

Kırıkkale Üniversitesi

EO TABANLI BULANIK MANTIK, EO TABANLI PID, PSO TABANLI PID VE RGA TABANLI PID İLE KONTROL EDİLEN DC MOTORUN PERFORMANS ANALİZİ

Hazırlayan: Nelson Luis Manuel

Birim/Bölüm: FBE/Elektrik-Elektronik Müh.(YL/Tezli)

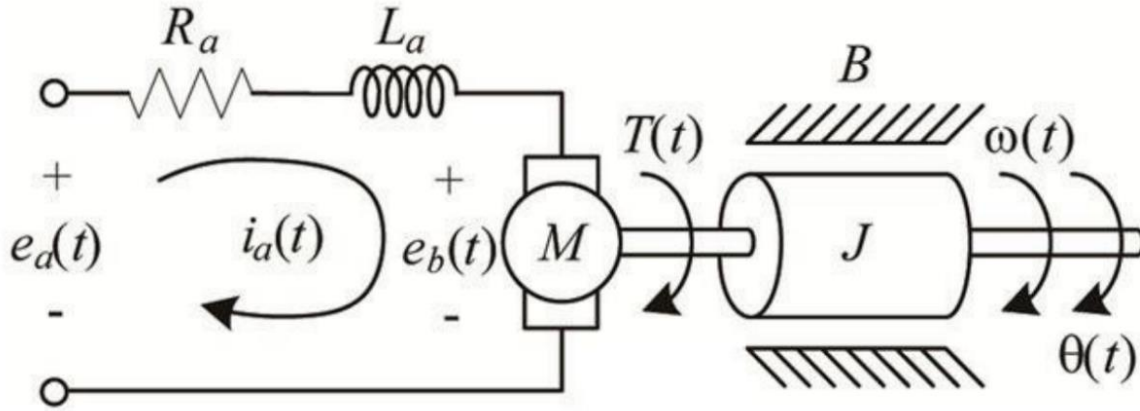
Ders: Optimizasyon

- ① DC Motor
- ② Amaç fonksiyonu
- ③ EO Tabanlı Bulanık Mantık Denetleyicisi
- ④ EO Tabanlı PID Denetleyicisi
- ⑤ PSO Tabanlı PID Denetleyicisi
- ⑥ RGA Tabanlı PID Denetleyicisi
- ⑦ Denetleyicilerin Performans Karşılaştırması
- ⑧ Sonuçlar
- ⑨ Kaynakça

1-DC Motor

1.1 DC Motorun Eşdeğer Devresi

Aşağıdaki şekil 1, DC motorun eşdeğer devresini temsil etmektedir.



Şekil 1: DC motorun eşdeğer devresi.

Tablo 1: DC motorun eşdeğer devresinin parametreleri

Parametreler	Açıklama	Kullanılacak değerler
$e_a(t)$	Armatür gerilimi	200 V
$i_a(t)$	Armatür akımı	*
$e_b(t)$	Geri emf	*
R_a	Armatür direnci	0.5 Ω
L_a	Armatür endüktansı	0.02 H
J	Atalet momenti	0.1 Kg. m ²
B	Vizkoz sürtünme	0.008 N. m. s/rad
K_t	Tork katsayısı	0.5 N. m/A
K_b	Geri emk katsayısı	1.25 V. s/rad
T_L	Yük torku	0.5 N. m
T_e	Motorun ürettiği tork	*
w	Açısal hız	*

* Bu değerler, bir sonraki slaytta denklemler (1-5) kullanılarak hesaplanır.

1-DC Motor

1.2 DC Motorun Matematiksel Modeli

Şekil 1'den, aşağıdaki DC motor denklemleri belirlenebilir:

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

$$e_b(t) = K_b \omega(t) \quad (2)$$

$$\dot{i}_a(t) = \frac{e_a(t)}{L_a} - \frac{K_b}{L_a} \omega(t) - \frac{R_a}{L_a} i_a(t) \quad (3)$$

$$T(t) = T_l + J \frac{d\omega(t)}{dt} + B\omega(t) \quad (4)$$

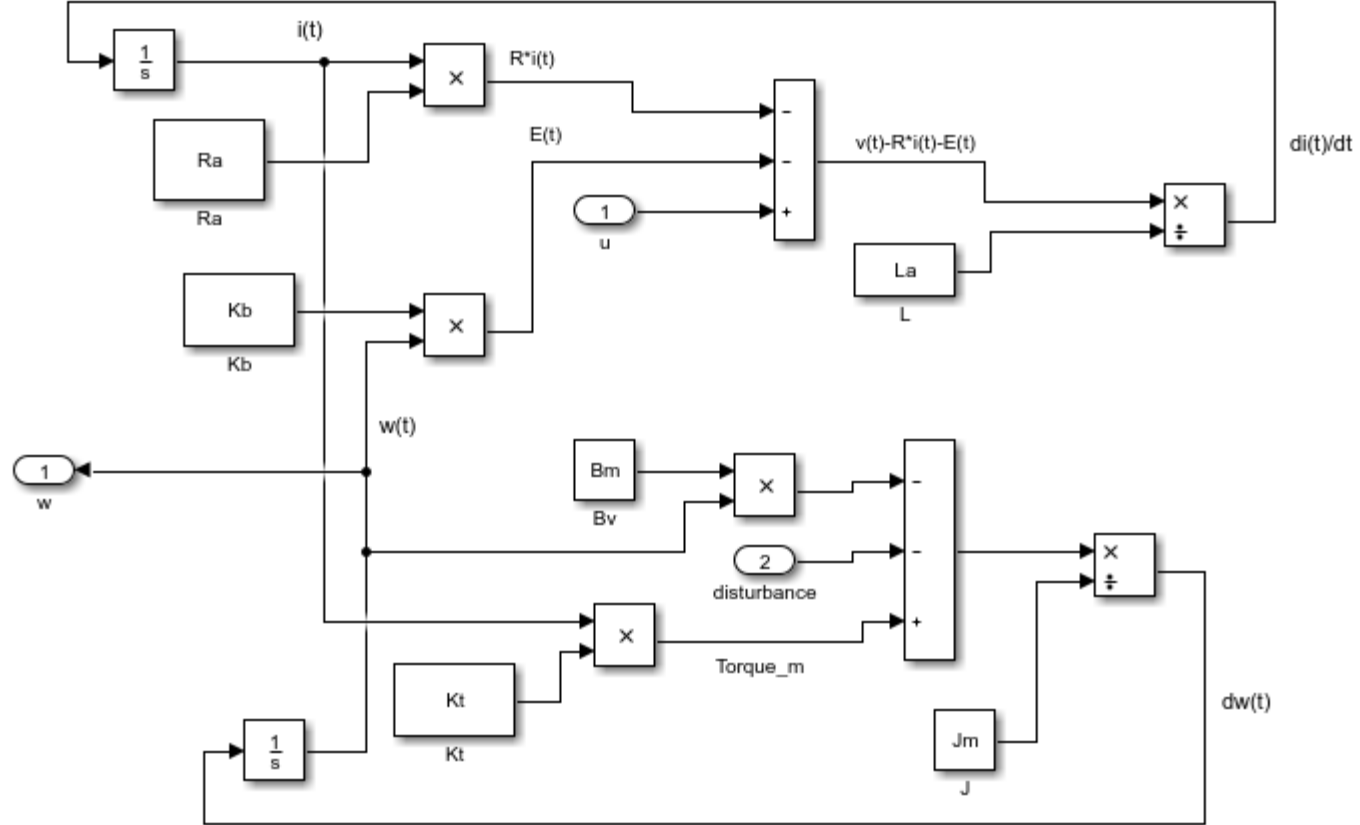
$$T(t) = K_t i_a(t) \quad (5)$$

(1-5) denklemlerine Laplace dönüşümü uygulayarak, çıkış (motor hızı) ve giriş (armatür voltajı) arasındaki ilişki elde edilebilir:

$$\frac{\omega(s)}{e_a(s)} = \frac{\frac{K_t}{J L_a}}{\left(\left(\left(s + \frac{R_a}{L_a} \right) \left(s + \frac{B}{J} \right) \right) + \frac{K_b K_t}{J L_a} \right)} \quad (6)$$

1-DC Motor

1.3 DC Motorun Simulink Modeli



Şekil 2: Matlab'de yapılan DC motorun Simulink modeli.

2-Amaç fonksiyonu

Literatürde, genellikle, güç sistemlerinin problemlerini optimize etmek için maliyet fonksiyonları olarak 4 performans indeksi kullanılmaktadır [2] :

Kare Hata İntegrali :

$$\text{ISE} = \int_0^{\infty} [e(t)]^2 dt \quad (7)$$

Mutlak hata değerinin integrali :

$$\text{IAE} = \int_0^{\infty} |e(t)| dt \quad (8)$$

Zaman ağırlıklı IAE:

$$\text{ITAE} = \int_0^{\infty} t|e(t)| dt \quad (9)$$

Zaman ağırlıklı ISE:

$$\text{ITSE} = \int_0^{\infty} t[e(t)]^2 dt \quad (10)$$

Simülasyonlar sırasında, öncelikle, tüm performans indeksleri test edildi ve ardından **ITAE** indeksi seçilmiştir. Bu çalışmada ITAE, PID ve Bulanık Mantık Denetleyicisi için optimal katsayıları bulmak için objektif fonksiyon olarak kullanılmıştır.



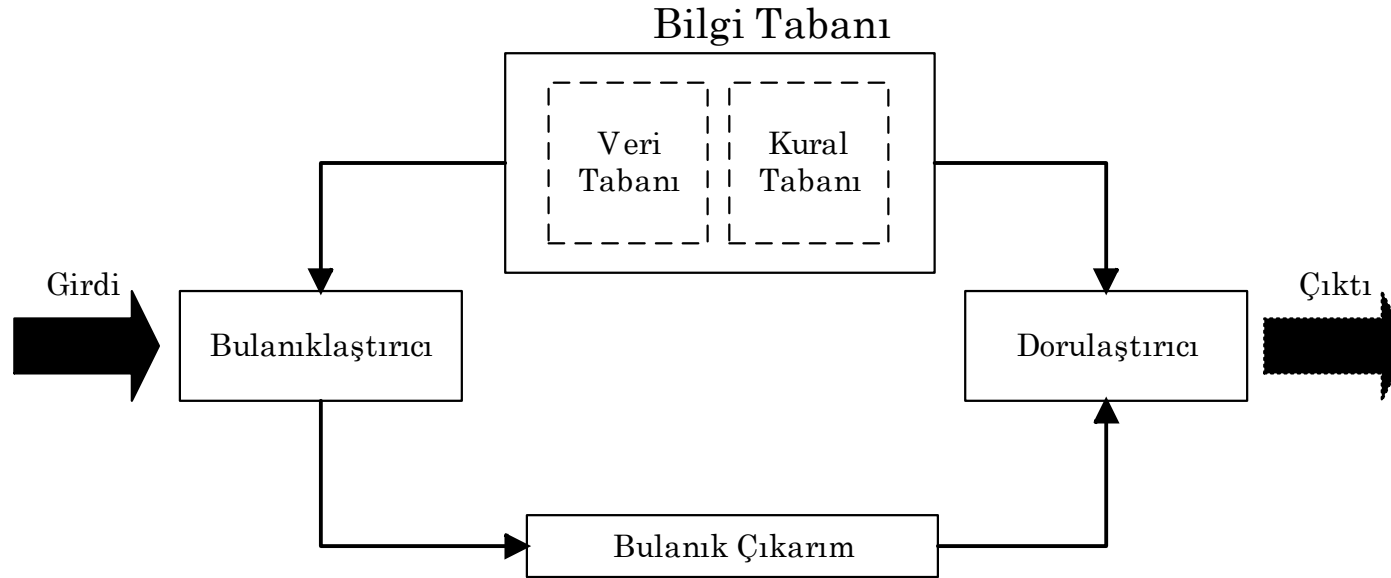
3-EO Tabanlı Bulanık Mantık Denetleyicisi

3.1 Bulanık Mantık Denetleyicisinin Yapısı

devam ediyor...

Bulanık kontrol sistemlerinin dayandığı temel nokta, uzman bir sistem operatörünün bilgi, deneyim, sezgi ve kontrol stratejisini, kontrolör tasarımında bilgi tabanı olarak oluşturmaktır.

Aşağıdaki şekil, bulanık mantık denetleyicisinin temel yapısını göstermektedir:



Şekil 3: Bulanık Mantık Denetleyici Yapısı.

3-Bulanık Mantık Denetleyicisi

3.1 Bulanık Mantık Denetleyicisinin Yapısı

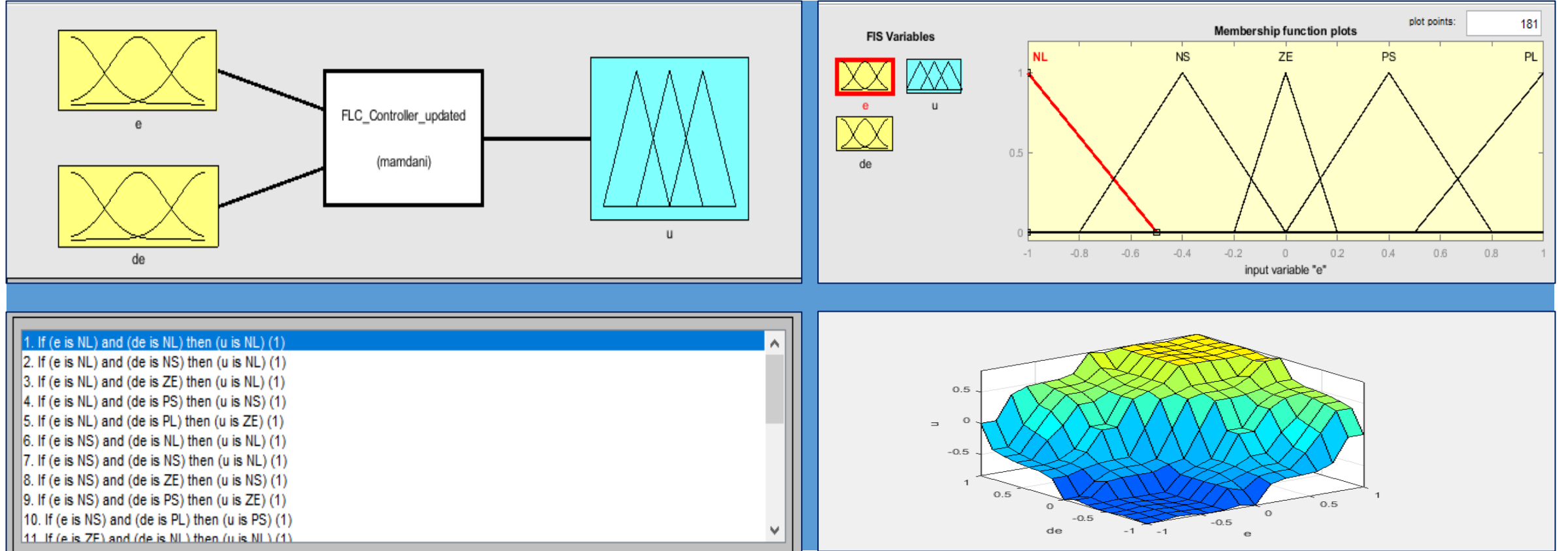
Önceki şekilden görüldüğü gibi, bulanık mantık denetleyici 4 ana kısımdan oluşur [11] :

- **Bulandırma birimi**, sistemden alınan giriş bilgilerini dilsel niteleyiciler olan sembolik değerlere dönüştürme işlemidir.
- **Bulanık çıkarım birimi**, bulandırma biriminden gelen bulanık değerleri, kural tabanındaki kurallar üzerinde uygulayarak bulanık sonuçlar üretilmektedir.
- **Durulama**, bulanık bilgilerin kesin sonuçlara dönüştürülmesi işlemidir.
- **Bilgi tabanı**, denetlenecek sistemle ilgili bilgilerin toplandığı bir veri tablosundan ibarettir. Girişler ve çıkışlar arasındaki bağlantılar, kural tabanındaki kurallar kullanılarak sağlanır.



3-Bulanık Mantık Denetleyicisi

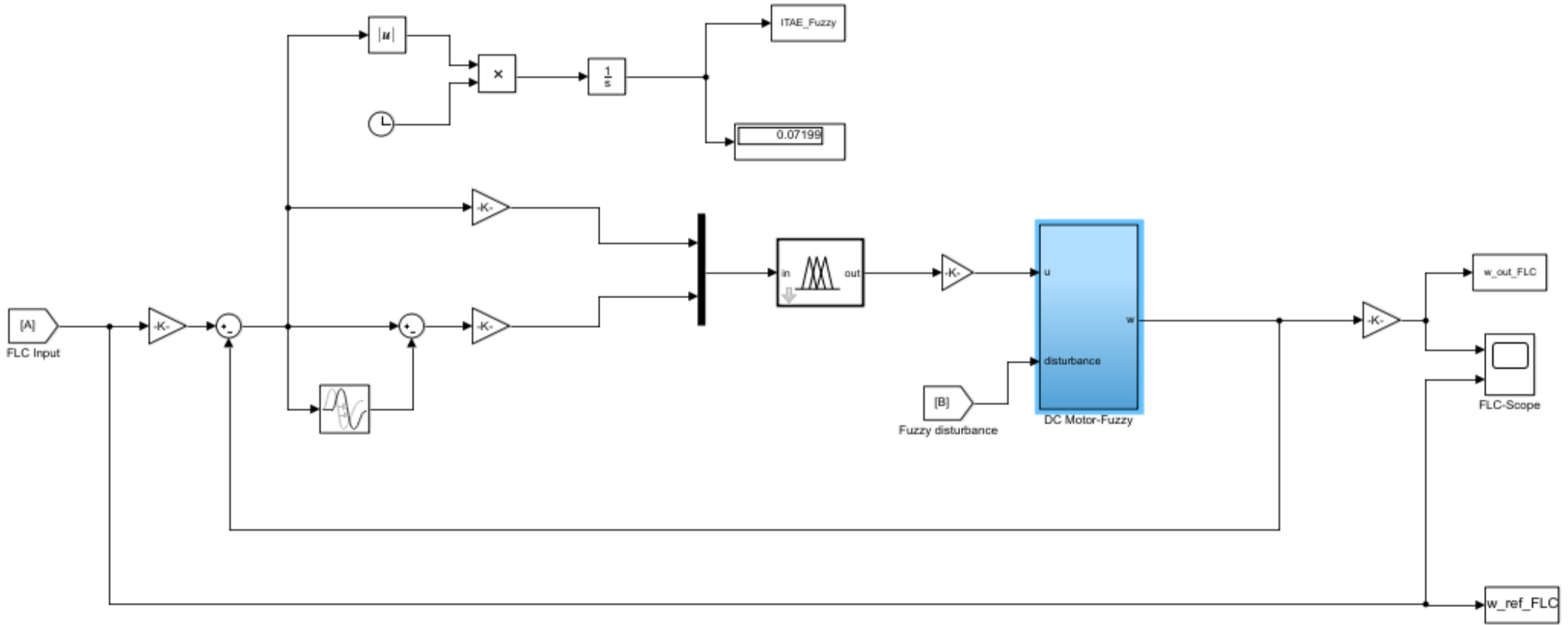
3.2 Bulanık Mantık Denetleyicisinin Ekran Görüntüleri



Şekil 4: Matlab'de yapılan FLC'un ekran görüntüleri.

3-Bulanık Mantık Denetleyicisi

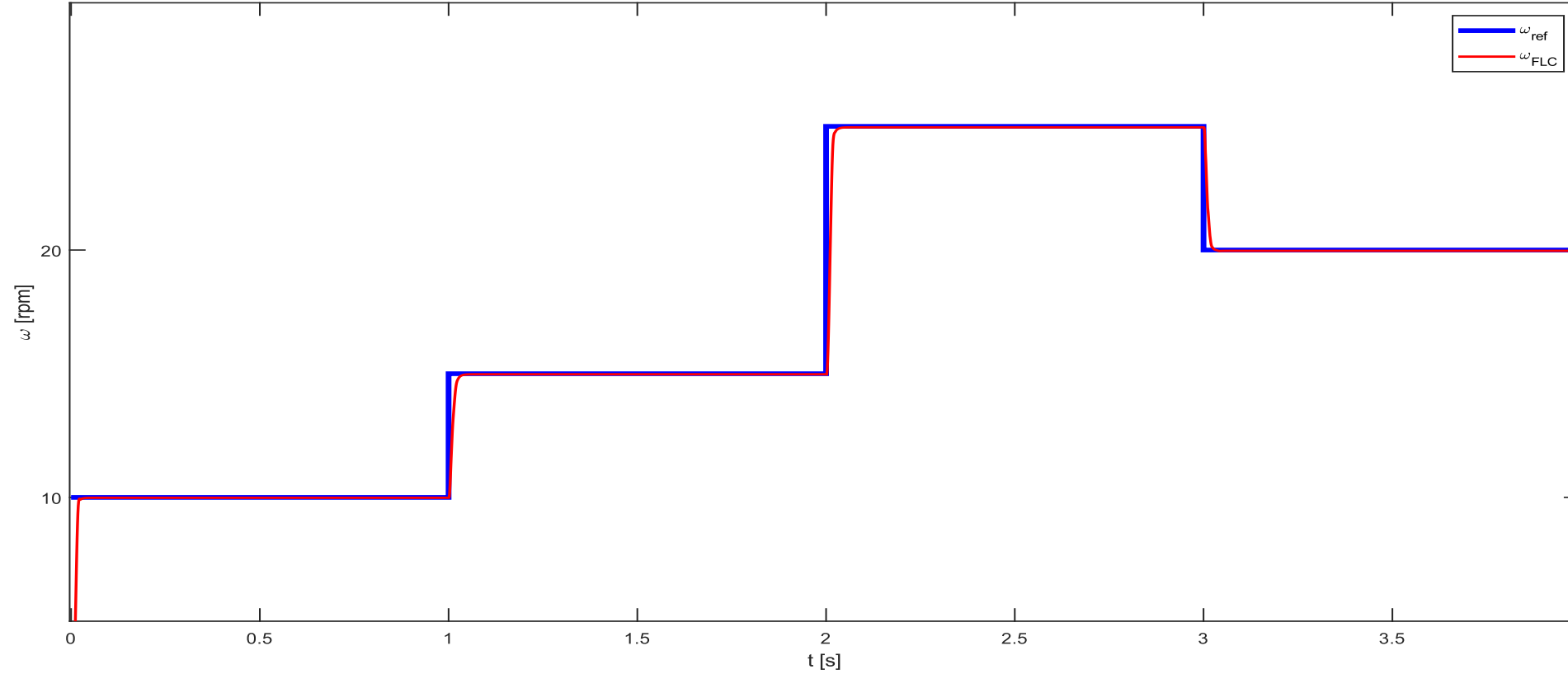
3.3 Simulink Modeli



Şekil 5: Simulink'te yapılan DC motor ve EO-FLC modeli.

3-Bulanık Mantık Denetleyicisi

3.4 EO-FLC'nin Performansı



Şekil 6: Değişken giriş altında EO – FLC tarafından kontrol edilen DC motorun performansı.



4-EO Tabanlı PID Denetleyicisi

4.1 Equilibrium Optimizer Algoritması

Equilibrium Optimizer (EO), 2019 yılında Faramarzi ve diğerleri tarafından yayınlanan yeni bir optimizasyon algoritmasıdır [7] .

EO, bir kontrol hacminin kütle dengesinden esinlenmiştir. Her arama ajanı (bir parçacık ve ilgili konsantrasyonu), optimum sonucu elde etmek için denge adaylarına (şimdiye kadarki en iyi çözümler) göre konsantrasyonunu günceller.

Aşağıdaki denklem, EO algoritmasının parçacık konsantrasyonunun güncelleme yasasını temsil etmektedir:

$$C = C_{eq} + (C_0 - C_{eq})F + \frac{G}{\lambda V}(1 - F) \quad (11)$$

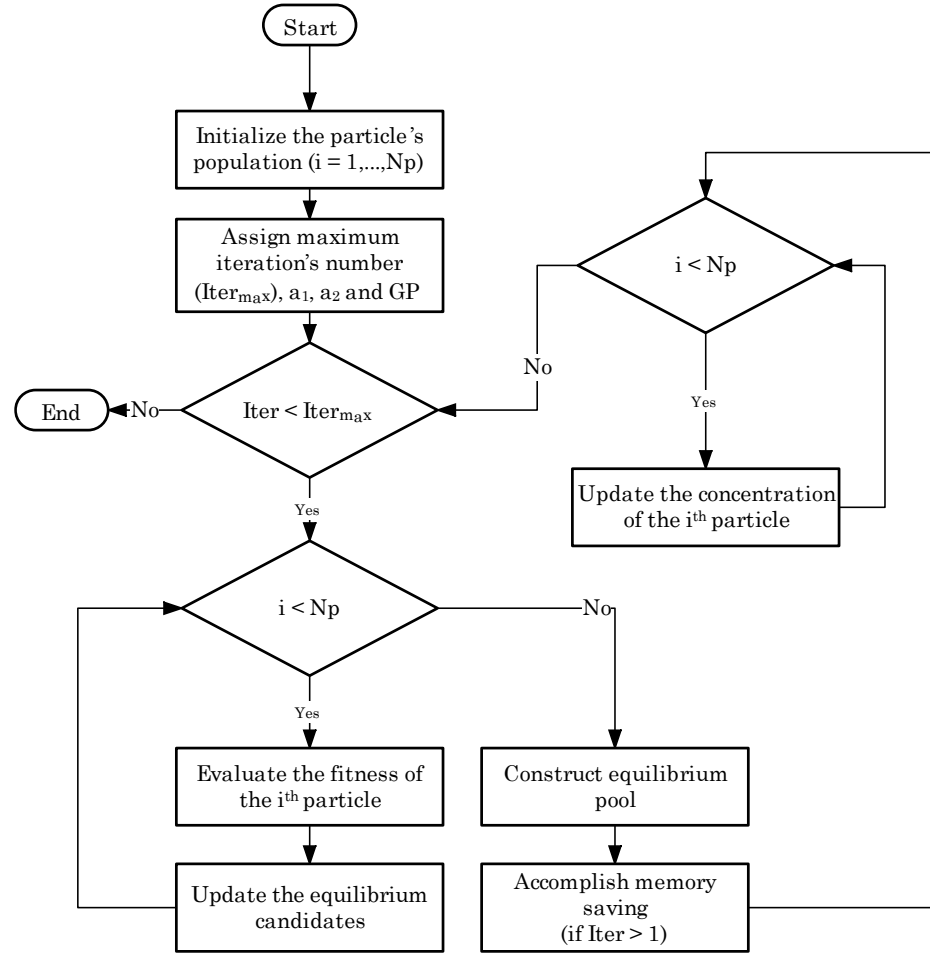
C kontrol hacmindeki konsantrasyondur; C_0 başlangıç konsantrasyonudur; C_{eq} denge adayının konsantrasyonunu temsil eder; G üretim hızıdır; F üstel terim olarak adlandırılır ve algoritmanın keşif ve kullanım süreçlerini dengelemeye yardımcı olur; λ tersi ikamet süresi ve V kontrol hacmidir.



4-EO Tabanlı PID Denetleyicisi

4.2 Sözde kod

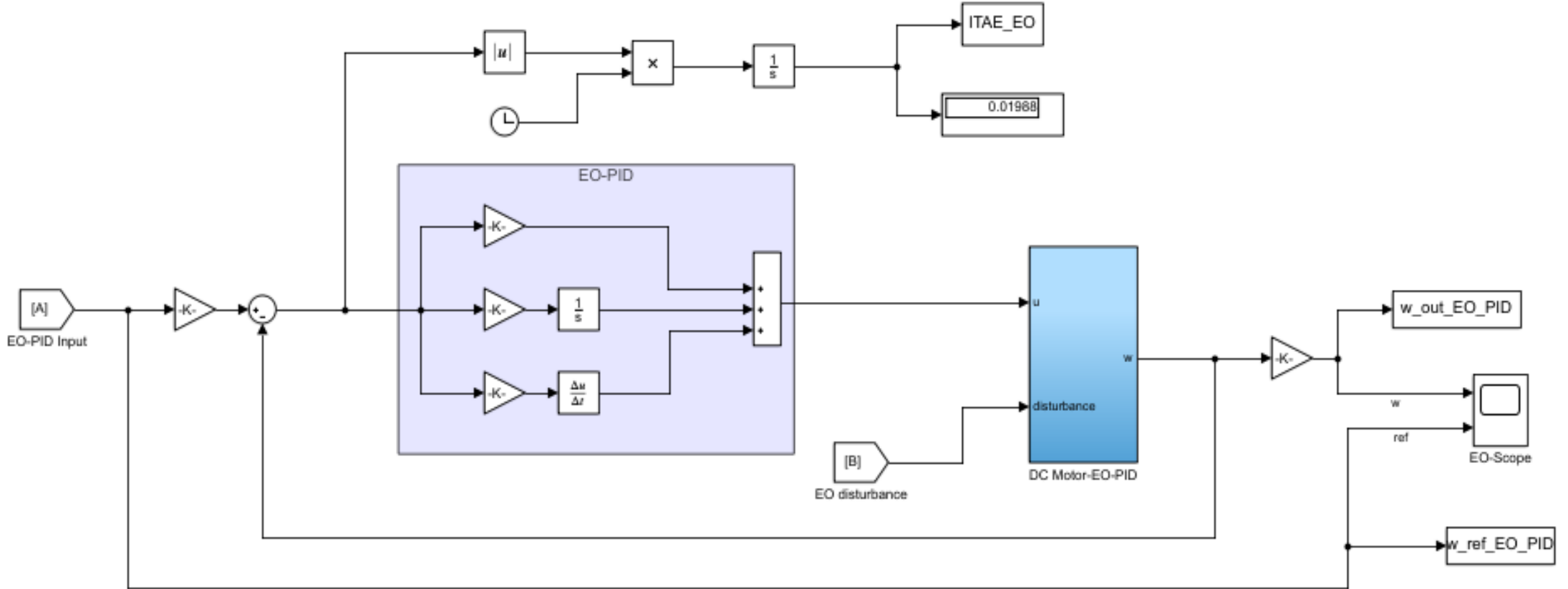
Aşağıdaki şekil, Equilibrium Optimizer algoritmasının sözde kodudur:



Şekil 7: EO Algoritmasının Sözde Kodu.

4-EO Tabanlı PID Denetleyicisi

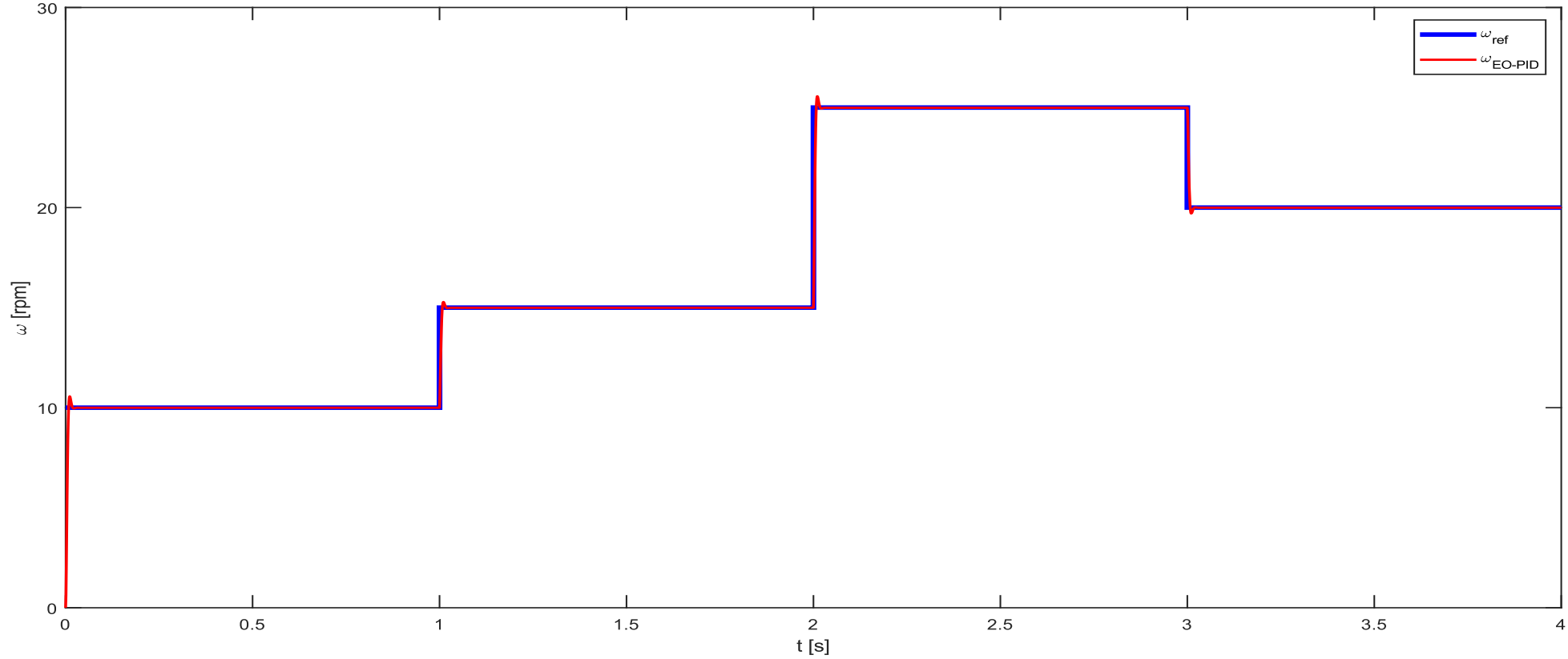
4.3 Simulink Modeli



Şekil 8: Simulink'te yapılan DC motor ve EO-PID modeli.

4-EO Tabanlı PID Denetleyicisi

4.4 EO-PID'nin Performansı



Şekil 9: Değişken giriş altında EO – PID tarafından kontrol edilen DC motorun performansı.



5-PSO Tabanlı PID Denetleyicisi

5.1 Parçacık Sürüsü Optimizasyon Algoritması

Parçacık Sürü Optimizasyonu (PSO), Kennedy ve Eberhart (1995) tarafından keşfedilen balıklar ile kuşların sürü şeklindeki davranışından esinlenen bir optimizasyon yöntemidir.

Tüm iterasyonlarda algoritma, evrende daha iyi konumlar bulmak için parçacıkların konumlarını güncellemeye çalışır. Yinelemelerle bir parçacığın konumunun değişimini temsil eden denklemler şu şekilde verilir [5]:

$$V_i^{k+1} = (w^{k+1} * V_{ij}^k) + c_1 \cdot r_1 (p_{ij}^k - x_{ij}^k) + c_2 \cdot r_2 (g_{ij} - x_{ij}^k) \quad (12)$$

$$x_{ij}^{k+1} = x_{ij}^k + v_{ij}^{k+1} \quad (13)$$

Denklem **12**, parçacığın hızını temsil eder. Denklem **13**, parçacığın konum değişimini temsil eder.

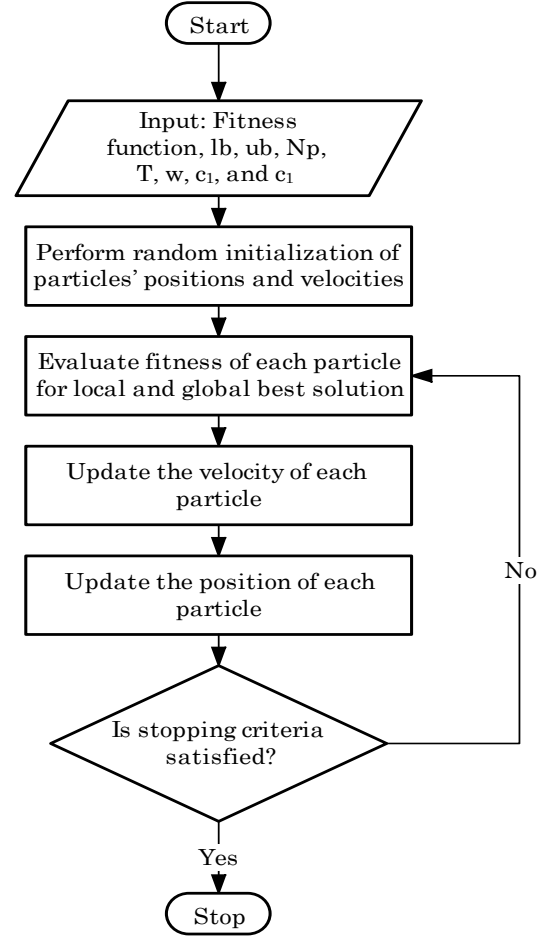
V_i , i'inci parçacığın hızıdır; w , parçacıkların ataletidir; c_1 ve c_2 ivme katsayılarıdır; r_1 ve r_2 rastgele sayılardır $\in [0,1]$; p , i'inci parçacığın yerel en iyisidir; g , global en iyisidir ve X_i , i'inci parçacığın konumudur.



5-PSO Tabanlı PID Denetleyicisi

5.2 Sözde kod

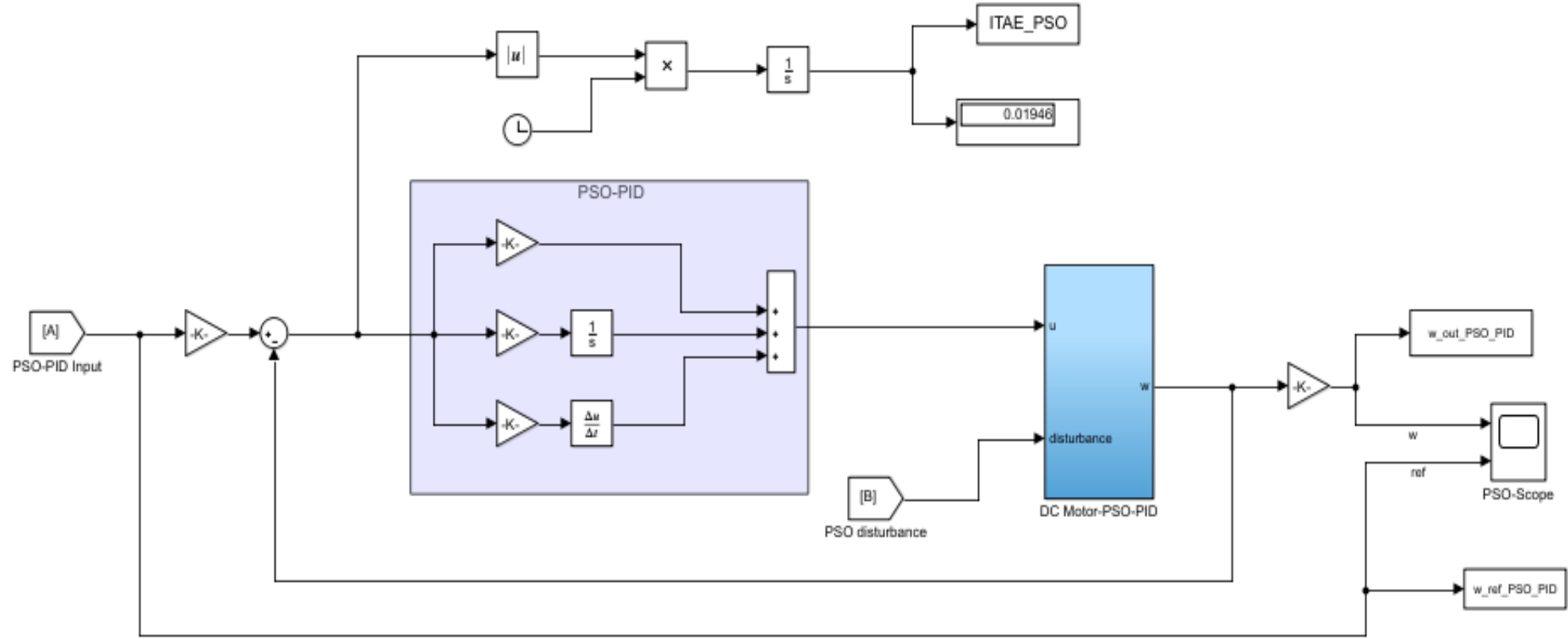
Aşağıdaki şekil, PSO algoritmasının sözde kodudur:



Şekil 10: PSO Algoritmasının Sözde Kodu.

5-PSO Tabanlı PID Denetleyicisi

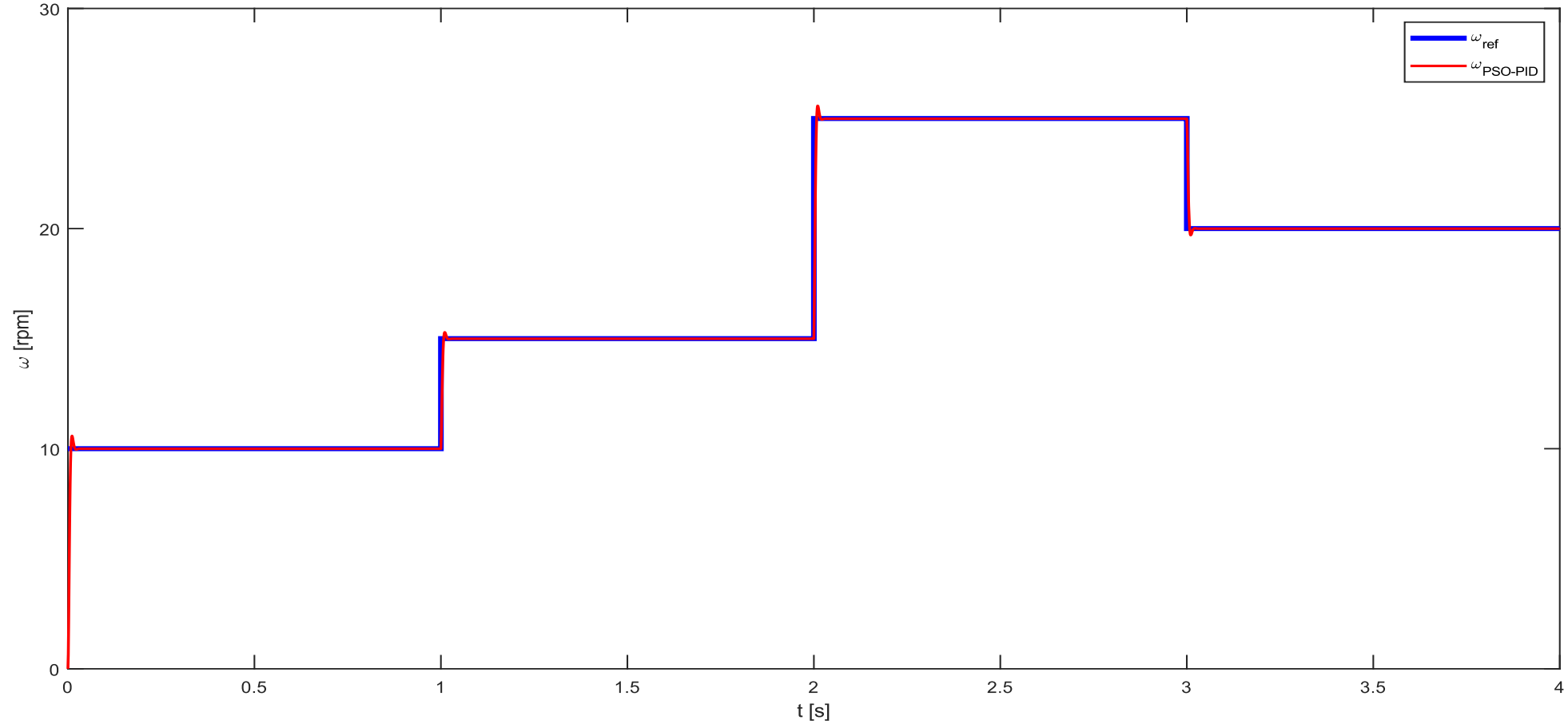
5.3 Simulink Modeli



Şekil 11: Simulink'te yapılan DC motor ve PSO-PID modeli.

5-PSO Tabanlı PID Denetleyicisi

5.4 PSO-PID'nin Performansı



Şekil 12: Değişken giriş altında PSO – PID tarafından kontrol edilen DC motorun performansı.



6-RGA Tabanlı PID Denetleyicisi

6.1 Genetik algoritma

Genetik Algoritmalar (GA), 1970'li yıllarda John H. Holland ve üniversitedeki çalışma arkadaşları tarafından geliştirilmeye başlanmış bir optimizasyon yöntemidir.

Genetik Algoritmanın çalışması şu şekilde özetlenebilir:

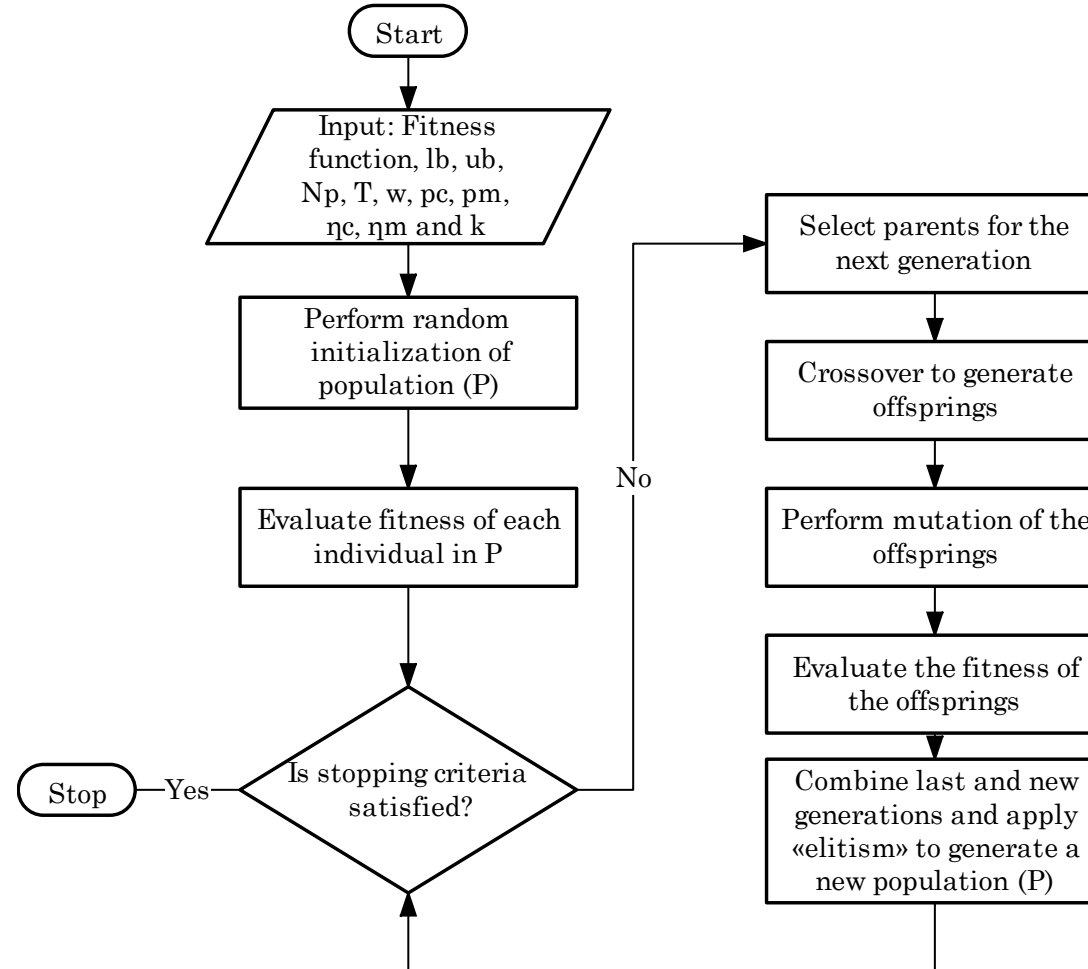
- ❖ Rastgele adaylardan oluşan başlangıç popülasyonu oluşturulur. **(Initialization)**
- ❖ Adayların uygunluk değerleri hesaplanır.
- ❖ Doğal seçim yoluyla en iyi adaylar seçilir **(Selection)** ve çaprazlanır. **(Crossover)**
- ❖ Çaprazlama sonucu oluşan yeni bireyler mutasyona uğratılır **(Mutation)** ve ebeveynlerinin popülasyondaki yerlerini almaları sağlanır.
- ❖ Adayların uygunluk değeri yeniden hesaplanır. Bu işlemler, adayların çözüme istenilen seviyede yakınsaması sağlanana kadar tekrar ettirilir. **(Termination)**



6-RGA Tabanlı PID Denetleyicisi

6.2 Sözde kod

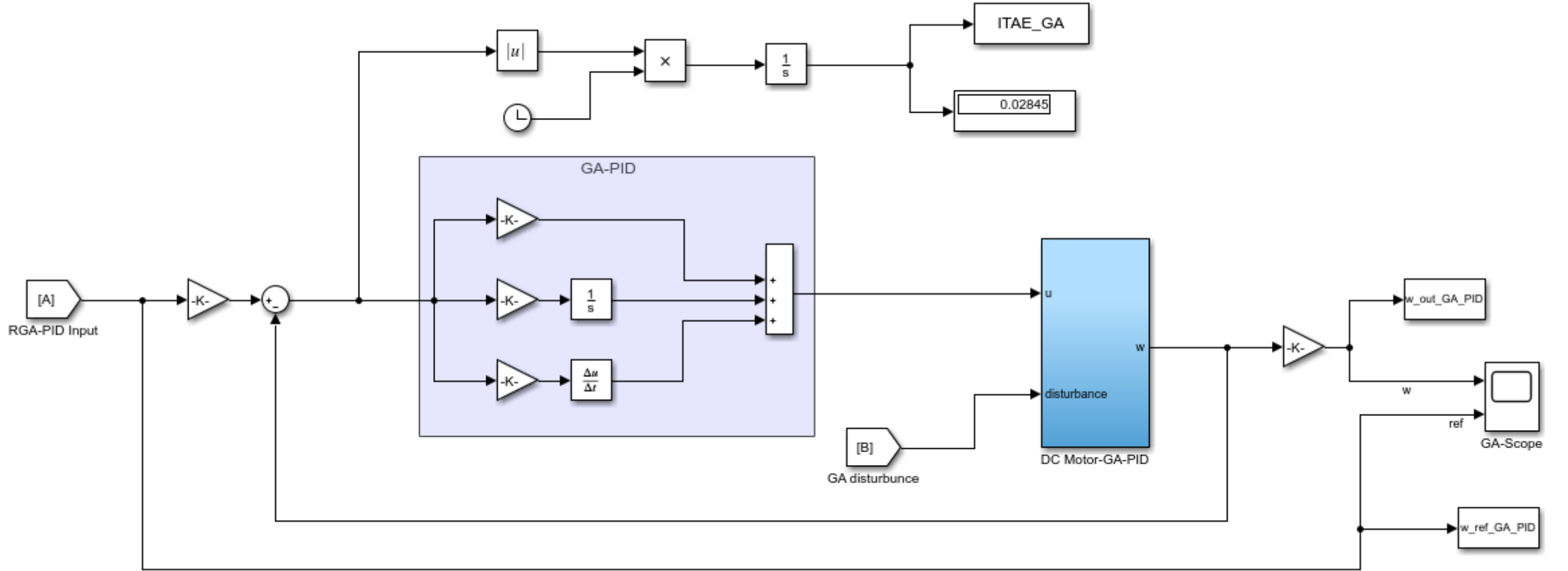
Aşağıdaki şekil, GA algoritmasının sözde kodudur:



Şekil 13: GA Algoritmasının Sözde Kodu.

6-RGA Tabanlı PID Denetleyicisi

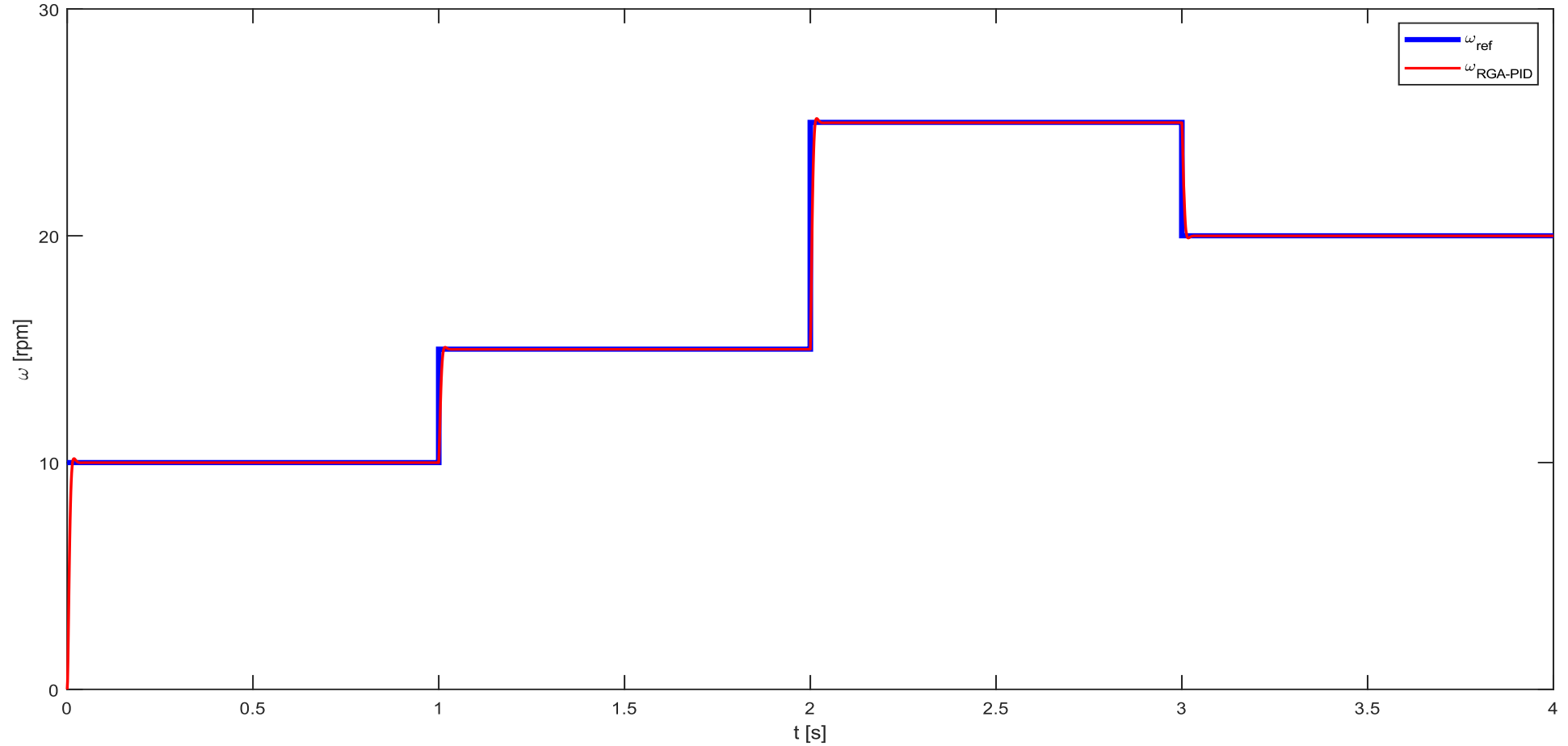
6.3 RGA-PID'nin Performansı



Şekil 14: Simulink'te yapılan DC motor ve RGA-PID modeli.

6-RGA Tabanlı PID Denetleyicisi

6.4 RGA-PID'nin Performansı

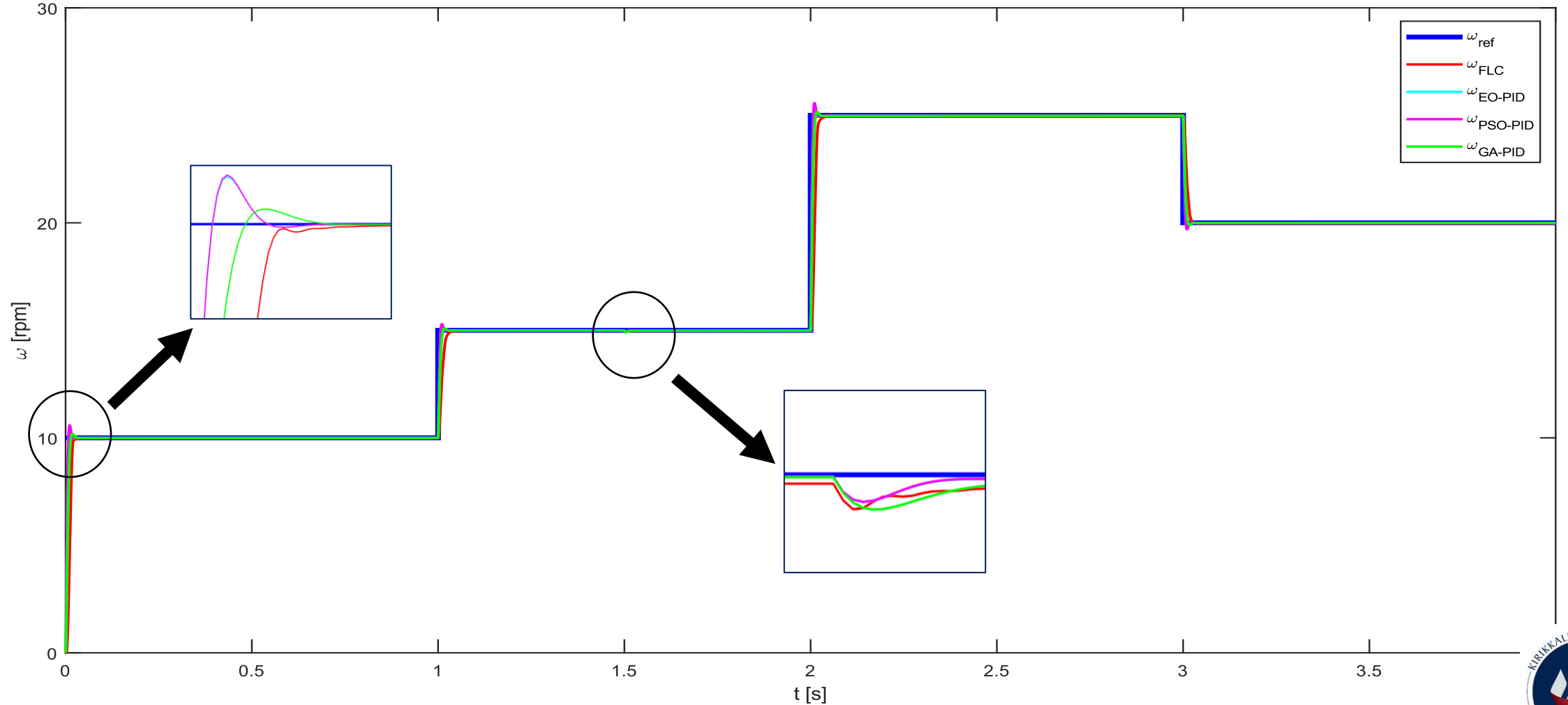


Şekil 15: Değişken giriş altında RGA – PID tarafından kontrol edilen DC motorun performansı.



7-Denetleyicilerin Performans Karşılaştırması

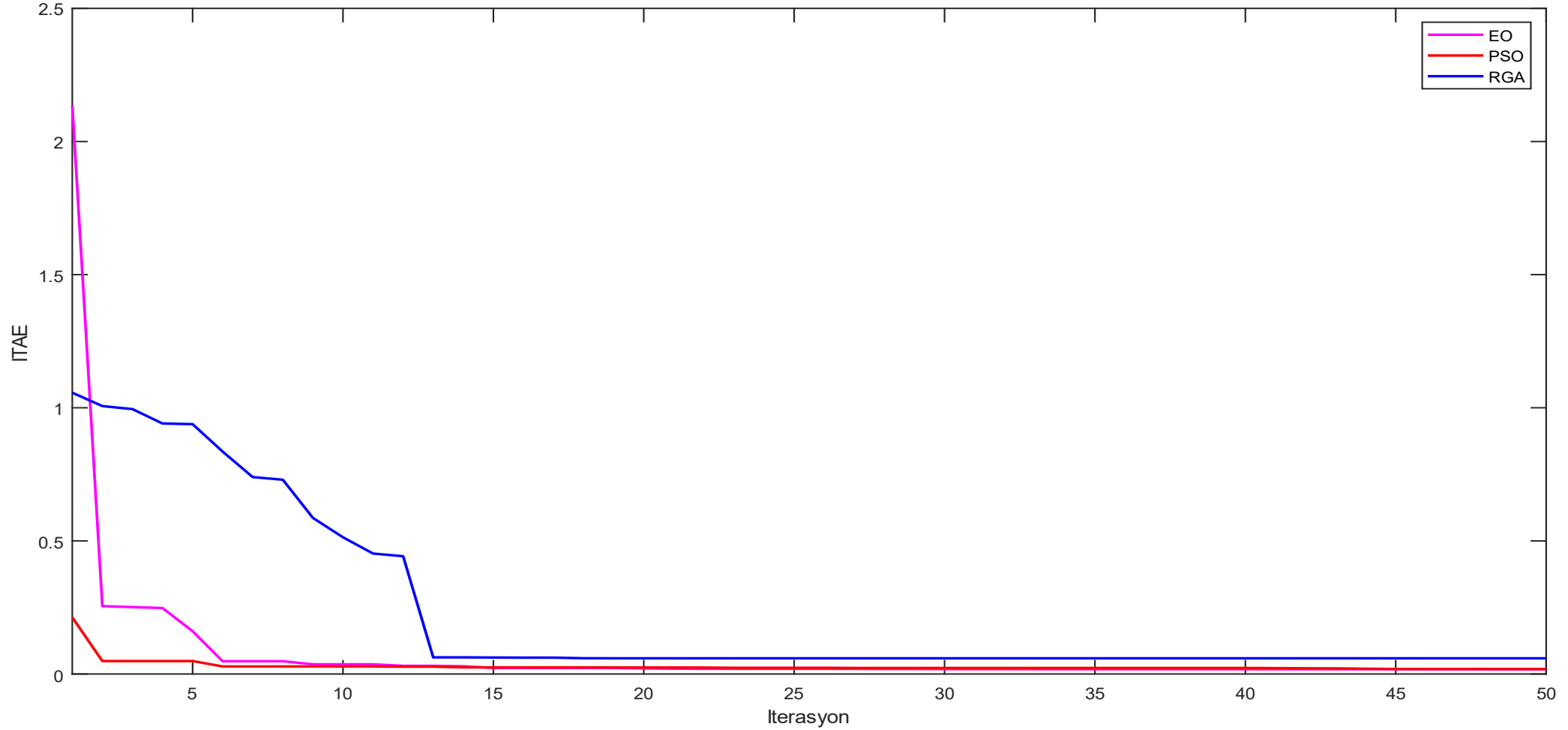
7.1 Kontrolörlerin Performans Karşılaştırması



Şekil 16: Bir bozulma altındaki kontrolörlerin Performans Karşılaştırması.

7-Denetleyicilerin Performans Karşılaştırması

7.2 Algoritmaların Yakınsama Karşılaştırması



Şekil 17: ITAE amaç fonksiyonu için EO, PSO ve RGA yakınsamalarının karşılaştırılması.

7-Denetleyicilerin Performans Karşılaştırması

7.3 Zaman yanıt performansı

Tablo 2: Kontrol Tekniklerinin DC Motor Sistemi Üzerinden Zaman Tepkisi Performansı

PERFORMANS GÖSTERGELERİ	EO-FLC	EO-PID	PSO-PID	RGA-PID
Yükselme Süresi (sn)	0.0047	0.0054	0.0054	0.0088
Aşma (%)	2.2204e-14	5.6840	5.8359	1.7987
Yerleşme Süresi (sn)	0.0169	0.0156	0.0156	0.0132
Tepe Süresi (sn)	0.2460	0.0110	0.0110	0.0180



- Bu çalışmada DC motoru kontrol etmek için EO tabanlı Fuzzy Logic Controller, EO tabanlı PID, PSO tabanlı PID ve RGA tabanlı PID kullanılmıştır.
- Öncelikle her bir kontrolörün performansı ayrı ayrı analiz edilmiş ve ardından kontrolörlerin karşılaştırması yapılmıştır.
- Elde edilen sonuçlar, PSO tabanlı PID ve EO tabanlı PID'nin parçacıkların ilk popülasyonundan bağımsız olarak yaklaşık olarak aynı performansa sahip olduğunu göstermektedir (Şekil 17).
- Yakınsama açısından, EO tabanlı PID ve PSO tabanlı PID'ye kıyasla RGA tabanlı PID'nin nihai optimum ITAE'sine yakınsamasının daha fazla zaman aldığı görülebilir (Şekil 17).
- Aynı sayıda yineleme ve aynı sayıda popülasyon için, EO algoritması ve PSO algoritması, RGA'ya göre ITAE'nin daha optimal değerini bulur (Şekil 17).



8-Sonuçlar

- Diğer kontrolörlerle karşılaştırıldığında RGA tabanlı PID daha fazla yükselme süresine sahiptir. Öte yandan, RGA tabanlı PID, diğer kontrolörlere kıyasla daha iyi yerleşme süresine sahiptir (Tablo 2).
- EO tabanlı Bulanık Mantık Denetleyicisi, diğer denetleyicilere kıyasla ihmal edilebilir aşma oranına sahiptir (Tablo 2).



- [1] C. L. Xia, *Permanent Magnet Brushless DC Motor Drives and Controls*. 2012.
- [2] M. T. Özdemir and D. Öztürk, "Comparative performance analysis of optimal PID parameters tuning based on the optics inspired optimization methods for automatic generation control," *Energies*, vol. 10, no. 12, 2017, doi: 10.3390/en10122134.
- [3] A. Ma'arif and N. R. Setiawan, "Control of DC Motor Using Integral State Feedback and Comparison with PID: Simulation and Arduino Implementation," *J. Robot. Control*, vol. 2, no. 5, pp. 456–461, 2021, doi: 10.18196/jrc.25122.
- [4] F. A. Al-durzi and N. H. Al-nuaimi, "Load Frequency Control Using Artificial Intelligent Techniques for A Single Area Power System Load Frequency Control Using Artificial Intelligent Techniques for A Single Area Power System," no. November, 2019.
- [5] J. Kennedy and R. Eberhart, "Particle swarm optimization" pp. 1942–1948, 1995.



- [6] D. V. Griffiths and I. M. Smith, "Numerical Methods for Engineers," Numer. Methods Eng., 2006, doi: 10.1201/9781420010244.
- [7] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," Knowledge-Based Syst., vol. 191, p. 105190, 2020.
- [8] S. Ekinici, A. Demirören, H. L. Zeynelgil, and B. Hekimoğlu, "An opposition-based atom search optimization algorithm for automatic voltage regulator system," J. Fac. Eng. Archit. Gazi Univ., vol. 35, no. 3, pp. 1141–1157, 2020, doi: 10.17341/GAZIMMFD.598576.
- [9] H. R. Ozcalik, A. Gani, E. Kilic, and O. F. Kececioglu, "DC Servo Motor Speed Control in Nonlinear Load by Using Different Fuzzy Membership Functions," Acad. Platf. J. Eng. Sci., vol. 2, no. 3, pp. 1–8, 2014, doi: 10.5505/apjes.2014.02486.



- [10] H. CALGAN, R. YAMAN, E. İLTEN, and M. DEMİRTAŞ, "Fırçasız DA motorunun hız kontrolünde PI katsayılarının Pareto tabanlı çok amaçlı optimizasyonu," Balıkesir Üniversitesi Fen Bilim. Enstitüsü Derg., vol. 20, no. 2, pp. 1–17, 2018, doi: 10.25092/baunfbed.468544.
- [11] G. SHAHGHOLIAN and M. MAGHSOODI, "Analysis and Simulation of Speed Control in DC Motor Drive By Using Fuzzy Control Based on Model Reference Adaptive Control," Cumhuriyet Sci. J., vol. 37, no. 3, p. 197, 2016, doi: 10.17776/csj.15867.
- [12] T. Success et al., "Kahramanmaraş Sutcu Imam University Journal of Engineering Sciences and Fuzzy Type Inspection Methods in Speed Control of DC Motor," vol. 20, no. 4, pp. 158–167, 2017.
- [13] M. Bulut, M. B. Kurt, and M. D. E. M. İ. R. T. A. Ş, "Genetik-Bulanık Kontrolörün DC Motora Uygulanması," Politeknik Derg., vol. 7, no. 4, pp. 277–285, 2004.



- [14] R. Klempka and B. Filipowicz, "Optimization of a DC motor drive using a firefly algorithm," 2018 Int. Symp. Electr. Mach. SME 2018, pp. 0–5, 2018, doi: 10.1109/ISEM.2018.8442512.
- [15] B. Yenİpınar, C. Yılmaz, Y. Sönmez, and M. F. Işık, "Asenkron Motorlarda Rotor Oluğunun Farklı Optimizasyon Yöntemleri ile Optimizasyonunun Gerçekleştirilmesi ve Performansa Etkilerinin İncelenmesi Optimizing the Rotor Slot Dimensions of Asynchronous Motor Using Different Optimization Methods and Investigating Effects on the Motor Performance," vol. 21, no. 1, pp. 1–6, 2018.
- [16] B. HEKİMOĞLU, "Speed Control of DC Motor Using PID Controller Tuned via Kidney- inspired Algorithm," BEU J. Sci., vol. 8, no. 2, pp. 652–663, 2019.
- [17] S. Tang, Y. Zhu, F. Yang, and Q. Xu, "Based on Ant Colony Optimization Algorithm," pp. 231–239, 2014.



Teşekkür ederim

Nelson Luís Manuel

E-posta: 198804001@kku.edu.tr / nelsonluismanuel@gmail.com