

**BAŐKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**GENİŐLETİLMİŐ KALMAN SÜZGECİ İLE GEZGİN ROBOT
KONUMUNUN BELİRLENMESİ**

**MOBILE ROBOT LOCALIZATION WITH EXTENDED
KALMAN FILTER**

K. HASAN HATİPOĐLU

YÜKSEK LİSANS TEZİ

ANKARA

OCAK, 2007

**GENİŞLETİLMİŞ KALMAN SÜZGEÇİ İLE GEZGİN ROBOT
KONUMUNUN BELİRLENMESİ**

**MOBILE ROBOT LOCALIZATION WITH EXTENDED
KALMAN FILTER**

K. HASAN HATİPOĞLU

Başkent Üniversitesi
Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin
ELEKTRİK-ELEKTRONİK Mühendisliği Anabilim Dalı İçin Öngördüğü
YÜKSEK LİSANS TEZİ
olarak hazırlanmıştır.

OCAK, 2007

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma, jürimiz tarafından **ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Üye (Danışman)

.....

Yrd. Doç. Dr. Hamit Erdem

Üye

.....

Yrd. Doç. Dr. Mustafa Doğan

Üye

.....

Yrd. Doç. Dr. Mustafa Sert

ONAY

Bu tez/...../2007 tarihinde Enstitü Yönetim Kurulunca belirlenen yukarıdaki jüri üyeleri tarafından kabul edilmiştir.

...../...../2007

Prof.Dr. Emin AKATA

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

TEŐEKKÜR

Bu tez alıőmasının yapılmasında deęerli grüşlerine ve engin bilgilerine başvurduğum, deęerlendirmeleri ile yardım ve destek veren, tez alıőmasının her aşamasında yol gösterici olan saygıdeęer hocam ve danışmanım Yrd. Doę. Dr. Hamit Erdem'e; tez alıőması süresince bilgilerinden ve önerilerinden yararlanılmasına olanak saęlayan, ilgisini esirgemeyen ok deęerli hocam Yrd. Doę. Dr. Mustafa Doęan'a; tez alıőmasında deęerli grüş ve düşüncelerini belirten, hataların düzeltilmesinde yardımcı olan sayın hocam Yrd. Doę. Dr. Mustafa Sert'e, rahat bir alıőma ortamı ile beraber imkanların kullanılmasına olanak saęlayan Prof. Dr. Emin Akata'ya teőekkürlerimi sunarım.

Maddi manevi desteklerini hiçbir zaman eksik etmeyen, her zaman yanımda olan en deęerli varlığım aileme sonsuz teőekkürler.

ÖZ

GENİŞLETİLMİŞ KALMAN SÜZGEÇİ İLE GEZGİN ROBOT KONUMUNUN BELİRLENMESİ

K. Hasan Hatipođlu

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliđi Anabilim Dalı

Otonom gezgin robotların işlevlerini yerine getirebilmeleri için en önemli ve gerekli özelliklerden biri kendi konumlarını bilmeleridir. Robotlar kendi konumlarını bildiđi sürece algıladıkları ortamın özelliklerini doğru şekilde analiz edebilirler. Genel olarak robot konumunun belirlenmesi için iki tip ölçüm yöntemi kullanılır. Bu yöntemler mutlak ve görelî ölçüm yöntemleridir. Çođunlukla bu iki ölçüm yöntemi bir arada kullanılır.

Bu tez çalışmasında, robot konumunun belirlenmesi için mutlak ve görelî ölçüm yöntemlerinden birlikte yararlanılmıştır. Mutlak ölçüm için ses ötesi algılayıcıları, görelî ölçüm için optik şaft kodlayıcıları kullanılmıştır. Bu algılayıcıların ve robot sisteminin kinematik modellemesi yapılmış, gezgin robotun yöneliminde ve konumunda odometriden kaynaklanan hataların azaltılabîlmesi için genişletilmiş Kalman süzgeci kullanılmıştır.

Deneyisel çalışmada, üzerinde 8 adet ses ötesi algılayıcı ve tekerlekleri üzerinde iki adet optik kodlayıcı bulunan, Pioneer 3-DX otonom gezgin robotu kullanılmıştır. Robot konumunun belirlenmesi kapalı oda içi ortamında gerçekleştirilmiştir. Odometri ve algılayıcı birleştirilmesi yöntemleri kullanılarak yapılan gezgin robot konumunun belirlenmesindeki verimlilik, benzetim ve fiziksel ortamlardaki deneyler ile karşılaştırılmıştır. Matlab programı ile gerekli yazılımlar geliştirilmiştir. Gezgin robotun son pozisyon hatasının GKS kullanımı ile azaldığı gözlenmiştir.

Anahtar Sözcükler: Gezinim, Konum Belirleme, Genişletilmiş Kalman Süzgeci, Odometri

Danışman: Yrd.Doç.Dr. Hamit Erdem, Başkent Üniversitesi, Elektrik-Elektronik Mühendisliđi Bölümü.

ABSTRACT

MOBILE ROBOT LOCALIZATION WITH EXTENDED KALMAN FILTER

K. Hasan Hatipoğlu

Başkent University Institute of Science

The Department of Electrical and Electronics Engineering

One of the most important and necessary features of the autonomous mobile robots to perform their functions in the environment is their ability to know their locations. Robots can analyze the features of their environment correctly as long as they know their own positions. In general, two types of measurement methods are used for robot localization. They are absolute and relative measurement methods. Usually, these two measurement methods are used together.

In this thesis, both absolute and relative measurement methods were utilized to localize the robot. Sonar sensors and optical shaft encoders were used for absolute and relative measurements respectively. These sensors and kinematical robot systems were modeled, and extended Kalman filter was used in order to reduce the odometric errors in orientation and position of the mobile robot.

In experimental work, Pioneer 3-DX, an autonomous mobile robot with 8 sonars and two optical encoders located on the wheels was used. Localization of the mobile robot was performed in a closed indoor environment. The efficiency of the mobile robot localization performed by using odometry and sensor fusion methods were compared with the simulation and physical environment experiments. The required software was developed by using Matlab. It is observed that last position error of the mobile robot is decreased with the use of EKF.

Keywords: Navigation, Localization, Extended Kalman Filter, Odometry

Advisor: Asst. Hamit Erdem, Başkent University, The Department of Electrical and Electronics Engineering.

İÇİNDEKİLER LİSTESİ

Sayfa

TEŞEKKÜR.....	
ÖZ	i
ABSTRACT	ii
İÇİNDEKİLER LİSTESİ	iii
ŞEKİLLER LİSTESİ.....	v
ÇİZELGELER LİSTESİ.....	vii
SİMGELER VE KISALTMALAR LİSTESİ	viii
1. GİRİŞ	1
2. KONUM BELİRLEME	5
2.1. Konum Belirleme Problemi	5
2.2. Uygun Veriler.....	6
2.3. Görelî Konum Ölçümleri	9
2.4. Mutlak Konum Ölçümleri	10
2.5. Algılamaların Birleştirilmesi	12
3. OLASILIKSAL KONUM BELİRLEME YAKLAŞIMI	13
3.1. İnanış İşlevi	13
3.2. Hareket Ve Algılama Modeli	14
3.3. Olasılıksal Konum Belirleme.....	15
3.4. Konum Belirleme Yöntemleri	18
4. RASGELE SÜREÇLER VE RASSAL SİSTEMLER	23
4.1. Olasılık Ve Rasgele Değişkenler	23
4.2. Rasgele Değişkenlerin İstatistiksel Özellikleri.....	25
5. KALMAN SÜZGEÇİ.....	30

5.1.	Kalman Süzgecinin Kullanım Alanları.....	31
5.2.	Genel Kavram.....	32
5.3.	Kabuller	35
5.4.	Doğrusal Kalman Süzgeci	39
5.5.	Genişletilmiş Kalman Süzgeci	45
6.	ROBOT SİSTEMİ VE MODELLER	49
6.1.	ActivMedia Robotlar	49
6.2.	Sistem Modeli	53
6.3.	Ölçüm Modeli.....	59
7.	PİONEER 3-DX ROBOT KONUMUNUN İZLENMESİ	64
7.1.	Benzetim Ortamı.....	65
7.2.	Gerçek Ortam	86
8.	SONUÇ	96
	KAYNAKLAR LİSTESİ	99
	ÖZGEÇMİŞ	102
	EKLER	103

ŞEKİLLER LİSTESİ

	<u>Sayfa</u>
Şekil 3.1. Robotun olası konumları.....	19
Şekil 3.2. Ortamın Izgara Harita ile Gösterimi	21
Şekil 4.1 Gauss yoğunluk işlevi	28
Şekil 5.1 Kalman Süzgeci Döngüsü	32
Şekil 5.2 Standart model-tabanlı Kalman Süzgeci tasarım şekli.....	34
Şekil 5.3 Beyaz gürültü gösterimi: zaman tabanında (solda), sıklık tabanında (sağda).....	37
Şekil 5.4 Doğrusal Kalman Süzgeci Algoritması.....	45
Şekil 5.5 Genişletilmiş Kalman Süzgeci Algoritması.....	48
Şekil 6.1 Bütünsel ortamda robot konumu.....	53
Şekil 6.2 Denetim girdileri ile robotun hareketi	53
Şekil 6.3 Robotun dönüş hareketi.....	55
Şekil 6.4 Robotun dönüş hareketi.....	56
Şekil 6.5 Ses ötesi algılayıcısının tipik yoğunluk dağılımı.....	59
Şekil 6.6 Ses ötesi algılayıcısının T.O.F ölçümü	59
Şekil 6.7 Sonar algılayıcıların konumu ve algılama doğrultusu	60
Şekil 6.8 d uzaklığı, engel uzaklığı	61
Şekil 7.1 Robot hareketi programı akış diyagramı.....	64
Şekil 7.2 MobileSim programının görünümü.....	65
Şekil 7.3 Robot gezinimi için ortam	67
Şekil 7.4 (a) robotun gerçek hareketi; (b) robotun odometrik hareketi.....	67
Şekil 7.5 (a) gerçek doğrusal hız; (b) varsayılmış doğrusal hız	69
Şekil 7.6 (a) robotun gerçek hareketi; (b) robotun odometrik hareketi.....	70
Şekil 7.7 (a) robotun gerçek hareketi; (b) robotun odometrik hareketi.....	72

Şekil 7.8 Ortam haritası.....	74
Şekil 7.9 (a) robotun ortamdaki hareketi; (b) robotun odometrik veri ile hareketi..	75
Şekil 7.10 Izgara tabanlı harita tanımlamalı ortam	77
Şekil 7.11 Izgara harita tanımlamalı ortamda engel tespiti	77
Şekil 7.12 Sonar verisindeki hata hesabı programının akış diyagramı	78
Şekil 7.13 Robotun KS ile kestirilmiş hareketi	79
Şekil 7.14 Robotun bir önceki durum belirsizliği P_k^-	81
Şekil 7.15 8 adet sonardan alınan verilerle hesaplanmış gürültü değerleri	82
Şekil 7.16 Sonarların gürültü varyansları R_k	83
Şekil 7.17 Kalman Kazanç matrisindeki her elemanın değişimi.....	84
Şekil 7.18 Robotun bir sonraki durum belirsizliği P_k^+	85
Şekil 7.19 Robotun odometrik hareketi.....	87
Şekil 7.20 Robotun KS ile hareketi.....	88
Şekil 7.21 Robotun denetim girdilerinin varyansları.....	90
Şekil 7.22 Robotun bir önceki durum belirsizliği P_k^-	91
Şekil 7.23 8 adet sonardan alınan verilerle hesaplanmış gürültü değerleri	92
Şekil 7.24 Sonarların gürültü varyansları R_k	93
Şekil 7.25 GKS algoritmasındaki Kalman kazanç matrisi	94
Şekil 7.26 GKS algoritmasındaki bir sonraki durum belirsizliği P_k^+	95

ÇİZELGELER LİSTESİ

	<u>Sayfa</u>
Çizelge 5.1: Sistem modeli parametreleri	35
Çizelge 5.2: Ölçüm modeli parametreleri	36
Çizelge 6.1 Pioneer 3-DX Ön Sonarlarının Konumları	52
Çizelge 6.2 Robot yapısına bağlı birkaç parametre	52
Çizelge 7.1 Sonar verileri(mm)	66
Çizelge 7.2 Denetim girdileri.....	66
Çizelge 7-3 Hata oranları	71
Çizelge 7-4 Hata oranları	73
Çizelge 7.5 Hata oranları.....	80
Çizelge 7.6 Hata oranları.....	89

SİMGELER VE KISALTMALAR LİSTESİ

GPS	:	bütünsel konumlandırma sistemi
r.d.	:	rassal değişken
h.o.t.	:	yüksek mertebeli terimler
DA	:	doğru akım
D.H.D	:	doğrusal hız değişimi
KS	:	Kalman Süzgeci
GKS	:	Genişletilmiş Kalman Süzgeci
T.O.F	:	Uçuş süresi (time of flight)

x_k	:	k zamanındaki durum
x_{k-1}	:	k-1 zamanındaki durum
u_{k-1}	:	denetim girdileri
w_{k-1}	:	sistem gürültüsü
A	:	Durumun zamana göre dönüşüm işlevi
B	:	Denetim girdileri ile sistem arası dönüşüm işlevi
z_k	:	k zamanındaki ölçüm
v_k	:	ölçüm gürültüsü
H	:	sistem ile ölçüm arasındaki dönüşüm işlevi

1. GİRİŞ

Günümüzde robotlar birçok alanda kullanılmaktadır. Artan kullanım alanı robotların çok farklı ortamlarda çalışabilmeleri ihtiyacını doğurmuştur. Bu nedenle, robotların farklı ortamlarda çalışabilmeleri için geliştirme çalışmaları hız kazanmıştır. Robotların gelişimi açısından birçok çalışma yürütülmektedir. Robotların birçoğu bilgisayar ya da işlemciler tarafından denetlenen kollardan oluşmakta, bazıları da tekerlekler, bacaklar veya paletler üzerinde hareket eden yapılardan oluşmaktadır. Bazı robotlar da denizaltı dalışlarında araştırma amaçlı kullanılırlar. Bunun yanında robotlar, başlangıçtaki durağan yapılarına ve işlevselliklerine kıyasla daha gezgin ve daha çok kendi kendilerine hareket edebilen yapılara kavuşmuşlardır.

Robot kelimesi ilk defa Karel Capek tarafından R.U.R (Rossum's universal Robots) adlı eserinde kullanılmıştır. Ancak, genel olarak bilinenin aksine robot kelimesini icat eden kişi Karel Kapek'in abisi Joseph Kapek'tir[1]. Carel Kapek R.U.R. adlı oyununda robot kelimesine karşılık olarak, kendi kendilerine çalışabilen işçiler tanımlamasını yapmıştır.

Gezginlik, robotların dünyadaki yerlerini değiştirebilme yeteneklerinin olup olmaması olarak tanımlanabilir. Robot evriminin başlangıcındaki robotlar endüstriyel amaçlı geliştirildikleri ve büyük ölçekli yapılara sahip olduklarından dolayı gezginlik özelliğine sahip değillerdi. Günümüzde ise robotlar, endüstriyel üretim haricinde günlük hayatımızda ve uzay araştırmalarında kullanılmaya başlanmıştır. Örneğin, yer süpürgesi robotu, posta dağıtım robotu, çöp toplama robotu veya Mars'a gönderilen robotlar gezginlik özelliğine sahiplerdir.

Robotların gezginlik özelliğini yerine getirebilmeleri için değişen durumlara göre hareketlerini değiştirme yetisine sahip olmaları gerekir. Kısacası otonom özellikleri olmalıdır. Otonom olma özelliği, robot üzerinde bulunan işlemci sayesinde robotun daha önceden belirlenmiş veya belirlenmemiş bir durum karşısında karar verebilme yeteneğine sahip olmasıdır. Robot hareketine karar vermeden önce çevresinden veri toplamalıdır. Robotun algılayıcıları aracılığı ile çevreden elde ettiği veriler bütünü sayesinde robot, sonraki hareketine karar verebilir. Örnek olarak Mars'a, Ay'a, insan için tehlikeli olabilecek ortamlara veya

insanın giremeyeceği bir mağaraya gönderilecek robotun otonom olması gerekmektedir.

Gezgin robotların görevlerini yerine getirebilmeleri için bulunduğu ortamı bilmesi veya öğrenmesi, dolayısı ile kendi konumunu bilmesi gerekmektedir. Robotun ortamdaki gezinimi sırasında öncelikle engellerden korunması gerekmektedir. Bu nedenle robotların gözü kulağı olan algılayıcılar önem kazanmaktadır.

Robot, gezinim yapabilmesi için başlıca üç soruya yanıt vermelidir. Bunlar “Neredeyim?”, “Nereye gitmek istiyorum?”, “Oraya nasıl ulaşırım?” sorularıdır[2]. İlk soru robotun ortamda nerede olduğunu yanıtını arar. Bu soru konum belirleme problemi olarak ele alınır. İkinci soru, robotun gitmek istediği konuma ulaşabilmesi için hedefin belirlenmesi gerekliliğini ortaya koyar. Üçüncü soru ise robotun gidilecek noktaya ilerlerken nasıl bir rota izleyeceğinin sorusudur. Yol planlaması uygulamaları bu soruya yanıt arar[22]. İkinci ve üçüncü sorulara yanıt verilebilmesi için öncelikle birinci soruya yanıt verilmelidir. Gidilmek istenen noktanın belirlenebilmesi için öncelikle robotun kendi konumunu bilmesi gerekir. Gidilmek istenen noktaya ulaşmak için izlenecek olan yolun planlanmasının yapılabilmesi için yine robotun kendi konumunu bilmesi gerekir. Dolayısıyla robot çalışmalarının temelini konum belirleme probleminin çözümü oluşturmaktadır.

Gezgin robot uygulamalarında robot konumunun belirlenmesi başlıca problemlerden biridir. Bu problemin çözümü yapılmadığı sürece robotun hareketi sadece engellerden kaçmak olur. Robot kendi konumunu bildiği sürece, bulunduğu ortamın haritasının çıkarılması, gitmesi gereken noktaya gidilmesi, engellerin şeklinin anlaşılması gibi birçok amacı yerine getirebilir. Bu nedenle gezgin robotun ortamda nerede olduğu veya hareketi sırasında izlediği yol bilgisi çok önemlidir[3].

Robotun konumu, kapalı ve yüzeyin düz olduğu, kısacası ortamın iki boyutlu olarak tanımlandığı durumlarda x,y ve robotun yönelimini belirten θ açısı ile ifade edilir. Bu konumun belirlenmesinde en temel uygulama yöntemi odometrik verilerden yararlanılmasıdır. Odometrik veriler, gezgin robotun motorlarının üzerinde bulunan kodlayıcılarından elde edilmiştir. Ancak odometrik veriler ile yapılan çalışmalar, konum belirleme problemlerinde yalnızca odometrik verilerden

ayarlanmanın hatalı sonuçlar verdiğini göstermiştir [11;20;21]. Odometrik veriler robotun kayması, tekerleklerin patinaj yapması, sistemdeki asimetrilere kaynaklanabilir. Bu hataların azaltılması için ek algılayıcılar kullanılır. Bu ek algılayıcılar genellikle ses ötesi algılayıcıları, jiroskoplar, pusulalar, lazer algılayıcılar, stereo ve mono görüntü algılayıcıları ve dış ortamlar için GPS algılayıcılarıdır. Robot konumunun belirlenmesi için görüntü tabanlı[23], ses ötesi algılayıcı [24] , lazer algılayıcı[25] robot yapıları kullanılmıştır. Bu algılayıcılar ile odometrik verilerin ayarlanması da yapılabilmektedir[4].

Kullanılan ek algılayıcı verileri ile odometrik verilerin bir araya getirilmesi sonucunda daha doğru konum belirlemeleri yapılabilir. Zamanla artan odometrik hataların, kullanılan diğer algılayıcılar ile elde edilen veriler doğrultusunda azaltılması için yaygın olarak Kalman süzgeci algoritmalarından yararlanılır[11,20,21]. Kalman Süzgeci esasında bir veri işleme algoritmasıdır. Kalman süzgeci hatayı en aza indirgeyebilen özyineli bir yapıya sahiptir. Hatanın azaltılabilmesi için sistemdeki bütün girdilerden yararlanılabilir. Hatalı odometrik veriler Kalman Süzgecinden geçirilerek hatalarından büyük ölçüde arındırılabilir. Sistem durumunun en iyi şekilde kestiriminin yapılabildiği Kalman süzgecinde odometrik hatalar, kullanılan ek algılayıcılar sayesinde azaltılabilir. Odometrik verilerdeki hataların azaltılması sonucu robot konumunun daha doğru olarak belirlenmesi sağlanabilir[26]. Kalman süzgecinin bu özellikleri doğrusal sistemlerde olduğu kadar doğrusal olmayan sistemlerde de geçerliliğini sürdürür. Doğrusal olmayan sistemler için Kalman süzgeci genişletilerek, belirli varsayımlar doğrultusunda, sistemin doğrusallaştırılması yapılır.

Bu tez çalışmasında bölgesel konum belirlemede, genişletilmiş Kalman süzgeci kullanılarak haritası belli ortamdaki gezgin robot konumunun belirlenmesi amaçlanmıştır. Kalman süzgeci odometrik hatanın azaltılması için kullanılmıştır. Kalman süzgeci uygulamalarında genellikle sistem ve ölçüm modelindeki gürültü varyansları sabit olarak alınır[20,21]. Bu çalışmada bu varyans değerleri sabit alınmamış her döngüde tekrar hesaplanmıştır. Deneysel çalışmalar için öğretim ve uygulama amaçlı olarak üretilmiş Pioneer 3-DX robotu kullanılmıştır[29]. Robot üzerinde 8 adet ses ötesi algılayıcı(sonar) ve yüksek çözünürlükte kodlayıcılar bulunmaktadır. Kodlayıcılardan elde edilen odometrik verilerde oluşabilecek

hatalar, sonarlardan elde edilen veriler yardımı ile genişletilmiş Kalman süzgeci algoritması kullanılarak indirgenmeye çalışılmıştır.

Tezin, ikinci bölümünde robot konumunun belirlenme yöntemleri ve konum belirlemesinin nasıl yapılabileceği açıklanmıştır. Üçüncü bölüm olan “olasılıksal konum belirleme yaklaşımı” bölümünde Kalman süzgeci algoritmasının ve Kalman süzgeci denklemlerinin temelini oluşturan olasılıksal yaklaşımın konum belirleme problemi ile olan ilişkisi açıklanmıştır. Dördüncü bölümde, rassal bir değişken olan robot konumunun ve oluşturduğu rassal sistemin tanımlanabilmesi için gerekli olasılık ifadeleri verilmiştir. Beşinci bölümde Kalman süzgecinin tanımlanması ve robot sisteminde kullanılacak olan genişletilmiş Kalman süzgecinin matematiksel ifadeleri verilmiştir. Altıncı bölümde robotun ve algılayıcıların oluşturduğu sistem modellenmiş ve kullanılacak genişletilmiş Kalman süzgeci algoritması ile ilişkilendirilmiştir. Benzetim ortamında ve gerçek ortamda yapılan deneysel çalışmaların sonuçları yedinci bölümde gösterilmiştir. Bu bölümde, robot hareketini gerçekleştiren program, akış diyagramı ile açıklanmış, yazılan genişletilmiş Kalman süzgeci programının sonuçları benzetim ve gerçek ortam için verilmiştir. Odometrik veriler ve ek algılayıcı kullanılarak yapılan konum belirleme işlemleri, robotun son konumunda oluşan hatalar yüzdesel olarak verilmiştir. Sonuç ve öneriler bölümünde yapılan deneysel çalışmaların sonuçları değerlendirilmiş ve önerilerde bulunulmuştur.

2. KONUM BELİRLEME

Konum belirleme problemi, robot uygulamalarında önemli bir yer tutar. Robot konumunun belirlenmesi, robotun daha sonraki hareketlerinin belirlenmesi ve uygulanması için gereklidir. Eğer robot bulunduğu ortamda nerede olduğunu belirleyemezse daha sonra ne yapacağına da karar veremez. Birçok yazara göre robotun tam olarak otonom olabilmesi için başlıca gereklilik, robotun konumunu belirleyebilmesi özelliğine sahip olmasıdır. [5]

2.1. Konum Belirleme Problemi

Genel konum belirleme problemi zamanla daha zor problemlerle karşılaşmaktadır. Bu problemlerin artması zamanla yapılan varsayımların azaltılması sonucunda olmuştur.

Konum belirleme problemlerinden başlangıç olarak ele alınanı konum izleme problemidir. Burada robotun başlangıç konumu bilinir. Başlangıç konumu bilinen robotun konumu, robotun hareketi ile beraber izlenir ve son konumu belirlenmeye çalışılır. Bu problemin çözümünü sağlayan yordamlar izleme veya bölgesel yordamlar olarak adlandırılırlar. [6]

Uyanma ya da bütünsel konum belirleme problemi ise konum izleme problemine göre daha zordur. Burada robot başlangıç durumunu bilmez. Robotun konumunu belirlemesi için çabalaması gerekir. Dolayısıyla robot bulunabileceği konum hakkında çok farklı bilgilere sahip olabilir. Bu problemin çözümünde kullanılan yordamlar bütünsel yordamlar olarak adlandırılır. [6]

Kaçırılmış robotun konumunun belirlenmesi problemi ise çok daha zordur. Robotun konumu bilinirken, robota herhangi bir bilgi verilmeden, aniden, robot başka bir konuma aktarılır veya kaçırılırsa robot yeni konumu hakkında yanlış bilgi sahibi olur. Buradaki esas sorun robotun kaçırıldığını veya başka bir konuma aktarıldığının farkına varabilmesidir. Burada kullanılan bütünsel konum belirleme yordamı aynı zamanda uyanma konum belirleme problemi için de kullanılır. Ancak

uyanma konum belirleme problemindeki fark, robota kaçırıldığıının veya başka bir konuma aktarıldığıının söylenmiş olmasıdır.

Belirtilen konum belirleme problemlerinin çözümünde kullanılan birçok yordamda, robotun bulunduğu ortam durağan olarak kabul edilmiştir. Bunun anlamı robot, bulunduğu ortamda tek hareket eden nesnedir. Ancak gerçek hayatta durağan ortam yerine değişken ortamlar bulunmaktadır. Değişken ortamlarda bulunan robotların konum belirleme işlemi sırasında, ortamda bulunan hareket halindeki nesnelere çözüme engel teşkil ederler. Dolayısıyla bütün sistemden alınacak verilerin konum belirleme probleminin çözümünde kullanılacak uygun verilerin olması önemli ve gereklidir.

2.2. Uygun Veriler

Robot kendi konumunu belirlemek için iki farklı veriye ulaşabilmelidir. Bunlardan ilki öncelikli bilgilerdir. Bu bilgiler, robotun kendisinden veya dışarıdaki bir kaynaktan sağlanabilir. Bu bilgiler robotun başlangıç evresinde durumunun belirlenmesinde önemlidir. İkinci olarak da robotun gezinişi sırasında çevreden elde edeceği ortam verileri ve hareket verileri robotun konumunun belirlenmesinde rol oynar.

2.2.1. Elde olan(öncelikli) veriler

Robot başlangıçta, bulunduğu ortam hakkındaki bilgilere sahip ise bu veriler elde olan, öncelikli veriler olarak belirtilir. Öncelikli veriler farklı yapılarda olabilirler.

2.2.1.1. Haritalar

Robot bazen bulunduğu ortamın haritasına sahip olabilir. Bu haritalar çeşitli yapılarda olabilirler. Haritalar tanımlanma şekillerine göre,

- Geometrik haritalar,
- Topolojik haritalar,
- Izgara tabanlı haritalar olarak belirtilebilirler.

Geometrik haritalar santimetre gibi uzaklık birimleri ile belirtilmiş karayolları haritalarına benzer yapıdadırlar.

Topolojik haritalar geometrik haritalara göre daha az bilgiye sahiplerdir. Bu haritalarda hareketin hangi noktadan hangi nokta veya noktalara olacağı belirtilmiştir.

Izgara tabanlı haritalarda ise ortam belirli oranlarda parçalara ayrılır. Böylelikle ayrılan bölge sınırları içerisinde kalan her veri tek bir veriymiş gibi değerlendirilir.

Geometrik haritalar dışında kalan haritalar daha az hassasiyete izin vermelerine karşın hafıza kullanımında daha az yer kapladıkları için uygulamanın kısıtlamalarına göre tercih edilirler.

Daha önce belirtilmiş haritalar başlangıçta robota verilebileceği gibi robotun hareketi sırasında da oluşturulabilir. İşte robotun ortamdaki gezişi sırasında, kendi konumunu belirlerken aynı zamanda ortamın haritasını oluşturma veya öğrenme işlemine EKBH, Eşzamanlı Konum Belirleme ve Haritalama(*Simultaneous Localization and Mapping*) denmektedir.

2.2.1.2. Yer gösterici konumları

Öncelikli verilerden bir tanesi de işaretçilerin konumlarıdır. Eğer işaretçilerin konum bilgileri robota bildirilirse, robot gezinişi esnasında bu işaretçilere bakarak kendi konumunu belirleyebilir. İşaretçilerin konumları sabit olduğundan robot, üzerinde bulunan algılayıcılar yardımı ile kendi konumunu elde ettiği mesafe bilgilerine bakarak konumunu belirleyebilir.

2.2.2. Elde edilen veriler

Robotun konumunu belirlemesinde kullanacağı diğer yapılarıdaki veriler de elde edilen verilerdir. Bu yapıdaki veriler robotun hareketi sırasında elde ettiği verilerdir. Genel olarak bu veriler, sürüş ve algılama verileri olarak ikiye ayrılabilir.

2.3. Görelî Konum Ölçümleri

Konum belirleme probleminin çözümünün başlangıç zamanlarında görelî konum ölçümlerinden yararlanılmıştır. Konum belirlemede yaygın olarak, gözü kapalı tahmin olarak belirtilen görelî konum ölçümlerinden faydalanarak yapılan konum kestirimi, uzun zamandır kullanılan bir konum belirleme yöntemidir.

Genel olarak bu yöntem, hızı ve doğrultusu belli olan nesnelere bu bilgiler yardımıyla, süreye bağılı olarak, daha sonra olacağı konumunun kestirilmesi olarak tanımlanabilir. Bu yöntemde bir sonraki konumun belirlenmesi bir önceki konuma bağılı olduğundan, daha önce oluşan bir hata devamlı olarak sonuca yansıtacaktır. Zaman içerisinde de bu hata yeni oluşabilecek hatalar ile birlikte katlanarak büyüyecektir.

Robotik uygulamalarda gözü kapalı tahmin yönteminin uygulanabilmesi için odometri veya ataletsel gezinim verilerinin kullanılması gerekmektedir.

2.3.1. Odometri

Odometri, zamanla artan bilgilerin birbiri üzerine eklenmesi ile oluşur. Tekerleklerin üzerinde bulunan kodlayıcılardan alınan veriler ile robotun hareket bilgileri elde edilmeye çalışılır. Örneğin her tekerlekteki kodlayıcı, tekerleğin bir dönüşündeki vuruş sayısı bilgisini sağlayabilir. Bu bilginin değerlendirilmesi sonucu robotun hızı, yönü ve aldığı mesafe gibi bilgiler elde edilebilir.

Konum belirleme probleminde odometriden yararlanılmasının nedenlerinden bazıları, uygulamasının ucuz olması, kısa mesafelerde doğruluğunun yüksek olması ve yüksek örnekleme sağlayabilmesidir. Bu olumlu özelliklerine karşın, tekerleklerin patinajından, robotun kaymasından ve yüzey değişikliklerinden kolaylıkla etkilenmesi gibi olumsuz yönleri de vardır. Bu olumsuzluklar sonucu oluşan hatalar konumun belirlenmesi sırasında kalıcı olarak çözüme eklenirler. Daha sonra yapılacak konum belirlemelerin de elde edilen çözüm üzerinden gerçekleşeceği için odometrideki hata, zamanla katlanarak büyür. Odometrik hata ortamdaki kaynaklanabileceği gibi sistemden de

kaynaklanabilir. Örneğin, teker çaplarının aynı olmaması da odometrik hatanın oluşmasına neden olur.

Olumlu ya da olumsuz özelliklerine rağmen odometrik veriler konum belirleme probleminin çözümünde yaygın olarak kullanılmaktadır.

2.3.2. Ataletsel gezinim

Ataletsel gezinim yöntemlerinde jiroskoplar ve ivmeölçerlerden yararlanılır. Bu ölçüm araçları kullanılarak robotun dönüş ve hızlanma bilgileri elde edilmeye çalışılır[7].

Jiroskoplar yönelimdeki ufak değişimleri tespit etmede kullanılırlar. İvmeölçerler ise robot ivmesindeki ufak değişimleri algılamak için kullanılır. İvmeölçerler robotun yüzeyde kaymasından veya robotun hareket yüzeyindeki tümseklerden etkilenirler.

Odometride olduğu gibi, robot hızının ve aldığı mesafenin algılanması için ataletsel gezinim algılayıcılarından elde edilen ölçümler bir araya getirilir. Sistem, odometri bilgilerinin elde edilmesinde olduğu gibi ataletsel gezinim bilgilerini de kendi üzerinden sağlar. Dışarıdan herhangi bir veri gelmediği için, daha önceden belirtilen nedenlerden dolayı kaynaklanacak herhangi bir hata, robot konumunun belirlenmesi sırasında kalıcı olarak ve zamanla giderek artacaktır.

2.4. Mutlak Konum Ölçümleri

Mutlak konum ölçümleri, daha önce elde edilmiş olan robotun konum verilerinden bağımsız olarak gerçekleşir. Dolayısıyla, robot konumunun belirlenmesi için daha önceki verilerin bir araya getirilmesi yerine sadece tek bir ölçüm yeterli olmaktadır. Buradaki hata, göreceli konum ölçümlerinde olduğu gibi, ardı ardına gelen ölçümlerle dahil olmadığından sürekli olarak artmaz.

Mutlak konum ölçümlerini elde etme yöntemleri yer gösterici tabanlı ve harita tabanlı olarak ikiye ayrılabilir.

2.4.1. Yer gösterici tabanlı ölçüm

Yer gösterici tabanlı ölçümler, yer göstericinin yapısına göre farklılıklar gösterir. Yer gösterici yerini belirtmek için kendisi ortama veya robota işaret gönderiyorsa aktif, göndermiyorsa pasif yer gösterici olarak tanımlanabilir.

2.4.1.1. Aktif yer göstericiler

Aktif yer göstericilerden yararlanılarak, robotun mutlak konumunun belirlenmesi yöntemlerinden biri üçgenleme yöntemidir. Bu yöntemde üç ya da daha fazla yer göstericiden elde edilen mesafe ve açı bilgileri ile robotun mutlak konumu belirlenir[8]. Ancak bu yöntemde robota yer göstericilerin konumlarının bilgisi daha önceden verilmelidir. Ayrıca, yer göstericiler genellikle işaretlerini bütün yönlerde ortama veremedikleri için her yerden algılanamayabilirler.

Bir diğer aktif yer göstericili konum belirleme yöntemi de bütünsel konumlandırma sistemi kısacası GPS'dir. Bu sistem, uydudan gönderilen radyo işaretlerinin uçuş süresinin ölçümü prensibine dayanır. Ancak bu yöntemde gönderilen ve alınan işaretlerin uçuş sürelerinin fazla olması ve hava koşullarından etkilenmelerinden dolayı, robotun tam konumunun belirlenmesinde hatalı sonuçlar elde edilebilir.

2.4.1.2. Pasif yer göstericiler

Pasif yer göstericiler ise robot tarafından algılanırlar. Pasif yer göstericiler yapay olarak oluşturulabileceği gibi doğal olarak da ortamda bulunabilirler.

Yapay yer göstericiler robot üzerinde bulunan algılayıcılar tarafından rahatlıkla algılanabilecek yapıda oluşturulurlar. Ancak burada yapay yer göstericiler, robotun bulunduğu her ortama yerleştirilmek zorundadır.

Doğal yer göstericiler her ortamda bulunabilecek kapı, pencere, tavan lambası, duvar köşesi gibi robotun algılayıcıları ile tanımlanabilecek yapılardır. Bu tip pasif yer göstericilerin algılanması, yapay yer göstericilere oranla belirlenmesi işlemsel olarak daha zordur.

2.4.2. Harita tabanlı ölçüm

Mutlak ölçüm yöntemlerinden bir diğeri de harita tabanlı ölçümdür. Bu yaklaşımda ortamın geometrik özelliklerinden faydalanılır. Haritada belirtilen düz çizgiler duvarları, kapıları veya engelleri temsil edebilir. Robotun üzerinde bulunan algılayıcıları sayesinde elde ettiği ölçümlerin haritadaki bilgiler ile karşılaştırılması sonucu robot konumunun belirlenmesi işlemi gerçekleştirilebilir. Ancak burada robotun bulunduğu ortamın haritasını bilmesi gerekmektedir. 2.2.1.1 bölümünde de belirtilmiş olan haritalardan uygun olanına ihtiyaç duyulur. Buradaki uygunluk yapılan uygulamaya ve kullanılacak hafıza oranına göre değişiklik gösterebilir. Ayrıca robotun algılayıcıları ile elde ettiği verilerin harita verileri ile karşılaştırılması işlemsel güç ve algılama hassasiyeti gerektirir.

2.5. Algılamaların Birleştirilmesi

Konum belirleme probleminin çözümü için elde olan veya elde edilecek bütün verilerin bir araya getirilmesi gerekmektedir. Bu verilerin elde edilmesinde kullanılmış olan algılayıcılar birbirlerinden farklı yapıda olabilirler. Bunun sonucu olarak aynı özelliği farklı yapıdaki algılayıcılar farklı olarak algılar veya algılayamaz. Dolayısıyla farklı yapıdaki algılayıcılardan elde edilmiş verilerin, konum belirleme probleminin çözümünde kullanılabilmesi için anlamlarının tek bir yapıya indirgenmeleri gerekmektedir. Kısacası birçok algılayıcıdan elde edilmiş olan veri yumağı tek bir algılayıcıdan elde edilmiş gibi işleme sokulmalıdır. Bu problem çoklu-algılamaların birleştirilmesi problemi olarak araştırılmaktadır [8, 9].

3. OLASILIKSAL KONUM BELİRLEME YAKLAŞIMI

Genel konum belirleme problemi Bayes tabanlı kestirim problemi olarak tanımlanabilir. Bayes teoremi, bir durumun oluşma olasılığının durum hakkında elde edilen ek bilgi veya bilgiler edinilmesi halinde nasıl değişeceğini gösteren bir teoremdir[11].

Konum belirleme probleminde istenilen, gürültülü ölçüm verilerinden yararlanılarak robot konumunun kestirilmesidir. Problem olasılık tabanlı ifade edilecek olursa robot, kendi konumunun belirli bir yerde olma olasılık yoğunluğuna sahiptir. Robotun belirli bir yerde olma olasılığına robotun *inancı* denir. Robot inancını her yeni veri elde ettiğinde yeniler. Böylelikle robotun konum inancı, bulunduğu ortam içerisinde yenilenerek doğru olana yakınsar. Olasılıksal konum belirleme problemi, inanış işlevinin elde edilmeye çalışılması veya kestirilmesidir.

Konum belirleme probleminde Thrun ve çalışma arkadaşları, Bayes kuralı tabanlı kestirim yöntemini kullanmışlardır[6]. Bu yöntem Markov konum belirleme yöntemidir. Markov konum belirleme yönteminde, robotun inanış işlevinin elde edilmesi için birçok algılayıcıdan gelen veriler bir araya getirilir.

3.1. İnanış İşlevi

Robot nerede olduğuna dair bir inancıya sahiptir. Bu inanış robotun bulunduğu ortamda herhangi bir yerde olma olasılık yoğunluğudur. İnanış işlevi şu şekilde gösterilebilir.

$$Bel(x_k) = P(x_k | d_{0,\dots,k}) \quad (3-1)$$

$Bel(x_k)$ olasılığı, k anına kadar verilmiş bilgiler ya da elde edilmiş bütün $d_{0,\dots,k}$ verileri doğrultusunda, robotun x_k konumunda olma olasılık yoğunluğunu belirtir. Bu olasılık dağılımında belirtilen robot konumu, robotun bulunduğu ortamda olabilirliği en yüksek olan konumu göstermektedir. *Eğer robot inancı, belirli bir konumda en yüksek değerine ulaşır ve diğer konumlarda olma olasılığı sıfır olursa, robot tam olarak bulunduğu ortamda nerede olduğunu anlamış olur.*

Robot ortamdaki gezinimi sırasında görel ve mutlak ölçümler elde eder. Bu iki farklı ölçümün bir araya getirilmesi ile birlikte robotun konum olasılığı da değişiklik gösterir. Robotun elde ettiği bu ölçümler, bir önceki ve bir sonraki olmak üzere iki tane inanış işlevi oluşturulmasını sağlar.

Önceki inanış işlevi $Bel^-(x_k)$ ile gösterilir. Bu olasılık yoğunluk işlevinin elde edilmesi sırasında k anına kadar elde edilmiş bütün veriler ve en son alınan görel ölçümler kullanılır.

Sonraki inanış işlevi $Bel^+(x_k)$ ile gösterilir. Sonraki inanış işlevinin elde edilmesinde de en son alınan mutlak ölçümler inanış işlevine dahil edilir.

3.2. Hareket Ve Algılama Modeli

İnanış işlevinin yenilenmesi sırasında kullanılacak olan ölçüm verilerinin de olasılık tabanlı olarak yazılması gerekir. Bu nedenle hareketin olasılıksal modelinin yazılabilmesi için görel ölçümlerin olasılıksal olarak tanımlanması gerekir. Benzer şekilde, algılamanın olasılıksal modelinin yazılabilmesi için mutlak ölçümlerin de olasılıksal tabanlı yazılması gerekir.

3.2.1. Hareket modeli

Robot, bulunduğu ortamda hareket etmesi ile birlikte konumunu değiştirmiş olur. Robotun k anında yaptığı hareketine a_k dersek, robotun ortamdaki hareketini olasılıksal yaklaşımla ifade edebiliriz.

$$P(x_k | x_{k-1}, a_{k-1}) \quad (3-2)$$

(3-2)'de belirtilen olasılık yoğunluğu robotun $k-1$ anında x_{k-1} konumunda bulunurken, a_{k-1} hareketi sonucunda x_k konumunda olma olasılık yoğunluğunu belirtmektedir. Bu olasılık yoğunluk işlevi hareket ya da eylem modeli olarak adlandırılır. Bu olasılık yoğunluk işlevinin elde edilmesi sırasında, olasılık ifadesinden de anlaşılacağı gibi robot ilk konumunu ve yapacağı hareketi

bilmektedir. Robot k anında olacağı x_k konumunun kestirimini, yaptığı görelî ölçümler doğrultusunda yapmaya çalışır.

3.2.2. Algılama Modeli

İkinci model olan algılama modelinin de olasılıksal olarak tanımlanması gerekmektedir. Robotun k anında algılayıcılardan elde ettiği ölçümlere z_k dersek robotun x_k konumundaki olasılık yoğunluk işlevini yazabiliriz.

$$P(z_k | x_k) \quad (3-3)$$

(3-3)'de belirtilen olasılık yoğunluğu duyma ya da algılama modeli olarak adlandırılır. Algılama modeli genellikle zamandan bağımsızdır. Hareket modelindeki geçiş yoğunluk işlevinin elde edilmesine göre algılama modelinin geçiş yoğunluğunun elde edilmesi daha zor olabilir. Bunun nedeni algılayıcılardan gelen bilgilerin çok boyutlu olabilmeleridir. Algılayıcılardan gelen her bilgi için robot konumunun olasılık yoğunluğu değişiklik gösterebilir. Bu olasılık yoğunluğu verilebilir veya ölçümler doğrultusunda elde edilebilir. Genellikle bu olasılık yoğunluğu uygulamalarda verilmektedir[10].

3.3. Olasılıksal Konum Belirleme

Robotun ortam içerisinde konumunu belirleyebilmesi için öncelikle başlangıç inanışına sahip olması gerekmektedir. Robot ortam içerisinde hareket ederek konumunu belirlemeye çalışır. Robotun hareketi ile birlikte robotun ortamdaki konumu değişir. Dolayısıyla (3-2)'deki hareket modeliyle belirtilen geçiş yoğunluğuna bağılı olarak robotun konumu belirlenmeye çalışılır. Robot hareketi sırasında aynı zamanda algılayıcıları sayesinde ortam hakkında bilgi edinir. Bu bilgiler kullanılarak (3-3)'deki algılama modelinde belirtilen olasılık dağılımı da robot konumunun belirlenmesinde kullanır.

3.3.1. Başlangıç inanış işlevi

Robot, ortamdaki hareketine başlamadan önce nerede olabileceğine dair bir başlangıç işlevine sahip olmalıdır. Bu olasılık ifadesini $Be\Gamma(x_0)$ olarak $k=0$ anı için gösterebiliriz. Başlangıç inanışının tanımlanmasına göre robot konum belirleme sorununun tanımlanmasında farklılıklar olur.

Eğer robotun başlangıç konumu biliniyorsa $Be\Gamma(x_0)$ ifadesinin değeri o konum için en büyük olasılık değerini alır. Robotun başlangıç konumu biliniyorsa konum belirleme probleminin amacı *Konum Belirleme* bölümünde belirtilen robot konumunun takip edilmesidir.

Eğer robotun başlangıç konumu bilinmiyorsa o halde $Be\Gamma(x_0)$ ifadesi birbiçimli dağılım gösterir. Yani robot konumunun ortam içerisinde herhangi bir yerde olma olasılık yoğunluğu, diğer konumlardakine eşittir. Buradaki konum belirleme problemi, *Konum Belirleme* bölümünde belirtilen uyanma ya da bütünsel konum belirleme problemi olarak tanımlanır.

Eğer robotun başlangıç konumu yanlış biliniyorsa, yani robotun konumu varsayılan noktada değil de başka bir konumda ise $Be\Gamma(x_0)$ ifadesi hatalı olarak değerlendirmeye alınır. Ancak robot konumunun belirlenmesine çalışılırken robot inanış işlevinin hatalı olduğunu bilmez. Öncelikle bu hatalı inanış işlevinin farkına varıp daha sonra da konumunu belirlemeye çalışmalıdır. Bu durumdaki konum belirleme problemi, *Konum Belirleme* bölümünde de belirtilmiş olan kaçırılmış robotun konumunun belirlenmesi problemidir.

3.3.2. İnanış işlevinin güncellenmesi

Robot başlangıç inanışı doğrultusunda, ortam içerisindeki hareketini gerçekleştirir ve hareketini sürdürürken algılayıcılarından elde ettiği ölçümler ile konumunu belirlemeye çalışır. Kısacası inanış işlevinin, robotun yaptığı hareket ve algılayıcılarından elde ettiği veriler yardımıyla güncellenmesi gerekmektedir.

Robot, algılayıcılardan gelen yeni ölçüm verilerini değerlendirmeden hemen önce, $k-1$ anındaki a_{k-1} hareketi ile k anındaki bir önceki inanış işlevini elde eder.

$$Bel^-(x_k) = P(x_k | z_1, a_1, z_2, a_2, \dots, z_{k-1}, a_{k-1}) \quad (3-4)$$

(3-5)'de belirtildiği gibi, k anında elde ettiği mutlak ölçümler ile birlikte robot, bir sonraki inanış işlevini elde eder.

$$Bel^+(x_k) = P(x_k | z_1, a_1, z_2, a_2, \dots, z_{k-1}, a_{k-1}, z_k) \quad (3-5)$$

Yukarıda elde edilen olasılık yoğunluk işlevinin sonucu bulunduğu takdirde robotun ortamdaki konumu olasılıksal olarak bulunmuş olacaktır.

3.3.2.1. Bir önceki inanış işlevi

Gerçekleştirilen görelî ölçümlerin değerlendirilmesi ile robot bir önceki inanış işlevini elde edebilir. $Bel^-(x_k)$ olasılık yoğunluğunun elde edilmesi sırasında hesaplama kolaylığının sağlanabilmesi için toplam olasılık teoreminden ve Markov varsayımından yararlanılabilir[11]. Bir önceki inanış olasılık yoğunluğunun varsayımlar ve sadeleştirmeler sonucu elde edilmiş son hali (3-6)'da verilmiştir.

$$Bel^-(x_k) = \int_{\Xi} P(x_k | z_{k-1}, a_{k-1}) \times Bel^+(x_{k-1}) dx_{k-1} \quad (3-6)$$

$Bel^-(x_k)$ olasılık yoğunluğu, robotun x_{k-1} konumundayken $k-1$ anındaki a_{k-1} hareketi sonucu, k anında x_k konumunda olma olasılık yoğunluğunu ifade eder.

3.3.2.2. Bir sonraki inanış işlevi

$Bel^+(x_k)$ olasılık yoğunluğu elde edildiği anda, yani robot k anında son hareketini gerçekleştirmiş durumdayken robot, algılayıcılarından elde ettiği mutlak ölçümleri bu olasılık yoğunluğuyla bir araya getirerek bir sonraki inanış işlevini elde edebilir. Bir sonraki olasılık işlevinin hesaplamasında kolaylık sağlamak için Bayes kuralı ve Markov varsayımlarından yararlanılabilir[11]. Bir sonraki inanış olasılık yoğunluğunun varsayımlar ve sadeleştirmeler sonucu elde edilmiş son hali (3-7)'de verilmiştir.

$$Bel^+(x_k) = \frac{P(z_k | x_k) Bel^-(x_k)}{P(z_k | z_1, a_1, \dots, z_{k-1}, a_{k-1})} \quad (3-7)$$

(3-7) de belirtilmiş bir sonraki olasılık yoğunluk işlevindeki payda terimi normalleştirme sabitidir. Bu sabit değer, bütün x_k konumları için hesaplanır.

3.3.3. Konum belirleme formülü

Robotun hareket ve algılama modellerinden alınan bilgiler sayesinde robot konumunun olasılıksal ifadesi elde edilebilir. Robotun (3-7)'de belirtilmiş sonraki inanış işlevindeki $Bel^-(x_k)$ ifadesinin yerine (3-6)'daki ifade yazılırsa, (3-8)'deki robot konumunun olasılıksal ifadesi elde edilir.

$$Bel^+(x_k) = \frac{P(z_k | x_k) \cdot \int_{\Xi} P(x_k | z_{k-1}, a_{k-1}) \times Bel^+(x_{k-1}) \cdot dx_{k-1}}{P(z_k | z_1, a_1, \dots, z_{k-1}, a_{k-1})} \quad (3-8)$$

(3-8)'deki denklemin sonuçlandırılabilmesi için üç tane olasılık dağılımının bilinmesi gerekir.

1. Hareket Modeli: $P(x_k | x_{k-1}, a_{k-1})$, robotun x_{k-1} konumundan a_{k-1} hareketi ile geleceği x_k konumunun olasılık yoğunluğunu ifade eder.
2. Algılama Modeli: $P(z_k | x_k)$, robotun x_k konumundayken gözlenebilecek z_k verilerinin olma olasılık yoğunluğunu ifade eder.
3. Başlangıç İnanışı: $Bel^-(x_0)$, robotun "0" anında x_0 konumunda olma olasılık yoğunluğunu ifade eder.

3.4. Konum Belirleme Yöntemleri

3.4.1. Kesikli inanış işlevi

İnanış işlevinin tanımlanma şekline göre (3-8)'deki inanış işlevi farklılık gösterir. Bölüm (3.4.1)'de gösterilen inanış işlevi sürekli zaman için yazılmıştır.

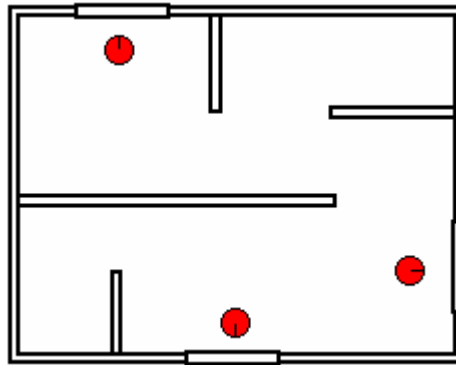
Eğer inanış işlevi kesikli zamanda yazılmak istenirse, (3-8) eşitliğindeki tümlev(\int) işareti yerine toplam(Σ) işareti yazılmalıdır.

Konumun belirleneceği ortam eğer kesikli olarak tanımlanacak ise inanış işlevi ortam üzerinde tam olarak ve saklanarak hesaplanmalıdır. İnanış işlevinin kesikli olarak tanımlanması yaklaşımı Saklı Markov Modelleri olarak adlandırılır. Ortamın kesikli olarak tanımlanmasında birçok yaklaşım yapılmıştır. Bu yaklaşımlardan en önemli birkaç tanesi topolojik çizgeler, ızgaralar ve parçacık süzgeçleridir.

3.4.1.1. Topolojik çizgeler

Topolojik çizgelerde robotun bulunduğu ortam, ayrıık olarak bölünür. Ortamın ayrıklaştırılması sırasında robotun hareketleri kısıtlanmış olarak kabul edilir. Robotun hareket edebileceği doğrultular ve robotun bulunabileceği düğümler harita üzerinde işaretlenir.

Topolojik yapıdaki olasılıksal yaklaşımda, robotun konumu için çok sayıda varsayıma ulaşılabilir. Robotun harita üzerinde birden fazla yerde olma olasılığı vardır. Daha sonra robotun hareketi ve algılayıcıların elde ettiği veriler yardımıyla çok olasılıklı yapıdan tek bir konuma yakınsama yapılarak robotun konumunu belirlenir. Örneğin robotun bulunduğu ortamda birden fazla kapı olsun ve robot algılayıcıları ile bir kapının önünde olduğunu algılamış olsun.



Şekil 3.1. Robotun olası konumları

Şekilden de görüldüğü gibi eğer robot konumu çok olasılıklı yapıda tanımlandıysa, robotun konumu herhangi bir kapının önü olabilir. Daha sonra robot

hareketi sırasında robotun algılayıcıları sayesinde elde edilen veriler ile robotun hangi kapıda olduğunu olasılığı artar ve doğru konuma yakınsama yapılmış olur.

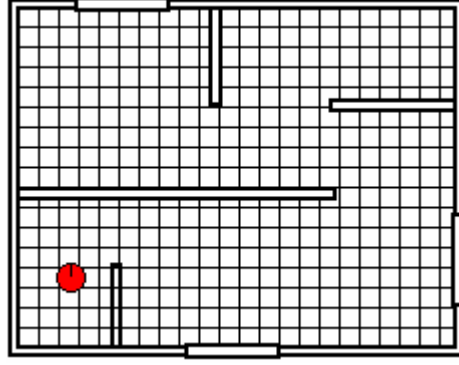
Buradaki konum belirleme tekniği bütünsel konum belirlemede kullanılır. Ancak topolojik yapıdaki haritalama tekniği ile yapılan konum belirlemede haritadaki düğümlerin ve bağlantıların yapısına göre robot konumunun kesin özelliklerinin elde edilmesi zordur. Robot konumunun belirlenmesi haritanın ayrıntılı tanımlanmasına bağlıdır.

Topolojik gösterimde, robotun hareket ve algılama modellerinin tanımlanma karmaşıklığı azaltılmış olur. Ancak topolojik gösterimde robotun konumunun belirlenmesi işlenmemiş verilerden yapılacak çıkarımlara bağlıdır. Yani robot, topolojik haritada yapılmış olan tanımlamaları aldığı verilerden çıkarabildiği sürece inanış işlevini yenileyebilir. Eğer robot aldığı verilerden yeni özellikler çıkaramazsa inanış işlevini güncelleyemez.

3.4.1.2. Izgaralar

Ortam topolojik yapıdan farklı olarak parçalara bölünebilir. Parçalar sabit uzaklıklı ya da değişen uzaklıklı olabilir. Ortamın bu şekilde parçalara bölünmesi ızgara yapılı haritalama olarak adlandırılır[12,13]. Izgara yapılı haritalar topolojik haritalara oranla daha çok konumu belirtir. Bu özelliği sayesinde daha kesin olarak konumun belirlenmesini sağlar.

Topolojik yapıdaki gibi bu yaklaşımda da konum olasılığı ortamdaki her düğüm ya da parça için tanımlanmalıdır. Dolayısıyla ızgara yapılı haritalama yapıldığında daha çok hafızaya ve daha çok hesaplama ihtiyacı duyur.



Şekil 3.2. Ortamın Izgara Harita ile Gösterimi

Bu maliyetler [6]'da belirtilmiş seçici güncelleme ve algılama modelinin önceden hesaplanması yöntemleri ile azaltılabilir. Algılama modelinin önceden hesaplanması yönteminde (3-3) eşitliğinde belirtilmiş olasılık yoğunluk işlevi, belirli ölçüm mesafeleri için önceden hesaplanır. Seçici güncelleme yöntemi ise robotun olma olasılığı yüksek olan ızgara hücreleri ile daha çok ilgilenilmesi temeline dayanır. Robotun olma olasılığı düşük olan ızgara hücrelerinin güncellemesi yapılmayarak hesaplama maliyeti azaltılmış olur. Böylelikle robotun ortamdaki hareketi devam ettiği sürece ilgilendiği ortamın boyutu azalmış olur. Bunun sonucunda da hafıza gereksinimi giderek azalır.

3.4.1.3. Parçacık süzgeçleri

Ortamın kesikli olarak tanımlanmasında başka bir yaklaşım da ortamın parçacıklara ayrılması işlemidir. Bu yaklaşım inanış işlevinin ağırlıklı örneklerle tanımlanması ile oluşur[14]. Her örnek, ortamda robotun olabileceği konumu belirtir. Bu örnekler, önem çarpanı ile tanımlanmışlardır. Her örneğin önem çarpanı en son elde edilen mutlak gözlem ile belirlenir.

Parçacık süzgeci yaklaşımı ile yapılan konum belirleme yöntemi Monte Carlo konum belirleme yöntemidir[14]. Bütünsel ve bölgesel konum belirleme yöntemleri için geliştirilen birçok yöntemle oranda Monte Carlo konum belirleme yöntemi daha etkili ve daha doğru sonuçlar ortaya koymuştur. Bu yöntemde ızgara yapısında karşılaşılan hafıza sorunu ile karşılaşmaz. Ortamın bütünü yerine robotun olma olasılığı yüksek olan yerler örnek olarak alınır. Alınan örnek

sayısının ayarlanması ile robot konumunun kesinliğine karşı hafıza maliyeti arasında ayarlama yapılabilir.

3.4.2. Sürekli inaniş işlevi

Kesikli ortam tanımlamalarında kullanılan yöntemlerden farklı olarak ortamın sürekli tanımlanmasında her konum için ayrı ayrı olasılık hesaplaması yapılmaz. Yalnızca olasılık işlevini belirleyen parametrelerin olasılık yoğunlukları saklanır.

Sürekli inaniş işlevinde yer alan olasılık yoğunluklarının hesaplanma karmaşıklığını ortadan kaldırmak için bu olasılık yoğunluklarının Gauss dağılımlı olduğu varsayılır. Gauss dağılımı, ortalama ve varyans parametreleri ile tanımlanabilir, çan eğrisi yapısındadır. Ortalama parametresi olasılıkların ağırlıklı ortalamasını verir. Varyans ise kesinsizliği belirtir.

3.4.2.1. Kalman süzgeci

Kalman Süzgeci sürekli inaniş işlevinin bir uygulaması olarak ele alınabilir. Kalman süzgecinde hareket ve algılama modelleri Gauss gürültülerine sahip olduğu kabul edilir. Hareket ve algılama modelleri ile elde edilen inaniş işlevleri de Gauss işlevi olarak ortaya çıkar.

4. RASGELE SÜREÇLER VE RASSAL SİSTEMLER

Rassal süreç kavramı, rassal değişken kavramına zamanın da eklenmesine dayanır. Rasgele süreçler ve rassal sistemler, fiziksel sistemin dinamik sürecinde ve ölçümlerindeki belirsizliklerin tanımlanmasını sağlarlar. Dinamik süreçte ve ölçüm sürecinde oluşabilecek belirsizlikler rasgele süreçler ve rassal sistemler ile modellenir[15]. Rassal sistem modeli, dinamik süreç hakkında nelerin bilindiğini ve bu bilinenler ile ilgili belirsizlikleri ortaya koyar.

Dinamik sistemlerin özellikleri, bazı istatistiksel parametreler ile tanımlanabilir. Bu istatistiksel parametreler ortalama, korelasyon ve kovaryans gibi birinci ve ikinci dereceden istatistiksel işlemlerdir. Parametrelerin tanımlanmasında birbirine diklik, durağanlık, Gauss olasılık dağılımına sahip olunması gibi istatistiksel özelliklerden yararlanır.

4.1. Olasılık Ve Rasgele Değişkenler

4.1.1. Olasılık

Bir sistemin çıkışında olabilecek bütün olayların arasından belirli bir tanesinin(A) olma olasılığı (4-1) eşitliğindeki gibi gösterilebilir.

$$p(A) = \frac{A}{\text{bütün olaylar}} \quad (4-1)$$

(4-1) eşitliğinde A , A olayının gerçekleşmesini sağlayan sistem çıktılarının toplamını, *bütün olaylar* sistem çıktılarının toplamını göstermektedir.

A ve B olaylarının herhangi birinin olma olasılığı (4-2)'de gösterilmiştir.

$$p(A \cup B) = p(A) + p(B) \quad (4-2)$$

Eğer A ve B olayları birbirinden bağımsız ise her iki olayın aynı anda olma olasılığı (4-3)'de gösterilmiştir.

$$p(A \cap B) = p(A) \cdot p(B) \quad (4-3)$$

B olayı verildiğinde A olayının olma olasılığı da koşullu olasılık olarak (4-4) eşitliğinde gösterildiği gibi tanımlanır.

$$p(A|B) = \frac{p(A \cap B)}{p(B)} \quad (4-4)$$

İstatistiksel deneyin bütün çıktılarının olasılıklarının toplamı 1'e eşittir. Dolayısıyla sistemin herhangi bir çıkışının \diamond olasılığı 0 ile 1 değerleri arasında değer alır.

$$p(\text{bütün olaylar}) = \sum p(\diamond) = 1 \quad (4-5)$$

$$0 \leq p(\diamond) \leq 1 \quad (4-6)$$

4.1.2. Rasgele değişkenler

Rasgele değişken, bir deneyin olası sonuçlarının bir işlevi olarak tanımlanır. Rasgele değişkenler fiziksel işlemin durumunu sayısal olarak gösterir[15]. Örneğin robot hareketinin gerçekleşmesi için verilen elektriksel gerilim seviyesi gibi sürekli değişim gösteren girdilerle elde edilen sistem çıktıları sürekli olur. Belirli zaman aralıkları ile elde edilen hareket verileri, sayısal olarak sistemin fiziksel durumunu gösterir.

Fiziksel sistemin davranışının araştırılması istatistiksel deneyler ile olabilir. İstatistiksel deneyden elde edilen bütün çıktılar o sistemin örnek uzayını oluşturur. Rasgele değişkenler, örnek uzayı gerçek sayılarla ifade edilmesini sağlayan bir işlev olarak gösterilebilir. Örneğin sürekli $X(t)$ rasgele değişkeni robotun konumunu belirtsin. Herhangi bir zamandaki $X(t)$ robotun beklenen konumunu gösterebilir.

Rasgele değişkenlerin farklı değerler almasının olasılık dağılım fonksiyonu $F_X(x)$ ile gösterilebilir.

$$F_X(x) = p(-\infty, x] \quad (4-7)$$

(4-7) eşitliğinde sürekli rasgele değişken olan X 'in, x olayı da dahil, kaç farklı değer alacağıının olasılık dağılımını göstermektedir.

X rasgele değişkeni sürekli olduğu için (4-7)'deki gösterim yerine daha genel olan olasılık yoğunluk işlevi şu şekilde tanımlanabilir.

$$f_X(x) = \frac{d}{dx} F_X(x) \quad (4-8)$$

4.2. Rasgele Değişkenlerin İstatistiksel Özellikleri

Ortalama değer, N tane X rasgele ayırık değişkeninin ortalaması (4-9) eşitliğinde gösterilmiştir.

$$\bar{X} = \frac{X_1 + X_2 + \dots + X_N}{N} \quad (4-9)$$

Beklenen değer veya istatistiksel ortalama, bütün x_i rasgele değişkenlerinin $p(x_i)$ olasılıkları ile çarpımlarının toplamıdır. Sürekli bir rasgele değişkenin beklenen değeri aşağıda verilmiştir.

$$E[X] = \int_{-\infty}^{\infty} x \cdot f_X(x) dx \quad (4-10)$$

Eğer X rasgele değişkeni N tane farklı x_i değeri alabiliyorsa ve her birinin olasılığı $p(x_i)$ ise, kısacası:

$$f_X(x) = \sum_{i=1}^N p(x_i) \delta(x - x_i) \quad (4-11)$$

(4-11) eşitliğini (4-10) eşitliğinde yerine koyarsak:

$$E[X] = \sum_{i=1}^N x_i \cdot p(x_i) \quad (4-12)$$

Ayrık zamanlı olasılık için X 'in beklenen değeri elde edilir.

Ayrıca rasgele değişkenler ile tanımlanmış işlevlerin de beklenen değerleri elde edilmelidir. X sürekli rasgele değişkeninin bir işlevi $g(X)$ ile gösterilirse, bu işlevin beklenen değerleri (4-13)'de gösterilmiştir.

$$E[g(X)] = \int_{-\infty}^{\infty} g(x) \cdot f_X(x) dx \quad (4-13)$$

X rasgele değişkeni ayrık zamanlı ise,

$$E[g(X)] = \sum_{i=1}^N g(x_i) \cdot p(x_i) \quad (4-14)$$

şeklinde gösterilir.

Momentler, X rasgele değişkeninin tanımladığı işlevin üstel değeri ile adlandırılırlar. $g(X)$ işlevi X^n olarak tanımlanırsa m_n değeri X 'in n 'inci momentidir.

$$m_n = E[X^n] = \int_{-\infty}^{\infty} x^n \cdot f_X(x) dx \quad (4-15)$$

X rasgele değişkeninin 0. momenti 1'e eşittir. $m_0=1$

X rasgele değişkeninin 1. momenti ortalama değerine eşittir. $m_1 = \bar{X}$

Varyans, $g(X)$ işlevi X^2 olarak tanımlanması ile elde edilen X 'in ikinci momentidir.

$$\sigma^2 = E[X^2] = E[(X - \bar{X})^2] = \int_{-\infty}^{\infty} x^2 \cdot f_X(x) dx \quad (4-16)$$

(4-16) eşitliğinde σ^2 ile gösterilen varyans eşitliğini çözümlediğimizde:

$$\begin{aligned} \sigma^2 &= E[X^2 - 2X\bar{X} + \bar{X}] = E[X^2] - 2\bar{X}E[X] + \bar{X}^2 \\ &= E[X^2] - \bar{X}^2 = m_2 - m_1^2 \end{aligned} \quad (4-17)$$

(4-17) eşitliğinde de görüldüğü gibi varyans 1. ve 2. momentler ile hesaplanabilir. Varyans rasgele işaretlerin tanımlanmasında önemli bir özelliktir.

Varyansı bilinen rasgele değişkenin hangi değer etrafında salındığı ve bu salınımın, başka bir ifadeyle gürültünün, ne kadar olduğu hakkında fikir sahibi olunabilir.

Standart sapma, varyansın karekökünün alınması ile elde edilir.

$$\sigma = \sqrt{\sigma^2} = \sqrt{m_2 - m_1^2} \quad (4-18)$$

4.2.1. Korelasyon ve kovaryans

$X(t)$ rassal sürecinin varyansı değişimin değerini verir. Ancak $X(t)$ rassal sürecinin hangi oranda değiştiğinin bulunması için $X(t)$ 'nin **korelasyon** işlevi hesaplanmalıdır. Başka bir ifade ile işarette bulunan gürültünün değişim oranı korelasyon işlevinin hesaplanması ile bulunabilir.

Rassal süreç $X(t)$ 'nin otokorelasyon işlevi, t_1 ve t_2 zamanlarında tanımlı $X_1 = X(t_1)$ ve $X_2 = X(t_2)$ 'nin korelasyonudur(R_{XX}).

$$\begin{aligned} R_{XX}(t_1, t_2) &= E[X(t_1)X(t_2)] \\ \left. \begin{array}{l} t_1 = t \\ t_2 = t_1 + \tau \end{array} \right\} R_{XX}(\tau) &= E[X(t)X(t + \tau)], \quad \tau \in R \end{aligned} \quad (4-19)$$

$X(t)$ ve $Y(t)$ gibi iki rassal sürecin çapraz korelasyonu, çarpımlarının beklenti değeridir.

$$\begin{aligned} R_{XY}(t_1, t_2) &= E[X(t_1)Y(t_2)] \\ \left. \begin{array}{l} t_1 = t \\ t_2 = t_1 + \tau \end{array} \right\} R_{XY}(t, t + \tau) &= E[X(t)Y(t + \tau)] \end{aligned} \quad (4-20)$$

Eğer $X(t)$ ve $Y(t)$ rassal süreçleri istatistiksel olarak birbirlerinden bağımsızlarsa (4-20) eşitliğinde belirtilmiş olan korelasyon işlevi aşağıdaki gibi yazılabilir.

$$R_{XY}(t_1, t_2) = E[X(t_1)Y(t_2)] = E[X(t_1)]E[Y(t_2)] \quad (4-21)$$

Eğer $R_{XY}(t_1, t_2) = 0$ eşitliği yazılabiliyorsa $X(t)$ ve $Y(t)$ rassal süreçlerinin değerleri diklik özelliğine sahiptir.

Bir rasgele değişkenin başka bir rassal değişkene bağlantı derecesi **kovaryans** ile belirtilir. Kovaryans C ile gösterilir. $X(t)$ ve $Y(t)$ rassal süreçlerinin çapraz kovaryansı (4-22)'de gösterilmiştir.

$$C_{XY}(t, t + \tau) = E[\{X(t) - E[X(t)]\}E\{Y(t + \tau) - E[Y(t + \tau)]\}] \quad (4-22)$$

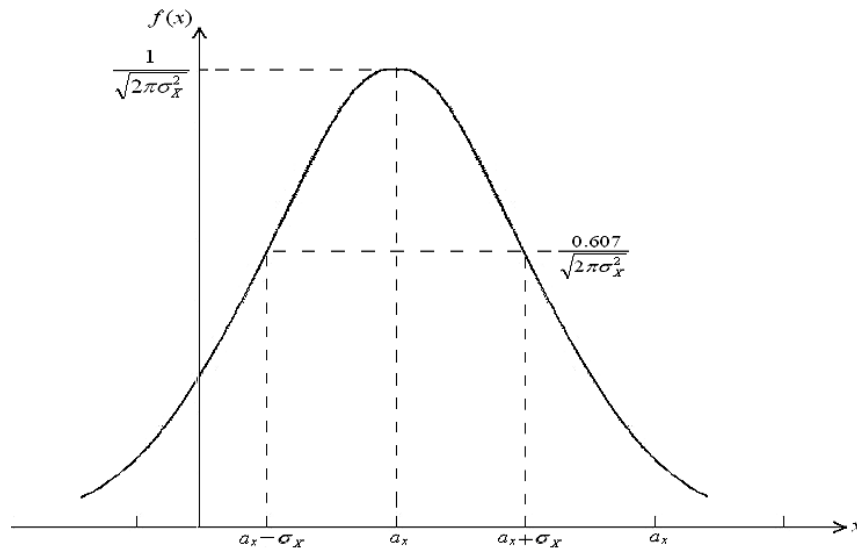
veya,

$$C_{XY}(t, t + \tau) = R_{XY}(t, t + \tau) - E[X(t)].E[Y(t + \tau)] \quad (4-23)$$

4.2.2. Gauss olasılık yoğunluğu

Rassal sistem modellerinde rasgele değişkenlerin dağılımı genellikle Gauss olasılık dağılımı kullanılarak gösterilir. Gauss olasılık yoğunluğu çok önemli ve kullanışlı bir yoğunluk işlevidir. Bu olasılık yoğunluğu özellikle deneysel ve gerçek ortamlarda ilgilenilen miktarların birçok rasgele değişken tarafından etkilendiği durumlar için durumun net olarak belirlenmesi amacıyla kullanılır.[19]

$$f(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} e^{-(x-m_x)^2/2\sigma_x^2} \quad (4-24)$$



Şekil 4.1 Gauss yoğunluk işlevi

Burada $f(x)$ işlevindeki a_x ortalamayı σ_x terimi de işlevin standart sapmasını belirtir. Görüldüğü gibi gauss olasılık dağılımı birinci ve ikinci dereceden momentler ile ifade edilebilir. Bir boyutlu bir rassal değişkenin olasılık yoğunluk işlevi (4-25)'deki gibi gösterilebilir.

$$x \sim N(a_x, \sigma_x^2) \quad (4-25)$$

İki rassal değişken de Gauss olasılık dağılımına sahipse aralarındaki ilişki bu iki rassal değişkenin kovaryansları ile tanımlanabilir ve

$$x \sim N(m, C_x) \quad (4-26)$$

gauss ifadesi ile gösterilebilir.

5. KALMAN SÜZGEÇİ

Elektronik devrelerde süzgeç terimi, istenmeyen gürültülerin, çıkışta istenilen işaretten ya da sisteme alınan işaretlerden ayrıştırılması anlamında kullanılmaktadır. Kalman Süzgeci, gürültülü verilerden, sistem çıkışında doğru bilgi elde edilmesinde kullanılır. Bu algoritmanın süzgeç olarak nitelendirilmesinin nedeni, çıkışta elde edilen verilerin giriş işaretlerindeki gürültülerden ayrıştırılarak doğru bir şekilde elde edilmesini sağladığı içindir.

Kalman Süzgeci, 1960 yılında durum uzay modeli üzerinden yinelemeli olarak durumun elde edilebilmesi için Kalman tarafından geliştirilmeye başlandı. Kalman Süzgeci esasında bir veri işleme algoritmasıdır. Başka bir söylemle “Kalman Süzgeci sahip olunan tüm ölçüm verilerini, sistem ve ölçüm aletleri hakkındaki ön bilgileri birleştirerek istenen değişkenlerin tahmini değerlerini, hataları istatistiksel olarak en aza indirecek şekilde hesaplayan sayısal bir süzgeçtir.” [16]

Kalman Süzgeci beyaz gürültü ile sarsıma uğratılmış doğrusal dinamik bir sistemin durumunu yine beyaz gürültü ile sarsıma uğratılan ve sistemin durumu ile doğrusal olarak bağlantılı olan ölçüm değerlerinden yararlanarak kestiren, en iyi, yinelemeli doğrusal bir süzgeçtir. [15]

Kalman Süzgeci özyineli, en iyi hata indirgeme algoritmasıdır. Kalman Süzgeci, gürültülü algılayıcı ölçümlerinden yararlanılarak rassal durum kestirimleri yapabilmek için en çok kullanılan ve bilinen bir araçtır. Rasgele süreçlerin davranışları daha önceden tanımlanamaz. Kalman Süzgeci kullanılarak sistemin bir sonraki durumu, tam anlamıyla olmasa da, bir önceki durumdan faydalanılarak parçalı bir şekilde tanımlanabilir. Belirli varsayımlar yapıldığında Kalman Süzgeci en iyi, özyineli veri işleyebilen bir algoritmadır.

Kalman Süzgeci *en iyidir*: Kalman Süzgecinin en iyi başarımı göstermesinin nedenlerinden bir tanesi, sistemden anlık elde edebildiği bütün verileri kullanabiliyor olmasıdır. Kalman Süzgeci doğru sayılabilecek bir durum kestirimi yapabilmek için elde ettiği bütün verileri kullanır. Bunu gerçekleştirebilmek için sistem dinamikleri, sistem gürültüsünün istatistiksel tanımlamaları, ölçüm

gürültüsü, sistem dinamiklerindeki kararsızlıklar, başlangıç durumu hakkındaki en ufak bilgiler gibi bilgilerin bir araya getirilmesi gerekmektedir.

Kalman Süzgeci *özyinelidir*: Bu özelliği bütün verilerin depolanması zorunluluğunu ortadan kaldırır. Dolayısıyla her gelen bilgi için tekrar tekrar sürecin işletilmesi gerekmez. Sürekli elde edilen kazanç sayesinde sistemin yalnızca bir önceki durumunun bilinmesi, sistemin bir sonraki durumunun kestirilmesine yeterli olmaktadır.

Kalman Süzgeci kullanımında yalnızca sistemin girdileri ve çıktıları hakkında edinilen bilgi, sistemin bir sonraki durumunun kestiriminde yeterlidir. İlgilenilen değişkenler doğrudan ölçülemeseler bile kullanılabilir veri yapısına dönüştürülmek zorundadırlar. Süzgeç, gürültülü ortamdan elde ettiği verileri kullanarak ilgilenilen değişkenlerin en iyi kestirimini yapmaya çalışır.

5.1. Kalman Süzgecinin Kullanım Alanları

Kalman Süzgeci birçok uygulamada kullanılır. Kalman Süzgeci özellikle otonom veya gezinim yapan sistemler için kullanılır[17]. Robot gezinişinden uçak veya gemilerin radar aracılığı ile yerlerinin tespitine; üretimden demografik modellemeye kadar Kalman süzgeci birçok alanda kullanılır. Genellikle dinamik sistemlerin durum denetimi ve öngörüsü için kullanılırlar.

Kalman Süzgeci dinamik sistemlerin denetimi için kullanıldığında durumun kestirimi amaçlı kullanılır. Sistemin denetimi sırasında sistem hakkında alınabilecek en fazla bilginin elde edilmesi durum kestiriminin daha doğru yapılmasını sağlar. Sistem hakkında her bilginin elde edilmesi bazen mümkün değildir. Kalman Süzgeci, elde edilemeyen verileri diğer elde ettiği veriler aracılığıyla kestirimini yaparak durum analizini gerçekleştirebilir.

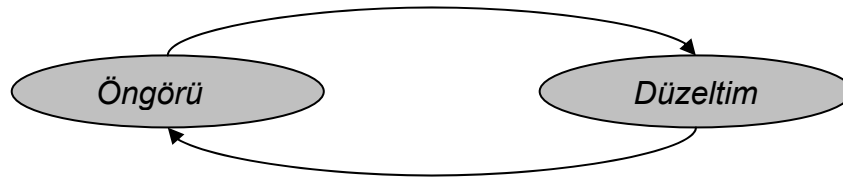
Kalman Süzgeci kestirim amaçlı kullanıldığında ise daha çok istatistiksel verilere dayanarak sonraki durumun kestiriminin gerçekleştirilmesi için kullanılır. Bir nehrin taşması durumundaki akış hızının öngörüsünün yapılması örnek olarak verilebilir.

Kalman Süzgecinin temeli, doğrusal dinamik sistemlerin zaman tabanında ayrıklaştırılmasıdır. Ancak bu özyineli algoritma aynı zamanda doğrusal olmayan sistemler için de kullanılmaktadır. Doğrusal olmayan sistemlerin kullanımında öncelikle, doğrusallaştırma işlemi yapılarak algoritma uygulanabilir.

Robotik uygulamalarında Kalman Süzgeci, robotun algılayıcıları ile algıladığı gürültülü verileri kullanarak robot konumunun kestirilmesinde kullanılabilir. Bu durumda süzgeç robot sistemini ve ölçümlerini etkileyecek gürültüler ve kararsızlıkları dikkate almak zorundadır.

5.2. Genel Kavram

Kalman Süzgeci öngörü-düzeltilim yapısının özyineli olarak işletilmesi ile durum kestirimini gerçekleştirir. Bu özyineli yapı, inanış işlevinin önceden öngörülüp daha sonra sistemden algılanan verilerle, öngörülen durumun düzeltilmesi işlemlerinin tekrarlanması ile gerçekleşir(Şekil 5.1).



Şekil 5.1 Kalman Süzgeci Döngüsü

5.2.1. Durum kestirimi

Kalman Süzgecinin kullanım amacı, durum kestirimini yapılarak doğru veya olması gereken durumun elde edilmesidir. Eğer durum, zamanla değişen bir yapıdaysa Kalman Süzgecinin doğru kestirim yapması zorlaşır. Doğru durum kestirimini zorlaştıran başka bir etken de sistem değişkenlerinin doğrudan gözlemlenemiyor olmasıdır. Bu durumda Kalman Süzgeci kullanılarak algılayıcılardan gelen veriler ile sistem değişkenlerinin gözlenmesi sağlanabilir. Algılayıcılardan gelen verilerin gürültülü olacağı da göz önünde bulundurulduğunda durum kestirimini yapılması daha zor bir probleme dönüşür.

Ancak Kalman Süzgeci bu olumsuz etkilere karşın durum kestirimini en iyi şekilde gerçekleştirir.

5.2.2. İnanış işlevi

Kalman Süzgeci, bölüm 3.1de belirtilen inanış işlevinin, başka bir yaklaşımla x_k durumunun kestirimini gerçekleştirir. x_k durumunun alınan bütün z_1, \dots, z_k ölçümlerine bağılı olarak elde edilen koşullu olasılık yoğunluğu (5-1) eşitliğinde gösterilmiştir.

$$Bel(x_k) = P(x_k | z_{0\dots k}) \quad (5-1)$$

Bölüm 3'de belirtildiği gibi inanış işlevini bir önceki ve bir sonraki inanış işlevleri olarak iki ayrı eşitlikte yazabiliriz.

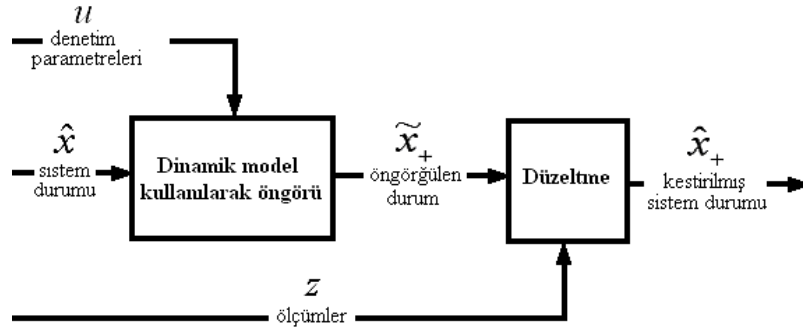
$$Bel^-(x_k) = \int_{\Xi} P(x_k | x_{k-1}) \times Bel^+(x_{k-1}) dx_{k-1} \quad (5-2)$$

$$Bel^+(x_k) = \frac{P(z_k | x_k) \cdot Bel^-(x_k)}{P(z_k | z_1, \dots, z_{k-1})} \quad (5-3)$$

(5-2) ve (5-3) eşitliklerinden de görüleceği gibi Kalman Süzgeci'nin inanış koşullu olasılık yoğunluğunu elde edebilmesi için sistem modeline $P(x_k|x_{k-1})$ ve ölçüm modeline $P(z_k|x_k)$ ihtiyacı vardır.

5.2.3. Öngörü ve düzeltim

Kalman Süzgecinin inanış işlevini hesaplayabilmesi için öncelikle bir önceki inanış işlevini hesaplaması, daha sonra da bir sonraki inanış işlevini hesaplaması gerekmektedir.



Şekil 5.2 Standart model-tabanlı Kalman Süzgeci tasarım şekli.

Şekil 5.2'deki dinamik model, bir sonraki durumun \tilde{x}_+ öngörülmesinde kullanılmıştır. Ölçümler bu öngörüü düzeltmede kullanılmıştır.

5.2.3.1. Öngörü

Bir önceki inanış işlevinin hesaplanması, inanış işlevinin öngörüsü olarak belirtilebilir. Başka bir ifade ile sistem durumunun bir zaman sonraki durumunun öngörüsü denebilir. Bu öngörünün yapılması sırasında bir zaman sonraki ölçümler değerlendirmeye alınmadan sistemin bir zaman sonraki durumu öngörülür. Burada yalnızca sistemin hareket modelinden $P(x_k | x_{k-1}, a_{k-1})$ yararlanır.

5.2.3.2. Düzeltilme

Sistemden kaynaklanan gürültülerden dolayı, durum öngörüsünde hatalar oluşabilir. Kalman Süzgecinin hesapladığı bir önceki inanış işlevinden hemen sonra alınan yeni ölçümler sistemin gerçek durumu hakkındaki verileri sağlar. Bu ölçümler öngörüde yapılmış veya olabilecek hataların düzeltilmesinde kullanılır. Kalman Süzgeci yapacağı düzeltim işlemi için sistemin ölçüm modelini $P(z_k | x_k)$ kullanır.

Belirtilmiş olan öngörü ve düzeltim süreci Kalman Süzgecinin temelini oluşturmaktadır. Kalman Süzgeci kullanılarak sistemin bir sonraki durumunun kestirimi gerçekleştirilir. İlk olarak hareket modeli kullanılarak sistem durumunun öngörüsü gerçekleştirilir. Daha sonra da sistemin ölçüm modeli kullanılarak

düzeltilme işlemi gerçekleştirilerek sistemin bir zaman sonraki kestirimi yapılmış olur.

5.3. Kabuller

5.3.1. Sistem

Kalman Süzgeci doğrusal dinamik sistemler için geliştirilmiş bir algoritmadır. Ancak doğrusal olmayan sistemler için de sistem doğrusallaştırılarak kullanılabilir. Doğrusal dinamik sistemler, sistem modeli ve ölçüm modeli ile tanımlanabilirler.

5.3.2. Sistem modeli

Sistem modeli, sistemin zamanla nasıl değiştiğini belirtir. Kalman Süzgeci sistem modeline dayanarak sistemin bir zaman sonraki durumunun öngörüsünü gerçekleştirir.

$$x_k = A.x_{k-1} + B.u_{k-1} + w_{k-1} \quad (5-4)$$

Çizelge 5.1: Sistem modeli parametreleri

<i>Parametreler</i>	<i>Açıklamalar</i>
x_k	<i>k zamanındaki durum</i>
x_{k-1}	<i>k-1 zamanındaki durum</i>
u_{k-1}	<i>denetim girdileri</i>
w_{k-1}	<i>sistem gürültüsü</i>
A	<i>Durumun zamana göre dönüşüm işlevi</i>
B	<i>Denetim girdileri ile sistem arası dönüşüm işlevi</i>

(5-4) eşitliğinde doğrusal dinamik bir sistemin durum uzayı gösterimi ile modellenmesi yapılmıştır.

5.3.3. Ölçüm modeli

Ölçüm modeli sistem durumu ile ölçümler arasındaki ilişkiyi belirtir. Kalman Süzgeci ölçüm modelini, daha önceden öngördüğü sistem durumunun düzeltilmesi için kullanır.

$$z_k = H.x_k + v_k \quad (5-5)$$

Çizelge 5.2: Ölçüm modeli parametreleri

<i>Parametreler</i>	<i>Açıklamalar</i>
z_k	<i>k zamanındaki ölçüm</i>
x_k	<i>k zamanındaki durum</i>
v_k	<i>ölçüm gürültüsü</i>
H	<i>sistem ile ölçüm arasındaki dönüşüm işlevi</i>

Sistem modelini belirten (5-4) eşitliğinden de görüleceği gibi sistemin bir sonraki (k anındaki) durumu, daha önceki bütün durumlarına bağlı değildir. Benzer şekilde, ölçüm modelini belirten (5-5) eşitliğindeki bir sonraki ölçüm değeri, daha önceki durum ya da ölçümlere bağlı değildir. Dolayısıyla, işlemsel süreçte sonraki durum ve ölçümlerin hesaplanması sırasında, daha önce elde edilmiş bütün durum ve ölçüm verilerinin kayıtlı olması gerekmemektedir.

5.3.4. Gürültü

(5-4) eşitliğinde belirtilen sistem modelinde, w sistem kaynaklı gürültüyü yani süreç gürültüsünü belirtmektedir. (5-5) eşitliğinde belirtilen ölçüm modelinde, v ölçüm gürültüsünü belirtmektedir. Kalman Süzgecinin kullanımında sistem gürültüsünün ve ölçüm gürültüsünün rasgele değerlerden oluştuğu varsayılmaktadır. Bu rasgele değişkenlerin, kısacası gürültülerin, birbirinden bağımsız, beyaz, sıfır-ortalımalı Gauss olasılık dağılımlı oldukları varsayılmaktadır.

5.3.5. Bağımsızlık varsayımı

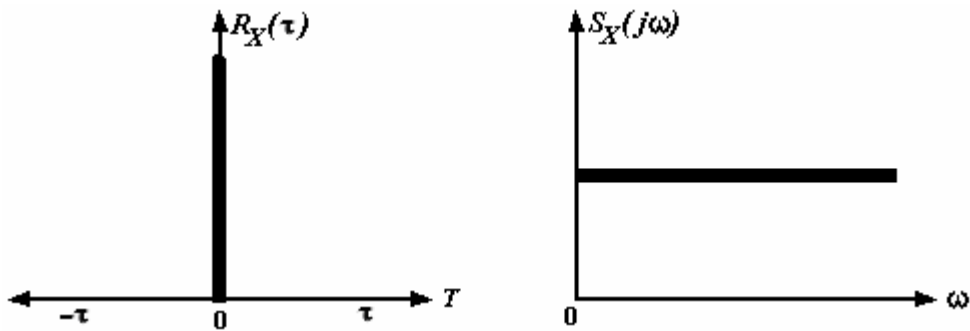
Gürültülerin bağımsız olmasının kabulü sistemin öngörüsünü kolaylaştırır. Gürültülerin birbirinden bağımsız olması kabulü genel olarak gerçek bir varsayımdır. Örneğin, ölçüm gürültüsünün kaynağı olarak sonar ya da lazer algılayıcılarının tipik özelliklerinden kaynaklanacak gürültüler robotun odometrik algılayıcı kaynaklı gürültülerinden bağımsızdır.

$$E[xy] = E[x]E[y] \quad (5-6)$$

(5-6) eşitliğinin sağlanabilmesi için x ve y rasgele değişkenlerinin birbirinden bağımsız olması gerekmektedir.

5.3.6. Beyaz gürültü varsayımı

Beyaz gürültü varsayımı, matematiksel olarak Kalman Süzgecinde işlem kolaylığı sağlamaktadır. Beyaz gürültü rasgele bir işarettir. Bu rasgele işaretin gücü, tüm frekans bölgesinde eşit oranda dağılmıştır. Yani beyaz gürültü işaretinin güç spektral yoğunluğu, her frekansta sabit bir değere sahiptir. Başka bir ifade ile işaretin herhangi bir zamandaki örneği ile başka bir zamandaki örneği arasında herhangi bir ilinti yoktur.



Şekil 5.3 Beyaz gürültü gösterimi: zaman tabanında (solda), sıklık tabanında (sağda)

Şekil 5-3'de zaman tabanında gösterildiği gibi, beyaz gürültü özelliğine sahip işaretin $\tau = 0$ anında otokorelasyon işlevi *dirac* $\delta(\tau)$ işlevine eşittir. $\tau \neq 0$ durumlarında otokorelasyon işlevinin sonucu 0'dır. Dolayısıyla sistemin gürültüsü başka bir zamandaki gürültü ile ilintili değildir.

5.3.7. Gauss dağılımı varsayımı

Gauss olasılık yaklaşımı gürültünün genliği ile ilişkilidir. Sistemin içerdiği gürültünün olasılık yoğunluğunun çan eğrisi şeklinde dağılmış olduğu kabul edilir.

Sistem ve ölçüm verileri birçok gürültü kaynağından etkilenirler. Gürültü kaynaklarının her birinin sisteme sağladığı gürültü miktarı bir araya getirilerek gürültüler toplandığında oluşan gürültünün Gauss dağılımlı olduğu varsayılabilir.

Matematiksel olarak ifade edilirse, pek çok birbirinden bağımsız değişken rasgele bir araya getirildiğinde ortaya çıkan toplam etki Gauss olasılık yoğunluğuyla ifade edilebilir. Ayrıca Gauss olasılık yoğunluğu matematiksel olarak işlem kolaylığı da sağlamaktadır. Özellikle aritmetik ortalama ve varyans gibi birinci ve ikinci dereceden istatistiksel işlemler ile ifade edilebilen gürültüler için Gauss olasılık yoğunluğu tanımlaması kolaylıkla yapılabilir.

Yukarıda belirtilmiş olan varsayımlar dikkate alınarak, sıfır ortalamalı ve Gauss dağılımına sahip gürültüler kolaylıkla tanımlanabilir.

$$Q_k = E[(w_k)(w_k)^T] \quad (5-7)$$

Eşitliğinde Q_k k anındaki sürecin gürültü kovaryansını belirtmektedir.

$$R_k = E[(v_k)(v_k)^T] \quad (5-8)$$

Eşitliğinde R_k k anındaki ölçüm gürültü kovaryansını belirtmektedir.

(5-7) ve (5-8) eşitliklerinde belirtilen gürültü kovaryansları ele alındığında süreç ve ölçüm gürültüleri şu şekilde tanımlanabilir.

$$w_k \sim N(\mu_w, Q_k) \quad (5-9)$$

$$v_k \sim N(\mu_v, R_k) \quad (5-10)$$

5.4. Doğrusal Kalman Süzgeci

Herhangi bir $k-1$ anında x rasgele değişkeninin beklenen değeri $E[x_{k-1}] = \hat{x}_{k-1}$ eşitliği ile gösterilir ve kovaryans matrisi P_k ile gösterilir. k anındaki öngörü durumu ise \hat{x}_k^- ve kovaryansı P_k^- ile gösterilir.

Genel olarak Kalman Süzgeci'nde sistem dinamikleri, bir önceki durumun doğrusal dönüşümü olarak tanımlanır ve denetim girdileri ile sistem dinamikleri sıfır ortalamalı Gauss dağılımlı olarak tanımlanır.

$$\hat{x}_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1} \quad (5-11)$$

Durumun bazı gürültülü ölçümler ile belirlenmeye çalışıldığı dikkate alındığında sistemin ölçümleri de modellenmelidir. Standart Kalman Süzgeçlerinde bu model sistemin o anki durumu ve algılayıcı kaynaklı gürültüleri ile tanımlanır.

$$z_k = H_k x_k + v_k \quad (5-12)$$

Yukarıda belirtilen her iki modeldeki gürültüler de sıfır ortalamalı Gauss olasılıklı olduğu kabul edilir.

$$w_k \sim N(0, Q) \quad , \quad v_k \sim N(0, R) \quad (5-13)$$

(5-13)'de belirtilmiş olan Q ve R matrisleri gürültülerin kovaryanslarını belirtmektedir.

$$Q = E[w_k w_k^T] \quad , \quad R = E[v_k v_k^T] \quad (5-14)$$

Süreç ve ölçüm gürültüleri herhangi bir durum ya da ölçüm ile ilintili(correlated) olmadıkları için aşağıdaki eşitlikler yazılabilir.

$$E[x_i v_j^T] = 0, \quad \forall i, j \quad (5-15)$$

$$E[x_i w_j^T] = 0, \quad i \leq j \quad (5-16)$$

$$E[z_i v_j^T] = 0, \quad i \leq j - 1 \quad (5-17)$$

$$E[z_i w_j^T] = 0, \quad i \leq j \quad (5-18)$$

$$E[v_i w_j^T] = 0, \quad \forall i, j \quad (5-19)$$

$$E[v_i v_j^T] = 0, \quad i \neq j, \quad E[v_i v_j^T] = R, \quad i = j \quad (5-20)$$

$$E[w_i w_j^T] = 0, \quad i \neq j, \quad E[w_i w_j^T] = Q, \quad i = j \quad (5-21)$$

Kalman Süzgeci'nin esas işlevi, bir sonraki durumun kestirimini daha önceki duruma, denetim girdilerine ve k zamanındaki ölçümlere dayanarak yapmaya çalışmaktır. Başka bir ifade ile $p(x_k | x_{k-1}, u_{k-1}, z_k)$ olasılık yoğunluğunun hesaplanmasıdır. Buradaki amaç durumun hata kovaryansını azaltmaktır.

5.4.1. Bir önceki inanış işlevi

Bu bölümde *olasılıksal konum belirleme yaklaşımı* bölümünde ayrıntılı olarak ele alınan, bir önceki inanış işlevinin Kalman Süzgeci'ndeki ifadesi incelenecektir. Kalman Süzgecinde bir önceki inanış işlevi, gauss ifadesi ile $Bel^-(x_k) = N(\hat{x}_k^-, P_k^-)$ şeklinde gösterilebilir. Gauss ifadesindeki ilk terim ortalamayı, ikinci terim de kovaryansı ifade eder. Dolayısıyla burada, \hat{x}_k^- ortalamayı başka bir ifade ile bir önceki durumun beklenen değerini, P_k^- ise kovaryansı başka bir ifade ile bir önceki durum kestirimindeki hata kovaryansını belirtir.

(5-11) eşitliğinde belirtilen dinamik model ele alındığında k anındaki bir önceki durumun beklenen değeri aşağıda belirtilmiştir.

$$\hat{x}_k^- = E[x_k^-] \quad (5-22)$$

$$\begin{aligned} \hat{x}_k^- &= E[Ax_{k-1} + Bu_{k-1} + w_{k-1}] \\ &= E[Ax_{k-1}] + E[Bu_{k-1}] + E[w_{k-1}], \quad E[w_{k-1}] = 0 \\ &= E[Ax_{k-1}] + E[Bu_{k-1}] \\ &= AE[x_{k-1}] + BE[u_{k-1}], \quad u_{k-1} \notin r.d. \\ &= AE[x_{k-1}] + Bu_{k-1} \end{aligned} \quad (5-23)$$

$$\hat{x}_{k-1}^+ = E[x_{k-1}] \quad (5-24)$$

(5-24) eşitliği (5-23) eşitliğinde yerine konulduğunda:

$$\hat{x}_k^- = A\hat{x}_{k-1}^+ + Bu_{k-1} \quad (5-25)$$

(5-25) eşitliğinin hesaplanabilmesi için kovaryansın k anındaki beklenen değeri kestirilmelidir. Bu işlem için durum kestirimi \hat{x}_{k-1}^+ verildiğinde hata aşağıdaki gibi tanımlanabilir.

$$\begin{aligned}
e_k^- &= x_k - \hat{x}_k^- \\
&= (Ax_{k-1} + Bu_{k-1} + w_{k-1}) - (A\hat{x}_{k-1}^+ + B\hat{u}_{k-1}) \\
&= A(x_{k-1} - \hat{x}_{k-1}^+) + B(u_{k-1} - \hat{u}_{k-1}) + w_{k-1} \\
&= Ae_{k-1}^+ + Be_{u,k-1}^+ + w_{k-1}, \quad \text{denetim girdi hatası } e_{u,k-1}^+ = 0 \\
&= Ae_{k-1}^+ + w_{k-1}
\end{aligned} \tag{5-26}$$

Bir önceki durum kestirimindeki hata kovaryansı (5-26) eşitliğinde elde edilmiş olan hata tanımlamasının yardımıyla şu şekilde ifade edilebilir.

$$\begin{aligned}
P_k^- &= E[(e_k^-)(e_k^-)^T] \\
&= E[(x_k - \hat{x}_k^-)(x_k - \hat{x}_k^-)^T] \\
&= E[(Ae_{k-1}^+ + w_{k-1})(Ae_{k-1}^+ + w_{k-1})^T] \\
&= AE[(e_{k-1}^+) + (e_{k-1}^+)^T]A^T + E[(w_{k-1}) + (w_{k-1})^T] \\
&= AP_{k-1}^+A^T + Q_{k-1}
\end{aligned} \tag{5-27}$$

(5-27) eşitliğinde belirtilmiş olan Q_{k-1} sürecin, $k-1$ anındaki öngörülemeyen gürültüsünün kovaryansını ifade eder.

Kalman Süzgeci algoritmasında, (5-25) eşitliğinde k anında olması gereken durumun öngörüsü yapılır. (5-27) eşitliğinde ise bu yapılan öngörüdeki belirsizliği belirten hata kovaryansı hesaplanır. Sonuç olarak durum öngörüsünün yapıldığı bir önceki inanış işlevi, Gauss dağılımı ile (5-28)'deki gibi gösterilebilir.

$$Bel^-(x_k) = N(\hat{x}_k^-, P_k^-) = N(A\hat{x}_{k-1}^+ + Bu_{k-1}, AP_{k-1}^+A^T + Q_{k-1}) \tag{5-28}$$

5.4.2. Bir sonraki inanış işlevi

Kalman Süzgecinde bir sonraki inanış işlevi, bir önceki inanış işlevine benzer bir şekilde, gauss ifadesi ile $Bel^+(x_k) = N(\hat{x}_k^+, P_k^+)$ şeklinde gösterilebilir. Sürecin ölçüm modeli (5-12) eşitliğinde gösterilmiştir. Bu eşitlikte v_k Gauss olasılık

dağılımlı olarak (5-13) eşitliğindeki gibi tanımlanmıştır. Ölçüm modeli de Gauss olasılık dağılımlı olduğu için ölçüm modelinin diğer ifadesi $H_k x_k$ da Gauss olasılık dağılımlı olarak belirtilmelidir.

$$H_k x_k \sim N(H_k x_k, 0) \quad (5-29)$$

(5-13)'de ve (5-29)'da belirtilmiş Gauss olasılık dağılımlarının toplamı sonucunda yine Gauss olasılık dağılımına sahip bir ifade elde edilir. Bu ifadenin olasılıksal yaklaşımı, *olasılıksal konum belirleme yaklaşımı* bölümünde belirtilmiş olan algılama modelinde, robotun k anında x_k konumunda olma olasılığı (5-3) eşitliğindeki ifadeye eşittir.

$$p(z_k | x_k) = N(H_k x_k, R) \quad (5-30)$$

(1-28) eşitliğinde elde edilmiş bir önceki inanış işlevi ve (1-30) eşitliğinde elde edilmiş Gauss temelli olasılık ifadeleri, (5-7) eşitliğinde yerlerine konulduğunda bir sonraki inanış işlevi elde edilmiş olur.

$$Bel^+(x_k) = \frac{N_{z_k}(Hx_k, R_k) \cdot N_{x_k}(\hat{x}_k^-, P_k^-)}{P(z_k | z_1, a_1, \dots, z_{k-1}, a_{k-1})} \quad (5-31)$$

(5-31) eşitliğinde N_{z_k} ve N_{x_k} , sırasıyla z_k uzayındaki Gauss olasılık dağılımını ve x_k uzayındaki olasılık dağılımını ifade eder. Bu iki ayrı uzaydaki olasılık dağılımlarının aynı uzayda tanımlanması gerekmektedir. (5-30)'daki ölçüm uzayında tanımlanan olasılık dağılımını durum uzayında tanımlamak gerekir. Bu dönüşüm işlemi [18]'de gösterilmiştir.

$$N_{z_k}(Hx_k, R_k) = k(z_k) N_{x_k}(\mu, \Sigma), \quad (5-32)$$

$$\mu = (H^T R_k^{-1} H)^{-1} H^T R_k^{-1} z_k$$

$$\Sigma = (H^T R_k^{-1} H)^{-1}$$

(5-31) eşitliğini tekrar yazdığımızda,

$$Bel^+(x_k) = \frac{k(z_k) N_{x_k}(\mu, \Sigma) \cdot N_{x_k}(\hat{x}_k^-, P_k^-)}{P(z_k | z_1, a_1, \dots, z_{k-1}, a_{k-1})} \quad (5-33)$$

iki Gauss olasılık dağılımının çarpım ifadesi oluşur. Bu ifade tekrar Gauss olasılık dağılımını ifade edecek şekilde yazılabilir. Bu ifade [18]'e göre aşağıdaki gibi yazılabilir.

$$Bel^+(x_k) = \frac{k(z_k)}{P(z_k | z_1, a_1, \dots, z_{k-1}, a_{k-1})} N_{x_k}(\mu'', \Sigma'') N_{x_k}(\mu', \Sigma'), \quad (5-34)$$

$$\begin{aligned} \mu' &= ((P_k^-)^{-1} + \Sigma^{-1})^{-1} ((P_k^-)^{-1} \hat{x}_k^- + \Sigma^{-1} \mu) \\ \Sigma' &= ((P_k^-)^{-1} + \Sigma^{-1})^{-1} \\ \mu'' &= \mu \\ \Sigma'' &= P_k^- + \Sigma \end{aligned}$$

Bu ifadedeki $\frac{k(z_k) N_{x_k}(\mu'', \Sigma'')}{P(z_k | z_1, a_1, \dots, z_{k-1}, a_{k-1})}$ [18]'e göre 1 değerine eşittir. Sonuç olarak bir sonraki inaniş işlevi Gauss olasılık dağılımı ifadesi ile yazılabilir.

$$Bel^+(x_k) = N(\hat{x}_k^+, P_k^+) = N_{x_k}(\mu', \Sigma') \quad (5-35)$$

$$\begin{aligned} \hat{x}_k^+ &= \mu' \\ &= \left((P_k^-)^{-1} + (H^T R_k^{-1} H)^{-1} \right)^{-1} \\ &\quad \times \left((P_k^-)^{-1} \hat{x}_k^- + (H^T R_k^{-1} H)^{-1} (H^T R_k^{-1} H)^{-1} H^T R_k^{-1} z_k \right) \\ &= \left((P_k^-)^{-1} + H^T R_k^{-1} H \right)^{-1} \times \left((P_k^-)^{-1} \hat{x}_k^- + H^T R_k^{-1} z_k \right) \end{aligned} \quad (5-36)$$

Bir sonraki durumun beklenen değeri (5-36) eşitliğinde gösterilmiştir. Bazı matris özellikleri sonucunda bu ifade (5-37)'deki kısaltılmış hale getirilebilir.

$$\hat{x}_k^+ = \hat{x}_k^- + \left(P_k^- H^T (H P_k^- H^T + R_k)^{-1} \right) (z_k - H \hat{x}_k^-) \quad (5-37)$$

(5-37) eşitliğinde Kalman kazancı açık olarak yazılmıştır. Burada Kalman kazancını K ile gösterebiliriz.

$$K = P_k^- H^T (H P_k^- H^T + R_k)^{-1} \quad (5-38)$$

(5-35) eşitliğinde belirtilmiş olan bir sonraki durum kestirimindeki hata kovaryansı Σ' şu şekilde ifade edilebilir.

$$\begin{aligned}
P_k^+ &= \Sigma' \\
&= \left((P_k^-)^{-1} + \left((H^T R_k^{-1} H)^{-1} \right)^{-1} \right)^{-1} \\
&= \left((P_k^-)^{-1} + H^T R_k^{-1} H \right)^{-1} \\
&= P_k^- - P_k^- H^T (H P_k^- H^T + R_k^{-1})^{-1} H P_k^- \\
&= (I - KH) P_k^-
\end{aligned} \tag{5-39}$$

Kalman Süzgeci algoritmasında, (5-37) eşitliğinde k anında öngörüsü yapılmış olan durumun düzeltimi yapılır. (5-39) eşitliğinde ise bu yapılan düzeltim işleminin belirsizliğini belirten hata kovaryansı hesaplanır. Sonuç olarak durum düzeltiminin yapıldığı bir sonraki inanış işlevi, Gauss dağılımı ile (5-40)'daki gibi gösterilebilir.

$$Bel^+(x_k) = N(\hat{x}_k^+, P_k^+) = N(\hat{x}_k^- - K(z_k - H\hat{x}_k^-), (I - KH)P_k^-) \tag{5-40}$$

5.4.3. Doğrusal Kalman Süzgeci algoritması

Kalman Süzgeci temel olarak 3 başlıktan oluşur.

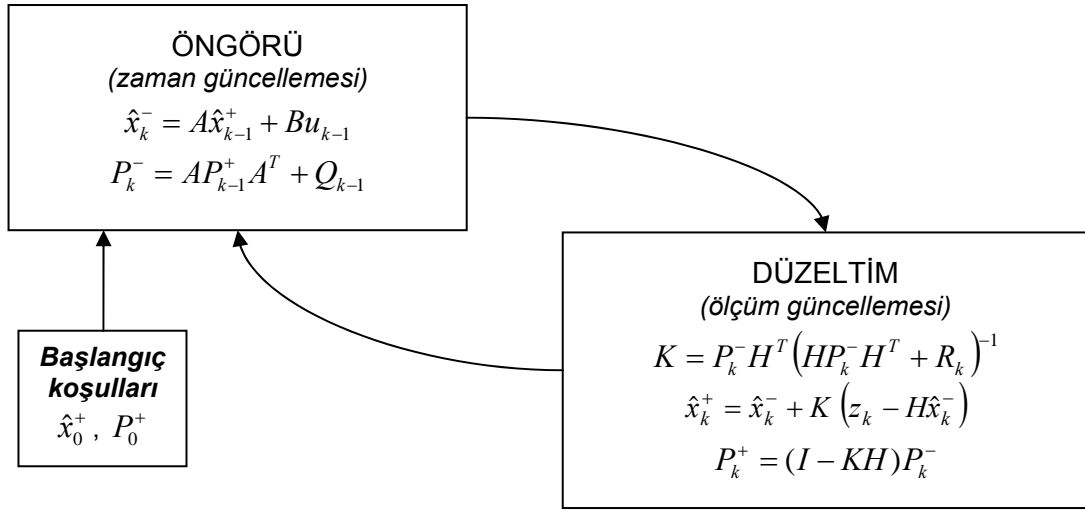
- Başlangıç koşulu
- Öngörü
- Düzeltim

Başlangıç koşulu nun belirlenmesi için $Bel^+(x_0) = N(\hat{x}_0^+, P_0^+)$ eşitliğindeki başlangıç durumu \hat{x}_0^+ ve başlangıç belirsizliği P_0^+ verilmelidir.

Öngörünün yapılabilmesi için $Bel^-(x_k) = N(\hat{x}_k^-, P_k^-)$ eşitliğindeki bir önceki durumun beklenen değeri \hat{x}_k^- ve bu kestirimin belirsizliği P_k^- hesaplanmalıdır. Öngörü bölümü zaman güncellemesinin yapıldığı bölümdür.

Düzeltilimin yapılabilmesi için $Bel^+(x_k) = N(\hat{x}_k^+, P_k^+)$ eşitliğindeki bir sonraki durumun beklenen değeri \hat{x}_k^+ ve düzeltimdeki belirsizlik P_k^+ hesaplanmalıdır.

Düzeltilim sonucu elde edilmiş bir sonraki inanış işlevi, bir zaman sonra bir önceki inanış işlevinin yerini alır. Bu özyinelemeli algoritma şekilde gösterilmiştir.



Şekil 5.4 Doğrusal Kalman Süzgeci Algoritması

5.5. Genişletilmiş Kalman Süzgeci

Kalman Süzgeci doğrusal olmayan sistemlerin kestirimi için kullanılabilmesi için bazı tanımlamalar yapılmalıdır. Bazı sistemler (5-11) eşitliğinde gösterildiği gibi doğrusal olarak modellenemeyebilir. Doğrusal olmayan rassal süreç için, bir zaman önceki durum ile şimdiki durum arasındaki dönüşümü sağlayan işlev (5-41)'deki gibi yazılabilir.

$$\hat{x}_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (5-41)$$

f işlevi bir zaman önceki durum ile şimdiki durum arasındaki doğrusal olmayan ilişkiyi tanımlar. Ölçüm modelinde de h işlevi kullanılarak k anındaki durum ile ölçümler arasındaki ilişki belirtilebilir.

$$z_k = h(x_k, v_k) \quad (5-42)$$

Kalman Süzgeci'nin doğrusal olmayan sistemler için kullanılması için (5-41) ve (5-42) eşitliklerindeki f ve h işlevlerinin doğrusallaştırılması gerekmektedir. Bunun nedeni, *Gauss olasılık yoğunluk işlevinin doğrusal olmayan dönüşüm sonucu artık Gauss özelliği göstermemesidir.*

Durum denklemindeki doğrusal olmayan f işlevi, sözde(*nominal*) yörünge olarak adlandırılan,

$$x_k^{nom} = f(x_{k-1}^{nom}) \quad (5-43)$$

durum denklemini sağlayan, $\hat{x}_0^{nom} = E(\hat{x}_0) = m_0$ başlangıçlı $\{x_k^{nom} : k = 0,1,2,\dots\}$ dizisinin etrafında *Taylor* serisi açılarak doğrusallaştırılabilir. f sürekli türevlere sahip olduğunda yörünge üzerindeki bozulma etkileri nominal yörünge etrafında yapılan *Taylor* açılımı ile ifade edilebilir. ¹

$$f(x + \delta x) = f(x) + f^{(1)}(x)\delta x + \frac{f^{(2)}(x)}{2!}\delta x^2 + \dots + \frac{f^{(n)}(x)}{n!}\delta x^n + \dots, \quad (5-44)$$

δ sözde yörüngedeki bozulmayı gösterirse, $\delta x_k = x_k - x_k^{nom}$ gösterimi altında f işlevinin x_{k-1}^{nom} noktası komşuluğundaki *Taylor* açılımından,

$$\begin{aligned} x_k &= f(x_{k-1}), \\ &= f(x_{k-1}^{nom}) + \left. \frac{\partial f(x_{k-1})}{\partial x} \right|_{x_{k-1}=x_{k-1}^{nom}} \times \delta x_{k-1} + h.o.t., \\ &= x_k^{nom} + \left. \frac{\partial f(x_{k-1})}{\partial x} \right|_{x_{k-1}=x_{k-1}^{nom}} \times \delta x_{k-1} + h.o.t. \end{aligned} \quad (5-45)$$

olup, $h.o.t.$ ² ihmal edilmesi ile

$$\delta x_k = \left. \frac{\partial f(x_{k-1})}{\partial x} \right|_{x_{k-1}=x_{k-1}^{nom}} \times \delta x_{k-1} \cong A_k \delta x_{k-1} \quad (5-46)$$

$$A_k = \left. \frac{\partial f(x_{k-1})}{\partial x} \right|_{x_{k-1}=x_{k-1}^{nom}} \quad (5-47)$$

A_k *Jacobian* matrisi elde edilmiş olur.

Ölçüm denklemindeki doğrusal olmayan h işlevi,

¹ Taylor açılımındaki $f^{(i)}$ terimi f 'in x 'e göre i mertebesindeki türevini belirtir.

² $h.o.t.$ = yüksek mertebeli terimler

$$z_k^{nom} = h(x_k^{nom}) \quad (5-48)$$

durum denklemini sağlayan, $\{z_k^{nom} : k = 0,1,2,\dots\}$ dizisinin etrafında *Taylor* serisi açılarak doğrusallaştırılabilir.

Δ nominal ölçüm yörüngesindeki bozulmayı gösterirse, $\Delta z_k = z_k - z_k^{nom}$ gösterimi altında h işlevinin z_k^{nom} noktası komşuluğundaki *Taylor* açılımından,

$$\begin{aligned} z_k &= h(x_k), \\ &= h(x_k^{nom}) + \left. \frac{\partial h(x_k)}{\partial x} \right|_{x_k=x_k^{nom}} \times \Delta x_k + h.o.t., \\ &= z_k^{nom} + \left. \frac{\partial h(x_k)}{\partial x} \right|_{x_k=x_k^{nom}} \times \Delta x_k + h.o.t. \end{aligned} \quad (5-49)$$

olup, *h.o.t.* ihmal edilmesi ile

$$\Delta z_k = \left. \frac{\partial h(x_k)}{\partial x} \right|_{x_k=x_k^{nom}} \times \Delta x_k \cong H_k \Delta x_k \quad (5-50)$$

$$H_k = \left. \frac{\partial h(x_k)}{\partial x} \right|_{x_k=x_k^{nom}} \quad (5-51)$$

H_k *Jacobian* matrisi elde edilmiş olur.

5.5.1. Genişletilmiş Kalman Süzgeci algoritması

Kalman Süzgeci'nin doğrusal olmayan sistemler için kullanılması için yapılan doğrusallaştırma işlemi sırasında, durumun ya da ölçümün sözde yörüngesi çevirimiçi olarak hesaplanırsa Genişletilmiş Kalman Süzgeci algoritması kullanılmış olur.

$$x_{k-1}^{nom} = \hat{x}_{k-1}^+ \quad (5-52)$$

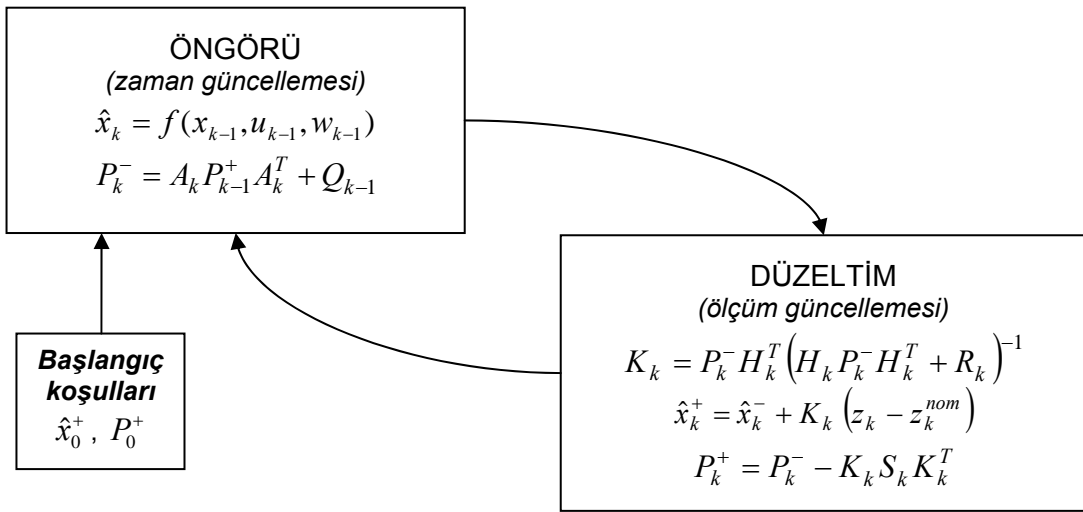
(5-52) eşitliğinde sistem durumunun kestirilmiş son hali durumun sözde yörüngesi olarak kabul edilmiştir. Ölçüm sözde yörüngesi de duruma bağlı olduğu için ölçüm yörüngesi de kestirilmiş duruma göre değişiklik gösterecektir.

Dolayısıyla A_k ve H_k *Jacobian* matrisleri çevrimiçi olarak, durumun her yeni kestiriminde yeniden hesaplanacaktır.

$$A_k = \left. \frac{\partial f(x)}{\partial x} \right|_{x=\hat{x}_{k-1}^+} \quad (5-53)$$

$$H_k = \left. \frac{\partial h(x)}{\partial x} \right|_{x=\hat{x}_k^-} \quad (5-54)$$

k alt indisi her k zamanı için bu matrislerin farklılığını belirtir. Bu matrislerin her zaman için yeniden hesaplanması süre sorununun oluşması demektir.



Şekil 5.5 Genişletilmiş Kalman Süzgeci Algoritması

Genişletilmiş Kalman Süzgeci algoritması, Şekil 5-4'de gösterilmiş doğrusal Kalman Süzgeci algoritmasından çok farklı değildir. Ancak burada dikkat edilmesi gereken A ve H matrislerinin bir önceki durum kestirimine göre yeniden hesaplanıyor olmasıdır.

Bir sonraki durum belirsizliğinin hesaplanmasında kısaca S_k ile belirtilen matris Kalman kazancının hesaplanmasında kullanılan yenilenme kovaryans matrisidir(5-55).

$$S_k = H_k P_k^- H_k^T + R_k \quad (5-55)$$

6. ROBOT SİSTEMİ VE MODELLER

6.1. ActivMedia Robotlar

Her ActivMedia robot dengeli sürüş sisteminden(ör: diferansiyel iki tekerlek ve bir mobilya tekerleği), tersine dönebilen DA motorlar, motor sürücü ve denetim elemanları, yüksek çözünürlüklü hareket kodlayıcılardan oluşmaktadır. Bütün bu elemanlar, robotların üzerlerinde bulunan işlemci ve gezgin robot yazılımları sayesinde yönetilirler. Bu robotlar iki tekerli ve dört tekerli yapılara sahiptirler.



Şekil 6.1 1995 yılında üretilen Pioneer1 robotu

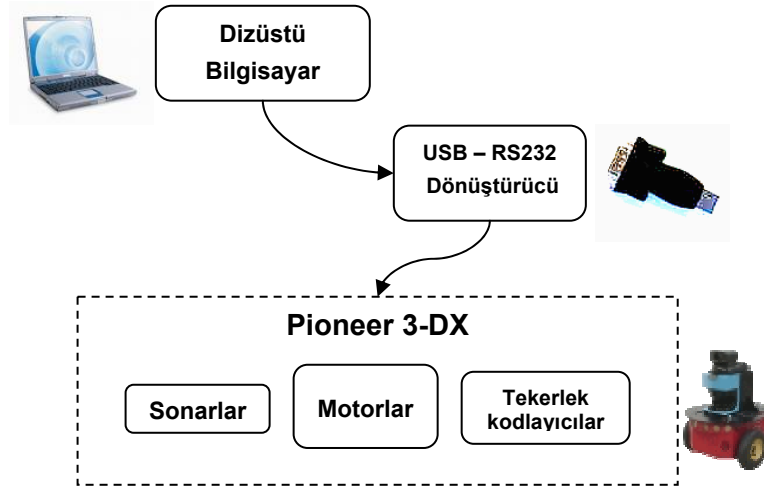
ActivMedia robotların yukarıda da belirtilen özelliklerini denetleyebilmesi için geliştirilmiş ARIA(Advanced Robotics Interface for Applications) adlı yazılıma sahiptir. Bu yazılım C++ tabanlı açık-kaynak geliştirilmeli bir yazılımdır. ARIA yazılımı sayesinde ActivMedia robotların denetimi ve yedek parça sistemlerindeki değişimlere göre uyum sağlayacak bir iletişim arayüzü sağlanmıştır.

Tez çalışmasında kullanılan Pioneer 3-DX modelinin kullanılan özellikleri:

- 4 tane RS232 seri port, 9.6 ile 115.2 kilobaud arasında ayarlanabilir,
- 4 tane Sonar dizisi, her birinde en fazla 8 sonar
- Yüksek çözünürlüklü hareket kodlayıcıları
- Kullanıcı denetim paneli
- Ana güç ve güç seviyesi gösterici iki renkli LED
- RESET ve MOTORS basma düğmeleri
- Piezo buzzer

Pioneer 3-DX robotunun daha ayrıntılı özellikleri EK 1'de verilmiştir.

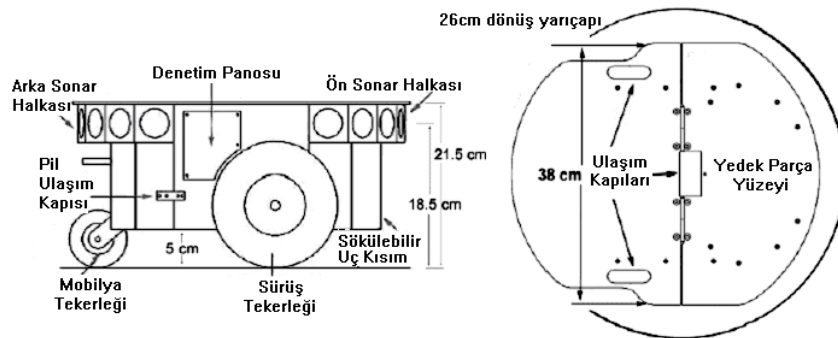
Pioneer 3-DX, üzerinde bulunan kişisel bilgisayar(PC) ile birlikte, tam olarak otonom, akıllı gezgin robot özelliğine sahiptir. Eğer robotun tam olarak otonom akıllı gezgin robot olması isteniyorsa, bunun için üzerinde bulunan bilgisayar aracılığı ile işlevselliğinin gerçekleştirilmesi gerekmektedir. Pioneer 3-DX'in belirtilen bağlantı şekillerinden Şekil 1-2'de gösterilmiş olan donanım bağlantı yapısı kullanılmıştır.



Şekil 6.2 Pioneer 3-DX donanım bağlantıları

6.1.1. Dış yapısı

Daha yaygın olan robotlara oranla Pioneer 3-DX'in makul boyutu, dar ve karmaşık alanlarda gezinim yapmasına imkan sağlamaktadır.



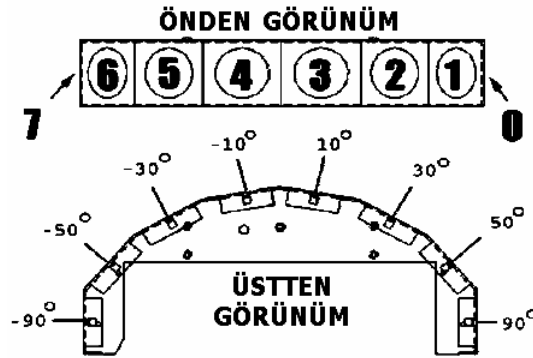
Şekil 6.3 Pioneer 3-DX'in fiziksel boyutu ve dönüş yarıçapı

Şekil 6.3'de gösterildiği gibi, Pioneer 3-DX'in dönüş yarıçapı, yani kendi etrafında döndüğü sırada kullandığı serbest uzayın iki boyuttaki yarıçapı 26 cm'dir.

Sonar algılayıcıların merkezlerinin yerden yüksekliği 18.5 cm robotun üzerinde yük olmadığındaki yüksekliği de 21.5 cm'dir.

6.1.2. Ses ötesi algılayıcılar(sonarlar)

ARCOS(ActivMedia's Robot Control and Operations Software)- tabanlı ActivMedia robotları en fazla 4 ses ötesi algılayıcı dizisine sahiptirler. Her bir dizide en fazla sekiz adet alıcı-verici bulunmaktadır. Robotlar bu alıcı-vericileri nesnelere mesafelerinin ölçümlerini yapmada kullanırlar. Yaptıkları bu mesafe ölçümlerinin değerlendirilmesi sayesinde robotlar kendi konumlarını ve etrafındaki nesnelere kendilerine göre konumlarını bulabilirler.



Şekil 6.4 Pioneer 3-DX ses ötesi algılayıcı dizisi

Pioneer 3-DX üzerindeki ses ötesi algılayıcılar robotun önünde ve arkasında sekizer tane olmak üzere yerleştirilmişlerdir. Şekil 6.4'te robot üzerinde bulunan ses ötesi algılayıcı dizilerinden bir tanesi gösterilmiştir. Bu dizide bulunan ses ötesi algılayıcılardan iki tanesi robotun her iki yanında geri kalan altı tanesi de 20'şer derecelik açı aralıklarıyla, merkezleri yerden 18.5 cm yüksekliğe gelecek şekilde yerleştirilmiştir. Robotun önüne ve arkasına yerleştirilecek bu ses ötesi algılayıcı dizileri sayesinde, neredeyse tamamı algılanabilen, 360 derecelik algılama uzayı elde edilmiştir.

Çizelge 6.1 Pioneer 3-DX Ön Sonarlarının Konumları

Sonar Numarası	X(cm)	Y(cm)	Th(derece)
0	0.069	0.136	90
1	0.114	0.119	50
2	0.148	0.078	30
3	0.166	0.027	10
4	0.166	-0.027	-10
5	0.148	-0.078	-30
6	0.114	-0.119	-50
7	0.069	-0.136	-90

Pioneer 3-DX üzerinde bulunan ses ötesi algılayıcı dizileri çoklanmış yapıda bağlanmışlardır. Ses ötesi algılayıcı dizisinde sekiz diskten her seferinde yalnızca bir disk çalışır. Bunun yanında dört dizi aynı anda bir tane alıcı-vericiyi çalıştırabilir. Pioneer 3-DX üzerindeki ses ötesi algılayıcıları 25 Hz'lik(40 milisaniye her bir alıcı-verici için) sıklığa ayarlanmıştır. Algılayıcıların hassasiyeti yaklaşık olarak 10 cm ile 4 m arasındadır. Algılayıcıların çalıştırılma sırası soldan sağa doğru yani 0 numaralı algılayıcıdan 7 numaralı algılayıcıya doğrudur. İstenildiği takdirde bu sıra yazılımsal olarak değiştirilebilir.

6.1.3. Motorlar, tekerlekler ve pozisyon kodlayıcıları

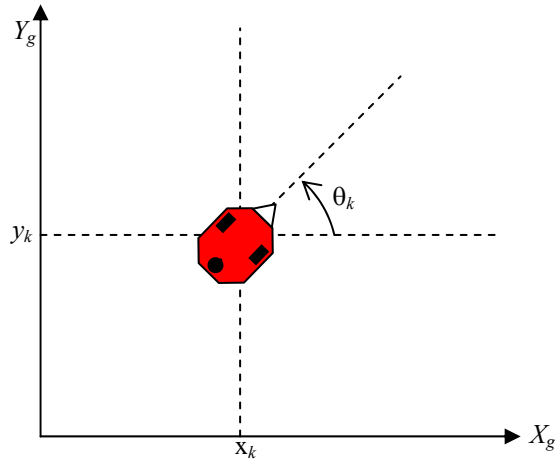
Pioneer 3-DX yüksek hızlı, yüksek torklu, çift yönlü DA motorlara sahiptir. Pioneer 3-DX robotlar üzerinde dolma teker kullanılmaktadır. Bu tekerleklerin çapları 19.53 cm'dir. Tekerleklerin aşınması sonucu bu değerde değişiklik olabilir. Motorların şaftı üzerine yerleştirilmiş yüksek çözünürlüklü kodlayıcılar sayesinde robotun hız, yön ve aldığı mesafe bilgileri elde edilebilmektedir. Kullanılan kodlayıcılar bir dönüşte 500 adet vuruş üretebilme özelliğine sahiptirler. Pioneer-3DX robotunda kullanılan kodlayıcı ve hareket elemanları için bazı değerler çizelge 6.2'de verilmiştir.

Çizelge 6.2 Robot yapısına bağlı birkaç parametre

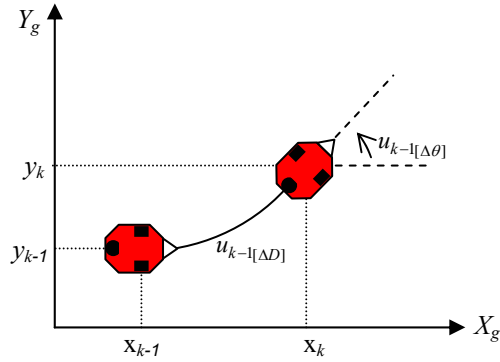
Değerler	
PARAMETRELER	P3-DX
kodlayıcı vuruş/dönüş	500
dişli oranı	38.3:1
tekerlek çapı (mm)	195.3
kodlayıcı vuruş/mm	128

6.2. Sistem Modeli

Kullanılan robotun tekerlekleri birbirlerinden bağımsız olarak denetlenebilmektedir. Tekerleklerin bağımsızlığı robotun iki boyutlu düzlemde hareketi gerçekleştirmesini sağlar. Robotun bütünsel ortamdaki konumu x_k , y_k , θ_k parametreleriyle tanımlanabilir. x_k ve y_k parametreleri robotun k anında ortamdaki konumunu θ_k ise robotun bütünsel düzleme göre açılal durumunu ifade eder(Şekil 6.1).



Şekil 6.1 Bütünsel ortamda robot konumu



Şekil 6.2 Denetim girdileri ile robotun hareketi

Denetim girdileri u ile robot, yönünü ve konumunu değiştirmiş olur. Şekil 6.2'de robot, u denetim girdileri sayesinde (x_{k-1}, y_{k-1}) konumundan (x_k, y_k) konumuna hareket etmiştir. Bu hareketi sırasında robot ΔD mesafesi kadar ve $\Delta \theta$ açısı kadar yerini değiştirmiştir. u denetim girdisi sayesinde robotun ne kadar yer değiştireceği, robotun hareket elemanlarının fiziksel boyutlarına bağlıdır.

$$u_k = [u_{k[\Delta D]} \quad u_{k[\Delta\theta]}]^T \quad (6-1)$$

Bu tez çalışmasındaki denetim girdileri, robotun doğrusal hızı ve dönme hızı ile belirtilmiştir. Robotun doğrusal hızı $u_{k[\Delta D]}$, her iki tekerleğinin dönüş hızlarının ortalamalarından hesaplanır. Robotun dönüş hızı $u_{k[\Delta\theta]}$ ise her bir tekerleğin dönüş hızları arasındaki fark ve tekerleklerin merkezleri arasındaki uzaklıktan hesaplanır.

$$v_k = \frac{v_{kR} + v_{kL}}{2} = \frac{R \cdot w_{kR} + R \cdot w_{kL}}{2} \quad (6-2)$$

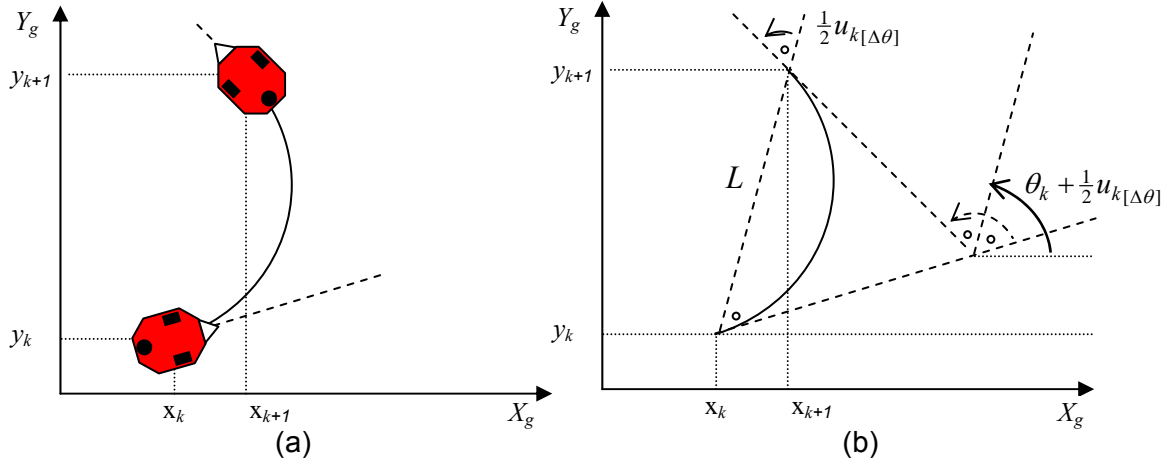
$$w_k = \frac{v_{kR} - v_{kL}}{b} = \frac{R \cdot w_{kR} - R \cdot w_{kL}}{b} \quad (6-3)$$

$$u_{k[\Delta D]} = v_k \cdot t_{samp} \quad , \quad u_{k[\Delta\theta]} = 0 \quad (6-4)$$

$$u_{k[\Delta\theta]} = w_k \cdot t_{samp} \quad (6-5)$$

(6-2) eşitliğinde v_{kR} ve v_{kL} sırasıyla sağ ve sol tekerleğin doğrusal hızlarını, (6-3) eşitliğindeki w_{kR} ve w_{kL} sırasıyla sağ ve sol tekerleğin açısal hızlarını, b ise tekerleklerin merkezleri arasındaki uzaklığı belirtir. (6-4) ve (6-5) eşitliklerindeki t_{samp} $k-1$ anından k anına kadar geçen süreyi kısacası örnekleme zamanını belirtir.

Robotun denetim girdileri ile bütünsel düzlemde ne kadar yer değiştirdiği şekil 6.3'te gösterilmiştir. Robot u_k denetim girdileri ile k anında (x_k, y_k, θ_k) konumundan bir zaman sonra $k+1$ anında $(x_{k+1}, y_{k+1}, \theta_{k+1})$ konumuna gelecektir.



Şekil 6.3 Robotun dönüş hareketi

Başlangıçta robotun bütünsel düzleme göre θ_k kadar bir açısı vardır. θ_k açısı X_g ile robot doğrultusu arasındaki açıdır. Robot $u_k = [u_{k[\Delta D]}, u_{k[\Delta \theta]}]^T$ girdisi ile ΔD kadar mesafe yol alacak ve $\Delta \theta$ açısı kadar yönünü değiştirecektir. Şekil 6.4'te robot yaptığı döngüsel hareket sonucunda, L uzaklığında yeni bir konumuna gelir.

Elde edilmek istenen yeni \hat{x}_k konumu, (6-6) denkleminde belirtildiği gibi, durum denklemleri ile ifade edilebilir. (5,41) denkleminde belirtilen doğrusal olmayan sistemin ifadesi açık olarak yazılırsa;

$$\hat{x}_k = f(x_{k-1}, u_{k-1}, w_{k-1}) = \begin{bmatrix} f_x(x_{k-1}, u_{k-1}, w_{k-1}) \\ f_y(x_{k-1}, u_{k-1}, w_{k-1}) \\ f_\theta(x_{k-1}, u_{k-1}, w_{k-1}) \end{bmatrix} \quad (6-6)$$

$$f_x = x_{k-1[x]} + u_{k-1[\Delta D]} \cdot \cos(x_{k-1[\theta]} + \frac{1}{2} u_{k-1[\Delta \theta]}) \quad (6-7)$$

$$f_y = x_{k-1[y]} + u_{k-1[\Delta D]} \cdot \sin(x_{k-1[\theta]} + \frac{1}{2} u_{k-1[\Delta \theta]}) \quad (6-8)$$

$$f_\theta = x_{k-1[\theta]} + u_{k-1[\Delta \theta]} \quad (6-9)$$

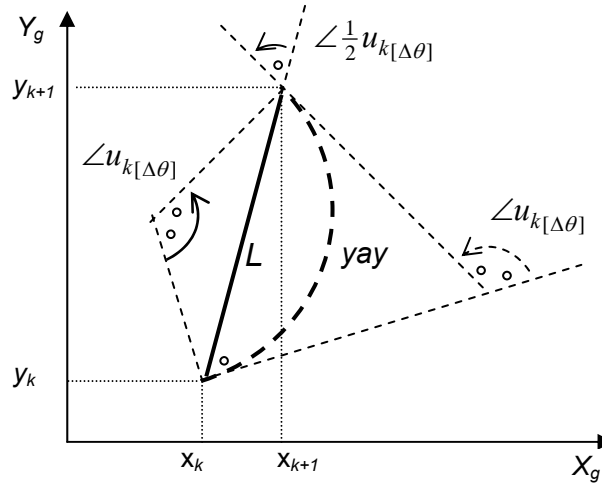
(6-7), (6-8) ve (6-9) eşitliklerinde sistem gürültüsü dahil edilmemiştir. Robot sisteminin gürültüsü denetim girdilerinden kaynaklanır. Denetim girdilerinin tekerlekler üzerinde bulunan kodlayıcılardan elde edildiği dikkate alınır, kodlayıcıların hassasiyeti sistem gürültüsünün değişimini etkileyecektir. (6-1) eşitliğinde belirtilmiş denetim girdileri gürültülü olarak ifade edilirse (6-10) eşitliği ile

gösterilebilir. Robotun doğrusal olmayan sistem modeli (6-11) eşitliğindeki gibi ifade edilebilir.

$$u_k = \begin{bmatrix} u_k[\Delta D] \\ u_k[\Delta \theta] \end{bmatrix} + \begin{bmatrix} q_k[\Delta D] \\ q_k[\Delta \theta] \end{bmatrix} \quad (6-10)$$

$$\hat{x}_k = \begin{bmatrix} x_{k-1}[x] + (u_{k-1}[\Delta D] + q_k[\Delta D]) \cdot \cos\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta \theta] + q_k[\Delta \theta])\right) \\ x_{k-1}[y] + (u_{k-1}[\Delta D] + q_k[\Delta D]) \cdot \sin\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta \theta] + q_k[\Delta \theta])\right) \\ x_{k-1}[\theta] + u_{k-1}[\Delta \theta] + q_k[\Delta \theta] \end{bmatrix} \quad (6-11)$$

Şekil 6.5'te gösterildiği gibi robotun hareketi dögüsel olduğunda (6-11) eşitliğinde gösterilmiş olan robotun doğrusal hızı $u_k[\Delta D]$ (6-4) eşitliğinden farklı olur. Bunun nedeni robotun dönüş sırasında alınan mesafe ile robot konumunun düzlemsel değişiminin birbirinden farklı olmasıdır. Şekil 6.5'te gösterilen dönüş hareketi sırasında robot, $yay = v_k \cdot t_{samp}$ kadar yol almasına karşılık L uzaklığı kadar bir yer değiştirme yapmıştır. Sonuç olarak robotun X-Y koordinatındaki yer değişimi, kısacası L uzaklığı yay denkleminde (6-12)'deki gibi elde edilir.



Şekil 6.4 Robotun dönüş hareketi

$$L = \frac{1}{(u_{k-1}[\Delta \theta] + q_k[\Delta \theta]) \times \pi} \times yay \times 360 \times \sin\left(\frac{u_{k-1}[\Delta \theta] + q_k[\Delta \theta]}{2}\right) \quad (6-12)$$

(6-12) eşitliğinde $u_{k-1[\Delta\theta]} + q_{k[\Delta\theta]} = 0$ olduğunda $L = u_{k-1[\Delta D]} + q_{k[\Delta D]}$ olur.

(6-11) eşitliğindeki durum denklemleri, (6-12) eşitliği doğrultusunda tekrar yazılırsa (6-13) eşitliği elde edilmiş olur.

$$\hat{x}_k = \begin{bmatrix} x_{k-1[x]} + L \cdot \cos\left(x_{k-1[\theta]} + \frac{1}{2}(u_{k-1[\Delta\theta]} + q_{k[\Delta\theta]})\right) \\ x_{k-1[y]} + L \cdot \sin\left(x_{k-1[\theta]} + \frac{1}{2}(u_{k-1[\Delta\theta]} + q_{k[\Delta\theta]})\right) \\ x_{k-1[\theta]} + u_{k-1[\Delta\theta]} + q_{k[\Delta\theta]} \end{bmatrix} \quad (6-13)$$

Süreç gürültüsünün kovaryansı, birbirinden bağımsız olduğu kabul edilen iki kaynak ile modellenenir. Bu kaynaklar denetim girdileridir. Süreç gürültüsünün sıfır ortalamalı Gauss olasılık dağılımına sahip olduğu varsayılırsa süreç gürültüsünün kovaryansı (6-16) eşitliğindeki gibi yazılabilir.

$$q_k \sim N(\hat{q}_k, U_k) \quad (6-14)$$

$$\hat{q}_k = [0 \ 0]^T \quad (6-15)$$

$$U_k = \begin{bmatrix} \sigma^2_{q_{k[\Delta D]}} & 0 \\ 0 & \sigma^2_{q_{k[\Delta\theta]}} \end{bmatrix} \quad (6-16)$$

(6-16) eşitliğindeki kovaryans matrisinin köşegen dışında kalan terimlerinin sıfır olmasının nedeni, gürültü kaynaklarının birbirinden bağımsız olmasının kabul edilmesidir.

Hareket modelinin belirtildiği durum geçiş işlevi bir zaman önceki durum ile şimdiki durum arasındaki doğrusal olmayan ilişkiyi belirtmektedir. (6-13) eşitliğinde belirtilmiş süreci, birbirinden bağımsız olduğu kabul edilen iki gürültü etkilemektedir. Bu gürültülerin sisteme kattığı belirsizliğin hesaplanması gerekmektedir. Bu belirsizlik (5-27) eşitliğinde belirtilmiş olan $k-1$ anındaki öngörülemez gürültü kovaryansı Q_{k-1} 'in hesaplanması ile bulunur. Q_{k-1} 'in hesaplanması için (6-16) eşitliğinde belirtilmiş, sürecin gürültü kovaryansının hesaplanması ve daha sonra denetim girdileri ile durum geçiş işlevinin kısmi türevinin alınması sonucu elde edilen ∇f Jacobian matrisinin kendisi ve devriğinin hesaplanarak çarpılması gerekir. Matematiksel olarak ifade edilecek olursa Q_{k-1} , (6-17) eşitliğindeki gibi bulunur.

$$Q_{k-1} = \Delta f \times U_{k-1} \times \Delta f^T \quad (6-17)$$

$$\Delta f = \left[\frac{\partial f(x_{k-1})}{\partial u_k} \right] = \left[\frac{\partial f(x_{k-1})}{\partial u_{k[\Delta\theta]}} \quad \frac{\partial f(x_{k-1})}{\partial u_{k[\Delta D]}} \right]$$

$$\Delta f = \left[\begin{array}{cc} -L.\sin\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta\theta] + q_{k[\Delta\theta]})\right) & \cos\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta\theta] + q_{k[\Delta\theta]})\right) \\ L.\cos\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta\theta] + q_{k[\Delta\theta]})\right) & \sin\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta\theta] + q_{k[\Delta\theta]})\right) \\ 1 & 0 \end{array} \right] \quad (6-18)$$

(6-17) eşitliğinde ifade edilen Q_{k-1} , sürecin belirsizliğini tanımlar. Başka bir belirsizlik de sistem durumunun belirsizliğidir. Bu belirsizlik, robot konumunun $k-1$ anındaki belirsizliğinin bir sonraki zamandaki konumu nasıl etkileyeceğini ifade eder. Bu belirsizliğin hesaplanabilmesi için (5-53) eşitliğinde belirtilen A_{k-1} Jacobian matrisi hesaplanmalıdır. Durum belirsizliğinin hesaplanması için gerekli A_{k-1} Jacobian matrisi (6-19)'da verilmiştir.

$$A_{k-1} = \left[\frac{\partial f(x_{k-1})}{\partial x} \right] = \left[\frac{\partial f(x_{k-1})}{\partial x_{k-1}[x]} \quad \frac{\partial f(x_{k-1})}{\partial x_{k-1}[y]} \quad \frac{\partial f(x_{k-1})}{\partial x_{k-1}[\theta]} \right]$$

$$A_{k-1} = \left[\begin{array}{ccc} 1 & 0 & -L.\sin\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta\theta] + q_{k[\Delta\theta]})\right) \\ 0 & 1 & L.\cos\left(x_{k-1}[\theta] + \frac{1}{2}(u_{k-1}[\Delta\theta] + q_{k[\Delta\theta]})\right) \\ 0 & 0 & 1 \end{array} \right] \quad (6-19)$$

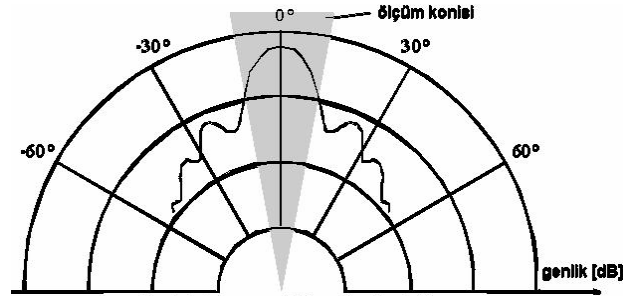
Daha önceki durum belirsizliğinin bir adım sonraki durum belirsizliğini nasıl etkileyeceğinin bulunması için, (5-27)'de belirtilmiş olan bir önceki durum kestirimindeki hata kovaryansı P_{k-1}^+ , A_{k-1} Jacobian matrisinin soldan kendisi ve sağdan devriği ile çarpılmalıdır. Bu belirsizlik hesabına daha önceden hesaplanan Q_{k-1} belirsizliği de eklenerek k anındaki bir önceki durum belirsizliği P_k^- , (6-20) eşitliğindeki gibi bulunur.

$$P_k^- = AP_{k-1}^+ A^T + Q_{k-1} \quad (6-20)$$

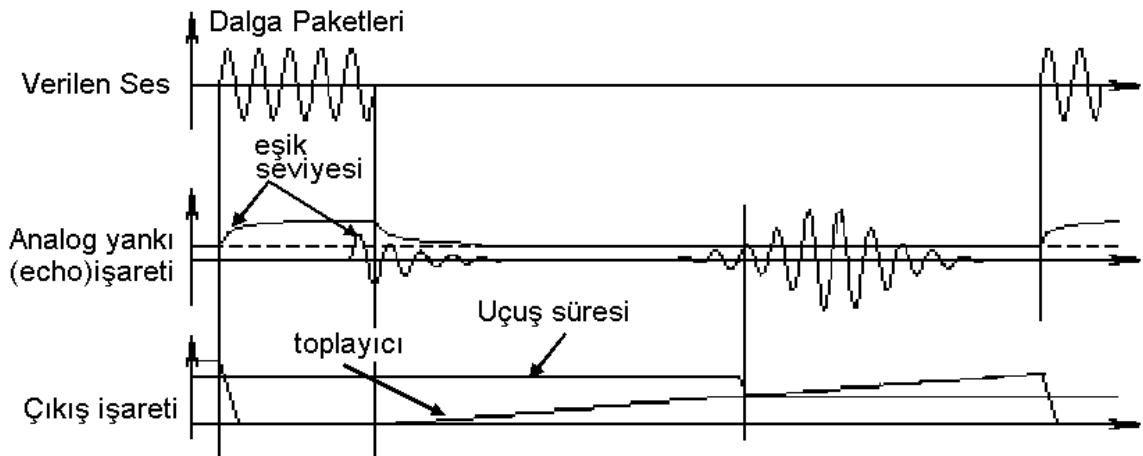
6.3. Ölçüm Modeli

Robot konumunun izlenmesinde kullanılan denetim girdileri kodlayıcılardan elde edilir. Bu izleme yönteminde sadece kodlayıcılardan yararlanılırsa bu yöntem *gözü kapalı tahmin* olarak adlandırılır. Konum takibine yardımcı olması için kullanılan bir diğer algılayıcı da robotun üzerinde konumlandırılmış ses ötesi algılayıcılardır. Bu algılayıcılar robotun bulunduğu ortam hakkındaki verileri elde ederler. Robot üzerinde bulunan her bir sonar algılayıcısının etrafından elde ettiği veri farklıdır. Dolayısıyla robot konumuna göre her algılayıcısının algılama doğrultusunun belirlenmesi gerekmektedir.

Gerçek hayatta sonar algılayıcılar, sesin yayılımı özelliğini kullandıkları için, mesafe bilgisini koni şeklinde bir alandan elde ederler. Ses ötesi algılayıcısının tipik yoğunluk dağılımı şekil 6.6'da gösterilmiştir.

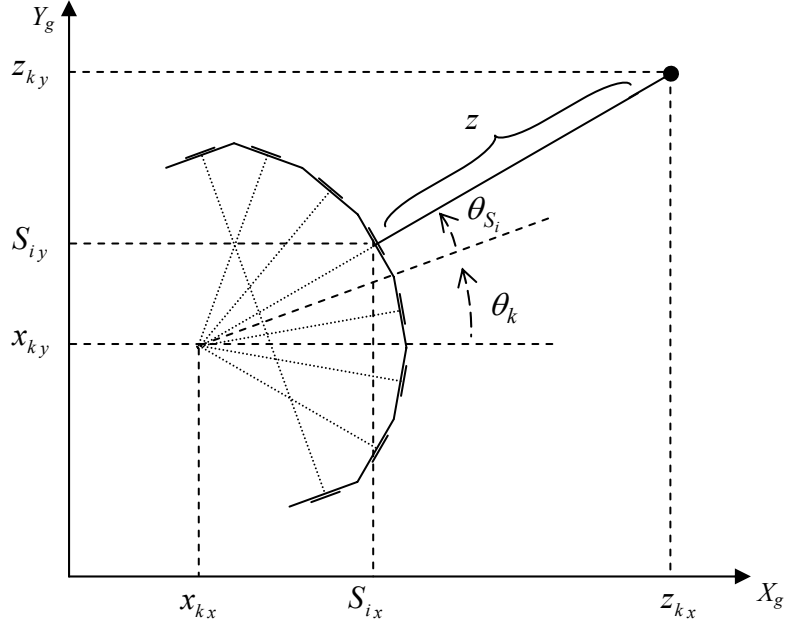


Şekil 6.5 Ses ötesi algılayıcısının tipik yoğunluk dağılımı



Şekil 6.6 Ses ötesi algılayıcısının T.O.F ölçümü

Sonar algılayıcılar sesin uçuş süresi(T.O.F) özelliğini kullanırlar. Sonar algılayıcılar gönderdikleri dalga paketlerinin belirli bir süre geri dönmelerini beklerler. T.O.F gönderilen dalga paketinin geri dönme süresidir.



Şekil 6.7 Sonar algılayıcıların konumu ve algılama doğrultusu

(6,42) eşitliğinde de belirtilen algılama modeli açık olarak (6-21) eşitliğindeki gibi ifade edilebilir.

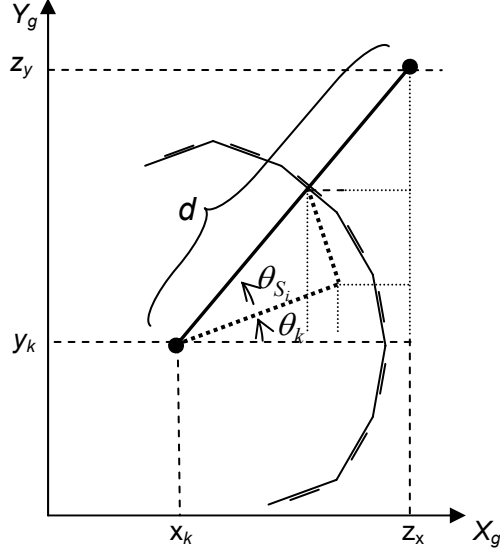
$$z_k = h(x_k, v_k) = \sqrt{(z_{kx} - x_{kx})^2 + (z_{ky} - x_{ky})^2} + v_k \quad (6-21)$$

Şekil 6.8'de de gösterildiği gibi sonar algılayıcı kendi konumu ile engel arasındaki z uzaklığını ölçer. Robotun merkezi ile engel arasındaki mesafe bilgisinin bulunması için, çizelge 6.1'de de belirtilen sonar algılayıcıların konum bilgilerinin, elde edilen z mesafe verisine eklenmesi gerekir. Sonar algılayıcıların konumlarının robotun merkezine olan konumları çizelge 6.1'de gösterilmiştir. Bu verilerden yararlanılarak engel ile robot merkezi arasındaki mesafe bilgisi (6-22) ve (6-23) denklemleri ile hesaplanabilir.

$$d_{xi} = z \cdot \cos(\theta_k + \theta_{Si}) + S_{ix} \cdot \cos(\theta_k) - S_{iy} \cdot \sin(\theta_k) \quad (6-22)$$

$$d_{yi} = z \cdot \sin(\theta_k + \theta_{Si}) + S_{ix} \cdot \sin(\theta_k) + S_{iy} \cdot \cos(\theta_k) \quad (6-23)$$

(6-22) ve (6-23) denklemlerinde yer alan d_{x_i} ve d_{y_i} ifadeleri, i 'nci sonar tarafından algılanan engel ile robot merkezi arasındaki d uzaklığının x ve y bileşenleridir. Şekil 6.9'da bir tane sonarın algıladığı engel ile robot merkezi arasındaki mesafe belirtilmiştir.



Şekil 6.8 d uzaklığı, engel uzaklığı

Her bir sonar algılayıcısından elde edilen veri, belirli bir gürültüye sahiptir. Bu gürültüler (6-21) eşitliğinde dahil edilmiştir. Algılayıcı gürültülerinin sıfır ortalamalı Gauss olasılık dağılımına sahip oldukları varsayılırsa, ölçüm gürültüsünün kovaryansı (6-26) eşitliğindeki gibi yazılabilir.

$$v_k \sim N(\hat{v}_k, R_k) \quad (6-24)$$

$$\hat{v}_k = [0 \ \dots \ 0]^T \quad (6-25)$$

$$R_k = \begin{bmatrix} \sigma^2_{S_1} & & 0 \\ & \ddots & \\ 0 & & \sigma^2_{S_8} \end{bmatrix} \quad (6-26)$$

(6-26) eşitliğindeki gürültü kovaryans matrisinin köşegen dışında kalan terimlerinin sıfır olmasının nedeni, gürültü kaynaklarının birbirinden bağımsız olmasının kabul edilmesidir. (6-26) eşitliğindeki $\sigma^2_{S_i}$ ifadesi i 'nci sonar

algılayıcısındaki hata varyansını ifade eder. (6-25) eşitliğinde \hat{v}_k matrisinin boyutu (8x1), R_k matrisinin boyutu (8x8)'dir.

Ölçüm modelinin belirtildiği (6-21) eşitliğindeki ölçüm hatası, robotun öngörülen konuma geldiğinde elde etmesi gereken ölçüm ile elde ettiği ölçüm arasındaki hatayı belirtir. Bu hatanın robotun konumundaki belirsizliğe dahil edilmesi gerekir. Bu nedenle ölçüm modelinin robot konumuna göre kısmi türevinin alınması ile (5-54) eşitliğinde belirtilmiş H_k Jacobian matrisi hesaplanmalıdır (6-27).

$$H_k = \frac{\partial h(x)}{\partial x} = \begin{bmatrix} \frac{\partial h(x)}{\partial x_{k-1}[x]} & \frac{\partial h(x)}{\partial x_{k-1}[y]} & \frac{\partial h(x)}{\partial x_{k-1}[\theta]} \end{bmatrix}$$

$$H_k = \begin{bmatrix} x_{k[x]} - z_{x1} / d_1 & x_{k[y]} - z_{y1} / d_1 & 0 \\ x_{k[x]} - z_{x2} / d_2 & x_{k[y]} - z_{y2} / d_2 & 0 \\ \vdots & \vdots & \vdots \\ x_{k[x]} - z_{x8} / d_8 & x_{k[y]} - z_{y8} / d_8 & 0 \end{bmatrix} \quad (6-27)$$

(6-27)'deki H_k Jacobian matrisi, (6-26) eşitliğindeki gürültü kovaryans matrisi R_k ve (6-20) eşitliğindeki bir önceki durum belirsizliği P_k^- 'nin bir araya getirilmesi ile Kalman kazancını yenilemede kullanılan yenilenme kovaryans matrisi S_k , (6.28)'deki gibi elde edilir.

$$S_k = H_k P_k^- H_k^T + R_k \quad (6-28)$$

Kalman kazancı (6.28) eşitliğindeki gibi hesaplanır.

$$K_k = P_k^- H_k^T S_k^{-1} \quad (6-29)$$

Bu hesaplamalar sonucunda, bir sonraki durum kestirimindeki hata kovaryansı (6-30)'daki şekilde ifade edilebilir.

$$P_k^+ = P_k^- - K_k S_k K_k^T \quad (6-30)$$

Sonuç olarak bir sonraki durumun kestirimi en son elde edilen sonar verileri ile gerçekleştirilir. Kalman kazancı ile ölçüm hatasının çarpımının duruma eklenmesi ile öngörülen durumun düzeltilmesi gerçekleştirilmiş olur(6-31).

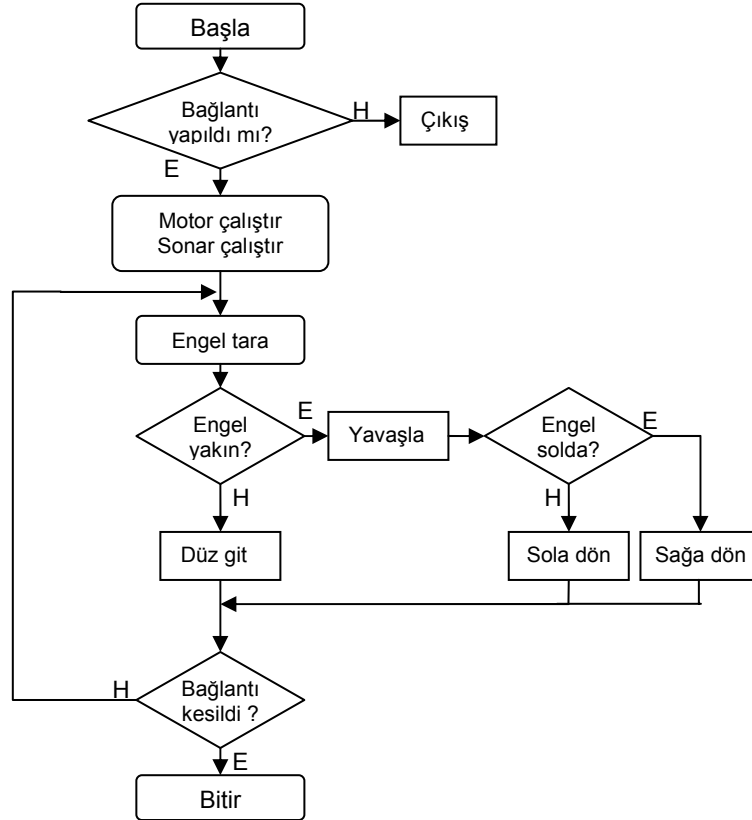
$$\hat{x}_k^+ = \hat{x}_k^- + K_k (z_k - z_k^{nom}) \quad (6-31)$$

7. PİONEER 3-DX ROBOT KONUMUNUN İZLENMESİ

Tez çalışmasında *ActivMedia Robotics* firması tarafından üretilmiş Pioneer 3-DX robotu kullanılmıştır. Robot konumunun izlenmesi işlemleri tez çalışmasında iki şekilde gerçekleştirilmiştir:

- Benzetim ortamında robottan elde edilen veriler ile
- Gerçek ortamda robottan elde edilen veriler ile

Gerçek ortamda robot hareketlerinin gerçekleştirilebilmesi için robot üzerine yerleştirilmiş bir dizüstü bilgisayar kullanılmıştır. Robotun hareketleri ve elde edilecek veriler için C++ tabanlı program yazılmıştır. Yazılan program sayesinde robotun ortam içerisinde engellere çarpmadan hareketi sağlanmıştır. Aynı zamanda bu program benzetim ortamında da kullanılmıştır. Bu programın basit akış diyagramı şekil 7.1'deki gibidir.

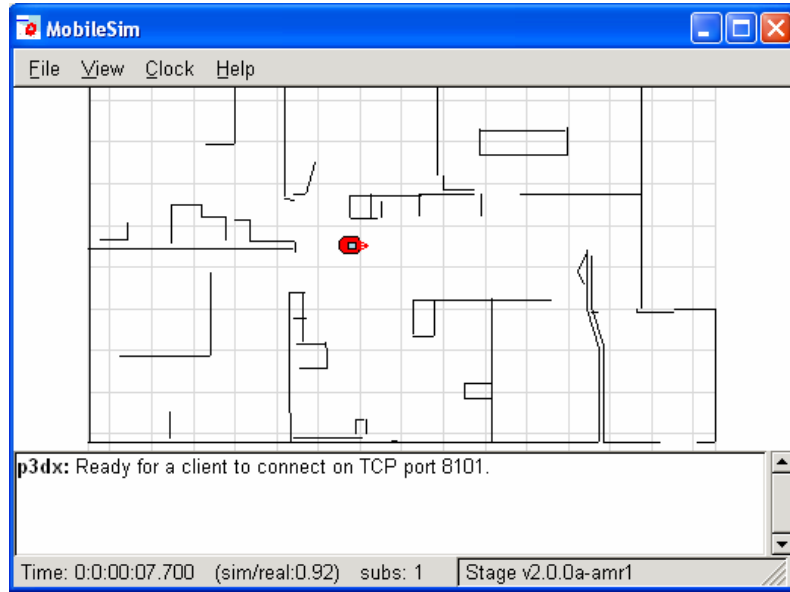


Şekil 7.1 Robot hareketi programı akış diyagramı

Akış diyagramında yer alan “Engel tara” işlemi robot üzerinde bulunan sonar algılayıcılar tarafından gerçekleştirilmektedir. Her bir sonar algılayıcı 40ms aralıklarla engel ile robot arasındaki mesafe bilgilerini elde eder. Engelin yakın ya da uzak olma koşulu yazılan programda verilen parametreye bağlıdır. Bu deneyde parametre 300 yani 30cm olarak verilmiştir. Eğer robot engele daha fazla yaklaşırsa durur ve boş olan alana doğru döner. Engelin solda ya da sağda olduğu sonucuna, sonarlardan elde ettiği verileri karşılaştırarak ulaşır.

7.1. Benzetim Ortamı

Robot hareketini sağlayan programın testi öncelikle benzetim ortamında yapılmıştır. Robotun benzetimi *MobileSim* adlı program aracılığı ile yapılmıştır. Ayrıca robotun konumunun izlenmesi işlemlerinde de bu benzetimciden yararlanılmıştır. Benzetim programının bir görüntüsü şekilde gösterilmiştir.



Şekil 7.2 MobileSim programının görünümü

Bu program sayesinde, robot hareketi için yazılan programların robota aktarılmadan sanal ortamda çalışırılığının testi gerçekleştirilebilmektedir. Program, ortamların 2 boyutlu olarak tanımlanmasına olanak sağlamaktadır. Dolayısıyla ortamda belirtilen her engelin yüksekliğinin çok olduğu kabul edilmektedir. Programda ayrıca sonar algılayıcıların mesafe bilgisi doğrusal olarak hesaplanır.

Robot hareketi sırasında robotun ön tarafındaki 8 adet ses ötesi algılayıcıdan belirli aralıklarla mesafe bilgisi alır. Alınan bu veriler dosyada kaydedilerek saklanır. Oluşturulan dosyadaki sonar verilerinden bir örnek çizelge7.1’de gösterilmiştir.

Çizelge 7.1 Sonar verileri(mm)

Sonar0	Sonar1	Sonar2	Sonar3	Sonar4	Sonar5	Sonar6	Sonar7
2869	2714	2809	465	439	2348	1285	864
2873	2706	2809	465	439	2348	1285	864

Çizelge 7.1, dosyada kayıtlı olan sonar verilerini göstermektedir. Bu veriler T.O.F. ölçümünün mm cinsinden ifade edilmiş biçimleridir.

Robot üzerinde bulunan bir diğer algılayıcı da optik kodlayıcıdır. Bu optik kodlayıcı her iki tekerlekte de bulunmaktadır. Dişli çark sistemi ile kodlayıcının tekerin bir tam tur dönüşünde elde ettiği vurum sayısı artırılmıştır. Deneyde kullanılan kodlayıcı bir teker sabitken robotun 180°lik dönüşünde 76,600 adet vurum gerçekleştirir. Şaft kodlayıcısının sağladığı yüksek çözünürlük sayesinde kısa mesafelerdeki odometrik hata oranı azaltılmıştır.

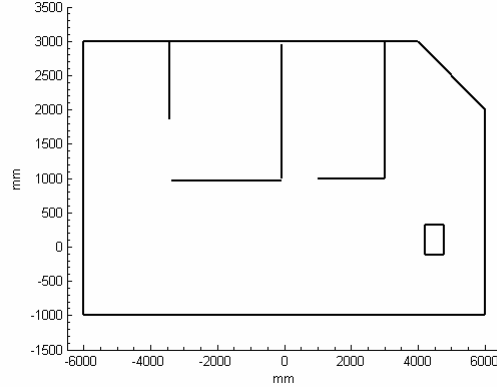
Robotun konumunun izlenmesi işleminin gerçekleştirilmesinde kodlayıcılardan yararlanılmıştır. Sistem modelinde belirtilen denetim girdileri, dönüş algılayıcı kodlayıcılardan elde edilen veriler ile tanımlanmıştır(7.1). Elde edilen denetim girdileri sayesinde robot konumunun izlenmesinin benzetimi gerçekleştirilmiştir.

Çizelge 7.2 Denetim girdileri

Doğrusal Hız (mm/sn)	Dönüş Hızı (derece/sn)	T _{samp} (msn)
130.5	14.919821	301
97.5	4.010705	295

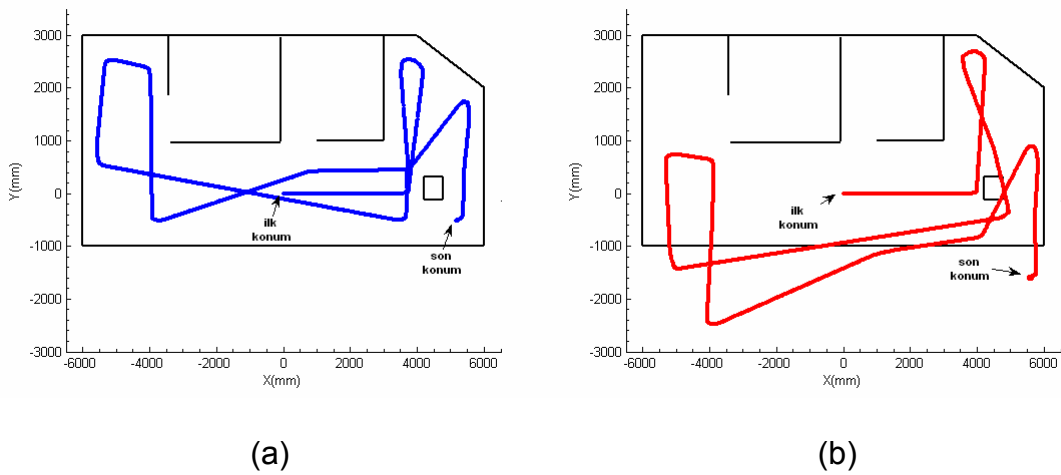
Çizelge 7.2, dosyada kayıtlı olan doğrusal hız, dönüş hızı ve elde edilen bu veriler arasında geçen süre verilerini göstermektedir. Bu veriler robottan elde edilen odometrik verilerdir. Bu odometrik veriler robot konumunun izlenmesi sırasında öngörünün yapılmasını sağlarlar.

Benzetim ortamında yapılan ilk çalışma, robot konumunun odometrik veriler kullanılarak izlenilmesidir. Öncelikle bu verilerin elde edilmesi için bir ortam yaratılmıştır. Bu ortam şekil 7.3'te gösterilmiştir.



Şekil 7.3 Robot gezinimi için ortam

Oluşturulan ortamda robotun gezinimi, Şekil 7.1'de akış diyagramı verilen program ile gerçekleştirilmiştir. Robot şekil 7.4'te gösterildiği gibi ortam içerisinde hareket etmiş ve belirli bir konumda robotun gezinimi sonlandırılmıştır. Benzetimciden elde edilen veriler sayesinde robotun ortam içerisindeki hareketi odometrik veriler kullanılarak çizilmiştir. Robot hareketlerinin izlenilmesi ve harita üzerinde gösterilmesi için Matlab programından yararlanılmıştır. İleride kullanılacak Kalman Süzgeci uygulamalarında, matris hesaplamalarının çokluğundan dolayı matris tabanlı çalışan Matlab programı tercih edilmiştir.



Şekil 7.4 (a) robotun gerçek hareketi; (b) robotun odometrik hareketi

Robot ortamda hareketine (0mm, 0mm, 0°) konumundan başlamıştır. Robot ortamda 268.527sn yol almış, toplam 40645mm mesafe gitmiştir. Robot hareketi sonucunda gerçekte (5234.6mm,-528.7mm,-6.9°) konumuna gelmiştir. Robot konumunun izlenmesi, odometrik verilerden yararlanılarak yapıldığında ise robot (5635.5mm, -1616.7mm, -2.6°) konumuna gelmiştir.

	X(mm)	y(mm)	θ (°)
Gerçek son konum	5234.6	-528.7	-6.9
Odomertik son konum	5635.5	-1616.7	-2.6

Şekil 7.4'ten da görüleceği gibi robot hareketinin izlenimi sadece odometrik verilerden yararlanılarak yapıldığında hem konumda hem de yönelimde hatalar oluşmaktadır. Gerçekleştirilen benzetimde sistem modelinde hata olmadığı varsayılmıştır. Başka bir ifade ile robotun her iki tekerleğinin çapları, genişlikleri, ve aralarındaki mesafe bilgisinde hata olmadığı varsayılmıştır. Ayrıca robot hareketi esnasında iki boyut dışına çıkmamış, tekerleklerinde herhangi bir patinaj veya kayma olmamış olarak kabul edilmiştir. Ancak odometrik hatanın oluşma nedeni, robot hareketi sırasında elde edilen verilerin kesikli olması ve dolayısıyla örnekleme zamanları arasındaki sürede robot hareketinin bilinmemesidir.

Robot hareketinin tamamı ele alınarak, konum hatası ve yönelim hatasının elde edilmesi için robot gezinişini tamamladığındaki son konum verileri kullanılmıştır. Robotun konumunu belirten (x_k, y_k, θ_k) parametreleri kullanılarak hatalar hesaplanmıştır. Konum hatasının yüzde olarak ifadesi için (7-1) eşitliği, yönelim hatasının yüzde olarak ifadesi için (7-2) eşitliği yazılabilir. Ortalama hata için konum ve yönelim hatalarının aritmetik ortalaması alınmıştır(7-3).

$$konum\ hatası = \frac{gerçek\ konum - son\ konum}{alınan\ yol} .100\% \quad (7-1)$$

$$yönelim\ hatası = \left| \frac{gerçek\ yönelim - son\ yönelim}{gerçek\ yönelim} \right| .100\% \quad (7-2)$$

$$ortalama\ hata = \frac{yönelim\ hatası + konum\ hatası}{2} \% \quad (7-3)$$

Şekil 7.4'te gösterilmiş örnek çalışmada odometriden kaynaklanan hata (7-1), (7-2) ve (7-3) eşitliklerinden yararlanılarak hesaplandığında aşağıdaki sonuçlar elde edilmiştir.

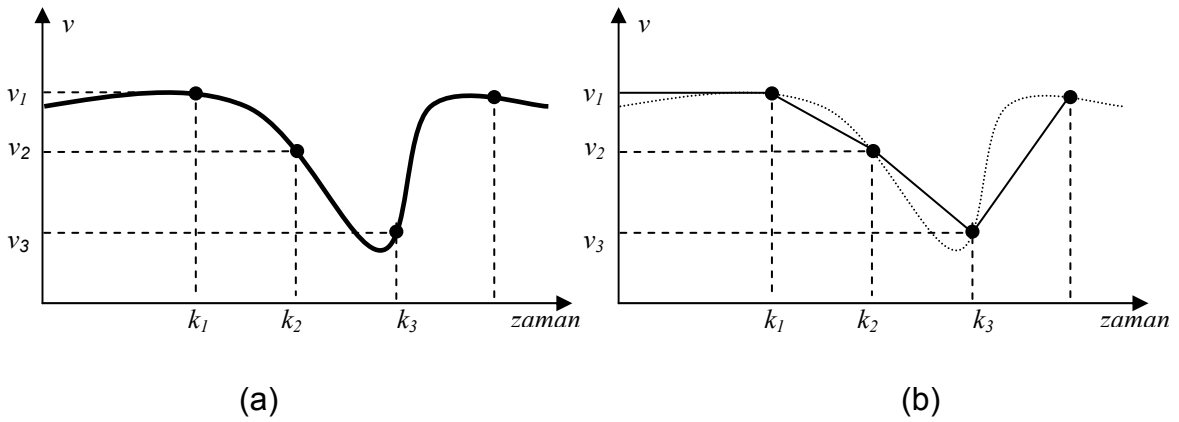
$$\text{konum hatası} = \frac{\sqrt{(5234.6 - 5635.5)^2 + (-528.7 + 1616.7)^2}}{40645} \times 100 = 2.853 \%$$

$$\text{yönelim hatası} = \left| \frac{6.9 - 2.6}{6.9} \right| \times 100 = 62.319 \%$$

$$\text{ortalama hata} = \frac{2.853 + 62.319}{2} \% = 32.586 \%$$

Bulunan hatalar, robot konumunun izlenilmesinde sadece odometrik verilerden yararlanıldığında ortaya çıkmıştır. Odometrik hatanın oluşmasındaki en büyük sorun zaman sorunudur. Literatüre gözü kapalı tahmin olarak yer almasının nedeni, örnekleme zamanı arasında robotun hareketi hakkında herhangi bir bilginin olmamasıdır. Dolayısıyla robot bir sonraki verileri elde edene kadar hareketini değiştirmeden gezinişini sürdürür.

(7-13) eşitliğinde gösterilmiş olan sistem modelinden de görüleceği gibi robotun o anlık doğrusal ve açısal hızları ile sürecin bir sonraki olabilecek durumu kestirilmeye çalışılır. Bu tez çalışmasında, (7-13)'deki sistem modeline ek olarak, robotun olası ivmesi de eklenmiştir. Bir önceki ve şimdiki zamanda elde edilen hız bilgilerinde herhangi bir değişim olduğunda başka bir ifade ile ivme sıfırdan farklı olduğunda, doğrusal hız değişimi sisteme dâhil edilmiştir.



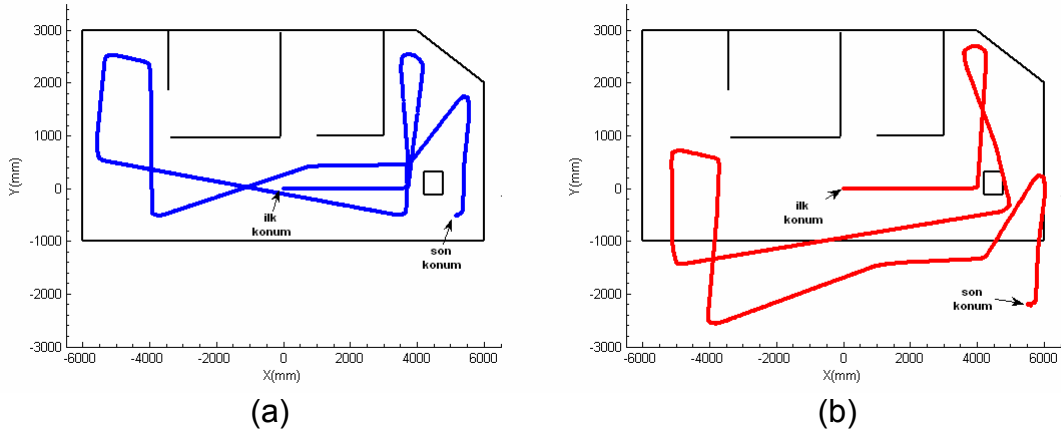
Şekil 7.5 (a) gerçek doğrusal hız; (b) varsayılmış doğrusal hız

Şekil 7.5'de robotun olabilecek gerçek doğrusal hızı (a)'da ve varsayılan hızı (b)'de gösterilmiştir. Robotun gerçek hız değişimi odometrik verilerden elde edilemez. Ancak alınan anlık hız değerleri arasındaki fark ile robotun iki zaman arasındaki değişimi doğrusal olduğu varsayılarak hesaplanabilir. Şekil 7.5.b'de k_2 ile k_3 zamanları arasındaki ivme hesabında hata yapılacağı görülmektedir. Bu hata olasılığı, örnekleme sıklığının artırılması ile azaltılabilir.

Şekil 7.5'de gösterilmiş doğrusal hız değişimi(D.H.D.) varsayımının sisteme eklenmesi için denetim girdileri yeniden tanımlanmıştır. (7-1) eşitliğinde belirtilmiş olan denetim girdileri (7-4) eşitliğindeki gibi yenilenmiştir.

$$u_k = \begin{bmatrix} u_k[\Delta D] \\ u_k[\Delta \theta] \end{bmatrix} = \begin{bmatrix} (u_k[\Delta D] + u_{k-1}[\Delta D])/2 \\ (u_k[\Delta \theta] + u_{k-1}[\Delta \theta])/2 \end{bmatrix} \quad (7-4)$$

Şekil 7-3'te tanımlanan ortamda yapılan deney, (7-4) eşitliğinde belirtilen denetim girdileri ile tekrarlandığında Şekil 7.6'de gösterilen robot hareketi elde edilmiştir.



Şekil 7.6 (a) robotun gerçek hareketi; (b) robotun odometrik hareketi

D.H.D. varsayımı ile elde edilen robot hareketi sonunda robot (5602.6mm, -2220.9mm, -6.0°) konumuna gelmiştir. Robotun bu son konumu için hata hesaplamaları aşağıdaki gibidir.

	x(mm)	y(mm)	θ (°)
Gerçek son konum	5234.6	-528.7	-6.9
D.H.D. varsayımlı son konum	5602.6	-2220.9	-6.0

$$\text{konum hatası} = \frac{\sqrt{(5234.6 - 5602.6)^2 + (-528.7 + 2220.9)^2}}{40645} \times 100 = 4.260 \%$$

$$\text{yönelim hatası} = \left| \frac{-6.9 - 6.0}{-6.9} \right| \times 100 = 13.043 \%$$

$$\text{ortalama hata} = \frac{4.260 + 13.043}{2} \% = 8.652 \%$$

Elde edilen bu sonuçlar, daha önceden D.H.D. varsayımı yapılmadan elde edilmiş sonuçlar ile karşılaştırıldığında, ortalama hata oranının azaldığı görülmüştür. Bu varsayımın sisteme eklenmesi ile daha önce yapılmış deney sonucundaki yönelim hatası büyük oranda düzeltilmiştir. Ancak konum hatası oranında çok büyük bir değişim olmamıştır. Karşılaştırmalar çizelge 7.3'te gösterilmiştir.

Çizelge 7-3 Hata oranları

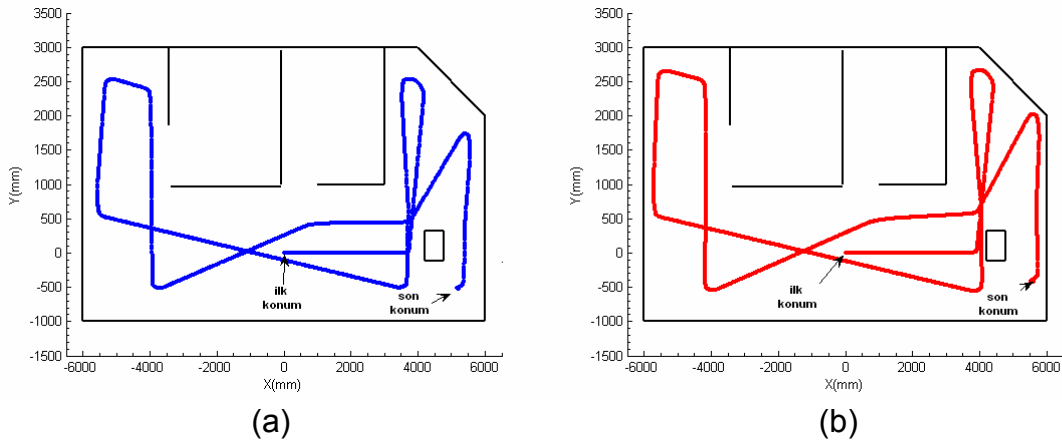
	Odometrik hata	Odometrik hata(D.H.D.)
Konum hatası(%)	2.853	4.260
Yönelim hatası(%)	62.319	13.043
Ortalama hata(%)	32.586	8.652

Elde edilen sonuçlar, robotun doğrusal hızı 200mm/sn ve örnekleme süresi yaklaşık 100msn için yapılmıştır. Odometrik hataların azaltılması için odometrik verilerin ayarı yapılabilir. Bu ayarlamalar tekerleklerden elde edilen hız bilgilerinin belirli katsayılar ile çarpılması ile yapılır.

Odometrik verilerin ayarlanması işleminde robottan elde edilen veriler ile gerçek veriler arasındaki farkın azaltılması için denetim girdilerini oluşturan sağ ve sol teker hızlarının ayarlanması yapılabilir. Ayarlama işlemi için, robotun birçok hareketinden elde edilen veriler kullanılabilir. Elde edilen verilerin gerçek veriler ile karşılaştırılmaları sonucunda belirli katsayılar elde edilir. Bu katsayıların robotun

sistem modeline dahil edilmesi ile birlikte robot sisteminin ayarlanması yapılmış olur. Ancak ayarlama değerleri sabit ve zamanla değişmeyen değerler ise odometrik hata azalmasına rağmen baki kalır. Bu ayarlama işlemlerinin ayrıntıları [27] ve [28]'de gösterilmiştir.

Şekil 7.3'te gösterilmiş ortam için ayarlanması yapılmış odometrik veri kullanıldığında ise robot hareketi şekil 7.7'deki gibi olmuştur.



Şekil 7.7 (a) robotun gerçek hareketi; (b) robotun odometrik hareketi

Ayarlanmış odometrik(A.O.) veriler kullanılarak elde edilmiş robotun son konumu (5605.9mm,-428.3mm,-6.0°) olarak bulunmuştur. A.O. veriler kullanılarak elde edilen robot konumunda oluşan hatalar (7-1), (7-2) ve (7-3) eşitliklerinden yararlanılarak hesaplanmıştır.

	x(mm)	y(mm)	θ (°)
Gerçek son konum	5234.6	-528.7	-6.9
A.O ile son konum	5605.9	-428.3	-6.0

$$\text{konum hatası} = \frac{\sqrt{(x_g - x_s)^2 + (y_g - y_s)^2}}{\text{alınan yol}} \times 100 = 0.946\%$$

$$\text{yönelim hatası} = \frac{\theta_g - \theta_s}{\theta_g} \times 100 = 13.587\%$$

$$\text{ortalama hata} = \frac{0.337 + 13.587}{2} \% = 7.266\%$$

Daha önce elde edilmiş hata sonuçlarının karşılaştırılması çizelge 7-4'te gösterilmiştir.

Çizelge 7-4 Hata oranları

	Odometrik hata	Odometrik hata(D.H.D.)	Odometrik hata(A.O.)
Konum hatası(%)	2.853	4.260	0.946
Yönelim hatası(%)	62.319	13.043	13.587
Ortalama hata(%)	32.586	8.652	7.266

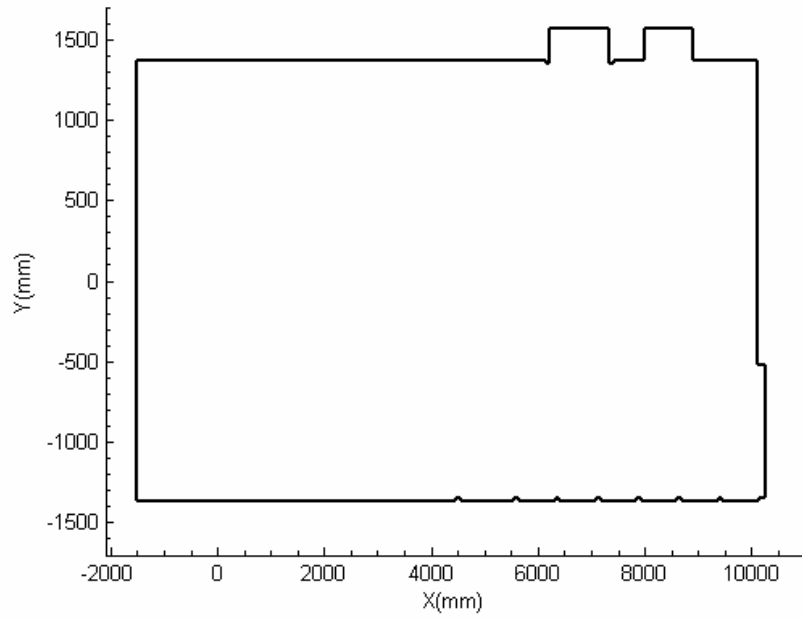
Benzetim ortamında yapılan deneyler ile robot konumunun izlenmesi sadece odometrik veriler kullanılarak gerçekleştirilmiştir. Çizelge 7-4'te A.O. veriler ile elde edilmiş hata oranları ve Şekil 7.7'de gösterilmiş robot konumunun izlenimine bakıldığında, A.O. verilerin diğer yöntemlerden daha doğru sonuçlar verdiği görülmektedir.

Sadece odometrik verilerden yararlanılarak yapılan konum izleme yönteminde hata tamamen ortadan kaldırılamaz. Robot üzerinde bulunan ve görelî ölçümü gerçekleştiren kodlayıcı veya algılayıcılar konum belirlemede yapılacak hatayı yok edemezler. Hatayı daha fazla azaltmak için mutlak ölçüm yapılmalıdır. Mutlak ölçümler hatasız bir şekilde yapılırsa robot konumunun izlenilmesinde hata yapılmamış olur. Ancak uygulamada hatasız veri elde edilmesi olanaksızdır. Bu tez çalışmasında mutlak ölçümün gerçekleştirilebilmesi için Pioneer 3-DX robotunun üzerinde bulunan 8 adet ses ötesi algılayıcıdan yararlanılmıştır. Robotun ölçüm modeli, *robot sistemi ve modeller* bölümünde ifade edildiği gibi, sonar verileri doğrultusunda modellenmiştir.

Ölçüm modelinden yararlanılarak robot konumunun izlenişi Kalman Süzgeci aracılığıyla yapılmıştır. Kalman Süzgeci kullanımında odometrik veri temel alınmış ve ölçüm modeli yardımıyla hatalar azaltılmaya çalışılmıştır. Kalman süzgeci uygulaması benzetim ve gerçek ortamdaki veriler ile gerçekleştirilmiştir. Benzetim sonuçları ile gerçek sonuçların karşılaştırılmalarının daha doğru yapılabilmesi için gerçek ortamın yaklaşık haritası oluşturulmuştur. Bu deneyde kullanılan ortam, Başkent Üniversitesi Mühendislik Fakültesi binasının bir koridorudur. Bu koridorda 2 adet kapı bulunmaktadır. Koridorun her iki tarafı da kapatılarak robotun aynı

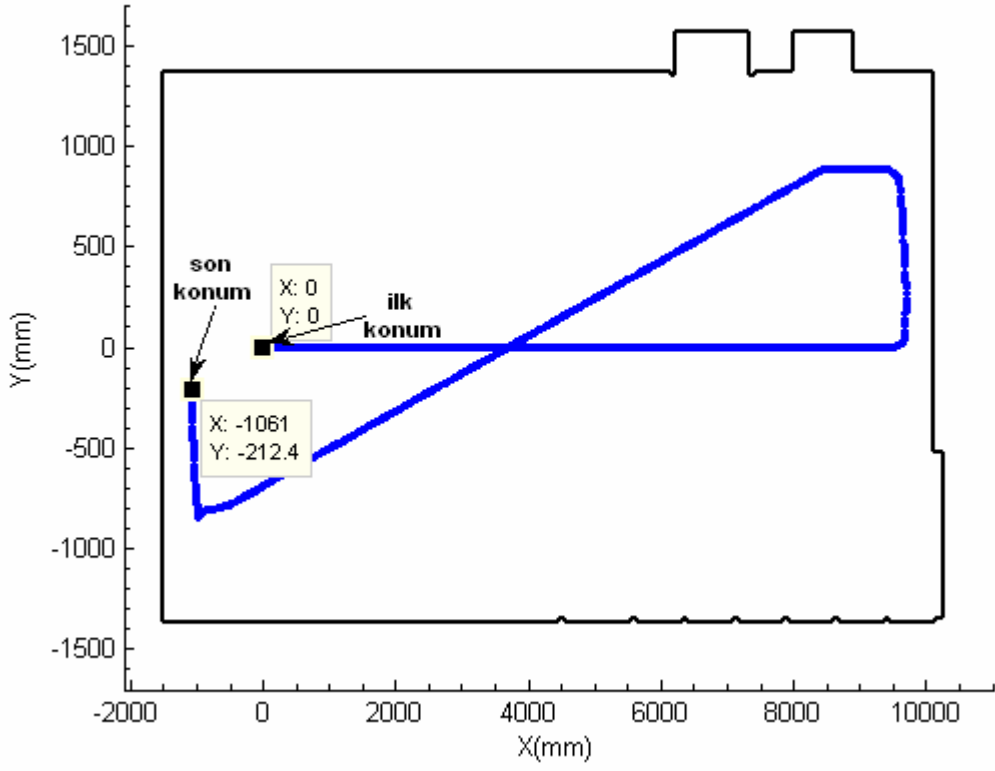
ortam içerisinde kalması sağlanmıştır. Ortamın benzetiminde oluşturulmuş yapısı şekil 7.8'de gösterilmiştir.

Öncelikle benzetimi yapılan robotun konum izlenişi yapılmıştır. Ancak burada benzetimcinin hatalı veri üretmediği göz önünde bulundurularak sisteme Kalman Süzgecinin varsaydığı sıfır ortalamalı Gauss olasılık dağılımı özelliğine sahip gürültüler rasgele olarak eklenmiştir.

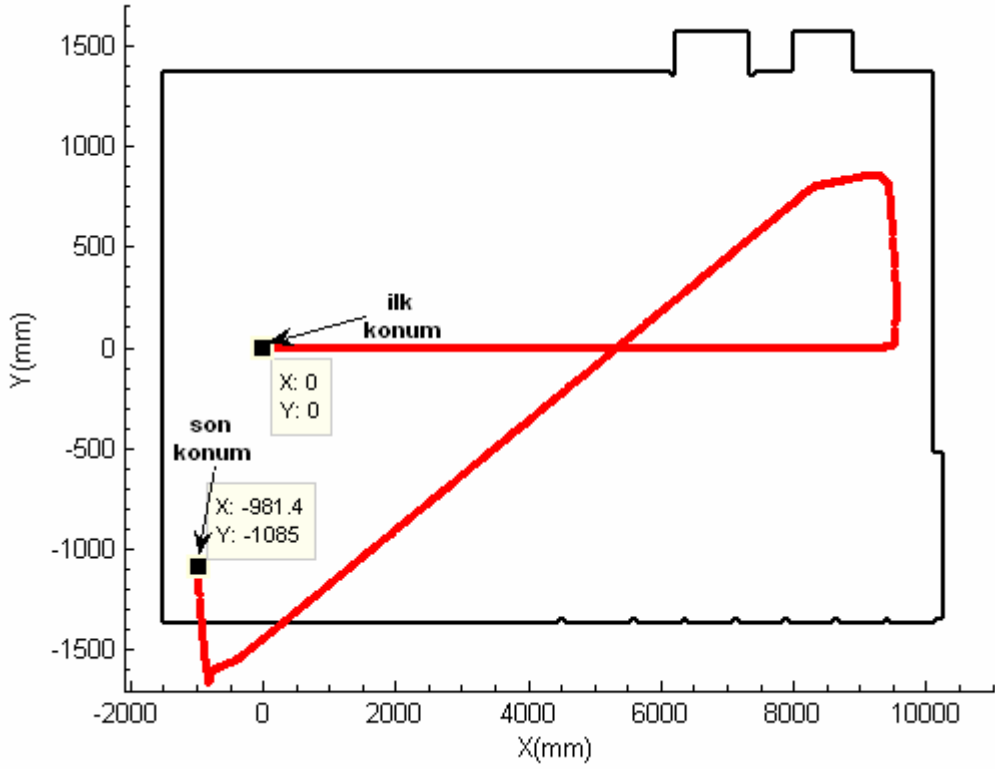


Şekil 7.8 Ortam haritası

Robotun ortam içerisindeki hareketinin benzetimi gerçekleştirilmiştir. Elde edilen veriler ile robotun esas yaptığı hareket ve sadece ayarlanmış odometrik veriler kullanılarak robot konumunun izlenimi, sırasıyla şekil 7.9.a ve şekil 7.9.b'de gösterilmiştir.



(a)



(b)

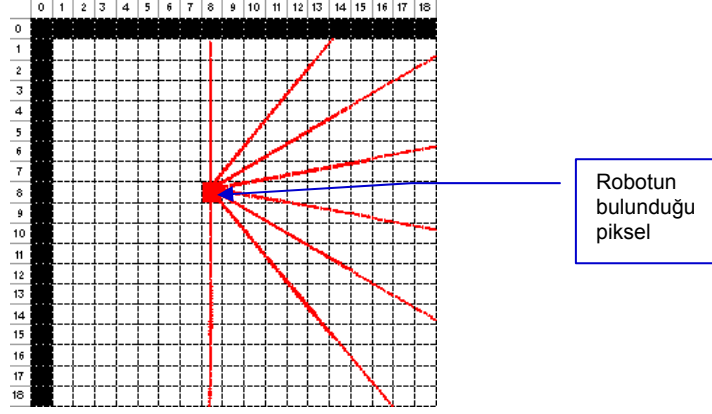
Şekil 7.9 (a) robotun ortamdaki hareketi; (b) robotun odometrik veri ile hareketi

Şekil 7.9'da beklenildiği gibi odometrik veri ile elde edilen sonuç gerçektekinden farklıdır. Robot ortamda 190.44sn süresince toplam 23663mm mesafe yol alarak hareketini tamamlamıştır. Robotun benzetim ortamındaki gerçek hareketi sonucu elde edilen robot konumu (-1060.7mm,-212.4mm,94.0°), odometrik hareketi sonucu elde edilen konum (-981.4mm,-1085.4mm,100.7°)'dir. Konum, yönelim ve ortalama hata oranları sırasıyla 3.704%, 7.224%, 5.464% olarak bulunmuştur.

	x(mm)	y(mm)	θ (°)
Gerçek son konum	-1060.7	-212.4	94.0
Odometrik son konum	-981.4	-1085.4	100.7

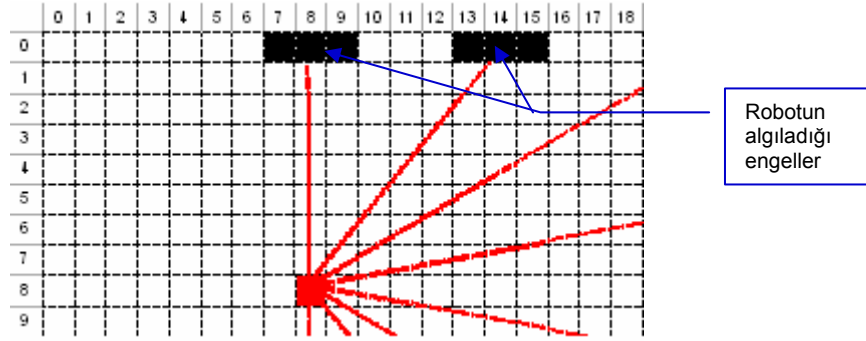
Bu hataları azaltmak için Kalman Süzgeci kullanılmıştır. Yazılan Kalman Süzgeci programının benzetim sonuçları üzerinde kullanılabilmesi için sonar verilerinden elde edilen mesafe ölçümlerindeki hataların sürece her adımda verilmesi gerekmektedir. Bu hata bilgilerinin hesaplanabilmesi için robotun öngörülen konumdaki ölçmesi gereken mesafe bilgisi ile robotun sonarlarından elde ettiği mesafe bilgisi arasındaki fark hesaplanmalıdır.

Yazılan Kalman Süzgeci programının benzetim ortamdan alınan sonuçlar üzerinde kullanılabilmesi için sonar verilerinden elde edilen mesafe ölçümlerindeki hataların hesaplanması gerekmektedir. Bu mesafe bilgilerinin hesaplanabilmesi için ortamın haritasının bilinmesi gerekmektedir. Ortamın haritası bilindiğinde, robot sisteminin, o anlık durumdaki elde etmesi gereken mesafe ölçüm değerleri hesaplanabilir. Robotun her konum bilgisi için harita üzerinden elde etmesi gereken mesafe bilgileri bulunur. Bu veriler ile robotun o anda elde ettiği veriler karşılaştırılarak, aradaki fark hata olarak hesaplanır.



Şekil 7.10 Izgara tabanlı harita tanımlamalı ortam

Şekil 7.10'da Bitmap yapısında oluşturulmuş ortam haritasından mesafe bilgisinin nasıl elde edildiği gösterilmiştir. Ortamın ızgara tabanlı haritası oluşturulmuştur. Bu haritadaki her bir piksel bölgesi gerçek ile oranlandığında 1/10 oranındadır. Kısacası bir pikselin eni veya boyu 10mm olarak belirlenmiştir.

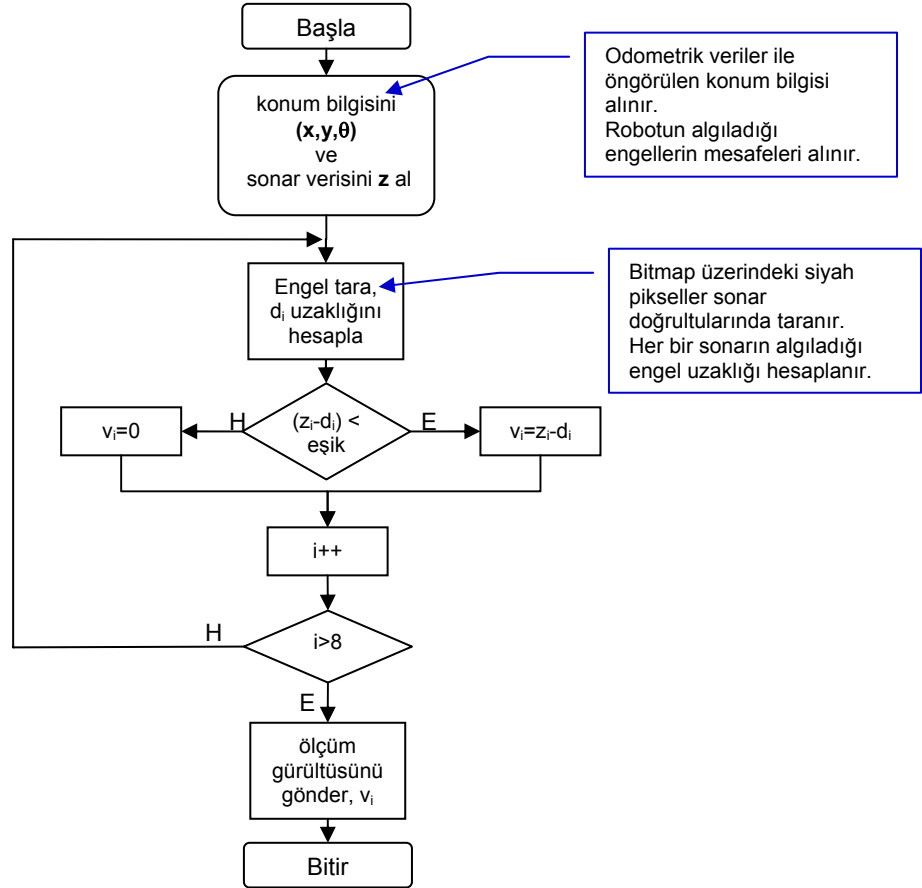


Şekil 7.11 Izgara harita tanımlamalı ortamda engel tespiti

Robotun bulunduğu konum ve yönelim bilgisi ile olası engellerin konumu şekil 7.11'de gösterildiği gibi bulunur ve robot ile engel arasındaki olması gereken mesafe bilgisi elde edilir. Dolayısıyla, *robot sistemi* ve *modeller* bölümünde şekil 9'da belirtilmiş engel ile robot arasındaki d uzaklığı her bir sonar için ayrı ayrı hesaplanmış olur. (7-42)'deki ölçüm modelinde belirtilmiş olan v_k ölçüm gürültüsü, elde edilen sonar verisinden hesaplanan d uzaklığının çıkarılması ile elde edilir.

Ölçüm gürültüsünün hesaplanabilmesi için şekil 7.12'de akış diyagramı verilmiş olan program yazılmıştır. Bu programda "*konum bilgisi*" verisi Kalman Süzgeci algoritmasının her bir döngüsünde yenilenir. Böylelikle robotun her öngörülen hareketi için elde etmesi gereken sonar verileri yaklaşık olarak

hesaplanmış olur. Hesaplanan d uzaklığı verileri, robotun öngörülen konuma geldiğinde elde etmesi gereken verilerdir. Hesaplanan d uzaklıkları ile sonarlardan alınan gerçek veriler arasındaki fark, başka bir ifade ile gürültüler, Kalman kazancını ve durum belirsizliklerini değiştirecek ve konum kestirimi gerçekleştirilecektir.



Şekil 7.12 Sonar verisindeki hata hesabı programının akış diyagramı

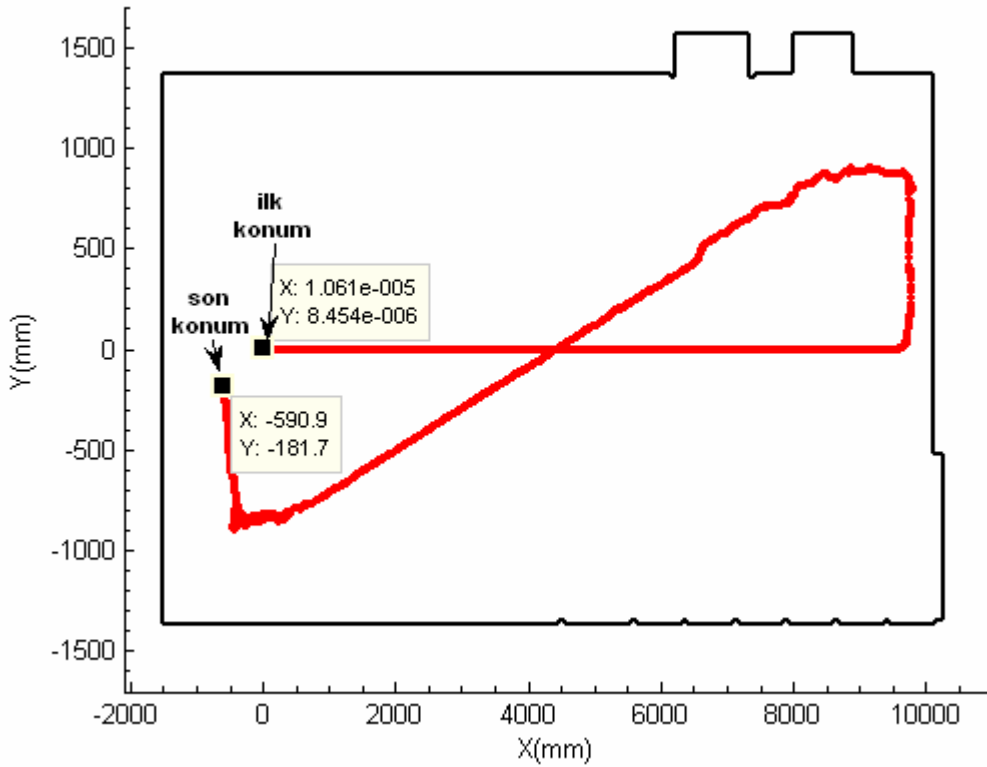
Şekil 7.12'deki akış diyagramındaki uzaklığın hesabı için (6-22) ve (6-23) denklemlerinden yararlanır. Eğer elde edilen gürültü değerleri 60mm'den fazla ise, başka bir ifade ile robotun sonralarından gelen veri fazla güvenilir ise bu durumda ölçümdeki hatanın çok yüksek olduğu varsayılır. Benzetim ortamında kullanılan eşik değeri 60mm'dir.

Gerçek ortamda yapılan deneylerde eşik değeri 500mm olarak alınmıştır. Bunun nedeni, gerçek ortamda alınan sonar verilerinin girişim, etkileşim, yansıma gibi sesin özelliklerinden etkilenmesidir.

Şekil 7.8'de gösterilmiş benzetim ortamında, robot konumunun takibi Kalman Süzgeci kullanılarak gerçekleştirilmiştir. Benzetim sonucunda robotun şekil 7.13'deki gibi bir yol izlemiş olduğu kestirilmiştir.

Şekil 7.13'de Kalman Süzgeci kullanılarak elde edilen sonuç gerçekte olması gereken sonuçlardan biraz farklıdır. Robotun gerçek hareketi sonucu elde edilen robot konumu (-1060.7mm,-212.4mm,94.0°), Kalman Süzgeci kullanılarak elde edilen konum (-590.9mm,-181.7mm,100.6°)'dir. Konum, yönelim ve ortalama hata oranları ise sırasıyla 2.411%, 7.021%, 4.716% olarak bulunmuştur.

	x(mm)	y(mm)	θ (°)
Gerçek son konum	-1060.7	-212.4	94.0
KS ile son konum	-590.9	-181.7	100.6



Şekil 7.13 Robotun KS ile kestirilmiş hareketi

Şekil 7.13'te robotun izlediği yolun, gerçekte olması gereken yolu takip etmeye çalıştığı gözlenmektedir. Kalman süzgeci ile odometrik veriler kullanılarak elde edilmiş son konum verilerindeki karşılaştırma çizelge 7.5'te verilmiştir.

Çizelge 7.5 Hata oranları

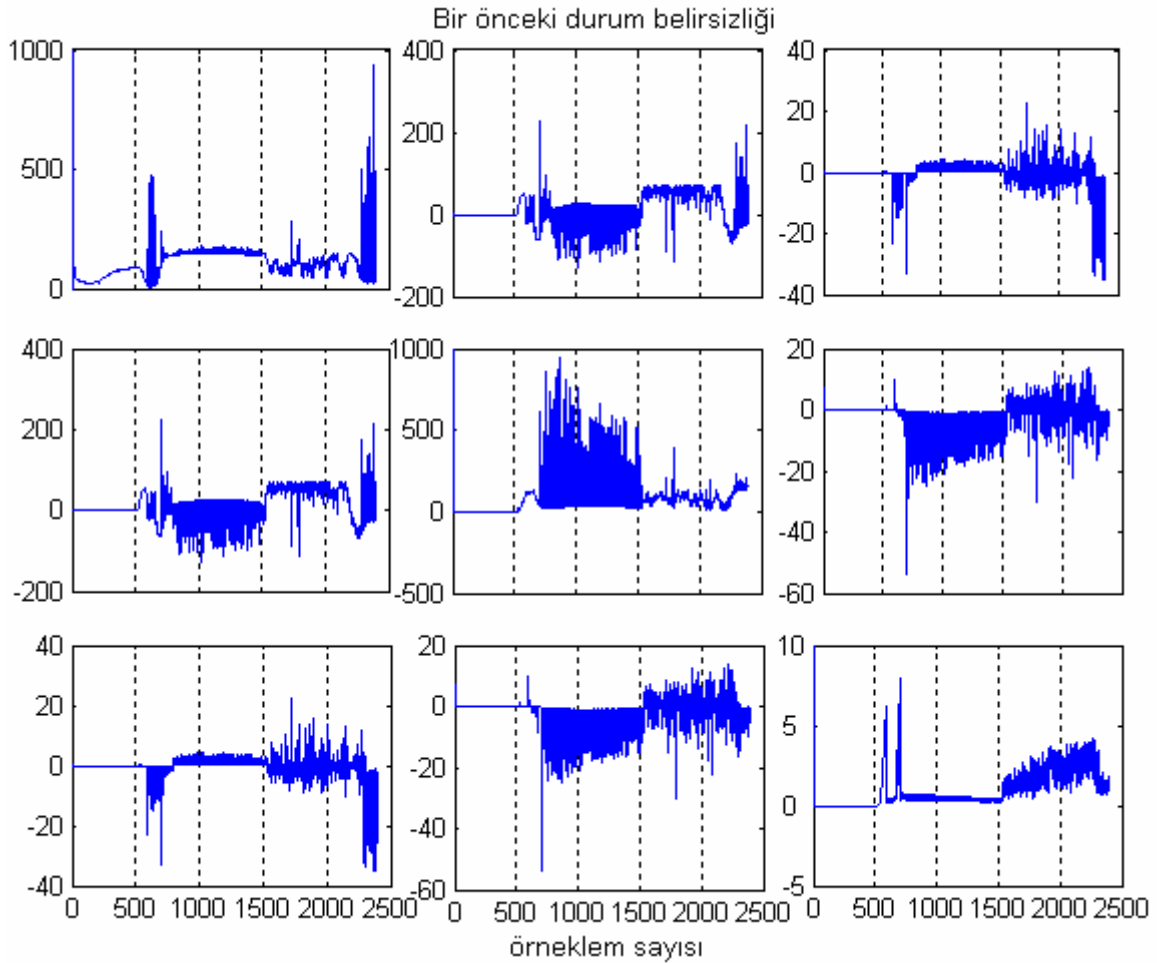
	Odometrik hata	Kalman S. hata
Konum hatası(%)	3.704	2.411
Yönelim hatası(%)	7.224	7.021
Ortalama hata(%)	5.464	4.716

Benzetim ortamında yapılan bu deney sonucunda görülmüştür ki robot konumunun izlenişinin Kalman Süzgeci kullanılarak yapılması ile son konumda oluşan hata oranı sadece odometri kullanılarak elde edilen son konumdaki hata oranından daha az olduğu görülmektedir. Çizelge 7.5'e bakıldığında hata oranlarındaki iyileşme görülmektedir. Ancak daha önemli olan iyileştirme, robot hareketinin olması gereken gerçek robot hareketine uymasındır. Robot konumunun izleniminde önemli olan robotun gerçekte yaptığı hareketlerin tümünün izleniminin gerçekleştirilmesidir.

Kullanılan Kalman Süzgecinin aşamaları sözel olarak açıklanmıştır.

Başlangıç Koşulları	(ilk durum)
\hat{x}_0^+	Başlangıç durumu belirtilir
P_0^+	Başlangıç durumunun belirsizliği verilir
ÖNGÖRÜ	(zaman güncellemesi)
$\hat{x}_k = f(x_{k-1}, u_{k-1}, w_{k-1})$	Durum kestirimi sistem dinamikleri ile güncellenir
$P_k^- = AP_{k-1}^+ A^T + Q_{k-1}$	Belirsizlik büyür
DÜZELTİM	(ölçüm güncellemesi)
$z_k = h(x_k, v_k)$	Algılayıcılardan veriler alınır
$r_k = (z_k - z_k^{nom})$	Ölçülen değer ile gerçek değer farkı bulunur
$S_k = H_k P_k^- H_k^T + R_k$	Yenilenme kovaryansı hesaplanır
$K_k = P_k^- H_k^T S_k^{-1}$	Kalman kazancı hesaplanır
$\hat{x}_k^+ = \hat{x}_k^- + K_k r_k$	Durum kestirimi için Kalman kazancı ile fark çarpılır
$P_k^+ = P_k^- - K_k S_k K_k^T$	Belirsizlik azalır

Deneyde verilen durum başlangıcı $\hat{x}_0^+ = (0,0,0)$ ve başlangıç durumunun belirsizliği $P_0^+ = 10^3.I$ olarak verilmiştir. Sisteme verilen denetim girdileri ile durum öngörüsü yapılmış ve yapılan bu durum öngörüsünün belirsizliği hesaplanmıştır. Özyinelemeli olan algoritmanın her döngüsünde durum öngörüsü ve belirsizlik hesabı yapılır. Şekil 7.14'te, elde edilen her ölçüm verisi için bir önceki durum belirsizliği gösterilmiştir.

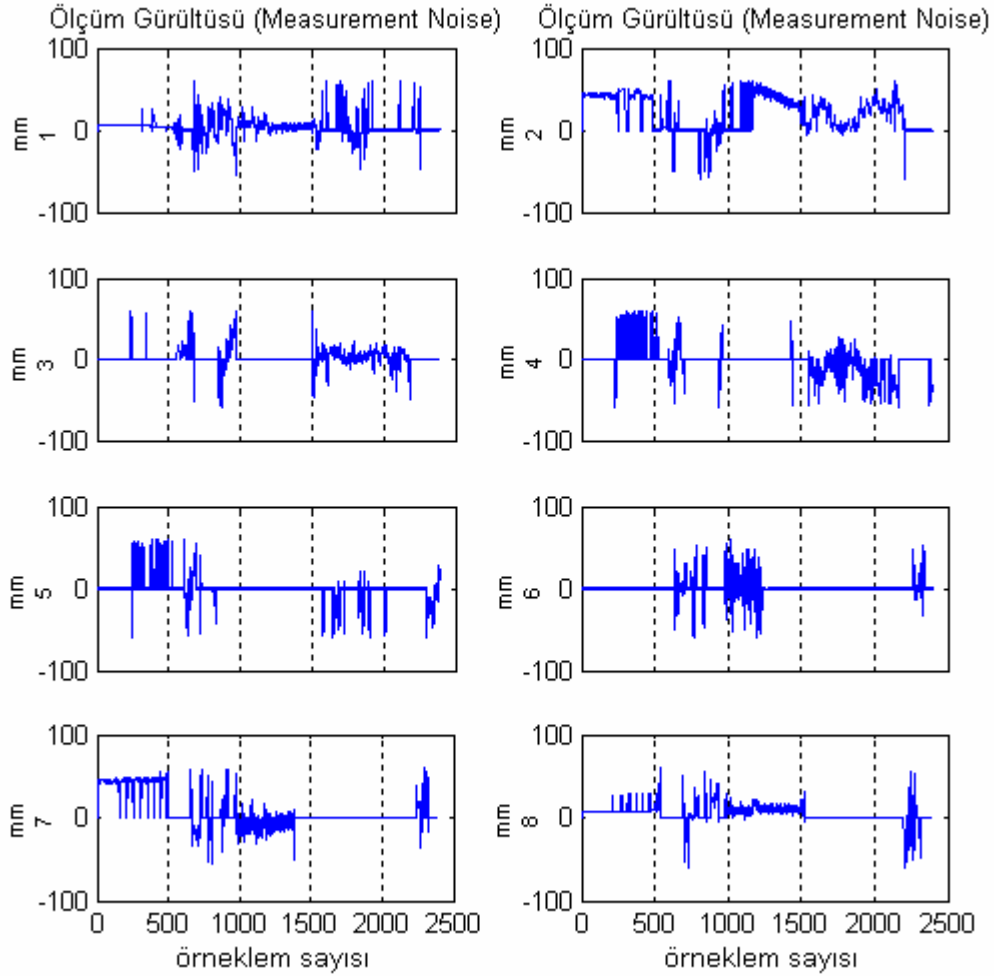


Şekil 7.14 Robotun bir önceki durum belirsizliği P_k^-

Belirsizlik matrisi 3x3 yapısında bir matristir. Köşegen üzerindeki terimler üst köşeden sırasıyla x, y, θ parametrelerinin belirsizlikleridir.

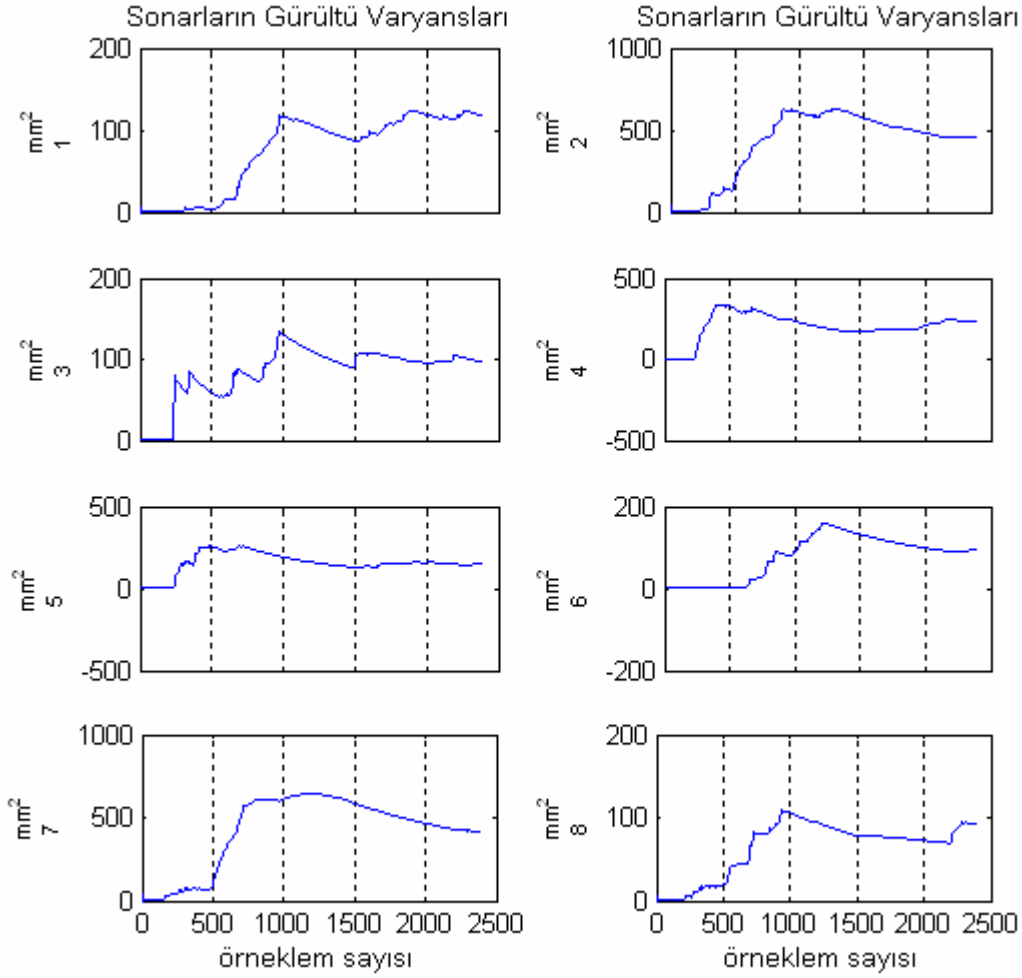
Burada θ parametresinin belirsizliğinin düşük olmasının nedeni sürekli değişmemesidir. x ve y parametrelerinin değişimine oranla θ parametresi daha uzun süre değişmeden kalır.

Kalman Süzgeci algoritmasının ölçüm güncellemesi bölümünde öngörülen konumda elde edilmesi gereken ölçüm değerleri ile robotun elde ettiği ölçüm değerleri arasındaki fark her döngüde yeniden hesaplanmıştır. r_k kalan değerleri gürültü olarak değerlendirilir. Elde edilen ölçüm gürültüleri şekil 7.15'te gösterilmiştir.



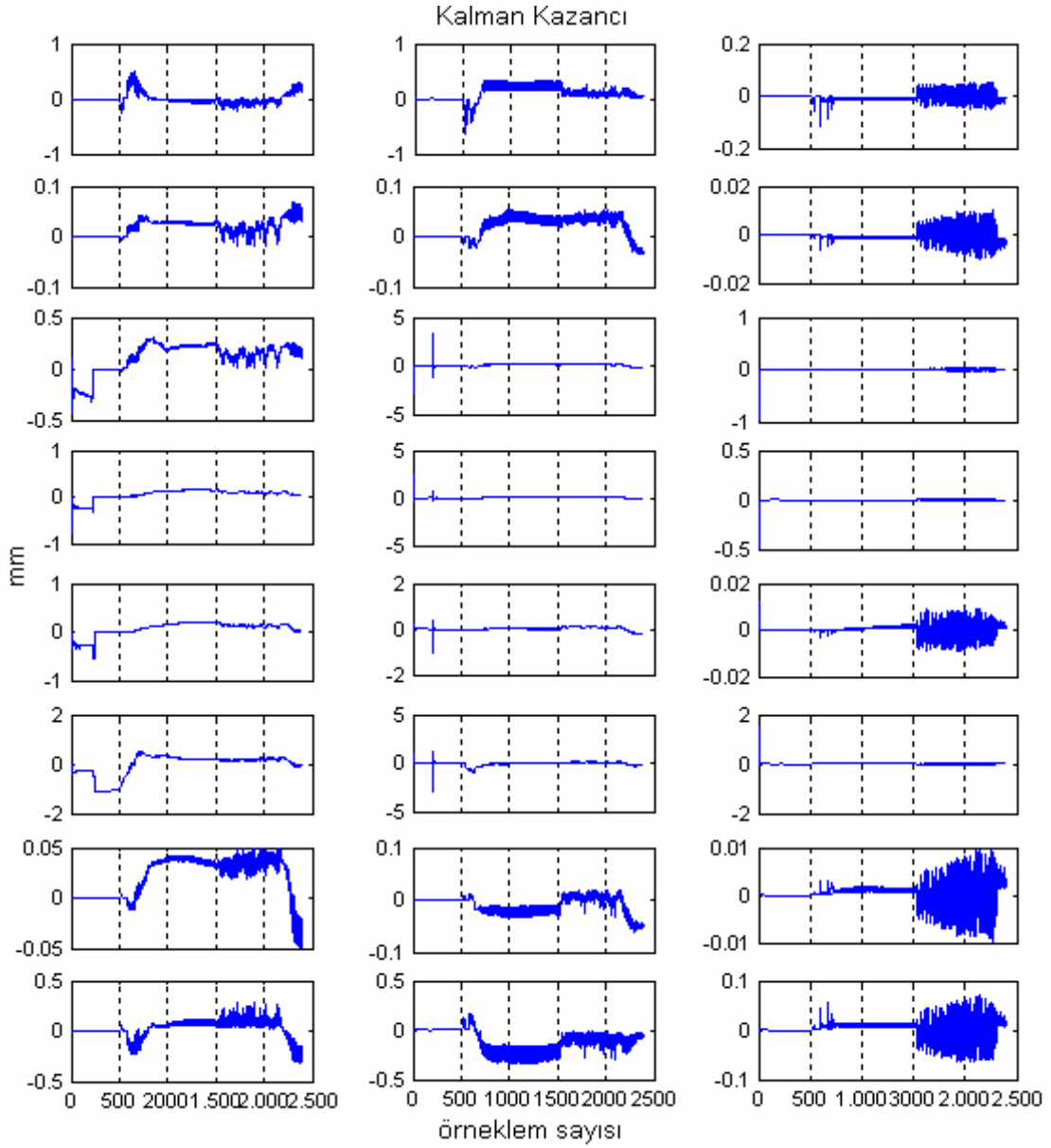
Şekil 7.15 8 adet sonardan alınan verilerle hesaplanmış gürültü değerleri

Kalan değerler Kalman kazancını değiştirecektir. Bu kalan değerlerin varyansları Kalman kazancına eklenir. Gürültülerin varyanslarının değerinin artması yani gürültüde oluşacak değişimlerin artması, durum öngörüsünün belirsizliğini ya da başka bir ifade ile güvenilirliğini azaltır. Elde edilen gürültü varyansları şekil 7.16'da gösterilmiştir.



Şekil 7.16 Sonarların gürültü varyansları R_k

Kalman Kazancı elde edilen bir önceki belirsizlik ve gürültü varyansları doğrultusunda hesaplanır. $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$ eşitliğinden yararlanılarak elde edilen Kalman kazançları her döngüde yeniden hesaplanır. Her öngörü-düzeltilim döngüsünde hesaplanmış kalman kazancının değişimi şekil 7.17’de gösterilmiştir. Kalman Kazancı K, 8x3 yapısında bir matristir. Bunun nedeni, ölçüm değerlerinin 8 farklı kaynaktan sağlanması ve durum değişkenlerinin sayısının 3 olmasıdır(x, y, θ).

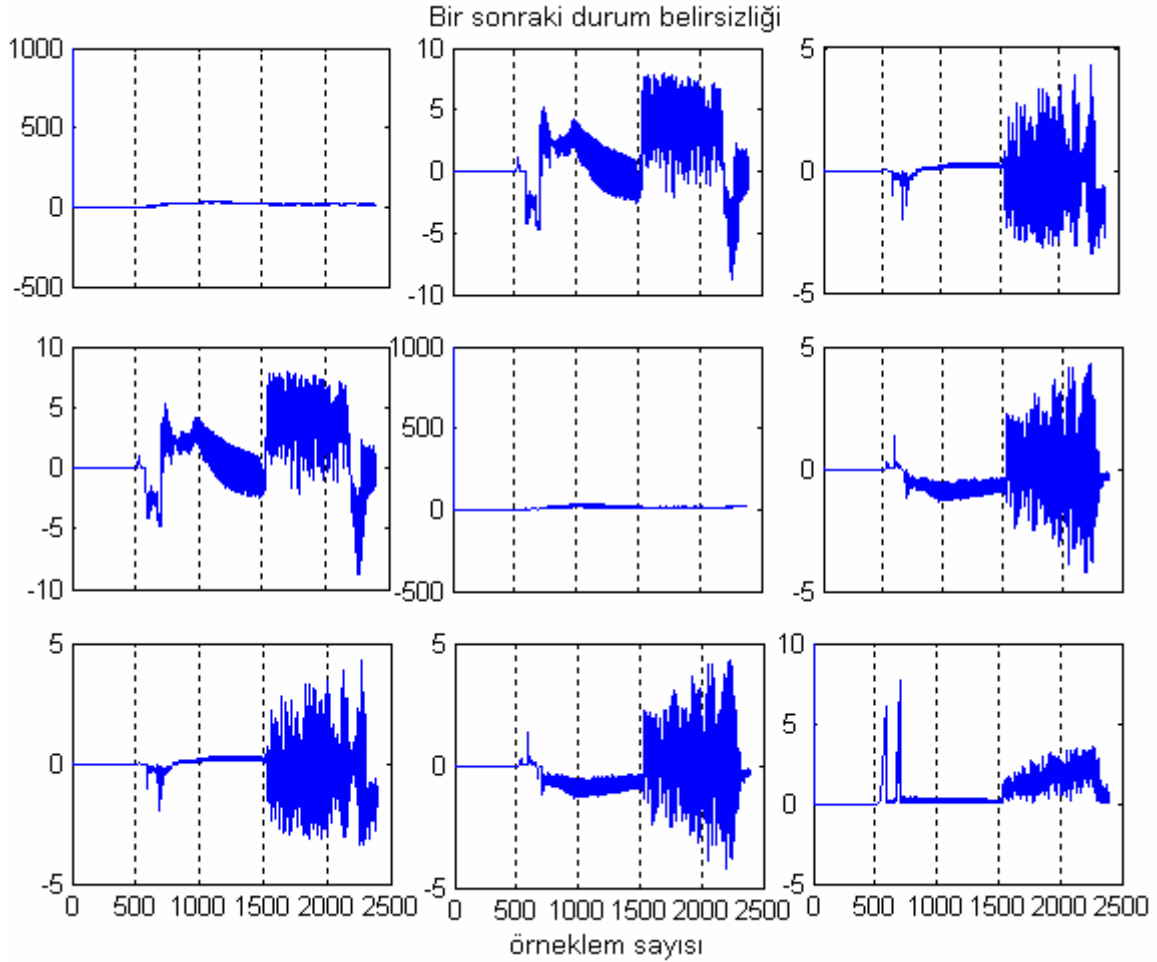


Şekil 7.17 Kalman Kazancı matrisindeki her elemanın değişimi

$$K_{2397} = \begin{bmatrix} 0.26940 & 0.03494 & -0.01793 \\ 0.05928 & -0.02877 & -0.00349 \\ 0.19610 & -0.20650 & -0.01021 \\ 0.03969 & -0.10310 & -0.00131 \\ 0.00553 & -0.1721 & -0.00037 \\ -0.1170 & -0.2693 & 0.0109 \\ -0.0473 & -0.05044 & 0.003691 \\ -0.2938 & -0.06043 & 0.01982 \end{bmatrix}$$

Elde edilen en son Kalman kazancı yukarıda verilmiştir. 2397 örnek sayıdır.

Kalman Kazancı elde edildikten sonra öngörülen durumun düzeltimi gerçekleştirilir. Daha sonra bir zaman sonra kullanılacak olan bir önceki belirsizlik kovaryansı P_k^+ hesaplanır. Elde edilen bir sonraki belirsizlik kovaryans matrislerinin her bir örneklemede nasıl değiştikleri şekil 7.18'de gösterilmiştir.



Şekil 7.18 Robotun bir sonraki durum belirsizliği P_k^+

Robotun bir sonraki durum belirsizliği ile bir önceki durum belirsizlikleri karşılaştırıldığında, bir sonraki durum belirsizlik kovaryans matrisinin her örneklemede aldığı değerlerin daha küçük olduğu görülmektedir. Bunun anlamı robotun öngörülen konumundaki düzeltme işlemi başarı ile gerçekleşmiştir. Başka bir ifade ile düzeltilmiş robot konumunun gerçek robot konumuna eşit olma olasılığı artmıştır.

7.2. Gerçek Ortam

Kalman süzgeci algoritması ile benzetim ortamında gerçekleştirilen robot konumunun izlenimi gerçek ortamda da tekrarlanmıştır. Deney, robotun doğrusal hızı en yüksek 200mm/sn ve örnekleme süresi yaklaşık 320msn için yapılmıştır. Doğrusal hız, engelin 30cm mesafede algılandığı anda 0 olacak şekilde ayarlanmıştır. Örnekleme, 8 tane sonarın mesafe bilgisini elde etmesinden sonra yapılmıştır. Her bir sonar yaklaşık 40ms'de mesafe bilgisini elde eder.

Robotun ortam içerisinde birçok gezinimi gerçekleştirilmiştir. Bu gezinimlerden bir tanesi bu tez çalışmasında ayrıntısı ile sunulmuştur. Yapılan gezinim süresince robotun hiçbir hareketi kısıtlanmamıştır. Robot gezinimi Şekil 7.1'de akış diyagramı verilmiş olan robot hareketi programı ile gerçekleştirilmiştir. Robot ortamdaki gezinişini 5440310msn'de başka bir ifade ile 9 dakika 4.0310sn'de tamamlamıştır. Bu süre sonunda toplam 103937mm başka bir ifade ile 103.937 metre yol almıştır. Gezinim sonunda toplam 1699 adet örneklem alınmıştır. Robotun elde edilen son konumu (5100mm,-460mm,-169°) olarak ölçülmüştür.

Gerçek ortamda robottan elde edilen veriler ile robot izlenimi sadece odometrik veriler kullanılarak yapılmıştır. Sadece odometrik veriler kullanılarak yapılan konum izlenimi şekil 7.19'da gösterilmiştir. Şekil 7.19'da odometrik veriler sonucu elde edilmiş son konum (2922.521mm,-1112,892mm,-119.857°) olarak bulunmuştur.

	x(mm)	y(mm)	θ (°)
Gerçek son konum	5100	-460	-169
Odometrik son konum	2922.521	-1112,892	-119.857

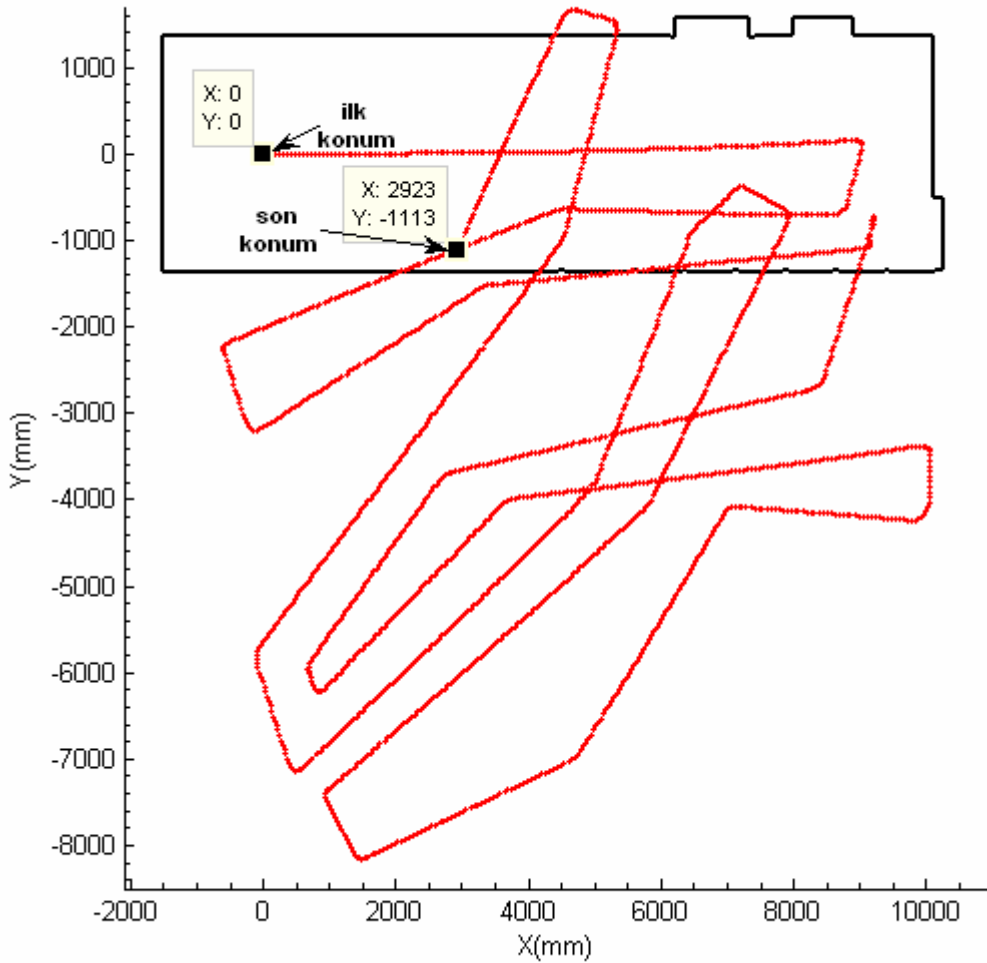
Odometrik veriler kullanılarak elde edilen robotun son konumunda oluşan hatalar (7-1), (7-2) ve (7-3) eşitliklerinden yararlanılarak hesaplanmıştır.

$$\text{konum hatası} = \frac{\sqrt{(5100 - 2922.5)^2 + (-460 + 1112.9)^2}}{103937} \times 100 = 2.187 \%$$

$$\text{yönelim hatası} = \left| \frac{-169 + 119.857}{-169} \right| \times 100 = 29.078 \%$$

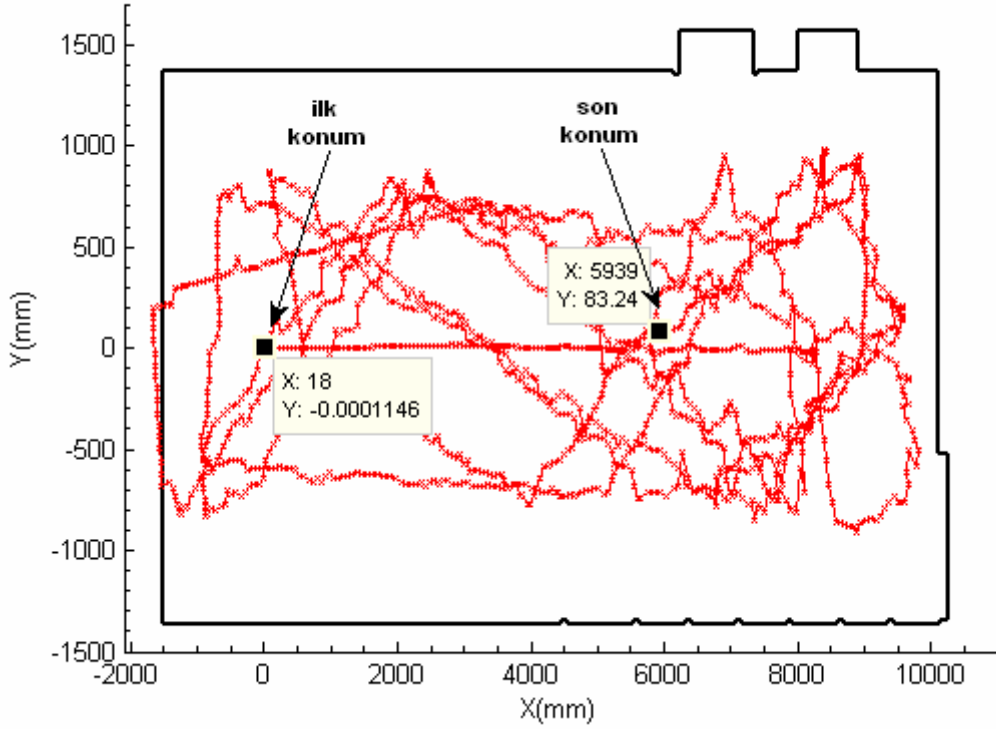
$$\text{ortalama hata} = \frac{2.187 + 29.078}{2} \% = 15.632 \%$$

Yapılan hata oranları, robotun hareketi süresince değişikliğe uğrayacaktır. Özellikle robotun varsayılan konumunun ortam dışında olduğu durumlarda bu hata oranları büyüyecektir. Şekil 7.19'a bakıldığında robot konumunun sadece odometrik veriler kullanılarak yapılmasının mümkün olamayacağı görülmektedir. Dolayısı ile hata oranlarının sadece son konuma bakılarak yapılması bütün gezinim esnasında aynı olmayacaktır.



Şekil 7.19 Robotun odometrik hareketi

Sadece odometrik veriler ile gerçekleştirilen robot konumunun izlenimi Kalman süzgeci kullanılarak gerçekleştirilmiştir. Kalman Süzgecinin durum başlangıcı $\hat{x}_0^+ = (0,0,0)$ ve başlangıç durumunun belirsizliği $P_0^+ = 10^3.I$ olarak verilmiştir. Şekil 7.20'de robot konumunun Kalman Süzgeci kullanılarak yapılan izlenimi gösterilmiştir.



Şekil 7.20 Robotun KS ile hareketi

Şekil 7.20'de Kalman Süzgeci kullanılarak elde edilen sonuç gerçekte olması gereken sonuçlardan biraz farklıdır. Kalman Süzgeci kullanılarak elde edilen konum (5938.725mm, 83.235mm, -167.325°)'dir.

	x(mm)	y(mm)	θ (°)
Gerçek son konum	5100	-460	-169
KS ile son konum	5938.725	83.235	-167.3

$$\text{konum hatası} = \frac{\sqrt{(5100 - 5938.7)^2 + (-460 - 83.2)^2}}{103937} \times 100 = 0.961 \%$$

$$\text{yönelim hatası} = \left| \frac{-169 + 167.3}{-169} \right| \times 100 = 0.991 \%$$

$$\text{ortalama hata} = \frac{0.961 + 0.991}{2} \% = 0.976 \%$$

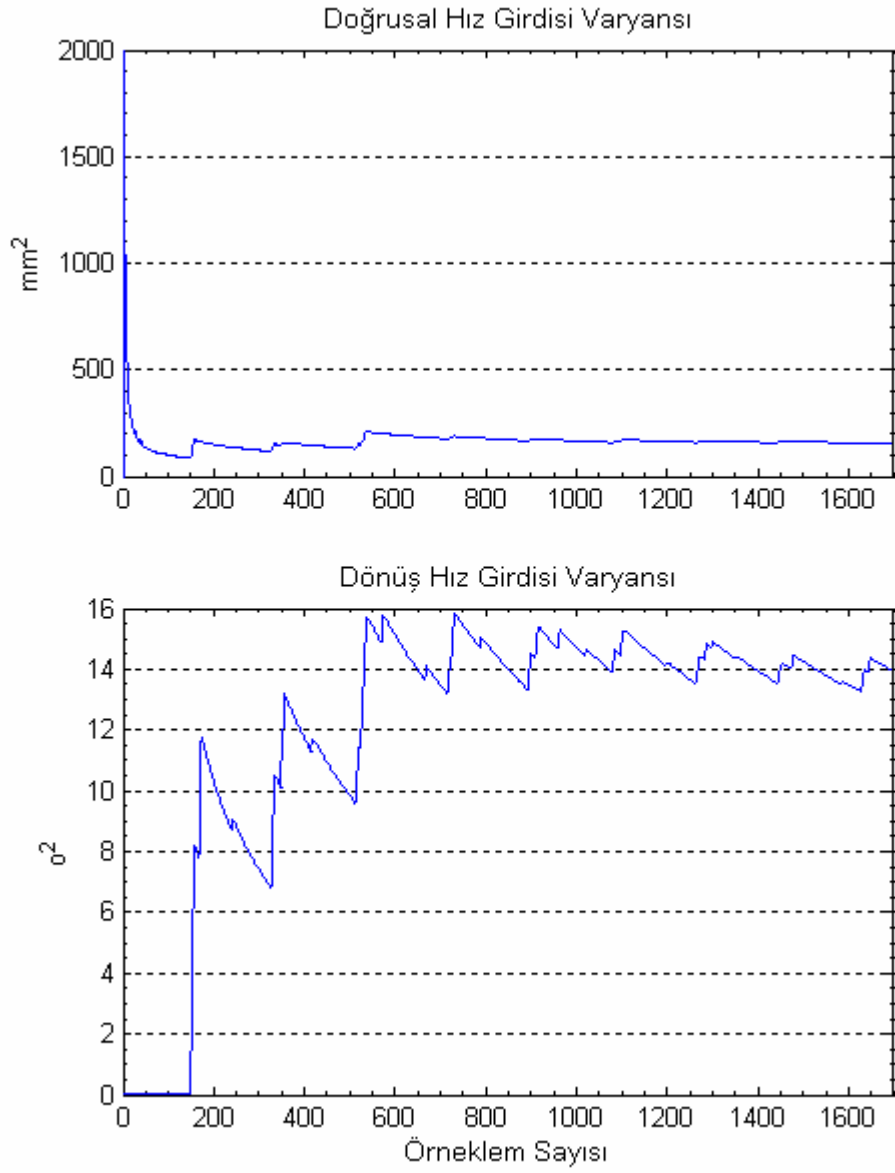
Konum, yönelim ve ortalama hata oranları ise sırasıyla 0.961%, 0.991%, 0.976% olarak bulunmuştur. Bulunan hata oranları odometri ile elde edilen hata oranları ile karşılaştırıldığında Kalman Süzgecinin daha az hatalı sonuçlar verdiği çizelge 7.6'da gösterilmiştir.

Çizelge 7.6 Hata oranları

	Odometrik hata	GKS hata
Konum hatası(%)	2.187	0.961
Yönelim hatası(%)	29.078	0.991
Ortalama hata(%)	15.632	0.976

Çizelge 7.6'da gösterilmiş hata oranları robotun hareketini tamamladığı andaki son konum için hesaplanmıştır. Şekil 7.19 ve Şekil 7.20'deki robot hareketleri incelendiğinde Kalman Süzgecinin robot konumunun izlenilmesinde çok iyi sonuçlar verdiği görülmektedir.

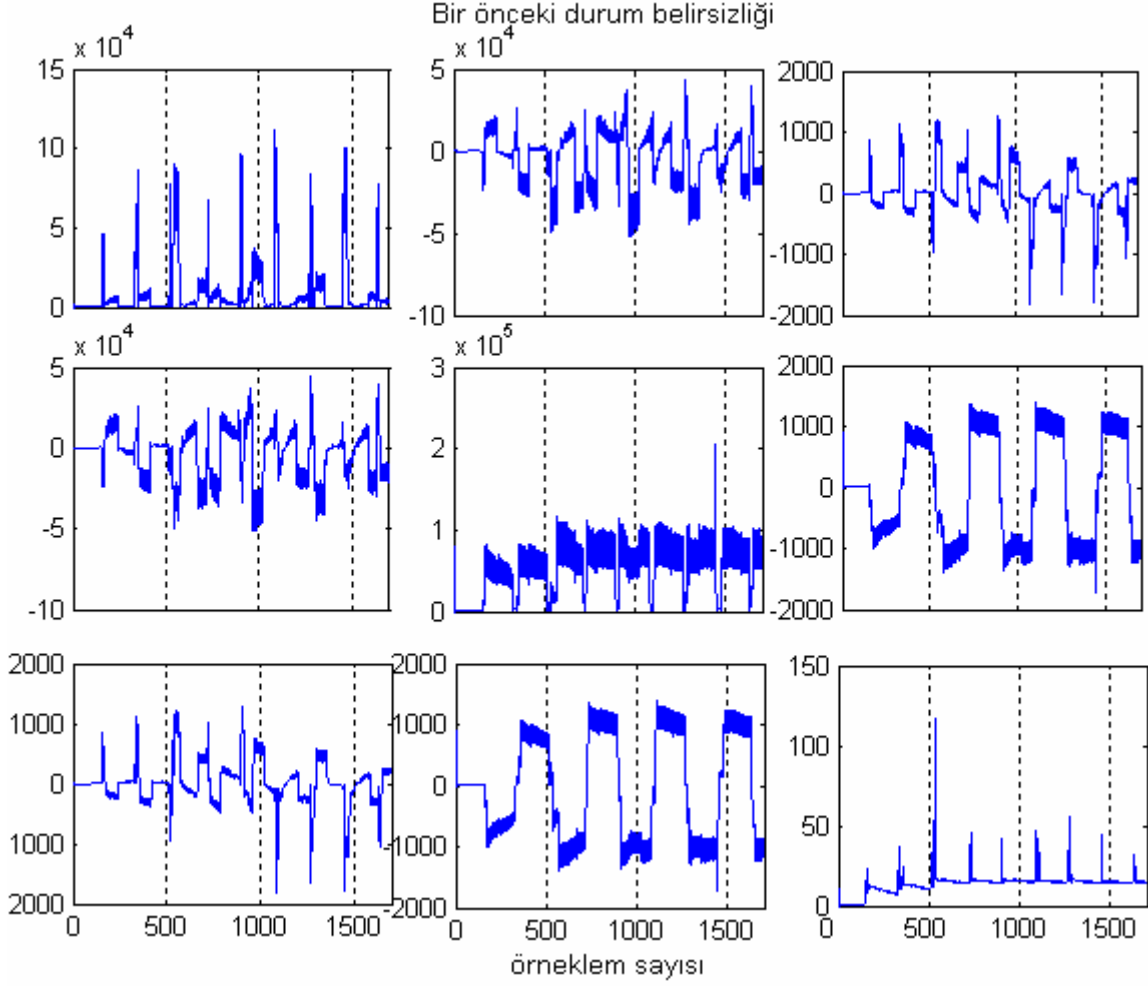
Robotun gerçek ortamdan elde edilen veriler ile yapılan konum belirleme işlemi Kalman Süzgeci kullanılarak yapılmıştır. Bu işlem sırasında her örnekleme elde edilmiş GKS parametrelerinden denetim girdilerinin varyansları şekil 7.21'de gösterilmiştir.



Şekil 7.21 Robotun denetim girdilerinin varyansları

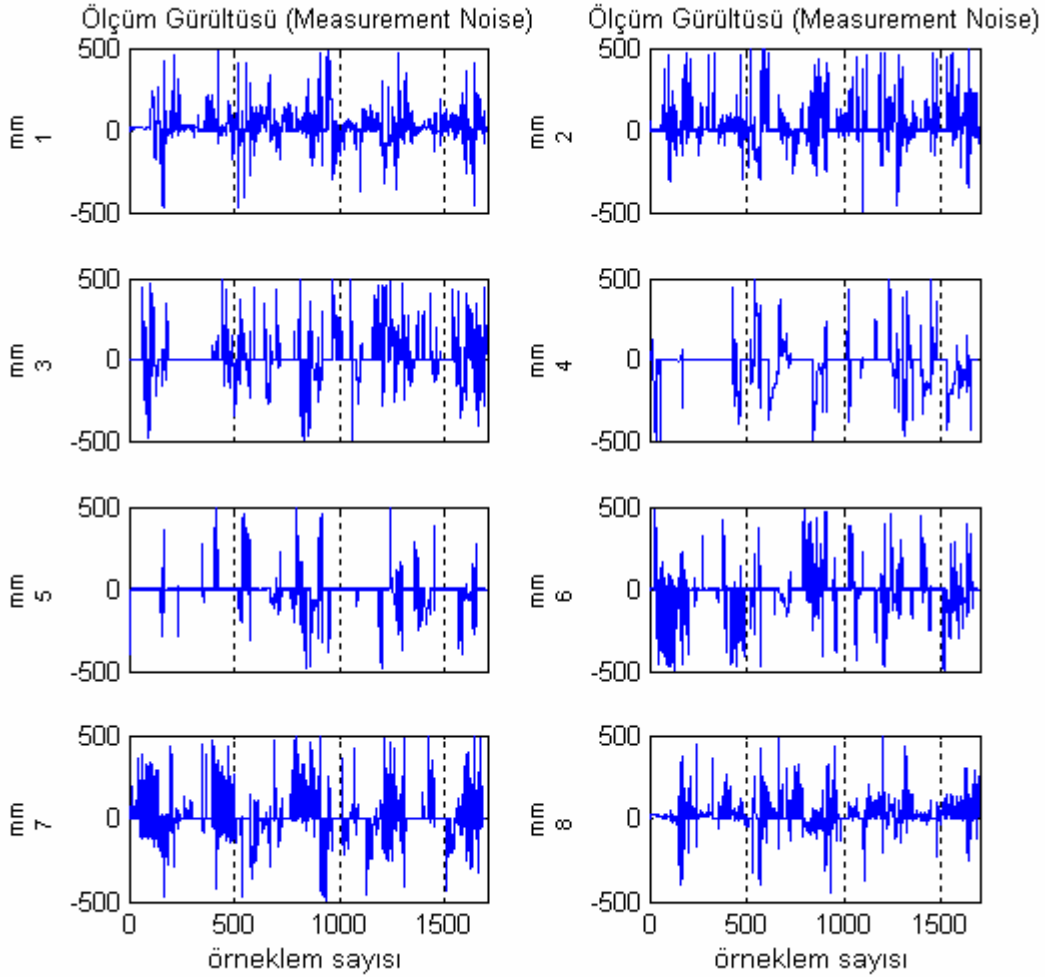
Şekil 7.21’de gösterilmiş varyanslar GKS uygulamalarında sabit değerler olarak alınır. Ancak bu uygulamada zaman sorunu olmadığı için denetim girdileri varyansları her örnekleme yeniden hesaplanmıştır.

Özyinelemeli olan GKS algoritmasının her döngüsünde durum öngörüsü ve belirsizlik hesabı yapılmıştır. Şekil 7.22’de, bir önceki durum belirsizliği her örneklem anı için gösterilmiştir.



Şekil 7.22 Robotun bir önceki durum belirsizliği P_k^-

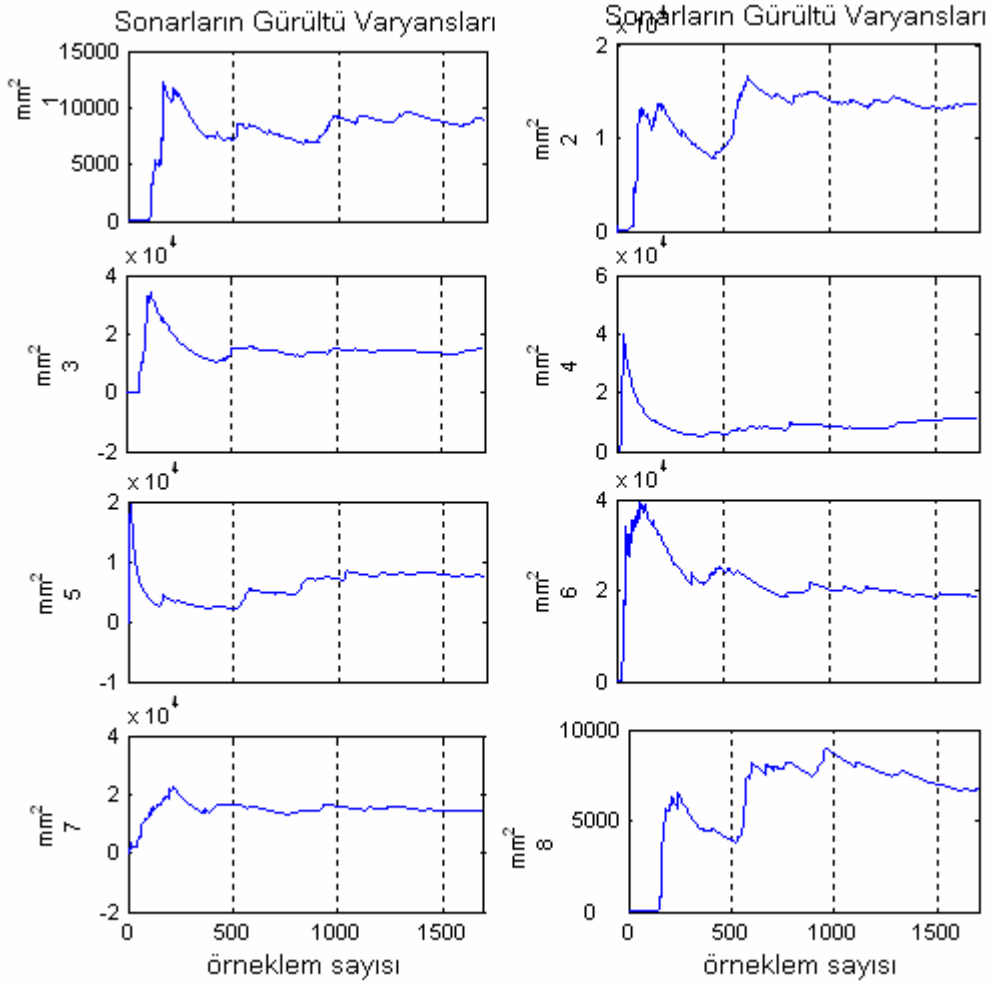
GKS algoritmasının ölçüm güncellemesi bölümünde öngörülen konumda elde edilmesi gereken ölçüm değerleri ile robotun elde ettiği ölçüm değerleri arasındaki fark her döngüde yeniden hesaplanmıştır. r_k kalan yani gürültü değerleri şekil 7.23'te gösterilmiştir.



Şekil 7.23 8 adet sonardan alınan verilerle hesaplanmış gürültü değerleri

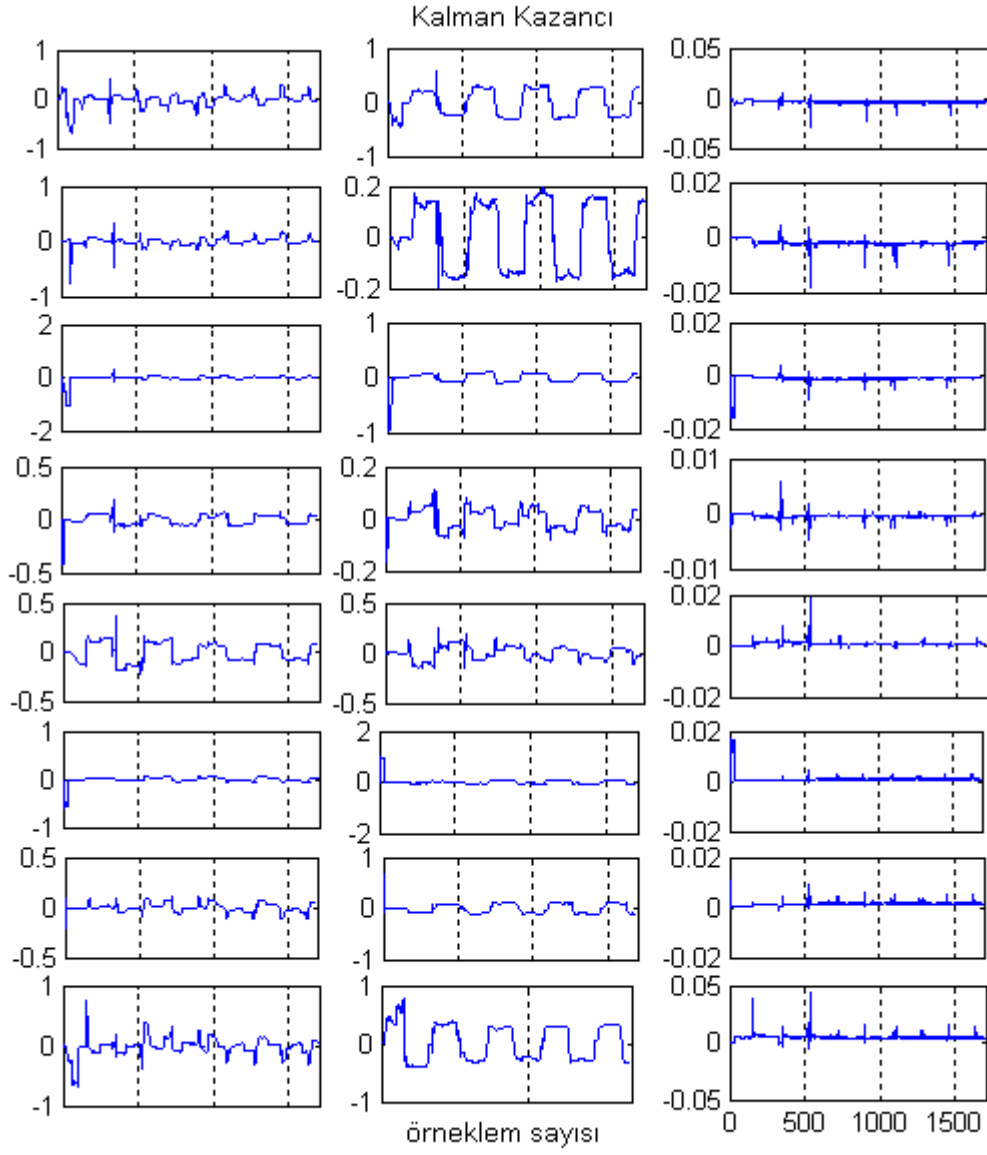
Şekil 7.23'ten de anlaşılacağı gibi gerçek ortamda yapılan deneyde sonarlardan elde edilen değerler ile gerçek alınması gereken değerler arasındaki fark 500mm'den fazla ise kabul edilmemektedir. Bu değer sınırlandırılması yapılmazsa Kalman kazancı çok fazla değişecek ve sistemin gürültü doğrultusunda kestirimini gerçekleştirecektir.

Şekil 7.23'te gösterilmiş olan gürültülerin varyansları şekil 7.24'te gösterilmiştir. Bu değerler GKS algoritmasındaki R matrisini tanımlarlar.



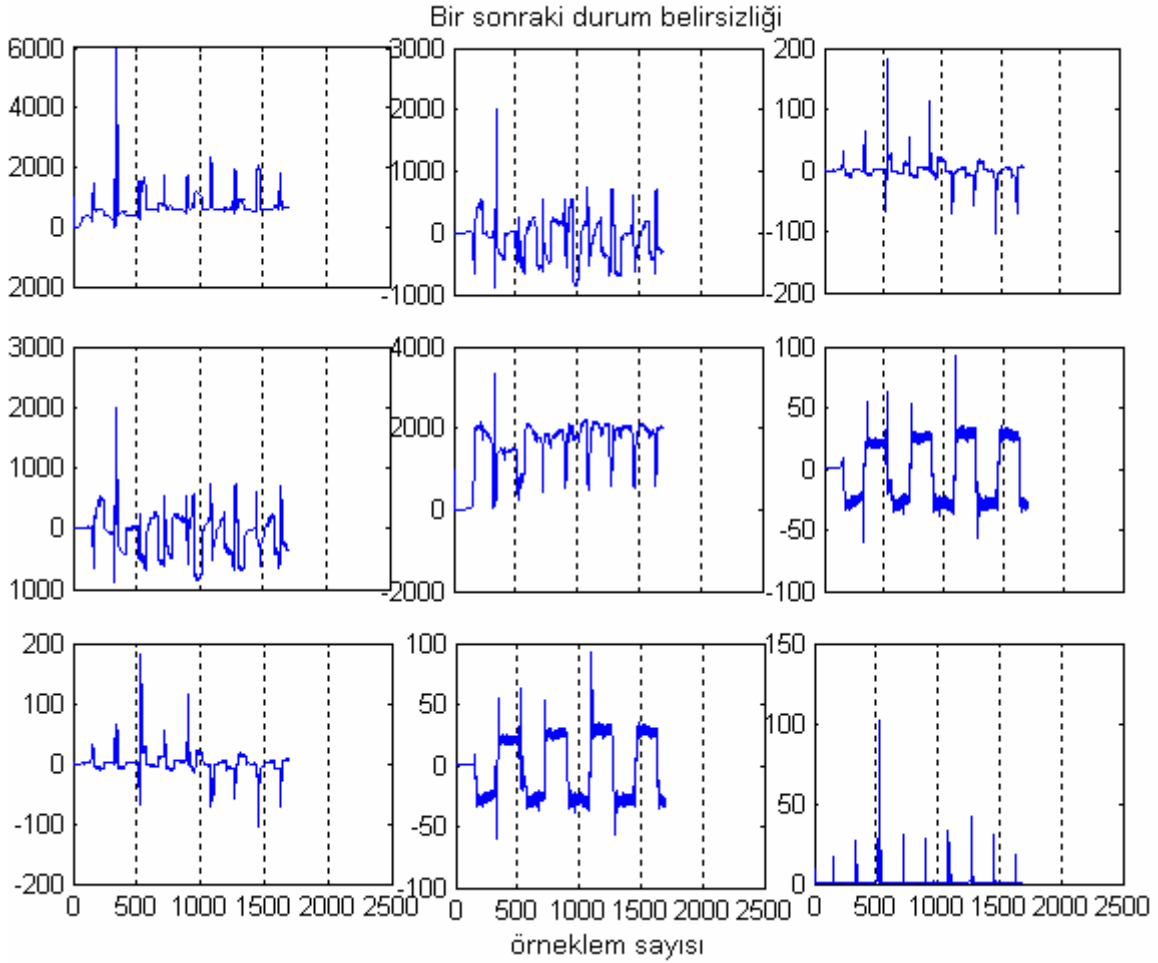
Şekil 7.24 Sonarların gürültü varyansları R_k

Her öngörü-düzeltilim döngüsünde hesaplanmış kalman kazancının değişimi
şekil 7.25'de gösterilmiştir.



Şekil 7.25 GKS algoritmasındaki Kalman kazanç matrisi

GKS algoritmasında Kalman kazancının bulunması ile bir sonraki durum belirsizliği elde edilir. Bu değerler de Şekil 7.26'de gösterilmiştir.



Şekil 7.26 GKS algoritmasındaki bir sonraki durum belirsizliği P_k^+

Bir önceki durum belirsizliği ile karşılaştırıldığında, bir sonraki durum belirsizliğini ifade eden kovaryans matrisinin değerlerinin az olduğu görülmüştür. Dolayısı ile odometrik verilerden alınarak yapılan konum öngörüsündeki belirsizlik sonarlar ile yapılan mutlak ölçüm ile düzeltilmiştir. Ancak burada tam anlamı ile düzeltim yapılabilmesi olanaksızdır. Bunun nedeni kullanılan sonarların elde edebileceği hatalı mesafe ölçümleridir.

8. SONUÇ

Gezgin bir robotun konumunun belirlenmesi, robota işlevsellik kazandırılması için gereklidir. Bu nedenle, robotun ortam içerisindeki hareketi takip edilerek robotun olması gereken konumu belirlenmelidir. Bu amaç doğrultusunda en basit yöntem olan odometrik veriler ile robot konumunun takibi yapılmış ve sadece odometrik veriler kullanılarak robot konumunun takibinin mümkün olamayacağı görülmüştür. Robotun konumunun daha iyi bir şekilde belirlenebilmesi için odometrik veriler ile sonar verileri birleştirilerek daha iyi sonuçlar elde edilmiştir.

Sonar algılayıcılardan gelen veriler genişletilmiş Kalman süzgecinde, robot konumunun düzeltiminde kullanılmıştır. Sonar algılayıcılardan gelen mesafe bilgilerinin çok hassas olmamasına rağmen odometrik algılayıcılardan elde edilen veriler ile yapılan konum öngörüsünün düzeltiminde olumlu etkiler yaptığı görülmüştür. Genişletilmiş Kalman süzgeci kullanımı ile robotun bütün hareketi izlenmiş ancak özyinelemeli özelliği dolayısı ile işlem sürecinde geçmiş bütün veriler kullanılmamıştır. Kalman süzgeci geçmiş verileri, varyans ve kovaryans hesaplamaları ile sisteme dahil etmiştir.

DeneySEL sonuçlar, benzetim ve gerçek ortamlarda değerlendirilmiştir. Şekil 7.9 ve şekil 7.13'de benzetim ortamındaki deney sonuçları gösterilmiştir. Bu sonuçlarda GKS kullanarak yapılan konum belirleme işleminin daha başarılı sonuçlar verdiği görülmektedir.

Gerçek ortamda yapılan deneylerde, sonar verilerinin Gauss özellik göstermemelerine rağmen, Kalman süzgecinin en iyi şekilde konum kestirimini yaptığı görülmüştür. Şekil 7.19 ve Şekil 7.20'de gerçek ortamdan alınan veriler ile elde edilen deney sonuçları gösterilmiştir. Gerçek ortamda yapılmış olan deney sonucunda, sadece odometrik veriler kullanılarak yapılan konum belirleme işleminde elde edilen son konumdaki ortalama hata %15.632 iken genişletilmiş Kalman süzgeci kullanıldığında konumdaki ortalama hata %0.976 değerine düşmüştür. Ancak GKS'nin etkisinin robotun son konumuna göre değerlendirilmesinin yeterli olmayacağı görülmüştür. GKS robotun bütün hareketini adım adım öngörmüş ve elde ettiği gözlem verileri ile öngördüğü robot konumunun

düzeltilmesini gerçekleştirmiştir. GKS'de yapılan öngörülerin ve düzeltilen robot konumlarındaki, belirsizlik kovaryans matrisleri elde edilmiştir.

Deneyde kullanılan odometrik veriler ve sonar verileri yaklaşık 320ms aralıklarla yenilenmiştir. Eğer örnekleme süresi azaltılırsa odometrideki hata azaltılabilir. Ancak sürenin azaltılması kodlayıcılardan elde edilecek vuruş sayısını azaltacağından denetim girdilerinin hassasiyeti azalacaktır. Bir başka iyileştirme olarak robotun hızı azaltılabilir. Böylelikle robot konumu çok değişmeden sonarlardan mesafe bilgileri alınacağından Kalman süzgecinin düzeltimi daha fazla olacaktır. Ancak buradaki olumsuzluk robotun yavaş hareket etmesi ve işlevini daha uzun sürede gerçekleştirecek olmasıdır.

Genişletilmiş Kalman süzgeci denklemlerinde Q ve R parametreleri ile belirtilen, sırasıyla öngörülemez sistem gürültüsünün kovaryans değeri ve ölçüm gürültüsünün kovaryans değeri, her döngüde tekrar hesaplanmıştır. Bu iki parametreyi oluşturan varyans değerleri sırasıyla şekil 7.21 ve şekil 7.24'te gösterilmiştir. Öngörülemez sistem gürültüsünü oluşturan denetim girdileri varyanslarından, doğrusal hız varyansı ve dönme hızı varyansı şekil 7.21'de; ölçüm gürültüsünü oluşturan her bir sonarın ölçüm hata varyansları şekil 7.24'te gösterilmiştir. Kalman Süzgeci uygulamalarında bu iki kovaryans matrislerinin değerleri çevirim dışı olarak hesaplanır ve sabit bir değer olarak Kalman süzgecine uygulanır. Bu tez çalışmasında elde edilen denetim girdilerinin varyansları ve ölçüm gürültülerinin varyansları sabit alınmamıştır. Dolayısıyla Q ve R kovaryanslarının değeri her döngüde değişmiş ve sürecin yeni durumları ile güncellenmiştir. Kalman süzgeci, denetim girdilerindeki değişim(varyans) fazla olduğunda ölçüm gürültülerine, ölçüm gürültülerindeki değişim(varyans) fazla olduğunda da sistem gürültülerine bağımlı olarak durum kestirimini gerçekleştirir. Bu çalışmada varyanslar sürekli olarak yenilendiği için Kalman süzgecinin çalışması robotun yaptığı hareketlere ve ölçüm değerlerine göre olur. Başka bir ifade ile Q kovaryansı artarsa bir önceki durum belirsizliği de artar ve durum kestiriminde ölçüm verilerinin etkisi artar; R kovaryansı artarsa bir sonraki durum belirsizliği de artar ve durum kestiriminde denetim girdilerinden alınan verilerin etkisi artar. Sonuç olarak yüksek gürültü değerleri durum kestiriminde daha az etki yapmış olur.

Şekil 7.21'deki varyans değişimlerine bakılarak robotun hangi hareketleri yaptığı anlaşılabilir. Örneğin dönüş hız girdisi varyansının başlangıçta sıfır olması, robotun başlangıçtan itibaren, yaklaşık 100 örnekleme süresince, dönüş hareketi yapmadığını gösterir. Her iki varyansın değişimi şeklinde testere dişi gibi oluşan değişimler robotun dönüş hareketinin gerçekleştiği anları belirtir. Şekil 7.24'te her bir sonarın ölçümlerindeki hataların varyansları verilmiştir. Bu varyans değerlerinin belirli bir örneklem sayısından sonra sabit değerlere yakınsadığı görülmektedir. R arttığı sürece Kalman kazancı azalarak, durumun daha yavaş gerçek duruma yakınsamasını sağlar. Başka bir ifade ile ölçüm verilerine güvenilmez. R azaldıkça bu yakınsama hızı artar. R değerinin sıfıra gitmesi durumunda ise ölçüm verilerine güvenilirlik artar ve daha doğru durum kestirimi yapılır. Kısacası istenilen R 'nin azalmasıdır. Yapılan çalışmada R değerlerinin azalmaya başladığı sırada tekrar artarak belirli değerler arasında kalması, robot hareketinin sürekli olarak değişmesinden, başka bir ifade ile denetim girdilerindeki varyans değerlerinin sürekli olarak değişmesinden kaynaklanmaktadır. Her iki varyans değerinin değişiminin belirli değerlere yakınsamasının nedeni budur. Dolayısı ile Kalman kazancı belirli değerler arasında kalarak durum kestirimini gerçekleştirir.

Bu tez çalışmasında sonar verilerinden elde edilen mesafe bilgileri, doğrusal olarak değerlendirilmiştir. Ancak kullanılan sonar algılayıcıların algılama konisi 25° 'liktir. Bu varsayımın yapılmasına rağmen, GKS kullanılarak elde edilen robot konumu odometrik veriler ile elde edilen robot konumundan daha doğru sonuçlar vermiştir. Robot konumunun belirlenmesi için odometrik verilere ek olarak kullanılan sonar algılayıcılar yerine lazer algılayıcılar kullanmak daha verimli olabilir. Ancak birçok uygulama sonar algılayıcılar ile yapılmaya çalışılmaktadır. Bunun nedeni, lazer algılayıcıların algılama konisinin sonar algılayıcılara göre çok az olmasına karşın düzlemsel algılama yapmalarından ve sonar algılayıcıların maliyetlerinin daha az olmasıdır.

Sonuç olarak gezgin robot konumunun belirlenmesinde genişletilmiş Kalman süzgeci kullanımı ile odometrik verilerden kaynaklanan konum hataları azaltılmış ve daha doğru sonuçlar elde edilmiştir.

KAYNAKLAR LİSTESİ

- [1] <http://capek.misto.cz/english/robot.html> (18 Haziran 2006,WEB)
- [2] Leonard, J. ve Durrant-Whyte, H. (1992). Directed Sonar Sensing for Mobile Robot Navigation. Boston: Kluwert Accademic Publisher.
- [3] J Borenstein, H. R. Everett, ve L. Feng, Where am I? Sensors and Methods for Mobile Robot Positioning., Ann Arbor, MI 48 109:, University of Michigan, 1996.
- [4] N. Roy ve S. Thrun, Online self-calibration for mobile robots:' in In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 1999.
- [5] I. J. Cox, Blanche: Position estimation for an autonomous robot vehicle, Autonomous Mobile Robots: Control, Planning, and Architecture(Vol. 2), IEEE Computer Society Press, Los Alamitos, CA (1991),285-292.
- [6] D. Fox, W. Burgard, ve S. Thrun, Markov localization for mobile robots in dynamic environments. Journal of Artificial Intelligence Research 11 (1999), 391-427.
- [7] J. Borenstein ve L. Feng, Measurement and correction of systematic odometry errors in mobile robots. IEEE Transactions on Robotics and Automation 12 (1996), 869-880.
- [8] A. Singhal, Issues in Autonomous Mobile Robot Navigation (1997).
- [9] J. Manyika and H. Durrant-Whyte, Data Fusion and Sensor Management, a decentralized information-theoretic approach, Ellis Horwood Limited, Chichester, West Sussex (1994).
- [10] J. Borenstein, H. Everett, L. Feng, ve D. Wehe, Mobile robot positioning: Sensors and techniques. Journal of Robotic Systems 14,4 (1997), 231-249.
- [11] NEGENBORN, Rudy, Robot Localization and Kalman Filters, On fnding your position in a noisy world, Utrecht University (2003),23-26
- [12] Moravec, H. P., and Elfes, A. 1985. High resolution maps from wide angle sonar. In *Proc. IEEE Int. Conf. Robotics and Automation*, 116–121.
- [13] W. Burgard, D. Fox, D. Hennig, and T. Schmidt, Estimating the absolute position of a mobile robot using position probability grids. Proceedings of the Fourteenth National Conference on Artificial Intelligence (1996), 896-901.

- [14] D. Fox, S. Thrun, W. Burgard, ve F. Dellaert, Particle Filters for mobile robot localization. Sequential Monte Carlo Methods in Practice. Springer. New York (2001).
- [15] GREWAL, Mohinder S., ANDREWS, Angus P., Kalman Filtering: Theory and Practice Using MATLAB, 2nd Edition, John Wiley & Sons 2001
- [16] MAYBECK, P.S., Stochastic Models, estimation and control, Academic Pres, 1979
- [17] WELCH, Greg ve BISHOP, Gary, An Introduction to the Kalman Filter, SIGGRAPH 2001 course pack edition, Annual Conference on Computer Graphics & Interactive Techniques. ACM Press, Addison-Wesley, 2001.
- [18] WELLING, M., The Kalman Filter, California Institute of Technology 136-93 <http://www.ics.uci.edu/~welling> (20 Temmuz 2006).
- [19] PEEBLES, Peyton Z, Probability, Random Variables, and Random Signal Principles, 4th edition, McGraw-Hill, 1993.
- [20] E. Ivanjko, I. Petrović, Extended Kalman Filter based Mobile Robot Pose Tracking using Occupancy Grid Maps, IEEE MELECON 2004, May 12-15, 2004, Dubrovnik, Croatia.
- [21] Edouard Ivanjko, Mario Vasak, and Ivan Petrovic, Kalman Filter Theory Based Mobile Robot Pose Tracking Using Occupancy Grid Maps, 2005 International Conference on Control and Automation (ICCA2005), June 27-29, 2005, Budapest, Hungary.
- [22] Atsushi Fujimori, Takafumi Murakoshi ve Yoshinao Ogawa, Navigation ve Path-Planning of. Mobile Robots with Real-Time Map-Building, Shizuoka University, 2002
- [23] Arati Gopalakrishnan, Sheldon Grene ve Ali Sekmen, Vision-Based Mobile Robot Learning and Navigation, 2005, IEEE International Workshop on Robots and Human Interactive Communication
- [24] Bilgic, T. & Turksen, I, Model-based localization for an autonomous mobile robot, Proceedings of the 1995 IEEE Conference on Systems, Man and Cybernetics, IEEE, Vancouver, B.C. Canada. <http://citeseer.ist.psu.edu/314475.html>
- [25] A. Howard, M. J. Matarić, ve G. S. Sukhatme. Localization for mobile robot teams: A distributed MLE approach. In Submitted to the 8th International Symposium on Experimental Robotics (ISER'02), 2002. <http://citeseer.ist.psu.edu/howard02localization.html>
- [26] Fang Fang, Xudong Ma ve Xianzhong Dai, A Multi-sensor Fusion SLAM Approach for Mobile Robots*, IEEE International Conference on Mechatronics & Automation, Niagara Falls, Canada, July 2005

- [27] Gianluca Antonelli ve Stefano Chiaverini, Experimental Odometry Calibration of the Mobile Robot Khepera II Based on the Least-Squares Technique, IEEE International Conference on Robotics and Automation Barcelona, Spain, April 2005
- [28] Agostino Martinelli, Nicola Tomatis, Adriana Tapus ve Roland Siegwart, Simultaneous Localization and Odometry Calibration for Mobile Robot, International Conference on Intelligent Robots and Systems, Las Vegas, Nevada, October 2003
- [29] <http://www.mobilerobots.com/>

ÖZGEÇMİŞ

Yazar 1980 yılında Ankara'da doğdu. İlkokul eğitimini Köy Hizmetleri İlkokulu'nda, ortaokul ve lise eğitimini Bahçelievler Deneme Lisesi'nde tamamladı. 2000 yılında Başkent Üniversitesi Yabancı Dil Hazırlık Bölümünü bitirdikten sonra, 2000-2004 yılları arasında lisans öğrenimini Başkent Üniversitesi Mühendislik Fakültesi Elektrik Elektronik Mühendisliği Bölümü'nden bölüm birinciliği derecesi ile mezun olarak tamamlamıştır. 2004 yılında başladığı yüksek lisans eğitimini Başkent Üniversitesi Fen Bilimleri Enstitüsü-Elektrik Elektronik Mühendisliği Anabilim Dalı'nda sürdürmektedir.

EKLER

EK 1

FEATURE	ROBOT BASE = Pioneer 3-DX
Length	44.5cm (44)
Width	40.0cm (38)
Height (body)	24.5cm (22)
Body clearance	6.5cm (6)
Weight (with min. battery capacity)	9kg
Payload of base platform with included battery (other drive ratios available on large orders)	23kg flat 14kg @ 13% grade
Body	1.6mm CNC fabricated, painted aluminum
Battery Access	Hinged, latched access door
Assembly	Allen hex screws
Battery	12V sealed, lead-acid
Charge	252 watt-hr
Run time, base platform	18-24 hrs
Recharge time, standard charger	12 hrs
Recharge time, 1 high-capacity charger	2.4 hrs
Docking station available	Yes
Drive	2-wheel drive, plus rear balancing caster
Wheel composition	Foam filled nylon, hard casters
Dr. Wheel diam.	19cm
Dr. Wheel width	5cm
Steering	Differential
Gear ratio	38.3:1
Pushing force	6kg
Swing radius	32cm
Turn radius	0cm
Translate speed max	1.2 m/sec
Traversable step max	2.5cm
Traversable gap max	8.9cm
Traversable slope max	25% grade
Traversable terrains	Wheelchair accessible
ARIA Robotics API	Yes
ACTS Color-Track	Requires onbd PC
Festival/Sphinx	Requires onbd PC
Laser Mapping & Nav.	Requires laser
Front sonar ring	8 included 1 each side 6 forward @ 20° intervals
Rear sonar ring	8 optional 1 each side 6 rear @ 20° intervals
Sonar range	15cm - 5m
Std. Position encoders	500 tick encoders
Surveillance option	Yes
IR Table Sensors	No
Compass option	Yes
Arm	Yes
Gripper option	Yes
Bumper option	Yes
Wireless Ethernet Operation option	Yes
Optional onboard computer	Embedded size
Max. no. Cards & ports	3 PC104+ 2 USB, 4 serial
Speaker	Piezo std., opt. high decibel
Laser option	Yes
Gyro option	Yes
GPS option	Yes
Vision/ACTS Color- Tracking options	Yes
Speech/Voice option	Yes
StereoCam Rangefinder option	Yes
Processor	Hitachi H8S
Sonar inputs	16 max
Custom I/O connections	8-bit external I/O bus w/ up to 16 devices + PC104 I/O boards
Analog/Digital	5 @ 0-5 VDC
Communications ports	3 RS-232 serial ports on microcontroller, 4 RS-232 and 1 Ethernet on optional embedded computer
Wireless Communications options	Radio modem pair without embedded computer; Ethernet station adapter & access point with
Flash Memory	1 mB
Power switches	1 main; 2 auxiliary
LCD display	-na-
Reset pushbutton	Warm reboot
Function pushbutton	Self-test
Main power switch	Robot power; 12VDC; red LED indicator
Radio power switch	Radio modem or other 12VDC option
Aux power switch	12VDC
Motors pushbuttons	Single enable/disable
Serial comm ports	9-pin RS232 with Rcv and Xmt LED indicators
Joy drive port	Off & opt. onbd.
Charging	12VDC charge port & Docking

