

*Araştırma Makalesi - Research Article*

# Doğadan Esinlenen Optimizasyon Algoritmaları Tabanlı Kesir Dereceli PID Denetleyicilerle Kontrol Edilen Bir Santral Modelinin Performansının İncelemesi

## Investigation of the Performance of a Power Plant Model Controlled by Fractional Order PID Controllers Based on Nature-Inspired Optimization Algorithms

Kunter Sercan Sezer<sup>1</sup>, Nevra Bayhan<sup>2\*</sup>

*Geliş / Received: 26/04/2021*

*Kabul / Accepted: 04/05/2021*

### ÖZ

Bu çalışmada, klasik oransal-integral-türev (PID) kontrolörlerin gelişmiş hali olan kesir dereceli oransal-integral-türev (FOPID) kontrolörlerden faydalanılarak basitleştirilmiş gaz türbinli bir santral modeli için denetlemenin hassas şekilde yapılabilmesi sağlanmıştır. Klasik PID kontrolörler (denetleyiciler) üç parametre içerirken, kesir dereceli PID kontrolörler beş parametre içerir. Parametre sayısının fazla olması daha hassas denetlemenin yapılabilmesine olanak sağlar, ancak bu durum kontrolörün optimizasyonunu zorlaştırır. Kesir dereceli PID kontrolörlerin geleneksel matematiksel yöntemler ile optimizasyonu zor olduğu için; bu çalışmada, doğadan esinlenen (meta-sezgisel) optimizasyon algoritmaları arasında yer alan balina optimizasyon algoritması (BOA), salp sürüsü algoritması (SSA), yapay arı kolonisi (YAK) ve atom arama optimizasyon algoritması (AAO) kullanılmıştır. Bu dört farklı algoritmayla optimize edilen FOPID kontrolör parametreleri, basitleştirilmiş gaz türbini enerji santrali modeline uygulanmış ve sistem çıkış sinyallerinin geçici yanıt performansları karşılaştırılmıştır. Bu amaçla yerleşme süresi ve yüzde en büyük aşım, karşılaştırma kriteri olarak kullanılmıştır. Simülasyon sonuçları, yapay arı kolonisi (YAK) algoritmasıyla optimize edilmiş FOPID kontrolörün, bu santral modeli için yerleşme süresi ve yüzde en büyük aşım kriterleri açısından diğer algoritmalarla optimize edilmiş FOPID kontrolörlere göre daha iyi performans sergilediğini göstermektedir.

**Anahtar Kelimeler-** Gaz Türbinli Güç Santrali, Kesir Dereceli PID Kontrolör, Doğadan Esinlenen Optimizasyon Algoritmaları

### ABSTRACT

In this study, it is ensured that the control of a simplified gas turbine power plant can be performed precisely with the fractional order proportional-integral-derivative (FOPID) controllers, which are the advanced form of conventional proportional-integral-derivative (PID) controllers. While conventional PID controllers contain three parameters, fractional order proportional-integral-derivative PID controllers contain five parameters. The large number of parameters allows more precise control to be made, but this makes difficult optimization of the controller. Since fractional order PID controllers are difficult to optimize with traditional mathematical methods; in this study, whale optimization algorithm (WOA), salp swarm algorithm (SSA), artificial bee colony (ABC) and

<sup>1</sup> İletişim: [kuntersercansezer@gmail.com](mailto:kuntersercansezer@gmail.com) (<https://orcid.org/0000-0001-9428-2595>)

Elektrik-Elektronik Mühendisliği, İstanbul Üniversitesi-Cerrahpaşa, Avcılar Kampüsü, İstanbul, Türkiye

<sup>2\*</sup>Sorumlu yazar iletişimi: [nevra@iuc.edu.tr](mailto:nevra@iuc.edu.tr) (<https://orcid.org/0000-0002-7497-2377>)

Elektrik-Elektronik Mühendisliği, İstanbul Üniversitesi-Cerrahpaşa, Avcılar Kampüsü, İstanbul, Türkiye

atomic search optimization algorithm (ASO), which are among the nature-inspired (metaheuristic) optimization algorithms, are used. FOPID controller parameters which are optimized from four different nature-inspired algorithms are applied to the simplified gas turbine power plant model and the transient responses performances of the system output signals are compared. For this aim, settlement time and percentage maximum overshoot are used as comparison criteria. The simulation results show that the FOPID controller optimized with the artificial bee colony (ABC) algorithm performs better than the rest of FOPID controllers on the settling time and overshoot for this power plant model.

**Keywords- Gas Turbine Power Plant Model, Fractional Order PID, Nature Inspired Optimization Algorithms**

## I. GİRİŞ

Günümüzde gaz türbinli güç santralleri enerji üretiminde büyük yer tutmaktadır. Bu santrallerin dinamiği, devreye alınan ya da çıkarılan yükün etkisi ile ani olarak değişiklik gösterebilmektedir. Santral çıkışındaki yükte oluşabilen bu ani değişim santral jeneratörünün dönüş hızını etkileyerek yavaşlamasına veya hızlanmasına sebep olabilir. PI ve PID gibi geleneksel denetleme metotları, böyle değişimlere hızlı cevap veremeyebilir [1, 2].

Mevcut sistemlerin yükteki değişimlere daha hızlı ve hassas yanıtlar vererek üretim-yük dengesinin sağlanabilmesi için bu çalışmada PID kontrolörlerin genelleştirilmiş hali olan kesir dereceli PID (FOPID) kontrolörler kullanılmıştır. Kesirli dereceli PID denetleyicileri, geleneksel PID denetleyicilerine kıyasla daha karmaşık bir yapıya sahiptir. Geleneksel PID kontrolörleri için, türev ve integral etkileri tamsayı derecelerine sahiptir ve üç katsayı parametresi içerir. Öte yandan, kesirli mertebeden PID denetleyiciler, yapılarında türev ve integral dereceleri tam sayı olması gerekmeyen toplam beş parametreye sahiptir [3, 4]. FOPID kontrolörler, PID kontrolörlere göre daha hassas ayar yapabilmeye olanak sağlarlar ancak bu avantajının yanı sıra parametre sayısının fazla olması optimize edilmelerini zorlaştırmaktadır. Bu problemin aşılabilmesi için literatürdeki doğadan esinlenen optimizasyon algoritmaları kullanılabilir [5].

Doğadan esinlenen optimizasyon algoritmalarının kullanımı ve pek çok sisteme uygulanan bu algoritmaların karşılaştırmaları literatürde gittikçe önem kazanmaktadır [6]. Bu çalışmada, doğadan esinlenen optimizasyon algoritmalarından olan balina optimizasyon algoritmasından (BOA), salp sürüsü optimizasyon algoritmasından (SSA), yapay arı kolonisi (YAK) ve atom arama algoritmasından (AAO) yararlanılmıştır. Farklı hayvan davranışları ve içgüdülerinden esinlenerek Mirjalili ve diğerleri farklı doğa ilhamlı algoritmalar ortaya koymuştur. Bunlar arasında balina optimizasyonu algoritması ve salp sürüsü algoritması önde gelen yöntemlerdendir [7, 8]. 2005 yılında yapılan bir çalışma sonucunda bulunmuş olan yapay arı kolonisi optimizasyon algoritması, doğadan esinlenmiş optimizasyon algoritmaları arasındaki önemli çalışmalardan biridir ve bu algorithmada arı kolonilerinin besin ararken gösterdiği davranışlarından esinlenilmiştir [9]. 2019 yılında jeoloji alanında çok bilinmeyenli denklemlerin çözümü için yüklü parçacıkların potansiyel enerjileri sebebiyle birbirini itme ve çekme kuvvetlerinden esinlenilerek oluşturulan fizik tabanlı olan atom arama algoritması ise Zhao ve diğerleri tarafından geliştirilmiştir [10].

Bu çalışmada, elektrik üretimi için tasarlanmış olan gaz türbinli bir elektrik santralının [1, 2]'deki basitleştirilmiş modeli için son yıllarda yaygın kullanımları ile önde gelen yukarıdaki dört farklı algoritma kullanarak optimize edilmiş FOPID kontrolörler uygulanarak sistemin performans analizi karşılaştırılması yapılmıştır. Bu amaçla ikinci bölümde öncelikle gaz türbinli güç santralının çalışma prensibi açıklanarak bu santralin basitleştirilmiş modeli ve FOPID kontrolör yapıları hakkında teorik bilgiler verilmiştir. Ardından, ele alınan santral modeline uygulanarak karşılaştırması yapılacak olan optimizasyon algoritmaları hakkında teorik bilgiler verilmiştir. Bu algoritmalar sistem modeline uygulanırken kullanılan parametreler açıklanarak, algoritmaların en uygun parametre değerlerini saptaması için ihtiyaç duyulan amaç fonksiyonu (ITAE: zaman ağırlıklı mutlak hata toplamı) hakkında teorik içerik aktarılmıştır.

Bu çalışmanın üçüncü bölümünde basitleştirilmiş gaz türbinli santral modelinin kontrolü için FOPID kontrolör içeren sistemin simülasyonu yapıldı [11], yukarıda bahsi geçen doğa ilhamlı algoritmalar ile optimizasyonu sağlanarak tasarlanmış olan FOPID kontrolörler yardımıyla santral çıkışındaki istenmeyen güç dalgalanmalarına oldukça kısa sürede cevap verilmesi sağlanmıştır. Bahsedilen optimizasyon algoritmaları ile parametre değerleri belirlenmiş FOPID kontrolörün uygulandığı basitleştirilmiş gaz türbinli elektrik güç santral modelinin çıkış işaretleri, yükselme süresi ve yüzde en büyük aşım değeri kriterlerine göre performans

karşılaştırmaları yapılmıştır. Son bölümde ise elde edilen bulgular değerlendirilmiş ve bu doğrultuda çıkarımlar yapılmıştır.

## II. TEORİ VE YÖNTEM

### A. Basitleştirilmiş Gaz Türbin Modeli

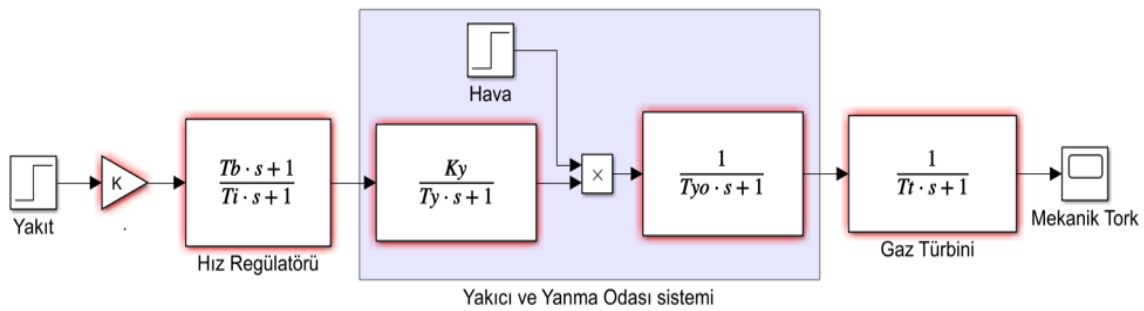
Gaz türbinleri genellikle güç kaynağı olarak kullanılır. Çalışma prensibi basittir. Güç elde etmek için gaz, türbine doğru sıkıştırılır, yakılır ve genişmesi sağlanır. Hacmi artan gaz, türbine yönlendirilir ve türbini çevirir. Bu çalışmada, [1, 2] 'deki gaz türbini modeli referans alınmıştır. Bu model yakıttan mekanik güç üretmek için bir gaz akışı sürecidir. Bu basitleştirilmiş model için aşağıdaki kabuller yapılmıştır:

- Hava akışkan olarak kullanılmıştır ve döngü boyunca ideal gaz olarak kabul edilmiştir.
- Yanma işlemi ihmal edilmiştir ve ısının dışarıdan sabit basınçta sağlandığı varsayılmıştır.
- Kompresör emiş ve türbin egzoz işlemleri ihmal edilmiştir. Sabit basınç altında türbinden çıkan akışkan, kompresöre geri döner.

Temel gaz türbini modeli Şekil 1'de gösterildiği gibi hız regülatör bloğu, yanma ve yanma odası bloğu, türbin bloğu olmak üzere üç ana bölümden oluşmaktadır. Şekil 1'deki basitleştirilmiş gaz türbini modeli blok diyagramında kullanılan parametreler, Tablo 1'de verilmiştir.

Tablo 1. Basitleştirilmiş gaz türbin model parametreleri [1, 2]

Parametre	Değer	Açıklama
K	1	Hız regülatörü kazancı
Tb	0.65	Hız regülatörü birinci gecikme sabiti
Ti	0.1	Hız regülatörü ikinci gecikme sabiti
Ky	1	Yakıcı sistem kazanç sabiti
Ty	0.05	Yakıcı sistemin gecikme sabiti
Tyo	0.4	Yanma odası gecikme sabiti
Tt	0.1	Türbin dinamik özelliklerine bağlı gecikme sabiti



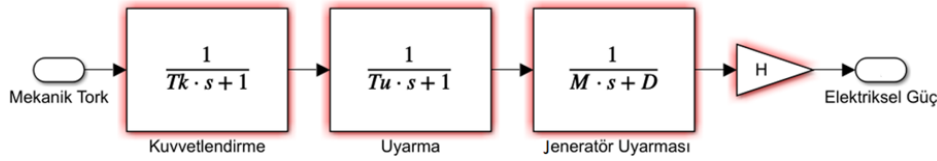
Şekil 1. Basitleştirilmiş gaz türbin modeli blok diyagramı [1, 2]

### B. Basitleştirilmiş Jeneratör Modeli

Jeneratör, mekanik gücü elektrik enerjisine çeviren bir cihazdır. Bu çalışmada gaz türbininin oluşturduğu mekanik güç, elektrik üretiminde kullanılmıştır. Simülasyonlar için kullanılan basitleştirilmiş jeneratör modeli Şekil 2'de verilmiştir [1, 2]. Kuvvetlendirme, uyarma ve jeneratör uyarma bloklarını içerir ve parametreleri Tablo 2'de verilmiştir.

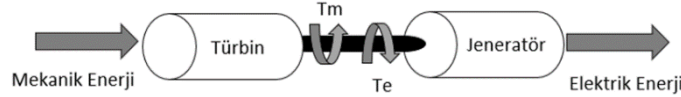
**Tablo 2.** Basitleştirilmiş jeneratör modeli parametreleri

Parametre	Değer	Açıklama
Tk	0.25	Kuvvetlendirme gecikme sabiti
Tu	1	Uyarma gecikme sabiti
M	0.1	Jeneratör gecikme sabiti
D	1	Sönüm sabit değeri
H	1	Hız regülatörü sabit değeri (Hz/MW)



**Şekil 2.** Basitleştirilmiş jeneratör modeli blok diyagramı [1, 2]

Türbin ve jeneratör arasındaki mekanik güç transferi ilişkisi Şekil 3'te modellenmiştir. Burada,  $T_m$  gaz türbini tarafından üretilen mekanik tork ve  $T_e$  ise yüke bağlı olarak değişkenlik gösteren elektriksel torktur.



**Şekil 3.** Görselleştirilmiş türbin-jeneratör ilişkisi [1, 2]

Burada  $T_m$ : mekanik tork ve  $T_e$ : elektriksel torktur. Türbin sistemi ile yakılan yakıttan elde edilen enerji, hızlandırma etkisine sahiptir ve  $T_m$  artar. Jeneratör tarafından beslenen şebekede tüketimdeki artış  $T_e$  'nin artmasına neden olur. Bu durumda,  $T_e > T_m$  olduğunda, merkezi şaft yavaşlama eğilimindedir ve frekans azalır. Frekansı sabit tutmak için  $T_m$ 'yi artırmak gerekir. Bunun için yakıt tüketimini yükseltmek ve  $T_m$  ile  $T_e$ 'yi eşitleyerek dengelemek gereklidir.  $T_m < T_e$  olduğu durumlarda yakıt tüketimini azaltarak  $T_m$ 'yi düşürmek ve frekansı korumak için  $T_e$  ile eşitleyerek kuvvetin dengesini sağlamak gereklidir.

Santralin işletilmesi sırasında bu işlemler defalarca tekrarlanır. Santralin kontrol altında tutulması ve üretilen elektriğin sabit frekansta tutulması için bu kontrol süreci gereklidir. Bu noktada jeneratör açılma hızı ile yükteki güç değişimi arasındaki ilişkiyi ifade eden denklemler, aşağıdaki gibidir:

$$T_T = I a \quad (1)$$

$$M = I \omega \quad (2)$$

$$P_T = \omega T_T = \omega I a = M a \quad (3)$$

$T_T$  toplam tork olduğunda,  $I$  faz başına akım,  $\omega$  açısal hız,  $a$  alan akısı ve  $M$  momentumdur.  $\omega_0$ , jeneratör milinin başlangıçtaki dönüş frekansdır ve  $\Delta\omega$ , frekans değişimi olarak adlandırılır.  $P_T$  toplam güç,  $P_m$  gaz türbini tarafından üretilen mekanik güç ve  $P_e$  ise yük değişimiyle ilişkili elektriksel güçtür. Başlangıç güç durumları ile rotor frekansı arasındaki ilişkiyi gösteren denklemler de aşağıda verilmiştir.

$$\omega = \omega_0 + \Delta\omega = \omega_0 + a t \quad (4)$$

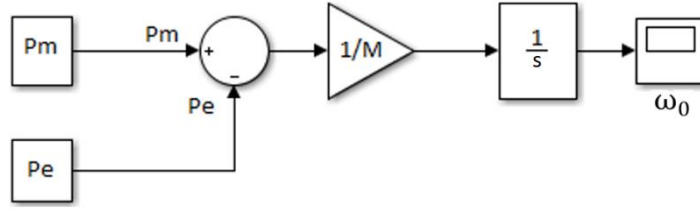
$$T_T = I a = I \frac{d}{dt} \Delta\omega \quad (5)$$

$$P_T = P_m - P_e \quad (6)$$

$$P_T = \omega_0 I \frac{d\Delta\omega_0}{dt} = M \frac{d\Delta\omega_0}{dt} \quad (7)$$

$$\omega_0 = \frac{1}{M_S} P_T = \frac{1}{M_S} P_m - P_e \quad (8)$$

Denklem (8) yardımıyla Şekil 4'teki blok diyagramı, oluşturulur.



Şekil 4. Rotor hızı ile mekanik ve elektriksel güç arasındaki ilişkisi gösteren blok diyagramı[1, 2]

Yük değişimi ve frekansa etkisi de denklem (9) 'da ifade edilmiştir.

$$\Delta P_e = \Delta P_L + \Delta P_D \rightarrow \Delta P_L = D \Delta \omega \rightarrow D = \frac{\Delta \omega}{\Delta P_L} \quad (9)$$

Burada, D, sönümlenme sabitiyken  $\Delta P_L$  ise yükteki güç değişimidir. D, yük değişimi için yaklaşık %1 olan frekanstaki değişim yüzdesidir. Örneğin, %1'lik bir yük değişiminde %1'lik bir frekans değişikliği olduğunda, sönümlenme oranı denklem (9)'dan elde edilir. Sönümlenme faktörü  $D = 1/1 = 1$ 'dir. Yük sönümlenme modeli, Şekil 5'de görülmektedir.



Şekil 5. Jeneratör uyarılma diyagramı [1, 2]

Otomatik voltaj regülatör takviyesi ve söndürme blokları eklendikten sonra, jeneratör modeli Şekil 2'deki gibi olur [1, 2]. Denklem (8) ve (9)'a göre, bu çalışmada analiz edilen kontrolör frekans değişimini engellemeli ve sabit tutmalıdır.

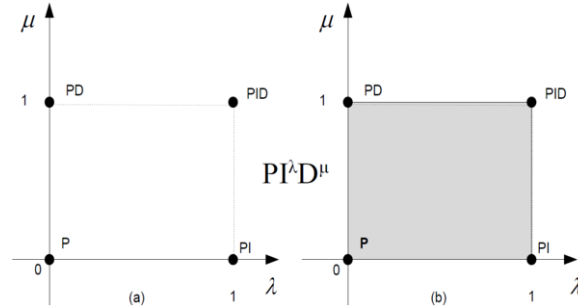
### C. Kesir Dereceli PID Kontrolör (FOPID Kontrolör)

Kesir dereceli PID kontrolör, klasik PID kontrolörlerin genelleştirilmiş halidir [3-5]. Klasik PID kontrolörlerdeki gibi kesir dereceli PID kontrolörler de oransal katsayı, türev ve integral etkilerinden oluşurlar. Ancak klasik PID kontrolörlerde türev ve integral etki mertebesi tamsayı iken; kesir dereceli PID kontrolörlerde türev ve integral etki mertebeleri kesir sayılı olabilir. FOPID'nin genelleştirilmiş transfer fonksiyonu denklem (10) 'da verilmiştir.

$$G(s) = K_p + K_I \frac{1}{s^\lambda} + K_D s^\mu, \quad (\mu, \lambda > 0) \quad (10)$$

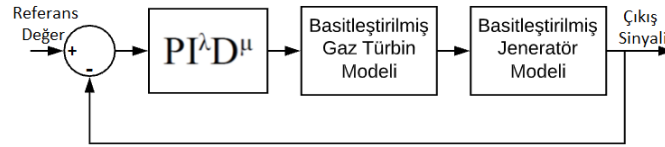
Burada,  $K_p$  oransal kazanç,  $K_I$  integral kazanç,  $K_D$  türevsel kazançtır ve  $\lambda$  integral terimin kesirli mertebesidir,  $\mu$  ise türev terimin kesirli mertebesidir. Burada  $\lambda$  ve  $\mu$ , 1'e eşitse, kontrolör bir PID tipi olacaktır.  $\lambda = 1$  ve  $\mu = 0$  ise, bir PI kontrolörü olacaktır.  $\lambda$  ve  $\mu$ , her ikisi de 0'a eşitse, P kontrolör elde edilir. Tüm bu denetleyici türleri, FOPID kontrolörün özel durumlarıdır. Şekil 6'daki mertebeler haritaları, FOPID'nin daha geniş

çözüm kümesine sahip olduğunu ve geleneksel PID'den daha hassas kontrol sağlayabileceğini göstermektedir [5, 12].



Şekil 6. (a) Klasik PID kontrolör türev-integral kuvvet haritası, (b) Kesir dereceli PID kontrolör türev-integral kuvvet haritası [5]

Şekil 7, FOPID ( $PI^\lambda D^\mu$ ) denetleyicisini içeren basitleştirilmiş santral modelinin kapalı döngü sisteminin genel yapısını göstermektedir. FOPID denetleyicinin geleneksel PID denetleyicisine göre iki ekstra parametresi vardır. Bu, daha iyi bir denetleyici performansı sağlayabilir ve genel sistemin performansını da olumlu yönde etkilemesi beklenir.



Şekil 7. FOPID kontrolör sistem modeli

Yüksek sayıda parametre, FOPID denetleyici optimizasyonunu zorlaştırır. FOPID denetleyicisini zorlu matematiksel yöntemlerle optimize etmek kolay olmadığından; doğadan esinlenen optimizasyon algoritmaları bu konuda kolaylık sağlar.

#### D. Doğadan Esinlenen Optimizasyon Algoritmaları

Doğadan esinlenen algoritmalar, çözülmesi zor olan optimizasyon problemleri için kabul edilebilir sürede optimum çözüme yakın değerler verebilen alternatif yöntemlerdir. Amaçları, arama uzayını özel yöntemlerle tarayarak istenilen değerleri bulmaktır. Doğadan esinlenen optimizasyon algoritmaları, optimuma en yakın değerleri belirlemek için amaç fonksiyonuna ihtiyaç duyarlar. Literatürde çeşitli amaç fonksiyonları yaygın olarak kullanılmaktadır [5]. Bunlardan en popüler olanları hatanın mutlak değerinin integrali (IAE), hatanın karesinin integrali (ISE), zaman ağırlıklı hatanın karesinin integrali (ITSE) ve hatanın mutlak değerinin integrali (ITAE)'dir. Bu çalışmada parametrelerin uygunluğunu belirlemek için denklem (11)'de verilen ve performans ölçütü değerlerinin minimizasyonunu amaçlayan ITAE kullanılmıştır, çünkü en kısa yerleşme süresine ve yüzde aşım miktarına sahip olunması amaçlanmıştır.

$$ITAE = \int_0^T t |e| dt \quad (11)$$

Bu çalışmada kullanılan BOA, SSA, YAK ve AAO doğadan esinlenen optimizasyon algoritmaları ele alınan basitleştirilmiş gaz türbinli santral modeline uygulanırken ITAE fonksiyonu ile parametrelerin uygunluk değerleri saptanmıştır. Bu dört algoritmayı ana hatlarıyla inceleyecek olursak, sırasıyla aşağıdaki gibidirler:

1) *Balina Optimizasyonu Algoritması (BOA)*: Kambur balinaların avlanmaları sırasında sergiledikleri davranışlar incelenerek bilgisayar ortamında matematiksel olarak benzetimi yapılmış ve balinanın avını yakalamış gibi bilgisayardaki bir dizi işlem sonucunda amaçlanan çözüme yakın değerlere ulaşılmasını sağlayan balina optimizasyonu algoritması geliştirilmiştir [7]. Kambur balinalar küçük balıklarla beslenirler ve genellikle bu küçük balıkları toplu şekilde avlarlar. Avlanırken kendilerine özgü "kabarcık ağı" diye adlandırılan özel bir davranış sergileyerek küçük balık sürüsüne gidip daralan bir çember çizerek etraflarında kabarcıklar çıkartmak

yolu ile bir araya toparlanmalarını sağlar ve sonrasında küçük balık sürüsüne doğrudan saldırırlar. Bu yöntem ile daha az çaba harcayarak avlanma sürecinden daha verimli sonuçlar elde ederler. Bunun yanı sıra zaman zaman direkt olarak küçük balık sürülerine yöneldikleri de bilinmektedir. Bu balinaların avlanma davranışlarının matematiksel benzetimi denklem (12) ve (13)'de verilmiştir [5, 7].

$$D = |C X_{best} - X_t| \quad (12)$$

$$X_{t+1} = \begin{cases} X_{best} - A D, & A < 1 \\ X_{rand} - A D, & A \geq 1 \end{cases}, \quad p < 0.5 \text{ ise } C = 2r \\ D e^{bl} \cos(2\pi l) + X_{best}, \quad p \geq 0.5 \text{ ise } C = 1 \quad (13)$$

Denklem (12)'de  $D$  besin ile balina konumu arasındaki doğrusal uzaklığı;  $X_{best}$  uygunluk değeri en iyi besin konumunu;  $X_t$  balina konumunu temsil etmektedir.  $C$ , denklem (14)'de verilmiştir. Burada  $r_1$  ise  $[0,1]$  aralığında rastgele bir sayıdır.

$$C = 2 r_1 \quad (14)$$

Denklem (13)'de  $X_{t+1}$  balinanın yeni konumunu,  $b$  spiral şeklin sağlanması için gerekli katsayı sabitini;  $l$   $[-1,1]$  aralığında rastgele bir sayıyı,  $p$  ise  $[0,1]$  aralığında rastgele bir sayı olan besinin olasılık değerini ifade etmektedir. Burada  $A$ , denklem (15)'de verilen şekilde hesaplanır. (15)'de  $r_2$ ,  $[0,1]$  aralığında rastgele bir sayı,  $a$   $[-1,1]$  aralığında iterasyon sayısı ile doğru orantılı olarak azalan bir parametredir

$$A = 2 a r_2 - a \quad (15)$$

Algoritma başlangıcında balina sayısı belirlenir ve arama uzayında rastgele dolaşmaları için rastgele konumlara atanır. Ardından buldukları konumların amaç fonksiyonu yardımı ile uygunluk değerleri hesaplanır ve bu değere bağlı olarak denklem (13) kullanılarak konumları güncellenir ve bu süreç iterasyon sayısı kadar tekrarlanarak uygunluk değeri en iyi olan konum algoritma çıktısı olarak atanır.

2) *Salp Sürüsü Algoritması (SSA)*: Salpler yapıları denizanasına benzeyen renksiz deniz canlılarıdır. Hareket ederken vücut şekillerinden faydalanarak suyu pompalarlar. Sürü halinde yaşayan bu canlı türü avlanırken birbirine tutunarak zincir oluşturur. Bu şekilde hareket kabiliyetlerini arttırarak daha geniş bir alanı tarayarak besine ulaşabilirler. Bu avlanma davranışından faydalanılarak salp sürüsü algoritması oluşturulmuştur [8]. Algoritma başlangıcında sürü bireyleri arama uzayına rastgele olarak dağıtılır ve amaç fonksiyonu yardımı ile uygunluk değerleri hesaplanır. Uygunluk değerleri en iyi olan konum lider salpi temsil eder. Geriye kalanlar sürü bireyidir. Salp zinciri besin arama süreci boyunca yer değiştirirken yönlendirmeyi lider salp yapar ve diğer salpler zinciri bozmayacak şekilde onu takip eder. Bu süreç algoritma sırasında iterasyon sayısı kadar tekrarlanır. Lider salpin yer değiştirme hareketi matematiksel olarak modellenmiş ve denklem (16)'da verilmiştir.

$$X^{1(lider)} = \begin{cases} X_{besin} + c_1 ((ub - lb)c_2 + lb), & c_3 \geq 0 \\ X_{besin} - c_1 ((ub - lb)c_2 + lb), & c_3 < 0 \end{cases} \quad (16)$$

Denklem (16)'da  $X^l$  lider salpin konumunu,  $X_{besin}$  besin konumunu,  $C_2$  ve  $C_3$   $[0,1]$  aralığında rastgele sayıları belirtir. Burada  $c_1 = 2e^{-(4l/L)}$  ile hesaplanır. Diğer salplerin hareketleri ise denklem (17) kullanılarak gerçekleştirilecektir  $X_{yeni}$  salplerin yeni konumlarını temsil etmektedir.

$$X_{yeni}^i = \frac{X^i + X^{i-1}}{2} \quad (17)$$

İterasyon sayısı kadar tekrarlanan süreç sonunda uygunluk değeri en iyi olan konum, algoritma çıktısı olarak atanır.

3) *Yapay Arı Kolonisi (YAK)*: Yapay arı kolonisi algoritması, arı topluluklarının besin toplama esnasındaki davranışlarından esinlenilerek oluşturulmuş matematiksel benzetimin optimizasyon problemlerinin çözümünde kullanıldığı işlem dizisidir [9]. Arı kolonilerinde bireyler arasında hiyerarşik yapı vardır. Üç tip arı mevcuttur. Bunlar: kraliçe arı, erkek arı ve işçi arılardır. Kraliçe arı ve erkek arıların görevleri yönetimde yer alarak

kolonideki arı popülasyonunun devamlılığını sağlamaktır. İşçi arılar ise görevlerine göre üçe ayrılır. Bunlar besin kaynağından kovana besin taşıyan görevli işçi arılar, besin konumundan kovana ulaşan görevli işçi arılardan besin hakkında bilgi alan gözcü işçi arılar ve yeni besin kaynakları keşfetmekten sorumlu olan izci işçi arılardır. Algoritma başlangıcında işçi arı sayısı belirlendikten sonra arama uzayında konumları rastgele olarak denklem (18) kullanılarak atanır.

$$X_0^i = X_{\min} + \text{rand}() (X_{\max} - X_{\min}), \quad i \in [1, \dots, n] \text{ ve } n > 1 \quad (18)$$

Burada  $X_0^i$ , arama uzayı içerisindeki n adet parçacık (besin) konumunu ifade eder.  $X_{\min}$  arama uzayının alt sınırlarını ve  $X_{\max}$  arama uzayının üst sınırlarını ifade eder.  $\text{rand}()$  fonksiyonu ise  $[0,1]$  aralığında eşit olasılıklara sahip rastgele sayıları ifade eder. Belirlenmiş olan besin konumlarının uygunluk (fitness) değerleri hesaplanır. Ardından amaç fonksiyonu kullanılarak uygunluk (fitness) değerleri hesaplanır. Besinlerin uygunluk değerleri göz önünde bulundurularak gözcü arılar, görevli işçi arıları denklem (19) kullanılarak besin ve kovan arasında yönlendirir.

$$X_{t+1}^i = X_t^i + \Phi_i * (X_t^i - X_t^k) \quad (19)$$

$X_{t+1}^i$ , besinin komşu konumunu ve  $X_t^i$  değişkeninin yeni değerini sembolize etmektedir.  $X_t^i$  mevcut besin konumu;  $X_t^k$  bilinen besin kaynaklarından rastgele birinin konumudur.  $\Phi_i$  ise  $[-1,1]$  aralığında rastgele değer alan ağırlık katsayısıdır. Bir besine algoritma başında belirlenmiş olan limit değeri (L) kadar görevli arı yollandığında ve bu süreçte uygunluk değeri mevcut en iyi değerden daha iyi bir değere ulaşamazsa bu besin kaynağı tükenmiş olarak kabul edilir ve görevli işçi arı, izci işçi arı olarak yeni besin kaynağı arayacak şekilde arama uzayında rastgele konumlara yollanır. Bu süreç algoritma başlangıcında belirlenmiş olan iterasyon sayısı kadar tekrarlanır ve sonuç olarak ulaşılabilen en iyi uygunluk değerine sahip konum, algoritma çözümü olarak atanır.

4) *Atom Arama Algoritması (AAO)*: Atom arama algoritması, fizik kurallarından esinlenilerek atomlar arasındaki etkileşimin matematiksel olarak benzetimi ile oluşturulmuş ve optimizasyon problemlerine çözüm olarak sunulan bir işlem dizisidir [10]. Maddenin yapısı gereği atomlar arasında etkileşim mevcuttur ve itme ile çekme kuvvetleri mevcuttur. Bu kuvvetlerden atomlar arası potansiyel enerji atomlar arası uzaklık cinsinden ifade edilmek istendiğinde literatürde yaygın olarak Lennard-Jones potansiyel denklemi kullanılmaktadır. Buna ek olarak atomlar arasında kovalanet bağ kaynaklı ikinci bir kuvvet daha vardır. Denklem (20)'de uygulanan net kuvvet  $F_{net}$  verilmiştir.

$$F_{net} = F_i + G_i \quad (20)$$

$F_i$  atomlar arası potansiyel enerji kuvveti;  $G_i$  atomlar arası kovalent bağ etkileşim kuvvetini temsil etmektedir. Burada  $F_i$  ve  $G_i$  sırasıyla denklem (21-24) kullanılarak hesaplanır.

$$F_{ij} = -\eta(t) \left[ 2(h_{ij}(t))^{-13} - (h_{ij}(t))^{-7} \right] \quad (21)$$

$$\eta(t) = a \left( 1 - \frac{t-1}{1} \right)^3 e^{-\frac{20t}{T}} \quad (22)$$

$$G_i(t) = \lambda(t) (X_{best}^d(t) - X_i^d(t)) \quad (23)$$

$$\lambda(t) = \beta e^{-\frac{20t}{T}} \quad (24)$$

Denklem (21)'deki  $\eta(t)$  derinlik parametresi denklem (22) ile hesaplanır.  $h_{ij}(t)$  parametresi atom türünün karakteristik özellikleri ve konumlarına bağlı olarak hesaplanabilen bir değişkendir.  $a$  derinlik ağırlığı,  $t$ : o an gerçekleştirilen deneme (iterasyon) numarası,  $T$  toplamda gerçekleştirilecek olan deneme (iterasyon) sayısıdır. Denklem (23)'teki  $X_{best}^d(t)$  uygunluk değeri en iyi olan atom konumunu;  $X_i(t)$  o an konum güncellemesi için kuvvet hesabı yapılan atomun konumunu;  $\lambda(t)$  Lagrangian çarpanını;  $\beta$  ise Lagrangian çarpan ağırlığıdır. Bir



atoma etkiyen net kuvvetin hesaplanmasından sonra Newton yasalarından faydalanılarak  $m_i$ 'nin atom kütlelerini temsil eden eşitlik (25) ile ise atomun ivmesi bulunur.

$$a_i = \frac{F_i + G_i}{m_i} \quad (25)$$

Denklem (25)'de denklem (21) ve (23) yerine konulduğunda denklem (26) elde edilir.

$$\begin{aligned} a_i(t) &= \frac{F_i(t)}{m_i(t)} + \frac{G_i(t)}{m_i(t)} \\ &= -\left(1 - \frac{t-1}{1}\right)^3 e^{-\frac{20t}{T}} \sum_{j \in K_{best}} \frac{rand_j \left[ 2 \left( h_{ij}(t) \right)^{-13} - \left( h_{ij}(t) \right)^{-7} \right]}{m_i(t)} \\ &\quad + \beta e^{-\frac{20t}{T}} \frac{X_{best}^d(t) - X_i^d(t)}{m_i(t)} \end{aligned} \quad (26)$$

Burada  $m_i(t)$  denklem (27) ve (28) kullanılarak bulunur.

$$M_i(t) = e^{-\frac{Fit_i(t) - Fit_{best}(t)}{Fit_{worst}(t) - Fit_{best}(t)}} \quad (27)$$

$$m_i(t) = \frac{M_i(t)}{\sum_{j=1}^N M_j(t)} \quad (28)$$

Burada  $Fit_i$  ivmesi hesaplanan atomun amaç fonksiyonuna sokulması ile elde edilen uygunluk değeridir (fitness).  $Fit_{best}$  mevcut atom konumları içerisinde en iyi (min) uygunluk değeri;  $Fit_{worst}$  ise mevcut atom konumları arasında en kötü (max) uygunluk değeridir.  $N$  ise toplam atom sayısını temsil eder. Atom ivme değeri bulunduktan sonra denklem (29) ve (30) ile anlık hız  $V(t+1)$  ve bir sonraki konum  $X(t+1)$  belirlenir.

$$V_i(t+1) = rand v_i(t) + a_i(t) \quad (29)$$

$$X_i(t+1) = X_i(t) + v_i(t+1) \quad (30)$$

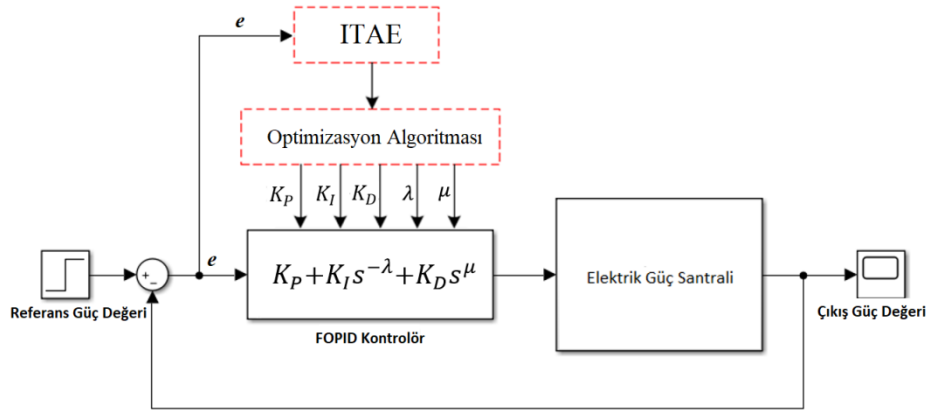
Bu matematiksel modeller kullanılarak bir atoma etkiyen kuvvet ve algoritma işleyişindeki bir sonraki konumu saptanır. Bu işlem algoritma başlangıcında belirlenmiş olan atom sayısı ve bu atomlar arasında iterasyon sayısı göz önünde bulundurularak tekrarlanır. İşlem dizisi sonucunda uygunluk değeri en iyi olan konum, algoritma çıktısı olarak atanır.

### III. BULGULAR

Denetleyici ve optimizasyon algoritmalarının sistem modeline nasıl uygulandığı Şekil 8'de görülmektedir. Şekil 8'de  $e$ , hatayı temsil etmektedir ve sistem çıkış değeri ile referans değer arasındaki farka eşittir. Optimizasyon algoritmalarının sistem modeline uygulanması sırasında,  $e$  (hata) sinyalini sıfıra yakın minimum seviyede tutmak ve sistem çıkış sinyalinin kısa bir yerleşme süresinin sağlanması için ITAE hesaplama yöntemi kullanılmıştır. Algoritmaların çalışması sırasında, her bir ajan için FOPID denetleyicisinin başlangıç parametreleri, kullanılan algoritma prensiplerine uygun şekilde belirlenir. Daha sonra parametreler sistem modelinde yazılarak ITAE hesaplanır ve uygunluk değerleri bulunur. Bu süreç tüm arama bireyleri (parçacıkları) için geçerlidir (nüfus kadar). Algoritma çalıştırılarak uygunluk değerleri belirlenen popülasyon bireylerinin pozisyonları güncellenir. Bu süreç, yineleme sayısı boyunca tekrar tekrar devam eder ve sistem modeli üzerinde farklı parametre grupları test edilir. Yinelemeler sonucunda en düşük ITAE değerine sahip ajan (parametre grubu), yani en düşük uygunluğa sahip parametre grubu ( $K_p, K_I, K_D, \lambda, \mu$ ) algoritmanın çıktısı olarak belirlenir.

Burada, amaç fonksiyonu sonucunda en düşük değere sahip parametre değerlerini bulmak, algoritmaların

temel amacıdır. Şekil 8'de, FOPID denetleyicisinin ve parametrelerinin ( $K_p$ ,  $K_I$ ,  $K_D$ ,  $\lambda$ ,  $\mu$ ) basitleştirilmiş gaz türbini enerji santrali modeline uygulanması, optimizasyon algoritmasının sisteme uygulanması ve amaç fonksiyonunun (ITAE) hata sinyali ( $e$ ) ile hesaplanması görselleştirilerek verilmiştir. Bu çalışmada simülasyon referans güç değeri olarak 50 saniye +1, 50 saniye -1 uygulanmıştır. Böylelikle sistemin hem pozitif hem de negatif alanlardaki cevapları ölçülerek +1'den -1'e geçerken sistemin ani değişime verdiği tepkiyi gözlemlemek mümkün olmuştur. Simülasyonlarda, arama uzayı sınırları  $K_p$ ,  $K_I$  ve  $K_D$  kazanç parametreleri için 0.001'den 20'ye ve  $\lambda$ ,  $\mu$  parametreleri için ise 0.001'den 2'ye kadar kabul edildi.



Şekil 8. Kesir dereceli PID kontrolör ile kontrol edilen basitleştirilmiş güç santral modeli [5]

BOA algoritması ile yapılan simülasyonlarda Tablo 3'de verilmiş olan parametre değerleri kullanılmıştır.

Tablo 3. BOA için simülasyon parametreleri

Parametreler	Değer
Parçacık sayısı	50
İterasyon sayısı	30
$K_p$ , $K_I$ , $K_D$ , $\lambda$ , $\mu$ (alt sınır)	[0.001, 0.001, 0.001 0,001 0,001]
$K_p$ , $K_I$ , $K_D$ , $\lambda$ , $\mu$ (üst sınır)	[20, 20, 20, 2, 2]
b	1

SSA algoritması ile yapılan simülasyonlarda Tablo 4'de verilmiş olan parametre değerleri kullanılmıştır.

Tablo 4. SSA için simülasyon parametreleri

Parametreler	Değer
Parçacık sayısı	50
İterasyon sayısı	30
$K_p$ , $K_I$ , $K_D$ , $\lambda$ , $\mu$ (alt sınır)	[0.001, 0.001, 0.001 0,001 0,001]
$K_p$ , $K_I$ , $K_D$ , $\lambda$ , $\mu$ (üst sınır)	[20, 20, 20, 2, 2]

YAK algoritması ile yapılan simülasyonlarda Tablo 5’de verilmiş olan parametre değerleri kullanılmıştır.

**Tablo 5.** YAK için simülasyon parametreleri

Parametreler	Değer
Parçacık sayısı	50
İterasyon sayısı	30
$K_p, K_I, K_D, \lambda, \mu$ (alt sınır)	[0.001, 0.001, 0.001 0,001 0,001]
$K_p, K_I, K_D, \lambda, \mu$ (üst sınır)	[20, 20, 20, 2, 2]
Limit (L)	80

AAO algoritması ile yapılan simülasyonlarda Tablo 6’da verilmiş olan parametre değerleri kullanılmıştır.

**Tablo 6.** AAO için simülasyon parametreleri

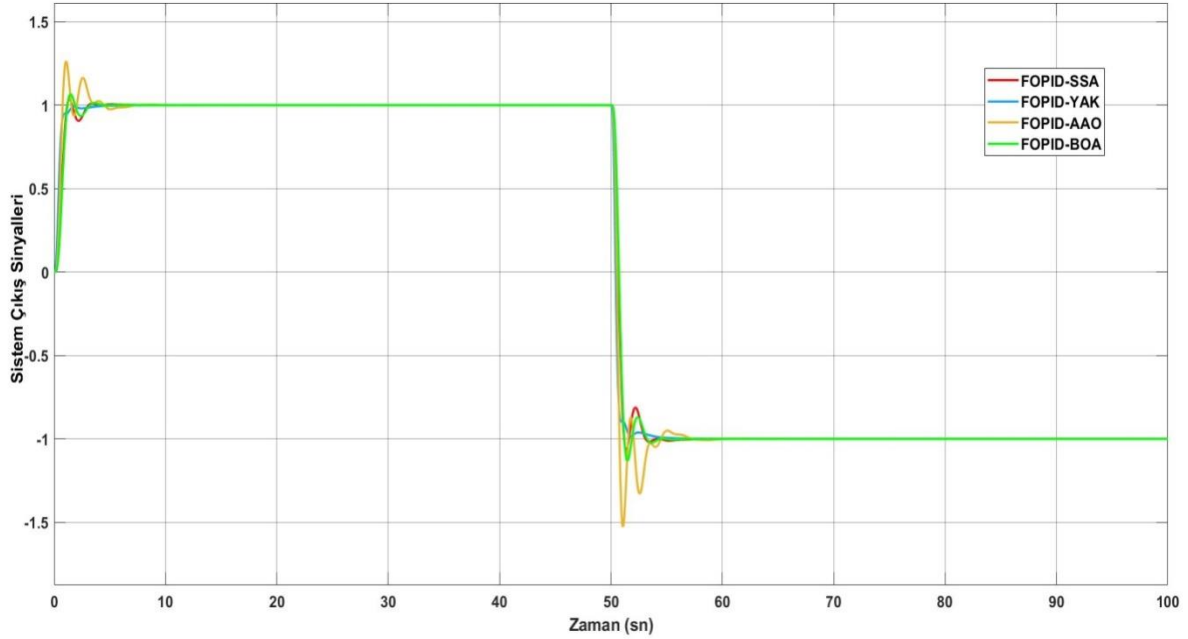
Parametreler	Değer
Parçacık sayısı	50
İterasyon sayısı	30
$K_p, K_I, K_D, \lambda, \mu$ (alt sınır)	[0.001, 0.001, 0.001 0,001 0,001]
$K_p, K_I, K_D, \lambda, \mu$ (üst sınır)	[20, 20, 20, 2, 2]
$\alpha$ (Derinlik ağırlığı)	50
$\beta$ (Lagrangian çarpanı ağırlığı)	0.2

Tablo 3-Tablo 6’da verilen başlangıç değerleri ile gaz türbini santral modeline bu çalışmada ele alınan dört doğa ilhamlı optimizasyon algoritması uygulanarak Tablo7’de verilen FOPID kontrolör parametreleri bulunmuştur.

**Tablo 7.** FOPID kontrolör parametre değerleri

BOA-FOPID	SSA-FOPID	YAK-FOPID	AAO-FOPID
$K_p = 0.7225$	$K_p = 1.2776$	$K_p = 2.74$	$K_p = 1.3061$
$K_I = 1.3633$	$K_I = 1.3483$	$K_I = 1.9575$	$K_I = 3.4435$
$K_D = 0.7399$	$K_D = 0.4471$	$K_D = 0.5182$	$K_D = 1.1787$
$\lambda = 1.0144$	$\lambda = 1.1975$	$\lambda = 1.0110$	$\lambda = 1.4377$
$\mu = 0.3308$	$\mu = 0.5463$	$\mu = 1.3100$	$\mu = 0.5512$

Tablo 7’deki parametre değerleri kullanılarak gaz türbini santral modeline FOPID kontrolör uygulandığında Şekil 9’da gösterilen çıkış sinyalleri elde edilir.

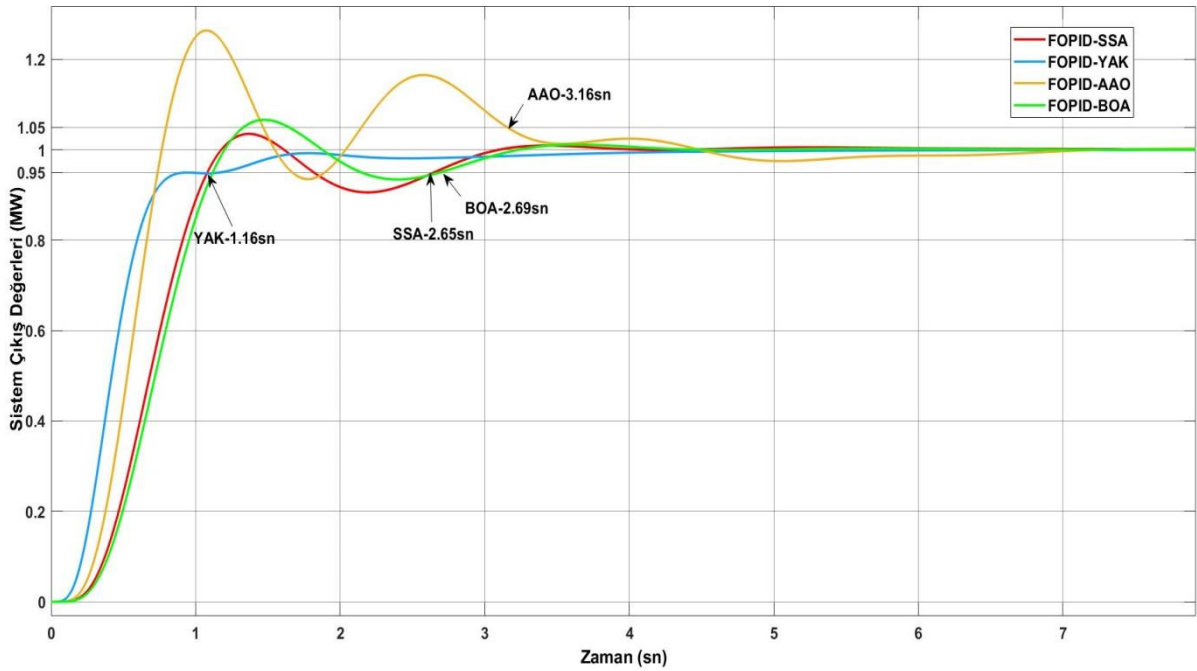


Şekil 9. Sistem çıkış sinyalleri genel görünümü

Optimize edilmiş FOPID denetleyicilerinin net bir performans karşılaştırmasını yapmak için yerleşme süresi ve en büyük yüzde aşım kriterleri kullanılmıştır. Bu kriterler değerlendirilirken referans değerinin hem pozitif (0-50 sn arası +1) hem de negatif (50-100 sn arası -1) olarak uygulanması, sistem davranışının farklı koşullar altında gözlemlenmesini sağlamıştır.

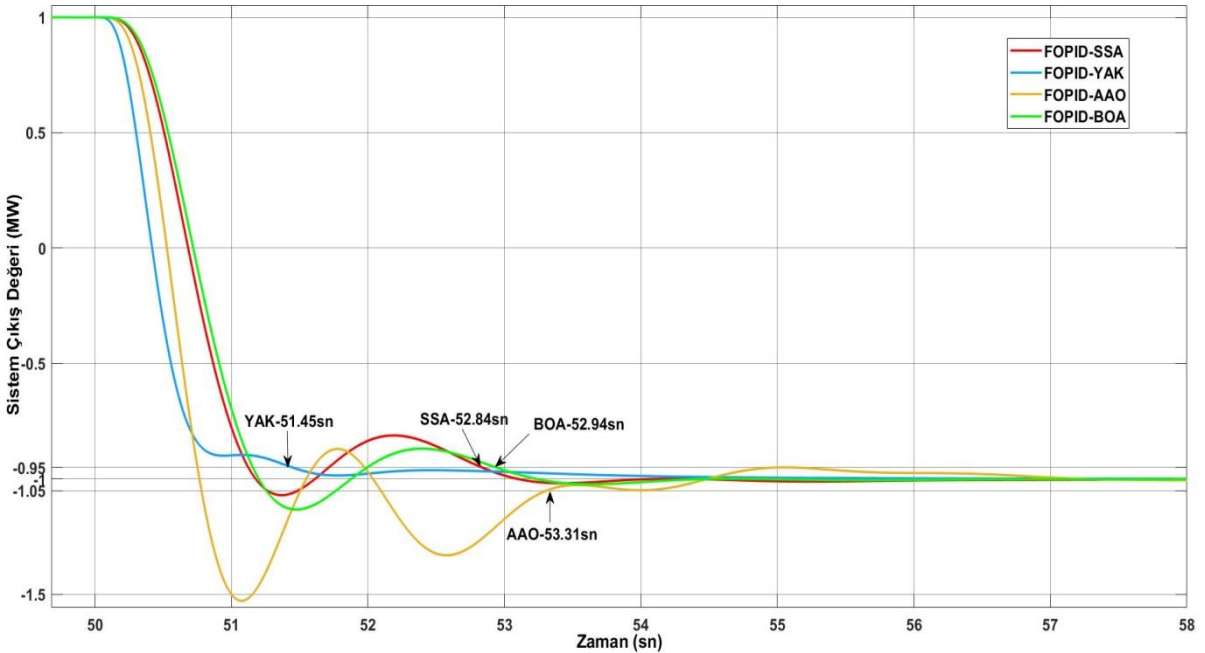
Yerleşme süresi, literatürde en çok kullanılan yaklaşım kriteri olan % 5'lik tolerans bandı esas alınarak uygulanmıştır ve çıkış sinyalinin ilk kez % 5'lik bant aralığına girdiği ve tekrar çıkmadığı süredir. Karşılaştırmaları daha net bir şekilde yapabilmek için, pozitif bölgedeki (0-50 sn arası) çıkış sinyallerinin geçici rejimi Şekil 10'da verilmiştir. Şekil 10'da BOA, SSA, YAK ve AAO algoritmaları ile optimize edilmiş FOPID kontrolörler uygulanmış gaz türbin santral modeli sisteminin çıkış sinyallerinin üzerinde gerekli yerlerde yerleşme süreleri, oklarla gösterilmiştir. Pozitif bölgede %5'lik bant aralığı değerlendirildiğinde Şekil 10'da verilmiş olan çıkış işaretlerinin yerleşme süreleri incelendiğinde; YAK-FOPID'nin uygulandığı sistem çıkışının en küçük yerleşme süresine sahip olduğu görülmüştür. Geriye kalan üç optimizasyon algoritmasının uygulandığı sistem çıkış eğrilerine bakıldığında ise, yerleşme sürelerinin en küçükten en büyüğe doğru sırayla SSA-FOPID, BOA-FOPID, AAO-FOPID için olduğu görülür.

Yüzde en büyük aşım ise sistem yanıtının en büyük değeri ile sabit durum değeri arasındaki farkın sabit durum değerine oranının yüzde ifadesidir. Şekil 10'da görülen sistem çıkış işaretleri için pozitif bölgede yüzde en büyük aşım değerlendirildiğinde; YAK-FOPID'nin uygulandığı sistem çıkışının en küçük aşımına sahip olduğu görülür. Geriye kalan üç optimizasyon algoritmasının uygulandığı sistem çıkış eğrilerine bakıldığında ise, aşım miktarlarının en küçükten en büyüğe doğru sırayla SSA-FOPID, BOA-FOPID, AAO-FOPID için olduğu görülür.



Şekil 10. Pozitif bölgede yaklaşılandırılmış sistem çıkış sinyalleri

Negatif bölgedeki (50-100 sn arası) çıkış sinyallerinin geçici rejimi Şekil 11'de verilmiştir. Negatif bölgedeki yerleşme zamanlarının (oklarla gösterilen) ve en büyük aşımın sıralamaları, pozitif bölgedekiyle aynıdır.



Şekil 11. Negatif bölgede yaklaşılandırılmış sistem çıkış sinyalleri

Pozitif ve negatif bölgeler için yerleşme süreleri ve yüzde aşım değerleri Tablo 8'de verilmiştir.

**Tablo 8.** Sistem cevabının geçici rejim performans kriter değerleri.

Algoritma-Kontrolör	Pozitif Bölge		Negatif Bölge	
	Yerleşme süresi	Aşım	Yerleşme süresi	Aşım
BOA-FOPID	2.69 sn	% 6,65	2.94 sn	% 13,0
SSA-FOPID	2.65 sn	% 3,51	2.84 sn	% 6,82
YAK-FOPID	1.16 sn	% 0	1.45 sn	% 0
AAO-FOPID	3.16 sn	% 26,40	3.31 sn	% 51,2

YAK algoritması ile optimize edilmiş FOPID kontrolörü uygulamanın, gaz türbini santralinin çıkış sinyalinin hem pozitif hem de negatif bölgesinde yerleşme süresi açısından diğerlerine göre daha iyi sonuç verdiği Tablo 8 incelendiğinde görülmektedir. Diğer üç algoritma da değerlendirildiğinde hem pozitif hem de negatif bölge için yerleşme süreleri en küçükten en büyüğe doğru sıralandığında sırasıyla SSA-FOPID, BOA-FOPID, AAO-FOPID olduğu görülür. Yüzde en büyük aşım açısından değerlendirildiğinde ise YAK algoritması ile elde edilen FOPID denetleyici parametrelerinin uygulandığı sistemin hem pozitif hem de negatif bölgelerdeki çıkış eğrisi değerlerinin diğerlerinden küçük olduğu ve hatta hiç aşım yapmadığı (%0 olduğu) görülmektedir. SSA algoritması ile optimize edilmiş FOPID kontrolörü uygulanmış gaz türbini santral sistemi cevap eğrisinin, pozitif bölgede % 3,51 aşım ile; negatif bölgede ise % 6,82 aşım ile ikinci en küçük yüzde aşım değerlerine sahip olduğu görülmektedir. BOA ve AAO algoritmaları ile optimize edilmiş olan FOPID kontrolörlerin kullanıldığı sistem çıkış eğrileri incelendiğinde BOA ile optimize edilen sistem çıkış sinyalinin hem pozitif bölgede hem de negatif bölgede üçüncü sırada aşım değerine sahip olduğu ve AAO ile optimize edilen FOPID kontrolöre sahip sistem çıkış sinyalinin ise diğerlerine göre en büyük yüzde aşım miktarına sahip olduğu görülmektedir. Bu durumda YAK algoritması ile optimize edilen FOPID kontrolör parametrelerinin hem yüzde aşım miktarı kriterine hem de yerleşme süresi kriterine göre diğerlerinden bu sistem için daha üstün olduğu anlaşılmaktadır.

#### IV. SONUÇ VE DEĞERLENDİRME

Bu çalışmada, basitleştirilmiş bir gaz türbinli güç santral modelinin kontrolü için doğadan esinlenen algoritmalar olan BOA, SSA, YAK ve AAO algoritmaları yardımıyla parametre optimizasyonu yapılmış kesir dereceli PID (FOPID) kontrolörler kullanılmıştır. Bu algoritmalarla optimize edilmiş FOPID kontrolörlerin uygulandığı gaz türbin santralinin çıkış sinyallerinin hem pozitif hem de negatif bölgelerinde yerleşme süresi ve yüzde aşım kriterleri açısından karşılaştırmalar yapıldığında; ele alınan dört optimizasyon algoritması içinde yapay arı kolonisi algoritması (YAK) ile elde edilen FOPID kontrolör sayesinde sistem çıkış sinyalinin hiç aşım yapmadığı ve diğerlerine göre kısa sürede yerleştiği görülmüştür. Bu sonuçlarla, YAK ile optimize edilmiş FOPID kontrolörlerin gaz türbinli elektrik santrallerine uygulandığında; oluşan yük değişimlerine vereceği hızlı cevaplar ile verim artışının sağlanabileceği düşünülmektedir.

#### KAYNAKLAR

- [1] Arıkuşu, Y., Bayhan, N. & Tiryaki, H. (2019), Comparison of Conventional and Modern Controllers for a Gas Turbine Power Plant, *Electrica*, 19(2), 146-157.
- [2] Tiryaki, H. (2013). Modern Kontrol Yöntemlerinin Yük Dağıtım Sistemlerinde Uygulanması, Doktora Tezi, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [3] Podlubny, I. (1994) Fractional Order Systems and Fractional Order Controllers, Slovak Academy of Sciences Institute of Experimental Physics.
- [4] Göktürk, E. (2009). Kesirli PID Tasarım Yöntemi ve Klasik PID ile Karşılaştırmalar, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [5] Sezer, K. S. (2021), Metasezgisel Algoritmalarla Optimize Edilmiş Kesir Dereceli PID kontrolörler ile Gaz Türbin Elektrik Santralinin Kontrolü, Yüksek Lisans Tezi, İstanbul Üniversitesi-Cerrahpaşa, Lisansüstü Eğitim Enstitüsü, İstanbul.
- [6] Tzanetos, A. & Dounias, G. (2021), Nature inspired optimization algorithms or simply variations of metaheuristics?, *Artificial Intelligence*, 54, 1841–1862
- [7] Mirjalili, S. & Lewis, A. (2016), The Whale optimization Algorithm, *Advances in Engineering Software*, 95, 51-67.

- [8] Mirjalili, S., Gandomi, A. H., Mirjalili, S.Z., Saremi, S., Faris, H. & Mirjalili, S.M., (2017), Salp Swarm Algorithm: A Bio-Inspired Optimizer for Engineering Design Problems, *Advances in Engineering Software*, 114, 163-191.
- [9] Karaboğa, D. (2005), An Idea Based on Honey Bee Swarm For Numerical Optimization, Technical Report-TR06, Erciyes University, Kayseri, October 2005.
- [10] Zhao, W., Wang, L. & Zhang, Z., (2018), A Novel Atom Search Optimization for Dispersion Coefficient Estimation in Groundwater, *Future Generation Computer Systems*, 91, 601-610.
- [11] MATLAB 2020a Reference Manual, 2020. (Licence no: 40914345)
- [12] Hekimoğlu, B. (2020). Çekirge optimizasyon algoritması kullanılarak çok makinalı güç sistemi için gürbüz kesir dereceli PID kararlı kılıcısı tasarımı, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 35(1), 165-180, DOI: 10.17341/gazimmfd.449685.