

**MOBİL ROBOTLARDA ÇOKLU-SENSÖR VERİ
FÜZYONU TEKNİKLERİ İLE LOKALİZASYON VE
HARİTA OLUŞTURMA**

**T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ**

**DOKTORA TEZİ
Y. Müh. Levent YENİLMEZ
(504942019)**

104252

**Tezin Enstitüye Verildiği Tarih : 9 Ekim 2000
Tezin Savunulduğu Tarih : 18 Mayıs 2001**

Tez Danışmanı :

Doç.Dr. Hakan TEMELTAŞ

Diğer Jüri Üyeleri

Prof.Dr. Mahmut ÜN (İ.Ü.)

Doç.Dr. Yağmur DENİZHAN (B.Ü.)

Doç.Dr. Seta BOĞOSYAN (İ.T.Ü.)

Doç.Dr. Çoşkun SÖNMEZ (İ.T.Ü.)

MAYIS 2001

ÖNSÖZ

Bu çalışma, 197E043 nolu, “Mobil Robotlarda Çoklu Sensör Tabanlı Veri Füzyonu ile Çevre Haritalama” adlı TÜBİTAK EEEAG projesi kapsamında gerçekleştirilmiştir.

Öncelikle bu çalışmayı yapabilme imkanı sağlayan Hava Kuvvetleri Komutanlığı’na, İstanbul Teknik Üniversitesi’ne, Türkiye Bilimsel ve Teknik Araştırma Kurumu’na ve değerli yardımlarıyla bu çalışmaya yön veren hocam sayın Doç. Dr. Hakan TEMELTAŞ’a en içten dileklerimle teşekkür ederim.

Son olarak bu çalışmalar sırasında bana sabırla tahammül eden ve her zaman desteğini esirgemeyen değerli eşim Nurhan YENİLMEZ’e sonsuz teşekkürler ederim.

Levent YENİLMEZ

İstanbul, Mayıs 2001

İÇİNDEKİLER

KISALTMALAR	vi
TABLO LİSTESİ	vii
ŞEKİL LİSTESİ	viii
SEMBOL LİSTESİ	x
ÖZET	xiii
SUMMARY	xv
1. GİRİŞ	1
1.1. Çalışmanın Amacı	1
1.2. Çalışmanın Kapsamı ve Literatürde Yer Alan Çalışmalar	2
1.3. Çalışma Olanakları	4
1.4. Çoklu-Sensör Veri Füzyonuna Genel Bakış	4
2. EŞ-ZAMANLI ÜÇ TEKERLEK YAPISINDAKİ MOBİL ROBOTLARDA KİNEMATİK MODELLEME	8
2.1. Giriş	8
2.2. Tanımlar ve Varsayımlar	9
2.3. Koordinat Çerçevelerinin Atanması	10
2.4. Dönüşüm Matrislerinin Çıkarılması	14
2.5. Eş Zamanlı Üç Tekerlekli Mobil Robotlarda Dönüşüm Matrisleri	16
2.6. Sonuç	21
3. SENSÖRLER VE MODELLERİ	22
3.1. Giriş	22
3.2. Odyometrik Sensör Sistemi ve Modeli	23
3.2.1. Odyometrik hataların ölçülmesi	25
3.3. Ultrasonik Sensörler ve Modeli	33
3.3.1. Ultrasonik sensörlerde konum hata analizi	37
3.3.2. Çevre haritalama amaçlı sensör modeli	39
3.4. Kızılötesi Sensörler ve Modeli	42
3.5. Manyetik Pusula	43
3.6. Fiber-Optik Jireskop	45
3.7. Sonuç	46
4. LOKALİZASYON YÖNTEMLERİ VE SINIRLANDIRILMIŞ DİNAMİK SENSÖR FÜZYONU-SDSF	48
4.1. Giriş	48
4.2. Konumu Bilinen Nesnelerle Lokalizasyon	50
4.2.1. Robot konumunun kestirimi	51
4.2.2. Robot yönünün kestirimi	53
4.2.3. Newton özyinelemeli lokalizasyon-NÖL yöntemi	54
4.2.4. NÖL yönteminin Nomad200 mobil robotu üzerinde testi	57
4.3. Genişletilmiş Kalman Filtreleme-GKF Yöntemiyle Lokalizasyon	60
4.3.1. Robot durum uzayı modeli	61

4.3.2. Sensör verilerinin GKF ile birleştirilmesi	64
4.3.2.1. Durum öngörüsü	64
4.3.2.2. Durum kestirimlerinin güncellenmesi	64
4.3.3. GKF yönteminin Nomad200 mobil robotu üzerinde testi	66
4.4. Sınırlandırılmış Dinamik Sensör Füzyonu-SDSF Yöntemiyle Lokalizasyon	69
4.4.1. İki boyutlu sensör füzyonu probleminin tanımı	69
4.4.2. Mobil robot konum ve hızının bulunması	70
4.4.3. Robot yönünün bulunması	74
4.4.4. SDSF yönteminin Nomad200 mobil robotu üzerinde testi	77
4.4.4.1. SDSF yönteminin odyometre (odo), jireskop (jir) ve pusula (pus) sensör üniteleriyle testi	78
4.4.4.2. SDSF yönteminin ultrasonik konumlandırma (ult), jireskop (jir) ve pusula (pus) sensör üniteleriyle testi	79
4.4.4.3. SDSF yönteminin odyometre (odo), ultrasonik konumlandırma (ult), jireskop (jir) ve pusula (pus) sensör üniteleriyle testi	81
4.5. SDSF Yönteminin GKF Yöntemiyle Karşılaştırılması	82
4.6. Sonuç	87
5. ÇOKLU SENSÖR VERİ FÜZYONU TEKNİKLERİ İLE ÇEVRE HARİTALAMASI	88
5.1. Giriş	88
5.2. Mobil Robotlarda Sensörlere Dayalı Çevre Modellemesi	90
5.2.1. Ultrasonik sensörlerle çevre haritalama	91
5.2.2. Çalışılan laboratuvar ortamının modellenmesi	93
5.3. Bayes Yöntemi ile Çevre Modeli	93
5.3.1. Yöntemin ultrasonik sensörlere uygulanması	94
5.3.2. Yöntemin kızılötesi sensörlere uygulanması	96
5.3.3. Dempster-Shafer kuralıyla iki sayısal haritanın birleştirilmesi	97
5.4. Ardışıl Temel Bileşenler-ATB Yöntemiyle Veri Birleştirme	98
5.4.1. Yöntemin ultrasonik sensörlerle kullanımı	100
5.4.2. Yöntemin ultrasonik ve kızılötesi sensörlerle kullanımı	101
5.4.3. Yöntem için karar verme algoritması	102
5.4.4. Yöntem üzerine yapılan deneysel çalışmalar	103
5.5. ATB Yönteminin Bayes Yöntemiyle Karşılaştırılması	106
5.6. Sonuç	107
6. KONUMLANDIRMA VE ÇEVRE HARİTALAMA YÖNTEMLERİNİN NAVİGASYON AMAÇLI BİRLEŞTİRİLMESİ	109
6.1. Giriş	109
6.2. Lokalizasyon ve Çevre Haritalama Bilgilerinin Birleştirilmesi	111
6.3. Navigasyon Amaçlı Engellerden Sakınma ve Yol Planlama	111
6.3.1. Yapay Potansiyel Alanı-YPA yöntemiyle yol planlama	112
6.3.1.1. Mobil robot hız bilgisinin oluşturulması	115

6.3.1.2. YPA algoritmasının Nomad200 mobil robotu üzerinde testi	116
6.4. Eş-Zamanlı Üç Tekerlek Yapısındaki Robotlarda Hareket Planlama	117
6.5. Sonuç	119
7. SONUÇLAR VE ÖNERİLER	121
7.1. Sonuçlar	121
7.2. Geleceğe yönelik tavsiyeler	123
KAYNAKLAR	124
EKLER	130
ÖZGEÇMİŞ	134



KISALTMALAR

odo	:Odyometre
ult	:Ultrasonik
kzl	:kızılötesi
pus	:manyetik pusula
jir	:fiber-optik jireskop
diag	:diyagonal
sy	:saat yönünde
syt	:saat yönünün tersinde
UMBmark	:University of Michigan Benchmark
Matlab	:Mathematical laboratory
SF	:Scale Factoring
NÖL	:Newton Özyinelemeli Lokalizasyon
GKF	: Genelleştirilmiş Kalman Filtresi
SDSF	: Sınırlandırılmış Dinamik Sensör Füzyonu
GTOM	:Gridleme Tabanlı Olasılıksal Modelleme
HIMM	:Histogramic In-Motion mapping
ATB	:Ardışıl Temel Bileşenler
CMU	:Carnegie-Mellon University
CV	:Certanity value
YPA	:Yapay Potansiyel Alanı
YSA	:Yapay Sinir Ağları
BM	:Bulanık Mantık
VD	:Voronoi Diyagramları

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 3.1. Amaçlarına göre kullanılan sensör üniteleri	22
Tablo 3.2. Saat yönündeki (sy) odyometrik ölçüm hataları	26
Tablo 3.3. Saat yönü tersindeki (syt) odyometrik ölçüm hataları	26
Tablo 3.4. Saat yönündeki (sy) kalibreli odyometrik ölçüm hataları	31
Tablo 3.5. Saat yönü tersindeki (syt) kalibreli odyometrik ölçüm hataları	31
Tablo 3.6. Ultrasonik sensörlerin d=300 cm uzaklıktaki bir nesneyi değişen ortam sıcaklıklarına göre okuması ile hesaplanan ses hızları	34
Tablo 3.7. KVH C100 pusulasının teknik özellikleri	44
Tablo 3.8. KVH Jireskopunun Teknik Özellikleri	46
Tablo 4.1. İki tur sola dönüş (syt) sonrasındaki konum ve yön hataları.....	58
Tablo 4.2. İki tur sağa dönüş (sy) sonrasındaki konum ve yön hataları	59
Tablo 4.3. Kalman filtresiyle iki defa sola dönüşle (syt) oluşan konum ve yön hataları	68
Tablo 4.4. Kalman filtresiyle iki defa sağa dönüşle (sy) oluşan konum ve yön hataları	68
Tablo 4.5. Odyometre, jireskop ve pusula ile sola (syt) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları	78
Tablo 4.6. Odyometre, jireskop ve pusula ile sağa (sy) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları	79
Tablo 4.7. Ultrasonik sistem, jireskop ve pusula ile sola (syt) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları	80
Tablo 4.8. Ultrasonik sistem, jireskop ve pusula ile sağa (sy) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları	80
Tablo 4.9. Odometrik ve ultrasonik sistemin jireskop ve pusula ile beraber sola (syt) dönüşler şeklindeki iki turu sonucunda oluşan konum ve yön hataları	81
Tablo 4.10. Odometrik ve ultrasonik sistemin jireskop ve pusula ile beraber sağa (sy) dönüşler şeklindeki iki turu sonucunda oluşan konum ve yön hataları	81

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1 : Bir tekerin ilerletilmesine yönelik iki serbestlik dereceli ve döndürülmesine yönelik bir serbestlik dereceli yüzeysel modeli. (a) Yüzeysel model, (b) Uygun bir teker modeli.....	11
Şekil 2.2 : Tekerlekli bir mobil robot için atanan koordinat çerçeveleri.....	12
Şekil 2.3 : Bir teker için koordinat çerçevelerinin atanması (dönüşüm diyagramı).....	13
Şekil 2.4 : Koordinat çerçevelerinin konumları.....	13
Şekil 2.5 : Üç tekerlekli robot için dönüşüm grafiği.....	16
Şekil 2.6 : Nomad200 mobil robotu.....	17
Şekil 2.7 : (a) Nomad200 tekerleklerinin z-eksenine göre (üstten bakıldığında), koordinat çerçeveleri, (b) Tekerlekleri sağa ve sola çevirebilen dişli mekanik sistem.....	18
Şekil 2.8 : (a) Üstten bakıldığında birinci tekere ait koordinat çerçeveleri, (b) Üstten bakıldığında ikinci tekere ait koordinat çerçeveleri, (c) Üstten bakıldığında üçüncü tekere ait koordinat çerçeveleri, (d) Yandan bakıldığında birinci tekere ait koordinat çerçeveleri....	20
Şekil 3.1 : Odyometrik hata ölçümleri için laboratuarda planlanan yol.....	27
Şekil 3.2 : Saat yönü istikametinde (sürekli sağa dönüşlerle) ortaya çıkan odyometrik hatalar.....	27
Şekil 3.3 : Saat yönünün tersi istikametinde (sürekli sola dönüşlerle) ortaya çıkan odyometrik hatalar.....	28
Şekil 3.4 : Saat yönü istikametinde (sürekli sağa dönüşlerle) ortaya çıkan ayarlanmış odyometrik hatalar.....	32
Şekil 3.5 : Saat yönünün tersi istikametinde (sürekli sola dönüşlerle) ortaya çıkan ayarlanmış odyometrik hatalar.....	32
Şekil 3.6 : Çalışan ortam sıcaklığına göre değişen havadaki ses hızı.....	35
Şekil 3.7 : Çalışılan ortam sıcaklığına göre ultrasonik okumaları kalibre eden C_T değerleri.....	36
Şekil 3.8 : Ultrasonik sensörlerin ışıma örüntüsü.....	37
Şekil 3.9 : Ultrasonik sensörlerde konumsal hata dağılımı.....	38
Şekil 3.10 : Ultrasonik sensörlerin açısal hata dağılımı.....	39
Şekil 3.11 : Bir ultrasonik sensörün konik açılımının iki-boyutlu görünümü....	40
Şekil 3.12 : Ultrasonik sensöre bağlı Olasılıksal Yoğunluk Fonksiyonunun atanacağı $\pm\beta$ grid hücreleri.....	41
Şekil 3.13 : Kızılötesi sensörlerde konumsal hata dağılımı.....	43
Şekil 4.1 : Ultrasonik sensörlerle NÖL yöntemi kullanılarak gerçekleştirilen lokalizasyon.....	51
Şekil 4.2 : Ultrasonik sensörler tarafından referans nesnelerin yakalanması...	52
Şekil 4.3 : Robot koordinat sisteminin global koordinat sistemindeki görüntüsü.....	53

Şekil 4.4	: Test amaçlı planlanan robot yolu ve ortamdaki referans silindirik nesnelerin konumları.....	57
Şekil 4.5	: Bir tur sola dönüş (syt) sonrasında planlanan yolda ortaya çıkan sapmalar.....	59
Şekil 4.6	: Bir tur sağa dönüş (sy) sonrasında planlanan yolda ortaya çıkan sapmalar.....	60
Şekil 4.7	: Bir örnekleme (τ) süresince değişen robot pozisyonu.....	62
Şekil 4.8	: A ve B sensör okumalarının 2-boyutlu füzyonu.....	72
Şekil 4.9	: Global koordinat sistemindeki robot koordinatları.....	74
Şekil 4.10	: Robot baş açısı θ_r 'ı veren A ve B sensör okumalarının üst ve alt sınırları.....	77
Şekil 4.11	: Sola (syt) iki tur dönüşler sırasında odyometre, pusula ve jireskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.....	84
Şekil 4.12	: Sağa (sy) iki tur dönüşler sırasında odyometre, pusula ve jireskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.....	84
Şekil 4.13	: Sola (syt) iki tur dönüşler sırasında ultrasonik konumlandırma ünitesi, pusula ve jireskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.....	85
Şekil 4.14	: Sağa (sy) iki tur dönüşler sırasında ultrasonik konumlandırma ünitesi, pusula ve jireskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.....	85
Şekil 4.15	: Sola (syt) iki tur dönüşler sırasında Odyometre, ultrasonik konumlandırma ünitesi, pusula ve jireskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları	86
Şekil 4.16	: Sağa (sy) iki tur dönüşler sırasında odyometre, ultrasonik konumlandırma ünitesi, pusula ve jireskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.....	86
Şekil 5.1	: Ultrasonik sensörün iki-boyutlu konik açılımı.....	92
Şekil 5.2	: Çevre modelinin koordinat hücreleri şeklinde gridlenmesi.....	94
Şekil 5.3	: Ultrasonik sensörlerle çıkarılan olasılıksal çevre.....	96
Şekil 5.4	: Kızılötesi sensörlerle çıkarılan olasılıksal çevre.....	97
Şekil 5.5	: Birleştirilmiş olasılıksal çevre.....	98
Şekil 5.6	: Basit bir lineer yapı.....	99
Şekil 5.7	: 3'lü ardışıl ultrasonik sensör grupları.....	101
Şekil 5.8	: Ultrasonik ve Kızılötesi sensör grubu.....	102
Şekil 5.9	: Karar verme algoritmasında sensör seçimi.....	103
Şekil 5.10	: Nomad200 ile gerçek zamanda oluşturulan çevre haritası.....	105
Şekil 5.11	: $P > P_{eşik} = 0.75$ olasılıksal değerleri ile oluşturulan çevre haritası....	106
Şekil 6.1	: Birleştirilmiş lokalizasyon ve çevre haritalama sisteminin blok diyagramı.....	112
Şekil 6.2	: Robotla birlikte hareket eden 11×11 lik aktif A^* bölgesi.....	114
Şekil 6.3	: Robot üzerinde oluşturulan kuvvet alanları.....	115
Şekil 6.4	: Laboratuvar ortamında YPA algoritması ile planlanan robot yolu.....	117
Şekil 6.5	: Robot yolunun dairesel parçalarla oluşturulması.....	118

SEMBOL LİSTESİ

${}^R[T]_N$	: Standart dönüşüm matrisi
${}^R[\theta]_N$	: İki çerçeve arasındaki dönüş açısıdır
${}^R[v]_N$	: Hız dönüşüm matrisi
${}^R[\omega]_N$	: Açısal hız dönüşüm matrisi
${}^R[\alpha]_N$	: İvme dönüşüm matrisi
ω	: Açısal hız
α	: Açısal ivme
v_x	: Doğrusal hızın x-bileşeni
α_x	: Doğrusal ivmenin x-bileşenidir
τ	: Örnekleme zamanı
$v(t)$	: Robotu ileri süren hız
$w(t)$	: Robot tekerinin sağa veya sola dönüş hızı
$[g_1 \ g_2 \ g_3]^T$	: Odyometrik gürültüler
$(\Delta x_r)_{sy/syt}$	: Odyometrik sistem hatasının x-bileşeni
$(\Delta y_r)_{sy/syt}$	: Odyometrik sistem hatasının y-bileşeni
n	: Robotun yaptığı tur miktarı
$r_{sy/syt}$	: Odyometrik sistem hatasının yarıçapı
$E_{maks,odo}$	: Odyometrik hata miktarı
$(\theta_{odo})_{sy}$	: Tekerlerin sağa dönüşünden sonra okunan odyometrik robot yönü
$(\theta_{odo})_{sy-ger}$	: Sağa dönüşlerde olması gereken gerçek odyometrik robot yönü,
$(\theta_{odo})_{syt}$	: Tekerlerin sola dönüşünden sonra okunan odyometrik robot yönü,
$(\theta_{odo})_{syt-ger}$	: Sola dönüşlerde olması gereken gerçek odyometrik robot yönüdür.
d_u	: Ultrasonik sensörlerle ölçülen mesafe
V_s	: Havada sesin yaptığı hız
t_o	: Gönderilen ses sinyalinin dönüş zamanı
0C	: Ortam sıcaklığı
d_{ger}	: Ultrasonik sensörlerden ortam sıcaklığına göre alınan gerçek mesafe
C_T	: Ultrasonik sensör okumalarını ortam sıcaklığına ayarlayan değer
$E(d_u)_{mes}$	: Ultrasonik sensörden kaynaklanan mesafeye bağlı okuma hatası
β	: Ultrasonik sensörün konik alanı içinde yer alan açılar
$E(\beta)_{açı}$	: Ultrasonik sensörden kaynaklanan açığa bağlı okuma hatası
$P_u(d_u r, \beta)$	: Ultrasonik okumalara göre engellerin hücrelerde temsil olasılıkları
r	: Robotla ölçülen nesne arasındaki gerçeğe yakın mesafe değerini
$P_k(d_k r)$	: Kızılötesi okumalara göre engellerin hücrelerde temsil olasılıkları

d_k	: Kızılötesi sensörün ölçtüğü mesafe
$E(d_k)_{mes}$	: Kızılötesi sensörden kaynaklanan mesafeye bağlı okuma hatası
h_x, h_y	: Manyetik alan şiddetinin yatay bileşenleri
θ_o	: Manyetik pusuladan ölçüm yapılmadan önce okunan sabit değer
θ_{pus}	: Manyetik pusulanın okuduğu yön miktarı
SF	: Jireskopun üretici firması tarafından verilen sabit değeri
w_g	: Jireskopun okuduğu açısal dönüş hızı
w_{bias}	: Jireskopun durağan haldeyken ısıya bağlı ürettiği sabit değer
w_{jir}	: Jireskop tarafından robota sağlanan net açısal dönüş hızı
θ_r	: Robot yönü (baş açısı)
x_r, y_r	: Robot konumunun x ve y -bileşenleri
$B_1, \dots, B_6$	: Laboratuvarda kullanılan ve konumu bilinen referans nesneler
x_{bi}, y_{bi}	: Konumu bilinen referans nesnelerin koordinatları
d_i	: ultrasonik sensörle ölçülen robot ile referans nesne arasındaki mesafe
β_j	: j nci ultrasonik sensörün robot koordinat sistemindeki bulunduğu açı
x_R, y_R	: Robot koordinat sisteminin x ve y -eksenleri
x_G, y_G	: Global koordinat sisteminin x ve y -eksenleri
θ'_r	: Robot ve global koordinat sistemlerinin x -eksenleri arasındaki açı
f_i	: Robot konumunun bulunmasında kullanılan fonksiyonlar
u, v	: f_1 ve f_2 fonksiyonlarını ifade eden semboller
x_{rk}, y_{rk}	: k -anındaki robot konumunun x ve y -bileşenleri
x_{rk+1}, y_{rk+1}	: $k+1$ -anındaki robot konumunun x ve y -bileşenleri
u_k, v_k	: k -anında f_1 ve f_2 fonksiyonlarını ifade eden semboller
$J(x_{rk}, y_{rk})$	: k -anındaki robot konumuna ait Jacobian matrisi
$\Delta u, \Delta v$	: f_1 ve f_2 fonksiyonlarında oluşan değişim
$\Delta x_r, \Delta y_r$	: Robot konumunda oluşan değişim
x_o, y_o, θ_o	: Robot konum ve yönünün başlangıç noktası
X_k, X_{k+1}	: k ve $k+1$ anlarında robot konumu, yönü, doğrusal ve açısal hızını içeren durum vektörü
v_k, v_{k+1}	: k ve $k+1$ anlarındaki robotun doğrusal hızı
w_k, w_{k+1}	: k ve $k+1$ anlarındaki robotun açısal hızı
$\theta_{rk}, \theta_{rk+1}$	: k ve $k+1$ anlarındaki robot yönü
$f(X_k)$	: τ örnekleme anında robot konumu, yönü, doğrusal ve açısal hızında meydana gelen farkı ifade eden durum fonksiyonu
$x_{odo}, y_{odo}, \theta_{odo}$	: Odyometrik sistemden alınan robot konum ve yön bilgileri
$x_{ult}, y_{ult}, \theta_{ult}$	: Ultrasonik sensörlerde sağlanan robot konum ve yön bilgileri
w_{jir}	: Fiber-optik jireskoptan sağlanan robotun yaptığı açısal hız bilgisi
θ_{pus}	: Manyetik pusuladan sağlanan robot yön bilgisi
Y_k	: Gözlem vektörü
C	: Gözlem matrisi
ε_k	: k -anında durum uzayı modeline eklenen hatalar
η_k	: k -anında sensör ölçümlerine eklenen gürültüler
Q	: Durum uzayı modeline eklenen hataların ortak değişinti matrisi
R	: Sensör ölçümlerine eklenen gürültülerin ortak değişinti matrisi
$\hat{X}_{k+1 k}$	: k -anında yapılan ölçümlerle $k+1$ anı için robot durum öngörüsü
$P_{k+1 k}$	: $k+1$ anında öngörülen hata ortak değişinti matrisi
$P_{k k}$	: Önceki adımdaki hata ortak değişinti matrisi

J_k	: $\hat{X}_{k k}$ öngörüsündeki $f(\hat{X}_{k k})$ fonksiyonunun Jacobian matrisi
$C(i)$	: Gözlem matrisi C 'nin i -ninci satırı
r_i	: i -adımında ilgili sensör okumalarındaki ölçüm kalıntıları
σ_i	: i -adımında ilgili sensör okumalarındaki standart sapma
s_i	: i -adımında yapılan ölçümün öngörü değışintisi (varyansı)
K_i	: i -adımındaki Kalman filtre kazancı
p_i	: i -sensör biriminden alınan robot konumu, yönü, doğrusal ve açısal hızını veren ifade
x_i, y_i, θ_i	: i - sensör biriminden alınan robot konum ve yön bilgileri
x'_i, y'_i, θ'_i	: i - sensör biriminden alınan robot doğrusal ve açısal hızları
$\Delta x_i, \Delta y_i$	: i - sensör biriminden alınan robot konumunda oluşan hatalar
$\Delta x'_i, \Delta y'_i$	: i - sensör biriminden alınan robot doğrusal hızında oluşan hatalar
$\Delta \theta'_i$	: i - sensör biriminden alınan robot açısal hızında oluşan hatalar
Δt_i	: i - sensör biriminin zamana ait gürültüleri
$f(t)$	: Robotun izlediğı yolun parçalı sürekli bir fonksiyon şeklinde ifadesi
$P_u[s(h_i) = işgal \mid d]$	: Bayesian yöntemiyle bir hücrenin üzerinde engel taşıma olasılığı
$P_u[d \mid s(h_i) = işgal]$	: Ultrasonik okumalara göre engellerin hücrelerde temsil olasılıkları
$P_u[s(h_i) = işgal]$	: Bayesian yöntemiyle hücrenin bir önceki değeri
$\vec{w}$	: ağırlık vektörü
d_i	: Sensör okumaların i ' ninci bileşeni
Δw_i	: i ' ninci okumaya ait ağırlıktaki değışim
ξ	: Öğrenme faktörü
$[R]$	: İlinti (korelasyon) matrisi
λ	: $[R]$ ' nin bütün özdeğerleri
λ_{maks}	: Maksimum özdeğer
$\vec{F}_{i,j}$	: (i, j) aktif grid hücresinin robota uyguladığı itici kuvvet
a^*_{ij}	: (i, j) aktif hücresinin engel taşıma olasılık değeri
F_{cr}	: İtici kuvvet sabiti
x_b, y_b	: Her bir (i, j) aktif grid hücresinin konumu
$d(i,j)$	: Her bir (i, j) aktif grid hücresinin robota olan uzaklığı
$\vec{F}_r$	: Aktif hücrelerin robota uyguladığı itici kuvvetlerin vektörel toplamı
$\vec{F}_t$	: Hedef tarafından robota uygulanan çekici kuvvet
F_{ct}	: Çekici kuvvet sabiti
d_t	: Planlanan hedef ile robot arasındaki mesafe
x_b, y_b	: Hedef noktasının koordinatları
$\vec{R}$	: Robot üzerine uygulanan itici ve çekici kuvvetlerin bileşkesi
β_R	: Robot üzerinde oluşan bileşke vektörün yönü
V_m	: Azami robot hızı
μ	: Hız düzenleme faktörü
M_x, M_y	: Robot yolunu kesen dairenin merkez koordinatları

MOBİL ROBOTLARDA ÇOKLU-SENSÖR VERİ FÜZYONU TEKNİKLERİ İLE LOKALİZASYON VE HARİTA OLUŞTURMA

ÖZET

Bu çalışmada temel olarak deneysel bir mobil robotun lokalizasyonu (konumu ve yönü) ve harita oluşturması için gereken çoklu-sensör veri füzyonu teknikleri verilmiştir. Bu tezde birbirinden bağımsız ilerleyebilen, dönebilen ve gövdesini döndürebilen eş-zamanlı sürücü mekanizmasına sahip Nomad200 [6] mobil robotu kullanılmıştır. Bu nedenle, yapılan çalışmaları destekleyici yönde, üç tekerlekli eş-zamanlı sürücü yapısındaki mobil robotların kinematik modeli çıkartılmıştır. İlâveten, ultrasonik sensörleri, kızılötesi sensörleri, manyetik pusulayı ve fiber-optik jiroskop'u içeren robot sensör sisteminin matematiksel modelleri çıkartılarak bunların hatalı veri okuma durumları analiz edilmiştir. Lokalizasyon ve çevre haritalama işlemlerinde kullanılan bu sensörlerin olasılıksal modelleri üzerinde ayrıca durulmuştur. Bu tezin yan katkısı olarak, yaptığımız çalışmaları destekleyici yönde, sadece robot üzerindeki ultrasonik sensörleri kullanan Newton Özyinelemeli Lokalizasyon yöntemi tanıtılmaktadır.

Robot üzerindeki ultrasonik, kızılötesi, manyetik pusula ve fiber-optik jiroskop sensörlerinden toplanan mesafe ve yön ölçümleri başlıca iki bölümde birleştirilir. Birinci bölümde genel olarak, konum ile yön tahmin yöntemleri ve bu konuda geliştirilen yeni bir yöntem tanıtılmakta ve tartışılmaktadır. Bu tezin temel katkısı olan ve ismine Sınırlandırılmış Dinamik Sensör Füzyonu denilen yeni yöntem tanınmış bir lokalizasyon yöntemi olan Genelleştirilmiş Kalman Filtreleme yöntemiyle karşılaştırılır. Burada, bizim yöntemimiz ile mobil robotun konum ve yönünü sürekli oluşabilecek hatalar biriktirilmeden tam olarak tahmin edilebileceği gösterilmektedir. Diğer bölüm ise, harita oluşturma yöntemlerinin tartışılması ve bu konuda geliştirilen yeni bir yöntemin tanıtılması üzerinedir. Yine bu tezin temel katkısı olan ve ismine Ardışıl Temel Bileşenler denilen yeni yöntem tanıtılarak tanınmış bir çevre-haritalama yöntemi olan Bayes yöntemi ile karşılaştırılmaktadır. Bu karşılaştırma sonucu geliştirilen yöntemin hesaplamada verimli, yanlış sensör okumalarına karşı duyarlı ve mobil robotun herhangi bir engelde durmaksızın sürekli hareket edebilmesine olanak sağlayabildiği sonucu ortaya konmaktadır.

Ayrıca bu her iki bölüm, araştırmacıların lokalizasyon tahmini, çevre haritasının oluşturulması ve engellerden sakınan güvenli bir navigasyon konularında karşılaşılan problemlerle ortak bir şekilde uğraşabilmelerine olanak sağlayabilecek şekilde tek bir yapıda birleştirilir. Birleştirilen bu yapıdan elde edilen veriler bilinen bir yol planlama yöntemi olan Yapay Potansiyel Alanlar [57-59] yönteminde mobil robota güvenli bir yol planlayabilmek amacıyla kullanılır. Eş-zamanlı sürücü mekanizmasına sahip mobil robot, planlanan yolu bu tezin ekler kısmında verilen hareket denklemleri ile izler.

Daha önce birçok araştırmacı karşılaştıkları lokalizasyon ve harita oluşturma konularındaki problemlerin çözümüne yönelik sonuçları sunmuşlardır, fakat her iki problemle aynı anda uğraşmak üzerine az bir çaba sarf edilmiştir. Bu tezde geliştirilen sensör füzyonu yöntemleri bizim laboratuvar mobil robotu olan Nomad200'e uygulanmış ve ona güvenli bir navigasyon için gereken lokalizasyon ve çevre haritası bilgileri birlikte sağlanmıştır.



LOCALIZATION AND MAP BUILDING FOR MOBILE ROBOTS USING MULTI-SENSOR DATA FUSION TECHNIQUES

SUMMARY

In this study, basically, multi-sensor data fusion techniques for the localization (position and direction) and map building of an experimental mobile robot are presented. The mobile robot used in this thesis is Nomad200 [6], which has a synchronous drive mechanism enabling it to translate, steer, and rotate its turret independently. For this reason, the kinematics models of the three wheeled synchronous drive mobile robots were constructed in the direction of supports our studies. In addition, the mathematical models of the robot's sensor system, which include ultrasonic sensors, infrared sensors, magnetic compass, and fiber-optic gyroscope, were constituted to analyze their misreading. Probabilistic models of these sensors used in the process of the localization and map building were discussed with particular attention. The secondary contribution of this thesis, to support our studies, Newton Iterative Localization method, which uses only ultrasonic sensors on the mobile robot, are introduced.

Range and direction measurement data collected from ultrasonic, infrared, magnetic compass and fiber-optic gyroscope sensors on the robot are integrated mainly in two parts. In the first part generally, position and direction estimation methods and a new developed method about this subject are introduced and discussed. This new method, the primary contribution of this thesis, which name is Dynamic Sensor Fusion with Boundaries is compared with the Generalized Kalman Filter, which is the well-known localization method. Here, it is shown that with our method, the position and direction of the mobile robot will be able to estimate precisely, in which position errors are not cumulated. The other part is to discuss the map building methods and explain a new developed method about this subject. This new method, also the primary contribution of this thesis, which name is Sequential Principles of the Components is compared with the Bayesian method, which is the well-known map-building method. Here, it is concluded that our method is computationally efficient and insensitive to misreading, and it allows continuous motion of the mobile robot without stopping for any obstacles.

These two parts are also combined in a unique framework that permits the researchers to treat in a common way with the problems of location estimation, environmental map building and safely navigation avoiding from obstacles. Data obtained from this construction are used in the Artificial Potential Field method [57-59], well-known path planning method, to plan a safety path for the mobile robot. The mobile robot having a synchronous drive mechanism traces the planned road with the motion equations given in the appendix of this thesis.

Previously many researchers have presented their results about the localization and map building problems but very little trial has been made to treat simultaneously with both problems. With this thesis, the developed sensor fusion methods were applied to our laboratory mobile robot, Nomad200, and supplied the localization and map building information simultaneously to navigate it safely.



1. GİRİŞ

1.1 Çalışmanın Amacı

Son yıllarda robotik çalışmaları, çevre düzenlemesi kısmen bilinen yada bilinmeyen ortamlarda verilen görevleri yapabilen kendinden kontrollü mobil robot (Autonomous Mobile Robots) sistemleri şeklinde anılan çalışmalar doğrultusunda gelişmektedir. Buradaki “kendinden kontrollü” terimi mobil robotun statik engellere ve verilen görevin yapılmasını engelleyebilecek dinamik olaylara karşı reaksiyon gösterebilme yeteneğini ifade etmektedir. Kendinden kontrollü Mobil Robotların bu işlevleri yerine getirebilmesi için, bulundukları noktaya ait pozisyon ve mutlak konum bilgilerinin elde edilmesi (Lokalizasyon), içinde bulunduğu ortamın haritasının çıkarılması ve sürekli olarak güncellenmesi, ortam içinde bir noktadan diğer bir noktaya gidilebilmesi için gerekli yolun planlanarak izlenmesi (Navigasyon) ve engellerden sakınma fonksiyonlarının oluşturulması gerekir.

Gerek lokalizasyon gerekse ortam haritasının oluşturulması işlemi, bu amaç için kullanılan sensör verilerinin değerlendirilmesi ile gerçekleştirilmektedir. Bu tip sensörlere örnek olarak radar, ultrasonik, infrared, laser gibi uzaklık sensörleri, jiroskop türü hız ve ivme sensörleri verilebilir. Literatürde tek tip sensör kullanarak çeşitli konum ve yön belirleme, yol planlaması ve engellerden sakınma algoritmalarının geliştirilmesine ilişkin çalışmalara sıkça rastlanmaktadır. Ancak tek tip sensör kullanmanın çeşitli sakıncaları vardır. Bunlar :

- Tek sensör çalışma ortamına ait ancak kısmi bilgi üretir,
- Tek sensörle yapılan ölçümlerde o sensörün tipine ve ölçüm bölgesine göre daima bir hata payı veya belirsizlik söz konusudur,
- Tek sensör hatalı duruma geldiğinde tüm sistem çalışamaz duruma gelir.

Diğer taraftan aynı fiziksel büyüklüğü ölçmek için farklı tipte sensörler kullanıldığında bunlar farklı hata toleranslarında bilgi üretecektir. Bu bilgilerin değerlendirilmesi ile ölçülmesi istenen büyüklük her durumda minimum hata ile elde edilecektir. Bu sebeple çoklu tipte sensörlerin bir arada kullanılması diğer bir deyişle

bir sensör füzyonunun mobil robotlar üzerinde oluşturulmasının amacı bunların çevre algılamadaki hata oranlarının minimuma indirilmesidir.

Bu çalışmada bir mobil robota sensör füzyonu teknikleri uygulayarak, ileri düzey yol planlaması ve engel sakınma işlevlerinin yerine getirilebilmesi için gerekli olan lokalizasyon ve çevre haritasının oluşturulması amaçlanmıştır. Bu amaçla hem lokalizasyon hem de çevre haritasının çıkarılmasında en az 2 en fazla 4 farklı sensör grubu kullanılarak yapılan algılama ile yukarıda sözü edilen fonksiyonlar üzerindeki başarısı gösterilmiştir.

1.2 Çalışmanın Kapsamı ve Literatürde Yer Alan Çalışmalar

Genel olarak bakıldığında bu çalışma kapsamında gerçekleştirilen 5 ana kısım vardır. Bunlar :

A-) Eş-zamanlı üç tekerlek yapısındaki mobil robotların kinematik modelinin çıkartılması: Bölüm 2’de genel olarak mobil robotların kinematik modellemeleri üzerinde durularak uygulamaların denendiği eş-zamanlı üç tekerleğe sahip Nomad200 mobil robotunun kinematik denklemleri çıkarılmaktadır. Literatürde bu konuda yer alan çalışmalar [3-6]’da verilmiştir.

B-) Mobil robotlarda yaygın kullanılan odyometre, ultrasonik, kızılötesi, pusula ve jiroskop sensör modellerinin çıkartılarak hata analizlerinin yapılması: Nomad200 üzerinde bulunan ve kullanılan odyometre, ultrasonik, kızılötesi, manyetik pusula ve fiber-optik jiroskop sensörleri ve bağlı bulundukları sistemler Bölüm 3’de sensörler ve modeller başlığı altında tanıtılarak bu sensörlere ait hata analizlerine geniş bir şekilde yer verilmektedir. Ayrıca bu bölümde lokalizasyon ve çevre haritalama çalışmalarında kullanılmak üzere sensörlerin olasılıksal modellemeleri de yapılmaktadır. Literatürde bu konuda yer alan çalışmalar [7-15]’de verilmiştir.

C-) Mobil robot lokalizasyonunun sensör-füzyon teknikleri ile gerçekleşmesi; Bölüm 4’de bu konuda yapılan çalışmalar aşağıda verildiği gibi 3 kısımda gerçekleştirilmektedir:

- i. Bu çalışmanın bir katkılması olarak robot üzerindeki ultrasonik sensörlerin lokalizasyon amaçlı kullanımını sağlayabilecek şekilde geliştirilen ve ismine yapısından dolayı Newton Özyinelemeli Lokalizasyon (kısaca NÖL) denilen yöntem tanıtılmaktadır. Ayrıca lokalizasyon için belirlenen sensörlere uygun filtreleme ve füzyon yapılarının (Genelleştirilmiş Kalman Filtresi-

GKF [27,28] vb.) seçilerek, karşılaştırma amaçlı algoritmalarının simule edilecek şekilde yazılması ve mobil robotlara uygulanması,

- ii. Yine bu çalışmanın bir katkılması olarak lokalizasyon amaçlı geliştirilen ve Sınırlandırılmış Dinamik Sensör Füzyonu (kısaca SDSF) ismi verilen yöntemin tanıtılması ve mobil robotun dinamik modeli de göz önüne alınarak simule edilerek tasarlanması,
- iii. Geliştirilen SDSF filtre yapısının mobil robot üzerinde test edilmesi ve elde edilen sonuçların bilinen GKF yöntemi sonuçlarıyla tablo ve grafikler yoluyla karşılaştırılması.

Literatürde bu konuda yer alan çalışmalar [9-10, 16-31]'de verilmiştir.

D-) Çevre haritasının sensör füzyonu teknikleri ile gerçekleştirilmesi; Bölüm 5'de mobil robotların içinde çalıştıkları ortamı tanıyabilmeleri ve engellere çarpmadan hedeflerine varabilmeleri amacıyla yapılan çevre haritalama konusunda yapılan çalışmalar aşağıda verildiği gibi 3 kısımda gerçekleştirilmektedir:

- i. 20X20 cm. karelik hücrelere ayrılan ortamda çevre haritalama için kullanılacak sensörlere uygun filtreleme ve füzyon yapılarının (Bayes [36,48], Dempster-Shafer [48,49] vb.) seçilerek, karşılaştırma amaçlı algoritmalarının simule edilecek ve mobil robotlara uygulanabilecek şekilde yazılması,
- ii. Bu çalışmanın diğer bir katkılması olarak çevre haritalama amaçlı geliştirilen ve ismine Ardışıl Temel Bileşenler (kısaca ATB) denilen yöntemin tanıtılması ve mobil robotun dinamik modeli de göz önüne alınarak simule edilip tasarlanması,
- iii. Geliştirilen ATB sensör füzyonu yapısının mobil robot üzerinde test edilmesi ve elde edilen sonuçların bilinen Bayes ve Dempster-Shafer yöntemlerinden sağlanan sonuçlarla görsel olarak karşılaştırılması.

Literatürde bu konuda yer alan çalışmalar [32-51]'de verilmiştir.

E-) Geliştirilen lokalizasyon ve çevre haritalama amaçlı yöntemlerin navigasyon amaçlı birleştirilmesi; Son olarak Bölüm 6'da yapılan çalışmalar aşağıda verilen 3 kısım altında toplanmaktadır:

- i. SDSF lokalizasyon ve ATB çevre haritalama yöntemlerinin birbirini destekleyebilecek bir yapıda birleştirilmesi,

- ii. Birleştirilen bu yapıdan elde edilen verilerin bilinen bir yol planlama yöntemi olan Yapay Potansiyel Alanlar-YPA [57-59] yönteminde kullanılması ve bu sayede robotun verilen hedefe çarpışmalardan uzak yolunun planlanması,
- iii. Planlanan yolu eş-zamanlı üç tekerlek yapısına sahip mobil robotlara izlettirebilmek amaçlı bilinen yöntemlerle [62] hareket planlarının yapılması. Ek-A' da detayları verilen yöntemin Nomad200 mobil robotu üzerindeki uygulamalarına da yer verilmektedir.

Literatürde bu konuda yer alan çalışmalar [32, 52-62]'de verilmiştir.

1.3 Çalışma Olanakları

Laboratuvar ortamında bu çalışmayı gerçekleştirebilmek üzere çeşitli alt yapı imkanları mevcuttur. Bunlar 2 grupta belirlenebilir :

A-) Mevcut Donanım: Nomad200 Mobil Robotu [6], Sensus 200 Sabit Sonar Sistemi [6], Sensus 300 Kızılötesi Sistem [6], C100 KVH Pusula Sistemi [7], KVH Fiber-Optik Jiroskop Sistemi [8], Pentium-III 800 MHz. PC, Telsiz Modem ve diğer laboratuvar cihazları.

B-) Mevcut Yazılım: Nomad200 mobil robotu kendi üzerinde Unix işletim sisteminde çalışan bir bilgisayara (Pentium-133) sahiptir. Bu bilgisayar üzerindeki yazılım robot tekerleklerine ve bütün sensörlere kumanda ederken ayrıca dışarıdaki bir bilgisayar ile robot haberleşmesini sağlamaktadır. Robot dışındaki bilgisayar yine Unix işletim sisteminde çalışan ve C dilinde yazılmış, grafiksel ve simülasyon ortamlarını da içeren lisanslı bir Nomad200 yazılım paketine sahiptir.

1.4 Çoklu-Sensör Veri Füzyonuna Genel Bakış

Son yıllarda çoklu-sensör veri füzyonu bir çok araştırmacı tarafından ilgi duyulan bir konu olmuştur. Temel olarak robotik, yapay zeka, endüstri, uzay ve havacılık mühendisliği dallarını içeren geniş bir yelpazede, bu konuda geliştirilen yeni yöntemler ve teoriler periyodik dergilerde bol miktarda yayınlanmaktadır. İlk bakışta bu konunun bu kadar ilgi çekmesi garip gelebilir. Konunun önemini anlatmadan önce, veri-füzyonu araştırmacılarını motive eden etkenler ve sensör verilerini birleştirebilme amaçlı ortaya konan kesin yöntemleri bulabilmek için niçin büyük bir efor sarf edildiği üzerinde durulmalıdır.

İlk kullanıma uygun bilgisayarlar 1940'lı yıllarda geliştirildi. Çünkü o yıllarda elektronik veri işleme ve uygulamalarının kullanımında bir patlama meydana gelmişti. Son yıllarda ise geliştirilen yöntemler ile hesaplama ve muhakeme yapabilen makinelerin boyutları ve maliyetleri aşağı çekilirken yetenekleri arttırılabilmektedir.

Buna rağmen, bazı önemli veri işleme uygulamalarındaki gelişmeler, örneğin otomatik seyrüsefer sistemleri, öngörülenden daha zor gerçekleştirilebilmektedir. Robotik alanındaki gelişmeler de beklenenin gerisinde kalmıştır. Robotik bilimi, gerçek dünya ile doğrudan etkileşime girebilen sistemlerle uğraşır. Çok iyi tanımlanmış veri arabirimleri ile geliştirilen sistemler, ihtiyaç duydukları bilgiye en doğru şekilde erişebilirler. Gerçek dünya iyi tanımlanamadığından otomasyon sistemlerinin beklenmeyen durumlara reaksiyon gösterebilme kabiliyetine ihtiyaç duyulur. Bu sistemler çeşitli sensörler kullanarak bulundukları ortamdaki değişimleri algılamaya ve kendilerini adapte etmeye çalışırlar.

Sensörlere yalın halde kullanıldıkları taktirde, bu bölümde tartışılacak çeşitli sebeplerden dolayı güvenilmeyebilir. Sensörler kendilerini çevreleyen ortamlardan bilgi toplayan elemanlardır. Günümüzde sayılamayacak miktarda sensörler ve bu sensör bilgilerinin kaydedilme yöntemi elde edilebilir. Bunların sınırları ucuz kameralardan trilyonlarca liraya mal olan yeryüzü gözlem uydularına kadar genişletilebilir. Bu kadar çeşitliliğe rağmen, bütün sensörlerde birkaç ortak özellik vardır. Her sensör çeşitli gürültülerin etkisine maruz kalır, dolayısıyla sınırlı bir doğruluğa sahiptir ve bazı koşullar altında doğru çalışmaz.

Günümüzde ikinci nesil robotlar ismi de verilen [1] zeki robotlar, etraflarını algılayabilme ve kendilerine verilen görevleri, çalıştığı ortamda olabilecek değişimlere rağmen başarılı bir şekilde tamamlayabilme yeteneğine sahiptirler. Robotik sistemlerin etrafları ile etkileşim kabiliyetini şu andaki sensör teknolojisi büyük oranda engellemektedir. Bir çok robotik uygulamalarında kullanılan sensörler hala hassas bir doğruluğa sahip olmayıp güvenilirlikleri tartışılmaktadır. Daha özele indirgeyecek olursak, modern endüstriyel ortamlarda robotların hareket esnekliği taşıma, istifleme, üretime ait parçaların toplanması v.b. amaçlar için gereklidir. Bu aktivitelerin gerçekleşmesi için robotlar bulundukları ortamdaki birçok makinenin ve onlara bağlı sistemlerin koordinasyonuna ihtiyaç duyarlar. Hareketli robotlar kendilerine verilen görevler doğrultusunda planlayacakları yolun ve diğer makinelerle ihtiyaç duyulan koordinasyonun sağlanmasında doğrudan doğruya kullandıkları sensörlerin kalitesine bağımlıdır.

Sensör füzyonu sensörlerin teknolojidten kaynaklanan eksikliklerinin üstesinden gelmeye çalışır. Birkaç bağımsız sensörden yapılacak okumalar, bir sistemin tek bir sensörden göreceği zararı azaltabilir. Birkaç farklı yapıya sahip sensörlerden yapılacak okumaları birleştirme, kullanıldığı sisteme diğer sensör yapılarındakilerden daha doğru bilgi verebilecektir. Aynı tip birkaç sensörden yapılacak okumaları birleştirme ise kullanıldığı sisteme gürültüye daha az duyarlı bilgi sağlayacaktır.

Aynı tip sensör okumalarının birleştirilmesinin sağladığı avantajlara ek olarak, karar verme sürecinin, aynı duruma birkaç farklı açıdan bakılarak kolaylaştırılmasını sayabiliriz. Hayvanlar genellikle karar verme işlemini birkaç farklı duyusuna dayanarak gerçekleştirir. Örneğin, engerek yılanları avlarının konumlarını gözleri ve kafalarının iki yanında yer alan ısı sensörleri ile tayin ederler [2]. Bir insanın yemek yerken kaç duyusunu kullandığını göz önüne alabiliriz. Otomatik sistemler bulundukları ortamlara akla uygun reaksiyonlar gösterebilmek için sık olarak birkaç ayrı kaynaktan gelen bilgilerin birleştirilmesine ihtiyaç duyarlar.

Sensör füzyonu ve tümleştirilmesindeki diğer bir problem ise bilginin aşırı yüklenmesidir. Doğru bir karar ancak yerinde ve sağlıklı bir bilgi ile mümkündür. Bir karar verme sürecine erişmek için ihtiyaç duyulan zaman doğru orantılı olarak bilginin elde edilmesi için gereken zaman ile hızla artar. Sensör füzyonu ve tümleştirilmesi birçok bağımsız kaynaktan gelen okumalar arasından en iyisini seçmek ve onlar arasındaki uyumsuzlukları ortadan kaldırmak için gereklidir. Sensör füzyonuyla bir ortamın farklı açılardan görünüşlerini uygun tek bir yapıda birleştirme hedeflenir.

Şayet sensör füzyonu uygun bir şekilde uygulanırsa sınırlı doğruluk ve güvenilirliğe sahip birçok bağımsız kaynaktan gelen bilgileri, doğruluğuna güvenilebilir bir bilgiyi verebilecek şekilde birleştirebilir. Bunun için bilimin diğer dallarında olduğu gibi bilgisayar, robotik ve istatistik bilimlerinde de uzmanlık gerektirir.

Göz önüne alınması gereken diğer bir konu da, maliyettir. Birçok sensörü bir sisteme tümleme işleminde çoğu zaman kaliteli veri sağlayan pahalı cihazların yerine ucuz cihazlar kullanılabilir. Ayrıca sensör tümleme teknolojisi de sistem tasarımcılarına bir esneklik sağlayabilir. Böylece sistem içindeki veri trafiği azaltılabilir. Bu teknolojiyi kullanarak yapılacak iyi bir tasarım cihazların üretiminde daha verimli ve esnek sonuçlara yol açabilecektir.

Teknolojik değişim modern dünyanın ayrılmaz bir parçasıdır. Bu teknolojinin bulunduğu ortam ile doğrudan temasında, sistemler güvenilebilir ve doğru veriler elde etmek zorundadır. Bir yalıtık cihazın bu tip bir veriyi sağlama kabiliyeti son

derece sınırlıdır. Bu sınırlama bağımsız sensörleri tümleştirici yollar bulma ve aralarındaki farkları kullanabilme ile ortadan kaldırılabilir.

Sensör füzyonu konusundaki birçok problem, günümüzde diğer sahalarda yapılan araştırmaları da yakından ilgilendirmektedir. Örneğin, sistem güvenilirliği, dağıtık veri işleme, otomatik öğrenme, resim işleme ve aykırı veri yorumlama konularındaki araştırmalar sensör füzyonuna oldukça bağımlıdır. Bu tez ile yapılan çalışmalarda da mobil robotların lokalizasyon ve çevre haritalama amaçlı geliştirilen sensör füzyon tekniklerini kullanabilmesi ve ortaya çıkan problemlerin giderilmesi hedeflenmiştir.



2. EŞ-ZAMANLI ÜÇ TEKERLEK YAPISINDAKİ MOBİL ROBOTLARDA KİNEMATİK MODELLEME

2.1 Giriş

Bu bölümde eşzamanlı sürülebilen üç tekerleğe sahip bir mobil robotun kinematik modelinin çıkartılmasına yönelik yapılan çalışmalar anlatılmıştır. Ortaya çıkarılan kinematik model odyometrik sistemlerin anlaşılabilirliğini sağlarken aynı zamanda odyometrik hataların hesaplanmasında da kullanılmaktadır. Mobil robotların lokalizasyon ve haritalama işlemlerinde aktif olarak kullanılan odyometre okumalarında oluşan sistematik hatalar, kinematik model ile ortaya çıkarılan dönüşüm matrislerinin yapılan fiziksel mesafe ölçümlerinde kullanılmasıyla en aza indirilebilmektedir.

İlk olarak 1986 yılında Muir ve Neuman [3] tekerlekli yapıdaki mobil robotların hareketlerini kinematik denklemlerle formülleştirdiler. Yaptıkları çalışmalarda ağırlıklı olarak robot manipülatör kinematiğinden faydalanmışlardır. Ancak tekerlekli robotların kinematik modellemesi manipülatörlerin modellemelerinden oldukça farklıdır. Robot manipülatörlerin modellenmesi açık-link zinciri şeklinde ele alınırken, bu durum tekerlekli robotlarda tekerleklerin yüzeyle temas ettiği bir kapalı-link zinciri şeklinde ele alınır.

Tekerlekli robotların kinematiği iç ve dış kinematikler şeklinde iki ayrı bölümde inceleyebiliriz. İç kinematikler robot içindeki linkler arasındaki ilişkileri anlatır. Bu ilişkiler özellikle robota dönüş kumandası verilirken oldukça önem kazanır. Dış kinematikler ise robot ile çalıştığı ortam arasındaki ilişkileri anlatır. Örneğin robotun planlanan bir yolu izlemesinde bu ilişki kendini gösterir.

Mobil robot kinematiğini iyi bir şekilde anlatabilmek için tekerlekli robot ile manipülatör kol arasındaki farkları ortaya çıkartmak gerekir. Bunlar sırası ile;

- A. Tekerlekli bir mobil robot, üzerinde hareket ettiđi yüzey ile temas halinde olan bir den fazla tekerleđe sahiptir. Bu nedenle çoklu bir kapalı-link zinciri şeklinde modellenir. Halbuki bir manipölatör kol herhangi bir yere dokunmadıđı sürece (serbest uzayda) tek bir açık-link zincir şeklinde modellenir. Ancak herhangi bir yer ile temas olursa bir kapalı-link zinciri şeklinde modellenir.
- B. Bir robot manipölatörüne ait bütün (eklemler) bir serbestlik derecesi ile, uç noktasının serbestlik derecesi de verilen görev doğrultusunda sınırlandırılır. Halbuki, bir mobil robotun tekerleđi kendisi ile zemin arasındaki temas noktasına göre hem dönebilir hemde ilerleyebilir. Bu sözde temas noktasını “yüksek-seviyede eşlenme” şeklinde tanımlayabiliriz. Yüksek-seviyede eşlenme bir nokta veya hat teması ile sınırlandırılırken düşük-seviyede eşlenme ise ortak yüzeylerin birbiriyle teması ile sınırlandırılır. Prizmatik bir eklem buna örnek olarak verilebilir.
- C. Bir robot manipölatöründeki tüm eklemler, uç noktasının konum ve yönelimini kontrol etmek amacıyla harekete geçirilebilir. Bir mobil robot da ise bazı tekerlekler bütünüyle kumanda edilmeyebilir ve bazı tekerleklerde bazı serbestlik dereceleri ile ortaya çıkmayabilir.
- D. Bir robot manipölatörün uç noktasının yörüngesini kontrol edebilme amacıyla her bir eklemin pozisyonu, hızı ve ivmesi ölçülmelidir. Halbuki bir mobil robotun yörüngesinin kontrolunda her bir tekerleđin her bir serbestlik derecesinin pozisyonunu, hızını ve ivmesini ölçmeye ihtiyaç yoktur.

Üçüncü bölümde verilen odyometrik hataların hesaplanmasında, dördüncü ve beşinci bölümlerde anlatılan lokalizasyon ve harita oluşturma işlemlerinde yer alan odyometrede oluşan sistematik hataların azaltılmasında kinematik modellemede ortaya çıkarılan dönüşüm matrisleri aktif olarak kullanılmaktadır.

2.2 Tanımlar ve Varsayımlar

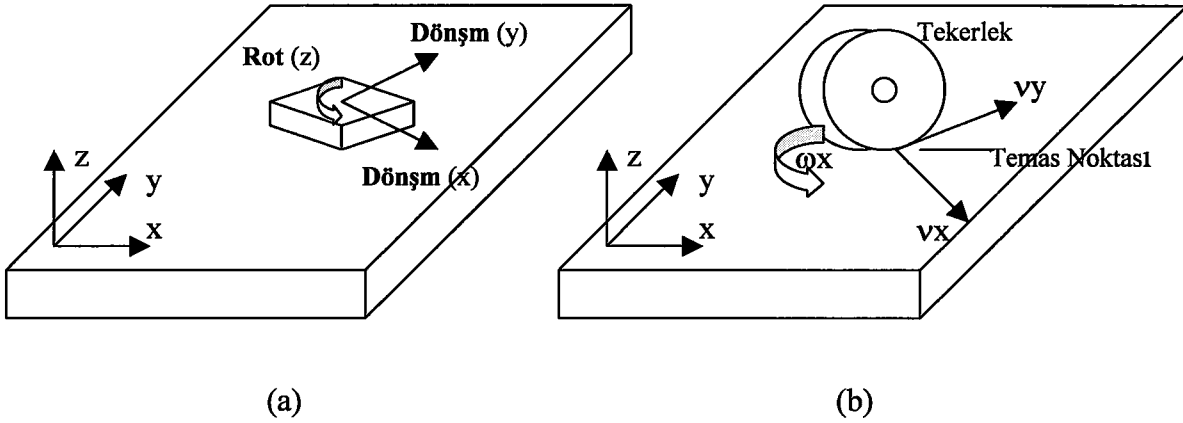
Kendi üzerinde yapılandırılan ve yüzey ile temas halinde olan bir tekerlek sistemi ile hareket edebilme kabiliyetine sahip bir robot “tekerlekli bir mobil robot” dur. Bu

tanımdan yola çıkarak robot gövdesi, ona bağlı tekerlek sistemi ve yüzey arasındaki ilişkileri ortaya çıkartarak tekerlekli mobil robotların kinematik modelini çıkartmaya yönelik bir çerçeve tanımlanmalıdır. Bu amaçla aşağıdaki varsayımlar dikkate alınmıştır [4];

- A. Robot katı bir yapıda oluşturulmalıdır,
- B. Robot, her bir tekerleği için en fazla bir adet döndürme amaçlı bağlantı elemanına (link) sahip olmalıdır,
- C. Bütün tekerlek dönüş eksenleri yere dik olmalıdır,
- D. Tekerleğin temas halinde olduğu yüzey düz bir saha olmalıdır,
- E. Tekerlek ve yüzey arasında hareket esnasında bir kayma olmamalıdır,
- F. Tekerleğin yüzey ile temasında dönme esnasında meydana gelen sürtünme tekerleği dik ekseninde kolayca döndürecek kadar küçük olmalıdır.

2.3 Koordinat Çerçevelerinin Atanması

Çoklu bir kapalı-link zincirinde, koordinat çerçevelerinin atanmasında Denavit-Hartenberg yapısı link bağlantı düzenlerinin açık olmamasından dolayı dönüşüm matrislerinde belirsizliklere yol açar. Bu problemi çözmek için 1986 yılında Muir ve Newman [3] koordinat çerçevelerini, 1971 yılında Sheth-Uicker [5] tarafından geliştirilen yapıda tanımladılar ve temas noktasında her bir tekerleği modellediler. Bu model Şekil 2.1(a)'da gösterildiği gibi iki ilerletme ve bir döndürme eksenine sahiptir. Bu yaklaşımla, tekerlek hareketinin çoklu serbestlik derecesi dönüşüm matrislerinde herhangi bir belirsizliğe yol açmaksızın modellenenir. Şekil 2.1(b)'de de gösterildiği gibi uygun bir tekerlek sadece iki serbestlik derecesine sahiptir. Burada tekerlek de herhangi bir yanıl kayma olmayacağından tekerlek hızının x-bileşeni olan üçüncü serbestlik derecesi sıfır alınmıştır. Tekerlek hızının y-bileşenini ise açısal hızın tekerlek yarıçapı ile çarpılması oluşturur.

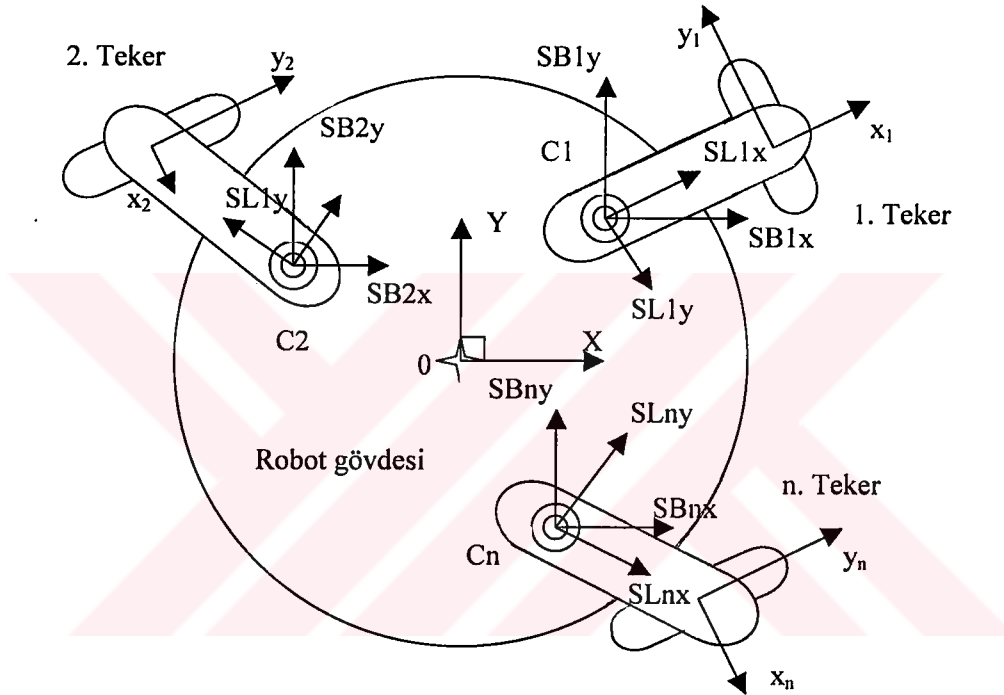


Şekil 2.1 Bir tekerleğin ilerletilmesine yönelik iki serbestlik dereceli ve döndürülmesine yönelik bir serbestlik dereceli yüzeysel modeli. (a) Yüzeysel model, (b) Uygun bir tekerlek modeli.

Sheth-Uicker yapısında, koordinat çerçeveleri her bir link'in her iki ucuna birden atanır. Bu nedenle, her bir eklem yerinde iki koordinat çerçevesi vardır. Bu iki koordinat çerçevesinin iç içe olduğu durumlarda, ortak eklem (birleşim) değişkeni sıfırdır. Tekerlekli bir mobil robotun linkleri Şekil 2.2'den de görülebileceği gibi yüzey, robot gövdesi ve dönüşü sağlayan linklerdir. Bu üç link birbirlerine üç ortak nokta vasıtasıyla bağlanır. Bunlar sırası ile; tekerlek, döndürme eksenini ve robotun orta noktasıdır. Sonuncu ortak bağlaşım noktası fiziksel bir bağlaşım değildir. Fakat robot gövdesi ile yüzey arasındaki ilişki Şekil 2.1(a)'da ki gibi robotun merkezinde konumlandırılmış bir yüzeysel ilişki şeklinde modellenir. Tekerlek ise tekerlek ve yüzey arasındaki temas noktasında konumlandırılmış bir yüzeysel ilişki şeklinde modellenir. Dönme eksenini gerçek bir birleşim yeri olabilir veya bazı mekanik düzenlerin olduğu bir yer de olabilir ve döngüsel bir yüzeysel ilişki şeklinde modellenir. Bütün bu çerçevelerin z-ekseni, modellemede kullanılan eksnlere diktir ve iki-boyutlu analizde dikkate alınmamıştır.

Eğer bir tekerleği Şekil 2.3 ve Şekil 2.4'de gösterildiği gibi tek başına ele alırsak, tekerlek bu haliyle üç link'e, üç ortak bağlantı noktasına ve altı koordinat çerçevesine sahiptir. Bu koordinat çerçevelerinden iki tanesi (C_F ve R_F çerçeveleri) Şekil 2.2'de gösterilmemiştir. C_F ve R_F çerçeveleri anlık olarak çakışan koordinat sistemleridir. Bu gözlem anında hareket eden bir koordinat çerçevesi gibi aynı noktada konumlanmış durağan bir koordinat çerçevesidir. Bu durum anlık olarak bir yüzey

koordinat çerçevesinin hareket eden bir çerçeve gibi aynı noktaya yerleştirilmesiyle eşdeğerdir. Böylece, iki çerçeve arasındaki konumsal dönüşüm sıfırdır, fakat hız ve ivme dönüşümleri değildir. Bu durumda yüzey çerçevesine göre hareketli olan çerçevenin hız ve ivme dönüşümleri aynı değildir. Çünkü mesafe dönüşümünü sıfırlamak hareket eden çerçevenin açısal hızının doğrusal hıza olan katkısını yok eder. Anlık olarak çakışan koordinat çerçeveleri, bu çok-boyutlu hareket eden koordinat sisteminin hız ve ivmesinin aracın bulunduğu konumdan bağımsız olarak hesaplanabilmesine olanak sağladığı için tanıtılmıştır.

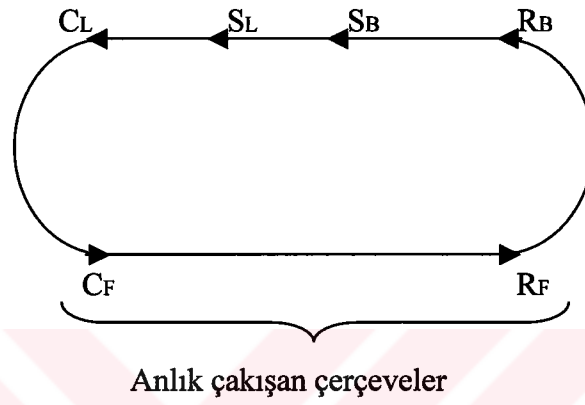


Şekil 2.2 Tekerlekli bir mobil robot için atanan koordinat çerçeveleri.

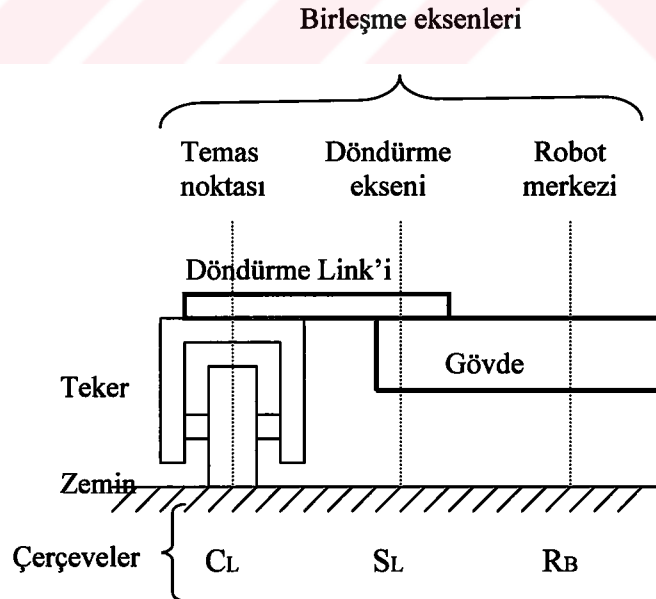
Anlık robot çerçevesi olan R_F gözlem anındaki robot konumundan bağımsız olarak robotun hız ve ivmesini belirtmek için kullanılır. Anlık temas çerçevesi C_F ise tekerlek temas noktasının konumu sensörlerle ölçülemediği için tekerlek hız ve ivmelerinin hesaplanması amacıyla kullanılır. Bu çerçeveler anlık olarak robota değil yüzeye göre sabit tutulur. Yine bu çerçeveler incelendiğinde, onların robot üzerindeki çerçevelerle çakıştığı görülür. R_F çerçevesi R_B çerçevesi ile iç içedir ve C_F çerçevesi de C_B çerçevesi ile iç içedir.

Yüzey koordinat çerçevesi F durağandır ve robotun yaptığı hareketlere referans teşkil eder. Robot çerçevesi R_B ise robotun merkezinde konumlandırılır ve üzerinde

bulunduğu yüzey çerçevesine göre olası robot konumunu ifade edebilme amacıyla kullanılır. Robot gövdesine göre tekerleği hareket ettirerek robot dönüşünü sağlayan bağlantı noktasında iki çerçeve konumlandırılır. Bunlardan biri olan S_B çerçevesi gövdeye bağlıyken diğer çerçeveyi oluşturan S_L ise link'e bağlıdır. Son olarak, C_L çerçevesi tekerlek ve taban yüzeyi arasındaki temas noktasında tekerlek ile bağlantılıdır. Link'e göre tekerlek dönüş eksenini tekerlek ve yüzey arasındaki temas noktasına dik olan bir eksen olduğu varsayılmıştır.



Şekil 2.3 Bir tekerlek için koordinat çerçevelerinin atanması (dönüşüm diyagramı).



Şekil 2.4 Koordinat çerçevelerinin konumları.

2.4 Dönüşüm Matrislerinin Çıkarılması

Bu modelleme tekniğinde dönüşümler manipülatörlerde olduğu gibi türdeş dönüşümlerle ifade edilirler. Tekerlek dönüşlerinin sadece z-ekseninde olması dönüşüm ifadesini basitleştirir. Robot üç boyutlu bir şekle sahip olduğundan, bütün yönlerde aktarmalar söz konusudur. Böylece, tekerlekli bir mobil robot için standart dönüşüm (2.1)'deki eşitlikte verilen matris ile ifade edilir.

$${}^R[T]_N = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & P_x \\ \sin\theta & \cos\theta & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

Burada, $\theta = {}^R[\theta]_N$ iki çerçeve arasındaki dönüş açısıdır.

Yukarıdaki eşitlikte verilen standart dönüşüm matrisinin tersi (2.2)'de verilen eşitlik şeklinde elde edilirken hız ve ivme dönüşüm matrisleri ise gerçekleştirilen türevsel işlemlerle (2.3) ve (2.4)'deki gibi hesaplanabilir.

$${}^R[T]_N^{-1} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & -P_x\cos\theta - P_y\sin\theta \\ -\sin\theta & \cos\theta & 0 & P_x\sin\theta - P_y\cos\theta \\ 0 & 0 & 1 & -P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

$${}^R[v]_N = \frac{d}{dt} {}^R[T]_N = \begin{bmatrix} -\omega \cdot \sin\theta & -\omega \cdot \cos\theta & 0 & v_x \\ \omega \cdot \cos\theta & -\omega \cdot \sin\theta & 0 & v_y \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.3)$$

$${}^R[\alpha]_N = \frac{d}{dt} {}^R[v]_N = \begin{bmatrix} -\alpha \cdot \sin\theta - \omega^2 \cdot \cos\theta & -\alpha \cdot \cos\theta + \omega^2 \cdot \sin\theta & 0 & \alpha_x \\ \alpha \cdot \cos\theta - \omega^2 \cdot \sin\theta & -\alpha \cdot \sin\theta - \omega^2 \cdot \cos\theta & 0 & \alpha_y \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.4)$$

Burada, $\omega = {}^R[\omega]_N$ = açısal hız,

$\alpha = {}^R[\alpha]_N$ = açısal ivme,

$v_x = {}^R[v]_{Nx}$ = doğrusal hızın x-bileşeni,

$\alpha = {}^R[\alpha]_{Nx}$ = doğrusal ivmenin x-bileşenidir.

Herhangi iki koordinat çerçevesi arasındaki dönüşüm basit olarak, her ikisinin arasında yer alan koordinat çerçevelerinin çarpılmasıyla hesaplanabilir. Örneğin Şekil 2.4’de verilen tek bir tekerlek link’i için robotun merkezinden tekerlek temas noktasına kadar olan dönüşüm (2.5)’deki eşitlikle bulunabilir.

$${}^{RB}[T]_{CF} = {}^{RB}[T]_{SB} \cdot {}^{SB}[T]_{SL} \cdot {}^{SL}[T]_{CL} \cdot {}^{CL}[T]_{CF} = {}^{CF}[T]_{RF} \cdot {}^{RF}[T]_{RB} \quad (2.5)$$

Şekil.2.2 yapısındaki bir çok mobil robotun tekerlek döndürme bağlantısı (link) yoktur. Bu tip robotlarda tekerleğin döndürme eksenini ile temas noktasındaki eksen çakışır ve bu eksenler arasındaki dönüşüm birim matrise indirgenir. Böylece eşitlikler basit bir robot konfigürasyonunu tanımlayabilecek şekilde basitleştirilir. Sadece robot konumu dikkate alınacaksa, iki anlık çakışan koordinat çerçevesi C_F ile R_F diğer C_L ve R_B koordinat çerçeveleriyle çakışarak aralarındaki dönüşümü birim matrise indirgerler. Bu haliyle (2.5)’deki eşitlik basitleşerek (2.6)’da verilen konumsal dönüşüm eşitliğini oluşturur.

$${}^{RB}[T]_{CF} = {}^{RB}[T]_{SB} \cdot {}^{SB}[T]_{SL} \cdot {}^{SL}[T]_{CL} = {}^{CF}[T]_{RF} \quad (2.6)$$

Bu eşitlik ile, bir koordinat çerçevesinde (yeni çerçeve) tanımlanan bir robot üzerindeki herhangi bir “p” noktasının konumu diğer bir çerçeveye göre (referans çerçeve) eşitlik (2.7) kullanılarak bulunabilir.

$${}^R[p] = {}^R[T]_N \cdot {}^N[p] \quad (2.7)$$

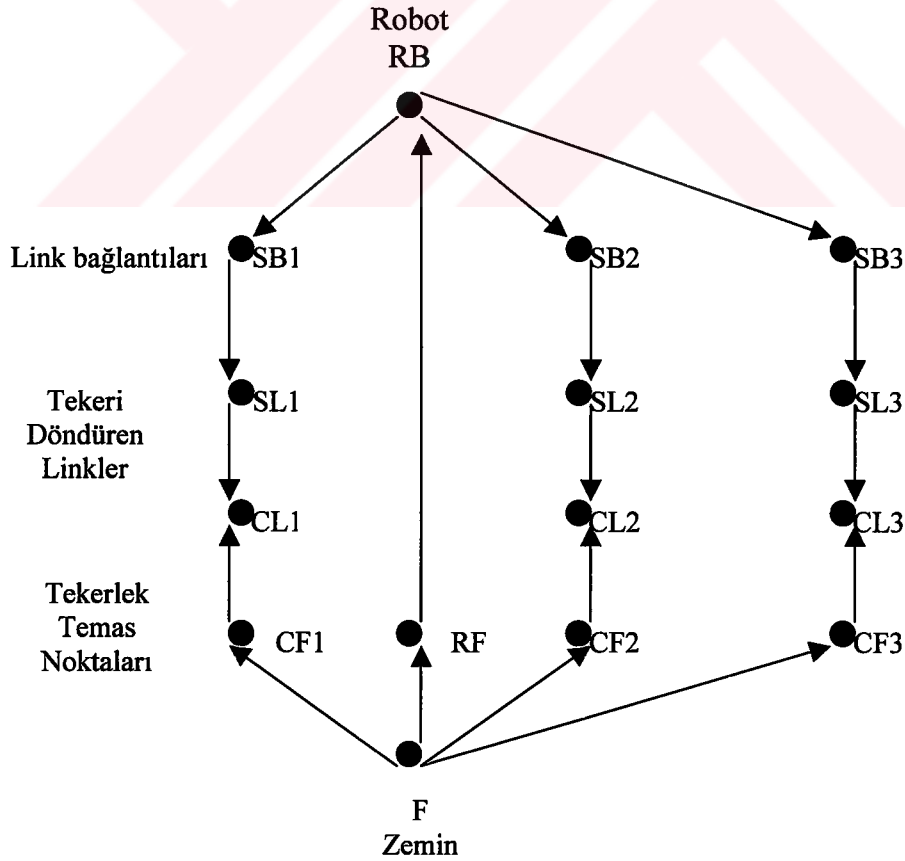
2.5 Eş Zamanlı Üç Tekerlekli Mobil Robotlarda Dönüşüm Matrisleri

Şekil 2.2’de verilen “n” tekerlekli yapıyı üç tekerlekli bir yapıya uyarlırsak, bu yapının kinematik modeli Şekil 2.5’de gösterildiği gibi dört ayrı yola sahip bir paralel dönüşüm grafiği ile ifade edilebilir. Grafikte her bir tekerlek için birer yol ve bir de ortak dönüş yolu gösterilmiştir. Bu grafik çoklu bir kapalı-link zincirinin paralel dönüşüm grafiğidir ve tamamıyla üç-tekerlekli bir mobil robotun iç kinematikini anlatır. Bu mobil robotun dönüşüm eşitlikleri eşitlik (2.8), (2.9) ve (2.10)’da verilmiştir.

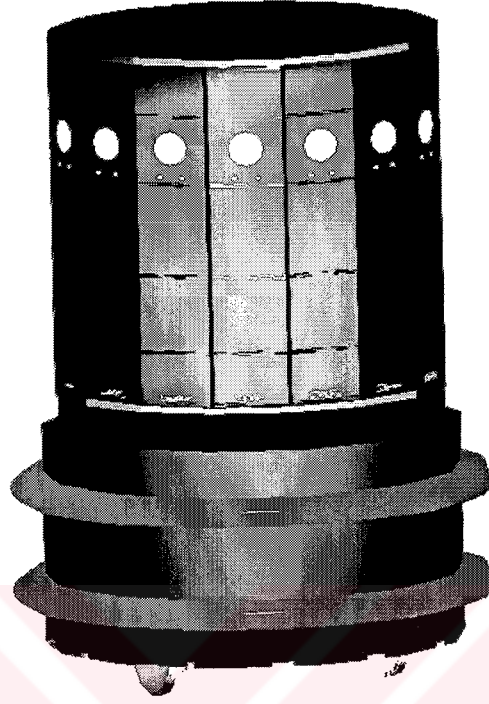
$${}^{RF}[T]_{RB} = {}^{RB}[T]_{SB1} \cdot {}^{SB1}[T]_{SL1} \cdot {}^{SL1}[T]_{CL1} \cdot {}^{CL1}[T]_{CF1} \cdot {}^{CF1}[T]_F \cdot {}^F[T]_{RF} \quad (2.8)$$

$${}^{RF}[T]_{RB} = {}^{RB}[T]_{SB2} \cdot {}^{SB2}[T]_{SL2} \cdot {}^{SL2}[T]_{CL2} \cdot {}^{CL2}[T]_{CF2} \cdot {}^{CF2}[T]_F \cdot {}^F[T]_{RF} \quad (2.9)$$

$${}^{RF}[T]_{RB} = {}^{RB}[T]_{SB3} \cdot {}^{SB3}[T]_{SL3} \cdot {}^{SL3}[T]_{CL3} \cdot {}^{CL3}[T]_{CF3} \cdot {}^{CF3}[T]_F \cdot {}^F[T]_{RF} \quad (2.10)$$



Şekil 2.5 Üç tekerlekli robot için dönüşüm grafiği.

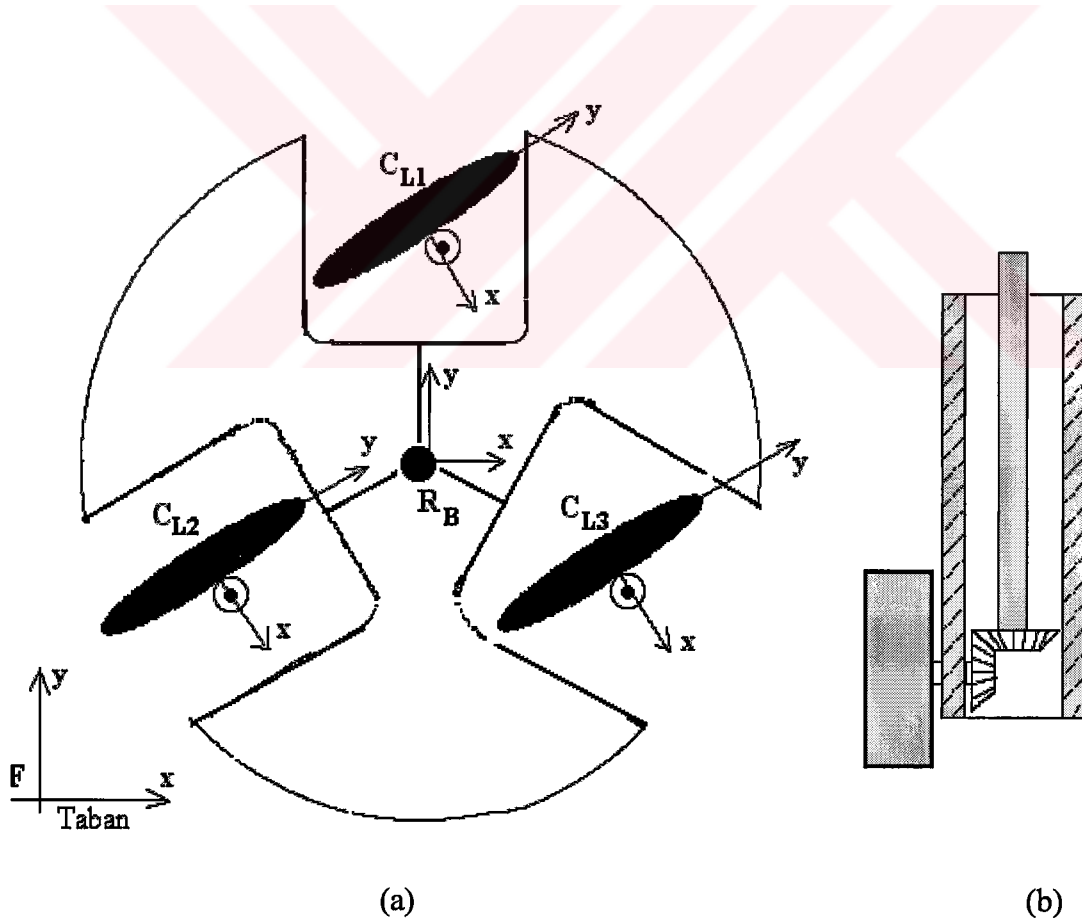


Şekil 2.6 Nomad200 mobil robotu.

Daha sonraki hata analiz çalışmalarında kullanılmak üzere bu modelleme tekniği, yapılan çalışmaların bütün aşamalarında kullanılan ve “Nomadic Technologies Inc.” tarafından üretilen Şekil 2.6’daki Nomad200 [6] mobil robotuna bağlı tekerlek sisteminin modellenmesinde kullanılmıştır. Şekil 2.7(a)’da verildiği gibi Nomad200 mobil robotu eş zamanlı sürülebilir ve bütün yönlerde dönebilir, aynı yapıda, birbirlerine göre 120° ’lik açılarla yerleştirilmiş üç adet tekerleğe sahiptir [6]. Tekerlekler ileri doğru hareket ederken aynı zamanda Şekil 2.7(b)’deki mekanik sistemle sağa ve sola dönme yeteneğine sahiptir. Burada her bir tekerlek aynı yapıda fakat robot referans noktasına göre farklı konumlarda olmasından dolayı kinematik modelleri ayrı ayrı incelenmiştir. Her bir tekerleğin z-ekseninde dönüşünü sağlayan dişli yapısında yaklaşık 3 cm.’lik bir link vardır. Bu nedenle Şekil 2.5’de üç-tekerlekli mobil robotlar için verilen dönüşüm grafiği Nomad200 mobil robotu için de geçerlidir. Robot koordinat çerçevesi R_B nin yer aldığı referans noktası, yükseklik olarak (z-ekseninde) link koordinat çerçeveleri S_B ve S_L ile aynı hizada alınmıştır.

Şekil 2.8(a)'da detaylandırılmış koordinat çerçeveleri verilen birinci tekerleğin robota göre hareketi incelenirse; onun dönüşüm matrisi eşitlik (2.5)'den yararlanılarak aşağıdaki gibi hesaplanabilir:

$$\begin{aligned}
 {}^{RB}[T]_{CLI} &= {}^{RB}[T]_{SBI} \cdot {}^{SBI}[T]_{SLI} \cdot {}^{SLI}[T]_{CLI} \\
 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & \ell_a \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(-90^\circ) & -\sin(-90^\circ) & 0 & 0 \\ \sin(-90^\circ) & \cos(-90^\circ) & 0 & \ell_b \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} \sin\theta & \cos\theta & 0 & -\ell_b \cdot \sin\theta \\ -\cos\theta & \sin\theta & 0 & \ell_a + \ell_b \cdot \cos\theta \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.11)
 \end{aligned}$$



Şekil 2.7 (a) Nomad200 tekerleklerinin z-eksenine göre (üstten bakıldığında), koordinat çerçeveleri, (b) Tekerlekleri sağa ve sola çevirebilen dişli mekanik sistem.

Aynı şekilde Şekil 2.8(b) ve (c)'de koordinat çerçeveleri gösterilen ikinci ve üçüncü tekerlek için hesaplanan dönüşüm matrisleri (2.12) ve (2.13)'de verilmiştir. Tekerleklerin robot merkezi R_B ye olan uzaklıkları Şekil 2.8(d)'de sadece birinci tekerlek üzerinden x-eksenine göre (yandan bakıldığında) gösterilmiştir. Diğer tekerleklerin robot merkezine olan uzaklıkları da aynıdır.

$$\begin{aligned}
 {}^{RB}[T]_{CL2} &= {}^{RB}[T]_{SB2} \cdot {}^{SB2}[T]_{SL2} \cdot {}^{SL2}[T]_{CL2} \\
 &= \begin{bmatrix} 1 & 0 & 0 & -\ell_c \\ 0 & 1 & 0 & -\ell_a \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(-90^\circ) & -\sin(-90^\circ) & 0 & 0 \\ \sin(-90^\circ) & \cos(-90^\circ) & 0 & \ell_b \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} \sin\theta & \cos\theta & 0 & -\ell_c - \ell_b \cdot \sin\theta \\ -\cos\theta & \sin\theta & 0 & -\ell_a + \ell_b \cdot \cos\theta \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.12)
 \end{aligned}$$

$$\begin{aligned}
 {}^{RB}[T]_{CL3} &= {}^{RB}[T]_{SB3} \cdot {}^{SB3}[T]_{SL3} \cdot {}^{SL3}[T]_{CL3} \\
 &= \begin{bmatrix} 1 & 0 & 0 & \ell_c \\ 0 & 1 & 0 & -\ell_a \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(-90^\circ) & -\sin(-90^\circ) & 0 & 0 \\ \sin(-90^\circ) & \cos(-90^\circ) & 0 & \ell_b \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} \sin\theta & \cos\theta & 0 & \ell_c - \ell_b \cdot \sin\theta \\ -\cos\theta & \sin\theta & 0 & -\ell_a + \ell_b \cdot \cos\theta \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.13)
 \end{aligned}$$

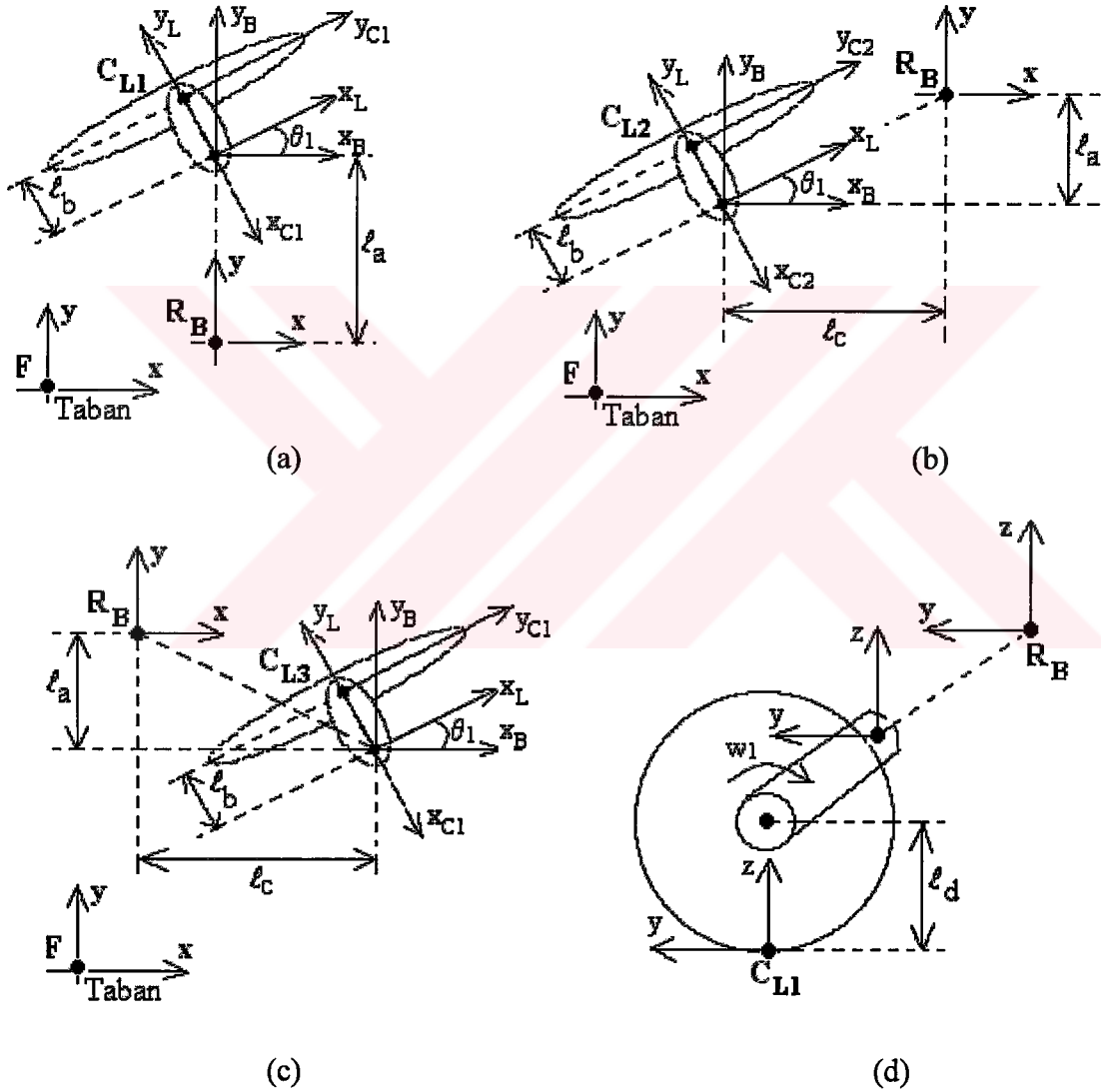
Örneğin $\theta = 90^\circ$ ise tekerlek temas noktasında C_L koordinat çerçevesi ile robot tekerlek sisteminin orta noktasında yer alan R_B robot koordinat çerçevesi aynı yönlere bakan “x” ve “y” eksenlerine sahip olurlar. Bu durumda her bir tekerleğin temas noktası R_B çerçevesine göre aşağıdaki dönüşüm matrisleri ile tanımlanabilir:

$\theta = 90^\circ$ için;

$${}^{RB}[T]_{CL1} = \begin{bmatrix} 1 & 0 & 0 & -\ell_b \\ 0 & 1 & 0 & \ell_a \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$${}^{RB}[T]_{CL2} = \begin{bmatrix} 1 & 0 & 0 & -\ell_c - \ell_b \\ 0 & 1 & 0 & -\ell_a \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$${}^{RB}[T]_{CL3} = \begin{bmatrix} 1 & 0 & 0 & \ell_c - \ell_b \\ 0 & 1 & 0 & -\ell_a \\ 0 & 0 & 1 & -\ell_d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Şekil 2.8 (a) Üstten bakıldığında birinci tekerleğe ait koordinat çerçeveleri, (b) Üstten bakıldığında ikinci tekerleğe ait koordinat çerçeveleri, (c) Üstten bakıldığında üçüncü tekerleğe ait koordinat çerçeveleri, (d) Yandan bakıldığında birinci tekerleğe ait koordinat çerçeveleri.

Robot referans noktasına göre koordinat çerçeveleri arasında yapılan dönüşümlere ilave olarak, robot üzerindeki herhangi bir “ p ” noktasının hareket esnasındaki hız ve ivmesi de robot referans noktası R ye göre (2.14) ve (2.15)’deki eşitlikler kullanılarak hesaplanabilir.

$${}^R[v]_P = \frac{d}{dt} \left\{ {}^R[T]_N \right\} \cdot {}^N[P] \quad (2.14)$$

$${}^R[a]_P = \frac{d^2}{dt^2} \left\{ {}^R[T]_N \right\} \cdot {}^N[P] \quad (2.15)$$

Yapılan çalışmalarda sadece koordinat dönüşümlerinin kullanılması yeterli olacağından kinematik modellemeler (2.11), (2.12) ve (2.13)’de verilen dönüşüm matrisleri ile sınırlandırılmıştır. Tekerlek hareketlerine duyarlı sensörlerle gerçekleştirilebilecek bir odyometrik sistemde robot hız ve ivmesinin hesaplanabilmesine yönelik yapılacak diğer çalışmalarda (2.14) ve (2.15) denklemleri kullanılabilir. Yapılan çalışmada ise odyometrik bilgiler robot üzerinde kurulu sistemden hazır alındığından sadece sistematik odyometre hatalarının ortaya çıkarılmasında dönüşüm matrislerinden yararlanılmıştır.

2.6 Sonuç

Bu bölümde yapılan çalışmada üç-tekerlekli mobil robotlar üzerindeki her bir tekerleğe bağlı olarak robot konumunun bulunmasını sağlayan ve eşitlik (2.8), (2.9) ve (2.10)’da verilen kinematik model ortaya çıkarılmıştır. Yine bu model kullanılarak, çalışmaların üzerinde test edildiği Nomad200 mobil robotunun kinematik modellemesinin bir parçası olan koordinatsal dönüşüm matrisleri eşitlik (2.11), (2.12) ve (2.13)’deki şekilde hesaplanmıştır. Bu verilen eşitliklerden herhangi biri robot konumunun ortaya çıkarılmasında yeterlidir. Çalışmada kullanılan odyometrik bilgiler robot üzerinden hazır alındığından oluşabilecek odyometrik hataların ortaya çıkarılabilmesinde bu dönüşüm matrisleri kullanılmıştır. Burada yapılan çalışmaların kullanıldığı odyometrik hataların ölçülmesine yönelik çalışmalar ise Bölüm 3.1’de detaylı bir şekilde verilmektedir.

3. SENSÖRLER VE MODELLERİ

3.1 Giriş

Bu bölümde çevre haritalama ve bu çevre içindeki robot konum ve yönelimini bulmaya yönelik kullanılan sensörler ve sensör sistemleri ile bunların modellemeleri üzerine yapılan çalışmalar anlatılmaktadır. Tablo 3.1’de verildiği gibi dördüncü bölümde lokalizasyon konusunda yapılan çalışmalarda robot üzerindeki odyometrik sensör ünitesi ve ultrasonik sensörler ile robota sonradan ilave edilen manyetik pusula ve fiber-optik jiroskop kullanılmıştır. Ayrıca 5. bölümde verilen çevre haritalama üzerine yapılan çalışmalarda yine ultrasonik ve kızılötesi sensörler aktif olarak kullanılmıştır. Lokalizasyon ve çevre haritalama konularında geliştirilen yöntemlerin sağlıklı çalışabilmeleri için kullanılan sensörlerin fiziksel yapıları ve özellikleri anlaşılabilir olmalıdır. Örneğin, ultrasonik ve kızılötesi sensörler gibi ortama bir sinyal yayan sensörlerin ışıma örüntülerinin ve yansıma özelliklerinin bilinmesi gereklidir. Bu amaçla, yapılan çalışmaların bütün aşamalarında kullanılan sensörlerin matematiksel modelleri deneysel yollardan çıkartılarak, yapısal özellikleri tablolar halinde verilmiştir. Ayrıca bu sensörler için daha detaylı bilgiler [7, 8]’den elde edilebilir.

Tablo 3.1 Amaçlarına göre kullanılan sensör üniteleri

Mevcut Sensörler	Lokalizasyon Amaçlı	Çevre Haritalama Amaçlı
Odyometre (odo)	√	
Ultrasonik (ult)	√	√
Kızılötesi (kzl)		√
Pusula (pus)	√	
Jiroskop (jir)	√	

İlk olarak Bölüm 3.2’de robot üzerindeki odyometrik sistem ve modeli tanıtılarak bu sistem üzerinde meydana gelen hataların tespiti amacıyla yapılan deneysel çalışmalara ve sonuçlarına yer verilmiştir. Sonra Bölüm 3.3’de yapılan çalışmaların bütün aşamalarında yaygın bir şekilde kullanılan ultrasonik sensörler ve modeli detaylı bir şekilde tanıtılmış ve bu sensörlerin konumsal ve açısal hata analizleri üzerine yapılan testlere geniş yer verilmiştir. Ayrıca bu sensörlerin çevre haritalama amaçlı kullanımından da bahsedilmiştir. Bölüm 3.4 ise sadece çevre haritalama amaçlı kullanılan kızılötesi sensörler tanıtılarak çevre modeli ortaya konmaktadır. Daha sonra Bölüm 3.5 ve 3.6’da lokalizasyon amaçlı kullanılan bir manyetik pusula ve fiber-optik jiroskop tanıtılarak teknik özelliklerinden detaylı bir şekilde bahsedilmiştir.

3.2 Odyometrik Sensör Sistemi ve Modeli

Odyometreler mobil robotların lokalizasyonunda çok yaygın bir şekilde kullanılan bir navigasyon (seyrüsefer) yardımcı ünitesidir. Sistem düşük maliyeti ile yüksek örnekleme hızlarında mobil robota nerede bulunduğu dair bir bilgi sağlar. Fakat odyometrelerin temel felsefesi zaman içinde artan mobil robot hareketlerinin tümleştirilmesidir. Bu durum kaçınılmaz bir şekilde meydana gelen konum hatalarının birikmesine sebep olur. Özellikle, robot baş açısının bulunmasında meydana gelen hatalar, büyük konum hatalarına yol açarak, robot tarafından kat edilen mesafe ile orantılı olarak daha da artacaktır. Bu sınırlamalara rağmen, birçok araştırmacı odyometrenin bir robot navigasyon sisteminin en önemli parçası olduğu konusunda uzlaşır. Şayet odyometreden sağlanan bilgi tam bir doğruluğa kavuşturulabilirse, planlanan navigasyon görevleri basit bir şekilde uygulanabilir hale gelecektir. Örneğin 1991’de Cox [9], 1992’de Chenavier ve Crowley [10], daha güvenli bir konum tahmini için Odyometrik verilerin mutlak konum ölçümleriyle birleştirilmesine yönelik yöntemler ileri sürmüşlerdir. Burada mutlak konum ölçümleri bir manyetik pusuladan, aktif ışıldaklardan, GPS (Global Konumlandırma Sistemi) sisteminden, konumu bilinen nesneler kullanılarak veya model eşleştirme yöntemleri kullanılarak sağlanabilir.

Odyometrik sistem basit olarak (3.1)’de verilen eşitlikleri kullanarak verilen “ τ ” örnekleme zamanlarında robotun baş açısını ve konumunu günceller. Odyometrik

sistemden alınan bilgi, mobil robotun konum ve baş açısını (x_r, y_r, θ_r) bildirir ve aşağıdaki denklemlerle tanımlanabilir:

$$\begin{bmatrix} x_r(t+\tau) \\ y_r(t+\tau) \\ \theta_r(t+\tau) \end{bmatrix} = \begin{bmatrix} x_r(t) \\ y_r(t) \\ \theta_r(t) \end{bmatrix} + \tau \cdot \begin{bmatrix} \cos\theta_r(t) & 0 \\ \sin\theta_r(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} \quad (3.1)$$

Burada, τ : örnekleme zamanı, $v(t)$: robotu ileri süren hız, $w(t)$: robot tekerinin sağa veya sola dönüş hızı, $g=[g_1 \ g_2 \ g_3]^T$: odyometrik gürültüler olup sıfır ortalamalı ortak değişinti matrisi Bölüm 3.2.1’de anlatılan yapısal hatalar giderildikten sonra yapılan deneysel çalışmalarda $diag\{7,5 \times 10^{-3}; 7,5 \times 10^{-3}; 10^{-4}\}$ şeklinde bulunmuştur.

Robot tekerlekleri hareket ettikleri düz ve temiz bir yüzey üzerinde doğrusal bir şekilde ilerlediğinde, yapılan ölçümlerin oldukça doğru sonuçlar verdiği gözlenmiştir. Fakat tekerleklerin kayması veya buna benzer nedenlerden dolayı tekerlek dönüşleri sağlıklı olmazsa yanlış okumalar meydana gelir. Ortaya çıkan bu odyometrik okuma hataları 1996’da Borentein ve Feng [11] tarafından sistematik ve sistematik olmayan hatalar şeklinde kategorize edildi. Araştırmacılara göre sistematik hatalar daha çok robot tekerlek düzeneğindeki çok ufak yapısal dengesizliklerden kaynaklanmaktadır. Örneğin, tam olarak birbirine eşit olmayan tekerlek çapları veya tekerleklerin bağlı olduğu tabandaki buna benzer ufak kinematik yapısal kusurlar sistematik hataları ortaya çıkartır. Sistematik olmayan hatalar ise tekerlekle hareket ettiği yüzey arasında meydana gelebilecek kayma, çarpma veya buna benzer etkilerden kaynaklanmaktadır.

Gerçekleştirilen çalışmada [12]’de verilen hata ölçme yöntemlerinden faydalanılarak Nomad200 üzerindeki odyometrik konum hataları belirlenmiştir. Bu konuya ait çalışmalar bir alt bölümde detaylı bir şekilde verilmektedir. Sistemde meydana gelebilecek Δt zamana, Δv doğrusal hıza ve $\Delta \omega$ döngüsel hıza bağlı gürültülerde dikkate alınırsa yapılan her bir ölçüm $t \pm \Delta t$ zaman sınırları içinde olmalıdır. Bu konuda yapılan çalışmalar Bölüm 4.4’de detaylı olarak anlatılmıştır. Sistem üzerinde oluşan hataları en aza indirebilmek amacıyla robot üzerindeki odyometrik ünite Bölüm 4.2’de anlatılan konumu bilinen nesneler ile yapılan ölçümlerle desteklenmiştir.

3.2.1 Odyometrik Hataların Ölçülmesi

Mobil robotlarda karşılaşılan zorluklardan biride odyometrik hataların niceliksel ölçümlerinin kolay olmayışıdır. Odyometrik hataların nicelikselleştirilmesi için gereken iyi tanımlanmış bir ölçme prosedürünün eksikliği mobil robotlarda zayıf kalibrasyona ve dolayısıyla odyometreden sağlanan bilgilerin doğruluğu konusunda şüphelere yol açar. Bu problemi çözebilmek için 1994 yılında Borenstein ve Feng [12] sistematik ve bir dereceye kadarda sistematik olmayan odyometre hatalarını niceliksel olarak ölçebilen UMBmark yöntemini geliştirdiler. UMBmark yöntemine benzer şekilde daha basit yapıda geliştirilen bir yöntemi Nomad200 mobil robotunda uygulayabilmek için, robota Şekil 3.1’de verildiği gibi 4x4 metrelik bir kare yol ilk önce saat yönünde (sy) köşelerde 90°’lik sağa dönüşler şeklinde beş defa, daha sonrada saat yönünün tersi istikamette (syt) köşelerde 90°’lik sola dönüşler şeklinde beş defa izlettirilmiştir. Bu izletme sonucunda varış noktasında okunan odyometrik konum, ölçülen gerçek konum ve ikisinin farkıyla ortaya çıkarılan konum hataları ile görelî konum hataları sürekli sağa dönüşler için Tablo 3.2’de ve sürekli sola dönüşler için de Tablo 3.3’de izlenen yönlerle göre verilmiştir. Her bir turlama sonucu elde edilen odyometrik konum bilgileri ile gerçek konum arasındaki Δx ve Δy konumsal hatalar sağa dönüşler için Şekil 3.2’de sola dönüşler için de Şekil 3.3’de gösterilmiştir.

Daha sonra odyometrik sistemde yapılacak iyileştirmelere bir karşılaştırma sağlayabilmek amacıyla yüzde olarak $\% \Delta \rho / \rho$ şeklinde görelî konum hataları hesaplanmıştır. Burada $\Delta \rho$ her bir tur sırasında x ve y koordinatlarında ortaya çıkan konum hatalarının karelerinin karekökü, ρ ise robotun yol aldığı turun uzunluğudur. Örneğin Tablo 3.2’de saat yönündeki 1. turda hesaplanan konum hataları $\Delta x_{sy}=31.8$ cm., $\Delta y_{sy}=12.0$ cm.’dir. Buradan $\Delta \rho = \sqrt{(31.8^2 + 12.0^2)} \cong 33.99$ cm. şeklinde bulunabilir. Birinci tur sonunda kat edilen mesafe de $\rho=1600$ cm. olduğundan bu tur için görelî konum hatası $\% \Delta \rho / \rho = \% 2.12$ olarak hesaplanabilir. Tablo 3.2 ve Tablo 3.3’e dikkat edildiğinde robotun sürekli sağa dönüşlerle gerçekleştirdiği saat yönündeki (sy) turlarda görelî konum hataları $\% 2.07$ ile $\% 2.15$ arasında toplanırken, sürekli sola dönüşlerle gerçekleştirilen saat yönünün tersi (syt) istikametindeki turlarda ise $\% 1.18$ ile $\% 1.30$ arasında oluşmaktadır.

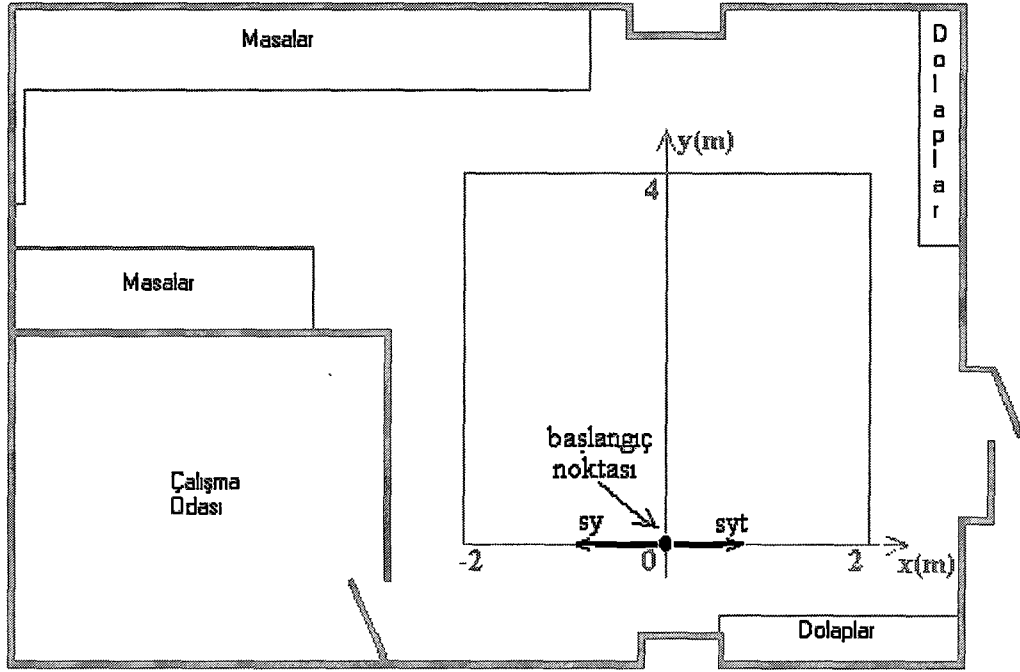
Şekil 3.2 ve Şekil 3.3'den de görülebileceği gibi izlenen yolun saat yönünde ve tersi istikametteki yönde oluşuna göre ortaya çıkan konumsal hatalar iki ayrı bölgede kümelenmişlerdir. İki ayrı kümede oluşan bu hata dağılımı ağırlıklı olarak sistematik hatalar sonucunda ortaya çıkmaktadır. Yapılan deneysel çalışmalar sonucunda, düz beton zeminlerde hareket eden kalibre edilmemiş bir araçta oluşan toplam odometre hatalarına sistematik hataların katkısının sistematik olmayan hataların katkısından daha fazla olduğu görülmüştür. Çalışılan zemine döşenen kare biçimindeki beton parke taşları arasında yer yer çok ufak seviye farklarının olması robot dönüşlerini etkileyen sistematik hataların artmasına sebep olmaktadır. Yapılan hata hesaplarında sistematik olmayan hatalar sistematik hatalara göre ihmal edilebilir düzeyde olduğundan sadece sistematik hatalar ele alınmıştır.

Tablo 3.2 Saat yönündeki (sy) odyometrik ölçüm hataları

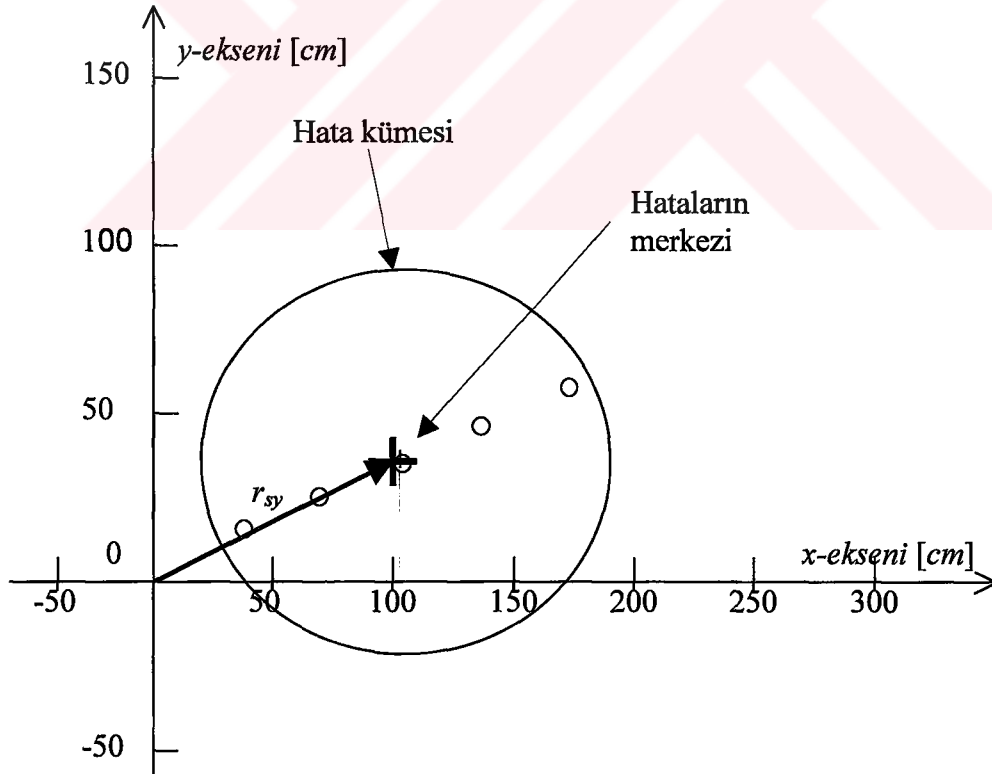
Tur Sayısı	Odyometrik Ölçümler [cm]		Gerçek Ölçümler [cm]		Konum Hataları [cm]		Görelî Konum Hataları [%]
	x_{odo}	y_{odo}	x_{ger}	y_{ger}	Δx_{sy}	Δy_{sy}	$\Delta\rho/\rho$
1	21.8	1.0	-10.0	-11.0	31.8	12.0	% 2.12
2	45.7	1.3	-17.0	-20.0	62.7	21.3	% 2.07
3	71.0	1.7	-25.0	-30.0	96.0	31.7	% 2.11
4	95.0	2.2	-33.0	-40.5	128.0	42.7	% 2.11
5	121.0	2.8	-42.5	-51.0	163.5	53.8	% 2.15

Tablo 3.3 Saat yönü tersindeki (syt) odyometrik ölçüm hataları

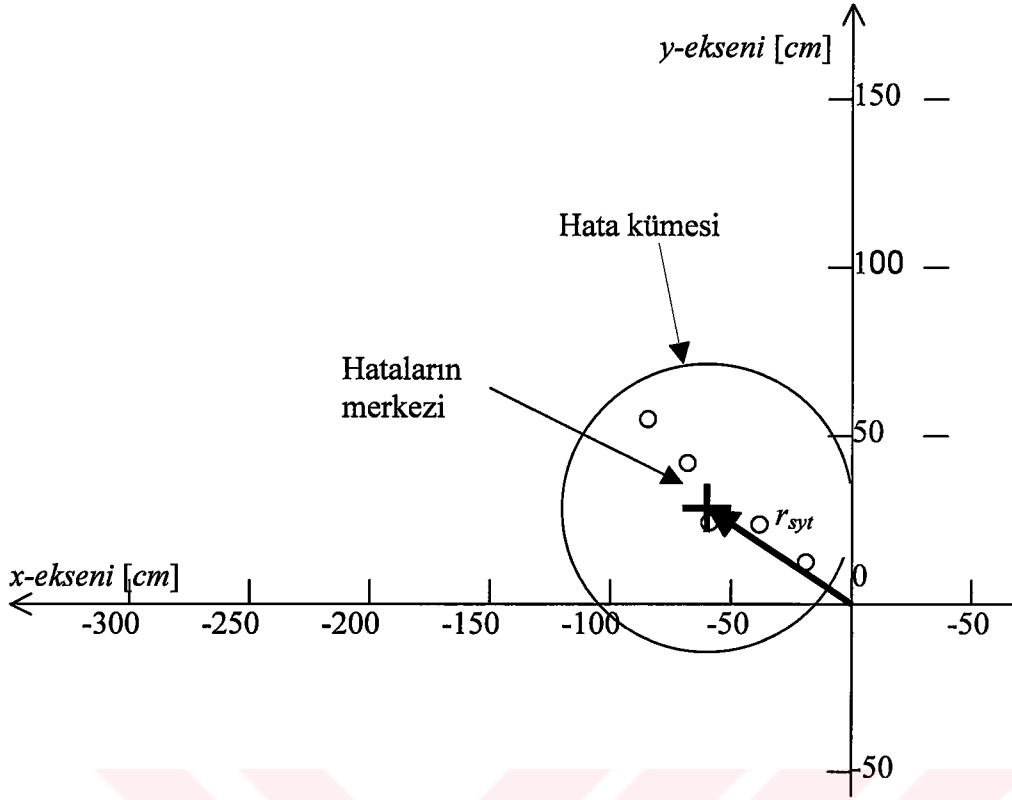
Tur Sayısı	Odyometrik Ölçümler [cm]		Gerçek Ölçümler [cm]		Konum Hataları [cm]		Görelî Konum Hataları [%]
	x_{odo}	y_{odo}	x_{ger}	y_{ger}	Δx_{syt}	Δy_{syt}	$\Delta\rho/\rho$
1	27.0	5.0	43.0	-6.0	-16.0	11.0	% 1.21
2	52.0	6.2	87.0	-16.0	-35.0	22.2	% 1.30
3	75.0	10.0	130.0	-13.0	-55.0	23.0	% 1.24
4	96.0	13.5	160.0	-27.0	-64.0	40.5	% 1.18
5	120.0	18.0	200.0	-35.0	-80.0	53.0	% 1.20



Şekil 3.1 Odyometrik hata ölçümleri için laboratuvar ortamında planlanan yol.



Şekil 3.2 Saat yönü istikametinde (sürekli sağa dönüşlerle) ortaya çıkan odyometrik hatalar.



Şekil 3.3 Saat yönünün tersi istikametinde (sürekli sola dönüşlerle) ortaya çıkan odyometrik hatalar.

Dikdörtgen yolun sy ve syt yönlerinde izlenmesi sonucunda ortaya çıkan ağırlık merkezlerinin asimetric oluşu A-tipi ve B-tipi [11] ismi verilen iki tip sistematik hatadan kaynaklanmaktadır. A-tipi hatalar robot yönündeki hatalar olup robotun hem sy hem de syt istikametlerinde kare yolu izlerken yaptığı teker dönüşlerinin miktarlarını azaltır veya çoğaltır. Diğer tarafta ise B-tipi hatalar yolun sy doğrultusunda izlenmesi sırasında tekerlerin dönüş miktarını azaltırken syt istikametinde artırır veya tam tersi olarak sy de artırırken syt de azaltır. A-tipi hataların başlıca kaynağı robot teker sistemini oluşturan mekanik aksamlardan ortaya çıkan belirsizliklerdir. B-tipi hataların kaynağı ise teker çapları arasındaki imalattan kaynaklanan ufak farklılıklardır.

Nomad200 mobil robotu üzerinde yapılan testler sonucunda, robota ilerleme ve dönme hareketini eş-zamanlı sağlayan her bir teker çapının mükemmel bir doğrulukta eşit olmasından dolayı B-tipi sistematik hataların A-tipi sistematik hatalara göre ihmal edilebilir olduğu görülmüştür. Bu durum dikkate alınarak odyometrik hataların sadece A-tipi sistematik hatalardan kaynaklandığı varsayılmıştır. Şekil 3.1'deki laboratuvar ortamında robotun sürekli sağa ve sürekli

sola döndürülerek yapılan testlerde, Şekil 3.2 ve Şekil 3.3'deki gibi ortaya çıkan odyometrik konum hatalarının ağırlık merkezleri (3.2a) ve (3.2b)'deki eşitliklerden, bu merkezlerin oluşturduğu yarıçaplar ise (3.3a) ve (3.3b)'deki eşitliklerden hesaplanabilir. Odyometrik doğruluğun ölçütü olan bu yarıçaplardan eşitlik (3.4)'de verildiği gibi büyük olan aynı zamanda odyometrik sistemin yaptığı hata olarak ele alınmaktadır [11].

$$(\Delta x_r)_{sy/syt} = \frac{1}{n} \sum_{i=1}^n \Delta x(i)_{sy/syt} \quad (3.2a)$$

$$(\Delta y_r)_{sy/syt} = \frac{1}{n} \sum_{i=1}^n \Delta y(i)_{sy/syt} \quad (3.2b)$$

Burada n : robotun yaptığı tur miktarı (yapılan test için $n=5$).

$$r_{sy} = \sqrt{(\Delta x_r)_{sy}^2 + (\Delta y_r)_{sy}^2} \quad (3.3a)$$

$$r_{syt} = \sqrt{(\Delta x_r)_{syt}^2 + (\Delta y_r)_{syt}^2} \quad (3.3b)$$

$$E_{maks,odo} = maks(r_{sy}; r_{syt}) \quad (3.4)$$

Yapılan testlerde r_{sy} ve r_{syt} değerleri yaklaşık 102 cm. ve 58.3 cm. olarak hesaplanmıştır. Bu hesaplamalar sonucunda bulunan odyometrik hata $E_{maks,odo}=102$ cm. dir.

Şekil 3.1'deki laboratuvar ortamında Nomad200 mobil robotu ile yapılan testlerde A-tipi sistematik hataların her bir 90^0 'lik sağa dönüşlerde (sy) tekerlerin dönüşünü ortalama $4^0 \pm 1^0$, sola dönüşlerde ise ortalama $5^0 \pm 1^0$ azalttığı görülmüştür. Bir başka deyişle 90^0 'lik sağa dönüş kumandası ile robot ancak 85^0 ile 87^0 arasında sağa, 90^0 'lik sola dönüş kumandası ile de 84^0 ile 86^0 arasında sola dönebilmiş, buna karşılık odyometrik sistem de her iki taraf için 90^0 'lik dönüş yaptığını göstermiştir. Bu durumda robot 360^0 'de tamamlaması gerektiği saat yönündeki (sürekli sağa dönerek) bir turunu ortalama 16^0 eksik, saat yönünün tersine (sürekli sola dönerek) olan turunu da ortalama 20^0 eksik tamamladığından Şekil 3.2 ve Şekil 3.3'deki

odyometrik hatalar meydana gelmektedir. Oluşan bu hataları en aza indirebilmek için eşitlik (3.5a) ile sağa, eşitlik (3.5b) ile de sola dönüşler sonrasında odyometrik sistemden alınan robot yönü bilgisi yeniden ayarlanmıştır.

$$(\theta_{odo})_{sy-ger} = (\theta_{odo})_{sy} \cdot \left(1 - \frac{(4^0 \pm 1^0)}{90^0} \right) \quad (3.5a)$$

$$(\theta_{odo})_{syt-ger} = (\theta_{odo})_{syt} \cdot \left(1 - \frac{(5^0 \pm 1^0)}{90^0} \right) \quad (3.5b)$$

Burada:

- $(\theta_{odo})_{sy}$: Tekerlerin sağa dönüşünden sonra okunan odyometrik robot yönü,
- $(\theta_{odo})_{sy-ger}$: Tekerlerin sağa dönüşünden sonra olması gereken gerçek odyometrik robot yönü,
- $(\theta_{odo})_{syt}$: Tekerlerin sola dönüşünden sonra okunan odyometrik robot yönü,
- $(\theta_{odo})_{syt-ger}$: Tekerlerin sola dönüşünden sonra olması gereken gerçek odyometrik robot yönüdür.

A-tipi sistematik hataları azaltan önlem alındıktan sonra gerçekleştirilen sy-istikametindeki testlerin sonuçları Tablo 3.4’de, syt-istikametindekiler ise Tablo 3.5’de verilmiştir. Bu tablolarda yer alan her bir tur sonucunda okunan ayarlanmış odyometrik robot konumu, gerçek robot konumu arasındaki farktan oluşan konum hataları sağa dönüşler (sy) için Şekil 3.4’de, sola dönüşler (syt) için de Şekil 3.5’de işaretlenmiştir. Bu şekillere göre sağa dönüşlerin oluşturduğu hataların ağırlık merkezi $(\Delta x_r)_{sy}=45.8$ cm, $(\Delta y_r)_{sy}=17.2$ cm, sola dönüşlerin ağırlık merkezi ise $(\Delta x_r)_{syt}=-33.7$ cm, $(\Delta y_r)_{syt}=15.1$ cm koordinatlarında oluşmaktadır. Bu değerler kullanılarak hesaplanan odyometrik doğruluğun ölçütü olan yarıçaplar sağa dönüşler için $r_{sy}=48.9$ cm, sola dönüşler içinse $r_{syt}=36.9$ cm dir. Bu durumda eşitlik (3.4) e göre odyometrik sistemin yaptığı hata $E_{maks,odo}=48.9$ cm olacaktır.

Odyometrik sistem eşitlik (3.5a) ve (3.5b) ile ayarlandıktan sonra sistematik odyometre hatalarının $(E_{maks-odo})$ yaklaşık 1.5 ile 2.5 kat arasında azaldığı görülmektedir. Örneğin Şekil 3.2 ve şekil 3.3’deki UMBmark testi sonucunda Nomad200 mobil robotunda gözlenen odyometrik hatalar $(E_{maks-odo} = 102$ cm.) geliştirilen kalibrasyon yöntemi ile Şekil 3.4 ve Şekil 3.5 de gösterildiği gibi en aza indirilebilmektedir $(E_{maks-odo} = 48.9$ cm.).

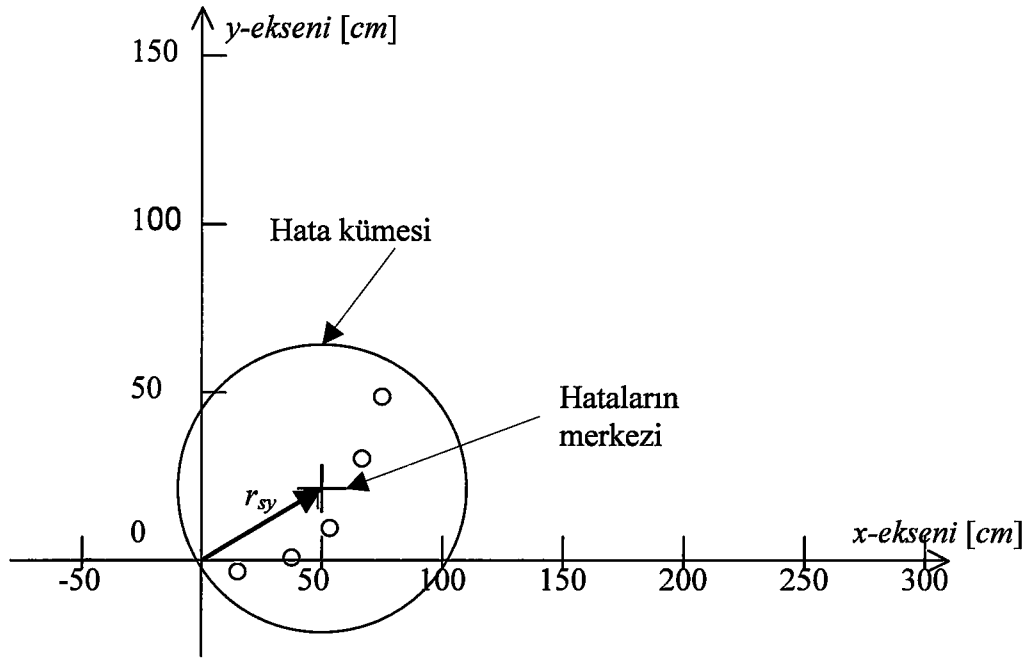
Tablo 3.4 Saat yönündeki (sy) kalibreli odyometrik ölçüm hataları

Tur Sayısı	Kalibreli Odyometrik Ölçümler [cm]		Gerçek Ölçümler [cm]		Konum Hataları [cm]		Görelî Konum Hataları [%]
	x_{odo}	y_{odo}	x_{ger}	y_{ger}	Δx	Δy	$\Delta \rho/\rho$
1	15.2	2.2	2.9	5.5	12.3	-3.3	% 0.80
2	40.1	2.0	5.9	1.1	34.2	0.9	% 1.07
3	60.0	3.1	10.2	-6.9	49.8	10.0	% 1.06
4	81.2	3.8	18.7	-26.7	62.5	30.2	% 1.08
5	100.5	4.7	30.4	-44.2	70.1	48.3	% 1.06

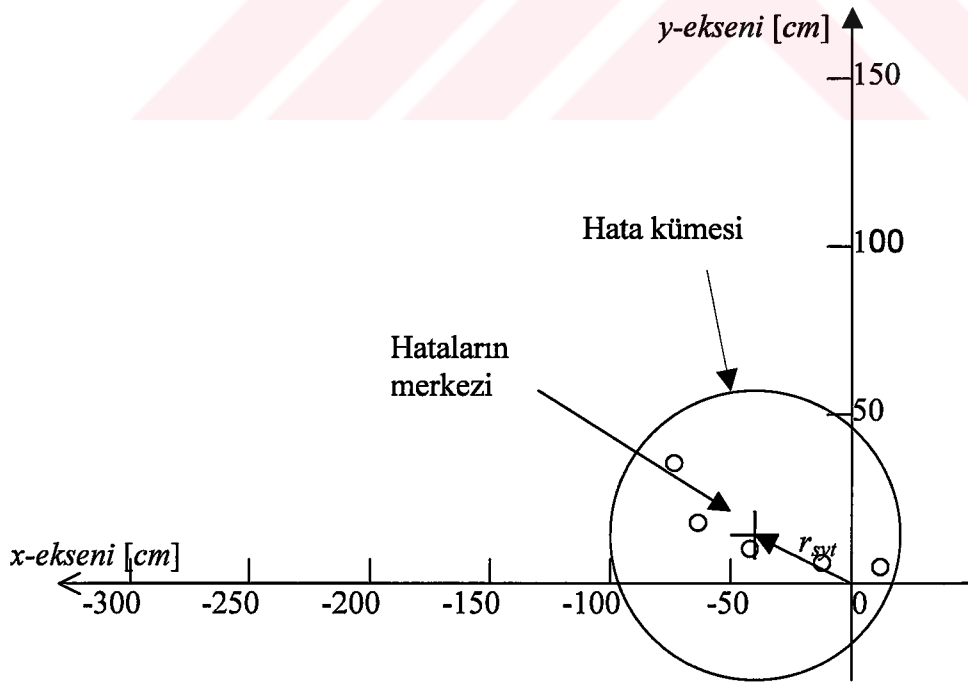
Tablo 3.5 Saat yönü tersindeki (syt) kalibreli odyometrik ölçüm hataları

Tur Sayısı	Kalibreli Odyometrik Ölçümler [cm]		Gerçek Ölçümler [cm]		Konum Hataları [cm]		Görelî Konum Hataları [%]
	x_{odo}	y_{odo}	x_{ger}	y_{ger}	Δx	Δy	$\Delta \rho/\rho$
1	21.5	3.0	8.4	-2.0	13.1	5.0	% 0.88
2	45.3	4.5	56.2	-1.6	-10.9	6.1	% 0.39
3	55.5	6.2	95.3	-4.1	-39.8	10.3	% 0.86
4	73.2	9.8	133.7	-8.3	-60.5	18.1	% 0.99
5	98.1	15.0	168.4	-20.8	-70.3	35.8	% 0.99

Odyometreden okunan robot yönü bilgisinin eşitlik (3.5a) ve (3.5b) ile yeniden ayarlanması, görelî konum hatalarında da bir azalmaya yol açmıştır. Tablo 3.4 ve Tablo 3.5’de görülebileceği gibi yapılan iyileştirme sonucunda sürekli sağa dönüşlerle gerçekleştirilen turlarda (sy) görelî konum hataları % 0.80 ile % 1.08 arasına, sürekli sola dönüşlerle yapılan turlarda (syt) ise % 0.39 ile % 0.99 arasına düşürülmüştür.



Şekil 3.4 Saat yönü istikametinde (sürekli sağa dönüşlerle) ortaya çıkan ayarlanmış odyometrik hatalar.



Şekil 3.5 Saat yönünün tersi istikametinde (sürekli sola dönüşlerle) ortaya çıkan ayarlanmış odyometrik hatalar.

3.3 Ultrasonik Sensörler ve Modeli

Mobil robotlarda lokalizasyon ve çevre haritalama konularında çalışan bir çok araştırmacı mesafe ölçümlerinde ucuz ve kolay işlenebilir olmalarından dolayı ultrasonik sensörleri tercih etmişlerdir. Kullanılan Nomad200 mobil robotu üzerinde de bu türden bir mesafe okuyucu bulunduğundan lokalizasyon ve çevre haritalama konusunda yapılan bütün çalışmalarda ultrasonik sensörler aktif olarak yer almıştır. Bu tür sensörler kendi frekansındaki darbeleri dış ortama ilettikten sonra ortamdan geriye yansıyan sinyalleri alabilmek için alıcı moduna geçerler. Mesafe ölçümünde, ortam sıcaklığına göre değişen bir ses hızında gönderilen sinyalin geri dönüş zamanı dikkate alınarak nesnelerin robota olan uzaklıkları eşitlik (3.6) kullanılarak bulunabilir.

$$d_u = V_s \left(\frac{t_o}{2} \right) \quad (3.6)$$

Burada, d_u : ultrasonik sensörlerle ölçülen mesafe, V_s : havada sesin yaptığı hız, t_o : gönderilen ses sinyalinin dönüş zamanıdır.

Ses hızının ortam sıcaklığına göre havada yaptığı hız yaklaşık olarak (3.7)'deki eşitlikten hesaplanabilir. Robot üzerindeki ultrasonik sensör okumaları $20^\circ C$ ortam sıcaklığındaki $V_s = 343 m/sn$ ses hızını dikkate almaktadır.

$$V_s = 331.4 \sqrt{\frac{{}^0C + 273}{273}} \text{ m/sn.} \quad (3.7)$$

Burada, 0C : ortam sıcaklığıdır.

Çalışılan ortamın sıcaklığı mevsimlere göre değişeceğinden okunan ultrasonik değerlerin bu sıcaklık değerlerine göre yeniden ayarlanabilmesi için çeşitli ortam sıcaklıklarında testler yapılmıştır. Bu amaçla robot üzerindeki ultrasonik sensörlerden birinin 300 cm. uzağına bir kutu yerleştirilmiş ve değişen sıcaklıklara göre bu mesafe okutturularak çıkan sonuçlar Tablo 3.6'da verilmiştir. Sensörün

bütün okumaları 20°C lik ortam sıcaklığına göre yaptığı dikkate alınarak, eşitlik (3.6) ve (3.7)'nin kullanılmasıyla değişen ortam sıcaklıklarına göre değişen ses hızları Tablo 3.6'da verildiği şekilde hesaplanabilmektedir. Matlab yazılımı [13] ile Tablo 3.6'daki değişen ortam sıcaklığına göre değişim gösteren havadaki ses hızının grafiği çizdirilerek Şekil 3.6'da verilmiştir. Şekilden de görülebileceği gibi çıkan sonuç (3.7)'deki eşitliği doğrular niteliktedir.

Tablo 3.6 Ultrasonik sensörlerin $d=300\text{ cm}$ uzaklıktaki bir nesneyi değişen ortam sıcaklıklarına göre okuması ile hesaplanan ses hızları

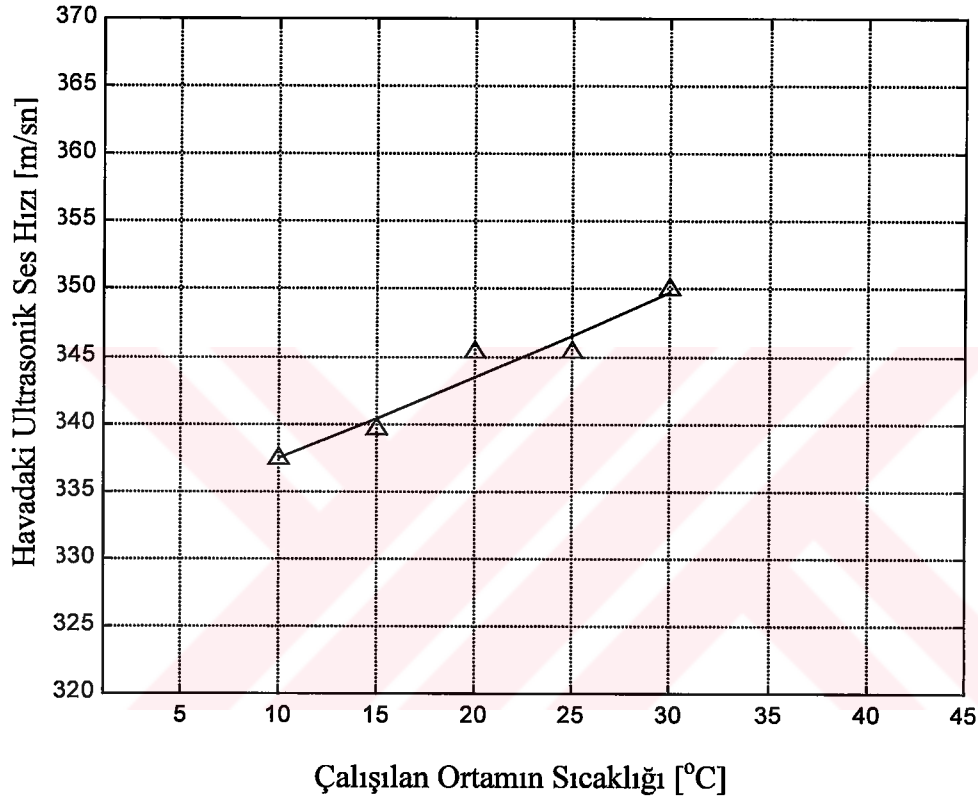
Ortam Sıcaklıkları [$^{\circ}\text{C}$]	$d=300\text{cm. için}$ Ölçülen Değerler [cm]	Hesaplanan Ses Hızları [m/sn]
10	305	337.4
15	303	339.6
20	298	345.3
25	298	345.3
30	294	350.0

Yapılan çalışmada robot üzerindeki yerden 0.6 metre yükseklikte 22.5° aralıklarla eşit olarak yerleştirilmiş 16 adet Polaroid 6500 serisi ultrasonik alıcı/verici sensör kullanılmaktadır. 20° 'lik konik açılımı içinde yer alan nesnelerin kendisine olan uzaklığını ölçebilen sensör okumaları, okuma hatalarını en aza indirebilmek amacıyla ortam sıcaklığına göre eşitlik (3.8) kullanılarak yeniden ayarlanır. Nomad200 mobil robotunun sürekli 20°C kabul ederek yaptığı mesafe ölçümleri (d) Şekil 3.7'deki grafikte verilen C_T değişkeni ile ortam sıcaklığına göre yeniden düzenlenir. Eşitlik (3.8)'de kullanılan ve ortam sıcaklığına göre değişen bu C_T değerleri Şekil 3.6 yardımıyla hesaplanmıştır.

$$d_{ger} = C_T \cdot d_u \quad (3.8)$$

Burada, d_u : ultrasonik sensörlerin ölçtüğü mesafe değeri, d_{ger} : ortam sıcaklığına göre olması gereken gerçek mesafedir.

Ultrasonik sensörlerin ihmal edilemeyecek bazı dezavantajları da vardır. Her şeyden önce, sesin havadaki hızı, hareket halindeki robotun çevresindeki nesneleri tespit edebilme zamanını sınırlar. Kuvvetli gelen eko sinyalleri daha önceden belirlenen eşik gerilimini aşarak çok doğru sonuçlar verirken, zayıf eko sinyalleri gecikmeden kaynaklanan hatalı okumalara sebep olur.

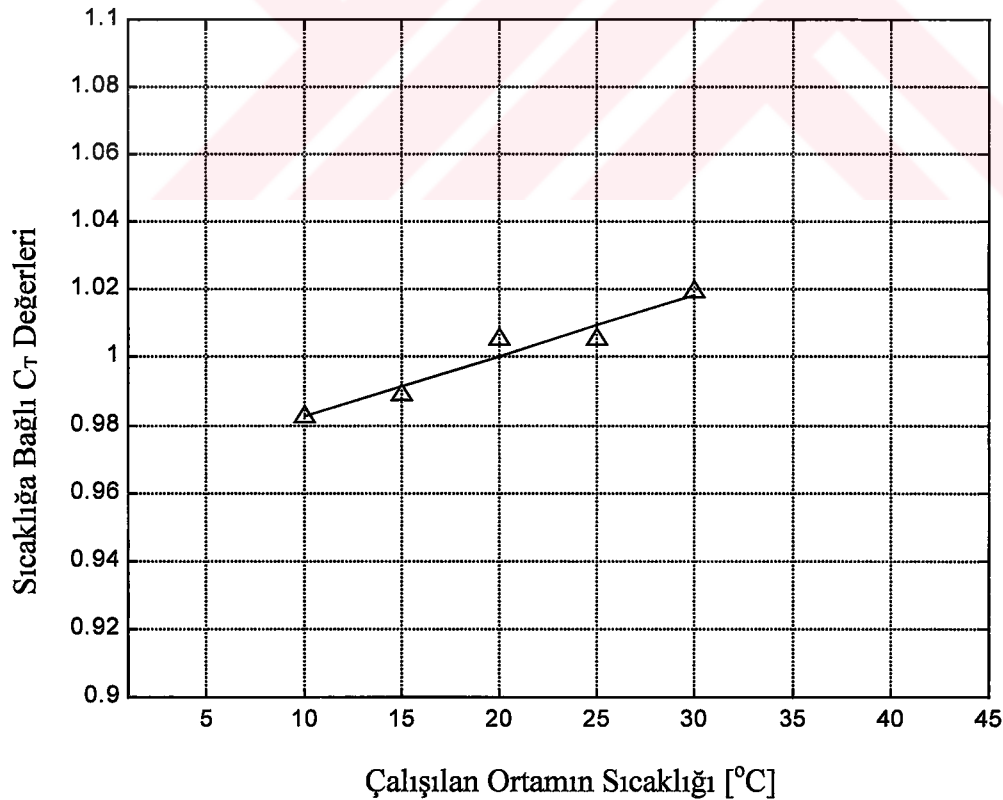


Şekil 3.6 Çalışan ortam sıcaklığına göre değişen havadaki ses hızı.

Diğer önemli bir konu da çapraz-karışma problemidir. Bu durum bir sensör tarafından gönderilen sinyale ait ekoların komşu sensörler tarafından alınması ile oluşan hatalı okumalara sebep olur. Çapraz-karışmayı önlemek için ultrasonik sensörler aynı anda ateşlenmemelidir. Robot üzerindeki ateşleme sistemi, gerçekleştirilecek bir yazılım ile 4 msn. ile 1024 msn. arasındaki zaman aralığında sensörleri ardışıl olarak ateşleme kapasitesine sahiptir. Bu yüzden ultrasonik sensörleri kullanan algoritmalarda sensörlerin ateşlenme dizisi birbirine 180° karşılık gelecek şekilde düzenlenmiştir.

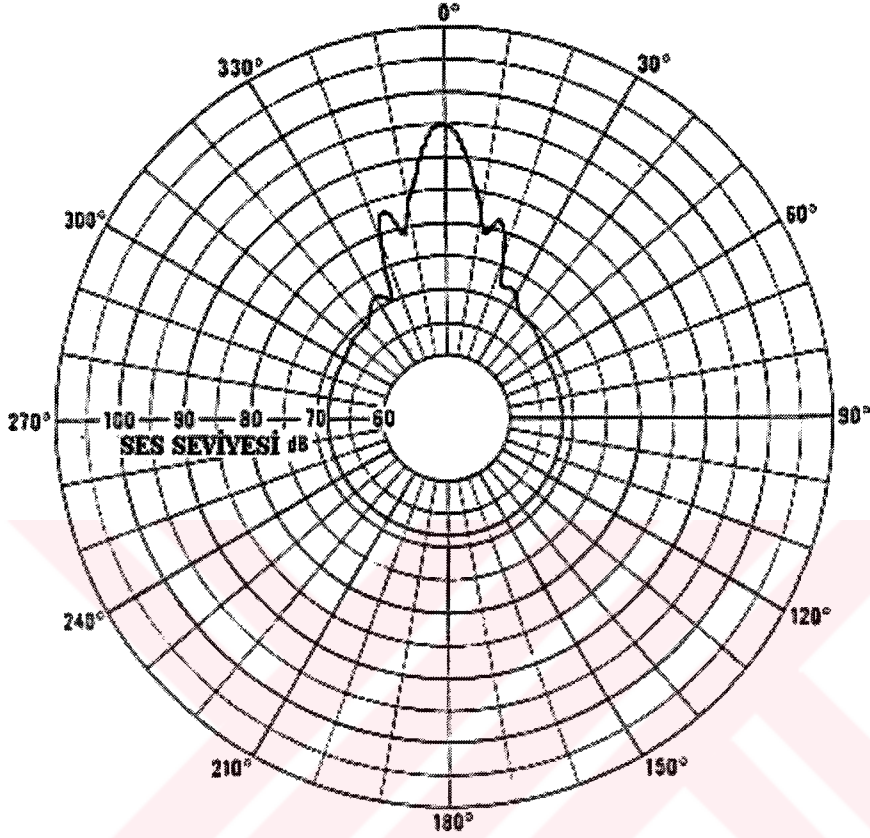
Ultrasonik sensörlerin diğer bir problemi de spekülative yansımalar dır. Bu tür yansımalar gönderilen ultrasonik dalğanın konik eksenini ile ortamdaki nesnelerin yansıtan yüzeylerinin dikgen (normal) açısı arasındaki farkın büyük olmasından kaynaklanmaktadır. Böyle durumlardaki nesne yüzeyleri gelen ultrasonik dalgaları sensörden oldukça uzaklara yansıtarak o nesnelerin tespit edilebilmesini engelleyebilmekte veya olduğundan daha yakın tespit edilebilmesine (yüzeyin sadece sensöre yakın parçasının görülmesi) sebep olmaktadır. Yapılan ölçümlerde gönderilen ultrasonik dalğanın konik eksenini ile engellerin yansıtan yüzeylerinin dikgen açısı arasındaki fark 40 dereceden fazla olduğunda bu tür bir problemle karşılaşılmalıdır.

Son olarak, ultrasonik sensörler Şekil 3.8’de verildiği gibi 20°’lik konik ışımaya örüntüsüyle zayıf yönlenmeye sahiptir. Gelen eko sinyalinin merkezde ve her iki yanında yer alan ışımalarından hangisine ait olduğu belli değildir. Bu problem, karşılaşılan nesnenin ultrasonik sensörün konik eksenine olan açısına bağılı olarak ortaya çıkar. Dolayısıyla, tespit edilen nesnenin konik açılımının neresinde yer aldığı bilinmezliğine yol açar.



Şekil 3.7 Çalışılan ortam sıcaklığına göre ultrasonik okumaları kalibre eden C_T değerleri.

Bahsedilen bütün bu dezavantajlar ultrasonik sensörlerin hatalı okumalarına sebep olmaktadır. Bu sensörlerin yanında farklı yapıda sensörler kullanılarak gerek lokalizasyon gerekse de çevre haritalama konusunda yapılan çalışmalarda geliştirilen sensör füzyonu teknikleriyle bu eksiklikler giderilmeye çalışılmıştır.



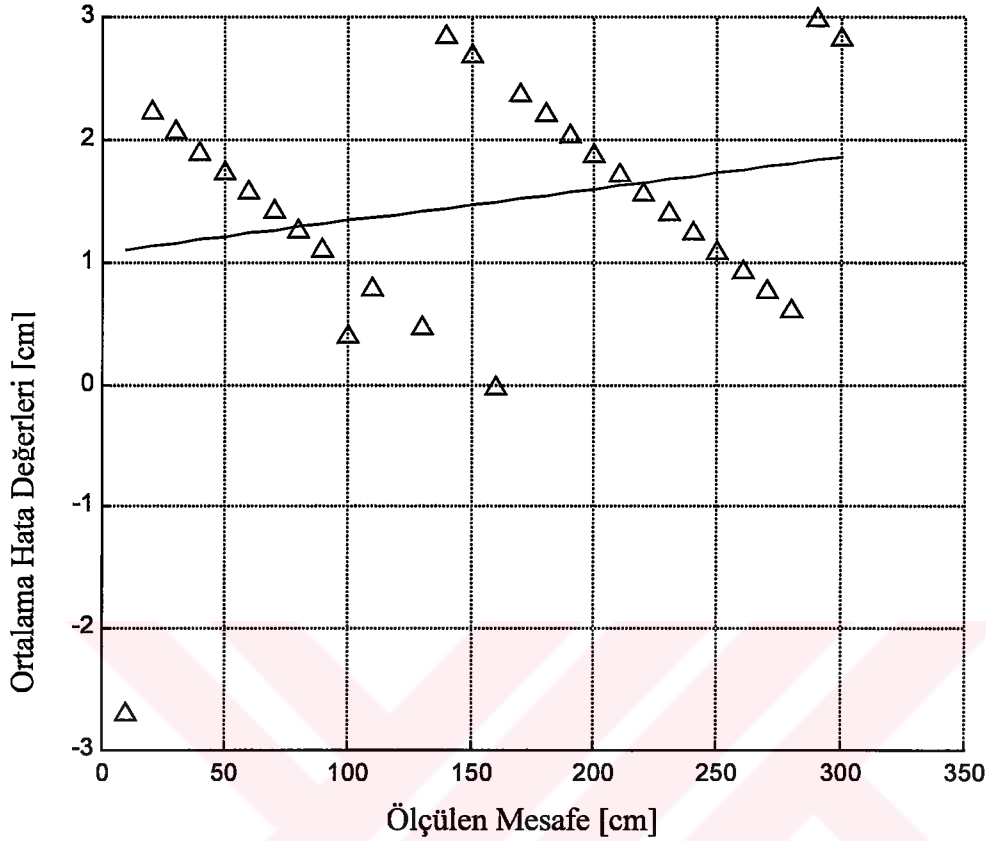
Şekil 3.8 Ultrasonik sensörlerin ışıma örüntüsü.

3.3.1 Ultrasonik Sensörlerde Konum Hata Analizi

Ultrasonik sensörlerin yapısından kaynaklanan sınırlamaların ortaya çıkardığı okuma hataları sensörün değişen sıcaklıklara göre kalibre edilmesi ile sınırlı değildir. Durağan halindeki Nomad200 mobil robotu üzerindeki ultrasonik sensörlerle yapılan deneylerde ölçülen mesafeye ve açılara göre okuma hatalarının sırası ile Şekil 3.9 ve Şekil 3.10'deki gibi değiştiği görülmüştür. Şekil 3.9'daki deneyde nesneler ultrasonik sensörün konik eksenine bakacak şekilde yerleştirilmiştir. Sensör 10 ile 300 cm ötesine yerleştirilen nesneleri ortalama 1.48 cm. lik bir hata ile ($\sigma_d=1.14$) ölçebilmiştir. Mesafeye göre değişen bu hataları aşağıdaki formülle ifade edebiliriz:

$$E(d_u)_{mes} = 0.0026 * d_u + 1.0768 \quad (3.9)$$

Burada $E(d_u)_{mes}$: sensörün cm cinsinden $\pm$ mesafe okuma hatası, d_u : ise cm cinsinden ölçülen mesafedir.

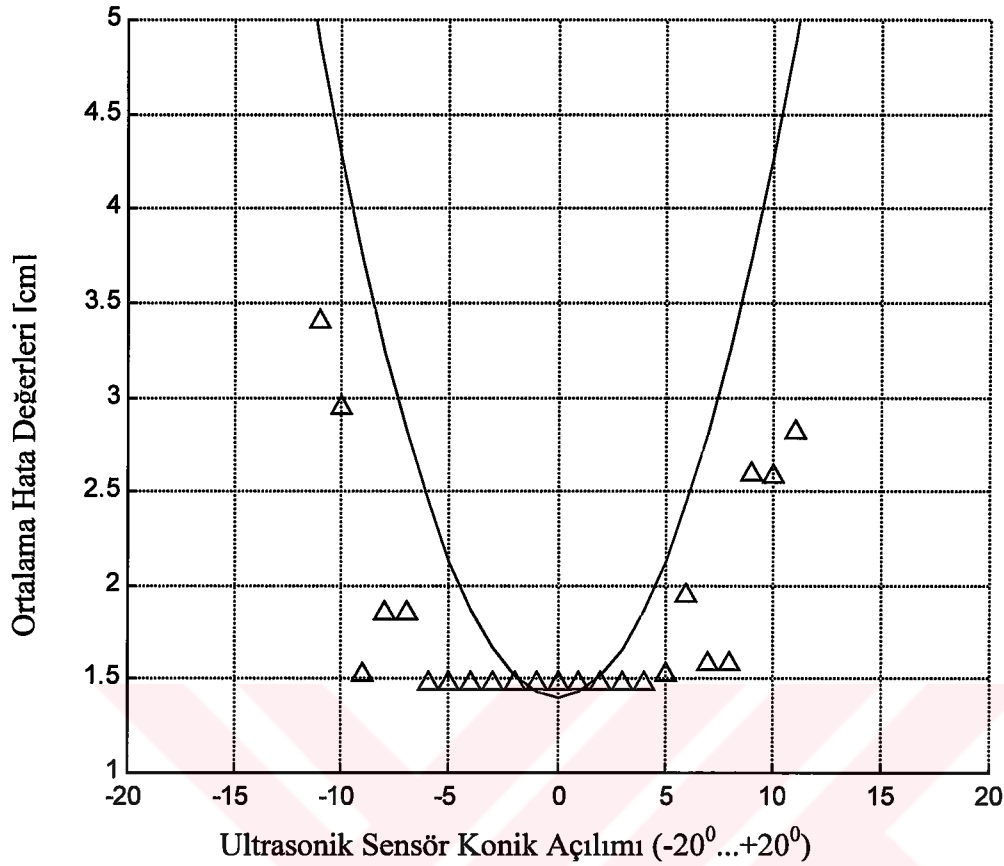


Şekil 3.9 Ultrasonik sensörlerde konumsal hata dağılımı.

Şekil 3.10'daki deneyde ise ultrasonik sensörün $\pm 20^\circ$ 'lik konik açılımı içine 50 cm-300 cm arasındaki çeşitli uzaklıklara yerleştirilen bir nesnenin sensöre olan uzaklığı çeşitli açılardan ölçülmeye çalışılmıştır. Yaklaşık $\pm 9^\circ$ 'den sonra oldukça hatalı ölçümler yapmakta olan sensör ortalama 1.85 cm. lik bir hata ile ($\sigma_\beta=0.584$) konik açılımı içindeki nesnelerin kendisine olan uzaklığını ölçebilmiştir. Konik eksene göre değişen " β " açısına bağlı bu hatalar aşağıdaki formülle ifade edilebilir:

$$E(\beta)_{açı} = 0.0287*\beta^2 - 0.0002*\beta + 1.4045 \quad (3.10)$$

Eşitlik (3.9) ve (3.10)'da mesafeye ve konik açılıma göre tarifi yapılan bu hatalar bir sonraki kısımda verilmekte olan sensör modellerinde ve Bölüm 4.4'de anlatılan lokalizasyon amaçlı Sınırlandırılmış Dinamik Sensör Füzyonu tekniğinde robotun olası konumunun sınırlarının tayininde kullanılmıştır.



Şekil 3.10 Ultrasonik sensörlerin açısal hata dağılımı.

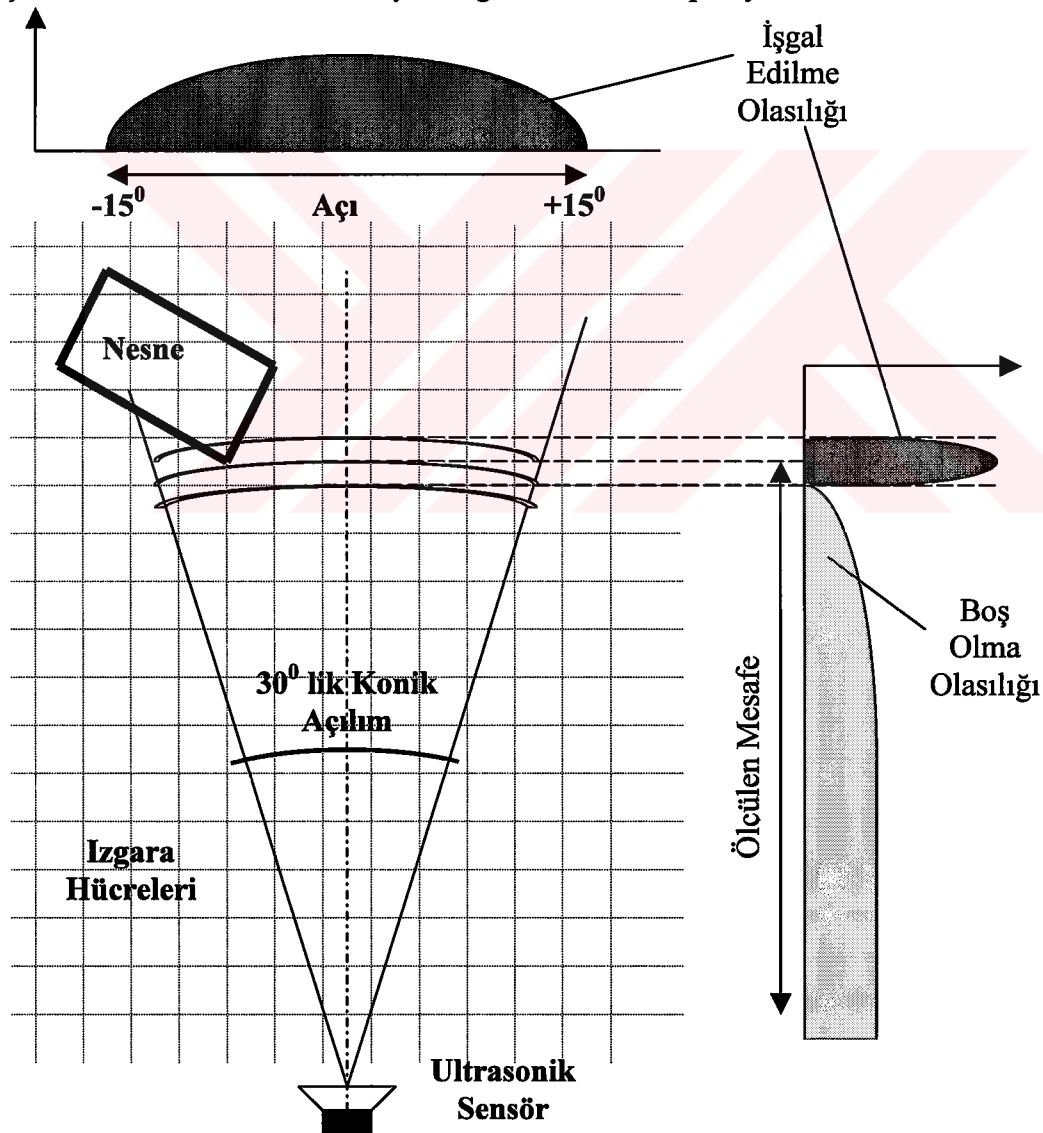
3.3.2 Çevre Haritalama Amaçlı Sensör Modeli

Hareket halindeki Nomad200 mobil robotu üzerinde yapılan deneysel çalışmalar sonucunda genel olarak ultrasonik sensör okumalarının sıfır ortalama ve varyanslı σ^2 Gauss gürültüsü tarafından etkilendiği görülmüştür. Bu gürültüye en büyük sebepler arasında robot üzerindeki motorlar ve çevresel faktörler sayılabilir. Çalışılan ortamda yer alan çeşitli cihazların sensör çalışma frekanslarına yakın yaydıkları sinyaller bu faktörlerin başında yer almaktadır. Buna göre Şekil 3.11'de gösterildiği gibi ultrasonik sensörün yaptığı okumalar gerek kendi fiziksel sınırlamalarından ve gerekse de çevresel faktörlerden kaynaklanan bütün olumsuzluklar göz önüne alınarak eşitlik (3.11)'de verilen olasılıksal yoğunluk fonksiyonu ile ifade edilebilir. Mobil robot elde ettiği ultrasonik okumaları bu fonksiyon ile işleyerek etrafındaki

engellerin bulunma olasılıklarını çıkartabilir. Bu modelleme Bölüm 5.2’de verilen çevre haritalama yönteminde kullanılmıştır.

$$P_u(d_u | r, \beta) = \frac{1}{2\pi\sigma_d\sigma_\beta} \exp \left[-\frac{1}{2} \left(\frac{(d_u - r)^2}{\sigma_d^2} + \frac{\beta^2}{\sigma_\beta^2} \right) \right] \quad (3.11)$$

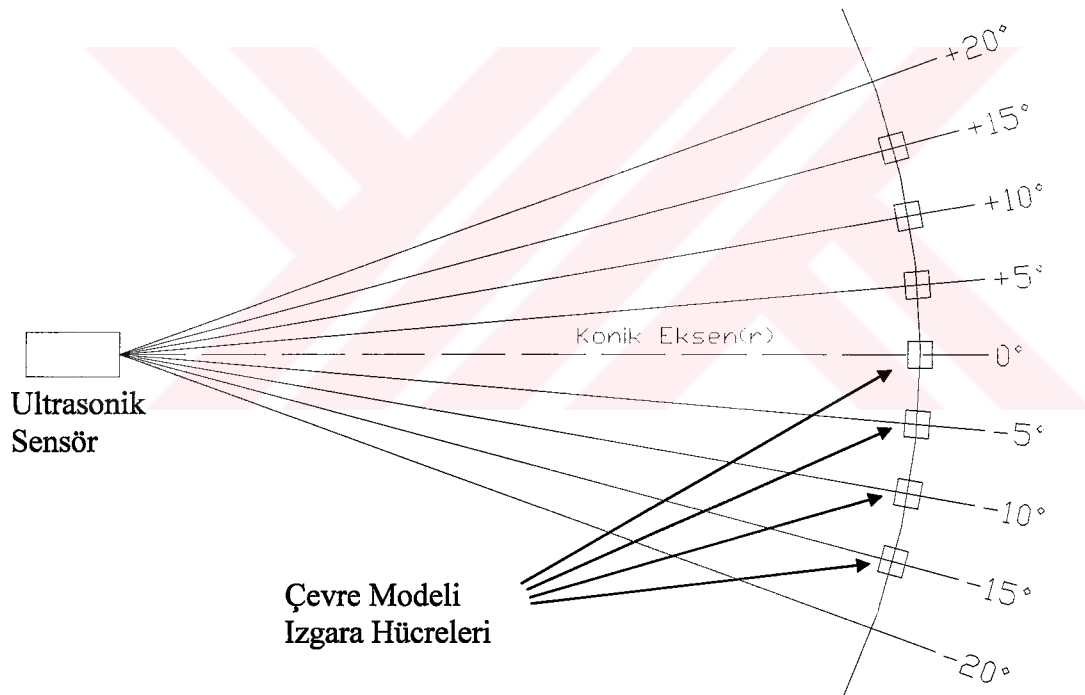
Burada, d_u : ultrasonik sensörün ölçtüğü mesafeyi, r : robotla ölçülen nesne arasındaki gerçeğe yakın mesafe değerini, β : Şekil 3.12’de gösterildiği gibi ultrasonik sensörün konik ekseninin sıfır alınmasıyla bu eksenin her iki tarafında “ d ” yarıçaplı bir yay doğrultusunda oluşan 0° , $\pm 5^\circ$, $\pm 10^\circ$ ve $\pm 15^\circ$ şeklindeki açısal değerleri, σ_d : konik eksene göre mesafe okumalarında meydana gelen standart sapmayı, σ_β : ise $\pm 15^\circ$ ’lik açısal mesafe okumalarında meydana gelen standart sapmayı temsil eder.



Şekil 3.11 Bir ultrasonik sensörün konik açılımının iki-boyutlu görünümü.

Eşitlik (3.11)'de yer alan (d_u-r) değeri ölçülen mesafe ile gerçek mesafe arasındaki farkı verdiğinden bu haliyle eşitlik (3.9)'da verilen hatayı tarif etmektedir. Bu yüzden $(d_u-r)^2$ ifadesi yerine " d_u " mesafe ölçümüne karşılık gelen hata değeri eşitlik (3.9) hesaplanarak karesi olan $E^2(d_u)_{mes}$ değeri konur.

Bölüm 5.2'de detaylı bir şekilde anlatılan ızgaralara bölünmüş bir ortamda ultrasonik sensörlerin kullanıldıkları durumlarda, " r " mesafesine karşılık gelen her bir " β " açısı için hesaplanan $P_u(d_u|r, \beta)$ olasılıksal değeri kendilerine en yakın ızgara hücrelerine atanır. Örneğin; $C_1, C_2, C_3, C_4, C_5, C_6$ ismi verilen ızgara hücrelerine karşılık gelen engel temsil edilme olasılıkları sırası ile $C_1=P_u(d_u|r, 15^\circ)$, $C_2=P_u(d_u|r, 10^\circ)$, $C_3=P_u(d_u|r, 5^\circ)$, $C_4=P_u(d_u|r, 0^\circ)$, $C_5=P_u(d_u|r, -5^\circ)$, $C_6=P_u(d_u|r, -10^\circ)$, $C_7=P_u(d_u|r, -15^\circ)$ şeklindedir.



Şekil 3.12 Ultrasonik sensöre bağlı Olasılıksal Yoğunluk Fonksiyonunun atanacağı $\pm\beta$ ızgara hücreleri.

3.4 Kızılötesi Sensörler ve Modeli

Kızılötesi alıcı sensörlerin okuduğu mesafe bilgileri kızılötesi vericilerin yaydığı ışığın çarptığı nesnelerden yansıyan şiddetidir. Çalışılan ortamdaki aydınlatma sistemi ve yer alan nesnelerin renkleri kızılötesi alıcı sensörlerinin zaman zaman hatalı okumalarına sebep olabilir. Nomad200 üzerinde kullanılan kızılötesi sensörler $\pm 30^\circ$ 'lik konik açılımla sadece 38 cm'ye kadar okuma yapabilmektedir. Yapılan deneyler sonucunda kısa mesafelerde etkili okuma yapabilen sensörün, ortamdaki ışık şiddetinden ve kullanılan nesnelerin renklerinden kaynaklanan sıfır ortalama ve varyanslı σ^2 bir Gauss gürültüsünden etkilendiği görülmüştür. Sensörün konik açılım etkisi mesafenin kısıllığından dolayı dikkate alınmamıştır. Buna göre oluşturulan kızılötesi sensörün olasılıksal yoğunluk fonksiyonu eşitlik (3.12)'de verilmiştir.

$$P_k(d_k | r) = \frac{1}{\sqrt{2\pi}\sigma_k^2} \exp\left[-\frac{1}{2} \frac{(d_k - r)^2}{\sigma_k^2}\right] \quad (3.12)$$

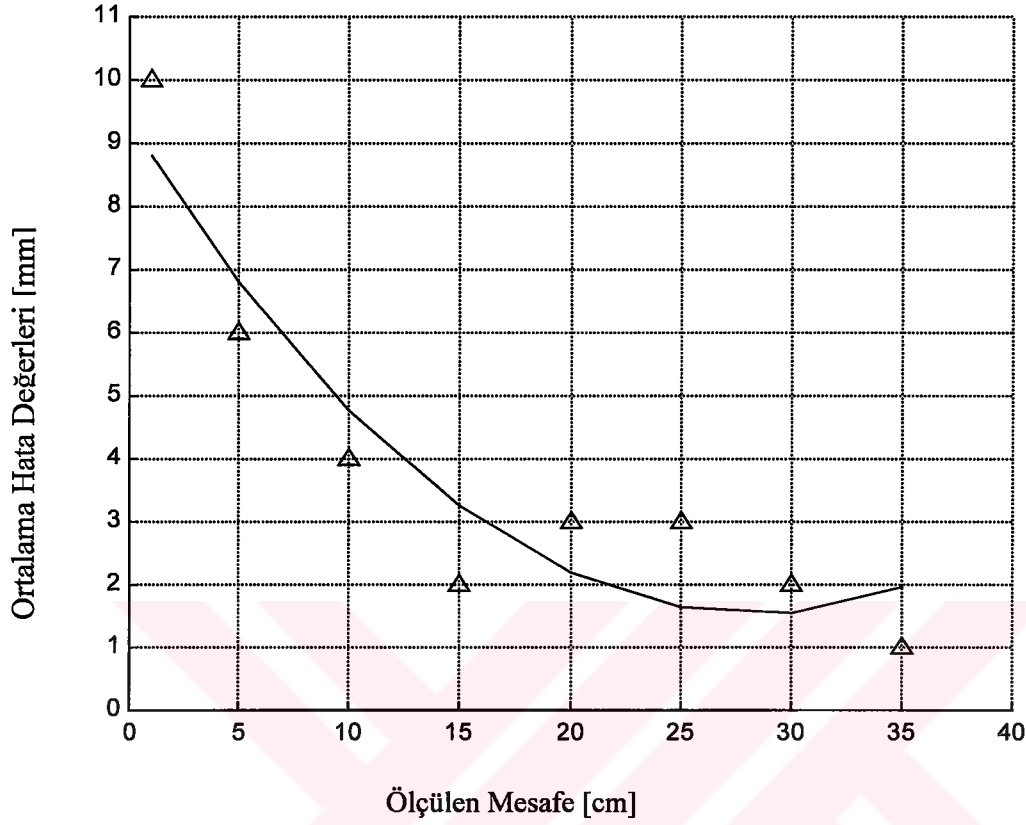
Burada, d_k : kızılötesi sensörün ölçtüğü mesafe, r : ise robotla ölçülen nesne arasındaki gerçeğe yakın mesafe bilgisidir.

Kızılötesi sensörlerin tek başına kullanıldıkları durumlarda, “ r ” mesafesine karşılık gelen $\pm 30^\circ$ 'lik konik açılım içindeki her bir 5° 'lik açı aralığına en yakın ızgara hücrelerine aynı $P_k(d_k|r)$ olasılıksal değeri atanır.

Eşitlik (3.12)'de verilen $(d_k - r)$ ölçüm hatası, yapılan deneylerde ölçülen mesafeye göre Şekil 3.13'deki gibi değiştiği görülmüştür. Deneyde nesneler kızılötesi sensörün konik eksenine bakacak şekilde yerleştirilmiştir. Sensör 2 ile 38 cm. ötesine yerleştirilen nesneleri ortalama 3.9mm.'lik bir hata ile ($\sigma_k=0.62$) ölçebilmektedir. Mesafeye göre değişen bu hatalar cm. cinsinden aşağıdaki formülle ifade edilir:

$$E(d_k)_{mes} = 0.001*(d_k)^2 - 0.055*(d_k) + 0.934 \quad (3.13)$$

Ultrasonik sensör modelinde olduğu gibi eşitlik (3.12)'deki $(d_k - r)^2$ ifadesi yerine “ d_k ” mesafe ölçümüne karşılık gelen hata değeri eşitlik (3.13)'den hesaplanarak karesi olan $E^2(d_k)_{mes}$ değeri konur.



Şekil 3.13 Kızılötesi sensörlerde konumsal hata dağılımı.

3.5 Manyetik Pusula

Bir mobil robotun çalışılan ortam içindeki görevine yönelik yapacağı bütün hareketlerin planlanmasında " θ ," robot baş açısının bilinmesi çok önemlidir. Odyometrik sistemden gelen " θ ," bilgisinde, Bölüm 3.2'de bahsedilen nedenlerden dolayı ortaya çıkabilecek hataların birikmesini önlemek için bazı alternatif sensör okumalarına ihtiyaç duyulur. Manyetik pusulalar mutlak bir açı ölçer olma özelliğinden dolayı bu ihtiyaca en iyi cevap verebilen sensörlerdir. Fakat kendisini çevreleyen manyetik alan kuvvet çizgilerine oransal olarak üzerinde meydana gelen gerilmeyi ölçebilen bu cihazlar yapılarından dolayı bazı dezavantajlara sahiptirler. Örneğin yeryüzü kaynaklı manyetik alan çizgileri güç hatlarının veya çelik ve benzeri yapıların yakınlarda bozulmaya uğrar. Bu şekilde bozulmuş manyetik alan

izgileri pusulanın hatalı okumalarına sebep olur. zellikle bina ii uygulamalarda bu okuma hataları olduka sıklasır. Sensrn tm bu olumsuz yanlarına nlem olarak, Blm 4’de kullanılan konumlandırma algoritmalarında hatalı okumaların yoğunlaştığı anlarda veya blgelerde manyetik pusula okumaları dikkate alınmamıştır.

Yapılan alıřmada Nomad200 mobil robotuna kolay adapte edilebilir ve ucuz olduėu iin teknik zellikleri Tablo 3.7’de verilen KVH firmasının rettiėi C100 pusulası [7] kullanılmıřtır. Sensr yeryz kaynaklı $H=(h_x, h_y, h_z)$ manyetik alan řiddetinin yatay bileřenlerini ler. Burada h_x ileri ynde, h_y ise bu ynn matematiksel olarak karřı tarafındaki ynde ve h_z de dikey ynde llen manyetik alan bileřenleridir. H-vektr x-y koordinat dzleminde bir daireyi ifade eder. Bu nedenle robot bař aısı (yn) manyetik alanın x-y bileřenlerinin ařaėıda eřitlik (3.14) verilen dnřmleri ile bulunabilir.

$$\theta_{pus} = \tan^{-1}(h_y / h_x) - \theta_o \quad (3.14)$$

Burada θ_o : lm yapılmadan nce okunan sabit deėerdir.

Herhangi bir dıř etki altında kalmadıėı anlarda $\pm 0.5^\circ$ ’lik bir hata hassasiyeti ile tam doėruya yakın lme yapabilen bu sensrden doėabilecek okuma hataları diėer sensr okumalarının Blm 4’de verilen fzyon yntemleriyle birleřtirilmesi sayesinde giderilmiřtir.

Tablo 3.7 KVH C100 pusulasının teknik zellikleri [7]

Parametreler	Deėerleri
znrlk	$\pm 0.1^\circ$
Doėruluk	$\pm 0.5^\circ$
Yinelenebilirlik	$\pm 0.2^\circ$
Boyut	4.6 cm x 11.4 cm x 2.8 cm
Aėırlık	64 gram
Elektriksel G	< 1W
ekilen Akım	40mA DC
Giriř Voltajı	+8V ... +18V DC

3.6 Fiber-Optik Jiroskop

Mobil robotlarda jiroskoplar robot dönüş oranlarını ölçerek robota bulunduğu yön bilgisini sağlarlar. Her bir jiroskop içinde onun algılayıcılarından gelen değerlerini yükselten ve okunabilir hale getiren bir DC yükselteç vardır. Bu yükseltecin çıkışında çalıştığı ortama ve içindeki elektronik elemanların meydana getirdiği ısıya bağlı olarak, göstermesi gereken gerçek değerden bazı oranlarda sapmalar (offset) oluşur. Bu durum, her jiroskopun en önemli problemi olan ve kendi imalat yapısından kaynaklanan “kayma oranı” veya “sapma oranı” diyebileceğimiz bir problemi oluşturur. Zaman içinde sapma oranının meydana getirdiği birikimler ölçülecek açısal değerlerin olduğundan büyük çıkmasına neden olur. Çeşitli tiplerdeki jiroskopların teknik incelemelerini 1995’de Everett [14] yayınlamıştır. İki ayrı jiroskop üzerinde yapılan deneysel ve teorik çalışmalar ise 1995’de Barshan ve Durrant-Whyte [15] tarafından sunulmuştur.

Yapılan çalışmada, en son jiroskop teknolojisi olan KVH fiber-optik jiroskop [8] kullanılmıştır. KVH jiroskopunun teknik özellikleri Tablo 3.8’de verilmiştir. Bu tür jiroskopların sapma oranları oldukça düşük olduğundan yeni bir sapma oranı ölçülmeden önce robot hareketlerinden birkaç dakikalık sağlıklı bir ölçüm sağlanabilir. Bu amaçla, yapılan çalışmada mobil robot zaman zaman durmak zorunda olduğu anlarda 20 saniyelik sapma testi yapılmıştır. Robot durağan haldeyken elde edilen bu verilerin ortalaması alınarak statik w_{bias} değeri bulunmuş ve bu değer robot hareket halindeyken elde edilen dönme hızı w_g değerinden düşülmüştür. Daha önceki yapılan çalışmalarda ise ısıya bağlı bir sapma oranı hesaplanarak, robot hareket halindeyken yapılan w_g ölçümlerinden düşülmüştür. Uygulanan bu yaklaşım önceki uygulamalara göre daha iyi sonuç vermiştir.

Fiber-optik jiroskopların diğer bir problemi de onun doğrusal olmayan ölçeklendirme faktörüdür (SF). Sabit bir değer olan SF jiroskopun üretici firması tarafından kullanma kılavuzlarında verilir. Verilen bu SF değeri de hesaba katılarak jiroskopun saniyede kaç derecelik bir dönüş ölçümü yaptığı aşağıdaki eşitlikle hesaplanabilir:

$$w_{jir} = (w_g - w_{bias}) \cdot SF \quad (3.15)$$

burada kullanılan jiroskopun SF değeri sabit sıcaklıklarda (20° - 24° C) 0.005, 40° 'nin üstündeki sıcaklıklarda ise 0.01 olduğundan ortalama olarak 0.075 alınmıştır. Robot durağan iken ölçülen w_{bias} değerleri 0.005° - 0.025° /sn. arasında değişmektedir. Çalışılan laboratuvar ortamındaki sıcaklık genelde 20° - 25° C arasında değişmektedir.

Gerçekleştirilen jiroskop okumaları robotun sadece dönüş yaptığı anlarda bir robot yön açısı " θ_r ," bilgisi sağlamaktadır. Bu bilgi Bölüm 3.2'de anlatıldığı şekilde, odyometreden, pusuladan ve çevreye serpiştirilmiş konumu bilinen silindirik nesnelerden sağlanan diğer " θ_r ," bilgileri ile birleştirilerek robot baş açısının sağlıklı bir şekilde ölçülmesine çalışılmıştır.

Tablo 3.8 KVH Jiroskopunun Teknik Özellikleri [8]

Parametreler		Değerler
Giriş dönme oranı		$\pm 100^{\circ}$ /sn
Anlık bandgenişliği		100Hz
Sapma Oranı (24° C)		0.005° /sn 18° /saat
Isıl oranlar	Çalışma esnasında	-40° C ... $+75^{\circ}$ C
	Muhafaza esnasında	-50° C ... $+85^{\circ}$ C
Isınma zamanı		1sn
Boyutlar		11.5cm x 9cm x 4.1cm
Ağırlık		250gram
Elektriksel Güç		< 3W
	Çekilen akım	100mA DC
	Giriş voltajı	+24V DC

3.7 Sonuç

Bu bölümde yapılan çalışmalarda bir mobil robotun güvenli navigasyonu için gereken lokalizasyon, çevre haritalama ve yol planlama amaçlı kullanılan ve genellikle robot üzerinde konumlandırılan odyometrik, ultrasonik, kızılötesi, manyetik pusula ve fiber-optik jiroskop sensörleri tanıtılarak modellemeleri yapılmıştır.

Robot tekerleklerinde yer alan odyometrik sensör ünitesinden alınan robot konumu (x_r, y_r) ile yönü (θ_r) bilgilerinde oluşan sistematik ve sistematik olmayan hatalar analiz edilerek, bu hataları en aza indirgeyecek bazı önlemler geliştirilmiştir. Yine robot gövdesinde yerden 60 cm. yükseklikte yer alan ultrasonik sensörler üzerinde, kendi fiziksel yapılarından kaynaklanan kusurlar ile çalışılan ortamın atmosfer sıcaklığı da dikkate alınarak konumsal ve açısal hata analizleri yapılmıştır. Yapılan bu analizler neticesinde ortaya lokalizasyon amaçlı bir hatalı ölçüm aralığı çıkartılmıştır. Diğer bir tarafta ise, aynı sensörler çevre haritalama amaçlı kullanılmak üzere, ortamın ızgara hücreleri şeklinde bölündüğü de dikkate alınarak bir olasılıksal yoğunluk fonksiyonu şeklinde modellenmiştir. Aynı şekilde robot gövdesi üzerinde yerden 12 cm. yükseklikte yer alan ve sadece çevre haritalama amaçlı kullanılan kızılötesi sensörler ultrasonik sensörler de olduğu gibi bir olasılıksal yoğunluk fonksiyonu şeklinde modellenmiştir.

Robot baş açısında denilen yönünü (θ_r) ölçme konusunda büyük hatalar yapabilen odyometrik sensör ünitesini desteklemek amacıyla robot üzerine sonradan ilave edilen manyetik pusula ve fiber-optik jiroskopun teknik özellikleri tablolar halinde verilerek, çalışma prensipleri matematiksel modelleriyle anlatılmıştır. Özellikle manyetik pusulanın kapalı ortamlarda sık sık zafiyete düşmesi durumlarında robot dönüş bilgisini hassasiyetle verebilen fiber-optik jiroskop hakkında detaylı bilgi verilmiştir.

Bu bölümdeki modellemeler 4. ve 5. bölümlerde verilen lokalizasyon ve çevre haritalama konuları ile 6. bölümde verilen yol ve hareket planlama konularında oldukça yaygın bir şekilde kullanılmaktadır.

4. LOKALİZASYON YÖNTEMLERİ VE SINIRLANDIRILMIŞ DİNAMİK SENSÖR FÜZYONU-SDSF

4.1 Giriş

Mobil robot uygulamalarında bir aracın lokalizasyonunun (konum ve açı bilgisi) tam olarak bilinmesi önemli bir problemdir. Araştırmacılar bu problemin çözümüne yönelik çeşitli sensörler, sistemler ve teknikler geliştirmişlerdir. Bu konuda yapılan ilk çalışmalarda geliştirilen algoritmaların zaman alıcı ve o yıllardaki bilgisayarların da yavaş olmasından dolayı lokalizasyon işlemi süresince robotun hareketsiz kaldığı gözlenmektedir. Lokalizasyon işleminde genel olarak kamera, optik mesafe okuyucular, ultrasonik sensör ve parıldaklar (beacons) gibi çeşitli sensörler kullanılmaktadır. Bir lokalizasyon sistemi için hız, maliyet ve doğruluk önemli birer kriterdir. Kak ve Kosaka [16] tarafından gerçekleştirilen bir kamera tabanlı lokalizasyon işleminde azalan hız ve artan maliyeti ile tam bir doğruluk sağlanmıştır. Ultrasonik parıldaklarla gerçekleştirilen işlemlerde lokalizasyon hızı maliyetindeki beklenen artışı ile göze çarpıcı bir şekilde arttırılmıştır. İstatistiksel harita eşlemeye (matching) dayanan bir statik lokalizasyon algoritmasının bilgisayarda işlenmesi ise çok fazla zaman aldığından, lokalizasyon hızı ile robot hızını önemli miktarda düşürür. Bir duvar-eşlemeye dayalı teknikte [9] ise optik mesafe okuyucular kullanılarak iyi bir sonuç elde edilmiştir. Bu sonucu etkileyen en önemli faktör optik sensörlerin dar bir huzme açıklığına (beamwidth) sahip olmasıdır. Şayet geniş bir huzme açıklığına sahip ultrasonik sensörler kullanılsaydı, bu lokalizasyon tekniğinin [9] performansı oldukça azalacaktı. Preciado ve arkadaşları [17] ise uyguladıkları bir özyinelemeli yöntem ile sadece işlenmemiş ferdi sensör bilgilerini birleştirerek lokalizasyon işleminde bir gelişme sağlamıştır. Buna rağmen çalıştıkları sensörlerin huzme açıklığını dikkate almadıklarından tam bir doğruluk sağlayamamışlardır. Lokalizasyon işlemi özyinelemeli bir yapıda ardışıl olarak ilk önce mobil robotun baş açısının (yönünün) güncellenmesi ve hemen arkasından konumunun bulunması şeklinde gerçekleştirilir.

Diğer bir taraftan, alternatif bir yöntem olarak Kalman filtreleme robot baş açısının ve konumunun aynı anda güncellenmesinde kullanılabilir. Bu nedenle, Kalman filtresini kullanarak eldeki tüm sensör ve lokal harita bilgilerini birleştirebilen ve mobil robotun olası lokalizasyonunu sağlayabilen bir sistem tasarlanabilir. Nitekim Chatila [18] tarafından geliştirilen bir yöntem, robot üzerindeki laser tarayıcısının sağladığı çevre bilgilerini Kalman filtresi kullanarak birleştirir. Bu yöntemde ultrasonik sensörlere göre daha az belirsizlikler olmasına rağmen, laser tarayıcıların bilinen tüm dezavantajlarına sahiptir. Diğer taraftan, Durrant-Whyte ve Leonard [19] tarafından geliştirilen bir sistemde, ölçüm fonksiyonlarında “Geometrik Parıldaklar” kavramı kullanılmış, ancak bu sistemde de herhangi bir nümerik sonuç rapor edilmemiş olup sadece birkaç uygulama problemi analiz edilmiştir. Schiele ve Crowley [20] ise ilk defa olarak Hough dönüşümünü kullanarak işgal edilmiş ızgaralardan doğrusal parçalar çıkarmışlardır. Bu araştırmacılar yöntemlerinde iki farklı işgal edilmiş ızgara sistemi kullanmışlardır. Bunlardan biri mobil robotun karşılaşılabileceği engellerden hemen reaksiyon göstererek sakınabileceği lokal ızgara sistemi, diğeri ise robotun tüm çalışma alanını kapsayan global bir ızgara sistemidir. Araştırmacılar geliştirdikleri bu sistemde lokal ızgaralarda oluşturulan doğrusal parçalarla global ızgaralarda yer alan doğrusal parçaları eşleştiren teknikleriyle robotun olası konumunu tayin edebilmektedir. Ayrıca, Leonard, Durrant-Whyte ve Cox [21] tarafından geliştirilen diğer bir yöntemde ise, ultrasonik sensörlerle Kalman filtreleme tekniği kullanılarak çalışılan ortamın modeli çıkarılmıştır. Daha sonra bu çıkarılan model üzerinde mobil robotun olası konumu tahmin edilmeye çalışılmıştır.

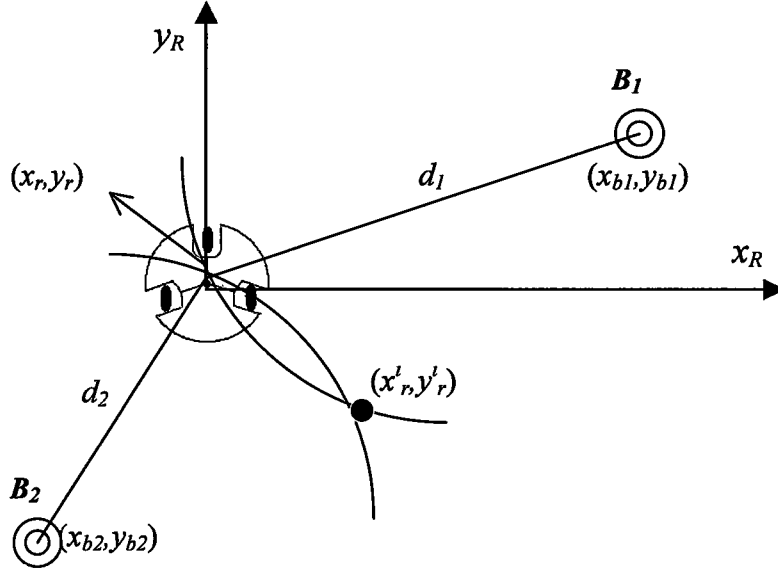
Bu çalışmada oluşturulan yöntemler 3 tekerlekli silindirik yapıda, sıfır-jayro çaplı (olduğu yerde dönebilen holonomik olmayan yapı) bir mobil robot olan Nomad-200 üzerinde uygulanmıştır. Robot üzerinde ultrasonik, kızılötesi ve dokunma sensör halkaları mevcuttur. Her biri 10’ar adet sensör içeren dokunma halkaları iki adettir. Robot etrafına 22.5°’lerle yerleştirilen ultrasonik sensörler 16 adettir. Odyometre sensörleri ise mobil robot lokalizasyon bilgilerini sağlayabilmek amaçlı, tekerlekler üzerindeki eşzamanlı sürücü sistem üzerine konumlandırılmıştır. Odyometrik sistemden alınan bilgi mobil robotun konfigürasyonu şeklindeki konum ve baş açısını (x_r, y_r, θ_r) vermektedir. Bu üniteye ait modelleme Bölüm 3.2’de detaylı bir şekilde verilmiştir.

Yapılan çalışmada robot tekerleklerinde meydana gelebilecek herhangi bir kayma ve buna benzer hataların birikmeden telafi edilebilmesi için konumları bilinen silindirik borular kullanılmıştır. Mobil robotun üzerindeki ultrasonik sensörlerle en az iki tanesini yakalayabilecek sıklıkta çalışma ortamına yerleştirilen borular yaklaşık 10 cm. çaplı ve 1 metre uzunluğundadır. Robot bizim daha önce yaptığımız çalışma ile [22] yakaladığı en az iki silindirik borudan kendi konumunu ve baş açısını (x_r, y_r, θ_r) tahmin etmeye çalışır. Bu çalışmada, odyometre ve konumu bilinen silindirik borulardan elde edilen robot baş açısı (θ_r), aynı zamanda robot üzerindeki bir pusula ve jiroskop bilgileri ile birleştirilerek sensör hatalarının en aza indirgendiği daha sağlıklı bir yön bilgisinin elde edilebileceği gösterilmiştir.

4.2 Konumu Bilinen Nesnelerle Lokalizasyon

Robot tekerleklerinde yer alan optik kod çözücülü odyometrik sistemlerin robot konumunu tahminindeki olumsuzlukların giderilmesinin bir yolu da konumu bilinen referans nesnelerden destek almaktır. Bu amaçla geliştirilen Newton Özyinelemeli Lokalizasyon-NÖL yönteminde [22] konumu bilinen 10 cm. çaplı 1 metre uzunluğundaki silindirik referans nesneler kullanılmıştır. Mobil robotun ultrasonik sensörleri ile en az iki tanesini yakalayabileceği şekilde ortama yerleştirilen silindirik parçalar dar bir genişliğe sahip olduğundan robot tarafından bir nokta şeklinde algılanırlar. Geliştirilen yöntemde sadece konumu bilinen nesnelerin bulundukları koordinatlar ile robota olan uzaklıkları kullanıldığından ultrasonik sensörlerin konik açılımlarından kaynaklanan ölçüm hataları robot konumunun kestirimini olumsuz yönde etkilememiştir.

Yöntemin anlatımında da kullanılmak üzere hazırlanan laboratuvar ortamında mobil robot Şekil 4.1’de gösterildiği gibi ortamda bulunan B_1 ve B_2 ismi verilen silindirik nesneleri yakalamaktadır. Ultrasonik sensörlerle robota olan uzaklıkları d_1 ve d_2 olarak ölçülen bu nesneleri robot konumunun tahmininde kullanan bu yöntem Bölüm 4.3.3’de detaylı olarak anlatılmaktadır.



Şekil 4.1 Ultrasonik sensörlerle NÖL yöntemi kullanılarak gerçekleştirilen lokalizasyon.

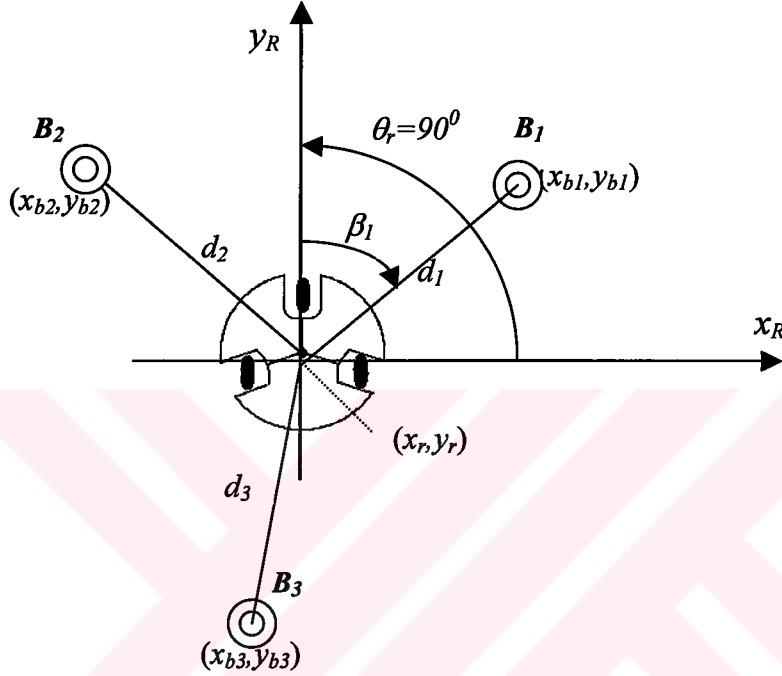
4.2.1 Robot Konumunun Kestirimi

Robot yönünün (θ_r) bilinmemesi durumunda Şekil 4.2’de verildiği biçimde ortamdaki referans nesnelerden en az iki tanesinin ultrasonik sensörlerle yakalanması gerekmektedir. Robot yönünün bilinmesi durumunda konumlandırma için tek bir referans nesnenin sensörler tarafından görülmesi yeterli olmaktadır. Bu durum eşitlik (4.1)’de Şekil 4.2’deki B_1 , B_2 ve B_3 ($i=1,2,3$) nesnelerinden sadece birinin yakalanması durumuna göre verilmiştir.

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x_{bi} - d_i \cdot \cos(\theta_r - \beta_j) \\ y_{bi} - d_i \cdot \sin(\theta_r - \beta_j) \end{bmatrix} \quad (4.1)$$

burada, θ_r : robot tekerlerinin baktığı yön (robot yönü), x_r , y_r : robot koordinatları, x_{bi} , y_{bi} : konumu bilinen referans nesnelerin koordinatları, d_i : ultrasonik sensörlerle ölçülen robot ile referans nesneler arasındaki mesafe, β_j : ise referans nesneleri yakalayan j nci ultrasonik sensörün robot koordinat sistemine (x_R , y_R) göre bulunduğu açıdır. Örneğin üzerinde testler yapılan Nomad200 robotu 16 ultrasonik sensöre sahiptir. Bu durumda $j=0,1,2,...,15$ dir ve robot koordinat sistemine göre sensör konumları $\beta_0=0^\circ$, $\beta_1=22.5^\circ$, $\beta_2=45^\circ, ..., \beta_{15}=337.5^\circ$ şeklindedir.

Robot yönünün bilinmediği varsayılararak geliştirilen Bölüm 4.2.3’de anlatılan NÖL yönteminin anlatımını ve kullanımını kolaylaştırabilmek amacıyla, Şekil 4.1’deki B_1 ve B_2 referans nesnelerinin sensörlere olan uzaklıklarına sırası ile d'_1 ve d'_2 , robot yarıçapına da d_o denirse, bu nesnelerin robot merkezine (x_r, y_r) olan uzaklıkları da $d_1 = d'_1 + d_o$ ve $d_2 = d'_2 + d_o$ eşitlikleri ile ifade edilecektir. Nomad200 mobil robotu için $d_o=45$ cm. dir.



Şekil 4.2 Ultrasonik sensörler tarafından referans nesnelerin yakalanması.

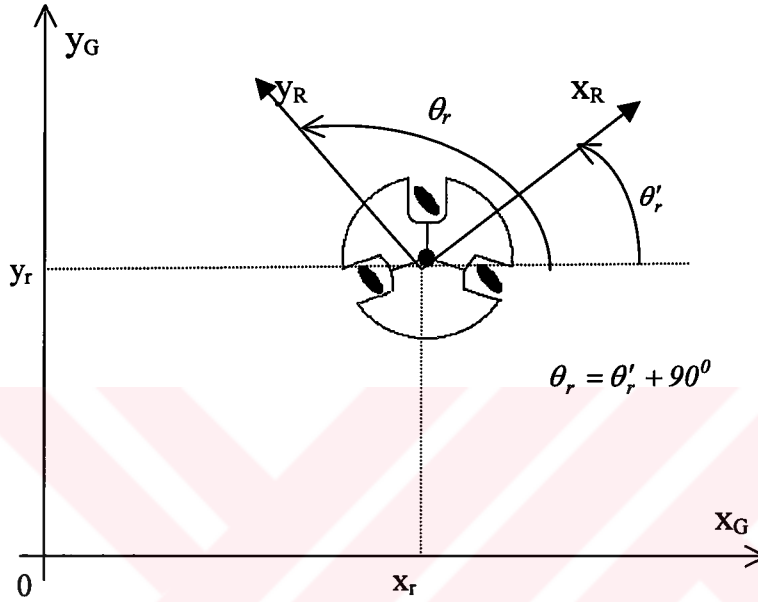
Mobil robotun noktasal konum koordinatlarının (x_r, y_r) hesaplanabilmesi amacıyla Şekil 4.1’deki gibi yakalanan en az iki referans nesnenin kendi konumları ve robota olan uzaklıkları hesaba katılarak çıkarılan (4.2)’de ki doğrusal olmayan eşitliğin Bölüm 4.2.3’deki NÖL yöntemiyle çözülmesi gereklidir.

$$d_i^2 = (x_{bi} - x_r)^2 + (y_{bi} - y_r)^2, \quad i=1,2 \quad (4.2)$$

Mobil robotun ortamda bulunan silindirik nesnelerden en az iki tanesini yakalayamaması durumunda konum bilgisi eşitlik (4.2)’den sağlanamayacağından bir sonraki ölçüm zamanına kadar sadece güncellenmiş odyometrik verilerine bağlı kalınır.

4.2.2 Robot Yönünün Kestirimi

Robot yönü (θ_r) Şekil 4.3'den de görüleceği gibi robot koordinat sistemi (x_R, y_R) ile global koordinat sisteminin (x_G, y_G) saat yönünün tersi istikamette x -eksenleri arasındaki açının (θ'_r) 90° ilerisidir ($\theta_r = \theta'_r + 90^\circ$). Şekilde robot tekerleklerinin baktığı doğrultu Bölüm 2'de detaylı bir şekilde verildiği gibi robot koordinat sisteminin y -eksenini oluşturmaktadır.



Şekil 4.3 Robot koordinat sisteminin global koordinat sistemindeki görüntüsü.

Nomad200 mobil robotunun dış gövdesini oluşturan, üzerinde ultrasonik, kızılötesi, manyetik pusula ve fiber-optik jiroskop gibi sensörleri barındıran ve robot orta noktasına göre 360° dönme yeteneğine sahip yapıya Turret denmektedir. Lokalizasyon konusunda yapılan çalışmaların üzerinde test edildiği bu robotunun her bir tekerleği aynı anda Turret sisteminden bağımsız olarak hareket etmektedir. Robot konum ve yön bilgilerinin belli bir sistematik içinde hesaplanabilmesi amacıyla robot tekerlekleriyle Turret aynı oranda döndürülür. Bu durumda birinci ultrasonik sensör daima tekerleklerin baktığı doğrultuyu izler. Aynı zamanda manyetik pusuladan ve fiber-optik jiroskoptan okunan bilgi de robot yönünü (θ_r) verecektir.

Nomad200 üzerinde yapılan sensör ve tekerlek koordinasyonu ve NÖL yöntemiyle robot konum koordinatlarının bulunması işlemlerinden sonra robot yönü (4.4)'deki eşitlikler kullanılarak hesaplanabilir. Bu eşitliğin çıkarılmasında, robot yönünün

bulunması için Şekil 4.1'deki gibi en az iki referans nesnenin yakalanması gerçeğinden yola çıkarak yazılan (4.3)'teki eşitlikten yararlanılmıştır.

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x_{b1} - d_1 \cdot \sin(\beta_i - \theta'_r) \\ y_{b1} - d_1 \cdot \cos(\beta_i - \theta'_r) \end{bmatrix} = \begin{bmatrix} x_{b2} - d_2 \cdot \sin(\beta_j - \theta'_r) \\ y_{b2} - d_2 \cdot \cos(\beta_j - \theta'_r) \end{bmatrix} \quad (4.3)$$

burada, β_i ve β_j : sırası ile B_1 ve B_2 referans nesnelerini yakalayan i nci ve j nci sensörlerin robot koordinat sisteminde bulundukları açılar, θ'_r : robot koordinat sistemi ile global koordinat sistemlerinin x -eksenleri arasındaki açıdır.

$$\theta'_r = \text{ArcTan} \left(\frac{\Delta y_b \cdot d_2 \cdot (\sin \beta_j - \sin \beta_i) - \Delta x_b \cdot (d_2 \cdot \cos \beta_j - d_1 \cdot \cos \beta_i)}{\Delta y_b \cdot d_1 \cdot (\cos \beta_j - \cos \beta_i) - \Delta x_b \cdot (d_2 \cdot \sin \beta_j - d_1 \cdot \sin \beta_i)} \right) \quad (4.4a)$$

$$\theta_r = \theta'_r + 90^\circ \quad (4.4b)$$

Burada $\Delta y_b = y_{b2} - y_{b1}$, $\Delta x_b = x_{b2} - x_{b1}$, dir.

4.2.3 Newton Özyinelemeli Lokalizasyon-NÖL Yöntemi

Doğrusal olmayan cebirsel eşitliklerin çözümünde özyinelemeli yapıya sahip olan algoritmalar yaygın olarak kullanılır. Bu amaçla incelenen “Basit İterasyon”, “Newton” ve “Dik İniş” yöntemlerinden en hızlı ve doğru cevabı verebilen Newton yöntemi Şekil 4.1'deki B_1 ve B_2 referans noktalarına göre robot konumunun bulunmasını sağlayacak olan (4.2)'deki eşitliğin çözülmesinde kullanılmıştır. Şekildeki referans noktalara (4.2)'de verilen eşitliği uyguladığımızda her bir referans noktası için (4.5) ve (4.6)'da ki eşitlikleri elde ederiz.

$$d_1^2 = (x_{b1} - x_r)^2 + (y_{b1} - y_r)^2 \quad (4.5)$$

$$d_2^2 = (x_{b2} - x_r)^2 + (y_{b2} - y_r)^2 \quad (4.6)$$

Eşitlik (4.5) ve (4.6)'da verilen doğrusal olmayan yapıyı çözebilecek şekilde yeniden uyarlanarak ve yapısındaki özyineleme özelliği de dikkate alınarak yöntemin ismine

“Newton Özyinelemeli Lokalizasyon”, kısaca NÖL denmiştir. Yöntemin anlatımında kullanılan konum ve mesafe ifade eden semboller Şekil 4.1’den alınmıştır.

Eşitlik (4.5) ve (4.6)’da verilen doğrusal olmayan cebirsel eşitlikler çözüme yönelik olarak (4.7) ve (4.8)’de verilen f_i fonksiyonları haline getirilebilirler.

$$f_1(x_r, y_r) = (x_{b1} - x_r)^2 + (y_{b1} - y_r)^2 - d_1^2 = 0 \quad (4.7)$$

$$f_2(x_r, y_r) = (x_{b2} - x_r)^2 + (y_{b2} - y_r)^2 - d_2^2 = 0 \quad (4.8)$$

Problemin çözümüne yönelik olarak; x_r ve y_r ifadelerinin (4.7) ve (4.8)’deki eşitliklerin çözümü olduğu varsayılarak, u ve v sembollerini aşağıdaki eşitliklerde kullanılıp sıfıra eşitlenirse;

$$u = f_1(x_r, y_r) = 0 \quad (4.9)$$

$$v = f_2(x_r, y_r) = 0 \quad (4.10)$$

(4.9) ve (4.10)’da verilen eşitlikler x - y düzleminde u - v düzlemine dönüştürme işlemi şeklinde göz önüne alınabilir. Bu dönüşümün (x_{rk}, y_{rk}) noktası yakınındaki davranışı (4.11) ve (4.12)’deki eşitliklerde verilen fonksiyonların kısmi türevleriyle ifade edilebilir.

$$u - u_k \approx \frac{\partial}{\partial x} f_1(x_{rk}, y_{rk}) \cdot (x_{rk+1} - x_{rk}) + \frac{\partial}{\partial y} f_1(x_{rk}, y_{rk}) \cdot (y_{rk+1} - y_{rk}) \quad (4.11)$$

$$v - v_k \approx \frac{\partial}{\partial x} f_2(x_{rk}, y_{rk}) \cdot (x_{rk+1} - x_{rk}) + \frac{\partial}{\partial y} f_2(x_{rk}, y_{rk}) \cdot (y_{rk+1} - y_{rk}) \quad (4.12)$$

Şayet $J(x_{rk}, y_{rk})$ Jacobian matrisi kullanılırsa, bu ilişki aşağıdaki gibi açık bir şekilde ifade edilebilir.

$$\begin{bmatrix} u - u_k \\ v - v_k \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} f_1(x_{rk}, y_{rk}) & \frac{\partial}{\partial y} f_1(x_{rk}, y_{rk}) \\ \frac{\partial}{\partial x} f_2(x_{rk}, y_{rk}) & \frac{\partial}{\partial y} f_2(x_{rk}, y_{rk}) \end{bmatrix} \cdot \begin{bmatrix} x_{rk+1} - x_{rk} \\ y_{rk+1} - y_{rk} \end{bmatrix} \quad (4.13)$$

Eşitlik (4.7) ve (4.8)'i çözebilmeye yönelik Newton yöntemini geliştirmek için fonksiyonların (x_r, y_r) noktası yakınındaki küçük değişimlerinin göz önüne alınmaya ihtiyaç vardır. Bu değişimler (4.14) ve (4.15)'deki eşitlikler ile tanımlanmıştır.

$$\Delta u = u - u_k, \quad \Delta v = v - v_k \quad (4.14)$$

$$\Delta x_r = x_{rk+1} - x_{rk}, \quad \Delta y_r = y_{rk+1} - y_{rk} \quad (4.15)$$

(4.9) ve (4.10)'da verilen eşitliklere göre $(u, v) = (0, 0)$ olduğuna göre bağımsız değişkenlerdeki değişimler (4.16) ve (4.17)'deki eşitlikler şeklinde yazılabilir.

$$u - u_k = f_1(x_r, y_r) - f_1(x_{rk}, y_{rk}) = 0 - f_1(x_{rk}, y_{rk}) \quad (4.16)$$

$$v - v_k = f_2(x_r, y_r) - f_2(x_{rk}, y_{rk}) = 0 - f_2(x_{rk}, y_{rk}) \quad (4.17)$$

Doğrusal bir yaklaşıklık elde etmek için (4.16) ve (4.17)'deki eşitliklerin sonucu (4.7) ve (4.8)'de kullanıldığında (4.18)'deki yapı elde edilir.

$$\begin{bmatrix} \frac{\partial}{\partial x} f_1(x_{rk}, y_{rk}) & \frac{\partial}{\partial y} f_1(x_{rk}, y_{rk}) \\ \frac{\partial}{\partial x} f_2(x_{rk}, y_{rk}) & \frac{\partial}{\partial y} f_2(x_{rk}, y_{rk}) \end{bmatrix} \cdot \begin{bmatrix} \Delta x_r \\ \Delta y_r \end{bmatrix} = - \begin{bmatrix} f_1(x_{rk}, y_{rk}) \\ f_2(x_{rk}, y_{rk}) \end{bmatrix} \quad (4.18)$$

Eğer (4.18)'de verilen $J(x_{rk}, y_{rk})$ Jacobian'ı tekil değilse, Δx_r ve Δy_r için çözüm (4.19)'da verildiği şekilde olacaktır.

$$\begin{bmatrix} \Delta x_r \\ \Delta y_r \end{bmatrix} \approx -[J(x_{rk}, y_{rk})]^{-1} \cdot \begin{bmatrix} f_1(x_{rk}, y_{rk}) \\ f_2(x_{rk}, y_{rk}) \end{bmatrix} \quad (4.19)$$

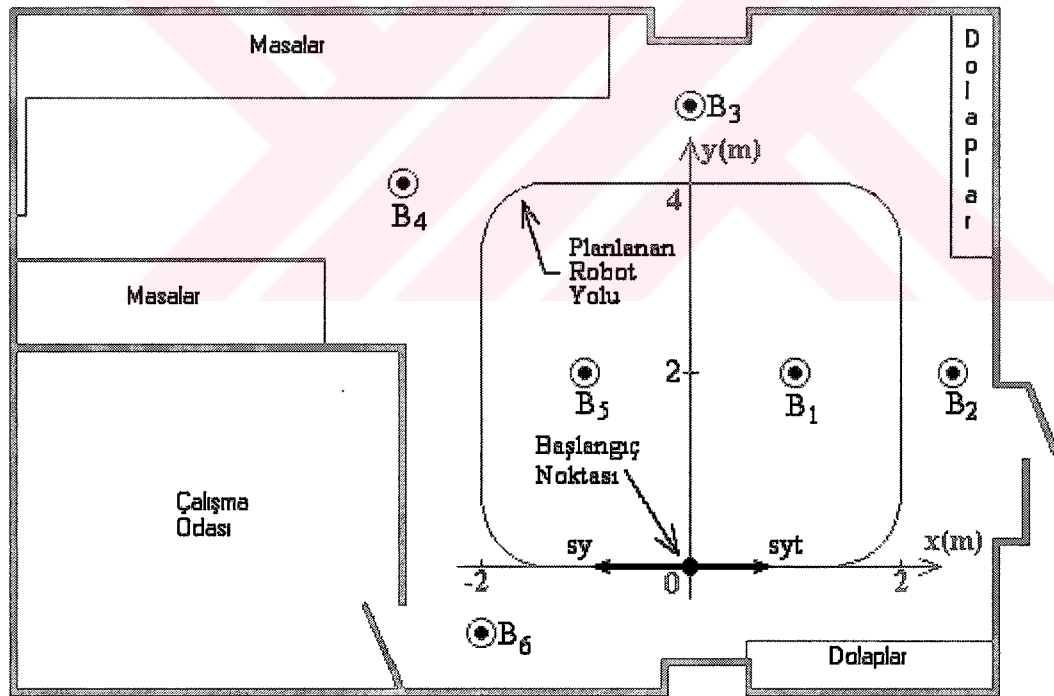
(4.19)'da ki eşitlikten (x_r, y_r) çözümüne olan net yaklaşıklık (x_{rk+1}, y_{rk+1}) ifadesi (4.20) ve (4.21)'deki eşitlikler kullanılarak elde edilebilir.

$$x_{rk+1} = x_{rk} + \Delta x_r \quad (4.20)$$

$$y_{rk+1} = y_{rk} + \Delta y_r \quad (4.21)$$

4.2.4 NÖL Yönteminin Nomad200 Mobil Robotu Üzerinde Testi

Yöntemin Nomad200 mobil robotu üzerinde test edilebilmesi ve diğer lokalizasyon yöntemleri ile karşılaştırılabilmesi için laboratuvar içinde planlanan yörünge Şekil 4.4'de verilmiştir. Robot tarafından ikişer defa saat yönü (sy) ve saat yönünün tersine (syt) izlenecek yörüngenin etrafına uygun aralıklarla 6 adet referans silindirik nesne ($i=1, \dots, 6$) yerleştirilmiştir.



Şekil 4.4 Test amaçlı planlanan robot yolu ve ortamdaki referans silindirik nesnelerin konumları.

İlk olarak başlangıç noktası olan $(x_o, y_o, \theta_o)^T = (0, 0, 0)$ pozisyonuna yerleştirilen robotun tekerlek ilerleme hızı yaklaşık $v=26$ cm/sn. değerine, tekerlek dönme hızı

ise yaklaşık $w=30^0/\text{sn}$ değerine ayarlanır. Bir turun uzunluğu yaklaşık olarak 15.14 metredir. Yapılan ilk deneyde pozitif x-yönünde (syt) robota, planlanan yörünge üzerinde iki tur attırılmıştır. Bu durumda robot tekerleği sola dönüşler yaparak toplam $\rho \approx 30.3$ metrelik bir mesafe üzerinde hareket etmektedir. Yapılan ikinci deneyde ise robota, diğer yönde (syt) yörünge yine iki defa izlettirilmiştir.

Robot yörünge izleme algoritmasının temelini Bölüm 6'da detaylı bir şekilde anlatılan eş-zamanlı robotlarda kullanılan hareket denklemleri oluşturur. Aynı algoritma içine yerleştirilen NÖL konumlandırma yöntemiyle ve (4.4a)ve (4.4b)'de verilen eşitlikler yardımıyla, robot hareket halindeyken 2 saniyede bir (örnekleme periyodu, τ) ortamdaki konumu bilinen silindirik nesnelerden ($B_1, \dots, B_6$) en az iki tanesi ultrasonik sensörlerle yakalanarak olası robot konum ve yönü $(x_r, y_r, \theta_r)^T$ sürekli güncellenmektedir. Yapılan iki türlü izlemeler sonucunda yaklaşık 60 örnekleme yapılarak robot konum ve yönünde düzeltmeler yapılmaktadır.

Robotun yörünge izlemesi ve yapılan işlemler sonucunda aynı zamanda başlangıç noktası olan vardığı son noktada oluşan konum hataları Tablo 4.1 ve 4.2'de gösterilmiştir. Robot, yörüngesini izlerken ve konum ile yönünü tahmin ederken kendi üzerindeki odyometrik sensör ünitesi ile ultrasonik sensörleri kullanmıştır. Elde edilen sonuçları değerlendirmek ve karşılaştırma yapabilmek için iki farklı sensör grubundan alınan ham bilgiler ayrı ayrı saklanarak, her ikisinin tek başına kullanımıyla ortaya çıkabilecek konum ve yön hataları incelenmiştir.

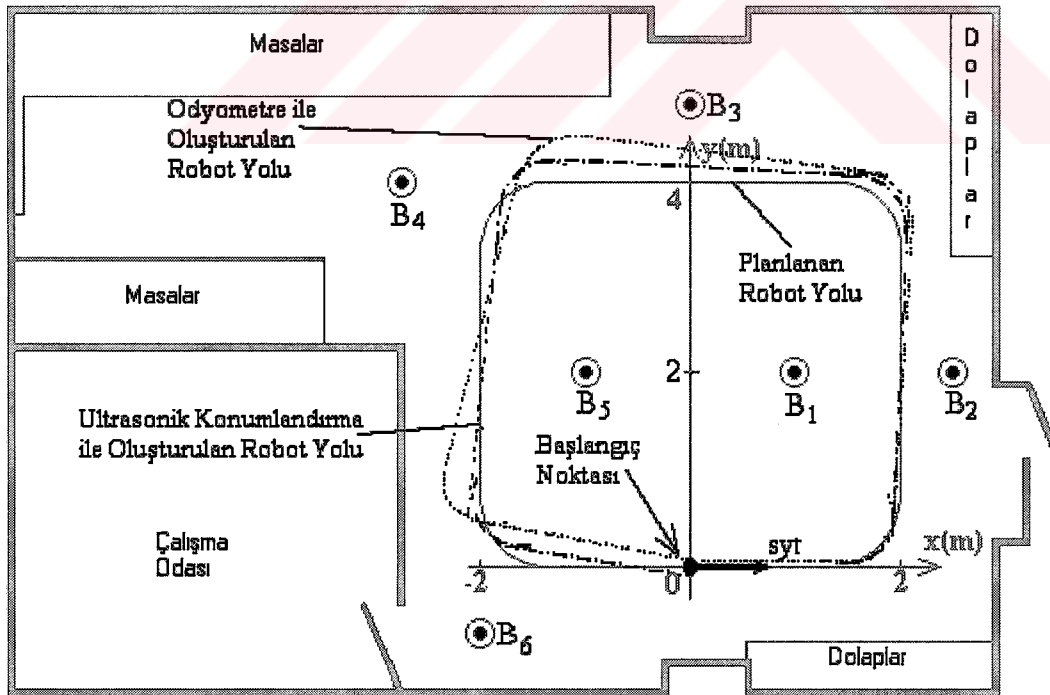
Tablo 4.1 İki tur sola dönüş (synt) sonrasındaki konum ve yön hataları.

<i>HATALAR</i> <i>SENSÖRLER</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
Odometre (odo)	56.2	-1.6	-12 0	% 1.86
Ultrasonik Lokalizasyon (ult)	-5.1	-4.0	-6.5 0	% 0.21

Tablo 4.2 İki tur sağa dönüş (sy) sonrasındaki konum ve yön hataları.

HATALAR SENSÖRLER	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
Odometre (odo)	5.9	1.1	-9.5 $^{\circ}$	% 0.20
Ultrasonik Lokalizasyon (ult)	4.3	-3.5	-5.0 $^{\circ}$	% 0.18

Tablo 4.1 ve 4.2’de sadece odyometrik üniteden (odo) ve sadece ultrasonik lokalizasyon algoritmasından (ult) elde edilen veriler gösterilmiştir. Burada Δx x-koordinatındaki, Δy y-koordinatındaki ve $\Delta \theta$ ise robot yönündeki sapmalardır. Son sütunda da, $\Delta \rho / \rho$ ile bağıl mesafe hatası yüzde olarak hesaplanmaktadır, burada $\Delta \rho$ başlangıç ve varış noktaları arasındaki Öklid mesafesidir. Tablolara göre sadece odyometrenin kullanımında robot yönündeki hatalar ve bununla beraber x, y kartezyen koordinatlarındaki konum hataları hızla artmaktadır. Ayrıca yapılan ultrasonik lokalizasyon işlemi ile (NÖL yöntemiyle) kartezyen konum hatalarının azaldığı ve bununla robot yönünün tahmininde bir iyileştirme sağladığı görülmektedir.



Şekil 4.5 Bir tur sola dönüş (syt) sonrasında planlanan yolda ortaya çıkan sapmalar.

mesafe ölçücü sensörlerle odyometrik sistemden gelen bilgiyi birleştirmek için kullanmışlardır [24].

Konum kestiriminde temel konu robot hareketlerinin kinematik denklemlerle modellenmesi olduğundan, bu modeldeki hatalar ilave edilen gürültülerle dikkate alınır. Çeşitli sensörlerin kendi yapılarından ve çevresel faktörlerden kaynaklanan gürültülerin etkisi altında yaptıkları ölçümlerden konum kestirimi yapabilmek için, modelden ve ölçümlerden kaynaklanan hataların istatistiksel durumlarının bilinerek, kullanılacak filtre ile en az değişim duyarlılığında en iyi konum kestirimi yapılabilmelidir. Bu amaçla yapılan çalışmada genel bir Kalman filtreleme (GKF) algoritması [25, 26] kullanılarak farklı sensörlerin belirli zaman aralıklarında robot konumunu en doğru şekilde tahmin edebilmesi sağlanmıştır.

Kullanılan GKF algoritması aynı yapıdaki ardışıl sensör bilgilerini işleyebilen ve farklı yapıdaki sensör bilgileri ile birleştirebilen bir yapıdadır. Esnek bir yapıya sahip olan algoritma için gerekli olan hesaplama zamanı en aza indirilmiştir. Her bir sensör sisteminin konumlandırmaya etkisinin ayrı ayrı dikkate alındığı bu esneklik sayesinde istenirse bazı sensör verileri birleştirme algoritması devreye girmeden kolaylıkla atılabilmektedir.

4.3.1 Robot Durum Uzayı Modeli

Mobil robot üzerinde düşük hızlarda çalışıldığından ve robot durumlarının tahminindeki örnekleme periyodunun robot dinamikleri ile karşılaştırıldığında küçük olmasından dolayı doğrusal ve açısal hızların bir örnekleme periyodu süresince sabit kaldığı varsayılmıştır (Ek-A). Bir başka deyişle, ortaya çıkabilecek bir ivme, sistem hatası olarak algılanacaktır. Bir robotun konfigürasyonu onun konumu ve yönü x_r , y_r , θ_r ile tanımlanırken (ilgilenilen durum değişkenleri), hareketi de Ek-A'da verilen (A.1.1) ve (A.1.2) eşitlikleri anlatılabilir. Şekil 4.7'den de görülebileceği gibi robot konumu olarak onun orta noktası alınmıştır. Yine robotun kendi y -ekseni (lokal koordinat sisteminde) boyunca hareket edebildiğini ve kendi z -ekseni etrafında dönebildiği dikkate alınarak doğrusal hız " v " ve açısal hız " w " eşitlik (4.23)'deki X_k durum vektörüne eklenmiştir. Bütün bu varsayımlar altında (A.1.1) ile (A.1.2) eşitliklerinin çözümü ile de ortaya çıkan durum vektörü X_k robotun döngüsel hareketleri için ($w \neq 0$) eşitlik (4.22a)'da, doğrusal hareketleri için de ($w=0$) eşitlik

(4.22b)'de verilmiştir. Her iki eşitliğin ortak ifadesini de eşitlik (4.23)'deki gibi gösterilebilir.

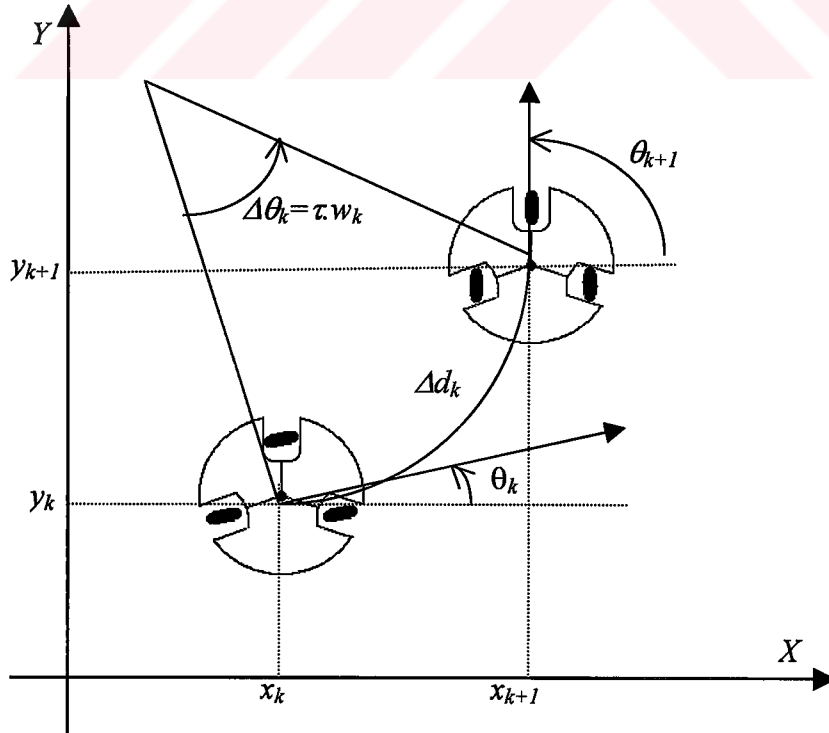
$w \neq 0$ için

$$\begin{bmatrix} x_r \\ y_r \\ \theta_r \\ v \\ w \end{bmatrix}_{k+1} = \begin{bmatrix} x_r \\ y_r \\ \theta_r \\ v \\ w \end{bmatrix}_k + \begin{bmatrix} (v_k / w_k) [\sin(\theta_k) - \sin(\theta_k + \tau \cdot w_k)] \\ (v_k / w_k) [\cos(\theta_k) - \cos(\theta_k + \tau \cdot w_k)] \\ \tau \cdot w_k \\ 0 \\ 0 \end{bmatrix} \quad (4.22a)$$

$w=0$ için

$$\begin{bmatrix} x_r \\ y_r \\ \theta_r \\ v \\ w \end{bmatrix}_{k+1} = \begin{bmatrix} x_r \\ y_r \\ \theta_r \\ v \\ w \end{bmatrix}_k + \begin{bmatrix} \tau \cdot v_k \cdot \cos(\theta_k) \\ \tau \cdot v_k \cdot \sin(\theta_k) \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.22b)$$

$$X_{k+1} = X_k + f(X_k) \quad (4.23)$$



Şekil 4.7 Bir örnekleme (τ) süresince değişen robot pozisyonu.

Her bir τ örnekleme periyodunda gerçek zamanda elde edilen ölçümler robot üzerindeki odyometrik sistemden, ultrasonik sensörlerden, manyetik pusuladan ve fiber-optik jiroskop dan sağlanmaktadır. Ultrasonik sensör verileri bir önceki Bölüm 4.2’de eşitlik (4.3) ve (4.4a, 4.4b)’de verildiği gibi konumu bilinen referans nesnelerin gözlenmesi şeklindedir. Yapılan sensör ölçümleri açık olarak (4.24a)’da verilen gözlem eşitliği ile özet olarak da eşitlik (4.24b)’de verilmiştir.

$$\begin{bmatrix} x_{odo} \\ y_{odo} \\ \theta_{odo} \\ x_{ult} \\ y_{ult} \\ \theta_{ult} \\ w_{jir} \\ \theta_{pus} \end{bmatrix}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_r \\ y_r \\ \theta_r \\ v \\ w \end{bmatrix}_k \quad (4.24a)$$

$$Y_k = C \cdot X_k \quad (4.24b)$$

Burada, x_{odo} , y_{odo} , θ_{odo} : odyometrik sistemden alınan robot konum ve yön bilgileri, x_{ult} , y_{ult} , θ_{ult} : ultrasonik sensörlerle ortamdaki referans nesnelerden sağlanan robot konum ve yön bilgileri, w_{jir} : fiber-optik jiroskoptan sağlanan robotun yaptığı açısal hız bilgisi, θ_{pus} : manyetik pusuladan sağlanan robot yön bilgisidir.

Durum uzayı modelindeki hatalar artı olarak ε_k gürültüsüyle, sensör ölçümlerini bozan etkenlerde η_k gürültüsüyle ifade edilmektedir. Bu durumda sistemin olasılıksal temsili eşitlik (4.25a) ve (4.25b)’de verildiği gibi ortaya çıkarılmış olur.

$$X_{k+1} = X_k + f(X_k) + \varepsilon_k \quad (4.25a)$$

$$Y_k = C \cdot X_k + \eta_k \quad (4.25b)$$

burada, ε_k ve η_k aralarında bir ilişki olmayan, ortak değişinti matrisleri sırası ile Q ve R olan sıfır ortalamalı beyaz gürültülerdir.

4.3.2 Sensör Verilerinin GKF ile Birleştirilmesi

Robot durum tahmini onun olasılıksal dağılımının ilk iki unsuru olan kestirimi yapılmış durum vektörünün ortalaması ve ortak değişinti matrisi ile ifade edilir [27, 28].

4.3.2.1 Durum Öngörüsü

$t=k\tau$ anındaki o ana kadar yapılmış bütün gözlemlerle bir sonraki adım olan $t=(k+1)\tau$ için robot durum öngörüsü, (4.25a)'da verilen doğrusal olmayan eşitlikteki a_k gürültüsü ihmal edilerek eşitlik (4.26) ile yapılabilir.

$$\hat{X}_{k+1|k} = \hat{X}_{k|k} + f(\hat{X}_{k|k}) \quad (4.26)$$

Bu öngörüdeki belirsizliği ifade edebilmek için onun doğrusallaştırılmasına ihtiyaç duyulur. Bu durumda öngörülen hata ortak değişinti matrisi eşitlik (4.27)'deki gibi verilebilir.

$$P_{k+1|k} = J_k \cdot P_{k|k} \cdot J_k^T + Q \quad (4.27)$$

burada, Q : Bölüm 4.3.1'de ki durum uzayı modelindeki gürültünün ortak değişinti matrisi, $P_{k|k}$: önceki adımdaki hata ortak değişinti matrisi, J_k : eşitlik (4.28)'de verildiği şekilde $\hat{X}_{k|k}$ öngörüsündeki $f(\hat{X}_{k|k})$ fonksiyonunun Jacobian matrisidir [29].

$$J_k = \frac{\partial f(\hat{X}_{k|k})}{\partial \hat{X}_{k|k}} \quad (4.28)$$

4.3.2.2 Durum Kestirimlerinin Güncellenmesi

Sensör ölçümlerinin istatistiksel olarak birbirinden bağımsız kaynaklardan sağlandığı varsayımıyla, çıkış gürültülerinin ortak değişinti matrisi köşegen bir yapıya sahiptir. (4.25b)'de verilen gözlem eşitliği sekiz alt parçaya eşitlik (4.29)'da verilen biçiminde bölünebilir [28].

$$y_k(i) = C(i) \cdot X_k + \eta_k(i) \quad , \quad i=1, \dots, 8 \quad (4.29)$$

Burada, $C(i)$: gözlem matrisi C 'nin satırlarıdır.

Daha sonra görülebileceği gibi bu ardışıl işlemin ana gayesi gerçek zamanda yapılacak olan matris terslerinin alınması işleminden kaçınmaktır. Ayrıca, ardışıl işlem, algoritmayı daha esnek hale getireceğinden algoritma kullanılan sensör ölçümlerinin sayısından bağımsız olacaktır.

Durum tahmininin güncellenmesi özyinelemeli olarak yapılmaktadır. Başlangıç durum vektörü ve durum ortak değişinti matrisi $\hat{X}(0) = \hat{X}_{k+1|k}$ ve $p(0) = p_{k+1|k}$ şeklindedir.

Özyinelemeli algoritma:

$i=1$ den 8'e kadar olan döngüde

a) ölçüm kalıntılarını (sonuçlar arasındaki farkı) hesapla:

$$r_i = y_k(i) - C(i) \cdot \hat{X}(0) \quad (4.30)$$

b) kalıntıyı izle ve ölçümün güvenilirliğini kontrol et. Örneğin burada σ_i ölçümün standart sapması olsun. Eğer $|r_i| \geq 2 \cdot \sigma_i$ ise ilgili sensör okumasının kabul edilmez olduğu ifade edilerek daha sonra yapılacak olan işlemlere dahil edilmeyerek bir sonraki " i " adımına geçilir.

c) yapılan ölçümün öngörü değişintisini (varyansını) hesapla:

$$s_i = C(i) \cdot p(i-1) \cdot C^T(i) + \sigma_i \quad (4.31)$$

d) filtre kazancını hesapla: yapılan ölçümlerin öngörü varyansları olan σ_i skaler olduğundan klasik Kalman filtreleme algoritması aşağıda verilen karşılığı olan değere azaltılmış olur:

$$K(i) = \frac{1}{s_i} p(i-1) \cdot C^T(i) \quad (4.32)$$

e) robot durumunu güncelle:

$$\hat{X}(i) = \hat{X}(i-1) + K(i) \cdot [y_k(i) - C(i) \cdot \hat{X}(i-1)] \quad (4.33)$$

f) güncellenmiş robot durum tahmininin hata ortak değişinti matrisini hesapla:

$$p(i) = p(i-1) - s_i \cdot K(i) \cdot K^T(i) \quad (4.34)$$

Algoritma sonucunda ortaya çıkacak güncellenmiş durum vektörü ve durum ortak değişinti matrisi: $\hat{X}_{k+1|k+1} = \hat{X}(8)$ ve $p_{k+1|k+1} = p(8)$ şeklinde olacaktır.

Yapılan ölçümler ve gözlemler sonucunda standart sapmalar manyetik pusulada 0.02, fiber-optik jiroskopta 0.01, odyometrik sistemde 0.087, ultrasonik sensörlerde 0.01 dir. Kullanılan sensörlerle modelin bütün değişkenleri tam olarak gözlenememektedir. Örneğin robot tekerleğinin doğrusal hızı v gibi. Bu nedenle bütün değişkenlerde oluşan belirsizliklerin tahmininde bazı sapmalar meydana gelmektedir. Uygulanan filtreleme algoritmasında bu belirsizlikleri azaltmak için harici farklı sensörler kullanılabilir. Örneğin robot tekerleğinin doğrusal hızını ölçebilecek bir akselometre gibi.

4.3.3 GKF Yönteminin Nomad200 Mobil Robotu Üzerinde Testi

Yöntemin Nomad200 mobil robotu üzerinde test edilebilmesi için Bölüm 4.2.4'de NÖL yöntemi için hazırlanan ve Şekil 4.4'de detaylı bir şekilde gösterilen laboratuvar ortamı kullanılmıştır. Burada da yöntemin testi için yapılan ilk deneyde robota pozitif x -yönünde (syt) planlanan yörünge iki defa izlettirilerek bitiş noktasında oluşan konum, yön ve bağıl mesafe hataları Tablo 4.3'de verilmiştir. Yapılan ikinci deneyde ise aynı işlem ters istikamette (sy) uygulanarak sonuçları Tablo 4.4'de gösterilmiştir. Robot konum ve yönünü tahmin ederken ve aynı zamanda planlanan yolunu izlerken eşitlik (4.24a)'da işaret edildiği gibi kendi üzerindeki kızılötesi sensörler hariç bütün sensörleri kullanmıştır. Kızılötesi

sensörlerin okuma aralığı çok az olduğu için (0-38cm.) nesnelere çok yaklaşmak gerekir. Bu durum robotun zaman zaman ortamdaki nesnelere çarpmasına sebep olabildiğinden bu sensörler tercih edilmemiştir. Elde edilen sonuçları değerlendirebilmek amacıyla sensör verileri ayrı ayrı saklanarak değişik sensör konfigürasyonları ile bulunabilecek konum ve yön tahminlerine yönelik bilgisayar simülasyonları yapılmıştır.

Tablo 4.3 ve 4.4’de GKF algoritmasının sadece odyometrik sensör ünitesi (odo) ile ultrasonik lokalizasyon ünitelerine (ult) uygulanmasının yanında bu iki sensör ünitesinin jiroskop ve pusula ile ayrı ayrı birleşimine (odo+jires, odo+pus, ult+jires, ult+pus) ve son olarak da bütün sensörlerin birleşimine yönelik uygulamaların sonuçlarını göstermektedir. Sadece odyometrik sensör ünitesinin kullanımı sonucunda elde edilen tahminlerde robot yönünde ve dolayısıyla konumunda oluşan hataların hızlı bir şekilde arttığı ve syt-yönündeki hataların sy-yönündeki hatalardan daima büyük olduğu görülmektedir. İlave edilen jiroskop bilgisi ile robot yönündeki hatalar azalmakta ve bu durum robot konumu tahminini daha doğru hale getirmektedir. Fakat odyometrik sensör ile manyetik pusulanın birleşiminden daha iyi sonuç beklenirken, yapılan çalışmalar bina içinde olduğundan (kapalı ortam) çoğu zaman pusuladan ya hiç bilgi alınamamış veya olması gereken değerin çok üstünde bilgi alınabilmiştir (bu gibi durumlarda algoritmaya ilave edilmemiştir). Bunun dışında normal gelen pusula değerleri ile odyometrik sistemden gelen bilgilerin birleşimi Tablo 4.3 ve 4.4’den görülebileceği gibi odyometre ve jiroskop birleşimine göre yakın veya daha az hatalı konum ve yön bilgisi vermektedir. Odyometrik sistemin bu iki yön sensörüyle birleştirilmesi ile syt-yönünde lokalizasyon tahmininde bir iyileşme, sy-yönünde ise bozulma görülmektedir.

Ultrasonik sensörlerle ve ortamdaki konumu bilinen referans silindirik nesnelerle yapılan ölçümlerden, odyometrik sisteme ve ona yardımcı sensör ünitelerine göre daha iyi randıman alınarak konum ve yön bilgisi doğruya yakın tahmin edilebilmektedir. Jiroskoptan alınan açısal hız ve yön bilgisinin ultrasonik lokalizasyon sistemine eklenmesi bu doğruluğa yeterli bir fayda sağlayamamaktadır. Manyetik pusula okumaları ise yine bina içi uygulamadan kaynaklanan probleminden dolayı GKF filtreleme algoritması tarafından çoğu zaman dikkate alınmadığından ultrasonik lokalizasyon sistemine katkısı çok az olmuştur. Şayet bu deneyler açık ortamlarda yapılsaydı, manyetik pusulanın katkısı çok daha fazla

olabilirdi. Bütün sensörlerin Kalman filtresiyle birleşiminden elde edilen konum ve yön tahmininin ultrasonik lokalizasyon ünitesinden elde edilen tahminlerin gerisinde kaldığı gözlenmektedir. Bununda nedeni odyometrik sensör ünitesinden kaynaklanan hatalı okumaların ultrasonik sensörlerden elde edilen doğru okumaları aşağı çekmesidir.

Tablo 4.3 Kalman filtresiyle iki defa sola dönüşle (syt) oluşan konum ve yön hataları

HATALAR SENSÖRLER	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
(odo)	30.7	-9.8	-15.0 ⁰	% 1.06
(odo)+(jir)	25.5	-6.3	-11.0 ⁰	% 0.87
(odo)+(pus)	28.4	-7.4	-12.0 ⁰	% 0.97
(odo)+(jir)+(pus)	21.0	-4.9	-9.5 ⁰	% 0.71
(ult)	4.5	3.8	3.5 ⁰	% 0.19
(ult)+(jir)	5.2	3.0	2.0 ⁰	% 0.20
(ult)+(pus)	4.0	2.5	1.5 ⁰	% 0.16
(ult)+(jir)+(pus)	3.8	2.0	0.5 ⁰	% 0.14
(ult)+(odo)	17.3	-6.2	-9.0 ⁰	% 0.61
Hepsi	14.6	-6.0	-7.5 ⁰	% 0.52

Tablo 4.4 Kalman filtresiyle iki defa sağa dönüşle (sy) oluşan konum ve yön hataları

HATALAR SENSÖRLER	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
(odo)	5.1	0.9	-7.0 ⁰	% 0.17
(odo)+(jir)	6.5	1.8	-5.5 ⁰	% 0.22
(odo)+(pus)	5.5	0.7	-2.0 ⁰	% 0.18
(odo)+(jir)+(pus)	6.1	1.0	-4.0 ⁰	% 0.20
(ult)	3.8	2.1	-3.0 ⁰	% 0.14
(ult)+(jir)	4.5	-3.2	-2.5 ⁰	% 0.18
(ult)+(pus)	3.5	1.5	-1.0 ⁰	% 0.13
(ult)+(jir)+(pus)	4.0	-2.2	-1.5 ⁰	% 0.15
(ult)+(odo)	6.5	-3.2	-4.5 ⁰	% 0.24
Hepsi	5.5	2.7	-1.0 ⁰	% 0.20

Sonuç olarak, ultrasonik sensörlerle ortamdaki referans nesneleri algılayan bir ünite ile manyetik pusula veya jiroskop bilgisinin bir GKF algoritmasıyla birleştirilmesi robot konum ve yönü tahmininde yapılan hataları en aza indirmiştir. Testlerin yapıldığı laboratuvar ortamında, bütün sensörlerin kullanımında, bir döngüde algoritma için gerekli olan hesaplama zamanı ortalama 950 msn. dir. Elde edilen bu veriler bir sonraki bölümde yer alan geliştirilmiş “Sınırlandırılmış Dinamik Sensör Füzyonu” (SDSF) yöntemi sonuçları ile karşılaştırılmak amacıyla kullanılmıştır.

4.4 Sınırlandırılmış Dinamik Sensör Füzyonu-SDSF Yöntemiyle Lokalizasyon

Çalışmanın bu kısmında, robot üzerindeki odyometrik, ultrasonik, pusula ve jiroskop sensörleri ile ortamda konumu bilinen nesneleri kullanarak gerçekleştirilen yeni bir lokalizasyon yöntemi tanıtılmaktadır. Burada sensör füzyon problemine geniş bir açıdan bakılarak geliştirilmiş bir Sınırlandırılmış Dinamik Sensör Füzyon (SDSF) algoritması tanıtılacaktır.

4.4.1 İki Boyutlu Sensör Füzyonu Probleminin Tanımı

Sensör füzyonu ile ilgili problemler zamana duyarlı veri toplayan bir çok farklı veya aynı yapıdaki sensörlerin kullanımına göre sınıflandırılabilir. Veri füzyonunun doğasında olan bu problemler 1996’da Brooks [30, 31] tarafından tartışılmıştır. Gerçekleştirilen çalışmada da bu probleme gerçek-zamanda iki boyutlu olarak bakılmıştır. Dinamik füzyonun doğasında olan zamana bağımlılık, kullanılacak olan algoritmanın çok hızlı ve verimli olmasını gerektirirken beraberinde de fazladan bazı belirsizlikleri gündeme getirir.

Probleme uygun bir çözüm sağlayabilmek için şu şekilde tanımlayabiliriz: Bir mobil robotun hareketi esnasında konumunu sağlayıcı “ N ” adet bağımsız $u_1, u_2, \dots, u_n$ ünitesi kullanılmaktadır. Burada her bir ünite kendi devre, sensör ve okuma algoritmasına sahip olduğundan sensör kelimesi yerine “*ünite*” kullanılmıştır. Sistemde yer alan üniteler aynı veya farklı cinsten olabilir.

Robot önceden belirlenen bir hızda hareket ederken her bir “ u_i ” ünitesi bağımsız olarak iki boyutlu $p_i = (x_i, y_i, \theta_i, x'_i, y'_i, \theta'_i)$ robot konum ve yön bilgileri ile doğrusal ve açısal hız bilgilerini sağlar. Burada her bir sensör sınırlı bir doğruluğa sahip iken

zaman zamanda hatalı okumalar yapabilir. Üniteler konumlandırma işlemine yönelik sensör okumalarında parçası olduğu platformdan veya kendi sisteminden kaynaklanan bazı belirsizliklere sahip olabilir. Ünitelere yansıyan bu belirsizlikleri konumsal gürültüler $(\Delta x, \Delta y)$, robot hızına bağlı olarak oluşan gürültüler $(\Delta x', \Delta y')$ ve okunan veriler ile işleme zamanları arasında veya buna benzer nedenlerle ortaya çıkan zamana ait gürültüler (Δt) şeklinde sınıflandırabiliriz.

Problemin çözümüne yönelik olarak, mobil robotun iki boyutlu ortamda izlediği yol, parçalı sürekli bir fonksiyon olan $f(t)$ ile tanımlanabilir. Dinamik sensör füzyonu problemlerinde $f(t_i)$ nin Taylor serilerindeki açılımı olan

$$f(t_i) + f'(t_i)(t - t_i) + f''(t_i)(t - t_i)^2 + f'''(t_i)(t - t_i)^3 + \dots + f^n(t_i)(t - t_i)^n + \dots$$

fonksiyonunu $t_i - \Delta t \leq t \leq t_i + \Delta t$ zaman aralığı için yazabiliriz. Burada Δt değeri küçük olduğundan $\Delta t \rightarrow 0$ iken $n \rightarrow \infty$ için Δt^n değerleri sıfıra yaklaşır. Bu nedenle, verilen herhangi bir $f(t)$ 'de sadece Δt nin ufak değerleri oluşacağından Taylor açılımının ilk iki terimi hariç diğerleri ihmal edilebilir.

4.4.2 Mobil Robot Konum ve Hızının Bulunması

Elde edilen Taylor açılıma Δt zamana ait gürültülerin yanında Δx_i ve Δy_i konumsal, $\Delta x'_i$ ve $\Delta y'_i$ hıza bağlı gürültüleri eklediğimizde mobil robotun “i” sensör ünitesine göre “x” ve “y” koordinatlarındaki konumu aşağıdaki (4.35a) ve (4.35b)'de verilen eşitliklerin oluşturdukları sınırlar içinde yer alacaktır.

$$x_i \pm \Delta x_i \pm (x'_i + \Delta x'_i) \cdot \Delta t_i \quad (4.35a)$$

$$y_i \pm \Delta y_i \pm (y'_i + \Delta y'_i) \cdot \Delta t_i \quad (4.35b)$$

Eşitliklerdeki “-” ifadesi “i” sensör ünitesinin okuduğu robot konumuna ait alt sınırı, “+” ifadesi de üst sınırı vermektedir. Burada alt sınır, mobil robot konumunun $t_i + \Delta t_i$ anında okunduğu varsayılarak robotun bulunması gerektiği yerden $\Delta x_i + (x'_i + \Delta x'_i) \cdot \Delta t_i$ kadar ileri gideceği göz önüne alındığında $t=t_i$ anı için robot

konumunun tekrar hesaplanmasıyla elde edilir. Bu durumda eşitlik (4.35a) ve (4.35b) kullanılarak $t=t_i$ için robot konumu x -koordinatlarında $(x_i)_{alt} = x_i - \Delta x_i - (x'_i + \Delta x'_i) \cdot \Delta t_i$ ve y -koordinatlarında ise $(y_i)_{alt} = y_i - \Delta y_i - (y'_i + \Delta y'_i) \cdot \Delta t_i$ şeklinde olacaktır. Aynı şekilde üst sınır hesabında, robot konumunun $t_i - \Delta t_i$ anında okunduğu varsayıldığında robotun bulunması gereken $t=t_i$ anı için $\Delta x_i + (x'_i + \Delta x'_i) \cdot \Delta t_i$ kadar geride olduğu dikkate alınır. Bu durumda yine eşitlik (4.35a) ve (4.35b) kullanılarak robot konumundaki üst sınır $t=t_i$ anı için x -koordinatlarında $(x_i)_{üst} = x_i + \Delta x_i + (x'_i + \Delta x'_i) \cdot \Delta t_i$, y -koordinatlarında ise $(y_i)_{üst} = y_i + \Delta y_i + (y'_i + \Delta y'_i) \cdot \Delta t_i$ şeklinde olacaktır.

Şekil 4.8’de A ve B isimleriyle adlandırılan iki bağımsız sensör ünitesinden t anında robot konumuna ait alınan $p_A = (x_a, y_a, \theta_a, x'_a, y'_a, \theta'_a)$, $p_B = (x_b, y_b, \theta_b, x'_b, y'_b, \theta'_b)$ bilgilerinin (4.35)’de verilen eşitliklerde kullanımı ile ortaya çıkan alt ve üst sınırlar gösterilmiştir. Her iki sensör ünitesine göre x ve y koordinatlarında robot konumuna ait oluşan alt ve üst sınırlar birer dikdörtgen meydana getirmektedir. Şekilde gösterim kolaylığı açısından t -zaman boyutu gösterilmemiştir. Şayet gösterilseydi her bir sensöre ait zamansal $t_i \pm \Delta t_i$ sınırlar ile x - y koordinatlarındaki karşılığı sensörlerin oluşturduğu dikdörtgenlerin birleşimi dikdörtgensel prizmatik yapılar meydana getirecekti.

Geliştirilen SDSF algoritması ile kullanılan sensör ünitelerine ait dikdörtgensel prizmatik yapılar oluşturularak bunların kesişme yerlerinde meydana gelen ufak dikdörtgensel prizmatik yapı ortaya çıkartılır (Şekil 4.8’deki taralı alan). Algoritmanın bulduğu kesişimsel prizmatik yapının tam ortası olası robot konumu x_r ve y_r olarak atanır.

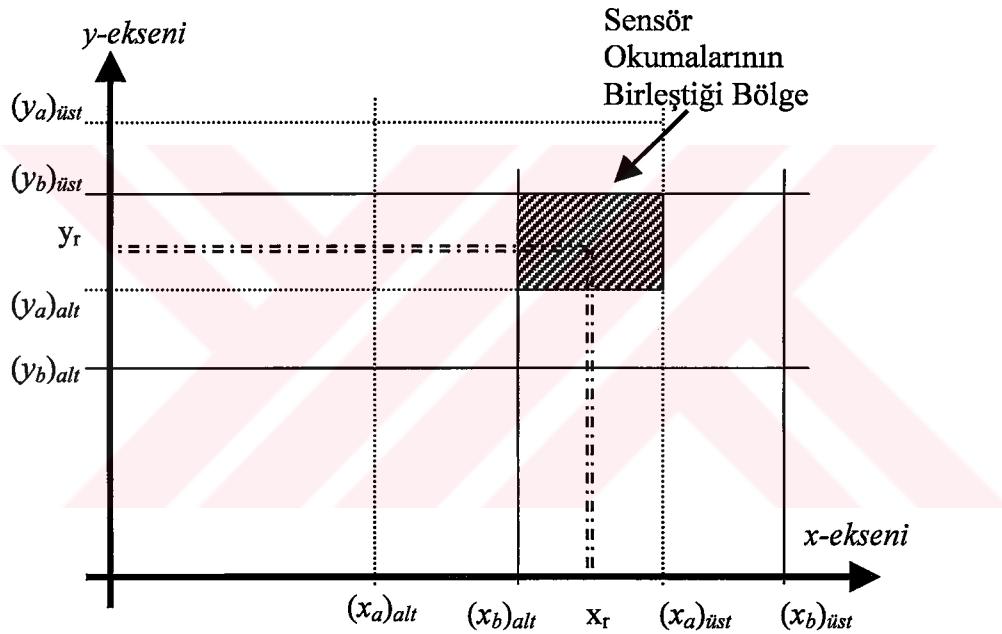
Bir robot, navigasyon algoritmasına girilen komutlar ile belirli bir doğrusal (v) ve açısal (w) hızlarda hareket ettirilebilir. İlgili sensör üniteleri tarafından yapılan okumalar kullanılarak bu hızların bileşenleri ve eklenen gürültüleri $(x'_i \pm \Delta x'_i, y'_i \pm \Delta y'_i, \theta'_i \pm \Delta \theta'_i)$ hesaplanabilir. Örneğin, Şekil 4.9’da bir “ i ” sensör ünitesi $t=t_n$ anında robot konum ve yönünü x_n, y_n ve θ_n , $t=t_{n+1}$ anında ise x_{n+1}, y_{n+1} ve θ_{n+1} olarak ölçmektedir. “ v ” doğrusal ve “ w ” açısal hız komutlarıyla ilerleyen

robotun $t=t_{n+1}$ anındaki doğrusal hız bileşenleri eşitlik (4.36a) ve (4.36b), açısal hız ise (4.36c) kullanılarak hesaplanabilir.

$$x'_{n+1} = \frac{|x_{n+1} - x_n|}{t_{n+1} - t_n} \quad (4.36a)$$

$$y'_{n+1} = \frac{|y_{n+1} - y_n|}{t_{n+1} - t_n} \quad (4.36b)$$

$$\theta'_{n+1} = \frac{|\theta_{n+1} - \theta_n|}{t_{n+1} - t_n} \quad (4.36c)$$



Şekil 4.8 A ve B sensör okumalarının 2-boyutlu füzyonu.

Bu durumda $t=t_{n+1}$ anında “i” sensör ünitesinden elde edilen $p_i(t_{n+1}) = (x_{n+1}, y_{n+1}, \theta_{n+1}, x'_{n+1}, y'_{n+1}, \theta'_{n+1})$ bilgisi iki boyutlu robot konum ve yönü ile onlara ait hızları içerecektir. Yine bu örneğe ait hız belirsizlikleri SDSF algoritmasında kullanılmak üzere eşitlik (4.37a) ve (4.37b)’den bulunabilir.

$$\Delta x'_{n+1} = \begin{cases} |x'_{n+1} - v \cdot \cos(\theta_{n+1} - \theta_n)|; \text{şayet } w = 0 \text{ ise} \\ \left| x'_{n+1} - \frac{F_x(t_{n+1})}{t_{n+1} - t_n} \right|; \text{şayet } w \neq 0 \text{ ise} \end{cases} \quad (4.37a)$$

$$\Delta y'_{n+1} = \begin{cases} |y'_{n+1} - v \cdot \sin(\theta_{n+1} - \theta_n)|; \text{şayet } w = 0 \text{ ise} \\ |y'_{n+1} - \frac{F_y(t_{n+1})}{t_{n+1} - t_n}|; \text{şayet } w \neq 0 \text{ ise} \end{cases} \quad (4.37b)$$

Açısal hıza ait belirsizlikler hakkında detaylı bilgi Bölüm 4.4.3’de verilmiştir.

Örnekteki $t=t_{n+1}$ anı için Δx_{n+1} ve Δy_{n+1} konum hataları ise robot tekerinin dönüp dönmemesine bağlı olarak eşitlik (4.38a) ve (4.38b)’de verildiği gibi, herhangi bir “i” sensör okumasının $t=t_{n+1}$ anında tespit ettiği robot konumu x_{n+1}, y_{n+1} bilgileri ile, $t=t_n$ anında SDSF algoritmasıyla bulunan tahmini robot konumu $x_r(t_n), y_r(t_n)$ bilgilerine robot “v” hızı sayesinde ilave olunan yol arasındaki fark ile bulunur.

$$\Delta x_{n+1} = \begin{cases} |x_{n+1} - x_r(t_n) - v \cdot (t_{n+1} - t_n) \cdot \cos(\theta_{n+1} - \theta_n)|; \text{şayet } w = 0 \text{ ise} \\ |x_{n+1} - x_r(t_n) - F_x(t_{n+1})|; \text{şayet } w \neq 0 \text{ ise} \end{cases} \quad (4.38a)$$

$$\Delta y_{n+1} = \begin{cases} |y_{n+1} - y_r(t_n) - v \cdot (t_{n+1} - t_n) \cdot \sin(\theta_{n+1} - \theta_n)|; \text{şayet } w = 0 \text{ ise} \\ |y_{n+1} - y_r(t_n) - F_y(t_{n+1})|; \text{şayet } w \neq 0 \text{ ise} \end{cases} \quad (4.38b)$$

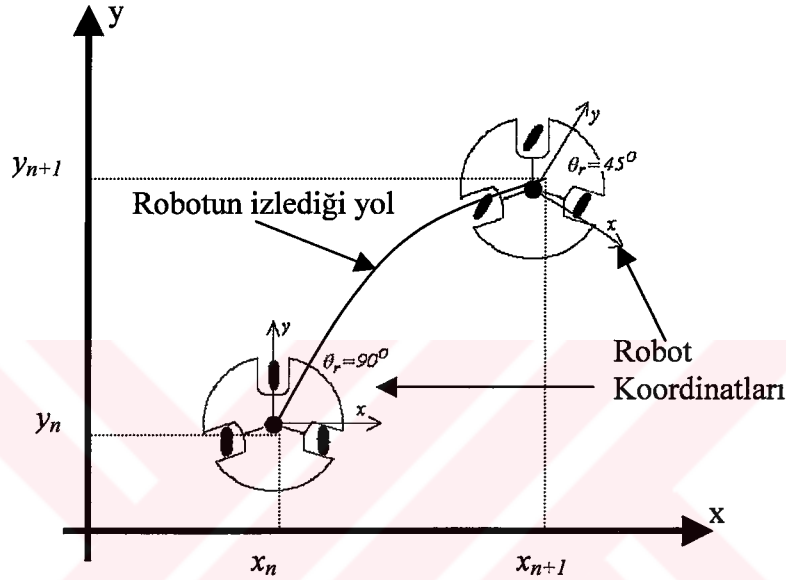
Son olarak bu örneğe ait, $t=t_{n+1}$ anında (4.35)’deki eşitliklerde yer alan zamansal Δt_{n+1} gürültüsü eşitlik (4.39)’dan konum ve hız belirsizlikleri kullanılarak bulunabilir.

$$\Delta t_{n+1} = \frac{\sqrt{(\Delta x_{n+1})^2 + (\Delta y_{n+1})^2}}{\sqrt{(x'_{n+1} + \Delta x'_{n+1})^2 + (y'_{n+1} + \Delta y'_{n+1})^2}} \quad (4.39)$$

Gerçekleştirilen sistemde robotun x ve y-koordinatlarındaki konumunun bulunması amacıyla sadece iki ünite kullanılmıştır. Bunlardan biri mobil robot üzerindeki “*odyometrik ünite*”, diğeri ise robot etrafındaki konumu bilinen nesneleri tespit ederek robot konumunu tahmin etmeye çalışan “*ultrasonik konumlandırma ünitesi*” dir. Her iki üniteden elde edilen konum bilgileri üç boyutun her birinde alt ve üst sınırlara $(x_{alt}, x_{üst}, y_{alt}, y_{üst}, t - \Delta t, t + \Delta t)$ sahiptir. Bu sınırların oluşturduğu üç boyutlu şekle (iki koordinat, bir zaman boyutu) “prizmatik-dikdörtgen” ismini verebiliriz. Burada amaç, her iki üniteden elde edilen prizmatik-dikdörtgenin

kesiřmesi ile ortaya ıkabilecek en kk prizmatik-dikdrtgenin bulunmasıydı. Elde edilecek bu blge robot konumunun da iinde yer aldıđı optimal bir blge olacaktır.

Algoritma her iki niteden elde edilen konum bilgilerinden bir kesiřim blgesi ıkartamadıđı durumlarda “*ultrasonik konumlandırma nitesi*” nden sađlanan bilgiyi referans almaktadır. Bununda nedeni Blm 4.3’de GKF algoritması ile yapılan alıřmada bu nitenin “*odyometrik nite*” ye gre daha iyi sonu vermesidir.



řekil 4.9 Global koordinat sistemindeki robot koordinatları.

4.4.3 Robot Ynnn Bulunması

Gerekleřtirilen alıřmada kullanılan Nomad200 Mobil robotu holonomik olmayan bir yapıya sahip olduđundan, verilen “w” aısal hız komutuyla her  tekeri aynı anda dnmektedir. Bu nedenle, řekil 4.9’da verildiđi gibi robot bař aısı “ θ_r ” robot tekerleklerinin global koordinatlarda yaptıđı aı olarak dikkate alınmıřtır. Tekerleklerin baktıđı dođrultu, aynı zamanda robot konumunun merkezini oluřturduđu robot koordinat sisteminin y-ekseni olarak alınmıřtır. Bir bařka deyiřle, “ θ_r ” robot koordinat sisteminin y-ekseni ile global koordinat sisteminin x-eksenleri arasındaki aıdır.

Mobil robot bař aısı “ θ_r ” robot zerindeki drt ayrı nite tarafından llr. Bu niteler sırası ile “*odyometrik nite*”, “*manyetik pusula nitesi*”, *fiber-optik jiroskop*

ünitesi” ve “*ultrasonik konumlandırma ünitesi*” dir. Yapılan her bir ölçümün zamana ait gürültüleri de dikkate alınarak ortaya çıkartılacak alt ve üst sınırları eşitlik (4.40) kullanılarak bulunabilir.

$$\theta_i \pm (\Delta\theta_i \pm (\theta'_i + \Delta\theta'_i) \cdot \Delta t_i) \quad (4.40)$$

Burada θ'_i robot tekerinin sağa veya sola dönüşlerinde meydana gelen açısal hız, $\Delta\theta'_i$ ise tekere dönüş sağlayan sistemden kaynaklanan açısal hıza ait gürültüler, $\Delta\theta_i$ ve Δt_i ise yapılan ölçüme eklendiği varsayılan açısal gürültü ile zamana ait gürültülerdir. Bölüm 4.4.2’de verilen örnekte bir “*i*” sensör ünitesinin t_n ve t_{n+1} anlarında yaptığı robot konum ve yön ölçümlerinden t_{n+1} anındaki açısal teker hızı θ'_{n+1} eşitlik (4.36c)’den hesaplanmaktaydı. Bu hıza bağlı belirsizlik $\Delta\theta'_{n+1}$ ise eşitlik (4.41)’den bulunabilir.

$$\Delta\theta'_{n+1} = |\theta'_{n+1} - w| \quad (4.41)$$

Burada, w : algoritmada robota yapacağı açısal teker hızı miktarını bildiren komuttur. Örnekteki $t=t_{n+1}$ anı için $\Delta\theta_{n+1}$ yön hatası ise eşitlik (4.42)’de verildiği gibi herhangi bir “*i*” sensör ünitesinin $t=t_{n+1}$ anında okuduğu robot yönü “ θ_{n+1} ” bilgisi ile, $t=t_n$ anında SDSF algoritmasıyla bulunan tahmini robot yönü “ $\theta_r(t_n)$ ” bilgisine robot tekerinde oluşan “ w ” hızı sayesinde ilave olunan açı arasındaki fark ile bulunur.

$$\Delta\theta_{n+1} = |\theta_{n+1} - \theta_r(t_n) - w \cdot (t_{n+1} - t_n)| \quad (4.42)$$

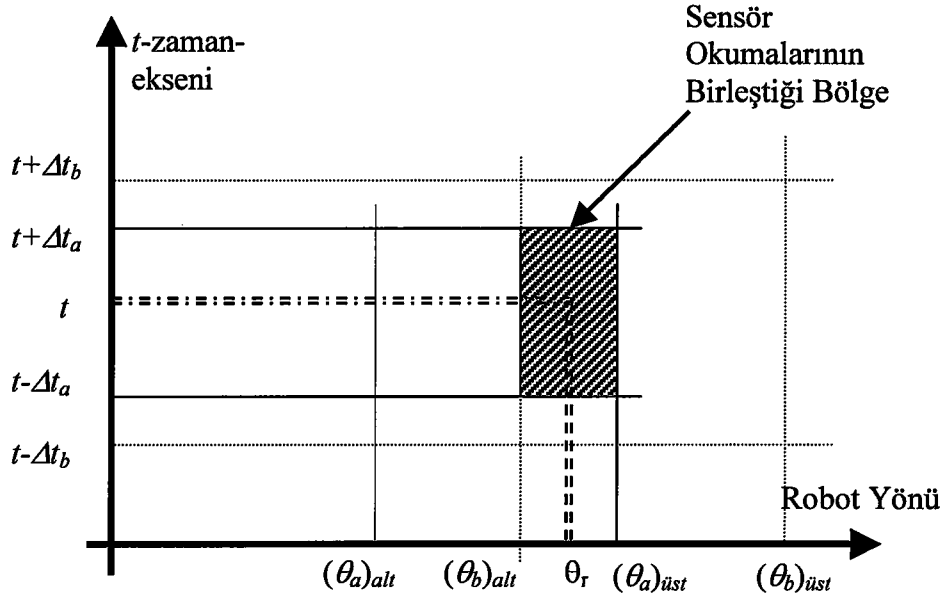
Örnekteki “*i*” sensör ünitesinin “*odyometrik ünite*” veya “*ultrasonik konumlandırma ünitesi*” olması durumunda $t=t_{n+1}$ anında bunlara ait zamansal gürültüler eşitlik (4.39) ile bulunabilmektedir. Şayet örnekteki “*i*” sensör ünitesi bir manyetik pusula veya fiber-optik jiroskop ise $t=t_{n+1}$ anında bunlara ait zamansal gürültüler ise eşitlik (4.43) kullanılarak hesaplanabilir.

$$\Delta t_{n+1} = \frac{\Delta\theta_{n+1}}{\theta'_{n+1} + \Delta\theta'_{n+1}} \quad (4.43)$$

Şekil 4.10'da A ve B bağımsız sensör üniteleri robot yönü " θ_r "ı bulmak için kullanılmıştır. Şekilde gösterildiği gibi robot baş açısının içinde yer aldığı sınırları ortaya çıkaran eşitlik (4.40)'da kullanılan "-" ifadesi alt sınır, "+" ifadesi ise üst sınırı tanımlamaktadır. Robot baş açısının $t_i + \Delta t_i$ anında ölçüldüğü varsayıldığında ölçülen " θ_i " açı bilgisine " w " döngüsel hızının yarattığı $(\theta'_i + \Delta\theta'_i) \cdot \Delta t_i$ açısı ilave edilecektir. Bu durum $t=t_i$ anına çekildiğinde bu farkın çıkarılmasıyla robot baş açısının alt sınırı olan $\theta_i - \Delta\theta_i - (\theta'_i + \Delta\theta'_i) \cdot \Delta t_i$ ifadesi elde edilir. Benzer bir şekilde robot baş açısının $t_i - \Delta t_i$ anında ölçüldüğü varsayıldığında ise, ölçülen " θ_i " açı bilgisi içinde " w " döngüsel hızının yarattığı $(\theta'_i + \Delta\theta'_i) \cdot \Delta t_i$ açısı kadar bir eksiklik oluşacaktır. Bu durum yine $t=t_i$ anına çekildiğinde eksik kalan bu farkın eklenmesiyle robot baş açısının üst sınırını veren $\theta_i + \Delta\theta_i + (\theta'_i + \Delta\theta'_i) \cdot \Delta t_i$ ifadesi elde edilir. Şekilde gösterim kolaylığı olması bakımından sadece iki adet sensör ünitesi (A ve B) kullanılarak, bunlara ait " θ_a " ve " θ_b " robot baş açısı bilgilerinin üst ve alt sınırları ile yine bu ünitelere ait Δt_a ve Δt_b zamansal ölçüm gürültüleri dikkate alınmıştır. Şekildeki yön ekseninde beliren üst ve alt sınırlar ile zamansal eksende bunlara ait ölçüm gürültülerinin oluşturduğu üst ve alt sınırlar birer dikdörtgen meydana getirmektedir. Her bir ünite için oluşturulan bu dikdörtgenlerin kesişmesi ile şekilde taralı alan olarak gösterilen bir bölge ortaya çıkar. Robot baş açısı " θ_r " $t=t_i$ anında bu bölge içinde yer alır. Gerçekleştirilen SDSF füzyon algoritmasının robot yönünün bulunması aşamasında, "*odyometrik*", "*ultrasonik konumlandırma*", "*manyetik pusula*" ve "*fiber-optik jiroskop*" ünitelerinden sağlanan sırası ile θ_{odo} , θ_{ult} , θ_{pus} , ve θ_{jir} robot yönü bilgilerinin alt ve üst sınırları ile bu ünitelere ait Δt_{odo} , Δt_{ult} , Δt_{pus} ve Δt_{jir} zamansal gürültülerin oluşturduğu sınırlardan dikdörtgen yapılar ortaya çıkartılır. Daha sonra, oluşan bu dikdörtgen yapıların ara kesitinden oluşan dikdörtgen alan Şekil 4.10'daki taralı bölge gibi bulunarak (robot yönünün içinde yer aldığı optimal bölge) tam orta nokta noktası " θ_r " robot yönü olarak atanır.

Hatalı sensör okumaları bu optimal bölgenin dışında kalır. Şayet sensörler arasında hiçbir kesişme sağlanamaz ise algoritma tarafından güvenilirliklerine göre sırası ile "*fiber-optik jiroskop*", "*ultrasonik konumlandırma*" ve son olarak da "*odyometrik sensör*" ünitesinden sağlanan yön bilgisi " θ_r " robot yönü olarak atanır. Çalışılan ortamın kapalı olmasından dolayı manyetik pusula bilgisi sağlıklı elde

edilemediğinden bu gibi durumlarda algoritma tarafından dikkate alınmaz. Çalışılan ortam açık bir alan olması durumunda (açık oto parkları gibi) öncelik manyetik pusulaya verilebilirdi.



Şekil 4.10 Robot baş açısı θ_r 'ı veren A ve B sensör okumalarının üst ve alt sınırları.

4.4.4 SDSF Yönteminin Nomad200 Mobil Robotu Üzerinde Testi

Yapılan çalışmanın bu kısmında lokalizasyon amaçlı geliştirilen SDSF sensör füzyon yönteminin Nomad-200 mobil robotu üzerinde uygulanması ile elde edilen sonuçlar verilmiştir. Bu bölümde verilen bütün sonuçlar mobil robot üzerinde bulunan odyometrik, ultrasonik konumlandırma, manyetik pusula ve fiber-optik jiroskop ünitelerinin lokalizasyon amaçlı yaptıkları gözlemlerin geliştirilen sensör füzyon algoritması ile işlenmesiyle elde edilmiştir.

Geliştirilen SDSF yöntemini mobil robot üzerinde test edebilmek ve diğer yöntemlerle karşılaştırmak amacıyla yine Şekil 4.4'de verilen laboratuvar ortamında planlanan yol kullanılmıştır. Yapılan testlerde robot hızı 26 cm/sn.'de sabit tutulmakta ve 2 sn.'de bir sensörlerden okuma yapılmaktadır. Yöntemin sağlıklı ve açık bir şekilde uygulanabilmesi için yapılan testler üç aşamalı olarak gerçekleştirilmiştir. Bunlar:

- i. Yöntemin odyometre (odo), jiroskop (jir) ve pusula (pus) sensör üniteleriyle testi,

- ii. Yöntemin ultrasonik konumlandırma (ult), jiroskop (jir) ve pusula (pus) sensör üniteleriyle testi,
- iii. Yöntemin odyometre (odo), ultrasonik konumlandırma (ult), jiroskop (jir) ve pusula (pus) sensör üniteleriyle testi.

4.4.4.1 SDSF Yönteminin Odyometre (odo), Jiroskop (jir) ve Pusula (pus) Sensör Üniteleriyle Testi,

Yapılan testin bu aşamasında daha önce farklı yöntemlerle hata analizi detaylı olarak yapılan robot üzerindeki odyometre ünite ile buna ek olarak manyetik pusula ve fiber-optik jiroskop sensör üniteleri ayrı ayrı ve birlikte kullanılmaktadır. Bölüm 4.3’de, GKF algoritması kullanılarak odyometrik sensör sistemi ile Şekil 4.4’deki laboratuvar ortamında yapılan lokalizasyon deneylerinin sonuçları, karşılaştırma amacıyla Tablo 4.5 ve 4.6’da gösterilmiştir. Yine Şekil 4.4’deki laboratuvar ortamında Bölüm 4.3’deki deneylere benzer şekilde, SDSF yönteminin testi için yapılan birinci deneyde, planlanan pozitif-x (syt) yönündeki yörünge robota iki defa izlettirilerek, kullanılan sensör ünitelerine göre varış noktasında oluşan konum, yön ve bağıl mesafe hataları Tablo 4.5’de verilmektedir. Yöntemin aynı sensörler ile test edildiği ikinci deneyde ise robot sürekli sağa dönüşlerle (sy) yörüngesini iki defa turlamıştır. Varış noktasında oluşan konum, yön ve bağıl mesafe hataları da Tablo 4.6’da gösterilmektedir.

Tablo 4.5 Odyometre, jiroskop ve pusula ile sola (syt) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları

<i>HATALAR</i> <i>SENSÖRLER</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
GKF(odo)	30.7	-9.8	-15.0 ⁰	% 1.06
(odo)+(jir)	18.2	1.2	-8.0 ⁰	% 0.60
(odo)+(pus)	24.6	-7.1	-13.0 ⁰	% 0.85
(odo)+(jir)+(pus)	14.3	1.0	-4.0 ⁰	% 0.47

Tablo 4.6 Odyometre, jiroskop ve pusula ile sağa (sy) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları

<i>HATALAR</i> <i>SENSÖRLER</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
GKF(odo)	5.1	0.9	-7.0 ⁰	% 0.17
(odo)+(jir)	5.0	-1.5	-6.0 ⁰	% 0.17
(odo)+(pus)	4.1	1.0	-3.5 ⁰	% 0.14
(odo)+(jir)+(pus)	3.5	1.2	-2.0 ⁰	% 0.12

Lokalizasyon amaçlı syt-yönündeki testlerde robot üzerindeki odyometre ünitesine jiroskopun katkısı pusuladan daha fazla olmuştur. Şayet testler açık bir alanda yapılsaydı, pusulanın katkısı daha fazla olabilecekti. Zaman zaman okunamayan pusula bilgileri odyometrik üniteden okunan robot yönü bilgisine yapması gerektiği olumlu katkıyı yapamamıştır. Her üç sensörün katılımıyla sürekli sola dönüşler şeklinde yapılan testlerde en az hata ile robot lokalizasyonu gerçekleştirilebilmektedir.

Testlerin sy-yönünde yapılan kısmında ise odyometrik üniteye jiroskopun katkısı olumsuz iken pusulanın katkısı olumludur. Fakat jiroskop ve pusulanın beraber kullanımı odyometrik hataları en aza çekebilmektedir. Tablo 4.5 ve 4.6'ya dikkat edildiğinde sy-yönünde yapılan test sonuçları syt-yönünde elde edilen sonuçlardan daha iyidir. Bunun anlamı; robot genel olarak sola dönüşlerde sağa dönüşlere göre daha fazla hata yapmaktadır. Bu yüzden sağa dönüşler şeklindeki testlerde jiroskop ve pusulanın katkısı az olmuştur. Tablo sonuçlarına bakıldığında, her üç sensörün kullanımı ile sy-yönünde ölçülen en az %0.12'lik bağıl mesafe hatası GKF sonuçlarından (%0.20) daha iyi çıkmıştır. Bu aşamada algoritma için gereken hesaplama zamanı bir döngü için ortalama 650 msn. dir.

4.4.4.2 SDSF Yönteminin Ultrasonik Konumlandırma (ult), Jiroskop (jir) ve Pusula (pus) Sensör Üniteleriyle Testi,

SDSF yöntemi için gerçekleştirilen testin ikinci aşamasında ise Bölüm 4.2'de hata analizi detaylı bir şekilde anlatılan ultrasonik konumlandırma ünitesi ile buna ek olarak fiber-optik jiroskop ve manyetik pusula sensör üniteleri ayrı ayrı ve birlikte kullanılmaktadır. Bölüm 4.3'de GKF yöntemiyle Şekil 4.4'de verilen laboratuvar ortamında ultrasonik konumlandırma ünitesi ile yapılan lokalizasyon testlerinin

sonuçları Tablo 4.3 ve 4.4’de verilmişti. Aynı sonuçlara, bu aşamada yapılan test sonuçları ile karşılaştırmak amacıyla Tablo 4.7 ve 4.8’de tekrar yer verilmektedir. Şekil 4.4’deki ortamda, bir önceki bölümdeki testlerde olduğu gibi, iki türlü sola (syt) ve sağa (sy) dönüşlerle yapılan yörünge izlemeleri sonucunda ultrasonik konumlandırma ünitesinin jiroskop ve pusula ile ayrı ayrı ve birlikte kullanımına uygulanan yöntemin hata sonuçları sırası ile yine Tablo 4.7 ve 4.8’de verilmektedir.

Tablo 4.7 Ultrasonik sistem, jiroskop ve pusula ile sola (syt) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları

<i>HATALAR</i> <i>SENSÖRLER</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
GKF(ult)	4.5	3.8	3.5 ⁰	% 0.19
(ult)+(jir)	3.0	2.5	4.0 ⁰	% 0.13
(ult)+(pus)	4.1	3.3	2.0 ⁰	% 0.17
(ult)+(jir)+(pus)	2.4	2.5	2.5 ⁰	% 0.11

Tablo 4.8 Ultrasonik sistem, jiroskop ve pusula ile sağa (sy) dönüşlerle iki tur sonucunda oluşan konum ve yön hataları

<i>HATALAR</i> <i>SENSÖRLER</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [$^{\circ}$]	$\Delta \rho / \rho$
GKF(ult)	3.8	2.1	-3.0 ⁰	% 0.14
(ult)+(jir)	3.2	-1.1	2.0 ⁰	% 0.11
(ult)+(pus)	2.8	2.0	-1.5 ⁰	% 0.11
(ult)+(jir)+(pus)	3.0	1.7	-1.0 ⁰	% 0.11

Lokalizasyon amaçlı yapılan testlerde, Bölüm 4.2’deki NÖL yöntemiyle bulunan en küçük bağıl mesafe hatası olan %0.18 Bölüm 4.3’deki GKF yöntemiyle %0.14’e ve bu kısımdaki SDSF yöntemiyle yapılan testlerde ise %0.11’e kadar azaltılabilmektedir. Yapılan her iki testte de, jiroskop ve pusulanın ultrasonik konumlandırma ünitesine katkısı, bağıl mesafe hatasının azaltılması ve robot yönünün az bir hata ile bulunması şeklinde olmuştur. Bu aşamada ise bir döngüde algoritma için gereken hesaplama zamanı ortalama 850 msn. dir.

4.4.4.3 SDSF Yönteminin Odyometre (odo), Ultrasonik Konumlandırma (ult), Jiroskop (jir) ve Pusula (pus) Sensör Üniteleriyle Testi,

SDSF yönteminin test edileceği bu son aşamada ise mutlak konum ve yön bilgisi veren robot üzerindeki odyometrik sensör ünitesi ile Bölüm 4.2'deki ultrasonik konumlandırma ünitesi beraber kullanılmıştır. Yöntemin bu iki sensör ünitesiyle Şekil 4.4'deki ortamda, daha önceki testlerde olduğu gibi robota sola (syt) ve sağa (sy) dönüşler şeklinde yaptırılan iki tur hareketi sonucunda ortaya çıkan konum ve yön hataları Tablo 4.9 ve 4.10'da verilmektedir. Ayrıca bu iki sensör ünitesine jiroskop ve pusula sensör ünitelerinin ayrı ayrı ve birlikte ilaveleri ile ortaya çıkan test sonuçları da aynı tablolarda gösterilmektedir.

Tablo 4.9 Odometrik ve ultrasonik sistemin jiroskop ve pusula ile beraber sola (syt) dönüşler şeklindeki iki turu sonucunda oluşan konum ve yön hataları

<i>HATALAR</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [°]	$\Delta \rho / \rho$
<i>SENSÖRLER</i>				
(odo)+(ult)	-4.8	-2.1	-3.0 ⁰	% 0.17
(odo)+(ult)+(jir)	-5.0	-1.9	-3.5 ⁰	% 0.18
(odo)+(ult)+(pus)	-4.1	1.8	1.5 ⁰	% 0.15
(odo)+(ult)+(jir)+(pus)	-5.4	1.1	1.0 ⁰	% 0.18

Tablo 4.10 Odometrik ve ultrasonik sistemin jiroskop ve pusula ile beraber sağa (sy) dönüşler şeklindeki iki turu sonucunda oluşan konum ve yön hataları

<i>HATALAR</i>	Δx [cm]	Δy [cm]	$\Delta \theta$ [°]	$\Delta \rho / \rho$
<i>SENSÖRLER</i>				
(odo)+(ult)	4.0	3.1	-3.5 ⁰	% 0.17
(odo)+(ult)+(jir)	4.3	2.9	-3.0 ⁰	% 0.17
(odo)+(ult)+(pus)	3.5	3.0	-1.0 ⁰	% 0.15
(odo)+(ult)+(jir)+(pus)	3.8	3.1	-1.0 ⁰	% 0.16

Sola dönüşlerle (syt) yapılan testlerde herhangi bir algoritma tarafından işlenmemiş odyometrik bilgiden kaynaklanan %1.86'lık bağıl mesafe hatasının ultrasonik konumlandırma ünitesi tarafından %0.17'lere çekildiği Tablo 4.9'da görülmektedir. Bu iki üniteye pusulanın katkısı olumlu iken jiroskop herhangi bir fayda

sağlamamıştır. Aksine bütün sensörlerin katılımıyla yapılan testlerde lokalizasyon hatalarını arttırıcı bir rol oynamaktadır.

Sağa doğru dönüşlerle (sy) yapılan testlerde ise, işlenmemiş odyometrik bilgisinden kaynaklanan %0.20'lik bağıl mesafe hatasının, ultrasonik konumlandırma ünitesi ve manyetik pusulanın katkılarıyla %0.15'lere kadar çekildiği görülmektedir.

Uygulanan SDSF algoritmasında sensörler arasında ortak bir prizmatik yapı bulunmadığında daima olumlu sonuç veren ultrasonik konumlandırma ünitesinden sağlanan bilgiler referans alınmaktadır. Tablo 4.9 ve 4.10'da, farklı sensör konfigürasyonlarına rağmen, birbirine yakın sonuçlar elde edilmesi, bu aşamada jiroskop ve pusulanın beklendiği kadar faydalı olamadığını göstermektedir.

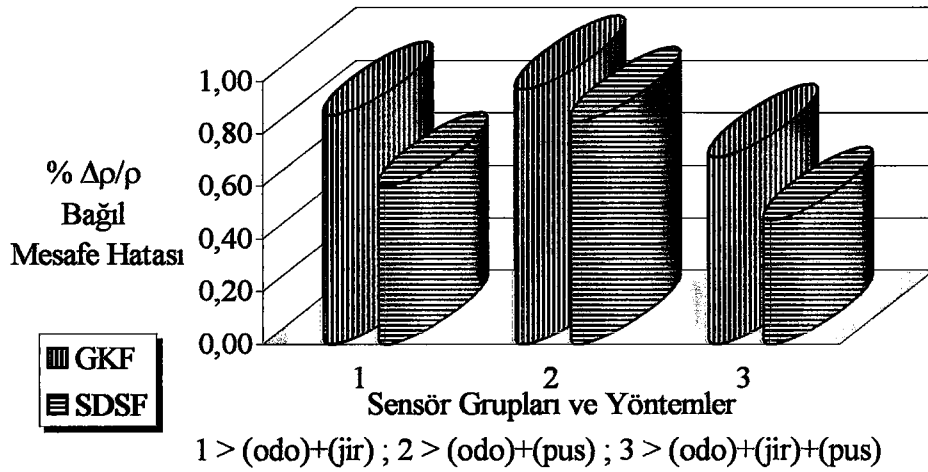
Geliştirilen SDSF algoritmasının robot üzerindeki odyometrik sistem, ultrasonik konumlandırma, fiber-optik jiroskop ve manyetik pusula sensörlerinin katılımı ile uygulanması sonucunda elde edilen sola dönüşlerde %0.18, sağa dönüşlerde %0.16 olan bağıl mesafe hataları GKF algoritmasında sırası ile %0.52 ve %0.20 bulunmuştur. Ayrıca testler sırasında sadece bir döngüde algoritma için gereken hesaplama zamanı sensör okumaları dahil SDSF yönteminde ortalama 900 msn. iken GKF yönteminde ortalama 950 msn. dir. Yapılan testler sonucunda ortaya çıkan bu iki faktör SDSF algoritmasının oluşturulan laboratuvar şartlarında GKF algoritmasına göre daha iyi sonuçlar verdiğini göstermektedir. Ancak yapılan bütün lokalizasyon amaçlı testler içinde, GKF algoritması sadece ultrasonik konumlandırma ünitesi ve pusula ile %0.13 gibi en iyi bağıl mesafe hatasını hesaplayabilmiştir.

4.5 SDSF Yönteminin GKF Yöntemiyle Karşılaştırılması

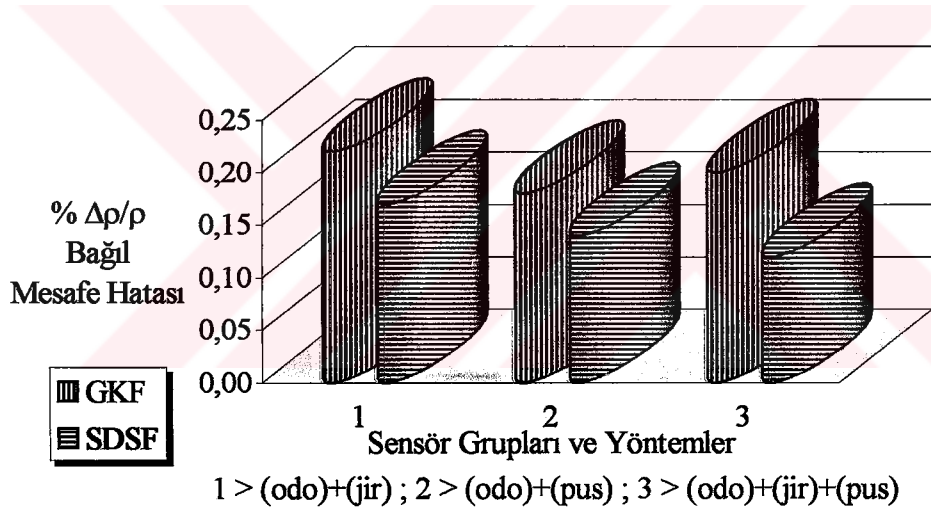
Kapalı veya açık alanlarda bir mobil robot lokalizasyonu için kullanılan yöntemlerden en tanınmış ve en fazla kullanılanı Bölüm 4.3'de verilen Genelleştirilmiş Kalman Filtreleme-GKF yöntemidir. Bu nedenle, geliştirilen Sınırlandırılmış Dinamik Sensör Füzyonu-SDSF yöntemi ile bir karşılaştırma yapabilmek amacıyla GKF yöntemi seçilmiştir.

Her iki yöntemin testi için Şekil 4.4'deki laboratuvar ortamında mobil robotun lokalizasyon amaçlı takip edeceği yol planlanmıştır. Planlanan yörünge üzerinde mobil robota s_{yt} ve s_y yönlerinde ayrı ayrı ikişer tur attırılarak 2sn.'de bir üzerindeki odyometre, ultrasonik, manyetik pusula ve fiber-optik jiroskop sensörlerinden veriler alınmaktadır. Odyometrik sensör ünitesi robot tekerleklerine doğrudan bağlıdır ve herhangi bir işlem yapılmadan mutlak robot konumu ve yönü bilgilerini vermektedir. Ultrasonik sensörler ise robot etrafındaki konumu bilinen referans nesnelerin kendisine olan uzaklığını ifade eden mesafe bilgilerini sağlar. Daha sonra bu mesafe bilgileri Bölüm 4.2'de geliştirilen Newton-Özyinelemeli Lokalizasyon-NÖL yöntemiyle işlenerek mutlak robot konum ve yönünü veren bilgi elde edilmektedir. Ultrasonik sensörlerle NÖL yönteminin oluşturduğu bu yapıya “ultrasonik lokalizasyon ünitesi” adı verilerek lokalizasyon konusunda yapılan bütün çalışmalarda kullanılmaktadır. Robot üzerinde yer alan diğer sensörlerden pusula mutlak robot yönü bilgisi, jiroskop ise robotun yaptığı açısal hızı ölçerek bağlı robot yönü bilgisi verir.

Odyometrik sensör üzerine Bölüm 4.2.4'de yapılan ilk testlerde robotun Şekil 4.4'deki yörüngeyi s_{yt} ve s_y yönlerinde ayrı ayrı iki defa turlaması sonucunda bağlı mesafe hataları sırası ile %1.86 ve %0.20 olarak bulunmuştu. Odyometrik sensör verilerinin Bölüm 4.3'deki GKF algoritmasıyla işlenmesi sonucunda bağlı mesafe hataları s_{yt} -yönünde yapılan testlerde %1.06'ya s_y -yönünde yapılan testlerde %0.17'ye düşürülebilmektedir. Bu durum GKF algoritmasının mobil robot konumunu tahmininde oldukça etkili olduğunu göstermektedir. Yine GKF algoritmasıyla Bölüm 4.3.3'de yapılan testlerde odyometrik ünite ile jiroskop ve pusula bilgileri ayrı ayrı ve birlikte birleştirilerek bağlı mesafe hatası s_{yt} -yönünde sırası ile %0.87, %0.97 ve %0.71, s_y -yönünde ise %0.22, %0.18 ve %0.20 olarak bulunmaktadır. Aynı şartlar içinde Bölüm 4.4'de verilen SDSF algoritması ile yapılan testlerde ise odyometrik ünitenin jiroskop ve pusula ile ayrı ayrı ve birlikte birleştirilmesi sonucunda oluşan bağlı mesafe hataları Tablo 4.5 ve 4.6'da verildiği gibi s_{yt} -yönünde sırası ile %0.60, %0.85 ve %0.47, s_y -yönünde ise %0.17, %0.14 ve %0.12 olarak hesaplanmıştır. Bu durum, her iki yönteme uygulanan s_{yt} -yönündeki testler için Şekil 4.11'de, s_y -yönündeki testler için de Şekil 4.12'de karşılaştırmalı olarak verilmektedir.

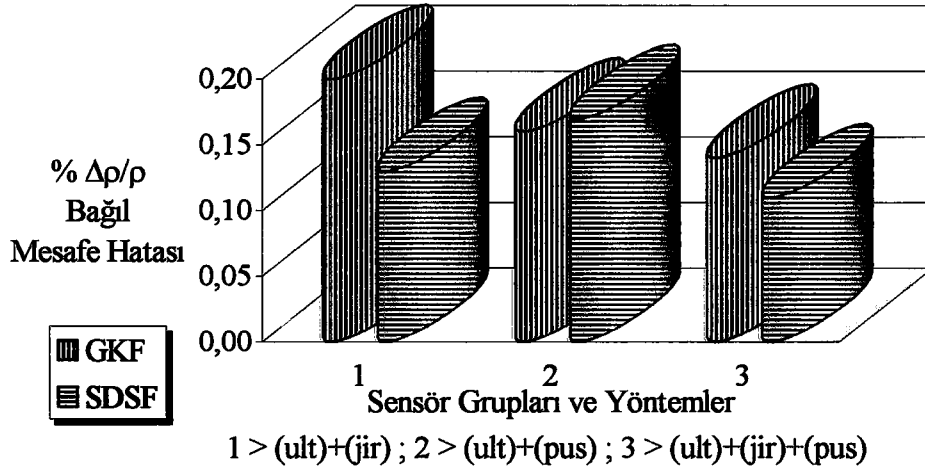


Şekil 4.11 Sola (syt) iki tur dönüşler sırasında odyometre, pusula ve jiroskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.

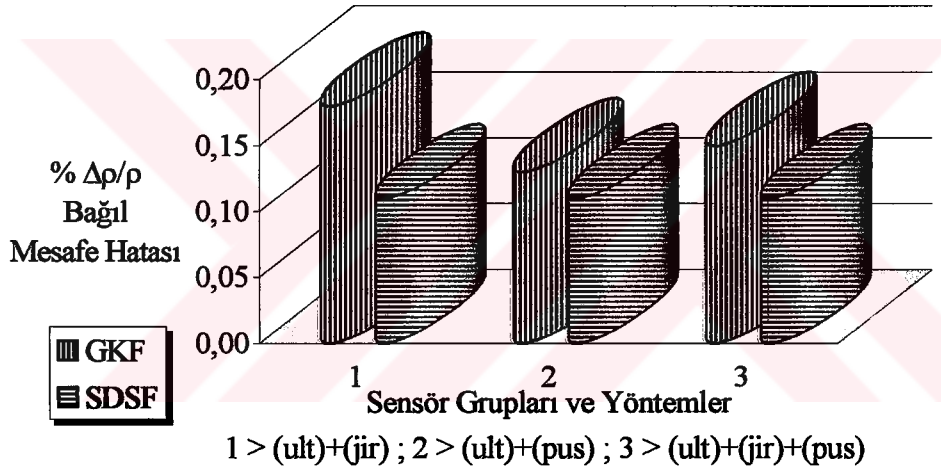


Şekil 4.12 Sağa (sy) iki tur dönüşler sırasında odyometre, pusula ve jiroskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.

Ultrasonik konumlandırma ünitesi ile jiroskop ve pusulanın ayrı ayrı ve beraber kullanımı ile gerçekleştirilen *syt* ve *sy*-yönlerindeki testler sonucunda oluşan Tablo 4.3, 4.4, 4.7 ve 4.8'deki bağıl mesafe hataları ise sırası ile Şekil 4.13 ve 4.14'de karşılaştırmalı olarak verilmektedir.



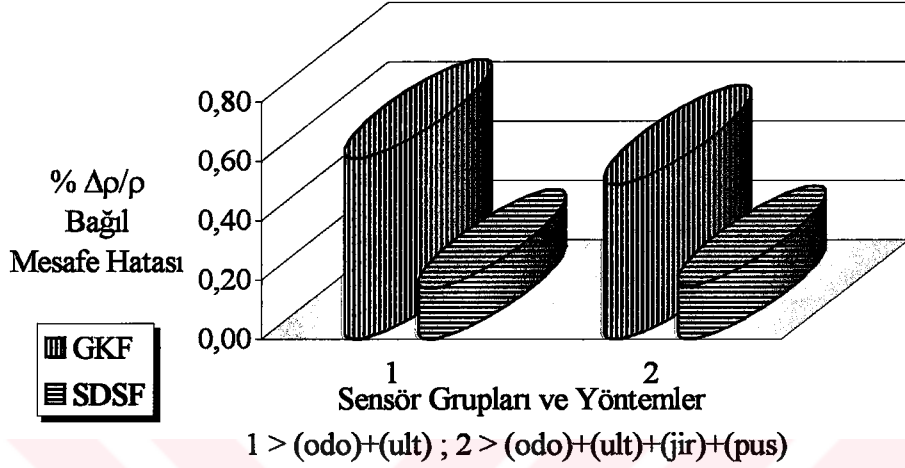
Şekil 4.13 Sola (syt) iki tur dönüşler sırasında ultrasonik konumlandırma ünitesi, pusula ve jiroskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.



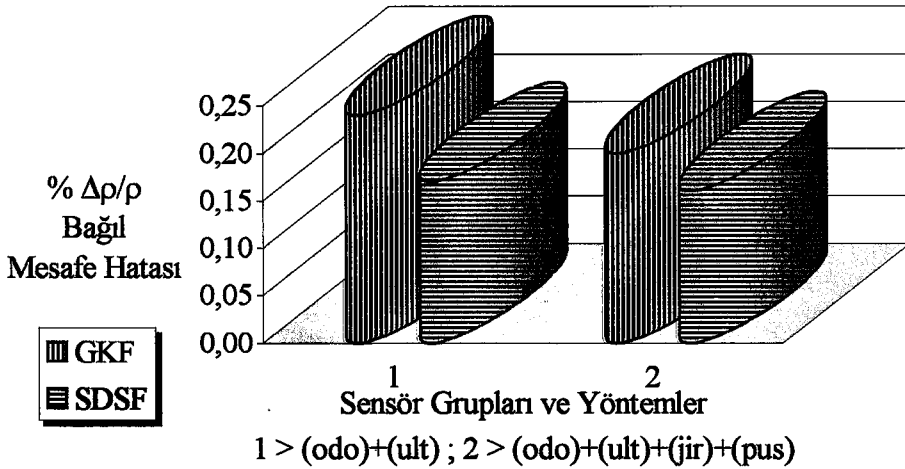
Şekil 4.14 Sağa (sy) iki tur dönüşler sırasında ultrasonik konumlandırma ünitesi, pusula ve jiroskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.

Son olarak robot üzerinde yer alan kızılötesi sensörler hariç bütün sensörlerin katılımıyla yapılan testlerde, öncelikle Tablo 4.3, 4.4, 4.9 ve 4.10 sonuçları verilen odyometre ile ultrasonik lokalizasyon ünitesinden sağlanan bilgiler sırası ile GKF ve SDSF algoritmalarıyla birleştirilmektedir. Daha sonra bu iki sensör ünitesine jiroskop ve pusula bilgileri de ilave edilerek meydana getirilen grafikler *syf* ve *sy-yön*lerinde sırası ile Şekil 4.15 ve 4.16'da verilmektedir. Grafiklerden de ortaya çıktığı gibi bütün sensörlerden elde edilen verilerin geliştirilen SDSF algoritmasında

birleştirilmesi GKF ile yapılan birleştirmenin verdiği sonuçların altında kalmaktadır. Bu durum SDSF algoritması ile yapılacak mobil robot lokalizasyon işleminin oldukça iyi çalışan GKF algoritmasına göre daha hassas sonuçlar verebileceğini göstermektedir.



Şekil 4.15 Sola (syt) iki tur dönüşler sırasında Odyometre, ultrasonik konumlandırma ünitesi, pusula ve jiroskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.



Şekil 4.16 Sağa (sy) iki tur dönüşler sırasında odyometre, ultrasonik konumlandırma ünitesi, pusula ve jiroskop sensörlerine uygulanan lokalizasyon yöntemlerinin sonuçları.

4.6 Sonuç

Gerçekleştirilen çalışmanın bu bölümünde, bir mobil robot navigasyonu için gereken en önemli unsurlardan biri olan mobil robot lokalizasyonu konusunda yapılan çalışmalar anlatılmıştır. İlk olarak, tarihsel süreç içerisinde mobil robot lokalizasyonunda yapılan gelişmeler aktarılmış ve bizim 1996 yılında bu konuda yaptığımız ve yayınladığımız [22] ilk çalışma olan Newton Özyinelemeli Lokalizasyon-NÖL yöntemi tanıtılmıştır. Yarı bilinen ortamlarda robot üzerindeki ultrasonik sensörlerle, robot etrafında bulunan, konumu en yakın bilinen referans nesnelerin robota olan uzaklığının ölçülmesi ve bu ölçümlerin özyinelemeli bir yöntemle değerlendirilmesi prensibine dayalı bu yöntem, Nomad200 mobil robotu üzerinde test edilerek ortaya çıkan sonuçlar şekil ve tablolar halinde verilmiştir.

Bu bölümde ayrıca, mobil robot lokalizasyonu konusunda kendini kanıtlamış ve bir çok araştırmacı tarafından kullanılan bir yöntem olan Genelleştirilmiş Kalman Filtrelemesi-GKF yöntemi ile yine burada yapılan çalışmalar sırasında geliştirilen Sınırlandırılmış Dinamik Sensör Füzyonu-SDSF yöntemi tanıtılmıştır. Her iki yöntem Şekil 4.4’de verilen laboratuvar şartlarında planlanan bir yörünge üzerinde Nomad200 mobil robotu ile test edilerek çıkan sonuçlar tablolar halinde gösterilmiştir. Bu tablolarda, robotun yörüngesini izledikten sonra ortaya çıkan konum, yön ve yüzde şeklinde bağıl mesafe hatalarına yer verildiğinden geliştirilen SDSF yöntemi kolaylıkla GKF yöntemiyle karşılaştırılabilmiştir.

Bu bölümün son kısmında ise, her iki yöntemden elde edilen tablolar, karşılaştırma işlemini kolaylaştırabilmek amacıyla görsel hale getirilerek grafiklerle temsil edilmiştir. Bu grafikler incelendiğinde mobil robot lokalizasyonu konusunda SDSF yönteminin GKF yöntemine göre daha az konum hataları ürettiği görülmüştür.

Burada lokalizasyon konusunda yapılan bütün çalışmalar bir sonraki bölümde yer alan çevre haritalama işleminde ve 6ncı bölümdeki mobil robot navigasyonunda geniş bir şekilde kullanılmaktadır.

5. ÇOKLU SENSÖR VERİ FÜZYONU TEKNİKLERİ İLE ÇEVRE HARİTALAMASI

5.1 Giriş

Bir mobil robotun çalıştığı ortam içinde kendisine verilen görevleri ortam içindeki hareketli veya hareketsiz nesnelere çarpmadan güvenli bir şekilde yapabilmesi için çevrenin sayısal bir haritasına sahip olması ve bunu güncelleyebilmesi gereklidir. Bu sebeple, mobil robotlarda çevre haritalama ile bir çevre modellemesi ve deneysel çalışma gerektirir. Günümüze kadar yapılan araştırmalarda genel olarak çevrenin temsiline yönelik iki temel yaklaşım geliştirilmiştir. Bunlar;

- i) Gridleme (ızgaralama) Tabanlı Olasılıksal Modelleme-GTOM,
- ii) Öznitelik bulmaya yönelik modellemedir.

Bugüne kadar yapılan çalışmalar, genellikle birinci yöntem üzerine yoğunlaşmıştır. Bunun en büyük nedeni geniş ışıma aralığına sahip ucuz ultrasonik sensörlerin ızgaralama tabanlı haritalama işleminde kolayca kullanımıdır. Bu konuda yapılan çalışmalarda da, Nomad200 üzerinde gelişmiş bir ultrasonik sensör sistemi bulunduğundan, bu çalışmada da ızgaralama tabanlı olasılıksal modelleme tercih edilmiştir. Öznitelik bulmaya yönelik modellemelerde [32] ise kamera ve laser tarayıcılar gibi yüksek çözünürlü sensör sistemleri kullanılarak tespit edilen nesnelerin biçimsel özelliklerine dayalı bir çevre modellemesi gerçekleştirilir.

GTOM biçimindeki bir modellemenin temel prensibinde, robotun çalışma ortamı hücre adı verilen küçük kare parçacıklarına bölünür. Her bir hücre kendi üzerindeki alan içerisinde bir engelin bulunabilme olasılığına göre bir değer alır.

Bu konuda yapılan ilk çalışmalarda, robot sensörleri ile etrafı tararken hareketsiz kalmaktaydı ve farklı konumlardan yapılan birkaç tarama çevre modelinin oluşturulması amacıyla birleştirilmekteydi. Moravek ve Elfes [33] 1985 yılında

olasılıksal dağılım fonksiyonlarını kullanarak kesin olmayan ultrasonik mesafe verilerini haritalama amaçlı ızgaralara yerleştirebilen bir yöntemi tanıttılar. Bu yöntemde, üzerinde 24 ultrasonik sensör bulunan bir halka taşıyan robot, bulunduğu çevrenin panoramik taramasını yaparken hareketsiz kalmaktaydı. Fryxell [34] ise 1987 yılında sonarları bir ışın demeti şeklinde modelleyen ve ızgara hücresinin sonik ışın demetleriyle kaç kez görüldüğünü veya görülebildiğini gözlemleyebilen bir olasılıksal yöntem kullanır. Yöntem sonucunda yapılan oylamayla hücrenin ortamdaki bir nesne tarafından işgal edilip edilmediğine karar verilir. Beckerman ve Oblow [35] 1988 yılında aynı yöntemi sonar sensörlerini 18^0 lik bir konik açı ile modelleyerek kullandılar. Bu yöntemde, bir hücrenin dört farklı durumu vardır. Bunlar, yapılan sensör gözlemleri sonuçlarına göre;

- i) Üzerinde bir nesne olup olmadığına karar verilemeyen “ihtilafli” hücreler,
- ii) Üzerinde bir nesne olduğu tespit edilen “işgal edilmiş” hücreler,
- iii) Üzerinde bir nesne olmadığı tespit edilen “boş” hücreler,
- iv) Üzerinde herhangi bir sensör gözlemi yapılmamış “bilinmeyen” hücrelerdir.

Yöntemin uygulanmasında ortaya çıkan sonuçlar Fryxell’in yöntemiyle benzerlik göstermesine rağmen ortamın daha detaylı bir haritasını çıkarabilmiştir. Daha sonra 1989 yılında Elfes [36] tarafından yapılan çalışmada işgal edilmiş ızgara kavramı kullanılmıştır. Bir önceki yöntemle benzer konseptlere sahip olmasına rağmen bu yöntemde sensörler Gauss belirsizliği ile modellenmekte ve daha kesin bir matematiksel model uygulanmaktaydı. Cahut ve arkadaşları [37] tarafından 1998 yılında geliştirilen diğer basit bir yöntemde ise olasılıksal dağılım kullanılmamıştır. Yöntemde ultrasonik sensörlerden alınan mesafe verileri kümelenendirilerek, her bir kümeyle bir hat birleştirme algoritması uygulanmıştır.

Robotun hareket halindeyken sensörleriyle çevre taraması yapabildiği durumlar ele alındığında, Gilbreath ve Everett [38] 1988 yılında geliştirdikleri yöntemde daha hızlı ve az hesaplama zamanına sahip algoritmalarını ROBART-II robotu üzerinde uyguladılar. Yöntemde basitleştirilmiş bir olasılıksal yaklaşım kullandılar. Diğer bir tarafta ise Koren ve Borenstein [39] 1991 yılında olasılık fonksiyonunu tamamen kaldırarak daha da basitleştirilmiş bir model geliştirdiler. Bu modelde ortamdaki

nesneyi yakalayan sensör okumasının konik ekseninde yer alan tek bir hücrenin sayısal değeri arttırılmaktadır. Kısaca HIMM denilen hareket halindeyken histogramik haritalama yönteminde ultrasonik sensör verileri istatistiksel olarak iki boyutlu bir ızgara histogramı şeklinde modellenir. Robot hareket halindeyken sensörlerin örneklenmesiyle bir histogramik olasılık dağılımı oluşturulur. Bu dağılım vasıtasıyla aynı hücrelerin birden fazla gözlenmeleriyle aldıkları değerler daha yüksek bir doğruluk ifade ederler. Mesafe verilerinin işlenmesine yönelik diğer bir yaklaşımda ise Vandorpe ve arkadaşları [40] 1996 yılında mesafe verilerine olasılık fonksiyonunu uyguladılar. Onların çevre modeli sensörler tarafından yakalanan nesnelerin oluşturabileceği düz hatların ve dairelerin çıkarılması üzerinedir.

Üzerinde çalışılan laboratuvar ortamının modellenmesi hakkında detaylı bilgi Bölüm 5.2’de verilmektedir. Bölüm 5.3’de ise Bayes yöntemi ile yapılan bir çevre modellemesine yer verilmiştir. Bu modelleme Bölüm 5.4’de verilen geliştirdiğimiz Ardışıl Temel Bileşenler-ATB yöntemiyle bir karşılaştırma yapabilme amacını taşımaktadır. Bayes ve ATB yöntemlerinin testinde robot hareket halindeyken ultrasonik ve kızılötesi sensörler beraber kullanılmaktadır. Sensörlerin etrafı sürekli taramaları sonucunda karşılaşılan bir engel, eko oluşturursa, o engelin üzerinde bulunduğu hücrenin değeri onun robota olan uzaklığı ile ilintili olarak Bölüm 5.3 ve 5.4’deki gibi çeşitli yöntemlerle hesaplanabilir. Ortamdaki katı hareketsiz nesneler üzerinde bulundukları hücrenin engel temsil değerinin yüksek çıkmasına sebep olurlar. Diğer bir tarafta ise nadiren meydana gelen yanlış okumalar ilgili hücre değerlerini daha az arttırır.

5.2 Mobil Robotlarda Sensörlere Dayalı Çevre Modellemesi

Robot hareket halindeyken çeşitli sensörlerden alınan çevre bilgileriyle ortamın tam bir sayısal haritası çıkartılabilir ve gerekli anlarda da güncellenebilir. Yapılan çalışmada bu amaçla robot üzerindeki ultrasonik ve kızılötesi sensörler kullanılmaktadır. Kızılötesi sensörlerin okuma aralığının kısa olmasından dolayı (0-35cm.) ancak yakın çevre bilgisinde ultrasonik sensörlerle beraber kullanılabilir. Daha uzak mesafelerde ise ($d > 35cm.$) ultrasonik sensörler tek başına çevre haritalama işlemini gerçekleştirmektedir. Her iki sensör modelinin birbirine benzemesinden dolayı Bölüm 5.2.1’de sadece ultrasonik sensörlerle yapılan

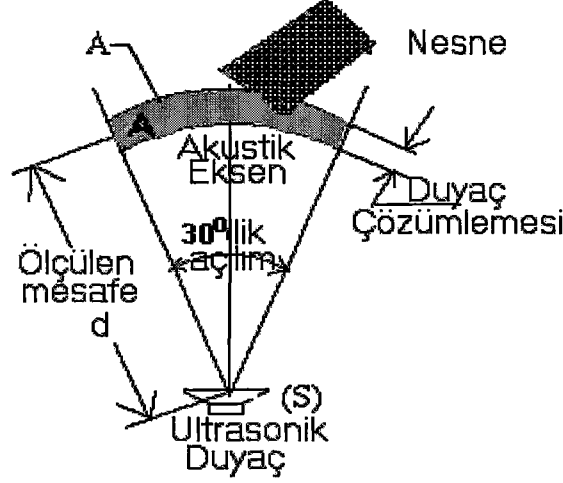
evre haritalama konusunda detaylı bilgi verilerek Blm 5.2.2’de anlatılan laboratuvar ortamının modellenmesine bir alt yapı oluřturulmuřtur.

5.2.1 Ultrasonik Sensrlerle evre Haritalama

Ultrasonik mesafe lmlerinden bir evre haritası oluřturabilmek iin, ortamın ncelikle panoramik olarak taranması gereklidir. Birok mobil robot bu amaca ynelik olarak zerinde eřitli sayılarda ultrasonik sensr tařıyan bir yatay halka ile donatılmıřtır [41-44]. Bu yatay halka ile ortamın taranmasında herhangi bir dnen para veya motora ihtiya duyulmadıėından ortamın 360⁰ lik panoramik taraması hızlı bir řekilde yapılabilmektedir. Fakat, daha nce Blm 3.3’de anlatıldıėı gibi karřılıklı bir apraz karřımayı sebep olmasın diye btn sensrler bir defada ateřlenemez. apraz karřımayı azaltmak iin zel bir sensr ateřleme dizisine ihtiya duyulur, ama bu da btn ortamın taranması iin gereken zamanı arttırır. Referanslarda verilen alıřmalarda ultrasonik sensrlerle ortamın tam bir panoramik taraması iin geen zaman 100 ile 500 msn. arasında deėiřmektedir. Bu konuda Blm 5.3 ve 5.4’de yapılan alıřmalarda ise ortalama tarama zamanı 300 msn. ye ayarlanmıřtır.

Izgara-tipi bir evre modelinde engellerin olasılıksal temsilinde nclk yapan bir yntem Carnegie-Mellon niversitesinde-CMU geliřtirilmiřtir [33, 44, 45]. Sonu evre modeli zellikle ultrasonik, kamera ve yakınlık sensrleri gibi farklı yapıdaki sensrlerden gelen verilerin birleřtirilmiř temsili [44] iin uygundur. Byle ızgara-tipi bir evre modelinde, robotun alıřma sahası hcreler řeklindeki kare elemanların oluřturduėu iki boyutlu dizilerle temsil edilir. Her bir hcre kendi zerinde bir engelin bulunma olasılıėını gsteren bir kesinlik deėerine (*CV*) sahiptir. *CV* deėerleri kullanılan sensrn karakteriřtiėini de dikkate alan bir olasılıksal fonksiyon ile gncellenir. rneėin ultrasonik sensrler konik bir aılım ile etrafını grrler. Robot zerinde kullanılan ultrasonik sensr [46] bu konik aılım iindeki en yakın bir nesnenin kendisine olan mesafesini lerken onun bu konik aılım iinde hangi aısal konumda olduėunu belirleyemez. řekil 5.1’den de grlebileceėi gibi bir “*d*” mesafe lm ultrasonik sensrn “*A*” konik alanı iinde herhangi bir yere konumlanmış bir nesnenin sonucudur. Bununla beraber, akustik eksene (konik aılımın merkez ekseni) yakın bir yerde bulunan bir nesne bu eksene uzak olan bir nesneye gre daha uygun bir ekonun retilmesini saėlar [47]. Sonu olarak, CMU

yöntemiyle Şekil 5.1'deki "A" alanı içindeki her bir hücreye CV değerleri atanır. Yapılan bu işlemde, kullanılan olasılıksal fonksiyon tarafından akustik eksene yakın hücrelere daha yüksek değerler atanır.



Şekil 5.1 Ultrasonik sensörün iki-boyutlu konik açılımı.

Okunan mesafe bilgilerinden, Şekil 5.1'deki "S" ve "A" arasında kalan bölümle ilgili de bir yorum yapılabilir. Şayet " d " mesafesindeki bir nesneden eko alındıysa, bu durumda "S" ile "A" arasındaki bölgede herhangi bir nesne olmamalıdır. [33] ve [45]'de verilen referanslarda bu boş alanda yer alan hücreler olasılık fonksiyonunun negatif değer atamalarıyla temsil edilir. Yapılan çalışmalarda ise bu boş alandaki hücrelerin engel temsil değerleri sıfır alınmıştır.

CMU'nun CV değerli ızgara-tipi yönteminde [33, 44, 45] mobil robot, üzerinde 24 ultrasonik sensör bulunan halkası ile etrafın panoramik taramasını yaparken sabit kalmaktadır. Her bir hücrenin CV değeri 24 mesafe bilgisinin her birine uygulanan bir C_x olasılıksal fonksiyonu ile güncellenir. Daha sonra robot yeni bir konuma hareket eder, durur ve yaptığı işlemleri tekrarlar. Robotun bu şekilde bütün bir odayı gezmesi sonucunda oluşan CV değerleri odanın doğruya yakın bir haritasını temsil eder. Bu harita güncellenirken, bir global yol-planlama yöntemi özyinelemeli bir şekilde çalışarak robotun izleyeceği yolu planlar.

Elfes [36] tarafından bu konuda yapılan en son çalışmada CV değerlerine benzer işgal edilmiş ızgaraları tanıtır. İşgal edilmiş ızgara yöntemi ultrasonik sensörleri Gauss belirsizliği ile modeller ve özyinelemeli okumalara daha karmaşık bir matematiksel model uygularlar.

5.2.2 Çalışılan Laboratuvar Ortamının Modellenmesi

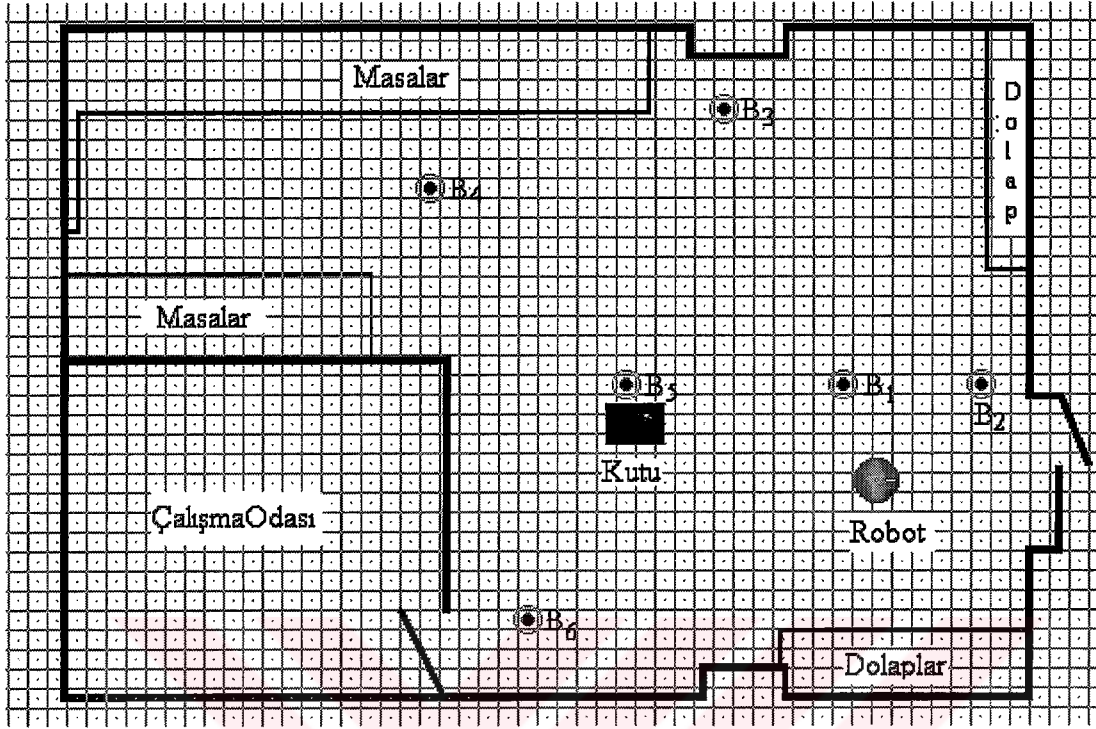
Yapılan çalışmada gerek kullanılan sensörlerin yapısına göre gerekse de sensörlerden kaynaklanan okuma hatalarını en aza indirebilmek amacıyla çevre modellemesi olarak “Gridleme Tabanlı Olasılıksal Modelleme-GTOM” yaklaşımı kullanılmıştır. Bu amaçla, çalışılan laboratuvar ortamı Şekil 5.2’de gösterildiği gibi 20x20 cm.lik koordinat hücreleri ile ızgaralanarak modellenmiştir. Toplam 1998 adet ızgara hücresine sahip ortamda (37 satır, 54 sütun) her bir hücre (i, j) kendi alanı üzerinde bir engelin bulunma olasılığını veren $h(i,j)$ değerine sahiptir. Sıfırla bir arasında değişen bu değer büyük olması hücrenin bir engel tarafından işgal edilme olasılığının büyük olduğunu gösterir.

Robotun laboratuvar ortamında yaptığı planlı hareketlerde, kendi üzerindeki sensörleri vasıtasıyla yaptığı çevresel mesafe okuma bilgileri Bölüm 5.3 ve 5.4’deki yöntemlerle ilgili ızgara hücrelerine yansıtılır. Mobil robot konum ve yön bilgisinin Bölüm 4’de anlatılan yöntemlerle bilindiği varsayılarak yapılan panoramik çevre taramalarında temel olarak ultrasonik ve kızılötesi mesafe bulucu sensörler kullanılmaktadır. Belirli örnekleme anlarında sensörlerden elde edilen mesafe bilgileri robot lokalizasyonu referans alınarak x, y koordinatlarında noktasal bir konum bilgisine dönüştürülmekte ve bu noktaya en yakın ızgara hücresine, ilerleyen bölümlerdeki yöntemler kullanılarak olasılıksal değerler atanmaktadır. Ortamdaki engellerin konumlarına göre olasılıksal değerlerle doldurulan bu ızgara hücreleri vasıtasıyla çevrenin tam bir sayısal haritası elde edilir.

5.3 Bayes Yöntemi ile Çevre Modeli

Bir mobil robotun çalıştığı ortamda verilen başlangıç ve hedef noktaları arasındaki yolunu ve hareketlerini güvenli bir şekilde sağlayabilmesi için sensörlerden gelen bilgileri kullanabilen bir olasılıksal çevre modelinin kurulmasına ihtiyaç vardır. Bayes yönteminin uygulanacağı çevre Bölüm 5.2.2’de anlatıldığı gibi 20x20 cm. karelik hücrelerle ızgaralanmıştır. Her bir hücre kendi alanı üzerinde engellerin bulunma olasılıklarını gösteren tahmini değerlere sahiptir. Hücrelerle ızgaralanmış bir olasılıksal çevrenin oluşturulabilmesi için ultrasonik ve kızılötesi sensörler kullanılmıştır. Robot lokalizasyon bilgisi Bölüm 4’de verilen yöntemlerle elde

edildikten sonra sensör bilgileri ile birleştirilerek robot etrafında yer alan hücrelere olasılıksal değerleri Bölüm 5.3.1, 5.3.2’deki sensör modelleri ve 5.3.3’deki çevre modeli kullanılarak atanır.



Şekil 5.2 Çevre modelinin koordinat hücreleri şeklinde ızgaralanması.

5.3.1 Yöntemin Ultrasonik Sensörlere Uygulanması

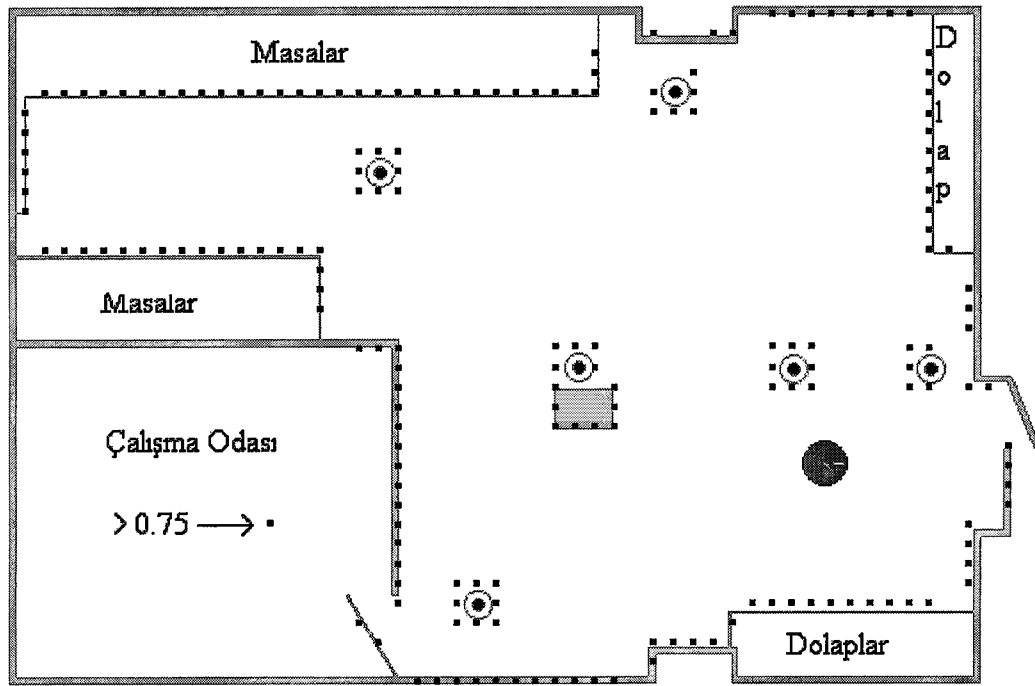
Çevre haritalama amaçlı kullanılan ultrasonik sensörlerin sıfır ortalama ve σ^2 varyanslı Gauss gürültülerinden etkilendiği varsayılarak oluşturulan olasılık yoğunluk fonksiyonu Bölüm 3.3.2’de eşitlik (3.11)’de verilmişti. Diğer bir tarafta, mobil robot etrafındaki çevre, karesel hücrelerden oluşmuş bir ızgaralanma yapısında olup her bir hücre ölçülen mesafe verilerine dayalı kendi durumunu belirten bir olasılıksal değere sahiptir. Böyle bir çevrenin gerçekleştirilebilmesi için sensörlerden elde edilen verilere eşitlik (5.1)’deki Bayes yöntemi [36, 48] uygulanarak bir hücrenin üzerinde engel taşıma olasılığı hesaplanır.

$$P_u[s(h_i) = i\text{şgal} | d] = \frac{P_u[d | s(h_i) = i\text{şgal}] \cdot P_u[s(h_i) = i\text{şgal}]}{\sum P_u[d | s(h_i)] \cdot P_u[s(h_i)]} \quad (5.1)$$

Burada $P_u[d | s(h_i) = i\text{şgal}]$ değeri eşitlik (3.11)’deki ultrasonik sensör modeli kullanılarak bulunabilir, $P_u[s(h_i) = i\text{şgal}]$ ise hücrenin bir önceki değeridir.

Şekil 5.2’de verilen laboratuvar ortamının ultrasonik sensörlerle Bayes yöntemi kullanılarak ortaya çıkarılan olasılıksal modeli Şekil 5.3’de verilmiştir. Şekil üzerinde gösterilen çevre haritası aşağıda verilen üç ayrı aşamada oluşturulmaktadır :

- i) Robota laboratuvar ortamında herhangi bir hedef olmaksızın engellere çarpıtılmadan serbest bir dolaşma yaptırılmıştır. Yaklaşık 0.20m/sn. hız ile yaptırılan bu navigasyon sırasında robot sensörleriyle çevre taraması yaparak Bölüm 4.4’de anlatılan SDSF yöntemiyle 2sn.’de bir konumunu hesaplar.
- ii) Şekilde çalışılan laboratuvarın duvarları ve engeller kalın gri çizgilerle gösterilmiştir. Duvar kenarlarında yer alan ve laboratuvar araç-gereçleri ile bilgisayarları üzerinde barındıran masalar ince gri çizgilerle gösterilmiştir. Bu masaların ultrasonik ve kızılötesi sensörler tarafından rahatlıkla algılanabilmeleri için dış tarafa bakan bacakları yere kadar uzun kağıtlarla kaplanmıştır. Ayrıca ortamda bulunan 60x40x100 cm. ebatlarında kartondan bir kutu ve lokalizasyon amaçlı kullanılan silindirik referans nesneler ($B_1...B_6$) gösterilmiştir. Dikine yerleştirilen referans nesneler ultrasonik sensörlerin algılaması sırasında spekülative yansımalarla sebep olmadığı için bütün yönlerden kolaylıkla tespit edilebilmektedir. Fakat yapılarının ince ve uzun olmasından dolayı uzak mesafelerden yapılan ölçümlerde düz yüzeylere göre daha zayıf bir eko ürettiğinden zaman zaman görülememiştir.
- iii) Çalışılan laboratuvar ortamı hücresel ızgaralara bölündüğünden tespit edilen nesneler Şekil 5.3’de bu ızgaralar üzerinde gösterilmiştir. Her bir hücre 20x20 cm. boyutlu bir karedir. Toplanan ultrasonik sensör verilerinin Bayes yöntemiyle işlenmesi sonucunda üzerinde engel olduğu tespit edilen hücrelere bu engellerin bulunma olasılığı ile orantılı büyüklüğe sahip küçük siyah noktalar atanır. Diğer hücreler ise boş bırakılır. Engelleri temsil eden ve adına piksel denilen küçük siyah noktalar sıfırdan bire kadar değişen olasılıksal değerlere sahiptir ve üç ayrı sınıfta değerlendirilmiştir. 0.10-0.45 arası tek piksel (.), 0.46-0.75 arası üç piksel (-) ve 0.76-1.00 arası ise dokuz piksel (■) ile temsil edilmiştir.



Şekil 5.4 Kızılötesi sensörlerle çıkarılan olasılıksal çevre.

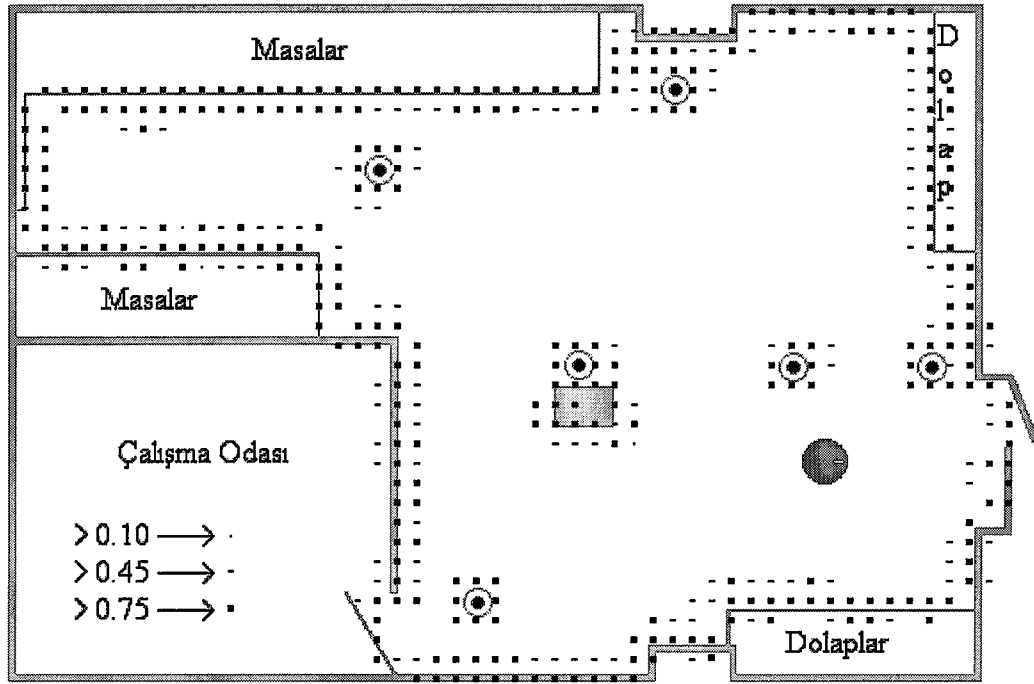
5.3.3 Dempster-Shafer Kuralıyla İki Sayısal Haritanın Birleştirilmesi

Şekil 5.3 ve 5.4'deki her bir çevre haritası hücrelerin $P_u[s(h_i) = işgal]$ ve $P_k[s(h_i) = işgal]$ işgal edilme olasılıklarıyla ifade edilmiştir. Elde edilen çevre haritalarını birleştirmek amacıyla eşitlik (5.2)'de verilen Dempster-Shafer kuralı [48], [49] uygulanarak hücre değerlerine atanan olasılıksal değerlerin güvenilirliği artırılmıştır.

$$P[s(h_i) = işgal] = \frac{P_u[s(h_i) = işgal] \cdot P_k[s(h_i) = işgal]}{1 - \lambda_s} \quad (5.2)$$

Burada, $\lambda_s = P_u(1 - P_k) + P_k(1 - P_u)$, P_u ve P_k değerleri ise sırası ile ultrasonik haritalamada ve kızılötesi haritalamada elde edilen olasılık değerleridir.

Şekil 5.3 ve 5.4'deki olasılıksal çevre haritalarının eşitlik (5.2)'deki Dempster-Shafer kuralıyla birleştirilmesiyle Şekil 5.5'deki birleştirilmiş olasılıksal çevre elde edilir.



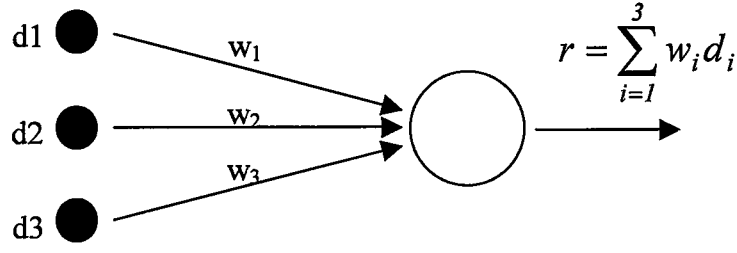
Şekil 5.5 Birleştirilmiş olasılıksal çevre.

5.4 Ardışıl Temel Bileşenler-ATB Yöntemiyle Veri Birleştirme

Çalışmanın bu kısmında geliştirilen Ardışıl Temel Bileşenler-ATB yönteminde ultrasonik ve kızılötesi sensörlerden kaynaklanan okuma hatalarını en aza indiren Şekil 5.6'daki basit bir lineer çıkış ünitesi göz önüne alınmıştır. Yöntemde ardışıl olarak üç sensör okuması temel bileşenler şeklinde alınarak bir “ r ” skaler değeri oluşturulmaktadır. Bir veri birleştirme şeklinde ortaya konan yöntem ile elde edilen “ r ” değeri ardışıl okuma yapan üç sensörün ortasında yer alan sensöre gerçek okuma şeklinde atanır. Şekil 5.6'de yer alan basit lineer birimin çıkışı “ r ” eşitlik (5.3)'de verilmiştir.

$$r = \sum_{i=1}^3 w_i . d_i = \bar{w}^T . d = d^T . \bar{w} \quad (5.3)$$

Burada, “ w ” ağırlık vektörünü, “ d ” ardışıl sensör okumalarını ve “ d_i ” ise bu okumaların i ' ninci bileşenini vermektedir.



Şekil 5.6 Basit bir lineer yapı.

Hebb' sin kuralına göre [50], ağırlık vektörleri eşitlik (5.4)'de kullanılmak üzere eşitlik (5.5)'de verildiği şekilde güncellenebilir.

$$w_i(t+1) = w_i(t) + \Delta w_i \quad (5.4)$$

$$\Delta w_i = \xi \cdot r \cdot d_i \quad (5.5)$$

burada, “ d_i ” sensör okumalarının i ' ninci bileşeni, “ Δw_i ” i ' ninci okumaya ait ağırlıktaki değişim, ξ ise öğrenme faktörü ismi verilen bir sabittir. Denge noktasında eşitlik (5.6)'da verildiği gibi, ağırlıklardaki değişim ortalamaları sıfır olmalıdır.

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \Delta \vec{w}_1 \\ \Delta \vec{w}_2 \\ \Delta \vec{w}_3 \end{bmatrix} = \begin{bmatrix} r \cdot d_1 \\ r \cdot d_2 \\ r \cdot d_3 \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^3 w_j \cdot d_j \cdot d_1 \\ \sum_{j=1}^3 w_j \cdot d_j \cdot d_2 \\ \sum_{j=1}^3 w_j \cdot d_j \cdot d_3 \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^3 R_{1j} \cdot w_j \\ \sum_{j=1}^3 R_{2j} \cdot w_j \\ \sum_{j=1}^3 R_{3j} \cdot w_j \end{bmatrix} = [R] \vec{w} \quad (5.6)$$

$$[R] \vec{w} = \lambda \cdot \vec{w} \quad (5.7)$$

Burada, $[R] = \vec{d} \cdot \vec{d}^T$, λ : ise $[R]$ ' nin bütün özdeğerleridir. $[R]$ detaylı şekliyle

$$[R] = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} = \begin{bmatrix} d_1 d_1 & d_1 d_2 & d_1 d_3 \\ d_2 d_1 & d_2 d_2 & d_2 d_3 \\ d_3 d_1 & d_3 d_2 & d_3 d_3 \end{bmatrix} \text{ biçimindedir.}$$

İlinti (korelasyon) matrisi ismini verdiğimiz $[R]$ matrisinin özellikleri :

- $[R]$ simetrik yapıdadır. Bu nedenle $[R]'$ nin bütün özdeğerleri (λ) gerçektir sayılardır ve özvektörleri ortogonal olarak ortaya çıkar.
- $[R]$ yarı kesin pozitif matrisdir. Bu nedenle bütün özdeğerleri pozitif veya sıfırdır.

Eşitlik (5.6) bize eşitlik (5.7) ye göre $\vec{w}$ vektörünün $[R]'$ nin sıfır özdeğerli ($\lambda=0$) bir özvektörü olduğunu ifade eder. Fakat bu değer denge noktasını oluşturmaz. Çünkü $[R]$ en azından bir veya birkaç pozitif değere sahip olacaktır ve pozitif özdeğerli bir özvektör boyunca bir bileşene sahip herhangi bir değişim zamanla üstel (exponential) olarak büyüyecektir. Bu nedenle büyümenin maksimum özdeğere (λ_{maks}) karşılık gelen bir yön doğrultusunda çok fazla olacağı aşıkardır. Böylece, maksimum temel bileşen bizim yöntemimize bir çözüm oluşturur.

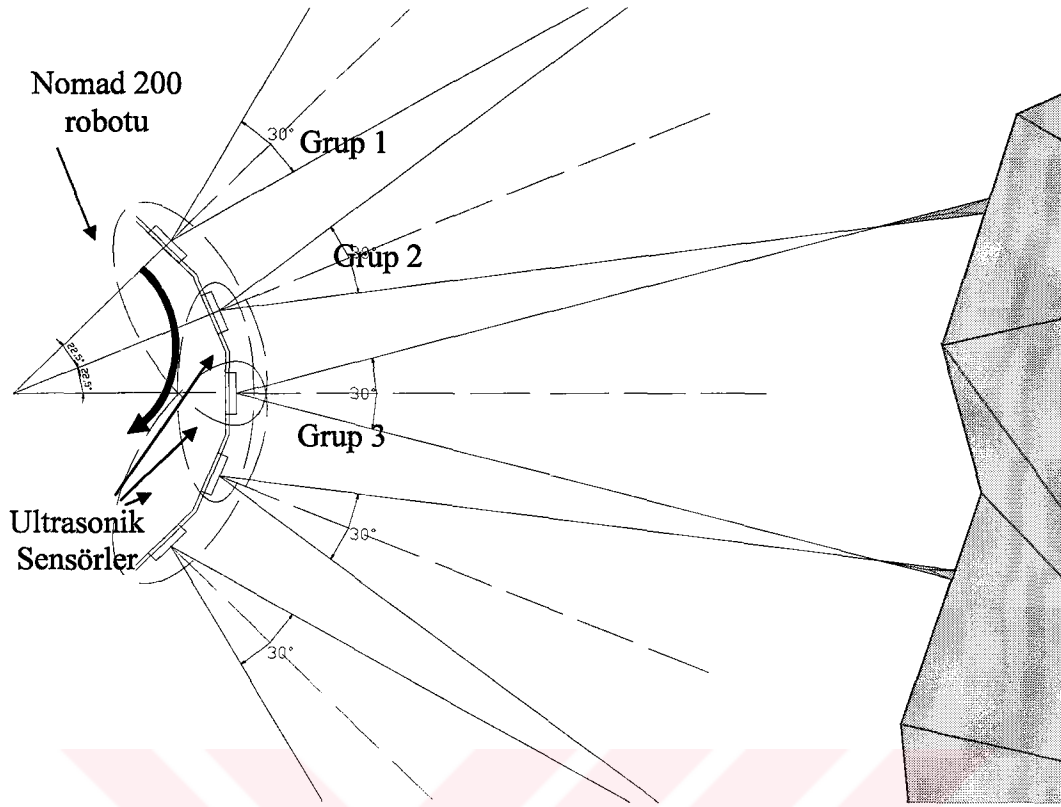
5.4.1 Yöntemin Ultrasonik Sensörlerle Kullanımı

Nomad200 mobil robotu üzerinde Sensus 200 [51] adı verilen bir halka vardır. Bu halka kenarlarında 22.5° derecelerle konumlandırılmış 16 adet Polaroid 6500 [46] sonar mesafe okuyucu modülü mevcuttur. Modül mobil robotlarda yaygın kullanımı olan ve 12 cm.'den 647 cm. ye kadar olan mesafeleri ölçebilen bu akustik mesafe okuyucular üzerinde yapılan analizler Bölüm 3.3'de detaylı bir şekilde verilmiştir.

Ardışıl temel bileşenler yöntemi ultrasonik sensörlere Şekil 5.7'de verildiği gibi üçerli gruplar halinde uygulanır. Bu uygulama sonucunda her grup için $[R]$ ilinti matrisi ve matrisin maksimum özdeğeri olan λ_{maks} hesaplanır. Yine eşitlik (5.7) ve (5.8)'de $[R]$ matrisi ve λ_{maks} değerleri kullanılarak uygun ağırlık vektörü $\vec{w}$ bulunur.

$$w_1 + w_2 + w_3 = 1 \quad (5.8)$$

Eşitlik (5.3)'de verildiği gibi hesaplanan $\vec{w}$ ağırlık vektörü üçerli gruplarda yer alan ilgili sensör okumaları ile etkileşime girerek “ r ” değerini oluşturur. Bu değer grubun ortasında yer alan sensörün okuma hataları en aza indirilmiş gerçek okuma değeri olarak ele alınır.

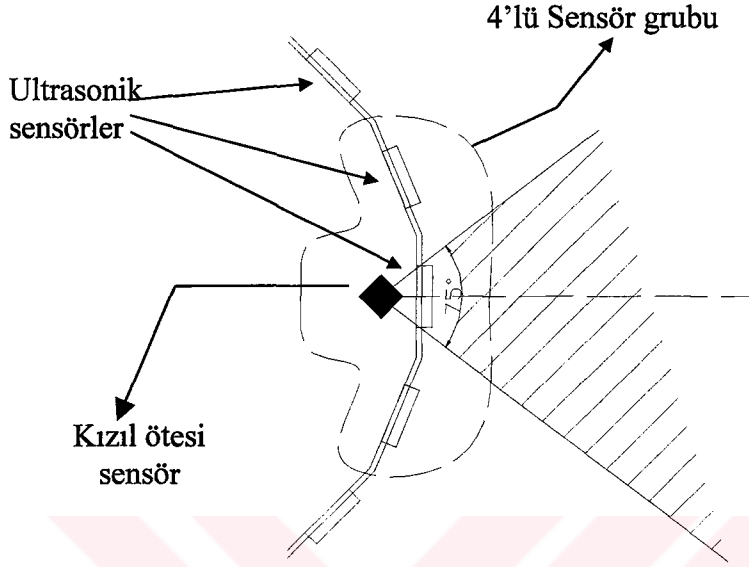


Şekil 5.7 3'lü ardışıl ultrasonik sensör grupları.

5.4.2 Yöntemin Ultrasonik ve Kızılötesi Sensörlerle Kullanımı

Nomad200 mobil robotu üzerinde mesafe okuyucu cinsinden ultrasonik sensörlerin yanı sıra kızılötesi sensörlerde konumlandırılmıştır. Sensus 300 [51] ismi verilen halka üzerinde 22.5 derecelik açılarla dizilen kızılötesi sensörler etraflarındaki nesnelerden yansıyan kızılötesi ışık şiddetlerini ölçerek nesnelerin kendilerine olan uzaklıklarını tayin ederler. Bölüm 3.4'deki sensör modelinde de verildiği gibi 2 cm. ile 38 cm. arasındaki mesafeleri $\pm 30^\circ$ derecelik bir konik açılımda gayet hassas bir şekilde ölçerler. Bu konik açılım bir ultrasonik sensörün gördüğü konik açılamdan daha fazladır. Bu nedenle, Şekil 5.8'de gösterildiği gibi iki farklı sensörün aynı anda kullanımı durumunda Ardışıl Temel Bileşenler yöntemi üçlü ultrasonik sensör gruplarına bir kızılötesi sensör okuması ilave edilerek gerçekleştirilir. İlave kızılötesi sensör ultrasonik sensör gruplarının ortasına karşılık gelen konumlardadır. Ultrasonik sensörlerin etkili okuma mesafeleri 12 cm. ile 647 cm. arasında değiştiğinden güvenli okuma açısında ilgili yöntem her iki sensöre emniyet payı da dikkate alınarak ancak

15 cm. yi geçen okuma mesafeleri için uygulanır. Bu mesafenin altında kalan okumalar için sadece kızılötesi sensör okumaları temel alınır.



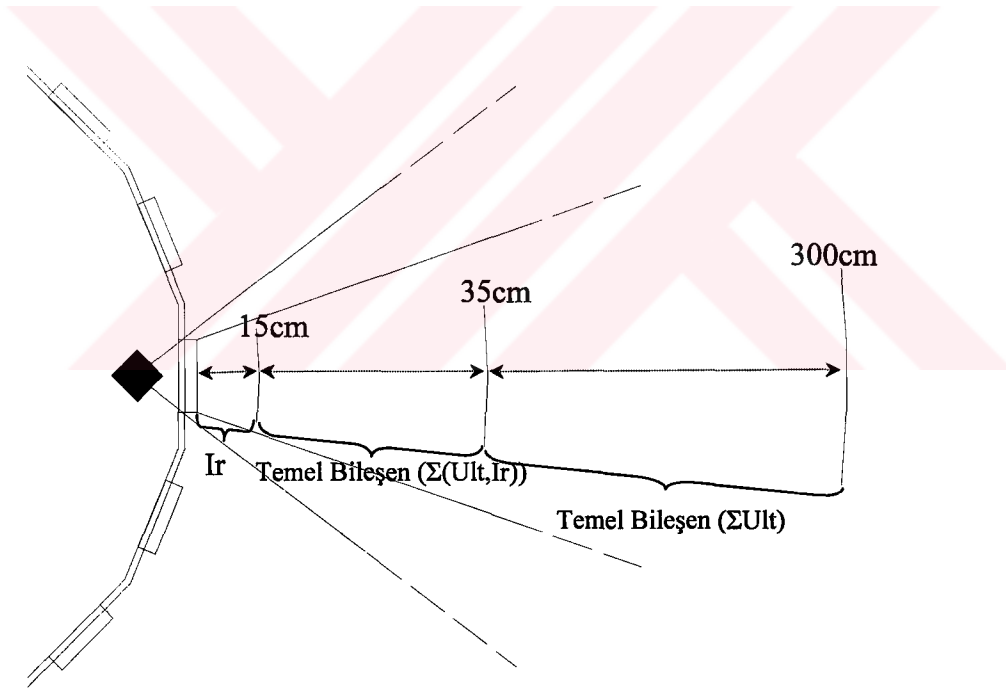
Şekil 5.8 Ultrasonik ve Kızılötesi sensör grubu.

5.4.3 Yöntem İçin Karar Verme Algoritması

Ultrasonik sensörlerle kızılötesi sensörlerin etkin olarak hangi mesafelerde tek başına veya beraber kullanılacağını bilmesi önemlidir. Bu amaçla Şekil 5.9'da gösterildiği gibi okunan mesafelere göre bir karar verme algoritması sisteme ilave edilmiştir. Algoritmaya göre 12 cm. den yakın nesneleri göremeyen ultrasonik sensörlerin yaratacağı zafiyet 2 cm. ile 15 cm. arasındaki uzaklıklarda yer alan nesnelerin sadece kızılötesi sensörlerle algılanmasıyla giderilmiştir. Kızılötesi sensörlerin okuyabileceği en uzak mesafe 38 cm. olduğundan emniyet payıda göz önüne alınarak 15 cm. ile 35 cm. arasında yer alan nesnelerin okunmasında ultrasonik sensörlerle beraber kullanılırlar. İşlenmemiş sensör verileri Bölüm 5.4.2 anlatıldığı gibi ATB yöntemiyle birleştirilir.

Kızılötesi sensörlerin zafiyete düştüğü 35 cm. den uzak nesnelerin algılanmasında ise sadece ultrasonik sensörler kullanılır. Mobil robot, etrafındaki nesneleri hareket halindeyken yakalamakta ve kendisine 300 cm. den uzak olan nesneleri görülmemiş olarak ele almaktadır. Emniyet payları da göz önüne alınarak 35 cm. ile 300 cm. arasında yer alan nesneleri görebilen ultrasonik sensörlerden elde edilen veriler yine ATB yöntemi kullanılarak Bölüm 5.4.1'de anlatıldığı gibi birleştirilir.

Karar verme algoritmasının devreye girmesinden sonra ATB yöntemiyle bulunan yeni, gerçeğe yakın " d_{ger} " ölçüm mesafesi sadece "kızılötesi" sensörlerle elde edildiyse sensörün eşitlik (3.12)'de verilen modeli kullanılarak ilgili ızgara hücrelerine engel temsil edilme (işgal) olasılık değerleri atanır. Şayet bu mesafe değeri "ultrasonik" veya "ultrasonik+kızılötesi" sensörlerle elde edildiyse ortamdaki ilgili ızgara hücrelerinin değerlerini tayin eden olasılıksal yoğunluk fonksiyonu olarak eşitlik (3.11) kullanılır.



(Ult: Ultrasonik , Ir: Kızılötesi)

Şekil 5.9 Karar verme algoritmasında sensör seçimi.

5.4.4 Yöntem Üzerine Yapılan Deneysel Çalışmalar

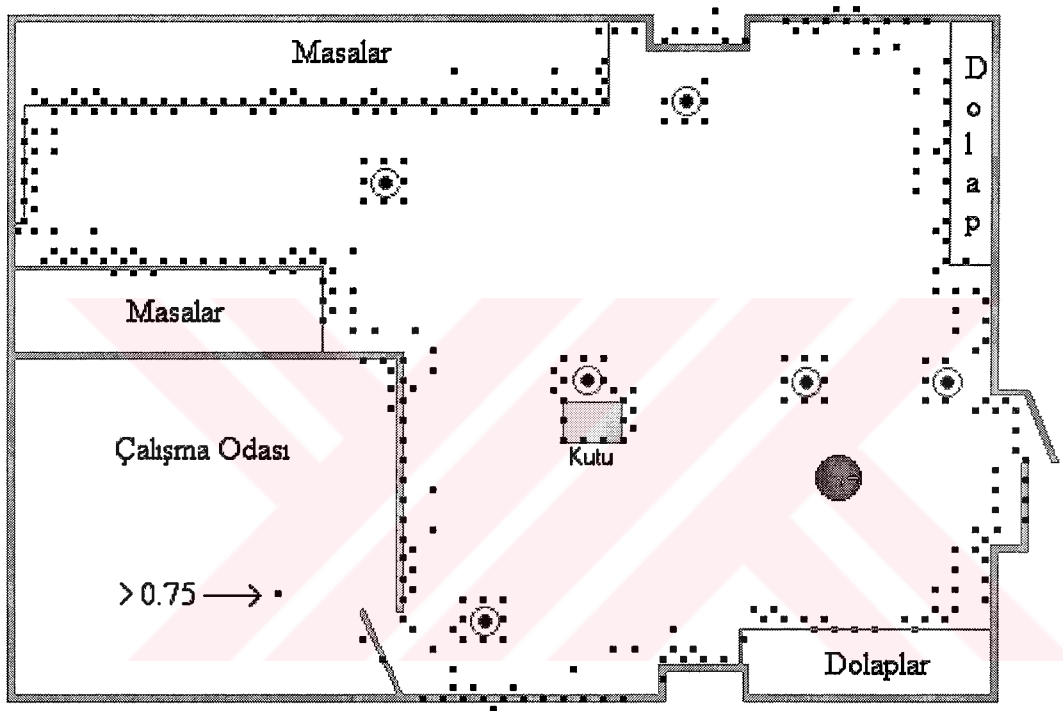
Çevre haritasını oluşturan algoritma Nomad200 mobil robotu üzerinde test edilmiştir. Robotun engellerin seyredildiği yerlerde yapabildiği azami hız $v_{maks}=0.26$

metre/saniye, tekerleklerin azami dönüş oranı ise $w=30^0$ /saniye olarak seçilmiştir. Mobil robot içinde araç hareketlerini düzenleme ve sensör verilerini toplama amaçlı bir Pentium-133 bilgisayar konumlandırılmıştır. Ayrıca mobil robot dışında çevre haritalama ve engel sakınma algoritmalarının çalıştırıldığı başka bir Pentium-133 bilgisayar kullanılmaktadır. Bu bilgisayar robot ile üzerindeki bir radyo-ethernet arabirimi aracılığı ile haberleşerek robot üzerindeki gerekli gördüğü sensörlerden veri toplayabilmekte ve bu verilere dayalı çevre haritası oluşturarak robota yapacağı hareket değişimlerini iletebilmektedir.

Şekil 5.10'da laboratuvarda yapılan deneysel bir çalışmanın sonuçları gösterilmiştir. Aynı şekil üzerinde gösterilen çevre haritası aşağıda verilen üç ayrı aşamada oluşturulmaktadır :

- i) Robota laboratuvar ortamında herhangi bir hedef olmaksızın engellere çarpıtılmadan serbest bir dolaşma yaptırılmıştır. Yaklaşık 0.20m/sn. hız ile yaptırılan bu navigasyon sırasında robot sensörleriyle çevre taraması yaparak Bölüm 4.4'de anlatılan SDSF yöntemiyle 2 sn.'de bir konumunu hesaplar.
- ii) Şekilde çalışılan laboratuvarın duvarları ve engeller kalın gri çizgilerle gösterilmiştir. Duvar kenarlarında yer alan ve laboratuvar araç-gereçleri ile bilgisayarları üzerinde barındıran masalar ince çizgilerle gösterilmiştir. Bu masaların ultrasonik ve kızılötesi sensörler tarafından rahatlıkla algılanabilmeleri için dış tarafa bakan bacakları yere kadar uzun kağıtlarla kaplanmıştır. Ayrıca ortamda bulunan 60x40x100 cm. ebatlarında kartondan bir kutu ve lokalizasyon amaçlı kullanılan silindirik referans nesneler ($B_1...B_6$) gösterilmiştir. Dikine yerleştirilen referans nesneler ultrasonik sensörlerin algılaması sırasında spekülâtif yansımalarla sebep olmadığı için bütün yönlerden kolaylıkla tespit edilebilmektedir. Fakat yapılarının ince ve uzun olmasından dolayı uzak mesafelerden yapılan ölçümlerde düz yüzeylere göre daha zayıf bir eko ürettiğinden zaman zaman görülememiştir.
- iii) Çalışılan laboratuvar ortamı hücresel ızgaralara bölündüğünden tespit edilen nesneler Şekil 5.10'da bu ızgaralar üzerinde gösterilmiştir. Her bir hücre

düştüğü gözlenmiştir. Çalışılan laboratuvar ortamında bazı bölgelerde robot düşük hızda hareket ederken mesafe sensörleri gereğinden fazla bilgi toplar. Bu istenmeyen fazlalık hücresel ızgaralar üzerindeki engellerin olasılıksal temsiline doğal olarak azami bir ile ($P_{maks}=1.00$) sınırlanmış olmasından dolayı dikkate alınmaz. Diğer bir taraftan mobil robot ortalama hızlarda veya erişebildiği azami hızında hareket ederken birkaç okuma ile tespit ettiği engelleri bulundukları hücre ızgaralarında olabildiğince yüksek olasılık değerlerinde temsil edebilmektedir.



Şekil 5.11 $P > P_{eşik} = 0.75$ olasılıksal değerleri ile oluşturulan çevre haritası.

5.5 ATB Yönteminin Bayes Yöntemiyle Karşılaştırılması

Izgara hücrelerine bölünmüş bir çevre üzerinde haritalama amaçlı kullanılan yöntemlerden Bölüm 5.3’de verilen Bayes yöntemi en tanınmışıdır. Bu yöntemin denenmesi için ızgaralanan bir laboratuvar ortamında mobil robot gezdirilerek bir çok farklı konumdan ultrasonik ve kızılötesi sensörlerle çevre taraması yaptırılmıştır. Bu sensör okuma değerleri ilgili ızgara hücrelerinde eşitlik (3.11) ve (3.12) kullanılarak ayrı ayrı olasılıksal değerlere dönüştürülmekte ve bu değerlerde eşitlik (5.1)’de verilen Bayes yöntemiyle geçmiş değerleri ile işlenmektedir. Daha sonra, ilgili ızgara hücrelerinde her bir sensör için bulunan işgal olasılık değerleri eşitlik

(5.2)'de verilen Dempster-Shafer [48, 49] kuralıyla birleştirilerek tek bir değere indirgenmekteydi. Bu durum, zaman zaman baştan hatalı gelen sensör okumalarını ilgili ızgara hücrelerine aktararak çevre haritalama işleminin sağlıklı yapılmasını engellemektedir. Ayrıca bu yöntemde her bir sensör okumasına karşılık gelen geniş bir yelpazedeki olasılıksal ızgara hücre değerlerinin Bayes yönteminden sonra tekrar Dempster-Shafer veya benzeri yöntemlerle birleştirilmeye ihtiyaç vardır. Bu durum algoritmanın harcadığı zamanı oldukça arttırmaktadır. Örneğin, laboratuvarında yer alan örnek bir ortamda, lokalizasyon işleminden sonra Bayes yöntemiyle yapılan bir periyotluk çevre modellemesi için algoritmaya gereken hesaplama zamanı ortalama 850msn. dir. Halbuki ATB yöntemiyle yapılan çevre haritalama işleminde yanlış sensör okumalarına daha başında müdahale edilerek, ilgili ızgara hücre değerlerinin doğru bir şekilde kıymetlendirilmesi sağlanmaktadır. Olasılıksal hücre değerleri sadece birleştirilmiş sensör okumalarını kullanan eşitlik (3.11) ve (3.12) ile oluşturulduğundan ayrıca bir birleştirme yöntemine ihtiyaç duyulmaz. Bu durum bilgisayarın çevre haritalama işlemi için harcadığı zamanı oldukça azaltmaktadır. Örneğin, aynı robot konumunda yapılan bir çevre taramasında Bayes yöntemini kullanan algoritma için gereken hesaplama zamanı 850 ms. iken ATB yöntemini kullanan algoritma ise sadece 500 msn. ye ihtiyaç duyabilmektedir.

Bayes yöntemiyle oluşturulan bir çevrede yer alan engeller göründüklerinden daha büyük bir şekilde çevre haritasında yer almaktadır. Bu durumda, bir sonraki aşamada yer alacak olan yol-planlama safhasında problemler yaşanmaktadır. Örneğin robotun arasından rahatlıkla geçebileceği iki nesnenin çevre haritasındaki görünümü robot yolunun bu aradan geçmesini engelleyerek daha uzun yolların planlanmasına sebep olabilmektedir. Halbuki ATB yöntemiyle çevredeki nesneler gerçeğe yakın konumlarında algılanarak hedefe vardırıcı en kısa yolun planlanması sağlanabilmektedir.

5.6 Sonuç

Bu bölümde yapılan çalışmalarda mobil robotun konumunu bildiği varsayılarak içinde bulunduğu çevrenin haritasını çıkartabilmesi için kullanılan Bayes yöntemi ile bu amaçla geliştirilen ATB yöntemi tanıtılmıştır. İzgarasal hücrelere ayrılmış çevrelerde haritalama amaçlı kullanılan yöntemlerin en yaygını olan Bayes yöntemi

laboratuvar ortamında bir mobil robot üzerinde denenerek olasılıksal yaklaşımlarla çevrenin sayısal bir haritası çıkartılmıştır. Bir sonraki kısımda anlatılan geliştirilmiş ATB yöntemiyle bir karşılaştırma yapabilme amacı taşıyan bu yöntemin sonuçları şekillerle anlatılmıştır.

Çevre haritalama amaçlı geliştirilen Ardışıl Temel Bileşenler-ATB yöntemiyle mobil robotlarda çoklu sensör veri füzyonu ve çevre modeli tanıtılarak test edilmiştir. Bu yöntemde, kendi yapılarından kaynaklanan güvenilir olmayan ultrasonik ve kızılötesi verileri olasılıksal olarak iki boyutlu hücresel ızgaralar üzerinde modellenmiştir. Ortamdaki engellerin olasılıksal temsili mobil robot hareket halindeyken yapılan hızlı ve sürekli sensör okumaları ile elde edilmiştir. ATB yöntemiyle farklı sensörlerden elde edilen mesafe bilgileri birleştirilerek hızlı bir şekilde harita üzerinde temsil edilebilmekte ve kullanılan herhangi bir engel-sakınma ve yol planlama algoritmasına (yapay potansiyel alanları vb.) gereken verileri zaman geciktirmeden sağlayabilmektedir.

Bu bölümde verilen çevre haritalama yöntemleriyle 4 ncü bölümde verilen lokalizasyon yöntemleri bir sonraki 6ncı bölümde birleştirilerek, bir mobil robotun engellere çarpmadan hedefine olan yolunu ve hareketlerini planlayabilmesi amacıyla kullanılmıştır.

6. KONUMLANDIRMA VE ÇEVRE HARİTALAMA YÖNTEMLERİNİN NAVİGASYON AMAÇLI BİRLEŞTİRİLMESİ

6.1 Giriş

Mobil robot alanında yapılan çalışmaların en önemli amaçlarından biride, robota kapalı düzenli ortamlarda bir çok görevi yapabilme kabiliyetini kazandırabilmektir. Bunu yapabilmesi için robot bulunduğu çevreden ve kendi konumundan haberdar olmalıdır. Robotun bunu başarabilmesi için, çevre şartlarına müdahale etmeyen ve beklenmeyen engellerden etkilenmeyen bir lokalizasyon yöntemini bulunduğu ortamın sayısal haritasını çıkaran ve sensör verileri ile onu sürekli güncelleyen bir çevre haritalama algoritması ile birleştirilmesi gereklidir. Hareket esnasında yapılan sürekli örneklemelerle doğru ölçemeyen ultrasonik sensörler gibi gürültülü sensörlerin modellemesine uygun bir sayısal çevre temsili ortaya çıkarılabilir. Daha önce yapılan bir çok araştırmada lokalizasyon ve çevre haritalama problemlerinin çözümüne yönelik sonuçlar ayrı ayrı açıklanmıştır. Fakat her iki problemin beraber çözümüne yönelik çalışmalara daha az ağırlık verilmiştir.

Çevre haritalama ve lokalizasyon problemleri ile aynı anda uğraşan ortamdaki referans işaretlere dayalı bir yöntem Thrun ve arkadaşları [52] tarafından önerilmiştir. Yöntem, belirli zaman aralıklarında örnekleme yaparak robot konumunu güçlü bir şekilde tahmin edebilen bir lokalizasyon algoritmasını kullanır. Uygulamalar gerçek zamanda olmayıp referans işaretlerin önceden belirlenmesini gerektirir. Castellanos ve arkadaşları [53] tarafından sunulan diğer bir öneride ise CCD kamera ve 2-D laser mesafe bulucular gibi harici sensörler kullanılır. Yönteme göre, robot önceden belirlenen bir yolu izlemekte ve bilgi toplama amacıyla düzenli aralıklarda durmaktadır. Daha sonra, sürekli yapılan bir lokalizasyon sonucu kullanılan bir sayısal çevre haritalama algoritması Yamauchi ve arkadaşları [54] tarafından geliştirildi. Yöntemde sonarlar bir laser mesafe sensörü gibi güvenilir bir

şekilde kullanılır. Haritalama işlemi ise ortamdaki duvar ve nesnelerin kenarlarını keşfe dayalı bir yöntem ile yapılmaktadır.

Araştırmacılar lokalizasyon problemlerinden kaçınmak amacıyla robot konumundan bağımsız olarak bir çevre temsilinin yöntemlerini bulmak üzerine de yönlendiler. Bu araştırmalar neticesinde daha önce Bölüm 5.1’de bahsedilen öznitelik bulmaya yönelik çevre modelleri ortaya çıkarıldı. Araujo ve Grupen [32] tarafından çevrenin spekülative yansıma modeli üzerine çalışıldı. Onlara göre kapalı ortamlar birçok spekülative yüzey içerirler ve dolayısıyla bu yüzeylerin sonar dalgasıyla etkileşimi çevrede bulunan nesnelerin geometrik yapıları hakkında detaylı bilgi vermekteydi. Chong ve Kleeman [55] ise Julier-Uhlmann Kalman filtresi ile düzlemsel ve köşeli yapıların temsili üzerine olan yöntemlerini tanıttılar. Onların yöntemleri gruplandırılmayan öznitelikler, kenar tespitleri ve robot konumunun büyük belirsizlikler taşıyan ortamlarda bulunabilmesi üzerineydi.

Bahsedilen bütün bu algoritmalar sensör modeli ve konfigürasyonu üzerine olup seçilen özniteliklerin nasıl tespit edilebileceğine dayalıdır. Spekülative yansımalarla sahip ultrasonik sensörler gibi güvenilir sensör sistemlerinin kullanıldığı durumlarda, bu modelleme yöntemleri ayrıca problemlere sahiptir. Bütün bunların ötesinde, bu yöntemler yüksek oranda robota verilen görevlere bağımlıdır. Bu nedenle yöntemlerin kullanıldığı parametrelerin çalışılan ortamın kendine özgü koşullarına göre tekrar ayarlanması gereklidir.

Tezin bu aşamasında Bölüm 4.4 ve 5.4’de anlatılan lokalizasyon ve çevre haritalama yöntemleri güvenli bir robot navigasyonu sağlayabilecek biçimde Bölüm 6.2’de anlatıldığı şekilde birleştirilmiştir. Bölüm 6.3’de ise, içindeki nesnelerin konumlarının yarı bilindiği veya bilinmediği kapalı ortamlarda, geliştirilen lokalizasyon ve çevre haritalama yöntemlerinin engellerden sakınma ve yol planlama biçiminde navigasyon amaçlı kullanımı anlatılmıştır. Yarı bilinen ortamlarda yol planlaması “Yapay Potansiyel Alanı-YPA” yöntemiyle yapılırken, mobil robotu verilen hedefe vardırıcı en kısa güvenli yol planlanabilmiştir.

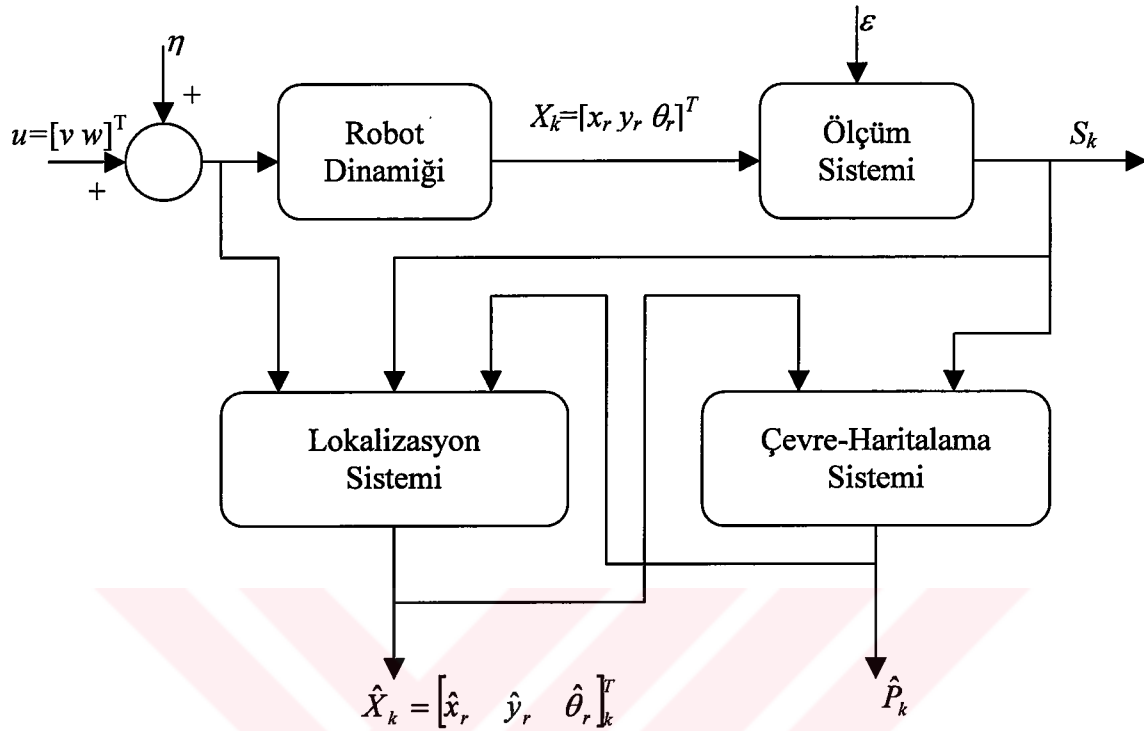
6.2 Lokalizasyon ve Çevre Haritalama Bilgilerinin Birleştirilmesi

Güvenli ve verimli bir robot navigasyonu için lokalizasyon ve çevre haritalama yöntemleri vasıtasıyla elde edilen bilgiler birbirini destekleyici yönde birleştirilmelidir. Bu amaçla yapılan çalışmada t_k örnekleme zamanlarında S_k sensör ölçümleri toplanarak Şekil 6.1’de gösterildiği gibi lokalizasyon ve çevre haritalama algoritmalarına gönderilir. Robot tekerleklerine gönderilen $u=[v \ w]^T$ ilerletme ve döndürme komutları ile verilen görevi yapmak üzere hareket ettirilen robot t_k anında eşitlik (3.1)’e göre $X_k = [x_r \ y_r \ \theta_r]^T$ konumundadır. Bu aşamada yapılan odyometrik, ultrasonik, kızılötesi, manyetik pusula ve fiber-optik jiroskop ölçüm okumaları (S_k) 4. ve 5. bölümlerde anlatılan lokalizasyon ve çevre haritalama algoritmalarına gönderilir. Robota verilen “ u ” tekerlek komutlarına ve çevresel harita bilgisine sahip lokalizasyon algoritması üç farklı bilgiyi birleştirerek olası robot konumunu $\hat{X}_k = [\hat{x}_r \ \hat{y}_r \ \hat{\theta}_r]^T$ tahmin etmeye çalışır. Lokalizasyon algoritmasından elde edilen tahmini robot konumu ile t_k anında yapılan ölçüm okumalarını birleştiren çevre haritalama algoritması ise $\hat{P}_k$ şeklinde ifade edilen robot etrafındaki hücresel grid değerlerini günceller. Lokal haritalama ismi de verilen robot etrafındaki hücreler Bölüm 6.3’de anlatılan engel sakınma ve yol planlama algoritmalarında kullanıldıktan sonra global haritadaki yerlerine atanırlar. Şekil 6.1’de sisteme eklenen gürültüler; robot kinematikinden kaynaklananlar için η , sensör okumalarına eklenen gürültüler için ε sembolleri ile sınıflandırılmaktadır.

6.3 Navigasyon Amaçlı Engellerden Sakınma ve Yol Planlama

Bir mobil robot navigasyonunda beklenen, verilen hedef noktasına kaybolmadan ve herhangi bir şeye çarpmadan varabilmesidir. Bir navigasyon işlemi üç ana görevi içerir, bunlar; haritalama, planlama ve hareket etmektir. İşlemin haritalama kısmı ve onun destekleyici unsuru olan lokalizasyon 4. ve 5. bölümlerde detaylı anlatılmaktadır. Engellere çarpmadan, verilen hedefe vardırıcı en kısa yol ve zaman gibi özelliklere sahip bir çok yol planlama algoritması geliştirilmiştir. Bu algoritmalarda başlıca “Yapay Sinir Ağları-YSA” [56], “Yapay Potansiyel Alanlar-YPA” [57-59], “Bulanık Mantık-BM” [60], “Voronoi Diyagramları-VD” [61] kullanılmaktadır. Tez’e konu teşkil eden lokalizasyon ve çevre haritalama

alanlarında geliştirilen algoritmaların bir bütünlük içinde navigasyon işleminde kullanımını sağlamak amacıyla yarı bilinen kapalı ortamlarda yol planlama işlemi bir sonraki bölümde YPA ile gerçekleştirilmektedir.



Şekil 6.1 Birleştirilmiş lokalizasyon ve çevre haritalama sisteminin blok diyagramı.

6.3.1 Yapay Potansiyel Alanı-YPA Yöntemiyle Yol Planlama

Bir mobil robotu bulunduğu ortamdaki hareketli veya hareketsiz engellerden koruyabilmek ve daha da önemlisi çevrenin mobil robot çarpmalarından zarar görmesini önleyebilmek çok önemlidir. Bu nedenle, mobil robotların verilen hedefler doğrultusunda görevlerini yerine getirirken meydana gelebilecek olası çarpışmalardan uzak yolunu planlayabilmesi için çeşitli algoritmalar geliştirilmiştir. Bunlardan biride YPA algoritmasıdır [57-59].

Yöntemin uygulamasında, Şekil 6.2’de gösterildiği şekilde robotla beraber yer değiştiren ve 11x11 hücrelik bir alanı kaplayan kare şeklinde aktif bir bölge (A*) vardır. Aktif A* bölgesinin merkezi robotun bulunduğu koordinatlardır. Bu bölgeyi meydana getiren 11x11’lik kare alan, kartezyen koordinat sisteminde aynı koordinatları paylaştıkları grid hücrelerinin sayısal değerlerini taşırlar ve aktif

hücreler a_{ij}^* adını alırlar. Bir başka deyişle robotla hareket eden aktif bölge 11x11 adet aktif hücreden meydana gelir ve üzerine denk düştükleri kartezyen koordinat hücrelerinin değerleriyle anılırlar. Bir sonraki paragrafta anlatılacağı gibi bu aktif hücrelerin robot üzerinde doğrudan yapay bir potansiyel alan etkisi vardır.

Algoritma çeşitli duyaçlardan elde edilen mesafe bilgilerinin 5. bölümde verilen çevre haritalama yöntemleriyle işlenmesi sonucunda sayısal değerler kazanan hareketli aktif bölge hücrelerinin mobil robota itici bir kuvvet, varılacak hedefin ise çekici bir kuvvet uyguladığını varsayarak bir bileşke sonuç vektörünün hesaplanması ve robotun bu vektör doğrultusunda yönlendirilmesi üzerinedir.

Şekil 6.3’de gösterildiği gibi aktif hücreler robota sanal bir itici kuvvet ($F_{i,j}$) uygular. Bu kuvvetin miktarı her bir a_{ij}^* aktif hücresinin sayısal değeriyle doğru orantılı, robota olan uzaklıkları ile de ters orantılıdır. Her bir a_{ij}^* aktif hücresinin robota uyguladığı itici kuvvet ve bu kuvvetlerin robot üzerinde oluşturduğu toplam kuvvet (6.1) ve (6.2)’deki eşitlikler kullanılarak hesaplanabilir.

$$\vec{F}_{i,j} = \frac{F_{cr} \cdot a_{i,j}^*}{d^2(i,j)} \left((x_i - x_r) \cdot \vec{x} + j(y_j - y_r) \cdot \vec{y} \right) \quad (6.1)$$

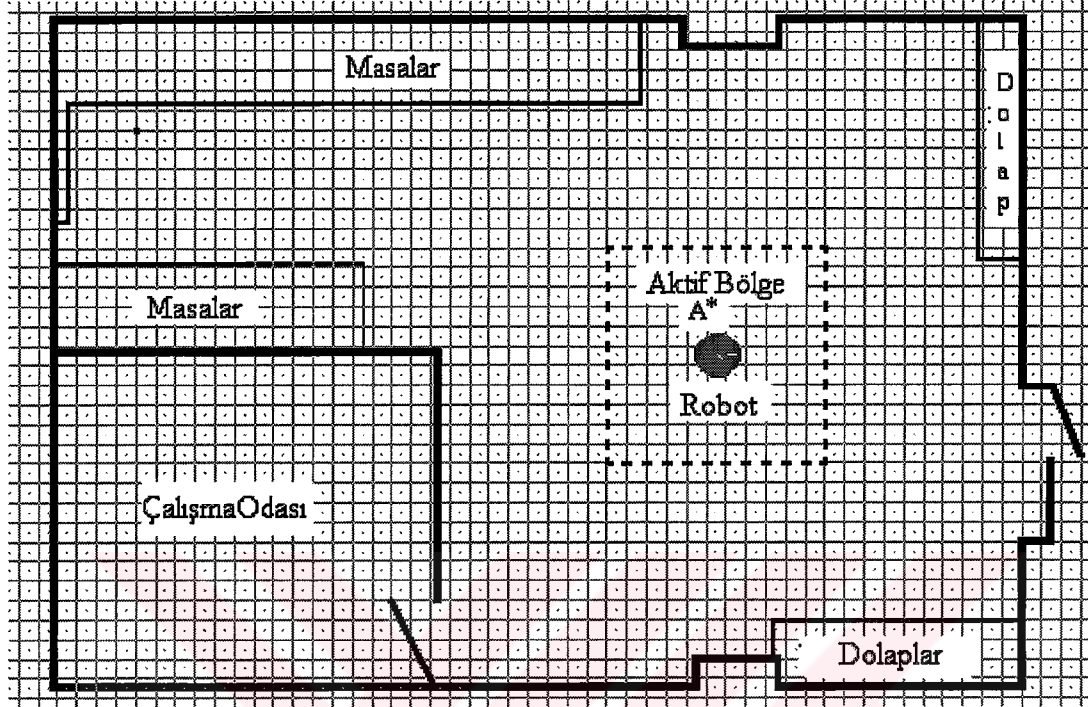
$$\vec{F}_r = \sum_{i,j} \vec{F}_{i,j} \quad (6.2)$$

burada, F_{cr} : itici kuvvet sabiti, x_r, y_r : robot konumu, x_i, y_j : her bir aktif grid hücresinin konum, $d(i,j)$: her bir aktif hücrenin robota olan uzaklığıdır.

(6.1)’deki eşitlikte kuvvet sabiti $d^2(i,j)$ ile bölündüğünden yakın mesafedeki engeller kuvvetli, uzak mesafedeki engeller ise zayıf itici kuvvetler meydana getirir. Mobil robot üzerinde meydana gelen toplam itici kuvvet her bir hücrenin meydana getirdiği kuvvetlerin toplamı şeklinde (6.2)’deki eşitlik kullanılarak hesaplanabilir.

Verilen hedefin mobil robota herhangi bir anda uyguladığı çekici kuvvet ise (6.3)’deki eşitlik kullanılarak kolayca hesaplanabilir. Hesaplanan F_t vektörü (6.3)’deki eşitlikte verildiği gibi robotun hedefe olan mutlak uzaklığından bağımsızdır. Böylece verilen hedeflerin yakın veya uzak oluşlarına göre değil de açısal konumlarına göre mobil robota çekici bir kuvvet uygulanması sağlanmıştır.

Şekil 6.3’de mesafe bilgisi şeklinde açığa çıkartılan engellerin robota uyguladığı toplam itici kuvvet ile hedefin oluşturduğu çekici kuvvetlerin vektörel toplamı (6.4)’deki eşitlikte verilmiştir.



Şekil 6.2 Robotla birlikte hareket eden 11x11 lik aktif A^* bölgesi.

Robotun verilen hedefe ulaşabilmesi için hangi yön doğrultusunda hareket edeceği R vektörü kullanılarak (6.5)’deki eşitlik ile hesaplanabilir.

$$\vec{F}_t = F_{ct} \left(\frac{x_t - x_r}{d_t} \vec{x} + \frac{y_t - y_r}{d_t} \vec{y} \right) \quad (6.3)$$

$$\vec{R} = \vec{F}_r + \vec{F}_t \quad (6.4)$$

$$\beta_R = \frac{\vec{R}}{|\vec{R}|} \quad (6.5)$$

burada, F_{ct} : çekici kuvvet sabiti, d_t : planlanan hedef ile robot arasındaki mesafe, x_b , y_t : hedef noktasının koordinatlarıdır. Eşitlik (6.1) ve (6.3)’de yer alan F_{cr} itici ve F_{ct} çekici kuvvet sabitleri ortamdaki engel yoğunluğuna bağlı olarak 1 ile 5 arasında

6.3.1.1 Mobil Robot Hız Bilgisinin Oluşturulması

$$V = V_m \left(1 - \mu \frac{|\vec{F}_r|}{|\vec{R}|} \right) \quad 0 < \mu \leq 1 \quad (6.6)$$

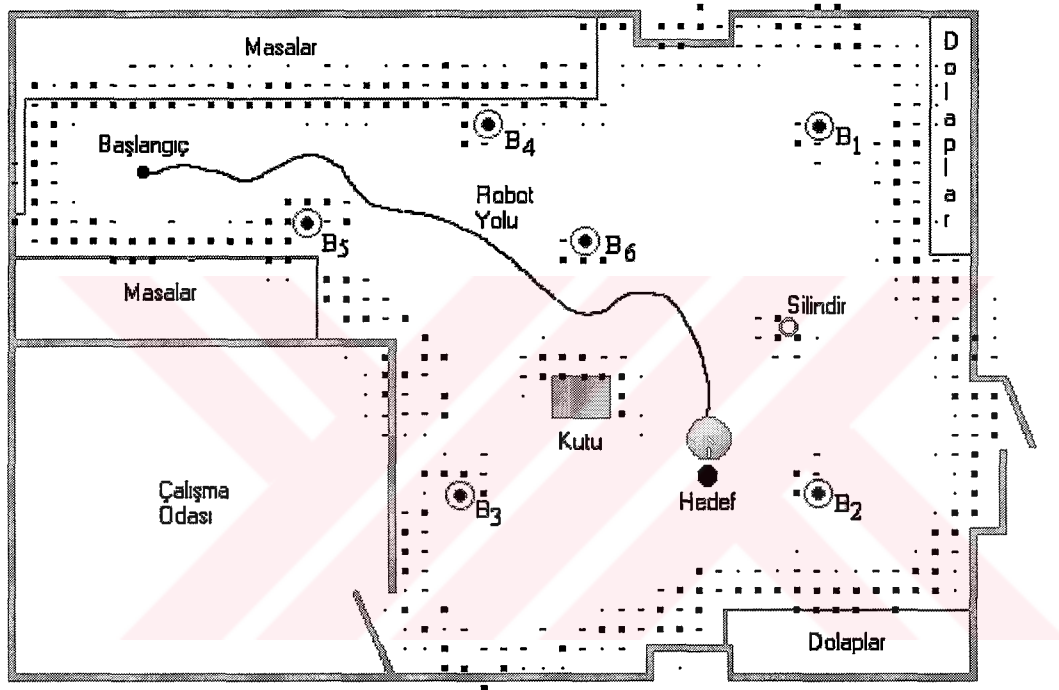
burada, V_m Nomad200 mobil robotunun yapabileceği en fazla hız olan 0.508 m/s dir. μ değişkeninin F_r vektörü ve bir önceki hıza bağlı olarak alabileceği olası değerler bir tablo halinde bilgisayar belleğine saklanmıştır. Çalışılan ortamdaki engel yoğunlukları dikkate alınarak hazırlanan tablo sayesinde zaman kaybedilmeden robot hızı yeni verilere göre ayarlanır.

6.3.1.2 YPA Algoritmasının Nomad200 Mobil Robotu Üzerinde Testi

Bu bölümde genel kuvvet denklemleriyle verilen YPA algoritması 4. ve 5. bölümlerde anlatılan lokalizasyon ve çevre haritalama yöntemleri ile birlikte Nomad200 mobil robotuna uygulanmıştır. Laboratuvar içinde oluşturulan ortam yarı bilinen bir ortamdır. Bu ortamda robota, sadece hareketine başlamadan önceki ilk konum ve yönü ile çevreye uygun aralıklarla dağıtılmış referans nesnelerin ($B_1...B_6$) konumları verilmektedir. İlk konumunu bilen robot, ultrasonik ve kızılötesi sensörleriyle etrafını tarayarak Bölüm 5.4’de anlatılan ATB yöntemiyle kendisine uzaklığı azami 300 cm. olan bütün nesneleri tespit eder ve bu nesnelerin üzerinde bulundukları grid hücrelerine nesnelerin temsil edilme olasılık değerlerini atar. Robot bu aşamada kendisinden 300 cm.ye kadar uzaklıktaki bir çevre bilgisine sahiptir ve hedef koordinatları doğrultusunda yolunu planlamaya hazırdır. Planlama safhasında, robotu merkez alan 220cm.x220cm. lik bir kare alan (11x11 grid hücresi) içinde yer alan bütün grid hücrelerinin robota uyguladığı itiş kuvvet vektörleri F_{ij} ve robot üzerinde oluşan toplam itiş kuvveti vektörü F_r (6.1) ve (6.2)’deki eşitliklerden hesaplanır. Sonra, verilen hedef noktasının robot üstünde yarattığı kendisine doğru çekiş kuvveti vektörü olan F_t eşitlik (6.3) ile bulunur. Robot üzerinde oluşan itiş ve çekiş kuvvet vektörlerinden planlanacak yolun yönünü de eşitlik (6.5)’deki gibi gösterecek olan bileşke kuvvet vektörü olan R eşitlik (6.4)’den hesaplanır. R vektörünün yönü robot tekerleğine uygulanacak dönüş miktarını bildirmektedir. Hesaplanan yön doğrultusunda robotun hangi hızlarda hareket edeceği ise eşitlik (6.6) kullanılarak bulunabilir.

YPA algoritmasının uygulandığı robotun yapacağı maksimum navigasyon hızı algoritmanın bilgisayarda harcadığı zamandan ziyade ultrasonik sensörlerin örnekleme oranı ile sınırlandırılmaktadır. Nomad200 mobil robotunun kullandığı sistemde 16 ultrasonik sensörden veri alınması (≈ 110 msn.) ve işlenmesi aşamasında geçen zaman 250 msn. dir. Ultrasonik sensör bilgileri çevre haritalamanın yanında

lokalizasyon amaçlıda işlendiğinden bilgisayarda harcanan zaman iki katına çıkabilmektedir. Robot üzerindeki bilgisayar ve RF modemle haberleştiği bilgisayar Pentium-133 dür. Şayet bu bilgisayarlar daha hızlı bir bilgisayarla yer değiştirilirse sensör bilgilerinin işlenmesi sırasında geçen zaman en aza indirilerek robot hızı arttırılabilir. Mevcut donanım imkanlarıyla, robotun herhangi bir engele çarpmaması için hızı maksimum 26 cm/sn ye ayarlanmıştır. Şekil 6.4’de gösterilen laboratuvar ortamında robot verilen hedefine yolunu planlarken minimum 5cm/sn, ortalama ise 15 cm/sn ile hareket etmiştir.



Şekil 6.4 Laboratuvar ortamında YPA algoritması ile planlanan robot yolu.

6.4 Eş-Zamanlı Üç Tekerlek Yapısındaki Robotlarda Hareket Planlama

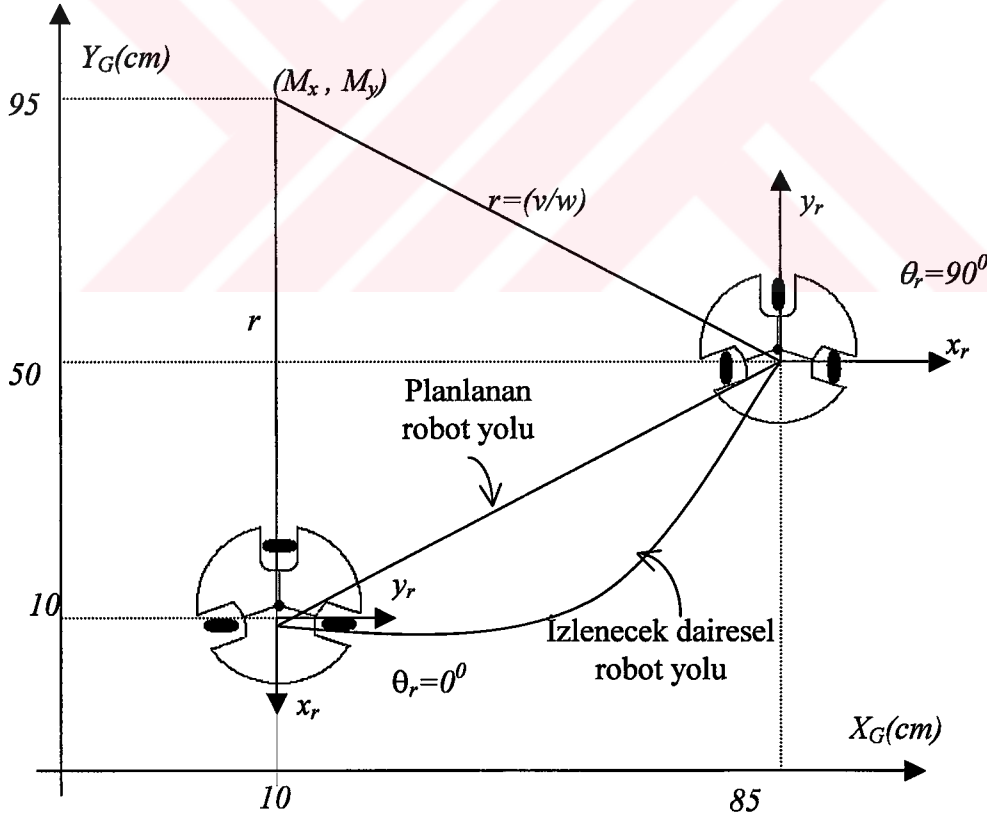
Eş zamanlı sürücü sisteme sahip mobil robotlarda tekerleği ilerleten ve döndüren hızlar sınırlı bir tork ile birbirinden bağımsız olarak kontrol edilebilir. Dieter Fox ve arkadaşları tarafından bu durum dikkate alınarak işlenen hareket denklemlerinde [62] hız zamanda parçalı değişmez bir fonksiyon gibi modellenmiştir. Bu varsayımlar altında robotun izlediği yollar bir dizi sınırlı dairesel parçalar içerir. Böyle bir yapılanma, engellerden sakınma açısından yararlıdır. Zira daireler ile robot yolunun bir engel ile kesişip kesişmediğini kontrol etmek kolaydır.

Eş-zamanlı mobil robotlara ait hareket denklemlerinin çıkarılması “Genel Hareket Denklemleri” ve “Yaklaşık Hareket Denklemleri” başlıkları altında Ek-A’da verilmektedir.

Ek-A’da verilen yaklaşık hareket denklemleri Şekil 6.5’deki robot koordinat sistemine (lokal) uygulandığında, robotun verilen “ v ” doğrusal ve “ w ” açısal tekerlek hızı komutlarıyla izleyeceği yol eşitlik (6.1)’de verilen daire ile tanımlanabilir. Buradaki lokal (robot) koordinat çerçevesi anlatım kolaylığı olsun diye saat yönünün tersine 90^0 döndürülerek verilmektedir. Bu durum algoritmanın sisteme entegrasyonunda dikkate alınmaktadır.

$$(x_r - M_x)^2 + (y_r - M_y)^2 = \left(\frac{v}{w}\right)^2 \quad (6.7)$$

burada, x_r, y_r : lokal koordinat sistemindeki robot konumu, M_x, M_y : robot yolunu kesen dairenin merkezi, (v/w) : dairenin yarıçapıdır.



Şekil 6.5 Robot yolunun dairesel parçalarla oluşturulması.

Şekil 6.5’de sayısal değerlerle verilen örnekteki dairenin yarıçapı Ek-A’daki $x(t_n)$, $y(t_n)$ eşitliklerinden ilk koşullar sıfır alınarak (lokal koordinat sistemine geçilmekte) $r=(v/w)=85 \text{ cm/rad}$ olarak hesaplanabilir. Buradan da lokal koordinat sistemindeki dairenin merkezi, robotun baş açısı $\theta_r=0^0$ olduğu dikkate alınarak $(M_x, M_y)=(0.0, 85.0) \text{ cm}$. şeklinde bulunabilir. Kullanılan eşitlikler yaklaşık ifadeler olduğundan planlanan doğrusal yol ile dairesel robot yolu bitiş noktasında çakışmamaktadır. Bu durum Şekil 6.5’deki örnek için doğrusal yola ait $y=(40/75)*x$ ve dairesel yola ait $x^2+(y-r)^2=r^2$ eşitlikleri kullanılarak giderilebilir. Verilen örnekte $(85.0, 50.0) \text{ cm}$. koordinatlarında çakışması gereken yollar yaklaşık $(81.0, 48.0) \text{ cm}$. koordinatlarında çakışmaktadır.

Robotun sadece verilen “v” doğrusal tekerlek hızı komutlarıyla (bu durumda $w=0$) bir yolu izlemesi durumunda global koordinat sistemindeki robot konumu eşitlik (6.8a) ve (6.8b)’den kolayca hesaplanabilir.

$$x_r(t) = x_r(t_o) + v \cdot t \cdot \cos \theta_r(t) \quad (6.8a)$$

$$y_r(t) = y_r(t_o) + v \cdot t \cdot \sin \theta_r(t) \quad (6.8b)$$

burada, v : robot tekerleğine uygulanan doğrusal hız miktarı (cm/sn), θ_r : robot baş açısı, $x_r(t_o)$, $y_r(t_o)$: robotun hareket etmeden önce global koordinat sisteminde bulunduğu konumdur.

6.5 Sonuç

Bu bölümde yapılan çalışmalarda 4. ve 5. bölümlerde verilen lokalizasyon ve çevre haritalama yöntemlerinin birlikte kullanımı ile bir mobil robotun navigasyonunun temel unsuru olan hedefe olan yolun planlanması ele alınmıştır. İlk olarak mobil robotlarda lokalizasyon ve çevre haritalama işleminin beraber kullanımı konusunda geçmişte yapılan çalışmalara yer verilerek yol planlamasında kullanılabilecek algoritmalarından bahsedilmiştir. Çalışmaların yapıldığı ortam yarı bilinen kapalı bir ortam (mobil robot laboratuvarı) olduğundan bu algoritmaların en uygunu olan Yapay Potansiyel Alanları-YPA algoritması yol planlama amaçlı seçilerek, yöntem hakkında detaylı bilgi verilmiştir. Daha sonra lokalizasyon ve çevre haritalama

yöntemlerini beraber kullanabilen YPA algoritması deney amaçlı bir mobil robot olan Nomad200 üzerinde test edilmiştir. Laboratuvar ortamındaki masa, dolap, kutu ve birtakım referans nesneler arasından robotun bu yöntemleri kullanarak, verilen başlangıç ve hedef noktaları arasında planladığı yol çizdirilmiş, sensörlerin örnekleme zamanları ve navigasyon süresince oluşan robot hızı konusunda karşılaşılan sorunlara yer verilmiştir.

Kullanılan mobil robot eş-zamanlı üç tekerleğe sahiptir ve sensör okumalarını sürekli hareket halindeyken yapmaktadır. Planlanan yolda zaman içinde keskin dönüşler olabilmektedir. Robotun bu değişimlere zorlanmadan uyum gösterebilmesi için bu bölümde ayrıca Ek-A'da geniş bir şekilde verilen eş-zamanlı üç tekerlek yapısındaki robotlarda hareket planlama konusunda da bilgi verilmiştir.



7. SONUÇLAR VE ÖNERİLER

7.1 Sonuçlar

Mobil robot alanında yapılan çalışmaların en önemli amaçlarından biride, robota kapalı düzenli ortamlarda bir çok görevi yapabilme kabiliyetini kazandırabilmektir. Bunu yapabilmesi için robot bulunduğu çevreden ve kendi konumundan haberdar olmalıdır. Robotun bunu başarabilmesi için, çevre şartlarına müdahale etmeyen ve beklenmeyen engellerden etkilenmeyen bir lokalizasyon yöntemini bulunduğu ortamın sayısal haritasını çıkaran ve sensör verileri ile onu sürekli güncelleyen bir çevre haritalama algoritması ile birleştirilmesi gereklidir. Hareket esnasında yapılan sürekli örneklemelerle doğru ölçemeyen ultrasonik sensörler gibi gürültülü sensörlerin modellemesine uygun bir sayısal çevre temsili ortaya çıkarılabilir. Daha önce yapılan bir çok araştırmada lokalizasyon ve çevre haritalama problemlerinin çözümüne yönelik sonuçlar ayrı ayrı açıklanmıştır. Fakat her iki problemin beraber çözümüne yönelik çalışmalara daha az ağırlık verilmiştir.

Çalışmada lokalizasyon ve çevre haritalama konularıyla beraber uğraşarak bu amaca yönelik geliştirilen yöntemler tanıtılmış ve bir laboratuvar mobil robotu olan Nomad200 üzerinde yapılan uygulamalara yer verilmiştir. Ayrıca karşılaştırma yapabilme amacıyla bu konularda iyi sonuçlar veren bazı eski yöntemlerin de uygulamaları robot üzerinde yaptırılmıştır.

Mobil robot hareketlerinde meydana gelebilecek hataları analiz edebilmek için, ilk olarak mobil robot tekerlek sistemlerinin kinematik modellemesi üzerinde detaylı durularak, günümüzde çok yaygın kullanılan eş-zamanlı üç tekerleğe sahip mobil robotların kinematik modellemesi yapılmıştır. Robot üzerinde yer alan ve robotun mutlak konum ve yönünü bildiren odyometrik sistemde meydana gelebilecek hatalar, onun kinematik modeli kullanılarak ortaya çıkarılabilmüş ve bu hataları en aza indirici tedbirler alınabilmıştır. Bununla beraber, robot üzerinde yer alan ultrasonik ve kızılötesi sensörler üzerinde hata analizi yapılarak çevre haritalama amaçlı kullanılabilecek matematiksel modelleri tanıtılmıştır. Ayrıca robot üzerine harici takılabilen bir manyetik pusula ile bir fiber optik jiroskopun teknik özellikleri detaylı verilerek çalışma şartları anlatılmıştır. Mobil robot yönünü mutlak olarak okuyabilen manyetik pusulanın kapalı ortamlarda zaman zaman okuma yapamadığı, bunun yanı

sıra okuma yapabildiği anlarda da 0.01 derecelik bir hata payı ile çok hassas okuma yapabildiği tespit edilmiştir.

Lokalizasyon konusunda yapılan çalışmalarda geliştirilen Newton Özyinelemeli Lokalizasyon yöntemi sadece ultrasonik sensörleri kullanarak robot etrafındaki referans nesneleri ($B_1...B_2$) yakaladığından bu sisteme Ultrasonik Konumlandırma Ünitesi adı verilmiştir. Bu üniteden elde edilen bilgi odyometre sistemi gibi belirli bir hata ile mutlak robot konumu ile yönünü verdiğiinden yine lokalizasyon amaçlı kullanılan Genişletilmiş Kalman Filtreleme-GKF yönteminde ve geliştirilen Sınırlandırılmış Dinamik Sensör Füzyonu-SDSF yöntemlerinde kullanılmıştır. Bu yöntemlerde ayrıca manyetik pusula ve fiber-optik jiroskop okumalarından da yararlanılmıştır. Her iki yöntemin Nomad200 mobil robotuna uygulanması sonucunda elde edilen veriler karşılaştırılarak geliştirilen SDSF yönteminin GKF yönteminden geri kalmadığı grafiklerle vurgulanmıştır.

Çevre haritalama konusunda yapılan çalışmalarda ise ortamın 20X20 cm² lik hücreler şeklinde ızgaralandığı varsayılarak bu hücreler üzerinde engel bulunma olasılıkları Bayes ve geliştirilen Ardışıl Temel Bileşenler-ATB yöntemleriyle hesaplanabilmiştir. Her iki yöntemde de ultrasonik ve kızılötesi sensörlere ait modeller çevre haritalama amaçlı kullanılmaktadır. Lokalizasyon bilgisini alan çevre haritalama algoritmaları ultrasonik ve kızılötesi sensör okumalarına göre hücreler üzerinde engel bulunma olasılıklarını hesaplamaktadır. Her iki yöntem karşılaştırıldığında ATB yönteminin hesaplama zamanının aynı şartlarda Bayes yöntemine göre %40 daha çabuk olduğu gözlenmiştir. ATB yönteminde yanlış sensör okumalarına daha başında müdahale edilerek, ilgili ızgara hücre değerlerinin doğru bir şekilde kıymetlendirilmesi sağlanırken Bayes yönteminde hatalı sensör okumaları doğrudan yöntemin içinde yer alarak hatalı atamalara sebep olmaktadır. ATB yönteminde okunan sensör verilerinin olasılıksal değerleri doğrudan atanırken Bayes yöntemi ile yapılan çalışmalarda Demster-Shafer gibi birleştirici algoritmalara ihtiyaç duyulmaktadır.

Lokalizasyon ve çevre haritalama konusunda yapılan çalışmaların bir bütün haline getirilerek mobil robot navigasyonunda kullanılabilmesi için Şekil 6.1'deki yapı planlanmıştır. Bu yapıda ilk olarak robot nerede olduğunu ortamdaki referans nesneleri bulmaya çalışarak sorgular. Lokalizasyon işleminin bu aşamada tamamlanmasından sonra robotun navigasyonunu gerçekleştireceği yol üzerinde bir çevre haritalama işlemi yapabilmesi için gereken şartlar tam olarak yerine getirilmiştir. Izgara hücreleri şeklinde tanımlanan ortama ultrasonik ve kızılötesi verilerine göre engel bulunma olasılık değerleri atanmasından sonra robotu verilen

hedefine engellere çarpmadan götürecek bir yolun planlanmasına gerek duyulur. Bu aşamada yol planlama algoritmaları içinden en kolay uygulanabilir ve hesaplama zamanı en kısa olan Yapay Potansiyel Alanları-YPA yöntemi bu amaçla seçilmiştir. Ayrıca YPA algoritması ile planlanan yol üzerinde robot hareketlerinin olası keskin dönüşlere karşı yumuşatmak amacıyla eş-zamanlı tekerlek sistemine uygun bir hareket planlama algoritması da kullanılmıştır.

Mobil robotun lokalizasyonunda kullanılan sensörlerden biri olan jiroskop ile robotun yaptığı açısal hız ölçülerek robotun yaptığı bağıl dönüş açıları ortaya çıkarılabilmektedir. Fakat robotun yaptığı düz hareketlerde ivme ölçülemediğinden oluşan hız farkları sistem hatası olarak ele alınmıştır. Bu çalışmadan sonra gerçekleştirilecek çalışmalarda bir ivme ölçer (akselometre) kullanılırsa robot hızında meydana gelen değişimler kolaylıkla algılanarak lokalizasyon işleminin daha az hata ile yapılabilmesine olanak sağlayacaktır.

İleriye yönelik artan bilgisayar hızları ve yapılacak olan optimizasyonlar ile mesafe okuyucu sensör bilgileri yüksek hızlarda birleştirilerek işlenebilecektir. Bu durumda mobil robot planladığı yol üzerinde yüksek hızlarda hareket edebilecek ve aniden önüne çıkabilecek engellerden sakınabilecektir.

7.2 Geleceğe Yönelik Tavsiyeler

Kullanılan sensörlerle modelin bütün değişkenleri tam olarak gözlenememektedir. Örneğin robot tekerleğinin doğrusal hızı v gibi. Bu nedenle bütün değişkenlerde oluşan belirsizliklerin tahmininde bazı sapmalar meydana gelmektedir. Uygulanan filtreleme algoritmasında bu belirsizlikleri azaltmak için harici farklı sensörler kullanılabilir. Örneğin robot tekerleğinin doğrusal hızını ölçebilecek bir akselometre gibi.

Robot referans noktasına göre koordinat çerçeveleri arasında yapılan dönüşümlere ilave olarak, robot üzerindeki herhangi bir “ p ” noktasının hareket esnasındaki hız ve ivmesi de robot referans noktası R ye göre (2.14) ve (2.15)’deki eşitlikler kullanılarak hesaplanabilir.

KAYNAKLAR

- [1] **Coiffet, P. and Chirouze, M.**, 1982. An Introduction to Robot Technology, *Publisher: McGrawHill*, pp.22, London.
- [2] **Luo, R. And Kay, M.**, 1989. Multisensor Integration and Fusion in Intelligent Systems, *in IEEE Transactions on Systems, Man, and Cybernetics*, 19 5(Sept.-Oct), pp.901-930.
- [3] **Muir, P.F. and Neuman, C.P.**, June 1986. Kinematic Modeling of Wheeled Mobile Robots, *in Robotics Institute Technical Report*, No. CMU-RI-TR-86-12, Carnegie Mellon University, Pittsburgh, PA 15213,.
- [4] **Everett, H.R.**, 1995. Sensors for Mobile Robots: Theory and Application, *Publisher: A K Peters Ltd*, Wellesley, MA, ISBN: 1-56881-048-2.
- [5] **Sheth, P. N. and Uicker, J. J.**, May, 1972. IMP (Integrated Mechanisms Program), A Computer-Aided Design Analysis Sustem for Mechanisms and Linkages, *ASME Journal of Engineering for Industry*, , pp.454-464.
- [6] **Nomad200**, Nomadic Technologies, 2133 Leghorn Street, Mountain View, CA 94043, nomad@robots.com.
- [7] **KVH Industries, Inc.**, C100 Compass Engine Technical Manual, 50 Enterprise Center, Middletown, RI 02842 USA, <http://www.kvh.com>.
- [8] **KVH Industries**, Autogyro Navigator Installation Manuel, 50 Enterprise Center, Middletown, RI 02842 USA, <http://www.andrew.com>.
- [9] **Cox, I.J.**, April 1991. Blanche-An experimental in guidance and navigation of an autonomous robot vehicle, *in IEEE Trans. Robotics and Autotomation*, vol.7, No.2, pp.193-204,.
- [10] **Chenavrier, F. and Crowley, J.L.**, May 1992. Position estimation for a mobile robot using vision and odometry, *in Proceedings of the 1992 IEEE Int. Conf. on Robotics and Automation*, pp.2588-2592, Nice France.

- [11] **Borenstein, J. and Feng, L.**, 1996. Measurement and correction of systematic odometry errors in mobile robots, *in IEEE J. Rob. Autom.*, 12(6), pp.869-880.
- [12] **Borenstein, J. and Feng, L.**, 1994. UMBmark-A method for measuring, comparing and correcting dead-reckoning errors in mobile robots, *in Technical Report, The University of Michigan*, UM-MEAM-94-22.
- [13] **MATLAB Software**, High Performance Numeric Computation and Visualization Software, The Math Works, Inc., Cochituate Place, 24 Prime Park Way, Natick, Mass., 01760, USA.
- [14] **Everett, H.R., Gage, D.W., Gilbreth, G.A., Laird, R.T., and Smurlo, R.P.**, 1994. Real-World Issues in Warehouse Navigation, *in Proceedings SPIE Mobile Robots IX*, Volume 2352, Boston, MA, Nov.2-4.
- [15] **Barshan, B. and Durrant-Whyte, H.F.**, 1995. Inertial navigation systems for mobile robots, *in IEEE Trans. Rob. Autom.* 11(3), pp.328-342.
- [16] **Kosaka, A. and Kak, A.C.**, November 1992. Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties, *in Computer Vision Graphics and Image Processing, Image Understanding*.
- [17] **Preciado, A., Meizel, A., Segovia, A. and Rombaut, M.**, April 1991. Fusion of multi-sensor data: A geometric approach, *in IEEE Int. Conf. Robot. Automat.*, pp. 277-289.
- [18] **Chatila, R.**, 1985. Mobile robot navigation: Space modelling and decisional process, *in Proc. Third Int. Symp. Robot. Res.*, pp. 415-421.
- [19] **Durrant, H. F. and Leonard, J. J.**, 1989. Navigation by correlating geometric sensor data, *in Proc. IEEE/RSJ Int. Workshop Intel. Robot. and Syst. '89*, Tsukuba, Japan, pp. 990-996.
- [20] **Sciele, B. and Crowley, J. L.**, 1994. A comparison of position estimation techniques using occupancy grids, *in Robot. And Autom. Syst.*, vol. 12, pp. 153-171.
- [21] **Leonars, J. J., Durrant, H. F. and Cox, I. J.**, August 1992. Dynamic map building for an autonomous mobile robot, *in Int. J. Robot. Res.*

- [22] **Yenilmez, L., Temeltaş, H.**, Dec., 1996. Enabling an Autonomous Mobile Robot to Determine its Actual Position and Orientation by Using an Ultrasonic Ring, *in ICARCV'96 Fourth Int. Conf. On Control, Automation, Robotics and Vision*, pp.815-820, Singapore.
- [23] **Leonard, J.J. and Durrant-White, H.F.**, 1991. Mobile Robot Localization and by Tracking Geometry Beacons, *in IEEE J. Robotics & Autom.*, 7, (3), pp(376)-(381).
- [24] **Moutarlier, P. and Chatila, R.**, June 1989. An experimental system for incremental environment modelling by an autonomous mobile robot, *in Exp. Robotics I, The First Int. Sympos.*, pp.327-346, Montreal.
- [25] **Yenilmez, L., Temeltaş, H.**, 1998. Mobil Robotlarda Kalman Filtreleme Tabanlı Konum Kestirimi ve Yol Planlaması, *SİU'98 6th Sinyal İşleme ve Uygulamaları Kurultayı*, sf. Kızılcahamam, Ankara/Türkiye.
- [26] **Yenilmez, L., Temeltaş, H.**, 1999. Reliable Mobile Robot Localization and Navigation Using Incremental Odometry, a Compass and a Gyroscope, *in 2nd International Conference on Recent Advances in Mechatronics-ICRAM'99*, pp. 484-488, Istanbul/Türkiye.
- [27] **Bar Shalom Y., Fortmann T.E.**, 1988. Tracking and Data Association, Academic Press, 1988.
- [28] **Hardt, H.J., Wolf, D. And Husson, R.**, 1996. Localization of a wheeled mobile robot using incremental odometry, a gyroscope and a magnetic compass, *in ICARCV'96 Fourth Int. Conf. On Control, Automation, Robotics and Vision*, pp.1116-1120, Singapore.
- [29] **Smith, R., and Cheeseman, P.**, 1986. On the representation and estimation of spatial uncertainty, *in the International Journal of Robotics Research*, vol.5, no.4, pp. 56-68.
- [30] **Brooks R. R., Iyengar S. S.**, 1996. Robust Distributed Computing and Sensing Algorithm, *in IEEE Computer*, vol.29 No.6, pp.53-60, June.
- [31] **Brooks R. R., Iyengar S. S.**, 1998. Multi-Sensor Fusion -Fundamentals and Applications with Software, Prentice-Hall, Inc.

- [32] **Araujo, E.G. and Grupen, R.A.**, May 1998. Feature detection and identification using a sonar-array, in *IEEE Int. Conf. Robot. And Automat.*, Leuven, Belgium.
- [33] **Moravec, H. P. and Elfes, A.**, March 1985. High resolution maps from wide angle sonar, in *proceedings of the IEEE Conference on Robotics and Automation*, Washington, D.C., pp.116-121.
- [34] **Fryxell, R. C.**, November 1987. Navigation planning using quad trees, in *SPIE Mobile Robots II*, Cambridge, MA, pp. 256-261.
- [35] **Beckerman, M. and Oblow, E.M.**, February 1988. Treatment of systematic errors in the processing of wide angle sonar sensor data for robotic navigation, in *Oak Ridge National Laboratory Technical Memo.*, CESAR-88/07.
- [36] **Elfes, A.**, June 1989. Using occupancy grids for mobile robot perception and navigation, in *IEEE Computer*, vol. 22, no. 6, pp. 46-57.
- [37] **Cahut, L., Valavanis, K.P. and Deli, H.**, May 1998. Sonar resolution-based environment mapping, in *Proc. IEEE 1998 Int. Conf. Robot. And Automat.*, Leuven, Belgium, pp.2541-2547.
- [38] **Gilbreath, G.A. and Everett, H.R.**, November 1988. Path planning and collision avoidance for an indoor security robot, in *SPIE Mobile Robots III*, Cambridge, MA, pp.19-27.
- [39] **Borenstein, J. and Koren, Y.**, August 1991. Histogramic in-motion mapping for mobile robot obstacle avoidance, in *IEEE Trans. Robot. Automat.*, vol.7,no.4, pp.535-539.
- [40] **Vandorpe, J., Van Brussel, H. and Xu, H.**, April 1996. Exact dynamic map building for a mobile robot using geometrical primitives produced by a 2D range finder, in *Proc. IEEE Int. Conf. Robot. and Automat.*, Minneapolis, Minnesota, pp.901-908.
- [41] **Borenstein, J. and Koren, Y.**, Sept/Oct 1989. Real-time obstacle avoidance for fast mobile robots, in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.19, No.5, pp.1179-1187.
- [42] **Crowley, J.L.**, May 1989. World modeling and position estimation for a mobile robot using ultrasonic ranging, in *Proceedings of the 1989 IEEE*

International Conference on Robotics and Automation, Scottsdale, Arizona.

- [43] **Korba, L.W., Liscano, R. and Durie, N.**, 1988. An intelligent mobile platform for health care applications, in *Proceedings of the ICAART 88 Conference*, pp.462-463, Montreal, Canada.
- [44] **Moravec, H.P.**, 1988. Sensor fusion in certainty grids for mobile robots, in *AI magazine*, pp.61-74.
- [45] **Elfes, A.**, 1987. Sonar-based real-world mapping and navigation, in *IEEE journal of robotics and automation*, vol.RA-3, no.3, pp.249-265.
- [46] **Polaroid Corporation**, Ultrasonic Components Group, 119 Windsor Street, Cambridge, Massachusetts 02139.
- [47] **Borenstein, J. and Koren, Y.**, 1988. Obstacle avoidance with ultrasonic sensors, in *IEEE Journal of robotics and automation*, vol.RA-4, no.2, pp.213-218.
- [48] **Kang, D., Luo, R.C., Hashimoto, H. And Harashima, F.**, 1994. Position estimation for mobile robot using sensor fusion, in *Proc. of the 1994 IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Sys. (MFI'94)*, pp.647-652, Las Vegas, Oct.2-5.
- [49] **Shafer, G.**, 1976. Mathematical Theory of Evidence, in *princeton University Press*.
- [50] **Hebb, D.O.**, 1949. The organization of behavior, *Wiley*, New York.
- [51] **Nomad200, User's Manuel**, Nomadic Technologies, 2133 Leghorn Street, Mountain View, CA 94043, nomad@robots.com.
- [52] **Thrun, S., Fox, D. and Burgard, W.**, May 1998. Probabilistic mapping of an environment by a mobile robot, in *Proc. of IEEE Int Conf. Robot. and Automat.*, Leuven, Belgium.
- [53] **Castellanos, J.A, Martinez, J.M., Neira, J. and Tardis, J.D.**, May 1998. Simultaneous map building and localization for mobile robots: A multisensor fusion approach, in *Proc. IEEE Int. Conf. Robot. and Automat.*, Leuven, Belgium.

- [54] **Yamauchi, B., Schultz, A. and Adams, W.**, May 1998. Mobile robot exploration and map-building with continuous localization, in *Proc. IEEE Int. Conf. Robot. and Automat.*, Leuven, Belgium.
- [55] **Chong, K.S. and Kleeman, L.**, April 1997. Sonar based map building for mobile robot, in *IEEE Int. Conf. Robot. And Automat.*, Albuquerque, New Mexico.
- [56] **Owen, C. and Nehmzow, U.**, 1996. Route learning in mobilr robots through self-organization, in *Euromicro workshop on advanced mobile robotics*, IEEE Compute Society, ISBN 0-8186-7695-7.
- [57] **Warren, C. W.** 1989. Global path planning using artificial potantial field, in *IEEE J. Robotics and Automation*, page 316 to 610.
- [58] **Yenilmez, L., Temeltaş, H.**, May 1998. Real Time Multi-Sensor Fusion and Navigation for Mobile Robots, in *Melecon'98 : 9th Mediterranean Electrotechnical Conference*, pp.203-211, Tel-Aviv, İsrail.
- [59] **Borenstein, J. and Koren, Y.**, June 1991. The vector field histogram-Fast obstacle avoidance for mobile robots, in *IEEE Transactions on Robotics and Automation*, Vol.7, No.3, pp.278-288.
- [60] **Luo, R. and Kay, M.**, 1989. Multisensor integration and fusion in intelligent systems, in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.19, No.5, pp.901-930.
- [61] **Lee, D. T. and Drysdale, R. L.**, 1981. Generalized Voronoi Diagrams in the Plane, in *SIAM J. Comput.*, Vol.10, No.1, pg.73.
- [62] **Fox, D., Buhmann, J., Burgard, W., Cremers, A.B., Hoffmann, T.**, 1995. The mobile robot Rhino, in *AI Magazine*, 16(1).

EK.A. EŞ-ZAMANLI TEKERLEK YAPISINDAKİ MOBİL ROBOTLARDA HAREKET DENKLEMLERİ

Bu bölümde eş-zamanlı tekerlek sürücüsüne sahip bir mobil robot için üretilen temel hareket denklemleri anlatılmaktadır [4]. (A.1) ve (A.2)'de verilen denklemlerin çıkarılmasında robot tekerleklerinin doğrusal ve dönüşsel hızlarının sınırlandırılmış torklar ile birbirinden bağımsız kontrol edilebildiği dikkate alınmıştır. Denklemleri daha pratik hale getirebilmek için hızları zamanda parçalı değişmez bir fonksiyon gibi modelleyen bir yaklaşım [Dieter Fox] (A.3) ve (A.4)'de "Yaklaşık Hareket Denklemleri" başlığı altında verilmiştir. Bu yaklaşıma göre, robotun izlediği yollar sonlu dairesel parçalar dizisini içermektedir. Böyle bir yaklaşım, engellerin daireler ile kesişimlerini kolayca kontrol edebilme açısından, bir başka deyişle robotun yolu üzerindeki engellere çarpma noktasının kolayca tespiti açısından uygundur.

A.1 Genel Hareket Denklemleri

Bir global koordinat sisteminde t -zamanında robotun koordinatları $x(t)$ ve $y(t)$ ile yönü (baş açısı) ise $\theta(t)$ ile gösterilsin. Bu durumda üçlü $\langle x, y, \theta \rangle$ ifadesi robotun kinematik konfigürasyonunu anlatır. Bir eş-zamanlı sürücülü robotun hareketi doğrusal hızın (v) daima robotun dönüşü (θ) istikametinde olması ile kısıtlanmıştır. Bu durum holonomik olmayan robotlara özgü bir kısıtlamadır [9]. Robotun x -koordinatlarında t_o ve t_n anlarındaki konumu $x(t_o)$ ve $x(t_n)$ ile gösterilsin. Robotun t -zamanındaki teker doğrusal hızı $v(t)$ ile dönüş hızı ise $w(t)$ ile ifade edilsin. Bu durumda $x(t_n)$ ve $y(t_n)$ fonksiyonları $x(t_o)$, $v(t)$ ve $\theta(t)$ 'ye bağlı olarak üretilebilir:

$$x(t_n) = x(t_o) + \int_{t_o}^{t_n} v(t) \cdot \cos \theta(t) \cdot dt \quad (A.1)$$

$$y(t_n) = y(t_o) + \int_{t_o}^{t_n} v(t) \cdot \sin \theta(t) \cdot dt \quad (A.2)$$

Robot hızlarına bağlı (A.1) ve (A.2) eşitlikleri doğrudan kullanılamaz. Bunun sebebi de, $v(t)$ hızının t_o anındaki $v(t_o)$ doğrusal hızına ve $\hat{t} \in [t_o, t]$ zaman aralığındaki $\dot{v}(\hat{t})$ doğrusal ivmesine bağımlı olmasıdır. Aynı şekilde, $\theta(t)$ yönlenme açısı t_o anındaki başlangıçtaki yönlenme $\theta(t_o)$, başlangıçtaki dönüş hızı $w(t_o)$ ile $\hat{t} \in [t_o, t]$ zaman aralığındaki $\dot{w}(\hat{t})$ dönüş ivmesine bağımlıdır. Bu durumda $v(t)$ ve $\theta(t)$ 'nin yerine

karşılığı olan başlangıç kinematik ve dinamik konfigürasyonları $v(t_o)$, $\theta(t_o)$, $w(t_o)$ ile $\dot{v}(\hat{t})$ ve $\dot{w}(\hat{t})$ ivmeleri konulduğunda aşağıdaki ifadeler ortaya çıkar:

$$x(t_n) = x(t_o) + \int_o^t \left(v(t_o) + \int_o^{\hat{t}} \dot{v}(\hat{t}) \cdot d\hat{t} \right) \cdot \cos \left(\theta(t_o) + \int_o^t \left(w(t_o) + \int_o^{\hat{t}} \dot{w}(\hat{t}) \cdot d\hat{t} \right) \cdot d\hat{t} \right) \cdot dt \quad (\text{A.3})$$

$$y(t_n) = y(t_o) + \int_o^t \left(v(t_o) + \int_o^{\hat{t}} \dot{v}(\hat{t}) \cdot d\hat{t} \right) \cdot \sin \left(\theta(t_o) + \int_o^t \left(w(t_o) + \int_o^{\hat{t}} \dot{w}(\hat{t}) \cdot d\hat{t} \right) \cdot d\hat{t} \right) \cdot dt \quad (\text{A.4})$$

Üretilen bu eşitlikler robotun t_o anındaki başlangıç dinamik konfigürasyonu ile kontrol edilebilir ivmelerine bağımlı olarak onun izlediği yol üzerindeki konumunu ifade etmektedir. Bir çok mobil robot için onun hareketini sağlayan ivmeler teker sistemini süren motorlardan akan akımların tekdüze bir fonksiyonu şeklindedir [6]. Bu nedenle motor akımlarındaki sınırlar doğrudan ivmelerin sınırlarını belirler.

Robot tekerleklerini süren motorlara uygulanan akımın bir şekilde değişmesi gibi (bu durumda ivmelerde de değişim olur) donanımdan kaynaklanan ilave kısıtlamalarda mevcuttur. Bu nedenle (A.3) ve (A.4)'de verilen eşitlikler robotun t_o ve t_n zamanlarındaki iki keyfi nokta arasında birçok sonlu ivme kumandası ile kontrol edilebildiği dikkate alınarak basitleştirilebilir. Burada “ n ” ile zaman imlerinin sayısı ifade edilsin. Bu durumda, $i=1...n$ için $\dot{v}_i$ ve $\dot{w}_i$ ivmeleri $[t_i, t_{i+1}]$ zaman aralığında sabit tutulur. $i=1...n$ dizin aralığında Δ_t^i ve $\Delta_{\hat{t}}^i$ sembolleri $\Delta_t^i = t - t_i$ ve $\Delta_{\hat{t}}^i = \hat{t} - t_i$ ifadeleri için kullanılsın. Bu ayrık formu kullanarak, eş-zamanlı sürücü sisteme sahip mobil robotların dinamik davranışları, parçalı değişmez ivmeler varmış gibi dikkate alınarak aşağıdaki (A.5) ve (A.6) da ki eşitlikler şeklinde ifade edilebilir.

$$x(t_n) = x(t_o) + \sum_{i=0}^{n-1} \int_i^{i+1} \left(v(t_i) + \dot{v}_i \cdot \Delta_t^i \right) \cdot \cos \left(\theta(t_i) + w(t_i) \cdot \Delta_t^i + \frac{1}{2} \cdot \dot{w}_i \cdot (\Delta_t^i)^2 \right) \cdot dt \quad (\text{A.5})$$

$$y(t_n) = y(t_o) + \sum_{i=0}^{n-1} \int_i^{i+1} \left(v(t_i) + \dot{v}_i \cdot \Delta_t^i \right) \cdot \sin \left(\theta(t_i) + w(t_i) \cdot \Delta_t^i + \frac{1}{2} \cdot \dot{w}_i \cdot (\Delta_t^i)^2 \right) \cdot dt \quad (\text{A.6})$$

A.2 Yaklaşık Hareket denklemleri

Her ne kadar eşitlik (A.5 ve (A.6) mobil robot kontrolünün genel durumunu anlatsa da, gerçek dönüş yönünün bulunmasında faydasız kalmaktadır. Bunun da nedeni bu eşitlikler tarafından ortaya konan robot yollarının karmaşık ve karşılaşılabilecek

engellerle kesişim noktalarının kontrolü amacıyla yapılacak geometrik işlemlerin hesaplama açısından zaman alıcı olmasıdır.

Daha pratik bir model çıkartabilmek amacıyla eşitlik (A.5) ve (A.6) $[t_i, t_{i+1}]$ zaman aralığındaki robot hızları sabit alınarak basitleştirilir. Eşitlik (A.9) ve (A.10) şeklinde ortaya çıkan sonuç hareket denklemleri zaman aralığının uzunluğu sıfıra giderken (A.5) ve (A.6)'da ki eşitliklere yakınsar. Bu varsayımlar altında bir robotun izlediği yol parçalı dairesel yaylar şekline dönüştürülebilir.

Şayet $[t_i, t_{i+1}]$ zaman aralığı oldukça kısa ise, $v(t_i) + \dot{v}_i \cdot \Delta_t^i$ terimi zaman içindeki robot hareketlerini düzleştirmek amacıyla bir $v_i \in [v(t_i), v(t_{i+1})]$ doğrusal hız şeklinde atanabilir. Aynı şekilde $\theta(t_i) + w(t_i) + \frac{1}{2} \cdot \dot{w}_i \cdot (\Delta_t^i)^2$ terimi yerine $w_i \in [w(t_i), w(t_{i+1})]$ olacak şekilde $\theta(t_i) + w_i \cdot \Delta_t^i$ terimi de atanabilir. Bu durumda (A.7) ve (A.8)'deki hareket denklemleri ortaya çıkartılır.

$$x(t_n) = x(t_o) + \sum_{i=0}^{n-1} \int_{t_i}^{t_{i+1}} v_i \cdot \cos(\theta(t_i) + w_i \cdot (\hat{t} - t_i)) \cdot d\hat{t} \quad (A.7)$$

$$y(t_n) = y(t_o) + \sum_{i=0}^{n-1} \int_{t_i}^{t_{i+1}} v_i \cdot \sin(\theta(t_i) + w_i \cdot (\hat{t} - t_i)) \cdot d\hat{t} \quad (A.8)$$

Eşitlik (A.7) ve (A.8)'deki entegrallerin çözümüyle de (A.9) ve (A.10)'da ki eşitlikler elde edilebilir.

$$x(t_n) = x(t_o) + \sum_{i=0}^{n-1} (F_x^i(t_{i+1})) \quad (A.9)$$

$$y(t_n) = y(t_o) + \sum_{i=0}^{n-1} (F_y^i(t_{i+1})) \quad (A.10)$$

burada,

$$F_x^i(t) = \begin{cases} \frac{v_i}{w_i} \cdot (\sin \theta(t_i) - \sin(\theta(t_i) + w_i \cdot (t - t_i))), \dots w \neq 0 \dots \text{için} \\ v_i \cdot \cos(\theta(t_i)) \cdot t, \dots w = 0 \dots \text{için} \end{cases} \quad (A.11)$$

$$F_y^i(t) = \begin{cases} -\frac{v_i}{w_i} \cdot (\cos \theta(t_i) - \cos(\theta(t_i) + w_i \cdot (t - t_i))), \dots w \neq 0 \dots \text{için} \\ v_i \cdot \sin(\theta(t_i)) \cdot t, \dots w = 0 \dots \text{için} \end{cases} \quad (A.12)$$

Verilen eşitliklere göre dikkat edilmesi gereken nokta, eğer $w_i=0$ ise robot düz bir hattı takip eder. Şayet $w_i \neq 0$ ise robotun izlediği yol merkez koordinatları eşitlik (A.13) ve (A.14)'de verilen bir daireyi tanımlar.

$$M_x^i = -(v_i/w_i) \cdot \sin \theta(t_i) \quad (\text{A.13})$$

$$M_y^i = -(v_i/w_i) \cdot \cos \theta(t_i) \quad (\text{A.14})$$

Bu durumda merkez noktası tanımlanan daireyi ifade eden denklem ise (A.15)'deki gibi yazılabilir.

$$(F_x^i - M_x^i)^2 + (F_y^i - M_y^i)^2 = (v_i/w_i)^2 \quad (\text{A.15})$$

Eşitlik (A.15) deki ifade robot yolunun i nci parçasının (M_x^i, M_y^i) merkezli $M_r^i = (v_i/w_i)$ yarıçaplı bir M_i dairesi olduğunu gösterir. Böylece hızların parçalı değişmez olduğu varsayımından yola çıkılarak bir robotun izleyebileceği yolun dairesel ve doğrusal yay parçalarından oluşmuş bir diziye yakınsayacağı ortaya çıkarılmıştır.

Başlangıç koşullarının olmadığı durumlarda (A.7) ile (A.12) arasında yer alan eşitlikler sadece hızlara bağımlı olacaktır.

ÖZGEÇMİŞ

Levent YENİLMEZ 1962 yılında İstanbul, Kartal'da doğdu. İlk öğrenimini Eskişehir'de, orta öğrenimini İstanbul'da tamamladı. 1979 yılında Orta Doğu teknik Üniversitesi-ODTÜ, Elektrik ve Elektronik Mühendisliği Bölümü'ne girdi. 1984 yılında mezun olduktan sonra Hava Kuvvetlerine Komutanlığına muvazzaf subay olarak girdi. Hava Kuvvetleri Komutanlığı'nda 1990 yılına kadar F-104 Uçuş Simülatörleri'nde çalıştıktan sonra yine ODTÜ, Elektrik ve Elektronik Mühendisliği Bölümü'nde Yüksek Lisans çalışmalarına başladı. 1993 yılında mezun olduktan sonra Hava Harp Okulu Elektronik Mühendisliği Bölümü'nde öğretim elemanı olarak göreve atandı. 1994 yılında İstanbul Teknik Üniversitesi-İTÜ, Elektrik-Elektronik Fakültesi, Kontrol ve Bilgisayar Mühendisliği Bölümü'nde Doktora çalışmalarına başladı.