



**EŞ ZAMANLI KONUM BELİRLEME VE HARİTALAMA PROBLEMİNDE
İYİLEŞTİRİLMİŞ ÇÖZÜM İÇİN GKF, AGKF VE FASTLAM
UYGULAMALARI**

Serhat KARAÇAM

**YÜKSEK LİSANS TEZİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

KASIM 2021

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Serhat KARAÇAM

18/11/2021

EŞ ZAMANLI KONUM BELİRLEME VE HARİTALAMA PROBLEMİNDE İYİLEŞTİRİLMİŞ ÇÖZÜM İÇİN GKF, AGKF VE FASTSLAM UYGULAMALARI

(Yüksek Lisans Tezi)

Serhat KARAÇAM

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Kasım 2021

ÖZET

Eş zamanlı yer tayini ve haritalandırma (EZYTH) bir mobil robotun bilinmeyen bir ortamda hareketi sürecinde sadece sensör verilerinden faydalanarak ortamın haritasını çıkartması ve çıkarttığı bu haritayı da kullanarak kendi konumunu belirlemesi durumudur. EZYTH robotun otonom hareketinin sağlanması için çözülmesi gereken bir problemdir. Bu probelemi çözümünde olasılık tabanlı filtreler olan Parçacık Filtreleri (PF) ve Kalman Filtreleri (KF) kullanılır. Genişletilmiş Kalman Filtresi (GKF) en yaygın kullanılan olasılık tabanlı filtrelerdendir. Doğru bir şekilde tanımlanmış sistem modeli ve bilinen sabit gürültü istatistikleri sağlandığında GKF tutarlı sonuçlar veren bir yaklaşımdır. Fakat bu şartlar ideal benzetim ortamlarında sağlanabilir. Gerçek hayatta gürültünün karakteristiği sebebiyle sabit kabul edilemez hatta sürekli artan bir yapıda olduğunu bilinir. Bu sebepten dolayı gürültü istatistiklerini yinelemeli olarak tahmin eden Adaptif Genişletilmiş Kalman Filtresi (AGKF) önerilmiştir. Gürültü kovaryanslarının pozitifliğini korumak adına Değişim Kovaryans Tahmini (DKT) AGKF ile birlikte kullanılmıştır. PF'nin EZYTH probleminin çözümünde kullanıldığı FastSLAM uygulamalarının en önemli adımlarından birisi yeniden örneklemedir. Yeniden örnekleme sayesinde parçacıkların dejenere olma ihtimalinin önüne geçilir. Literatür içinde önerilen çok sayıdaki yeniden örnekleme metodundan en çok kullanılan beş tanesi bu çalışmada karşılaştırmalar için kullanılmıştır. Bu çalışmada değişen ortam şartlarında GKF, AGKF, AGKF ve PF çözümleri karşılaştırılmıştır. İşlem gürültüsünün hız ve açı bileşeninin, ölçüm gürültüsünün mesafe ve yön bileşeninin başlangıç değerindeki değişimlerin filtrelerin performansına olan etkileri ayrı ayrı incelenmiştir. Bu incelemeler yapılırken, robotun yaptığı gözlem ile tahmin arasındaki bağıntıyı kurma durumu olan veri ilişkilendirmesinin hareket öncesinde robota verildiği ve verilmediği durumlar göz önünde bulundurulmuştur. Karşılaştırmalar, robotun ve yer işaretlerinin doğru konumları ile tahmin edilen konumları arasındaki farkların RMSE değerleri üzerinden yapılmıştır. Karşılaştırmalar sonucu AGKF'nin diğer yöntemlerden daha iyi sonuçlar verdiği, DKT'nin bu sonuçlarda iyileştirmeler sağladığı görülmüştür.

Bilim Kodu : 90535

Anahtar Kelimeler : Eş zamanlı yer tayini ve haritalandırma (EZYTH), Genişletilmiş Kalman Filtresi (GKF), FastSLAM, Yeniden Örnekleme Metotları, Adaptif GKF

Sayfa Adedi : 101

Danışman : Doç. Dr. Tuğba Selcen NAVRUZ

EKF, AEKF AND FASTSLAM APPLICATIONS FOR IMPROVED SOLUTION OF SLAM PROBLEM

(M. Sc. Thesis)

Serhat KARAÇAM

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

November 2021

ABSTRACT

Simultaneous Localization and Mapping (SLAM) problem is that a mobile robot maps the environment by using only sensor data during its movement in an unknown place and determines its position by using this map. SLAM is a problem that needs to be solved in order to ensure the autonomous motion of robot. Particle Filters (PF) and Kalman Filters (KF), which are probability-based filters, are used to solve this problem. Extended Kalman Filter (EKF) is one of the most widely used probability-based filter. Provided a properly defined system model and known fixed noise statics, EKF is an approach that gives acceptable results. However, these conditions can be done in ideal simulation environments. It is known that in real life, noise cannot be considered constant due to its characteristics, and even noise has a constantly increasing structure. Therefore, Adaptive Extended Kalman Filter (AEKF) is used to estimate noise statics recursively. In order to maintain the positive definite characteristics of noise covariance Innovation Covariance Estimation (ICE) was used together with AGKF. One of the most important step of FastSLAM applications where PF is used to solve the SLAM problem is resampling. With resampling, degenerating possibility of particles is avoided. Five of the most used resampling methods in the literature are used for comparisons in this study. In this study, EKF, AEKF, AEKF-ICE and PF solutions are compared under varying environmental conditions. The effects of variances in the initial value of velocity and steering component of process noise, and range and bearing component of measurement noise on the performance of the filters are investigated separately. During these investigations, cases where data association, which is the state of establishing correlation between observation and prediction made by the robot, is given to the robot before motion or not, has been taken into consideration. Comparisons are made over the RMSE values of differences between true and predicted positions of robot and landmarks. As a result of the comparisons, it is seen that AEKF gave better results than other methods, and ICE provided improvements in these results.

Science Code : 90535

Key Words : Simultaneous Localization and Mapping (SLAM), Extended Kalman Filter (EKF), FastSLAM, Resample Methods, Adaptive EKF

Page Number : 101

Supervisor : Assoc. Prof. Dr. Tuğba Selcen NAVRUZ

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren, hatalarımı sabırla düzelten, akademik hayat ile ilgili tecrübelerini aktaran, hiçbir konuda yardımını esirgemeyen ve her zaman desteğini hissettiren kıymetli Hocam Doç. Dr. Tuğba Selcen NAVRUZ'a, bugünlere gelmemi sağlayan ve benim için hiçbir fedakarlıktan kaçınmayan sevgili aileme, bana inanan ve desteğini her zaman gösteren sevgili dostlarıma sonsuz teşekkürlerimi sunarım.



İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ.....	xii
SİMGELER VE KISALTMALAR.....	xix
1. GİRİŞ.....	1
2. LİTERATÜR TARAMASI.....	5
3. TEORİK ALTYAPI	9
3.1. Kalman Filtresi	9
3.2. Genişletilmiş Kalman Filtresi	11
3.2.1. Taylor açılımı ile doğrusallaştırma işlemi.....	13
3.2.1. GKF algoritması.....	15
3.3. Adaptif Genişletilmiş Kalman Filtresi	16
3.3.1. Maksimum olabilirlik tahmini.....	17
3.3.2. Beklenti maksimizasyonu	19
3.3.3. Tek adımlı ve sabit noktalı düzleme algoritması ile GKF'nin geliştirilmesi	22
3.3.4. Tekrarlı ve zamanla değişen gürültü istatistikleri formu.....	23
3.3.5. Değişim kovaryansı tahmincisi	26
3.3.6. AGKF algoritması	27
3.4. Parçacık Filtresi	28
3.4.1. Önem dağılımı ve Önem ağırlıkları.....	30

	Sayfa
3.4.2. Yeniden örnekleme	32
3.5. Eş Zamanlı Yer Tayini Ve Haritalandırma (EZYTH).....	37
3.5.1. Hareket ve gözlem modelleri	39
3.5.2. Veri ilişkilendirmesi	41
4. BENZETİM SONUÇLARI VE TARTIŞMA.....	45
4.1. GKF ve AGKF'nin Karşılaştırması	48
4.1.1. İşlem gürültü bileşeni q_V 'nin değişimi.....	48
4.1.2. İşlem gürültü bileşeni q_G 'nin değişimi.....	51
4.1.3. İşlem gürültü bileşenleri q_V ve q_G 'nin değişimi	55
4.1.4. Ölçüm gürültü bileşeni r_R 'nin değişimi	58
4.1.5. Ölçüm gürültü bileşeni r_B 'nin değişimi	61
4.1.6. Ölçüm gürültü bileşenleri r_R ve r_B 'nin değişimi	65
4.2. AGKF ve AGKF-DKT'nin Karşılaştırması.....	68
4.2.1. İşlem gürültü bileşenlerinin değişimi.....	68
4.2.2. Ölçüm gürültü bileşenlerinin değişimi.....	70
4.3. FastSLAM Yeniden Örnekleme Metotlarının Karşılaştırılması	72
4.3.1. İşlem gürültü bileşeni q_V 'nin değişimi.....	73
4.3.2. İşlem gürültü bileşeni q_G 'nin değişimi.....	75
4.3.3. İşlem gürültü bileşenleri q_V ve q_G 'nin değişimi	77
4.3.4. Ölçüm gürültü bileşeni r_R 'nin değişimi	80
4.3.5. Ölçüm gürültü bileşeni r_B 'nin değişimi	82
4.3.6. Ölçüm gürültü bileşenleri r_R ve r_B 'nin değişimi	84
5. SONUÇLAR VE ÖNERİLER	87
KAYNAKLAR	91

	Sayfa
EKLER.....	97
EK-1. AGKF ve AGKF-DKT'nin veri ilişkilendirmesi robota verildiği zaman değişen gürültü başlangıç değerleri için hesaplanan RMSE değerlerinin gösterildiği grafikler.....	98
ÖZGEÇMİŞ	100



ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Kalman Filtresi algoritması.....	10
Çizelge 3.2. Genişletilmiş Kalman Filtresi algoritması	15
Çizelge 3.3. Adaptif Genişletilmiş Kalman Filtresi algoritması	27
Çizelge 3.4. Parçacık Filtresi algoritması	32
Çizelge 4.1. Gürültü bileşenlerinin uygulanan testlerdeki başlangıç değerleri	46
Çizelge 4.2. Robot ve yer işareti konum hatalarının değişen q_v gürültü başlangıç değerlerindeki ortalama hata değerleri	49
Çizelge 4.3. Robot ve yer işareti konum hatalarının değişen q_G gürültü başlangıç değerlerindeki ortalama hata değerleri	52
Çizelge 4.4. Robot ve yer işareti konum hatalarının değişen r_R gürültü başlangıç değerlerindeki ortalama hata değerleri	56
Çizelge 4.5. Robot ve yer işareti konum hatalarının değişen r_B gürültü başlangıç değerlerindeki ortalama hata değerleri	59
Çizelge 4.6. Robot ve yer işareti konum hatalarının değişen işlem gürültü değerlerindeki ortalama hata değerleri	62
Çizelge 4.7. Robot ve yer işareti konum hatalarının değişen ölçüm gürültü değerlerindeki ortalama hata değerleri	66
Çizelge 4.8. Robot ve yer işareti konum hatalarının değişen işlem gürültü değerlerindeki ortalama hata değerleri	69
Çizelge 4.9. Robot ve yer işareti konum hatalarının değişen ölçüm gürültü değerlerindeki ortalama hata değerleri	72
Çizelge 4.10. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen q_v gürültü başlangıç değerlerindeki ortalama hata değerleri	74
Çizelge 4.11. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen q_G gürültü başlangıç değerlerindeki ortalama hata değerleri	76
Çizelge 4.12. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen r_R gürültü başlangıç değerlerindeki ortalama hata değerleri	78

Çizelge	Sayfa
Çizelge 4.13. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen r_B gürültü başlangıç değerlerindeki ortalama hata değerleri	81
Çizelge 4.14. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen işlem gürültü başlangıç değerlerindeki ortalama hata değerleri	83
Çizelge 4.15. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen ölçüm gürültü başlangıç değerlerindeki ortalama hata değerleri	85



ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 3.1. Gauss rastgele değişkeninin doğrusal olmayan fonksiyon içindeki değişimi. (a) Gauss olmayan sonuç dağılımı (b) doğrusal olmayan $g(x)$ fonksiyonu (c) Gauss dağılımına sahip olan $p(x)$ olasılık dağılımı.....	12
Şekil 3.2. Gauss rastgele değişkeninin Taylor yaklaşımı ile yaklaşık doğrusallaştırılmış fonksiyon içindeki(kırmızı) değişimi. (a) olasılık dağılımı yaklaşık doğrusallaştırılmış fonksiyondan geçtikten sonra korunan Gauss dağılımı(kırmızı) (b) doğrusal olmayan $g(x)$ fonksiyonunun Taylor yaklaşımı (kırmızı) (c) Gauss dağılımına sahip olan $p(x)$ dağılımı.....	13
Şekil 3.3. Standart bir Gauss olasılık dağılımı.....	17
Şekil 3.4. PF'de kullanılan parçacık gösterimi. a) Doğrusal olmayan fonksiyon sonucu rastgele değişkenin aldığı form. b) Doğrusal olmayan $g(x)$ fonksiyonu. c) Gauss rastgele değişkeni $p(x)$ ve bu değişkenden alınan örnekler	29
Şekil 3.5. PF için önem örneklemesinin gösterimi. a) Hedeflenen dağılım $p(x)$ b) Parçacıkların alındığı $\pi(x)$ öneri dağılımı. c) $p(x)$ ve $\pi(x)$ oranlaması sonucu elde edilen önem ağırlıkları.	31
Şekil 3.6. Yeniden örnekleme örneği.....	33
Şekil 3.7. Çok terim yeniden örnekleme görselleştirilmiş hali.....	34
Şekil 3.8. Çok katmanlı yeniden örnekleme görselleştirilmiş hali	35
Şekil 3.9. Sistemik yeniden örnekleme görselleştirilmesi	36
Şekil 3.10. Harita bilindiğinde yer tahmini örneği	37
Şekil 3.11. Robot konumları bilindiğinde haritalandırma örneği	38
Şekil 3.12. Eş zamanlı yer tahmini ve haritalandırma örneği	39
Şekil 3.13. x_k durum vektörü ($[x_R, m]$), u_k komut vektörü ve z_k gözlem vektörünün ilişkisini gösteren grafik modeli	40
Şekil 3.14. PF'de veri ilişkilendirmesi sorununun gösterimi	42
Şekil 3.15. Doğrulama kapısının gösterimi.....	43
Şekil 3.16. Gözlemin hangi doğrulama kapısı aralığına ait olduğu ile ilgili belirsizliğin gösterimi	43

Şekil	Sayfa
Şekil 4.1. Deneylerde kullanılacak olan harita	45
Şekil 4.2. q_v 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	48
Şekil 4.3. q_v 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum	49
Şekil 4.4. Veri ilişkilendirmesi verildiği durumda, $q_v = 0.16m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	50
Şekil 4.5. Veri ilişkilendirmesi verilmediği durumda, $q_v = 0.16m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	50
Şekil 4.6. Veri ilişkilendirmesi bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	51
Şekil 4.7. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	51
Şekil 4.8. q_G 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	52
Şekil 4.9. q_G 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum	52
Şekil 4.10. Veri ilişkilendirmesi verildiği durumda, $q_G = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	53
Şekil 4.11. Veri ilişkilendirmesi verilmediği durumda, $q_G = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	53
Şekil 4.12. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	54

Şekil	Sayfa
Şekil 4.13. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	54
Şekil 4.14. r_R 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum	55
Şekil 4.15. r_R 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	55
Şekil 4.16. Veri ilişkilendirmesi verildiği durumda, $r_R = 0.16m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	56
Şekil 4.17. Veri ilişkilendirmesi verilmediği durumda, $r_R = 0.16m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	57
Şekil 4.18. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	57
Şekil 4.19. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	58
Şekil 4.20. r_B 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum	58
Şekil 4.21. r_B 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	59
Şekil 4.22. Veri ilişkilendirmesi verildiği durumda, $r_B = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	60
Şekil 4.23. Veri ilişkilendirmesi verilmediği durumda, $r_B = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni	60

Şekil	Sayfa
Şekil 4.24. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	61
Şekil 4.25. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	61
Şekil 4.26. q_v ve q_G 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	62
Şekil 4.27. q_v ve q_G 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	62
Şekil 4.28. Veri ilişkilendirmesi verildiği durumda, $q_v = 0.16m$ ve $q_G = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni.....	63
Şekil 4.29. Veri ilişkilendirmesi verilmediği durumda, $q_v = 0.16m$ ve $q_G = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni.....	64
Şekil 4.30. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	64
Şekil 4.31. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	64
Şekil 4.32. r_R ve r_B 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	65
Şekil 4.33. r_R ve r_B 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durum b) veri ilişkilendirmesi bilinmediği durum.....	65
Şekil 4.34. Veri ilişkilendirmesi verildiği durumda, $r_R = 0.16m$ ve $r_B = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni.....	66

Şekil	Sayfa
Şekil 4.35. Veri ilişkilendirmesi verilmediği durumda, $r_R = 0.16m$ ve $r_B = 1.6\pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni.....	67
Şekil 4.36. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	67
Şekil 4.37. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF	68
Şekil 4.38. q_v değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu	69
Şekil 4.39. q_G değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu	69
Şekil 4.40. q_v ve q_G değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu.....	70
Şekil 4.41. r_R değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu	71
Şekil 4.42. r_B değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu	71
Şekil 4.43. r_R ve r_B değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu	71
Şekil 4.44. q_v değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	73
Şekil 4.45. q_v değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	73
Şekil 4.46. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	74
Şekil 4.47. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	75

Şekil	Sayfa
Şekil 4.48. q_G değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	75
Şekil 4.49. q_G değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	76
Şekil 4.50. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	77
Şekil 4.51. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	77
Şekil 4.52. r_r değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	78
Şekil 4.53. r_r değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde ...	78
Şekil 4.54. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	79
Şekil 4.55. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	79
Şekil 4.56. r_b değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	80
Şekil 4.57. r_b değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde ...	80
Şekil 4.58. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	81
Şekil 4.59. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	82

Şekil	Sayfa
Şekil 4.60. q_v ve q_g değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	82
Şekil 4.61. q_v ve q_g değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	83
Şekil 4.62. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	84
Şekil 4.63. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	84
Şekil 4.64. r_R ve r_B değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde	85
Şekil 4.65. r_R ve r_B değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verilmediğinde b) veri ilişkilendirmesi verildiğinde	85
Şekil 4.66. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	86
Şekil 4.67. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ	86

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklamalar
m	Metre
s	Saniye
m/s	Metre/Saniye
rad	Radyan
Σ	Kovaryans
μ	Ortalama
x_k	k anındaki durum vektörü
P_k	k anındaki durum kovaryans matrisi
u_k	k anındaki komut vektörü
ϵ_k	İşlem gürültüsü
δ_k	Ölçüm gürültüsü
q_k	İşlem gürültü ortalama vektörü
Q_k	İşlem gürültü kovaryans matrisi
q_v	İşlem gürültüsü hız bileşeni
q_G	İşlem gürültüsü yön bileşeni
r_k	Ölçüm gürültü ortalama vektörü
R_k	Ölçüm gürültü kovaryans matrisi
r_R	Ölçüm gürültüsü mesafe bileşeni
r_B	Ölçüm gürültüsü açı bileşeni
z_k	Gözlem vektörü
K_k	Kalman kazancı
F_k	İşlem fonksiyonu jakobiyeen matrisi
H_k	Ölçüm fonksiyonu jakobiyeen matrisi
$e_{z,k}$	Değişim vektörü
n	Durum vektörü uzunluğu
m	Ölçüm vektörü uzunluğu

Simgeler**Açıklamalar**

k	Zaman
B	Beklenti
tr	Matris izi
I	Durum matrisi
d_k	Ağırlık katsayısı
b	Sönümleme faktörü
N	Pencere boyutu
M	Parçacık sayısı
X_k	Parçacık kümesi
w_k^i	i . parçacığın ağırlığı
π	Öneri dağılımı
J_k^m	Kümülatif toplam
s_m	Katmanlar
V_k	Robotun k anındaki hızı
G_k	Robotun k anındaki dönüş açısı
WB	Ön ve arka teker arasındaki mesafe
∇k	Zaman aralığı
M_k	Mahalanobis mesafesi
Λ_k	İlişkilendirme olasılığı
γ_k	Doğrulama kapısı

Kısaltmalar**Açıklamalar**

AGKF	Adaptif Genişletilmiş Kalman Filtresi
AMC	Ardışık Monte-Carlo
AYÖ	Artık yeniden örnekleme
BM	Beklenti-Maksimizasyon
ÇKYÖ	Çok katmanlı yeniden örnekleme
ÇTYÖ	Çok terimli yeniden örnekleme
DKT	Değişim kovaryansı tahmini
EZYTH	Eş zamanlı yer tayini ve haritalama

Kısaltmalar**Açıklamalar****GKF**

Genişletilmiş Kalman Filtresi

GPS

Küresel Konumlandırma Sistemi

KF

Kalman Filtresi

MYÖ

Metropolis yeniden örnekleme

NEYK

Normalize edilmiş yakın komşuluk

ODF

Olasılık dağılım fonksiyonu

PF

Parçacık Filtresi

RMSE

Root Mean Square Error

SYÖ

Sistemik yeniden örnekleme

1. GİRİŞ

Eş Zamanlı Yer Tayini ve Haritalama (EZYTH) problemi uzun yıllardır araştırmacıların ilgisini çeken, otonom robot hareketinin temelini oluşturan bir konudur. Bulunduğu yeri bilmeyen ve ortam hakkında bir fikri olmayan bir robotun sadece çevresi ile kendi konumu arasında yaptığı gözlemlere dayanarak bir harita oluşturmaya daha sonra da bu haritayı kullanarak gideceği yolu seçmesi, harita üzerindeki konumunu belirlemesi bir robotu gerçekten otonom hale getirir. EZYTH probleminin çözümü robotun otonom hareket etmesinin de anahtarıdır [1,2].

Robotun çevresini iyi anlaması ve bulunduğu ortamdaki konumunu doğru şekilde belirlemesi önemlidir. Robotun konumu genellikle haritadaki konumu ve oryantasyonunun bulunduğu bir robot durumu olarak tanımlanır. Harita ise duvarlar, engeller, yer işaretleri gibi çevreyi tanımlayan bir dizi özelliklere sahip olan, bir mobil robotun ortamdaki yerini bulması ve yol planlaması yapabilmesi için temel bir gereksinimdir [3]. Robotun sensör verileri yardımıyla, hakkında ön bir bilgiye sahip olunan bir ortamda konumunu tahmin etmesine robot lokalizasyonu denir [4]. Bununla birlikte tamamen otonom bir mobil robotun ortam ile ilgili bir ön bilgiye sahip olmadığı veya sahip olduğu bilgilerin kısıtlı olduğu durumda da ortamı keşfedebilmesi beklenir. Robot konumunu belirlemek için Küresel Konumlandırma Sisteminden (GPS) faydalanabilir [5] fakat iç alan, yer altı veya su altı gibi GPS'in kullanılamadığı durumlarda robotun konumunu belirlemesi güçtür [6]. Ayrıca kullanıcı tarafından oluşturulan haritaların yerini robotun kendisinin oluşturduğu haritaların almasına olan ilginin artması haritalama gereksinimini zorunlu hale getirmiştir [3,7].

Robotların günlük hayata aktif olarak katılmaya başlaması, evlerde, bahçelerde, su altı ve hava uygulamalarında, uzay faaliyetlerinde ve birçok alanda robotların kullanılması EZYTH probleminin çözülmesi gerekliliğini ortaya çıkartmıştır. EZYTH probleminin çözümü son yıllarda robotik topluluğunun önemli başarılarından birisi olmuştur. EZYTH bir dizi farklı formda teorik bir problem olarak formüle edilmiş ve çözülmüştür. Teorik ve kavramsal düzeyde EZYTH artık çözülmüş bir problem olarak kabul edilse de çözümlerin pratik olarak gerçekleştirilmesinde ve özellikle algısal olarak zengin haritaların oluşturulmasında ve kullanılmasında önemli sorunlar günümüzde hala devam etmektedir [2].

Birçok bilim insanı, EZYTH probleminin çözümünün daha kararlı, daha iyi performans verecek durumu gelmesi için farklı çalışmalar yürütmeye devam etmektedir. EZYTH üzerinde 25 yılı aşkın süredir devam eden çalışmalar birçok iç ve dış mekân uygulamasında EZYTH'nin pratik, gerekli ve önemli bir çözüm yöntemi olabileceğini göstermiştir. Birçok çözüm yöntemi üretilmiş olmakla beraber araştırmacılar yeni yöntemler üzerinde çalışmaya devam etmektedir. Bu çalışmaların genel kapsamı çözümün, az tahmin hatasıyla, uzun ömürlü ve en hızlı haritalamayı düşük güç ve düşük hesaplama maliyeti ile yapılmasıdır [3].

EZYTH probleminin çözümü için olasılık tabanlı ve olasılıksal olmayan yaklaşımlar kullanılır. Olasılık tabanlı yaklaşımlar EZYTH probleminin çözümünde ana odak noktasını oluşturur. Olasılık yöntemleri ağırlıklı olarak Bayes tahmin yöntemine dayanan Parçacık Filtreleri (PF) ve Kalman Filtreleri (KF)'dir [3]. Kalman filtresi sistemi tamamen doğrusal kabul eder [8]. Buna karşın gerçek hayat uygulamalarında sistemler doğrusal değildir. Doğrusal olmayan sistemlerin Taylor açılımı kullanılarak doğrusallaştırılması ve KF içinde kullanılması ile Genişletilmiş Kalman Filtresi (GKF) oluşturulmuştur [9]. Her ne kadar benzetim ortamlarında GKF etkili şekilde kullanılabilse de GKF'nin tutarlı bir sistem modeli ve bilinen gürültü istatistiklerine ihtiyacı vardır. Bu şartlar ise gerçek hayat uygulamalarında sağlanamaz [10]. Robotun hareket sırasında değişen gürültü istatistiklerini yaklaşık olarak tahmin edebilmesi ile Adaptif Genişletilmiş Kalman Filtresi (AGKF) ortaya çıkmıştır [11]. AGKF sonucu tahmin edilen işlem ve ölçüm gürültü kovaryans matrislerinin boyutları ve içerdiği değerler açısından yapıları oldukça karmaşıktır. Robot ilerledikçe bu matrislerin karmaşıklığı ve pozitif tanımlılığını yitirme ihtimali artar. Bu sebeple de Değişim Kovaryansı Tahmini (DKT) adı verilen yöntem ile kovaryans matrislerinin pozitif tanımlılığını koruması garanti altına alınmıştır [12].

PF'nin amacı GKF'nin aksine olasılık dağılımını Gauss ile göstermeyip yine aynı dağılımdan alınmış rastgele durum örneklemeleri ile ifade etmektir [6]. PF ve GKF beraber kullanılarak FastSLAM algoritması oluşturulmuştur [13]. PF yöntemlerinin hepsi parçacık yayılımı, ağırlıkların hesaplanması ve yeniden örnekleme adımlarına sahiptir. Bu adımlar sırasıyla parçacıkların üretilmesi, ağırlıkların atanması ve parçacık kümesini ve ağırlıklarını başka bir küme ile değiştirilmesidir. Yeniden örnekleme için birden fazla yöntem önerilmiştir [14].

Bu tez çalışmasında EZYTH probleminin çözümünde kullanılan GKF AGKF ve PF algoritmalarının farklı gürültü başlangıç seviyelerinde karşılaştırılması yapılmıştır. Aynı zamanda PF'nin yeniden örnekleme adımı için önerilen farklı metotların da gürültü değişimine verdikleri tepki gözlemlenmiştir. Bu gürültü değişim ifadeleri robotun hızını ve dönüş açısını etkileyen işlem gürültüsü ve sensörün yaptığı gözlemin mesafesi ve yönünü etkilen ölçüm gürültüsünden oluşur. Bu karşılaştırmalar yapılırken veri ilişkilendirmesinin robota hareket öncesinde verildiği ve verilmediği durumlar incelenmiş ve bu incelemeler sonucu robotun ve yer işaretlerinin gerçek konumlarıyla tahmin edilen konumları arasındaki hata oranları, Root Mean Square Error (RMSE) değerleri hesaplanmıştır.

Bu tez çalışmasının 2. bölümünde EZYTH problemi ve onun çözümünde kullanılan olasılık tabanlı yöntemler olan KF, GKF, AGKF ve PF'nin tarihsel gelişimi, 3. bölümünde bu yöntemlerin ve EZYHT probleminin teorik altyapısının açıklanması verilmiştir. 4. bölümde GKF, AGKF ve PF yöntemleri farklı gürültü başlangıç seviyelerinde ve veri ilişkilendirmesinin robota hareketten önce verilip verilmediği durumlarda benzetim ortamında test edilmiş ve RMSE, hata oranları bulunarak yöntemlerin karşılaştırılması yapılmıştır. 5. ve son bölümde yapılan çalışmanın sonuçları, çıkarımlar ve gelecek çalışmalar için yapılan öneriler bulunmaktadır.

2. LİTERATÜR TARAMASI

EZYTH problemi ilk olarak 1986 yılında Kaliforniya’da yapılan IEEE Robotik ve Otomasyon Konferansında ele alınmıştır. O zamanlar olasılık tabanlı metotlar hem robotik hem de yapay zekâ alanında yeni yeni tanıtılmaya başlanmıştır. Bu konferansın sonucunda tutarlı bir olasılık tabanlı haritalandırmanın çözülmesi için gereken önemli sorunlar olduğu fark edilmiştir [2]. Devam eden yıllarda Simith ve Cheesman nesnelerin göreceli konumlarını temsil eden koordinatları ile beklenen kovaryans hatasının arasındaki ilişkiyi tahmin eden genel bir yöntem önermiştir [15]. Durrant-Whyte bir robot hareket sistemine etki eden belirsizliklerin tutarlı bir şekilde temsil edilmesi ve hesaplanabilmesi için istatistiksel bir çözüm önermiştir [16]. Bu iki çalışma bir haritadaki farklı yer işaretlerinin konum tahminleri arasında bir bağlantı olması gerektiğini ve bu bağlantının gözlemler arttıkça büyüyeceğini göstermiştir.

Bir mobil robotun bilinmeyen bir ortamda hareket ederken yaptığı gözlemlerin araç konumundaki hatalar nedeniyle yer işaretlerinin tahmini ile araç konumunun birbiriyle ilişkili olduğu gösterilmiştir [17]. Yer tayini ve haritalandırma probleminin tam bir çözümü için her gözlem sonrasında yer işaretinin güncellenmesi ve araç konumu ile yer işaretlerinin konumlarından oluşan ortak bir durumun oluşturulması gerekir. Ancak buna karşılık olarak oluşacak durum vektörünün boyutu oldukça büyük olacaktır [2]. Fakat bu çalışma sırasında haritanın yakınsamayacağı ve bunun yerine sınırsız hata büyümesi ile rastgele bir hareket sergileyeceği varsayılmıştı. Bu nedenle harita probleminin hesaplama zorluğu göz önüne alındığında araştırmacılar yer işaretleri arasındaki bağlantıyı en aza indiren, hatta ortadan kaldırılmasını varsayan bir dizi yaklaşım önermişlerdir [18,19].

Problem çözümü için önemli bir atılım en aza indirilmeye çalışılan yer işaretleri arasındaki bağlantının aslında büyüdükçe çözümü iyileştirdiğinin anlaşılması ile olmuştur. 1995 yılında Uluslararası Robotik Araştırmaları Sempozyumunda sunulan bir araştırma makalesinde EZYTH kısaltma olarak ilk kez kullanılmış ve EZYTH probleminin yapısı anlatılmıştır [20]. Csorba bu teori üzerine olan çalışmaları ve ilk sonuçları doktora tezinde ortaya koymuştur [21]. Bu sempozyumdan sonra dünya çapında EZYTH probleminin çözümü için birçok ciddi çalışma başlamıştır [22–25].

Çalışmalar daha sonrasında hesaplama verimliliğini artırmaya ve veri ilişkilendirme ve döngü kapatma ile ilgili sorunları ele almaya odaklanmıştır. 1999 yılında Uluslararası Robotik Araştırma Sempozyumunda (ISRR '99) ilk EYZTH oturumu yapılmış ve burada Thrun ve ark. tarafından önerilen yaklaşımda KF tabanlı EYZTH yöntemleri ile yer tayini ve haritalama yöntemleri arasında bir derecede benzerlik elde edilmiştir [26].

KF, sistemlerin doğrusal olduğu yaklaşımı ile EYZTH problemini çözmeye çalışmıştır [8]. Fakat sistemler gerçek hayatta doğası gereği doğrusal değildir. Bunun için doğrusal olmayan sistemler doğrusal hale getirilerek problemin yaklaşık bir çözümü elde edilebilir. Birinci derecen Taylor açılımı kullanılarak yapılan bu doğrusallaştırma işlemi sonucu GKF ortaya çıkmıştır [4]. GKF EYZTH probleminin çözümüne en yaygın kullanılan yaklaşımlardan birisidir [27]. Kolay uygulanabilirliği ve yüksek kalitede çözümler üretebilmesi EYZTH çözümünde GKF'nin yaygın olarak kullanılmasında oldukça etkili olmuştur [28,29]. GKF tabanlı aralıklı ölçümler ile mobil robotun yer tayininin yapılması [30], GKF tabanlı EYZTH'nin gözlemlenebilirlik açısından incelenmesi [31] gibi çalışmalar GKF'nin robot yolunu ve belirli bir ortama yayılan yer işaretlerini tahmin etmede oldukça tutarlı sonuçlar verdiğini göstermiştir. Fakat bu çalışmaların hepsi benzetim uygulamalarında etkin sonuçlar vermiştir. GKF'nin verimli çalışması için tutarlı bir sistem modeline ve bilinen gürültü istatistikleri değerlerine ihtiyacı vardır. Karakteristiği gereği belirsizlikler sabit kalmaz ve sürekli artan bir yapıya sahiptir. Bu da sistemin zaman ilerledikçe verimsiz çalışmasına ve doğru olmayan sonuçlar vermesine sebep olur [12]. Bu sebeple gürültü istatistikleri önden belirlenemez, aksine robot ilerledikçe değişen bir yapıya sahiptir. Gürültü istatistikleri yaklaşık olarak tahmin edilerek GKF iyileştirilmiş ve adaptif GKF olarak isimlendirilmiştir [10]. Huang ve ark. otonom sualtı araçlarının yer tayininin yapılmasında bilinmeyen ölçüm gürültülerinin tahmini için hata kovaryansı ve ölçüm gürültü kovaryanslarının tahminini yaptığı bir yöntem önermiştir. Bu tahmin beklenti-maksimizasyonu (BM) ve Maksimum Olabilirlik Tahmini (MOT) kullanılarak yinelemeli şekilde yapılır [11]. Shao ve ark. otonom su altı araçlarında en son ölçüm verilerini kullanarak ölçüm ve işlem gürültü kovaryans matrisini tahmin ettiği bir yaklaşım önermiştir [32]. KH Kim ve ark. AGKF tabanlı bir iki aşamalı GKF çözümü önermişlerdir. Bu çözüm adaptif sönümlemeli bir GKF yaklaşımı ile yapılmıştır [33].

Çoğu yaklaşım gürültü istatistiklerinin belirli kısımlarını, örneğin sadece ölçüm gürültü kovaryansını tahmin ederken geriye kalan işlem gürültüsü değerleri için bir tahmin işlemi

yapmamaktır. Tian ve ark. MOT [34] ve BM [35] tabanlı tüm gürültü değerlerinin tahmin edildiği bir yöntem önermiştir [10]. Bu yaklaşım aynı zamanda Değişim Kovaryans Tahmini yöntemi kullanarak elde edilen gürültü istatistikleri kovaryans matrislerinin pozitif tanımlı kalması sağlamıştır.

Del Moral 1996 yılında Parçacık Filtresi terimini ilk kez kullanmıştır [36]. Ayrıca Liu ve ark. yayınladığı makalede PF yerine aynı uygulama için Ardışık Monte-Carlo (AMC) ifadesini tercih etmişlerdir ve önerdikleri bu yöntem sistemin doğrusal olup olmadığına bakmaksızın yer tayini sorununu verimli şekilde çözebilen, gözlemlerle ilgili Gauss gürültü varsayımına ihtiyaç duymayan bir filtre türüdür [37]. PF doğrusal ve Gauss yapısında olmayan modelleri oldukça iyi tahmin edebilen, hesaplaması kolay ve herhangi bir olasılık dağılımını rahatlıkla tahmin edebilen bir yaklaşımdır [38]. PF'ye dayalı yöntemlerde üretilen parçacıklar, içerisinde robotun o anki konumunun ve ortamın bilgilerini içerir. Bu sebeple durum uzayının boyutu ve yer işareti sayısı arttıkça PF'nin hesaplama maliyeti fazlasıyla artar ve karmaşık ortamlarda gerçek zamanlı uygulamalarda kullanılmasının zorlaşmasına sebep olur [39]. PF ve GKF'nin beraber kullanılması sonucu önerilen EZYTH problemi çözümü FastSLAM olarak adlandırılır [13]. PF'nin en büyük sorunlarından biri seçilen parçacıkların dejenere olma ihtimalidir. Bunun önüne geçebilmek için robot konumunu daha doğru tahmin ettiğine inanılan, ağırlığı fazla olan parçacıklar tekrarlı olarak parçacık setine seçilir ve bu yönteme yeniden örnekleme denir [40]. Yeniden örneklemenin birçok farklı yöntemi literatürde bulunabilir [14].

Bu tez çalışmasında son yıllara ait, gürültü istatistiklerinin her adımda yenilemeli olarak tahmin edildiği AGKF [10] ile GKF, FastSLAM algoritmaları birbirleri arasında karşılaştırılmıştır. Ayrıca literatürde en çok kullanılan beş yeniden örnekleme metodu [14] kendi aralarında kıyaslanmıştır. Karşılaştırmalar yapılırken Bailey tarafından hazırlanan ve açık kaynaklı olarak araştırmacılarla paylaşılan “SLAM Simulator” kullanılmıştır [2,41]. Karşılaştırmaların ve yorumların gerçek hayat uygulamalarına benzer olması için farklı gürültü başlangıç seviyelerinde ve veri ilişkilendirmesinin robota önceden verilip verilmeme durumuna göre yapılmıştır. Robotun ve yer işaretlerinin gerçek konumları ile tahmin edilen konumları arasındaki hata oranları bulunmuş ve bunlar tablolar ve grafikler üzerinde gösterilmiştir.



3. TEORİK ALTYAPI

Bayes filtreleri robotun, önceki konum bilgileri ve yaptığı gözlemleri birleştirerek çevresi hakkındaki belirsizliği ve çevresi hakkındaki tahminini modellemesini sağlar. Gauss filtreleri ise olasılık dağılımlarının hepsinin Gauss dağılımına sahip olduğu varsayılan, Bayes filtrelerinin uygulamalarındandır. Gauss filtrelerinin tümünde Eş. 3.1’de gösterildiği gibi durum tahmininin olasılık dağılımının çok değişkenli normal dağılımlarla ifade edildiği düşünülür [4].

$$p(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right\} \quad (3.1)$$

Burada x durum vektörünü μ ve Σ , sırasıyla, durum vektörünün ortalamasını ve kovaryans matrisini gösterir.

Bu başlık altında Bayes filtrelerinin önemli uygulamaları olan Kalman Filtresi, Genişletilmiş Kalman Filtresi ve Parçacık Filtresi anlatılacaktır. Ayrıca GKF’nin iyileştirilmiş bir formu olan Adaptif Genişletilmiş Kalman Filtresi anlatılacaktır.

3.1. Kalman Filtresi

Bayes filtreleri arasında en yaygın kullanılan uygulamalardan birisi olan Kalman Filtresi (KF) 1950li yıllarda Rudolph Emil Kalman tarafından doğrusal olan sistemlerde filtreleme ve tahmin tekniği olarak önerilmiştir [8].

Kalman Filtresinde olasılık dağılımlarının hepsinin Gauss dağılımına sahip olduğu varsayılır. Aynı zamanda hareket ve gözlem modelleri doğrusal olduğu varsayılır. Bu modeller aşağıdaki gibi gösterilebilir.

$$x_k = A_k x_{k-1} + B_k u_k + \epsilon_k \quad (3.2)$$

$$z_k = C_k x_k + \delta_k \quad (3.3)$$

Burada k zamanı, x_k ve x_{k-1} , sırasıyla, k ve $k - 1$ anlarındaki durum vektörlerini, u_k kontrol vektörünü ve z_k ise gözlem vektörünü gösterir. A_k , robota herhangi bir komut

uygulanmadığında durum vektöründeki değişimi gösterirken B_k komut uygulandığı zaman durum vektöründeki değişimi ifade eder. C_k ise x_k durumunu z_k gözlemiyle ilişkilendirir. ϵ_k ve δ_k , sırasıyla, işlem ve gözlem gürültülerini ifade eden rastgele değişkenlerdir ve q_k ve r_k ortalama değerine ve Q_k ve R_k kovaryanslarına sahiptirler. Eş. 3.2 ve 3.3 tüm KF ailesinin temelini oluşturur. Çizelge 3.1’de KF algoritmasının genel formu verilmiştir.

Çizelge 3.1: Kalman Filtresi Algoritması [4]

1	Kalman Filtresi Algoritması
2	$\bar{\mu}_k = A_k \mu_{k-1} + B_k u_k$
3	$\bar{\Sigma}_k = A_k \Sigma_{k-1} A_k^T + Q_k$
4	$K_k = \bar{\Sigma}_k C_k^T (C_k \bar{\Sigma}_k C_k^T + R_k)^{-1}$
5	$\mu_k = \bar{\mu}_k + K_k (z_k - C_k \bar{\mu}_k)$
6	$\Sigma_k = (I - K_k C_k) \bar{\Sigma}_k$
7	return μ_k, Σ_k

Çizelge 3.1’de verilen algoritmada μ_{k-1} ve Σ_{k-1} bir önceki durumun ortalama vektörünü ve kovaryans matrisini, u_k hareket komut vektörünü, $\bar{\mu}_k$ ve $\bar{\Sigma}_k$ ilk durum tahmininin ortalama vektörünü ve kovaryans matrisini, K_k Kalman kazancını, z_k gözlem vektörünü, μ_k ve Σ_k son durum tahmininin ortalama vektörünü ve kovaryans matrisini gösterir.

Çizelge 3.1’de verilen algoritmaya göre 2. ve 3. adımlarda ortalama vektörü ve kovaryans matrisi, $k - 1$ zamanındaki robota uygulanan komut ile tahmin edilir. Bu kısma durum tahmin adımı denir. Durum ortalama vektörü tahmini A_k matrisine bağlı olduğu için kovaryans tahmini de A_k ’ye bağlı olacaktır. Burada iki kere çarpılmasının sebebi kovaryans matrisinin kare matris olmasıdır. İşlem gürültüsü robot konum tahminindeki belirsizliği değiştireceği için kovaryans matrisine eklenir. Gürültü matrisi değeri ne kadar küçükse robot hareketi boyunca tahmin edilen konum gerçek konuma o kadar yakın olacaktır.

2. ve 3. adımlarda yapılan tahminler, 4. 5. ve 6. adımlarda ölçümler kullanılarak güncellenir ve robotun konumuna ait son durum ortalaması ve kovaryans ifadeleri elde edilir. Bu kısma doğrulama veya güncelleme adımı denir. 4. adımda K_k olarak geçen ifade Kalman kazancı olarak adlandırılır. Kalman Kazancı ile yapılan ölçümlerin, yapılacak son durum tahminini ne derece etkilediği görülebilir. Buradaki ölçüm gürültü kovaryansı, Q_k , Kalman kazancın oluşumunda önemli bir rol oynar. Ölçümlerin kusursuz bir şekilde alındığı düşünülürse bir önceki adımda tahmin edilen ortalama değerinin bir önemi kalmaz ve son tahmin değeri

ölçümlere eşit olur. Aynı şekilde R_k 'nin çok büyük olduğu durumda ölçümlerin bir önemi olmayacak ve son durum tahmini durum tahmin adımında elde edilen ortalama değere eşit olacaktır. 5. adımda Kalman kazancı ve ölçümler kullanılarak ortalama güncellenir. Son olarak 6. adımda yeni kovaryans matrisi hesaplanır.

3.2. Genişletilmiş Kalman Filtresi

Kalman Filtresinde tahmin ve güncelleme yapılırken durum ve ölçüm ifadelerinin doğrusal olduğu varsayılmıştır. Fakat gerçek hayat uygulamalarında bu ifadelerin doğrusal olma ihtimali oldukça düşüktür. Bir robotun hareketinin incelendiği düşünülürse robot, yer değiştirme ve dönme hızına sahip olacaktır. Robot, sonraki adımı tanımlanırken doğrusal olarak gösterilemeyen dairesel bir yörüngede olacak. Bu durum, basit robotik problemleri hariç, Kalman Filtresinin en önemli varsayımını yıkmış ve filtreyi uygulanamaz hale getirmiştir [10].

Şekil 3.1'de bir Gauss rastgele değişkeninin doğrusal olmayan fonksiyon içindeki değişimi görülmektedir. Şekil 3.1.(c)'de grafik Gauss dağılımına sahip $p(x)$ fonksiyonunun değişimini göstermektedir. Daha sonra bu dağılım Şekil 3.1.(b)'deki doğrusal olmayan $g(x)$ fonksiyonundan geçtiğinde Şekil 3.1.(a)'da görüldüğü gibi bu fonksiyon Gauss olmaktan uzaklaşmıştır. Bu da Kalman Filtresinin doğrusal olmayan fonksiyon ile kullanılamayacağını göstermektedir.

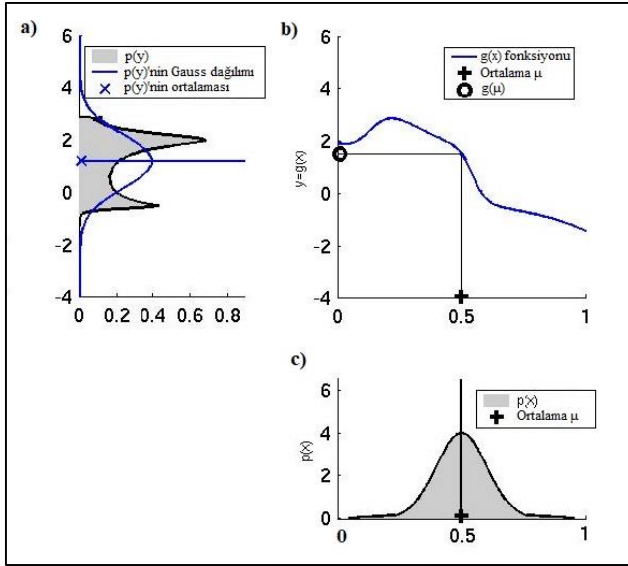
Kalman Filtresinin doğrusal sistemler dışında uygulanabilir hale gelmesi Genişletilmiş Kalman Filtresi (GKF) ile mümkündür [42]. GKF'de k anındaki durum vektörü x_k ve gözlem vektörü z_k aşağıdaki gibi verilmiştir.

$$x_k = g(u_k, x_{k-1}) + \varepsilon_k \quad (3.4)$$

$$z_k = h(x_k) + \delta_k \quad (3.5)$$

KF'de Eş. 3.2 ve 3.3 ile gösterilen doğrusal Gauss modeli, GKF ile Eş. 3.4 ve 3.5'deki gibi doğrusal olmayan formda yazılabilir. Burada $g(.)$ fonksiyonu Eş. 3.2'deki A_k ve B_k matrislerinin yerini alırken $h(.)$ fonksiyonu ise Eş. 3.3'deki C_k matrisinin yerini alır. Bu yapılan işlem sonucunda rastgele seçilen $g(.)$ ve $h(.)$ fonksiyonları ile elde edilen yeni

tahmin olasılığı Gauss dağılımına sahip olmayacaktır. Bayes filtresinin kapalı form çözümü olmadığından doğrusal olmayan $g(\cdot)$ ve $h(\cdot)$ fonksiyonları ile durum tahmini ve güncellemesi yapmak imkânsızdır.



Şekil 3.1. Gauss rastgele değişkeninin doğrusal olmayan fonksiyon içindeki değişimi. (a) Gauss olmayan sonuç dağılımı (b) doğrusal olmayan $g(x)$ fonksiyonu (c) Gauss dağılımına sahip olan $p(x)$ olasılık dağılımı [4]

Kalman Filtresinin doğrusal sistemler dışında uygulanabilir hale gelmesi Genişletilmiş Kalman Filtresi (GKF) ile mümkündür [42]. GKF’de k anındaki durum vektörü x_k ve gözlem vektörü z_k aşağıdaki gibi verilmiştir.

$$x_k = g(u_k, x_{k-1}) + \varepsilon_k \quad (3.4)$$

$$z_k = h(x_k) + \delta_k \quad (3.5)$$

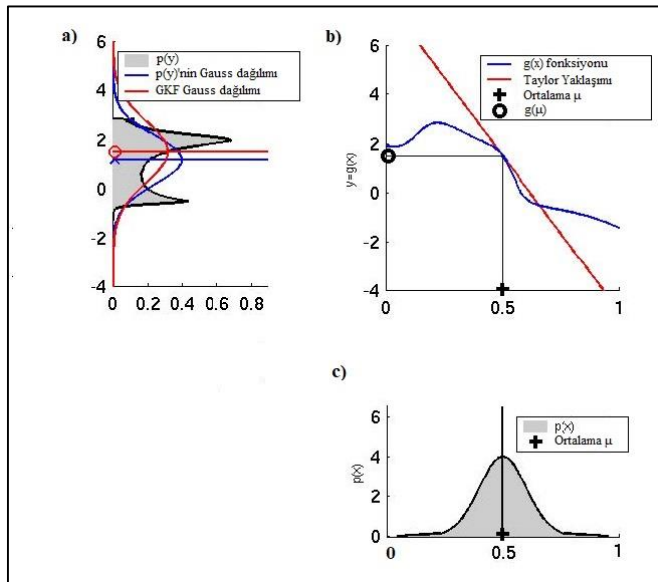
KF’de Eş. 3.2 ve 3.3 ile gösterilen doğrusal Gauss modeli, GKF ile Eş. 3.4 ve 3.5’deki gibi doğrusal olmayan formda yazılabilir. Burada $g(\cdot)$ fonksiyonu Eş. 3.2’deki A_k ve B_k matrislerinin yerini alırken $h(\cdot)$ fonksiyonu ise Eş. 3.3’deki C_k matrisinin yerini alır. Bu yapılan işlem sonucunda rastgele seçilen $g(\cdot)$ ve $h(\cdot)$ fonksiyonları ile elde edilen yeni tahmin olasılığı Gauss dağılımına sahip olmayacaktır. Bayes filtresinin kapalı form çözümü olmadığından doğrusal olmayan $g(\cdot)$ ve $h(\cdot)$ fonksiyonları ile durum tahmini ve güncellemesi yapmak imkânsızdır.

Genişletilmiş Kalman Filtresi (GKF) son durum tahminini yaklaşık bir değer ile ifade eder

ve bu değer Gauss dağılımına sahiptir. Durum tahmini ve güncellemesi Kalman filtresinde olduğu gibi yapılır fakat Kalman filtresinin aksine kesin sonuç vermez, yaklaşık bir sonuç bulunur. GKF'nin amacı bir tahmin dağılımının ortalamasını ve kovaryansını verimli şekilde tahmin etmektir [4].

3.2.1. Taylor Açılımı ile doğrusallaştırma işlemi

GKF'nin KF'ye göre daha yaygın kullanılmasının sebebi doğrusallaştırma yapmasıdır. Durum fonksiyonu $g(x)$ 'in Şekildeki gibi doğrusal olmayan bir fonksiyon olduğu varsayıldığında, bu fonksiyon kullanılarak elde edilen ve Şekil 3.2.a'da görülen yeni tahmin fonksiyonunun dağılımı Gauss olmamaktadır. $g(x)$ fonksiyonu $p(x)$ 'in Gauss şeklini bozmaktadır. Bu sebeple $g(x)$ fonksiyonunun doğrusallaştırılması gerekmektedir. Bu doğrusallaştırma işlemi Şekil 3.2.b'de gösterildiği gibi, $p(x)$ 'in ortalama değerinin $g(x)$ 'de karşılık geldiği noktadan teğet bir doğru çizilmesi ile yapılır. Bu doğru $g(x)$ 'in doğrusallaştırılmış fonksiyonunu verir. Doğrusallaştırma işlemi sayesinde Gauss olarak tanımlanan $p(x)$ 'in yeni tahmin fonksiyonunda da Gauss kalması sağlanır. $g(x)$ fonksiyonu bu şekilde doğrusal hale getirildiğinde tahmin olasılık dağılımının hesaplanması Kalman Filtresi ile aynı olur.



Şekil 3.2. Gauss rastgele değişkeninin Taylor yaklaşımı ile yaklaşık doğrusallaştırılmış fonksiyon içindeki(kırmızı) değişimi. (a) olasılık dağılımı yaklaşık doğrusallaştırılmış fonksiyondan geçtikten sonra korunan Gauss dağılımı(kırmızı) (b) doğrusal olmayan $g(x)$ fonksiyonunun Taylor yaklaşımı (kırmızı) (c) Gauss dağılımına sahip olan $p(x)$ dağılımı [4]

GKF ile gözlem işleminde tanımlanan doğrusal olmayan h fonksiyonu da aynı işlemler yapılarak doğrusal hale getirilir.

Doğrusal olmayan bir fonksiyonu doğrusallaştırmanın birden çok yolu vardır. GKF’de bu yöntemlerden birisi olan birinci dereceden Taylor açılımı yöntemi kullanılır. Taylor açılımı, fonksiyonun değerinden ve eğiminden aynı fonksiyonun doğrusal haline bir yaklaşım yapılmasını önerir [43]. Eğim aşağıda olduğu gibi kısmi türev ile hesaplanır.

$$g'(u_k, x_{k-1}) = \frac{\partial g(u_k, x_{k-1})}{\partial x_{k-1}} \quad (3.6)$$

Eş 3.6’da görüleceği üzere g fonksiyonunun hem değeri hem de eğimi g ’nin genliğine bağlıdır. Doğrusallaştırma sırasında genliği seçerken g fonksiyonunun en olası durumunu seçmek en mantıklı seçim olacaktır. Gauss şeklini de korumak için burada yapılacak seçim önceki durumun ortalaması olan değer, μ_{k-1} , olacaktır. Bu değerde türev alındıktan sonra elde edilen doğrusal fonksiyon ile olasılık dağılım fonksiyonunun (ODF) işleme girmesi sonucu yine Gauss dağılımına sahip yeni yaklaşık durum tahmini aşağıdaki gibi elde edilir.

$$\begin{aligned} g(u_k, x_{k-1}) &\approx g(u_k, \mu_{k-1}) + g'(u_k, \mu_{k-1})(x_{k-1} - \mu_{k-1}) \\ &= g(u_k, \mu_{k-1}) + G_k(x_{k-1} - \mu_{k-1}) \end{aligned} \quad (3.7)$$

Burada G_k , bir kare matristir ve genellikle jakobiyen matrisi olarak adlandırılır. Jakobiyenin değeri u_k ve μ_{k-1} ’e bağlı olduğu için zaman içinde farklı noktalar için farklı değerler alır.

GKF aynı doğrusallaştırma işlemini ölçüm fonksiyonu h için de uygular. Bu doğrusallaştırma yapılırken Taylor açılımı, robotun olasılık dağılımının en büyük olduğunu kabul ettiği $\bar{\mu}_k$ etrafında gerçekleştirilir.

$$\begin{aligned} h(x_k) &\approx h(\bar{\mu}_k) + h'(\bar{\mu}_k)(x_k - \bar{\mu}_k) \\ &= h(\bar{\mu}_k) + H_k(x_k - \bar{\mu}_k) \end{aligned} \quad (3.8)$$

Burada H_k , Eş. 3.8’de yer alan G_k ’ye benzer şekilde jakobiyen matrisi olarak adlandırılır. G_k ’de olduğu gibi H_k de her bir zaman aralığında farklı değerler alacaktır.

3.2.2. GKF algoritması

Genişletilmiş Kalman Filtresinin algoritmasının en sade hali [4] Çizelge 3.2’de verilmiştir.

Çizelge 3.2. Genişletilmiş Kalman Filtresi Algoritması [4]

1	Genişletilmiş Kalman Filtresi Algoritması
2	$\bar{\mu}_k = g(u_k, \mu_{k-1}) + q_{k-1}$
3	$\bar{\Sigma}_k = G_k \Sigma_{k-1} G_k^T + Q_k$
4	$\bar{z}_k = h(\bar{\mu}_k) + r_k$
5	$e_{z,k} = z_k - \bar{z}_k$
6	$S_k = H_k \bar{\Sigma}_k H_k^T + R_k$
7	$K_k = \bar{\Sigma}_k H_k^T S_k^{-1}$
8	$\mu_k = \bar{\mu}_k + K_k(e_{z,k})$
9	$\Sigma_k = (I - K_k H_k) \bar{\Sigma}_k$
10	return μ_k, Σ_k

Çizelgede verilen algoritmada 2. ve 3. adımlar Kalman filtresinde yer alan A_k ve B_k doğrusal matrisler yerine $g(.)$ doğrusal olmayan fonksiyonu ve G_k jacobian matrisi kullanılmıştır. 2 ve 3. adımlar tahmin adımını oluşturur. Bu adımlarda q_{k-1} işlem gürültüsünün ortalaması ve Q_k işlem gürültüsünün kovaryansını gösterir. Q_k ’ye temelde hareket komutu u_k sebep olur. Temelde bu iki adım ile yaklaşık ortalama tahmini yapılır ve buna bağlı olarak kovaryans içinde değişiklikler gerçekleştirilir.

4-9. adımlarda önceki adımlarda elde edilen durum tahmininin düzeltme işlemleri yapılır. Doğrusal olmayan gözlem fonksiyonu, $h(.)$, doğrusallaştırılır ve yaklaşık gözlem tahminleri elde edilir. Bu işlem Kalman filtresinde yer alan $C_k \bar{\mu}_k$ işleminin yerini alır. $h(.)$ fonksiyonunun jacobianı H_k düzeltme adımındaki C_k doğrusal matrisinin yerini alır. 6. adım da tahmin edilen gözlem ile gerçek konum arasındaki fark, $e_{z,k}$, bulunur. Bu sayede Çizelge 3.1’in 6. adımındaki işlem için ön hazırlık yapılmış olur. Daha sonra 7 ve 8 adımlarda kalman kazancı hesaplanır. Bu kazanç ve $e_{z,k}$ kullanılarak düzeltilmiş yeni tahmin değerleri bulunur.

Sensörden gelen bilgi gürültüsü (r_k ve R_k) yükseldikçe algoritma kendi durum tahminine ağırlık vererek yeni tahmini oluşturur. İşlem gürültüsü (q_k ve Q_k) büyük olduğunda ise sensörden gelen ölçüm bilgisine ağırlık vererek yeni tahmini oluşturur.

Özetle, GKF, KF içindeki doğrusal tahminlerin doğrusal olmayanlar ile yer değiştirdiği halidir. GKF’de, KF içindeki doğrusal sistem matrisleri A_k , B_k ve C_k yerine G_k ve H_k jakobiye matrisleri kullanılır. KF durum tahminlerinde kesin değerler bulurken GKF, tahminlerini yaklaşık olarak hesaplar.

Genel olarak robotik problemlerinin hepsi doğrusal olmayan fonksiyonlar içerir. GKF’nin kullandığı yaklaşımın tutarlı ve kaliteli sonuçlar vermesi iki temel faktöre bağlıdır. Bunların ilki doğrusallaştırılan fonksiyonun doğrusal olmama derecesidir. Eğer fonksiyonlar yaklaşık olarak doğrusala yakın ise GKF yeteri kadar tutarlı ve kaliteli sonuçlar verebilir. İkinci olarak da belirsizlik seviyesinin derecesi GKF yaklaşımı için önemlidir [10]. Bunun için durum tahminlerindeki belirsizlik, GKF’nin iyi çalışması ve fonksiyonların doğrusal olmaması durumundan kötü etkilenmemesi için olabildiği kadar küçük ve sabit tutulur [4].

3.3. Adaptif Genişletilmiş Kalman Filtresi (AGKF)

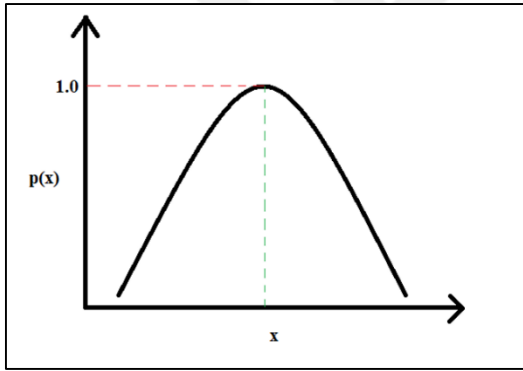
GKF, uygulaması kolay ve kaliteli sonuçlar veren bir filtre olduğu için SLAM uygulamasında oldukça yaygın olarak kullanılmaktadır. Tahmin hatasını azaltmaya yönelik birçok çalışma yapılmıştır [30], [31]. Fakat bu yaklaşımlar daha çok benzetim çalışmalarında etkili olarak kullanılabilmiştir. Çünkü GKF uygulamalarının verimli olması için doğrusal olmayan fonksiyonların doğrusal olmama seviyesinin oldukça düşük olması ve gürültü istatistiklerinin sabit ve düşük değerlere sahip olması gerekir. Bu iki koşul gerçek hayat uygulamalarının çoğunda sağlanabilecek durumlar değildir. Özellikle gürültü istatistikleri, sabit olmayan ve sürekli artan bir karakteristiğe sahiptir. Bu yüzden tanımlanan sistem modeli zamanla doğru olmayan sonuçlar vermeye başlayacaktır. Gürültü istatistikleri bu yüzden sistemin yapısına göre zamanla değişmelidir [12]. Bu sorunu çözmek için AGKF yaklaşımı önerilmiştir [44]. Bu yaklaşımın amacı işlem ve ölçüm gürültü istatistiklerinin olasılık dağılımlarının ortalama ve kovaryans değerlerini yaklaşık olarak tahmin etmektir. Bu tahmin işlemi optimum MOT-BM çözümü ile gerçekleştirilmektedir. Bu çözümü gerçekleştirmek için gerekli bazı değerler klasik GKF sürecinde hesaplanamamaktadır. Bu sorunu çözmek için de tek adımlı düzleme metodu kullanılarak klasik GKF üzerinde bazı geliştirmeler yapılmıştır [10]. MOT-BM kullanılarak yapılan bu tekrarlamalı gürültü istatistiği tahmini tarafsız tahmine göre yapılmıştır. Fakat bu işlemler sonucu elde edilecek gürültü istatistiklerinin kovaryans matrisleri negatif tanımlı matrisleri gibi istenmeyen karakteristikler gösterebilir. Bunun için matrislerin pozitif tanımlı olmasını garantileyecek

ICE metodu uygulanmıştır [12].

Özetle AGKF, GKF içinde sabit kabul edilen gürültü terimlerinin tahminini yapmak ve gerçek hayata daha rahat uyum sağlayabilecek bir çözüm sunmak için önerilmiştir. GKF'nin AGKF'ye dönüşüm süreci aşağıdaki başlıklarda anlatılmıştır.

3.3.1. Maksimum olabilirlik tahmini (MOT)

Bir dizi parametre değeri verildiğinde, bu değerlere karşılık gelen olasılık dağılım fonksiyonu Şekil 3.3'te verilen eğriye benzer bir değişim gösterir. Şekilde de görüldüğü gibi bazı verilerin dağılım içinde bulunma olasılığı diğerlerine göre daha fazladır. Burada dağılım üzerinden veriler ile ilgili yorum yapılabilir [45].



Şekil 3.3. Standart bir Gauss olasılık dağılımı

Fakat veri üzerinden bu yorum yapılmak istenirse ters bir işleme ihtiyaç duyulur. Bunun için gözlenen veriler ve ilgilenilen model içinde verileri üretme olasılığı en yüksek olan ve olabilirlik fonksiyonu olarak tanımlanan olasılık dağılım fonksiyonu bulunur. Olabilirlik fonksiyonu olasılık dağılım fonksiyonunun tam tersi işlem yapar. Yani parametre değerleri üzerinden veriye değil, veri üzerinden parametre değerlerine ulaşır [45]. Örnek gösterim aşağıdaki gibidir.

$$L(\mu, \Sigma|x) = p(x|\mu, \Sigma) \quad (3.9)$$

Burada x veri vektörünü, μ ve Σ ise x veri vektörünün ortalama ve kovaryans parametrelerini gösterir.

GKF olasılık dağılım fonksiyonu $p(x_k, z_k | q, r, Q, R)$ olarak yazılır. Buradaki k ayrık zaman, q, r, Q, R ise gürültü istatistikleridir. $x_k = [x_1, \dots, x_k]$ ve $z_k = [z_1, \dots, z_k]$ şeklinde gösterilen durum ve gözlem vektörleridir. Bu ODF'ye ait olabilirlik fonksiyonu aşağıdaki gibi yazılır.

$$L(q, r, Q, R | x_k, z_k) = p(x_k, z_k | q, r, Q, R) \quad (3.10)$$

Eş. 3.10 Bayes kuralına göre ayrıldığında denklem aşağıdaki gibi olur.

$$L(q, r, Q, R | x_k, z_k) = p(x_k | q, r, Q, R) p(z_k | x_k, q, r, Q, R) \quad (3.11)$$

Sürecin gelecekteki durumlarının olasılık dağılımları eğer sadece mevcut duruma bağlı ise Markov özelliğine sahiptir [46]. Buna göre gelecekteki durum sadece mevcut duruma bağlıdır ve geçmişteki durumlar gelecekteki durumu etkilemez. Geçmişteki durumlardan koşullu bağımsız olduğu varsayılır. Markov özelliği bir yaklaşımdır. Bu yaklaşım sayesinde Eş. 3.11'deki işlem sıralı çarpım haline getirilir. Ayrıca x_k tahmininde Eş. 3.2-3.3'te gösterildiği gibi sadece q ve Q etkili iken z_k tahmininde de r ve R etkilidir. Gerekli işlemler yapıldığında ODF aşağıdaki şekli alacaktır.

$$p(z_k, x_k | \theta) = p[x_o] \prod_{i=1}^k p[x_i | x_{i-1}, q, Q] \prod_{i=1}^k p[z_i | x_i, r, R] \quad (3.12)$$

Burada $p[x_o]$ sistemin ön (prior) bilgisini gösterir. Gürültü istatistikleri $\theta = (q, r, Q, R)$ şeklinde gösterilmiştir.

MOT'a göre elde edilmek istenen olasılık dağılımı, gözlemlenen verileri en olası hale getiren bir dağılım içinde olmalıdır. Bu L olabilirlik fonksiyonunu maksimize eden parametre değerlerinin bulunması ile mümkündür. Hesaplama kolaylığı için MOT, L 'nin logaritmik fonksiyonunu maksimize ederek bulunur [34]. Logaritma işlemi fonksiyonun yapısında değişiklik yapmadığı gibi tepe değerlerini de değiştirmez.

Gürültü istatistiklerinin, θ , MOT'u aşağıdaki gibi bulunur.

$$\hat{\theta}^{ML} = \operatorname{argmax}_{\theta} \{\ln [L(q, r, Q, R | x_k, z_k)]\} \quad (3.13)$$

Eş. 3.10'a göre olabilirlik fonksiyonu ODF'ye eşit olduğu için Eş. 3.12 aynı zamanda olabilirlik fonksiyonunu gösterir. Ön bilginin normal dağılıma sahip olduğu düşünülürse, olabilirlik fonksiyonu, Eş. 3.1'de yer alan Gauss formunda yazıldığında aşağıdaki gibi olacaktır.

$$L(q, r, Q, R|x_k, z_k) = p(x_k, z_k|q, r, Q, R) = \frac{1}{(2\pi)^{\frac{-k(n+m)+n}{2}}} |P_0|^{-\frac{1}{2}} |Q|^{-\frac{1}{2}} |R|^{-\frac{1}{2}} \exp -\frac{1}{2} \{ [\|x_0 - \hat{x}_0\|_{P_0^{-1}}^2 + \sum_{i=1}^k \|x_i - g(x_{i-1}) - q_{i-1}\|_{Q^{-1}}^2 + \sum_{i=1}^k \|z_i - h(x_i) - r_i\|_{R^{-1}}^2] \} \quad (3.14)$$

Eş. 3.14'ün logaritmik olabilirlik fonksiyonu alındığında eşitlik aşağıdaki gibi ifade edilir.

$$\ln(L) = -\frac{k(n+m)+n}{2} \ln(2\pi) - \frac{1}{2} \ln(|P_0|) - \frac{k}{2} \ln(|R|) - \frac{k}{2} \ln(|Q|) - \frac{1}{2} \|x_0 - \hat{x}_0\|_{P_0^{-1}}^2 - \frac{1}{2} [\sum_{i=1}^k \|x_i - g(x_{i-1}) - q_{i-1}\|_{Q^{-1}}^2 + \sum_{i=1}^k \|z_i - h(x_i) - r_i\|_{R^{-1}}^2] \quad (3.15)$$

Eş. 3.15'de $\ln(L)$ ile gösterilen kısım Eş. 3.14'teki $L(q, r, Q, R|x_k, z_k)$ olabilirlik fonksiyonunun logaritmik değerini gösterir. Burada bulunan gürültü istatistikleri ile bağlantısı olmayan değerleri C ile gösterilirse Eş. 3.15'i aşağıdaki gibi yazılır.

$$C = -\frac{k(n+m)+n}{2} \ln(2\pi) - \frac{1}{2} \ln(|P_0|) - \frac{1}{2} \|x_0 - \hat{x}_0\|_{P_0^{-1}}^2 \quad (3.16)$$

$$\ln(L) = c - \frac{k}{2} \ln(|R|) - \frac{k}{2} \ln(|Q|) - \frac{1}{2} [\sum_{i=1}^k \|x_i - g(x_{i-1}) - q_{i-1}\|_{Q^{-1}}^2 + \sum_{i=1}^k \|z_i - h(x_i) - r_i\|_{R^{-1}}^2] \quad (3.17)$$

3.3.2. Beklenti-Maksimizasyonu (BM)

Beklenti-Maksimizasyonu (BM) algoritması verilerin eksik olduğu veya olabilirlik fonksiyonunun gizli değişkenleri barındırdığı MOT için kullanılan bir algoritmadır [47]. Buradaki eksik veri ve gizli değişken ifadeleri birbirleri ile ilişkilidir. Bir gizli değişken olduğunda gözlemlenen verilerin eksik olduğu, veri eksik olduğunda da gizli değişkenler olduğu düşünülebilir. BM sırasıyla beklenti adımı (B-Adımı) ve maksimizasyon adımı (M-Adımı) olmak üzere iki adım arasında yinelemeli olarak gerçekleştirilir [48]. B-Adımında

mevcut modelde verilen eksik verilerin tamamlanması üzerine bir olasılık dağılımı tahmini yapılır. Ardından M-Adımında bu dağılım üzerinden logaritmik olabilirlik olasılığının maksimizasyonu yapılır. BM özellikle gözlemlenen verilerin logaritmik olasılığını maksimize eden parametreleri bulmaya çalışır.

Eş. 3.17 ile elde edilmiş olan MOT denkleminin çözümü BM ile gerçekleştirilebilir. BM yaklaşımı ile tahmin etmek istediğimiz gürültü istatistik değerleri yaklaşık olarak bulunabilir. Bu denklemin çözüm adımları aşağıdaki başlıklarda verilmiştir.

B-Adımı

MOT denkleminin beklentisi, B, aşağıdaki gibi hesaplanır.

$$J = B[\ln L(q, r, Q, R|x_k, z_k)] = C - \frac{k}{2} \ln(|R|) - \frac{k}{2} \ln(|Q|) - \frac{1}{2} B \left[\sum_{i=1}^k \|x_i - g(x_{i-1}) - q_{i-1}\|_{Q^{-1}}^2 + \sum_{i=1}^k \|z_i - h(x_i) - r_i\|_{R^{-1}}^2 \right] \quad (3.18)$$

Matris izi (trace of matrix) bir kare matrisin esas köşegenindeki elemanların toplamını ifade eder ve tr ile gösterilir. Aynı zamanda özdeğerlerinin toplamıdır. Matris izi sayesinde bir lineer işlemin izi tanımlanabilir. Kullback-Leibler'in iki adet çok değişkenli Gauss dağılımları arasındaki sapma eşitliğine göre, Gauss dağılımlarının beklenti değeri matris izlerine eşittir [49]. Bu eşitlik aşağıdaki gibi gösterilir.

$$B[(x - \mu)^T \Sigma^{-1} (x - \mu)] = tr(B[\Sigma^{-1} (x - \mu)(x - \mu)^T]) \quad (3.19)$$

Tanımı gereği $\|a\|_{b^{-1}}^2 = a^T b^{-1} a$ olduğundan Eş. 3.18, Eş.3.19 yardımı ile aşağıdaki şekilde düzenlenir.

$$J = B[\ln L(q, r, Q, R|x_k, z_k)] = C - \frac{k}{2} \ln(|R|) - \frac{k}{2} \ln(|Q|) - \frac{1}{2} \sum_{i=1}^k B \{ tr[Q^{-1} (x_i - f(x_{i-1}) - q_{i-1})(x_i - f(x_{i-1}) - q_{i-1})^T] \} - \frac{1}{2} \sum_{i=1}^k B \{ tr[R^{-1} (z_i - h(x_i) - r_i)(z_i - h(x_i) - r_i)^T] \} \quad (3.20)$$

M-Adımı

M-Adımında J'nin maksimum değerleri bulunarak gürültü istatistikleri elde edilir. Maksimum değerler, J'nin gürültü istatistikleri değerlerine göre (q, r, Q, R) , parçalı türevlerinin alınıp 0'a eşitlenmesi ile bulunur.

$$\frac{\partial J}{\partial q} = 0, \frac{\partial J}{\partial r} = 0, \frac{\partial J}{\partial Q} = 0, \frac{\partial J}{\partial R} = 0 \quad (3.21)$$

R ve Q matrisleri pozitif tanımlı olduğu için bu iki matrisin determinantları, matris izlerine eşit olur.

$$\begin{cases} |Q| = \text{tr}(Q) \\ |R| = \text{tr}(R) \end{cases} \quad (3.22)$$

J ifadesinin türevi alınıp sıfıra eşitlendiğinde gürültü istatistikleri için eşitlikler aşağıdaki gibi olur.

$$\hat{q}_k = \frac{1}{k} \sum_{i=1}^k x_{i|k} - f(x_{i|k}) \quad (3.23)$$

$$\hat{r}_k = \frac{1}{k} \sum_{i=1}^k z_{i|k} - h(x_{i|k}) \quad (3.24)$$

$$\hat{Q}_k = \frac{1}{k} \sum_{i=1}^k (x_{i|k} - f(x_{i-1|k}) - q)(x_{i|k} - f(x_{i-1|k}) - q)^T \quad (3.25)$$

$$\hat{R}_k = \frac{1}{k} \sum_{i=1}^k (z_{i|k} - h(x_{i|k}) - r)(z_{i|k} - h(x_{i|k}) - r)^T \quad (3.26)$$

Eş. 3.23-3.26'de $x_{i|k}$ ve $x_{i-1|k}$ karmaşık terimlerin sadeleşmesi ile aşağıdaki eşitlikler elde edilir.

$$\hat{q}_k = \frac{1}{k} \sum_{i=1}^k x_{i|i} - f(x_{i-1|i}) \quad (3.27)$$

$$\hat{r}_k = \frac{1}{k} \sum_{i=1}^k z_{i|i} - h(x_{i|i}) \quad (3.28)$$

$$\hat{Q}_k = \frac{1}{k} \sum_{i=1}^k (x_{i|i} - f(x_{i-1|i}) - q)(x_{i|i} - f(x_{i-1|i}) - q)^T \quad (3.29)$$

$$\hat{R}_k = \frac{1}{k} \sum_{i=1}^k (z_{i|i} - h(x_{i|i}) - r)(z_{i|i} - h(x_{i|i}) - r)^T \quad (3.30)$$

3.3.3. Tek adımlı ve sabit noktalı düzleme algoritması ile GKF'nin geliştirilmesi

Birçok durum-uzay uygulamasında, bir veri dizisi $p(x_k|y_{1:n})$ 'in belirli bir zaman aralığında olasılık dağılımı hesaplanır. Eğer $k > n$ olursa tahmin, $k = n$ olursa filtreleme ve $k < n$ ise düzleme işlemi yapılıyor demektir. Düzleme işlemi de kendi içinde 3 farklı şekilde yapılır. Tüm zaman aralıklarının dahil olduğu, $k = 1, \dots, K$ aralığında $p(x_k|y_{1:K})$ hesaplanırken sabit-aralıklı düzleme, $L > 0$ durumunda belirli bir zaman aralığına kadar $p(x_k|y_{1:k+L})$ hesaplanırken sabit gecikmeli düzleme ve $D > k$ durumunda sabit bir k değerinde $p(x_k|y_{1:D})$ hesaplanırken sabit noktalı düzleme kullanılır [50,51].

Eş. 3.23-3.26 arasındaki $x_{i-1|i}$ ifadesi klasik GKF tarafından bulunmaz. Bu ifade bir düzleme ifadesidir. Hesaplanacak hedef dağılım, sabit bir değer için hesaplanacağından sabit noktalı düzleme kullanılmalıdır. $x_{i-1|i}$ ifadesinin hesaplanması gürültü istatistiklerinin hesaplanmasında kilit bir role sahiptir.

Geliştirilmiş GKF iki ana adımdan oluşur. İlk adımda, tek adımlı düzleme kazancını hesaplamak için sabit noktalı düzleme algoritması ile $k - 1$ zamanındaki tahmin matrisini, daha sonra k zamanındaki değişimi (innovation) hesaplar ve $x_{i-1|i-1}$ tahmin değerini tek adımlı düzleme değeri $x_{i-1|i}$ ile değiştirir. Son olarak hem tahmin değeri hem de tek adımlı düzleme değeri geliştirilmiş GKF içinde tekrarlı olarak üretilir.

Geliştirilmiş GKF ile tek adımlı düzleme değeri $x_{i-1|i}$ aşağıdaki gibi bulunur.

$$\tilde{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k) + q_{k-1} \quad (3.31)$$

$$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q_{k-1} \quad (3.32)$$

$$\tilde{z}_{k|k-1} = h(\tilde{x}_{k|k-1}) + r_k \quad (3.33)$$

$$\check{e}_{z,k|k-1} = z_k - \tilde{z}_{k|k-1} \quad (3.34)$$

$$\check{S}_k = \check{H}P_{k|k-1}\check{H}^T + R_k \quad (3.35)$$

$$\check{K}_k = P_{k|k-1}\check{H}^T(\check{S}_k)^{-1} \quad (3.36)$$

$$\hat{x}_{k-1|k} = \hat{x}_{k-1|k-1} + \check{K}_k\check{e}_{z,k|k-1} \quad (3.37)$$

Burada $\hat{x}_{k-1|k-1}$ ön değeri, $\check{x}_{k|k-1}$ ön tahmin değerini, $\check{e}_{z,k|k-1}$ değişimi, $\check{K}_k$ kazancı ve $\hat{x}_{k-1|k}$ tek adımlı düzleme değerini gösterir.

Gürültü istatistiklerini, özellikle q_k ve Q_k değerlerini, tahmin etmek için $x_{k|k}$ ve $\hat{x}_{k-1|k}$ arasındaki bağıntıyı kurmak önemlidir. Bu bağıntı aynı zamanda gürültü istatistiklerinin kapalı döngü olarak tahmin edilmesinde de büyük bir öneme sahiptir. Klasik GKF içinde Eş. 3.30'daki $\hat{x}_{k-1|k-1}$ ön değeri ile $\hat{x}_{k-1|k}$ tek adımlı düzleme değerinin yer değiştirmesi ile elde edilir. Bu şekilde, geliştirilmiş GKF aşağıdaki gibi yazılır.

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k) + q_{k-1} \quad (3.38)$$

$$\hat{z}_{k|k-1} = h(\hat{x}_{k|k-1}) + r_k \quad (3.39)$$

$$e_{z,k|k-1} = z_k - \hat{z}_{k|k-1} \quad (3.40)$$

$$S_k = HP_{k|k-1}H^T + R_k \quad (3.41)$$

$$K_k = P_{k|k-1}H^T(S_k)^{-1} \quad (3.42)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_{z,k|k-1} \quad (3.43)$$

$$P_{k|k} = (I - K_k H)P_{k|k-1} \quad (3.44)$$

Burada $\hat{x}_{k|k}$ geliştirilmiş GKF'nin son tahmin değerini, $P_{k|k}$ bu tahminin kovaryans matrisini gösterir.

3.3.4. Tekrarlı ve zamanla değişen gürültü istatistikleri formu

Gürültü istatistiklerinin tekrarlı olarak hesaplanması için Eş. 3.31 - 3.44'te bulunan ifadeler Eş. 3.27 - 3.30 içine eklenmelidir. Fakat bu işlemler sonucunda optimum tahmin kalitesi düşebilir. Bu olasılığı azaltmak için unbiased(tarafsız) tahmin konsepti uygulanmalıdır [10].

Bu işlem için önce Eş. 3.38 ile Eş. 3.43 ile toplanır. Bu toplam sonucunda Eş. 3.27 ve 3.29'deki $x_{i|i} - f(x_{i-1|i})$ ifadesi aşağıdaki gibi yazılır.

$$x_{i|i} - f(x_{i-1|i}) = K_k e_{z,i} + q_{i-1} \quad (3.45)$$

Buradaki $e_{z,i}$ terimi $e_{z,i|i-1}$ 'e karşılık gelmektedir. Daha sonra Eş. 3.28 ve 3.30'daki $z_{i|i} - h(x_{i|i})$ ifadesi için Eş. 3.42'deki $x_{i|i}$ ifadesi yerine yazılır ve genel ifade aşağıdaki gibi olur.

$$z_{i|i} - h(x_{i|i}) = z_i - h(x_{i|i-1} + K_k e_{z,i}) \quad (3.46)$$

Eş. 3.45 ve 3.46'yı Eş. 3.27-3.30'deki MOT-BM'ye uygulanırsa yeni gürültü istatistiği eşitlikleri aşağıdaki gibi elde edilir.

$$\hat{q}_k = \frac{1}{k} \sum_{i=1}^k K_i e_{z,i} + q_{i-1} \quad (3.47)$$

$$\hat{Q}_k = \frac{1}{k} \sum_{i=1}^k K_i e_{z,i} e_{z,i}^T K_i^T \quad (3.48)$$

$$\hat{r}_k = \frac{1}{k} \sum_{i=1}^k (I - HK_i) e_{z,i} + r_i \quad (3.49)$$

$$\hat{R}_k = \frac{1}{k} \sum_{i=1}^k (I - HK_i) e_{z,i} e_{z,i}^T (I - HK_i)^T \quad (3.50)$$

Yukarıdaki eşitliklerde de görüldüğü gibi değişim, $e_{z,i}$ ve onun kovaryansı $e_{z,i} e_{z,i}^T$ gürültü istatistiği tahminin her adımda bulunur. Bu sebeple değişim aşağıdaki gibi yazılır.

$$e_{z,i} = z_i - h(x_{i|i-1}) - r_i = h(x_{i|i-1}) + (v_i - r_i) \quad (3.51)$$

Eş. 3.48'deki ifade [10] gösterilen işlemler sonucunda aşağıdaki halini alır.

$$K_i e_{z,i} e_{z,i}^T K_i^T = F_{i-1} P_{i-1|i-1} F_{i-1}^T + Q_{i-1} - P_{i|i} \quad (3.52)$$

Değişim ifadelerinin beklenti değerleri aşağıdaki gibidir.

$$E[e_{z,i}] = 0 \quad (3.53)$$

$$E[e_{z,i} e_{z,i}^T] = H P_{i|i-1} H^T + R_i \quad (3.54)$$

Bu tanımlamalar ve sadeleştirmelerden sonra Eş. 3.47-3.50'nin beklenti değerleri aşağıdaki gibi hesaplanır.

$$E[\hat{q}_k] = q_k \quad (3.55)$$

$$E[\hat{r}_k] = r_k \quad (3.56)$$

$$E[\hat{Q}_k] = Q_k + E \left[\frac{1}{k} \sum_{i=1}^k F_{i-1} P_{i-1|i-1} F_{i-1}^T - P_{i|i} \right] \quad (3.57)$$

$$E[\hat{R}_k] = R_k + \left[\frac{1}{k} \sum_{i=1}^k H K_i P_{i|i-1} H^T - H P_{i|i-1} H^T \right] \quad (3.58)$$

Buradaki q_k, r_k, Q_k ve R_k Eş. 3.47-3.50'daki $\hat{q}_k, \hat{r}_k, \hat{Q}_k$ ve $\hat{R}_k$ 'ya karşılık geldiği için tarafsız yinelemeli gürültü eşitliklerinin özelliklerini taşıyan eşitlikler aşağıdaki gibi olur.

$$\hat{q}_k = \frac{1}{k} \sum_{i=1}^k K_i e_{z,i} + q_{i-1} \quad (3.59)$$

$$\hat{r}_k = \frac{1}{k} \sum_{i=1}^k (I - H K_i) e_{z,i} + r_i \quad (3.60)$$

$$\hat{Q}_k = \frac{1}{k} \sum_{i=1}^k K_i e_{z,i} e_{z,i}^T K_i^T + F_{i-1} P_{i-1|i-1} F_{i-1}^T - P_{i|i} \quad (3.61)$$

$$\hat{R}_k = \frac{1}{k} \sum_{i=1}^k (I - H K_i) [e_{z,i} e_{z,i}^T (I - H K_i)^T + H P_{i|i-1} H^T] \quad (3.62)$$

Buraya kadar yapılan çıkarımların hepsi gürültü istatistiklerinin bilinmeyen sabitler olduğu varsayımı altında yapılmıştır. Fakat gürültü istatistikleri genellikle zamanla değişir hatta hareket ve gözlem sayısı arttıkça değeri artar. Bu istatistikler zamanla değişir olarak kabul edildiğinde son değişimlerin etkisinin ağırlığını artıracak bir ağırlıklı sönümleme metodu önerilmiştir [12]. Eş. 3.59-3.62'de yer alan üstel $\frac{1}{k}$ ifadesi ağırlık katsayısı d_k ile değiştirilerek zamanla değişen gürültü istatistikleri bulunur.

$$\hat{q}_k = \hat{q}_{k-1} + d_k (K_k e_{z,k}) \quad (3.63)$$

$$\hat{r}_k = \hat{r}_{k-1} + d_k [(I - H K_k) e_{z,k}] \quad (3.64)$$

$$\hat{Q}_k = (I - d_k) \hat{Q}_{k-1} + d_k [K_k e_{z,k} e_{z,k}^T K_k^T + F_{k-1} P_{k-1|k-1} F_{k-1}^T - P_{k|k}] \quad (3.65)$$

$$\hat{R}_k = (I - d_k) \hat{R}_{k-1} + d_k (I - H K_k) [e_{z,k} e_{z,k}^T (I - H K_k)^T + H P_{k|k-1} H^T] \quad (3.66)$$

Burada d_k :

$$d_k = \frac{1-b}{1-b^k} \quad (3.67)$$

Burada $b, 0 < b < 1$ sönümleme faktörüdür.

3.3.5. Değişim kovaryansı tahmincisi (DKT)

Eş. 3.54-3.57 arasındaki gürültü istatistikleri eşitlikleri elde edilirken Q ve R kovaryans matrislerinin pozitif tanımlı matrisler oldukları varsayımı yapılmıştır. Eş. 3.65 ve 3.66'da yapılan Q ve R matris tahmini değişim kovaryansı $e_{z,k}e_{z,k}^T$ 'a bağlıdır. Bu tahmin süresi uzadığında ve işlem sayısı arttığında Q ve/veya R matris değerleri, ani ölçüm atlamaları veya uç değerler ile büyüyen matris yapılarını bozabilir ve en başta pozitif tanımlı kabul edilen bu iki kovaryans matris pozitif tanımlı olma özelliğini kaybedebilir. Bu durum filtrenin performansının düşmesine sebep olur. Hatta bu uç değerler filtrenin ıraksamasına bile yol açabilir. Bu olası sorunu çözmek için Değişim Kovaryansı Tahmincisi (DKT) önerilmiştir [12]. DKT, son N -adımında elde edilen değişim kovaryanslarının ortalamalarını alarak değişim kovaryansı tahmini yapmaktadır. Bu sayede Q ve R matrislerinin pozitif tanımlı matrisler olarak kalması sağlanmıştır. DKT aşağıdaki gibi yazılabilir.

$$DKT_k = \frac{1}{N} \sum_{j=0}^N e_{z,k} - j e_{z,j-k}^T = DKT_{k-1} + \frac{1}{N} [e_{z,k}e_{z,k}^T - e_{z,k-N}e_{z,k-N}^T] \quad (3.68)$$

Burada N pencere boyutudur (window size). Pencere boyutu DKT'nin verimli çalışmasında oldukça önemli bir rol oynar. DKT'nin doğruluğu N 'nin ayarlanmasına bağlıdır. Sistemin dinamik özellikleri ve hareket modeli uygun N seçiminde önemlidir. Genellikle bu seçim deneysel ve gözlemsel bilgiler üzerinden yapılır. Örneğin hızlı değişimlere uyum sağlayabilmesi için N 'nin küçük bir değer alması gerekirken sistem dinamikleri yavaş değiştiğinde DKT hesabının kararlılığının sağlanması için N 'nin büyük bir değer alması gerekir.

Eş 3.68'deki DKT eşitliği Eş. 3.63 ve 3.66'deki $e_{z,k}e_{z,k}^T$ yerine yazılırsa Q ve R kovaryans matrisleri aşağıdaki gibi hesaplanır.

$$\hat{Q}_k = (I - d_k)\hat{Q}_{k-1} + d_k[K_k DKT_k K_k^T + F_{k-1}P_{k-1|k-1}F_{k-1}^T - P_{k|k}] \quad (3.69)$$

$$\hat{R}_k = (I - d_k)\hat{R}_{k-1} + d_k(I - HK_k)[DKT_k(I - HK_k)^T + HP_{k|k-1}H^T] \quad (3.70)$$

3.3.5. AGKF algoritması

Adaptif Genişletilmiş Kalman Filtresinin algoritmasının en sade hali Çizelge 3.3'te verilmiştir.

Çizelge 3.3. Adaptif Genişletilmiş Kalman Filtresi Algoritması

1	Adaptif Genişletilmiş Kalman Filtresi Algoritması
2	$\tilde{x}_{k k-1} = f(\hat{x}_{k-1 k-1}, u_k) + q_{k-1}$
3	$P_{k k-1} = F_{k-1}P_{k-1 k-1}F_{k-1}^T + Q_{k-1}$
4	$\tilde{z}_{k k-1} = h(\tilde{x}_{k k-1}) + r_k$
5	$\check{e}_{z,k k-1} = z_k - \tilde{z}_{k k-1}$
6	$\check{S}_k = \check{H}P_{k k-1}\check{H}^T + R_k$
7	$\check{K}_k = P_{k k-1}\check{H}^T(\check{S}_k)^{-1}$
8	$\hat{x}_{k-1 k} = \hat{x}_{k-1 k-1} + \check{K}_k\check{e}_{z,k k-1}$
9	$\hat{x}_{k k-1} = f(\hat{x}_{k-1 k}, u_k) + q_{k-1}$
10	$\hat{z}_{k k-1} = h(\hat{x}_{k k-1}) + r_k$
11	$e_{z,k k-1} = z_k - \hat{z}_{k k-1}$
12	$S_k = HP_{k k-1}H^T + R_k$
13	$K_k = P_{k k-1}H^T(S_k)^{-1}$
14	$\hat{x}_{k k} = \hat{x}_{k k-1} + K_k e_{z,k k-1}$
15	$P_{k k} = (I - K_k H)P_{k k-1}$
16	$DKT_k = DKT_{k-1} + \frac{1}{N}[e_{z,k k-1}e_{z,k k-1}^T - e_{z,k-N}e_{z,k-N}^T]$
17	$\hat{q}_k = \hat{q}_{k-1} + d_k(K_k e_{z,k})$
18	$\hat{Q}_k = (I - d_k)\hat{Q}_{k-1} + d_k[K_k DKT_k K_k^T + F_{k-1}P_{k-1 k-1}F_{k-1}^T - P_{k k}]$
19	$\hat{r}_k = \hat{r}_{k-1} + d_k[(I - HK_k)e_{z,k}]$
20	$\hat{R}_k = (I - d_k)\hat{R}_{k-1} + d_k[(I - HK_k)(DKT_k(I - HK_k)^T + HP_{k k-1}H^T)]$
21	return $x_k, P_k, q_k, Q_k, r_k, R_k$

Çizelgede verilen algoritmada 1-7. adımlar tek adımlı düzleme değeri $\hat{x}_{k-1|k}$ tahmini yapılır. Bu tahmin işlemlerinde önce 1. adımda ön durum $\hat{x}_{k-1|k-1}$ kullanılarak $\tilde{x}_{k|k-1}$ durum tahmini yapılır ve 2. adımda tahmin edilen duruma ait $P_{k|k-1}$ kovaryansı hesaplanır. 1, 2, 3 ve 5. adımlardaki q_{k-1} , Q_{k-1} , r_k ve R_k sırasıyla işlem gürültüsü ortalaması ve kovaryansının ön değerlerini ve ölçüm gürültüsü ortalaması ve kovaryansının ön değerlerini oluşturur. Bu kısımda 1-2. adımlar tahmin, 3-7. adımlar ise güncelleme adımlarıdır.

9-15. adımlarda tek adımlı düzleme değerinin kullanıldığı, son durum tahmini, $\hat{x}_{k|k}$, ve kovaryansının, $P_{k|k}$, hesaplandığı geliştirilmiş GKF kısmı bulunur. Bu tahmin işlemlerinde

önce 9. adımda tek adımlı düzleme değeri ile bir durum tahmini, $\hat{x}_{k|k-1}$, yapılır. 14 ve 15. adımlarda son durum tahmini ve onun kovaryansı hesaplanır. Bu kısımda 8. adım tahmin, 14-15. adımlar ise güncelleme adımlarıdır.

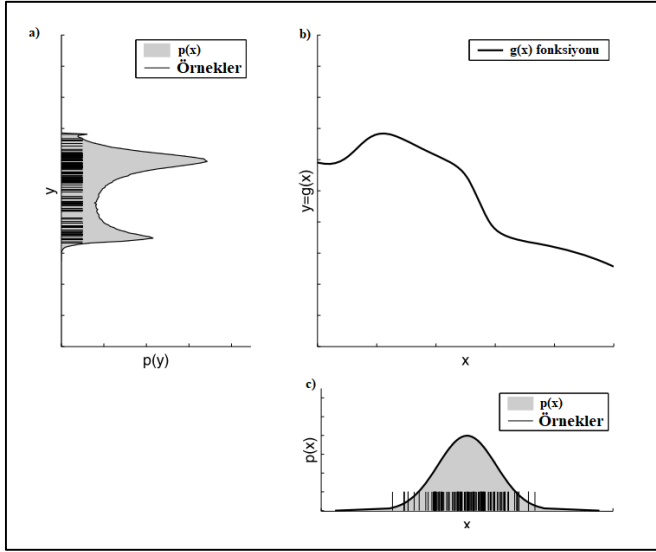
16. adımda gürültü istatistiklerinin pozitif tanımlı matrisler olarak kalmasını sağlayan DKT_k hesaplanır. Bu hesap yapılmadan önce deneysel veriler ışığında pencere boyutu, N , seçilir.

17-20. adımlarda son olarak gürültü istatistikleri tahmin edilir. $\hat{q}_k$ tahmin edilen işlem gürültü ortalamasını, $\hat{Q}_k$ işlem gürültü ortalamasının kovaryans matrisini, $\hat{r}_k$ tahmin edilen ölçüm gürültü ortalamasını ve $\hat{R}_k$ ölçüm gürültü ortalamasının kovaryans matrisini ifade eder. 17-18. adımlar işlem gürültüsü tahmin kısmını ve 19-20. adımlar ölçüm gürültüsü tahmin kısmını oluşturur.

3.4. Parçacık Filtresi

Kalman filtresi ve uzantıları sadece Gauss dağılımları ile modellenenbilmektedir. Parçacık filtresi (PF)'nin amacı ise Gauss dağılımlarının aksine olasılık dağılımını ($p(x)$) yine aynı dağılımdan alınmış rastgele durum örneklemeleri ile ifade etmektir [52]. Dağılımı parametrik formda, normal dağılımın üstel bir fonksiyon ile tanımlamak yerine dağılımdan alınan rastgele örneklemeler ile parametrik olmayan bir formda gösterir. Bu yaklaşık bir gösterim olmakla birlikte parametrik olmadığı için Gauss dağılımlarına kıyasla daha geniş bir dağılım alanını temsil edebilir [4]. PF'nin bir diğer avantajı ise Şekil 3.4'te görüleceği gibi doğrusal olmayan dönüşümleri de modelleyebilmesidir.

Şekil 3.4'te PF içinde kullanılan parçacık gösterimi özetlenmiştir. Gauss rastgele dağılım fonksiyonu, $p(x)$ 'den Şekil 3.4.(c)'de gösterildiği gibi örnekler alınır. Bu örnekler Şekil 3.4.(b)'de yer alan doğrusal olmayan $g(x)$ fonksiyonundan geçerek Şekil 3.4.(a)'daki örnekleri oluşturur. Bu örnekler göre çizilen rastgele fonksiyon Şekil 3.4.(a)'da gösterilmiştir.



Şekil 3.4. PF'de kullanılan parçacık gösterimi. a) Doğrusal olmayan fonksiyon sonucu rastgele değişkenin aldığı form. b) Doğrusal olmayan $g(x)$ fonksiyonu. c) Gauss rastgele değişkeni $p(x)$ ve bu değişkenden alınan örnekler [4]

PF'de olasılık dağılımından alınan örnekler parçacık olarak tanımlanır ve parçacık kümesi X_k aşağıdaki gibi gösterilir.

$$X_k = x_k^{[1]}, x_k^{[2]}, \dots, x_k^{[M]} \quad (3.71)$$

Her parçacık $x_k^{[m]}$, $1 \leq m \leq M$ olmak şartıyla, her k anındaki durumun bir örneklemesidir. Yani her parçacık k zamanında bulunduğu ortamın durumu hakkında bir varsayıma sahiptir. Her biri robotun durumu ile ilgili belli oranda bilgi taşımaktadır. Burada M parçacık sayısını ifade eder ve genellikle oldukça büyük bir sayıdır [53].

Parçacık filtrelerinin amacı X_k parçacık kümesi yardımıyla yaklaşık olarak $p(x_k)$ durumunu tahmin etmektir. Bunun için seçilecek olan parçacık kümesi olasılık dağılımı $p(x_k)$ ile orantılı olmalıdır.

$$x_k^{[m]} \sim p(x_k | z_{1:k}, u_{1:k}) \quad (3.72)$$

Burada $z_{1:k}$ gözlem vektörü, $u_{1:k}$ komut vektörünü ifade eder. Diğer filtrelerde de olduğu gibi PF durum tahmini x_k 'yi bir önceki durum x_{k-1} 'den yinelemeli olarak hesaplar. PF'de

tahmin parçacıklar üzerinden yapıldığı için X_k parçacık kümesi bir önceki durumun parçacık kümesi olan X_{k-1} ’den oluşturulur.

Optimum Bayes çözümü örneklerin ağırlıklı topları ile elde edilir.

$$p(x_{0:k}|z_{1:k}) \approx \sum_{i=1}^M w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (3.73)$$

Burada $\{w_k^i, x_k^i\}_{i=1}^M$ M adet örnek ve ağırlıklarının olduğu kümeyi temsil eder. w_k^i her bir x_k^i örneğinin ağırlığıdır ve ağırlıkların toplamı 1’e eşittir. Yüksek ağırlıklı örneklerin gerçek konuma daha yakın olduğu ve düşük ağırlıklı örneklerin gerçek konumdan uzakta olduğu kabul edilir. $\delta(\cdot)$ dirac delta fonksiyonunu gösterir ve $\delta(a)$, a hariç her yerde 0’dır. İntegrali 1’e eşittir [53].

3.4.1. Önem dağılımı ve Önem ağırlıkları

Tahmin edilecek olan olasılık dağılımı bilinmediği için bu dağılım üzerinden örnekleme yapmak imkânsızdır. Bu yüzden örnekler farklı bir dağılımdan alınmalıdır. Örneklerin alınacağı dağılıma öneri (proposal) dağılımı denir ve π ile gösterilir. Bu yaklaşıma da önem örnekleme denir. Dağılımın pozitif olduğu yerde pozitif olan herhangi bir fonksiyon öneri dağılımı olarak kullanılabilir. Önem ağırlıkları sayesinde örneklerin alındığı fonksiyonun tahmin edilecek fonksiyon olmaması bir sorun yaratmaz [53][54]. Önem ağırlıkları aşağıdaki şekilde hesaplanır.

$$w_k^i \propto \frac{p(x_{0:k}^i|z_{i:k})}{\pi(x_{0:k}^i|z_{1:k})} \quad (3.74)$$

Bu eşitlikte ağırlıklar, hedef dağılım ve öneri dağılım arasındaki oran ile bulunur. Önem ağırlığı p ve π arasındaki uyumsuzluk oranıdır. Şekil 3.5’te ağırlıkların belirlenmesine ait bir gösterim sunulmuştur [4].

Şekil 3.5.a.’da örnekleme yapılmak istenen hedef dağılım $p(x)$ görülmektedir. Fakat örnekler bu dağılım üzerinden alınamayacağı için örnekler Şekil 3.5.b’de gösterilen π dağılımı üzerinden alınır ve parçacıklar elde edilir. Fakat oluşturulan parçacık seti $\pi(x)$

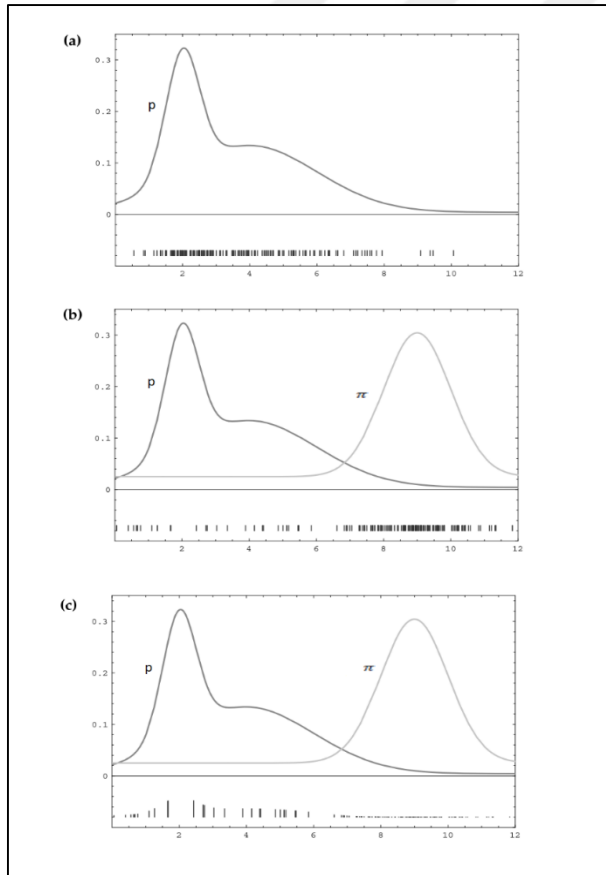
fonksiyonuna aittir. Bu farkı ortadan kaldırmak için elde edilen bu parçacıklar Eş 3.74'te olduğu gibi $p(x)$ ve $\pi(x)$ dağılımların oranları ile ağırlıklandırılır. Sonuç olarak elde edilen ağırlıklı parçacıklar Şekil 3.5.c'de gösterildiği gibi olur.

Yinelemeli filtrede tüm zaman aralıkları yerine sadece k zamanındaki dağılımı incelemek yeterlidir. Ayrıca Bölüm 3.4'te anlatıldığı gibi parçacıklar bir önceki durumdan elde edileceği için eşitlik aşağıdaki şekilde yeniden yazılabilir.

$$w_k^i \propto w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{\pi(x_k^i | x_{k-1}^i, z_k)} \quad (3.75)$$

Burada w_{k-1}^i bir önceki durumda i parçacığının ağırlığını ifade eder.

Parçacık filtresinin en temel algoritması Çizelge 3.4'te verilmiştir.



Şekil 3.5. PF için önem örneklemesinin gösterimi. a) Hedeflenen dağılım $p(x)$ b) Parçacıkların alındığı $\pi(x)$ öneri dağılımı. c) $p(x)$ ve $\pi(x)$ oranlaması sonucu elde edilen önem ağırlıkları [4]

Çizelge 3.4. Parçacık Filtresi Algoritması [4]

1	Parçacık Filtresi Algoritması
2	$\bar{X}_k = X_k = \emptyset$
3	<i>for</i> $m = 1$ <i>to</i> M <i>do</i>
4	<i>sample</i> $x_k^{[i]} \sim \pi(x_k)$
5	$w_k^{[j]} = \frac{p(x_k^{[j]})}{\pi(x_k^{[j]})}$
6	$\bar{X}_k = \bar{X}_k + \langle x_k^{[i]}, w_k^{[j]} \rangle$
7	<i>endfor</i>
8	<i>return</i> X_k

Başlangıçta parçacık kümesi boş kabul edilir. 4. adımda m indeksi ile gösterilen parçacığın öneri dağılımı üzerinden parçacıklar üretilir. 5. adımda üretilen bu parçacıkların önem ağırlıkları bulunur. 6. adımda ise bulunan ağırlığa göre hedef dağılım güncellenir.

Robot lokalizasyonunda işlem ve ölçüm gürültülerinin kovaryans matrisleri Q ve R 'nin, başlangıçta sıfır ortalamalı Gauss dağılımına sahip esas köşegenli matrisler olduğu varsayılır. Burada önem örnekleme fonksiyonu olarak $p(x_k^i | x_{k-1}^i)$ seçilebilir. Seçilen bu fonksiyondan üretilecek parçacıklar aşağıdaki gibi olur.

$$x_k^i \sim \mathcal{N}(x_{k-1}^i + B_k u_k, Q) \quad (3.76)$$

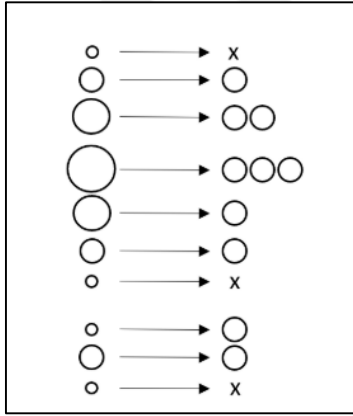
Bu eşitlikte $\mathcal{N}$ dağılımın bir Gauss dağılımı olduğunu ifade eder. Bu eşitliğe göre parçacıklar ortalaması $x_{k-1}^i + B_k u_k$ ve kovaryans matrisi Q olan bir Gauss dağılımından üretilir. Ortalama değeri Eş 3.2'ye benzer şekilde robotun aldığı hareket konumuna göre önceki konumundaki değişim üzerinden bulunur. Q ise işlem gürültü kovaryans matrisidir. Seçilen bu önem örnekleme fonksiyonu Eş. 3.75'te yerine yazılırsa önem ağırlığı aşağıdaki eşitlikte olduğu gibi hesaplanır [53].

$$w_k^i \propto w_{k-1}^i p(z_k | x_k^i) \quad (3.77)$$

3.4.2. Yeniden örnekleme

Çizelge 3.4'te gösterilen algoritma belli bir süre kullanıldıktan sonra bir parçacığın ağırlığı bire yaklaşacak ve diğer parçacıkların ağırlığı neredeyse sıfır olacaktır. Bu duruma dejenere

olma sorunu denir ve bundan kaçınmak mümkün değildir [3]. Dejenere sonucu harcanan tüm işlem yükü tahmine hiç katkısı olmayan parçacıklar üzerine olacak ve en önemlisi tahmin tek bir parçacık üzerinden yapılıyor olacaktır. Bu durum filtrenin tahmin yeteneğini fazlasıyla sınırlayacaktır. Tahmin hataları artacak ve filtrenin performansı düşmeye başlayacaktır. Bu sorun yeniden örnekleme yapılarak çözülür. Yeniden örnekleme ile yeni parçacıklar ağırlıklı parçacıklar kümesinden rastgele seçilir. Bir parçacığın yeniden seçilme ihtimali ağırlığı ile orantılıdır [53], [55], [56]. Eğer bir parçacığın ağırlığı diğerlerinden daha fazla ise yeniden seçilme olasılığı da daha fazladır. Yeniden örneklenen parçacıklar eşit ağırlığa sahip olur ve durum tahminine katkı sağlayan parçacık sayısı değişmeden filtre çalışmaya devam eder. Yeniden örnekleme örneği Şekil 3.6'da gösterilmiştir.



Şekil 3.6. Yeniden örnekleme örneği [53].

Yukarıdaki şekilde ağırlıklı parçacıkların yeniden örnekleme öncesi durumu soldaki yuvarlaklar ile gösterilmiştir. Daha büyük yuvarlağa sahip parçacığın ağırlığı daha fazladır. Yeniden örnekleme sırasında, bazı parçacıklar birden çok kez seçilirken, diğer parçacıklar 'x' ile gösterildiği gibi hiç seçilmez.

Yeniden örnekleme için literatürde birden çok yöntem önerilmiştir [14]. Bu örnekleme yöntemleri devam eden alt başlıklarda anlatılmıştır.

Çok Terimli Yeniden Örnekleme (CTYÖ)

Bu örnekleme metodunun temel fikri, $(0, 1]$ üzerindeki düzgün dağılımdan bağımsız olarak N adet u_k^n rastgele sayı seçmek ve bunları X_k^n parçacık kümesinden parçacıkları seçmek için kullanmaktır [57]. n 'inci seçimde uygun parçacık aşağıdaki şart sağlandığında seçilir.

$$J_k^{m-1} < u_k^n \leq J_k^m \quad (3.78)$$

Burada J_k^m ağırlıkların kümülatif toplamını ifade eder ve aşağıdaki gibi hesaplanır.

$$J_k^m = \sum_{k=1}^m w_k^m \quad (3.79)$$

Seçilen rastgele u_k^n sayı ağırlıkların kümülatif toplamı ile sınırlanan bölgede ise o bölgeye karşılık gelen parçacık yeni parçacık seti için seçilir. Görselleştirilmiş hali Şekil 3.7’de verilmiştir.



Şekil 3.7. Çok terim yeniden örnekleme yönteminin görselleştirilmiş hali

Burada $w_1, \dots, w_m$ parçacıklara ait ağırlıkları, $u_1, \dots, u_m$ seçilen rastgele sayıları gösterir. Örneğin u_3 rastgele sayısı için düşünülürse seçilen bu rastgele sayının değeri Eş. 3.78’e göre ilk 4 parçacığın kümülatif toplamı ile ilk 3 parçacığın kümülatif toplamı arasında olduğu için 3. parçacık yeni parçacık kümesine seçilir.

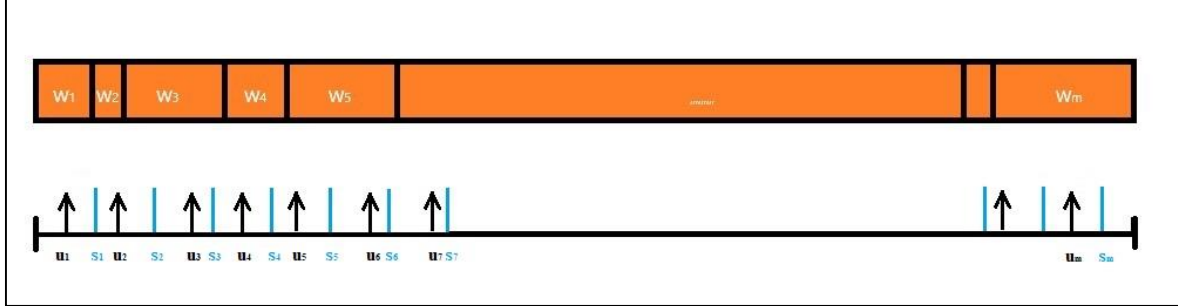
Çok Katmanlı Yeniden Örnekleme (ÇKTÖ)

Çok Katmanlı Yeniden Örnekleme metodunda parçacık topluluklarının katman adı verilen alt topluluklara bölündüğü varsayılmıştır. $(0,1]$ aralığı N adet alt bölüme ayrılır. Rastgele sayılar u_k^N her aralıkta birbirinden bağımsız olarak rasgele şekilde seçilir [58].

$$u_k^N \sim U\left(\frac{n-1}{N}, \frac{n}{N}\right], n = 1, 2, \dots, N \quad (3.80)$$

Seçilen bu rastgele sayılar ile çok terimli yeniden örneklemede olduğu gibi Eş. 3.78’deki kümülatif toplam aralıklarına denk gelmesi durumuna göre yeni parçacık setine uygun

parçacıklar seçilerek yeniden örnekleme işlemi tamamlanır. Bu işlem Şekil 3.8’de gösterilmiştir.



Şekil 3.8. Çok katmanlı yeniden örneklemenin görselleştirilmiş hali

Yukarıdaki şekilde $s_1, s_2, \dots, s_m$ ile isimlendirilen mavi çizgiler katmanları ifade eder. Siyah oklar ise katmanlar arasında rastgele seçilen u_k^N rastgele sayıdır. Bu rastgele sayılara denk gelen parçacıklar seçilerek yeni parçacık setine eklenir. Örneğin s_2 ve s_3 katman çizgileri arasında kalan u_3 rastgele sayısı ağırlığı w_3 olan parçacığı işaret etmektedir. Bu katman sınırları için yeniden örneklenen parçacık ağırlığı w_3 olan parçacık olacaktır.

Sistemik Yeniden Örnekleme (SYÖ)

Sistemik yeniden örnekleme metodu çok katmanlı yeniden örneklemeyle benzer şekilde katmanlara ayrılmış alt parçacık grupları kullanır. Burada parçacık seçme aşamasında farklı bir yaklaşım uygulanır. İlk katmandan rastgele bir sayı seçildikten sonra diğer rastgele sayılar $1/N$ oranında artırılarak elde edilir [58,59].

$$u_t^1 \sim U\left(0, \frac{1}{N}\right) \quad (3.81)$$

$$u_k^n = u_t^1 + \frac{n-1}{N} \quad (3.82)$$

Burada u_t^1 ilk katmandan seçilen rastgele sayıdır. Devamında seçilen rastgele sayılar bu seçilen rastgele sayıya $1/N$ eklenerek hesaplanır. Bu işlem Şekil 3.9’da gösterilmiştir.



Şekil 3.9. Sistematik yeniden örneklemenin görselleştirilmesi

Burada çok katmanlı yeniden örneklemeden farklı olarak ilk katmandan kırmızı ok ile gösterilen rastgele sayı seçilir ve siyah ok ile gösterilen rastgele sayılar bu sayıya göre belirlenir. Bu yöntemin çok katmanlı yeniden örnekleme göre en büyük avantajı tek bir rastgele eleman seçme işlemi barındırdığı için hesaplama açısından daha verimli olmasıdır.

Artık Yeniden Örnekleme (AYÖ)

Artık yeniden örnekleme iki aşamadan oluşur. İlk aşamada ağırlığı $1/N$ 'den büyük olan parçacıklar yeni parçacık seti için seçilir. İkinci adımda geri kalan parçacıklar (artık parçacıklar) kullanılarak rastgele örnekleme yapılır. Artık yeniden örnekleme ile m 'inci parçacık $N_k^m + R_k^m$ kere yenilenir, burada N_k^m ilk adımda yenilenen parçacık sayısı, R_k^m ise ikinci adımda yenilenen parçacık sayısıdır. $N_k^m = [Nw_k^m]$ şeklinde hesaplanır. İlk adımda toplam yenilenen parçacık sayısı $N_k = \sum_{m=1}^M N_k^m$ şeklinde bulunurken ikinci adımda toplam yenilenen parçacık sayısı $R_k = M - N_k$ şeklinde bulunur [60], [61]. Ağırlıkların artıkları ise aşağıdaki gibidir.

$$\hat{w}_k^m = w_k^m - \frac{N_k^m}{M} \quad (3.83)$$

İkinci aşamada parçacıklar artık ağırlıklara, $\hat{w}_k^m$, göre çok terimli yeniden örnekleme kullanılarak seçilir.

Metropolis Yeniden Örnekleme (MYÖ)

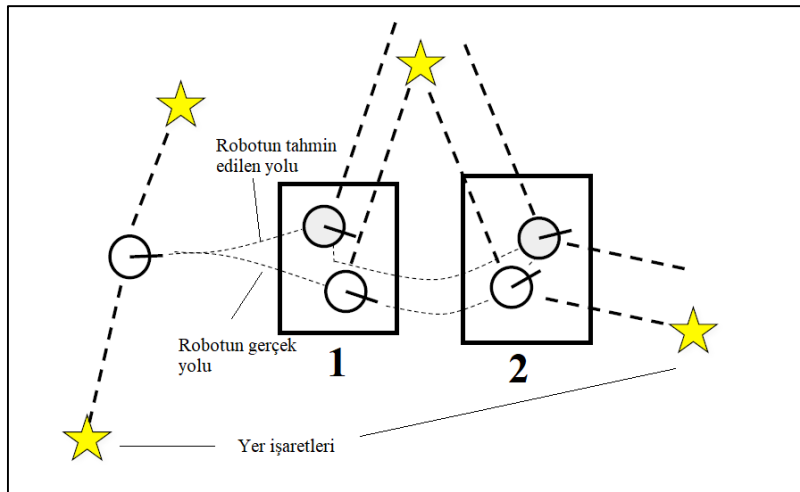
Metropolis Yeniden Örnekleme metodunda diğer yeniden örnekleme metodlarının aksine ağırlıkların normalize edilmesine gerek yoktur. Burada önemli olan ağırlıklar arasındaki

orandır. Önceden seçilmiş parçacığın ağırlığı ile $1, \dots, N_c$ aralığında, tüm parçacıklar arasından rastgele seçilen parçacıkların ağırlıkları ile oranlanır ve bu oran u_k rastgele sayısı ile kıyaslanır. Eğer bu oran u_k 'dan büyük ise $1, \dots, N_c$ aralığından seçilen rastgele parçacık yeni parçacık setine eklenir. u_k 'dan küçük ise önceden seçilmiş parçacık yeni parçacık setine yeniden seçilir [62].

Bu yöntemde N_c seçimi çok önemlidir. Bu seçim hız ve güvenilirlik arasında birinden ödün vererek gerçekleştirilebilir. Eğer N_c küçük seçilirse burada güvenilirlik azalır ve örnekleme taraflı olur fakat işlem maliyeti azalır, örnekleme süresi kısalmır. N_c büyük seçilirse seçilecek parçacıklar için hassasiyet artar fakat aynı şekilde işlem maliyeti de artar. Bu yeniden örnekleme yönteminin en büyük dezavantajı N_c 'nin uygun şekilde ayarlanmasının oldukça zor olmasıdır.

3.5. Eş Zamanlı Yer Tahmini ve Haritalandırma (EZYTH)

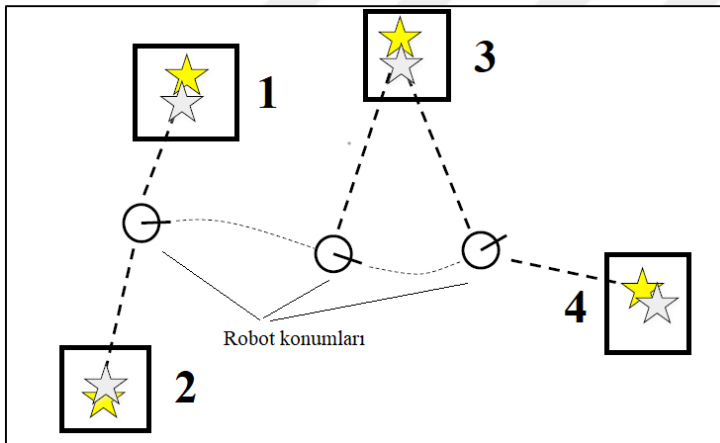
EZYTH problemi robotun bulunduğu ortamın haritasına ulaşamadığı ve bulunduğu konumu bilmediği durumda ortaya çıkar. EZYTH'de robot bulunduğu ortamı haritalarken aynı zamanda o haritaya göre kendi konumunu da belirlemelidir. Harita bilinmediği için konum belirlemek zor, konum bilinmediği için de haritayı çıkartmak zordur. Aynı şekilde bu iki durumun da sürekli olarak tahmin edilmesi problemin zorluğunu daha da artırmaktadır. Şekil 3.10'da yer tahmini için bir örnek gösterilmiştir.



Şekil 3.10. Harita bilindiğinde yer tahmini örneği

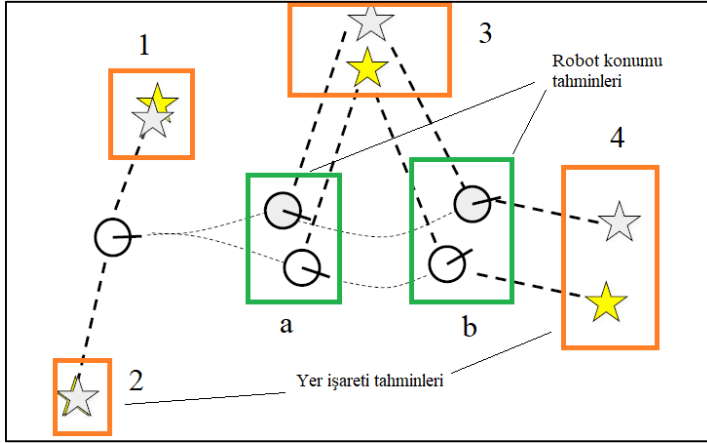
Bu şekilde sarı yıldızlar yer işaretlerini, yuvarlaklar ise robotun konumunu göstermektedir. Yer tahmini örneğinde, önceden yer işaretlerinin konumları robota verilmiştir. Robot harekete başlar ve 1 numaralı konuma gelir. Bulunduğu konumda aldığı ölçüm bilgisinde görmesi gereken yer işaretini göremediği için konumunda bir güncelleme yapıp alttaki konumuna geçer ve 2 numaralı konuma doğru harekete başlar. Burada da aynı şekilde yine bir konum güncellemesi yapar ve hareketini tamamlar. Burada yapılan işlem yer tahminidir ve robot önceden kendine verilen haritaya göre kendi konumunu tahmin eder.

Şekil 3.11’de ise bir haritalandırma örneği verilmiştir. Burada robotun konumu hakkında bilgi sahibiyken yer işaretleri hakkında bilgimiz yoktur. Robot ortamda hareket ederken sensörleri vasıtasıyla aldığı verilere göre 1, 2, 3 ve 4’te gösterildiği gibi bazı yer işareti tahminleri yapar. Sensörlerden aldığı veriler gürültülü olduğu için yer işaretlerinin gerçek konumları ve tahmin edilen konumları arasında bazı farklar oluşur. Burada sarı yıldızlar gerçek konumları gösterirken gri yıldızlar tahmin edilen konumlardır.



Şekil 3.11. Robot konumları bilindiğinde haritalandırma örneği

Her iki şekilde de ya robot konumu biliniyordu ya da ortamın haritası yani yer işaretlerinin konumları verilmişti. EZYTH probleminde ise robot, hem yer işaretlerinin yerini hem de konumunu bilmez. Şekil 3.12 bu durumun bir örneğini göstermektedir. Robot yer işaretlerinin yerlerini bilmediği için sensörden gelen bilgiye göre bir tahmin yapar. Kendi konumu için de referans alacağı bir haritası olmadığı için Şekil 3.10’da olduğu gibi bir güncelleme yapmadan hareketine devam eder. Robotun yer işaretleri için yaptığı tahminin kalitesi robotun konumu için önemli olduğu kadar robotun konum tahmini de yer işaretlerinin tahmini için oldukça önemlidir.



Şekil 3.12. Eş zamanlı yer tahmini ve haritalandırma örneği

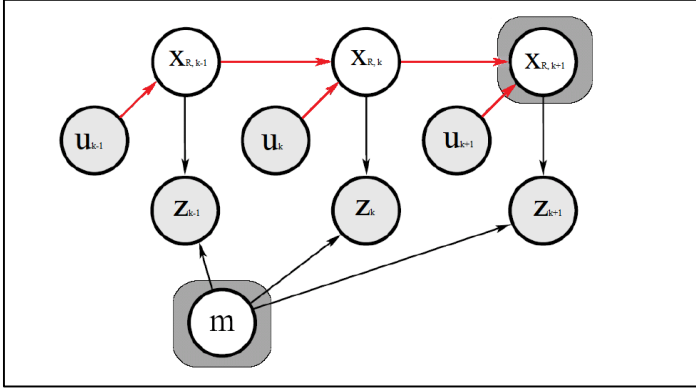
3.5.1. Hareket ve gözlem modelleri

Robot, hareketi boyunca ona verilen komutlar ve sensörlerden aldığı bilgiler dışında hiçbir bilgiye sahip değildir. Robot yapacağı tüm tahminleri komutlar ve sensörlerden gelen bilgiler dâhilinde yapar. Komutlar u_k ile gösterilirken sensörden gelen gözlem bilgileri z_k ile gösterilir. u_k ve z_k kullanılarak durum vektörü x_k tahmin edilir. Durum vektörü, robotun ve yer işaretlerinin konum bilgilerini içerir.

$$x_k = [x_{r,k}, y_{r,k}, \theta_{r,k}, x_{L,x}^1, x_{L,y}^1, \dots, x_{L,x}^i, x_{L,y}^i]^T \quad (3.84)$$

Burada $x_R = [x_{r,k}, y_{r,k}, \theta_{r,k}]^T$ robotun konumunu temsil eder. $x_{r,k}, y_{r,k}$ robotun koordinat sistemi üzerindeki, k zamanındaki konumunu ve $\theta_{r,k}$ rotasyon açısını (heading) gösterir.

$m = x_L = [x_{L,x}^1, x_{L,y}^1, \dots, x_{L,x}^i, x_{L,y}^i]^T$ yer işaretlerinin konumlarını gösterir. $x_{L,x}^i$ ve $x_{L,y}^i$ i'inci yer işaretinin koordinat sistemi üzerindeki konumunu ifade eder [5]. EZYTH süresince komutlar ve sensörlerin, yer tahmini ve haritalandırma ile ilişkisi Şekil 3.13'te verilen grafik modelinde gösterilmiştir.



Şekil 3.13. x_k durum vektörü ($[x_R, m]$), u_k komut vektörü ve z_k gözlem vektörünün ilişkisini gösteren grafik modeli [4]

Bu grafik modelindeki oklar, yuvarlakların içindeki vektörlerin birbirleri arasındaki bağlantıyı gösterir. $x_{R,k}$ 'yi bulabilmek için kırmızı oklarla gösterilen $x_{R,k-1}$ ve u_k hesaba katılmalıdır. z_k için siyah oklarla gösterilen bağlantılar ile x_k durum vektörü ($[x_R, m]$) hesaba katılmalıdır. Bu iki bağlantı sırasıyla hareket ve gözlem modeli olarak adlandırılır.

Hareket modeli

Hareket modeli, $p(x_k | x_{k-1}, u_k)$ 'yi tanımlamak için bir bisikletin ön tekerleğinin yörüngesi için yuvarlanma hareketine tâbi kinematik model kullanılır [63]. Bu modele göre hareket sonrası robotun tahmini konumu aşağıdaki gibi hesaplanır.

$$x_{R,k} = f_R(x_{R,k-1}, u_k) = \begin{bmatrix} x_{r,k-1} + \mathcal{V}_k \nabla k \cos(\theta_{r,k-1} + \mathcal{G}_k) \\ y_{r,k-1} + \mathcal{V}_k \nabla k \sin(\theta_{r,k-1} + \mathcal{G}_k) \\ \theta_{r,k-1} + \frac{\mathcal{V}_k \nabla k}{WB} \sin(\mathcal{G}_k) \end{bmatrix} \quad (3.85)$$

Burada $x_{r,k-1}, y_{r,k-1}$ ve $\theta_{r,k-1}$ robotun $x_{R,k-1}$ durumundaki konum bilgileridir. $\mathcal{V}_k$ ve $\mathcal{G}_k$ robota uygulanan komut vektörü $u_k = [\mathcal{V}_k, \mathcal{G}_k]^T$ 'yi oluşturur. $\mathcal{V}_k$ robotun hızını, $\mathcal{G}_k$ robotun direksiyon açısını temsil eder. WB robotun ön ve arka tekerleri arasındaki teker mesafesini, ∇k ise k zamanı ile $k - 1$ arasında geçen süreyi tanımlar. x_k durum vektörü Eş. 3.84'te gösterildiği gibi $x_{R,k}$ ve $m = x_L$ durağan yer işaretlerinden oluşur. Hareket modeli yer işaretlerinin de kinematik model içine eklenmesi ile tamamlanır.

$$x_k = f(x_{k-1}, u_k) = \begin{bmatrix} x_{R,k} = f_R(x_{R,k-1}, u_k) \\ m = x_L \end{bmatrix} \quad (3.86)$$

Gözlem Modeli

Gözlem modeli, $p(z_k|x_k)$ 'yi tanımlar. Eğer daha önceden haritada işaretlenmiş bir yer işareti mesafe-yön (range-bearing) sensörü tarafından gözlemlenmiş ise z_k gözlem vektörü aşağıdaki gibi tanımlanır [5].

$$z = \begin{bmatrix} r \\ \theta \end{bmatrix} \quad (3.87)$$

Burada (r, θ) sırasıyla gözlemciye yani robota göre mesafe ve yön bilgilerini içerir. Yer işaretlerinin bulunduğu m durum vektörü $[x_{L,x}^1, x_{L,y}^1, \dots, x_{L,x}^i, x_{L,y}^i]^T$ şeklinde tanımlanmıştır. Gözlemlenmiş $(x_{L,x}^i, x_{L,y}^i)$ yer işaretçisi aşağıdaki bağıntı sayesinde harita ile ilişkilendirilir.

$$z_{i_k} = h_i(x_k) = \begin{bmatrix} \sqrt{(x_{L,x}^i - x_{r,k})^2 + (y_{L,x}^i - y_{r,k})^2} \\ \arctan \frac{y_{L,x}^i - y_{r,k}}{x_{L,x}^i - x_{r,k}} - \theta_{r_k} \end{bmatrix} \quad (3.88)$$

Robot haritada ilerlerken yeni yer işaretleri görür. Bu işaretler haritaya dolayısıyla durum vektörü x_k 'ya eklenmelidir. Bu işlem ters gözlem kavramı kullanılarak gerçekleştirilir. Durum vektörü içinde bulunan mevcut robot bilgisi ve yer işaretinin konumu kullanılarak bu süreç başlatılır ve durum vektörü yeni gözlem z_i 'nin kutup değerlerini kullanarak genişletilir. Bu işleme durum artırımı (state augmentation) denir [5].

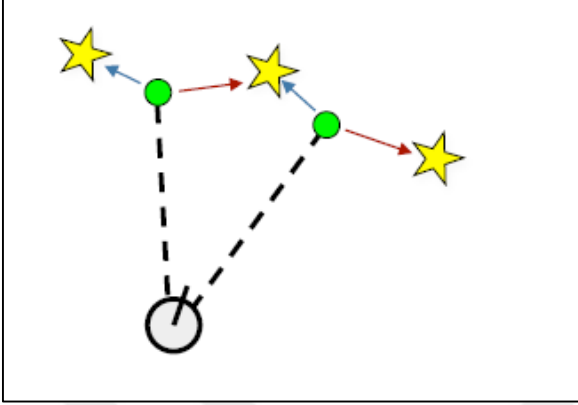
$$x_L^{new} = h^{-1}(x_{R,k}, z_k) = \begin{bmatrix} x_{L,x}^{new} \\ x_{L,y}^{new} \end{bmatrix} = \begin{bmatrix} x_{r,k} \\ y_{r,k} \end{bmatrix} + \begin{bmatrix} r \cos(\theta + \theta_{r_k}) \\ r \sin(\theta + \theta_{r_k}) \end{bmatrix} \quad (3.89)$$

Burada (r, θ) Eş. 3.87'de gösterilen mesafe ve yön bilgileridir.

3.5.2. Veri İlişkilendirmesi

Veri ilişkilendirmesi, daha önce görülen yer işaretleri ile yeni gözlemler arasındaki ilişkiyi kurar. GKF içinde hem tahminlerin hem de gözlemlerin Gauss belirsizliğine sahip olduğu varsayıldığından ve PF içinde de Şekil 3.14'de gösterildiği gibi gözlemin birden fazla

parçacık ile yapılmasından dolayı verilen bir gözlemin herhangi bir hedefe karşılık gelmesi mümkündür.



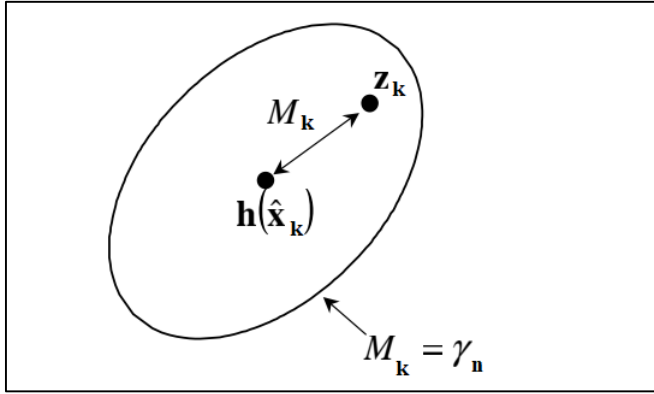
Şekil 3.14. PF'de veri ilişkilendirmesi sorununun gösterimi

Olası olmayan gözlemleri elemek için yalnızca bir gözlemin olası komşuluğu dikkate alınmalıdır. Veri ilişkilendirmesi genellikle gözlem uzayında gerçekleştirilir ve doğrulama kapısı (validation gate) x_k durumu için z_k gözlemi ile $\hat{z}_k = h(\hat{x}_k)$ gözlem tahmini arasındaki izin verilen maksimum tutarsızlığı tanımlar.

En yaygın kullanılan istatistiksel doğrulama kapısı, normalize edilmiş yenilik karesi (NEYK) diğer bir ismi ile Mahalanobis mesafesidir. Bir yenilik vektörü $e_k = z_k - h(\hat{x}_k)$ ve onun kovaryans matrisi S_k verildiğinde NEYK aşağıdaki gibi tanımlanır [5], [65].

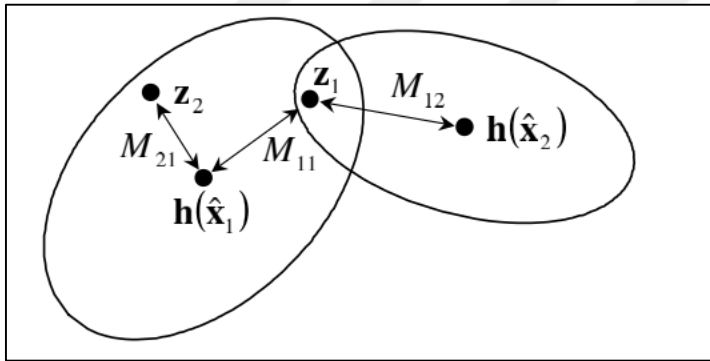
$$M_k = e_k^T S_k^{-1} e_k \quad (3.90)$$

Burada M_k Mahalanobis mesafesini ifade eder. Doğrulama kapısı Şekil 3.15'te gösterildiği gibi tahmin edilen gözlem $h(\hat{x}_k)$ etrafında ortalananmış bir elips olarak görselleştirilebilir. Bir gözlemin kabul edilebilir olması için bu elipsin içinde bulunması gerekir. γ_n n boyutlu yenilik vektörü için seçilecek maksimum NEYK eşiğini gösterir.



Şekil 3.15. Doğrulama kapısının gösterimi [5]

Gözlem tahmininin doğrulama kapısı içinde birden çok gözlemin bulunması veya bir tek gözlemin birden çok gözlem tahmininin doğrulama kapısı içinde bulunması durumunda veri ilişkilendirmesi belirsizliği ortaya çıkar. Bu belirsizliğin bir örneği Şekil 3.16'da gösterilmiştir. Benzersiz bir kapı uygulaması durumunda bile bu belirsizlik gözlemlenemeyen veya yanlış gözlemlenen yer işaretleri ortaya çıkarabilir.



Şekil 3.16. Gözlemin hangi doğrulama kapısı aralığına ait olduğu ile ilgili belirsizliğin gösterimi [5]

Belirsizlik sorununun çözümünde en yaygın kullanılan yöntem en yakın komşu (nearest neighbour) veri ilişkilendirmesidir [66]. x tahmininin doğrulama kapısı içerisinde olan $z = \{z_1, \dots, z_n\}$ gözlem kümesi verilmiş olsun. Her $z_k \in z$ için ilişkilendirme olasılığı Λ_k aşağıdaki gibi olur.

$$\Lambda_k = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{|S_k|}} \exp\left(-\frac{1}{2} e_k^T S_k^{-1} e_k\right) \quad (3.91)$$

Burada n yenilik vektörünün boyutunu ifade eder. Bu ifade aslında bir Gauss bağlantısının olabilirlik fonksiyonunu ifade eder. Eşdeğer bir gösterim, normalize edilmiş N_k mesafesi ile Eş. 3.91'nin logaritması alınarak aşağıdaki gibi bulunur.

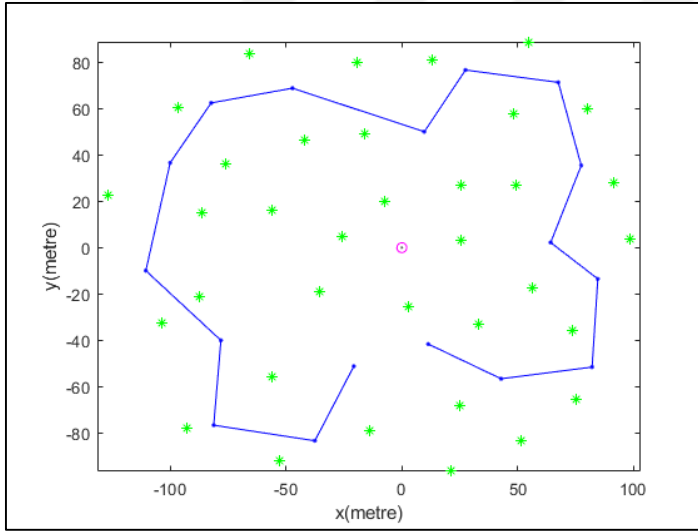
$$N_k = e_k^T S_k^{-1} e_k + \ln |S_k| \quad (3.92)$$

En yakın komşu veri ilişkilendirmesi N_k 'yi minimize eden veya Λ_k 'yi maksimize eden gözlemi seçerek veri ilişkilendirmesini sağlar. Eş. 3.91 ve Eş. 3.92 birbirine eşit olduğu için bu yöntemle maksimum olabilirlik veri ilişkilendirmesi de denir.



4. BENZETİM SONUÇLARI VE TARTIŞMA

Bu bölüm içerisinde Bölüm 3'te anlatılan EZYTH problemi çözüm yöntemleri MATLAB benzetim programı içerisinde farklı gürültü istatistiklerinde ve veri ilişkilendirmesi durumunda karşılaştırılmıştır. Bu karşılaştırmalarda amaç anlatılan bu çözüm yöntemlerinin gerçek hayat şartlarında verdikleri tepkileri görmektir. Tim Bailey tarafından tasarlanmış ve açık kaynak kodlu olarak tüm araştırmacılarla paylaşılmış olan “SLAM SIMULATOR” [2], [41] karşılaştırılmak istenen durumlara göre düzenlenmiştir. Bu benzetim programında kullanılacak olan harita Şekil 4.1'de gösterilmiştir. Burada nokta ile gösterilen yerler yol noktalarını (way point) mavi çizgi robotun alacağı doğru yolu ve yeşil yıldızlar ortamda bulunan yer işaretlerini gösterir. Robota uygulanan $u_k = [V, G]^T$ komut vektörü sırasıyla hız ve direksiyon açısından oluşur. Robot sıradaki yol noktasına gelene kadar V hızı ile hareket eder ve yol noktasına vardıktan sonra diğer yol noktasının konumuna göre G açısı ile dönüşünü gerçekleştirir.



Şekil 4.1. Deneylerde kullanılacak olan harita

Bölüm 3.3'te gösterilen gürültü istatistiklerinin vektörel ve matris olarak gösterimi aşağıdaki gibidir.

$$q = \begin{bmatrix} q_v \\ q_G \end{bmatrix}, Q = \begin{bmatrix} q_v^2 & 0 \\ 0 & q_G^2 \end{bmatrix} \quad (4.1)$$

$$r = \begin{bmatrix} r_R \\ r_B \end{bmatrix}, R = \begin{bmatrix} r_R^2 & 0 \\ 0 & r_B^2 \end{bmatrix} \quad (4.2)$$

Burada q işlem gürültüsünün ortalama durum vektörünü, Q ise kovaryans matrisini ifade eder. q_v ve q_G , sırasıyla u_k komut vektöründeki hız bileşenini ve dönüş açısını etkileyen gürültü bileşenleridir. r ve R , sırasıyla, ölçüm gürültüsünün ortalama durum vektörünü ve kovaryans matrisini gösterir. r_R ölçüm mesafesi, r_B ise yönelimle ilgili olan gürültü bileşenlerini ifade eder. Robot, bir ortamda hareket ederken bu dört gürültü bileşeninden ayrı ayrı etkilenebilir. Bu sebeple yapılan testler altı ana başlık altına toplanmıştır. Bunlar Çizelge 4.1’de verilmiştir.

Çizelge 4.1. Gürültü bileşenlerinin uygulanan testlerdeki başlangıç değerleri

	q_v (m/s)	$q_G(\pi/180 \text{ rad})$	$r_R(m)$	$r_B(\pi/180 \text{ rad})$
Gürültü testi – 1	[0,1: 0,02: 0,2]	1,0	0,1	1,0
Gürültü testi – 2	0,1	[1,0: 0,2: 2,0]	0,1	1,0
Gürültü testi – 3	0,1	1,0	[0,1: 0,02: 0,2]	1,0
Gürültü testi – 4	0,1	1,0	0,1	[1,0: 0,2: 2,0]
Gürültü testi – 5	[0,1: 0,02: 0,2]	[1,0: 0,2: 2,0]	0,1	1,0
Gürültü testi – 6	0,1	1,0	[0,1: 0,02: 0,2]	[1,0: 0,2: 2,0]

Gürültü testi 1-4 arasında her gürültü bileşeninin ayrı ayrı değişimlerinin filtreler üzerindeki etkisi incelenmek istenmiştir. 5-6. testlerde ise işlem gürültüsü bileşenlerinin değişimi ile ölçüm gürültüsünün değişiminin aynı anda etkisi gözlemlenmiştir. Sağlıklı ölçümler alabilmek ve anlamlı karşılaştırmalar yapabilmek için artış miktarları ve seçim aralıkları eşit olacak şekilde ayarlanmıştır.

Bu karşılaştırmalar yapılırken doğru değerler ile tahmin edilen değerler arasındaki farkın verimli şekilde hesaplanabileceği RMSE değerleri bulunmuştur. RMSE, EZYTH performansını izlemek için yaygın şekilde kullanılan bir hata ölçüm parametresidir.

$$RMSE_{x,k} = \sqrt{\|x_k - \hat{x}_k\|^2} \quad (4.3)$$

$$RMSE_{y,k} = \sqrt{\|y_k - \hat{y}_k\|^2} \quad (4.4)$$

Burada x_k ve y_k robotun k zamanındaki doğru konumunun (x, y) bileşenlerini, $\hat{x}_k$ ve $\hat{y}_k$ k zamanındaki tahmin edilen konumunun (x, y) bileşenlerini ifade eder. Bu iki eşitlik kullanılarak bir hareket boyunca istenilen bir zaman aralığında robotun ortalama RMSE değerleri aşağıdaki gibi bulunur.

$$RMSE_{robot\ konumu,x,ortalama} = \sqrt{\frac{1}{K} \sum_{k=1}^K \|x_k - \hat{x}_k\|^2} \quad (4.4)$$

$$RMSE_{robot\ konumu,y,ortalama} = \sqrt{\frac{1}{K} \sum_{k=1}^K \|y_k - \hat{y}_k\|^2} \quad (4.5)$$

Burada K ölçüm aralığını ifade eder. Yer işaretlerinin ortalama RMSE değerinin hesaplanması ise aşağıda verilmiştir.

$$RMSE_{yer\ işaretlerinin\ konumu,ortalama} = \sqrt{\frac{1}{K} \sum_{k=1}^K \left(\frac{1}{2N} (m_k - \hat{m}_k)^T (m_k - \hat{m}_k) \right)} \quad (4.6)$$

Burada m_k Bölüm 3.5.1.'de gösterildiği gibi robotun k zamanında gözlemlediği yer işaretinin gerçek konumunu, $\hat{m}_k$ tahmin edilen konumunu, N , k zamanında gözlemlenmiş yer işareti sayısını gösterir. Eş. 4.4 ve 4.5 kullanılarak robotun konumunun ortalama RMSE değeri aşağıdaki gibi düzenlenebilir.

$$RMSE_{robotun\ x,y\ konumu,ortalama} = \sqrt{\frac{1}{K} \sum_{k=1}^K (\|x_k - \hat{x}_k\|^2 + \|y_k - \hat{y}_k\|^2)} \quad (4.7)$$

RMSE yapılan tahminlerin gerçek değerlere ne kadar yakın olduğunu gösterir. Ne kadar küçükse tahminler gerçek değerlere o kadar yakındır [67].

Yapılan karşılaştırmalar dört ana başlık altında toplanmıştır. Bu başlıklarda Bölüm 3.2-3'de bilgileri verilen çözüm yöntemleri kendi aralarında aşağıda verilen başlıklar altında incelenmiştir.

- GKF ile AGKF'nin her gürültü bileşeni için RMSE değerlerinin karşılaştırılması
- AGKF ve AGKF-DKT'nin her gürültü bileşeni için RMSE değerlerinin karşılaştırılması
- PF-FastSLAM yeniden örnekleme yöntemlerinin her gürültü bileşeni için RMSE değerlerinin karşılaştırılması

Veri ilişkilendirmesi, Bölüm 3.5.2'de anlatıldığı gibi yeni yapılan gözlemleri daha önceden yapılan gözlemlerle ilişkilendirme durumudur. Robot sensörlerden bir gözlem verisi elde ettiğinde bu veriyi kendi oluşturduğu haritadaki gözlem tahmini ile karşılaştırır. Eğer bu veri

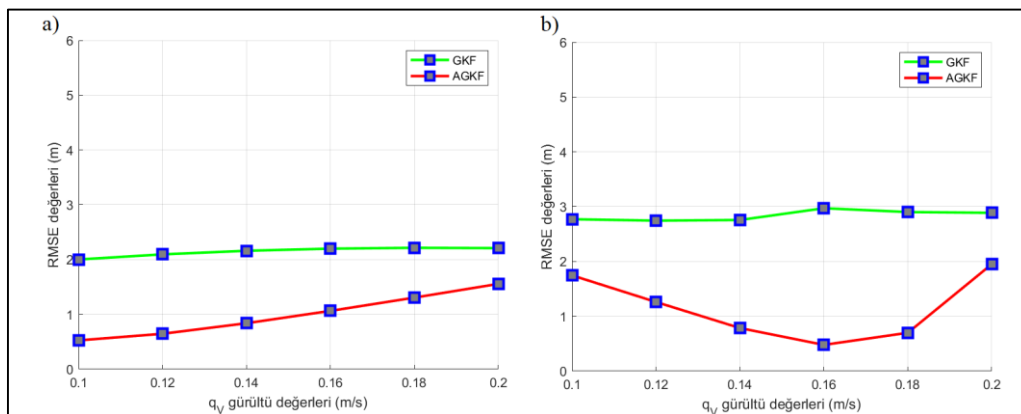
yaptığı tahminin doğrulama aralığında ise tahmin-veri arasında bir ilişkilendirme yapar. Bu ilişkilendirmeler bazı sensörlerde önceden robota sağlanabilir. RFID gibi gözlem verilerinin yer işaretleri ile ilişkilendirildiği sensörlerde veri ilişkisi bilinmektedir [68]. Bu sebeple yukarıda verilen karşılaştırma başlıkları, kendi içlerinde veri ilişkilendirmesinin robota daha önceden sağlandığı ve sağlanmadığı durumlarda incelenmiştir.

4.1. GKF ve AGKF'nin Karşılaştırması

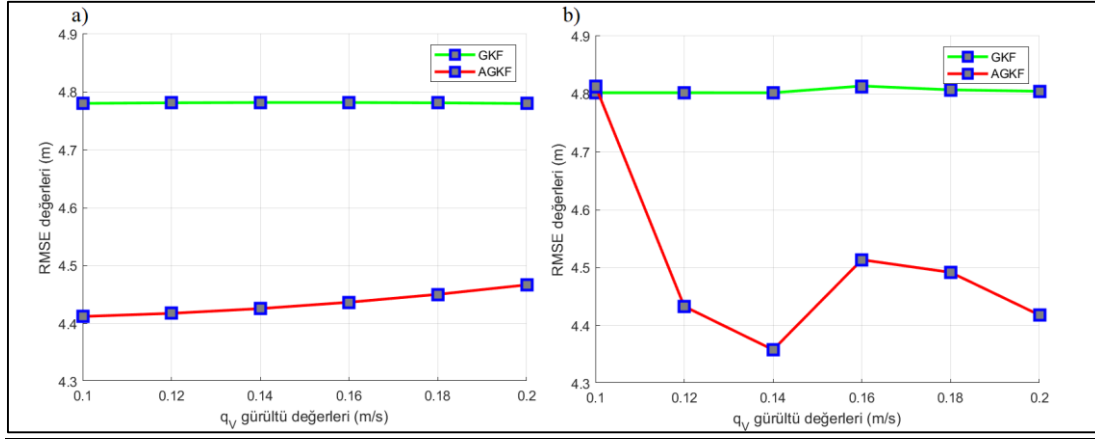
AGKF'nin GKF'nin adaptif hala getirilmesi Bölüm 3'te anlatılmıştır. Bu başlık altında bu iki çözüm yöntemi değişen gürültü altında ve veri ilişkilendirmesine sahip olduğu ve olmadığı durumlarda karşılaştırılmıştır.

4.1.1. İşlem gürültü bileşeni q_v 'nin değişimi

Robot, harekete kendisine verilen V , hız komutu ile başlar. Bu komut tekerlere iletilir ve tekerler dönmeye başlar fakat tekerler tam olarak uygulanan hızda dönemez. Buna q_v işlem gürültü bileşeni sebep olur. Bu bölümde q_v 'nin başlangıç değerindeki değişimin robotun yaptığı tahminler üzerindeki etkisi incelenmiştir. q_v 'nin başlangıç değerleri Çizelge 4.1'de Gürültü Testi – 1 olarak seçilmiştir. Testler yapılırken robot, harita üzerindeki gözlemleri ile tahminlerini, veri ilişkilendirmesinin verildiği ve verilmediği durumlar ayrı ayrı göz önüne alınarak yapmıştır. Yer işaretlerinin ve robotun, gerçek ve tahmini konumları arasındaki RMSE değerleri bulunmuştur. Gürültü Testi – 1 aralığında q_v için hesaplanan her RMSE değeri Şekil 4.2 ve 4.3'te, bu değerlerin ortalaması Çizelge 4.2'de verilmiştir.



Şekil 4.2. q_v 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durumda b) veri ilişkilendirmesi bilinmediği durumda

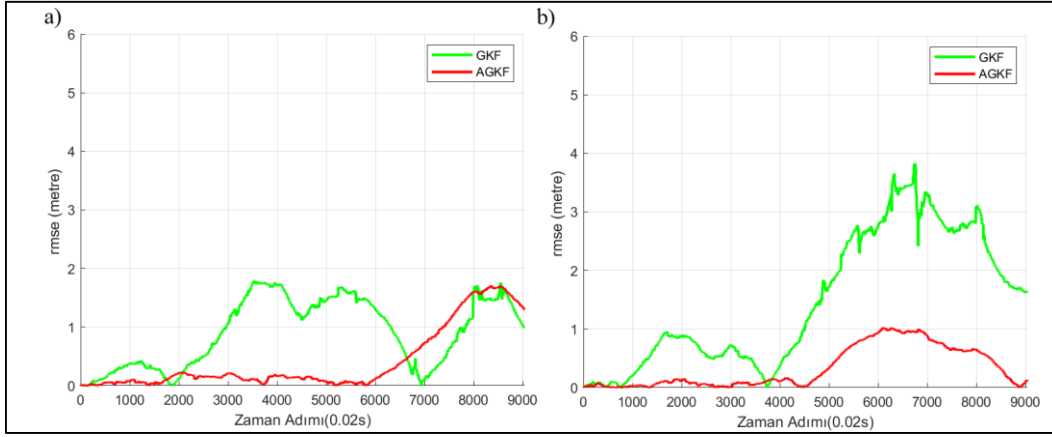


Şekil 4.3. q_v 'nin değişimine göre yer işareti konumları için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiğinde b) veri ilişkilendirmesi bilinmediğinde

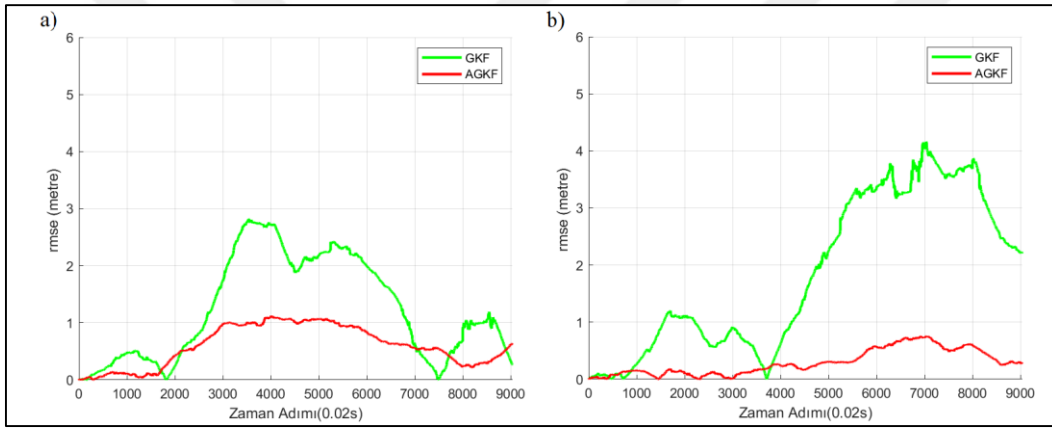
Çizelge 4.2. Robot ve yer işareti konum hatalarının değişen q_v gürültü başlangıç değerlerindeki ortalama hata değerleri

	GKF		AGKF	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
Robot konumunun hata değeri (m)	2,1475	2,8391	0,9901	1,1537
Yer işaretinin konumunun hata değeri (m)	4,7812	4,8054	4,4351	4,5049

Şekil 4.2.(a) ve 4.2.(b)'de görüldüğü gibi AGKF, q_v gürültüsünün değişen başlangıç değerlerinde robot konumu tahmininde GKF'ye kıyasla daha az hatalı tahminlerde bulunmuştur. Gürültü başlangıç değerinin farklı olmasının adaptasyon oranını değiştirdiği Şekil 4.2.(b)'de görülmektedir. Veri ilişkilendirmesi robota hareket öncesinde verildiğinde hata oranlarında bir iyileşme olduğu gözlemlenmiştir. Çizelge 4.2.'de verilen ortalama RMSE değerlerine göre veri ilişkilendirmesinin bulunduğu durumda GKF'deki iyileşme AGKF'ye göre daha fazla olmuştur. Yer işaretlerinde ise veri ilişkilendirmesi daha önceden yapıldığı için tahmin hatalarında fazla bir değişim olmadığı aynı çizelgede görülmektedir. AGKF'nin yer işaretleri konum tahmininde Şekil 4.3.(a) ve 4.3.(b)'de görüldüğü gibi GKF'ye göre daha başarılı olmuştur ancak tahmin hataları arasındaki fark çok az olmuştur. $q_v = 0,14m/s$ olduğu durumda RMSE değerlerinin zamanla değişimi Şekil 4.4 ve 4.5'te verilmiştir.

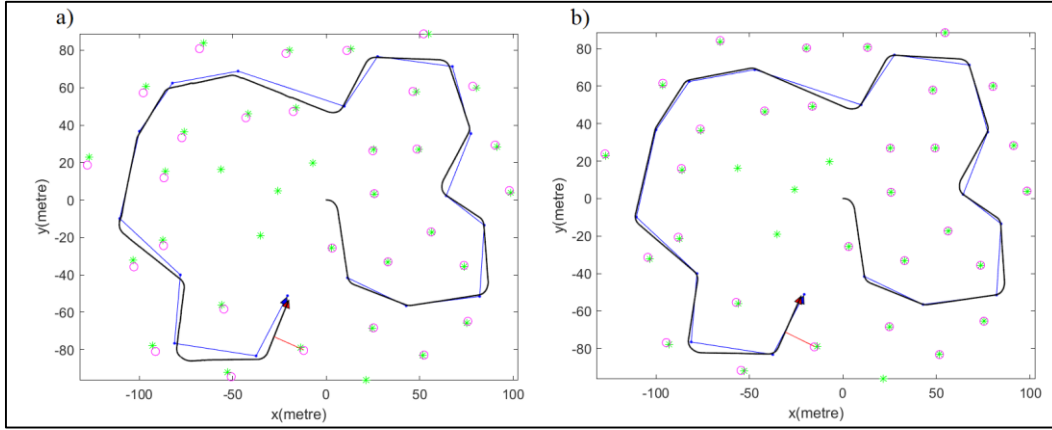


Şekil 4.4. Veri ilişkilendirmesi verildiği durumda, $q_v = 0,14m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

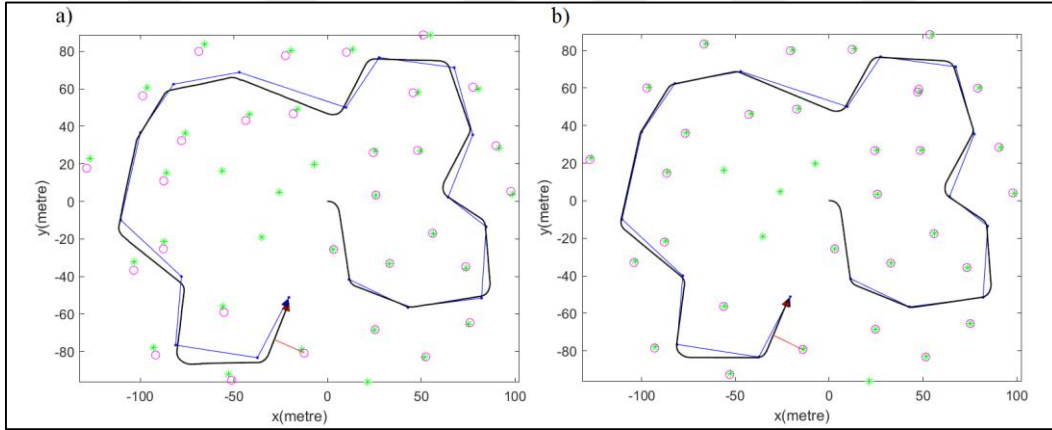


Şekil 4.5. Veri ilişkilendirmesi verilmediği durumda, $q_v = 0,14m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) robot konumunun x bileşeni için b) robot konumunun y bileşeni için

Şekil 4.4 ve 4.5'te görüldüğü gibi robotun haritada hareketi süresince konumu ile ilgili tahminindeki hata oranlarının artmaya ve azalmaya başladığı noktalar iki filtre için de yaklaşık olarak aynıdır. GKF'deki artış miktarı daha fazla iken AGKF'deki artış daha azdır. GKF'de artış ve azalış şekli daha keskindir. AGKF zaman adımlarında hata oranını değişmeden daha uzun süre korumuştur. Aynı gürültü seviyesi için robotun harita üzerinde aldığı yolun ve yer işaretleri tahmininin gösterimi Şekil 4.6 ve 4.7'de verilmiştir. İki haritaya da bakıldığında hem robot konumu hem de işaretlerinin tahmininde AGKF daha başarılı olmuştur.



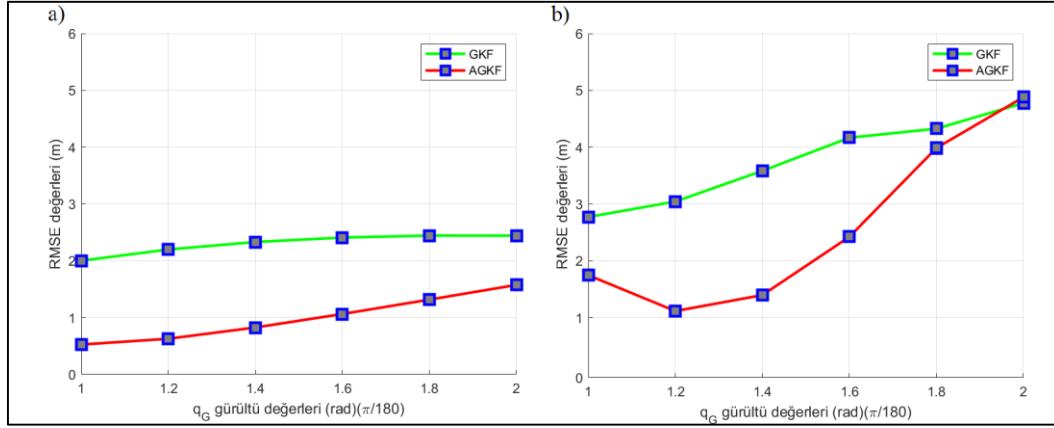
Şekil 4.6. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF



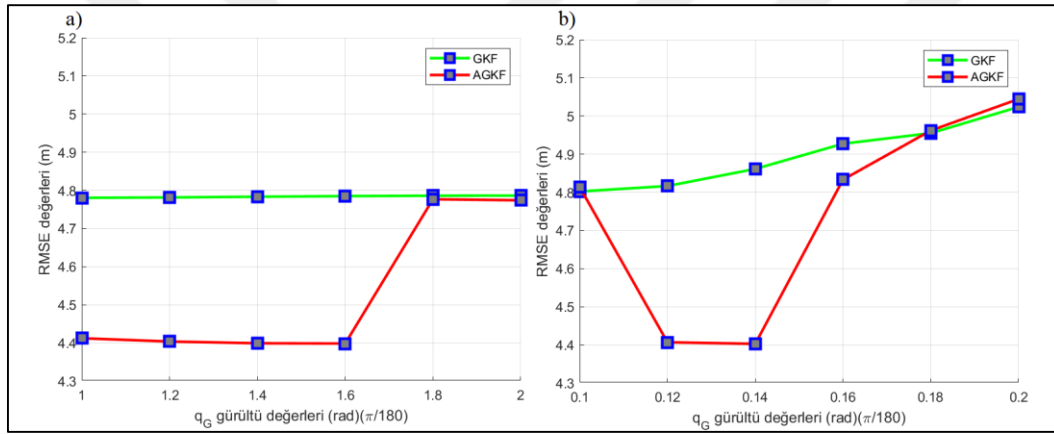
Şekil 4.7. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF

4.1.2. İşlem gürültü bileşeni q_G 'nin değişimi

Robotun harita üzerinde belirlenen yol noktaları arasında geçişler yapabilmesi için yönünü değiştirmesi gerekir. Bu yön değişimi robotun direksiyon açısının değişimi ile mümkündür. Hızda olduğu gibi robot verilen açı değişim komutunu tam olarak uygulayamaz ve q_G gürültü bileşeni kadar bir sapma ile komutu uygulayabilir. q_G 'nin başlangıç değerleri Çizelge 4.1 Gürültü Testi – 2'deki belirtilen aralıklarda değiştirken robotun veri ilişkilendirmesine sahip olduğu ve olmadığı durumların robotun konumunun ve yer işaretlerinin tahminindeki etkisi incelenmiştir. q_G 'nin bu aralıktaki her değeri için hesaplanan RMSE değerleri Şekil 4.8 ve 4.9'da verilmiştir. Hesaplanan bu değerlerin ortalaması Çizelge 4.3'te gösterilmiştir.



Şekil 4.8. q_G 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durumda b) veri ilişkilendirmesi bilinmediği durumda



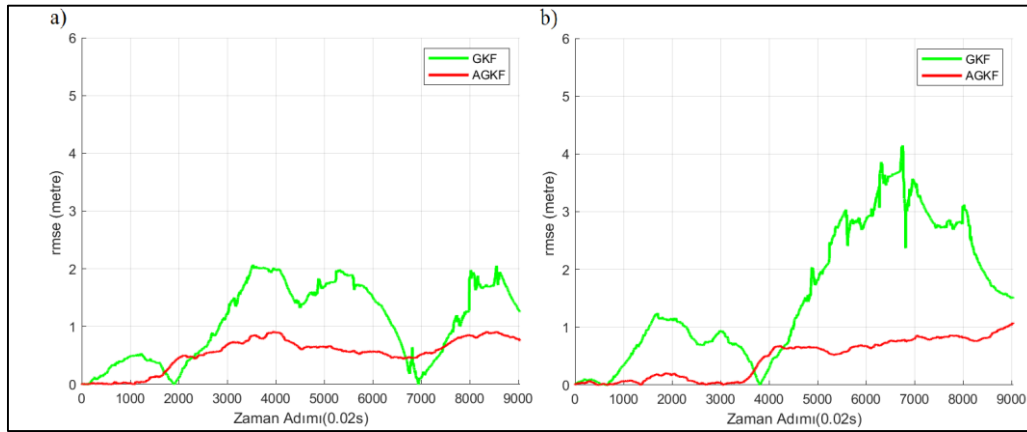
Şekil 4.9. q_G 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiğinde b) veri ilişkilendirmesi bilinmediğinde

Çizelge 4.3. Robot ve yer işareti konum hatalarının değişen q_G gürültü başlangıç değerlerindeki ortalama hata değerleri

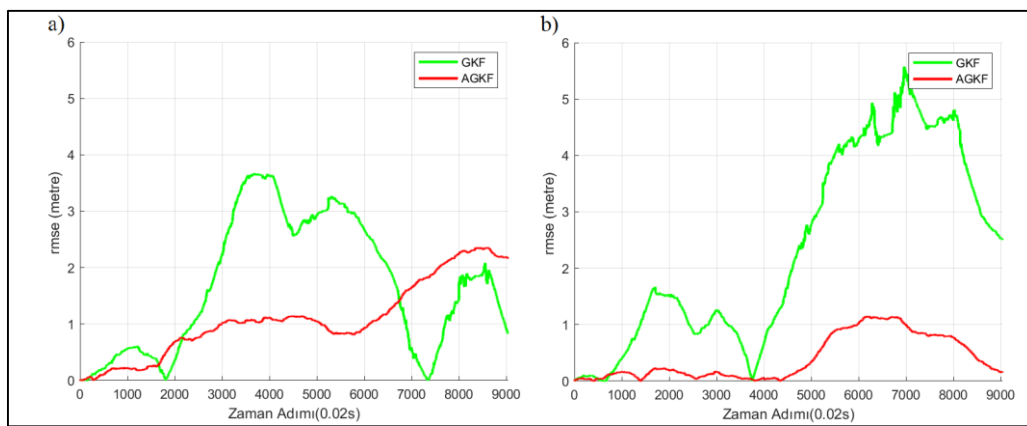
	GKF		AGKF	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
Robot konumunun hata değeri (m)	4,4521	6,6166	1,9708	3,7451
Yer işaretinin konumunun hata değeri (m)	4,7837	4,8977	4,5274	4,7439

q_G 'nin başlangıç değerindeki artışın q_V 'ye kıyasla robot konum tahminini daha fazla etkilediği Şekil 4.8.(a) ve 4.8.(b)'de görülmektedir. Özellikle veri ilişkilendirmesinin verilmediği durumda q_G 'nin artışı hata oranındaki artışı hızlandırmıştır. $q_G = [1,0; 1,6]$

aralığındayken AGKF'nin gürültü değişimine daha rahat adapte olabildiği Şekil 4.8(b) ve Şekil 4.9(b)'de verilen hem robot konumu hem de yer işareti tahminindeki hata oranları ile kolaylıkla görülebilir. Veri ilişkilendirmesi hareket öncesinde robota sağlandığında hem yer işaretlerinin hem de robot konumunun tahmininin her iki filtre tarafından daha başarılı şekilde yapıldığı saptanmıştır. Şekil 4.8 ve Şekil 4.9'daki RMSE değerlerinin ortalamaları Çizelge 4.3'te verilmiştir ve AGKF robot konumu ve yer işaretleri tahmininde GKF'ye kıyasla veri ilişkilendirmesinin iki durumu için de daha az hatalı tahminlerde bulunmuştur. Robotun hareketi boyunca $q_G = 1,6\pi/180$ için hata değerleri hesaplanmış ve her zaman adımı için Şekil 4.10 ve 4.11'de verilmiştir.



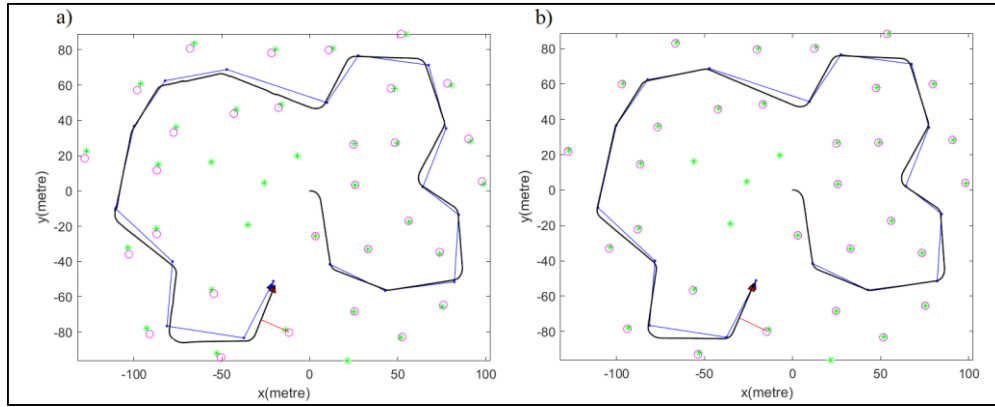
Şekil 4.10. Veri ilişkilendirmesi verildiği durumda, $q_G = 1,6\pi/180$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni



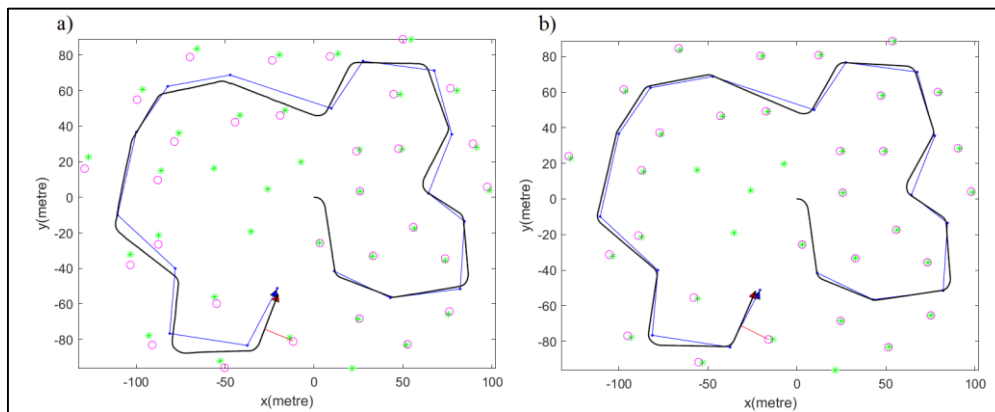
Şekil 4.11. Veri ilişkilendirmesi verilmediği durumda, $q_G = 1,6\pi/180$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

q_V 'ye benzer şekilde tahmin hata oranlarının artmaya ve azalmaya başladığı zaman adımları iki filtre için de yaklaşık olarak aynı olmakla birlikte robot hareketini devam ettirdiği sürece

GKF tahminlerinin hata oranı sürekli olarak artmış, AGKF’de ise bu artış hem daha yavaş olmuş hem de bir noktadan sonra hata oranı sabitlenmiştir. Özellikle robotun konumunun y bileşeni tahmininin verildiği Şekil 4.10.(b) ve 4.11.(b)’de görüleceği gibi GKF’nin hata oranı oldukça fazla çıkmıştır. Şekil 4.10 ve 4.11’deki RMSE değerlerinin değişim grafiğinin tepe değerleri veri ilişkilendirmesinin verilmesi ve verilmemesi durumunda farklılık göstermiş, verilmemesi durumunda hata oranının ulaştığı maksimum değer, verilmesi durumuna göre daha fazla olmuştur. Robotun hareketi süresince bulunmuş olabileceği yerlerin tahmini konumlarının birleştirilmesi ile oluşturulan yol ve bulunduğu ortam ile ilgili gözlemleri ile oluşturduğu harita örneği Şekil 4.12 ve 4.13’te verilmiştir.



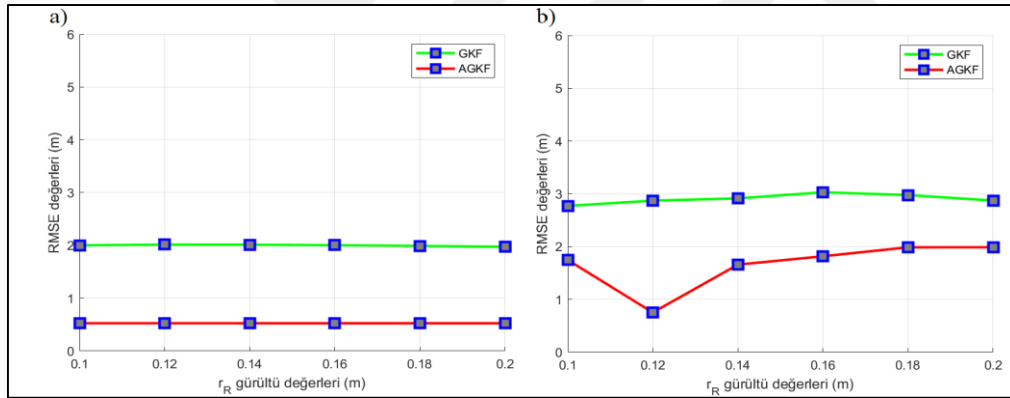
Şekil 4.12. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF



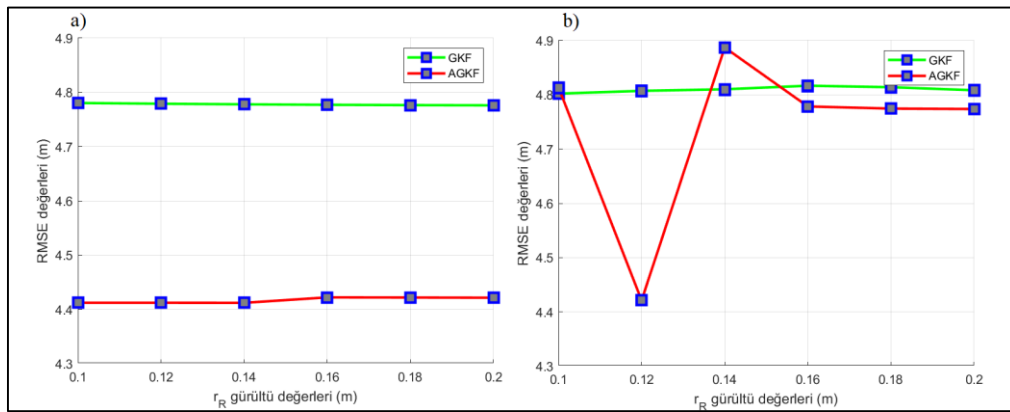
Şekil 4.13. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF

4.1.3. Ölçüm gürültü bileşeni r_R 'nin değişimi

Bölüm 4.1.1 ve 4.1.2'de işlem gürültü bileşenlerinin etkileri ayrı ayrı karşılaştırılmıştı. Bölüm 4.1.3 ve 4.1.4'te ise ölçüm gürültü bileşenlerinin robot hareketi üzerindeki etkileri incelenmiştir. Robot, üzerinde bulunan sensörler vasıtasıyla gözlemler yapar ve bu gözlemlerle elde edilen verilerden bir tanesi mesafedir. Kusursuz bir sensör hatasız bir mesafe ölçümü sunabilirken gerçek hayatta bu ölçüm bir sapma miktarı ile yapılmaktadır. Bu sapmaya sebep olan gürültü bileşeni r_R olarak gösterilir. r_R 'nin Çizelge 4.1'de Gürültü Testi – 3'e göre aldığı başlangıç değerleri kullanılarak robotun konum ve yer işareti tahminleri yapılmış ve doğru konum ile arasındaki RMSE hataları r_R 'nin her değeri için bulunmuştur. Bu hata oranları veri ilişkilendirmesinin de verilip verilmeme durumuna göre Şekil 4.14 ve 4.15'te gösterilmiştir. Verilen bu değerlerin tüm gürültü aralıkları için genel ortalaması ise Çizelge 4.4'te verilmiştir.



Şekil 4.14. r_R 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durumda b) veri ilişkilendirmesi bilinmediği durumda

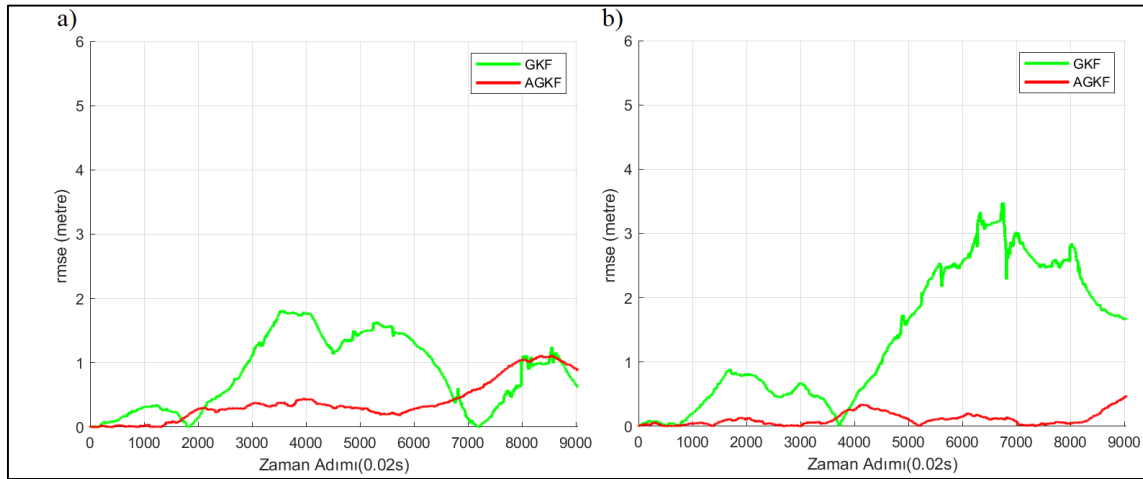


Şekil 4.15. r_R 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durumda b) veri ilişkilendirmesi bilinmediği durumda

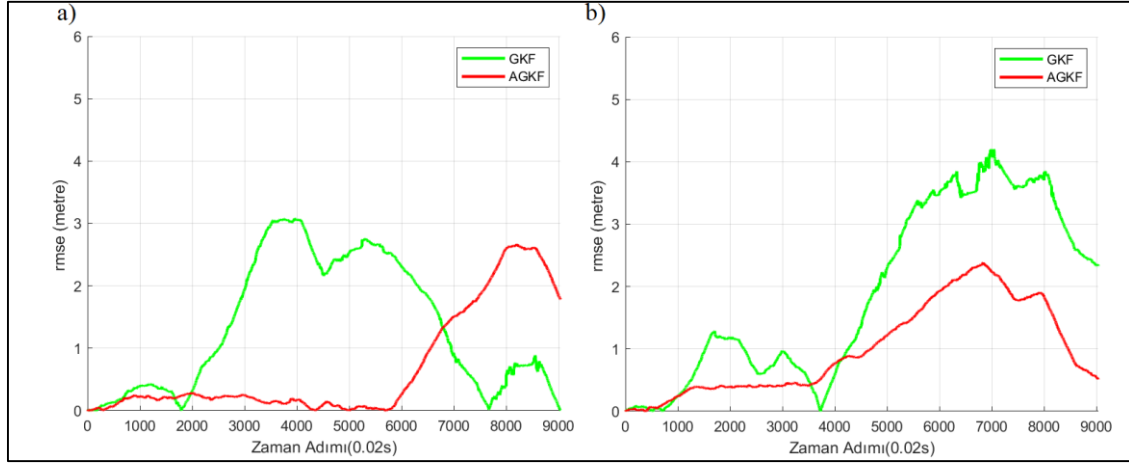
Çizelge 4.4. Robot ve yer işareti konum hatalarının değişen r_R gürültü başlangıç değerlerindeki ortalama hata değerleri

	GKF		AGKF	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
Robot konumunun hata değeri (m)	1,9999	2,9058	0,5253	1,6582
Yer işaretinin konumunun hata değeri (m)	4,7779	4,8098	4,4171	4,7415

r_R 'nin başlangıç değerinin değişimi veri ilişkilendirmesinin robota verildiği ve verilmediği durumlar için Şekil 4.14 ve 4.15'te gösterildiği gibi iki filtre için de yer işareti ve robot konumu tahmininin değişimini çok fazla etkilememiştir. Veri ilişkilendirmesi robota verildiğinde farklı r_R değerleri için robotun tahmin kabiliyetinin geliştiği görülmüştür. Veri ilişkilendirilmesinin hem verildiği hem de verilmediği durumda Çizelge 4.4'te de görülebileceği gibi AGKF, GKF'ye göre tahminlerinde daha az hata yapmıştır. Şekil 4.16 ve 4.17'de sensörün mesafe ölçümünü etkileyen gürültü bileşeni, $r_R = 0,16m$ olarak seçildiğinde robotun hareketi süresince her zaman adımında hesapladığı RMSE değerleri gösterilmiştir.

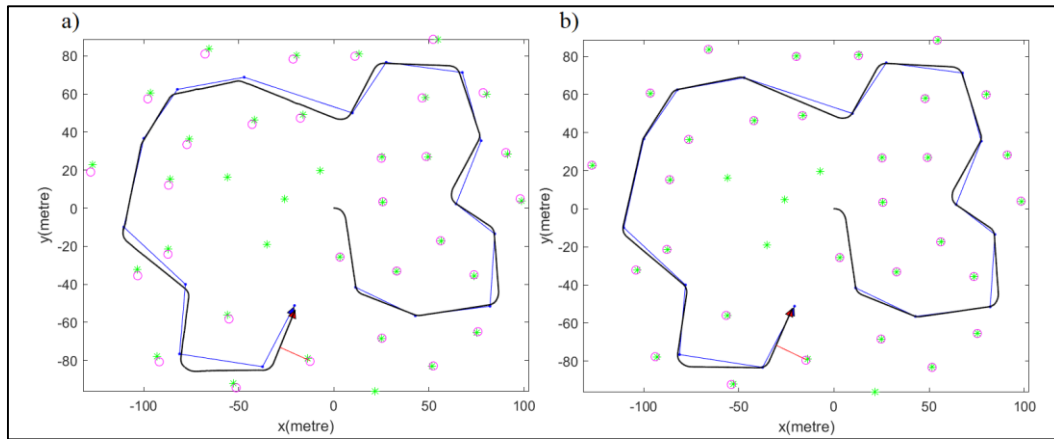


Şekil 4.16. Veri ilişkilendirmesi verildiği durumda, $r_R = 0,16m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

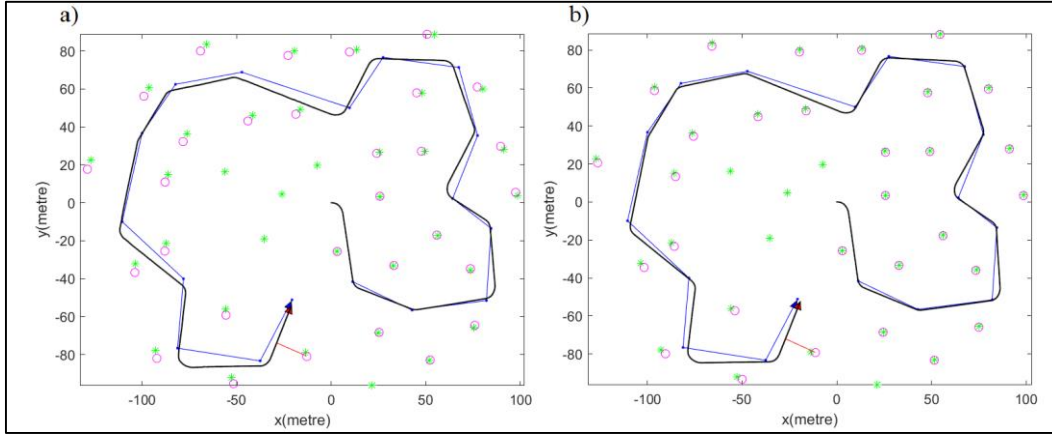


Şekil 4.17. Veri ilişkilendirmesi verilmediği durumda, $r_R = 0,16m$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

Veri ilişkilendirmesinin verildiği durumda AGKF'nin neredeyse hatasız şekilde tahminler yaptığı GKF'nin ise tahmin hatasının AGKF'ye kıyasla çok daha fazla olduğu Şekil 4.16'da görülmektedir. Veri ilişkilendirilmesi verilmediğinde ise GKF'deki hata değişiminin şekil olarak bir önceki duruma benzediği ancak tepe değerinin artırdığı, AGKF'nin hata oranında bir artış olsa bile GKF'ye göre yaptığı tahminlerde bir iyileştirme olduğu görülmektedir. Bu iyileştirme robotun hareketi boyunca konumu ve yer işaretleri için yaptığı tahminlerle oluşturduğu ve Şekil 4.18 ve 4.19'da verilen haritada da görülmektedir.



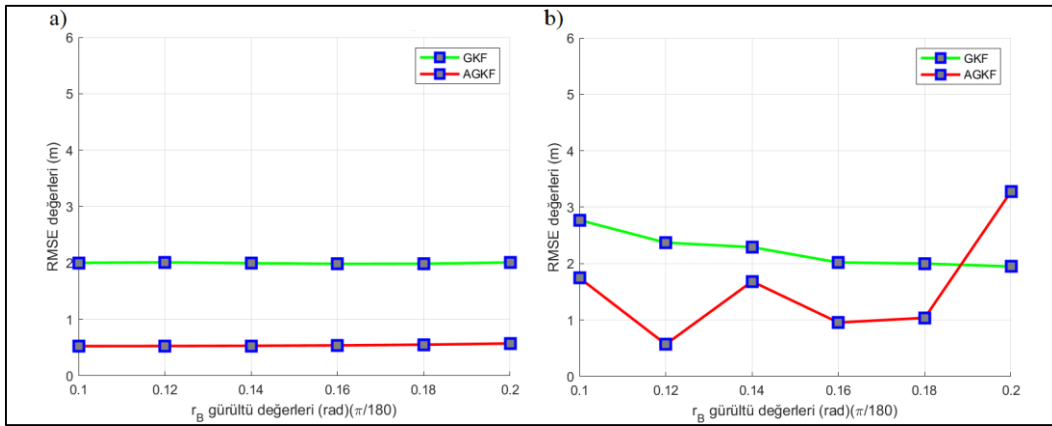
Şekil 4.18. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF



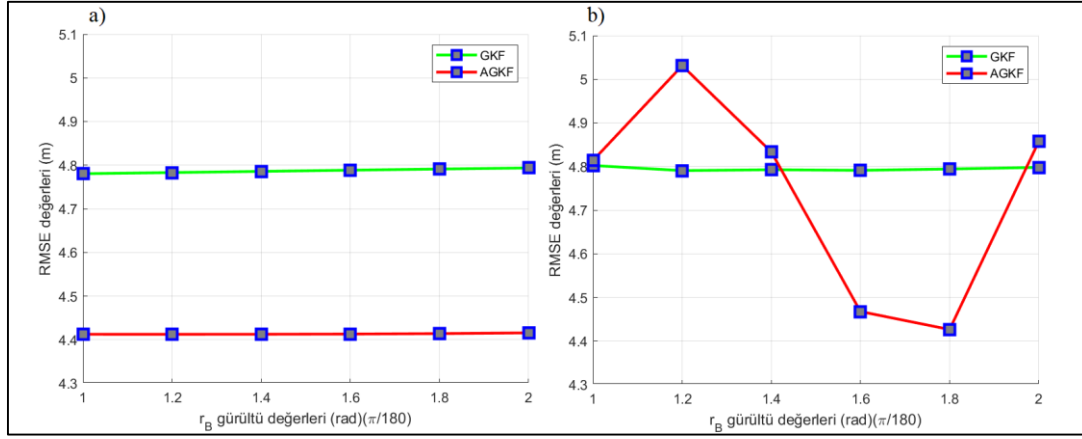
Şekil 4.19. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF

4.1.4. Ölçüm gürültü bileşeni r_B 'nin değişimi

Robot, yer işaretlerinin konumunu hareket doğrultusuna göre farklı açılar ile gözlemleyebilir. Mesafede olduğu gibi bu açılarda da gürültüye bağlı sapmalar olur. Bu sapmalara sebep olan gürültü bileşeni r_B ile ifade edilir. r_B 'nin başlangıç değerleri Çizelge 4.1'de verilen Gürültü Testlerinden 4.süne göre seçilmiş ve harekete başlarken seçilen değer ile gürültü ifade edilmiştir. Değişen başlangıç değerlerinde robotun konumu için hesaplanan RMSE tahmin hataları Şekil 4.20'de, yer işaretleri için hesaplanan RMSE hataları ise Şekil 4.21'de verilmiştir. Çizelge 4.4'te ise gürültü değişimleri süresince hesaplanan RMSE değerlerinin ortalaması gösterilmiştir.



Şekil 4.20. r_B 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durumda b) veri ilişkilendirmesi bilinmediği durumda

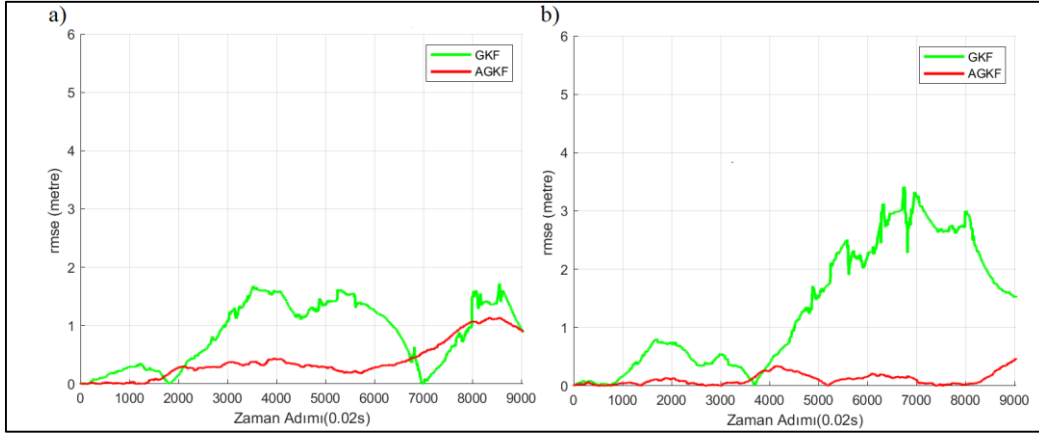


Şekil 4.21. r_B 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiği durumda b) veri ilişkilendirmesi bilinmediği durumda

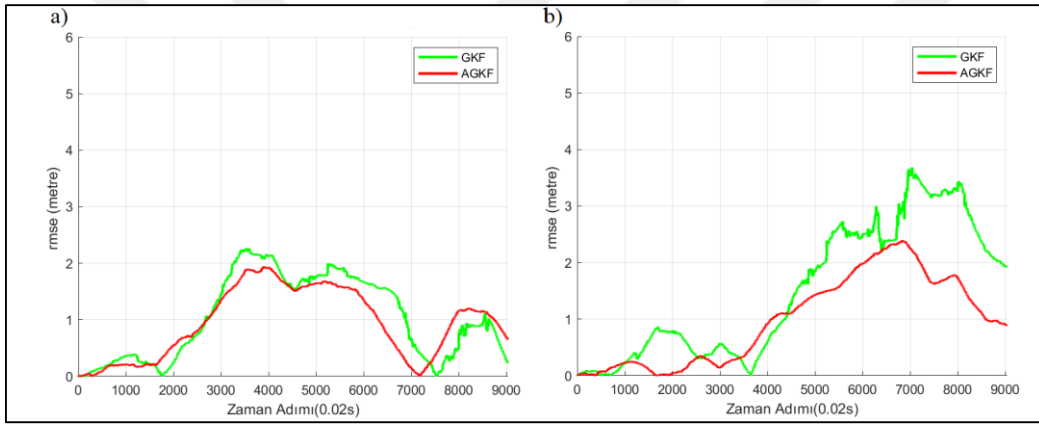
Çizelge 4.5. Robot ve yer işareti konum hatalarının değişen r_B gürültü başlangıç değerlerindeki ortalama hata değerleri

	GKF		AGKF	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
Robot konumunun hata değeri (m)	1,9980	2,2338	0,5417	1,5450
Yer işaretinin konumunun hata değeri (m)	4,7871	4,7947	4,4132	4,7382

r_B 'nin değişimi r_R 'ye benzer şekilde veri ilişkilendirilmesinin verildiği durumda hem robotun hem de yer işaretlerinin konumunun tahminlerini Şekil 4.20.(a) ve Şekil 4.21.(a)'da görüldüğü gibi hiç etkilememiş, sadece RMSE değerinin büyüklüğü değişmiştir. Veri ilişkilendirmesi robota verilmediğinde robot konumu için hesaplanan RMSE değerleri GKF için gürültü arttıkça azalırken AGKF düzensiz bir değişim göstermiştir. Yer işaretlerinde ise, AGKF bazı gürültü değerlerinde GKF'den daha doğru tahminler yaparken bazı değerlerde GKF daha iyi sonuçlar vermiştir. Çizelge 4.5'e bakıldığında veri ilişkilendirmesi AGKF'yi GKF'den daha çok etkilemiş olmasına karşın AGKF'nin, GKF'den daha başarılı tahminlerde bulunduğu görülmüştür. $r_B = 1,6 \pi/180$ rad olarak robota başlangıçta verildiğinde bırakıldığı ortamda hareketi boyunca tüm zaman adımlarında hesaplanan RMSE değerleri Şekil 4.22 ve 4.23'te gösterilmiştir.

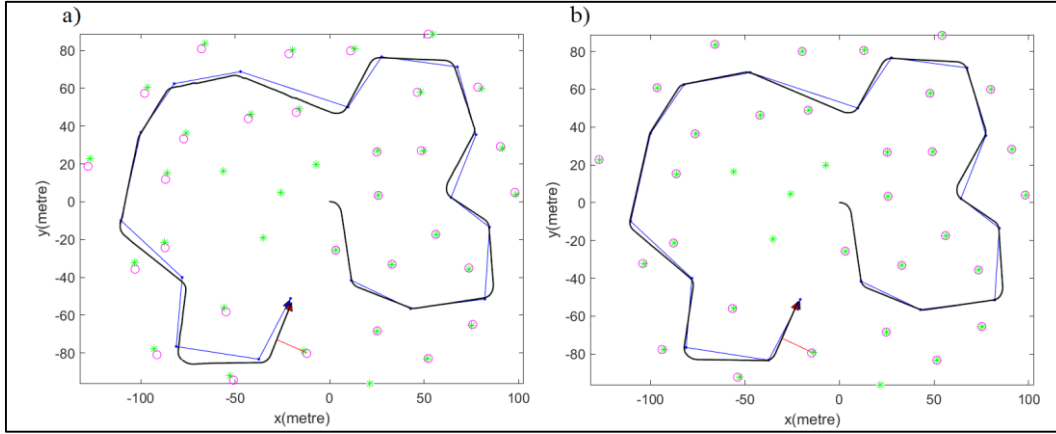


Şekil 4.22. Veri ilişkilendirmesi verildiği durumda, $r_B = 1,6 \pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

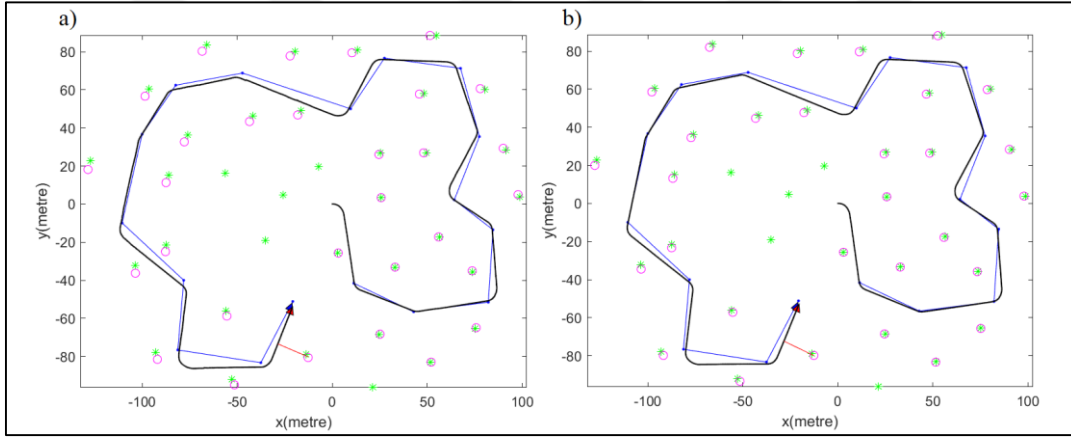


Şekil 4.23. Veri ilişkilendirmesi verilmediği durumda, $r_B = 1,6 \pi/180 \text{ rad}$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

Veri ilişkilendirilmesinin verildiği Şekil 4.22'ye göre robotun hareketi boyunca AGKF, GKF'ye göre çok daha iyi tahminlerde bulunmuştur. Veri ilişkilendirmesi robota verilmediği durumda ise zamanla değişen RMSE hatası GKF için çok fazla değişmemiş, sadece tepe değerinde bir miktar artış olmuştur. AGKF ise Şekil 4.23'te görüleceği üzere veri ilişkilendirmesinden fazlasıyla etkilenmiş fakat RMSE değeri yine de GKF'den daha düşük olmuştur. Veri ilişkilendirmesinin bu etkisi Şekil 4.24 ve 4.25'te gösterilmiş haritalar üzerinde de kendisini belli etmektedir. Şekil 4.24.(b)'de AGKF tarafından oluşturulmuş haritadaki iyileşme oranı Şekil 4.25(b)'de olandan çok daha fazladır.



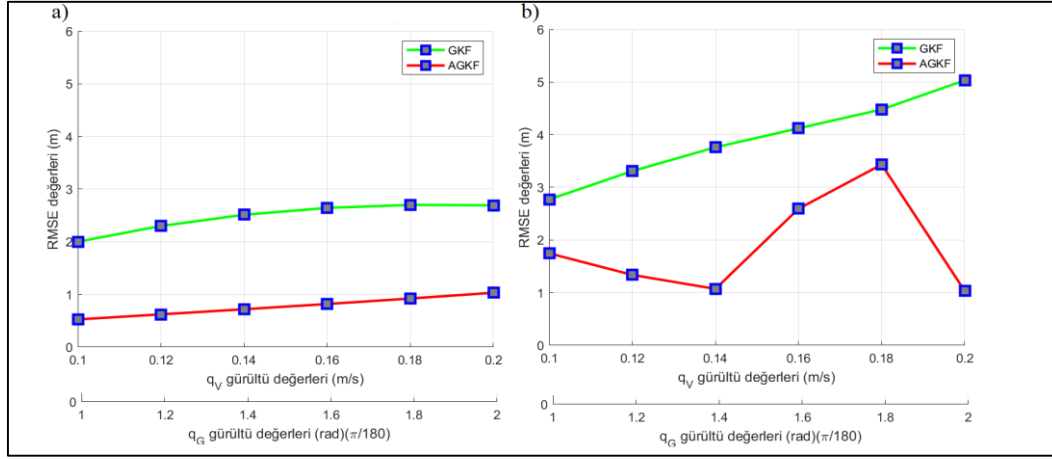
Şekil 4.24. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF



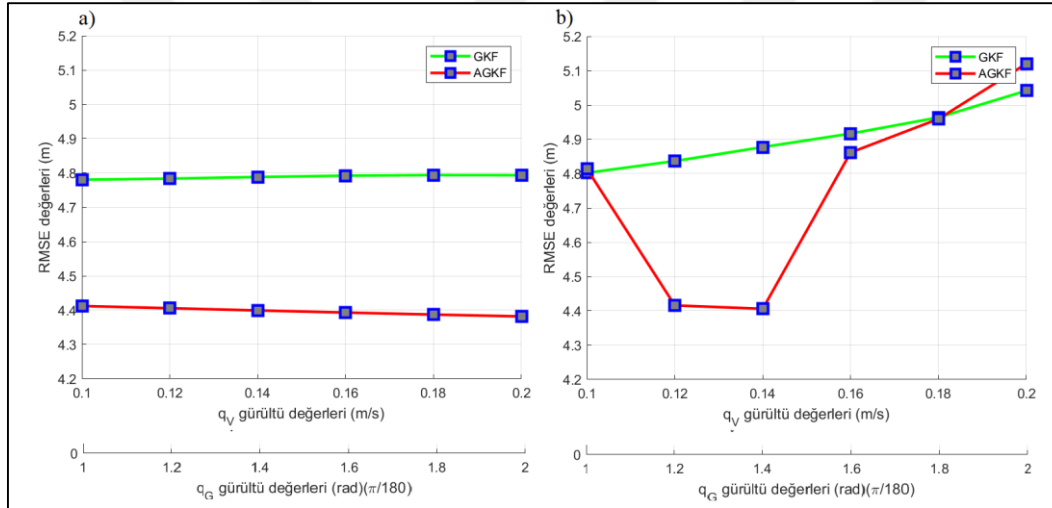
Şekil 4.25. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF

4.1.5. İşlem gürültü bileşenleri q_v ve q_G 'nin değişimi

Tüm gürültü bileşenlerinin harekete olan etkisi Bölüm 4.1.1 - 4.1.4 arasında yapılan testlerde anlatılmış ve filtrelerin çalışmasındaki etkileri gösterilmiştir. Bu bölümde Bölüm 4.1.1 ve 4.1.2'de ayrı ayrı incelenmiş olan işlem gürültü bileşenlerinin Çizelge 4.1, Gürültü Testi – 5'te verilen başlangıç değerlerine göre yapılan tahminler ve oluşturulan haritalar üzerindeki etkileri, veri ilişkilendirmesinin durumuna göre incelenmiş ve elde edilen sonuçlar Şekil 4.26 ve Şekil 4.27'de verilmiştir. Bu şekillerde elde edilen RMSE değerlerinin genel ortalaması Çizelge 4.5'te gösterilmiştir.



Şekil 4.26. q_v ve q_G 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiğinde b) veri ilişkilendirmesi bilinmediğinde

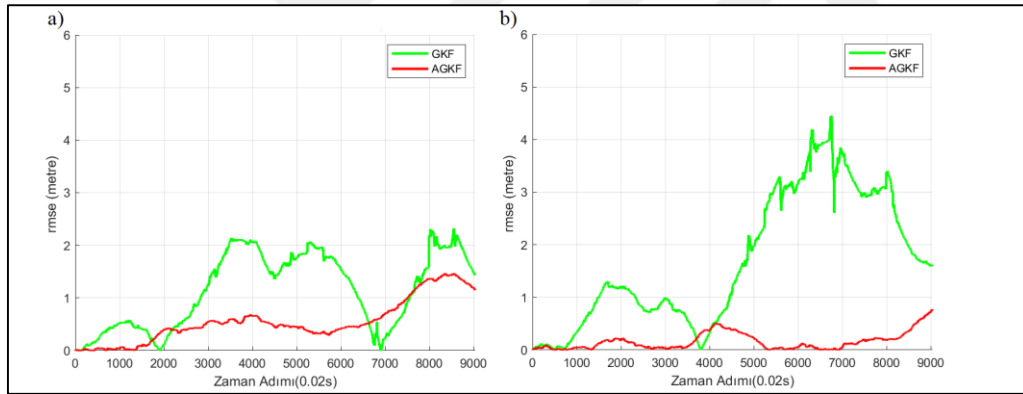


Şekil 4.27. q_v ve q_G 'nin değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiğinde b) veri ilişkilendirmesi bilinmediğinde

Çizelge 4.5. Robot ve yer işareti konum hatalarının değişen q_v ve q_G gürültü başlangıç değerlerindeki ortalama hata değerleri

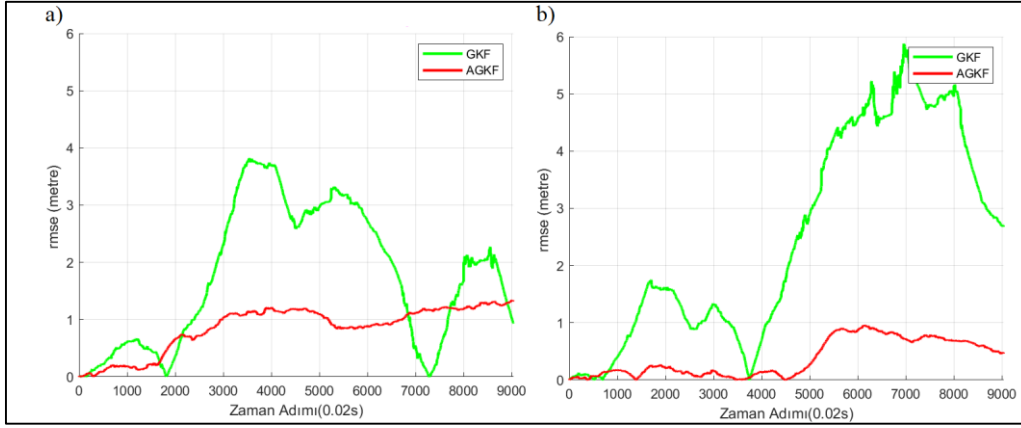
	GKF		AGKF	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
Robot konumunun hata değeri (m)	2,4757	3,9126	0,7736	1,8692
Yer işaretinin konumunun hata değeri (m)	4,7886	4,9064	4,3967	4,7629

Çizelge 4.6’da verilen ortalama RMSE değerlerine göre q_v ve q_G değişimi AGKF’yi GKF’den daha çok etkilemiştir. Yine de AGKF’nin tahmin doğruluğu GKF’den daha iyidir. Bu gürültü değerleri beraber değiştiğinde önceki durumlara kıyasla veri ilişkilendirmesinin bilindiği ve bilinmediği durumda robot konumundaki hatanın GKF için artış miktarında bir yükselme olduğu Şekil 4.26’da görülmektedir. Bu artışta en büyük payın q_G ’e ait olduğu yorumu Şekil 4.8’e bakılarak yapılabilir. Robot konumunun tahmininde q_v ve q_G değişiminde AGKF’nin gürültüye adapte olduğu yerlerde hataların çok azaldığı ve GKF’ye göre çok daha başarılı sonuçlar verdiği Şekil 4.26.(b)’de görülmektedir. Yer işaretlerinin tahmininde ise veri ilişkilendirmesinin verildiği durumda bir değişim olmazken hem verildiği hem de verilmediği durumda Şekil 4.27’ye göre AGKF’nin hata oranı GKF’den daha azdır. Robot konumu için hesaplanan RMSE değerlerinin $q_v = 0,16m$ ve $q_G = 1,6\pi/180$ başlangıç değerlerindeyken zaman adımına göre değişimi Şekil 4.28 ve 4.29’de verilmiştir.

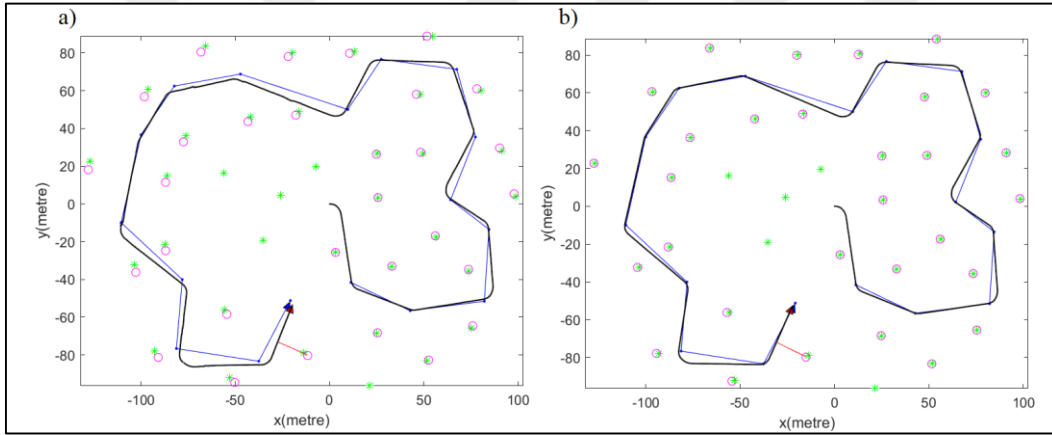


Şekil 4.28. Veri ilişkilendirmesi verildiği durumda, $q_v = 0,16m$ ve $q_G = 1,6\pi/180$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni
b) y bileşeni

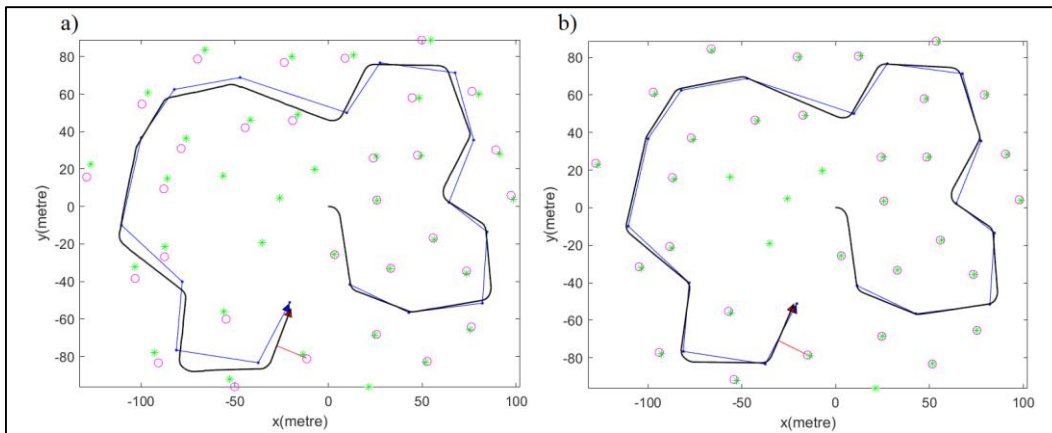
Şekil 4.28 ve Şekil 4.29’a bakıldığında GKF’nin tahmin kalitesi AGKF’ye göre çok daha düşüktür. GKF’de veri ilişkilendirilmesinin verilme durumuna göre RMSE değerinin ulaştığı maksimum noktasında iki kata yakın bir değişim görülürken AGKF her iki durumda da daha tutarlı ve yüksek bir tahmin kalitesine sahiptir. Robotun zaman adımlarında yaptığı konum yer işareti tahminleri ile oluşturduğu haritalar da bu kalite farkını göstermektedir. Hem Şekil 4.30’da hem de Şekil 4.31’de AGKF’nin oluşturduğu harita GKF’ye kıyasla oldukça başarılıdır.



Şekil 4.29. Veri ilişkilendirmesi verilmediği durumda, $q_v = 0,16m$ ve $q_G = 1,6 \pi/180$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni



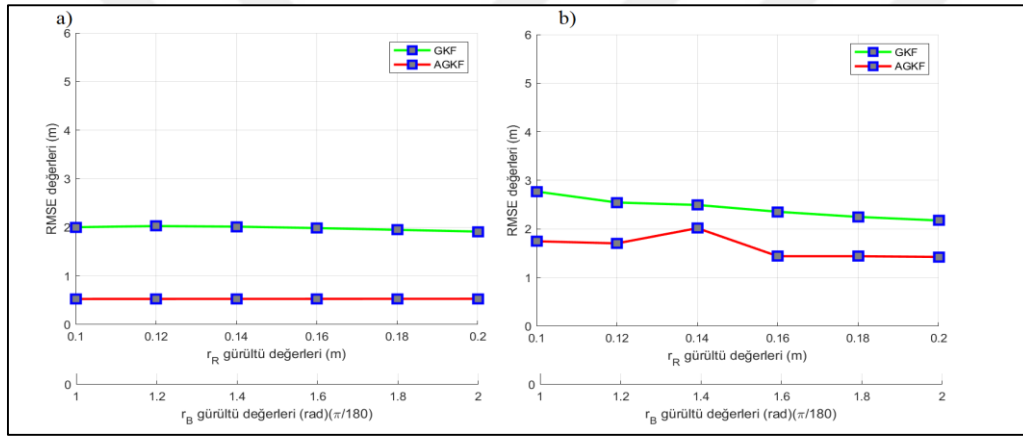
Şekil 4.30. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF



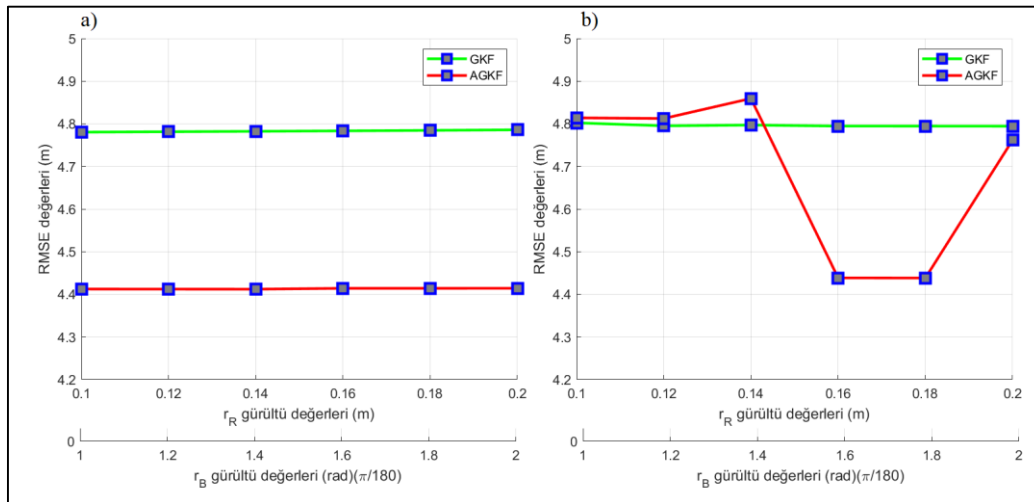
Şekil 4.31. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF

4.1.6. Ölçüm gürültü bileşenleri r_R ve r_B 'nin değişimi

İşlem gürültü bileşenlerinden sonra Bölüm 4.1.3 ve 4.1.4'te incelenen ölçüm gürültü bileşenlerinin başlangıç değerlerinin aynı anda değişimine göre GKF ve AGKF'nin yaptığı tahminlerin RMSE değerleri hesaplanmıştır. Şekil 4.32 ve 4.33'te, sırasıyla, veri ilişkilendirmesinin verildiği ve verilmediği duruma göre RMSE değerlerinin karşılaştırılması yapılmıştır. Bu karşılaştırmalar sonucu çıkan hata değerlerinin tüm gürültü değişimi sürecindeki ortalama RMSE değeri hesaplanmış ve Çizelge 4.7'ye kaydedilmiştir. Bu işlemler yapılırken robotun başlangıç gürültü değerlerinin Çizelge 4.1'de verilen testlerden Gürültü Testi – 6'daki gibi olduğu varsayılmıştır.



Şekil 4.32. r_R ve r_B 'nin değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiğinde b) veri ilişkilendirmesi bilinmediği

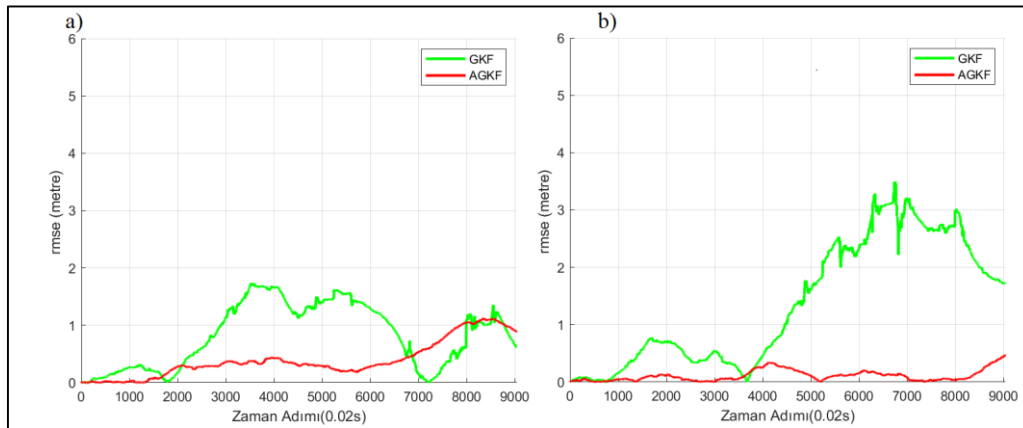


Şekil 4.33. r_R ve r_B 'nin değişimine göre yer işaretleri konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi bilindiğinde b) veri ilişkilendirmesi bilinmediği

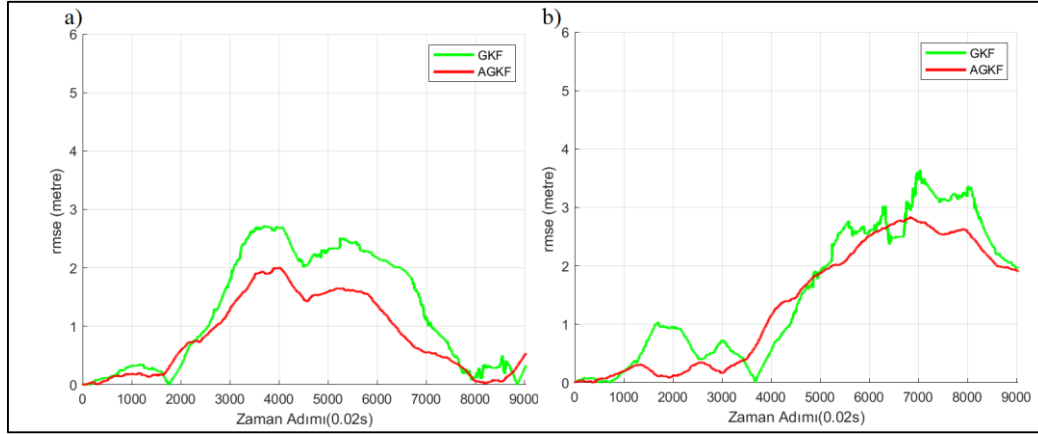
Çizelge 4.7. Robot ve yer işareti konum hatalarının değişen r_R ve r_B gürültü başlangıç değerlerindeki ortalama hata değerleri

	GKF		AGKF	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
Robot konumunun hata değeri (m)	1,9816	2,4313	0,5275	1,6287
Yer işaretinin konumunun hata değeri (m)	4,7830	4,7963	4,4131	4,6873

Robot harita oluştururken ve konumunu tahmin ederken veri ilişkilendirmesi bilgisine önceden sahip ise r_R ve r_B 'nin farklı başlangıç değerlerinin tahmin doğruluğuna hiçbir etkisinin olmadığı Şekil 4.32.(a) ve 4.33.(a)'de görülmektedir. Çizelge 4.7'ye göre AGKF'nin tahmin kalitesi GKF'ye göre daha yüksektir. Veri ilişkilendirmesi bilgisi robota verilmediğinde ise tahmin hatalarının ortalama değerleri her gürültü değeri için artış göstermiş, buna karşılık gürültü değerleri arttıkça her iki filtre için de hata oranlarının azaldığını gözlemlenmiştir. Bu durum Şekil 4.32(b) ve 4.33(b)'de görselleştirilmiştir. Veri ilişkilendirmesinin iki durumu için de gürültü değerleri arttıkça GKF'nin hata oranı AGKF'den fazla bulunmuştur. Robotun harita üzerinde aldığı bir tur boyunca başlangıç gürültü değerleri $r_R = 0,16m$ ve $r_B = 1,6 \pi/180$ olduğunda zaman adımına göre tahmin hatasının değişim grafiği Şekil 4.34 ve 4.35'te verilmiştir.

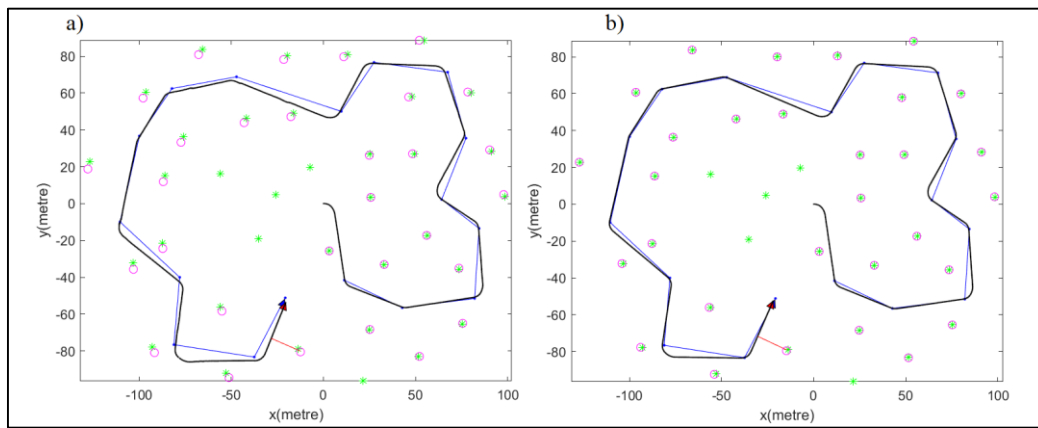


Şekil 4.34. Veri ilişkilendirmesi verildiği durumda, $r_R = 0,16m$ ve $r_B = 1,6 \pi/180$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni
b) y bileşeni

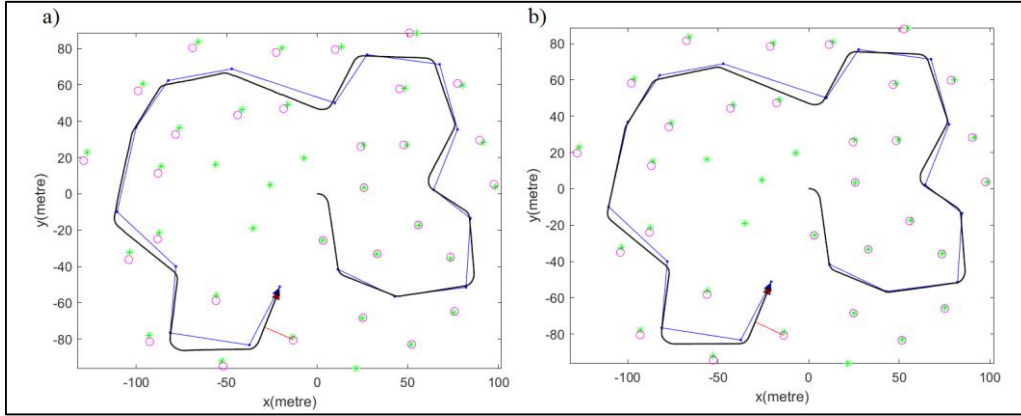


Şekil 4.35. Veri ilişkilendirmesi verilmediği durumda, $r_R = 0,16\text{m}$ ve $r_B = 1,6 \pi/180$ iken her zaman adımında robot konumunun RMSE değerinin değişimi. a) x bileşeni b) y bileşeni

İki durumda da AGKF GKF'den daha iyi sonuçlar vermiş ve özellikle Şekil 4.35'te veri ilişkilendirmesinin verildiği durumda robotun (x,y) konumu için AGKF'nin tahminleri GKF'ye göre oldukça üstün gelmiştir. AGKF GKF'ye kıyasla veri ilişkilendirmesi durumundan daha çok etkilenmiş ve veri ilişkilendirmesi verilmediğinde ise tahmin hata oranları birbirine yaklaşmıştır. Bu yakın tahminlerde bulunma durumu Şekil 4.37'de çizilmiş olan haritada görülebilir. Aynı zamanda veri ilişkilendirilmesinin verildiği durumda robotun gerçek konumuna fazlasıyla yakın tahminde bulunarak oluşturulmuş harita Şekil 4.36'da verilmiştir.



Şekil 4.36. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF



Şekil 4.37. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) GKF b) AGKF

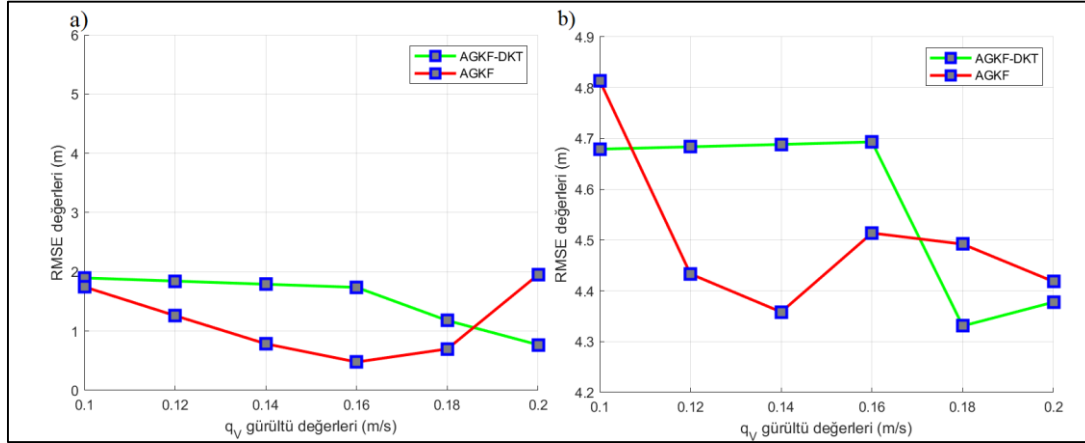
4.2. AGKF ile AGKF-DKT'nin Karşılaştırılması

Bölüm 4.1 altında yapılan karşılaştırmalarda GKF'nin gürültü seviyesine göre adaptasyonu ile oluşturulmuş olan AGKF yaklaşımının tahminler noktasında oldukça iyileştirilmiş sonuçlar verdiği görülmüştür. AGKF'nin en büyük riski başlangıçta pozitif tanımlı olarak kabul edilen Q ve R gürültü kovaryans matrislerinin filtre çalıştıkça bu özelliğini kaybetme olasılığının olmasıdır. Bu matrislerin pozitif tanımlı kalabilmeleri için önerilen DKT yaklaşımı Bölüm 3.3.5'te anlatılmıştır. Bu bölümde DKT yaklaşımının AGKF'ye etkisi Çizelge 4.1'de verilen testler altında incelenmiştir. Bu testler yapılırken bir önceki bölümde, veri ilişkilendirmesinin verildiği durumların verilmediği durumlara karşı tahmin kalitesinde daha üstün olduğunun görüldüğü ve DKT'nin Eş. 3.68'de görüldüğü gibi gözlem tahmini ile gerçek konumu arasındaki farkın bir fonksiyonu olması sebebiyle veri ilişkilendirmesinin robota daha önceden verilmediği durumlar karşılaştırılmıştır. Veri ilişkilendirmesi robota verildiğinde DKT etkisi gözlenmemiştir. Bununla ilgili elde edilmiş grafikler ve haritalar Eklerde paylaşılmıştır.

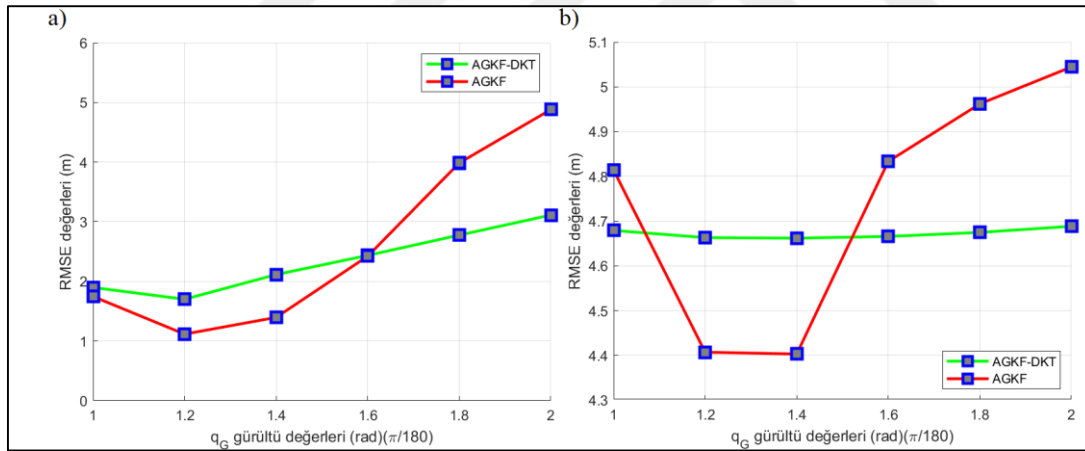
4.2.1. İşlem gürültü bileşenlerinin değişimi

AGKF ile DKT'nin AGKF içinde kullanımının değişen gürültü değerlerine göre kıyaslaması ilk olarak işlem gürültü bileşenleri için yapılmıştır. Çizelge 4.1'e göre robot sırasıyla Gürültü Testi – 1,2 ve 5'te verilen q_v ve q_G 'nin farklı başlangıç değerleri için harita içerisinde gezdirilmiş ve hem kendi konumu hem de yer işaretlerinin konumları için tahminlerde

bulunmuştur. Daha sonra bu tahminler ile doğru konumlar karşılaştırılmış, değişen her gürültü değeri için RMSE değerleri hesaplanmıştır. Hesaplanan bu değerler Şekil 4.38-4.40'ta gösterilmiştir.



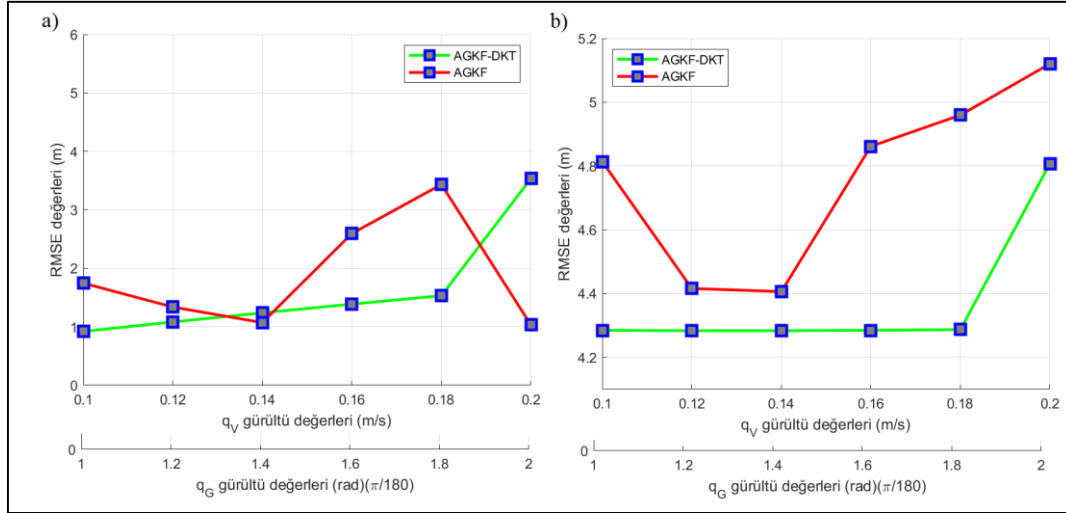
Şekil 4.38. q_v değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu



Şekil 4.39. q_G değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu

Çizelge 4.8. Robot ve yer işareti konum hatalarının değişen işlem gürültüsü başlangıç değerlerindeki ortalama hata değerleri

	AGKF			AGKF-DKT		
	Sadece q_v değişirken	Sadece q_G değişirken	q_v ve q_G değişirken	Sadece q_v değişirken	Sadece q_G değişirken	q_v ve q_G değişirken
Robot konumunun hatası (m)	1,1537	2,5914	1,8692	1,5365	2,3394	1,6158
Yer işaretinin konumu hatası (m)	4,5049	4,7439	4,7629	4,5754	4,5283	4,3718



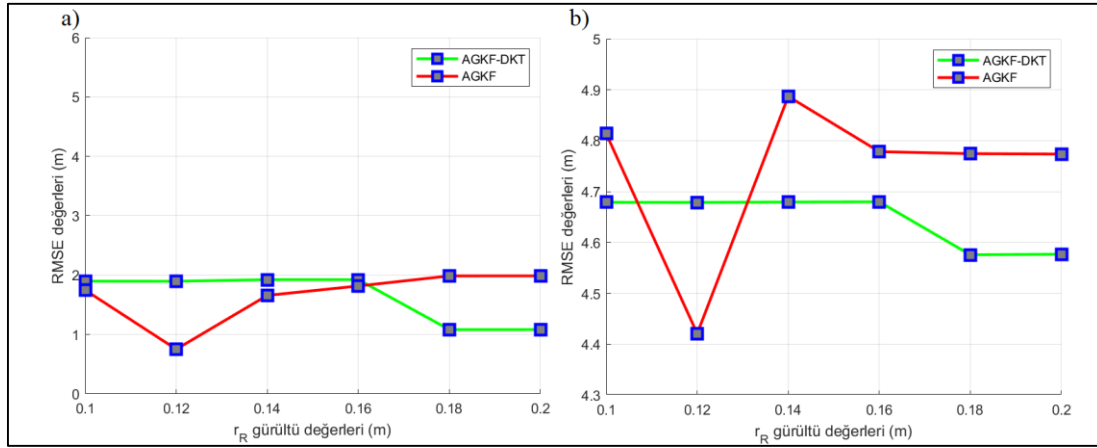
Şekil 4.40. q_v ve q_G değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu

DKT, AGKF'ye uygulandığında hem yer işaretlerinin hem de robotun konumundaki tahminlerinde iyileştirmeler sağladığı, tahmin kalitesini artırdığı gözlemlenmiştir. Yer işaretleri tahmininde q_v ve q_G 'nin beraber değiştiği durumda Şekil 4.40.(b)'de de görülebileceği gibi yapılan hata miktarı azalmıştır. Aynı zamanda AGKF-DKT'nin değişen gürültü seviyesinden etkilenme oranı DKT olmadan önceki duruma göre daha azdır. Şekil 4.38-4.40 arasında elde edilen RMSE değerlerinin belirtilen test aralığında uygulanan tüm gürültü değerlerine göre ortalaması alınmış ve Çizelge 4.8 oluşturulmuştur.

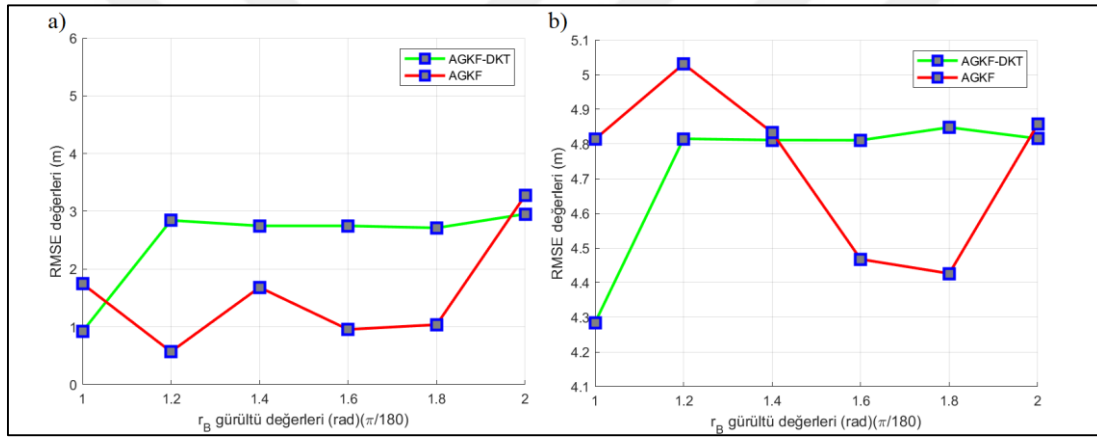
Çizelge incelendiğinde sadece q_v 'nin değiştiği durum hariç tüm işlem gürültüsü değişimlerinin ortalama hata RMSE'leri incelendiğinde DKT'nin eklenmesi AGKF'nin pozitif tanımlılığını korumakla birlikte tahmin kalitesini de güçlendirmiştir.

4.2.2. Ölçüm gürültü bileşenlerinin değişimi

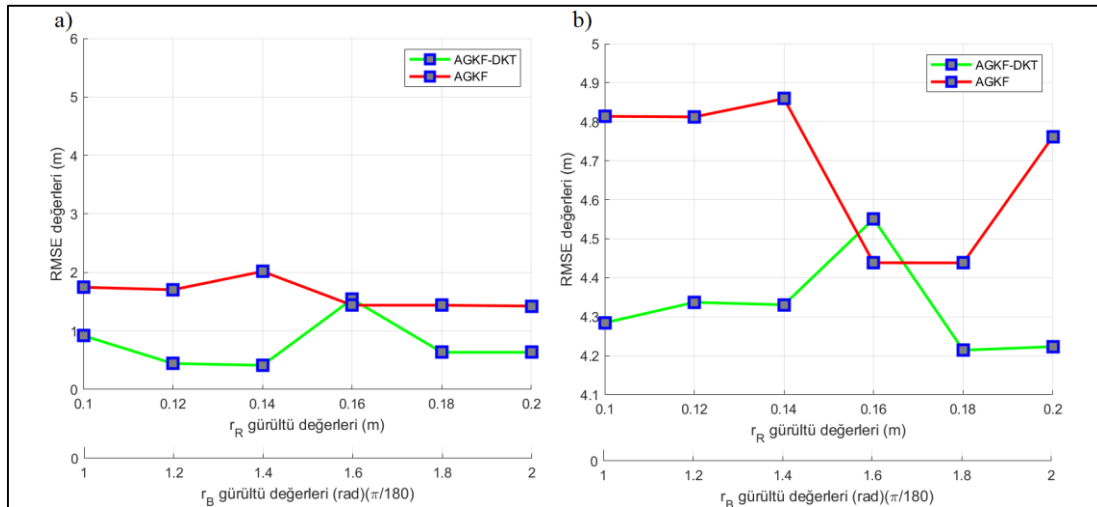
Gürültü karşılaştırmasında ikinci gürültü bileşeni olan ölçüm gürültüsünün başlangıç değerleri bu başlık altında hem AGKF'de hem de AGKF'nin içinde DKT'nin eklendiği durumda değiştirilerek incelenmiş, elde edilen RMSE değerleri Şekil 4.41-4.43'te gösterilmiştir. Ölçüm gürültüleri Çizelge 4.1'den seçilmiş ve sırasıyla Gürültü Testi – 3,4 ve 6 robota başlangıç gürültü seviyesi olarak uygulanmıştır. Seçilen bu testler boyunca hesaplanan RMSE değerlerinin ortalamaları bulunarak Çizelge 4.9'a kaydedilmiştir.



Şekil 4.41. r_R değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu



Şekil 4.42. r_B değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu



Şekil 4.43. r_R ve r_B değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu

Şekil 4.41.(a)'da görülebileceği gibi r_R 'nin aldığı farklı başlangıç değerlerine göre robot konum tahmininde DKT, AGKF'nin tahmin kalitesini iyileştirmiştir. Yer işaretleri tahmininde ise Şekil 4.41.(b)'de görüldüğü gibi gürültü seviyesi arttıkça DKT sayesinde AGKF'nin adaptifliği sağlaması kolaylaşmış ve tahmin hatasında azalma görülmüştür. r_B 'nin farklı değerleri için ise durum tam tersi olmuştur. Farklı r_B değerlerinde robotun ve yer işaretlerinin konumunda AGKF'nin daha az hata yaptığı Şekil 4.43'te görülmektedir. Ölçüm gürültü değerlerinin aynı anda değiştiği durumda ise DKT sayesinde tahminlerin başarısı artmış hem yer işaretlerinin hem de robot konumunun hatasında AGKF-DKT AGKF'ye kıyasla daha doğru sonuçlar vermiştir. Bulunan RMSE değerlerinin, gürültünün değiştiği aralıktaki ortalamaları alındığında, özellikle r_R ve r_B 'nin beraber değiştiği durumda DKT yaklaşımı sayesinde doğru konuma daha yakın tahminler yapılabildiği Çizelge 4.9'da görülmektedir.

Çizelge 4.9. Robot ve yer işareti konum hatalarının değişen ölçüm gürültüsü başlangıç değerlerindeki ortalama hata değerleri

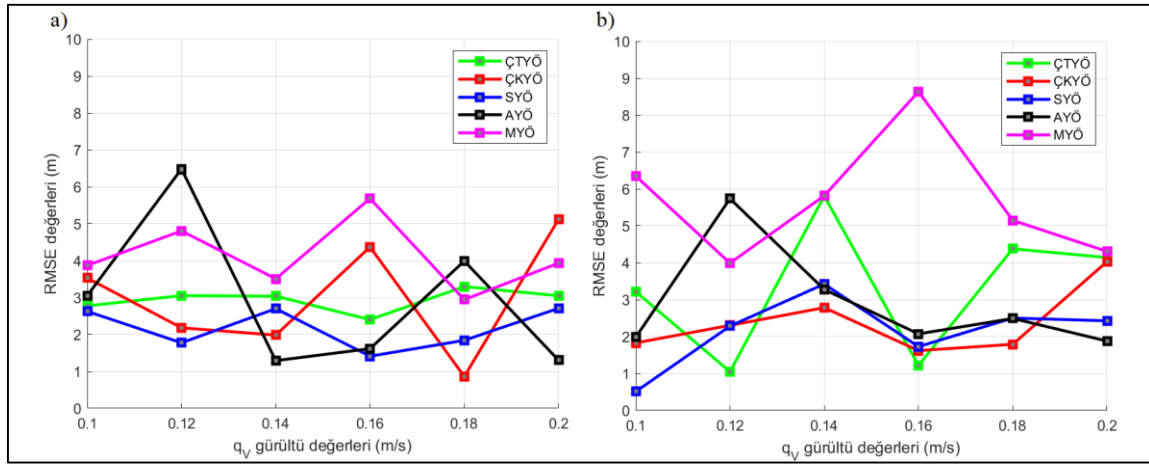
	AGKF			AGKF-DKT		
	Sadece r_R değişirken	Sadece r_B değişirken	r_R ve r_B değişirken	Sadece r_R değişirken	Sadece r_B değişirken	r_R ve r_B değişirken
Robot konumunun hatası (m)	1,6582	1,5450	1,6287	1,6347	2,4880	0,7650
Yer işaretinin konumu hatası (m)	4,7415	4,7382	4,6873	4,6451	4,7309	4,3055

4.3. FastSLAM Yeniden Örnekleme Metotlarının Karşılaştırılması

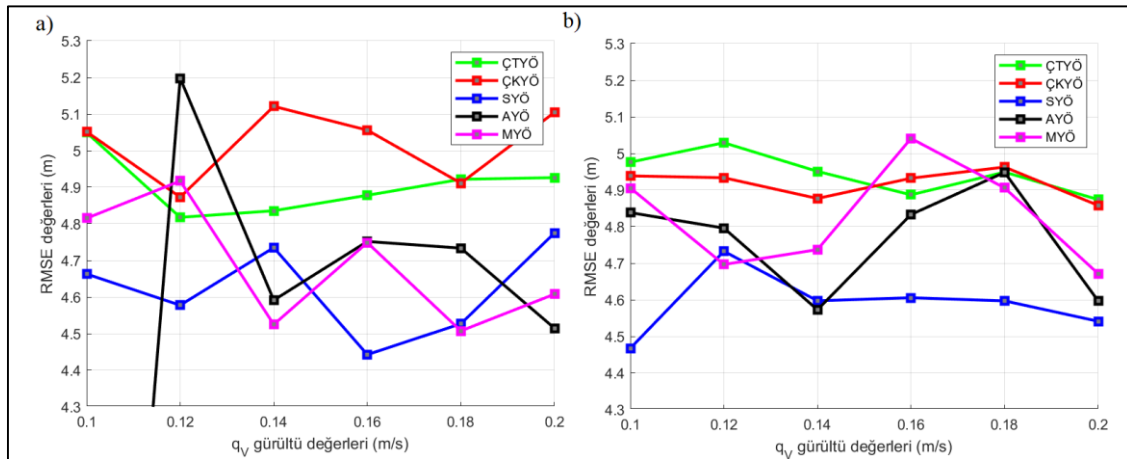
Bölüm 4.1 ve 4.2'de GKF ile GKF'nin gürültü istatistiklerine adapte olabildiği AGKF yaklaşımlarının gürültü değişimleri karşısında değişimleri karşılaştırılmıştı. Bu bölümde ise EZYTH'nin PF ile çözümünde önemli bir adım olan yeniden örnekleme metotları kendi arasında karşılaştırılmıştır. Yeniden örnekleme sayesinde parçacıkların dejenere olma ihtimali azaltılmıştır. Robotun en olası konum bilgisini taşıdığına inanılan, ağırlığı en yüksek olan parçacıklar parçacık kümesine eklenir ve bu şekilde tahmin kalitesi artırılır. Yeniden örnekleme için literatürde birçok yöntem önerilmiş ve bu yöntemlerden en çok kullanılanlar Bölüm 3.4.5'te anlatılmıştır. Yeniden örnekleme metotları Çizelge 4.1'de verilen gürültü istatistikleri başlangıç değerleri kullanılarak veri ilişkilendirmesinin verildiği ve verilmediği durumlarda karşılaştırılmıştır. Hesaplanan RMSE değerlerinin ortalamaları Çizelge 4.10-4.15 arasında gösterilmiştir.

4.3.1. İşlem gürültü bileşeni q_v 'nin değişimi

Yeniden örnekleme yöntemleri işlem gürültüsü bileşeni q_v 'nin değişen başlangıç değerleri için FastSLAM algoritması içinde test edilmiş, robot ve yer işaretleri konumlarının doğru konumları ile tahmin edilen konumları arasındaki hata RMSE değerleri hesaplanmıştır. RMSE değerlerinin değişen robot gürültüsüne göre değişimi Şekil 4.44 ve Şekil 4.45'te verilmiştir. Bu testler yapılırken veri ilişkilendirmesinin robota hareket öncesi verildiği ve verilmediği durumlar göz önünde bulundurulmuştur.



Şekil 4.44. q_v değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

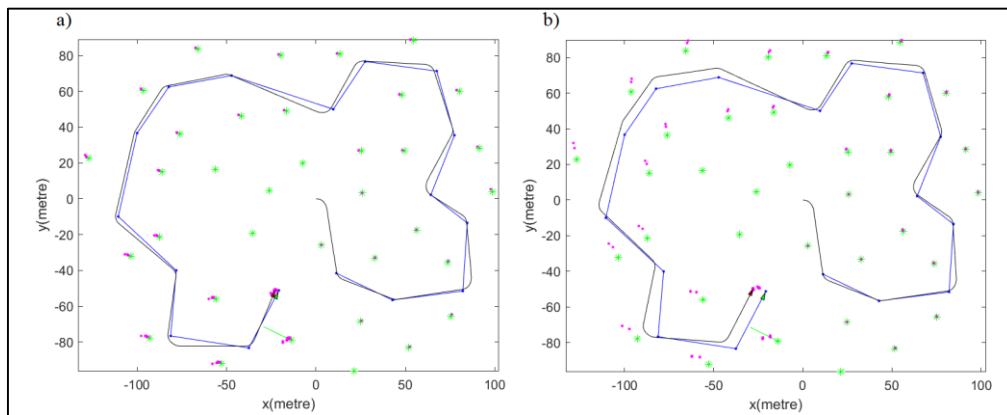


Şekil 4.45. q_v değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

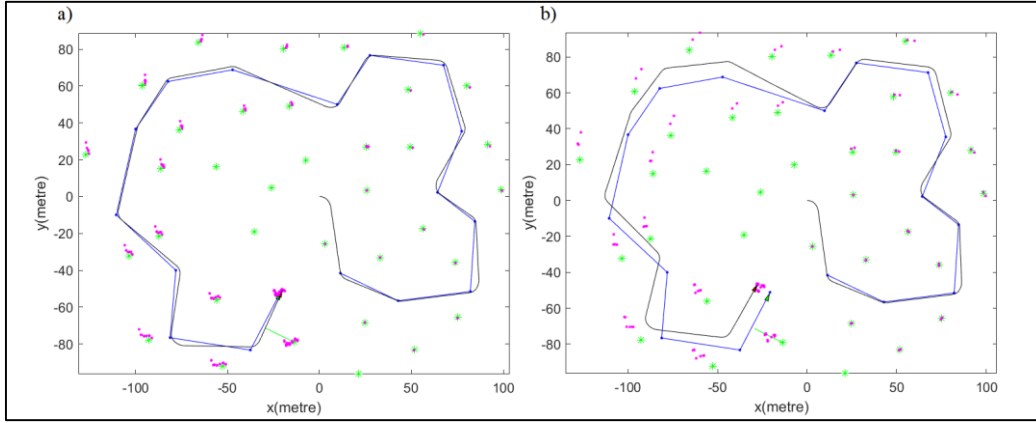
Çizelge 4.10. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen q_v gürültü başlangıç değerlerindeki ortalama hata değerleri

	Robot konumunun hata değeri (m)		Yer işaretinin konumunun hata değeri (m)	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
ÇTYÖ	2,9364	3,3027	4,9047	4,9443
ÇKYÖ	3,0079	2,3936	5,0195	4,9169
SYÖ	2,1777	2,1460	4,6200	4,5902
AYÖ	2,9565	2,9064	4,3152	4,7642
MYÖ	4,1244	5,7070	4,6869	4,8262

Tüm yeniden örnekleme metotları için değişen q_v gürültü başlangıç değerlerinde robot konumu tahmin edilirken özellikle de veri ilişkilendirmesinin robota sağlanmadığı durumda hata oranlarının düzensiz bir şekilde değiştiği Şekil 4.44'e bakılarak söylenebilir. 5 farklı yeniden örnekleme yönteminden SYÖ'nün diğer yeniden örnekleme metotlarına göre daha iyi sonuçlar verdiği, AYÖ'nün ise veri ilişkilendirmesinin robota sağlandığı durumda yer işareti konumlarında daha başarılı tahminlerde bulunduğu görülmüştür. Robot konumunda en kötü tahminleri MYÖ, yer işaretleri konumunda ise ÇKYÖ metodu yapmıştır. ÇKYÖ'nün geliştirilmiş versiyonu olan SYÖ değişen gürültü değerlerinde ÇKYÖ'ye göre daha doğru tahminlerde bulunmuştur. En yüksek ve en düşük RMSE değerlerine ulaşıldığı $q_v = 0,16 \text{ m/s}$ başlangıç değeri için oluşturulan haritalar Şekil 4.46 ve 4.47'de verilmiştir. SYÖ'nün yaptığı tahminlerin MYÖ'ye göre çok daha iyi olduğu bu iki haritada da rahatlıkla görülebilir.



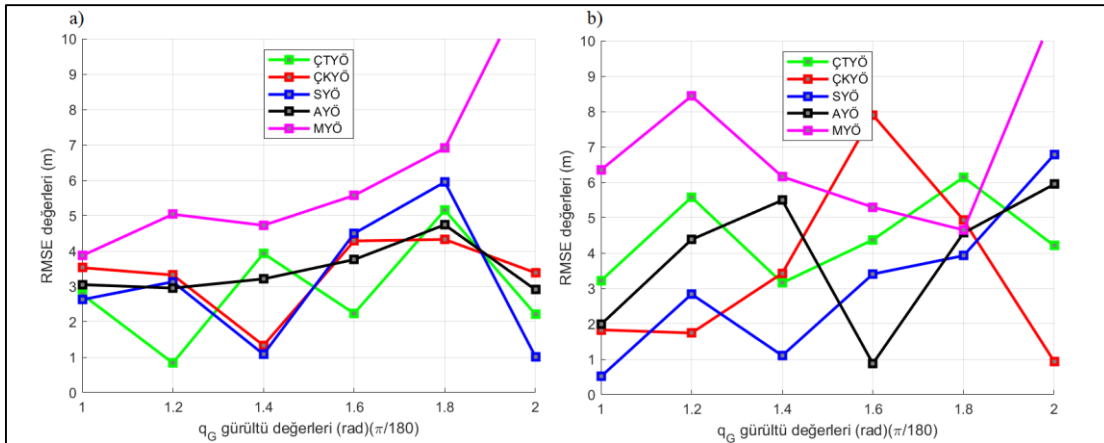
Şekil 4.46. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ



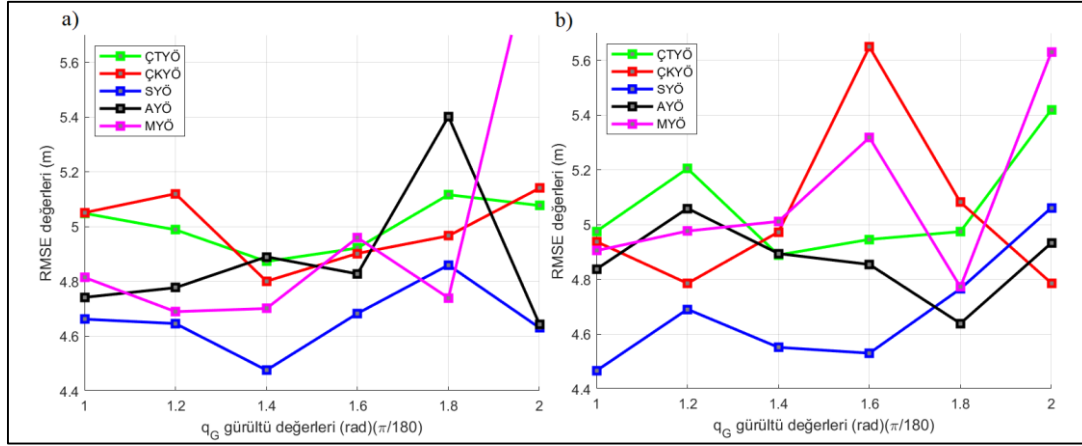
Şekil 4.47. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ

4.3.2. İşlem gürültü bileşeni q_G 'nin değişimi

Robotun hız bileşenine etki eden gürültü bileşeninden sonra dönüş açısına etki eden gürültü bileşeni q_G 'nin değişiminin yeniden örnekleme metotları üzerindeki etkisi veri ilişkilendirilmesinin verildiği ve verilmediği duruma göre bu başlık altında incelenmiştir. Bu değişim Çizelge 4.1'in Gürültü Testi – 2'deki değerlerine göre yapılmıştır. Değişimlere göre hesaplanan RMSE değerleri Şekil 4.48 ve 4.49'da gösterilmiştir.



Şekil 4.48. q_G değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde



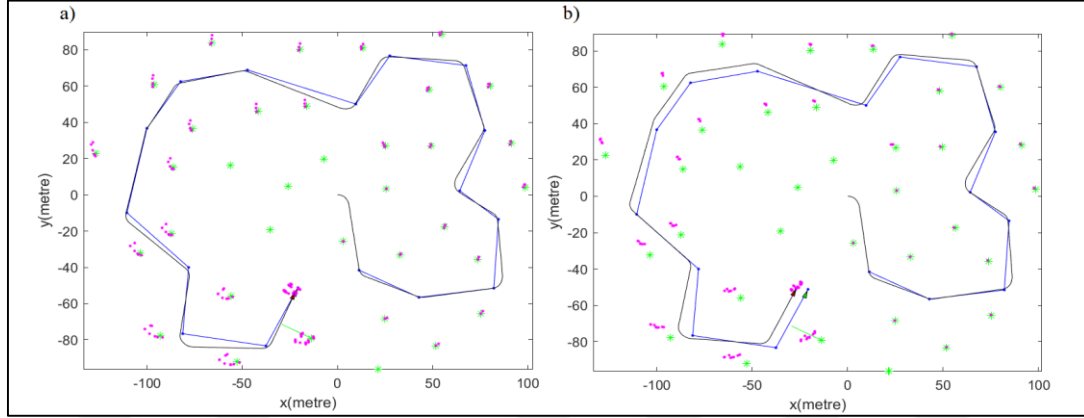
Şekil 4.49. q_v değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

Çizelge 4.11. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen q_G gürültü başlangıç değerlerindeki ortalama hata değerleri

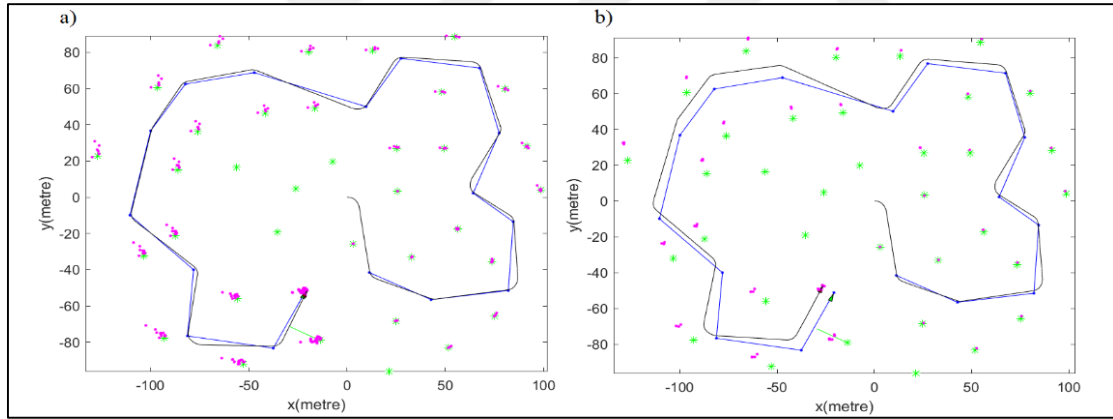
	Robot konumunun hata değeri (m)		Yer işaretinin konumunun hata değeri (m)	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
ÇTYÖ	2,8576	4,4462	5,0048	5,0688
ÇKYÖ	3,3638	3,4619	4,9973	5,0360
SYÖ	3,0500	3,0937	4,6604	4,6787
AYÖ	3,4370	3,8741	4,8810	4,8703
MYÖ	6,3559	6,9500	5,0055	5,1029

q_G 'nin başlangıç değerinin değişimi ile robot konumu için hesaplanan hata değerleri, robota veri ilişkilendirmesinin verilmesi ve verilmemesi durumunda yeniden örnekleme metotuna göre büyük farklılıklar göstermektedir. Şekil 4.48'e bakıldığında ÇTYÖ'nün hata oranında büyük bir yükseliş görülürken SYÖ'nün hata oranında bir değişim görülmemiştir. q_v 'nin değişiminde olduğu gibi konum tahmininde SYÖ baskın olan metottur. Veri ilişkilendirmesi verildiğinde ÇTYÖ SYÖ'den daha kaliteli bir sonuç verse bile veri ilişkilendirmesi verilmediğinde SYÖ'de değişim çok küçük olmuş ÇTYÖ'de ise değişim çok fazla olmuştur. En kötü tahminler ise MYÖ tarafından yapılmıştır. Yer işaretlerinin tahmininde veri ilişkilendirmesinin çok fazla etki göstermediği Şekil 4.49'da görülebilir. Her iki durum için de değişen gürültü başlangıç değerlerinde SYÖ'nün çok daha doğru sonuçlar verdiği görülmektedir. Robot konumunda veri ilişkilendirmesi verilmesi ve verilmemesi durumuna göre hatanın en az olduğu $q_G = 1,4 \pi/180 \text{ rad}$ durumunda SYÖ ve MYÖ'nün kullanıldığı

FastSLAM uygulamasında oluşturulan tahmini robot yolu ve harita Şekil 4.50 ve 4.51’de gösterilmiştir.



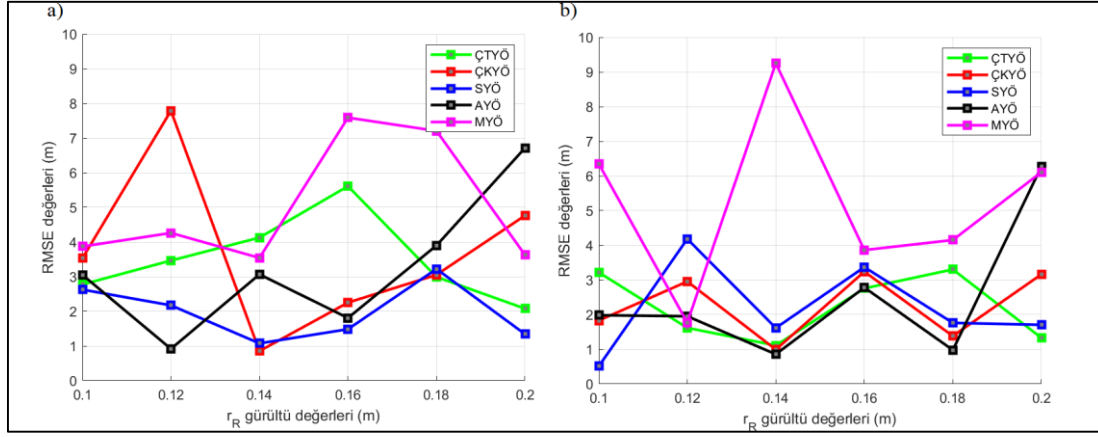
Şekil 4.50. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ



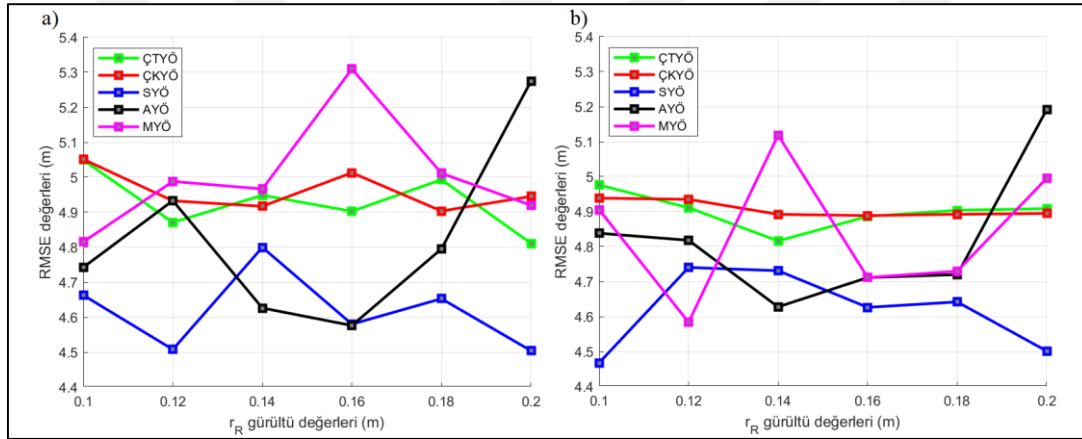
Şekil 4.51. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) MYÖ

4.3.3. Ölçüm gürültü bileşeni r_R ’nin değişimi

Bu bölümde robotun yaptığı ölçümlere etki eden gürültü bileşenlerinden r_R ’nin yeniden örnekleme metotlarına dolayısıyla FastSLAM algoritması üzerine olan etkisi incelenmiştir. r_R için Çizelge 4.1’de verilen gürültü testlerinden 3.’sü uygulanmış ve hareket boyunca yapılan tüm tahminler için RMSE değerleri hesaplanmıştır. Her gürültü değeri için hesaplanan RMSE değerleri Şekil 4.52 ve 4.53’te verilmiştir.



Şekil 4.52. r_R değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde



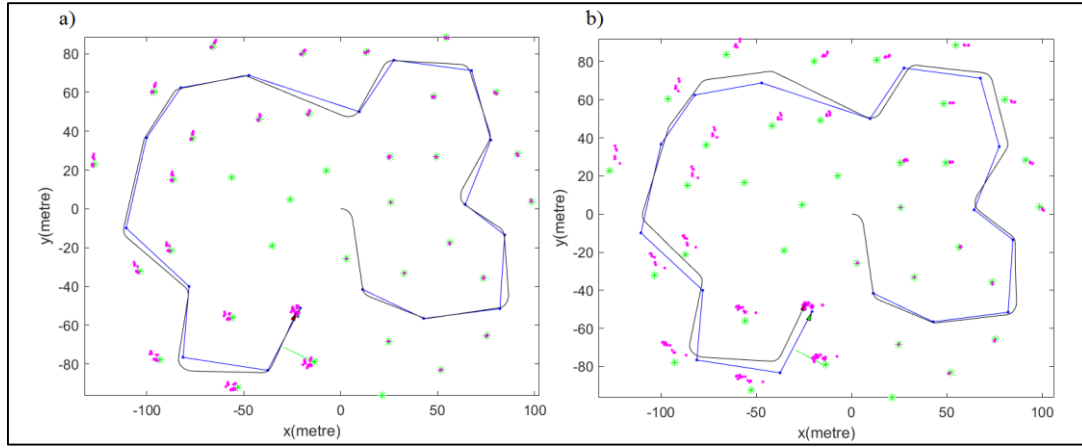
Şekil 4.53. r_R değişimine göre yer işaretlerinin konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

Çizelge 4.12. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen r_R gürültü başlangıç değerlerindeki ortalama hata değerleri

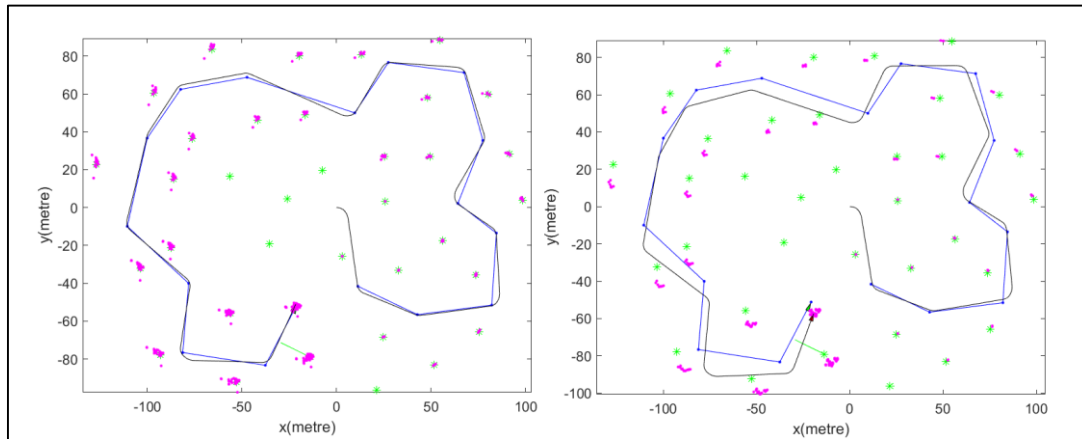
	Robot konumunun hata değeri (m)		Yer işaretinin konumunun hata değeri (m)	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
ÇTYÖ	3,5074	2,2219	4,9293	4,9003
ÇKYÖ	3,7063	2,2593	4,9605	4,9069
SYÖ	1,9873	2,1901	4,6179	4,6180
AYÖ	3,2429	2,4698	4,8247	4,8179
MYÖ	5,0153	5,2480	5,0018	4,8407

Hem robot hem de yer işareti konumu için verilen grafiklerde SYÖ metodunun veri ilişkilendirmesinin verildiği/verilmediği durumlarda diğerlerine göre doğru konumlara daha

yakın tahminlerde bulunduğu söylenebilir. Değişen gürültü şartlarında robot konumu için hata oranlarının veri ilişkilendirmesinin verilmediği durumda, MYÖ hariç, verildiği durumdan daha az olduğu gözlemlenmiştir. Aynı zamanda r_R 'nin değişiminin hata oranlarında büyük değişimler yaratmadığı Şekil 4.52.(b)'de görülebilir. Yer işaretleri tahmininde ÇKYÖ'nin daha düzenli sonuçlar verdiğine ve SYÖ ile bu sonuçların iyileştirildiği Şekil 4.53 incelenerek söylenebilir. Veri ilişkilendirmesi robota verildiği durumda robot konum hatasının en az ve en çok olduğu yeniden örnekleme metotları SYÖ ve AYÖ ile oluşturulmuş haritalar $r_R = 0,2m$ başlangıç seviyesindeyken Şekil 4.55 ve 4.56'daki gibi olmuştur.



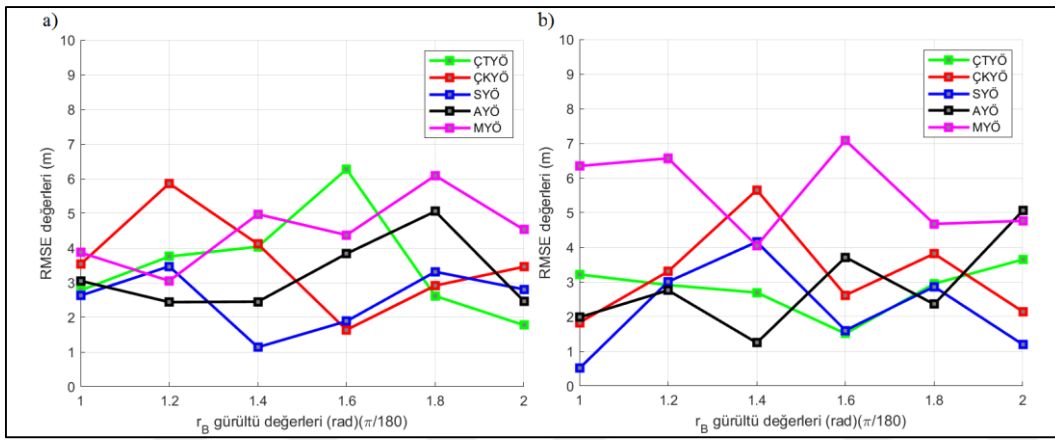
Şekil 4.54. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) AYÖ



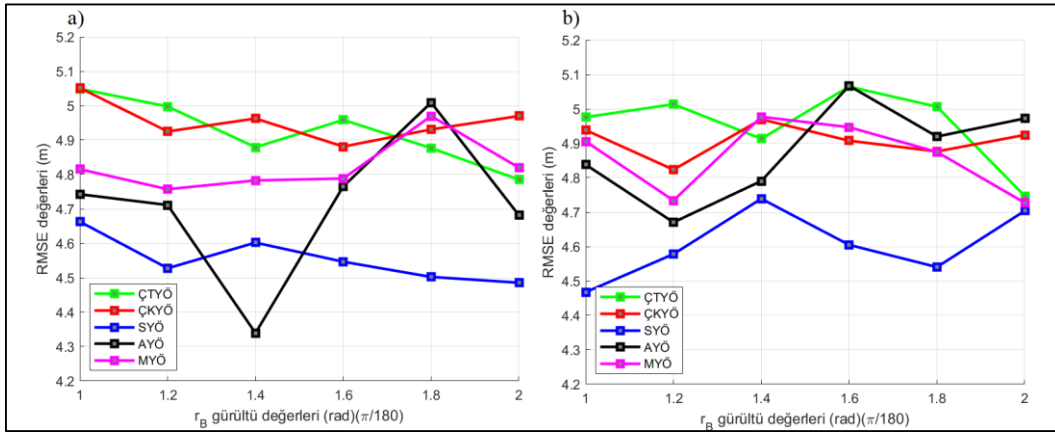
Şekil 4.55. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) AYÖ

4.3.4. Ölçüm gürültü bileşeni r_B 'nin değişimi

Robotun hareketi boyunca yaptığı gözlemleri etkileyen bir diğer gürültü bileşeni ise r_G 'dir r_G , Çizelge 4.1 Gürültü testi – 4'te verilen gürültü başlangıç değerlerini aldığı FastSLAM algoritmasında kullanılan yeniden örnekleme metotlarının farklı r_G seviyelerinde yaptığı tahminlerin doğruluğu incelenmiş ve tahminlerde yaptığı hataların RMSE değerleri hesaplanmıştır. Hesaplanan bu değerler grafik halinde Şekil 4.55 ve 4.56'da gösterilmiştir.



Şekil 4.56. r_G değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde



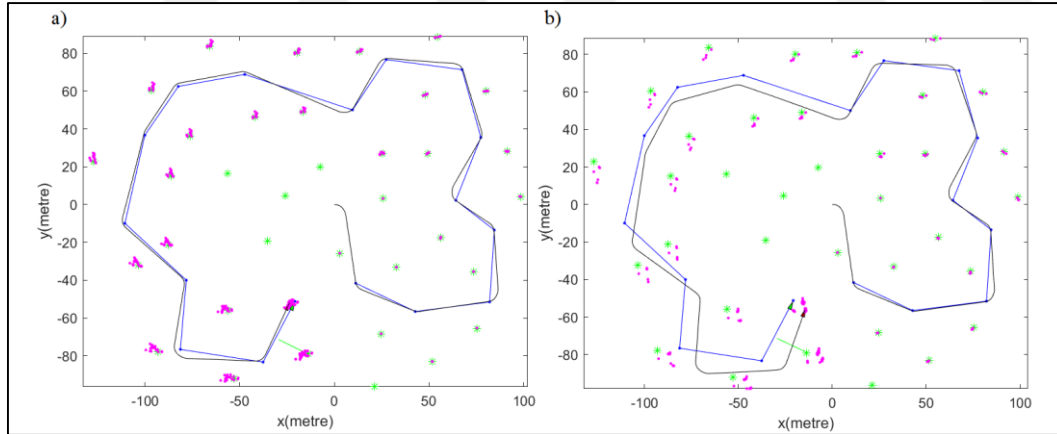
Şekil 4.57. r_G değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

r_R 'de olduğu gibi değişen gürültü şartlarında veri ilişkilendirmesinin verilmediği durumda yeniden örnekleme metotlarının robot konum tahmininde daha iyi sonuçlar verdiği Şekil 4.55'te görülmektedir. Aynı şekilde SYÖ metodunun tahmin kalitesi hem yer işaretleri

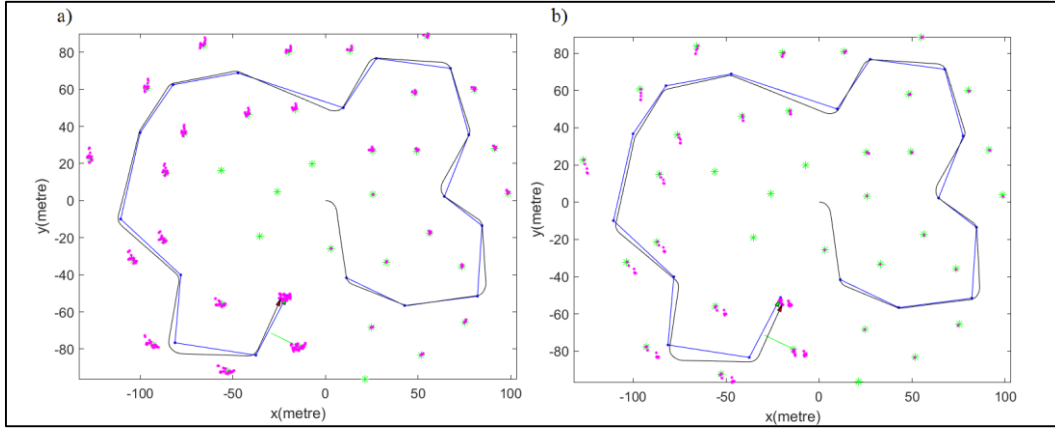
tahmininde hem de robot konum tahmininde diğerlerine göre çok daha iyi olmuştur. Robot konum tahmininde MYÖ, yer işaretlerinin tahmininde ise ÇTYÖ en hatalı tahminlerde bulunmuştur. Yer işaretlerinin tahmininde veri ilişkilendirmesinin FastSLAM uygulaması altında çok büyük bir etkisi olmadığı Şekil 4.55'te görülmektedir. Hareket sonucu yeniden örneklemede SYÖ ve ÇTYÖ'nün kullanıldığı ve $r_B = 1,6 \pi/180 \text{ rad}$ başlangıç gürültü seviyesinin seçildiği durumda oluşturulan haritalar Şekil 4.57 ve 4.58'de verilmiştir.

Çizelge 4.13. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen r_B gürültü başlangıç değerlerindeki ortalama hata değerleri

	Robot konumunun hata değeri (m)		Yer işaretinin konumunun hata değeri (m)	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
ÇTYÖ	3,5388	2,8225	4,9244	4,9535
ÇKYÖ	3,5860	3,2286	4,9535	4,9067
SYÖ	2,5402	2,2231	4,5545	4,6056
AYÖ	3,2170	2,8529	4,7082	4,8764
MYÖ	4,4813	5,5822	4,8222	4,8609



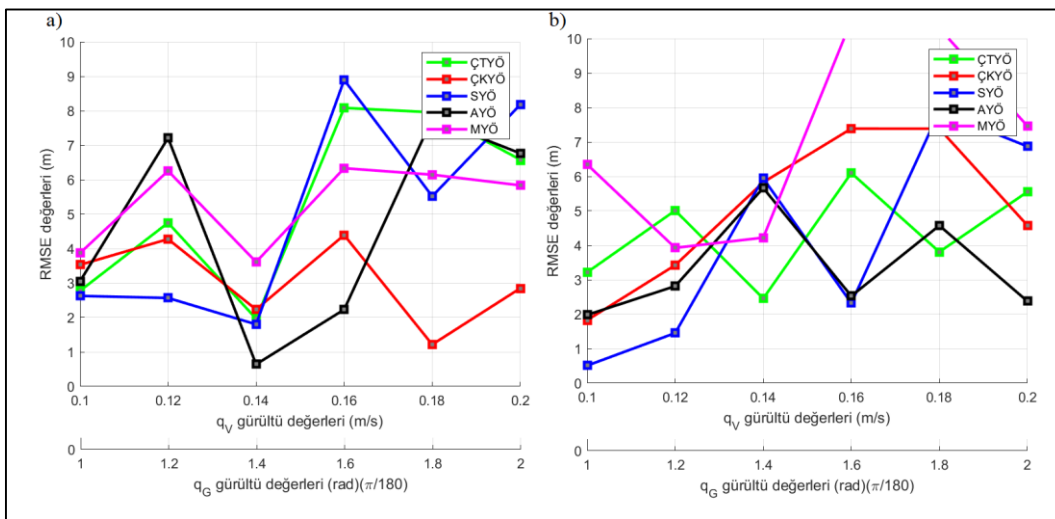
Şekil 4.58. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) ÇTYÖ



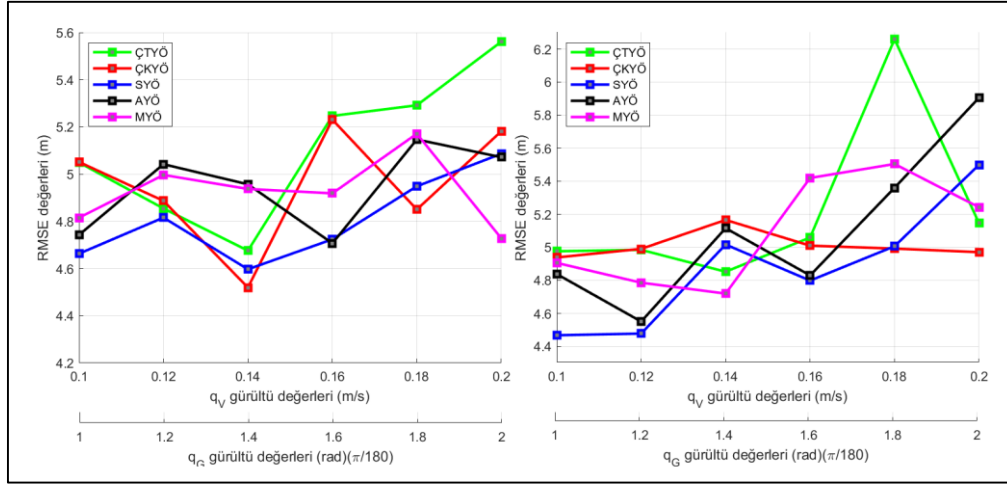
Şekil 4.59. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) SYÖ b) ÇTYÖ

4.3.5. İşlem gürültü bileşenleri q_v ve q_G 'nin değişimi

İşlem gürültü bileşenlerinin robot tahmini üzerindeki etkileri tek tek incelendikten sonra q_v ve q_G 'nin Çizelge 4.1'de tanımlanmış Gürültü Testlerinden 5.'sinde verilen aralıklarda değiştiği durumda robot tahmini üzerindeki etkisi incelenmiştir. Yeniden örnekleme metotlarının hareket boyunca yaptığı tahminlerin kalitesi doğru değerler ve tahmin değerleri arasındaki fark ile oluşan RMSE değerleri hesaplanarak test edilmiş ve bulunan değerler Şekil 4.59 ve 4.60'ta gösterilmiştir.



Şekil 4.60. q_v ve q_G değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

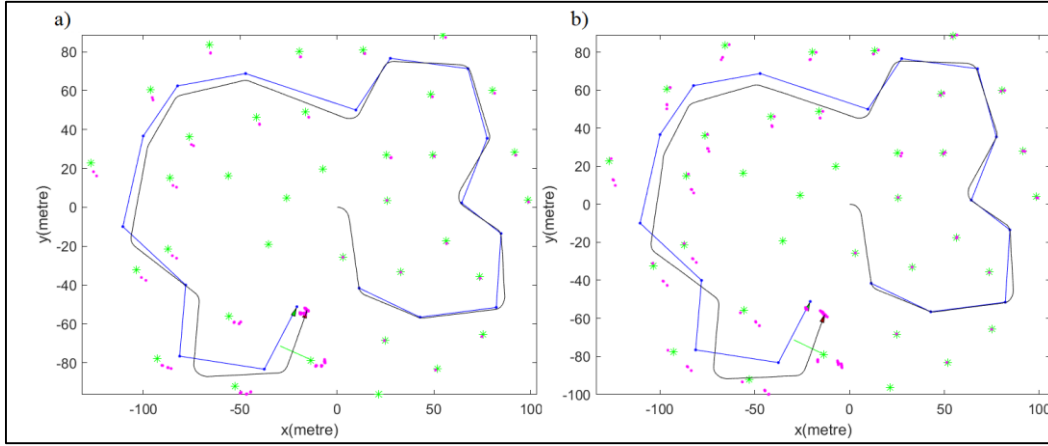


Şekil 4.61. q_v ve q_G değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

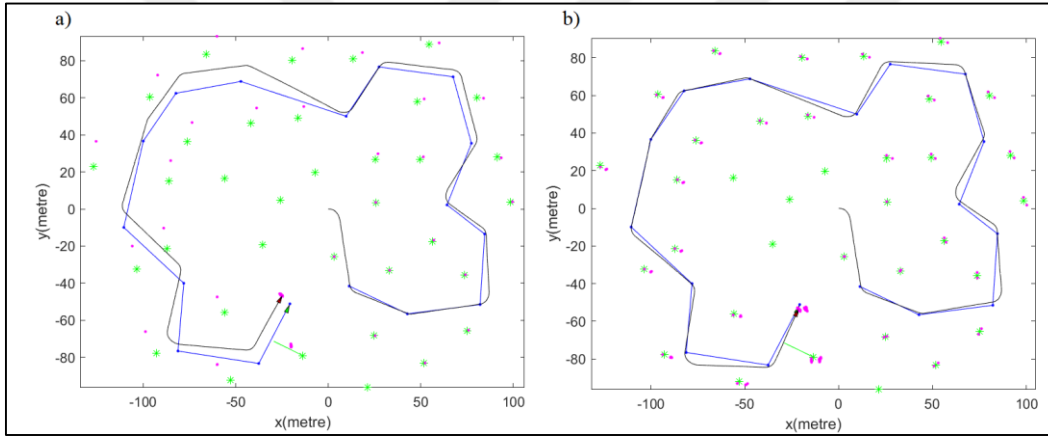
Çizelge 4.14. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen q_v ve q_G gürültü başlangıç değerlerindeki ortalama hata değerleri

	Robot konumunun hata değeri (m)		Yer işaretinin konumunun hata değeri (m)	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
ÇTYÖ	5,3514	4,3581	5,1132	5,2130
ÇKYÖ	3,0823	5,0724	4,9534	5,0112
SYÖ	4,9312	4,1660	4,8057	4,8777
AYÖ	4,6117	3,3240	4,9445	5,1000
MYÖ	5,3415	7,1296	4,9279	5,0962

İşlem gürültüsünün beraber değiştiği durumda önceki testlere kıyasla robot konumunun tahmininde SYÖ ÇKTÖ ve AYÖ'nün gerisinde kalmıştır. Şekil 4.59.(a)'da görüleceği gibi veri ilişkilendirmesi verildiğinde ÇKTÖ, Şekil 4.59.(b)'de görüleceği gibi veri ilişkilendirmesi verilmediğinde AYÖ en az hatalı tahminlerde bulunmuştur. Aynı şekilde her iki yöntem gürültü değişiminden en az etkilenen yeniden örnekleme yöntemleri olmuştur. Yer işaretleri tahmininde ise SYÖ diğer yöntemlere kıyasla daha başarılı tahminler yapmıştır. Diğer testlerden farklı olarak öne çıkan ÇKTÖ ve AYÖ metotlarının kullanıldığı ve gürültü başlangıç değerlerinin $q_v = 0,14m$ ve $q_G = 1,4 \pi/180 rad$ olduğu durumda robotun harita üzerinde aldığı yol ve yer işaretleri için yaptığı tahminler Şekil 4.61 ve 4.62'de verilen haritalarda gösterilmiştir.



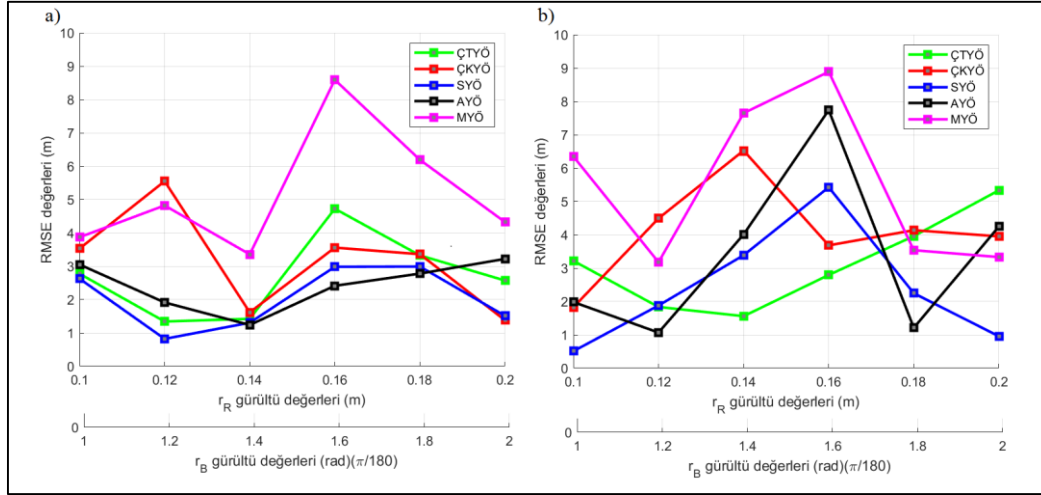
Şekil 4.62. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) AYÖ b) ÇKYÖ



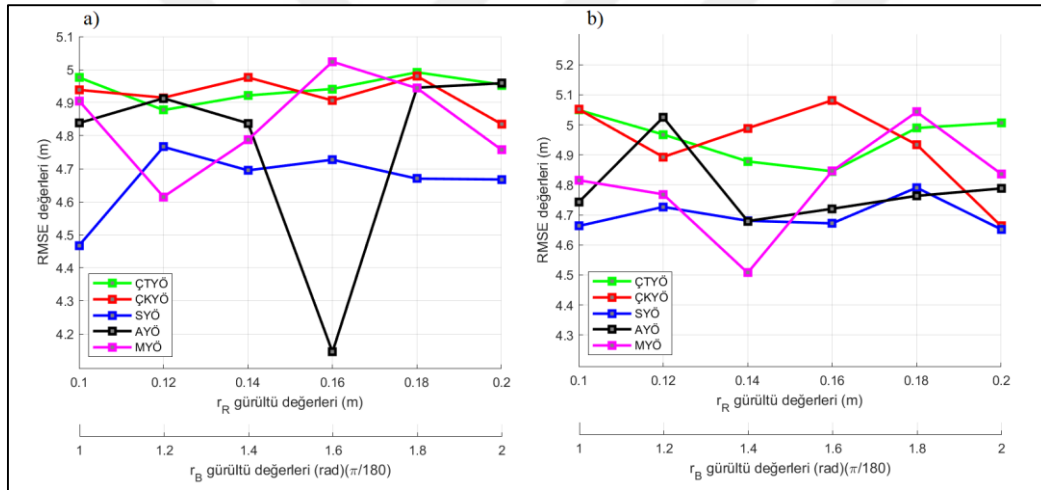
Şekil 4.63. Veri ilişkilendirmesinin bilinmediği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) AYÖ b) ÇKYÖ

4.3.6. Ölçüm gürültü bileşenleri r_R ve r_B 'nin değişimi

Yeniden örnekleme metotlarının değişen gürültü şartlarına olan tepkilerinin son olarak inceleneceği test ölçüm gürültü bileşenlerinin beraber değiştiği ve başlangıç değeri olarak Çizelge 4.1'de Gürültü Testi – 5'te verilen değerleri aldığı testtir. Ölçüm gürültüsünün değiştiği durumda hem robotun hem de yer işaretlerinin doğru konumlarına karşılık tahmin edilen konumları arasındaki fark kullanılarak hesaplanan RMSE değerleri Şekil 4.63 ve 4.64'te gösterilmiştir.



Şekil 4.64. r_R ve r_B değişimine göre robot konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verildiğinde b) veri ilişkilendirmesi verilmediğinde

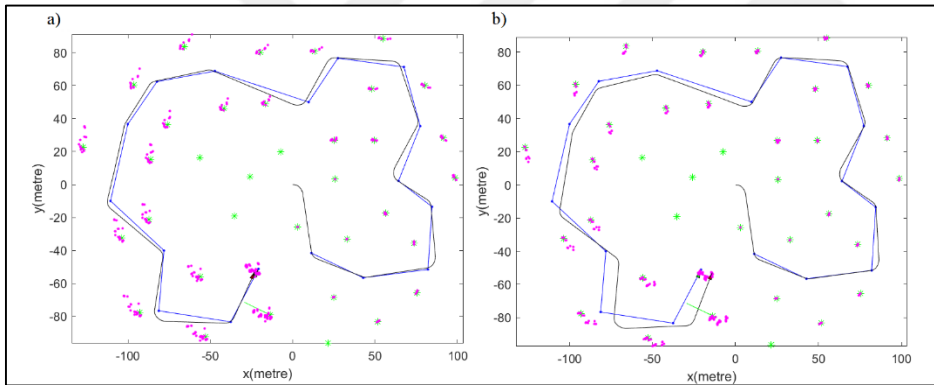


Şekil 4.65. r_R ve r_B değişimine göre yer işareti konumu için hesaplanan RMSE değerleri. a) veri ilişkilendirmesi verilmediğinde b) veri ilişkilendirmesi verildiğinde

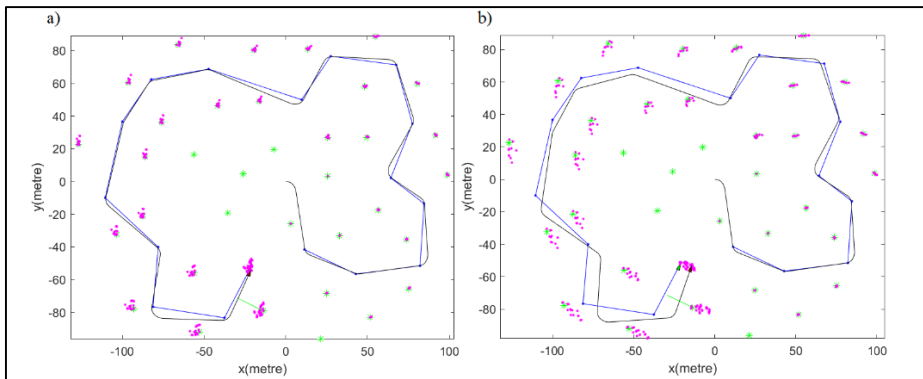
Çizelge 4.15. Robot ve yer işareti konum hatalarının farklı yeniden örnekleme metotlarında değişen işlem gürültü başlangıç değerlerindeki ortalama hata değerleri

	Robot konumunun hata değeri (m)		Yer işaretinin konumunun hata değeri (m)	
	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde	Veri ilişkilendirmesi verildiğinde	Veri ilişkilendirmesi verilmediğinde
ÇTYÖ	2,6938	3,1170	4,9561	4,9438
ÇKYÖ	3,1658	4,1033	4,9850	4,9252
SYÖ	2,0402	2,4007	4,6974	4,6655
AYÖ	2,4366	3,3764	4,7864	4,7730
MYÖ	5,1926	5,4881	4,8027	4,8388

r_R ve r_B 'nin ayrı ayrı incelendiği durumlarda olduğu gibi iki ölçüm gürültü bileşeninin aynı anda değişti durumda da SYÖ hem robot konumu tahmininde hem de yer işareti tahmininde diğer yöntemlere göre daha az hatalı tahminlerde bulunmuştur. Bu tahmin kalitesi robot konumu için veri ilişkilendirmesi robota verilmediği durumda verildiği duruma göre iyileşme gösterirken yer işareti konumunda hatanın bir miktar arttığı gözlemlenmiştir. Şekil 4.64'te görüleceği gibi yer işareti tahmininde SYÖ değişen gürültü değerlerinden etkilenmemiş ve tüm gürültü değerlerinde yaklaşık aynı hata oranında tahminlerde bulunmuştur. Robot konumu tahmininde veri ilişkilendirmesinin verilmediği durumda MYÖ tarafından yapılan tahminlerin diğer yeniden örnekleme metotlarının yaptığı tahminlere göre çok daha fazla olması dikkat çekmektedir. Hata oranının veri ilişkilendirmesinin verildiği durumda en az olduğu gürültü başlangıç değeri $r_R = 0,14m$ ve $r_B = 1,4\pi/180 \text{ rad}$ için SYÖ ve ÇKYÖ'nün yeniden örnekleme metodu olarak kullanıldığı FastSLAM uygulamasında robotun çizdiği harita ve konum tahmini Şekil 4.65 ve 4.66'da verilmiştir.



Şekil 4.66. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. a) AYÖ b) ÇKYÖ



Şekil 4.67. Veri ilişkilendirmesinin bilindiği durumda robotun aldığı yolun tahmini (siyah) ve yer işaretlerinin konumunun tahmini (pembe) ile doğru yol (mavi) ve yer işaretleri (yeşil) konumlarının harita üzerinde gösterimi. A) AYÖ b) ÇKYÖ

5. SONUÇLAR VE ÖNERİLER

Bu çalışmada robotun bulunduğu ortamın haritasına ulaşamadığı ve bulunduğu konumu bilmediği durumda hem haritasını hem de konumunu yaptığı gözlemler ve tahminler doğrultusunda kendisinin belirlediği EZYTH problemi incelenmiştir. Literatürde EZYTH problemin çözümü için olasılık tabanlı filtreler olan KF ve PF'ler yaygın olarak kullanılmaktadır. KF ailesinin önemli bir uygulaması olan GKF ve GKF'nin gürültü istatistiklerine adapte olabilen hali olan AGKF çözümleri EZYTH'ye uygulanmış ve farklı gürültü başlangıç seviyelerinde sonuçlar karşılaştırılmıştır. AGKF, DKT ile birlikte düzenlenmiş ve DKT'nin AGKF'de sağladığı iyileştirmeler incelenmiştir. PF'nin SLAM üzerindeki uygulaması olan FastSLAM içerisinde literatürde en çok kullanılan 5 farklı yeniden örnekleme metodunun karşılaştırmaları farklı gürültü başlangıç seviyelerinde gerçekleştirilmiştir. Bu karşılaştırmalar yapılırken EZYTH'nin önemli problemlerinden birisi olan veri ilişkilendirmesinin robota hareket öncesi ve sonrası sağlandığı durum göz önünde bulundurulmuştur.

EZYTH probleminin çözümünde en yaygın olarak kullanılan metotlardan birisi olan GKF'de gürültü istatistiklerinin sabit olduğu varsayıldığından benzetim ortamı ile gerçek hayat uygulaması arasında tutarsızlıklar ortaya çıkabilir. Bunun için gürültü istatistiklerine adapte olabilen AGKF yaklaşımı GKF ile karşılaştırılmıştır. Bu karşılaştırmalar yapılırken gürültü istatistiklerinin bileşenlerinin başlangıç değerleri farklı seçilmiş ve her bileşenin yapılan tahminlere olan etkisi ayrı ayrı gözlemlenmiştir. Gürültü bileşenlerinden q_v değişiminde veri ilişkilendirmesi robota verildiği ve verilmediği durumda robot konum hatası için GKF'de yaklaşık $0.7m$ 'lik bir artış gözlemlenirken AGKF için bu artış $0.2m$ olmuştur. AGKF'nin GKF'ye göre yaptığı tahminlerde ise $1 - 2m$ arasında daha az hata yaptığı görülmüştür. Yer işaretlerinin tahmininde ise veri ilişkilendirmesinin verilmesi ve verilmemesi durumundan yaklaşık olarak GKF $0.02m$, AGKF ise $0.07m$ lik hata farkı ile etkilenmiş fakat tahmin değerlerinde AGKF, GKF'ye göre $0.4m$ daha az hata yapmıştır. q_G 'nin başlangıç değerlerinin değiştiği durumda q_v 'ye göre yapılan tahmin hatasının arttığı görülmüş ve robotun dönüş açısındaki gürültünün hem robot konumu tahmininde hem de yer işareti tahmininde büyük bir etkisi olduğu belirlenmiştir. Robot konumu hatasında GKF'nin hata oranlarının q_v 'ye göre $2 - 3m$ arasında arttığı, AGKF'nin ise $1 - 2m$ arasında arttığı, AGKF'deki artış oranının GKF'den daha az olduğu görülmüştür. Veri

ilişkilendirmesi verilmemesi durumunda GKF'nin tahminindeki hatası AKGF'den daha fazla artmıştır. Yer işaretlerinin tahmininde ise q_v 'ye benzer bir durum gözlemlenmiştir. İşlem gürültüsünün etkileri ayrı ayrı incelendikten sonra ölçüm gürültüsü bileşenlerinin değişiminin filtreler üzerindeki etkisi gözlemlenmiştir. r_R 'nin başlangıç değerindeki değişimde, AGKF'nin yaptığı tahminlerde GKF'nin yaptığı tahminlere kıyasla robot konumu için yaklaşık $1.5m$ yer işaretleri için ise yaklaşık $0.3m$ kadar daha az hata yaptığı görülmüştür. Ortalama hata değerlerinde r_G 'nin değişimi de r_R 'ye benzer şekilde olmuştur. Tahmin hatalarındaki en büyük farklardan birisi işlem gürültü değerlerinin aynı anda değiştiği durumda gözlemlenmiştir. GKF'nin değişen gürültü değerlerinde tahminlerinin git gide kötüleştiği, AGKF'nin ise bazı aralıklarda tahmin kalitesini iyileştirdiği gözlemlenmiştir. İki yöntem arasındaki fark bu testte robot konumu için $2m$, yer işareti konumu için $0.4m$ olarak hesaplanmıştır. Ölçüm gürültüsünde ise ayrı ayrı etkilediği durumdan farklı bir etki görülmemiş, veri ilişkilendirmesi verildiği durumda hata oranında bir artışın olmadığı gözlemlenmiştir. Sonuç olarak AGKF robot konumu için yaptığı tahminlerde değişen gürültü başlangıç değerlerine GKF'den daha rahat uyum sağlamış hem veri ilişkilendirmesinin verildiği hem de verilmediği durumda GKF'nin tahminlerinden daha tutarlı ve doğru tahminler yapmıştır.

AGKF her ne kadar GKF'den daha doğru sonuçlar verse de filtrenin oluşturulması sırasında R ve Q kovaryans matrislerinin pozitif tanımlı olarak kabul edilmesi robot hareket ettikçe filtrenin ırsaksamasına sebep olabilir. Tahminler ve ölçümlerin sayısı arttıkça R ve Q pozitif tanımlılığını kaybedebilir. Bunun için son ölçüm değerlerin etkisinin artırıldığı DKT yöntemi AGKF'ye uygulanmıştır. Tüm gürültü değişimlerinde veri ilişkilendirmesinin robota sağlandığı durumda hem AGKF'nin hem de AGKF-DKT'nin tahminleri benzer, hata oranları aynı çıkmıştır. Veri ilişkilendirmesi sağlanmadığında ise önce işlem gürültü bileşenleri sonra da ölçüm gürültü bileşenlerinin başlangıç gürültü değerleri değiştirilerek karşılaştırmalar yapılmıştır. DKT AGKF'nin R ve Q kovaryans matrislerinin pozitif tanımlılığının korunmasını sağladığı gibi aynı şekilde tahmin doğruluğunda da iyileştirme sağlamıştır.

Son karşılaştırma FastSLAM için kullanılan yeniden örnekleme metotları üzerinden yapılmıştır. Önem örneklemesi olarak robotun hareket modeli seçildiği için FastSLAM parçacıklarını gürültü değerlerine göre üretir. Üretilen bu parçacıklar sürekli olarak yenilenir. Bunun için yeniden örnekleme metodunun gürültü değerlerinin başlangıç

seviyesindeki değişime verdiği tepki önemlidir. Yapılan testler sonucunda uygulanan beş yöntemden SYÖ'nün değişen gürültü değerlerine göre yaptığı tahminlerin diğer yöntemlere göre daha tutarlı ve doğru olduğu görülmüştür. MYÖ metodu ise robot konumunda en çok hatanın ortaya çıktığı metot olmuştur. Yeniden örnekleme metodu olarak MYÖ'nün kullanıldığı durumlarda SYÖ'ye göre robot konumu tahmininde gerçek konumuna göre 2m ve daha fazla hata yapıldığı hesaplanmıştır. Veri ilişkilendirmesinin GKF ve AGKF'nin aksine FastSLAM'i çok fazla etkilemediği sadece q_G 'nin başlangıç değerlerinin değişiminde veri ilişkilendirmesi verilmediğinde hata oranında bir artış olduğu gözlemlenmiştir.

Bu çalışma sonucunda gürültü istatistiklerine adaptasyonu ile AGKF'nin GKF'den daha iyi sonuçlar verdiği, DKT'nin AGKF'nin tahmin değerlerini iyileştirdiği gözlemlenmiştir. FastSLAM için önerilen yeniden örnekleme metotlarının arasında SYÖ'nün tahmin kalitesinin çoğu aralıkta daha iyi olduğu fakat bazı başlangıç değerlerinde farklı yöntemlerin daha iyi sonuçlar da verebildiği belirlenmiştir. EZYTH probleminin çözümünde kullanılan bu metotların birbiri arasında karşılaştırılması yapıldığında AGKF'nin hem FastSLAM'den hem de GKF'den çok daha doğru tahminler yaptığı görülmüştür. FastSLAM ise SYÖ yeniden örnekleme metodu kullanıldığında işlem gürültüsünün başlangıç değerinin beraber değiştiği durum haricinde robot ve yer işareti konumunun tahmininde GKF'den daha iyi sonuçlar bulmuştur.

Gelecek çalışmalarda hem ortam modeline hem de sensör modeline göre yeniden örnekleme metotlarının dinamik bir şekilde kullanılabileceği bazı çalışmalar yapılabilir. AGKF'nin değişen gürültü istatistiklerine olan adaptasyonun tahminlerde iyileşme sağladığı görüldüğü için FastSLAM için de uygun bir gürültü adaptasyon yöntemi uygun bir yeniden örnekleme modeli ile kullanılarak tahmin kalitesi artırılabilir düşünlümlenmektedir. Yer işaretlerinin konumunun değiştiği ve veri ilişkilendirmesinin zorlaştığı dinamik ortamlarda bu yöntemler denenmelidir. FastSLAM için önem örnekleme yöntemi iyileştirilebilir, parçacık çeşitliliğini artıran ve dejenere olmasını etkili şekilde önleyebilecek genetik algoritmaları gibi farklı yeniden örnekleme metotları kullanılabilir.



KAYNAKLAR

1. Dissanayake, M. W. M., Newman, P., Clark, S., Durrant-Whyte, H. F. ve Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem, *IEEE Trans. Robot. Autom.*, 17(3), 229-241.
2. Durrant-Whyte, H. ve Bailey, T. (2006). Simultaneous localization and mapping: part I, *IEEE Robot. Autom. Mag.*, 13(2), 99–110.
3. Taheri, H. ve Xia, Z. C. (2021). SLAM; definition and evolution, *Eng. Appl. Artif. Intell.*, 97, 104032.
4. Thrun, S.(2005). *Probabilistic robotics*, (1. Baskı), Londra: MIT Press, 646 .
5. Bailey, T. (2002). *Mobile Robot Localisation and Mapping in Extensive Outdoor Environments*, Doktora Tezi, Sidney Üniversitesi Mekanik ve Mekatronik Mühendisliği, Sidney.
6. İnternet: Thrun, S., Montemerlo, M., Koller, D., Wegbreit, B., Nieto, J. ve Nebot, E. (2004). FastSLAM: An Efficient Solution to the Simultaneous Localization And Mapping Problem with Unknown Data Association. Web: <http://robots.stanford.edu/papers/Thrun03g.pdf> Erişim Tarihi: 25.11.2021.
7. Cadena, C. (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age, *IEEE Trans. Robot.*, 32(6), 1309–1332.
8. Kalman, R. E. ve Bucy, R. S. (1964). New Results in Linear Filtering and Prediction Theory, *J. Basic Eng.*, 83(1), 95–108.
9. Julier, S. J. ve Uhlmann, J. K. (2001). *A counter example to the theory of simultaneous localization and map building*, Proc. - IEEE Int. Conf. Robot. Autom., Seoul, 4238-4243.
10. Tian, Y., Suwoyo, H., Wang, W., Mbemba, D. ve Li, L. (2020). An AEKF-SLAM Algorithm with Recursive Noise Statistic Based on MLE and EM, *J. Intell. Robot. Syst. Theory Appl.*, 97(2), 339–355.
11. Huang, Y., Zhang, Y., Xu, B., Wu, Z. ve Chambers J. A. (2018). A New Adaptive Extended Kalman Filter for Cooperative Localization, *IEEE Trans. Aerosp. Electron. Syst.*, 54(1), 353–368.
12. Gao, W., Li, J., Zhou, G., ve Li, Q. (2015). Adaptive Kalman filtering with recursive noise estimator for integrated SINS/DVL systems, *J. Navig.*, 68(1), 142–161.
13. Murphy, K. P. (1999). *Bayesian Map Learning in Dynamic Environments*, NIPS'99: Proceedings of the 12th International Conference on Neural Information Processing Systems, Denver, 1015–1021.
14. Li, T., Bolic, M. ve Djuric, P. M. (2015). Resampling Methods for Particle Filtering: Classification, implementation, and strategies, *IEEE Signal Process. Mag.*, 32(3), 70–86.

15. Smith, C. ve Cheeseman, P. (1986). On the Representation and Estimation of Spatial Uncertainty, *Int. J. Rob. Res.*, 5(4), 56–68.
16. Durrant-Whyte, H. F. (1988). Uncertain geometry in robotics, *IEEE J. Robot. Autom.*, 4(1), 23–31.
17. Leonard, J. J. ve Durrant-Whyte, H. F. (1991). *Simultaneous map building and localization for an autonomous mobile robot*, Proceedings IROS '91:IEEE/RSJ International Workshop on Intelligent Robots and Systems '91, Osaka, 1442–1447.
18. Barshan, B. ve Durrant-Whyte, H. F. (1995). Inertial navigation systems for mobile robots, *IEEE Trans. Robot. Autom.*, 11(3), 328–342.
19. Andrade-cetto, J. ve Sanfelu, A. (2002). Concurrent Map Building And Localization On Indoor Dynamic Environments, *Int. J. Pattern Recognit. Artif. Intell.*, 16(3), 361–374.
20. Barvinok, A. ve Veomett, E. (2008). The computational complexity of convex bodies, 117–137.
21. Csorba, M. (1997). *Simultaneous Localisation and Map Building*, Doktora Tezi, Oxford Üniversitesi Robotik Araştırma Grubu Mühendislik Bilimi Bölümü, Oxford.
22. Chong, K. S. ve Kleeman, L. (1999). Feature-Based Mapping in Real, Large Scale Environments Using an Ultrasonic Array, *Int. J. Rob. Res.*, 18(1), 3–19.
23. Leonard, J. J. ve Feder, H. J. S. (2000). *A Computationally Efficient Method for Large-Scale Concurrent Mapping and Localization*, Proceedings Robotics Research: the Ninth International Symposium, 169–176, Londra.
24. Guivant, J., Nebot, E. ve Baiker, S. (2000). Localization and map building using laser range sensors in outdoor applications, *J. Robot. Syst.*, 17(10), 565–583.
25. Castellanos, J. A., Martínez, J. M., Neira, J. ve Tardós, J. D. (1998). Experiments in Multisensor Mobile Robot Localization and Map Building, *IFAC Proc. Vol.*, 31(3), 69–374.
26. Thrun, S., Burgard, W. ve Fox, D. (1998). A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots, *Mach. Learn.*, 31(1-3), 29–53.
27. Kurt-Yavuz, Z. ve Yavuz, S. (2012). *A comparison of EKF, UKF, FastSLAM2.0, and UKF-based FastSLAM algorithms*, IEEE 16th International Conference on Intelligent Engineering Systems (INES), 37–43, Lizbon.
28. Rullán-Lara, L., Salazar S. ve Lozano, R. (2011). Real-time localization of an UAV using Kalman filter and a Wireless Sensor Network, *J. Intell. Robot. Syst.*, 65(1), 283–293.
29. Parra-Tsunekawa, I., Ruiz-del-Solar, J. ve Vallejos, P. (2015). A Kalman-filtering-based Approach for Improving Terrain Mapping in off-road Autonomous Vehicles, *J. Intell. Robot. Syst. Theory Appl.*, 78(3-4), 577–591.

30. Ahmad, H. ve Namerikawa, T. (2013). Extended Kalman filter-based mobile robot localization with intermittent measurements, *Syst. Sci. Control Eng.*, 1(1), 113–126.
31. Huang, G. P., Mourikis, A. I., ve Roumeliotis, S. I. (2008). *Analysis and improvement of the consistency of extended Kalman filter based SLAM*, Proceedings - IEEE International Conference on Robotics and Automation, 473–479, Kaliforniya.
32. Duan, H. (2017). *The application of AUV navigation based on cubature Kalman filter*, IEEE Underwater Technology (UT), 1–4, Busan.
33. Kim, K. H., Lee, J. G. ve Park, C. G. (2009). Adaptive Two-Stage Extended Kalman Filter for a Fault-Tolerant INS-GPS Loosely Coupled System, *IEEE Trans. Aerosp. Electron. Syst.*, 45(1), 125–137.
34. Gao, B., Gao, S., Hu, G., Zhong, Y. ve Gu, C. (2018). Maximum likelihood principle and moving horizon estimation based adaptive unscented Kalman filter, *Aerosp. Sci. Technol.*, 73, 184–196.
35. Logothetis, A. ve Krishnamurthy V. (1999). Expectation maximization algorithms for MAP estimation of jump Markov linear systems, *IEEE Trans. Signal Process.*, 47(8), 2139–2156.
36. Moral, P. (1997). Nonlinear Filtering: Interacting Particle Resolution, *Comptes Rendus l'Academie des Sci. - Ser. I Math.*, 325(6), 653–658.
37. Liu, J. S. ve Chen, R. (1998). Sequential monte carlo methods for dynamic systems, *J. Am. Stat. Assoc.*, 93(443), 1032–1044.
38. Arulampalam, S., Maskell, S., Gordon, N. ve Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, *IEEE Trans. Signal Process.*, 50(2), 174–188.
39. Nie, F., Zhang, W., Yao, Z., Shi, Y., Li, F. ve Huang, Q. (2020). LCPF: A Particle Filter Lidar SLAM System with Loop Detection and Correction, *IEEE Access*, 8, 20401–20412.
40. Carpenter, J. ve Clifford, P. (1999). Improved particle filter for nonlinear problems, *IEE Proc. Radar, Sonar Navig.*, 146(1), 2–7.
41. Bailey, T. ve Durrant-Whyte, H. (2006). Simultaneous localization and mapping (SLAM): Part II, *IEEE Robot. Autom. Mag.*, 13(3), 108–117.
42. Sasiadek, J. Z., Monjazez, A. ve Neculescu, D. (2008). *Navigation of an autonomous mobile robot using EKF-SLAM and FastSLAM*, Mediterranean Conference on Control and Automation - Conference Proceedings, MED'08, Ajaccio, Fransa, 517–522.
43. Farhadi, F., Mohammad, Y., Khosrowjerdi, J., Farhadi, Yadkuri, F., Khosrowjerdi, M. J. ve Yadkuri, F. F. (2014). Methods for Improving the Linearization Problem of Extended Kalman Filter, *J. Intell. Robot. Syst.* 2014 783,78(3), 485–497.

44. Suwoyo, H., Tian, Y., Wang, W., Hossain, M. M. ve Li, L. (2019). A Mapaekf-Slam Algorithm With Recursive Mean And Covariance Of Process And Measurement Noise Statistic, *SINERGI*, 24(1), 37.
45. Myung, I. J. (2003). Tutorial on maximum likelihood estimation, *J. Math. Psychol.*, 47(1), 90–100.
46. Arulampalam, M. S., Maskell, S., Gordon, N. ve Clapp, T. (2007). A tutorial on particle filters for online nonlinear/nongaussian bayesian tracking, *Bayesian Bounds Param. Estim. Nonlinear Filtering/Tracking*, 50(2), 723–737.
47. Moon, T. K. (1996). The Expectatio Maximization Algorithm, *IEEE Signal Processing Magazine*, 47–60,.
48. Do, C. B. ve Batzoglou, S. (2008). What is the expectation maximization algorithm?, *Nat. Biotechnol.*, 26(8), 897–899.
49. Tumminello, M., Lillo, F. ve Mantegna, R. N. (2007). Kullback-Leibler distance as a measure of the information filtered from multivariate data, *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, 76(3).
50. Briers, M., Doucet, A. ve Maskell, S. (2010). Smoothing algorithms for state–space models, *Ann. Inst. Stat. Math.*, 62(1), 61–89.
51. Biswas, K. K. ve Mahalanabis, A. K. (1972). An Approach to Fixed-Point Smoothing Problems, *IEEE Trans. Aerosp. Electron. Syst.*, AES-8(5), 676–682.
52. Gustafsson, F. (2010). Particle filter theory and practice with positioning applications, *IEEE Aerosp. Electron. Syst. Mag.*, 25(7) pp. 53–81.
53. Elfring, J., Torta, E. ve van de Molengraft, R. (2021). Particle Filters: A Hands-On Tutorial, *Sensors*, 21(2), 438.
54. Montemerlo, M. ve Thrun, S. (2003). *Simultaneous localization and mapping with unknown data association using FastSLAM*, Proc. - IEEE Int. Conf. Robot. Autom., Taipei, Taiwan, 1985–1991.
55. Douc, R., Cappé, O. ve Moulines, E. (2005). *Comparison of resampling schemes for particle filtering*, Image Signal Process. Anal.. ISPA 2005. Proc. 4th Int. Symp., Nanjing, China, 64–69.
56. Hol, J. D (2004). *Resampling in particle filters*, Yüksek Lisans Tezi, Linköping Üniversitesi Elektrik-Elektronik Mühendisliği Bölümü, Linköping, İsveç.
57. Gordon, N. J., Salmond, D. J. ve Smith, A. F. M. (1993). Novel approach to nonlinear/non-gaussian Bayesian state estimation, *IEE Proceedings, Part F Radar Signal Process.*, 140(2), 107–113.
58. Kitagawa, G. (1996). Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models, *J. Comput. Graph. Stat.*, 5(1), 1–25.

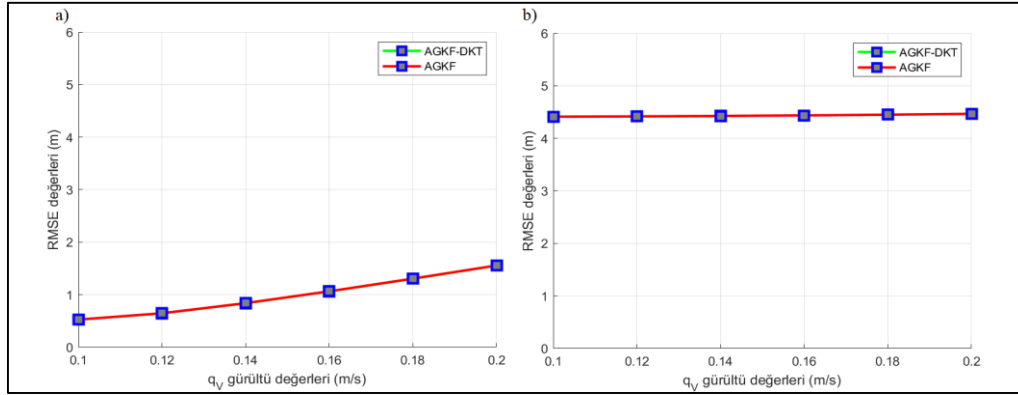
59. Carpenter, J., Clifford, P. ve Fearnhead P. (1999). Improved particle filter for nonlinear problems, *IEE Proc. - Radar, Sonar Navig.*, 146(1), 2.
60. Liu, J. S. ve Chen, R. (1998). Sequential Monte Carlo Methods for Dynamic Systems, *J. Am. Stat. Assoc.*, 93(443), 1032–1044.
61. Beadle, E. R. ve Djuric, P. M. (1997). A fast-weighted Bayesian bootstrap filter for nonlinear model state estimation, *IEEE Trans. Aerosp. Electron. Syst.*, 33(1), 338–343.
62. Sankaranarayanan, A. C., Srivastava, A. ve Chellappa, R. (2008). Algorithmic and Architectural Optimizations for Computationally Efficient Particle Filtering, *IEEE Trans. Image Process.*, 17(5), 737–748.
63. Bailey, T., Nieto, J., Guivant, J., Stevens, M. ve Nebot E. (2006). *Consistency of the EKF-SLAM algorithm*, IEEE Int. Conf. Intell. Robot. Syst., Beijing, China 3562–3568.
64. Bar-Shalom, Y., Li, X.-R. ve Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation*. New York, John Wiley & Sons, Inc..
65. Bahraini, M., Bozorg, M. ve Rad, A. (2018). A new adaptive UKF algorithm to improve the accuracy of SLAM, *Int J Robot*, 5(1), 1–12.
66. Nieto, J., Guivant, J., Nebot, E. ve Thrun, S. (2003). *Real time data association for FastSLAM*, IEEE International Conference on Robotics and Automation, Taipei, Taiwan 412–418,
67. Tang, M., Chen, Z. ve Yin, F. (2021). An Improved Adaptive Unscented FastSLAM with Genetic Resampling, *Int. J. Control. Autom. Syst.*, 19(4), 1677–1690.
68. Martinelli, F. (2019). Simultaneous Localization and Mapping Using the Phase of Passive UHF-RFID Signals, *J. Intell. Robot. Syst. Theory Appl.*, 94(3-4), 711–725.



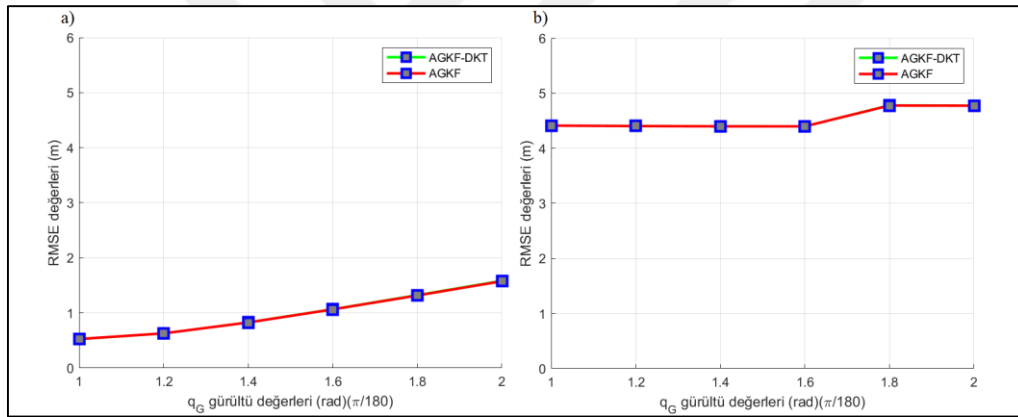


EKLER

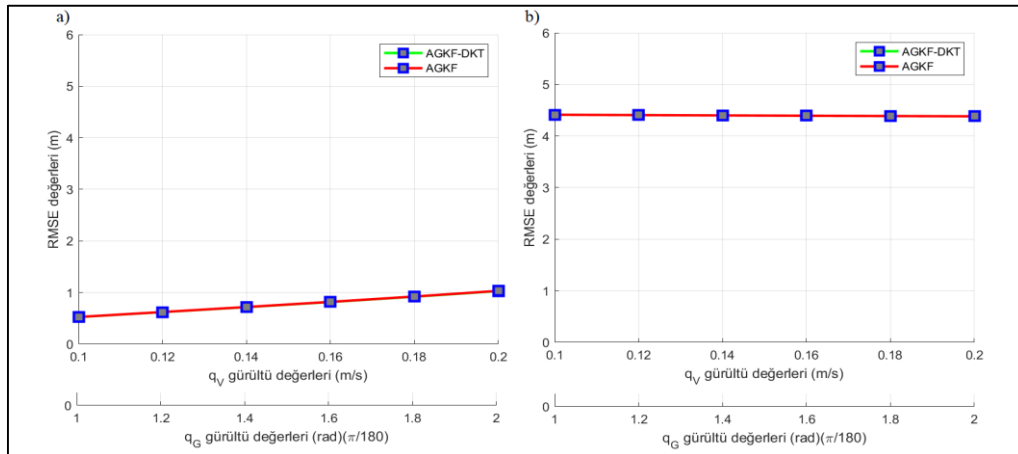
EK-1. AGKF ve AGKF-DKT'nin veri ilişkilendirmesi robota verildiği zaman değişen gürültü başlangıç değerleri için hesaplanan RMSE değerlerinin gösterildiği grafikler



Şekil 1.1. q_v değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu

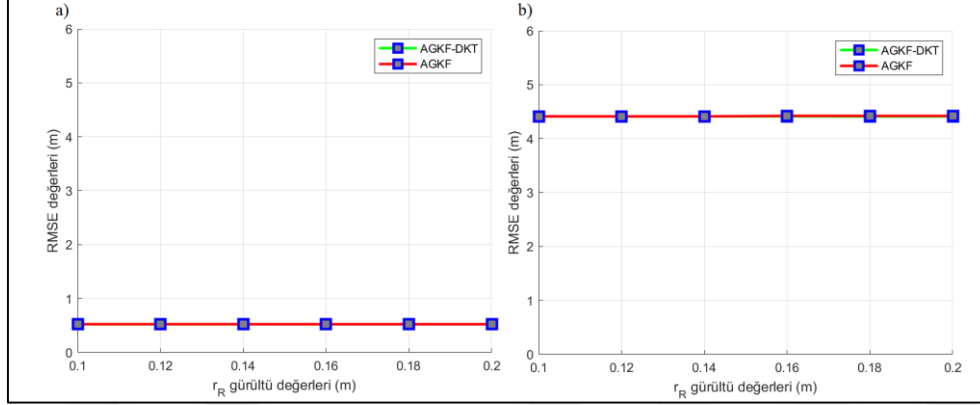


Şekil 1.2. q_G değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu

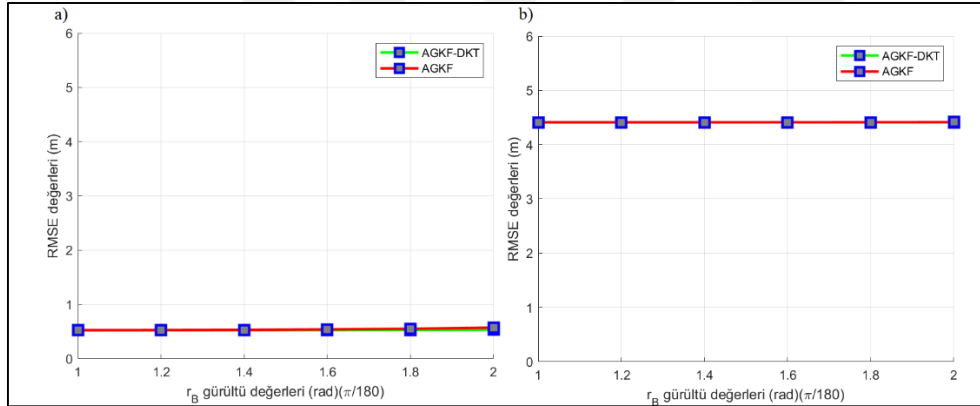


Şekil 1.3. q_v ve q_G değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu

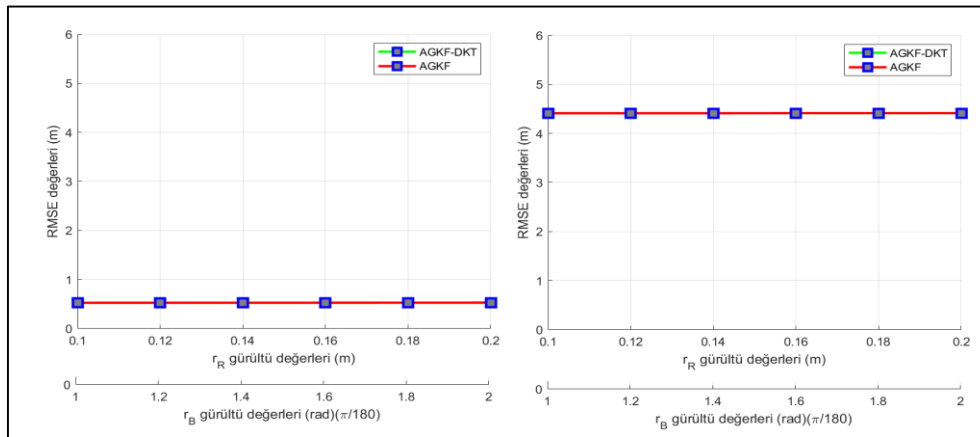
EK-1. (devam) AGKF ve AGKF-DKT'nin veri ilişkilendirmesi robota verildiği zaman değişen gürültü başlangıç değerleri için hesaplanan RMSE değerlerinin gösterildiği grafikler



Şekil 1.4. r_R değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu



Şekil 1.5. r_B değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu



Şekil 1.5. r_R ve r_B değişimine göre hesaplanan RMSE değerleri. a) robot konumu b) yer işaretlerinin konumu



GAZİ GELECEKTİR..