

**ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

DOKTORA TEZİ

**EŞZAMANLI KONUM BELİRLEME VE HARİTA OLUŞTURMA
PROBLEMİNİN ÇÖZÜMÜNDE KULLANILAN YÖNTEMLERİN
İYİLEŞTİRİLMESİ**

Haydar ANKIŞHAN

ELEKTRİK - ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**ANKARA
2015**

Her hakkı saklıdır

TEZ ONAYI

Haydar ANKIŞHAN tarafından hazırlanan “**Eşzamanlı Konum Belirleme ve Harita Oluşturma Probleminin Çözümünde Kullanılan Yöntemlerin İyileştirilmesi**” adlı tez çalışması 14 Eylül 2015 tarihinde aşağıdaki jüri tarafından oy çokluğu ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Elektrik - Elektronik Mühendisliği Anabilim Dalı’nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Danışman: Yard. Doç. Dr. Fikret ARI
Ankara Üniversitesi, Elektrik - Elektronik Mühendisliği Anabilim Dalı

Jüri Üyeleri :

Başkan : Doç. Dr. Suat ÖZDEMİR
Gazi Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı

Üye : Doç. Dr. Hamit ERDEM
Başkent Üniversitesi, Elektrik-Elektronik Mühendisliği Anabilim Dalı

Üye : Doç. Dr. Murat EFE
Ankara Üniversitesi, Elektrik-Elektronik Mühendisliği Anabilim Dalı

Üye : Yard. Doç. Dr. Ahmet AKBULUT
Ankara Üniversitesi, Elektrik-Elektronik Mühendisliği Anabilim Dalı

Üye : Yard. Doç. Dr. Fikret ARI
Ankara Üniversitesi, Elektrik-Elektronik Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım.

Prof. Dr. İbrahim DEMİR
Enstitü Müdürü

ETİK

Ankara Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

14.09.2015

Haydar ANKIŞHAN

ÖZET

Doktora Tezi

EŞZAMANLI KONUM BELİRLEME VE HARİTA OLUŞTURMA PROBLEMİNİN ÇÖZÜMÜNDE KULLANILAN YÖNTEMLERİN İYİLEŞTİRİLMESİ

Haydar ANKIŞHAN

Ankara Üniversitesi
Fen Bilimleri Enstitüsü

Elektrik - Elektronik Mühendisliği Anabilim Dalı

Danışman: Yard. Doç. Dr. Fikret ARI

Eşzamanlı konum belirleme ve harita oluşturma (Simultaneous Localization and Mapping – SLAM) 90'lı yılların başlarında ortaya atılmış, bilinmeyen pozisyon ve çevre koşullarında robot/özerk aracın pozisyonunun hesaplanması ve etrafının haritasının oluşturulması için kullanılan bir yöntem olarak bilinmektedir. SLAM uygulamalarında, ölçüm gürültüsü minimizasyonu, etrafın haritası oluşturulurken karşılaşılan veri ilişkilendirme zorluğu, harita boyutu ve kullanılan yöntemin karmaşıklığına bağlı işlem yükü gibi temel problemler mevcuttur.

Bu tez çalışmasında, SLAM uygulamalarında genel olarak karşılaşılan bu temel problemlerin çözümü için, FastSLAM tabanlı iyileştirilmiş bir yöntem önerilmiştir. Literatürde var olan yaklaşımlardan farklı olarak, konvansiyonel FastSLAM yapısına adaptif merkez fark Kalman süzgeci A-(CDKF) ile birlikte birleşik olasılıksal veri ilişkilendirme (JPDA) algoritması uyarlanmıştır. Böylece, önerilen yeni FastSLAM yapısı sayesinde daha az sayıda parçacık kullanılarak, hem veri ilişkilendirme probleminin çözümü sunulmuş, hem de daha az işlem yüküyle kestirim doğruluğu yüksek sonuçlar elde edilmiştir. Bunun yanı sıra elde edilen sonuçlar, yine tez çalışması kapsamında geliştirilen doğadan ilham alan optimizasyon (DIAO) algoritmalarından Parçacık Sürü Optimizasyonu (PSO) ve farksal evrim (DE) ile iyileştirilen FastSLAM yöntemleriyle kıyaslanmıştır. Deneysel çalışmalar önerilen yöntemin, hem gerçek zamanlı uygulanabilmesi sayesinde DIAO tabanlı FastSLAM yöntemlerine üstünlük sağladığını hem de U-FastSLAM ve FastSLAM II yöntemlerinden az sayıda parçacık kullanılarak işlem yükünü azalttığını ve gürbüz sonuçlar elde edildiğini göstermiştir.

Eylül 2015, 149 sayfa

Anahtar Kelimeler: Eşzamanlı konum belirleme ve harita oluşturma, Kalman süzgeci, veri ilişkilendirme, birleşik olasılıksal veri ilişkilendirme, parçacık sürü optimizasyonu, farksal evrim.

ABSTRACT

Ph. D. Thesis

IMPROVING THE METHODS USED IN SIMULTANEOUS LOCALIZATION AND MAPPING

Haydar ANKIŞHAN

Ankara University
Graduate School of Natural and Applied Science
Department of Electrics and Electronics Engineering

Supervisor: Assist. Prof. Dr. Fikret ARI

Simultaneous Localization and Mapping (SLAM), was introduced in the early 90s, is known as a method for the building the map of the environment and simultaneously estimating the robot/ autonomous vehicle path in the unknown position information and unknown environment conditions. SLAM applications have some basic problems such as; measurement noise minimization, complexity of the data association in case of building map of the environment, building the large environment map and method complexity depending on the real time processing.

In this thesis, to solve these basic problems encountered in general practice SLAM, an improved FastSLAM based method has been proposed. Unlike the approaches that exist in the literature, adaptive central difference Kalman filter A-(CDKF) and joint probabilistic data association (JPDA) algorithms have been implemented in conventional FastSLAM structure. So, thanks to new structure of the FastSLAM, both presented as a solution for the problem of SLAM data association and obtained high accuracy results with less processing time. As well as the obtained results have been compared with the particle swarm optimization based and differential evolution based fastSLAM approaches which has been again improved in thesis studies. Experimental results have shown that the proposed method has property of the real time implementation, both has provided superior than optimize based FastSLAM approaches and although having less particle, minimizing the processing time and obtaining robust results than FastSLAM II and U-FastSLAM.

September 2015, 149 pages

Key Words: Simultaneous Localization and mapping, Kalman filter, data association, joint probabilistic data association, particle swarm optimization, differential evolution.

TEŐEKKÜR

Doktora alıŐmalarımı beraber sŸrdŸrebilme fırsatını bana tanıyan, araŐtırmalarımı yapabilmem iin her zaman destek olan, tezimin her aŐamasına tecrŸbeleri ve bilgisi ile yŸn veren danıŐmanım Yard. Do. Dr. Fikret ARI'ya (Ankara Ÿniversitesi Elektrik Elektronik MŸhendisliĐi Anabilim Dalı),

Tez alıŐmalarım sŸresince yapılan Tez İzleme Komitesi (TİK) toplantılarına katılarak, fikirleriyle alıŐmamın olgunlaŐmasını saĐlayan ve tecrŸbeleriyle gerekli motivasyonu veren TİK hocalarıma,

Tez alıŐmalarım sŸresinde beni yalnız bırakmayan manevi desteĐini eksik etmeyen aileme,

TeŐekkŸrlerimi sunarım.

Haydar ANKIŐHAN
Ankara, EylŸl 2015

İÇİNDEKİLER

TEZ ONAY SAYFASI	
ETİK	i
ÖZET	ii
ABSTRACT.....	ii
TEŞEKKÜR	iv
KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	ix
ÇİZELGELER DİZİNİ	xii
1. GİRİŞ	1
2. KURAMSAL TEMELLER	10
2.1 SLAM için Kullanılan Temel İstatistiksel Yöntemler	14
2.1.1 Kalman tabanlı süzgeçler	14
2.1.1.1 Kalman süzgeci	15
2.1.1.2 Genişletilmiş Kalman süzgeci	20
2.1.1.3 Kokusuz dönüşüm ve kokusuz Kalman süzgeci algoritması	22
2.1.1.4 Merkez fark Kalman süzgeci	26
2.1.2 Beklenti enbüyültme tabanlı yöntemler.....	29
2.1.3 Sıralı Monte Carlo yaklaşımları.....	31
2.1.3.1 Parçacık süzgeci tabanlı FastSLAM yaklaşımları	32
2.1.3.1.1 FastSLAM algoritması	33
2.2 SLAM Probleminde Veri İlişkilendirme	41
2.2.1 FastSLAM için JPDA	43
3. MATERYAL ve YÖNTEMLER	49
3.1 FastSLAM Metotlarında Optimizasyon Algoritmaları	50
3.1.1 FastSLAM için parçacık sürü optimizasyonu	51
3.1.1.1 PSOPC tabanlı parçacık süzgeçleri.....	51
3.1.2 FastSLAM için farksal evrim algoritmaları	52
3.1.3 JPDA uyarlı merkez fark Kalman süzgeci tabanlı FastSLAM algoritması	54
3.1.3.1 Robot/özerk araç durum hesabı	55
3.1.3.2 Nesne/öznitelik durum hesabı.....	58
3.1.3.3 Nesne başlangıç işlemleri.....	60
3.1.3.4 Önem ağırlık hesabı ve tekrar örnekleme	61
3.1.4 Q ve R matrislerinin CDKF-FastSLAM için uyarlanması.....	61
3.1.4.1 Ölçüm gürültüsünün uyarlanması	62

3.1.4.2 Süreç gürültüsünün uyarlanması	63
4. BULGULAR.....	64
4.1 Gözlem Gürültüsü Minimizasyonu	64
4.1.1 Karakök kokusuz ve karakök merkez fark Kalman süzgeçleri	64
4.1.2 SLAM'de uyarlı bulanık mantık çıkarımı destekli Kalman süzgeci.....	67
4.1.3 karakök kokusuz Kalman süzgeci tabanlı FastSLAM	70
4.2 SLAM Problemi için Veri İlişkilendirme.....	74
4.3 Optimizasyon Yöntemlerine Dayalı FastSLAM Yaklaşımları.....	84
4.3.1 Karakök kokusuz FastSLAM yöntemi ile parçacık sürü optimizasyonu pasif toplama.....	84
4.3.2 Farksal evrim tabanlı CDKF-FastSLAM yaklaşımları.....	89
4.4 Uyarlı FastSLAM Yaklaşımı.....	104
5. TARTIŞMA VE SONUÇ	118
KAYNAKLAR	122
EKLER	128
EK 1 Mahalanobis Mesafesi (Mahalanobis Distance, MD).....	129
EK 2 Matris Teorisi	132
EK 3 Merkez Fark Kalman Süzgeci.....	134
EK 4 Ayarlanabilir Sinirsel Bulanık Mantık Çıkarımı Sistemi (ANFIS).....	143
EK 5 SLAM ve FastSLAM Akış Diyagramları.....	145
ÖZGEÇMİŞ.....	147

KISALTMALAR DİZİNİ

ADG	Ackermann Direksiyon Geometrisi
ANFIS	Adaptif Network Bulanık Mantık Çıkarımı Sistemi
CD	Parazit Yankı Önceliği
CDKF	Merkez Fark Kalman Süzgeci
CDKF-FastSLAM	Merkez Fark Kalman Süzgeci FastSLAM
DE	Farksal Evrim
DE-SR-CDKF-FastSLAM	Farksal Evrim Karekök Merkez Fark Kalman Süzgeci FastSLAM
DFS	Derin İlk Arama
ECC	Edmonton Convention Center
EKF	Genişletilmiş Kalman Süzgeci
EM	Beklenti Enbüyültme
FF	Uygunluk Fonksiyonu
FOV	Görüş alanı
GA	Genetik Algoritma
GPS	Global Konumlama Sistemi
JPDA	Birleşik Olasılıksal Veri İlişkilendirme
JPDA-UKF	Birleşik Olasılıksal Veri İlişkilendirme- Kokusuz Kalman Süzgeci
JPDA-CDKF-FastSLAM	Birleşik Olasılıksal Veri İlişkilendirme Tabanlı Merkez Fark Kalman süzgeci FastSLAM
KF	Kalman Süzgeci
KKS	Küresel Konumlama Sistemi
MHT	Çoklu Hipotez izleyici
MSE	Ortalama Kare Hatası
NIS	Normalleştirilmiş İnovasyon Karesi
NN-EKF	En Yakın Komşuluk İlişkili Genişletilmiş Kalman Süzgeci
NN-UKF	En Yakın Komşuluk İlişkili Kokusuz Kalman Süzgeci
PD	Algılama Olasılığı

PDA	Olasılıksal Veri İlişkilendirme
PF	Yanlı Algılama Olasılığı
PSO	Parçacık Sürü Optimizasyonu
PSOPC	Parçacık Sürü Optimizasyonu Pasif Toplama
PSOPC-CDKF-FastSLAM	Parçacık Sürü Optimizasyonu Pasif Toplama Tabanlı Merkez Fark Kalman Süzgeci FastSLAM
RBPF	RaoBlacwellized Parçacık Süzgeci
RTS	Runch Tung Striebel
SLAM	Eşzamanlı Konum Belirleme ve Harita Oluşturma
SMC	Sıralı Monte Carlo
SNR	Sinyal Gürültü Oranı
SPI	Stirling's Polinomsal İnterpolasyon
SR-CDKF	Karekök Merkez Fark Kalman Süzgeci
SR-CDKF-Aid-FastSLAM	Karekök Merkez Fark Kalman Süzgeci Yardımlı FastSLAM
SR-CDKF-FastSLAM	Karekök Merkez Fark Kalman Süzgeci FastSLAM
SR-CDKF-PF-FastSLAM	Karekök Merkez Fark Kalman Süzgeci-Parçacık Süzgeci Tabanlı FastSLAM
SR-UKF	Karekök Kokusuz Kalman Süzgeci
SRU-FastSLAM	Karekök Kokusuz FastSLAM
SRUPF-FastSLAM	Karekök Kokusuz Parçacık Süzgeci FastSLAM
SRU-PSOPC-FastSLAM	Karekök Parçacık Sürü Optimizasyonu Pasif Toplama FastSLAM
SUT	Ölçeklenebilir Kokusuz Dönüşüm
TP	Geçici Parçacık
U-FastSLAM	Kokusuz FastSLAM
U-Aid-FastSLAM	Kokusuz Yardımlı FastSLAM
UKF	Kokusuz Kalman Süzgeci
UPF-FastSLAM	Kokusuz Parçacık Süzgeci FastSLAM
UT	Kokusuz Dönüşüm
VG	Geçerlilik Kapısı

ŞEKİLLER DİZİNİ

Şekil 1.1.a İç mekan uygulaması,b. SLAM, Robot ve odometrik veriden elde edilen iç mekan haritası.....	1
Şekil 1.2 SLAM için hedef çevre uygulamaları.....	3
Şekil 1.3 Robot pozisyonundaki kestirim belirsizliğinin artması (Montemerlo (2003)' ten yararlanılmıştır).....	4
Şekil 1.4 İstatistiki olarak bağımlı ölçüm hatasının haritalama sonucuna etkisi	5
Şekil 1.5 Gerçek zamanlı nesne sayısı yüksek olan SLAM uygulaması (Bosse 2007)	6
Şekil 1.6 Nesne işaretlerinin yakın olması sonucu ortaya çıkan ölçüm belirsizliği durumu (Montemerlo (2003)'ten yararlanılmıştır).....	7
Şekil 1.7 Pozisyon belirsizliği sonucunda veri belirsizliği (Montemerlo (2003)'ten yararlanılmıştır).....	8
Şekil 2.1 SLAM yönteminde kestirim işlemi blok diyagramı	10
Şekil 2.2 Haritası oluşturulan ve robot/özerk araç pozisyonu kestirilen temsili bir görüntü parçası	11
Şekil 2.3 Robotun/özerk aracın koordinat sistemiyle tanımlanan kinematik model gösterimi...	12
Şekil 2.4 Kalman süzgeci Tahmin-Güncelleme döngüsü	17
Şekil 2.5 Dinamik bir model için Kalman süzgecinin verilen bilgiler dahilinde kestirilen blok diyagramı.....	18
Şekil 2.6 Doğrusal Kalman süzgeci blok diyagramı	19
Şekil 2.7 EKF algoritması işlem blok diyagramı	21
Şekil 2.8 İki boyutlu uzayda ortalama ve kovaryans değerlerinin EKF ve UKF için temsili benzetim sonuçları	24
Şekil 2.9 UKF algoritması blok diyagramı	25
Şekil 2.10 CDKF algoritması blok diyagramı	28
Şekil 2.11 Grid tabanlı haritalama yaklaşımıyla EM yönteminde a. elde edilen b. gerçek harita (Thrun vd. 2000)	30
Şekil 2.12 Robot pozisyonu ve parçacıklara bağlı Parçacık süzgeci modeli gösterimi	33
Şekil 2.13 Olasılık hareket modelinden örneklerin olasılık alanında dağıtılmasına örnek bir gösterim.....	34
Şekil 2.14 Amaçlanan dağılımdan olasılığa bağlı olarak ağırlıklandırılmış örneklerinin temsili gösterimi.....	39
Şekil 4.1 SR-CDKF'nin WebMap (Bailey 2003) haritası için kestirim sonuçları ($\sigma_w = 0,1$ m/s ve $\sigma_s = 1^\circ$).	65

Şekil 4.2 Süzgeçlerin Ortalama Kare Hatası Sonuçları:	
Ukf: Kokusuz Kalman süzgeci, Fast II: FastSLAM II, Ekf: Genişletilmiş Kalman süzgeci, SrCdkf: Karakök merkez fark Kalman süzgeci ve SrUkf: karakök kokusuz Kalman süzgeci' ni temsil etmektedir.....	66
Şekil 4.3 ANFIS destekli SR-CDKF kestirim sonucu: gerçek rota (yeşil) ve kestirilen rota (siyah).....	68
Şekil 4.4 Önerilen senaryo için Metotların kestirim hata sonuçları.....	69
Şekil 4.5.a. U-FastSLAM, b. SRU-FastSLAM kestirim sonuçları	71
Şekil 4.6 Süzgeçlerin ilgili senaryo için kestirim ortalama kare hatası sonuçları:	
a. pozisyon hatası, b. başlık açısı hatası	72
Şekil 4.7 JPDA -UKF algoritmasının ilgili senaryo için kestirim sonuçları.....	75
Şekil 4.8 Süzgeçlerin robot pozisyonu ve başlık açısı ortalama kare hata kestirim sonuçları	76
Şekil 4.9 Bütün algılanan nesne işaretleri için nesne işaretleri konum kestiriminin bütün süzgeçler için standart sapma değerleri:	
a. X Koordinatı, b. Y-koordinatı.....	78
Şekil 4.10 JPDA -UKF algoritması statik parazit yankılı ortamda SLAM problemi için kestirim sonuçları.....	80
Şekil 4.11 Süzgeçler robot pozisyon ve başlık açısı ortalama kare hata kestirim sonuçları.....	81
Şekil 4.12 Bütün algılanan nesne işaretleri için nesne işaretleri konum kestiriminin bütün süzgeçler için standart sapma değerleri, Y-koordinatı:	
a. NN-EKF, b. NN-UKF, c. FastSLAM II, ve d. JPDA-UKF (Y-Koordinatı için) ..	82
Şekil 4.13 Hesaplanan ve gerçek araç rotası ile hesaplanan ve gerçek nesnelere:	
a. SRU - Kalman süzgeci tabanlı PSOPC optimizasyonlu (SRU-PSOPC)-FastSLAM, b. SRU-FastSLAM.....	86
Şekil 4.14 Araç pozisyonunun X-Y koordinatına bağlı hata kareleri ortalaması sonuçları.....	86
Şekil 4.15.a. U-FastSLAM, b. SRU-FastSLAM ve c. SRU-PSOPC-FastSLAM için kestirim sonuçları.....	88
Şekil 4.16 Hesaplanan ve gerçek araç rotaları ve nesne pozisyonları:	
a. FastSLAM II, b. SR-CDKF-Aid-FastSLAM, c. SR-CDKF-PF-FastSLAM ve d. SR-CDKF-FastSLAM.....	91
Şekil 4.17 Yöntemler için araç pozisyon ve başlık açısı ortalama kare hataları (MSE):	
a. pozisyon hatası, b. başlık açısı hatası (Radyan), c. kısmi başlık açısı hatası (Radyan).....	92
Şekil 4.18 Şekil 4.18 Yöntemlerin Kutu-Plot çizimleri:	
1. FastSLAM II, 2. SRCD-Aid-FastSLAM, 3. SRCD-PF-FastSLAM, 4. SRCD-FastSLAM.....	94

Şekil 4.19 Yöntemler için kestirilen, gerçek rota bilgisi ve nesne pozisyonları bilgisi:	
a. U-FastSLAM, b. SRCD-FastSLAM, c. DE-SRCD-FastSLAM.	95
Şekil 4.20 Metotların MSE sonuçları:	
a. U-FastSLAM, SRCD-FastSLAM ve DE-SRCD-FastSLAM pozisyon hataları,	
b. başlık açısı hataları.....	96
Şekil 4.21.a. FastSLAM II, b. U-FastSLAM ve c. DE-SRCD-FastSLAM kestirim sonuçları...	99
Şekil 4.22 Metotların geçerlilik grafikleri:	
a. FastSLAM II, 100 parçacık, b. U-FastSLAM, 100 parçacık ve c. DE-SRCD-	
FastSLAM, 10 parçacık. MCR Monte Carlo Simulasyonu kısaltmasıdır.....	100
Şekil 4.23 Victoria Park, Sydney, Australia, GPD verisiyle birlikte gösterilmiştir.....	101
Şekil 4.24 Victoria Park, Sydney, Australia, SRCD-FastSLAM hesaplanan araç rotası.....	102
Şekil 4.25.a FastSLAM II, b. U-FastSLAM, ve c. SRCD-FastSLAM Victoria park veri seti	
kestirim sonuçları.....	103
Şekil 4.26 İç mekan uygulaması:	
a. gerçek mimari, b. mimarinin simülasyon çalışması, izlenecek rota.....	105
Şekil 4.27 SLAM iç mekan uygulaması:	
a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-	
FastSLAM, d. U-FastSLAM ve e. FastSLAM-II.....	106
Şekil 4.28 SLAM iç mekan uygulaması pozisyon ve başlık ortalama kare hata sonuçları.....	107
Şekil 4.29 SLAM iç mekan uygulaması, yöntemlerin Kutu-Plot çizimleri.....	107
Şekil 4.30 Yoğun gürültülü, nesne sayılarının fazla ve birbirine yakın olduğu benzetim	
ortamı.....	110
Şekil 4.31 SLAM iç mekân uygulaması yoğun gürültülü ve nesne sayılı ortam:	
a. FastSLAM II, b. U-FastSLAM, c. PSOPC-CDKF-FastSLAM, d. DE-CDKF-	
FastSLAM ve e. JPDA-CDKF-FastSLAM.....	111
Şekil 4.32 JPDA-CDKF-FastSLAM yöntemi kestirim sonucu.....	112
Şekil 4.33 Önerilen senaryo için metotların pozisyon ve başlık açısı hataları:	
a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-	
FastSLAM, d. U-FastSLAM.....	113
Şekil 4.34 Metotların ilgili senaryo için ortalama pozisyon ve başlık açısı hata grafikleri.....	115
Şekil 4.35 Metotların Normalize edilmiş kestirim hata kareleri:	
a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-	
FastSLAM, d. U-FastSLAM.....	116

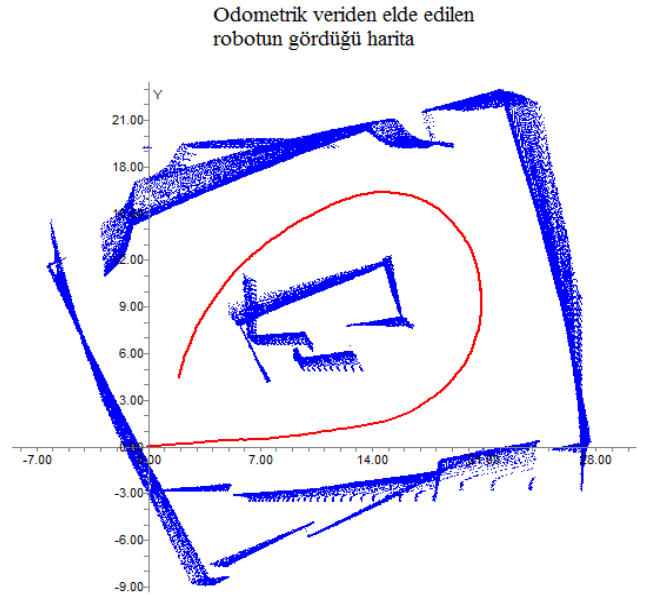
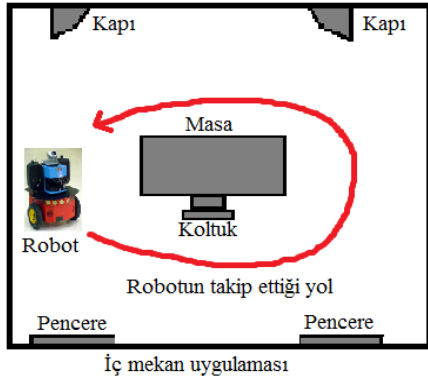
ÇİZELGELER DİZİNİ

Çizelge 4.1 Süzgeçlerin Ortalama Kare hataları ve işlem süreleri.....	67
Çizelge 4.2 Süzgeçlerin Ortalama Kare hataları ve İşlem Süreleri	69
Çizelge 4.3 Süzgeçlerin konum ve başlık açısı ortalama kare hataları ve varyans değerleri.....	73
Çizelge 4.4 Robot başlık açısı ve konum ortalama kare hataları.....	79
Çizelge 4.5 Parazit yankılı ortamda süzgeç hata kareleri ortalamaları.....	83
Çizelge 4.6 PSOPC için başlangıç parametreleri.....	85
Çizelge 4.7 Metotların ilgili senaryo için çıktıları.....	87
Çizelge 4.8 Önerilen SLAM algoritmalarının performansları.....	89
Çizelge 4.9 Metotların ortalama kare hataları.....	93
Çizelge 4.10 Metotların ortalama kare hataları.....	96
Çizelge 4.11 İç mekan uygulaması yöntemlerin ortalama kare hataları.....	109

1. GİRİŞ

Harita oluşturma, robotların/özerk araçların, fiziksel çevrelerini x-y düzlemindeki iz düşümlerine göre gösterme problemi olarak bilinir, robotlar/özerk araçlar ve yapay zekâ alanlarında özellikle son yirmi yıldır aktif olan bir araştırma konusudur. Çevre haritasının oluşturulması, günümüzde kullanılan robotlar/özerk araçlar için önemli problemlerden birisidir.

Literatürde harita oluşturma problemi, robot pozisyonu belirleme problemi ile birlikte ele alınır. Çünkü etraftaki nesnelerin konumunu bilmek için başka bir deyişle harita oluşturabilmek için robotun/özerk aracın kendi konumunu ve etrafındaki nesnelerin konumunu bilmesi gereklidir; robotun kendi konumunu belirleyebilmesi için de gezindiği ortamda etrafında bulunan nesnelerin konumunu bilmelidir. Şekil 1.1.a,b, gezinen bir robotun odometrik veriden elde edilen haritasını göstermektedir.



Şekil 1.1.a. İç mekan uygulaması, b. SLAM, Robot ve odometrik veriden elde edilen iç mekan haritası

Robot, haritasını oluşturmayı amaçladığı ortamda gezinirken dış dünyayı algılayabilmelidir. Bunun için de kamera, laser veya kızılötesi ışın teknolojilerini kullanan mesafe ölçerler, pusulalar veya küresel konumlama sistemleri (KKS) kullanabilirler. Fakat algılayıcıların verisi tamamen güvenli olmamakta, ölçümler gürültülü olabilmekte veya çoğu algılayıcının görme – algılama mesafesi kısıtlı olabilmektedir. Algılayıcıların bu sınırlandırılmalarından ötürü robot sabit kalarak bulunduğu ortamın haritasını çıkaramaz, bu ortamda gezinmek durumundadır. Haritası çıkarılacak ortamda gezinirken üretilen hareket (kontrol) komutları, farklı algılayıcı ölçümlerinin alındığı konumlarla ilgili bilgi içerdikleri için haritalamada önemlidir.

Robotun/özerk aracın kendi konumunu bilmesi ve aynı zamanda etrafın haritasını çıkarması bir problemdir ve literatürde Eşzamanlı konum belirleme ve harita oluşturma (SLAM) olarak bilinir. SLAM problemi, ilk defa 1980'lerin ortalarında geliştirilmiş bir yöntemdir. SLAM kullanılmaya başlandığı ilk yıllarda temel olarak metrik ve topolojik yöntemler olarak ikiye ayrılmıştır. Metrik haritalar çevrenin geometrik özelliklerini, topolojik haritalar ise farklı bölgelerin birbirine bağlı olup olmadığını gösterir. Chatila ve Laumond (1985) çevrenin geometrisini tanımlamak amacıyla bir dizi çokgen kullanarak metrik haritalama yapmışlardır. Mataric (1990) ile Kuipers ve Byun (1991)'da topolojik haritalama yöntemlerini kullanmışlardır.

SLAM algoritmalarının tarihsel olarak gelişim sürecine bakıldığında dünya-merkezli ve robot-merkezli olmak üzere iki sınıflandırma yapıldığı görülmektedir. Dünya merkezli haritalama modelinde dünya çapında bir koordinat düzleminde harita gösterimi yapılmaktadır. Robot merkezli harita oluşturmada ise robotun başlangıç konumu sıfır kabul edilerek etrafın haritası bu konuma bağlı olarak oluşturulur. İlk bakıldığında robot-merkezli haritalama daha kolaymış gibi görünür. Fakat daha önce gezilmemiş yerleri tespit etmek ve birbirine benzeyen farklı pozisyonları ayırt etmek robot-merkezli haritalama için zordur. Bu sebeplerden ötürü genellikle dünya-merkezli haritalama SLAM uygulamalarında tercih edilmektedir.

Olasılıksal yöntemler 1990'lı yılların başlangıcından itibaren harita oluşturma ve SLAM uygulamalarında daha sık kullanılmaya başlanmıştır. Harita ve robot pozisyonu

kestiriminde istatistiksel yöntemlerden en sık kullanılanı Kalman tabanlı kestiricilerdir. Kalman süzgeçlerine alternatif olarak kullanılan diğer olasılıksal yöntemler ise Dempster'in (1977) geliştirdiği beklenti enbüyültme (expectation maximization, BE) ve parçacık süzgeçleri olarak verilebilir. Günümüzde bu algoritmalarla SLAM problemine çözüm bulmaya çalışan birçok yaklaşım vardır (Norgaard vd. 2000, Thrun vd. 1998, Bailey vd. 2003, Kim vd. 2008, Montemerlo 2002, Montemerlo vd. 2003). Robotun etrafını keşfetmesi ve haritasını oluşturması için günümüzde üretilen çözümler çoğunlukla maliyeti yüksek sistemler ve aparatlarına ihtiyaç duyarlar. SLAM yöntemleri buna rağmen her geçen gün uygulama alanını genişletmektedir. İç mekân (Mark vd.2013, Leonard vd. 2001) , dış mekân (Bailey 2002, Montemerlo 2002, 2003, Dissanayke vd. 2001, Guivant vd. 2000, Wang vd. 2003), sualtı (Williams vd. 2001), hava teknoloji uygulamaları (Shi vd. 2013, Schleicher vd. 2009) gibi alanlarda kullanılmaktadır. Şekil 1.2 bazı SLAM çevrelerini göstermektedir.



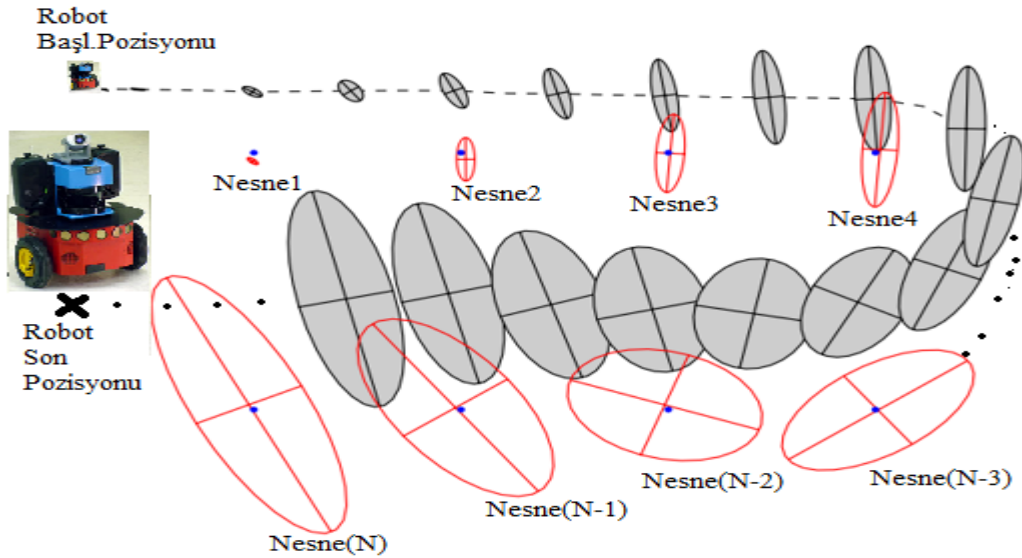
Şekil 1.2 SLAM için hedef çevre uygulamaları

SLAM modelinde robot/özerk araç algılayıcılardan ve kontrol girişlerinden gelen bilgilere göre hareket ederken ölçüm gürültüsü minimizasyonu, veri ilişkilendirme, işlem yükü gibi problemlerle karşılaşabilir, dolayısıyla aracın yeni pozisyonunu belirlerken sadece kontrol işaretlerine güvenmek doğru değildir. Örneğin ölçüm gürültüsü her bir zaman adımında, istatistiki olarak birbirine bağımlı olduğundan, zamanla hatanın büyümesi ve robotun/özerk aracın haritasının gezinim boyunca nesne sayısının artması sonucu haritanın hatalı olarak çıkarılmasına neden olmaktadır. Diğer bir problem ise çevre haritasının oluşturulması esnasında karşılaşılan çoklu nesne takibine bağlı olarak algılayıcılardan alınan bilgilerin önceki verilerle ilişkilendirilmesi problemidir. Nesne sayısının artması ve birbirine yakın nesnelerin model kestirim

belirsizliğini artırması haritanın yanlış oluşmasına yol açar. Bunların yanı sıra harita boyutunun büyümesi (pratik uygulamalarda nesne sayıları 100'lerle ifade edilirken, gerçek zamanlı uygulamalarda on binlere kadar çıkabilmektedir), nesne şekillerinin simetrik bir yapıya sahip olmaması ve dinamik nesnelere belirli SLAM problemleri arasında yer alır.

SLAM uygulamalarında ölçüm gürültüleri istatistiksel olarak daha önceki nesne işaretleri ve araç pozisyonuna bağlı olduğundan her bir adımda tahmin hataları sonucu büyümekte ve böylece oluşturulacak haritanın yanlış elde edilmesine yol açmaktadır. Daha önceki çalışmalarda bu problemin çözümüne yönelik olarak birçok farklı yaklaşım önerilmiştir (Bailey 2006, Montemerlo 2003, Thrun vd. 2001).

Şekil 1.3'te görüldüğü gibi başlangıçta küçük bir başlık açısı hatası olmasına rağmen bu durum ilerleyen adımlarla artarak haritası oluşturulacak mekânın yanlış haritasının elde edilmesine yol açmaktadır. Bu durum robot pozisyonuna da yansımaktadır. Böylece tahmin ve kestirim aşamasında hem nesnelerin hem de robot pozisyonundaki belirsizlik şekil 1.3'te görüldüğü gibi artmaktadır.



Şekil 1.3 Robot pozisyonundaki kestirim belirsizliğinin artması sonucu nesnelerin Kestirim belirsizliği de aynı şekilde artmaktadır (Montemerlo (2003)'ten yararlanılmıştır.

Şekil 1.3 Robot pozisyonundaki kestirim belirsizliğinin artması sonucu nesnelerin kestirim belirsizliği de aynı şekilde artmaktadır (Montemerlo (2003)' ten yararlanılmıştır).

Şekil 1.3'te, gri renkli siyah sürekli çizgili elipsler robot pozisyonundaki belirsizliği, boş kırmızı renkli sürekli çizgili elipsler ise nesne pozisyonlarındaki belirsizlikleri göstermektedir. Artan belirsizlikler sonucunda elde edilen bir harita örneği şekil 1.4'te gösterilmektedir.



Şekil 1.4 İstatistikî olarak bağımlı ölçüm hatasının haritalama sonucuna etkisi

Sol taraftaki harita gerçek mimariyi, sağ taraftaki ise olması gereken (Mavi çizgili) ve elde edilen (Kırmızı çizgili) haritayı göstermektedir (Rückert 2009). Şekil 1.4'te sağ tarafta oluşturulan haritada, mavi patika gerçek yolu, kırmızı, ölçüm gürültüsü etkisinde kestirilen haritayı ve yeşil işaretler ise nesne bilgilerini vermektedir.

SLAM uygulamaları için diğer bir problem ise harita boyutunun büyük olmasıdır. Haritası çıkarılacak bölgenin nesne sayısı fazla ise bu SLAM için gerçek zamanlı

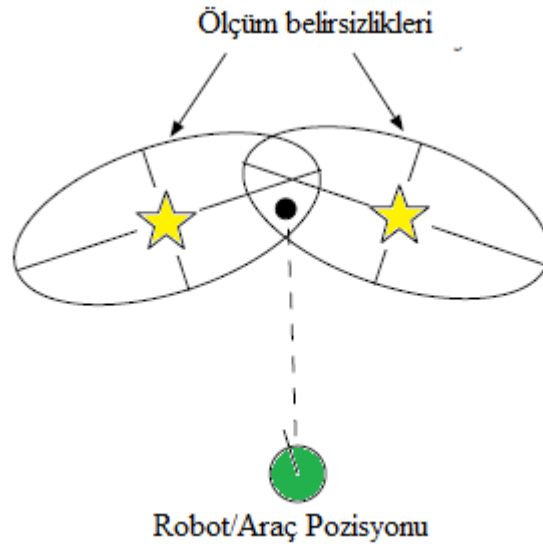
uygulamalarda bir problem teşkil etmektedir. Şekil 1.5, nesne sayısı fazla bir dış mekan haritasını göstermektedir.



Şekil 1.5 Gerçek zamanlı nesne sayısı yüksek olan SLAM uygulaması (Bosse 2007)

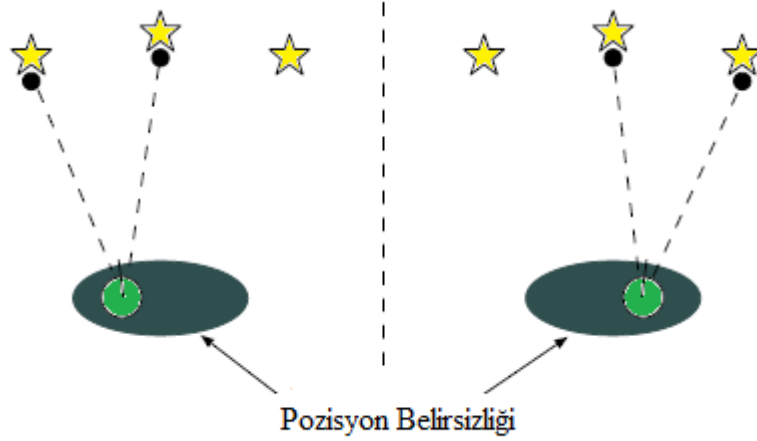
Simulasyon uygulamalarında SLAM için geliştirilen yöntemler genel olarak 100' lerle ifade edilen nesne işaretlerini göz önünde bulundurarak işlem yapmaktadırlar (Oysa gerçek zamanlı uygulamalarda bu rakam on binlere çıkmaktadır). SLAM modelinde durum vektörü nesne işaretlerine paralel büyümektedir. Robot/özerk araç pozisyonu ve nesne işaretlerinin (x,y) koordinat bilgisi toplamında durum vektörü zaman içerisinde harita oluşturulurken büyümektedir. Bu esnada sistem hata kovaryans matrisi ise N boyutlu durum vektörüne sahip SLAM modelinde $N \times N$ boyutlu olmaktadır. Gerçek zamanlı uygulamalarda N çok büyüdüğünden bütün önerilen yöntemler her zaman çözüm sağlayamamaktadır.

SLAM uygulamalarında diğer karşılaşılan bir problem ise veri ilişkilendirmedir. Model birden fazla hedef olduğunu kabul eder. Hedeflerden bir tanesi robotun/özerk aracın kendisi diğerleri ise algılayıcı gözlemleri tarafından tahmin edilen nesnelere. Nesne sayısının artması ve birbirine yakın nesnelerin model kestirim belirsizliğini artırması haritanın gerçek harita bilgisinden farklı olarak yanlış oluşmasına, robot/özerk araç pozisyonunun sapmasına yol açacaktır. Bunun yanı sıra robot/özerk araç geçtiği yerden bir daha geçtiğini hatırlayamayacaktır. Ölçüm belirsizliğinin artması veri ilişkilendirme zorluğunu da artırmaktadır. Çünkü ölçümlerdeki belirsizliğin artması daha büyük sayıda ölçüm verisi arasında ilişkilendirme yapılmasını zorunlu kılacaktır. Ölçüm belirsizliğine örnek şekil 1.6'da verilmiştir. Ölçüm belirsizliğinin artması veri belirsizliğinin de artmasını paralelinde getirecektir.



Şekil 1.6 (Montemerlo (2003)'ten yararlanılmıştır) Nesne işaretlerinin yakın olması sonucu ortaya çıkan ölçüm belirsizliği durumu

Sarı yıldızlar nesne işaretlerini, siyah büyük nokta gözlemi göstermektedir. Elipsler ise nesnelerin belirsizlik alanlarını gösterir. Pozisyon belirsizliği de benzer etkiye sahiptir fakat bu aşamada gözlemlenen hangi nesnenin gerçekte nerede olduğu bilinmemektedir. Şekil 1.7'de bu durum açıkça gösterilmektedir.



Şekil 1.7 (Montemerlo (2003)'ten yararlanılmıştır) Pozisyon belirsizliği sonucunda veri belirsizliği

Burada sarı yıldızlar nesne işaretlerini, koyu mavi elipsler robot/araç pozisyonu belirsizlik alanını, yeşil yuvarlaklar ilgili nesnelere ilişkilendirilmiş tahmini robot/araç pozisyonunu göstermektedir. Pozisyon kestirimindeki koyu elipsler robot pozisyonundaki belirsizliği göstermektedir. Buna bağlı olarak pozisyon ve başlık elipsin içerisinde her hangi bir yerde olabilir. Bu yüzden hangi nesnenin ölçümle ilişkilendirileceği karmaşık bir durumu ifade etmektedir.

Bahsedilen problemler göz önüne alınarak tez çalışması kapsamında iyileştirilmiş ve uyarlı bir FastSLAM modeli literatüre sunulmuştur. Özellikle FastSLAM'in işlem yükü, veri ilişkilendirmesi ve gürültü minimizasyonu gibi problemlerinin çözümü üzerinde durulmuştur. İşlem yükü problemi, parçacık sayısı seçimi ile orantılıdır, parçacık sayısı azaltıldığında yöntemin kestirim performansı azalmakta, artırıldığında ise işlem yükü artmaktadır. Çalışma kapsamında bu problemi çözmek için doğadan ilham alan optimizasyon yöntemlerinden yararlanılmıştır. Az sayıda parçacık seçilmiş ve bu parçacıklar olasılıkları yüksek bölgelere optimizasyon algoritmaları yardımıyla kaydırılarak bir anlamda parçacık dejenerasyonu probleminin önüne geçilmeye çalışılmıştır. Böylece FastSLAM'in işlem yükü azaltılırken kestirim doğruluğu da korunmuştur, fakat optimizasyon yöntemlerinin kendi iç iteratif yapısından kaynaklı gerçek zamanlı uygulamalar için bir problem oluşturduğu görülmüştür. Bu yaklaşımla birlikte tez çalışması kapsamında hem veri ilişkilendirmesi yapabilen, hem de kestirim

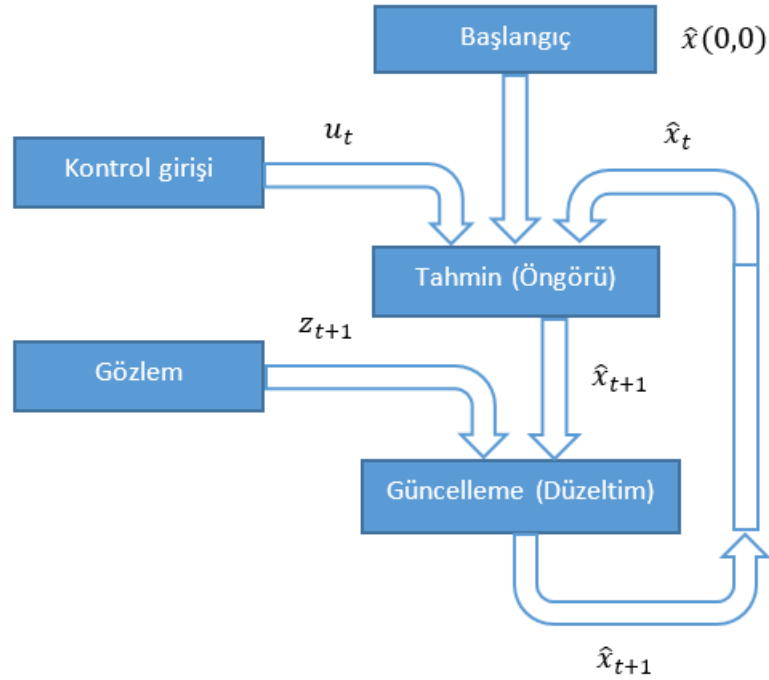
dođruluđu aısından optimizasyon yntemleri kullanan yaklařıma alternatif olabilecek uyarlı bir FastSLAM yaklařımı nerilmiřtir. Metot FastSLAM'in ierisinde kullanılan Kalman tabanlı szgecin sre ve gzlem grlt matrislerini uyarlı hale getirerek optimizasyon algoritmasından kaynaklanan i iteratif yapının iřlem ykn ortadan kaldırmıřtır. Sonular daha nceden literatre sunulan yntemlerle kıyaslanmıř ıktıları benzetim alıřmalarıyla gsterilmiřtir. nerilen metodun SLAM'de karřılařılan grlt minimizasyonu, veri iliřkilendirme, byk harita boyutu ve gerek zamanlı alıřma gibi problemlerin hepsine birden zm sunabildiđi gzlenmiřtir.

Bu tez alıřması řu řekilde devam etmektedir; ikinci blmde SLAM modeline bađlı olarak kuramsal temeller yer almıřtır. Kuramsal temeller SLAM, nc blmde tez kapsamında geliřtirilen materyal ve yntemler, drdnc blmde geliřtirilen yntem ve daha nceden kullanılan yntemlerin deneysel ıktıları ve bulgular, son blmde ise tartıřma ve sonu yer almaktadır.

2. KURAMSAL TEMELLER

SLAM uygulamaları, harita oluşturma ve robot/özerk araç pozisyonu belirleme gibi iki önemli işlemin eş zamanlı çalışması sonucu ortaya çıkmaktadır. Metot eşzamanlı olarak robot pozisyonunu tahmin eder ve etrafın haritasını oluşturmaya çalışır.

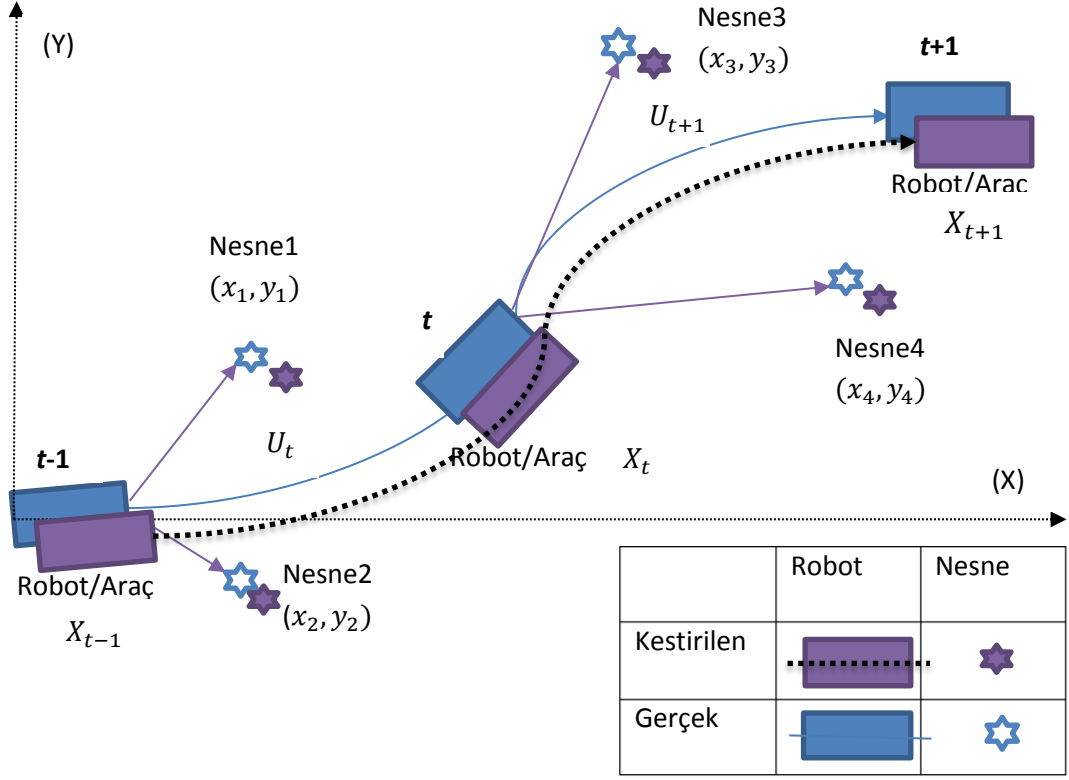
Robotlar/özerk araçlar için mutlak konum bilgisi mevcut değilse, bu araçlar uzun süreler dâhilinde küçük hatalarla gezinebilirler (Nebot vd. 2006). SLAM yönteminde, bilinmeyen bir çevrede, bir aracın durumu, literatürde tanımlanmış bilinen nesne yapılarıyla (Dissanayake vd. 2008, Marzorati vd. 2007, Paz vd. 2008), Ackermann direksiyon geometrisi (King-Hele 2002) kullanılarak şekil 2.1'deki gibi iki adımda gerçekleştirilir. Detaylı SLAM blok diyagramı EK 5'te verilmiştir.



Şekil 2.1 SLAM yönteminde kestirim işlemi blok diyagramı

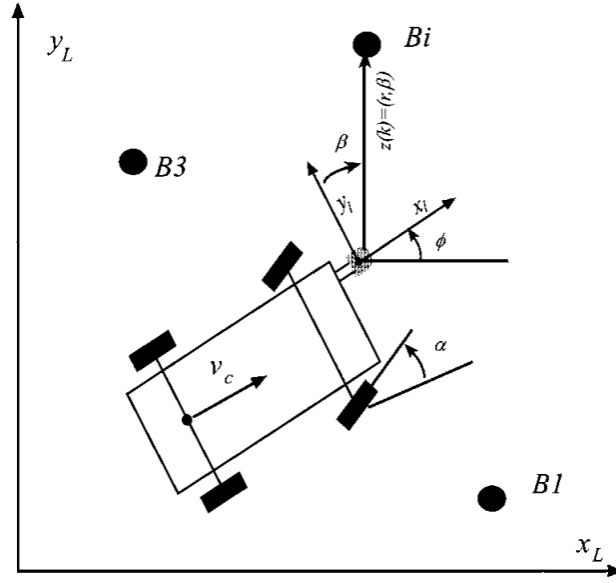
Robot/özerk araç, araca ait kontrol bilgileri (aracın hız ve başlık açısı bilgisi) ve algılayıcılar aracılığıyla topladığı gözlemleri kullanarak nesnelere tespit edebilir. Bu sayede robotun/özerk aracın pozisyonu şekil 2.1'deki blok diyagramı kullanılarak kestirilir ve çevre haritası oluşturulabilir. Şekil 2.2, nesnelere harita oluşturma ve

robot/araç pozisyon kestirimi işlemine örnek gösterilecek temsili bir görüntüyü göstermektedir.



Şekil 2.2 Haritası oluşturulan ve robot/araç pozisyonu kestirilen temsili bir görüntü parçası

Burada, (x_i, y_i) 'ler nesne pozisyon bilgilerini, X_t t anındaki robot/araç lokasyonunu, U_t ise t anındaki kontrol bilgisini temsil etmektedir. Doğru kestirim ve hesaplamalar sayesinde robot/özerk araç etrafın haritasını oluşturabilir ve kendi pozisyonunu kestirebilir. Eğer başlangıç pozisyonu global referansa göre biliniyor ise, yani; mutlak pozisyon bilgisi, gezinme esnasında Global Konumlama Sistemi (GPS) ve benzeri kanallardan elde ediliyor ise, oluşturulan harita global olarak işlenebilir. SLAM uygulamaları için robot/özerk araç kinematik (hareket) modeli şekil 2.3'te verilmiştir.



Şekil 2.3 Robotun/özerk aracın koordinat sistemiyle tanımlanan kinematik modeli gösterimi

Burada, aracın direksiyon kontrolü α ve hızı v_c , aracın pozisyonunu Ackermann direksiyon geometrisi (ADG) ile tahmin etmek için kullanılır. Bununla birlikte harici algılayıcı bilgileri (GPS, Laser veya Görsel olabilir) çevrenin haritasının çıkarılmasında kullanılır. Koordinat üzerindeki $B_{i(i=1\dots n)}$ 'ler harici algılayıcı bilgilerini verir. Bu bilgiler robot/özerk araç pozisyonuna bağlı olarak ilişkili menzil (r) ve yön açısının (β), $z_t = (r, \beta)$, elde edilmesinde kullanılır. Robot/özerk araç, hız ve yön açısına bağlı olarak, arka aksın yörüngesini ADG'yle tahmin eder. ADG modeli,

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\varphi}_c \end{bmatrix} = \begin{bmatrix} v_c \cdot \cos(\varphi) \\ v_c \cdot \sin(\varphi) \\ \frac{v_c}{L} \cdot \tan(\alpha) \end{bmatrix} + \gamma \quad (2.1)$$

olarak tanımlanır (Nebot vd. 2006). Burada L teker aksları arasındaki mesafe, γ ise beyaz gürültüdür. Robot/ özerk araç durumuyla ilgili gözlem eşitlikleri,

$$z^t = h(X^t, x_i^t, y_i^t) = \begin{bmatrix} z_\gamma^{i,t} \\ z_\beta^{i,t} \end{bmatrix} = \begin{bmatrix} \sqrt{(x_i^t - x_L^t)^2 + (y_i^t - y_L^t)^2} \\ \text{atan}\left(\frac{(y_i^t - y_L^t)}{(x_i^t - x_L^t)}\right) - \varphi_L^t + \pi/2 \end{bmatrix} + \gamma_h^t \quad (2.2)$$

formülünden elde edilir. Burada z^t gözlem vektörü, (x_i^t, y_i^t) t anındaki nesne koordinatıdır. $(x_L^t, y_L^t, \varphi_L^t)$ t anındaki harici algılayıcı pozisyonundan elde edilen araç durum bilgisi ve γ_h^t ise algılayıcı gürültüsüdür. Çoklu gözlem durumunda gözlem vektörü,

$$Z = \begin{bmatrix} z^{1,t} \\ \vdots \\ z^{m,t} \end{bmatrix} \quad (2.3)$$

olarak gösterilir. SLAM çalışmalarında robot/özerk araç, bir başlangıç noktasından başlayarak öngörülen belirsizlik bilgisiyle birlikte bilinmeyen bir pozisyondan başlar ve robot/özerk araç lokasyonuna bağlı olarak çevrenin haritasını oluşturur. Burada bahsedilen belirsizlik bilgisi, ölçüm ve sistem gürültüleri bilgisiyle elde edilen kovaryans matrisini tanımlar, temsil edilen gerçek değerden olası sapma miktarı olarak tanımlanabilir. Robot/özerk araç lokasyonu ve çevrenin harita bilgisi hem robotun/özerk aracın hareket bilgisiyle hem de yeni ölçümlerle birlikte artımlı olarak devam eder. Robot/özerk araç ilerledikçe yeni nesnelere algılanmaya başlar ve harita giderek büyür. Harita bilgisi ve robot/özerk araç pozisyon bilgisi bir durum vektöründe gösterilir. Tespit edilen nesnelere sayısı arttıkça etrafın haritası şekillenmeye başlar. Böylece durum vektörü yeni eklenen yeni nesnelere büyür. Durum vektörü

$$X = \begin{bmatrix} X_L^t \\ X_I^t \end{bmatrix} \quad (2.4)$$

$$X_L^t = (x_L^t, y_L^t, \varphi_L^t)^T \in R^3$$

$$X_I^t = (x_1^t, y_1^t, \dots, x_N^t, y_N^t)^T \in R^{2N}$$

formülleriyle gösterilir. Burada $(x_L^t, y_L^t, \varphi_L^t)$ ve (x_i^t, y_i^t) robot/özerk araç durumu ve haritayla ilişkili nesne bilgilerini göstermektedir.

ADG'ye bağı olarak SLAM uygulamaları, genel olarak kullanılan farklı istatistiksel yöntemlerle gerçekleştirilmektedir. Tez çalışması kapsamında bu istatistiksel yöntemlerden kısaca bahsedilmiştir.

2.1 SLAM için Kullanılan Temel İstatistiksel Yöntemler

SLAM için geliştirilen algoritmalarından bazıları (Kalman süzgeçleri, Parçacık süzgeçleri gibi) artımlı yöntemler olduğundan verinin tümünü bilmesine gerek yoktur, sadece bir önceki adımı bilmesi yeterlidir. Dolayısıyla gerçek zamanlı uygulamalarda kullanılabilirler. Fakat bazı algoritmalar (EM vb.) daha önceden geçilen yolu bilmek zorundadır. Bu algoritma modelleri bu yüzden gerçek zamanlı uygulanamazlar. Bu yöntemlerin ortak tarafları ise hepsinin istatistikî yöntemlere dayanıyor olmasıdır.

2.1.1 Kalman tabanlı süzgeçler

SLAM uygulamalarının ilk literatüre girdiği tarihten bu tarafa kullanılan yöntemlerin en bilineni Kalman süzgeci tabanlı yaklaşımlardır. Kalman süzgeçleri soncul (*posterior*) durum tahminini $p(x_L^t; x_i^t | z^t; u^t)$ ' yi Gauss dağılımı ile gösteren özel Bayes tabanlı bir yapıya sahiptir (Kalman 1960, Maybeck 1979).

Kalman süzgeçlerinin SLAM uygulamalarında yalnız başlarına kullanıldıklarında bazı dezavantajları vardır. Bunlar şu şekilde sıralanabilir;

- Yapay sınır duvarlara ihtiyaç duyar,
- Veri ilişkilendirme problemini çözemez,
- Direk ham algılayıcı verisi kullanılamaz, veriler ön işlemlerden geçirilmelidir,
- Algılayıcı verisinin sahip olduğu gürültü tipi Gauss tipinde olmalıdır,
- Başlangıçta robot pozisyonunda kayma olmadıysa sonuç iyi çıkar, fakat kayma olduysa ölçüm gürültüsünün istatistikî olarak bir önceki ölçüm gürültüsüne bağımlı olmasından zamanla bu kayma büyüyeceğinden iyi bir sonuç elde edilemez.

2.1.1.1 Kalman süzgeci (KF)

SLAM uygulamalarında doğrusal sistem durumları için en genel çözüm olarak Kalman süzgeçleri tercih edilmektedir. Kalman süzgeci, Bayesian tabanlı durum uzay kestirim yapısına sahiptir. Bayes teoremi, bir durum hakkında ek bilgilere sahip olunarak durumun nasıl değişeceğini gösteren bir teoremdir. Teorem iki önemli olasılık yaklaşımı olan birleşik ve koşullu olasılığı kullanır. SLAM uygulamalarında Bayes kuralı kullanılarak soncul dağılım,

$$p(s_t, \Theta | z^t, u^t, n^t) = \eta p(z_t | s_t, \Theta, z^{t-1}, u^t, n^t) p(s_t, \Theta | z^{t-1}, u^t, n^t) \quad (2.5)$$

eşitliğiyle gösterilir (Montemerlo 2003). Burada, s_t t anındaki robot/özerk araç pozisyonunu, Θ bütün harita bilgisini, z^t t anına kadar bütün ölçüm bilgilerini, u^t t anına kadar bütün kontrol giriş bilgilerini ve n^t ise veri ilişkilendirme bilgisini göstermektedir. Bayes kuralına göre payda, normalleştirilmiş bir sabittir, (2.5)'te η olarak gösterilmiştir. SLAM uygulamalarında z_t , sadece robotun/özerk aracın pozisyonunun bir fonksiyonunu (s_t), Θ , haritasını ve n_t , en son veri ilişkilendirmeyi gösterir ve bu semboller ölçüm modelinde tanımlanmıştır. Durum, soncul dağılım için tekrar yazılırsa,

$$= \eta p(z_t | s_t, \Theta, n_t) p(s_t, \Theta | z^{t-1}, u^t, n^t) \quad (2.6)$$

eşitliği elde edilir. Olasılık teoremi yardımıyla, (2.6) eşitliğinin sağ tarafının sağ terimi $t-1$ anında,

$$= \eta p(z_t | s_t, \Theta, n_t) \int p(s_t, \Theta | s_{t-1}, z^{t-1}, u^t, n^t) p(s_{t-1} | z^{t-1}, u^t, n^t) ds_{t-1} \quad (2.7)$$

eşitliğine dönüşür. İntegralin iç kısmının en sol tarafı koşullu olasılık tanımından

$$= \eta p(z_t | s_t, \Theta, n_t) \quad (2.8)$$

$$\int p(s_t | \Theta, s_{t-1}, z^{t-1}, u^t, n^t) p(\Theta | s_{t-1}, z^{t-1}, u^t, n^t) p(s_{t-1} | z^{t-1}, u^t, n^t) ds_{t-1}$$

eşitliğini sağlar. İntegralin birinci terimi, s_t yalnızca s_{t-1} ve u_t 'in bir fonksiyonu olduğu için, basitleştirilerek

$$= \eta p(z_t | s_t, \Theta, n_t) \quad (2.9)$$

$$\int p(s_t | s_{t-1}, u^t) p(\Theta | s_{t-1}, z^{t-1}, u^t, n^t) p(s_{t-1} | z^{t-1}, u^t, n^t) ds_{t-1}$$

formuna dönüşür (Montemerlo 2003). Bu durumda integralin sağ iki terimi birleştirilerek

$$= \eta p(z_t | s_t, \Theta, n_t) \quad (2.10)$$

$$\int p(s_t | s_{t-1}, u^t) p(s_{t-1}, \Theta | z^{t-1}, u^t, n^t) ds_{t-1}$$

eşitliği elde edilir. Eğer s_{t-1} için, anlık kontrol bilgisi u_t ve veri ilişkilendirme n_t bilgisi yok ise, integralin en sağ tarafı ihmal edilebilir (Montemerlo 2003). Sonuç olarak t anındaki SLAM soncul dağılımı, $t-1$ anındaki SLAM soncul dağılımından hesap edilir ve tekrarlı bir formül hareket modeline $p(s_t | s_{t-1}, u_t)$ ve ölçüm modeline $p(z_t | s_t, \Theta, n_t)$ bağlı olarak

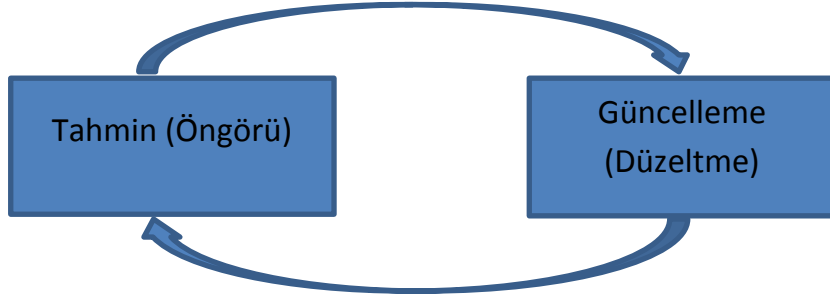
$$p(s_t, \Theta | z^t, u^t, n^t) = \eta p(z_t | s_t, \Theta, n_t) \quad (2.11)$$

$$\int p(s_t | s_{t-1}, u_t) p(s_{t-1}, \Theta | z^{t-1}, u^{t-1}, n^{t-1}) ds_{t-1}$$

yazılır.

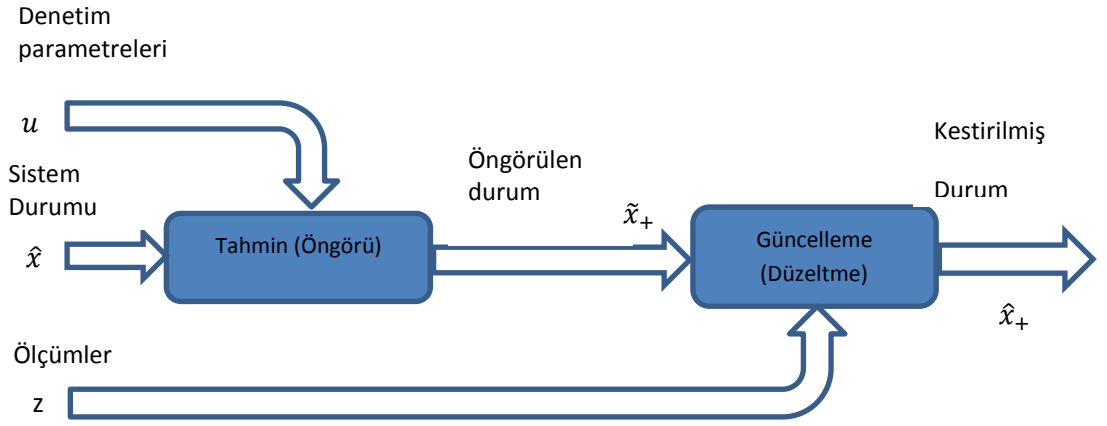
Kalman süzgeçlerinde bütün olasılık yoğunluk fonksiyonları (OYF) Gauss dağılımına uyduğu kabul edilir. Gaussian dağılım, sadece ortalama (μ) ve varyans (σ^2) ifadeleriyle tanımlanabildiğinden, süzgeç soncul dağılımı daha kolayca kestirilebilir.

Kalman süzgeci, tahmin ve güncelleme yapısı sayesinde özyineli olarak durum kestirimi gerçekleştirir. Bu işlem, öngörülen durumun, sistemden algılanan verilerle düzeltilmesi işleminin tekrarlanması ile şekil 2.4'teki gibi gerçekleşir.



Şekil 2.4 Kalman süzgeci Tahmin-Güncelleme döngüsü

Eğer sistem durumu zamanla değişiyorsa geleneksel Kalman süzgecinin kestirim doğruluğu düşer ve performansı azalır. Sistem değişkenlerinin gözlemlenebiliyor olması, sistemin durum kestirimini etkileyen diğer bir etken olarak gösterilebilir. Sistem değişkenleri doğrudan gözlemlenemez ise algılayıcılardan gelen veriler ile sistem değişkenlerinin gözlenmesi sağlanabilir. Algılayıcılardan gelen veriler gürültüye sahip olduğu için sistem durum kestirimi daha zor bir hal alacaktır. Bu olumsuzluklara rağmen Kalman süzgeci kestirim işlemini doğrusal durumlar için en iyi şekilde gerçekleştirir. Şekil 2.5, dinamik bir model yapısı için Kalman süzgecinin alınan veriler dâhilinde kestirim şeklini göstermektedir.



Şekil 2.5 Dinamik bir model için Kalman süzgecinin verilen bilgiler dâhilinde kestirim blok diyagramı

Kalman süzgeçlerinde sistem dinamikleri, durum-uzay modeli gösteriminde hareket modeli ve ölçüm modeli olarak gösterilir.

$$\hat{x}_t = Fx_{t-1} + Gu_{t-1} + w_{t-1} \quad (2.12)$$

$$z_t = Hx_t + v_t \quad (2.13)$$

Burada, F, G ve H matrisleri durum-uzay modelini oluşturan sırasıyla durum, giriş ve çıkış matrisleridir. Ayrıca, t zaman göstergesini, x sistemin durumunu, u sistemin bilinen girişini, z ölçülen çıkışı, w ve v sırasıyla sistem ve ölçüm gürültülerini göstermektedir. Bu bileşenler çoğunlukla birden fazla elemana sahiptir. x durum vektörü sistemin mevcut durumuyla ilgili tüm bilgiyi taşır, fakat x doğrudan ölçülmez. Bunun yerine x 'in bir fonksiyonu olan ve v beyaz gürültüsüyle bozulmuş z 'yi doğrudan ölçebiliriz. z kullanılarak durum vektörü kestirimi yapılabilir. Öte yandan z çıkışı da gürültünün etkisindedir. Dolayısı ile ölçüm sonuçlarına güvenilebilir, fakat tamamı ile güvenmek kestirim sonuçlarında hatalara yol açacaktır.

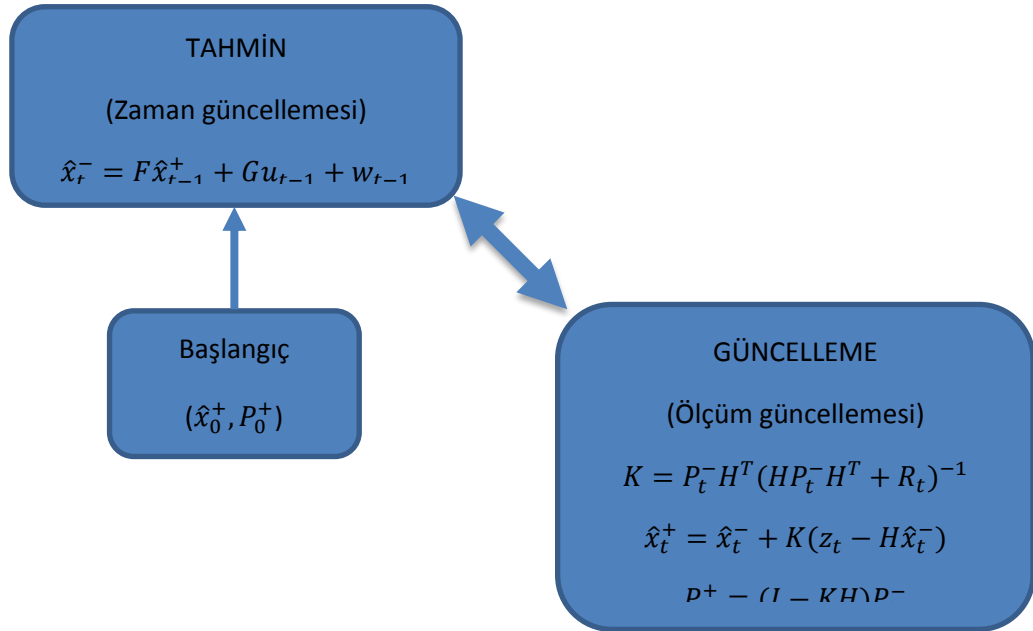
Kalman süzgeci algoritması belirli başlangıç koşullarında iki aşamada gerçekleşir;

- Tahmin (Öngörü)
- Güncelleme (Düzeltilme)

Kalman süzgeçleri özyineli bir yapıya sahip olduğu için başlangıç koşulları önemlidir. Başlangıç durum bilgisi \hat{x}_0^+ ve başlangıç durum belirsizliği P_0^+ iyi seçilmeli veya tanımlanmalıdır.

Süzgeç çalışma esnasında, öngörünün yapılabilmesi için kontrol girdi bilgileri u_t ve bir önceki durumun öngörüsü \hat{x}_t^- bilgisine sahip olmalıdır. Böylece bir önceki öngörüye ait belirsizlik P_t^- değeri hesap edilebilir. Böylece Kalman süzgeci tahmin aşamasında zaman güncelleme işlemini gerçekleştirmiş olacaktır.

Güncelleme aşamasında, süzgeç bir sonraki durum kestirimini \hat{x}_t^+ ve belirsizliği P_t^+ hesaplar. Bu işlem sonucu elde edilen durum ve belirsizlik bilgisi, bir sonraki zaman adımında önceki durum ve belirsizlik işlevinin (\hat{x}_t^-, P_t^-) yerini alır. İşlem şekil 2.6'da özetlenmiştir.



Şekil 2.6 Doğrusal Kalman süzgeci blok diyagramı

Burada, I birim matrisi, R_t t anındaki ölçüm gürültüsü matrisini, Q_{t-1} , $t-1$ anındaki sistem gürültüsü matrisini, K ise Kalman kazancını tanımlamaktadır. Algoritma özetlenirse, tahmin aşamasında, sistemin durum ve belirsizliğinin öngörü yapılır,

güncelleme aşamasında ise sistemden alınan ölçümler kullanılarak öngörü düzeltilir. Böylece her hangi bir t anında kestirim işlemi gerçekleştirilmiş olur.

2.1.1.2 Genişletilmiş Kalman Süzgeci (EKF)

Kalman süzgeci doğrusal sistemler için optimal sonuçlar sağlayabilmektedir (Kalman 1960). Doğrusal olmayan durumlar içinde Kalman süzgeci kullanılmak istenirse algoritmada bazı değişiklikler yapılması gerekmektedir. Gerçek hayatta birçok sistem (2.12) ve (2.13) eşitliklerinde bahsedildiği gibi modellenemez. Doğrusal olmayan sistemler için durum-uzay model gösterimi,

$$\hat{x}_t = f(x_{t-1}, u_{t-1}, w_{t-1}) \quad (2.14)$$

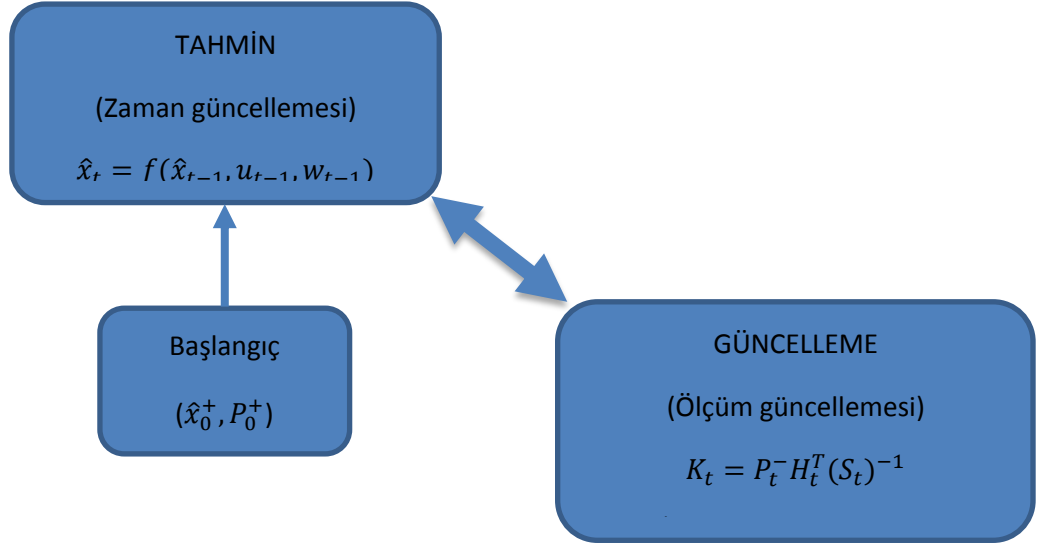
$$z_t = h(x_t, v_t) \quad (2.15)$$

formuna dönüşür. Bu durumda Kalman süzgeçlerinin doğrusal olmayan sistemler için kullanımı $f(\cdot)$ ve $h(\cdot)$ matrislerinin doğrusallaştırılmasıyla olur. Bu durum şu şekilde açıklanabilir; doğrusal olmayan dönüşüm işlemlerinden sonra Gauss dağılımı özelliğini yitirmektedir, yani soncul Gauss özelliği gösterememektedir. Dolayısı ile $f(\cdot)$ ve $h(\cdot)$ matrisleri Taylor serisi açılımı yardımıyla birinci dereceden kısmi türevleri alınarak doğrusallaştırılır. Böylece F ve H 'ın Jakobiye matrisleri (2.16) ve (2.17)'te gösterildiği şekilde hesaplanır.

$$F_t = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}_{t-1}, u_{t-1}} \quad (2.16)$$

$$H_t = \left. \frac{\partial h}{\partial x} \right|_{\hat{x}_t} \quad (2.17)$$

EKF algoritması, KF'yle karşılaştırıldığında çok farklı olmadığı görülür. En belirgin farklılığının sistem ve ölçüm matrislerinin kısmi türevlerinin alınmasıdır. Burada Jakobiye işlemi öz yineleme işleminin her bir zaman adımında tekrar hesaplanır. EKF algoritması şekil 2.7'de verilmiştir.



Şekil 2.7 EKF algoritması işlem blok diyagramı

Burada, $S_t = H_t P_t^- H_t^T + R_t$ inovasyon kovaryans matrisi olarak ifade edilir. Bir sondaki durum belirsizliğinin hesaplanmasında bu matris kullanılır.

İlk başlarda, doğrusal olmayan sistem dinamikleriyle SLAM için EKF tercih edilse de bazı dezavantajlara sahip olduğu gözlenmiştir. Bunların başında quadratik kompleksite (belirsizlik matrisinin durum vektörüne bağlı olarak boyutunun çok büyümesi sonucu işlem yükü fazlalığı) ve tek hipotezli veri ilişkilendirme özelliği gelmektedir. Matematiksel işlemlerin sayısı, kontrol giriş bilgisi ve gözlemlerin süzgeç içerisinde işlenmesini gerektirmektedir. K_t ve S_t^T 'nin boyutlarının $2N+3$ olduğu göz önüne alınırsa işlem yükünün fazlalığının belirsizlik matrisi için ne kadar büyüdüğü anlaşılır. İkinci problem incelenirse, SLAM uygulamalarında n_t 'nin bilinmediği kabul edilir. EKF, her bir gözlem için maksimum sezgisel komşuluğu kullanarak tek veri ilişkilendirme hipotezini devam ettirir. Eğer, algılayıcılardan gelen anlık nesne bilgisi olasılığı düşük ise algılanan nesne bilgisi yeni nesne olarak atanır. Bu durumda sezgisel olarak seçilen veri ilişkilendirme hatalı ise, bu gözlem için diğer nesnelere ve robotla/araçla işbirliğinin etkisi asla göz ardı edilemez. Eğer çoklu veri ilişkilendirme hatası gerçekleşirse EKF'nin doğru sonuçtan iraksadığı görülür (Newman 1999, Montemerlo 2003).

2.1.1.3 Kokusuz dönüşüm (Unscented Transform, UT) ve kokusuz Kalman süzgeci (UKF) algoritması

UKF ile EKF arasındaki temel fark, doğrusal olmayan bir model için durum dağılımlarının farklı olmasıdır. UKF yapısında kestirim işlemi gerçekleştirilirken doğrusallaştırma yapılmaz. UKF’de, x rasgele değişkenleri, doğrusal olmayan z rasgele değişkenine ($z = f(x)$) dönüştürülerek işlem yapılır. Bu rasgele değişkenlerin doğrusal olmayan bir fonksiyonla birlikte UKF öngörü adımında kullanılması UT sayesinde olur. UT, fonksiyonları doğrusal hale dönüştürmeden dağılımın üzerindeki belirli sayıda olan noktaları kullanır. Bu noktalar, ortalama, kovaryans ve olası yüksek dereceli momentler göz önünde bulundurularak Gauss değişkeniyle eşleşecek şekilde belirlenir. Ortalama ve kovaryans, doğrusal olmayan fonksiyon üzerinde sigma noktaları sayesinde hesaplanmakta, bu sayede UKF, klasik doğrusallaştırma kullanan metotlara göre daha doğru kestirim sonuçları sağlamaktadır (Julier vd. 1995). UKF’de önerilen temel düşünce, fonksiyonun kendisi yerine olasılığına yaklaşım sağlamaktır (Julier vd. 1995, Van ve Merwe 2006). Böylece amaçlanan yaklaşımla, hem hesaplama karmaşıklığı azaltılır hem de daha doğru ve gülbüz sonuçlar elde edilir.

UKF’nin doğrusal olmayan durumlarda EKF’ye nazaran daha uygun sonuçlar verdiği söylene de bazı kısıtlara sahip olduğu bilinir. Örneğin, süzgecin olasılığı üzerinde seçilen sigma noktaları arasındaki uzaklık ağırlık parametrelerine bağlıdır. Bu parametreler dağılımın özelliğine bağlı olarak belirli limitler dâhilinde seçilir. Sigma noktalarının olasılık üzerinde yayılımları ve bahsedilen bu limitlerin altında seçilmesi durumunda, pozitif yarı tanımlı korelasyona sahip matrisler elde edilemez. Pozitif yarı tanımlılık kavramı EK 2’de anlatılmıştır. Noktalar arasındaki mesafe ayrıca durum-uzayın boyutuyla değişir. Boyutun artmasıyla ortaya çıkan yüksek sigma yayılımı yüksek dereceli doğrusal olmayan modellerde yayılımdan kaynaklı problemler ortaya çıkaracaktır. Yine de literatür incelendiğinde UKF’nin özellikle doğrusal olmayan ortamlarda EKF’den daha başarılı kestirim sonuçları verdiği bilinmektedir. UKF’yi anlamak için UT’yi bilmek gereklidir.

UT'yle, herhangi bir x rasgele değişkeninin doğrusal olmayan bir fonksiyonla z rasgele değişkenine dönüştürülme ($z = f(x)$) işlemi gerçekleştirilir. x rasgele değişkeninin ortalaması \bar{x} ve kovaryansı P_x olarak düşünülürse, UT'de, sigma noktaları kümesinin örneklem ortalamaları \bar{x} ve örneklem kovaryansları P_x olacak şekilde seçilir. Burada doğrusal olmayan fonksiyon her bir sigma noktasına uygulanır. Böylece dönüşümden elde edilen noktalar ve bu noktaların \bar{z} ve P_{zz} istatistikleri elde edilir (Julier vd. 1997). Sigma noktalarının olasılık içerisinde seçim yerleri deterministik bir algoritmaya göre belirlenir. UT ile \bar{x} ortalamalı ve P_x kovaryanslı N boyutlu x rasgele değişkeni, $2N+1$ adet ağırlıklandırılmış noktaya dönüştürülür. UT dönüşümünde sigma noktaları,

$$\begin{aligned} \chi_0 &= \bar{x}, & W_0^{(m)} &= \frac{\lambda}{(N+\lambda)} \\ \chi_i &= \bar{x} + (\sqrt{(N+\lambda)P_x})_i & W_0^{(c)} &= \frac{\lambda}{(N+\lambda)} + (1 - \alpha^2 + \beta) \quad i = 1, \dots, N \\ \chi_i &= \bar{x} - (\sqrt{(N+\lambda)P_x})_{i-N} & W_i^{(c)} &= W_i^{(m)} = \frac{1}{2(N+\lambda)} \quad i = N+1, \dots, 2N \end{aligned} \quad (2.18)$$

formülleriyle gösterilir. Burada λ ölçekleme parametresi olup $\lambda = \alpha^2(N + \kappa) - N$ ifadesiyle gösterilir. κ ise ikincil ölçekleme parametresidir, Gauss dağılımlarda genel olarak 0 seçilir. α , x ortalamasının etrafında sigma noktalarının dağılmasını sağlayan parametredir, genelde küçük bir pozitif değer (1e-3 gibi) olarak alınır. $(\sqrt{(N+\lambda)P_x})_i$, belirsizlik matrisinin karekökünün i 'nci satır veya sütunundaki değerini ifade etmektedir. W 'lar ise ilgili ağırlıkları temsil etmektedir. Bu dönüşüm, UKF'nin EKF'den temel farklılığıdır. UKF'de sigma noktaları doğrusal olmayan fonksiyona eşitlik (2.19)'daki formülasyonla yayılır.

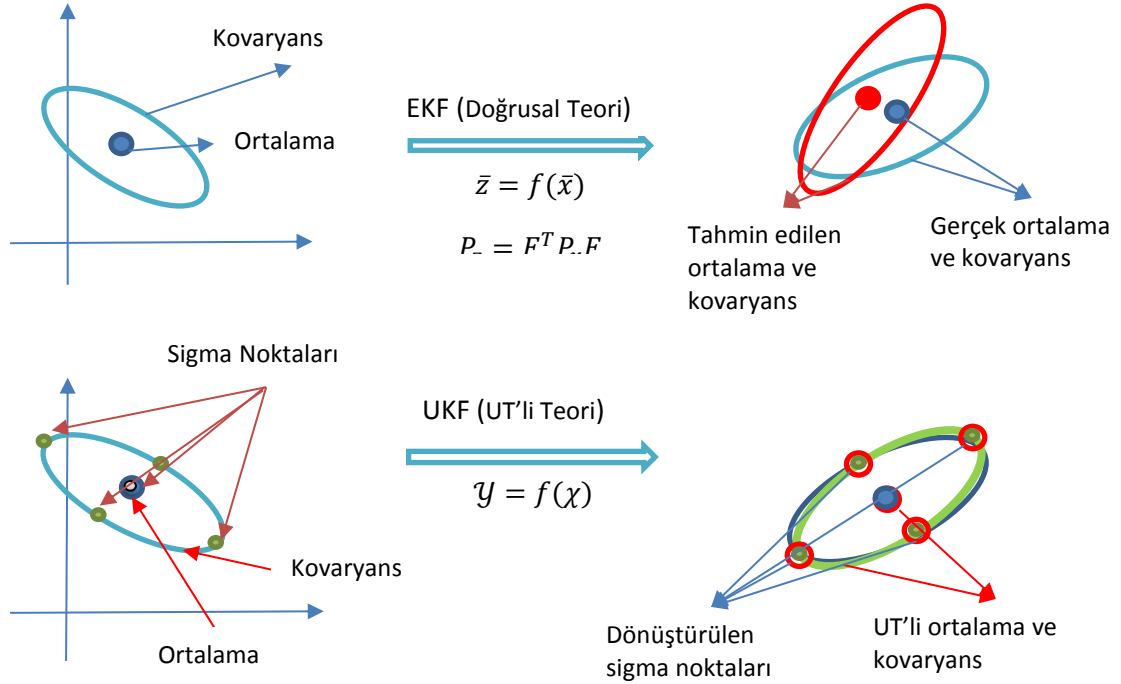
$$y_i = f(\chi_i), \quad i = 0, \dots, 2N \quad (2.19)$$

z için ortalama ve kovaryans, sigma noktalarının sonucunun ağırlıklandırılmış ortalama ve soncul dağılımın kovaryansı bilgisi,

$$\bar{z} \approx \sum_{i=0}^{2N} W_i^{(m)} y_i \quad (2.20)$$

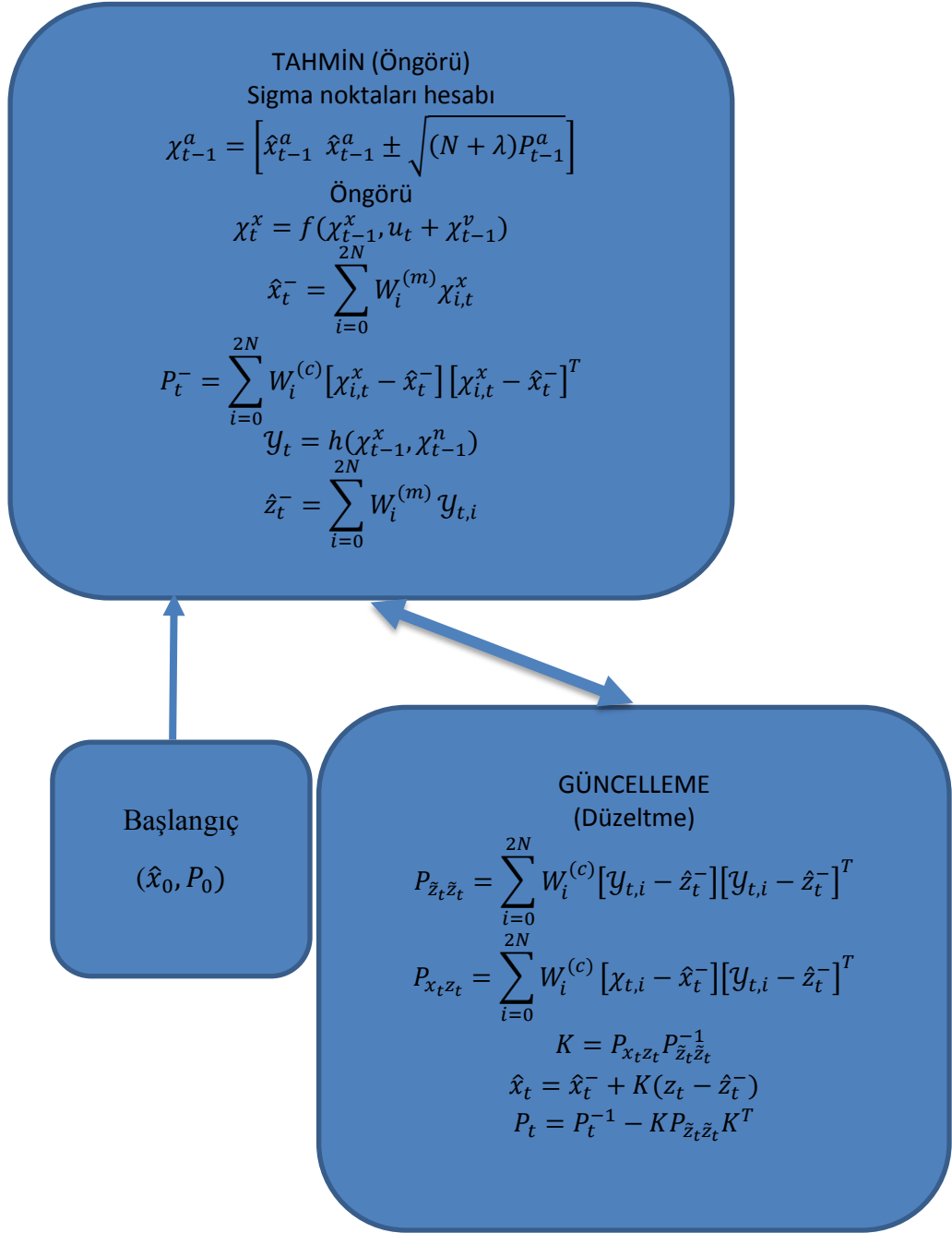
$$P_{zz} \approx \sum_{i=0}^{2N} W_i^{(c)} \{y_i - \bar{z}\} \{y_i - \bar{z}\}^T \quad (2.21)$$

hesaplanır. UT, geleneksel örneklem metotlarından (örneğin Monte-Carlo metotları, parçacık süzgeçleri gibi) bütünüyle farklıdır. Durumun soncul hesabının doğru olasılığa yayılabilmesi için, durumun boyutuna bağlı olarak, dağılımda daha fazla örnek noktaya ihtiyaç duyulur. Bu durum UT için bir geri adımdır. Öte yandan UT'nin, üçüncü dereceden Gauss girişlerde bile başarı sağlaması bir diğer avantajıdır (Van ve Merwe 2006). İki boyutlu bir uzayda ortalama ve kovaryans değerlerinin EKF ve UKF için örnek sonuçları şekil 2.8'deki benzetimde gösterilmiştir.



Şekil 2.8 İki boyutlu bir uzayda ortalama ve kovaryans değerlerinin EKF ve UKF için temsili benzetim sonuçları

Şekil 2.8 temsili olarak oluşturulmuş iki boyutlu uzayda doğrusal dönüşüme uğramış ve doğrusal dönüşüm olmadan gerçekleştirilen ortalama ve kovaryans hesaplarını göstermektedir. Doğrusallaştırılarak hesaplanan ortalama ve kovaryans değerleri gerçek değerlerden uzaklaşırken, UT dönüşümlü yaklaşımda gerçek değere yakın sonuçlar elde edilmiştir. UKF süzgecinin eşitlikleri şekil 2.9'da verilmiştir.



Şekil 2.9 UKF algoritması blok diyagramı

Burada, \hat{z}_t^- , tahmin edilen öncül ölçüm bilgisini, y_t , sigma noktalarının yayıldığı doğrusal olmayan fonksiyonu, $P_{\tilde{z}_t \tilde{z}_t}$, soncul dağılımın kovaryansını, $P_{x_t z_t}$, çapraz kovaryansı, K , Kalman kazancını, \hat{x}_t ve P_t ise sırasıyla düzeltilen durum ve kovaryans matrisini göstermektedir.

2.1.1.4 Merkez fark Kalman süzgeci (Central Difference Kalman filter, CDKF)

CDKF, doğrusal olmayan durumlar için UKF'ye alternatif olarak sunulmuş Kalman tabanlı bir süzgeçtir (Ito ve Xiong 2000). Literatür incelendiğinde CDKF ve UKF'nin aynı ortam koşullarında kestirim performanslarının yaklaşık olarak aynı doğruluk oranında çıktığı bilinmektedir (Ito ve Xiong 2000, Merwe 2004). Öte yandan CDKF'nin, süzgeç döngüsünde tahmin adımında kullanmış olduğu yöntem ve Gauss dağılımlar için, yalnızca 1 adet ölçekleme parametresi (h) kullanması avantajları arasında sayılır. CDKF'nin algoritmasına geçmeden önce öngörü adımında kullandığı Stirling's polinomsal interpolasyon (SPI) yaklaşımını tanımak gereklidir.

CDKF'nin temeli Stirling's Polinomsal İnterpolasyon (SPI)'ye dayanır. Kokusuz dönüşümde (UT) olduğu gibi olasılık dağılımı üzerinde simetrik ($2N+1$) adet sigma noktaları seçilir. Sigma noktaları,

$$\begin{aligned}\chi_0 &= \bar{x}, \\ \chi_i &= \bar{x} + (h\sqrt{P_x})_i \quad i = 1, \dots, N \\ \chi_i &= \bar{x} - (h\sqrt{P_x})_{i-N} \quad i = N + 1, \dots, 2N\end{aligned}\quad (2.22)$$

eşitliklerinden elde edilir. Burada N , x 'in boyutunu göstermektedir. Her bir nokta doğrusal olmayan fonksiyona (2.23)'teki gibi yayılır,

$$y_i = f(\chi_i), \quad i = 0, \dots, 2N \quad (2.23)$$

(2.22-23) eşitliklerden SPI, sonculun ortalamasını, kovaryansı ve çapraz kovaryansı ağırlıklandırılmış sigma noktalarının doğrusal bir bileşeni gibi eşitlik (2.24-26)'daki gibi hesaplanır.

$$\hat{z} = \sum_{i=1}^{2L} w_i^{(m)} y_i \quad (2.24)$$

$$P_{\tilde{z}_t \tilde{z}_t} = \sum_{i=1}^N \{w_i^{(c_2)} [y_{t,i} + y_{t,i+N} - 2y_0] [y_{t,i} + y_{t,i+N} - 2y_0]^T \} \quad (2.25)$$

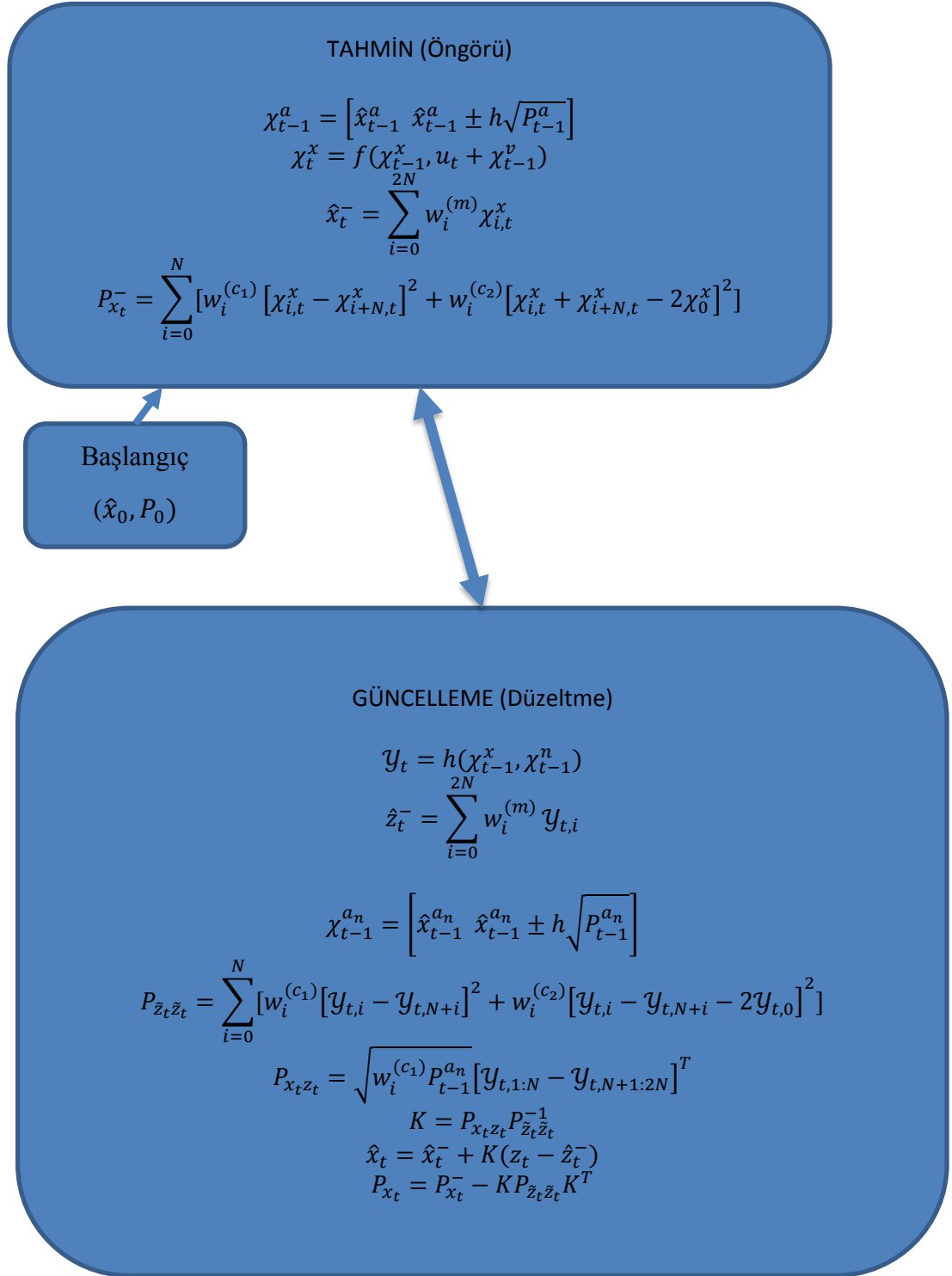
$$+ \sum_{i=1}^N w_i^{(c_1)} [y_{t,i} - y_{t,i+N}] [y_{t,i} - y_{t,i+N}]^T \} \quad (2.26)$$

$$P_{x_t z_t} = \sqrt{w_i^{(c_1)}} P_{x_t} [\chi_i - \hat{x}] [y_{t,i} - y_{t,i+N}]^T$$

Burada, $w_i^{(m)}$ ağırlıkların kümesini temsil etmektedir ve soncul ortalamanın hesaplanması için kullanılır. $w_i^{(c)}$ ise kovaryansın ve çapraz kovaryansın düzeltilmesi için kullanılan ağırlıkları göstermektedir. Bu ağırlıklar,

$$\begin{aligned} w_0^{(m)} &= \frac{h^2 - N}{h} & , w_i^{(m)} &= \frac{1}{2h} \\ w_i^{(c_1)} &= \frac{1}{4h^2} & , w_i^{(c_2)} &= \frac{h^2 - 1}{4h^2}, \quad i = 1, \dots, 2N \end{aligned} \quad (2.27)$$

olarak tanımlanır. Burada tanımlanan h parametresi ağırlıkların hesaplanması ve sigma noktalarının dağılım üzerinde pozisyonlarının belirlenmesinde önem arz etmektedir. Gauss rasgele değişkenler için $h = \sqrt{3}$ olarak alınır (Merwe 2004). SPI ve CDKF'nin detayları için EK 3'e bakılabilir. SPI kullanılarak elde edilen CDKF'nin eşitlikleri şekil 2.10'da verilmiştir.



Şekil 2.10 CDKF algoritması blok diyagramı

2.1.2 Beklenti enbüyültme tabanlı yöntemler

SLAM uygulamalarında karşılaşılan diğer bir yöntem ise Beklenti Enbüyültme (Expectation Maximization, EM)'dir. EM, en büyük olasılırlık (maximum likelihood) tahmin modelinden üretilmiş, saklı değişkenleri olan istatistiksel bir yapıya sahiptir (Dempster vd. 1977).

Bu yöntemde aynı veriyi sistem birçok kez kullanır. Algoritma yapısına bağlı olarak model sadece bir önceki adımdaki duruma bağlı olmayıp, o ana kadar olan tüm durumlara bağlıdır. EM yönteminin Kalman süzgeçlerinden en belirgin farklılığı budur.

Fakat bu yöntemler de bazı kısıtlara sahiptir. SLAM uygulamaları genel olarak gerçek zamanlı olarak kullanılmaktadır, gerçek zamanlı çalışabilen algoritma modeline ihtiyaç duyulur. Bu durum EM için bir dezavantajdır. Öte yandan son zamanlarda EM'den türetilmiş gerçek zamanlı çalışabilen yaklaşımlar öne sürülmüştür (Corff vd. 2011, Faundorfer ve Lee 2013). EM yöntemi çalışırken harita için robot konumunun soncul durumunun hesaplandığı bir beklenti E (Lokalizasyon) adımı ve robot konumu kullanılarak en olası haritanın hesaplandığı bir maksimize M (Harita hesaplama) adımı bulunmaktadır.

E-Adımı:

Robot/özerk araç pozisyonu ($s^{(0)}, \dots, s^{(T)}$) üzerindeki dağılım E adımıyla gerçekleştirilir.

$$Bel_{(j)}(s^{(t)}) = P(s^{(t)} | m_{(j-1)}, d) \quad \forall t = 1, \dots, T \quad (2.28)$$

Buradaki P olasılık fonksiyonunu, m haritayı, d ise robotla/araçla ilgili elde edilen bilgiyi göstermektedir. Bel ise inanış (*Belief*) olarak bildirilen en son tahmin bilgisini temsil etmektedir.

M-Adımı:

M adımıyla tahmin edilen robot pozisyonu için en olası harita hesaplanır.

$$m_{(j)} = \underset{m}{\operatorname{argmax}} P(m|Bel_{(j)}(s^{(0)}), \dots, Bel_{(j)}(s^{(T)}), d) \quad (2.29)$$

$$= \int \dots \int \underset{m}{\operatorname{argmax}} P(m|s^{(0)}, \dots, s^{(T)}, d) Bel_{(j)}(s^{(0)}) \dots Bel_{(j)}(s^{(T)}) ds^{(0)}, \dots, ds^{(T)} \quad (2.30)$$

E adımının başlangıcında, $m_{(0)}$ boş bir haritadır ve

$$Bel_{(j)}(s^{(t)}) = P(s^{(t)}|d) \quad \forall t = 1, \dots, T \quad (2.31)$$

Sonuçta gittikçe kesinleşen ve gerçek ortama benzeyen bir dizi harita elde edilir. $m_{(0)}$; $m_{(1)}$; $m_{(2)}$; ... EM yöntemiyle off-line olarak elde edilmiş bir harita modeli şekil 2.11'de verilmiştir.



Şekil 2.11 Grid Tabanlı Haritalama yaklaşımıyla EM yönteminde elde edilen:

a. elde edilen, b. gerçek harita (Thrun vd. 2000)

EM metodunun detaylı SLAM uygulaması için (Thrun vd. 2000)'e ve algoritmanın derivasyonu için (Dempster vd. 1977)'e bakılabilir.

2.1.3 Sıralı Monte Carlo yaklaşımları

Literatürde SLAM uygulamalarında başarılı sonuçlar verecek ve bahsedilen problemleri aynı anda çözebilecek bazı çalışmalar önerilmiştir. (Montemerlo 2002, 2003) yöntemlerinde SLAM uygulamalarında karşılaşılan üç problemi gerçek zamanlı bir robota/özerk araca uygulayarak çözmeye çalışmıştır. (Montemerlo 2002) ve arkadaşları çalışmalarında şu başlıklar üzerinden problemi çözmeye çalışmışlardır;

- Yöntem doğrusal olmayan durumlarda çalışabilmelidir,
- Gürültü Gauss olmasa da SLAM için anlamlı bir çözüm üretilebilmelidir,
- Sistem gürültüyü minimize ederken aynı zamanda robotun/özerk aracın geçtiği yeri hatırlaması sağlanmalıdır,
- Büyük harita boyutlarında dahi sistemin işlem yükü fazla olmamalıdır.

(Montemerlo 2002) ve arkadaşları bu durum için SLAM modelinde robot/özerk araç pozisyon tahminini Sıralı Monte Carlo (SMC) yaklaşımli parçacık süzgeçleriyle (Raoblackwellized particle filter, RBPF) gerçekleştirmeyi düşünmüşlerdir. Bu durumda sistem gerçek zamanlı uygulamalarda her ortam için geçerli bir çözüm üretebilir hale gelmiştir. Diğer bir durum ise robot/özerk araç pozisyon kestirimi için kullanılan modelin parçacık süzgeçlerinin örneklem ağırlık yapısından yararlanarak veri ilişkilendirme yapabilmesidir. Yani, her bir parçacık bir olası haritayı temsil etmektedir. Örneklem adımından sonra ağırlıklar atanırken olasılığı eşik değerin altında kalan parçacıklar elimine edileceğinden, dolayısı ile parçacığa bağlı haritalarda elimine edilecektir. Diğer bir problem ise nesne sayılarının fazla olduğu ortamlarda sistemin işlem yükünün parçacık süzgeci yapısı sayesinde azaltılmış olmasıdır. (Montemerlo 2002) ve arkadaşları bu durum için algılayıcılardan her bir ölçüm alınması esnasında kaç adet nesne algı alanına girdi ise o kadar paralel EKF kullanılmasını önermiş ve her bir nesne için ayrı EKF kullanarak modeli paralelleştirilmiş, nesnelerin konumunu bu sayede tahmin etmiştir.

2.1.3.1 Parçacık süzgeci tabanlı FastSLAM yaklaşımları

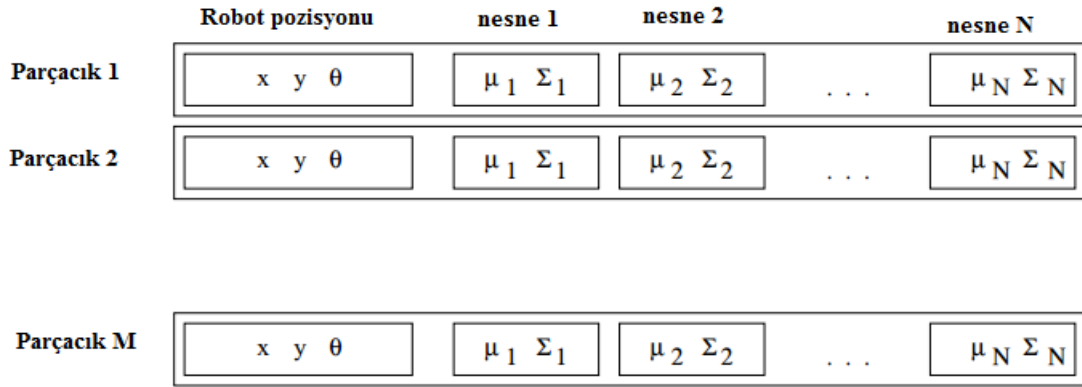
Araştırmacılar son zamanlarda SLAM uygulamalarında SMC tabanlı metotları bahsedilen avantajlarından ötürü kullanmayı tercih etmektedirler (Montemerlo vd. 2002, 2003, Kim vd. 2008, 2009, Ankişhan vd. 2014). Montemerlo (2002) ve arkadaşlarının geliştirmiş olduğu FastSLAM I ve II modeli bu çalışmaların temelini oluşturmaktadır. Bu tez çalışmasında, FastSLAM yaklaşımları ve türevleri üzerinde durulmuş, mevcut çalışmaların eksik yönleri tespit edilerek gerekli iyileştirmeler yapılmıştır. Böylece daha gürbüz sonuçlar veren ve gerçek zamanlı çalışabilen FastSLAM tabanlı yöntemler tez çalışmasıyla SLAM literatürüne sunulmuştur.

SLAM için gürbüz ve daha düşük bir belirsizliğe sahip çözüm üretilebilmesi için FastSlam II yönteminde hem ölçüm gürültüsü minimizasyonu hem de veri ilişkilendirme problemi göz önünde bulundurulmalıdır. Yöntem, işlem yükünü; yani çalışma süresini, RBPF'deki paralellik özelliğinden yararlanarak $O(N^2)$ süreden $M\log N$ e düşürmüştür. Bu işlemi FastSLAM II'de EKF' ler kullanarak gerçekleştirmektedir. Öte yandan bazı çalışmalar, SLAM uygulamalarında EKF'nin yoğun gürültülü doğrusal olmayan ortamlarda durum ortalaması ve hata kovaryansında ıraksama problemi olduğunu göstermiştir (Bailey vd. 2006). Literatürde bu problem göz önünde bulundurularak FastSLAM için farklı Kalman tabanlı yaklaşımlar önerilmiştir (Kim vd. 2008, 2009). Bunun için ilk önce doğrusallaştırma işleminden vazgeçilmiştir. Literatürde bu problem için geliştirilen yaklaşımlar mevcuttur. Bunların en başında UT'li süzgeçler gelmektedir. Deterministik bir yaklaşımla tahminin soncul değeri üzerinden sigma noktaları seçilerek doğrusal olmayan bir fonksiyon üzerinde bu noktalar dağıtılarak ölçüm tahmini yapılmaktadır.

Kim (2008) ve arkadaşları çalışmalarında U-FastSLAM adı altında kokusuz dönüşümden yararlanarak iyileştirilmiş bir FastSLAM II yöntemi geliştirmişlerdir. Yöntem parçacık süzgecinin robot/özerk araç pozisyonunu tahmin etme aşamasında kokusuz Kalman süzgecinden (Unscented Kalman filter, UKF) yararlanmaktadır. UKF'nin yanı sıra CDKF'yi de kullanan bazı çalışmalar mevcuttur (Ankişhan vd. 2014).

2.1.3.1.1 FastSLAM algoritması

FastSLAM I algoritmasında robot pozisyonunun soncul dağılımı bir parçacık süzgeciyle hesaplanır. Geriye kalan N tane nesnenin koşullu sonculu $p(\theta_n | s^t, z^t, u^t, n^t)$ paralel EKF' ler kullanılarak hesaplanır. Burada, her bir EKF tek bir nesnenin pozisyonunu takip eder ve bu yüzden düşük boyutlu ve sabit boyuttadır. Algoritmada nesnelere EKF' leri robot pozisyonuyla ilişkilendirilir, parçacık süzgecinin her bir parçacığı haritasına bağlı olarak kendi EKF setine sahiptir. Toplamda $N \times M$ EKF vardır, M parçacık sayısını, N ise nesne sayısını göstermektedir. Parçacık süzgeci modeli şekil 2.12' de gösterilmektedir.



Şekil 2.12 Robot pozisyonu ve parçacıklara bağlı Parçacık süzgeci modeli gösterimi

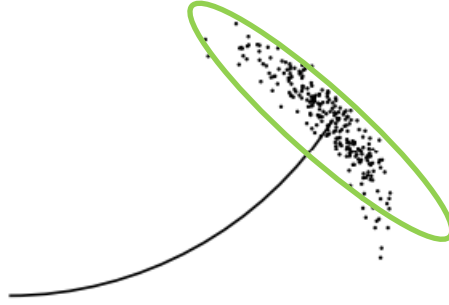
Parçacık süzgecinde M adet parçacık vardır. SLAM için geliştirilmiş bu parçacık süzgecinde her bir parçacık N adet bağımsız EKF içermektedir. Nesnelere arasında çapraz - korelasyon yoktur. Literatürde bu yapı Rao-Blackwellized parçacık süzgeci (RBPF) olarak tanımlanmaktadır. Her bir FastSLAM parçacığı

$$S_t^{[i]} = \left\langle s^{t,[i]}, \mu_{1,t}^{[i]}, \Sigma_{1,t}^{[i]}, \dots, \mu_{N,t}^{[i]}, \Sigma_{N,t}^{[i]} \right\rangle \quad (2.32)$$

formunda gösterilir. Burada $[i]$ parçacık indeksini göstermektedir; $s^{t,[i]}$ i ' inci parçacığın rota - pozisyon kestirimi ve $\mu_{n,t}^{[i]}, \Sigma_{n,t}^{[i]}$ n ' inci nesne pozisyonunun ortalama ve kovaryansdır. Bütün eşitliklerde M adet parçacığın soncul dağılımında i ' inci parçacık

$S_t^{[i]}$ ile gösterilmektedir. Filtreleme yani $t-1$ zamanından t anındaki sonculun hesaplanması S_{t-1} ' den yeni set S_t ' yi üretmek olarak tanımlanır. Yeni parçacık kümesi en son kontrol u_t ve ölçüm z_t ' yi kapsar. Bu güncelleme işlemi şu adımlarla gerçekleşir:

İlk önce yeni bir robot pozisyonu her bir parçacık için en son kontrol girişleriyle oluşturulur. Her kestirilen pozisyon, kestirilen uygun robot rotasına $s^{t-1,[i]}$ eklenir. Devamında nesnelere ilişkilendirilmiş süzgeçlerin gözlenmiş nesnelere bağlantısı oluşturulur, sonrasında yeni gözlemlerle birlikte güncelleme işlemi gerçekleşir. Şekil 2.13, örneklerin olasılık alanında dağılmasını temsili olarak göstermektedir.



Şekil 2.13 Olasılık hareket modelinden örneklerin olasılık alanında dağılmasına örnek bir gösterim

Şekil 2.13'te elipsin içinde ve çevresinde dağılan noktalar parçacıkları temsil etmektedir. Elips ise olasılık bölgesini göstermektedir. Siyah çizgi, tahmini araç rotasıdır. RBPF' deki robot/özerk araç pozisyon parçacığı gerçek pozisyonun sonculundan oluşturulmadığından her bir parçacığa bir önem ağırlığı atanır, böylece aradaki çeşitliliğin oluşması sağlanır. Parçacıkların yeni kümesi S_t , önem tekrar örnekleme kullanılarak ağırlıklandırılmış parçacıklardan oluşturulur. Bu tekrar örnekleme adımı parçacıkların doğru soncula yaklaşılması için önem arz etmektedir. Aşamalar detaylarıyla incelenirse;

- **Yeni pozisyonun örnekleme**

Parçacık kümesi S_t , $t-1$ anında S_{t-1} , gözlem z_t ve kontrol girişi u_t ' den artımlı bir şekilde hesaplanır. FastSLAM algoritmasının ilk adımı, t zamanında olasılıksal olarak robot/özerk araç pozisyonunun her bir parçacığı $s_t^{[i]}$ için tahmin edilir. Bu tahmin, olasılık hareket modeline bağlı örnekleme tarafından oluşturulur.

$$s_t^{[i]} \sim p(s_t | u_t, s_{t-1}^{[i]}) \quad (2.33)$$

Elde edilen kestirim sonucu, parçacıkların oluşturulan geçici kümesine, $s^{t-1,[m]}$ ile birlikte eklenir. Parçacıkların kümesinin S_{t-1} , $p(s^{t-1} | z^{t-1}, u^{t-1}, n^{t-1})$ dağılımına sahip olduğu düşünülürse, yeni parçacıklar:

$$p(s^t | z^{t-1}, u^t, n^{t-1}) \quad (2.34)$$

formunda yazılır. Bu dağılım, parçacık süzgeçlerinin amaçlanan dağılımı (proposal distribution) olarak bilinir.

SLAM uygulamalarında hareket modeli, her hangi bir doğrusal olmayan fonksiyon olabilir. Bu durum doğrusal olmayan modelin doğrusallaştırılmasını gerektirebilir. Sadece bazı uygulamalarda ölçüm modeli örneklerle güvenilir bir şekilde oluşturulabilir (Montemerlo vd. 2003). Amaçlanan dağılımı düşünmeden yeni pozisyonu oluşturmak, her bir parçacık için sabit bir zamana ihtiyaç duyar. Bu, haritanın boyutuna bağlı değildir.

- **Nesne kestirimlerinin güncellenmesi**

FastSLAM düşük boyutlu EKF kullanarak koşullu nesne kestirimini $p(s^t | z^{t-1}, u^t, n^t)$ formunda gösterir (Montemerlo 2002, Montemerlo vd. 2003). Burada, nesnelerin veri

ilişkilendirmesi n^t 'nin bilindiği kabul edilsin. Nesnelere kestirimi robot pozisyon kestirimine bağlı olduğundan N adet EKF, S_t ' de her bir parçacıkla ilişkilendirilir. Burada soncul dağılım, n ' nci nesne pozisyonu θ_n ' den elde edilir. Hesaplama, t anında, nesne θ_n gözlenirse de gözlenmese de gerçekleştirilir. Gözlenen nesne θ_{n_t} için tez içerisinde Bayes kuralı uygulanmıştır.

$$p(\theta_{n_t} | s^t, z^t, u^t, n^t) \quad (2.35)$$

$$\stackrel{Bayes}{\iff} \eta p(z_t | \theta_{n_t}, s^t, z^{t-1}, u^t, n^t) p(\theta_{n_t} | s^t, z^{t-1}, u^t, n^t)$$

(2.35) eşitliğinde Markov özelliği kullanılarak eşitliğin her iki tarafı sadeleştirilir. (2.35)'de gözlem z_t , yalnızca θ_{n_t} , s_t ve n_t ' ye bağlıdır. Benzer olarak θ_{n_t} , z_t olmadan s_t , u_t ve n_t ' den etkilenmemektedir.

$$p(\theta_{n_t} | s^t, z^t, u^t, n^t) \quad (2.36)$$

$$\stackrel{Markov}{\iff} \eta p(z_t | \theta_{n_t}, s_t, n_t) p(\theta_{n_t} | s^{t-1}, z^{t-1}, u^{t-1}, n^{t-1})$$

Eğer $n \neq n_t$ ise her bir nesne soncul dağılımı değişmeden kalır.

$$p(\theta_{n \neq n_t} | s^t, z^t, u^t, n^t) = p(\theta_{n \neq n_t} | s^{t-1}, z^{t-1}, u^{t-1}, n^{t-1}) \quad (2.37)$$

FastSLAM II, (2.37) eşitliğini EKF' ler kullanarak uygulamaktadır. EKF tabanlı FastSLAM yaklaşımlarında süzgeç, soncul dağılımı hesaplarken, dağılımı doğrusal Gauss olarak kabul eder. Hareket modeli doğrusal olmasa bile doğrusal Gauss, gözlem modeliyle birlikte elde edilen sonuç dağılımı $p(\theta_n | s^t, z^t, u^t, n^t)$ Gauss olarak alacaktır. Bu robot/özerk araç pozisyonunda örnekleme adımının bir sonucudur.

Doğrusal olmayan ölçüm modeli $h(s_t, \theta_{n_t})$ birinci dereceden Taylor serisi açılımı kullanılarak yakınsatılabilir. Böylece nesne kestirimi elde edilen robot/özerk araç rotasında gerçekleşir, bu yüzden Taylor serisindeki açılım sadece θ_{n_t} üzerinde geçerli olur. Ölçüm gürültüsünün Gauss dağılımına sahip olduğu ve kovaryansının R_t olduğu kabul edilirse,

$$\begin{aligned}
\hat{z}_t &= h\left(s_t^{[i]}, \mu_{n_t, t-1}\right) \\
G_{\theta_{n_t}} &= \nabla_{\theta_{n_t}} h\left(s_t, \theta_{n_t}\right) \Big|_{s_t=s_t^{[i]}, \theta_{n_t}=\mu_{n_t, t-1}^{[i]}} \\
h\left(s_t, \theta_{n_t}\right) &\approx \hat{z}_t + G_{\theta}\left(\theta_{n_t} - \mu_{n_t, t-1}^{[i]}\right)
\end{aligned} \tag{2.38}$$

kabullenmeler ışığında (2.37)'de eşitliğin sol tarafı

$$p(z_t | \theta_t, s_t, n_t) \sim N(z_t; \hat{z}_t + G_{\theta}\left(\theta_{n_t} - \mu_{n_t, t-1}^{[i]}\right), R_t) \tag{2.39}$$

formunda dağılabilir, sağ tarafı ise Gauss olduğundan $t-1$ anındaki EKF' nin durumuna eşittir.

$$p(\theta_t | s^{t-1}, z^{t-1}, u^{t-1}, n^{t-1}) \sim N(\theta_{n_t}; \mu_{n_t, t-1}^{[i]}, \Sigma_{n_t, t-1}^{[i]}) \tag{2.40}$$

Sonucun ortalama ve kovaryansı, standart EKF güncelleme eşitliklerinden yararlanılarak elde edilebilir.

$$\begin{aligned}
\hat{z}_t &= h\left(s_t^{[i]}, \mu_{n_t, t-1}\right) \\
G_{\theta_{n_t}} &= \nabla_{\theta_{n_t}} h\left(s_t, \theta_{n_t}\right) \Big|_{s_t=s_t^{[i]}, \theta_{n_t}=\mu_{n_t, t-1}^{[i]}} \\
S_{n, t} &= G_{\theta_{n_t}} \Sigma_{n_t, t-1}^{[i]} G_{\theta_{n_t}}^T + R_t \\
K_t &= \Sigma_{n_t, t-1}^{[i]} G_{\theta_{n_t}}^T S_{n, t}^{-1} \\
\mu_{n_t, t}^{[i]} &= \mu_{n_t, t-1}^{[i]} + K_t(z_t - \hat{z}_t) \\
\Sigma_{n_t}^{[i]} &= (I - K_t G_{\theta_{n_t}}) \Sigma_{n_t, t-1}^{[i]}
\end{aligned} \tag{2.41}$$

Burada, K Kalman kazancını, μ , ortalama ve Σ , kovaryansı, ∇ jakobiyeni, G Jakobiyen matrisini, S ise inovasyon kovaryans matrisini temsil etmektedir. Nesnelerin güncelleme işlemi her bir parçacık için sabit zaman işlemi olarak gerçekleştirilir, çünkü nesne süzgeçleri sabit büyüklüğe sahiptir.

- **Önem ağırlıkları hesabı**

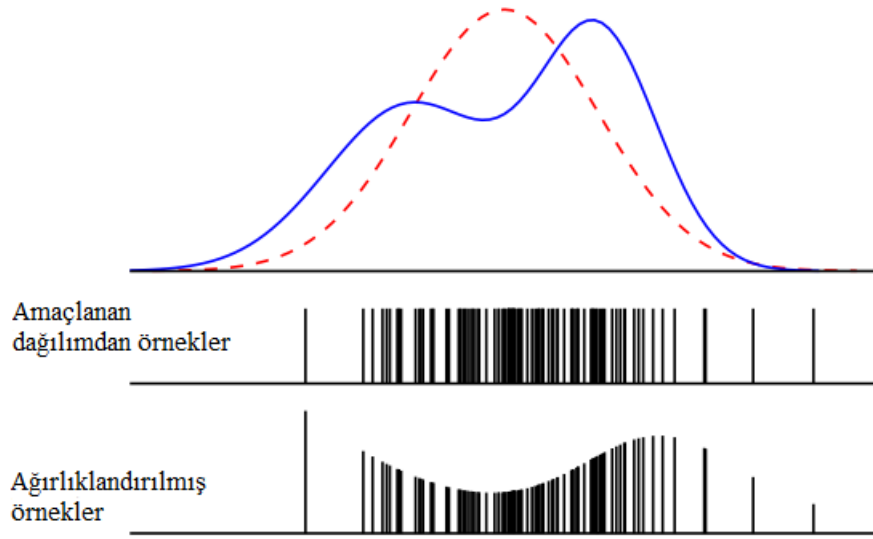
Parçacıklar, hareket modelinden $p(s^t|z^{t-1}, u^t, n^{t-1})$ 'na bağlı olarak olasılık üzerinde dağıldığı için, istenilen soncul dağılımla $p(s^t|z^t, u^t, n^t)$ eşleşmezler. Bu fark önem örnekleme adı verilen işlemle düzeltilir (Montemerlo vd. 2003). Parçacık süzgeçlerinde önem örnekleme, bir fonksiyondan örnekleme yapılan 'teknik' olarak bilinir (Mackay 1998). Hedef fonksiyondan direk olarak örnekleme yerine, örnekler basit bir fonksiyon (the proposal, *Amaç*) yardımıyla örneklenir. Her bir örneğe bir ağırlık atanır, ağırlıkların aynı uzayda olasılıkları birbirine eşit alınır. Yeni ağırlıklandırılmamış örneklerin kümesi ağırlıklara bağlı olarak olasılıklarıyla birlikte ağırlıklandırılmış kümeden oluşturulur. Bu işlem Rubin (1988)' in örnekleme önemi yeniden örnekleme (SIR – Sampling importance resampling) teoremi olarak bilinir (Rubin 1988).

Önem örneklemesine bir örnek şekil 2.14'te verilmiştir. Hedef dağılımdan (mavi sürekli çizgi) direk örnekleme yerine, örnekler daha basit olarak amaçlanan dağılımdan (kırmızı kesikli çizgi) bir Gauss gibi örneklenebilirler. İlgili bölgede hedef, amaçlanan dağılımdan daha büyüktür, böylece örnekler daha yüksek ağırlıklar alırlar. Bu sayede örnekler bu bölgede toplanacaklardır. Bununla birlikte hedef dağılım amaçlanan dağılımdan daha küçük olduğunda daha düşük değerdeki ağırlıklara sahip olacaklardır.

Literatüre bağlı olarak FastSLAM uygulamalarında her bir parçacığın önem ağırlıkları $w_t^{[i]}$ SLAM' in soncul oranına eşittir ve amaçlanan dağılım şu şekilde tanımlar:

$$w_t^{[i]} = \frac{\text{hedef dağılım}}{\text{amaçlanan dağılım}} = \frac{p(s^{t,[i]}|z^t, u^t, n^t)}{p(s^{t,[i]}|z^{t-1}, u^t, n^{t-1})} \quad (2.42)$$

Amaçlanan dağılımdan örneklerin ağırlıklandırılmış örnekleri şekil 2.14'te gösterilmiştir.



Şekil 2.14 Amaçlanan dağılımdan olasılığa bağlı olarak ağırlıklandırılmış örneklerinin temsili gösterimi

Hedef dağılımdan örnekler güvenilir bir şekilde dağılamaz (*mavi çizgili*). Bunun yerine önem örnekleyici (*importance sampling*), amaçlanan dağılımdan örnekleri oluşturabilir (*kırmızı çizgili*) (Montemerlo 2002). Bayes kuralı kullanılarak (2.42) eşitliği genişletilebilir. Normalleştirme sabiti, Bayes kuralında güvenli bir şekilde ihmal edilebilir. Çünkü parçacık ağırlıkları tekrar örneklemeden önce normalleştirilir.

$$w_t^{[i]} \propto (\text{Bayes}) \frac{p(z_t | s^{t,[i]}, z^{t-1}, u^t, n^t) p(s^{t,[i]} | z^{t-1}, u^t, n^t)}{p(s^{t,[i]} | z^{t-1}, u^t, n^{t-1})} \quad (2.43)$$

Payın ikinci tarafı en son gözlemle, z_t , koşullandırılmaz, bu yüzden veri ilişkilendirme n_t , robot/özerk araç pozisyonu hakkında her hangi bir bilgi sağlayamaz. Böylece eşitlik

$$w_t^{[i]} \stackrel{\text{Markov}}{\longleftrightarrow} \frac{p(z_t | s^{t,[i]}, z^{t-1}, u^t, n^t) p(s^{t,[i]} | z^{t-1}, u^t, n^t)}{p(s^{t,[i]} | z^{t-1}, u^t, n^{t-1})} \quad (2.44)$$

$$= p(z_t | s^{t,[i]}, z^{t-1}, u^t, n^t)$$

formuna dönüşür. Burada nesne kestiricisi bir EKF' dir, olasılık gözlemi buradan kapalı bir formda hesaplanabilir. Olasılık genellikle inovasyon tarafından ya da gerçek gözlem

z_t ile tahmin edilen gözlem \hat{z}_t arasındaki farktan hesap edilir (Montemerlo 2002). EKF' de inovasyon sıfır ortalamalı ve kovaryansı $Z_{n_t,t}$ olan Gauss dağılımına sahip seridir (Shalom ve Fortmann 1988). Burada $Z_{n_t,t}$ inovasyon kovaryans matrisini temsil etmektedir. Gözlemin olasılığı z_t Gauss tarafından üretilen inovasyonun $z_t - \hat{z}_t$ olasılığına eşittir. Gözlemin olasılığı

$$w_t^{[i]} = \frac{1}{\sqrt{|2\pi Z_{n_t,t}|}} \exp\left\{-\frac{1}{2}(z_t - \hat{z}_{n_t,t})^T [Z_{n_t,t}]^{-1} (z_t - \hat{z}_{n_t,t})\right\} \quad (2.45)$$

formunda yazılabilir. Hesaplanan önem ağırlığı her bir parçacık için sabit zaman işlemidir. Bu hesaplama gözlemin boyutuna bağlıdır ve verilen uygulamalar için sabit alınmıştır.

- **Önem tekrar örneklemesi**

Geçici parçacıklar (Temporary Particles, TP) ağırlıklara atandıklarında yeni örnekler kümesi S_t , ağırlıklara bağlı olarak olasılıklarıyla birlikte oluşturulur. FastSLAM yapısında Madow (1949)'un örnekleme tekniği tercih edilmiştir. Genel perspektiften bakıldığında Madow' un sistematik örnekleme algoritması (Madow 1949) basit, doğru sonuçlar vermesi ve kolay adapte edilebilir olmasından dolayı en fazla tercih edilen örnekleme yöntemi olarak kullanılır (Montemerlo 2003). Kolayca adapte edilebilmesinin yanı sıra, N adet nesne sayısı için tekrar örnekleme işlemi, doğrusal bir hesaplama zamanına ihtiyaç duyar. Her bir parçacık, yeni parçacık kümesine kopyalanır, burada her bir parçacığın uzunluğu N ile orantılıdır. Toplam nesnelere küçük bir kısmı her hangi bir anda gözlenirse, bütün parçacıkların kopyalanması yerine bu kısım kopyalanır. FastSLAM yaklaşımları bu sayede işlem yükünü $MO(N)$ ' e düşürmüşlerdir (Montemerlo vd. 2002, Montemerlo 2003).

- **Robotun tekrar soncul ziyareti**

İlk bakışta robotun rotasını kullanarak SLAM probleminin $MO(N)$ olarak işlenmesi iyi bir fikir gibi gelmeyebilir, çünkü FastSLAM parçacıklarının uzunluğu zamanla büyüyecektir. Bununla birlikte FastSLAM güncelleme eşitliklerinden hiç birisi, toplam rota uzunluğuna t ' de bağlı değildir. Aslında sadece en son pozisyon $s_{t-1}^{[i]}$ parçacık kümesinin güncellenmesinde hiç kullanılmaz. Sonuç olarak, burada en son robot pozisyonu hariç sessizce diğer pozisyon bilgileri “unut” ulur (Montemerlo 2003).

2.2 SLAM Probleminde Veri İlişkilendirme

Veri ilişkilendirme, SLAM uygulamalarında sıkça karşılaşılan bir problem olarak bilinir. Bazı uygulamalar çevrenin haritasını çıkarırken bilinen veri ile hareket etmişlerdir (Aron 2005, Bailey 2002), diğerleri gerçek zamanlı uygulamalarda etrafın nesne bilgilerinin olmadığı kabulüyle çalışmalar yapmışlardır (Montemerlo ve Thrun 2003, Bailey vd. 2000). Literatürde bu konu ile ilgili yeteri kadar çalışma yapılmış olup farklı veri ilişkilendirme yöntemlerinden yararlanılmıştır (Rex vd. 2010, Ankişhan ve Arı 2013).

SLAM için başarılı bir veri ilişkilendirme, gözlenmiş ölçümün kendisinden elde edilen durumla doğru bir şekilde nesne işaretlerinin ilişkilendirilmesi olarak tarif edilebilir. SLAM uygulamalarında algoritma, yeni nesne işaretlerini doğru bir şekilde tahmin edebilmeli (başlatabilmeli), yanlış ölçümler var ise algılayabilmeli, var olan ölçüm bilgilerini doğru bir şekilde takip edebilmelidir (Rex vd. 2010). Veri ilişkilendirmesinde kullanılan en temel algoritma en yakın komşuluk (Nearest Neighbor-*NN*) algoritmasıdır. Bu algoritma işlem sırasında Mahalanobis mesafesini kullanır. Mahalanobis mesafesi, nesnenin gözlemlenen ölçümü ve hesaplanan pozisyonu arasındaki mesafedir (Mahalanobis 1936). Detayları EK 1’de anlatılmıştır. Algoritma tahmin edilen hedef pozisyonuna en yakın ölçüm değerini gerçek değer olarak kabul eder. Algoritma, bir sonraki nesne işareti yenilemesi için ölçüm kestirimine göre bir kabul bölgesi oluşturur, kabul bölgesi kapı olarak adlandırılır. Fakat *NN*

süzgecinde ölçüm ilişkili en yakın nesneden gelmeyebilir, süzgeç bu durumla ilgilenmez. Bu yüzden durum vektörü güncellenirken böyle durumlarda süzgeç kestiriminde ıraksama gerçekleşebilmektedir. Aynı zamanda dinamik çevrelerde performansının iyi olmadığı gözlenmiştir (Rex vd. 2010).

Birbirine yakın nesne sayısının çok olduğu ve yüksek gürültülü durumlarda *NN* algoritmasının performansının iyi sonuçlar vermediği daha önceki çalışmalarda gözlenmiştir (Rex vd. 2010, Montemerlo vd. 2002). Bunun sebebinin kapı içerisindeki ölçümlerden en düşük mesafedeki kestirim değerinin gerçek ölçüm olarak anılmasıdır. *NN* algoritmasında sonuca çabuk varmak için işlem süresi kısaltılırken tüm verilerin kullanılmaması yapıyı optimal durumdan uzaklaştırır. Olasılıksal veri ilişkilendirme (Probabilistic Data Association, PDA) algoritması kapı içindeki bütün ölçümlerin hedef nesne olma olasılıklarını hesaplar. PDA, bir birleşik inovasyon değeri hesaplar, bunun ile belirsizlik problemini olasılıksal olarak çözmeye çalışır. Hipotezlerin ilişkisel olasılığını kullanarak birleşik bir inovasyon sağlar ve bununla birlikte birleşik geçerlilik kapısını (Validation Gate, VG) oluşturur. Algoritma çalışma esnasında her bir hedefi bağımsız olarak ele alır ve komşu hedeflerle karışmadığını farz eder. Öte yandan SLAM probleminde nesne işaretlerinin birbiriyle korelasyonu mevcuttur. Bu hedef nesnelerin karşılıklı iletişim halinde olmasından ötürü SLAM için PDA bir geri adım olarak algılanır (Rex vd. 2010). Bu yüzden çoklu nesne, hedef izleme ve dinamik durumlar için PDA algoritmasının uygun olmadığı düşünülmektedir (Rex vd. 2010), ya da algoritma her bir nesne işareti için tekrar koşturulmalıdır. SLAM uygulamalarında PDA'ya alternatif olarak bileşik PDA (Joint Probabilistic Data Association, JPDA) algoritmasının kullanılması düşünülmüştür. JPDA, ortamda bulunan tüm nesne işaretlerinin PDA mantığıyla bir döngüde takip edilebilmesi için geliştirilmiş bir algoritmadır.

Literatürde SLAM uygulamalarında veri ilişkilendirme problemine yönelik JPDA algoritmasını kullanan çalışmalar mevcuttur (Rex vd. 2010, Ankişhan ve Arı 2013). Rex (2010) ve arkadaşları çalışmalarında kablosuz sensör şebekelerindeki SLAM problemi için bu algoritmayı önermiş ve 3- tarama JPDA yöntemini kullanmışlardır. Gürültülü sensör bilgisine dayanarak ve olası parazit yankılara bağlı olarak gelen sinyalin

gürültülü olduğunu ve bunun da pozisyon ve açı ölçümünde belirsizliğe yol açtığını söylemişlerdir. (Rex vd. 2010) çalışmalarında hareketli parazit yankılı ortamda ölçümleri parazit yankıdan ayırmanın zor olduğunu çünkü yeni nesne işaretlerinin belirsizliğe yol açtığını göstermişlerdir. Diğer bir JPDA temelli yaklaşım ise Zhou ve Bose (1993) tarafından önerilmiştir. Zhou ve Bose (1993) önerdiği Fast- JPDA yöntemi ile birlikte “*Depth-first- search*” (DFS) yöntemini birleştirmişlerdir. Fakat çoklu hipotez izleme (Multiple Hypothesis Tracking, MHT) yönteminde olduğu gibi geri izleme temelli bir yöntem olduğu için veri ilişkilendirme hipotezleri burada geniş hesaplama maliyeti getirmiştir. Bu çalışmalar genel olarak SLAM probleminin veri ilişkilendirme sorununu çözmek için geliştirilen yaklaşımlar olmuşlardır, kullanılan algoritmalarla birlikte süzgeçlerin performanslarına değinilmemiştir. Sadece belirsizlik esnasında bazı karşılaştırmalar yapılmıştır.

2.2.1 FastSLAM için JPDA

Çalışmanın bu bölümünde JPDA algoritmasının FastSLAM yöntemine uyarlanması üzerinde durulmuştur. Kalman süzgeçlerinde ölçüm tahmini ve ölçüm farkı inovasyon vektörünü vermektedir.

$$v = z_t - \hat{z}_t^{[m]} \quad (2.46)$$

İnovasyon kovaryansı ise,

$$S = h(\hat{x}_{t-1}) \sum_v [h(\hat{x}_{t-1})]^T + Q_t \quad (2.47)$$

olarak tanımlanır (Kalman 1960). Burada $h(\hat{x}_{k/k-1})$ ölçüm fonksiyonu, Σ_v durum kovaryansı ve Q_t beyaz gürültü kovaryansını göstermektedir. İstatistiksel geçerlilik kapısı, gözlem ve en yakın ölçüm arasındaki eşleştirmeyi sağlar (Zhou ve Bose 1993). Bu tanımdan yola çıkarak Mahalanobis mesafesi (ya da normalleştirilmiş inovasyon karesi- *NIS*),

$$M_x = v^T S^{-1} v \leq \gamma_n \quad (2.48)$$

olarak tanımlanır (Rex vd. 2010). Normalleştirilmiş İnovasyon Karesi (NIS) serbestliğinin N boyutu ile Chi-Kare (χ^2) dağılımına sahiptir. γ_n kapı eşik değeridir. NN süzgeci, ölçümün gelen sinyalinin geçerliliğini test etmek için bu eşik değerini kullanır. Bütün olası ölçümler arasında, hangisi nesnenin tahmin edilen en yakın değerine sahipse o geçerli ölçüm olarak atanır. Eğer çoklu hedeflerde NN süzgecinde kapılar üst üste çakışırsa, sensör ölçümleri birbirinden bağımsız olsa bile belirsizlik durumu ortaya çıkar, süzgeç bu durumda doğru veri ilişkilendirmesi yapamayabilir. Öznitelik tabanlı SLAM uygulamalarında bütün algılayıcı ölçümleri birbirinden bağımsız olsalar bile, tahmin edilen ölçümler robotun benzer pozisyon hatası tarafından istatistiksel olarak birbiriyle ilişkilidir. Bu durumdan, PDA çoklu hedef tahmini durumunu çözemez, ya da tekrarlı olarak her bir hedef için süzgeç algoritması tekrar koşturulur. Algoritmanın tekrar koşturulması sistemin çok yavaşlamasına yol açar, sisteme ekstra işlem yükü katar. Bu yüzden çoklu hedef kestirimi problemini çözmek için JPDA algoritmaları önerilmiştir.

Hedef nesnelerin kümesi t anında $X_t = \{x_1, x_2, \dots, x_n\}$ olarak düşünülürse, ilgili zamanda bu küme ile ilişkilendirilen geçerli ölçümlerin toplamı m olarak kabul edilir (Optimizasyon algoritmalarında kullanılan parçacık indeksinden farklıdır.) ve t anındaki ölçümler kümesi $Z_t = \{z_1, z_2, \dots, z_M\}$ olarak yazılır. Burada her bir ölçüm ya hedeften ya da parazit yankıdan gelmektedir. Parazit yankı x_0 olarak yazılır. Tahmin edilen hedef ölçümü $\tilde{z}_{t,m}^x$, ölçüm m ' in inovasyon vektörü ise

$$\tilde{z}_{t,m}^x = z_{t,m} - \hat{z}_{t,x} \quad (2.49)$$

olarak yazılır (Shalom vd. 2009). Her bir x hedefi için birleştirilmiş (ağırlıklandırılmış) inovasyon,

$$\tilde{z}_t^m = \sum_{m=1}^M \beta_{t,m}^x \tilde{z}_{t,m}^x \quad (2.50)$$

formülüyle gösterilir (Shalom vd. 2009). Burada β_m^x , x hedefinden gelen ilişkilendirilmiş olasılığı göstermektedir ve β_0^x ise t anındaki ölçümlerin hiçbirisinin hedeften gelmediği olasılığı gösterir.

$$\beta_{t,m}^x = \sum_{\phi} P\{\Phi_t | Z^t\} a_{t,m}^x(\Phi) \quad (2.51)$$

$$\beta_0^x = 1 - \sum_{m=1}^M \beta_{t,m}^x \quad (2.52)$$

burada $m=1,2,\dots,M$; $x=0,1,\dots,N$

$$a_{t,m}^x(\Phi) = \begin{cases} 1 & \text{ölçüm hedeften gelmektedir} \\ 0 & \text{ölçüm hedeften gelmemektedir} \end{cases} \quad (2.53)$$

burada Φ_t ilgili zamanda birleşik ilişkisel olayları gösterir ve

$$\Phi_t = \bigcap_{m=1}^M \phi_{t,m}^x \quad (2.54)$$

olarak yazılır, $\phi_{t,m}^x$ bireysel ilişkisel olayı gösterir, \bigcap olayların kesişimini gösterir ve $a_{t,m}^x(\Phi)$ geçerlilik matrisindeki ölçüm z^t ve hedef x^t arasındaki ilişkisel hipotezi verir,

$$\Omega = [a_{t,m}^x(\Phi)] \quad (2.55)$$

$$\Omega = 1 \begin{matrix} x_0 & x_1 & x_2 & \cdot & \dots & x_N \\ \left(\begin{array}{cccccc} a_1^1 & a_1^2 & \dots & a_1^N \\ a_2^1 & a_2^2 & \dots & a_2^N \\ \vdots & \vdots & \vdots & \vdots \\ a_M^1 & a_M^2 & \dots & a_M^N \end{array} \right) \begin{matrix} z_1 \\ z_2 \\ \vdots \\ z_M \end{matrix} \end{matrix} \quad (2.56)$$

Eşitlik (2.51)' den, t anındaki tüm ölçümlerin birleşik olaylarının olasılığı eşitlik (2.57)' de verilmiştir.

$$P\{\Phi_t | Z^t\} = P\{\Phi_t | Z^t, Z^{t-1}\} = \frac{1}{c} p(Z^t | \Phi_t, Z^{t-1}) P\{\Phi_t\} \quad (2.57)$$

Buradaki normalizasyon sabiti c ölçümlerin birleşik öncül yoğunluğu ve Φ_t ' deki bütün değerlerin toplamını göstermektedir. Birleşik olasılık yoğunluk fonksiyonu,

$$p(Z^t | \Phi_t, Z^{t-1}) = \prod_{m=0}^M p[z_{t,m} | \phi_{t,m}^x, Z^{t-1}] \quad (2.58)$$

x hedefi ile ilişkilendirilen m ölçümü, Gauss yoğunluğuna bağlı olduğundan

$$p[z_{t,m} | \phi_{t,m}^x, Z^{t-1}] = \begin{cases} N(z_{t,m}; \hat{z}_{t,m}^x, S_{t,m}^x) & \text{eğer } a_{t,m}^x(\Phi) = 1 \\ A^{-1} & \text{eğer } a_{t,m}^x(\Phi) = 0 \end{cases} \quad (2.59)$$

normal yoğunluk dağılım fonksiyonu,

$$N(z_{t,m}; \hat{z}_{t,m}^x, S_{t,m}^x) = P_G^{-1} (2\pi S_{t,m}^x)^{1/2} e^{-\frac{1}{2} \hat{z}_{t,m}^x (S_{t,m}^x)^{-1} (\hat{z}_{t,m}^x)^T} \quad (2.60)$$

gösterilir. Burada P_G , doğru ölçümlerin olasılığı, m , x hedef kapısının içerisindeki ölçüm, $\hat{z}_{t,m}^x$ ve $S_{t,m}^x$ inovasyon vektörü ve kovaryansıdır. Ölçümler eğer her hangi bir hedefle ilişkilendirilmezse o zaman ölçümler ilgili kapının dışında olacaktır ve hesaba katılmayacaktır. (2.57) eşitliğindeki ikinci faktör birleşik olayların öncül olasılığını vermektedir. Burada, yanlış alarmların toplam sayısı m_0 olarak tanımlanır. Doğru ölçümlerin sayısı $m_c = M - m_0$ olarak verilir, M burada toplam geçerlilik alanı içerisindeki ölçümlerin sayısını gösterir. Buradan öncül olasılık,

$$P\{\Phi_t\} = P\{\Phi_t | \delta(\Phi), \varphi(\Phi)\} P\{\delta(\Phi), \varphi(\Phi)\} \quad (2.61)$$

ifade edilir. Burada $\delta(\Phi)$ ikili düzende hedef algılama göstericisidir (1 ya da 0).

$$\delta_x(\Phi) = \sum_{m=1}^M a_{t,m}^x(\Phi) \leq 1 \quad x = 1, \dots, N \quad (2.62)$$

ve $\varphi(\Phi)$ olaydaki yanlış ölçümlerin sayısını vermektedir. Bütün hedefler ikili gösterici ilişkisine bağlı olarak tanımlanırsa $\tau_m(\Phi)$ ilgili olayda m ölçümü ile bütün hedeflerin ilişkisini göstermektedir.

$$\tau_m(\Phi) = \sum_{x=1}^N a_{t,m}^x(\Phi), m = 1, \dots, M \quad (2.63)$$

$$\varphi(\Phi) = \sum_{m=1}^M [1 - \tau_m(\Phi)] \quad (2.64)$$

(2.61)' nin ikinci terimi;

$$P\{\delta(\Phi), \varphi(\Phi)\} = \prod_{t=1}^N (P_D)_t^\delta (1 - P_D)_t^{1-\delta} P_F(m_0) \quad (2.65)$$

ifade edilir. Burada P_F yanlış alarmlar için Poisson olasılık yoğunluk fonksiyonunu göstermektedir (Shalom vd. 2009),

$$P_F(m_0) = e^{-\lambda A} \frac{(\lambda A)^{m_0}}{m_0!} \quad (2.66)$$

burada λ yanlış ölçümlerin uzaysal yoğunluğu ve A geçerlilik bölgesinin alanını vermektedir. Eşik değerinin de işleme katılmasıyla oluşturulan kapı bölgesinin alanı

$$A = \pi |\gamma S_t|^{1/2} \quad (2.67)$$

olarak verilir. Buradan eşitlik (2.61)

$$P\{\Phi_t\} = \frac{m_0!}{M!} e^{-\lambda A} \frac{\lambda A}{m_0!} \prod_{t=1}^N (P_D)_t^\delta (1 - P_D)_t^{1-\delta} \quad (2.68)$$

ölçüm eğer her hangi bir hedefle ilişkilendirilmezse iki boyutlu uzayda gözlem A alanında düzgün dağılıma sahiptir denilir. Yanlış alarmlar için düzgün yoğunluk fonksiyonu m_0 ' in üstü olarak $A - m_0$ gibi tanımlanır (Rex vd. 2010). (2.59) ve (2.60) birleştirilerek (2.58) tekrar elde edilirse,

$$p(Z^t | \Phi_t, Z^{t-1}) = A^{-m_0} \prod_{m=1}^M N(z_{t,m}; \hat{z}_{t,m}^x, S_{t,m}^x) \quad (2.69)$$

(2.68) ve (2.69) ile (2.57) tekrar yazılırsa,

$$P\{\Phi_t | Z^t\} = \frac{\lambda^{-m_0}}{c} \prod_{m=1}^M N(z_{t,m}; \hat{z}_{t,m}^x, S_{t,m}^x)^{\tau} \times \prod_{t=1}^N (P_D)_t^{\delta} (1 - P_D)_t^{1-\delta} \quad (2.70)$$

elde edilir.

3. MATERYAL ve YÖNTEMLER

Tez çalışması kapsamında bahsedilen FastSLAM yöntemlerinin üstünlükleri olduğu gibi bazı problemleri de mevcuttur. EKF tabanlı FastSLAM yaklaşımlarının tekrar örnekleme esnasında parçacıklarının çoğunun ağırlıkları zamanlara sıfıra yaklaşmaktadır ve bu yüzden dejenerasyona problemi ortaya çıkmaktadır. Bu problem EKF' den kaynaklı geçiş matrisi ve ölçüm matrislerinin jakobiyenlerinin hesaplanarak doğrusallaştırılmasının bir sonucudur. Diğer bir problem ise parçacık süzgeçlerindeki parçacık sayısı seçimidir. Yüksek miktarda parçacık kullanıldığı zaman işlem yükü problemi ortaya çıkmaktadır, az seçildiğinde ise kestirim performansı düşmektedir.

Daha önce bahsedilen (Kim vd. 2008)' nin yapmış oldukları çalışmada FastSLAM II yönteminin eksikliklerinden bahsedilmiş ve EKF yerine UKF tabanlı süzgeçlerle FastSLAM modelinde iyileştirmeler yapılmıştır. UT'le, EKF tabanlı süzgeçlerde doğrusallaştırma sonucu ortaya çıkan problemlerin yeni FastSLAM yapısıyla çözüldüğü gösterilmiştir. Bunun yanı sıra (Kim vd. 2008, 2009), UKF tabanlı FastSLAM yaklaşımının parçacık sayısı problemine de çözüm getirdiğini deneysel çalışmalarında vurgulanmış ve örneklerle açıklamışlardır. Ama literatürden görülmektedir ki U-FastSLAM yaklaşımında da bazı problemler olduğu görülmektedir. Bunlar (Ankışhan vd. 2014)'de şu başlıklar altında gösterilmiştir:

- Sigma noktaları hesaplanırken üç adet ölçekleme parametresinin her farklı Gauss durum için tekrar ayarlama sorunu,
- Parçacık sayısı seçimi probleminin UKF tabanlı yaklaşımlarda bahsedildiği gibi çözüm oluşturamaması,
- Hata kovaryans matrisinin modelden kaynaklı pozitif yarı tanımlı olma zorunluluğu,
- Hata kovaryans matrisinin süzgecin çalışması esnasında, her güncelleme adımında karekökünün alınması gerekliliğidir.

(Ankışhan vd. 2014) EKF ve UKF tabanlı FastSLAM yaklaşımlarındaki problemleri gözlemleyerek literatüre CDKF tabanlı iyileştirilmiş bir FastSLAM algoritması sunmuşlardır. Çalışmalarında CDKF'nin yanı sıra doğadan ilham alan optimizasyon algoritmalarından da yararlanarak kestirim performansını optimize etmeye çalışmışlardır. Ankışhan (2014) ve arkadaşlarının yanı sıra SLAM uygulamalarında optimizasyon algoritmalarını kullanan bazı çalışmalar da vardır (Song vd. 2012, Dongbo vd. 2012).

3.1 FastSLAM metotlarında optimizasyon algoritmaları

FastSLAM yaklaşımlarında parçacık süzgeçlerinin işlem yükü fazlalığı problemine doğadan ilham alan optimizasyon yöntemleriyle çözüm bulmaya çalışan bazı çalışmalar mevcuttur (Ankışhan vd. 2014, Song vd. 2012, Dongbo vd. 2012). Ankışhan (2014) ve arkadaşları farksal evrim (DE) algoritmasını parçacık süzgeçlerindeki parçacıkları iyileştirmek için kullanmışlardır. Burada kullanılan DE, basit ve etkili bir evrimsel optimizasyon algoritmasıdır, FastSLAM uygulamalarında olasılığı düşük parçacıkları olasılığı yüksek olan parçacıkların olduğu bölgeye kaydırarak daha az sayıda parçacıkla doğruluğu daha yüksek sonuçlar elde edilmesini sağlar. DE'nin yanı sıra parçacık sürü optimizasyonu (PSO) ve genetik algoritma (GA) temelli yaklaşımlarda mevcuttur (Havangi vd. 2010, Yi-min ve Yi-min 2011, Dongbo vd. 2012, Ankışhan vd. 2014). Bu yaklaşımlar belirlenen bir maliyet fonksiyonu sayesinde parçacıkları olasılığı düşük bölgeden olasılığı yüksek bölgelere kaydırarak parçacıkları yine DE'de olduğu gibi iyileştirmeye çalışmışlardır. Fakat bu çalışmalar, optimizasyon algoritmalarının gerçek zamanlı sistemlerde, yöntemlerin kendi iç iterasyonundan kaynaklı olarak, uygulanmasının pek mümkün olmadığını göstermişlerdir. Yöntemler parçacık sayısı problemini çözmüşlerdir. Öte yandan, optimizasyon yöntemleri her bir adım esnasında parçacıkları yüksek olasılıklı bölgeye yaklaştırmak için tanımlanan ölçütte iterasyon kullanırlar. Sonuçta yine bir zaman kaybı oluşur. 'Off-line' çalışmalarda çözüm getirebilse de 'On-line' çalışmalarda istenilen performansı gerçekleştiremedikleri görülmektedir.

3.1.1 FastSLAM için parçacık sürü optimizasyonu (PSO) algoritması

Algoritma adımında, D boyutlu bir arama uzayı için m' nci parçacığın pozisyonu $X_m = [x_{m1}, x_{m2}, \dots, x_{mD}]$, her bir parçacığın hızı $V_m = [v_{m1}, v_{m2}, \dots, v_{mD}]$ olarak gösterilir. Her bir parçacığın hız ve pozisyonu ilgili aralıkta rasgele vektörler tarafından başlatılır (Kennedy ve Eberhart 1997, Shi ve Eberhart 1998). PSO algoritması iteratif bir şekilde çalışır, bütün hızlar ve bütün parçacık pozisyonlarını maliyet fonksiyonunun değerine göre takip eden eşitliklerde olduğu gibi güncellenir:

$$\begin{aligned} v_{md} &= wv_{md} + c1rand1()*(p_{md} - x_{md}) + c2rand2()*(p_{gd} - x_{md}) \\ x_{md} &= x_{md} + v_{md} \end{aligned} \quad (3.1)$$

İvmelendirme katsayıları $c1$, $c2$ ve atalet ağırlığı w kullanıcı tarafından veya sistemsel olarak belirli aralıklarda tanımlanabilir. $rand1()$ ve $rand2()$, 0 ile 1 arasında düzgün olarak rasgele üretilmiş sayıları temsil eder. Hız denkleminde p_{md} m' nci parçacığın en iyi maliyet fonksiyonu değerini elde ettiği pozisyonudur. p_{gd} ise bütün parçacıkların içindeki en iyi maliyet fonksiyonu değerini veren en iyi pozisyonu simgeler.

3.1.1.1 PSOPC tabanlı parçacık süzgeçleri

FastSLAM uygulamalarında RBPF için PSO algoritması, parçacıklardaki çeşitliliği artırmak için ve parçacıkların optimal değerlere ulaşması için kullanılmıştır (Ankışhan vd. 2013). Parçacık süzgeçlerini optimize etmek, ya da diğer bir deyişle parçacıkların iyileştirilmesini sağlamak için doğadan ilham alan optimizasyon yöntemleri ya da evrimsel algoritmalarından yararlanılabilir. Bu yöntemlerdeki başlıca amaç; parçacıklardan olasılıkları düşük olanları yüksek olasılıklı bölgeye kaydırarak zamanla dejenerasyona uğramalarını engellemektir. SLAM uygulamalarında bu algoritmalar iteratif olarak uygulanmaktadır. Uygunluk fonksiyonunun (Fitness Function, FF) tanımlanması, durum kestirimi PSO ile parçacıkların iyileştirilmesi, parçacıkların

ağırlık hesabı ve tekrar örnekleme aşaması olarak gerçekleşmektedir. Uygunluk fonksiyonu tanımı;

$$Uyg.fonksiyonu = \exp\left\{-\frac{1}{2R_t}(z_t - \hat{z}_t^{[i]})^2\right\} \quad (3.2)$$

olarak verilmektedir (Havangi vd. 2010, Ankişhan vd. 2014). Burada R_t araç ölçüm gürültü varyansı, z_t t anındaki ölçüm bilgisi ve $\hat{z}_t^{[i]}$ ise t anındaki kestirilen ölçümdür. Bütün parçacıklar hesaplanan uygunluk fonksiyonuna bağlı olarak düşük olasılıklı bölgeden yüksek olasılıklı bölgeye kaydırılmaktadır. PSOPC (pasif toplanma, passive congregation – PC) yöntemi, daha önceki çalışmalarda yararlı ve durağan sonuçlar verdiği için tez çalışmasında SLAM için kullanılmıştır (He vd. 2004). Algoritmanın burada başlangıç hız değeri sıfır alınmıştır. Algoritmada parçacık süzgeçlerinin başlangıç değerleri sürünün başlangıç pozisyonlarına atanır. Dağılıma bağlı olarak uygunluk fonksiyonu, iteratif yapısı sayesinde örnekleme adımında parçacıkları iyileştirir. Parçacık ağırlıkları son gözleme bağlı olarak

$$W_t^{[i]} = \left|2\pi B_t^{[i]}\right|^{-1/2} \exp\left\{-(1/2)(z_t - \hat{z}_t^{[i]})^T * (B_t^{[i]})^{-1} (z_t - \hat{z}_t^{[i]})\right\} \quad (3.3)$$

formülünden hesaplanır. Burada $B_t^{[i]} = (P_t^{x,L,[i]})^T (\Sigma_t^{[i]})^{-1} P_t^{x,L,[i]} + S_t^{[i]}$ olarak gösterilir. Tekrar örnekleme adımı CDKF ve UKF tabanlı süzgeçlerde kullanılan FastSLAM modeliyle aynıdır.

3.1.2 FastSLAM için farksal evrim (DE) algoritması

Tez çalışmasında, PSOPC' nin yanı sıra farksal evrim (DE) algoritması da FastSLAM modeline uygulanmıştır. Literatür incelendiğinde DE' nin PSO ve GA gibi yöntemlere nazaran bazı avantajları vardır. Örneğin PSOPC'ye göre kendi iç yapısında daha az sayıda parçacığa ihtiyaç duyar. Popülasyondaki çeşitliliği artırmak ve doğru sonuca yakınsamak için diğer yöntemlerden daha az parçacıkla iterasyon işlemini gerçekleştirir

(Vesterstrom vd. 2004). Karşılaştırmalı çalışmalarda yakınsama özelliğinin daha başarılı olduğu vurgulanmıştır (Das 2011). Yöntem dört aşamaya sahiptir:

- **Başlangıç**

Diğer popülasyon tabanlı algoritmalarından olduğu gibi DE, popülasyonu arama uzayına dağıtır. Arama uzayı D boyutundadır. DE' de, g . iterasyondaki popülasyon P_g , N_p adet hedef vektöre sahiptir ve ilk jenerasyon hedef vektörleri

$$x_{j,i,0} = b_{j,l} + rand_j[0,1).(b_{j,u} - b_{j,l}) \quad (3.4)$$

denklemleriyle başlatılır. Burada $j=1,2,\dots,D$ vektörün boyutunu, $i=1,2,\dots, N_p$ vektör numarasını gösterir.

- **Mutasyon**

Popülasyondaki çeşitliliği artırmak için DE mutasyon ve çaprazlama olarak iki operatör kullanır. Mutasyon operatörü, bir mutant vektörü elde etmek için farklı vektörleri kullanarak basitleştirilir. Bu işlem mutasyon operasyonunda kullanılan temel vektöre bağlı olarak değişkenlik gösterebilir. DE' nin değişkenleri DE/x/y/z şeklinde gösterilebilir. Burada x temel vektörü, y vektörlerde kullanılan fark sayısını ve z ise çapraz metodu göstermektedir. Klasik gösterimde DE/rand/1/bin mutant vektörü

$$v_{[m],g} = x_{r_0,g} + F.(x_{r_1,g} - x_{r_2,g}) \quad (3.5)$$

formülüyle gösterilir. Burada r_0 , r_1 ve r_2 rasgele seçilen fark indekslerini ve hedef indeksten farklılığı gösterir. Ölçekleme faktörü $F \in (0,1+)$ pozitif bir sayıdır ve genelde 1' den küçük alınır. Tez çalışması simülasyonlarında en iyi vektör, daha hızlı yakınsama için temel vektör olarak kullanılmıştır. Dolayısıyla mutant vektör

$$v_{[m],g} = x_{best,g} + F.(x_{r1,g} - x_{r2,g}) \quad (3.6)$$

formülüyle ifade edilir.

- **Çaprazlama**

Çaprazlama esnasında mutant vektörü, çaprazlama operatörü tarafından, hedefteki vektörü tekrar birleştirmek için kullanılır. İki vektörün karıştırılmış parametreleri

$$u_{i,g} = \begin{cases} v_{j,i,g}, & \text{if } (rand_j[0,1] \leq C_r \text{ or } j = j_{rand}) \\ x_{j,i,g} & \text{otherwise} \end{cases} \quad (3.7)$$

olarak verilir. $u_{i,g}$ geçici vektördür.

- **Seçim aşaması**

Sonuncu olarak seçim işlemi, geçici vektörün uygunluk fonksiyon değeri, hedef vektörler için basitçe karşılaştırılır ve en iyi vektör gelecek nesil için hedef vektör olarak kaydedilir. Bu iterasyonla N_p adet vektör için işlem gerçekleştirilir ve güncellenen popülasyon, bir jenerasyonun sonunda gelecek jenerasyon için hazır olur. Bundan sonraki işlemler bir önceki bölümde anlatılan PSOPC modelinde olduğu gibi FastSLAM uygulamalarında devam eder.

3.1.3 JPDA uyarlı merkez fark Kalman süzgeci tabanlı FastSLAM algoritması

Merkez fark Kalman süzgeci (CDKF), (Merwe 2003) tarafından önerilen UKF'ye alternatif olarak çıkarılan sigma nokta tabanlı bir algoritmadır. Literatürde farklı uygulamaları olduğu bilinmektedir. Çalışma kapsamında CDKF'nin uyarlı bir versiyonu FastSLAM yapısına adapte edilmiştir. Çalışmanın algoritması üç aşamada CDKF'yi kullanmaktadır; robot/özerk araç kestirimi, nesne pozisyon kestirimi ve önem ağırlıklarının hesaplanması olarak tanımlayabiliriz.

3.1.3.1 Robot/özerk araç durum hesabı

Önem örneklemesi ve nesnelere bağlı olarak harita oluşturulması RBPF tabanlı FastSLAM uygulamaları için önem arz etmektedir. Amaçlanan dağılımın oluşturulması için doğru soncul dağılıma ihtiyaç vardır. Eğer amaçlanan dağılım istatistiksel dağılımdan daha büyük olursa amaçlanan dağılımın gerçek dağılıma yakınsaması doğru olmayabilir. Literatürdeki bazı deneysel teknikler amaçlanan dağılımın iyileştirilmesi için kullanılmıştır (Davison ve Murray 2002, Bailey 2002, Norgaard vd. 2000, Mohammad ve Schwarz 1999). EKF ve UKF tabanlı FastSLAM yaklaşımlarının bazı eksikliklerinden daha önce bahsedilmişti. Bu çalışmada CDKF tabanlı bir FastSLAM yaklaşımının uyarlı hale getirilmesi ve veri ilişkilendirme için JPDA temelli bir yöntemin kullanılarak sistemde doğru bir veri ilişkilendirme gerçekleştirilmesi amaçlanmıştır.

Doğru bir önem örneklemesi ve harita oluşturulması için doğru bir araç pozisyon kestirimi şarttır. İyi bir robot/özerk araç pozisyon kestirimi için uyarlı bir CDKF yaklaşımının iyi olacağı Ankişhan (2014) ve arkadaşlarının' nın çalışmalarında görülmektedir. Robot/özerk araç durum kestirimi tahmin ve güncelleme adımlarından oluşmaktadır.

- **Tahmin aşaması**

Artırılmış robot/özerk araç durum vektörü, kovaryans matrisi kontrol girişleri ve gözlemler kullanılarak (3.8)' deki gibi hesaplanır.

$$x_{t-1}^{a,[i]} = \begin{bmatrix} x_{t-1}^{[i]} & 0 & 0 \end{bmatrix}^T = \begin{bmatrix} x_{x,t-1}^{[i]} & x_{y,t-1}^{[i]} & x_{\phi,t-1}^{[i]} & 0 & 0 \end{bmatrix}^T$$

$$P_{t-1}^{a,[i]} = \begin{bmatrix} P_{t-1}^{[i]} & 0 & 0 \\ 0 & Q_t & 0 \\ 0 & 0 & R_t \end{bmatrix} \quad (3.8)$$

Burada $x_{t-1}^{a,[i]}$ genişletilmiş durumu ve $x_{t-1}^{[i]}$ ise i ' nci parçacığın durumunu vermektedir. CDKF' de sigma noktaları ölçekleme parametresi h yardımıyla hesap edilir. Simetrik olarak dağılmış $2N+1$ adet sigma noktası ölçeklenebilir kokusuz dönüşüm (Scaled Unscented Transform, SUT)' e benzer şekilde robotun/özerk aracın eklenmiş durum vektörü için

$$\begin{aligned}\chi_{t-1}^{a(0),[i]} &= x_{t-1}^{a,[i]} \\ \chi_{t-1}^{a(k),[i]} &= x_{t-1}^{a,[i]} + (hS_{t-1}^{a,[i]})_k \quad (k = 1, \dots, N) \\ \chi_{t-1}^{a(k),[i]} &= x_{t-1}^{a,[i]} - (hS_{t-1}^{a,[i]})_k \quad (k = N+1, \dots, 2N)\end{aligned}\quad (3.9)$$

olarak verilir. k bir matrisin k ' nci sütununu göstermektedir. Her bir sigma noktası $\chi_{t-1}^{a(k),[i]}$ durum, kontrol ve ölçüm komponentlerini göstermektedir,

$$\chi_{t-1}^{a(k),[i]} = \left[\chi_{t-1}^{(k),[i]} \quad \chi_t^{u(k),[i]} \quad \chi_t^{z(k),[i]} \right]^T \quad (3.10)$$

CDKF yaklaşımında da SUT' de olduğu gibi doğrusal olmayan bir fonksiyona yaklaştırılır.

$$\bar{\chi}_t^{(k),[i]} = f(u_t^{[i]} + \chi_t^{u(k),[i]}, \chi_{t-1}^{(k),[i]}) = \left[\bar{\chi}_{x,t}^{(k),[i]} \quad \bar{\chi}_{y,t}^{(k),[i]} \quad \bar{\chi}_{\phi,t}^{(k),[i]} \right]^T \quad (3.11)$$

CDKF' de soncul ortalama, aracın çapraz kovaryansı ve dönüştürülen sigma noktalarının ağırlıklandırılmış regresyonları Stirling's polinomsal interpolasyon (SPI) metodu yardımıyla hesaplanır.

$$\begin{aligned}x_{t/t-1}^{[i]} &= \sum_{k=0}^{2N} w_g^{(k)} \bar{\chi}_t^{(k),[i]} \\ S_{t/t-1}^- &= qr \left\{ \left[\begin{array}{l} \sqrt{w_{c_1}^{(k)}} (\bar{\chi}_{t,1:N}^{(k),[i]} - \bar{\chi}_{t,N+1:2N}^{(k),[i]}) \\ \sqrt{w_{c_2}^{(k)}} (\bar{\chi}_{t,1:N}^{(k),[i]} + \bar{\chi}_{t,N+1:2N}^{(k),[i]} - 2\bar{\chi}_{t,0}^{(k),[i]}) \end{array} \right] \right\} \quad (3.12)\end{aligned}$$

Burada $w_g^{(k)}$, soncul ortalamanın hesaplanmasında kullanılır ve diğer ağırlık $w_{c_i}^{(k)}$ kovaryans ve çapraz kovaryansın hesaplanmasında kullanılır. Bu ağırlıklar (3.13)' de verilmiştir.

$$\begin{aligned} w_g^{(0)} &= \frac{h^2 - L}{h}, w_g^{(k)} = \frac{1}{2h} \\ w_{c_1}^{(k)} &= \frac{1}{4h^2}, w_{c_2}^{(k)} = \frac{h^2 - 1}{4h^2}, \quad k = 1, \dots, 2N \end{aligned} \quad (3.13)$$

- **Güncelleme aşaması**

Güncelleme aşamasında sigma noktaları,

$$\begin{aligned} \chi_t^{a(k),[i],j} &= [x_t^{a(k),[i],j}, x_t^{a(k),[i],j} \pm (hS_{t-1}^{a(k),[i],j})_k] \\ \bar{\chi}_t^{(k),[i],j} &= \chi_t^{(k),[i],j} \end{aligned} \quad (3.14)$$

olarak ifade edilir. Eğer ölçüm bilgileri mevcut ise araç durumu, kovaryans ve çapraz kovaryans her bir nesne için ilişkilendirilerek güncellenir. Ölçüm eşitliği

$$\begin{aligned} \bar{Z}_t^{(k),[i]} &= h(\bar{\chi}_t^{(k),[i]}, \mu_{\hat{m},t-1}^{[i]}) + \bar{\chi}_t^{z(k),[i]} \\ \hat{z}_t^{[i]} &= \sum_{k=0}^{2N} w_g^{(k)} \bar{Z}_t^{(k),[i]} \end{aligned} \quad (3.15)$$

$\mu_{\hat{m},t-1}^{[i]}$ m ' nci önceki nesnenin ortalaması, $\hat{z}_t^{[i]}$ ise tahmin edilen ölçümdür. CDKF' de ölçüm güncelleme eşitlikleri,

$$\begin{aligned} S_{z,t}^- &= qr \left\{ \left[\begin{array}{c} \sqrt{w_{c_1}^{(k)}} (\bar{Z}_{t,1:N}^{(k),[i]} - \bar{Z}_{t,N+1:2N}^{(k),[i]}) \\ \sqrt{w_{c_2}^{(k)}} (\bar{Z}_{t,1:N}^{(k),[i]} - \bar{Z}_{t,N+1:2N}^{(k),[i]} - 2\bar{Z}_{t,0}^{(k),[i]}) \end{array} \right] \right\} \\ \Sigma_{x_t z_t} &= \sqrt{w_{c_1}^{(k)}} S_{t/t-1}^- [\bar{Z}_{t,1:N}^{(k),[i]} - \bar{Z}_{t,N+1:2N}^{(k),[i]}]^T \end{aligned} \quad (3.16)$$

$$\begin{aligned}
K_t^{[i]} &= (\Sigma_{x_t z_t} / (S_{z,t}^-)^T) / S_{z,t}^- \\
x_t^{[i]} &= x_{t/t-1}^{[i]} + K_t^{[i]}(z_t - \hat{z}_t^{[i]}) \\
U &= K_t^{[i]} S_{z,t}^- \\
S_t^{[i]} &= cholupdate\{S_{z,t}^-, U, -1\}
\end{aligned} \tag{3.17}$$

Burada $S_{z,t}^-$ inovasyon kovaryansını, $K_t^{[i]}$ ise Kalman kazancını gösterir. $x_t^{[i]}$ t anındaki ortalama kestirim ve $S_t^{[i]}$ ise onun kovaryansını göstermektedir. Her bir parçacık için ortalama ve kovaryans Gauss dağılımını üretmektedir ve ortalama,

$$x_t^{[i]} \sim N(x_t^{[i]}, S_t^{[i]}) \tag{3.18}$$

olarak ifade edilir.

3.1.3.2 Nesne/öznitelik durum hesabı

FastSLAM uygulamalarında nesne durum güncellemesi sigma noktalarıyla tanımlanır.

- **Nesne/öznitelik güncelleme aşaması**

Amaçlanan algoritmada sigma noktaları her bir nesne için

$$\begin{aligned}
\chi^{(0),[i],j} &= \mu_{L_t,t-1}^{[i],j} \\
\chi^{(k),[i],j} &= \mu_{L_t,t-1}^{[i],j} + (hP_{L_t,t-1}^{[i],j}), \quad (k=1,\dots,L) \\
\chi^{(k),[i],j} &= \mu_{L_t,t-1}^{[i],j} - (hP_{L_t,t-1}^{[i],j}), \quad (k=L+1,\dots,2L)
\end{aligned} \tag{3.19}$$

Burada $L=2$ dir. $\chi^{(k),[i],j}$ nesnelerin sigma noktalarını göstermektedir ve i ise parçacık indeksini temsil etmektedir. $P_{L_t,t-1}^{[i],j}$ 2×2 lik nesne, kovaryans matrisidir. Nesnelerin ağırlıkları (3.13)' deki gibidir. Her bir nesne için ölçüm güncelleme eşitliği (3.20)'da gösterilmiştir.

$$\begin{aligned}\bar{Z}_t^{(k),[i],j} &= h(\chi^{(k),[i],j}, x_t^{[i],j}), \quad k=0,1,\dots,2L \\ \hat{z}_t^{[i],j} &= \sum_{k=0}^{2L} w_g^{(k)} \bar{Z}_t^{(k),[i],j}\end{aligned}\quad (3.20)$$

$\hat{z}_t^{[i],j}$ 'nci hedefin tahmin edilen ölçümü ve $h(\cdot)$ ise doğrusal olmayan gözlem modelini göstermektedir ve $\bar{Z}_t^{(k),[i],j}$ dönüştürülen sigma noktalarını göstermektedir. (3.20) eşitliğine bağlı olarak Kalman kazancı, çapraz kovaryans ve inovasyon kovaryansı,

$$\begin{aligned}S_{y,t}^{-,j} &= qr \left\{ \begin{bmatrix} \sqrt{w_{c_1}^{(k)}} (\chi_{t,1:N}^{(k),[i],j} - \chi_{t,N+1:2N}^{(k),[i],j})^* \\ \sqrt{w_{c_2}^{(k)}} (\chi_{t,1:N}^{(k),[i],j} + \chi_{t,N+1:2N}^{(k),[i],j} - 2\chi_{t,0}^{(k),[i],j}) \end{bmatrix} \right\} \\ P_{x_t y_t}^j &= \sum_{k=0}^{2L} w_c^{(k)} [\chi^{(k),[i],j} - \mu_{L_t,t-1}^{[i],j}] [\chi^{(k),[i],j} - \mu_{L_t,t-1}^{[i],j}]^T\end{aligned}\quad (3.21)$$

$$K_t^{[i],j} = (P_{x_t y_t}^j / (S_{y,t}^{-,j})^T) / S_{y,t}^{-,j}$$

Eşitliklerinde tanımlanmıştır. j ' nci hedefteki i 'nci parçacığın güncellenmiş nesne ortalaması $\mu_{L_t,t}^{[i],j}$ ve kovaryansı $P_{L_t,t}^{[i],j}$,

$$\begin{aligned}\mu_{L_t,t}^{[i],j} &= \mu_{L_t,t-1}^{[i],j} + K_t^{[i],j} (z_t - \hat{z}_t^{[i],j}) \\ V^j &= K_t^{[i],j} S_{y,t}^{-,j} \\ P_{L_t,t}^{[i],j} &= cholupdate\{S_{y,t}^{-,j}, V^j, -1\}\end{aligned}\quad (3.22)$$

Eşitliklerinden hesap edilir. Burada z_t gerçek ölçümü vermektedir. Her bir nesne Cholesky faktörizasyon yöntemiyle güncellenir. CDKF tabanlı Kalman süzgeçlerinde j ' nci nesne için inovasyon,

$$\tilde{z}_t^{[i],j} = z_t - \hat{z}_t^{[i],j}\quad (3.23)$$

olarak ifade edilmektedir. Her bir i ' nci parçacık için j ' nci nesnenin birleştirilmiş (ağırlıklandırılmış) inovasyon (Shalom 2009)' a göre,

$$\tilde{z}_t^{[i],j} = \sum_{j=1}^M \beta_t^{[i],j} \tilde{z}_t^{[i],j}\quad (3.24)$$

formülünden elde edilir. Burada (3.24), i 'nci parçacıktan gelen j 'nci nesnenin ilişkilendirilmiş olasılığını vermektedir. Literatüre göre $\beta_0^{i,j}$ t anındaki ölçümlerden hiç birisinin hedef nesnelere gelmediğini gösterir.

$$\beta_t^{i,j} = \sum_{\theta \in \theta_{[i],j}} p(\theta | Z_t^{[i]}) \quad (3.25)$$

Burada $Z_t^{[i]}$ t anındaki i 'nci parçacığın toplam algılanan nesne sayısıdır. $\theta_{[i],j}$ ise birleşik ilişkiyel olayları göstermektedir. JPDA tabanlı Kalman yaklaşımlarında i 'nci parçacığın j 'nci nesnesinin durum hesabı ise (3.22)'teki durum yerine (3.26) formülüyle ifade edilir,

$$\mu_{L_t,t}^{[i],j} = \beta_o^{[i],j} \mu_{L_t,t-1}^{[i],j} + \sum_{k=1}^{m_j(t)} \beta_t^{[i],j} \mu_{L_t,t}^{[i],j}(k) \quad (3.26)$$

Kovaryans için durum güncellemesi ise yine (3.61)'teki kovaryans formülü yerine (3.27)'daki gibi ifade edilir,

$$P_{L_t,t}^{[i],j} = \beta_o^{[i],j} P_{L_t,t-1}^{[i],j} + \sum_{k=1}^{m_j(t)} [P_{L_t,t}^{[i],j}(k) + (\mu_{L_t,t}^{[i],j}(k) - \mu_{L_t,t}^{[i],j})(\mu_{L_t,t}^{[i],j}(k) - \mu_{L_t,t}^{[i],j})^T] \beta_t^{[i],j} \quad (3.27)$$

olarak tanımlanmıştır. JPDA tabanlı Kalman süzgeç algoritmasının detayları Shalom (2009)'da anlatılmıştır.

3.1.3.3 Nesne başlangıç işlemleri

FastSLAM CDKF modelinde nesne başlangıç işlemleri j 'nci hedef yani nesne başlangıcında her bir parçacık için

$$\begin{aligned}
\varphi^{(0),[i],j} &= z_t \\
\varphi^{(k),[i],j} &= z_t + (h\sqrt{R_t}), (k = 0, \dots, L) \\
\varphi^{(k),[i],j} &= z_t - (h\sqrt{R_t}), (k = L+1, \dots, 2L)
\end{aligned} \tag{3.28}$$

$$\begin{aligned}
M_t^{(k),[i],j} &= h^{-1}(\varphi^{(k),[i],j}, \hat{\chi}_t^{[i],j}), (k = 0, \dots, 2L) \\
\mu_{L,t}^{[i],j} &= \sum_{k=0}^{2L} w_g^{(k)} M_t^{(k),[i],j} \\
P_{L,t}^{[i],j} &= qr \left\{ \left[\sqrt{w_c^{(k)}} (M_t^{(k),[i],j} - \mu_{L,t}^{[i],j}) \quad \sqrt{R_t} \right] \right\} \\
P_{L,t}^{[i],j} &= cholupdate \left\{ P_{L,t}^{[i],j}, M_t^{(0),[i],j} - \mu_{L,t}^{[i],j}, w_c^{(0)} \right\}
\end{aligned} \tag{3.29}$$

eşitlikleriyle ifade edilir (Ankışhan vd. 2014).

3.1.3.4 Önem ağırlığı hesabı ve tekrar örnekleme

Son gözleme bağlı olarak j ' nci hedefin i ' nci parçacık için önem ağırlık hesabı,

$$W_t^{[i],j} = \left| 2\pi B_t^{[i],j} \right|^{-(1/2)} \exp \left\{ -(1/2) (z_t - \hat{z}_t^{[i],j})^T (B_t^{[i],j})^{-1} (z_t - \hat{z}_t^{[i],j}) \right\} \tag{3.30}$$

formülünden hesaplanır. Burada $B_t^{[i],j} = (P_t^{x,L,[i],j})^T (S_t^{[i],j})^{-1} P_t^{x,L,[i],j} + S_{y,t}^{[i],j}$

' dir. Tekrar örnekleme aşamasında efektif sayıda parçacık için

$$T_{eff} = \frac{1}{\sum_{i=1}^M (\hat{W}^{[i],j})^2} \tag{3.31}$$

M parçacıkların toplam sayısıdır ve $\hat{W}^{[i],j}$ ' nci nesnenin i ' nci parçacık için normalleştirilmiştir.

3.1.4 Q ve R matrislerinin CDKF- FastSLAM için uyarlanması

Kalman süzgeçlerinin iyi bir şekilde çalışması, Q ve R matrislerinin doğru olarak bilinmesine bağlıdır. Öte yandan bu matrisler tam olarak bilinmezler. Yani model ile gerçek arasında her zaman bir farklılık söz konusudur. Yanlış önsel istatistiklerin kullanılması süzgecin zaman içerisinde yanlış kestirimlere yol açmasına böylece süzgeç

iraksamasına yol açabilmektedir. Özellikle doğrusallığın düşük olduğu durumlarda söz konusu durum öne çıkmaktadır. Kalman süzgeçlerinin gelen veriye göre uyum sağlayan bazı versiyonları literatürde mevcuttur (Almagible vd. 2010, Hu vd. 2003, Assa 2013). Bunlardan bir tanesi ise Ankişhan (2014) ve arkadaşlarının çalışmalarında kullanmış oldukları CDKF modelinin uyarlanmasıdır. Almagible (2010) ve arkadaşları tarafından türetilen çalışma, tez çalışması kapsamında örnek alınarak ortaya çıkarılmıştır.

3.1.4.1 Ölçüm gürültüsünün uyarlanması

Ankişhan (2014) ve arkadaşlarının çalışmalarında inovasyon gerçek ölçüm ile tahmin edilen ölçümün farkı olarak gösterilmiştir. Almagible (2010) ve arkadaşlarına göre inovasyon dizisi

$$d_t = z_t - H_t^{[i]} \bar{x}_t^{[i]} \quad (3.32)$$

olarak tanımlanmıştır. Burada $H_t^{[i]}$ i ' nci parçacığın ölçüm matrisi, z_t gerçek ölçüm ve $\bar{x}_t^{[i]}$ ise ortalama durumu göstermektedir. Süzgecin optimal olması durumu inovasyon dizisinin beyaz gürültü dağılımına sahip olmasıyla açıklanmıştır (Almagible vd. 2010).

$$E[d_t d_t^T] = H_t^{[i]} S_{t/t-1}^{[i]} H_t^{[i],T} + R_t \quad (3.33)$$

Tahmin edilen inovasyon kovaryansı, inovasyon vektöründen hesaplanmaktadır. Eğer inovasyon kovaryansı biliniyor ise gözlem hata kovaryansı,

$$\tilde{S}_t^{[i]} = E[d_t d_t^T] = \frac{1}{N} \sum_{j=0}^{N-1} d_{t-j} d_{t-j}^T \quad (3.34)$$

formülünden hesaplanır. Burada $\tilde{S}_t^{[i]}$ inovasyon serisine bağlı olarak hesaplanan kovaryans matrisini verir (Almagible vd. 2010). t anında N boyutlu kaydırmalı bir pencerenin ortalaması sonucu $\tilde{S}_t^{[i]}$ hesaplanmıştır. Gözlem hata kovaryansı,

$$\widehat{R}_t = \widetilde{S}_t^{[i]} - H_t^{[i]} S_{t-1}^{[i]} H_t^{[i],T} \quad (3.35)$$

olarak tanımlanır.

3.1.4.2 Süreç gürültüsünün uyarlanması

Tahmin edilen inovasyon kovaryansı inovasyon gözlemlerinden hesaplanan kovaryansla aynı olmalıdır (Almagible vd. 2010). Bu durum (3.36) eşitliğini geçerli kılar.

$$\frac{1}{N} \sum_{j=0}^{N-1} d_{t-j} d_{t-j}^T = H_t^{[i]} \widetilde{S}_{t-1}^{[i]} H_t^{[i],T} + R_t \quad (3.36)$$

Eğer süzgeç çalışmasında (3.36) eşitliği sağlanamazsa R_t yada $S_{t-1}^{[i]}$ yanlış hesaplanır. $\widetilde{S}_{t-1}^{[i]}$ tahmin edilen gözlem gürültü kovaryansından tanımlanabilir. Eğer tahmin edilen süreç gürültü ve gözlem gürültü kovaryansları biliniyor ise Q_t ölçekleme parametresi yardımıyla hesap edilir.

$$\begin{aligned} \eta &= \frac{\text{trace}\{\widetilde{S}_t^{[i]} - R_t\}}{\text{trace}\{H_t^{[i]} S_{t-1}^{[i]} H_t^{[i],T}\}} \\ &= \frac{\text{trace}\{H_t^{[i]} (F_t^{[i]} \widetilde{S}_{t-1}^{[i]} F_t^{[i],T} + \hat{Q}_{t-1}) H_t^{[i],T}\}}{\text{trace}\{H_t^{[i]} (F_t^{[i]} \widetilde{S}_{t-1}^{[i]} F_t^{[i],T} + Q_{t-1}) H_t^{[i],T}\}} \end{aligned} \quad (3.37)$$

Parametre yardımıyla Q_t ölçeklenebilir,

$$\hat{Q}_t = Q_{t-1} \sqrt{\eta} \quad (3.38)$$

Ölçekleme parametresi birden büyük ya da küçük olabilmektedir (Almagible vd. 2010).

4. BULGULAR

Tez çalışmasının bu bölümünde bahsedilen SLAM problemlerinin çözümleri için önerilen metotların sonuçları mevcut metotlarla kıyaslanmıştır. Bulgular dört ana başlık altında gerçekleştirilmiştir; gözlem gürültüsü minimizasyonu, veri ilişkilendirme yöntemleriyle uyarlanan SLAM yaklaşımı, optimizasyon yöntemlerine dayalı iyileştirilmiş FastSLAM yaklaşımları ve Uyarlı FastSLAM yaklaşımıdır.

4.1 Gözlem gürültüsü minimizasyonu

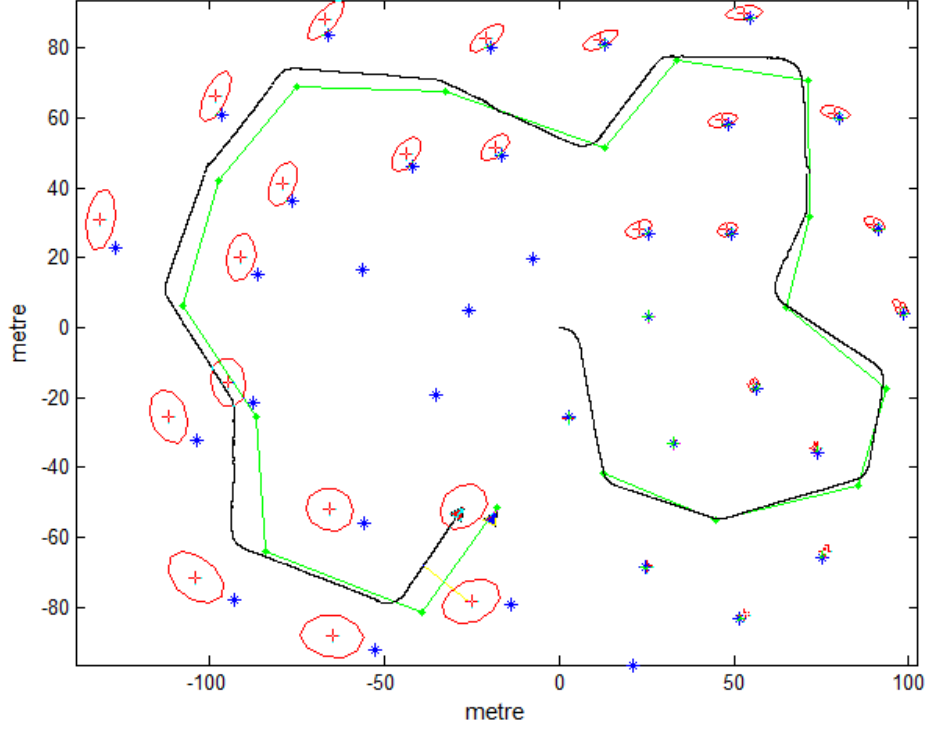
Gözlem gürültüsü azaltımı, diğer bir adıyla minimizasyonu, SLAM uygulamalarının ana problemleri arasında yer alır. Daha önceki çalışmalarda SLAM çıktılarının kestirim doğruluğunu artırmak için farklı istatistiksel yöntemler önerilmiştir. Bu bölümde daha önceki kullanılan yöntemlerden farklı olarak üç yeni metodun sonuçları gösterilmiş ve yorumlanmıştır.

4.1.1 Karakök kokusuz ve karakök merkez fark Kalman süzgeçleri

Bu çalışmada önerilen süzgeçlerin performanslarını hesaplayabilmek için Bailey (2003)' teki senaryolar tercih edilmiştir. Bailey (2003)' ün tasarlamış olduğu senaryolar, var olan süzgeçlerle kıyaslama imkânı sunduğu için SLAM uygulamalarında tercih edilen bir araç (*Tool*) olarak bilinir. Tasarlanan ortamda, robot/özerk araç belirli ara noktaları takip eder ve sınır taşlarının konfigürasyonu ile birlikte çevrenin haritasını oluşturur.

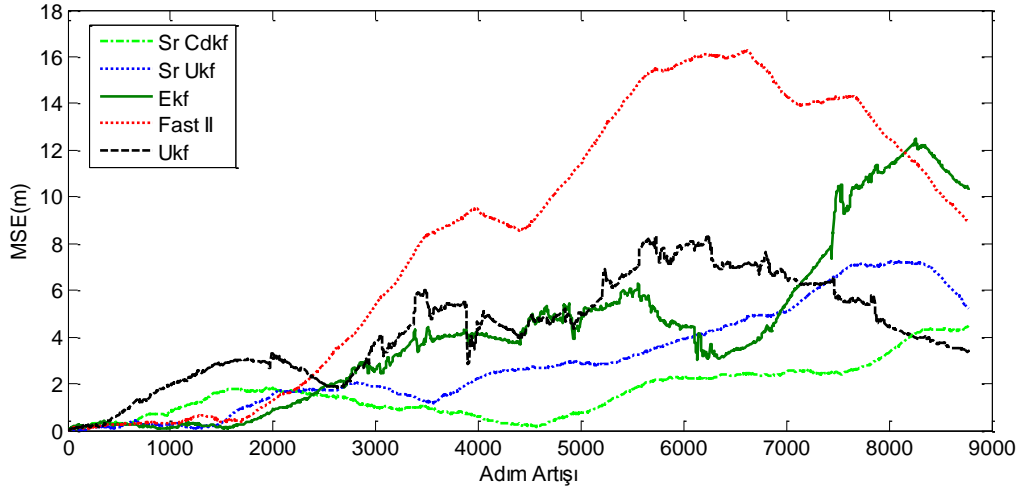
Benzetim çalışmanın bu bölümünde kontrol girişlerindeki belirsizlik $\sigma_w = 0.1$ m/s ve $\sigma_s = 1^\circ$ olarak alınmıştır. Gözlem adımı ve ilişkilendirilmiş güncelleme adımı her bir sekiz adımdan sonra kestirici aracılığıyla gerçekleştirilmektedir. Çalışmada ilgili harita üzerinden farklı belirsizlik değerleri için de karşılaştırma yapılmıştır.

Önerilen süzgeç modelleri Bailey (2003)'teki harita ve sınır taşlarına bağlı kalınarak konum ve başlık ortalama kare hataları geleneksel EKF, UKF ve parçacık süzgeci tabanlı FASTSLAM II ile kıyaslanmıştır. Şekil 4.1'de SR-CDKF' nin, şekil 4.2'de ise bütün süzgeçlerin ortam üzerindeki kestirim hataları görülmektedir.



Şekil 4.1 SR-CDKF'nin WebMap (Bailey 2003) haritası için kestirim sonuçları ($\sigma_w = 0,1$ m/s ve $\sigma_s = 1^\circ$)

Burada kırmızı elipsler nesne kestirim belirsizliklerini, siyah yol tahmin edilen robot/özerk araç rotasını ve yeşil yol ise gerçek robot/özerk araç rotasını göstermektedir.



Şekil 4.2 Süzgeçlerin ortalama kare hatası sonuçları

Ukf: Kokusuz Kalman süzgeci, Fast II: FastSLAM II, Ekf: Genişletilmiş Kalman süzgeci, SrCdkf: Karakök merkez fark Kalman süzgeci ve SrUkf: karakök kokusuz Kalman süzgeci' ni temsil etmektedir.

Şekil 4.2' den görülmektedir ki; süzgeçler kestirim esnasında özellikle 5000' inci adım artışının olduğu bölgeye kadar FastSLAM II hariç iyi tahmin sonuçları elde etmişlerdir. Şekil 4.2'den en uygun sonuç veren süzgecin SR-CDKF olduğu görülmektedir. Dönüş esnasında konum ve açısız başlık hatası en düşük hata karesi sonuçlarını sağlamıştır. Özellikle süzgeç, araç rotası tahmini olarak gerçek değere en yakın kestirimi vermiştir. Bununla birlikte SR-UKF süzgecinin, geleneksel UKF' den konum hatası bakımından daha uygun sonuçlar verdiği yine şekil 4.2'den gözlenmektedir. Eklenecek olursa, süzgecin çalışma açısından daha kısa işlem süresine sahip olduğu ve durum tahminine bağlı olarak zaman güncellemesini UKF' den daha iyi yapabildiği yine Ankışhan ve Efe (2010)'nin çalışmalarında gözlenmiştir. Senaryo için diğer kullanılan bir yöntem ise FastSLAM II'dir. FASTSLAM II, bu ortam koşulları için Kalman tabanlı kestiriciler kadar iyi sonuçlar sağlayamamıştır. Çünkü bu senaryoda kullanılan parçacık sayısı yeterli değildir. FastSLAM II için dezavantajlardan birisi parçacık sayısı seçimi problemi, diğer adıyla işlem yükü problemidir. Eğer ki parçacık sayısı artırılırsa süzgeç daha uygun sonuçlar verebilmektedir. Fakat parçacık sayısı arttıkça işlem yükü artmaktadır. Bu ise model için istenilen bir durum değildir. Yöntemlerin bu ortam için ortalama konum, başlık hataları ile işlem sürelerini gösteren sonuçları Çizelge 4.1' de verilmiştir.

Çizelge 4.1 Süzgeçlerin Ortalama Kare hataları ve işlem süreleri

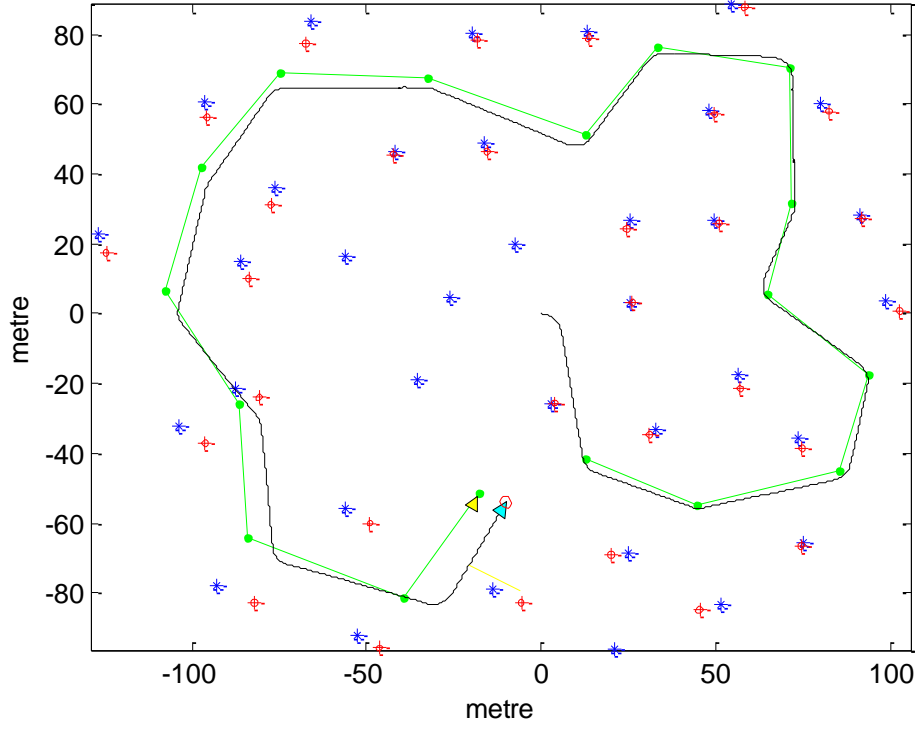
	Pozisyon hatası(m)	Başlık hatası (Radyan)	İşlem süresi (Sn.)
EKF	0.7240	0.0143	65.40
UKF	0.6823	0.0122	68.76
FASTSLAM II (50P.)	0.8756	0.0143	237.03
SR-UKF	0.6597	0.0117	61.35
SR-CDKF	0.6376	0.0103	59.08

Çizelge 4.1’de sonuçlar incelendiğinde karekök tabanlı süzgeçlerin sonuçlarında bir iyileştirme olduğu görülmektedir. Bu çalışmada FastSLAM II için elde edilen sonuçlar Matlab programı ile elde edildiğinden, parçacıklar arasında bir paralelleştirme yapılmamıştır. Süzgeç her bir parçacık için çalıştırılmış, yani 50 kez işleme tabi tutulmuştur. Dolayısı ile elde edilen FastSLAM II işlem süresi olması gerekenden daha uzun çıkmıştır. Sonuçların objektif değerlendirilmesi açısından kıyaslama esnasında FastSLAM II’nin işlem süresi göz önünde bulundurulmamıştır. Çizelge 4.1, en uygun kestirim doğruluğuna sahip yöntem olarak SR-CDKF’ nin olduğunu göstermektedir.

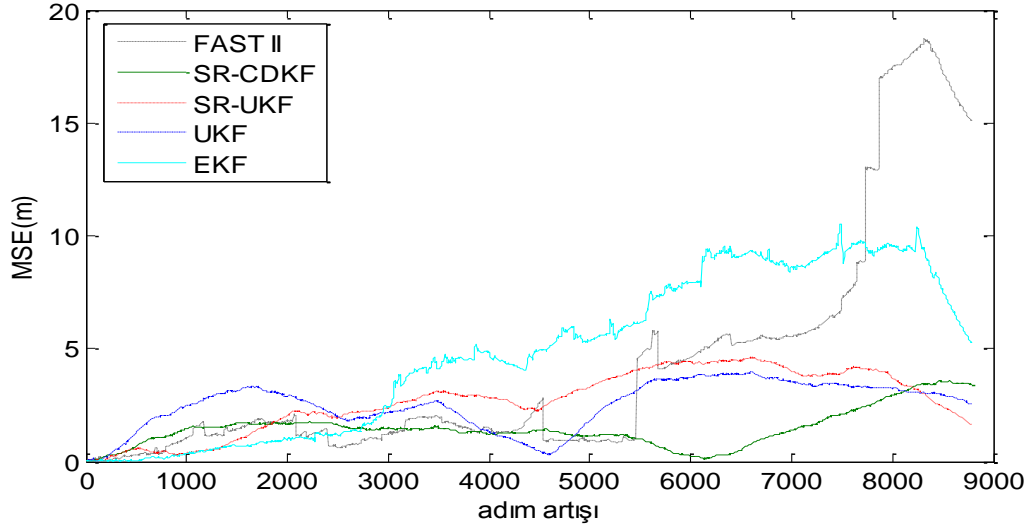
4.1.2 SLAM’ de uyarlı bulanık mantık çıkarımı destekli Kalman süzgeci

SLAM probleminde temel sorunlardan bir tanesi ölçüm gürültüsünün istatistikî olarak birbirine bağlı olmasıdır. Ölçüm gürültüsü harita boyunca her adımda büyüyerek haritanın kaymasına, kullanılan metodun hatalı kestirim yapmasına sebep olmaktadır. Tez çalışmasının bu kısmında, ölçüm gürültüsünün zamanla büyümesinden kaynaklanan bu hatalı sonuçların iyileştirilmesi maksadıyla, belirsizlik matrislerinin modele uyarlanarak hatayı minimize etmesi amaçlanmıştır. Uyarlı bulanık mantık çıkarımı sistemi (ANFIS), SLAM algoritmasında çalışırken Q ve R matrislerini hata durumuna göre sisteme adapte etmeyi ve hatayı bu şekilde minimize etmeyi amaçlamıştır. ANFIS algoritması ve detayları EK 4’te verilmiştir.

Bu çalışmada SLAM için önerilen süzgeç modeli harita ve nesnelere bağlı olarak konum ve başlık ortalama kare hataları geleneksel EKF, UKF, SR-CDKF ve FASTSLAM II ile kıyaslanmıştır. Şekil 4.3'te önerilen algoritmanın kestirim sonucu, şekil 4.4'te ise bütün kestiricilerin ortam üzerindeki $\sigma_w = 0.3$ m/s ve $\sigma_s = 3^\circ$ değerleri için kestirim hataları görülmektedir.



Şekil 4.3 ANFIS destekli SR-CDKF kestirim sonucu: gerçek rota (yeşil) ve kestirilen rota (siyah)



Şekil 4.4 Önerilen senaryo için Metotların kestirim hata sonuçları

Q ve R matrislerinin ANFIS ve Rauch-Tung-Striebel (RTS) destekli uyarlanabilir modeli süzgeç kestirim doğruluğunu artırmıştır. Şekil 4.3-4 süzgeçleri kestirim performansları hakkında bilgiler vermektedir. Süzgeçlerin senaryo için ortalama konum, başlık hatalarını ve işlem sürelerini gösteren sonuçları çizelge 4.2’ de verilmiştir.

Çizelge 4.2 Süzgeçlerin ortalama kare hataları ve işlem süreleri

Metotlar	Pozisyon hatası (m)	Başlık hatası (Radyan)	İşlem süresi (Sn)
EKF	4.8336	0.4343	68.40
UKF	2.6909	0.0410	88.83
FASTSLAM II (100P.)	4.1633	0.3343	359.89
SR-UKF	2.5204	0.0369	63.51
SR-CDKF	1.4565	0.0269	76.88

4.1.3 KaraKök kokusuz Kalman süzgeci tabanlı FastSLAM

Benzetim çalışmalarının bu kısmında FastSLAM II modelinin içerisinde geleneksel yaklaşımlardan farklı olarak SR-UKF tabanlı süzgeçler kullanılmıştır. Sonuçlar, SRU-FastSLAM, SRU-PF-FastSLAM, kokusuz (U)-FastSLAM, kokusuz süzgeç yardımcı (U-Aid) FastSLAM, kokusuz parçacık süzgeç tabanlı (UPF)-FastSLAM ve FastSLAM II süzgeçleriyle kıyaslanmıştır. Bu süzgeçlerin çalışma prensipleri hakkında kısa bilgiler verilirse; SRU-FastSLAM, hem robot/özerk araç pozisyonunun parçacık süzgeciyle tahmini esnasında her bir parçacık için, hem de nesne işaretlerinin konum tahmini için SR-UKF' yi kullanmıştır. SRU-PF-FastSLAM süzgeci, robot/özerk araç pozisyonunun tahmini için parçacık süzgecinin örnekleme aşamasında her bir parçacık için SR-UKF' yi kullanmış, nesne işaretleri konum tahmini için ise FastSLAM II' de olduğu gibi EKF kullanmıştır. U-FastSLAM süzgeci hem robot/özerk araç hem de nesne işaretleri için UKF' yi kullanmıştır. U-Aid-FastSLAM süzgeci ise nesne işaretlerinin durum tahmin ve güncellemesi için, UPF-FastSLAM süzgeci ise robot/özerk araç pozisyon tahmini için kullanılan parçacık süzgecinin örnekleme aşamasında her bir örneklem için UKF' yi kullanmıştır. U-FastSLAM, U-Aid-FastSLAM ve SRU-PF-FastSLAM süzgeçleri hakkında detaylı bilgi için (Kim vd. 2009) bakılabilir.

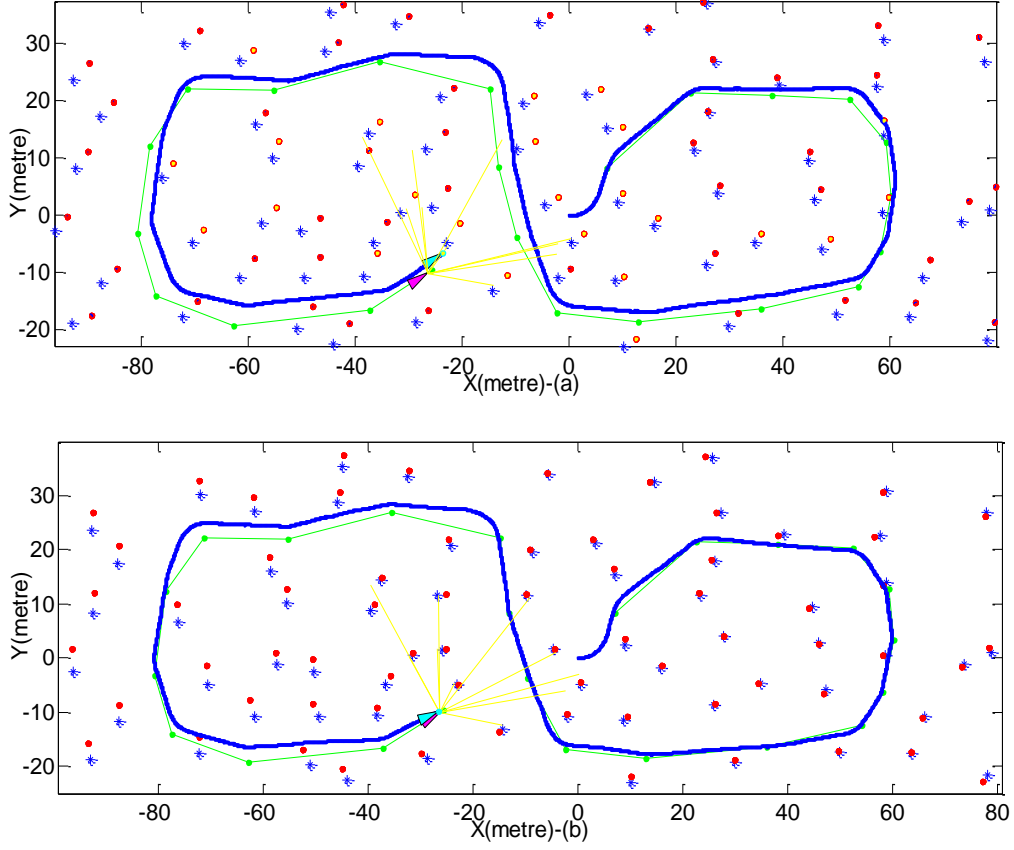
Bu çalışmada (Bailey 2003)' ün üretmiş olduğu senaryodan ilham alınarak hazırlanan robot/özerk araç pozisyonu ve nesne işaretleri kullanılmıştır. Benzetim ortamında sistem kontrol parametreleri; araç hızı 3m/sn, maksimum başlık açısı $30\pi/180$ radyan, araç dingil mesafesi 4m, kontrol frekansı 5 Hz olarak alınmıştır. Bunların yanı sıra ölçüm ve kontrol gürültü matrisleri sırasıyla,

$$Q = \begin{bmatrix} 0.5^2 & 0 \\ 0 & [5 * (\pi / 180)]^2 \end{bmatrix} \quad (4.1)$$

$$R = \begin{bmatrix} 0.8^2 & 0 \\ 0 & [8 * (\pi / 180)]^2 \end{bmatrix} \quad (4.2)$$

olarak alınmıştır. Çalışmada robotun/özerk aracın laser maksimum algılama mesafesi 30m, gözlemler arası tarama frekansı 40 Hz olarak belirlenmiştir. Bunların yanı sıra veri ilişkilendirme için kapı elips genişliği maksimum mesafesi 5m, yeni nesne işareti algılama minimum mesafesi ise 25m olarak alınmıştır. Alınan sensör bilgilerine $\sigma^2 = 1$ varyansına sahip beyaz gürültü eklenerek senaryo oluşturulmuştur.

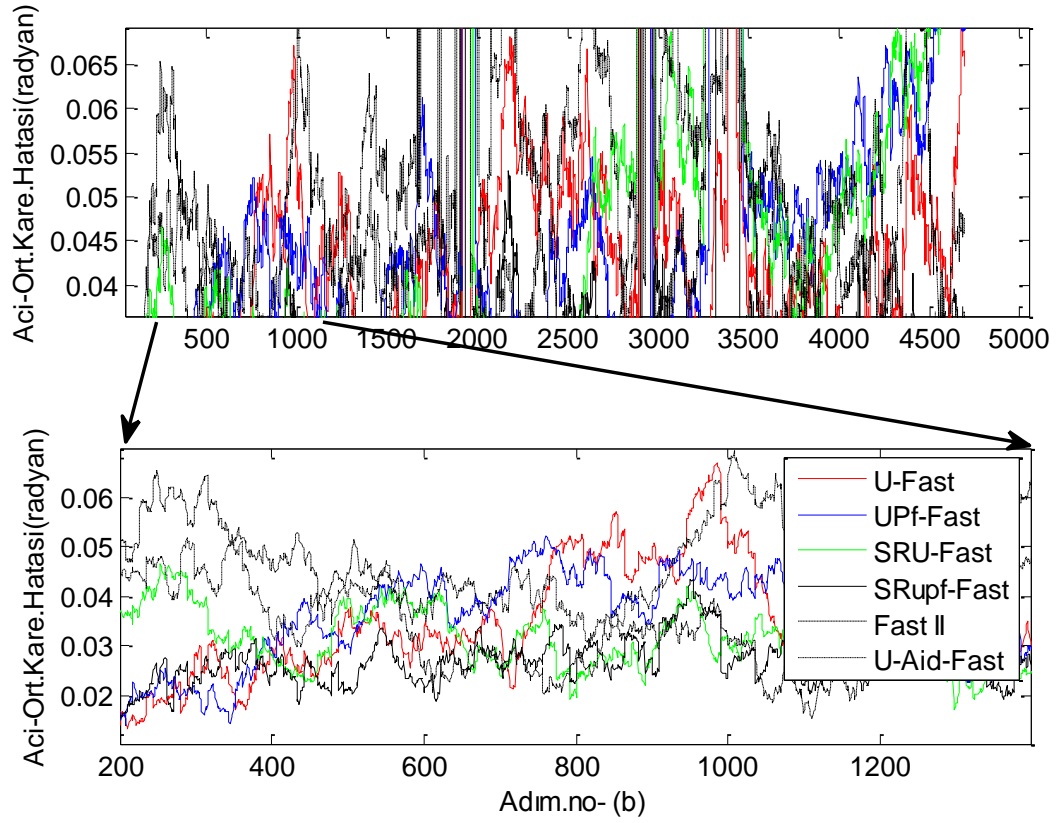
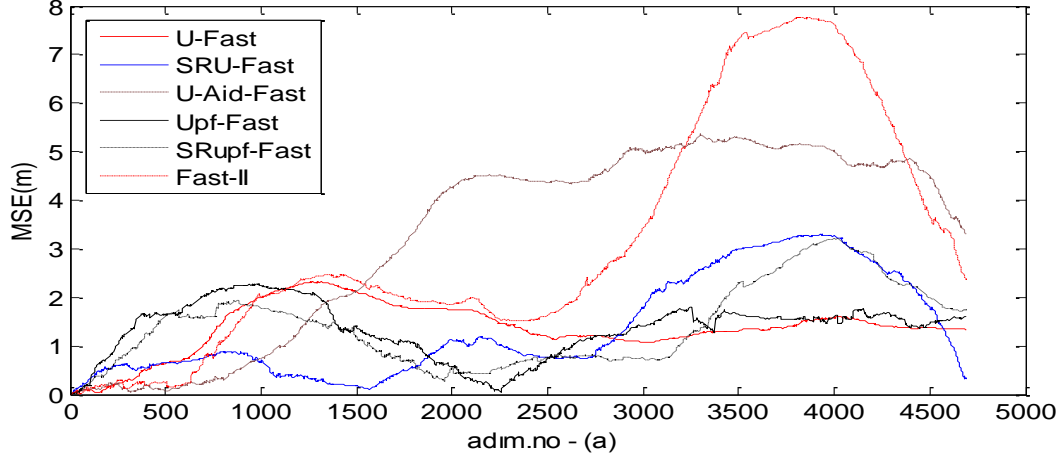
Şekil 4.5.a,b ilgili senaryo için U-FastSLAM ve SRU-FastSLAM kestirim sonuçlarını göstermektedir.



Şekil 4.5.a. U-FastSLAM, b. SRU-FastSLAM kestirim sonuçları

Çalışma için üretilen senaryoda 69 nesne işareti ve 23 yol noktası kullanılmıştır. Yeşil yol gerçek rotayı, mavi ise tahmin edilen rotayı göstermektedir. Mavi nesne işaretleri gerçek pozisyonu, kırmızılar ise tahmin edilen nesne işaretlerini göstermektedir.

Çalışmanın ortalama kare konum hataları ve robotun/özerk aracın başlık açısı hatası elde edilen sonuçları şekil 4.6' da verilmiştir.



Şekil 4.6 Süzgeçlerin ilgili senaryo için kestirim ortalama kare hatası sonuçları:
a. pozisyon hatası, b. başlık açısı hatası

Şekil 4.6 süzgeçlerin ilgili senaryo için pozisyon ve ortalama kare hatalarını göstermektedir. Şekil 4.6.b’ den süzgeçlerin SRU-PF-FastSLAM hariç genel olarak bütün benzetim uygulaması esnasında aynı davranışı sergilediği gözlenmektedir; 500-1000 ve 2500-3600 adımları arasında hataları artmakta, 1200-1500 ve 3600-5000 arasında ise azalmaktadır. Şekil 4.6-a’ dan hataları en iyi minimize edebilen süzgecin SRU-FastSLAM olduğu görülmektedir. Şekil 4.6-b incelendiğinde bütün yöntemler için robotun/özerk aracın dönüş yaptığı pozisyonlarda başlık açılarının gürültüden anlık olarak başlık pozisyonunu kaçırdıkları ve buna bağlı olarak istenmeyen oranda büyüdükleri gözlenmektedir. Süzgeçlerin ortalama kare hataları ve varyansları çizelge 4.3’ te verilmiştir. Çizelge 4.3’ te gürültüden kaynaklı yüksek çıkan (1500-2200, 2900-3100 ve 3300-3500 arası) bazı değerler varyans hesaplamasına katılmamıştır. Çizelge 4.3 sonuçları için benzetim her bir süzgeç için 10 kez çalıştırılmış, ortalaması alınmıştır.

Çizelge 4.3 Süzgeçlerin konum ve başlık açısı ortalama kare hataları ve varyans değerleri

Yöntem	Konum (x,y) $\sqrt{x^2 + y^2}$	Konum- Varyans	Başlık (°)	Başlık- Varyans (°)
FastSLAM II	2.6135	1.8690	0.2258	0.7289
U-Aid-FastSLAM	2.4974	1.1487	0.2397	0.4746
UPF-FastSLAM	2.3149	0.9353	0.2122	0.3925
U-FastSLAM	2.0438	1.2252	0.1753	0.6019
SRU-FastSLAM	1.4096	0.5314	0.1401	0.3002
SRU-PF-FastSLAM	1.8224	0.6937	0.1857	0.4122

(x,y) koordinatına bağlı olarak hazırlanan senaryoda süzgeçler farklı performanslar sergilemişlerdir. FastSLAM II süzgeci robot/özerk araç pozisyon hatası ve konum varyans değerleri en yüksek sonucu vermiştir. Çünkü süzgeç kullandığı EKF ile doğrusallaştırma esnasında gerçek değerden uzaklaşmıştır. Bununla birlikte bu süzgecin, başlık hatasının U-Aid-FastSLAM’den daha uygun çıktığı gözlenmiştir.

Çizelge 4.3' te göze çarpan en önemli durum kullanılan SR-UKF' nin FastSLAM II süzgecinde gerçekleştirmiş olduğu iyileştirmedir. Sonuçlar UT dönüşümlü U-FastSLAM yöntemine nazaran daha iyi çıkmıştır. SR-UKF gerek tahmin esnasında gerekse ölçüm güncellemesi esnasında QR çarpanlarına ayırma yönteminden yararlanmaktadır. Ayrıca hata kovaryanslarını hesaplamakta ve ardından bu kovaryansları Cholesky güncelleme yöntemiyle güncellemektedir. Bu işlem algoritmanın tahmin ve güncellenme aşamasında gerçekleşmiştir ve UKF, EKF süzgeçlerinden SLAM modeli için daha başarılı sonuçlar sağlamıştır. FastSLAM II modelinde kullanılan parçacık süzgecinin örnekleme aşaması SR-UKF tahmin kabiliyetiyle yapıldığından SRU-PF-FastSLAM süzgecinin de sonuçları UT tabanlı FastSLAM ve FastSLAM II modellerinden daha uygun sonuçlar vermiştir.

4.2 SLAM problemi için veri ilişkilendirme

Bu bölümde, yine daha önceki deneylerde olduğu gibi Tim Bailey (2003)' nin senaryosundan yararlanılarak; iyileştirilmiş süzgeç yardımıyla yapılan benzetim çalışması ve diğer süzgeç modelleriyle yapılan karşılaştırmalar gösterilmiştir. Bu çalışmada iki senaryo üzerinden uygulama gerçekleştirilmektedir; ilk uygulama SLAM' in gürültülü ortamda nesne işaretleriyle birlikte gerçekleştirilen iç mekân uygulaması olarak bilinmekte, ikinci uygulama ise yine aynı ortamda gerçekleştirilmiştir. Fakat ortamda rasgele durağan olarak dağıtılmış parazit yankılar mevcuttur. Her bir senaryo için işlem aşaması ise iki adımda gerçekleştirilmiştir. İlk adım algılama; sinyal gürültü oranına (Signal to Noise Ratio-SNR) bağlı olarak, algılama olasılığı (PD), ve yanlış alarmların olasılığı (PF)'dir. Eğer SNR düşük olursa PF tarafından PD' nin daha fazla etkilendiği görülmüştür (Wong vd. 2010). İkinci adım ise haritalama ve veri ilişkilendirme algoritmaları üzerinde durmaktadır.

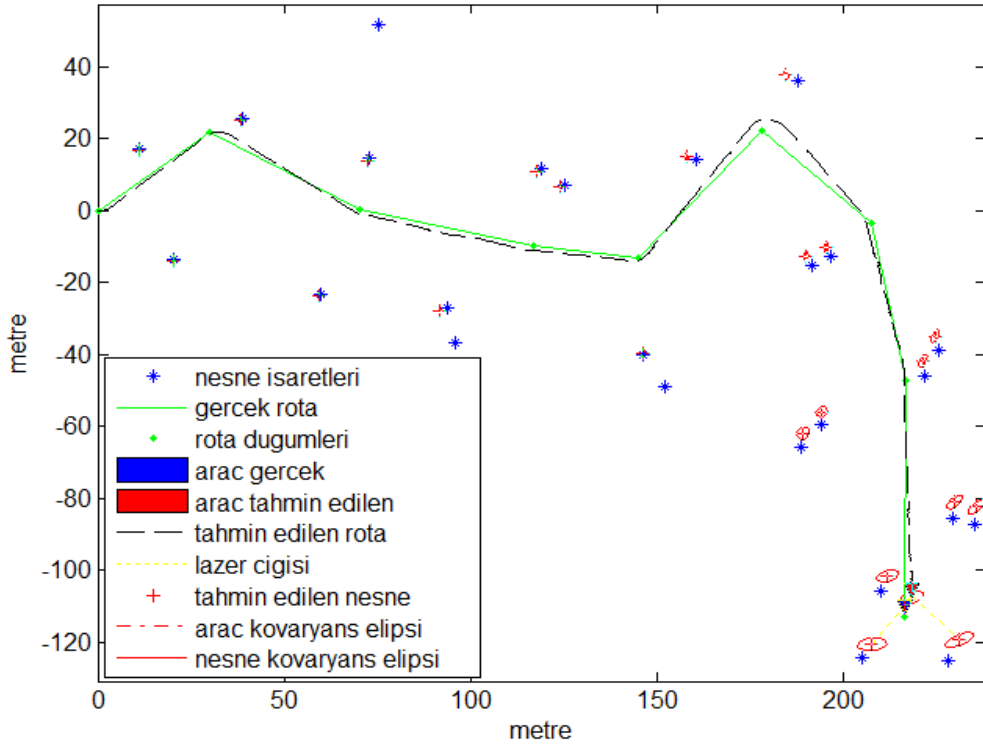
Kullanılan benzetim için ilgili sistem kontrol parametreleri; araç hızı 3m/sn, maksimum başlık açısı $30 * \pi / 180$ radyan, araç dingil mesafesi 4m, kontrol sinyalleri arası 40Hz olarak alınmıştır. Bunun yanı sıra kontrol ve ölçüm gürültüleri,

$$Q = \begin{bmatrix} 0.5^2 & 0 \\ 0 & [5 * (\pi / 180)]^2 \end{bmatrix} \quad (4.3)$$

$$R = \begin{bmatrix} 0.5^2 & 0 \\ 0 & [5 * (\pi / 180)]^2 \end{bmatrix} \quad (4.4)$$

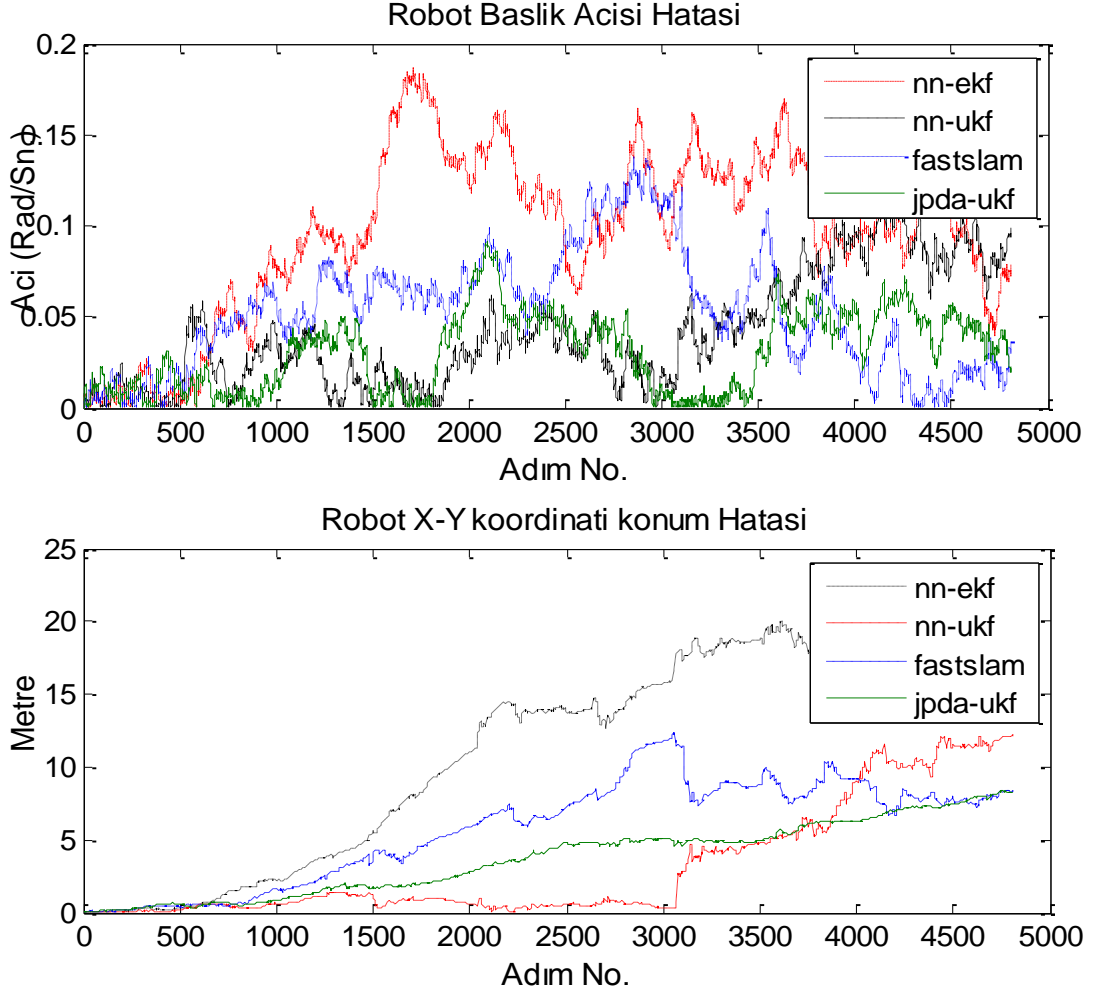
olarak alınmıştır. Gözlem parametreleri olarak laser maksimum algılama mesafesi 30m, araç gözlemler arası tarama zamanı ise ortalama $8 * 0.025$ saniye olarak belirlenmiştir. İlişkilendirme için kapı elips genişliği maksimum mesafe 5 metre, yeni gözlemlenen nesne işareti için minimum mesafe ise 25 metre olarak kabul edilmiştir. Çalışma esnasında alınan sensör bilgilerine $\sigma^2 = 1$ olarak beyaz gürültü eklenmiştir.

Senaryo I. İlk senaryo ortamda parazit yankıların olmadığını kabul etmektedir. Kabul edilen kontrol ve ölçüm gürültüleri eşitlik (4.3) ve (4.4)' te verilmiştir. İlgili senaryo (0m,0m) noktasında bilinmeyen bir ortam için başlamakta (220m,-110m) noktasında bitmektedir. JPDA-UKF süzgecinin birbirine çok yakın nesne işaretli ortamda vermiş olduğu kestirim sonuçları şekil 4.7' de verilmiştir.



Şekil 4.7 JPDA -UKF algoritmasının ilgili senaryo için kestirim sonuçları

Senaryo I için aracın pozisyon ve başlık hatası *NN-EKF*, *NN-UKF*, *FastSLAM II* ve *JPDA-UKF* yöntemleriyle kıyaslandığında şekil 4.8-9 sonuçları elde edilmiştir. Çıktılar 30 Monte Carlo Run (MCR) sonucunda elde edilmiştir.

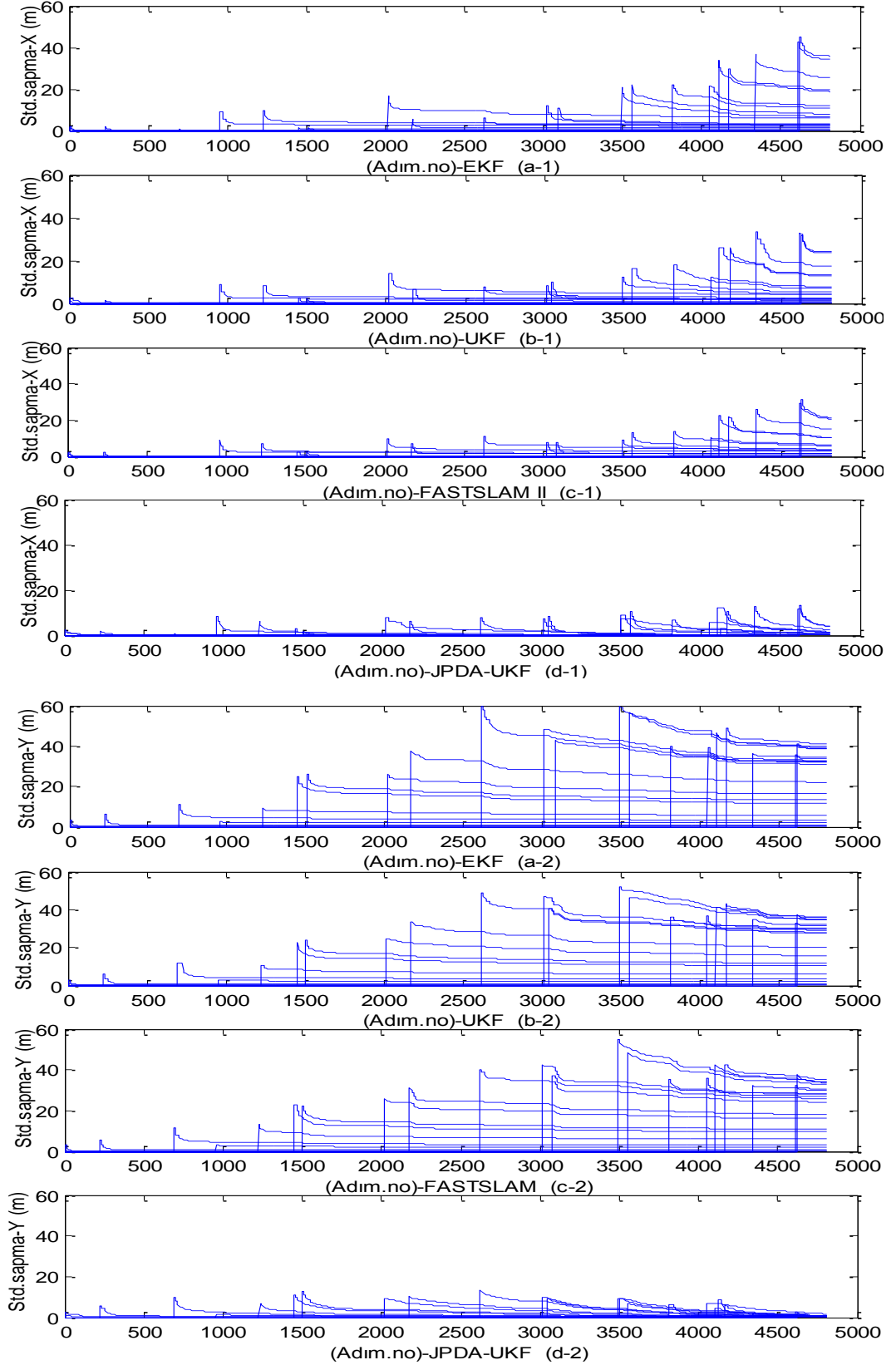


Şekil 4.8 Süzgeçlerin robot pozisyonu ve başlık açısı ortalama kare hata kestirim sonuçları

Şekil 4.8, dört süzgecin yoğun gürültülü parazit yankının olmadığı ortamda ortalama kare hataları hakkında bilgiler sunmuştur. Süzgeçler genel olarak 1500'ncü adıma kadar doğrusal ortalama kare hatası artışına sahipken, bu adımdan itibaren ortam şekil 4.7 ve 4.8' den de görüleceği üzere, nesne işaretlerinin sayısının da paralel artışı ve belirsizliklerin artması sebebiyle hata kareleri farklı oranda ivme artışına girmişlerdir. Çünkü süzgeçler bu adımdan itibaren rotadaki (başlık açısı ve pozisyon değişiminin

büyükülüyle alakalı olarak) dönmeler ve yeni kestirilen nesne işaretlerinin geçiş matrisinde yaratılan büyüme etkisiyle, gürültünün istatistiksel olarak zaman içerisinde büyümesi sonucu hata ortalamasının büyümesine yol açmıştır.

Şekil 4.9, haritadaki bütün nesne işaretlerinin ilk kestiriminden itibaren harita boyunca güncellenmesi ve belirsizlik alanları hakkındaki bilgileri içermektedir (teorik olarak nesne işaretlerinin varyanslarının ilk kestirildiği andan itibaren değerlerinin sıfıra doğru yaklaşması gereklidir). Görüldüğü üzere süzgeçlerin X-Y koordinatındaki standart sapma değerleri, süzgeçlerin kestirim doğruluğu ile paralel sonuçlar vermiştir. Teorik olarak haritalama esnasında nesne işaretlerinin standart sapma değerleri süzgeçlerin kestirimi ve doğruluğuyla uygun aralıklarda daralmalıdır. Şekil 4.9' dan görüldüğü üzere en uygun minimizasyonu (monoton olarak standart sapma daralma olayı) JPDA-UKF süzgecinin sağladığı görülmüştür. Süzgeç zaman adımı ilerledikçe özellikle nesne işaretlerinin birbirine yakın olduğu durumlarda ilişkilendirmeyi diğer süzgeçlere göre daha başarılı yapabilmiştir, böylece kestirim doğruluğu olumlu yönde etkilenmiştir.



Şekil 4.9 Bütün algılanan nesne işaretleri için nesne işaretleri konum kestiriminin bütün süzgeçler için standart sapma değerleri:
a.X-Koordinatı, b.Y-Koordinatı

Gözlendiği gibi nesne işaretleri pozisyonunda , şekil 4.9.b Y koordinatındaki standart sapma değerleri, şekil 4.9.a X koordinatındaki standart sapma değerlerini vermektedir, belirsizlik monoton olarak azalmaktadır.

Şekil 4.9, X-Y koordinatında hesaplanan her bir nesne işaretini ve akabinde yapılan korelasyonlu iyileştirme sonucunu vermektedir. Görüldüğü gibi süzgeçlerden en uygun harita güncelleme işlemi JPDA-UKF tarafından gerçekleştirilmiştir. Çünkü güncelleme nesne işaretlerinin de standart sapma değerlerinin sifıra yaklaşmasıyla sağlanmıştır. Nesne işaretlerini ilgili adım aralığı içerisinde minimize edebilen ve harita güncellemesini en uygun aralıklarda yapabilen JPDA-UKF süzgeci sonuçları şekil 4.8 ve 4.9' da görülmektedir. Diğer süzgeçlerin hata kareleri ortalamaları ve başlık açısı hata ortalamaları çizelge 4.4' te verilmiştir.

Açısal başlık hataları için yine konum hatası ve varyans minimizasyonu ile paralel olarak en uygun sonucu JPDA-UKF süzgecinde görmekteyiz.

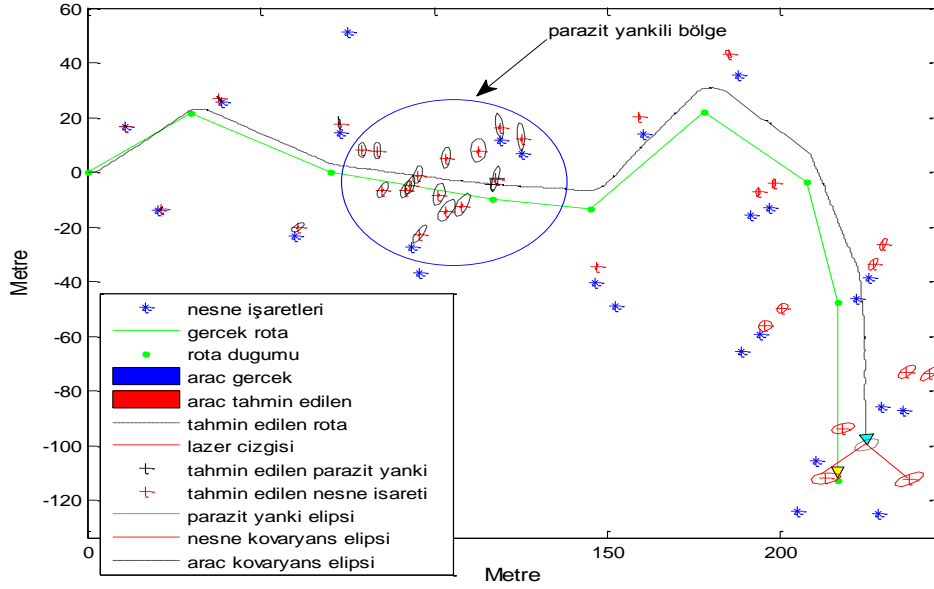
Çizelge 4.4 Robot başlık açısı ve konum ortalama kare hataları

Süzgeç	Açısal başlık hatası	Konum hatası
NN-EKF	0.3101	3.3582
NN-UKF	0.2026	1.9287
FastSLAM II	0.2300	2.3915
JPDA-UKF	0.1747	1.7903

Senaryo II. Benzetim çalışmasının bu kısmında haritanın 4. düğümünde 30x30 m²' lik bir alanda statik parazit yankı üretilmiştir. Burada parazit yankının SLAM için kısa bir tanımı yapılırsa; Çevre koşulları algılayıcı sisteminden alınan ölçümlerle birlikte devreye girerek süzgeç hesaplamalarını olumsuz yönde etkiler. Genel olarak, ortamın doğal gürültüsü olarak bilinen veya kasti/ kasıtlı olmayan elektronik enterferanslar parazit yankı olarak tanımlanır. Çalışmanın bu bölümünde parazit yankı önceliği (gelme olasılığı) 0.2, yoğunluğu $CD=1/(30 \times 30)$ olarak alınmıştır. Kabul edilen kontrol ve ölçüm gürültüleri yine senaryo I' de olduğu gibi eşitlik (4.3) ve (4.4)' deki değerlerdir.

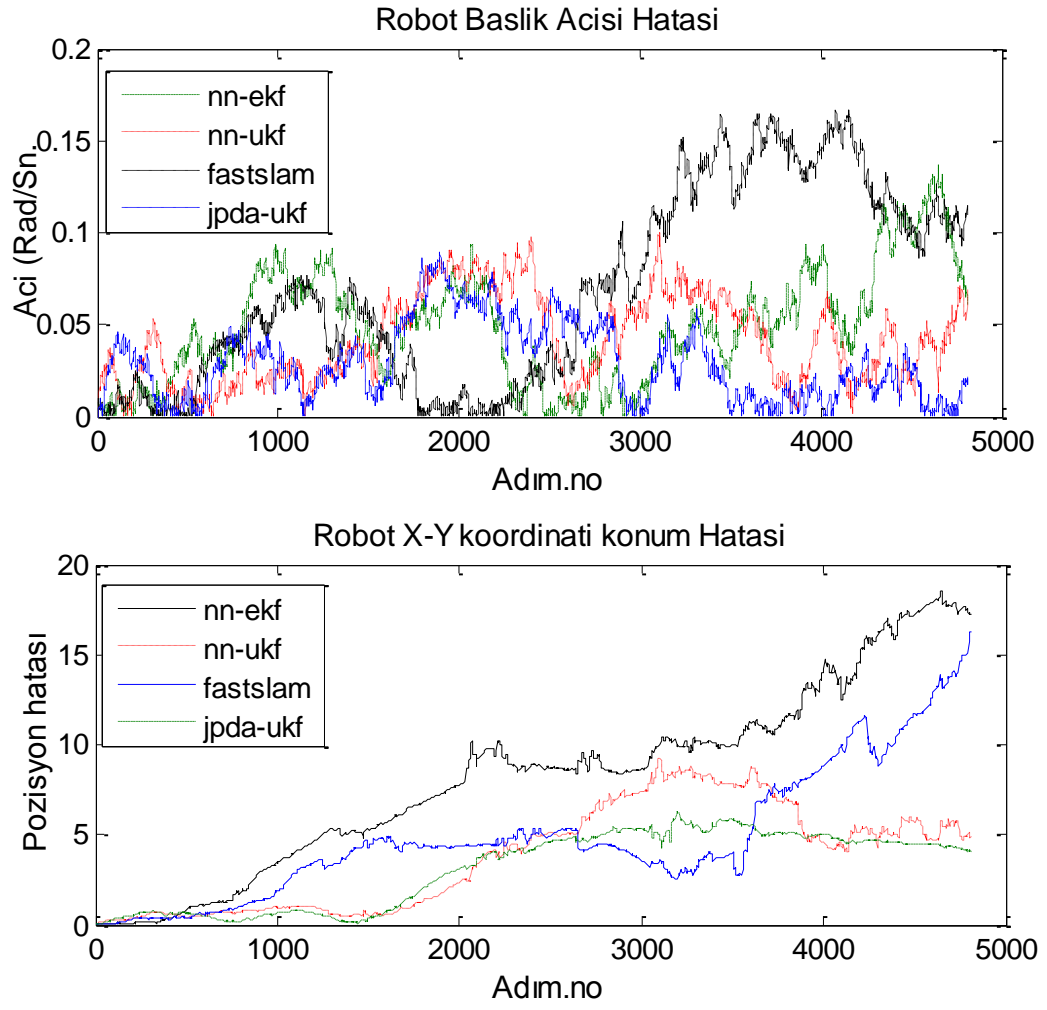
İlgili senaryo (0m,0m) noktasında bilinmeyen bir ortam için başlamakta (220m,-110m) noktasında bitmektedir.

JPDA-UKF süzgecinin birbirine çok yakın nesne işaretli ortamda vermiş olduğu kestirim sonuçları şekil 4.10' da gösterilmiştir. Senaryo II için deneysel çıktılar 30 MCR yapılarak elde edilmiştir.

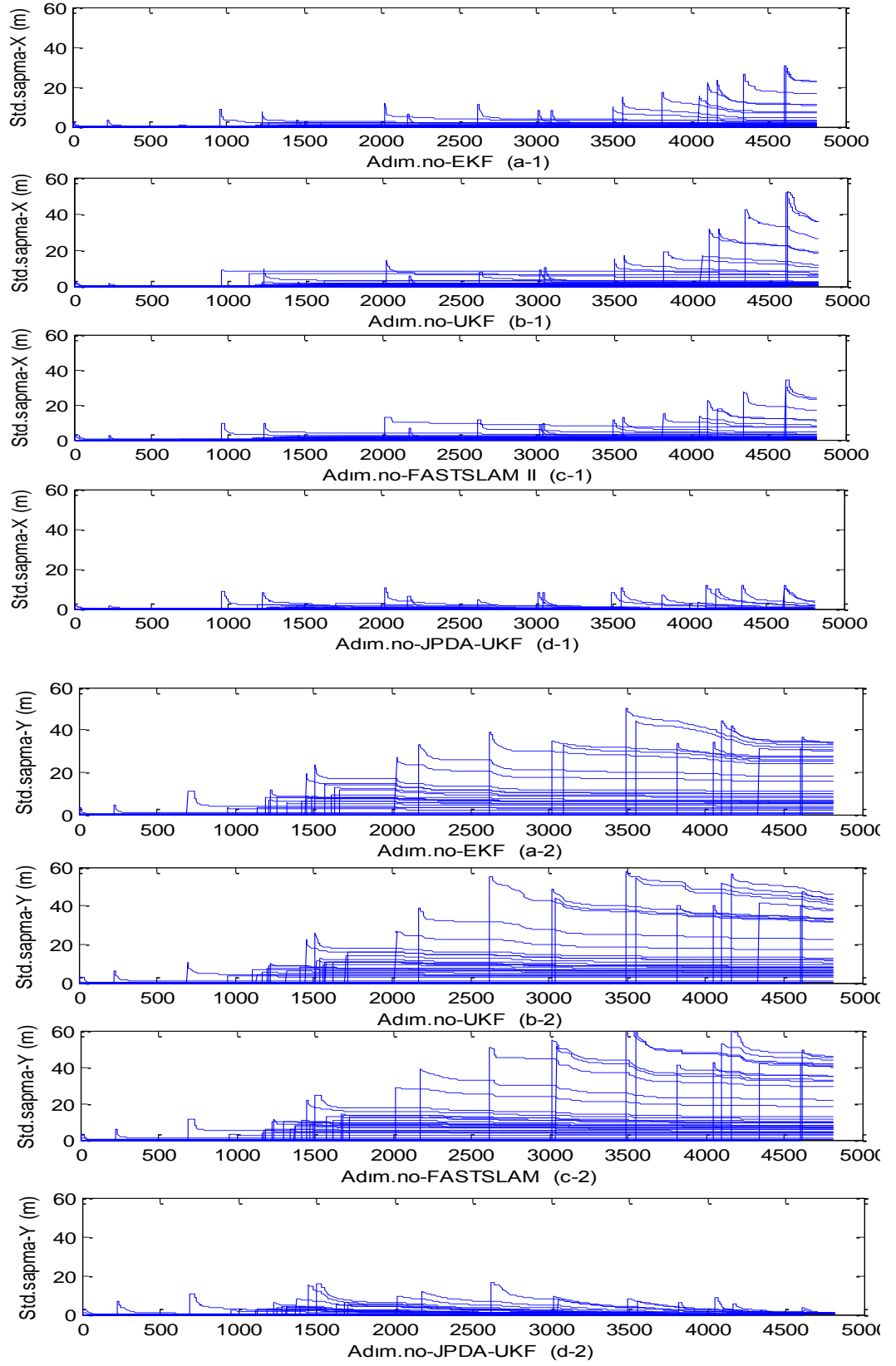


Şekil 4.10 JPDA -UKF algoritması statik parazit yankılı ortamda SLAM problemi için kestirim sonuçları

Şekil 4.10, senaryo I' deki gibi aynı gürültü oranına sahip statik parazit yankılı ortamda JPDA-UKF süzgecinin kestirim sonuçlarını göstermektedir. Süzgeç, gerçek rotadaki 4. düğüme kadar ortalama bir hata ile ilerlemiştir, 4. düğüme parazit yankılarla karşılaşınca kestirim doğruluğunun asgari düzeyde etkilendiği görülmektedir. Senaryo ile ilgili bütün süzgeçlerin konum, açılacak başlık hataları ve nesne işaretleri için hata kovaryansı standart sapma değerleri şekil 4.11 ve 4.12' de verilmiştir.



Şekil 4.11 Süzgeçler robot pozisyon ve başlık açısı ortalama kare hata kestirim sonuçları



Şekil 4.12 Bütün algılanan nesne işaretleri için nesne işaretleri konum kestiriminin bütün süzgeçler için standart sapma değerleri
a. X-koordinatı, b. Y-Koordinatı; a) *NN*-EKF, b) *NN*-UKF, c) FastSLAM II, ve d) JPDA-UKF

Çizelge 4.5 Parazit yankılı ortamda süzgeç hata kareleri ortalamaları.

Süzgeç	Açısal başlık hatası (Radyan)	Konum hatası (m)
<i>NN-EKF</i>	0.3217	3.4630
<i>NN-UKF</i>	0.2045	1.9520
FastSLAM II	0.2623	2.2075
JPDA-UKF	0.1806	1.8634

Şekil 4.11-12 ve çizelge 4.5’ deki sonuçlar, kullanılan dört adet süzgeç için parazit yankılı ortamdaki kestirim sonuçlarını göstermektedir. Sonuçlar senaryo I ile kıyaslandığında özellikle gerçek rotanın 4. düğümüne karşılık gelen 30x30 m’ lik bölgede (şekil 4.10’ da gösterilmiştir) kestiriciler statik parazit yankılarla karşılaşmışlardır. Bu bölge detaylı olarak incelendiğinde; süzgeçlerin kestirim değerlerinin bu bölgeden itibaren belirgin olarak büyüdüğü, yanlış harita oluşumuna ve pozisyon kestirimine yol açtığı gözlenmiştir. Aynı zamanda, algılama esnasında süzgeçler parazit yankıların olduğu bölgeden geçerken nesne işaretlerinin de standart sapma değerlerinde bir miktar artış olduğu, yani belirsizlik durumunun yükseldiği görülmüştür. Bu beklenen bir durumdur; süzgeçlerin performansları bu bölgede olumsuz yönde etkilenmiştir.

Çizelge 4.5’e bakıldığında aynı ortam ve gürültüler göz önünde bulundurulduğunda hata kareleri ortalamalarında bir miktar artış olduğu görülmektedir. Aynı zamanda benzetim çalışmasında, kestirim doğruluğu bütün süzgeçlerde düşerken, işlem sürelerinde belirgin bir değişim olmadığı gözlenmiştir. Öte yandan JPDA-UKF süzgecinin parazit yankılı ortamdan geçerken bir miktar yavaşladığı görülmüştür. Bunun sebebi JPDA-UKF algoritması veri ilişkilendirmesi yaparken kapılar içerisindeki tahmin edilen en yakın komşu kestirim değerine değil de yine kapının içerisindeki bütün olası ölçümleri aday tahmin değeri olarak gördüğünden süzgeç bu bölgede işlem süresi olarak diğer süzgeçlerden daha fazla çalışmakta, bu ise olumsuz etkilenmesine yol açmaktadır. Fakat bu durumun süzgecin kestirim doğruluğuna yansımadağı görülmektedir.

Senaryo II' de ortam gürültüsünün yoğun olması, yani *SNR* değerinin yüksek olması, süzgeçlerin parazit yankıdan daha az etkilendiğini deneysel çalışmalar esnasında göstermektedir. *SNR* değerinin düşük olduğu parazit yankılı ortamda ise süzgeçlerin daha fazla etkilendiği görülmektedir. Bunların yanı sıra senaryo I ile benzer kestirim sonuçları elde edilmiştir; nesne konumlarının yakın olduğu (belirsizliğin yüksek olduğu) durumlarda süzgeçlerin performansının düştüğü görülürken, JPDA-UKF süzgecinde minimum etkiyle kestirim sonuçlarının sağlandığını göstermiştir. Şekil 4.12-(a, b)' de süzgeçlerin senaryo II için nesne işareti standart sapma değerleri görülmektedir. Yine ilk senaryoda olduğu gibi bu parazit yankılı ortamda da kullanılan süzgeçlerin içerisinde en uygun standart sapma değerini minimize edebilme kabiliyeti olarak JPDA-UKF gösterilebilir.

4.3 Optimizasyon Yöntemlerine Dayalı FastSLAM yaklaşımları

4.3.1 Karakök kokusuz FastSLAM yöntemi ile parçacık Sürü Optimizasyonu – pasif toplama (PSOPC)

Karakök kokusuz FastSLAM modeli daha önceki çalışmalarda anlatılmıştır (Ankışhan vd. 2014). Yöntem araç pozisyonu, nesne pozisyonu ve parçacıkların ağırlıklarını kokusuz Kalman süzgeciyle hesaplamaktadır. Amaç Cholesky faktörizasyon hesabından kaynaklı olarak kovaryans matrisinin, süzgecin her bir döngüsünde karakökünün alınmasından kaynaklı zaman kaybının önüne geçilmesidir. Böylece işlem süresi kısaltılmış, kovaryans matrisinin pozitif yarı tanımlı olma durumunun garanti altına alınmıştır.

Çalışmanın bu bölümünde daha önceden (Bailey T, 2002) tarafından geliştirilen yazılımdan yararlanılarak bir senaryo üretilmiştir. İki adet benzetim seti oluşturulmuştur. Birinci benzetim setinde sistem parametreleri; araç hızı 3m/sn, araç tekerleri arası mesafe 4m ve kontrol frekansı 40Hz olarak alınmıştır. Kontrol ve ölçüm gürültü varyansları

$$Q = \begin{bmatrix} 1.0^2 & 0 \\ 0 & [10 * (\pi / 180)]^2 \end{bmatrix} \quad (4.5)$$

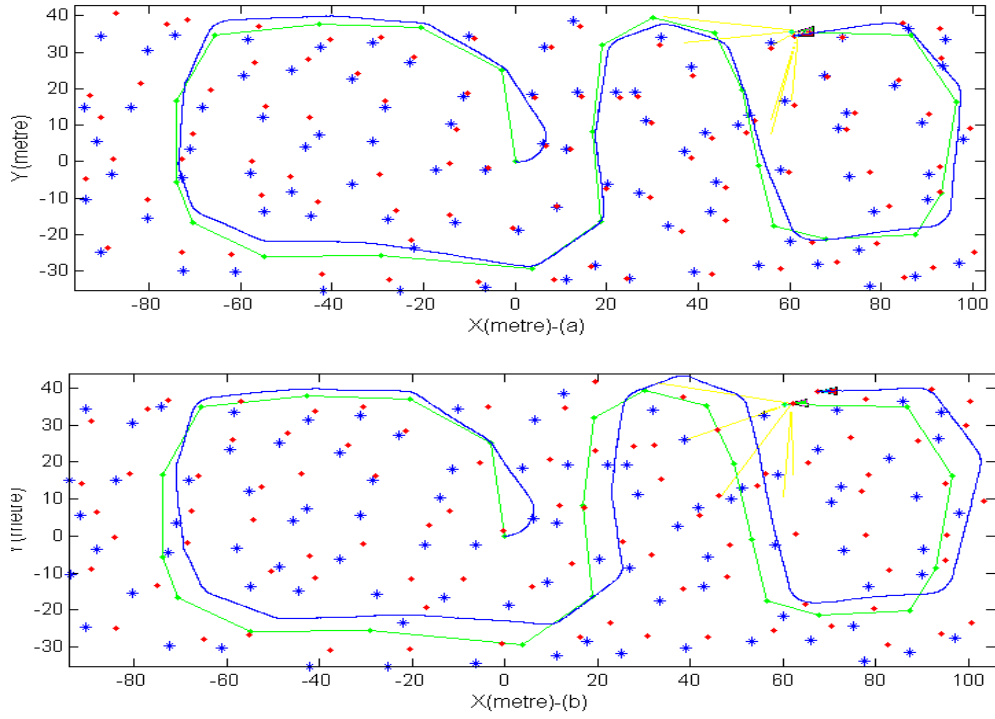
$$R = \begin{bmatrix} 1.2^2 & 0 \\ 0 & [12 * (\pi / 180)]^2 \end{bmatrix} \quad (4.6)$$

olarak seçilmiştir. Gözlem parametreleri ise maksimum lazer algılama mesafesi 30m araç tarama frekansı arası süre 5Hz' dir. Deneysel çalışmanın bu bölümünde veri ilişkilendirmesinin bilindiği kabul edilmiştir. PSOPC için başlangıç parametreleri çizelge 4.6' da verilmiştir.

Çizelge 4.6 PSOPC için başlangıç parametreleri

Sürü büyüklüğü (Swarm)	20
Üretim Max. Sayısı	100
Problem boyutu	3
İvmelendirme sabitleri C1, C2	2.0
Pasif toplama sabiti C3	2.0
Atalet ağırlığı (w)	0.7

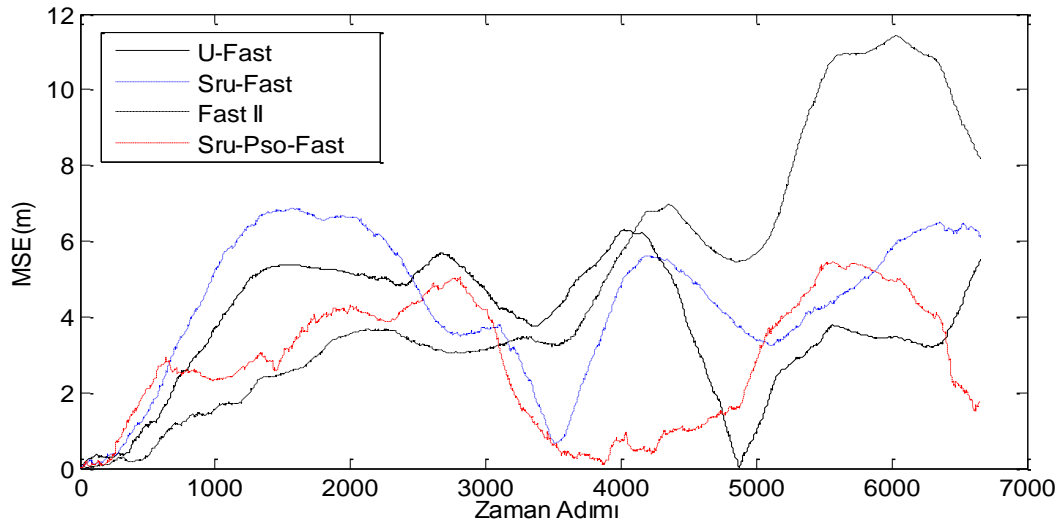
Oluşturulan ilk senaryo için FastSLAM II, U-FastSLAM, SRU-FastSLAM ve SRU-PSOPC-FastSLAM metotlarının sonuçları kıyaslanmıştır. Şekil 4.13 SRU-FastSLAM ve SRU-PSOPC-FastSLAM'in senaryo için kestirim sonuçlarını göstermektedir.



Şekil 4.13 Hesaplanan ve gerçek araç rotası ile hesaplanan ve gerçek nesnelere:

a. SRU - Kalman süzgeci tabanlı PSOPC optimizasyonlu (SRU-PSOPC)- FastSLAM, b. SRU-FastSLAM

Burada yeşil çizgiler ve mavi nesnelere gerçek rota ve gerçek nesne pozisyonlarını göstermektedir. Mavi kalın çizgi kestirilen araç pozisyon ortalaması ve kırmızı noktalar kestirilen nesne pozisyonlarıdır.



Şekil 4.14 Araç pozisyonununun X-Y koordinatına bağlı hata kareleri ortalaması sonuçları

Araç pozisyon hataları ve başlık açısı hatalarının sonuçları çizelge 4.7’ de ve şekil 4.13, 4.14’ de görülmektedir. Sonuçlar 30 MCR ile elde edilmiştir. Şekil 4.13 incelendiğinde SRU–PSOPC–FastSLAM sonuçlarının SRU–FastSLAM’ den daha tutarlı olduğu gözlenmektedir. SRU-PSOPC–FastSLAM sonuçları, gerçek rotaya daha yakın kestirim sonuçları vermiştir. Çünkü parçacıklar PSOPC tarafından yüksek olasılıklı bölgeye kaydırılmış ve böylece zamanla dejenerasyona uğrama problemi ortadan kaldırılmıştır. Durumun sonuçları deneysel çıktılara yansımıştır. Sonuç olarak gerçek pozisyona daha yakın kestirim sonuçları elde edilmiştir.

Çizelge 4.7 Metotların ilgili senaryo için çıktıları

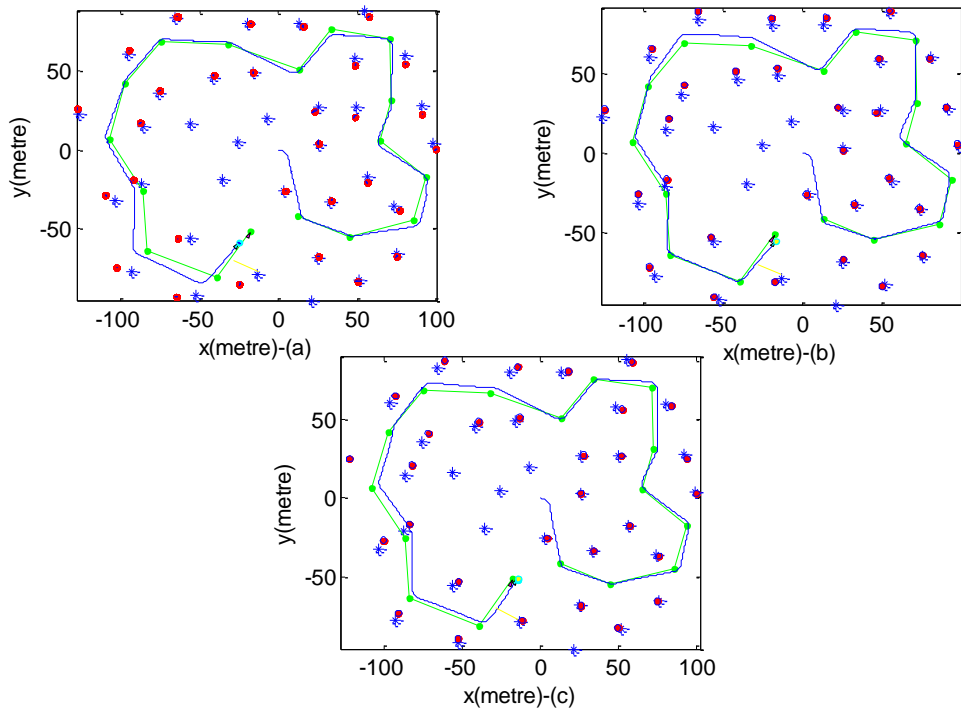
Metotlar	MSE (m)		(radyan)
	ortalama	varyans	Başlık hatası
FastSLAM II	4.8968	2.1537	0.3721
U-FastSLAM	4.3674	1.4186	0.2164
SRU-FastSLAM	3.9615	1.0050	0.1788
SRU-PSOPC-FastSLAM	2.7992	0.7021	0.1413

Ek olarak parçacıklarda gerçekleştirilen bu iyileştirme, nesnelerin kestirim sonuçlarını da etkilemiştir. Çünkü araç pozisyonu SLAM için ne kadar iyi kestirilirse etrafın haritası o ölçekte daha başarılı elde edilebilmektedir (Montemerlo 2003). İlk benzetim için üretilen senaryoda önerilen diğer metotların sonuçlarıyla da karşılaştırmalar yapılmıştır. Sonuçlar şekil 4.14’ te gözlenmektedir. Sonuçlardan SRU–FastSLAM için optimizasyon algoritmasının daha iyi performans sağladığı görülmektedir.

U–FastSLAM metodunun sonuçları SRU tabanlı yöntemlerle kıyaslandığında, gerçek rotaya SRU tabanlı yöntemler kadar yakın olmadığı, hata kare ortalamasının daha yüksek çıktığı gözlenmiştir. Bunun sebebi, UKF yaklaşımı FastSLAM uygulamasında yine parçacık sayısı seçimi problemi olduğu söylenebilir. (Kim vd. 2008) çalışmalarında her ne kadar parçacık sayısını azalttıklarını söyleseler de tez çalışmasında daha anlamlı sonuçlar elde edilmiştir. U-FastSLAM yaklaşımında hem Cholesky çarpanlarına ayırma yönteminden kaynaklı hem de kovaryans matrisinin sistemin gözlemlenebilir olması

için pozitif yarı tanımlı olma zorunluluğu vardır. Karakök kokusuz yapısıyla UKF tabanlı FastSLAM yaklaşımı, bu problemi (Merwe 2001)' de olduğu alternatif bir yaklaşım olarak önermiş ve deneysel çalışmalarda göstermiştir.

İkinci senaryoda veri ilişkilendirmesinin bilinmediği varsayılmıştır. Geliştirilen senaryo için benzetim sonuçları 30 MCR ile elde edilmiştir. Senaryo 2 için sonuçlar şekil 4.15 ve Çizelge 4.8' de verilmektedir.



Şekil 4.15.a. U-FastSLAM, b. SRU-FastSLAM, c. SRU-PSOPC-FastSLAM için kestirim sonuçları

Burada yeşil çizgiler gerçek rota ve mavi yıldızlar gerçek nesne pozisyonlarını vermektedir. Mavi kalın çizgi kestirilen araç rotası ve kırmızı noktalar ise kestirilen nesne pozisyonlarıdır.

Çizelge 4.8 Önerilen SLAM algoritmalarının performansları

Metotlar	Parçacık sayısı	Pozisyon hatası (m)	Başlık açısı hatası (radyan)
U-FastSLAM	50	3.6970	0.0833
SRU-FastSLAM	5	3.5327	0.0475
SRU-PSOPC-FastSLAM	5	2.8246	0.0667

Bu benzetim çalışmasında SRU-FastSLAM ve SRU-PSOPC-FastSLAM yaklaşımları 5 adet parçacık kullanmasına rağmen U-FastSLAM yaklaşımı 50 adet parçacık kullanmıştır. Sonuçlardan görülmektedir ki; en uygun kestirim sonuçları SRU-PSOPC-FastSLAM yaklaşımıyla sağlanmıştır. Daha az sayıda parçacıklar U-FastSLAM' den daha iyi kestirim sonuçları vermiştir.

4.3.2 Farksal Evrim tabanlı CDKF-FastSLAM yaklaşımları

Çalışmanın bu bölümünde simülasyon çalışmaları 4 aşamada gerçekleştirilecektir. İlk aşamada üretilen senaryo için FastSLAM II, U-FastSLAM, SR-CDKF-Aid-FastSLAM, SR-CDKF-PF-FastSLAM ve SR-CDKF-FastSLAM metotları karşılaştırılacaktır. Harita için veri ilişkilendirmesinin bilindiği varsayılmıştır. İkinci aşamada U-FastSLAM, SR-CDKF-FastSLAM ve DE-SR-CDKF-FastSLAM metotları yeni bir senaryo için karşılaştırılmıştır. Bu aşamada veri ilişkilendirmesinin bilinmediği varsayılmış ve yöntem olarak en yakın komşuluk metodu (Nearest neighbour – NN) kullanılmıştır. Üçüncü benzetim için önerilen yöntemler ve süzgeç tutarlılıkları tartışılmıştır. Burada DE- SR-CDKF-FastSLAM sonuçları U-FastSLAM ve FastSLAM II metotlarıyla kıyaslanmıştır. Son çalışmada ise gerçek zamanlı bir veri setinde SR-CDKF-FastSLAM, U-FastSLAM ve FastSLAM II sonuçları kıyaslanmıştır.

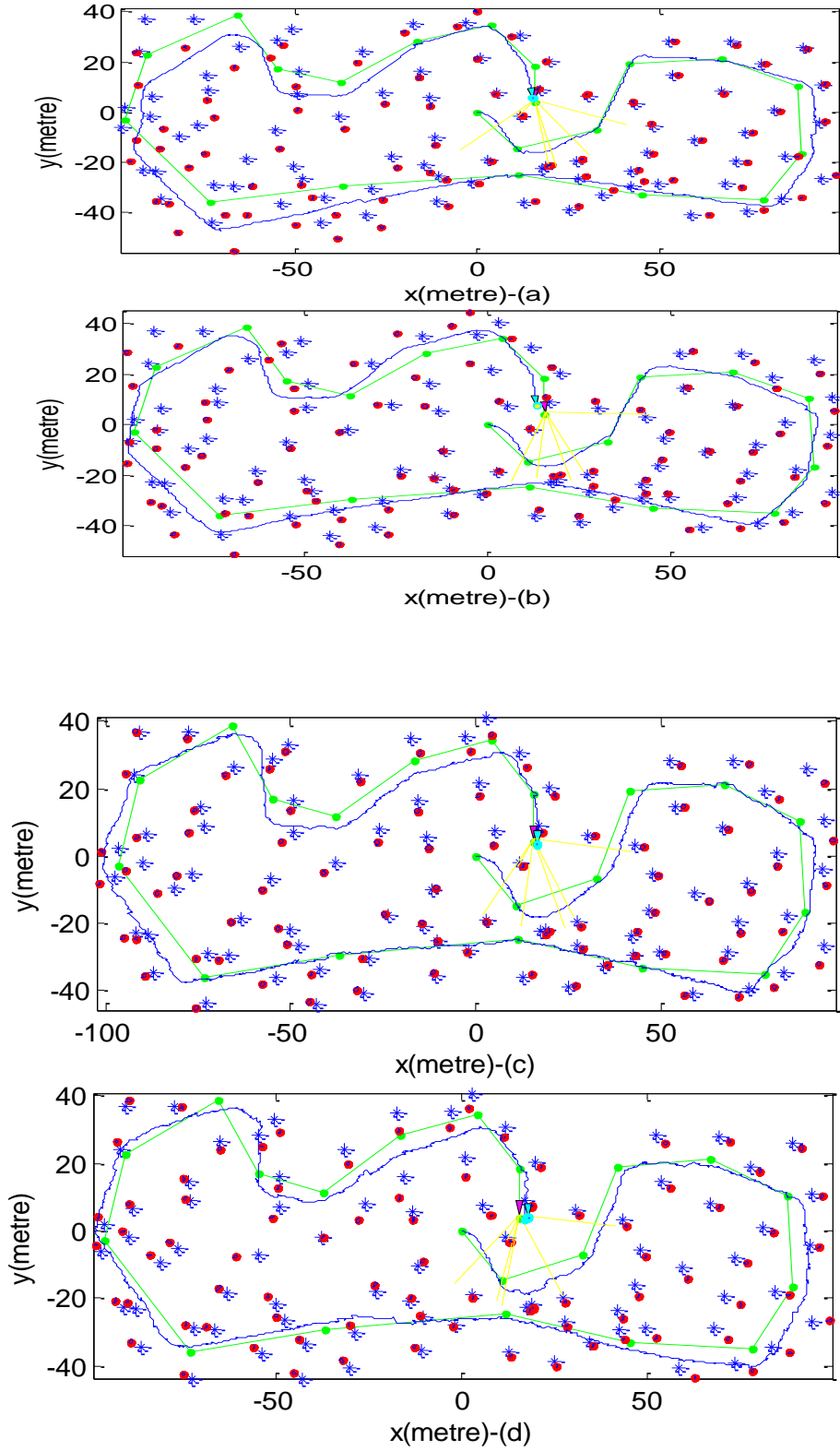
Benzetim 1. Metotların kıyaslanması

Bu simülasyon çalışmasında (Bailey 2003) yazılımından yararlanılarak senaryolar üretilmiştir. Senaryolar için seçilen parametreler; araç hızı 3m/sn, araç dingil mesafesi 4m ve kontrol frekansı 40 Hz olarak alınmıştır. Kontrol ve ölçüm gürültü matrisleri,

$$Q = \begin{bmatrix} 1.0^2 & 0 \\ 0 & [10 * (\pi / 180)]^2 \end{bmatrix} \quad (4.7)$$

$$R = \begin{bmatrix} 1.2^2 & 0 \\ 0 & [12 * (\pi / 180)]^2 \end{bmatrix} \quad (4.8)$$

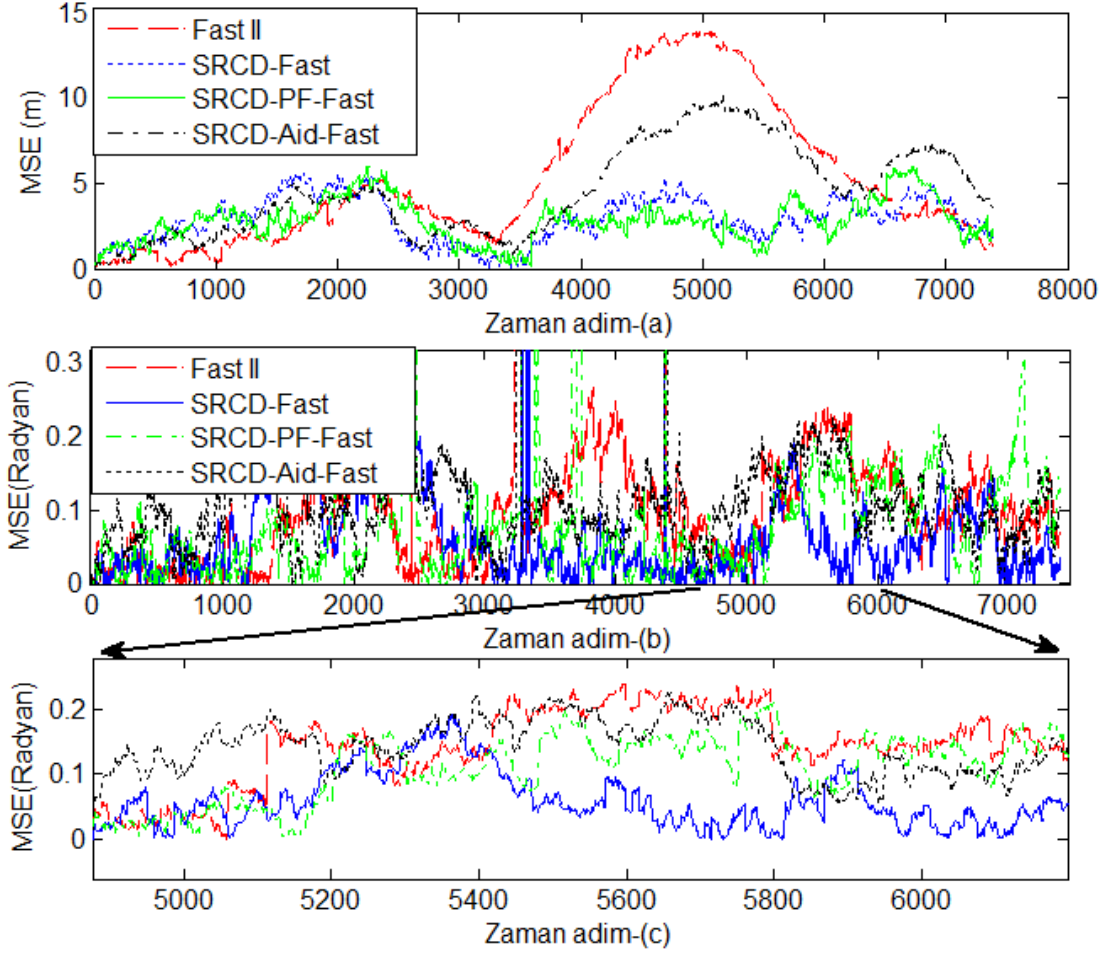
olarak seçilmiştir. Maksimum lazer mesafesi 30m ve araç lazer tarama frekansı 5Hz' dir. İlgili senaryo için elde edilen sonuçlar şekil 4.16.a, b, c, d ve araç pozisyon ve başlık açısı ortalama kare hataları şekil 4.17 ve çizelge 4.9' da verilmiştir.



Şekil 4.16 Hesaplanan ve gerçek araç rotaları ve nesne pozisyonları:

a. FastSLAM II, b. SR-CDKF-Aid-FastSLAM, c. SR-CDKF-PF-FastSLAM ve d. SR-CDKF-FastSLAM

Burada yeşil çizgi ve mavi yıldızlar gerçek rota ve nesne pozisyonlarını vermektedir. Mavi kalın çizgi kestirilen rota ve kırmızı noktalar ise kestirilen nesne pozisyonlarını vermektedir.



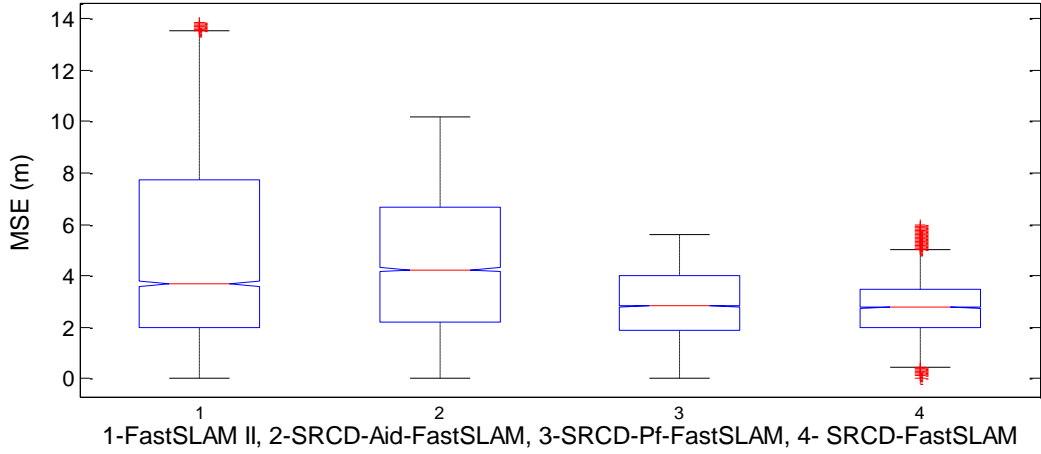
Şekil 4.17 Yöntemler için araç pozisyon ve başlık açısı ortalama kare hataları (MSE):
 a. pozisyon hatası, b. başlık açısı hatası (Radyan), c. kısmi başlık açısı hatası (Radyan)

Şekil 4.17'deki SRCD, SR-CDKF'nin kısaltmasıdır. Benzetim sonuçlarından en büyük hata kare ortalamasına sahip yöntemin FastSLAM II olduğu söylenebilir. Özellikle 3500-6000 adımları arası hata oranlarının yükseldiği gözlenmiştir. Robotun/Özerk aracın FastSLAM II için pozisyon hatası diğer yöntemlere kıyasla daha yüksek çıkmıştır. Bu yükselme durumunun tamamıyla EKF' den kaynaklandığı düşünülmektedir. Şekil 4.16 ve 4.17 SR-CDKF-FastSLAM tarafından gerçekleştirilen kestirim sonuçlarını göstermektedir. Ortalama kare hataları detayları ise Çizelge 4.9' da verilmiştir.

Çizelge 4.9 Metotların ortalama kare hataları

Metotlar	Pozisyon MSE (ortalama) [m]	Pozisyon MSE (Varyans)	Başlık açısı (Radyan)
FastSLAM II	5.2617	17.2117	0.1784
SRCD-Aid-FastSLAM	4.5262	6.9328	0.2338
SRCD-PF-FastSLAM	2.8735	1.7646	0.1453
SRCD-FastSLAM	2.8211	1.5997	0.1211

SLAM uygulamaları için EKF tabanlı FastSLAM metotlarında doğusallaştırma, süzgecin kestirim doğruluğunda zamanla ortalama sapması ve ıraksama problemine yol açmaktadır (Bailey 2002, 2003). Aynı durum deneysel çıktılarda da gözlenmiştir. Bu durum, parçacık süzgeçlerindeki olasılığı düşük parçacıkların zamanla ağırlıklarının sifira yaklaşmasına ve örnekleme çeşitliliğinin zamanla kaybolmasına yol açmaktadır. Bu yüzden robot/özerk araç pozisyon kestirimi ve çevre haritası çıkarımının doğru bir şekilde elde edilmesinin önüne geçilmiş olmaktadır. 4.16 ve 4.17’ de elde edilen sonuçlarda kestirim performansına etkileri gözlenmektedir. Amaçlanan SRCD–FastSLAM tabanlı yaklaşım ve diğer metot sonuçları şekil 4.18’ de ‘*kutu-plot*’ çiziminde gösterilmiştir.



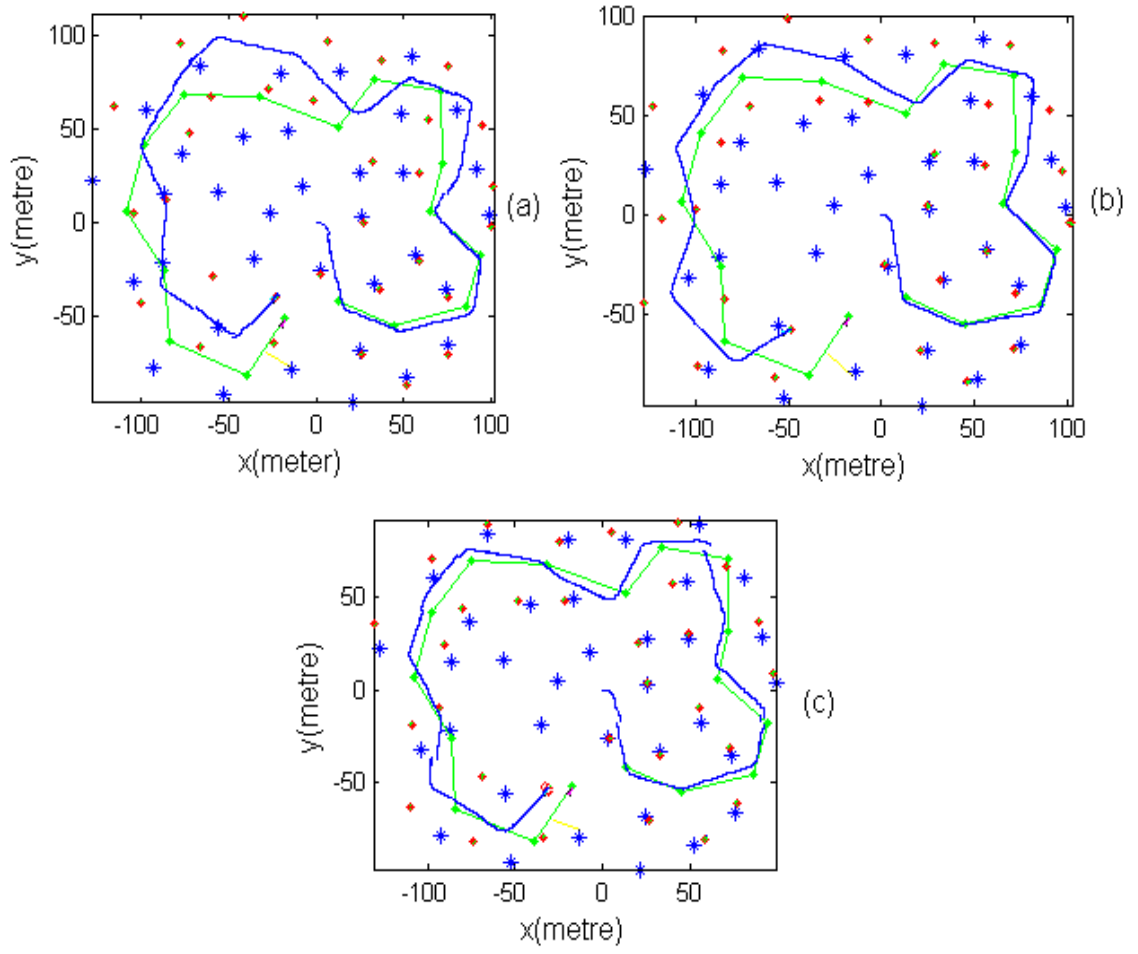
Şekil 4.18 Yöntemlerin *Kutu-Plot* çizimleri

1. FastSLAM II, 2. SRCD-Aid-FastSLAM, 3. SRCD-PF-FastSLAM, 4. SRCD-FastSLAM

SRCD-FastSLAM tabanlı yaklaşımlar incelendiğinde SRCD-PF ve SRCD-FastSLAM yaklaşımlarının birbirine yakın sonuçlar verdiği gözlenmiştir. SRCD-Aid-FastSLAM yaklaşımının sonuçları her ne kadar FastSLAM II' den iyi çıksa da önerilen diğer süzgeçler kadar başarılı değildir. Burada araç pozisyon kestirimi daha iyi çıkmasına rağmen nesne pozisyon kestirimlerinde aynı başarı elde edilemediği gözlenmiştir. Araç başlık açıları hatası incelendiğinde SRCD-Aid-FastSLAM'in en yüksek hataya sahip olduğu görülmektedir. Amaçlanan FastSLAM metodunda FastSLAM II' den her koşulda iyi sonuçlar elde edildiği gözlenmiştir.

Benzetim 2. DE tabanlı metodun sonuçları

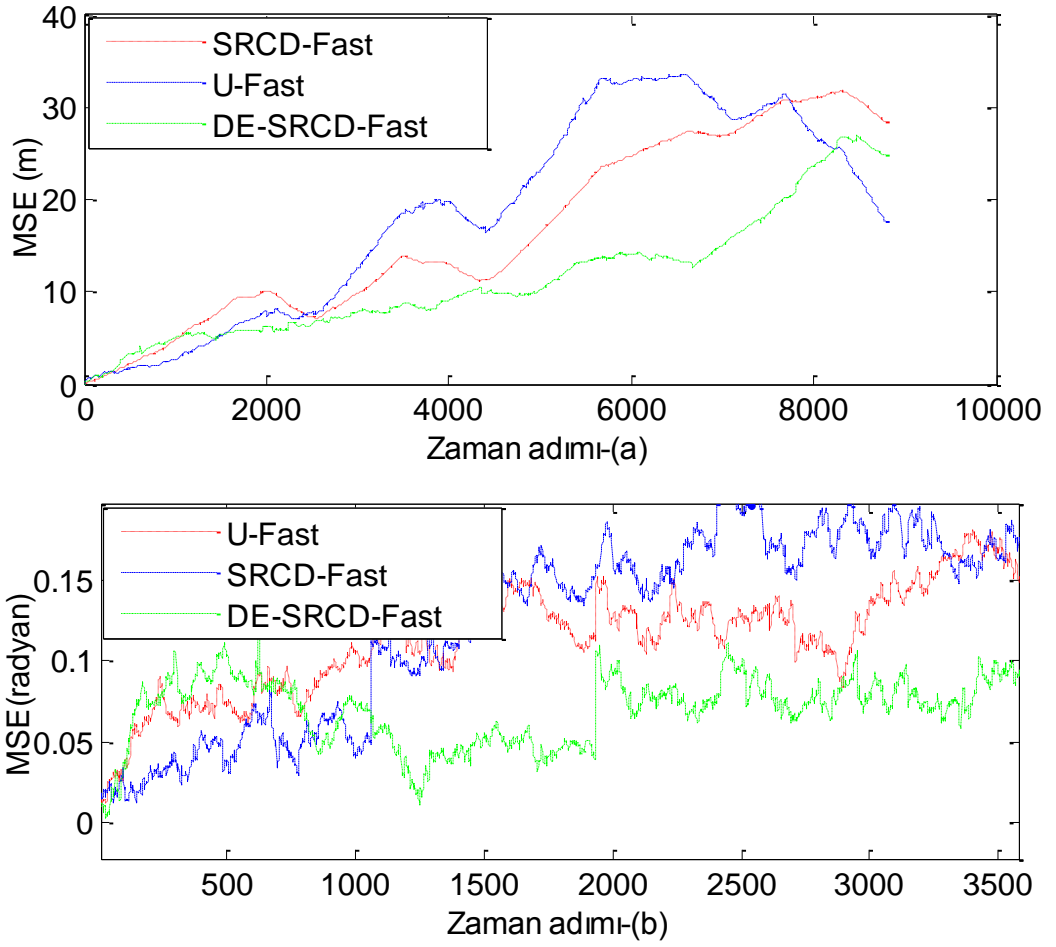
Bu simülasyonda veri ilişkilendirmesinin bilinmediği farz edilmiştir ve yöntem olarak NN algoritması kullanılmıştır. Simülasyonda, DE-SRCD-FastSLAM sonuçları SRCD-FastSLAM ve U-FastSLAM metotlarıyla karşılaştırılmıştır. Bailey (2003) tarafından MATLAB ortamında geliştirilen yazılımdan yararlanılarak bir senaryo üretilmiştir. SRCD-FastSLAM ve U-FastSLAM yaklaşımları için parçacık sayısı 30 seçilirken DE-SRCD-FastSLAM için 3 olarak alınmıştır. Sistem ve gözlem parametreleri benzetim 1' deki ile aynıdır. DE algoritması için ise çaprazlama olasılığı (Cr) 0.8 ve mutasyon ölçekleme faktörü (F) 0.4 olarak alınmıştır. Senaryo sonuçları şekil 4.20 ve MSE sonuçları ise şekil 4.21 ve Çizelge 4.10' da görülmektedir.



Şekil 4.19 Yöntemler için kestirilen, gerçek rota bilgisi ve nesne pozisyonları bilgisi

a. U-FastSLAM, b. SRCD-FastSLAM, c. DE-SRCD-FastSLAM

Burada yeşil çizgi gerçek rota, mavi kalın çizgi kestirilen rotayı gösterir. Mavi yıldızlar gerçek nesne pozisyonları, kırmızılar ise kestirilen nesne pozisyonlarını verir.



Şekil 4.20 Metotların MSE sonuçları:

a. U-FastSLAM, SRCD-FastSLAM ve DE-SRCD-FastSLAM pozisyon hataları, b. başlık açısı hataları

Çizelge 4.10 Metotların ortalama kare hataları

Metotlar		U-FastSLAM	SRCD-FastSLAM	DE-SRCD-FastSLAM
Pozisyon (ortalama)[m]	MSE	18.4513	16.5021	11.2039
Pozisyon (varyans)	MSE	120.0046	95.3682	42.8834
Başlık açısı (radyan)	hatası	0.5669	0.5156	0.4157

Şekil 4.20' de veri ilişkilendirmesinin bilinmediği kabul edilmiştir. Sonuçlar incelendiğinde en uygun sonuçları DE tabanlı önerilen yaklaşım verdiği gözlenmiştir. Parçacıkların DE tarafından iyileştirilmesinin böyle bir etki oluşturduğu

düşünülmektedir. Burada, parçacıklar DE yardımıyla düşük olasılıklı bölgeden yüksek olasılıklı bölgeye iterasyonla kaydırılmıştır. Böylece süzgecin kestirim performansındaki iraksama problemi ve parçacık ağırlıklarının sıfıra gitme problemi ortadan kaldırılmış olmaktadır. Bu sonuç metodun performansına yansımıştır ve kestirim doğruluğunu artırmıştır. Böylece çevrenin haritasının oluşturulması daha anlamlı hale gelmiştir. Toplam hata sonuçları Çizelge 4.10’da verilmiştir. Çizelge 4.10’da, MSE değerleri ve varyans değerlerinin SRCD ve U-FastSLAM için aynı oranda büyüdüğü gözlenmektedir. Öte yandan DE-SRCD-FastSLAM yaklaşımlarında bu oran daha düşük çıkmıştır.

Deneysel çalışmanın sonuçları incelendiğinde EKF ve UKF tabanlı yaklaşımlara nazaran SRCD tabanlı yaklaşımların daha başarılı olduğu ve optimizasyon yöntemleriyle iyileştirilerek daha anlamlı sonuçlar elde edildiği gözlenmiştir.

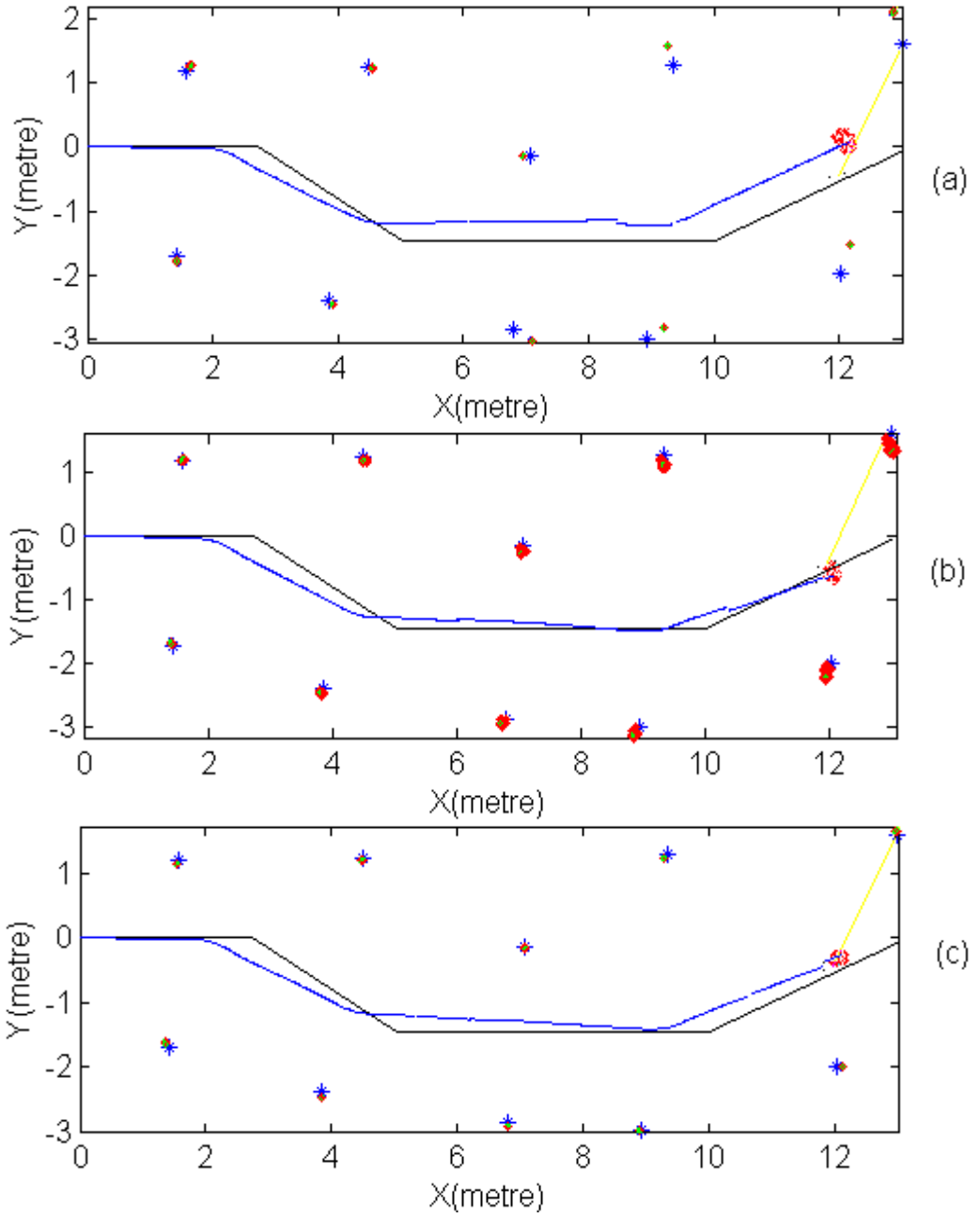
Önerilen SRCD tabanlı yöntemin, UKF tabanlı yöntemlerden bazı avantajlara sahip olduğu bilinmektedir (Merwe 2004). Süzgecin her bir döngüsünde, sigma noktaları hesabında, her bir ayrı süzgeç modeli için 3 adet ölçekleme parametresi (α , β , ve λ ,) ayarı gerekmektedir. Öte yandan SRCD’ de böyle bir ayar söz konusu değildir. Stirling’s interpolasyon yöntemine bağlı olarak bu işlem tek bir ölçekleme parametresi ve Gauss dağılımlar için tek bir değerle çözülmüştür. Ayrıca hata kovaryans matrisinin böylece pozitif yarı tanımlı olma durumu da SRCD süzgeci için ortadan kaldırılmıştır. (Kim vd. 2008, 2009) çalışmalarında her ne kadar U-FastSLAM için parçacık sayısı problemini çözdüklerini iddia etseler de deneysel çalışmalar bunun hala bir problem olduğunu göstermiştir. Bir optimizasyon algoritmasına ihtiyaç duyulduğu gözlenmiştir.

Benzetim çalışmanın bu kısmında DE optimizasyon algoritması olarak tercih edilmiştir. Literatür incelendiğinde DE, GA ve PSO’ya göre daha az ayarlama parametresine ihtiyaç duyduğu görülmektedir (Vesterstrom 2004). Öte yandan popülasyondaki çeşitliliğin bu yöntemle daha başarılı bir şekilde artırıldığı yine daha önceki çalışmalardan gözlenmektedir (Das ve Suganthan 2011). Diğer bir avantajı ise PSO’ dan daha gürbüz bir şekilde yakınsama özelliğine sahip olmasıdır.

Benzetim 3. DE-SRCD-FastSLAM yaklaşımının tutarlılığı

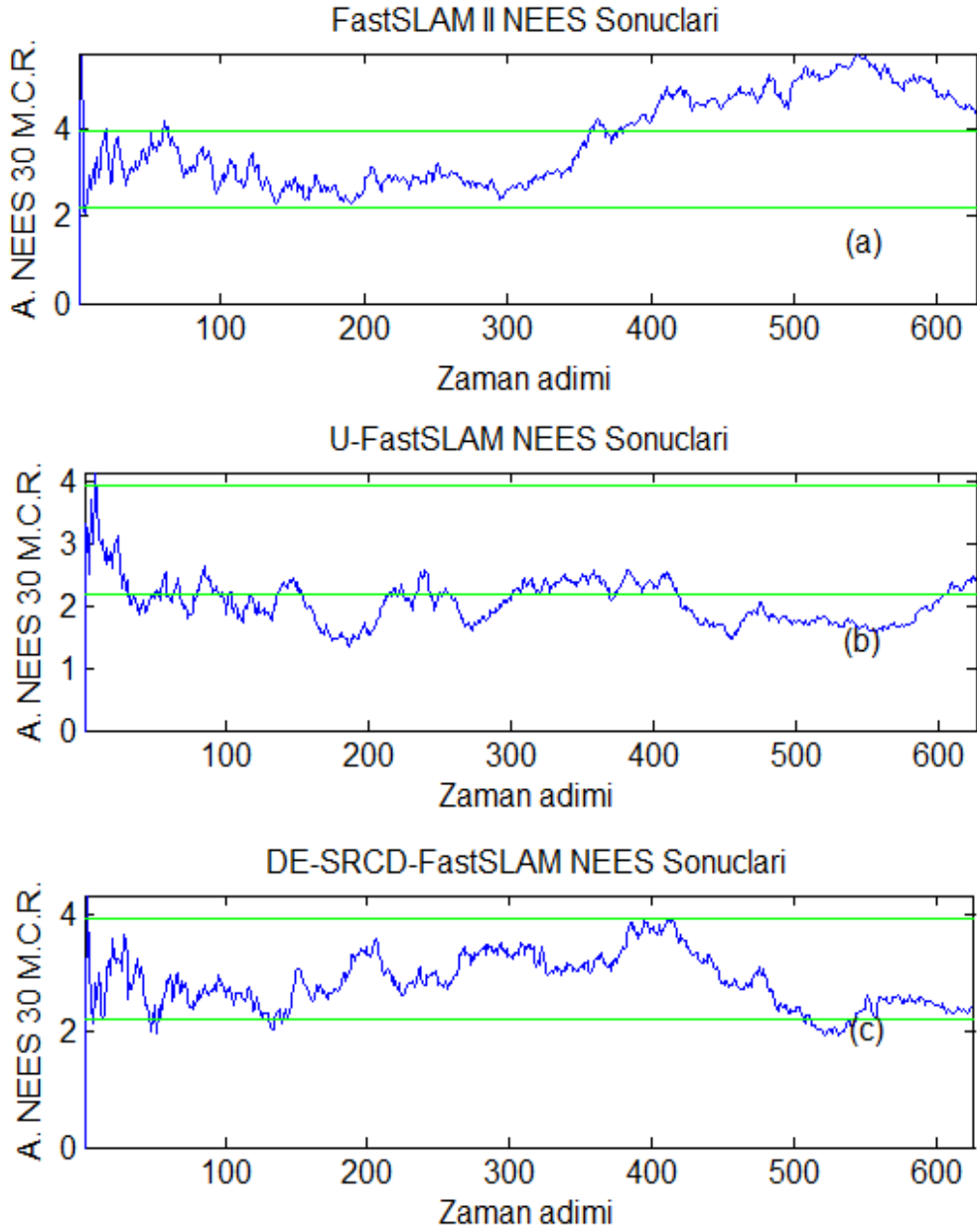
Çalışmanın bu bölümünde normalleştirilmiş kestirim hata karelerine (NEES) bağlı olarak önerilen DE-SRCD tabanlı FastSLAM yaklaşımının tutarlılığı değerlendirilecektir. Çalışmada iki taraflı 95% güven aralığı dikkate alınmış ve 30 MCR her bir metot için yapılmıştır. İlgili geçerlilik aralığı 2.19 ve 3.93 arasındadır.

Burada araç dingil mesafesi 0.26m ve araç hızı 0.8m/sn olarak seçilmiştir. Kontrol frekansı 40Hz ve algılayıcı frekansı 5Hz' dir. Kontrol gürültüleri 0.4 m/sn hızında ve 4° teker açıları olarak alınmıştır. Ölçüm gürültüsü ise 0.2m/sn ve 2° menzil olarak seçilmiştir. Algılayıcı maksimum mesafesi 5m ve ön görüş açısı 180° ' dir. Üretilen senaryo ve metotların kestirim performansları şekil 4.21 ve 4.22' de verilmiştir.



Şekil 4.21.a. FastSLAM II, b. U-FastSLAM ve c. DE-SRCD-FastSLAM kestirim sonuçları

Burada siyah rota gerçek kalın mavi ise kestirilen rotadır. Mavi yıldızlar gerçek nesne pozisyonları, kırmızılar ise kestirilen değerleridir. Bu çalışma için FastSLAM II ve U-FastSLAM için 100 parçacık, DE-SRCD-FastSLAM için ise 10 parçacık kullanılmıştır.



Şekil 4.22 Metotların geçerlilik grafikleri:

a. FastSLAM II, 100 parçacık, b. U-FastSLAM, 100 parçacık ve c. DE-SRCD-FastSLAM, 10 parçacık

Benzetim çalışmasında durum kestirimi ve belirsizlikleri üzerinden grafikler elde edilmiştir. Görülmektedir ki metot karakteristikleri kullanılan süzgeç modeline bağlı olarak değişmektedir. NEES sonuçları U-FastSLAM ve DE-SRCD-FastSLAM için yaklaşık olarak aynı çıkmıştır. Yine de en uygun aralıkta önerilen metodun çıktığı kanısına varılabilir. Amaçlanan metot senaryo adımı boyunca güven aralığında daha tutarlı kalmayı başarmıştır.

Benzetim 4. Dış çevre uygulaması-Victoria parkı, Sydney

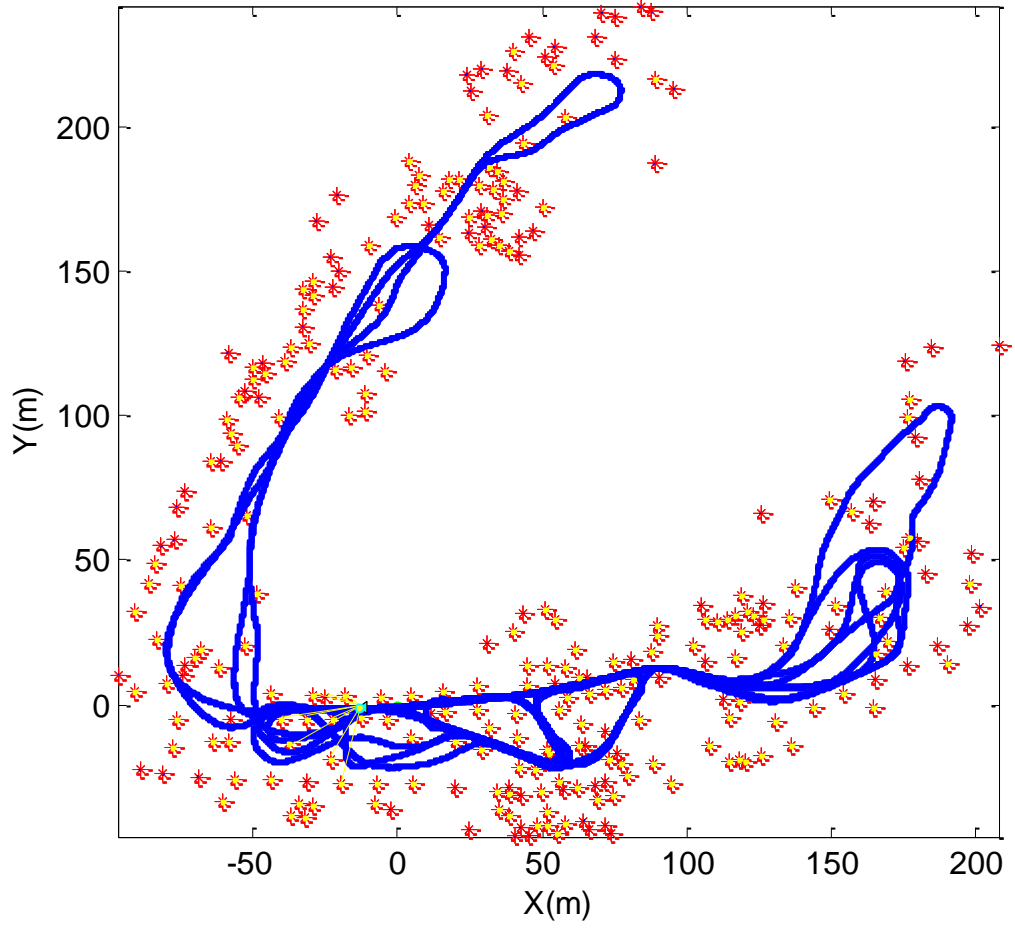
Çalışmanın bu kısmında SLAM dış çevre uygulaması gerçekleştirilecektir. SRCD-FastSLAM yaklaşımı ilk defa FastSLAM II ve U –FastSLAM yöntemleriyle Victoria park verisi için karşılaştırılmıştır. Victoria park verisi GPS’ le birlikte şekil 4.23’ te verilmiştir.



Şekil 4.23 Victoria Park, Sydney, Australia, GPS verisiyle birlikte gösterilmiştir

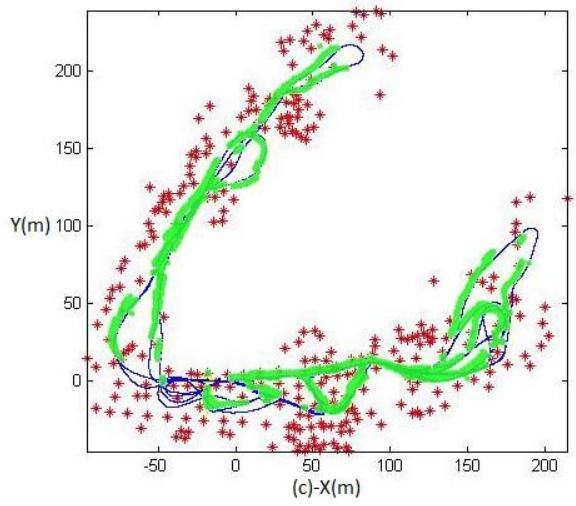
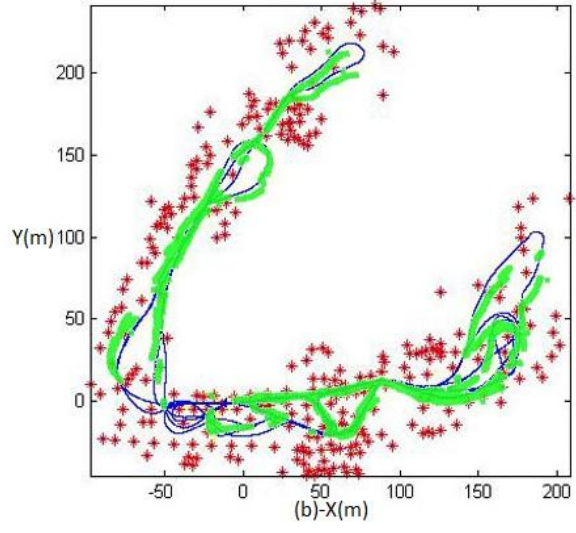
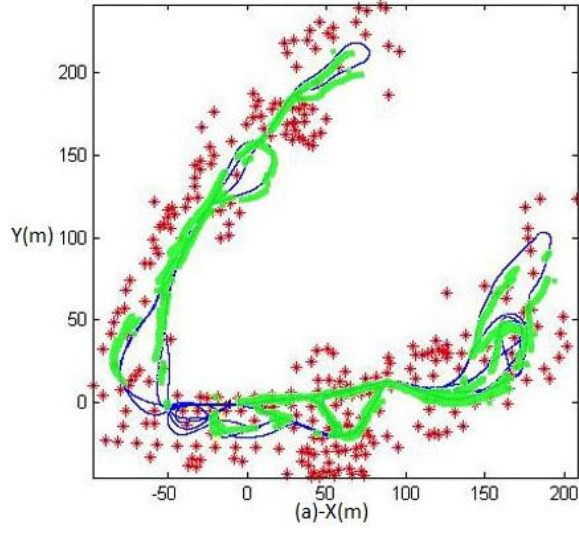
Veri seti (Guivant vd. 2000) tarafından elde edilmiş olup birçok SLAM uygulamasında önerilen metotlar için karşılaştırma olanağı sunmuştur. Kullanılan araç bilgisi; araç dingil mesafesi 2.83m ve lazer sistemi bulucusu 180° görüş açısına sahiptir. Aracın hızı ve başlık açısı enkoder tarafından elde edilmiştir. Fakat bilgi ölçülürken tekerlerin ve aracın hareketine bağlı olarak bazı istenmeyen hatalar oluşmuştur. Bu yüzden odometri bilgisi enkoderden zayıf olarak alınmıştır.

Parkta araç GPS ile donatılmasına rağmen algılayıcı sınırlı uydu bilgisinden temiz veri sağlayamamıştır. Böylece aracın yer pozisyonu süzgecin araç durum hesabı için zayıftır. Buna rağmen önerilen süzgeçle şekil 4.24’teki sonuçlar elde edilmiştir.



Şekil 4.24 Victoria Park, Sydney, Australia, SRCD-FastSLAM hesaplanan araç rotası

Burada, araç rotası (mavi çizgi) ve nesnelere (kırmızı yıldızlar) için üç adet parçacık kullanılmıştır. Metot Victoria park tahmin sonuçları Şekil 4.25’te verilmiştir.



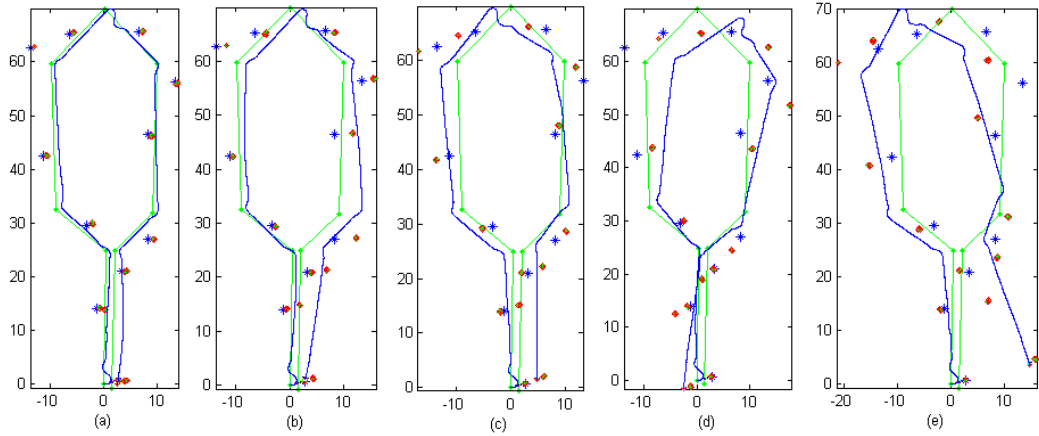
Şekil 4.25.a. FastSLAM II, b. U-FastSLAM, ve c. SRCD-FastSLAM Victoria park veri seti kestirim sonuçları

Bütün şekiller, kalın yeşil çizgiler GPS datası ve mavi çizgiler ise kestirilen araç pozisyonunu vermektedir. Bu deney için kontrol gürültüleri $\sigma_v = 0.4$ m/s, $\sigma_\gamma = 1.4^0$, ve ölçüm gürültüleri $\sigma_v = 0.5$ m/s, $\sigma_v = 1.5^0$. Bu deney için veri ilişkilendirmenin olmadığı kabul edilmiştir ve veri ilişkilendirme için 2σ büyüklüğünde kabul bölgesi oluşturulmuştur.

4.4 Uyarlı FastSLAM yaklaşımı

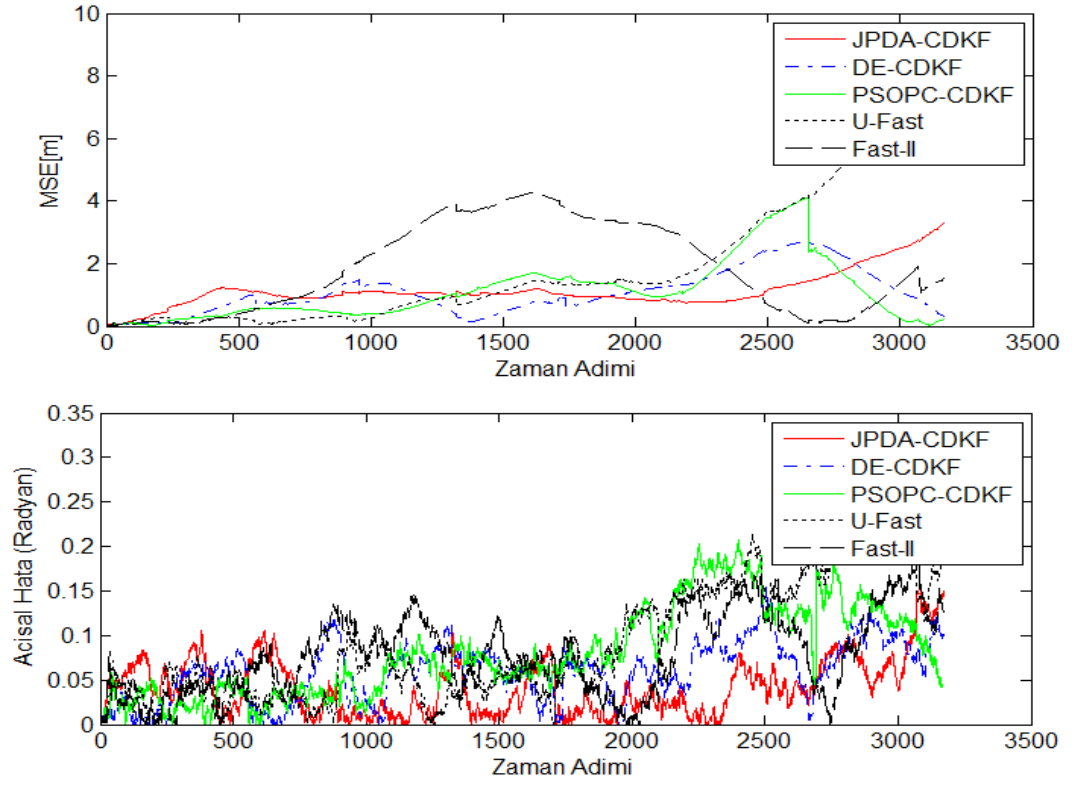
Kalman süzgeç ve türevleri, FastSLAM yapısında iyileştirmeler sağlayarak RBPF' de parçacık sayısına bağlı işlem yükünü azaltmış olsa da gerçek zamanlı uygulamalarda bir çözüm getirememişlerdir. Bu çalışmada daha öncekilerden farklı olarak uyarlı bir FastSLAM yapısı alternatif olarak sunulmuştur. Yöntem belirli iyileştirmeler içermektedir. Uyarlı yapısı sayesinde daha önceki optimize edilmiş modellere göre hızlı ve doğru sonuçlar sağlayabilmekte, bununla birlikte veri ilişkilendirme yaklaşımı JPDA' la belirsizlik durumlarında daha objektif kararlar verebilmesi amaçlanmıştır. Bu bölümde kullanılan JPDA hakkında bilgiler vermek doğru olacaktır. Çalışmanın sonuçları iki aşamada gösterilmiştir. İlk simülasyonda U-FastSLAM, (Merkez Fark Kalman Süzgeci) CDKF-FastSLAM, DE-CDKF-FastSLAM, PSO-CDKF-FastSLAM ve önerilen JPDA tabanlı uyarlı FastSLAM (JPDA-CDKF-FastSLAM) yöntemleri oluşturulan senaryo için karşılaştırılmıştır. İkinci simülasyonda önerilen yöntemin SLAM için yeterliliği tartışılmıştır. Bu bölümdeki bütün benzetim çalışmalarında veri ilişkilendirmenin bilinmediği kabul edilerek, JPDA tabanlı FastSLAM metodu hariç bütün metotlarda NN algoritması veri ilişkilendirme için kullanılmıştır.

Yeşil asteriksler oluşturulan nesnelar, mavi çizgi robot/araç rotası ve kırmızı yuvarlak düğümler robot/araç rota dönüm noktalarıdır. İç mekan uygulamasında robotun/araçın hızı 1.6m/sn, araç dingil mesafesi 0.26m, kontrol frekansı 40Hz, kontrol ve ölçüm gürültü varyansları ise eşitlik 4.1 ve 4.2' deki gibi alınmıştır. Ayrıca gözlem parametreleri maksimum laser mesafesi 5m, araç tarama frekansı 5Hz seçilmiştir. Senaryoda yöntemler için parçacık sayısı JPDA-CDKF-FastSLAM, DE-CDKF-FastSLAM, PSO-CDKF-FastSLAM için 10, U-FastSLAM ve FastSLAM II için 100 olarak seçilmiştir. Yöntemlerin deneysel sonuçları şekil 4.27'de verilmiştir. Şekil 4.28 ortalama kare hatalarını, şekil 4.29 ise 'Kutu-Plot' çizimlerini göstermektedir.

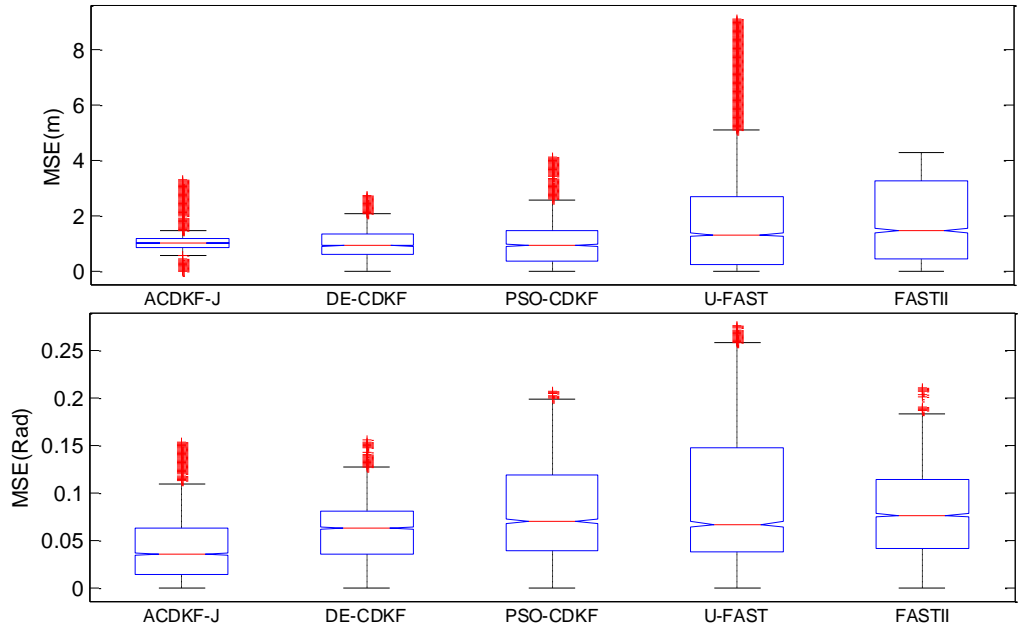


Şekil 4.27 SLAM iç mekan uygulaması:

a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-FastSLAM, d. U-FastSLAM ve e) FastSLAM-II



Şekil 4.28 SLAM iç mekan uygulaması pozisyon ve başlık ortalama kare hata sonuçları



Şekil 4.29 SLAM iç mekan uygulaması, yöntemlerin *Kutu-Plot* çizimleri

Şekiller 4.27-29 incelendiğinde, CDKF tabanlı FastSLAM yaklaşımlarının SUT yöntemini kullanan yaklaşıma göre daha az ortalama kare hatası verdiği şekil 4.28 ve 4.29’ daki sonuçlardan görülmektedir. Özellikle robotun/aracın rotayı takip ederken dönüş esnasında daha uygun tahminlerde bulunarak kestirimler yaptığı gözlenmektedir. Optimizasyon algoritması kullanan yaklaşımlar ve makalede önerilen uyarlı yapıli yaklaşımın sonuçları karşılaştırıldığında aynı sayıdaki parçacık sayılarına sahip olmalarına rağmen kestirim doğrulukları birbirine yakın çıkmıştır. Çalışmanın ana amacı da zaten optimizasyon yöntemlerine ihtiyaç duyulmadan elde edilen kestirim doğruluklarında sistemin gerçek zamanlı uygulamalar için çalışıyor olmasıdır.

Şekil 4.27 incelendiğinde FastSLAM II ve PSOPC-CDKF-FastSLAM kestirimlerinde belirli bölgelerde araç rotasında kopukluklar olduğu gözlenmektedir. Bunun sebebi, gürültüye bağıli olarak aracın sensorlerden almış olduğu ölçüm yanlışlıklarından kaynaklanmasıdır. Kopukluklardan önce, oluşturulan öznitelik tabanlı haritalarda, yanlış önsel istatistiksel bilgiler haritanın yanlış elde edilmesine yol açmıştır. Oysa belirli bir anda, sensorler yine aynı nesneyi ölçümlerden elde ettiğinde bu nesneye bağıli harita güncellenirken araç pozisyonunda da şekilde görüldüğü gibi kopukluklar oluşmaktadır. Kısaca özetlenecek olursa; yoğun gürültüye bağıli yanlış harita oluşması sonucu farklı zamanlarda alınan algılayıcı ölçümleri birbirinden farklı olduğunda, aracın tahmin edilen rotasında büyük hatalar meydana getirmekte, sonuç itibariyle rotaların kopmasına sebep olmaktadır.

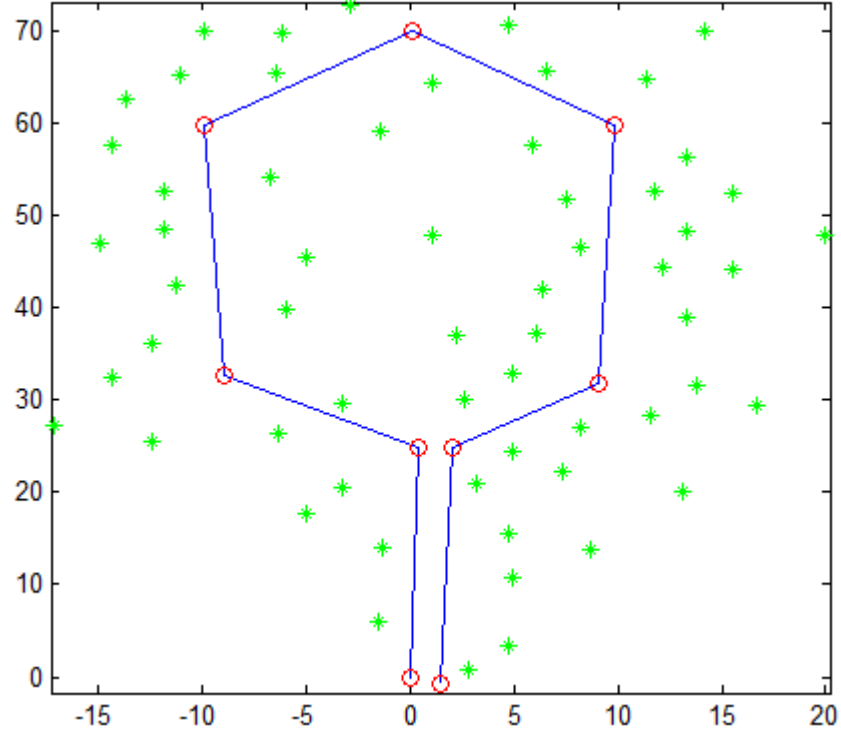
Şekil 4.28’de ortalama kare hataları hakkında genel bilgiler verilmektedir. Çizelge 4.11’de ise araçların kestirim sonuçlarının 100 MCR (Monte Carlo Run) – gerçekleştirilmiş ortalama kare hataları hakkında bilgiler verilmiştir.

Çizelge 4.11 İç mekan uygulaması yöntemlerin ortalama kare hataları (100 MCR ile elde edilmiştir)

Metotlar		JPDA-	DE-	PSOPC-	U-	FastSLAM
		CDKF-	CDKF-	CDKF-	FastSLAM	II
		FastSLAM	FastSLAM	FastSLAM		
Pozisyon	MSE	1.1012	1.0862	1.0917	1.5349	1.6968
(m)						
Pozisyon		0.3523	0.3456	0.3332	1.3213	1.5667
Varyans						
Açı (Radyan)		0.0423	0.0417	0.0439	0.0672	0.0698
Açı (Varyans)		0.0016	0.0017	0.0021	0.0029	0.0032

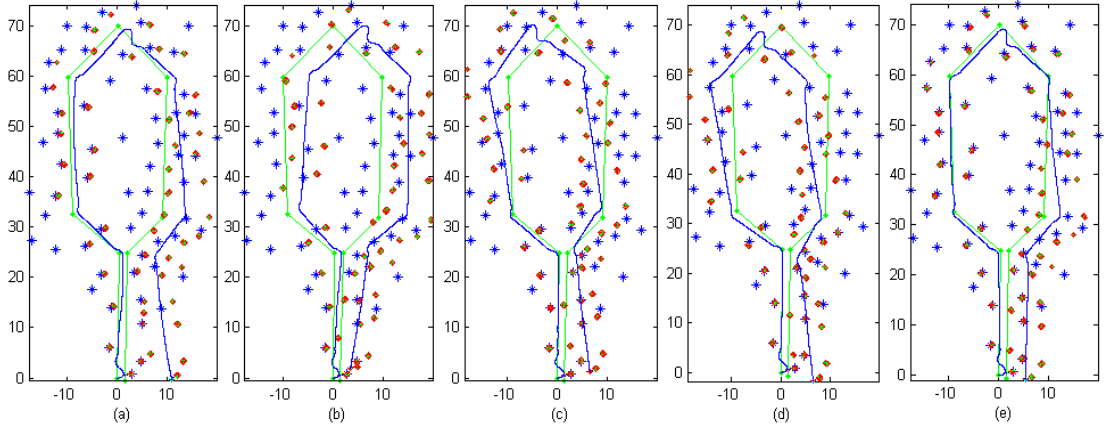
Deneysel çıktılar incelendiğinde, optimizasyon tabanlı FastSLAM yöntemlerinin sonuçlarının UKF tabanlı yaklaşımlar ve yalın CDKF tabanlı FastSLAM' den uygun çıktığını söyleyebiliriz. Çünkü aynı ortam koşulları altında optimizasyon algoritmaları süzgeçlerin kestirimleri ardından olasılığı düşük parçacıkları olasılığı yüksek bölgelere kaydırarak parçacıkların zaman içerisinde dejenerasyona uğramasını engellemişlerdir. Bu işlem parçacıklardaki çeşitlilik durumunun korunmasından kaynaklı olarak süzgeç doğruluğunu artırmıştır. Öte yandan yalın CDKF ve UKF tabanlı FastSLAM uygulamalarından iyi sonuçlar vermiş olmasına rağmen bu yöntemler gerçek zamanlı uygulamalarda içyapılarındaki iterasyon, yani SLAM için düşünülürse parçacıkların olasılık bölgesinde istenilen kadar kaydırılması işlemi, sebebi ile kullanılamaz durumdadırlar. Zaten bu sebepten ötürü aynı kestirim sonuçlarının elde edilebileceği uyarlı bir FastSLAM yapısı çalışma kapsamında önerilmiştir. Görülmektedir ki işlem zaman hesabı da göz önünde bulundurularak gayet başarılı kestirim doğruluğu sağladığı deneysel çalışmada görülmektedir.

Diğer bir uygulama ise önerilen yaklaşımın yoğun gürültülü (gürültü miktarı aynı senaryo için iki katına çıkarılmıştır) ve nesne sayılarının fazla ve birbirine daha yakın olduğu senaryo şekil 4.30’da verilmiştir.



Şekil 4.30 Yoğun gürültülü, nesne sayılarının fazla ve birbirine yakın olduğu benzetim ortamı

Şekil 4.30 için elde edilen kestiricilerin kestirim sonuçları şekil 4.31’de gösterilmektedir. Sonuçlar 100 MCR ile elde edilmiştir. Bu benzetim ortamında JPDA tabanlı FastSLAM yaklaşımı hariç diğer metotlarda NN yöntemi veri ilişkilendirmesi için kullanılmıştır.



Şekil 4.31 SLAM iç mekan uygulaması yoğun gürültülü ve nesne sayılı ortam

a. FastSLAM II, b. U-FastSLAM, c. PSOPC-CDKF-FastSLAM, d. DE-CDKF-FastSLAM ve e. JPDA-CDKF-FastSLAM

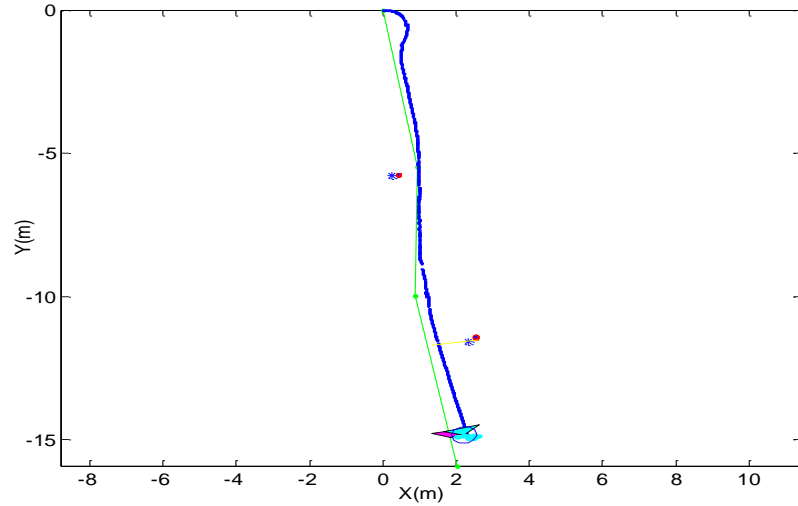
Şekil 4.31, SLAM iç mekan uygulaması için metotların performansları hakkında bilgiler içermektedir. Bir önceki benzetim ortamına kıyasla sonuçlar incelendiğinde benzer tabloyla karşılaşmaktayız. Öte yandan araçların hepsinde 1500 ve 2000’nci adımların bulunduğu bölgelerden geçerken rotada bir salınım (düz gidememe, yalpalama) olduğu gözlenmiştir. Bunun sebebinin aracın geometrik yapısından kaynaklı olduğu tespit edilmiştir. Ön ve arka tekerler arasındaki aks mesafesi ve aracın hızı bu salınıma yol açmaktadır. Hız değiştirilebilmektedir, fakat aks mesafesi değişmediğinden sonuç benzetilen araç modeli için değişmemektedir.

Metotların tutarlılığı dikkate alındığında, yoğun ve gürültülü ortamlar için aynı sonuçların sağlandığı gözlenmiştir. Önerilen metodun sonuçlarının FastSLAM II ve U-FastSLAM’den iyi çıktığı, PSOPC-CDKF-FastSLAM ve DE-CDKF-FastSLAM’e kıyasla yaklaşık sonuçların elde edildiği benzetim çalışmalarından görülmüştür.

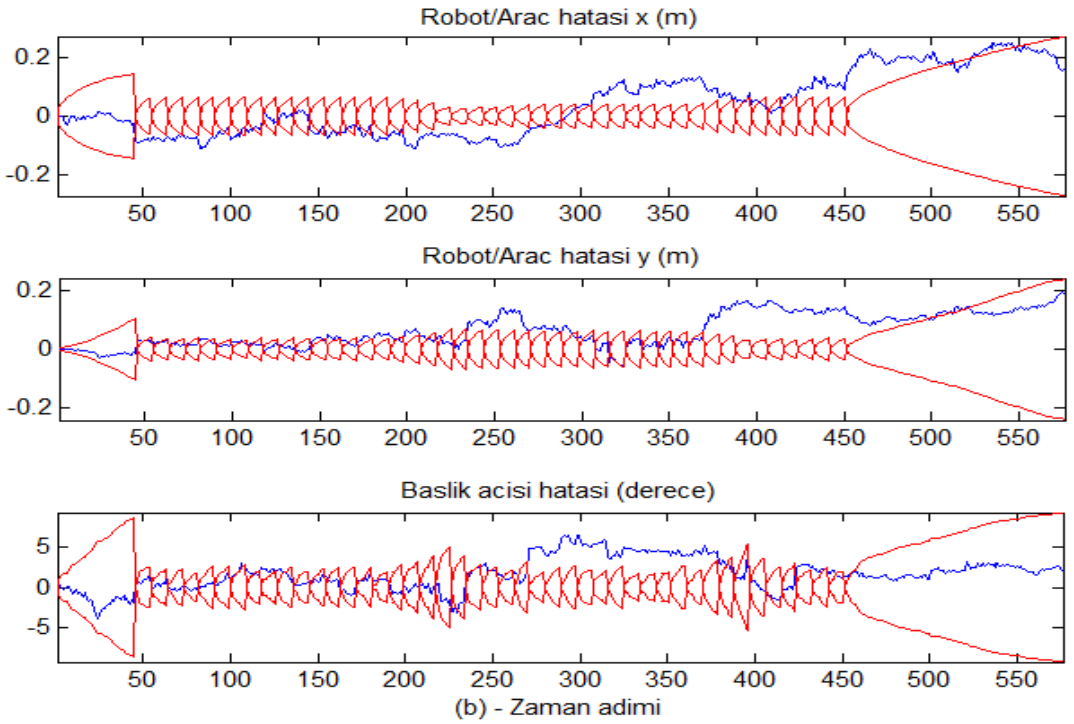
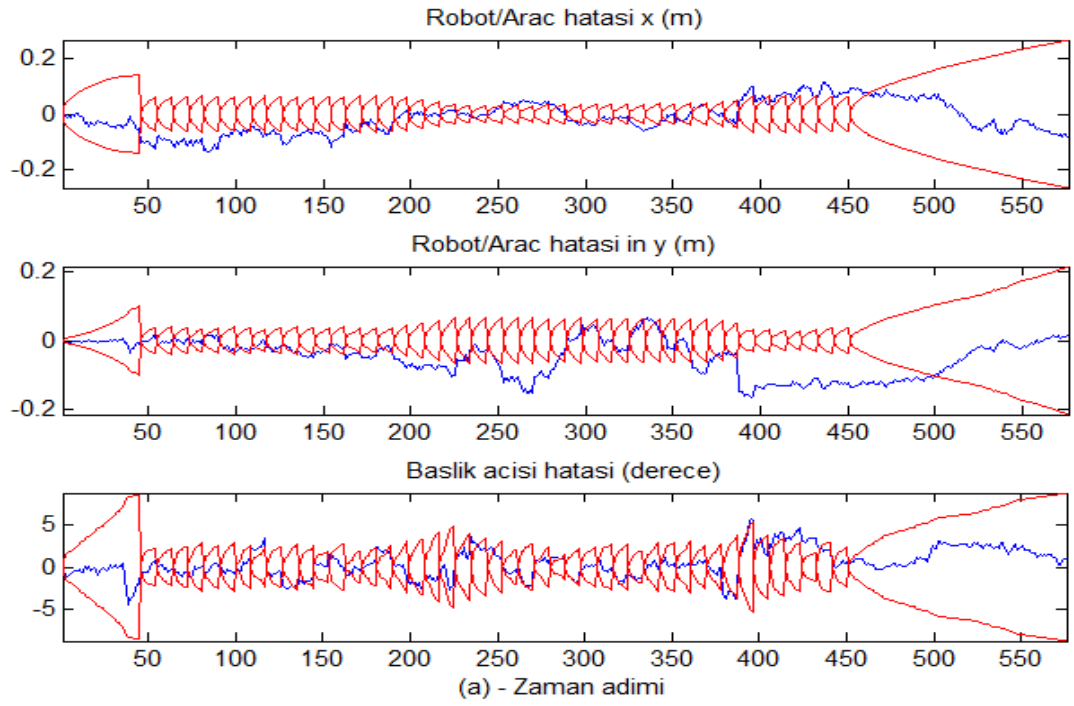
Benzetim 2 Önerilen yaklaşım yeterliliği

Çalışmanın bu bölümünde önerilen süzgeç modelinin SLAM uygulamaları için yeterliliği kıyaslanmıştır. U-FastSLAM, DE-CDKF-FastSLAM, PSOPC-CDKF-FastSLAM ve JPDA-CDKF-FastSLAM modelleri, önerilen senaryo için karşılaştırılmıştır. Benzetimde parçacık sayısı her bir süzgeç için 50 olarak belirlenmiştir. Sistem ve gözlem parametreleri yine bir önceki senaryodaki gibi alınmıştır. DE tabanlı FastSLAM için 'cross over' (Cr) olasılığı 0.8 ve mutasyon faktörü ($F=0.4$) seçilmiştir. PSO için sürü (swarm) büyüklüğü 20, maksimum jenerasyon sayısı 100, problem boyutu 3, ivmelenme sabitleri 2.0 olarak belirlenmiştir.

Senaryo için kullanılan aracın dingil mesafesi 0.26m ve hızı ise 0.64 m/s, kontrol frekansı 40Hz, algılayıcı frekansı ise 5Hz olarak alınmıştır. Kontrol gürültüleri 0.4 m/s hızda 4° başlık açısına sahip ve ölçüm gürültüsü ise 0.2 m/s aralığında 2° menzil olarak alınmıştır. Maksimum algılayıcı mesafesi ise 5m ve görüş alanı (Field of view, FOV) 180° olarak alınmıştır. Önerilen yöntem ve karşılaştırılan metotların kestirim doğrulukları şekil 4.32-35' de verilmiştir. Ortalama kare hataları ise çizelge 4.12'de gösterilmiştir.

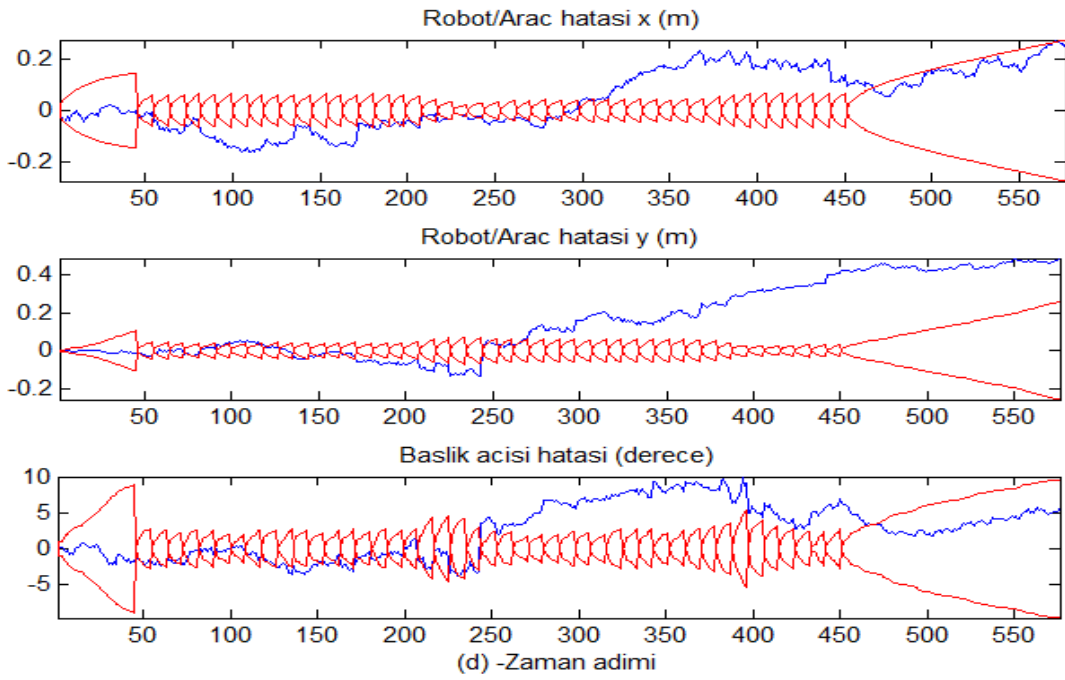
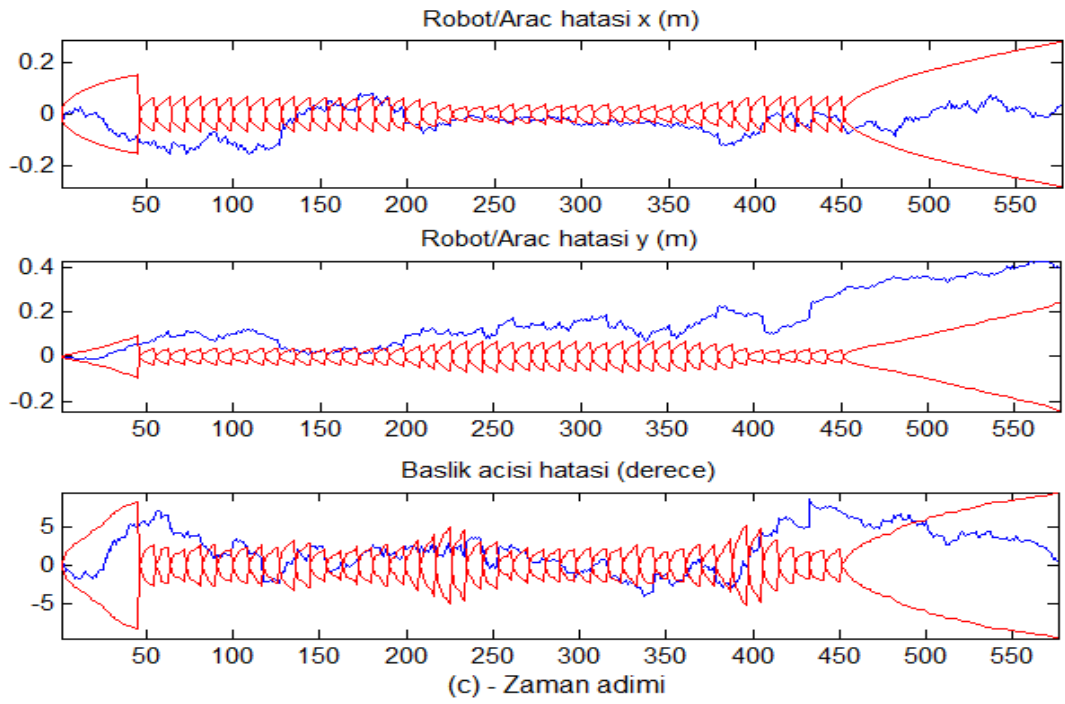


Şekil 4.32 JPDA-CDKF-FastSLAM yöntemi kestirim sonucu

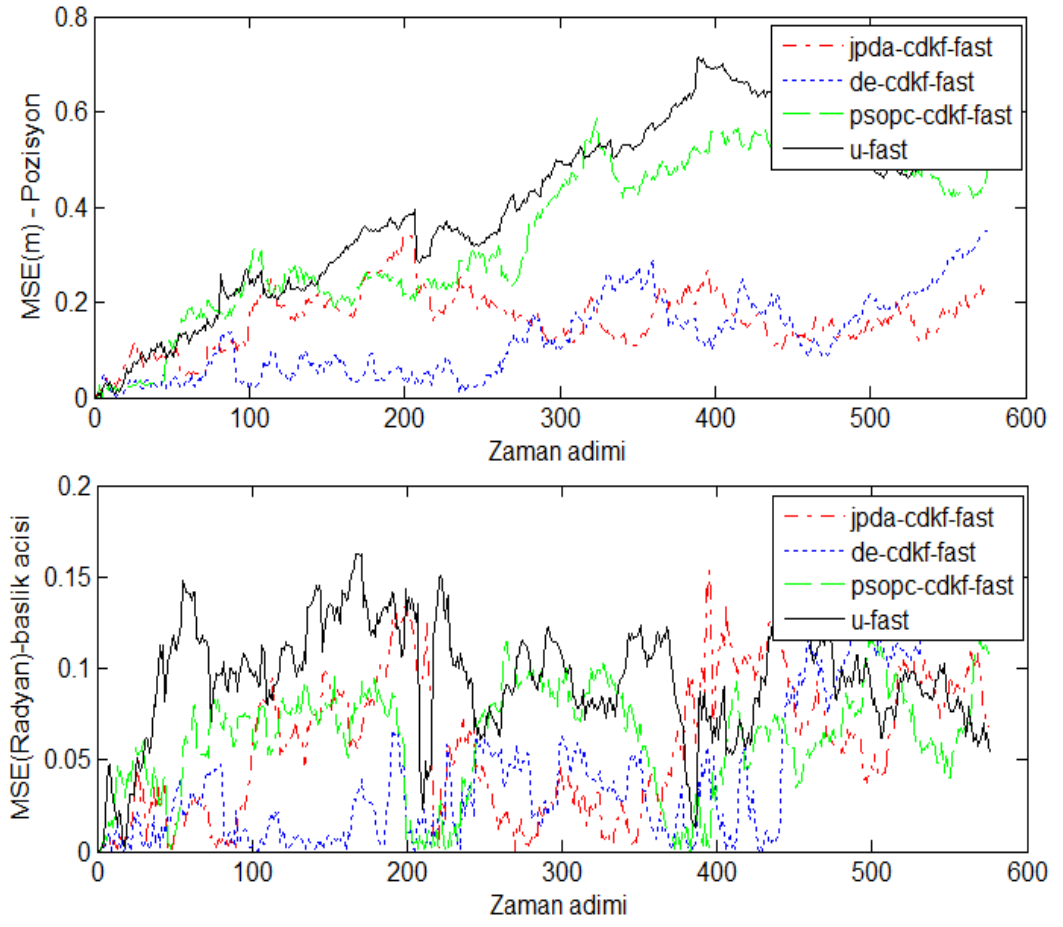


Şekil 4.33 Önerilen senaryo için metotların pozisyon ve başlık açısı hataları

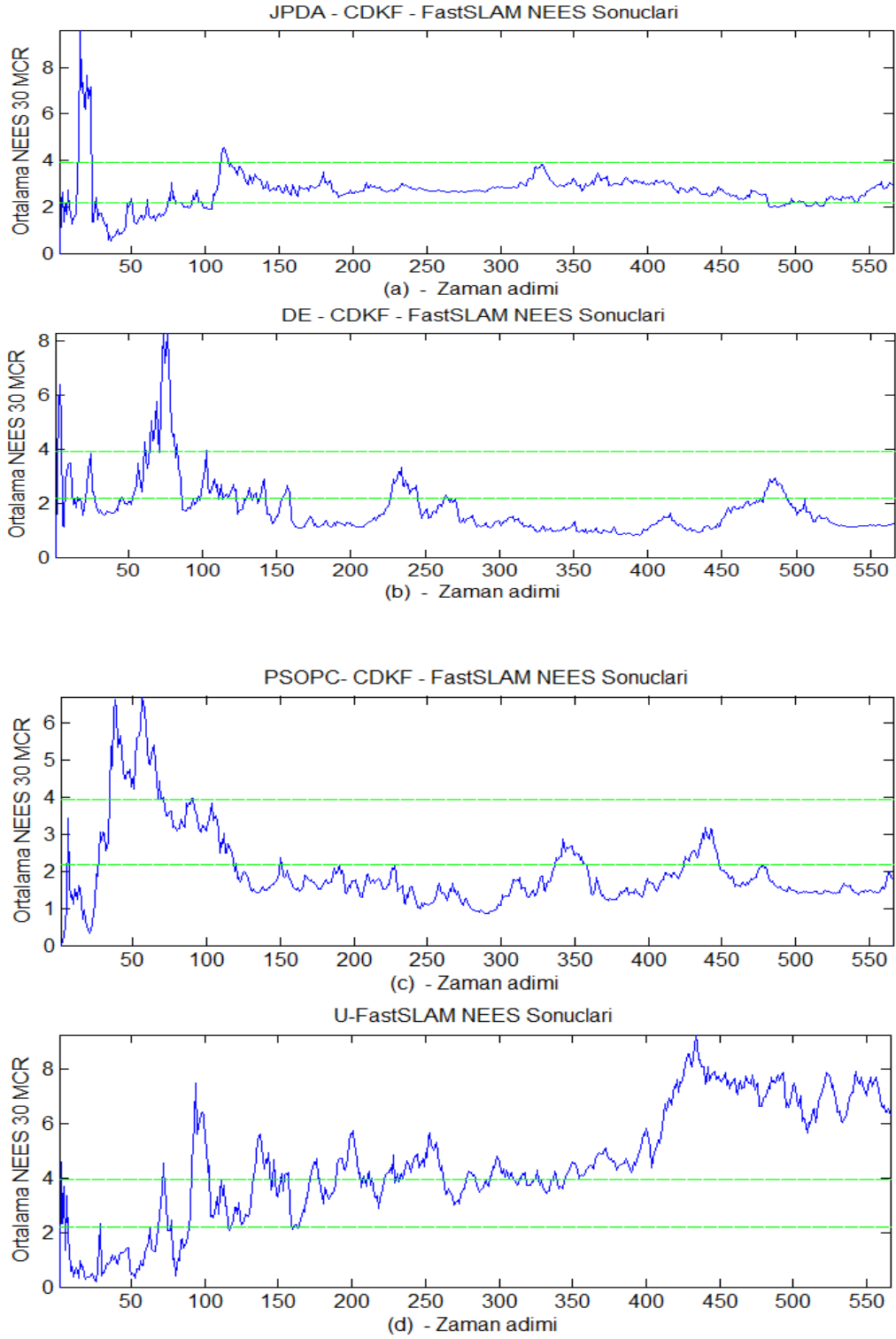
a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-FastSLAM, d. U-FastSLAM



Şekil 4.33 Önerilen senaryo için metotların pozisyon ve başlık açısı hataları (devam)
a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-FastSLAM, d. U-FastSLAM



Şekil 4.34 Metotların ilgili senaryo için ortalama pozisyon ve başlık açısı hata grafikleri



Şekil 4.35 Metotların Normalize edilmiş kestirim hata kareleri:

a. JPDA-CDKF-FastSLAM, b. DE-CDKF-FastSLAM, c. PSOPC-CDKF-FastSLAM, d. U-FastSLAM

Sonular incelendiğinde bütn kestirimler metotların sonularına baėlı olarak verilmiřtir. Őekil 4.33 ve 4.34’te en uygun sonuların nerilen yntem iin saėlandığı grlmektedir. Optimizasyon algoritmaları her ne kadar uygun aralıkta sonular vermiř olsa da aynı paracık sayısı ve ortam kořulları iin JPDA-CDKF-FastSLAM yaklařımının da bařarılı olduėu gzlenmektedir. Sonuların genel olarak gven aralıėı ierisinde olduėu grlmektedir.

5. TARTIŞMA ve SONUÇ

Eşzamanlı konum belirleme ve harita oluşturma (SLAM), son zamanlarda mobil robotlar/özerk araçlar tarafından tercih edilen bir yöntem olarak bilinmektedir. Tez çalışmasında bahsedildiği üzere, SLAM bazı problemlere sahiptir. Çalışma kapsamında bu problemlerin çözümleri üzerinde durulmuş ve iyileştirilmiş, Kalman süzgecini kullanan yöntemler önerilmiştir. Önerilen yöntemlerin, karşılaşılan problemler için teker teker veya hepsine birden çözüm oluşturabildiği deneysel çalışmalarda gözlenmiştir. Bu problemler şu başlıklarda ele alınmıştır:

- Ölçüm gürültüsü minimizasyonu için simülasyon çalışmaları üç başlık altında incelenmiştir. İlk çalışmada SR-UKF ve SR-CDKF süzgeçlerinin performansları, EKF, UKF ve FastSLAM II metotlarıyla kıyaslanmıştır. Simülasyon sonuçları incelendiğinde en uygun sonucu SR-CDKF süzgecinin sağladığı görülmüştür. Özellikle simülasyonda, SR-CDKF aracın dönüş yaptığı bölgelerde en düşük ortalama kare hatalarının elde edilmesini sağlamıştır. Bunun yanı sıra SR-UKF süzgeci geleneksel UKF' den konum hatası bakımından daha uygun çıktılar vermiştir. Aynı zamanda işlem yükü karmaşıklığı göz önüne alındığında CDKF tabanlı süzgeçlerin daha kısa işlem süresine ihtiyaç duyduğu benzetim çalışmalarından görülmüştür. İkinci çalışmada ise bulanık mantık destekli bir Kalman süzgeci yine gürültü minimizasyonu için önerilmiştir. Benzetim çalışmasında ANFIS, Q ve R matrislerini off-line olarak uyarlı bir yaklaşımla adapte etmeye çalışmıştır. Simülasyon sonuçları aynı senaryo için EKF, UKF, SR-CDKF ve FastSLAM II metotlarıyla kıyaslanmıştır. Q ve R matrislerinin ANFIS ve RTS destekli uyarlanabilir modelinin kestirim doğruluğunu artırarak gürültü minimizasyonunu sağladığı görülmüştür. Ağırlıklı olarak gürültü minimizasyonu dikkate alındığında, geliştirilen SR-UKF tabanlı FastSLAM yaklaşımı SLAM probleminin çözümü olarak ilk bölümün son çalışması için önerilmiştir. Sonuçlar SRU-FastSLAM, SRU-PF-FastSLAM, U-Aid-FastSLAM, UPF-FastSLAM ve FastSLAM II metotlarıyla kıyaslanmıştır. Benzetim çalışmaları incelendiğinde en uygun ortalama kare hatası minimizasyonunu sağlayan

yöntemin SRU-FastSLAM olduğu gözlenmiştir. Sonuçlar genel olarak, yöntemlerin robotun/özerk aracın dönüş yaptığı durumlarda başlık açılarının, gürültüden anlık olarak etkilendikleri ve buna bağlı olarak istenmeyen oranda büyüdüklerini göstermiştir. SUT'un SLAM uygulamalarında UKF için bir artı sağladığı ve böylece kestirim doğruluğunu artırdığı yine simülasyon sonuçlarından gözlenmiştir.

- Benzetim çalışmalarında ele alınan diğer bir problem ise veri ilişkilendirmedir. Tez çalışması kapsamında bu problem, parazit yankılı ortamlarda harita güncelleme işlemi için ve parazit yankısız ortamda konum belirleme ve harita oluşturma işlemleri için çözümlenmeye çalışılmıştır. Bu kapsamda iki uygulama gerçekleştirilmiştir. İlk uygulama SLAM'in gürültülü ortamda nesne işaretleriyle birlikte gerçekleştirilen iç mekân uygulaması olarak, ikinci uygulama ise yine aynı ortamda fakat rasgele dağıtılmış parazit yankılı ortam olarak ele alınmıştır. İlk çalışmada sonuçlar NN-UKF, NN-EKF ve FastSLAM II yaklaşımlarıyla kıyaslanmıştır. Sonuçlar incelendiğinde, nesne sayısının artışının nesnelerin belirsizliklerinde de artışa neden olduğu, dolayısıyla hata karelerinin de arttığını göstermiştir. Fakat hata kareleri ortalamaları ve kestirim belirsizliklerinin her metot için farklı oranda değiştiği benzetim çalışmalarından gözlenmiştir. Benzetim çalışmalarında, JPDA tabanlı UKF'nin en uygun kestirim sonuçlarını sağladığı görülmüştür. JPDA'nın belirsizliğin yüksek olduğu durumlarda daha uygun ilişkilendirme sonuçları verdiği ve kestirim doğruluğunda iyileştirmeler sağladığı görülmüştür. Metodun, her bir zaman adımı için, özellikle nesne işaretlerinin birbirine yakın olduğu durumlarda, ilişkilendirmeyi diğer metotlara nazaran daha başarılı gerçekleştirdiği görülmüştür. İkinci benzetim çalışmasında ortamda belirli oranda parazit yankı olduğu varsayılmıştır. Yine ilk benzetimde olduğu gibi bu aşamada da JPDA-UKF en uygun kestirim sonuçlarını vermiştir. Bunun sebebi olarak metotun veri ilişkilendirmesi yaparken kapılar içerisindeki bütün olası ölçümleri aday tahmin değeri olarak alması sonucu ve SUT yapısı sayesinde kestirimlerde sağladığı iyileştirmelerle bu sonuca ulaştığı gözlenmiştir. Sonuçlardan JPDA-UKF'nin en az etki altında kalarak belirsizliği minimize edebildiği görülmüştür.

- Benzetim çalışmalarının bu bölümünde optimizasyon yöntemlerine dayalı FastSLAM yaklaşımları, SLAM probleminde karşılaşılan genel problemlerin hepsinin bir model altında çözümlenmesi için önerilmiştir. SRU-FastSLAM yaklaşımı için PSOPC tabanlı yeni bir yöntem önerilmiştir. Sonuçlar FastSLAM II, U-FastSLAM ve SRU-FastSLAM yöntemleriyle kıyaslanmıştır. PSOPC'nin kullanım amacı en başta daha az parçacık sayısı ile daha optimal kestirim sonuçları elde etmektir. PSOPC bu aşamada RBPF'nin ağırlıkları düşük parçacıkları bir uygunluk fonksiyonu sayesinde olasılığı düşük bölgelerden daha yüksek bölgelere kaydirmiştir. Böylece parçacık süzgecinin çeşitliliği artırılmıştır. Simülasyon çalışmaları yapılan işlemlerin doğru bir seçim olduğunu göstermiştir. En uygun ortalama kare hatası sonuçlarının PSOPC tabanlı model için gerçekleştiğini göstermiştir. İkinci optimizasyon tabanlı yaklaşım ise benzetim çalışmasının bu bölümünde ikinci kısımda verilmiştir. Bu bölümde DE tabanlı bir SR-CDKF-FastSLAM yaklaşımı FastSLAM II, U-FastSLAM, SR-CDKF-FastSLAM, SR-CDKF-Aid-FastSLAM, SR-CDKF-PF-FastSLAM yöntemleriyle kıyaslanmıştır. Çalışma bu aşama için dört adet benzetim çalışmasıyla gerçekleştirilmiştir. İlk bölüm önerilen metod için bahsedilen süzgeçlerin kıyaslanması ve sonuçlarını içermektedir. İkinci aşamada U-FastSLAM, SR-CDKF-FastSLAM ve DE-SR-CDKF-FastSLAM metodları kıyaslanmıştır. Burada, veri ilişkilendirmenin bilinmediği farzedilmiştir. Üçüncü aşamada önerilen DE-SR-CDKF-FastSLAM'in tutarlılığı tartışılmıştır ve sonuçlar FastSLAM II ve U-FastSLAM ile kıyaslanmıştır. Son simülasyon çalışmasında ise bir dış çevre uygulaması gerçekleştirilmiştir. Victoria parkı veri seti kullanılarak sonuçlar FastSLAM II ve U-FastSLAM ile kıyaslanmıştır. Sonuçlar incelendiğinde, metodun parçacıklarda çeşitlilik oluşmasını sağlamasıyla DE tabanlı önerilen yaklaşımın en uygun sonuçları sağladığı gözlenmiştir. Aynı zamanda önerilen metodla, U-FastSLAM modelinde karşılaşılan pozitif yarı tanımlılık problemi çözümlenebilmiştir. Önerilen metodun diğer bir artışı ise daha az parçacıkla daha uygun sonuçlar elde edilmesidir. Dezavantajı ise parçacık sayısını azaltmasına rağmen gerçek zamanlı uygulamalarda optimizasyon yönteminde gerçekleşen iterasyondan kaynaklı zaman kaybı sonucunda uygulanmasının zor olmasıdır. Öte yandan,

iterasyon sayısı azaltılarak gerçek zamanlı uygulanabilmektedir, fakat bu durumda istenilen kestirim performansının elde edilememesi dezavantaj olarak görülmüştür.

- Tez çalışması kapsamında önerilen son yaklaşım ise JPDA tabanlı uyarlı CDKF-FastSLAM modelidir. Bu çalışmada diğerlerinden farklı olarak uyarlı bir FastSLAM yapısı önerilmiştir. Benzetim çalışmalarından FastSLAM'in uyarlı modeli sayesinde optimize edilmiş modellere göre hızlı ve doğru sonuçlar sağladığı gözlenmiştir. Yöntem gürültü minimizasyonu, veri ilişkilendirme, işlem yükü fazlalığı, gerçek zamanlı uygulanabilirliği ve Gauss olmayan ortamlarda da çalışabilmesi gibi temel problemlere çözüm getirmesi için önerilmiştir. Sonuçlar üç aşamada gösterilmiştir: ilk aşamada önerilen metot U-FastSLAM, CDKF-FastSLAM, DE-CDKF-FastSLAM ve PSOPC-CDKF-FastSLAM yöntemleriyle kıyaslanmıştır. İkinci aşamada önerilen yöntemin tutarlılığı tartışılmış, son aşamada ise iç mekân ve dış çevre uygulamaları gösterilmiştir. Deneysel sonuçlar önerilen yöntemin gerek iç mekân uygulamalarında gerekse dış çevre uygulamalarında gerçek zamanlı verilerde diğer metotlara kıyasla daha uygun kestirim sonuçları sağladığını göstermiştir. Daha gürbüz sonuçlar elde edilerek parazit yankısı olmayan ortamlar için SLAM'in bahsedilen problemleri göz önünde bulundurularak, önerilen yöntem ile uygun sonuçlar verdiği, belirsizlik durumlarında, istenilen düzeyde kararlar vererek kestirim doğruluğunu önceki yöntemlere nazaran daha küçük ortalama kare hatası elde ederek başardığı gözlenmiştir.

KAYNAKLAR

- Ankışhan, H. ve Efe, M. 2010. Eşzamanlı konum belirleme ve haritalamaya Kalman filtre Yaklaşımları. Dicle üniversitesi mühendislik fakültesi dergisi, 1(1); 13-20.
- Ankışhan, H. ve Efe, M. 2010. Eşzamanlı Lokalizasyon ve Haritalamaya Kalman Filtre Yaklaşımları. Bilimde Modern Yöntemler Sempozyumu 2010. Diyarbakır, Türkiye.
- Ankışhan, H., Arı, F., Tartan, E. O., and Pakfiliz, A. G. 2014. Central Difference Based FastSLAM Approach to SLAM Simulations improved by Differential Evolution. TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES, (online-Kabul).
- Ankışhan, H. 2011. Smoother Aided Neuro Fuzzy Kalman Filter Approach To Slam Problem. The International Association of Science and Technology for Development (IASTED-SIP 2011): Dallas, ABD.
- Ankışhan, H., Tartan, E. O. and Arı, F. 2013. Square Root Unscented Based FastSlam Optimized By Particle Swarm Optimization Passive Congregation. IEEE International Conference on Mechatronics and Automation. Takamatsu, Kagawa, Japan, 3-6 August.
- Ankışhan, H. ve Arı, F. 2013. Parazit Yankılı Ortamda Eşzamanlı Konum Belirleme ve Harita Oluşturma Problemi için Veri İlişkilendirme. Dicle Üniversitesi Mühendislik Fakültesi dergisi.
- Ankışhan, H. ve Efe, M. 2011. Eşzamanlı konum belirleme ve harita oluşturmada ayarlanabilir bulanık mantık destekli Kalman filtre yaklaşımı. 19. IEEE Sinyal işleme ve iletişim uygulamaları kurultayı: Kemer, Antalya.
- Ankışhan, H. ve Arı, F. 2013. SLAM Probleminin Çözümünde Kare Kök Kokusuz Süzgeç Tabanlı FastSLAM Yaklaşımı. 21. IEEE SİNYAL İŞLEME VE İLETİŞİM UYGULAMALARI KURULTAYI, Kıbrıs, 22/04/2013 - 24/04/2013.
- Bailey, T. 2002. Mobile robot localization and mapping in extensive outdoor environments. Ph.D. dissertation, Univ. Sydney, Sydney, NSW, Australia.
- Bailey T. 2003. Constrained Initialization for Bearing-Only SLAM. IEEE International Conference on Robotics and Automation.

- Chatila R. and Laumond, J.P. 1985. Position references and Consistent World Modeling for Mobile Robots. Second International Symposium of Robotics Research, Kyoto.
- Cochran, W. 1977. Sampling Techniques, Third Edition. John Wiley and Sons.
- Das, S. and Suganthan, P.N. 2011. Differential evolution: A survey of the state-of-the-art. IEEE Transaction on Evolutionary Computation, 15: 4-31.
- Davison, A. J. and Murray, D.W. 2002. Simultaneous localization and map-building using active vision. IEEE Trans. Pattern Anal. Machine Intell. (24) 7; 865–880, Jul.
- Dempster A. P., Laird N. M. and Rubin D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological), Vol. 39, No. 1, pp. 1-38.
- Doucet, A., De F., N., Murphy, K. and Russell, S. 2000. Rao–Blackwellised particle filtering for dynamic Bayesian networks. Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence. pp. 176–183. CiteSeerX: 10.1.1.137.5199.
- Duan, Y., Shibasaki, R., Ji, S., Shi, Y. and Shi, Z. 2013. GPS-Supported Visual SLAM with a Rigorous Sensor Model for a Panoramic Camera in Outdoor Environments. Sensors, 13; 119-136, doi:10.3390/s130100119 sensors ISSN 1424-8220.
- Fort, G. and Moulines, E. 2011. Online Expectation Maximization algorithm to solve the SLAM problem. Statistical Signal Processing Workshop (SSP). Le Corff, S. ; LTCI, TELECOM ParisTech-CNRS, Paris, France.
- Guivant, J., Nebot, E., and Baiker, S. 2000. Localization and map building using laser range sensors in outdoor applications. J. Robot. Syst., 17; pp: 565–583, Oct.
- He. S., Wu, Q. H., Wen, J. Y., Saunders J. R. and Paton, R. C. 2004. A particle swarm optimizer with passive congregation,” BioSyst., vol. 78, no. 1–3, pp. 135–147, Dec.

- Hele, D.K. 2002. Erasmus Darwin's Improved Design for Steering Carriages. The Royal Society, London. (Accessed April 2008).
- Hu, C, Chen, Y. and Liu, D. 2003. Adaptive Kalman Filtering for Vehicle Navigation. Department of Land Surveying and Geo-Information, Hong Kong Polytechnic University, Hong Kong.
- Ito, K. and Xiong, K. 2000. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45(5); 910–927.
- Janabi-Sharifi, F. 2013. A Robust Vision-Based Sensor Fusion Approach for Real-Time Pose Estimation. *IEEE Transaction on Cybernetics*, 44(2).
- Jang J-S.R. 1993. ANFIS: Adaptive Network Based Fuzzy Inference System. *IEEE Transactions On Systems, Man, and Cybernetics*. (23): 3.
- Julier, S., Uhlmann, J. and Whyte, H.D. 1995. A new approach for filtering nonlinear systems. *Proceedings of the 1995, American Control Conference*. IEEE Press, June 1995.
- Julier, S. and Uhlmann, J. 1997. A New Extension of The Kalman Filter to Nonlinear Systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*. Orlando.
- Kalman, R.E. 1960. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering* 82 (1): 35–45. doi:10.1115/1.3662552. Retrieved 2013-04-08.
- Kennedy, J. and Eberhart, R. 1995. Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks IV*. pp. 1942–1948. Doi: 10.1109/ICNN.1995.488968.
- Kuipers, B. J. and Byun, Y.T., 1991. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8; 47-63.
- Lee, G.H., Faundorfer, F. and Pollefeys, M. 2013. Robust Pose-Graph Loop-Closures

with Expectation-Maximization. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

Leonard, J.J., Newman, P.M., Rikoski, R.J., Neira J. and Tardós, J.D. 2001. Towards Robust Data Association and Feature Modelling for Concurrent Mapping and Localization. ISRR-01.

MacKay, D. 1998. Introduction to Monte Carlo Methods, Learning in Graphical Models, Kluwer Academic Publishers, pp:175–204.

Madow, W. 1949. On the theory of systematic sampling, II. Annals of Mathematical Statistics, 20:333–354.

Marzorati, D. 2007. Particle-based Sensor Modeling for 3D-Vision SLAM. IEEE International Conference on Robotics and Automation, Italy, April.

Matari, M. J. 1990. Integration of representation into goal-driven behavior-based robots. International Conference on Simulation of Adaptive behavior. MIT Press, 169-175.

Maybeck, P. S. 1979. Stochastic Models, Estimation, and Control. Mathematics in Science and Engineering. New York: Academic Press. pp. 423. ISBN 0124807011.

Mohammad, A. and Schwarz, K. 1999. Adaptive Kalman Filtering for INS/GPS. Journal of Geodesy,(73), pp: 193-203.

Merwe, V.D. R. and Wan, E. A. 2001. The Square-Root Unscented Kalman Filter for State and Parameter-Estimation. In International Conference on Acoustics, Speech, and Signal Processing. Salt Lake City, Utah.

Merwe, V.D.R. 2003. Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. In Workshop on Advances in Machine Learning, Montreal. <http://www.iro.umontreal.ca/~kegl/CRMWorkshop/program.html>.

Merwe, V.D.R. 2004. Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. PhD thesis, OGI School of Science & Engineering at Oregon Health & Science University. Portland, OR.

- Montemerlo, M. Thrun, S. Koller, D. and Wegbreit, B. 2002. FastSLAM: A factored solution to the simultaneous localization and mapping problem. Proceedings of The AAAI National Conference on Artificial Intelligence. pp: 593–598.
- Montemerlo, M. and Thrun, S. 2003. Simultaneous localization and mapping with unknown data association using FastSLAM. In IEEE International conference on robotics and automation (ICRA), Taipei, Taiwan.
- Newmann, P. 1999. On the Structure and Solution of the Simultaneous Localisation and Map Building Problem. PhD Thesis, University of Sydney.
- Norgaard, M., Poulsen, N., and Ravn, O. 2000. New Developments in State Estimation for Nonlinear Systems. *Automatica*, 36(11); 1627–1638.
- Paz, L.M., Pini'es, P., Tard'os, J.D. and Neira, J. 2008. Large- Scale 6-DOF SLAM With Stereo-in-Hand. *IEEE Transactions On Robotics*. (24) 5; October.
- Quine B., Uhlmann J., and Whyte, H. D. 1995. Implicit jacobians for linearised state estimation in nonlinear systems. Proceedings of the 1995, American Control Conference, IEEE Press, June 1995.
- Rubin, D. B. 1988. *Bayesian Statistics 3*. Oxford University Press.
- Rückert, E.A. 2009. Simultaneous Localisation and Mapping for Mobile Robots with Recent Sensor Technologies. M.S. Thesis, Graz University of Technology.
- Schleicher, D., Bergasa, L. M., Barea, M.O. and López, E. 2009. Real-Time Hierarchical GPS Aided Visual SLAM on Urban Environments. IEEE International Conference on Robotics and Automation, Kobe International Conference Center Kobe, Japan, May 12-17.
- Shalom, Y. B. and Fortmann, T. 1988. *Tracking and Data Association*. Academic Press.
- Shi, Y. and Eberhart, R.C. 1998. A modified particle swarm optimizer. Proceedings of IEEE International Conference on Evolutionary Computation. pp. 69–73.

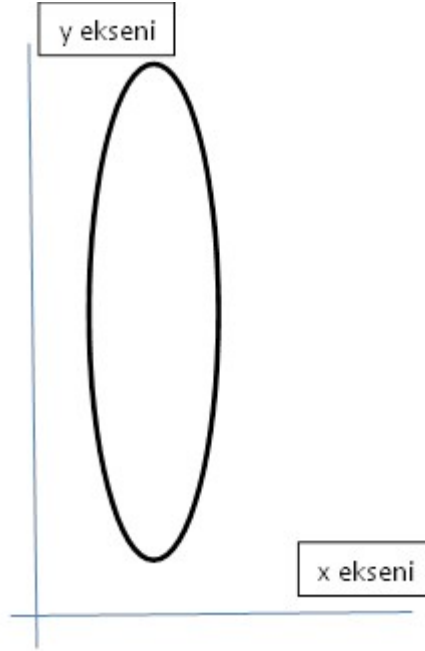
- Sukhatme, G.S. and Wolf, D. 2004. Online Simultaneous Localization and Mapping in Dynamic Environments. Proceedings of the Intl. Conf. on Robotics and Automation ICRA. New Orleans, Louisiana, Apr, 2004.
- Thrun, S., Burgard, W., and Fox, D. 1998. A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots. Machine Learning and Autonomous Robots (joint issue), 31(5); 1-25.
- Welch, G. and Bishop, G. 2001. An introduction to the Kalman filter. ACM SIGGRAPH Tutorial, http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf.
- Williams S.B, Dissanayake, G. and Whyte, D. H. 2001. Constrained Initialisation of the Simultaneous Localization and Mapping Algorithm. 3rd International Conference on Field and Service Robotics. Service Robotics (FSR 2001) June 10-13, Espoo, Finland, pp: 315- 320.
- Xia, Y.M. and Yang, Y.M. 2011. An Improved FastSLAM Algorithm Based on Genetic Algorithms, Information and Automation Communications in Computer and Information Science 86; 296-302.
- Vesterstrom, J. and Thomsen, R. 2004. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. Evolutionary Computation, 2:1980-1987.
- Zhou, W., VallsMiro, J. and Dissanayke, G. 2008. Information-Efficient 3-D Visual SLAM for Unstructured Domains. IEEE TRANSACTIONS ON ROBOTICS. (24); 1.

EKLER

EK 1 Mahalanobis Mesafesi (Mahalanobis Distance, MD).....	129
EK 2 Matris Teorisi	132
EK 3 Merkez Fark Kalman Süzgeci.....	134
EK 4 Ayarlanabilir Sinirsel Bulanık Mantık Çıkarımı Sistemi (ANFIS).....	143
EK 5 SLAM ve FastSLAM Akış Diyagramları.....	145

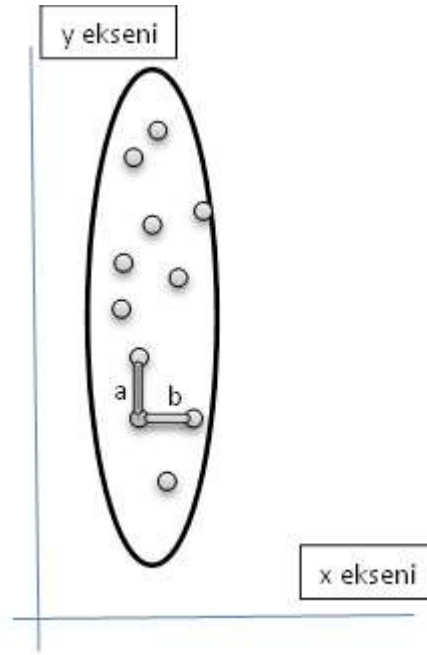
EK 1 Mahalanobis mesafesi (Mahalanobis Distance, MD)

İlk önce istatistik bilimleri olmak üzere, pek çok alanda kullanılan bir mesafe ölçüm sistemidir. Diğer ölçüm sistemlerinden en belirgin farkı, mesafe ayırımını elipsel bir düzlem üzerinde yapmaktadır (Mahalanobis 1936). Basit bir elipsi ele alırsak:



Şekil 1 Ellips Ekseni

Elipsin x eksenindeki genişliği, y eksenindeki boyuna göre daha kısadır. Bu durumda elips üzerinde dağılmış olan noktaların da aralarındaki mesafe göreceli olarak daha farklı değerlendirilmelidir.



Şekil 2 Örnek Öklit mesafe gösterimi

Örnek verilecek olursa, a ve b noktaları arasındaki uzaklık öklit mesafesine (Euclidean Distance) göre aynı olmasına rağmen Mahalanobis mesafesi farklı olmaktadır. Çünkü elipsin boyu ve eni arasındaki boyut farkı öklit mesafesindeki gibi değerlendirilmez. Buradaki en boy farkı iki stokastik sürecin kovaryansı olarak da düşünülebilir. Örneğin bir üniversitede akademisyenlerin maaşları ve yaşları iki boyutlu uzayda gösterilmek istendiğinde; yaşlar arasındaki mesafe ile maaşlar arasındaki fark aynı cinsten olmayacaktır. Maaşlar çok yüksek mesafelerle belirtilirken (500 TL gibi), yaşlar daha küçük mesafelerle (5 Yaş gibi) gösterilir.

Mahalanobis (1936) mesafesiyle bu fark çok değişkenli bir vektör için, aşağıdaki şekilde gösterilebilir:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (1)$$

Vektöre bağı olarak n adet deęiřkenden oluřan bir vektörün ortalama deęerleri (*mean*) için

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{bmatrix} \quad (2)$$

gösterimi kullanılıyor olsun. Bu deęiřkenlere bağı olarak kovaryans matrisi S ile gösterilsin. Bu durumda Mahalanobis mesafesi ařaęıdaki řekilde hesaplanır:

$$D(x, y) = \sqrt{\left(\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} - \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \vdots \\ \mu_n \end{bmatrix} \right)^T S^{-1} \left(\begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_n \end{bmatrix} - \begin{bmatrix} \mu_1 & \mu_2 & \mu_3 & \dots & \mu_n \end{bmatrix} \right)} \quad (3)$$

Buna göre Mahalanobis mesafesi, (4)'teki formül ile özetlenebilir:

$$D_M(x) = \sqrt{(x - \mu)^T S^{-1} (x - \mu)} \quad (4)$$

EK 2 Matris Teorisi

Tanımlar

- A $N \times N$ boyutlarında bir kare matrisi tanımlasın. Eğer A matrisinin eşleniğinin devriği (Transpozu) kendisine eşitse A matrisi bir hermisyen matristir.
- A hermisyen matrisi bir $N \times 1$ 'lik x vektörünün eşleniği ve x vektörünün kendisiyle çarpılırsa, $(x^*Ax > 0)$ pozitif tanımlı bir matris elde edilir. Eğer $x^*Ax \geq 0$ ifadesine eşit veya büyükse pozitif yarı tanımlı bir matris elde edilir.
- A $N \times N$ boyutlarında bir matrisi tanımlasın. Eğer $A^*A = AA^*$ ise normaldir.

Hermisyen Matrislerin Özellikleri

Hermisyen matris, karmaşık eşleniğinin devriği kendisine eşit olan matristir. Bu durum bazı kısıtlar getirir, hermisyen matrisinin kare matris olması zorunludur. Bunun yanı sıra hermisyen matrisinin köşegen elemanlarının gerçel sayı olma zorunluluğu da vardır. Bir A matrisinin hermisyen olabilmesi için bazı şartlar gereklidir. Örneğin,

$$A^* = A \quad (1)$$

olması gereklidir. Hermisyen matrislerin en önemli özellikleri arasında üniter bir değişim sayesinde köşegenleştirilebilir olmalarıdır. Köşegen elemanların gerçel olma zorunluluğundan kaynaklı olarak matrisin özdeğerlerinin de gerçel olmaları gerekmektedir.

Pozitif tanımlı matrislerin özellikleri

- İki pozitif yarı tanımlı matrisin toplamında elde edilen matriste pozitif tanımlıdır.

- Pozitif yarı tanımlı bir matrisin bütün öz değerleri sıfıra eşit veya sıfırdan büyüktür. Bununla birlikte pozitif tanımlı matrisin bütün öz değerleri sıfırdan büyük olmak zorundadır.
- Bütün pozitif yarı tanımlı matrisler bir matrisin ürünü (product) ve hermişyen eklenmiş (adjoint)'dir.
- Eğer bir A matrisi pozitif yarı tanımlı ise her hangi bir B matrisiyle oluşturduğu form BAB^* 'de pozitif yarı tanımlıdır.
- Her hangi iki pozitif yarı tanımlı A ve B matrisinin determinant eşitliğini

$$\det(A + B) \geq \det(A) \quad (2)$$

sağlar.

- Bir pozitif yarı tanımlı matrisin alt matrislerinin de köşegen değerleri pozitif yarı tanımlılık tanımına uyar.

EK 3 Merkez Fark Kalman Süzgeci

UKF süzgecinin ilk geliştirildiği zamanlarda, iki farklı grup doğrusal olmayan sistemler için EKf yerine doğrusallaştırmaya ihtiyaç duyulmayan yeni bir yöntem üzerinde çalışmışlardır. Önerdikleri yöntem Sterling's polinomsal interpolasyonu kullanmaktadır (Merwe 2004). Bu yöntem, Ito ve Xiong (2000)'un merkez fark süzgeç modelini ve Norgaard (2000)'ın bölümlenmiş fark süzgecini temel almıştır. İki süzgeç modeliyle merkez fark Kalman süzgeci (CDKF) geliştirilmiştir. Bu süzgecin detaylı derivasyonuna geçmeden evvel Sterling's polinomsal interpolasyon yaklaşımını tanımak gereklidir. İkinci dereceden Sterling's polinomsal interpolasyon CDKF'nin temelini oluşturduğu için anlatıma bu yaklaşımla devam edilecektir.

İkinci dereceden Sterling's polinomsal interpolasyon

Her hangi bir nokta çevresinde, x rasgele değişkeninin doğrusal olmayan fonksiyonunun Taylor serisi açılımı,

$$\begin{aligned} g(x) &= g(\bar{x}) + D_{\delta_x} g + \frac{1}{2!} D_{\delta_x}^2 g + \dots \\ &= g(\bar{x}) + (x - \bar{x}) \frac{dg(\bar{x})}{dx} + \frac{1}{2!} (x - \bar{x})^2 \frac{d^2g(\bar{x})}{dx^2} \end{aligned} \quad (1)$$

formunda gösterilir. Burada x , herhangi bir nokta, \bar{x} ise onun ortalamasıdır. Eşitlikte açılım ikinci dereceden terim ile sınırlandırılmıştır. Taylor serisi açılımına alternatif olabilecek diğer bir yaklaşım ise interpolasyon formu kullanılarak belirli bir aralık üzerinden doğrusal olmayan bir fonksiyona yakınsatılma işlemi olarak bahsedilebilir. Örnek olarak Sterling's polinomsal interpolasyon (SPI) yöntemi verilebilir. Bu yöntem merkez bölümlenmiş farkı kullanır. İkinci dereceden SPI,

$$g(x) = g(\bar{x}) + \bar{D}_{\Delta_x} g + \frac{1}{2!} \bar{D}_{\Delta_x}^2 g \quad (2)$$

ifadesiyle gösterilir (Merwe 2004). Burada $\bar{D}_{\Delta_x} g$ ve $\bar{D}_{\Delta_x}^2 g$, $g(x)$ 'in üzerinde etki eden birinci ve ikinci merkez bölünlenmiş fark operatörleridir. Skaler olarak tanımlanacak olursa,

$$\bar{D}_{\Delta_x} g = (x - \bar{x}) \frac{g(\bar{x} + h) - g(\bar{x} - h)}{2h} \quad (3)$$

$$\bar{D}_{\Delta_x}^2 g = (x - \bar{x})^2 \frac{g(\bar{x} + h) - g(\bar{x} - h) - 2g(\bar{x})}{h^2} \quad (4)$$

Burada h aralık uzunluğu veya merkez fark adım büyüklüğüdür. \bar{x} ise dağılacığı bölgenin etrafındaki x 'in öncül ortalaması olarak bilinir. Burada derivasyondan kurtulmak için Taylor serisi açılımı yerine merkez bölünlenmiş fark metodu kullanılmıştır. Çok boyutlu durumlar için eşitlikler genişletilirse,

$$z = S_x^{-1} x \quad (5)$$

$$\tilde{g}(z) = g(S_x z) = g(x) \quad (6)$$

doğrusal dönüşüm yardımıyla (5-6) ifadeleri elde edilecektir. Buradaki S_x , x 'in kovaryans matrisinin Cholesky faktörüdür,

$$P_x = S_x S_x^T \quad (7)$$

Bir x rasgele değişkeninin ortalaması $\bar{x} = E[x]$ ve kovaryansı $P_x = E[(x - \bar{x})(x - \bar{x})^T]$ olarak kabul edilsin. Bahsedilen dönüşümle stokastiksel olarak x 'deki değişkenlerin ilişkisi yok edilebilir, z 'in bireysel komponentleri içsel olarak böylece ilişkisiz hale gelecektir,

$$P_z = E[(z - \bar{z})(z - \bar{z})^T] = I \quad (8)$$

Eşitliklerden anlaşılacağı üzere, $\tilde{g}(z)$ 'nin elemanlarından bağımsız olarak skaler merkez farksal operasyonun uygulanmasıyla çok boyutlu merkez fark operatörü,

$$\tilde{D}_{\Delta_z} \tilde{g} = \left(\sum_{i=1}^L \Delta_{z_i} m_i d_i \right) \tilde{g}(\bar{z}) \quad (9)$$

$$\tilde{D}_{\Delta_z}^2 \tilde{g} = \left(\sum_{i=1}^L \Delta_{z_i}^2 d_i^2 + \sum_{j=1}^L \sum_{\substack{q=1 \\ q \neq j}}^L \Delta_{z_j} \Delta_{z_q} (m_j d_j)(m_q d_q) \right) \tilde{g}(\bar{z}) \quad (10)$$

eşitlikleriyle gösterilebilir (Merwe 2004). Burada, $\Delta_{z_i} = (z - \bar{z})_i$, $z - \bar{z}$ ($i=1, \dots, L$)'in i 'nci komponentidir. L , x 'in boyutudur. d_i, m_i ve d_i^2 sırasıyla, parçalı birinci dereceden fark, ortalama operatörü ve parçalı ikinci dereceden fark operatörleri olarak gösterilir.

$$d_i \tilde{g}(\bar{z}) = \frac{1}{2h} [\tilde{g}(\bar{z} + h e_i) - \tilde{g}(\bar{z} - h e_i)] \quad (11)$$

$$m_i \tilde{g}(\bar{z}) = \frac{1}{2} [\tilde{g}(\bar{z} + h e_i) + \tilde{g}(\bar{z} - h e_i)] \quad (12)$$

$$d_i^2 \tilde{g}(\bar{z}) = \frac{1}{2h^2} [\tilde{g}(\bar{z} + h e_i) + \tilde{g}(\bar{z} - h e_i) - 2\tilde{g}(\bar{z})] \quad (13)$$

Burada e_i , i 'nci birim vektördür. (5-6)'den yararlanılarak,

$$\begin{aligned} \tilde{g}(\bar{z} \pm h e_i) &= g(S_x[\bar{z} \pm h e_i]) \\ &= g(S_x \bar{z} \pm h S_x e_i) \\ &= g(\bar{x} \pm h s_{x_i}) \end{aligned} \quad (14)$$

eşitliği elde edilir. Burada s_{x_i} , x 'in kovaryans matrisinin kare Cholesky faktörünün i 'nci sütunudur.

$$s_{x_i} = S_x e_i = (S_x)_i = (\sqrt{P_x})_i \quad (15)$$

Soncul ortalama ve kovaryansın Sterling's Polinomsal İnterpolasyon ile hesaplanması

Daha önceden L boyutlu bir x rasgele değişkeninin $g(\cdot)$ doğrusal olmayan herhangi bir fonksiyona yayıldığından bahsedilmişti. Ortalamasının \bar{x} ve kovaryansının P_x olduğu kabul edilen eşitlik (16)'da verilmiştir.

$$y = g(x) \quad (16)$$

İkinci dereceden çok boyutlu SPI formunda (15) tekrar yazılırsa,

$$\begin{aligned} y &= g(x) \\ &= \tilde{g}(z) \\ &\approx \tilde{g}(\bar{z}) + \tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \end{aligned} \quad (17)$$

Burada, $\bar{z} = S_x \bar{x}$ ' dir. y 'nin soncul ortalaması, kovaryans ve çapraz kovaryansı,

$$\bar{y} = E[y] \quad (18)$$

$$P_y = E[(y - \bar{y})(y - \bar{y})^T] \quad (19)$$

$$P_{xy} = E[(x - \bar{x})(y - \bar{y})^T] \quad (20)$$

formunda gösterilir. İkinci dereceden Sterling's interpolasyon formu, y 'nin yerine eşitliklerde tekrar yazılırsa ortalama;

$$\bar{y} \approx E \left[\tilde{g}(\bar{z}) + \tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right] \quad (21)$$

$$= \tilde{g}(\bar{z}) + E \left[\frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right] \quad (22)$$

$$= \tilde{g}(\bar{z}) + E \left[\frac{1}{2h^2} \left(\sum_{i=1}^L \Delta_{z_i}^2 d_i^2 \right) \tilde{g}(\bar{z}) \right] \quad (23)$$

$$= \tilde{g}(\bar{z}) + \frac{1}{2h^2} \sum_{i=1}^L [\tilde{g}(\bar{z} + he_i) + \tilde{g}(\bar{z} - he_i) - 2\tilde{g}(\bar{z})] \quad (24)$$

$$= \frac{h^2 - L}{h^2} \tilde{g}(\bar{z}) + \frac{1}{2h^2} \sum_{i=1}^L [\tilde{g}(\bar{z} + he_i) + \tilde{g}(\bar{z} - he_i)] \quad (25)$$

olacaktır. Burada, $\Delta_z = z - \bar{z}$ 'dir. (14) eşitliği (25)'e yerleştirilirse, x 'in öncül istatistiğinden soncul ortalama

$$\bar{y} \approx \frac{h^2 - L}{h^2} g(\bar{x}) + \frac{1}{2h^2} \sum_{i=1}^L [g(\bar{x} + hs_{x_i}) + g(\bar{x} - hs_{x_i})] \quad (26)$$

olarak hesaplanır. Kovaryans eşitliği tekrar yazılarak (19) eşitliği genişletilir.

$$P_y = E[(y - \bar{y})(y - \bar{y})^T] \quad (27)$$

$$= E[(y - g(\bar{x}))(y - g(\bar{x}))^T] - E[y - g(\bar{x})]E[y - g(\bar{x})]^T \quad (28)$$

$$= E[(y - \tilde{g}(\bar{z}))(y - \tilde{g}(\bar{z}))^T] - E[y - \tilde{g}(\bar{z})]E[y - \tilde{g}(\bar{z})]^T \quad (29)$$

Bununla birlikte soncul ortalama,

$$\begin{aligned} \bar{y} &= E[y] \\ &= E[y] + g(\bar{x}) - g(\bar{x}) \\ &= E[y] + g(\bar{x}) - E[g(\bar{x})] \\ &= g(\bar{x}) + E[y - g(\bar{x})] \end{aligned} \quad (30)$$

ifadesi olarak yazılır. (17)'den

$$y - \tilde{g}(\bar{z}) = \tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \quad (31)$$

$y - \tilde{g}(\bar{z})$ 'nin ikinci dereceden Stirling's interpolasyon yaklaşımı (31)'deki gibi gösterilir. Eldeki sonuç (29)'a yerleştirilirse,

$$\begin{aligned} P_y \approx E \left[\left(\tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right) \left(\tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right)^T \right] \\ - E \left[\tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right] E \left[\tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right]^T \end{aligned} \quad (32)$$

denklemini elde edilir. Beklenen değer ve genişletme operasyonlarına bağlı olarak (32) eşitliğinde kovaryans tanımı,

$$\begin{aligned} P_y \approx \frac{1}{4h^2} \sum_{i=1}^L [g(\bar{x} + hs_{x_i}) - g(\bar{x} - hs_{x_i})] [g(\bar{x} + hs_{x_i}) - g(\bar{x} - hs_{x_i})]^T \\ + \frac{h^2 - 1}{4h^4} \sum_{i=1}^L [g(\bar{x} + hs_{x_i}) + g(\bar{x} - hs_{x_i}) \\ - 2g(\bar{x})] [g(\bar{x} + hs_{x_i}) + g(\bar{x} - hs_{x_i}) - 2g(\bar{x})]^T \end{aligned} \quad (33)$$

Çapraz kovaryans,

$$P_{xy} = E[(x - \bar{x})(y - \bar{y})^T] \quad (34)$$

$$\approx E \left[(S_x(z - \bar{z})) \left(\tilde{D}_{\Delta_z} \tilde{g} + \frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} - E \left[\frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right] \right)^T \right] \quad (35)$$

$$(36)$$

$$= E \left[(S_x \Delta_z) (\tilde{D}_{\Delta_z} \tilde{g})^T \right] + \frac{1}{2} E \left[(S_x \Delta_z) (\tilde{D}_{\Delta_z}^2 \tilde{g})^T \right] - \frac{1}{2} E[(S_x \Delta_z)] E \left[\frac{1}{2} \tilde{D}_{\Delta_z}^2 \tilde{g} \right]^2 \quad (37)$$

$$= E \left[(S_x \Delta_z) (\tilde{D}_{\Delta_z} \tilde{g})^T \right] \quad (38)$$

$$= \frac{1}{2h} \sum_{i=1}^L s_{x_i} [\tilde{g}(\bar{z} + h e_i) - \tilde{g}(\bar{z} - h e_i)]^T \quad (39)$$

$$= \frac{1}{2h} \sum_{i=1}^L s_{x_i} [g(\bar{x} + h s_{x_i}) - g(\bar{x} - h s_{x_i})]^T$$

olarak elde edilir. Eşitliklerden elde edilen Merkez fark Kalman süzgeci eşitlikleri aşağıda özetlenmiştir.

Başlangıç :

$$\begin{aligned} \hat{x}_0 &= E[x_0] & P_{x_0} &= E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \\ \bar{v} &= E[v] & R_v &= E[(v - \bar{v})(v - \bar{v})^T] \\ \bar{n} &= E[n] & R_n &= E[(n - \bar{n})(n - \bar{n})^T] \end{aligned} \quad (40)$$

for $k = 1, \dots, \infty$:

1. Zaman güncelleme için kovaryans karekök sütun vektörünün hesabı :

$$s_{k-1}^{x,i} = h(\sqrt{P_{x_{k-1}}})_i, \quad i = 1, \dots, L_x \quad (41)$$

$$s_{k-1}^{v,i} = h(\sqrt{R_v})_i, \quad i = 1, \dots, L_v \quad (42)$$

2. Zaman güncelleme eşitlikleri :

$$\begin{aligned} \hat{x}_k^- &= \frac{h^2 - L_x - L_v}{h^2} f(\hat{x}_{k-1}, \bar{v}, u_{k-1}) + \\ & \frac{1}{2h^2} \sum_{i=1}^{L_x} [f(\hat{x}_{k-1} + s_{k-1}^{x,i}, \bar{v}, u_{k-1}) + f(\hat{x}_{k-1} - s_{k-1}^{x,i}, \bar{v}, u_{k-1})] \\ & + \frac{1}{2h^2} \sum_{i=1}^{L_v} [f(\hat{x}_{k-1}, \bar{v} + s_{k-1}^{v,i}, u_{k-1}) + f(\hat{x}_{k-1}, \bar{v} - s_{k-1}^{v,i}, u_{k-1})] \end{aligned}$$

(43)

$$\begin{aligned}
P_{x_k}^- &= \frac{1}{4h^2} \sum_{i=1}^{L_x} [f(\hat{x}_{k-1} + s_{k-1}^{x,i}, \bar{v}, u_{k-1}) - f(\hat{x}_{k-1} + s_{k-1}^{x,i}, \bar{v}, u_{k-1})]^2 \\
&\quad + \frac{1}{4h^2} \sum_{i=1}^{L_v} [f(\hat{x}_{k-1}, \bar{v} + s_{k-1}^{v,i}, u_{k-1}) \\
&\quad - f(\hat{x}_{k-1}, \bar{v} - s_{k-1}^{v,i}, u_{k-1})]^2 \\
&\quad + \frac{h^2 - 1}{4h^4} \sum_{i=1}^{L_x} [f(\hat{x}_{k-1}, \bar{v} + s_{k-1}^{x,i}, u_{k-1}) \\
&\quad - f(\hat{x}_{k-1}, \bar{v} - s_{k-1}^{x,i}, u_{k-1}) - 2f(\hat{x}_{k-1}, \bar{v}, u_{k-1})]^2 \\
&\quad + \frac{h^2 - 1}{4h^4} \sum_{i=1}^{L_v} [f(\hat{x}_{k-1}, \bar{v} + s_{k-1}^{v,i}, u_{k-1}) \\
&\quad - f(\hat{x}_{k-1}, \bar{v} - s_{k-1}^{v,i}, u_{k-1}) - 2f(\hat{x}_{k-1}, \bar{v}, u_{k-1})]^2
\end{aligned} \tag{44}$$

3. Ölçüm düzeltme için kovaryans karekök sütun vektörünün hesabı :

$$s_k^{x,i} = h(\sqrt{P_{x_k}^-})_i, \quad i = 1, \dots, L_x \tag{45}$$

$$s_k^{n,i} = h(\sqrt{R_n})_i, \quad i = 1, \dots, L_n \tag{46}$$

4. Ölçüm düzeltme eşitlikleri :

$$\begin{aligned}
\hat{y}_k^- &= \frac{h^2 - L_x - L_n}{h^2} h(\hat{x}_k^-, \bar{n}) \\
&\quad + \frac{1}{2h^2} \sum_{i=1}^{L_x} [h(\hat{x}_k^- + s_k^{x,i}, \bar{n}) + h(\hat{x}_k^- - s_k^{x,i}, \bar{n})] \\
&\quad + \frac{1}{2h^2} \sum_{i=1}^{L_n} [h(\hat{x}_k^-, \bar{n} + s_k^{n,i}) + h(\hat{x}_k^-, \bar{n} - s_k^{n,i})]
\end{aligned} \tag{47}$$

$$\begin{aligned}
P_{\bar{y}_k} &= \frac{1}{4h^2} \sum_{i=1}^{L_x} [h(\hat{x}_k^- + s_k^{x,i}, \bar{n}) - h(\hat{x}_k^- - s_k^{x,i}, \bar{n})]^2 \\
&\quad + \frac{1}{4h^2} \sum_{i=1}^{L_n} [h(\hat{x}_k^-, \bar{n} + s_k^{n,i}) - h(\hat{x}_k^-, \bar{n} - s_k^{n,i})]^2 \\
&\quad + \frac{h^2 - 1}{4h^4} \sum_{i=1}^{L_x} [h(\hat{x}_k^- + s_k^{x,i}, \bar{n}) + h(\hat{x}_k^- - s_k^{x,i}, \bar{n}) \\
&\quad - 2h(\hat{x}_k^-, \bar{n})]^2 \\
&\quad + \frac{h^2 - 1}{4h^4} \sum_{i=1}^{L_n} [h(\hat{x}_k^-, \bar{n} + s_k^{n,i}) + h(\hat{x}_k^-, \bar{n} - s_k^{n,i}) \\
&\quad - 2h(\hat{x}_k^-, \bar{n})]^2
\end{aligned} \tag{48}$$

$$P_{x_k y_k} = \frac{1}{2h^2} \sum_{i=1}^{L_x} s_k^{x,i} [h(\hat{x}_k^- + s_k^{x,i}, \bar{n}) - h(\hat{x}_k^- - s_k^{x,i}, \bar{n})]^T \tag{49}$$

$$K_k = P_{x_k y_k} P_{\bar{y}_k}^{-1} \tag{50}$$

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - \hat{y}_k^-) \tag{51}$$

$$P_{x_k} = P_{x_k}^- - K_k P_{\bar{y}_k} K_k^T \tag{52}$$

Parametreler: Burada h , skaler merkez fark aralık büyüklüğünü gösterir. Gauss dağılımların $h = \sqrt{3}$ alınır. L_x, L_v ve L_n sırasıyla; durum boyutunu, işlem gürültü boyutunu ve gözlem gürültü boyutunu vermektedir. R_v ve R_n ise işlem ve gözlem kovaryanslarını temsil etmektedirler.

EK 4 Ayarlanabilir Sinirsel Bulanık Mantık Çıkarımı (ANFIS)

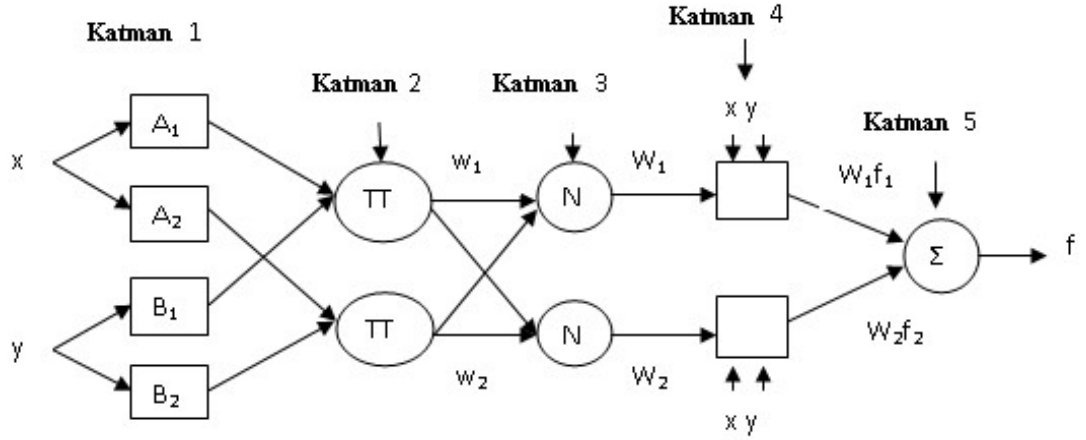
Ayarlanabilir sinirsel bulanık mantık çıkarımı sistemi (ANFIS) Jang tarafından 1993 yılında bulunmuştur. Model “eğer-o zaman” bulanık mantığına bağlı olarak çalışmakta, giriş ile çıkış arasında bir ilişkilendirme kurarak en uygun bir üyelik fonksiyonu yaratmaktadır. ANFIS hibrid bir öğrenme modeli kullanarak bulanık çıkarım sisteminin Sugeno tipinin parametrelerini belirlemeye çalışır (Jang 1993). Bunu bulanık çıkarım sistem üyelik fonksiyon parametrelerini eğitmek için en küçük kareler metodunun ve geri yayılım azalan eğitim metodunun bileşkesinden verilen veri setinin eğitimi için uygular.

ANFIS bulanık mantık modelini oluşturmak için beş katmanda uygulanır. Her bir katman düğüm fonksiyonu tarafından tanımlanan birkaç düğümden oluşur. Basit çalışma mantığı; bir önceki katmanda düğümlerden çıkış sinyali ilgili katman tarafından giriş sinyali olarak kabul edilir. İki bulanık mantık modeli mevcuttur; bu çalışmada Sugeno modeli kullanılmıştır,

Kural1:(x eğer A_1) ve (y eğer B_1) o zaman ($f_1 = p_1x + q_1y + r_1$)

Kural2: (x eğer A_2) ve (y eğer B_2) o zaman ($f_2 = p_2x + q_2y + r_2$)

Burada $x_{1,2}$ giriş parametreleri, A_i ve B_i bulanık kümeleri, f_i kurallar tarafından belirlenen bulanık bölge ile birlikte çıkışı, p_i , q_i , ve r_i ise eğitim işlemi esnasında tasarlanan parametreleri göstermektedir. ANFIS mimarisi şekil 1’ de verilmiştir.

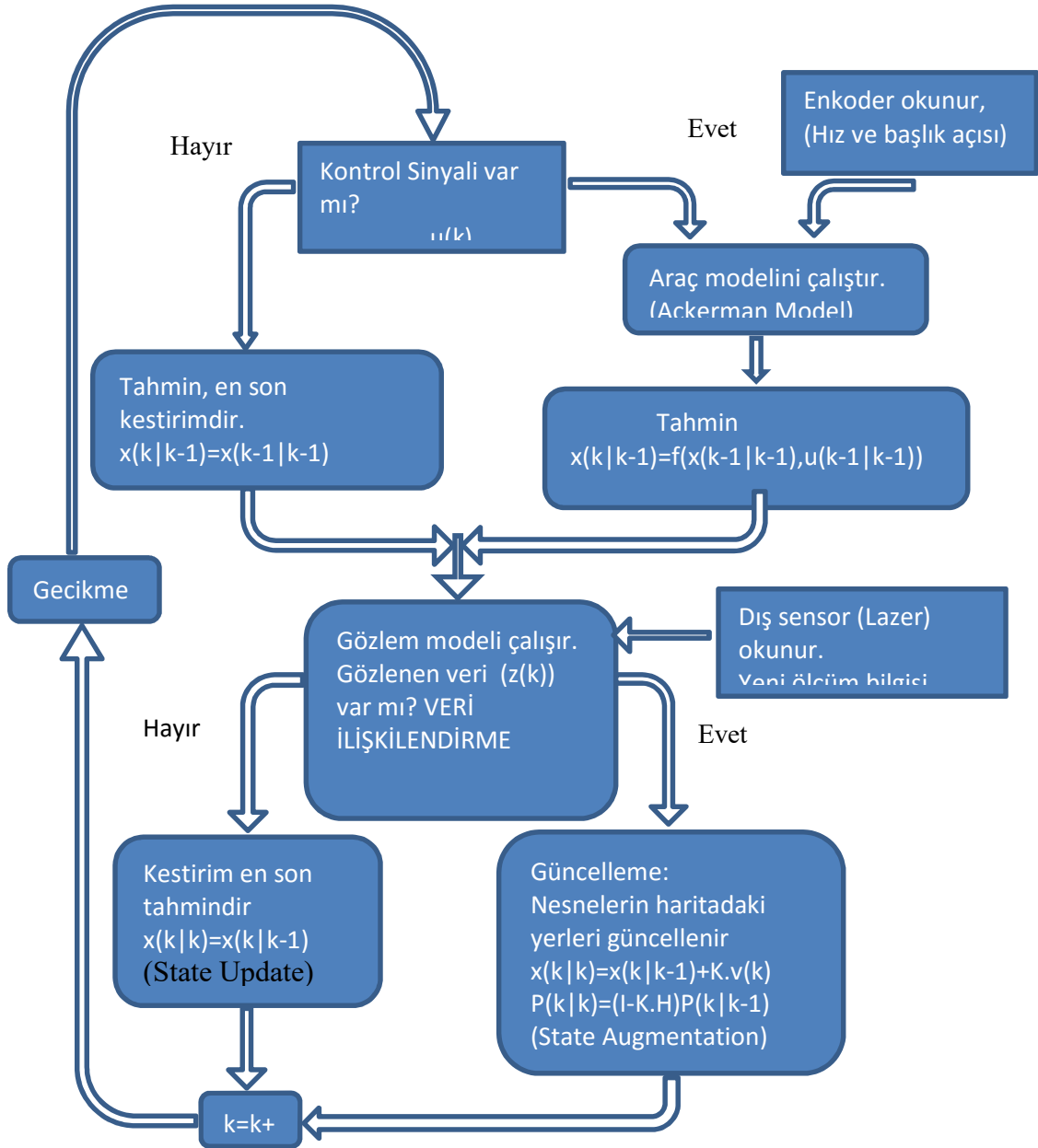


Şekil 1. Sugeno tipi ayarlanabilir sinirsel bulanık mantık çıkarımı sistemi (ANFIS) (Jang 1993).

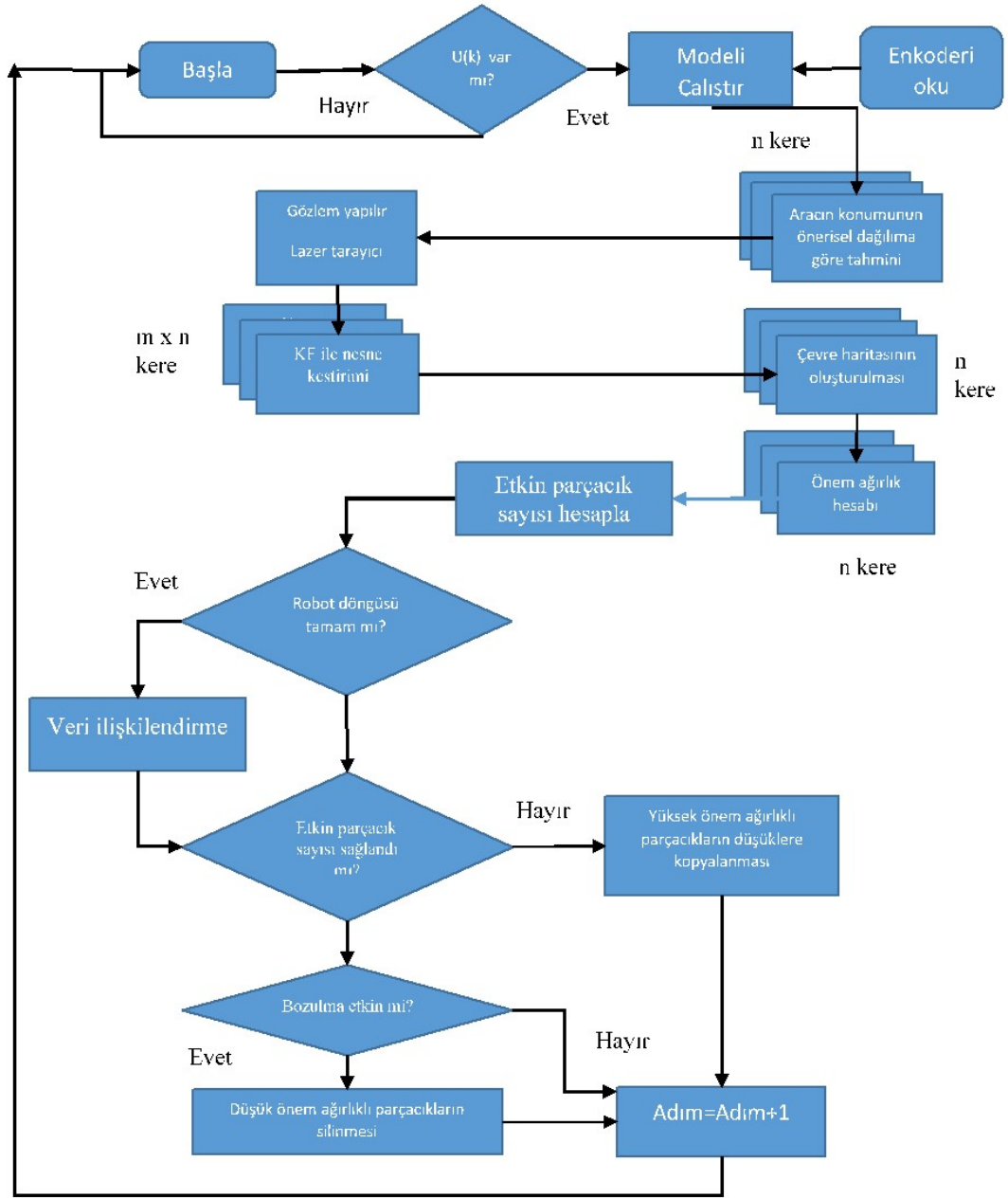
Burada kare düğümler ayarlanabilir düğümleri, çember düğümler ise sabit ayarlanamaz düğümleri göstermektedir.

Sistem SLAM problemi için kullanıldığında giriş parametreleri olarak filtre konum ve açısal başlık hatası verilmektedir. Çıkış olarak ise sistemden Q ve R gürültü matris değerleri istenmektedir. Konum ve açısal başlık hatalarına göre önce sistem eğitilmekte sonra benzetimlik çalıştığında eğitilen değerlere göre gürültü matris değerleri ayarlanarak hata minimize edilmeye çalışılmaktadır.

EK 5 SLAM ve FastSLAM AKIŞ Diyagramları



Şekil 1. SLAM Metodu Akış Diyagramı



Şekil 2 FastSLAM Metodu Akış Diyagramı

ÖZGEÇMİŞ

Adı Soyadı : Haydar Ankişhan

Doğum Yeri : Ankara

Doğum Tarihi : 1979

Medeni Hali : Evli

Yabancı Dili : İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise: Tuzluçayır Lisesi

Lisans: G.Antep Üniversitesi 2003

Yüksek Lisans: Ankara Üniversitesi 2007

Çalıştığı Kurum/Kurumlar ve Yıl

Başkent Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Öğretim Görevlisi 2009-devam

SCI - SCI Expanded Yayınları

Yılmaz, D., **Ankişhan, H.**; Analysis of snore sounds by using the largest lyapunov exponents. Journal Of Concrete And Applicable Mathematics, 2011; 9(2):146-153.(Math SCI)

Ankişhan, H., Yılmaz, D.; Comparison of SVM and ANFIS for Snore Related Sounds Classification by Using the Largest Lyapunov Exponent and Entropy. Computational and mathematical methods in medicine, 2013; 2013(2013):-.(SCI-E) <http://www.hindawi.com/journals/cmmm/2013/238937/>

Bulut, M., **Ankişhan H.**, Demircioğlu, E., Arı, S., Şengül, O.; A novel approach for estimating heat transfer coefficients of ethylene glycol water mixtures. Neural Computing and Applications, ; (Accepted).(SCI-E) <http://link.springer.com/article/10.1007%2Fs00521-013-1453-4>

Ankişhan, H., Arı, F., Tartan, E. O., Pakfiliz, A. G.; Central Difference Based FastSLAM

Approach to SLAM Simulations improved by Differential Evolution. TURKISH Journal Of Electrical Engineering & Computer Sciences. (Kabul edildi).(SCI-E), <http://mistug.tubitak.gov.tr/bdyim/kabul.php?dergi=elk>

Ankışhan, H., Arı, F.; A New Adaptive FastSLAM Approach to SLAM Problem, International Journal of Robotics and Automation, (SCI-E) (Girildi).

Ulusal hakemli dergilerde yayınlanan makaleler

Ankışhan, H., Efe M.; Eşzamanlı konum belirleme ve haritalamaya Kalman filtre yaklaşımları. Dicle üniversitesi mühendislik fakültesi dergisi, 2010; 1(1):13-20.

Ankışhan H., Arı F.; Parazit Yankılı Ortamda Eşzamanlı Konum Belirleme ve Harita Oluşturma Problemi için Veri İlişkilendirme. Dicle üniversitesi mühendislik fakültesi dergisi, Kasım, 2013.

Uluslararası Kongre Sunumları

Yılmaz, D., **Ankışhan, H.**; Analysis of snore sounds by using the largest Lyapunov exponent. 3. International interdisciplinary Chaos Symposium on Chaos and Complex Systems: istanbul; 21/05/2010 - 24/05/2010

Ankışhan, H.; Smoother Aided Neuro Fuzzy Kalman Filter Approach To Slam Problem. The International Association of Science and Technology for Development (IASTED-SIP 2011): Dallas, ABD; 14/12/2011 - 16/12/2011

Ankışhan, H., Yılmaz D.; Feature Extraction And Classification Of Snore Related Sounds. The International Association of Science and Technology for Development (IASTED-SIP 2011): Dallas, ABD; 14/12/2011 - 16/12/2011

Ankışhan, H., Arı F.; Snore Related Sound Classification Based on Time-Domain Features by Using ANFIS model. International Symposium on INnovations in Intelligent SysTems and Applications (INISTA 2011): İstanbul; 15/06/2011 - 18/06/2011

Ankışhan H., Tartan E. O., Arı F.; Square Root Unscented Based FastSlam Optimized By Particle Swarm Optimization Passive Congregation. IEEE International Conference on Mechatronics and Automation. Takamatsu, Kagawa, Japan, 3-6 August, 2013.

Ankışhan, H., Arı, F.; Adaptive Kalman Filter Based FastSLAM Approach to SLAM Problem. International Conference on Advanced Control, Automation and Robotics (ACAR2014). September 21-22, 2014, Hong Kong

Ulusal Kongre Sunumları

Ankışhan H., Yılmaz D.; Horlama ses sinyallerinin otoregresif tabanlı incelenmesi. IEEE 18th Signal Processing and Communications Applications Conference: Diyarbakır; 22/04/2010 - 24/04/2010

Ankışhan H., Ari F.; Horlamaya İlişkin Seslerin Kaotik Analizi. 20.Ieesinyal İşleme Ve İletişim Uygulamaları Kurultayı: Fethiye, Muğla, Türkiye; 18/04/2012 - 20/04/2012

Ankışhan H., Yılmaz D.; Horlama Seslerinin Ayarlanabilir Bulanık Mantık Çıkarımı Sistemi İle Sınıflandırılması. Tıp Teknolojileri Ulusal Kongresi, TIPTEKNO2011: belek, Antalya; 13/10/2011 - 16/10/2011

Ankışhan H., Efe M.; Eşzamanlı Lokalizasyon ve konum belirlemeye Kalman filtre yaklaşımları. Bilimde Modern Yöntemler Sempozyumu 2010, Dicle Üniversitesi Kongre Merkezi, 14-16 Ekim 2010-DİYARBAKIR: Diyarbakır; 14/10/2010 - 16/10/2010

Ankışhan H., Efe M.; Eşzamanlı konum belirleme ve harita oluşturmada ayarlanabilir bulanık mantık destekli Kalman filtre yaklaşımı. 19.IEEE Sinyal işleme ve iletişim uygulamaları kurultayı: kemer/antalya; 20/04/2011 - 22/04/2011

Ankışhan H., Arı F.; SLAM Probleminin Çözümünde Kare Kök Kokusuz Süzgeç Tabanlı FastSLAM Yaklaşımı. 21.IEEE Sinyal İşleme Ve İletişim Uygulamaları Kurultayı: Kıbrıs; 22/04/2013 - 24/04/2013