



Farklı Yük Çeşitleri İçin Parçacık Sürü Optimizasyonu ve Ziegler-Nichols Metodunun DC Motor Hız Kontrolü Probleminde Karşılaştırılması

Celal Onur Gökçe^{1*}, Volkan Durusu², Rıdvan Ünal³

^{1*} Afyon Kocatepe Üniversitesi, Mühendislik Fakültesi, Elektrik Mühendisliği Bölümü, Afyonkarahisar, Türkiye, (ORCID: 0000-0003-3120-7808),

drcelalonurgokce@gmail.com

² Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi, Elektrik Elektronik Mühendisliği Bölümü, Afyonkarahisar, Türkiye (ORCID: 0000-0001-7726-5435),

durusuvolkan@gmail.com

³ Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi, Elektrik Elektronik Mühendisliği Bölümü, Afyonkarahisar, Türkiye (ORCID: 0000-0001-6842-7471),

runal@aku.edu.tr

(İlk Geliş Tarihi 13 Kasım 2021 ve Kabul Tarihi 3 Ocak 2021)

(DOI: 10.31590/ejosat.1022991)

ATIF/REFERENCE: Gökçe C.O., Durusu V., & Ünal R. (2022). Farklı Yük Çeşitleri İçin Parçacık Sürü Optimizasyonu ve Ziegler-Nichols Metodunun DC Motor Hız Kontrolü Probleminde Karşılaştırılması. *Avrupa Bilim ve Teknoloji Dergisi*, (33), 88-92.

Öz

Bu çalışmada DC motor hız kontrolü problemini Ziegler-Nichols (ZN) ve Parçacık Sürü Optimizasyonu (PSO) yaklaşımları ile çözerek performans karşılaştırılması yapılmıştır. Farklı referans çeşitleri ve farklı yük çeşitleri ile sistem sürülerek çıkış gözlemlenmiştir. Mutlak hataların toplamı cinsinden bir maliyet hesabı ve maliyet hesabının tersi olarak performans hesabı yapılmıştır. Bütün referans ve yüklerde PSO yaklaşımının performansı ZN yaklaşımının performansından iyi olduğu gözlemlenmiştir. Özellikle referans ve yük karmaşıklaştıkça bu performans farkının arttığı sonucuna varılır sonuçlar hem şekil hem de tablo olarak verilmiştir.

Anahtar Kelimeler: DC Motor, Parçacık Sürü Optimizasyonu, Ziegler-Nichols.

Comparison of Particle Swarm Optimization and Ziegler-Nichols Methods in the Problem of DC Motor Speed Control Under Different Loads

Abstract

In this study, DC motor velocity control problem is solved using two different approaches; namely Ziegler-Nichols (ZN) and Particle Swarm Optimization (PSO). The performances of two approaches are measured with different types of references and disturbances. Performance is measured using sum of absolute errors. It is observed that PSO approach shows better performance in all configurations. An increased performance difference is observed especially in complex references and complex disturbances. Results are given both in figures and as a table.

Keywords: DC Motor, Particle Swarm Optimization, Ziegler-Nichols.

* Sorumlu Yazar: drcelalonurgokce@gmail.com

1 Giriş

DC motor, endüstride ve piyasada en sık kullanılan motor türlerinden birisidir. DC motorun hız kontrolü, başta robotik, CNC ve otomasyon olmak üzere çeşitli uygulamaları olan bir problemdir. Boşta dönen bir DC motorun hızını kontrol etmek çok çalışılmış ve çözümü nispeten kolay bir problemdir. Gerçek uygulamalarda ise motora bir yük binmekte ve bu yük zamanla değişmektedir.

Zahir, Alhady, Othman ve Ahmad 2018 yılında yaptıkları çalışmada Fırçalı DC Motor PID kontrolünde PID parametrelerini belirlerken Genetik Algoritmayı (GA) kullanmış ve Ziegler-Nichols yönteminden daha iyi bir başarı gösterdiğini ortaya koymuştur. (A. A. M. Zahir, S. S. N. Alhady, W. A. F. W. Othman and M. F. Ahmad, 2018)

Yazgan, Yener, Soysal ve Gür 2019 yılından yayınlanan çalışmalarında Genetik Algoritma (GA) ve PSO ile PID parametrelerini optimize edip 5 kriterde karşılaştırmasını yapmıştır. Fırçasız DC Motor üzerinde gerçekleştirilen testlerde PSO'nun GA'dan daha iyi sonuç verdiğini tespit etmiştir. (Harun Yazgan, Furkan Yener, Semih Soysal, Ahmet Gür, 2019)

Song, Xiao ve Xu 2020 yılında yayınlanan çalışmalarında Fırçasız DC Motor için Bulanık (Fuzzy) PI kontrolör parametrelerinin optimizasyonu için PSO ve Yerçekimsel Arama Algoritmasını (YAA) birleştiren bir yaklaşım kullanmıştır. Bu iki yöntemi birleştirerek optimizasyon için başlangıçta kullanılan rastgele parametrelerin işlemlerinin daha hızlı sonuca ulaştığını gözlemlemiştir. (Baoye Song, Yihui Xiao, and Lin Xu, 2020)

Ibrahim, Hassan ve Shomer 2014 yılında yayınlanan çalışmalarında PID parametrelerinin belirlenmesinde PSO ve Bakteriyel Yiyecek Arama (Bacterial Foraging-BF) yöntemlerini kullanıp, yöntemlerin karşılaştırmasını yapmıştır. Çalışma Matlab ortamında simülasyon olarak gerçekleştirilmiştir. Kriterler göz önünde bulundurulduğunda gözler görülür bir iyileştirme olduğu ifade edilmiştir. (H.E.A.Ibrahima, F.N.Hassan, Anas O.Shomer, 2014)

Mohamadwasel ve Bayat 2019 yılında yaptıkları çalışmada PSO destekli Bulanık Mantık algoritması kullanarak PD kontrolör parametrelerini optimize etmiştir. (Noorulden Basil Mohamadwasel, Oguz Bayat, 2019)

Qi, Shi, ve Zhang 2019 yazdıkları makalede stokastik gecikmelere maruz bırakılan CAN tabanlı bir DC motorun PID parametrelerini PSO kullanarak optimize etmek için önceki verilere dayanarak simülasyonunu gerçekleştirmiş ve deneysel sonuçlar ile algoritmanın sonuçlarını doğrulamıştır. (Zhi Qi, Qian Shi, and Hui Zhang, 2019)

Weerasooriya ve El-Sharkawi 1991 yılında geri yayılım tabanlı yapay sinir ağları kullanarak DC Motorun lineer olmayan dinamiğini kontrol etmeyi amaçlamıştır. (Siri Weerasooriya M. A. El-Sharkawi, 1991)

Khalilpour, Razmjoooy, Hosseini ve Moallem 2011 yılında yaptıkları çalışmada parametrelerin optimizasyonu konusunda İstilacı Ot Optimizasyonu (IOO) ile Parçacık Sürü Optimizasyonu (PSO) yöntemlerinden hangisinin daha verimli olduğunu bulmak için karşılaştırma yapmıştır. Uygulama Matlab ortamında

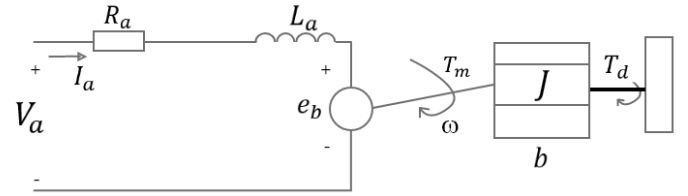
gerçekleştirilmiştir. (Optimal Control of DC motor using Invasive Weed Optimization (IWO) Algorithm, 2011)

Bu çalışmamızda, iki tür yük ele alınmıştır. Birincisi, basamak şeklinde binen yük. İkincisi ise belirli bir frekansa sahip sinusoidal olarak değişen yük. Bu iki yük çeşidinin büyüklükleri de performansa etki etmektedir. Farklı büyüklükler için deneyler yaparak parçacık sürü optimizasyonu ile optimize edilmiş PID kontrolcüsünün standart Ziegler-Nichols metodu ile katsayıları bulunmuş PID kontrolcüsünden daha yüksek performans sergilediği gösterilmiştir.

1.1 DC Motor Matematiksel Modeli

DC motorun modeli Şekil 1'de gösterilmiştir.

Şekil 1. DC Motor Modeli (Figure 1: DC Motor Model)



Burada V_a ; armatür voltajı, i_a ; armatür akımı, R_a ; armatür resistansı, L_a ; armatür indüktansı, e_b ; ters indüklenme voltajını ifade etmektedir. T_m ; motorun oluşturduğu torku, w ; rotorun açısal hızını, J ; rotorun eylemsizlik momentini, b ; yapışkanlık sürtünme katsayısını ve son olarak da T_d ; dış bozucu yük torkunu ifade etmektedir. Elektriksel kısmının matematiksel modelini, Kirchhoff'un voltaj yasasından bulabiliriz:

$$V_a = R_a i_a + L_a \frac{di_a}{dt} + e_b \quad (1)$$

Manyetik alan içerisinde hareket eden bir telde, hareket hızıyla orantılı büyüklükte ters elektromotif voltaj indüklenir prensibine göre:

$$e_b = K_b w \quad (2)$$

Manyetik alan içerisinde üzerinden akım geçen bir tele, akımın büyüklüğüyle orantılı bir kuvvet uygulanır prensibine göre:

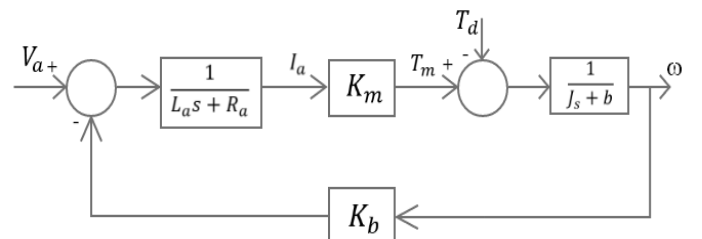
$$T_m = K_m i_a \quad (3)$$

Mekanik kısmını ise Newton'un ikinci hareket yasasına göre modelleyebiliriz:

$$T_m - T_d = J \frac{dw}{dt} + bw \quad (4)$$

Bu denklemlere göre DC motorun blok şeması Şekil 2'deki gibidir:

Şekil 2. DC Motor Blok Şeması (Figure 2: DC Motor Block Diagram)

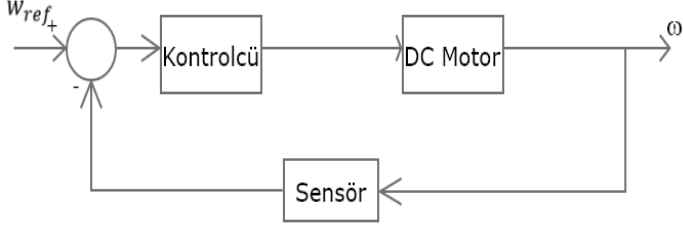


DC motorda; V_a giriş, w çıkıştır. Dolayısıyla istenilen w değerleri için, gerçek w bir hız sensörü ile ölçülecek ve gereken V_a giriş olarak verilerek motorun hızı kontrol edilecektir.

1.2 DC Motor Kontrol Sistemi

Kontrolcü ile beraber tüm sistemin blok şeması Şekil 3'te görüldüğü gibidir:

Şekil 3. Sistemin Blok Şeması (Figure 3: System Block Diagram)



w_{ref} , referans hız olup motorun dönmesi istenilen hızdır. w , motorun gerçek hızı olup bir sensör ile gerçek zamanlı olarak ölçülmektedir. Kontrolcü, referans hız ile ölçülen hızı karşılaştırıp aradaki hata sinyaline göre bir kontrol sinyali üretip motora vermekte, bu şekilde motorun hızının istenilen hızda tutulmasına çalışılmaktadır.

Kontrolcü olarak, PID kontrol kullanılmaktadır. Hata sinyalini e ile gösterirsek kontrolcünün çıkışı:

$$V_a = K_p e + K_i \int e dt + K_d \frac{de}{dt} \quad (5)$$

İle ifade edilmektedir.

Burada önemli olan K_p , K_i , K_d sabit katsayılarının en iyi performans gösterecek şekilde seçilmesidir. Bu çalışmamızda, PID kontrolcü katsayılarını bulmak için iki yöntem kullandık. Birincisi, otomatik kontrol kitaplarında klasik olarak anlatılan Ziegler-Nichols metodu. İkincisi, daha yeni ve çok popüler bir optimizasyon tekniği olan parçacık sürü optimizasyonu (PSO). Çalışmamızın özgünlüğü, farklı yük koşulları altında sistematik deneyler yaparak iki yöntemin performansını karşılaştırmaktır.

2. Materyal ve Metot

2.1 Ziegler-Nichols Metodu

PID parametrelerinin sisteme uygun olarak belirlenmesi çok önemlidir. PID parametrelerinin (K_p , K_i , K_d) belirlenmesinde çok kullanılan yöntemlerden birisi Ziegler-Nichols yöntemidir. Ziegler-Nichols yöntemi 1942 yılında John G. Ziegler ve Nathaniel B. Nichols tarafından bulunmuştur. Bu yöntemle göre kapalı çevrim bir kontrol sisteminde öncelikle K_i ve K_d katsayıları 0 alınarak K_p katsayısı için değişik değerler kullanılarak sistemin çıktısının salınım yaptığı duruma gelmesi sağlanır. Bulunan bu K_p değeri kritik değer olarak adlandırılır. Bu değer ile Ziegler ve Nichols tarafından oluşturulan tablodan Tablo 1'de görüldüğü gibi K_i ve K_d parametreleri hesaplanır. (PID Kontrol Parametrelerinin Ayarlanması, 2018)

Tablo 1. Ziegler-Nichols Parametre Hesaplama Tablosu (Table 1: Table of Parameter Calculation with Ziegler-Nichols) (Tri Kuntoro Priyambodo, Agfianto Eko Putra, Andi Dharmawan, 2015)

Kontrolör Tipi	K_p	T_i	T_d
P	$0,5 * K_c$	0	0
PI	$0,45 * K_c$	$T_u / 1,2$	-
PID	$0,6 * K_c$	$T_u / 2$	$T_u / 8$

2.2 Parçacık Sürü Optimizasyonu

Analitik olarak çözülmesi zor veya imkansız optimizasyon problemlerinde çok başarılı sonuçlar almış bir teknik olan PSO, özellikle kontrol problemlerine uygulanması uygun olan bir tekniktir. (B Gökçe; YB Koca; Y Aslan; CO Gökçe, 2021)

PSO'nun temel mantığı, arama uzayında rasgele olarak başlatılan noktaları, o ana kadar en başarılı olan bölgeye doğru teşvik etmektir. Her parçacığın pozisyonu, optimize edilmek istenilen parametreleri içeren bir vektördür. Her parçacık için bu pozisyon vektörü, başlangıçtan itibaren her iterasyonda güncellenmekte ve sınırlı sayıda kabul edilebilir bir süre sonunda kabul edilebilir bir performansa yakınsaması beklenmektedir. Her parçacığın her iterasyonda bir pozisyonu ve bir sonraki iterasyondaki pozisyonunu belirleyecek olan bir hızı vardır. Her iterasyonda parçacığın bir önceki pozisyonu ile o an için hesaplanan hızı toplanarak yeni pozisyonu bulunmaktadır:

$$x_n = x_{n-1} + v_n \quad (6)$$

Burada x_n parçacığın pozisyonunu v_n ise hızını temsil etmektedir.

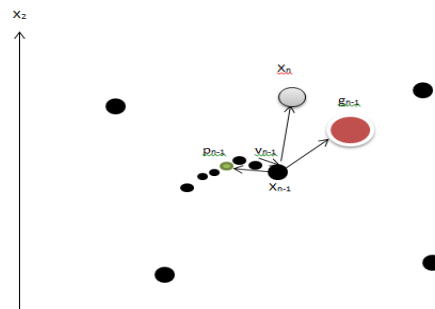
Burada kritik olan husus, v_n değerinin nasıl hesaplanacağıdır. v_n değerinin hesabında üç husus göz önünde bulundurulmaktadır:

- 1) Parçacığın momentumu, yani en son iterasyondaki hızı: v_{n-1}
- 2) Parçacığın o ana kadar bulduğu en iyi nokta pozisyonu: p_{n-1}
- 3) Tüm parçacıkların o ana kadar bulduğu en iyi noktanın pozisyonu: g_{n-1}

Bu üç değer belirli katsayılarla harmanlanarak, parçacığın yeni hızı Şekil 4'te görüldüğü gibi bulunmaktadır:

$$v_n = c_1 v_{n-1} + c_2 (x_{n-1} - p_{n-1}) + c_3 (x_{n-1} - g_{n-1}) \quad (7)$$

Şekil 4. PSO Çalışma Mekanizması (Figure 4: PSO Working Mechanism)



Bizim çalışmamızda, x_n vektörü, K_p , K_i , K_d 'yi içeren üç boyutlu bir vektördür. Dolayısıyla optimum parametreleri arama işlemi üç boyutlu uzayda yapılmaktadır.

Çalışmamızda simülasyon için Python programlama dili ve Spyder IDE'si kullanılmıştır. Kesikli zaman türevler, yamuk kuralına göre hesaplanmıştır. Performans ölçüm değeri olarak mutlak toplam hata alınmış, her noktada referans ile gerçek değer arasındaki farkın mutlak değeri alınarak toplanmış ve $1e6$ değerini sonuca bölerek bir performans değeri çıkarılmıştır.

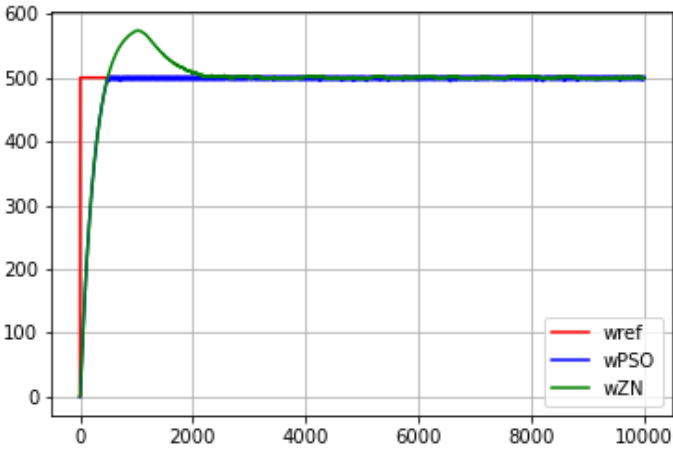
DC motor parametreleri olarak gerçekçi değerler kullanılmıştır. Yük olarak iki çeşit yük, basamak ve sinusoidal, çeşitli büyüklüklerde verilmiştir. Referans olarak da iki çeşit referans, basamak ve sinusoidal verilmiş ve sonuçlar aşağıda listelenmiştir.

3. Araştırma Sonuçları ve Tartışma

Her deneyde performans hesabı en basit haliyle hataların karelerinin toplamı şeklinde yapılmıştır. Önce boşta dönen motor için deneyler yapılmıştır.

Basamak girişi ile yüksüz dönen motor için PSO ve ZN çıkışları aşağıdaki şekilde görülmektedir.

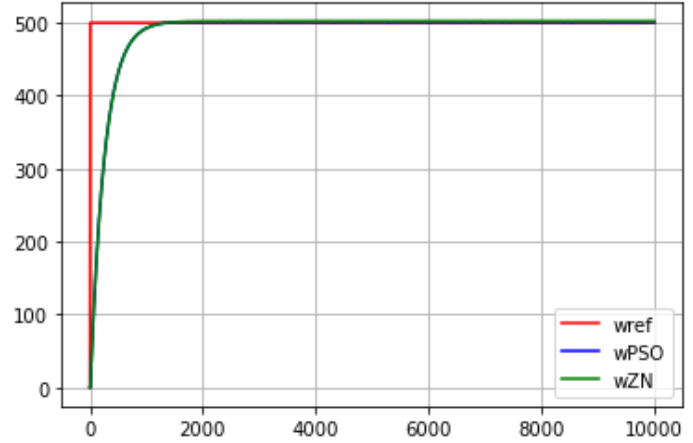
Şekil 5. Boşta Dönen Motora Basamak Referans Uygulanması (Figure 5: Applying a Step Reference to an Idling Motor)



Şekilde 5'te görüldüğü üzere, PSO parametreleri ile sistem referansı hızlı bir şekilde izlerken, ZN parametrelerinde aynı yükseliş zamanı fakat maksimum aşma ile izlemektedir. Bahsedilen performans hesaplayıcı ile hesaplanılan performanslar; PSO için 9.496, ZN için 6.316'dır. Yaklaşık bir buçuk katlık bir performans farkı dolayısıyla toplam hata farkı bulunmaktadır.

Sabit bir yük ile sistem test edilince, Şekil 6'da görüldüğü gibi ZN parametreleri ile yapılan kontrolcünün performansında küçük bir artış gözlemlenmiştir. Bu artışın sebebi, yük ile beraber sistemin maksimum aşma yapmamasıdır.

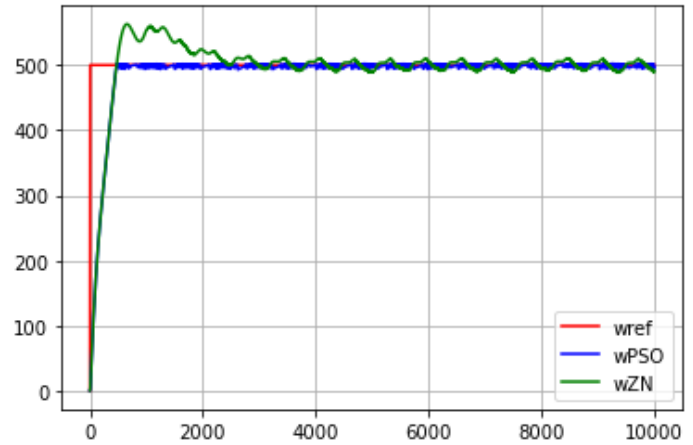
Şekil 6. Basamak Yük Uygulanan Motora Basamak Referans Verilmesi (Figure 6: Step Reference to a Step-Loaded Motor)



İki yaklaşım arasındaki performans farkının minimum olduğu bu durumda bile PSO yaklaşımı ZN yaklaşımına göre biraz daha üstündür. Hesaplanan performanslar; PSO için 7.464, ZN için 6.912'dir.

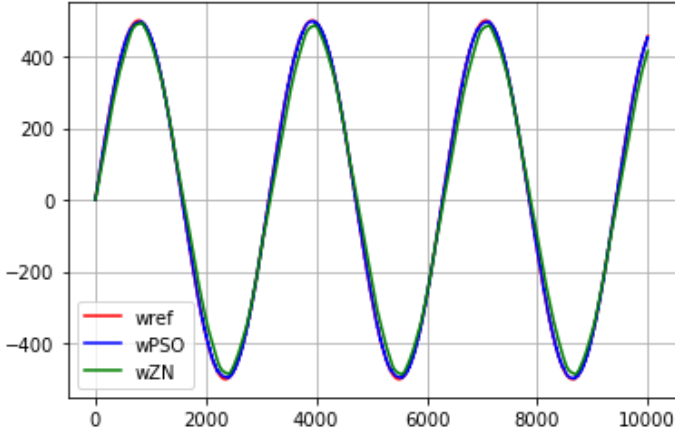
Üçüncü olarak sistem, basamak referans ve sinusoidal yük ile test edilmiş, sonuç aşağıdaki resimde görüldüğü gibi alınmıştır.

Şekil 7. Sinusoidal Yük Uygulanmış Motora Basamak Referans Verilmesi (Figure 7: Step Reference to a Sinusoidal Load Applied Motor)



Son olarak sistem, sinusoidal referans ve sinusoidal yük ile test edilmiş ve bu konfigürasyon ile PSO yaklaşımı ZN yaklaşımına göre performans farkını Şekil 7'de görüldüğü gibi oldukça artırmıştır. Hesaplanan performanslar; PSO için 8.379, ZN için 4.817'dir. İki katına yakın bir toplam hata farkı vardır.

Şekil 8. Sinüsoidal Yükte Dönen Motora Sinüsoidal Referans Verilmesi (Figure 8 : Sinusoidal Reference to a Rotating Motor on Sinusoidal Load)



Hesaplanan performanslar; PSO için 35.493, ZN için 4.193'tür. Şekil 8'de görüldüğü gibi ZN yaklaşımı referansı takip etmekte zorlanırken, PSO yaklaşımı ZN yaklaşımının yaklaşık onda biri hata ile referansı çok daha başarılı bir şekilde takip edebilmektedir.

Değişik yükler için PSO ve ZN metodlarının performansları tabloda sunulmuştur:

Tablo 2. Farklı yükler için PSO ve ZN performansları

Referans	Yük çeşidi	Büyükölük	PSO	ZN
Basamak	Yüksüz	-	9.496	6.316
Basamak	Basamak	0.01	8.883	5.877
Basamak	Basamak	0.03	7.725	4.992
Basamak	Basamak	0.05	7.464	6.912
Basamak	Sinüs	0.01	9.132	5.569
Basamak	Sinüs	0.03	8.529	4.662
Basamak	Sinüs	0.05	8.379	4.817
Sinüs	Sinüs	0.01	36.003	4.147
Sinüs	Sinüs	0.03	36.654	4.393
Sinüs	Sinüs	0.05	35.493	4.193

Şekil 8'de ve Tablo 2'de görüldüğü gibi, bütün referans çeşitleri ve bütün yük çeşit ve değerleri için PSO algoritması ZN metoduna kıyasla daha üstün bir performans sergilemiştir. Özellikle sinusoidal referans ve sinusoidal yükte bu performans farkı artmaktadır. Sinüsoidal referansta PSO algoritmasının yüksek başarı göstermesinin sebeplerinden birisi, referans değerinin daha yavaş değişip motorun referansı takip etmesindeki kolaylık olarak açıklanabilir. Öte yandan, yük çeşidi karmaşıklıkça, PSO algoritmasının daha iyi bozucu reddetme performansı gösterdiği söylenebilir. İleriki çalışmalarda, gerçek sistem ve pekiştirmeli öğrenme metodu kullanılarak performans karşılaştırılması yapılması hedeflenmektedir. Ayrıca; mobil robot, biped robot ve kuadrokopter gibi daha karmaşık sistemlerde de bu metodların

karşılaştırılması yapılarak literature kazandırılması hedeflenmektedir.

Kaynakça

- Optimal Control of DC motor using Invasive Weed Optimization (IWO) Algorithm. (2011). Majlesi Conference on Electrical Engineering.
- A. A. M. Zahir, S. S. N. Alhady, W. A. F. W. Othman and M. F. Ahmad. (2018). Genetic Algorithm Optimization of PID Controller for Brushed DC Motor. Intelligent Manufacturing & Mechatronics. Lecture Notes in Mechanical Engineering. Springer, Singapore, 427-437.
- B Gökçe; YB Koca; Y Aslan; CO Gökçe. (2021). PARTICLE SWARM OPTIMIZATION-BASED OPTIMAL PID CONTROL OF AN AGRICULTURAL MOBILE ROBOT. Comptes rendus de l' Academie bulgare des Sciences, 568-575.
- Baoye Song, Yihui Xiao, and Lin Xu. (2020). Design of fuzzy PI controller for brushless DC motor based on PSO-GSA algorithm. Systems Science & Control Engineering , 67-77.
- Gökbulut, P. D. (2019, Ocak). Kontrol Sistemlerinin Analiz ve Tasarımı. 10 25, 2021 tarihinde eem.tf.firat.edu.tr: http://eem.tf.firat.edu.tr/subdomain_files/eem.tf.firat.edu.tr/files/36/Kontrol%20sistemleri%20ders%20notu.pdf adresinden alındı
- H.E.A.Ibrahima, F.N.Hassan, Anas O.Shomer. (2014). Optimal PID control of a brushless DC motor using PSO and BF techniques. Ain Shams Engineering Journal, 2(5), 391-398.
- Harun Yazgan, Furkan Yener, Semih Soysal, Ahmet Gür. (2019). Comparison Performances of PSO and GA to Tuning PID Controller for the DC Motor. Sakarya University Journal of Science, 162-174. doi:10.16984/saufenbilder.376464
- Noorulden Basil Mohamadwasel, Oguz Bayat. (2019). Improve DC Motor System using Fuzzy Logic Control by Particle Swarm Optimization in Use Scale Factors. International Journal of Computer Science and Mobile Computing, 3(8), 152-160.
- PID Kontrol Parametrelerinin Ayarlanması. (2018, 03 9). https://acikders.ankara.edu.tr/pluginfile.php/70147/mod_resource/content/0/GDM404_11.pdf adresinden alındı
- Siri Weerasooriya M. A. El-Sharkawi. (1991). IDENTIFICATION AND CONTROL OF A DC MOTOR USING BACK-PROPAGATION NEURAL NETWORKS. IEEE Transactions on Energy Conversion, 6(4), 663-669.
- Tri Kuntoro Priyambodo, Agfianto Eko Putra, Andi Dharmawan. (2015). Optimizing control based on ant colony logic for Quadrotor stabilization. 2015 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES) (s. 1-4). IEEE.
- Zhi Qi, Qian Shi, and Hui Zhang. (2019). Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-based DC motor subject to stochastic delays. IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS. doi: DOI 10.1109/TIE.2019.2934030