

Parçacık Sürüsü Optimizasyonu Kullanılarak Ters Sarkaç Sisteminin Kesir Dereceli PID ile Kontrolü

Control of Inverted Pendulum Using Fractional Order PID Controllers Based on Particle Swarm Optimization

Cemilcan Macit^{1,2}, Banu Ataşlar-Ayyıldız^{1,3}

¹Elektronik ve Haberleşme Mühendisliği Bölümü, Kocaeli Üniversitesi, Kocaeli, Türkiye

²{cemilcanmacit}@gmail.com

³{banu.ayyildiz}@kocaeli.edu.tr

Özetçe —Robotik alanında yapılan bilimsel çalışmalarda, denge konusunu en iyi temsil eden örnek ters sarkacın denge hareketidir. Bu sebeple, kararsız ve doğrusal olmayan yapısıyla ters sarkaç sistemleri, kontrol tasarım yöntemlerinin uygulandığı ve performans karşılaştırmasının yapıldığı başlıca sistemlerden biridir. Bu çalışmada, ters sarkaç sisteminde yer alan sarkaç ve araç konumunu kontrol etmek için Parçacık Sürüsü Optimizasyonu (PSO) algoritması ile optimize edilen kesir dereceli PID kontrolör tasarımı yapılmıştır. Ayrıca, önerilen kontrolörün performansını test etmek amacıyla, aynı optimizasyon algoritması kullanılarak PID kontrolör tasarlanmıştır. Tasarımı yapılan kontrolörler için kontrol sonuçları simülasyonlarla elde edilerek, grafiksel olarak ters sarkaç sistemi üzerinde performansları karşılaştırılmıştır.

Anahtar Kelimeler—ters sarkaç sistemi; kesir dereceli PID; parçacık sürüsü optimizasyonu.

Abstract—Issue of balance in robotics is best represented by the balancing act of the inverted pendulums. With their unstable and nonlinear behaviour, Inverted Pendulum systems are quite popular systems in which various control design methods are applied and performance comparisons are carried out. In this study, in order to control the pendulum angle and car position of the inverted pendulum system, a fractional order PID controller is designed. Gains of the designed controller are optimized by Particle Swarm Optimization method. On the other hand, for the aim of comparison of performance, conventional PID controllers are used for controllers. The optimum values of gains for PID controllers are also found by same optimization algorithm. The simulation results of both controllers are compared for the inverted pendulum.

Keywords—inverted pendulum; fractional order PID; particle swarm optimization.

I. GİRİŞ

Robotik alanında yapılan bilimsel çalışmalarda, denge konusunu en iyi temsil eden örnek ters sarkacın denge hareketidir. Ayrıca, doğrusal olmayan ve kararsız yapıda olmaları sebebiyle, geliştirilmiş olan modern kontrol yöntemlerinin performans değerlendirmesinin yapılması amacıyla tercih edilen başlıca uygulama sistemlerinden biridir.

Ters sarkaç sistemlerinin kontrol edilmesi için yapılmış olan çalışmalar, genellikle sistemin kararsız tepe noktasında dengede kalması üzerinedir. Yapılmış olan çalışmalar PID kontrolör gibi klasik kontrolörler [1]–[3], modern kontrol yöntemleri [4]–[8] ve yapay zeka kontrol yöntemleri (yapay sinir ağları, bulanık mantık ve genetik algoritma) [9]–[13] başlıkları altında örneklendirilebilir.

Bu çalışma kapsamında öncelikle, ters sarkaç sisteminin modeli çıkarılmıştır. Doğrusal olmayan bu model MATLAB/SIMULINK ortamına aktarılmıştır. Ardından, birinin sistemin sarkaç açısını ve diğerinin ise arabanın konumunu kontrol etmesi amacıyla, iki adet kontrolörün yer aldığı bir geribeslemeli kontrol yapısı tasarlanmıştır. Her bir kontrolör için, kesir dereceli PID kontrolör kullanılarak, beş adet kazanç değerini de içeren birer kontrolör yapısı oluşturulmuştur. Toplam on adet kazanç değişkeninin optimum değerler alması için Parçacık Sürüsü Optimizasyonu yöntemi tercih edilmiştir. İntegral Mutlak Hata (IAE) uygunluk fonksiyonu üzerinden optimum kazanç değerleri elde edilerek, ters sarkaç sisteminin kesir dereceli PID kontrolü sağlanmıştır. Ayrıca, karşılaştırma yapmak amacıyla, her iki kontrolör yerine geleneksel PID kontrolörler yerleştirilmiş ve Oransal, İntegral ve Türev olmak üzere üçer adet olan toplam altı kazanç değeri Parçacık Sürüsü Optimizasyonu ile elde edilerek ters sarkaç sisteminin kontrolü sağlanmıştır.

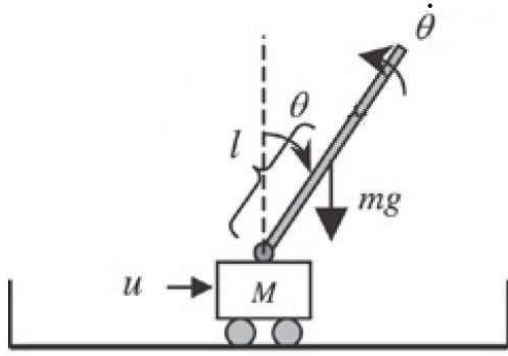
II. TERS SARKAÇ SİSTEMİ (TSS)

A. Ters Sarkaç Sisteminin Doğrusal Olmayan Dinamik Modeli

Şekil 1’de şematik olarak yapısı görülen ters sarkaç sisteminin dinamik denklemlerin bulunması için Lagrange fonksiyonundan yararlanılmaktadır. (1)’de verilen Lagrange denkleminde, T toplam kinetik enerjiyi, V ise toplam potansiyel enerjiyi ifade etmektedir.

$$L = T - V \quad (1)$$

Kinetik enerji denklemi sarkaç ve araba için çıkarılan kinetik enerji denklemlerinin toplamıdır [9]:



Şekil 1: Ters Sarkaç Sistemi

$$T = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}^2 - 2\dot{x}l\cos\theta\dot{\theta} + l^2\dot{\theta}^2) + \frac{1}{2}I\dot{\theta}^2 \quad (2)$$

Sistemin potansiyel enerjisini ise, araba yatay ekseninde hareket ettiği için sadece sarkaç belirlemektedir:

$$V = -mgl\cos\theta \quad (3)$$

Toplam kinetik enerji ve toplam potansiyel enerji ifadeleri (1)'de yerine yazılarak gerekli düzenleme ve sadeleştirmeler yapıldığında (4) ve (5)'deki dinamik denklemler elde edilmektedir [9]. Bu denklemlerde yer alan sisteme ait tüm parametreler ve değerleri Tablo I'de sunulmuştur.

$$(M + m)\ddot{x} - ml\cos\theta\ddot{\theta} + ml\sin\theta\dot{\theta}^2 = f(t) - c\dot{x} \quad (4)$$

$$(ml^2 + I)\ddot{\theta} - ml\ddot{x}\cos\theta + mgl\sin\theta = -b\dot{\theta} \quad (5)$$

Tablo I: Ters Sarkaç Sisteminin Parametreleri [9]

Parametre	Sembolü	Değeri	Birimi
Arabanın kütlesi	M	1.095	kg
Sarkacın kütlesi	m	0.2	kg
Sarkacın kütle merkezinin uzaklığı	l	0.325	m
Sarkaç-araba arası sürtünme katsayısı	b	0.1	$N/m/s$
Araba-yüzey arası sürtünme katsayısı	c	0.1	$N/m/s$
Sarkaca ait atalet momenti	I	0.0001	$kg.m^2$
Yer çekim ivmesi	g	9.81	m/s^2

III. KESİR DERECELİ PID KONTROLÖR

Kesir dereceli matematik, integral ve türevin derecesinin tam sayı olmadığı, reel veya hatta kompleks değerli olduğu işlemleri içermektedir [14]. Kesir dereceli türev ve integral için tanımlarından iki tanesi oldukça yaygın olarak kullanılmaktadır: Grünwald-Letnikov (GL) ve Riemann-Liouville (RL). Bunlardan Riemann and Liouville [15] tanımı aşağıdaki gibidir:

$${}_aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \quad (6)$$

Burada $n-1 < \alpha < n$ olmak üzere, $\Gamma(\cdot)$ Euler Gamma fonksiyonudur. a ve t sırasıyla integral fonksiyonunun alt ve üst limitleridir. Tam sayı olmayan değere sahip olan α ise, türev veya integralin derecesidir.

Sistem kontrolü uygulamalarında yaygın olarak kullanılan kontrolör tiplerinden biri olan PID kontrolörde türev ve integral derecelerinin tam sayı olmadığı duruma karşılık gelen kontrolör yapısı Kesir dereceli PID kontrolör olarak ifade edilmektedir [11]. $PI^\lambda D^\mu$ gösterimine sahip olan kesir dereceli PID kontrolör bir anlamda geleneksel PID kontrolörün genelleştirilmiş halidir. $\lambda = 1$ ve $\mu = 1$ olduğu durumda, geleneksel PID'ye karşılık gelmektedir. $PI^\lambda D^\mu$ 'ye ait diferansiyel denklem:

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^\mu e(t) \quad (7)$$

şeklinde. Laplace dönüşümü sonucunda, transfer fonksiyonu ise aşağıdaki gibi elde edilmektedir:

$$C(s) = K_p + K_i \frac{1}{s^\lambda} + K_d s^\mu \quad (8)$$

Eşitlik (8)'de verilen transfer fonksiyonunun karşılığı MATLAB'de doğrudan bulunmamaktadır. Kesirli dereceli transfer fonksiyonlarının yerine kullanılmak üzere tam sayı değerli yaklaşıklık yöntemleri geliştirilmiştir [17]–[19].

Bu çalışmada ise, MATLAB (R2013b sürümü)'de, bir kesir dereceli transfer fonksiyonunun ayrık ya da sürekli yaklaşımını bulmak için geliştirilmiş olan NINTEGER TOOLBOX'ı kullanılmıştır. Kesir dereceli PID kontrolörlere ait transfer fonksiyonlarının CRONE yaklaşıklık yöntemi kullanılarak, $10^{-2} - 10^2$ frekans aralığında ve 5. dereceden yaklaşıklıkla elde edilmiştir.

IV. PARÇACIK SÜRÜSÜ OPTİMİZASYONU

Parçacık Sürüsü Optimizasyonu (PSO) kuş sürülerinin sosyal davranışları temel alınarak 1995 yılında geliştirilmiş olan, evrimsel bir optimizasyon tekniğidir [20]. Başlangıçta rastgele olarak atanan ve parçacık adı verilen çözümlerle başlayan ve kuşların en iyi besin lokasyonunu aramasının benzetimi olan bir algorithmadan oluşmaktadır. Diğer evrimsel optimizasyon tekniklerinden farklı olarak, Parçacık Sürüsü Optimizasyonunda her bir parçacığın hız bilgisi bulunmaktadır. Parçacıklar önceki davranışlarına göre belirlenen hızlarla arama uzayında dolaşırlar. Böylece, parçacıklar arama rotası üzerinden daha iyiye doğru yol alırlar. Her parçacık geçmişteki konumlarından daha iyi olana gitme ve aynı zamanda sürü içerisindeki yiyeceğe en yakın konumda bulunan parçacığı takip etme eğiliminde bulunur.

Parçacık Sürüsü Optimizasyonu Algoritmasının her bir iterasyonunda parçacıkların hızları ve pozisyonları sırasıyla aşağıdaki ifadelerle güncellenmektedir:

$$v_i^{k+1} = wv_i^k + c_1 rand_1(pbest_i - p_i^k) + c_2 rand_2(gbest - p_i^k) \quad (9)$$

$$p_i^{k+1} = p_i^k + v_i^{k+1} \quad (10)$$

Bu denklemlerde v_i^k , k iterasyonu için i inci parçacığın hızını, p_i^k , k iterasyonu için i inci parçacığın pozisyonunu, w eylemsizlik ağırlık fonksiyonunu, $c_{1,2}$ öğrenme faktörlerini, $rand_{1,2}$ ise, $[0,1]$ aralığında atanan rastgele sayı değerlerini ifade etmektedir. Ayrıca, $pbest_i$, i inci parçacığın o ana kadar elde ettiği en iyi çözümü sağlayan koordinatlarıdır. $gbest$ ise, tüm parçacıklar tarafından elde edilen en iyi çözümü sağlayan koordinatlarıdır.

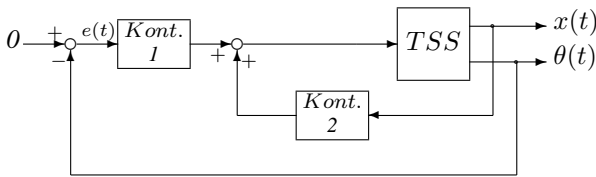
w eylemsizlik ağırlığı, global ve yerel arama yeteneğini dengelemek için kullanılmaktadır. Büyük değerdeki eylemsizlik ağırlığı global arama, küçük değerdeki ise, yerel arama yapılmasını kolaylaştırmaktadır. Bu çalışmada $w = 0.9$ olarak seçilmiştir.

(9)–(10) ifadelerini de içeren Parçacık Sürüsü Algoritması aşağıdaki adımlarla özetlenebilmektedir:

- 1. Adım:** Başlangıç sürüsünü, pozisyonlarını ve hızlarını ata.
- 2. Adım:** Sürüdeki her bir parçacığın uygunluk fonksiyonu değerini hesapla.
- 3. Adım:** Her bir parçacığın uygunluk değerini önceki en iyisi ile karşılaştır. Daha iyi ise, en iyi olarak ata ($pbest$).
- 4. Adım:** En iyi yerel değerleri kendi arasında karşılaştır ve en iyi olanı global en iyi olarak ata ($gbest$).
- 5. Adım:** Hız ve pozisyon değerlerini güncelle.
- 6. Adım:** Maksimum iterasyon sayısına ulaşıldı mı?
VEYA
Uygunluk değeri istenen değere ulaştı mı?
HAYIR ise, 2. Adıma dön.
EVET ise, algoritmayı sonlandır.

V. TERS SARKAÇ SİSTEMİNİN KONTROL TASARIMI

Bu çalışmada, ele alınan ters sarkaç sisteminin kontrolü iki adet kontrolör ile yapılmıştır. Birinin sistemin sarkaç açısını (θ), diğerinin ise arabanın konumunu (x) kontrol etmesi amacıyla, iki adet kontrolörün yer aldığı bir geribeslemeli kontrol yapısı oluşturulmuştur (Şekil 2).



Şekil 2: TSS için Geribeslemeli Kontrol Yapısı

A. Kesir Dereceli PID Kontrolör Tasarımı

Ters sarkaç sisteminin kontrolü için hazırlanan Şekil 2'deki geribeslemeli kontrol yapısında kontrolör olarak kesir dereceli PID kontrolör kullanılmıştır.

Hazırlanmış olan her bir kesir dereceli PID kontrolörde yer alan beşer adet kazanç değerinin (K_p , K_i , λ , K_d ,

μ) optimum değerlerini belirlemek amacıyla, Bölüm IV'de yer verilen Parçacık Sürüsü Optimizasyonu yöntemi kullanılmıştır. Parçacık Sürüsü Optimizasyonu Algoritması için MATLAB'de program yazılmıştır. Hazırlanan algorithmada kullanılan parametreler Tablo II'de sunulmuştur.

Tablo II: PSO Algoritması Parametreleri

Parametre	Değişken	Değeri
Sürü boyutu	n	50
Maksimum iterasyon sayısı	$iter_max$	20
Momentum	w	0.9
1. Öğrenme Faktörü	c_1	0.12
2. Öğrenme Faktörü	c_2	1.2

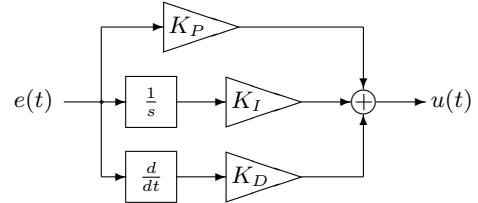
Parçacık Sürüsü Optimizasyonunda kuş sürülerinde bir kuşun bulunduğu koordinatın yiyeceğe ne kadar uzaklıkta olduğu bilgisine, probleme uygun olarak belirlenmekte olan *Uygunluk Fonksiyonu* karşılık gelmektedir. Bu çalışmada, uygunluk fonksiyonu olarak İntegral Mutlak Hatası (IAE) fonksiyonu kullanılmıştır:

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

B. PID Kontrolör Tasarımı

Tasarlanmış olan Bulanık Mantık tabanlı PID Kontrolör için performans karşılaştırması yapmak amacıyla, dönel ters sarkaç sisteminin kontrolü için hazırlanmış olan Şekil 2'deki yapıda geleneksel PID kontrolörler de kullanılmıştır (Şekil 3).

Her bir PID kontrolöründe yer alan oransal, integral ve türevden oluşan üçer adet kazanç değerinin (K_p , K_i , K_d) optimum değerlerini belirlemek amacıyla, Parçacık Sürüsü Optimizasyonu yöntemi kullanılmıştır. Optimizasyon parametreleri Bulanık Mantık kontrolörde olduğu gibi Tablo II'deki değerlerdir.



Şekil 3: Geleneksel PID Kontrolör Yapısı

VI. BENZETİM ÇALIŞMALARI

Parçacık Sürüsü Optimizasyonu çalışması ile elde edilmiş olan, Kesir Dereceli PID kontrolörlere ait kazanç değerleri ve geleneksel PID kontrolörlere ait kazanç değerleri sırasıyla Tablo III ve Tablo IV'de sunulmuştur. Elde edilmiş olan bu kazanç değerleri kullanılarak ters sarkaç sistemi kontrol edildiğinde, elde edilen sarkaç açısı ve araç konumu grafikleri sırasıyla Şekil 4 ve Şekil 5'de sunulmuştur. Grafikler incelendiğinde, hem Kesir Dereceli PID kontrolörün ve hem de PID kontrolörün sistemi başarı ile kontrol ettiği görülmektedir. Her ikisinde de sarkaç açısı ve araba konumu sifıra oturmaktadır.

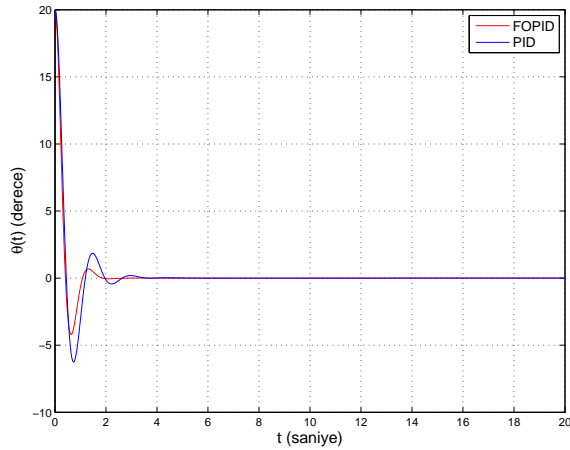
Tablo III: Kesir Dereceli PID Kontrolör Kazanç Değerleri

	K_p	K_i	λ	K_d	μ
Kesir Der. PID Kont. 1	0.72	3.68	0.42	6.89	0.66
Kesir Der. PID Kont. 2	0.10	0.10	0.79	0.46	0.51

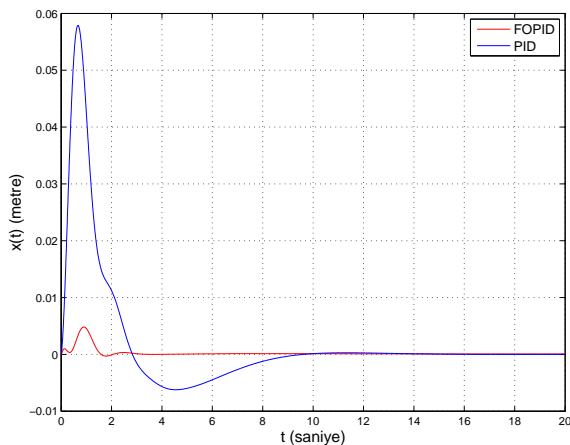
Tablo IV: PID Kontrolör Kazanç Değerleri

	K_P	K_I	K_D
PID Kont. 1	4.14	1.77	4.92
PID Kont. 2	4.50	1.39	0.10

İki kontrolör tipi için elde edilen sonuçlar birbirleri ile kıyaslandığında, sarkaç açısı için, Kesir Dereceli PID kontrolörün sonucunun 2 saniyeden daha kısa zamanda sıfır dereceye oturduğu görülmektedir. Geleneksel PID kontrolör sonucunda ise, sarkaç açısı yaklaşık 4 saniyede sıfır dereceye oturmaktadır. Araba konum grafikleri karşılaştırıldığında da, Kesir Dereceli PID kontrolör sonucunun daha iyi olduğu görülmektedir. PID kontrolör sonucunda, Kesir Dereceli PID kontrolöre oranla yüksek maksimum üst aşım yüzdesi oluşmaktadır.



Şekil 4: Sarkaç Açısı



Şekil 5: Araba Konumu

VII. SONUÇLAR

Bu çalışmada, ters sarkaç sisteminde yer alan sarkaç açısını ve araba konumunu kontrol etmek için Parçacık Sürüşü Optimizasyon algoritması ile optimize edilen kesir dereceli PID kontrolör tasarımı yapılmıştır. Kontrol sonucunda sarkaç açısının ve araba konumunun sıfıra oturması ile başarılı sonuçlar alınmıştır.

Elde edilen sonuçlar, kazanç değerleri Parçacık Sürüşü Optimizasyon algoritması ile optimize edilen geleneksel PID kontrolörlerle elde edilen sonuçlarla karşılaştırılmıştır. Kesir dereceli PID kontrolör ile geleneksel PID kontrolör sonuçlarına göre hem daha hızlı ve hem de daha yumuşak değişimli sonuçlar alınmıştır.

KAYNAKÇA

- [1] J.-J. Wang, "Simulation studies of inverted pendulum based on PID controllers," *Simulation Modelling Practice and Theory*, 19(1), s. 440–449, 2011.
- [2] J.-J. Wang, "Position and speed tracking control of inverted pendulum based on double PID controllers," *34th Chinese Control Conference (CCC)*, 2015.
- [3] K. Razzaghi, A. A. Jalali, "A New Approach on Stabilization Control of an Inverted Pendulum, Using PID Controller," *2011 International Conference on Control, Robotics and Cybernetics*, 2011.
- [4] A. Yazıcı ve A. Karamancıoğlu, "Ters Sarkaç Sisteminin Kontrol Eğitiminde Test Aracı Olarak Kullanılması," *Elektrik Elektronik Bilgisayar Biyomedikal Mühendislikleri Eğitimi IV. Ulusal Sempozyumu*, Eskişehir, 2009.
- [5] V. Kumar, J. Jeromeb, "Robust LQR Controller Design for Stabilizing and Trajectory Tracking of Inverted Pendulum," *International Conference On Design and Manufacturing*, s. 169–178, 2013.
- [6] A. Yüce, N. Tan, "Ters Sarkaç Sistemi için Lag/Lead Kontrolör Tasarımı," *Otomatik Kontrol Ulusal Toplantısı (TOK 2013)*, s. 303–308, 2013.
- [7] V. Sukontanakarn, M. Parnichkun, "Real-Time Optimal Control for Rotary Inverted Pendulum," *American Journal of Applied Sciences*, 6(6), s. 1106–1115, 2009.
- [8] M.A. Khanezar, M. Teshnehlal, M. A. Shoorehdeli, "Sliding Mode Control of Rotary Inverted Pendulum," *15th Mediterranean Conference on Control and Automation*, 2007.
- [9] S. Kizir, "Doğrusal olmayan ters sarkaç sisteminin tasarımı ve kontrolü," *Yüksek Lisans Tezi*, Kocaeli Üniversitesi, 2008.
- [10] S. Horikawa, M. Yamaguchi, T. Furuhashi, Y. Uchikawa, "Fuzzy Control for Inverted Pendulum Using Fuzzy Neural Networks," *JRM*, c. 7, no. 1, s. 36–44, 1995.
- [11] Zadeh, I.H. and S. Mobayen, "PSO-based controller for balancing rotary inverted pendulum," *Journal of Applied Science*, c. 16, 2907–2912, 2008.
- [12] M. I. H. Nour, J. Ooi, K. Y. Chan, "Fuzzy Logic Control vs. Conventional PID Control of an Inverted Pendulum Robot," *International Conference on Intelligent and Advanced Systems*, s. 209–214, 2007.
- [13] A. Khosla, G. Leena, M. K. Soni, "Interval Type-2 Fuzzy Logic Controller to Control the Velocity and Angle of Inverted Pendulum," *International Journal of Intelligent Systems and Applications* c. 6, No. 7, 2014.
- [14] M. D. Ortigueira, "Fractional Calculus for Scientists and Engineers," Springer, Berlin, Germany, 2011.
- [15] D. Y. Xue, and Y. Q. Chen, "A comparative Introduction of Four Fractional order Controllers," *Proc. of the 4th World Congress on Intelligent Control and Automation*, c. 4, s. 3228–3235, Shanghai, 2002.
- [16] I. Podlubny, "Fractional-Order Systems and PID Controllers," *IEEE Transactions on Automatic Control*, c. 44, no. 1, s. 208–214, 1999.
- [17] A. Oustaloup, "La commande CRONE: commande robuste d'ordre non entiere," Herme's, Paris, 1991.

- [18] A. Oustaloup, F. Levron, B. Mathieu, F. M. Nanot, "Frequency-band complex noninteger differentiator: characterization and synthesis," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, c. 47, no. 1, s. 25–39, 2000.
- [19] D. Valério and J. S. da Costa, "Time-domain implementation of fractional order controllers," *IEE Proceedings Control Theory and Applications*, c. 152, no. 5, s. 539–552, 2005.
- [20] J. Kennedy, ve R. C. Eberhart, "Particle swarm optimization," *Proc. IEEE int'l Conf. on Neural Networks*, c. IV, s. 1942–1948, 1995.