

Hata Fonksiyonlarının Meta-Heuristik Algoritmalara Etkisi ve Transfer Fonksiyonu Optimizasyonu

Şehmus FİDAN¹

¹Teknik Bilimler MYO/Elektronik Programı, Batman Üniversitesi, Türkiye

*(sehmus.fidan@batman.edu.tr)

Özet – Sistem tanımlama uygulamaları kontrol mühendisliği alanının önemli bir alt dalıdır. Sistem tanımlama temel olarak bir sistemin giriş/çıkış verilerine bağlı olarak matematiksel model oluşturma sürecidir. Lineer kontrolör geliştiren araştırmacılar sistemin iyi bir şekilde bilinmesine ihtiyaç duymaktadır ancak optimal transfer fonksiyonu belirlenirken minimum komplekslik ve maksimum performanslı bir model elde etmek her zaman kolayca mümkün olmamaktadır. Bu problemi çözmek ve optimal transfer fonksiyonlarını elde etmek için meta-heuristik (MH) algoritmaları kullanmak mümkündür. Bununla birlikte meta-heuristik algoritmalarıyla optimal çözümü araştırırken hata fonksiyonunu doğru seçmek performansı etkilemektedir. Literatürde yaygın olarak bilinen hatanın karesinin integrali (Integral of Square Error-ISE), hatanın mutlak değerinin integrali (Integral of the Absolute Value of Error -IAE), hatanın zaman ağırlıklı karesinin integrali (Integral of Time-Weighted Squared Error-ITSE) ve hatanın mutlak değerinin zamanla çarpım integrali (Integral of Time Multiply Absolute Error-ITAE) gibi yöntemler bulunmaktadır. Bu makalede zaman cevabı tabanlı bir hata fonksiyonu (Transient Response Error-TRE) önerilmiş ve IAE, ISE, ITSE ve ITAE gibi yöntemler ile karşılaştırılmıştır. Hataların performans etkisini incelemek ve optimal transfer fonksiyonunu belirlemek için çiçek tozlaşma (Flower Pollination Algorithm-FPA), karınca aslanı algoritması (Ant Lion Optimization-ALO), güve-alev (Moth-Flame Optimizer-MFO) su döngüsü (Water Cycle Algorithm-WCA) ve mercan resifi (Coral Reef Optimizer-CRO) optimizasyon algoritmaları önerilmiştir. Önerilen hata fonksiyonları ve meta-heuristik algoritmaların sonuçlarının değerlendirilmesi için hava ısıtıcısı deney setinden alınan giriş/çıkış verileri kullanılmış ve karşılaştırmalar yapılmıştır. Yapılan çalışmalar sonucunda önerilen hata fonksiyonunun meta-heuristik algoritmalarının performansını arttırdığı görülmüştür.

Anahtar Kelimeler – Hata Fonksiyonları, Sistem Tanımlama, Meta-Heuristik Algoritma, Transfer Fonksiyonu

I. GİRİŞ

Doğadan ilham alan meta-heuristik algoritmaların kullanımı, karmaşık ve çok boyutlu optimizasyon problemlerini çözmek için giderek artan bir ilgi görmektedir. Meta-heuristik algoritmalar (MH), global çözüme ulaşmak için genellikle belirli başlangıç noktalarından başlayarak ve problem alanına özgü olmayan bir dizi kural veya yöntem kullanırlar [1]. Birçok çeşitli meta-heuristik algoritması mevcut olmasına rağmen, örnek olarak yapay arı koloni algoritması [2], karınca kolonisi optimizasyonu [3], ve

parçacık sürü optimizasyonu [4] gibi algoritmalar verilebilir. Her bir MH algoritması, kendisini model aldığı doğal sistemlerden farklı karakteristiklere sahip ve çeşitli optimizasyon problemlerine uyarlanabilir [5]. Farklı amaçlarla kullanılan meta-heuristik algoritmalar çeşitli sistemlerin matematiksel modelini elde etmek içinde kullanılabilir [6].

Sistem tanımlama, bir sistemin veya sürecin giriş/çıkış verilerinden yararlanarak matematiksel bir modelini oluşturma veya belirleme sürecidir. Bu süreç, sistemin davranışını, bileşenlerini ve

etkileşimlerini açıklayan bir matematiksel denklem veya model geliştirmeyi içerir. Sistem tanımlamada, parametrik modeller sistemin belirli bir matematiksel modele sahip olduğunu varsayarak kullanılırken, parametrik olmayan modeller, modele bağımlı olmadan sistemin davranışlarını tahmin etmek için tercih edilir. Parametrik modeller daha iyi modelleme ve kontrol sağlar ve daha esnek bir yaklaşım sunar. Model seçimi, belirli koşullar ve gereksinimler göz önüne alınarak yapılmalıdır [7]. MH algoritmalar, bir sistemin matematiksel modelinin optimal parametrelerini elde etmek için kullanılabilir [8]. Bu çalışmayla ikinci dereceden transfer fonksiyonu modeli önerilmiş dört adet parametrenin optimizasyonu yapılmıştır.

Çiçek tozlaşma algoritması (FPA), bitki tozlaşması sürecinden esinlenerek tasarlanmıştır. Bitki tozlaşması, bitkilerin üreme sürecinde polenin çiçekler arasında taşınması ve döllenme işlemini gerçekleştirmesi anlamına gelir [9]. FPA algoritması esnek giriş sisteminin modelinin oluşturulmasında [10], sonsuz dürtü tepkisi (Infinite Impulse Response-IIR) sistemi [11], sıcaklık süreç modelinin elde edilmesi [12], doğru akım motorlarında [13], fırçasız doğru akım motorlarında [14], kısmi kesirli kaotik sistem parametrelerinin tahmininde [15-16]. Karınca aslanı algoritması (ALO), bu canlıların doğadaki avlanma mekanizmasını taklit eder [17]. Hidrolik türbin tahrik sistemi [18], IIR filtresinin modelinin belirlenmesi [19] vb. sistem tanımlama amaçlı kullanılmıştır.

Güve-alev (MFO) optimizasyon algoritması, ışık kaynakları etrafında güvelerin uçuşundan etkilenecek geliştirilen bir algoritmadır [20]. MFO algoritmaları tek fazlı inverterlerde [21] doğrusal olmayan sistem tanımlamada [22], iletim hattı parametrelerinin tanımlanmasında [23], LCL filtrenin model parametrelerinin belirlenmesinde [24] kullanılmıştır. Eskandar ve ark., hidrolojik döngüden esinlendikleri su döngüsü algoritmasını (WCA) tanıtmışlardır [25]. WCA, PV sistem model parametrelerinin belirlenmesinde [26], hibrit PV/Rüzgâr sisteminin modelinin elde edilmesinde [27] kullanılmıştır. Mercan resifi optimizasyon (Coral Reef Optimizer-CRO) algoritması, mercanların bulunduğu resifte diğer mercanlarla olan mücadeleyi modellemektedir [28]. CRO algoritması Magnetostriktive tahrik ekipmanlarının model parametrelerinin belirlenmesinde [29] ve

adaptif IIR tabanlı sistem tanımlama amaçlı olarak çalışılmıştır [30]. Bu çalışma özet olarak;

1- Giriş/çıkış verilerini kullanarak FPA, ALO, MFO, WCA ve CRO algoritmaları bir transfer fonksiyonu elde etmek için kullanılmıştır.

2- Bazı algoritmaların, kabul edilebilir veya optimal çözüme ulaşması uzun zaman alabilir. Ancak, gerçek dünya uygulamalarında genellikle sınırlı bir zaman dilimi içinde çözüm bulunması gerekmektedir. Ayrıca uzun süre çalışan algoritmalar, hesaplama maliyetleri açısından sıkıntılar barındırmaktadır. Bu sebeplerden dolayı 25 saniyelik zaman kısıtlaması altında performanslar incelenmiştir.

3- Hata fonksiyonları (IAE, ISE, ITSE, ITAE), kontrol sistemlerinde birinci dereceden transfer fonksiyonuna sahip PID kontrolör parametrelerinin belirlenmesinde önemlidir. Önerilen sistem ikinci dereceden bir transfer fonksiyonu olduğu için bu hata fonksiyonlarını sistem tanımlama amaçlı olarak da kullanmak mümkündür. En uygun hata fonksiyonunu seçerek, sistem performansı optimize edilebilir ve istenmeyen dalgalanmaları veya aşırı tepkileri azaltabilir. Bu çalışmayla IAE, ISE, ITSE, ITAE hata fonksiyonlarına ek olarak geçici durum cevabına bağlı (TRE) hata fonksiyonu önerilmiş ve performans üzerindeki etkileri incelenmiştir.

4-Yapılan karşılaştırmalar sonucunda 25 saniye içinde WCA algoritması diğer algoritmalara kıyasla daha yüksek performans göstermiştir. Ayrıca TRE hata fonksiyonunun performansı arttırdığı görülmüştür.

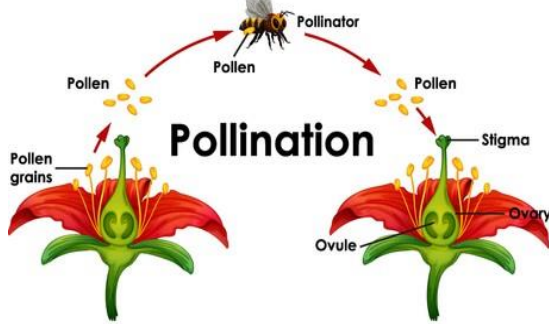
5- Bu araştırmada, Python programlama dili; maliyetsiz oluşu, geniş geliştirici topluluğu ve düşük kapasiteli donanımlarda bile etkin çalışabilme yeteneği sebebiyle seçilmiştir.

II. MATERYAL VE YÖNTEM

A. Çiçek Tozlaşma Algoritması

Çiçek tozlaşma algoritması, çiçekli bitkilerin üremesinden ilham alan bir optimizasyon yöntemidir. Doğada çiçeklerin tozlaşması, bitkilerin üremesi ve genetik çeşitliliğin artması için hayati bir süreçtir. Tozlaşma, polenlerin erkek organlardan dişi organlara taşınması işlemidir ve bu sayede döllenme gerçekleşir. Tozlaşma süreci iki ana yolla gerçekleşebilir: biyotik ve abiyotik. Biyotik tozlaşma yönteminde, polenler canlı polinatörler (arılar, kelebekler, kuşlar, sinekler vb.) tarafından taşınır. Bu süreç, genellikle polinatörlerin çiçeklerden nektar veya polen

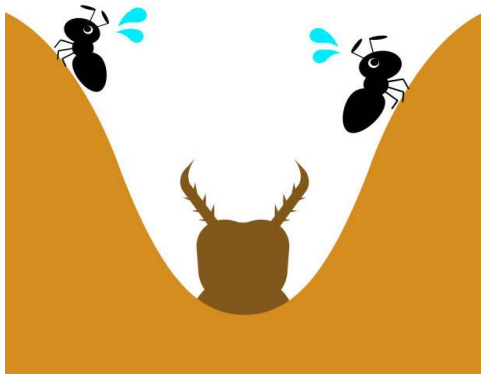
toplaması sırasında gerçekleşir. Abiyotik tozlaşma yönteminde, polenler rüzgâr, su veya yer çekimi gibi doğal unsurlar tarafından taşınır. Abiyotik tozlaşma genellikle rastgele ve daha az hedeflenmiş bir süreçtir [9]. Şekil 1 çiçek tozlaşma algoritmasının stratejisini göstermektedir.



Şekil 1. Çiçek tozlaşma stratejisi

B. Karınca Aslanı Algoritması

Karınca Aslanı Algoritması (ALO), bu avlanma sürecinden ilham alarak, karınca aslanlarının tuzak kurma ve avlanma stratejilerini matematiksel bir modelle simüle eder. Algoritma, karınca aslanlarının (optimizasyon arayıcıları) ve karıncaların (çözüm adayları) etkileşimlerini kullanarak, bir optimizasyon problemi için en uygun çözümleri arar. Karınca aslanları, çözüm alanında rastgele hareket eden karıncaları yakalamaya çalışırken, karıncaların hareketleri potansiyel çözümleri temsil eder. Algoritma, karınca aslanlarının avlarını yakalama stratejilerini taklit ederek, çözüm alanında global ve yerel arama yapar [17]. Karınca aslanlarının avlanma mekanizması Şekil 2’de gösterilmektedir.



Şekil 2. Karınca aslanı avlanma stratejisi

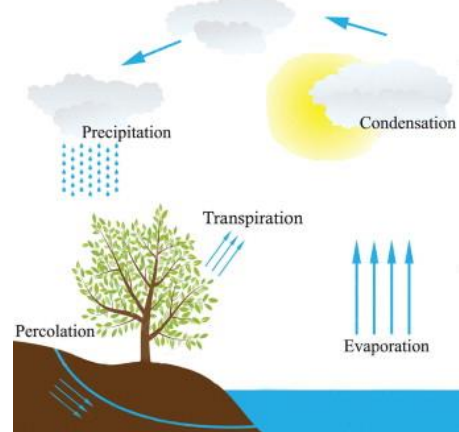
C. Güve-Alev Optimizasyon Algoritması

MFO, güvelerin ışık kaynaklarına karşı sergiledikleri spiral hareketlerden esinlenerek geliştirilmiş bir optimizasyon algoritmasıdır. Güvelerin bu davranışı, ışık kaynaklarına doğru

uçarken belirli bir açıyı sabit tutmaya çalışmalarından kaynaklanır. Bu, güvelerin ışığa spiral bir yol izleyerek yaklaşmalarına neden olur. MFO algoritması da bu fikri kullanarak, popülasyon içerisindeki güvelerin pozisyonlarını değiştirerek en iyi çözüme ulaşmayı hedefler [20].

D. Su Döngüsü Algoritması (WCA)

Bu algoritma, gerçek dünyada nehirlerin oluşumundan esinlenerek tasarlanmıştır. WCA algoritmasında, yağmur damlaları ve nehirler potansiyel çözüm adayları olarak kullanılır. Her yağmur damlası ve nehir, çözüm uzayında bir noktayı temsil eder. Deniz, anlık olarak bulunan en iyi çözümü temsil eder. Nehirler ve yağmur damlaları, bu en iyi çözüme doğru hareket eder. Yağmur damlalarının buharlaşması ve yağış olarak yeryüzüne geri dönmesi, algoritmanın arama mekanizmasını oluşturur. Bu süreç, çözüm uzayında yeni çözüm adaylarının oluşturulmasını sağlar. Nehirler, en iyi çözüme doğru akar. Bu, algoritmanın çözüm uzayında global arama yapmasını sağlar. Nehirlerin akış yönü ve hızı, çözüm uzayındaki en iyi çözüme bağlıdır. Yağmur damlaları, çözüm uzayında rastgele hareket eder. Bu durum, algoritmanın yerel arama yapmasını ve yerel optimumlardan kaçınmasını sağlar [25]. Şekil 3 su döngüsü sürecine örnek olarak gösterilebilir.



Şekil 3. Su döngüsü temsili gösterimi

E. Mercan Resifi Algoritması

CRO, mercan resiflerinin evrimsel süreçlerini modelleyen bir optimizasyon algoritmasıdır. Bu algoritma, çözüm adaylarını temsil eden bir ızgara üzerinde çalışır ve bu adaylar mercanlar olarak adlandırılır. Algoritma, cinsel ve eşeysiz üreme yoluyla yeni çözüm adayları oluşturur ve bu yeni adaylar rastgele yerleştirilir. Düşük performanslı çözümler elenirken, en uygun çözümler hayatta

kalır ve resifte yer bulur. Bu süreç, karmaşık problemlerin en uygun çözümlerini bulmak için tekrarlanır [28].

F. Sistem Tanımlama

Kontrol mühendisliğinin önemli bir alt alanı olan sistem tanımlama, giriş ve çıkış verilerinden matematiksel denklem üretme amacıyla kullanılmaktadır [31-33]. Sistem tanımlama amaçlı kullanılan çeşitli yöntemler vardır. Bununla birlikte siyah-kutu modelleme yaygın olarak kullanılmaktadır. Bu modelleme yönteminde sistem için genel bir denklem önerilir ve bu denkleme ilişkin parametreler tahmin edilmeye çalışılır. Kara kutu sistem tanımlama Denklem 1'de sunulmuştur.

$$y(t) = f[y_1(t-1), \dots, y_1(t-n_{y_1}), \dots, y_r(t-n_{y_r}), \\ u_1(t-1), \dots, u_1(t-n_{u_1}), \dots, u_m(t-n_{u_m}), \\ e_1(t-1), \dots, e_1(t-n_{e_1}), \dots, e_r(t-1), \dots, e_r(t-n_{e_r})] \quad [1]$$

G. Hata Fonksiyonları

Hatanın karesinin integrali (Integral of Square Error-ISE), hatanın mutlak değerinin integrali (Integral of the Absolute Value of Error -IAE), hatanın zaman ağırlıklı kare integrali (Integral of Time-Weighted Squared Error-ITSE) ve hatanın zamanla çarpılmış mutlak değerinin integrali (Integral of Time Multiply Absolute Error-ITAE) gibi yöntemler, sistem parametrelerini belirlemek ve performanslarını incelemek için kullanılabilir [46]. ISE, belirli bir kontrol sistemi tarafından üretilen istenen sinyal ile gerçek sistem çıkışı arasındaki farkların karelerinin toplamını ifade eder. Daha düşük bir ISE değeri, kontrol sisteminin daha iyi performans gösterdiğini gösterir. IAE, referans sinyal ile gerçek sistem çıkışı arasındaki farkların mutlak değerlerinin toplamını temsil eder. Kare hata tabanlı metriklerin aksine, IAE hatanın büyüklüğünü doğrudan dikkate alır ve onu karelemez. ITSE, her zaman adımında hatanın karesinin toplamını içerir, burada hata tipik olarak istenen referans sinyal ile gerçek sistem çıkışı arasındaki farktır. ITAE, her zaman adımında mutlak hatanın ve zamanın çarpımını dikkate alarak kontrol sistemi performansını değerlendiren integral tabanlı bir metriktir. Yalnızca mutlak hataya (IAE gibi) veya kare hataya (ISE gibi) odaklanan metriklerin aksine, ITAE hatanın büyüklüğünü hatanın süresiyle birleştirir [8]. Geri besleme sisteminde bir süre boyunca birikmiş

hatayı ölçmek için kullanılan zamanla çarpılmış mutlak hatanın integrali (ITAE), sistemin performansını ölçmek ve iyileştirilebilecek alanları belirlemek için kullanılır. Bu nedenle, bu makalede hatayı ölçmek için önerilmektedir. Tablo 1, yaygın olarak bilinen hata amaç fonksiyonlarının isimlerini ve formüllerini sunmaktadır. Bu tabloda ayrıca önerilen hata fonksiyonu sunulmuştur. Bu tabloda e hata, t zaman değeri, T_r yükselme zamanı, T_s oturma zamanı olarak ifade edilmektedir.

Tablo 1. Hata fonksiyonları

Hata Fonksiyonları	Formül
ISE	$ISE = \int_0^{\infty} e(t)^2 dt$
IAE	$IAE = \int_0^{\infty} e(t) dt$
ITSE	$ITSE = \int_0^{\infty} te^2(t) dt$
ITAE	$ITAE = \int_0^{\infty} t e(t) dt$
TRE	$O = \max(e - 1)$ $\beta = T_s, \alpha = T_r$ $FRE = e\beta + O\alpha$

BULGULAR

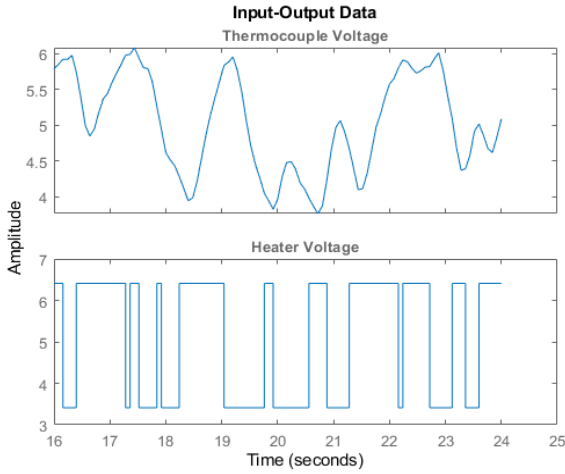
A. Ön Analiz

Bu çalışmada kullanılan bilgisayarın 2.1 GHz hızında 16 çekirdekli Intel Core i7-13700 CPU, 32 Gb RAM ve 1 TB SSD özelliğindedir. Yazılım olarak Python 3.8, Mealpy 3.0, Pandas 1.4.2 ve Control 0.9.2 kütüphaneleri kullanılmıştır. Veri seti için laboratuvar ölçeğinde hava ısıtıcısı deney setinden toplanan veriler kullanılmıştır [34]. Bu setin veri setinin giriş/ çıkış verilerine ilişkin gerilim değerleri Şekil 4'te verilmiştir.

Tablo 2. MH algoritmalarının parametre ayarları

Algoritma	Populasyon Boyutu	Diğer Parametreler
MFO [9]	100	-
FPA [17]	100	$P_s = 0.8, levy_p = 0.1$
WCA [20]	100	$N_{sr} = 4, w_c = 2.0, d_{max} = 1e-6$
ALO [25]	100	-
CRO [28]	100	$P_o = 0.4, F_b = 0.9, F_a = 0.1,$ $F_d = 0.1, G_{CR} = 0.1, Y_{min} = 0.02$ $Y_{max} = 0.2, n_{trials} = 3$

Tablo 2’de FPA, ALO, MFO, WCA ve CRO için ön parametreleri sunulmuştur. Bu parametreler algoritmalar çalıştırılmadan önce ayarlanmıştır. Seçilen değerler ilgili kaynaklardan alınmıştır. Rastgele başlangıç değerlerine sahip olan MH algoritmalarının optimize ettiği parametreler farklılık gösterebilmektedir. Bu sebeple ortalama bir performans değerlendirmesi yapmak gerekmektedir. Bu sebeple makalede bağımsız çalışma sayısı 25 olarak belirlenmiştir. Alt bölümlerde çeşitli hata fonksiyonları dikkate alınarak elde edilen optimal çözümlere ait performans sonuçları sunulmuştur.



Şekil 4. Isıtıcı girişine uygulanan gerilime karşılık çıkışta ölçülen termokupl gerilimi

B. Güvenilirlik Analizi ve Sınırlılıklar

MH algoritmalarının zaman kısıtlatmalı olarak çalışması, pratik uygulanabilirlik, kaynak kullanımı, hesaplama karmaşıklığı ve yakınsama süreçlerinin incelenmesi açısından önemlidir. Bununla birlikte performanstan bir miktar ödün vererek önemli kazanımlar elde edilebilir. Bu sebeple bu çalışmada 25 sn’lik bir zaman kısıtlaması altında sonuçlar incelenmiştir.

Tablo 3 farklı hata fonksiyonları için MFO algoritmasının tanımlayıcı istatistik değerlerini göstermektedir. TRE hata fonksiyonu için ortalama R^2 değeri 0.290518 olarak hesaplanmıştır. IAE, ISE, ITAE ve ITSE için ortalama R^2 hata değerleri sırasıyla 0.273276, 0.362068, 0.366978, 0.314182 olarak hesaplanmıştır.

Tablo 4 farklı hata fonksiyonları için FPA algoritmasının tanımlayıcı istatistik değerlerini göstermektedir. TRE hata fonksiyonu için ortalama R^2 değeri 0.664446 olarak hesaplanmıştır. IAE, ISE, ITAE ve ITSE için ortalama R^2 hata değerleri sırasıyla 0.20963, 0.504772, 0.530679, 0.487591 olarak hesaplanmıştır. Tüm çalışma parametreleri standart tutulmuş olmasına rağmen önerilen TRE hata fonksiyonunun ortalama R^2 değerlerinin daha başarılı olduğu açıktır. Ayrıca TRE hata fonksiyonu kullanıldığında maksimum R^2 değerinin de diğerlerine göre daha yüksek olduğu görülmüştür. Burada TRE hata fonksiyonunun R^2 performansının üzerinde olumlu bir etkisi olduğu gibi FPA algoritmasının yapısının da etkisi bulunmaktadır.

Tablo 3- Farklı hata fonksiyonları için MFO algoritmasının tanımlayıcı istatistik değerleri

	IAE	ISE	ITAE	ITSE	TRE
Ortalama R^2	0.273276	0.362068	0.366978	0.314182	0.290518
Standart Sapma	0.23682	0.260207	0.255695	0.24253	0.217804
Minimum R^2	0.03695	0.08518	0.104675	0.03761	0.01615
25%	0.131219	0.178236	0.17854	0.151424	0.132947
50%	0.173798	0.256251	0.303985	0.199046	0.201341
75%	0.293333	0.484883	0.436198	0.433537	0.475373
Maksimum R^2	0.93507	0.896	0.98338	0.88831	0.6909

Tablo 4- Farklı hata fonksiyonları için FPA algoritmasının tanımlayıcı istatistik değerleri

	IAE	ISE	ITAE	ITSE	TRE
Ortalama R^2	0.20963	0.504772	0.530679	0.487591	0.664446
Standart Sapma	0.16095	0.235217	0.22635	0.216228	0.233458
Minimum R^2	0.47081	0.12457	0.00795	0.02962	0.07079
25%	0.598665	0.390638	0.363203	0.334733	0.562878
50%	0.72035	0.47677	0.573145	0.512	0.706135
75%	0.88745	0.662885	0.70685	0.677538	0.8505
Maksimum R^2	0.584716	0.91576	0.93268	0.7561	0.94524

Tablo 5, IAE, ISE, ITAE, ITSE ve TRE hata fonksiyonları için WCA algoritmasının tanımlayıcı istatistik değerlerini göstermektedir. TRE hata fonksiyonu için ortalama R^2 değeri 0.943586 olarak hesaplanmıştır. IAE, ISE, ITAE ve ITSE için ortalama R^2 hata değerleri sırasıyla 0.943285, 0.941415, 0.936602, 0.901111 hesaplanmıştır.

WCA algoritması 25 saniyelik çözüm süresi içinde farklı hata fonksiyonları önerilse de oldukça başarılı performans göstermiştir. Ancak önerilen hata fonksiyonunun küçükte olsa ortalama R^2 değerlerinin daha başarılı olduğu ortaya çıkmıştır. TRE hata fonksiyonu ile en yüksek R^2 değeri 0.95759 de elde edilmiştir. Ayrıca TRE'ye ait minimum R^2 değerinin de yüksek olması diğer önemli bir avantajdır.

IAE, ISE, ITAE, ITSE ve TRE hata fonksiyonları için ALO algoritmasının tanımlayıcı istatistik değerleri Tablo 6'da görülmektedir. TRE hata fonksiyonu için ortalama R^2 değeri 0.930758 olarak hesaplanmıştır. Bu değer diğer hata fonksiyonlarına göre daha yüksektir. IAE (0.943285), ISE (0.941415), ITAE (0.936602) ve ITSE (0.901111) hata fonksiyonları da yüksek ortalama R^2 değerine sahiptir. Burada TRE için önemli bir diğer ayrıntıda minimum R^2 (0.86429)

değerinin diğer algoritmalara göre yüksek olmasıdır.

Tablo 7'de IAE, ISE, ITAE, ITSE ve TRE fonksiyonları için CRO algoritmasının tanımlayıcı istatistik değerleri sunulmuştur. Önerilen hata fonksiyonu için ortalama R^2 değeri 0.751193 olarak hesaplanmıştır. Bu performans değeri IAE (0.590838), ISE (0.641974), ITAE (0.620716) ve ITSE (0.614154) hata fonksiyonlarına göre daha yüksektir. Ayrıca TRE hata fonksiyonu diğer hata fonksiyonlarına göre daha yüksek minimum R^2 değerine sahiptir.

Yapılan analizler neticesinde WCA algoritmasının yapısı sistem tanımlama problemlerinin çözümü için diğer algoritmalara göre daha başarılı olduğu belirlenmiştir. Elbette Tablo 2'de verilen ön parametrelerin değişiminin problemi çözme performansına etkisi olacaktır ancak bu çalışmanın temel amacı bu parametrelerin optimize edilmesi değildir. Bu sebeple literatürde tanımlanan ön tanım değerleri kullanılmıştır. Kullanılan hata fonksiyonlarından başka fonksiyonlar önermekte mümkündür. Ancak burada önemli olan bu fonksiyonların performansa etkisinin olduğunu anlamaktır. Bu durumda WCA algoritması ve önerilen TRE algoritması oldukça yeterli sayılabilecek bir performans sergilemiştir.

Tablo 5- Farklı hata fonksiyonları için WCA algoritmasının tanımlayıcı istatistik değerleri

	IAE	ISE	ITAE	ITSE	TRE
Ortalama R^2	0.943285	0.941415	0.936602	0.901111	0.943586
Standart Sapma	0.013169	0.008464	0.014009	0.176996	0.012275
Minimum R^2	0.9095	0.93178	0.90951	0.150656	0.90951
25%	0.941755	0.93178	0.93178	0.93178	0.934345
50%	0.949655	0.94187	0.937855	0.94498	0.94095
75%	0.951035	0.950833	0.949648	0.950655	0.95089
Maksimum R^2	0.95176	0.9529	0.95126	0.95128	0.95759

Tablo 6- Farklı hata fonksiyonları için ALO algoritmasının tanımlayıcı istatistik değerleri

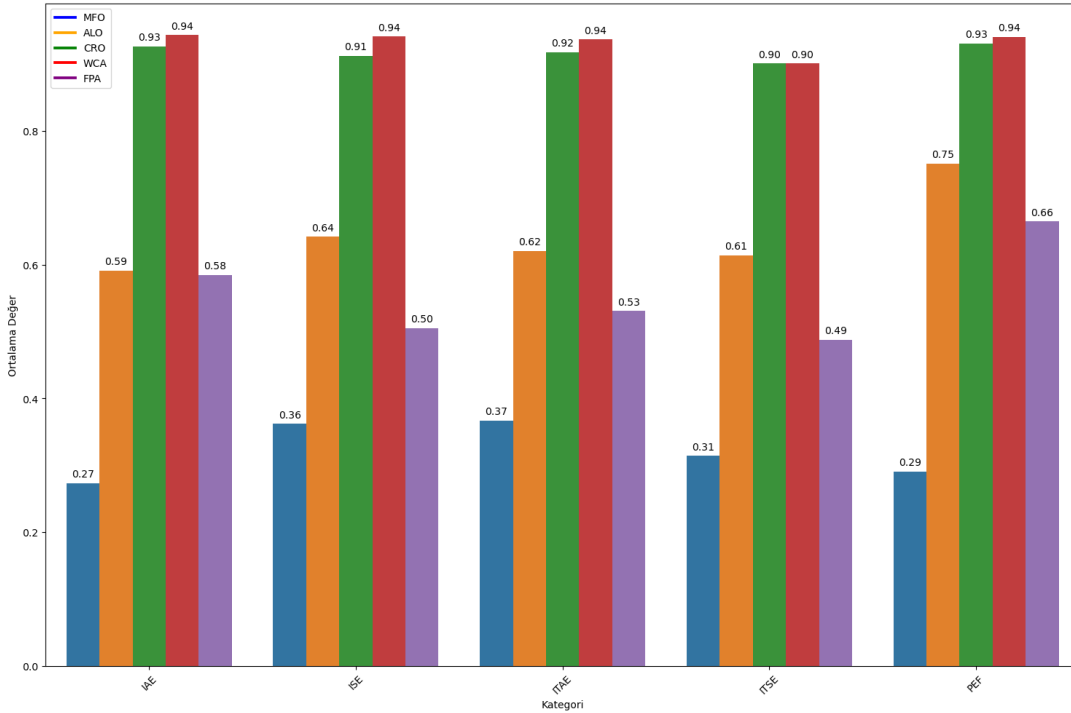
	IAE	ISE	ITAE	ITSE	TRE
Ortalama R^2	0.926664	0.912058	0.917809	0.900688	0.930758
Standart Sapma	0.032826	0.049698	0.069802	0.108422	0.020208
Minimum R^2	0.80247	0.73279	0.62757	0.48076	0.86429
25%	0.926218	0.905915	0.928275	0.923408	0.927098
50%	0.93767	0.92779	0.93622	0.936085	0.93556
75%	0.939985	0.937898	0.939595	0.940895	0.942848
Maksimum R^2	0.9496	0.9534	0.95163	0.94918	0.95351

Tablo 7- Farklı hata fonksiyonları için CRO algoritmasının tanımlayıcı istatistik değerleri

	IAE	ISE	ITAE	ITSE	TRE
Ortalama R ²	0.590838	0.641974	0.620716	0.614154	0.751193
Standart Sapma	0.213328	0.230931	0.221819	0.247292	0.120478
Minimum R ²	0.11327	0.05733	0.07641	0.13426	0.51034
25%	0.506738	0.531423	0.4648	0.458185	0.697905
50%	0.597695	0.66978	0.646195	0.70411	0.73157
75%	0.74142	0.835198	0.80527	0.805743	0.850065
Maksimum R ²	0.90156	0.93924	0.89435	0.92049	0.91921

Şekil 5'te IAE, ISE, ITAE, ITSE ve TRE hata fonksiyonlarının bar grafikleri sunulmuştur. Ayrıca farklı MH algoritmaları için performans (R²) değerleri gösterilmiştir. R² değerlerinin 1 değerine yakın olması istenir. Ortalama R² değerlerinin gösterildiği bu grafikte TRE hata fonksiyonu düşük performanslı MFO algoritması hariç diğerlerinde

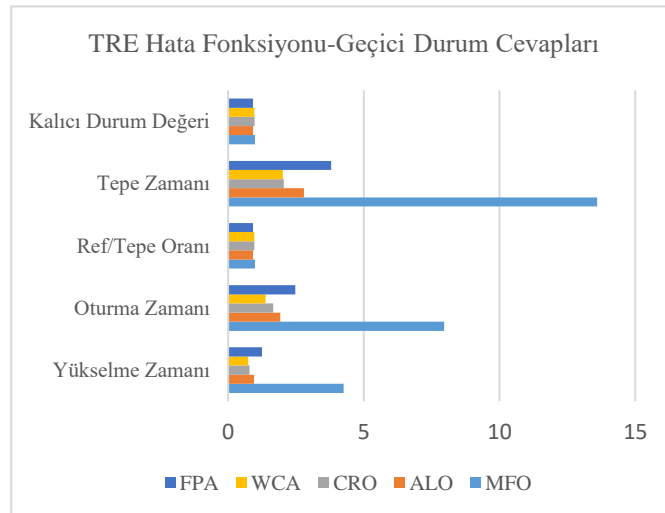
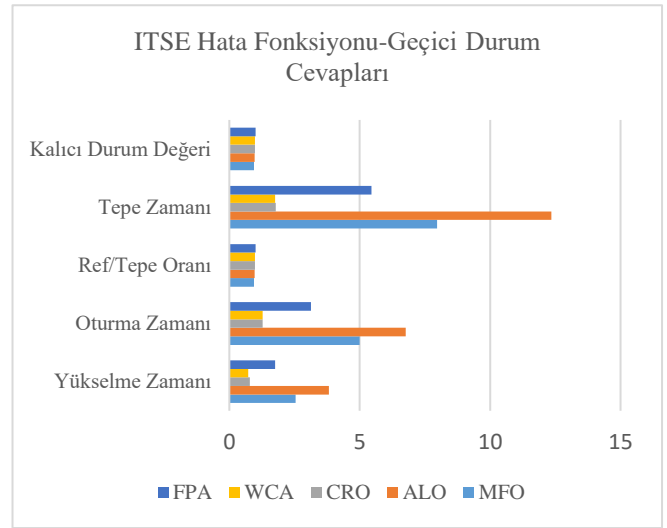
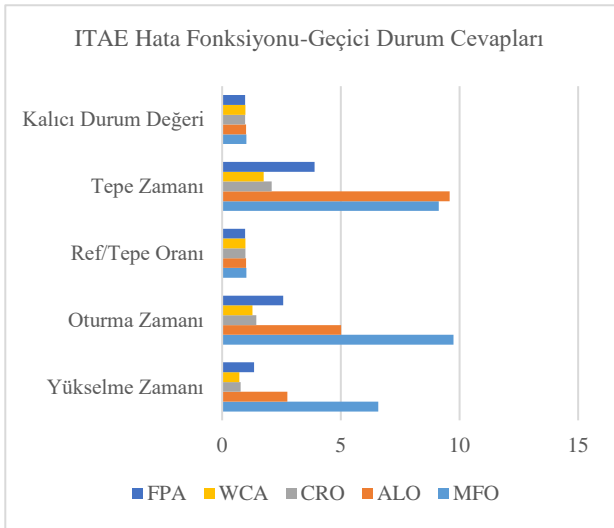
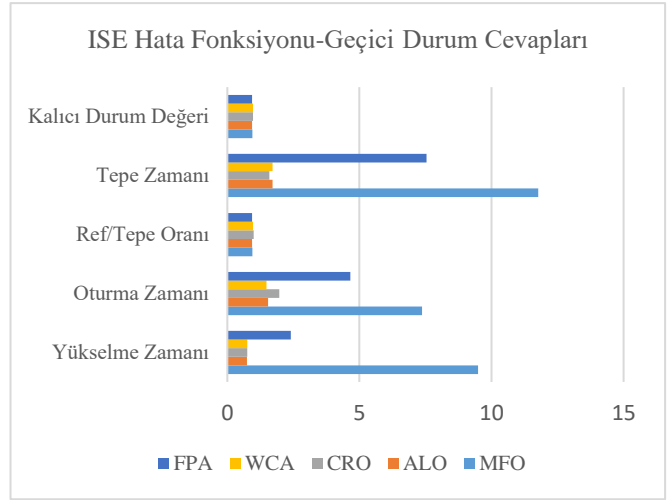
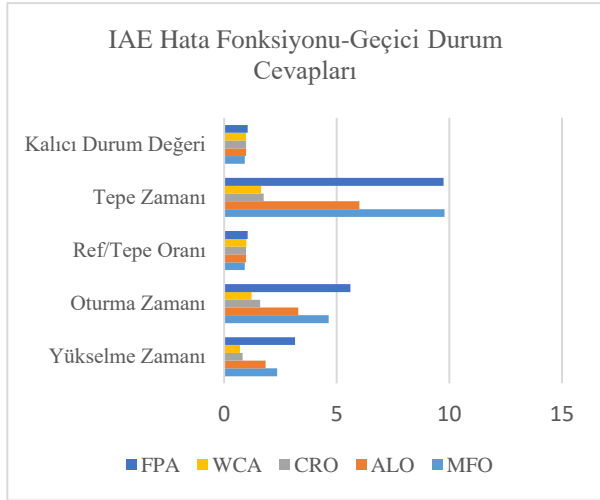
daha başarılı veya en azından benzer performansa sahip olduğu görülmektedir. Tablo 3-7'de daha detaylı istatistiksel veriler WCA algoritmasının performansının yüksek olduğunu göstermekle birlikte TRE fonksiyonu sayesinde daha iyi bir performansa ulaşmasına yardımcı olmuştur.



Şekil 5- MH algoritmalarının farklı hata fonksiyonlarında performans grafiği

Tablo 8-Zaman sınırlılıkları altında çözüm yeteneklerinin incelenmesi

Algoritmalar	MAPE	MAE	MSE	R ²	Zaman (Sn)	Transfer Fonksiyonları
ALO	0.06798	0.33375	0.16516	0.7045	46.14	$T_f(s) = \frac{-10.81s + 97.34}{s^2 + 60.67s + 100}$
CRO	0.06597	0.31821	0.15216	0.72777	45.32	$T_f(s) = \frac{-9.97s + 73.91}{s^2 + 43.19s + 75.97}$
FPA	0.06345	0.30803	0.14393	0.74248	46.23	$T_f(s) = \frac{-4.90s + 35.35}{s^2 + 27.67s + 36.32}$
MFO	0.06811	0.32467	0.16567	0.7036	45.18	$T_f(s) = \frac{-13.82s + 100}{s^2 + 62.52s + 100}$
WCA	0.02775	0.13349	0.02779	0.95029	45.89	$T_f(s) = \frac{-3.81s + 22.63}{s^2 + 9.61s + 23.12}$



Şekil 6- MH algoritmaları ve hata fonksiyonları için geçici duruma cevapları

C. Performans İndekslerinin Karşılaştırılması

Sistem tanımlama problemini çözmek için önerilen MH algoritmalarının optimize ettiği

transfer fonksiyonu ve bu modele ilişkin performans indeksleri Tablo 8'de sunulmuştur. Performans indeksi olarak MAPE, MAE, MSE, R^2 ve zaman (sn) gibi kriterler ele alınmıştır. Bu

bölümde algoritmalar bağımsız olarak 1 defa çalıştıktan sonra optimize edilmiş transfer fonksiyonları sunulmaktadır. Hata fonksiyonu olarak önerilen TRE dikkate alınmıştır. Tablo 8 zaman sınırlılığı altında (25 sn) TRE hata fonksiyonu ile ALO, CRO, FPA, MFO ve WCA algoritmaları kullanılarak optimize edilmiş transfer fonksiyonları sunulmaktadır.

Tek seferlik bağımsız çalışma sonucunda WCA (0,95029) algoritmasının performansı oldukça yüksek çıkmaktadır. ALO (0.7045), CRO (0.72777), FPA (0.74248), MFO (0.7036) algoritmaları ise daha düşük performans değerlerine ulaşmıştır.

D. Geçici ve Kalıcı Durum Cevapları

Şekil 6'da MH algoritmalarına karşılık farklı hata fonksiyonları için elde edilen transfer fonksiyonlarının geçici ve kalıcı zaman cevapları sunulmuştur. Şekil 1'de yükselme zamanı, oturma zamanı, aşım miktarı, tepe değeri ve tepe zamanı değerleri sunulmuştur. Bu değerler içinde özellikle minimum tepe zamanı ve oturma zamanı değerleri, kontrol sistemi tasarımında önemli bir rol oynamaktadır. IAE hata fonksiyonu için WCA algoritması en düşük tepe (0.69) değerine sahiptir. CRO algoritmasının da performansı yüksek olmakla birlikte WCA algoritması bir adım öne çıkmaktadır. ISE hata fonksiyonu için WCA algoritmasının tepe (1.71) ve oturma (1.48) zamanları birlikte değerlendirilecek olursa diğer algoritmalara göre daha öndedir. Burada CRO algoritması ile WCA algoritmasının cevapları benzer görünmektedir.

ITAE hata fonksiyonu için WCA algoritması tepe (1.27) ve oturma (1.75) zaman değerine sahiptir. Burada CRO algoritması ile WCA algoritmasının cevapları benzer görünmektedir. ITSE hata fonksiyonu içinde WCA ve CRO algoritmaları öne çıkmaktadır. WCA algoritması için yükselme zamanı (0,739), oturma zamanı (1,375), ref/tepe oranı (0,970), tepe zamanı (2,012) ve kalıcı durum değerlerine (0,968) sahiptir. Bu sonuçlar hata fonksiyonlarının performansını göstermekten çok MH algoritmaların geçici durumları hakkında bilgi vermektedir. Elde edilen sonuçlar MH algoritmalarından özellikle WCA ve CRO algoritmasının diğer algoritmalara nazaran daha başarılı olduğunu göstermektedir.

IV-TARTIŞMA VE SONUÇLAR

Bu çalışmada çiçek tozlaşma, karınca aslanı, güve-alev, su döngüsü ve mercan resifi tabanlı doğadan ilham alan meta-heuristik algoritmalar, transfer fonksiyonu elde etmek için kullanılmıştır. Bunun için hava ısıtıcısı deney setinden elde edilmiş giriş/çıkış verileri kullanılmış ve önerilen transfer fonksiyonunun parametreleri hesaplanmıştır. MH algoritmalar problem çözümünde ISE, IAE, ITSE, ITAE gibi hata fonksiyonları kullanılmaktadır. Bu çalışmada ise farklı olarak TRE adlı bir hata fonksiyonu önerilmiştir. Ele alınan MH algoritmalarının 25 saniyelik kısıtlama altında çalıştırılıp istatistiksel analizler sunulmuştur. Bu analizler sonucunda WCA algoritması diğer algoritmalara göre daha başarılı performans sergilemiştir. 5 farklı hata fonksiyonu içinde TRE genel anlamda diğer hata fonksiyonlarına göre R^2 performansını daha da arttırmıştır. WCA algoritması en yüksek performans değerine R^2 0.943586 ulaşmıştır. Ayrıca CRO ve WCA algoritmalarının geçici durum cevapları benzer görünmekle birlikte sistemi temsil etmek için oldukça yeterli durmaktadır.

KAYNAKLAR

- [1] D. Izci, "A novel modified arithmetic optimization algorithm for power system stabilizer design", *Sigma J. Eng. Nat. Sci.*, c. 40, sy 3, Art. sy 3, Eki. 2022.
- [2] D. Karaboga ve B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", *J. Glob. Optim.*, c. 39, sy 3, ss. 459-471, Kas. 2007, doi: 10.1007/s10898-007-9149-x.
- [3] M. Dorigo, M. Birattari, ve T. Stutzle, "Ant colony optimization", *IEEE Comput. Intell. Mag.*, c. 1, sy 4, ss. 28-39, Kas. 2006, doi: 10.1109/MCI.2006.329691.
- [4] J. Kennedy ve R. Eberhart, "Particle swarm optimization", içinde *Proceedings of ICNN'95 - International Conference on Neural Networks*, Kas. 1995, ss. 1942-1948 c.4. doi: 10.1109/ICNN.1995.488968.
- [5] X.-S. Yang, "Nature-inspired optimization algorithms: Challenges and open problems", *J. Comput. Sci.*, c. 46, s. 101104, Eki. 2020, doi: 10.1016/j.jocs.2020.101104.
- [6] S. Ding, Z. Shi, K. Chen, ve A. T. Azar, "Mathematical Modeling and Analysis of Soft Computing", *Math. Probl. Eng.*, c. 2015, s. e578321, Mar. 2015, doi: 10.1155/2015/578321.
- [7] Y. Ji, X. Jiang, ve L. Wan, "Hierarchical least squares parameter estimation algorithm for two-input Hammerstein finite impulse response systems", *J. Frankl. Inst.*, c. 357, sy 8, ss. 5019-5032, May. 2020, doi: 10.1016/j.jfranklin.2020.03.027.

- [8] A. Mohammadi, F. Sheikholeslam, ve S. Mirjalili, "Inclined planes system optimization: Theory, literature review, and state-of-the-art versions for IIR system identification", *Expert Syst. Appl.*, c. 200, s. 117127, Ağu. 2022, doi: 10.1016/j.eswa.2022.117127.
- [9] X.-S. Yang, "Flower Pollination Algorithm for Global Optimization", içinde *Unconventional Computation and Natural Computation*, J. Durand-Lose ve N. Jonoska, Ed., içinde Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2012, ss. 240-249. doi: 10.1007/978-3-642-32894-7_27.
- [10] A. A. M. Fadzli, M. S. Hadi, R. T. P. Eek, M. H. Ab. Talib, H. M. Yatim, ve I. Z. M. Darus, "PID Controller Based on Flower Pollination Algorithm of Flexible Beam System", içinde *Recent Trends in Mechatronics Towards Industry 4.0*, A. F. Ab. Nasir, A. N. Ibrahim, I. Ishak, N. Mat Yahya, M. A. Zakaria, ve A. P. P. Abdul Majeed, Ed., içinde Lecture Notes in Electrical Engineering. Singapore: Springer, 2022, ss. 173-183. doi: 10.1007/978-981-33-4597-3_17.
- [11] S. Singh, A. Ashok, T. K. Rawat, ve M. Kumar, "Optimal IIR system identification using flower pollination algorithm", içinde *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, Tem. 2016, ss. 1-6. doi: 10.1109/ICPEICES.2016.7853666.
- [12] C. Sompracha ve S. Rukkaphan, "Fractional-Order System Identification of Temperature Process Rig Control System using Flower Pollination Algorithm", içinde *2021 9th International Electrical Engineering Congress (iEECON)*, Mar. 2021, ss. 309-312. doi: 10.1109/IEECON51072.2021.9440322.
- [13] D. Puangdownreong, S. Hlungnamtip, C. Thammarat, ve A. Nawikavatan, "Application of flower pollination algorithm to parameter identification of DC motor model", içinde *2017 International Electrical Engineering Congress (iEECON)*, Mar. 2017, ss. 1-4. doi: 10.1109/IEECON.2017.8075889.
- [14] P. Khluabwannarat, A. Nawikavatan, ve D. Puangdownreong, "Fractional-Order Model Parameter Identification of BLDC Motor by Flower Pollination Algorithm", c. 13, 2018.
- [15] D. Yousri vd., "Fractional chaos maps with flower pollination algorithm for chaotic systems' parameters identification", *Neural Comput. Appl.*, c. 32, sy 20, ss. 16291-16327, Eki. 2020, doi: 10.1007/s00521-020-04906-7.
- [16] Y. Chen, D. Pi, ve B. Wang, "Enhanced global flower pollination algorithm for parameter identification of chaotic and hyper-chaotic system", *Nonlinear Dyn.*, c. 97, sy 2, ss. 1343-1358, Tem. 2019, doi: 10.1007/s11071-019-05052-z.
- [17] S. Mirjalili, "The Ant Lion Optimizer", *Adv. Eng. Softw.*, c. 83, ss. 80-98, May. 2015, doi: 10.1016/j.advengsoft.2015.01.010.
- [18] T. Tian, C. Liu, Q. Guo, Y. Yuan, W. Li, ve Q. Yan, "An Improved Ant Lion Optimization Algorithm and Its Application in Hydraulic Turbine Governing System Parameter Identification", *Energies*, c. 11, sy 1, Art. sy 1, Oca. 2018, doi: 10.3390/en11010095.
- [19] S. S. Nair, K. P. S. Rana, V. Kumar, ve A. Chawla, "Efficient Modeling of Linear Discrete Filters Using Ant Lion Optimizer", *Circuits Syst. Signal Process.*, c. 36, sy 4, ss. 1535-1568, Nis. 2017, doi: 10.1007/s00034-016-0370-z.
- [20] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm", *Knowl.-Based Syst.*, c. 89, ss. 228-249, Kas. 2015, doi: 10.1016/j.knosys.2015.07.006.
- [21] Z. Wu, D. Shen, M. Shang, ve S. Qi, "Parameter Identification of Single-Phase Inverter Based on Improved Moth Flame Optimization Algorithm", *Electr. Power Compon. Syst.*, c. 47, sy 4-5, ss. 456-469, Mar. 2019, doi: 10.1080/15325008.2019.1607922.
- [22] M. Canayaz, "Training Anfis System with Moth-Flame Optimization Algorithm", *Int. J. Intell. Syst. Appl. Eng.*, c. 7, sy 3, Art. sy 3, Eyl. 2019, doi: 10.18201/ijisae.2019355375.
- [23] M. S. Shaikh, S. Raj, R. Babu, S. Kumar, ve K. Sagrolikar, "A hybrid moth-flame algorithm with particle swarm optimization with application in power transmission and distribution", *Decis. Anal. J.*, c. 6, s. 100182, Mar. 2023, doi: 10.1016/j.dajour.2023.100182.
- [24] B. Long vd., "Moth-Flame-Optimization-Based Parameter Estimation for FCS-MPC-Controlled Grid-Connected Converter With LCL Filter", *IEEE J. Emerg. Sel. Top. Power Electron.*, c. 10, sy 4, ss. 4102-4114, Ağu. 2022, doi: 10.1109/JESTPE.2022.3140228.
- [25] H. Eskandar, A. Sadollah, A. Bahreininejad, ve M. Hamdi, "Water cycle algorithm – A novel metaheuristic optimization method for solving constrained engineering optimization problems", *Comput. Struct.*, c. 110-111, ss. 151-166, Kas. 2012, doi: 10.1016/j.compstruc.2012.07.010.
- [26] D. Kler, P. Sharma, A. Banerjee, K. P. S. Rana, ve V. Kumar, "PV cell and module efficient parameters estimation using Evaporation Rate based Water Cycle Algorithm", *Swarm Evol. Comput.*, c. 35, ss. 93-110, Ağu. 2017, doi: 10.1016/j.swevo.2017.02.005.
- [27] M. I. Nazir, A. Ahmad, ve I. Hussain, "Water Cycle Algorithm Based Parametric Tuning of Non-Negative LMMN Control of Grid Tied Renewable Energy Systems", *IETE J. Res.*, c. 0, sy 0, ss. 1-17, Haz. 2022, doi: 10.1080/03772063.2022.2089748.
- [28] S. Salcedo-Sanz, J. Del Ser, I. Landa-Torres, S. Gil-López, ve J. A. Portilla-Figueras, "The Coral Reefs Optimization Algorithm: A Novel Metaheuristic for Efficiently Solving Optimization Problems", *Sci. World J.*, c. 2014, s. e739768, Tem. 2014, doi: 10.1155/2014/739768.
- [29] Y. Yang, B. Yang, ve M. Niu, "Parameter identification of Jiles-Atherton model for magnetostrictive actuator using hybrid niching coral reefs optimization algorithm", *Sens. Actuators Phys.*, c. 261, ss. 184-195, Tem. 2017, doi: 10.1016/j.sna.2017.05.009.
- [30] Y. Yang, B. Yang, ve M. Niu, "Adaptive infinite impulse response system identification using

- opposition-based hybrid coral reefs optimization algorithm”, *Appl. Intell.*, c. 48, sy 7, ss. 1689-1706, Tem. 2018, doi: 10.1007/s10489-017-1034-9.
- [31] M. ZALOĞLU, Ş. Fidan, ve E. ERKAN, “Meta-Heuristik Optimizasyon Algoritmalarının Sistem Tanımlama Problemine Uygulanması ve Performans Karşılaştırması”, içinde *International Conference on Engineering, Natural and Social Sciences*, 2023, ss. 510-515.
- [32] R. Celikel ve A. Gundogdu, “System identification-based MPPT algorithm for PV systems under variable atmosphere conditions using current sensorless approach”, *Int. Trans. Electr. Energy Syst.*, c. 30, sy 8, s. e12433, 2020.
- [33] Ş. Fİdan, D. Sevlm, ve E. Erkan, “System Identification and Control of High Voltage Boost Converter”, içinde *2022 Global Energy Conference (GEC)*, Eki. 2022, ss. 25-31. doi: 10.1109/GEC55014.2022.9986621.
- [34] T. Kumbasar, I. Eksin, M. Guzelkaya, ve E. Yesil, “Adaptive fuzzy model based inverse controller design using BB-BC optimization algorithm”, *Expert Syst. Appl.*, c. 38, sy 10, ss. 12356-12364, Eyl. 2011, doi: 10.1016/j.eswa.2011.04.015.