

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

YENİ NESİL KAOTİK TABANLI KÖK GELİŞİM ALGORİTMALARI

**DOKTORA TEZİ
Fahrettin Burak DEMİR**

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Adnan Fatih KOCAMAZ

AĞUSTOS 2020

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

YENİ NESİL KAOTİK TABANLI KÖK GELİŞİM ALGORİTMALARI

DOKTORA TEZİ

**Fahrettin Burak DEMİR
23615190351**

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Adnan Fatih KOCAMAZ

AĞUSTOS 2020

TEŞEKKÜR

Doktora eğitimimin ve hayatımın her anında bana destek olan akademik bilgi ve tecrübelerinin yanı sıra hayat tecrübelerini ve bilgi birikimini sabır ve cömertlikle benimle paylaşan kıymetli danışman hocam Doç.Dr. Adnan Fatih KOCAMAZ'a,

Akademik eğitimim boyunca bana destek olan ve tez yazımım süresince bilgi ve birikimlerini benden esirgemeyen kıymetli hocalarım Doç. Dr. Türker TUNCER, Emeritus Prof. Dr. Bülent SANKUR, Prof. Dr. Ali KARCI, Prof. Dr. Bilal ALATAŞ, Doç. Dr. Şengül DOĞAN, Dr. Öğr. Üyesi Fatih ERTAM, Doç. Dr. Ömerul Faruk DURSUN ve İbrahim Levent BELENLİ'ye,

İnönü Üniversitesi Bilgisayar Mühendisliği Bölümü akademisyenleri kıymetli hocalarıma ve idari personellerine,

Tez yazımım boyunca sürekli olarak beni motive eden, bana destek olan arkadaşlarım Dr. Muammer TÜRKÖĞLU, Dr. Emrah DÖNMEZ, Fatih OKUMUŞ ve Yahya ALTUNTAŞ'a,

Bugünlere gelmem de büyük emekleri olan, hayatımın her anında bana destek olan, haklarını hiçbir zaman ödeyemeyeceğim kıymetli annem ve babama,

Doktora eğitimim boyunca bütün sıkıntıları, yoğunluğumu ve stresimi anlayışla karşılayan ve sürekli yanımda olan sevgili Eşim Nesrin DEMİR'e ve varlığı ile hayatıma anlam ve neşe katan oğlum Alp Bilge DEMİR'e,

Şükranlarımı sunuyor, teşekkür ediyorum.

ONUR SÖZÜ

Doktora Tezi olarak sunduğum “Yeni Nesil Kaotik Tabanlı Kök Gelişim Algoritmaları” başlıklı bu çalışmanın bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurmaksızın tarafımdan yazıldığını ve yararlandığım bütün kaynakların, hem metin içinde hem de kaynakçada yöntemine uygun biçimde gösterilenlerden oluştuğunu belirtir, bunu onurumla doğrularım.

Fahrettin Burak DEMİR



İÇİNDEKİLER

TEŞEKKÜR	i
ONUR SÖZÜ	ii
İÇİNDEKİLER	iii
ÇİZELGELER DİZİNİ	v
ŞEKİLLER DİZİNİ	vi
SEMBOLLER VE KISALTMALAR	vii
ÖZET	xi
ABSTRACT	xii
1. GİRİŞ	1
1.1. Literatür İncelemesi	3
1.1.1. Optimizasyon algoritmaları ile ilgili yapılan çalışmalar	3
1.1.2. Kök gelişimi tabanlı optimizasyon algoritmaları ile ilgili yapılan çalışmalar	8
1.1.3. Kaotik haritalar ile ilgili yapılan çalışmalar	9
1.2. Önerilen Sistemin Genel Çerçevesi	11
1.2.1. Tezin amacı	11
1.2.2. Tezin gerekçeleri	12
1.2.3. Tezin çıktıları	13
1.2.4. Tezin yenilikçi yönü ve ar-ge niteliği	13
1.2.5. Tezin organizasyonu	14
2. META-SEZGİSEL ALGORİTMALAR	15
2.1. Fizik Tabanlı Algoritmalar	16
2.2. Kimya Tabanlı Algoritmalar	18
2.3. Biyoloji Tabanlı Algoritmalar	19
2.4. Matematik Tabanlı Algoritmalar	20
2.5. Sosyal Tabanlı Algoritmalar	21
2.6. Müzik Tabanlı Algoritmalar	22
2.7. Spor Tabanlı Algoritmalar	22
2.8. Sürü Zekâsı Tabanlı Algoritmalar	23
3. KAOTİK HARİTALAR	26
3.1. Tez Çalışmasında Hibrit Modellerde Kullanılan Tek Boyutlu Kaotik Haritalar	31
3.1.1. Lojistik Harita	31
3.1.2. Gauss Haritası	31
3.1.3. Singer Haritası	31
3.1.4. Sinüs Haritası	32
3.1.5. Tent (Çadır) Haritası	32
4. ÖNERİLEN YÖNTEMLER	33
4.1. Lojistik-Gauss Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Yöntemi	33
4.1.1. Önerilen Lojistik-Gauss Hibrit Kaotik Harita Tabanlı Sürü Optimizasyon Yöntemi	34
4.2. Lojistik-Singer Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Yöntemi	34
4.2.1. Önerilen Lojistik-Singer Hibrit Kaotik Harita Tabanlı Sürü Optimizasyon Yöntemi	35
4.3. Kaotik Darcy Optimizasyon Yöntemi	37
4.3.1. Lojistik-Sinüs Kaotik Haritası	37

4.3.2. Darcy Teoremi	38
4.3.3. Önerilen Kaotik Darcy Optimizasyon Yöntemi	39
4.4. Kaotik Dalton Optimizasyon Yöntemi	40
4.4.1. Kullanılan Kaotik Haritalar.....	40
4.4.2. Dalton Yasası	41
4.4.3. Önerilen Kaotik Dalton Optimizasyon Yöntemi	42
4.5. Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi	42
4.5.1. Önerilen Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi... ..	42
5. ÖNERİLEN YÖNTEMLERİN TEST FONKSİYONLARINA UYGULANMASI VE BAŞARI ANALİZİ	45
5.1. Lojistik-Gauss ve Lojistik-Singer Kaotik Optimizasyon Algoritmalarına Ait Deneysel Sonuçlar.....	47
5.2. Kaotik Darcy ve Kaotik Dalton Optimizasyon Algoritmaları'na Ait Deneysel Sonuçlar	48
5.3. Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi'ne Ait Deneysel Sonuçlar.....	51
5.3.1. LS-KOA'nın sıkıştırılmış yay dizayn problemine uygulanması ve başarı analizi.....	55
6. KAOTİK DARCY OPTİMİZASYON ALGORİTMASININ HEPATOSELÜLER KARSİNOM ARAŞTIRMASINDA (HKA) ÖZELLİK SEÇİCİ OLARAK KULLANILMASI VE ANALİZİ.....	58
6.1. Eksik Özellik Tamamlama Yöntemi.....	60
6.2. Önerilen Kaotik Darcy Optimizasyon Algoritması Tabanlı Özellik Seçici Metodu	61
6.3. Sınıflandırıcı.....	63
6.4. Sonuçlar	64
6.4.1. Veri kümesi	64
7. SONUÇ VE ÖNERİLER.....	69
7.1. Sonuçların Değerlendirilmesi	69
7.2. Öneriler	72
8. KAYNAKLAR	73
ÖZGEÇMİŞ	94

ÇİZELGELER DİZİNİ

Çizelge 3.1. Zaman Alanlarına Göre Kaotik Haritalar (G:Gerçek, AK:Akılcı, K:Karmaşık, AY:Ayrık)	26
Çizelge 4.1. Dalton Optimizasyon Algoritmasında Kullanılan Kaotik Haritalar	41
Çizelge 5.1. Çalışmalarda kullanılan kıyaslama fonksiyonları	45
Çizelge 5.2. Kaotik Haritalı Optimizasyon Algoritmalarıyla karşılaştırma sonuçları...	47
Çizelge 5.3. Kaotik Darcy Optimizasyon Algoritması'nın kıyaslama fonksiyonları üzerinde çalıştırılması sonucu elde edilen sonuçlar	48
Çizelge 5.4. Kaotik Dalton Optimizasyon Algoritması'nın kıyaslama fonksiyonları üzerinde çalıştırılması sonucu elde edilen sonuçlar	49
Çizelge 5.5. Kaotik Darcy ve Dalton Optimizasyon Algoritmaları'nın literatürde sıklıkla kullanılan optimizasyon yöntemleriyle karşılaştırılması	50
Çizelge 5.6. Önerilen optimizasyon yönteminin kıyaslama fonksiyonları üzerinde elde ettiği sonuçlar	52
Çizelge 5.7. Algoritmaların performans karşılaştırmaları	53
Çizelge 5.8. Önerilen LS-KOA'nın Sıkıştırılmış Yay Dizayn Problemi üzerinde başarımlı analizi	57
Çizelge 6.1. Kullanılan dokuz makine öğrenme algoritmasının açıklaması	63
Çizelge 6.2. Önerilen KDar-OA ve diğer yaygın olarak kullanılan Optimizasyon Algoritmalarının performansı.	64
Çizelge 6.3. Kullanılan dokuz sınıflandırıcının sonuçları (%)	66
Çizelge 6.4. Önerilen KDar-OA yönteminin ve diğer HCC sınıflandırma yöntemlerinin doğrulukları ve F1 Skoru puanları (%).	67

ŞEKİLLER DİZİNİ

Şekil 1.1.	Sezgisel Yöntemler.....	2
Şekil 4.1.	Lojistik-Gauss Tabanlı Kaotik Sürü Optimizasyonuna ait sözde kod.....	34
Şekil 4.2.	Lojistik-Singer Tabanlı Kaotik Sürü Optimizasyonuna ait sözde kod.....	37
Şekil 4.3.	Çatallanma Grafiği (a) Lojistik Harita, (b) Sinüs Harita, (c) Lojistik-Sinüs Harita	38
Şekil 4.4.	Kaotik Darcy Optimizasyon Algoritması'na ait sözde kod.....	40
Şekil 4.5.	Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Algoritmasına ait akış diyagramı	43
Şekil 5.1.	Kıyaslama fonksiyonlarına ait minimizasyon eğrileri.....	54
Şekil 5.2.	Sıkıştırılmış Yay Dizayn Problemi'nin grafiksel gösterimi	56
Şekil 6.1.	Önerilen Kaotik Darcy Optimizasyon Algoritması temelli HKA sınıflandırma yönteminin şematik açıklaması	60
Şekil 6.2.	Önerilen KDar-OA tabanlı özellik seçicinin özeti.	61
Şekil 6.3.	Parçacığın ikili değerinin kullanarak özellik seçme prosedürü	62
Şekil 6.4.	KDar-OA tabanlı yönteme ait karışıklık matrisi ve SNN sonuçları.....	68

SEMBOLLER VE KISALTMALAR

ÇOA	: Çekirge Optimizasyon Algoritması
MOA	: Martı Optimizasyon Algoritması
BOA	: Balina Optimizasyon Algoritması
FSOA	: Fil Sürüsü Optimizasyon Algoritması
GFSOA	: Geliştirilmiş Fil Sürüsü Optimizasyon Algoritması
MSOA	: Mayıs Sineği Optimizasyon Algoritması
KOA	: Kelebek Optimizasyon Algoritması
HÇ-PSO	: Hibrit Çok Amaçlı Parçacık Optimizasyon Algoritması
G-AOA	: Güve-Alev Optimizasyonu Algoritması
GG-AOA	: Geliştirilmiş Güve-Alev Optimizasyon Algoritması
TSO	: Tavuk Sürüsü Optimizasyonu
ÖÖTO	: Öğretme-Öğrenme Tabanlı Optimizasyon
GTS-ÖÖTO	: Geliştirilmiş Tavuk Sürüsü – Öğretme-Öğrenme Tabanlı Optimizasyon
HŞO	: Harris Şahinleri Optimizasyonu
DA	: Denge Ayarlayıcı
PSO	: Parçacık Sürü Optimizasyonu
BFO	: Beyin Fırtınası Optimizasyonu
KE-BFO	: Küresel En-İyi Beyin Fırtınası Optimizasyonu
KDPSTO	: Kendini Düzenleyen Parçacık Sürü Optimizasyonu
YKPSO	: Yalın Kemik Parçacık Sürü Optimizasyonu
KÖPSO	: Kapsamlı Öğrenme Parçacık Sürü Optimizasyonu
KAOA	: Karınca Aslanı Optimizasyon Algoritması
SKA	: Sinüs-Kosinüs Algoritması
GA	: Genetik Algoritma
SÖO	: Sosyal Örümcek Optimizasyonu
YAKA	: Yapay Arı Kolonisi Algoritması
GKAA	: Guguk Kuşu Arama Algoritması
GDT-GKAA	: Gauss Dağılımı Tabanlı Guguk Kuşu Arama Algoritması
İOO	: İstilacı Ot Optimizasyonu
AFO	: Ağaç Fizyolojisi Optimizasyonu
BBSA	: Bitki Büyüme Simülasyon Algoritması
KSAYA	: Kablosuz Sensör Ağı Yerelleştirme Algoritması
BTA	: Benzetilmiş Tavlama Algoritması
ÇLKO	: Çim Lifli Kök Optimizasyon Algoritması

KSBA	: Kök Sistemi Büyüme Algoritması
ABA	: Ağaç Büyüme Algoritması
CEO	: Çoklu Evren Optimizasyonu
KCEO	: Kaotik Çoklu Evren Optimizasyonu
KKOA	: Kaotik Kelebek Optimizasyon Algoritması
YSO	: Yunus Sürüsü Optimizasyonu
KYSO	: Kaotik Yunus Sürüsü Optimizasyonu
EAOA	: Elektromanyetik Alan Optimizasyonu Algoritması
EMOA	: Elektromanyetizma Optimizasyon Algoritması
YAA	: Yerçekimi Arama Algoritması
SÇA	: Su Çevrimi Algoritması
OEOA	: Optikten Esinlenilen Optimizasyon Algoritması
BPBBOA	: Büyük Patlama-Büyük Büzülme Optimizasyon Algoritması
YFOA	: Yapay Fizik Optimizasyon Algoritması
MKOA	: Merkezi Kuvvet Optimizasyon Algoritması
YKRA	: Yapay Kimyasal Reaksiyon Algoritması
YPAA	: Yapay Atom Algoritması
GA	: Genetik Algoritma
DGA	: Diferansiyel Gelişim Algoritması
BCTOA	: Biyo-Coğrafya Temelli Optimizasyon Algoritması
YBOA	: Yapay Bağışıklık Optimizasyon Algoritması
KASA	: Kaotik Altın Sinüs Algoritması
EYOA	: Emperyalist Yarışmacı Optimizasyon Algoritması
POA	: Parlemler Optimizasyon Algoritması
AAA	: Armoni Arama Algoritması
MAA	: Melodi Arama Algoritması
LŞA	: Lig Şampiyonası Algoritması
CBO	: Cırcır Böceği Optimizasyonu
KKA	: Karınca Kolonisi Algoritması
LG-KOA	: Lojistik-Gauss Kaotik Optimizasyon Algoritması
LS-KOA	: Lojistik-Singer Kaotik Optimizasyon Algoritması
KDar-OA	: Kaotik Darcy Optimizasyon Algoritması
KDal-OA	: Kaotik Dalton Optimizasyon Algoritması
HKA	: Hepatoselüler Karsinom Araştırması
LSinüs-KOA	: Lojistik-Sinüs Kaotik Optimizasyon Algoritması

DT	: Karar Ağacı (Decision Tree)
LD	: Doğrusal Diskriminant (Linear Discriminant)
QD	: Karesel Diskriminant (Quadratic Discriminant)
LR	: Mantıksal Regresyon (Logistic Regression)
SVM	: Destek Vektör Makinesi (Support Vector Machine)
KNN	: En Yakın Komşuluk (K-Nearest Neighbors)
BT	: Torbalı Ağaç (Bagged Tree)
SD	: Alt Uzay Ayırıcı (Subspace Discriminant)
SNN	: Alt Uzay En Yakın Komşuluğu (Subspace Nearest Neighbor)
HKA	: Hepatoselüler Karsinom Araştırması
HCC	: Hepatocellular Carcinoma
Sin	: Sinüs
TP	: (True Positive) Doğru Pozitif
FN	: (False Negative) Yanlış Negatif
FP	: (False Pozitif) Yanlış Pozitif
TN	: (True Negatif) Doğru Negatif
US	: Üst Sınır
AS	: Alt Sınır
Π	: Pi değeri
μ	: Ortalama
σ^2	: Varyans
x_n	: Kaotik değişken
mod	: Tepe değer
<i>pn</i>	: Parçacık sayısı
rand	: Rastgele sayı
T	: Tohum vektörü
<i>iter</i>	: İterasyon
<i>p_{best}</i>	: En iyi parçacık değeri
step	: Adım
Q	: Hacimsel akış
K	: Hidrolik iletkenlik
<i>i</i>	: Hidrolik eğim
A	: Kesit alanı
L	: Mesafe
ln	: Doğal logaritma

p_n	: Parçacık sayısı
h	: Suyun yükseklik değeri
min	: Minimum değer
E	: Buharlaşma miktarı
c	: Rüzgar katsayısı
e_w	: Suyun buhar basıncı
e_a	: Havanın buhar basıncı
x_1, w	: Telin çapı
x_2, d	: Asıl yay çapı
x_3, L	: Yay uzunluğu



ÖZET

Doktora Tezi

YENİ NESİL KAOTİK TABANLI KÖK GELİŞİM ALGORİTMALARI

Fahrettin Burak DEMİR

İnönü Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

95+xii sayfa

2020

Danışman: Doç. Dr. Adnan Fatih KOCAMAZ

Bilim insanları için problemlerin çözümünde en iyi yöntemin aranması, son yıllarda dikkat çeken bir konu haline gelmiştir. Klasik yöntemlerle çözülemeyen matematiksel problemlerin çözümü için birçok meta-sezgisel yöntem önerilmiştir. Meta-sezgisel yaklaşımlar, en iyi çözümü bulma noktasında garanti vermezler. Fakat en iyiye yakın sonuçlar vermeye çalışırlar. Meta-sezgisel yaklaşımların problemlere uyarlanabilir olması, zamanla önemini daha da artırmıştır. Fizik, kimya, biyoloji, matematik, sosyal, müzik, spor, sürü zekâsı tabanlı ve melez olmak üzere 9 farklı kategoride değerlendirilen meta-sezgisel algoritmalar içerisinde özellikle sürü zekâsı tabanlı algoritmalar, problemlerin çözümündeki başarıları sebebiyle daha popüler hale gelmiştir. Sürü zekâsı tabanlı algoritmalar, hayvan ve bitki davranışlarını inceleyerek en iyileme yöntemleri geliştirmeyi amaçlar. Bitkiler hakkında yapılan en son çalışmalar, bitkilerin de en az hayvanlar kadar akıllı davranışlar ortaya koyduğunu göstermiştir. Bitkilerin akıllı yaklaşımları konusunda ise en popüler çalışma alanlarından birisi de Kök Gelişim Algoritmaları olmuştur. Diğer taraftan, günlük hayattaki matematiksel problemlerin zorlaşması, önerilen optimizasyon yöntemlerinin de geliştirilmesi gerekliliğini ön plana çıkarmış ve bu amaçla farklı sayı üreteçleri kullanılmaya başlanmıştır. Sayı üreteçleri alanında en çok kullanılan yöntemlerden birisi de iyi istatistiksel özellikler sunabilen kaotik haritalar olmuştur.

Bu tez çalışmasında, literatürde yer alan meta-sezgisel optimizasyon yöntemleri, kök gelişim algoritmaları ve kaotik haritalar incelenmiştir. Optimizasyon algoritmalarının performansını ve başarımını artırmak için yeni nesil hibrit kaotik haritalar önerilmiş ve bu haritalar ile yeni nesil hibrit kaotik harita tabanlı kök gelişim algoritmaları geliştirilmiştir. Ayrıca bu yöntemlerin başarımını test etmek amacıyla, önerilen algoritmalar, literatürde sıklıkla kullanılan kıyaslama fonksiyonları ile karşılaştırılmıştır. Ardından mühendislik problemleri üzerinde ve özellik seçici olarak veri setinde performans ölçümü yapılmış ve sonuçlar karşılaştırılarak sunulmuştur.

Anahtar Kelimeler: Optimizasyon, Kök Gelişim Algoritmaları, Meta-Sezgisel Algoritma, Özellik Seçici, Darcy Teoremi, Dalton Yasası.

ABSTRACT

Ph.D.Thesis

NEW GENERATION CHAOTIC BASED ROOT DEVELOPMENT ALGORITHMS

Fahrettin Burak DEMİR

Inonu University
Graduate School of Nature and Applied Sciences
Department of Computer Engineering

95+xii pages

2020

Supervisor: Assoc. Prof. Dr. Adnan Fatih KOCAMAZ

The search for the best method for solving problems for scientists has become a remarkable subject in recent years. Many meta-heuristic methods have been proposed for the solution of mathematical problems that cannot be solved by classical methods. Meta-intuitive approaches do not guarantee the best solution. But they try to give near best results. The fact that meta-heuristic approaches can be adapted to problems has increased its importance over time. Physics-based, chemistry-based, biology-based, math-based, social-based, music-based, sports-based, herd intelligence-based and hybrid-based meta-intuitive algorithms, which are evaluated in 9 different categories, especially herd intelligence-based algorithms are more popular due to their success in solving problems. Herd intelligence-based algorithms are aimed at developing optimization methods by examining animal and plant behavior. Recent studies on plants have shown that plants exhibit as smart behaviors as animals. Root Development Algorithms have taken the most popular field of study on the smart approaches of plants. Because Root Development Algorithms provide great benefits in modeling and solving real life problems. In addition, the difficulty of mathematical problems in daily life has brought to the fore the necessity of developing the recommended optimization methods and for this purpose, chaotic maps have been used in meta-heuristic optimization algorithms.

In this thesis, meta-heuristic optimization methods, root development algorithms and chaotic maps in the literature are examined. To increase the performance and performance of optimization algorithms, new generation hybrid chaotic maps have been proposed and new generation hybrid chaotic map based root development algorithms have been developed with these maps. Also, in order to test the performance of these methods, the proposed algorithms have been run with benchmarking functions commonly used in the literature. Then, performance measurement was made on the engineering problems and feature selector in the data set and the results were compared and presented.

Keywords: Optimization, Root Development Algorithms, Meta-Heuristic Algorithm, Feature Selector, Darcy's Theorem, Dalton's Law.

1. GİRİŞ

Optimizasyon, sınırlamaları dikkate alarak problemlerde en iyinin aranması işlemi olarak bilinmektedir. Literatürde çok fazla tanımı bulunan optimizasyonu, en basit biçimde şu şekilde açıklayabiliriz: Farklı çözümleri bulunan bir problem için en optimum çözümü seçme işlemidir [1].

Bir optimizasyon problemi matematiksel olarak incelendiğinde:

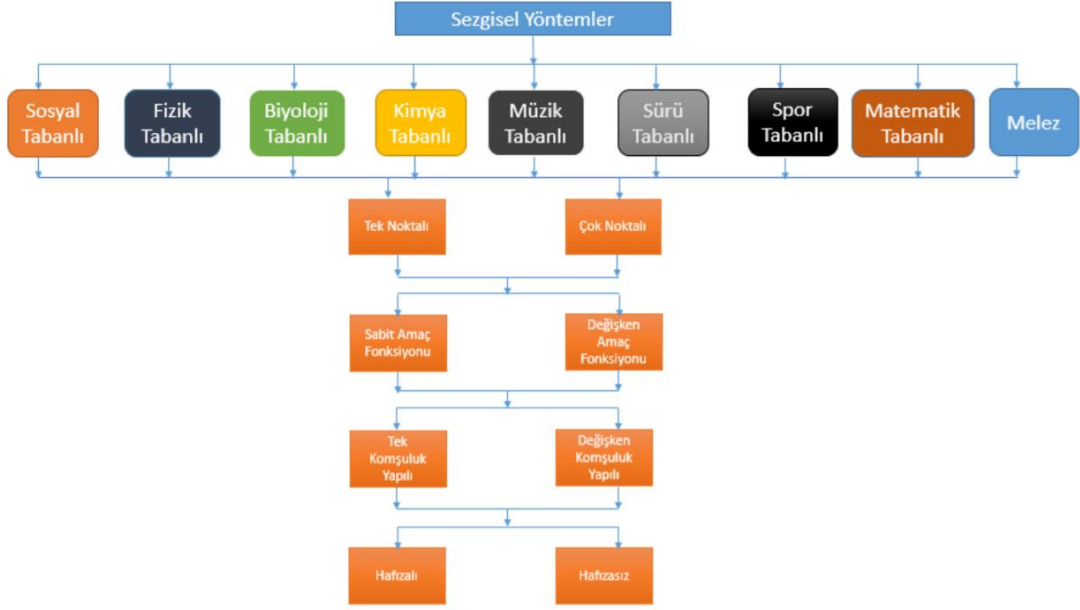
- Optimizasyon için kullanılan fonksiyon, $f(x)$,
- Kısıt fonksiyonları, $f(x) = 0, g(x) \leq 0$,
- Tasarım değişkenleri $x_l \leq x \leq x_u$ olarak karşımıza çıkmaktadır [2].

Optimizasyon işleminde kullanılan fonksiyon, uygunluk veya hedef fonksiyon olarak isimlendirilmektedir. Uygunluk fonksiyonunun çözümünde uyulması gereken fonksiyonlar, kısıt fonksiyonu olarak ifade edilir. Uygunluk fonksiyonundaki değişkenlere ise tasarım değişkenleri denilmektedir. Uygunluk fonksiyonu, problemin amacına göre minimize ve maksimize edilerek en iyileme işlemini sağlamaktadır [3].

Optimizasyon yöntemleri, hiyerarşik düzende öncelikle, modelleme ve çözümleme olarak iki bölüme ayrılmaktadır [2,4]. Modelleme, gerçek hayat problemlerinin matematiksel ifadesini, çözümleme ise bu modellemedeki en iyi çözümün elde edilmesini ifade etmektedir [4]. Optimizasyonun geçmişine bakıldığında, araştırmacıların önceliğinin modelleme olduğu görülmektedir. Bu konudaki ilk araştırmalar Amerika Birleşik Devletleri'nde ekonomi alanında gerçekleştirilmiştir [5,6]. Aradan geçen uzun yıllardan sonra araştırmacılar modellemeden çözümlemeye geçmişler ve çözümleme işleminde çok sayıda algoritma önermişlerdir. Bu algoritmalar deterministik ve stokastik algoritmalar olarak isimlendirilmektedir.

Deterministik algoritmalar, parametrelerin sabit değer aldığı fonksiyonların kullanıldığı algoritmalar [7] iken stokastik algoritmalarda parametreler rastgele değerler almaktadır.

Stokastik algoritmaların temelini oluşturduğu sezgisel algoritmalar, kesin çözüme ulaşma noktasında garanti veremezler. Fakat kesin çözüme en yakın bir çözümün bulunmasına garanti verebilirler. Bu tür algoritmalar, yakınsama özelliğini kullanarak her adımda kesin çözüme yaklaştırmaya çalışırlar. Bu yöntemler incelendiğinde, genel olarak Şekil 1.1'deki gibi sınıflandırılmaktadır [8].



Şekil 1.1. Sezgisel Yöntemler

Meta-sezgisel algoritmalar ise problemlerin çözümünde sezgisel algoritmalar içerisinde hangi birinin daha iyi bir çözüm sağlayacağına karar veren bir yapıya sahiptir. Meta-sezgisel optimizasyon algoritmaları genellikle klasik matematiksel yöntemlerle çözülmesi imkansız olan problemlerin çözümünde kullanılmıştır [9]. Meta-sezgisel yöntemler problemleri çözmek için uygunluk fonksiyonuna sürekli rastgele değerler gönderir ve en iyi değerleri de hafızada tutar [2,10]. Hafızada tutma işlemi sırasında komşu elemanların da değerlerini dikkate aldıkları için daha etkili çözümler sağlamaktadırlar.

Araştırmacılar, yeni ve etkili meta-sezgisel optimizasyon yöntemleri önermek için doğa bilimleri, biyoloji, matematik, uzay bilimleri, fizik gibi çeşitli bilim dallarını araştırmaktadır [11]. Araştırmacıların en yaygın kullandığı algoritmalarından birisi genetik algoritmadır. Genetik algoritmalar, biyoloji tabanlı bir algoritma olup, evrim teorisinden esinlenerek geliştirilmiştir. [12,13]. Diferansiyel Gelişim Algoritması, Genetik Algoritma'ya ait operatörleri kullanan parçacık tabanlı bir algoritmadır [14]. Bunun yanı sıra, hayvan davranışlarını modelleyerek küresel en iyi noktayı bulmayı amaçlayan bir çok meta-sezgisel optimizasyon algoritmaları vardır. Bu algoritmalar içerisinde Karınca Koloni Algoritması [15], Yapay Arı Kolonisi Algoritması [16], Guguk Kuşu Algoritması [17], Ateş Böceği Algoritması [18], Kriket Algoritması [2], Parçacık Sürü Algoritması [16], Yarasa Optimizasyonu Algoritması [19] ve Kedi Algoritması [20] gibi algoritmalar bulunmaktadır.

Bitki davranışlarını modelleyen algoritmalar içerisinde Çiçek Tozlaşma Algoritması [21], İnvaziv Yabancı Ot Algoritması [22], Çeltik Alan Algoritması [23], Kök Kütle Optimizasyonu [24], Yapay Bitki Algoritması [25], Fotosentetik Algoritma [26], Bitki Büyüme Algoritması [27], Fidan Büyüme Algoritması [28] ve Kök Gelişim Algoritması [29] örnek gösterilebilir. Bitki davranışlarının modellenmesine dayalı algoritmaların en kapsamlı kaynaklarından birisi Akyol ve Alataş tarafından sunulmuştur [30].

Fizik, kimya, matematik ve hatta spordan esinlenen optimizasyon teknikleri de sıklıkla kullanılır. Matematiksel programlama ve meta-sezgisel algoritmaları birleştiren matematiksel algoritmalar oluşturulmuştur [31]. Bu alandaki kapsamlı çalışmalardan biri Özbay ve Alataş tarafından yapılmıştır [32]. Matematikten ilham alan optimizasyon algoritmalarından biri olan Sinüs-Kosinüs Algoritması, çözülmesi gerçekten zor olan problemleri çözmek için kullanılmıştır [33].

Optimizasyon problemlerinde belirlenecek olan yöntem, problemin tipine göre farklılık gösterebilmektedir. Doğrusal olmayan bir problemin çözümü için küresel en iyi noktayı arayan meta-sezgisel algoritmanın tercih edilmesi daha uygundur [2].

1.1. Literatür İncelemesi

Bu bölümde, optimizasyon algoritmaları ve kaotik haritalar ile ilgili literatürde yapılan çalışmalar kapsamlı bir şekilde sunulmuştur.

1.1.1. Optimizasyon algoritmaları ile ilgili yapılan çalışmalar

Optimizasyon algoritmaları, bilgi güvenliği ve kriptolojide [34-37], sosyal ağlarda [38-42], yapay zekâ uygulamalarında [43,44], görüntü işlemede [45-48], sinyal işlemede [49-52], yapay sinir ağlarında [53-57], sınıflandırıcı ve büyük veri yöntemlerinde [58-60], nesne ve resim segmentasyonunda [61-63], lojistikte [64-66], ticari sistemlerde [67-69], makine öğrenmesinde [70-72] ve derin öğrenmede [73-75] sıklıkla kullanılmaktadır. Bu algoritmalar, her alanda esnek kullanım özelliği sayesinde kullanım alanını daha da genişletmektedir. Her gelişen teknoloji ve sektör, optimizasyon algoritmalarının çalışma sahası haline dönüşmektedir.

Saremi ve diğ. yaptıkları çalışmada [76], daha önce önerilen Çekirge Optimizasyonu Algoritması'na (ÇOA) ait detaylı bir inceleme gerçekleştirmiştir. Çalışmada, ÇOA'ya ait farklı versiyonlar ve çalışmalara ait literatür araştırması

verilmiştir. Ayrıca ÇOA'nın performansını test etmek için tek ve çok modlu kıyaslama fonksiyonları kullanılmıştır. Sonuçlar, ÇOA'nın yakınsama eğrisinin hızlandırılmasındaki başarısını göstermiştir. Ayrıca ÇOA, el duruşu tahmini alanındaki zorlu bir problem üzerinde test edilmiştir. Sonuç olarak kameradan alınan el görüntüsünün konfigürasyonunda doğru bir yöntem olduğu gözlemlenmiştir.

Panagant ve diğ. yaptıkları çalışmada [77], araç braketinin şekil optimizasyonunu çözmek için Martı Optimizasyon Algoritması (MOA) kullanılmıştır. Tasarım problemi, yapısal kütleyi en aza indirirken stres kısıtlamasında yapısal şekli de bulmaktadır. Fonksiyon değerlendirmeleri, sonlu elemanlar analizi kullanılarak yapılmaktadır ve bir Kriging modeli kullanılarak tahmin edilmektedir. Sonuçlar, MOA'nın da tıpkı Balina Optimizasyon Algoritması (BOA) ve Salp Sürüsü Optimizasyon Algoritması (SSOA) gibi iyi özelliklere sahip olduğunu göstermiştir.

Li ve diğ. yaptıkları çalışmada [78], daha önce önerilmiş olan Fil Sürüsü Optimizasyon Algoritması'nın (FSOA) geliştirilmiş versiyonunu sunmaktadır ve Geliştirilmiş Fil Sürüsü Optimizasyon Algoritması (GFSOA) olarak isimlendirilmektedir. Önerilen GFSOA yöntemi, bireylerin hızını ve pozisyonunu güncellemek için küresel bir hız ve yeni bir öğrenme stratejisi kullanılmıştır. Ayrıca, nüfusun çeşitliliğini korumak için yeni bir ayırma yöntemi sunulmuştur. Elitizm stratejisi, en uygun bireylerin gelecek nesilde korunmasını sağlamak için de kullanılmaktadır. GFSOA'nın performansını analiz etmek için 30 farklı kıyaslama fonksiyonu kullanılmıştır. Sonuçlar sekiz farklı optimizasyon algoritması ile karşılaştırılmış ve GFSOA'nın diğer algoritmalarından daha iyi sonuçlar verdiğini ortaya koymuştur.

Zervoudakis ve Tsafarakis yaptığı çalışmada [79], mayıs sineklerinin uçuş davranışları ve çiftleşme sürecinden etkilenerek yeni bir sürü tabanlı optimizasyon algoritması önermiştir. Mayıs Sineği Optimizasyon Algoritması'nın (MSOA) ismi verilen bu optimizasyon yönteminin performansını ölçmek için 38 farklı kıyaslama fonksiyonu kullanılarak, yedi farklı optimizasyon algoritması üzerinde çalıştırılmıştır. Karşılaştırma sonuçları, MSOA'nın yakınsama hızı konusunda diğer tüm algoritmalarından daha iyi sonuçlar verdiğini göstermiştir.

Yıldız ve diğ. yaptığı çalışmada [80], Kelebek Optimizasyon Algoritması'nı (KOA) kullanarak cıvatalı jant bağlantı problemi ve süspansiyon kolu şekil

optimizasyonu problemi çözülmeye çalışılmıştır. KOA, süspansiyon kolu şekil optimizasyonu probleminde %32,9 oranında ağırlık azalması sağlamıştır.

Pham ve diğ. yaptığı çalışmada [81], Balina Optimizasyon Algoritması'nı (BOA) kullanarak kablosuz ağlarda kaynak tahsisi sorunlarını çözmeye çalışmışlardır. Bu amaçla, kablosuz parazit ağlarda enerji verimlilik dengelenmesi, güvenli iş hacmini yükseltecek güç tahsisi ve mobil uç hesaplama işlemleri için BOA kullanılmıştır.

Cui ve diğ. yaptığı çalışmada [25], kömür üretiminde güvenli ve verimli çalışabilmek için çok amaçlı bir optimizasyon algoritması önermiştir. Bu çalışmada, enerji, ekoloji, ekonomi gibi faktörleri dikkate alarak Hibrit Çok Amaçlı Parçacık Sürü Optimizasyonu (HÇ-PSO) geliştirilmiştir. HÇ-PSO performans ölçümü için iki farklı kıyaslama fonksiyonu ve beş farklı optimizasyon algoritması ile çalıştırılmıştır. Kömür üretiminde bölgesel planlamada başarılı sonuçlar vermiştir.

Pelusi ve diğ. yaptığı çalışmada [82], enine yönlendirme adı verilen bir arama algoritması olan Güve-Alev Optimizasyonu Algoritması (G-AOA)'na ait küresel arama kabiliyeti ve yakınsama hızının bozulmasından kaynaklanan kusurların üstünden gelmesi için Geliştirilmiş Güve-Ateş Optimizasyonu Algoritması (GG-AOA) önerilmiştir. GG-AOA, kıyaslama fonksiyonları ve altı farklı tasarım problemi üzerinde çalıştırılmış ve yakınsama performansı konusunda karşılaştırma fonksiyonları içerisinde en iyi sonucu vermiştir.

Deb ve diğ. yaptığı çalışmada [83], yerel optimum noktalarda tuzağa düşen Tavuk Sürüsü Optimizasyonu (TSO)'nun horoz parametresinin güncellenmiş bir versiyonunu önermişlerdir. Ayrıca Geliştirilmiş Tavuk Sürüsü Optimizasyonu ile Öğretme-Öğrenme Tabanlı Optimizasyonu'nu (ÖÖTO) hibrit modelle kullanarak Geliştirilmiş Tavuk Sürüsü ve Öğretme-Öğrenme Tabanlı Optimizasyonu'nu (GTS-ÖÖTO) önermiştir. GTS-ÖÖTO, hem TSO'nun hem de ÖÖTO'nun güçlü yönlerine sahiptir. Önerilen yeni algoritma sekiz temel, on beş hesaplanması güç kıyaslama fonksiyonuyla ve iki gerçek dünya problemiyle performans testine tabi tutulmuştur. Önerilen GTS-ÖÖTO'nun diğer algoritmaların çoğundan daha iyi sonuçlar verdiği görülmüştür.

Heidari ve diğ. yaptığı çalışmada [84], Harris Şahinleri Optimizasyonu (HŞO) ismi verilen popülasyon tabanlı ve doğadan ilham alan bir optimizasyon algoritması önermiştir. HŞO'nun ilham kaynağı, Harris şahinlerinin doğuştan gelen sürpriz atak

denilen işbirlikçi davranışı ve kovalama tarzıdır. Bu akıllı stratejide, birkaç şahin, sürpriz yapmak için işbirliği yaparak farklı yönlerden av gerçekleştirmektedir. Önerilen HŞÖ'nün performans testinin yapılabilmesi için literatürde sıklıkla kullanılan 29 farklı kıyaslama fonksiyonu ve 21 optimizasyon algoritması ile çalıştırılmıştır. Sonuç olarak HŞÖ'nün pek çok algoritmadan daha iyi veya rekabetçi sonuçlar verdiği gözlemlenmiştir.

Arora ve Singh yaptığı çalışmada [85], kelebeklerin gıda arama ve çiftleşme davranışlarını modelleyen Kelebek Optimizasyonu Algoritması'nı (KOA) sunmuşlardır. KOA, temelde nektar veya çiftleşme partnerinin yerini belirlemek için koku duyularını kullanan kelebeklerin davranış stratejilerine dayanmaktadır. KOA, performans ölçümü için 30 farklı kıyaslama fonksiyonu, 9 farklı optimizasyon algoritması ve 3 farklı mühendislik problemi üzerinde çalıştırılmıştır. Alınan sonuçlar, KO'nun diğer algoritmalarından daha başarılı sonuçlar verdiğini göstermiştir.

Faramarzi ve diğ. yaptığı çalışmada [86], hem dinamik hem de denge durumlarını tahmin etmek için kullanılan kütle dengesi modelinden esinlenerek Denge Ayarlayıcı (DA) ismi verilen bir optimizasyon algoritması önermiştir. DA'da konuma sahip her bir parçacık bir arama ajanı olarak kabul edilir. Önerilen DA, 58 farklı kıyaslama fonksiyonu ve 3 farklı mühendislik problemiyle karşılaştırılmıştır. DA'ya ait sonuçlar, ayrıca Parçacık Sürü Optimizasyonu (PSO), Gri Kurt Algoritması (GKA), Genetik Algoritma (GA), Yerçekimi Arama Algoritması (YAA), ve Salp Sürüsü Algoritması (SSA) ile karşılaştırılmıştır. DA, PSO'dan %60, GKA'dan %69, GA'dan %94, YAA'dan %96 ve SAA'dan %77 daha iyi performans göstermiştir.

Wang ve diğ. yaptığı çalışmada [87], pek çok problemin çözümünde etkili bir yöntem olan Parçacık Sürü Optimizasyonu'nun (PSO), erken yakınsama ve çok modlu problemlerde zayıf performans göstermesi gibi sorunlarını ortadan kaldırabilecek Ayarlanabilir Öğrenme Stratejisi kullanan hibrit bir PSO algoritması önermiştir. Önerilen bu yöntem, keşif yeteneğini sağlamak için kendi kendine öğrenmeye dayalı bir aday oluşturma stratejisi ve rekabetçi bir öğrenme tabanlı tahmin stratejisi kullanmaktadır. Önerilen PSO yönteminin başarımı için 40 farklı kıyaslama fonksiyonu kullanılmış ve sonuçlar 5 farklı PSO ile karşılaştırılmıştır. Sonuçlar Ayarlanabilir Öğrenme Stratejisi kullanan hibrit PSO'nun yakınsama doğruluğu ve yakınsama hızında diğer PSO'lardan daha iyi sonuç verdiğini göstermiştir.

El-Abd yaptığı çalışmada [88], Beyin Fırtınası Optimizasyonu'nuna (BFO) ait değişkenleri değişken başına güncelleme metodu kullanarak Küresel En-İyi Beyin Fırtınası Optimizasyonu (KE-BFO) olarak literatüre kazandırmıştır. Ek olarak, önerilen algoritma popülasyonun mevcut durumu tarafından tetiklenen yeniden başlatma şeması içermektedir. Önerilen KE-BFO, çok çeşitli BFO varyasyonlarıyla karşılaştırılmıştır. Karşılaştırmalar nihai çözümlere ve yakınsama özelliklerine dayanmaktadır. Sonuçlar, önerilen KE-BFO'nun çok çeşitli klasik fonksiyonlarda ve farklı problem boyutlarında önceki BFO varyasyonlarından daha iyi performans gösterdiğini göstermiştir.

Tanweer ve diğ. yaptığı çalışmada [89], Kendini Düzenleyen Parçacık Sürü Optimizasyonu (KDPSO) ismi verilen yeni bir algoritma önermiştir. KDPSO'nun ilham kaynağı, bilişsel psikolojideki öğrenme stratejileridir. KDPSO, bilişsel psikolojideki bu stratejilerden yola çıkarak PSO için küresel algılama yönünün kendi kendini algılamasını, bu sayede de çözüm alanının diğer parçacıklar tarafından kullanılmasını sağlamıştır. KDPSO 25 farklı kıyaslama fonksiyonu üzerinde çalıştırılmış, ayrıca sonuçlar Yalın Kemik Parçacık Sürü Optimizasyonu (YKPSO) ve Kapsamlı Öğrenme Parçacık Sürü Optimizasyonu (KÖPSO) ile karşılaştırılmıştır. KDPSO, %95 güven seviyesi ile diğer optimizasyon algoritmalarından daha iyi bir sonuç vermiştir.

Ali ve diğ. yaptığı çalışmada [90], çeşitli dağıtım ağlarında yenilenebilir dağıtılmış üretim kaynaklarının optimum tahsisi ve boyutlandırılması için Karınca Aslanı Optimizasyon Algoritması (KAOA) önermiştir. Önerilen KAOA, 33 ve 69 veri yolu radyal dağıtım sistemlerinde test edilmiştir. Önerilen algoritmanın toplam güç kayıplarını azaltmada ve net tasarrufu en üst düzeye çıkarmada başarılı sonuçlar verdiği gözlemlenmiştir.

Hafez ve diğ. yaptığı çalışmada [91], Sinüs-Kosinüs Algoritması (SKA) kullanarak yeni bir özellik seçici sistem önermiştir. SKA, en uygun çözümü hızlı bir şekilde bulmak için keşif ve sömürüyü uyarlamalı bir şekilde dengelemektedir. Önerilen yöntem 18 veri kümesi üzerinde test edilmiştir. Sonuç olarak önerilen yöntemin Parçacık Sürü Algoritması (PSO) ve Genetik Algoritma (GA) ile kıyaslanması sonucunda daha iyi sonuçlar verdiği gözlemlenmiştir.

Cuevas ve diğ. yaptığı çalışmada [92], sosyal örümceklerin işbirlikçi davranışlarından esinlenerek yeni bir optimizasyon algoritması önermiştir. Bu algoritma, Sosyal Örümcek Optimizasyonu (SÖO) ismi verilen ve sosyal örümceklerin

cinsiyetlerine göre gerçekleştirdikleri işbirliklerini gösteren bir yöntemdir. Bu algoritma, evrimsel operatörler içerdiği için Yapay Arı Kolonisi Algoritması (YAKA) ve Parçacık Sürü Algoritması (PSO) gibi iki farklı evrimsel algoritma tabanlı algoritma ile kıyaslanmıştır. Sonuç olarak önerilen algoritmanın, diğer algoritmalarından çok daha iyi sonuçlar verdiği görülmüştür.

Zheng ve Zhou yaptığı çalışmada [93], gauss dağılımı tabanlı yeni bir Guguk Kuşu Arama Algoritması (GKAA) geliştirmiştir. Gauss Dağılımı Tabanlı Guguk Kuşu Arama Algoritması (GDT-GKAA), 6 farklı kıyaslama fonksiyonu üzerinde ve GKAA ile çalıştırılmıştır. Sonuç olarak, GDT-GKAA'nın bütün kıyaslama fonksiyonları üzerinde daha iyi yakınsama özelliği gösterdiği gözlemlenmiştir.

1.1.2. Kök gelişimi tabanlı optimizasyon algoritmaları ile ilgili yapılan çalışmalar

Literatür incelendiğinde, kök gelişim algoritmalarına ait pek çok çalışma bulunduğu görülmektedir. Kök gelişimine dayalı algoritmalar, güç sistemlerinde kapasitör yerleşiminde [94], sayısal optimizasyon yöntemlerinde [24], toprak-su potansiyel ölçümlerinde [95], kısıtlı çok amaçlı optimizasyon problemlerinin çözümünde [96] ve sayısal fonksiyon optimizasyonlarında [97] kullanılmıştır.

Kök gelişimi tabanlı algoritmalarla ait çalışmalar aşağıda özetlenmiştir.

Pourjafari yaptığı çalışmada [98], doğrusal olmayan denklem sistemlerinin çözümü için yeni bir İstilacı Ot Optimizasyonu (İOO) önermiştir. Önerilen yöntem, literatürdeki 4 farklı optimizasyon algoritması ile karşılaştırılmış ve sonuçlar İOO'dan daha iyi veya rekabetçi çözümler ortaya koyduğunu göstermiştir.

Halim ve İsmail yaptığı çalışmada [99], Ağaç Fizyolojisi Optimizasyonu'nu (AFO) kullanarak ağ tabanlı bir çıkarım sistemini eğitmek için kullanmıştır. Parçacık Sürü Optimizasyonu (PSO) ve Genetik Algoritma (GA) ile benzer özellikler gösteren bu algoritma, gelecekte benzer problemlerin çözümleri için umut verici sonuçlar üretmiştir.

Tang ve diğ. yaptığı çalışmada [100], Bitki Büyüme Simülasyon Algoritması'na (BBSA) dayanan yeni bir Kablosuz Sensör Ağı Yerelleştirme Algoritması (KSAYA) önerilmiştir. KSAYA, Benzetilmiş Tavlama Algoritması (BTA) ile performans testine tabi tutulmuş ve elde edilen sonuçlar KSAYA'nın daha iyi yerelleştirme hassasiyetine ve daha hızlı hesaplama hızına sahip olduğunu göstermiştir.

Guo ve diğ. yaptığı çalışmada [101], Çim Lifli Kök Optimizasyon Algoritması'nın (ÇLKOA) değiştirilmiş bir modeli ile polimer yakıt hücrelerinin parametre tanımlaması için yeni bir optimal metodoloji sunmuştur. Bu yeni algoritma literatürdeki 4 farklı optimizasyon algoritması ile karşılaştırılmış ve algoritmanın yüksek duyarlılık değerlerine sahip olduğu gözlemlenmiştir.

Yun ve diğ. yaptığı çalışmada [102], Kök Sistemi Büyüme Algoritması (KSBA) kullanarak mürekkep püskürtmeli yazıcıların baskı kalitesini artıran bir yöntem geliştirmiştir. Bu çalışma, mürekkep püskürtmeli yazıcılarda, püskürtmeden sonra kalan titreşimin baskı kalitesini etkilemesinin önüne geçebilecek etkili bir yöntemdir. Bu yöntem 10 farklı kıyaslama fonksiyonu ve 4 farklı optimizasyon algoritması kullanılarak performans testine tabi tutulmuş ve önerilen yöntemin başarılı sonuçlar verdiği gözlemlenmiştir.

Cheraghalipour ve diğ. yaptığı çalışmada [103], ağaçların ışık ve yiyeceğe ulaşmasında kullandığı metodolojiye dayanan Ağaç Büyüme Algoritması'nı (ABA) geliştirmiştir. ABA, 30 farklı kıyaslama fonksiyonu ve 8 farklı optimizasyon algoritması üzerinde çalıştırılmış. Ayrıca 4 farklı gerçek dünya probleminin çözümüne uyarlanmıştır. Elde edilen sonuçlar, önerilen algoritmanın en iyi veya en iyiye yakın sonuçlar verdiğini göstermiştir.

Akkar ve Mahdi yaptığı çalışmada [104], Çim Lifli Kök Optimizasyon Algoritması (ÇLKOA) önermiştir. Bu algoritmada, her çim küresel bir aday çözümünü ifade ederken, yenilenen ikincil kökler ise yerel olarak elde edilen çözümleri ifade etmektedir. ÇLKOA'nın performansını ölçmek için önerilen algoritma, literatürde sıkça kullanılan 7 farklı kıyaslama fonksiyonu ve 9 farklı optimizasyon algoritması üzerinde çalıştırılmıştır. Sonuç olarak ÇLKOA'nın hemen hemen bütün algoritmalar içerisinde en iyi sonuçları verdiğini gözlemlenmiştir.

1.1.3. Kaotik haritalar ile ilgili yapılan çalışmalar

Kaotik haritalar, literatürde en geniş kullanım alanı olan sistemlerden birisidir. Optimizasyon yöntemlerinde [84,105,106], görüntü şifrelemede [107,108], kriptoloji ve şifrelemede [109,110], gizli anahtar şifreleme üretiminde [111] ve yapay sinir ağlarında [112] sıklıkla kullanılmaktadır.

Kaotik haritalara ait çalışmalar aşağıda özetlenmiştir.

Abdullah ve Abdullah yaptığı çalışmada [113], Arnold'un Kedisi, Henon ve Lojistik Kaotik Haritaları'nı kullanarak yeni bir görüntü şifreleme algoritması önermiştir. Önerilen algoritmanın güvenlik analizinde, algoritmanın 7,9975'lik bir entropi değerine ve piksel değişim oranında %99,66'lık bir başarıma sahip olduğu görülmüştür.

Han yaptığı çalışmada [114], değiştirilmiş lojistik kaotik harita kullanarak yeni bir görüntü şifreleme algortması önermiştir. Güvenlik performans analizi için önerilen algoritma, anahtar alan boyutu, anahtar hassasiyet gücü, gri histogram homojenliği ve bitişik piksellerin korelasyonu gibi parametreler için çalıştırılmıştır. Elde edilen sonuçlar, önerilen algoritmanın saldırılara karşı etkili bir şekilde direnebileceğini göstermiştir.

Naseer ve diğ. yaptığı çalışmada [115], 3 boyutlu hibrit kaotik harita kullanarak görüntü şifreleme için yeni bir teknik önermiştir. Bu yöntem, mevcut kriptanaliz yöntemleriyle test edilmiş ve önerilen yöntemin güvenli görüntü aktarımı için güvenilir bir sistem olduğu görülmüştür.

Ewees ve diğ. yaptığı çalışmada [116], Çoklu Evren Optimizasyonu'nun (ÇEO) yerel minimum noktalara takılmasının önüne geçebilecek yeni bir Kaotik Çoklu Evren Optimizasyonu (KÇEO) önermiştir. Önerilen algoritma Sinüsoidal, Piecewise, Singer, Tent ve Lojistik Haritalarını kullanmıştır. Ayrıca KÇEO, literatürde sıkça kullanılan 4 farklı optimizasyon algoritmasıyla da karşılaştırılmıştır. Lojistik Harita Tabanlı KÇEO'nun diğer tün yöntemlerden daha iyi sonuçlar verdiği gözlemlenmiştir.

Awad ve diğ. yaptığı çalışmada [117], karmaşıklığın sürekli olarak arttığı problemlerin çözümü için yeni bir Kaotik Kelebek Optimizasyon Algoritması (KKOA) önermiştir. KKOA, performans analizi için 16 farklı kıyaslama fonksiyonu ve 6 farklı optimizasyon algoritması ile karşılaştırılmış ve sonuç olarak KKOA'nın standart Kelebek Optimizasyon Algoritması'ndan %95 daha fazla doğruluk değeri yakaladığı gözlemlenmiştir.

Wei ve diğ. yaptığı çalışmada [118], kaotik sinyali şifreleme metniyle hibritleme yerine şifre çözme işlemlerini yönetmek için gürültü benzeri bir değişken kullanarak yeni bir kaotik şifreleme yöntemi önermiştir. Önerilen bu yöntem Lojistik Harita'yı kullanmıştır. Yapılan analizler ve simülasyonlar, bu yöntemin, şifreli metin uzunluğu veya doğrudan güvenlikte iyi sonuçlar verdiğini göstermiştir.

Mishra ve Saharan yaptığı çalışmada [119], dijital görüntülerin güvenliğini sağlamak için kullanılacak yeni bir kaotik tabanlı şifreleme yöntemi önermiştir. Bu

yöntem, şifrelemeyi sağlamak için Henon Kaotik Haritası'nı ve 128 bitlik bir gizli anahtar kullanmaktadır. Koatik haritaların başlangıç parametleri üzerindeki hassasiyetini dikkate alınarak gerçekleştirilen bu yöntem, sinyal-gürültü oranı, entropi, histogram gibi performans metrikleri üzerinde test edilmiştir. Önerilen algoritmanın istatistiksel saldırılar karşısında güvenli sonuçlar verdiği ortaya konulmuştur.

Sheela ve diğ. yaptığı çalışmada [120], geliştirilmiş Henon ve Sinüs Haritaları kullanarak kaotik harita tabanlı yeni bir metin şifreleme algoritması geliştirmiştir. Önerilen algoritma, farklı metin verileri için çalıştırılmış ve başarılı sonuçlar elde edilmiştir. Deneysel bulgular, farklı dosya boyutlarındaki metin verileri için güvenli iletişim sağladığını ortaya koymuştur.

Muhammad ve Özkaynak yaptığı çalışmada [121], kaotik ve kriptografik sistemlerin iyi yönlerini birleştiren bir mimariye sahip yeni bir görüntü şifreleme algoritması geliştirmiştir. Bu algoritma, dijital görüntü güvenliği ile ilgili birçok uygulamada başarılı bir biçimde kullanılabileceğini göstermiştir.

Qiao ve Yang yaptığı çalışmada [122], Yunus Sürüsü Algoritması'nın (YSA) yerel optimum noktalara düşme eğilimini ortadan kaldıracak yeni bir Kaotik Yunus Sürüsü Algoritması (KYSA) önermiştir. KYSA, 8 farklı kıyaslama fonksiyonu üzerinde çalıştırılmıştır. Elde edilen sonuçlar, Kent Kaotik Haritası tabanlı KYSA'nın en optimum sonuçları sağladığını göstermiştir.

1.2. Önerilen Sistemin Genel Çerçevesi

Bu bölümde, tez çalışması içerisinde ortaya koyulan hedefler, gerekçeler ve çıktılar alt başlıklar şeklinde sunulmuştur. Ayrıca optimizasyon yöntemlerinin ve kaotik haritalar için geliştirilen yöntemin amaçları, Ar-Ge özelliği ve teknolojik yönü ile tez çalışmasının organizasyonu ile ilgili bilgiler de aşağıda verilmiştir.

1.2.1. Tezin amacı

Bu tez çalışmasının temel amacı, optimizasyon problemlerinin çözümünde kullanılan meta-sezgisel algoritmaları ve kaotik haritaları inceleyerek, doğadan esinlenerek oluşturulan algoritmalarından birisi olabilecek yeni nesil kaotik tabanlı kök gelişim optimizasyon algoritmalarını tanıtmaktır.

Tez çalışmasının bir diğer amacı ise, geliştirilen algoritmanın, gerçek dünya problemleri üzerindeki yetkinliğinin test edilmesi amacıyla özellik seçici olarak

kullanılmasıdır. Bu sayede, büyük veriler üzerinde, daha doğru analizler yapılabilmesine imkan sağlanacaktır.

Literatürde yer alan meta-sezgisel optimizasyon algoritmalarının geliştirilmesinde kaotik haritaların kullanılmasının son zamanlardaki başarısından ötürü yeni nesil hibrit kaotik haritaları da önerilmiştir. Bu sayede, halihazırda var olan optimizasyon tekniklerinin de iyileştirmesi amaçlanmıştır.

1.2.2. Tezin gerekçeleri

Tez kapsamında yeni nesil hibrit kaotik harita tabanlı optimizasyon yöntemleri önerilmiştir. Gerçek dünya problemlerinin gün geçtikçe daha karmaşık hale gelmesiyle beraber optimizasyon yöntemlerinin geliştirilmesi de büyük önem kazanmıştır. Bu amaçla tezin gerekçeleri aşağıdaki gibi verilebilir:

- Özellikle teknolojinin hızlı gelişimi, ekonomi, sanayi, lojistik, havacılık, sağlık, kriptografi v.b. pek çok alanda verilerin büyümesine ve bu alanlardaki problemlerinin çözümünün zorlaşmasına sebep olmuştur.
- Klasik matematiksel yöntemler, gerçek dünya problemlerinin çözümünde yetersiz kalmaktadır.
- Büyük veriler üzerinde çalışırken, günümüzdeki bilgisayarlar, veri işlemede yetersiz kalmaktadır.
- İyi tasarlanmış meta-sezgisel yöntemler, zaman ve iş maliyetini düşürmekte ve ekonomik kazancı artırmaktadır.
- Var olan meta-sezgisel yöntemler de problemlerin gelişmesiyle beraber yetersiz kalmaktadır.
- Kaotik haritaların optimizasyon yöntemlerinde kullanılması gün geçtikçe önem kazanmaktadır.
- Yeni nesil optimizasyon algoritmalarının sadece bilgisayar veya kağıt üzerinde değil, gerçek dünya problemlerinde kullanılması büyük önem arz etmektedir.

Yukarıda belirtilen problemlerin çözümündeki gereksinimler genel olarak şunlardır:

- Meta-sezgisel algoritmaların güncel ve gerçek problemler üzerinde denenmesi,
- Var olan meta-sezgisel algoritmaların kaotik haritalar ve benzeri yöntemler kullanılarak geliştirilmesi,
- Meta-sezgisel algoritmaların problemlere uyarlanabilir ve esnek olması.

Tüm bu nedenlerden dolayı, bu problemlerin giderilme ihtiyacı, tez çalışmasının ortaya çıkma gerekçesini oluşturmuştur.

1.2.3. Tezin çıktıları

Bu tez çalışmasında, yeni nesil hibrit kaotik haritalar önerilmiştir. Önerilen yeni hibrit kaotik haritalar kullanılarak, yeni nesil kök gelişim algoritmaları önerilmiştir. Önerilen bu yöntemler ile gerçek hayat problemlerinin çözümünde performansın artırılması amaçlanmıştır.

Tez çalışması sonucunda elde edilen yeni nesil hibrit kaotik harita tabanlı optimizasyon algoritmaları test fonksiyonları ile performans ölçümüne tabi tutulmuştur. Ayrıca sıkıştırılmış yay dizayn problemi üzerinde test edilmiştir. Son olarak kaotik harita tabanlı darcy optimizasyon algoritması, sağlık verisi üzerinde özellik seçici olarak kullanılmıştır.

1.2.4. Tezin yenilikçi yönü ve ar-ge niteliği

- 1) Tez çalışması içerisinde geliştirilen hibrit kaotik yöntemler ve yeni nesil hibrit kaotik tabanlı kök gelişim algoritmaları, literatürdeki kıyaslama fonksiyonları ve optimizasyon problemleri kullanılarak test edilmiştir ve performansları karşılaştırılmıştır. Elde edilmiş olan performans sonuçları, önerilen yöntemlerin var olan yöntemlerden daha başarılı olduğunu göstermiştir. Gerçekleştirilen çalışmalar neticesinde, geliştirilen yöntemlerin etkili, esnek ve güvenilir olduğu görülmüştür.
- 2) Geliştirilen optimizasyon algoritması, yeni nesil hibrit kaotik harita tabanlı olarak tasarlandığı için en iyileme sürecinde daha başarılı sonuçlar vermektedir.
- 3) Geliştirilen hibrit kaotik harita tabanlı optimizasyon algoritması, gerçek dünya problemleri üzerinde test edilmiştir ve başarılı sonuçlar alınmıştır. Ayrıca

geliştirilen yöntemin veri seti üzerinde özellik seçici olarak kullanılması, literatürde özgün bir yöntem olarak yer almıştır.

1.2.5. Tezin organizasyonu

Bu tez çalışması sekiz bölümden oluşmaktadır. Birinci bölümde, tez ile ilgili detaylı bir literatür taraması ve incelemesi yapılmıştır. Tezin amacı, gerekçeleri ve çıktıları ifade edilerek literatüre yaptığı katkıları bu bölümde belirtilmiştir. Tezin organizasyonu da bu bölümde sunulmuştur.

İkinci bölümde, meta-sezgisel algoritmalar üzerinde durulmuş ve fizik tabanlı, kimya tabanlı, biyoloji tabanlı, matematik tabanlı, sosyal tabanlı, müzik tabanlı, spor tabanlı, sürü zekâsı tabanlı olarak sınıflandırılan yöntemler açıklanmıştır.

Üçüncü bölümde, tez çalışmasının da önemli bir parçası olan kaotik haritalar hakkında detaylı bilgi verilmiştir.

Dördüncü bölümde, tez çalışmasında önerilen yöntemler açıklanmıştır. Bu yöntemler: *Lojistik-Gauss Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Yöntemi*, *Lojistik, Singer Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Yöntemi*, *Kaotik Harita ve Darcy Teoremi Tabanlı Yeni Bir Optimizasyon Algoritması*, *Kaotik Harita ve Dalton Yasası Tabanlı Yeni Bir Optimizasyon Algoritması*'dır. Bu bölümde, algoritmalarındaki temel bilgiler ve fizik kanunları açıklanacaktır.

Beşinci bölümde, önerilen yöntemler, kıyaslama fonksiyonları ve problemler üzerinde test edilmiş ve sonuçlar karşılaştırmalı olarak verilmiştir.

Altıncı bölümde, Kaotik Darcy Optimizasyon Algoritması özellik seçici olarak kullanılmış ve elde edilen sonuçlar bu bölümde verilmiştir.

Yedinci bölümde ise sonuç ve öneriler bölümü yer almaktadır. Bu bölümde, geliştirilen yöntemlerin genel bir değerlendirilmesine yer verilmiştir. Son olarak sekizinci bölümde kaynaklar yer almaktadır.

2. META-SEZGİSEL ALGORİTMALAR

Sezgisel algoritmaların üstünde yer alan meta-sezgisel algoritmalar, problem çözümlerinde hangi yöntemin kullanılacağına karar veren yöntemlerdir [2]. Meta-sezgisel algoritmalar, kod basitliği ve anlaşılabilirlik, esnek ve uyarlanabilir, analiz kolaylığı sağlayabilir ve başarılı hesaplama zamanı gibi kriterlere sahip olmalıdır. Sezgisel algoritmalar, bazı durumlarda başlangıç çözümlerine bağlı olarak yerel en iyi çözümler üretebilmektedir. Bu durumla genellikle iteratif yöntemlerde karşılaşılmaktadır. Böyle durumlarda, yerel en iyi çözümler, genel en iyi çözümlerden daha uzak değerler verebilmektedir. Sezgisel algoritmaların dezavantajlarını gidermek için basitlik ilkesi saklı kalmak şartıyla aşağıdaki durumlar dikkate alınabilir [123]:

- Çok sayıda başlangıç çözümü belirleyerek, algoritmanın defalarca çalıştırılması sağlanmalıdır. Fakat bu durum maliyeti artırmaktadır.
- Karmaşık komşuluk yapısı sağlanarak, daha iyi komşuluk çözümlerinin üretilmesi amaçlanmalıdır.
- Algoritmanın çalıştırılması sırasında veri toplamaya olanak sunan karmaşık öğrenme yöntemlerinin tercih edilmesi ve bu verinin her çalıştırma sonunda bazı çözümleri cezalandırıcı olarak kullanılması düşünülebilir.
- Yerel araştırma metotlarında, yerel en iyilikten kaçınmak için bazı kısıtların kullanılması gerekmektedir.

Meta-sezgisel algoritmaların diğer optimizasyon yöntemlerinden ayrılan en önemli özelliği var olan verilerin arama esnasında kullanılmasıdır. Ve bu algoritmaların büyük bir çoğunluğu sürü zekâsı tabanlı algoritmalarda meydana gelmektedir. Sürü tabanlı meta-sezgisel algoritmaları bazı benzer özelliklere sahiptir. Mesela; meta-sezgisel algoritmalar rastgele popülasyon değerlerinin belirlenmesiyle başlar. Bu işlem, değişkenler için belirlenen üst ve alt limitlerin arasında olacak şekilde değerlerin belirlenmesi işlemidir. Algoritmanın en başında ihtiyaç duyulan parametreler belirlenir ve bu değerler uygunluk fonksiyonuna gönderilir. Bu sayede popülasyondaki her bireyin uygunluk değeri hesaplanmış olur. Bu değerler belirlendikten sonra, algortmada bulunan adımlar takip edilerek istenilen en iyi değer için algoritma çalıştırılır. Bütün bu süreç, bazı algoritmalarda sınırlı iterasyon adedince, bazılarında ise belirli tolerans

değerine kadar sürmektedir. Bu sınırlara ulaşıldığı zaman iterasyon durmakta ve sonuçlar gönderilmektedir.

Sezgisel yöntemler Şekil 1.1’de de verildiği üzere fizik tabanlı, kimya tabanlı, biyoloji tabanlı, matematik tabanlı, sosyal tabanlı, müzik tabanlı, spor tabanlı, sürü zekâsı tabanlı ve melez sistemler olmak üzere 9 çeşitli kategoride listelenmektedir. Melez sistemler, birden fazla kategorideki algoritmaların, birlikte kullanılmasıyla ortaya çıkmıştır ve bu sebeple literatürde çok fazla sayıda melez algoritma bulunmaktadır.

Tezin bu aşamasında, literatürde bilinirliği en yüksek algoritmalar başlıklar altında sunulmuştur.

2.1. Fizik Tabanlı Algoritmalar

Literatürde, evrendeki fizik yasalarını dikkate alarak tasarlanmış pek çok optimizasyon yöntemi mevcuttur. Bunlar içerisinde en bilinen algoritmalar: Elektromanyetik Alan Optimizasyon Algoritması [124], Elektromanyetizma Optimizasyon Algoritması [124,125], Yerçekimi Arama Algoritması [126], Su Çevrimi Optimizasyon Algoritması [127], Optikten Esinlenilen Optimizasyon Algoritması [128], Büyük Patlama-Büyük Büzülme Optimizasyon Algoritması [127,128], Yapay Fizik Optimizasyon Algoritması [131], Merkezi Kuvvet Optimizasyon Algoritması [132] ve Benzetilmiş Tavlama Algoritması’dır [133].

Elektromanyetik Alan Optimizasyonu Algoritması (EAOA), elektromanyetik alan teorisine dayanan sürü tabanlı bir algoritmadır. Bu algoritma, akımın geçtiği tel üzerinde oluşan elektromanyetik alanı baz alır. Elektromanyetik alan teorisinde, artı yüklü durumlar en iyi çözüm olarak kabul edilirken, eksi yüklü durumlar en kötü çözümler olarak ifade edilir [124].

Elektromanyetizma Optimizasyon Algoritması (EMOA), elektrik yüklü cisimler arasındaki itme-çekme prensibine dayalı olarak tasarlanmış bir algoritmadır. Bu algoritmada, yüklü cisimler, amaç fonksiyonuyla orantılı bir şekilde, çözüm uzayında bir noktayı temsil etmektedir. Yüklü cisimler, kendi yüklerine göre, birbirlerine karşı itme-çekme uygulamaktadır. Yüklü cisimlerin her birini etkileyen kuvvetler, ayrı bir şekilde hesaplanır ve bileşke kuvvet hesaplanır. Ve tüm parçacıklar hesaplanan bu bileşke kuvvetin vektörel yönünde hareket ettirilir. Bu sayede, iterasyonlar ilerletilerek en iyi çözüme ulaşılması amaçlanmaktadır [125].

Yerçekimi Arama Algoritması (YAA), yer çekimi kanunu ilkelerini baz alarak tasarlanmıştır [126]. Algoritmanın temel prensibinde, her cismin bir kütlesinin olduğu ve cisimlerin birbirlerini çektiği ilkesi vardır. Bu durumda cisimler, çekim kuvvetinin etkisiyle en büyük cisme doğru hareket eder. Optimizasyon yönteminde, her cisim bir çözümü ifade etmektedir. Çözüm ağırlığı ise performansları ortaya koymaktadır. İterasyonlar, cisimlerin, en büyük cisme doğru yakınsaması ile devam eder, şart sağlandığı vakit algoritmanın çalıştırılması durdurulur. Ağırlığı en fazla olan çözüm, genel en iyi çözümü vermektedir [126].

Su Çevrimi Algoritması (SÇA), akarsular, denizler ve yağmurlar arasındaki ilişkiyi modelleyen bir optimizasyon algoritmasıdır. Doğada gözlemlenen ideal hidrolojik döngüden esinlenmiştir. Dağlardaki en yüksek konumdan başlayarak, yokuş aşağı bir şekilde denizde son bulan hidrolojik döngüde, akarsular, su kütlelerini yağmurlardan ve diğer yeraltı sularından almaktadır. Akarsular denizlerde birikir. Hem akarsulardan hem bitkilerden hem de denizlerin buharlaşması ile bulutlar oluşur. Ardından yağış ile yeniden akarsu kaynaklarına su geri kazanımı görülür. Parçacık Sürü Algoritması'nın tersine, bu algoritma, akarsuların denize kadar dolaylı hareketleri konseptine dayanarak geçici en iyi çözümler sunmaktadır. Kaotik haritalar kullanılarak yeni nesil su çevrimi algoritmaları tasarlanmış ve kısıtlardan bağımsız çözümler de elde edilmeye çalışılmıştır [128,129].

Optikten Esinlenilen Optimizasyon Algoritması (OEOA), ışığın içbükey ve dışbükey aynalardaki hareket prensibinden esinlenerek geliştirilmiş bir algoritmadır. Bu algoritmada, içbükey aynalar ışığı toplarken, dış bükey aynalar ise ışığı dağıtan bir özellik sergilemektedir. Bu algoritmada, bütün yeni çözümler, gönderilen yapay ışığın fonksiyonun içbükey veya dışbükeylik gösterme durumuna göre üretilmektedir [128].

Büyük Patlama-Büyük Büzülme Optimizasyon Algoritması (BPBBOA), evrenin oluşumunu modelleyen bir algoritmadır. Algoritmanın iki aşaması vardır. İlk aşama olan büyük patlama sürecinde, ortaya çıkan enerjinin sönümlemesinin rastsal oluşumu dikkate alınır. İkinci aşama olan büyük büzülme aşamasında ise rastsal bir biçimde dağılmış olan parçacıkların bir araya gelme durumu dikkate alınmaktadır. Algoritmanın başlangıcında çözümlerin rastsal dağılımı, büyük patlamadaki parçacıkların dağılımına benzetilmektedir. Bir sonraki aşama olan büyük büzülmede ise parçacıkların bir araya gelmesiyle oluşan kütlelerin merkezi hesaplanmaktadır. Algoritmada iterasyonlar yeni patlamalar için veya eşik değeri belirlenene kadar devam

etmektedir ve kütle merkezleri her iterasyon için hesaplanmaktadır. Kütle merkezinin değişmediği durum en iyi durum olarak görülmektedir. Algoritma bittiğinde tüm parçacıklara ait çözümler en iyi çözüm çevresinde toplanmaktadır [129].

Yapay Fizik Optimizasyon Algoritması (YFOA), elektromanyetizma prensibine benzer bir şekilde çalışan bir algoritmadır. Bu algoritmada, her bireyin birer kütlesi, konumu ve hız değeri vardır ve en büyük kütleyle sahip olan birey diğer tüm bireyleri kendine çekmektedir. İterasyonlar, diğer cisimler, en büyük kütleyle ulaşıncaya veya eşik değerine ulaşıncaya kadar devam eder [131].

Merkezi Kuvvet Optimizasyon Algoritması (MKOA), doğadan ilham alan ve yerçekimi kinematiği metaforuna dayanan bir algoritmadır. Bu algoritmada prob uyduların, fiziksel evrendeki hareketi modellenmektedir. Burada, yerçekimi etkisi altında hareket eden prob uyduların bulunduğu konumlar ve ivmeler dikkate alınır. Amaç, yerçekimi kuvveti uygulayan kütlelerin etrafında kapalı bir yörüngede uyduyu sabitleyebilmektir [132].

Benzetilmiş Tavlama Algoritması (BTA), ilk defa Kirkpatrick vd. [133] tarafından ortaya atılmıştır. Kesikli (kombinatorial) optimizasyon problemlerinin çözümünde kullanılan yöntem, katıların erime noktasına kadar ısıtılması ve kristalize olana dek yavaş yavaş soğutulmasından yola çıkarak tasarlanmıştır. Maddede sıcaklığı düşürdüğünüzde atomik enerji de düşer ve düzgün yapılı bir kristal yapıda sistemin minimum enerjide olduğu kabul görmektedir. Şayet soğuma hızlı bir şekilde yapılırsa, kristalize yapıda bozulmalar ortaya çıkar. Fiziksel tavlama tekniği, belli bir sıcaklıkta sistemin enerjisinin olasılık dağılımını hesaplayan bir termodinamik yapıyı ifade eder.

2.2. Kimya Tabanlı Algoritmalar

Kimya tabanlı algoritmalar kimya ile ilgili olayları baz alarak geliştirilmiş algoritmalarlardır. Atomlardaki iyonik ve kovalent bağları baz alarak tasarlanmış Yapay Atom Algoritması ve kimyasal tepkimelerin reaksiyonundan esinlenerek önerilen Yapay Kimyasal Reaksiyon Optimizasyon Algoritması, kimya tabanlı algoritmalarla örnek olarak gösterilebilir.

Yapay Kimyasal Reaksiyon Algoritması (YKRA), açık, kapalı ve izole şeklinde gerçekleşen kimyasal reaksiyonları baz alarak geliştirilmiştir bir algoritmadır. Açık sistem içerisinde, çevreyle madde ve enerji alışverişi gerçekleşir. Kapalı sistemde

çevreyle enerji alışverişi gerçekleşirken madde alışverişinin olmadığı kabul edilir. İzole sistemlerde ise çevreyle madde ve enerji alışverişi gerçekleşmez. Yapay Kimyasal Reaksiyon Optimizasyon Algoritması'nda moleküller düzenli olacak biçimde başlangıç uzayında türetilmiştir. Bu aşamadan sonra gerçekleşen tepkimelerde moleküller içerisinde bazı bağlar koparken bazıları ise yeni bağlar oluşturmaktadır. Bu durum tepkimeye giren yapının yapısında ve enerjisinde değişimlere neden olur. Yeni oluşan molekül, çift yönlü reaksiyon içerisinde yeniden başlangıç yapısına da dönüşebilir. Aynı şekilde farklı bir reaksiyon için de yeni bir yapı meydana gelmiş olur. Bu algoritma benzer yapılara göre daha az parametreye ihtiyaç duyduğu için literatürde sıklıkla kullanılmaktadır [134,135].

Yapay Atom Algoritması (YPAA), atomların iyonik ve kovalanet bağ yapısının modellenmesi ile tasarlanmış sezgisel bir algoritmadır. Algoritma içerisinde problem atom yapısı olarak ifade edilmektedir. Elektronlar ise problemin parametreleri olarak kullanılmıştır. Başlangıç olarak birden fazla atom ile atom kümesi oluşturulmakta ve iterasyona başlanmaktadır. Algoritmanın 2 işlemi mevcuttur. İlk işlem olan kovalent bağ işleminde hangi elektronların birden çok hangi atomlar arasında ortaklaşa kullanacağı belirlenir. İkinci işlem olan iyonik bağ işleminde de, hangi elektronların değer değiştirmesi gerekliliği saptanmaktadır. İyonik bağ işlemi, kötü değere sahip elektronlara, yeni değer atamada kullanılmaktadır. Algoritmada, her atom için amaç fonksiyonu değeri hesaplanır ve bu sayede en iyi sonuç elde edilmiş olur [137].

2.3. Biyoloji Tabanlı Algoritmalar

Biyoloji tabanlı algoritmalar, genel olarak doğadaki canlıların davranışlarının modellenmesi ile ortaya çıkmıştır. Biyoloji tabanlı algoritmaların, sürü zekâsı tabanlı algoritmalar ile büyük bir kesişim alanı mevcuttur. Zira sürü zekâsı tabanlı algoritmaların büyük çoğunluğu canlı çeşitliliği içerisindeki yapıları modellemektedir. Bu yönüyle biyoloji tabanlı algoritmalar ile benzeşmektedirler. Genetik Algoritma (GA), Diferansiyel Gelişim Algoritması (DGA), Biyo-Coğrafya Temelli Optimizasyon Algoritması (BCTOA) ve Yapay Bağışıklık Algoritması (YABA) gibi algoritmalar biyoloji tabanlı algoritalara örnek olarak gösterilebilir.

Genetik algoritmalar, John Holland tarafından ortaya atılan bir optimizasyon algoritmasıdır. Algoritmada değişken ve çözüm olarak kromozomlar dikkate alınır. Algoritmada kromozomların bir araya gelmesi popülasyonu meydana getirmektedir.

Amaç fonksiyonu ile kromozomlara ait uygunluk değerleri ile hesaplanmaktadır. Genetik algoritmalar, seçim, çaprazlama ve mutasyon olmak üzere 3 farklı genetik operatöre sahiptir. Seçim operatörü, uygunluk değeri en iyi kromozomun belirlenmesini sağlar. Belirlenen en iyi kromozomlar yeni nesil oluşturmada kullanılır. Çaprazlama operatörü, alt kromozomların elde edilmesinde kullanılır. Mutasyon operatörü de alt neslin tekrara düşmemesi için kullanılır. Algoritmadaki en temel amaç en iyi çözümü sağlayacak yeni neslin meydana getirilmesini sağlamaktır [2,137].

Biyo-Coğrafya Temelli Optimizasyon Algoritması (BCTOA), canlıların yaşam alanlarını değiştirmek için gerçekleştirdiği göçleri, üremesini ve soylarının tükenmesini sayısal olarak tanımlayan bir algoritmadır. Algoritmanın temel amacı, canlılar için en iyi yaşam alanının tespit edilmesidir. En iyi yaşam alanının belirlenmesi için uygunluk indeks değişkeni kullanılır. Uygunluk indeksinin büyük olması çok fazla canlının yaşayabileceğini, düşük olması ise soylarının tükenebileceğini gösterir. En iyi çözüm, yaşam alanı için en iyi değeri veren değişken değeriyle belirlenmektedir [139].

Yapay Bağışıklık Optimizasyon Algoritması (YBOA), canlılardaki doğal immün sistemin sayısal olarak modellenmesi işlemini baz alan bir optimizasyon yöntemidir [140]. YBOA, bilgisayar virüslerinin verdiği zararlar sebebiyle ortaya atılmış bir algoritmadır. YBOA sayesinde virüs tespiti ve virüslerin yok edilmesi konularında başarımlar elde edilmiştir. Bu yöntem, yapay zekâ, makine öğrenmesi, kriptoloji gibi alanlarda literatürde kendisine sıklıkla yer bulmaktadır [140-144].

2.4. Matematik Tabanlı Algoritmalar

Matematik tabanlı algoritmalar genellikle matematiksel programlamanın ve meta-sezgisel algoritmaların birleşiminden oluşan algoritmalarlardır [31]. Matematiksel yöntemler kullanılarak en iyi çözüm yönteminin aranması işlemi yapılır. Literatürde çalışmaların büyük bir çoğunluğunda kaotik haritaların kullanıldığı da gözlemlenmektedir. Nümerik Optimizasyonlar için Kaotik Altın Sinüs Algoritması (KASA) çalışmasında Tanyıldızı ve Demir [146], sinüs fonksiyonu ile altın oran yöntemini birleştirerek en iyi çözümü geliştirmeye çalışmıştır. Demir vd. Lojistik-Gauss Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Algoritması [147] ve Lojistik-Singer Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Algoritması [148] çalışmalarında sürü tabanlı optimizasyon yöntemlerinde sayısal fonksiyonlar ve hibrit kaotik haritalar kullanarak en iyi çözümleri bulmaya çalışmıştır.

2.5. Sosyal Tabanlı Algoritmalar

Sosyal tabanlı algoritmalar, günlük yaşamdaki olaylardan etkilenilerek geliştirilmiş algoritmalarlardır. Bu algoritmalar içerisinde en çok bilinenleri Emperyalist Yarışmacı Optimizasyon Algoritması (EYOA) [149], Parleментар Optimizasyon Algoritması (POA) [149,150], Beyin Fırtınası Optimizasyon (BFO) [152] ve Öğretme-Öğrenme Tabanlı Optimizasyonu'dur (ÖÖTO) [153].

EYOA, evrimsel algoritmalarla benzer şekilde başlangıç popülasyonu ile oluşturulur [154]. Popülasyonda, bazı ülkeler en iyi emperyalist ülke olmak için belirlenir. Geri kalan ülkeler ise bu ülkelerin kolonileri olarak belirlenir. Emperyalist ülkeler için belirlenen bu koloni ülkeleri daha uygun emperyalist ülkelere doğru hareket ederler. Emperyalist ülkeler için toplam güç, emperyalist ülkenin ve kolonilerinin toplam gücüne bağlı olarak değişir. Bu algorithmada, zamanla güçlü emperyalist ülkeler gücünü artırır. Zayıf kalan ülkeler ise yıkılır. Bu algorithmada, son bir emperyalist ülke kalana kadar yarış devam eder ve sonunda bütün ülkeler son kalan emperyalist ülkenin kolonisi haline gelir [149]. Emperyalist yarışmacı algoritması, gezici robotlarda en iyi konumlandırma [155], DC motorlarda hız kontrolünün optimizasyonu [156], yüzey taşıma işleminin optimizasyonu [157], çoklu makine üretim ağındaki üretim maliyeti düşürme [158], kaotik tabanlı yapay sinir ağlarının eğitimi [159] ve renkli görüntülerin fraktal katsayı değerlendirilmesinin optimizasyonu [160] gibi çalışmalarda kullanılmıştır.

POA, parleментар sistemlerden esinlenerek geliştirilmiş algoritmadır. Parleментар sistemler, meclis ve partilerden meydana gelmektedir. Bu partiler, seçimlerde birbirleriyle yarış sürecine girer ve partilerin adayları halkın oylarıyla belirlenmektedir. Aday üyeler, grup içi yarışma yöntemiyle seçilmektedir. Seçilen grup içi üyeler ile gruplar arası yarışma yapılır ve en iyi gruplar belirlenmektedir. Güçsüz kalan gruplar ise silinmektedir. Algoritmanın sonunda bütün güçsüz gruplar silinerek en iyi grup, çözüm olarak belirlenmektedir [161].

BFO, problemlerin çözümü için yeni ve yeterli fikirler ortaya atmak amacıyla geliştirilmiş bir algoritmadır. Öneriler için uygunluk değerleri hesaplanır ve tüm fikirler içerisinde en iyi çözümün tespit edildiği bir algoritma geliştirilmiş olur [2].

ÖÖTO, öğretmenin sınıftaki öğrenciler üstündeki etki değerine göre hareket eden bir algoritmadır. Algorithmada, öğrencilerin ve öğretmenlerin, öğretme-öğrenme yetenekleri baz alınır. Öğrencilerin olduğu topluluk bir popülasyondur. Öğrenme

aşamasındaki konular ise probleme ait değişkenler olarak kabul görmektedir. Bu algoritma tasarımında en iyi çözüm öğretmen olarak kabul edilir. Öğretme sürecinde öğretmen bilgi paylaşımı yapan kişidir ve öğrencilerin başarısı üzerinde bu paylaşım çok önemli bir noktadır. Zira iyi öğretmenlerin öğrencilerinin başarı durumları da genellikle gelişim gösterir. Ve en iyi öğretmen, bu topluluklar içerisindeki en iyi çözümü sağlamış olur [153].

2.6. Müzik Tabanlı Algoritmalar

Müzik Tabanlı Algoritmalar, genellikle en iyi melodiyi elde edebilmek için geliştirilmiş algoritmalarlardır. Müzik tabanlı algoritmalar içerisinde en çok bilinen algoritma Armoni Arama Algoritması'dır (AAA) [162]. Bu algoritmanın temelinde, jaz müzisyenlerinin doğaçlama çaldıkları müzikler yatmaktadır. Popülasyon tabanlı evrimsel bir optimizasyon algoritması olan AAA [163] hem sürekli değişkenli hem de ayrık problemler üzerinde uygulanabilmektedir [164]. Armoni Arama Algoritması, evrimsel algoritmalarından farklı olarak yeni çözüm için popülasyonun içerisinde seçim yapabilmektedir. Karar değişkenlerinin başlangıç değeri gerektirmemesi de ayrı bir avantajdır.

Büyük ölçekli doğrusal olmayan dışbükey problemlerin çözümü için geliştirilen Senfoni Orkestrası Arama Algoritması [165] ve AAA'nın dezavantajlarının giderilmesi için geliştirilen ve perde ve akorların uyumundan esinlenerek geliştirilen Melodi Arama Algoritması (MAA) [164,165] da literatürde sıklıkla kullanılan optimizasyon yöntemlerine örnek olarak verilebilir.

2.7. Spor Tabanlı Algoritmalar

Spor Tabanlı Algoritmalar, şampiyonluk yarışı içerisindeki takımları baz alarak geliştirilmiş bir algoritmadır. Literatürde en çok bilinen algoritmalarından birisi Lig Şampiyonası Algoritması'dır (LŞA). Bu algoritmada, yapay olarak oluşturulan takımların birbirleriyle olan mücadelesini sayısal olarak modelleme fikri vardır. İlk olarak rastgele bir şekilde ligin programı belirlenir. Ardından takımlar ikili olarak mücadeleye girerek, galip gelmekte veya yenilmektedir. Sezon sonuna kadar devam eden algoritmada takımlar transfer yaparak güçlerini artırabilmektedir. Bu yönüyle genetik algoritmadaki mutasyon işlemine benzemektedir. Sezon tamamlandığında en iyi takım, en iyi çözüm olarak belirlenmektedir [166,167].

2.8. Sürü Zekâsı Tabanlı Algoritmalar

Doğada bulunan canlıların büyük bir kısmı sürü halinde yaşamaktadır. Sürü Zekâsı Tabanlı Algoritmalar da işte bu canlıların hareket ve davranışlarının modellenmesiyle ortaya çıkmış algoritmalarlardır. Daha önce yukarıda bahsedilen biyoloji tabanlı algoritmaların büyük bir bölümü de sürü zekâsı tabanlı algoritmalar sınıflandırmasına dahil edilebilir.

Sürü Zekâsı Tabanlı Algoritmalar içerisinde literatürde en çok bilineni kuş ve balık sürülerinin davranışlarının modellenmesi sonucu ortaya çıkan Parçacık Sürü Optimizasyonu'dur (PSO) [170]. Sürü zekâsı tabanlı optimizasyon algoritmalarının pek çoğu en iyi çözümü belirlemek için bu algortmadan esinlenmişlerdir. Yapay Arı Kolonisi Algoritması (YAKA) [53], Karınca Kolonisi Algoritması (KKA) [171], Ateş Böceği Algoritması (ABA) [172], Kedi Algoritması [173], Cırcır Böceği Optimizasyonu (CBO) [2] ve Kurt Kolonisi Algoritması [174] gibi algoritmalar sürü zekâsı tabanlı algoritmalarla örnek olarak verilebilir.

YAKA, arıların bal için gerekli nektarı bulabilmeleri için gerçekleştirdikleri süreçleri modelleyen bir algortmadır. Algoritma, nektar kaynak bölgelerinin alt ve üst limitler içerisinde belirlenmesi ile başlar. Ardından belirlenen nektar bölgelerinde eşit sayıda işçi arılar seçilir. Belirlenen bu işçi arıların asli hedefi, bölgelerinin ve komşuluğunda bulunan bölgelerin içerdiği nektar kaynaklarının miktarlarını ve kalitesini tespit etmektir. Belirlenen en iyi kaynak hafızaya atılır. Burada nektarın kalitesi uygunluk fonksiyonu ile belirlenir. Bir sonraki iterasyonda bulunan kaynağın daha iyi olduğu durumda eski kaynağın verisi hafızadan silinerek yeni kaynak verisi hafızaya eklenir. Algoritmada nektar kaynağının verisi iyileştirilemediği zaman iyileştirememeye sayacı kullanılmaktadır. Sayacın en üst değeri iterasyonların en başındayken belirlenmektedir. Çözüm iyileştirildiği durumlarda sayaç sıfırlanmaktadır. Algoritmada işçi arılar her iterasyon sonunda kovana geri gelerek diğer arıları nektar hakkında bilgilendirmektedir. Ardından gözcü arılar nektar ölçüsünde olasılıksal bir seçim gerçekleştirir. Bu seçim uygunluk değeri ile belirlenir. Olasılık değerinin belirlenmesinin akabinde [0,1] arasında rastgele bir değer üretilir ve olasılık değeri ile kıyaslanmaktadır. Şayet olasılık değeri, üretilen değerden daha büyük ise gözcü arılar nektar kaynağı bölgesinde yeni bir çözüm değeri üretmektedir. Eldeki yeni çözümler uygunluk fonksiyonu ile yeniden değerlendirilir ve şayet daha iyi değer üretilmezse

iyileştirilememe sayacı yeniden kullanılır. Süreç gözcü arıların hepsinin bölgelere dağılmalarına dek sürer. İyileştirememe sayaçları nektarın bitip bitmediği konusunda bilgi vermektedir [175].

KKK, karıncaların yuvalarına götürecekleri besinleri arama stratejilerinden yola çıkarak önerilmiş bir algoritmadır. Bu yöntemin asıl amacı karıncaların yiyecek arayışı içerisinde en kısa yolu belirleyebilmesidir. Karıncalar, yiyecek aramak için yuvalarından çıkarak rastgele bir biçimde dağılırlar ve ilerledikleri yollar üzerinde feromon adı verilen ve uçuculuk özelliği olan bir kimyasal madde bırakırlar. Aynı güzergâhtaki feromon yoğunluğuna göre diğer karıncalar en kısa yoldan yiyeceğe ulaşmış olurlar. Algoritmada en kısa yoldan yiyeceğe ulaştıran çözümler en iyi çözüm olarak belirlenmektedir [171]. Bu algoritma, işletme, ticaret, lojistik gibi alanların yanı sıra sosyal medya uygulamalarında da en sık kullanılan algoritmalarından birisidir.

AAA, ateş böceklerine ait davranışları modelleyen bir algoritmadır. Ateş böcekleri ışıklarını yakıp söndürürken diğer ateş böceklerini kendisine doğru çekmeyi amaçlar. Bu sinyal sistemi, aynı zamanda düşmanlardan korunma, topluluk halinde hareket etme ve avlarını kendilerine çekme gibi eylemleri de bünyesinde barındırır. Bu algoritmada, en iyi çözümler, probleme ait amaç fonksiyonu ile ateş böceğinin yaydığı ışığın şiddeti arasındaki ilişkiye bağlıdır. Bütün ateş böceklerinin aynı cinsten olduğu kabul edilen algoritmada, ateş böceğinin parlaklığı diğer ateş böcekleri için çekiciliği temsil etmektedir. Bu sebeple en parlak ateşböceğine doğru her iterasyonda bir hareket söz konusudur ve algoritma, en parlak ateş böceğine doğru yönelim ve varış tamamlandığında son bulur [170,174,175].

Kedi Algoritması, kedilerin yiyecek ararken sergiledikleri davranışların modellenildiği algoritmadır. Algoritma iki aşamadan oluşmaktadır. İlk aşamada kediler tetikte beklemeye başlarlar. Bu aşama kedilerin avlarını bçgrdükten sonra bulunacağı pozisyonun belirlenmesi için gereklidir. Arama süreci sırasında arama hafıza havuzu, kedinin pozisyonu, seçilmiş olan boyut arama aralığı ve değişken boyut sayısı gibi dört farklı faktör dikkate alınır. İlk olarak arana hafıza havuzundaki kedilerin hepsi için çözümler kopyalanır. Değişken boyutların sayısı baz alınarak boyutun arama aralığına ait değerlerin yüzdesi rastgele bir şekilde artırılır veya azaltılır. Ve bu sayede uygunluk fonksiyonu, aday çözümler için hesaplanır. Bütün uygunluk değerleri eşit olduğunda bütün adaylar için seçilme olasılığı hesaplanır ve kedilerin rastgele bir yere konumlandırılması işlemi uygulanır. İkinci aşamada, izleme sürecine geçilir. İzleme

süreci, kedilerin hız parametrelerinin güncellenmesi ile başlar. Hız parametresi, maksimum hız limitinin üzerinde olduğu zaman, bu değerler belirlenen limitlere çekilir. Ardından bu hız değerleri ile kedilerin pozisyon verileri güncellenir. Algoritma bitirme ölçütlerini sağladığında en iyi pozisyonunda yer alan kedinin pozisyon verisi küresel en iyi çözüm olarak kabul edilir [173].

CBO, cırcır böceklerinin davranışlarından esinlenerek geliştirilen bir algoritmadır. Cırcır böcekleri, yaşam ortamlarında sesle iletişim sağlayan ve kanat çırpışlarıyla, havanın sıcaklığını tahmin edebilme yeteneğine sahip hayvanlardır. Cırcır böceklerinin çıkarmış oldukları sesin farklı amaçları vardır. Örneğin, cırcır böceklerinin erkekleri, çıkardıkları sesler sayesinde dişi cırcır böceklerini kendilerine çekebilmektedir. Dişi cırcır böcekleri en yüksek sesli cırcır böceğine doğru yönelim göstermektedir [176,177]. Cırcır böcekleri kanatları yardımıyla sesleri çıkarabilmektedir. Aynı zamanda cırcır böceklerinde, çıkardıkları sesleri algılayan iki adet anten vardır. CBO'daki sese yönelim, ateş böceği algoritmasındaki ışığa yönelim teorisine benzemektedir. Cırcır böceği algoritması, Dolbear Yasası'nda [178-181] bahsedilen kanat çırpış sayısı ile hava sıcaklığı arasındaki ilişkiden yola çıkarak geliştirilmiştir [2].

Kurt Kolonisi Algoritması, kurtların organize bir şekilde gerçekleştirdikleri av stratejilerinden esinlenerek geliştirilmiş bir algortitmadır. Bu algortitmada, bazı kurtlar, av arama görevine atanır. Arama görevine atanan kurtlar, avı keşfettiğinde uluyarak, diğer kurtlara avı haber verir. Diğer kurtlar ise ava yaklaşarak, onu kuşatır. Bu algortitmada avın başarılı gerçekleşebilmesi ve en iyi optimizasyonu sağlamak için av öncelikle en güçlü kurda atanır [174].

3. KAOTİK HARİTALAR

Kaotik sistemleri modellerken ve bunlarla alakalı analizlerde, uzun sayı dizileri sıkça kullanılır [147]. Bu diziler, bellekte fazla yer kapladığı için, en iyi başlangıç değerlerinin ve en iyi veri kümesinin belirlenmesi önemlidir. Kaos, başlangıç şartları ve parametrelere göre, kaotik düzeni ifade eden ve doğrusal olmayan değişken yapıları içeren deterministik yapıdır [184]. Kaotik haritalar, zaman alanı içerisinde sürekli ve ayrık, uzay alanı içerisinde ise gerçek ve karmaşık olarak alt alanlarda incelenmektedir. Uzay alanı içerisinde 2 boyutlu akılcı kaotik harita [185] ve ayrık lambic haritası [186] gibi gerçek ve karmaşık harita sistemlerinden bağımsız kaotik haritalar da mevcuttur.

Kaotik haritalar, rastgele sayı dizileri için oldukça önemli bir yer tutar. Üretilen sayıların aynı seçilmemesi, geniş yayımlı spektrum özelliği göstermesi, sayıların oluşturulmasının daha az zamanda gerçekleştirilmesi ve saklama maliyeti değerinin düşük olması istenilen durumlardır. Üretilen rastgele sayıların kısıtlı alanlarda toplanmış olması, yerel en iyi noktalara takılmaya sebebiyet verdiği için kaotik haritaların kullanılması önerilmektedir [187]. Kaotik sistemler, yapay sinir ağları [186-188], özellik seçici [189,190], kümeleme analizinde [191,192], parçacık sürü optimizasyonu algoritmalarında [146,147,182,193], yapay zekâ çalışmalarında [196] gibi pek çok alanda sıklıkla kullanılmaktadır.

Literatürde sıklıkla kullanılan kaotik haritalar, bu tez çalışmasının da önemli bir parçasını oluşturmaktadır. Kaotik haritaların, zaman alanına göre sınıflandırılmış bilgileri Çizelge 3.1’de verilmiştir. Bu tez çalışmasında kaotik haritalar, hibrit model olarak kullanılmıştır. Tezde kullanılan kaotik haritalara ait matematiksel açıklamalar bu bölümün alt başlıklarında verilmiştir.

Çizelge 3.1. Zaman Alanlarına Göre Kaotik Haritalar (G: Gerçek, AK: Akılcı, K: Karmaşık, AY: Ayrık)

ZAMAN ALANI SÜREKLİ OLAN KAOTİK HARİTALAR		ZAMAN ALANI AYRIK OLAN KAOTİK HARİTALAR	
Haritanın Adı	Uzay Alanı	Haritanın Adı	Uzay Alanı
3-Hücrelerinin CNN sistemi (3-cells CNN system) [195,196]	G	2B Dairesel Kaotik Harita (2D circular chaotic map) [198]	G
Van der Pol Sistemi (Van der Pol System) [199]	G	2B Lorenz Sistemi (2D Lorenz System) [200]	G

ACT Kaotik Çekici (ACT Chaotic Attractor) [201]	G	2B Akılcı Kaotik Harita (2D Rational Chaotic Map) [185]	AK
Aizawa Kaotik Çekici (Aizawa Chaotic Attractor) [202]	G	Arnold'un Kedi Haritası (Arnold's Cat Map) [199,200]	G
Arneodo Kaotik Sistemi (Arneodo Chaotic System) [205]	G	Baker'ın Haritası (Baker's Map) [206]	G
Brusselator (Brusselator) [207]	G	Havza Kaotik Harita (Basin Chaotic Map) [208]	G
Burke-Shaw Kaotik Çekici (Burke-Shaw Chaotic Attractor) [209]	G	Beta Kaotik Harita (Beta Chaotic Map) [210]	G
Chen Kaotik Çekici (Chen Chaotic Attractor) [211]	G	Bogdanov Harita (Bogdanov Map) [212]	G
Chen-Celikovsky Sistemi (Chen-Celikovsky System) [211]	G	Çember Haritası (Circle Map) [213]	G
Chen-Lee Sistem (Chen-Lee System) [214]	G	Kompleks Kuadratik Haritası (Complex Quadratic Map) [215]	K
Chua Devresi (Chua Circuit) [216]	G	Karmaşık Kare Alma Haritası (Complex Squaring Map) [217]	K
Duffing Denklemi (Duffing Equation) [205,206]	G	Karmaşık Kübik Haritası (Complex Cubic Map) [220]	K
Finans Sistemi (Finance System) [221]	G	Clifford Fraktal Haritası (Clifford Fractal Map) [222]	G
Katlanmış-Havlu Aşırı Kaotik Haritası (Folded-Towel hyperchaotic map) [223]	G	De Jong Fraktal Haritası (De Jong Fractal Map) [224]	G
Genesio-Tesi Kaotik Çekici (Genesio-Tesi chaotic attractor) [225]	G	Gecikmeli Lojistik Sistemi (Delayed-Logistic System) [226]	G
Hadley Kaotik Sirkülasyon (Hadley Chaotic Circulation) [227]	G	Çift Rotor Haritası (Double Rotor Map) [228]	G
Halvorsen Kaotik Çeker (Halvorsen Chaotic Attractor) [229]	G	Duffing (Holmes) Haritası (Duffing Map) [230]	G
Hindmarsh-Rose Nöronal Modeli (Hindmarsh-Rose Neuronal Model) [231]	G	Diyadik Dönüşümü (Dyadic Transformation) [232]	G
Hiper Kaotik Chen Sistemi (Hyperchaotic Chen system) [233]	G	Üstel Haritası (Exponential Map) [234]	K

Hiper Newton Leipnik Sistemi (Hyper Newton-Leipnik System) [235]	G	Feigenbaum Tuhaf Kaotik Olmayan Haritası (Feigenbaum Strange Nonchaotic Map) [236]	G
Hiper-Lorenz Kaotik Çeker (Hyper-Lorenz Chaotic Attractor) [237]	G	Fraktal-Rüya Sistemi (Fractal-Dream System) [238]	G
Hiper-Lu Kaotik Sistemi (Hyper-Lu Chaotic System) [239]	G	Gauss Harita (Gauss (Gaussian) Map) [147]	G
Hiper-Rössler Kaotik Çekici (Hyper-Rössler Chaotic Attractor) [240]	G	Genelleştirilmiş Baker Haritası (Generalized Baker Map) [241]	G
Hiper Kaotik Çekici (Hyperchaotic Attractor) [242]	G	Gingerbreadman Haritası (Gingerbreadman Map) [243]	G
Ikeda Kaotik Çeker (Ikeda Chaotic Attractor) [244]	G	Grinch Ejderha Fraktal (Grinch Dragon Fractal) [245]	G
Düğüm Tutucu Kaotik Osilatör (Knot-Holder Chaotic Oscillator) [246]	G	Gumowski / Mira Haritası (Gumowski/Mira Map) [247]	G
Kuramoto-Sivashinsky Denklemi (Kuramoto-Sivashinsky equation) [248]	G	Yarı Ters Rössler Çekici (Half-Inverted Rössler Attractor) [249]	G
Li, Simetrik, Toroidal Kaos (Li Symmetrical Toroidal Chaos) [250]	G	Henon Haritası (Hénon Map) [251]	G
Lorenz Sistemi (Lorenz System) [222,223]	G	5.Dereceden Polinom ile Henon (Hénon with 5th Order Polynomial) [254]	G
Lorenz 96 Modeli (Lorenz 96 Model) [255]	G	Hitzl-Zele Haritası (Hitzl-Zele Map) [256]	G
Lotka-Volterra Sistemi (Lotka-Volterra System) [257]	G	At Nalı Haritası (Horseshoe Map) [258]	G
Moore-Spiegel Kaotik Osilatör (Moore-Spiegel Chaotic Oscillator) [259]	G	Hopalong Yörünge Fraktal (Hopalong Orbit Fraktal) [260]	G
Kaydırma-Çekici (Scroll-Attractor) [261]	G	Hiper Lojistik Harita (Hyper Logistic Map) [236]	G
Sarsıntı Devresi (Jerk Circuit) [262]	G	Ikeda Haritası (Ikeda Map) [263]	G
Newton-Leipnik Sistem (Newton-Leipnik System) [264]	G	Aralıklı Değişim Haritası (Interval Exchange Map) [265]	G

Nordmark Kesik Haritası (Nordmark Truncated Map) [266]	G	Kaplan-Yorke Haritası (Kaplan-Yorke Map) [267]	G
Nosé-Hoover Sistemi (Nosé-Hoover System) [268]	G	Düğüm Fraktal Haritası (Knot Fractal Map) [269]	G
Yeni Kaotik Sistem (Novel Chaotic System) [270]	G	Lambic Haritası (Lambic Map) [271]	AY
Pickover Fractal Haritası (Pickover Fractal Map) [222]	G	Lojistik Haritası (Logistic Map) [272]	G
Polinom A-B-C Tipi Fraktal Haritası (Polynom Type-A-B-C Fractal Map) [273]	G	Lorenz Sisteminin Poincare Dönüş Haritası (Lorenz System's Poincaré Return Map) [274]	G
Quasi-Periodicity Haritası (Quasi-Periodicity Map) [275]	G	Lozi Haritası (Lozi Map) [276]	G
Mikhail Anatoly Kaotik Çekici (Mikhail Anatoly Chaotic Attractor) [277]	G	Pomeau-Manneville Aralıklı Kaos için Haritalar (Pomeau-Manneville Maps for Intermittent Chaos) [278]	G
Rayleigh-Benard Kaotik Osilatör (Rayleigh-Benard Chaotic Oscillator) [279]	G	Quadrup-Two Yörünge Fraktal (Quadrup-Two Orbit Fractal) [280]	G
Rikitake Kaotik Çeker (Rikitake Chaotic Attractor) [281]	G	Shobu-Ose-Mori Parçalı-Doğrusal Haritası (Shobu-Ose-Mori Piecewise-Linear Map) [282]	G
Rössler Çekici (Rössler Attractor) [283]	G	Sinai Haritası (Sinai Map) [284]	G
Rucklidge Sistemi (Rucklidge System) [285]	G	Standart (Chirikov-Taylor) Harita (Standart (Chirikov'un-Taylor) Map) [286]	G
Sakarya Kaotik Çekici (Sakarya Chaotic Attractor) [287]	G	Simplektik Harita (Symplectic Map) [288]	G
Shaw-Pol Kaotik Osilatör (Shaw-Pol Chaotic Oscillator) [289]	G	Tanjant Haritası (Tangent Map)	G

			[290]	
Shimizu-Morioka Sistemi (Shimizu-Morioka System) [291]	G	Sinüsoidal Harita (Sinusoidal Map) [292]	G	
Sprott B-C Kaotik Sistemleri (Sprott B-C Chaotic System) [293]	G	Çadır Harita (Tent Map) [294]	G	
Sprott-Linz A-B-C-D-E-F-G-H-I-J-K-L-M-N-O-P-Q-R-S Kaotik Çekiciler (Sprott-Linz A-B-C-D-E-F-G-H-I-J-K-L-M-N-O-P-Q-R-S Chaotic Attractor) [295]	G	Tinkerbelle map (Tinkerbelle map) [296]	G	
Strizhak-Kawczynski Kaotik Osilatör (Strizhak-Kawczynski Chaotic Oscillator) [297]	G	Zaslavskii Haritası (Zaslavskii Map) [283,284]	G	
Simetrik Akış Çekici (Symmetric Flow attractor) [300]	G	Zeraoulia-Sprott Haritası (Zeraoulia-Sprott Map) [301]	G	
Sinüs Haritası (Sine Map) [302]	G			
Singer Haritası (Singer Map) [148]	G			
Thomas'ın Döngüsel Simetrik Çekici (Thomas' Cyclically Symmetric Attractor) [303]	G			
Ueda Kaotik Osilatör (Ueda Chaotic Oscillator) [304]	G			
Van der Pol Osilatör (Van der Pol Oscillator) [305]	G			
Willamowski-Rössler Modeli (Willamowski-Rössler Model) [306]	G			
WINDMI Kaotik Çekici (WINDMI Chaotic Attractor) [307]	G			
Yinelenen Harita (Iterative Map) [308]	G			

Bu bölümde her üç temel adım için literatürde var olan ve bu tez çalışmasında kullanılan yöntemler alt başlıklar halinde detaylandırılmıştır. Ayrıca tez çalışmasının süresince geliştirilen yöntemler ve modeller ilgili bölümde anlatılmıştır.

3.1. Tez Çalışmasında Hibrit Modellerde Kullanılan Tek Boyutlu Kaotik Haritalar

Bu tez çalışmasında, Lojistik-Gauss, Lojistik-Singer ve Lojistik-Sinüs hibrit kaotik haritaları kullanılmıştır. Lojistik, Gauss, Singer, Sinüs ve Tent haritalarına ait matematiksel tanımlar sırasıyla aşağıda sunulmuştur.

3.1.1. Lojistik Harita

Lojistik haritalar, literatürde en çok tercih edilen haritalardandır. Lojistik haritalar, sayı üretimi ve kriptoloji başta olmak üzere, görüntü işleme ve optimizasyon yöntemlerinde sıklıkla kullanılmaktadır [184]. Lojistik harita, doğrusal olmayan ayrık zamanlı dinamik bir haritadır. Lojistik haritaya ait matematik tanım Denklem (3.1)'de verilmiştir. Bu harita, $[0,1]$ aralığında rastgele değerler oluşturulmaktadır.

$$X_{i+1} = rxi(1 - xi), r \in (3.57, 4), x \in (0, 1) \text{ ve } x \neq \{0.25, 0.5, 0.75\} \quad (3.1)$$

3.1.2. Gauss Haritası

Gauss Haritası, literatürde en çok tercih edilen haritalardan birisidir. Ayrık zamanlı ve gerçek uzay alanlı bir harita olan Gauss Haritası, $[0, 1]$ aralığında rastgele sayı üretebilmek için kullanılmaktadır [309]. Gauss Haritası'na ait matematiksel tanım Denklem (3.2)'te verilmiştir.

$$x_{r+1} = \begin{cases} 0, & X_r = 0 \\ 1/X_r \bmod(1), & X_r \in (0, 1) \end{cases} \quad (3.2)$$

$$1/X_r \bmod(1) = \frac{1}{X_r} - \left\lfloor \frac{1}{X_r} \right\rfloor$$

3.1.3. Singer Haritası

Singer Haritası, sürekli zaman alanına sahip, gerçek uzay alanında değerler üretebilen bir kaotik haritadır [310]. Singer Haritası'na ait matematiksel tanım Denklem (3.3)'te verilmiştir.

$$x_{i+1} = m(7.86x_i - 23.31x_i^2 + 28.75x_i^3 - 13.302875x_i^4) \quad (3.3)$$

3.1.4. Sinüs Haritası

Sinüs haritası, sinyalleri $[0, \pi]$ 'den $[0, 1]$ 'e dönüştüren ve orijinal çıkış aralığını koruyan sürekli zaman aralığına sahip kaotik bir haritadır [295,296]. Sinüs Haritası'na ait matematiksel tanım Denklem.(3.4)'te verilmiştir.

$$x_{i+1} = S(x_i) = \mu \sin(\pi x_i) \quad (3.4)$$

3.1.5. Tent (Çadır) Haritası

Çadır haritası literatürde en çok tercih edilen haritalardan birisidir. Bu harita şifrelemede, rastgele sayı üretiminde, yer değiştirme kutusu üretiminde ve optimizasyon yöntemlerinde rastgele parçacık üretimi için kullanılmaktadır. Çadır haritasının en önemli özelliği istatistiksel olarak optimum sonuçlar üretebilen bir fonksiyon olmasıdır. Çadır haritasının matematik tanımı Denklem (3.5)'te, çatallanma diyagramı ise Şekil 4.3'te verilmiştir [297,298].

$$x_{n+1} = \begin{cases} \frac{x_n}{0.7}, & \text{if } x_n < 0.7 \\ \frac{10}{3}x_n(1 - x_n), & \text{if } x_n \geq 0.7 \end{cases} \quad (3.5)$$

4. ÖNERİLEN YÖNTEMLER

Bu tez çalışması kapsamında yeni nesil hibrit kaotik haritalar geliştirilmiş ve bu kaotik haritalar geliştirilen yeni nesil parçacık sürü tabanlı ve kök tabanlı optimizasyon algoritmalarında kullanılmıştır. Tezin bu bölümünde, Lojistik-Gauss Hibrit Kaotik Tabanlı Yeni Bir Optimizasyon Yöntemi, Lojistik-Singer Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Yöntemi, Kaotik Harita Tabanlı Darcy Optimizasyon Yöntemi, Kaotik Harita Tabanlı Dalton Optimizasyon Yöntemi ve Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemleri'ne ait detaylı bilgiler alt başlıklar halinde sunulmuştur.

4.1. Lojistik-Gauss Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Yöntemi

Kaotik haritalar, karmaşık problemlerin çözümünde, kriptolojide, sayısal analizde, sezgisel algoritmalarda ve görüntü şifrelemede sıklıkla kullanılmaktadır. Kaotik sistemlerde sayı dizilerinin üretilmesi basit ve hızlıdır. Başlangıç parametreleri ve fonksiyonlar ile istenilen büyüklükte dizileri üretmek mümkündür.

Kaotik haritalar ile oluşturulan sayıların, geniş spektrumlu ve periyodik olmayan yapıya sahip olduğu teorik olarak ispat edilmiştir [187].

Bu çalışmanın önemli parametreleri şunlardır [147]:

- Çalışmada yeni nesil 1D hibrit kaotik harita kullanılmıştır (Lojistik-Gauss Harita)
- Kaotik haritaların iyi istatistiki sonuçlar ürettiği bilinmektedir. Bu sebeple diferansiyel tabanlı yeni bir optimizasyon yöntemi geliştirilmiştir ve algorithmada hız parametresi, önerilen hibrit kaotik yöntem kullanılarak hesaplanmıştır.
- Önerilen kaotik optimizasyon yönteminin başarımını ölçmek için literatürde sıkça kullanılan 6 farklı kıyaslama fonksiyonu üzerinde çalıştırılmıştır. Önerilen algoritmanın, tüm kıyaslama fonksiyonlarında en iyi sonuçları elde ettiği görünmüştür.

Önerilen hibrit kaotik haritaya ait matematiksel ifade Denklem (4.1)'de verilmiştir.

$$r(i + 1) = \text{mod} \left(\frac{1}{4 * r(i) * (1 - r(i))} \right) \quad (4.1)$$

4.1.1. Önerilen Lojistik-Gauss Hibrit Kaotik Harita Tabanlı Sürü Optimizasyon Yöntemi

Bu çalışmada, lojistik ve gauss kaotik haritaları hibrit bir modelle kullanılarak yeni bir sürü optimizasyon yöntemi önerilmiştir. Bu yöntemin asıl amacı, daha geniş spektrumlu kaotik sistemleri kullanarak en iyi noktayı bulmaktır. Literatürde önerilen kaotik arama yöntemlerinin aksine adım büyüklüğü parametresinde, lojistik-gauss hibrit kaotik haritasının ürettiği değerler kullanılmıştır [147].

Önerilen yöntemin, iyi değeri seçme, tohum değerlerini meydana getirme, kaotik yöntem ile rastgele değer oluşturma, parçacık değerlerini güncelleme ve en iyi değeri güncelleme gibi aşamaları mevcuttur.

Önerilen yönteme ait sözde kod Şekil 4.1’de sunulmuştur.

<i>Girdi: Üst sınır (US), alt sınır (AS), tohum değerleri, parçacık sayısı (pn), uygunluk fonksiyonu</i> <i>Çıktı: En iyi değer</i>
<pre>01: for i=1 to pn do 02: Parçacıkları rastgele üret $p_i = rand[0,1] * [US - AS] + AS$ 03: end for i 04: En iyi yerel uygun değer bul. 05: for k=1 to max_iter do // Maximum iterasyon sayısına dek dön 06: Tohum değerlerini belirle 07: for j=1 to pn do 08: Tohum değerlerini güncelle 09: end for j 10: En iyi değeri güncelle 11: end for k</pre>

Şekil 4.1. Lojistik-Gauss Tabanlı Kaotik Sürü Optimizasyonuna Ait Sözde Kod

4.2. Lojistik-Singer Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Yöntemi

Bütün meta-sezgisel optimizasyon yöntemi rastgelelik sistemine göre hareket etmektedir. Üretilen rastgele sayıların en iyi değerlere sahip olması için kaotik haritalardan faydalanılmaktadır. Bu çalışmada, lojistik ve singer kaotik haritaları

kullanılarak yeni bir hibrit kaotik harita önerilmiş ve bu kaotik harita yardımıyla yeni bir kaotik tabanlı optimiasyon algoritması geliştirilmiştir.

Çalışmanın literatüre katkıları aşağıda sunulmuştur.

- Önerilen yeni kaotik hibrit harita, lojistik ve singer haritaları ile oluşturulmuştur. Yeni kaotik harita, daha geniş bir kaotik etki alanına sahip olduğu görülmüştür.
- Önerilen kaotik tabanlı sürü optimizasyon algoritmasında, parçacık değerlerinin güncellenmesinde hibrit lojistik-singer kaotik haritası kullanılmıştır.
- Önerilen kaotik optimizasyon yöntemi performans ölçümü için literatürde sıkça kullanılmakta olan 6 farklı kıyaslama fonksiyonu üzerinde çalıştırılmıştır. Önerilen algoritmanın, tüm kıyaslama fonksiyonlarında en iyi sonuçları elde ettiği tespit edilmiş ve sonuçlar Çizelge 5.2’de verilmiştir.

Önerilen Lojistik-Singer Hibrit Kaotik Haritası’na ait matematiksel tanım Denklem (4.2)’de verilmiştir.

$$x_{n+1} = \text{mod} \left((4-r) \left(7.86rx_n(1-x_n) - 23.31((4-r)x_n(1-x_n))^2 + 28.75(rx_n(1-x_n))^3 - 13.302875((4-r)x_n(1-x_n))^4 \right), 1 \right) \quad (4.2)$$

4.2.1. Önerilen Lojistik-Singer Hibrit Kaotik Harita Tabanlı Sürü Optimizasyon Yöntemi

Bu çalışmada, lojistik ve singer kaotik haritaları hibrit bir model ile kullanılmıştır. Daha iyi kaotik arama aralığında çözümler elde edebilmek için daha geniş spektrumlu hibrit kaotik lojistik-singer haritası tercih edilmiştir. Önerilen hibrit kaotik haritaya ait adımlar sırasıyla aşağıda sunulmuştur.

1.Adım: Başlangıç popülasyonu rastgele değerler ile oluşturulur.

$$p_i = \text{rand}[0,1] \times (US - AS) + LB, i = \{1,2, \dots, n\} \quad (4.3)$$

Verilen denklemde p_i değeri, $\text{rand}[0,1]$ 0 ile 1 arasında rastgele değer üreten fonksiyon, UB arama uzayının üst sınırı, LB arama uzayının alt sınırı ve n kullanılan parçacık sayısını göstermektedir.

2.Adım: En iyi değer seçilir.

$$p_{best} = \min(f(p_i)) \quad (4.4)$$

Burada f , amaç fonksiyonudur.

3.Adım: Tohum değeri oluşturulur.

$$x_1 = \frac{k}{iter + 1}, h = \frac{4k}{iter + 1} \quad (4.5)$$

k döngü değişkenin anlık değerini, $iter$ maksimum döngü sayısını, h kaos çarpanını ve x_1 tohum değerini ifade etmektedir.

4.Adım: Her parçacık değeri için x sayı dizisi oluşturulur.

5.Adım: Parçacık değerleri güncellenir.

$$p_i = p_i + x_i(p_{best} - p_i) \quad (4.6)$$

6.Adım: Parçacık değerleri sınır değerini aştı ise parçacık aşağıda verilen şekilde güncellenir.

$$p_i = \begin{cases} p_{best}, & p_i < LB \\ p_{best}, & p_i > UB \end{cases} \quad (4.6)$$

7.Adım: En iyi parçacık değeri güncellenir.

8.Adım: Küresel en iyi noktaya veya maksimum iterasyon sayısına kadar 3. ve 7. adımlar tekrarlanır.

Geliştirilen algoritmaya ait sözde kod, Şekil.4.2’de verilmiştir.

<i>Girdi: Üst sınır (US), alt sınır (AS), tohum değerleri, parçacık sayısı (pn), uygunluk fonksiyonu</i> <i>Çıktı: En iyi değer</i>
01: Başlangıç popülasyonunu üret. 02: Başlangıç popülasyonunu uygunluk fonksiyonu kullanarak değerlendir ve yerel en iyi değeri bul. 03: Adım değerini hesapla $step = \frac{pn}{US-AS}$ 04: for iter 1 to MI(Maximum Iteration) do 05: Tohum değerlerini belirle (r, x_1) 06: $p_i = p_i + x_i(p_{best} - p_i)step$ 07: if $p_i < AS$ or $p_i > US$ then 08: $p_i = p_{best}$ 09: end if 10: Uygunluk fonksiyonunu kullanarak tüm parçacıkları değerlendir. 11: p_{best} 'i güncelle 12: İstenen çözüme ulaşıldıysa dur 13: end for iter

Şekil 4.2. Lojistik-Singer Kaotik Optimizasyon Algoritması’na ait sözde kod.

4.3. Kaotik Darcy Optimizasyon Yöntemi

Bu çalışmada kök gelişim algoritmasından esinlenerek yeni kaotik tabanlı Darcy Optimizasyon Algoritması geliştirilmiştir. Darcy teoremi topraktaki su yoğunluğunu tespit edebilen fiziksel bir formüldür. Geliştirilen algorithmada Darcy'nin bu özelliği köklerin suyu bulmada su yoğunluğunun fazla olduğu yere yönelmesini belirleyen matematiksel model olarak kurgulanmıştır. Bu sayede geliştirilen algoritma doğadan esinlenilen yeni bir algoritma olarak kabul edilebilir.

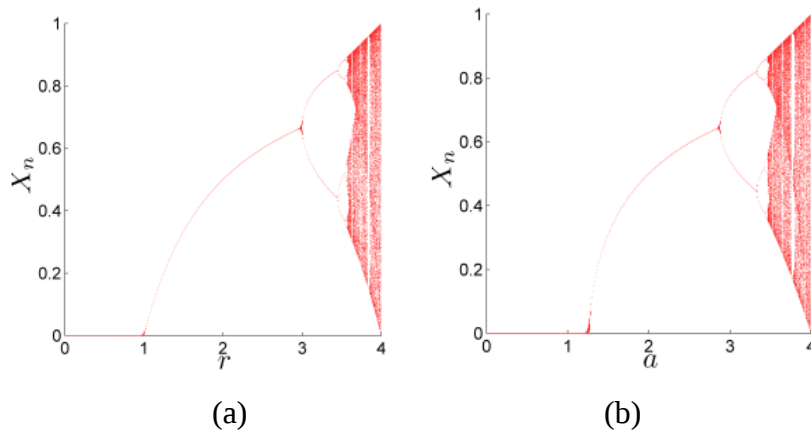
Bu algorithmada, geniş spektrumlu dağılım sağlayabilmek için Lojistik-Sinüs hibrit kaotik harita kullanılmıştır.

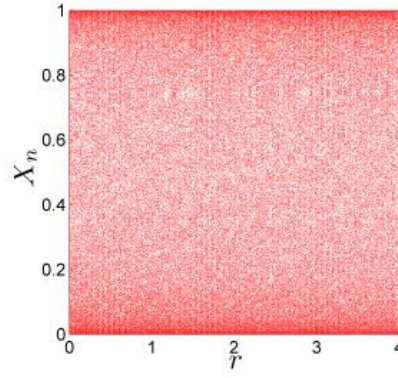
4.3.1. Lojistik-Sinüs Kaotik Haritası

Optimizasyon algoritmalarında, daha iyi bir homojen dağılım sağlayabilmek için kaotik haritaların kullanılması gerekmektedir. Bu çalışmada, daha iyi istatistiksel özellikleri doğrudan kullanabilmek için lojistik ve sinüs haritaları kullanılarak hibrit kaotik bir model önerilmiştir [43,299]. Önerilen Lojistik-Sinüs Hibrit Kaotik Harita'ya ait matematiksel tanım Denklem (4.7)'de verilmiştir.

$$x_{i+1} = \left(rx_i(1 - x_i) + \frac{(4 - r) \sin(\pi x_i)}{4} \right) (mod 1) \quad (4.7)$$

Burada x rastgele oluşturulmuş bir sayı dizisidir ve r bir kaos çarpanıdır. Lojistik-Sinüs Hibrit Kaotik Harita'ya ait kaotik aralığı göstermek için lojistik, sinüs ve Lojistik-Sinüs Hibrit Kaotik haritalarına ait çatallanma diyagramları Şekil 4.3'te gösterilmiştir.





(c)

Şekil 4.3. Çatallanma Grafiği (a) Lojistik Harita, (b) Sinüs Harita, (c) Lojistik-Sinüs Harita

Şekil 4.3'te görüldüğü gibi, geliştirilen yeni kaotik harita, lojistik ve sinüs haritalarına göre daha geniş spektrumlu bir yayılım göstermektedir.

4.3.2. Darcy Teoremi

Henry Darcy, 1803-1858 yılları arasında Fransa'da yaşamış bir bilim insanıdır. Hidroloji mühendisi olarak, kum sütunları arasındaki suyun hareketini ve su kaynağına erişimi gözlemlemiştir. Toprağın hidrolik iletkenliğinin, kesit alanı ve akış yolu uzunluğu ile doğrudan ilişkili olduğunu ispatlamıştır. Darcy Teoremi, gözenekli yapı içerisinde, toprak altındaki suyun laminar (düzenli akış hareketi) hareketlerinin tespiti için kullanılmaktadır [316]. Bu teorem, türbülans ve değişken akış hareketine sahip suyun kantitatif (miktarsal) analizi için kullanılmaz.

Bilindiği üzere, doğadaki pek çok bitki çeşidi su ve besin maddelerinin yoğunlaştığı yerlere doğru kök hareketleri sağlamaktadır. Aynı şekilde, çorak toprak, zararlı maddeler ve tuz yoğunluğundan da uzaklaşmaktadırlar. Bu çalışmada, Darcy Teoremi ile yeraltı su debisi hesaplanarak bitkilerin suyun akış yoğunluğuna doğru yönelimi yeni geliştirilen optimizasyon algoritmasına uyarlanmıştır.

Darcy Teoremi, serbest yüzeyli akifer (suyun birikebileceği geçirgen bir yapıya sahip kaya parçası) ve basınçlı akifer olmak üzere iki farklı matematiksel denkleme sahiptir. Bitkiler genellikle yüzey sularından beslendiği için, bu çalışmada serbest yüzeyli akiferlerde kullanılan Darcy Teoremi tercih edilmiştir. Serbest yüzeyli akifer için Darcy Teoremi'ne ait matematiksel tanım Denklem (4.8)'de verilmiştir

$$Q = -K * i * A, \quad i = \frac{\Delta h}{\Delta L} \quad (4.8)$$

Burada Q hacimsel akışı, K hidrolik iletkenliği, i hidrolik eğimi, A kesit alanını, L mesafeyi ifade eder. Serbest yüzeyli akifer kuyusunda r yarıçapı Dupuit hipotezi kullanılarak hesaplanır. Darcy Teoremi'nin matematiksel diğer tanımı Denklem (4.9)'da verilmiştir. Denklem (4.9)'da, h ifadesi, su seviyesine ait yükseklikleri ifade etmektedir.

$$Q = \pi * K * \left(\frac{h_2^2 - h_1^2}{\ln \left(\frac{r_2}{r_1} \right)} \right) \quad (4.9)$$

4.3.3. Önerilen Kaotik Darcy Optimizasyon Yöntemi

Kaotik Darcy Optimizasyon Yöntemi'ne ait adımlar aşağıda verilmiştir.

1.Adım: Denklem (4.10) kullanarak başlangıç parçacıkları oluşturulur.

$$P_i = rand[0,1] * (US - AS) + AS \quad (4.10)$$

2.Adım: Denklem (4.11) kullanılarak en iyi parçacık değeri bulunur.

$$P_{best} = \min(act(P)), i = \{1, 2, \dots, pn\} \quad (4.11)$$

3.Adım: Kaotik harita kullanılarak T vektörü hesaplanır. T vektörünü hesaplayabilmek için Denklem (4.7) kullanılmıştır.

4.Adım: Denklem (4.12) kullanılarak Darcy Teoremi ve T vektörü kullanılarak parçacıklar güncellenir.

$$P_i^{t+1} = P_i^t + \frac{2 \frac{P_{best}^t - P_i^t}{T_{i+1} - T_i} (US - AS)}{pn} \quad (4.12)$$

5.Adım: Güncellenmiş parçacık arama alanının alt veya üst sınırı aşıyorsa parçacık Denklem (4.13) kullanılarak güncellenir.

$$P_i = P_{best} T_i \quad (4.13)$$

6.Adım: Maksimum iterasyona veya belirlenen hata değerine ulaşana kadar 2-5 adımları tekrarlanır.

Önerilen yönteme ait akış diyagramı Şekil 4.4.'te verilmiştir.

Girdi: Üst sınır (US), alt sınır (AS), tohum değerleri, parçacık sayısı (pn), uygunluk fonksiyonu
Çıktı: En iyi değer

```

01: for i=1 to pn do
02:   Parçacıkları rastgele üret

$$x_{i+1} = \left( rx_i(1 - x_i) + \frac{(4 - r) \sin(\pi x_i)}{4} \right) (mod\ 1)$$

03: end for i
04: En iyi yerel uygun değer bul.
05: for k=1 to max_iter do // Maximum iterasyon sayısına dek dön
06:   Tohum değerlerini belirle
07:   for j=1 to pn do
08:     Tohum değerlerini güncelle

$$P_i = P_{best} T_i$$

09:   end for j
10:   En iyi değeri güncelle
11: end for k

```

Şekil 4.4. Kaotik Darcy Optimizasyon Algoritması'na ait sözde kod

4.4. Kaotik Dalton Optimizasyon Yöntemi

Bu çalışmada, kök gelişim algoritmasından esinlenerek kaotik tabanlı Dalton Optimizasyon Algoritması geliştirilmiştir. Dalton Yasası, suyun buharlaşma sürecini matematiksel olarak modellemeye çalışan ampirik tabanlı bir yasadır. Yüzey sularının buharlaşması, bitki köklerinin suya yönelimi hakkında bizlere bilgi vermektedir. Bu yönüyle önerilen bu algoritma, doğadan esinlenilen bir algoritma olarak kabul edilebilir. Bu algortmada, geniş spektrumlu dağılım sağlayabilmek ve kıyaslama yapabilmek için Lojistik, İteratif, Gauss, Çember, Sinüs, Çadır ve Singer haritaları kullanılmıştır. Algoritmaya ait arka plan alt başlıklarda açıklanmıştır.

4.4.1. Kullanılan Kaotik Haritalar

Bu çalışmada, Kaotik Dalton Optimizasyon Algoritması'nı geliştirirken 6 farklı tek boyutlu kaotik harita kullanılmıştır. Bu kaotik haritalar sırasıyla; Lojistik, İteratif, Gauss, Çember, Sinüs, Çadır ve Singer haritalarıdır. Bu haritalara ait matematiksel tanımlar Çizelge 4.1'te verilmiştir.

Çizelge 4.1. Dalton Optimizasyon Algoritması'nda kullanılan kaotik haritalar

Sayı	Kaotik Harita	Denklem
1	Lojistik	$T_{i+1} = rT_i(1 - T_i), r \in [3.57, 4]$ (4.14)
2	İteratif	$T_{i+1} = \sin\left(\frac{r\pi}{T_i}\right), r = 0.7$ (4.15)
3	Gauss	$T_{i+1} = \begin{cases} 0, T_i = 0 \\ \frac{1}{T_i \pmod{1}}, T_i \in (0, 1) \end{cases}$ (4.16)
4	Çember	$T_{i+1} = T_i + b - \frac{r}{2\pi} \sin(2\pi T_i) \pmod{1}, r = 0.5, b = 0.2$ (4.17)
5	Sinüs	$T_{i+1} = \frac{r}{4} \sin(\pi T_i), r \in (0, 4)$ (4.18)
6	Çadır	$T_{i+1} = \begin{cases} \frac{10T_i}{7}, T_i < 0.7 \\ \frac{10}{3}T_i(1 - T_i), T_i \geq 0.7 \end{cases}$ (4.19)
7	Singer	$T_{i+1} = r(7.86T_i - 23.31T_i^2 + 28.75T_i^3 - 13.3028T_i^4), r = 1.07$ (4.20)

4.4.2. Dalton Yasası

Su, yapısı gereği, yüzeyindeki moleküllerin sahip olduğu kinetik enerjiden ötürü yerçekimi etkisinden kurtulabilir. Havaya geçen molekül sayısı artar ve buharlaşma meydana gelir. Buharlaşma oranını sağlayan buhar basıncı, sıcaklık, rüzgâr, su kütlesi büyüklüğü, kirlilik ve tuzlanma gibi olaylardan doğrudan veya dolaylı olarak etkilenmektedir. Suyun buharlaşması yağmur oluşumuna sebep olduğu gibi, aşırı buharlaşma da kuraklığa ve dehidrasyona neden olmaktadır. Buharlaşmayı etkileyen çok fazla faktör olduğu için buharlaşma miktarını tam olarak hesaplamak oldukça güçtür. Bununla birlikte, buharlaşma miktarı, Su Denge Yöntemi, Enerji Dengesi Yöntemi, Kütle Arama Yöntemi gibi deterministik ve Dalton Yasası, Meyer Formülü ve Rohmer Kuralı gibi ampirik yöntemler kullanılarak tahmin edilebilir. [301,302]. Bu çalışmada, kolay hesaplanabilir ve ampirik yöntemlerden birisi olan Dalton Yasası tercih edilmiştir. Dalton Yasası'na ait matematiksel tanım Denklem (4.21)'de verilmiştir.

$$E = C(e_w - e_a) \quad (4.21)$$

Bu denklemde, E = Buharlaşma Miktarı, C = Rüzgar Katsayısı, e_w = Suyun Buhar Basıncı, e_a = Hava Buhar Basıncı'dır [319].

4.4.3. Önerilen Kaotik Dalton Optimizasyon Yöntemi

Bu çalışma, Kaotik Darcy Optimizasyon Algoritması'ndan farklı olarak parçacıkları güncellemede Dalton Yasası'ndan faydalanmaktadır. Dalton Yasası tabanlı parçacık güncelleme formülüne ait matematiksel tanım Denklem (4.22)'de verilmiştir.

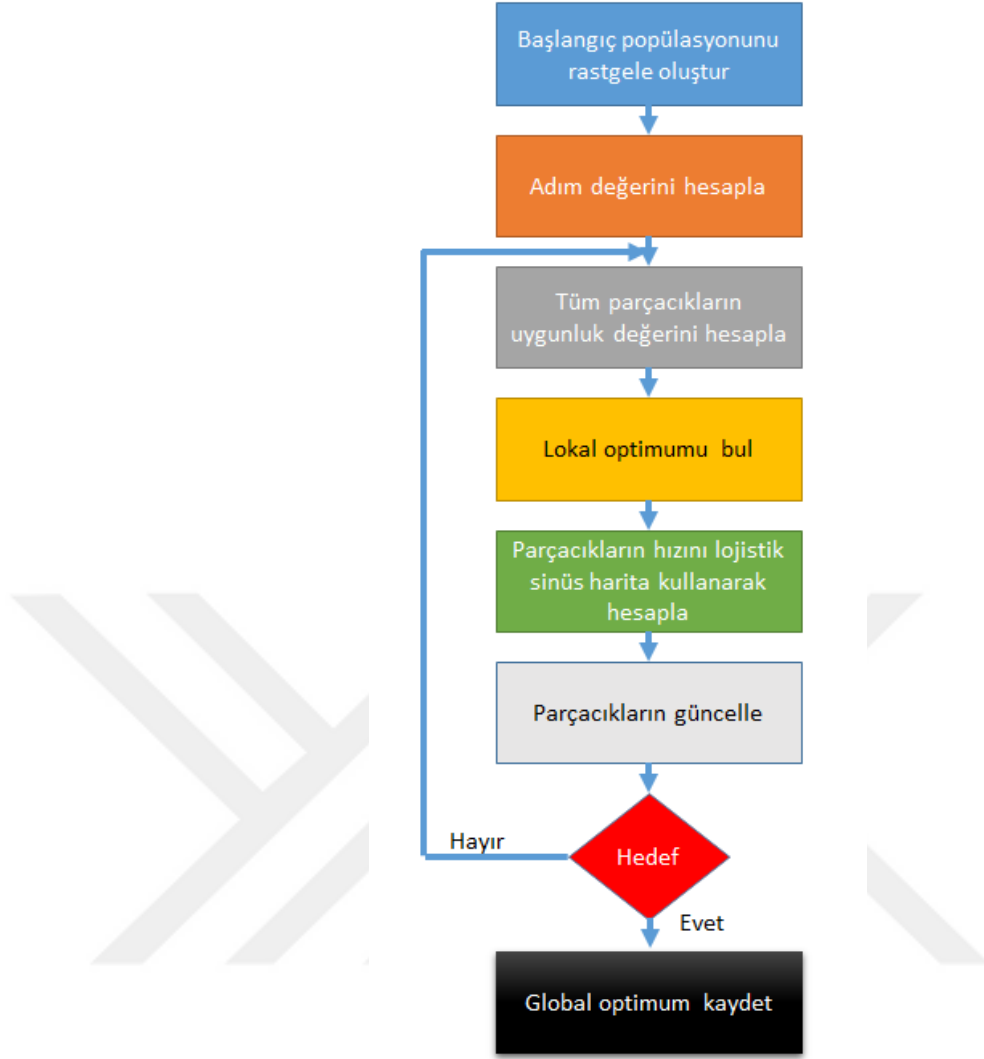
$$P_i^{t+1} = P_i^t + \frac{(P_{best}^t - P_i^t)T_i(UB - LB)}{pn} \quad (4.22)$$

4.5. Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi

Bu çalışmada, lojistik-sinüs hibrit kaotik harita kullanarak yeni bir optimizasyon yöntemi önerilmiştir. Daha geniş bir alanda dağılım kazanabilmek için hibrit kaotik harita tercih edilmiştir. Lojistik-Sinüs Hibrit Kaotik Haritası'na ait matematiksel tanımlar, Kaotik Darcy Optimizasyon Yöntemi başlığı altında sunulmuştur. Bu bölümde, geliştirilen algoritmaya ait açıklamalar verilmiştir.

4.5.1. Önerilen Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi

Kaotik optimizasyon teknikleri genellikle sinüs, lojistik, gauss, çadır haritaları gibi tek boyutlu kaotik haritaları kullanır. Bu çalışmada, yüksek performans sağlamak için hibrit kaotik bir harita olan lojistik-sinüs haritası kullanılmıştır. Kaotik optimizasyon tekniğine dayanan lojistik-sinüs haritası parçacık tabanlıdır. Kısaca LSinüs-KOA olarak ifade edilir. LSinüs-KOA'ya ait akış diyagramı Şekil 4.5'te gösterilmektedir.



Şekil 4.5. Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Algoritması'na ait akış diyagramı

Önerilen lojistik-sinüs hibrit kaotik harita tabanlı optimizasyon yönteminin adımları aşağıda verilmiştir.

1.Adım: Alt sınır ve üst sınır değerlerini kullanarak ilk popülasyonu rastgele oluşturulur.

$$p = (US - AS) \times rand[0,1] + LB \quad (4.23)$$

UB üst sınır, LB alt sınır ve $rand[0,1]$, $[0,1]$ aralığında rasgele oluşturulmuş bir sayıdır.

2.Adım: Adım değerleri hesaplanır.

$$step = \frac{US - AS}{P_{num}} \quad (4.24)$$

p_{num} parçacık sayısıdır.

3.Adım: Uygunluk fonksiyonu kullanılarak kişisel en iyi hesaplanır.

4.Adım: LojistikSsinüs Haritası kullanılarak rastgele sayılar üretilir.

5.Adım: Parçacıklar güncellenir.

$$p_i^{t+1} = p_i^t + x_i(p_{best}^t - p_i^t) \times step \quad (4.25)$$

6.Adım: Parçacıklar alt veya üst sınırları aşarsa kişisel en iyi değeri kullanarak konumunu ayarlanır.

$$p_i = p_{best} \quad (4.26)$$

7.Adım: Kişisel en iyi değerler güncellenir.

8.Adım: İstenen değerlere veya maksimum yineleme sayısına ulaşılan kadar 4-7 arasındaki adımları tekrarlanır.



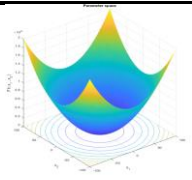
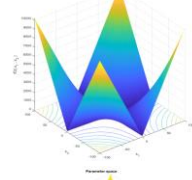
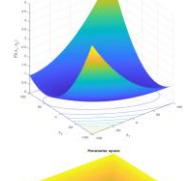
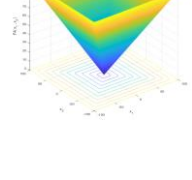
5. ÖNERİLEN YÖNTEMLERİN TEST FONKSİYONLARINA UYGULANMASI VE BAŞARI ANALİZİ

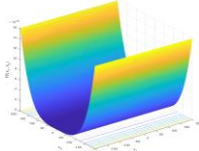
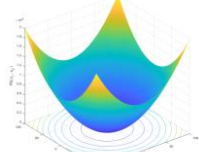
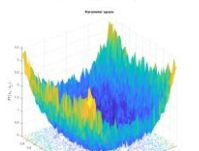
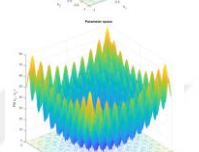
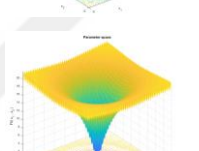
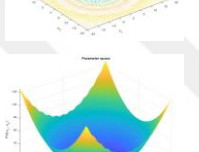
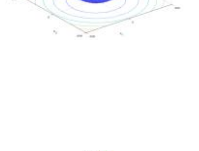
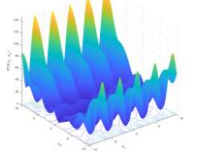
Önerilen yöntemlerin performans ölçümlerinin yapılabilmesi için literatürde sıklıkla kullanılan kıyaslama fonksiyonları ve problemler üzerinde uygulamalar çalıştırılmış ve sonuçlar, çalışmanın bu bölümünde gösterilmiştir. Uygulama, i7-7500U işlemciye, 8 GB RAM'e ve Windows 10 Pro İşletim Sistemi'ne sahip bir bilgisayarda çalıştırılmış ve MATLAB 2018a programı ve sürümü kullanılmıştır.

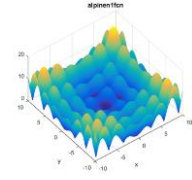
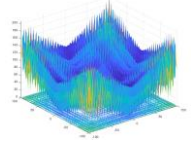
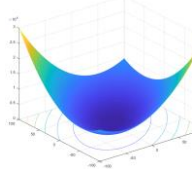
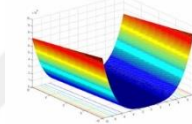
Algoritmada Çizelge 5.1'de verilen 16 farklı kıyaslama test fonksiyonu ile çalışılmıştır [285,301,304,305]. Bu kıyaslama fonksiyonlarının 8 adedi tek modlu (F1, F2, F3, F4, F5, F6, F7, F16), 8 adedi ise çok modlu (F8, F9, F10, F11, F12, F13, F14, F15) olarak ifade edilmektedir.

Çizelge 5.1'te kıyaslama fonksiyonlarına ait matematiksel tanımlar, şekiller ve hangi çalışmalarda kullanıldığına dair bilgiler yer almaktadır. Tezde geliştirilen algoritma ve yöntemler Çizelge 5.1'de sırasıyla Lojistik-Gauss Kaotik Optimizasyon Algoritması (LG-KOA), Lojistik-Singer Kaotik Optimizasyon Algoritması (LS-KOA), Kaotik Darcy Optimizasyon Algoritması (KDar-OA), Kaotik Dalton Optimizasyon Algoritması (KDal-OA) ve Lojistik-Sinüs Kaotik Optimizasyon Algoritması (LSinüs-KOA) olarak verilmiştir.

Çizelge 5.1. Çalışmalarda kullanılan kıyaslama fonksiyonları [302]

Etiket	Kıyaslama Fonksiyonu	Denklem	Şekil	Kullanılan Algoritmalar
F1	Sphere Function	$f_1(x) = \sum_{i=1}^n x_i^2, x \in [-100,100]^n$		LG-KOA, LS-KOA, KDar-OA, KDal-OA, LSinüs-KOA
F2	Schwefel 2.22 Function	$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i , x \in [-10,10]^n$		LG-KOA, LS-KOA, KDar-OA, KDal-OA, LSinüs-KOA
F3	Schwefel 1.2. Function	$f_3(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2, x \in [-100,100]^n$		LG-KOA, LS-KOA, KDar-OA, KDal-OA, LSinüs-KOA
F4	Schwefel 2.21 Function	$f_4(x) = \max\{ x_i , 1 \leq i \leq n\}, x \in [-100,100]^n$		LG-KOA, LS-KOA, KDar-OA, KDal-OA, LSinüs-KOA

F5	Rosenbrock's Function	$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2], x \in [-30, 30]^n$		LG-KOA, LS-KOA, KDar-OA, KDal-OA, LSinüs-KOA
F6	Step Function	$f_6(x) = \sum_{i=1}^n x_i + 0.5 ^2, x \in [-100, 100]^n$		LG-KOA, LS-KOA, KDar-OA, KDal-OA, LSinüs-KOA
F7	Noise Function	$f_7(x) = \sum_{i=1}^n ix_i + rand(0,1), x \in [-1.28, 1.28]^n$		KDar-OA, KDal-OA, LSinüs-KOA
F8	Rastrigin's Function	$f_8(x) = \sum_{i=1}^n (x_i^2 + 10 \cos(2\pi x_i) + 10), x \in [-5.12, 5.12]^n$		KDar-OA, KDal-OA, LSinüs-KOA
F9	Ackley's Function	$f_9(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e, x \in [-32, 32]^n$		KDar-OA, KDal-OA, LSinüs-KOA
F10	Griewank function	$f_{10}(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right), x \in [-600, 600]^n$		KDar-OA, KDal-OA, LSinüs-KOA
F11	Generalized Penalized 1 Function	$f_{11}(x) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^n u(x_i, 10, 100, 4) y_i = 1 + \frac{1}{4} (x_i + 1), u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ k(-x_i - a)^m, & x_i < -a, x \\ 0, & -a \leq x_i \leq a \end{cases} \in [-50, 50]^n$		KDar-OA, KDal-OA, LSinüs-KOA
F12	Generalized Penalized 2 Function	$f_{12}(x) = \frac{1}{10} \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4) u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ k(-x_i - a)^m, & x_i < -a, x \in [-50, 50]^n \\ 0, & -a \leq x_i \leq a \end{cases}$		KDar-OA, KDal-OA, LSinüs-KOA

F13	Alphine 1 Function	$f_{13}(x) = \sum_{i=1}^n x_i \sin(x_i) + \frac{x_i}{10} , x \in [-10, 10]^n$		LSinüs-KOA
F14	Alphine 2 Function	$f_{14}(x) = \prod_{i=1}^n \sin(x_i) \sqrt{x_i} , x \in [-10, 10]^n$		LSinüs-KOA
F15	Trid Function	$f_{15}(x) = \sum_{i=1}^n (x_i - 1)^2 + \sum_{i=1}^n x_i x_{i-1}, x \in [-100, 100]^n$		LSinüs-KOA
F16	Dixon Price	$f_{15}(x) = \sum_{i=1}^n (x_i - 1)^2 + \sum_{i=1}^n x_i x_{i-1}, x \in [-100, 100]^n$		LSinüs-KOA

5.1. Lojistik-Gauss ve Lojistik-Singer Kaotik Optimizasyon Algoritmalarına Ait Deneysel Sonuçlar

Bu bölümde önerilen Lojistik-Gauss ve Lojistik-Singer Hibrit Kaotik Optimizasyon Algoritmalarının performansı, Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock's ve Step kıyaslama fonksiyonları çalıştırılarak test edilmiştir. Ayrıca elde edilen bu sonuçlar, literatürde sıkça kullanılmakta olan Kaotik Bio-Coğrafya Tabanlı Optimizasyon Algoritması (KBTOA) [322], Kaotik Kril Sürüsü Optimizasyonu (KKSOA) [323] ve Kaotik Yerçekimi Arama Optimizasyonu Algoritması (KYAOA) [324] isimli sürü tabanlı optimizasyon yöntemleriyle de karşılaştırılmıştır. Karşılaştırmada kullanılan algoritmaların popülasyon sayısı 40 ve çalıştırma sayısı 30 defa olarak belirlenmiştir. Alınan sonuçlar Çizelge 5.2'de verilmiştir. Önerilen yöntemler, aynı kıyaslama fonksiyonları ve optimizasyon yöntemleri üzerinde test edildiği için aynı başlık ve tablo altında gösterilmiştir.

Çizelge 5.2. Kaotik haritalı optimizasyon algoritmalarıyla karşılaştırma sonuçları

Etiket	KBTOA	KKSOA	KYAOA	LG-KOA	LS-KOA
F1	1.77×10^0	7.89×10^{-1}	1.18×10^{-16}	$1.7842e^{-128}$	0
F2	1.20×10^{-1}	2.31×10^2	2.70×10^{-1}	$4.5880e^{-139}$	0
F3	7.55×10^3	4.31×10^2	6.55×10^2	$1.6587e^{-279}$	0
F4	1.71×10^1	9.71×10^0	4.86×10^0	$7.4665e^{-134}$	0
F5	4.18×10^2	1.91×10^2	5.41×10^1	0	0
F6	1.45×10^0	5.15×10^{-1}	7.77×10^{-17}	0	0

Çizelge 5.2'deki sonuçlar incelendiğinde, önerilen algoritma, literatürde daha önce önerilmiş olan 3 farklı kaotik tabanlı optimizasyon yöntemiyle ve 6 farklı kıyaslama fonksiyonu üzerinde çalıştırılmıştır. Karşılaştırma sonucunda önerilen algoritmanın bütün fonksiyonlar içerisinde en iyi veya en iyiye yakın sonuçları verdiği tespit edilmiştir.

5.2. Kaotik Darcy ve Kaotik Dalton Optimizasyon Algoritmaları'na Ait Deneysel Sonuçlar

Kaotik Darcy ve Dalton Optimizasyon Algoritmaları, bitkilerin kök gelişiminden esinlenerek geliştirilmiş olan iki farklı optimizasyon yöntemidir. Bu algoritmalara ait matematiksel tanımlar önceki başlıklarda verilmiştir.

Çalışmanın bu bölümünde, Kaotik Darcy ve Dalton Önerilen yöntemlerin performans ölçümlerine ait deneysel sonuçlara yer verilmiştir. Her iki optimizasyon algoritması da benzer kıyaslama fonksiyonları ve algoritmalarla beraber çalıştırıldığı için aynı başlık altında sunulmuştur.

Kaotik Darcy ve Dalton Optimizasyon Algoritmaları'na ait performans ölçümü için, Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock's, Step, Noise, Rastrigin's, Ackley's, Griewank, Generalized Penalized 1, Generalized Penalized 2, olmak üzere 12 farklı kıyaslama fonksiyonları kullanılmıştır. Ayrıca algoritma, 7 farklı kaotik harita (Lojistik, İteratif, Gauss, Çember, Sinüs, Çadır ve Singer Haritaları) üzerinde çalıştırılmıştır.

Algoritmalar için kullanılan parametreler sırasıyla şu şekilde verilmiştir:

- Maksimum Döngü Sayısı = 500
- Popülasyon Sayısı = 40
- Boyut = 30

Kaotik Darcy ve Dalton Optimizasyon Algoritmaları'nın kıyaslama fonksiyonları üzerinde çalıştırılması sonucunda elde edilen sonuçlar sırasıyla Çizelge 5.3 ve Çizelge 5.4'te verilmiştir.

Çizelge 5.3. Kaotik Darcy Optimizasyon Algoritması'nın Kıyaslama Fonksiyonları Üzerinde Çalıştırılması Sonucu Elde Edilen Sonuçlar

Etiket	Kriter	Lojistik	İteratif	Gauss	Çember	Sinüs	Çadır	Singer
F1	Ortalama	0	0	3.51×10^{-66}	0	0	0	0
	Standart Sapma	0	0	1.64×10^{-65}	0	0	0	0

F2	Ortalama	0	0	0	0	0	0	0
	Standart Sapma	0	0	0	0	0	0	0
F3	Ortalama	0	0	2.09×10^{-67}	0	0	0	0
	Standart Sapma	0	0	8.71×10^{-67}	0	0	0	0
F4	Ortalama	0	0	1.29×10^{-33}	0	0	0	0
	Standart Sapma	0	0	6.15×10^{-33}	0	0	0	0
F5	Ortalama	0	0	0	0	0	0	0
	Standart Sapma	0	0	0	0	0	0	0
F6	Ortalama	0	1.02×10^{-34}	5.42×10^{-23}	0	0	0	0
	Standart Sapma	0	5.26×10^{-34}	2.02×10^{-22}	0	0	0	0
F7	Ortalama	3.73×10^{-4}	1.41×10^{-4}	2.18×10^{-4}	1.35×10^{-4}	2.29×10^{-4}	2.26×10^{-4}	0
	Standart Sapma	5.28×10^{-4}	1.62×10^{-4}	2.97×10^{-4}	1.56×10^{-4}	2.81×10^{-4}	3.80×10^{-4}	0
F8	Ortalama	0	0	0	0	0	0	0
	Standart Sapma	0	0	0	0	0	0	0
F9	Ortalama	8.88×10^{-16}	8.88×10^{-16}	4.00×10^0	8.88×10^{-16}	8.88×10^{-16}	8.88×10^{-16}	8.88×10^{-16}
	Standart Sapma	0	0	8.13×10^0	0	0	0	0
F10	Ortalama	2.51×10^{-2}	0	2.97×10^{-1}	7.90×10^{-3}	2.13×10^{-2}	1.03×10^{-1}	5.26×10^{-3}
	Standart Sapma	2.37×10^{-2}	0	2.25×10^{-1}	4.00×10^{-3}	2.31×10^{-2}	1.24×10^{-1}	5.00×10^{-3}
F11	Ortalama	4.37×10^{-31}	4.71×10^{-31}	4.36×10^{-13}	4.71×10^{-31}	4.71×10^{-31}	4.71×10^{-31}	4.71×10^{-31}
	Standart Sapma	8.90×10^{-47}	8.90×10^{-47}	1.00×10^{-12}	8.90×10^{-47}	8.90×10^{-47}	8.90×10^{-47}	8.90×10^{-47}
F12	Ortalama	1.34×10^{-32}	1.34×10^{-32}	6.24×10^{-14}	1.34×10^{-32}	1.34×10^{-32}	1.34×10^{-32}	1.34×10^{-32}
	Standart Sapma	5.56×10^{-48}	5.67×10^{-48}	1.31×10^{-13}	5.56×10^{-48}	5.56×10^{-48}	5.56×10^{-48}	5.56×10^{-48}

Çizelge 5.4. Kaotik Dalton Optimizasyon Algoritması'nın Kıyaslama Fonksiyonları Üzerinde Çalıştırılması Sonucu Elde Edilen Sonuçlar

Etiket	Kriter	Lojistik	İteratif	Gauss	Çember	Sinüs	Çadır	Singer
F1	Ortalama	0	0	2.51×10^{-78}	0	0	0	0
	Standart Sapma	0	0	6.77×10^{-78}	0	0	0	0
F2	Ortalama	0	0	8.98×10^{-19}	0	4.78×10^{-85}	0	0
	Standart Sapma	0	0	4.92×10^{-18}	0	2.14×10^{-84}	0	0
F3	Ortalama	0	0	1.96×10^{-77}	0	0	0	0
	Standart Sapma	0	0	9.13×10^{-77}	0	0	0	0
F4	Ortalama	0	0	1.75×10^{-39}	0	0	0	0
	Standart Sapma	0	0	3.06×10^{-39}	0	0	0	0
F5	Ortalama	0	0	0	0	0	0	0
	Standart Sapma	0	0	0	0	0	0	0
F6	Ortalama	2.91×10^{-2}	3.95×10^{-2}	1.94×10^{-5}	3.71×10^{-2}	1.94×10^{-2}	2.57×10^{-2}	0

F7	Standart Sapma	2.87x10 ⁻²	4.68x10 ⁻²	8.21x10 ⁻⁵	5.51x10 ⁻²	3.04x10 ⁻²	2.81x10 ⁻²	0
	Ortalama	7.47x10 ⁻⁵	8.21x10 ⁻⁵	2.55x10 ⁻⁴	1.99x10 ⁻⁴	1.18x10 ⁻⁴	2.17x10 ⁻²	0
F8	Standart Sapma	7.04x10 ⁻⁵	8.22x10 ⁻⁵	2.23x10 ⁻⁴	3.39x10 ⁻⁴	1.07x10 ⁻⁴	1.48x10 ⁻²	0
	Ortalama	0	0	6.38x10 ⁻²	0	0	0	0
F9	Standart Sapma	0	0	2.42x10 ⁻¹	0	0	0	0
	Ortalama	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶
F10	Standart Sapma	0	0	0	0	0	0	0
	Ortalama	5.56x10 ⁻²	0	5.01x10 ⁻¹	1.30x10 ⁻³	1.96x10 ⁻²	1.60x10 ⁻¹	5.26x10 ⁻³
F11	Standart Sapma	4.48x10 ⁻²	0	5.38x10 ⁻¹	1.00x10 ⁻³	1.59x10 ⁻²	3.07x10 ⁻¹	5.00x10 ⁻³
	Ortalama	1.33x10 ⁻¹	2.93x10 ⁻¹	1.03x10 ⁻¹	7.89x10 ⁻¹	1.04x10 ⁰	3.96x10 ⁰	4.71x10 ⁻³¹
F12	Standart Sapma	3.06x10 ⁻¹	6.05x10 ⁻¹	5.68x10 ⁻¹	1.03x10 ⁰	1.03x10 ⁰	5.11x10 ⁰	8.90x10 ⁻⁴⁷
	Ortalama	1.65x10 ⁻²	1.06x10 ⁻²	4.35x10 ⁻⁶	3.19x10 ⁻²	3.00x10 ⁻²	6.00x10 ⁻²	1.34x10 ⁻³²
F12	Standart Sapma	1.49x10 ⁻²	1.18x10 ⁻²	2.15x10 ⁻⁵	2.76x10 ⁻²	3.02x10 ⁻²	3.32x10 ⁻²	5.56x10 ⁻⁴⁸

Çizelge 5.3 ve Çizelge 5.4’te 1×10^{-100} ’den küçük sonuçlar 0 olarak kabul edilir.

Önerilen Kaotik Darcy ve Dalton Optimizasyon Algoritmaları, literatürde sıklıkla kullanılan Sinüs-Kosinüs Algoritması (SKA) [15], Balina Optimizasyon Algoritması (BOA) [325], Çekirge Optimizasyon Algoritması (ÇOA) [326], Parçacık Sürü Algoritması (PSO) [170], Gri Kurt Optimizasyonu (GKO) [327], Çoklu Evren Optimizasyonu (ÇEO) [328] ve Kaotik Dinamik Ağırlıklı Parçacık Sürü Optimizasyonu (KDA-PSO) ile karşılaştırılmış ve sonuçlar Çizelge 5.5’te verilmiştir.

Çizelge 5.5. Kaotik Darcy ve Dalton Optimizasyon Algoritmaları’nın Literatürde Sıklıkla Kullanılan Optimizasyon Yöntemleriyle Karşılaştırılması

Etiket	Kriter	SKA	BOA	ÇOA	PSO	GKO	ÇEO	KDA-PSO	KDal-OA	KDar-OA
F1	Ortalama	1.11x10 ¹	1.41x10 ⁻³⁰	1.72x10 ⁻⁸	3.90x10 ⁻¹	6.59x10 ⁻²⁸	2.08x10 ¹	0	0	0
	Standart Sapma	1.83x10 ¹	4.91x10 ⁻³⁰	1.77x10 ⁻⁸	2.03x10 ⁻¹	6.34x10 ⁻⁵	6.48x10 ⁻¹	0	0	0
F2	Ortalama	1.62x10 ⁻²	1.06x10 ⁻²¹	2.16x10 ⁰	2.03x10 ⁻¹	7.18x10 ⁻¹⁷	1.59x10 ¹	0	0	0
	Standart Sapma	2.12x10 ⁻²	2.93x10 ⁻²¹	3.08x10 ⁰	4.58x10 ⁻¹	2.90x10 ⁻²	4.47x10 ¹	3.79x10 ⁻³⁶	0	0
F3	Ortalama	2.10x10 ⁴	5.39x10 ⁻⁷	4.80x10 ⁻⁶	5.73x10 ²	3.29x10 ⁻⁶	4.53x10 ²	0	0	0
	Standart Sapma	1.12x10 ⁴	2.93x10 ⁻⁶	1.65x10 ⁻⁵	2.90x10 ²	7.91x10 ¹	1.77x10 ²	0	0	0
F4	Ortalama	7.37x10 ¹	7.25x10 ⁻²	1.33x10 ⁻⁴	4.15x10 ⁰	5.61x10 ⁻⁷	3.12x10 ⁰	0	0	0
	Standart Sapma	2.11x10 ¹	3.97x10 ⁻¹	1.42x10 ⁻⁴	9.55x10 ⁻¹	1.31x10 ⁰	1.58x10 ⁰	5.26x10 ⁻⁴³	0	0
F5	Ortalama	1.70x10 ⁶	2.78x10 ²	3.61x10 ²	2.10x10 ²	2.68x10 ¹	1.27x10 ³	2.81x10 ¹	0	0
	Standart Sapma	4.86x10 ⁶	7.63x10 ⁻¹	6.73x10 ²	2.04x10 ²	6.99x10 ¹	1.47x10 ³	4.82x10 ¹	0	0
F6	Ortalama	3.69x10 ¹	3.11x10 ¹	2.94x10 ⁻⁸	1.05x10 ⁰	8.16x10 ⁻¹	2.29x10 ¹	0	1.94x10 ⁻⁵	0
	Standart Sapma	8.84x10 ¹	5.32x10 ⁻¹	5.68x10 ⁻⁸	1.23x10 ⁰	1.26x10 ⁻⁴	6.30x10 ⁻¹	0	8.21x10 ⁻⁵	0

F7	Ortalama	6.65x10 ⁻¹	1.42x10 ⁻³	1.05x10 ⁻¹	8.38x10 ⁻⁸	2.21x10 ⁻³	5.19x10 ⁻²	0	7.47x10 ⁻⁵	0
	Standart Sapma	1.81x10 ⁰	1.14x10 ⁻³	1.86x10 ⁻¹	5.02x10 ⁻⁸	1.00x10 ⁻¹	2.96x10 ⁻²	0	7.04x10 ⁻⁵	0
F8	Ortalama	5.91x10 ¹	0	8.79x10 ⁰	2.44x10 ¹	3.10x10 ⁻¹	1.18x10 ²	0	0	0
	Standart Sapma	3.65x10 ¹	0	6.25x10 ⁰	6.18x10 ⁰	4.73x10 ¹	3.93x10 ¹	0	0	0
F9	Ortalama	1.15x10 ¹	7.40x10 ¹	6.03x10 ⁻¹	4.92x10 ⁻¹	1.06x10 ⁻¹³	4.07x10 ⁰	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶	8.88x10 ⁻¹⁶
	Standart Sapma	9.95x10 ⁰	9.89x10 ¹	8.43x10 ⁻¹	3.21x10 ⁻¹	7.78x10 ⁻²	5.50x10 ⁰	0	0	0
F10	Ortalama	1.04x10 ⁰	2.89x10 ⁻⁴	1.75x10 ⁻¹	4.96x10 ⁻¹	4.48x10 ⁻³	9.38x10 ⁻¹	0	0	5.26x10 ⁻³
	Standart Sapma	2.94x10 ⁻¹	1.58x10 ⁻³	1.12x10 ⁻¹	1.05x10 ⁻¹	6.65x10 ⁻³	5.95x10 ⁻²	0	0	5.00x10 ⁻³
F11	Ortalama	4.17x10 ⁶	3.39x10 ⁻¹	6.21x10 ⁻⁴	1.22x10 ⁻²	5.34x10 ⁻²	2.45x10 ⁰	2.93x10 ⁻¹	1.33x10 ⁻¹	4.71x10 ⁻³¹
	Standart Sapma	1.38x10 ⁷	2.14x10 ⁻¹	2.80x10 ⁻³	2.83x10 ⁻²	2.07x10 ⁻²	7.91x10 ⁻¹	8.29x10 ⁻¹	3.06x10 ⁻¹	8.90x10 ⁻⁴⁷
F12	Ortalama	1.51x10 ⁷	1.88x10 ⁰	1.80x10 ⁻³	1.08x10 ⁻¹	6.54x10 ⁻¹	2.22x10 ⁻¹	2.31x10 ⁰	4.35x10 ⁻⁶	1.34x10 ⁻³²
	Standart Sapma	2.87x10 ⁷	2.66x10 ⁻¹	4.90x10 ⁻³	7.95x10 ⁻²	4.47x10 ⁻³	8.64x10 ⁻²	4.29x10 ⁰	2.15x10 ⁻⁵	5.56x10 ⁻⁴⁸

Çizelge 5.5'te önerilen Kaotik Darcy ve Dalton Optimizasyon Algoritmaları'nın diğer optimizasyon algoritmalarından daha iyi sonuçlar verdiği gözlemlenmiştir. Ayrıca Çizelge 5.5 incelendiğinde Kaotik Darcy Optimizasyon Algoritması'nın 11 kıyaslama fonksiyonunda karşılaştırılan diğer algoritmalar içinde en iyi sonuçları elde ettiği açıkça görülmektedir.

5.3. Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi'ne Ait Deneysel Sonuçlar

Meta-sezgisel algoritmalar, matematiksel olarak çözülmesi çok zor olan problemlerin çözümü için kullanılmaktadır. Optimizasyon yöntemlerinin temel amacı önceden tanımlanmış koşullar altında en iyi çözüme ulaşabilmektir [122,313-316]. Kaotik haritaların hibrit modelleme ile kullanılmasının optimizasyon algoritmalarının başarımını artırdığı yapılan çalışmalar ile gözlemlenmiştir [146,285]. Bununla birlikte, hibrit tek boyutlu kaotik haritaların da genellikle klasik tek boyutlu kaotik haritalardan daha yüksek performansa sahip olduğu gözlemlenmiştir [302]. Bu çalışmada, diğer optimizasyon yöntemlerinden daha iyi sayısal sonuçlar elde etmek için hibrit kaotik harita olarak Lojistik-Sinüs Hibrit Kaotik Haritası tercih edilmiştir. Önerilen yöntem sürü tabanlı bir yöntemdir ve başarımını test etmek için literatürde sıkça kullanılan 16 farklı sayısal kıyaslama fonksiyonu (Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock's, Step, Noise, Rastrigin's, Ackley's, Griewank, Generalized Penalized 1 ve Generalized Penalized 2, Alphine 1, Alphine 2, Trid ve Dixon Price)

seçilmiştir. Önerilen kaotik optimizasyon yönteminin başarımını test edebilmek için ayrıca, literatürde sıkça kullanılan gerçek dünya problemlerinden birisi olarak bilinen “Sıkıştırılmış Yay Dizayn Problemi” üzerinde test edilmiş ve sonuçlar Çizelge 5.6’da verilmiştir. Kıyaslama ve sonuçlar, önerilen kaotik optimizasyon yönteminin başarılı olduğunu açıkça göstermiştir.

Lojistik-Sinüs Kaotik Tabanlı Optimizasyon Algoritması için kullanılan parametreler sırasıyla şu şekilde verilmiştir:

- Maksimum Döngü Sayısı = 500
- Popülasyon Sayısı = 40
- Boyut = 30
- Kaotik Çarpan = 0.86

Önerilen optimizasyon yönteminin kıyaslama fonksiyonları üzerinde elde edilen sonuçları Çizelge 5.6’da listelenmiştir.

Çizelge 5.6. Önerilen optimizasyon yönteminin kıyaslama fonksiyonları üzerinde elde ettiği sonuçlar

F	Kriter	Sonuç	F	Kriter	Sonuç	F	Kriter	Sonuç	F	Kriter	Sonuç
	Min.	0		Min.	0		Min.	8.88×10^{-16}		Min.	0
F1	Ort.	0	F5	Ort.	0	F9	Ort.	8.88×10^{-16}	F13	Ort.	1.11×10^{-16}
	S.S.	0		S.S.	0		S.S.	0		S.S.	2.03×10^{-17}
F2	Min.	0		Min.	0		Min.	0		Min.	0
	Ort.	0	F6	Ort.	0	F10	Ort.	1.40×10^{-2}	F14	Ort.	9.89×10^{-17}
	S.S.	0		S.S.	0		S.S.	2.83×10^{-2}		S.S.	2.69×10^{-16}
	Min.	0		Min.	1.22×10^{-5}		Min.	8.90×10^{-47}		Min.	0
F3	Ort.	0	F7	Ort.	1.53×10^{-4}	F11	Ort.	4.71×10^{-31}	F15	Ort.	0
	S.S.	0		S.S.	2.09×10^{-4}		S.S.	4.71×10^{-31}		S.S.	0
	Min.	0		Min.	0		Min.	5.45×10^{-48}		Min.	0
F4	Ort.	0	F8	Ort.	0	F12	Ort.	1.34×10^{-32}	F16	Ort.	0
	S.S.	0		S.S.	0		S.S.	1.34×10^{-32}		S.S.	0

Çizelge 5.6’da küresel en iyi değer 0’dır. Çizelge 5.6 incelendiğinde, önerilen algoritmanın, F1, F2, F3, F4, F5, F6, F8, F15 ve F16 için küresel en iyi sonuca ulaşırken, diğer kıyaslama fonksiyonları için de 0’a yakın sonuçlar elde ettiği görülmektedir. Bu da algoritmanın kıyaslama fonksiyonları üzerindeki başarısını göstermektedir.

Önerilen algoritma, ayrıca Sinüs-Kosinüs Algoritması (SKA) [15], Balina Optimizasyon Algoritması (BOA) [325], Çekirge Optimizasyon Algoritması (ÇOA)

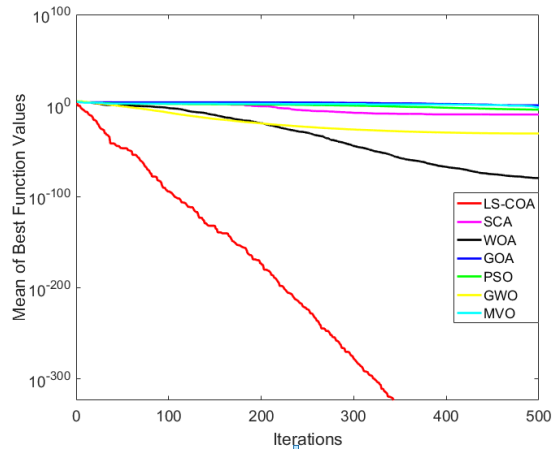
[326], Parçacık Sürü Algoritması (PSO) [170], Gri Kurt Optimizasyonu (GKO) [327], Çoklu Evren Optimizasyonu (ÇEO) [328] ve Kaotik Dinamik Ağırlıklı Parçacık Sürü Optimizasyonu (KDA-PSO) ile 12 farklı kıyaslama fonksiyonu (Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock's, Step, Noise, Rastrigin's, Ackley's, Griewank, Generalized Penalized 1 ve Generalized Penalized 2) kullanılarak karşılaştırılmıştır. Elde edilen sonuçlar Çizelge 5.7'de verilmiştir. Çizelge 5.7'de, 10^{-50} 'den daha küçük sonuçlar 0 olarak ifade edilmiştir.

Çizelge 5.7. Algoritmaların performans karşılaştırmaları

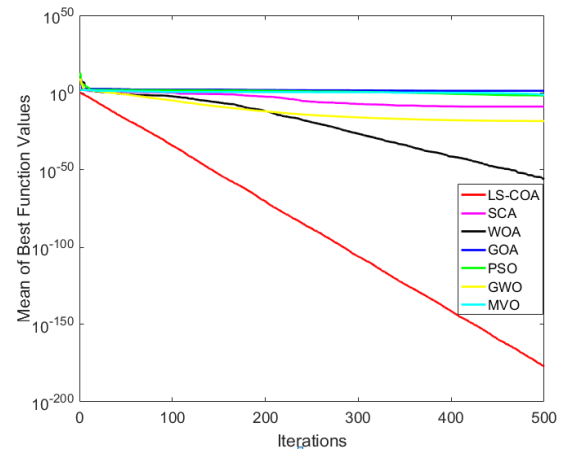
Etiket	Kriter	SKA	BOA	ÇOA	PSO	GKO	ÇEO	KDA-PSO	LS-KOA
F1	Mean	1.11×10^1	1.41×10^{-30}	1.72×10^{-8}	3.90×10^{-1}	6.59×10^{-28}	2.08×10^1	0	0
	S.D.	1.83×10^1	4.91×10^{-30}	1.77×10^{-8}	2.03×10^{-1}	6.34×10^{-5}	6.48×10^{-1}	0	0
F2	Mean	1.62×10^{-2}	1.06×10^{-21}	2.16×10^0	2.03×10^{-1}	7.18×10^{-17}	1.59×10^1	0	0
	S.D.	2.12×10^{-2}	2.93×10^{-21}	3.08×10^0	4.58×10^{-1}	2.90×10^{-2}	4.47×10^1	3.79×10^{-36}	0
F3	Mean	2.10×10^4	5.39×10^{-7}	4.80×10^{-6}	5.73×10^2	3.29×10^{-6}	4.53×10^2	0	0
	S.D.	1.12×10^4	2.93×10^{-6}	1.65×10^{-5}	2.90×10^2	7.91×10^1	1.77×10^2	0	0
F4	Mean	7.37×10^1	7.25×10^{-2}	1.33×10^{-4}	4.15×10^0	5.61×10^{-7}	3.12×10^0	0	0
	S.D.	2.11×10^1	3.97×10^{-1}	1.42×10^{-4}	9.55×10^{-1}	1.31×10^0	1.58×10^0	5.26×10^{-43}	0
F5	Mean	1.70×10^6	2.78×10^2	3.61×10^2	2.10×10^2	2.68×10^1	1.27×10^3	2.81×10^1	0
	S.D.	4.86×10^6	7.63×10^{-1}	6.73×10^2	2.04×10^2	6.99×10^1	1.47×10^3	4.82×10^1	0
F6	Mean	3.69×10^1	3.11×10^1	2.94×10^{-8}	1.05×10^0	8.16×10^{-1}	2.29×10^1	0	0
	S.D.	8.84×10^1	5.32×10^{-1}	5.68×10^{-8}	1.23×10^0	1.26×10^{-4}	6.30×10^{-1}	0	0
F7	Mean	6.65×10^{-1}	1.42×10^{-3}	1.05×10^{-1}	8.38×10^{-8}	2.21×10^{-3}	5.19×10^{-2}	0	1.53×10^{-4}
	S.D.	1.81×10^0	1.14×10^{-3}	1.86×10^{-1}	5.02×10^{-8}	1.00×10^{-1}	2.96×10^{-2}	0	2.09×10^{-4}
F8	Mean	5.91×10^1	0	8.79×10^0	2.44×10^1	3.10×10^{-1}	1.18×10^2	0	0
	S.D.	3.65×10^1	0	6.25×10^0	6.18×10^0	4.73×10^1	3.93×10^1	0	0
F9	Mean	1.15×10^1	7.40×10^1	6.03×10^{-1}	4.92×10^{-1}	1.06×10^{-13}	4.07×10^0	8.88×10^{-16}	8.88×10^{-16}
	S.D.	9.95×10^0	9.89×10^1	8.43×10^{-1}	3.21×10^{-1}	7.78×10^{-2}	5.50×10^0	8.88×10^{-16}	0
F10	Mean	1.04×10^0	2.89×10^{-4}	1.75×10^{-1}	4.96×10^{-1}	4.48×10^{-3}	9.38×10^{-1}	0	1.40×10^{-2}
	S.D.	2.94×10^{-1}	1.58×10^{-3}	1.12×10^{-1}	1.05×10^{-1}	6.65×10^{-3}	5.95×10^{-2}	0	2.83×10^{-2}
F11	Mean	4.17×10^6	3.39×10^{-1}	6.21×10^{-4}	1.22×10^{-2}	5.34×10^{-2}	2.45×10^0	2.93×10^{-1}	4.71×10^{-31}
	S.D.	1.38×10^7	2.14×10^{-1}	2.80×10^{-3}	2.83×10^{-2}	2.07×10^{-2}	7.91×10^{-1}	8.29×10^{-1}	4.71×10^{-31}
F12	Mean	1.51×10^7	1.88×10^0	1.80×10^{-3}	1.08×10^{-1}	6.54×10^{-1}	2.22×10^{-1}	2.31×10^0	1.34×10^{-32}
	S.D.	2.87×10^7	2.66×10^{-1}	4.90×10^{-3}	7.95×10^{-2}	4.47×10^{-3}	8.64×10^{-2}	4.29×10^0	1.34×10^{-32}

Çizelge 5.7 incelendiğinde LS-KOA'nın F7 (Noise) ve F10 (Griewank) kıyaslama fonksiyonları haricindeki bütün fonksiyonlar için literatürde belirtilen diğer algoritmalarından daha iyi sonuçlar verdiği görülmektedir.

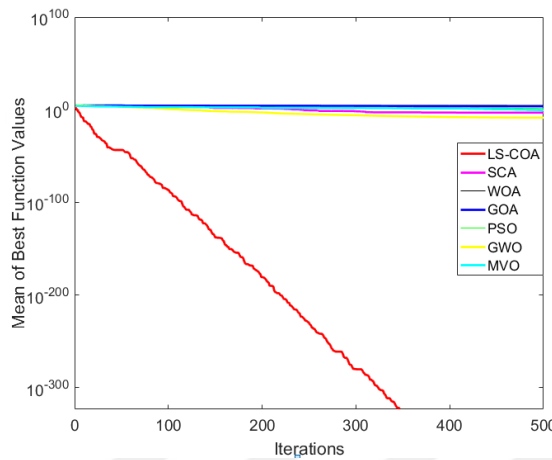
LS-KOA'da kullanılan 12 farklı kıyaslama fonksiyonuna ait minimizasyon eğrileri Şekil 5.1'de verilmiştir.



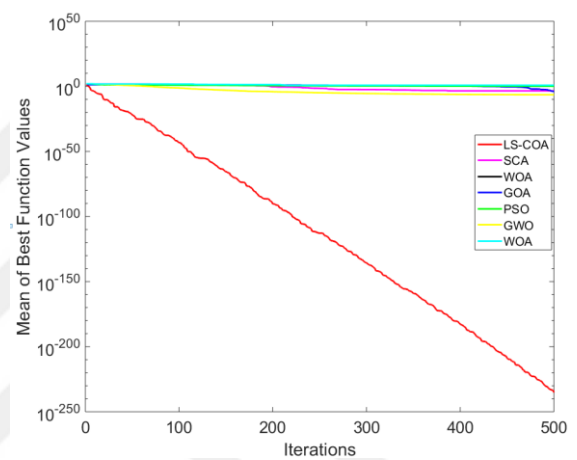
F1



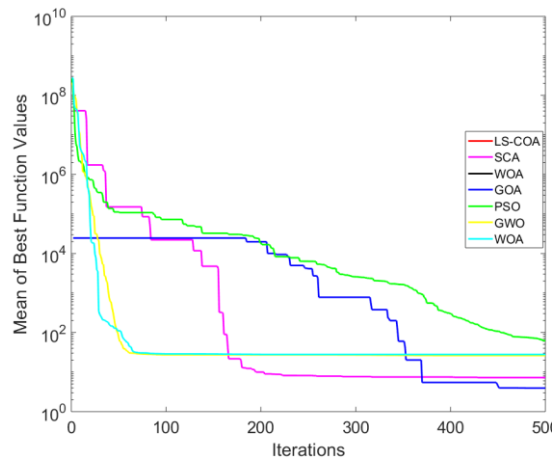
F2



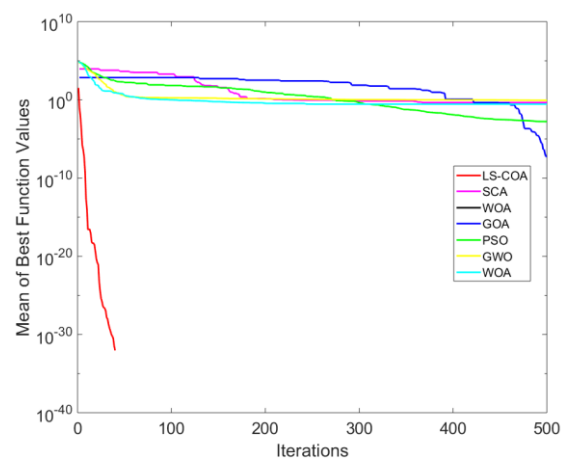
F3



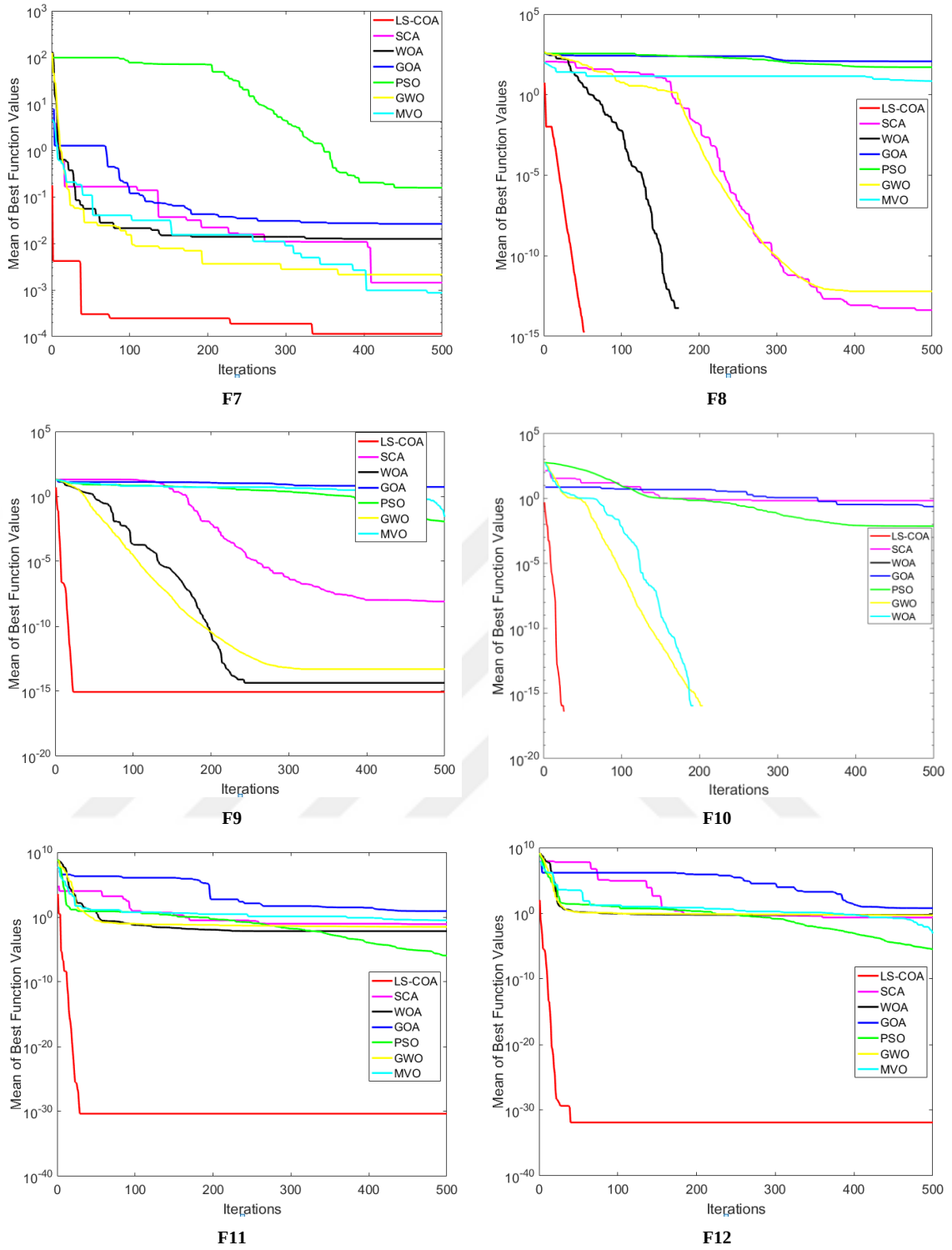
F4



F5



F6



Şekil 5.1. Kıyaslama fonksiyonlarına ait minimizasyon eğrileri

5.3.1. LS-KOA'nın sıkıştırılmış yay dizayn problemine uygulanması ve başarı analizi

Yay sistemleri pek çok farklı alanda kullanılır. Yaygın olarak kullanıldıkları için en iyi şekilde optimize edilmeleri gerekir. Yay dizaynlarında 3 farklı değişken



Şekil 5.2. Sıkıştırılmış Yay Dizayn Problemi'nin grafiksel gösterimi [10]

$$f(x_1, x_2, x_3) = (x_3 + 2)x_1^2x_2$$

$$f_1(x_1, x_2, x_3) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0 \quad (5.2)$$

$$g_3(x_1, x_2, x_3) = 1 - \frac{140.45x_1}{x_2^2 x_3} \leq 0 \quad (5.4)$$

$$x_1 \in [0.05, 2], x_2 \in [0.25, 1.3], x_3 \in [2, 15] \quad (5.6)$$

LS-KOA'nın Sıkıştırılmış Yay Dizayn Problemi üzerindeki performansını karşılaştırmalı olarak gösterebilmek için literatürde sıklıkla kullanılan Sinüs-Kosinüs Algoritması (SKA) [33], Parçacık Sürü Optimizasyonu (PSO) [170], Karınca Koloni Algoritması (KKA) [171] ve Balina Optimizasyon Algoritması (BOA) [335] tercih edilmiştir. Sıkıştırılmış Yay Dizayn Problemi'ne ait karşılaştırmalı sonuçlar Çizelge 5.8'de verilmiştir.

Çizelge 5.8. Önerilen LS-KOA'nın Sıkıştırılmış Yay Dizayn Problemi üzerinde başarımların analizi

Algoritma	Sonuç
SKA	0.012676
PSO	0.012665
KKA	0.013263
BOA	0.012680
LS-COA	0.011485

Çizelge 5.8, önerilen LS-KOA'nın seçilen algoritmalar içerisinde en iyi sonucu elde ettiğini açıkça göstermektedir.



6. KAOTİK DARCY OPTİMİZASYON ALGORİTMASININ HEPATOSELÜLER KARSİNOM ARAŞTIRMASINDA (HKA) ÖZELLİK SEÇİCİ OLARAK KULLANILMASI VE ANALİZİ

Kanser, günümüzde oldukça yaygın ve ölüm riski yüksek hastalıklardan birisidir. Primer karaciğer tümörlerinin çoğu hepatoselüler karsinomdur (Hepatocellular Carcinoma (HCC)) [336]. Bir kanser türü olan HCC, yaygın bir tümördür ve genellikle kronik karaciğer hastalığı veya sirozu olan kişilerde görülmektedir [337]. HCC, en yaygın görülen beş kanser hastalığı arasında yer almaktadır [338]. Genel olarak, karaciğer kanseri hastaları, tanıdan önce herhangi bir belirti göstermezler. Bu sebeple de daha az semptom görüldüğünden tedavi edilmesi de oldukça zordur [339]. Erken tanı bu sebeple karaciğer hastalarının iyileşmesinde çok önemli bir faktördür. Erken tanı için, serum belirteci ve madikal tıbbi görüntüleme kullanılır [340]. Kanserli doku, bilgisayarlı tomografi görüntüleri kullanılarak doğru bir şekilde tanımlanabilir. Görüntü işleme yaklaşımında, klinisyene karar verme sürecinde yardımcı olmak için karaciğer kanserini sınıflandırabilecek, bilgisayar destekli tanı kullanılabilir [324,325]. HCC, tedavi edilemediğinde yüzbinlerce insanın ölümüne neden olmaktadır [326,327,328].

Son zamanlarda, HCC ile alakalı hastalıklar hakkında çok sayıda çalışma yapılmıştır [320,322,323,325,327-330]. Książek ve ark. [347] HCC veri kümesini sınıflandırmak için saf Bayes, doğrusal diskriminant, rastgele orman, SVM, kNN, lojistik regresyon, çok katmanlı algılayıcı gibi yaygın olarak bilinen makine öğrenme yöntemlerini kullanmıştır. Bu çalışmada, %88.5'lik bir doğruluk performansı gösterilmiştir. Sawhney ve diğ. [348] HCC tanısı için, rastgele orman algoritmasını ve ateşböceği optimizasyon yöntemini birlikte kullanmıştır.

Optimizasyon algoritmaları, özellikleri ve sınıflandırıcıların en iyi parametrelerini [349] seçmek için kullanılır. Özellik seçimi, fazla ve gereksiz özellikleri kaldırarak belirli bir özellik kümesinden küçük bir alt küme seçme tekniğidir. Santos ve diğ. [346] yeni bir küme tabanlı örnekleme algoritması kullanmıştır. Çalışmalarında veri kümesini dengelemek için yeni bir teknik geliştirdiler ve lojistik regresyon ve sinir ağı yöntemlerini kullandılar.

Patoloji görüntülerinden tanı, çoğu kanser olmak üzere birçok hastalığın teşhisinde önemli bir konudur. Li ve diğ. [350], HCC çekirdeğinin doğru segmentasyonu için patoloji görüntülerinin kaba segmentasyonu için yeni bir aşırı öğrenme makinesi (Extreme Learning Machine (ELM)) tabanlı bir model önerdi. Das ve diğ. [342] bilgisayarlı karaciğer tomografi görüntülerindeki kanser lezyonunu etkili

bir şekilde tanımlamak için yoğunluk temelli segmentasyondan oluşan havza, Gauss tabanlı derin öğrenme tekniği adı verilen bilgisayar destekli bir tanı modeli önerdi. İstatistiksel bilgi, dokusal ve geometrik özellikler, HCC'nin yanı sıra hemanjiyom ve metastatik karsinom tiplerini karaciğer tümörlerini ayırt etmek için Derin Sinir Ağı kullanılarak sınıflandırıldı. Fa ve diğ. [351] derin öğrenmeye dayanan çalışmalarında, HCC hastalarında Pathifier metodolojisine dayanarak hastalığa özgü özellikler tanımlamışlardır. Chaudhary ve diğ. [352] HCC veri setleri üzerinde derin öğrenme hesaplamalı bir çerçeve kullanmışlardır. Çok kodlu entegrasyon için derin öğrenme uygulaması olarak otomatik kodlayıcı çerçevesini seçtiler. Bu algoritmaların klinik sonuçlarla ilişkili özellikler üretmek için oldukça etkili yaklaşımlar olduğu kanıtlanmıştır [336,337]. Palazzo ve diğ. [355], hastaların tümör örneklerinde gen başına basit genetik mutasyonlar kullanılarak alkolle ilişkili HCC tümör alt tipleriyle ilişkili virüsü sınıflandırmak için bir veri analizi boru hattı (paralel çalışma) kullandılar. Veri analizi yapmak için özellik seçimi ve sınıflandırma için Destek Vektör Makinelerini kullandılar. Doğan ve Türkoğlu [356], Hiperlipidemi tanısında büyük verileri analiz etmek için Destek Vektör Makinelerini kullandılar. Benzer şekilde Özyurt [357], beyaz kan hücrelerini tespit etmek için ELM yöntemini kullanmıştır. Aydemir ve arkadaşları [358], yaptıkları çalışmada epilepsi hastalığını teşhis etmek için çok seviyeli bir makine öğrenme yöntemi kullanmışlardır. Tuncer ve arkadaşları [359], Elektrokardiyografi (EKG) sinyallerini kullanarak kalp sağlığının otomatik tanısı için sinyal işleme algoritmali makine öğrenme yöntemlerini kullanmıştır. Tuncer ve Doğan, Parkinson hastalığını otomatik olarak tanımak için makine öğrenme yöntemleri kullandılar [360].

Önerilen Kaotik Darcy Optimizasyon Algoritması'na ait tanımlamalar ve matematiksel ifadeler önceki başlıklarda verilmiştir. Bitki köklerinin suya yönelimiyle alakalı doğadan esinlenilen bir optimizasyon algoritması modeli olan Kaotik Darcy Optimizasyon Algoritması, bu çalışmada özellik seçici olarak kullanılmıştır.

Literatürde, anket önemli bir veri toplama yöntemi olarak karşımıza çıkmaktadır. Fakat, anketlerde genellikle eksik veya fazla veriler bulunmaktadır. Bu sebeple, eksik özellik tamamlama ve özellik seçimi, anketlerden daha anlamı veriler çıkarmak için kullanılmaktadır. Literatürdeki bu iki sorunu çözmek için bir hipotez geliştirilmiştir. Hipotezin geliştirilmesi için bir sınıflandırma yöntemi sunulmuştur. Önerdiğimiz hipotez, istatistiksel bir moment (ortalama) ile eksik özellik tamamlama ve Kaotik Darcy Optimizasyonu kullanarak özellik seçici olarak kullanma işlemlerinden

oluşmaktadır. İlk olarak, Hepatoselüler Karsinom Araştırması'na (HKA) ortalama tabanlı bir denetimli özellik tamamlama yöntemi uygulanır. Kullanılan HKA anketi 49 özellikten oluşmaktadır. Ayırıcı özellikleri seçmek için Kaotik Darcy Optimizasyon Algoritması özellik seçici olarak kullanılır ve bu özellik seçici en iyi 31 ayırıcı özelliği seçmektedir. Kaotik Darcy Optimizasyon Algoritması'nın HKA üzerinde 0,9879 doğruluk oranında başarı elde ettiği gözlemlenmiştir.

Önerilen HKA sınıflandırıcı yöntem üç temel aşamadan oluşur. Bunlar sırasıyla; denetimli ortalama yöntem kullanılarak özellik tamamlama, Kaotik Darcy Optimizasyon Algoritması kullanılarak özellik seçme ve sınıflandırmadır. Kaotik Darcy Optimizasyon Algoritması temelli HKA sınıflandırma yönteminin diyagram gösterimi Şekil 6.1'de gösterilmektedir.



Şekil 6.1. Önerilen Kaotik Darcy Optimizasyon Algoritması temelli HKA sınıflandırma yönteminin şematik açıklaması

6.1. Eksik Özellik Tamamlama Yöntemi

Burada denetimli bir eksik özellik tamamlama yöntemi kullanıldı ve bu yöntem daha önce Tuncer ve Ertam [361] tarafından sunulmuştur. Bu tamamlama yönteminde, gözlemler, hedef değere göre parçalara ayrılır. Daha sonra, eksik olmayan gözlemlerin ortalama değeri hesaplanır, daha sonra bu değer eksik değerlere atanır. Kullanılan HCC veri kümesinde iki sınıf vardır ve bu sınıflar canlı veya ölü birey sınıflarıdır. Bu nedenle, veriler iki sınıfa ayrıldı ve iki ortalama değer hesaplandı. Bu değerler, özellikleri eksik olan canlı ve ölü sınıflar için atanmıştır.

6.2. Önerilen Kaotik Darcy Optimizasyon Algoritması Tabanlı Özellik Seçici Metodu

Bu çalışmada, özellik seçimi için Kaotik Darcy Optimizasyon Optimizasyon Algoritması kullanılmıştır. Hata parametresi, bu yöntemde bir form koruma fonksiyonu olarak kabul edilir ve hata parametresi, Denklem (6.1)'de gösterilmiştir.

$$Err = 1 - \frac{TP + TN}{TP + FN + FP + TN} \quad (6.1)$$

Burada:

Err (Error) = Hata,

TP (True Positive) = Doğru Pozitif,

FN (False Negative) = Yanlış Negatif,

FP (False Positive) = Yanlış Pozitif,

TN (True Negative) = Gerçek Negatif'tir.

KDar-OA'nın asıl amacı minimum hata ile en iyi özellikleri seçmektir. KDar-OA tabanlı özellik seçiminde temel, uygunluk ve özellik parametreleri seçim işlevleri olarak belirlenmiştir. Önerilen yöntem Şekil.6.2'deki gibi gösterilebilir.



Şekil.6.2. Önerilen KDar-DOA tabanlı özellik seçicinin özeti.

Önerilen KDar-OA tabanlı özellik seçicinin adımları aşağıda verilmiştir.

1.Adım: HCC veri kümesi yüklenir.

2.Adım: Başlangıç parametreleri ayarlanır.

$$UB = 2^{49} - 1 \quad (6.2)$$

$$LB = 1 \quad (6.3)$$

$$p_{num} = 250 \quad (6.4)$$

$$niter = 1000 \quad (6.5)$$

Denklem (6.2-6.5)'te KDar-OA parametreleri verilmiştir. niter maksimum yineleme sayısıdır. T vektörünün ilk değeri rastgele atanır.

3.Adım: Denklem 4.kullanılarak başlangıç değerleri oluşturulur.

4.Adım: Her bir parçacık ikili sayı sistemine çevrilir. Bu adımda her bir parçacık 49 bit olarak kodlanmıştır.

5.Adım: Parçacıklar kullanılarak özellikler seçilir. Özellik seçme prosedürü Şekil 6.3'te verilmiştir.

6.Adım: Fitness fonksiyonu kullanılarak her bir parçacığın hata değerini hesaplanır.

$$predict = KNN(X_{particle_i}^S, y) \quad (6.6)$$

Burada $X_{particle_i}^S$, tahmin edilen değer, KNN en yakın komşu sınıflandırıcıdır (k=1, distance=city block (KNN uzaklık metriği), 10 fold CV (10 fold cross validation - test eğitim stratejisi) ve y değeri hedefdir (gerçek çıktı). Tahmin (predict) ve y kullanılarak, TP, FP, FN ve TN değerleri hesaplanır. Denklem (6.6), bir hata elde etmek için hesaplanan TP, FP, FN ve TN'yi uygular.

7.Adım: Önerilen KDar-OA'nın 2-6 adımları kullanılır.

Procedure: <i>FeatSelect</i> (<i>X</i> , <i>particle_i</i>)
Giriş: 165 x 49 boyutlu tamamlanmış HCC veri kümesi özellikleri (X), 49 boyutunda ikili değer parçacığı (particle _i).
Çıkış: 165 x 49 boyutunda seçilen özellikler ($X_{particle_i}^S$).
<pre> 1: for i=1 to 165 do 2: <i>k</i> = 1; 3: for j=1 to 49 do 4: if <i>particle_i</i>(<i>j</i>) = 1 then 5: $X_{particle_i}^S(i, k) = X(i, j);$ 6: <i>k</i> = <i>k</i> + 1; 7: end if 8: end for j 9: end for i </pre>

Şekil 6.3. Parçacığın ikili değerinin kullanarak özellik seçme prosedürü

6.3. Sınıflandırıcı

Seçilen özelliklerin KDar-OA ile kıyasını gösterebilmek için, makine öğrenmesinde en fazla kullanılan 9 farklı sınıflandırıcı seçilmiştir. Bu sınıflandırıcılar; Karar Ağacı (Decision Tree (DT)) [362], Doğrusal Diskriminant (Linear Discriminant (LD)) [363], Karesel Diskriminant (Quadratic Discriminant (QD)) [364], Mantıksal Regresyon (Logistic Regression (LR)) [365], Destek Vektör Makinesi (Support Vector Machine (SVM)) [366], En Yakın Komşuluk (K-Nearest Neighbors (KNN)) [367], Torbalı Ağaç (Bagged Tree (BT)) [368], Alt Uzak Ayırıcı (Subspace Discriminant (SD)) [369] ve Alt Uzak En Yakın Komşuluğu (Subspace Nearest Neighbor (SNN)) [370] 'dur. İlk altı sınıflandırıcı geleneksel sınıflandırıcıdır ve son üç sınıflandırıcı topluluk sınıflandırıcısıdır. Bu sınıflandırıcıların özellikleri Çizelge 6.1'de listelenmiştir.

Çizelge 6.1. Kullanılan dokuz makine öğrenme algoritmasının açıklaması

No	Sınıflandırıcı	Sınıflandırıcının özellikleri
1	DT	Gini'nin kriteri ve maksimum bölünme sayısı 100 olarak seçildi.
2	LD	Kovaryans yapısı dolu.
3	QD	Kovaryans yapısı dolu.
4	LR	Bu sınıflandırıcının gelişmiş bir seçeneği yoktur. Parametrik değildir.
5	SVM	Doğrusal kernel işlevi SVM kullanılır. 1, kutu sınırlama seviyesi olarak atanmıştır.
6	KNN	K ve uzaklık metriği sırasıyla bir ve şehir bloğu olarak seçilir.
7	BT	Öğrenenlerin sayısı ve bölünmeler sırasıyla 30 ve 164 olarak seçilmiştir.
8	SD	Öğrenci sayısı ve alt uzak boyutu sırasıyla 30 ve 16 olarak seçilmiştir. Öğrenme hızı olarak 0.1 seçilmiştir.
9	SNN	Öğrenen sayısı ve alt uzak boyutu sırasıyla 30 ve 16 olarak seçilmiştir. Öğrenme oranı olarak 0.1 seçilmiştir.

6.4. Sonuçlar

Önerilen KDar-OA tabanlı en iyi özellik seçim yönteminin sonuçları bu bölümde verilmiştir.

6.4.1. Veri kümesi

Bu çalışmada veri kümesi olarak literatürdeki çalışmalarda da sıklıkla kullanılan ve HCC hastalarının sağlık verilerini kapsayan bir anket kullanılmış ve bu anket 49 sorudan oluşmaktadır. Bu anket, 165 deneğe uygulanmış ve elde edilen cevaplar özellik olarak kullanılmıştır. Bu araştırmanın temel amacı, hayatta kalan veya ölü deneklerin bu özellikleri kullanarak tespit edilmesidir. Ancak, bu veri kümesinde eksik özellikler (yanıtlanmamış sorular) bulunmaktadır. Bu veri kümesine <https://archive.ics.uci.edu/ml/datasets/HCC+Survival> URL (referans ver) adresinden ulaşılabilir. Bu veri kümesine göre, 102 kişi hayatta kalmış ve 63 kişi ise vefat etmiştir. Kullanılan veri setinin detayları Şekil 6.4'te gösterilmektedir.

Çizelge 6.2. Kullanılan HCC veri kümesinin ayrıntıları

NO	Özellik (Feature)	Aralık (Range)	Eksik veri miktarı (%)	NO	Özellik (Feature)	Aralık (Range)	Eksik veri miktarı
1	Gender	1,0	0	26	Pack of cigarettes (per year)	0-510	32
2	Symptoms	1,0	11	27	Performance	0-4	0
3	Alcohol	1,0	0	28	Encephalopathy degree	1-3	1
4	Hepatitis B(HBsAG)	1,0	10	29	Ascites degree	1-3	1
5	Hepatitis B(HBeAg)	1,0	24	30	Int. Normalized Ratio	0,84-4,82	2
6	Hepatitis B(HBcAb)	1,0	15	31	AFP	1,2-1,810	5
7	Hepatitis C(HCVAb)	1,0	5	32	Hemoglobin	5-18,7	2
8	Cirrhosis	1,0	0	33	Corpuscular Volume	69,5-119,6	2

9	End. Countries	1,0	24	34	Leukocytes	2,2-13	2
10	Smoke	1,0	25	35	Platelets	1,71-459	2
11	Diabetes	1,0	2	36	Albumin	1,9-4,9	4
12	Obesity	1,0	6	37	Total Bilirubin	0,3-40,5	3
13	Hemochromatosis	1,0	14	38	ALT	11-420	2
14	Hypertension (AHT)	1,0	2	39	AST	17-553	2
15	Renal Insufficiency	1,0	1	40	GGY-T	23-1575	2
16	Virus (HIV)	1,0	8	41	ALP	1,28-980	2
17	NASH	1,0	13	42	TP	3,9-102	7
18	Esophageal Varices	1,0	32	43	Creatinine	0,2-7,6	4
19	Splenomegaly	1,0	9	44	Nodules (number)	0-5	1
20	Hypertension	1,0	7	45	Nodules (majör dimension)	1,5-22	12
21	Vein Thrombosis	1,0	2	46	Bilirubin	0,1-29,3	27
22	Metastasis	1,0	2	47	Iron	0-224	48
23	Hallmark	1,0	1	48	Oxygen Saturation	0-126	48
24	Age at diagnosis	20-93	0	49	Ferritin	0-2230	48
25	Grams of Alcohol (per day)	0-500	29	Target	Class	0,1	2

Önerilen KDar-OA'nın başarısını göstermek için 12 sayısal kıyaslama fonksiyonu kullanılmıştır. Bu fonksiyonlara ait tanımlar Çizelge 6.2'de verilmiştir. Bu çalışmada, Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock's, Step,

Noise, Rastrigin's, Ackley's, Griewank, Generalized Penalized 1, Generalized Penalized 2, olmak üzere 12 farklı kıyaslama fonksiyonu kullanılmıştır.

Bu fonksiyonlar, literatürde sıklıkla kullanılan Sinüs-Kosinüs Algoritması (SKA) [33,354], Balina Optimizasyon Algoritması (BOA) [325], Çekirge Optimizasyon Algoritması (ÇOA) [326], Parçacık Sürü Algoritması (PSO) [170], Gri Kurt Optimizasyonu (GKO) [311,355], Çoklu Evren Optimizasyonu (ÇEO) [328] ve Kaotik Dinamik Ağırlıklı Parçacık Sürü Optimizasyonu (KDA-PSO) ile karşılaştırılmış ve sonuçlar Çizelge 5.5'te verilmiştir.

Önerilen KDar-OA, tanımlanan HCC veri kümesine uygulanmış ve sonuçlar elde edilmiştir. Önerilen kaotik özellik seçici yöntem uygulandıktan sonra 31 özellik belirlenmiştir. Bu özellikler küresel optimumun ikili değerleri kullanılarak seçilir ve bu değerler "1101111001100010110001101011010110111110100111111" dir. Şekil 6.3'e göre, 0 değerleri gereksiz özellik olarak tanımlanır. Optimum ikili değerler kullanılarak seçilen özellikler, seçilen dokuz sınıflandırıcının girişi olarak kullanılır. Önerilen yöntem MATLAB2018a programlama ortamında uygulanmıştır. Hassasiyet, hatırlama, F1 skoru, doğruluk ve hata metrikleri seçilir. Bunların matematiksel açıklamaları aşağıda verilmiştir.

$$precision = \frac{TP}{FP + TP} \quad (6.7)$$

$$recall = \frac{TP}{FN + TP} \quad (6.8)$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (6.9)$$

$$accuracy = 1 - error \quad (6.10)$$

Kullanılan dokuz sınıflandırıcının sonuçlarını elde etmek için, her sınıflandırıcının testi 100 kez çalıştırıldı. Elde edilen sonuçlar Çizelge 6.3'te listelenmiştir.

Çizelge 6.3. Kullanılan dokuz sınıflandırıcının sonuçları (%)

Sınıflandırıcı	Doğruluk	Hatırlama	Hassasiyet	F1 Skoru	Hata
DT	95.15	94.86	95.47	94.86	4.85
LD	93.94	92.37	95.00	93.66	6.06
QD	90.91	90.64	90.28	90.40	9.09
LR	90.91	90.24	90.47	90.35	9.09

SVM	95.15	93.65	96.36	94.99	4.85
KNN	95.15	94.86	95.47	94.86	4.85
BT	97.58	96.83	98.11	97.47	2.42
SD	92.73	90.48	94.74	92.56	7.27
SNN	98.79	98.41	99.04	98.72	1.21

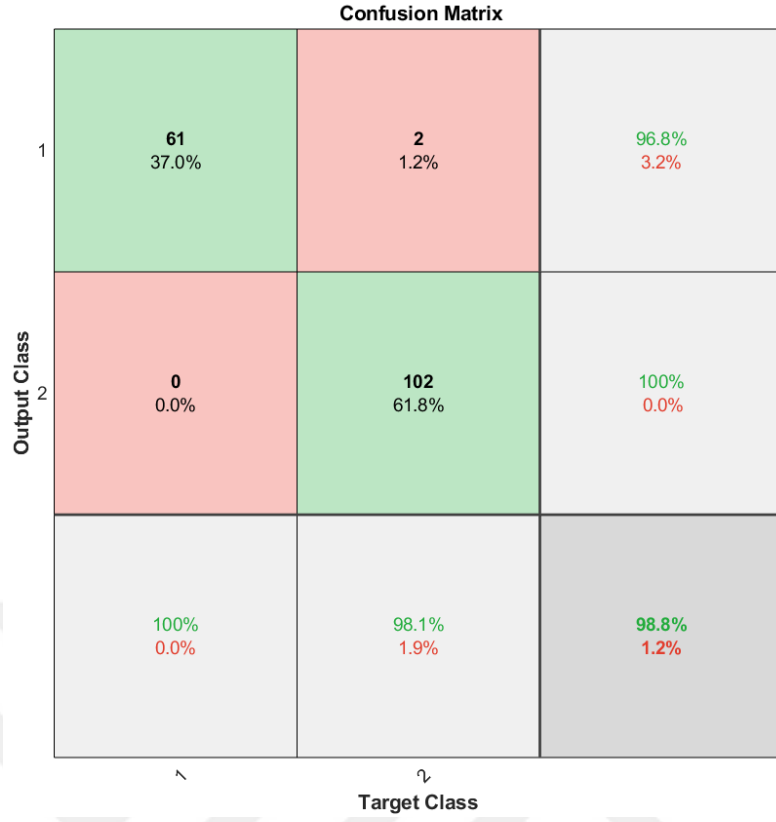
Çizelge 6.3, HCC sınıflandırması için en iyi sınıflandırıcının SNN olduğunu göstermektedir, Bütün sınıflandırıcılar %90 üzerinde sınıflandırma oranı elde etmiştir. SNN %98.79 oranı ile en yüksek başarımlı sonucunu vermiştir.

Önerilen yöntemin başarısını test etmek için, bu yöntem, literatürdeki diğer sınıflandırma yöntemleriyle karşılaştırıldı ve elde edilen karşılaştırmalı sonuçlar Çizelge 6.4'te gösterilmiştir.

Çizelge 6.4. Önerilen KDar-OA yönteminin ve diğer HCC sınıflandırma yöntemlerinin doğrulukları ve F1 Skoru puanları (%)

Sınıflandırıcı	Doğruluk	Hatırlama
Santos ve diğ.'nin metodu [346]	75.19 ± 1.05	66.50 ± 1.82
Sawhney ve diğ.'nin metodu [348]	83.50	-
Książek ve diğ.'nin metodu [347]	88.49	87.62
Tuncer ve Ertam'ın metodu [361]	92.12	91.61
Önerilen KDar-OA tabanlı metot (SNN tabanlı metot)	98.79	98.72

Çizelge 6.4 incelendiğinde önerilen KDar-OA literatürdeki diğer yöntemlerden daha başarılı sonuçlar ortaya koyduğu gözlemlenmiştir. KDar-OA, DT, SVM, KNN, BT, SD ve SNN yönteminde diğer yöntemlerden daha iyi sonuç vermiştir. SNN tabanlı yöntemde ise sadece 2 gözlem yanlış tahmin edilmiştir. En iyi sonuca ait karışıklık matrisi Şekil 6.5'te gösterilmektedir.



Şekil.6.4. KDar-OA tabanlı yöntemle ait karışıklık matrisi ve SNN sonuçları

7. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, kaotik haritalar ve optimizasyon algoritmaları incelenmiş ve kaotik tabanlı ve kaotik kök gelişim tabanlı yeni nesil optimizasyon algoritmaları geliştirilmiştir. Geliştirilen optimizasyon algoritmaları Çizelge 5.1’de bulunan 16 farklı sayısal kıyaslama fonksiyonu üzerinde test edilmiştir. Önerilen kaotik tabanlı optimizasyon algoritmaları sırasıyla, Lojistik-Gauss Hibrit Kaotik Harita Tabanlı Optimizasyon Yöntemi (LG-KOA), Lojistik-Singer Hibrit Kaotik Harita Tabanlı Optimizasyon Algoritması (LS-KOA), Kaotik Darcy Optimizasyon Algoritması (KDar-OA), Kaotik Dalton Optimizasyon Algoritması (KDal-OA) ve Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Optimizasyon Algoritması (LSinüs-KOA)’dır.

Önerilen bütün optimizasyon algoritmaları, rastgele sayı üretiminde kaotik haritalardan faydalanmaktadır. Tek boyutlu kaotik haritaların daha iyi dağılım sağlayabilmesi için hibrit kaotik haritalar önerilmiş (Lojistik-Gauss, Lojistik-Singer, Lojistik-Sinüs) ve bu yöntemler algoritmalarda kullanılmıştır. Kaotik haritaların kullanılması yeni nesil optimizasyon algoritmalarının da başarısını artırdığı görülmüştür.

Geliştirilen LSinüs-KOA yöntemi, gerçek hayat problemlerinden Sıkıştırılmış Yay Dizayn Problemi üzerinde uygulanmış ve sonuçları Çizelge 5.8’de verilmiştir. Ayrıca KDar-OA yöntemi HCC hastalarıyla ilgili araştırmada özellik seçici olarak kullanılmış ve başarılı sonuçlar elde edilmiştir. Önerilen yöntem aynı zamanda literatürdeki diğer çalışmalarla karşılaştırılmış ve hepsinden daha iyi sonuçlar aldığını gösteren sonuçlar Çizelge 6.2 ve Çizelge 6.4’te verilmiştir.

Yeni nesil kaotik tabanlı optimizasyon algoritmalarına ve kök gelişim algoritmalarına ait önerilen yöntemlerin sonuçları aşağıda verilmiştir.

7.1. Sonuçların Değerlendirilmesi

Beşinci bölümde, önerilen LG-KOA, LS-KOA, KDar-OA, KDal-OA ve LSinüs-KOA yöntemlerine ait sonuçların değerlendirmesi maddeler halinde aşağıda verilmiştir:

- 5.1.bölümde, Lojistik-Gauss ve Lojistik-Singer Hibrit Kaotik Haritalı Optimizasyon Yöntemleri önerilmiştir. Bu yöntemleri test etmek için Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock’s ve Step kıyaslama fonksiyonları kullanılmıştır. Kaotik Bio-Coğrafya Tabanlı Optimizasyon Algoritması (KBTOA), Kaotik Kril Sürüsü

Optimizasyonu (KKSOA) ve Kaotik Yerçekimi Arama Optimizasyonu Algoritması (KYAOA) isimli sürü tabanlı optimizasyon yöntemleriyle de karşılaştırılmıştır. Elde edilen karşılaştırmalı sonuçlar Çizelge 5.2’de gösterilmiştir. Sonuçlar incelendiğinde, LG-KOA’nın F5 ve F6’da küresel en iyi sonucu bulduğu görülmektedir. LS-KOA ise altı sayısal kıyaslama fonksiyonunun tamamında küresel en iyi sonucu bulmuştur. LG-KOA, ayrıca F1, F2, F3 ve F4 için KBTOA, KKSOA ve KYAOA’dan daha iyi sonuçlar vermiştir.

- 5.2.bölümde, Kaotik Darcy ve Kaotik Dalton Optimizasyon Algoritmaları isminde iki yeni kök gelişim algoritması geliştirilmiştir. Bu yeni algoritmaları test etmek için Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock’s, Step, Noise, Rastrigin’s, Ackley’s, Griewank, Generalized Penalized I ve Generalized Penalized II sayısal kıyaslama fonksiyonları kullanılmıştır. Algoritmaların performansını test etmek için tek boyutlu kaotik haritalardan, Lojistik, İteratif, Gauss, Çember, Sinüs, Çadır ve Singer haritaları kullanılmıştır. Darcy tabanlı yöntemin tek boyutlu kaotik haritalar üzerindeki sonuçlar Çizelge 5.3’te, Dalton tabanlı yöntemin tek boyutlu kaotik haritalar üzerindeki sonuçlar Çizelge 5.4’te verilmiştir. Çizelge 5.3 ve Çizelge 5.4’te 1×10^{-100} ’den küçük sonuçlar 0 olarak kabul edilir. Sonuçlar incelendiğinde, Darcy tabanlı yöntem, Lojistik Harita’da F1, F2, F3, F4, F5, F6 ve F8’de, İteratif Harita’da F1, F2, F3, F4, F5 ve F8’de, Gauss Harita’da F2, F5 ve F8’de, Çember Harita’da F1, F2, F3, F4, F5, F6 ve F8’de, Sinüs Harita’da F1, F2, F3, F4, F5, F6 ve F8’de, Çadır Harita’da F1, F2, F3, F4, F5, F6 ve F8’de, Singer Harita’da F1, F2, F3, F4, F5, F6, F7 ve F8’de en iyi sonucu elde etmiştir. Dalton tabanlı yöntem, Lojistik Harita’da F1, F2, F3, F4, F5 ve F8’de, İteratif Harita’da F1, F2, F3, F4, F5, F8 ve F10’da, Gauss Harita’da F5’te, Çember Harita’da F1, F2, F3, F4, F5 ve F8’de, Sinüs Harita’da F1, F3, F4, F5 ve F8’de, Çadır Harita’da F1, F2, F3, F4, F5 ve F8’de, Singer Harita’da F1, F2, F3, F4, F5, F6, F7 ve F8’de en iyi sonucu elde etmiştir. Önerilen KDal-OA ve KDar-OA yöntemleri, literatürde sıklıkla kullanılan Sinüs-Kosinüs Algoritması (SKA), Balina Optimizasyon Algoritması (BOA), Çekirge Optimizasyon Algoritması (ÇOA), Parçacık Sürü Algoritması (PSO), Gri

Kurt Optimizasyonu (GKO), Çoklu Evren Optimizasyonu (ÇEO) ve Kaotik Dinamik Ağırlıklı Parçacık Sürü Optimizasyonu (KDA-PSO) ile karşılaştırılmış ve sonuçlar Çizelge 5.5'te verilmiştir. Çizelge 5.5 incelendiğinde KDa1-OA'nın F6, F7, F11 ve F12, KDa2-OA'nın ise F10 hariç bütün fonksiyonlarda en iyi sonucu bulduğu görülmektedir.

- 5.3.bölümde, Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Algoritması önerilmiştir. Bu algorithmada, Sphere, Schwefel 2.22, Schwefel 1.2, Schwefel 2.21, Rosenbrock's, Step, Noise, Rastrigin's, Ackley's, Griewank, Generalized Penalized 1 ve Generalized Penalized 2, Alphine 1, Alphine 2, Trid ve Dixon Price olarak 16 adet sayısal kıyaslama fonksiyonu kullanılmıştır. Sayısal kıyaslama fonksiyonları üzerindeki sonuçlar Çizelge 5.6'da gösterilmiştir. Çizelge 5.6 incelendiğinde, önerilen algoritmanın F1, F2, F3, F4, F5, F6, F8, F15 ve F16 için en optimum sonuçları verdiği görülmüştür. Önerilen algoritma, ayrıca Sinüs-Kosinüs Algoritması (SKA), Balina Optimizasyon Algoritması (BOA), Çekirge Optimizasyon Algoritması (ÇOA), Parçacık Sürü Algoritması (PSO), Gri Kurt Optimizasyonu (GKO), Çoklu Evren Optimizasyonu (ÇEO) ve Kaotik Dinamik Ağırlıklı Parçacık Sürü Optimizasyonu (KDA-PSO) kullanılarak karşılaştırılmıştır. Elde edilen sonuçlar Çizelge 5.7'de verilmiştir. Çizelge 5.7'de, 10^{-50} 'den daha küçük sonuçlar 0 olarak ifade edilmiştir. Çizelge 5.7'ye göre önerilen algoritma, F7 ve F10 haricinde tüm fonksiyonlarda diğer algoritmalarından daha iyi sonuçlar elde etmiştir.
- 5.3.1.bölümde, Lojistik-Sinüs Hibrit Kaotik Harita Tabanlı Yeni Bir Optimizasyon Algoritması'nın, sıkıştırılmış yay dizayn problemi üzerindeki performansı ölçülmüştür. Önerilen algoritmanın performans ölçümü için literatürde sıklıkla kullanılan Sinüs-Kosinüs Algoritması (SKA), Parçacık Sürü Optimizasyonu (PSO), Karınca Koloni Algoritması (KKA) ve Balina Optimizasyon Algoritması (BOA) tercih edilmiştir. DeneySEL sonuçlar Çizelge 5.8'de verilmiş ve L Sinüs-KOA'nın diğer algoritmalarından daha iyi sonuç verdiği gözlemlenmiştir.
- 6.bölümde, Kaotik Darcy Optimizasyon Algoritması, Hepatoselüler Karsinom Araştırması'nda (HKA) özellik seçici olarak kullanılmıştır. Bu çalışmada, denetimli bir eksik özellik tamamlama yöntemi kullanıldı.

Ardından da literatürde sıklıkla kullanılan 9 farklı sınıflandırıcı seçilmiş (DT, LD, QD, LR, SVM, KNN, BT, SD ve SNN) ve Doğruluk, Hatırlama, Hassasiyet, F1 Skoru ve Hata parametreleri için sonuçlar hesaplanmış ve sonuçlar Çizelge 6.3'te gösterilmiştir. En düşük hata değerini SNN sınıflandırıcısı sağlamıştır. Son olarak önerilen algoritma, Santos ve diğ., Sawhney ve diğ., Książek ve diğ. Ve Tuncer ve Ertam'ın çalışmaları ile karşılaştırılmıştır. Elde edilen sonuçlar Çizelge 6.4'te gösterilmiştir ve 98.79 doğruluk oranı ile en iyi sonucu KDar-OA yöntemi sağlamıştır.

7.2. Öneriler

Tez çalışması sırasında gerçekleştirilen çalışmalar ve elde edilen bulgular dikkate alındığında ileride yapılması düşünülen çalışmalar şunlardır:

- Geliştirilen optimizasyon algoritmaları, daha fazla gerçek dünya problemi üzerinde denenerek, performans analizi daha da geliştirilebilir.
- Önerilen yöntemler, hız parametresi için kaotik haritaların kullanılmasının performansı artırdığını göstermiştir. Bu sayede bundan sonra geliştirilecek olan optimizasyon algoritmaları için yeni bir metot geliştirilmiştir.
- Kaotik tabanlı algoritmaların hibrit modellerle tasarlanmasının başarımı artırdığı gözlemlenmiştir. Bundan sonra geliştirilecek olan algoritmalar için bu metodun daha iyi sonuçlar verdiği dikkate alınacaktır.
- Literatüre kaotik tabanlı yeni nesil kök gelişim algoritmaları geliştirilerek, katkıda bulunmuştur. Bu algoritmalar ileride daha fazla problem üzerinde uygulanabilir.
- Geliştirilen optimizasyon yöntemleri, literatürdeki diğer veri kümeleri üzerinde de özellik seçici olarak kullanılabilir.
- Literatürdeki benzer arama algoritmaları veya sezgisel algoritmalar ile hibrit çalışmalar geliştirilerek kısıtlı, sürekli, ikili ve çok amaçlı optimizasyon problemlerini çözebilmek için kullanılabilir.

8. KAYNAKLAR

- [1] E. Alba, *Parallel Metaheuristics: A New Class of Algorithms*, 2005 .
- [2] M. C.-İ. Ü. F. B. E. D. Tezi and undefined 2015, *Cırcır böceği algoritması: Yeni bir meta-sezgisel yaklaşım ve uygulamaları*, .
- [3] R. V. Rao and V. Patel, *An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems*, **Sci. Iran.**, 20:3 (2013) 710–720.
- [4] D. Metin Türkay Koç Üniversitesi, E. Mühendisliği Bölümü, and R. Yolu, *OPTİMİZASYON MODELLERİ VE ÇÖZÜM METODLARI*, .
- [5] W. L.-T. Q. J. of Economics and undefined 1933, *The use of indifference curves in the analysis of foreign trade*, .
- [6] L. K.-M. Science and undefined 1939, *The mathematical method of production planning and organization*, .
- [7] X. Y.-M. in water, geotechnical and transport, and undefined 2013, *Optimization and metaheuristic algorithms in engineering*, .
- [8] T. Üniversitesi, M. Fakültesi, and B. Mühendisliği Bölümü, *Güncel Sürü Zekâsı Optimizasyon Algoritmaları Sinem AKYOL ve Bilal ALATAŞ*, **Nevşehir Üniversitesi Fen Bilim. Enstitü Derg.**, 1 (2012).
- [9] J. Deuri and S. S. Sathya, *Cricket chirping algorithm: An efficient meta-heuristic for numerical function optimisation*, **Int. J. Comput. Sci. Eng.**, 16:2 (2018) 162–172.
- [10] S. Mirjalili, A. L.-A. in engineering software, and undefined 2016, *The whale optimization algorithm*, .
- [11] F. Özkaynak, *A novel method to improve the performance of chaos based evolutionary algorithms*, **Optik (Stuttg.)**, 126:24 (2015) 5434–5438.
- [12] S. Dogan, *A reversible data hiding scheme based on graph neighbourhood degree*, **J. Exp. Theor. Artif. Intell.**, 29:4 (2017) 741–753.
- [13] G. O. Koca, S. Dogan, and H. Yilmaz, *A multi-objective route planning model based on genetic algorithm for cuboid surfaces*, 59:1 (2018) 120–130.
- [14] J. Niu, W. Zhong, Y. Liang, N. Luo, F. Q.-K.-B. Systems, and undefined 2015, *Fruit fly optimization algorithm based on differential evolution and its application on gasification process operation optimization*, .
- [15] M. Dorigo and M. Birattari, *Ant colony optimization*. *Encyclopedia of machine learning*, (2010).
- [16] D. Karaboga and B. Basturk, *A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm*, 39 (2007) 459–471.
- [17] X. Yang, & S. D.-2009 W. congress on nature, and undefined 2009, *Cuckoo search via Lévy flights*, .
- [18] X. S. Yang, *Firefly algorithm, stochastic test functions and design optimization*, **Int. J. Bio-Inspired Comput.**, 2:2 (2010) 78–84.
- [19] X. S. Yang, *A new metaheuristic Bat-inspired Algorithm*, *Studies in Computational Intelligence*, (2010) pp.65–74.

- [20] S.-C. Chu, P. Tsai, and J.-S. Pan, Cat Swarm Optimization. pp. 854–858, 2006.
- [21] X. S. Yang, *Flower pollination algorithm for global optimization*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2012) pp.240–249.
- [22] A. Mehrabian, C. L.-E. informatics, and undefined 2006, *A novel numerical optimization algorithm inspired from weed colonization*, .
- [23] X. Kong, Y. L. Chen, W. Xie, and X. Wu, *A novel paddy field algorithm based on pattern search method*, 2012 IEEE International Conference on Information and Automation, ICIA 2012, (2012) pp.686–690.
- [24] X. Qi, Y. Zhu, H. Chen, D. Zhang, and B. Niu, *An idea based on plant root growth for numerical optimization*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2013) pp.571–578.
- [25] Z. Zhao, Z. Cui, J. Zeng, X. Y.-2011 second international, and undefined 2011, *Artificial plant optimization algorithm for constrained optimization problems*, .
- [26] H. M.-C. and E. in Agriculture and undefined 2000, *Finite element inverse analysis using a photosynthetic algorithm*, .
- [27] W. Cai, W. Yang, X. C.-2008 I. conference on, and undefined 2008, *A global optimization algorithm based on plant growth theory: plant growth optimization*, .
- [28] M. Demir, A. Karcı, V. Mehmet“ozdemir, and M. Mehmet“ozdemir, *Fidan Geli,simGeli,sim Algoritması Yardımı ile DNA Motiflerinin Ke,sfiKe,sfi*, **Univ. J. Sci. Eng.**, 8:1 (2011).
- [29] H. Zhang, Y. Zhu, and H. Chen, *Root growth model: A novel approach to numerical function optimization and simulation of plant root system*, **Soft Comput.**, 18:3 (2014) 521–537.
- [30] S. Akyol and · Bilal Alatas, *Plant intelligence based metaheuristic optimization algorithms*, **Artif Intell Rev**, 47 (2017) 417–462.
- [31] M. A. Boschetti, V. Maniezzo, M. Roffilli, and A. Bolufé Röhlér, *Matheuristics: Optimization, simulation and control*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2009) pp.171–177.
- [32] F. Ozbay, ... B. A.-C. on E. T. and A., and undefined 2016, *Review of musics based computational intelligence algorithms*, .
- [33] S. M.-K. systems and undefined 2016, *SCA: a sine cosine algorithm for solving optimization problems*, .
- [34] R. Vohra and B. Patel, *AN EFFICIENT CHAOS-BASED OPTIMIZATION ALGORITHM APPROACH FOR CRYPTOGRAPHY*, **Int. J. Commun. Netw. Secur.**, :1 (2012).
- [35] A. Kumar and & M. K. Ghose, *Overview of Information Security Using Genetic Algorithm and Chaos*, *Information Security Journal: A Global Perspective*, **Inf. Secur. J. A Glob. Perspect.**, 18:6 (2009) 0–0.
- [36] M. Geetha and K. Akila, *SURVEY: CRYPTOGRAPHY OPTIMIZATION ALGORITHMS*, .

- [37] E. Preprint, B. Ibrahimi, Z. Dika, and A. Luma, *Optimization Algorithms in Visual Cryptography*, (2020).
- [38] W. Chen, D. Tan, Q. Yang, ... T. G.-I. T. on, and undefined 2019, *Ant colony optimization for the control of pollutant spreading on social networks*, .
- [39] K. Rizman Žalik, Evolution Algorithm for Community Detection in Social Networks Using Node Centrality. pp. 73–87, 2019.
- [40] L. Jain, R. Katarya, S. S.-E. S. with Applications, and undefined 2020, *Opinion leader detection using whale optimization algorithm in online social network*, .
- [41] I. Messaoudi and N. Kamel, *A multi-objective bat algorithm for community detection on dynamic social networks*, .
- [42] F. Zhan, *How to Optimize Social Network Influence*, .
- [43] F. B. Demir, T. Tuncer, A. F. Kocamaz, and F. Ertam, *A survival classification method for hepatocellular carcinoma patients with chaotic Darcy optimization method based feature selection*, **Med. Hypotheses**, 139 (2020).
- [44] H. Q. Nguyen, H.-B. Ly, V. Q. Tran, T.-A. Nguyen, T.-T. Le, and B. T. Pham, *materials Optimization of Artificial Intelligence System by Evolutionary Algorithm for Prediction of Axial Capacity of Rectangular Concrete Filled Steel Tubes under Compression*, .
- [45] Z. Ye, F. Wang, R. K.-2020 I. 15th International, and undefined 2020, *Image Enhancement based on Whale Optimization Algorithm*, .
- [46] M. A. El Aziz, A. A. Ewees, A. E. Hassanien, M. Mudhsh, and S. Xiong, Multi-objective whale optimization algorithm for multilevel thresholding segmentation, *Studies in Computational Intelligence*, 730. Springer Verlag, pp. 23–39, 2018.
- [47] E. Tuba, I. Strumberger, ... N. B.-2019 I. C., and undefined 2019, *Brain Storm Optimization Algorithm for Thermal Image Fusion using DCT Coefficients*, .
- [48] G. Ding, F. Dong, H. Z.-A. S. Computing, and undefined 2019, *Fruit fly optimization algorithm based on a hybrid adaptive-cooperative learning and its application in multilevel image thresholding*, .
- [49] X. Li and J. Du, *Performance optimization algorithm of radar signal processing system*, **Cluster Comput.**, 20 (2017) 359–370.
- [50] C. Yue *et al.*, *A novel multiobjective optimization algorithm for sparse signal reconstruction*, .
- [51] Y. Zhao, J. Z.-7th I. S. on Parallel, and undefined 2004, *Particle swarm optimization algorithm in signal detection and blind extraction*, .
- [52] F. Fang, T. Chen, R. R.-2002 I. International, and undefined 2002, *Floating-point bit-width optimization for low-power signal processing applications*, .
- [53] D. Karaboga, B. Akay, and C. Ozturk, *Artificial Bee Colony (ABC) optimization algorithm for training feed-forward neural networks*, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, (2007) pp.318–329.
- [54] I. Aljarah, H. Faris, and S. Mirjalili, *Optimizing connection weights in neural networks using the whale optimization algorithm*, **Soft Comput.**, 22:1 (2018).
- [55] A. Meng, J. Ge, H. Yin, S. C.-E. C. and Management, and undefined 2016, *Wind speed forecasting based on wavelet packet decomposition and artificial neural*

- networks trained by crisscross optimization algorithm, .
- [56] B. Garro, H. Sossa, R. V.-2009 I. Joint, and undefined 2009, *Design of artificial neural networks using a modified particle swarm optimization algorithm*, .
 - [57] J. Paul and T. Yusiong, *Intelligent Systems and Applications*, **Intell. Syst. Appl.**, 01 (2013) 69–80.
 - [58] P. Chang, J. Lin, C. L.-C. methods and programs in, and undefined 2012, *An attribute weight assignment and particle swarm optimization algorithm for medical database classifications*, .
 - [59] Q. Shen, W. Shi, W. Kong, B. Y.- Talanta, and undefined 2007, *A combination of modified particle swarm optimization algorithm and support vector machine for gene selection and tumor classification*, .
 - [60] K. C. Lin, K. Y. Zhang, Y. H. Huang, J. C. Hung, and N. Yen, *Feature selection based on an improved cat swarm optimization algorithm for big data classification*, **J. Supercomput.**, 72:8 (2016) 3210–3221.
 - [61] W. Tao, H. Jin, L. L.-P. R. Letters, and undefined 2007, *Object segmentation using ant colony optimization algorithm and fuzzy entropy*, .
 - [62] L. Li, D. L.-P. in N. Science, and undefined 2008, *Fuzzy entropy image segmentation based on particle swarm optimization*, .
 - [63] S. Song, H. Jia, and J. Ma, *A Chaotic Electromagnetic Field Optimization Algorithm Based on Fuzzy Entropy for Multilevel Thresholding Color Image Segmentation*, .
 - [64] M. Sun and D. Pang, *Vehicle routing optimisation algorithm for agricultural products logistics distribution*, **Int. J. Appl. Decis. Sci.**, 10:4 (2017) 327–334.
 - [65] S. Zhang, N. Chen, X. Song, J. Y.-R. P. A. P. and Practice, and undefined 2019, *Optimizing decision-making of regional cold chain logistics system in view of low-carbon economy*, .
 - [66] X. Hua, X. Hu, W. Y.- Optik, and undefined 2016, *Research optimization on logistics distribution center location based on adaptive particle swarm algorithm*, .
 - [67] S. Deng, Y. Li, H. Guo, B. L.-D. D. in N. and Society, and undefined 2016, *Solving a closed-loop location-inventory-routing problem with mixed quality defects returns in e-commerce by hybrid ant colony optimization algorithm*, .
 - [68] W. Yang, Q. Xie, M. L.-J. of C. Research, and undefined 2018, *Inventory Control Method of Reverse Logistics for Shipping Electronic Commerce Based on Improved Multi-objective Particle Swarm Optimization Algorithm*, .
 - [69] J. F.-C. on I. Transportation, & B. D., and undefined 2020, *Optimization Method of Cross-Border E-Commerce Logistics Distribution Route Based on Improved Genetic Algorithm*, .
 - [70] P. Sharma, K. Choudhary, K. Gupta, ... R. C.-A. I. in, and undefined 2020, *Artificial plant optimization algorithm to detect heart rate & presence of heart disease using machine learning*, .
 - [71] M. Alam, A. Ash-Saki, and S. Ghosh, *Accelerating Quantum Approximate Optimization Algorithm using Machine Learning* (2020) pp.686–689.
 - [72] D. Gupta et al., *Artificial plant optimization algorithm to detect infected leaves*

using machine learning, *Expert Systems*, (2020).

- [73] Z. Yin and N. Razmjooy, *PEMFC IDENTIFICATION USING DEEP LEARNING DEVELOPED BY IMPROVED DEER HUNTING OPTIMIZATION ALGORITHM*, **Int. J. Power Energy Syst.**, 40:2 (2020).
- [74] T. Ozcan and • Alper Basturk, *Human action recognition with deep learning and structural optimization using a hybrid heuristic algorithm*, .
- [75] F. Hoseini, A. Shahbahrani, and P. Bayat, *Journal of Advances in Computer Research Quarterly pISSN: 2345-606x eISSN: 2345-6078 Sari Branch*, 9:4 (2018).
- [76] S. Saremi, S. Mirjalili, S. Mirjalili, and J. Song Dong, Grasshopper optimization algorithm: Theory, literature review, and application in hand posture estimation, *Studies in Computational Intelligence*, 811. Springer Verlag, pp. 107–122, 2020.
- [77] N. Panagant, N. Pholdee, S. Bureerat, A. R. Yıldız, and S. M. Sait, *Seagull optimization algorithm for solving real-world design optimization problems*, **Mater. Test.**, 62:6 (2020) 640–644.
- [78] W. Li, G. Wang, A. A.-K.-B. Systems, and undefined 2020, *Learning-based elephant herding optimization algorithm for solving numerical optimization problems*, .
- [79] K. Zervoudakis, S. T.-C. & I. Engineering, and undefined 2020, *A mayfly optimization algorithm*, .
- [80] B. S. Yıldız, A. R. Yıldız, E. I. Albak, H. Abderazek, S. M. Sait, and S. Bureerat, *Butterfly optimization algorithm for optimum shape design of automobile suspension components*, **Mater. Test.**, 62:4 (2020) 365–370.
- [81] Q. Pham, S. Mirjalili, ... N. K.-I. T., and undefined 2020, *Whale optimization algorithm with applications to resource allocation in wireless networks*, .
- [82] D. Pelusi, R. Mascella, L. Tallini, J. Nayak, ... B. N.-K.-B., and undefined 2020, *An Improved Moth-Flame Optimization algorithm with hybrid search phase*, .
- [83] S. Deb, X. Z. Gao, K. Tammi, K. Kalita, and P. Mahanta, *A New Teaching–Learning-based Chicken Swarm Optimization Algorithm*, **Soft Comput.**, 24:7 (2020) 5313–5331.
- [84] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, *Harris hawks optimization: Algorithm and applications*, **Futur. Gener. Comput. Syst.**, 97 (2019) 849–872.
- [85] S. Arora and S. Singh, *Butterfly optimization algorithm: a novel approach for global optimization*, **Soft Comput.**, 23:3 (2019) 715–734.
- [86] A. Faramarzi, M. Heidarinejad, ... B. S.-K.-B., and undefined 2020, *Equilibrium optimizer: A novel optimization algorithm*, .
- [87] F. Wang et al., *A hybrid particle swarm optimization algorithm using adaptive learning strategy*, .
- [88] M. E.-A.-S. and evolutionary computation and undefined 2017, *Global-best brain storm optimization algorithm*, .
- [89] M. Tanweer, S. Suresh, N. S.-I. Sciences, and undefined 2015, *Self regulating particle swarm optimization algorithm*, .
- [90] E. Ali, S. A. Elazim, A. A.- Energy, and undefined 2016, *Ant lion optimization*

algorithm for renewable distributed generations, .

- [91] A. I. Hafez, H. M. Zawbaa, E. Emary, and A. E. Hassanien, *Sine cosine optimization algorithm for feature selection*, Proceedings of the 2016 International Symposium on INnovations in Intelligent SysTems and Applications, INISTA 2016, (2016).
- [92] E. Cuevas, M. Cienfuegos, ... D. Z.-E. S. with, and undefined 2013, *A swarm optimization algorithm inspired in the behavior of the social-spider, .*
- [93] H. Zheng and Y. Zhou, *A Novel Cuckoo Search Optimization Algorithm Base on Gauss Distribution **, **J. Comput. Inf. Syst.**, 8:10 (2012).
- [94] S. Huang, X. L.-I. T. on P. Systems, and undefined 2012, *A plant growth-based optimization approach applied to capacitor placement in power systems, .*
- [95] V. M. Dunbabin *et al.*, *Modelling root-soil interactions using three-dimensional models of root growth, architecture and function*, **Plant Soil**, 372:1–2 (2013) 93–124.
- [96] H. Zhang, Y. Zhu, D. Z.-2013 I. C. on, and undefined 2013, *Self-adaptive root growth model for constrained multi-objective optimization, .*
- [97] H. Zhang, Y. Zhu, and H. Chen, *Root growth model for simulation of plant root system and numerical function optimization*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2012) pp.641–648.
- [98] E. Pourjafari, H. M.-S. and E. Computation, and undefined 2012, *Solving nonlinear equations systems with a new approach based on invasive weed optimization algorithm and clustering, .*
- [99] A. Halim, ... I. I.-I. S. C. on R. and, and undefined 2013, *Nonlinear plant modeling using neuro-fuzzy system with Tree Physiology Optimization, .*
- [100] C. Tang, R. Liu, J. N.-E. ir Elektrotechnika, and undefined 2013, *A novel wireless sensor network localization approach: localization based on plant growth simulation algorithm, .*
- [101] H. Guo, H. Tao, S. Salih, Z. Y.-E. Reports, and undefined 2020, *Optimized parameter estimation of a PEMFC model based on improved Grass Fibrous Root Optimization Algorithm, .*
- [102] J. Yun *et al.*, *Droplet property optimization in printable electronics fabrication using root system growth algorithm, .*
- [103] W. Zhengjun, F. Xiang, ... Y. H.-A. R. of, and undefined 2019, *Root growth swarm optimization algorithm, .*
- [104] H. A. R. Akkar and F. R. Mahdi, *Grass Fibrous Root Optimization Algorithm Nano-Electronics View project robotic system View project Grass Fibrous Root Optimization Algorithm*, **Artic. Int. J. Intell. Syst. Appl.**, 6 (2017) 15–23.
- [105] Q. Fan, Z. Chen, Z. Li, Z. Xia, J. Yu, and D. Wang, *A new improved whale optimization algorithm with joint search mechanisms for high-dimensional global optimization problems*, 1 3.
- [106] S. Gupta, D. Kumar, K. M.-J. of Multiple-Valued, and undefined 2020, *Chaotic and Co-variance Based Artificial Bee Colony Algorithm., .*
- [107] S. Mazloom, A. E.-M.- Chaos, S. & Fractals, and undefined 2009, *Color image*

encryption based on coupled nonlinear chaotic map, .

- [108] X. J. Tong *et al.*, *An image encryption algorithm based on the perturbed high-dimensional chaotic map*, 80 (2015) 1493–1508.
- [109] A. Pisarchik, M. Z.-E. Methods, S. and Security, and undefined 2010, *Chaotic map cryptography and security, .*
- [110] P. Zhen, G. Zhao, L. Min, ... X. L.-N. I. C., and undefined 2014, *A survey of chaos-based cryptography, .*
- [111] T. Habutsu, Y. Nishio, I. Sasase, and S. Mori, *A secret key cryptosystem by iterating a chaotic map*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (1991) pp.127–140.
- [112] S. Masood, M. Doja, P. C.-J. of Computational, and undefined 2020, *Chaos Based Network Initialization Approach for Feed Forward Artificial Neural Networks, .*
- [113] H. Abdullah, H. A.-2017 I. C. on, and undefined 2017, *Image encryption using hybrid chaotic map, .*
- [114] C. H.- Optik and undefined 2019, *An image encryption algorithm based on modified logistic chaotic map, .*
- [115] Y. Naseer, D. Shah, T. S.-M. and Microsystems, and undefined 2019, *A novel approach to improve multimedia security utilizing 3D mixed chaotic map, .*
- [116] A. A. Ewees, M. A. El Aziz, and A. E. Hassanien, *Chaotic multi-verse optimizer-based feature selection*, **Neural Comput. Appl.**, 31:4 (2019) 991–1006.
- [117] A. A. Awad, A. F. Ali, and T. Gaber, *Feature Selection Method Based on Chaotic Maps and Butterfly Optimization Algorithm*, *Advances in Intelligent Systems and Computing*, (2020) pp.159–169.
- [118] J. Wei, X. Liao, K. Wong, T. X.- Chaos, S. & Fractals, and undefined 2006, *A new chaotic cryptosystem, .*
- [119] K. Mishra and R. Saharan, *A fast image encryption technique using henon chaotic map*, *Advances in Intelligent Systems and Computing*, (2019) pp.329–339.
- [120] S. Sheela, K. Suresh, ... D. T.-I. C. on, and undefined 2017, *Secured text communication using chaotic maps, .*
- [121] Z. Muhammad, F. Ö.-I. Access, and undefined 2020, *An Image Encryption Algorithm Based on Chaotic Selection of Robust Cryptographic Primitives, .*
- [122] W. Qiao, Z. Y.-I. Access, and undefined 2019, *Modified dolphin swarm algorithm based on chaotic maps for solving high-dimensional function optimization problems, .*
- [123] D. Karaboğa, *Yapay Zeka Optimizasyon Algoritmaları*, (2017).
- [124] H. Abedinpourshotorban, ... S. S.-S. and E., and undefined 2016, *Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, .*
- [125] Ş. I. Birbil and S. C. Fang, *An electromagnetism-like mechanism for global optimization*, **J. Glob. Optim.**, 25:3 (2003) 263–282.

- [126] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, *GSA: A Gravitational Search Algorithm*, **Inf. Sci. (Ny)**, 179:13 (2009) 2232–2248.
- [127] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, *Water cycle algorithm - A novel metaheuristic optimization method for solving constrained engineering optimization problems*, **Comput. Struct.**, 110–111 (2012) 151–166.
- [128] A. Husseinazadeh Kashan, *A new metaheuristic for optimization: Optics inspired optimization (OIO)*, **Comput. Oper. Res.**, 55 (2015) 99–125.
- [129] O. K. Erol and I. Eksin, *A new optimization method: Big Bang-Big Crunch*, **Adv. Eng. Softw.**, 37:2 (2006) 106–111.
- [130] A. Ersöz, *Büyük Patlama Büyük Çöküş Optimizasyon Algoritması Tabanlı Bulanık Modelleme Yöntemi ve Yazılımı*, (2011).
- [131] L. Xie, J. Zeng, & Z. C.-2009 W. C. on N., and undefined 2009, *General framework of artificial physics optimization algorithm*, .
- [132] R. A. Formato, *Central Force Optimization*, .
- [133] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, *Optimization by simulated annealing*, **Science (80-.)**, (1983).
- [134] A. Asghar Heidari, • Rahim, A. Abbaspour, • Ahmad, and R. Jordehi, *An efficient chaotic water cycle algorithm for optimization tasks*, **Neural Comput. Appl.**, 28 (2037).
- [135] B. A.-E. S. with Applications and undefined 2011, *ACROA: artificial chemical reaction optimization algorithm for global optimization*, .
- [136] B. A.-E. S. with Applications and undefined 2012, *A novel chemistry based metaheuristic optimization method for mining of classification rules*, .
- [137] A. E. Yildirim and A. Karci, *Applications of artificial atom algorithm to small-scale traveling salesman problems*, **Soft Comput.**, 22:22 (2018) 7619–7631.
- [138] J. H.-S. american and undefined 1992, *Genetic algorithms*, .
- [139] D. Simon, *Biogeography-based optimization*, **IEEE Trans. Evol. Comput.**, 12:6 (2008) 702–713.
- [140] M. Baygın and M. Karaköse, *Adaptif Yapay Bağışık Sistem Tabanlı Grup Asansör Kontrol Algoritması*, (2011).
- [141] E. A.-F. Ü. F. B. E. Y. Lisans and undefined 2007, *Kablosuz Ağlarda Yapay Bağışıklık Sistemi Kullanılarak Saldırı Tespiti ve Güvenlik*, .
- [142] D. D.-I. computational intelligence magazine and undefined 2006, *Advances in artificial immune systems*, .
- [143] L. De Castro, F. V. Z.-I. transactions on, and undefined 2002, *Learning and optimization using the clonal selection principle*, .
- [144] İ. AYDIN, M. KARAKOSE, E. A.-J. of the F. of, and undefined 2009, *NEGATİF SEÇİM TABANLI BULANIK ARIZA TEŞHİS MODELİ*, .
- [145] R. Fotohi, S. Firoozi Bari, and M. Yusefi, *Securing Wireless Sensor Networks Against Denial-of-Sleep Attacks Using RSA Cryptography Algorithm and Interlock Protocol*, **Int. J. Commun. Syst.**, 33:4 (2020).
- [146] E. TANYILDIZI and G. DEMİR, *Nümerik Optimizasyon için Kaotik Altın Sinüs Algoritması*, **Fırat Üniversitesi Mühendislik Bilim. Derg.**, 31:1 (2019) 91–97.

- [147] F. Burak Demir, T. Tuncer, A. Fatih Kocamaz, M. Turgut, Ö. Üniversitesi Bilgisayar, and T. Bölümü, *Lojistik-Gauss Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Yöntemi*, (2019).
- [148] F. B. Demir, T. Tuncer, and A. F. Kocamaz, *Lojistik-singer harita tabanlı yeni bir kaotik sürü optimizasyon yöntemi*, 2019 International Conference on Artificial Intelligence and Data Processing Symposium, IDAP 2019, (2019).
- [149] E. Atashpaz-Gargari and C. Lucas, *Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition*, 2007 IEEE Congress on Evolutionary Computation, CEC 2007, (2007) pp.4661–4667.
- [150] A. Borji, *A new global optimization algorithm inspired by parliamentary political competitions*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2007) pp.61–71.
- [151] A. Borji and M. Hamidi, *A NEW APPROACH TO GLOBAL OPTIMIZATION MOTIVATED BY PARLIAMENTARY POLITICAL COMPETITIONS*, **Int. J. Innov. Comput. Inf. Control ICIC Int. c**, x (2008).
- [152] Z. H. Zhan, J. Zhang, Y. H. Shi, and H. L. Liu, *A modified brain storm optimization*, 2012 IEEE Congress on Evolutionary Computation, CEC 2012, (2012).
- [153] R. V. Rao and V. Patel, *An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems*, **Sci. Iran.**, 20:3 (2013) 710–720.
- [154] S. Kiziloluk and ve Bilal ALATAŞ, *SOSYAL TABANLI GÜNCEL SEZGİSEL OPTİMİZASYON ALGORİTMALARI*, 13 (2012).
- [155] A. Tamimi, H. Sadjadian, and H. Omranpour, *Mobile robot global localization using Imperialist Competitive Algorithm*, ICACTE 2010 - 2010 3rd International Conference on Advanced Computer Theory and Engineering, Proceedings, (2010).
- [156] S. L. Ghalehpardaz and M. Shafiee, *Speed control of DC motor using imperialist competitive algorithm based on PI-like FLC*, Proceedings - CIMSIm 2011: 3rd International Conference on Computational Intelligence, Modelling and Simulation, (2011) pp.28–33.
- [157] A. Afsari, M. Ramezani, S. Heidari, and J. Karimisharifabadi, *Imperialist Competitive Algorithm (ICA) Approach for Optimization of the Surface Grinding Process*, **J. Mod. Process. Manuf. Prod.**, 9:1 (2020).
- [158] D. Lei, Y. Yuan, J. Cai, and D. Bai, *International Journal of Production Research An imperialist competitive algorithm with memory for distributed unrelated parallel machines scheduling An imperialist competitive algorithm with memory for distributed unrelated parallel machines scheduling*, **Int. J. Prod. Res.**, 58:2 (2020) 597–614.
- [159] M. Abdechiri, K. Faez, and H. Bahrami, *Neural network learning based on chaotic imperialist competitive algorithm*, Proceedings - 2010 2nd International Workshop on Intelligent Systems and Applications, ISA 2010, (2010).
- [160] B. Gupta, M. Joshi, ... R. B.-A. at S., and undefined 2020, *A New Approach to Assess Fractal Coefficients using Imperialist Competitive Algorithm for Color*

Images, .

- [161] F. Altunbey and B. Alatas, *Overlapping Community Detection in Social Networks Using Parliamentary Optimization Algorithm*, **Int. J. Comput. Networks Appl.**, 2:1 .
- [162] Zong Woo Geem, Joong Hoon Kim, and G. V. Loganathan, *A New Heuristic Optimization Algorithm: Harmony Search*, **Simulation**, 76:2 (2001) 60–68.
- [163] K. Lee, Z. G.-C. methods in applied mechanics and, and undefined 2005, *A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice*, .
- [164] G. Özdemir, *HARMONİ ARAMA ALGORİTMASI İLE ELEKTRİK DAĞITIM ŞEBEKELERİNDE GÜÇ YÖNETİMİ İÇİN VOLT-VAR CİHAZLARININ KULLANIMININ OPTİMİZASYONU*, .
- [165] M. Kiani-Moghaddam, M. S.-B.-I. C. for, and undefined 2017, *An Innovative Multi-Stage Multi-Dimensional Multiple-Inhomogeneous Melody Search Algorithm: Symphony Orchestra Search Algorithm (SOSA)*, .
- [166] S. Ashrafi, A. D.-2011 11th I. Conference, and undefined 2011, *A novel and effective algorithm for numerical optimization: melody search (MS)*, .
- [167] S. Ashrafi, A. D.-E. applications of artificial intelligence, and undefined 2013, *Performance evaluation of an improved harmony search algorithm for numerical optimization: Melody Search (MS)*, .
- [168] A. H. Kashan, *League Championship Algorithm: A new algorithm for numerical function optimization*, SoCPaR 2009 - Soft Computing and Pattern Recognition, (2009) pp.43–48.
- [169] M. Optimizasyon et al., *Fen Bilimleri Dergisi Fırat Univ*, **J. Sci.**, 27:1 (2015).
- [170] J. Kennedy, R. E.-N. Networks, T. I. I. Conference, and undefined 1995, *R. "Particle Swarm Optimization,"* .
- [171] M. Dorigo, T. S.-M. T. M. P. [Google Scholar], and undefined 2004, *Ant colony optimization*. Cambridge, .
- [172] X.-S. Yang, *Firefly Algorithms for Multimodal Optimization*, **Lect. Notes Comput. Sci.**, 5792 (2009).
- [173] S.-C. Chu, P. Tsai, and J.-S. Pan, *Cat Swarm Optimization*. Springer, Berlin, Heidelberg, pp. 854–858, 2006.
- [174] C. Liu, X. Yan, C. Liu, H. W.-C. J. of Electronics, and undefined 2011, *The wolf colony algorithm and its application*, .
- [175] S. Aslan, D. K.-A. S. Computing, and undefined 2020, *A genetic Artificial Bee Colony algorithm for signal reconstruction based big data optimization*, .
- [176] T. Apostolopoulos and A. Vlachos, *Application of the Firefly Algorithm for Solving the Economic Emissions Load Dispatch Problem*, **Int. J. Comb.**, 2011 (2011).
- [177] X.-S. Yang, *Firefly Algorithm, Lévy Flights and Global Optimization*, *Research and Development in Intelligent Systems XXVI*. Springer London, pp. 209–218, 2010.
- [178] W. D. Brown, *Mate choice in tree crickets and their kin*, **Artic. Annu. Rev. Entomol.**, 44:1 (1999) 371–396.

- [179] R. Stephen, J. H.-J. of E. Biology, and undefined 1995, *Sound production in crickets*, (1995).
- [180] A. E. Dolbear, *The Cricket as a Thermometer*, **Am. Nat.**, 31:371 (1897) 970–971.
- [181] A. Hedrick, R. K.-A. Behaviour, and undefined 2006, *Hiding behaviour in two cricket populations that differ in predation pressure*, .
- [182] B. Hedwig, *Pulses, patterns and paths: neurobiology of acoustic behaviour in crickets*, .
- [183] A. Hedrick, B. M.-J. of neuroscience methods, and undefined 2004, *A multichannel electronic monitor of acoustic behaviors, and software to parse individual channels*, .
- [184] B. Alataş, E. Akın, B. Ö.-E. O. Ü. Elektrik, and undefined 2007, *Kaotik Haritalı Parçacık Sürü Optimizasyon Algoritmaları*, .
- [185] Z. Elhadj, J. S.-I. J. of B. and Chaos, and undefined 2011, *On the dynamics of a new 2-D rational mapping*, **World Sci.**, 21:1 (2011) 155–160.
- [186] D. L.- Chaos, S. & Fractals, and undefined 2015, *A new discrete chaotic map based on the composition of permutations*, .
- [187] E. Tanyildizi, T. Cigal, F. Üniversitesi, and Y. M. Bölümü, *Kaotik Haritalı Balina Optimizasyon Algoritmaları*, 29:1 (2017).
- [188] M. Canayaz, R. Ö.-2018 26th S. P. and, and undefined 2018, *Training artificial neural network with Chaotic Cricket Algorithm*, .
- [189] S. Özdemir, I. A.-I. I. ve Finans, and undefined 2014, *Hisse senedi piyasalarının kaotik yapısı ve yapay sinir ağları ile öngörüsü: İMKB-100 örneği*, .
- [190] H. Nozawa, *A neural network model as a globally coupled map and applications based on chaos*, 2:3 (1992) 377–386.
- [191] H. M. Zawbaa, E. Emary, and C. Grosan, *Feature selection via chaotic antlion optimization*, **PLoS One**, 11:3 (2016).
- [192] F. Demir, T. Tuncer, A. Kocamaz, F. E.-M. Hypotheses, and undefined 2020, *A survival classification method for hepatocellular carcinoma patients with chaotic Darcy optimization method based feature selection*, .
- [193] C. Marangi, L. Angelini, ... F. D. C.-I. and S., and undefined 2001, *Clustering by inhomogeneous chaotic maps in landmine detection*, .
- [194] L. Chuang, C. Hsiao, C. Y.-E. systems with Applications, and undefined 2011, *Chaotic particle swarm optimization for data clustering*, .
- [195] İ. H. Karaçizmeli, S. Kaya, and A. Gümüşçü, *HARRAN ÜNİVERSİTESİ MÜHENDİSLİK DERGİSİ Hibrit Ateşböceği ve Parçacık Sürü Algoritmasının Kaotik Haritalar ile İyileştirilmesi Improving Hybrid Firefly and Particle Swarm Algorithm with Chaotic Maps Yazar(lar) (Author(s)): İbrahim Berkan AYDİLEK 1 , Mehmet Emin TENKEKİ*, .
- [196] E. N. Benderskaya, *Nonlinear Trends in Modern Artificial Intelligence: A New Perspective*. pp. 113–124, 2013.
- [197] P. Arena, R. Caponetto, L. Fortuna, and D. Porto, *Bifurcation and chaos in noninteger order cellular neural networks*, **Int. J. Bifurcat. Chaos**, 8:7 (1998) 1527–1539.

- [198] J. C. Sprott, (2020). <http://sprott.physics.wisc.edu/chaos/eulermap.htm> (on-line access on 10, Jul., 2020).
- [199] J. C. Sprott, (2020). <http://sprott.physics.wisc.edu/chaos/eulermap.htm> (on-line access on 8, Jul., 2020).
- [200] J. Hateley, *The Lorenz system*, .
- [201] Y. Yu, S. Z.- Chaos, S. & Fractals, and undefined 2004, *Adaptive backstepping synchronization of uncertain chaotic system*, .
- [202] M. G.-B. 2018 C. Proceedings and undefined 2018, *3d printing chaos*, .
- [203] G. Peterson, *Arnold's Cat Map*, .
- [204] R. Rahim, S. Tinggi, and E. Hariyanto, *Arnold's Cat Map Algorithm in Digital Image Encryption*, **Artic. Int. J. Sci. Res.**, 5 (2016).
- [205] Y. Liu, Z. Li, X. Cai, Y. Y.-A. M. and Materials, and undefined 2012, *Local stability and Hopf bifurcation analysis of the Arneodo's system*, **Trans Tech Publ.**, .
- [206] L. Liu and S. Miao, *An image encryption algorithm based on Baker map with varying parameter*, **Multimed. Tools Appl.**, 76:15 (2017) 16511–16527.
- [207] Q. Din, *A novel chaos control strategy for discrete-time Brusselator models*, **J. Math. Chem.**, 56:10 (2018) 3045–3075.
- [208] E. Ott, (2020). <https://web.archive.org/web/20140701200254/http://www.ba.infn.it/~zito/ds/basin.html> (on-line access on 10, Jul., 2020).
- [209] C. Letellier, T. D. Tsankov, G. Byrne, and R. Gilmore, *Large-scale structural reorganization of strange attractors*, **Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.**, 72:2 (2005).
- [210] R. Zahmoul, R. Ejbali, M. Z.-O. and L. in Engineering, and undefined 2017, *Image encryption based on new Beta chaotic maps*, .
- [211] J. Lü and G. Chen, *A new chaotic attractor coined*, **Int. J. Bifurcat. Chaos**, 12:3 (2002) 659–661.
- [212] R. B.-P. of P. Seminar and undefined 1976, *Bifurcations of a limit cycle for a family of vector fields on the plane*, .
- [213] M. H. Jensen, P. Bak, and T. Bohr, *Complete devil's staircase, fractal dimension, and universality of mode- locking structure in the circle map*, **Phys. Rev. Lett.**, 50:21 (1983) 1637–1639.
- [214] L. Tam, J. Chen, H. Chen, W. T.- Chaos, S. & Fractals, and undefined 2008, *Generation of hyperchaos from the Chen–Lee system via sinusoidal perturbation*, .
- [215] B. Jacek Graczyk and G. Swi, *INDUCED EXPANSION FOR QUADRATIC POLYNOMIALS*, (1996).
- [216] L. O. CHUA, M. ITOH, L. KOCAREV, and K. ECKERT, *CHAOS SYNCHRONIZATION IN CHUA'S CIRCUIT*, **J. Circuits, Syst. Comput.**, 03:01 (1993) 93–108.
- [217] Y. Liu, X. Tong, S. H.-S. P. I. Communication, and undefined 2013, *A family of new complex number chaotic maps based image encryption algorithm*, .

- [218] I. Kovacic and M. Brennan, *The Duffing equation: nonlinear oscillators and their behaviour*, (2011).
- [219] L. Sheu, H. Chen, J. Chen, L. T.- Chaos, S. & Fractals, and undefined 2007, *Chaotic dynamics of the fractionally damped Duffing equation*, .
- [220] J. Milnor, *Remarks on iterated cubic maps*, **Exp. Math.**, 1:1 (1992) 5–24.
- [221] G. Cai and J. Huang, *A New Finance Chaotic Attractor*, **Int. J. Nonlinear Sci.**, 3:3 (2007).
- [222] C. P.-C. and Fractals and undefined 1998, *A vacation on Mars—an artist's journey in a computer graphics world*, .
- [223] G. Baier, M. K.-P. L. A, and undefined 1990, *Maximum hyperchaos in generalized Hénon maps*, .
- [224] P. B. January and U. February, *Peter de Jong Attractors*, :February (2004).
- [225] L. J.-G.-C. Physics and undefined 2005, *Chaotic dynamics and synchronization of fractional-order Genesio-Tesi systems*, .
- [226] G. C. Wu and D. Baleanu, *Discrete chaos in fractional delayed logistic maps*, **Nonlinear Dyn.**, 80:4 (2015) 1697–1703.
- [227] K. Bowman, P. C.-J. of the atmospheric sciences, and undefined 1997, *Interhemispheric exchange by seasonal modulation of the Hadley circulation*, .
- [228] C. Grebogi, E. Kostelich, E. Ott, and J. A. Yorke, *MULTI-DIMENSIONED INTERTWINED BASIN BOUNDARIES: BASIN STRUCTURE OF THE KICKED DOUBLE ROTOR*, (1987).
- [229] S. Vaidyanathan and A. T. Azar, *Adaptive control and synchronization of halvorsen circulant chaotic systems*, *Studies in Fuzziness and Soft Computing*, 337. Springer Verlag, pp. 225–247, 2016.
- [230] S. S. Hreshee, A. Mahdi, and A. K. Jawad, *Digital Chaotic Scrambling of Voice Based on Duffing Map*, **Commun. Eng. J.**, 1:2 (2016) 16–21.
- [231] G. Innocenti, A. Morelli, R. Genesio, and A. Torcini, *Dynamical phases of the Hindmarsh-Rose neuronal model: Studies of the transition from bursting to spiking chaos*, 17:4 (2007).
- [232] S. Prajwalasimha, ... U. S.-I. C. on, and undefined 2017, *Multimedia data encryption based on discrete dyadic transformation*, .
- [233] Z. Y.-A. M. and Computation and undefined 2005, *Controlling hyperchaos in the new hyperchaotic Chen system*, .
- [234] C. Bregler, J. M.-P. 1998 I. C. Society, and undefined 1998, *Tracking people with twists and exponential maps*, .
- [235] A. Khan and A. Tyagi, *Fractional order disturbance observer based adaptive sliding mode hybrid projective synchronization of fractional order Newton–Leipnik chaotic system*, **Int. J. Dyn. Control**, 6:3 (2018) 1136–1149.
- [236] V. S. Anishchenko and G. I. Strelkova, *Irregular attractors*, **Discret. Dyn. Nat. Soc.**, 2 (1998).
- [237] X. Li, L. Wang, Y. Yan, P. L.- Optik, and undefined 2016, *An improvement color image encryption algorithm based on DNA operations and real and complex chaotic systems*, .

- [238] Y. G. Guen, M. Le Ravalec, and L. Ricard, *Upscaling: Effective Medium Theory, Numerical Methods and the Fractal Dream*, 163:5–6 (2006) 1175–1192.
- [239] Z. Xiaohong, Z. Z.- TELKOMNIKA, and undefined 2013, *Hyper-chaotic Lü System Simulation Design of Digital Circuit Based on DSP Builder*, .
- [240] O. R.-P. L. A and undefined 1979, *An equation for hyperchaos*, .
- [241] F. Pichler and J. Scharinger, *Finite Dimensional Generalized Baker Dynamical Systems for Cryptographic Applications*, .
- [242] (2020).
https://web.archive.org/web/20151222083004/http://jlswebs.blogspot.de/2011/12/hyperchaotic_23.html (on-line access on 10, Jul., 2020).
- [243] M. Khan and Z. Asghar, *A novel construction of substitution box for image encryption applications with Gingerbreadman chaotic map and S8 permutation*, **Neural Comput. Appl.**, 29:4 (2018) 993–999.
- [244] K. Ikeda and K. Matsumoto, *Study of a high-dimensional chaotic attractor*, **J. Stat. Phys.**, 44:5–6 (1986) 955–983.
- [245] R. Bagula, (2020). <http://paulbourke.net/fractals/grinchdragon/> (on-line access on 10, Jul., 2020).
- [246] G. P. King and S. Gaito, *Bistable chaos. II. Unfolding the cusp Article in Physical Review A* , 46:6 (1992) 3092–3099.
- [247] H. Ben Maallem, P. Richard, J.-L. Ferrier, and A. Labib, *Using Gumowski-Mira Maps for Artistic Creation*, .
- [248] N. A. Kudryashov, *Exact solutions of the generalized Kuramoto-Sivashinsky equation*, 147:5 (1990).
- [249] C. Letellier, E. Roulin, O. R.- Chaos, S. & Fractals, and undefined 2006, *Inequivalent topologies of chaos in simple equations*, .
- [250] B. DENG, *CONSTRUCTING HOMOCLINIC ORBITS AND CHAOTIC ATTRACTORS*, **Int. J. Bifurc. Chaos**, 04:04 (1994) 823–841.
- [251] M. Benedicks, L. C.-A. of Mathematics, and undefined 1991, *The dynamics of the Hénon map*, .
- [252] J. Lu, G. Chen, S. Celikovský, and L. Jinhu, *Toward Stronger Robustness of Network Controllability View project synchronizability of multiplex networks View project BRIDGE THE GAP BETWEEN THE LORENZ SYSTEM AND THE CHEN SYSTEM*, **Artic. Int. J. Bifurc. Chaos**, 12:12 (2002) 2917–2926.
- [253] U. Arshad, S. I. Batool, and M. Amin, *A Novel Image Encryption Scheme Based on Walsh Compressed Quantum Spinning Chaotic Lorenz System*, **Int. J. Theor. Phys.**, 58:10 (2019) 3565–3588.
- [254] H. Nusse and J. Yorke, *Dynamics: numerical explorations: accompanying computer program dynamics*, (2012).
- [255] D. J. Gagne, H. M. Christensen, A. C. Subramanian, and A. H. Monahan, *Machine Learning for Stochastic Parameterization: Generative Adversarial Networks in the Lorenz '96 Model*, **J. Adv. Model. Earth Syst.**, 12:3 (2020).
- [256] D. Hitzl, F. Z.-P. D. N. Phenomena, and undefined 1985, *An exploration of the Hénon quadratic map*, .

- [257] V. Azhmyakov, C. Londoño, ... P. O.-2019 I. 4th, and undefined 2019, *Consistent Approximations of the Control-Affine Systems with Bounded Uncertainties: Application to the Controlled Lotka-Volterra Dynamics*, .
- [258] O. R.-P. L. A and undefined 1977, *Horseshoe-map chaos in the Lorenz equation*, .
- [259] A. Negou, D. T.- Chaos, S. & Fractals, and undefined 2018, *Periodicity, chaos and multiple coexisting attractors in a generalized Moore–Spiegel system*, .
- [260] (2020). <http://www.jamesh.id.au/fractals/orbit/hopalong.html> (on-line access on 10, Jul., 2020).
- [261] Y. Liu, H. I.-I. Access, and undefined 2020, *Antimonotonicity, Chaos and Multidirectional Scroll Attractor in Autonomous ODEs Chaotic System*, .
- [262] R. Harrison, *True Random Number Generation from a High Frequency Chaotic Jerk Circuit*, (2019).
- [263] ... X. J.-I. C. on I. C. and and undefined 2010, *Image Encryption using the Ikeda map*, .
- [264] S. Bendoukha and S. Abdelmalek, *Complete synchronization of the Newton--Leipnik reaction diffusion chaotic system*, (2019).
- [265] W. V.-A. of Mathematics and undefined 1982, *Gauss measures for transformations on the space of interval exchange maps*, .
- [266] A. B. Nordmark, *NON-PERIODIC MOTION CAUSED BY GRAZING INCIDENCE IN AN IMPACT OSCILLATOR*, .
- [267] N. Singh, A. S.- Optik, and undefined 2010, *Digital image watermarking using gyrator transform and chaotic maps*, .
- [268] G. J. Martyna, M. L. Klein, and M. Tuckerman, *Nosé-Hoover chains: The canonical ensemble via continuous dynamics*, **J. Chem. Phys.**, 97:4 (1992) 2635–2643.
- [269] (2020). <https://web.archive.org/web/20151222103231/http://jlswebs.blogspot.de/2012/03/knot.html> (on-line access on 10, Jul., 2020).
- [270] T. Wang, K. Wang, N. J.-M. P. in Engineering, and undefined 2011, *Chaos control and hybrid projective synchronization of a novel chaotic system*, .
- [271] D. Lambi and D. Lambi, *A novel method of S-box design based on discrete chaotic map*, 87 (2017) 2407–2413.
- [272] F. Özkaynak and A. Özer, *Lojistik Harita ile Rasgele Sayı Üretilmesi ve İstatistikî Yöntemlerle Sınanması*, (2006).
- [273] (2020). <http://www.chaoscope.org/doc/attractors.htm> (on-line access on 10, Jul., 2020).
- [274] V. N. Belykh, N. V Barabash, and I. V Belykh, *A Lorenz-type attractor in a piecewise-smooth system: Rigorous results*, 29:10 (2019) 103108.
- [275] M. Widom, *Renormalization group analysis of quasi-periodicity in analytic maps*, **Commun. Math. Phys.**, 92:1 (1983) 121–136.
- [276] N. Pourmousa, S. Ebrahimi, M. M.-S. Energy, and undefined 2019, *Parameter estimation of photovoltaic cells using improved Lozi map based chaotic*

optimization Algorithm, .

- [277] V. Dobrushkin, (2020). <http://www.cfm.brown.edu/people/dobrush/am34/Mathematica/ch3/chaos.html> (on-line access on 10, Jul., 2020).
- [278] T. Aikawa, *The Pomeau-Manneville intermittent transition to chaos in hydrodynamic pulsation models*, **Astrophys. Space Sci.**, 139:2 (1987) 281–293.
- [279] T. H. Solomon and J. P. Gollub, *Chaotic particle transport in time-dependent Rayleigh-Bénard convection*, **Phys. Rev. A**, 38:12 (1988) 6280–6286.
- [280] K. Gdawiec, *Aesthetic patterns from the perturbed orbits of discrete dynamical systems*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2013) pp.358–366.
- [281] U. V.-P. L. A and undefined 2005, *Synchronization of Rikitake chaotic attractor using active control*, .
- [282] V. T.-2018 I. C. on and undefined 2018, *Image Encryption using Integer Wavelet Transform, Chaotic Maps and DNA rules*, .
- [283] C. Letellier, O. R.- Scholarpedia, and undefined 2006, *Rosler attractor*, .
- [284] L. Skanderova and I. Zelinka, *Arnold cat map and sinai as chaotic numbers generators in evolutionary algorithms*, Lecture Notes in Electrical Engineering, (2014) pp.381–389.
- [285] S. Vaidyanathan, *Sliding Mode Control of Rucklidge Chaotic System for Nonlinear Double Convection*, **Artic. Int. J. ChemTech Res.**, (2015).
- [286] R. Venegeroles, *Calculation of superdiffusion for the Chirikov-Taylor model*, **Phys. Rev. Lett.**, 101:5 (2008).
- [287] (2020). <https://web.archive.org/web/20151222172911/http://jlswws.blogspot.de/2011/10/sakarya.html> (on-line access on 10, Jul., 2020).
- [288] H. Dong, Y. Zhang, X. Z.-C. in N. S. and, and undefined 2016, *The new integrable symplectic map and the symmetry of integrable nonlinear lattice equation*, .
- [289] (2020). <https://web.archive.org/web/20151222101232/http://jlswws.blogspot.de/2011/10/saw-haw-pol.html> (on-line access on 10, Jul., 2020).
- [290] J. R.-B. of the A. M. Society and undefined 1972, *Topological conjugacy and structural stability for discrete dynamical systems*, .
- [291] S. O. Kareem, K. S. Ojo, and A. N. Njah, *Function projective synchronization of identical and non-identical modified finance and Shimizu-Morioka systems*, **Indian Acad. Sci.**, 79:1 (2012) 71–79.
- [292] N. B. Cowan and E. Agol, *Inverting Phase Functions to Map Exoplanets*, **Astrophys. J.**, 678:2 (2008) L129–L132.
- [293] Q. Lai and S. Chen, *Generating multiple chaotic attractors from Sprott B System*, **Int. J. Bifurc. Chaos**, 26:11 (2016).
- [294] P. Kumar et al., *A robust image encryption scheme using chaotic tent map and cellular automata*, **Nonlinear Dyn.**, 100 (2020) 2877–2898.

- [295] J. C. Sprott, (2020). <http://www.3d-meier.de/tut19/Seite300.html> (on-line access on 10, Jun., 2020).
- [296] V. Aboites, Y. Barmenkov, A. Kir'yanov, M. W.-R. in Physics, and undefined 2012, *Tinkerbells in a non-linear ring resonator*, .
- [297] F. Bilimleri et al., *BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ KAOTİK ELEKTRONİK DEVRE TASARIMI, GERÇEKLEMESİ VE BİR HABERLEŞME UYGULAMASI*, (2013).
- [298] L. dos S. Coelho, M. P.-E. S. with Applications, and undefined 2011, *A tuning strategy for multivariable PI and PID controllers using differential evolution combined with chaotic Zaslavskii map*, .
- [299] B. Stoyanov and M. Todorova, *Two Zaslavsky maps in pseudorandom byte generation Chaos based hash functions View project Special Issue "Chaos Based Computer Security and Functions" View project*, 2164 (2019).
- [300] J. C. Sprott, (2020). <http://sprott.physics.wisc.edu/chaos/symmetry.htm> (on-line access on 10, Jul., 2020).
- [301] G. Chen, E. V. Kudryashova, N. V. Kuznetsov, and G. A. Leonov, *Dynamics of the Zeraoulia-Sprott map revisited*, **Int. J. Bifurc. Chaos**, 26:7 (2016).
- [302] F. B. Demir, T. Tuncer, and A. F. Kocamaz, *A chaotic optimization method based on logistic-sine map for numerical function optimization*, **Neural Comput. Appl.**, (2020).
- [303] F. Musanna, D. Dangwal, S. Kumar, and V. Malik, *The Imaging Science Journal A chaos-based image encryption algorithm based on multiresolution singular value decomposition and a symmetric attractor*, **Taylor Fr.**, 68:1 (2020) 24–40.
- [304] L. Aguirre, ... S. B.-T. on C. and S. I., and undefined 1994, *Model reference control of regular and chaotic dynamics in the Duffing-Ueda oscillator*, .
- [305] M. Peng, Z. Zhang, Z. Qu, Q. B.-P. A. S. M. and its, and undefined 2020, *Qualitative analysis in a delayed Van der Pol oscillator*, .
- [306] J. Güémez and M. A. Matías, *Internal fluctuations in a model of chemical chaos*, **Phys. Rev. E**, 48:4 (1993).
- [307] S. Vaidyanathan et al., *Adaptive Backstepping Controller Design for the Anti-Synchronization of Identical WINDMI Chaotic Systems with Unknown Parameters and its SPICE Implementation*, **Artic. J. Eng. Sci. Technol. Rev.**, 8:2 (2015) 74–82.
- [308] D. He, C. He, L. Jiang, ... H. Z.-I. T. on, and undefined 2001, *Chaotic characteristics of a one-dimensional iterative map with infinite collapses*, .
- [309] H. K.-2019 I. C. on Engineering and undefined 2019, *Parkinson's Disease Recognition using Gauss Map based Chaotic Particle Swarm-Neural Network*, .
- [310] R. A. Ibrahim, D. Oliva, A. Abd, E. Ewees, S. Lu, and A. A. Ewees, *Feature Selection Based on Improved Runner-Root Algorithm Using Chaotic Singer Map and Opposition-Based Learning Hyperheuristics and hybrid optimization algorithms View project Interaction between fractional calculus and different metaheuristics optimization algorithms View project Feature Selection Based on Improved Runner-Root Algorithm Using Chaotic Singer Map and Opposition-Based Learning*, 10638 LNCS (2017) 156–166.

- [311] H. Zhu, Y. Zhao, Y. S.-I. Access, and undefined 2019, *2D logistic-modulated-sine-coupling-logistic chaotic map for image encryption*, .
- [312] C. Li, D. Lin, J. Lü, F. H.-I. MultiMedia, and undefined 2018, *Cryptanalyzing an image encryption algorithm based on autoblocking and electrocardiography*, .
- [313] J. Martínez-Ñonthe, ... A. C.-S.-M., and undefined 2012, *Chaotic block cryptosystem using high precision approaches to tent map*, .
- [314] Ç. Haritası, T. Kaotik, O. Yöntemi, and T. Tuncer, *Fen Bilimleri Dergisi*, .
- [315] Z. Hua, Y. Zhou, ... C. P.-2014 I. I., and undefined 2014, *Image encryption using 2D Logistic-Sine chaotic map*, .
- [316] G. Mala, D. L.-I. journal of heat and fluid flow, and undefined 1999, *Flow characteristics of water in microtubes*, .
- [317] L. J. Gillespie, *The Gibbs-Dalton law of partial pressures*, **Phys. Rev.**, 36:1 (1930) 121–131.
- [318] M. DÖLARSLAN, G. E.-T. J. of S. Reviews, and undefined 2012, *Toprak bitki ilişkileri açısından tuzluluk*, .
- [319] M. Bayazıt, *Hidroloji*, (1995).
- [320] S. Mirjalili, A. Gandomi, S. Mirjalili, S. S.-... in E. Software, and undefined 2017, *Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems*, .
- [321] S. M.-K. systems and undefined 2015, *Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm*, .
- [322] S. Saremi, S. Mirjalili, and A. Lewis, *Biogeography-based optimisation with chaos*, **Neural Comput. Appl.**, 25:5 (2014) 1077–1097.
- [323] S. Saremi, S. Mirjalili, S. M.-P. Technology, and undefined 2014, *Chaotic krill herd optimization algorithm*, .
- [324] H. Mittal, A. Kulhari, M. Saraswat, and R. Pal, *Chaotic Kbest gravitational search algorithm (CKGSA)*, .
- [325] K. Deb, Multi-objective optimization, *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, Second Edition. Springer US, pp. 403–450, 2014.
- [326] S. Z. Mirjalili, S. Mirjalili, S. Saremi, H. Faris, and I. Aljarah, *Grasshopper optimization algorithm for multi-objective optimization problems*, **Appl. Intell.**, 48:4 (2018) 805–820.
- [327] M. Kohli, S. A.-J. of computational design and, and undefined 2018, *Chaotic grey wolf optimization algorithm for constrained optimization problems*, .
- [328] S. Mirjalili, • Seyed, M. Mirjalili, and A. Hatamlou, *Multi-Verse Optimizer: a nature-inspired algorithm for global optimization*, .
- [329] E. Tanyildizi, F. Ö.-I. Access, and undefined 2019, *A new chaotic S-box generation method using parameter optimization of one dimensional chaotic maps*, .
- [330] I. Batra and S. Ghosh, *A Novel Approach of Congestion Management in Deregulated Power System Using an Advanced and Intelligently Trained Twin Extremity Chaotic Map Adaptive Particle Swarm Optimization Algorithm*, **Arab.**

J. Sci. Eng., 44:8 (2019) 6861–6886.

- [331] G. Kaur, S. A. C. D. and Engineering, and undefined 2018, *Chaotic whale optimization algorithm*, .
- [332] E. Avaro and ~ Glu, *Pseudorandom number generator based on Arnold cat map and statistical analysis*, .
- [333] M. Jaberipour, E. K. N. S. and N. Simulation, and undefined 2010, *Two improved harmony search algorithms for solving engineering optimization problems*, .
- [334] M. Li, H. Zhao, X. Weng, T. H.-A. in E. Software, and undefined 2016, *A novel nature-inspired algorithm for optimization: Virus colony search*, .
- [335] S. Mirjalili, A. L.-A. in engineering software, and undefined 2016, *The whale optimization algorithm*, .
- [336] D. Maucourt-Boulch, C. de Martel, S. Franceschi, and M. Plummer, *Fraction and incidence of liver cancer attributable to hepatitis B and C viruses worldwide*, **Int. J. Cancer**, 142:12 (2018) 2471–2477.
- [337] G. L. Davis *et al.*, *Hepatocellular Carcinoma: Management of an Increasingly Common Problem*, **Taylor Fr.**, 21:3 (2008) 266–280.
- [338] C. Gadiparthi, E. Yoo, V. Are, P. Charilaou, D. K.-A. of hepatology, and undefined 2019, *Hepatocellular carcinoma is leading in cancer-related disease burden among hospitalized baby boomers*, .
- [339] S. Li *et al.*, *Structure convolutional extreme learning machine and case-based shape template for HCC nucleus segmentation*, .
- [340] M. Attwa, S. E.-E.-W. journal of hepatology, and undefined 2015, *Guide for diagnosis and treatment of hepatocellular carcinoma*, .
- [341] I. K.-A. I. in medicine and undefined 2001, *Machine learning for medical diagnosis: history, state of the art and perspective*, .
- [342] A. Das, U. Acharya, S. Panda, S. S.-C. S. Research, and undefined 2019, *Deep learning based liver cancer detection using watershed transform and Gaussian mixture model techniques*, .
- [343] M. Iavarone, M. C.-C. in liver disease, and undefined 2013, *HBV infection and hepatocellular carcinoma*, .
- [344] D. DeWaal, V. Nogueira, A. Terry, ... K. P.-N., and undefined 2018, *Hexokinase-2 depletion inhibits glycolysis and induces oxidative phosphorylation in hepatocellular carcinoma and sensitizes to metformin*, .
- [345] S. Singh, P. P. Singh, L. R. Roberts, and W. Sanchez, Chemopreventive strategies in hepatocellular carcinoma, *Nature Reviews Gastroenterology and Hepatology*, 11, :1. Nature Publishing Group, pp. 45–54, 13-Jan-2014.
- [346] M. Santos, P. Abreu, ... P. G.-L.-J. of biomedical, and undefined 2015, *A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients*, .
- [347] W. Książek, M. Abdar, U. Acharya, P. P.-C. S. Research, and undefined 2019, *A novel machine learning approach for early detection of hepatocellular carcinoma patients*, .
- [348] R. Sawhney, P. Mathur, and R. Shankar, *A firefly algorithm based wrapper-*

- penalty feature selection method for cancer diagnosis*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2018) pp.438–449.
- [349] S. Mitra, S. Saha, S. A.-E. S. with Applications, and undefined 2018, *Fusion of stability and multi-objective optimization for solving cancer tissue classification problem*, .
 - [350] S. Liu, S. Liu, W. Cai, S. Pujol, ... R. K.-2014 I. 11th, and undefined 2014, *Early diagnosis of Alzheimer's disease with deep learning*, .
 - [351] B. Fa et al., *Pathway-based biomarker identification with crosstalk analysis for robust prognosis prediction in hepatocellular carcinoma*, .
 - [352] K. Chaudhary, O. B. Poirion, L. Lu, and L. X. Garmire, *Statistics in CCR Deep Learning-Based Multi-Omics Integration Robustly Predicts Survival in Liver Cancer*, (2018).
 - [353] J. TAN, M. UNG, C. CHENG, and C. S. GREENE, *UNSUPERVISED FEATURE CONSTRUCTION AND KNOWLEDGE EXTRACTION FROM GENOME-WIDE ASSAYS OF BREAST CANCER WITH DENOISING AUTOENCODERS*, *Biocomputing 2015*, (2014) pp.132–143.
 - [354] L. Chen, C. Cai, V. Chen, and X. Lu, *Learning a hierarchical representation of the yeast transcriptomic machinery using an autoencoder model*, **BMC Bioinformatics**, 17:1 (2016).
 - [355] M. Palazzo, P. Beausery, and P. Yankilevich, *Hepatocellular Carcinoma tumor stage classification and gene selection using machine learning models*, .
 - [356] S. Dogan and I. Turkoglu, *DIAGNOSING HYPERLIPIDEMIA USING ASSOCIATION RULES*, **Math. Comput. Appl.**, 13:3 (2008).
 - [357] O. "Fatih and Zyurt, *A fused CNN model for WBC detection with MRMR feature selection and extreme learning machine*, **Soft Comput.**, 24.
 - [358] E. Aydemir, T. Tuncer, S. D.-M. Hypotheses, and undefined 2020, *A Tunable-Q wavelet transform and quadruple symmetric pattern based EEG signal classification method*, .
 - [359] T. Tuncer, S. Dogan, P. Pławiak, U. A.-K.-B. Systems, and undefined 2019, *Automated arrhythmia detection using novel hexadecimal local pattern and multilevel wavelet transform with ECG signals*, .
 - [360] T. Tuncer, S. D.-A. Acoustics, and undefined 2019, *A novel octopus based Parkinson's disease and gender recognition method using vowels*, .
 - [361] T. Tuncer, F. E.-P. A. S. M. and its Applications, and undefined 2020, *Neighborhood component analysis and reliefF based survival recognition methods for Hepatocellular carcinoma*, .
 - [362] T. Shaikhina, D. Lowe, S. Daga, ... D. B.-... S. P. and, and undefined 2019, *Decision tree and random forest models for outcome prediction in antibody incompatible kidney transplantation*, .
 - [363] W. Ibrahim and M. S. Abadeh, *Protein fold recognition using Deep Kernelized Extreme Learning Machine and linear discriminant analysis*, **Neural Comput. Appl.**, 31:8 (2019) 4201–4214.
 - [364] K. Gyamfi, J. Brusey, A. Hunt, E. G.- Neurocomputing, and undefined 2019, *A*

- dynamic linear model for heteroscedastic LDA under class imbalance*, .
- [365] S. D. Desai, S. Giraddi, P. Narayankar, N. R. Pudakalakatti, and S. Sulegaon, *Decease Detection in Plant leaves using Deep Learning View project Reconstruction of image from incomplete medical data View project Back-Propagation Neural Network Versus Logistic Regression in Heart Disease Classification*, 702 (2019) 133–144.
 - [366] H. Alshamlan, G. Badr, and Y. Alohal, *Microarray Gene Selection and Cancer Classification Method Using Artificial Bee Colony and SVM Algorithms (ABC-SVM)*, Lecture Notes in Electrical Engineering, (2019) pp.575–584.
 - [367] S. Ayyad, A. Saleh, L. L.- Biosystems, and undefined 2019, *Gene expression cancer classification using modified K-Nearest Neighbors technique*, .
 - [368] G. D.- Ecology and undefined 2007, *Boosted trees for ecological modeling and prediction*, **Wiley Online Libr.**, .
 - [369] I. Allegretta, B. Marangoni, P. Manzari, C. P.- Talanta, and undefined 2020, *Macro-classification of meteorites by portable energy dispersive X-ray fluorescence spectroscopy (pED-XRF), principal component analysis (PCA) and*, .
 - [370] T. K. Ho, *Nearest neighbors in random subspaces*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (1998) pp.640–648.
 - [371] S. M. Mirjalili, S. Z. Mirjalili, S. Saremi, and S. Mirjalili, Sine cosine algorithm: Theory, literature review, and application in designing bend photonic crystal waveguides, *Studies in Computational Intelligence*, 811. Springer Verlag, pp. 201–217, 2020.
 - [372] S. Mirjalili, S. Mirjalili, A. L.-A. in engineering software, and undefined 2014, *Grey wolf optimizer*, .

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı Fahrettin Burak DEMİR
Doğum Yeri- Tarihi Elazığ-1988
Adres Malatya Turgut Özal Üniversitesi Doğanşehir Vahap Küçük
MYO Bilgisayar Teknolojileri Bölümü / Doğanşehir /
MALATYA
E-posta fahrettin.demir@ozal.edu.tr

EĞİTİM BİLGİLERİ

Doktora İnönü Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar
Mühendisliği Anabilim Dalı (2015-2020)
Yüksek Lisans Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve
Bilgisayar Eğitimi Anabilim Dalı (2011-2014)
Lisans Fırat Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve
Bilgisayar Öğretmenliği (2007-2011)
Lise Elazığ Anadolu Lisesi (2002-2006)

MESLEKİ DENEYİMLER

İnönü Üniversitesi, Sürgü MYO, Bilgisayar Teknolojileri
Bölümü, Öğretim Görevlisi, 2013-2018
Üniversite Malatya Turgut Özal Üniversitesi, Doğanşehir Vahap Küçük
MYO, Bilgisayar Teknolojileri Bölümü, Öğretim Görevlisi,
2018-Devam Ediyor

Tezden Türetilen Yayınlar

1- Uluslararası hakemli dergilerde yayınlanan makaleler (SCI)

A1. Demir, F. B., Tuncer, T., & Kocamaz, A. F. (2020). A chaotic optimization method based on logistic-sine map for numerical function optimization. *Neural Computing and Applications*, 1-13.

A2. Demir, F. B., Tuncer, T., Kocamaz, A. F., & Ertam, F. (2020). A survival classification method for hepatocellular carcinoma patients with chaotic Darcy optimization method based feature selection. *Medical Hypotheses*, 139, 109626.

2- Ulusal ulakbimde ve dergi parkta taranan dergilerde yapılan alışmalar

B1. Demir, F. B., Tuncer, T., & Kocamaz, A. F. (2019). Lojistik-Gauss Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Yöntemi. *Anatolian Science-Bilgisayar Bilimleri Dergisi*, 4(1), 47-53.

3- Uluslararası Bilimsel Toplantılarda Sunulan ve Bildiri Kitabında Basılan Bildiriler

C1. Demir, F. B., Tuncer, T., & Kocamaz, A. F. (2019, September). Lojistik-Singer Harita Tabanlı Yeni Bir Kaotik Sürü Optimizasyon Yöntemi. 2019, *International Artificial Intelligence and Data Processing Symposium (IDAP)* (pp. 1-5). IEEE.

