

Dört Bacaklı Robotun Gerçek Zamanlı PID Kontrolü

¹İ. Hüseyin GÜVENÇ, ¹Ömer YENTÜR, ¹Hasan YILDIZ, ¹Mutlu KINCI ve ¹H. Metin ERTUNÇ
¹Mekatronik Mühendisliği Bölümü, Kocaeli Üniversitesi, Umuttepe Kampüsü, Türkiye

Özet

Bu çalışmada son dönemde teknoloji dünyasında örneklerini sıklıkla gördüğümüz 12 aktüatörden oluşan dört bacaklı robot tasarımı yapılmıştır. Robot tasarımında ileri ve ters kinematik hesaplamaları kullanılırken robotun yürüyüş kontrolü PID ile gerçekleştirilmiştir. Kinematik denklemi üç serbestlik dereceli robot kol analizinin denklemleri çıkartılmıştır. Bu denklemlere göre bir sürücü yazılarak servo motorların kontrolü sağlanmıştır. Sistemde işlem kapasitesi ve uzaktan kontrol imkânı sebebiyle JETSON Nano kartı kullanılmıştır. Robotun kontrolü için basit bir mobil uygulama tasarlanarak Wi-Fi üzerinden cep telefonu ile kontrol edilebilir hale getirilmiştir. Sistemin yazılım geliştirmesi safhası için GAZEBO programı üzerinden simülasyonu yapılarak sistem optimize edilmiştir. Aynı zamanda sistem ROS (Robot İşletim Sistemi) yazılımı ile gerçekleştirilerek kontrolü daha yalın hale getirilerek kullanımı kolaylaştırılmıştır.

Anahtar Kelimeler: 4 bacaklı Robot, Yürüyüş (Gait), PID Kontrol, Kinematik Modelleme.

Abstract

In this study, a four-legged robot design consisting of 12 actuators, examples of which we have often seen in the technology world in the recent period, has been made. While forward and reverse kinematics calculations were used in the robot design, the gait control of the robot was performed with PID. The kinematic equation of the robot arm analysis with three degrees of freedom equations are extracted. According to these equations, a drive is written and servo motors are controlled. The JETSON Nano card was used in the system due to its processing capacity and remote control capability. A simple mobile application has been designed to control the robot and it has been made controllable by mobile phone via Wi-Fi. The system has been optimized by simulating the system through the GAZEBO program for the software development phase. At the same time, the system is realized with ROS (Robot Operating System) software, making the control simpler and making it easier to use

Key words: Quadruped Robot, Gait, PID Control, Kinematic Modeling.

1. Giriş

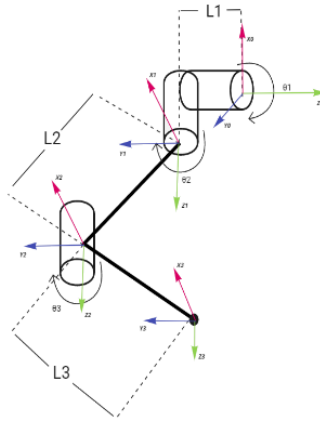
Robotların günümüz dünyasında kullanım alanı oldukça artmış durumdadır. Öyle ki hayvanlara benzer robotların modellemeleri ve kullanımı mümkün hale gelmiştir [1]. Bu bağlamda robotlar çeşitli insan görevlerini yapabilecek durumdadır. Bunun yanında insanlarla iletişim halinde olabilen dört bacaklı hayvansı robotlarda günümüzde bir hayli popülerdir [2].

Bu çalışmada dört bacaklı robot tasarlanmıştır. Öncelikle gerekli araştırmalar ile teknik bilgi elde

edildikten sonra robot tasarımı ve kinematik hesaplamalar yapılmıştır [3,4]. Sonra bir test robotu üretilip hareket algoritmasının prototip ürüne aktarılması gerçekleştirilmiştir. Bu bağlamda ilk olarak servo motorlardan oluşan bir tasarım yapılmıştır. İlk üretim sonrasında karşılaşılan hataları düzeltmek amacıyla robotun ikinci ve üçüncü prototipleri 3 boyutlu yazıcı ile üretilmiştir.

2. Materyal ve Yöntem

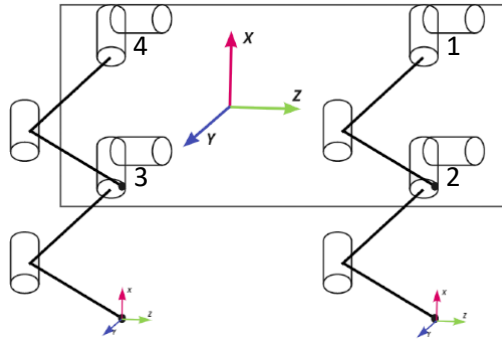
Dört bacaklı robot, rijit bir gövdeye bağlı dört bacak ve her bacakta bulunan 3 ekleme sahiptir. Eklemler dönel mafsallarla birbirine bağlanmıştır. Eklemlere tahrik veren aktüatörler servo motorlardır. Bu motorların her biri 35 kg/cm durak torkuna sahiptir. Şekil 1’de görüldüğü gibi bir bacak 3 eklemden oluşmuştur. Şekilde eklemlerin eksen yerleşimleri gösterilmiştir.



Şekil 1. Robotun Fiziksel Modeli

2.1 Denavit-Hartenberg Tablosu ve Parametrelerin Gösterimi

Dört bacaklı robotun kinematik olarak çözülmesi gereken başlıca problemi robotun bacak koordinatlarına bağlı olarak gövdesinin konfigürasyonlarının hesaplanmasıdır. Robotun gövdesinin ve ekleme uç noktalarının aksenal yerleşimleri Şekil 2’de gösterilmiştir. 12 DoF (serbestlik derecesine) sahip olan robotun Denavit-Hartenberg [3] parametreleri Tablo 1’de listelenmiştir.



Şekil 2. Robotun Dörtlü Fiziksel Modeli ve Aksenal Yerleşimleri

Tablo 1. Denavit-Hartenberg Parametreleri

Link	α_{i-1}	a_{i-1}	d_i	θ_i
0-1	0	0	0	θ_1
1-2	$-\pi/2$	0	L_1	θ_2
2-3	0	L_2	0	θ_3
3-4	0	0	L_3	0

İlk olarak çalışma alanı olarak belirlenen alanda yani Y_1 , Y_2 ve Y_3 açılarının gövdeye göre Y ekseninde 0° , 0° , -45° değerleri için hesaplanmıştır. Tablo 1'e göre çıkartılan Transformasyon matrisleri aşağıdaki denklemlerde gösterilmiştir.

$${}^0_1T = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$${}^1_2T = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & 0 \\ 0 & 0 & 1 & L_1 \\ -\sin(\theta_2) & -\cos(\theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$${}^2_3T = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & L_2 \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$${}^3_4T = \begin{bmatrix} 1 & 0 & 0 & L_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

2.2 İleri Kinematik Hesaplamaları

Robotun ileri kinematiği, robot bağlantılarının konumları, hızları ve ivmeleri arasındaki ilişki ile ilgilenir. Ters kinematik, robotun uç noktasının konum ve yön verilerine göre eklem değişkenlerinin değerlerini bulma işlemidir [4]. Yani robot uç noktasını istenilen konuma hareket ettirebilmek için mafsalların dönme değerlerinin ters kinematik analiz ile belirlenmesi gerekmektedir. Bacaklar birbiriyle farklı yönlerde fakat aynı yapıdadır. Bu nedenle tek bir bacağın ileri ve ters kinematik analizini araştırmak yeterlidir. İleri kinematik matrisi Eşitlik 5'te ve denklemleri Tablo 2'de verilmiştir.

$${}^0_4T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Tablo 2. İleri Kinematik Denklemleri

$r_{11} = \cos(\theta_1)\cos(\theta_2 + \theta_3)$	$r_{31} = -\sin(\theta_2 + \theta_3)$
$r_{12} = -\cos(\theta_1)\sin(\theta_3 + \theta_2)$	$r_{32} = -\cos(\theta_2 + \theta_3)$
$r_{13} = -\sin(\theta_1)$	$r_{33} = 0$
$r_{21} = \cos(\theta_2 + \theta_3)\sin(\theta_1)$	$p_x = L_3(\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) - \cos(\theta_1)\sin(\theta_2)\sin(\theta_3)) - L_1\sin(\theta_1) + L_2\cos(\theta_1)\cos(\theta_2)$
$r_{22} = -\sin(\theta_3 + \theta_2)\sin(\theta_1)$	$p_y = L_1\cos(\theta_1) - L_3(\sin(\theta_1)\sin(\theta_2)\sin(\theta_3) - \cos(\theta_2)\cos(\theta_3)\sin(\theta_1)) + L_2\cos(\theta_2)\sin(\theta_1)$
$r_{23} = \cos(\theta_1)$	$p_z = -L_3(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2)) - L_2\sin(\theta_2)$

2.2 Ters Kinematik Hesaplamaları

Dörtlü robotların yürüyüş (gait) hareketini otonom halde yapabilmesi için her bacağın uç noktası için bir yörünge belirlenmesi gerekmektedir. Bu konu ile ilgili ayrıntılara bir sonraki bölümde yer verilmiştir. Ters kinematik, robotun eklem uç noktalarının yörüngelerini hesaplayarak eklem o pozisyona gelmesi için kullanılır [5,6,7].

$${}^0T_4 = {}^0T_1{}^1T_2{}^2T_3{}^3T_4 \quad (6)$$

$$[{}^0T_1]^{-1}{}^0T_4 = {}^1T_2{}^2T_3{}^3T_4 \quad (7)$$

$$[{}^0T_1]^{-1}[{}^0T_4] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$${}^1T_2{}^2T_3{}^3T_4 = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

Tablo 3. Ters Kinematik A matrisi

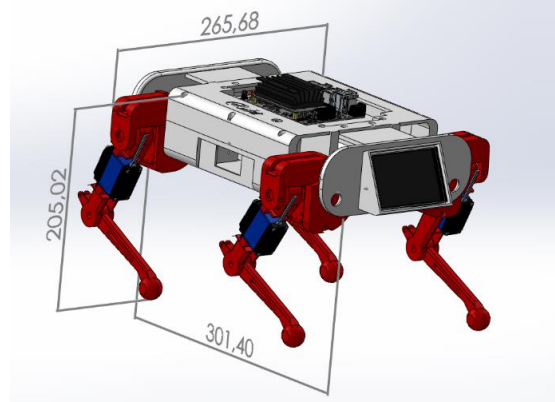
$a_{14} = p_x\cos(\theta_1) + p_y\sin(\theta_1)$
$a_{24} = p_y\cos(\theta_1) - p_x\sin(\theta_1)$
$a_{34} = p_z$

Tablo 4. Ters Kinematik B matrisi

$b_{14} = L_3\cos(\theta_2 + \theta_3) + L_2\cos(\theta_2)$
$b_{24} = L_1$
$b_{34} = -L_3\sin(\theta_2 + \theta_3) - L_2\sin(\theta_2)$

Ters kinematik problemlerinin çözümünde doğrusal olmayan denklemler vardır. Hesaplanan her matematiksel ifade için fiziksel bir çözüm olmayabilir. Ayrıca, bacak uç noktasının istenen pozisyona gitmesi için birden fazla çözüm olabilir. Bu nedenle, robotun (1 ve 3) bacakları ve robotun (2 ve 4) bacakları aynı kinematik yapıda, ancak farklı konfigürasyonlarda gerçekleştirilmiştir. Tablo 4'te görülen Y_1 , Y_2 ve Y_3 açıları bu denklemlerden elde edilir. Şekil 3'te robotun genel tasarımı resmedilirken, bacak uzunlukları ise aşağıdaki gibi verilmiştir:

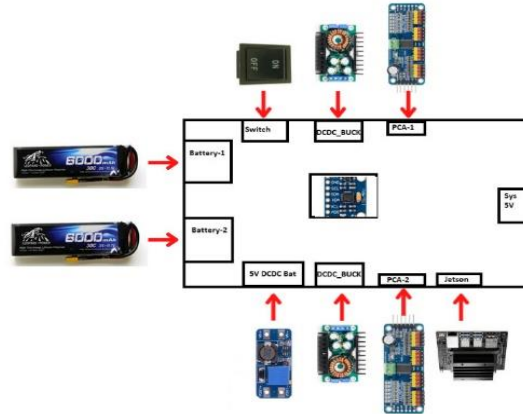
$$L_1 = 26 \text{ mm} , L_2 = 106 \text{ mm} , L_3 = 114 \text{ mm} .$$



Şekil 3. Robotun Genel Tasarımı

2.3. Sistemin Elektronik Tasarımı

Robotun elektronik sisteminde 12 adet servo motor, 2 adet Li-Po pil, 2 adet 9A güç modülü, 1 adet 5V güç modülü, 1 adet MPU6050 ivmeölçer, 2 adet PCA9685 servo motor sürücüsü, kontrolör olarak 1 adet JETSON Nano kullanılmıştır. Servo motorların kontrolü üzerinden çalışacak bir sistem hazırlandığı için bu motorların kinematik çıktılarıyla sürülmesi gerekmektedir. Servo motor sürücü devresi olarak kullanılan PCA9685 sürücü kartı I2C haberleşme protokolü ile kontrolör ile haberleşmektedir. İki adet sürücü kartına bağlı 12 adet motorun açılarını anlık olarak yörüngelere göre aktarması için motorlara anlık olarak açı verilerinin iletilmesi gerekmektedir [8]. Bu işlem aynı zamanda değişken olarak pilden çekilen akım miktarını da etkilemektedir. Pilden aşırı akım çekilmemesi ve istenilen gerilimin kusursuz şekilde sağlanması için ayarlanabilir DC-DC çevirici kullanılmıştır. Şekil 4’te bağlantılar gösterilmiştir. Motorların kontrolü sürücülerden sağlanan PWM sinyali ile senkronize bir şekilde sağlanmaktadır [9].



Şekil 4. Malzemelerin Yerleşimi

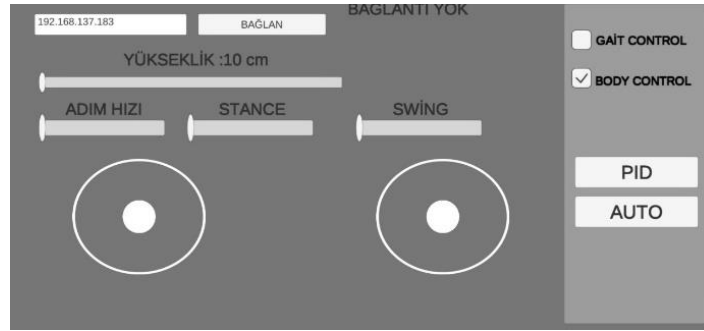
2.4. Sistemin ROS ile Yazılımsal Tasarımı

Sistemde kullanılan yazılım temelde JETSON üzerinde çalışır. Yazılım sistemini tasarlamak için ise ROS kullanılmıştır [15]. Robot İşletim Sistemi (ROS), robotik alanındaki araştırmacılar için çok kullanışlı ve güçlü bir platformdur. ROS, yazılım geliştiricinin hızını artırmada yardımcı olur. Ayrıca kodun uzunluğunun azalması nedeniyle algoritmayı yeniden kullanma yeteneği verir. ROS, düğümlerden oluşur Şekil 5'te görüldüğü gibi düğümler gösterilmiştir. ROS ile sistem temelde çalışması şöyledir: Düğümlerden oluşan yazılım ağında kontrol sağlamak için bir mobil uygulamadan hareket bilgileri Wi-Fi haberleşmesi ile alınmaktadır. Alınan yönelim ve hız bilgisini seçilen yürüyüş moduna göre kinematik olarak anlamlandırır [15,16], İstenilen hareket şekli düğümlerdeki ilgili bölüm aracılığıyla devreye alınmaktadır. Böylece ROS sisteminin kullanımı çok yönlü ve hızlı şekilde gerçekleşmektedir. Şekil 5'te görülen ROS düğüm ağacı temsili olarak oluşturulmuştur.

```
Folder PATH listing for volume HDD_DISK
Volume serial number is 30E9-CB4E
E:.
|--src
|   |--Robit_Controller
|   |   |--GaitController.py
|   |   |--key_board_control.py
|   |   |--PIDController.py
|   |   |--RobotController.py
|   |   |--robot_controller_gazebo.py
|   |   |--TrotGaitController.py
|   |--Robit_Utillities
|   |   |--Calibration.py
|   |   |--JointDrive.py
|   |   |--robot_Inverse_Kinematic.py
|   |   |--Transformations.py
|   |--Robot_Peripheral_Units
|   |   |--imu_node.py
|   |   |--PCA9685.py
|   |   |--registers.py
|   |   |--ServoMotor.py
|   |   |--Wifi_Communication.py
```

Şekil 5. ROS Node Ağacı

Aynı zamanda bahsedilen mobil uygulamanın arayüzü Şekil 6'tadır. Arayüz sayesinde ROS ile yazılımdaki herhangi bir parametrenin ayarlaması yapılabilmektedir. Bu yürüyüş modu, adım uzunluğu ve yüksekliği, PID ve kinematik kontrol yöntemlerini içermektedir [18].

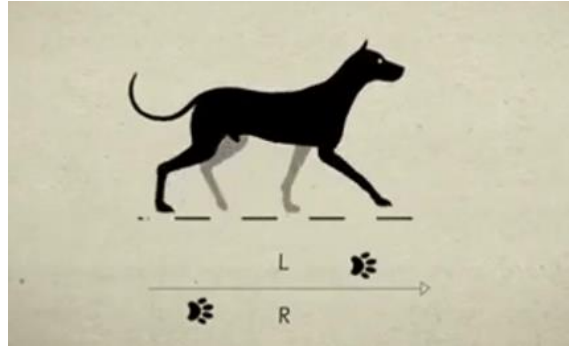


Şekil 6. Mobil Uygulama Arayüzü

2.4.1. Robot Yürüyüş Planlaması ve Vücut Kontrolü

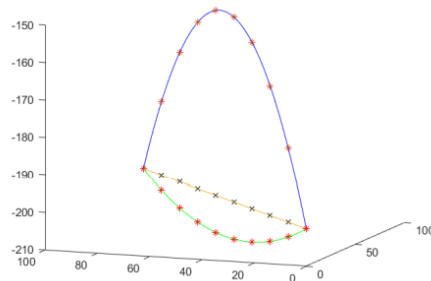
Kinematik yürüyüş (Gait), dengeyi ve istikrarı korumaya yardımcı olan vücut hareketini içeren 4 fazlı bir yürüyüştür. Çapraz bacakların aynı anda hareket ettiği alternatif bir hızlı yürüyüş biçimidir

[19]. Daha hızlı yürüyüş hızları elde edebilir, ancak daha az kararlıdır ve robotun kütle merkezinin dikkatli bir şekilde konumlandırılmasını gerektirir.



Şekil 7. Yürüyüş Modeli

Yörünge planlamada kullanılan en başlıca yöntem bir matematiksel eğriyi takip etmektir [20]. Şekil 8’de görülen eğri basit bir polinom grafiğidir. Grafikte gösterilen kırmızı noktalar bacağın uç noktasının gitmesi gereken konumlardır. Robot ileri hareket komutu aldığı anda adım atacağı noktaları polinom grafiğindeki gibi hesaplayarak motor açılarını o noktaya göre ayarlamaktadır. Böylelikle ayağın uç noktası öncelikle yükselir tekrar yere değene kadar yörüngeyi izler ve yere değdiğinde ise yine benzer bir grafik üzerindeki noktaları izlemektedir. Burada yere temas söz konusu olduğunda yere kuvvet uygulayarak ileri ivmelenme gerçekleşmektedir [21]. Bu eğri matematikte Bezier eğrisi olarak tanımlanmaktadır.



Şekil 8. Planlanan Yörünge Örneği

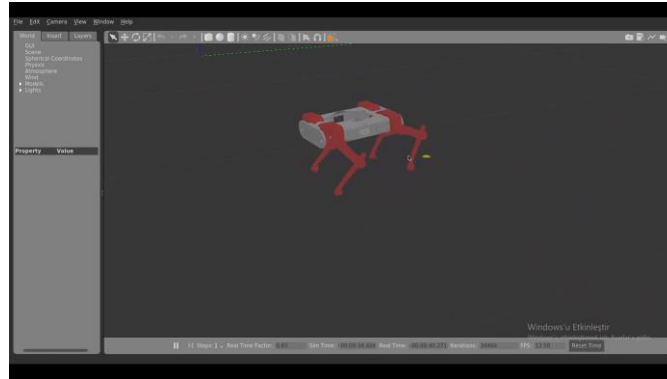
Robot dört omuzuyla çeşitli hareketler yapabilir. Bunlar sağ ve sol eğim, ileri ve geri eğimdir. Bunlar, omuzları ve eklemleri belirli bir yörüngede Şekil 9’da gösterildiği gibi hareket ettirmek için yapılıdır.



Şekil 9. Vücut Kontrol Hareketleri

2.4.2. Robotun GAZEBO ile Simülasyonun Yapılması

Robotun hareket ve yürüyüş testlerini yapmak fiziksel ortamda çok maliyetli olmaktadır. Bu sebeple GAZEBO ortamına tasarımlar aktarılarak simülasyon üzerinden yazılım geliştirilmiştir [18]. GAZEBO'ya erişim sağlamak için ROS kullanılmaktadır [22]. Şekil 10'da robotun GAZEBO ortamında çalışması gösterilmiştir.



Şekil 10. GAZEBO Simülasyon Ortamı

2.5. Sistemin Kontrolör Tasarımı

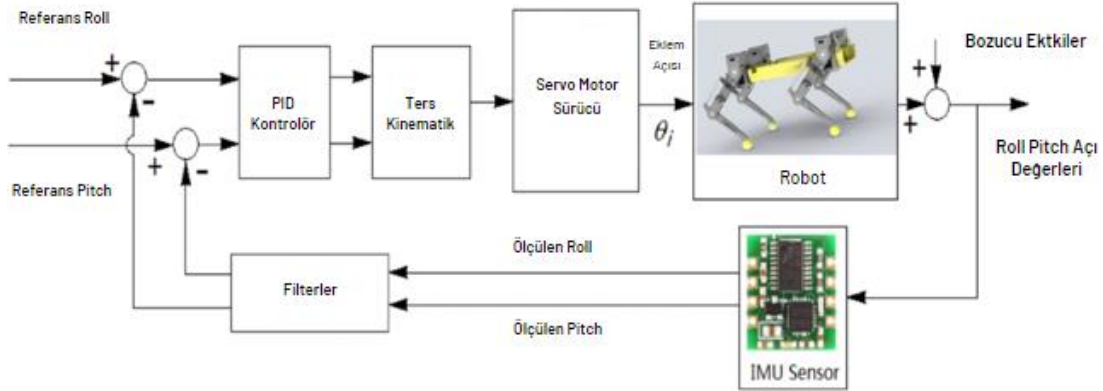
Projede kullanılan kontrol sistemleri PID kontrol algoritmalarından oluşmaktadır. PID tasarımında karmaşık ve ağır bir yöntemden basit ve hızlı yanıt alınabilecek bir kontrol tasarlanmıştır [23]. Sistemin blok şeması Şekil 11'de görüldüğü gibi iki giriş ve tek çıkıştan oluşmaktadır. Girişler gövdenin X ve Y eksenindeki açısal referans değerine yani "sıfır" değerine eşittir. Geri besleme

olarak MPU6050 ivme sensöründen alınan veriler kullanılmaktadır. Hareket halinde sürekli dengede durması istenen robot için kontrol isterleri aşağıdaki gibi sıralanabilir:

- Hızlı manevra kabiliyeti
- Motorların gövde hareketine hızlı cevap vermesi
- Motorların anlık veri ile beslenmesi
- Eğimli ve dağınık zeminlerde hareket kabiliyeti

2.5.1. PID Kontrolör

PID kontrol, basit yapısı nedeniyle sektörde kullanılan yaygın kontrol yöntemlerinden biridir. PID kontrol blok diyagramı Şekil 11’de gösterilmektedir. PID kontrolü ile her bir eklem ters kinematik tarafından kontrol edilir ve roll-pitch açıları buna göre elde edilir. Bu kapalı sistemi kullanırken MPU 6050 ivme sensörü verileri ‘complementary’ filtre ile PID kontrolüne işlemi yapılarak ters kinematik çıktıları alınmaktadır [24,25]. Akabinde motor sürücülere iletilmektedir. Motor sürücüler robotun üzerindeki servo motorların açısal kontrolüyle robotun hareket etmesi sağlanmaktadır.



Şekil 11. PID Blok Diyagramı

Hata sinyali ve kontrol denklemi aşağıda verilmiştir.

$$e(t) = r(t) - y(t) \quad (10)$$

$$u(t) = K_p e(t) + K_I \int e(t) dt + K_D \frac{d}{dt} e(t) \quad (11)$$

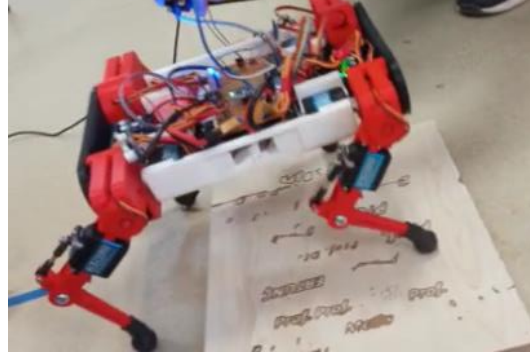
PID parametreleri, tabloda verilmiştir. Parametreleri iyileştirmek için denemeler yapılmış. Efektif sonuç alınan değerler Tablo 6’da gösterilmiştir [26,27].

Tablo 6. PID parametreleri

KP: 0.01521	KI: 0.02214	KD:0
--------------------	--------------------	-------------

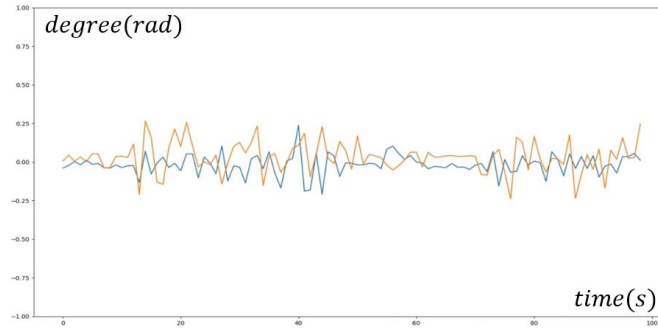
3. Tartışma

Test çalışmalarında robotun PID kontrole göre performansı farklı zemin koşullarında test edilmiştir. Yürüyüş modunda ve eğimli arazide başarılı sonuçlar vermiştir. Şekil 12’de laboratuvar ortamında 30 mm’lik bir engeli devrilmeden aşabilmektedir.



Şekil 12. Farklı yükseklikte yürütme testi.

Aynı zamanda yürüyüş esnasında elde edilen Roll ve Pitch eksenin sapma açısının grafiği Şekil 13’te gösterilmiştir. PID kontrolcü ani değişimler oluşmasına karşı gösterdiği cevaplar radyan birimiyle grafiğe dökülmüştür. Böylece sistemin çalışabilirliği başarılı bir şekilde test edilmiştir.



Şekil 13. Referans değer sıfıra göre roll ve pitch eksen grafikleri

4. Sonuçlar

Bu çalışmada, dördü robotun kinematik analizi, vücut ve alt sistem tasarımı, yazılım teknikleri ve test çalışması yapılmıştır. Ayrıca çalışmada gövdeyi başarılı bir şekilde dengeleyebilen, arazide yürüeyebilen, dengede kalıp PID kontrolü sağlanan dördü bir robot geliştirilmiştir.

Dört bacaklı robotun kinematik denklemleri çıkartılarak robotun kinematik modeli analiz edilmiştir. Kinematik model ile yürüyüş hareketi sağlamak adına yörünge planlaması yapılmıştır.

Belirli bir yörüngede adım atan robotun yürüyüş esnasında denge kontrolü için PID kontrolcü tasarlanmıştır. Ede edilen sonuçlara göre PID kontrolü robotun eğimli bir zeminde hareketine olanak sağlamaktadır. Aynı zamanda farklı yükseltilere sahip ortamlara geldiğinde robot gövdesini dengede tutmayı başarmaktadır. Çalışmalardan elde edilen bulgulara göre robot verimli bir yürüyüş ve daha hızlı cevap veren hareketlere sahip olması için kontrol yöntemlerini değiştirmek gerekmektedir. Bununla birlikte robotun hafifletilerek daha stabil ve kararlı hareketler sergilemesi sağlanabilir. Elektronik olarak sistemin güç katını ve sinyal katını daha iyi tasarlayarak motorların verimi artırılabilir. Ek olarak geri besleme yöntemlerine farklı sensörler eklenerek daha doğru ölçümlene ile kontrol sağlanabilir.

Bu robotun bazı yazılım ve tasarım yönleri için "Notspot" [18], "Stanford Quadruped" [26], "Spotmicro" [28] adlı açık kaynaklı projeler kullanıldı. Yakın gelecekte, bacaklı robotlar çoğu hayati sistemler için otonom robotlar olarak kullanılacaktır.

Tüm çalışma dosyalarına aşağıdaki linkten erişilebilir:

https://drive.google.com/drive/folders/1rr4aptyY1WZ_ndDsWCqrPm7DT3JrMTYb?usp=sharing

Teşekkür

Bu çalışma kapsamındaki deneyler, Kocaeli Üniversitesi Mekatronik Mühendisliği Bölümü Sensör Laboratuvarında gerçekleştirilmiştir. Bildiri yazarları, Sensör Laboratuvarı çalışanlarına verdikleri destekten ötürü teşekkür ederler. Ayrıca, projenin maliyetini karşıladığı için TUBİTAK RUTE ve Kocaeli Büyükşehir Belediyesine ayrıca teşekkür ederler.

Referanslar

[1] Priyaranjan Biswal, Prases K. Mohanty, Development of quadruped walking robots: A review, Ain Shams Engineering Journal, Volume 12, Issue 2, 2021, Pages 2017-2031, ISSN 2090-4479, <https://doi.org/10.1016/j.asej.2020.11.005>.

[2] Website, Online, <https://www.bostondynamics.com/products/spot>

[3] J. Denavit and R.S. Hartenberg, "A kinematic notation for Lower-pair mechanisms based on matrices," ASME Jappl. Mechan, pp. 215-221, June 1955.

[4] Şen, Muhammed Arif & Bakırcıoğlu, Veli & Kalyoncu, Mete. (2017). Inverse Kinematic Analysis Of A Quadruped Robot. International Journal of Scientific & Technology Research. 6.

[5] J. H. Park and M. Choi, "Generation of an optimal gait trajectory for biped robots using a genetic algorithm," JSME Int. J. Ser. C, vol. 47, no. 2, pp. 715–721, 2004.

- [6] A. Spröwitz, M. Ajallooeian, A. Tuleu and A. Ijspeert. Kinematic primitives for walking and trotting gaits of a quadruped robot with compliant legs, in *Frontiers in Computational Neuroscience*, vol. 8, num. 27, p. 1-13, 2014.
- [7] A. Sproewitz, A. Tuleu, M. Vespignani, M. Ajallooeian and E. Badri et al. Towards Dynamic Trot Gait Locomotion—Design, Control and Experiments with Cheetah-cub, a Compliant Quadruped Robot, accepted in *International Journal of Robotics Research*.
- [8] Yong Gao, Wu Wei, Xinmei Wang, Dongliang Wang, Yanjie Li, Qiuda Yu, Trajectory tracking of multi-legged robot based on model predictive and sliding mode control, *Information Sciences*, Volume 606, 2022, Pages 489-511, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2022.05.069>.
- [9] S. Sugiyama, Multi-Stage PWM DC Servo Motor Control, *IFAC Proceedings Volumes*, Volume 22, Issue 18, 1989, Pages 341-345, ISSN 1474-6670, [https://doi.org/10.1016/S1474-6670\(17\)52864-5](https://doi.org/10.1016/S1474-6670(17)52864-5).
- [10] John R. Rebula; Peter D. Neuhaus; Brian V. Bonnlander; Matthew J. Johnson; Jerry E. Pratt , A Controller for the LittleDog Quadruped Walking on Rough Terrain , 10-14 April 2007. Rome, Italy. DOI: 10.1109/ROBOT.2007.363191 IEEE;1050-4729
- [11] Boston Dynamics, Spot, Website: <https://shop.bostondynamics.com/spot>
- [12] Evan Ackerman, Boston Dynamics Announces New WildCat Quadruped Robot, 04 October 2013, Online: <https://spectrum.ieee.org/whoa-boston-dynamics-announces-new-wildcat-quadruped>
- [13] Xiaomipedia, Cyberdog, Website: <https://xiaomipedia.com/en/p/cyber-dog/> , Release date: August 2021
- [14] Aysha M, Solo 8, an affordable robot dog?, Published on July 6, 2020. Online: <https://www.3dnatives.com/en/solo-8-robot-dog-060720206/#!>
- [15] Virbhadrappa Kalburgi, P.M. James, Pramod Sreedharan, Control system design for four-legged walking robot with insect type leg using ROS, *Materials Today: Proceedings*, Volume 46, Part 10, 2021, Pages 5092-5097, ISSN 2214-7853, <https://doi.org/10.1016/j.matpr.2020.10.428>.
- [16] E. H. Maciel, R. Ventura Bayan Henriques and W. F. Lages, "Control of a Biped Robot Using the Robot Operating System," 2014 Joint Conference on Robotics: SBR-LARS Robotics Symposium and Robocontrol, 2014, pp. 247-252, doi: 10.1109/SBR.LARS.Robocontrol.2014.46.

[17] Priyaranjan Biswal, Prases K. Mohanty. Development of quadruped walking robots: A review. Ain Shams Engineering Journal, Available online 3 December 2020, <https://doi.org/10.1016/j.asej.2020.11.005>

[18] Website, Online: https://github.com/lnotspotl/notspot_sim_py

[19] Nan Hu, Shaoyuan Li, Dan Huang, Feng Gao, Crawling Gait Planning for a Quadruped Robot with High Payload Walking on Irregular Terrain, IFAC Proceedings Volumes, Volume 47, Issue 3, 2014, Pages 2153-2158, ISSN 1474-6670, ISBN 9783902823625, <https://doi.org/10.3182/20140824-6-ZA-1003.01851>.

[20] Amritanshu Manu, Shakti S. Gupta, Mangal Kothari, Path Tracking Strategy for Quadruped Robots Using a Hierarchical Framework, IFAC-PapersOnLine, Volume 55, Issue 1, 2022, Pages 192-197, ISSN 2405-8963, <https://doi.org/10.1016/j.ifacol.2022.04.032>.

[21] Kalakrishnan M, Buchli J, Pastor P, Mistry M, Schaal S. Learning, planning, and control for quadruped locomotion over challenging terrain. The International Journal of Robotics Research. 2011;30(2):236-258. doi:10.1177/0278364910388677

[22] P. Anggraeni, I. Rokhim and R. M. Salam, "Design and Development of Multiple Mobile Manipulator Robots using Gazebo-ROS," 2020 International Conference on Applied Science and Technology (iCAST), 2020, pp. 672-676, doi: 10.1109/iCAST51016.2020.9557660

[23] Gürsel Özmen, N., & Marul, M. (2022). Stabilization and tracking control of an x-z type inverted pendulum system using Lightning Search Algorithm tuned nonlinear PID controller. Robotica, 40(7), 2428-2448. doi:10.1017/S0263574721001727

[24] Website, Online: <https://pupper.readthedocs.io/en/latest/index.html>

[25] Howard Chen, Mark C. Schall, Nathan B. Fethke, Measuring upper arm elevation using an inertial measurement unit: An exploration of sensor fusion algorithms and gyroscope models, Applied Ergonomics, Volume 89, 2020, 103187, ISSN 0003-6870, <https://doi.org/10.1016/j.apergo.2020.103187>.

[26] Website, Online: <https://github.com/stanfordroboticsclub/StanfordQuadruped>

[27] Kushwah, Manoj & Patra, Ashis. (2014). PID Controller Tuning using Ziegler-Nichols Method for Speed Control of DC Motor.

[28] Website, Online: <https://github.com/mike4192/spotMicro>