



## Bioinspiration-based deep learning algorithm

Mehmet Akif Çifçi<sup>1,2,3\*</sup>, Peren Jerfi Canatalay<sup>4</sup>, Emrah Arslan<sup>5</sup>, Samina Kausar<sup>6</sup>

<sup>1</sup>The Institute of Computer Technology, Tu Wien University, 1040 Vienna, Austria

<sup>2</sup>Department of Computer Engineering, Bandırma Onyedi Eylül University, 10200 Balıkesir, Türkiye

<sup>3</sup>Engineering and Informatics Department, Klaipėdos Valstybinė Kolegija/Higher Education Institution, 92294 Klaipėda, Lithuania

<sup>4</sup>Department of Computer Engineering, Engineering, and Natural Sciences Faculty, Istinye University, 34000 Istanbul, Türkiye

<sup>5</sup>Computer Engineering, Engineering of Faculty, İstanbul Aydın University, 34000 İstanbul, Türkiye

<sup>6</sup>Shanghai University, Shanghai, China University of Kotli Azad Jammu and Kashmir, Kotli Azad Kashmir, Pakistan

### Highlights:

- Innovative optimization algorithm inspired by an epidemiological model
- Parameter optimization enhanced with deep learning methods
- Ability to provide high-performance solutions for engineering and healthcare challenges

### Keywords:

- SIMO neural learning
- Optimisation algorithms
- Engineering design optimisation
- Metaheuristics, deep learning

### Article Info:

Research Article

Received: 22.01.2024

Accepted: 04.08.2024

### DOI:

10.17341/gazimmfd.1424002

### Correspondence:

Author: Mehmet Akif Çifçi

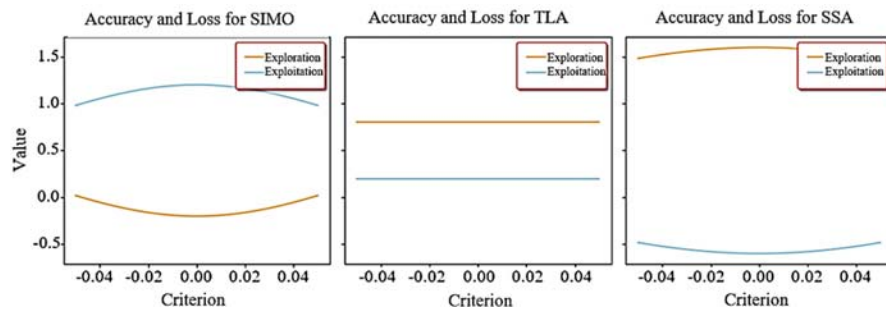
e-mail:

akif.cifci@tuwien.ac.at

phone: +90 532 066 4035

### Graphical/Tabular Abstract

This research introduces the use of ML techniques to determine optimal parameter-value configurations for metaheuristic optimization algorithms. To evaluate this approach, a new metaheuristic algorithm is proposed with the acronym SIMO. SIMO aims to improve parameter selection in the optimization process using a combination of metaheuristic algorithms and ML. First, the LSTM model is used to generate and enrich the initial population of the search space. Furthermore, a parameter grid is populated with candidate values corresponding to six SIMO-parameter schemes. These values are then permutationally combined to train the SIMO algorithm and generate image representations of the optimized solution space. This experimental phase is critical to identify unique features in image representations using ESA and to learn the optimal parameter-value representations to fully train the SIMO algorithm. Experiments were conducted to show and compare the convergence of individuals in the solution space and the results are shown in Figure A.



**Figure A.** Exploration and exploitation plots for ABC, AO, AOA, ArchOA, EOSA, FFA, IWO, MA, SFO, SIMO, SSA and TLO using the C14 function

**Purpose:** SIMO aims to improve parameter selection in the optimization process using a combination of metaheuristic algorithms and ML.

**Theory and Methods:** Infection-Sensitive Artificial Intelligence Optimization Model (SIMO), an innovative learned heuristic inspired by biological systems and Deep Learning (DL) techniques. The SIMO optimization algorithm is inspired by the epidemiological partitioning model with Infection-Sensitive Artificial Intelligence to estimate the susceptibility of the population to infection, active infections and the recovering population at any point in time. SIMO integrates the IA method into the initialization method and parameter tuning components to improve the search process, allowing it to exhibit intelligent and autonomous behavior.

**Results:** After selecting the best parameter-value configuration, SIMO is fully trained and tested using CEC functions. Comparative discussions and analyses are carried out to evaluate SIMO's performance against similar methods. The results show that the proposed approach achieves the best solutions in all benchmarking functions. Furthermore, by analyzing SIMO's exploratory search process graph and convergence during the optimization phase, the method demonstrates its competitive performance against current optimization methods. The findings of the study emphasize that the proposed approach eliminates the dependence of similar algorithms on commonly used stochastic methods and queries a random method for assigning values to control parameters. This is noteworthy with the potential of utilizing SL techniques to accurately identify the best parameter values within a dataset

**Conclusion:** As a suggestion for future work, the use of additional standard benchmarking functions such as Ackley, Griewank, Bent Cigar, Brown and Alpine can facilitate the exploration of parameter-value configurations for similar optimization methods. Furthermore, the robustness of the SIMO algorithm can be applied to complex real-world optimization problems in various computational domains with more different benchmarking functions such as CEC 2011 and CEC 2014.



## Biyoinspirasyon tabanlı derin öğrenme algoritması

Mehmet Akif Çifçi<sup>1,2,3\*</sup>, Peren Jerfi Canatalay<sup>4</sup>, Emrah Arslan<sup>5</sup>, Samina Kausar<sup>6</sup>

<sup>1</sup>The Institute of Computer Technology, Tu Wien University, 1040 Vienna, Austria

<sup>2</sup>Mühendislik ve Doğa Bilimleri Fakültesi, Bilgisayar Mühendisliği, Bandırma Onyedü Eylül Üniversitesi, 10200 Balıkesir, Türkiye

<sup>3</sup>Engineering and Informatics Dept, Klaipėdos Valstybinė Kolegijs/Higher Education Institution, Lithuania

<sup>4</sup>Mühendislik ve Doğa Bilimleri Fakültesi, Bilgisayar Mühendisliği, İstinye Üniversitesi, 34000 İstanbul, Türkiye

<sup>5</sup>Mühendislik Fakültesi, Bilgisayar Mühendisliği, İstanbul Aydın Üniversitesi, 34000 İstanbul, Türkiye

<sup>6</sup>Shanghai University, Shanghai, China University of Kotli Azad Jammu and Kashmir, Kotli Azad Kashmir, Pakistan

### Ö N E Ç I K A N L A R

- Epidemiyolojik modelden esinlenen yenilikçi optimizasyon algoritması
- Derin öğrenme yöntemleri ile güçlendirilmiş parametre optimizasyonu
- Mühendislik ve sağlık sorunlarına yüksek performanslı çözümler sunma yeteneği

#### Makale Bilgileri

Araştırma Makalesi

Geliş: 22.01.2024

Kabul: 04.08.2024

#### DOI:

10.17341/gazimmfd.1424002

#### Anahtar Kelimeler:

SIMO nöral öğrenme;  
optimizasyon algoritmaları,  
mühendislik tasarım  
optimizasyonu,  
metaheuristik,  
derin öğrenme

#### ÖZ

Bu makale, biyolojik sistemler ve Derin Öğrenme (DÖ) tekniklerinden esinlenen yenilikçi bir öğrenilmiş sezgisel yöntem olan Enfeksiyona Duyarlı Yapay Zekâ Optimizasyon Modeli (SIMO) işlenmektedir. SIMO optimizasyon algoritması, Enfeksiyona Duyarlı Yapay Zekâ ile epidemiyolojik bölme modelinden ilham alarak herhangi bir zamandaki nüfusun enfeksiyona duyarlılığını, aktif enfeksiyonları ve iyileşen popülasyonu tahmin etmektedir. SIMO, arama sürecini iyileştirmek amacıyla başlatma yöntemi ve parametre ayarlama bileşenlerine DÖ metodunu entegre eder, bu sayede zeki ve otonom davranış sergileyebilmektedir. DÖ entegrasyonu, algoritmanın etkin, etkili ve güçlü arama sonuçlarına yönlendirilmesine olanak tanıyan nöral modellere dayalı başlangıç çözümleri üretmeyi kolaylaştırmaktadır. Bu yaklaşım, algoritmanın performansını üst düzey çözümler elde ederek, daha hızlı bir şekilde yakınsamasını sağlayarak, güçlülüğünü artırarak ve hesaplama gereksinimlerini azaltarak geliştirir. SIMO algoritmasının etkinliğini doğrulamak için 2017 IEEE Evrimsel Hesaplama Kongresi (CEC 2017) benchmarking fonksiyonlarından alınan iki veri seti kullanılmıştır ve deneysel sonuçlar yenilikçi algoritmalarla karşılaştırılmıştır. Detaylı karşılaştırmalar, SIMO'nun birçok benzer modeli geride bıraktığını, daha az kontrol parametresi kullanarak yüksek performanslı çözümler sunduğunu göstermektedir. Ayrıca, SIMO'nun performansı gerçek hayat problemlerine uyarlanmıştır. Sonuçlar, SIMO'ya öğrenme sürecini entegre etmenin, mevcut literatürdeki diğer optimizasyon yaklaşımlarına kıyasla üstün hassasiyet ve hesaplama verimliliği sağladığını açıkça göstermektedir.

## Bioinspiration-based deep learning algorithm

### H I G H L I G H T S

- Innovative optimization algorithm inspired by an epidemiological model
- Parameter optimization enhanced with deep learning methods
- Ability to provide high-performance solutions for engineering and healthcare challenges

#### Article Info

Research Article

Received: 22.01.2024

Accepted: 04.08.2024

#### DOI:

10.17341/gazimmfd.1424002

#### Keywords:

SIMO neural learning;  
optimization algorithms,  
engineering design  
optimization,  
metaheuristics,  
deep learning

#### ABSTRACT

This paper presents the Infection Susceptible Artificial Intelligence Optimization Model (SIMO, susceptible-infected-removed model optimizer), an innovative learned heuristic inspired by biological systems and Deep Learning (DL) techniques. The SIMO optimization algorithm estimates the susceptibility of the population to infection, active infections and the recovering population at any point in time, inspired by the epidemiological partition model with Infection-Sensitive Artificial Intelligence. SIMO integrates the IA method into the initialisation method and parameter tuning components to improve the search process, so that it can exhibit intelligent and autonomous behaviour. The integration of the IO facilitates the generation of initial solutions based on neural models, which allows the algorithm to be guided towards efficient, effective and robust search results. This approach improves the performance of the algorithm by obtaining high-level solutions, allowing it to converge faster, increasing its robustness and reducing its computational requirements. Two datasets from the 2017 IEEE Congress on Evolutionary Computing (CEC 2017) benchmarking functions are used to validate the effectiveness of the SIMO algorithm and the experimental results are compared with innovative algorithms. Detailed comparisons show that SIMO outperforms many similar models, offering high performance solutions using fewer control parameters. Furthermore, the performance of SIMO is adapted to real-life problems. The results clearly show that integrating the learning process into SIMO provides superior accuracy and computational efficiency compared to other optimization approaches in the existing literature.

\*Sorumlu Yazar/Yazarlar / Corresponding Author/Authors : \*akif.cifci@tuwien.ac.at, mcifci@bandirma.edu.tr, meh.cifci@kvk.lt, peren.canatalay@istinye.edu.tr, emraha@stu.aydin.edu.tr, samnamalik7@yahoo.com / Tel: +90 532 066 4035

## 1. Giriş (Introduction)

Son yıllarda, sayısal ve kombinatoriyal optimizasyon problemlerini çözmek amacıyla güçlü, etkili ve verimli global arama tekniklerinin geliştirilmesine dair çabaların belirgin bir şekilde arttığı gözlemlenmektedir [1-3]. Ancak, gerçek dünya optimizasyon problemlerinin karmaşıklığı, endüstriyel süreçlerdeki ilerlemeler ve toplumsal evrimle birlikte pek çok yeni sorunu beraberinde getirmiştir. Bu durum, mevcut optimizasyon teknikleri, özellikle klasik yöntemler için bir zorluk oluşturmaktadır [4]. Amaçları günlük hayatta problem çözme olsa da literatürde bu optimizasyon tekniklerinin pek çok kısıtlama ile karşılaştığı anlaşılmaktadır [5, 6]. Bununla birlikte, optimizasyon algoritmaları, bu zorluklar ile başa çıkmak için yeni tasarım varyantları ve optimizasyon tekniklerinden yararlanmaktadır [7, 8]. Metaheuristik algoritmalar, aday optimizasyon problemleri için kesin veya tam çözümler üretmemelerine rağmen tercih edilen optimizasyon tekniği olarak popülerlik kazanmıştır [9]. Bu algoritmalar yaklaşık çözümler sağlasa da probleme özgü bilgi veya tekniklerin gerekliliği, çözüm odaklı optimalite eksikliği, teorik veya matematik temelinin bulunmaması, çoklu arama parametrelerine bağımlılık ve aynı problem için farklı çözümlere yol açan stokastik arama süreçleri gibi sorunlar mevcuttur. Ancak, metaheuristik tekniklerin gerçek dünya problemlerine pratik uygulanabilirlik sağlamada önemli bir alternatif olduğu aşikardır [10]. Doğadan esinlenen metaheuristik algoritmalar, içlerinde buldukları doğanın rastgelelik ve zorlu optimizasyon görevlerine karşı kara kutu yaklaşımlarını göz önüne aldığında, çeşitli problemleri çözmek için güçlü araçlar olarak öne çıkmıştır [11, 12]. Bu algoritmalar, mühendislik tasarım problemlerinden tıbbi görüntü segmentasyonu ve sınıflandırmasına, paralel makine zamanlamasından kutu paketleme problemlerine kadar birçok alanda uygulanmıştır.

Son çalışmalar, Makine Öğrenimi (MÖ) tekniklerinin, çözüm süreci boyunca üretilen verilerden kullanışlı bilgiler çıkarmak suretiyle metaheuristik algoritmalara katkıda bulunabileceğini göstermektedir [13]. Bilginin arama sürecine entegre edilmesi, metaheuristik algoritmaların daha bilinçli kararlar almasına öncülük etmekte ve bu entegrasyon, zekâ düzeyini artırarak çözüm kalitesini, yakınsama hızını ve sağlamlığı önemli ölçüde iyileştirmektedir. Bu durum, metaheuristik algoritmaların performansını artırma potansiyelini taşımaktadır.

Bu çalışmada, doğadan esinlenmiş bir optimizasyon algoritması olan ve Enfeksiyona Duyarlı Yapay Zekâ Optimizasyon Modeli (SIMO) algoritması işlenmektedir [14]. Algoritma, hastalık kontrolünde Enfeksiyona Duyarlı Yapay Zekâ (SIM) modelin başarılı uygulamasından ilham alınarak tasarlanmıştır. SIMO algoritması, duyarlı (S), enfekte (I) ve iyileşmiş (M) olmak üzere hastalık yayılımını üç evresini işler. Bu algoritma, başlatma ve parametre konfigürasyon ayarlarını kolaylaştırmak için önemli bir MÖ bileşeni olan akıllı başlatma özelliğine sahiptir. Bu özellik, algoritmanın başlangıçta yüksek kaliteli çözümler bulmasına yardımcı olarak, zeki bir arama sürecini yönlendirmektedir.

Bu çalışma, önerilen SIMO algoritmasının tasarımını, matematik modelini, çeşitli benchmarking testlerinde elde edilen sonuçları ve gerçek dünya problemlerine uygulanması sonucunda elde edilen performansı ayrıntılı bir şekilde sunmaktadır. Önerilen SIMO algoritması, metaheuristik optimizasyon alanına önemli katkılar sağlamaktadır. Bu katkılar şöyle sıralanabilir:

*Epidemiyolojik Model İlhamı:* SIMO, hastalık kontrolünde kullanılan başarılı bir SIM epidemiyolojik modelinden esinlenerek tasarlanmıştır. Bu yaklaşım, algoritmanın etkili ve doğru arama davranışlarını modellemesine olanak tanır.

*MÖ Entegrasyonu:* SIMO, başlatma yöntemi ve parametre ayarlamalarında MÖ kavramlarını kullanarak, algoritmanın başlangıçta yüksek kaliteli çözümleri seçmesini sağlar. Bu, zeki ve otonom bir arama sürecini mümkün kılar.

*Esnek ve Güçlü Uygulanabilirlik:* SIMO, çeşitli sayısal test fonksiyonları ve mühendislik kullanım durumları üzerinde değerlendirilmiş ve geniş bir uygulanabilirlik yelpazesi sunmuştur. Bu özellik, algoritmanın farklı problemler üzerinde etkili bir şekilde kullanılabilmesini sağlar.

*Performans ve Hız:* Yapılan kapsamlı karşılaştırmalar, SIMO'nun birçok benzer modeli geride bıraktığını, daha az kontrol parametresi kullanarak yüksek performanslı çözümler sunduğunu göstermektedir. Ayrıca, önerilen algoritmanın hızlı yakınsama sağladığı ve hesaplama verimliliğini artırdığı vurgulanmıştır.

*Vurgulanan önemli noktalar şöyle sıralanabilir:*

*Doğadan İlham Alınan Tasarım:* SIMO'nun SIM modeline dayalı tasarımı, algoritmanın gerçek dünya problemlerinde etkili bir şekilde çalışabilmesini sağlar.

*MÖ ile Güçlendirilmiş Başlangıç Çözümleri:* Algoritmanın başlangıçta MÖ ile elde edilen yüksek kaliteli çözümleri seçmesi, hızlı yakınsama ve üstün performans sağlar.

*Geniş Uygulanabilirlik:* SIMO'nun çeşitli test fonksiyonları ve mühendislik problemleri üzerinde başarılı bir şekilde test edilmiş olması, algoritmanın geniş bir uygulama yelpazesi sunma potansiyelini vurgular.

*Optimizasyon Topluluğuna Yenilikçi Bir Yaklaşım:* SIMO'nun getirdiği yeni bakış açısı, optimizasyon topluluğuna çeşitli problemlerde etkili ve güçlü bir metaheuristik sunma potansiyeli taşımaktadır.

Çalışmanın geri kalan kısmı şu şekilde işlenmiştir. Bölüm 2, mevcut en son teknoloji metaheuristik yöntemleri ve optimizasyon algoritmalarının tasarım ve uygulamasında MÖ kullanımını incelemektedir. Bölüm 3, Önerilen algoritmanın esinlendiği SIMO algoritmasının matematik modelini ve tasarımını ayrıntılı bir şekilde ele alır. Bölüm 4, SIMO algoritmasının çeşitli benchmarking fonksiyonlarına uygulandığında elde edilen nitel ve nicel sonuçları sunmaktadır. Bölüm 5, SIMO algoritmasının gerçek dünya problemlerine uygulanması işlenmektedir. Son olarak, Bölüm 6, çalışmayı özetleyerek gelecek araştırma yönergeleri için önerilerde bulunarak sona ermektedir.

## 2. Literatür Çalışması (Related Works)

Bu bölüm, mevcut en son teknoloji metaheuristik algoritmaların kapsamlı bir incelemesini sunmaktadır. Bu inceleme, algoritmaların güçlüklerini, sınırlamalarını ve çeşitli optimizasyon problemlerine uygulanabilirliklerini ortaya koymaktadır. Ardından, MÖ ve DÖ modellerini metaheuristik algoritmalarla birleştiren hibrit yöntemlerin pratik uygulamaları üzerine bir tartışma işlenmektedir. Bu hibrit yaklaşımlar, MÖ tekniklerinin gücünü metaheuristik algoritmaların performansını ve verimliliğini artırmak amacıyla kullanılmaktadır. Bu bölüm, MÖ veya DÖ modellerini başarıyla metaheuristik algoritmalarla entegre eden dikkate değer çalışmaları vurgulamaktadır [15].

### 2.1. Metaheuristik Optimizasyon Yöntemleri (Metaheuristic Optimization Methods)

Chen vd., [16] çalışmasına göre, doğadan esinlenen metaheuristik algoritmalar, evrim, sürü, fizik ve insan davranışı temelli metaheuristik optimizasyon yöntemlerini kapsamaktadır. Evrim temelli metaheuristik algoritmalar, doğal evrimden esinlenerek, doğal

seçim ve tür göçü gibi kavramları içermektedir [17]. Genetik Algoritma (GA), 1975 yılında Holland tarafından önerilen yaygın ve popüler bir evrim temelli metaheuristik algoritmadır; problem çözümlerini optimize etmek amacıyla Darwinist doğal seçimi simüle etmektedir [18]. GA'nın ilk uygulamalarından biri, 8.000 kural içeren bir çalışan sınıflandırıcı sisteminin optimize edilmesinde kullanılmıştır [19]. Başka bir tanınmış evrim temelli metaheuristik algoritma olan Diferansiyel Evrim (DE), 1996 yılında Storn ve Price tarafından tanıtılmıştır [20]. DE, sayısal optimizasyon için tasarlanmış basit bir evrimsel algoritmadır ve vektörleri ağırlıklı rastgele vektör farkları ekleyerek değiştirme işlemi içermektedir [21]. DE'nin ilk uygulamaları, çoklu-modlu fonksiyonları optimize etmeye odaklanmıştır [22]. Sürü temelli metaheuristik algoritmalar hayvanların, bitkilerin ve diğer organizmaların zeki davranışlarından esinlenerek geliştirilen, sürü özelliklerini sergileyen algoritmalarlardır. Bu algoritmalar, doğadaki sosyal yaratıklardaki gözlemlenen öz-düzenleme özelliklerini taklit ederler. Bu nedenle, geniş bir doğadan esinlenmiş metaheuristik algoritmalar yelpazesinde oldukça önemli bir yer tutmaktadır [23]. En popüler sürü temelli metaheuristik algoritmalarından bir tanesi, kuş sürülerinin dinamiklerini taklit eden parçacık sürü optimizasyonu (PSO)'dur [24]. Bir başka bilinen sürü temelli metaheuristik algoritma olan Guguk Kuşu Algoritması (CS), belirli yuva parazitliği gösteren bazı bildiren türlerine dayanmaktadır [25]. CS, guguk kuşlarının üreme ve Levy uçuş davranışlarını taklit ederek çalışır [26]. Bu davranış, karmaşık optimizasyon problemlerinde PSO ve GA'ya kıyasla üstün bir performans sergilemesine neden olmaktadır. Diğer bir sürü temelli metaheuristik örnek, ateşböceği (FA) algoritmasıdır, bu da tropikal ateşböceklerinin flaş ışığı özellikleri ve etkileşim desenlerine dayanarak sosyal davranışlarını model almaktadır [27]. Balina optimizasyon algoritması (WOA), diğer bir popüler sürü temelli metaheuristik olup balina sırtındaki balon ağının avlanma stratejilerinden esinlenmiştir [28]. Fizik temelli metaheuristik algoritmalar, fiziksel veya kimyasal olgulardan esinlenerek geliştirilen bir algoritma türüdür. Bu algoritmalar, yerçekimi kuvveti, manyetik kuvvet, iklim ve kimyasal olaylar, termodinamik, optik ve mekanik gibi olguları modelleyerek, arama uzayında verimli bir şekilde gezinmeyi ve en iyi çözümü bulmaya çalışırlar [29]. Popüler fizik temelli metaheuristik algoritmalarından biri, Rashedi vd., [30] tarafından önerilen yerçekimi arama algoritması (GSA), kütle etkileşimi dinamiklerini dikkate alan bir sezgisel optimizasyon algoritmasıdır. GSA, yerçekimi yasaları ve Newton'un ikinci hareket yasası kullanılarak formüle edilmiştir [31]. Fizik temelli metaheuristik algoritmaların bir diğer temsilcisi, Henry'nin gaz yasasına dayanan Henry gaz çözünürlüğü optimizatörü (HGSO)'dur. HGSO'da, her bir gazın ilişkilendirilmiş bir kısmı basıncı vardır [32]. HGSO, gaz moleküllerinin kümeleme davranışını taklit eder ve keşif süreçlerini dengelemek için kullanılır. Başka bir algoritma ise Zhao vd., [33] tarafından önerilen atom arama optimizasyonu (ASO) ve Yadav vd., [34] tarafından önerilen yapay elektrik alan algoritmasından (AEFA) esinlenmiştir.

İnsan faaliyeti temelli metaheuristik algoritmalar, çeşitli insan sosyal, duygusal ve kültürel özelliklerinden esinlenir ve insan yaşam tarzını yansıtan algoritmalar [35]. Bu kategorideki geniş çapta tanınan bir popülasyon tabanlı algoritma, Rao vd., [36] tarafından önerilen öğrenme-öğretme tabanlı optimizasyon (TLBO)'dur. TLBO, bir öğretmenin bir öğrenme ortamında eğitilen üzerindeki etkisini model alır. Özellikle, bir çözüm sınıfını temsil eden çözümler veya bir öğrenme sınıfı kullanarak global çözüme doğru yaklaşırlar. TLBO algoritması, mekanik tasarım optimizasyon problemlerini çözmekte üstünlüğünü kanıtlamıştır. Bu sınıftaki diğer önemli bir algoritma, Reynolds vd., [37] tarafından işlenen kültürel algoritma (CA) [38]. CA, meta-seviye etkileşim ve sürüleşme modellerini kullanarak bilgi kaynaklarını oluşturur ve inanç uzayında "bilgi kümeleri" oluşturur. Bu kategorideki diğer insan faaliyeti temelli metaheuristik algoritmalar arasında, Kaveh vd., [39] tarafından önerilen halat çekme

optimizasyonu (TWO) ve Zhang vd. [40] tarafından önerilen sıra arama (QSA) algoritmasıdır.

## 2.2. Metaheuristik Algoritmasının uygulamasında MÖ kullanımı (Use of ML in the application of the Metaheuristic Algorithm)

Cadenas vd. [41], optimizasyon problemlerini çözmek için merkezi bir hibrit metaheuristik iş birliği stratejisi önermiştir. Bu strateji, performansı artırmak için hibrit iş birliği stratejisinin veya metaheuristik sistemin bir düzeyinde zekâ entegre etmektedir. Araştırmacılar, bireysel metaheuristikler tarafından döndürülen sonuç kayıtlarına uygulanan bir bilgi çıkarma sürecinden elde edilen kurallar ve modeller kümesi aracılığıyla bilgi entegre ederek optimizasyon sürecini başarabilmiştir. Bu bilgi, bireysel metaheuristik algoritmalarca bulunamayan çözümleri bulmak için kullanılabilir. Yaklaşımlarının etkinliği, bireysel metaheuristik algoritmalar tarafından elde edilen sonuçların, işbirlikçi stratejilere dayanan diğer mevcut yöntemlerle karşılaştırıldığı bir dizi hesaplama deneyi kullanılarak test edilmiştir. Deneyler, önerilen stratejinin diğer yöntemlere kıyasla daha iyi performans gösterdiğini kanıtlamıştır [42]. Özellikle, önerilen strateji, kooperatif olmayan yöntemlerden %20 daha iyi performans göstermiştir ve işbirlikçi yöntemlerden %10 daha iyi performans göstermiştir. Bu, önerilen stratejinin, optimizasyon problemlerini çözmek için etkili bir yöntem olduğunu göstermektedir.

Choong vd. [43] tarafından geliştirilen optimizasyon yöntemi, takviyeli öğrenme tekniklerine dayalı otomatik hiper-heuristik bir yöntemdir. Bu yöntem, hiper-heuristik modelin yüksek seviyeli sezgisel tasarımını otomatik olarak oluşturur. Q-learning tekniği, modelin optimizasyon sürecinin farklı aşamalarında uygun bileşenleri seçmesine öncülük eder. Araştırmacılar, yöntemlerinin performansını Hiper-heuristik Flexible Framework'un altı problem alanından çeşitli örnekler kullanarak değerlendirmiştir. Yapılan deneyler, yöntemlerinin mevcut literatürdeki en iyi performans gösteren hiper-heuristik modellerle rekabetçi olduğunu ortaya koymuştur. Sonuç olarak, bu araştırmacıların yöntemi, takviyeli öğrenme tekniklerini kullanarak hiper-heuristik modellerin tasarımını otomatikleştirmek için etkili bir yaklaşım sunmaktadır. Yöntem, çeşitli problem alanlarında rekabetçi bir performans sergilemektedir.

Binaların ve enerji sistemlerinin işletme optimizasyonu için lineerleştirme yaklaşımlarının yüksek maliyetini göz önünde bulundurarak, bu alandaki çoğu lineer modelin sınırlı hesaplama kaynakları içinde yüksek maliyetli modelleme maliyetleri çektiği genel bir bilgi olarak bilinmektedir. Bu doğrultuda, Ikeda ve Nagai [44], bina enerji sistemlerinin günlük işletme programlarını optimize etmek için metaheuristik ve MÖ tekniğinin içeren hibrit bir algoritma önermiştir [45]. Makine karakteristiklerinin doğrusallığına odaklanan bu çalışmada, derin sinir ağı MÖ tekniği, bütünlümlü soğutma kulesi sistemlerinin optimal işlemlerini tahmin etmek için kullanılmıştır; diğer bileşenlerin işlemleri ise metaheuristik algoritmalarla optimize edilmiştir. Deneysel sonuçlara dayanarak, araştırmacılar uyguladıkları yöntemin günlük işletme maliyetini %13,4'ten fazla azaltabileceğini öngörmüşlerdir [46].

Calvet vd., [47] ise metaheuristik algoritmalar MÖ yöntemleriyle birleştiren çeşitli mevcut yöntemleri incelemiş ve bir tür hibrit algoritma olan "learnheuristics" kavramını tanıtmıştır. Araştırmacılar, önerdikleri learnheuristics'in, model amaç fonksiyonu veya kısıtlama setinde bulunan dinamik giriş öğeleriyle kombinatoriyal optimizasyon problemlerini çözmek için kullanılabilirliğini iddia etmişlerdir. Araştırmacılara göre, bu giriş öğeleri genellikle önceden sabitlenmemiştir. Çözüm, bazı sezgisel tabanlı yinelenen bir süreç tarafından kısmen oluşturulduğu için, bu giriş öğeleri mevcut çözüm konfigürasyonuna bağlı olarak farklı değerlere sahiptir. Bu nedenle,

bu yaklaşım, öngörüler kullanarak yüksek kaliteli bir çözüm oluşturmak için MÖ tekniklerini benimser ve aynı zamanda learnheuristics olarak bilinen bir yapıcı sezgisel kullanılmaktadır [48]. Deneysel sonuçlar, önerilen yaklaşımın rekabetçiliğini ve etkinliğini ortaya koymaktadır.

### 3. Uygulanan SIMO Modeli (The Implementation of SIMO)

Bu bölüm, önerilen SIMO algoritmasının genel bir bakışını sunmaktadır. Bu genel bakış, SIMO'nun ilham kaynağını ve optimizasyon modelleme tasarım adımlarını içermektedir. Ayrıca, bölüm, SIMO'nun çeşitli yönlerinin düzgün bir şekilde sunulması için iki ana alt bölüme ayrılmıştır. İlk olarak, SIMO'nun popüler SIM modelinden aldığı ilham tartışılmıştır; ardından SIMO'nun uygulandığı ayrıntılı bir şekilde açıklanmıştır.

#### 3.1 Veri Seti (Data Set)

Kullanılan veri seti, biyolojik sistemlerden ve derin öğrenme tekniklerinden esinlenen yenilikçi bir öğrenilmiş sezgisel yöntem olan enfeksiyona duyarlı yapay zekâ optimizasyon modeli (SIMO) için eğitilmiş ve test edilmiştir. Bu veri seti, açık kaynak olarak yayınlanan 2017 IEEE Evrimsel Hesaplama Kongresi (CEC 2017) benchmarking fonksiyonlarından alınmış iki veri setini içermektedir. Elde edilen deneysel sonuçlar, SIMO'nun performansını literatürdeki mevcut algoritmalarla karşılaştırmak için kullanılmıştır.

2017 IEEE Evrimsel Hesaplama Kongresi (CEC 2017) benchmarking fonksiyonlarından elde edilen iki ayrı veri seti kullanılmıştır. Bu veri setleri:

- 1. Fonksiyon Değerleri:** Farklı matematiksel fonksiyonların değerlerini içerir. Bu fonksiyonlar, optimizasyon algoritmalarının performansını değerlendirmek için yaygın olarak kullanılan standart test fonksiyonlarıdır. Her bir fonksiyon, çeşitli parametre değerleri üzerinde çalıştırılarak elde edilen çıktılar içerir.
- 2. Parametreler:** Fonksiyonların parametre değerlerini içerir. Bu parametreler, her bir fonksiyonun davranışını ve performansını belirleyen değişkenlerdir. Örneğin, Sphere fonksiyonu için  $x_1$  ve  $x_2$  gibi parametreler bulunabilir.

Bu veri setleri, enfeksiyona duyarlı yapay zekâ optimizasyon modeli (SIMO) için eğitilmiş ve test edilmiştir. SIMO, biyolojik sistemlerden ve derin öğrenme tekniklerinden esinlenen yenilikçi bir öğrenilmiş sezgisel yöntemdir.

Veri setinin temel özellikleri şunlardır:

- Çeşitli matematiksel fonksiyonlar için çeşitli parametre değerleri içerir. Örnek fonksiyonlar arasında Sphere, Rastrigin ve Rosenbrock bulunabilir.
- Her bir fonksiyon için farklı değer aralıkları ve optimum noktalar vardır.
- SIMO modelinin performansını değerlendirmek için kullanılmıştır. Bu, modelin farklı fonksiyonlar üzerinde ne kadar iyi performans gösterdiğini belirlemek için yapılmıştır.
- Açık kaynak olarak yayınlanmıştır ve araştırmacılar tarafından kullanılabilir. Bu, veri setinin yeniden kullanılabilir ve çoğaltılabilir olduğu anlamına gelir.

Tablo 1'de veri setinin bir örneği bulunmaktadır:

**Tablo 1.** Veri Seti Örneği (Dataset example)

Fonksiyon	Parametre	Değer
Sphere	$x_1, x_2$	0,1234, 0,5678
Rastrigin	$x_1, x_2$	-1,2345, 2,5678
Rosenbrock	$x_1, x_2$	3,4567, 8,9012

### 3.2. Enfeksiyona Duyarlı Yapay Zekâ Optimizasyon Modeli (Infection Sensitive Artificial Intelligence Optimization Model)

Bu çalışmada önerilen metaheuristik algoritma, hastalığı modelleme konusunda gerçekçi bir teknik haline gelen klasik SIM modelinden ilham alınarak tasarlanmıştır. Hastalığın yayılmasında oluşturulan alt grupların anlayışından türetilmiştir: bu alt gruplara atanmış bireylerin, duyarlılık, enfeksiyon ve çıkarma şanslarının eşit olduğu varsayılmaktadır. Bu model, genellikle bir toplum veya nüfus içinde hastalığın yayılımını gösteren bir dizi değişken ve bölmeden oluşmaktadır. SIM temel ve klasik modeli sunsa da literatürde hastalık modelleme sürecini daha da zenginleştirmek için daha ilginç bölme önerileri bulunmaktadır. Bu modelin örnekleri, SIMO ve duyarlı enfekte duyarlı (SIS) gibi SIM modelinin varyasyonlarıdır, bunlar da salgınları modellemede araştırılmıştır. Duyarlı, enfekte ve iyileşmiş bölümleri modelleyen üç farklı bağımsız non-lineer fonksiyonu içeren bir tek sistemle (ODE) kullanılarak temsil edilmektedir [49].

Bazı durumlarda, iyileşen bireyler bağışıklık kaybı durumunda duyarlı gruba geri dönebilir. Bu durum, Şekil 2'de gösterilmektedir. Diğer modelleme durumlarında ise iyileşen bireyler artık duyarlı grupta dikkate alınmaz. Bu durum, Şekil 1'de gösterilmektedir. SIM modelinin modellenmesini etkileyen bir başka varsayım ise nüfusun sabit olduğu ve nüfusa yeni doğum izin verilmediğidir. Bu varsayım, modelin daha basit ve daha kolay çözülebilmesini sağlamaktadır. Temel SIM modeli, nüfus üzerindeki değişiklikleri tasvir eden bir dizi ordinary differential equation (ODE) sistemi ile temsil edilir. Bu sistemdeki denklemler Eş.1, Eş. 2 ve Eş. 3'te gösterilmiştir:

$$\frac{ds}{dt} = -\beta SI \quad (1)$$

$$\frac{dI}{dt} = \beta SI - \gamma I \quad (2)$$

$$\frac{dM}{dt} = \gamma I \quad (3)$$

Bu model, her enfekte bireyin birim zamanda  $S(t)$  teması olduğunu ve  $I(t)$ 'nin toplam nüfus büyüklüğü  $R(t)$  ile bağımsız olduğunu varsaymaktadır. Bu temaslardan  $\kappa S/N$  duyarlı bireylerle gerçekleşmektedir. Yeterli temasın  $\tau$  kesiri ile bulaşmaya neden olduğu varsayıldığında, her enfekte birey, birim zamanda  $\kappa\tau S/N$  duyarlı bireyi enfekte etmektedir. Bu nedenle,  $\beta = \kappa\tau/N$  şeklinde,  $b = \kappa\tau$  olarak ifade edilmektedir. Parametre  $\tau$ , bulaşıcı hastalıkların bulaşması olarak adlandırılır. Temel değişkenler şöyle sıralanmıştır.

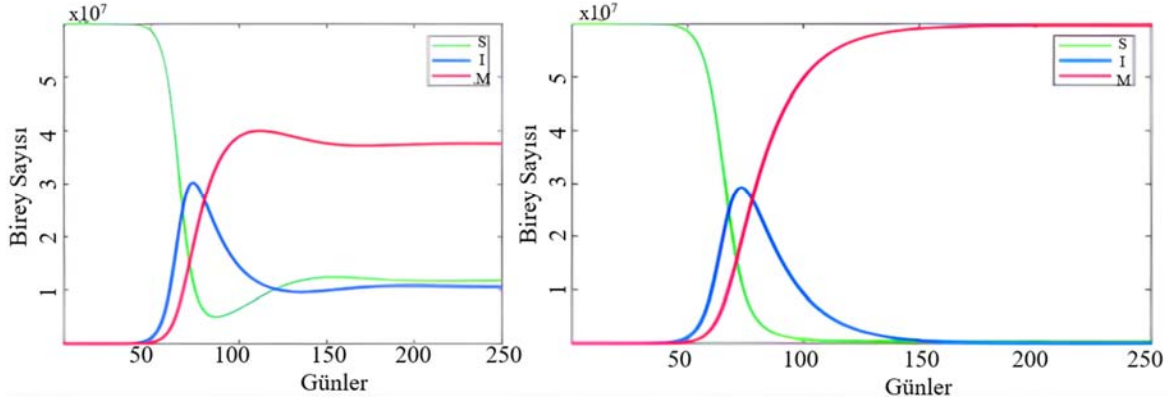
$S(t)$ : Duyarlı (duyarlı) bireylerin sayısı,  
 $I(t)$ : Enfekte (hasta) bireylerin sayısı,  
 $R(t)$ : İyileşen veya bağışıklık kazanan bireylerin sayısı.

Bu temel değişkenlerle ilişkilendirilen parametreler:

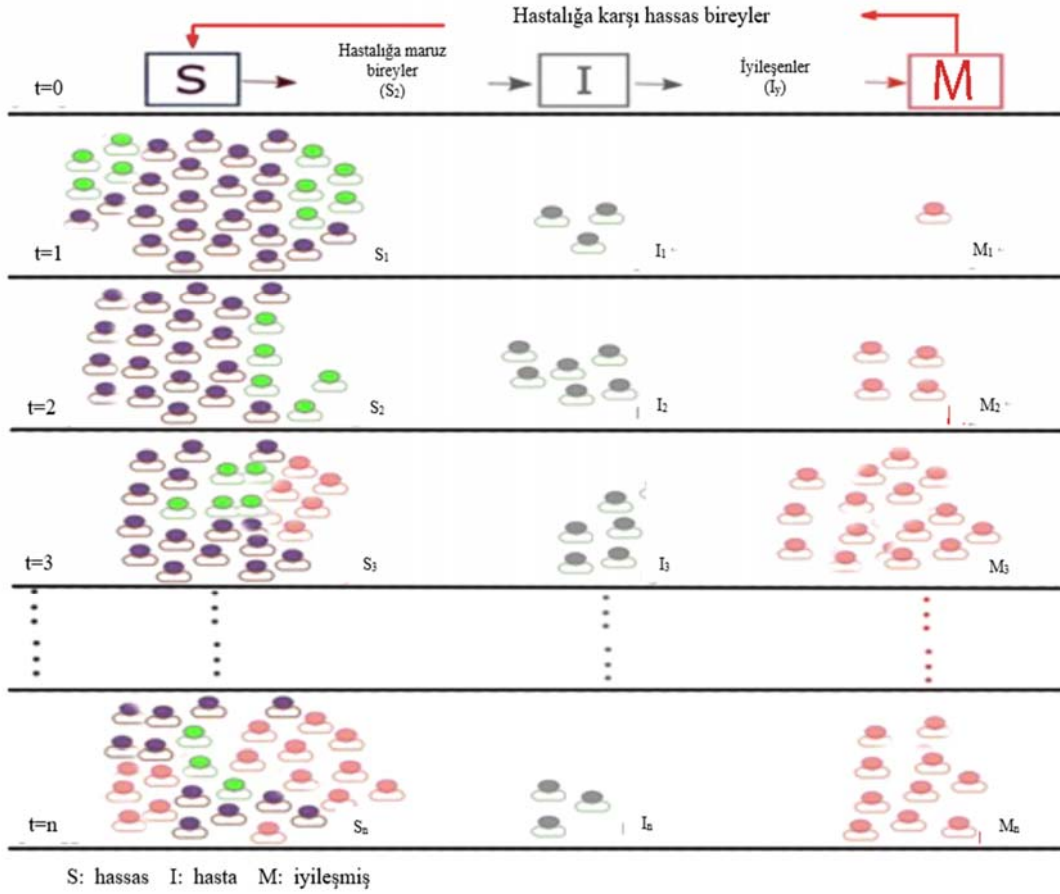
$\kappa$ : Her enfekte bireyin birim zamanda temas ettiği birey sayısını temsil etmektedir,  
 $N$ : Toplam nüfus büyüklüğünü temsil etmektedir,  
 $\tau$ : Yeterli temasın bulaşmaya neden olduğu kesiri temsil etmektedir,  
 $\beta$ : Bulaşma oranını temsil etmektedir,  
 $b$ : Enfekte bir bireyin birim zamanda duyarlı bireyi enfekte etme oranını temsil etmektedir.

Bu bileşenler, modelin matematiksel ifadesiyle birlikte hastalık modelleme sürecini daha anlaşılır ve bütünlük taşıyan bir şekilde tanımlamaktadır.

SIM modelinin hastalık yayılma ve yayılma modelleme konusunda etkili bir araç olduğunu gösteren geniş bir başarı raporlarına



Şekil 1. Temel SIM model gösterimi  $\beta = 0,7/50000$ ,  $\gamma = 1/5$ ,  $S(0) = 49955$ ,  $I(0) = 5$ ,  $R(0) = 0$   
(Basic SIM model representation  $\beta=0.7/50000$ ,  $\gamma=1/5$ ,  $S(0)=49955$ ,  $I(0)=5$ ,  $R(0)=0$ )



Şekil 2. Optimization problemlerini çözme sürecinde SIM modelinin gösterimi  
(Representation of SIM model in the process of solving optimization problems)

dayanarak, bu çalışma, klasik SIM modeline dayalı yeni bir metaheuristik algoritma önermektedir. Bu önerilen metaheuristik algoritma, SIM modelinin temel prensiplerine dayanmaktadır ve hastalığın yayılma dinamiklerini daha ayrıntılı bir şekilde analiz etmek üzere tasarlanmıştır. Bu bağlamda, çalışmanın amacı, SIM modelini temel olarak hastalık yayılma süreçlerini daha etkili bir şekilde optimize etmek ve analiz etmek üzere tasarlanmış bir metaheuristik algoritma sunmaktır. Önerilen algoritma, SIM modelinin temel değişkenleri ve parametreleri üzerinde çeşitli optimizasyon teknikleri uygulayarak, salgın kontrol stratejilerini

geliştirmek ve hastalık yayılma modellerine daha iyi uyum parametre setlerini belirlemek için kullanılmaktadır. Bu yaklaşımın temelinde, önceki çalışmalarda olduğu gibi SIM modeli temel alınmakta, ancak bu modelin analizi ve optimizasyonu için yeni bir bakış açısı sunulmaktadır. Çalışmanın sonuçları, önerilen metaheuristik algoritmanın hastalık modellerinin daha hassas bir şekilde ayarlanmasına ve kontrol stratejilerinin daha etkili bir şekilde uygulanmasına olanak tanıdığını göstermektedir. Bu şekilde, SIM modeline dayalı metaheuristik algoritma, hastalık salgınlarının daha iyi anlaşılmasına ve kontrol edilmesine katkıda bulunabilir.

### 3.3. SIMO Algoritma Modellemesi (SIMO Algorithm Modelling)

Bu bölümde, önerilen SIMO algoritmasının tamamlanmış prosedürünü açıklayan tasarım ve model formülasyonu tartışılmaktadır. SIMO'nun optimizasyon süreci, önce üç bölümden oluşan bir SIM tabanlı model kullanılarak açıklanır. İkinci olarak, SIMO yönteminin matematiksel modeli, nüfus başlangıcından bölme güncellemesine kadar olan konseptten çıkararak tasarlanmıştır. Ayrıca, bu bölümde SIMO'nun algoritmik temsilini açıklayan bir yabancı dil program kodu ve algoritmanın karmaşıklığının analizi de sunulmaktadır. Ek olarak, nüfusu iyileştirmek ve parametre kombinasyonunu optimize etmek için kullanılan sinir öğrenme yöntemi de açıklanmaktadır. Bu çalışmada uygulanan SIM modeli, hastalık yayılmasını modelleme için klasik modeli takip eder, böylece gereksiz bölümler ortadan kaldırılır. Şekil 3'te, SIM modelinin ve optimizasyon sürecinin birleşik temsili gösterilmektedir. Model tarafından yakalanan üç bölme, duyarlı (S), enfekte (I) ve iyileşmiş (M) bireyleri içerir. SIM modelinin başlangıç nüfusu, bütün bulaşıcı hastalıkların enfekte nüfusu oluşturmak için duyarlı nüfusa hedef aldığından, S bölgesine atanır. Klasik SIM modelinin genel doğası göz önüne alındığında, her hastalığın hastalık yayılma modeli, bireylerin I bölgesine atanmasını belirler. Bu çalışmada, bireylerin S'den I'ye geçişi, belirli bir oranda hastalığa maruz kalma koşullarına bağlıdır ve bu oran  $S\pi$  tarafından gösterilir. Bağışıklık, aşılama, hastaneye yatış, tedavi ve kendi kendine iyileşme gibi koşullar, bireyleri genellikle I bölgesinden R bölgesine yeniden atar. Bireylerin R'ye geçişi,  $I_g$  sırasını takip eder, bu da hastalıktan iyileşmeyi simgeler. Bir noktada, R'deki bireyler tekrar S'ye geçer, bu da onları duyarlı bireyler haline getirir ve bağışıklığın bulaşmayı hafifletmediği sürece I'ye alınabilirler.

Bir hastalık salgınının toplam süresi  $T$ ,  $t_1, t_2, \dots, t_n$  olarak gösterilebilir ve şekilde  $t=0, t=1, t=2, \dots, t=n$  olarak temsil edilir, burada  $t_0$  veya  $t=0$ , belirli bir nüfusta hastalık salgınının ne zaman rapor edildiğini gösteren bir gösterimdir. Bu çalışmadaki SIM modeli için, P'nin nüfus büyüklüğünün konstant  $c$  olarak tutulduğu varsayılmaktadır, bu da doğum veya ölümün  $T$  süresi tamamlanana kadar donmuş olaylar olduğu anlamına gelir. Bu, nüfusun büyüklüğünü sabit tutar, bu da bireylerin S, I ve R'de bulunanları içeren toplamı içerir. Bu nedenle, SIM modellerinin optimizasyon süreci şöyle takip eder:  $t=0$  veya  $t=0$ 'da P'deki tüm bireyler S'de temsil edilir ( $size(P)=size(S)$ ).  $t=1$  zamanında, S'deki bazı bireyler hastalığa maruz kalmış ve şimdi virüs tarafından enfekte olabilir durumdadır, bu da hastalığa yol açan bir süre sonrasında meydana gelir, ki bu genellikle hastalıkların doğasıyla özgüdür.  $t=2$ 'de, S'deki maruz kalan bireylerin şimdi enfekte olduğu ve I bölgesine yeniden atanmış oldukları gözlemlenebilir. Bu sırada, aynı zamanda, enfekte vakalardan birinin iyileştiği ve şimdi R bölgesine atanmış olduğu bildirilmiştir.  $T$  arttıkça, bireylerin bölme içindeki yeniden atanması, her süreç örneğinin S, I ve R'ye farklı desenlerle tahsis edilmesiyle dinamik bir süreç oluşturur. Optimizasyon süreci, her bireyin sürekli anatomik değişiklikleri ile devam eder; mutasyon ve arama alanındaki yer değiştirmesi, kullanışlı bir arama deseni sunar. Tablo 2, SIM modelini açıklamak için kullanılan tüm parametreleri ve bunların karşılık gelen tanımlamalarını özetlemektedir. Enfeksiyon oranı  $\pi$ , iyileşme oranı  $g$ , enfekte bireylerle temas oranı  $\beta$  ve bir popülasyonun doğal ölüm oranı  $G$  dikkate alınmıştır.

**Tablo 2.** SIM Model Parametreleri ve Tanımlamaları (SIM Model Parameters and Definitions)

Semboller	Açıklamalar
$\pi$	Enfeksiyon oranı
$\gamma$	İyileşme oranı
$\Gamma$	Hastalığa bağlı ölüm oranı
$\tau$	Doğal ölüm oranı
$\beta_1$	Enfekte kişilerle temas oranı
$\beta_2$	İyileşmiş kişilerle temas oranı

Bu çalışmada önerilen klasik SIM modelinin genel doğası göz önüne alındığında, her bir parametre için belirli değerler belirlemek uygun değildir. Ayrıca, parametrelere değer atamak için stokastik modeller kullanmak, literatürde yaygın olarak kullanılan popüler bir yaklaşımı önerir. Bu çalışmada, önerilen SIMO algoritması için kullanılan tüm parametrelerin seçimi ve kombinasyonu için yeni bir MÖ tabanlı model önerdik. Bu arada, algoritmanın matematiksel modeli önce sunulmuş ve ardından öğrenmeye dayalı yöntemin açıklaması da aşağıdaki paragraflarda tartışılmıştır.

SIMO modelinin tüm popülasyonu, herhangi bir zaman  $t$ 'de S, I ve R içindeki tüm bireylerin toplamını veren Eş. 4'te temsil edilir:

$$P = S + I + M \quad (4)$$

S, I ve R bölümlerine bireylerin oluşturulması, sırasıyla Eş. 5, Eş. 6 ve Eş. 7'de yakalanan üç alt modeli içeren bir sistem diferansiyel denklemin bağımlıdır.

$$S_t = \frac{\partial S(t)}{\partial t} = S\pi - I\beta \quad (5)$$

$$I_t = \frac{\partial I(t)}{\partial t} = I + S\beta - I_g \quad (6)$$

$$M_t = \frac{\partial M(t)}{\partial t} = M + gI \quad (7)$$

Burada  $S_t, I_t$  ve  $M_t$ , sırasıyla S, I ve M bölümlerine tahsis edilen birey sayılarını temsil eder, burada  $t_i$ , zamanındadır. Bu bölmelere yapılan güncellemeler, eğitim veya optimizasyon sürecinin her iterasyon aşamasında gerçekleşir. Her güncelleme olduğunda, bireylerin farklı bölümlere yeniden atanmış olduğu anlamına gelir ve bu nedenle bireylerin bir bölümlerle ilişkilendirilmiş bir operasyona göre mutasyona uğradığı bir durum söz konusudur. Popülasyon başlatma aşamasında, tüm bireylere aşağıda açıklanan belirli bir kompozisyon tahsis edilir.

### 3.4 Temel ve Sinir Ağı Tabanlı Başlatma Yöntemleri (Basic and Neural Network-Based Initialization Methods)

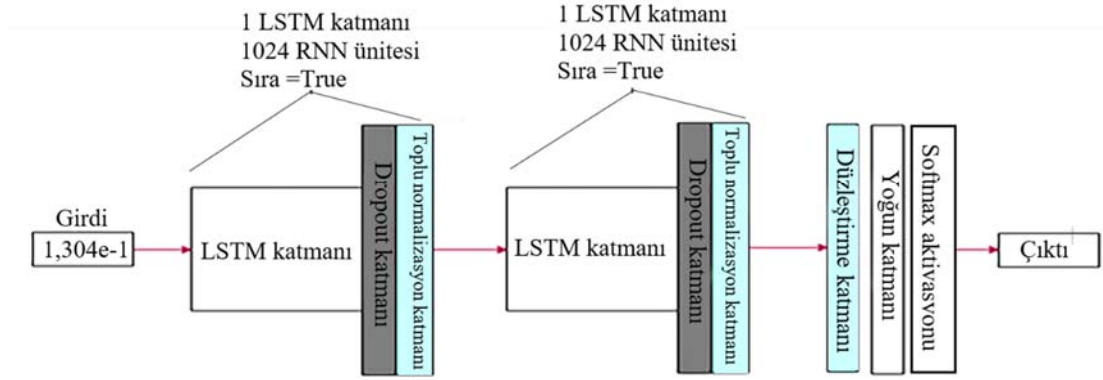
Bu çalışmada, popülasyon başlatma ve popülasyon tabanlı optimizasyon algoritmaları için temsil, genellikle stokastik bir yaklaşımı varsayar veya literatürde raporlanan bazı güncel yöntemleri kullanır. Bu çerçevede, evrimsel bir yaklaşıma dayanan bir stokastik popülasyon başlatma yöntemi geliştirilmiştir. İlk aşamada, popülasyonun temsili için matematiksel bir model kullanılmıştır. Bu model, matris temsili kullanılarak iki boyutlu bir yaklaşımı içermekte ve bu matris, optimizasyon probleminin boyutuna denk gelen 'd' sayısına sahiptir (Eş. 8). Bu temsil, başlangıçta popülasyonun yapısını oluşturmak için temel alınır.

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,d-1} & x_{1,d} \\ x_{2,1} & x_{2,2} & \dots & x_{2,d-1} & x_{2,d} \\ \vdots & \vdots & & x_{i,j} & \vdots \\ x_{n,1} & x_{n,2} & \dots & x_{n,d-1} & x_{n,d} \end{bmatrix} \quad (8)$$

Eğer belirli bir  $x_i$  elemanı için Eş. 9 tarafından temsil edilen stokastik yöntemi uyguluyorsak, bu hesaplama X kümesindeki her  $x_i$  için gerçekleştirilir. Bu süreçte genellikle optimizasyon problemlerinde kullanılan üst (U) ve alt (L) sınırlar da dikkate alınır.

$$x_i = L + rand(0,1) * (U - L) \quad (9)$$

X içinde elde edilen sonuç verisi, ilgili biyoloji temelli ve hastalık yayılımı optimizasyon algoritmasını eğitmek için kullanıldı. Bu veriler, önce belirli bir çözüm uzayında potansiyel bir başlangıç çözümü üretebilen uygun bir DÖ modeline uygulandı. Eğitim görevi



**Şekil 3.** SIMO için uygun ve optimal başlangıç çözümler üretmek için kullanılan LSTM mimarisi (LSTM architecture used to generate suitable and optimal initial solutions for SIMO)

için tekrarlayan sinir ağı (RNN) ve genellikle DÖ olarak sınıflandırılan uzun kısa vadeli bellek (LSTM) gibi modeller düşünüldü. LSTM'nin mimari temsili Şekil 4'te gösterilmiştir. Bu model, elde edilen çözümlerin trajedisini düzenleyebilir ve önerilen SIMO algoritmasını eğitmek için kullanılacak potansiyel bir başlangıç çözümü üretebilir.

DÖ mimarisi, her biri bir dropout katmanı ve toplu-normalizasyonu katmanı tarafından takip edilen iki LSTM katmanından oluşur. Modeldeki girişler, biyolojik temelli optimizasyon algoritmalarının eğitimi sırasında elde edilen bazı en iyi çözümleri temsil eder. Her LSTM katmanının yapısı, 1024 RNN biriminden oluşur ve eğitilen modelin bir üreteç olarak görev yapabilmesine izin vermek için sıralama kullanımına olanak tanır. İki blok LSTM katmanının ardından mimari, düzeltme ve yoğun katmanlarla genişletilmiştir, bu katmanlar genellikle LSTM katmanlarında çıkarılan tüm özellikleri düzleme veya vektörleştirme için kullanılır. SoftMax aktivasyon fonksiyonu, vektörleştirilmiş özelliklere uygulanır, ardından modelden bir çıktı elde edilir. Eğitilmiş model, SIMO için kullanılacak olan MÖ başlangıç çözüm uzayını temsil eden benzer çıktıları üreten bir üreteç olarak kullanılır. Uygun aktivasyon fonksiyonu ile çalıştırılmak üzere, sırasıyla 4096 ve 512 düğümden oluşan yoğun katmanlar; çıktı katmanının elde edilmesi için P değerleri %20 ve %50 olarak belirlenmiştir.

Üreteç G'den gelen çıktı R, işlenmeye veya ayrıştırmaya ihtiyaç duyan bir örnek ham çözümdür ve bunun kullanılabilirliği için arama alanı olarak kullanılmasına izin vermek için bazı işlemlere tabi tutulur. Bu üreteç ve ayrıştırıcı, Eş. 10'da gösterildiği gibi, dim (P) SIMO algoritması tarafından gereken popülasyonun boyutunu ve boyutunu belirtir:

$$R = \text{parse}(G(X, \text{dim}(P))) \quad (10)$$

Ayrıştırılmış başlangıç çözümü daha sonra Eş. 11'de S'ye yerleştirilen bireyleri üretmek için kullanılır:

$$S = \{s_i \in R^{\mathcal{F}d} \mid 0 \leq i \leq \text{dim}(P)\} \quad (11)$$

Burada  $\mathcal{F}$ , R üzerinde saat yönünde bir integral işlemi yapan N-ary operatördür ve d, popülasyondaki önceki S(i-1)'den sonraki  $s_i$  bireyinin başlangıcını ve sonunu gösteren bir ayıraçtır. Her  $s_i$ 'nin problemin alan sınırları içinde tutulmasını sağlamak için Eş. 12, tüm  $s_i$ 'leri düzeltmek için uygulanır, burada ub ve lb sırasıyla üst ve alt sınırları temsil eder:

$$S = \{s_i \in [\text{amend}(S, i)]_{lb}^{ub} \mid 0 \leq i \leq \text{dim}(P)\} \quad (12)$$

Optimizasyon sürecindeki iterasyonlar sırasında, global en iyi  $s_{best}$  bireyi, Eş. 13 kullanılarak elde edilir. Bu,  $s_{best}$  önceki olarak elde edilen değer ile karşılaştırılır:

$$s_{best} = \begin{cases} s_{best}, \text{fits}(s_{cbest}) < \text{fits}(s_{best}) \\ s_{cbest}, \text{fits}(s_{cbest}) \geq \text{fits}(s_{best}) \end{cases} \quad (13)$$

Her  $s_i$ 'nin arama uzayındaki konumu, bireyin hastalık süper bulaşma noktasına ulaştığında kullanılır, bu tür bir birey enfekte olduğunda. Sonuç olarak, her iterasyonun başında, I içindeki bireylerin mevcut konumları hesaplanır ve algoritmanın yoğunlaştırma ve keşif mekanizmasının ne zaman gerçekleşeceğini belirlemek için kullanılır. Eş. 14 ve Eş. 15'te, I'deki her bireyin konumunun hesaplanması gösterilmektedir:

$$lpos_i^{t+1} = lpos_i^t + \text{rand}(0, 1) * \rho \quad (14)$$

$$M(s) = \text{lrate} * \text{rand}(0, 1) + M(\text{Ind}_{best}) \quad (15)$$

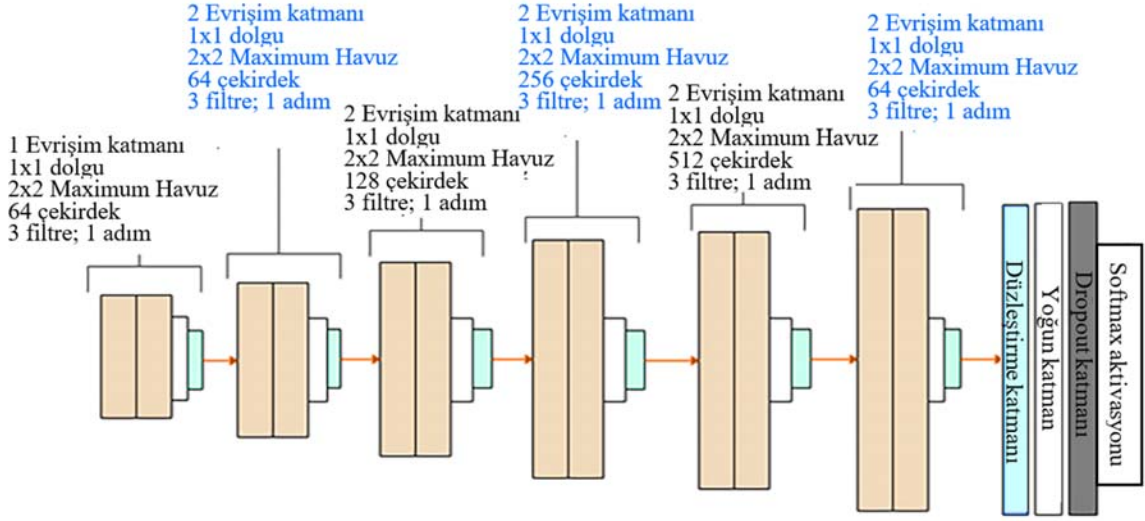
Burada  $\rho$ , bir bireyin yer değiştirmesinin ölçek faktörünü temsil eder;  $lpos_i^{t+1}$  ve  $lpos_i^t$  sırasıyla zaman t ve t+1'de güncellenen ve orijinal konumu temsil eder ve  $\text{rand}(0,1)$ , 0 ile 1 arasında rastgele bir değer üretir.  $lpos_i^{t+1}$  için elde edilen değer,  $lpos_i^{t+1} < 0,5$  ise srate'e, aksi takdirde lrate'e atanır.

### 3.5. SIMO Sinir Ağı Tabanlı Parametre Seçimi (SIMO Neural Network Based Parameter Selection)

Diğer ilgili yöntemlerle karşılaştırıldığında, SIMO kontrol ve optimizasyon süreci için az sayıda parametre kullanır. Bu avantajı daha da ileri taşımak için, bu çalışma, en iyi performans elde etmek için gereken en iyi parametre değer konfigürasyonunu zekice bir şekilde birleştiren ve seçen yeni bir DÖ yaklaşımı önermektedir. Bu amaçla, bir dizi parametre değeri konfigürasyonu oluşturulur ve bu konfigürasyonlar bir dizi benchmarking fonksiyonunu optimize etmek için uygulanır, böylece her konfigürasyon için SIMO'nun görüntü performansı, DÖ modeli için bir veri kümesi olarak toplanır.

Evrişimli Sinir Ağı (ESA) mimarisi, altı blok konvolüsyon-havuzlama biriminden oluşur. Her bir blok, iki konvolüsyon katmanı, bir sıfır dolgu katmanı ve bir maksimum havuzlama katmanı içerir. Mimarideki tüm konvolüsyon katmanları için 3x3 filtre boyutu kullanılır, ancak her konvolüsyon-havuzlama bloğu için filtre sayısı  $2^k$  formülünü takip eder, burada k [50] arasında değişir. Mimaride düzeltme katmanı, Dense katmanı ve Dropout katmanı kullanılarak tamamlanır ve sınıflandırma amaçları için SoftMax aktivasyon fonksiyonu kullanılır.





**Şekil 4.** SIMO'nun en iyi konfigürasyonu ve parametre değerlerinin en iyi kombinasyonu  
(The best configuration of SIMO and the best combination of parameter values)

Bir ESA mimarisi kullanılarak, bu veri kümeleri modelin eğitimi için girdi olarak sağlanır, böylece eğitilmiş model, en iyi SIMO algoritmasının en iyi performansını elde etmek için uygun olan en iyi konfigürasyonu tahmin etmek için kullanılabilir [51-52]. Eş. 16'da, deneyimle ilgili kullanılan konfigürasyon havuzunu oluşturan eşleme fonksiyonunun modellenmesi açıklanmıştır:

$$pv(\pi, g, \beta, G) = pv : \{p_1, p_2, p_3, p_4\} \rightarrow \{v_1, v_2, v, v_4\} \quad (16)$$

Burada,  $pv$ , çıktısı SIMO parametrelerinin benzersiz kombinasyonlarından oluşan  $C$  kümesi olan bir eşleme fonksiyonunu temsil eder.  $C$ 'nin elemanları,  $\{c \in pv(\pi, g, \beta, G)_i \mid 0 \leq i \leq N\}$  olarak açıklanır. Her  $c$ , parametrelerin bir kombinasyonuna ve bunların karşılık gelen üretilmiş değerlerine eşdeğerdir, böylece bu parametre-değer eşleme kısmi eğitim SIMO için uygulandığında grafiklerin yakınsaması üretilir ve bu, ESA 'in eğitimi için bir veri kümesi olarak toplanır. Bu eğitim için kullanılan ESA 'in mimarisi Şekil 5'te gösterilmiştir, tamamen eğitilmiş ağ daha sonra hangi en iyi  $c_i$ 'nin tam eğitim SIMO için uygun olduğunu tahmin etmek için kullanılır.

### 3.6. SIMO Algoritmaları ve Hesaplama Karmaşıklığı (SIMO Algorithms and Computational Complexity)

Önerilen SIMO metaheuristik algoritmasının sözdizimi, Tablo 3'te sunulmaktadır. Algoritma, önceki paragraflarda açıklanan tüm prosedürleri listeler ve iteratif süreçte optimizasyonun nasıl gerçekleştirildiğini özetler. Algoritmanın girdileri, modeli eğitmek için gereken iterasyon sayısı, popülasyon boyutu  $N$  ve SIMO'yu eğitmek için kullanılacak en iyi  $c_i$ 'yi elde etmek için ESA modelini eğitmek için gereken veri kümesini içerir. Beklenen çıkış, her iterasyon için elde edilen tüm en iyi çözümlerin kombinasyonunu ve ayrıca nihai global en iyiyi içerir. Algoritmanın 1. satırı,  $S$ ,  $I$  ve  $R$  bölmelerinin başlatılmasını ve en iyi çözümleri depolamak için bir konteynerin oluşturulmasını gösterir. LSTM modelinin SIMO'yu eğitmek için en iyi başlangıç çözümünü üretmesi için, stokastik olarak üretilen başlangıç çözümü  $X$ , LSTM modelini eğitmek için giriş olarak sağlanır. Ardından eğitilmiş LSTM modeli, SIMO için öğrenilmiş çözüm uzayını tam bir eğitim için üretir. Bu sırada, 5-7. satırlar, SIMO'nun tüm parametrelerini kullanmak için uygun en iyi konfigürasyonu üretmek amacıyla ESA modelinin uygulanmasını gösterir.  $S$ 'deki tüm bireylerin uygunluk değerleri elde edilir ve global en iyi çözüm 8-9. satırlarda atanır. SIMO algoritmasının iteratif

eğitimi, 10-31. satırlarda listelenmiştir ve dönüş değerleri 32. satırda bulunmaktadır. 1. bölmesine birey üretme ve mutasyonu, 12-21. satırlarda görülürken, üç bölmedeki bireylerin güncellenmesi, 22-25. satırlarda sergilenmiştir. Son olarak, her iterasyon için mevcut ve global en iyi çözümün güncellenmesi, 26-30. satırlarda raporlanmıştır.

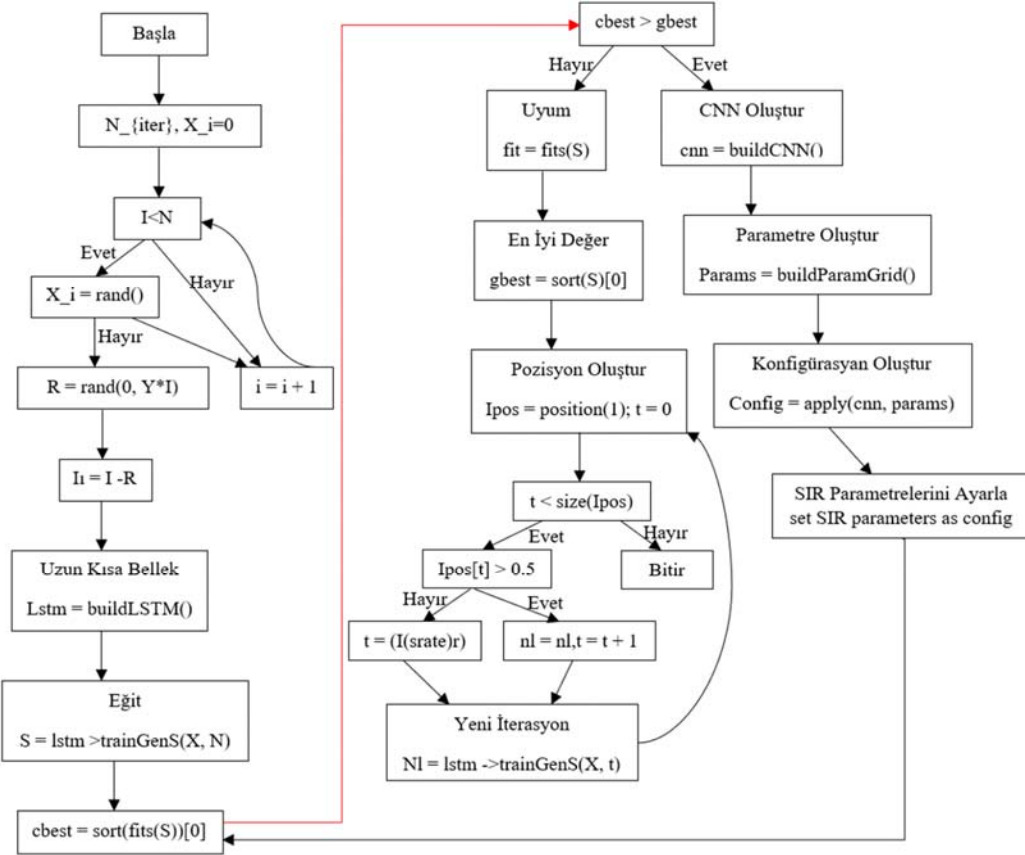
**Tablo 3.** SIMO metaheuristik algoritması  
(SIMO metaheuristic algorithm)

```

fonksiyon SIMO(ite, N, veri kümesi):
duyarlılar, enfekte, iyileşen, çözümler = [], [], [], []
X = başlat(N, duyarlılar)
duyarlılar = eğit(duyarlılar, X, N)
model = cnn_modeli
parametreler = {"π": π, "Γ": Γ, "γ": γ, "β": β}
π, Γ, γ, β = model(parametreler)
duyarlılar.uyum(asc)
global_en_ iyi, mevcut_en_ iyi, enfekte[0] = duyarlılar[0]
for dönem in range(ite):
yerel_pozisyon = pozisyon(enfekte)
for i in range(len(enfekte)):
yeni_enfekte = enfekte[i]
if yerel_pozisyon[i] < 0,5:
tmp = rastgele(0, 3 * enfekte[i] * s_oranı, y, β)
else:
tmp = rastgele(0, 3 * enfekte[i] * l_oranı, 7, β)
yeni_enfekte += tmp
enfekte[i] = yeni_enfekte
r = rastgele(0, 4 * enfekte[i], 7)
iyileşen.append(m)
duyarlılar[i] = enfekte[i] - r
mevcut_en_ iyi = uyumlu(duyarlılar)
if mevcut_en_ iyi > global_en_ iyi:
global_en_ iyi = mevcut_en_ iyi
çözümler.append(duyarlılar)
dönüş global_en_ iyi, çözümler

```

Önerilen SIMO algoritmasının karmaşıklığı, algoritmayı başlatma, optimize etme ve sonuçları döndürme aşamalarına bölerek hesaplanabilir. İlk aşama, yani başlatma aşaması, 5, 7 ve 8. Satırlarda,  $O(n)$  hesaplamalarına yol açacak ve diğer tüm satırlar  $O(1)$  olarak değerlendirilen temel işlemlerin kanıtlarını içerir [53]. Hesaplama açısından ifade edecek olursak,  $O(n) + O(n) + O(n)$ 'nin toplamı



Şekil 5. Önerilen SIMO algoritmasının akış şeması, popülasyon başlatma ve algoritma parametre seçimi (Flowchart of the proposed SIMO Algorithm, population initialisation and Algorithm parameter selection)

Tablo 4. İki Aşamalı DÖ Modelleri Kullanan SIMO Algoritması (SIMO Algorithm Using Two-Stage DM Models)

Değişken	İşlev	İşlev	İşlev	İşlev	İşlev	İşlev
Notasyon	$\pi$	-	$\beta_1$	$\beta_2$	-	-
Parametre Aralığı	[0,1, 0,9]	randf(0, 1)	[0,1, 0,9]	[0,1, 0,9]	randf(0, 1)	randf(0, 1)
Adım	0,1	N/A	0,1	0,1	N/A	N/A
Boyut	10	10	10	10	10	10

algoritması analizi kuralına göre  $O(n)$ 'ye eşit olduğu açıktır. Optimizasyon aşamasındaki 10-31. Satırlarda görülen iç içe geçmiş döngü,  $O(n^2)$ 'ye eşittir. Sonuçları döndürme işleminin karmaşıklığı  $O(1)$  ile eşitlenebileceğinden, tüm algoritmanın karmaşıklığı  $O(n) + O(n^2) + O(1)$ 'i toplamakla değerlendirilebilir ve bunun sonucunda  $O(n^2)$  elde edilir. Önceki paragrafta açıklanan sözdizim akış şeması, algoritmanın optimizasyon sürecini gösteren iterasyon aşamasını gösteren Şekil 6'da verilmiştir. Bu iterasyon aşamasında, SIMO algoritmasının keşif ve sömürü aşamasını vurguluyoruz, burada  $Ipos_i^{t+1} < 0,5$  değerlendirilmektedir. Algoritmanın başlatma aşamasını Tablo 3'te tartışıldığı gibi düşündüğümüzde, akış şeması çözüm uzayı ve parametre değerlerinin nasıl elde edildiğini gösterir.

Sonraki bölümde, önerilen SIMO algoritmasının detaylı deneysel sonuçları ile ilgili kapsamlı bir tartışma sunulmaktadır. SIMO'nun uygulanması için, Tablo3'te açıklandığı gibi SIM modelinin çözüm uzayı başlatma, parametre ayarlama ve optimizasyon süreçlerini birleştirdik.

#### 4. Deneysel Sonuçlar (Experimental Results)

Son bölümde açıklanan metodoloji, yaklaşımın uygulanabilirliğini göstermek için deneyimleniyor. Bu bölümde, deney sürecini

destekleyen hesaplama ortamını ve parametre ayarlarını özetliyoruz. Ayrıca, önerilen SIMO yönteminin optimizasyon sürecini incelemek için seçilen IEEE CEC fonksiyonları listesine dair detaylar sunuyoruz.

##### 4.1. Sistem ve Parametre Yapılandırması (System and Parameter Configuration)

Deneyimleme için hesaplama kurulumu, Google işbirliği (Google Colab) platformu kullanılarak gerçekleştirildi. Bu platform, 12GB belleğe ve 100GB disk boyutuna sahiptir ve her ikisi de Python 3 Google compute engine backend ve bir grafik işlem birimi (GPU) ile bağlıdır. Daha fazla deney, aşağıdaki yapılandırmalar kullanılarak oluşturulan bir örnek kullanılarak Google Cloud compute engine üzerinde gerçekleştirildi: Intel Broadwell merkezi işlem birimi (CPU) platformunda n1-highmem-8 makine türü, 200GB önyükleme disk, 8 vCPU ve 50GB bellek. Algoritmanın parametre yapılandırmaları Tablo 4'te detaylı olarak açıklanmıştır. Bu değer kümesi, önerilen SIMO algoritması için en iyi optimal parametre seçimini öğrenme tekniği kullanarak keşfetmek amacıyla uygulanan deneyime uygulanmıştır. SIMO optimizasyon yönteminin türetilmesi gereken belirli altı parametre değeri bulunmaktadır. Her parametrenin belirli bir değeri, o parametre için belirlenmiş olan bir değer aralığından elde edilebilir. Tablo, işe alım oranı ( $\pi$ ), bulaşıcı kişi ile temas oranı ( $\beta_1$ ) ve iyileşen kişi ile temas oranı ( $\beta_2$ ) parametreleri için değerlerin [0,1]

aralığından türetildiğini göstermektedir. Diğer yandan, hastalığa bağlı ölüm oranı ( $\Gamma$ ), iyileşme oranı ( $\gamma$ ) ve doğal ölüm oranı ( $\tau$ ) değerleri,  $[0,1]$  aralığında rastgele bir modelden oluşturulmuş olan ondalık değerleridir. Her parametre ile ilgili adımlar değişkeni, belirli değerleri kullanarak bir adım işlemi uygulamak için kullanılırken, değerlerin aralık içinde türetilmesini sağlar. Her ızgara değerinden, DÖ modelinin kullanılacak en uygun değer kombinasyonunu bulmak için denemelerde seçilecek olan potansiyel değerlerin bir listesini içermesi beklenir. Her durumda ızgara boyutu, her parametre için aday değerlerle ilgili olarak on değeri alır.

Parametreler ve bu parametrelere ait üretilen aday değerler, DÖ modelinin eğitimine uygulandı ve elde edilen en iyi çözüm, SIMO'nun eğitimi için uygundur. Bir sonraki alt bölümde, sonuçlanan SIMO konfigürasyonunu sıkı bir şekilde test etmek için uygulanan IEEE CEC fonksiyonlarını açıklıyoruz.

#### 4.2. IEEE CEC Fonksiyonları Karşılaştırma (Comparison of IEEE CEC Functions)

Karşılaştırmalı performans analizi için geleneksel olarak benchmarking fonksiyonları kullanılır. Standart benchmarking fonksiyonları ve IEEE CEC fonksiyonları, optimizasyon algoritmalarının performanslarını test etmek için geniş bir şekilde kabul edilmiş ve uygulanmıştır. Çünkü IEEE CEC fonksiyonları, gerçek hayattaki optimizasyon problemleri ile karşılaştırılabilir bir benzersizlik ve karmaşıklık gösterir, SIMO algoritmasının değerlendirilmesine bu fonksiyonlar üzerinde odaklandı. Tablo 5'te, CEC'nin kısıtlı fonksiyonları özetlenmiştir, bunlar temel fonksiyonlardan oluşur. CEC fonksiyonları, 14 temel CEC fonksiyonunun kaydırma ve döndürme işlemi uygulanarak daha karmaşık bir temsilden türetilmiştir. Bu türetime, 14 temel CEC fonksiyonunun yirmi sekiz (28) hibrit türetilmiştir. C1-C8 ve C10, karşılık gelen CEC fonksiyonlarının kaydırılmış versiyonlarıdır, C9 ve C11-C16 ise karşılık gelen CEC fonksiyonlarının kaydırma-döndürme versiyonlarıdır. C17-C22 fonksiyonları, belirli CEC fonksiyonlarının birleşimlerinin kaydırılmış versiyonlarını temsil ederken, C23-C28, önceden tanımlanmış CEC hibritleridir. Toplamda, deneylerde 42 CEC tabanlı fonksiyon kullanılmıştır.

Bu çalışmada gerçekleştirilen deney üç bölüme ayrılabilir: LSTM modelinin eğitilmesi, SIMO'nun çözüm alanına uygun potansiyel çözümler üretmek için; altı SIMO parametre şeması için tüm aday karşılık gelen değerlerle parametre tablosunun doldurulması, böylece bunlar permütasyon olarak birleştirilip SIMO eğitilmek ve optimize edilmiş çözüm alanının görüntü temsiline üretmek; ve ESA kullanarak optimize edilmiş çözüm alanının görüntü temsiline gömülü

özellikleri öğrenmek, amacıyla en uygun temsili öğrenmek ve ilgili parametrenin seçileceği temsilin seçilmesi. İlerleyen bölümde, sonuçları sunuyor ve tabloda belirtilen CEC fonksiyonlarına dayalı bir karşılaştırma sağlıyoruz.

### 5. Sonuçlar ve Tartışmalar (Results and Discussions)

Bu bölümde sunulan sonuçlar, SIMO'nun DÖ yöntemleri kullanılarak ön eğitimi ve CEC fonksiyonları kullanılarak SIMO'nun eğitimi temel alınarak elde edilmiştir. Bölüm, üç ana konuya odaklanarak üç alt bölüme ayrılmıştır: 1) ESA kullanılarak elde edilen SIMO parametre seçme yeteneğinin değerlendirilmesi; 2) SIMO'nun diğer benzer optimizasyon yöntemleri ile karşılaştırmalı analizi, DÖ destekli yöntemin tercih edilip edilemeyeceğini göstermek için ve 3) çalışmadan elde edilen bulguların bir tartışma olarak derlenmesi.

#### 5.1. SIMO Parametre Ayarlarının ESA ile Değerlendirilmesi (Evaluation of SIMO Parameter Settings with ESA)

SIMO Parametre Ayarlarının ESA ile Değerlendirilmesi SIMO'nun fayda sağlaması için alışlagelmiş meta-sezgisel algoritma tasarlama yönteminden farklı yeni bir yaklaşım izlenerek DÖ yöntemi iki aşamada uygulanmıştır. İlk olarak, DÖ, kaliteli bireylerin başlangıç arama alanına eklenmesini sağlamak için kullanılmıştır. Bu yaklaşım, başlangıç arama alanına stokastik bir teknik kullanma geleneksel yönteminden farklıdır. DÖ metodunun SIMO'yu geliştirmek için bir sonraki adımı, bir parametre-değer kombinasyonunun kullanılabilirliğini öngörmek için kullanılmaktadır. Bu parametre tablosu çözümü için altı parametre şemasının potansiyel değerleri birleştirilerek SIMO'yu eğitmek için 513 aday yapılandırma elde edildi. Bu 513 aday yapılandırma, daha sonra SIMO'nun performansını test etmek için kullanıldı ve çözüm alanını temsil eden dağılım grafiği görüntü formatında saklandı. Bu görüntü temsilleri daha sonra ESA 'i eğitmek için daha iyi bir kümeleşmiş noktalar gösteren en iyi görüntüyü öngörmek için kullanıldı. Her dağılım grafiği, SIMO'yu eğitmek için takılacak değerlere sahip parametre yapılandırmasına karşılık gelir. Ancak, tamamen SIMO'yu eğitmek için iyi bir yakınsama gösteren en iyi parametre yapılandırmasının seçimini rehberlik eden yakınsama temsili ve elde edilen çözümlerin uygunluğu bizi ilgilendirmektedir.

Tablo 6, 513 potansiyel yapılandırmadan yirmi parametre yapılandırmasını listeler. Sonuçlar, hastalığa neden olan ölüm oranı ( $\Gamma$ ), iyileşme oranı ( $\gamma$ ) ve doğal ölüm oranı ( $\tau$ ) için değerlerin tabloda listelenen tüm yapılandırmalarda benzer olduğunu göstermektedir. Bununla birlikte, işe alım oranı ( $\pi$ ), bulaşıcı temas oranı ( $\beta_1$ ) ve iyileşen temas oranı ( $\beta_2$ ) için elde edilen değerler çoğu durumda

**Tablo 5.** SIMO yöntemlerini değerlendirmek için uygulanan CEC fonksiyonları (CEC functions applied to evaluate SIMO methods)

İşlev	İşlev	Hibrit CEC Fonksiyon	CEC Fonksiyon	A2
C1	C15	S [CEC01]	SR CEC13	1,2,3,4
C2	C16	S [CEC02]	SR CEC14	2,3,4,5
C3	C17	S [CEC09, CEC08, CEC01]	S CEC03	3,4,5,6
C4	C18	S [CEC02, CEC12, CEC08]	S CEC04	4,5,6,7
C5	C19	S [CEC07, CEC06, CEC04, CEC14]	S CEC05	5,6,7,8
C6	C20	S [CEC12, CEC03, CEC13, CEC08]	S CEC06	6,7,8,9
C7	C21	S [CEC14, CEC12, CEC04, CEC09, CEC01]	S CEC07	7,8,9,10
C8	C22	S(1,2,3,4,5) [C4, C1, C2, C3, C1]	S CEC08	8,9,10,11,12
C9	C23	S(1,2,3) [C10, C9, C14]	SR CEC08	9,10,11
C10	C24	S(1,2,3) [C11, C9, C1]	S CEC09	10,11,12
C11	C25	S(1,2,3,4,5) [C11, C13, C1, C6, C7]	SR CEC09	11,12,13,14,15
C12	C26	S(1,2,3,4,5) [C14, C09, C11, C06, C1]	SR CEC10	12,13,14,15,16
C13	C27	S(1,2,3,4,5) [C15, C13, C13, C11, C16, C1]	SR CEC11	13,14,15,16,17,18
C14	C28	S(4,5,6) [C17, C18, C19]	SR CEC12	14,15,16

**Tablo 6.** SIMO'nun performansını test etmek için kullanılan parametre-değer kombinasyonu (Parameter-value combination used to test SIMO's performance)

Etiket	$\Pi$	$\Gamma$	$\beta_1$	$\gamma$	$\beta_2$	$\tau$
A1	0,184	0,2723	0,185	0,5034	0,185	0,4392
A2	0,386	0,2723	0,185	0,5034	0,385	0,4392
A3	0,888	0,2723	0,285	0,5034	0,285	0,4392
A4	0,485	0,2723	0,285	0,5034	0,785	0,4392
A5	0,885	0,2723	0,485	0,5034	0,186	0,4392
A6	0,687	0,2723	0,485	0,5034	0,686	0,4392
A7	0,185	0,2723	0,585	0,5034	0,287	0,4392
A8	0,488	0,2723	0,585	0,5034	0,785	0,4392
A9	0,887	0,2723	0,685	0,5034	0,587	0,4392
A10	0,387	0,2723	0,685	0,5034	0,788	0,4392
A11	0,885	0,2723	0,786	0,5034	0,186	0,4392
A12	0,484	0,2723	0,785	0,5034	0,887	0,4392
A13	0,284	0,2723	0,884	0,5034	0,285	0,4392
A14	0,182	0,2723	0,885	0,5034	0,385	0,4392
A15	0,181	0,2723	0,887	0,5034	0,485	0,4392
A16	0,885	0,2723	0,885	0,5034	0,585	0,4392
A17	0,187	0,2723	0,886	0,5034	0,785	0,4392
A18	0,584	0,2723	0,884	0,5034	0,786	0,4392
A19	0,386	0,2723	0,386	0,5034	0,385	0,4392

farklıdır. Bu sonuç, parametre değeri yapılandırmasının SIMO'nun performansını iyileştirmek için hangi değer kombinasyonunun kullanılacağını anlamamın gerekli olduğunu göstermektedir. İşe alım oranı ( $\pi$ ) için seçilen değerler arasında 0,1-0,8 bulunmakta ve 0,9-1,0 kullanılmamakta, sık sık 0,1 tekrarlanmaktadır.  $\beta_1$  ve  $\beta_2$  için elde edilen değerler de 0,1-0,8 aralığındadır, burada  $\beta_1$  için sık sık 0,8 ve  $\beta_2$  için 0,7 sık sık tekrarlanmaktadır. Bu sonuçların anlamı, belirli bir değerin SIMO'nun performansını artırmak için daha uygun olduğunu doğrular. Bu, en iyi değeri belirlemek için kontrol parametrelerine atanacak değerlerin bir aralığından doğru bir şekilde belirlenmesi için DÖ yöntemini kullanmanın önemini bir kez daha gösterir. Bu, optimizasyon algoritmalarının kontrol parametrelerine rastgele değer atamak için popüler stokastik yöntemlere karşı çıkar. Yani, rastgele seçilen değerlerin, SIMO'nun daha iyi performans göstermesine yardımcı olması için belirli bir şekilde seçilmesi gerektiği savunuluyor. Bu nedenle, DÖ yöntemi kullanılarak bu değerlerin en iyi kombinasyonlarının belirlenmesi ve bu değerlerin performansını artırmak için en uygun olanlarının seçilmesi önerilmektedir. Bu, kontrol parametrelerinin belirli bir düzen içinde seçilmesinin, rastgele seçimden daha etkili olabileceğini göstermektedir.

Her bir parametre-değer konfigürasyonu için SIMO algoritmasının eğitimi, 50 iterasyon kullanılarak gerçekleştirildi. Bu eğitimin amacı, çözüm uzayındaki çözümlerin konverjans eğrilerini oluşturmak için kısmi bir eğitim yapmaktır. Bu nedenle, çözümlerin konverjansını temsil eden dağılım grafikleri (scatter plot), ESA modelini eğitmek için görsel girdi olarak elde edildi. Şekil 7, Tablo 6'da listelenen parametre-değer konfigürasyonları için bu dağılım grafiği sunmaktadır. Her bir grafik, çözümlerin belirli bir eğitim süresi sonrasında nasıl işlediğini gösteren dağılım grafiği içindeki noktaların benzersiz bir dağılımını gösterir.

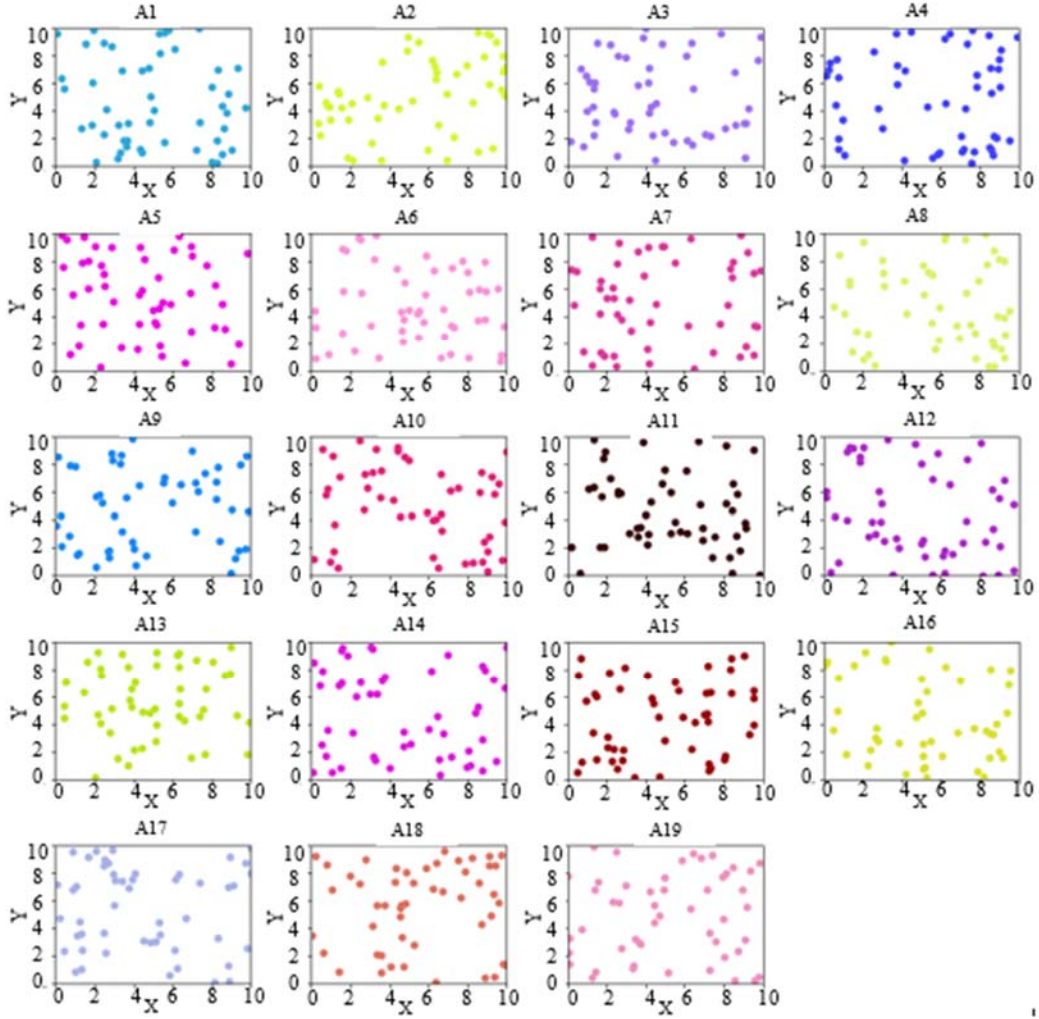
ESA mimarisinin tam eğitimi sonrasında, daha iyi bir yakınsama gösteren doğru tahmin edilmiş görüntülerden biri seçildi. Seçilen görüntünün karşılık gelen parametre değeri daha sonra Tablo 7'de gösterildiği gibi elde edilmiştir. Şekil olarak oluşturulan görüntü örnekleri, ESA mimarisini eğitmek için kullanıldı ve görüntülerde temsil edilen desenleri öğrenmesi amaçlandı. Hedef, sinir ağının girişlerdeki farklı yakınsamaları öğrenmesini sağlamak ve sonuç olarak SIMO yöntemini tam olarak eğitmek için en uygun görüntü temsili öğrenmesini sağlamaktır.

**Tablo 7.** SIMO parametreleri ve bunlara ait notasyonlar ile SIMO'nun eğitimi için uygulanan değerler (SIMO parameters and their notations and the values applied for SIMO training)

Değişken	Notasyon	Değer
İşe Alım Oranı	$\pi$	0,3
Hastalığa Neden Olan Ölüm Oranı	$\Gamma$	0,0783
Bulaşıcı İletişim Oranı	$\beta_1$	0,3
İyileşenlerle İletişim Oranı	$\beta_2$	0,3
İyileşme Oranı	$\gamma$	0,1914
Doğal Ölüm Oranı	$\tau$	0,1272

Sonraki alt bölümde, DÖ ile elde edilen parametre-değer konfigürasyonu kullanılarak eğitilen SIMO'nun performansı sunulmuş ve SIMO'nun diğer benzer yöntemlerle karşılaştırılması analizi yapılmıştır.

Tablo 8-Tablo 11'de belirtildiği gibi, C22-C28 aralığındaki fonksiyonlar, C1-C19 aralığındaki fonksiyonlara kaydırma işlemi uygulanarak elde edilmektedir. Zaten karmaşık olan C1-C19 aralığındaki fonksiyonlar birleştirilip ardından kaydırma işlemine tabi tutularak daha yüksek bir karmaşıklık düzeyine ulaşılmaktadır. SIMO'nun bu kategori için değerlendirilen kıyaslama fonksiyonu, en iyi uygunluk değerlerini C22 ve C28 için sunduğunu göstermektedir. Bu durum, SIMO'nun C22-C28 fonksiyonlarıyla benzer zor optimizasyon problemleriyle başa çıkabilme yeteneğini ortaya koymaktadır. Ayrıca, her algoritmanın çözüm uzayındaki çözümler için en iyi uygunluk değerini belirleme sayısını hesaplamak için tüm algoritmaların ortalama performansı dikkate alınmıştır. Bu sonuçlar, ilgili algoritmalar için her sütun altında özetlenmiştir. Örneğin, ABC, AO, AOA, ArchOA, EOSA, FFA, IWO, MA, SFO, SIMO, SSA ve TLO'nun 42 CEC fonksiyonu için diğer yöntemlere göre üstünlük gösterdiği durumların sayısı sırasıyla 15, 25, 21, 27, 28, 13, 17, 11, 11, 27, 27, 34 ve 27 olarak belirlenmiştir. Aynı şekilde, her algoritmanın döndürdüğü en kötü çözümün performansı gözlemlenmiş ve ABC, AO, AOA, ArchOA, EOSA, FFA, IWO, MA, SFO, SIMO, SSA ve TLO için en kötü durumun diğerlerinden daha iyi performans gösterdiği durumların sayısının sırasıyla 8, 8, 1, 11, 8, 8, 7, 8, 11, 16, 8, 32 ve 9 olduğu test edilmiştir. Şekil 7'de, ABC, AO, AOA, ArchOA, EOSA, FFA, IWO, MA, SFO, SIMO, SSA ve TLO için keşif ve kullanım grafikleri gösterilmektedir.



Şekil 6. Seçilmiş bazı parametre-değer konfigürasyonları gösteren dağılım grafiği  
(Scatter plot showing some selected parameter-value configurations)

**Tablo 8.** CEC fonksiyonları kullanılarak, insan, evrim, sürü zekası, biyoloji, fizik ve matematik temelli yöntemler ile SIMO'nun diğer optimizasyon algoritmalar ile karşılaştırması  
(Comparison of SIMO with other optimisation algorithms using CEC functions, human, evolution, swarm intelligence, biology, physics and mathematics based methods)

Fonk.	Ölçü	ABC	AO	AOA	ArchOA	EOSA	FFA	IWO	MA	SFO	SIMO	SSA	TLO
CEC01	En iyi	3,6E+3	2,0E-2	0,0E+0	1,2E-36	0,0E+0	7,6E+3	2,6E+5	8,8E+5	1,8E-7	0,0E+0	6,7E-52	3,2E-117
	En kötü	8,3E+5	3,2E+5	0,0E+0	1,2E+6	2,2E+6	2,1E+6	5,3E+5	8,8E+5	3,1E+2	1,7E+5	5,5E+0	2,1E+6
CEC02	En iyi	1,3E+7	2,7E+2	0,0E+0	2,5E-101	0,0E+0	5,1E+5	2,5E+8	1,8E+8	2,3E-5	0,0E+0	3,1E-52	3,3E-116
	Std	1,2E+8	1,2E+8	0,0E+0	1,2E+8	2,3E+6	6,9E+7	2,0E+7	6,7E-8	3,2E+1	2,2E+6	5,0E+1	3,0E+7
CEC03	En kötü	1,3E+5	1,2E+5	0,0E+0	1,5E+5	2,8E+3	1,1E+5	5,8E+3	1,8E+3	3,9E+2	3,8E+3	1,1E+3	5,7E+3
	En iyi	6,9E+0	3,5E-1	0,0E+0	5,1E-30	0,0E+0	1,6E+3	2,6E+3	1,3E+5	1,5E-5	0,0E+0	3,0E-32	2,8E-119
CEC04	Std	3,1E+3	3,8E+3	0,0E+0	3,2E+3	2,6E+3	1,3E+3	2,2E+3	2,9E-11	3,3E+1	2,3E+3	1,1E+0	1,6E+3
	En kötü	2,6E+3	1,1E+6	0,0E+0	3,6E+3	1,5E+5	1,7E+3	3,2E+3	1,3E+5	7,6E+2	3,6E+3	7,1E+2	3,2E+3
CEC05	En iyi	1,5E+1	1,8E+0	0,0E+0	8,1E-20	0,0E+0	1,4E+3	3,8E+4	1,9E+4	3,5E-4	0,0E+0	1,1E-23	1,2E-119
	Std	4,4E+3	1,8E+3	0,0E+0	7,7E+3	2,0E+3	1,2E+3	2,1E+3	3,1E-9	2,1E+0	4,6E+3	3,9E-1	2,4E+3
CEC06	En kötü	5,5E+3	3,5E+6	0,0E+0	1,2E+4	8,8E+5	2,3E+3	4,3E+3	1,9E+4	1,8E+3	2,2E+4	1,8E+2	3,7E+3
	En iyi	3,8E+3	3,1E-2	0,0E+0	2,6E-35	0,0E+0	4,1E+3	5,7E+4	1,6E+5	1,9E-6	0,0E+0	1,1E-45	2,1E-120
CEC07	Std	5,1E+3	3,8E+3	0,0E+0	5,2E+3	3,9E+3	6,7E+4	2,7E+4	4,1E-10	2,1E+1	6,8E+3	3,2E-1	1,9E+4
	En kötü	5,2E+5	3,8E+5	0,0E+0	8,8E+5	2,1E+6	1,7E+5	4,3E+4	1,6E+5	7,5E+2	4,2E+5	1,3E+2	6,1E+4
CEC08	En iyi	2,5E+4	1,7E-1	0,0E+0	2,4E-29	0,0E+0	5,6E+2	7,3E+3	2,5E+4	1,2E-5	0,0E+0	1,1E-36	1,4E-120
	Std	5,7E+4	3,6E+3	0,0E+0	2,9E+4	3,5E+3	4,7E+2	8,6E+2	4,0E-10	1,2E+1	2,1E+3	1,2E-1	1,8E+3
CEC09	En kötü	1,5E+6	1,1E+6	0,0E+0	2,6E+6	1,7E+6	1,3E+3	1,2E+3	2,5E+4	3,4E+2	1,7E+6	1,1E+2	3,6E+3
	En iyi	2,8E+1	2,4E+0	0,0E+0	2,7E-20	0,0E+0	5,4E+3	3,1E+4	5,5E+4	3,6E-4	0,0E+0	5,6E-40	3,2E-121
CEC10	Std	6,2E+3	1,9E+3	0,0E+0	5,4E+3	1,8E+3	1,8E+3	2,2E+3	1,1E-8	1,8E+0	6,4E+3	1,2E-1	2,3E+3
	En kötü	3,5E+3	3,6E+6	0,0E+0	7,8E+3	6,3E+5	1,5E+4	2,2E+4	5,5E+4	1,4E+3	3,4E+4	1,8E+2	1,5E+4

**Tablo 9.** CEC fonksiyonları kullanılarak, insan, evrim , sürü zekâsı, biyoloji, fizik ve matematik temelli yöntemler ile SIMO'nun diğer optimizasyon algoritmalar ile karşılaştırması devamı (Comparison of SIMO with other optimization algorithms using CEC functions, human, evolution, swarm intelligence, biology, physics and mathematics based methods continue)

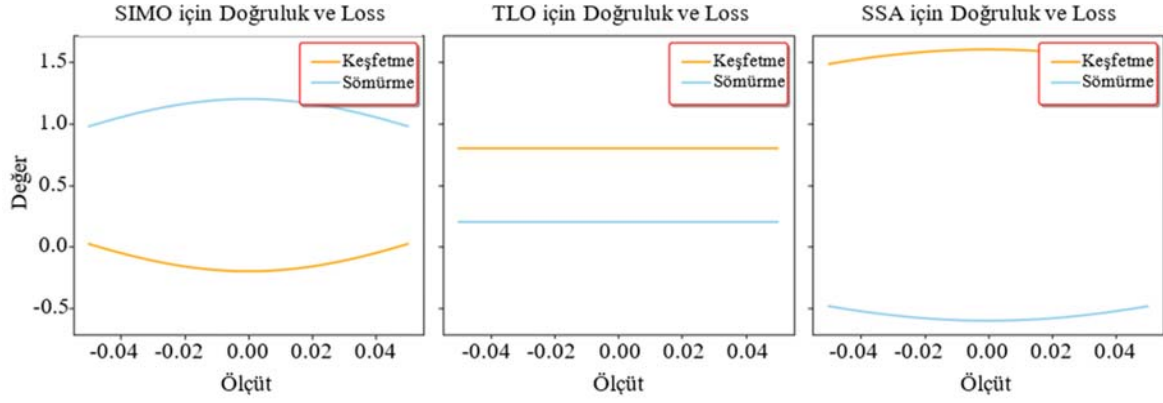
Fonk.	Ölçü	ABC	AO	AOA	ArchOA	EOSA	FFA	IWO	MA	SFO	SIMO	SSA	TLO
CEC08	En iyi	1,5E+2	3,0E+0	0,0E+0	5,5E-19	0,0E+0	1,4E+4	6,2E+4	1,9E+5	8,3E-4	0,0E+0	2,3E-53	1,2E-122
	Std	2,2E+4	2,1E+3	0,0E+0	6,7E+3	1,3E+4	6,3E+3	2,6E+3	6,1E-9	2,1E+0	1,1E+4	7,5E-2	3,2E+3
	En kötü	5,5E+3	3,1E+6	0,0E+0	2,1E+4	3,8E+5	1,6E+5	2,8E+4	1,9E+5	1,2E+3	7,7E+4	1,8E+2	1,9E+4
CEC09	En iyi	3,5E+1	2,1E+0	0,0E+0	2,1E-25	0,0E+0	3,2E+3	3,3E+4	1,4E+4	7,6E-4	0,0E+0	1,3E-49	1,3E-123
	Std	6,1E+3	2,3E+3	0,0E+0	1,1E+4	2,0E+3	1,1E+3	1,8E+3	6,6E-9	1,2E+0	1,3E+4	2,6E-1	1,4E+3
	En kötü	4,3E+3	3,3E+6	0,0E+0	7,7E+3	8,1E+5	2,3E+3	2,5E+3	1,4E+4	1,7E+3	3,6E+4	2,1E+2	4,7E+3
CEC10	En iyi	1,3E+1	2,6E+0	0,0E+0	4,1E-25	0,0E+0	5,3E+3	2,7E+4	7,7E+4	8,6E-4	0,0E+0	1,5E-37	3,1E-124
	Std	3,2E+3	1,3E+3	0,0E+0	6,1E+3	8,8E+3	4,5E+3	1,9E+3	7,5E-9	1,2E+0	4,5E+3	9,5E-2	1,6E+3
	En kötü	5,3E+3	3,7E+6	0,0E+0	1,7E+4	3,6E+5	3,3E+4	2,8E+4	7,7E+4	1,5E+3	1,6E+5	3,6E+2	7,2E+3
CEC11	En iyi	1,5E+1	3,4E+0	0,0E+0	4,4E-20	0,0E+0	3,6E+3	6,1E+4	1,5E+5	8,5E-4	0,0E+0	1,5E-58	4,3E-125
	Std	2,3E+4	2,2E+3	0,0E+0	6,7E+3	1,3E+4	6,3E+3	2,5E+3	7,5E-9	2,1E+0	1,1E+4	7,5E-2	3,2E+3
	En kötü	5,6E+3	3,4E+6	0,0E+0	1,8E+4	3,4E+5	1,9E+5	2,7E+4	1,5E+5	1,2E+3	7,9E+4	1,5E+2	1,9E+4
CEC12	En iyi	3,7E+2	3,7E+0	0,0E+0	3,3E-21	0,0E+0	4,4E+3	6,5E+4	1,8E+5	4,4E-3	0,0E+0	2,5E-58	5,4E-126
	Std	4,4E+4	1,7E+3	0,0E+0	5,9E+3	1,2E+4	5,4E+3	1,4E+3	4,6E-8	1,2E+0	9,1E+3	1,9E-1	2,8E+3
	En kötü	3,3E+3	3,4E+6	0,0E+0	1,7E+4	3,1E+5	1,8E+5	3,5E+4	1,8E+5	8,2E+2	1,7E+5	1,2E+2	1,6E+4
CEC13	En iyi	2,8E+2	3,2E+0	0,0E+0	5,1E-21	0,0E+0	2,4E+4	1,1E+5	2,5E+5	1,0E-2	0,0E+0	2,8E-70	6,1E-127
	Std	3,5E+4	1,7E+3	0,0E+0	4,5E+3	7,9E+3	4,2E+3	1,6E+3	1,3E-7	2,1E+0	8,1E+3	2,4E-1	2,5E+3
	En kötü	2,1E+3	3,3E+6	0,0E+0	2,5E+4	2,8E+5	1,7E+5	3,3E+4	2,5E+5	7,6E+2	1,2E+5	1,1E+2	1,3E+4
CEC14	En iyi	3,8E+2	3,8E+0	0,0E+0	1,2E-21	0,0E+0	4,3E+4	1,3E+5	2,9E+5	1,4E-2	0,0E+0	1,8E-71	7,1E-128
	Std	4,6E+4	1,6E+3	0,0E+0	4,7E+3	8,7E+3	4,7E+3	1,8E+3	1,9E-7	1,8E+0	7,8E+3	3,4E-1	2,6E+3
	En kötü	3,2E+3	3,2E+6	0,0E+0	1,7E+4	3,4E+5	1,7E+5	3,5E+4	2,9E+5	7,2E+2	1,3E+5	1,1E+2	1,5E+4

**Tablo 10.** CEC fonksiyonları kullanılarak, insan, evrim , sürü zekâsı, biyoloji, fizik ve matematik temelli yöntemler ile SIMO'nun diğer optimizasyon algoritmalar ile karşılaştırması devamı (Comparison of SIMO with other optimization algorithms using CEC functions, human, evolution, swarm intelligence, biology, physics and mathematics based methods continue)

Fonk.	Ölçü	ABC	AO	AOA	ArchOA	EOSA	FFA	IWO	MA	SFO	SIMO	SSA	TLO
CEC15	En iyi	3,2E+2	2,5E+0	0,0E+0	1,6E-19	0,0E+0	3,3E+3	1,8E+5	1,4E+5	5,9E-4	0,0E+0	1,2E-69	8,0E-129
	Std	5,6E+4	2,3E+3	0,0E+0	8,2E+3	1,1E+4	7,1E+3	3,6E+3	3,3E-8	1,2E+0	1,4E+4	6,2E-1	4,3E+3
	En kötü	3,2E+3	2,9E+6	0,0E+0	1,4E+4	2,5E+5	1,6E+5	2,1E+4	1,4E+5	6,3E+2	5,2E+4	7,7E+1	1,4E+4
CEC16	En iyi	4,6E+2	3,6E+0	0,0E+0	1,4E-21	0,0E+0	3,7E+4	1,2E+5	2,5E+5	1,4E-2	0,0E+0	9,7E-73	9,4E-130
	Std	5,6E+4	2,0E+3	0,0E+0	5,9E+3	1,1E+4	6,6E+3	2,7E+3	1,9E-7	2,1E+0	7,8E+3	2,9E-1	3,3E+3
	En kötü	3,2E+3	3,2E+6	0,0E+0	1,6E+4	3,2E+5	1,6E+5	3,2E+4	2,5E+5	5,5E+2	9,8E+4	9,6E+1	1,3E+4
CEC17	En iyi	3,2E+2	3,0E+0	0,0E+0	1,4E-19	0,0E+0	2,9E+4	1,6E+5	1,1E+5	5,3E-4	0,0E+0	3,2E-70	1,2E-130
	Std	5,6E+4	1,8E+3	0,0E+0	8,2E+3	1,1E+4	5,4E+3	2,4E+3	2,1E-7	2,1E+0	1,3E+4	5,3E-1	2,9E+3
	En kötü	3,3E+3	3,2E+6	0,0E+0	1,4E+4	2,6E+5	1,6E+5	2,1E+4	1,1E+5	6,4E+2	5,0E+4	7,9E+1	1,3E+4
CEC18	En iyi	5,2E+2	3,6E+0	0,0E+0	8,3E-22	0,0E+0	4,2E+4	1,4E+5	2,8E+5	1,5E-2	0,0E+0	3,5E-73	1,5E-131
	Std	7,4E+4	1,9E+3	0,0E+0	7,2E+3	1,5E+4	7,1E+3	2,7E+3	2,4E-7	2,1E+0	9,4E+3	4,3E-1	3,3E+3
	En kötü	3,6E+3	3,2E+6	0,0E+0	1,9E+4	3,7E+5	2,1E+5	4,3E+4	2,8E+5	6,6E+2	1,1E+5	1,1E+2	1,6E+4
CEC19	En iyi	3,8E+2	3,6E+0	0,0E+0	1,7E-22	0,0E+0	4,2E+4	1,5E+5	2,7E+5	1,3E-2	0,0E+0	7,9E-73	1,9E-132
	Std	7,4E+4	2,0E+3	0,0E+0	6,7E+3	1,5E+4	6,5E+3	2,3E+3	1,8E-7	2,1E+0	8,9E+3	3,6E-1	3,2E+3
	En kötü	3,7E+3	3,2E+6	0,0E+0	1,8E+4	3,6E+5	2,1E+5	4,2E+4	2,7E+5	5,9E+2	1,1E+5	1,1E+2	1,5E+4
CEC20	En iyi	4,2E+2	3,6E+0	0,0E+0	1,3E-22	0,0E+0	4,1E+4	1,4E+5	2,7E+5	1,5E-2	0,0E+0	3,5E-73	1,7E-133
	Std	6,2E+4	1,9E+3	0,0E+0	6,9E+3	1,4E+4	6,7E+3	2,4E+3	2,4E-7	2,1E+0	8,8E+3	3,2E-1	3,3E+3
	En kötü	3,2E+3	3,2E+6	0,0E+0	1,7E+4	3,4E+5	1,9E+5	4,2E+4	2,7E+5	5,8E+2	1,0E+5	1,2E+2	1,5E+4
CEC21	En iyi	3,8E+2	3,6E+0	0,0E+0	1,3E-22	0,0E+0	4,2E+4	1,5E+5	2,7E+5	1,5E-2	0,0E+0	3,5E-73	2,0E-134
	Std	7,4E+4	2,0E+3	0,0E+0	6,7E+3	1,5E+4	6,5E+3	2,3E+3	1,8E-7	2,1E+0	8,9E+3	3,6E-1	3,2E+3
	En kötü	3,7E+3	3,2E+6	0,0E+0	1,8E+4	3,6E+5	2,1E+5	4,2E+4	2,7E+5	5,8E+2	1,1E+5	1,1E+2	1,5E+4

**Tablo 11.** CEC fonksiyonları kullanılarak, insan, evrim , sürü zekâsı, biyoloji, fizik ve matematik temelli yöntemler ile SIMO'nun diğer optimizasyon algoritmalar ile karşılaştırması devamı (Comparison of SIMO with other optimization algorithms using CEC functions, human, evolution, swarm intelligence, biology, physics and mathematics based methods continue)

Fonk.	Ölçü	ABC	AO	AOA	ArchOA	EOSA	FFA	IWO	MA	SFO	SIMO	SSA	TLO
CEC22	En iyi	3,7E+2	3,7E+0	0,0E+0	2,0E-22	0,0E+0	4,4E+4	1,5E+5	2,7E+5	1,3E-2	0,0E+0	3,7E-73	1,9E-135
	Std	7,1E+4	2,0E+3	0,0E+0	6,7E+3	1,4E+4	6,6E+3	2,4E+3	1,9E-7	2,1E+0	8,7E+3	3,2E-1	3,3E+3
	En kötü	3,7E+3	3,2E+6	0,0E+0	1,8E+4	3,5E+5	2,1E+5	4,2E+4	2,7E+5	5,8E+2	1,1E+5	1,1E+2	1,5E+4
CEC23	En iyi	4,1E+2	3,5E+0	0,0E+0	1,4E-21	0,0E+0	4,0E+4	1,4E+5	2,7E+5	1,4E-2	0,0E+0	3,9E-74	1,6E-136
	Std	5,6E+4	1,9E+3	0,0E+0	6,9E+3	1,4E+4	6,7E+3	2,3E+3	2,4E-7	2,1E+0	8,8E+3	3,2E-1	3,2E+3
	En kötü	3,3E+3	3,2E+6	0,0E+0	1,7E+4	3,3E+5	1,9E+5	4,2E+4	2,7E+5	5,9E+2	1,0E+5	1,2E+2	1,5E+4
CEC24	En iyi	3,7E+2	3,7E+0	0,0E+0	1,7E-22	0,0E+0	4,4E+4	1,5E+5	2,7E+5	1,3E-2	0,0E+0	3,7E-73	1,9E-137
	Std	7,2E+4	2,0E+3	0,0E+0	6,6E+3	1,4E+4	6,5E+3	2,3E+3	1,9E-7	2,1E+0	8,8E+3	3,2E-1	3,3E+3
	En kötü	3,7E+3	3,2E+6	0,0E+0	1,8E+4	3,5E+5	2,0E+5	4,2E+4	2,7E+5	5,9E+2	1,1E+5	1,1E+2	1,5E+4
CEC25	En iyi	3,9E+2	3,5E+0	0,0E+0	1,5E-21	0,0E+0	3,9E+4	1,4E+5	2,7E+5	1,4E-2	0,0E+0	4,3E-74	1,7E-138
	Std	5,6E+4	2,0E+3	0,0E+0	7,0E+3	1,4E+4	6,6E+3	2,3E+3	2,4E-7	2,1E+0	8,8E+3	3,2E-1	3,3E+3
	En kötü	3,3E+3	3,2E+6	0,0E+0	1,7E+4	3,3E+5	1,9E+5	4,1E+4	2,7E+5	5,9E+2	1,0E+5	1,2E+2	1,5E+4



**Şekil 7.** C14 fonksiyonunu kullanarak ABC, AO, AOA, ArchOA, EOSA, FFA, IWO, MA, SFO, SIMO, SSA ve TLO için keşif ve sömürü grafikleri

(Exploration and exploitation plots for ABC, AO, AOA, ArchOA, EOSA, FFA, IWO, MA, SFO, SIMO, SSA and TLO using the C14 function)

Şekil 7'de görüldüğü üzere, iyi bir optimizasyon algoritmasının temel görevi, çözüm uzayındaki tüm bireyleri, problemi çözmek için uygun bir çözüme dönüştürmek üzere değiştirmektir. Bu sebeple, çözüm uzayındaki her birey, bir tür dönüşümü takip ederek çözüm uzayında yakınsayan bir mutasyon yörüngesine tabi tutulur. Bu tür bireylerin çözüm uzayındaki grafiği, bir metasezgisel algoritmanın en iyi çözümü seçebileceği aday bir çözüme nasıl yaklaştığını açıkça göstermektedir. Bu çalışmada, bireylerin çözüm uzayındaki yakınsamalarını göstermek ve karşılaştırmak amacıyla deneyler yapılmış sonuçlar ise Şekil 7'de gösterilmiştir.

## 6. Sonuçlar (Conclusions)

Bu araştırma, metaheuristik optimizasyon algoritmaları için optimal parametre-değer konfigürasyonlarını belirlemek amacıyla MÖ tekniklerinin kullanımını tanıtmaktadır. Bu yaklaşımı değerlendirmek için, SIMO kısaltmasıyla yeni bir metaheuristik algoritma önerilmiştir. SIMO, metaheuristik algoritmalar MÖ birleşimini kullanarak optimizasyon sürecinde parametre seçimini iyileştirmeyi amaçlamaktadır. İlk olarak, LSTM modeli, arama uzayının başlangıç popülasyonunu üretmek ve zenginleştirmek için kullanılmaktadır. Ayrıca, bir parametre ızgarası, altı SIMO-parametre şemasına karşılık gelen aday değerlerle doldurulmaktadır. Bu değerler daha sonra SIMO algoritmasını eğitmek ve optimize edilmiş çözüm uzayının görüntü temsillerini oluşturmak üzere permütasyon şeklinde birleştirilmektedir. Bu deney aşaması, ESA kullanarak görüntü temsillerindeki benzersiz özellikleri tanımlamak ve SIMO algoritmasını tam eğitmek için en uygun parametre-değer temsillerini öğrenmek açısından kritiktir.

Ardından, en iyi parametre-değer konfigürasyonunu seçtikten sonra SIMO, CEC fonksiyonları kullanılarak tamamen eğitilmiş ve test edilmiştir. Karşılaştırmalı tartışmalar ve analizler yapılarak SIMO'nun benzer yöntemlere karşı performansı değerlendirilmiştir. Elde edilen sonuçlar, önerilen yaklaşımın, tüm benchmarking fonksiyonlarında en iyi çözümleri elde ettiğini göstermektedir. Ayrıca, SIMO'nun optimizasyon aşamasındaki keşif arama süreci grafiği ve konverjansını analiz ederek, metodun günümüz optimizasyon yöntemlerine karşı rekabetçi performansını ortaya koymuştur. Çalışmanın bulguları, önerilen yaklaşımın, benzer algoritmaların yaygın olarak kullanılan stokastik yöntemlere olan bağımlılığını ortadan kaldırdığını ve kontrol parametrelerine değer atamak için rastgele bir yöntemi sorguladığını vurgulamaktadır. Bu, DÖ tekniklerinin kullanılmasının, bir veri kümesi içindeki en iyi parametre değerlerini doğru bir şekilde belirleme potansiyeli ile

dikkat çekmektedir. Gelecekteki çalışmalar için öneri olarak, Ackley, Griewank, Bent Cigar, Brown ve Alpine gibi ek standart benchmarking fonksiyonlarının kullanılması, benzer optimizasyon yöntemleri için parametre-değer konfigürasyonlarını keşfetmeyi kolaylaştırabilir. Ayrıca, SIMO algoritmasının dayanıklılığı, CEC 2011 ve CEC 2014 gibi daha farklı bilgilendirme fonksiyonlarıyla çeşitli hesaplama alanlarında karmaşık gerçek dünya optimizasyon problemleri ile uygulanabilir.

## Kaynaklar (References)

1. Wang L., Jiang Z., Liang Z., Liu J., Prediction of socketed pile settlement based on a hybrid form of multilayer perceptron via meta-heuristic algorithms, *Multiscale and Multidisciplinary Modelling, Experiments and Design*, 1-16, 2023.
2. Osman I. H., Kelly J. P., Meta-heuristics theory and applications, *Journal of the Operational Research Society*, 48 (6), 657-657, 1997.
3. Glover F. W., Kochenberger G. A. (Eds.), *Handbook of Metaheuristics*, Springer Science & Business Media, 57, 2006.
4. Süsler B., *Finansal Başarısızlığın Yapay Sinir Ağları ve Çok Değişkenli İstatistiksel Analiz Teknikleri ile Tahmin Edilmesi: Borsa İstanbul'da Bir Uygulama*, (Doktora Tezi, Bursa Uludağ Üniversitesi, Türkiye), 2022.
5. Pálfi B., Arora K., Kostopoulou O., Algorithm-based advice taking and clinical judgement: impact of advice distance and algorithm information, *Cognitive Research: Principles and Implications*, 7 (1), 70, 2022.
6. Mansori A. R., Nguyen S. K., Quantum-Inspired Genetic Algorithms for Combinatorial Optimization Problems, *Algorithm Asynchronous*, 1 (1), 16-23, 2023.
7. Yang J. M., Hwang C. N., Optimization of corrugated bulkhead forms by genetic algorithm, *Journal of Marine Science and Technology*, 10 (2), 10, 2021.
8. El Mouden S., Ouhimmou S., New multiobjective optimization algorithm using NBI-SASP approaches for mechanical structural problems, *International Journal for Simulation and Multidisciplinary Design Optimization*, 13, 4, 2022.
9. Rashidi H., Tsang E. P., *Port Automation and Vehicle Scheduling: Advanced Algorithms for Scheduling Problems of AGVs*, CRC Press, 2022.
10. You G., Sustainable vehicle routing problem on real-time roads: The restrictive inheritance-based heuristic algorithm, *Sustainable Cities and Society*, 79, 103682, 2022.
11. Hoseini Z., Varace H., Rafeizonooz M., Kim J. H. J., A New Enhanced Hybrid Grey Wolf Optimizer (GWO) Combined with Elephant Herding Optimization (EHO) Algorithm for Engineering Optimization, *Journal of Soft Computing in Civil Engineering*, 6 (4), 2022.
12. Forestiero A., Heuristic recommendation technique in Internet of Things featuring swarm intelligence approach, *Expert Systems with Applications*, 187, 115904, 2022.

13. Mirjalili S., Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm, *Knowledge-based Systems*, 89, 228-249, 2015.
14. Gupta K. D., Dwivedi R., Sharma D. K., Prediction of Covid-19 trends in Europe using generalized regression neural network optimized by flower pollination algorithm, *Journal of Interdisciplinary Mathematics*, 24 (1), 33-51, 2021.
15. Tian Y., Zhu W., Zhang X., Jin Y., A practical tutorial on solving optimization problems via PlatEMO, *Neurocomputing*, 518, 190-205, 2023.
16. Halim A. H., Ismail I., Das S., Performance assessment of the metaheuristic optimization algorithms: An exhaustive review, *Artificial Intelligence Review*, 54, 2323-2409, 2021.
17. Coello C. A. C., Brambila S. G., Gamboa J. F., Tapia M. G. C., Multi-objective evolutionary algorithms: past, present, and future, *Black Box Optimization, Machine Learning, and No-Free Lunch Theorems*, 137-162, Springer, 2021.
18. Parend P., Collet P., A review on complex system engineering, *Journal of Systems Science and Complexity*, 33, 1755-1784, 2020.
19. Asghari A., Sohrabi M. K., Yaghmaee F., Task scheduling, resource provisioning, and load balancing on scientific workflows using parallel SARSA reinforcement learning agents and genetic algorithm, *The Journal of Supercomputing*, 77, 2800-2828, 2021.
20. Ghosh A., Das S., Das A. K., A simple two-phase differential evolution for improved global numerical optimization, *Soft Computing*, 24, 6151-6167, 2020.
21. Cantún-Avila K. B., González-Sánchez D., Díaz-Infante S., Peñuñuri F., Optimizing functionals using differential evolution, *Engineering Applications of Artificial Intelligence*, 97, 104086, 2021.
22. Karimzadeh Parizi M., Keynia F., OWMA: An improved self-regulatory woodpecker mating algorithm using opposition-based learning and allocation of local memory for solving optimization problems, *Journal of Intelligent & Fuzzy Systems*, 40 (1), 919-946, 2021.
23. Piotrowski A. P., Napiorkowski J. J., Piotrowska A. E., Population size in particle swarm optimization, *Swarm and Evolutionary Computation*, 58, 100718, 2020.
24. Yang J., Qu L., Shen Y., Shi Y., Cheng S., Zhao J., Shen X., Swarm intelligence in data science: applications, opportunities and challenges, *International Conference on Swarm Intelligence*, 3-14, Springer, 2020.
25. Mahdi M. N., Mohamed Zabil M. H., Ahmad A. R., Ismail R., Yusoff Y., Cheng L. K., Happala Naidu H., Software project management using machine learning technique-A Review, *Applied Sciences*, 11 (11), 5183, 2021.
26. Kaur M., Kaur G., Sharma P. K., Jolfaei A., Singh D., Binary cuckoo search metaheuristic-based supercomputing framework for human behavior analysis in smart home, *The Journal of Supercomputing*, 76, 2479-2502, 2020.
27. Sabonchi A. K. S., Obaid Z. H., Ensuring Information Security in Smart Door Lock Systems Using the Cuckoo Search Algorithm, *Journal of Cybersecurity*, 4 (4), 2022.
28. Xu M., Cao L., Lu D., Hu Z., Yue Y., Application of Swarm Intelligence Optimization Algorithms in Image Processing: A Comprehensive Review of Analysis, Synthesis, and Optimization, *Biomimetics*, 8 (2), 235, 2023.
29. Shehadeh H. A., A hybrid sperm swarm optimization and gravitational search algorithm (HSSOGSA) for global optimization, *Neural Computing and Applications*, 33 (18), 11739-11752, 2021.
30. Kouziokas G., *Swarm intelligence and evolutionary computation: theory, advances and applications in machine learning and deep learning*, CRC Press, 2023.
31. Islam J., Vasant P. M., Negash B. M., Laruccia M. B., Myint M., Watada J., A holistic review on artificial intelligence techniques for well placement optimization problem, *Advances in Engineering Software*, 141, 102767, 2020.
32. Garcia Marquez F. P., Peinado Gonzalo A., A comprehensive review of artificial intelligence and wind energy, *Archives of Computational Methods in Engineering*, 29 (5), 2935-2958, 2022.
33. Tao H., Hameed M. M., Marhoon H. A., Zounemat-Kermani M., Heddad S., Kim S., Yaseen Z. M., Groundwater level prediction using machine learning models: A comprehensive review, *Neurocomputing*, 489, 271-308, 2022.
34. Khondakar K. R., Kaushik A., Role of wearable sensing technology to manage long COVID, *Biosensors*, 13 (1), 62, 2022.
35. Santamaria-Granados L., Mendoza-Moreno J. F., Ramirez-Gonzalez G., Tourist recommender systems based on emotion recognition—a scientometric review, *Future Internet*, 13 (1), 2, 2020.
36. Zhou G., Zhou Y., Deng W., Yin S., Zhang Y., Advances in teaching-learning-based optimization algorithm: A comprehensive survey, *Neurocomputing*, 2023.
37. Abdolrazzagah-Nezhad M., Radgohar H., Salimian S. N., Enhanced cultural algorithm to solve multi-objective attribute reduction based on rough set theory, *Mathematics and Computers in Simulation*, 170, 332-350, 2020.
38. Al-Tirawi A., Sustainable Learning in Cultural Algorithms Using Common Value, (Doktora Tezi, Wayne State University), 2021.
39. Goli A., Khademi-Zare H., Tavakkoli-Moghaddam R., Sadeghieh A., Sasanian M., Malekalipour Kordestanizadeh R., An integrated approach based on artificial intelligence and novel meta-heuristic algorithms to predict demand for dairy products: a case study, *Network: Computation in Neural Systems*, 32 (1), 1-35, 2021.
40. Achouri F., Khatir A., Smahi Z., Capozucca R., Ouled Brahim A., Structural health monitoring of beam model based on swarm intelligence-based algorithms and neural networks employing FRF, *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 45 (12), 621, 2023.
41. Xu T., Zhao F., Tang J., Du S., Jonrinaldi, A knowledge-driven monarch butterfly optimization algorithm with self-learning mechanism, *Applied Intelligence*, 53 (10), 12077-12097, 2023.
42. Zhang X., Ahmed R. R., A queuing system for inert construction waste management on a reverse logistics network, *Automation in Construction*, 137, 104221, 2022.
43. Zhang Y., Bai R., Qu R., Tu C., Jin J., A deep reinforcement learning based hyper-heuristic for combinatorial optimisation with uncertainties, *European Journal of Operational Research*, 300 (2), 418-427, 2022.
44. Ikeda S., Nagai T., A novel optimization method combining metaheuristics and machine learning for daily optimal operations in building energy and storage systems, *Applied Energy*, 289, 116716, 2021.
45. Fan Q., Li Y., Yao Y., Cohn J., Liu S., Xu Z., Cianfrocco M., CryoRL: Reinforcement learning enables efficient cryo-EM data collection, *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 7892-7902, 2024.
46. Canpolat O., Demir H. İ., Erden C., Meta-Heuristic Algorithms for Integrating Manufacturing and Supply Chain Functions, Available at SSRN 4598690.
47. Mbarek M. A., A robust multi-task Representation Learning methodology for sustainable management of Radio Access Networks, (Doktora Tezi, Politecnico di Torino), 2023.
48. Saffari A., Khishe M., Zahirri S. H., Fuzzy-ChOA: An improved chimp optimization algorithm for marine mammal classification using artificial neural network, *Analog Integrated Circuits and Signal Processing*, 111 (3), 403-417, 2022.
49. Xu T., Zhao F., Tang J., Du S., Jonrinaldi, A knowledge-driven monarch butterfly optimization algorithm with self-learning mechanism, *Applied Intelligence*, 53 (10), 12077-12097, 2023.
50. Mohamed T. I., Oyelade O. N., Ezugwu A. E., Automatic detection and classification of lung cancer CT scans based on deep learning and ebola optimization search algorithm, *PLOS ONE*, 18 (8), e0285796, 2023.
51. Akbulut U., Çiğçi M. A., İşler B., Aslan Z., Learnings M., Comparison of different machine learning techniques in river flow prediction, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 40(1), 467-485, 2025.
52. Çavga S. H., Performance of neural networks and heuristic models for disease prediction from liver enzymes: Application to biochemistry device output, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 39 (4), 2263-2270, 2024.
53. Üzülmüş S., Çiğçi M. A., Early diagnosis of lung cancer using deep learning and uncertainty measures, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 39 (1), 385-400, 2024.