T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YOĞUN KARGAŞA ORTAMINDA ÇOKLU HAVA HEDEF TAKİBİ

KÜBRA TURGUT

YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

GEBZE
2018

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YOĞUN KARGAŞA ORTAMINDA ÇOKLU HAVA HEDEF TAKİBİ

KÜBRA TURGUT
YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
YRD. DOÇ. DR. ALİ KÖKSAL HOCAOĞLU

GEBZE
2018

T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MULTIPLE AIR TARGETS IN HEAVY CLUTTER

KÜBRA TURGUT

A THESIS SUBMITTED FOR THE DEGREE OF

MASTER OF SCIENCE

DEPARTMENT OF ELECTRONIC ENGINEERING

THESIS SUPERVISOR

ASSIST. PROF. DR. ALİ KÖKSAL HOCAOĞLU

GEBZE

2018

GEBZE TEKNİK ÜNİVERSİTESİ	**YÜKSEK LİSANS JÜRİ ONAY FORMU**

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 10/01/2018 tarih ve 2018/03 sayılı kararıyla oluşturulan jüri tarafından 25/01/2018 tarihinde tez savunma sınavı yapılan Kübra TURGUT'un tez çalışması Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE (TEZ DANIŞMANI)	: Yrd. Doç. Dr. Ali Köksal HOCAOĞLU
ÜYE	: Yrd. Doç. Dr. Serdar Süer ERDEM
ÜYE	: Prof. Dr. İbrahim ALTUNBAŞ

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun

...../...../..... tarih ve ...../..... sayılı kararı.

İMZA/MÜHÜR

ÖZET

Bu tez çalışması kapsamında, çoklu hedef takibindeki önemli aşamalardan biri olan iz başlatma aşaması için hatalı izlerin sayısının azaltılmasına yönelik doğrulamaya dayalı iki yöntem geliştirilmiştir. Ayrıca, gerçek zamanlı çalışabilecek kadar hızlı ve basit olması, diğer yandan bunları yaparken de takip performansının düşmemesini amaçlayan ve takip sistemlerinde takip performansını olumsuz etkileyen durumlar (hedefin manevra yapması, sensörlerden kaynaklı sapmalar, düzensiz ölçüm aralıkları, ölçüm gürültüleri, ölçümlerin alındığı koordinata göre ölçüm dönüşümlerinden kaynaklı sapmaların oluşumu gibi durumlar) karşısında hedef takibinde kullanılan algoritmaların daha güçlü olması üzerine çalışılmakta ve takip performansları incelenmektedir. Bu amaçla farklı Etkileşimli Çoklu Model (IMM, *ing.* Interacting Multiple Model) algoritmalarına koordinat dönüşümlerinden kaynaklı sapmanın azaltılmasına yönelik literatürde mevcut olan Dönüştürülmüş Ölçümler Kalman Filtresi (CMKF, *ing.* Converted Measurements Kalman Filtrer) uyarlanarak üç boyutlu hareket modeli için performans değerlendirmesi yapılmıştır.

Anahtar Kelimeler: Hedef Takibi, Etkileşimli Çoklu Model, Dönüştürülmüş Ölçümler Kalman Filtresi, Sensör Sapması, Düzensiz Örnekleme

SUMMARY

In this study, we propose two validation-based algorithms to reduce the number of false tracks for track initiation stage, which is one of the key steps to track multiple targets. Also, we study the target tracking algorithms and investigated their tracking performances fort he case of being fast and simple enough to be able to work in real time. We aim not to decrease the tracking performance in situations which negatively affect the tracking perofrmance, such as target maneuvering, sensor biases, irregular measuremnts from targets, measurement noises, bias due to measurement transformations. For this purpose, we adapt Converted Measurements Kalman Filter to reduce the bias resulting from coordinate transformations to Interacting Multiple Model algorithms and evaluate their performances for three-dimensional motion model.

Key Words: Target Tracking, Interacting Multiple Model, Converted Measurements Kalman Filter, Sensor Bias, Irregular Sampling

TEŞEKKÜR

Başta, yüksek lisans eğitimimde, akademik hayatımda ve her başım sıkıştığında desteğini ve yardımlarını hiçbir zaman esirgemeyip bilgisi ile bu çalışmanın oluşmasının yolunu açan danışmanım Yrd. Doç. Dr. Ali Köksal HOCAOĞLU'na,

Yardımlarını hiçbir zaman esirgemeyen çalışma arkadaşım Hasan Hüseyin SÖNMEZ'e,

Ve her zaman yanımda duran, bana destek olan ve sevgilerini her daim hissettiğim, özellikle annecim NURAN KUL'a, babacım Muammer KUL'a, eşim İsa TURGUT'a ve diğer aile üyelerime teşekkürlerimi sunarım.

İÇİNDEKİLER

	Sayfa
ÖZET	v
SUMMARY	vi
TEŞEKKÜR	vii
İÇİNDEKİLER	viii
SİMGELER ve KISALTMALAR DİZİNİ	xi
ŞEKİLLER DİZİNİ	xiii
TABLOLAR DİZİNİ	xv
1. GİRİŞ	1
1.1. Tezin Amacı, Katkısı ve İçeriği	2
1.2. Motivasyon	4
1.3. Tezin Organizasyonu	4
2. HEDEF TAKİBİ	5
2.1. Koordinat Sistemleri ve Hedef Hareket Modelleri	5
2.1.1. Kartezyen ve Kutupsal/ Küresel Koordinat Sistemleri	5
2.1.2. Kartezyen Koordinatlarda Hareket Modeli	7
2.1.2.1. İki Boyutta Hareket Modeli	7
2.1.2.2. Üç Boyutta Hareket Modeli	10
2.1.3. Küresel Koordinatlarda Hareket Modeli	11
2.2. Hedef Takip Algoritmaları	13
2.2.1. Kalman Filtresi	13
2.2.2. Genişletilmiş Kalman Filtresi	18
2.2.3. Kokusuz Kalman Filtresi	20
2.2.4. Dönüştürülmüş Ölçümler Kalman Filtresi	22
2.3. Sensör Sapması	23
2.4. Ölçümlerin Alındığı Koordinata Göre Sapma Oluşumu	26
3. RADAR SİSTEMLERİ İÇİN DÖNÜŞTÜRÜLMÜŞ ÖLÇÜMLER	31
3.1. CM Algoritması ile Sapma Etkisinin Azaltılması	31
3.2. İki Boyutta CM Algoritmaları	33
3.2.1. CM_L	33

3.2.2. CM_D 34
3.2.3. CM_U 35
3.2.4. CM_MU 36
3.2.5. CM_DU 37
3.3. İki Boyutlu CM için Performans Karşılaştırmaları 39
3.3.1. Yansızlık 39
3.3.2. Tutarlılık 43
3.3.3. Kalman Filtresine Uyarlanması 47
3.4. Üç Boyutta CM Algoritmaları 51
3.4.1. CM_U 52
3.4.2. CM_MU 53
3.4.3. CM_DU 54
3.5. Üç Boyutlu CM için Performans Karşılaştırmaları 57
3.5.1. Yansızlık 57
3.5.2. Tutarlılık 58
3.5.3. Kalman Filtresine Uyarlanması 61
4. ETKİLEŞİMLİ ÇOKLU MODEL 65
4.1. Genel Bilgi ve Kullanım Nedeni 65
4.2. Yöntem 66
4.3. IMM-CM Algoritmaları 69
4.4. Testler 72
4.5. Mod Olasılık Hesabında Markov Zinciri'nin Etkisi 84
4.6. SIMM-DIMM-MIMM Algoritmaları ve UKF ve CMKF ile kullanımı 86
4.6.1. SIMM, DIMM ve MIMM Algoritmaları 86
4.6.2. CMKF veUKF Uyarlamalı SIMM, DIMM ve MIMM Algoritmalarının Performans Karşılaştırmaları 89
5. ÇOKLU HEDEF TAKİBİ 96
5.1. Genel Bilgi 96
5.2. Düzensiz Örnekleme 99
5.3. Hatalı İzlerin Azaltılması 100
5.4. Testler 106
5.4.1. Ölçüm Aralıklarının Düzensiz Olması Durumu 107
5.4.2. Sensör Sapması Durumu 112

KAYNAKLAR 117
ÖZGEÇMİŞ 123
EKLER 124

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler ve Kısaltmalar		**Açıklamalar**
$\boldsymbol{I}$	:	Birim Matris
σ^2	:	Varyans
μ	:	Ortalama
$\boldsymbol{K}$	:	Kalman Kazanç Matrisi
$\boldsymbol{P}$	:	Kestirim Kovaryans Matrisi
$\boldsymbol{R}$	:	Ölçüm Gürültüsü Kovaryans Matrisi
$\boldsymbol{Q}$	:	Süreç Gürültüsü Kovaryans Matrisi
w	:	Süreç Gürültüsü
v	:	Ölçüm Gürültüsü
ANEES	:	Average Normalized Estimation Error Squared
ATC	:	Air Traffic Control
CA	:	Constant Acceleration
CDMKF	:	Converted Doppler Measurement Kalman Filter
CV	:	Constant Velocity
CM	:	Converted Measurements
CMKF	:	Converted Measurements Kalman Filter
CMKF_D	:	Converted Measurements Kalman Filter with Debiasing
CMKF_DU	:	Converted Measurements Kalman Filter with Decorrelated Unbiased
CMKF_L	:	Converted Measurements Kalman Filter with Linearization
CMKF_MU	:	Converted Measurements Kalman Filter with Modified Unbiasing
CMKF_U	:	Converted Measurements Kalman Filter with Unbiasing
CT	:	Coordinated Turn
DIMM	:	IMM weigted by diagonal matrices
EKF	:	Extended Kalman Filter
GM	:	Gaussian Mixture
IMM	:	Interacting Multiple Model
IPDA	:	Integrated Probability Data Association
KF	:	Kalman Filter

LS	:	Least Squares
MC	:	Monte Carlo
MIMM	:	IMM weigted by general matrices
MLE	:	Maximum Likelihood Estimator
MMSE	:	Minimum Mean Square Error
NES	:	Normalized Error Squared
RLS	:	Recursive Least Squares
RMSE	:	Root Mean Square Error
SIMM	:	IMM weigted by scalars
SF-CMKF	:	Statically Fused Converted Measurement Kalman Filter
SPRT	:	Sequential Probability Ratio Test
UKF	:	Unscented Kalman Filter

ŞEKİLLER DİZİNİ

Şekil No:		Sayfa
2.1:	İki boyutlu sistem.	6
2.2:	Üç boyutlu sistem.	6
2.3:	İki boyut için kartezyen koordinatta gözlemci-hedef ilişkisi.	8
2.4:	Üç boyut için kartezyen koordinatta gözlemci-hedef ilişkisi.	10
2.5:	İki boyut için polar koordinatta gözlemci-hedef ilişkisi.	11
2.6:	Üç boyut için küresel koordinatta gözlemci-hedef ilişkisi.	12
2.7:	Kalman Filtresi Akış Diyagramı .	14
2.8:	Radar ölçümlerinin küreselde alınması durumunda Kalman Filtresi akış diyagramı.	28
2.9:	Ölçümlerin küresel koordinattan kartezyen koordinata dönüşümü sonucunda oluşan sapma etkisi.	29
2.10:	Polar ve kartezyen koordinatlarda elde edilen MC sonucu, gerçek konum ve dönüşüm sonucu elde edilen konum karşılaştırması.	30
3.1:	CMKF akış diyagramı.	47
4.1:	IMM akış diyagramı.	66
4.2:	Sabit Δt senaryosu için örnek senaryo görüntüsü.	72
4.3:	Sabit Δt senaryosu için hedef hız grafiği.	73
4.4:	Ölçüm gürültüsü fazla olan senaryo için RMSE grafiği (T=1 sn).	75
4.5:	IMM-CMKF_DU için örnek ölçüm, hedef rotası ve kestirim sonucu grafiği.	76
4.6:	IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği.	77
4.7:	IMM-CMKF_DU için örnek mod olasılık grafiği.	79
4.8:	Ölçüm gürültüsü az olan senaryo için RMSE grafiği (T=1 sn).	80
4.9:	IMM-CMKF_DU için örnek mod olasılık grafiği.	80
4.10:	Ölçüm gürültüsü fazla olan senaryo için RMSE grafiği (T=3 sn).	82
4.11:	Ölçüm gürültüsü fazla olan senaryo için RMSE grafiği (T=5 sn).	82
4.12:	KF için IMM algoritmaları karşılaştırması.	90
4.13:	UKF için IMM algoritmaları karşılaştırması.	91

4.14:	CMKF için IMM algoritmaları karşılaştırması.	92
4.15:	KF, UKF ve CMKF için SIMM algoritmaları karşılaştırması.	93
4.16:	CMKF için IMM ve SIMM algoritmaları ortalama kestirim grafiği.	94
5.1:	Önerilen iz doğrulama algoritmasına ait sözde kod.	102
5.2:	*K* kümesi için RANSAC örneği.	103
5.3:	Senaryo 8 için önerilen yöntem sonucunda oluşan izler.	105
5.4:	$\boldsymbol{\lambda}$=3 için örnekleme aralığı.	108
5.5:	λ=3 için takip algoritmalarının RMSE grafiği.	109
5.6:	λ=7 için örnekleme aralığı.	111
5.7:	λ=7 için takip algoritmalarının RMSE grafiği.	111
5.8:	Sensör sapma durumunda T=1 sn için RMSE grafiği.	112
5.9:	Sensör sapması durumunda IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği (T=1 sn).	113
5.10:	Sensör sapması durumunda IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği (T=5 sn).	113
5.11:	Sensör sapma durumunda T=5 sn için RMSE grafiği.	114
5.12:	Sensör sapma durumunda $\boldsymbol{\lambda}$=7 için RMSE grafiği.	115
5.13:	Sensör sapması durumunda IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği ($\boldsymbol{\lambda}$=7).	116

TABLOLAR DİZİNİ

Tablo No:		Sayfa
3.1:	CM dönüşüm çeşitleri için Gauss yansızlık testi .	40
3.2:	Polar ve kartezyen koordinattaki ölçümler ve CM çeşitleri için beklenen değerleri.	41
3.3:	İki boyutta CM yöntemlerinin dönüşüm yanlılığı.	42
3.4:	İki boyutta CM yöntemlerinin dönüşüm yanlılık çarpanı.	42
3.5:	İki boyutta dönüşümlerin NES grafik tablosu.	45
3.6:	İki boyutta dönüşümlerin x ve y eksenleri için NES grafik tablosu.	46
3.7:	İki boyutlu sistem için test senaryo bilgileri.	48
3.8:	İki boyutlu sistem için MSE ve ANEES performans grafikleri.	51
3.9:	Üç boyutta CM yöntemlerinin dönüşüm yanlılık çarpanı.	57
3.10:	Üç boyutta dönüşümlerin toplam ve x ekseni için NES grafik tablosu.	59
3.11:	Üç boyutta dönüşümlerin y ve z eksenleri için NES grafik tablosu.	60
3.12:	Üç boyutlu sistem için test senaryo bilgileri.	61
3.13:	Üç boyutlu sistem için hedef senaryo görüntüsü.	62
3.14:	Üç boyutlu sistem için RMSE ve ANEES performans grafikleri.	64
4.1:	Sabit Δt senaryoları için hedef rotası ve ölçümler.	74
4.2:	Ölçüm gürültüsü fazla olan senaryo için gerçekleme süresi.	75
4.3:	IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu manevra ayrıntı grafikleri.	77
4.4:	Ölçüm gürültüsü az olan senaryo için gerçekleme süresi.	79
4.5:	T değerlerine göre ölçüm sıklığının değişimi.	81
4.6:	IMM-CMKF_DU için farklı T değerinde ortalama kestirim grafikleri.	83
4.7:	Markov zinciri, bir MC için filtre ağırlık grafikleri.	85
4.8:	Markov zinciri, yüz MC için ortalama filtre ağırlık grafikleri.	85
4.9:	KF için IMM algoritmaları işlem süresi karşılaştırması.	90
4.10:	UKF için IMM algoritmaları işlem süresi karşılaştırması.	91
4.11:	CMKF için IMM algoritmaları işlem süresi karşılaştırması.	92

4.12:	KF, UKF ve CMKF için SIMM algoritmaları işlem süresi karşılaştırması.	93
4.13:	CMKF için IMM ve SIMM algoritmaları ortalama kestirim sonucu manevra ayrıntı grafikleri.	95
5.1:	Senaryolarda bulunan hedef sayısı.	104
5.2:	Önerilen yönteme ait simülasyon sonuçları.	106
5.3:	Farklı $\boldsymbol{\lambda}$ değerleri ile oluşturulan ölçüm grafikleri	107
5.4:	λ=3 için takip algoritmalarının gerçekleme süreleri.	109
5.5:	$\boldsymbol{\lambda}$=3 ve T=3 için elde edilen örnek ölçüm grafikleri.	110
5.6:	λ=7 için takip algoritmalarının gerçekleme süreleri.	110

1. GİRİŞ

Çoklu hedef takibi, günümüzde birçok alanda uygulamaya sahiptir. Hava savunması, navigasyon, hava trafik kontrolü, robotik, görüntü işleme, uzay gibi hem askeri hem de sivil alanlardaki uygulamalar buna örnek olarak verilebilir.

Hedef takip sistemlerinin, gerçek zamanlı çalışabilecek kadar hızlı ve basit olması ve aynı zamanda da takip performansının düşmemesi amaçlanmaktadır. Takip sistemlerinde takip performansını olumsuz etkileyen durumlar mevcuttur; hedefin manevra yapması, sensörlerden kaynaklı sapmalar, hedeften sürekli bilgi alınamaması, ölçüm gürültüleri bunlardan bazılarıdır. Hedef takibi algoritmalarının bu durumları dikkate alarak geliştirilmesi gerekmektedir.

Bu tez kapsamında, temel olarak iki konu çalışılmıştır. Birincisi, yoğun kargaşa ortamında hatalı izlerin sayısının azaltılmasıdır. Bu kapsamda, doğrulamaya dayalı iki yöntem geliştirilmiştir. Çoklu hedef takibi aşamalarından olan iz başlatma aşamasında oluşturulan hatalı iz sayısı yoğun kargaşa ortamında artmaktadır ve bu durum takip performansını olumsuz etkilemektedir. Bu nedenle hatalı izlerin sayısının azaltılmasına yönelik Hough dönüşümü ve RANSAC algoritmaları kullanılarak doğrulamaya dayalı iki yöntem geliştirilmiştir. Bu tez kapsamında ele alınan ikinci konu ise, koordinat dönüşümlerinden kaynaklı sapmanın giderilmesi ve manevralı hedef takibidir. Bu amaçla koordinat dönüşümlerinden kaynaklı sapmaların giderilmesi için Dönüştürülmüş Ölçümler Kalman Filtresi (CMKF, *ing.* Converted Measurements Kalman Filtrer) ve birden fazla hedef hareket modelinin kullanılmasına olanak sağlayan Etkileşimli Çoklu Model (IMM, *ing.* Interacting Multiple Model) algoritmaları üzerinde durulmaktadır.

CMKF algoritması farklı uygulamalarda kullanılmaktadır ve uygulamaya yönelik farklı CMKF algoritmaları geliştirilmiştir. Doppler radarının kullanıldığı sistemler için Doppler ölçümlerinin kullanılması durumunda Dönüştürülmüş Doopler Ölçümler Kalman Filtresi (CDMKF, *ing.* Converted Doppler Measurement Kalman Filter), Gauss Karışımı (GM, *ing.* Gaussian mixture) kestirim algoritması kullanılması durumunda GM-CDMKF algoritması kullanılmaktadır [1]. [2]'de İstatistiksel Kaynaştırılmış Dönüştürülmüş Ölçüm Kalman Filtresi (SF-CMKF, *ing.* Statically Fused Converted Measurement Kalman Filter) geliştirilmiştir. [3]'te ise geleneksel

ağırlıklı en küçük kareler füzyonu (*ing.* traditional weighted-least squares fusion) kullanılarak CMKF algoritmasının modifiye edilmiş hali öne sürülmektedir.

IMM algoritması, birçok farklı uygulamada kullanılmaktadır. Kontrol sistemlerindeki uygulamasına örnek olarak, [4]'te robot duruş tahmin sistemi geliştirilmiştir ve bu amaçla Kokusuz Kalman Filtresi (UKF, *ing.* Unscented Kalman Filter) tabanlı IMM filtresi kullanılmaktadır. [5]'te otonom araçlar için IMM tabanlı genişletilmiş nesne takipçisi öne sürülmektedir. [6]'da manevralı hedef takibi için geliştirilmiş uyarlamalı IMM filtresi üzerinde durulmaktadır. [7]'de ise parazit yankılı ortamda yer araçlarının takibi, IMM ve Entegre Olasılıksal Veri İlişkilendirme (IPDA, *ing*. Integrated Probabilistic Data Association) yöntemlerinin birleştirilmesiyle yapılmaktadır.

Bu tez kapsamında CMKF koordinat dönüşümlerinden kaynaklı sapmanın azaltılması için, IMM algoritması ise hedeflerin birden fazla modele göre hareket etmesi durumu için kullanılacaktır.

1.1. Tezin Amacı, Katkısı ve İçeriği

Bu çalışmada, çoklu hedef takibinde kullanılan bazı takip algoritmalarının düzensiz örneklemenin, sensör sapmasının ve ölçüm gürültüleri ile koordinat dönüşümlerinden kaynaklı sapmaların takip performansları üzerindeki etkileri incelenmekte ve olumsuz etkiyi azaltıcı çözüm önerilmektedir. Birçok hedef takip algoritmasındaki varsayımların aksine, hedefler sabit ivme veya sabit hız gibi sadece tek bir modele uygun olarak hareket etmezler. Hedef bazen sabit hızlı, bazen sabit ivmeli, bazen de manevra yapan bir davranış sergileyebilmektedir. Dolayısıyla, takip algoritmalarındaki hedef hareketi tek bir model ile modellendiğinde takip performansı düşmektedir. Oysa, birden fazla modelin birlikte kullanılması takip performansını iyileştirmektedir. Ayrıca, çoklu hedef takibinde kullanılan radar sistemleri ölçümleri küresel koordinatta almaktadır. Bu durum, takip algoritmalarında doğrusal olmayan bir ölçüm modeli kullanımını gerektirmektedir. Öte yandan, ölçüm modelinin doğrusal olması, daha basit ve hızlı filtrelerin kullanımı mümkün kılmaktadır. Küresel koordinatlarda sıfır ortalamalı Gauss gürültüsüne sahip ölçümlerin küresel koordinattan kartezyen koordinata dönüştürülmesi durumunda, kartezyen koordinattaki ölçüm gürültüsünün ortalama ve varyansını değiştirmektedir. Bu durum,

takip performansını olumsuz etkilemektedir. Bu tezde, koordinat dönüşümlerinden kaynaklı sapmaların etkisinin azaltılması için CMKF, birden fazla hareket modelinin kullanımını sağlayan IMM algoritması ile birlikte kullanılarak IMM-CMKF algoritması önerilmiştir. Böylece hedef takip sistemlerinde ölçüm dönüşümlerinden kaynaklı sapmayı azaltan, hesaplama yükü az olan ve hedef hareket modellerine uyum sağlayan bir algoritma üzerinde durulmuştur. Ayrıca takip performansını olumsuz etkileyen ölçüm aralıklarının düzensiz olması ve sensör sapmasının mevcut olduğu durumlarda takip algoritmalarının performansları karşılaştırılarak, bahsi geçen durumlarda hangi algoritmanın kullanılmasının daha tutarlı olacağı gösterilmiştir. Ayrıca literatürde IMM filtresinin mod ağırlık hesabına değinerek füzyon kriterlerine dayalı geliştirilen skaler ile ağırlıklandırılmış IMM (SIMM, *ing.* IMM weigted by scalars), köşegen matrisle ağırlıklandırılmış IMM (DIMM, *ing.* IMM weigted by diagonal matrices) ve genelleştirilmiş matrisle ağırlıklandırılmış IMM (MIMM, *ing.* IMM weigted by general matrices) algoritmaları [8]'de önerilmektedir. Önerilen IMM algoritmaları, literatürde iki boyutta Kalman filtresi ile kullanılmıştır. Bu tez kapsamında, Kalman filtresinin kullanıldığı IMM algoritmalarının testi üç boyuta genişletilmiştir. Buna ek olarak koordinat dönüşümlerinden kaynaklı sapmanın azaltılması için kullanılan CMKF algoritması SIMM, DIMM ve MIMM algoritmalarına uyarlanarak SIMM-CMKF, DIMM-CMKF ve MIMM-CMKF algoritmaları öne sürülmüştür. Ayrıca, sık kullanılan UKF filtresi de SIMM, DIMM ve MIMM algoritmalarına uyarlanarak SIMM-UKF, DIMM-UKF ve MIMM-UKF algoritmaları geliştirilmiştir. Buna ek olarak Kalman filtresi, CMKF ve UKF algoritmalarının kullanıldığı IMM algoritmaları simüle edilen manevralı veriyle test edilerek karşılaştırılmıştır.

Bu tezin literatüre katkısı, takip algoritmaları için ölçümlerin periyodik olmayan aralıklarda alınması durumu için takip algoritmalarının uyarlanması, düzensiz örnekleme durumunda takip algoritmalarının performans karşılaştırmaları, hedef takibindeki hatalı izlerin azaltılması için doğrulamaya dayalı iki yöntemin geliştirilmesi ve CMKF algoritmasının farklı IMM algoritmalarına uyarlanarak IMM-CMKF algoritmasından daha az hataya sahip yeni bir algoritmanın geliştirilmesidir. Sözkonusu katkıların takip performansını iyileştirdiği gösterilmiştir.

1.2. Motivasyon

Günümüzde, hedef takip sistemlerinden hava hedef takibi, gerek askeri alanda gerekse sivil alanda önem taşıyan bir konudur. Hedef takip sistemlerinde hızlı, fazla hesap yükü gerektirmeyen ve takip performansını iyileştirmeye yönelik bir çalışma hem bilimsel hem de uygulamaya yönelik olması ilgi çekicidir. Ayıca takip sistemlerinde takip performansını olumsuz etkileyen bazı durumlarda, hedef takip algoritmalarının takip performanslarının bilinmesi uygulamaya yönelik takip algoritmasının seçiminde veya algoritmadaki olumsuz performansın giderilmesi için hangi durumlarda performansın olumsuz etkilendiğinin bilinmesi önemlidir. Bu nedenle, bu konuda çalışmamızdaki motivasyonumuz, hedef takip sistemlerinde kolayca uygulanabilir, hesap yükü fazla olmayan, gerçek zamanlı çalışabilecek kadar hızlı olan hedef takip algoritmalarının geliştirilmesi ve hedef takibinde karşılaşılan ve takip performansının olumsuz etkilendiği bazı durumlarda takip algoritmalarının performans karşılaştırılmasının yapılmasıdır.

1.3. Tezin Organizasyonu

Tezin ileriki bölümleri şu şekilde düzenlenmiştir. Bölüm 2'de hedef takip, hedef takipte kullanılan koordinat sistemleri, hedef takip algoritmaları hakkında kısaca bilgi verilerek, sensör sapması durumunda sapma modeli ve koordinat dönüşümlerinden kaynaklı sapmanın oluşumuna değinilmektedir. Bölüm 3'te takip algoritmalarından biri olan ve bu tezdeki konuyu ele alan CMKF algoritması üzerinde durulmuştur. Bölüm 4'te, takip algoritmalarında performansın artırılması için geliştirilmiş IMM algoritması hakkında bilgi verilmektedir ve CMKF gibi takip filtrelerinin IMM algoritması ile kullanımı mevcuttur. Ayrıca farklı IMM algoritmalarının CMKF ve UKF algoritmaları ile kullanımı ve elde edilen algoritmaların testleri yapılarak test sonuçları verilmektedir. Bölüm 5'te ise çoklu hedef takibi, düzensiz örnekleme ve sensör sapması durumunda takip algoritmaları, çoklu hedef takibindeki iz başlatma sonrasında elde edilen hatalı izlerin doğrulama yöntemi ile doğrulanarak hatalı izlerin elenmesi konusu ele alınmıştır ve son olarak tezde bahsedilen takip algoritmalarının farklı senaryolardaki performans analizleri verilmektedir.

2. HEDEF TAKİBİ

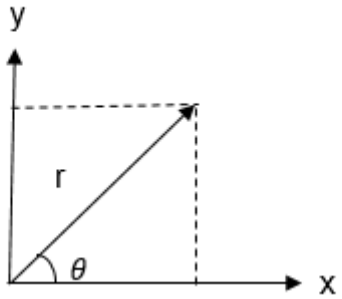
Hedef takibi, hedefe ait önceki kestirilen konum, hız, ivme gibi bilgiler ile hedefin o anki bilgilerinin kullanılarak hedefin bir sonraki durumunun kestirilmesidir. Hedef takibinde kullanılan algoritmalar, hedef hareketine, radar sistemine, ölçümlerin alındığı koordinat sistemine ve ihtiyaca göre şekillenmektedir. Takip algoritmaları hedef takibinin önemli bir parçasıdır. Bu nedenle bu bölümde temel takip algoritmaları ve bu algoritmaların kullanımı için gerekli olan hedef hareket modelleri üzerinde durulmaktadır.

2.1. Koordinat Sistemleri ve Hedef Hareket Modelleri

Hareket halindeki cisimlerin hareket dinamiğinin modellenmesi hem takip algoritmalarının performans karşılaştırılmalarının yapılması açısından hem de takip için kullanılacak yöntemlerin belirlenmesi açısından önemli rol oynuyor. Hedef hareketinin modellenmesi konusu ele alındığında hedefin sabit hızlı, sabit ivmeli, manevra yapıp yapmadığı gibi hareket dinamiği ile ilgili parametrelere ilave olarak sistem modelinde kullanılacak olan koordinat sisteminin polar, küresel ya da kartezyen koordinat olup olmaması ile takipte kullanılan bileşenlerin yalnızca kerteriz, menzil ve kerteriz, üç boyutlu konum bilgisi olup olmaması da önemli olmaktadır. Bu bölümde, sonraki bölümlere referans olması ve notasyon birliği açısından öncelikle farklı koordinat sistemleri hakkında bilgi verilecek ve farklı koordinat sistemleri arasındaki dönüşümler ele alınacaktır.

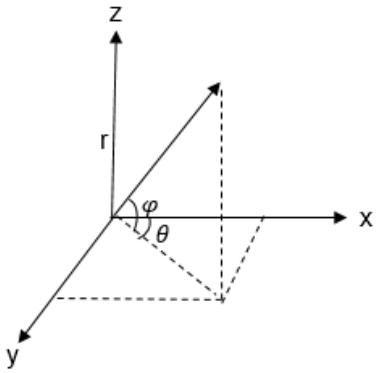
2.1.1. Kartezyen ve Kutupsal/ Küresel Koordinat Sistemleri

Hedef takibinin yapıldığı iki-boyutlu sistemler için genel gösterim Şekil 2.1'de gösterildiği gibidir. Kartezyen koordinatlarda (x, y) bileşenleri varken kutupsal koordinatlarda bu bileşenler (r, θ)'ya dönüşmektedir.



Şekil 2.1: İki boyutlu sistem.

Üç-boyutlu sistemler için ise genel gösterim Şekil 2.2'de gösterilmektedir. Kartezyen koordinatlardaki (x, y, z) bileşenleri, küresel koordinatlarda (r, θ, φ)'ye dönüşmektedir. Şekil 2.2'de görüldüğü üzere r mezili, θ yanal açıyı, φ ise düşey açıyı temsil etmektedir.



Şekil 2.2: Üç boyutlu sistem.

Küresel koordinat sisteminden kartezyen koordinata dönüşüm aşağıdaki gibidir.

$$x = r\ cos\varphi\ cos\theta \tag{2.1}$$

$$y = r\ cos\varphi\ sin\theta \tag{2.2}$$

$$z = r\ sin\varphi \tag{2.3}$$

Küresel koordinattan kartezyen koordinata dönüşüm ise aşağıda verildiği şekilde yapılmaktadır.

$$r = \sqrt{x^2 + y^2 + z^2} \tag{2.4}$$

$$\theta = \tan^{-1}(y/x) \tag{2.5}$$

$$\varphi = \tan^{-1}(z/\sqrt{x^2 + y^2}) \tag{2.6}$$

Kartezyen ve küresel koordinatlarda hız aşağıdaki gibi hesaplanır [9]. Denklem (2.8)'deki r mezili metre olarak alınırsa, v_r hız bileşeninin birimi m/sn, v_θ ve v_φ hız bileşenlerinin birimi ise 1/sn olmaktadır.

$$\text{Kartezyen koordinat için } v = \sqrt{v_x^2 + v_y^2 + v_z^2} \tag{2.7}$$

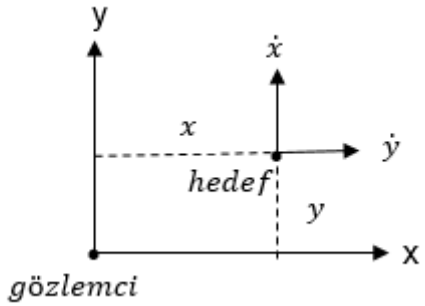
$$\text{Küresel koordinat için } v = \sqrt{v_r^2 + r^2\, sin^2\varphi\, v_\theta^2 + r^2\, v_\varphi^2} \tag{2.8}$$

2.1.2. Kartezyen Koordinatlarda Hareket Modeli

Hareket modellerinde genelde konum ve hız ilişkileri mevcutken bazı durumlarda ivme ilişkisi de bulunmaktadır. Bu çalışmada, hareket modelleri sabit hızlı ve sabit ivmeli olmak üzere iki durumda incelenecektir.

2.1.2.1. İki Boyutta Hareket Modeli

İki boyutta hareket modeli için Şekil 2.1'de gösterilen koordinat sistemi geçerli olmaktadır. Gözlemci ve hedefin koordinat sistemi üzerindeki ilişkisi ise Şekil 2.3'te belirtilmektedir.



Şekil 2.3: İki boyut için kartezyen koordinatta gözlemci-hedef ilişkisi.

Sabit hız modeli, manevra yapmayan hedefin hareket modelini temsil etmektedir ve iki boyutta kullanılan temel parametreler şöyledir;

$$\text{İki boyutlu sistem için } \boldsymbol{x} = [x, y, \dot{x}, \dot{y}]^T \tag{2.9}$$

İki boyutlu sistemde $\boldsymbol{x}$ durum vektörü için x ve y konum bileşenlerini, $\dot{x}$ ve $\dot{y}$ ise hız bileşenlerini temsil etmektedir.

Denklem (2.10) hedefin sürekli zaman hareket denklemidir. Burada, w_x ve w_y, *x-y* eksenleri boyunca karşı gelen gürültü bileşenleridir ve bu bileşenleri temsil etmek için ortalaması sıfır olan Gauss gürültüsü kullanılmaktadır.

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \ddot{x} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} . \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} . \begin{bmatrix} w_x \\ w_y \end{bmatrix} = \boldsymbol{F}x + \boldsymbol{\Gamma}w \tag{2.10}$$

Tez kapsamında sürekli zaman değil ayrık zaman hareket denklemleri kullanılacaktır. Bu nedenle doğrusallaştırma işlemi yapılmalıdır. [10]'ye göre Denklem (2.10)'daki sürekli zaman hareket modelinin çözümü Denklem (2.11)'deki gibidir.

$$\boldsymbol{x}_{t+T} = \boldsymbol{F}^{CV}(t + \Delta t, t)\, \boldsymbol{x}_k + \int_t^{\Delta t + t} \boldsymbol{F}^{CV}(t + \Delta t, \tau)\, \boldsymbol{\Gamma} w\, d\, \tau \tag{2.11}$$

Burada $\boldsymbol{F}^{CV}(t+\Delta t, t)$ durum geçiş matrisidir. Zamanla değişmeyen sistem için, t = 0 varsayımı yapılınca çözüm $e^{A\Delta t}$ olmaktadır ve $e^{A\Delta t}$'nin Taylor seri açılımı Denklem (2.13)'te belirtilmiştir [10].

$$\boldsymbol{F}^{CV}(\Delta t) = \boldsymbol{F}^{CV}(\Delta t, 0) = e^{A\Delta t} \tag{2.12}$$

$$e^{A\Delta t} = \sum_{i=0}^{\infty} \frac{(A\Delta t)^i}{i!} = I + A\Delta t + \frac{(A\Delta t)^2}{2!} + \cdots \tag{2.13}$$

Çıkan sonuca göre iki boyutlu sistem için seri açılımının sadece ilk iki teriminin sıfırdan farklı olması durumunda çözüm basitçe bulunabilir. İki boyutlu denklem için hareket denklemleri Denklem (2.14)'teki gibi olur ve zaman (t) yerine adım (k); Δt yerine T kullanılarak sürekli zaman hareket denklemi ayrık zamanlı hareket denklemine dönüştürülebilir. Burada kullanılan T ise adımlar arası geçen süre olarak düşünülebilir. Sonuç olarak hareketli hedefin iki boyuttaki ayrık zamanlı hareket modeli Denklem (2.14)' te belirtildiği gibidir [10].

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}^{CV}.\boldsymbol{x}_k + \boldsymbol{\Gamma}^{CV}.\boldsymbol{w}_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.\boldsymbol{x}_k + \begin{bmatrix} {T^2}/{2} & 0 \\ 0 & {T^2}/{2} \\ T & 0 \\ 0 & T \end{bmatrix}.\boldsymbol{w}_k \tag{2.14}$$

Burada w_k gürültüsüyle temsil edilen genelde bilinmeyen ivmelerdir. Bu ivme değerleri genelde beyaz gürültü ivme modeli (*white-noise acceleration model*) olarak bilinen ve ivmeleri beyaz gürültü gibi bağımsız süreçler olan model ile temsil edilir. Sabit hız modelinde, eğer hedefin manevra yapması temsil edilmek isteniyorsa ivme için kullanılan gürültünün normalden daha yüksek bir değerde olması gerekmektedir [10].

İki boyutta sabit ivmeli model için kullanılan temel parametreler ise aşağıda verildiği gibidir ve denklemde yer alan $\ddot{x}$ ve $\ddot{y}$ ivme bileşenlerini temsil etmektedir.

$$\boldsymbol{x} = [x, y, \dot{x}, \dot{y}, \ddot{x}, \ddot{y}]^T \tag{2.15}$$

Sabit ivme modeli için hareket denklemi Denklem (2.16)'da, hareket modeli ise Denklem (2.17)'de belirtilmektedir.

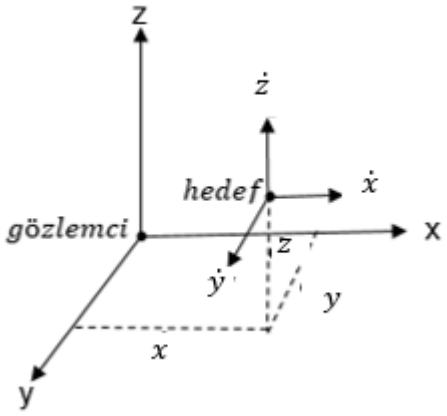
$$\boldsymbol{x}_{k+1} = \boldsymbol{F}^{CA}.\boldsymbol{x}_k + \boldsymbol{\Gamma}^{CA}.\boldsymbol{w}_k \tag{2.16}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}^{CA}.\boldsymbol{x}_k + \boldsymbol{\Gamma}^{CA}.\boldsymbol{w}_k = \begin{bmatrix} I_2 & T.I_2 & {}^{T^2}/_{2}.I_2 \\ 0 & I_2 & T.I_2 \\ 0 & 0 & I_2 \end{bmatrix}.x_k + \begin{bmatrix} {}^{T^2}/_{2}.I_2 \\ T.I_2 \\ I_2 \end{bmatrix}.w_k \tag{2.17}$$

Hareket modelinde kullanılan I_2 2x2'lik birim matrisi temsil etmektedir.

2.1.2.2. Üç Boyutta Hareket Modeli

Üç boyutta Şekil 2.2'de gösterilen koordinat sistemi geçerli olmaktadır. Gözlemci ve hedefin koordinat sistemi üzerindeki ilişkisi ise Şekil 2.4'te belirtilmektedir.



Şekil 2.4: Üç boyut için kartezyen koordinatta gözlemci-hedef ilişkisi.

Üç-boyutta kullanılan temel parametreler şu şekildedir:

$$\boldsymbol{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T \tag{2.18}$$

I_3 birim matrisi 3x3'lük birim matrisi temsil etmek üzere, 3 boyuttaki ayrık zamanlı hareket modeli ise Denklem (2.19)'da verilmektedir [10].

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} I_3 & T.I_3 \\ 0 & I_3 \end{bmatrix}.\boldsymbol{x}_k + \begin{bmatrix} {}^{T^2}/_{2}.I_3 \\ T.I_3 \end{bmatrix}.\boldsymbol{w}_k = \boldsymbol{F}^{CV}.\boldsymbol{x}_k + \boldsymbol{\Gamma}^{CV}.\boldsymbol{w}_k \quad (2.19)$$

Üç boyutta sabit ivmeli model, iki boyutta bulunan hareket modelindeki I_2 birim matrisi yerine, I_3birim matrisi konularak kolayca elde edilebilir [10].

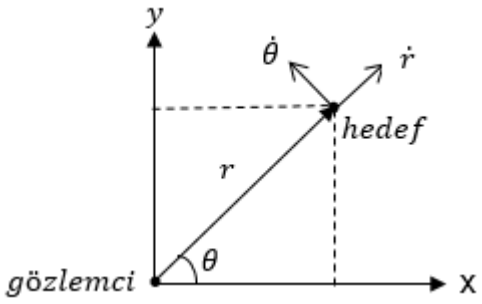
$$\boldsymbol{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}]^T \quad (2.20)$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}^{CA}.\boldsymbol{x}_k + \boldsymbol{\Gamma}^{CA}.\boldsymbol{w}_k = \begin{bmatrix} I_3 & T.I_3 & {}^{T^2}/_{2}.I_3 \\ 0 & I_3 & T.I_3 \\ 0 & 0 & I_3 \end{bmatrix}.\boldsymbol{x}_k + \begin{bmatrix} {}^{T^2}/_{2}.I_3 \\ T.I_3 \\ I_3 \end{bmatrix}.\boldsymbol{w}_k \quad (2.21)$$

2.1.3. Küresel Koordinatlarda Hareket Modeli

Küresel koordinatlardaki hareket modelleri kullanılan parametrelere bağlı olarak kartezyen koordinattaki hareket modeli ile benzerlik göstermektedir. Yine bu koordinat sisteminde de konum, hız ve (duruma göre) ivme arasında nasıl bir ilişki olduğu incelenmektedir. Yani hareket modelleri sabit hızlı ve sabit ivmeli olmak üzere iki durumda incelenecektir.

Küresel koordinatlarda iki boyutlu sistem için genel yapı Şekil 2.1'de, gözlemci-hedef ilişkisi ise Şekil 2.5'te görüldüğü gibidir.



Şekil 2.5: İki boyut için polar koordinatta gözlemci-hedef ilişkisi.

Küresel koordinatlardaki sabit hızlı model için temel parametreler aşağıda verilmiştir.

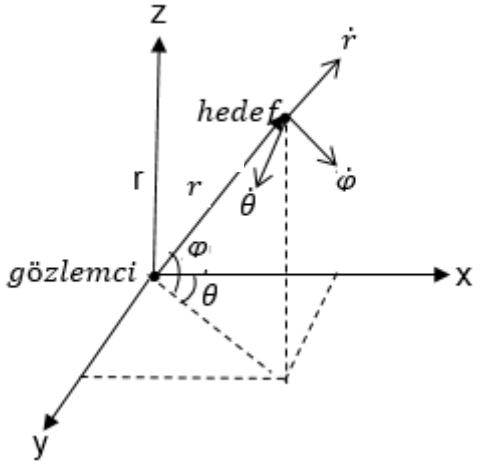
$$\text{İki boyutlu sistem için } \boldsymbol{x} = [r, \theta, \dot{r}, \dot{\theta}]^T \tag{2.22}$$

Ayrık zaman için kartezyen koordinatlarda genel hareket denklemleri önceki bölümlerde çıkarılmıştı ve o çıkarımlar kullanılarak küresel koordinatlardaki hareket modeli Denklem (2.14) ile aynı olup sadece kullanılan $\boldsymbol{x}_k$ modeli değişmektedir. Küresel koordinatlarda kullanılacak olan $\boldsymbol{x}$ yukarıda belirtildiği gibidir.

Sabit ivmeli model yine kartezyen koordinatlardaki sabit ivmeli model ile aynıdır. Fakat kullanılan x durum vektörü iki boyutta aşağıdaki gibi olmalıdır.

$$\text{İki boyutlu sistem için } \boldsymbol{x} = [r, \theta, \dot{r}, \dot{\theta}, \ddot{r}, \ddot{\theta}]^T \tag{2.23}$$

Üç-boyut durumunda ise Şekil 2.2'de gösterilen koordinat sistemi geçerli olmaktadır. Gözlemci ve hedefin koordinat sistemi üzerindeki ilişkisi ise Şekil 2.6'da belirtilmektedir.



Şekil 2.6: Üç boyut için küresel koordinatta gözlemci-hedef ilişkisi.

Küresel koordinatta üç boyutlu hareket denklemi kartezyen koordinattaki model (Denklem (2.19)) ile aynı olup kullanılmakta olan durum vektörü $\boldsymbol{x}$ ise Denklem (2.24)'te belirtilen yapıdadır.

$$\text{Üç boyutlu sistem için } \boldsymbol{x} = [r, \theta, \varphi, \dot{r}, \ddot{\theta}, \ddot{\varphi}]^T \tag{2.24}$$

Sabit ivmeli model ise Denklem (2.21)'de belirtildiği gibidir. Mevcut $\boldsymbol{x}$ ise aşağıda verildiği gibidir.

$$\text{Üç boyutlu sistem için } \boldsymbol{x} = [r, \theta, \varphi, \dot{r}, \dot{\theta}, \dot{\varphi}, \ddot{r}, \ddot{\theta}, \ddot{\varphi}]^T \tag{2.25}$$

2.2. Hedef Takip Algoritmaları

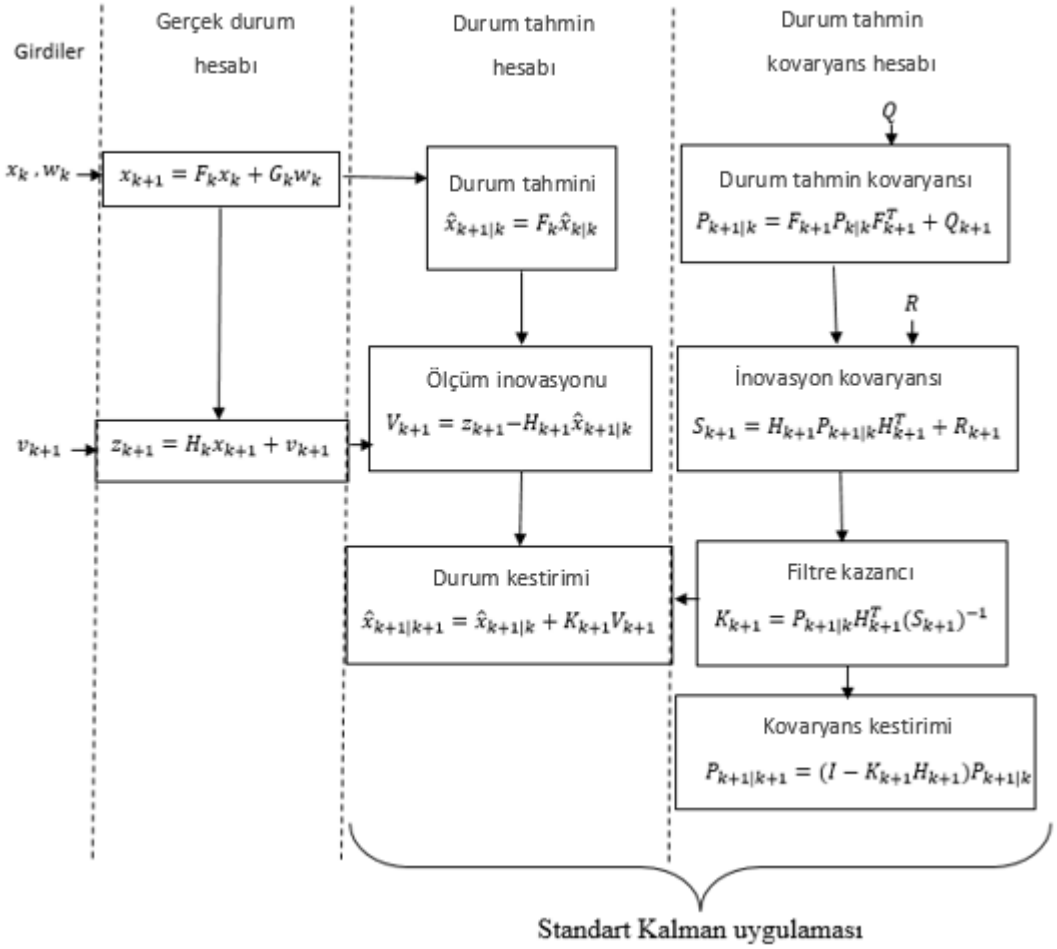
Hedef takip algoritmaları kullanılan hedef hareket modeline ve ölçüm modeline göre farklılık göstermektedir. Bu farklılıklar için $\alpha - \beta - \gamma$ filtresi, Kalman filtresi, Genişletilmiş Kalman Filtresi (EKF, *ing.* Extended Kalman Filter), UKF, parçacık filtresi gibi farklı algoritmalar kullanılmaktadır. Hareket ve ölçüm modelinin doğrusal olması durumunda $\alpha - \beta - \gamma$ filtresi ya da Kalman filtresi kullanılırken, ölçüm modelinin doğrusal olmadığı durumlarda EKF veya UKF filtreleri kullanılmaktadır.

Kalman filtresinin birçok kullanım alanı vardır ve bu alanlarda özellikle havacılık alanında hedef navigasyonunda sistemin durumu hakkında tahmin sağlanması için kullanılmıştır. Algoritmada gürültü girdileri mevcutken sistem durumu için optimize edilmiş tahminleri sağlamaktadır. Algoritma bu tahminleri sağlarken özyinelemeli (*ing.* recursive) şekilde çalışmakta ve hataları en küçük kareler yöntemine göre minimize etmektedir.

Kalman filtresinin, farklı sistem modeline göre farklı çeşitleri mevcuttur. Gözlemlenen değişkenlerle durum parametreleri arasındaki ilişki doğrusal ise $\alpha - \beta - \gamma$ filtresi veya standart Kalman filtresi kullanılırken, ilişkilerin doğrusal olmaması durumunda EKF, UKF, CMKF veya parçacık filtresi kullanılmaktadır. Bu bölümde bu takip algoritmaları hakkında kısaca bilgi verilecektir.

2.2.1. Kalman Filtresi

Kalman filtresi, durum uzay modeliyle gösterilen bir sistemde, modelin önceki bilgileriyle buna ek olarak sistemin giriş ve çıkış bilgilerinden yararlanarak ilgili durumun tahmin edilmesini sağlayan bir filtredir. Kalman filtre blok şeması Şekil 2.7'de verilmiştir [11],[12].



Şekil 2.7: Kalman Filtresi Akış Diyagramı.

Kalman Filtresi için model bakımından $\boldsymbol{x}$ durum vektörü ve $\boldsymbol{z}$ gözlem vektörü olmak üzere temelde iki denklem mevcuttur ve bu denklemler Denklem (2.26) ve Denklem (2.27)'de belirtildiği gibidir [13] [14];

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{w}_k \tag{2.26}$$

$$\boldsymbol{z}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k \tag{2.27}$$

Denklem (2.26) ve Denklem (2.27)'de belirtilen k alt indisi, ilgili k zamanını belirtmektedir. Örneğin ölçüm aralığı T ms olarak alındığı düşünüldüğünde $k = T\ ms$ iken $k + 1 = 2T\ ms$ durumunu belirtmektedir. Ayrıca $\boldsymbol{F}$ dönüşüm matrisi, $\boldsymbol{H}$ gözlem matrisi, $\boldsymbol{w}$ model (süreç) gürültüsü ve $\boldsymbol{v}$ ölçüm gürültüsü olarak adlandırılmaktadır. $\boldsymbol{F}$ ve $\boldsymbol{H}$ matrisleri kullanılan modele göre değişirken; $\boldsymbol{w}$ ve $\boldsymbol{v}$ gürültüleri ise genelde ortalaması sıfır standart sapmaları σ_w ve σ_v olan Gauss gürültüleri olarak alınır

[13],[15]. Fakat bilindiği üzere pratikte karşılaşılan gürültülerin tamamı Gauss gürültüsü ile modellenemez fakat çoğu kez Gauss gürültüsü olarak ele alınmaktadır. Ayrıca gürültü parametreleri olan ortalama ve standart sapma (μ, σ) doğru tahmin edilmese bile Kalman filtresi doğru tahmin değerlerine yakınsayabilmektedir. Fakat başlangıç parametreleri ne kadar iyi tahmin edilirse, Kalman filtresi o kadar iyi tahmin sonuçları üretir ve gerçek değere o kadar yaklaşır.

Sabit hızlı doğrusal yörüngede ilerleyen ve üç boyutlu kartezyen koordinattaki modele sahip olan hedef için örnek $\boldsymbol{x}$ durum vektörü ve Denklem (2.26)'da belirtilen dinamik model aşağıdaki gibi gösterilmektedir.

$$\boldsymbol{x}^T = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}] \tag{2.28}$$

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{I}_3 & (T)\boldsymbol{I}_3 \\ 0 & \boldsymbol{I}_3 \end{bmatrix} \tag{2.29}$$

Sabit ivme ile doğrusal yörüngede ilerleyen üç boyutlu kartezyen koordinattaki modele sahip olan hedef için ise durum şöyledir [13].

$$\boldsymbol{x}^T = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}\ \ddot{x}\ \ddot{y}\ \ddot{z}] \tag{2.30}$$

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{I}_3 & (T)\boldsymbol{I}_3 & (T^2/2)\boldsymbol{I}_3 \\ 0 & \boldsymbol{I}_3 & (T)\boldsymbol{I}_3 \\ 0 & 0 & \boldsymbol{I}_3 \end{bmatrix} \tag{2.31}$$

Denklemlerde kullanılan $\boldsymbol{I}$ birim vektörü, sistem üç boyutlu olduğu için 3x3'lük matristir ve bu denklemler için Denklem (2.27)'de belirtilen ölçüm modeli ile $\boldsymbol{z}$ gözlem vektörü ise Denklem (2.32)-(2.34)'daki şekilde tanımlanmaktadır.

$$\boldsymbol{z} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.32}$$

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{2.33}$$

$$\boldsymbol{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{2.34}$$

Kalman filtresi için iki denklem takımı vardır. Bunlar tahmin (*ing.* predict) ve güncelleme (ing. update) denklem takımlarıdır [15].

Tahmin denklem takımı Denklem (2.35) ve Denklem (2.36)'da verilmektedir. Bu iki denklem ölçümler alınmadan önce bir sonraki durumu tahmin etmede kullanılmaktadır.

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F}_k \hat{\boldsymbol{x}}_{k-1|k-1} \tag{2.35}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{F}_k^T + \boldsymbol{Q}_k \tag{2.36}$$

Belirtilen bu iki denklemdeki $\boldsymbol{F}$ bir önceki bölümde belirtilenler ile aynı iken; $\hat{\boldsymbol{x}}$ Kalman tahminini, $\boldsymbol{P}$ Kalman kestirimin hassasiyetiyle ilgili bilgiyi içeren durum kestirim kovaryans matrisini ve $\boldsymbol{Q}$ ise model gürültüsünün kovaryans matrisini belirtmektedir.

İlk durumda $\boldsymbol{P}$ bilinmediği için $\boldsymbol{P}$'nin ilk değeri $\boldsymbol{Q}$ model gürültü kovaryansı ile başlatılabilir. $\hat{\boldsymbol{x}}$'nın ilk değeri ise alınan ilk ölçüm değeri olarak alınır. Diğer değerler ise verilen formüllere göre iterasyonla bulunur.

Güncelleme denklem takımı Denklem (2.37)-(2.40)'da belirtilmektedir. Bu denklem takımı ölçümler alındıktan sonra mevcut durumu güncellemek için uygulanır.

$$\boldsymbol{S}_k = \boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T + \boldsymbol{R}_k \tag{2.37}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T (\boldsymbol{S}_k)^{-1} \tag{2.38}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k (\boldsymbol{z}_k - \boldsymbol{H}_k \hat{\boldsymbol{x}}_{k|k-1}) \tag{2.39}$$

$$\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_{k|k-1} \tag{2.40}$$

Denklemlerdeki $\boldsymbol{P}, \boldsymbol{H}, \boldsymbol{z}$ ve $\widehat{\boldsymbol{x}}$ önceki belirtilen notasyonlarla aynı iken; $\boldsymbol{R}$ matrisi ölçüm gürültüsünün kovaryans matrisini, $\boldsymbol{I}$ birim matrisi, $\boldsymbol{K}$ Kalman kazanç matrisini ve **S** ise inovasyon kovaryansını temsil etmektedir. Denklem (2.39)'da görülen $(\boldsymbol{z}_k - \boldsymbol{H}_k \widehat{\boldsymbol{x}}_{k|k-1})$ fark terimi inovasyon (yenilenme) hesabı olarak bilinmektedir ve $\boldsymbol{V}_k$ ile temsil edilmektedir [16]. Formüllerde bilinmesi gereken $\boldsymbol{R}$ ve $\boldsymbol{Q}$ matrisleri aşağıdaki şekildedir [16].

$$\boldsymbol{Q} = \boldsymbol{\Gamma} \sigma_w^2 \boldsymbol{\Gamma}^T \tag{2.41}$$

$$\boldsymbol{R} = \sigma_v^2 \tag{2.42}$$

Denklem (2.41)'de yer alan $\boldsymbol{\Gamma}$ üç boyutlu sabit hız modeli için $\boldsymbol{\Gamma} = \begin{bmatrix} (T^2/2)\boldsymbol{I_3} \\ (T)\boldsymbol{I_3} \end{bmatrix}$ formundadır [11].

Burada $\boldsymbol{R}$ matrisi ölçüm gürültüsünü modellediği için belirlenmesi daha kolay iken, $\boldsymbol{Q}$ matrisi model gürültüsünün kovaryansını temsil ettiği için hedefin o anki hareketine bağlıdır ve bilinmesi daha zordur.

$\boldsymbol{R}$ matrisi tek boyut için Denklem (2.42)'de belirtildiği gibi iken çok boyutluda boyuta göre matris boyutları değişmektedir. Örneğin iki boyutlu kartezyen koordinat için Denklem (2.43)'te belirtilen şekilde olur [13], [17].

$$\boldsymbol{R} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 \\ \sigma_{xy}^2 & \sigma_y^2 \end{bmatrix} \tag{2.43}$$

Eğer koordinatlar kartezyen koordinatta olduğu gibi birbirinden bağımsız ise üç boyutlu $\boldsymbol{R}$ matrisi Denklem (2.44)'teki şekli almaktadır.

$$\boldsymbol{R} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} \tag{2.44}$$

Bunun yanında kartezyen koordinattaki gürültü varyansları her koordinat bileşeni için aynı ve σ_v^2 ise $\boldsymbol{R}$ matrisi Denklem (2.45)'te belirtilen şekle dönüşmektedir.

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \sigma_v^2 \tag{2.45}$$

2.2.2. Genişletilmiş Kalman Filtresi

Radar sistemlerinde genelde ölçümler polar/küresel koordinatlarda alınmaktadır. Fakat Bölüm 2.2.1' de görüldüğü üzere kullanılan modeller kartezyen koordinattadır. Kalman filtresi için Denklem (2.26) ve (2.27)'de verilen model ve ölçüm dinamiği EKF filtresi için Denklem (2.46) ve (2.47)'de verilmiştir.

$$\boldsymbol{x}_{k+1} = f_k(\boldsymbol{x}_k) + \boldsymbol{w}_k \tag{2.46}$$

$$\boldsymbol{z}_k = h_k(\boldsymbol{x}_k) + \boldsymbol{v}_k \tag{2.47}$$

Hedef takibinde küresel koordinatlar cinsinden ölçüm alınması durumunda Denklem (2.46) ve (2.47) doğrusal olmayan bir yapıya sahip olacaktır. EKF bu gibi durumlardan kaynaklı doğrusal olmayan sistemler için uygun bir filtredir. Bu filtrede durum tahmini için mevcut durumun ortalaması ve kovaryansı civarında bir doğrusallaştırmaya dayanmaktadır. Böylece doğrusal olmayan sistemlerde doğrusal yapıya sahipmiş gibi kullanılabilmektedir. Fakat EKF filtresinin doğrusal olmayan sistemlerde kullanılabilme avantajına sahip olmasına karşın bazı dezavantajları mevcuttur. Bunların başında EKF'de aynı Kalman filtresinde olduğu gibi ölçüm ve süreç gürültüsünün ortalaması sıfır olan Gauss dağılımında olduğu varsayılmaktadır. Diğer bir sorun ise alınan ölçümler arasındaki sürenin fazla olması durumunda doğrusallaştırma işlemi kararsızlığa yol açabilmektedir [18]. Ayrıca başlangıç durumunda tahmindeki büyük hata durumunda iz kaybolabilmektedir [19]. Bununla birlikte doğrusallaştırma için kullanılan Jakobian matrislerinin çıkarımı da dezavantajına örnek olarak verilebilir.

Üç boyutlu sistemde yaklaşık sabit hız modeli için durum vektörü $\boldsymbol{x}_k$ Denklem (2.28)'de, ölçüm vektörü $\boldsymbol{z}_k$ ise aşağıda verilmektedir.

$$\boldsymbol{z}_k = \begin{bmatrix} r \\ \theta \\ \varphi \end{bmatrix} + \boldsymbol{v}_k \tag{2.48}$$

EKF algoritması için gerekli hesaplamalar Denklem (2.49)-(2.57)'de verilmektedir [20],[18].

$$\widehat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} = f_{k-1}\left(\widehat{\boldsymbol{x}}_{\boldsymbol{k-1|k-1}}\right) \tag{2.49}$$

$$\boldsymbol{P}_{\boldsymbol{k|k-1}} = \boldsymbol{Q}_{\boldsymbol{k-1}} + \widehat{\boldsymbol{F}}_{\boldsymbol{k-1}}\boldsymbol{P}_{\boldsymbol{k-1|k-1}}\widehat{\boldsymbol{F}}_{\boldsymbol{k-1}}^{\boldsymbol{T}} \tag{2.50}$$

$$\widehat{\boldsymbol{x}}_{\boldsymbol{k|k}} \;\; = \widehat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} + \boldsymbol{K}_{\boldsymbol{k}}\left(\boldsymbol{z}_{\boldsymbol{k}} - h_k\left(\widehat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}\right)\right) \tag{2.51}$$

$$\boldsymbol{P}_{\boldsymbol{k|k}} = (\boldsymbol{I} - \boldsymbol{K}_{\boldsymbol{k}}\boldsymbol{H}_{\boldsymbol{k}})\boldsymbol{P}_{\boldsymbol{k|k-1}} \tag{2.52}$$

$$\boldsymbol{S}_{\boldsymbol{k}} \;\; = \widehat{\boldsymbol{H}}_{\boldsymbol{k}}\boldsymbol{P}_{\boldsymbol{k|k-1}}\widehat{\boldsymbol{H}}_{\boldsymbol{k}}^{\boldsymbol{T}} + \boldsymbol{R}_{\boldsymbol{k}} \tag{2.53}$$

$$\boldsymbol{K}_{\boldsymbol{k}} \;\; = \boldsymbol{P}_{\boldsymbol{k|k-1}}\widehat{\boldsymbol{H}}_{\boldsymbol{k}}^{\boldsymbol{T}}\boldsymbol{S}_{\boldsymbol{k}}^{\boldsymbol{-1}} \tag{2.54}$$

$\widehat{\boldsymbol{F}}_{\boldsymbol{k-1}}$ ve $\widehat{\boldsymbol{H}}_{\boldsymbol{k}}$ Jakobian matrisleri olup aşağıdaki formüller ile hesap edilir.

$$\widehat{\boldsymbol{F}}_{\boldsymbol{k-1}} = \left[\nabla_{x_{k-1}} f_{k-1}^{T}(\boldsymbol{x}_{\boldsymbol{k-1}})^{T}\right]^{T}|_{\boldsymbol{x_{k-1}=\widehat{x}_{k-1|k-1}}} \tag{2.55}$$

$$\widehat{\boldsymbol{H}}_{\boldsymbol{k}} = \left[\nabla_{x_k} h_k^{T}(\boldsymbol{x}_{\boldsymbol{k}})^{T}\right]^{T}|_{\boldsymbol{x_k=\widehat{x}_{k|k-1}}} \tag{2.56}$$

$$\nabla_{x_k} = \left[\frac{\partial}{\partial \boldsymbol{x}_{\boldsymbol{k}}[1]} \dots \dots \frac{\partial}{\partial \boldsymbol{x}_{\boldsymbol{k}}[n_k]}\right]^{T} \tag{2.57}$$

2.2.3. Kokusuz Kalman Filtresi

Kokusuz Kalman filtresi kokusuz dönüşüme (*ing.* Unscented Transform) dayanmaktadır. Bu dönüşümde doğrusal olmayan fonksiyon EKF'deki gibi lineerleştirmek yerine bir noktalar kümesi oluşturulur ve bu noktalar fonksiyonda kullanılır. Bu yöntemdeki temel bakış açısı fonksiyonun kendisi yerine olasılık dağılımına yakınsamasıdır. Böylece hesaplama yükü fazla olmadan doğrudan hızlı bir şekilde sonuca ulaşılabilmektedir. Bu filtrede nokta kümesini oluşturmak için sigma noktaları kullanılmaktadır ve bu noktalar kullanılarak durum tahmini yapılmaktadır.

Filtre için kullanılan durum vektörü $\boldsymbol{x}_k$ ve ölçüm vektörü $\boldsymbol{z}_k$ modeli sırasıyla Denklem (2.46)'da ve Denklem (2.47)'de verildiği gibidir. Kokusuz dönüşüm için kullanılan $2n$+1 sigma σ noktaları ve ağırlıklarının hesabı için kullanılan denklemler aşağıda sıralanmaktadır. Denklemlerdeki $\bar{x}$, x'in ortalamasını, $\boldsymbol{P}_X$ kovaryans matrisini, n vektör boyutunu, W_i^m tahmin edilen örnek noktaların ortalama ağırlıklarını, W_i^c kovaryans ağırlıklarını, λ ise merkezden uzaklaşmayı belirleyen skala faktörünü temsil etmektedir. α sigma noktalarının $\bar{x}$ etrafındaki yayılımını tanımlar ve genelde $0 < \alpha < 1$ olan çok küçük değerli bir sayıdır. β x'in dağılımının önceki bilgilerini birleştirmede kullanılır ve Gauss dağılımı durumunda 2 değeri kullanılmaktadır [21], [22]. Denklem (2.64)'teki k değeri ise genelde 0 olarak alınmaktadır [21].

$$\boldsymbol{x_0} = \bar{x} \tag{2.58}$$

$$\boldsymbol{x_i} = \bar{x} + \left(\sqrt{(n+k)\boldsymbol{P}_X}\right)_i \quad i = 1, \dots n \tag{2.59}$$

$$\boldsymbol{x_i} = \bar{x} - \left(\sqrt{(n+k)\boldsymbol{P}_X}\right)_i \quad i = n+1, \dots 2n \tag{2.60}$$

$$W_0^m = \frac{\lambda}{n+\lambda} \tag{2.61}$$

$$W_0^c = \frac{\lambda}{n+\lambda} + 1 - \alpha^2 + \beta \tag{2.62}$$

$$W_i^c = W_i^m = \frac{1}{2\big((n+\lambda)\big)}\, i = 1,2,\dots 2n \tag{2.63}$$

$$\lambda = \alpha^2(n+k) - n \tag{2.64}$$

$\boldsymbol{x_i}$ sigma vektörleri $i = 0,\dots,2n$ için $y_i = g(\boldsymbol{x_i})$ kullanılarak lineer olmayan fonksiyon üzerinden geçirilir ve elde edilen y'nin ortalaması ve kovaryansı sırasıyla Denklem (2.65) ve Denklem (2.66) kullanılarak hesaplanır [20], [21].

$$\bar{y} = \sum_{i=0}^{2n} W_i^{(m)} y_i \tag{2.65}$$

$$\boldsymbol{P}_y = \sum_{i=0}^{2n} W_i^{(c)} (y_i - \bar{y})(y_i - \bar{y})^T \tag{2.66}$$

UKF filtresi için zaman güncelleme/durum tahmini denklemleri Denklem (2.67)-(2.71)'de, ölçüm güncelleme/düzeltme denklemleri ise Denklem (2.72)-(2.76)'da verilmektedir [20]-[24].

$$\boldsymbol{x_{k|k-1}} = \boldsymbol{F}[\boldsymbol{x_{k-1}^x}, \boldsymbol{x_{k-1}^v}] \tag{2.67}$$

$$\widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{-} = \sum_{i=0}^{2n} W_i^{(m)} x_{i,k|k-1} \tag{2.68}$$

$$\boldsymbol{P_k^-} = \sum_{i=0}^{2n} W_i^{(c)} \left[x_{i,k|k-1} - \widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}\right]\left[x_{i,k|k-1} - \widehat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}\right]^T \tag{2.69}$$

$$\boldsymbol{y_{k|k-1}} = \boldsymbol{H}[\boldsymbol{x_{k|k-1}}, \boldsymbol{x_{k-1}^n}] \tag{2.70}$$

$$\widehat{\boldsymbol{y}}_{\boldsymbol{k}}^{-} = \sum_{i=0}^{2n} W_i^{(m)} y_{i,k|k-1} \tag{2.71}$$

$$\boldsymbol{P}_{\hat{\boldsymbol{y}}_{\boldsymbol{k}}\hat{\boldsymbol{y}}_{\boldsymbol{k}}} = \sum_{i=0}^{2n} W_i^{(c)} \left[y_{i,k|k-1} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{-}\right]\left[y_{i,k|k-1} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{-}\right]^T \tag{2.72}$$

$$\boldsymbol{P}_{\hat{\boldsymbol{x}}_{\boldsymbol{k}}\hat{\boldsymbol{y}}_{\boldsymbol{k}}} = \sum_{i=0}^{2n} W_i^{(c)} \left[x_{i,k|k-1} - \hat{x}_k^{-}\right]\left[y_{i,k|k-1} - \hat{y}_k^{-}\right]^T \tag{2.73}$$

$$\mathrm{K} = \boldsymbol{P}_{\hat{\boldsymbol{x}}_{\boldsymbol{k}}\hat{\boldsymbol{y}}_{\boldsymbol{k}}} \boldsymbol{P}_{\hat{\boldsymbol{y}}_{\boldsymbol{k}}\hat{\boldsymbol{y}}_{\boldsymbol{k}}}^{-1} \tag{2.74}$$

$$\hat{x}_k = \hat{x}_k^{-} + K(y_k - y_k^{-}) \tag{2.75}$$

$$\boldsymbol{P}_{\boldsymbol{k}} = P_k^{-} - K\boldsymbol{P}_{\hat{\boldsymbol{y}}_{\boldsymbol{k}}\hat{\boldsymbol{y}}_{\boldsymbol{k}}} K^T \tag{2.76}$$

2.2.4. Dönüştürülmüş Ölçümler Kalman Filtresi

İzleme filtrelerin seçiminde genelde küresel (veya polar) ve kartezyen koordinat olmak üzere iki koordinat sistemi kullanılmaktadır. Kutupsal koordinatlar kullanıldığında ölçümler lineer olurken durum denklemleri lineer olmamaktadır. Lineer hedef modeli için ise kartezyen koordinatlardaki durum denklemleri lineer olduğu için avantajlıdır fakat bu durumun dezavantajı ise alınan ölçüler lineer değildir. Ölçümlerdeki lineer olmama durumu EKF, UKF ve ya filtrelemeden önce ölçümlerin kartezyen koordinata dönüştürülmesiyle ele alınabilir. Bu nedenle ikinci yaklaşım olan kartezyen koordinata dönüşüm yansız yapılabilirse iyi sonuçlar elde edilebilir [25].

CMKF algoritması küresel koordinatlardan (r, θ, φ) kartezyen koordinatlara (x, y, z) geçiş yaparken oluşabilecek hataları azaltmak için kullanılan bir yöntemdir. Bu yöntemde direk koordinat dönüşümlerinden elde edilen değerler direkt kullanılmayıp bir fonksiyon yardımıyla gerçeğe daha yakın olacak şekilde düzeltme yapılmaktadır. Bunun için küresel koordinatlarda alınan radar ölçümleri, sapma ortalamasının sıfır yapılabilmesi veya sıfıra yakınlaştırılabilmesi için direk olarak kartezyene dönüştürülmemektedir. Bunun için farklı dönüşümler mevcuttur ve bu dönüşüm çeşitlerine Bölüm 3'te değinilecektir. Bunun yanında ölçümlerin küresel

koordinattan kartezyen koordinata dönüştürülmesi nedeniyle küresel koordinatlarda kovaryans matrisi $\boldsymbol{R}$ olan Gauss ölçüm gürültüsü kartezyen koordinatta artık $\boldsymbol{R}$ kovaryansına sahip olmamaktadır. Koordinat dönüşümlerinden kaynaklı değişen kovaryans matrisi Bölüm 3'te belirtilen kovaryans matrisleri kullanılarak yansız hale getirilmeye çalışılmaktadır.

Bu algoritma, Bölüm 3'te daha detaylı olarak bahsedilmektedir.

2.3. Sensör Sapması

Çoklu sensör çoklu hedef takibi, modern Hava Trafik Kontrol (ATC, *ing*. Air Traffic Control) sistemlerinin temelini oluşturmaktadır. Mevcut tüm sensör ölçümlerini kullanan oldukça verimli (IMM gibi) hedef takip algoritmaları geliştirilmiştir. Fakat bu filtrelerde ölçüm ile model arasında bir uyuşmazlık olabilmektedir ve bu tespit performansını ve dolayısıyla hedef takip performansını olumsuz etkilemektedir [26]. Bu nedenle, bu gibi uygulamalarda sensörlerin sapma kestirimine ve bunun düzeltilmesine odaklanılmıştır. Günümüzde çoğu takip sisteminde (yüksek doğruluk, geniş kapsamlı, düzeltme gibi) farklı avantajları olan farklı sensörler kullanılmaktadır ve füzyon aşamasında bu durum yeni zorluklar getirmektedir. Bu nedenle, özellikle sapma kestirimi üzerinde yoğunlaşılmaktadır. [26].

Sapma, çeşitli alt tiplere ayrılmaktadır. Bunlar [26];

- Sensör ilişkili sapma: Hedeften bağımsız olarak aynı değere sahip sensör ilişkili sapmalardır.
- Hedef ilişkili sapma: Sensörden bağımsızdır ve aynı tip sensörler için eşit değerlidir.
- Sensör-Hedef çifti ilişkili sapma: Her sensör-hedef çifti için farklıdır.

Bu tez kapsamında sadece sensör ilişkili sapma durumu ele alınmaktadır.

Hedef takip sistemlerinde birden fazla sensör kullanılabilmektedir. Bu durumda, hedef konumunun algılanması her sensörde aynı zamanda olamayabilir ve zaman farkları oluşabilmektedir. Dolayısıyla, ölçüm zaman farklarından kaynaklı olarak hedef konumu her sensör için farklı olabilmektedir. Sensörlerden elde edilen hedef konumları daha sonra kaynaştırılarak hedefe ait nihai konum belirlendiği için

sensörlerin hedefi algılama zamanlarından kaynaklı sensör sapması takip algoritmalarını olumsuz etkilemektedir. Sensörler, senkronize ve senkronize olmayan olmak üzere iki grupta incelenebilir. Senkronize sensörlerde, hedef konumu her sensörde aynı anda algılandığı varsayılmaktadır. Senkronize olmayan sensörlerde ise hedef konumunun algılanması aynı anda olmamakta, zaman kaymaları oluşabilmektedir. Senkronize olan sensörler için sapma tahminine yönelik öne sürülen yöntemler vardır [27]-[31]. [31]'de senkron sensörler için sapma modeli bilinmeyen bir sabit ise özyinelemeli en küçük kareler (RLS, *ing.* Recursive Least Squares) kestirimcisi [31], maksimum olasılık kestirimcisi (MLE, ing. Maximum Likelihood Estimator) veya ağırlıklı en küçük kareler (LS, Least Squares) [32]; Gauss varsayımı altında rastgele dizi ise de optimal en küçük karesel hata (Optimal MMSE, ing. Optimal Minimum Mean Square Error) kestirimcisi [31] yöntemleri kullanılmaktadır. Senkronize olmayan sensörler konusuna ise [32]'de değinilmektedir.

Gözlem bölgesindeki *N* tane hedefin takip edildiği, ölçümlerin polar koordinatta alındığı *M* adet sabit sensörlü sistem için sapmalı ölçüm modeli Denklem **(2.77)**'de verilmektedir [30]-[32]. Denklemdeki $\boldsymbol{z}_i^p(k)$, *k* anında, *i* sensörüne ait polar koordinattaki ölçümü temsil etmektedir. Aynı şekilde $r_i(k)$ ve $\theta_i(k)$ ise *k* anında, *i* sensörüne ait sırasıyla menzil ve yanal açıyı temsil etmektedir. Denklem **(2.77)**'de yer alan $r_i(k)$ ve $\theta_i(k)$ terimleri sırasıyla Denklem (2.78) ve Denklem (2.79)'da verilmiştir. Denklemlerdeki $r_i^t(k)$ ve $\theta_i^t(k)$ sırasıyla gerçek menzil ve yanal açıyı, $v_i^r(k)$ ve $v_i^\theta(k)$ sırasıyla menzil ve yanal açı için sıfır ortalamalı, ${\sigma_r}^2$ve σ_θ^2 varyanslı Gauss ölçüm gürültüsünü; $b_i^r(k)$, $\epsilon_i^r(k)$, $b_i^\theta(k)$ ve $\epsilon_i^\theta(k)$ ise sensör sapma vektörünün bileşenlerini temsil etmektedir. Sensör sapma vektörü olan $\boldsymbol{\beta}_i(k)$, Denklem (2.80)'de belirtildiği gibidir.

$$\boldsymbol{z}_i^p(k) = \begin{bmatrix} r_i(k) \\ \theta_i(k) \end{bmatrix} \tag{2.77}$$

$$r_i(k) = [1 + \epsilon_i^r(k)]r_i^t(k) + b_i^r(k) + v_i^r(k) \tag{2.78}$$

$$\theta_i(k) = \left[1 + \epsilon_i^\theta(k)\right]\theta_i^t(k) + b_i^\theta(k) + v_i^\theta(k) \tag{2.79}$$

$$\boldsymbol{\beta}_i(k) \triangleq \begin{bmatrix} b_i^r(k) \\ b_i^\theta(k) \\ \epsilon_i^r(k) \\ \epsilon_i^\theta(k) \end{bmatrix} \tag{2.80}$$

Denklem (2.77)'de verilmekte olan polar koordinattaki ölçüm düzenlendiğinde aşağıda belirtilen form elde edilmektedir ve denklemde kullanılan $\boldsymbol{C}_i(k)$ Denklem (2.82)'de verilmiştir.

$$\boldsymbol{z}_i^p(k) = \begin{bmatrix} r_i^t(k) \\ \theta_i^t(k) \end{bmatrix} + \boldsymbol{C}_i(k)\boldsymbol{\beta}_i(k) + \begin{bmatrix} v_i^r(k) \\ v_i^\theta(k) \end{bmatrix} \tag{2.81}$$

$$\boldsymbol{C}_i(k) \triangleq \begin{bmatrix} 1 & 0 & r_i^t(k) & 0 \\ 0 & 1 & 0 & \theta_i^t(k) \end{bmatrix} \tag{2.82}$$

Ölçümlerin küresel koordinatta alındığı üç boyutlu sistem için $\boldsymbol{z}_i^s(k)$ ise, polar koordinat için elde edilen ve Denklem (2.81)'de belirtilen formun küresel koordinata genişletilmiş halidir ve Denklem (2.83)'te verilmektedir. Denklemlerde kullanılan φ bileşenleri, θ yanay açısı için olan bileşenlerin düşey açı içinde genişletilmiş halidir.

$$\boldsymbol{z}_i^s(k) = \begin{bmatrix} r_i^t(k) \\ \theta_i^t(k) \\ \varphi_i^t(k) \end{bmatrix} + \boldsymbol{C}_i(k)\boldsymbol{\beta}_i(k) + \begin{bmatrix} v_i^r(k) \\ v_i^\theta(k) \\ v_i^\varphi(k) \end{bmatrix} \tag{2.83}$$

$$\boldsymbol{C}_i(k) \triangleq \begin{bmatrix} 1 & 0 & 0 & r_i^t(k) & 0 & 0 \\ 0 & 1 & 0 & 0 & \theta_i^t(k) & 0 \\ 0 & 0 & 1 & 0 & 0 & \varphi_i^t(k) \end{bmatrix} \tag{2.84}$$

Polar koordinatlarda alınan ölçüm için denklemlerde kullanılan Denklem (2.82)'deki $\boldsymbol{C}_i(k)$ hesaplamasında [32] ve [30]'da $r_i^t(k)$ ve $\theta_i^t(k)$ için sırasıyla $r_i^m(k)$ ve $\theta_i^m(k)$ ölçüm değerleri; [31]'de ise gözlem değerleri olan $\hat{r}_i(k)$ ve $\hat{\theta}_i(k)$ kullanılabileceği önerilmiştir. Benzer şekilde Denklem (2.84) içinde uygulamaya göre $r_i^t(k)$,$\theta_i^t(k)$ ve $\varphi_i^t(k)$ veya $\hat{r}_i(k)$, $\hat{\theta}_i(k)$ ve $\hat{\varphi}_i(k)$ kullanılabilir.

Polar koordinatlarda alınan ölçüm modeli için kullanılan $\boldsymbol{z}_i^p(k)$, kartezyen koordinatlarda iki boyutlu olarak Denklem (2.85)'teki formda olmaktadır. Denklemlerde kullanılan matris ve vektörler denklemin devamında verilmiştir. $\boldsymbol{x}(k)$ hedef durum vektörünü, $\boldsymbol{H}_i(k)$ ise ölçüm matrisini temsil etmektedir. $\boldsymbol{B}_i(k)$ dönüşüm matrisi, gerçek menzil ve yanal açı değerlerinin doğrusal olmayan bir fonksiyonudur. Eğer $\boldsymbol{B}_i(k)\boldsymbol{C}_i(k)$ sabit ise, sadece farklı sensörlerdeki sapmanın farkı gözlemlenebilir [30]-[32]. $\boldsymbol{B}_i(k)$ matrisi hesabında, [31]'de Denklem (2.88)'de görülmekte olan menzil ve yanal açı tahminleri yerine, [30] ve [32]'de menzil ve yanal açının ölçüm değerleri kullanılmaktadır. $\boldsymbol{R}_i(k)$ matrisi ise $v_i(k)$ ölçüm gürültüsünün kovaryans matrisini temsil etmektedir ve pratikte r_i ve θ_i için ölçüm değerleri olan r_i^m ve θ_i^m kullanılmaktadır [30]-[32].

$$\boldsymbol{z}_i(k) = \boldsymbol{H}_i(k)\boldsymbol{x}(k) + \boldsymbol{B}_i(k)\boldsymbol{C}_i(k)\boldsymbol{\beta}_i(k) + v_i(k) \tag{2.85}$$

$$\boldsymbol{x}(k) = [x(k)\ y(k)\ \dot{x}(k)\ \dot{y}(k)]^T \tag{2.86}$$

$$\boldsymbol{H}_i(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.87}$$

$$\boldsymbol{B}_i(k) = \begin{bmatrix} \cos(\hat{\theta}_i(k)) & -\hat{r}_i(k)\sin(\hat{\theta}_i(k)) \\ \sin(\hat{\theta}_i(k)) & \hat{r}_i(k)\cos\hat{\theta}_i(k) \end{bmatrix} \tag{2.88}$$

$$\boldsymbol{R}_i(k) = \begin{bmatrix} r_i^2\sigma_\theta^2\sin^2\theta_i + \sigma_\theta^2\cos^2\theta_i & (\sigma_r^2 - r_i^2\sigma_\theta^2)\sin\theta_i\cos\theta_i \\ (\sigma_r^2 - r_i^2\sigma_\theta^2)\sin\theta_i\cos\theta_i & r_i^2\sigma_\theta^2\cos^2\theta_i + \sigma_\theta^2\sin^2\theta_i \end{bmatrix} \tag{2.89}$$

2.4. Ölçümlerin Alındığı Koordinata Göre Sapma Oluşumu

Hedef dinamiği kartezyen koordinatlı olan hedef için, ölçüm modelinin kartezyen koordinatlı olması durumuna Denklem (2.32)–(2.34)'te değinilmiştir. Ölçüm modeli kartezyen koordinatta lineer denklemlere sahip olabileceği gibi lineer olmayan denklemlere de sahip olabilir. Bu durum radar sistemlerinin ölçümleri küresel koordinatta alması durumunda gözlenmektedir. Radar sistemleri genellikle küresel

koordinatta ölçüm almaktadır. Dolayısıyla alınan bu ölçümler kullanılırken kartezyene dönüştürülmektedir. Ölçümlerin küresel koordinatta alınması ölçüm gürültüsünün de küresel koordinatta eklenmesi anlamına gelmektedir. Küresel koordinatta eklenen Gauss gürültüleri kartezyen koordinata dönüştürüldüğünde küreselden kartezyene/ kartezyenden küresele dönüşüm denklemlerinden kaynaklı sapma etkisi görülmektedir. Bu bölümde ölçümlerin küresel koordinatta alınması halinde ölçüm denkleminin yapısı ve dönüşümün yapılması sonucu elde edilen denklemlerin lineer olmamasından kaynaklı oluşan sapma etkisi incelenecektir. Radar ölçümlerinin küresel koordinatta alınması durumunda ölçüm modelindeki $\boldsymbol{z}$ ve ölçüm gürültüsü olan $\boldsymbol{v}$, Denklem (2.90)-(2.92)'de görüldüğü üzere küresel koordinattadır.

$$\boldsymbol{z}_{\text{küresel}} = \begin{bmatrix} r \\ \theta \\ \varphi \end{bmatrix} \tag{2.90}$$

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{2.91}$$

$$\boldsymbol{v} = \begin{bmatrix} v_r \\ v_\theta \\ v_\varphi \end{bmatrix} \tag{2.92}$$

Ölçüm gürültüsü, Denklem (2.27)'ye benzer şekilde eklenerek $z_{\text{küresel}}$ elde edilmektedir. Fakat elde edilen z ölçümleri küresel koordinatta ve model dinamiği kartezyen de olduğu için alınan ölçümler Denklem (2.93)-(2.95)'te belirtildiği şekilde küreselden kartezyene dönüştürülmektedir.

$$x = r\ cos\varphi\ cos\theta \tag{2.93}$$

$$y = r\ cos\varphi\ sin\theta \tag{2.94}$$

$$z = r\ sin\varphi \tag{2.95}$$

Koordinat dönüşümleri sonucunda elde edilen $\boldsymbol{z}$ Denklem (2.96)'daki gibi kartezyen koordinatta olmaktadır.

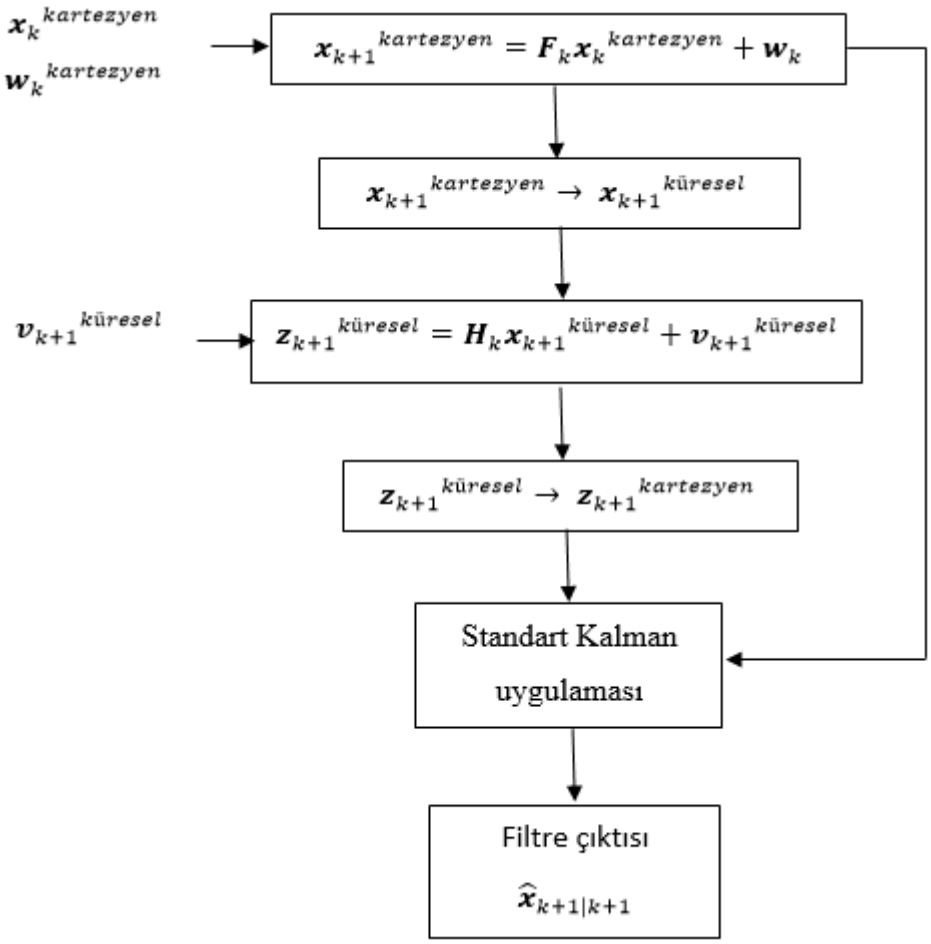
$$\boldsymbol{z} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.96}$$

Ele alınan durumun blok şeması aşağıda görüldüğü gibidir. Blok şemada görülmekte olan standart Kalman uygulaması Şekil 2.7'de belirtilen ile aynıdır.

Lineer olmayan bu koordinat dönüşümlerinden kaynaklı şu sorunlar oluşmaktadır [11];

- Sapma etkisi oluşur. Bu ise sıfır ortalama değerli gürültüye sahip Kalman filtresinin temel varsayımını ihlal etmektedir.
- Liner olmayan dönüşümler için linerleştirmeye dayalı kovaryans hesabı yanlış kovaryansa yol açabilir. Kovaryanslara dayalı Kalman kazancı olması gerektiği gibi olmadığı için performans düşüşü söz konusu olacaktır.

Bu nedenle sapma etkisinin giderilmesi için öne sürülen ölçüm dönüşümü yöntemlerinde (Bölüm 3'te ayrıntıları bulunan ölçüm dönüşüm çeşitleri) ortalama ve kovaryans hesabı üzerinde durulmuştur.



Şekil 2.8: Radar ölçümlerinin küreselde alınması durumunda Kalman Filtresi akış diyagramı.

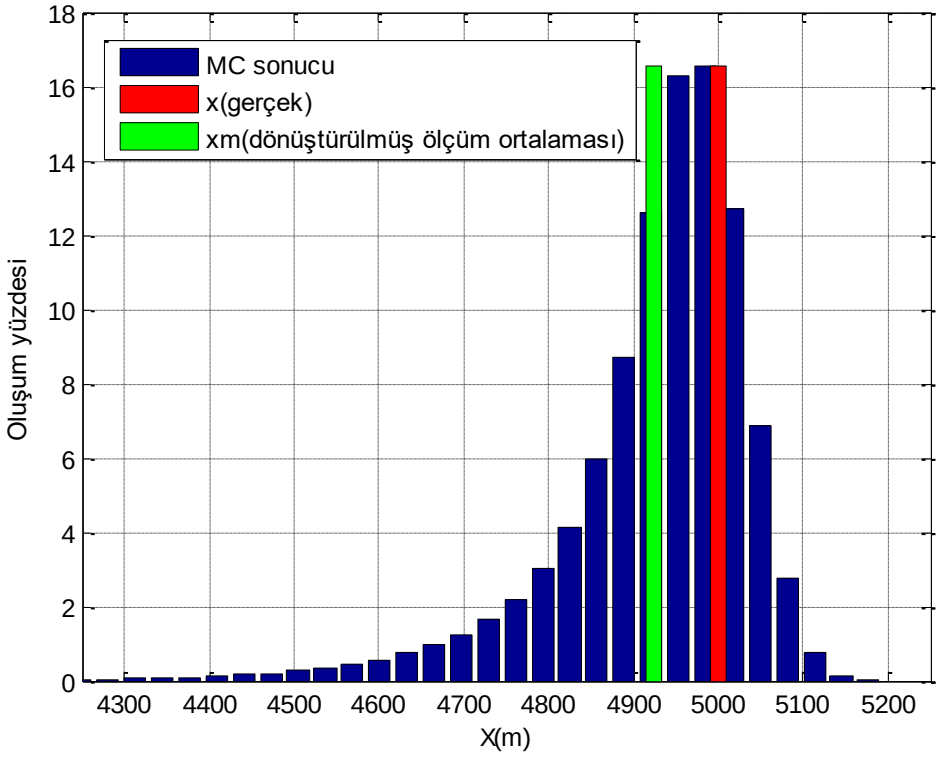
Şekil 2.8'de belirtilen blok şemadaki $x_{k+1}^{kartezyen}$ kartezyen koordinatta üretilen veriyi temsil etmektedir. Fakat radar sistemleri genelde ölçümleri küresel koordinatta almaktadır. Bu nedenle kartezyende üretilen veri Denklem (2.97)-(2.99) yardımıyla küresel koordinata dönüştürülmektedir. Veriler küresel koordinatta üretilmekte ise, $x_{k+1}^{küresel}$ algoritmada direk olarak yerine konulabilir. Bunun yanında küreselden kartezyen koordinata dönüşüm denklemleri ise Denklem (2.93)-(2.95)'te verilmiştir.

$$r = \sqrt{x^2 + y^2 + z^2} \tag{2.97}$$

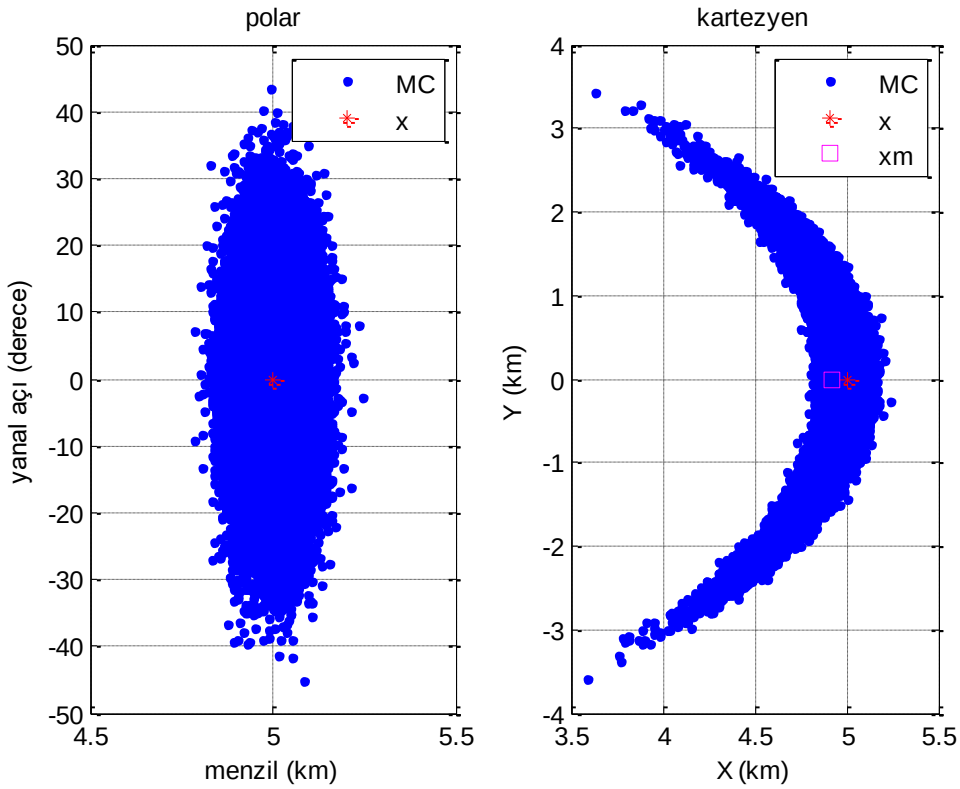
$$\theta = \tan^{-1}(y/x) \tag{2.98}$$

$$\varphi = \tan^{-1}(z/\sqrt{x^2 + y^2}) \tag{2.99}$$

Bu durumu incelemek için menzil mesafesi 5 km, yanal açısı 0^o, ölçüm gürültüsü ilişkisiz ve σ_r^2= 50 m, $\sigma_\theta^2 = 10^o$ olan bir senaryo 10^5 kez çalıştırılmakta ve ölçümler Denklem (2.93)-(2.95) kullanılarak Denklem (2.96)'da belirtilen şekilde kartezyen koordinatta elde edilmektedir ve elde edilen Monte Carlo (MC) sonuçlarının histogramı ise Şekil 2.9'da görüldüğü gibidir.



Şekil 2.9: Ölçümlerin küresel koordinattan kartezyen koordinata dönüşümü sonucunda oluşan sapma etkisi.



Şekil 2.10: Polar ve kartezyen koordinatlarda elde edilen MC sonucu, gerçek konum ve dönüşüm sonucu elde edilen konum karşılaştırması.

Şekillerden de görüldüğü üzere küresel koordinatta alınan ölçümler kartezyen koordinata dönüştürüldüğünde MC sonucunun beklenen değeri ile doğru konum karşılaştırıldığında sapma oluştuğu açıkça görülmektedir. Grafiklerde yer alan *x* değeri hedefin doğru olan konumunu, *xm* değeri ise küresel koordinattan kartezyen koordinata dönüştürülmüş ölçümlerin beklenen değerini temsil etmektedir.

3. RADAR SİSTEMLERİ İÇİN DÖNÜŞTÜRÜLMÜŞ ÖLÇÜMLER

Bu bölümde Bölüm 2.2.4'te verilen CMKF algoritması ile Bölüm 2.4'te belirtilen sapma etkisinin azaltılması, Dönüştürülmüş ölçümler (CM, ing. Converted Measurements) algoritmasının çeşitleri, bu çeşitlerin Kalman filtresine uyarlanması, filtrelerin birbirleri üzerindeki üstünlüklerine ve çeşitli kriterlerdeki performanslarına değinilmektedir. Ayrıca Bölüm 5.4'teki testlerde CM algoritmalarının hata performans farkı olmaması durumunda, uygulamaya göre CM dönüşümünün seçilmesi için CM dönüşümlerinin yansızlık, tutarlılık gibi kriterlerdeki performansları incelenmektedir.

3.1. CM Algoritması ile Sapma Etkisinin Azaltılması

Bölüm 2.2'de kısaca takip algoritmalarına, radar ölçümlerinin genelde küresel koordinatta alındığına, takip algoritmalarında ise genelde doğrusal denklemlerin kullanıldığına değinilmişti. Takip algoritmalarında denklemlerin doğrusal olması için çeşitli doğrusallaştırma yöntemleri kullanılmaktadır. Örneğin, bu amaçla sık olarak kullanılan EKF algoritmasında Taylor açılımı yöntemi kullanılarak doğrusallaştırma yapılmaktadır (Bölüm 2.2.2). Diğer yandan [11]'de doğru kovaryansa sahip CMKF algoritmasının, EKF algoritmasından daha küçük hatalara sahip olduğu ve EKF performansının 1.5^o ve üstü açı sapmasına sahip uzun menziller için zayıfken; CMKF performansının 10^o açı sapmasına sahip olsa bile daha kararlı sonuçlar verdiği belirtilmektedir.

CM algoritmasında küresel koordinatta alınan ölçüm verileri takip algoritmasında kullanılmak için küresel koordinattan kartezyen koordinata dönüştürülmektedir. Bu dönüşüm için çeşitli dönüşüm denklemleri kullanılmaktadır.

Bölüm 2.2.4'te CMKF algoritmasından ve genel dönüşüm yaklaşımından bahsedilmektedir. Bu algoritmada küresel koordinatta alınan ölçümlerde mevcut olan Gauss ölçüm gürültüsü takip algoritmasında kullanılmak üzere kartezyen koordinata dönüştürüldüğünde ölçüm gürültüsünün ortalaması ve kovaryans hesabı

değişmektedir. Koordinat dönüşümünün oluşturduğu sapma etkisinin azaltılması için öne sürülen CMKF algoritmaları başlıca şunlardır;

- Converted Mesuarements Kalman Filter with Linearization (CMKF-L)
  Bu filtrede standart koordinat dönüşümleri kullanılır ve lineerleştirmeye dayalı kovaryans matrisi ($\boldsymbol{R}$) gerçek menzil ve açı varyanslarına bağlıdır. Pratikte gerçek menzil ve açı değerleri bilinmediği için bunların yerine ölçüm değerleri kullanılmaktadır.
- Converted Mesuarements Kalman Filter with Debiasing (CMKF-D) (1993) [33]
  CMKF-L yönteminde gerçek menzil ve açı değerlerinin kullanılması pratikte mümkün olmadığı için CMKF-D yöntemi bu soruna alternatif olarak ölçümlerden faydalanılarak oluşturulan öngörülü ortalama ve kovaryans matrisini kullanmaktadır. Koordinat dönüşümlerinde standart dönüşümden bu yöntemde önerilen ortalama çıkarılarak ölçümler yansız (debiased) yapılmaktadır. Kullanılması önerilen kovaryans matrisi ise yansızlaştırılmaya dayalıdır.
- Converted Mesuarements Kalman Filter with Unbiasing (CMKF-U) (1998) [34]
  Bu filtrede yansızlaştırmak için farklı bir kovaryans matrisi önerilmektedir.
- Converted Mesuarements Kalman Filter with Modified Unbiasing (CMKF-MU) (2004) [35]
  Bu dönüşümde yine yansızlaştırma işlemi yapılmış ve kovaryans matrisi ölçüm değerlerine bağlı şekilde ifade edilmiştir. Kullanılan dönüşüm ise CMKF-U yönteminde kullanılan dönüşümün modifiye edilmiş halidir.
- Converted Mesuarement Kalman Filter with Decorrelated Unbiased (CMKF-DU) (2009) [36], [37]
  Bu dönüşüm önceki dönüşümlerin az da olsa ölçüm gürültüleriyle ilişkili olduğunu, dolayısıyla yanlı olduklarını iddia etmektedir. Bu nedenle kovaryans tahmini ile ölçüm gürültülerinin ilişkisiz hale getirilmesine dayalı yansızlaştırılma işlemi yapılması önerilmiştir. Elde edilen kovaryans matrisi ölçüm verilerine dolayısıyla ölçüm varyanslarına bağlı değildir. Bu yöntemde ölçüm verileri yerine tahmin verileri kullanılmaktadır ve kovaryans matrisi de bu tahmini menzil ve açı varyanslarına bağlıdır.

3.2. İki Boyutta CM Algoritmaları

Bu bölümde üzerinde durulmakta olan iki boyutlu sistem Şekil 2.1'de belirtildiği şekildedir. Bu durumda radardan alınan ölçümler Denklem (3.1)-(3.2)'de belirtildiği gibi olmaktadır. Buradaki r gerçek menzil ve θ yansal açı değerini; $\tilde{r}$ ve $\tilde{\theta}$ ise ortalaması 0, varyansları sırasıyla σ_r^2 ve σ_θ^2 olan Gauss ölçüm gürültülerini temsil etmektedir.

$$r_m = r + \tilde{r} \tag{3.1}$$

$$\theta_m = \theta + \tilde{\theta} \tag{3.2}$$

CM algoritmasında kullanılması için elde edilen kartezyen koordinata dönüştürülmüş ölçüm vektörü $\boldsymbol{z}^c$ Denklem (3.3)'te, kullanılan kovaryans matrisi $\boldsymbol{R}$ ise Denklem (3.4)'te belirtildiği gibidir.

$$\boldsymbol{z}^c = \begin{bmatrix} x_m^c \\ y_m^c \end{bmatrix} = \begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} - \boldsymbol{\mu} \tag{3.3}$$

$$\boldsymbol{R} = \begin{bmatrix} R_{11} & R_{12} \\ R_{12} & R_{22} \end{bmatrix} \tag{3.4}$$

Denklem (3.3)'te mevcut olan $\boldsymbol{\mu}$ vektörü ve $\boldsymbol{R}$ kovaryans matrisi farklı CM algoritmaları için takip eden bölümlerde verilmektedir.

3.2.1. CM_L

Bu algoritma için dönüştürülmüş ölçümler Denklem (3.5)'te belirtildiği gibidir. Dolayısıyla Denklem (3.3)'te mevcut olan $\boldsymbol{\mu}$ vektörü bu algoritma için 0 vektörünü temsil etmektedir [33].

$$\begin{bmatrix} x_m^{L,2D} \\ y_m^{L,2D} \end{bmatrix} = \begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} \tag{3.5}$$

[12]'de bu dönüşümde sapma olduğu belirtilmiştir. Bunun için Denklem (3.5)'te r_m ve θ_m değerleri Denklem (3.1) ve Denklem (3.2) olarak yazılır ve beklenen değeri bulunmak istenirse sonuç Denklem (3.6)'daki gibi bulunmaktadır. Bu durumda dönüştürülmüş ölçümün hedefin gerçek açı varyansına bağlı olarak sapmaya sahip olduğu açıkça görülmektedir.

$$E\begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} = \begin{bmatrix} (r+\tilde{r})\cos(\theta+\tilde{\theta}) \\ (r+\tilde{r})\sin(\theta+\tilde{\theta}) \end{bmatrix} = e^{-\sigma_\theta^2/2}\begin{bmatrix} rcos\theta \\ rsin\theta \end{bmatrix} \tag{3.6}$$

Bu durumda Denklem (3.4)'te belirtilen $\boldsymbol{R}$ kovaryans matrisinin elemanları ise Denklem (3.7)-(3.9)'da verilmektedir [33], [38].

$${R_{L,2D}}^{11} = r^2\sigma_\theta^2 sin^2\theta + \sigma_r^2 cos^2\theta \tag{3.7}$$

$${R_{L,2D}}^{22} = r^2\sigma_\theta^2 cos^2\theta + \sigma_r^2 sin^2\theta \tag{3.8}$$

$${R_{L,2D}}^{12} = (\sigma_r^2 - r^2\sigma_\theta^2) sin\,\theta cos\,\theta \tag{3.9}$$

3.2.2. CM_D

CM_D yöntemi için Denklem (3.3)'te mevcut olan $\boldsymbol{\mu}$ vektörü Denklem (3.10)'da, dönüştürülmüş ölçüm denklemleri ise Denklem (3.11)'de verilmektedir [33].

$$\boldsymbol{\mu} = \begin{bmatrix} r_m cos\theta_m (e^{-\sigma_\theta^2} - e^{-\sigma_\theta^2/2}) \\ r_m sin\theta_m (e^{-\sigma_\theta^2} - e^{-\sigma_\theta^2/2}) \end{bmatrix} \tag{3.10}$$

$$\begin{bmatrix} x_m^{d,2D} \\ y_m^{d,2D} \end{bmatrix} = \begin{bmatrix} r_m cos\theta_m (1 - e^{-\sigma_\theta^2} + e^{-\sigma_\theta^2/2}) \\ r_m sin\theta_m (1 - e^{-\sigma_\theta^2} + e^{-\sigma_\theta^2/2}) \end{bmatrix} \tag{3.11}$$

Bu algoritma için kullanılan $\boldsymbol{R}$ kovaryans matrisinin elemanları şöyledir;

$$
\begin{aligned}
R_{d,2D}^{11} = r_m^2 e^{-2\sigma_\theta^2}\big[&cos^2\theta_m\big(\cosh 2\sigma_\theta^2 - \cosh \sigma_\theta^2\big) \\
&+ sin^2\theta_m\big(\sinh 2\sigma_\theta^2 - \sinh \sigma_\theta^2\big)\big] \\
&+ \sigma_r^2 e^{-2\sigma_\theta^2}\big[cos^2\theta_m\big(2\cosh 2\sigma_\theta^2 - \cosh \sigma_\theta^2\big) \\
&+ sin^2\theta_m\big(2\sinh 2\sigma_\theta^2 - \sinh \sigma_\theta^2\big)\big]
\end{aligned}
\tag{3.12}
$$

$$
\begin{aligned}
R_{d,2D}^{22} = r_m^2 e^{-2\sigma_\theta^2}\big[&sin^2\theta_m\big(\cosh 2\sigma_\theta^2 - \cosh \sigma_\theta^2\big) \\
&+ cos^2\theta_m\big(\sinh 2\sigma_\theta^2 - \sinh \sigma_\theta^2\big)\big] \\
&+ \sigma_r^2 e^{-2\sigma_\theta^2}\big[sin^2\theta_m\big(2\cosh 2\sigma_\theta^2 - \cosh \sigma_\theta^2\big) \\
&+ cos^2\theta_m\big(2\sinh 2\sigma_\theta^2 - \sinh \sigma_\theta^2\big)\big]
\end{aligned}
\tag{3.13}
$$

$$
R_{d,2D}^{12} = \sin\theta_m \cos\theta_m \, e^{-4\sigma_\theta^2}\left[\sigma_r^2 + (r_m^2 + \sigma_r^2)\left(1 - e^{\sigma_\theta^2}\right)\right] \tag{3.14}
$$

3.2.3. CM_U

Bu algoritmada dönüşümün CM_L ve CM_D'ye göre daha yansız hale getirilebilmesi adına [34]'te λ_θ olarak ifade edilen Gauss gürültüsü için kompanzasyon faktörü öne sürülmektedir ve kullanılan dönüşüm Denklem (3.15)'te verilmektedir [34].

$$
\begin{aligned}
\begin{bmatrix} x_m^{u,2D} \\ y_m^{u,2D} \end{bmatrix} &= \begin{bmatrix} r_m cos\theta_m e^{\sigma_\theta^2/2} \\ r_m sin\theta_m e^{\sigma_\theta^2/2} \end{bmatrix} = e^{\sigma_\theta^2/2}\begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} \\
&= \lambda_\theta^{-1}\begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix}
\end{aligned}
\tag{3.15}
$$

Bu yöntem için $\boldsymbol{R}$ kovaryans matrisinin elemanları ise aşağıdaki şekilde güncellenmiştir.

$$
R_{u,2D}^{11} = \frac{1}{2}({r_m}^2 + {\sigma_r}^2)(1 + \cos(2\theta_m)\,{\lambda_\theta}') + ({\lambda_\theta}^{-2} - 2){r_m}^2 cos^2\theta_m \tag{3.16}
$$

$$R_{u,2D}^{22} = \frac{1}{2}({r_m}^2 + {\sigma_r}^2)(1 - \cos(2\theta_m)\,{\lambda_\theta}') + ({\lambda_\theta}^{-2} - 2){r_m}^2 sin^2\theta_m \tag{3.17}$$

$$R_{u,2D}^{12} = \frac{1}{2}({r_m}^2 + {\sigma_r}^2)(\sin(2\theta_m)\,{\lambda_\theta}') + ({\lambda_\theta}^{-2} - 2){r_m}^2 \cos\theta_m \sin\theta_m \tag{3.18}$$

Kompanzasyon faktörü olarak bilinen λ_θ değeri ve $\boldsymbol{R}$ kovaryans matrisinde kullanılan değeri Denklem (3.19) ve Denklem (3.20) yardımıyla bulunmaktadır.

$$\lambda_\theta = E\{cos\tilde{\theta}\} = e^{-\sigma_\theta^2/2} \tag{3.19}$$

$${\lambda_\theta}' = E\{cos2\tilde{\theta}\} = e^{-2\sigma_\theta^2} = {\lambda_\theta}^4 \tag{3.20}$$

Bu yöntemde ölçüm dönüşümü gerçek menzil ve yansal açı değerleri üzerinden koşullandırılarak türetilirken, kovaryans matrisi ölçümler üzerinden koşullandırılarak türetilmiştir[35].

3.2.4. CM_MU

CM_MU yöntemi CM_U yönteminin modifiye edilmiş hali olarak nitelendirilebilir. Dönüşümler için kullanılan kovaryans matrisi ölçüm değerleri üzerinden koşullandırılırken, ölçüm dönüşümleri gerçek değerler üzerinden koşullandırılmaktadır [39], [40]. Bu uyuşmazlığa [35]'te CM_MU yöntemi ile değinilmiştir. Bu algoritmada öne sürülen yansız ölçüm dönüşümü Denklem (3.21)'de ve kovaryans matris elemanları Denklem (3.22)-(3.24)'te verilmektedir. Bu denklemlerde yer alan λ_θ değeri CM_U yönteminde kullanılan ve Denklem (3.19)'da belirtilen kompanzasyon faktörü ile aynıdır.

$$\begin{aligned}\begin{bmatrix} x_m^{mu,2D} \\ y_m^{mu,2D} \end{bmatrix} &= \begin{bmatrix} r_m cos\theta_m e^{-\sigma_\theta^2/2} \\ r_m sin\theta_m e^{-\sigma_\theta^2/2} \end{bmatrix} = e^{-\sigma_\theta^2/2} \begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} \\ &= \lambda_\theta \begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix}\end{aligned} \tag{3.21}$$

$$R_{mu,2D}^{11} = \frac{1}{2}({r_m}^2 + {\sigma_r}^2)(1 + \cos(2\theta_m)\,{\lambda_\theta}') - {\lambda_\theta}^{-2}{r_m}^2 cos^2\theta_m \tag{3.22}$$

$$R_{mu,2D}^{22} = \frac{1}{2}({r_m}^2 + {\sigma_r}^2)(1 - \cos(2\theta_m)\,{\lambda_\theta}') - {\lambda_\theta}^{-2}{r_m}^2 sin^2\theta_m \tag{3.23}$$

$$R_{mu,2D}^{12} = \frac{1}{2}({r_m}^2 + {\sigma_r}^2)(\sin(2\theta_m)\,{\lambda_\theta}') - {\lambda_\theta}^{-2}{r_m}^2 \cos\theta_m \sin\theta_m \tag{3.24}$$

3.2.5. CM_DU

Bölüm 3.3.1'de CM_U yönteminin yansız, bir dönüşüm olduğu gösterilmektedir. Bu nedenle bu algoritmada CM_U dönüşümü uygulanmaktadır. Fakat bunun yanında Bölüm 3.3.3'te ise CM_MU dönüşümünün CM_U dönüşümüne göre daha küçük MSE hatası oluşturduğu gösterilmektedir. CM_DU yöntemi ise CM_U yönteminin yansızlığını kullanmakta ve buna ek olarak CM_MU kadar az MSE için Bölüm 3.3.3'te belirtildiği şekilde performans artışı yapılmıştır [41]. Bunun yanında CM_L,CM_D, CM_U ve CM_MU dönüşümlerinde hata kovaryans tahmini ile ölçüm gürültüsü ilişkilidir. CM_DU algoritmasında hata kovaryans tahmini ile ölçüm gürültüsünün ilişkili olmasının önüne geçilmektedir. Bu nedenle kovaryans matrisi hesaplaması yapılırken kovaryans denklemlerinden de görüldüğü üzere ölçüm değerleri değil de tahmin değerleri kullanılmaktadır [36], [41]-[43].

Bu algoritma için kullanılan ölçüm dönüşümü Denklem (3.15)'te verilen CM_U dönüşümü ile aynıdır.

$$\begin{bmatrix} x_m^{du,2D} \\ y_m^{du,2D} \end{bmatrix} = \begin{bmatrix} x_m^{u,2D} \\ y_m^{u,2D} \end{bmatrix} = \begin{bmatrix} r_m cos\theta_m e^{\sigma_\theta^2/2} \\ r_m sin\theta_m e^{\sigma_\theta^2/2} \end{bmatrix} = e^{\sigma_\theta^2/2} \begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} = \lambda_\theta^{-1} \begin{bmatrix} r_m cos\theta_m \\ r_m sin\theta_m \end{bmatrix} \tag{3.25}$$

Algoritmada kullanılan ***R*** kovaryans matrisi ise Denklem (3.26)-(3.32)'de verilmektedir. Denklem (3.29)-(3.32)'deki x_t ve y_t tahmin değerlerini temsil etmektedir [36], [37].

$$R_{du,2D}^{11} = \frac{1}{2}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[1 + \cos(2\theta_t)e^{-2\sigma_\theta^2}\, e^{-2\sigma_{\theta_t}^2}\right] e^{\sigma_\theta^2} - \frac{1}{2}\left({r_t}^2 + {\sigma_{r_t}}^2\right)\left[1 + \cos(2\theta_t)\, e^{-2\sigma_{\theta_t}^2}\right] \tag{3.26}$$

$$R_{du,2D}^{22} = \frac{1}{2}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[1 - \cos(2\theta_t)e^{-2\sigma_\theta^2}\, e^{-2\sigma_{\theta_t}^2}\right] e^{\sigma_\theta^2} - \frac{1}{2}\left({r_t}^2 + {\sigma_{r_t}}^2\right)\left[1 - \cos(2\theta_t)\, e^{-2\sigma_{\theta_t}^2}\right] \tag{3.27}$$

$$R_{du,2D}^{12} = \frac{1}{2}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[\sin(2\theta_t)e^{-2\sigma_\theta^2}\, e^{-2\sigma_{\theta_t}^2}\right] e^{\sigma_\theta^2} - \frac{1}{2}\left({r_t}^2 + {\sigma_{r_t}}^2\right)\left[\sin(2\theta_t)\, e^{-2\sigma_{\theta_t}^2}\right] \tag{3.28}$$

$$r_t = \sqrt{{x_t}^2 + {y_t}^2} \tag{3.29}$$

$$\theta_t = \tan^{-1}\left(\frac{y_t}{x_t}\right) \tag{3.30}$$

$${\sigma_{r_t}}^2 = \begin{bmatrix}\frac{\partial r_t}{\partial x_t} & \frac{\partial r_t}{\partial y_t}\end{bmatrix}\begin{bmatrix}P_{xx} & P_{xy}\\ P_{xy} & P_{yy}\end{bmatrix}\begin{bmatrix}\frac{\partial r_t}{\partial x_t}\\ \frac{\partial r_t}{\partial y_t}\end{bmatrix} = \frac{1}{{r_t}^2}\begin{bmatrix}x_t & y_t\end{bmatrix}\begin{bmatrix}P_{xx} & P_{xy}\\ P_{xy} & P_{yy}\end{bmatrix}\begin{bmatrix}x_t\\ y_t\end{bmatrix} = \frac{P_{xx}{x_t}^2 + 2P_{xy}x_t y_t + P_{yy}{y_t}^2}{{x_t}^2 + {y_t}^2} \tag{3.31}$$

$${\sigma_{\theta_t}}^2 = \begin{bmatrix}\frac{\partial \theta_t}{\partial x_t} & \frac{\partial \theta_t}{\partial y_t}\end{bmatrix}\begin{bmatrix}P_{xx} & P_{xy}\\ P_{xy} & P_{yy}\end{bmatrix}\begin{bmatrix}\frac{\partial \theta_t}{\partial x_t}\\ \frac{\partial \theta_t}{\partial y_t}\end{bmatrix} = \frac{1}{{r_t}^4}\begin{bmatrix}-y_t & x_t\end{bmatrix}\begin{bmatrix}P_{xx} & P_{xy}\\ P_{xy} & P_{yy}\end{bmatrix}\begin{bmatrix}-y_t\\ x_t\end{bmatrix} = \frac{P_{xx}{y_t}^2 - 2P_{xy}x_t y_t + P_{yy}{x_t}^2}{({x_t}^2 + {y_t}^2)^2} \tag{3.32}$$

3.3. İki Boyutlu CM için Performans Karşılaştırmaları

Bu bölümde Şekil 2.1'de gösterilmekte olan iki boyutlu sistem için CM algoritmalarının farklı kriterler de karşılaştırmaları mevcuttur. Bu kriterler yaygın olarak kullanılan, [37]'de ve [42]'de belirtilen kriterlere göre yapılmıştır.

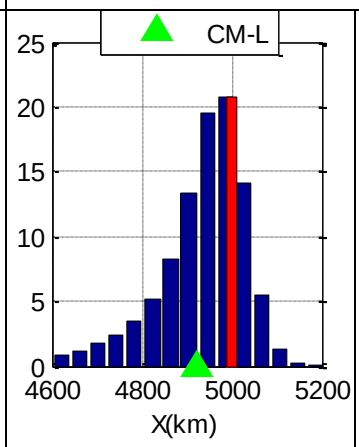
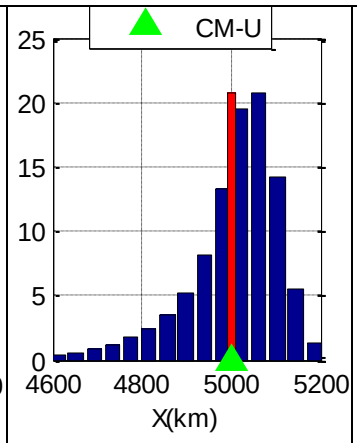
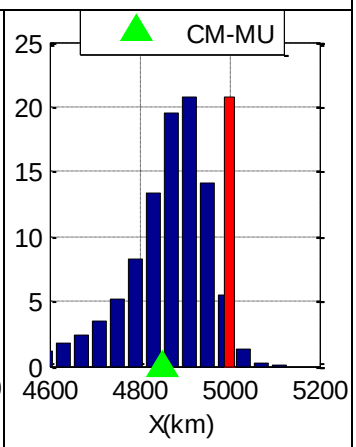
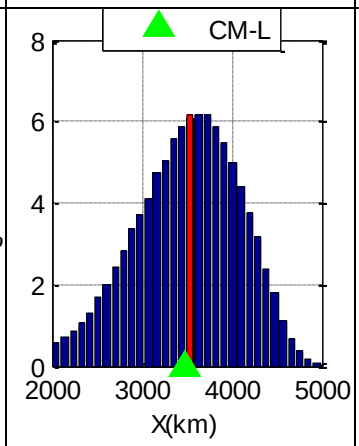
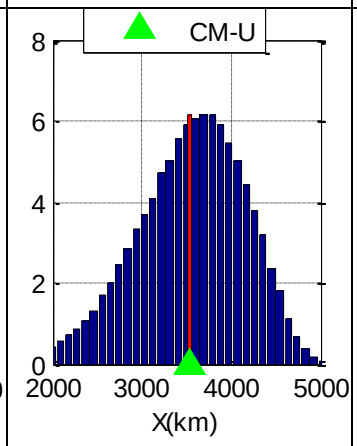
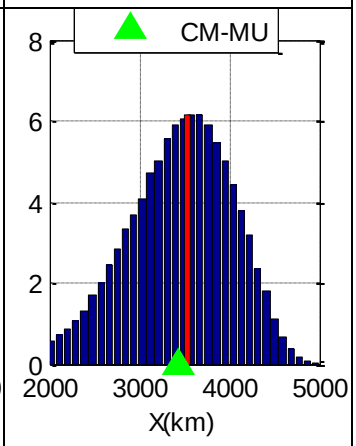
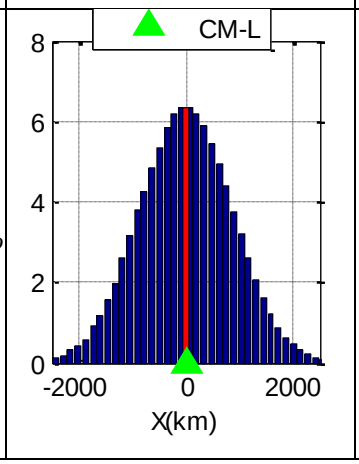
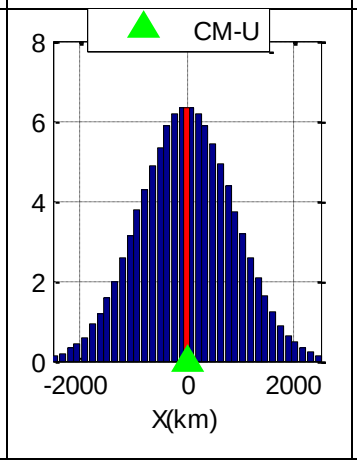
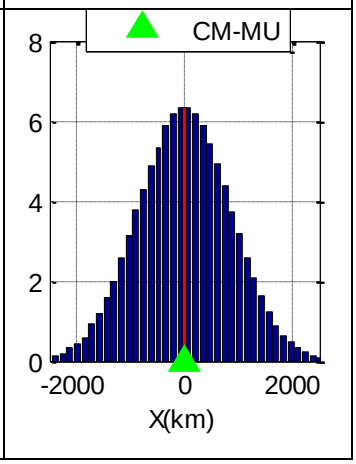
3.3.1. Yansızlık

Yansızlık CM için önemli bir kriterdir. Çünkü dönüştürülmüş ölçümlerin yansız olması yani sapmaya sahip olmaması istenmektedir. Bunun nedenlerine Bölüm 2.4 ve Bölüm 3.1'de değinilmişti. Bunun yanında takip sistemleri için en yaygın olan Kalman filtresinin kullanılmasında yapılan sıfır ortalamalı Gauss yaklaşımının sağlanması önemli bir şarttır. Bu bölümde yaygın olarak kullanılan CM-U, CM-MU ve CM-DU dönüşümlerinin standart dönüşüm olan CM-L ve birbirleri arasındaki yansızlık karşılaştırılması yapılacaktır.

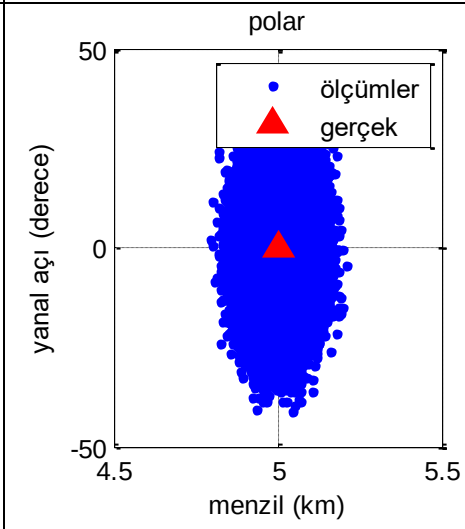
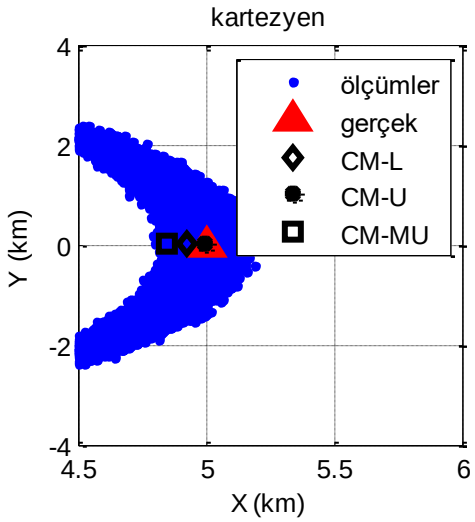
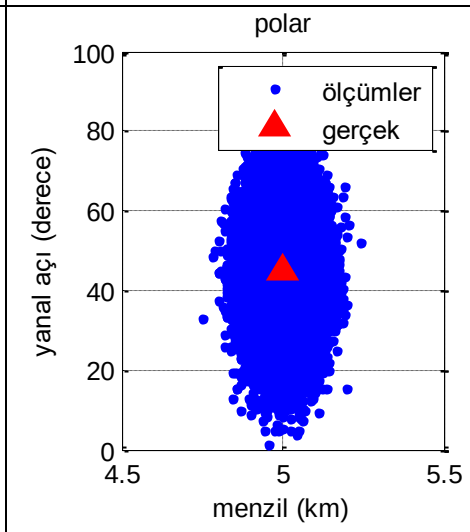
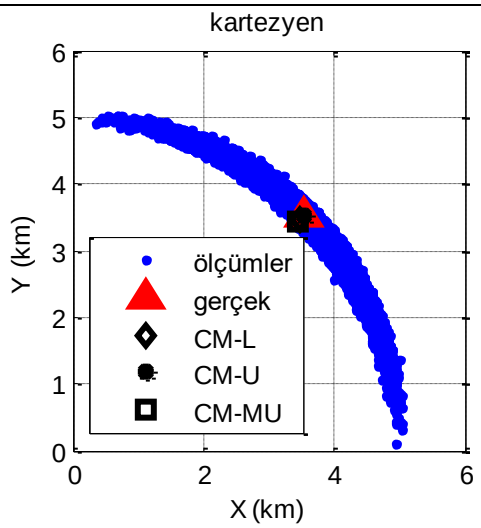
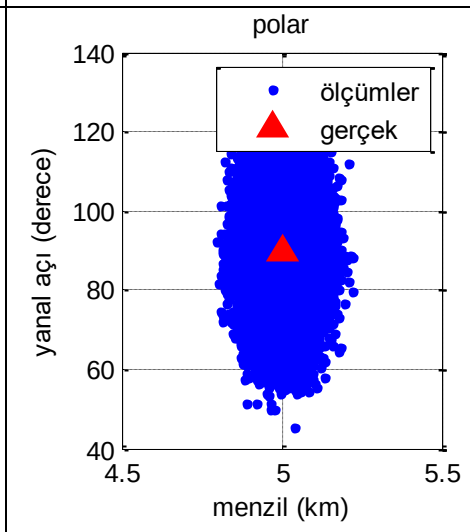
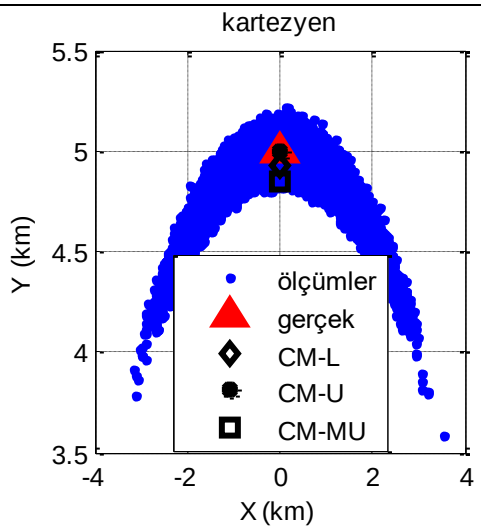
Yansızlık testi için menzil mesafesi 5 km, yanal açısı $0^o - 90^o$, ölçüm gürültüsü ilişkisiz ve σ_r= 50 m, $\sigma_\theta = 10^o$ olan bir senaryo 10^5 kez çalıştırılmakta ve elde edilen ölçümler belirtilen dönüşümler yapılarak kartezyen koordinata dönüştürülmektedir [42], [43]. Daha sonra ise her bir dönüşümde elde edilen kartezyen ölçümlerin beklenen değerleri bulunarak hedefin gerçek konumu ile kıyaslanmaktadır. Bunun için ilki Tablo 3.1'de verilmekte olan Gauss yansızlık testi *x* koordinatı için yapılmıştır. Bu durum için yanal açının $0^o - 90^o$'yi taraması için 0^o, 45^o ve 90^o için ayrı ayrı test edilmiştir. Elde edilen MC sonuç histogramlarına, kırmızı bar ile gösterilen hedefin gerçek konumuna ve her grafikte belirtilen dönüşüm sonuçlarına bakıldığında en yansız yöntemin CM-U yöntemi olduğu açıkça görülmektedir. Bunu yanında Bölüm 3.2.5'te belirtildiği üzere CM-DU yöntemi CM-U dönüşümünü kullandığı için CM-DU dönüşümü de en yansız dönüşüm olmaktadır.

Yapılan ikinci testte ise yine aynı senaryo kullanılmıştır ve ölçümlerin hem polar hem de kartezyen koordinattaki MC sonuçları Tablo 3.2'de gösterilmektedir. Bunun yanında her dönüşüm çeşidi için dönüşüm sonucundaki beklenen değerler ve hedefin gerçek konumu grafik üzerinde gösterilmiştir. Bu testte de görüldüğü üzere CM-U (aynı şekilde CM-DU dönüşümü de en yansızdır çünkü CM-U dönüşümünü içerir) yönteminde en yansız sonuç elde edilmiştir.

Tablo 3.1: CM dönüşüm çeşitleri için Gauss yansızlık testi.

θ	X koordinatı değerine karşı MC sonucunun oluşum yüzdesi histogramı		
0^o			
45^o			
90^o			

Tablo 3.2: Polar ve kartezyen koordinattaki ölçümler ve CM çeşitleri için beklenen değerleri.

θ	Polar koordinat	Kartezyen koordinat
0^o		
45^o		
90^o		

Tablo 3.3'te ise CM çeşitlerinin dönüşüm yanlılığı (sapması) verilmektedir [41];

Tablo 3.3: İki boyutta CM yöntemlerinin dönüşüm yanlılığı.

Dönüşüm Yöntemi	Yanlılık
CM-L	$r(e^{-\sigma_\theta^2/2}-1)$
CM-U	0
CM-MU	$r(e^{-\sigma_\theta^2}-1)$
CM-DU	0

Tablo 3.4: İki boyutta CM yöntemlerinin dönüşüm yanlılık çarpanı.

Dönüşüm Yöntemi	Yanlılık çapanı
CM-L	$e^{-\sigma_\theta^2/2}$
CM-U	1
CM-MU	$e^{-\sigma_\theta^2}$
CM-DU	1

CM-L dönüşümü için yanlılık ve yanlılık çarpanı aşağıda belirtilen şekilde bulunmaktadır. Diğer yöntemler içinde aynı yaklaşım kullanılarak tablolardaki sonuçlar bulunabilir.

$$\begin{aligned} E[r_m cos\theta_m] &= E\big[(r+\tilde{r})cos\big(\theta+\tilde{\theta}\big)\big] \\ &= E[r]E\big[cos\big(\theta+\tilde{\theta}\big)\big] + E[\tilde{r}]E\big[cos\big(\theta+\tilde{\theta}\big)\big] \\ &= rE\big[cos\big(\theta+\tilde{\theta}\big)\big] \\ &= rE\big[cos(\theta)cos\big(\tilde{\theta}\big) - sin(\theta)sin\big(\tilde{\theta}\big)\big] \\ &= rE\big[cos(\theta)cos\big(\tilde{\theta}\big)\big] = rcos(\theta)E\big[cos\big(\tilde{\theta}\big)\big] \\ &= rcos(\theta)e^{-\sigma_\theta^2/2} \end{aligned} \tag{3.33}$$

Denklem (3.46)'da $\tilde{r}$ ortalaması sıfır olan Gauss gürültüsü olduğundan $E[\tilde{r}] = 0$ eşitliği; $E[sin(\theta)]$ ise sin fonksiyonunun simetrik bir fonksiyon olmasından değeri sıfıra eşit olur.

CM-L için yanlılık hesabı ise Denklem (3.34)-(3.36)'da verilmektedir. Aynı şekilde diğer dönüşümler içinde bu mantık kullanılabilir.

$$Sapma_x^{CM-L} = E[r_m cos\theta_m] - rcos\theta = rcos(\theta)e^{-\sigma_\theta^2/2} - rcos(\theta) \\ = rcos(\theta)(e^{-\sigma_\theta^2/2} - 1) \tag{3.34}$$

$$Sapma_y^{CM-L} = E[r_m sin\theta_m] - rsin\theta = rsin(\theta)e^{-\sigma_\theta^2/2} - rsin(\theta) \\ = rsin(\theta)(e^{-\sigma_\theta^2/2} - 1) \tag{3.35}$$

Toplam sapma miktarı her iki eksen için meydana gelen sapma miktarlarının mesafesine eşittir. Dolayısıyla toplam sapma miktarı olan $Sapma_{toplam}^{CM-L}$ Denklem (3.36)'da belirtildiği gibi bulunur.

$$Sapma_{toplam}^{CM-L} = \sqrt{(Sapma_x^{CM-L})^2 + (Sapma_y^{CM-L})^2} \\ = \sqrt{(rcos(\theta)(e^{-\sigma_\theta^2/2} - 1))^2 + (rsin(\theta)(e^{-\sigma_\theta^2/2} - 1))^2} \\ = \sqrt{r^2(e^{-\sigma_\theta^2/2} - 1)^2[cos(\theta)^2 + sin(\theta)^2]} = \sqrt{r^2(e^{-\sigma_\theta^2/2} - 1)^2} \\ = r(e^{-\sigma_\theta^2/2} - 1) \tag{3.36}$$

3.3.2. Tutarlılık

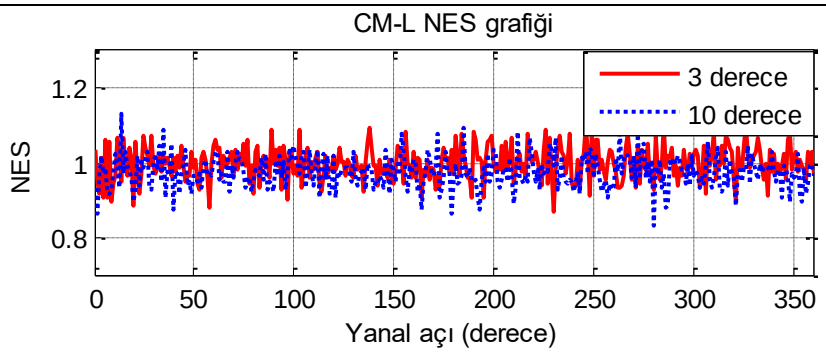
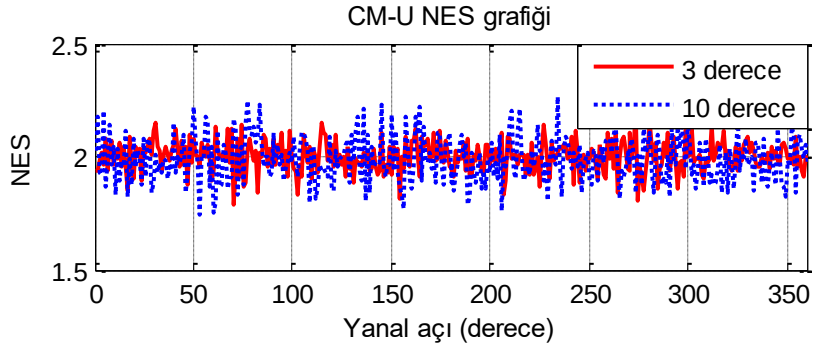
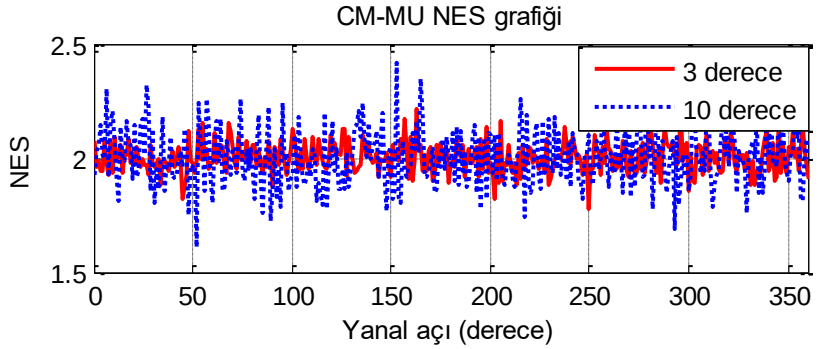
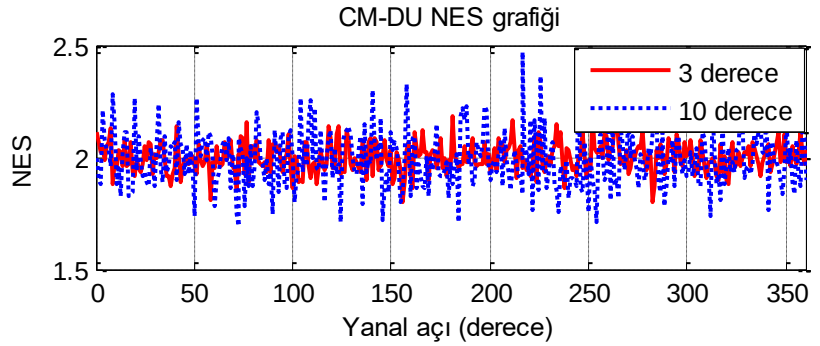
CM dönüşümlerinin önemli kriterlerinden biride ne kadar tutarlı bir dönüşüm olduğudur. Tutarlılık kriterini test etmek için [12]'de belirtilen normalize hata karesi (NES, *ing.* Normalized Error Squared) değeri ölçüt olarak kullanılmaktadır. NES değerini hesaplamak için kullanılan NES formülü Denklem (3.37)'de verilmiştir. Denklemdeki N Monte-Carlo sayısını, X_i hedefin gerçek konum vektörünü, $\hat{X}_i$ i'inci Monte-Carlo'da elde edilmiş dönüştürülmüş konum vektörünü ve R_i ise hata kovaryans matrisini temsil etmektedir.

$$NES = \frac{1}{N}\sum_{i=1}^{N}(X_i - \hat{X}_i)^T {R_i}^{-1}(X_i - \hat{X}_i) \tag{3.37}$$

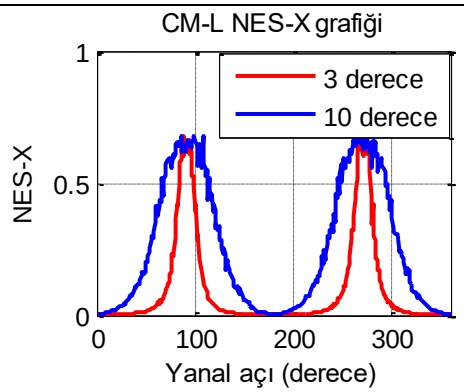
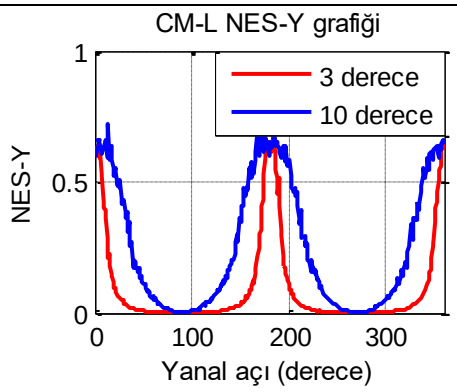
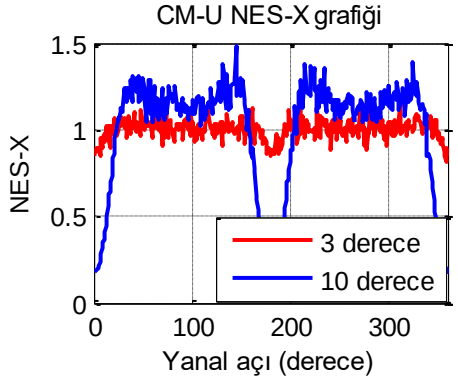
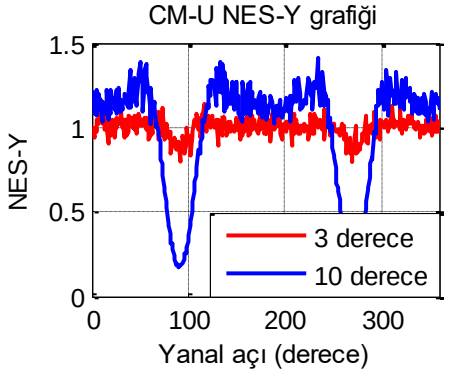
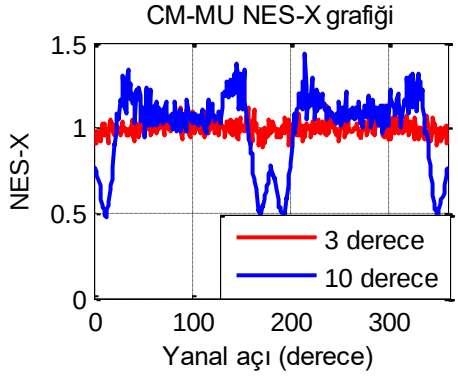
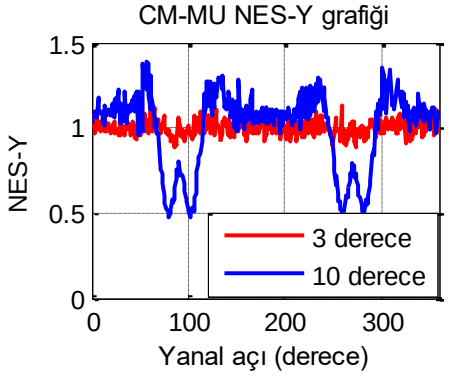
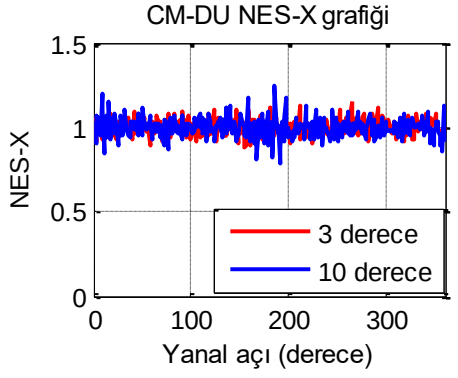
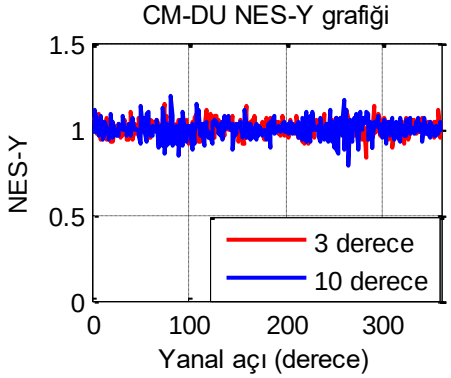
Tutarlılık testi için menzil mesafesi 5 km, yanal açısı $0^o - 360^o$, ölçüm gürültüsü ilişkisiz ve σ_r= 50 m, $\sigma_\theta = 3^o$ ve $\sigma_\theta = 10^o$ için senaryo 10^3 kez çalıştırılmıştır. Tablo 3.5 ve Tablo 3.6'da CM-L, CM-U, CM-MU ve CM-DU için tutarlık kriteri olan NES grafikleri verilmiştir.

NES değerinin sistemin boyutuna eşit olması dönüşümün tutarlı olduğu anlamına gelmektedir. Tablo 3.5'te CM-L dönüşümünün toplam NES grafiğine bakıldığında iki boyutlu olan sistem için NES değerinin 1 değerine yakınsadığı bu nedenle CM-L dönüşümünün tutarsız olduğu açıkça görülmektedir. CM-U, CM-MU ve CM-DU için elde edilen toplam NES değeri 2 değerine yakınsamaktadır dolayısıyla üç dönüşümde tutarlı sonuçlar vermektedir. Fakat her eksen için ayrı ayrı bakıldığında durum farklı olmaktadır. Tablo 3.6'da dönüşümlerin x ve y eksenleri için ayrı ayrı NES grafikleri verilmektedir. Bu durumda tek eksen için NES değerine tutarlılık açısından bakılırsa değerin 1 değerine yakınsaması dönüşümün tutarlı olduğunu göstermektedir. Bu açıdan Tablo 3.6'ya bakıldığında CM-DU dönüşümü tutarlı iken; CM-U ve CM-MU dönüşümleri bazı açı değerlerinde tutarlı bazılarında tutarsız olduğu görülmektedir. Örneğin CM-U dönüşümün x bileşeni için elde edilen NES grafiğine bakıldığında 0^o ve 180^o'de, y bileşeni için elde edilende ise 90^o ve 270^o yakınlarında tutarsızlık söz konusudur. CM-MU dönüşümü CM-U dönüşümü ile benzer bir tutum sergilerken, CM-DU dönüşümü her açı değerinde tutarlılık göstermektedir. CM-L dönüşümünün ise çoğunlukla tutarsız olduğu söylenebilir. Ayrıca yanal açının standart sapma değerinin etkisi elde edilen grafiklerden açıkça görülmektedir. Örneğin CM-U dönüşümü için NES-X grafiğine bakıldığında yanal açının standart sapma değeri 3^o olduğunda 10^o olması durumundakine göre daha hızlı tutarlılık değerine yakınsamaktadır.

Tablo 3.5: İki boyutta dönüşümlerin NES grafik tablosu.

Yöntem	NES grafiği
CM-L	
CM-U	
CM-MU	
CM-DU	

Tablo 3.6: İki boyutta dönüşümlerin x ve y eksenleri için NES grafik tablosu.

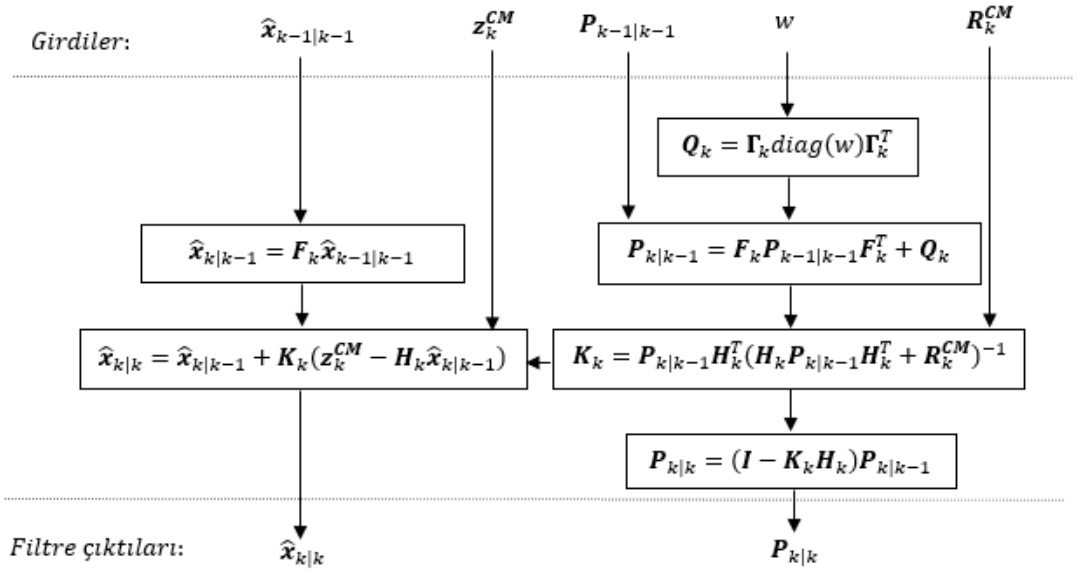
Yöntem	NES-X grafiği	NES-Y grafiği
CM-L		
CM-U		
CM-MU		
CM-DU		

3.3.3. Kalman Filtresine Uyarlanması

Bu bölümde CM dönüşümlerinin Kalman filtresine uyarlanması ve uyarlanması sonucunda elde edilen CMKF (Converted Measurement Kalman Filter) filtrelerin performansları üzerinde durulmaktadır.

Şekil 3.1'de CMKF için akış diyagramı verilmiştir. Şekil 2.7'deki standart Kalman filtresi ile benzerlik göstermektedir. Şekil 3.1'de gösterilen $\boldsymbol{z}_k^{CM}$ iki boyutlu sistem için Bölüm 3.2'de verilen ve CM çeşitlerine göre değişen $\boldsymbol{z}^c$ ölçüm vektörünü ve $\boldsymbol{R}_k^{CM}$ ise yine CM çeşitlerine göre değişen $\boldsymbol{R}$ kovaryans matrisini temsil etmektedir. Böylece, kullanılan CM'ye göre CMKF-L, CMKF-D, CMKF-U, CMKF-MU ve CMKF-DU gibi filtreler elde edilir.

Şekil 3.1'de gösterildiği gibi önceki tahmini temsil eden $\hat{\boldsymbol{x}}_{k-1|k-1}$, kovaryans matrisi $\boldsymbol{P}_{k-1|k-1}$, dönüştürülmüş ölçüm vektörü $\boldsymbol{z}_k^{\boldsymbol{CM}}$, dönüştürülmüş ölçüm için elde edilen ölçüm kovaryans matrisi $\boldsymbol{R}_k^{CM}$ ve süreç gürültüsünü temsil eden w filtre girdisidir. Ayrıca Bölüm 2.2.1'de belirtilen $\boldsymbol{F}_k$, $\boldsymbol{\Gamma}_k$ ve $\boldsymbol{H}_k$ matrisleri de filtreye girdi olarak verilmektedir.



Şekil 3.1: CMKF akış diyagramı.

Çok kullanılan CM-U, CM-MU ve CM-DU dönüşümlerinin Kalman filtresine uyarlanması sonucunda elde edilen CMKF-U, CMKF-MU ve CMKF-DU filtrelerinin

performans karşılaştırması için kullanılan senaryo bilgileri Tablo 3.7'de verilmektedir. Tabloda görüldüğü üzere test senaryoları hedefin radara olan mesafesinin yakın ve uzak olması; ölçüm gürültüsünün az ve çok olması durumlarını içermektedir. Senaryolarının testinde kullanılan Monte Carlo için 1500, kestirimin çabuk yakınsaması için ise takipteki $\boldsymbol{P}_{k-1|k-1}$ başlangıç kovaryans matris değeri her iki senaryo içinde diag([1000 1000 10 200]) olarak kullanılmıştır.

Tablo 3.7: İki boyutlu sistem için test senaryo bilgileri.

Hedef Senaryo Bilgileri	Senaryo-1 (radara yakın)	Senaryo-2 (radara uzak)
Hedef başlangıç konumu [x, y] (km)	[50, -300]	[200, -300]
Hedef hızı (km/saat)	900	900
Ölçüm gürültüsü standart sapması $[\sigma_r, \sigma_\theta]$ (m, derece)	(500, 1)	(1000, 1)
Sensör için süreç gürültüsü standart sapması $[q_x, q_y]$ (m, m)	[100, 100]	[500, 500]

Performans karşılaştırması için bu tezde yaygın olarak kullanılan karesel ortalama hata (RMSE, *ing.* Root Mean Square Error) ve Ortalama Normalize Tahmin Hata Karesi (ANEES, *ing.* Average Normalized Estimation Error Squared) testleri yapılmıştır. Her i anındaki örnek için elde edilen RMSE değeri Denklem (3.38)'de, ANEES değeri ise Denklem (3.45)'te verilmektedir. RMSE denklemindeki N Monte Carlo sayısını ve $X_i - \hat{X}_i$ i'inci örnekteki tahmin hatasını belirtmektedir.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - \hat{X}_i)^T (X_i - \hat{X}_i)} \tag{3.38}$$

Ölçümlerin küresel koordinatta alınması ve CM ile kartezyen koordinata dönüştürülmesinde oluşan en küçük ortalama karesel hata (Minimum Mean Square Error) aşağıdaki denklemlerle ifade edilmektedir [37], [42].

$$\begin{bmatrix} x_m^{MMSE} \\ y_m^{MMSE} \end{bmatrix} = \eta \begin{bmatrix} r_m \cos\theta_m \\ r_m \sin\theta_m \end{bmatrix} \tag{3.39}$$

r_m ve θ_m için Denklem (3.1) ve Denklem (3.2)'deki formları Denklem (3.39)'da yerine yazıldığında denklemin beklenen karesel hatası aşağıdaki gibi olmaktadır.

$$\begin{aligned} &E\left[\left(\eta(r+\tilde{r})\cos\left(\theta+\tilde{\theta}\right) - r\cos\theta\right)^2\right] \\ &+E\left[\left(\eta(r+\tilde{r})\sin\left(\theta+\tilde{\theta}\right) - r\sin\theta\right)^2\right] \\ &= \eta^2(r+{\sigma_r}^2) - 2\eta r^2 e^{-\sigma_\theta^2/2} + r^2 \end{aligned} \tag{3.40}$$

Denklem sıfıra eşitlenerek minimize edildiğinde η Denklem (3.41)'deki gibi bulunmaktadır.

$$\eta = \frac{r^2}{r^2+{\sigma_r}^2} e^{-\sigma_\theta^2/2} \tag{3.41}$$

η değeri Denklem (3.39)'da yerine konulduğunda denklemin yeni hali aşağıdaki gibi olmaktadır.

$$\begin{bmatrix} x_m^{MMSE} \\ y_m^{MMSE} \end{bmatrix} = \frac{r^2}{r^2+{\sigma_r}^2} e^{-\sigma_\theta^2/2} \begin{bmatrix} r_m \cos\theta_m \\ r_m \sin\theta_m \end{bmatrix} \tag{3.42}$$

Açıkça görüldüğü üzere elde edilen yeni form Denklem (3.21)'de verilmiş olan CM-MU dönüşümünün $\frac{r^2}{r^2+{\sigma_r}^2}$ ile ölçeklenmiş halidir. Bu nedenle CM-MU dönüşümü genelde CM-U dönüşümünden daha küçük bir MSE hatasına sahiptir [37],[42].

Kullanılan filtreler Kalman filtresine dayanmaktadır ve Kalman filtresi ölçümlerin yansız olduğu varsayımı altında çalışmaktadır. Dolayısıyla her ne kadar CM-MU filtresi genelde daha küçük MSE değerine sahip olsa da dönüşümün yansız olması için CM-U tercih edilebilir. Bu nedenle Bölüm 3.2.5'te de belirtildiği üzere bu

soruna çözüm olarak CM-DU filtresi ileri sürülmüştür. CM-DU filtresi CM-U dönüşümünü kullandığı için yansız bir dönüşüm sağlar. Bunun yanında CMKF-DU filtresinin MSE performansının arttırılmasına yönelik [41]'de bir yöntem önerilmektedir. [41]'e göre sistem için dönüşümün yansız olması küçük MSE performansından daha önemliyse Bölüm 3.2.5'te belirtilen CM-DU dönüşümü Şekil 3.1'de belirtildiği şekilde kullanılır. Sistem için dönüşümün yansızlığı önemsenmiyor ve MSE performansının CM-MU gibi düşük olması isteniyorsa Denklem (3.43) ve Denklem (3.44)'te belirtilen form sadece filtrenin çıkışına uygulanarak düşük MSE elde edilebilir. Dolayısıyla Denklem (3.43) ve (3.44)'ün kullanılıp kullanılmayacağı uygulamaya göre değişiklik göstermektedir [41]. Tahmin vektörü $\hat{x} = [x_t \; y_t \; \dot{x}_t \; \dot{y}_t]^T$ olarak tanımlanmaktadır ve model olarak sabit hız modeli kullanılmıştır. θ_t ifadesinin ne anlama geldiğine Bölüm 3.2.5'te değinilmiştir.

$$\hat{x}^o_{k|k} = e^{-\sigma^2_{\theta_t}} \hat{x}_{k|k} \tag{3.43}$$

$$P^o_{k|k} = e^{-2\sigma^2_{\theta_t}} P_{k|k} \tag{3.44}$$

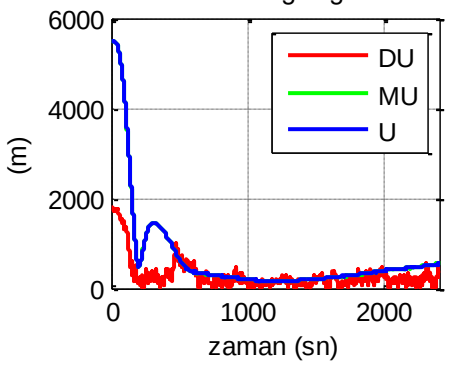
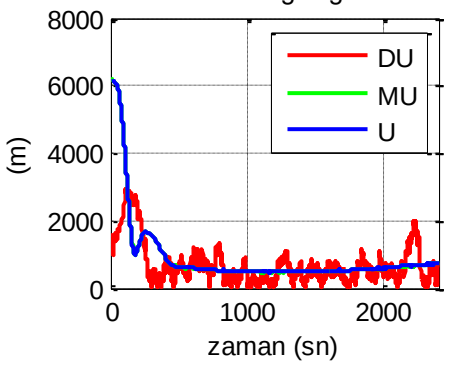
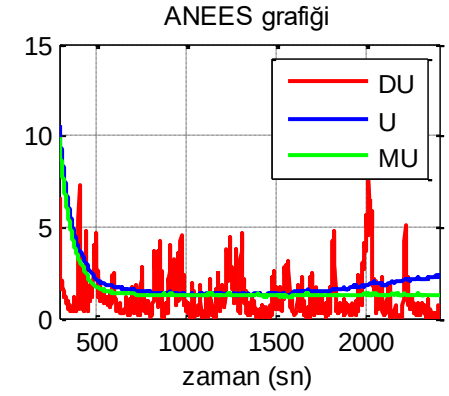
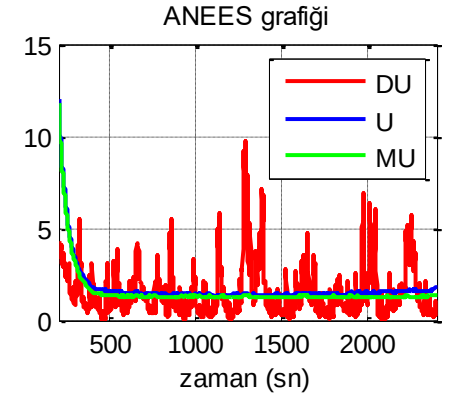
Performans karşılaştırması için kullanılan ANEES denklemi (3.45)'te verilmektedir. Denklemde yer alan N, Monte Carlo sayısını, n durum vektörünün boyutunu, $X_i - \hat{X}_i$ i'inci örnekteki tahmin hatasını ve P_i ise i'inci örnekteki hata kovaryans matrisini belirtmektedir. Tutarlı bir sonuç elde edilmesi için ANEES değerinin 1'e yakınsaması beklenmektedir.

$$ANEES = \frac{1}{Nn} \sum_{i=1}^{N} (X_i - \hat{X}_i)^T {P_i}^{-1} (X_i - \hat{X}_i) \tag{3.45}$$

Tablo 3.7'de belirtilen senaryolar için elde edilen RMSE ve ANEES performans grafikleri Tablo 3.8'de verilmektedir. Tablodan görüldüğü üzere CMKF-U ve CMKF-MU dönüşümleri hem RMSE hem de ANEES değerleri birbirine yakın sonuçlar vermiştir. Bunun yanında CMKF-DU dönüşümü bu veri için RMSE ve ANEES değerleri açısından diğer iki yönteme göre daha kararsız bir tutum sergilemiştir. ANEES değeri açısından CMKF-DU uç noktalar ihmal edildiğinde 1 değerine yakın

yerde salınım yaptığı söylenebilir. CMKF-U ve CMKF-MU ise CMKF-DU dönüşümüne göre daha kararlı fakat ANEES değeri 1 değerini aşmıştır. Bunun yanında RMSE değerleri açısından CMKF-U ve CMKF-MU çok yakın sonuçlar üretmiştir. Fakat CMKF-U algoritması için ANEES değeri CMKF-MU algoritmasından daha fazla ıraksamıştır. Bu nedenle iki boyutlu sistemde CMKF-MU algoritmasının tercih edilmesiyle daha tutarlı ve hatası az olan bir takip performansı elde edilebilir.

Tablo 3.8: İki boyutlu sistem için MSE ve ANEES performans grafikleri.

	Senaryo-1	Senaryo-2
RMSE		
ANEES		

3.4. Üç Boyutta CM Algoritmaları

Bu bölümde, Şekil 2.2'de belirtilen üç boyutlu sistem ele alınmaktadır. Bu durumda radardan alınan ölçümler (r_m, θ_m, φ_m) Denklem (3.46)-(3.48)'de belirtildiği gibi olmaktadır. Buradaki r gerçek menzil, θ yanal açı ve φ düşey açı değerini; $\tilde{r}$, $\tilde{\theta}$

ve $\tilde{\varphi}$ ise ortalaması 0, varyansları sırasıyla σ_r^2, σ_θ^2 ve σ_φ^2 olan Gauss ölçüm gürültülerini temsil etmektedir.

$$r_m = r + \tilde{r} \tag{3.46}$$

$$\theta_m = \theta + \tilde{\theta} \tag{3.47}$$

$$\varphi_m = \varphi + \tilde{\varphi} \tag{3.48}$$

CM algoritması ile elde edilen kartezyen koordinata dönüştürülmüş ölçüm vektörü $\boldsymbol{z}^c$ Denklem (3.49)'da, kullanılan kovaryans matrisi $\boldsymbol{R}$ ise Denklem (3.50)'de belirtildiği gibidir [33].

$$\boldsymbol{z}^c = \begin{bmatrix} x_m^c \\ y_m^c \\ z_m^c \end{bmatrix} = \begin{bmatrix} r_m cos\theta_m cos\varphi_m \\ r_m sin\theta_m cos\varphi_m \\ r_m sin\varphi_m \end{bmatrix} - \boldsymbol{\mu} \tag{3.49}$$

$$\boldsymbol{R} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{12} & R_{22} & R_{23} \\ R_{13} & R_{23} & R_{33} \end{bmatrix} \tag{3.50}$$

Denklem (3.49)'daki dönüşüm ve $\boldsymbol{R}$ kovaryans matrisi, farklı CM algoritmaları için sonraki bölümlerde verilmektedir.

3.4.1. CM_U

Yansız dönüşüm için üç boyutta kullanılan denklemler aşağıda verilmektedir ve kompanzasyon faktörü olarak bilinen $\lambda_\theta = e^{-\sigma_\theta^2/2}$ ve $\lambda_\varphi = e^{-\sigma_\varphi^2/2}$'dir [34].

$$\begin{aligned} \begin{bmatrix} x_m^{u,3D} \\ y_m^{u,3D} \\ z_m^{u,3D} \end{bmatrix} &= \begin{bmatrix} e^{\sigma_\theta^2/2}\, e^{\sigma_\varphi^2/2}\, r_m cos\theta_m cos\varphi_m \\ e^{\sigma_\theta^2/2}\, e^{\sigma_\varphi^2/2}\, r_m sin\theta_m cos\varphi_m \\ e^{\sigma_\varphi^2/2}\, r_m sin\varphi_m \end{bmatrix} \\ &= \begin{bmatrix} \lambda_\theta^{-1}\, \lambda_\varphi^{-1}\, r_m cos\theta_m cos\varphi_m \\ \lambda_\theta^{-1}\, \lambda_\varphi^{-1}\, r_m sin\theta_m cos\varphi_m \\ \lambda_\varphi^{-1}\, r_m sin\varphi_m \end{bmatrix} \end{aligned} \tag{3.51}$$

$$R_{u,3D}^{11} = \left(\left(\lambda_\theta \lambda_\varphi\right)^{-2} - 2\right) {r_m}^2 cos^2\theta_m cos^2\varphi_m + \frac{1}{4}({r_m}^2 + {\sigma_r}^2) \\ [1 + \lambda'_\theta \cos(2\theta_m)]\left[1 + \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.52}$$

$$R_{u,3D}^{22} = \left(\left(\lambda_\theta \lambda_\varphi\right)^{-2} - 2\right) {r_m}^2 sin^2\theta_m cos^2\varphi_m + \frac{1}{4}({r_m}^2 + {\sigma_r}^2) \\ [1 - \lambda'_\theta \cos(2\theta_m)]\left[1 + \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.53}$$

$$R_{u,3D}^{33} = \left({\lambda_\varphi}^{-2} - 2\right) {r_m}^2 sin^2\varphi_m + \frac{1}{2}({r_m}^2 + {\sigma_r}^2)\left[1 - \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.54}$$

$$R_{u,3D}^{12} = \left(\left(\lambda_\theta \lambda_\varphi\right)^{-2} - 2\right) {r_m}^2 sin\theta_m cos\theta_m cos^2\varphi_m + \frac{1}{4}({r_m}^2 + {\sigma_r}^2) \\ \lambda'_\theta \, sin(2\theta_m)\left[1 + \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.55}$$

$$R_{u,3D}^{13} = \left(\lambda_\theta^{-1} \lambda_\varphi^{-2} - \lambda_\theta^{-1} - \lambda_\theta\right) {r_m}^2 cos\theta_m sin\varphi_m cos\varphi_m \\ + \frac{1}{2}({r_m}^2 + {\sigma_r}^2)\lambda_\theta \, \lambda'_\varphi cos\theta_m sin(2\varphi_m) \tag{3.56}$$

$$R_{u,3D}^{23} = \left(\lambda_\theta^{-1} \lambda_\varphi^{-2} - \lambda_\theta^{-1} - \lambda_\theta\right) {r_m}^2 sin\theta_m sin\varphi_m cos\varphi_m \\ + \frac{1}{2}({r_m}^2 + {\sigma_r}^2)\lambda_\theta \, \lambda'_\varphi sin\theta_m sin(2\varphi_m) \tag{3.57}$$

3.4.2. CM_MU

Modifiye yansız dönüşüm eşitlikleri Denklem (3.58)-(3.64)'te verilmiştir ve bu denklemlerde yer alan λ_θ ve λ_φ kompanzasyon faktörleri bir önceki bölümde bahsedilen kompanzasyon değerleri ile aynıdır [35].

$$\begin{bmatrix} x_m^{mu,3D} \\ y_m^{mu,3D} \\ z_m^{mu,3D} \end{bmatrix} = \begin{bmatrix} e^{-\sigma_\theta^2/2}\, e^{-\sigma_\varphi^2/2} r_m cos\theta_m cos\varphi_m \\ e^{-\sigma_\theta^2/2}\, e^{-\sigma_\varphi^2/2} r_m sin\theta_m cos\varphi_m \\ e^{-\sigma_\varphi^2/2} r_m sin\varphi_m \end{bmatrix} = \begin{bmatrix} \lambda_\theta\, \lambda_\varphi r_m cos\theta_m cos\varphi_m \\ \lambda_\theta\, \lambda_\varphi r_m sin\theta_m cos\varphi_m \\ \lambda_\varphi r_m sin\varphi_m \end{bmatrix} \tag{3.58}$$

$$R_{mu,3D}^{11} = -{\lambda_\theta}^2 {\lambda_\varphi}^2 {r_m}^2 cos^2\theta_m cos^2\varphi_m + \frac{1}{4}({r_m}^2 + {\sigma_r}^2) [1 + \lambda'_\theta \cos(2\theta_m)]\left[1 + \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.59}$$

$$R_{mu,3D}^{22} = -{\lambda_\theta}^2 {\lambda_\varphi}^2 {r_m}^2 sin^2\theta_m cos^2\varphi_m + \frac{1}{4}({r_m}^2 + {\sigma_r}^2) [1 - \lambda'_\theta \cos(2\theta_m)]\left[1 + \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.60}$$

$$R_{mu,3D}^{33} = -{\lambda_\varphi}^2 {r_m}^2 sin^2\varphi_m + \frac{1}{2}({r_m}^2 + {\sigma_r}^2)\left[1 - \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.61}$$

$$R_{mu,3D}^{12} = -{\lambda_\theta}^2 {\lambda_\varphi}^2 {r_m}^2 cos\theta_m sin\theta_m cos^2\varphi_m + \frac{1}{4}({r_m}^2 + {\sigma_r}^2) \lambda'_\theta \sin(2\theta_m)\left[1 + \lambda'_\varphi \cos(2\varphi_m)\right] \tag{3.62}$$

$$R_{mu,3D}^{13} = -\lambda_\theta\, {\lambda_\varphi}^2 {r_m}^2 cos\theta_m sin\varphi_m cos\varphi_m + \frac{1}{2}({r_m}^2 + {\sigma_r}^2)\lambda_\theta\, \lambda'_\varphi\, cos\theta_m \sin(2\varphi_m) \tag{3.63}$$

$$R_{mu,3D}^{23} = -\lambda_\theta\, {\lambda_\varphi}^2 {r_m}^2 sin\theta_m sin\varphi_m cos\varphi_m + \frac{1}{2}({r_m}^2 + {\sigma_r}^2)\lambda_\theta\, \lambda'_\varphi\, sin\theta_m \sin(2\varphi_m) \tag{3.64}$$

3.4.3. CM_DU

Üç boyut için ilişkisiz ve yansız dönüşüm Bölüm 3.2.5'te anlatılmakta olan iki boyuta mevcut olan dönüşüm mantığının üç boyuta genişletilmiş halidir.

Kullanılmakta olan dönüşüm denklemleri ve kovaryans matrisi Denklem (3.66)-(3.71)'de verilmektedir [37].

$$\begin{bmatrix} x_m^{du,3D} \\ y_m^{du,3D} \\ z_m^{du,3D} \end{bmatrix} = \begin{bmatrix} x_m^{u,3D} \\ y_m^{u,3D} \\ z_m^{u,3D} \end{bmatrix} = \begin{bmatrix} e^{\sigma_\theta^2/2}\, e^{\sigma_\varphi^2/2}\, r_m cos\theta_m cos\varphi_m \\ e^{\sigma_\theta^2/2}\, e^{\sigma_\varphi^2/2}\, r_m sin\theta_m cos\varphi_m \\ e^{\sigma_\varphi^2/2}\, r_m sin\varphi_m \end{bmatrix} \tag{3.65}$$

$$\begin{aligned} R_{du,3D}^{11} = \frac{1}{4}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[1 + \cos(2\theta_t)e^{-2\sigma_\theta^2}\, e^{-2\sigma_{\theta_t}^2}\right] e^{\sigma_\theta^2} \\ \left[1 + \cos(2\varphi_t)e^{-2\sigma_\varphi^2}\, e^{-2\sigma_{\varphi_t}^2}\right] e^{\sigma_\varphi^2} - \frac{1}{4}\left({r_t}^2 + {\sigma_{r_t}}^2\right) \\ \left[1 + \cos(2\theta_t)\, e^{-2\sigma_{\theta_t}^2}\right]\left[1 + \cos(2\varphi_t)\, e^{-2\sigma_{\varphi_t}^2}\right] \end{aligned} \tag{3.66}$$

$$\begin{aligned} R_{du,3D}^{22} = \frac{1}{4}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[1 - \cos(2\theta_t)e^{-2\sigma_\theta^2}\, e^{-2\sigma_{\theta_t}^2}\right] e^{\sigma_\theta^2} \\ \left[1 + \cos(2\varphi_t)e^{-2\sigma_\varphi^2}\, e^{-2\sigma_{\varphi_t}^2}\right] e^{\sigma_\varphi^2} - \frac{1}{4}\left({r_t}^2 + {\sigma_{r_t}}^2\right) \\ \left[1 - \cos(2\theta_t)\, e^{-2\sigma_{\theta_t}^2}\right]\left[1 + \cos(2\varphi_t)\, e^{-2\sigma_{\varphi_t}^2}\right] \end{aligned} \tag{3.67}$$

$$\begin{aligned} R_{du,3D}^{33} = \frac{1}{2}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[1 - \cos(2\varphi_t)e^{-2\sigma_\varphi^2}\, e^{-2\sigma_{\varphi_t}^2}\right] \\ e^{\sigma_\varphi^2} - \frac{1}{2}\left({r_t}^2 + {\sigma_{r_t}}^2\right)\left[1 - \cos(2\varphi_t)\, e^{-2\sigma_{\varphi_t}^2}\right] \end{aligned} \tag{3.68}$$

$$\begin{aligned} R_{du,3D}^{12} = \frac{1}{4}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)\left[\sin(2\theta_t)e^{-2\sigma_\theta^2}\, e^{-2\sigma_{\theta_t}^2}\right] e^{\sigma_\theta^2} \\ \left[1 + \cos(2\varphi_t)e^{-2\sigma_\varphi^2}\, e^{-2\sigma_{\varphi_t}^2}\right] e^{\sigma_\varphi^2} - \frac{1}{4}\left({r_t}^2 + {\sigma_{r_t}}^2\right) \\ \left[\sin(2\theta_t)\, e^{-2\sigma_{\theta_t}^2}\right]\left[1 + \cos(2\varphi_t)\, e^{-2\sigma_{\varphi_t}^2}\right] \end{aligned} \tag{3.69}$$

$$\begin{aligned} R_{du,3D}^{13} = \frac{1}{2}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right)cos\theta_t e^{-\sigma_\theta^2/2}\left[\sin(2\varphi_t)e^{-2\sigma_\varphi^2}\, e^{-2\sigma_{\varphi_t}^2}\right] \\ e^{\sigma_\varphi^2} - \frac{1}{2}\left({r_t}^2 + {\sigma_{r_t}}^2\right)cos\theta_t e^{-\sigma_\theta^2/2}\left[\sin(2\varphi_t)\, e^{-2\sigma_{\varphi_t}^2}\right] \end{aligned} \tag{3.70}$$

$$R_{du,3D}^{23} = \frac{1}{2}\left({r_t}^2 + {\sigma_r}^2 + {\sigma_{r_t}}^2\right) sin\theta_t e^{-\sigma_\theta^2/2}\left[\sin(2\varphi_t) e^{-2\sigma_\varphi^2} e^{-2\sigma_{\varphi t}^2}\right] e^{\sigma_\varphi^2} - \frac{1}{2}\left({r_t}^2 + {\sigma_{r_t}}^2\right) sin\theta_t e^{-\sigma_\theta^2/2}\left[\sin(2\varphi_t)\, e^{-2\sigma_{\varphi t}^2}\right] \tag{3.71}$$

Denklemlerde ölçüm değerleri yerine kullanılan r_t, θ_t ve φ_t menzil, yanal ve düşey açılarını temsil etmektedir ve bu değerler dönüşüm sonucunda tahmin edilen x_t, y_t ve z_t değerleri kullanılarak hesaplanmaktadır. θ_t ve ${\sigma_{\theta_t}}^2$ sırasıyla Denklem (3.30) ve Denklem (3.32)'de belirtildiği gibi iken; r_t, ${\sigma_{r_t}}^2$, φ_t ve ${\sigma_{\varphi_t}}^2$ Denklem (3.72)-(3.76)'da verilmektedir [37], [41].

$$r_t = \sqrt{{x_t}^2 + {y_t}^2 + {z_t}^2} \tag{3.72}$$

$${\sigma_{r_t}}^2 = \begin{bmatrix}\frac{\partial r_t}{\partial x_t} & \frac{\partial r_t}{\partial y_t} & \frac{\partial r_t}{\partial z_t}\end{bmatrix} \boldsymbol{P} \begin{bmatrix}\frac{\partial r_t}{\partial x_t} \\ \frac{\partial r_t}{\partial y_t} \\ \frac{\partial r_t}{\partial z_t}\end{bmatrix} = \frac{1}{{r_t}^2}\begin{bmatrix}x_t & y_t & z_t\end{bmatrix} \boldsymbol{P} \begin{bmatrix}x_t \\ y_t \\ z_t\end{bmatrix} \tag{3.73}$$

$$\varphi_t = \tan^{-1}\left(\frac{z_t}{\sqrt{{x_t}^2 + {y_t}^2}}\right) \tag{3.74}$$

$${\sigma_{\varphi_t}}^2 = \begin{bmatrix}\frac{\partial \varphi_t}{\partial x_t} & \frac{\partial \varphi_t}{\partial y_t} & \frac{\partial \varphi_t}{\partial z_t}\end{bmatrix} \boldsymbol{P} \begin{bmatrix}\frac{\partial \varphi_t}{\partial x_t} \\ \frac{\partial \varphi_t}{\partial y_t} \\ \frac{\partial \varphi_t}{\partial z_t}\end{bmatrix} = \frac{1}{({x_t}^2 + {y_t}^2){r_t}^4}\begin{bmatrix}-x_t z_t & -y_t z_t & {x_t}^2 + {y_t}^2\end{bmatrix} \boldsymbol{P} \begin{bmatrix}-x_t z_t \\ -y_t z_t \\ {x_t}^2 + {y_t}^2\end{bmatrix} \tag{3.75}$$

$$\boldsymbol{P} = \begin{bmatrix} P_{xx} & P_{xy} & P_{xz} \\ P_{xy} & P_{yy} & P_{yz} \\ P_{xz} & P_{yz} & P_{zz} \end{bmatrix} \tag{3.76}$$

3.5. Üç Boyutlu CM için Performans Karşılaştırmaları

Bu bölümde Şekil 2.2'de gösterilmekte olan üç boyutlu sistem için CM çeşitlerinin farklı kriterler de karşılaştırmaları mevcuttur. Kullanılan kriterler Bölüm 3.3'te kullanılan kriterler ile aynıdır. Bu nedenle kriter detayları için Bölüm 3.3'e bakılabilir.

3.5.1. Yansızlık

Tablo 3.9'da üç boyut için CM çeşitlerinin dönüşüm yanlılık çarpanı verilmektedir ve çarpanın 1 olması dönüşümün yansız olduğunu göstermektedir [41]. Tablodan da görüldüğü üzere CM-U ve CM-DU dönüşümleri yansız dönüşümlerdir. Tabloda verilmekte olan yanlılık çarpanı değerleri Bölüm 3.3.1'de iki boyut için yapılan işlemlerin üç boyuta genişletilmesi sonucunda bulunmaktadır.

Tablo 3.9: Üç boyutta CM yöntemlerinin dönüşüm yanlılık çarpanı.

Dönüşüm Yöntemi	Yanlılık çarpanı (X ve Y ekseni)	Yanlılık çarpanı (Z ekseni)
CM-L	$e^{-\sigma_\theta^2/2} e^{-\sigma_\varphi^2/2}$	$e^{-\sigma_\theta^2/2}$
CM-U	1	1
CM-MU	$e^{-\sigma_\theta^2} e^{-\sigma_\varphi^2}$	$e^{-\sigma_\theta^2}$
CM-DU	1	1

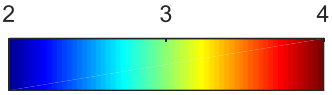
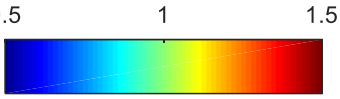
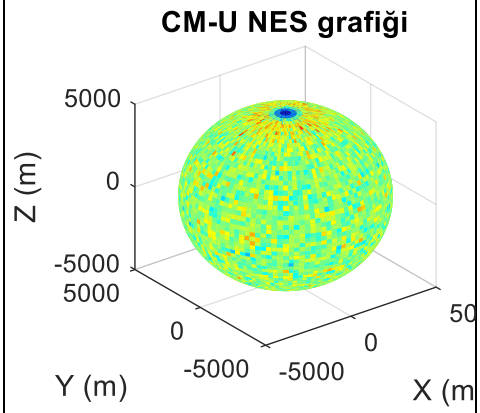
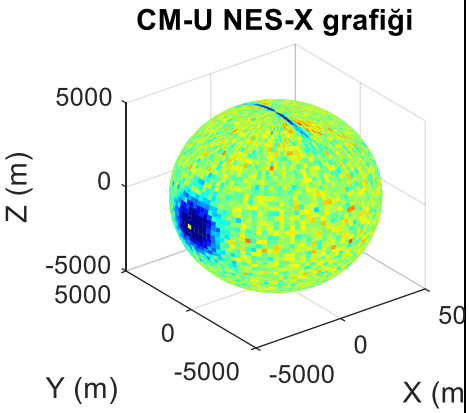
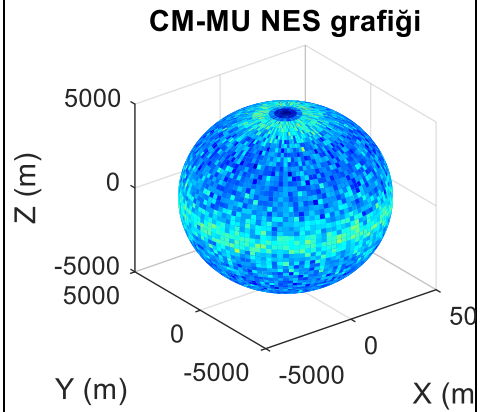
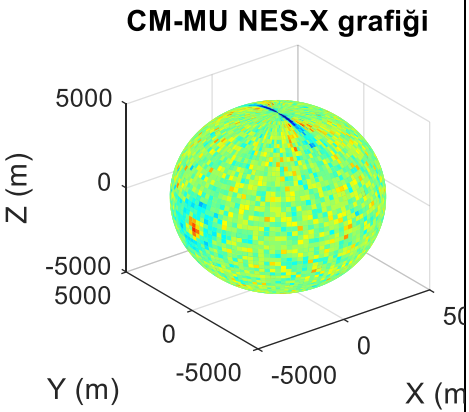
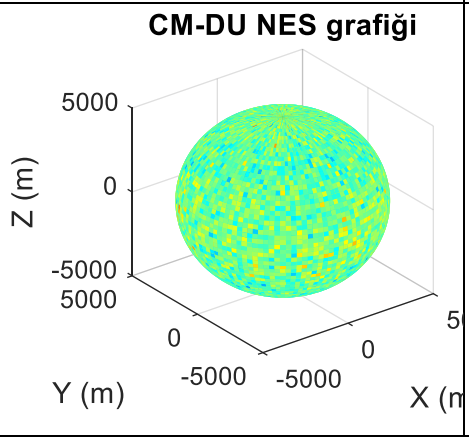
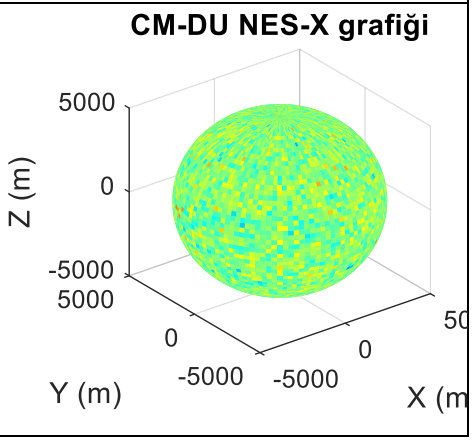
3.5.2. Tutarlılık

Tutarlılık ölçütü için kullanılan NES kriterinin eşitliği Denklem (3.37)'de verilmiştir. Bölüm 3.3.2'de de belirtildiği üzere NES değerinin dönüşüm uygulanan sistemin boyutuna eşit olması dönüşümün tutarlı olduğu anlamına gelmektedir.

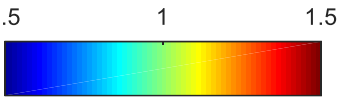
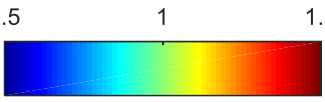
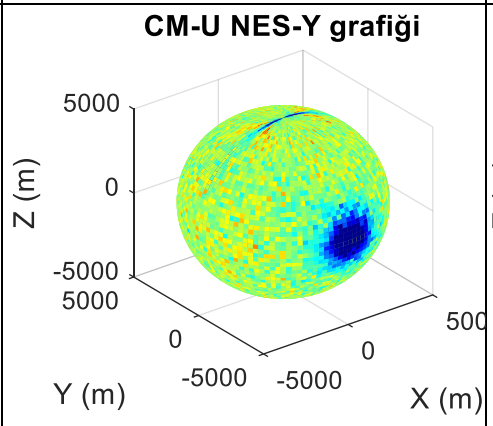
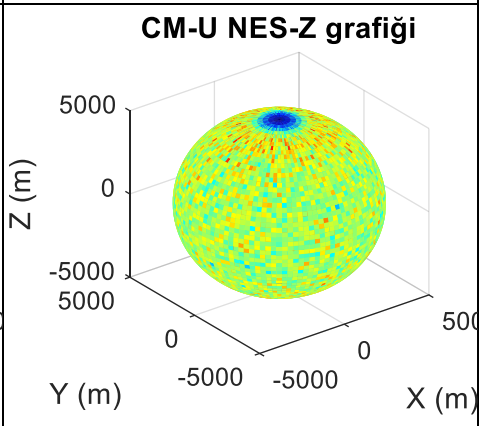
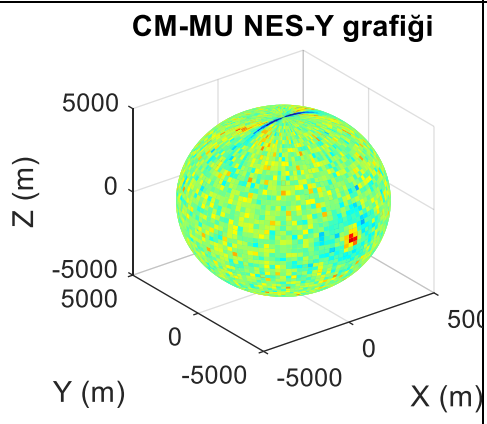
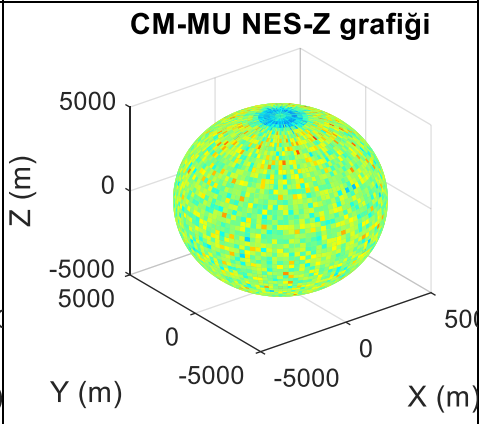
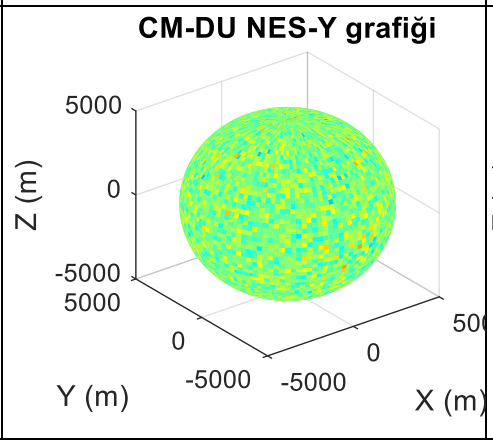
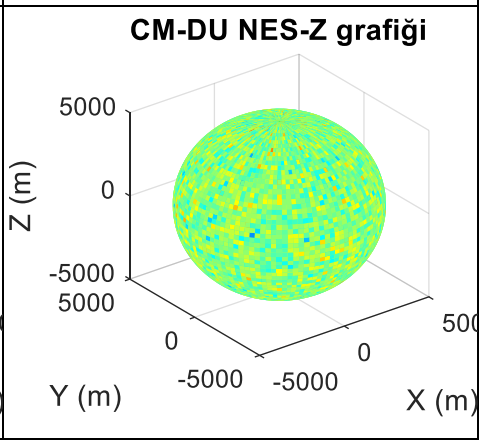
Üç boyutlu sistem için CM-U, CM-MU ve CM-DU algoritmalarının tutarlılık sonuçları Tablo 3.10 ve Tablo 3.11'de verilmiştir. Verilen tablolardan açıkça görüldüğü üzere toplam sistem için elde edilen NES değeri her üç algoritma içinde yaklaşık üç değerindedir. Fakat CM-U ve CM-MU dönüşümlerinde her yanal ve dikey açı değeri için aynı değere sahip değildir. Fakat CM-DU algoritmasına bakıldığında ise toplam sistemin her açı değerlerinde NES değeri yaklaşık aynı değerdedir. Bunun yanında X, Y ve Z eksenlerine ayrı ayrı bakıldığında ise NES değerinin 1 olması beklenmektedir. Çünkü her eksen için NES değeri boyutun değerine yani 1 değerine yakınsamalıdır. Bu açıdan CM-U ve CM-MU dönüşümlerinin NES değerleri yanal ve dikey açılara göre değişiklik gösterirken, CM-DU dönüşümünün NES değeri hemen hemen her noktada yaklaşık aynı değere sahiptir. Bu nedenle CM-DU dönüşümü CM-U ve CM-MU dönüşümlerine göre daha tutarlı bir dönüşümdür.

Tutarlılık testi için menzil mesafesi 5 km, yanal ve dikey açılar $0^o - 360^o$, ölçüm gürültüsü ilişkisiz ve σ_r= 50 m, $\sigma_\theta = 5^o$ ve $\sigma_\varphi = 5^o$ için senaryo 500 kez çalıştırılmıştır.

Tablo 3.10: Üç boyutta dönüşümlerin toplam ve x ekseni için NES grafik tablosu.

Yöntem	NES grafiği	NES-X grafiği
		
CM-U		
CM-MU		
CM-DU		

Tablo 3.11: Üç boyutta dönüşümlerin y ve z eksenleri için NES grafik tablosu.

Yöntem	NES-Y grafiği	NES-Z grafiği
		
CM-U		
CM-MU		
CM-DU		

3.5.3. Kalman Filtresine Uyarlanması

Bu bölümde üç boyutlu sistem için CMKF filtresinin oluşturulması ve filtre performansları üzerinde durulmaktadır. Üç boyutlu sistem için CMKF akış diyagramı Bölüm 3.3.3'teki iki boyutlu sistem için olan Şekil 3.1'de verilen CMKF akış diyagramı ile aynıdır. İki boyutlu sistem ile üç boyutlu arasındaki farklar ise kullanılan $\hat{\boldsymbol{x}}_{k-1|k-1}$ tahmin vektörü, $\boldsymbol{P}_{k-1|k-1}$ kovaryans matrisi, $\boldsymbol{F}_k$, $\boldsymbol{\Gamma}_k$ ve $\boldsymbol{H}_k$ matrislerinin yanısıra dönüşürülmüş ölçüm için elde edilen $\boldsymbol{z}_k^{CM}$ ve $\boldsymbol{R}_k^{CM}$ matrisleridir. Üç boyutlu sistem için kullanılan $\boldsymbol{z}_k^{CM}$ Bölüm 3.4'teki CM çeşitlerine göre değişen $\boldsymbol{z}^c$ ölçüm vektörü ve $\boldsymbol{R}_k^{CM}$ ise yine Bölüm 3.4'teki CM çeşitlerine göre değişen $\boldsymbol{R}$ kovaryans matrisini temsil etmektedir. Böylece kullanılan CM çeşitlerine göre üç boyutta CMKF-U, CMKF-MU ve CMKF-DU gibi filtreler elde edilir.

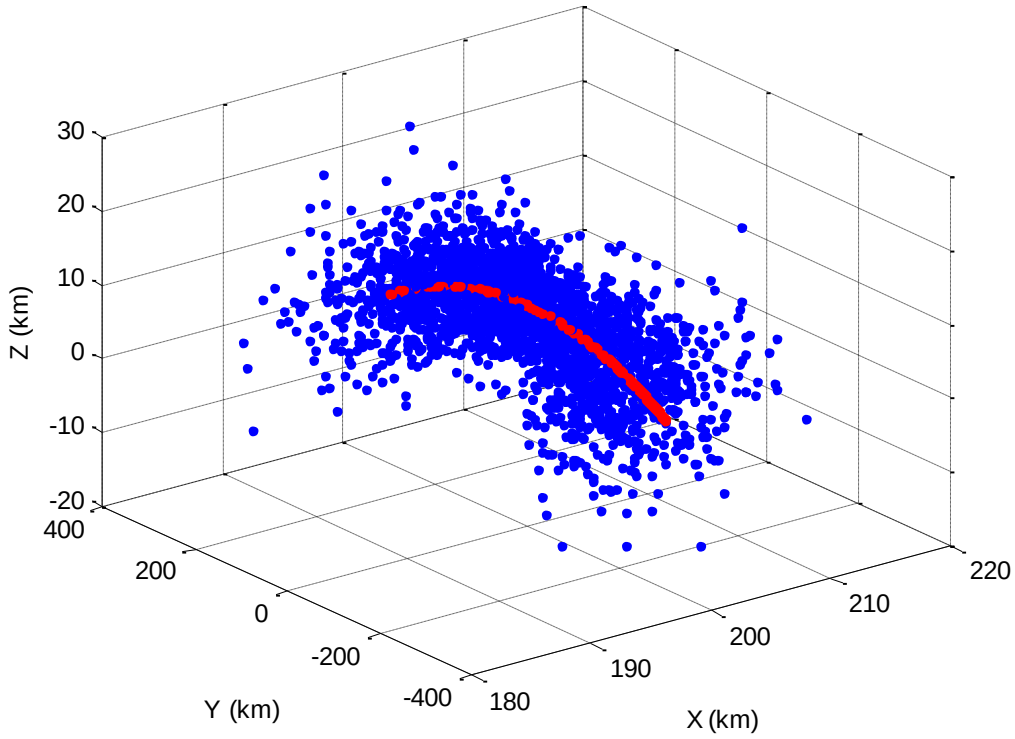
Bu bölümde CM-U, CM-MU ve CM-DU dönüşümlerinin Kalman filtresine uyarlanmasıyla elde edilen CMKF-U, CMKF-MU ve CMKF-DU filtrelerinin performans karşılaştırması için kullanılan hedef senaryo bilgileri Tablo 3.12'de, hedefin gerçek konumu ise Tablo 3.13'te verilmektedir. Monte Carlo için 1500, takip için *P* başlangıç kovaryans matrisi değeri algoritmaların daha çabuk yakınsaması için her iki senaryo içinde diag([1000 1000 1000 0 200 50]) olarak kullanılmıştır.

Tablo 3.12: Üç boyutlu sistem için test senaryo bilgileri.

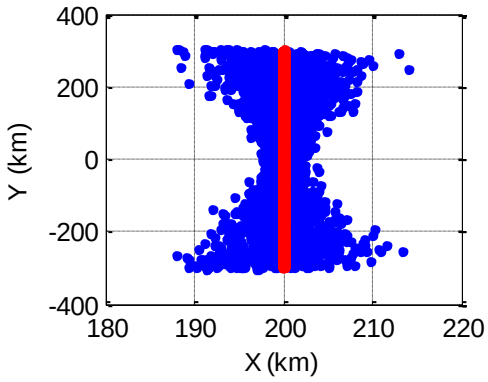
Hedef Senaryo Bilgileri	Senaryo-1	Senaryo-2
Hedef başlangıç konumu [x, y, z] (km)	[50, -300, 8]	[200, -300, 8]
Hedef hızı (km/saat)	920	920
Ölçüm gürültüsü standart sapması $[\sigma_r, \sigma_\theta, \sigma_\varphi]$ (m, derece, derece)	(500, 1, 1)	(1000, 1, 1)
Sensör için süreç gürültüsü standart sapması $[q_x, q_y, q_z]$ (m, m, m)	[100, 100, 100]	[500, 500, 500]
Takip ölçüm parametresi $[\sigma_r, \sigma_\theta, \sigma_\varphi]$ (m, derece, derece)	(250, 1, 1)	(500, 1, 1)

Tablo 3.13: Üç boyutlu sistem için hedef senaryo görüntüsü.

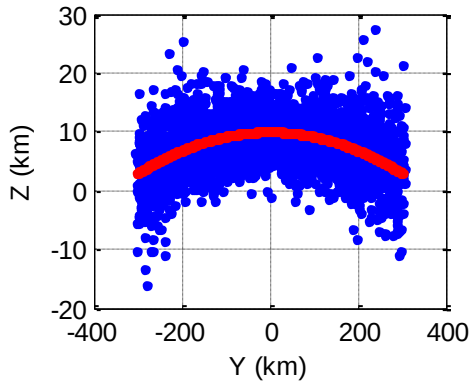
X-Y-Z ekseni



X-Y ekseni kuşbakışı



Y-Z ekseni kuşbakışı



Hedef senaryoları dünya şekli göz önünde bulundurularak üretilmiştir ve hedef radar sistemine önce yakınlaşıp daha sonra uzaklaşmaktadır. Tablo 3.13'teki mavi noktalar örnek ölçüm değerlerini, kırmızı eğri ise hedefin doğrultusunu temsil etmektedir. Y-Z ekseni kuşbakışı grafiğinde görüldüğü üzere hedef güzergahı dünyanın eğriliği de göz önünde bulundurularak üretilmiştir. Ayrıca radar [0,0,0] konumundadır. Bu durumda hedefin radardan uzaklaşması durumunda elde edilen ölçümlerdeki saçılmanın artması

grafiklerde de açıkça görülmektedir. Bunun temel nedeni ise açı değeri sapmasının uzak menzilde yakın menzildekine göre daha fazla mesafeye karşılık gelmesidir.

Bu bölümde performans karşılaştırması için Bölüm 3.3.3'teki gibi RMSE ve ANEES testleri yapılmıştır. Her k anındaki örnek için elde edilen RMSE değeri ve ANEES değeri Bölüm 3.3.3'te sırasıyla Denklem (3.38) ve Denklem (3.45)'te verilmişti.

RMSE sonuçları Tablo 3.14'te verilmektedir. Üç boyutlu sistem için CM-DU dönüşümün küçük MSE değerine sahip olması için Bölüm 3.3.3'te iki boyutlu sistemde olduğu gibi üç boyutlu sistemde de filtre çıkışı belli bir katsayı ile çarpılmaktadır. Bu katsayı üç boyutlu sistem için Denklem (3.77)-(3.81)'de belirtilmektedir. İki boyutu sistemde olduğu gibi üç boyutlu sistemde de Denklem (3.77) ve Denklem (3.78)'in kullanılmak istenip istenmemesi uygulamaya bağlı olarak değişebilmektedir [41]. Denklem (3.79)'da belirtilen $diag$ ifadesi diagonal (köşegen) matrisi temsil etmektedir. Tahmin vektörü $\hat{x} = [x_t\ y_t\ z_t\ \dot{x}_t\ \dot{y}_t\ \dot{z}_t]^T$ olarak tanımlanmaktadır ve model olarak sabit hız modeli kullanılmıştır. θ_t ve φ_t ifadesinin ne anlama geldiğine Bölüm 3.4.3'te değinilmiştir.

$$\hat{x}^o_{k|k} = \Lambda \hat{x}_{k|k} \tag{3.77}$$

$$P^o_{k|k} = \Lambda P_{k|k} \Lambda^T \tag{3.78}$$

$$\Lambda = diag(\Lambda_1, \Lambda_1, \Lambda_2, \Lambda_1, \Lambda_1, \Lambda_2) \tag{3.79}$$

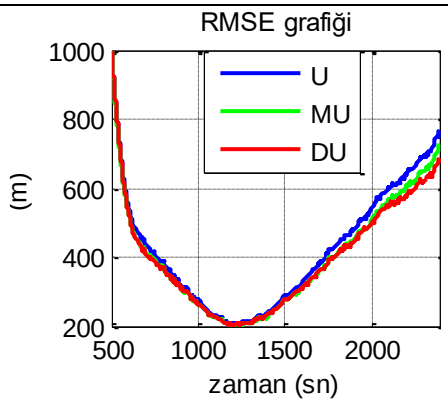
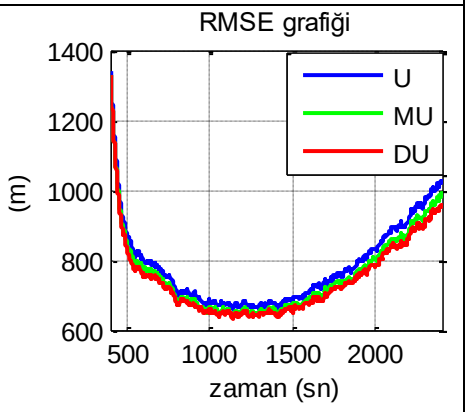
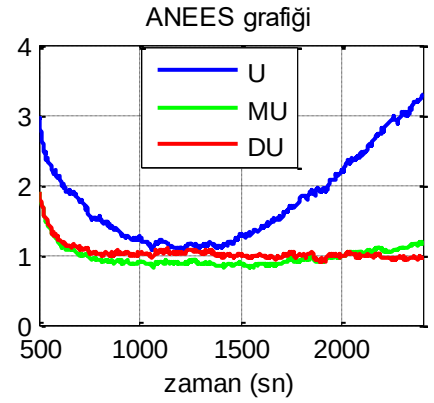
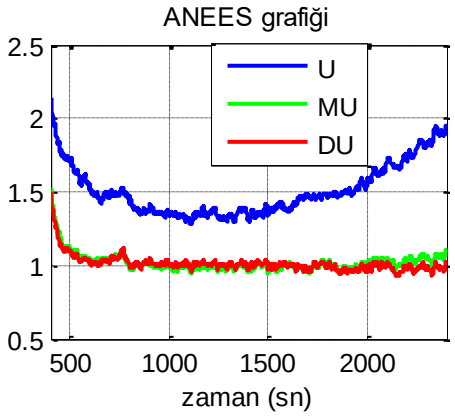
$$\Lambda_1 = e^{-\sigma^2_{\dot{\theta}_t}} e^{-\sigma^2_{\dot{\phi}_t}} \tag{3.80}$$

$$\Lambda_2 = e^{-\sigma^2_{\dot{\theta}_t}} \tag{3.81}$$

Tabloda grafiklerin daha iyi görünebilmesi için kestirimin yakınsadığı örnekten sonrası çizdirilmiştir. Tabloda da görüldüğü üzere CMKF-MU algoritması CMKF-U algoritmasından daha az RMSE değerine sahiptir. Denklem (3.42)'de CMKF-MU dönüşümünün CMKF-U dönüşümden genelde daha az MMSE hatasına sahip olduğuna değinilmişti. Grafiklerden de görüldüğü üzere CMKF-DU dönüşümü ise CMKF-MU dönüşümünden dolayısıyla CMKF-U dönüşümünden de daha az hata

değerine sahiptir. Bunun yanında dönüşümün tutarlılığını temsil eden ANEES grafiklerine bakıldığında ise CMKF-MU ve CMKF-DU algoritmalarının sonuçlarının her iki senaryo içinde yakın olduğu görülmektedir. Fakat değerlere daha hassas bakıldığında CMKF-DU algoritmasının 1'e daha yakın olduğu görülmektedir. Bu durumda CMKF-DU algoritması diğer iki algoritmadan daha tutarlı diyebiliriz. CMKF-U algoritmasının ise ıraksadığı açıkça görülmektedir.

Tablo 3.14: Üç boyutlu sistem için RMSE ve ANEES performans grafikleri.

	Senaryo-1	Senaryo-2
RMSE		
ANEES		

4. ETKİLEŞİMLİ ÇOKLU MODEL

Bu bölüm, Etkileşimli Çoklu Model algoritması hakkında genel bilgileri, Bölüm 2.2'de belirtilen hedef takip algoritmalarından Kalman filtresi, UKF ve CMKF algoritmalarının IMM ile kullanımını ve IMM-KF, IMM-UKF ve IMM-CMKF gibi hedef takip algoritmalarının performans karşılaştırılmasının yapıldığı Bölüm 5.4'te kullanılan IMM filtre durum modelleri hakkındaki bilgileri içermektedir. Ayrıca literatürde füzyon kriterlerine dayalı geliştirilen SIMM, DIMM ve MIMM algoritmalarının IMM algoritmasından farkına, CMKF algoritmasının bu algoritmalara uyarlanmasına ve elde edilen hibrit algoritmaların takip performanslarına değinilmektedir.

4.1. Genel Bilgi ve Kullanım Nedeni

Hedef takip algoritmaları hedefe ait gürültülü ölçümleri kullanarak hedefin durum kestirimini yapmaktadır. Hedef takibi için yaygın olarak kullanılan birçok algoritma mevcuttur ve bu algoritmalar uygulanırken hedef hareket modelinin doğru seçilmesi gerekmektedir. Hareket modelinin doğru seçilmemesi filtre performansını olumsuz etkilemektedir. Bu nedenle hareket modeli seçimi yapılırken olası hedef dinamikleri göz önünde bulundurulmaktadır. Sabit hız modeli (CV, *ing.* Constant Velocity), sabit ivme modeli (CA, *ing.* Constant Acceleration) ve koordineli dönüş modeli (CT, *ing.* Coordinated Turn) en yaygın kullanılan hareket modellerinden bazılarıdır. Bu modellerden CV modeli manevra yapmayan ve yaklaşık olarak sabit hızlı hareket eden hedefler için uygunken, CA modeli sabit ivmeli hareket eden hedefler için uygundur. Bunun yanında koordineli dönüş modeli ise manevra yapan hedeflerin hareketini modellemek için kullanılmaktadır.

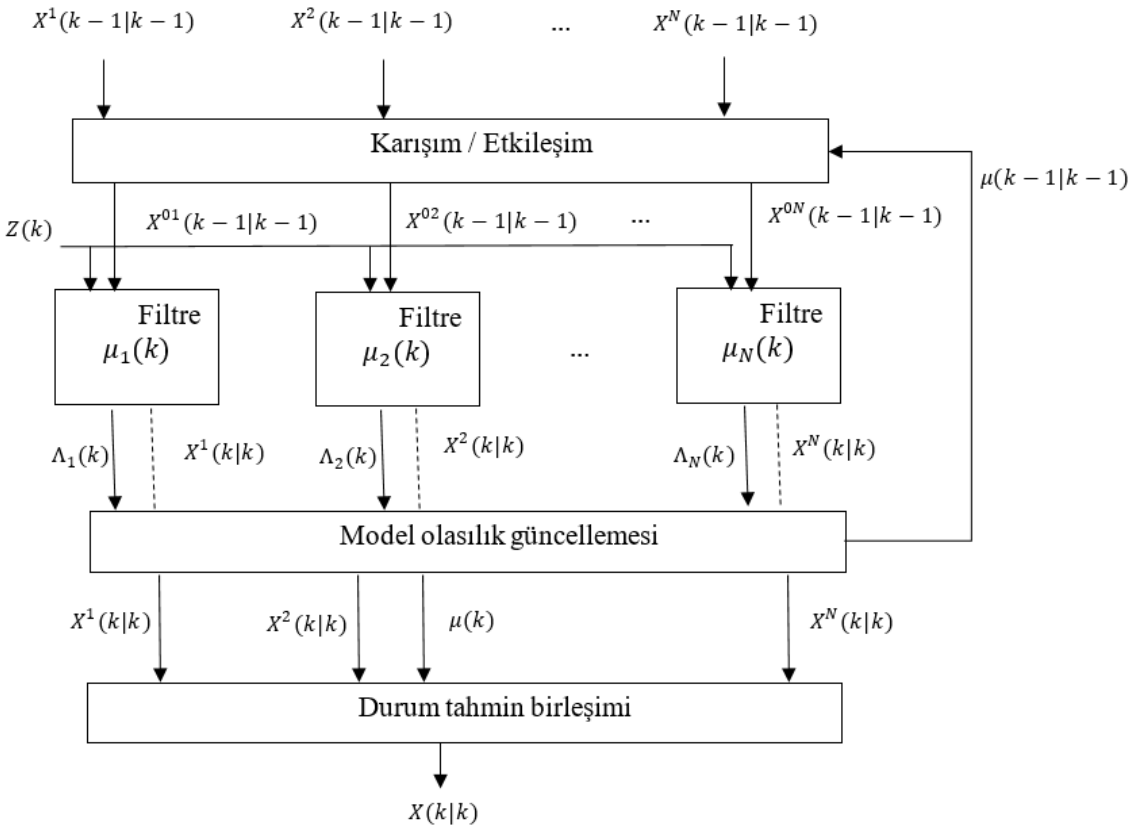
Hedefler daimi olarak aynı modelle hareket etmemektedir. Belli bir müddet sabit hıza sahip bir hedef bir müddet sonra manevra yapabilmektedir. Bu durumda filtre modeli CV seçildiğinde, manevra durumunda hedefe ait konum ve hız kestirimlerinin performansı düşer ve hedef kaybedilebilir. Aynı şekilde, filtre modeli olarak koordineli dönüş modeli seçildiğinde ise hedefin doğrusal hareket ettiği sabit hızlı modele sahip olması durumunda CT modelinin performansı CV modelinin

performansından daha düşük olmaktadır. Bu ikilemlerin üstesinden gelmek için etkileşimli çoklu model (Interacting Multiple Model- IMM) filtresi önerilmektedir [44]-[46]. IMM filtresi, hedef dinamiği kesin olmayan veya hedef dinamiği ile filtre modeli arasında geçişlerin olduğu durumlarda kullanılmaktadır. Bu algoritma, takip sisteminin sonlu sayıda modellerden oluştuğunu varsaymaktadır ve algoritma farklı modeller (modlar) arasında geçiş yaparak sistemin parametrelerini kestirmektedir.

IMM kestirim filtresi, hibrit durum kestirimi algoritmalarından en verimli ve az maliyetli olanıdır [47], [48]. Modlara ait değişkenlerin birbiri içindeki ilişki geçişleri Markov zinciri ile modellenebilmektedir [27], [46], [47], [49].

4.2. Yöntem

N adet model içeren IMM filtresinin akış diyagramı Şekil 4.1'de verilmektedir [50]. Şekilde de görüldüğü üzere her model bir filtre içermektedir ve bu algoritma filtre girdilerinin karışımını, model olasılıklarını ve filtre çıktılarının birleşimini hesaplayan temelde üç aşamayı içermektedir.



Şekil 4.1: IMM akış diyagramı.

N modele sahip IMM filtresi gerçeklenirken *k* zamanı için uygulanacak adımlar aşağıdaki gibi sıralanabilir [51];

- Durum tahmini karışımı
- Model güncellemesi
- Model olasılık hesapları
- Model olasılık güncellemesi
- Durum kestirimlerinin birleşimi

Bu adımlar için kullanılan denklemler aşağıda verilmektedir [51];

- Durum Tahmin Karışımı:

Bu adımda her j modeli için önsel durum tahmini olan $X^{j}_{k-1|k-1}$, durum hata kovaryansı $P^{j}_{k-1|k-1}$ ve ilişkili model olasılığı $\mu_{k-1}(j)$ kullanılarak j modeline ait başlangıç durum kestirimi $X^{0j}_{k-1|k-1}$ ve başlangıç hata kovaryansı $P^{0j}_{k-1|k-1}$ aşağıdaki gibi hesaplanmaktadır.

$$X^{0j}_{k-1|k-1} = \sum_{i=1}^{N} X^{i}_{k-1|k-1}\mu_{k-1|k-1}(i|j) \tag{4.1}$$

$$\mu_{k-1|k-1}(i|j) = \frac{1}{\bar{c}_j} p_{ij}\mu_{k-1}(j) \tag{4.2}$$

$$\bar{c}_j = \sum_{i=1}^{N} p_{ij}\mu_{k-1}(j) \tag{4.3}$$

Denklem (4.2)'de verilmekte olan p_{ij}, i modelinden j modeline geçişteki dönüşüm olasılığını, $\bar{c}_j$ ise normalizasyon sabitini temsil etmektedir.

$$\begin{aligned} P^{0j}_{k-1|k-1} = \quad & \sum_{i=1}^{N} \mu_{k-1|k-1}(i|j)[P^{i}_{k-1|k-1} + (X^{i}_{k-1|k-1} \\ & - X^{oj}_{k-1|k-1})(X^{i}_{k-1|k-1} - X^{oj}_{k-1|k-1})^{T}] \end{aligned} \tag{4.4}$$

- Model Güncellemesi:

IMM algoritması, Kalman filtresi, CMKF, EKF veya UKF filtreleri ile kullanılabilecek bir yapıya sahiptir. KF diğer filtrelere göre daha basit bir yapıya sahiptir ve bu nedenle hesaplama yükü diğer filtrelere göre daha az olduğu için en yaygın olarak kullanılan filtredir. Denklem (4.5)-(4.11)'de IMM algoritmasında kullanılan Kalman filtresi denklemleri verilmektedir. Benzer şekilde diğer filtrelerde bu forma uyarlanabilir.

$$X_{k|k-1}^{j} = F_{k-1}^{j} X_{k-1|k-1}^{0j} \tag{4.5}$$

$$P_{k|k-1}^{j} = F_{k-1}^{j} P_{k-1|k-1}^{0j} (F_{k-1}^{j})^{T} + Q_{k}^{j} \tag{4.6}$$

$$S_{k}^{j} = H_{k}^{j} P_{k|k-1}^{j} (H_{k}^{j})^{T} + R_{k} \tag{4.7}$$

$$K_{k}^{j} = P_{k|k-1}^{j} (H_{k}^{j})^{T} (S_{k}^{j})^{-1} \tag{4.8}$$

$$\tilde{Z}_{k}^{j} = Z_{k}^{j} - H_{k}^{j} X_{k|k-1}^{j} \tag{4.9}$$

$$X_{k|k}^{j} = X_{k|k-1}^{j} + K_{k}^{j} [\tilde{Z}_{k}^{j}] \tag{4.10}$$

$$P_{k|k}^{j} = [I - K_{k}^{j} H_{k}^{j}] P_{k|k-1}^{j} \tag{4.11}$$

- Model Olasılık Hesapları:

Bu adımda k zamanındaki ölçüm inovasyonu $\tilde{Z}_{k}^{j}$ ve inovasyon kovaryansı S_{k}^{j} kullanılarak her j modeli için model olasılığı $\Lambda_{k}(j)$ aşağıdaki gibi hesaplanmaktadır. İlgili denklemdeki $|S_{k}|$ ifadesi inovasyon kovaryansının normunu temsil etmektedir.

$$\Lambda_{k}(j) = \frac{1}{\sqrt{2\pi |S_{k}|}} \exp[-0.5 (\tilde{Z}_{k}^{j})^{T} (S_{k}^{j})^{-1} \tilde{Z}_{k}^{j}] \tag{4.12}$$

- Model Olasılık Güncellemesi:

k zamanındaki j modelinin model olasılığı yada model ağırlığı olarak adlandırılan $\mu_k(j)$ ifadesi Denklem (4.13)'te belirtilen şekilde güncellenmektedir ve denklem içerisinde yer alan c terimi Denklem (4.14)'te, $\bar{c}_j$ terimi ise Denklem (4.3)'te verilmiştir.

$$\mu_k(j) = \frac{1}{c}\Lambda_k(j)\bar{c}_j \tag{4.13}$$

$$c = \sum_{i=1}^{N} \Lambda_k(j)\bar{c}_i \tag{4.14}$$

- Durum Kestirimlerinin Birleşimi:

Şekil 4.1'de görüldüğü üzere son aşama olarak her bir filtre sonucunda elde edilen kestirimler ve her moda ait model ağırlıkları kullanılarak durum kestirimlerinin birleşimi yapılmaktadır. Bu amaçla k zamanına ait durum kestirimi $X_{k|k}$ ve hata kovaryans matrisi $P_{k|k}$ Denklem (4.15) ve Denklem (4.16)'da verilmektedir.

$$X_{k|k} = \sum_{i=1}^{N} X_{k|k}^{i}\mu_k(i) \tag{4.15}$$

$$P_{k|k} = \sum_{i=1}^{N} \mu_k(i)[P_{k|k}^{i} + (X_{k|k}^{i} - X_{k|k})(X_{k|k}^{i} - X_{k|k})^T] \tag{4.16}$$

4.3. IMM-CM Algoritmaları

Şekil 4.1'de IMM yönteminin akış diyagramı verilmektedir ve akış diyagramında da görüldüğü üzere IMM algoritması ile algoritmada kullanılan filtre birbirinden bağımsızdır. Dolayısıyla genelde Kalman filtresi kullanılan IMM algoritması için ihtiyaca göre farklı filtrelerde uyumlu hale getirilebilir. Radar sisteminde ölçümlerin küresel koordinatta alınması durumunda ölçüm gürültüsünün de küresel koordinatta eklenmesi ve daha sonra Kalman filtresi için ölçümlerin

kartezyen koordinata dönüştürülmesi sonucunda gürültünün ortalama ve kovaryansının değiştiği, dolayısıyla Kalman filtresinin temel varsayımına ters düştüğü durumu Bölüm 2.4'te açıklanmıştı. Bu nedenle IMM filtresinin farklı hedef hareketleri için farklı modelleri kullanması avantajını kullanıp, aynı zamanda koordinat dönüşümlerinden kaynaklı sapma durumunu mümkün olduğunca elimine eden CMKF algoritmasını da içeren IMM-CMKF algoritması önerilmektedir.

IMM algoritmasının avantajları Bölüm 4.1'de, algoritma adımları ise Bölüm 4.2'de ele alınmıştır. Algoritma adımlarından '*model güncellemesi*' adımı için Kalman filtre denklemleri Denklem (4.5)-(4.11)'de örnek olarak verilmiştir. Bu denklemlerin yerine CMKF veya UKF gibi algoritma denklemleri uyarlanabilir. Belirtildiği üzere bu bölümde IMM-CMKF algoritması üzerinde durulmaktadır ve CMKF algoritmasının IMM yöntemine uyarlanmasındaki '*model güncellemesi*' adımı için gereken denklemler aşağıda verilmektedir. Denklemlerde kullanılan dönüştürülmüş ölçüm gürültüsü R_k^{CM} ve j moduna ait dönüştürülmüş ölçüm $Z_k^{CM,j}$, CMKF algoritmaları için kullanılmak istenen CM algoritmasına (CM-U, CM-MU, CM-DU gibi) ve sistem model boyutuna (iki veya üç boyut gibi) göre Bölüm 3'te belirtilen şekilde hesaplanmaktadır.

$$X_{k|k-1}^{j} = F_{k-1}^{j} X_{k-1|k-1}^{0j} \tag{4.17}$$

$$P_{k|k-1}^{j} = F_{k-1}^{j} P_{k-1|k-1}^{0j} (F_{k-1}^{j})^{T} + Q_k^{j} \tag{4.18}$$

$$S_k^{j} = H_k^{j} P_{k|k-1}^{j} (H_k^{j})^{T} + R_k^{CM} \tag{4.19}$$

$$K_k^{j} = P_{k|k-1}^{j} (H_k^{j})^{T} (S_k^{j})^{-1} \tag{4.20}$$

$$\tilde{Z}_k^{j} = Z_k^{CM,j} - H_k^{j} X_{k|k-1}^{j} \tag{4.21}$$

$$X_{k|k}^{j} = X_{k|k-1}^{j} + K_k^{j} [\tilde{Z}_k^{j}] \tag{4.22}$$

$$P_{k|k}^{j} = [I - K_k^{j} H_k^{j}] P_{k|k-1}^{j} \tag{4.23}$$

Bölüm 5.4'te farklı senaryolar için IMM-CMKF algoritmaları performanslarının analizi yapılarak IMM-KF ve IMM-UKF performansları ile karşılaştırılmıştır. Kullanılan her bir senaryo için düzensiz örnekleme, sensör sapması ve ölçüm gürültüsünün takip performansına etkisi incelenmektedir. IMM algoritması için CV, CA ve CT olmak üzere üç model kullanılmıştır. İki boyutlu sistem için kullanılan hedef hareket modellerinden sabit hız modeli için kullanılan $\boldsymbol{F}^{CV}$ ve $\boldsymbol{\Gamma}^{CV}$ Denklem (2.14)'te, sabit ivme modeli için kullanılan $\boldsymbol{F}^{CA}$ ve $\boldsymbol{\Gamma}^{CA}$ Denklem (2.17)'de verilmiştir. İki boyutlu sistemde koordineli dönüş modeli için kullanılan $\boldsymbol{F}^{CT}$ ve $\boldsymbol{\Gamma}^{CT}$ ise Denklem (4.24)-(4.25)'te verilmektedir [52], [53]. Denklemlerde yer alan ω dönüş oranını temsil etmektedir.

$$\boldsymbol{F}^{CT} = \begin{bmatrix} 1 & 0 & \sin(\omega T)/\omega & -(1-\cos(\omega T))/\omega \\ 0 & 1 & (1-\cos(\omega T))/\omega & \sin(\omega T)/\omega \\ 0 & 0 & \cos(\omega T) & -\sin(\omega T) \\ 0 & 0 & \sin(\omega T) & \cos(\omega T) \end{bmatrix} \tag{4.24}$$

$$\boldsymbol{\Gamma}^{CT} = \begin{bmatrix} (T^2/2)\boldsymbol{I}_2 \\ (T)\boldsymbol{I}_2 \end{bmatrix} \tag{4.25}$$

Üç boyutlu sistem için kullanılan $\boldsymbol{F}^{CV}$, $\boldsymbol{\Gamma}^{CV}$, $\boldsymbol{F}^{CA}$,$\boldsymbol{\Gamma}^{CA}$, $\boldsymbol{F}^{CT}$ ve $\boldsymbol{\Gamma}^{CT}$ ise sırasıyla Denklem (2.19), Denklem (2.21) ve Denklem (4.26)-(4.27)'de verilmektedir. Üç boyutlu sistemdeki koordineli dönüş *x* ve *y* eksenlerinde olduğu için *z* ekseni için sabit hız modelindeki değerler kullanılmıştır.

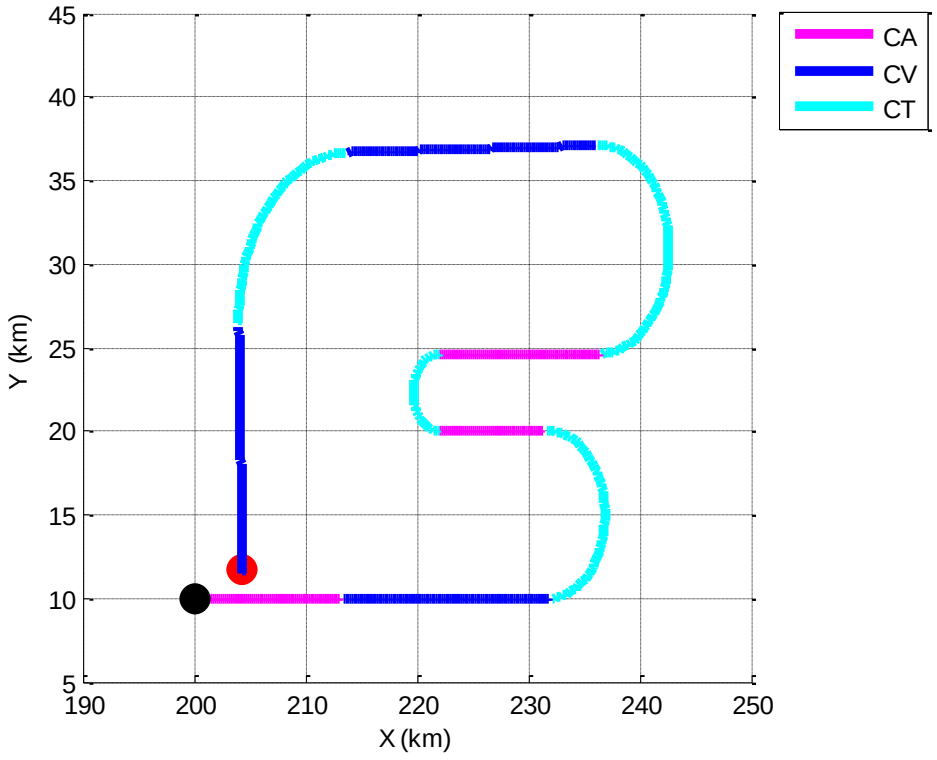
$$\boldsymbol{F}^{CT} = \begin{bmatrix} 1 & 0 & 0 & \sin(\omega T)/\omega & -(1-\cos(\omega T))/\omega & 0 \\ 0 & 1 & 0 & (1-\cos(\omega T))/\omega & \sin(\omega T)/\omega & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & \cos(\omega T) & -\sin(\omega T) & 0 \\ 0 & 0 & 0 & \sin(\omega T) & \cos(\omega T) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.26}$$

$$\boldsymbol{\Gamma}^{CT} = \begin{bmatrix} (T^2/2)\boldsymbol{I}_3 \\ (T)\boldsymbol{I}_3 \end{bmatrix} \tag{4.27}$$

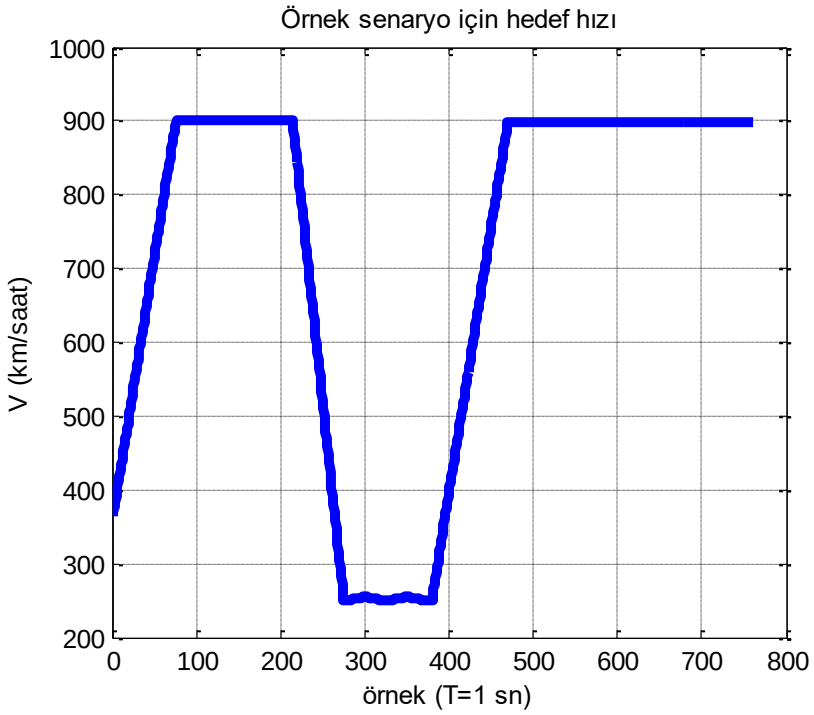
4.4. Testler

Bu bölümde, IMM-CMKF algoritmalarının performans analizi üç boyutta yapılmaktadır. Performans analizi için farklı senaryolar üretilmekte ve üzerinde durulan algoritma, hesaplama yükü ve hata miktarı bakımından sık kullanılan IMM-KF ve IMM-UKF algoritmaları ile karşılaştırılmıştır.

Test senaryosu için üretilen verinin *x-y* koordinat düzlemindeki görüntüsü Şekil 4.2'de verilmektedir. Hedefe ait yükseklik bilgisi ise, *z* ekseninde genliği 200 metre olacak şekilde sinüs dalgası şeklinde salınarak hareket etmektedir. Pratikte hedefler, sadece sabit hızlı hareket, sadece sabit ivmeli hareket veya sadece manevra hareketi yapmamaktadır. Bu hareketlerin hedef uçuşlarında beraber kullanılması daha gerçekçidir. Şekil 4.2'de, bir hedefe ait uçuş yörüngesi, bu üç model kullanılarak tasarlanmıştır. Senaryoda kullanılan modeller şekilde belirtildiği gibi sabit ivme modeli mor renkle, sabit hız modeli lacivertle ve koordineli dönüş modeli ise açık mavi ile gösterilmiştir.



Şekil 4.2: Sabit Δt senaryosu için örnek senaryo görüntüsü.

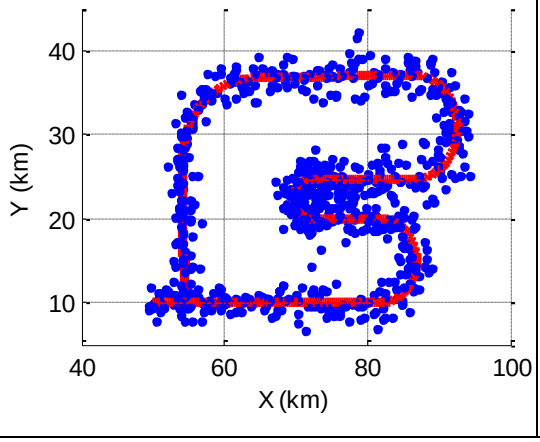
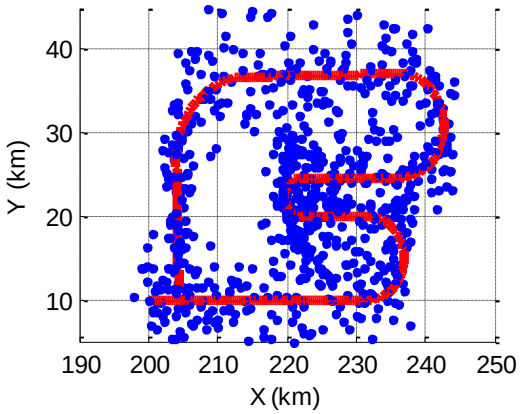


Şekil 4.3: Sabit Δt senaryosu için hedef hız grafiği.

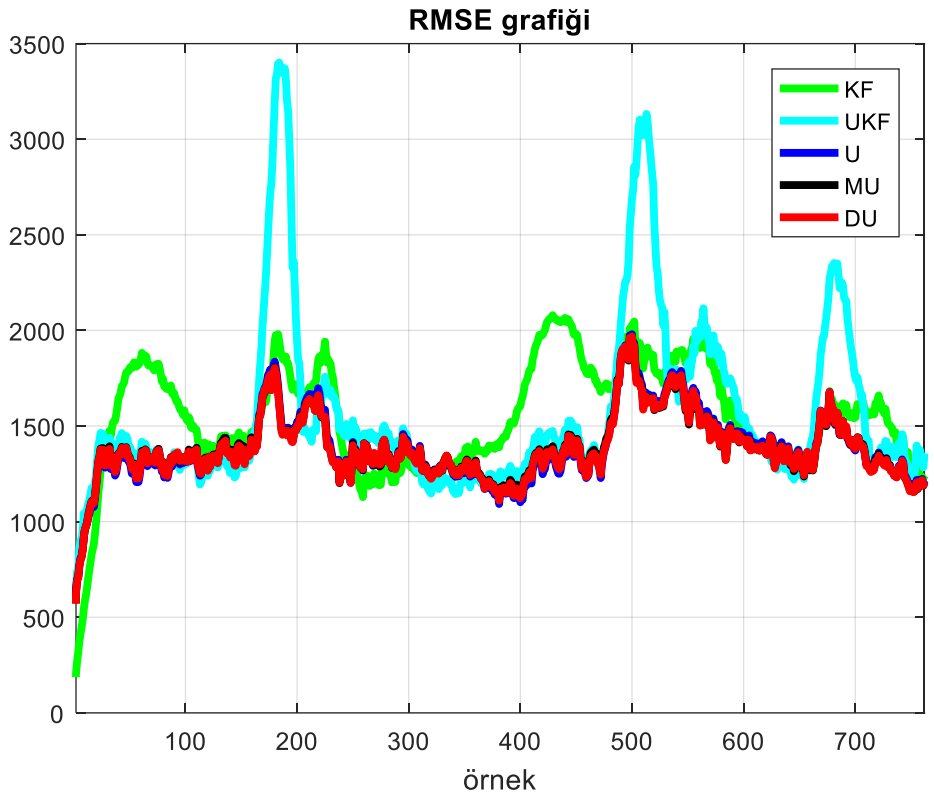
Şekil 4.3'te görüldüğü üzere senaryolardaki hedef hızı yaklaşık 250-900 km/saat aralığında alınmıştır ve hedef hareketini Şekil 4.2'de siyah nokta ile belirtilen yerden başlayarak kırmızı nokta ile belirtilen konuma kadar devam ettirmektedir. Veri üretiminde, her ölçüm örneği arasında geçen süre olan Δt değeri 1 saniye olarak alınmıştır ve hedefin hareket süresi yaklaşık 15 dakikadır.

Bu bölümde, algoritmaların takip performanslarının karşılaştırılmasında sabit örnekleme durumunda ölçüm gürültüsünün takip performanslarına etkisi incelenmektedir. Düşük ve yüksek gürültülü olmak üzere iki farklı senaryo oluşturulmuştur. Ölçüm gürültüsü, uzak mesafede açı sapmaları yakın mesafedekine göre çok daha fazla olabildiğinden, ölçüm menzilinin radar sistemine uzak olduğu durumlarda, yakın olduğu durumlara göre daha fazla olmaktadır. Bu nedenle ölçüm gürültüsü az ve çok olan durumlar için kullanılan senaryolar ve örnek ölçüm gürültüleri Tablo 4.1'de görüldüğü gibidir. Şekillerdeki kırmızı eğri hedef rotasını, mavi noktalar ise ölçümleri temsil etmektedir. Her iki senaryo içinde ölçüm gürültüsünün standart sapmaları r, θ ve φ için sırasıyla 1 km, 1 derece ve 1 derece olarak alınmıştır. Şekilde de görüldüğü üzere x, y ve z eksenlerinde [0,0,0] km konumlu radara uzaklığı yakın olan senaryo durumunda ölçüm gürültüsü az iken, uzak olan durum için ölçüm gürültüsü daha fazla dağınık durumdadır.

Tablo 4.1: Sabit Δt senaryoları için hedef rotası ve ölçümler.

Ölçüm gürültüsü az	Ölçüm gürültüsü fazla
	

Performans karşılaştırması için Denklem (3.38)'de verilen RMSE değeri kullanılmıştır ve algoritmalar 100 Monte Carlo'da çalıştırılmıştır. Şekil 4.4'te ölçüm gürültüsü fazla olan senaryo için elde edilen takip performanslarının RMSE grafikleri verilmiştir. Elde edilen sonuçlarda özellikle hedefin manevra yaptığı zamanlarda IMM-UKF algoritmasının diğer yöntemlerden daha kötü sonuç verdiği görülmektedir. Ayrıca IMM-KF algoritması da IMM-CMKF algoritmasından daha fazla hatalı sonuca sahiptir. IMM-CMKF yöntemleri ise genel olarak birbirine yakın hata üretmiştir. Bu nedenle Tablo 4.2'de verilen gerçekleme süreleri göz önünde bulundurulduğunda en az süreli IMM-KF algoritması iken ikinci en az süreli yöntem ise IMM-CMKF_DU yöntemidir. IMM-KF yönteminin uygulaması diğer yöntemlerden daha kısa sürmesine karşın takip performansı diğer yöntemlere göre çok daha kötü olduğu için IMM-CMKF_DU algoritması hem kısa sürede gerçeklenebilir olması hem de takip performansının iyi olması nedeniyle tercih edilebilir bir algoritmadır.

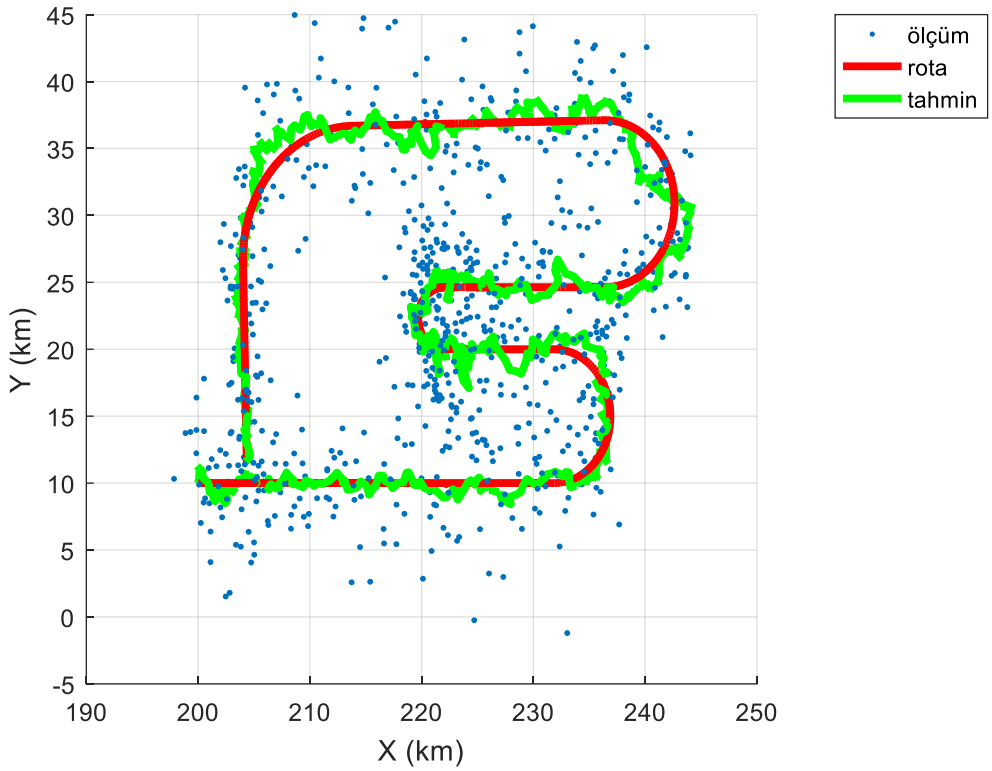


Şekil 4.4: Ölçüm gürültüsü fazla olan senaryo için RMSE grafiği (T=1 sn).

Tablo 4.2: Ölçüm gürültüsü fazla olan senaryo için gerçekleme süresi.

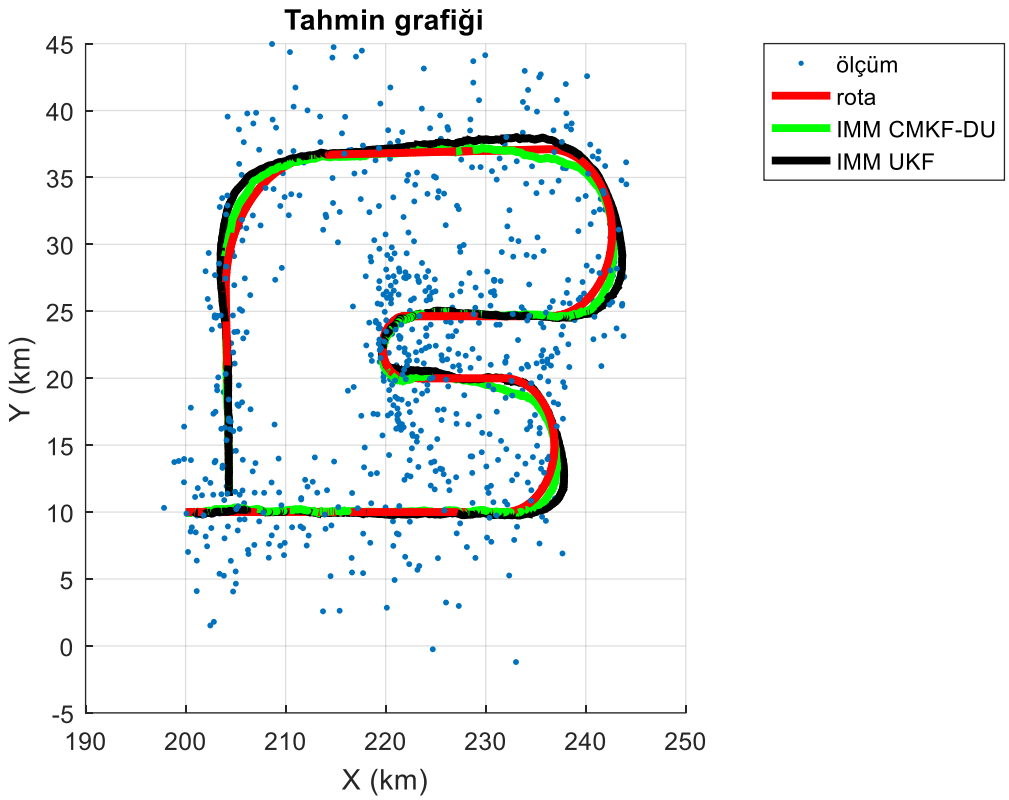
Yöntem	**IMM-KF**	**IMM-UKF**	**IMM-CMKF_U**	**IMM-CMKF_MU**	**IMM-CMKF_DU**
Süre (sn)	166	559	262	225	222

Şekil 4.5'te IMM-CMKF_DU algoritması için bir MC için örnek kestirim sonucu görüntüsü verilmektedir. Görüldüğü üzere ölçümler filtrelenerek hedefin gerçek güzergahına yakın kestirim sonuçları elde edilmiştir.



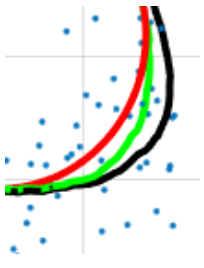
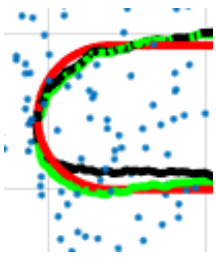
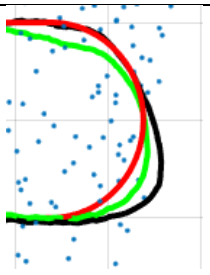
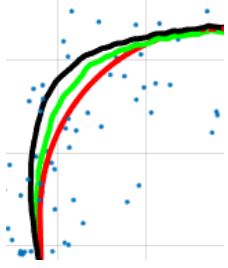
Şekil 4.5: IMM-CMKF_DU için örnek ölçüm, hedef rotası ve kestirim sonucu grafiği.

Şekil 4.6'da IMM-CMKF_DU VE IMM-UKF algoritmalarının 100 MC sonucu elde edilen ortalama kestirimlerinin grafiği verilmiştir. Tablo 4.3'te ise Şekil 4.6'daki manevra kısımlarının yakınlaştırılmış grafikleri verilmiştir. Hedef *x* ve *y* eksenlerinde sırasıyla 200 km ve 10 km konumundan hareketine başlayarak önce sola koordineli dönüş ardından sağa koordineli dönüş ve daha sonra iki kez yine sola koordineli dönüş hareketi yaparak hareketini tamamlamıştır. Tabloda ise ilgili koordineli dönüşler (CT) için manevra numarası verilmiştir. Tablodaki şekillerde görüldüğü üzere manevra durumunda IMM-UKF algoritması IMM-CMKF_DU algoritmasına göre daha fazla ıraksamaktadır. Testlerde, hedef hareketindeki tüm dönüşler için koordineli dönüş fazla ya da az olsun Denklem 4.26'daki ω koordineli dönüş oranlarının hepsi aynı değer seçilmiştir. Bu durumda dahi IMM-CMKF_DU algoritması manevraları yakalamakta fazla gecikmemiştir.



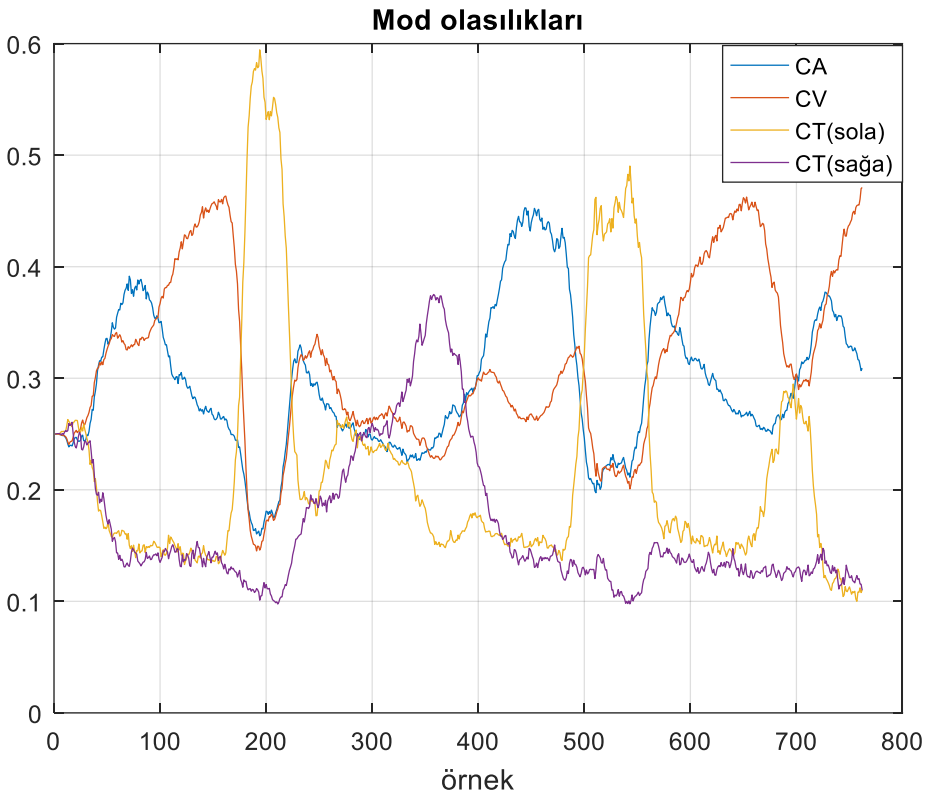
Şekil 4.6: IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği.

Tablo 4.3: IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu manevra ayrıntı grafikleri.

CT numarası	**1**	**2**	**3**	**4**
Dönüş grafiği				

Testlerde kullanılan tüm filtrelerde IMM mod olasılığı, kullanılan CV, CA, CT(sağa) ve CT(sola) olmak üzere dört mod içinde 0.25 değeri ile eşit olarak başlatılmıştır. Genelde mod olasılıkları başlangıçta bilinmediği için eşit olarak başlatılmaktadır fakat uygulamaya göre hangi modun daha fazla kullanıldığı bilinmesi durumunda kullanılan moda ait mod olasılığı diğerlerinden daha büyük seçilebilir. Fakat tüm mod olasılıklarının toplamının 1'e eşit olması gerektiğine dikkat edilmelidir.

Şekil 4.7'de IMM-CMKF_DU algoritması için elde edilen mod olasılıklarının ortalama grafiği verilmiştir. Grafikte görüldüğü üzere her mod eşit değerden başlatılmıştır ve kullanılan moda göre moda ait mod olasılık değeri artıp azalmıştır. Şekil 4.2'de senaryoya ait mod bilgileri verilmişti. Mod bilgileri ve mod olasılıklarının ortalama grafiğine bakıldığında sonuçların tutarlı olduğu görülmektedir. Başlangıçta hedef bir müddet CA modunda, daha sonra ise CV modunda hareket ederek sola doğru ilk dönüşünü gerçekleştirmiştir. Mod olasılık grafiğinde de öncelikle CA ve CV modlarının olasılığında sonra ise sol yönlü CT modunun olasılığında artış olmuştur. Hedef ilk dönüş manevrasını yaptıktan sonra ise bir müddet CA moduyla ilerleyerek ikinci manevrası olan sağa koordineli dönüş hareketini gerçekleştirmiştir ve yine CA moduyla bir müddet ilerledikten sonra tekrar sola koordineli dönüş hareketi yaparak manevra yapmıştır. Mod olasılık grafiğine bakıldığında da aynı şekilde sağ yönlü CT modunun, sonrasında CA modunun, daha sonrasında sol yönlü CT modunun ve son olarak tekrar CA modunun olasılığı artmıştır. Hedef üçüncü manevrasından sonra CV modunda, sol yönlü koordineli dönüş modunda ve son olarakta tekrar CV modunda ilerleyerek hareketini tamamlamıştır. Grafikte de görüldüğü üzere hedef CV modunda hareket ediyorken azda olsa CA modunun olasılığı da artış gösterebilmektedir. Bunun nedeni hedefin z ekseninde sinüs dalgası şeklinde hareket etmesidir. Aynı şekilde hedef CA modunda ilerlerken CV modunun olasılığında da küçük artışlar olabilmektedir. Bunu nedeni ise ivmenin bazı yerlerde 2 m/sn^2 gibi az olması ve hareketin kısa süreli olmasından kaynaklı adaptasyon gecikmesindendir.

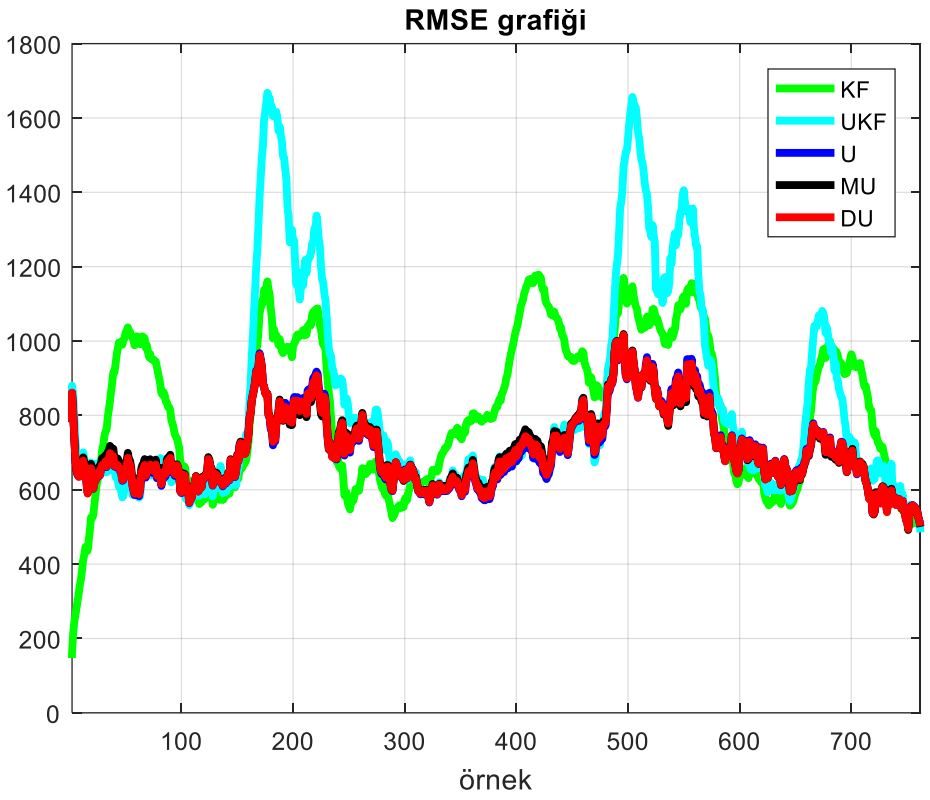


Şekil 4.7: IMM-CMKF_DU için örnek mod olasılık grafiği.

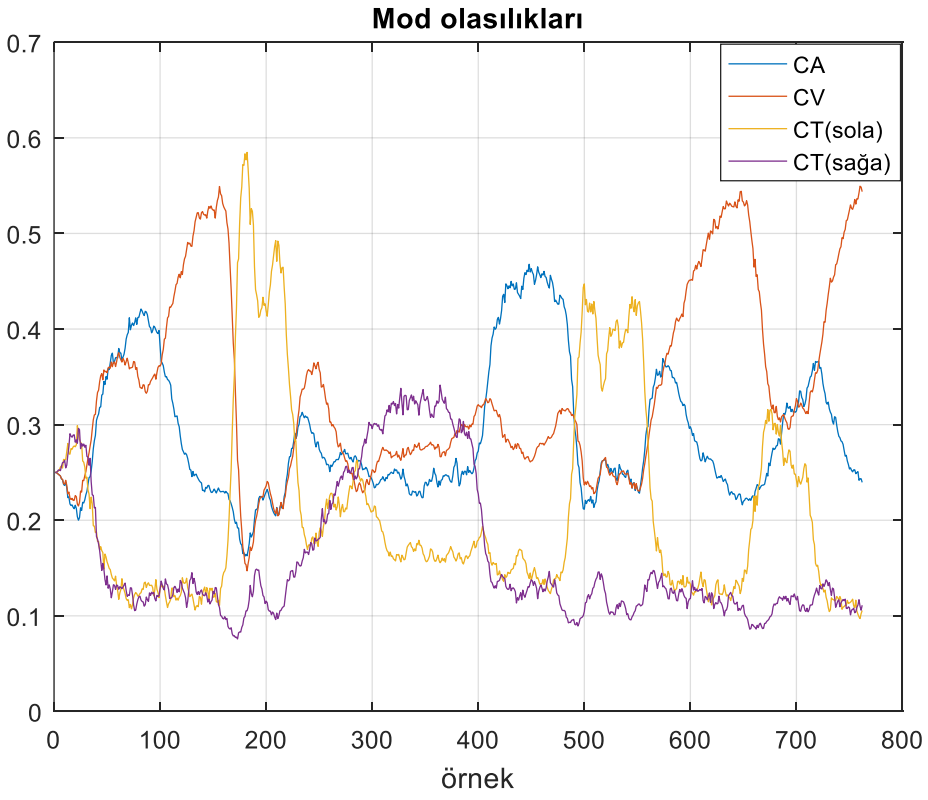
Şekil 4.8'de, Tablo 4.1'de ekran görüntüsü verilen, ölçüm gürültüsü az olan senaryo için elde edilen takip performanslarının RMSE grafikleri verilmiştir. Grafikte görüldüğü üzere ölçüm gürültüsü fazla olan senaryodaki gibi IMM-CMKF algoritmalarının performansı IMM-UKF ve IMM_KF filtrelerinin performansından daha iyidir ve IMM-CMKF algoritmaları manevra durumlarına daha çabuk adapte olmaktadır. Ayrıca algoritmaların gerçekleme sürelerine bakıldığında, Bölüm 2.2.3'te verilmekte olan IMM-UKF algoritmasının denklemlerinin fazlalığından ve karmaşıklığından işlem yükünün fazla olması beklenilir bir durumdur.

Tablo 4.4: Ölçüm gürültüsü az olan senaryo için gerçekleme süresi.

Yöntem	**IMM-KF**	**IMM-UKF**	**IMM-CMKF_U**	**IMM-CMKF_MU**	**IMM-CMKF_DU**
Süre (sn)	54	139	56	56	61



Şekil 4.8: Ölçüm gürültüsü az olan senaryo için RMSE grafiği (T=1 sn).

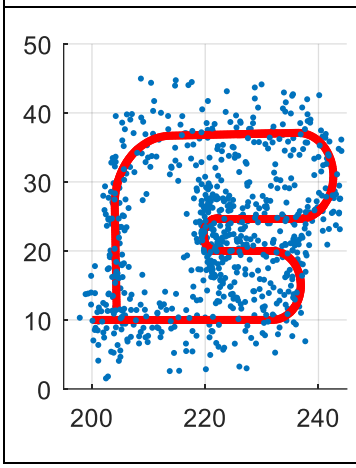
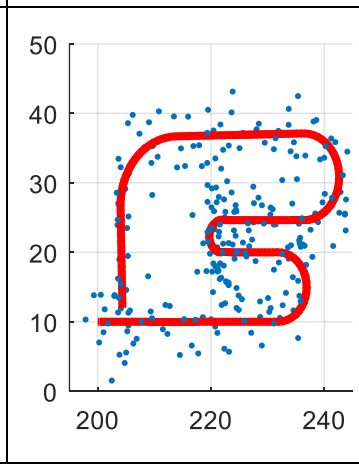
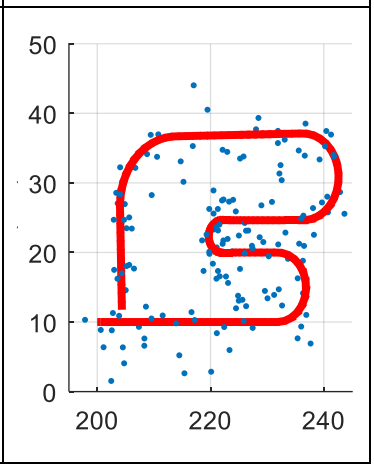


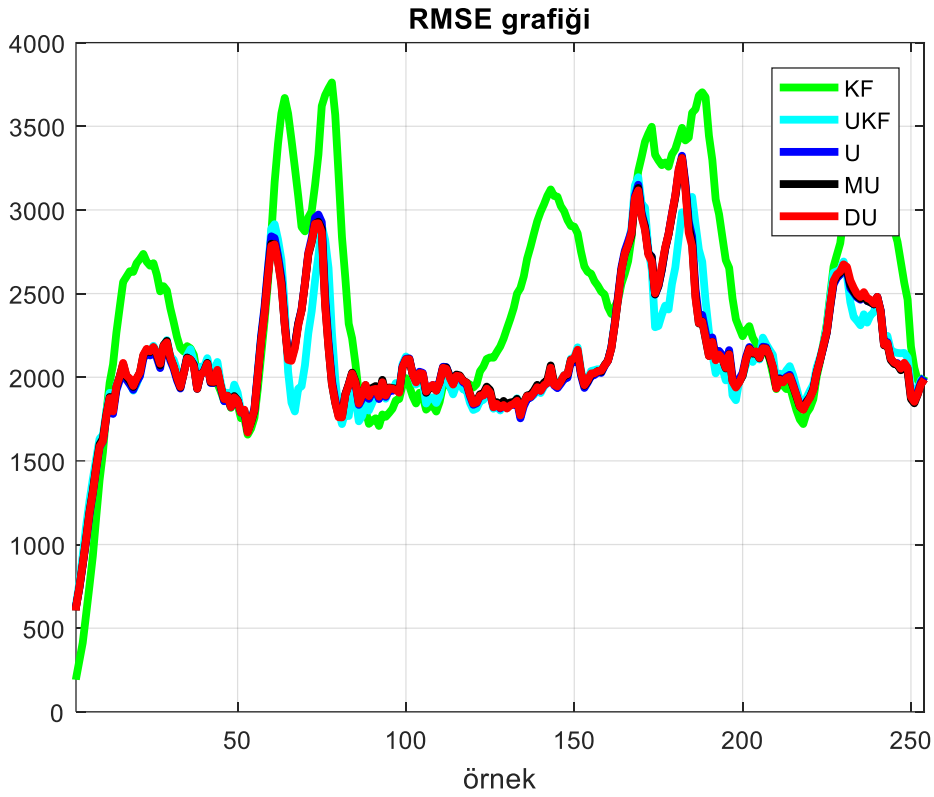
Şekil 4.9: IMM-CMKF_DU için örnek mod olasılık grafiği.

Şekil 4.4'teki ölçüm gürültüsü fazla olan senaryo ve Şekil 4.8'deki az olan senaryolar için elde edilen RMSE grafikleri karşılaştırıldığında ise gürültünün az olduğu senaryoda hatanın her örnek için daha az olduğu açıkça görülmektedir. Gürültü fazla olan durumda RMSE grafiği IMM-CMKF algoritmaları için ortalama 1450 metre iken, gürültünün az olduğu durumda hata ortalama 750 metreye kadar inmektedir. IMM-UKF açısından bakıldığında ise gürültü fazla iken manevra durumunda 3400 metrelere kadar sıçrama yapan hata değeri, gürültünün daha az olduğu durumda ise yaklaşık 1700 metrelik hata değerine sıçrama yapmıştır. Ölçüm gürültüsünün mod olasılıklarına etkisi bakımından incelendiğinde ise grafiklerden açıkça görüldüğü üzere küçük değişimlerin dışında fazla değişimin olmadığı söylenebilir.

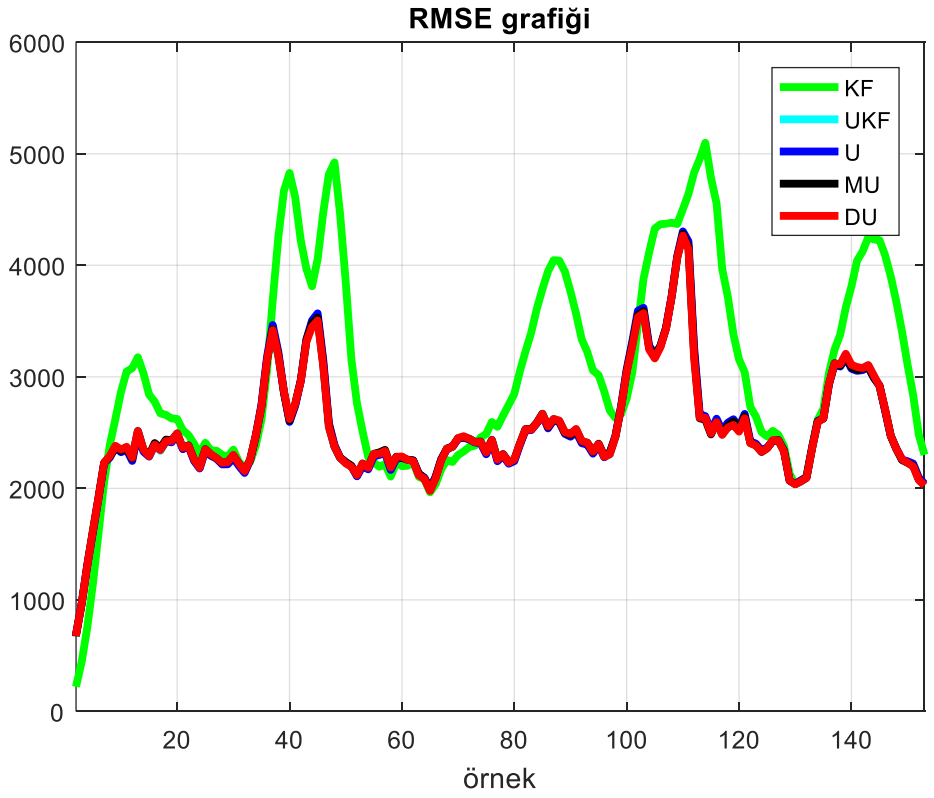
Tablo 4.5'te ölçüm gürültüsü fazla olan senaryo için alınan ölçümlerin, ölçümler arası süre olan T'ye göre değişimi verilmiştir. Grafiklerdeki kırmızı eğri hedef güzergahını, mavi noktalar ölçümleri, yatay eksen *x* (km) eksenini, düşey eksen ise *y* (km) eksenini göstermektedir. Grafiklerden de açıkça görüldüğü üzere ölçümler arası süre artıkça aynı hedefe ait alınan ölçümler azalmaktadır. Bu durumda ölçüm sıklığının azalmasının takip performanslarına etkisinin incelenmesi için takip algoritmalarının T=3 ve T=5 için elde edilen RMSE grafikleri sırasıyla Şekil 4.10 ve Şekil 4.11'de; IMM-CMKF_DU algoritması için bir MC sonucunda elde edilen kestirim grafikleri ise Tablo 4.6'da verilmektedir.

Tablo 4.5: T değerlerine göre ölçüm sıklığının değişimi.

T=1	**T=3**	**T=5**
		

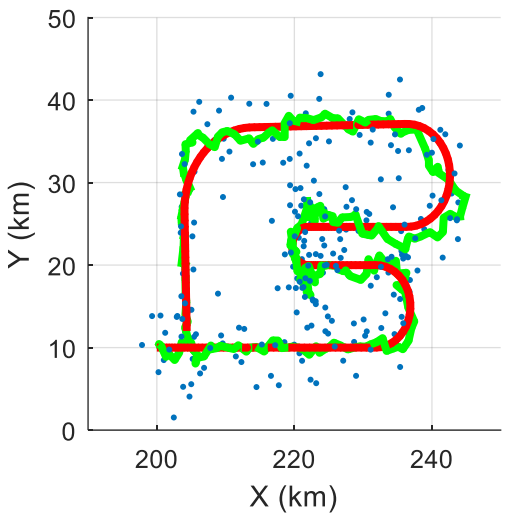
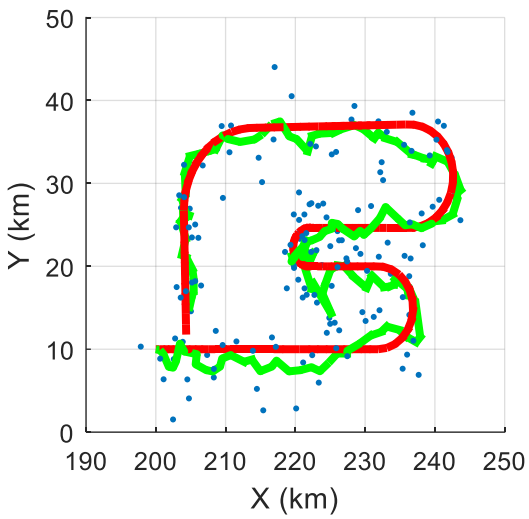


Şekil 4.10: Ölçüm gürültüsü fazla olan senaryo için RMSE grafiği (T=3 sn).



Şekil 4.11: Ölçüm gürültüsü fazla olan senaryo için RMSE grafiği (T=5 sn).

Tablo 4.6: IMM-CMKF_DU için farklı T değerinde ortalama kestirim grafikleri.

T=3	T=5
	

RMSE grafiklerinden görüldüğü üzere ölçüm sıklığı azaldıkça takip performansı olumsuz etkilenmektedir. Şekil 4.4'te T=1 için elde edilen RMSE grafiği ile Şekil 4.11'de T=5 için elde edilen RMSE grafiği karşılaştırıldığında performanstaki azalma açıkça görülmektedir. T=1 iken IMM-CMKF algoritmasının hatası manevra durumunda dahi 2 km'yi aşmaz iken, T=5 olduğu durumda algoritmanın kalıcı hale girmesinden sonraki hata minimum 2 km olmuştur ve manevra durumlarındaki hata değeri 4 km'yi aşmıştır. Buna karşın T=1 için IMM-UKF algoritmasının performansı IMM-CMKF algoritmalarına göre daha kötü iken, T=3 ve T=5 iken IMM-UKF algoritmasının performansı IMM-CMKF algoritmalarına yaklaşmıştır. Genel olarak hedef hakkında daha az konum bilgisinin alınması beklenildiği gibi takip performansını düşürmüştür. Ayrıca ölçüm bilgisinin az olması durumunda takip algoritmaları arasındaki performans farkları azalmıştır.

4.5. Mod Olasılık Hesabında Markov Zinciri'nin Etkisi

IMM algoritmasının durum tahmin karışımı aşamasında, Denklem (4.2) ve Denklem (4.3)'teki p_{ij} matrisi, i ve j model numarası olmak üzere Markov zinciri mantığına dayalıdır ve dört modelin kullanıldığı bir IMM algoritması için Denklem (4.28)'deki gibi tanımlanmaktadır. Her satırdaki değerlerin toplamı 1'e eşit olmak zorundadır. $\boldsymbol{p}_{ij}$ matrisi köşegene göre simetrik olmak zorunda değildir. Matristeki her değer mod arasındaki geçişi belirler. Örneğin $\boldsymbol{p}_{12}$ olasılık değeri birinci moddan ikinci moda geçişin olasılığı belirtmektedir. Bu nedenle, her $\boldsymbol{p}_{ij}$ değeri 0 ve 1 arasında değer almaktadır.

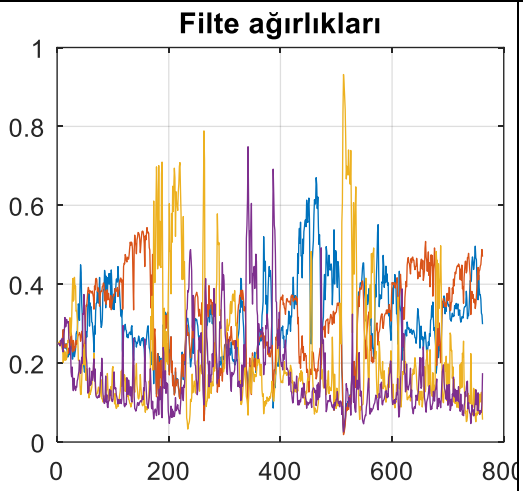
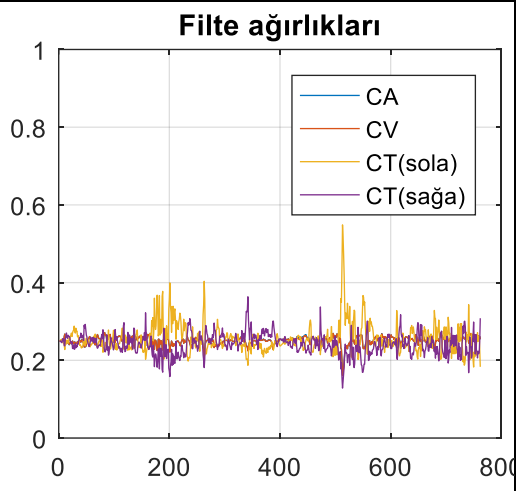
$$\boldsymbol{p}_{ij} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix} \tag{4.28}$$

Markov zincirinin mod olasılıkları üzerindeki etkisinin incelenmesi için Denklem (4.29) ve Denklem (4.30)'da verilen iki $\boldsymbol{p_{ij}}$ matrisi tanımlanarak, Tablo 4.1'deki ölçüm gürültüsü fazla olan senaryoda T=1 sn. için CMKF_DU algoritması kullanılarak ortalama filtre ağırlık grafikleri elde edilmiştir. $\boldsymbol{p_{ij}^{geçiş\ az}}$ ve $\boldsymbol{p_{ij}^{geçiş\ çok}}$ matrisleri tanımlanırken, modlar arası geçişin az ve çok olmasına göre değerlendirilmiştir. Modlar arasındaki geçiş olasılığının büyük olması modların sürekli değişmesine ve kararsız olmasına yol açmaktadır. CMKF_DU algoritması için bir MC sonucu elde edilen filtre ağırlık grafikleri Tablo 4.7'de, 100 MC sonucu elde edilen filtre ağırlık grafikleri ise Tablo 4.8' de verilmiştir.

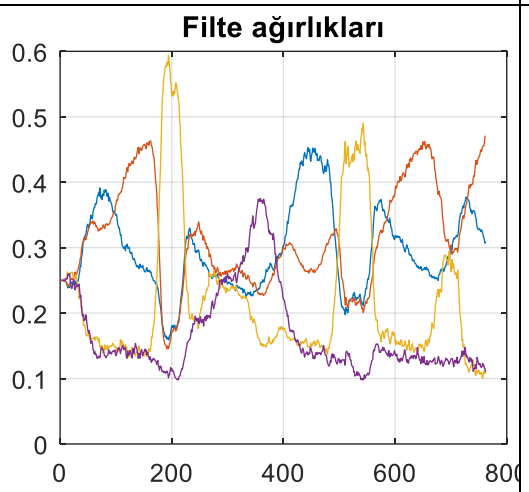
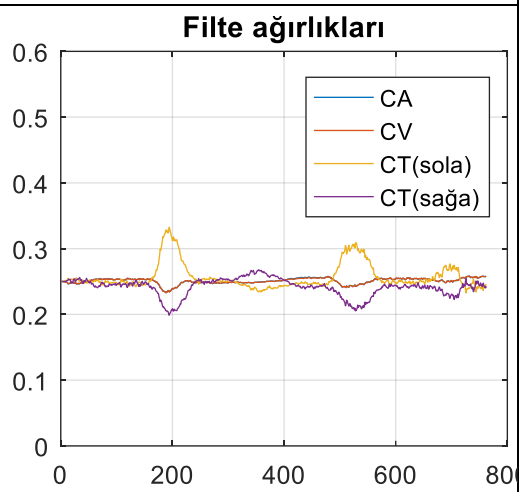
$$\boldsymbol{p}_{ij}^{geçiş\ az} = \begin{bmatrix} 0.97 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.97 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.97 \end{bmatrix} \tag{4.29}$$

$$\boldsymbol{p}_{ij}^{geçiş\ çok} = \begin{bmatrix} 0.85 & 0.05 & 0.05 & 0.05 \\ 0.05 & 0.85 & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.85 & 0.05 \\ 0.05 & 0.05 & 0.05 & 0.85 \end{bmatrix} \tag{4.30}$$

Tablo 4.7: Markov zinciri, bir MC için filtre ağırlık grafikleri.

Geçişin az olması durumu	Geçişin çok olması durumu
	

Tablo 4.8: Markov zinciri, yüz MC için ortalama filtre ağırlık grafikleri.

Geçişin az olması durumu	Geçişin çok olması durumu
	

Tablodaki grafiklerden açıkça görüldüğü üzere farklı modlar arasındaki geçiş olasılığının çok olması durumunda modlar arası geçiş belli bir bölgeye sıkışacak ve her mod filtrelerde yaklaşık aynı oranda kullanılacaktır. Bu durumda, sabit hız ile hareket eden hedef için sabit hız modu daha fazla kullanılmamakta, hedef manevra yapmamasına karşın koordineli dönüş modeli sabit hız modeli kadar takip algoritmasında kullanılacak, bu durum sonucunda ise takip performansını olumsuz etkileyecektir. Fakat farklı modlar arası geçiş olasılığının az olması durumunda ise

modlar arasındaki geçişler daha yavaş olacak, bu durumda ise hedef sabit hızlı hareket sonrasında manevra hareketine geçerken modlarında sabit hareket modelinden manevra hareket modeline geçmesi daha uzun sürecektir ve bunun sonucunda takip performansı olumsuz etkilenecektir. Bu nedenle kullanılan uygulamada hedeflerin hareket modellerindeki geçişlerde göz önünde bulundurularak mod olasılıkları belirlenmelidir.

4.6. SIMM-DIMM-MIMM Algoritmaları ve UKF ve CMKF ile kullanımı

Bu bölümde, [8]'deki füzyon kriterlerine dayalı olan SIMM, DIMM ve MIMM algoritmaları üç boyutta gerçeklenerek, Kalman filtresi yerine UKF ve CMKF algoritmaları uyarlanmıştır. Bu bölümde, öncelikle SIMM, DIMM ve MIMM algoritmaları hakkında genel bilgi ve IMM algoritmasından farkları verilmiştir. Sonraki bölümde ise bu tez kapsamında gerçekleştirilen üç boyutta SIMM, DIMM ve MIMM algoritmalarına UKF ve CMKF algoritmaları uyarlanarak, elde eldilen takip algoritmalarının RMSE performansları kıyaslanmaktadır.

4.6.1. SIMM, DIMM ve MIMM Algoritmaları

Bu bölümde [8]'deki SIMM, DIMM ve MIMM algoritmalarının IMM algoritmasından farkı ve bu algoritmaların temel yaklaşımlarına değinilecektir. Denklem (4.31)'de IMM algoritmasında $\boldsymbol{M_j^k}$ modelinin ağırlık güncellemesi verilmektedir. Denklemdeki c normalizasyon sabitidir. Denklemde de görüldüğü üzere ağırlık güncellemesi olasılık yoğunluk fonksiyonu ve kümülatif olasılık fonksiyonundan türetilmektedir. Kümülatif olasılık fonksiyonu [0-1] değerleri arasında değer alırken, olasılık yoğunluk fonksiyonunun bir sınırı yoktur. Bu nedenle iki farklı seviye kullanılır ve bunun sonucunda elde edilen $\boldsymbol{\mu_j^k}$ yaklaşık olasılıktır. Takip sistemlerinde, kestirim hatasının mümkün olduğunca küçük olması istenmektedir bu nedenle [8]'de füzyon kriterlerine dayalı modifiye edilmiş IMM algoritmaları öne sürülmektedir. Modifiye edilen IMM algoritmaları skalerle,

köşegenle ve matrisle ağırlıklandırılmıştır ve bu algoritmalar sırasıyla SIMM, DIMM ve MIMM algoritmalarıdır.

$$\mu_j^k \triangleq \boldsymbol{P}\{M_j^k | Z^{k-1}\} = \frac{p[\boldsymbol{z}(k)|M_j^k, Z^{k-1}].\boldsymbol{P}\{M_j^k|Z^{k-1}\}}{c} = \frac{N[\boldsymbol{z}(k)\hat{\boldsymbol{z}}_j[k\hat{\boldsymbol{x}}_{oj}(k|k)], S_j[k; \boldsymbol{P}_{oj}(k|k)]]\overline{c_j}}{c} \quad (4.31)$$

Bölüm 4.2'de IMM algoritması için Denklem (4.2)'de belirtilen karıştırılmış mod olasılık denklemindeki $\mu_{k-1}(j)$ yerine SIMM, DIMM ve MIMM algoritmalarında sırasıyla a_j^{k-1}, B_j^{k-1} ve Ω_j^{k-1} kullanılmaktadır. Bu parametrelerin nasıl hesaplandığı ise alt başlıklar altında anlatılmaktadır.

- SIMM:

Mod olasılıklarının ağırlık güncellemesinde olasılık hesabı yerine skaler değerler kullanılmaktadır ve Denklem (4.2)'de kullanılan $\mu_{k-1}(j)$ parametresi a_j^{k-1} ile temsil etmektedir ve hesaplama denklemi Denklem (4.32)'de verilmektedir. Denklemdeki *N* mod sayısını, $\boldsymbol{P}$ ise IMM algoritmalarında kullanılan Kalman filtresinin çıktısı olan kestirim matrisini belirtmektedir.

$$a_j^k = \left(\sum_{i=1}^{N} \frac{1}{tr\ \boldsymbol{P}_i^k}\right)^{-1} \frac{1}{tr\ \boldsymbol{P}_i^k} \quad (4.32)$$

IMM aşamalarından olan durum tahmin birleşimi için kullanılan ve Denklem (4.15) ile Denklem (4.16)'da verilen filtre çıktılarının hesaplanması denklemleri SIMM algoritma için Denklem (4.33) ve Denklem (4.34)'teki gibidir.

$$\boldsymbol{X}_{k|k} = \sum_{i=1}^{N} a_j^k\ \boldsymbol{X}_{k|k}^i \quad (4.33)$$

$$\boldsymbol{P}_{k|k} = \sum_{i=1}^{N} (a_j^k)^2 P_{k|k}^i \quad (4.34)$$

- DIMM:

Bu yöntem için IMM algoritmasında kullanılan $\mu_{k-1}(j)$ parametresi B_j^{k-1} ile temsil etmektedir ve hesaplama denklemi Denklem (4.35)'te, filtre çıktıları ise Denklem (4.39) ve Denklem (4.40)'ta verilmektedir. B_j^{k-1} denklemindeki *n* durum vektörünün boyutu, *N* ise mod sayısını belirtmektedir.

$$B_j = diag\left[b_{j1}, b_{j2}, \dots, b_{jn}\right], \qquad j = 1, \dots, N \tag{4.35}$$

$$b_i = [b_{1i}, b_{2i}, \dots, b_{Ni}] = \frac{\boldsymbol{e}^T(\boldsymbol{P}^i)^{-1}}{\boldsymbol{e}^T(\boldsymbol{P}^i)^{-1}\boldsymbol{e}} \tag{4.36}$$

$$\boldsymbol{e} = [\mathbf{1}, \dots, \mathbf{1}]_{1xN}^{\boldsymbol{T}} \tag{4.37}$$

$$\boldsymbol{P}^i = \begin{bmatrix} P_1^{(i)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & P_N^{(i)} \end{bmatrix} \tag{4.38}$$

$$\boldsymbol{X}_{k|k} = \sum_{i=1}^{N} B_j^k \, \boldsymbol{X}_{k|k}^i \tag{4.39}$$

$$\boldsymbol{P}_{k|k} = diag[PD_1, PD_2, \dots, PD_N] \tag{4.40}$$

$$PD_i = [\boldsymbol{e}^T(\boldsymbol{P}^i)^{-1}\boldsymbol{e}]^{-1} \tag{4.41}$$

- MIMM:

IMM algoritmasındaki $\mu_{k-1}(j)$ parametresi Ω_j^{k-1} ile temsil etmektedir. *k* anındaki Ω_j^k, Denklem (4.42)'de verilen eşitlik ile hesaplanmaktadır. Eşitlikte yer alan *N* mod sayısını, $P_{k|k}^i$ ise *i* modu için hesaplanan ve Kalman filtresinin çıktısı olan kestirim matrisini temsil etmektedir. Denklem (4.43) ve Denklem (4.44)'te ise MIMM algoritması için kullanılan durum tahmin birleşim hesapları verilmektedir.

$$\Omega_j^k = \left(\sum_{i=1}^{N} {P_{k|k}^i}^{-1} \right)^{-1} P_{k|k}^j \tag{4.42}$$

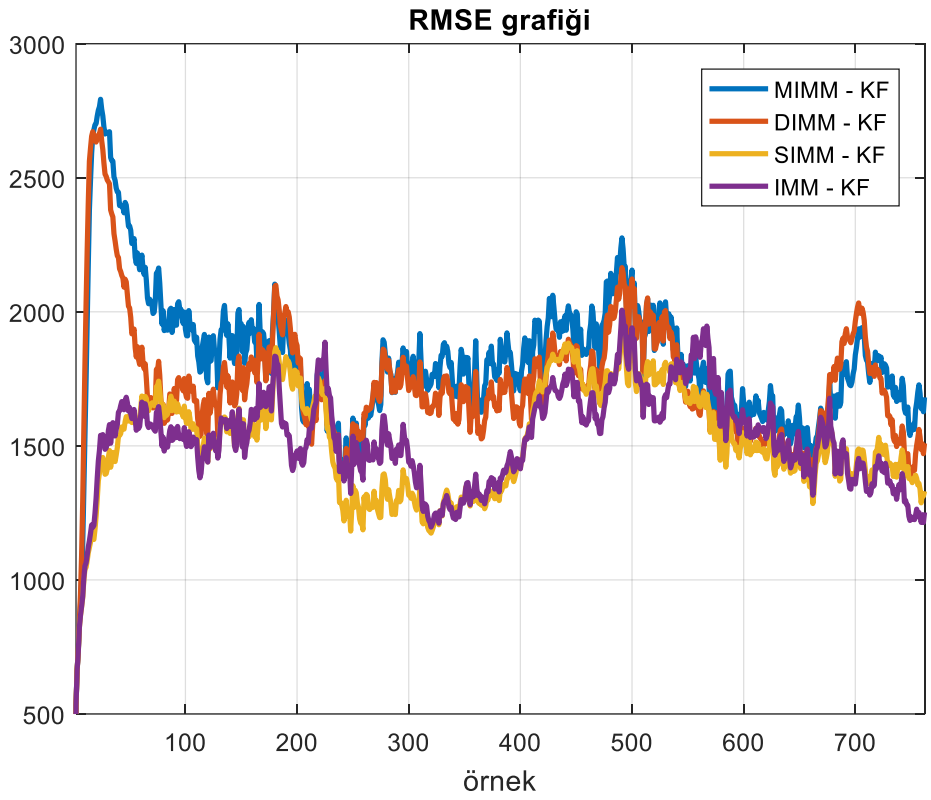
$$\boldsymbol{X}_{k|k} = \sum_{i=1}^{N} \Omega_j^k \, \boldsymbol{X}_{k|k}^i \tag{4.43}$$

$$\boldsymbol{P}_{k|k} = \left(\sum_{i=1}^{N} {P_{k|k}^i}^{-1} \right)^{-1} \tag{4.44}$$

4.6.2. CMKF ve UKF Uyarlamalı SIMM, DIMM ve MIMM Algoritmalarının Performans Karşılaştırmaları

Bu bölümde, ölçümlerin küresel koordinatta alınması sonucunda koordinat dönüşümlerinden kaynaklı sapmalar için kullanılan CMKF algoritması ve UKF algoritması SIMM, DIMM ve MIMM algoritmasına tez kapsamında uyarlanarak önerilen algoritmaların RMSE performansları karşılaştırılmıştır. Bu amaçla, düzenli örnekleme T=1 sn., MC 100 ve Bölüm 4.4'te belirtilen ölçüm gürültü fazla olan senaryoda testler yapılmıştır ve testlerde süreç gürültüsü az ve çok olan iki sabit hız modeli, bir sağa koordineli dönüş ve bir sola koordineli dönüş hareket modeli kullanılmıştır. Kullanılan az süreç gürültüsü *x, y* ve *z* eksenleri için 0.01 iken, çok süreç gürültüsü olarak 100 değeri seçilmiştir. Yapılan testlerde Kalman filtresi, UKF ve CMKF algoritmalarının IMM algoritması ve SIMM, DIMM ve MIMM algoritmalarında kullanılması durumunda takip algoritmalarının RMSE performansları karşılaştırılmıştır. Ayrıca 100 MC için gerçekleme süreleri tablolar halinde verilmiştir. Daha sonrasında takip algoritmalarının RMSE performansı en iyi olan algoritma için Kalman filtresi, UKF ve CMKF kullanılması durumundaki takip performansları incelenecektir.

Öncelikle algoritmalarda Kalman filtresi kullanılması durumunda elde edilen takip performanslarına değinilecektir ve IMM-KF, SIMM-KF, DIMM-KF ve MIMM-KF algoritmalarının RMSE performansları karşılaştırılmıştır. Bu algoritmalar için elde edilen RMSE grafikleri Şekil 4.12'de, MC 100 için gerçekleme süreleri ise Tablo 4.9'da verilmektedir.



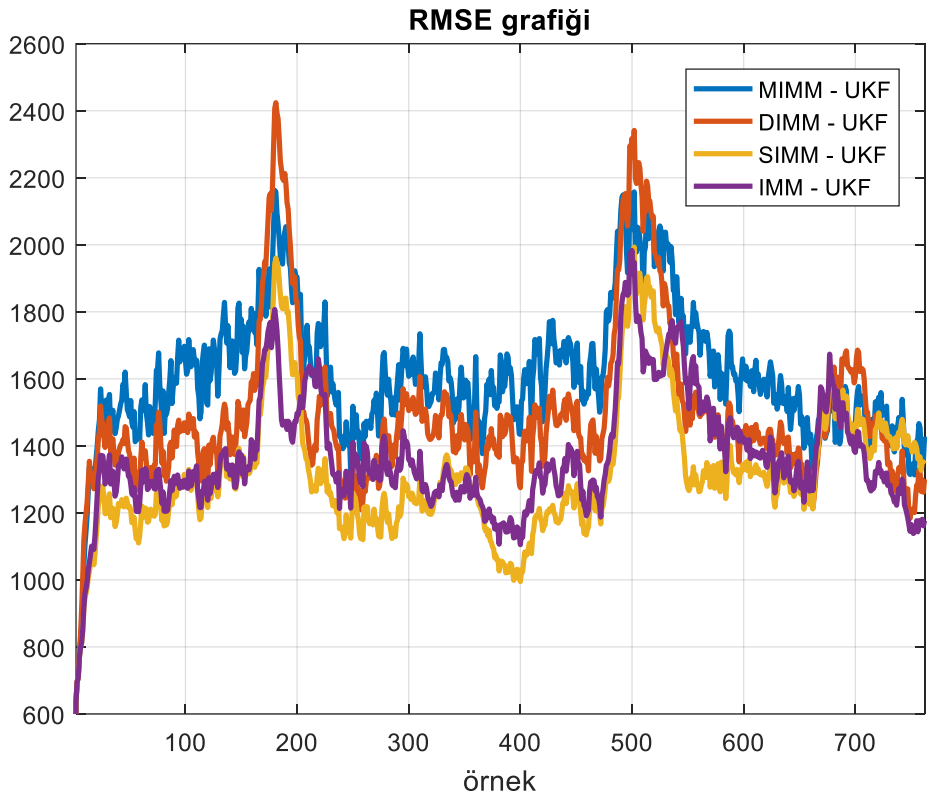
Şekil 4.12: KF için IMM algoritmaları karşılaştırması.

Tablo 4.9: KF için IMM algoritmaları işlem süresi karşılaştırması.

Yöntem	**IMM-KF**	**SIMM-KF**	**DIMM-KF**	**MIMM-KF**
Süre (sn)	27	26	40	37

Şekil 4.12'de Kalman filtresi için elde edilen RMSE grafiklerinden görüldüğü üzere IMM-KF ve SIMM-KF algoritmalarının hata performansları yaklaşık aynıdır ve diğer iki yöntemden daha az hataya sahiptirler. Buna karşın IMM algoritmasında mod ağırlık güncellemesinin yaklaşık hesaplanması göz önünde bulundurulduğunda, Kalman filtresi için SIMM algoritmasının tercih edilerek daha az hataya sahip kestirim elde edilebilir.

Şekil 4.13'te ise, IMM, SIMM, DIMM ve MIMM algoritmalarına üç boyutta tez kapsamında önerilen UKF algoritmasının uyarlanması sonucunda elde edilen takip algoritmalarının RMSE grafikleri verilmiştir. Grafikte de görüldüğü üzere SIMM-UKF algoritması IMM-UKF algoritmasından daha az hataya sahiptir.

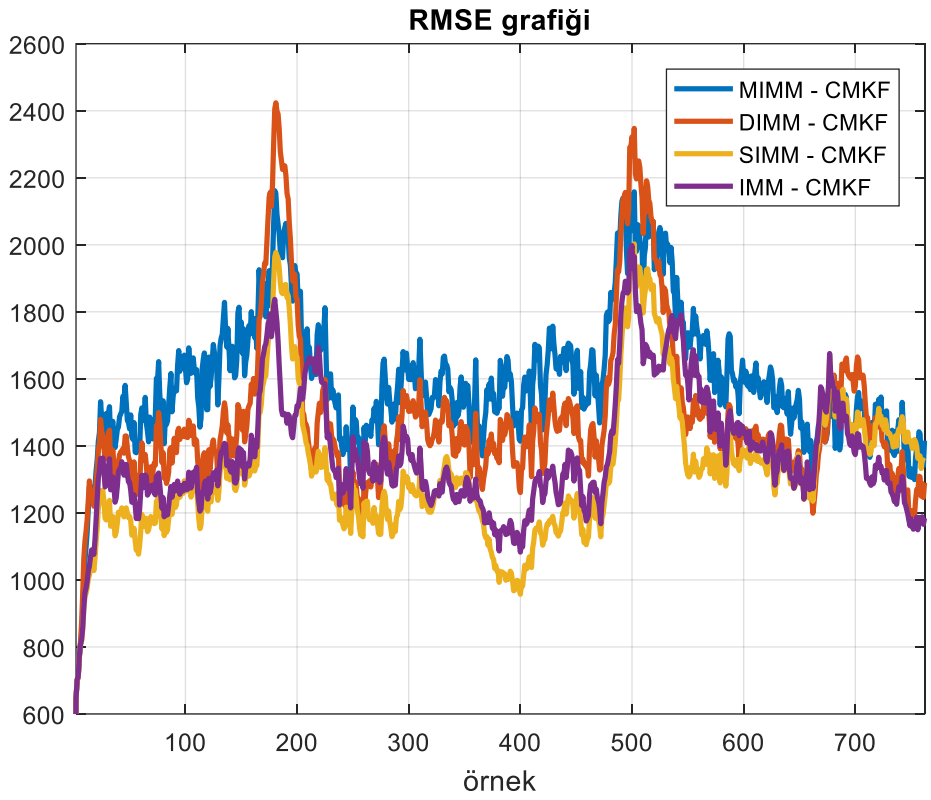


Şekil 4.13: UKF için IMM algoritmaları karşılaştırması.

Tablo 4.10: UKF için IMM algoritmaları işlem süresi karşılaştırması.

Yöntem	**IMM-UKF**	**SIMM-UKF**	**DIMM-UKF**	**MIMM-UKF**
Süre (sn)	80	79	93	90

Son olarak ise, ölçümlerin küresel koordinatta alınması sonucunda koordinat dönüşümlerinden kaynaklı sapmalar için kullanılan CMKF algoritması SIMM, DIMM ve MIMM algoritmasına tez kapsamında uyarlanarak önerilen algoritmaların RMSE performansları karşılaştırılmıştır. RMSE karşılaştırma grafiği Şekil 4.14'te verilmektedir ve UKF yöntemi için yapılan RMSE karşılaştırmasında olduğu gibi CMKF algoritması içinde SIMM yöntemi IMM, DIMM ve MIMM yöntemlerinden daha az hataya sahip olduğu gösterilmiştir.



Şekil 4.14: CMKF için IMM algoritmaları karşılaştırması.

Tablo 4.11: CMKF için IMM algoritmaları işlem süresi karşılaştırması.

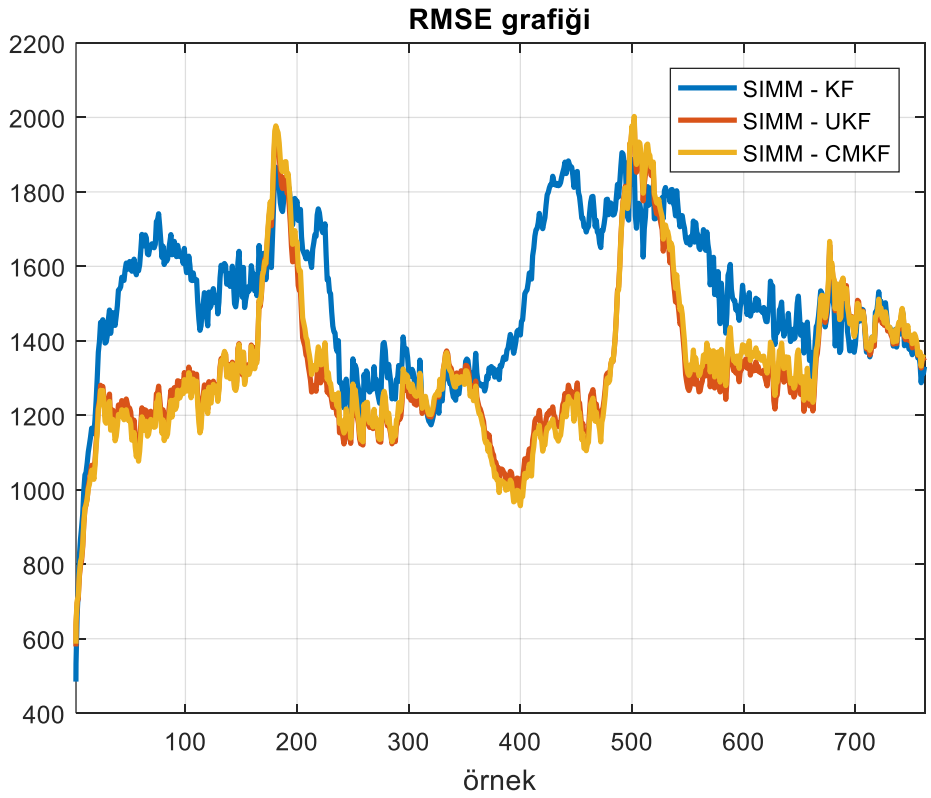
Yöntem	**IMM-CMKF**	**SIMM-CMKF**	**DIMM-CMKF**	**MIMM-CMKF**
Süre (sn)	30	28	43	40

Karşılaştırmalardan görüldüğü üzere SIMM algoritmasının performansı, IMM algoritmasının performansından daha iyidir. Ayrıca IMM algoritmasındaki filtre ağırlıkları hesaplanırken kümülatif olasılığın [0-1] aralığında, olasılık yoğunluk fonksiyonunun ise herhangi bir sınırının olmaması çelişkisi dezvantajına karşı, SIMM algoritmasında filtre ağırlık hesabında olasılık fonksiyonun hesaplanmaması avantajının yanısıra işlem yükü olarak IMM algoritmasından daha az işlem yüküne sahip olması nedeniyle SIMM algoritmasının tercih edilmiştir. SIMM algoritması için ise, SIMM-KF ve bu tez kapsamında önerilen SIMM-UKF ve SIMM-CMKF algoritmalarının performansları Şekil 4.15'te verilmiştir. Şekilde de görüldüğü üzere

SIMM-CMKF filtresi diğer iki filtreden daha az hataya sahiptir ve CMKF algoritmasının UKF algoritmasından daha az işlem karmaşasına sahip olması nedeniyle SIMM-CMKF algoritması tercih edilmiştir. Dolayısıyla SIMM-CMKF algoritması hem IMM-CMKF, DIMM-CMKF, MIMM-CMKF algoritmaları içerisinde hem de KF ve UKF kullanılan IMM, SIMM, DIMM ve MIMM algoritmaları içerisinde en az hataya sahiptir. Bu nedenle üç boyutlu sistem için IMM, Kalman filtresi ve UKF algoritmaları açısından, SIMM-CMKF algoritmasının tercih edilmesinin daha az hataya sahip kestirim üreteceği açıkça gösterilmektedir.

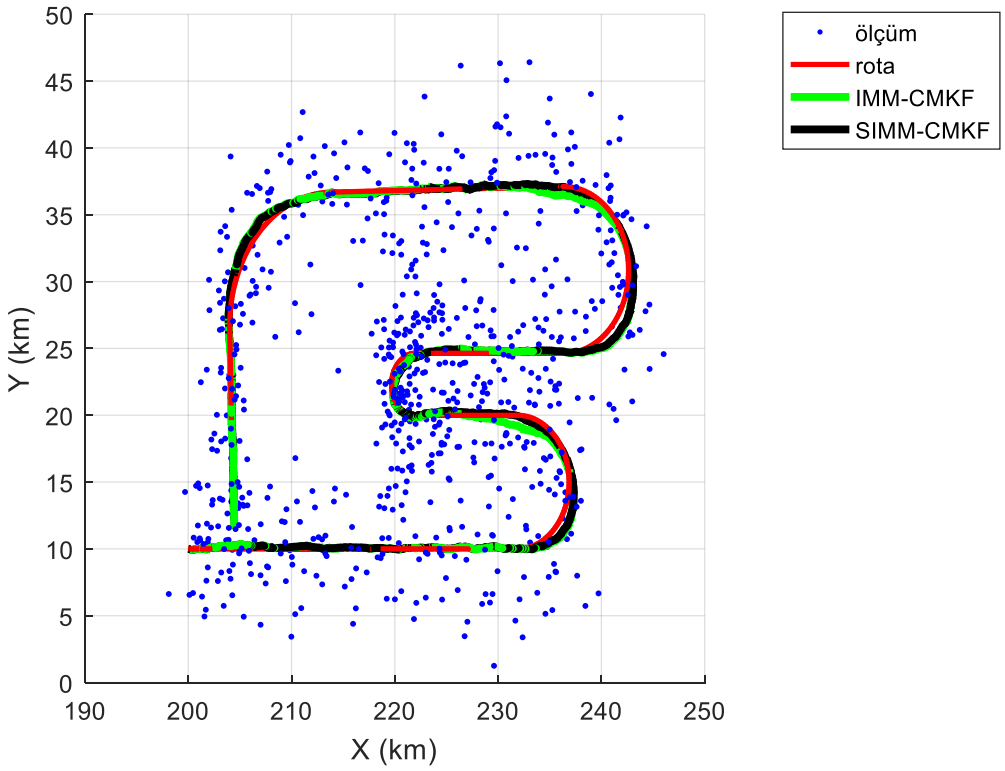
Tablo 4.12: KF, UKF ve CMKF için SIMM algoritmaları işlem süresi karşılaştırması.

Yöntem	**SIMM-KF**	**SIMM-UKF**	**SIMM-CMKF**
Süre (sn)	26	79	28



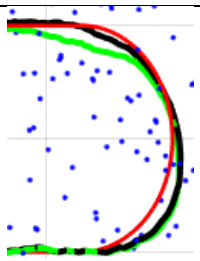
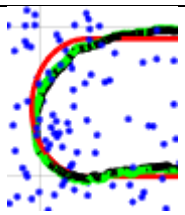
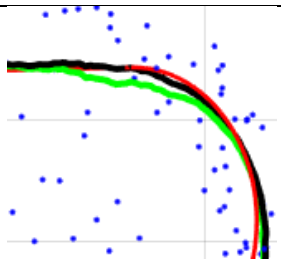
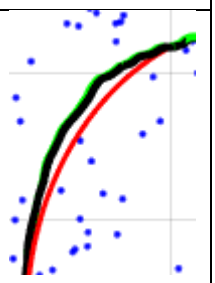
Şekil 4.15: KF, UKF ve CMKF için SIMM algoritmaları karşılaştırması.

Tezin bu bölümünde ve Bölüm 4.4'te yapılan testler kapsamında CMKF algoritmasının Kalman filtresi ve UKF algoritmasından daha az hataya sahip kestirim sonucu ürettiği gösterilmiştir. Ayrıca bu bölümde, SIMM algoritmasının IMM algoritmasındaki olasılık fonksiyon hesabı dezavantajına olmaması ve daha az hataya sahip olduğu gösterilmiştir. Bu nedenle, IMM-CMKF ve SIMM-CMKF algoritmalarının 100 MC değeri için ortalama kestirim grafikleri Şekil 4.16'da, hedefin manevra yapması durumundaki ayrıntılı grafikler ise Tablo 4.13'te verilmektedir. Tablodaki her manevra için IMM-CMKF ve SIMM-CMKF algoritmalarının güzergahları karşılaştırıldığında SIMM-CMKF algoritmasının hedefin asıl güzergahına daha yakın sonuçlar ürettiği görülmektedir. Bu durum özellikle 1. koordineli dönüşte ve 3. koordineli dönüşte açıkça görülmektedir.



Şekil 4.16: CMKF için IMM ve SIMM algoritmaları ortalama kestirim grafiği.

Tablo 4.13: CMKF için IMM ve SIMM algoritmaları ortalama kestirim sonucu manevra ayrıntı grafikleri.

CT numarası	**1**	**2**	**3**	**4**
Dönüş grafiği				

Sonuç olarak bu tez kapsamında öne sürülen SIMM-CMKF algoritması hata miktarının az olması, IMM algoritmasındaki mod ağırlık güncellemesi için gereken olasılık hesabının yapılmaması ve yaklaşık olasılık kullanmaması avantajları nedeniyle tez kapsamında değinilen diğer yöntemler yerine tercih edilmiştir.

5.ÇOKLU HEDEF TAKİBİ

Bu bölümde, öncelikle çoklu hedef takibi hakkında genel bilgiler ve hedef takibindeki mevcut aşamalar verilmiştir. Sonrasında ise Bölüm 5.2'de düzensiz örnekleme problemi ele alınmıştır. Hedef tespitinin düzensiz aralıklarla yapıldığı durumlar da karşımıza çıkmaktadır. Bölüm 4'te ele alınan izleme algoritmaları ölçümlerin düzenli aralıklarla alınması durumunda geçerlidir. Bu bölümde, ilgili algoritmalar periyodik olmayan ölçüm aralıkları için yeniden tanımlanmıştır. Bölüm 5.3'te ise hatalı iz azaltımı problemi ele alınmıştır. Ayrıca Bölüm 5.1'de değinilen hedef takibindeki aşamalarda karşılaşılan ve takip performansını olumsuz etkileyen durumlardan bazıları üzerinde durularak Bölüm 4.3'te verilen takip algoritmalarının bu durumlardaki takip performansları incelenmiştir.

Bu tezin literatüre katkısı kapsamında gerçekleştirilen düzensiz örnekleme durumu Bölüm 5.2'de, düzensiz örnekleme durumunda takip performanslarının incelenmesi Bölüm 5.4.1'de, hatalı izlerin azaltılması problemi ise doğrulamaya dayalı geliştirilen iki yöntem ile ele alınarak takip performansına katkıları üzerinde durulmaktadır.

5.1. Genel Bilgi

Çoklu hedef takibi hava savunması, navigasyon, hava trafik kontrolü, robotik, görüntü işleme, uzay gibi hem askeri hem de sivil alanlarda birçok uygulamaya sahiptir. Hedefler ilgilenilen takip uzayında rastgele ortaya çıkarak bir süre var olur ve takip uzayını terk ederler. Hedeflerin bu davranışı için takip uzayında bulundukları sürede hedeflere iz (*ing.* track) denilmektedir. Çoklu hedef takip sistemlerinde hedeflere ait konumlar radar sistemleriyle ölçüm hatası ile algılanmaktadır. Birçok takip yöntemi, önceki bölümlerde ele alındığı üzere, hedeflerin ilk durumlarının kestirilmesine, mevcut hedef sayısına ve kestirimi yapılan izlerin onaylanmasına ihtiyaç duymaktadır [54]. Çoklu hedef takibi için genel olarak şu adımlar mevcuttur;

- İz başlatma (*ing.* Track Initiation)
- İz sürdürme (*ing.* Track)

- Takip
- İz birleştirme (*ing.* Track Association)

Literatürde bu adımların her biri için farklı algoritmalar mevcuttur. Aşağıda bu adımlar hakkında genel bilgi ve adımlarda kullanılan algoritmalara örnekler verilmektedir.

- İz başlatma:

Çoklu hedef takibinin önemli kısımlarından biridir. Hedeflere ait konum bilgilerinin ölçüm hatasına sahip olması, *k* anında alınan ölçümlerin hangi hedefe ait olduğunun belirlenmesini zorlaştırmaktadır. Bunun yanında, bir sonraki adım olan iz sürdürme adımını desteklemek için başlangıçta alınan ölçümlerden hedeflerin ilk durumlarının kestirilmesi gerekmektedir. İz başlatma genel olarak sıralı ve yığılımlı yöntemler olmak üzere ikiye ayrılmaktadır [55], [56]. İz başlatma yöntemlerinden sıralı yöntem olan ve radar ve sonar takip sistemlerinde yaygın olarak kullanılan kural tabanlı ve mantık tabanlı yöntemler [57], yığılımlı yöntemlerden olan ve yoğun parazit yankı ortamındaki takip ve görüntü işlemede kullanılan Hough dönüşümü tekniği [58]-[60] ve Modifiye Hough dönüşümü tekniği [61] yöntemleri en yaygın kullanılan iz başlatma yöntemlerindendir [55]. Sıralı yöntemlere Sıralı Olasılık Oranı Testi (SPRT, *ing.* Sequential Probability Ratio Test) algoritması örnek olarak verilebilir [62]. Bir diğer iz başlatma yöntemi ise Entegre Olasılıksal Veri İlişkilendirme (IPDA, ing Integrated Probabilistic Data Association) yöntemidir [63].

- İz sürdürme/ güncelleme:

Çoklu hedef takibi adımlarından bir diğeri ise elde edilen izlerin sürdürülmesidir. İz güncellemesi için genel olarak kullanılan iki yöntem vardır ve bu iz güncelleme yöntemleri sensörlerin eş zamanlı olduğu durumlarda kullanılır [11];

- Sıralı güncelleme: İz güncellemesi her bir ölçüm geldiğinde bir sensöre ait ölçüm ile yapılmaktadır.
- Paralel güncelleme: Bu iz güncellemesinde *k* anında her sensörden alınan ölçümler biriktirilerek eş zamanlı bir güncelleme yapılmaktadır.

- Takip:

Bu adımda, elde edilen ve güncellenen izlerin takibi yapılmaktadır. Farklı uygulamalar ve farklı ihtiyaçlar için Bölüm 2.2'de değinilen Kalman filtresi, Genişletilmiş Kalman filtresi, Kokusuz Kalman filtresi, Dönüştürülmüş Ölçümler Kalman filtresi gibi farklı takip algoritmaları kullanılabilmektedir.

- İz birleştirme:

Çoklu hedef takibinin diğer bir önemli adımı ise iz birleştirme adımıdır. Yoğun parazit yankı ortamlarında oluşturulan iz sayısı, yoğun olmayan parazit yankı ortamındaki iz sayısından daha fazla olabilmektedir. Çünkü yoğun parazit yankı ortamında hatalı izlerin sayısı daha fazla olmaktadır. Bu durumda tek bir hedef için birden fazla iz oluşturulması olası bir durumdur. İz birleştirme adımı iki ayrı izin aynı hedeften kaynaklandığı kararının verilmesi ve sonraki aşamada bu izlerin birleştirilmesi adımıdır.

Çoklu hedef takibinde karşılaşılan ve sorun oluşturan bazı durumlar vardır. Bunlar;

– Sapma problemleri: Bunlar küresel koordinatta alınan ölçümlerin kartezyen koordinata dönüştürülmesi sonucu oluşan koordinat dönüşümü sapması ve sensörlerden kaynaklı olan sensör sapması olmak üzere iki grupta ele alınmaktadır. Sensör sapmalarına Bölüm 2.3'te, koordinat dönüşümlerinden kaynaklı sapmalara ise Bölüm 2.4'te değinilmektedir. Koordinat dönüşümlerinden kaynaklı sapma probleminin giderilmesine yönelik yöntemler ayrıntılı olarak Bölüm 3'te verilmektedir.

– Hedefin manevra yapması durumu: Hedefin manevra yapması durumunda kullanılan hedef hareket modelinin uygun olmaması veya kullanılan filtrenin manevra yapan hedef durumu için uygun olmaması çoklu hedef takip performansının olumsuz etkilenmesine neden olmaktadır. Hedef hareket modeli manevra yapan durumda kullanılan model olarak seçildiğinde ise hedef istikameti doğrusal olması durumunda hedef takip performansı düşecektir. Bu nedenle bu gibi durumlarda kullanılan ve tek hareket modeli yerine birden fazla model kullanılabilen IMM yöntemi Bölüm 4'te verilmektedir.

– Düzensiz örnekleme: Hedef takip sistemlerinde ölçümlerin genellikle düzenli aralıklarla alınması beklenmektedir fakat bazı durumlarda ölçüm aralıkları düzensiz

olabilmektedir. Düzenli ve sık olmayan ölçümler takip performansını olumsuz etkilemektedir. Ayrıca kullanılan model denklemleri bu durumda farklılaşmaktadır ve bu durum Bölüm 5.2'de ele alınacaktır.

5.2. Düzensiz Örnekleme

Takip sistemlerinde, alınan ölçümler düzenli olduğu gibi düzensiz de olabilmektedir. Ölçümler düzenli ve ne kadar sık olursa takip algoritmasının performansı daha iyi olmaktadır. Bu bölümde, ölçümlerin düzensiz olması durumu ele alınarak Kalman filtresi, UKF ve CM'nin düzensiz ölçüm aralıkları için uyarlanması ele alınmaktadır.

Ölçümlerin düzenli alındığı düşünülerek Kalman filtresi denklemleri Bölüm 2.2.1'de, UKF algoritmasının denklemleri Bölüm 2.2.3'te, CMKF algoritmaları denklemleri ise Bölüm 3.3.3'te verilmektedir. Düzenli örnekleme için denklemlerde kullanılan ölçümler arası zaman farkı olan T değeri, her ölçüm örneği arasında geçen süre farklı olduğu için düzensiz örneklemede ΔT_k olmaktadır. Bu nedenle, Kalman filtresi ve UKF algoritmasının durum tahmini denklemlerinde kullanılan $\boldsymbol{F}$ ve $\boldsymbol{\Gamma}_k$ zaman bağımlı olduğu için her biri aşağıdaki gibi değişmektedir ve ΔT_k ise Denklem (5.3)'te verildiği gibi hesaplanmaktadır.

$$\boldsymbol{F_k} = \begin{bmatrix} \boldsymbol{I}_3 & (\Delta T_k)\boldsymbol{I}_3 & ({\Delta T_k}^2/2)\boldsymbol{I}_3 \\ 0 & \boldsymbol{I}_3 & (\Delta T_k)\boldsymbol{I}_3 \\ 0 & 0 & \boldsymbol{I}_3 \end{bmatrix} \tag{5.1}$$

$$\boldsymbol{\Gamma}_k = \begin{bmatrix} ({\Delta T_k}^2/2)\boldsymbol{I_3} \\ (\Delta T_k)\boldsymbol{I_3} \end{bmatrix} \tag{5.2}$$

$$\Delta T_k = T_k - T_{k-1} \tag{5.3}$$

CMKF akış diyagramı Şekil 3.1'de verilmiştir ve akış diyagramında görüldüğü üzere diyagramda girdiler kısmında belirtilmemiş ama denklemlerde kullanılan $\boldsymbol{F}_k$, $\boldsymbol{\Gamma}_k$ ve $\boldsymbol{H}_k$ matrisleri de filtreye girdi olarak verilmektedir ve bu matrislerden $\boldsymbol{F}_k$ ve $\boldsymbol{\Gamma}_k$ matrisleri alınan ölçümlerin arasındaki zaman farkına bağlı olarak değişmektedir. Bu nedenle eşit olmayan zaman aralıkları kullanıldığı durum için bu matrislerin her alınan

ölçüm için güncellenerek sistemde öyle kullanılması gerekmektedir. Eşit olmayan zaman aralığı durumu için kullanılan $\boldsymbol{F}_k$ Denklem (5.1)'de ve $\boldsymbol{\Gamma}_k$ ise Denklem (5.2)'de verilmiştir.

Düzensiz örneklenme durumunda hedef takip algoritmalarının performans analizine Bölüm 5.4'te değinilmektedir.

5.3. Hatalı İzlerin Azaltılması

Çoklu hedef takibinde hatalı izlerin oluşması, çoklu hedef takibindeki iz başlatma adımıyla alakalıdır. Temel olarak, bir iz başlatıcı, ne zaman yeni bir hedef belirse, hedefe ait izi başlatabilecek ve hatalı izleri de en az yapacak kapasitede olmalıdır.

İz başlatma yöntemleri genel sıralı ve yığılımlı yöntemler olmak üzere ikiye ayrılmaktadır [55], [56]. Sıralı yöntemlerden kural tabanlı ve mantık tabanlı yöntemlerde ardışık N örnek içeren bir zaman penceresi kullanılır. Bu pencere içerisinde kalan tespit sayısı belirli bir değere ulaştığında iz başarılı şekilde başlatılmış olur. Aksi halde, zaman penceresi bir örnek ileri kaydırılır ve böylece devam eder. Potansiyel izlerin sayısının azaltılması yani hatalı izlerin elenmesi için kural tabanlı iz başlatma yönteminde hız ve ivmelenme kısıtı olarak iki basit kural kullanılırken, mantık tabanlı M'de N iz başlatma algoritması öngörü ve geçitleme işlemini çoklu hipotez tarzında yapmaktadır. Sıralı yöntemlerden olan SPRT yöntemi ise bir skor fonksiyonuna dayandırılarak iz başlatma ve onaylama problemi için olasılıksal bir yaklaşım kullanılmıştır [62]. Bu yaklaşımda, her bir örnek zamanında, her bir aday iz için bir olabilirlik skoru hesaplanarak bu skora dayanan olabilirlik testi uygulanmakta ve elde edilen sonuca göre bir hedefin var olup olmadığına karar verilmektedir.

Yığılımlı yaklaşımlardan olan Hough dönüşümü ve modifiye Hough dönüşümü ile iz başlatma yöntemlerinde, N taramada elde edilen ölçümler aynı anda işlenerek makul hedef güzergâhları belirlenmeye çalışılır. N taramalık veri kısa zaman aralıklarında incelendiğinde hedef güzergahları için elde edilen eğriler (genelde düz çizgiler) için hedef izleri başlatılmaktadır. Yığılımlı yaklaşımlardan olan özyinelemeli RANSAC algoritması ise [64]'te önerilmektedir.

Gözlem uzayında yoğun parazit yankı bulunması durumunda, gerçek hedeflere ait izlerin tespit edilebilirliği önemli oranda düşecektir. Bunun yanında, gerçek hedef

izinin tespit edilebildiği durumlarda ise, yanlış başlatılan izlerin sayısı artacaktır. Bu durumda radar sistemi çok sayıda hedefi izlemek zorunda kalacak, ayrıca radar ekranını takip eden operatör için çok sayıdaki ize göre karar vermek güçleşecektir. Bu sebeple, yoğun parazit yankı bulunan ortamlarda hatalı başlatılan izlerin sayısının azaltılması, ideal durumda tamamen yok edilmesi gerekmektedir. Bu çalışmada, iz doğrulama algoritması ile hatalı izlerin sayısını azaltarak iz başlatma algoritmasının, dolayısıyla takip sisteminin performansını artırmaya yönelik bir yaklaşım önerilmektedir. Önerilen yaklaşım ile gerçek hedeflere ait izler hatasız olarak oluşturulmakta, bunun yanında, parazit yankıdan kaynaklı oluşturulan hatalı izlerin sayısı önemli oranda azaltılmaktadır. Önerilen yöntem gerçek zamanlı olarak çalışabilmekte ve hatalı izleri kısa süre içerisinde yok edebilmektedir. Bu yöntemde iz başlatma adımı için SPRT algoritması kullanılmıştır. Bu algoritmanın kullanılmasının temel sebebi, skor tabanlı yöntemde parametre uzayının geniş olmasıdır. Böylece farklı gereksinimler için kolayca uyumlaştırılabilmektedir. Bunun yanında hatalı izlerin azaltılması için önerilen yöntemdeki iz başlatma aşaması tamamen modülerdir ve SPRT algoritmasından başka bir yöntemde kullanılabilmektedir.

Gözlem uzayında yoğun parazit yankı bulunması durumunda iz başlatma algoritmasının performansı başlatılan hatalı iz sayısı arttıkça düşmektedir. Bu çalışmada, doğru tespitine dayalı iki iz doğrulama önerilmekte ve bu doğrulama algoritmaları ile iz başlatma aşamasından sonra elde edilen hatalı izler doğrulama aşamasıyla elenmektedir. Doğrulama aşaması için Hough dönüşümü [65] ve RANSAC algoritmasından [64] yararlanılmıştır. Hough dönüşümü için her bir ölçüm, $\rho - \theta$ düzleminde temsil edilmekte ve doğrudan sapma derecesi ölçülmektedir. Hedefin doğrusal hareket etmediği durumlarda sapma derecesi artacaktır. Yalnızca az sayıdaki ölçüm setlerinde, yani kısa zaman aralıklarında hedef hareketi doğrusal harekete yaklaşacaktır. Diğer yansan, parazit yankıdan kaynaklı hatalı izler bu esneklikten kaynaklanarak oluşabilecektir. Dolayısıyla Hough dönüşümü ve RANSAC algoritmasının, yoğun parazit yankı durumunda kullanılabilmesi başka bir iz başlatma algoritmasının ürettiği hatalı izlerin tespit edilip yok edilmesi şeklinde kullanılabilir. Temel iz başlatma algoritması sonucunda elde edilen potansiyel izler, K örneklik kümeler halinde istiflenerek, *iz doğrulama* algoritmasına verilmektedir. İz doğrulama algoritması, her bir K örnek uzunluklu potansiyel iz parçasını $\rho - \theta$ düzlemine aktararak doğrudan sapma değerini hesaplamakta ve elde edilen $\Delta\rho$ sapma

değerine göre izin gerçek hedeften ya da hatalı izden kaynaklı olduğuna karar verilmektedir. Şekil 5.1'de Hough ve RANSAC algoritmaları kullanılan iz doğrulama yöntemine ait sözde kod verilmektedir.

Eşik değeri Th_ρ belirle;
skor = 0;
j = 1,2,3 için
- Skor tabanlı iz başlatma algoritması ile *K* iterasyon git
- *K* örnek uzunluklu her bir potansiyel iz parçası *i* için;
  - *Hough dönüşümü/RANSAC ile $\Delta\rho$* sapma miktarı hesapla
    - *eğer ($\Delta\rho < Th_\rho$)*
      - *skor = skor+1*
    - *Aksi halde*
      - *skor = skor*
    - *Son*
- *Son-j*
- *Eğer (skor > 2)*
  - *Karar: i'nci potansiyel iz gerçek bir hedefe aittir.*
- *Aksi halde*
  - *Karar: i'nci potansiyel iz parçası yanlış alarmdır. i'nci izi sil.*
- *son*

Şekil 5.1: Önerilen iz doğrulama algoritmasına ait sözde kod.

Her *K* örneklik kümeler için $\Delta\rho$ sapma değeri Hough dönüşümü kullanıldığında Denklem (5.4)-(5.7)'yi içeren aşağıdaki 5 adımla hesaplanmaktadır. Önerilen yöntem [55]'te belirtilen Hough dönüşümünün hatalı iz tespitine yönelik uyarlanmış halidir. Burada θ, $[0 - \pi]$ aralığında, ρ ise pozitif ve negatif değerler alabilmektedir.

i) θ, eşit $\Delta\theta$' lık N_θ parçaya ayrılır.

$$\Delta\theta = \pi/N_\theta \tag{5.4}$$

ii) Hesaplanan $\Delta\theta$ aralığı ile θ_n değerleri hesaplanır.

$$\theta_n = n * \Delta\theta\ , \quad n = 0, 1, \dots, N_\theta \tag{5.5}$$

iii) Her bir θ ve her bir *K* kümesine ait *k* örneği için ρ değeri bulunur.

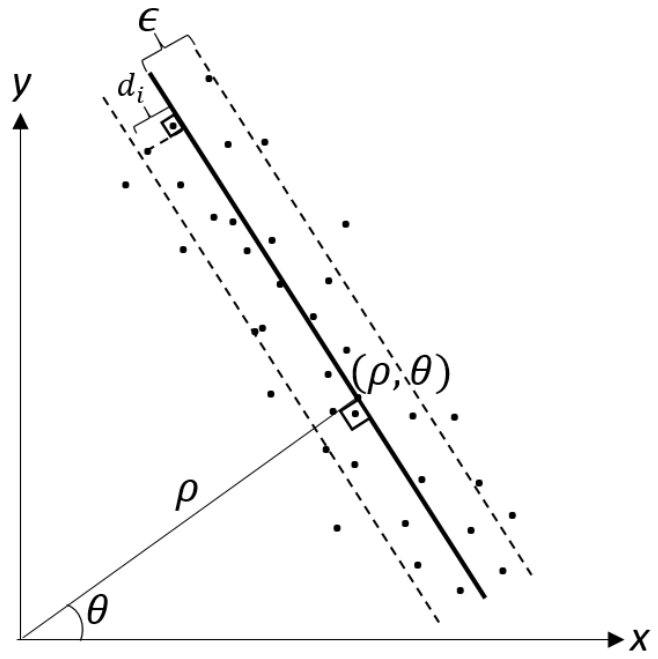
$$\rho_i(\theta_n) = x_i cos\theta_n + y_i sin\theta_n, \quad i = 1,2, \dots, K, n = 0, 1, \dots, N_\theta \tag{5.6}$$

iv) Her bir θ_n değeri için elde edilen ρ_i değerlerinin $\Delta\rho_i$ standart sapma değerleri hesaplanır.

v) Hesaplanan bu sapma değerlerinin en küçüğü önerilen algoritma için kullanılacak olan $\Delta\rho$ değerini vermektedir.

$$\Delta\rho = min\{\Delta\rho_i, i = 1,2,\dots,K\} \quad (5.7)$$

RANSAC algoritması kullanıldığında ϵ ve τ olmak üzere iki parametre optimize edilmektedir. RANSAC algoritması, Şekil 5.2'de belirtilen mesafe ϵ iken, iki kesikli çizgi arasında kalan bölgedeki toplam örnek sayısının K kümesindeki mevcut örnek sayısına oranı τ olduğunda elde edilecek optimum doğrunun parametrelerini (ρ,θ) vermektedir.



Şekil 5.2: *K* kümesi için RANSAC örneği.

RANSAC algoritması için $\Delta\rho$ sapma değeri aşağıdaki iki adımda hesaplanmaktadır.

i) *K* kümesine ait her örnek ile iz doğrultusu arasındaki mesafe (d_i) hesaplanır.

ii) Elde edilen d_i kullanılarak ortalama sapma miktarı belirlenir.

$$\Delta\rho = \sum d_i \ /K \ , \quad i = 1,2, \dots, K \tag{5.8}$$

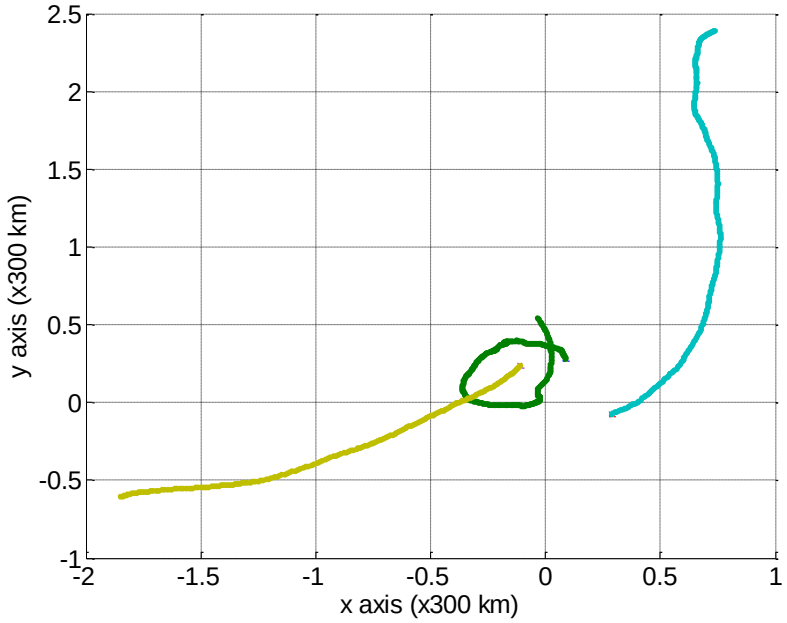
İz doğrulama algoritmasına verilen K uzunluklu potansiyel iz parçaları ne kadar doğrusalsa, $\Delta\rho$ sapma değeri o kadar küçük olmaktadır. Hesaplanan bu sapma miktarı, bir eşik değerinden (Th_ρ) küçükse, potansiyel iz parçası için "gerçek hedefe aittir" sonucuna varılmaktadır.

Ardışıl üç potansiyel iz parçasından en az ikisi için *olumlu* sonuç bulunursa, iz onaylanmaktadır. Aksi durumda, parazit yankıdan kaynaklandığı sonucuna varılarak iz silinmektedir. Böylece, yalnızca tutarlı olanlar gerçek iz olarak kabul edilmektedir. Ardışıl üç potansiyel iz parçası yerine sadece tek bir iz parçasının kullanılması durumunda, örneğin manevra halindeki hedefler için $\Delta\rho$ değeri daha büyük çıkacağından, bu tür hedeflere ait izler kaybedilebilir. Ayrıca, üç aşamalı doğrulama rastgele oluşan parazit yankıdan kaynaklı hatalı izlerin gerçek izlerden ayrışmasını kolaylaştırmaktadır.

Test için simülasyon ortamında, ölçüm periyodu 3 saniye olan her biri 45 dakikalık on farklı senaryo üretilmiştir. Bu senaryoların bazılarında hiç hedef yokken bazılarında ise birden fazla hedef bulunmaktadır. Hedef hareketleri doğrusal, manevralı, ani dönüş gibi hareketleri içermektedir ve hedef hızları yaklaşık olarak [500-800] km/s aralığında alınmıştır. Tablo 5.1'de hedef senaryolarında bulunan hedef sayısı verilmiştir ve Şekil 5.3'te ise S8 senaryosuna ait hedefler görülmektedir.

Tablo 5.1: Senaryolarda bulunan hedef sayısı.

Senaryo Numarası	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**
Hedef Sayısı	-	2	4	3	1	3	2	3	1	3



Şekil 5.3: Senaryo 8 için önerilen yöntem sonucunda oluşan izler.

Tablo 5.2'de önerilen iki yöntem için farklı *K* değerleri için elde edilen sonuçlar verilmektedir. *K* değeri potansiyel izlerin istiflenme uzunluğunu temsil etmektedir. Ayrıca tabloda belirtilen iz sayısı sütununda SPRT yöntemi sonucunda elde edilen potansiyel izlerin sayısı verilmektedir ve bu izler içerisinde hem hedef izi hem de hatalı izler mevcuttur. Tablo 5.1'de verilen hedef sayısı göz önünde bulundurulduğunda hatalı izlerin sayısının oldukça fazla olduğu görülmektedir. Hough dönüşümü ve RANSAC algoritması için ardışıl üç potansiyel izden en az ikisi için algoritmada *olumlu* sonuç elde edilirse izin onaylanarak gerçek hedefe ait olduğuna karar verildiğine değinilmişti. Üçlü doğrulama kullanıldığı için uzunluğu 3*K* uzunluğundan daha büyük olan izlerin sayısı Tablo 5.2'de verilmiştir. Aynı *K* değeri için Hough dönüşümü ve RANSAC algoritması sonucunda elde edilen iz sayısına bakıldığında, *K*=5 seçildiğinde RANSAC algoritması, *K*=10 seçildiğinde ise Hough dönüşüm sonucunda daha iyi sonuç elde edilmiştir. Dolayısıyla bu deneylerden, kısa periyotlarla (küçük *K* değeri için) iz doğrulaması yapılması durumunda RANSAC, daha uzun periyotlarla (büyük *K* değeri için) iz doğrulaması yapıldığı durumlarda ise Hough dönüşümü hatalı izleri elemek için daha uygundur denebilir .

Tablo 5.2: Önerilen yönteme ait simülasyon sonuçları.

Senaryo Bilgisi		**3*K*'dan uzun iz sayısı**		**Hough Dönüşümü**		**RANSAC algoritması**	
Senaryo	**İz sayısı**	***K*=5**	***K*=10**	***K*=5**	***K*=10**	***K*=5**	***K*=10**
S1	6755	1034	120	272	-	180	5
S2	6802	973	108	294	-	207	1
S3	6699	977	97	284	1	198	9
S4	6801	984	104	302	2	188	5
S5	6703	974	99	258	1	159	6
S6	3488	223	17	88	-	64	-
S7	3500	214	13	82	-	58	-
S8	3640	246	8	69	-	58	-
S9	3617	222	11	75	-	51	1
S10	3588	238	16	95	-	63	1

Sonuç olarak hedeflerin dairesel hareket, manevra, ani dönüş yapması durumunda dahi iki algoritma ile izler hatasız olarak tespit edilerek hatalı izlerin büyük bölümü elenebilmektedir. Bunun yanında Hough dönüşümü ve RANSAC algoritması yığılımlı yöntemler olmasına rağmen önerilen bu iki yöntem gerçek zamanlı çoklu hedef takibi uygulamalarında kullanılabilecek kadar hızlıdır. Ayrıca temel iz başlatma algoritmasından bağımsız olması, bu iki yöntemin başka iz başlatma algoritmalarıyla da kullanılabilmesini sağlamaktadır. Dolayısıyla ihtiyaca göre farklı iz başlatma algoritmaları ile kullanılabilir.

5.4. Testler

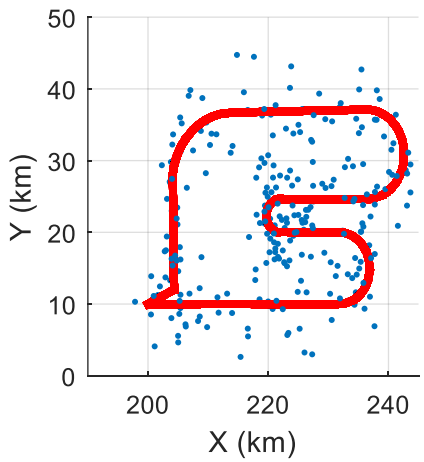
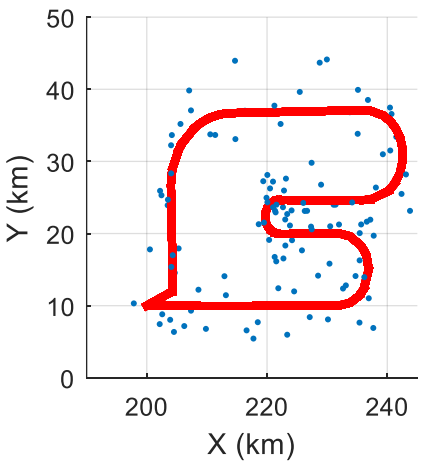
Bu bölümde, Bölüm 4.3'te detayları verilen IMM-CMKF algoritmalarının üç boyutta farklı durumlarda performans analizi yapılmaktadır. Bölüm 4.4'te hedef takip

performansını olumsuz etkileyen durumlardan olan koordinat dönüşümlerinden kaynaklı sapma ve ölçüm gürültüsünün takip performansına etkileri incelenmişti. Bu bölümde ise, hedef takip sistemlerinde karşılaşılan ve takip performansını olumsuz etkileyen düzensiz örnekleme durumu Bölüm 5.4.1'de, sensör sapmasının takip performansı üzerine etkileri ise Bölüm 5.4.2'de incelenecektir. Ayrıca üzerinde durulan IMM-CMKF algoritmaları, hesaplama yükü ve hata miktarı bakımından sık kullanılan IMM-KF ve IMM-UKF algoritmaları ile karşılaştırılmıştır. Performans analizi için üretilen senaryo bilgileri Bölüm 4.4'te verilmiştir. Test senaryosu için üretilen verinin *x-y* koordinat düzlemindeki görüntüsü Şekil 4.2'de, kullanılan ve ölçüm gürültüsü fazla olan senaryo için hedef rotası ve ölçümler ise Tablo 4.1'de verilmektedir. Performans karşılaştırması için Denklem (3.38)'de verilen RMSE değeri kullanılmıştır ve algoritmalar 100 Monte Carlo'da çalıştırılmıştır.

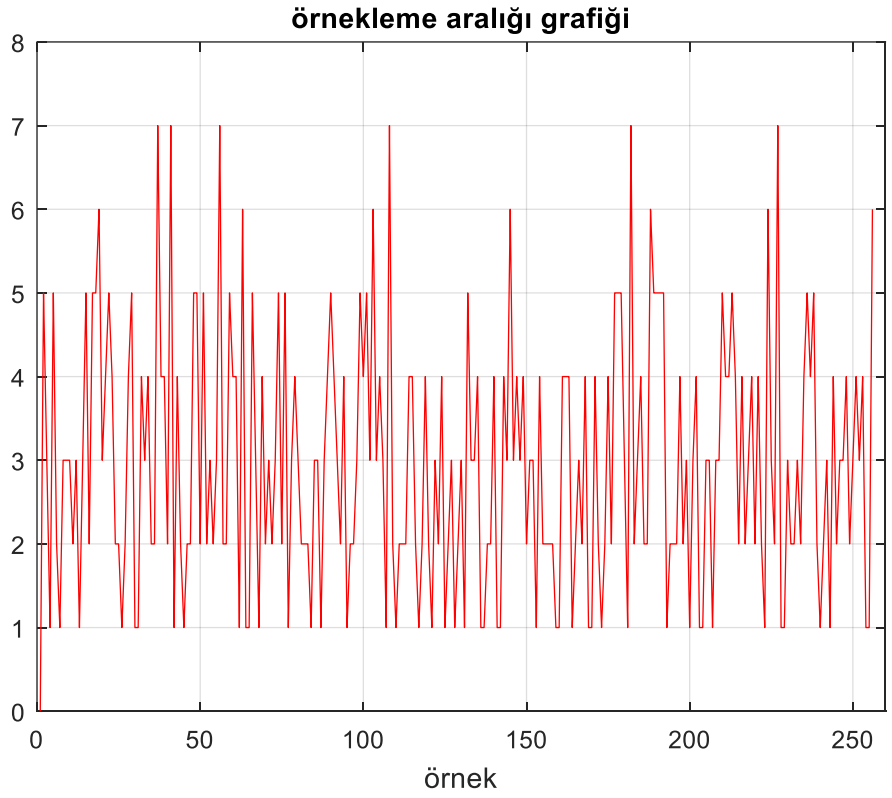
5.4.1. Ölçüm Aralıklarının Düzensiz Olması Durumu

Bu bölümde, öncelikle ölçüm aralıklarının düzensiz olması durumu ele alınarak bu durumun takip performansları üzerindeki etkileri incelenmektedir. Kullanılan senaryoda ölçüm aralıkları, poisson dağılımından rastgele sayı çekilerek oluşturulmuştur ve poisson dağılımın ortalamasını temsil eden farklı λ değerleri ile oluşturulan ölçümlerin örnek grafikleri Tablo 5.3'te verilmiştir.

Tablo 5.3: Farklı $\boldsymbol{\lambda}$ değerleri ile oluşturulan ölçüm grafikleri.

$\boldsymbol{\lambda = 3}$	$\boldsymbol{\lambda = 7}$
	

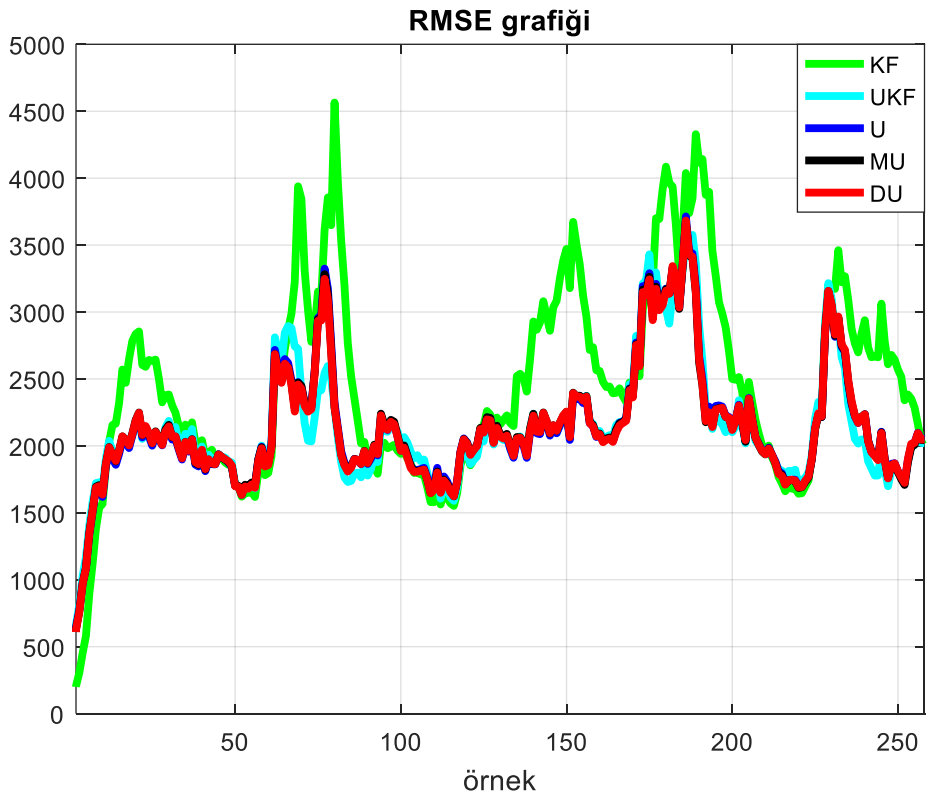
Şekil 5.4'te λ=3 için ölçümlerin alındığı sıklığı belirleyen ΔT değerlerinin grafiği örnek indeksine bağlı olarak verilmiştir. Bu durumda alınan ölçümlerin örnek görüntüsü Tablo 5.3'te, takip performanslarının RMSE grafiği Şekil 5.5'te, algoritmaların gerçekleme süreleri ise Tablo 5.4'te verilmektedir. Düzenli örnekleme durumu Bölüm 4.4'te ele alınmıştı ve Şekil 4.4 ve Şekil 4.8'de görüldüğü üzere düzenli örnekleme durumunda IMM-UKF algoritması IMM-CMKF algoritmalarından daha fazla hataya sahipti. Düzensiz örnekleme durumunda elde edilen Şekil 5.5'te verilen RMSE grafiğine bakıldığında ise IMM-UKF algoritmasının IMM-CMKF algoritmalarına yakın hata değerlerine sahip olduğu görülmektedir. Fakat takip algoritmalarının gerçekleme süreleri göz önüne alındığında IMM-UKF algoritması fazla işlem gerektirdiği için diğer yöntemlere göre daha yavaş kalmıştır. Bu nedenle, IMM-UKF ve IMM-CMKF algoritmaları RMSE açısından benzer performans göstermesine rağmen işlem yükü açısından UKF daha dezavantajlı olduğu için CMKF algoritması tercih edilmektedir.



Şekil 5.4: **λ**=3 için örnekleme aralığı.

Tablo 5.4: λ=3 için takip algoritmalarının gerçekleme süreleri.

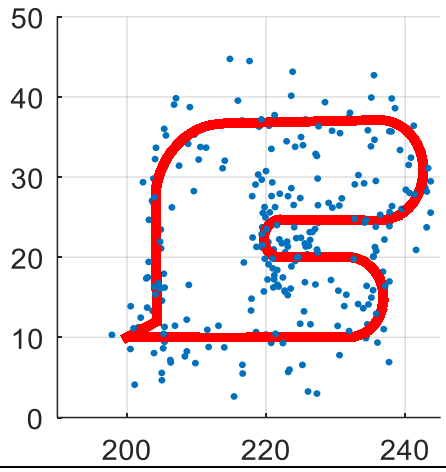
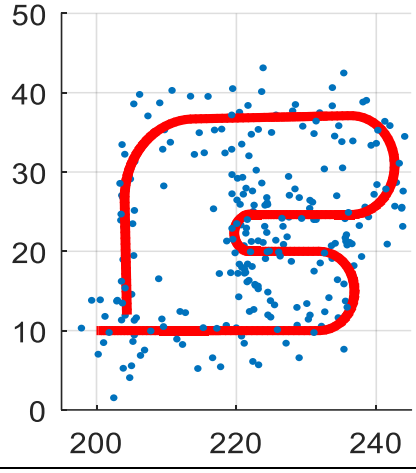
Yöntem	IMM-KF	IMM-UKF	IMM-CMKF_U	IMM-CMKF_MU	IMM-CMKF_DU
Süre (sn)	12	40	12	10	13



Şekil 5.5: λ=3 için takip algoritmalarının RMSE grafiği.

Düzensiz örnekleme λ=3 için takip algoritmalarının RMSE grafiği ile düzenli örnekleme T=3 sn için takip algoritmalarının RMSE grafiğinin verildiği Şekil 4.10 karşılaştırıldığında grafiklerin birbirine çok yakın olduğu görülmektedir ve bu beklenen bir durumdur. Bu durum aynı hedefe ait örneklerin her iki grafik için de yaklaşık 260 örnekten oluşması ile açıklanabilir. λ=3 ortalamalı düzensiz örneklerden elde edilen ölçümler ile T=3 düzenli örneklerden elde edilen bilgiler birbirine yakın bilgilerdir ve bu durum Tablo 5.5'teki örnek ölçüm grafiklerden açıkça görülmektedir. Grafiklerden görüldüğü üzere ölçümlerin dağılımı her iki durum için de yaklaşık aynıdır.

Tablo 5.5: λ=3 ve T=3 için elde edilen örnek ölçüm grafikleri.

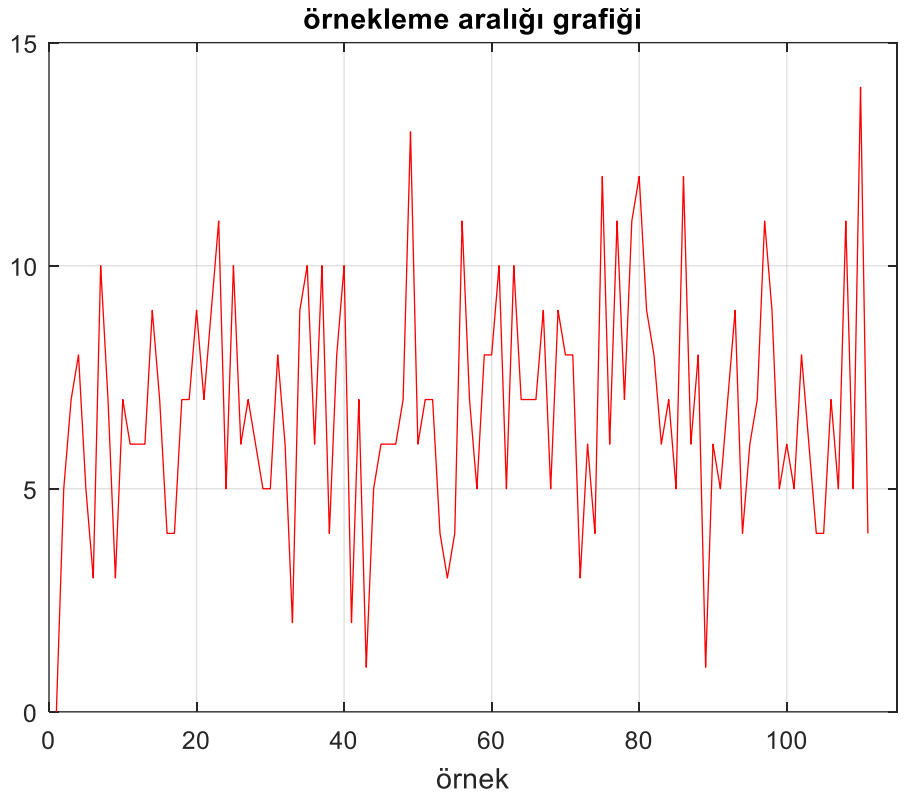
λ=3	T=3
	

Tablo 5.6'da düzensiz ölçüm aralıklarının λ=7 olması durumunda takip algoritmalarının gerçekleme süreleri verilmiştir. Ölçüm aralıklarının artması alınan ölçüm sayısını azaltmaktadır. Bu durum ise gerçekleme süresini azaltmıştır ve takip algoritmalarının gerçekleme süreleri arasındaki farkta azalmaktadır. IMM-KF ve IMM-CMKF algoritmasının hesaplama yükü az olduğu için hızlı, IMM-UKF algoritmasının hesaplama yükü daha fazla olduğu için daha yavaştır.

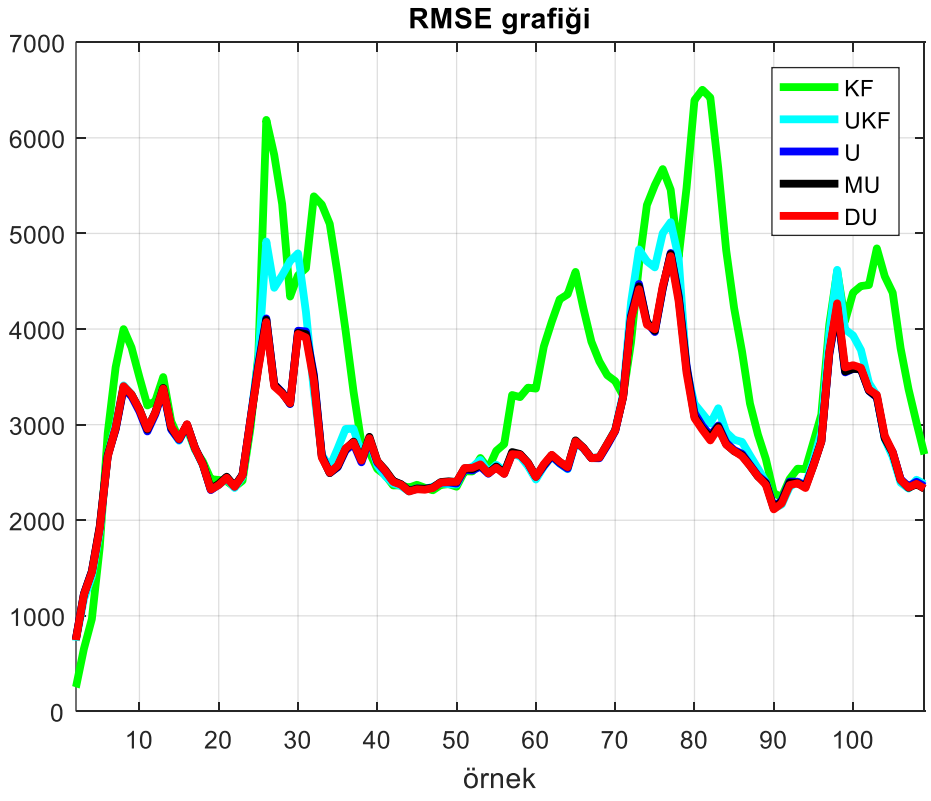
Tablo 5.6: λ=7 için takip algoritmalarının gerçekleme süreleri.

Yöntem	**IMM-KF**	**IMM-UKF**	**IMM-CMKF_U**	**IMM-CMKF_MU**	**IMM-CMKF_DU**
Süre (sn)	5	17	5	5	5

Şekil 5.6'da λ=7 için düzensiz ölçüm aralıkları çizdirilmiştir. Grafikten görüldüğü üzere ölçüm aralıkları ortalama 7 sn., minimum 1 sn. ve maksimum 14 sn. olacak şekilde oluşturulmuştur. Bu durumda elde edilen ölçümler kullanılarak takip algoritmalarının performanslarının karşılaştırılması için RMSE değerleri çizdirilmiş ve çizdirilen değerler Şekil 5.7'de verilmiştir. Hata grafiğine bakıldığında en çok hataya IMM-KF, daha sonrasında IMM-UKF son olarak ise IMM-CMKF algoritmaları sahiptir. Bu nedenle, düzensiz örnekleme için IMM-CMKF algoritmasının diğer algoritmalardan hem işlem yükü hem de performans bakımından daha avantajlı olması nedeniyle IMM-CMKF algoritması tercih edilmiştir.



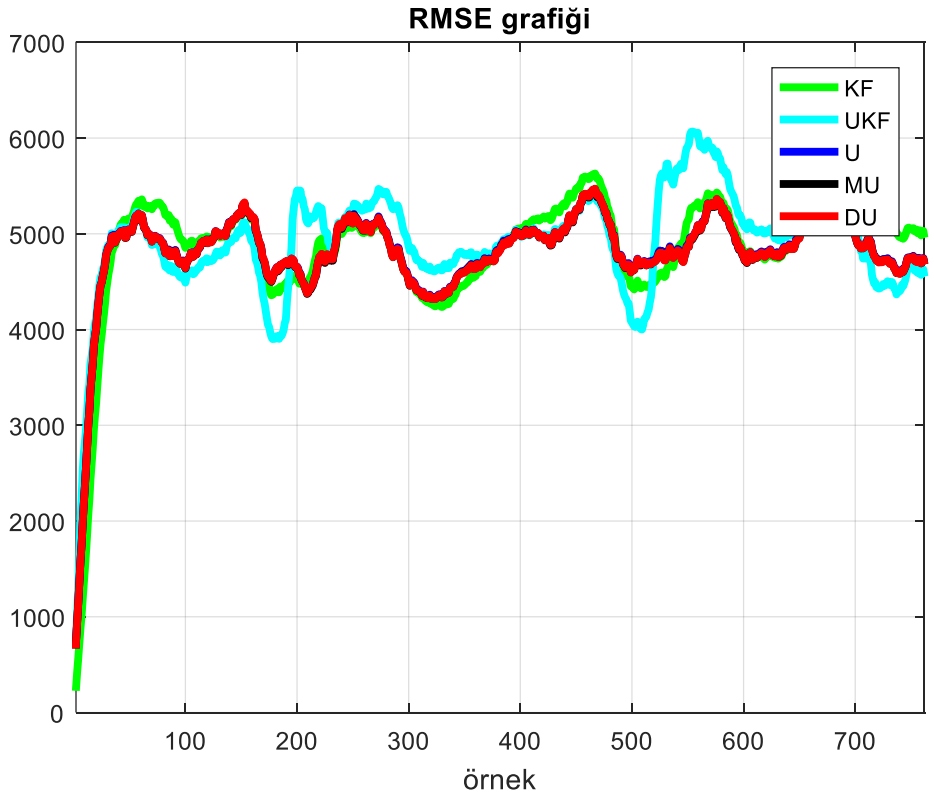
Şekil 5.6: λ=7 için örnekleme aralığı.



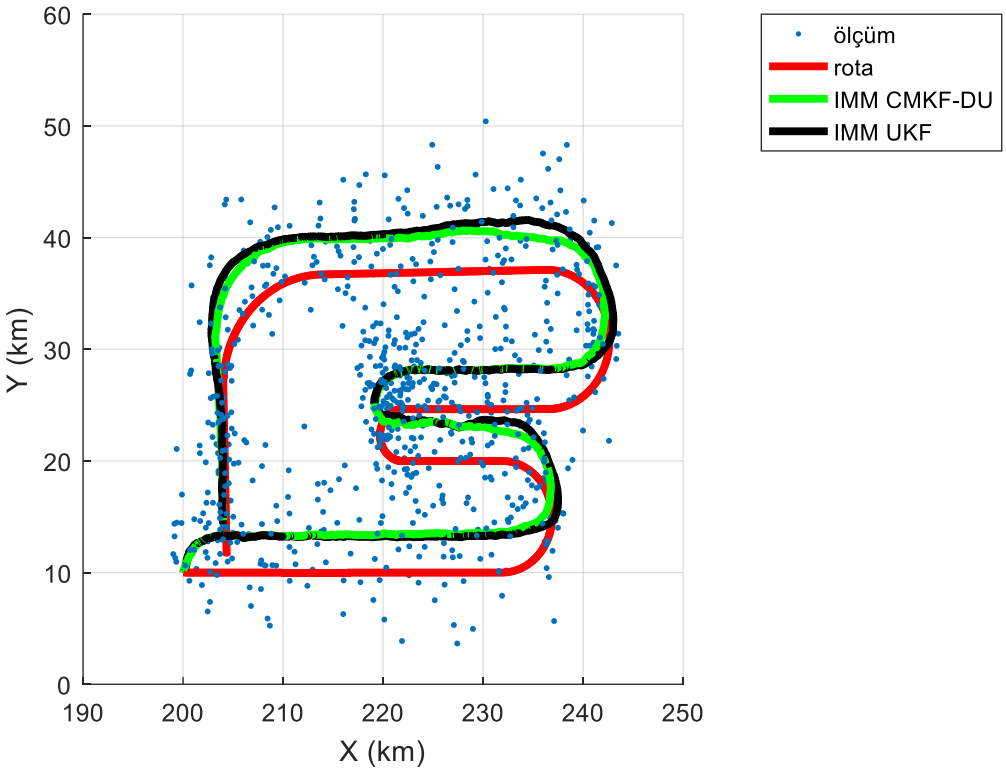
Şekil 5.7: λ=7 için takip algoritmalarının RMSE grafiği.

5.4.2. Sensör Sapması Durumu

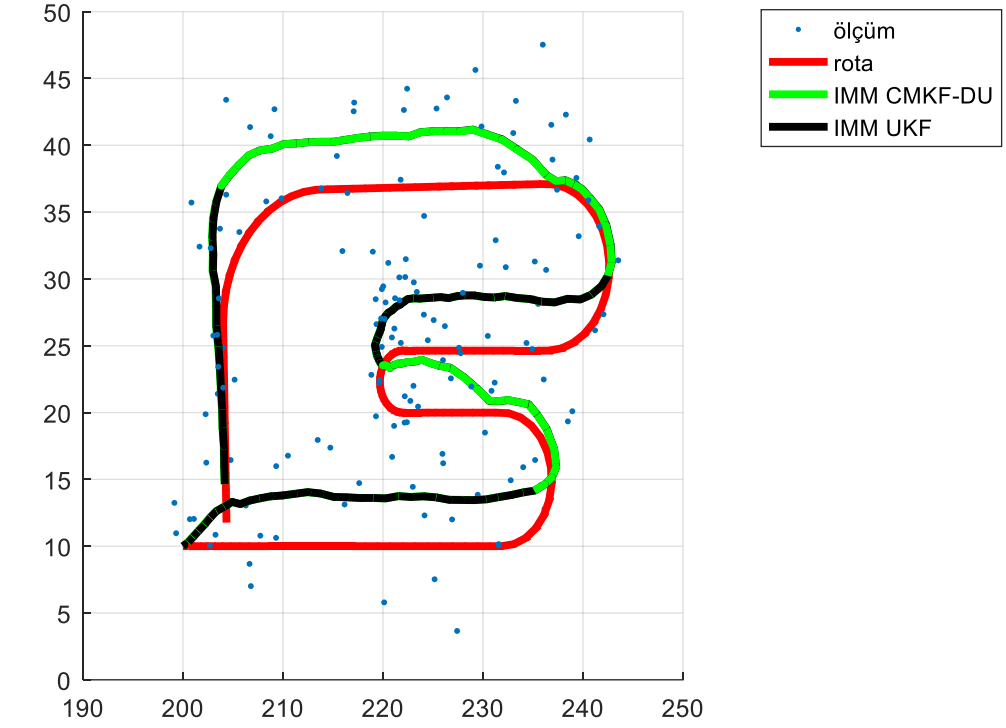
Takip performanslarını etkileyen diğer bir durum ise sensörlerden kaynaklı sapmadır. Sensör sapmaları hakkında genel bilgi ve sensör sapmalarının polar/küresel koordinatta ve kartezyen koordinattaki formları verilmişti. Sensör sapmasının takip performansı üzerindeki etkisini incelemek için kullanılan senaryo üç boyutlu sistemde ifade edilmektedir ve ölçümler küresel koordinatlarda alındığı için sensör sapması için küresel koordinattaki Denklem (2.83)'te verilen form kullanılmıştır. Sensör sapma vektörü olan $\boldsymbol{\beta}_i(k)$ için, b_i^r, b_i^θ, b_i^φ, ϵ_i^r, ϵ_i^θ, ϵ_i^φ sırasıyla 20 m, 0.015 mrad, 0.015 mrad, 10^{-4}, 10^{-3} ve 10^{-3} kullanılmıştır. Sensör sapmalı ölçüm durumundaki takip performansının RMSE grafiği T=1 sn için Şekil 5.8'de verilmektedir. T=1 sn için sensör sapmasız durumundaki takip performansı ise Şekil 4.4'te verilmişti. Bu iki grafik karşılaştırıldığında sensör sapmasının takip performansını olumsuz etkilediği ve RMSE grafiğinde görüldüğü üzere daha fazla hataya sahip olduğu söylenebilir. Sensörler sapmasız iken IMM-CMKF algoritması 1500 metrelik hata oluştururken, sensör sapması durumda hata miktarı 5000 metre olmaktadır. Şekil 5.9'da ise IMM-CMKF_DU ile IMM-UKF algoritmalarının ortalama kestirim sonucu verilmektedir.



Şekil 5.8: Sensör sapma durumunda T=1 sn için RMSE grafiği.

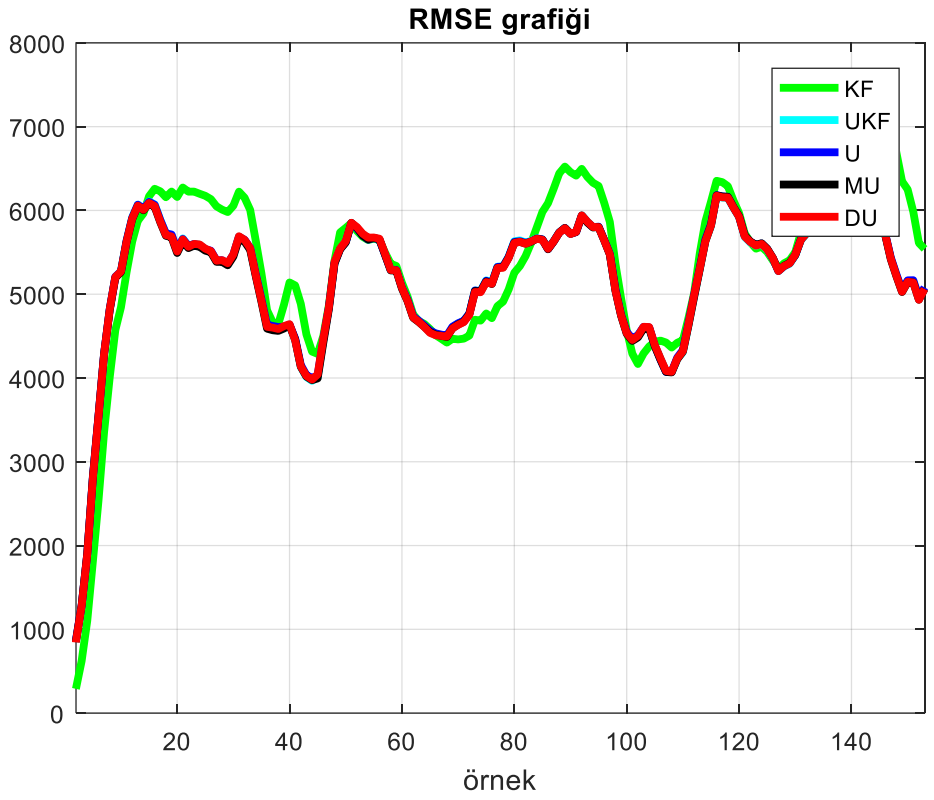


Şekil 5.9: Sensör sapması durumunda IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği (T=1 sn).



Şekil 5.10: Sensör sapması durumunda IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği (T=5 sn).

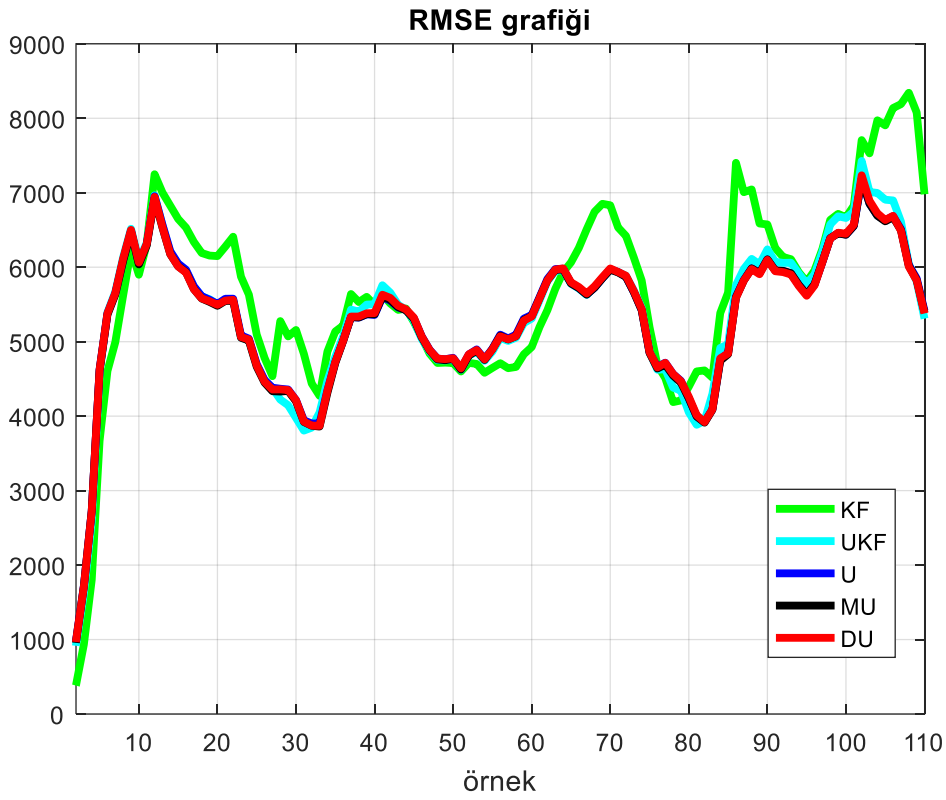
Şekil 5.10'da T=5 sn için elde edilen ortalama kestirim sonucu grafiği verilmiştir ve T=1 sn ile kıyaslandığında, T=5 saniye için ölçümlerin daha seyrek olduğunu ve dolayısıyla takip performansının olumsuz etkilendiği görülmektedir ayrıca RMSE grafikleri karşılaştırıldığında Şekil 5.11'de T=5 sn için elde edilen grafikte hataların daha fazla olduğu ve ölçümlerin seyrekliğinden kaynaklı konum bilgisi eksikliğinden takip performanslarının birbirine yaklaştığı söylenebilir. Fakat CMKF ve UKF algoritmaları Kalman filtresinden daha iyi sonuçlar üretmiştir. Sonuç olarak ölçüm aralıklarının fazla olduğu durumda, Kalman filtresini yerine CMKF veya UKF algoritmalarının kullanılması daha iyi takip performansı üretecektir.



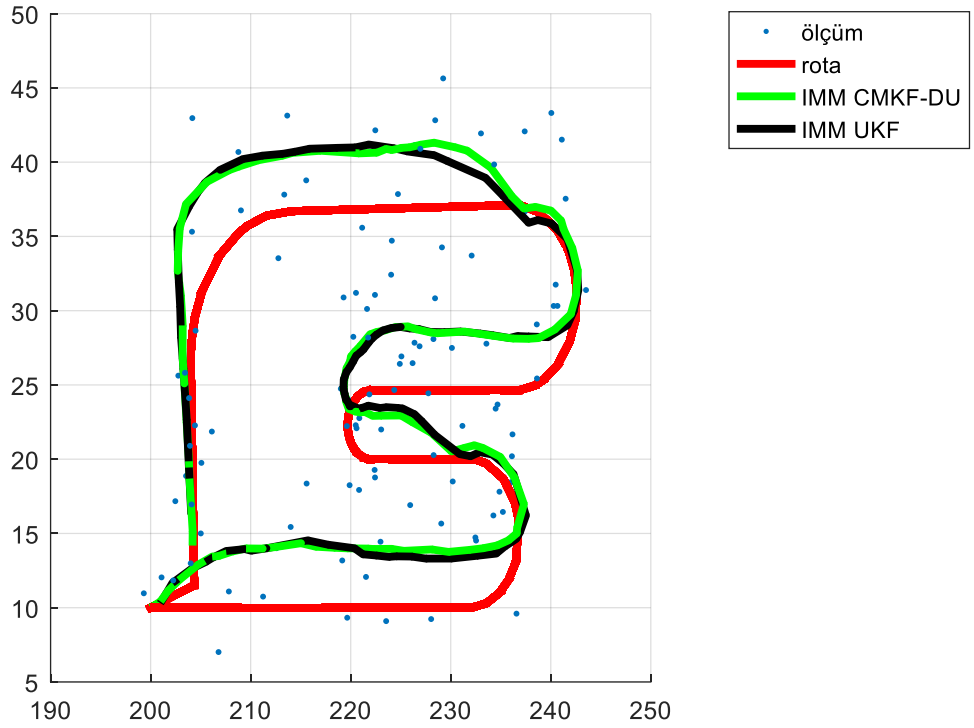
Şekil 5.11: Sensör sapma durumunda T=5 sn için RMSE grafiği.

Bu bölümde sırasıyla ölçümlerin düzensiz alındığı durum ve sensör sapmalı ölçümler için takip performansları ayrı ayrı incelenmiştir. Pratikte bu iki durumun beraber olması söz konusu olabilir bu nedenle hem ölçümlerin düzensiz aralıklarla alındığı hem de sensör sapmasının mevcut olduğu durum incelenmektedir. Aşağıda her iki durumu içeren λ=7 olması durumunda elde edilen ölçümler için takip performanslarının RMSE grafiği ve ortalama kestirim grafiği verilmektedir. Grafikten

görüldüğü üzere hedefe ait ölçüm bilgisinin az olması ve sensör sapmasınında mevcut olması takip performansını olumsuz etkilemiş ve hata değerlerinde artışa sebep olmuştur. Şekil 5.13'te görüldüğü üzere sensör sapması mevcutken ölçüm sıklığının az olması, T=5 için elde edilen ve Şekil 5.10'da verilen şekle benzer sonuç üretmiştir.



Şekil 5.12: Sensör sapma durumunda $\boldsymbol{\lambda}$=7 için RMSE grafiği.



Şekil 5.13: Sensör sapması durumunda IMM-CMKF_DU ve IMM-UKF için ortalama MC kestirim sonucu karşılaştırma grafiği (λ=7).

Sonuç olarak ölçüm aralıkları artıkça incelenen tüm takip algoritmalarının performansında düşüş gözlenmiştir. Bunun yanında ölçümlerin sık alındığı durum için takip algoritmalarının elde edilen performansları kıyaslandığında IMM-CMKF algoritmasının özellikle hedef manevra yaptığında IMM-UKF algoritmasından daha iyi takip performansı verdiği görülmüştür. Fakat bu durum, ölçüm aralıkları seyreldiğinde hedefe ait konum bilgisinin azalmasından kaynaklı olarak takip performanslarındaki farklılıklar azalmış, incelenen takip algoritmalarının performanslarının yaklaşık aynı sonuçları ürettiği gözlenmiştir. Fakat uygulamaya göre kullanılan sistemin gerçek zamanlı çalışabilir olması açısından bakıldığında UKF algoritmasının karmaşık ve fazla işlem gerektirmesine karşın CMKF algoritmasının basit ve az işlem gerektirmesi, gerçek zamanlı uygulamalarda CMKF algoritmasının daha uygun olduğunu göstermektedir.

KAYNAKLAR

[1] Guo Z., Zhou G., Xu R., (2015), "A Gaussian mixture Converted Doppler Measurement Kalman Filter", IET International Radar Conference 2015, 1–6, Hangzhou, China, 14-16 October.

[2] Zhou G., Pelletier M., Kirubarajan T., Quan T.,(2014), "Statically Fused Converted Position and Doppler Measurement Kalman Filters", IEEE Transactions on Aerospace and Electronic Systems, 50(1), 300–318.

[3] Spitzmiller J. N., Adhami R. R., (2009), "A novel data-fusion-based improvement to debiased CMKF tracking", 17th European Signal Processing Conference, 1052–1056, Glasgow,UK, 24-28 August.

[4] Liu Y., Tian X., Xu X., (2017), "Posture estimation system by IMM-based unscented Kalman filters", Chinese Automation Congress, 2363–2368, Jinan,China, 20-22 October.

[5] Yuan T., Krishnan K., Duraisamy B., Maile M., Schwarz T., (2017), "Extended object tracking using IMM approach for a real-world vehicle sensor fusion system", 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, 638–643, Daegu,South Korea, 16-18 November.

[6] Guo J. P., Xu J., Yan L., Xia X., Xiao X., Long T., Bian M., (2016), "An improved IMM algorithm based on maneuvering-adaptive model set", 2016 CIE International Conference on Radar, 1–5, Guangzhou, China, 10-13 October.

[7] Cho C. H., Chong S. Y., Song T. L., (2017), "IMM filtering for vehicle tracking in cluttered environments with glint noise", International Conference on Control, Automation and Information Sciences, 98–105.

[8] Fu X., Jia Y., Du J., Yu F., (2010), "New interacting multiple model algorithms for the tracking of the manoeuvring target [Brief Paper]", IET Control Theory& Applications, 4(10), 2184–2194.

[9] Meriam J. L., Kraige L. G.,(2012), "Engineering Mechanics-Dynamics", 7th Edition, Wiley.

[10] Yıldırım B., (2007), "A Comparative Evaluation of Conventional and Particle Filter Based Radar Target Tracking", Yüksek Lisans Tezi, Orta Doğu Teknik Üniversitesi.

[11] Bar-Shalom Y., Willett P. K., Tian X., (2011), "Tracking and Data Fusion: A Handbook of Algorithms", Storrs, Conn.: Yaakov Bar-Shalom.

[12] Bar-Shalom Y.,Li X.R., Kirubarajan T,(2001),"Estimation with Applications to Tracking and Navigation", 1st Edition,New York:Wiley-Interscience.

[13] Ramachandra K. V., (2000), "Kalman Filtering Techniques for Radar Tracking", 1st Edition, New York: CRC Press.

[14] Higgins W. T., (1975), "A Comparison of Complementary and Kalman Filtering", IEEE Transactions on Aerospace and Electronic Systems, AES-11 (3), 321–325.

[15] Welch G., Bishop G., (1995), "An Introduction to the Kalman Filter", University of North Carolina at Chapel Hill, Chapel Hill, NC, USA.

[16] Lana A., (2015), "Kalman Filtresi Ve Olasılıksal Veri İlişkilendirme Yöntemlerini Kullanan Çoklu Hedef İzleme Algoritmaları", Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi.

[17] Challa S., Morelande M. R., Mušicki D., Evans R. J., (2011), "Fundamentals of Object Tracking", 1st Edition, New York: Cambridge University Press.

[18] LaViola J. J., (2003), "A comparison of unscented and extended Kalman filtering for estimating quaternion motion", Proceedings of the 2003 American Control Conference, 3, 2435–2440, Denver, CO, USA, 4-6 June.

[19] Lin X., Kirubarajan T., Bar-Shalom Y., Maskell S., (2002), "Comparison of EKF, pseudomeasurement, and particle filters for a bearing-only target tracking problem", Proceedings of SPIE Signal and Data Processing of Small Targets, 240–250.

[20] Farina A., Ristic B., Benvenuti D., (2002), "Tracking a ballistic target: comparison of several nonlinear filters", IEEE Transactions on Aerospace and Electronic Systems, 38(3), 854–867.

[21] Wan E. A., Merwe R. V. D., (2000) "The unscented Kalman filter for nonlinear estimation", Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing Communications and Control Symposium, 153–158.

[22] Yan H., Huang G., Wang H., Shu R., (2013), "Application of Unscented Kalman Filter for Flying Target Tracking", 2013 International Conference on Information Science and Cloud Computing, 61–66, Guangzhou, China, 7-8 December.

[23] Li Q., Li R., Ji K., Dai W., (2015), “Kalman Filter and Its Application”, 2015 8th International Conference on Intelligent Networks and Intelligent Systems, 74–77, Tianjin, China, 1-3 November.

[24] Sustika R., Suryana J., (2015), “Nonlinear-Filtering with Interacting Multiple-Model Algorithm for Coastal Radar Target Tracking System”, Telkomnika Telecommunication, Computing, Electronics and Control, 13 (1), 211.

[25] Zollo S. Ristic B., (1999), “On polar and versus Cartesian coordinates for target tracking”, Proceedings of the Fifth International Symposium on Signal Processing and Its Applications, 2, 499–502.

[26] Besada J. A., Miguel G. D., Bernardos A. M., J. R. Casar, (2012), “Generic multisensor multitarget bias estimation architecture”, IET Radar Sonar&Navigation, 6 (5), 365–378.

[27] Blackman S. S., Popoli R., (1999), “Design and analysis of modern tracking systems”, Boston: Artech House.

[28] Okello N., Ristic B., (2003), “Maximum likelihood registration for multiple dissimilar sensors”, IEEE Transactions on Aerospace and Electronic Systems, 39 (3), 1074–1083.

[29] Nabaa N., Bishop R. H., (1999), “Solution to a multisensor tracking problem with sensor registration errors”, IEEE Transactions on Aerospace and Electronic Systems, 35 (1) , 354–363.

[30] Taghavi E., Tharmarasa R., Kirubarajan T., Bar-Shalom Y., Mcdonald M., (2016), “A practical bias estimation algorithm for multisensor-multitarget tracking”, IEEE Transactions on Aerospace and Electronic Systems,52 (1), 2–19.

[31] Lin X., Bar-Shalom Y., Kirubarajan T., (2003) “Exact multisensor dynamic bias estimation with local tracks”, Proceedings of the Sixth International Conference of Information Fusion, 159–167, Cairns, Queensland, Australia, 8-11 July.

[32] Lin X., Bar-Shalom Y., and Kirubarajan T., (2005), “Multisensor multitarget bias estimation for general asynchronous sensors”, IEEE Transactions on Aerospace and Electronic Systems, 41 (3), 899–921.

[33] Lerro D. Bar-Shalom Y., (1993), “Tracking with debiased consistent converted measurements versus EKF”, IEEE Transactions on Aerospace and Electronic Systems, 29 (3), 1015–1022.

[34] Longbin M., Xiaoquan S., Yiyu Z., Kang S. Z., Bar-Shalom Y., (1998), "Unbiased converted measurements for tracking", IEEE Transactions on Aerospace and Electronic Systems, 34(3), 1023–1027.

[35] Duan Z., Han C., Li X. R., (2004), "Comments on 'Unbiased converted measurements for tracking'", IEEE Transactions on Aerospace and Electronic Systems, 40 (4), 1374.

[36] Bordonaro S. V., Willett P., Bar-Shalom Y., (2012), "Unbiased tracking with converted measurements", 2012 IEEE Radar Conference, 0741–0745, Atlanta, GA, USA, 7-11 May.

[37] Bordonaro S., Willett P., Bar-Shalom Y., (2014), "Decorrelated unbiased converted measurement Kalman filter", IEEE Transactions on Aerospace and Electronic Systems, 50 (2), 1431–1444.

[38] Mei W., Bar-Shalom Y., (2009), "Unbiased Kalman filter using converted measurements: revisit", SPIE Signal and Data Processing of Small Targets, 74450U, San Diego, California, USA.

[39] Rong Li X.,Jilkov V. P, (2001), "A survey of maneuvering target tracking-Part III: Measurement models", Signal and Data Processing of Small Targets, 4473, 4473–41.

[40] Miller M. D., Drummond O. E., (2000), "Comparison of methodologies for mitigating coordinate transformation bias in target tracking", Signal and Data Processing of Small Targets, 4048, 414–428.

[41] Bordonaro S., (2015), "Converted Measurement Trackers for Systems with Nonlinear Measurement Functions", Doctoral Dissertation, University of Connecticut.

[42] Bordonaro S. V., Willett P., Bar-Shalom Y., (2012), "Bias elimination in tracking with converted position and Doppler measurements", 2012 IEEE 51st IEEE Conference on Decision and Control, 4089–4094, Maui, HI,USA, 10-13 December.

[43] Bordonaro S. V., Willett P., Bar-Shalom Y., (2011), "Tracking with Converted Position and Doppler Measurements", Signal and Data Processing of small Targets, San Diego, California, USA.

[44] Blom H. A. P., (1984), "An efficient filter for abruptly changing systems", The 23rd IEEE Conference on Decision and Control, 656–658, Las Vegas, Nevada, USA, 12-14 December.

[45] Blom H. A. P., (1986), “Overlooked potential of systems with Markovian coefficients”, 25th IEEE Conference on Decision and Control, 1758–1764, Athens, Greece, 10-12 December.

[46] Blom H. A. P. , Bar-Shalom Y., (1988), “The interacting multiple model algorithm for systems with Markovian switching coefficients”, IEEE Transactions on Automatic Control, 33 (8), 780–783.

[47] Aly S. M., Fouly R. E., Braka H., (2009), “Extended Kalman filtering and Interacting Multiple Model for tracking maneuvering targets in sensor netwotrks”, 2009 Seventh Workshop on Intelligent solutions in Embedded Systems, 149–156, Ancona, Italy, 25-26 June.

[48] Blair W. D., Watson G. A., (1992), “Interacting multiple bias model algorithm with application to tracking maneuvering targets”, Proceedings of the 31st IEEE Conference on Decision and Control, 3790–3795 (4), Tucson, AZ, USA, 16-18 December.

[49] Li X. R., Jilkov V. P., (2005), “Survey of maneuvering target tracking. Part V. Multiple-model methods”, IEEE Transactions on Aerospace and Electronic Systems, 41 (4), 1255–1321.

[50] Chen X., Gao J., Han X., (2014), “An algorithm based on interacting multiple models for maneuvering target tracking”, IEEE 7th Joint International Information Technology and Artificial Intelligence Conference, 405–408, Chongqing, China, 20-21 December.

[51] Blair W. D., Watson G. A., Rice T. R., (1991), “Interacting multiple model filter for tracking maneuvering targets in spherical coordinates”, IEEE Proceedings of Southeastcon ’91, 1055–1059 (2), Williamburg, VA, USA, 7-10 April.

[52] Simeonova I., Semerdjiev T., (2002), “Specific Features of IMM Tracking Filter Design”, Information&Security: International Journal, 9, 154-165.

[53] Yuan X., Han F., Han C., (2014), “Models and Algorithms for Tracking Target with Coordinated Turn Motion”, Mathematical Problems in Engineering, 2014, 1–10.

[54] Ji D., Li D., (2014), “An Improved Sequential Initiation Method for Multitarget Track in Clutter with Large Noise Measurement”, Mathematical Problems in Engineering.

[55] Hu Z., Leung H., Blanchette M., (1997), “Statistical performance analysis of track initiation techniques”, IEEE Transactions on Signal Processing., 45 (2), 445–456.

[56] Blackman S. S., (1986), “Multiple-Target Tracking with Radar Applications”, 1st Edition, Artech Print on Demand.

[57] Bar-Shalom Y., Fortmann T. E., (1988), “Tracking and Data Association”, 1st Edition, Boston: Elsevier Science.

[58] Hunt D. J., Nolte L. W., Reibman A. R., Howard Ruedger W., (1990), “Hough transform and signal detection theory performance for images with additive noise”, Computer Vision, Graphics and Image Processing, 52 (3), 386–401.

[59] Casasent D. P., Slaski J., (1988), “Optical track initiator for multitarget tracking”, Applied Optics, 27, 4546–53.

[60] Carlson B. D., Evans E. D., Wilson S. L., (1994), “Search radar detection and track with the Hough transform. I. system concept”, IEEE Transactions on Aerospace and Electronic Systems, 30 (1), 102–108.

[61] Lo T., Litva J., Leung H., Bridgewater A. W., (1994), “Multitarget track initiation using a modified Hough transform” Pointing and Tracking Systems, 8.

[62] Pektas F. Sabuncu M., (2012), “A score-based track initiation procedure to eliminate false tracks in the presence of sea clutter”, 2012 IEEE Radar Conference, 0830–0834, Atlanta, GA, USA, 7-11 May.

[63] Musicki D., Evans R., Stankovic S., (1994), “Integrated probabilistic data association”, IEEE Transactions on Automatic Control, 39 (6), 1237–1241.

[64] Niedfeldt P., (2014), “Recursive-RANSAC: A Novel Algorithm for Tracking Multiple Targets in Clutter”, Doctoral Dissertation, Ira A. Fulton College of Engineering and Technology.

[65] Leung H., Hu Z., Blanchette M., (1996), “Evaluation of multiple target track initiation techniques in real radar tracking environments”, IEE Proceedings - Radar, Sonar and Navigation, 143 (4), 246–254.

ÖZGEÇMİŞ

Kübra TURGUT, 1993 yılında Erzincan'da doğdu. 2010 yılında Erzincan Anadolu Lisesi'ni tamamladı ve ardından başladığı Gebze Teknik Üniversitesi'nde (GTÜ) bir yıl yabancı dil eğitimi aldıktan sonra, Mühendislik Fakültesi, Elektronik Mühendisliği Bölümü'nü 2015 yılında üçüncü olarak bitirdi. GTÜ Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Anabilim Dalı'nda aynı yıl yüksek lisans eğitimine başladı. Şubat 2016 tarihinden bu yana Gebze Teknik Üniversitesi Elektronik Mühendisliği Bölümü'nde araştırma görevlisi olarak Sinyal ve Görüntü İşleme alanında çalışmaktadır.

EKLER

Ek A: Tez Çalışması Kapsamında Yapılan Yayınlar

Sönmez H. H., Turgut K., Hocaoğlu A. K., (2017), "Improving track initiation performance by track validation algorithms for multi-target tracking in heavy clutter", 25th Signal Processing and Communications Applications Conference, 1–4, Antalya/Türkiye, 15-18 May.