

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ÇİZGİ TAKİP ROBOTUNUN MODELLENMESİ
VE DENETLEYİCİ TASARIMI**

Elektrik Müh. Onur AKBATI

**F.B.E. Elektrik Mühendisliği Anabilim Dalı
Kontrol ve Otomasyon Programında Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Prof. Dr. Galip CANSEVER

İSTANBUL, 2007

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iv
KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
ÇİZELGE LİSTESİ	vii
ÖNSÖZ	viii
ÖZET	ix
ABSTRACT	x
1. GİRİŞ	1
2. ARAÇ MATEMATİKSEL MODELİNİN OLUŞTURULMASI	4
2.1 Araç Geometrik Yapısı	4
2.2 Matematiksel Denklemler	5
2.3 Simülasyon İçin Sistem Bloklarının Oluşturulması	8
2.4 Elde Edilen Modelin Test Edilmesi	9
2.4.1 Aracın Düz Bir Yolda İlerlemesi Testi	9
2.4.2 Aracın Kendi Çevresinde Dönmesi Testi	10
2.4.3 Aracın Sol Tekerleği Çevresinde Dönmesi Testi	11
2.5 Hata Terimleri	12
2.5.1 Açık Hatasının Belirlenmesi	13
2.5.2 Uzaklık Hatasının Belirlenmesi	14
2.5.3 Hata Terimlerinin Toplu Halde İfade Edilmesi	14
2.5.4 Hata Dinamiğinin Oluşturulması	15
3. SİSTEM İÇİN DENETLEYİCİ TASARIMI	16
3.1 Sistemin Doğrusallaştırılması	16
3.2 Sistem İçin PD Denetleyici Tasarımı	18
3.3 Sistem İçin Durum Geri Beslemeli Denetleyici Tasarımı	22
3.4 Sistem İçin Bulanık Mantık Denetleyici Tasarımı	25
3.4.1 Sistem Hakkındaki Verilerin Sözel Olarak İfade Edilmesi	25
3.4.2 Üyelik Fonksiyonlarının Ve Kural Tablosunun Belirlenmesi	26
4. UYGULAMA	32
4.1 Uygulama düzeneğinin yapısı	32
4.2 Uygulama düzeneğinin donanımı	32
4.3 Uygulama düzeneğinin yazılımı	33
4.3.1 Hataların hesaplanması	33
4.3.2 Algoritmanın oluşturulması	35

5.	SONUÇLAR ve TARTIŞMA	37
	KAYNAKLAR.....	38

SİMGE LİSTESİ

M	Araç merkez noktası
v_1	Araç sağ tekerleğinin hızı
v_2	Araç sol tekerleğinin hızı
v_m	Araç merkez noktasının hızı
v_x	Araç merkez noktasının hızının yatay bileşeni
v_y	Araç merkez noktasının hızının dikey bileşeni
θ	Aracın oryantasyon açısı
M_z	Araç dönerken oluşan zahiri merkez
L	Aracın tekerlekleri arasındaki mesafe
r	Oluşan zahiri merkezin araç sol tekerleğine olan uzaklığı
x	Araç merkezinin yatay konumu
y	Araç merkezinin dikey konumu
t_f	Simülasyon süresi
e_d	Araç merkezinin konum hatası
e_θ	Aracın oryantasyon hatası
e_{tot}	Toplam hata
θ_{des}	Aracın ulaşması istenen oryantasyon açısı
ϕ	Aracın çizgiye göre oryantasyon hatası
c	Araç merkezinin hız vektörünün genliği
u	Kontrol işareti
G_c	Denetleyici denklemi
K_d	Türevsel kazanç
K_p	Oransal kazanç
T_{PD}	PD denetleyici içeren transfer fonksiyonu
T_s	Oturma zamanı
$\%OS$	Yüzde üst aşım
ω_n	Doğal frekans
ζ	Sönüm oranı
k_i	i numaralı duruma ait geri besleme kazancı
T_{GB}	Durum geri beslemesi uygulanmış sistemin transfer fonksiyonu
μ	Bulanık kümeye üyelik miktarı

KISALTMA LİSTESİ

PD	Proportion and Derivate (Oransal ve Türevsel)
N	Negatif
S	Sıfır
P	Pozitif
NB	Negatif Büyük
PB	Pozitif Büyük

ŞEKİL LİSTESİ

Şekil 1.1 Ackermann Sürüş Tekniği.....	1
Şekil 1.2 Diferansiyel Sürüş Tekniği.....	2
Şekil 2.1 Modellenecek olan araç yapısı	4
Şekil 2.2 Aracın kartezyen koordinat sisteminde gösterimi	5
Şekil 2.3 Zahiri noktanın tanımlanması.....	7
Şekil 2.4 Sistem blok diyagramı.....	9
Şekil 2.5 Aracın düz ilerlemesine ait simülasyon sonuçları	10
Şekil 2.6 Aracın kendi çevresinde dönmesine ait simülasyon sonuçları.....	11
Şekil 2.7 Aracın sol tekerleği çevresinde dönmesine ait simülasyon sonuçları	12
Şekil 2.8 Hataların geometrik olarak tanımlanması	13
Şekil 2.9 Uzaklık hatasının geometrik gösterimi.....	14
Şekil 3.1 Doğrusallaştırılmış sistem blok diyagramı.....	17
Şekil 3.2 PD Denetleyicili sistemin cevabı	20
Şekil 3.3 PD Denetleyicili sistemin çizgi takibi	21
Şekil 3.4 PD Denetleyicili sistemin oryantasyon hatası	21
Şekil 3.5 Durum geri beslemeli sistemin cevabı	23
Şekil 3.6 Durum geri beslemeli sistemin çizgi takibi.....	24
Şekil 3.7 Durum geri beslemeli sistemin oryantasyon hatası.....	24
Şekil 3.8 Hatalara ait giriş üyelik fonksiyonları	26
Şekil 3.9 Denetleyiciye ait çıkış üyelik fonksiyonu	27
Şekil 3.10 Bulanık denetleyicili sistemin cevabı.....	28
Şekil 3.11 Bulanık denetleyicili sistemin çizgi takibi	28
Şekil 3.12 Bulanık denetleyicili sistemin oryantasyon hatası	29
Şekil 3.13 Bulanık denetleyicili sistemin ikinci veri seti için cevabı.....	30
Şekil 3.14 Bulanık denetleyicili sistemin ikinci veri seti için çizgi takibi	30
Şekil 3.15 Bulanık denetleyicili sistemin ikinci veri seti için oryantasyon hatası	31
Şekil 4.1 Aracın pist üzerindeki görünümü	32
Şekil 4.2 Uygulama donanımının blok gösterimi.....	33
Şekil 4.3 Konum hatası için sensör ağırlıkları.....	34
Şekil 4.4 Açı hatası için sensör ağırlıkları	35
Şekil 4.5 Uygulama algoritmasının akış diyagramı.....	36

ÇİZELGE LİSTESİ

Çizelge 2.1 Aracın düz hareketine ait veri seti	9
Çizelge 2.2 Aracın kendi çevresinde dönmesine ait veri seti	10
Çizelge 2.3 Aracın sol tekerleği çevresinde dönmesine ait veri seti	11
Çizelge 3.1 Doğrusallaştırılmış sistem katsayıları için değer ataması	18
Çizelge 3.2 PD denetleyici tasarımı için simülasyon verileri.....	19
Çizelge 3.3 PD Denetleyici için seçilen kazançlar	19
Çizelge 3.4 Durum geri beslemeli denetleyici için seçilen kazançlar	22
Çizelge 3.5 Bulanık denetleyici kural tablosu	27
Çizelge 3.6 Bulanık mantık denetleyici için ikinci simülasyon verileri.....	29

ÖNSÖZ

Yaptığım bu çalışmada benden yardım ve desteğini esirgemeyen Tez Danışmanım Prof. Dr. Galip Cansever'e teşekkürü borç bilirim.

Çalışmalarımda bana sürekli yardımcı ve destek olan Türker Türker ve Akın Delibaşı'na ayrıca teşekkür ederim.

Eğitim hayatım süresince bana hem maddi hem de manevi olarak destek olan aileme teşekkür ederim.

ÖZET

Günümüzde hızla ilerleyen robot teknolojilerinde sıklıkla karşılaşılan bir problemde tekerlekli araçların konum kontrolüdür. Bu tez çalışmasında tekerlekli araçların kontrol probleminin bir çeşidi olan çizgi takibi ele alınmıştır.

Konu üzerine yapılan literatür araştırmalarında bu konuyla ilgili yapılmış az sayıda akademik çalışmanın bulunmuş olması çalışmayı daha da cazip hale getirmiştir.

Bu çalışmada öncelikle ikisi hareketi sağlamak üzere, toplam dört adet tekerlek içeren bir aracın matematiksel modeli oluşturulmuştur. Bu modelin doğruluğu değişik metotlar ile kontrol edilerek, sistemin işleyişinde hata olarak tanımlanacak terimler belirlenmiştir.

Çalışmanın ilerleyen kısımlarında oluşturulan bu hata terimlerinin sıfırlanması için değişik kontrol metotları denenmiş, bunların simülasyonu için MATLAB ve SIMULINK yazılımları kullanılarak, elde edilen sonuçlar kıyaslanmıştır.

Son olarak yapılan uygulama ile kullanılan kontrol metotlarından biri gerçek ortam şartları altında uygulama ile test edilmiştir.

Anahtar Kelimeler: Çizgi takibi, konum kontrolü, araç kontrolü

ABSTRACT

One of today's frequently encountered problem in Robotic Technologies is position control. In this study, we deal with line-following which is one kind of position control problem.

Finding less solutions in the literature research about this study made it more attractive.

In this study, first of all, we formed the mathematical model of a totally four wheeled vehicle twice of which is used for motion. Then the correctness of the model is checked with different methods and the error terms are defined.

In the following parts of study, to cancel the error terms, different control methods tested, simulation of these methods are done in MATLAB environment and then the results are compared.

At last, one of the control method used is tested under real physical conditions with an application.

Keywords: Line following, position control, cart control

1. GİRİŞ

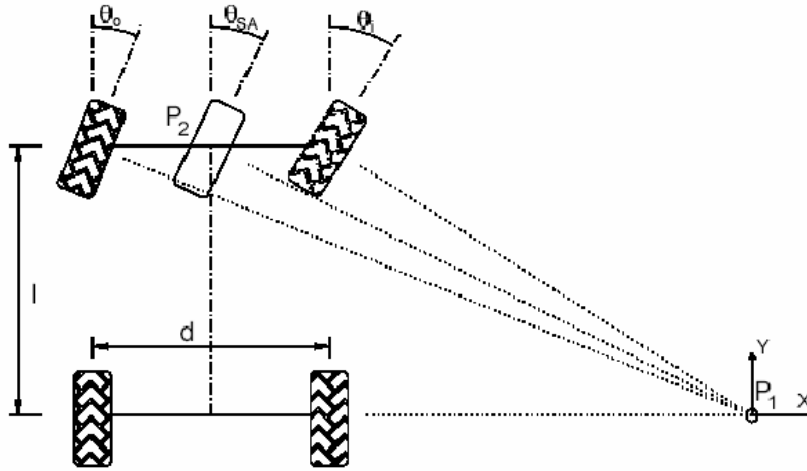
Robotik çalışmalarının farklı bir tipi olan çizgi takip araçları günümüzde teknolojinin ilerlemesi ile basit hobi çalışmaları olmaktan çıkıp, endüstriyel ortamda da kendine yer bulmuştur. Bunlara örnek olarak çizgi takip eden tekerlekli araçlar yardımı ile depolarda raflara malzeme yerleştiren robotlar, çizgi takip eden vinç sistemleri gibi örnekler verilebilir.

Bu araçlar genellikle birçoğu hobi çalışmalarında olmak üzere basit kontrol yöntemleri ve deneysel sonuçlar ile denetlenmektedir. Bu tip çalışmaların büyük bir kısmında sistemin matematiksel modeli ele alınmadan, uzman bilgisi doğrultusunda elde edilen sonuçlara yönelik denetleyiciler tasarlanarak bunun iyileştirilmesi yönüne gidilmektedir.

Bu tip bir sistemin ayrıntılı bir şekilde modellenmesine yönelik bir çalışma Almeida vd. (1998) tarafından gerçekleştirilmiştir.

Çizgi takip eden araçların, konum kontrolünün yapıldığı araçlardan farkı, bu araçların belli bir koordinata gitmek yerine, sürekli olarak değişen bir çizgiyi referans olarak alıp, bunun üzerinde en az hata ile ilerleyebilmesidir.

Temel olarak bu araçlarda Diferansiyel sürüş tekniği ve Ackermann sürüş tekniği olmak üzere iki tip sürüş tekniği kullanılmaktadır. Şekil 1.1' de Ackermann sürüş tekniğinin geometrik yapısı görülmektedir.



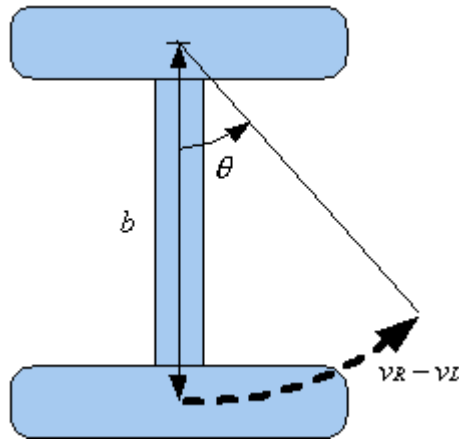
Şekil 1.1 Ackermann Sürüş Tekniği

Ackermann sürüş tekniğinin en yaygın örneği otomobilleridir. Günümüzde halen yaygın şekilde kullanılmaktadır. Bunun sebebi sistemin oluşturulmasının kolaylığı ve yol üzerindeki hareketinin diğer yöntemlere nazaran daha kararlı olmasıdır.

Ackermann sürüş tekniği basit olarak aracın ön tekerleklerini uygun açığa ulaşmasını sağlayan direksiyon sistemi ve araca hareketini verecek olan ön veya arka diferansiyel aksamından oluşur.

Diğer sürüş tekniği ise diferansiyel sürüş tekniğidir. Bu sürüş tekniğinde ise araca hareketini sağ ve sol olmak üzere iki tekerlek sağlar. Bunun dışında aracın ağırlık merkezine bağlı olarak serbest hareket eden bir veya duruma göre iki tekerlek eklemesi yapılır.

Diferansiyel sürüş tekniği adından da anlaşılacağı şekilde iki tekerlek arasındaki hız farkına bağlı olarak aracın kendi hizasına göre sağa veya sola doğru hareket etmesini sağlamaktadır. Şekil 1.2' de diferansiyel sürüş tekniğine ait basitleştirilmiş geometrik yapı görülmektedir.



Şekil 1.2 Diferansiyel Sürüş Tekniği

Şekil1.2' de aracın yol üzerinde devrilmeden durmasını sağlayacak olan yardımcı tekerlekler gösterilmemiştir. Görüldüğü gibi sağ tekerleğin yapacağı hareket sol tekerliği merkez alarak tekerlekler arası mesafe kadar yarıçapa sahip olan bir yay çizmektedir. Aynı şekilde sol tekerleğin yapacağı harekette sağ tekerleği merkez alarak bir yay çizecektir.

Diferansiyel sürüş tekniğinin en büyük avantajlarından biri aracın yapısı gereği keskin hareketler yapabilmesidir. Ackermann sürüş tekniği ile kıyaslandığında ise diferansiyel gibi aksamaların kullanılmasına gerek kalmamaktadır. Fakat Ackermann sürüş tekniğindeki kararlı yapı ortadan kalkmakta, bu durum gerçek bir ortamda araç çizgi takibi gibi işlemler yaparken sağ ve sol tekerleğe hareket sağlayan motorlar arasındaki yapısal farklar nedeni ile aracın küçük salınımlar yapması ile sonuçlanmaktadır.

Diferansiyel sürüş tekniği aracın yol hizasında gitmesini çok iyi bir şekilde sağlamasa da, ani yön değişimlerine olanak sağlamaktadır. (Akbatı ve Cansever, 2007)

Diferansiyel sürüşte tekerleklere uygulanan hızlardaki küçük hatalar farklı yörüngelerin izlenmesine sebep olmaktadır. Diferansiyel sürüşte aracı dengelemek için serbest dönen bir tekerleğe de ihtiyaç vardır. Bu yüzden yüzeydeki kasis ve hatalardan kolaylıkla etkilenmekte, bu da onların uygulanabilirliğini laboratuvar dışı kullanımını zorlaştırmaktadır. (Dudek ve Jenkin, 2000)

Literatürde yapılan çalışmalarda diferansiyel sürüş tekniği, kurulum kolaylığı açısından daha çok tercih edilmektedir. Buna karşın sistem modelinin çıkarılması konusunda farklı yaklaşımlar görülmektedir.

Ishihara vd.'e (1994) göre araç kontrolü için hata miktarının belirlenmesinde ve sistemin denetlenmesinde sayısal verilere dayalı bir çalışma yeterlidir.

Almeida vd.'e (1998) göre ise sistemin modellenmesinde, kullanılan motorların modelleri dahi ele alınarak, tam bir modele ulaşılmaya çalışılması uygun görülmüş, sistemin denetlenmesinde ise çok sayıda denetleyici çeşidinin farklı ölçütlerde avantaj sağladığı deneysel sonuçlarla tespit edilmiştir.

2. ARAÇ MATEMATİKSEL MODELİNİN OLUŞTURULMASI

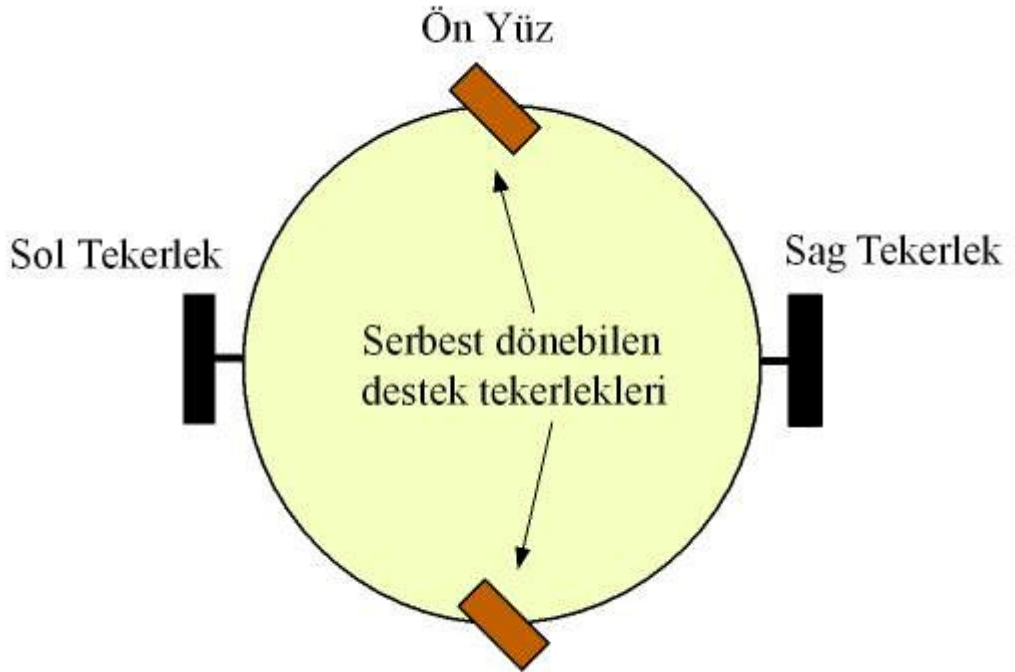
Uzman bilgisine dayalı kontrol metotları kullanmak yerine, öncelikle aracın modelinin oluşturulması daha uygundur. Böylelikle matematiksel denklemler yardımı ile sistemin kontrolünde daha etkin yöntemler geliştirilebilir.

Model oluşturulmasında öncelikle aracın geometrik yapısının kararlaştırılması, sonrasında araca ait tekerlek hızlarına bağlı hareket denklemlerinin oluşturulması yoluna gidilmiştir.

En son olarak aracın hareket denklemlerinin belirlenmesinden sonra çizgi takibini gerçeklenebilmesi için gerekli denetleyici yapının oluşturulmasında ihtiyaç duyulacak olan hata dinamikleri oluşturulur.

2.1 Araç Geometrik Yapısı

Araç geometrik yapısının oluşturulmasında öncelikle literatürdeki kullanılan modeller gözlemlenmiş, fakat denklemlerin daha basite indirgenmesi ve uygulamada çıkabilecek olan zorlukların önüne geçmek için simetrik bir araç yapısı benimsenmiştir. Şekil 2.1' de aracın yapısı açıkça görülmektedir.



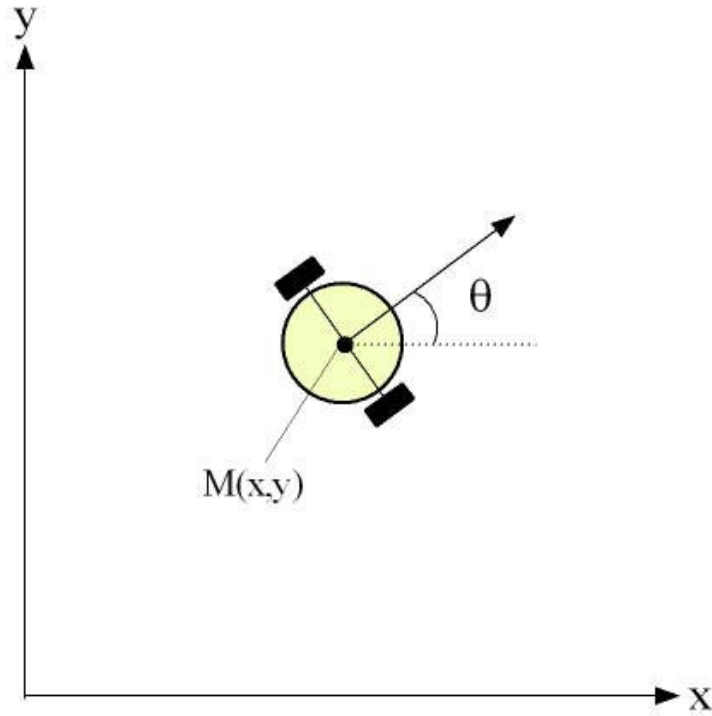
Şekil 2.1 Modelleneyecek olan araç yapısı

Kullanılacak olan araç modelinde iki adet kendi eksenine çevresinde serbest dönebilen destek tekerlekleri kullanılmıştır. Bu tekerleklerin oluşturacağı sürtünme ve diğer etkiler araç modelinde ihmal edilecektir.

Aracın hareket denklemleri yazılırken araç merkezi Şekil 2.1’ de görülen aracın tam orta noktası seçilecektir. Buradaki amaç yapının simetrik olmasını sağlayıp denklemlerin kolaylaştırılmasıdır.

2.2 Matematiksel Denklemler

Seçilmiş olan araç modeli, kartezyen koordinat sistemi altında düşünüldüğünde, araç ile ilgili iki temel veriden bahsedilebilir. Bunlardan biri aracın merkez noktasının x ve y düzlemi üzerindeki yerleşimi, diğeri ise aracın oryantasyonu veya bir diğer deyişle koordinat sistemi üzerindeki açısıdır (θ). Şekil 2.2’ de aracın kartezyen koordinatlardaki yerleşimi ile konum ve oryantasyonu gösterilmiştir.



Şekil 2.2 Aracın kartezyen koordinat sisteminde gösterimi

Öncelikle aracın merkezinin (M) konumsal olarak değişimini ele alacak olursak, aracın tekerleklerinin sahip oldukları hızları sağ tekerlek için v_1 ve sol tekerlek için v_2 olarak tanımlanır. Bu durumda araç merkezinin (M) sahip olduğu hızın vektörsel değeri (2.1)'deki şekilde gösterilir.

$$v_m = \frac{v_1 + v_2}{2} \quad (2.1)$$

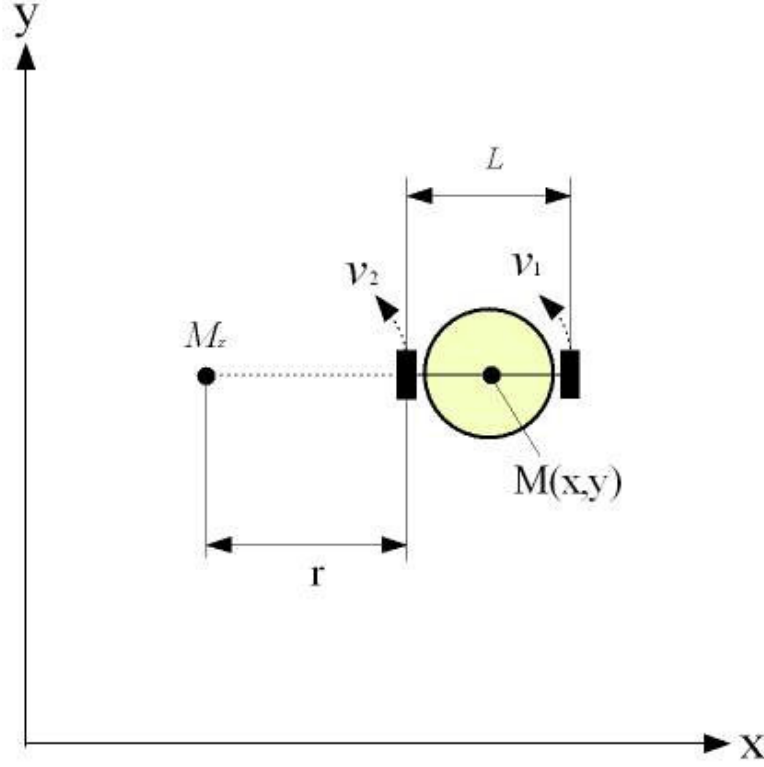
Bu sonuca bağlı olarak, aracın x ve y koordinatlarına göre sahip olduğu hız vektörleri (2.2)'deki gibi ayrıştırılabilir.

$$v_x = v_m \cos \theta = \frac{v_1 + v_2}{2} \cos \theta \quad (2.2)$$

$$v_y = v_m \sin \theta = \frac{v_1 + v_2}{2} \sin \theta$$

Aracın oryantasyon açısının(θ) değişiminin ($\dot{\theta}$) hesaplanabilmesi için araç merkez noktasının (M) açısal olarak dönüş hızının bulunması gerekmektedir. Bu zaten oryantasyon açısının değişimi ($\dot{\theta}$) olacaktır.

Aracın tekerleklerinin sahip olduğu hızlara bağlı olarak, araç zahiri bir nokta (M_z) çevresinde dönecektir. Açıkça görülmektedir ki v_1 ve v_2 hızları birbirine eşit olduğu durumda araç için tanımlanan bu zahiri nokta aracın merkez noktası (M) olacaktır, dolayısıyla araç açısında herhangi bir değişiklik olmayacaktır. Bununla ilgili geometrik tanımlar Şekil 2.3' te gösterilmiştir.



Şekil 2.3 Zahiri noktanın tanımlanması

Şekil 2.3' te tanımlanan M_z noktası araç sol tekerleğinden olan r mesafesinde oluşmaktadır. Bu uzaklık miktarı şekilde pozitif bir değişken olarak gösterilmiş, fakat aracın ters yönde hareket etmesi durumunda (θ açısının azalması durumunda) ve bu merkezin sol tekerlek ile merkez (M) arasında oluşması durumunda bu değer negatif olacaktır. Bunun sonucu olarak M_z noktası sol tekerleğin sağ tarafında oluşacaktır. Burada aracın tekerlekleri arasındaki uzaklık ise L olarak tanımlanmıştır.

Tanımlanan zahiri noktanın (M_z) hesaplanabilmesi için r mesafesinin hesaplanması gerekmektedir. Bu mesafe araç tekerleklerinin sahip olduğu hızla orantılıdır. Buna ait bağıntı (2.3)'te verilmiştir.

$$\frac{v_1}{L + r} = \frac{v_2}{r} \quad (2.3)$$

Buradan elde edilecek bir diğer bağıntı, araç merkez noktasının açısal oryantasyonun değişimidir ($\dot{\theta}$). Araç merkezinin (M), zahiri noktaya (M_z) olan uzaklığı kullanılarak bu bağıntı elde edilir. Bununla ilgili denklem (2.4)' te gösterilmiştir.

$$\dot{\theta} = \frac{v_m}{r + \frac{L}{2}} \quad (2.4)$$

Denklem (2.3)'te elde edilen bağıntı kullanılarak r uzaklığı hesaplanırsa (2.5)' teki denkleme ulaşılır.

$$r = \frac{v_2 L}{v_1 - v_2} \quad (2.5)$$

Elde edilen bu sonuç (2.4) denkleminde yerine konulup, gerekli sadeleştirmeler yapıldığında, elde edilen $\dot{\theta}$ açısal değişimi denklem (2.6)' da gösterilmiştir.

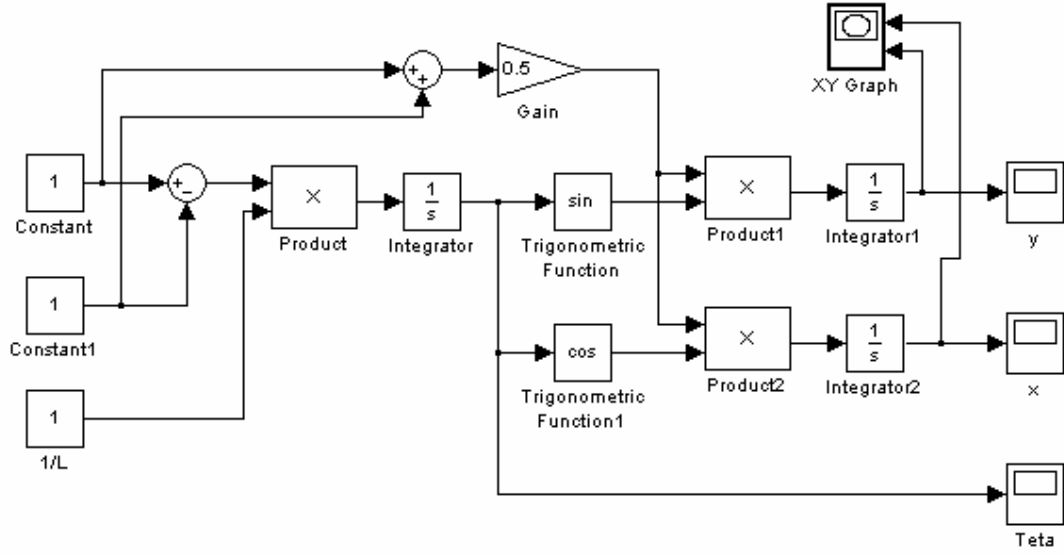
$$\dot{\theta} = \frac{v_1 - v_2}{L} \quad (2.6)$$

Bu denklemin elde edilmesi ile v_1 ve v_2 hızlarına bağlı olarak kartezyen koordinatlarda aracın yapacağı konum değişikliği hesaplanabilmektedir. Denklem (2.7)'de elde edilen denklemler matrisel formda ifade edilmiştir.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \cos \theta & \frac{1}{2} \cos \theta & 0 \\ \frac{1}{2} \sin \theta & \frac{1}{2} \sin \theta & 0 \\ \frac{1}{L} & -\frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \theta \end{bmatrix} \quad (2.7)$$

2.3 Simülasyon İçin Sistem Bloklarının Oluşturulması

Sistemin bilgisayar yazılımında simülasyonunun yapılabilmesi için (2.7)' de elde edilen sonuçlar göz önüne alınarak bir sistem blok yapısı oluşturulmuştur. Böylelikle sisteme verilecek hız giriş değerlerine göre sistemin davranışı gözlemlenebilecektir. Şekil 2.4' te sistemin MATLAB SIMULINK yazılımı altında bloklardan oluşturulan şekli görülmektedir.



Şekil 2.4 Sistem blok diyagramı

2.4 Elde Edilen Modelin Test Edilmesi

Modelin kontrolü için en kolay yöntem, araç hareketlerinin bilindiği bazı giriş hızlarının uygulanıp, denklemlerin oluşturacağı konum değişikliklerinin simülasyon yardımı ile elde edilip, bunun eldeki fiziksel veri ile karşılaştırılmasıdır.

Test için üç temel değer seti kullanılmıştır. Bunlar aracın düz bir istikamette ilerlemesi, olduğu yerde bir daire çizmesi ve bir tekerlek çevresinde daire çizmesi.

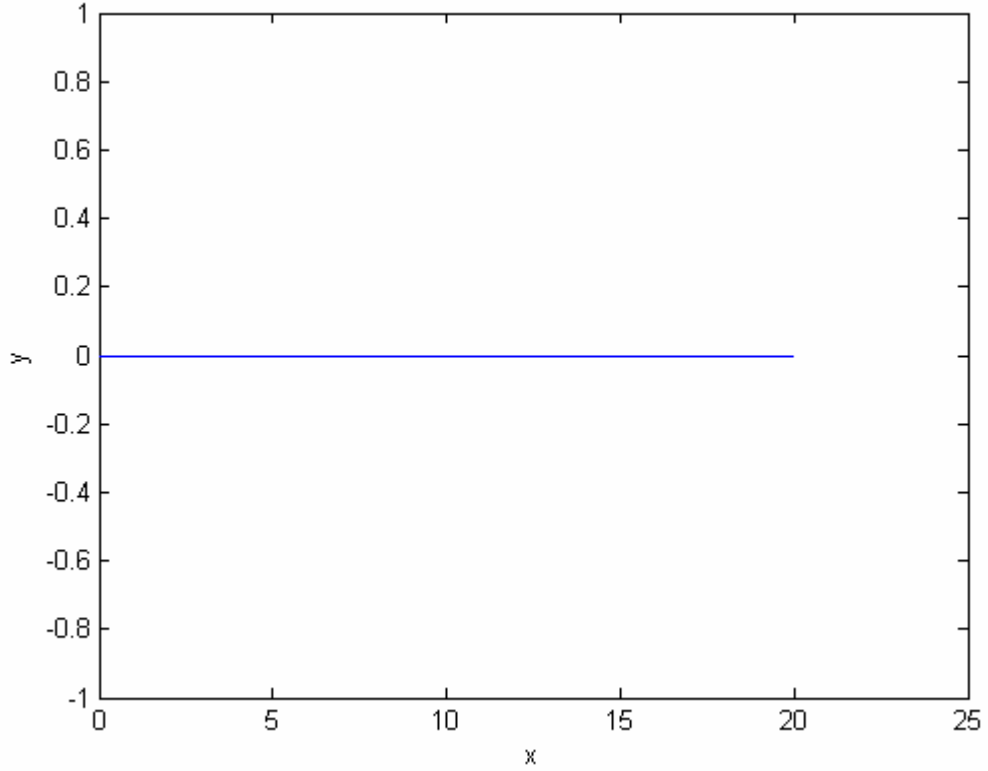
2.4.1 Aracın Düz Bir Yolda İlerlemesi Testi

Aracın düz olarak (Oryantasyonu değişmeden) ilerlemesi için Çizelge 2.1 'de verilen veri seti denklemlere uygulanmıştır.

Çizelge 2.1 Aracın düz hareketine ait veri seti

$x(t=0)$	$y(t=0)$	$\theta(t=0)$	v_1	v_2	L
0	0	0	1	1	1

MATLAB SIMULINK altında yapılan simülasyon sonuçları Şekil 2.5' te görülmektedir. Simülasyon süresi olarak $t_f = 20$ saniye olarak seçilmiştir.



Şekil 2.5 Aracın düz ilerlemesine ait simülasyon sonuçları

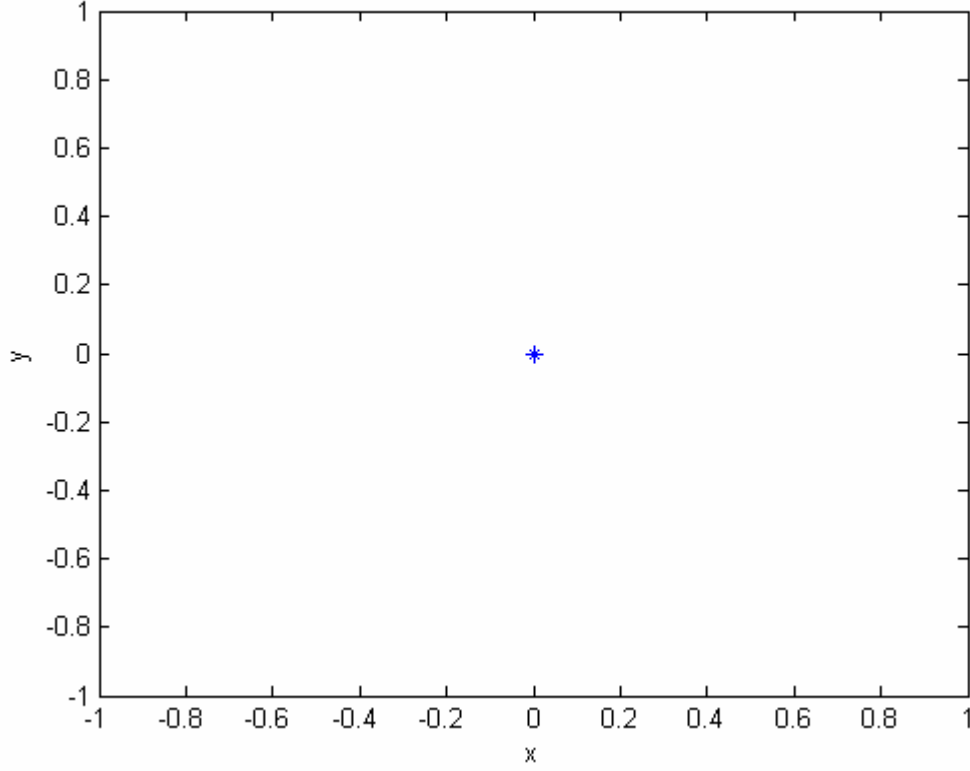
2.4.2 Aracın Kendi Çevresinde Dönmesi Testi

Aracın bulunduğu konumunu değiştirmeden, sadece olduğu yerde dönmesini sağlamak için tekerlek hızlarının birbirine göre ters işarete sahip ve mutlak değer olarak eşitlenmesi gerekir. Buna ait veri seti Çizelge 2.2’ de mevcuttur.

Çizelge 2.2 Aracın kendi çevresinde dönmesine ait veri seti

$x(t=0)$	$y(t=0)$	$\theta(t=0)$	v_1	v_2	L
0	0	0	1	-1	1

MATLAB SIMULINK altında yapılan simülasyon sonuçları Şekil 2.6’ da görülmektedir. Simülasyon süresi olarak $t_f = 20$ saniye olaraktan seçilmiştir.



Şekil 2.6 Aracın kendi çevresinde dönmesine ait simülasyon sonuçları

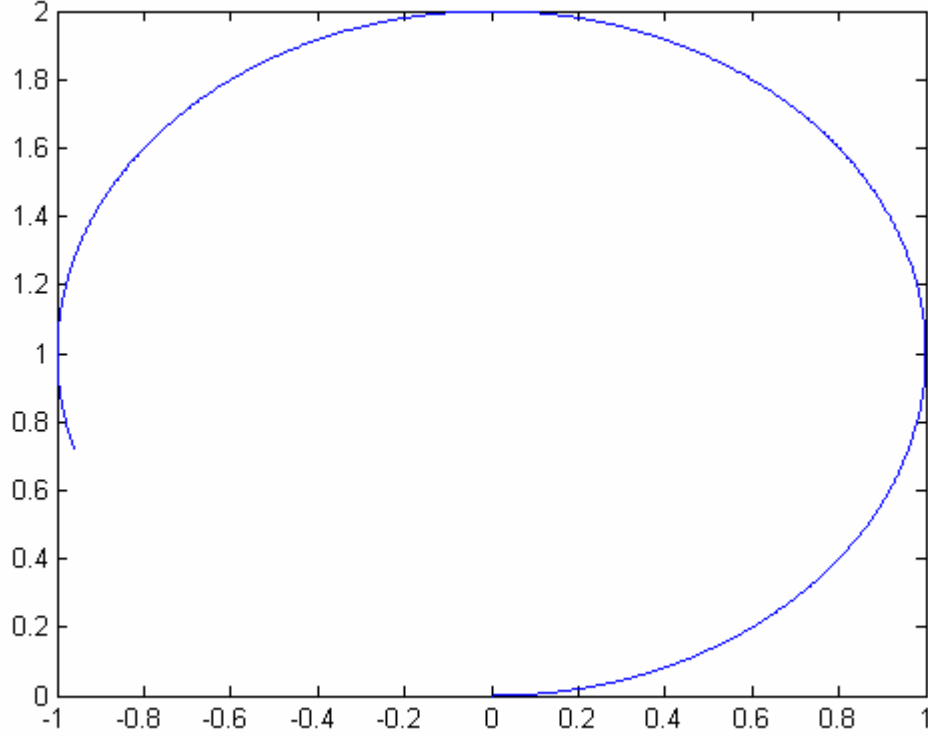
2.4.3 Aracın Sol Tekerleği Çevresinde Dönmesi Testi

Bu son test ile aracın açısal değişiminin doğru şekilde gerçekleştiği de gösterilmek istenmektedir. Normal olarak aracın sağ tekerleği hareket halinde iken sol tekerleği sabit durduğunda sol tekerlek merkezli bir dairenin araç merkezi tarafından çizilmesi beklenmektedir. Bu test için belirlenen veri seti Çizelge 2.3’ te gösterilmiştir.

Çizelge 2.3 Aracın sol tekerleği çevresinde dönmesine ait veri seti

$x(t=0)$	$y(t=0)$	$\theta(t=0)$	v_1	v_2	L
0	0	0	1	0	1

MATLAB SIMULINK altında yapılan simülasyon sonuçları Şekil 2.7’ da görülmektedir. Simülasyon süresi olarak $t_f = 10$ saniye olaraktan seçilmiştir.

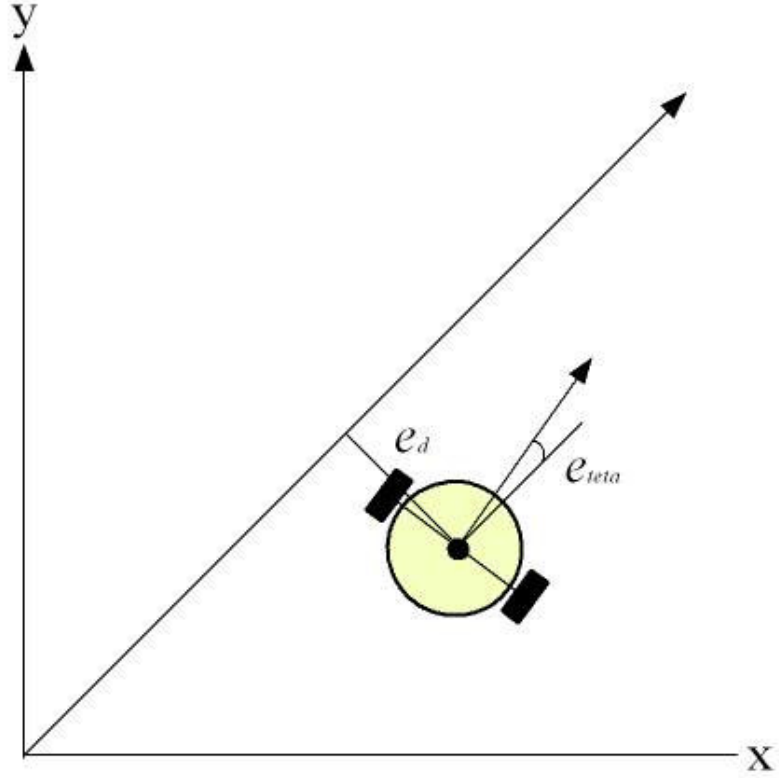


Şekil 2.7 Aracın sol tekerleği çevresinde dönmesine ait simülasyon sonuçları

2.5 Hata Terimleri

Hata terimlerinin oluşturulabilmesi için öncelikle aracın takip edeceği çizginin tanımlanması gerekir. Bu çizginin bir doğru parçası olduğu düşüncesinden hareketle tanımlanacak iki hata terimi yeterli olacaktır. Bunlardan biri araç merkezinin bu doğru parçasına olacak dik uzaklığı (e_d), diğeri ise aracın oryantasyon açısının doğru açısından olan farkıdır (e_θ).

Bu hataların geometrik tanımına ait çizimler Şekil 2.8’ de gösterilmiştir. Burada bir diğer önemli noktada bu hataların işaretidir. Belirlenen doğru parçasının üst tarafında uzaklık hatası pozitif olarak, alt tarafında ise negatif olarak tanımlanacaktır. Aynı şekilde açı hatasının belirlenmesinde de ters saat yönündeki dönüş pozitif, saat yönü istikametindeki dönüş ise negatif olarak kabul edilmiştir. Böylece hata terimleri elde edildiğinde aracın yolun hangi tarafında olduğu ve açısal olarak net bir bilgiye sahip olunabilir.



Şekil 2.8 Hataların geometrik olarak tanımlanması

Oluşturulacak denklemlerin basitliği açısından takip edilecek yolun denklemi $y = x$ şeklinde seçilmiştir. Böylece trigonometrik formüllerin karmaşıklığı azaltılarak, aracın hareket denklemleri ön plana çıkarılmış olacaktır.

2.5.1 Açı Hatasının Belirlenmesi

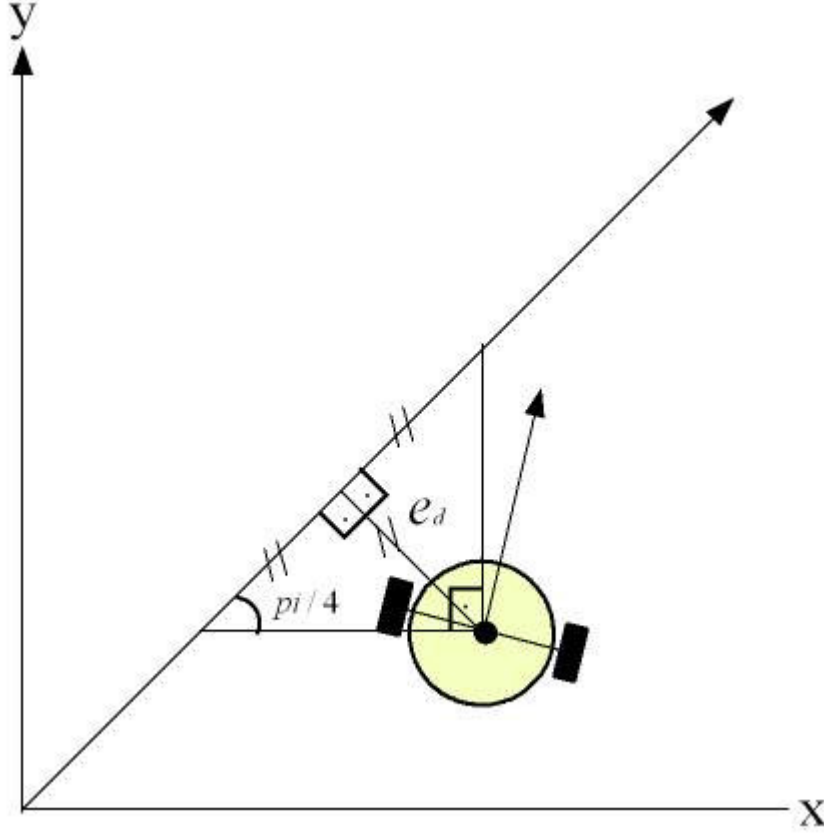
Takip edilecek olan çizgi eğer bir doğru ise, bu hatanın tanımı gayet kolaylaşmaktadır. Bu durumda açı hatası (e_θ) araç açısının (θ) istenilen çizgi açısından (θ_{des}) farkı olacaktır. Bu durum denklem (2.8)' de görülmektedir. Bu açı tanımlanan $y = x$ doğrusu için $\theta_{des} = \frac{\pi}{4}$ şeklinde olacaktır.

$$e_\theta = \theta - \theta_{des} \quad (2.8)$$

$$\theta_{des} = \frac{\pi}{4} \rightarrow e_\theta = \theta - \frac{\pi}{4}$$

2.5.2 Uzaklık Hatasının Belirlenmesi

Bu hata aracın takip edilen yola olan dik uzaklığıdır. Böylelikle yolun bir doğru parçası olduğu göz önüne alındığında aracın yola olan en yakın uzaklığı bulunmuş olacaktır. Takip edilen yol denklemi $y = x$ olduğu için bu değerin hesaplanmasındaki kullanılacak olan çok sayıdaki trigonometrik denklem ortadan kalkmış olur. Şekil 2.9' da bu uzaklığın hesaplanması için kullanılan geometrik yapı gösterilmiştir.



Şekil 2.9 Uzaklık hatasının geometrik gösterimi

Oluşan geometrik şekil, trigonometrik olarak çözümlendiğinde, aracın yola olan dik uzaklığı denklem (2.9)' da verildiği şekilde gösterilebilir.

$$e_d = \frac{y - x}{\sqrt{2}} \quad (2.9)$$

Bu hata teriminin elde edilmesi ile aracın takip edilecek çizgi ile ilgili ilişkisi ortaya çıkarılmıştır.

2.5.3 Hata Terimlerinin Toplu Halde İfade Edilmesi

Bu terimler bir araya getirilip, matrissel formda ifade edilecek olursa, denklem (2.10)' da

görülen yapı elde edilecektir.

$$\begin{bmatrix} e_\theta \\ e_d \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} -\frac{\pi}{4} \\ 0 \\ 0 \end{bmatrix} \quad (2.10)$$

Elde edilen bu yapı bize tanımlanan hataların çözümü için daha önce bulunan araç dinamiğinin çözümlenmesi gerektiğini gösterir. Böylelikle x, y, θ değerlerine ulaşılmış olur.

2.5.4 Hata Dinamiğinin Oluşturulması

Elde edilen sonuçlar ışığında tanımlanan iki hata teriminin değişimi hesaplanarak sistem dinamiği oluşturulabilir. Öncelikle aç hatasına ait denklem (2.11)'de gösterilmiştir.

$$\dot{e}_\theta = \frac{d}{dt} e_\theta = \frac{d}{dt} \left(\theta - \frac{\pi}{4} \right) = \dot{\theta} = \frac{v_1 - v_2}{L} \quad (2.11)$$

Benzer şekilde uzaklık hatasını hesaplanırsa, (2.12)' deki sonuçlar elde edilmektedir.

$$\dot{e}_d = \frac{d}{dt} e_d = \frac{d}{dt} \left(\frac{y-x}{\sqrt{2}} \right) = \left(\frac{\dot{y}-\dot{x}}{\sqrt{2}} \right) = \frac{v_1 + v_2}{2\sqrt{2}} (\sin \theta - \cos \theta) \quad (2.12)$$

Denklem (2.13)' te trigonometrik dönüşümlerden biri gösterilmiştir.

$$\sin(a-b) = \sin a \cdot \cos b - \cos a \cdot \sin b \quad (2.13)$$

(2.13) teki denklem, (2.12)' de bulunan denkleme uyarlanırsa (2.14)'teki gibi bir sonuç elde edilir.

$$\dot{e}_d = \frac{v_1 + v_2}{2} \left[\sin \left(\theta - \frac{\pi}{4} \right) \right] \quad (2.14)$$

3. SİSTEM İÇİN DENETLEYİCİ TASARIMI

Sistemin yapısı doğrusal olmadığı için, sistemin kontrolünü doğrusal denetleyiciler ile gerçeklemek için elde edilen modelin doğrusallaştırılması birçok ölçütün belirlenebilmesi için avantaj sağlayacaktır. Doğrusallaştırma işleminin başarılı olabilmesi, sistem değişkenlerinin belirlenen sınırların dışına çıkmaması ile sağlanabilir.

3.1 Sistemin Doğrusallaştırılması

Doğrusallaştırılmanın yapılabilmesi için sistem için belirlenen denklemlerin çalışacağı belirli bir aralık seçilmesi gerekmektedir. Bunun için belirlenen yaklaşık çalışma noktaları denklem (3.1)' de gösterilmiştir.

$$\theta \cong \frac{\pi}{4} \quad (3.1)$$

Sonrasında yeni bir açı değişkeni tanımlanarak, gereksiz terimlerden kurtulmak mümkündür. (3.2) denkleminde yeni açı değişkeni tanımlanmıştır.

$$\varphi = \theta - \frac{\pi}{4} \quad (3.2)$$

Bu noktadan hareketle kartezyen koordinatlardan kurtulmak için hata denklemleri ve dinamiği ele alınabilir. Bu bakış açısı ile bakıldığında hata dinamiği (3.3)' deki gibi elde edilecektir.

$$\begin{aligned} \dot{e}_\theta &= \dot{\varphi} = \frac{v_1 - v_2}{L} \\ \dot{e}_d &= \frac{v_1 + v_2}{2} \sin(\varphi), \varphi \cong 0 \rightarrow \dot{e}_d = \frac{v_1 + v_2}{2} \varphi \end{aligned} \quad (3.3)$$

Denklemlerde elde edilen $\frac{v_1 + v_2}{2}$ terimi sabit seçilebilir. Bu aracın merkez noktasının hız vektörüdür. Denklem (3.4)' te bu terim sabit olarak tanımlanmıştır.

$$c = \frac{v_1 + v_2}{2} \quad (3.4)$$

Bu noktadan hareketle hata dinamiklerine ait matrissel gösterim ise, denklem (3.5)' te görülmektedir.

$$\begin{bmatrix} \dot{e}_\theta \\ \dot{e}_d \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ c & 0 \end{bmatrix} \begin{bmatrix} e_\theta \\ e_d \end{bmatrix} + \begin{bmatrix} \frac{1}{L} & -\frac{1}{L} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (3.5)$$

Elde edilen doğrusallaştırılmış denklemler sayesinde sistemdeki doğrusal olmayan denklemlerin ortadan kaldırılması dışında, hata dinamiklerinin kullanımı ile tamamen takip edilecek olan çizgiye bağlı bir yapı elde edilmiştir.

Doğrusallaştırılmış sistemin denklemlerini kullanarak sistem transfer fonksiyonunu oluşturursak, öncelikle $\frac{v_1 - v_2}{L}$ teriminin yeni bir şekilde tanımlanması gerekmektedir.

Böylece sistem tek giriş ve tek çıkış haline gelecektir. Bu terimin yeni tanımlanmış hali denklem (3.6) da verilmiştir.

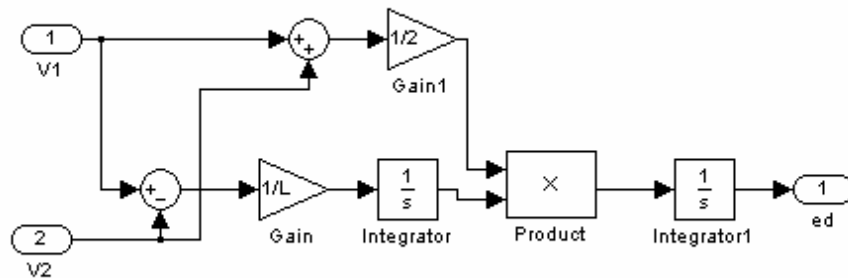
$$\frac{v_1 - v_2}{L} = \frac{u}{L} \rightarrow u = v_1 - v_2 \quad (3.6)$$

Bu dönüşümle beraber sistem transfer fonksiyonu denklem (3.7)' deki gibi elde edilmiş olacaktır.

$$G(s) = \frac{e_d}{u} = \frac{c}{Ls^2} \quad (3.7)$$

Elde edilen transfer fonksiyonu doğrusallaştırılmış sistemin Tip-2 bir sistem olduğunu göstermektedir. Transfer fonksiyonu içindeki integratör sayısı genelde sistem tipi olarak adlandırılır. (Skogestad ve Postlethwaite, 1996)

Doğrusallaştırılmış sistemin blok diyagramı Şekil 3.1' de görülmektedir.



Şekil 3.1 Doğrusallaştırılmış sistem blok diyagramı

3.2 Sistem İçin PD Denetleyici Tasarımı

Tip-2 bir sistemin kararlı hal hatası bulunmayacağı için ve P tip denetleyici kullanıldığı zaman sistem köklerinin kök-yer eğrisinde imajiner eksen üzerinde yer alıp, sistem sürekli bir salınım gerçekleştireceği için PD denetleyici seçilerek ikinci dereceden olan sistemin iki köküne de müdahale etmek mümkün olacaktır. Kullanılacak PD denetleyicinin denklemi (G_c) (3.8)' de verilmiştir.

$$G_c = K_d s + K_p \quad (3.8)$$

Bu denetleyici sistemin doğrusallaştırılmış denklemine uygulandığında, oluşacak yeni sistemin transfer fonksiyonu ($T_{PD}(s)$) hesaplanırsa, (3.9)' daki yapı elde edilecektir.

$$T_{PD}(s) = \frac{c(K_d s + K_p)}{Ls^2 + cK_d s + cK_p} \quad (3.9)$$

Doğrusallaştırılmış sistem oturma noktasından çok uzaklaşmadığı sürece, doğrusal olmayan gerçek sistemle yaklaşık olarak aynı sonuçları verecektir. Buna dayanarak doğrusallaştırılmış sistem üzerinden oturma zamanı (T_s), yüzde üst aşım ($\%OS$) gibi ölçütlere dayanarak tasarım yapılabilir.

Tasarım için kullanılacak ölçütlere ait denklemler (3.10)' da verilmiştir.

$$T_s = \frac{4}{\zeta \omega_n} \quad (3.10)$$

$$\%OS = 100e^{-\pi\zeta/\sqrt{1-\zeta^2}}$$

Oluşturulan doğrusallaştırılmış sistem için araç uzunluğu (L) ve hız vektörü genliği (c) değerlerinin belirlenmesi simülasyon için kolaylık sağlayacaktır. Buna dayalı olarak simülasyonu yapılacak sistem için belirlenen değerler Çizelge 3.1' de verilmiştir.

Çizelge 3.1 Doğrusallaştırılmış sistem katsayıları için değer ataması

c	L
1 m/s	1 m

Oluşan sistemin yeni denklemi (3.11)' de gösterilmiştir.

$$T_{PD}(s) = \frac{K_d s + K_p}{s^2 + K_d s + K_p} \quad (3.11)$$

Sistem katsayıları yardımı ile sistemin sönüm oranı (ζ) ve doğal frekansı (ω_n) hesaplanacak olursa, denklem (3.12)' deki sonuçlara varılacaktır.

$$\omega_n = \sqrt{K_p} \quad (3.12)$$

$$\zeta = \frac{K_d}{2\sqrt{K_p}}$$

Elde edilen bu denklemlere bağlı olarak Çizelge 3.2' de verilen şartlara göre sistemin simülasyonu gerçekleştirilecektir.

Çizelge 3.2 PD denetleyici tasarımı için simülasyon verileri

c	L	%OS	T_s	$e_\theta(t=0)$	$e_d(t=0)$
1 m/s	1 m	%10	3s	0	1 m

Burada verilen ölçütlerden konum hatası (e_d), gerçek sistemde tanımlanırken x ve y değerlerinin başlangıç değerleri için (x_0, y_0) denklem (3.13)'teki kabuller yapılmıştır.

$$x_0 = \sqrt{2} \quad (3.13)$$

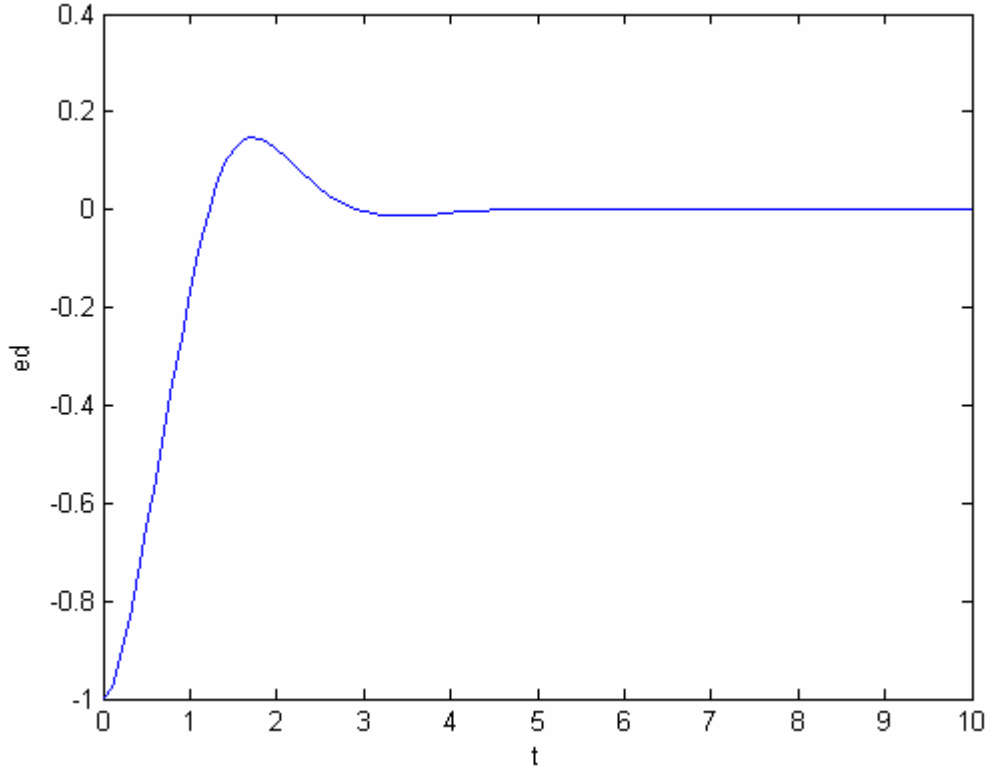
$$y_0 = 0$$

Bu kabuller sayesinde istenilen konum hatası değeri sağlanmış olur. Denetleyiciye ait olan değerler ise Çizelge 3.3' teki gibi elde edilecektir.

Çizelge 3.3 PD Denetleyici için seçilen kazançlar

K_p	K_d
5.07	2.65

Elde edilen bu deęerler iin sistemin cevabı ise Şekil 3.2’ deki grafikteki gibi olacaktır.

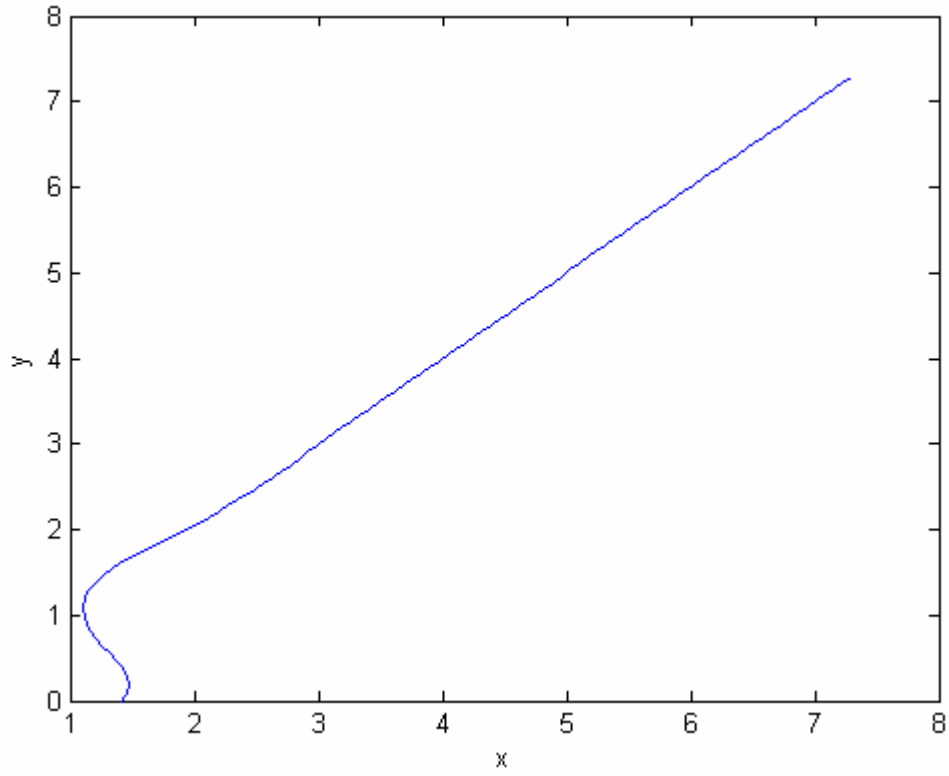


Şekil 3.2 PD Denetleyicili sistemin cevabı

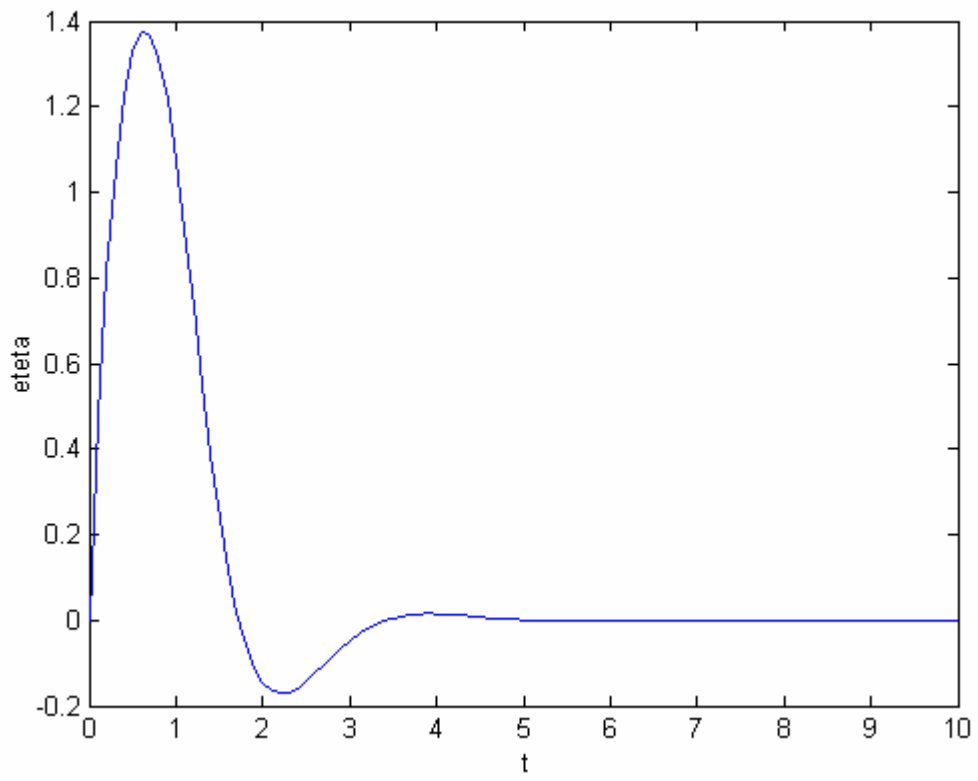
Buradan görüldüğü gibi sistem cevabı doğrusallaştırılmış denklemlerde elde edildiğı gibi olmasa da, istenilen kıstasları yakın bir şekilde gerçekleştirmiştir.

Aracın çizdiği yola ait grafik ise Şekil 3.3’ te görülmektedir. Burada aracın herhangi bir oryantasyon hatasına (e_θ) sahip deęil iken, takip edilecek çizgiye ulaşabilmek iin açısını deęiştirdiğı, çizgi üzerine yerleşirken tekrardan bu hatanın sıfırlandığı görülmektedir.

Aracın oryantasyon açısının hatasına ait deęişim ise Şekil 3.4’te gösterilmiştir.



Şekil 3.3 PD Denetleyicili sistemin çizgi takibi



Şekil 3.4 PD Denetleyicili sistemin oryantasyon hatası

3.3 Sistem İçin Durum Geri Beslemeli Denetleyici Tasarımı

Bu denetleyici yapısında ise, doğrusallaştırılmış sisteme ait durumlar olan e_θ ve e_d ’nin belirli sabit kazançlar (k_1, k_2) ile geri beslemesi yapılır. PD Denetleyicide olduğu gibi, elde edilen sistem yapısı üzerinden belirlenen kıstasların sağlanması amaçlanarak bu kazançlar hesaplanır.

Daha sonra ise sistem için elde edilen bu değerler gerçek sistem üzerinde uygulanır. Burada hesaplanan ve sistemin iç yapısındaki durumlardan biri olan e_θ gene sistemin içinden alınarak gerekli katsayı ile güçlendirilerek geri beslenir. Diğer hata zaten sistemin çıkışı olarak belirlenen e_d ’dir. Aynı şekilde bu hatada ilgili katsayı ile güçlendirilerek geri beslenecektir.

Sistem durum geri beslemesi ile denetlendiğinde, oluşturulacak olan kontrol sinyali u ’ya ait olan denklem (3.14)’te verilmiştir.

$$u = -k_1 e_\theta - k_2 e_d \quad (3.14)$$

Oluşturulan kontrol sinyali daha önce elde edilmiş olan doğrusallaştırılmış sistem denklemlerine uygulanırsa ve Çizelge 3.1’de verilmiş olan değerler aynı şekilde denkleme uygulanırsa, denklem (3.15)’teki transfer fonksiyonuna ulaşılacaktır.

$$T_{GB}(s) = \frac{1}{s^2 + k_1 s + k_2} \quad (3.15)$$

Buradan sisteme ait kıstaslar ise (3.16)’daki denklemlerle elde edilebilir.

$$k_1 = 2\zeta\omega_n \quad (3.16)$$

$$k_2 = \omega_n^2$$

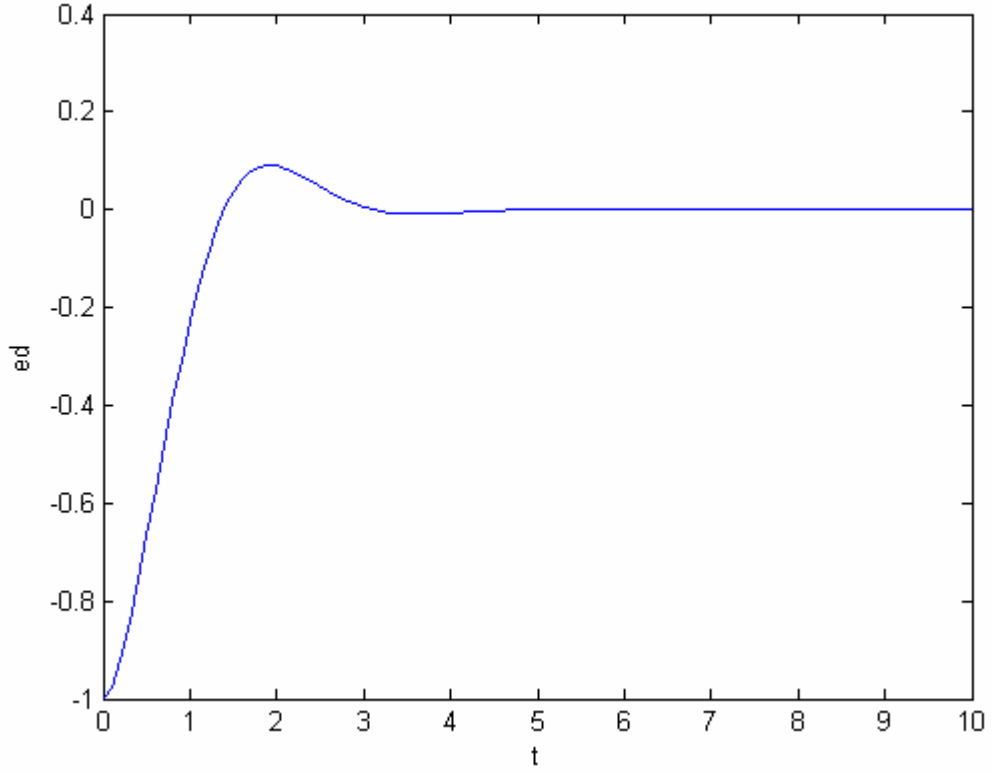
Daha sonra PD Denetleyici simülasyonunda kabul edilmiş olan değerler (Çizelge 3.2) aynı şekilde kabul edilirse, Çizelge 3.4’teki değerler elde edilecektir.

Çizelge 3.4 Durum geri beslemeli denetleyici için seçilen kazançlar

k_1	k_2
2.65	5.07

Görüldüğü üzere elde edilen değerlerin PD Denetleyici için elde edilen değerlerden bir farkı yoktur. Bunun sebebi $c = 1$ şeklinde yapılan seçimdir. Fakat elde edilecek sonucun birebir aynı olması beklenemez. Çünkü gerçek sistemdeki durumlardan ikisi birden kullanılacağı için doğrusallaştırmadan kaynaklanan hatalar sebebi ile farklılıklar olacaktır.

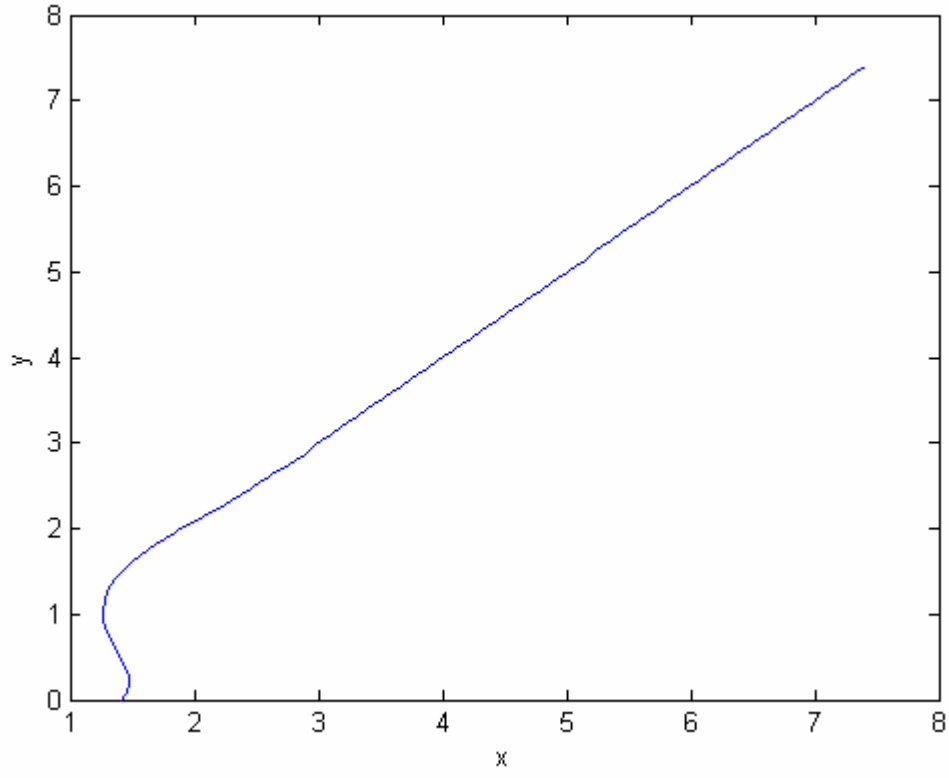
Yapılan bütün bu kabuller ışığında simülasyon gerçekleştirildiğinde, sistemin konum hatasının değişimi Şekil 3.5' teki gibi olacaktır.



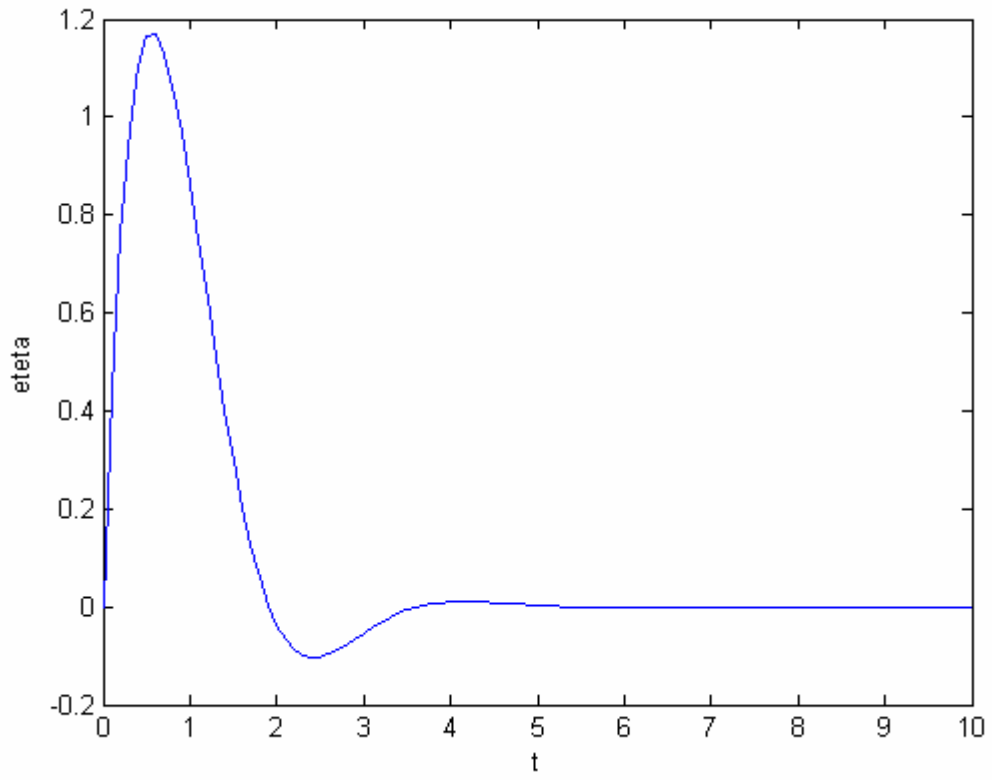
Şekil 3.5 Durum geri beslemeli sistemin cevabı

Aynı şekilde sistemin kartezyen koordinatlardaki hareketi Şekil 3.6' da gösterilmiştir. Aracın oryantasyon hatası durumu ise Şekil 3.7'de verilmiştir.

Elde edilen sonuçların PD denetleyicili sisteme göre çok büyük farklılıklar içermediği açıkça görülmektedir.



Şekil 3.6 Durum geri beslemeli sistemin çizgi takibi



Şekil 3.7 Durum geri beslemeli sistemin oryantasyon hatası

3.4 Sistem İçin Bulanık Mantık Denetleyici Tasarımı

Doğrusal olmayan sistemlerin kontrolünde bulanık mantık denetleyiciler en çok kullanılan denetleyici yapılarından biridir. Bunun sebebi olarak bu denetleyicilerin uzman bilgisine dayalı olarak oluşturulması ve doğrusal olmayan sistemin karmaşık denklemleriyle ilgili herhangi bir ek çalışmanın yapılmaması gösterilebilir.

Bulanık mantık, klasik mantığın aksine iki seviyeli değil, çok seviyeli işlemleri kullanmaktadır. Bulanık mantıkta sayısal ifadeler yerine sözel ifadeler kullanılmaktadır. (Elmas,2003)

Bir sistem hakkında daha fazla bilgi sahibi olundukça sistemin karmaşıklığı azalır ve sistemin anlaşılabilirliği artar. (Ross,1995)

3.4.1 Sistem Hakkındaki Verilerin Sözel Olarak İfade Edilmesi

Sistem hakkında sahip olunan temel verilerden ilki iki hata teriminin de (e_θ, e_d) çizgiye oturma gerçeğleşmesi için sıfıra gitmesi gerektiğidir.

Bu iki hatanın birbiri ile olan ilintisine bakıldığında ise, sadece bir tanesi sıfıra ulaştığında sistemin ilerideki hareketinde bu dengenin bozulacağı açık şekilde sistem denklemlerinden görölmektedir. Örneğın aracın oryantasyon hatasını, uzaklık hatasını göz önünde bulundurmadan, kontrol ederek sıfıra ulaşmasını sağladığımızda, sıfıra ulaşmanın gerçeğleştiğı andan itibaren araç takip edeceğı çizgiye paralel bir yol çizmeye başlar ama uzaklık hatası sıfırlanmayıp sabit bir değere ulaşmış olacaktır.

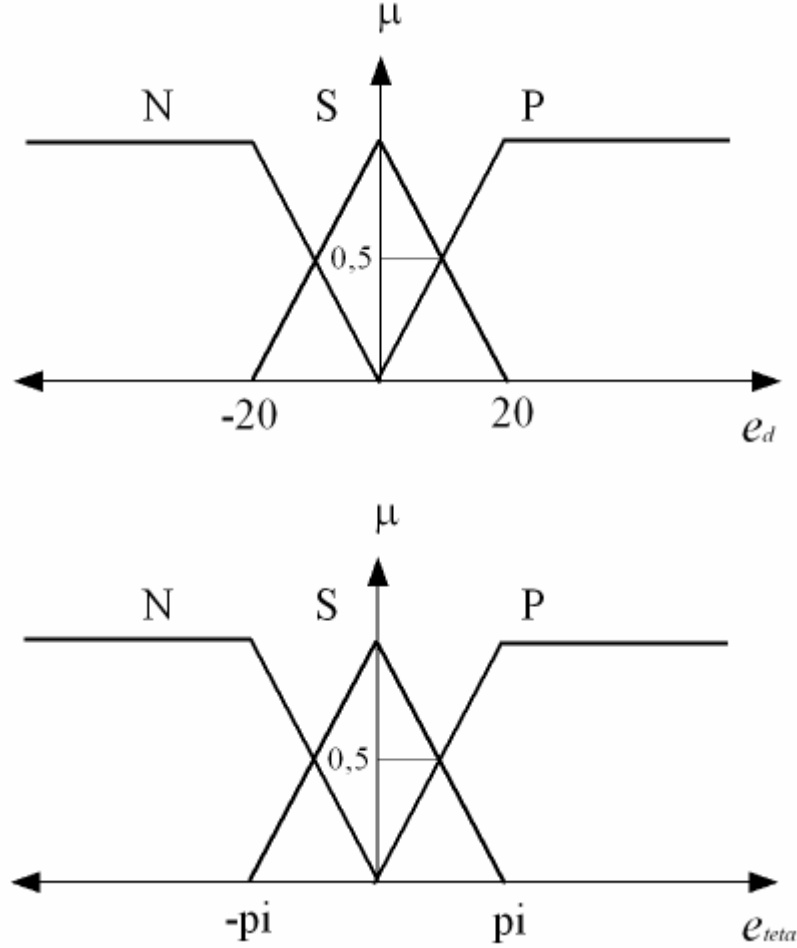
Aracın sadece konum hatasını sıfırlamaya çalıştığımızda ise, araç çizgi üzerine ulaştığında aç ı hatası sıfırlanmamış olacağından, bir sonraki anda tekrardan çizgiden çıkacaktır.

Bu verilere dayanarak şöyle bir uzman bilgisine ulaşılabilir, eğer araç yoldan uzaktaysa, aç ı hatasını sıfırlamaya çalışmak yerine, konum hatasına göre ters işaretle bir aç ı hatası oluşturmaya çalışılmalıdır. Yoldan uzaklaştıkça bu değ er yola dik olana dek arttırılabilir.

Böylece araç çizgiden uzaklaştıkça çizgiye olan yönelimi daha da artacak, fakat araç çizgiye yaklaştıkça, aç ı hatasını da sıfırlayacaktır. Böylece sadece bir hatayı değil, iki hatanın birden sıfırlanmasına yönelik bir kontrol yapısı ortaya çıkacaktır.

3.4.2 Üyelik Fonksiyonlarının Ve Kural Tablosunun Belirlenmesi

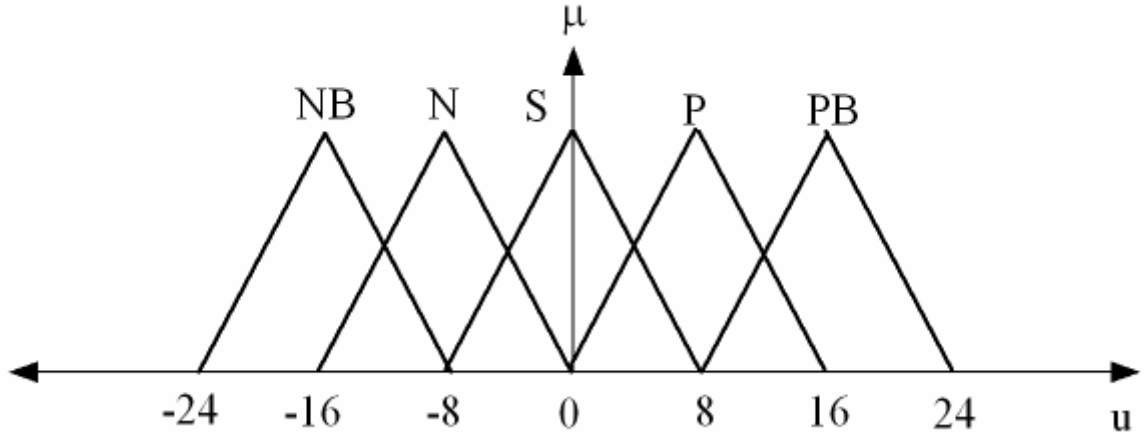
Önceki kısımda elde edilen sözel veriler ışığında hatalara ait giriş üyelik fonksiyonları Şekil 3.8’ deki gibi elde edilecektir.



Şekil 3.8 Hatalara ait giriş üyelik fonksiyonları

Giriş için üçgen üyelik fonksiyonları tercih edilmiştir. Üç üyelik kümesinin yeterli olacağı öngörülmüştür. İki hata için belirlenen bu fonksiyonların tek farkı fonksiyon için belirlenen sabitlerdir.

Aynı şekilde çıkış için uyarlanan çıkış fonksiyonu ise Şekil 3.9’ da gösterilmiştir.



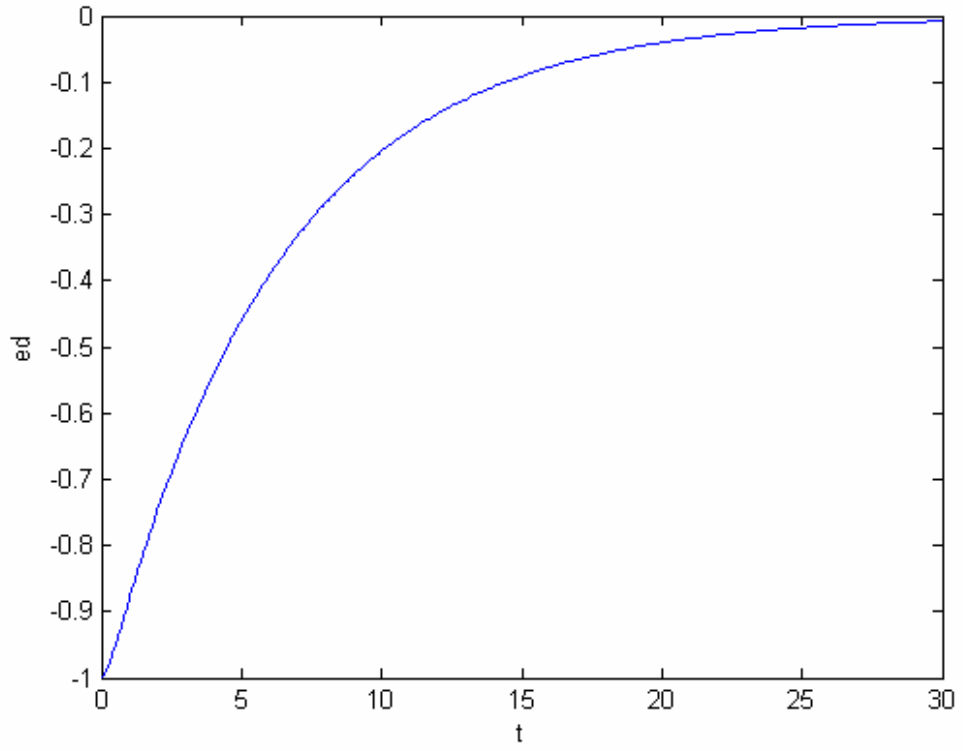
Şekil 3.9 Denetleyiciye ait çıkış üyelik fonksiyonu

Son olarak denetleyicinin tamamlanması için kural tablosunun oluşturulması gerekmektedir. Çizelge 3.5’ te bulanık denetleyici için oluşturulan kural tablosu mevcuttur.

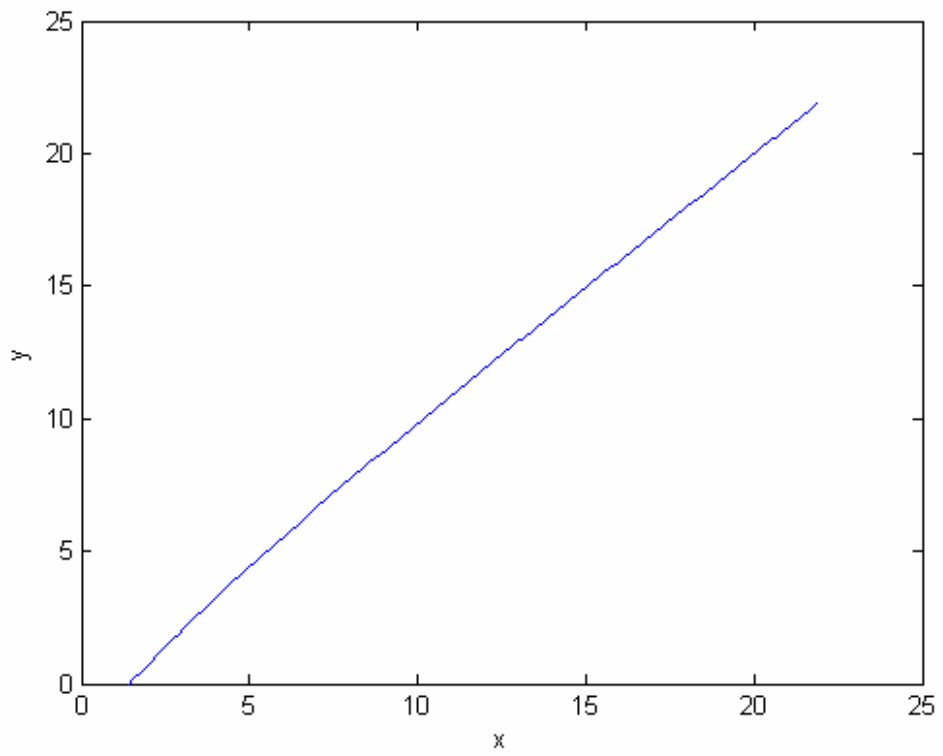
Çizelge 3.5 Bulanık denetleyici kural tablosu

$e_d \setminus e_\theta$	P	S	N
P	PB	P	S
S	P	S	N
N	S	N	NB

En son olarak durulama metodu olarak min-max metodu seçilirse denetleyici tamamlanmış olur. Daha önceki denetleyicilerde kullanılan simülasyon şartları (Çizelge 3.2) ele alındığında sisteme ait konum hatası grafiği Şekil 3.10’ da görülmektedir. (üst aşım ve oturma zamanı gibi kıstaslar denetleyici tasarımında yapı gereği dikkate alınamamıştır.) Sistemin kartezyen koordinatlarda çizdiği yol ise Şekil 3.11 de görülmektedir.

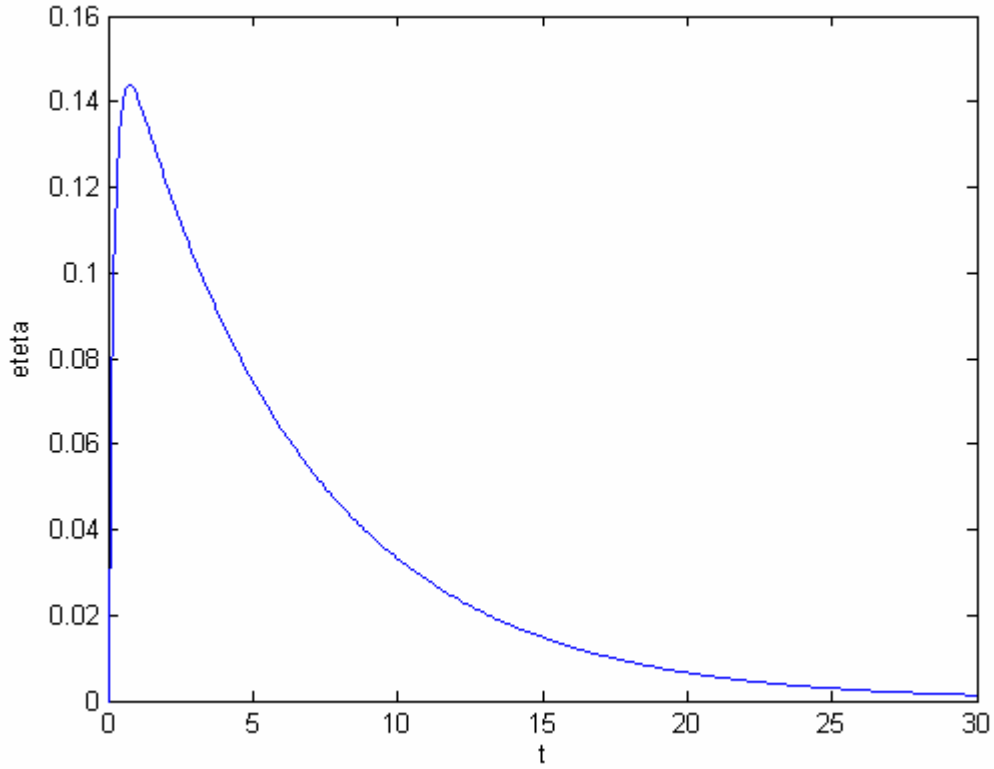


Şekil 3.10 Bulanık denetleyicili sistemin cevabı



Şekil 3.11 Bulanık denetleyicili sistemin çizgi takibi

Son olarak sistemin oryantasyon hatasının grafiği Şekil 3.12 de görülmektedir.



Şekil 3.12 Bulanık denetleyicili sistemin oryantasyon hatası

Elde edilen sonuçlar bulanık mantık denetleyicisinin sistemin kontrolünde daha başarılı olduğunu göstermektedir.

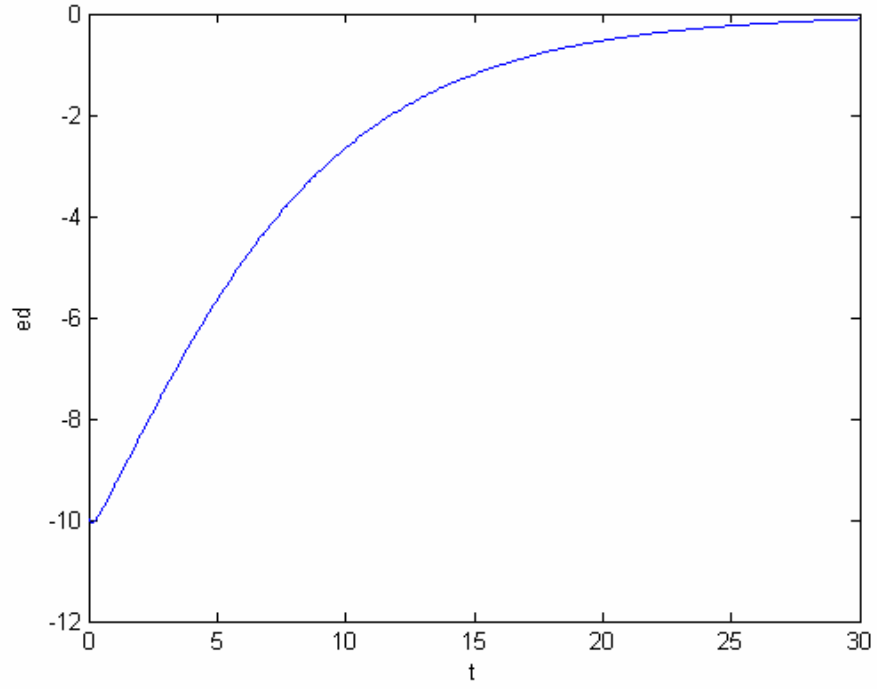
Bütün bu sonuçlara ilaveten, bulanık mantık denetleyiciler doğrusal olmayan sistemlerin kontrolünde doğrusallaştırma yapmaya gerek olmadan ya da kontrol edilecek değerlere sınırlama koymadan kullanılabilir.

Bu verinin ışığında bulanık mantık denetleyici Çizelge 3.6'da verilen daha zorlayıcı bir veri setinin denetlenmesi için kullanılmıştır.

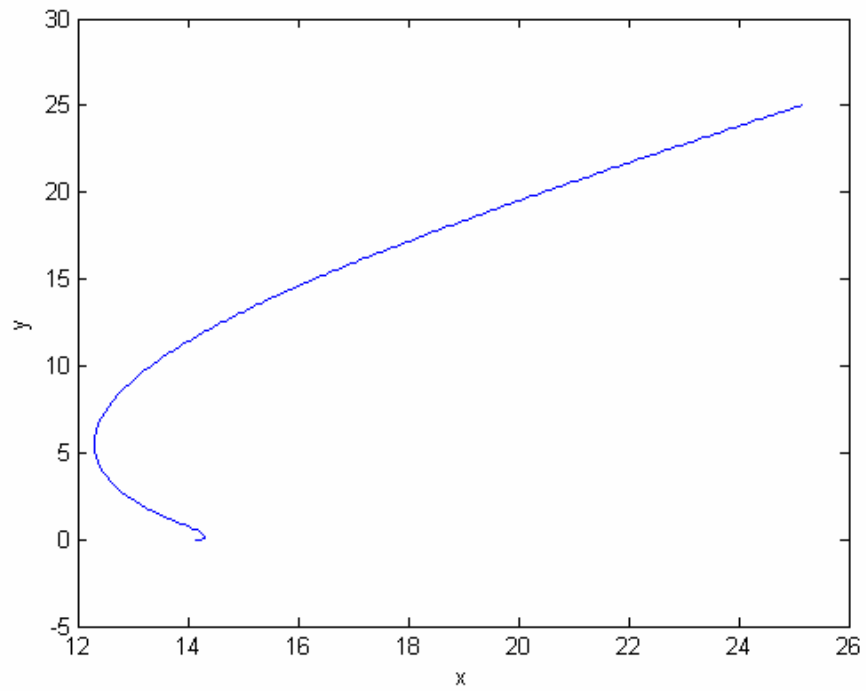
Çizelge 3.6 Bulanık mantık denetleyici için ikinci simülasyon verileri

c	L	$e_{\theta}(t=0)$	$e_d(t=0)$
1 m/s	1 m	$-\frac{\pi}{4}$	10 m

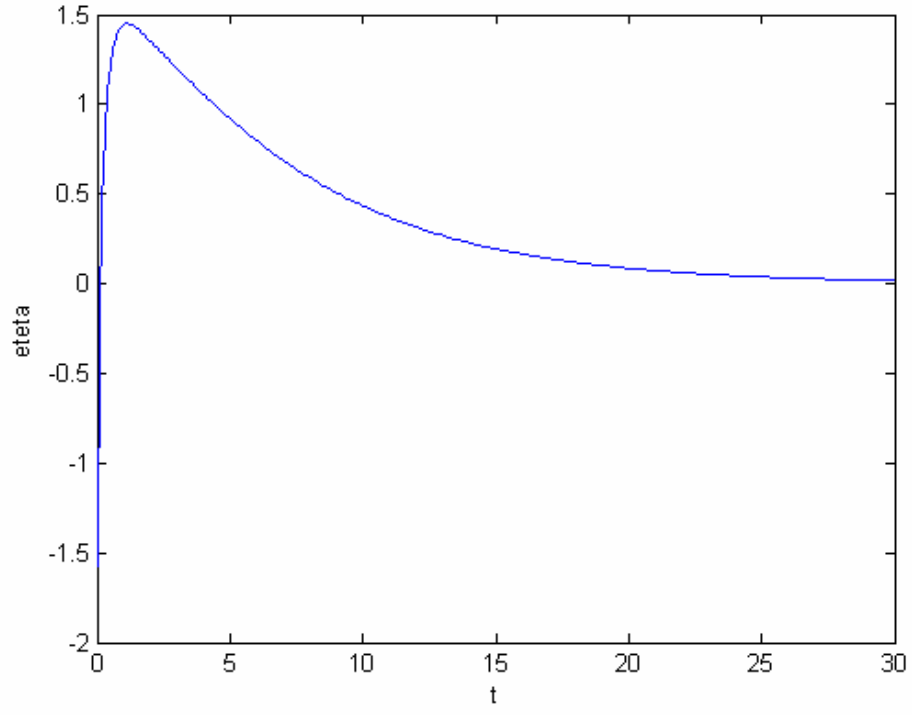
Bu simülasyon şartlarına ait bütün çizimler sırasıyla Şekil 3.13, Şekil 3.14 ve Şekil 3.15’ te verilmiştir.



Şekil 3.13 Bulanık denetleyicili sistemin ikinci veri seti için cevabı



Şekil 3.14 Bulanık denetleyicili sistemin ikinci veri seti için çizgi takibi

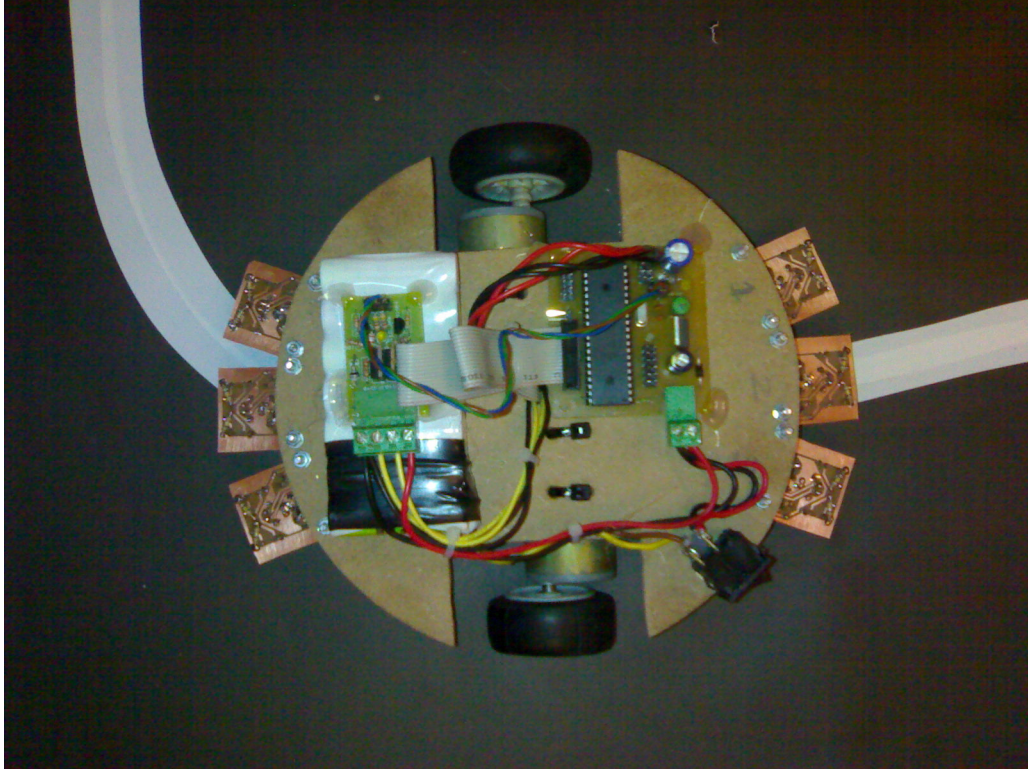


Şekil 3.15 Bulanık denetleyicili sistemin ikinci veri seti için oryantasyon hatası

4. UYGULAMA

4.1 Uygulama düzeneğinin yapısı

Uygulama için diferansiyel sürüş yapısının kullanıldığı bir düzenek oluşturulmuştur. Düzenekte iki adet redüktörlü motor ve bir adet serbest dönebilen tekerlek kullanılmıştır. Tek serbest dönen tekerlek kullanıldığı için aracın ağırlık merkezi aracın geometrik merkezinden serbest dönen tekerlek yönünde kaydırılmıştır. Şekil 4.1’ de aracın pist üzerindeki duruşu görülmektedir.



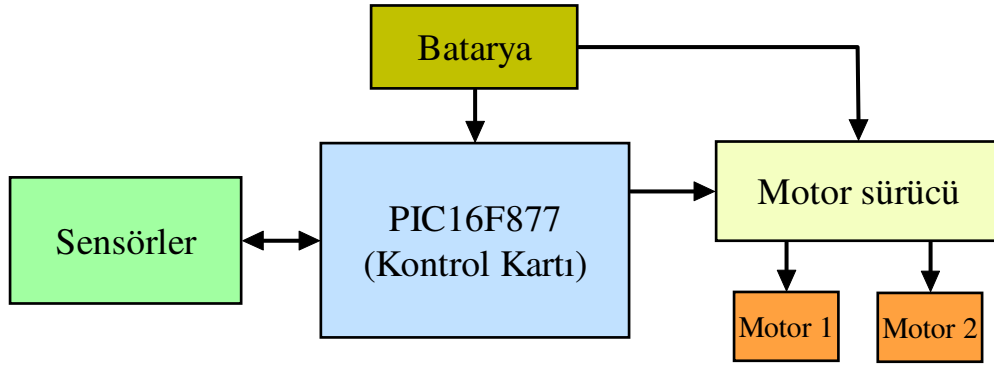
Şekil 4.1 Aracın pist üzerindeki görünümü

Sensörler araç etrafındaki zahiri bir dairenin üzerindeymiş gibi yerleştirilmiştir. Buradaki amaç aracın açı ve konum hatalarının daha iyi bir yaklaşımla hesaplanmasıdır.

4.2 Uygulama düzeneğinin donanımı

Araç elektriksel donanım olarak, bir adet batarya, bir kontrol kartı, altı adet iki sensörlü yol algılayıcı kart ve bir adet çift motor sürücü içeren devreden oluşmaktadır.

Kontrol kartı üzerinde Microchip firmasının PIC16F877 mikro denetleyicisi bulunmaktadır. Sistem yapısını açıklayan bir diyagram Şekil 4.2’ de gösterilmiştir.



Şekil 4.2 Uygulama donanımının blok gösterimi

Donanımda kullanılan sensörler altı adet önde ve altı adet arkada olmak üzere toplam oniki adettir. Her sensörde yola kızıl ötesi ışık gönderen bir adet kızıl ötesi ışık veren LED ve bu ışığı algılayan bir alıcı bulunmaktadır.

Kızıl ötesi alıcı beyaz bir zemin üzerinde daha fazla ışık alması sonucunda içinden geçen akım miktarı artmakta ve kurulan bir karşılaştırmacı düzenekle bu durum kıyaslanmaktadır. Karşılaştırmacı ilgili sensörün gördüğü durumlarda mikro denetleyicinin ilgili pinine 5 voltluk bir gerilim uygulamaktadır.

Motor sürücü, mikro denetleyicinin ürettiği darbe genlik modülasyon sinyalini yükselterek motorlara yansıtmaktadır.

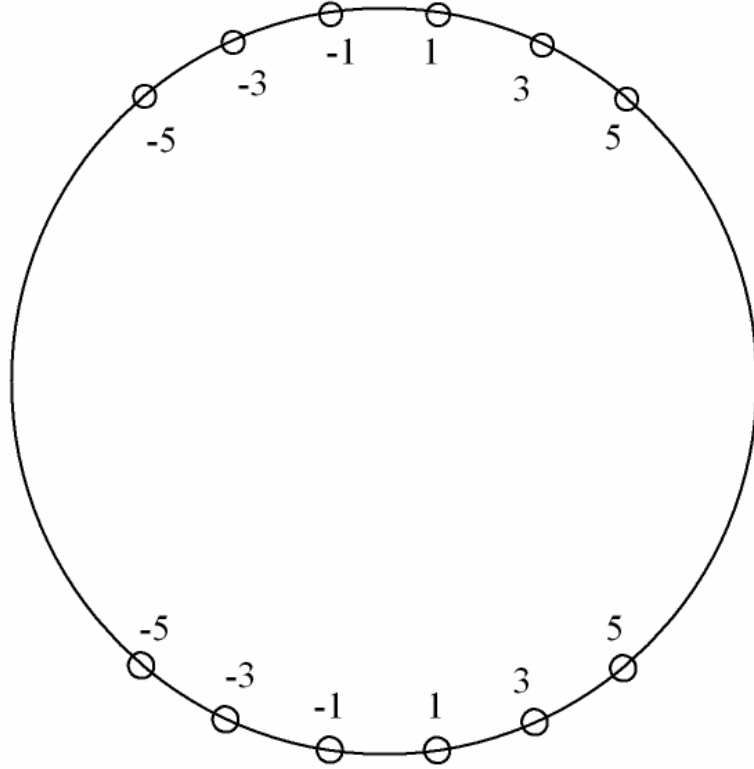
4.3 Uygulama düzeneğinin yazılımı

Mikro denetleyici sensörlerden aldığı sinyalleri kullanarak içinde bulunan algoritma yardımıyla uygun darbe genlik değerlerini hesaplar ve bunları motor sürücüyeye gönderir. Böylelikle kapalı çevrim bir sistem oluşmuş olur.

4.3.1 Hataların hesaplanması

Sensörlerden alınan veriler, her sensöre belirli bir ağırlık katsayısı verilerek konum ve açı hatasına dönüştürülür.

Konum hatasını temsil etmek üzere sensörlere verilen ağırlıklar Şekil 4.3' te gösterilmiştir.



Şekil 4.3 Konum hatası için sensör ağırlıkları

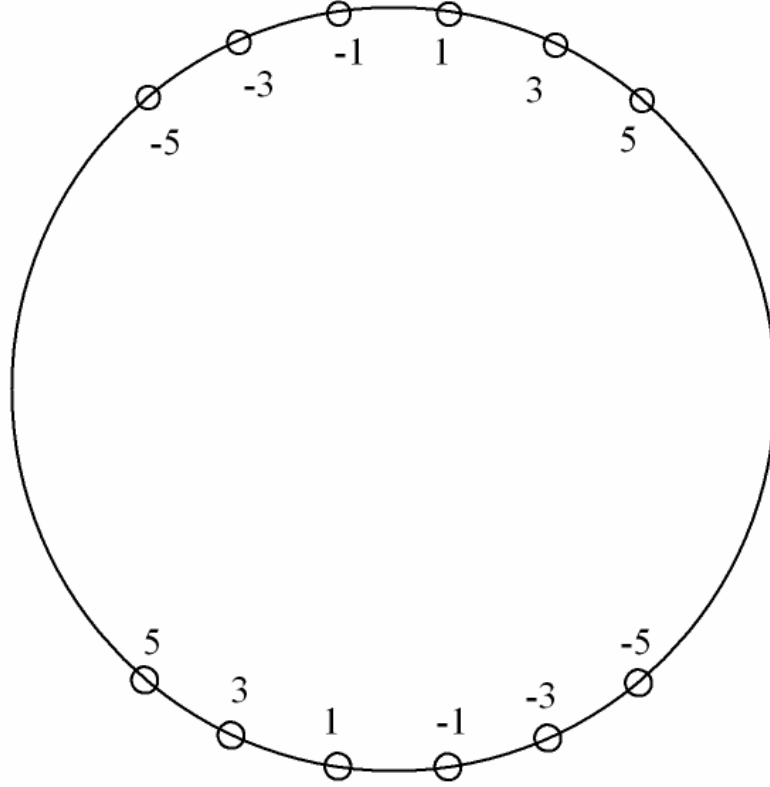
Konum hatasının yaklaşık hesabı için gören sensörler, ilgili ağırlık katsayısı ile çarpılarak gene gören sensörlere göre ortalama alınır. Buradan elde edilen değer konum hatasının belirli bir oranıdır. Örneğin en soldaki iki sensörün görmesi durumunda konum hatası miktarı (4.1)' de hesaplanmıştır.

$$e_d = \frac{-5 - 5}{2} = -5 \quad (4.1)$$

Aynı anda iki veya daha çok sensörün görmesi durumunda ise alınan ortalama sayesinde ara değerler elde edilebilmektedir. Örneğin soldaki sensörlerden üstten en soldan iki ve alttan en soldaki sensörün görmesi durumunda ise konum hatasının sonucu (4.2)' de görülmektedir.

$$e_d = \frac{-5 - 3 - 5}{3} = -4.33 \quad (4.2)$$

Benzer şekilde açı hatası içinde ağırlıklar belirlenmiştir. Buna ait diyagram Şekil 4.4' de görülmektedir.



Şekil 4.4 Açı hatası için sensör ağırlıkları

Benzer yaklaşımla açı hatası da elde edilmektedir. Örneğin aracın en soldan üstteki ve alttaki birer sensörünün görmesi durumunda açı hatası (4.3)' deki gibi olacaktır.

$$e_{\theta} = \frac{5 - 5}{2} = 0 \quad (4.3)$$

Böyle bir durumda araç yola paralel gitmekte, konum hatası bulunmasına karşın, açı hatası oluşmamaktadır.

4.3.2 Algoritmanın oluşturulması

Sistemde kontrol yöntemi olarak tam durum geri beslemesi kullanılmıştır. Uzman bilgisine dayalı olarak iki hata oranı eş miktarda ağırlıklara sahip olacak şekilde toplanarak toplam hata elde edilmiştir. Denklem (4.4)' te toplam hata terimi gösterilmiştir.

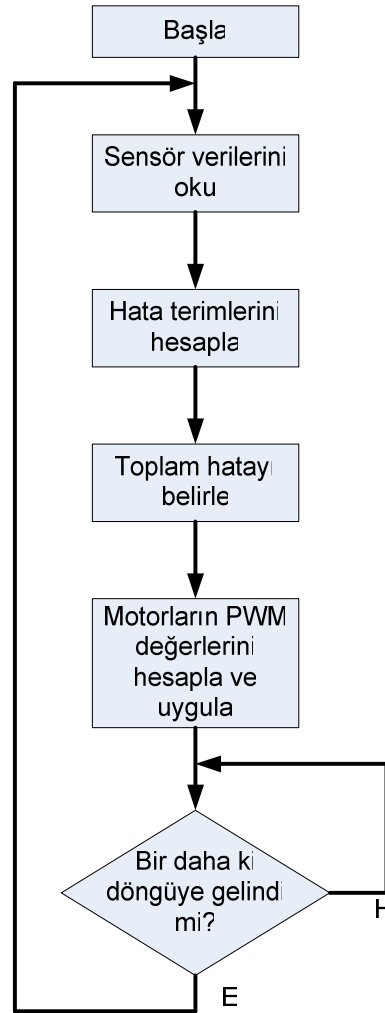
$$e_{tot} = e_d + e_{\theta} \quad (4.4)$$

Son olarak elde edilen toplam hata terimi belirli bir kazanç değeri ile çarpılarak, aracın sağ tekerleğine azalım, sol tekerleğine ise artım olarak yansıtılmaktadır.

Daha önce yapılan tam durum geri beslemesi yaklaşımında olduğu gibi iki tekerlek hızının toplamı sabit tutulmuştur. Hata terimi bir tekerleği yavaşlatırken, diğerini ise aynı miktarda hızlandıracaktır.

Darbe genlik modülasyonunun motor geriliminin bir oranı olduğu kabul edilerek, bununda tekerlek hızının bir oranı olduğu kabulünde bulunularak, tekerleklerden hız geri beslemesi alınmamaktadır.

Sistem algoritmasına ait akış diyagramı ise Şekil 4.5’ te görülmektedir.



Şekil 4.5 Uygulama algoritmasının akış diyagramı

5. SONUÇLAR ve TARTIŞMA

Elde edilen sonuçlardan görülmüştür ki, doğrusal olmayan sistemlerin doğrusallaştırılması sonucu oluşturulan doğrusal yapıya sahip denetleyiciler kullanıldığında doğrusal olmayan sistemler tanımlanan belirli hata bantları için doğrusal sistemler gibi davranmış ve basit hesaplamalar sayesinde kolaylıkla denetlenebilmiştir.

Fakat bu şekilde yapılan kontrolde sistem için istenen kıstaslar tam olması gerektiği şekilde elde edilememiş, ayrıca yapılan ek çalışmalarda hataların bazı değerleri için sistemin kararsız olduğu gözlemlenmiştir.

Bulanık mantık denetleyici kullanımı sayesinde, doğrusal olmayan sistem kolaylıkla denetlenmiş ve varılan sonuçlar bize aracın oldukça kötü başlangıç şartlarına rağmen çizgiye oturmayı başardığını göstermiştir.

Buna rağmen bulanık mantık denetleyici tasarımı yaparken, sistem cevabında görmek istediğimiz kıstasları ($\%OS, T_s$) sağlamaya yönelik bir tasarım yapma imkanımız genelde yoktur.

Uygulamada ise bulanık mantık denetleyicinin gerçekleşmesi mikro denetleyicinin işlem kapasitesinin yetersizliği nedeni ile imkansız olmuştur. Buna karşın sistemde oluşabilecek hata miktarlarının küçük olması nedeni ile sistemin denetlenmesinde tam durum geri beslemesinin yeterli olduğu görülmüştür.

KAYNAKLAR

- Akbatı, O. ve Cansever, G., (2007), “Depo Robot Sistemi”, 4. Otomasyon Sempozyumu, 19 Mayıs Üniv., 2007, Samsun.
- Almeida, L., Mota, A., Fonseca, P., (1998), “Comparing Control Strategies for Autonomous Line-tracking Robots”, Advanced Motion Control, 1998. AMC '98-Coimbra.
- Canudas, C., Siciliano, B., Bastin, G., (1996), Theory of Robot Control, Springer Verlag, London.
- Dixon, W.E., Dawson, D.M., Zergeroğlu, E., Behal, A., (2001), Nonlinear Control of Wheeled Mobile Robots, Springer Verlag, New York.
- Dudek, G., Jenkin, M. (2000), Computational Principles of Mobile Robots, Cambridge University Press, New York.
- Elmas, Ç., (2003), Bulanık Mantık Denetleyiciler, Seçkin, Ankara.
- Ishihara, H., Fukuda, T., Kosuge, K., Arai, F., (1994), “An Approach to Autonomous Microrobot - Micro Line Trace Robot with Reflex Algorithm”, Emerging Technologies and Factory Automation, 1994, Tokyo.
- Kuo, B.C., (1995), Automatic Control Systems, Prentice Hall, Englewood Cliffs, New Jersey.
- Ogata, K., (1994), Solving Control Engineering Problems with MATLAB, Prentice Hall, Englewood Cliffs, New Jersey.
- Ross., T.J., (1995), Fuzzy Logic with Engineering Applications, Mc Graw Hill, New York.
- Skogestad, S. ve Postlethwaite, I., (1996), Multivariable Feedback Control: Analysis and Design, John Wiley & Sons, New York.
- Şahin, H.T., Zergeroğlu, E., (2006), “Sınırlı Algılama Alanı Olan Tekerlekli Mobil Robotlar için Holonom Olmayan Bir Rota Planlama Yöntemi” , Türk Otomatik Kontrol Toplantısı, TOK, TOBB Üniv., Ankara.

ÖZGEÇMİŞ

Doğum tarihi 24.05.1983

Doğum yeri İstanbul

Lise 1993-2001 Kartal Burak Bora Anadolu Lisesi

Lisans 2001-2005 Yıldız Üniversitesi Elektrik-Elektronik Fak.
Elektrik Mühendisliği BölümüYüksek Lisans 2005-2007 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektrik Müh. Anabilim Dalı
Kontrol ve Otomasyon Programı**Çalıştığı kurum(lar)**

2005-2005 Teksan Jeneratör, Ar-Ge Mühendisi

2005- YTÜ Fen Bilimleri Enstitüsü, Araştırma Görevlisi