özellikle yüksek boyutlu 0-1 KP veri örnekleri için etkinlik ve global arama yeteneği hakkında avantajları kanıtlanmıştır (Wang ve Wang, 2021). Roayaei, set kodlamasına dayanan ve güncelleme stratejisinde set işlemlerini kullanan yeni bir ikili GWO algoritması (SetGWO) önermiştir. Önerilen algoritma; transfer fonksiyonuna ve sınır kontrolüne olan ihtiyacı ortadan kaldıran, daha düşük boyutlu arama ajanları kullanan tamamen farklı bir kodlama şeması kullanılmaktadır ve bu sayede çalışma süresi kısalmaktadır. Ayrıca, her arama ajanı için özel bir keşif seti kullanılarak, farklı bir mesafe ölçüsü tanımlanarak ve ayırık alanlarda bir kuşatma stratejisi kullanılarak çözümlerin kalitesini artırmak hedeflenmektedir. 0-1 KP'nin de dahil olduğu farklı gerçek dünya kombinatoriyal optimizasyon problemlerinin çözümü için önerilen SetGWO'nun performansı farklı veri kümeleri üzerinde test edilmiştir. Deneyler sonucunda SetGWO'nun çözüm kalitesi, çalışma süresi ve ölçeklenebilirlik açısından mevcut diğer ikili GWO algoritmalarından daha iyi performans gösterdiği kanıtlanmaktadır (Roayaei, 2021).

Yapay Arı Kolonisi (Artificial Bee Colony, ABC) bal arılarının yiyecek arama davranışlarından ilham alınarak tasarlanmış sürü zekasına dayanan metasezgisel bir optimizasyon algoritmasıdır. Cao ve arkadaşları, 0-1 KP'nin çözümü için Diferansiyel

Evrım ile İkili Yapay Arı Kolonisi Algoritması (Binary Artificial Bee Colony Algorithm with Differential Evolution, BABC-DE) önermiştir. BABC-DE'de, görevli arı aşamasında bellek ve komşu bilgilerini kapsamlı bir şekilde dikkate alan yeni bir ikili arama operatörü tasarlanmış ve gözcü arı aşamasında diferansiyel evrimin mutasyon ve çaprazlama işlemleri benimsenmiştir. Arama çözümünü optimum hale getirmek için açgözlü stratejiye dayalı bir onarım operatörü kullanılmıştır. Farklı boyutlardaki 0-1 KP verileri üzerinde gerçekleştirilen deneysel sonuçlar, önerilen yöntemin etkinliğini doğrulamakta ve diğer beş metasezgisel algoritmaya kıyasla üstün performans elde ettiğini göstermektedir (Cao ve ark., 2018).

Sosyal Örümcek Algoritması (Social Spider Algorithm, SSA), sosyal örümceklerin avlanacakları avın konumlarını belirlemek için örümcek ağındaki titreşimleri kullanarak sağladıkları yiyecek (av) arama stratejisine dayanan metasezgisel bir optimizasyon algoritmasıdır (James ve Li, 2015). Nguyen ve arkadaşları, SSA'nın yeni bir ikili versiyonu olan İkili Sosyal Örümcek Algoritması (Binary Social Spider Algorithm, BSSA)'nı 0-1 KP'nin çözümü için önermiştir. Önerilen yöntem, ayrık süreç ve kısıtlama işleme sürecinden oluşmaktadır. Ayrık süreçte, iyi bir sonuç elde etmek için sigmoid işlevi kullanılmaktadır. Kısıtlama işleme sürecinde ise, uygun olmayan çözümleri azaltmak ve verimliliği artırmak için ADD-DROP fazlı onarım (repair) operatörü kullanılmıştır. Önerilen yöntem, farklı boyuttaki 0-1 KP veri örnekleri üzerinde test edilmiş ve performans bakımından GA ve PSO ile karşılaştırılmıştır. Sonuçlar, diğer yöntemlere kıyasla BSSA'nın üstün performansını kanıtlamıştır (Nguyen ve ark., 2017).

Rizk-Allah ve Hassanien, 0-1 KP'nin çözümü için yeni bir İkili Yarasa Algoritması (NBBA) önermiştir. Geliştirilen yöntemde ikili yarasa algoritması (BBA) ile yarasaların keşif yeteneğini artırmak ve yerel arama şeması (local search scheme, LSS) ile sömürü eğilimlerini artırmak amaçlanmaktadır. NBBA, literatürdeki farklı büyüklükteki 0-1 KP veri örnekleri üzerinde test edilmiştir. Yapılan deneyler sonucunda, BBA-LSS'nin büyük ölçekli 0-1 KP'nin çözümü için umut verici bir alternatif olabileceği kanıtlanmıştır (Rizk-Allah ve Hassanien, 2018).

WOA, literatüre kazandırıldığı tarihten itibaren birçok problemin çözümünde başarılı bir şekilde kullanılmıştır. 0-1 KP için de sıklıkla başvurulmuş optimizasyon algoritmalarından bir tanesidir. Abdel-Basset ve arkadaşları, hem tek boyutlu hem de çok boyutlu 0-1 KP'nin çözümü için Geliştirilmiş Balina Optimizasyon Algoritması (Improved Whale Optimization Algorithm, IWOA) önermektedir. IWOA'da çözümlerin uygunluğunu değerlendirme fonksiyonuna bir ceza eklenerek yenilik kazandırılmıştır.

Sürekli değerleri ikili değerlere dönüştürmek için ise sigmoid fonksiyonu kullanılmıştır. Uygun olmayan çözümler için iki aşamalı onarım operatörü kullanan IWOA, bunların yanı sıra çeşitlilik ve yoğunlaşma arasında sağlam bir denge kurabilmek için Yerel Arama Stratejisi (LSS) ve Lévy Uçuş Yürüyüşleri (Lévy flight walks) yöntemlerini içermektedir. Önerilen yöntemin, 0-1 KP'nin çözümündeki etkinliğini doğrulamak için farklı son teknoloji algoritmalarla karşılaştırılmıştır. Yapılan deneyler sonucunda, IWOA'nın zorlu 0-1 KP'nin çözümünde kıyaslama yapılan diğer mevcut yaklaşımlardan daha verimli, etkili ve sağlam olduğunu gözlemlenmiştir (Abdel-Basset ve ark., 2019b). Masadeh, 0-1 KP'nin çözümü için WOA ve kümeleme algoritmasına (clustering algorithm) dayalı yeni bir yöntem önermektedir. 0-1 KP veri örnekleriyle yapılan deneyler sonucunda önerilen yöntemin tatmin edici sonuçlar verdiği gözlemlenmiştir (Masadeh, 2021). Alamri ve arkadaşları, 0-1 KP'de optimum çözümler elde etmek için Karşıtlık Tabanlı Balina Optimizasyon Algoritması (Opposition-based Whale Optimization Algorithm, OWOA) önermektedir. Önerilen yöntem farklı 0-1 KP veri örnekleriyle test edilmiş ve sonuçlar orijinal WOA ile diğer metasezgisel yöntemler ile kıyaslanmıştır. Deneysel sonuçlar, OWOA'nın minimum standart sapma değeri ile optimum çözüm ve stabilizasyon bazında önemli bir performansa sahip olduğunu göstermektedir. Bu bağlamda, OWOA'nın orijinal WOA sürümünü geliştirdiğini ve mevcut diğer algoritmalarla karşılaştırıldığında umut verici sonuçlara sahip olduğu kanıtlanmıştır (Alamri ve ark., 2019).

Bahsi geçen metasezgisel algoritmaların yanı sıra, 0-1 KP'nin çözümü için literatüre kazandırılan birçok farklı metasezgisel yöntem bulunmaktadır. Bunlardan bazıları aşağıda maddeler halinde listelenmiştir:

- Balıkçı Arama Prosedürü (Fisherman Search Procedure, FSP) (Cobos ve ark., 2016)
- Küresel Uyum Arama Algoritması (Global Harmony Search Algorithm, NGHS) (Zou ve ark., 2011)
- Karıştırılmış Kurbağa Atlama Algoritması (Shuffled Frog Leaping Algorithm, MDSFL) (Bhattacharjee ve Sarmah, 2014)
- Balçık Küf Algoritması (Slime Mould Algorithm, SMA) (Abdollahzadeh ve ark., 2021)
- Çiçek Tozlaşma Algoritması (Flower Pollination Algorithm, FPA) (Abdel-Basset ve Zhou, 2018; Abdel-Basset ve ark., 2019a; Basheer ve Algamal, 2021)

- Galaktik Sürü Optimizasyonu (Galactic Swarm Optimization, GSO) (Vásquez ve ark., 2020)
- Deniz Avcıları Algoritması (Marine Predators Algorithm, MPA) (Abdel-Basset ve ark., 2021a)
- Denge Optimize Edici (Equilibrium Optimizer, EO) (Abdel-Basset ve ark., 2021b)



3. MATERYAL VE YÖNTEM

Tez kapsamında önerilen IDGKO algoritması, özellik seçimi ve 0-1 sırt çantası

problemleri üzerinde test edilmiştir. Bu başlık altında ilgili problemler hakkında genel bilgilendirme yapılmış ve özellik seçimi problemi için kullanılan deney ortamı hakkında ayrıntılı bilgi sunulmuştur. Ayrıca sürekli optimizasyon problemlerinin çözümünde önerilen ve tezin temel çözüm yöntemi olan GWO'nun orijinal tanımını hakkında temel bilgilendirmeler yer almaktadır.

3.1. Özellik Seçimi Problemi

Veri; günümüzde çevrimiçi eğitim/toplantı/sunum, sosyal medya platformları, sağlık, biyoinformatik gibi sayısız alanda sıklıkla kullanılmakta ve büyük önem taşımaktadır (Li ve ark., 2017a). Bu sebeple içinde bulunduğumuz çağ, büyük veri çağı olarak anılmaktadır. Verilerin sık kullanımı ve her geçen gün veri sayısının artması veri yönetimini daha zor hale getirmektedir. Bunun yanında verilerin hem boyut hem sayı bakımından artması bazı problemleri beraberinde getirmektedir (Li ve ark., 2017a; Agrawal ve ark., 2021). Örneğin; özellikle boyutu yüksek olan veriler, bellek depolama gereksinimi ve hesaplama maliyeti bakımından önemli bir artışa sebep olabilmektedir. Yine, veri sayısının artışı veri analizi ve incelenmesi bakımından karmaşık hale gelerek yanlış işlem riskini daha da artırabilmektedir (Agrawal ve ark., 2021).

Kaynaklardan veya başka şekillerde elde edilen veri kümeleri çok sayıda özelliğe/özniteliğe sahip olabilmektedir. Ancak bu veri kümelerinde yer alan bazı öznitelik/öznitelikler gürültülü adı verilen alakasız ve gereksiz verilerden oluşmaktadır. Gereksiz veriler, sistemin performansını ve doğruluğunu doğrudan etkilemektedir (Agrawal ve ark., 2021; Al-Wajih ve ark., 2021). Bu gibi problemlerin ortaya çıkması bir çözüm yöntemi arayışına sebep olmuş ve özellik seçimi (feature selection, FS) olarak adlandırılan yöntemler geliştirilmiştir.

Veri madenciliği, bilgi teknolojilerinde sıkça bahsedilen bir prensiptir ve büyük ölçekli verilerden bilgiye ulaşılması, bu verilerin yararlı ve verimli bilgiye dönüştürülmesi konusunda çeşitli faaliyetler yürütmektedir (Yongjian, 1997). Özellik seçimi, veri madenciliğinde temel ön işleme adımlarından biridir ve aynı zamanda ikili optimizasyon problemi kapsamında yer almaktadır. Özellik seçimi, performans ve doğruluğu etkilemeden belirli bir veri kümesinden alakasız/gereksiz özellikleri ortadan kaldırarak toplanan verilerin uygunluğunu iyileştirmeyi amaçlayan, veri kümesini en iyi temsil edebilecek alt kümeyi belirleme işlemidir (Dash ve Liu, 1997; Al-Wajih ve ark.,

2021). Özellik seçiminin sağladığı birkaç fayda aşağıda verilmiştir (Kumar ve Minz, 2014):

- Veri setini gereksiz özelliklerden arındırarak algoritmanın hızını artırır.
- Veri setini daha basit ve anlaşılır hale getirir.
- Veri depolama için gerekli olan bellek miktarının en aza indirilmesine yardımcı olur.
- Veri setinden en önemli özellikleri seçerek modelin başarısını artırır.

Özellik seçimi sağladığı avantajlar sebebiyle sınıflandırma ve regresyon problemlerinin yanı sıra birçok gerçek dünya probleminde de kullanılan popüler bir yöntemdir (Bolón-Canedo ve ark., 2015; Remeseiro ve Bolon-Canedo, 2019). Örneğin;

- Biyomedikal alanında bir problem olan, aday gen üzerinden en iyi geni tespit etmek için özellik seçim yöntemleri kullanılmıştır (Ahmed ve ark., 2013; Agrawal ve ark., 2021).
- Renk, piksel vb. bakımdan görüntü analizinde en iyi görsel içeriğini belirlemek için özellik seçim yöntemleri kullanılmıştır (Ghosh ve ark., 2013; Agrawal ve ark., 2021).
- Metin madenciliği/sınıflandırma işlemlerinde en iyi kelime ve deyim terimlerini tespit etmek için özellik seçim yöntemleri kullanılmıştır (Aghdam ve ark., 2009; Agrawal ve ark., 2021).
- Yüz tanıma gibi işlemlerde de özellik seçim yöntemleri kullanılmaktadır (Swiniarski ve Skowron, 2003).

Özellik seçim problemi, matematiksel olarak aşağıdaki şekilde formülize edilebilir (Agrawal ve ark., 2021):

- V isimli bir veri kümesi olduğunu, bu veri kümesinin n adet özneliğe sahip olduğunu ve veri kümesinin sahip olduğu her bir özneliğin $a_1, a_2, a_3, \dots, a_n$ şeklinde temsil edildiğini varsayalım.
- $V = (a_1, a_2, a_3, \dots, a_n)$
- V veri kümesine özellik seçim yöntemi uygulandıktan sonra bu n öznelik arasından en iyi alt küme tespit edilecektir. Bu alt kümeyi de A ile temsil edersek yeni veri kümemiz aşağıdaki gibi olacaktır:

$$A = (a_1, a_2, a_3, \dots, a_k)$$

Burada $n > k$ ' dir.

Özellik seçim yöntemleri, en iyi özellik alt kümesine ulaşmak, değerlendirmek ve seçilen özelliklerin performansını tespit etmek için kendi içerisinde çeşitlenmektedir. Bu yöntemler filtre yöntemleri, sarmalayıcı yöntemler ve gömülü yöntemler olmak üzere üç kategoriden oluşmaktadır.

Filtre yöntemleri, herhangi bir öğrenme veya sınıflandırma algoritması kullanmadan yalnızca veri kümesindeki verilerin genel özelliklerine odaklanarak özelliğin kalitesini (seçilip seçilmemesine) belirlemeyi amaçlayan yöntemlerdir (Li ve ark., 2017a; Agrawal ve ark., 2021). Bu yöntemde veriler teker teker değerlendirilir. Filtre yöntemleri temelde iki ana adımdan oluşmaktadır. İlk olarak veri kümesindeki özellikler belirlenen özellik değerlendirme kriterine bağlı olarak önem derecesine göre sıralanır. Sıralama sonrasında düşük dereceye sahip özellikler filtrelenir (Li ve ark., 2017a). Filtrelemede kullanılan yöntemler genellikle veri kümesindeki tüm özelliklerin arasından en yüksek önem derecesine sahip özelliklerin seçilmesine dayanmaktadır (Al-Wajih ve ark., 2021). Filtre yöntemlerinden bazıları veri kümesindeki özellikleri önem derecesine göre sıralamakta ki-kare analizi, kaba küme ve bulanık-kaba küme tabanlı bağımlılık, bilgi kazancı ve simetrik belirsizlik, grup tabanlı bulanık-kaba, olasılıksal tutarlılık tabanlı ve korelasyon tabanlı gibi belirli sıralama kriterlerini kullanmaktadır (Kalaycı, 2019). Korelasyona dayalı özellik seçim stratejisi, boyut küçültme teknikleri ve sinir ağları gibi bazı yöntemler, sistem karmaşıklığını ve hesaplama yükünü azaltmada etkili yöntemlerdir (Al-Wajih ve ark., 2021). Filtre yöntemleri, hesaplama açısından daha verimlidir ancak veri kümesinden alt kümeyi seçme aşamasında herhangi bir öğrenme veya sınıflandırma algoritması kullanmaması sebebiyle seçilen özellikler optimum olmayabilmektedir (Li ve ark., 2017a; Remeseiro ve Bolon-Canedo, 2019).

Sarmalayıcı yöntemler, filtre yöntemlerinin aksine bir öğrenme veya sınıflandırma algoritması kullanılarak özellik seçiminin yapıldığı ve seçimlerin değerlendirildiği yöntemlerdir (Li ve ark., 2017a; Agrawal ve ark., 2021). Burada amaç; oluşturulan her bir alt kümenin doğruluğunu değerlendirerek daha iyi sonuç için hareket etmektir. Bu yöntemde veriler teker teker değil; alt kümeler halinde grup olarak değerlendirilir (Özger, 2019). Sarmalayıcı yöntemler temelde iki ana adımdan oluşmaktadır. İlk olarak belirli bir kural çerçevesinde veya rastgele bir şekilde alt kümeler oluşturulur. Ardından oluşturulan özellik alt kümesi bir öğrenme veya sınıflandırma algoritması kullanılarak değerlendirmeye alınır. Bu iki adım sonlanma kriteri sağlanana kadar devam ettirilir. Sonlanma kriteri sağlanana kadar elde edilen alt kümelerden değerlendirme sonucu en iyi performansa sahip alt küme, seçilen özellikler olarak

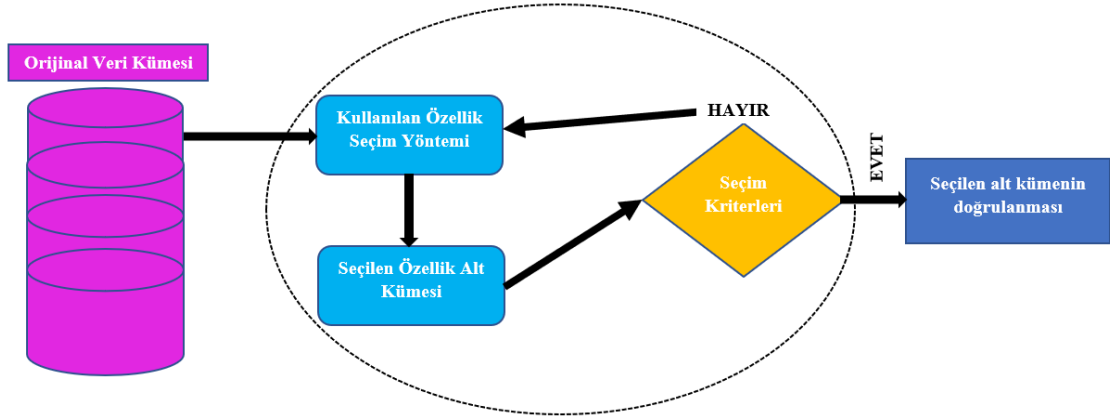
döndürülür (Li ve ark., 2017a). Sarmalayıcı yöntemler, öğrenme veya sınıflandırma algoritması kullanması sebebiyle filtre yöntemlerine göre daha iyi sonuçlar elde ederken; yine aynı sebeple hesaplama maliyeti filtre yöntemlerine göre daha fazladır (Remeseiro ve Bolon-Canedo, 2019; Agrawal ve ark., 2021). Sarmalayıcı yöntemler ile özellik seçiminde sıklıkla kullanılan bazı öğrenme veya sınıflandırma algoritmaları örnekleri şunlardır (Al-Wajih ve ark., 2021): karar ağacı, destek vektör makineleri, Naive Bayes, K-en yakın komşu, yapay sinir ağları, lineer diskriminant analizi, yerel geometrik yapı Fisher analizi ve yerel komşuluk yapısı koruma gömme.

Gömülü yöntemler, filtre ve sarmalayıcı yöntemlerin bir birleşimi olarak düşünülebilir. Temel fikri; hesaplama açısından maliyeti en az olan filtreyi kullanarak gereksiz bazı özellikleri yok etmek, ardından hesaplama açısından daha az maliyetli sarmalayıcı yöntem kullanmaktır (Remeseiro ve Bolon-Canedo, 2019). Veri kümesinden performansı en iyi alt kümeyi belirleme işlemi, sınıflandırıcı eğitimi aşamasında gerçekleştirilir. Örnek olarak; bir yapay sinir ağı düşündüğümüzde alt küme belirleme işleminin ağırlıkları optimize ederken gerçekleştirildiği verilebilir (Remeseiro ve Bolon-Canedo, 2019). Alt küme değerlendirme işlemi sarmalayıcı yöntemlere nazaran sürekli tekrarlanmadığı için gömülü yöntemler, sarmalayıcı yöntemlerden daha az hesaplama maliyetine sahiptir (Li ve ark., 2017a; Remeseiro ve Bolon-Canedo, 2019).

Kısaca özetleyecek olursak; özellik seçim süreci temelde beş ana unsur içermektedir. Bunlar aşağıda listelenmiştir:

- 1) Orijinal veri kümesi
- 2) Özellik seçimi sonucu elde edilen alt küme
- 3) Elde edilen özellik alt kümesinin değerlendirme işlemi
- 4) Özellik seçerken uygulanacak seçim kriterleri
- 5) Seçilen alt kümeyi doğrulama işlemi

Yukarıdaki özetlenen özellik seçim sürecinin çalışma mekanizmasını gösteren şema Şekil 3.1’de sunulmuştur.



Şekil 3.1. Özellik seçim sürecinin çalışma mekanizması (Agrawal ve ark., 2021)

3.2. Özellik Seçimi Problemi için Deneysel Veri Setleri

Deneysel veriler sırasında kullanılan veri setleri, Kaliforniya Üniversitesi Irvine Makine Öğrenmesi Veritabanı'ndan (UCI Machine Learning Repository) (Repository, 2021/2022) alınarak, önerilen yöntemin FS'nin çözümündeki performansını değerlendirmek için kullanılmıştır. Veri setleri, farklı sayıda örnek ve özelliğe sahiptir. Veri setlerinin farklı sayıda örnek ve özelliğe sahip olması, önerilen yöntemin farklı koşullar altında test edilerek performansının ölçülmesini sağlayacaktır. Deneysel verilerde kullanılan veri setlerine ait bilgiler Çizelge 3.1'de sunulmuştur.

Çizelge 3.1. Deneylerde kullanılan veri setlerine ait bilgiler

Veri Seti ID	Veri Seti	Özellik Sayısı	Örnek Sayısı	Sınıf Sayısı
F1	Breast Cancer	9	286	2
F2	Breast Cancer Wisconsin Diagnostic (BreastEW)	30	569	2
F3	Breast Cancer Wisconsin Original (Breastcancer)	10	699	2
F4	Chess (King-Rook vs. King-Pawn, KrvskpEW)	36	3196	2
F5	Congressional Voting Records (CongressEW)	16	435	2
F6	Connectionist Bench (SonarEW)	60	208	2
F7	Exactly	13	1000	2
F8	Exactly2	13	1000	2
F9	Ionosphere	34	351	2
F10	Lymphography	18	148	4
F11	M-of-n	13	1000	2
F12	PenglungEW	73	325	7
F13	Spect Heart (SpectEW)	22	267	2
F14	Statlog Heart (HeartEW)	13	270	2
F15	Tic Toc Tae	9	958	2
F16	Vote	16	300	2
F17	Waveform Database Generator (WaveformEW)	40	5000	3
F18	WineEW	13	178	3
F19	Zoo	17	101	7

3.3. Özellik Seçimi için Uygunluk Fonksiyonu

Sürü zekasına dayalı ve yinelemeli algoritmalar, popülasyonun kalitesini artırmak için bir döngü içinde çalışır. Popülasyondaki bir çözümün kalitesi, kullanılan problemle ilgili bir uygunluk fonksiyonu (fitness function) kullanılarak hesaplanır. Sonunda, algoritma döngü içerisinde o ana kadar bulunan en iyi çözümü çıktı olarak sunar. Minimizasyon problemlerinde uygunluk fonksiyonunun minimize edilmesi amaçlanır. GWO'daki bireylerin elde ettiği çözümlerin özellik seçimi probleminde uygunluğunu belirlemek için, literatürdeki çalışmalar ve avantajları göz önünde bulundurularak tez kapsamında uygunluk değerlendirmesi KNN sınıflandırıcısı kullanılarak yapılmıştır. Bahsi geçen uygunluk fonksiyonu Denklem (3.1)'de sunulmuştur.

$$FitnessValue = F = ErrNum/N_t \quad (3.1)$$

Denklemden $ErrNum$, KNN ile yanlış sınıflandırılan örneklerin sayısını ve N_t , test örneklerinin toplam sayısını temsil etmektedir.

3.4. Özellik Seçimi Problemi için Performans Belirleme Ölçütleri

Önerilen yöntemin tutarlılığını belirlemek adına gerçekleştirilen birbirinden bağımsız çalıştırmalar sonucunda elde edilen çözümleri değerlendirmek için farklı performans belirleme ölçütleri kullanılmaktadır. Tez kapsamında kullanılan performans belirleme ölçütleri ve matematiksel denklemler aşağıda başlıklar halinde sunulmuştur.

3.4.1. Sınıflandırma Doğruluğu

Sınıflandırma doğruluğu (*Acc*), veri kümesinden ilgili özellikleri seçmek için kullanılan sınıflandırıcının tutarlılığını belirlemek için kullanılır. Algoritmanın N kez çalıştırıldığı varsayıldığında *Acc*'nin matematiksel denklemi Denklem (3.2)'de verilmiştir.

$$Acc = 1 - FitnessValue \quad (3.2)$$

Denklemden, *FitnessValue* değeri Denklem (3.1)'den elde edilen uygunluk fonksiyonu değeridir.

3.4.2. Uygunluk Fonksiyonu Değerleri

İterasyonlar sonucunda elde edilen uygunluk fonksiyonu değerleri en iyi (*best*) ve ortalama (*average*) olarak değerlendirilebilir. Bahsi geçen değerlerin matematiksel gösterimleri Denklem (3.3) ve (3.4)'te sunulmuştur.

$$Best = \min_{t=1}^N F_t \quad (3.3)$$

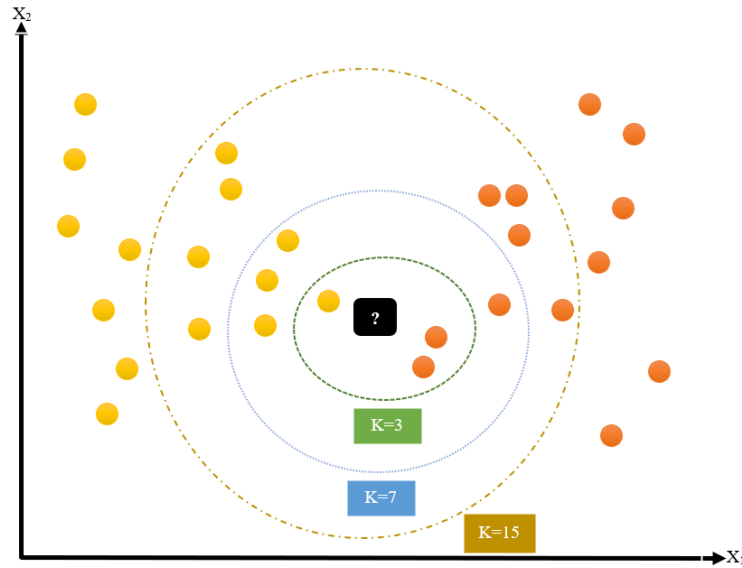
$$Average = \frac{\sum_{t=1}^N F_t}{N} \quad (3.4)$$

Denklemden F_t , t 'inci çalışma için elde edilen uygunluk fonksiyonu değeridir.

3.5. K- En Yakın Komşu Algoritması

Bu tez çalışmasında özellik seçimi problemi çözümünde sarmalayıcı yöntemler uygulanmış ve bu doğrultuda K- En Yakın Komşu algoritması (K-Nearest Neighbors Algorithm, KNN) kullanılmıştır.

KNN, sınıflandırma ve regresyon için sıklıkla kullanılan basit yapılı bir sınıflandırma algoritmasıdır. Sınıflandırma algoritmaları, veri kümesinde yer alan ve sınıf bilgisi bilinen verilerin dağılımlarından, özelliklerinden vb. yola çıkarak bir model elde etme amacıyla kullanılır. Elde edilen model daha sonra sınıf bilgisi olmayan yeni örnekler için sınıf tahmininde kullanılır (Özger, 2019). KNN algoritması örnekleri benzerliklerine göre sınıflandıran ve mesafeye dayalı parametrik olmayan bir yöntemdir. Parametrik olmayan yöntemler veri dağılımı ile ilgili bir varsayımda bulunmamaktadır ve kullanılan algoritma bir eğitim modeli ortaya çıkarmamaktadır (Aydın, 2019; Özger, 2019). KNN algoritmasında herhangi bir sınıf bilgisi bulunmayan örnek, mevcut örnekler arasından mesafe yöntemlerinden faydalanarak kendisine en yakın yani en az mesafeye sahip k adet örneğin sınıf bilgisine göre sınıfını belirler (Emary ve ark., 2016a). KNN algoritmasının çalışma mantığına ait bir görsel Şekil 3.2’de verilmiştir.



Şekil 3.2. KNN algoritmasının çalışma mantığı şeması

Şekil 3.2’de ‘?’ ile ifade edilen örnek sınıf bilgisi olmayan örnektir. Sarı renkle ifade edilen örnekler sınıf değeri belli olan ve aynı kategoride bulunan örneklerdir. Turuncu renkle ifade edilen örnekler sınıf değeri belli olan ve aynı kategoride bulunan örneklerdir. (Sarı ve turuncu renkli örneklerin sınıf değerleri farklıdır.) Belirlenen k

değerine ve mesafe ölçme yöntemine göre algoritma yürütülerek sınıf bilgisi olmayan örneğin sınıf bilgisi elde edilir. Örneğin $k=3$ iken, ‘?’ ile ifade edilen örneğin iki turuncu ve bir sarı komşusu bulunmaktadır. Bu sebeple örnek turuncu renkli sınıf kategorisine dahil edilir. Örneğin $k=7$ iken, ‘?’ ile ifade edilen örneğin üç turuncu ve dört sarı komşusu bulunmaktadır. Bu sebeple örnek sarı renkli sınıf kategorisine dahil edilir. Örneğin $k=15$ iken, ‘?’ ile ifade edilen örneğin yedi turuncu ve sekiz sarı komşusu bulunmaktadır. Bu sebeple örnek sarı renkli sınıf kategorisine dahil edilir.

Örnekler arasındaki mesafeyi belirlemek için çeşitli yöntemler mevcuttur. Bunlar arasında en popüler olanları Öklid Uzaklığı (Euclidean Distance), Manhattan, Mahalonobis, Minkowski ve Hamming yöntemleridir (Fix ve Hodges, 1989; Singh ve Pandey, 2016). Literatürde yapılan çalışmalar arasında da sıklıkla kullanılan ve basit bir yöntem olan Öklid Yöntemi, bu tez çalışmasında mesafe belirleme yöntemi olarak belirlenmiştir. Bahsi geçen Öklid mesafe yönteminin matematiksel formülü Denklem (3.5)’te sunulmuştur (Singh ve Pandey, 2016).

$$\text{Öklid Uzaklığı} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3.5)$$

Algoritmaya adını veren k parametresi performans açısından oldukça önemli bir role sahiptir ve bu değer belirlenmesi sürecinde hassas davranılmalıdır. k değeri için genellikle tek sayılar tercih edilmektedir (Özger, 2019). Ancak bu değeri belirlemek için halen bir kesin çözüm bulunmamaktadır ve kendi içerisinde k değerinin belirlenmesi de bir optimizasyon problemidir.

3.6. 0-1 Sırt Çantası Problemi

Günümüzde birçok işlemin, karar alma ve verme aşamasında elde bulunan verilere dayanarak matematiksel bir model veya yöntem geliştirilmesine ihtiyaç duyulmaktadır (Taha, 2000). Bu yöntemlerden birisi olan tamsayılı programlama; askeri, iktisat, işletme, endüstri, mühendislik gibi daha birçok bilim alanında karar değişkeni bakımından kesikli değerler içeren problemleri bünyesinde barındırmaktadır (Uygun, 2010). Sırt çantası problemi (KP), sonlu bir nesne kümesi içerisinde en uygun nesne/nesneleri tespit etmeye dayanan kombinatoriyal optimizasyon problemlerinden olup tamsayılı doğrusal programlama problemleri içerisinde değerlendirilmektedir.

(Saraç, 2007; Roayaei, 2021). KP için Dantzig'in yaptığı öncü çalışmalar ışığında günümüze kadar birçok bilim insanı tarafından çalışmalar yapılmış ve literatüre kazandırılmıştır (Dantzig, 1957; Zou ve ark., 2011; Bansal ve Deep, 2012; Hakli, 2020). Sırt çantası problemleri genel tanımı doğrultusunda zaman içerisinde etki alanını genişleterek her biri standart bir problem olarak tanımlanan alt alanlara ayrılmıştır. Bunlardan bazıları; 0-1 (binary) sırt çantası problemleri, sınırlı sırt çantası problemleri (bounded knapsack problems), sınırsız sırt çantası problemleri (unbounded knapsack problems), çok boyutlu sınırlı sırt çantası problemleri (multidimensional knapsack problems), çoklu sırt çantası problemleri (multiple knapsack problems), çok seçenekli sırt çantası problemleri (multiple-choice knapsack problems), karesel sırt çantası (quadratic knapsack problems) problemleridir (Uygur, 2010). Bu tez çalışması kapsamında yalnızca 0-1 sırt çantası problemi üzerine çalışmalar yapılmıştır.

0-1 sırt çantası problemi, sırt çantası problemlerinin arasında en çok çalışma konusu olan ve genellikle sırt çantası problemi denilince akla ilk gelen problem çeşididir. Genel olarak tanımı şu şekildedir (Zou ve ark., 2011; Hakli, 2020; Roayaei, 2021): Elde mevcut N adet nesne ve W ağırlığında bir sırt çantası olduğunu varsayalım. N adet nesnenin de kendine ait bir ağırlık değeri (w) ve kâr (fayda, p) değeri bulunmaktadır. Problemin amacı; verilen sırt çantası kapasitesini (ağırlığını) aşmadan uygun nesnelere seçerek sırt çantasının toplam kârını maksimum seviyeye ulaştırmaktır. Problemin 0-1 şeklinde isimlendirilmesinin sebebi ise ilgili nesnenin sırt çantasında yer alması (1) veya yer almamasını (0) ifade etme biçimidir. 0-1 sırt çantası problemi yapısal olarak oldukça basit ve kolay anlaşılabilir bir problemdir. Bahsi geçen problemin matematiksel formülü Denklem (3.6)'da verilmiştir.

$$\begin{cases} \text{Maksimum } f(x) = \sum_{i=1}^N x_i * p_i \\ \text{Kısıtlar } \sum_{i=1}^N x_i * w_i \leq W & x_i \in \{0,1\}, \quad i = 1,2,3, \dots, N \end{cases} \quad (3.6)$$

Denklemden x_i , i nesnesinin sırt çantasında yer alıp/almadığını ve p_i , i nesnesinin kâr değerini temsil etmektedir. Burada nesnelere ait tüm kâr ve ağırlık değerleri negatif değer olmamakla birlikte; nesne ağırlıkları sırt çantası kapasite değerinden küçük değerler olarak tanımlanmaktadır.

KP, günlük hayatta birçok farklı uygulama alanında kendine özgü isimlerle anılmaktadır. Uygulama alanının genişliği sebebiyle farklı alt alanlarda kullanılarak önemini korumaktadır. Kütük/tahta kesme problemleri, kargo yükleme problemleri,

sermaye bütçeleme problemleri, bagaj yükleme problemleri, portföy seçimi ve sırt çantası şifreleme sistemleri gibi alt uygulama alanlarında problem çözümü için KP yapısından faydalanılmaktadır (Saraç, 2007; Berberler, 2009; Uygur, 2010).

3.6.1. Onarım Fonksiyonu

0-1 KP'de problem kısıtı olarak yer alan, bu sebeple dikkat edilmesi gereken bir husus vardır. Bu husus, bireyler tarafından elde edilen çözümlerin uygunluk değerleri hesaplanırken, ilgili değerlerin sırt çantası kapasitesini aşmaması gerektiğidir. Şayet böyle bir çözüm elde edilir ise problem için belirlenen kısıt ortadan kalkmış ve çözüm kabul edilemez olacaktır (Abdel-Basset ve ark., 2021a). Sırt çantası problemlerinin genelinde yaşanabilen bu tür problemler için uygun olmayan çözümleri iyileştirmek ve uygun çözümlerin kalitesini artırmak amacıyla bazı onarım fonksiyonları (Repair Function, RF) geliştirilmiştir (Hakli, 2020). Bu bağlamda, Xiang ve arkadaşları tarafından sırt çantası problemlerinde ortaya çıkan bu tarz sorunların çözümü için iki aşamalı bir RF önerilmiştir (Xiang ve ark., 2014). Bu yöntemin amacı, uygun olmayan çözümleri iyileştirmek ve uygun çözümlerin kalitesini artırmaktır. 0-1 KP'nin çözümü için önerilen IDGKO'da, yerel minimumlardan kaçınmak ve yerel-global arama yeteneğini artırarak kararlı çözümler elde etmek için iki aşamalı RF kullanılmıştır. İki aşamalı onarım fonksiyonunun sözde kodu Şekil 3.3'te verilmiştir.

```

İlgili çözümü belirle ( $X$ )
 $X$  için sırt çantasının toplam ağırlık değerini hesapla  $toplamAğırlık = X * W_k$ 
if  $toplamAğırlık > SırtÇantasıKapasitesi$ 
  for  $j=1$  to  $N$ 
    Yuklenen öğelerin kar yoğunluğu değerini hesapla
     $A_j = (P_j/W_j) * X_j$ 
  end for
  Öğeleri artan  $A_j$  sırasına göre sırala
   $SA_j =$  Sıralı Sonuç Vektörü
   $ind_j =$  her  $SA_j$ 'nin orijinal dizisi
  for  $j=1$  to  $N$ 
    if  $SA_j == 0$ 
      continue
    end if
    Sırt çantasından  $ind_j$ 'th nesnesini çıkart
     $X_{ind_j} = 0$ 
     $toplamAğırlık = X * W_k$ 
    if  $toplamAğırlık \leq SırtÇantasıKapasitesi$ 
      break
    end if
  end for
end if
 $toplamAğırlık = X * W_k$ 
if  $toplamAğırlık < SırtÇantasıKapasitesi$ 
  Kar yoğunluk oranını ( $u_j$ ) hesapla
   $u_j = P_j/W_j$ 
  Tüm öğeleri azalan  $u_j$  sırasına göre sırala
   $Su_j =$  Sıralı Sonuç Vektörü
   $ind_j =$  her  $Su_j$ 'nin orijinal dizisi
  for  $j=1$  to  $N$ 
    if  $X_{ind_j} = 0$ 
       $X^* =$  Ara Çözüm Vektörü =  $X$ 
       $ind_j$ 'th öğesini çantaya eklemeyi dene
      Set  $X_{ind_j} = 1$ 
      if  $X^* * W_k \leq SırtÇantasıKapasitesi$ 
         $ind_j$ 'th öğesini çantaya ekle
         $X_{ind_j} = 1$ 
      end if
    end if
  end for
end if
end if

```

Çıkarma Aşaması

Ekleme Aşaması

Şekil 3.3. İki Aşamalı Onarım Fonksiyonu Algoritmasının Sözde Kodu (Xiang ve ark., 2014)

3.7. Gri Kurt Optimizasyon Algoritması

Gri Kurt Optimizasyon Algoritması (Grey Wolf Optimizer, GWO), 2014 yılında Mirjalili tarafından sürekli optimizasyon problemlerinin çözümü için önerilen sürü zekasına dayalı metasezgisel bir optimizasyon algoritmasıdır (Mirjalili ve ark., 2014b). Algoritmaya ismini veren gri kurtlar besin zincirinin en üst kısmında yer alan ve sürü halinde yaşayan bir canlı grubudur.

GWO, gri kurtların sosyal hiyerarşisinden ve avlanma mekanizmasından esinlenerek geliştirilmiştir. Doğadaki gri kurtlar, oldukça katı bir hiyerarşik yapıyı benimsemektedir. Bu hiyerarşik yapı, sürüye yön veren üç lider birey (kurt) ve geriye kalan bireyler olmak üzere dört temel isimlendirmeden oluşmaktadır.

Alfa kurt (alpha, α), hiyerarşik yapının ilk seviyesinde yer almakta olup aldığı kararlar bakımından tüm sürünün itaat ettiği üç lider bireyden biridir. Bu sebeple baskın kurt olarak da bilinmektedir (Muro ve ark., 2011; Mirjalili ve ark., 2014b). Alfa kurt uyuma ve uyanma zamanlarını belirleme, uygun uyuma yeri bulma, avlanma gibi

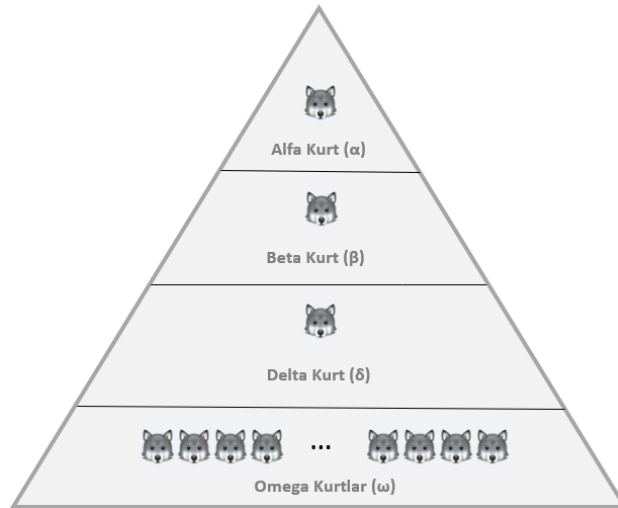
konularda karar vermekten sorumludur. Kısacası, sürünün temel organizasyon ve disiplini için yönetimi sağlayan temel birey alfa kurtlardır.

Beta kurtlar (β), öncü bireyler arasında olup hiyerarşik yapının ikinci seviyesinde yer almaktadır. Temel görevi, sürünün temel organizasyon ve disiplini için karar mekanizması olan alfa kurtlara yardımcı olmaktır.

Delta kurtlar (δ), öncü bireyler kategorisinin üçüncü ve son seviyesinde yer almaktadır. Delta kurt kategorisi avcı, bakıcı, izci, nöbetçi ve yaşlı gibi farklı görevlere sahip kurtlardan oluşmaktadır. Avcı kurtlar, avlama esnasında ve sürüye yiyecek temin etme noktasında alfa ve beta kurtlara yardımcı olmakla yükümlüdür. Bakıcı kurtlar, sürüdeki hasta, yaralı ve güç anlamında zayıf düşmüş kurtların temel bakımından sorumludur. İzci kurtlar, sürünün yer aldığı bölgenin sınırlarını korumak ve herhangi bir tehlike durumunda sürüyü uyarmakla görevlidir. Nöbetçi kurtlar, sürünün güvenliğini sağlamakla görevlidir. Son olarak yaşlı kurtlar ise geçmişte alfa veya beta kurt olarak görev yapmış kurtlardır.

Öncü birey kategorisinin dışında yer alan sürü mensubu bireyler omega bireyler (ω) olarak adlandırılmaktadır. Omega bireyler, lider bireylerin aldığı kararlara boyun eğmekle yükümlü bireylerdir.

GWO'daki gri kurtların sosyal hiyerarşik yapısı Şekil 3.4'te verilmiştir.



Şekil 3.4. Gri kurtların sosyal hiyerarşik yapısı (Mirjalili ve ark., 2014b; Karakoyun ve ark., 2021)

GWO'nun gelişiminde rol oynayan diğer bir ilham kaynağı, gri kurtların avlanma mekanizmasıdır. Avlanma mekanizması avı çevreleme, avlanma ve ava saldırma olmak üzere üç temel adımdan oluşmaktadır (Muro ve ark., 2011; Mirjalili ve ark., 2014b).

Avı Çevreleme: Gri kurtlar av esnasında ilk olarak avın etrafını sararak bir kalkan modeli oluştururlar. Bahsi geçen bu eylemin matematiksel formülü Denklem (3.7) ve (3.8)'de verilmiştir.

$$\vec{D} = |\vec{C} \times \vec{X}_p(t) - \vec{X}(t)| \quad (3.7)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \times \vec{D} \quad (3.8)$$

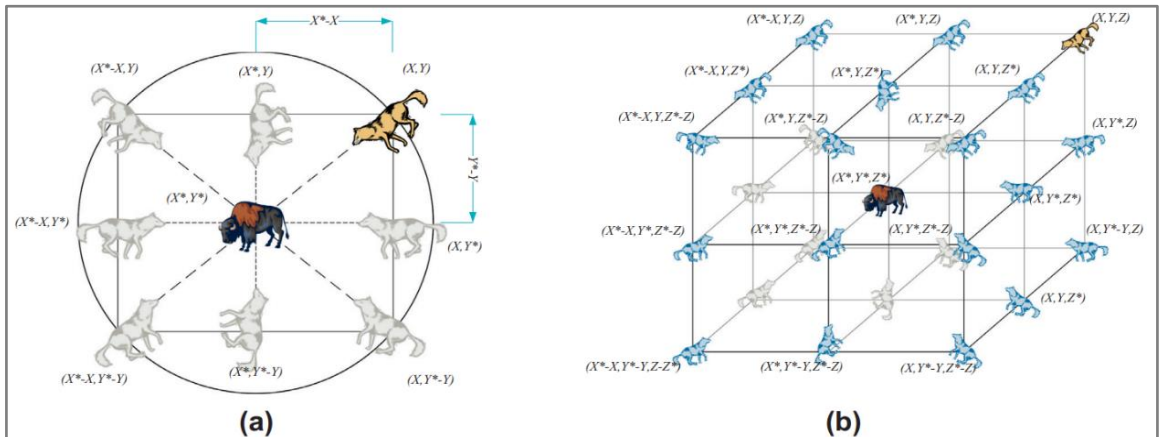
Verilen denklemlerde t iterasyon sayısını, \vec{X}_p avın konumunu ve \vec{X} sürüdeki ilgili bireyin mevcut konumunu temsil etmektedir. \vec{A} ve \vec{C} katsayı vektörleri olarak adlandırılmakta olup matematiksel formülleri sırasıyla Denklem (3.9) ve (3.10)'da verilmiştir.

$$\vec{A} = 2\vec{a} \times \vec{r}_1 - \vec{a} \quad (3.9)$$

$$\vec{C} = 2\vec{r}_2 \quad (3.10)$$

Verilen denklemlerdeki \vec{a} değeri belirlenen iterasyon sayısı tamamlanana dek -2'den 0'a doğru doğrusal olarak azalan bir değişken, \vec{r}_1 ve \vec{r}_2 ise $[0,1]$ aralığında rastgele üretilen sayılardır.

Avı çevreleme için geliştirilen Denklem (3.7) ve (3.8) sonucunda bir bireyin bir sonraki adımda sahip olabileceği olası konumlar ve formülasyondaki katsayı vektörleri, \vec{a} , \vec{r}_1 ve \vec{r}_2 gibi değişkenlerin konum belirlemedeki etkisi iki boyutlu ve üç boyutlu olarak Şekil 3.5'te sunulmuştur.



Şekil 3.5. Bir bireyin avı çevreleme sonucunda sahip olabileceği olası konumların iki boyutlu (a) ve üç boyutlu (b) gösterimi (Mirjalili ve ark., 2014b)

Ava Saldırma: Ava saldırma süreci temelde iki durum barındırmaktadır. İlk durumda, gri kurtlar hareket etmeyi durdurarak keşfettikleri ava saldırır ve avlanma sürecini bitirirler. Diğer durumda ise bireyler yeni av arayışına girerler. Ava saldırma aşamasında hangi durumun gerçekleşeceğine dair karar vermede \vec{A} değişkeni etkilidir. Ava yaklaşma eyleminin matematiksel modellenmesinde \vec{a} değeri iterasyona bağlı olarak -2'den 0'a doğru azalmaktadır. Burada dikkat çeken bir nokta; Denklem (3.9)'da \vec{A} değerinin \vec{a} değerine bağlı olarak değişmesidir. Bu sebeple \vec{A} değeri, $[-2\vec{a}, 2\vec{a}]$ aralığında rastgele değerler üreterek ava saldırma veya yeni av arama durumlarının hangisine karar verileceğini belirlemektedir. Burada, " $|\vec{A}| < 1$ ise ava saldır, $|\vec{A}| > 1$ ise yeni av ara" şeklinde bir karar mekanizması bulunmaktadır (Mirjalili ve ark., 2014a; Heidari ve Pahlavani, 2017; Karakoyun ve ark., 2021).

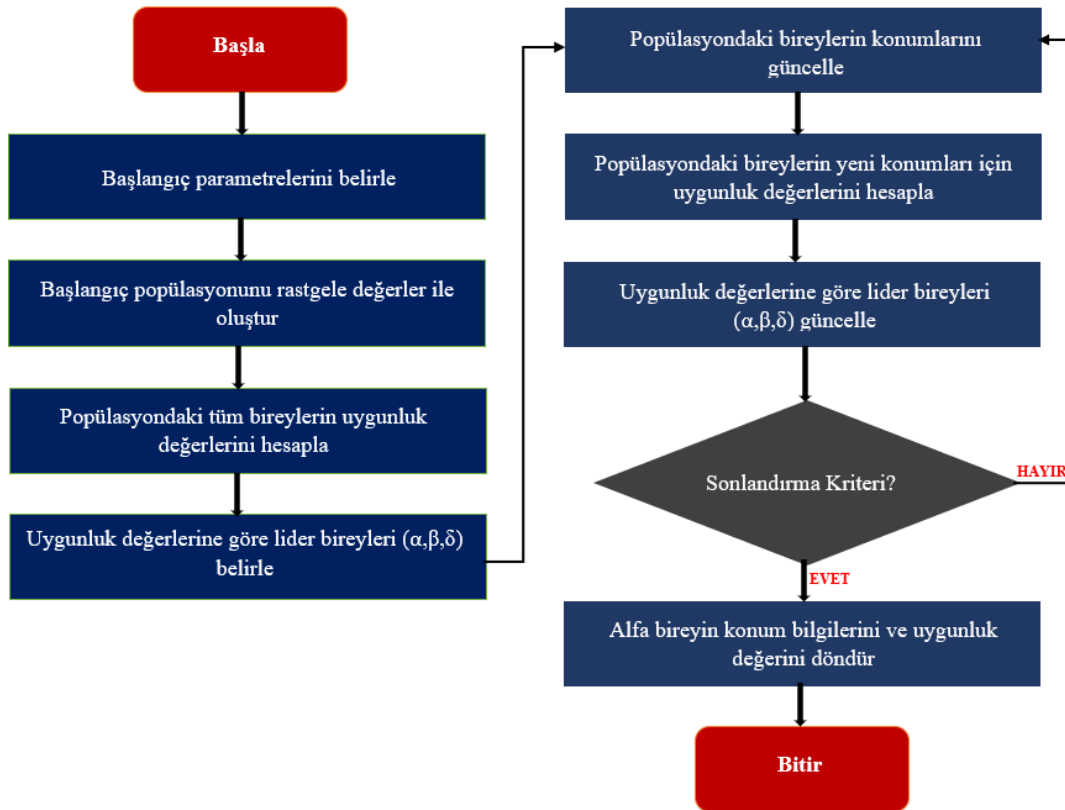
GWO algoritmasının sözde kodu Şekil 3.7'de ve akış şeması Şekil 3.8'de sunulmuştur.

```

Başlangıç parametrelerini belirle
Başlangıç popülasyonunu rastgele değerler ile oluştur
Popülasyondaki tüm bireylerin uygunluk değerlerini hesapla
Uygunluk değerlerine göre lider bireyleri belirle
     $X_\alpha =$  Alfa Kurt = en iyi uygunluk değerine sahip gri kurt
     $X_\beta =$  Beta Kurt = ikinci en iyi uygunluk değerine sahip gri kurt
     $X_\delta =$  Delta Kurt = üçüncü en iyi uygunluk değerine sahip gri kurt
while (t < Sonlandırma Kriteri)
     $\vec{a}$  değerini güncelle
    for popülasyondaki tüm bireyler için
         $\alpha$ ,  $\beta$  ve  $\delta$ 'ya göre ilgili bireyin konumunu güncelle
        İlgili bireyin uygunluk değerini hesapla
    end for
    Uygunluk değerlerine göre lider birey bilgilerini güncelle
    t++
end while
 $X_\alpha$  bilgilerini döndür ve algoritmayı sonlandır

```

Şekil 3.7. GWO algoritmasının sözde kodu (Karakoyun ve ark., 2019)



Şekil 3.8. GWO algoritmasının akış şeması (Mirjalili ve ark., 2014b; Karakoyun, 2021)

GWO, basit ancak oldukça etkili bir optimizasyon algoritmasıdır (Mirjalili ve ark., 2014b; Karakoyun, 2021). Şekil 3.7 ve 3.8'e bakıldığında algoritmanın işleyişi ilk olarak parametre ayarlarının yapılmasıyla başlamaktadır. Ardından arama uzayında optimum değeri bulmak için gerekli bireylerden oluşan popülasyon oluşturma ve ilk değer atama işlemleri gelmektedir. Sürü zekasına dayanan diğer algoritmaların çoğunda olduğu gibi GWO'da da ilk popülasyon (başlangıç popülasyonu) oluşturma işlemi belirlenen sınırlar aralığında rastgele değerler üretilerek yapılmaktadır. Ardından popülasyondaki her bir birey için uygunluk değeri hesaplanır ve en iyi ilk üç değere sahip birey konum bilgileri sırasıyla alfa, beta ve delta birey olarak güncellenir. Bu sayede hem ilk popülasyon hem de ilk lider bireyler belirlenmiş olur ve algoritma için ana döngü başlatılır. Popülasyondaki tüm bireyler önceki kısımlarda bahsedilen denklemler kullanılarak konum güncelleme işlemine tabii tutulur. Bu işlemden sonra yine tüm bireyler için uygunluk değeri hesaplanır ve en iyi üç değere göre alfa, beta, delta bireylerin konumları güncellenir. Ana döngünün sonlanma şartı sağlanana kadar bu adımlar gerçekleştirilir. Ana döngü sonlandığında popülasyondaki en iyi uygunluk değerine sahip alfa bireyin konum bilgileri ve uygunluk değeri optimum değerler olarak belirlenir ve algoritma sonlandırılır (Karakoyun ve Özkış; Mirjalili ve ark., 2014b; Karakoyun, 2021).

4. ÖNERİLEN YÖNTEM

Bu bölümde tez kapsamındaki problemlerin çözümü için önerilen yeni İkili Dinamik Gri Kurt Optimizasyon Algoritması (IDGKO) hakkında temel bilgilendirmeler yer almaktadır.

4.1. İkili Dinamik Gri Kurt Optimizasyon Algoritması

Orijinal GWO, sürekli optimizasyon problemlerinin çözümü için önerildiğinden algoritmada yer alan aritmetik işlemler doğrudan ikili arama uzayında kullanılamamaktadır. Bu sebeple önceki bölümlerde bahsi geçen ve tez kapsamında önerilen lojik-kapı tabanlı ikili operatörler kullanılarak algoritma ikili optimizasyon problemlerinin çözümü için uyarlanabilmektedir. Burada, orijinal algoritmada yer alan aritmetik işlemler yerine doğrudan ikili bitsel işlemler kullanılmaktadır.

Tez kapsamında ikili optimizasyon problemlerinin çözümü için önerilen IDGKO'nun geliştirilme sürecinde temel iki ilham kaynağı çalışma bulunmaktadır:

- Bunlardan ilki, (Ozsoydan, 2019) tarafından yapılan GWO temelli bir çalışmada “*GWO'nun performansı, popülasyondaki baskın üç bireyin performansı ile doğrudan ilgilidir. GWO da dahil olmak üzere metasezgisel algoritmaların çoğu kendi aralarında benzerlikler gösterse de GWO'nun bu özelliği ayırt edici mekanizma olarak kabul edilebilir. Bu açıdan, literatürde yeterince tartışılmayan, baskın kurtların çözüm kalitesi üzerindeki etkilerinin incelenmesi daha fazla araştırmayı hak etmektedir.*” şeklinde vurgulanan ifadedir.
- İkincisi ise, (Karakoyun, 2021; Karakoyun ve ark., 2021) tarafından yapılan GWO temelli çalışmalarda “*Lider ekipteki bireylerin çözüm kalitelerine göre pozisyon üzerindeki etkilerinin belirlenip uygulanması, etkilenen çözümü daha iyi bir konuma getirir mi?*” sorusuna cevap bulmak için önerilen dinamik katsayı yaklaşımı (dynamic coefficient approach, DCA)'dır.

Bu iki çalışmadan ilham alarak tez kapsamında, baskın kurtların çözüm kalitesi üzerindeki etkilerini artırmayı hedefleyen ve dinamik katsayı yöntemini kullanan yeni bir İkili Dinamik Gri Kurt Optimizasyon Algoritması (Binary Dynamic Grey Wolf Optimizer, IDGKO) önerilmiştir.

Lojik-kapı tabanlı dört adet ikili operatör bulunmaktadır. Bunlar *ve* operatörü (AND, &), *veya* operatörü (OR, +), *değil* operatörü (NOT, !) ve *özel veya* operatörü (XOR (Exclusive-OR), \oplus)'dür (Aslan ve ark., 2019). Lojik-kapı tabanlı ikili operatörler, yalnızca “0” ve “1” değerlerini kullandıkları için ikili optimizasyon problemleri için uygundur. Ancak, ikili optimizasyon problemlerinin çözümünde hangi ikili operatörün kullanılacağını belirleme aşamasına oldukça özen gösterilmesi gerekir. Bunun nedeni, konum güncelleme esnasında adil ve verimli bir şekilde yeni aday çözümleri belirleyebilmektir. Şekil 4.1'de lojik-kapı tabanlı ikili operatörlerin doğruluk tablosu ve “0” ve “1” değerlerinin dağılım yüzdeleri verilmiştir.

Karar Değişkenleri		Lojik-Kapı Tabanlı Operatörler		
A	B	AND Operatörü (A&B)	OR Operatörü (A+B)	XOR Operatörü (A \oplus B)
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0
		%75 \rightarrow 0 %25 \rightarrow 1	%25 \rightarrow 0 %75 \rightarrow 1	%50 \rightarrow 0 %50 \rightarrow 1

Şekil 4.1. Lojik-kapı tabanlı ikili operatörlerin doğruluk tablosu ve dağılım yüzdeleri (Aslan ve ark., 2019)

İkili optimizasyon sürecinde lojik-kapı tabanlı operatörlerin üç tanesi kullanılmaktadır (Sonuc, 2021). Şekil 4.1'de A ve B, yeni aday çözümün konum güncelleme denkleminde yer alan iki aday kurdun ortak boyutlarını temsil etmektedir (Aslan ve ark., 2019). Şekil 4.1'e göre, aday çözüm için konum güncelleme denkleminde AND veya OR operatörlerinin kullanılması adil bir seçim dağılımı sağlamamaktadır. AND operatörü tercih edildiğinde aday çözümün karşılık gelen yeni boyut değeri %75 olasılıkla 0, %25 olasılıkla 1 olacaktır. Benzer şekilde OR operatörü tercih edildiğinde aday çözümün karşılık gelen yeni boyut değeri %25 olasılıkla 0, %75 olasılıkla 1 olacaktır. Ancak, XOR operatörünün değer dağılımına bakacak olursak her iki değer de (0, 1) %50 olasılığı vardır. Bu nedenle, aday çözümler için konum güncelleme denkleminde XOR ikili operatörünün kullanılması adil ve verimli olacağından uygundur (Kiran ve Gündüz, 2013a; Aslan ve ark., 2019). Nitekim literatüre kazandırılan yöntemlerde de benzer nedenlerle XOR operatörü tercih edilmiştir (Kiran ve Gündüz, 2013a; Cinar ve Kiran, 2018; Aslan ve ark., 2019; Sonuc, 2021).

Baskın kurtların çözüm kalitesi üzerindeki etkilerini artırmayı hedefleyen ve dinamik katsayı yöntemi ile XOR operatörünü kullanan IDGKO'nun konum güncelleme denklemi Denklem (4.1)'de verilmiştir.

$$X_i = XOR(X_{i,a}, Alfa_a) \cup XOR(X_{i,b}, Beta_b) \cup XOR(X_{i,c}, Delta_c), \quad j = a + b + c \quad (4.1)$$

Denklemden X_i i . aday çözümü ve j problem boyutunu temsil etmektedir. *Alfa* popülasyondaki en iyi uygunluk değerine sahip alfa kurdun çözümünü, *Beta* popülasyondaki en iyi ikinci uygunluk değerine sahip beta kurdun çözümünü ve *Delta* popülasyondaki en iyi üçüncü uygunluk değerine sahip delta kurdun çözümünü temsil etmektedir. a, b ve c değerleri sırasıyla alfa, beta ve delta kurdun konum güncelleme denkleminde yeni aday çözümü etkileyeceği boyutların sayısını göstermektedir.

Orijinal GWO algoritmasının konum güncelleme denkleminde, en iyi üç kurdun (alfa, beta ve delta) konum bilgileri referans alınmakta ve aday çözümün konum güncellemesinde her biri eşit olarak 1/3 oranında etki etmektedir. Denklem (4.1)'de görüldüğü üzere, önerilen IDGKO yönteminde, baskın kurtların çözüm kalitesi üzerindeki etkilerini belirlemek adına eşit oran kullanılmamaktadır. Bunun yerine, baskın kurtların uygunluk değerine bağlı olarak elde edilen oranlar (a, b ve c) kullanılmaktadır. Böylece hem baskın kurtların çözüm kalitesi üzerindeki etkileri belirlenebilecek hem de bu yeni yöntem yerel arama ile küresel arama arasında bir denge sağlayacaktır (Kiran ve Gündüz, 2013b; Aslan ve ark., 2019).

Önerilen IDGKO'da baskın bireyler için oran belirleme işleminde DCA'dan esinlenerek yeni bir dinamik katsayı yöntemi (NDCA) benimsenmiştir. Orijinal DCA yöntemi, alfa kurt için 0.5'lik sabit bir oran; beta ve delta kurtlar için ise çözüm kalitesine bağlı bir oran üretmektedir. Ancak, NDCA yönteminde her baskın kurt için uygunluk değerine bağlı bir oran belirlenir ve yeni çözüm üzerindeki etki katsayısı hesaplanır. NDCA'da dikkat edilmesi gereken bir noktada şudur: Üç baskın kurt arasında alfa kurt için belirlenen etki katsayısı oranı beta kurt ve delta kurttan; beta kurt için belirlenen etki katsayısı oranı delta kurttan büyük olmalıdır. Oranlar belirlendikten sonra XOR işlemi uygulanarak yeni aday çözümün konum güncelleme işlemi tamamlanmaktadır. Böylece GWO algoritmasının temel yapısı bozulmamış ve ikili optimizasyon problemlerinde algoritmanın problem çözmedeki avantajları sağlanmıştır.

Optimizasyon problemlerinde, probleme bağlı olarak, optimum çözüm değeri mevcut çözümler arasından en küçük (minimum) veya en büyük (maksimum) değere

sahip çözüm olarak belirlenir. Optimum çözümü, mevcut çözümler arasındaki en küçük değere sahip çözüm olan problemlere minimizasyon problemi; optimum çözümü, mevcut çözümler arasındaki en büyük değere sahip çözüm olan problemlere ise maksimizasyon problemi denilmektedir. Tez kapsamında önerilen FS problemi, veri kümesinden alakasız/gereksiz özellikleri çıkararak optimum alt kümeyi elde etmeyi amaçladığından bir minimizasyon problemidir. Yine tez kapsamında önerilen 0-1 KP, sırt çantası kapasitesini aşmadan uygun nesnelere seçerek sırt çantasının toplam kârını maksimum seviyeye ulaştırmayı amaçladığından bir maksimizasyon problemidir. Bu sebeple, NDCA yöntemi, (Karakoyun, 2021) tarafından önerilen DCA yönteminde değişikliğe giderek hem FS hem de 0-1 KP için ayrı ayrı uyarlanmıştır.

NDCA yöntemini, maksimizasyon problemi olan 0-1 KP'nin çözümünde kullanırken popülasyondaki baskın kurtların ilk oran değerini belirlemede Denklem (4.2); minimizasyon problemi olan FS'nin çözümünde kullanırken popülasyondaki baskın kurtların ilk oran değerini belirlemede Denklem (4.3) kullanılmaktadır. Her iki problem için de baskın kurtların etki katsayı oranını belirlemek için Denklem (4.4) ve (4.5) kullanılmaktadır.

$$rate_{\alpha} = \frac{fitness_{\alpha}}{W}, \quad rate_{\beta} = \frac{fitness_{\beta}}{W}, \quad rate_{\delta} = \frac{fitness_{\delta}}{W} \quad (4.2)$$

$$rate_{\alpha} = \frac{1}{fitness_{\alpha}}, \quad rate_{\beta} = \frac{1}{fitness_{\beta}}, \quad rate_{\delta} = \frac{1}{fitness_{\delta}} \quad (4.3)$$

$$Total Rate = rate_{\alpha} + rate_{\beta} + rate_{\delta} \quad (4.4)$$

$$ECR_{\alpha} = rate_{\alpha}/Total Rate, \quad ECR_{\beta} = rate_{\beta}/Total Rate, \quad ECR_{\delta} = rate_{\delta}/Total Rate \quad (4.5)$$

Denklemlerde, $fitness_{\alpha}$ alfa bireyin uygunluk değerini, $fitness_{\beta}$ beta bireyin uygunluk değerini, $fitness_{\delta}$ delta bireyin uygunluk değerini, W sırt çantasının kapasite değerini (ağırlık değerini), ECR_{α} alfa bireyin çözüm kalitesi üzerindeki etki katsayı oranını, ECR_{β} beta bireyin çözüm kalitesi üzerindeki etki katsayı oranını, ECR_{δ} delta bireyin çözüm kalitesi üzerindeki etki katsayı oranını temsil etmektedir.

Popülasyondaki baskın kurtların etki katsayı oranlarını belirledikten sonra, bu üç kurdun aday çözümde kaç boyutu güncelleyeceğine ve bu boyutların hangi sıradaki boyutlar olacağına karar vermek gerekir. NDCA'da, baskın bireylerin toplamda kaç boyutu güncelleyeceğini her iki problem için de hesaplamak için Denklem (4.6) kullanılmaktadır. NDCA'da ilgili baskın kurt tarafından güncellenecek boyutların sırası

rastgele belirlenmektedir. Amaç, popülasyonda çeşitlilik sağlamak ve yeni çözümün boyutlarını güncellerken her üç baskın bireyden de bilgi aktarımı sağlamaktır.

$$NDU_{\alpha} = dim * ECR_{\alpha}, \quad NDU_{\beta} = dim * ECR_{\beta}, \quad NDU_{\delta} = dim * ECR_{\delta} \quad (4.6)$$

Denklemden dim problemin boyut değerini (0-1 KP için mevcut nesne sayısı; FS için veri kümesindeki mevcut özellik sayısı), NDU_{α} alfa bireyin yeni aday çözüm için güncelleyeceği boyutların sayısını, NDU_{β} beta bireyin yeni aday çözüm için güncelleyeceği boyutların sayısını, NDU_{δ} delta bireyin yeni aday çözüm için güncelleyeceği boyutların sayısını temsil etmektedir.

Denklem (4.6) ile ilgili dikkat edilmesi gereken bir husus vardır. Lider bireyler tarafından güncellenecek boyut sayısı belirlenirken formüle göre eksik kalan boyutlar olabilmektedir. Örneğin; problem boyutunun 15, sırasıyla alfa, beta ve delta bireyin uygunluk değerlerinin 0.1, 0.2 ve 0.5 olduğunu varsayalım. Bu değerler ile Denklem (4.2-4.5) gerçekleştirildiği takdirde sırasıyla alfa, beta ve delta birey tarafından güncellenecek boyut sayısının 8, 4 ve 1 olduğu görülmektedir. Mevcut güncellenecek boyut sayısı 13 (8+4+1) iken problem boyutu 15'dir. Bu sebeple böyle bir duruma sebebiyet vermemek adına gerekli kontrol sağlanarak $NDU_{\delta} = dim - (NDU_{\alpha} + NDU_{\beta})$ işlemi yapılmalıdır. Böylece sırasıyla alfa, beta ve delta birey tarafından güncellenecek boyut sayısı 8, 4 ve 3 (15-(8+4)) şeklinde olacaktır.

Önerilen IDGKO algoritmasının 0-1 KP'nin çözümü sırasında kullanılan versiyonunun sözde kodu Şekil 4.2'de verilmiştir.

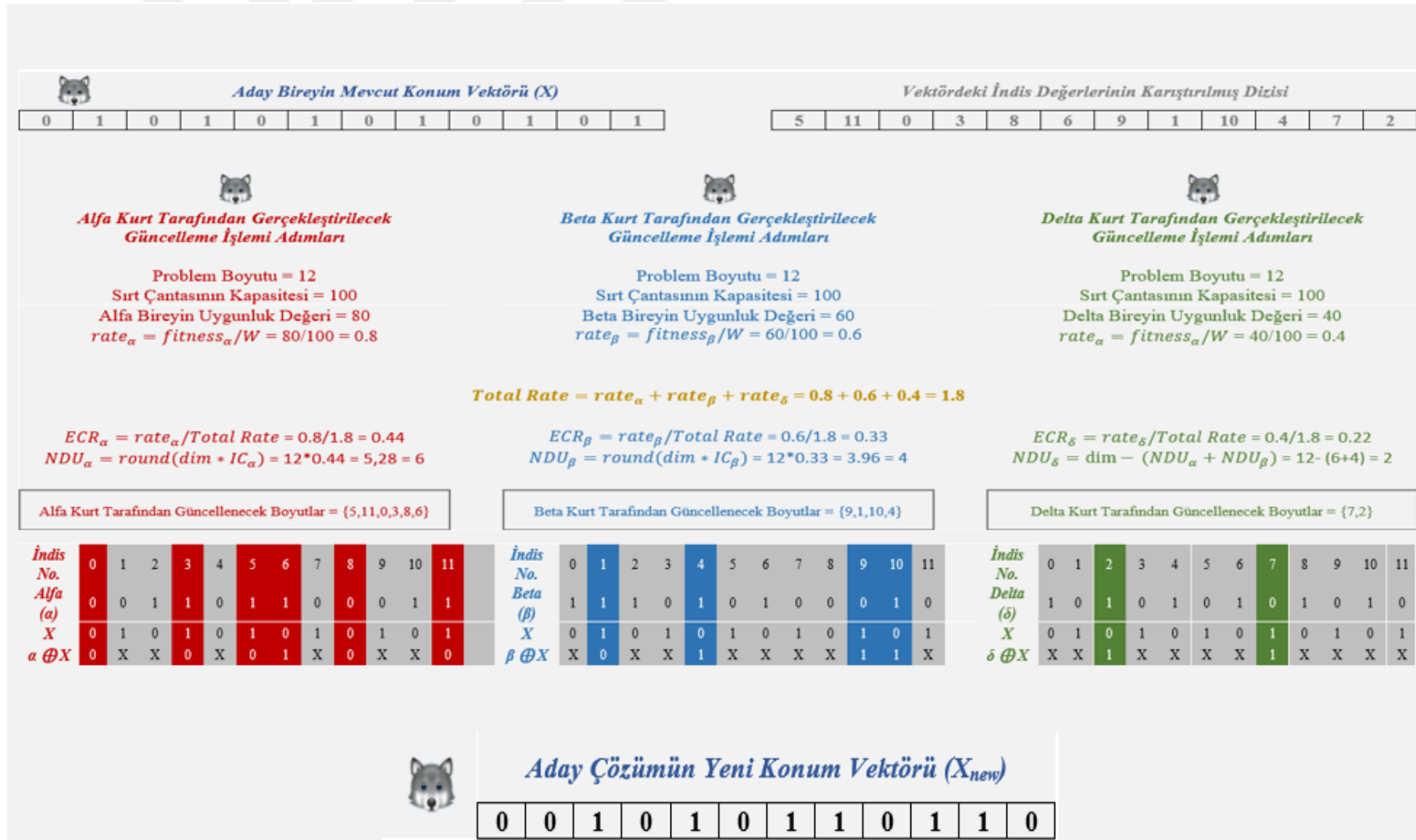
```

IDGKO için başlangıç parametrelerini belirle
İkili başlangıç popülasyonunu rastgele değerler ile oluştur  $X_i$  ( $X_i = 0$  or  $1$ ,  $i=1,2,\dots,n$ )
Popülasyondaki tüm bireylerin uygunluk değerlerini Denklem (2.1) ile hesapla
Uygunluk değerlerine göre lider bireyleri belirle
   $X_{\alpha}$  = Alfa Kurt = en iyi uygunluk değerine sahip gri kurt
   $X_{\beta}$  = Beta Kurt = ikinci en iyi uygunluk değerine sahip gri kurt
   $X_{\delta}$  = Delta Kurt = üçüncü en iyi uygunluk değerine sahip gri kurt
while ( $t <$  Sonlandırma Kriteri)
  for popülasyondaki tüm bireyler için
    Denklem (4.1-4.6) ile ilgili aday çözümün konumunu güncelle
  end for
  Popülasyondaki tüm bireylerin uygunluk değerlerini Denklem (2.1) ile hesapla
  Uygunluk değerlerine göre lider bireyleri belirle
   $t++$ 
end while
 $X_a$  bilgilerini döndür ve algoritmayı sonlandır

```

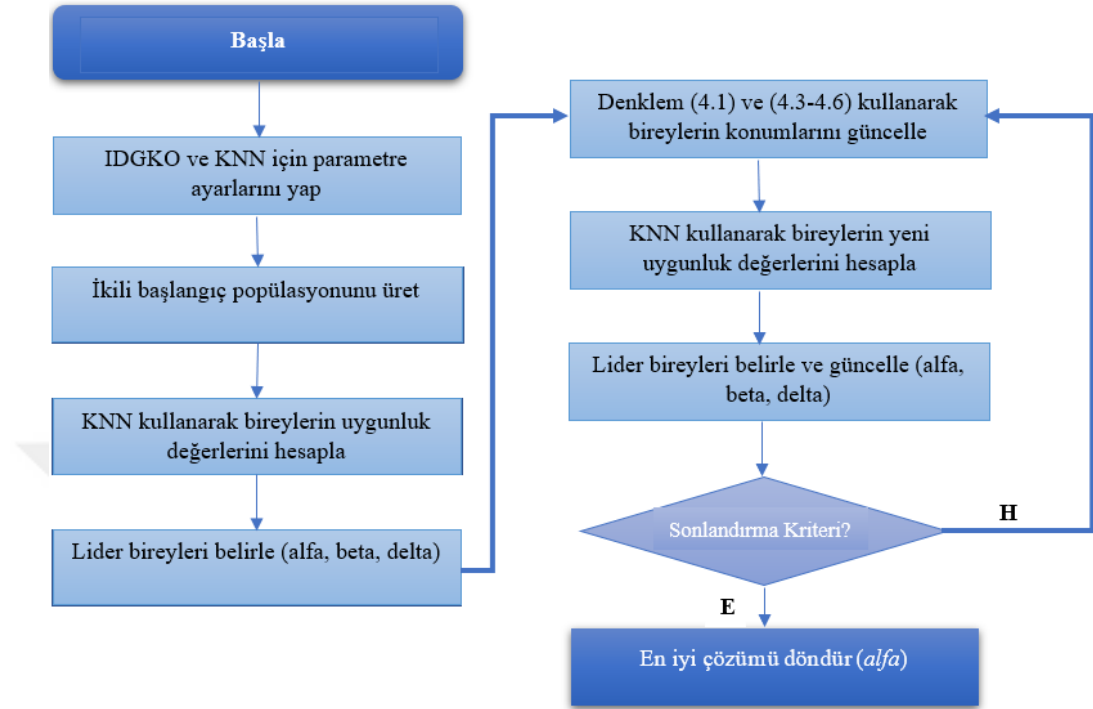
Şekil 4.2. 0-1 KP'nin çözümü sırasında kullanılan IDGKO'nun sözde kodu

NDCA yönteminin 0-1 KP'nin çözümü için kullanımını daha iyi anlamak için Şekil 4.3'te örnek bir çalışma şeması verilmiştir.



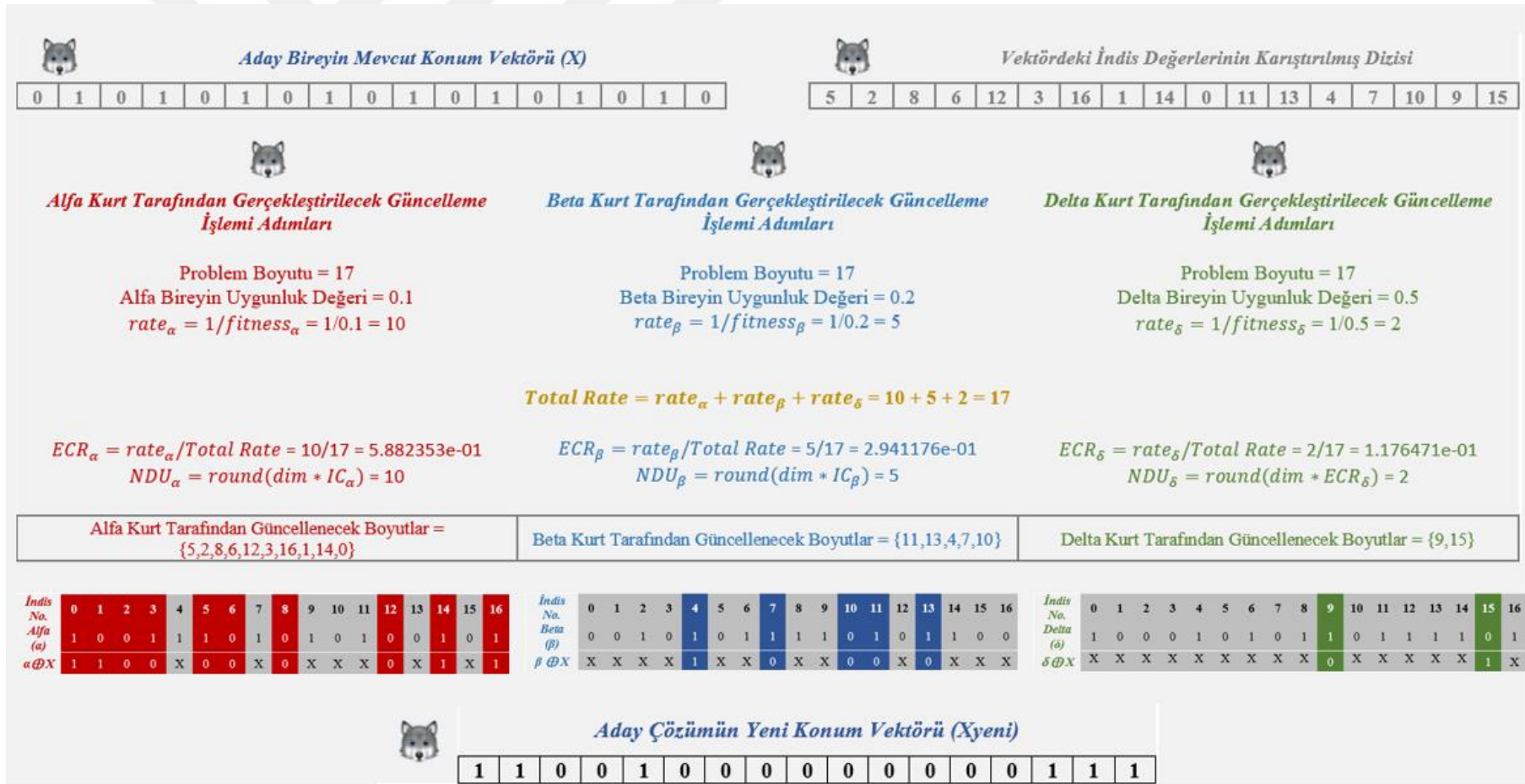
Şekil 4.3. 0-1 KP için NDCA'nın örnek çalışma şeması

Önerilen IDGKO algoritmasının FS'nin çözümü sırasında kullanılan versiyonunun akış şeması Şekil 4.4'te verilmiştir.



Şekil 4.4. FS'nin çözümü sırasında kullanılan IDGKO'nun akış şeması

NDCA yönteminin FS'nin çözümü için kullanımını daha iyi anlamak için Şekil 4.5'te örnek bir çalışma şeması verilmiştir.



Şekil 4.5. FS için NDCA'nın örnek çalışma şeması

5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu bölümde tez kapsamında önerilen yeni ikili IDGKO'nun, FS ve 0-1 KP'nin çözümünde performansını belirlemek adına yapılan deneylerin sonuçları ve deneyler sonucunda elde edilen sonuçların tartışması yer almaktadır.

Deneyel çalışmalar sırasında gerekli olan tüm kodlamalar i7-7500U CPU, 2.90 GHz ve 12,0 GB RAM özellikleri olan Windows 64 bit işletim sistemine sahip bir bilgisayarda gerçekleştirilmiştir.

5.1. FS için IDGKO'nun Araştırma Sonuçları

IDGKO'nun FS'nin çözümünde performansını belirlemek adına literatüre FS'nin çözümü için kazandırılan GWO tabanlı ve diğer metasezgisel algoritmalar tabanlı çalışmalar ile aynı deney şartları altında kıyaslama işlemleri yapılmıştır. Deneyler sırasında kullanılan veri setleri, çalışma ortamı, performans belirleme ölçütleri, parametre ayarları gibi bilgiler doğrudan kıyas yapılan çalışmalardan referans alınmıştır.

5.1.1. Deneyel Çalışma 1

Deneyel Çalışma 1 kapsamında, kaynak araştırması kısmında detaylı bilgileri verilen, Hu ve arkadaşları (2020) tarafından ikili optimizasyon problemlerinin çözümü için önerilen BGWO ve Emary ve arkadaşları (2016) tarafından özellik seçimi problemlerinin çözümü için önerilen bGWO varyantlarının tez kapsamında önerilen IDGKO ile kıyaslama çalışması yer almaktadır.

Deneyler sırasında kullanılan veri setleri, parametre ayarları, kıyas yapılan algoritma bilgileri ve çizelgelerde yer alan IDGKO verileri dışındaki tüm veriler BGWO (Hu ve ark., 2020) ve bGWO (Emary ve ark., 2016a) çalışmalarından alınmıştır. Kıyas yapılan çalışmalar ile aynı ortam sağlanarak IDGKO tarafından elde edilen sonuçlar Çizelge 5.1'e eklenmiştir.

Kıyaslamalarda BGWO, bGWO varyantları, IDGKO ve diğer algoritmalar için popülasyondaki birey sayısı 8, algoritma için iterasyon sayısı 70 ve bağımsız çalışma sayısı 20 olarak belirlenmiştir.

Çizelge 5.1’de Hu ve arkadaşları (2020) tarafından geliştirilen BGWO’nun ikili versiyonları, Emary ve arkadaşları (2016) tarafından geliştirilen bGWO’nun ikili varyantları ve IDGKO’nun, ortalama uygunluk değeri (Ort) ve algoritma performansı bazında sıralama (Sıra) değeri bilgileri sunulmuştur.



Çizelge 5.1. Algoritmaların ortalama uygunluk ve sıralama değerleri (Emary ve ark., 2016a; Hu ve ark., 2020)

Veri Seti	IDGKO		BGWO		ABGWO		ABGWO_V1		ABGWO_V2a		ABGWO_V3a		ABGWO_V4a		bGWO1		bGWO2		GA		PSO	
	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra	Ort.	Sıra
F3	0.0136	1	0.0286	5	0.0279	3	0.0293	7	0.0291	6	0.0296	8	0.0279	3	0.0300	9	0.0270	2	0.0270	2	0.0280	4
F5	0.0229	1	0.1610	8	0.1562	6	0.1636	9	0.1648	10	0.1651	11	0.1606	7	0.0570	5	0.0370	2	0.0440	3	0.0480	4
F6	0.0670	1	0.1936	10	0.1845	8	0.1772	5	0.1797	6	0.1816	7	0.1855	9	0.1740	4	0.1040	2	0.1540	3	0.1540	3
F9	0.0821	1	0.1356	11	0.1273	6	0.1277	7	0.1281	8	0.1306	9	0.1330	10	0.1060	4	0.0840	2	0.1110	5	0.1010	3
F10	0.0965	1	0.5364	11	0.5277	8	0.5329	10	0.5275	7	0.5297	9	0.5248	6	0.1960	5	0.1510	2	0.1680	4	0.1590	3
F13	0.1018	1	0.2625	11	0.2505	6	0.2587	9	0.2577	8	0.2572	7	0.2602	10	0.1780	5	0.1530	2	0.1600	3	0.1660	4
F14	0.1064	1	0.1776	9	0.1706	6	0.1771	8	0.1806	10	0.1725	7	0.1978	11	0.1360	3	0.1270	2	0.1420	4	0.1440	5
F15	0.1615	1	0.2201	2	0.2343	8	0.2277	5	0.2400	10	0.2390	9	0.2287	6	0.2330	7	0.2250	4	0.2330	7	0.2230	3
F17	0.2137	9	0.1768	2	0.1718	1	0.1796	5	0.1801	6	0.1786	4	0.1778	3	0.2140	10	0.2000	7	0.2060	8	0.2210	11
F19	0.0225	1	0.0974	7	0.0726	4	0.0712	2	0.0776	6	0.0745	5	0.0715	3	0.1270	10	0.1120	8	0.1180	9	0.1330	11
Ortalama	0.0888	1.8	0.19896	7.6	0.19234	5.6	0.1945	6.7	0.19652	7.7	0.19584	7.6	0.19678	6.8	0.1451	6.2	0.122	3.3	0.1363	4.8	0.1377	5.1

Çizelge 5.1’de IDGKO, 10 farklı ikili yöntem ile ortalama uygunluk değeri ve bu değerlerin sıralaması bakımından kıyaslanmıştır. Çizelgede görüldüğü üzere genel ortalama uygunluk değeri ve sıralama değerleri ortalaması bakımından sırasıyla 0.0888 ve 1.8 ile önerilen IDGKO algoritması ilk sırada yer almaktadır. Hu ve arkadaşları (2020) tarafından yapılan çalışmada geliştirilen ikili GWO yöntemleri yeni transfer fonksiyonları ile ikilileştirilirken; Emary ve arkadaşları (2016) tarafından yapılan çalışmada geliştirilen ikili GWO yöntemleri genetik operatörler ve transfer fonksiyonu kullanılarak ikilileştirilmektedir. Buradan yola çıkarak, ikilileştirme yöntemlerinden olan lojik-kapı tabanlı ikili operatör kullanımının yine ikilileştirme yöntemlerinden olan genetik operatörler ve transfer fonksiyonlarından GWO algoritması için daha başarılı olduğu söylenebilir. Önceki bölümlerde bahsi geçen transfer fonksiyonu kullanımının keşif ve sömürü açısından getirdiği dezavantajlar sebebiyle de yöntemlerin geride kaldığı söylenebilir. IDGKO’da, baskın kurtların çözüm kalitesi üzerindeki etkilerini belirlemek adına geliştirilen yöntem sayesinde yerel arama ile küresel arama arasında bir denge sağlanmış olması da algoritmanın başarılı olmasının temel sebepleri arasında olduğunu göstermektedir.

5.1.2. Deneysel Çalışma 2

Deneysel Çalışma 2 kapsamında GWO’nun farklı yöntemler kullanılarak gerçekleştirilen hibrit ikili versiyonları ele alınmaktadır. Bu bağlamda, Al-Tashi ve arkadaşları (2019) tarafından GWO ve PSO algoritmaları kullanılarak geliştirilen BGWOPSO (Al-Tashi ve ark., 2019) hibrit yöntemi, Al-Wajih ve arkadaşları (2021) tarafından GWO ve HHO (Harris Hawks Optimization) algoritmaları kullanılarak geliştirilen HBGWOHHO (Al-Wajih ve ark., 2021) hibrit yöntemi ve tez kapsamında önerilen IDGKO arasında kıyaslama çalışması yer almaktadır.

Al-Tashi ve arkadaşları (2019), özellik seçim problemi için BGWOPSO adlı yeni ikili bir hibrit algoritma önermiştir. İkili BGWOPSO algoritması, orijinal GWOPSO hibrit algoritmasının konum güncelleme denkleminde yeni bir değişiklik yapılarak geliştirilmiştir. Önerilen BGWOPSO, KNN sınıflandırıcı kullanılarak özellik seçimi problemi üzerinde test edilmiştir. BGWOPSO'nun performansı, öncesinde literatüre kazandırılan bGWO2 (Emary ve ark., 2016a), BPSO, BGA ve WOASAT-2 (Mafarja ve Mirjalili, 2017b) yöntemlerinin performansı ile karşılaştırılmıştır.

Al-Wajih ve arkadaşları (2021), GWO'daki keşif ve sömürü arasında dengeyi sağlamak için HHO ve GWO'ya dayalı yeni hibrit ikili bir algoritma (HBGWOHHO) (Al-Wajih ve ark., 2021) önermiştir. Önerilen yöntemde arama uzayı sigmoid transfer fonksiyonu ile ikili formata dönüştürülmektedir. Önerilen yöntem, özellik seçimi problemi üzerinde KNN sınıflandırıcısı kullanılarak gerçekleştirilmiştir. HBGWOHHO'nun performansı, öncesinde literatüre kazandırılan BGWO (Too ve ark., 2018), BHHO (Too ve ark., 2019a), BPSO, BGA (Mafarja ve Mirjalili, 2017b; Al-Tashi ve ark., 2019) ve BGWOPSO yöntemlerinin performansı ile karşılaştırılmıştır.

Deneyler sırasında kullanılan veri setleri, parametre ayarları, kıyas yapılan algoritma bilgileri ve çizelgelerde yer alan IDGKO verileri dışındaki tüm veriler BGWOPSO ve HBGWOHHO çalışmalarından alınmıştır. Kıyas yapılan çalışmalar ile aynı ortam sağlanarak IDGKO tarafından elde edilen sonuçlar Çizelge 5.2'e eklenmiştir.

Kıyaslamalarda BGWOPSO, HBGWOHHO, IDGKO ve diğer algoritmalar için popülasyondaki birey sayısı 10, algoritma için iterasyon sayısı 100 ve bağımsız çalışma sayısı 30 olarak belirlenmiştir.

Çizelge 5.2'de Al-Tashi ve arkadaşları (2019) tarafından geliştirilen BGWOPSO'nun, Al-Wajih ve arkadaşları (2021) tarafından geliştirilen HBGWOHHO ve IDGKO'nun, ortalama sınıflandırma doğruluğu (classification accuracy, Osd) ortalama uygunluk değeri (Ort), en iyi uygunluk değeri (Eud) ve algoritma performansı bazında sıralama (Sıra) değeri bilgileri sunulmuştur.

Çizelge 5.2. Algoritmaların sınıflandırma doğruluğu, ortalama uygunluk, en iyi uygunluk ve sıralama değerleri (Al-Tashi ve ark., 2019; Al-Wajih ve ark., 2021)

Veri Seti	IDGKO				BGWOPSO				HBGWOHHO				bGWO2				BPSO				BGA				WOASAT-2				BGWO				BHHO			
	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra	Osd.	Ort.	Eud.	Sıra
F2	0.98	0.02	0.00	2	0.97	0.04	0.03	3	0.99	0.01	0.00	1	0.93	0.03	0.02	5	0.94	0.03	0.02	4	0.94	0.04	0.02	4	0.94	0.03	0.02	4	0.97	0.03	0.03	3	0.99	0.00	0.00	1
F3	0.99	0.01	0.00	1	0.98	0.03	0.02	2	0.99	0.00	0.00	1	0.97	0.03	0.02	3	0.95	0.03	0.03	5	0.96	0.03	0.02	4	0.96	0.04	0.03	4	0.97	0.03	0.02	3	0.97	0.02	0.01	3
F4	0.97	0.03	0.01	2	0.98	0.02	0.02	1	0.97	0.03	0.01	2	0.95	0.04	0.02	3	0.94	0.05	0.03	4	0.92	0.07	0.03	5	0.98	0.02	0.02	1	0.97	0.03	0.02	2	0.98	0.03	0.02	1
F5	0.99	0.01	0.00	1	0.98	0.03	0.02	2	0.99	0.01	0.00	1	0.93	0.04	0.02	5	0.94	0.04	0.03	4	0.94	0.04	0.03	4	0.98	0.03	0.02	2	0.96	0.04	0.05	3	0.98	0.02	0.02	2
F6	0.97	0.03	0.00	1	0.96	0.05	0.02	2	0.92	0.08	0.05	3	0.72	0.10	0.07	7	0.74	0.13	0.07	5	0.73	0.13	0.07	6	0.97	0.03	0.01	2	0.86	0.15	0.12	4	0.97	0.04	0.00	1
F7	0.95	0.05	0.00	3	1.00	0.00	0.00	1	0.74	0.26	0.17	5	0.77	0.22	0.07	4	0.68	0.28	0.21	6	0.67	0.28	0.27	7	1.00	0.01	0.01	1	0.96	0.04	0.00	2	0.95	0.05	0.00	3
F8	0.79	0.21	0.18	1	0.76	0.24	0.23	4	0.77	0.23	0.22	3	0.75	0.25	0.23	5	0.75	0.25	0.22	5	0.76	0.25	0.22	4	0.75	0.25	0.23	5	0.76	0.24	0.24	4	0.78	0.22	0.21	2
F9	0.92	0.08	0.04	4	0.95	0.05	0.04	2	0.93	0.07	0.05	3	0.83	0.08	0.07	7	0.84	0.14	0.12	6	0.83	0.13	0.09	7	0.96	0.04	0.03	1	0.90	0.10	0.08	5	0.93	0.07	0.07	3
F10	0.93	0.07	0.00	1	0.92	0.08	0.06	2	0.90	0.10	0.06	3	0.70	0.15	0.10	8	0.69	0.19	0.14	9	0.71	0.17	0.12	7	0.89	0.11	0.09	4	0.85	0.15	0.12	6	0.86	0.15	0.11	5
F11	0.99	0.01	0.00	2	1.00	0.00	0.00	1	0.93	0.07	0.00	5	0.96	0.04	0.00	4	0.86	0.11	0.06	6	0.93	0.08	0.02	5	1.00	0.01	0.01	1	0.97	0.03	0.00	3	0.99	0.02	0.00	2
F12	0.97	0.02	0.00	2	0.96	0.05	0.03	3	1.00	0.00	0.00	1	0.58	0.23	0.17	7	0.72	0.22	0.13	5	0.70	0.22	0.13	6	0.94	0.06	0.03	4	0.58	0.14	0.12	7	1.00	0.00	0.00	1
F13	0.92	0.07	0.03	1	0.88	0.12	0.11	3	0.91	0.09	0.05	2	0.82	0.15	0.12	5	0.77	0.13	0.10	7	0.78	0.14	0.12	6	0.88	0.13	0.11	3	0.85	0.15	0.14	4	0.85	0.15	0.13	4
F14	0.90	0.10	0.07	2	0.85	0.15	0.12	4	0.90	0.10	0.03	2	0.77	0.13	0.10	7	0.78	0.15	0.13	6	0.82	0.14	0.12	5	0.85	0.16	0.13	4	0.86	0.14	0.13	3	0.93	0.07	0.05	1
F15	0.84	0.15	0.11	2	0.81	0.19	0.17	3	0.79	0.21	0.16	4	0.72	0.23	0.21	7	0.73	0.24	0.21	6	0.71	0.24	0.21	8	0.79	0.21	0.20	4	0.77	0.23	0.23	5	0.87	0.13	0.11	1
F16	0.98	0.02	0.00	2	0.97	0.03	0.03	3	0.98	0.02	0.01	2	0.92	0.03	0.00	5	0.89	0.05	0.03	6	0.89	0.05	0.03	6	0.97	0.04	0.02	3	0.96	0.05	0.04	4	0.99	0.00	0.00	1
F17	0.79	0.21	0.19	4	0.80	0.21	0.20	3	0.86	0.16	0.16	1	0.78	0.20	0.19	5	0.76	0.22	0.21	7	0.77	0.20	0.19	6	0.76	0.25	0.23	7	0.85	0.17	0.16	2	0.85	0.16	0.15	2
F18	0.99	0.01	0.00	2	1.00	0.00	0.00	1	1.00	0.00	0.00	1	0.92	0.01	0.00	6	0.95	0.02	0.00	4	0.93	0.01	0.00	5	0.99	0.01	0.00	2	0.98	0.03	0.01	3	0.99	0.02	0.01	2
F19	0.98	0.02	0.00	2	1.00	0.00	0.00	1	0.97	0.03	0.00	3	0.87	0.11	0.00	7	0.83	0.10	0.03	8	0.88	0.08	0.00	6	0.97	0.04	0.00	3	0.95	0.05	0.03	5	0.96	0.04	0.02	4
Ortalama	0.93	0.06	0.03	1.94	0.93	0.07	0.06	2.27	0.91	0.08	0.05	2.38	0.82	0.11	0.07	5.55	0.82	0.13	0.09	5.72	0.82	0.12	0.09	5.61	0.92	0.08	0.06	3.05	0.88	0.10	0.08	3.77	0.93	0.06	0.05	2.16

Çizelge 5.2’de IDGKO, 8 farklı ikili algoritma ile sınıflandırma doğruluğu, ortalama uygunluk, en iyi uygunluk ve başarı sıralaması değerleri bakımından kıyaslanmıştır. Çizelgede görüldüğü üzere genel sınıflandırma başarısı ortalaması bakımından 0.93 değeri ile önerilen IDGKO algoritması, BGWOPSO hibrit yöntemi ve BHHO ilk sırada yer almaktadır. Yine çizelgeye göre ortalama uygunluk değerlerinin, en iyi uygunluk değerlerinin ve sıralama değerlerinin genel ortalaması bakımında sırasıyla 0.06, 0.03 ve 1.94 değerleri ile önerilen IDGKO yöntemi, diğer yöntemleri geride bırakmaktadır. En iyi sınıflandırma doğruluğu değerlerine bakıldığında önerilen IDGKO yöntemi, 18 veri setinin 11’inde 0.00 değerini elde ederek en iyi özellik alt kümesini elde etmiş ve bu oranla diğer yöntemleri ciddi bir farkla geride bırakmıştır. Genel olarak çizelgeye göre IDGKO yönteminin, hibrit yöntemlere ve diğer ikili yöntemlere oranla veri setlerinin çoğunluğunda %90 ve üzeri sınıflandırma doğruluğu elde ettiği ve yine birçok veri setinde yüksek oranla en iyi özellik alt kümesini seçtiği doğrulanmaktadır.

5.2. 0-1 KP için IDGKO’nun Araştırma Sonuçları

IDGKO’nun 0-1 KP’nin çözümünde performansını belirlemek adına literatüre 0-1 KP’nin çözümü için kazandırılan GWO tabanlı ve diğer metasezgisel algoritmalar tabanlı çalışmalar ile aynı deney şartları altında kıyaslama işlemleri yapılmıştır.

Deneyler sırasında kullanılan veri setleri, parametre ayarları, kıyaslanan yöntemler gibi bilgiler doğrudan kıyas yapılan çalışmalardan alınmış olup ilerleyen bölümlerde sunulacaktır.

Maksimizasyon problemlerinde uygunluk fonksiyonunun maksimum değeri elde etmesi amaçlanır. GWO’daki bireyler tarafından 0-1 KP’nin çözümünde hangi nesnelere çantada yer alıp/almayacağını temsil eden konum vektörleri, önceki bölümlerde sunulan ve 0-1 KP’nin uygunluk fonksiyonu olarak kullanılan Denklem (3.6) ile uygunluk değeri hesaplama işlemine tabii tutulur.

5.2.1. Deneysel Çalışma 1

Haklı (2020, yakın tarihte literatüre kazandırılan Fil Sürüsü Optimizasyon Algoritması’nın (Elephant Herding Optimization Algorithm, EHO) etkili küresel arama yeteneği, az sayıda kontrol parametresine sahip olması ve kolay uygulanabilirliği gibi avantajları sebebiyle ikili optimizasyon problemlerinin çözümü için yeni ikili varyantını

(BinEHO) önermiştir (Hakli, 2020). Önerilen yöntemin performansını ve doğruluğunu belirlemek için üç önemli ikili optimizasyon problemi olan 0-1 KP, kapasitesiz tesis yerleştirme problemi (uncapacitated facility location problem), rüzgâr tribün yerleşimi problemi (wind turbine placement problem) üzerinde test edilmiştir. Yapılan deneyler ve diğer ikili yöntemler ile karşılaştırma sonucunda, BinEHO algoritmasının ikili optimizasyon için sağlam ve verimli bir yaklaşım olduğu kanıtlanmıştır.

Çalışmada, deneyler esnasında kullanılan veri setleri Bansal ve Deep (2012) tarafından yapılan çalışmadan temin edilmiştir (Bansal ve Deep, 2012). Veri setleri hakkında detaylı bilgi Çizelge 5.3'te sunulmuştur.

Çizelge 5.3. Deneylerde kullanılan veri setlerine ait bilgiler

Veri Seti ID	Veri Seti	Optimum Değer	Boyut/Nesne Sayısı
D1	ks_8a	3924400	8
D2	ks_8b	3813669	
D3	ks_8c	3347452	
D4	ks_8d	4187707	
D5	ks_8e	4955555	
D6	ks_12a	5688887	12
D7	ks_12b	6498597	
D8	ks_12c	5170626	
D9	ks_12d	6992404	
D10	ks_12e	5337472	
D11	ks_16a	7850983	16
D12	ks_16b	9352998	
D13	ks_16c	9151147	
D14	ks_16d	9348889	
D15	ks_16e	7769117	
D16	ks_20a	10727049	20
D17	ks_20b	9818261	
D18	ks_20c	10714023	
D19	ks_20d	8929156	
D20	ks_20e	9357969	
D21	ks_24a	13549094	24
D22	ks_24b	12233713	
D23	ks_24c	12448780	
D24	ks_24d	11815315	
D25	ks_24e	13940099	

Çizelge 5.3'te görüldüğü üzere, 0-1 KP'nin optimum değeri büyük haneli bir sayıdır. Bu sebeple, verilerin ve kıyaslama sonuçlarının kolay anlaşılabilmesi için GAP adı verilen fonksiyonlar kullanılmaktadır. Bahsi geçen GAP fonksiyonunun matematiksel ifadesi Denklem (5.1)'de verilmiştir.

$$GAP = \frac{Ortalama-Optimum}{Optimum} \times 100 \quad (5.1)$$

Denkleimde *Ortalama*, belirlenen çalıştırma sayısı boyunca elde edilen en iyi değerlerin ortalama değerini, *Optimum* ise ilgili problem veri seti için en iyi sonuç değerini temsil etmektedir.

Deneyler sırasında belirlenen parametre ayarları Çizelge 5.4'te sunulmuştur.

Çizelge 5.4. Deneylerde kullanılan veri setlerine ait bilgiler

Parametre Adı	Parametre Değeri
Popülasyondaki Birey Sayısı	40
İterasyon Sayısı	50
Maksimum Uygunluk Değerlendirme Sayısı	D1-D10 için 1000 D11-D15 için 5000

Haklı (2020) tarafından geliştirilen BinEHO, orijinal makalesinde 6 farklı ikili yöntem (BPSO (Kennedy ve Eberhart, 1997), MBPSO (Bansal ve Deep, 2012), NGHS (Zou ve ark., 2011), DGHS (Xiang ve ark., 2014), S-BAFSA (Azad ve ark., 2014), CGMA (Zhou ve ark., 2016)) ile kıyas edilmiştir. Çizelge 5.5'te BinEHO, diğer ikili yöntemler ve IDGKO'nun ortalama uygunluk değeri (M), GAP değeri (G) ve algoritma performansı bazında sıralama (R) değeri bilgileri sunulmuştur.

Çizelge 5.5. Algoritmaların sınıflandırma doğruluğu, ortalama uygunluk, en iyi uygunluk ve sıralama değerleri (Hakli, 2020)

Dataset	IDGKO			BinEHO			BPSO			MBPSO			NGHS			DGHS			S-bAFSA			CGMA				
	M	G	R	M	G	R	M	G	R	M	G	R	M	G	R	M	G	R	M	G	R	M	G	R		
D1	3,924,400.00	0	1	3,924,400.00	0	1	3,921,857.19	6.48E-02	2	3,924,400.00	0	1	3,924,400.00	0	1	3,924,400.00	0	1	3,924,400.00	0	1	3,924,400.00	0	1	1	
D2	3,813,669.00	0	1	3,813,669.00	0	1	3,807,911.86	1.51E-01	2	3,813,669.00	0	1	3,813,669.00	0	1	3,813,669.00	0	1	3,813,669.00	0	1	3,813,669.00	0	1	1	1
D3	3,347,452.00	0	1	3,347,452.00	0	1	3,328,608.71	5.60E-01	2	3,347,452.00	0	1	3,347,452.00	0	1	3,347,452.00	0	1	3,347,452.00	0	1	3,347,452.00	0	1	1	1
D4	4,187,707.00	0	1	4,187,707.00	0	1	4,186,088.27	3.87E-02	2	4,187,707.00	0	1	4,187,707.00	0	1	4,187,707.00	0	1	4,187,707.00	0	1	4,187,707.00	0	1	1	1
D5	4,955,555.00	0	1	4,955,555.00	0	1	4,932,737.28	4.60E-01	3	4,954,571.72	1.98E-02	2	4,955,555.00	0	1	4,955,555.00	0	1	4,955,555.00	0	1	4,955,555.00	0	1	1	1
D6	5,688,887.00	0	1	5,688,757.34	2.28E-03	2	5,683,694.29	9.13E-02	7	5,688,552.41	5.88E-03	3	5,687,724.44	2.04E-02	5	5,687,460.74	2.51E-02	6	5,688,498.02	6.84E-03	4	5,688,887.00	0	1	1	
D7	6,498,597.00	0	1	6,498,597.00	0	1	6,478,582.96	3.08E-01	4	6,493,130.57	8.41E-02	2	6,486,450.92	1.87E-01	3	6,498,597.00	0	1	6,498,597.00	0	1	6,498,597.00	0	1	1	
D8	5,170,626.00	0	1	5,170,626.00	0	1	5,166,957.08	7.10E-02	3	5,170,493.30	2.57E-03	2	5,165,079.86	1.07E-01	4	5,165,079.86	1.07E-01	4	5,170,626.00	0	1	5,170,626.00	0	1	1	
D9	6,992,404.00	0	1	6,992,404.00	0	1	6,989,842.73	3.66E-02	4	6,992,144.26	3.71E-03	2	6,991,791.10	6.19E-03	3	6,992,404.00	0	1	6,992,404.00	0	1	6,992,404.00	0	1	1	
D10	5,337,472.00	0	1	5,337,472.00	0	1	5,316,879.59	3.86E-01	3	5,337,472.00	0	1	5,273,974.06	1.19E+00	4	5,335,555.92	3.59E-02	2	5,337,472.00	0	1	5,337,472.00	0	1	1	
D11	7,850,983.00	0	1	7,850,224.60	9.66E-03	2	7,834,900.26	2.05E-01	4	7,843,073.29	1.01E-01	3	7,795,177.56	7.11E-01	5	7,850,983.00	0	1	7,850,983.00	0	1	7,850,983.00	0	1	1	
D12	9,352,998.00	0	1	9,352,998.00	0	1	9,334,408.62	1.99E-01	3	9,350,353.39	2.83E-02	2	9,253,153.36	1.07E+00	4	9,352,998.00	0	1	9,352,998.00	0	1	9,352,998.00	0	1	1	
D13	9,151,147.00	0	1	9,151,147.00	0	1	9,118,837.47	3.53E-01	3	9,144,118.38	7.68E-02	2	9,055,853.46	1.04E+00	4	9,151,147.00	0	1	9,151,147.00	0	1	9,151,147.00	0	1	1	
D14	9,345,718.00	3.39E-02	2	9,345,961.48	3.13E-02	1	9,321,705.87	2.91E-01	5	9,337,915.64	1.17E-01	4	9,310,913.74	4.06E-01	8	9,316,797.66	3.43E-01	7	9,321,195.66	2.96E-01	6	9,345,229.60	3.91E-02	3		
D15	7,767,930.00	1.53E-02	1	7,766,509.36	3.36E-02	3	7,758,572.21	1.36E-01	6	7,764,131.81	6.42E-02	4	7,738,799.78	3.90E-01	8	7,748,285.68	2.68E-01	7	7,761,387.72	9.95E-02	5	7,767,929.72	1.53E-02	2		
D16	10,727,049.00	0	1	10,727,049.00	0	1	10,707,360.91	1.84E-01	3	10,720,314.03	6.28E-02	2	10,592,304.82	1.26E+00	4	10,727,049.00	0	1	10,727,049.00	0	1	10,727,049.00	0	1	1	
D17	9,818,261.00	0	1	9,818,261.00	0	1	9,791,306.65	2.75E-01	3	9,805,480.48	1.30E-01	2	9,728,979.66	9.09E-01	4	9,818,261.00	0	1	9,818,261.00	0	1	9,818,261.00	0	1	1	
D18	10,713,144.50	8.19E-03	2	10,713,587.70	4.06E-03	1	10,703,423.34	9.89E-02	7	10,710,947.05	2.87E-02	4	10,585,114.14	1.20E+00	8	10,709,663.64	4.07E-02	6	10,710,831.94	2.98E-02	5	10,712,553.74	1.37E-02	3		
D19	8,924,753.00	4.93E-02	2	8,929,156.00	0	1	8,910,152.57	2.13E-01	7	8,923,712.21	6.10E-02	3	8,859,297.78	7.82E-01	8	8,916,496.80	1.42E-01	6	8,917,872.80	1.26E-01	5	8,918,423.20	1.20E-01	4		
D20	9,357,767.00	2.20E-03	2	9,357,751.44	2.32E-03	3	9,349,546.98	9.00E-02	7	9,355,930.35	2.18E-02	6	9,324,698.84	3.56E-01	8	9,357,518.34	4.82E-03	4	9,357,331.86	6.81E-03	5	9,357,922.38	4.98E-04	1		
D21	13,547,113.68	1.46E-02	1	13,546,986.14	1.56E-02	2	13,510,432.96	2.85E-01	7	13,532,060.07	1.26E-01	4	13,508,995.14	2.96E-01	8	13,527,887.56	1.57E-01	5	13,524,874.30	1.79E-01	6	13,538,097.18	8.12E-02	3		
D22	12,233,713.00	0	1	12,233,713.00	0	1	12,205,346.16	2.32E-01	3	12,223,442.61	8.40E-02	2	12,160,953.58	5.95E-01	4	12,233,713.00	0	1	12,233,713.00	0	1	12,233,713.00	0	1	1	
D23	12,448,712.00	5.46E-04	2	12,448,780.00	0	1	12,427,880.56	1.68E-01	7	12,443,349.03	4.36E-02	6	12,424,467.44	1.95E-01	8	12,448,158.86	4.99E-03	5	12,448,550.28	1.85E-03	4	12,448,618.30	1.30E-03	3		
D24	11,810,998.50	3.65E-02	1	11,810,682.68	3.92E-02	3	11,792,064.76	1.97E-01	7	11,803,712.38	9.82E-02	6	11,736,314.98	6.69E-01	8	11,810,261.56	4.28E-02	4	11,810,893.24	3.74E-02	2	11,808,614.90	5.67E-02	5		
D25	13,940,099.00	0	1	13,940,099.00	0	1	13,922,797.55	1.24E-01	3	13,932,526.16	5.43E-02	2	13,827,901.40	8.08E-01	4	13,940,099.00	0	1	13,940,099.00	0	1	13,940,099.00	0	1	1	

Çizelge 5.5'te sunulan verilerin daha kolay yorumlanabilmesi adına, her bir algoritma elde ettiği GAP değerleri üzerinden Friedman Testi'ne tabii tutulmuş ve sonuçlar Çizelge 5.6'da verilmiştir. Yine, algoritmaların elde ettiği başarı sıralamasına göre ortalaması hesaplanarak Çizelge 5.6'da sunulmuştur.

Çizelge 5.6. Algoritmaların Friedman ve ortalama başarı sıralaması bakımından kıyaslama sonuçları (Haklı, 2020)

Algoritma	Friedman Sıralama Değeri	Ortalama Başarı Sıralaması Değeri
IDGKO	2.66	1.20
BinEHO	2.88	1.36
BPSO	7.12	4.28
MBPSO	5.08	2.72
NGHS	7.00	4.68
DGHS	4.42	2.80
S-bAFSA	3.78	2.32
CGMA	3.06	1.64

Çizelge 5.5'te verilen ortalama uygunluk değerleri dikkate alındığında önerilen IDGKO yönteminin, 25 veri setinin 20'sinde diğer yöntemlere göre eşit veya daha iyi sonuçlar elde ettiği görülmektedir. GAP değerleri dikkate alındığında ise IDGKO'nun, 25 veri setinin 17'sinde 50 bağımsız çalışmanın tümü için optimum değere ulaştığı görülmektedir. Önerilen IDGKO yönteminin geride kaldığı durumlarda başarı sıralaması değeri 2 olup, en iyi yönleme yakın sonuçlar elde ettiği görülmektedir. Çizelge 5.6'da verilen Friedman ve ortalama başarı sıralaması değerleri dikkate alındığında önerilen IDGKO yöntemi, 1.20 ortalama başarı sıralama değeri ve 2.66 Friedman sıralama değeri ile diğer 7 yöntemi geride bırakarak ilk sırada yer almaktadır. Çizelge 5.5 ve 5.6'daki verilere dayanarak, IDGKO'nun sonuçlar açısından rekabetçi olan BinEHO ve CGMA ikili yöntemlerinden farkı; boyut arttıkça bile tutarlı sonuçlar almaya devam etmiş olmasıdır.

Çizelge 5.7'de, önerilen yöntemin sonuçları 7 farklı ikili yöntemle işaret bazında ve GAP değeri açısından karşılaştırılmıştır. Çizelgede yer alan işaretlerin açıklaması şu şekildedir:

- Önerilen yöntem, karşılaştırılan yöntemden daha iyi sonuçlar elde ettiyse → '+'
- Önerilen yöntemin, karşılaştırılan yöntemden daha kötü bir sonucu varsa → '-'
- Önerilen yöntem, karşılaştırılan yöntemle aynı sonuca ulaşırsa → '='

Çizelge 5.7. IDGKO'nun diğer ikili yöntemlerle işaret bazlı kıyaslama sonuçları (Hakli, 2020)

Veri seti	BinEHO	BPSO	MBPSO	NGHS	DGHS	S-bAFSA	CGMA
D1	=	+	=	=	=	=	=
D2	=	+	=	=	=	=	=
D3	=	+	=	=	=	=	=
D4	=	+	=	=	=	=	=
D5	=	+	+	=	=	=	=
D6	+	+	+	+	+	+	=
D7	=	+	+	+	=	=	=
D8	=	+	+	+	+	=	=
D9	=	+	+	+	=	=	=
D10	=	+	=	+	+	=	=
D11	+	+	+	+	=	=	=
D12	=	+	+	+	=	=	=
D13	=	+	+	+	=	=	=
D14	-	+	+	+	+	+	-
D15	+	+	+	+	+	+	+
D16	=	+	+	+	=	=	=
D17	=	+	+	+	=	=	=
D18	-	+	+	+	+	+	+
D19	-	+	+	+	+	+	+
D20	+	+	+	+	+	+	-
D21	+	+	+	+	+	+	+
D22	=	+	+	+	=	=	=
D23	-	+	+	+	+	+	+
D24	+	+	+	+	+	+	+
D25	=	+	+	+	=	=	=
+/-/	6/15/4	25/0/0	20/5/0	20/5/0	11/14/0	9/16/0	6/17/2

Çizelge 5.7'de görüldüğü gibi, IDGKO yöntemi tüm veri kümelerinde BPSO yönteminden daha iyi performans göstermiştir. IDGKO yöntemi, MBPSO ve NGHS yöntemleri ile karşılaştırıldığında 20 veri setinde daha iyi sonuçlar elde etmiştir. Kalan yöntemlerle yapılan karşılaştırma sonuçlarına bakıldığında, IDGKO'nun galibiyet elde ettiği veri seti sayısı, yenilgi elde ettiği veri seti sayısından daha fazladır. Şunu da belirtmek gerekir ki; IDGKO yöntemi, karşılaştırılan diğer 5 algoritma (BPSO, MBPSO, NGHS, DGHS, S -bAFSA) tarafından hiçbir zaman yenilgiye uğramamıştır. Tüm bu sonuçlardan yola çıkarak, IDGKO'nun 25 veri seti için diğer yöntemlere göre daha başarılı olduğu söylenebilir.

DeneySEL Çalışma 1 ile yapılan karşılaştırma sonucunda önerilen IDGKO yönteminin 0-1 KP için tutarlı, başarılı ve sağlam sonuçlar elde ettiği söylenebilir.

5.2.2. DeneySEL Çalışma 2

Özsoydan (2019), GWO'daki baskın bireylerin arama yeteneğine olan etkilerini analiz etmek amacıyla baskın kurtların varyasyonlarına dayanan yeni uzantılar

önermektedir. Geliştirilen yöntemlerin performansları, kapasitesiz tesis yerleştirme problemi ve 0-1 KP gibi kombinatoryal problemler de dahil olmak üzere hem kısıtlı hem de kısıtsız optimizasyon problemlerinde test edilmiştir. Önerilen yöntemler, literatürdeki diğer metasezgisel yöntemlerle performans açısından kıyaslanmıştır. Yapılan deneyler sonucunda, baskın kurtların GWO performansı üzerinde çok önemli etkileri olduğunu kanıtlanmıştır (Ozsoydan, 2019).

Bahsi geçen çalışma, önceki bölümlerde de yer verildiği üzere bu tez çalışmasının temel ilham kaynaklarından biridir. Çalışmada; GWO'daki baskın bireylerin çözüm kalitesi üzerindeki etkisinin amaçlanması ve 0-1 KP üzerinde deneyler gerçekleştirilmesi sebebiyle tez çalışmasında önerilen IDGKO ile performans açısından kıyaslanmıştır. Deneyler sırasında kullanılan veri kümeleri, parametre ayarları, performans belirleme ölçütleri, çizelgeler ve veriler doğrudan ilgili çalışmadan alınmıştır.

(Ozsoydan, 2019)'da 0-1 KP için kullanılan veri setleri iki kısımda kategorize edilmiştir. Birinci kısımda (Zou ve ark., 2011)'dan alınan 10 standart veri seti kullanılmıştır. İkinci kısımda ise, büyük ölçekli ve daha zorlayıcı kıyaslama veri setleri kullanılmıştır.

Deneylerin ilk kısmında kullanılacak olan 10 standart veri seti hakkında detaylı bilgiler Çizelge 5.8'de sunulmuştur.

Çizelge 5.8. Deneyleerde kullanılan birinci kısım veri setlerine ait bilgiler (Zou ve ark., 2011)

Veri seti	Nesne Sayısı (Boyut)	Çanta Kapasitesi	Ağırlık Değerleri (w)	Kâr Değerleri (p)	Optimum Çözüm Vektörü (x^*)	Optimum Değer
F1	10	269	$w=(95,4,60,32,23,72,80,62,65,4,6)$	$p=(55,10,47,5,4,50,8,61,85,87)$	$x^*=(0,1,1,1,0,0,0,1,1,1)$	295
F2	20	878	$w=(92,4,43,83,84,68,92,82,6,44,32,18,56,83,25,96,70,48,14,58)$	$p=(44,46,90,72,91,40,75,35,8,54,78,40,77,15,61,17,75,29,75,63)$	$x^*=(1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,1,0,1,1)$	1024
F3	4	20	$w=(6,5,9,7)$	$p=(9,11,13,15)$	$x^*=(1,1,0,1)$	35
F4	4	11	$w=(2,4,6,7)$	$p=(6,10,12,13)$	$x^*=(0,1,0,1)$	23
F5	15	375	$w=(56.358531,80.874050,47.987304,89.596240,74.660482,85.894345,51.353496,1.498459,36.445204,16.589862,44.569231,0.466933,37.788018,57.118442,60.716575)$	$p=(0.125126,19.330424,58.500931,35.029145,82.284005,17.410810,71.050142,30.399487,9.140294,14.731285,98.852504,11.908322,0.891140,53.166295,60.176397)$	$x^*=(0,0,1,0,1,0,1,1,0,1,1,1,0,1,1)$	481.0694
F6	10	60	$w=(30,25,20,18,17,11,5,2,1,1)$	$p=(20,18,17,15,15,10,5,3,1,1)$	$x^*=(0,0,1,0,1,1,1,1,0,0)$	50
F7	7	50	$w=(31,10,20,19,4,3,6)$	$p=(70,20,39,37,7,5,10)$	$x^*=(1,0,0,1,0,0,0)$	107
F8	23	10000	$w=(983,982,981,980,979,978,488,976,972,486,486,972,972,485,485,969,966,483,964,963,961,958,959)$	$p=(981,980,979,978,977,976,487,974,970,485,485,970,970,484,484,976,974,482,962,961,959,958,857)$	$x^*=(1,1,1,1,1,1,1,1,0,1,0,0,0,0,0,1,1,0,0,0,0,0)$	9767
F9	5	80	$w=(15,20,17,8,31)$	$p=(33,24,36,37,12)$	$x^*=(1,1,1,1,0)$	130
F10	20	879	$w=(84,83,43,4,44,6,82,92,25,83,56,18,58,14,48,70,96,32,68,92)$	$p=(91,72,90,46,55,8,35,75,61,15,77,40,63,75,29,75,17,78,40,44)$	$x^*=(1,1,1,1,1,1,1,1,0,1,1,1,1,0,1,0,1,1,1,1)$	1025

İkinci kısımdaki veri setlerine ait parametreler farklı ölçeklerdedir. Nesnelerin ağırlık ve kâr değerleri belirli aralıklar dahilinde rastgele belirlenmektedir. Popülasyon sayısı ve maksimum iterasyon sayısı, her veri seti için farklı değerlere sahiptir. İkinci kısımda kullanılan veri setleri rastgele verilerden oluşacağından, optimum çözümleri bilinmemektedir. Bu sebeple, ikinci kısım veri setleri için Dantzig Üst Sınırı (Dantzig Upper Bound, DUB) kullanılmıştır. DUB ile optimum çözüm değeri hesaplanırken Denklem (5.2) kullanılmaktadır.

$$DUB = \sum_{r=1}^{s-1} p_r + (C - \sum_{r=1}^{s-1} w_r) \times \frac{w_r}{p_r} \quad (5.2)$$

Denklem (5.2)'de matematiksel formülü verilen yöntemin temel mantığı şu şekildedir: Nesneler artmayan kâr/ağırlık sırasına göre sıralanır. Sıralı nesneler sırt çantası kapasitesini aşmadığı sürece sırayla sırt çantasına dahil edilir. Böylece ilgili veri setinin optimum sonucu tespit edilmiş olur (Ozsoydan, 2019).

Deneylerin ikinci kısmında kullanılan ve parametreleri belirli aralıklar dahilinde rastgele oluşturulacak olan veri setlerine ait detaylı bilgiler Çizelge 5.9'da verilmiştir.

Çizelge 5.9. Deneylerde kullanılan ikinci kısım veri setlerine ait bilgiler (Ozsoydan, 2019)

Veri seti	Popülasyondaki Birey Sayısı	İterasyon Sayısı	Sırt Çantası Kapasitesi
F1	100	15.000	11.100
F2	200	15.000	1500
F3	300	20.000	1700
F4	500	20.000	2000
F5	800	30.000	5000
F6	1000	30.000	10.000
F7	1200	40.000	14.000
F8	1500	40.000	16.000

Ozsoydan (2019) tarafından önerilen yöntemler, birinci kısım veri setleri kullanılarak PSO ve GWO yöntemleriyle performans bakımından kıyaslanmıştır. Çizelge 5.10, birinci kısımdaki veri setleri için performans sonuçlarını göstermektedir. Kıyasa tabii tutulan tüm yöntemler ve IDGKO için popülasyondaki birey sayısı 50, maksimum iterasyon sayısı 1000 ve bağımsız çalışma sayısı 30 olarak belirlenmiştir. Tez kapsamında önerilen IDGKO, aynı parametre değerlerini kullanarak diğer yöntemlerle karşılaştırılmıştır. Çizelge 5.10'da kıyasa tabii tutulan yöntemler ve IDGKO'nun ortalama uygunluk değeri (M), GAP değeri (G) ve optimum çözümün bulunduğu

çalıştırma sayısını temsil eden isabet değeri (hit, H) (Ozsoydan, 2019) bilgileri sunulmuştur.



Çizelge 5.10'daki sonuçlara bakıldığında, tüm algoritmaların rekabetçi bir yaklaşımla küçük boyutlu veri kümelerinde optimum çözümü elde ettiği görülmektedir.

Çizelge 5.11, ikinci kısımdaki veri setleri için IDGKO ve diğer yöntemlerin sonuçlarını içermektedir. Popülasyondaki her bir birey için ağırlık değerleri [5,20] ve kar değerleri [50,100] aralığında rastgele belirlenmiştir. Tüm yöntemler için 20 bağımsız çalışma gerçekleştirilmiş olup; sonuçlar en iyi uygunluk değeri (B), ortalama uygunluk değeri (M), GAP değeri (G) ve algoritma performansı bazında sıralama (R) değeri şeklinde Çizelge 5.11'de sunulmuştur.



Çizelge 5.11. Algoritmaların ikinci kısım veri setleri için en iyi uygunluk, ortalama uygunluk, GAP ve sıralama değerleri (Ozsoydan, 2019)

Veri Seti	Nesne Sayısı (Boyut)	Optimum Değer	IDGKO				prioGWO				learnGWO				prLeGWO				GWO				PSO			
			<i>B</i>	<i>M</i>	<i>G</i>	<i>R</i>	<i>B</i>	<i>M</i>	<i>G</i>	<i>R</i>	<i>B</i>	<i>M</i>	<i>G</i>	<i>R</i>	<i>B</i>	<i>M</i>	<i>G</i>	<i>R</i>	<i>B</i>	<i>M</i>	<i>G</i>	<i>R</i>	<i>B</i>	<i>M</i>	<i>G</i>	<i>R</i>
F11	100	7016,50	7003	6996,60	1.90E-1	1	7001	6994,96	2.20E-1	4	7003	6989,36	1.90E-1	5	7003	6995,26	1.90E-1	3	7003	6996,30	1.90E-1	2	6999	6985,93	2.40E-1	6
F12	200	11639,21	11639	11549,10	1.00E-3	2	11614	11545,16	2.10E-1	3	11552	11405,30	7.40E-1	5	11624	11571,66	1.30E-1	1	11574	11509,30	5.60E-1	4	10878	10763,23	6.54E+3	6
F13	300	14254,57	14253	14105,90	1.00E-2	1	14118	14010,63	9.50E-1	4	14036	13812,60	1.53E+3	5	14193	14102,40	4.30E-1	2	14152	14017,36	7.10E-1	3	12034	11900,30	15.50E+3	6
F14	500	18640,57	18215	18050,25	2.28E+3	1	18003	17651,80	3.42E+3	3	17478	17037,53	6.23E+3	5	18206	18020,10	2.33E+3	2	17952	17619,43	3.69E+3	4	14463	14208,83	22.40E+3	6
F15	800	40607,35	40513	40110,95	2.30E-1	1	39971	39645,56	1.56E+3	2	38711	37936,96	4.66E+3	5	39307	38878,10	3.20E+3	4	39779	39263,53	2.03E+3	3	33485	33111,96	17.50E+3	6
F16	1000	67256,33	67255	66505,20	1.00E-3	1	66238	65942,50	1.51E+3	4	65389	65130,93	2.77E+3	5	66528	66315,96	1.08E+3	2	66154	65990,63	1.63E+3	3	63069	62900,53	6.20E+3	6
F17	1200	86994,11	86800	86599,63	2.20E-1	1	86690	86593,40	3.40E-1	2	86225	86026,00	8.80E-1	5	86698	86516,20	3.40E-1	3	86611	86482,86	4.40E-1	4	85370	85207,70	1.86E+3	6
F18	1500	103745,10	103740	103522,50	4.00E-2	1	102529	102166,60	1.17E+3	3	101534	101080,13	2.13+3	5	102797	102521,73	9.10E-1	2	102453	102132,50	1.24E+3	4	99033	98715,30	4.54E+3	6

Çizelge 5.11’de sunulan verilerin daha kolay yorumlanabilmesi adına, her bir algoritma elde ettiği GAP değerleri üzerinden Friedman Testi’ne tabii tutulmuş ve sonuçlar Çizelge 5.12’de verilmiştir. Yine, algoritmaların elde ettiği başarı sıralamasına göre ortalaması hesaplanarak Çizelge 5.12’de sunulmuştur.

Çizelge 5.12. Algoritmaların Friedman ve ortalama başarı sıralaması bakımından kıyaslama sonuçları (Ozsoydan, 2019)

Algoritma	Friedman Sıralama Değeri	Ortalama Başarı Sıralaması Değeri
IDGKO	1.19	1.125
prioGWO	3.19	3.125
learnGWO	4.69	5.000
prLeGWO	2.38	2.375
GWO	3.56	3.375
PSO	6.00	6.000

Deneysel çalışma 2 ile yapılan karşılaştırmalar sonucunda IDGKO ve diğer ikili yöntemler birinci kısımda kullanılan veri setlerinin (F1-F10) tamamında optimum sonuca ulaşmış ve berabere kalmışlardır. Bu nedenle ikinci kısımda kullanılan veri setleri (F11-F18) yöntemlerin birbirlerine olan üstünlüklerinin/başarılarının belirlenmesinde önemlidir. Çizelge 5.11, deneylerde kullanılan her bir veri seti için 20 bağımsız çalışmanın en iyi uygunluk değeri, ortalama uygunluk değeri ve GAP değerlerini sunmaktadır. Ayrıca karşılaştırılan algoritmalar için ortalama değer cinsinden sıralamayı gösteren sıra değeri sunulmuştur. Çizelge 5.11’de verilen ortalama değerler dikkate alındığında, önerilen IDGKO yöntemi 12 veri setinin 11’inde diğer yöntemlere göre daha iyi sonuçlar elde etmiştir. Önerilen IDGKO yönteminin yetersiz kaldığı tek bir veri kümesi (F12) vardır. IDGKO’nun bu veri seti için sıralaması 2 olup en iyi yönteme yakın sonuçlar elde etmiştir. IDGKO ayrıca en iyi değer açısından diğer yöntemler arasında ilk sıradadır. Çizelge 5.12’de verilen Friedman Sıralama ve Ortalama Başarı Sıralaması değerleri dikkate alındığında, önerilen IDGKO yöntemi 1.125 ortalama başarı sıralaması değeri ve 1.19 Friedman sıralama değeri ile diğer 5 yöntemi geride bırakarak ilk sırada yer almaktadır. Çizelge 5.11’deki verilere göre, sonuçlar açısından rekabetçi olan prioGWO ve prLeGWO ikili yöntemlerinden parametre değerleri arttıkça bile IDGKO’nun daha tutarlı sonuçlar aldığı görülmektedir. Ayrıca onarım fonksiyonu kullanan IDGKO yönteminin diğer yöntemlere göre daha başarılı sonuçlar ürettiği ve onarım algoritmasının etkisinin dikkat çekici olduğu gözlemlenmiştir.

5.2.3. Deneysel Çalışma 3

Abdel-Basset ve arkadaşları (2021a), literatüre sürekli optimizasyon problemlerinin çözümü için kazandırılan Deniz Avcıları Algoritması'nın (Marine Predators Algorithm, MPA) yeni ikili versiyonunu (BMPA) önermiştir (Abdel-Basset ve ark., 2021a). Önerilen yöntemde ikili arama uzayına uyarlama kısmında, S-şekilli ve V-şekilli transfer fonksiyonları kullanılmıştır. Çalışmada ilk olarak en uygun sonucu veren transfer fonksiyonu seçimi yapılmış ve daha sonra yöntemin performansı 0-1 KP üzerinde test edilmiştir. BMPA'nın performansı literatüre kazandırılan birçok yöntem ile kıyaslanmış ve kıyaslamalar sonucunda önerilen yöntemin başarısı kanıtlanmıştır.

Çalışmada, deneyler esnasında kullanılan veri setleri doğrudan BMPA çalışmasından temin edilmiştir (Ortega; Ortega; Pisinger, 2005; Abdel-Basset ve ark., 2021a). Veri setleri hakkında detaylı bilgi Çizelge 5.13'te sunulmuştur.

Çizelge 5.13. Deneylerde kullanılan veri setlerine ait bilgiler (Ortega)

Veri Seti ID	Veri Seti	Sırt Çantası Kapasitesi	Nesne Sayısı (Boyut)	Optimum Değer
f1	KP1_100	995	100	9147
f2	KP1_200	1008	200	11.238
f3	KP1_500	2543	500	28.857
f4	KP1_1000	5002	1000	54.503
f5	KP1_2000	10.011	2000	110.625
f6	KP2_100	995	100	1514
f7	KP2_200	1008	200	1634
f8	KP2_500	2543	500	4566
f9	KP2_1000	5002	1000	9052
f10	KP2_2000	10.011	2000	18.051
f11	KP3_100	997	100	2397
f12	KP3_200	997	200	2697

Deneyler sırasında belirlenen parametre ayarları doğrudan BMPA çalışmasından referans alınmış olup, Çizelge 5.14'te sunulmuştur.

Çizelge 5.14. Deneylerde kullanılan veri setlerine ait bilgiler (Abdel-Basset ve ark., 2021a)

Parametre Adı	Parametre Değeri
Popülasyondaki Birey Sayısı	20
İterasyon Sayısı	5000
Maksimum Bağımsız Çalıştırma Sayısı	20

Abdel-Basset ve arkadaşları (2021a) tarafından geliştirilen BMPA, orijinal makalesinde 4 farklı yöntem (SA (Ezugwu ve ark., 2019), Hybrid IGA-SA (Ezugwu ve ark., 2019), GSA (Ezugwu ve ark., 2019), BWOA (Mirjalili ve Lewis, 2016)) ile kıyas

edilmiştir. Çizelge 5.15'te BMPA, diğer yöntemler ve IDGKO'nun ortalama uygunluk değeri (M), GAP değeri (G) ve algoritma performansı bazında sıralama (R) değeri bilgileri sunulmuştur.



Çizelge 5.15. Algoritmaların sınıflandırma doğruluğu, ortalama uygunluk, en iyi uygunluk ve sıralama değerleri (Abdel-Basset ve ark., 2021a)

Veri Seti	Optimum	IDGKO			BMPA			SA			Hybrid IGA-SA			GSA			BWOA		
		M	G	R	M	G	R	M	G	R	M	G	R	M	G	R	M	G	R
f1	9147	9147	0	1	9147	0	1	9147	0	1	8575	6.25E+3	2	2983	67.38E+3	3	9147	0	1
f2	11.238	11.238	0	1	11.238	0	1	10.163	9.56E+3	2	8576	23.68E+3	4	9865	12.21E+3	3	11.238	0	1
f3	28.857	28.838,30	6.00E-2	3	28.857	0	1	21.390	25.87E+3	4	12.072	58.16E+3	5	9865	65.81E+3	6	28.845,50	3.00E-2	2
f4	54.503	54.425,90	1.40E-1	1	54.408,75	1.70E-1	2	36.719	32.62E+3	4	14.563	73.28E+3	6	14.927	72.61E+3	5	53.382,20	2.05E+3	3
f5	110.625	110.592,80	2.00E-2	1	96.490,50	12.77E+3	2	95.739	13.45E+3	3	27.645	75.01E+3	5	25.579	76.87E+3	6	93.053	15.88E+3	4
f6	1514	1514	0	1	1514	0	1	1486	1.84E+3	2	1217	19.61E+3	3	1041	31.24E+3	4	1514	0	1
f7	1634	1634	0	1	1634	0	1	1537	5.93E+3	2	1347	17.56E+3	3	1073	34.33E+3	4	1634	0	1
f8	4566	4561,65	9.00E-2	1	4557	1.90E-1	2	3744	18.00E+3	4	3038	33.46E+3	5	2951	35.37E+3	6	4525,50	8.80E-1	3
f9	9052	9045,40	7.00E-2	1	8919,25	1.46E+3	2	6831	24.53E+3	4	5435	39.95E+3	6	5675	37.30E+3	5	8287	8.45E+3	3
f10	18.051	17.832	1.21E+3	1	17.167	4.89E+3	2	12.780	29.20E+3	4	10.938	39.40E+3	6	11.064	38.70E+3	5	14.827,45	17.85E+3	3
f11	2397	2397	0	1	2397	0	1	2296	4.21E+3	3	1481	38.21E+3	4	1095	54.31E+3	5	2396,70	1.00E-2	2
f12	2697	2697	0	1	2697	0	1	2594	3.81E+3	3	1495	44.56E+3	4	1095	59.39E+3	5	2695,50	5.00E-2	2

Çizelge 5.15'te sunulan verilerin daha kolay yorumlanabilmesi adına, her bir algoritma elde ettiği GAP değerleri üzerinden Friedman Testi'ne tabii tutulmuş ve sonuçlar Çizelge 5.16'da verilmiştir. Yine, algoritmaların elde ettiği başarı sıralamasına göre ortalaması hesaplanarak Çizelge 5.16'de sunulmuştur.

Çizelge 5.16. Algoritmaların Friedman ve ortalama başarı sıralaması bakımından kıyaslama sonuçları

Algoritma	Friedman Sıralama Değeri	Ortalama Başarı Sıralaması Değeri
IDGKO	1.63	1.16
BMPA	1.88	1.41
SA	3.79	3.00
Hybrid IGA-SA	5.33	4.41
GSA	5.67	4.75
BWOA	2.71	2.16

Çizelge 5.15, deneylerde kullanılan her bir veri seti için 20 bağımsız çalışmanın ortalamasını ve GAP değerlerini göstermektedir. Ayrıca karşılaştırılan algoritmalar için başarı sıralamasını gösteren sıra değeri sunulmuştur. Çizelge 5.15'te verilen ortalama uygunluk değerleri dikkate alındığında, önerilen IDGKO yöntemi 12 veri setinin 11'inde diğer yöntemlere eşit veya daha iyi sonuçlar elde etmiştir. IDGKO, GAP değerleri bakımından 12 veri setinin 6'sında 20 bağımsız çalışmanın tümü için optimum değere ulaşmıştır. Çizelge 5.15'te görüldüğü üzere, IDGKO yöntemi yalnızca f3 veri setinde yenilgiye uğramıştır. Ancak önerilen IDGKO yönteminin geride kaldığı durumda başarı sıralaması 3 olup, en iyi yöntemeye yakın sonuçlar elde etmiştir. Çizelge 5.16'da verilen Friedman Sıralama ve Ortalama Başarı Sıralaması değerleri dikkate alındığında, önerilen IDGKO yöntemi 1.16 ortalama başarı sıralaması değeri ve 1.63 Friedman sıralama değeri ile diğer 5 yöntemi geride bırakarak ilk sırada yer almaktadır. Çizelge 5.15'teki verilere göre, sonuçlar açısından rekabetçi olan BMPA ve BWOA ikili yöntemlerinden IDGKO'nun boyut arttıkça bile daha tutarlı sonuçlar elde ettiği görülmüştür. Ek olarak, Çizelge 5.15 ve Çizelge 5.16'daki verilere göre, önerilen IDGKO'nun büyük boyutlu veri kümeleri için rekabetçi ve başarılı olduğu açıkça görülmektedir. Önceki bölümlerde bahsi geçen transfer fonksiyon kullanımının keşif ve sömürü açısından getirdiği dezavantajlar sebebiyle de diğer yöntemlerin IDGKO'ya göre geride kaldığı söylenebilir.

6. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında, ikili optimizasyon problemlerinin çözümü için GWO algoritmasına dayalı yeni bir ikili IDGKO yöntemi önerilmiştir. GWO, sürekli optimizasyon problemlerinin çözümü için önerildiğinden ikili optimizasyon problemlerine doğrudan uygulanamamaktadır. Bu nedenle, GWO'nun orijinal konum güncelleme denklemine bağlı kalarak; XOR lojik-kapı tabanlı operatörünü kullanan yeni bir konum güncelleme denklemi önerildi. Ayrıca, tezin ana motivasyon kaynaklarından biri olan, GWO'daki baskın üç bireyin çözüm kalitesi üzerindeki etkisini artırmayı ve buna bağlı olarak çözümü iyileştirmeyi amaçlayan NDCA yöntemi önerildi. Bu yöntemlerle IDGKO'nun yerel arama ile küresel arama arasında etkin bir denge kurabilmesi amaçlanmaktadır. Yine, erken yakınsamayı önlemek ve yerel arama ile küresel arama arasında etkin bir denge sağlamak için onarım fonksiyonu algoritması (repair function) kullanıldı. Önerilen yöntemin amacına uygun sonuçlara ulaşp ulaşmadığını belirlemek için hem FS hem de 0-1 KP üzerinde farklı boyut ve özelliklere sahip birçok veri seti ve literatüre kazandırılan yakın tarihli yöntemler ile çeşitli deneyler yapılmıştır.

IDGKO yönteminin FS problemi üzerinde doğruluğunu ve başarısını tespit etmek için, literatüre yakın tarihte kazandırılmış ikili GWO yöntemleri ve diğer ikili metasezgisel yöntemlerden oluşan 4 farklı çalışma ile performans bakımından kıyaslaması yapılmıştır. Karşılaştırma yapılan ilk çalışmada IDGKO yöntemi; 8 adet ikili GWO algoritması, GA ve PSO ile 10 farklı veri seti üzerinde kıyaslanmıştır. Literatürdeki transfer fonksiyonu ve genetik operatör kullanılarak geliştirilen yöntemler ile yapılan deneyler sonucunda, IDGKO'nun 10 veri setinin 9'unda ortalama uygunluk değeri başarı sıralaması bakımından ilk sırada yer aldığı ve diğer yöntemleri geride bıraktığı gözlemlenmiştir. Karşılaştırma yapılan ikinci çalışmada IDGKO yöntemi; GWO ile yapılan hibrit algoritmalar, ikili diğer metasezgisel algoritmalar ve ikili GWO algoritmaları olmak üzere 8 adet yöntem ile çeşitli performans belirleme ölçütleri kullanılarak kıyaslanmıştır. Kıyaslamalar sonucunda IDGKO yönteminin, hibrit yöntemlere ve diğer ikili yöntemlere kıyasla 18 veri setinin büyük çoğunluğunda %90 ve üzerinde sınıflandırma doğruluğu elde ettiği ve yine birçok veri setinde yüksek oranla en iyi özellik alt kümesini tespit ettiği doğrulanmaktadır.

IDGKO yönteminin 0-1 KP üzerinde doğruluğunu ve başarısını tespit etmek için, literatüre yakın tarihte kazandırılmış ikili GWO yöntemleri ve diğer ikili metasezgisel

yöntemlerden oluşan 3 farklı çalışma ile performans bakımından kıyaslaması yapılmıştır. Karşılaştırılan ilk çalışmada IDGKO, 25 veri seti üzerinde 7 farklı ikili yöntemle karşılaştırılmıştır. Bu yöntemler arasında özellikle rekabetçi ve başarılı olanlar BinEHO, DGHS, S-bAFSA ve CGMA yöntemleridir. Ancak IDGKO, üç baskın bireyin çözüm kalitesi üzerindeki etkisini dengeli bir şekilde belirleyen NDCA ile arama uzayını verimli kullanarak bu yöntemleri geride bırakmıştır. Karşılaştırılan ikili yöntemlere uygulanan Friedman istatistiksel testi sonucunda en iyi değeri IDGKO elde etmiştir. Karşılaştırılan ikinci çalışmada, GWO'daki üç baskın bireyin çözüm kalitesi üzerindeki etkisini iyileştirmek için 3 farklı ikili yöntem önerilmiştir. IDGKO, 18 veri seti üzerinde bahsi geçen 3 ikili GWO algoritması, orijinal GWO ve PSO olmak üzere 5 farklı ikili yöntemle karşılaştırılmıştır. Yapılan deneyler sonucu; tüm yöntemlerin ilk 10 veri setinde optimum sonuca ulaştığı ve berabere kaldıkları gözlemlenmiştir. Geriye kalan 8 veri setinin 7'sinde IDGKO galibiyet elde ederek hakimiyetini korumuştur. Karşılaştırılan ikili yöntemlere uygulanan istatistiksel test sonucunda en iyi sonucu IDGKO elde etmiştir. Önerilen IDGKO'nun, karşılaştırma yapılan ikinci deneysel çalışma yöntemlerinden bir farkı onarım algoritması kullanmasıdır. Bu nedenle, IDGKO'nun başarısında hem kullanılan basit ama verimli yöntemlerin hem de onarım algoritmasının etkili olduğu söylenebilir. Karşılaştırılan üçüncü çalışmada IDGKO, 12 büyük ölçekli veri seti üzerinde 5 farklı ikili yöntemle karşılaştırılmıştır. Bu yöntemler arasında özellikle rekabetçi ve başarılı olanlar BMPA ve BWOA yöntemleridir. IDGKO, 12 veri setinden 6'sında en başarılı yöntem olarak ilk sırada yer almaktadır. Karşılaştırılan ikili yöntemlere uygulanan istatistiksel test sonucunda en iyi sonucu IDGKO elde etmiştir. Üçüncü deneysel çalışmada önerilen BMPA'nın ikilileştirme işleminde, transfer fonksiyonları kullanıldığından ve bu işlemin önceki kısımlarda belirtilen dezavantajlarından dolayı yöntemlerin geride kaldığı söylenebilir.

Bu tez kapsamında, FS için önerilen IDGKO yöntemi, farklı parametre ayarlarına ve boyutlarına sahip toplam 19 veri seti üzerinde test edilmiştir. IDGKO, literatürde başarısı ve etkinliği kanıtlanmış 18 farklı ikili yöntem ile performans belirleme ölçütleri kullanılarak karşılaştırılmıştır. 0-1 KP için önerilen IDGKO yöntemi, farklı parametre ayarlarına ve boyutlarına sahip toplam 55 veri seti üzerinde test edilmiştir. IDGKO, literatürde başarısı ve etkinliği kanıtlanmış 17 farklı ikili yöntem ile en iyi uygunluk değeri, ortalama uygunluk değeri ve GAP değerleri açısından karşılaştırılmıştır. Yine yöntemlerin istatistiksel analizi için her birine Friedman testi uygulanmıştır. FS ve 0-1 KP üzerinde gerçekleştirilen tüm deneyler sonucunda; IDGKO'nun çözüm kalitesi

açısından basit ama oldukça başarılı ve verimli bir yöntem olduğu söylenebilir. IDGKO, GWO'nun etkin yapısına bağlı kalarak önerilen basit, verimli, kolay uygulanabilir ve başarılı bir ikili yöntemdir. IDGKO gelecekteki çalışmalar için bir motivasyon kaynağı olabilme kapasitesine sahiptir.

Bu tez çalışması kapsamında önerilen IDGKO ikili yöntemi lojik-kapı tabanlı XOR operatörü kullanılarak ikili optimizasyon problemlerinin çözümü için geliştirilmiş ve literatürde adından sıkça söz ettiren birçok çalışmanın konusu olan FS ve 0-1 KP problemlerine uygulanmıştır. Bahsi geçen yöntem, önceki bölümlerde yer verilen diğer ikili optimizasyon problemlerine uyarlanıp başarısı ve doğruluğu tespit edilebilir. Yine, önerilen ikilileştirme yöntemi literatüre kazandırılmış metasezgisel diğer yöntemlere entegre edilerek yeni ikili yöntemler geliştirilebilir. Bu noktada, sürü zekasına veya genetiğe dayalı hibrit algoritmalara benzer yöntemler uyarlanarak sonuçlar daha da iyileştirilebilir. Kullanılan parametre değerleri, detaylı parametre analizi yapılarak optimize edilebilir. IDGKO, farklı özelliklere ve zorluklara sahip diğer veri kümeleri üzerinde test edilebilir. Böylece yöntemin tutarlılığı konusunda daha güçlü sonuçlar elde edilmiş olacaktır. Ayrıca yapay sinir ağları, destek vektör makineleri gibi yöntemler kullanılarak IDGKO sınıflandırma problemlerine de uygulanabilir.

7. KAYNAKLAR

- Abasi, A. K., Khader, A. T., Al-Betar, M. A., Naim, S., Makhadmeh, S. N. ve Alyasseri, Z. A. A., 2021, An improved text feature selection for clustering using binary grey wolf optimizer, *Proceedings of the 11th national technical seminar on unmanned system technology 2019*, 503-516.
- Abdel-Basset, M., Luo, Q., Miao, F. ve Zhou, Y., 2017, Solving 0–1 knapsack problems by binary dragonfly algorithm, *International conference on intelligent computing*, 491-502.
- Abdel-Basset, M. ve Zhou, Y., 2018, An elite opposition-flower pollination algorithm for a 0-1 knapsack problem, *International Journal of Bio-Inspired Computation*, 11 (1), 46-53.
- Abdel-Basset, M., El-Shahat, D. ve El-Henawy, I., 2019a, Solving 0–1 knapsack problem by binary flower pollination algorithm, *Neural Computing and Applications*, 31 (9), 5477-5495.
- Abdel-Basset, M., El-Shahat, D. ve Sangaiah, A. K., 2019b, A modified nature inspired meta-heuristic whale optimization algorithm for solving 0–1 knapsack problem, *International Journal of Machine Learning and Cybernetics*, 10 (3), 495-514.
- Abdel-Basset, M., Mohamed, R., Chakraborty, R. K., Ryan, M. ve Mirjalili, S., 2021a, New binary marine predators optimization algorithms for 0–1 knapsack problems, *Computers & Industrial Engineering*, 151, 106949.
- Abdel-Basset, M., Mohamed, R. ve Mirjalili, S., 2021b, A binary equilibrium optimization algorithm for 0–1 knapsack problems, *Computers & Industrial Engineering*, 151, 106946.
- Abdollahzadeh, B., Barshandeh, S., Javadi, H. ve Epicoco, N., 2021, An enhanced binary slime mould algorithm for solving the 0–1 knapsack problem, *Engineering with Computers*, 1-22.
- Abedinpourshotorban, H., Shamsuddin, S. M., Beheshti, Z. ve Jawawi, D. N., 2016, Electromagnetic field optimization: a physics-inspired metaheuristic optimization algorithm, *Swarm and Evolutionary Computation*, 26, 8-22.
- Abualigah, L. ve Diabat, A., 2020, A comprehensive survey of the Grasshopper optimization algorithm: results, variants, and applications, *Neural Computing and Applications*, 32 (19), 15533-15556.
- Abualigah, L., Diabat, A. ve Geem, Z. W., 2020a, A comprehensive survey of the harmony search algorithm in clustering applications, *Applied Sciences*, 10 (11), 3827.
- Abualigah, L., Shehab, M., Alshinwan, M. ve Alabool, H., 2020b, Salp swarm algorithm: a comprehensive survey, *Neural Computing and Applications*, 32 (15), 11195-11215.
- Abualigah, L., 2021, Group search optimizer: a nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications, *Neural Computing and Applications*, 33 (7), 2949-2972.
- Abualigah, L., Diabat, A., Mirjalili, S., Abd Elaziz, M. ve Gandomi, A. H., 2021a, The arithmetic optimization algorithm, *Computer methods in applied mechanics and engineering*, 376, 113609.
- Abualigah, L., Shehab, M., Alshinwan, M., Mirjalili, S. ve Elaziz, M. A., 2021b, Ant lion optimizer: a comprehensive survey of its variants and applications, *Archives of Computational Methods in Engineering*, 28 (3), 1397-1416.

- Aghdam, M. H., Ghasem-Aghaee, N. ve Basiri, M. E., 2009, Text feature selection using ant colony optimization, *Expert Systems with Applications*, 36 (3), 6843-6853.
- Agrawal, P., Abutarboush, H. F., Ganesh, T. ve Mohamed, A. W., 2021, Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019), *IEEE Access*, 9, 26766-26791.
- Agushaka, J. O. ve Ezugwu, A. E., 2021, Advanced arithmetic optimization algorithm for solving mechanical engineering design problems, *PLoS One*, 16 (8), e0255703.
- Ahmed, S., Zhang, M. ve Peng, L., 2013, Enhanced feature selection for biomarker discovery in LC-MS data using GP, *2013 IEEE Congress on Evolutionary Computation*, 584-591.
- Akay, B., 2009, Nümerik optimizasyon problemlerinde yapay arı kolonisi (artificial bee colony) algoritmasının performans analizi, Doktora Tezi, *Erciyes Üniversitesi Fen Bilimleri Enstitüsü*.
- Al-Tashi, Q., Kadir, S. J. A., Rais, H. M., Mirjalili, S. ve Alhussian, H., 2019, Binary optimization using hybrid grey wolf optimization for feature selection, *IEEE Access*, 7, 39496-39508.
- Al-Tashi, Q., Abdulkadir, S. J., Rais, H. M., Mirjalili, S., Alhussian, H., Ragab, M. G. ve Alqushaibi, A., 2020, Binary multi-objective grey wolf optimizer for feature selection in classification, *IEEE Access*, 8, 106247-106263.
- Al-Wajih, R., Abdulkadir, S. J., Aziz, N., Al-Tashi, Q. ve Talpur, N., 2021, Hybrid binary grey wolf with Harris hawks optimizer for feature selection, *IEEE Access*, 9, 31662-31677.
- Alamri, H. S., Zamli, K. Z., Razak, M. F. A. ve Firdaus, A., 2019, Solving 0/1 knapsack problem using opposition-based whale optimization algorithm (OWOA), *Proceedings of the 2019 8th International Conference on Software and Computer Applications*, 135-139.
- Alatas, B., 2011, ACROA: artificial chemical reaction optimization algorithm for global optimization, *Expert Systems with Applications*, 38 (10), 13170-13180.
- Alzaqebah, M., Briki, K., Alrefai, N., Brini, S., Jawarneh, S., Alsmadi, M. K., Mohammad, R. M. A., ALmarashdeh, I., Alghamdi, F. A. ve Aldhafferri, N., 2021, Memory based cuckoo search algorithm for feature selection of gene expression dataset, *Informatics in Medicine Unlocked*, 24, 100572.
- Arora, S., Singh, H., Sharma, M., Sharma, S. ve Anand, P., 2019, A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection, *IEEE Access*, 7, 26343-26361.
- Aslan, M., Gunduz, M. ve Kiran, M. S., 2019, JayaX: Jaya algorithm with xor operator for binary optimization, *Applied Soft Computing*, 82, 105576.
- Atashpaz-Gargari, E. ve Lucas, C., 2007, Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition, *2007 IEEE congress on evolutionary computation*, 4661-4667.
- Aydın, H. M., 2019, Wi-fi based indoor positioning with spatial filtering and feature selection / Özel filtreleme ve özellik seçimi ile wi-fi tabanlı iç konum analizi, Yüksek Lisans Tezi, *Bahçeşehir Üniversitesi / Fen Bilimleri Enstitüsü / Elektrik-Elektronik Mühendisliği Ana Bilim Dalı*.
- Azad, M. A. K., Rocha, A. M. A. ve Fernandes, E. M., 2014, A simplified binary artificial fish swarm algorithm for 0-1 quadratic knapsack problems, *Journal of Computational and Applied mathematics*, 259, 897-904.
- Bae, H.-O., Ha, S.-Y., Kang, M., Lim, H., Min, C. ve Yoo, J., 2022, A constrained consensus based optimization algorithm and its application to finance, *Applied Mathematics and Computation*, 416, 126726.

- Baliarsingh, S. K., Vipsita, S., Muhammad, K. ve Bakshi, S., 2019, Analysis of high-dimensional biomedical data using an evolutionary multi-objective emperor penguin optimizer, *Swarm and Evolutionary Computation*, 48, 262-273.
- Baliarsingh, S. K. ve Vipsita, S., 2020, Chaotic emperor penguin optimised extreme learning machine for microarray cancer classification, *IET Systems Biology*, 14 (2), 85-95.
- Banitalebi, A., Abd Aziz, M. I. ve Aziz, Z. A., 2016, A self-adaptive binary differential evolution algorithm for large scale binary optimization problems, *Information Sciences*, 367, 487-511.
- Bansal, J. C. ve Deep, K., 2012, A modified binary particle swarm optimization for knapsack problems, *Applied Mathematics and Computation*, 218 (22), 11042-11061.
- Basheer, G. T. ve Algamal, Z. Y., 2021, Improving Flower Pollination Algorithm for Solving 0–1 Knapsack Problem, *Journal of Physics: Conference Series*, 022097.
- Bekdaş, G., Nigdeli, S. M., Kayabekir, A. E. ve Yang, X.-S., 2019, Optimization in civil engineering and metaheuristic algorithms: a review of state-of-the-art developments, *Computational intelligence, optimization and inverse problems with applications in engineering*, 111-137.
- Berberler, M. E., 2009, Sırt çantası problem türleri ve uygulamaları, Knapsack type problems and applications, Doktora Tezi, *Ege Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Ana Bilim Dalı*.
- Bezerra, M. A., Santelli, R. E., Oliveira, E. P., Villar, L. S. ve Escalera, L. A., 2008, Response surface methodology (RSM) as a tool for optimization in analytical chemistry, *Talanta*, 76 (5), 965-977.
- Bhattacharjee, K. K. ve Sarmah, S. P., 2014, Shuffled frog leaping algorithm and its application to 0/1 knapsack problem, *Applied Soft Computing*, 19, 252-263.
- Bolón-Canedo, V., Sánchez-Marroño, N. ve Alonso-Betanzos, A., 2015, Recent advances and emerging challenges of feature selection in the context of big data, *Knowledge-Based Systems*, 86, 33-45.
- Cao, J., Yin, B., Lu, X., Kang, Y. ve Chen, X., 2018, A modified artificial bee colony approach for the 0-1 knapsack problem, *Applied Intelligence*, 48 (6), 1582-1595.
- Černý, V., 1985, Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm, *Journal of optimization theory and applications*, 45 (1), 41-51.
- Chang, S., Li, A., Wang, X. ve Wang, X., 2022, Joint optimization of e-commerce supply chain financing strategy and channel contract, *European Journal of Operational Research*.
- Cinar, A. C. ve Kiran, M. S., 2018, Similarity and logic gate-based tree-seed algorithms for binary optimization, *Computers & Industrial Engineering*, 115, 631-646.
- Cobos, C., Dulcey, H., Ortega, J., Mendoza, M. ve Ordoñez, A., 2016, A binary fisherman search procedure for the 0/1 knapsack problem, *Conference of the Spanish Association for Artificial Intelligence*, 447-457.
- Cui, X., Li, Y., Fan, J., Wang, T. ve Zheng, Y., 2020, A hybrid improved dragonfly algorithm for feature selection, *IEEE Access*, 8, 155619-155629.
- Çelik, Y., 2013, Optimizasyon problemlerinde bal arıları evlilik optimizasyonu algoritmasının (marriage in honey bee optimization-MBO) performansının geliştirilmesi.
- Dantzig, G. B., 1957, Discrete-variable extremum problems, *Operations research*, 5 (2), 266-288.

- Dasgupta, D. ve Michalewicz, Z., 2013, Evolutionary algorithms in engineering applications, Springer Science & Business Media, p.
- Dash, M. ve Liu, H., 1997, Feature selection for classification, *Intelligent data analysis*, 1 (1-4), 131-156.
- Deep, K., 2022, A Random Walk Grey Wolf Optimizer based on dispersion factor for feature selection on Chronic Disease Prediction, *Expert Systems with Applications*, 206, 117864.
- Devanathan, K., Ganapathy, N. ve Swaminathan, R., 2019, Binary grey wolf optimizer based feature selection for nucleolar and centromere staining pattern classification in indirect immunofluorescence images, *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 7040-7043.
- Dhiman, G., Oliva, D., Kaur, A., Singh, K. K., Vimal, S., Sharma, A. ve Cengiz, K., 2021, BEPO: A novel binary emperor penguin optimizer for automatic feature selection, *Knowledge-Based Systems*, 211, 106560.
- Dorigo, M., Birattari, M. ve Stutzle, T., 2006, Ant colony optimization, *IEEE computational intelligence magazine*, 1 (4), 28-39.
- Dönmez, H., 2020, Düello optimizasyon algoritmasının ikili optimizasyon problemlerine uygulanması, Doktora Tezi, *Karabük Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Ana Bilim Dalı*.
- Du, H., Wu, X. ve Zhuang, J., 2006, Small-world optimization algorithm for function optimization, *International conference on natural computation*, 264-273.
- Emary, E., Zawbaa, H. M. ve Hassanien, A. E., 2016a, Binary grey wolf optimization approaches for feature selection, *Neurocomputing*, 172, 371-381.
- Emary, E., Zawbaa, H. M. ve Hassanien, A. E., 2016b, Binary ant lion approaches for feature selection, *Neurocomputing*, 213, 54-65.
- Erol, O. K. ve Eksin, I., 2006, A new optimization method: big bang–big crunch, *Advances in engineering software*, 37 (2), 106-111.
- Ezugwu, A. E., Pillay, V., Hirasen, D., Sivanarain, K. ve Govender, M., 2019, A comparative study of meta-heuristic optimization algorithms for 0–1 knapsack problem: Some initial results, *IEEE Access*, 7, 43979-44001.
- Faramarzi, A., Heidarinejad, M., Stephens, B. ve Mirjalili, S., 2020, Equilibrium optimizer: A novel optimization algorithm, *Knowledge-Based Systems*, 191, 105190.
- Faris, H., Mafarja, M. M., Heidari, A. A., Aljarah, I., Ala'M, A.-Z., Mirjalili, S. ve Fujita, H., 2018, An efficient binary salp swarm algorithm with crossover scheme for feature selection problems, *Knowledge-Based Systems*, 154, 43-67.
- Feng, Y., Jia, K. ve He, Y., 2014, An improved hybrid encoding cuckoo search algorithm for 0-1 knapsack problems, *Computational intelligence and neuroscience*, 2014.
- Feng, Y., Wang, G.-G., Deb, S., Lu, M. ve Zhao, X.-J., 2017, Solving 0–1 knapsack problem by a novel binary monarch butterfly optimization, *Neural Computing and Applications*, 28 (7), 1619-1634.
- Feng, Y., Wang, G.-G., Dong, J. ve Wang, L., 2018, Opposition-based learning monarch butterfly optimization with Gaussian perturbation for large-scale 0-1 knapsack problem, *Computers & Electrical Engineering*, 67, 454-468.
- Firouzi, A. ve Meshkani, A., 2021, Risk-based optimization of the debt service schedule in renewable energy project finance, *Utilities Policy*, 70, 101197.
- Fix, E. ve Hodges, J. L., 1989, Discriminatory analysis. Nonparametric discrimination: Consistency properties, *International Statistical Review/Revue Internationale de Statistique*, 57 (3), 238-247.

- Fogel, D. B., 1998, Artificial intelligence through simulated evolution, Wiley-IEEE Press, p.
- Formato, R., 2007, Central force optimization: a new metaheuristic with applications in applied electromagnetics. *Prog Electromagn Res* 77: 425–491.
- Gandomi, A. H. ve Alavi, A. H., 2012, Krill herd: a new bio-inspired optimization algorithm, *Communications in nonlinear science and numerical simulation*, 17 (12), 4831-4845.
- García, J., Crawford, B., Soto, R. ve Astorga, G., 2019, A clustering algorithm applied to the binarization of swarm intelligence continuous metaheuristics, *Swarm and Evolutionary Computation*, 44, 646-664.
- Geem, Z. W., Kim, J. H. ve Loganathan, G. V., 2001, A new heuristic optimization algorithm: harmony search, *simulation*, 76 (2), 60-68.
- Ghanem, W. A. H. ve Jantan, A., 2016, Novel Multi-Objective Artificial Bee Colony Optimization for Wrapper Based Feature Selection in Intrusion Detection, *International Journal of Advances in Soft Computing & Its Applications*, 8 (1).
- Gholizadeh, S., Danesh, M. ve Gheytratmand, C., 2020, A new Newton metaheuristic algorithm for discrete performance-based design optimization of steel moment frames, *Computers & Structures*, 234, 106250.
- Ghorbani, N. ve Babaei, E., 2014, Exchange market algorithm, *Applied Soft Computing*, 19, 177-187.
- Ghosh, A., Datta, A. ve Ghosh, S., 2013, Self-adaptive differential evolution for feature selection in hyperspectral image data, *Applied Soft Computing*, 13 (4), 1969-1977.
- Glover, F., 1989, Tabu search—part I, *ORSA Journal on computing*, 1 (3), 190-206.
- Glover, F., 1990, Tabu search—part II, *ORSA Journal on computing*, 2 (1), 4-32.
- Gunavathi, C. ve Premalatha, K., 2015, Cuckoo search optimisation for feature selection in cancer classification: a new approach, *International journal of data mining and bioinformatics*, 13 (3), 248-265.
- Gupta, S. ve Deep, K., 2020, A memory-based grey wolf optimizer for global optimization tasks, *Applied Soft Computing*, 93, 106367.
- Hafez, A. I., Hassanien, A. E., Zawbaa, H. M. ve Emary, E., 2015, Hybrid monkey algorithm with krill herd algorithm optimization for feature selection, *2015 11th international computer engineering conference (ICENCO)*, 273-277.
- Hakli, H., 2020, BinEHO: a new binary variant based on elephant herding optimization algorithm, *Neural Computing and Applications*, 32 (22), 16971-16991.
- Han, N., Qiao, S., Yuan, G., Huang, P., Liu, D. ve Yue, K., 2019, A novel Chinese herbal medicine clustering algorithm via artificial bee colony optimization, *Artificial intelligence in medicine*, 101, 101760.
- Hashim, F. A., Houssein, E. H., Mabrouk, M. S., Al-Atabany, W. ve Mirjalili, S., 2019, Henry gas solubility optimization: A novel physics-based algorithm, *Future Generation Computer Systems*, 101, 646-667.
- Hatamlou, A., 2013, Black hole: A new heuristic optimization approach for data clustering, *Information Sciences*, 222, 175-184.
- Hegazy, A. E., Makhlof, M. ve El-Tawel, G. S., 2020, Improved salp swarm algorithm for feature selection, *Journal of King Saud University-Computer and Information Sciences*, 32 (3), 335-344.
- Heidari, A. A. ve Pahlavani, P., 2017, An efficient modified grey wolf optimizer with Lévy flight for optimization tasks, *Applied Soft Computing*, 60, 115-134.
- Holland, J. H., 1992, Genetic algorithms, *Scientific american*, 267 (1), 66-73.
- Hu, P., Pan, J.-S. ve Chu, S.-C., 2020, Improved binary grey wolf optimizer and its application for feature selection, *Knowledge-Based Systems*, 195, 105746.

- Huang, C.-F., 2012, Combinatorial optimization in biology using probability collectives multi-agent systems, *Expert Systems with Applications*, 39 (2), 1763-1771.
- Hussien, A. G., Hassanien, A. E., Houssein, E. H., Bhattacharyya, S. ve Amin, M., 2019, S-shaped binary whale optimization algorithm for feature selection, In: Recent trends in signal and image processing, Eds: Springer, p. 79-87.
- Ibrahim, H. T., Mazher, W. J., Uçan, O. N. ve Bayat, O., 2017, Feature selection using salp swarm algorithm for real biomedical datasets.
- James, J. ve Li, V. O., 2015, A social spider algorithm for global optimization, *Applied Soft Computing*, 30, 614-627.
- Jayabarathi, T., Raghunathan, T., Adarsh, B. ve Suganthan, P. N., 2016, Economic dispatch using hybrid grey wolf optimizer, *Energy*, 111, 630-641.
- Kalaycı, G., 2019, Application of metaheuristics for the feature selection problem / Metasezgisellerin özellik seçimi problemi için uygulanması, Yüksek Lisans, *Marmara Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Ana Bilim Dalı*.
- Kallioras, N. A., Lagaros, N. D. ve Avtzis, D. N., 2018, Pity beetle algorithm—a new metaheuristic inspired by the behavior of bark beetles, *Advances in engineering software*, 121, 147-166.
- Kalra, M., Kumar, V., Kaur, M., Idris, S. A., Öztürk, Ş. ve Alshazly, H., 2022, A Novel Binary Emperor Penguin Optimizer for Feature Selection Tasks, *Comput. Mater. Contin.*, 70, 6239-6255.
- Karaboga, D., 2005, An idea based on honey bee swarm for numerical optimization, *Technical report-tr06, Erciyes university, engineering faculty, computer ...*
- Karaboğa, D., 2014, Yapay zeka optimizasyon algoritmaları, Nobel Akademi Yayıncılık, p.
- Karakoyun, M. ve Özkış, A., Transfer Fonksiyonları Kullanarak İkili Güve-Alev Optimizasyonu Algoritmalarının Geliştirilmesi ve Performanslarının Karşılaştırılması, *Necmettin Erbakan Üniversitesi Fen ve Mühendislik Bilimleri Dergisi*, 3 (2), 1-10.
- Karakoyun, M., Onur, I. ve İhtisam, A., 2019, Grey Wolf Optimizer (GWO) Algorithm to Solve the Partitional Clustering Problem, *International Journal of Intelligent Systems and Applications in Engineering*, 7 (4), 201-206.
- Karakoyun, M., 2021, Çok amaçlı optimizasyon problemlerinin çözümü için Kurbağa Sıçrama ve Gri Kurt Optimizasyonu algoritmaları tabanlı hibrit bir yöntemin geliştirilmesi.
- Karakoyun, M., Gülcü, Ş. ve Kodaz, H., 2021, D-MOSG: Discrete multi-objective shuffled gray wolf optimizer for multi-level image thresholding, *Engineering Science and Technology, an International Journal*, 24 (6), 1455-1466.
- Karizaki, A. A. ve Tavassoli, M., 2019, A novel hybrid feature selection based on ReliefF and binary dragonfly for high dimensional datasets, *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, 300-304.
- Kaveh, A. ve Talatahari, S., 2010, A novel heuristic optimization method: charged system search, *Acta mechanica*, 213 (3), 267-289.
- Kaveh, A. ve Khayatazad, M., 2012, A new meta-heuristic method: ray optimization, *Computers & Structures*, 112, 283-294.
- Kaveh, A., 2014, Colliding bodies optimization, In: Advances in metaheuristic algorithms for optimal design of structures, Eds: Springer, p. 195-232.
- Kaveh, A. ve Mahdavi, V. R., 2014, Colliding bodies optimization: a novel meta-heuristic method, *Computers & Structures*, 139, 18-27.

- Kennedy, J. ve Eberhart, R., 1995, Particle swarm optimization, *Proceedings of ICNN'95-international conference on neural networks*, 1942-1948.
- Kennedy, J. ve Eberhart, R. C., 1997, A discrete binary version of the particle swarm algorithm, *1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation*, 4104-4108.
- Kiliç, Ü. ve Keleş, M. K., 2018, Feature selection with artificial bee colony algorithm on Z-Alizadeh Sani dataset, *2018 innovations in intelligent systems and applications conference (asyu)*, 1-3.
- Kiran, M. S. ve Gündüz, M., 2013a, XOR-based artificial bee colony algorithm for binary optimization, *Turkish Journal of Electrical Engineering & Computer Sciences*, 21 (Sup. 2), 2307-2328.
- Kiran, M. S. ve Gündüz, M., 2013b, XOR-based artificial bee colony algorithm for binary optimization, *Turkish Journal of Electrical Engineering and Computer Sciences*, 21 (8), 2307-2328.
- Kirkpatrick, S., Gelatt Jr, C. D. ve Vecchi, M. P., 1983, Optimization by simulated annealing, *science*, 220 (4598), 671-680.
- Korkmaz, S., 2019, İkili optimizasyon problemlerinin çözümü için yapay alg algoritması tabanlı yeni yaklaşımlar.
- Koza, J. R., 1992, Evolution of subsumption using genetic programming, *Proceedings of the first European conference on artificial life*, 110-119.
- Krogh, C., Bak, B. L., Lindgaard, E., Olesen, A. M., Hermansen, S. M., Broberg, P. H., Kepler, J. A., Lund, E. ve Jakobsen, J., 2021, A simple MATLAB draping code for fiber-reinforced composites with application to optimization of manufacturing process parameters, *Structural and Multidisciplinary Optimization*, 64 (1), 457-471.
- Kumar, V. ve Minz, S., 2014, Feature selection: a literature review, *SmartCR*, 4 (3), 211-229.
- Lanza-Gutierrez, J. M., Crawford, B., Soto, R., Berrios, N., Gomez-Pulido, J. A. ve Paredes, F., 2017, Analyzing the effects of binarization techniques when solving the set covering problem through swarm optimization, *Expert Systems with Applications*, 70, 67-82.
- Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J. ve Liu, H., 2017a, Feature selection: A data perspective, *ACM computing surveys (CSUR)*, 50 (6), 1-45.
- Li, Q., Chen, H., Huang, H., Zhao, X., Cai, Z., Tong, C., Liu, W. ve Tian, X., 2017b, An enhanced grey wolf optimization based feature selection wrapped kernel extreme learning machine for medical diagnosis, *Computational and mathematical methods in medicine*, 2017.
- Li, X.-l., 2002, An optimizing method based on autonomous animats: fish-swarm algorithm, *Systems Engineering-Theory & Practice*, 22 (11), 32-38.
- Ling, W., Fu, X., Menhas, M. ve Fei, M., 2010, A modified binary differential evolution algorithm. *Life System Modeling and Intelligent Computing*, Springer Berlin Heidelberg.
- Mafarja, M., Eleyan, D., Abdullah, S. ve Mirjalili, S., 2017a, S-shaped vs. V-shaped transfer functions for ant lion optimization algorithm in feature selection problem, *Proceedings of the international conference on future networks and distributed systems*, 1-7.
- Mafarja, M. ve Mirjalili, S., 2018, Whale optimization approaches for wrapper feature selection, *Applied Soft Computing*, 62, 441-453.

- Mafarja, M., Qasem, A., Heidari, A. A., Aljarah, I., Faris, H. ve Mirjalili, S., 2020, Efficient hybrid nature-inspired binary optimizers for feature selection, *Cognitive Computation*, 12 (1), 150-175.
- Mafarja, M. M., Eleyan, D., Jaber, I., Hammouri, A. ve Mirjalili, S., 2017b, Binary dragonfly algorithm for feature selection, *2017 International conference on new trends in computing sciences (ICTCS)*, 12-17.
- Mafarja, M. M. ve Mirjalili, S., 2017b, Hybrid whale optimization algorithm with simulated annealing for feature selection, *Neurocomputing*, 260, 302-312.
- Majidpour, H. ve Soleimanian Gharehchopogh, F., 2018, An improved flower pollination algorithm with AdaBoost algorithm for feature selection in text documents classification, *Journal of Advances in Computer Research*, 9 (1), 29-40.
- Masadeh, R., 2021, Whale Optimization Algorithm Applied To The 0/1 Knapsack Problem, *Journal of Theoretical and Applied Information Technology*, 99 (7).
- Mehedi, I. M., Ahmadipour, M., Salam, Z., Ridha, H. M., Bassi, H., Rawa, M. J. H., Ajour, M., Abusorrah, A. ve Abdullah, M. P., 2021, Optimal feature selection using modified cuckoo search for classification of power quality disturbances, *Applied Soft Computing*, 113, 107897.
- Mirjalili, S. ve Lewis, A., 2013, S-shaped versus V-shaped transfer functions for binary particle swarm optimization, *Swarm and Evolutionary Computation*, 9, 1-14.
- Mirjalili, S., Mirjalili, S. ve Lewis, A., 2014a, Grey wolf optimizer *Advances in Engineering Software*. 69 46–61.
- Mirjalili, S., Mirjalili, S. M. ve Lewis, A., 2014b, Grey wolf optimizer, *Advances in engineering software*, 69, 46-61.
- Mirjalili, S., 2015, The ant lion optimizer, *Advances in engineering software*, 83, 80-98.
- Mirjalili, S., 2016, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, *Neural Computing and Applications*, 27 (4), 1053-1073.
- Mirjalili, S. ve Lewis, A., 2016, The whale optimization algorithm, *Advances in engineering software*, 95, 51-67.
- Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H. ve Mirjalili, S. M., 2017, Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems, *Advances in engineering software*, 114, 163-191.
- Mittal, N., Singh, U. ve Sohi, B. S., 2016, Modified grey wolf optimizer for global engineering optimization, *Applied Computational Intelligence and Soft Computing*, 2016.
- Moghaddam, F. F., Moghaddam, R. F. ve Cheriet, M., 2012, Curved space optimization: a random search based on general relativity theory, *arXiv preprint arXiv:1208.2214*.
- Muro, C., Escobedo, R., Spector, L. ve Coppinger, R., 2011, Wolf-pack (*Canis lupus*) hunting strategies emerge from simple rules in computational simulations, *Behavioural processes*, 88 (3), 192-197.
- Naveenkumar, R. ve Baskar, G., 2021, Process optimization, green chemistry balance and technoeconomic analysis of biodiesel production from castor oil using heterogeneous nanocatalyst, *Bioresource Technology*, 320, 124347.
- Nguyen, P. H., Wang, D. ve Truong, T. K., 2017, A novel binary social spider algorithm for 0–1 knapsack problem, *Int. J. Innov. Comput. Inf. Control*, 13 (6), 2039-2049.
- Ortega, J., Instances of 0/1 Knapsack Problem, http://artemisa.unicauca.edu.co/~johnyortega/instances_01_KP/:
- Ozsoydan, F. B., 2019, Effects of dominant wolves in grey wolf optimization algorithm, *Applied Soft Computing*, 83, 105658.

- Ozturk, C., Hancer, E. ve Karaboga, D., 2015, A novel binary artificial bee colony algorithm based on genetic operators, *Information Sciences*, 297, 154-170.
- Özger, Z. B., Bolat, B. ve Dırı, B., 2016, A comparative study on binary artificial bee colony optimization methods for feature selection, *2016 International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, 1-4.
- Özger, Z. B., 2019, Büyük Boyutlu Verilerde Öznitelik Seçimi İçin İkili Yapay Arı Kolonisi Yaklaşımı, Doktora Tezi, *Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı*.
- Özkış, A., 2018, Çok Amaçlı Problemlerin Çözümü İçin Yeni Sezgisel Algoritma Geliştirilmesi, Doktora Tezi, *Selçuk Üniversitesi Fen Bilimleri Enstitüsü*.
- Pampará, G. ve Engelbrecht, A. P., 2011, Binary artificial bee colony optimization, *2011 IEEE Symposium on Swarm Intelligence*, 1-8.
- Pandey, A. C., Rajpoot, D. S. ve Saraswat, M., 2020, Feature selection method based on hybrid data transformation and binary binomial cuckoo search, *Journal of Ambient Intelligence and Humanized Computing*, 11 (2), 719-738.
- Pisinger, D., 2005, Where are the hard knapsack problems?, *Computers & Operations Research*, 32 (9), 2271-2284.
- Purushothaman, R., Rajagopalan, S. ve Dhandapani, G., 2020, Hybridizing Gray Wolf Optimization (GWO) with Grasshopper Optimization Algorithm (GOA) for text feature selection and clustering, *Applied Soft Computing*, 96, 106651.
- Rajamohana, S. P., Umamaheswari, K. ve Abirami, B., 2017, Adaptive binary flower pollination algorithm for feature selection in review spam detection, *2017 International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT)*, 1-4.
- Ramezani, F. ve Lotfi, S., 2013, Social-based algorithm (SBA), *Applied Soft Computing*, 13 (5), 2837-2856.
- Rani, M., 2021, Employing Artificial Bee Colony Algorithm for Feature Selection in Intrusion Detection System, *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, 496-500.
- Rani, R. R. ve Ramyachitra, D., 2017, Krill herd optimization algorithm for cancer feature selection and random forest technique for classification, *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 109-113.
- Rao, H., Shi, X., Rodrigue, A. K., Feng, J., Xia, Y., Elhoseny, M., Yuan, X. ve Gu, L., 2019, Feature selection based on artificial bee colony and gradient boosting decision tree, *Applied Soft Computing*, 74, 634-642.
- Rao, R. V., Savsani, V. J. ve Vakharia, D., 2011, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, *Computer-aided design*, 43 (3), 303-315.
- Rao, R. V., Savsani, V. J. ve Vakharia, D., 2012, Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems, *Information Sciences*, 183 (1), 1-15.
- Rashedi, E., Nezamabadi-Pour, H. ve Saryazdi, S., 2009, GSA: a gravitational search algorithm, *Information Sciences*, 179 (13), 2232-2248.
- Rechenberg, I., 1978, Evolutionsstrategien, In: *Simulationsmethoden in der Medizin und Biologie*, Eds: Springer, p. 83-114.
- Remeseiro, B. ve Bolon-Canedo, V., 2019, A review of feature selection methods in medical applications, *Computers in biology and medicine*, 112, 103375.
- Repository, U. M. L., 2021/2022, <https://archive.ics.uci.edu/ml/datasets.php>:

- Rizk-Allah, R. M. ve Hassanien, A. E., 2018, New binary bat algorithm for solving 0–1 knapsack problem, *Complex & Intelligent Systems*, 4 (1), 31-53.
- Roayaei, M., 2021, On the binarization of Grey Wolf optimizer: a novel binary optimizer algorithm, *Soft Computing*, 25 (23), 14715-14728.
- Rodrigues, D., Pereira, L. A., Almeida, T., Papa, J. P., Souza, A., Ramos, C. C. ve Yang, X.-S., 2013, BCS: A binary cuckoo search algorithm for feature selection, *2013 IEEE International symposium on circuits and systems (ISCAS)*, 465-468.
- Rodrigues, D., Pereira, L. A., Nakamura, R. Y., Costa, K. A., Yang, X.-S., Souza, A. N. ve Papa, J. P., 2014a, A wrapper approach for feature selection based on bat algorithm and optimum-path forest, *Expert Systems with Applications*, 41 (5), 2250-2258.
- Rodrigues, D., Pereira, L. A., Papa, J. P. ve Weber, S. A., 2014b, A binary krill herd approach for feature selection, *2014 22nd International Conference on Pattern Recognition*, 1407-1412.
- Sadollah, A., Sayyaadi, H., Lee, H. M. ve Kim, J. H., 2018, Mine blast harmony search: a new hybrid optimization method for improving exploration and exploitation capabilities, *Applied Soft Computing*, 68, 548-564.
- Saraç, T., 2007, Genelleştirilmiş karesel çoklu sırt çantası problemi için melez bir çözüm yaklaşımı / A hybrid solution method for the generalized quadratic multiple knapsack problem, Doktora Tezi, *Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Ana Bilim Dalı*.
- Sathiyabhama, B., Kumar, S. U., Jayanthi, J., Sathiya, T., Ilavarasi, A., Yuvarajan, V. ve Gopikrishna, K., 2021, A novel feature selection framework based on grey wolf optimizer for mammogram image analysis, *Neural Computing and Applications*, 33 (21), 14583-14602.
- Sawhney, R. ve Jain, R., 2018, Modified binary dragonfly algorithm for feature selection in human papillomavirus-mediated disease treatment, *2018 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, 91-95.
- Sayed, S. A.-F., Nabil, E. ve Badr, A., 2016, A binary clonal flower pollination algorithm for feature selection, *Pattern recognition letters*, 77, 21-27.
- Shah-Hosseini, H., 2011, Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation, *International Journal of Computational Science and Engineering*, 6 (1-2), 132-140.
- Shang, H., Zhang, K., Guan, Z. ve Zhang, X., 2022, Optimization of evidence-based research in the prevention and treatment of coronary heart disease with traditional Chinese medicine: A comprehensive review, *Journal of Traditional Chinese Medical Sciences*.
- Shi, H., 2006, Solution to 0/1 knapsack problem based on improved ant colony algorithm, *2006 IEEE International Conference on Information Acquisition*, 1062-1066.
- Shortt, R. L., Jones, L. M. ve Pino, L., 2022, Optimization of a processing and analysis workflow for proteome-wide structural biology, *Biophysical Journal*, 121 (3), 320a.
- Simon, D., 2008, Biogeography-based optimization, *IEEE transactions on evolutionary computation*, 12 (6), 702-713.
- Singh, A. ve Pandey, B., 2016, An euclidean distance based KNN computational method for assessing degree of liver damage, *2016 International Conference on Inventive Computation Technologies (ICICT)*, 1-4.

- Song, X.-f., Zhang, Y., Gong, D.-w. ve Sun, X.-y., 2021, Feature selection using bare-bones particle swarm optimization with mutual information, *Pattern Recognition*, 112, 107804.
- Sonuc, E., 2021, Binary crow search algorithm for the uncapacitated facility location problem, *Neural Computing and Applications*, 33 (21), 14669-14685.
- Storn, R. ve Price, K., 1997, Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces, *Journal of global optimization*, 11 (4), 341-359.
- Subbaih, T. ve Ramasamy, U., 2022, A stochastic constrained optimization on multi-echelon renovated supply chain with sustainable production and transportation, *Cleaner Engineering and Technology*, 100547.
- Sudha, M. ve Selvarajan, S., 2016, Feature selection based on enhanced cuckoo search for breast cancer classification in mammogram image, *Circuits and Systems*, 7 (04), 327.
- Swiniarski, R. W. ve Skowron, A., 2003, Rough set methods in feature selection and recognition, *Pattern recognition letters*, 24 (6), 833-849.
- Taha, H. A., 2000, Yöneylem araştırması, Literatür, p.
- Tawhid, M. A. ve Dsouza, K. B., 2018, Hybrid binary bat enhanced particle swarm optimization algorithm for solving feature selection problems, *Applied Computing and Informatics*.
- Tawhid, M. A. ve Ibrahim, A. M., 2020, Feature selection based on rough set approach, wrapper approach, and binary whale optimization algorithm, *International Journal of Machine Learning and Cybernetics*, 11 (3), 573-602.
- Too, J., Abdullah, A. R., Mohd Saad, N., Mohd Ali, N. ve Tee, W., 2018, A new competitive binary grey wolf optimizer to solve the feature selection problem in EMG signals classification, *Computers*, 7 (4), 58.
- Too, J., Abdullah, A. R. ve Mohd Saad, N., 2019a, A new quadratic binary harris hawk optimization for feature selection, *Electronics*, 8 (10), 1130.
- Too, J., Abdullah, A. R., Mohd Saad, N. ve Tee, W., 2019b, EMG feature selection and classification using a Pbest-guide binary particle swarm optimization, *Computation*, 7 (1), 12.
- Uygur, A., 2010, Sırt çantası problemlerinin çözüm yöntemleri ve bir uygulaması / Solution methods of knapsack problems and an application, Yüksek Lisans Tezi, *Ege Üniversitesi, Fen Bilimleri Enstitüsü, İstatistik Ana Bilim Dalı*.
- Vásquez, C., Lemus-Romani, J., Crawford, B., Soto, R., Astorga, G., Palma, W., Misra, S. ve Paredes, F., 2020, Solving the 0/1 Knapsack problem using a galactic swarm optimization with data-driven binarization approaches, *International Conference on Computational Science and Its Applications*, 511-526.
- Wang, G.-G., Deb, S. ve Cui, Z., 2019, Monarch butterfly optimization, *Neural Computing and Applications*, 31 (7), 1995-2014.
- Wang, J., Lin, D., Zhang, Y. ve Huang, S., 2022, An adaptively balanced grey wolf optimization algorithm for feature selection on high-dimensional classification, *Engineering Applications of Artificial Intelligence*, 114, 105088.
- Wang, L., Pan, Z. ve Wang, J., 2021a, A review of reinforcement learning based intelligent optimization for manufacturing scheduling, *Complex System Modeling and Simulation*, 1 (4), 257-270.
- Wang, L., Shi, R. ve Dong, J., 2021b, A hybridization of dragonfly algorithm optimization and angle modulation mechanism for 0-1 knapsack problems, *Entropy*, 23 (5), 598.

- Wang, X.-h., Zhang, Y., Sun, X.-y., Wang, Y.-l. ve Du, C.-h., 2020, Multi-objective feature selection based on artificial bee colony: An acceleration approach with variable sample size, *Applied Soft Computing*, 88, 106041.
- Wang, Y. ve Wang, W., 2021, Quantum-inspired differential evolution with grey wolf optimizer for 0-1 knapsack problem, *Mathematics*, 9 (11), 1233.
- Webster, B. ve Bernhard, P. J., 2003, A local search optimization algorithm based on natural principles of gravitation.
- Xiang, W.-l., An, M.-q., Li, Y.-z., He, R.-c. ve Zhang, J.-f., 2014, A novel discrete global-best harmony search algorithm for solving 0-1 knapsack problems, *Discrete Dynamics in Nature and Society*, 2014.
- Yan, C., Ma, J., Luo, H., Zhang, G. ve Luo, J., 2019, A novel feature selection method for high-dimensional biomedical data based on an improved binary clonal flower pollination algorithm, *Human heredity*, 84 (1), 34-46.
- Yan, C., Li, M., Ma, J., Liao, Y., Luo, H., Wang, J. ve Luo, J., 2022, A Novel Feature Selection Method Based on MRMR and Enhanced Flower Pollination Algorithm for High Dimensional Biomedical Data, *Current Bioinformatics*, 17 (2), 133-149.
- Yang, B., Lu, Y., Zhu, K., Yang, G., Liu, J. ve Yin, H., 2017, Feature selection based on modified bat algorithm, *IEICE TRANSACTIONS on Information and Systems*, 100 (8), 1860-1869.
- Yang, X.-S., 2009, Firefly algorithms for multimodal optimization, *International symposium on stochastic algorithms*, 169-178.
- Yang, X.-S., 2010a, A new metaheuristic bat-inspired algorithm, In: Nature inspired cooperative strategies for optimization (NICSO 2010), Eds: Springer, p. 65-74.
- Yang, X.-S., 2010b, Engineering optimization: an introduction with metaheuristic applications, John Wiley & Sons, p.
- Yassien, E., Masadeh, R., Alzaqebah, A. ve Shaheen, A., 2017, Grey wolf optimization applied to the 0/1 knapsack problem, *International Journal of Computer Applications*, 169 (5), 11-15.
- Yong, R., Chang, C., Zhang, D., Wu, J., Huang, H., Jing, D. ve Zheng, J., 2021, Optimization of shale-gas horizontal well spacing based on geology–engineering–economy integration: a case study of Well Block Ning 209 in the National Shale Gas Development Demonstration Area, *Natural Gas Industry B*, 8 (1), 98-104.
- Yongjian, F., 1997, Data mining: tasks, techniques and applications, *IEEE Potentials*, 16 (4), 18-20.
- Yu, X., Xu, W. ve Li, C., 2021, Opposition-based learning grey wolf optimizer for global optimization, *Knowledge-Based Systems*, 226, 107139.
- Zawbaa, H. M., Emary, E. ve Parv, B., 2015, Feature selection based on antlion optimization algorithm, *2015 Third world conference on complex systems (WCCS)*, 1-7.
- Zawbaa, H. M. ve Emary, E., 2018, Applications of flower pollination algorithm in feature selection and knapsack problems, In: Nature-Inspired Algorithms and Applied Optimization, Eds: Springer, p. 217-243.
- Zhang, G., Hou, J., Wang, J., Yan, C. ve Luo, J., 2020, Feature selection for microarray data classification using hybrid information gain and a modified binary krill herd algorithm, *Interdisciplinary Sciences: Computational Life Sciences*, 12 (3), 288-301.
- Zhang, X. ve Liu, S., 2008, Interval algorithm for global numerical optimization, *Engineering Optimization*, 40 (9), 849-868.
- Zhou, Y., Chen, X. ve Zhou, G., 2016, An improved monkey algorithm for a 0-1 knapsack problem, *Applied Soft Computing*, 38, 817-830.

- Zhou, Z., Li, D., Fan, X., Lin, S., Yuan, Y., Zhuang, P., Hu, H., Ge, M., Chen, S. ve Mei, X., 2022, Gold nanoclusters for optimizing the general efficacies of herbal medicines on nerve repair after spinal cord injury, *Materials & Design*, 215, 110465.
- Zou, D., Gao, L., Li, S. ve Wu, J., 2011, Solving 0–1 knapsack problem by a novel global harmony search algorithm, *Applied Soft Computing*, 11 (2), 1556-1564.

