

Kaotik Tabanlı PSO Algoritması ile PID Kontrol Parametrelerinin Optimizasyonu ve Sistem Performansının İncelenmesi

Muhammed Salih SARIKAYA^{1*} , Onur DEMİREL² , Sezgin KAÇAR³ , Adnan DİRDİYOK⁴ 

¹ Mekatronik Mühendisliği, Teknoloji Fakültesi, Sakarya Uygulamalı Bilimler Üniversitesi, Türkiye.

salihsarikaya@subu.edu.tr

² Mekatronik Mühendisliği, Teknoloji Fakültesi, Sakarya Uygulamalı Bilimler Üniversitesi, Türkiye.

onurdemirel@subu.edu.tr

³ Elektrik ve Elektronik Mühendisliği, Teknoloji Fakültesi, Sakarya Uygulamalı Bilimler Üniversitesi, Türkiye.

sakar@subu.edu.tr

⁴ Mekatronik Mühendisliği, Teknoloji Fakültesi, Sakarya Uygulamalı Bilimler Üniversitesi, Türkiye.

derdiyok@subu.edu.tr

ÖZ

Bu çalışmada, Parçacık Sürü Optimizasyon Algoritmasının kaotik sistemlerle birleştirilerek kontrolcü tasarımı üzerindeki performansının iyileştirilmesi amaçlanmaktadır. Kontrol parametrelerinin optimum şekilde ayarlanması metodolojik yöntemlerle (Ziegler-Nichols, Adaptif yöntemler vb) veya çoğunlukla uzman bilgisine dayalı deneme-yanılma yaklaşımıyla gerçekleştirilmektedir. Daha etkin çözümler sunabilmeleri açısından meta-sezgisel optimizasyon algoritmalarının kullanımı son yıllarda ön plana çıkmaktadır. Bu çalışma kapsamında, Parçacık Sürü Optimizasyonu ve bu algoritmanın kaotik sistemlerle birleştirilmiş versiyonu olan Kaotik Parçacık Sürü Optimizasyonu kullanılarak PID kontrolcünün parametreleri optimize edilmiştir. Optimizasyon sürecinde performans kriteri olarak Zaman Ağırlıklı Hatanın Karesinin İntegrali esas alınmıştır. Elde edilen bulgular, kaotik sistemlerin Parçacık Sürü Optimizasyon algoritmasına entegrasyonunun, algoritmanın minimum hata değerine yakınsama başarımını artırdığını ortaya koymuştur.

Anahtar Kelimeler: DC motor kontrol, Kaotik sistemler, Kaotik tabanlı optimizasyon, Parçacık sürü optimizasyonu

* Sorumlu yazarın e-posta adresi: salihsarikaya@subu.edu.tr

Cite as: Sarıkaya, M.S., Demirel, O., Kaçar, S. & Derdiyok A. (2025). Kaotik Tabanlı PSO Algoritması ile PID Kontrol Parametrelerinin Optimizasyonu ve Sistem Performansının İncelenmesi, *Journal of Smart Systems Research*, 6(1), 45-61. <https://doi.org/10.58769/joinssr.1692401>

Optimization of PID Control Parameters Using a Chaotic-Based PSO Algorithm and Analysis of System Performance

ABSTRACT

This study aims to improve the performance of optimization algorithms in controller design by combining them with chaotic systems. The optimal tuning of control parameters is typically carried out using methodological approaches (such as Ziegler-Nichols, adaptive methods, etc.) or trial-and-error strategies that often rely on expert knowledge. In recent years, the use of metaheuristic optimization algorithms has gained prominence due to their ability to offer more effective solutions. Within the scope of this study, the parameters of the PID controller were optimized using the Particle Swarm Optimization algorithm and its version combined with chaotic systems, known as the Chaotic Particle Swarm Optimization. The performance criterion employed in the optimization process was the Integral of Time-weighted Squared Error. The results demonstrated that the integration of chaotic systems into the Particle Swarm Optimization algorithm enhances its ability to converge toward the minimum value. This study has shown that optimization algorithms combined with chaotic structures can be successfully applied in control systems.

Keywords: DC motor control, Chaotic systems, Chaotic-Based optimization, Particle swarm optimization

1 Giriş

Parçacık Sürü Optimizasyonu (PSO), robotik, kontrol sistemleri ve yapay zekâ gibi çok çeşitli mühendislik alanlarında uygulanabilen, esnek ve güçlü bir meta-sezgisel optimizasyon yöntemi olarak dikkat çekmektedir. Yüksek boyutlu ve karmaşık çözüm uzaylarını etkili biçimde tarayabilen PSO, problemlere ilişkin optimal çözümler üretebilme yeteneğiyle öne çıkmaktadır.

PSO algoritmasının çeşitli uygulama alanlarındaki kullanımlarını inceleyen literatür çalışması yapılmış ve elde edilen sonuçlar derlenerek sunulmuştur. Zhong, PSO'nun bir robot kolunun dinamik parametrelerini etkili bir biçimde tanımlayabildiğini ve bu bağlamda Ağırlıklı En Küçük Kareler (AEKK) ve Genetik Algoritmalar (GA) gibi geleneksel yöntemlere kıyasla daha yüksek doğruluk ve daha hızlı yakınsama sağladığını ortaya koymuştur [1]. Benzer şekilde, Abedinifar vd. doğrusal olmayan dinamik sistemlerde parametre tanımlama amacıyla PSO'nun sıklıkla tercih edildiğini belirtmişlerdir [2]. Özellikle otonom sistemler bağlamında da PSO'nun dikkat çekici uygulamaları bulunmaktadır. Coşar, PSO ve diğer sürü zekâsı algoritmalarının insansız hava araçlarının (İHA) görev planlamasında hem bireysel hem de iş birliğine dayalı operasyonlarda önemli rol oynadığını vurgulamıştır [3]. Ayrıca, dinamik ve belirsizlik içeren ortamlarda çözüm kalitesini artırmak amacıyla PSO'nun adaptif versiyonları da geliştirilmiştir. Bu kapsamda, Adaptif PSO (APSO) gibi iyileştirilmiş varyantların, küresel optimum çözüme daha etkin ulaşabildiği gösterilmiştir [4]. PSO'nun yapay sinir ağları (YSA) ile entegrasyonu ise özellikle tahmin ve sınıflandırma görevlerinde öne çıkmaktadır.

Literatürde PSO, kontrol mühendisliği alanında da yaygın olarak kullanılmakta olup, özellikle PID kontrolörlerin parametrelerinin ayarlanmasında sıkça tercih edilmektedir. Peltier soğutma sistemi üzerinde gerçekleştirilen çalışmada, PSO ile optimize edilen PID parametrelerinin kontrol performansını anlamlı biçimde iyileştirildiği gösterilmiştir [5]. Tahtawi vd. AX-12 servo motorunun konum kontrolü için PSO-tabanlı bir PID kontrolör tasarlayarak, PSO'nun robotik sistemlerde kararlılık ve yanıt süresi açısından önemli iyileştirmeler sağladığını ortaya koymuştur [6]. Benzer şekilde, büyük zaman gecikmesine sahip boru hattı robotlarında PSO'nun etkinliğini test edilmiş ve algoritmanın bu tür sistemlerde başarılı sonuçlar verdiği gösterilmiştir [7]. Fırçasız DC (BLDC) motorların hız kontrolünde bulanık mantık ile PSO'yu birleştirilerek geleneksel yöntemlere kıyasla daha kararlı ve hızlı yanıt veren bir kontrol sistemi elde edilmiştir [8]. Moghaddas vd. benzer şekilde, DC motorların PID kontrolünde

PSO'nun kontrol performansını anlamlı şekilde artırdığını raporlamıştır [9]. Zhu vd. ise, fotovoltaiik sabit basınçlı su tedarik sistemine yönelik çalışmalarında PID parametrelerinin PSO ile optimize edilmesini önermiş; simülasyon sonuçları, klasik PID kontrolörlere kıyasla PSO ile optimize edilmiş kontrol yapısının daha üstün dinamik yanıt ve kararlı durum performansı sergilediğini ortaya koymuştur [10]. Bu çalışmalar, PSO algoritmasının farklı mühendislik disiplinlerinde geleneksel yöntemlere kıyasla daha etkili çözümler sunabildiğini ve özellikle kontrol uygulamalarında başarıyla uygulanabileceğini göstermektedir.

Kaotik Parçacık Sürü Optimizasyonu (KPSO), geleneksel PSO algoritmasının geliştirilmiş bir varyantı olup, çok sayıda çalışmada ele alınmıştır. Kaos teorisinin PSO algoritmasına entegrasyonu, arama alanında çeşitliliği artırarak algoritmanın erken yakınsama ve yerel minimumlara takılma gibi temel sınırlılıklarını aşmasına olanak tanımaktadır. Bu doğrultuda, literatürde KPSO'nun çeşitli uygulama alanlarındaki kullanımları incelenmiş ve ilgili çalışmalardan elde edilen bulgular derlenerek sunulmuştur. Mühendislik yapı tasarımı alanında kullanmak üzere, karmaşık başlangıç koşulları ve doğrusal olmayan optimizasyon tekniklerini içeren KPSO algoritmasını önerilmiş ve geleneksel PSO'ya kıyasla daha hızlı yakınsama ve daha yüksek çözüm kalitesi sağlandığı ortaya koyulmuştur [11]. Makine öğrenmesi ve tarımsal görüntü işleme alanlarında Dong, KPSO algoritmasını Destek Vektör Makineleri ile entegre ettiği bir çalışmada, bitki hastalıkları ve zararlılarının tespiti için kullanılan sistemde kaotik dizilerin başlangıç popülasyonundaki çeşitliliği artırarak optimizasyon sürecinin başarımını artırdığını ve yerel minimumlara takılma olasılığını azalttığını göstermiştir [12].

Kontrol sistemlerine yönelik çalışmalar ise KPSO'nun klasik denetleyici tasarımlarına sağladığı katkıyı ortaya koymaktadır. Enerji sistemleri kapsamında, Huang vd, doğrudan tahrikli bir dalga enerjisi dönüştürücüsü için KPSO'yu kullanarak, algoritmanın değişen çalışma koşulları altında kontrol sisteminin tepki süresini azaltma potansiyeline sahip olduğunu ortaya koymuştur [13]. Yüksek hızlı taşıma sistemlerinde, Cheng, trenlerin kaldırma kanadı açısı tahrik sistemlerinin hassas kontrolü için KPSO tabanlı bir denetim stratejisi önermiş ve bu algoritmanın yüksek doğruluk gerektiren uygulamalarda etkin bir çözüm sunduğunu göstermiştir [14]. Mobil robotlar ve konumlandırma sistemleri özelinde, mobil konum tahmini problemi için KPSO tabanlı yeni bir konumlandırma algoritması geliştirilmiştir [15]. Çalışmada, algoritmanın yüksek konum doğruluğu sağladığı ve sistemsel hata oranlarını azalttığı belirtilmiştir. Ayrıca, İnsansız Kara Araçları (İKA) için önerilen Model Öngörülü Kontrol (Model Predictive Control-MPC) yapısı için PSO-MPC tabanlı bir yol izleme algoritmasında, KPSO'nun karmaşık sürüş senaryolarında düşük doğruluk ve zaman gecikmesi gibi sorunları aşma konusundaki etkili rolü vurgulanmıştır [16]. Huang vd. direksiyon teli sistemleri için geliştirilen MPC yapısının optimizasyonunda KPSO'yu kullanmış ve bu sayede kontrol başarımını önemli ölçüde artırmışlardır [17]. Chu vd. kaotik başlangıç mekanizması ve uyarlanabilir hız faktörlerini entegre ederek İnsansız Hava Araçları (İHA) için KPSO tabanlı bir rota planlama algoritması geliştirmiş, bu yaklaşımın İHA'ların karmaşık çevrelerde güvenli ve etkin bir şekilde gezinmesini sağladığını göstermiştir [18]. KPSO, Lineer Kuadratik Regülatör ile entegre edilerek, PSO'nun hızlı yakınsama yeteneği ile kaotik algoritmaların keşif gücünü bir araya getirilmiş ve bu yapının kontrol uygulamalarındaki performans artışına önemli katkı sağladığı gösterilmiştir [19]. Engel önleyici bir kontrol sisteminde kendi kendini koruyabilen bir denetleyici parametrelerinin optimize edilmesinde KPSO'nun etkili sonuçlar verdiği bildirilmiştir [20]. Güç Sistemi Dengeleyicisi ve PID denetleyici parametrelerinin optimizasyonu için KPSO kullanılarak sistem kararlılığı ve dinamik yanıt açısından iyileştirmeler elde edilmiştir [21]. Bu kapsamlı literatür değerlendirmesi, KPSO algoritmasının yalnızca klasik PSO'ya kıyasla daha güçlü bir çözüm sunmakla kalmadığını, aynı zamanda enerji, yapay zekâ, konumlandırma, otonom sistemler ve özellikle kontrol mühendisliği gibi çok çeşitli disiplinlerde yüksek başarılı bir optimizasyon aracı olarak kullanılabildiğini göstermektedir. Literatürde kaotik PSO uygulamalarında, genellikle yalnızca atalet ağırlığının kaotikleştirilmesine odaklanıldığı ve kaotik yapı

olarak çoğunlukla ayırık zamanlı sistemlerin (örneğin, Logistic veya Tent map) tercih edildiği gözlemlenmektedir. Ancak bu yaklaşımlar, algoritmanın rastgele sayı üretim sürecine doğrudan müdahale etmediğinden ve kaotik etkinin algoritmanın tüm dinamiklerine yayılmasını sınırladığından, potansiyel performans iyileştirmeleri açısından yetersiz kalabilmektedir. Dolayısıyla bu çalışmada, PSO algoritmasında kullanılan rastgele sayılar sürekli zamanlı bir kaotik sistem aracılığıyla üretilmiştir. Böylece algoritmanın tüm parametre güncellemelerinde kaotik etkinin doğrudan yer alması sağlanmıştır. Bu yönüyle çalışma, hem kaotik yapıların PSO algoritmasına entegrasyonuna yönelik farklı bir yaklaşım sunmakta hem de sürekli zamanlı kaotik sistemlerin bu bağlamda uygulanabilirliğine dair literatüre özgün bir katkı sağlamaktadır.

2 Materyal ve Yöntem

Bu çalışmada DC motorun hız kontrolünde, kontrolcü parametrelerinin seçimine ait optimizasyon probleminin çözümü için PSO ve PSO'nun kaotik sistemlerle entegre edilmiş versiyonu olan KPSSO kullanılmıştır. Bu optimizasyon tekniklerinde amaç fonksiyonu olarak Zaman Ağırlıklı Hatanın Karesinin İntegrali (ZAHKİ) seçilmiştir. ZAHKİ, hata karelerini zamana göre ağırlıklandırarak sistemin hızlı ve kararlı bir şekilde nihai değerine ulaşmasını hedeflemektedir.

2.1 Parçacık Sürü Optimizasyonu Algoritması

PSO doğadan esinlenilerek geliştirilen bir optimizasyon algoritmasıdır. PSO kuş ya da balık sürülerinin grup halinde hareket ederek bir hedefe ulaşma davranışlarından ilham alınan sürü tabanlı bir optimizasyon algoritmasıdır. PSO bir çözüm kümesini temsil eden "parçacıklar" ile çalışır. Bu parçacıklar, çözüm uzayında rastgele başlangıç noktalarından başlar. Yiyecek arama alanında rastgele bir şekilde dağılan sürü elemanları, yiyecek bulamadıklarında aynı anda yönlerini ve pozisyonlarını değiştirir. Yaptıkları bu yeni yön ve pozisyon değişikliklerinin ardından, sürü içindeki bireyler birbirleriyle yiyeceğin konumu ve yiyeceğe en yakın kuşun pozisyonu hakkında bilgi paylaşarak hedefe ulaşmaya çalışır. Arama alanındaki her bir parçacık, kendi pozisyonunu ve hızını, yiyeceğe en yakın olan parçacığa göre günceller. Bu bilgi paylaşımı sürecinde, her parçacık aşağıdaki iki önemli değeri hafızasında tutar ve bu iki değere göre hareket eder.

-Bireysel En İyi Konum (*pBest*): Parçacığın şimdiye kadar elde ettiği en iyi çözüm

-Sürü En İyi Konum (*gBest*): Tüm parçacıklar arasında elde edilen en iyi genel çözüm

PSO'da her bir iterasyonda parçacıkların hızları güncellenir. Güncelleme için kullanılan denklem aşağıda verilmiştir.

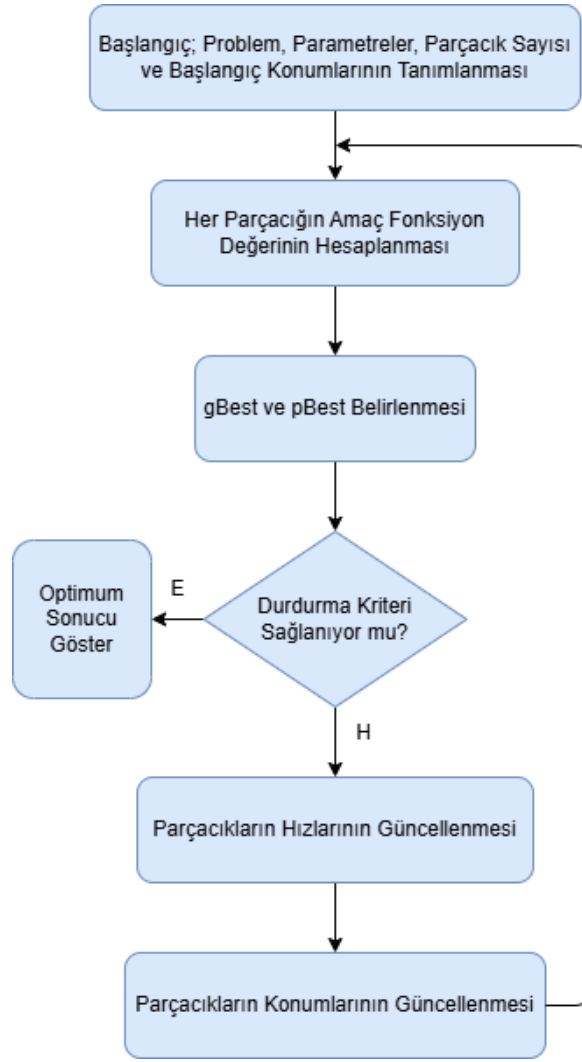
$$v_i(k+1) = w * v_i(k) + c_1 * r_1 * (pBest - x_i(k)) + c_2 * r_2 * (gBest - x_i(k)) \quad (1)$$

Denklem (1)'de verilen k güncel iterasyon değerini, i parçacık sayısını, v_i i . parçacığın hızını, c_1, c_2 öğrenme katsayılarını, x_i i . parçacığının pozisyonunu, w atalet ağırlığını ifade etmektedir. Atalet ağırlığı, parçacığın hızını ne kadar koruyacağını belirlerken, r_1 ve r_2 0 ile 1 arasında değişen rastgele değerlerdir.

Denklem (1) ile parçacıkların bir sonraki iterasyonda kullanacakları hız verileri elde edildikten sonra parçacıkların pozisyonları Denklem (2) ile güncellenir.

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad (2)$$

PSO algoritmasına ait akış şeması Şekil 1'de verildiği gibidir.



Şekil 1: PSO akış şeması

Şekil 1’de başlangıç aşamasında parçacıkların pozisyonu ve hızı rastgele atanır. Uygunluk kontrolü aşamasında her parçacığın mevcut konumu bir amaç fonksiyonuyla değerlendirilerek bireyin ve sürünün en iyi konumları $pBest$ ve $gBest$ güncellenir. Parçacıkların hızı ve konumu sırasıyla Denklem 1 ve Denklem 2’ye göre güncellenir. Durum kontrolü aşamasında belirlenen durdurma kriterine (maksimum iterasyon sayısı veya hedef uygunluk değeri) ulaşıldıysa döngü durdurulur; aksi takdirde döngü devam ettirilir.

2.2 Kaotik Parçacık Sürü Optimizasyonu

Kaotik Parçacık Sürü Optimizasyonu, klasik PSO algoritmasının kaotik haritalarla geliştirilmiş bir türevidir. KPSO, klasik PSO’nun yerel minimuma sıkışma ve yetersiz çeşitlilik gibi sorunlarını aşmak amacıyla kaos teorisinden faydalanır. Kaotik sistemler, deterministik olmalarına rağmen, başlangıç koşullarına son derece duyarlı doğrusal olmayan dinamik sistemlerdir. Bu özellikleri optimizasyon süreçlerinde daha etkili bir keşif sağlayabilir. KPSO, PSO algoritmasının bazı bileşenlerini kaotik davranışla düzenler. Bu genellikle kaosun matematiksel temsili olan kaotik haritalarla, rastgele sayıların yerini alarak algoritmanın daha etkili bir şekilde çalışmasını sağlar. Bu işlem PSO’nun hız güncelleme denklemindeki rastgele sayılarında (r_1, r_2) gerçekleştirilir. Böylece, klasik PSO’nun rastgele arama davranışı yerine, kaotik davranışların daha düzenli ve etkili bir şekilde uygulanması sağlanır. Bu yaklaşım, algoritmanın daha geniş bir çözüm uzayını keşfetmesine ve yerel minimumdan kaçmasına

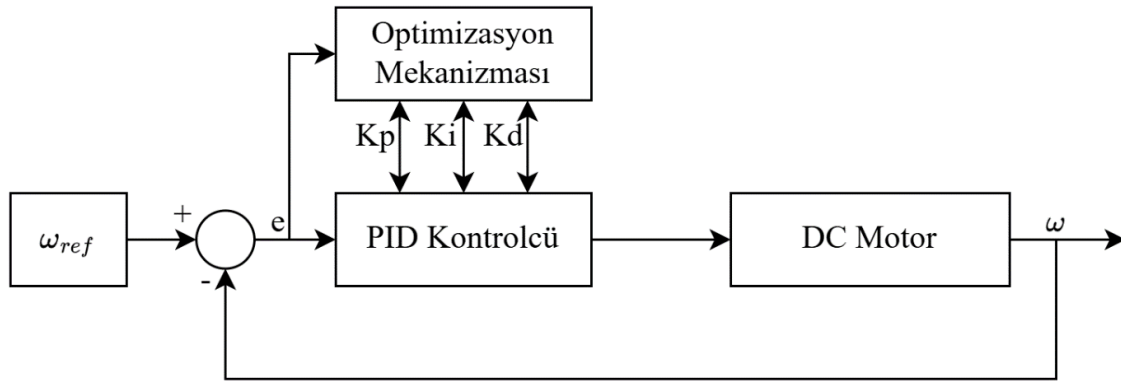
yardımcı olur. Rastgele sayıların elde edilmesi için kullanılan Duffing-Van Der Pol Kaotik sistemine ait denklem aşağıda verilmiştir [22].

$$\begin{aligned}\dot{x}(t) &= y(t) \\ \dot{y}(t) &= a * (1 - x^2(t)) * y(t) - x^3(t) + b * \cos(c * z(t)) \\ \dot{z}(t) &= 1\end{aligned}\quad (3)$$

Değişkenler $a=0.2$, $b=5.8$ ve $c=3$ olarak alınmıştır. Başlangıç koşulları ise $x(0) = 0$, $y(0) = 0$, $z(0) = 0$ olarak seçilmiştir. Otonom olmayan bu sistem $z=t$ dönüşümüyle otonom hale getirilmiştir [23]. Değişkenler ve başlangıç koşulları literatürde yaygın kullanılan değerlerden seçilmiştir.

2.3 DC Motor Hız Kontrolü için PID Parametrelerinin PSO ve KPSO ile Optimizasyonu

Bu çalışmada, önerilen optimizasyon yaklaşımının etkinliği, PID kontrolör parametrelerinin ayarlanması problemi üzerinden değerlendirilmiştir. Bu doğrultuda, kullanılan algoritmaların amaç fonksiyonunu minimum değere yakınsama performansları analiz edilmiştir. DC motorun hız kontrolü uygulaması kapsamında, PID kontrolörün parametreleri, hata tabanlı ZAHKİ amaç fonksiyonu esas alınarak, Parçacık Sürü Optimizasyonu ve Kaotik Parçacık Sürü Optimizasyonu algoritmaları ile optimize edilmiştir. DC motorun kapalı çevrim kontrolüne ait blok diyagram Şekil 2’de verilmiştir.



Şekil 2: Tasarlanan sistemin blok diyagramı

DC motorun dinamik davranışı diferansiyel denklemler ile ifade edilir ve matematiksel modeli karmaşık frekans düzleminde transfer fonksiyonu ile verilir. DC motorun matematiksel modeli aşağıdaki gibidir.

$$\begin{aligned}V_a &= i_a R_a + L_a \frac{di_a}{dt} + e_b \\ e_b &= K_e \omega(t) \\ T &= J \frac{d\omega}{dt} + B\omega + T_l \\ T &= K_m i_a(t)\end{aligned}\quad (4)$$

Denklem (4)’de V_a , i_a , R_a ve L_a sırasıyla motorun armatür gerilimi, akımı, direnci ve endüktansıdır. Ayrıca e_b zıt elektro motor kuvveti, K_e gerilim sabitini, T üretilen momenti, J atalet momentini, B sönümleme sabitini, T_l yük momentini, K_m moment sabitini ve ω motorun açısal hızını ifade etmektedir. DC motorun parametreleri Tablo 1’de verilmiştir [24].

Tablo 1: DC motor parametreleri

R_a	0.517Ω
L_a	$0.0573H$
B	0.000244
J	$0.00000145kgm^2/s^2$
K_e	$0.0112Vs/rad$
K_m	$0.0112Nm/A$
Redüktör oranı	1/52

DC motorun hız kontrolünde kontrolcü olarak PID seçilmiştir. PID kontrolcünün K_p , K_i ve K_d parametrelerinin, DC motorun arzu edilen performansı sergileyecek şekilde ayarlanması gerekmektedir. Bu çalışmada kontrolcü parametreleri PSO ve KPSO algoritmaları kullanılarak belirlenmiştir. PID parametrelerinin optimizasyonunda hata sinyali $e(t)$ minimize edilmesi amaçlanmıştır. Amaç fonksiyonu olarak ise Denklem (5) ile ifade edilen Zaman Ağırlıklı Kare Hata İntegrali kullanılmıştır.

$$ZAHKİ = \int te^2(t)dt \quad (5)$$

3 Bulgular

Bu bölümde, PSO ve KPSO algoritmaları kullanılarak optimize edilen PID kontrolcünün simülasyon ortamındaki performans sonuçları sunulmaktadır. Bu çalışma, MATLAB/Simulink R2022a sürümü kullanılarak gerçekleştirilmiştir. Simülasyon süresi 0.1 saniye olarak belirlenmiş, sabit zaman adımı ise 0.001 saniye olarak seçilmiştir. Tüm simülasyon çalışmaları, Windows-10 64-bit işletim sistemine sahip, 2.60 GHz işlemcili ve 12 GB RAM donanımlı bir bilgisayarda yürütülmüştür. PSO ve KPSO optimizasyon algoritmalarında kullanılan parametreler, Tablo 2'de detaylı bir şekilde verilmiştir.

Tablo 2: Optimizasyon yöntemlerinde kullanılan parametreler

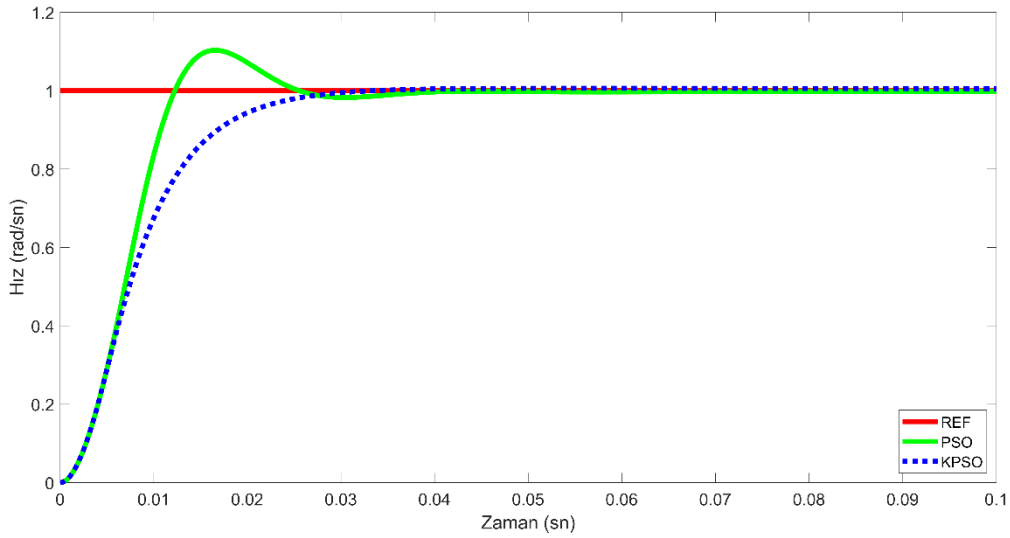
KPSO	Parçacık sayısı $i=50$ İterasyon sınırı=50 Atalet ağırlığı (w)=1 Bireysel öğrenme kat sayısı (c_1)=2.1 Sosyal öğrenme kat sayısı (c_2)=2.1
PSO	Parçacık sayısı $i=50$ İterasyon sınırı =50 Atalet ağırlığı (w)=1 Bireysel öğrenme kat sayısı (c_1)=2.1 Sosyal öğrenme kat sayısı (c_2)=2.1

PID kontrolcünün K_p , K_i ve K_d parametreleri, KPSO ve PSO algoritmaları ile Tablo 2'de belirtilen parametreler kullanılarak, hata tabanlı ZAHKİ amaç fonksiyonu üzerinden optimize edilmiştir. KPSO ve PSO algoritmaları ile elde edilen amaç fonksiyonu uygunluk değerleri ve optimize edilmiş kontrolcü parametreleri Tablo 3'te sunulmuştur.

Tablo 3: Optimize edilmiş PID parametreleri

	K_p	K_i	K_d	Uygunluk Değeri
KPSO	42.161766	166.226012	0.248816	0.000016
PSO	28.024663	176.891949	0.04789	0.000038

Tablo 3 incelendiğinde, KPSO algoritmasının daha düşük bir amaç fonksiyonu uygunluk değeri elde ettiği görülmektedir. Optimize edilen PID parametreleri kullanılarak DC motorun kapalı çevrim hız performansı simülasyon ortamında test edilmiş ve elde edilen sonuçlar Şekil 3'te sunulmuştur.



Şekil 3: Optimize edilmiş PID kontrolcülerin, birim basamak referans sinyali karşısında elde edilen sistem yanıtları

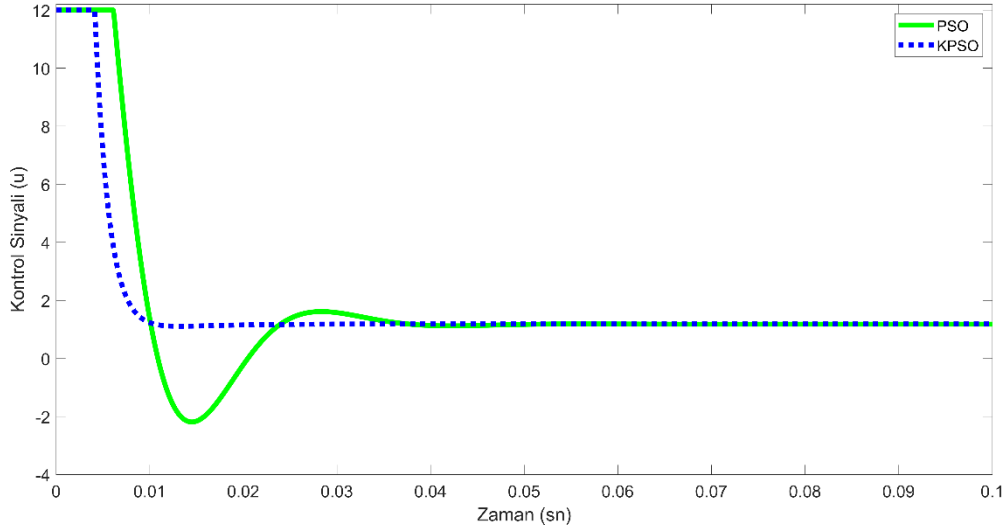
Şekil 3 incelendiğinde, PSO ile optimize edilmiş PID kontrolcünün aşım yaparak referans değerine oturduğu, KPSO ile optimize edilmiş PID kontrolcünün ise aşım yapmadan referans değere oturduğu görülmektedir. PSO ve KPSO ile optimize edilmiş kontrolcülerin performanslarını karşılaştırmak için yapılan testler sonucunda elde edilen yükselme zamanı (T_r), yerleşme süresi (T_s), yüzde aşım değeri (%OS) ve kararlı hal hatası (KHH) değerleri, Tablo 4'te sunulmuştur.

Tablo 4: PID kontrolcü cevaplarının karşılaştırılması

	T_r (sn)	T_s (sn)	%OS	KHH
KPSO	0.022	0.032	0.6	0.0025
PSO	0.014	0.036	8.3	0.0020

Tablo 4'te verilen sonuçlara göre, KPSO algoritması kullanılarak optimize edilmiş kontrolcü ile kontrol edilen sistemin cevabı, aşım yapmadan 0.032 saniyede %0.25 hata ile referans değerine oturduğu görülmektedir. PSO algoritması kullanılarak optimize edilmiş kontrolcü ile kontrol edilen sistemin cevabı ise %8.3 aşım gerçekleştirerek 0.036 saniyede %0.20 hata değeri ile referans değerine oturduğu görülmektedir.

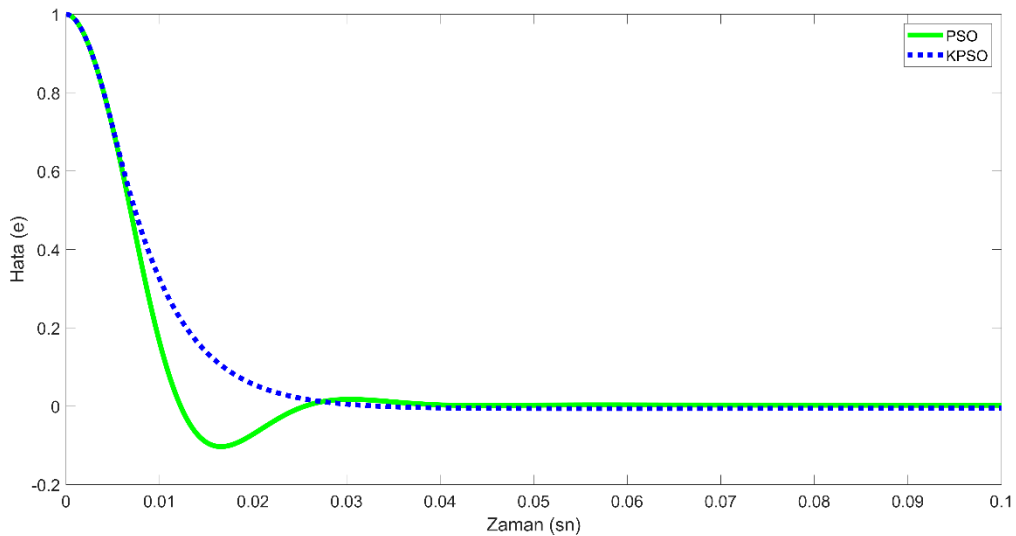
Şekil 4'te PSO ve KPSO algoritmalarıyla optimize edilen PID kontrolcülerinin birim basamak girişine karşılık ürettikleri kontrol sinyallerinin zamanla değişimi karşılaştırılmaktadır.



Şekil 4: PSO ve KPSO algoritmalarıyla ayarlanmış PID kontrolcüler tarafından birim basamak referans girişi için üretilen kontrol sinyalleri

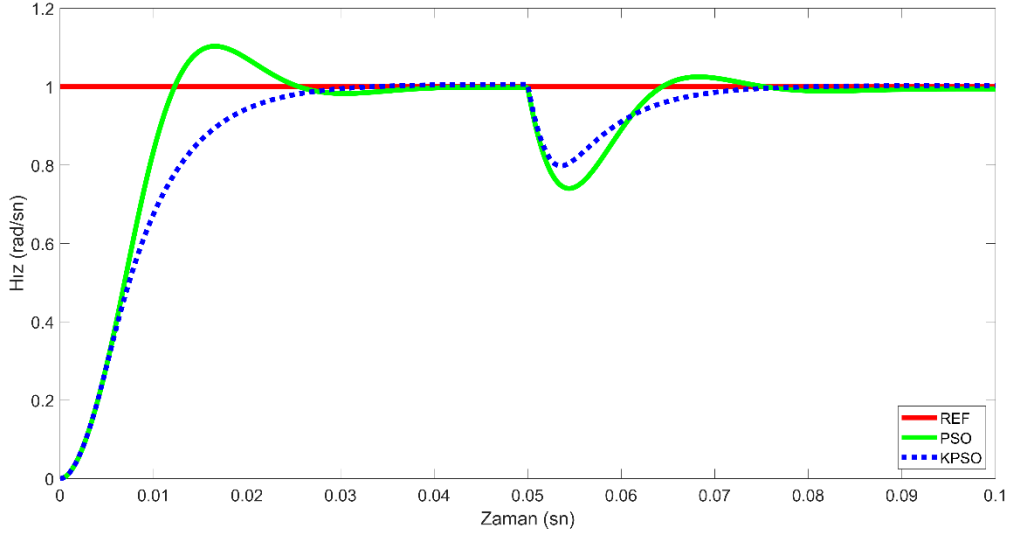
Şekil 4 incelendiğinde, PSO tabanlı kontrolcü, başlangıç bölgesinden çıktıktan sonra önemli düzeyde salınım sergilemekte ve bu durum geçici rejim davranışını olumsuz yönde etkilemektedir. Buna karşın KPSO algoritması ile optimize edilen kontrolcü daha yumuşak bir geçiş sergilemekte ve kısa sürede kararlı hale gelmektedir.

Şekil 5'te, PSO ve KPSO algoritmaları ile optimize edilen PID kontrolcülerin birim basamak referans sinyaline karşılık oluşan hata sinyallerinin zamana bağlı değişimleri karşılaştırmalı olarak sunulmuştur. PSO algoritmasına kıyasla, KPSO algoritması ile elde edilen PID kontrolcünün hata sinyalinin daha düşük salınım ve daha düzenli bir geçiş eğilimi sergilediği görülmektedir.



Şekil 5: PSO ve KPSO algoritmaları ile ayarlanmış PID kontrolcüler tarafından birim basamak referans girişi altında elde edilen hata sinyalleri

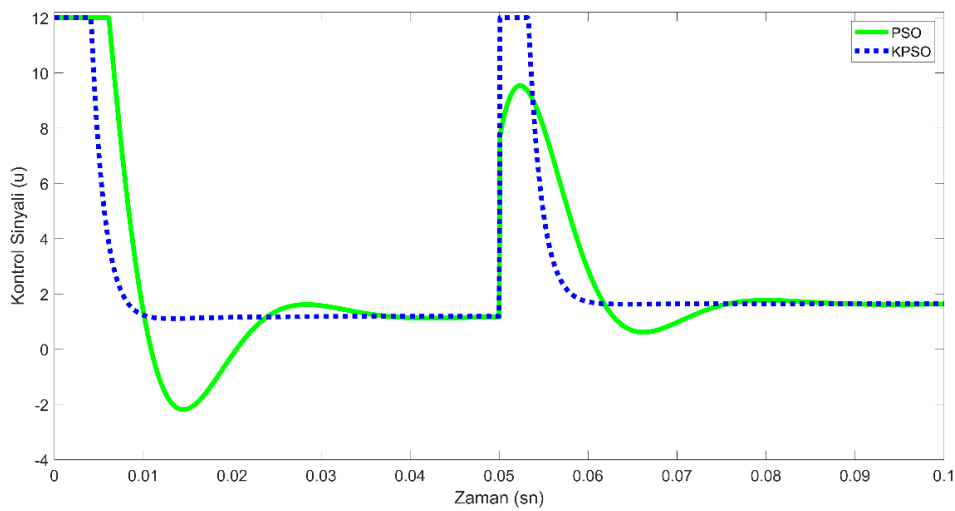
Simülasyon ortamında, DC motora 0.05 saniyede 0.01Nm büyüklüğünde bozucu bir etki uygulanmış ve elde edilen sistem tepkileri Şekil 6'da verilmiştir.



Şekil 6: Bozucu etki altında, optimize edilmiş PID kontrolcüler tarafından birim basamak referans sinyali için elde edilen sistem yanıtları

Kontrolcülerin performansları Şekil 6'da gösterilmiştir. Bozucu etkinin uygulanmasının ardından, PSO algoritması ile optimize edilmiş PID kontrolcüsüyle elde edilen sistem yanıtı, referans değere kıyasla %25 oranında azalma göstermiştir. KPSO algoritması ile optimize edilmiş PID kontrolcüsü için ise bu azalma %20 olarak gerçekleşmiştir. Bozucu etki süresince, PSO-PID kontrolcüsü %2.5 oranında aşım sergilemiş ve 0.086 saniyede %1'lik yerleşme bandı içerisine kalıcı olarak girmiştir. Öte yandan, KPSO-PID kontrolcüsü aşım gerçekleştirmemiş ve 0.071 saniyede %1'lik yerleşme bandı içerisine kalıcı olarak girmiştir.

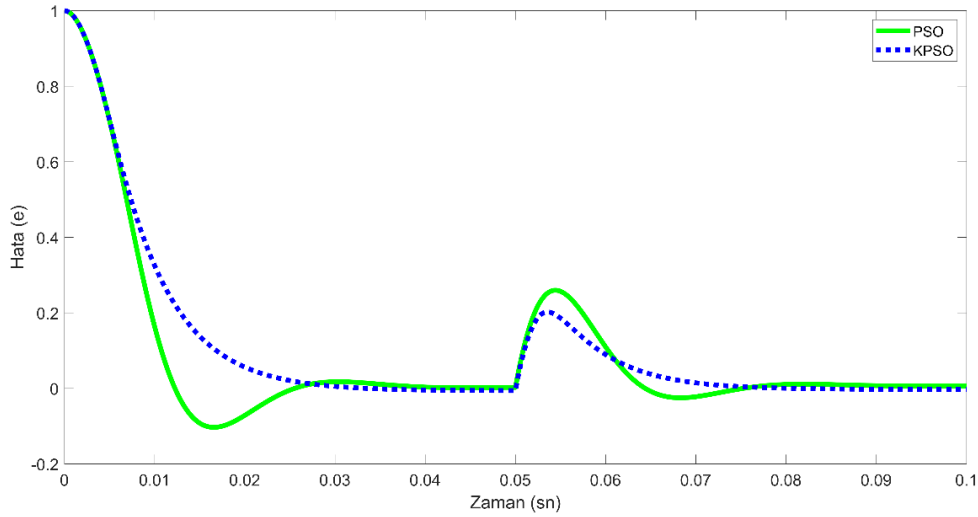
Şekil 7'de, 0.05 saniye anında sisteme 0.01 Nm büyüklüğünde bir bozucu etki uygulandığında, PSO ve KPSO algoritmalarıyla optimize edilmiş PID kontrolcülerinin ürettiği kontrol sinyallerinin zamana bağlı değişimi gösterilmektedir.



Şekil 7: PSO ve KPSO algoritmalarıyla ayarlanmış PID kontrolcüler tarafından, bozucu etki altında birim basamak referans girişi için üretilen kontrol sinyalleri

Şekil 7 incelendiğinde, bozucu etkinin uygulandığı anda, her iki kontrol yapısının da kontrol sinyalinde ani bir yükselme meydana gelmiş; sistemdeki denge bozulduğunda bu bozulmayı kompanse etmek için kontrolcülerin aktif şekilde müdahalede bulunduğu görülmektedir. Ancak, PSO ile elde edilen kontrol sinyali daha yüksek genlikli salınımlar sergileyerek tekrar dengeye ulaşmakta gecikirken, KPSO tabanlı kontrol yapısı bozucu etkinin etkisini daha hızlı ve daha az salınımla bastırabilmiştir.

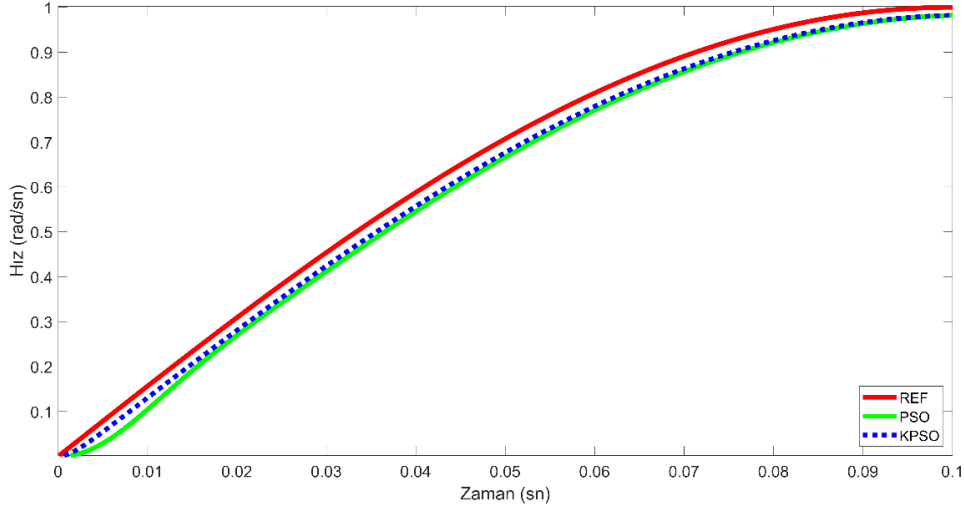
Şekil 8’de, 0.05 saniye anında uygulanan 0.01 Nm büyüklüğündeki bozucu etki sonrası PSO ve KPSO algoritmalarıyla optimize edilen PID kontrolcülerine ait hata sinyallerinin zamana bağlı değişimi sunulmuştur.



Şekil 8: PSO ve KPSO algoritmaları ile ayarlanmış PID kontrolcüler tarafından, bozucu etki altında birim basamak referans girişi için elde edilen hata sinyalleri

Şekil 8 incelendiği zaman, bozucu etkinin uygulandığı 0.05 saniye sonrasında her iki kontrol yapısında da hata sinyalinde ani bir artış gözlemlenmiştir. Ancak PSO tabanlı kontrolcü bu bozulmaya daha büyük bir hata artışıyla tepki verirken, KPSO kontrolcüsünün verdiği tepki daha yumuşak ve kısa süreli olmuştur. Ayrıca, bozucunun ardından PSO kontrolcüsünde tekrar salınımlar oluştuğu, buna karşın KPSO'nun daha hızlı ve salınımsız biçimde dengeye ulaştığı görülmektedir.

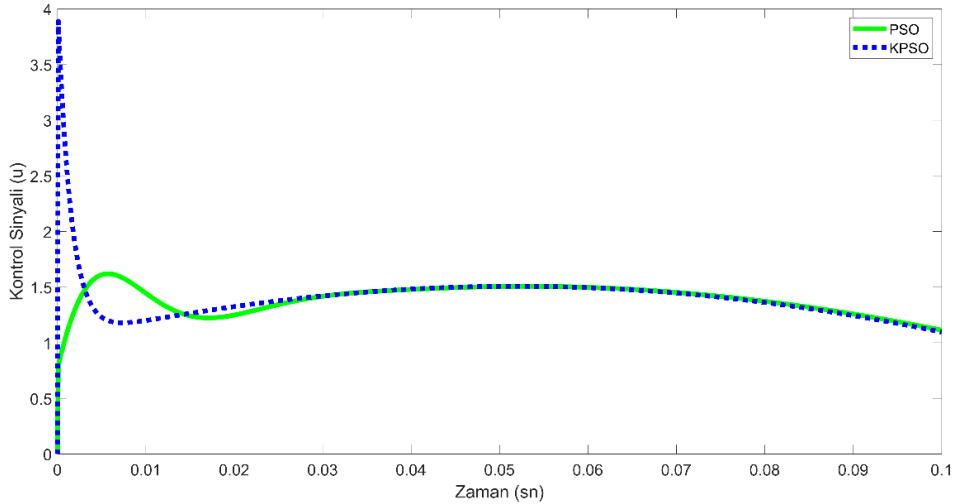
Kontrolcülerin dinamik referans sinyalini izleme yeteneklerini değerlendirmek amacıyla, DC motor sistemine çeyrek periyotluk sinüzoidal bir referans sinyali uygulanmıştır. Şekil 9’da PSO ve KPSO algoritmalarıyla optimize edilen PID kontrolcülere ait hız yanıtları karşılaştırmalı olarak sunulmaktadır.



Şekil 9: Optimize edilmiş PID kontrolcüler tarafından, çeyrek periyotluk sinüzoidal referans sinyali için elde edilen sistem yanıtları

Şekil 9 incelendiğinde, ilk aşamalarda yetersiz tepki vererek referansın altında seyretmiştir. KPSO ile optimize edilen kontrolcü ise daha kısa sürede referans eğrisine yaklaşmaktadır. Her iki algoritma ile elde edilen kontrol sistemlerinin, referans sinyale karşılık gelen hız profilini genel olarak başarılı bir şekilde izlediği görülmektedir. Ancak KPSO algoritması, referans sinyale daha yakın seyreden bir hız yanıtı sunarak izleme performansı açısından üstünlük sağlamıştır.

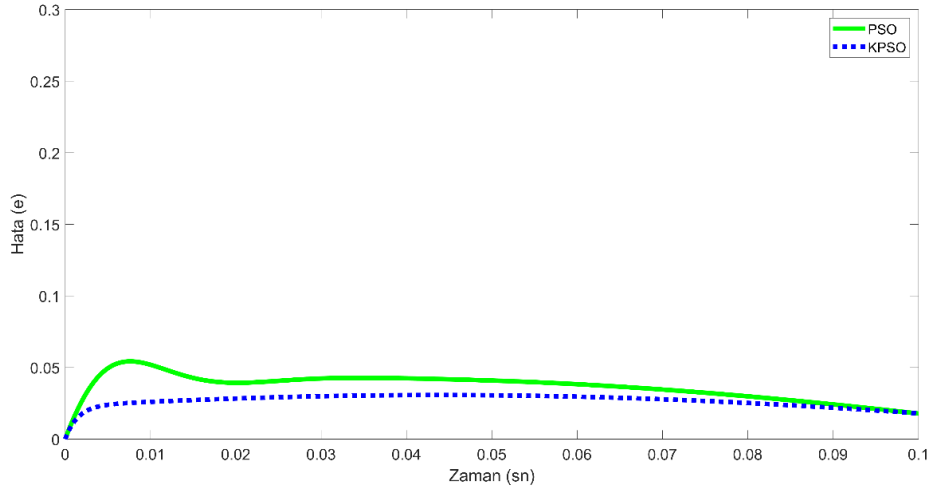
Şekil 10'da çeyrek periyotluk sinüzoidal referans sinyali altında PSO ve KPSO algoritmalarıyla optimize edilen PID kontrolcülerine ait kontrol sinyallerinin zamana bağlı değişimi sunulmaktadır.



Şekil 10: PSO ve KPSO algoritmalarıyla ayarlanmış PID kontrolcüler tarafından, çeyrek periyotluk sinüzoidal referans sinyali için üretilen kontrol sinyalleri

Şekil 10 incelendiğinde, KPSO tabanlı kontrolcünün başlangıçta oldukça yüksek bir kontrol sinyali ürettiği, ancak bu değer çok kısa sürede hızlı bir şekilde azaldığı ve sistemin kontrol sınırlarına daha hızlı şekilde ulaştığı görülmektedir. PSO algoritması ile optimize edilen kontrolcü ise daha düşük bir genlikle ve daha yumuşak bir kontrol tepkisi üretmiştir.

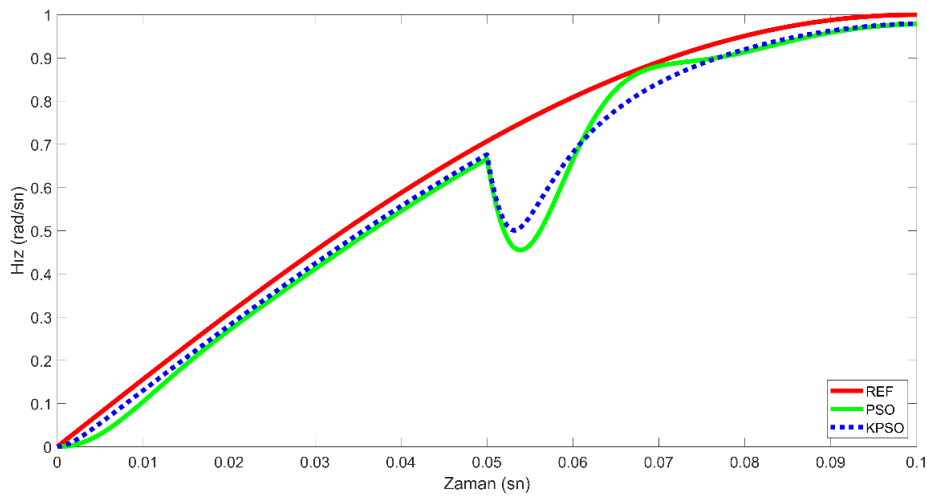
Şekil 11’de, sisteme çeyrek periyotluk sinüzoidal referans sinyali uygulanması durumunda PSO ve KPSO algoritmaları ile optimize edilen PID kontrolcülerinin oluşturduğu hata sinyallerinin zamana bağlı değişimi karşılaştırmalı olarak gösterilmektedir.



Şekil 11: PSO ve KPSO algoritmaları ile ayarlanmış PID kontrolcüler tarafından çeyrek periyotluk sinüzoidal referans sinyali altında elde edilen hata sinyalleri

Şekil 11 incelendiğinde, sistemin başlangıç tepkisinde PSO algoritması ile elde edilen PID kontrolcünün yaklaşık 0.055 seviyesinde bir maksimum hata değeri ürettiği gözlemlenmektedir. Buna karşılık, KPSO algoritması ile optimize edilen kontrolcü yaklaşık 0.03 değerinde daha düşük bir maksimum hata ile sistem yanıtını stabilize etmiş ve hatayı daha kısa sürede sönmülmüştür. KPSO algoritması hem daha düşük genlikte hem de daha kararlı bir hata profili sergileyerek izleme performansı açısından PSO algoritmasına göre üstünlük sağlamıştır. Oluşan hata sinyallerine ait Tip-1 norm değerleri incelendiğinde, PSO için 36.7112, KPSO için ise 26.7944 olarak hesaplanmıştır.

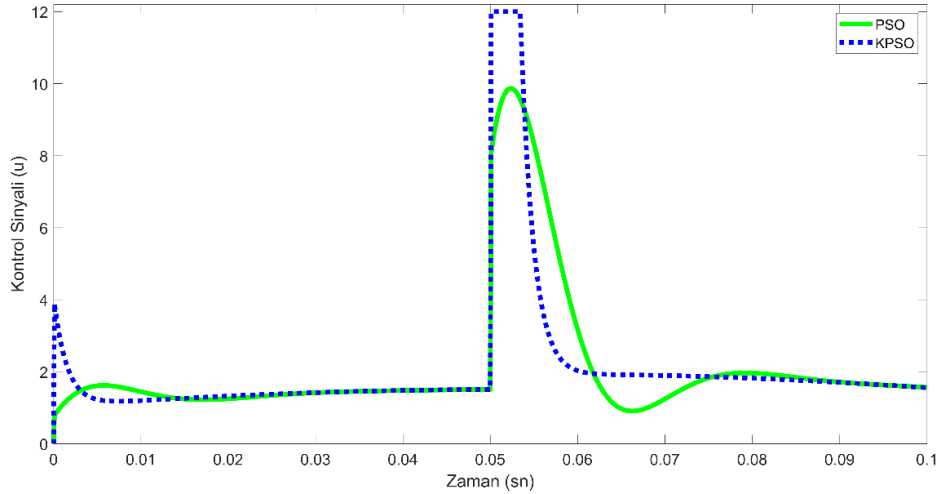
Şekil 12’de, DC motor sistemine uygulanan çeyrek periyotluk sinüzoidal referans sinyali altında, 0.05 saniyede bir bozucu etki uygulandığında PSO ve KPSO algoritmalarıyla optimize edilmiş PID kontrolcülerine ait hız yanıtları gösterilmektedir.



Şekil 12: Bozucu etki altında, optimize edilmiş PID kontrolcüler tarafından çeyrek periyotluk sinüzoidal referans sinyali için elde edilen sistem yanıtları

Şekil 12 incelendiği zaman, bozucu etkinin uygulandığı anda her iki kontrol yapısı da sistem yanıtında ani bir düşüş yaşamıştır. Ancak bu düşüşün büyüklüğü ve toparlanma süresi kontrolcü performansına göre farklılık göstermektedir. PSO tabanlı kontrolcü daha büyük bir sapma sergilerken, sistem cevabı referanstan önemli ölçüde uzaklaşmıştır. Buna karşılık, KPSO ile optimize edilen kontrolcü daha küçük bir sapma ile bozucu etkinin etkisini sönmülemiş ve referans sinyale daha hızlı şekilde geri dönmüştür.

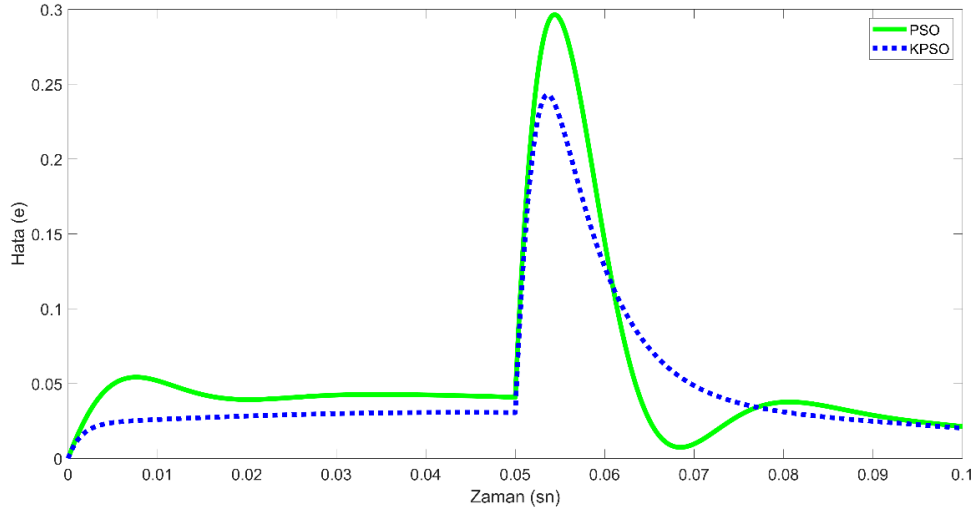
Şekil 13'te, DC motor sistemine çeyrek periyotluk sinüzoidal referans sinyali uygulanırken, 0.05 saniyede sisteme bir bozucu etki eklenmiş ve PSO ile KPSO algoritmalarıyla optimize edilen PID kontrolcülerinin ürettiği kontrol sinyalleri zamana bağlı olarak karşılaştırılmıştır.



Şekil 13: Bozucu etki altında, PSO ve KPSO algoritmalarıyla optimize edilmiş PID kontrolcüler tarafından çeyrek periyotluk sinüzoidal referans sinyali için üretilen kontrol sinyalleri

Şekil 13 incelendiğinde, uygulanan bozucu etki sonrasında kontrol sinyallerinde ani ve yüksek genlikli tepkiler oluşmuştur. Bu noktada, KPSO algoritmasıyla optimize edilen kontrolcü, maksimum kontrol sinyaline daha hızlı ulaşarak bozucu etkiyi daha agresif şekilde bastırma eğilimindedir. PSO ile optimize edilen kontrolcü ise daha yumuşak ancak daha gecikmeli bir tepki üretmiştir. Zaman ilerledikçe, KPSO algoritmasıyla optimize edilen kontrolcünün kontrol sinyali daha kısa sürede kararlı hale gelirken; PSO algoritmasıyla optimize edilen kontrolcü, bozucu etki sonrası salınımlar sergileyerek sistemin yeniden kararlılığa ulaşmasını geciktirmiştir.

Şekil 14'te, PSO ile KPSO algoritmalarıyla optimize edilen PID kontrolcülerine ait hata sinyalleri zamana bağlı olarak karşılaştırılmıştır.



Şekil 14: Bozucu etki altında, PSO ve KPSO algoritmaları ile ayarlanmış PID kontrolcüler tarafından çeyrek periyotluk sinüzoidal referans sinyali için elde edilen hata sinyalleri

Şekil 14 incelendiği zaman, bozucu etki uygulandığı anda her iki kontrol yapısında da hata seviyesinde ani bir artış yaşanmıştır. Bununla birlikte, PSO algoritması ile optimize edilen kontrolcüde hata büyüklüğü yaklaşık 0.29 seviyelerine kadar yükselmiş ve ardından dalgalı bir geçici rejim oluşmuştur. KPSO algoritması ile elde edilen kontrolcü ise bozucu etkinin etkisini daha düşük bir hata tepkisi ile karşılamış, yaklaşık 0.24 seviyelerinde tepe yaparak daha kısa sürede ve daha düzgün bir geçişle hatayı bastırmıştır. Oluşan hata sinyallerine ait Tip-1 norm değerleri incelendiğinde, PSO için 56.6999, KPSO için ise 48.9492 olarak hesaplanmıştır.

4 Tartışma

Bu çalışmada, PSO ve KPSO algoritmaları kullanılarak optimize edilen PID kontrolcülerinin performansları karşılaştırmalı olarak incelenmiştir. Aynı optimizasyon parametrelerini kullanan KPSO'nun, klasik PSO'ya kıyasla daha düşük bir uygunluk değerine ulaşması, PSO algoritmasının kaotik sistemlerle entegrasyonunun amaç fonksiyonunun minimuma yakınsamasında daha etkili olduğunu ortaya koymaktadır. Ayrıca simülasyon sonuçlarından KPSO'nun daha kısa yerleşme süresi sağladığı ve aşım oranını önemli ölçüde azalttığı görülmektedir. Aynı zamanda kontrol edilen sisteme bir bozucu etki uygulandığında KPSO'nun PSO'ya kıyasla daha sağlam bir kontrolcü sağladığı görülmektedir. Bu bulgular, KPSO'nun daha hızlı ve etkili bir şekilde optimal çözüme ulaştığını kanıtlamaktadır. KPSO algoritmasının yalnızca sabit referanslara değil, aynı zamanda zamanla değişen referanslara karşı da daha uygun bir kontrol davranışı sergileyebildiği gösterilmektedir.

KPSO'nun klasik PSO'ya göre PID kontrolcü parametrelerinin optimizasyonunda daha üstün bir performans sergilemesi, kaotik sistemlerin optimizasyon süreçlerine entegrasyonunun önemli faydalar sunduğunu göstermektedir. KPSO'nun kaotik yapısı, algoritmanın optimum çözüme daha etkin bir biçimde ulaşmasını sağlamaktadır. Elde edilen sonuçlar, hassas kontrol gerektiren uygulamalarda KPSO'nun önemli avantajlar sunduğunu ve daha etkili kontrol stratejilerinin geliştirilmesine olanak tanıdığını ortaya koymaktadır. KPSO'nun sağladığı iyileştirmeler, bu alanda yapılacak gelecekteki araştırmalara değerli katkılar sağlaması beklenmektedir.

5 Beyanname

5.1 Rakip Çıkarlar

Bu çalışmada herhangi bir çıkar çatışması yoktur.

5.2 Yazarların Katkıları

Muhammed Salih SARIKAYA: Araştırmanın tasarımı ve uygulanması, literatür taraması, sonuçların analizi, makale yazımı.

Onur DEMİREL: Literatür taraması, makaleyi gözden geçirme ve düzenleme.

Sezgin KAÇAR: Düzenleme ve denetim.

Adnan DERDİYOK: Düzenleme ve denetim.

Kaynakça

- [1] Zhong, F. (2023). Dynamic parameter identification based on improved particle swarm optimization and comprehensive excitation trajectory for 6R robotic arm. *Industrial Robot: The International Journal of Robotics Research and Application*, 51(1), 148–166. <https://doi.org/10.1108/ir-07-2023-0157>
- [2] Abedinifar, M., Ertuğrul, Ş., & Arguz, S. (2022). Nonlinear model identification and statistical verification using experimental data with a case study of the UR5 manipulator joint parameters. *Robotica*, 41(4), 1348–1370. <https://doi.org/10.1017/s0263574722001783>
- [3] Cosar, M. (2023). Path planning via swarm intelligence algorithms in unmanned aerial vehicle population. *The Eurasia Proceedings of Science Technology Engineering and Mathematics*, 26, 439–450. <https://doi.org/10.55549/epstem.1411059>
- [4] Wang, D., Liu, L., Ben, Y., Dai, P., & Wang, J. (2023). Seabed terrain-aided navigation algorithm based on combining artificial bee colony and particle swarm optimization. *Applied Sciences*, 13(2), 1166. <https://doi.org/10.3390/app13021166>
- [5] Wen, S., Liu, P., Wang, D., & Cao, F. (2014). Optimal tracking control for a Peltier refrigeration system based on PSO. *In Proceedings of the 2014 International Conference on Advanced Mechatronic Systems* (pp. 567–571). <https://doi.org/10.1109/icamechs.2014.6911610>
- [6] Tahtawi, A., Putri, F., & Martin, M. (2023). Position control of AX-12 servo motor using proportional-integral-derivative controller with particle swarm optimization for robotic manipulator application. *IAES International Journal of Robotics and Automation (IJRA)*, 12(2), 184–191. <https://doi.org/10.11591/ijra.v12i2.pp184-191>
- [7] Yu, Y., Xu, Y., Wang, F., Li, W., Mai, X., & Wu, H. (2020). Adsorption control of a pipeline robot based on improved PSO algorithm. *Complex & Intelligent Systems*, 7(4), 1797–1803. <https://doi.org/10.1007/s40747-020-00190-z>
- [8] Anshory, I., Hadidjaja, D., & Sulistiyowati, I. (2021). Measurement, modeling, and optimization speed control of BLDC motor using fuzzy-PSO based algorithm. *Journal of Electrical Technology UMY*, 5(1), 17–25. <https://doi.org/10.18196/jet.v5i1.12113>
- [9] Moghaddas, M., Dastranj, M., Changizi, N., & Rouhani, M. (2010). PID control of DC motor using particle swarm optimization (PSO) algorithm. *Journal of Mathematics and Computer Science*, 1(4), 386–391. <https://doi.org/10.22436/jmcs.001.04.16>

- [10] Zhu, X., Li, S., Cheng, H., & Fan, Z. (2022). Application of particle swarm optimization algorithm in photovoltaic constant pressure water supply electrical control system. *Journal of Physics: Conference Series*, 2203(1), 012061. <https://doi.org/10.1088/1742-6596/2203/1/012061>
- [11] Li, Q., & Ma, Z. (2021). A hybrid dynamic probability mutation particle swarm optimization for engineering structure design. *Mobile Information Systems*, 2021, 1–32. <https://doi.org/10.1155/2021/6648650>
- [12] Dong, Z. (2023). Crop disease and pest identification technology based on ACP SO-SVM algorithm optimization. *Engenharia Agrícola*, 43(5). <https://doi.org/10.1590/1809-4430-eng.agric.v43n5e20230104/2023>
- [13] Huang, J., Yang, J., Xie, D., & Wu, D. (2019). Optimal sliding mode chaos control of direct-drive wave power converter. *IEEE Access*, 7, 90922–90930. <https://doi.org/10.1109/access.2019.2925470>
- [14] Cheng, S. (2024). Synchronous control of high-speed train lift wing angle of attack drive system based on chaotic particle swarm optimization and linear auto-disturbance resistant controller. *Electronics*, 13(8), 1448. <https://doi.org/10.3390/electronics13081448>
- [15] Yue, Y., Cao, L., Hu, J., Cai, S., Hang, B., & Wu, H. (2019). A novel hybrid location algorithm based on chaotic particle swarm optimization for mobile position estimation. *IEEE Access*, 7, 58541–58552. <https://doi.org/10.1109/access.2019.2914924>
- [16] Jin, M. (2024). Method for the trajectory tracking control of unmanned ground vehicles based on chaotic particle swarm optimization and model predictive control. *Symmetry*, 16(6), 708. <https://doi.org/10.3390/sym16060708>
- [17] Huang, C., Naghdy, F., & Du, H. (2019). Fault tolerant sliding mode predictive control for uncertain steer-by-wire system. *IEEE Transactions on Cybernetics*, 49(1), 261–272. <https://doi.org/10.1109/tcyb.2017.2771497>
- [18] Chu, H., Yi, J., & Yang, F. (2022). Chaos particle swarm optimization enhancement algorithm for UAV safe path planning. *Applied Sciences*, 12(18), 8977. <https://doi.org/10.3390/app12188977>
- [19] Abdullah, H. (2021). An improvement in LQR controller design based on modified chaotic particle swarm optimization and model order reduction. *International Journal of Intelligent Engineering and Systems*, 14(1), 157–168. <https://doi.org/10.22266/ijies2021.0228.16>
- [20] Zhu, H. (2024). Optimizing active disturbance rejection control for a stubble breaking and obstacle avoiding control system. *Agriculture*, 14(5), 786. <https://doi.org/10.3390/agriculture14050786>
- [21] Mijbas, A., Hasan, B., & Salah, H. (2020). Optimal stabilizer PID parameters tuned by chaotic particle swarm optimization for damping low frequency oscillations (LFO) for single machine infinite bus system (SMIB). *Journal of Electrical Engineering and Technology*, 15(4), 1577–1584. <https://doi.org/10.1007/s42835-020-00442-5>
- [22] Sarıkaya, M. S., Hamida El Naser, Y., Kaçar, S., Yazıcı, İ., & Derdiyok, A. (2024). Chaotic-based improved Henry gas solubility optimization algorithm: Application to electric motor control. *Symmetry*, 16(11), 1435. <https://doi.org/10.3390/sym16111435>
- [23] Awrejcewicz, J., & Mrozowski, J. (1989). Bifurcations and chaos of a particular van der Pol-Duffing oscillator. *Journal of Sound and Vibration*, 132(1), 89–100. [https://doi.org/10.1016/0022-460X\(89\)90873-0](https://doi.org/10.1016/0022-460X(89)90873-0)
- [24] Sarıkaya, M. S., Demirel, O., Kaçar, S., & Derdiyok, A. (2025). Modelling and chaotic based parameter optimization of sliding mode controller. *Journal of Mathematical Sciences and Modelling*, 8(2), 42–55. <https://doi.org/10.33187/jmsm.1617412>



© 2020 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution 4.0 International (CC BY 4.0) license (<https://creativecommons.org/licenses/by/4.0/>)