Koc. Üni. Fen Bil. Der., 3(1): (2020) 99-108

Kocaeli Üniversitesi

Kocaeli Üniversitesi Fen Bilimleri Dergisi

http://dergipark.gov.tr/koufbd

PID ve Bulanık Mantık Kontrol Sistemleri ile İki Tekerlekli Kendini Dengeleyebilen Robotik Sistem Tasarımı

Design of a Two-Wheel Self-Balancing Robotic System with the PID and Fuzzy Logic Control Systems

Furkan PEÇE[1,*], Eser YARAR[2], Sedat KARABAY[3]

[1] *Makine Mühendisliği Bölümü, Mühendislik Fakültesi, Kocaeli Üniversitesi, Kocaeli, Türkiye,* ***Orcid:*** *0000-0001-5704-896X*
[2] *Makine Mühendisliği Bölümü, Mühendislik Fakültesi, Kocaeli Üniversitesi, Kocaeli, Türkiye,* ***Orcid:*** *0000-0003-1187-5382*
[3] *Makine Mühendisliği Bölümü, Mühendislik Fakültesi, Kocaeli Üniversitesi, Kocaeli, Türkiye,* ***Orcid:*** *0000-0002-3258-0957*

Araştırma Makalesi

Gönderilme Tarihi : *02/01/2020*
Kabul Tarihi : *01/06/2020*

Anahtar Kelimeler

Ters Sarkaç Sistemi
Kendi Kendini Dengeleme
Segway
Bulanık Mantık Kontrol Sistemi
PID Kontrol Sistemi

Özet

Bu çalışmada, iki tekerlekli kendini dengeleyebilen robotik sistemin tasarımı ve kontrol yöntemleri üzerine çalışılmıştır. Ters sarkaç sistemiyle aynı prensibe sahip olan iki tekerlekli kendini dengeleyebilen robotik sistemlerin kontrolü, PID ve bulanık mantık kontrol sistemi ile sağlanmıştır. Matlab Simulink yardımıyla blok diyagramları oluşturulup sonuçlar karşılaştırılmıştır. Giriş olarak motorlardaki elektrik girişi ve çıkış olarak sistemin açısal değişimi kabul edilmiştir. Matematiksel modeller Kirchhoff voltaj ve Newton hareket yasalarından elde edilmiştir. Nonlineer denklemleri lineer hale getirip, sistemin durum uzay matrisine ulaşılmıştır. Tasarlanan iki tekerlekli robotik sistem, ters sarkaç mantığına göre kendini dengeye getirebilmektedir. Sistem denge pozisyonundayken, gövdenin açısal hareket yapmasıyla sistemin ağırlık merkezi değişmektedir. Ağırlık merkezindeki değişim ivmeölçer ile programlayıcıya geri besleme yaparak iletilir. Programlayıcı da motor sürücülerine yetki vererek sistemin tekerleklerine gerekli hareketi sağlar. Böylece sistem, kendini tekrardan denge pozisyonuna getirir. Sonuçlara göre, PID kontrolcünün bulanık mantık kontrolcüye göre sistemin denge ve konum kontrolünde daha iyi olduğu anlaşılmıştır. Prototip üretimi için ise, Arduino Uno kart, Mpu 6050 sensör, motorlar, motor sürücüsü, güç kaynağı ve şase için makrolon malzemeler kullanılmıştır.

Research Paper

Received Date : *02/01/2020*
Accepted Date : *01/06/2020*

Keywords

Inverted Pendulum System
Self – Balancing
Segway
Fuzzy Logic Control System
PID Control System

Abstract

In this article, the design and control methods of the two-wheel self-balancing robotic system were studied. PID and Fuzzy Logic control systems were used to control two-wheel self-balancing robotic systems which have the same principle as the inverted pendulum system. Block diagrams were created with Matlab Simulink, and the results were compared. The input of the system, the electrical input of the motors, the output of the system was considered as angular changes of the system. Mathematical models were derived from Kirchhoff voltage and Newton's laws of motion. The state-space matrix of the system was determined by transforming nonlinear equations to linear. The designed robotic system can balance itself according to the inverted pendulum logic. While the system is in balance position, the center of gravity of the system changes with the angular movement of the body. The change in the center of gravity is fed back to the controller with the accelerometer. The controller authorizes motor drivers to move to the wheels of the system. Thus, the system is brought back to the balance position itself. According to the results, it was found that the PID controller is better than the Fuzzy Logic controller in balance and position control of the system. Arduino Uno card, Mpu 6050 sensor, motors, motor driver, power supply and makrolon materials for chassis have used to obtain the prototype.

1. Giriş

İki tekerlekli kendini dengeleyebilen robotik sistemler, elektrikli olması sebebiyle hava kirliliği oluşturmaması, diğer araçlara göre düşük maliyetli olması, her yaştan insanın kullanabilmesinin yanında hafif hacim ve yüksek manevraya sahip olmasından dolayı günümüzde oldukça yaygınlaşmıştır. Bu sistemlerin modellenmesi ve kontrolü ile ilgili sayısız araştırma yapılmış, birçok farklı konseptlerde imal edilmiştir. İnsan tarafından ve dışarıdan kontrolle de sürülebilmektedir. Bu anlayışın içinde ticari pazarda en çok başarı sağlamış, güvenlik ve performans

* Sorumlu Yazar *(Corresponding Author)*: f.pecee@gmail.com

açasından en iyi olan ürün, Dean Kamen tarafından üretilen Segway insan taşıyıcı robotudur. İnsan tarafından sürülen sistemlerin çalışma mantığı, sürücü öne ve geriye doğru eğilerek aracın hızını ve önündeki gidon sayesinde ise gideceği yönü belirlemesidir. Segway'den ilham alan birçok uygulama vardır. Bu çalışmalar robotik sistemin daha hafif ve daha düşük maliyetli olabileceğini ispatlamaktadır [1-4].

Bu sistemlerin kontrolünü sağlamak için birçok farklı yöntemler kullanılmıştır. 2005' te Pathak ve arkadaşları hız ve konum için kısmi geri beslemeden bahsetmiştir [5]. 2006' da Forrest ve arkadaşları yapay sinir ağları kullanarak araçların kontrolünü yapmaya çalışmıştır [6]. Aynı yılda Kim hız ve konum için lineer durum uzay modeli kullanmıştır [7]. Kim'in yaptığı çalışmadan faydalanarak Nawawi ve arkadaşları robotu dengeleyebilmek için kutup teorisini ortaya atmıştır [8]. 2007' de Jeong ve Takahashi doğrusal kuadratik düzenleyici ile geri beslemeyi denemiştir [9]. 2009' da Li ve Xu bulanık mantık teoremi uygulamıştır [10]. Huang 2010 da LQR metodunu kullanmıştır [11]. Li ve arkadaşları araçlar üzerinde PID kontrol kullanmıştır [12]. Mahadi ve arkadaşları 2011 yılında kendini dengeleyebilen robotlar üzerinde PD kontrol kullanmıştır [13]. Xu ve arkadaşları tarafından 2013 yılından üretilen iki tekerlekli bir robotun bulanık mantık ile kontrolü uygulaması yapılmıştır. İlk olarak integral kayma kontrol metodu kullanarak 2014 yılında Xu ve arkadaşları tarafından geliştirilmiştir [14].

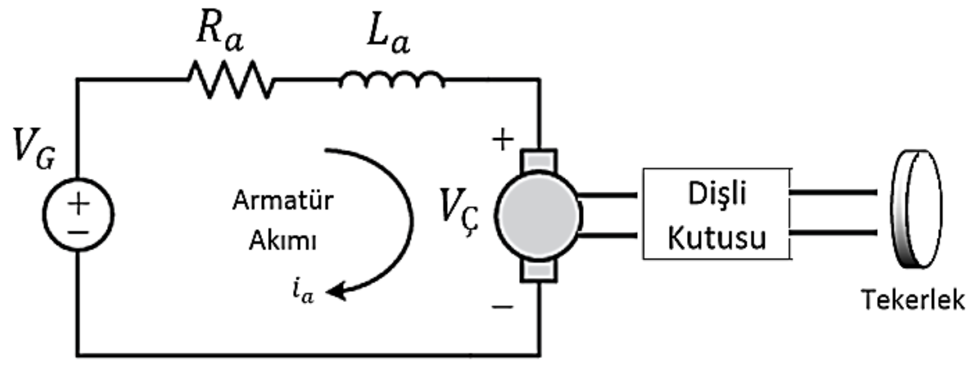
Literatür araştırması iki tekerlekli robotik sistemlerin kontrolü için farklı yöntemlerin kullanıldığını göstermiştir. Yapmış olduğumuz çalışma iki tekerlekli robotik sistemler için hem PID, hem de bulanık mantık kontrol uygulamasını sunmaktadır. Tasarlanan robotik sistem için elde edilen matematiksel denklemler Matlab Simulink ortamında PID ve bulanık mantık kontrolcüsü için uygulanmıştır. Çalışmanın amacı PID ve bulanık mantık kontrolcüsünün konum, hız ve ivme bakımından karşılaştırmaktır. Ayrıca elde edilen sonuçlara göre iki tekerlekli robotik sistem için bir prototip tasarlanmış ve PID kontrol yöntemi ile kontrol edilmiştir.

2. Matematiksel Model

Bu bölümde iki tekerlekli kendini dengeleyebilen robotik sistemin matematik denklemlerini elde edebilmek için izlememiz gereken adımlar belirtilmiştir. Tasarım aşamasında sistemin düz yolda herhangi bir engebeyle karşılaşmadığı düşünülerek hesaplamalar yapılmıştır.

Öncelikle tekerlere hareketi sağlayan DC motorlardan tekerlerin torkunun elde edilmesi gerekmektedir. Daha sonra tekerlere etki eden kuvvetlerin yardımıyla sistemdeki sürtünme kuvvetini ve x yönündeki ivmenin hesaplanması gerekmektedir. Sürtünme kuvveti ve x yönündeki ivme hesaplandıktan sonra sarkacın yani sistemin açısal hareketinden yararlanarak elde edilmek istenilen x yönündeki ivme ve sistemin açısal ivmesi bulunabilir. Lagrange ve Newton'un hareket yasalarından elde edilen matematiksel model, gerçek sistemin temsilidir. Matematik modelde önemli olan sistemin atalet momentinin hesaplanmasıdır. Elde edilen matematiksel modellerle de sistemin kontrolü sağlanmaktadır.

Yukarıda özetlenen matematiksel model aşağıda belirtilen denklemlerle elde edilebilir.



Şekil 1. DC motor modellemesi ve tekerlek ilişkisi

Şekil 1' de gösterilen DC Motor elektrik devre şemasında giriş voltajı V_G aşağıda belirtilmiş şekilde hesaplanmıştır.

$$V_G = i_a R_a + L_a \frac{di_a}{dt} + V_Ç \quad (2.1)$$

DC Motorlarda motor armatür indüktansı $L_a \cong 0$ kabul edilebilir. $L_a \cong 0$ kabulü yapıldığında (2.1)'de ki denklem aşağıdaki gibi olmaktadır.

$$V_G = i_a R_a + V_Ç \quad (2.2)$$

Kirchhoff Voltaj Yasasına göre $V_{Ç}$ değeri (2.3) nolu denklemde belirtildiği gibidir.

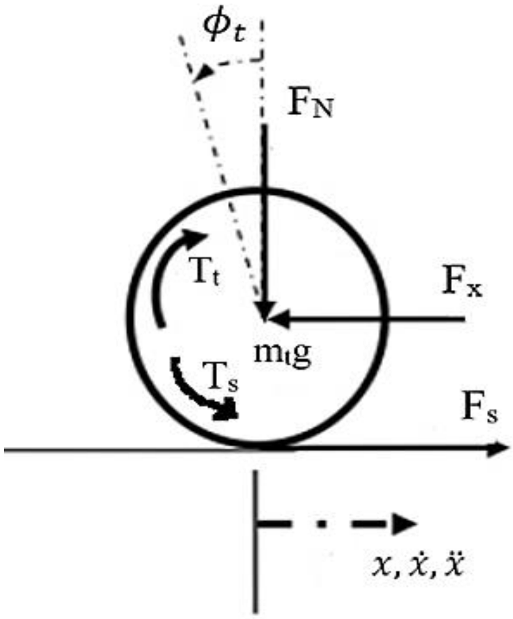
$$V_{Ç} = K_{Ç}\dot{\phi}_m = K_{Ç}\frac{di_a}{dt} \tag{2.3}$$

$$T_m = K_t\left(\frac{V_G}{R_a} - \frac{K_{Ç}}{R_a}\dot{\phi}_m\right) \tag{2.4}$$

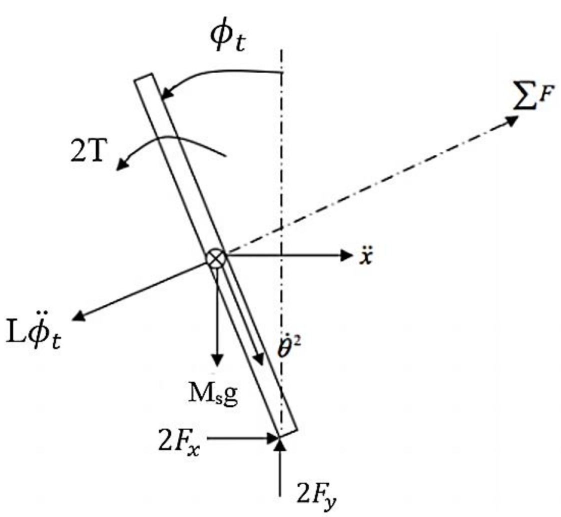
Tekerlek ve motor torku arasında ki redüktörden dolayı verim ilişkisi ile tekerleklerin ve motorun açısal pozisyonu arasında ki ilişki tekerlek tork denkleminde yazıldığında (2.9)'daki gibi bulunur.

$$T_t = \frac{K_t V_G \eta}{R_a} - \frac{K_t K_{Ç} \eta^2 \dot{\phi}_t}{R_a} \tag{2.5}$$

a)



b)



Şekil 2. Sisteme etki eden yükler **(a)** Sistemin tekerleğine etki eden kuvvetler, **(b)** Sistemin reaksiyon kuvvetleri

Newton yasasına göre x yönündeki toplam kuvvetler Şekil 2.a' dan elde edilir;

$$F_s - F_x = m_t \ddot{x}_t \tag{2.6}$$

Newton yasasına göre toplam momentler;

$$T_t - T_s - F_s r = J_t \ddot{\phi}_t \tag{2.7}$$

Motor şaftındaki sürtünme torku T_s 'nin şaft açısal hızı ile bağlantısı denklem (2.8) de belirtildiği gibidir.

$$T_s = k_m \dot{\phi}_t \tag{2.8}$$

$$m_t \ddot{x}_t = \frac{K_t V_G \eta}{R_a r} - \frac{K_t K_{Ç} \eta^2 \dot{x}_t}{R_a r^2} - \frac{k_m \dot{x}_t}{r^2} - \frac{J_t \ddot{x}_t}{r^2} - F_x \tag{2.9}$$

Matematik modeli çıkartılan robotik sistemde 2 adet tekerlek olduğu için bulunan (2.9) numaralı denklemi 2 katı ile genişletilmek zorundadır.

$$2\left(m_t + \frac{J_t}{r^2}\right)\ddot{x}_t = \frac{2K_t V_G \eta}{R_a r} - \frac{2K_t K_{Ç} \eta^2 \dot{x}_t}{R_a r^2} - \frac{2k_m \dot{x}_t}{r^2} - 2F_x \tag{2.10}$$

Her iki tekerleğin sisteme verdiği reaksiyon kuvvetleri aynı yönde ve aynı değerde olduğu kabul edilir. Şekil 2.b' de gösterilen hareket diyagramında x eksenindeki tüm kuvvetlerin toplamı sistemin kütlesi ile ivmelenmesinin çarpımına eşittir.

$$2F_x - M_s L \ddot{\phi}_t \cos\phi_t + M_s L \dot{\phi}_t^{\,2} \sin\phi_t = M_s \ddot{x}_s \tag{2.11}$$

Sisteme dik olan tüm kuvvetlerin toplamı, sistemin kütlesi ve bu yöndeki hızlanma ile doğru orantılıdır.

$$2F_x\cos\phi_t + 2F_y\sin\phi_t - M_s g\sin\phi_t - M_s L\ddot{\phi}_t = M_s \ddot{x}_s \cos\phi_t \tag{2.12}$$

Toplam momentler;

$$-2F_x L\cos\phi_t - 2F_y L\sin\phi_t - 2T = J_s \ddot{\phi}_t \tag{2.13}$$

(2.5) numaralı Tork denklemini (2.13) numaralı denklemde yerine yazıldığında;

$$-2F_x L\cos\phi_t - 2F_y L\sin\phi_t - 2\left(\frac{K_t V_G \eta}{R_a} - \frac{K_t K_{Ç} \eta^2 \dot{x}_s}{R_a r}\right) = J_s \ddot{\phi}_t \tag{2.14}$$

Denklemi elde edilir. (2.12) numaralı denkleminin her iki tarafı L uzunluğuyla çarpıldığında;

$$2F_xL\cos\phi_t + 2F_yL\sin\phi_t - M_sgL\sin\phi_t - M_sL^2\ddot{\phi}_t = M_s\ddot{x}_sL\cos\phi_t \tag{2.15}$$

$F_x\ ve\ F_y$ terimlerini (2.14) ve (2.12) numaralı denklemlerden çıkarmak için denklemler toplanır.

$$2\frac{K_tK_Ç\eta^2\dot{x}_s}{R_ar} - 2\frac{K_tV_G\eta}{R_a} - J_s\ddot{\phi}_t - M_sgL\sin\phi_t - M_sL^2\ddot{\phi}_t = M_s\ddot{x}_sL\cos\phi_t \tag{2.16}$$

(2.11) numaralı F_x denklemin (2.10) numaralı denklemde yerine yazılmasıyla F_x terimi ortadan kalkmış olur ve böylece yeni denklem;

$$2\left(m_t + \frac{J_t}{r^2}\right)\ddot{x}_s = \frac{2K_tV_G\eta}{R_ar} - \frac{2K_tK_Ç\eta^2\dot{x}_s}{R_ar^2} - \frac{2k_m\dot{x}_s}{r^2} - M_s\ddot{x}_s - M_sL\ddot{\phi}_t\cos\phi_t + M_sL\dot{\phi}_t^{\,2}\sin\phi_t \tag{2.17}$$

(2.16) ve (2.17) numaralı denklemler yeniden düzenlenirse;

$$\ddot{\phi}_t = 2\frac{K_tK_Ç\eta^2\dot{x}_s}{R_ar(J_s + M_sL^2)} - 2\frac{K_tV_G\eta}{R_a(J_s + M_sL^2)} - \frac{M_sgL\sin\phi_t}{(J_s + M_sL^2)} - \frac{M_s\ddot{x}_sL\cos\phi_t}{(J_s + M_sL^2)} \tag{2.18}$$

$$\ddot{x} = \frac{\frac{2K_tV_G\eta}{R_ar} - \frac{2K_tK_Ç\eta^2\dot{x}_s}{R_ar^2} - \frac{2k_m\dot{x}_s}{r^2} - M_sL\ddot{\phi}_t\cos\phi_t + M_sL\dot{\phi}_t^{\,2}\sin\phi_t}{\left(\frac{2m_tr^2 + 2J_t + M_sr^2}{r^2}\right)} \tag{2.19}$$

Denklemin lineerleştirilmesi için sarkaç sisteminin dik pozisyondan çok küçük θ_s açısı kadar uzakta olduğu varsayımı yapılır. Böylece;

$$\ddot{\theta}_s = \frac{M_sL}{(J_s + M_sL^2)}\ddot{x}_s + 2\frac{K_tK_Ç\boldsymbol{\eta}^2}{R_ar(J_s + M_sL^2)}\dot{x}_s - 2\frac{K_t\eta}{R_a(J_s + M_sL^2)}V_G + \frac{M_sgL}{(J_s + M_sL^2)}\theta_s \tag{2.20}$$

$$\ddot{x}_s = \frac{\frac{2K_t\eta}{R_ar}V_G - \left(\frac{2K_tK_Ç\eta^2}{R_ar^2} + \frac{2k_m}{r^2}\right)\dot{x}_s + M_sL\ddot{\theta}_s}{\left(\frac{2m_tr^2 + 2J_t + M_sr^2}{r^2}\right)} \tag{2.21}$$

Matlab’de lineerleştirilmiş denklemin jakobiyenini alındığında sistemin durum uzay matris gösterimi aşağıdaki gibi olmaktadır;

$$\begin{bmatrix}\dot{x}_s\\ \ddot{x}_s\\ \dot{\theta}_s\\ \ddot{\theta}_s\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & \left(\frac{2K_tK_ç\eta^2}{R_ar^2} + \frac{2k_m}{r^2}\right)\left(\frac{M_sLr - J_s - M_sL^2}{\propto}\right) & \frac{M_sgL^2}{\propto} & 0\\ 0 & 0 & 0 & 1\\ 0 & \frac{2K_tK_ç\eta^2}{R_ar^2}\left(\frac{r\beta - M_sL}{\propto}\right) & \frac{M_sgL\beta}{\propto} & 0\end{bmatrix}\begin{bmatrix}x_s\\ \dot{x}_s\\ \theta_s\\ \dot{\theta}_s\end{bmatrix} + \begin{bmatrix}0\\ \frac{2K_t\eta(J_s - M_sL^2 - M_sLr)}{R_ar\propto}\\ 0\\ \frac{2K_t\eta(M_sL - r\beta)}{R_ar\propto}\end{bmatrix}V_G \tag{2.22}$$

$$\propto = J_s\beta + 2M_sL^2\left(m_t + \frac{J_t}{r^2}\right) \tag{2.23}$$

$$\beta = \left(2m_t + \frac{2J_t}{r^2} + M_s\right) \tag{2.24}$$

3. Kontrol Yöntemleri

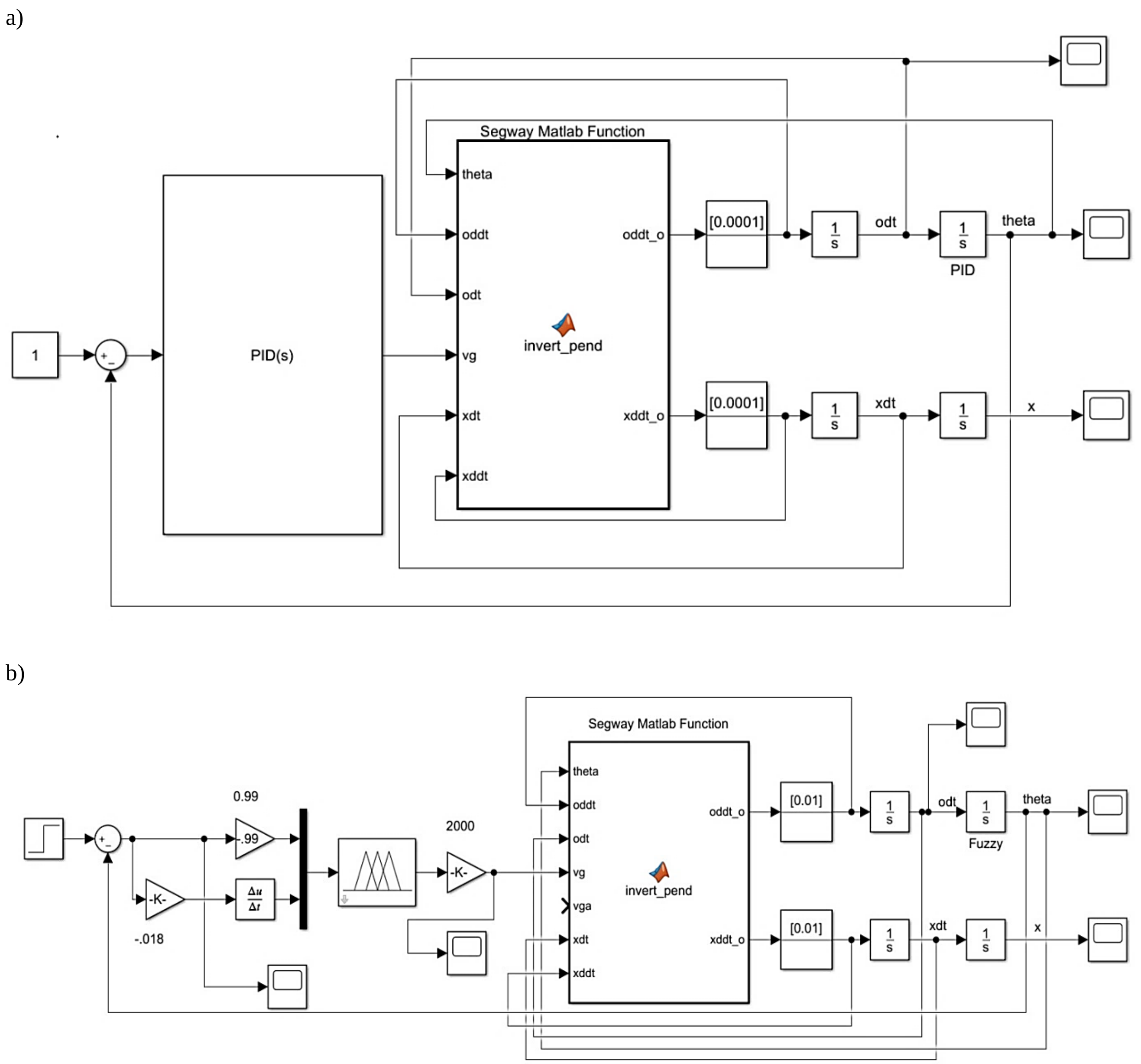
3 modlu denetleyici olarak da ifade edilen PID denetleyicisi, oransal, türevsel ve integral yöntemlerinin bir araya gelmesiyle elde edilmektedir. Türev bileşeni genellikle hata sinyalini daha önceden tahmin etmek, salınımları azaltmak için kullanılırken integral bileşeni ise oransal ofseti azaltmak ve sıfırlamak için kullanılır. Kısaca PID sistemin çıkışı ile referans değeri arasındaki hata farkının sıfır olmasını sağlamaktadır [15].

Hata fonksiyonu, türev bileşeni, integral bileşeni ve oransal bileşen ile ayrı ayrı çarpılır ve bunun sonucunda elde edilen veriler toplanır. Böylelikle PID katsayıları elde edilir. PID kontrolcünün en verimli şekilde çalışabilmesi için kazanç değerlerinin hesaplanması gerekmektedir.

K_p, K_i, K_d değerleri Matlab Simulink PID kontrolcünün içerisinde bulunan Tune kısmından hesaplanır. PID kontrolcü için hesaplanan değerler;

- K_p = 64.827
- K_i = 144.085
- K_d = 6.102

Elde edilen matematiksel modelin Matlab Simulink yardımıyla, PID kontrol yöntemiyle kontrol edilebilmesi için kazanç değerleriyle birlikte Şekil 3.a’ da gösterilen blok diyagramı kurulur.

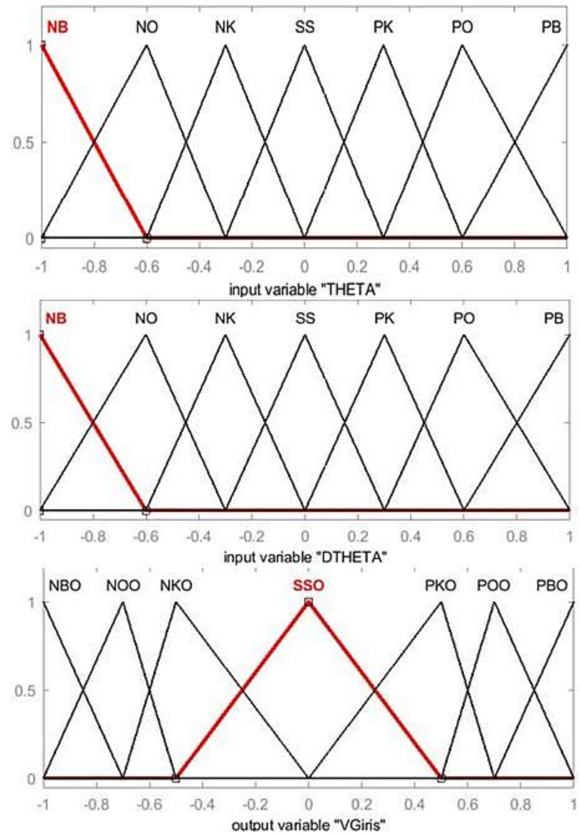


Şekil 3. Kontrol yöntemleri için blok diyagramları **(a)** Lineer sistem için kapalı çevrim PID kontrolcü blok diyagramı, **(b)** Bulanık mantık blok diyagramı

Çalışma kapsamında bir diğer kontrol yöntemi olarak bulanık mantık kullanılmıştır. İki tekerlekli kendini dengeleyebilen robotik sistem için önemli olan çıkış θ' dır. Bulanık mantık kontrolde θ' nın hata sinyalini belirli değerlerle çarpılması gerekmektedir. Bu değerler PID kontrolde program tarafından hesaplanırken bulanık mantıkta deneme yanılma yöntemleriyle hesaplanmaktadır. Bulanık mantık kontrol sistemi için oluşturulan blok diyagramı Şekil 3.b'de gösterilmiştir.

Bulanık mantık için giriş ve çıkış üyelik fonksiyonları ise Şekil 4' de gösterilmiştir.

Bulanık mantık, bulanık küme kuramına dayanır. Klasik kümeler sisteminde 0 ve 1 değerlerini alırken bulanık mantık sisteminde 0 ve 1 olduğu gibi 0 ve 1 arasındaki değerler de vardır. Yani klasik küme sisteminde bir elaman bir kümeye ya tamamen aittir (1) ya da tamamen hariçtir (0). Bulanık mantık sisteminde ise bir eleman belirlenen derecelere göre birçok kümeye ait olabilmektedir. Örneğin klasik kümeler sisteminde insan ya uzundur ya kısadır. Bulanık mantık sisteminde ise insan derecelerine göre 0.7 değeriyle uzun insanlar kümesine 0.3 değeriyle kısa insanlar kümesine girebilmektedir. Burada önemli olan sayıların alt ve üst limitleri arasındaki değerleri NB(Negatif Büyük), NO(Negatif Orta), SS(Sıfır), PO(Pozitif Orta) ve PB(Pozitif Büyük) gibi kümelerle kullanmaktır. Bu değerler ve kurallar Yoo ve ark. tarafından tablo haline dönüştürülerek Tablo 1'de gösterilmiştir [16].



Şekil 4. Bulanık mantık üyelik fonksiyonları

Tablo 1. Bulanık mantık kontrolcü kural tablosu [16]

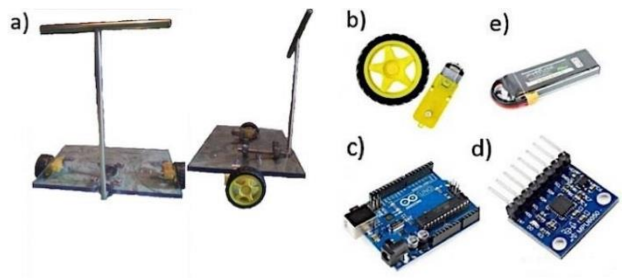
	$e_{\dot{\theta}}$						
e_θ	NB	NO	NK	SS-SR	PK	PO	PB
NB	NB	NB	NO	NO	NK	NK	SS
NO	NB	NO	NO	NK	NK	SS	PK
NK	NO	NO	NK	NK	SS	PK	PK
SS-SR	NO	NK	NK	SS	PK	PK	PO
PK	NK	NK	SS	PK	PK	PO	PO
PO	NK	SS	PK	PK	PO	PO	PB
PB	SS	PK	PK	PO	PO	PB	PB

4. Sistemi Oluşturan Ekipmanlar ve Maliyet Hesaplaması

Hazır sistemlerin pahalı olması sebebiyle bu sistemlerin daha uygun yollarla imal edilebilmesi için birçok araştırma yapılmıştır. Arduino Uno anakartı, ivmeölçer sensörü ve motor sürücüsünün kullanılmasıyla tüm işlemler basit hale gelmiştir. İvmeölçer sensöründen alınan bilgiler Arduino ana kartında işlenerek motor sürücülerine bilgi gönderir ve motorların tahriki sağlanır. Gidon sayesinde sistem sola ve sağa döner. Çalışma kapsamında tasarlanan iki tekerlekli kendini dengeleyebilen robotik sistem toplam 3 kg ağırlığa sahiptir.

4.1. Şase ve Malzeme Özellikleri

Şase Şekil 5.a'da gösterildiği gibi 10 mm kalınlığında 250mm x 300mm makrolon malzemeden yapılmıştır. Makrolon malzeme şeffaf olması, darbe dayanımı yüksek olması, camdan daha hafif olmasından dolayı tercih edilmiştir. Tekerlekler dengeyi sağlayabilmek için makrolonun ortasından bağlanmıştır. Gidon Ø10 mm civa çeliğinden kaynaklı bir şekilde tasarlanmıştır. Gidonun makrolona bağlanması 2 adet P000 numara ile adlandırılan sabit küre bilyalı yataklarla sağlanmıştır.



Şekil 5. Kullanılan Ekipmanlar **(a)** Robotik sistem şasesi, **(b)** DC motor ve tekerlek, **(c)** Arduino uno anakartı, **(d)** MPU 6050 sensör, **(e)** Li-Po pil

4.2. Motor ve Özellikleri

Prototip üretimi için Şekil 5.b'de gösterilen DC motorlar kullanılmıştır. DC motorlar, sistemin ağırlığı ölçüldükten sonra bu ağırlığı kaldırabilecek torku üretebilen ve piyasada kolaylıkla bulunan motorlar arasından seçilmiştir. Seçilen DC motor parametreleri Tablo 2'de gösterilmiştir.

Tablo 2. DC motor paremetreleri

Çalışma Gerilimi	3-12 V DC
Redüksiyon Oranı	1:48
Hız	250 rpm (@6V)
Akım	95mA-160mA
Ağırlık	29g
Tork	0.6Nm

4.3. Elektronik Ekipmanlar ve Özellikleri

İki tekerlekli kendini dengeleyebilen robotik sistemin sorunsuz bir şekilde çalışabilmesi için Arduino Uno anakartı kullanılmış olup sistemin beyni olarak da düşünülmüştür. Şekil 5.c'de gösterilen Ardunio sitemin verilerini işlemek ve buna göre gereken kararları almak için kullanılmıştır.

Ayrıca sistemin konum bilgisini alıp tekerleklere hareket vermesi için Şekil 5.d'de gösterilen MPU 6050 6 eksen ivme ve jiroskop sensörü kullanılmıştır.

Şekil 5.e'de gösterilen iki tekerlekli kendini dengeleyebilen robotik sisteme hareket 11.1V 850 mAh Li-Po pillerle sağlanmıştır.

4.4. Maliyet Hesaplaması

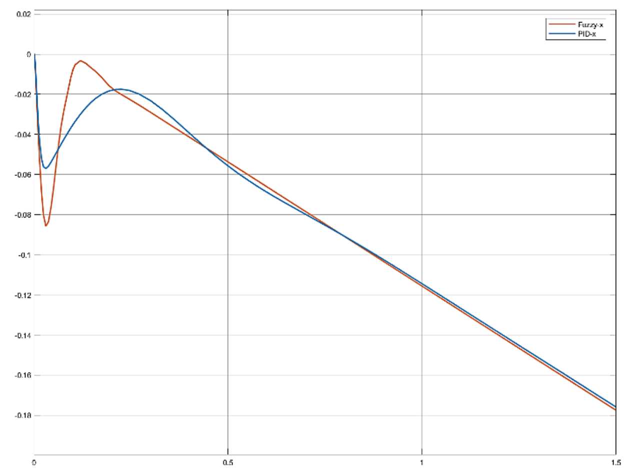
İki tekerlekli kendini dengeleyebilen robotik sistemin prototip üretimi için kullanılan ekipmanların fiyat bilgilendirmesi Tablo 3'de detaylı olarak belirtilmiştir.

Tablo 3. Sistemin maliyet bilgisi

Malzeme	Adet	Fiyat
Makrolon Şase	1	30 TL
Gidon	1	20 TL
DC Motor + Tekerlek	2	40 TL
Anakart	1	50 TL
İvme Ölçer Sensör	1	15 TL
Sabit Küre Bilyalı Yatak	2	70 TL
Pil	1	200 TL
Toplam Maliyet		425 TL

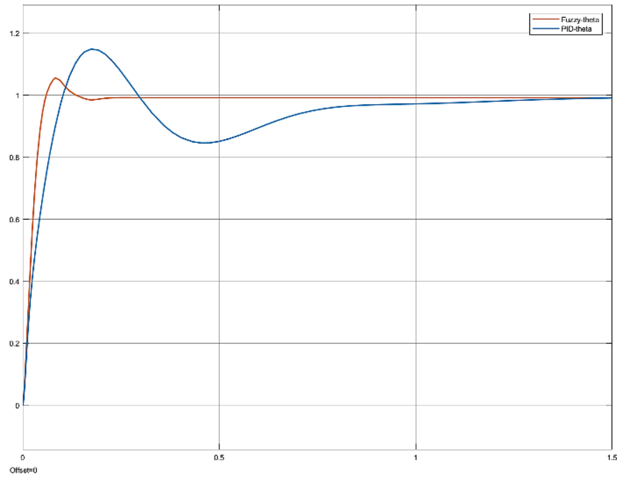
5. Sonuçlar ve Tartışma

Bu çalışmada, iki tekerlekli kendini dengeleyebilen robotik sistemin matematiksel modeli ve lineer denklemleri, Kirchhoff voltaj ve Newton hareket yasaları kullanılarak elde edilmiştir. Çalışma esnasında sistemin elde edilen matematiksel modeline Matlab Simulink ortamında açı geri beslemesine sahip PID ve bulanık mantık kontrolcüsü uygulanmıştır. Her iki kontrolcü içinde elde edilen sonuçlar konum, zaman ve ivme grafikleri değerlendirilerek ele alınmıştır.



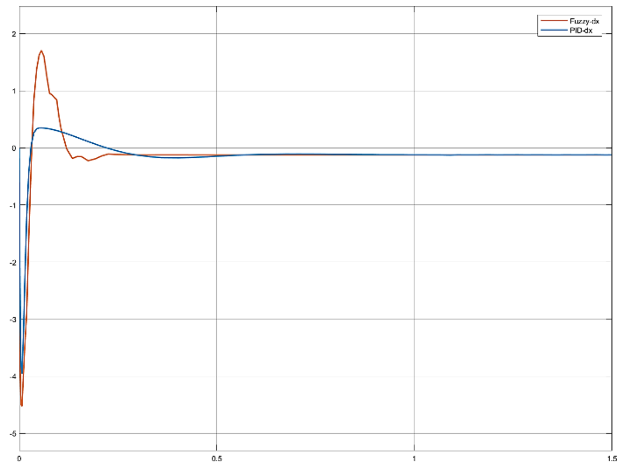
Şekil 6. PID – bulanık mantık lineer konum

Şekil 6'da sistemin PID ve bulanık mantık kontrolcü ile lineer konumundaki hareketinin sonucu gösterilmiştir; yani zaman arttıkça sonsuza yaklaşmakta olduğu görülmektedir. Bunun sebebi sisteme enerji verdiğimizde başlangıç noktasında sistem kendiliğinden dengede duramadığı için o an ki konumundan düşmekte olduğunu görülmektedir; fakat sistemi dengeye getirmek için gerekli şartlarda çalıştırıldığında sistemin konum değişimine daha doğru cevaplar verdiği görülmektedir.



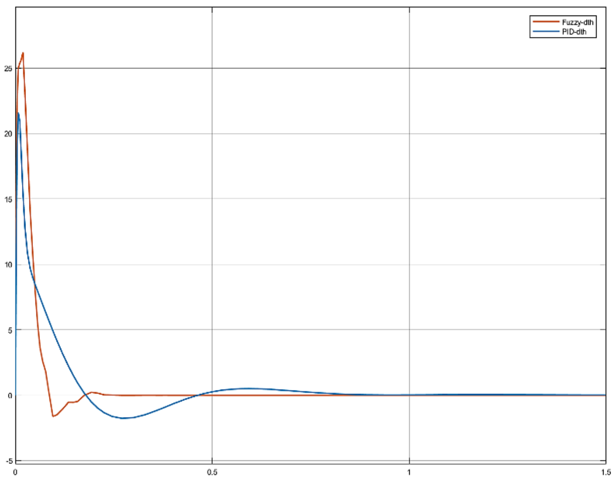
Şekil 7. PID – bulanık mantık açısal konum

Şekil 7'de sistemin PID ve bulanık mantık kontrolcü ile açısal konumundaki hareketinin sonucu gösterilmiştir. Bulanık mantık kontrolcüsünün 0.25s'de sistemi denge pozisyonuna getirdiği ancak PID kontrolcünün bulanık mantık kontrolcüye göre 1.25s geciktiği ve 0.2° açı salınımlarıyla kendini dengeye getirdiği gözlemlenmiştir. Şekil 7'de daha net bir şekilde fark edildiği üzere PID kontrolcü sisteminde daimi rejim hatasının olduğu ve yük altında titreme yapabileceği görülmektedir.



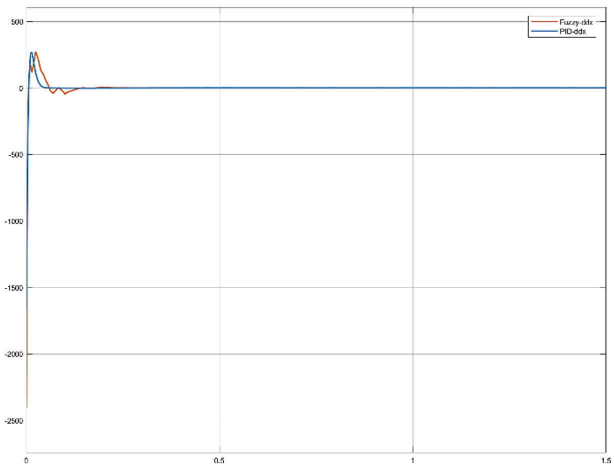
Şekil 8. PID – bulanık mantık lineer hız

Şekil 8'de sistemin PID ve bulanık mantık kontrolcü ile lineer hız grafiği gösterilmiştir. Bulanık mantık kontrolcünün tepe noktası PID kontrolcüye göre oldukça fazladır. Bu durumdan dolayı sistemi dengede tutabilmek için daha fazla enerji harcamak zorundadır. Her iki kontrolcüde de oturma zamanı ve yükselme zamanı aynı olduğu ama PID kontrolcünün kendini dengeye getirebilmesi için harcadığı enerji miktarının bulanık mantık kontrolcüye göre daha az olduğu Şekil 8'de görülmektedir. Böylelikle ani hareketlenme gibi durumlarda PID kontrolcüsü tercih edilir.



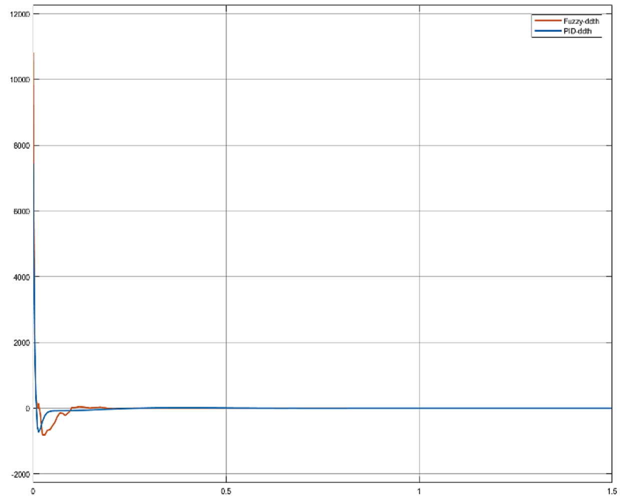
Şekil 9. PID – bulanık mantık açısal hız

Yukarıda gösterilen Şekil 9'da sistemin PID ve bulanık mantık kontrolcü için elde edilen açısal hız grafiği gösterilmiştir. PID kontrolcü ile bulanık mantık kontrolcünün tepe noktalarının farklı olduğu, öte yandan PID ve bulanık mantık kontrolcünün yükselme zamanının birbirleriyle hemen hemen aynı olduğu görülmektedir. Ancak bulanık mantık kontrolcünün oturma zamanı 0.3s iken PID kontrolcünün oturma zamanı 1 s'leri bulmaktadır. Şekil 9'dan anlaşıldığı üzere yüksek hızlarda bulanık mantık kontrolcüsü PID kontrolcüye göre daha iyi performans göstermektedir.



Şekil 10. PID – bulanık mantık lineer ivme

Şekil 10'da sistemin PID ve bulanık mantık kontrolcü ile lineer ivme grafiği gösterilmiştir. Her iki kontrolcüde de tepe noktalarının, yükselme zamanlarının ve oturma zamanlarının aynı olduğu gözlemlenmiştir; ancak tepe noktalarının çok yüksek değerlere sahip olması ani ivmelenmelerden dolayı sistemin çok fazla enerji harcamasına neden olmaktadır.



Şekil 11. PID – bulanık mantık açısal ivme

Şekil 11'de sistemin PID ve bulanık mantık kontrolcü ile açısal ivme grafiği gösterilmiştir. Her iki kontrolcüde de tepe noktalarının, yükselme zamanlarının ve oturma zamanlarının aynı olduğu gözlemlenmiştir. Açısal ivmenin daha fazla olması sistemin yavaşlamalarında ve hızlanmalarında oldukça önemlidir ancak ivmenin çok yüksek olması sistemi kazaya götürebilir. Bu yüzden optimal olması istenmiştir.

Tasarlamış olduğumuz iki tekerlekli kendini dengeleyebilen robotik sistemin her iki kontrolcü tarafından da kontrol edilebildiği görülmüştür. Açısal konum ve lineer ivme sistemi kontrol edebilmek için önem taşımaktadır. Şekil 7'den elde edilen sayısal veriler Tablo 4 açısal konum satırında, Şekil 10'dan elde edilen veriler Tablo 4 te lineer ivme satırında gösterilmiştir.

Bulanık mantık kontrolcüsü Mamdani – Centroid algoritmasıyla tasarlanmıştır ve bulanık mantık kontrolcüsüne ne kadar fazla kural girilirse sistemin o kadar kararlı hale geldiği tespit edilmiştir. Şekil 7'de görüldüğü üzere PID kontrolcüsünün yükselme zamanı bulanık mantık kontrolcüsüne göre daha fazladır. Her iki kontrolcünün de sinyalinin tepe noktası hemen hemen aynıdır. PID kontrolcüsü bulanık mantık kontrolcüsüne göre yaklaşık 1.25 sn geç oturmaktadır. Bulanık mantık kontrolcüsünün PID kontrolcüye göre daha hızlı ve daha az salınımla cevap verdiği görülmüştür ayrıca daha az hareket ederek denge kontrolü sağladığı Tablo 4 açısal konum satırında ve Şekil 7' de görülmüştür. Aynı zamanda bulanık mantık kontrolcü PID kontrolcüye göre referans açıyı daha iyi takip etmiştir. İki tekerlekli kendini dengeleyebilen robotik sistemin, bulanık mantık kontrolör verilerinin Matlab ortamında elde edilen sonuçlarının, sistemin denge ve konum kontrolünde başarılı olduğu Tablo 4'de anlaşılmıştır.

Tablo 4 lineer ivme satırından görüldüğü üzere PID kontrolcüsünün yükselme zamanı bulanık mantık kontrolcüsüne göre daha kısadır. Her iki kontrolcünün de sinyalinin tepe noktası aynıdır. Bulanık mantık kontrolcüsü PID kontrolcüsüne göre yaklaşık 0.2 sn geç oturmaktadır. Bulanık mantık kontrolcüsünün yüksek hız ve ivmelerde salınımla cevap verdiği ve aynı zamanda yüksek lineer ve açısal hız ile yüksek lineer ve açısal ivme gerekli durumlarda PID kontrolcünün bulanık mantık kontrolcüye göre üstün olduğu Tablo 4 lineer ivme satırında ve Şekil 10'da görülmüştür. Matematiksel modeli çıkartılan ve dinamiği hakkında yeterli bilgi sahibi olunan bir sistemin, PID kontrol verilerinin Matlab ortamında oluşturulmasıyla daha kısa sürede tasarımı yapılabileceği ve bulanık mantık kontrolcüye kıyasla sistemin kontrolünde oturma zamanının geç olduğu sonucuna ulaşılmıştır.

Tablo 4. PID – bulanık mantık kontrolcülerinin açısal konum ve lineer ivme için karşılaştırma sonuçları

	Kontrol Metodu	Yükselme Zamanı	Oturma Zamanı
Açısal Konum	PID	0.25s	1.5s
	Bulanık Mantık	0.15s	0.25s
Lineer İvme	PID	0.02s	0.05s
	Bulanık Mantık	0.04s	0.25s

Sonuç olarak, bulanık mantık kontrolcüsünün açısal konum kontrolünde PID kontrolcüye göre daha iyi olduğu ortaya çıkmıştır. Öte yandan PID kontrolcünün de yüksek hız ve ivme kontrolünde bulanık mantık kontrolcüsüne göre daha iyi sonuçlar verdiği gözlemlenmiştir. Bu çalışma kendiliğinden dengelenebilen iki tekerlekli robotik sistemlerin PID veya bulanık mantık kontrol yöntemlerinden biriyle kontrol edildiğinde konum, hız ve ivme bakımından hangisinin daha iyi sonuç verebileceğine ve tercih edilebileceğine aynı zamanda gelecekte bu alanda yapılacak çalışmalara da ışık tutacaktır.

Terimler Dizisi

θ_s :Sistemin Açısal Pozisyonu
ϕ_t :Tekerleklerin Açısal Pozisyonu
ϕ_m :Motorun Açısal Pozisyonu
x_s :Sistemin x Eksenindeki Lineer Yer Değiştirmesi
x_t :Tekerleklerin x Eksenindeki Lineer Yer Değiştirmesi
M_s :Sistemin Kütlesi
m_t :Tekerleklerin Kütlesi
J_s :Sistemin Eylemsizlik Momenti
J_t :Tekerleklerin Eylemsizlik Momenti
r :Tekerleklerin Yarıçapı
g :Yerçekimi İvmesi
a :İvme
k_m :Motorun ve Motor Şaftına Atfedilen Yükün Sürtünme Katsayısı
L :Sarkaç Kol Uzunluğu
F_x :x Eksenindeki Kuvvet
F_y :y Eksenindeki Kuvvet
F_s :Sürtünme Kuvveti
F_N :Normal Kuvvet
T :Tork
η :Redüktör Oranı
T_m :Motor Torku
T_s :Sürtünmeden Dolayı Oluşan Tork
$K_Ç$:Geri EMF Sabiti
K_t :Tork Sabiti
R_a :Motor Armatür Direnci
L_a :Motor Armatür İndüktansı
i_a :Motor Armatür Akımı
V_G :Giriş Voltajı
$V_Ç$:Çıkış Voltajı
U_t :Kontrol Değişkeni
K_p :Oransal Kazanç
K_i :İntegral Kazancı
K_d :Türev Kazanç

Kaynaklar

[1] Grasser F., D'Arrigo A., Colombi S., Rufer A. C., 2002. JOE: a mobile, inverted pendulum. IEEE Trans. Ind. Electron., 49(1), 107–114.

[2] Anderson D. P., nBot, a two wheel balancing robot. http://www.geology.smu.edu/~dpa-www/robo/nbot/ (Ziyaret tarihi: 30 Mart 2018).

[3] Hassenplug S., Steve's LegWay. http://www.teamhassenplug.org/robots/legway/. (Ziyaret tarihi: 30 Mart 2018)

[4] Umay Y., 2018. İki Tekerlekli Kendini Dengeleyebilen Mobil Bir Aracın Kontrolü. Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri, Elazığ, 523375

[5] Pathak K., Franch J., Agrawal S.K., 2005. Velocity and position control of a wheeled inverted pendulum by partial feedback linearization. IEEE Transactions on Robotics, 21(3), 505-513.

[6] Charles E. Forrest Jr., 2006. A Neural Network Control System for the Segway Robotic Mobility Platform. Thesis of Master of Graduate Faculty of North Carolina State University.

[7] Kim Y., Kim S.H., Kwak Y.K., 2006. Improving Driving Ability for a Two-Wheeled Inverted-Pendulum-Type Autonomous Vehicle. Proceedings of the IMechE Part D Journal of Automobile Engineering, 220(2), 165-175.

[8] Nawawi S.W., Ahmad, M.N., Osman J.H.S., 2007. Development of a two-wheeled inverted pendulum mobile robot. 2007 5th Student Conference on Research and Development, Selangor, Malaysia, 2007, pp. 1-5.

[9] Jeong S.H., Takahashi T., 2007. Wheeled inverted pendulum type assistant robot: inverted mobile, standing, and sitting motions. 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, 2007, pp. 1932-1937.

[10] Li Z., Xu C., 2009. Adaptive Bulanık mantık control of dynamic balance and motion for wheeled inverted pendulums. Fuzzy Sets and Systems, 160(12), 1787 - 1803.

[11] Chung-Neng Huang., 2010. The Development of Self-Balancing Controller for One-Wheeled Vehicles, Published Online April 2010 (http://www.SciRP.org/journal/eng).

[12] Li J., Gao X., Huang Q., Du Q., Duan X., 2007. Mechanical design and dynamic modeling of a two-wheeled inverted pendulum mobile robot. 2007 IEEE International Conference on Automation and Logistics, Jinan, China, 2007, pp. 1614-1619.

[13] Hasan M., Saha C., Rahman M., Sarker M., Aditya S., 2012. Balancing of an Inverted Pendulum Using PD Controller. Dhaka University Journal of Science, 60(1), 115-120.

[14] Xu J.X., Gao Z.Q., Lee T. H., 2014. Design and Implementation of Integral Sliding-Mode Control on an Underactuated Two-Wheeled Mobile Robot. IEEE Trans. Ind. Electron., 61(7), 3671–3681.

[15] Polat B.Denge Robotu Tasarımı ve Modellenmesi, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri, Elazığ, 2018,524841

[16] Ho Yoo H.and Jae Choi B., 2015. Design of Simple-Structured Fuzzy Logic Systems for Segway-Type Mobile Robot. IJFIS, 15(4), 232-239.