

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**MANEVRA YAPAN HEDEFLERİN KONUM VE
KİNEMATİK BİLGİLERİNİ EN İYİ KESTİREN
FİLTRELERİN İYİLEŞTİRİLMESİ VE YENİ BİR
YAKLAŞIM OLAN ŞABLON FİLTRESİNİN TASARIMI**

Müh. Erdiñ ÜSTÜAY

**FBE Elektronik ve Haberleşme Mühendisliğı Anabilim Dalı Haberleşme Programında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Yrd. Doç. Dr. Soner ÖZGÜNEL

İSTANBUL, 2007

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iii
KISALTMA LİSTESİ.....	iv
ŞEKİL LİSTESİ.....	v
ÖNSÖZ	viii
ÖZET	ix
ABSTRACT	x
1. GİRİŞ	1
1.1 Önceki Çalışmalar	2
1.2 Genel Test Senaryosu	3
1.3 Radar Modeli.....	4
1.4 Başarım Karşılaştırma Ölçütü	5
2. TEK FİLTRELİ MODEL	7
2.1 Alfa-Beta ve Alfa-Beta-Gama Filtresi.....	7
2.1.1 Alfa – Beta Filtresi.....	7
2.1.2 Alfa – Beta - Gama Filtresi	8
2.1.3 Alfa-Beta ve Alfa-Beta-Gama filtrelerinin testi	12
2.1.3.1 Alfa-Beta filtresinin test sonuçları.....	12
2.1.3.2 Alfa-Beta-Gama filtresinin test sonuçları:	15
2.1.4 Genel Senaryo Test Sonuçları	19
2.1.5 Sonuç	19
2.2 Alfa-Beta ve Alfa-Beta-Gama Filtrelerinde İyileştirme Yaklaşımı: Uyarlamalı Alfa-Beta ve Uyarlamalı Alfa-Beta-Gama Filtreleri	21
2.3 Kalman Filtresi	28
2.3.1 Kalman Filtresinin Genel Senaryo Test Sonuçları	34
3. ÇOK FİLTRELİ MODEL.....	39
3.1 IMM (Etkileşimli Çoklu Model) Filtresi	39
3.1.1 IMM Filtresinin Genel Senaryo Test Sonuçları	43
4. YENİ BİR YAKLAŞIM: ŞABLON FİLTRESİ	56
4.1.1 Şablon filtresinin genel senaryo test sonuçları	81
4.1.2 Şablon filtresinin diğer filtrelerle karşılaştırılması	84
4.1.3 Gerçek Zamanlı Şablon Filtresi.....	88
5. SONUÇLAR ve ÖNERİLER.....	92
KAYNAKLAR.....	96
ÖZGEÇMİŞ.....	97

SİMGE LİSTESİ

α	alfa
β	beta
γ	gama
σ_n^2	Ölçüm hatası varyansı
Kx	Konum varyansı azalma oranı
Kv	Hız varyansı azalma oranı
Ka	İvme varyansı azalma oranı
T	Ölçümler arası geçen zaman
V	Hız
$x(t_{k+1} / t_k)$	k+1 zamanı için k zamanında yapılmış durum tahmini
$x(t_{k+1} / t_{k+1})$	k+1 zamanı için karar verilen durum
$z(t_{k+1})$	k+1 zamanında alınmış ölçüm
$v(t_{k+1})$	k+1 zamanında yapılan yenilenme miktarı
$P(t_{k+1} / t_k)$	k+1 zamanı için k zamanında durum kovaryans matrisi için yapılmış tahmin
$P(t_{k+1} / t_{k+1})$	k+1 zamanında karar verilen durum kovaryans matrisi
$S(t_{k+1})$	k+1 zamanındaki yenilenme kovaryansı
H	Ölçümler için Jacobien matrisi
$K(t_{k+1})$	k+1 anında filtre kazancı
X	n x 1 durum vektörü
F	n x n durum değişim matrisi
G	n x 1 bozulma matrisi
w(t _k)	Sıfır ortalamalı Gauss beyaz gürültüsü
Q	Süreç gürültüsü kovaryans matrisi
H _k	Kalman filtresi için m x n ölçüm matrisi
R	Ölçüm gürültüsü varyans matrisi
v(t _k)	Sıfır ortalamalı beyaz Gauss gürültüsü
W _i (t _k)	i'inci filtrenin olasılığı
P _{ij}	i'den j'ye geçiş olasılığı
N	Filtre sayısı
E	Toplam karesel hata

KISALTMA LİSTESİ

AADO	Artan Aralıklı Doğru Oturtma
AB	Alfa-Beta
ABG	Alfa-Beta-Gama
DH	Dairesel Hareket
DDH	Düzgün Doğrusal Hareket
DDHH	Düzgün Doğrusal Hızlanan Hareket
GENESİS	Gemi Entegre Savaş İdare Sistemi
IMM	Interacting Multiple Model (Etkileşimli Çoklu Model)
MMSE	Minimum Mean Square Error (En Küçük Ortalama Karesel Hata)
NIMM	Non-Interacting Multiple Model (Etkileşimsiz Çoklu Model)
NKH	Normalize Konum Hatası
OHH	Ortalama Hız Hatası
SAÇO	Sabit Aralıklı Çember Oturtma
SADO	Sabit Aralıklı Doğru Oturtma

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1 Genel test senaryosu.....	4
Şekil 1.2 Radar modeli.....	5
Şekil 2.1 Konum varyansı azalma oranı	11
Şekil 2.2 Hız varyansı azalma oranı	11
Şekil 2.3 İvme varyansı azalma oranı	12
Şekil 2.4 Alfa-Beta filtresinin $\alpha = 0.99$ için başarımı	13
Şekil 2.5 Alfa-Beta filtresinin $\alpha = 0.6$ için başarımı	14
Şekil 2.6 Alfa-Beta-Gama filtresinin $\alpha = 0.99$ için başarımı.....	16
Şekil 2.7 Alfa-Beta-Gama filtresinin $\alpha = 0.6$ için başarımı.....	17
Şekil 2.8 Hedef dinamiği değişiminde Alfa-Beta-Gama filtresinin α 'ya göre tepkisi.....	18
Şekil 2.9 Alfa-beta ve alfa-beta-gama filtrelerinin $\alpha=0.6$ için normalize konum ve ortalama hız hatası karşılaştırması	19
Şekil 2.10 İvmeye dalgalanmanın oluşturduğu mutlak hata	20
Şekil 2.11 Normal Alfa-Beta-Gama filtresinde yenilenme ve ölçüm hatası.....	22
Şekil 2.12 Normal Alfa-Beta-Gama filtresinde normalize konum hatası.....	22
Şekil 2.13 Alfa-Beta-Gama filtresinin bulduğu ivme	23
Şekil 2.14 Alfa parametre değişimi (yeni yaklaşımda manevra yakalama).....	23
Şekil 2.15 Uyarlamalı Alfa-Beta-Gama filtresinde normalize konum hatası	23
Şekil 2.16 Normal Alfa-Beta ile Uyarlamalı Alfa-Beta filtrelerinin rotaya göre başarılarının karşılaştırması	24
Şekil 2.17 Normal Alfa-Beta-Gama ile Uyarlamalı Alfa-Beta-Gama filtrelerinin rotaya göre başarılarının karşılaştırması	25
Şekil 2.18 Normal Alfa-Beta ve Alfa-Beta-Gama filtrelerinin Uyarlamalı Alfa-Beta-Gama filtresi ile farklı m değerleri için ortalama hız hatası karşılaştırması.....	26
Şekil 2.19 Normal alfa-beta ve alfa-beta-gama filtrelerinin uyarlamalı alfa-beta filtresi ile farklı m değerleri için ortalama hız hatası karşılaştırması	27
Şekil 2.20 Kalman Filtresi Blok Diyagramı.....	28
Şekil 2.21 Kalman Filtresi akış diyagramı	31
Şekil 2.22 Kalman Filtresi test sonuçları	33
Şekil 2.23 Radar ölçüm çözünürlük bölgesi.....	35
Şekil 2.24 Kalman filtresinin genel senaryoda takip başarımları.....	37
Şekil 2.25 Kalman filtresinin değişik süreç gürültüleri ile genel senaryoda takip başarımları.....	

karşılaştırmaları.....	38
Şekil 3.1 IMM Algoritması	40
Şekil 3.2 IMM filtresinin genel senaryo üzerinde görsel test sonucu.....	45
Şekil 3.3 Test 1: IMM filtresinde alt filtre ağırlıkları değişimi ve normalize konum hatası başarımı	46
Şekil 3.4 Test 2: IMM filtresinde alt filtre ağırlıkları değişimi ve normalize konum hatası başarımı	47
Şekil 3.5 Test 1 ile Test 3'ün normalize konum hataları karşılaştırması.....	48
Şekil 3.6 Test 1 ile Test 3'ün normalize konum hataları karşılaştırması-2	49
Şekil 3.7 Test 1 ile Test 3'ün ortalama hız hataları karşılaştırması.....	49
Şekil 3.8 IMM Test4 NKH sonucu	50
Şekil 3.9 IMM Test4 filtre olasılıkları	50
Şekil 3.10 IMM Test4'te filtre 1'in bağımsız ürettiği sonuç.....	51
Şekil 3.11 IMM Test4'te kovaryans matris artırıldığı durumda filtre 1'in bağımsız ürettiği sonuç.....	52
Şekil 3.12 IMM Test4'te filtre 1'in kovaryans matrisi artırıldığında filtre olasılıkları	52
Şekil 3.13 NIMM filtresinde alt filtrelerin manevraya bağımlı olarak ağırlıklarının değişimi	53
Şekil 3.14 NIMM filtresi ile IMM filtresinin NKH karşılaştırması	54
Şekil 3.15 NIMM filtresi ile IMM filtresinin OHH karşılaştırması	54
Şekil 3.16 Filtre 1'in NIMM filtresinde bağımsız katkısı.....	55
Şekil 4.1 Düzgün doğrusal hareket.....	56
Şekil 4.2 Düzgün doğrudan hızlanan hareket.....	57
Şekil 4.3 Dönme hareketleri	58
Şekil 4.4 Dönme hareketi yapan hedefin bulunacağı konumlar.....	59
Şekil 4.5 Bir hava hedefinin hareketlerinin üç ana hareket çeşidi ile temsil edilebileceğini gösteren şekil	59
Şekil 4.6 Ölçümlere sabit aralıklı doğru oturtma örneği.....	60
Şekil 4.7 Sabit aralıklı doğru oturtma yönteminde karar verilen noktaların x ve y ekseninde iz düşümü	61
Şekil 4.8 Örnek verilere sabit aralıklı doğru oturtma algoritmasının Matlab'de test sonucu ..	63
Şekil 4.9 Ölçümlere artan aralıklı doğru oturtma örneği	64
Şekil 4.10 Artan aralıklı doğru oturtma yönteminde karar verilen noktaların x ekseninde iz düşümü	65
Şekil 4.11 Ölçümlere artan aralıklı doğru oturtma yönteminde doğrusallığın	

sağlanamayabileceğini gösteren şekil	68
Şekil 4.12 $x=a$ doğrusuna yakın hareket eden hedefe doğru oturtma modelleri	69
Şekil 4.13 Matlab’de örnek verilere artan aralıklı doğru oturtma algoritması test sonucu.....	70
Şekil 4.14 Ölçümlere sabit aralıklı çember oturtma örneği	72
Şekil 4.15 Matlab’de ölçümlere çember oturtma algoritması test sonucu	73
Şekil 4.16 Ölçümlerin çember üzerindeki iz düşümü.....	75
Şekil 4.17 İzdüşüm açılarının düzeltilmesi	75
Şekil 4.18 SAÇO filtresi akış diyagramı.....	77
Şekil 4.19 SAÇO filtresinin örnek verilerle Matlab’de test sonucu	78
Şekil 4.20 Şablon filtresi akış diyagramı	79
Şekil 4.21 Şablon filtresinin genel senaryoda başarımı - 1	81
Şekil 4.22 Şablon filtresinin genel senaryodaki başarımı - 2.....	82
Şekil 4.23 Şablon filtresi kinematik sonuçlarının analizi	83
Şekil 4.24 Şablon filtresinin motif başarısı açısından diğer filtrelerle karşılaştırılması	86
Şekil 4.25 Genel senaryo ile test edilen Şablon, IMM-2 ve Uyarlamalı(Dinamik) Alfa-Beta-Gama filtrelerinin NKH karşılaştırması	87
Şekil 4.26 Genel senaryo ile test edilen Şablon, IMM-2 ve Dinamik ABG filtrelerinin ortalama hız hataları karşılaştırması.....	88
Şekil 4.27 Gerçek Zamanlı Şablon Filtresinin akış diyagramı.....	89
Şekil 4.28 Genel senaryo ile test edildiğinde Gerçek Zamanlı Şablon Filtresi’nin motif başarısı.....	90
Şekil 4.29 Genel senaryo ile test edilen Gerçek Zamanlı Şablon, Geçmiş Düzeltmeli Şablon ve IMM-2(NIMM) filtrelerinin NKH karşılaştırması	91
Şekil 4.30 Genel senaryo ile test edilen Gerçek Zamanlı Şablon, Geçmiş Düzeltmeli Şablon ve IMM-2 filtrelerinin NKH karşılaştırması	91

ÖNSÖZ

Manevra yapan çoklu hedef takibi, gözetleme radar sistemlerinin en önemli parçalarından biridir. Son yıllarda ülkemizde önemli bir araştırma konusu olmuştur. Özellikle Deniz Kuvvetleri'nin Milli Gemi projesinde bu konuda önemli bir çalışma yapılmıştır. TÜBİTAK MAM'da çalışmış olduğum GENESİS projesinde gerçek dünya verileri üzerinde edindiğim tecrübeler sonucunda, literatürde manevra yapan hedef takibi için tasarlanmış filtrelerin gerçek hayatta uygulamaları konusunda yeterince irdelenmemiş olduklarını, bu filtrelerin topluca değerlendirilip başarı kriterlerinin yapılmadığını ve bu filtrelerin seçimi ve kullanımı konusunda kullanıcılar için yeterli referans kaynağının olmadığını gördüğüm için bu konuda çalışmaya karar verdim.

Yüksek Lisans tez çalışmalarım süresince bana yol gösteren değerli danışmanım Yrd. Doç. Dr. Soner Özgünel'e teşekkürü borç bilirim.

TÜBİTAK'taki çalışmalarımda bana ilham veren Sn. Y. Müh. Bülent Candan ile yardımlarından ötürü Y. Müh. Adnan Lana'ya, TÜBİTAK MAM'daki proje arkadaşlarıma, Müh. Aytaç Sartekin ve Y. Müh. Özleyiş Ocakoğlu'na teşekkürlerimi sunarım.

Eğitim hayatım süresince maddi manevi desteklerini esirgemeyen aileme ve tezimi hazırlamamda yardım eden sevgili eşim Hilal Arslan Üstüay'a sevgilerimle,

Temmuz, 2007

Erdoğan Üstüay

ÖZET

Yüksek manevra yapan hava hedeflerinin takibinde en önemli kısım filtrelemedir. Filtre hava tarama algılayıcısı'ndan (radar) alınan gürültülü ölçümler ile hava hedefinin konum ve kinematiklerini (konum, hız ve ivme) belirler. Bu kinematikler ne kadar doğru belirlenirse hedefin rotası ve tipi hakkında o kadar doğru bilgiye ulaşılır ve takibinin devamı sağlanır.

Sabit katsayılı filtrelerden Alfa-Beta ve Alfa-Beta-Gama filtreleri detaylı bir şekilde irdelenmiştir. Bu filtrelerin çalışma prensipleri, birbirlerine göre avantaj ve dezavantajları teorik olarak ortaya konmuştur. Filtreler gerçek koşullara yakın olarak yüksek hızlarda hareket eden ve yüksek manevra yapan bir hedefin benzetim verileri ile test edilmiş ve başarımları bu verilerle tekrar ortaya konmuş ve karşılaştırılmıştır. Bu filtrelere dinamiklik kazandıran bir mekanizma eklenerek Uyarlamalı Alfa-Beta ve Uyarlamalı Alfa-Beta-Gama filtreleri önerilmiş ve yine gerçeğe yakın verilerle test edilerek topluca başarımları karşılaştırılmıştır.

Değişken katsayılı olarak bilinen en başarılı filtrelerden ise Kalman ile çoklu model olarak bilinen en başarılı filtrelerden Etkileşimli Çoklu Model (IMM) filtrelerinin teorik olarak çalışma prensipleri anlatılarak avantaj ve dezavantajları ortaya konmuş, çeşitli parametreler ile başarımları test edilmiştir. IMM filtresinde gerçek koşullarda başarımları artırıcı parametreler önerilmiş, yine bu filtrenin etkileşim kısmı çıkarılarak Etkileşimsiz Çoklu Model (NIMM) tanımlanıp başarımları gerçek verilere yakın benzetim ortamında test edilmiştir.

Ayrıca geleneksel filtreleme tekniklerinden uzak, kitlesel veri işleme yöntemine dayalı yeni bir filtre (Şablon filtresi) tasarlanmış, bu filtrenin yine bu çalışma kapsamında başarımları artırılmış filtrelerle topluca karşılaştırılması gerçeğe yakın senaryo verileri kullanılarak yapılmıştır.

Anahtar Kelimeler: Hedef takibi, manevra, Kalman, IMM, Alfa-Beta, Alfa-Beta-Gama, NIMM, Şablon filtresi

ABSTRACT

The most important part of the maneuvering air target tracking systems is the filtering part. Filters find the position and kinematics (velocity and acceleration) of the targets using the noisy measurements taken from sensor (radar). The better positions and kinematics are calculated, the better route is found and target is determined. This is also important for the continuity of tracking.

Among known constant gain filters, Alpha-Beta and Alpha-Beta-Gamma filters, among the adaptive filters, Kalman Filter, and multiple model, IMM (Interacting Multiple Model) Filter are analysed in deep. Their working principles, advantages and disadvantages, success and fail conditions are presented in theory. Their usage in real world scenarios and the way to choose parameters are shown.

An improvement suggestion that brings adaptivity to Alpha-Beta and Alpha-Beta-Gamma filters is proposed. The improved filters, called Adaptive Alfa-Beta and Adaptive Alpha-Beta-Gamma filters, are analysed with their advantages and disadvantages and their performances are compared with those of basic Alpha-Beta and Alpha-Beta-Gamma Filters.

Additionally a modification to the IMM filter which removes the interaction part of IMM filter, which is called Non-interacting Multiple Model (NIMM) filter, is proposed and compared with basic IMM filter.

Finally a new filter, called Template Filter, is introduced. This filter's performance is tested with real world scenarios and compared with those of all previous filters.

All the filters are tested and compared with most difficult scenarios for trackers and air tracking radar works with real world conditions.

Keywords: Target tracking, maneuver, Kalman, IMM, Alpha-Beta, Alpha-Beta-Gamma, NIMM, Template filter

1. GİRİŞ

Manevra yapan hedef takibi problemi uzun yıllar çalışma konusu olmuştur. Manevralı hedef takibinde algoritmaların beklenmeyen sonuç üretmesine sebep olacak birkaç hata kaynağı vardır. En temel iki kaynak şunlardır:

1. Filtre tasarım modeli ile sistem modeli arasındaki uyumsuzluk
2. Hedefin manevrası sonucu sistem modelinin dinamik değişimleri

Hedef modeli ile hedef dinamikleri modeli arasındaki uyumsuzluk genelde hedef bir manevraya başladığında ve manevrayı devam ettirdiğinde oluşur. Aradaki fark büyüdükçe hedef takip algoritmasının başarısı azalır. Bu problemi çözmek için manevra anının başlangıcını ve değişimini tespit etmek ve yeni koşulları temsil eden parametreleri bulmak gerekir. Bu problem için çeşitli yaklaşımlar önerilmiştir. Bunların ilki ve en basiti manevrayı tespit ederek filtre durum modelini, manevranın boyutuna göre düzeltmektir. Bu metodun dezavantajı manevrayı tespit edene kadar geçecek sürenin getireceği kayıplardır. Bu problem için diğer bir çözüm, birden çok filtrenin aynı anda kullanılması ve böylece birden çok manevra koşullarına hazır filtrelerin aynı anda çalıştırılmasıdır. Uygulamada ise bu çözümün dezavantajı, çok sayıda filtre kullanma gerekliliği ve bundan doğan işlem yükü ve karmaşıklığın fazlalığıdır. Hangi filtrenin sonucunun genel sonuca nasıl katkıda bulunması gerektiği ayrı bir sorundur.

Hedef takibinde en eski çözüm alfa-beta ve alfa-beta-gama filtreleridir. Uyarlamalı olmayan bu filtreler değişen koşullara göre parametrelerini değiştiremezler. Bu sebeple filtreleme katsayıları sabittir. Ancak bu filtreler ek bir mekanizma ile uyarlamalı yani dinamik yapılabilir. Uygulama kolaylığı, işlemci gücü gereksinimlerinin az olması, değişen dinamiklere hızlı cevap verme gibi özellikleri sebebiyle bu filtreler incelenmeye değer görülmektedir.

Bir başka çözüm, en sık kullanılan Kalman Filtresi'dir. Diğer çözümler ise çoklu modellerdir. Birden çok filtre kullanan bu yöntem başarımı en yüksek seviyeye çıkarmaktadır.

Giriş bölümünde, problem için genel tanım yapılmakta ve bu çalışmada yapılacakların özeti verilmektedir. Ayrıca geçmişte yapılmış çalışmalardan da bu bölümde bahsedilmektedir. Yine bu bölümde filtrelerin analizinde kullanılan yapay veri üretici ve test senaryolarından ve de kullanılan hava tarama algılayıcısının yapısından bahsedilmektedir.

İkinci bölümde önceki çalışmalarda başarısı teorik olarak ispatlanmış ve uygulamada kullanılan tek filtreli modeller incelenecektir. Bu bölümde filtrelerden Alfa-Beta, Alfa-Beta-

Gama, Uyarlamalı Alfa-Beta, Uyarlamalı Alfa-Beta-Gama, en iyi MMSE hatası veren Kalman Filtresi, üçüncü bölümde çok filtreli modellerden Kalman kullanan Etkileşimli Çoklu Model (IMM) ve değiştirilmiş IMM filtreleri tanımlanıp başarımları karşılaştırılmaktadır. Dördüncü bölümde yeni bir yaklaşımla tasarlanmış olan Şablon filtresi'nin tanımı yapılmaktadır ve algoritması anlatılmaktadır. İki, üç ve dördüncü bölümlerde bahsedilen filtreler için iyileştirme yaklaşımları yapılmakta ve bu yaklaşımların başarımları test edilmektedir. Bütün filtreler birbirleriyle kıyaslanarak en iyi sıralaması yapılmaktadır.

Önceki çalışmalarda mevcut filtreler için yapılmış olan çalışmalar teorik temelli olup gerçek dünya verileriyle test edilmemiştir. Bu sebeple analiz sonuçları ve başarı karşılaştırmaları gerçekçi değildir. İki, üç ve dördüncü bölümlerde filtreler gerçeğe yakın manevraları benzeten bir benzetim ortamından alınan verilerle test edilmekte, bu filtreler için en iyi parametre setlerinin nasıl seçilmesi gerektiği uygulamalı olarak anlatılmaktadır. Her filtrenin diğer filtrelerle gerçek dünya verilerine göre başarımları karşılaştırılmakta ve toplu başarımları ele alınmaktadır.

Beşinci bölümde bütün filtrelerin sonuçları, başarımların sıralamaları, yapılan iyileştirme çalışmaları ve bu konuda ileride yapılabilecek geliştirme önerileri sunulmaktadır.

1.1 Önceki Çalışmalar

1955 yılında N. Wax tarafından çoklu hedef izleme sistemleri ortaya konduğunda hedef takibi radar ekranları üzerinden gözle yapılmaktaydı. Wax, bir nükleer fizik problemi ile benzerlik kurarak veri ilişkilendirme hariç hedef izleme probleminin diğer bileşenlerini oluşturan izleme başlatma, iz güncelleme ve izleme sonlandırmayı tanımlamıştır.

1964 yılına gelindiğinde Sittler kendisinden sonra gelen çalışmaların başlangıcını oluşturacak Bayes bağıntılarını kullanmıştır. 1970'lerde Kalman filtresinin yaygın kullanılmaya başlaması ile birlikte çoklu hedef izleme gözetleme sistemlerinin vazgeçilmez unsuru olmuştur. Bar-Shalom, Y. ve Li, X.R.(1993)'in hedef takibi konusunda yayını ile çağdaş anlamdaki çoklu hedef izleme teknikleri gelişmeye başlamıştır.

Otomatik hedef izleme sistemlerinin en önemli parçalarından biri olan filtreler de bu gelişme içinde önemli mesafe katetmiştir. Önceleri işlemci gücü açısından Alfa-Beta ve Alfa-Beta-Gama filtreleri kullanılırken daha sonra işlemci güçlerinin artmasıyla Kalman filtresi kullanılmaya başlanmıştır. Daha sonraları uyarlamalı Kalman filtresi tanıtılmıştır. Li, X. R. ve Bar Shalom, Y.(1993)'nin tanıttığı IMM filtresi ilk çoklu modellerdendir. Günümüzde en popüler filtreler olarak IMM ve Uyarlamalı IMM filtreleri üzerinde çalışmalara devam

edilmektedir. Efe, M. (1998) doktora tezinde Uyarlamalı IMM filtresi için yeni yaklaşımlar getirmiştir.

1.2 Genel Test Senaryosu

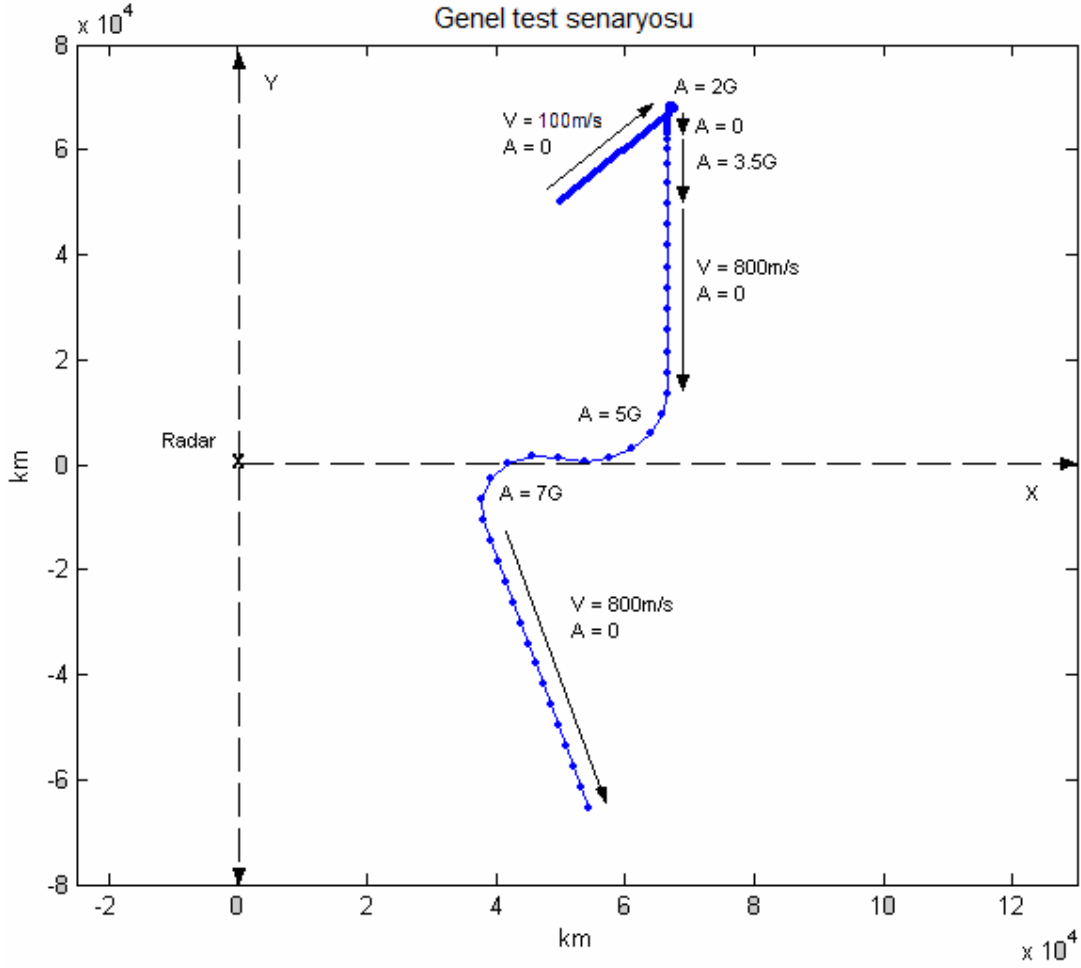
Tez kapsamında bütün filtreleri test etmek ve başarımlı karşılaştırması yapmak için gerçek dünya koşullarına yakın genel bir senaryo düşünülmektedir. Bu senaryo, savaş uçaklarının yapabileceği en zorlu manevraları içermekte olup senaryo üretici manevraları gerçek zamanlı ve gerçeğe en yakın olacak şekilde gerçekleştirmektedir.

Uçak radar'dan 75 km uzakta hareket etmektedir. Radar ise gerçeğe uygun olacak şekilde uzun menzil su üstü ve hava tarama radarlarında olduğu gibi ölçümleri menzil ve yatay açı koordinatlarında yapar. Burada bir savaş uçağının yapabileceği maksimum hız 800 m/s (2800 km/sa) minimum hız 100 m/s (360km/sa) maksimum ivme ise 7G (70m/s²) olduğu için senaryo bu koşulları sağlayacak şekilde hazırlanmaktadır. Bu değerler deniz kuvvetlerinden GENESIS projesi kapsamında yapılan çalışmalardan alıntıdır.

Hava tarama radarı menzilde maksimum ± 50 m normal dağılımlı hata ile ölçüm almaktadır. Yatay açıda ise $\sigma=0.4^\circ$ Gauss dağılımlı hata yapmaktadır. Radar 5 saniyede bir tur atabilmektedir yani gözlemler arası geçen süre 5 saniyedir. Bu değerler savaş gemilerindeki uzun menzil hava tarama radarlarının çalışma değerlerine oldukça yakındır. Burada ele alınan modelde uçak aşağıdaki hareketleri sırayla gerçekleştirmektedir:

1. 100m/s sabit hızla 250 saniye(~ 4 dakika) boyunca sabit hızla hareket etmektedir.
2. 2G ivme ile sola dönüş yapar ve 20 saniye boyunca döner.
3. Sabit hızla 50 saniye devam eder.
4. Hareket yönünde 3.5G ile 20 saniye hızlanır ve hızını 800m/s'ye çıkarır.
5. Aynı hızla 50 saniye devam eder.
6. 30 saniye boyunca sağa 5G ivme ile dönüş yapar.
7. Hemen ardında 25 saniye boyunca sola 7G'lik dönme yapar.
8. Mevcut hızla 125 saniye yoluna devam eder.

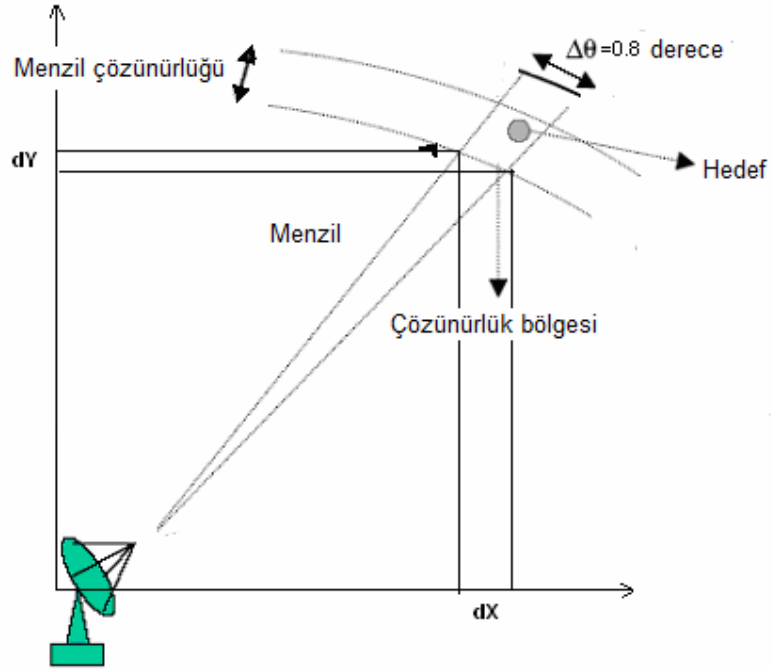
Bu senaryo, Şekil 1.1'de görülmektedir.



Şekil 1.1 Genel test senaryosu

1.3 Radar Modeli

Hava tarama radarları ölçümleri, menzil ve yatay açı koordinatlarında almaktadır. Bir üçüncü koordinat olarak yükselme açısı kullanılabilir. Bu çalışmada, sadece iki boyutlu olarak gözlemlerin alındığı yani hedeflerin sadece belli bir yükseklikte hareket ettikleri kabul edilmektedir. Üçüncü boyutta hareketin takibi bu çalışmanın kapsamı dışındadır. Şekil 1.2’de radar modeli gösterilmektedir.



Şekil 1.2 Radar modeli

Radardan alınan ölçümler x ve y koordinatlarına dönüştürülerek hedef takip filtrelerince kullanılmaktadır. Radarın 5 saniyede bir ölçüm aldığı, kargaşa nedeniyle oluşan hatalı ölçüm oranının sıfır olduğu ve her turda hedefe ait gözlem alındığı kabul edilmektedir. Hedeften gözlem alınamaması durumunda oluşacak durumun incelenmesi bu tezin kapsamı dışındadır.

Radar menzilde maksimum $\pm 50m$ yatay açıda ise $\pm 0,4$ derece standart sapmayla ölçüm almaktadır.

1.4 Başarım Karşılaştırma Ölçütü

Filtrelerin performanslarının ölçümünde Monte Carlo benzetim yöntemi kullanılmaktadır. Bu yöntemle göre bir algoritma N defa çalıştırılmaktadır. Gürültünün dağılımına göre her seferinde gürültü değeri farklı olmaktadır. Sonuç ise bütün koşulların karelerinin ortalaması olur.

Ayrıca açısal hata, uçağın menzili arttıkça orantılı bir şekilde artacağı için toplam ölçüm hatası da menzile göre orantılı olarak artar. Bu durumda performans ölçümü yaparken menzile göre düzeltme yapılması gerekir.

Test sonuçları karşılaştırılırken kullanılan genel ölçüt normalize konum hatasıdır. Lana A., (2001) ve Efe M., (1998) yaptığı çalışmalarda sadece bu ölçütü kullanmıştır.

NPE: Normalize konum hatası

RPE: Toplam konum hatası

RME: Toplam ölçüm hatası

$x_i(n)$: n anında gerçek konumun x-eksenindeki değeri

$\hat{x}_i(n)$: n anında filtrenin bulduğu konumun x-eksenindeki değeri

$y_i(n)$: n anında gerçek konumun y-eksenindeki değeri

$\hat{y}_i(n)$: n anında filtrenin bulduğu konumun y-eksenindeki değeri

$xm_i(n)$: n anında yapılan ölçümün x-eksenindeki değeri

$ym_i(n)$: n anında yapılan ölçümün y-eksenindeki değeri

N: Koşulan test sayısı

$$NPE(n) = \frac{RPE(n)}{RME(n)} = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N [(\hat{x}_i(n) - x_i(n))^2 + (\hat{y}_i(n) - y_i(n))^2]}}{\sqrt{\frac{1}{N} \sum_{i=1}^N [(xm_i(n) - x_i(n))^2 + (ym_i(n) - y_i(n))^2]}} \quad (1.1)$$

Başarım ölçütü olarak sadece normalize konum hatası kullanmak yeterli değildir. Uygulamaya göre bu değişebilir. Örneğin radar operatörü için cismin tam olarak bulunduğu konumu değil de yönü ve hızı önemli olabilir. Bu durumda ortalama hız hatası da değerlendirilmelidir. Ayrıca filtrelerin ürettiği hız değerindeki sapmaların varyansı da bir ölçüt olarak ele alınabilir çünkü operatöre verilen filtre çıktılarında örneğin hız değerinin sürekli olarak ve büyük boyutlarda değişmesi başarısızlık olarak düşünülebilir.

Başarım için diğer göze alınması gereken öge filtrenin ürettiği konum değerlerinin oluşturduğu motifin düzgünlüğüdür. Ölçümlerin gürültülü olmasından dolayı filtre sonucu olarak verilen düzeltilmiş konum değerleri cismin yaptığı hareketi en güzel şekilde göstermelidir. Örneğin aslında düzgün doğrusal giden bir cismin rotası, filtre sonucunda düzgün bir doğru olarak verilmelidir ve zigzaglar olmamalıdır.

2. TEK FİLTRELİ MODEL

Hedef takibinde kullanılan ilk filtre modeli tek filtreli modeldir. Bu modelde hedeflerin takibi tek bir filtre kullanılarak yapılmaktadır. Bu modelde filtre olarak alfa-beta, alfa-beta-gama, Kalman ve Uyarlamalı Kalman filtrelerinden biri kullanılabilir. Bu çalışmada tekli filtrelerden Alfa-Beta, Alfa-Beta-Gama ve Kalman filtreleri incelenmektedir.

2.1 Alfa-Beta ve Alfa-Beta-Gama Filtresi

2.1.1 Alfa – Beta Filtresi

Alfa-Beta filtresi, sabit katsayılı bir iki boyutlu bir filtredir. Bu filtrede hedefin sabit hızla hareket ettiği bu arada sıfır ortalamalı beyaz Gauss gürültüsü ile ölçümlerin bozulduğu kabul edilmektedir. Filtre kazanç değerinin sabit katsayılı olması uygulama kolaylığı getirmektedir. Bu filtre sadece konuma ait ölçümler alınabildiği durumda kullanılır. Filtreye ait bağıntılar aşağıda verilmektedir:

$$Xs(k) = Xp(k) + \alpha[Xm(k) - Xp(k)] \quad (2.1)$$

$$Vs(k) = Vp(k) + \frac{\beta}{qT} [Xm(k) - Xp(k)] \quad (2.2)$$

$$Xp(k) = Xs(k-1) + TVs(k-1) \quad (2.3)$$

$$Vp(k) = Vs(k-1) \quad (2.4)$$

$Xm(k)$: k anında ölçülen konum

$Xs(k)$: k anında filtre ile düzeltilmiş konum

$Xp(k)$: k-1 anında k zamanı için yapılmış konum tahmini

T: Ölçümler arası geçen zaman

α, β : Sabit katsayılı filtre parametreleri

q: Son ölçümden sonra geçen tur sayısı

Benzer şekilde V hızı temsil eder. Başlangıç değeri olarak hız için ortalama hız alınır:

$$\begin{aligned} Vs(2) &= \frac{Xm(2) - Xm(1)}{T} \\ Vs(1) &= 0 \\ Xs(2) &= Xp(2) = Xm(2) \\ Xs(1) &= Xp(1) = Xm(1) \end{aligned} \quad (2.5)$$

α, β katsayıları Kalata, P.R.,(1984) ve Blackman, S.S.,(1986)'in gösterdiği gibi aşağıdaki denklemlerin çözümü ile bulunur:

$$T^4 \frac{\sigma_m^2}{\sigma_n^2} = \frac{\beta^2}{1-\alpha}$$

$$\beta = 2(2-\alpha) - 4\sqrt{1-\alpha} \quad (2.6)$$

Burada

σ_m^2 : Modelleme hatası varyansı

σ_n^2 : Ölçüm hatası varyansı

Alfa-Beta filtresinin başarımı, varyans azaltma oranı ile ölçülür. Varyans azaltma oranı, filtrenin dengedeki çıktısının (hız ve konum) varyansının ölçüm gürültüsü varyansına oranıdır. Filtre başlangıçtaki hatalardan oluşan salınımlardan sonra dengeye oturmaktadır.

Varyans azaltma oranları:

$$K_x = \frac{\sigma_x^2}{\sigma_n^2} = \frac{2\alpha^2 + \beta(2-3\alpha)}{\alpha(4-\beta-2\alpha)} \quad (2.7)$$

$$K_v = \frac{\sigma_v^2}{\sigma_n^2} = \frac{2\beta^2}{\alpha(4-\beta-2\alpha)T^2} \quad (2.8)$$

Burada

σ_x^2 : Düzeltilmiş konum (filtre sonucu bulunan konum) varyansı

σ_v^2 : Düzeltilmiş hız (filtre sonucu bulunan hız) varyansı

Filtrenin dinamikleri düşünüldüğünde $\alpha - \beta$ filtresinin sabit hızlı bir hedefi takip edebildiği ve filtre dengeye ulaştığında sıfır ortalama hata ürettiği görülür. Hedef ivme yapmaya başladığında ise filtre dengede sıfırdan büyük bir ortalama hataya sahip olmaktadır.

Bu filtrenin hesap yükü her turda $C = 5$ S (toplam) + 3 M (çarpım) 'dır.

2.1.2 Alfa – Beta - Gama Filtresi

Alfa-Beta filtresinin genişletilmiş hali Alfa-Beta-Gama filtresidir. Alfa-Beta-Gama filtresi Alfa-Beta'ya ek olarak ivme tahmini de yapabilmektedir:

$$Xs(k) = Xp(k) + \alpha[Xm(k) - Xp(k)] \quad (2.9)$$

$$Vs(k) = Vp(k) + \frac{\beta}{qT}[Xm(k) - Xp(k)] \quad (2.10)$$

$$As(k) = Ap(k) + \frac{\gamma}{2T^2}[Xm(k) - Xp(k)] \quad (2.11)$$

$$Xp(k) = Xs(k-1) + TVs(k-1) + 0.5As(k-1)T^2 \quad (2.12)$$

$$Vp(k) = Vs(k-1) + TAs(k-1) \quad (2.13)$$

$$Ap(k) = As(k-1) \quad (2.14)$$

Filtrenin başlangıç değerleri olarak Alfa-Beta'daki gibi hız için ortalama hız değeri alınırken ivme için de ortalama ivme kullanılır:

$$\begin{aligned} Xs(2) &= Xp(2) = Xm(2) \\ Xs(1) &= Xp(1) = Xm(1) \\ Vs(2) &= \frac{Xm(2) - Xm(1)}{T} \\ Vs(1) &= 0 \\ As(2) &= \frac{Vs(2) - Vs(1)}{T} \\ As(1) &= 0 \end{aligned} \quad (2.15)$$

Alfa-Beta-Gama ($\alpha - \beta - \gamma$) filtresi ivmeyi sabit almaktadır. Böylece hedef manevra yapsa bile filtre ivme hesabı da yaparak ivmeyi bulabilmekte böylece hız ve konum değerlerini ivmeye göre düzeltebilmektedir.

Filtre katsayıları Kalata, P.R.,(1984) ve Blackman, S.S.,(1986)'in gösterdiği gibi aşağıdaki denklemlerin çözümü ile bulunur:

$$T^4 \frac{\sigma_m^2}{\sigma_n^2} = \frac{\gamma^2}{4(1-\alpha)} \quad (2.16)$$

$$\beta = 2(2-\alpha) - 4\sqrt{1-\alpha} \quad (2.17)$$

$$\gamma = \beta^2 / \alpha \quad (2.18)$$

α, β, γ filtresi için ölçüm varyansı σ_n^2 alınırsa varyans azalma oranları Blackman, S.S.,(1986)'da gösterildiği gibi bulunur:

$$Kx = \frac{\sigma_x^2}{\sigma_n^2} = \frac{2\beta(2\alpha^2 + \beta(2 - 3\alpha)) - \alpha\gamma(4 - 2\alpha - \beta)}{(2\alpha\beta + \alpha\gamma - 2\gamma)(4 - \beta - 2\alpha)} \quad (2.19)$$

$$Kv = \frac{\sigma_v^2}{\sigma_n^2} = \frac{4\beta^3 - 2\beta^2\gamma + 2\gamma^2(2 - \alpha)}{(2\alpha\beta + \alpha\gamma - 2\gamma)(4 - \beta - 2\alpha)T^2} \quad (2.20)$$

$$Ka = \frac{\sigma_a^2}{\sigma_n^2} = \frac{4\beta\gamma^2}{(2\alpha\beta + \alpha\gamma - 2\gamma)(4 - \beta - 2\alpha)T^4} \quad (2.21)$$

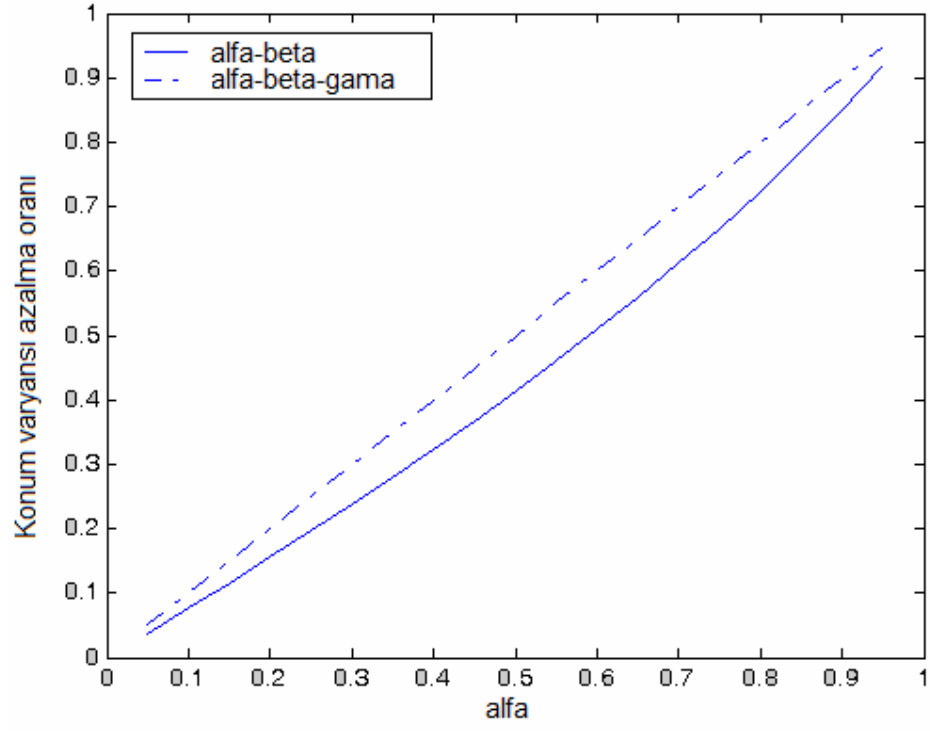
Burada σ_a^2 düzeltilmiş ivme varyansıdır.

Bu filtrenin hesap yükü her adımda $C = 9 \text{ S (toplam)} + 7 \text{ M (çarpım)}$ 'dir.

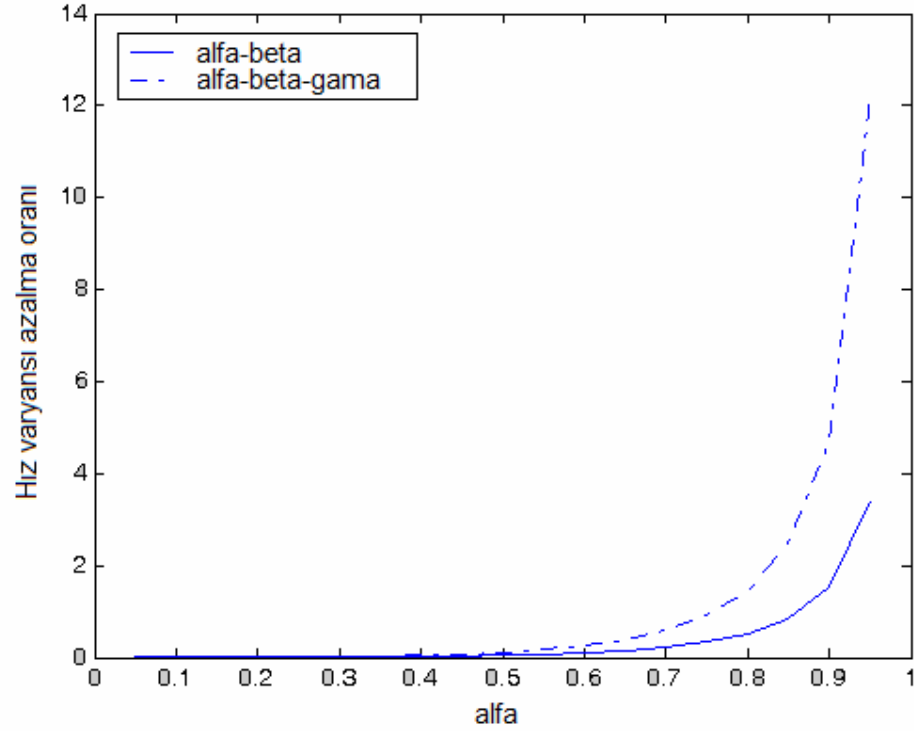
Sabit katsayılar α, β, γ nasıl seçilmelidir ?

Filtreleme bağıntılarından görüldüğü gibi filtre çıktısı olan düzeltilmiş değerler aslında ölçüm ile tahminin ağırlıklı ortalamasıdır. Katsayılar küçültülürse sonuçlar yapılan tahminlere yaklaşır. Böylece filtre gürültülü ölçümleri daha az kullanır ve ölçümlerden daha az etkilenir. Bu da filtrenin başarımını artırır. Şekil 2.1, Şekil 2.2 ve Şekil 2.3 'de varyans azalma oranının filtre katsayıları ile olan bağıntısı görülmektedir. Tam tersi durumda, yani katsayılar artırıldığında filtre alınan ölçümlere daha fazla ağırlık verir. Böylece filtre hedefin dinamik değişimlerine daha duyarlı olur. Filtre yapılan tahminleri daha az kullanır. O halde denebilir ki cisim sabit hızda kinematığını hiç bozmadan ilerliyorsa filtre dengeye ulaştığında yaptığı tahminler gerçeğe yakın olacaktır. Bu durumda başarımı daha da artırmak için yapılan tahminlere ağırlık verilmeli, ölçümlerin katsayısı düşürülmelidir. Cisim manevra yapıyorsa, filtrenin yapılan manevrayı daha çabuk öğrenebilmesi için alınan ölçümlerle daha fazla beslenmesi gerekir. Bu durumda yeni dinamiklere adapte olabilmesi için filtrenin katsayısı artırılmalıdır.

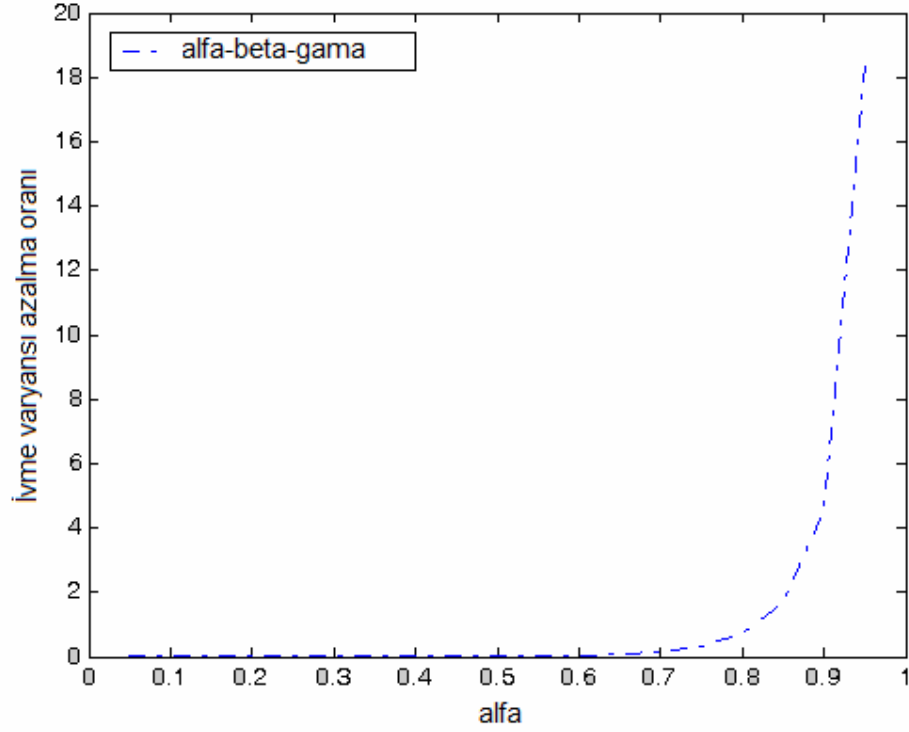
Unutulmamalıdır ki filtre katsayıları filtre ancak dengeye ulaştığında başarı sağlar. Filtre oturuncaya kadar geçen sürede büyük hatalar gözlenebilir. Şekil 2.1, Şekil 2.2 ve Şekil 2.3'de varyans azalma oranının alfa katsayısıyla ilişkisi verilmektedir.



Şekil 2.1 Konum varyansı azalma oranı



Şekil 2.2 Hız varyansı azalma oranı



Şekil 2.3 İvme varyansı azalma oranı

Şekil 2.1, Şekil 2.2 ve Şekil 2.3'de görüldüğü gibi alfa küçüldüçe varyans azalma oranı da küçülmektedir. Bu da başarımın artması demektir. Küçük alfa, filtre dengeye ulaştığında, cisim de hareketini değiştirmiyorsa, tahmin değerleri daha ağırlıklı ölçümler daha az kullanılacağı için daha az pozisyon, hız ve ivme hatası anlamına gelir.

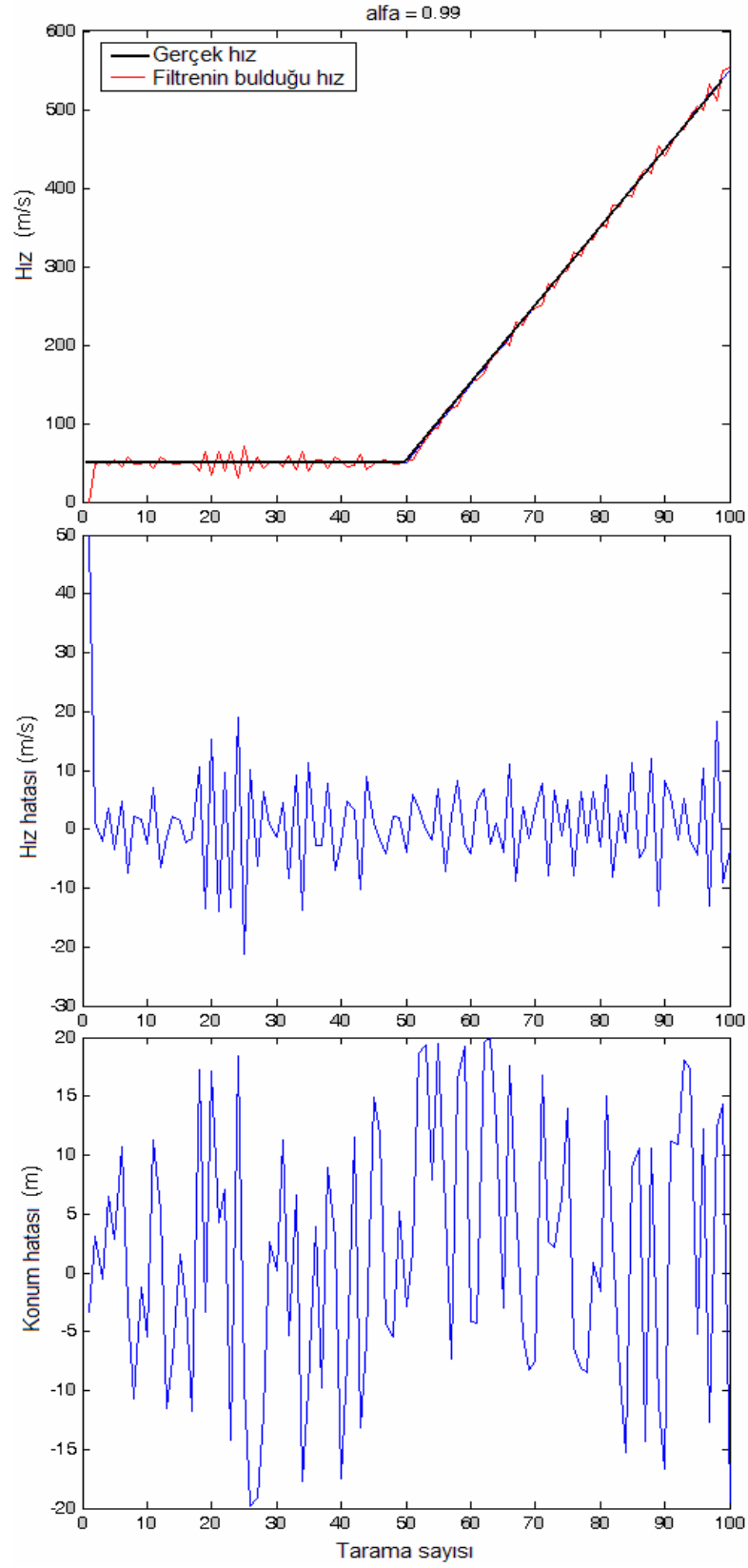
Şekil 2.1 ve Şekil 2.2'de dikkat edilecek bir diğer nokta, alfa-beta filtresinde aynı alfa değeri için konum ve hız değerindeki varyans azalma oranı, alfa-beta-gama filtresinden daha iyidir.

2.1.3 Alfa-Beta ve Alfa-Beta-Gama filtrelerinin testi

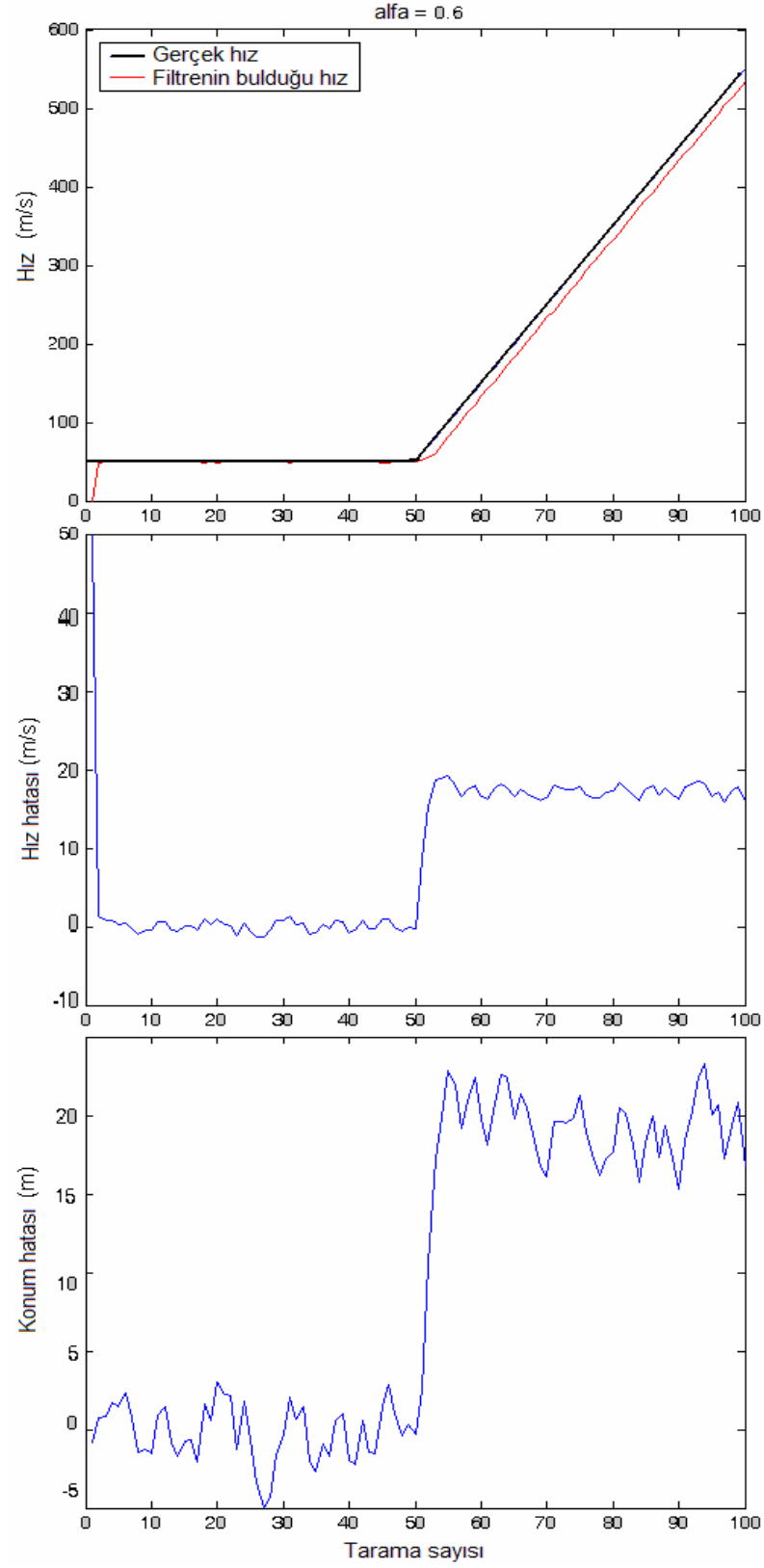
Alfa-beta filtresi ile alfa-beta-gama filtresi iki farklı alfa değeri için MATLAB'de test edilmiştir. Beta ve gama değerleri ise (2.17) ve (2.18) denklemleri kullanılarak alfa'dan bulunmuştur. Bu testte bir hedef 50m/s hızla 250s hareket etmektedir. 250s sonra hareket yönünde 2m/s^2 ivme ile hızlanarak yoluna devam etmektedir. 5 saniye aralıklarla bu hedeften ölçüm alınmakta ve en fazla 20m ölçümde hata yapılmaktadır.

2.1.3.1 Alfa-Beta filtresinin test sonuçları

Şekil 2.4'te $\alpha = 0.99$ için ve Şekil 2.5'te ise $\alpha = 0.6$ için Alfa-Beta filtresinin başarımları görülmektedir.



Şekil 2.4 Alfa-Beta filtresinin alfa = 0.99 için başarımı



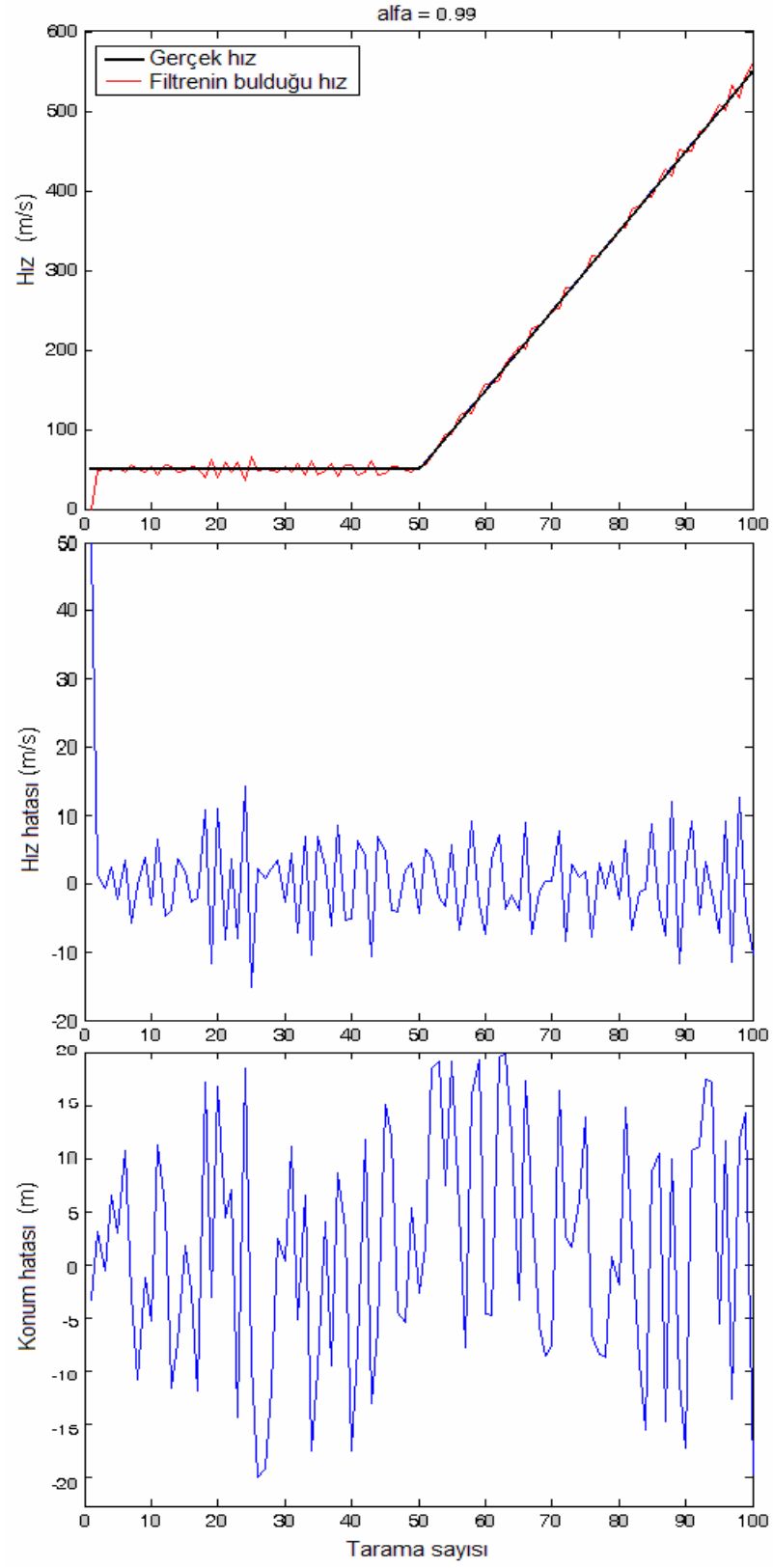
Şekil 2.5 Alfa-Beta filtresinin $\alpha = 0.6$ için başarımı

Test sonuçlarına bakılarak aşağıdaki yorumlar çıkarılabilir:

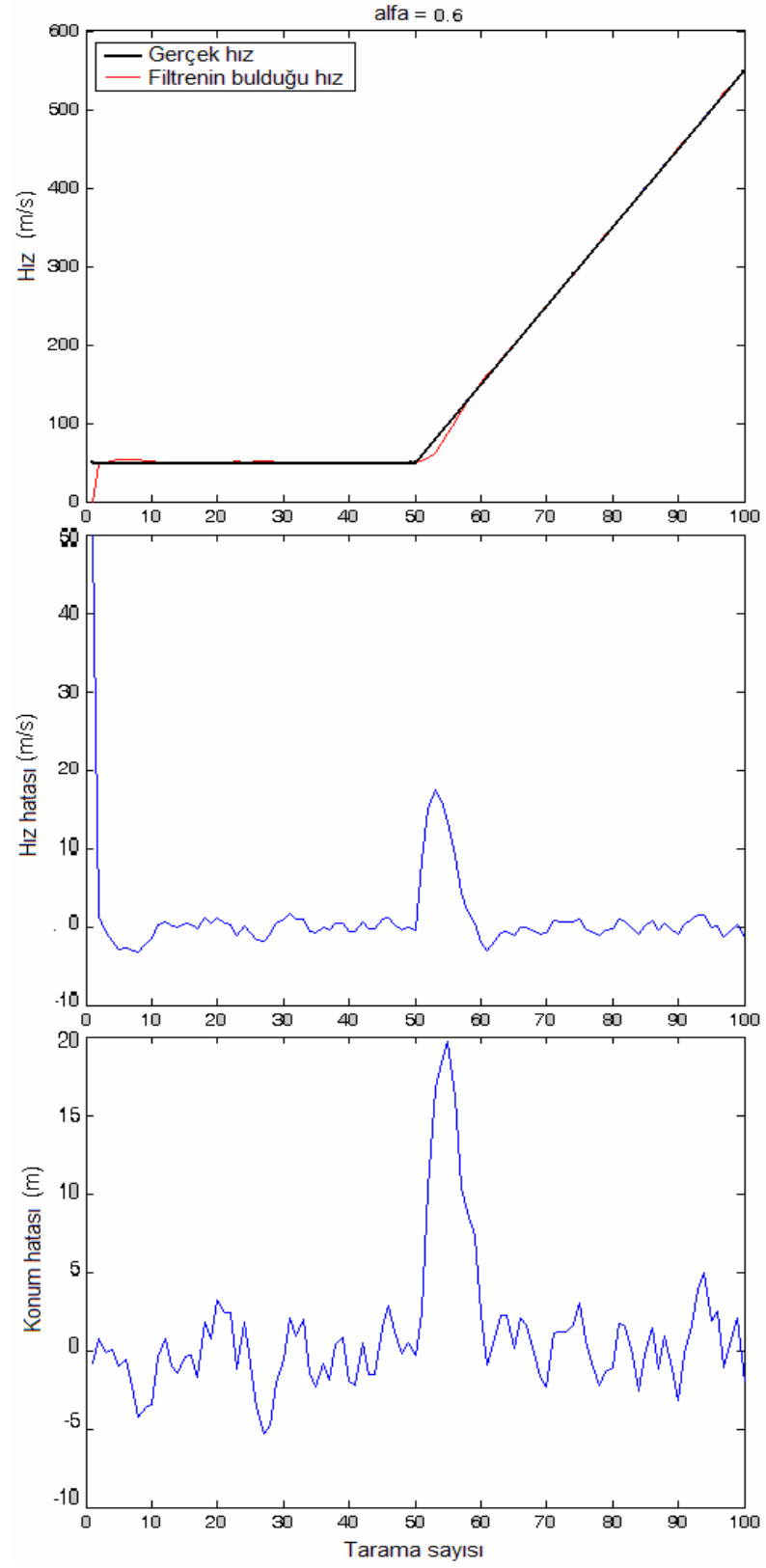
1. İlk 50 ölçümde hedef sabit hızla hareket etmektedir. Filtre, ortalamada sıfır hatalı sonuçlar üretmekte olup hatanın büyüklüğü, küçük alfa seçildiğinde daha az olmaktadır. Bu sonuç varyans azalma şekillerinde de görülmektedir.
2. 51. ölçümden sonra, hedef hızlanmaya başlamaktadır. Filtre bu sefer ortalamada sıfırdan farklı konum ve hız hataları üretmeye başlıyor. Hatadaki varyans yine küçük alfada daha küçük ancak sıfırdan farklı ortalama hata (hızda 20m/s, konumda 80m) o kadar büyük olmakta ki büyük alfa daha çok tercih edilebilir.

2.1.3.2 Alfa-Beta-Gama filtresinin test sonuçları:

Aynı senaryo Alfa-Beta-Gama filtresiyle test edildiğinde bu sefer filtrenin ivme hesabı da yaptığı, böylece konum ve hız tahmini yaparken ivmeyi de hesaba kattığı görülmektedir. Böylece, filtre, cismin ivmelenmesinden kaynaklanan hatayı kapatabilmektedir. Şekil 2.6'da $\alpha = 0.99$ için ve Şekil 2.7'de ise $\alpha = 0.6$ için Alfa-Beta-Gama filtresinin başarımları görülmektedir.

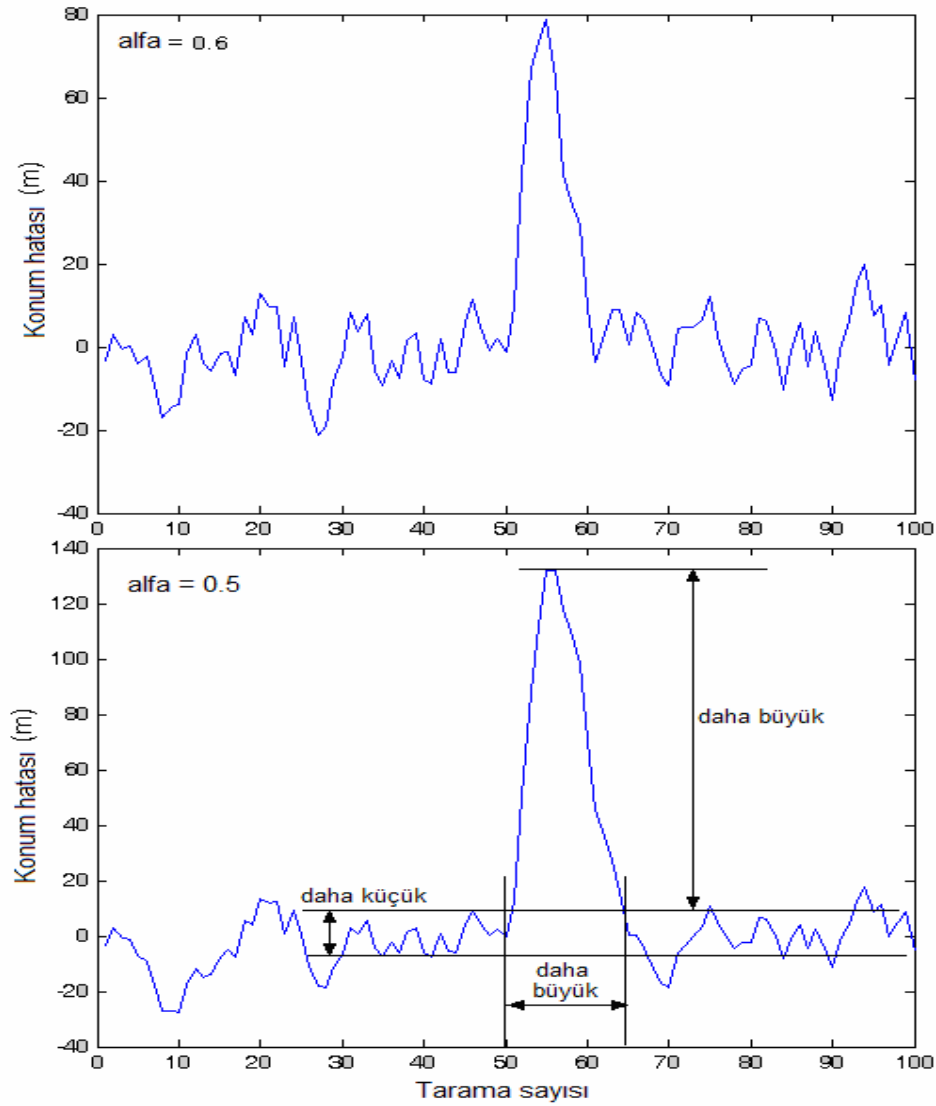


Şekil 2.6 Alfa-Beta-Gama filtresinin alfa = 0.99 için başarımı



Şekil 2.7 Alfa-Beta-Gama filtresinin $\alpha = 0.6$ için başarımı

Şekil 2.6 ve Şekil 2.7’den görüldüğü gibi sabit hızlı ve ivmeli hareket durumlarında filtre dengeye ulaştıktan sonra ortalamada sıfırdan farklı hata üretilmemektedir. Küçük alfa değerinde konum ve hız hataları daha küçüktür. Ayrıca cisim 51. gözleminde (hareket dinamiği değiştiği anda) hatada ani ancak kısa süreli bir artış olmuştur. Bu hatayı filtre 10 ölçüm sonucunda düzeltmektedir. Alfa daha da azaltılırsa denge anında filtre başarımı daha da artacak ancak cismin dinamiğinin değiştiği anlarda hatanın boyutu daha büyük, filtrenin hatayı kapatma süresi de daha uzun olacaktır. Bu analiz Şekil 2.8’de gösterilmektedir.

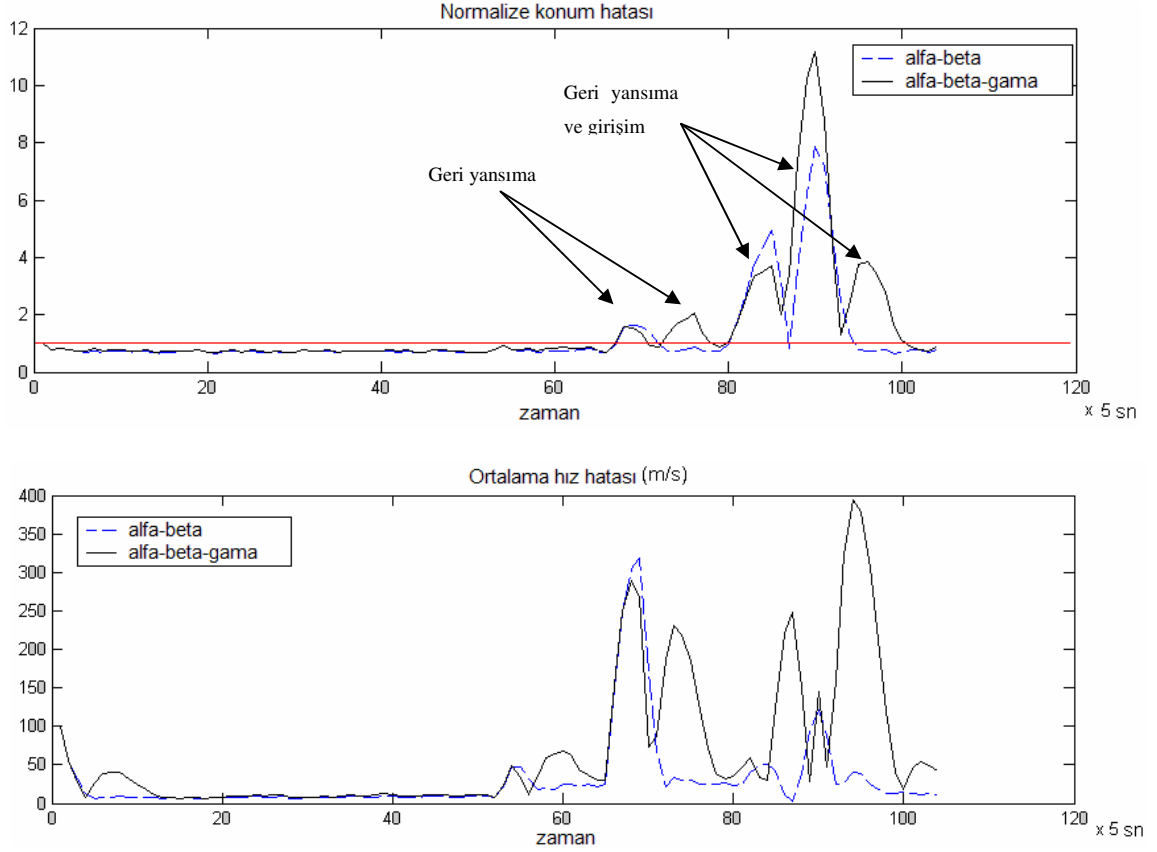


Şekil 2.8 Hedef dinamiği değişiminde Alfa-Beta-Gama filtresinin alfa'ya göre tepkisi

Görüldüğü gibi alfanın 0.6'dan 0.5'e düşürülmesi (0.1'lik bir azalma) konumdaki maksimum hatayı 50m artırmakta, düzelme süresini ise 10 turdan 15 tura çıkarmaktadır.

2.1.4 Genel Senaryo Test Sonuçları

Alfa-Beta ve Alfa-Beta-Gama filtrelerinin genel senaryo ile 100 Monte Carlo testinin sonucu Şekil 2.9'da verilmiştir. Bu testte $\alpha = 0.6$ seçilmektedir, beta ise (2.17) denkleminde α kullanılarak bulunmaktadır.



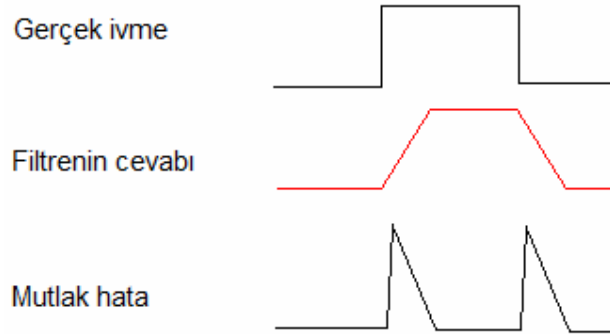
Şekil 2.9 Alfa-beta ve alfa-beta-gama filtrelerinin $\alpha=0.6$ için normalize konum ve ortalama hız hatası karşılaştırması

2.1.5 Sonuç

Bu bölümdeki bütün analiz ve testler sonucunda alfa-beta ile alfa-beta-gama filtreleri için aşağıdaki gibi genel bir yorumda bulunulabilir:

1. Sabit hızla doğrusal giden bir hedef için alfa-beta filtresi alfa-beta-gama filtresinden daha yüksek başarımlar sağlar. Bu durum varyans azalma oranları grafiklerinde görülmektedir.
2. İvme yapan hedefler için, alfa-beta-gama filtresi daha başarılı sonuçlar verir. Bu durum hem varyans azalma oranları grafiklerinden hem de benzetim sonuçlarından görülebilir. Ancak genel senaryoda durum böyle olmamıştır. Şekil 2.9'daki

normalize konum hatalarından da bu görülebilmektedir. Burada sebep olarak uçağın ardarda değişik manevralar yapmış olmasını gösterebiliriz. Ardarda manevralar sırasında alfa-beta-gama filtresi değişimlere uyum sağlayacak (diğer bir deyişle ivmeyi hesap edecek) gerekli zamanı bulmadan hedef yeni bir manevraya başlamaktadır. Manevra sırasında hesap edilen ivme diğer kinematikler olan hız ve konum hesaplamasında kullanılacağı için manevra değişim anında filtre çıktılarında büyük hatalar üretilmektedir. Örneğin hedef ivmesini 0'dan 1G'ye yükselttiğinde Alfa-Beta-Gama filtresinin bunu hesaplaması için en az 3 tur geçmesi gerekmektedir. Bu arada ivme öğrenilene kadar hatalar oluşacaktır. 4. veya 5. turda cisim ivmesini yeniden 0'a düşürdüğünde Alfa-Beta-Gama filtresinin tekrar bu değişimi hesaplaması için en az 3 tur gerekecektir. Bu öğrenme sürecinde de yeniden hatalar oluşacaktır. Sonuç olarak Alfa-Beta-Gama filtresi kullanıldığında, hedef bir manevralı harekete başlayıp onu sonlandırdığında 6 tur boyunca hatalı sonuç üretilirken hesapladığı ivmeyi sadece 1 veya 2 tur kullanabilmektedir. Gerçek ortamda yüksek manevra yapan savaş uçaklarının pilotlarının bu manevraları çok fazla sürdürmedikleri bilinmektedir. Ortalama 20-25 saniye sürdürülen bir manevra için ancak 4-5 ölçüm alınabilir. Tabii ki bu durum periodu 5s olan radarlar için geçerlidir. Gözlem sıklığı daha yüksek olan yakın mesafe radarlarında yeniden kazanç kaybı hesapları yapılmalıdır. Bu durum Şekil 2.10'da gösterilmektedir:



Şekil 2.10 İvmeye dalgalanmanın oluşturduğu mutlak hata

3. Alfa, başlangıçtaki dalgalanmaları bastırabilmek için yeterince büyük seçilmelidir. Filtre birkaç turdan sonra dengeye oturacaktır. Filtre dengeye oturduktan sonra ise alfa değeri basamak basamak küçültülmelidir çünkü küçük alfa ile filtre daha başarılı sonuçlar elde edecektir.

Bu strateji filtrelerin performansları gözönüne alındığında tek başına alfa-beta ve alfa-beta-gama filtreleri arasında en iyi sonucu verir. Başka bir yaklaşımda her iki filtre aynı anda kullanılırken ihtiyaca göre alfa-beta'dan alfa-beta-gama'ya geçiş düşünülebilir. Ancak cisim manevrasını sürekli değiştiriyorsa veya manevrayı uzun süre devam ettirmiyorsa bu durumda yukarıdaki sebepten dolayı alfa-beta-gama iyi bir seçenek değildir. Bunun yerine filtreyi sabit tutup alfa'yı artırıp azaltmak daha iyi bir seçenektir.

2.2 Alfa-Beta ve Alfa-Beta-Gama Filtrelerinde İyileştirme Yaklaşımı: Uyarlamalı Alfa-Beta ve Uyarlamalı Alfa-Beta-Gama Filtreleri

2.1.5 sonuç kısmında açıklandığı gibi alfa-beta ve alfa-beta-gama filtreleri, manevra yapılmadığı zamanlar küçük alfa, manevralı anlarda ise büyük alfa değerlerini kullandığında daha başarılı olmaktadır. Bu durumda hedef manevra yaptığınd alfa'yı büyütecek, manevra bittiğinde ise kademe kademe alfa'yı küçültecek bir mekanizma düşünülmektedir. Buradaki mantık şu şekildedir:

Hedef manevra yapmaya başladığında alfa artırılır böylece ölçümlere ağırlık verilmiş olur. Manevrayı belirlemek için filtrelemedeki yenilenme miktarının(ölçüm(i)-tahmin(i/i-1)) belirli bir eşik seviyesini geçmesi gerekir. Bu eşik seviyesi aşıldığında filtre, hedefin manevra yaptığına karar verir ve alfa tekrar en büyük seviyesine çıkarılır. İlerleyen turlarda alfa kademe kademe azaltılacaktır.

eğer ($\alpha > \text{ALPHA_MIN}$)

$\alpha = \alpha * k;$

eğer ($\alpha < \text{ALPHA_MIN}$)

$\alpha = \text{ALPHA_MIN};$

eğer ((yenilenme $\geq m * \text{ortalamaÖlçümHatası}$) && ($\alpha == \text{ALPHA_MIN}$))

$\alpha = \text{ALPHA_MAX};$

Burada k azalma faktörüdür.

Başlangıç alfa = 0.6

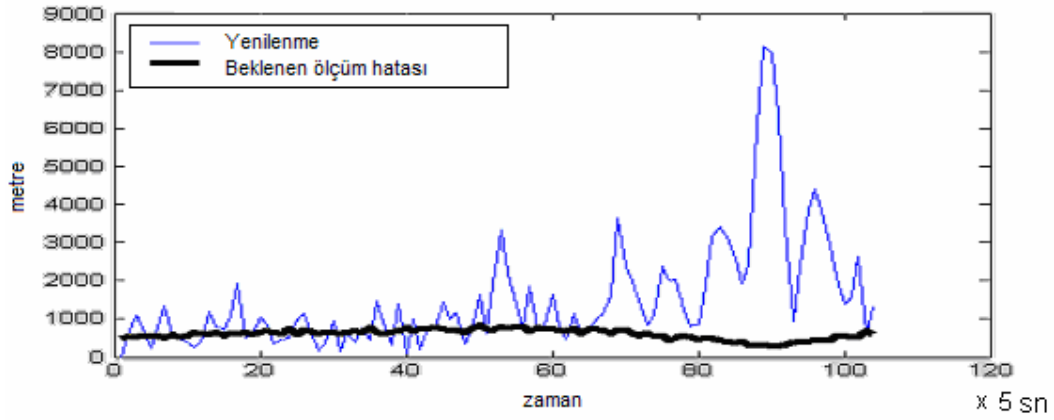
alfa_min = 0.6

alfa_max = 0.99

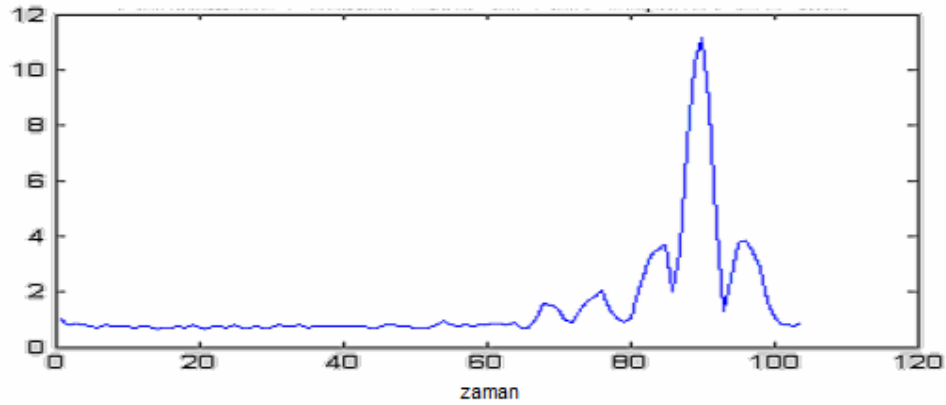
k = 0.95

m = 4

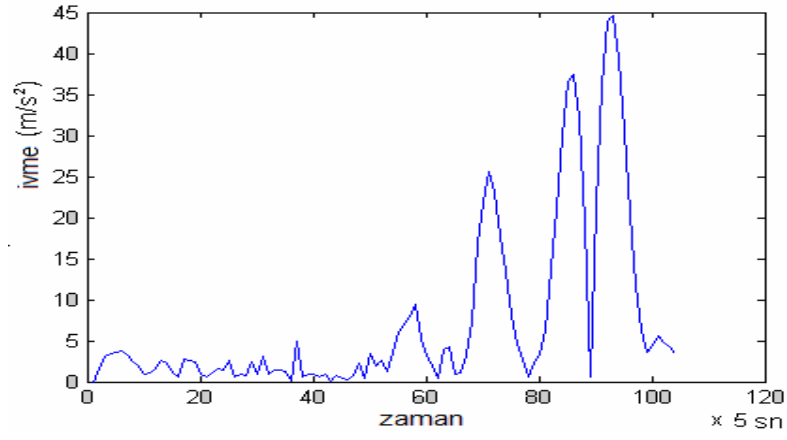
Bu yaklaşımın kullanılmadan önceki normalize konum hatası Şekil 2.12’de verilmektedir. Eşik seviyesi belirlemede kullanılan hata kıstasları Şekil 2.11’de verilmektedir. Alfa-Beta-Gama filtresinin bulduğu ivme değerleri ile uyarlamalı yaklaşımın alfa değerini nasıl değiştirdiği Şekil 2.13 ve Şekil 2.14’te verilmektedir. Uyarlamalı yaklaşım sonucu yeni normalize konum hatası Şekil 2.15’te görülmektedir.



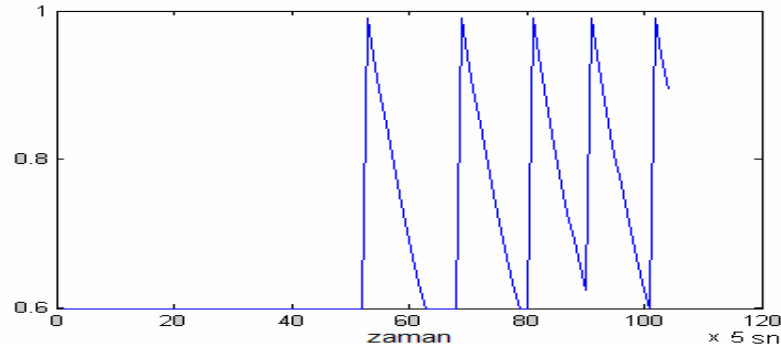
Şekil 2.11 Normal Alfa-Beta-Gama filtresinde yenilenme ve ölçüm hatası



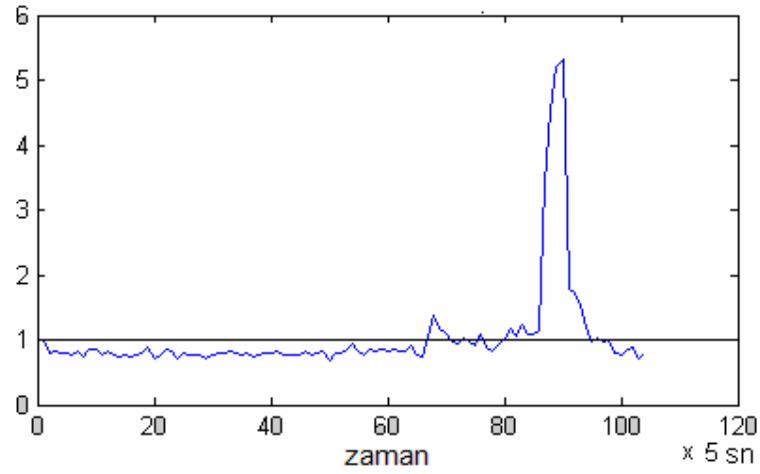
Şekil 2.12 Normal Alfa-Beta-Gama filtresinde normalize konum hatası



Şekil 2.13 Alfa-Beta-Gama filtresinin bulduğu ivme

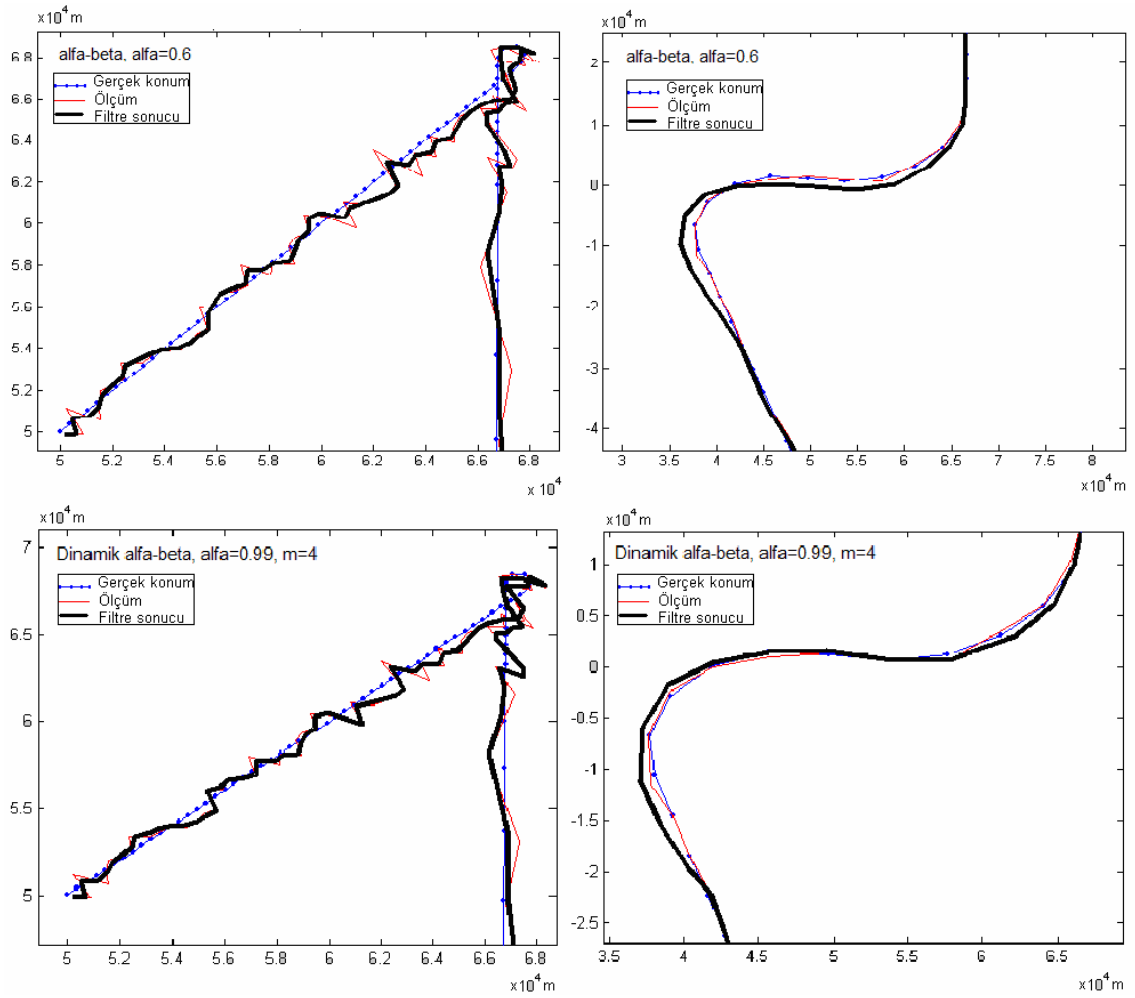


Şekil 2.14 Alfa parametre değişimi (yeni yaklaşımda manevra yakalama)

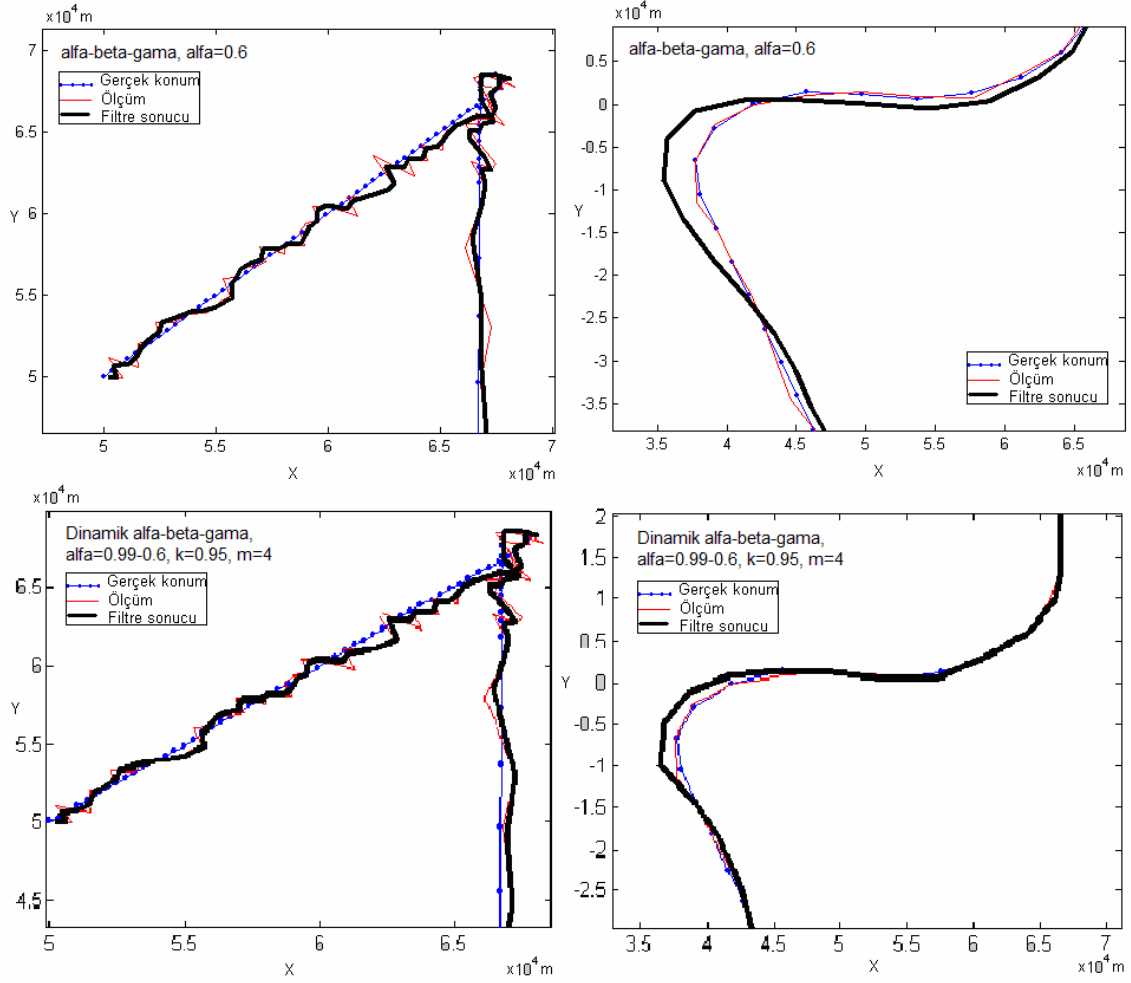


Şekil 2.15 Uyarlamalı Alfa-Beta-Gama filtresinde normalize konum hatası

Şekil 2.12 ve Şekil 2.15 karşılaştırıldığında normalize konum hatasına göre yeni dinamik yaklaşımın başarılı olduğu ve dinamik olmayan yaklaşıma göre hatayı büyük miktarda azalttığı görülmektedir. Sonuçlar alfa azalma faktörü k ile oynanarak daha da başarılı yapılabilir. Bu yaklaşım ile hem alfa-beta-gama filtresinin hesaplama yükündeki avantajdan yararlanılırken hem de dinamik değişimlere uyumluluk sağlanmıştır. Şekil 2.16 ve Şekil 2.17’de normal Alfa-Beta ve Alfa-Beta-Gama filtrelerinin Uyarlamalı Alfa-Beta ve Uyarlamalı Alfa-Beta-Gama filtreleri ile senaryo üzerindeki rota başarımlarını karşılaştırmaları verilmektedir. Uyarlamalı Alfa-Beta ve Alfa-Beta-Gama filtrelerinin uyarlamasız olanlara göre rotaya daha yakın sonuç ürettikleri görülmektedir.

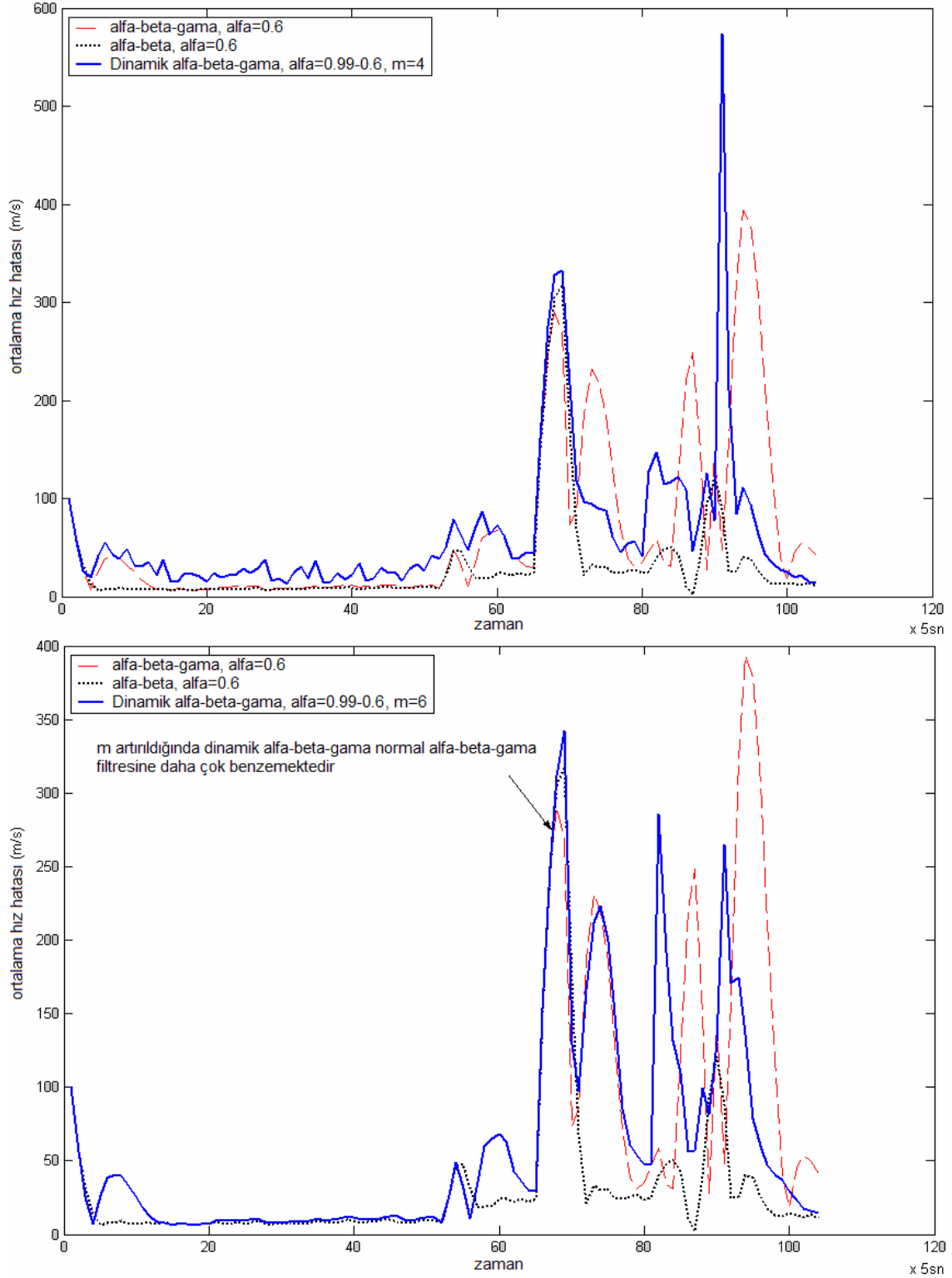


Şekil 2.16 Normal Alfa-Beta ile Uyarlamalı Alfa-Beta filtrelerinin rotaya göre başarımlarının karşılaştırması

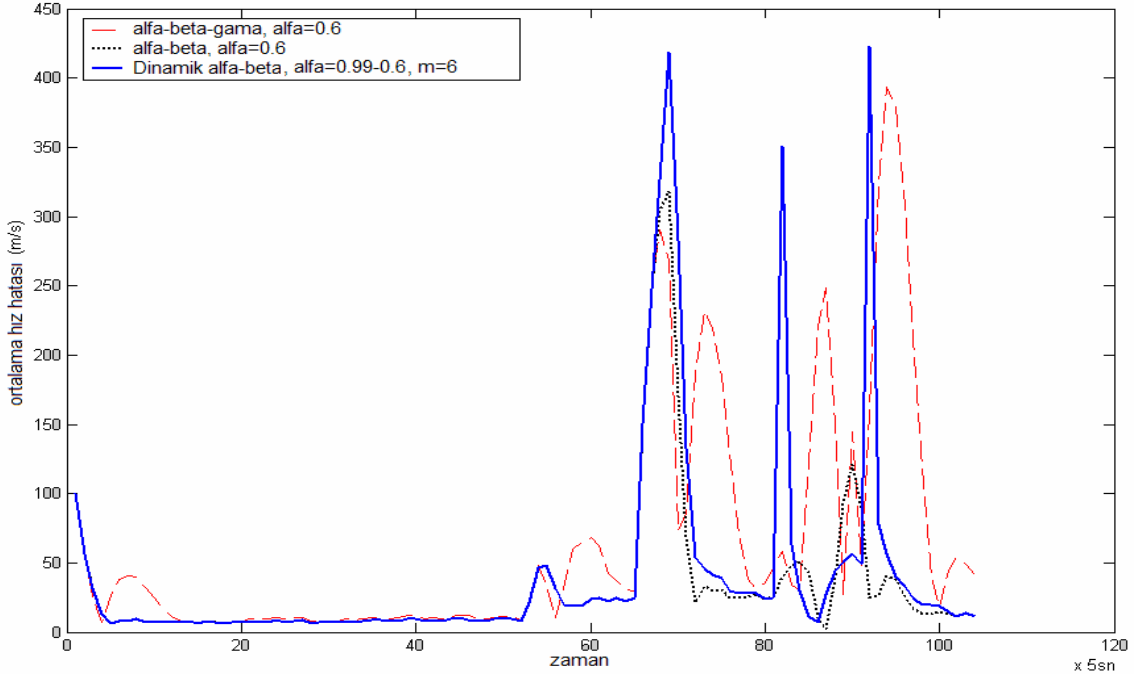


Şekil 2.17 Normal Alfa-Beta-Gama ile Uyarlamalı Alfa-Beta-Gama filtrelerinin rotaya göre başarılarının karşılaştırması

Şekil 2.18 ve Şekil 2.19 da ise normal Alfa-Beta ve Alfa-Beta-Gama filtrelerinin Uyarlamalı Alfa-Beta-Gama filtresi ile farklı m değerleri için ortalama hız hatası karşılaştırması verilmektedir. Yine hız hataları dikkate alındığında uyarlamalı yaklaşımın normal filtrelere göre daha küçük hız hatası ürettiği görülmektedir. Özellikle Alfa-Beta-Gama filtresinde manevra sonrası oluşan geri yansıma hatasının Uyarlamalı Alfa-Beta-Gama filtresinde oluşmadığı görülmektedir.



Şekil 2.18 Normal Alfa-Beta ve Alfa-Beta-Gama filtrelerinin Uyarlamalı Alfa-Beta-Gama filtresi ile farklı m değerleri için ortalama hız hatası karşılaştırması



Şekil 2.19 Normal alfa-beta ve alfa-beta-gama filtrelerinin uyarlamalı alfa-beta filtresi ile farklı m değerleri için ortalama hız hatası karşılaştırması

Alfa-Beta, Alfa-Beta-Gama, Uyarlamalı Alfa-Beta ve Uyarlamalı Alfa-Beta-Gama Arasında Genel Değerlendirme:

1. Ortalama hız hatası ile normalize konum hataları değerlendirildiğinde AB filtresi ABG filtresinden daha başarılıdır. Teorik olarak Alfa-Beta-Gama filtresinin manevralı anlarda daha başarılı olacağı düşünülür ancak bu durumun böyle olmadığı, yansıma hatalarından dolayı Alfa-Beta-Gama filtresinin daha büyük hatalar ürettiği Şekil 2.9 ve Şekil 2.18’de görülmektedir.
2. Benzetim sonuçlarına göre önerilen yeni dinamik yaklaşımın performansı artırdığı görülmüştür. Konum tahminlerinde bütün senaryolar düşünüldüğünde genel olarak bir performans artışı olmuştur. Bu gelişim hız tahminlerinde ise konumdaki kadar fazla değildir. Şekil 2.18 ve Şekil 2.19’da görüldüğü gibi hız hataları manevralı anlarda normal Alfa-Beta ve Alfa-Beta-Gama filtrelerine göre daha fazladır. Ancak senaryonun bütünü düşünüldüğünde (konum + hız hataları) dinamik yaklaşım başarıyı artırmıştır.
3. En başarısızdan en başarılıya doğru filtreler Alfa-Beta-Gama, Uyarlamalı Alfa-Beta-Gama, Alfa-Beta, Uyarlamalı Alfa-Beta olarak sıralanabilir.

2.3 Kalman Filtresi

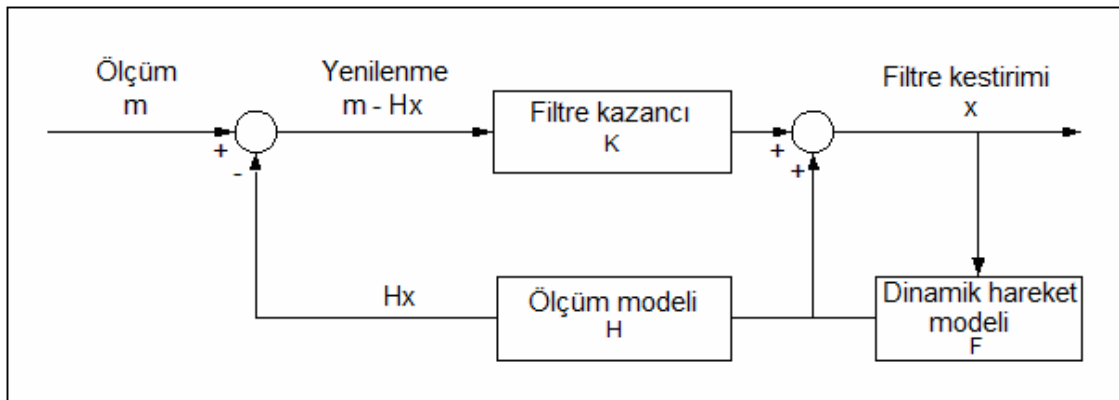
Artık gelenekselleşmiş olan ve en çok kullanılan durum tahmin filtresi, Kalman filtresidir. Kalman, R.E.,(1960)'da ispatlandığı ve de Grewal, M.S. ve Andrews, A.P.,(1993)'de açıklandığı üzere Kalman filtresi doğrusal kestirim problemlerinde (linear estimation problem) en küçük ortalama karesel hatayı verir. Aşağıdaki koşullar sağlandığında en iyi takip performansını vereceği öngörülmüştür:

1. Hedefe ait durum vektörü, doğrusal ve dinamik bir model ile sürülür ve bu model ölçüm ile beslenir.
2. Ölçüm gürültüsünün sıfır ortalamalı Gauss dağılımlı olduğu farzedilir.

En küçük ortalama karesel hatayı vermesinin yanında Kalman filtresinin önemli diğer avantajları şunlardır:

1. Filtrenin bant genişliği tanımlanmış hareket modeli ve ölçüm modeline göre otomatik olarak değişir. Böylece filtre parametreleri modelleme sınırları içinde hedefin değişen koşullarına kendiliğinden uyum sağlar.
2. Alınan ölçümlerle filtre, kovaryans matrisini kendiliğinden günceller. Kovaryans matrisi filtrenin tahminlerdeki başarımını gösterir. Bu değer manevra tahmini için önemli ipucu verir.

Kalman Filtresinin blok diyagramı Şekil 2.20'de verilmiştir:



Şekil 2.20 Kalman Filtresi Blok Diyagramı

Hedef dinamik sürecinin ayrık Markov süreci ile modellendiği düşünüldüğünde hedef dinamikleri aşağıdaki gibi yazılır:

$$X(t_{k+1}) = FX(t_k) + Gw(t_k) \quad (2.22)$$

$$E[w(t_k)w^T(t_k)] = Q(t_k) \quad (2.23)$$

Ölçüm modeli ise aşağıdaki gibidir:

$$Z(t_k) = H_k X(t_k) + v(t_k) \quad (2.24)$$

$$E[v(t_k)v^T(t_k)] = R(t_k) \quad (2.25)$$

Burada,

X: n x 1 durum vektörü

F: n x n durum değişim matrisi

G: n x 1 bozulma matrisi

w (t_k): Sıfır ortalamalı Gauss beyaz gürültüsü

Q: Süreç gürültüsü varyansı

H_k: m x n ölçüm matrisi

R: Ölçüm gürültüsü varyans matrisi

v (t_k): Sıfır ortalamalı beyaz Gauss gürültüsü

X durum vektörü genellikle hedefin konum, hız ve ivme değerini içerir. Yukarıdaki ölçüm ve hareket modeli kullanılarak Kalman filtre denklemleri aşağıda çıkarılmıştır:

Durum tahmini:

$$x(t_{k+1}/t_k) = Fx(t_k/t_k) \quad (2.26)$$

Yenilenme:

$$v(t_{k+1}) = z(t_{k+1}) - Hx(t_{k+1}/t_k) \quad (2.27)$$

Durum vektörü hata kovaryansı:

$$P(t_{k+1}/t_k) = FP(t_k/t_k)F^T + GQG^T \quad (2.28)$$

Yenilenme kovaryansı:

$$S(t_{k+1}) = HP(t_{k+1}/t_k)H^T + R \quad (2.29)$$

Filtre kazancı:

$$K(t_{k+1}) = P(t_{k+1}/t_k)H^T S(t_{k+1})^{-1} \quad (2.30)$$

Kovaryans tahmini:

$$P(t_{k+1}/t_{k+1}) = [1 - K(t_{k+1})H]P(t_{k+1}/t_k) \quad (2.31)$$

Durum kestirimi:

$$x(t_{k+1}/t_{k+1}) = x(t_{k+1}/t_k) + K(t_{k+1})v(t_{k+1}) \quad (2.32)$$

Burada:

$x(t_{k+1}/t_k)$: k+1 zamanı için k zamanında yapılmış durum tahmini

$z(t_{k+1})$: k+1 zamanında alınmış ölçüm

$v(t_{k+1})$: k+1 zamanında yapılan yenilenme miktarı

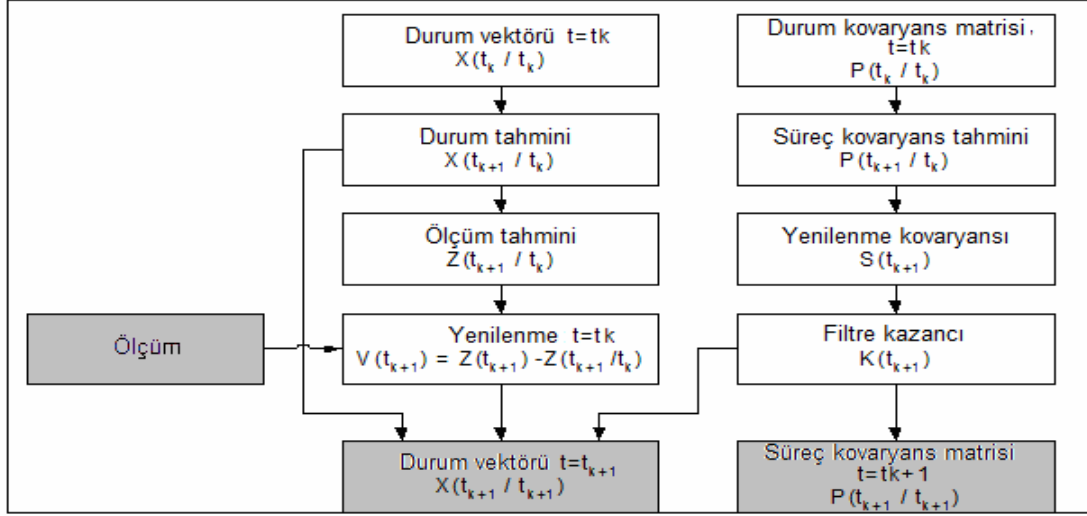
$P(t_{k+1}/t_k)$: k+1 zamanı için k zamanında durum kovaryans matrisi için yapılmış tahmin

$S(t_{k+1})$: k+1 zamanındaki yenilenme kovaryansı

H: Ölçümler için Jacobien matrisi

$K(t_{k+1})$: Filtre kazancı

Filtre akış diyagramı Şekil 2.21'de verilmiştir.



Şekil 2.21 Kalman Filtresi akış diyagramı

Kalman filtresinin başarısı veya başarısızlığı tamamen seçilen kovaryans matrisine bağlıdır. Eğer kovaryans matrisi olması gerekenden büyük seçilirse filtre dinamik değişikliklere daha duyarlı olacaktır böylece filtre yeni dinamiklere karşı kendini daha çabuk düzeltebilecektir. Büyük kovaryans filtrenin daha büyük adım atabilmesi demektir. Hedef dinamiğinde büyük değişiklik olduğunda filtre bu değişimlere büyük adımlar atarak çabucak yetişir. Ancak hedef dinamiğinde değişim olmadığı durumlarda en küçük gürültüyle birlikte filtre bu gürültülere karşı dengede kalabilmek için büyük adım atabilir. Bu da dinamiğin değişmediği doğrusal hareket anlarında gereksiz oynamalar ve toplamda hatada artışa sebep olur.

Tersi durumda ise yani filtre kovaryansı gereğinden küçük seçilirse dinamiğin değişmediği anlarda gürültülere karşı filtre fazla savrulmaz çünkü dengede kalmak için daha küçük adım atacağından filtrede fazla bir oynama olmamış olur. Ancak dinamiğin değiştiği anlarda filtrenin değişime karşı yetişmek için atabileceği adım küçük olacağından filtre değişimlere daha geç yetişir. Bu durumda da bu geçiş anlarında hatada büyüme olur. Bu daha riskli bir durumdur çünkü eğer hedef sürekli değişen manevralar yaparsa ve filtre bir manevraya yetişene kadar hedef başka manevraya başlayabilir böylece hiçbir zaman gerçek dinamiklere yetişemeyebilir. Dolayısıyla kovaryans matrisini yeterince büyük seçmek önemlidir. Alfa-beta-gama filtresinde de benzer bir durum alfa parametresinin seçimi konusunda vardı. Kalman filtresi mevcut kovaryans matrisi sınırları içinde kazanç katsayısını değiştirebilmektedir. Alfa-beta-gama filtresi sabit kazaca sahiptir. Bu durumda dinamik alfa-beta-gama yaklaşımının Kalman filtresi davranışını gösterdiğini, başarımının bu sebeple daha yüksek olduğu bir kez daha anlaşılabilmektedir.

Kalman filtresinin örnek bir senaryoda testi:

Kalman filtresinin temel incelenmesi için daha önce Alpha-Beta filtresinde kullanılan basit senaryo ile test yapılmıştır. Bu testte iki farklı kovaryans matrisi kullanılmıştır:

Süreç gürültüsü kovaryans matrisi: $Q = \text{diag}([0 \ 0 \ 1.33]), \text{diag}([0 \ 0 \ 0.0133])$ (diagonal matris)

Not: Burada matris $[\sigma_x^2 \ \sigma_v^2 \ \sigma_a^2]$ değerlerini temsil etmektedir. Süreç gürültüsü beklenmeyen ivme değişimlerinden kaynaklanmaktadır. İvmedeki ani değişimin olasılığının homojen dağıldığı ve en fazla 2m/s^2 olacağı düşünülürse varyans $= (a_{\max})^2 / 3 = 1.33$ çıkar.

Ölçüm gürültü varyansı: Benzer şekilde ölçümdeki en büyük hatanın 20m yapılacağı kabul edilirse ve ölçüm hatasının homojen dağıldığı düşünüldüğünde ölçüm gürültü varyansı

$$R = 20^2 / 3 = 133 \text{ çıkar.}$$

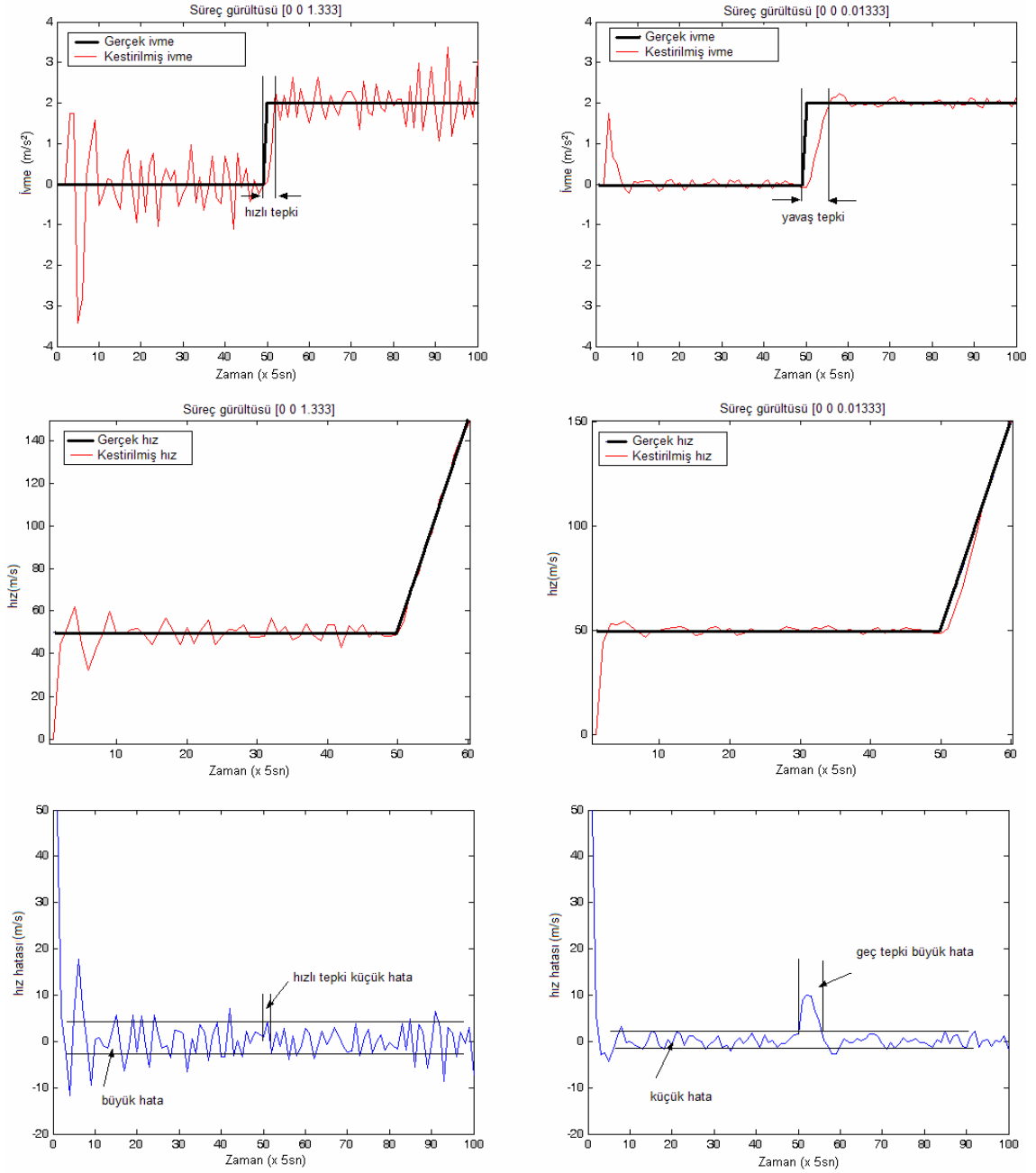
Durum kovaryans matrisi: $P = [0]_{3 \times 3}$ başlangıçta sıfırdır.

$$\text{Durum değişim matrisi: } F = \begin{bmatrix} 1 & \Delta t & 0.5\Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \Delta t = 5\text{sec} \ (\Delta t : \text{ölçümler arası geçen süre})$$

$$\text{Ölçüm matrisi: } H = [1 \ 0 \ 0]$$

$$\text{Durum vektörü: } X = [x \ Vx \ Ax]^T \ (\text{konum, hız ve ivme})$$

Seçilen değerlerle Kalman filtresinin başarımları Şekil 2.22'de verilmiştir:



Şekil 2.22 Kalman Filtresi test sonuçları

Şekil 2.22'deki figürlerden seçilen kovaryans matrisinin önemi daha net anlaşılmaktadır. Kovaryans matrisi gereğinden büyük seçildiğinde toplam hata da artmaktadır. İlk 50 gözlemden hedef ivmesiz hareket etmektedir. Bu anlarda küçük süreç gürültü kovaryansı daha az hataya yol açmaktadır. Ayrıca 51. gözlemden itibaren hedef sabit ivmeyle hareketine devam etmektedir. Filtre, burada da yapılan ivmeyi bulmuş ve dengeye oturduktan sonra aynı

performansı göstermektedir. Bu başarı Kalman filtresinin bant genişliğini değişik dinamiklere uyarlayabilme yeteneğinden gelmektedir.

Dinamikteki değişime karşı filtrelerin tepki süresi de şekillerde görülebilmektedir. Küçük kovaryanslı filtre yaklaşık 6 gözlemde bu değişimi kotarabilirken, büyük kovaryanslı filtre 1-2 gözlemde kotarabilmiştir.

Sonuç olarak tekli Kalman filtresi kullanılan çözümlerde Kalman filtresinin kovaryans matrisi muhtemel en büyük ivme yapıldığında bile oluşacak hatayı hızlıca kapatabilecek büyüklükte seçilmelidir.

Kalman Filtresinin Genel Sonuçları ve Karşılaştırma Analizi:

1. Kalman filtresi değişimlere uyumlu bir filtredir ve kazancını extra bir mekanizmaya ihtiyaç duymadan değiştirir. Alfa-Beta ve Alfa-Beta-Gama filtresine olan genel üstünlüğü buradan gelmektedir.
2. Kalman filtresi bant genişliğini değişen hedef dinamiklerine uygun olarak değiştirebilir.
3. Kalman filtresi en küçük karesel ortalama hatayı verir. Diğer filtreler Kalman filtresinin özel bir formu olarak kabul edilir.
4. En önemli dezavantajı kovaryans matrisi seçme konusunda getirdiği zorluktur ve bu matris gerçeğe çok yakın seçilmek zorundadır.
5. Kalman filtresi daha önce incelenen sabit katsayılı alfa-beta ve alfa-beta-gama filtrelerine göre daha çok işlem gücü gerektirir.

2.3.1 Kalman Filtresinin Genel Senaryo Test Sonuçları

Durum vektörü: $X = [x \ y \ V_x \ V_y \ A_x \ A_y]^T$ (konum, hız ve ivme)

Durum değişim matrisi

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0.5\Delta t^2 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0.5\Delta t^2 \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Delta t = 5s$$

Süreç gürültü varyansı :

Q süreç gürültüsü bilinmeyen ivme değişimlerinden gelmektedir ve bilinmeyen ivme değişikliği her adımda homojen dağılımlı ve en fazla 25m/s^2 olacağı kabul edilsin.

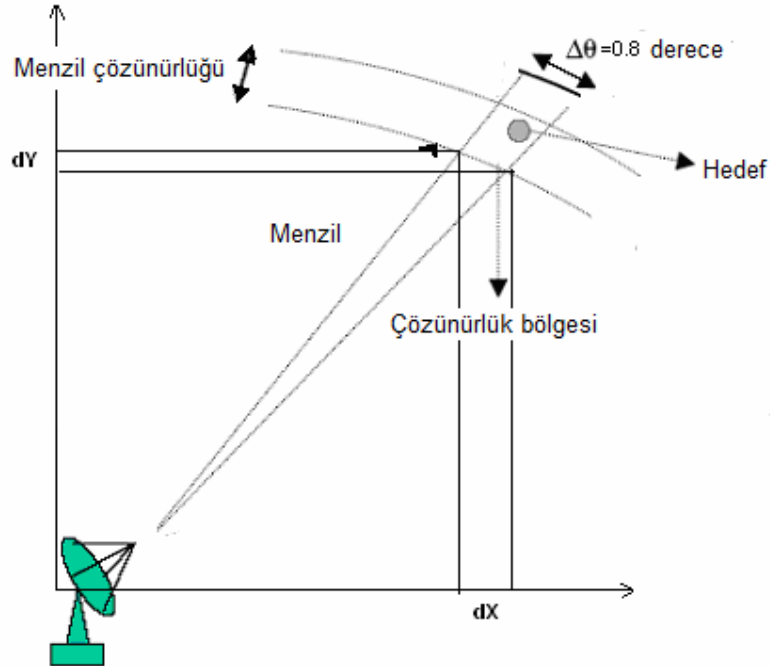
Bu durumda x ve y eksenlerinde ivme varyansı $25^2 / 3 = 308$ alınmalıdır.

$$Q = \begin{bmatrix} C_{XX} & C_{XY} & C_{XV_x} & C_{XV_y} & C_{XA_x} & C_{XA_y} \\ C_{YX} & C_{YY} & C_{YV_x} & C_{YV_y} & C_{YA_x} & C_{YA_y} \\ C_{V_xX} & C_{V_xY} & C_{V_xV_x} & C_{V_xV_y} & C_{V_xA_x} & C_{V_xA_y} \\ C_{V_yX} & C_{V_yY} & C_{V_yV_x} & C_{V_yV_y} & C_{V_yA_x} & C_{V_yA_y} \\ C_{A_xX} & C_{A_xY} & C_{A_xV_x} & C_{A_xV_y} & C_{A_xA_x} & C_{A_xA_y} \\ C_{A_yX} & C_{A_yY} & C_{A_yV_x} & C_{A_yV_y} & C_{A_yA_x} & C_{A_yA_y} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 308 & 0 \\ 0 & 0 & 0 & 0 & 0 & 308 \end{bmatrix}$$

Ölçüm gürültüsü varyansı:

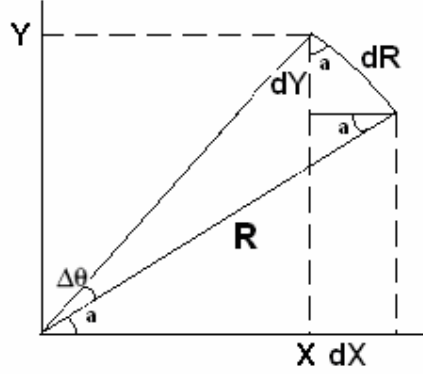
$$R = \text{diag}([\sigma_x^2, \sigma_y^2]) \text{ (diyagonal matris)}$$

Ölçüm gürültüsü varyansı menzile göre değişmektedir. Yatay açıda ölçüm açısı hatası Gauss dağılımlı ve 0 ortalamalı, standart sapma ise $\sigma = 0.4^\circ$ idi. Şekil 2.23'te radar'ın çözünürlük bölgesi gösterilmektedir.



Şekil 2.23 Radar ölçüm çözünürlük bölgesi

Açısal hataya göre menzilsel hata ihmal edilebilir çünkü 0.8 derecelik açısal yay 75km menzilde 1050m uzunluğundadır ancak menzilsel hata en fazla 100m 'dir. Bu durumda 0.8 yeterince küçük bir açı olduğu kabul edilerek aşağıdaki bağıntılar elde edilir:



$$dR = 2.\Pi.R.\frac{0.8}{360}$$

$$dX = dR.\sin a = 2.\Pi.\frac{0.8}{360}.R.\sin a \cong 2.\Pi.\frac{0.8}{360}.R.\sin(a + 0.8) = 2.\Pi.\frac{0.8}{360}.Y$$

$$dY = dR.\cos a = 2.\Pi.\frac{0.8}{360}.R.\cos a \cong 2.\Pi.\frac{0.8}{360}.R.\cos(a + 0.8) = 2.\Pi.\frac{0.8}{360}.X$$

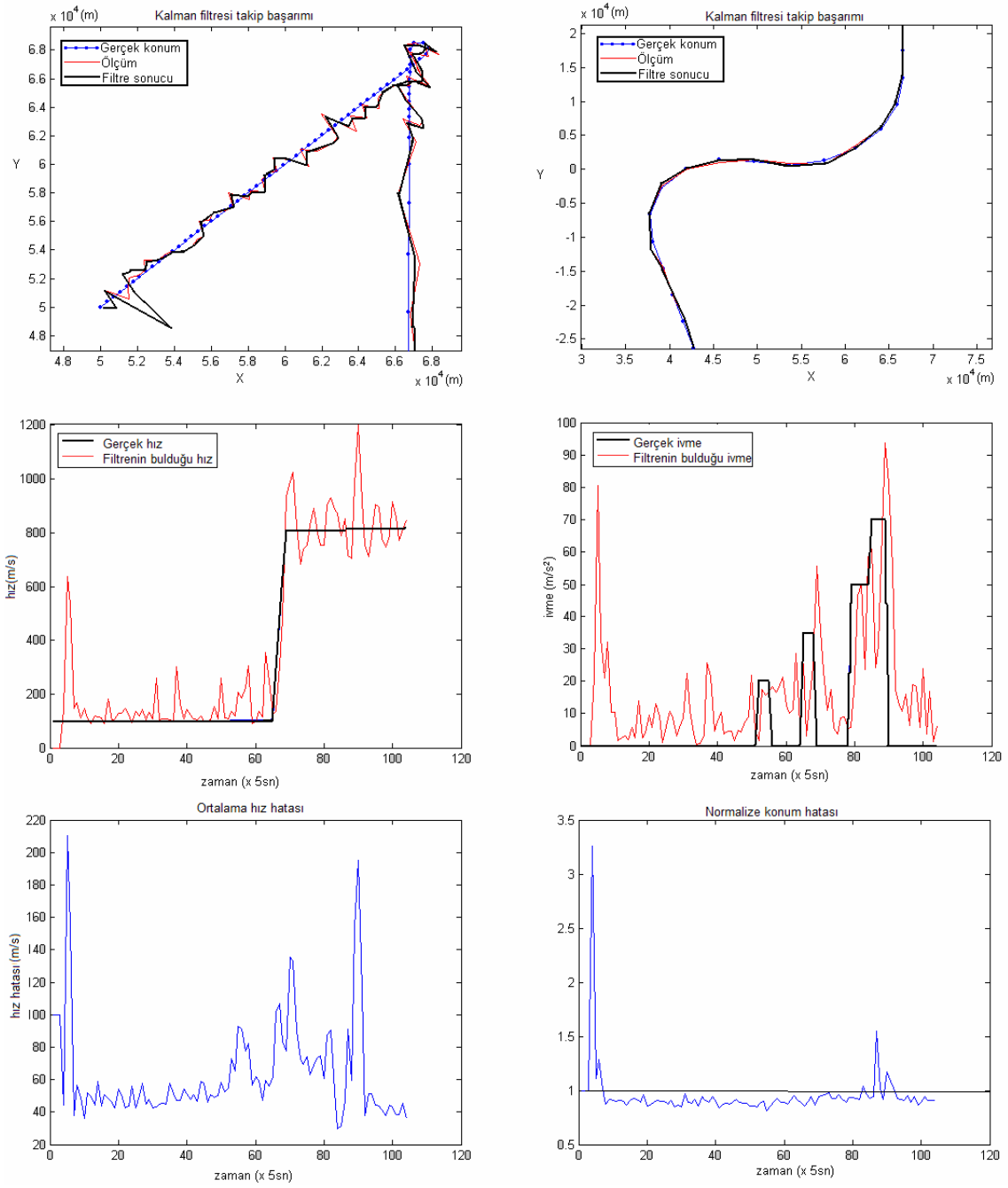
$$dX = 2.\Pi.\frac{0.8}{360}.Y \rightarrow \sigma_x = dX/2 \rightarrow \sigma_x^2 = (\sigma_x)^2$$

$$dY = 2.\Pi.\frac{0.8}{360}.X \rightarrow \sigma_y = dY/2 \rightarrow \sigma_y^2 = (\sigma_y)^2$$

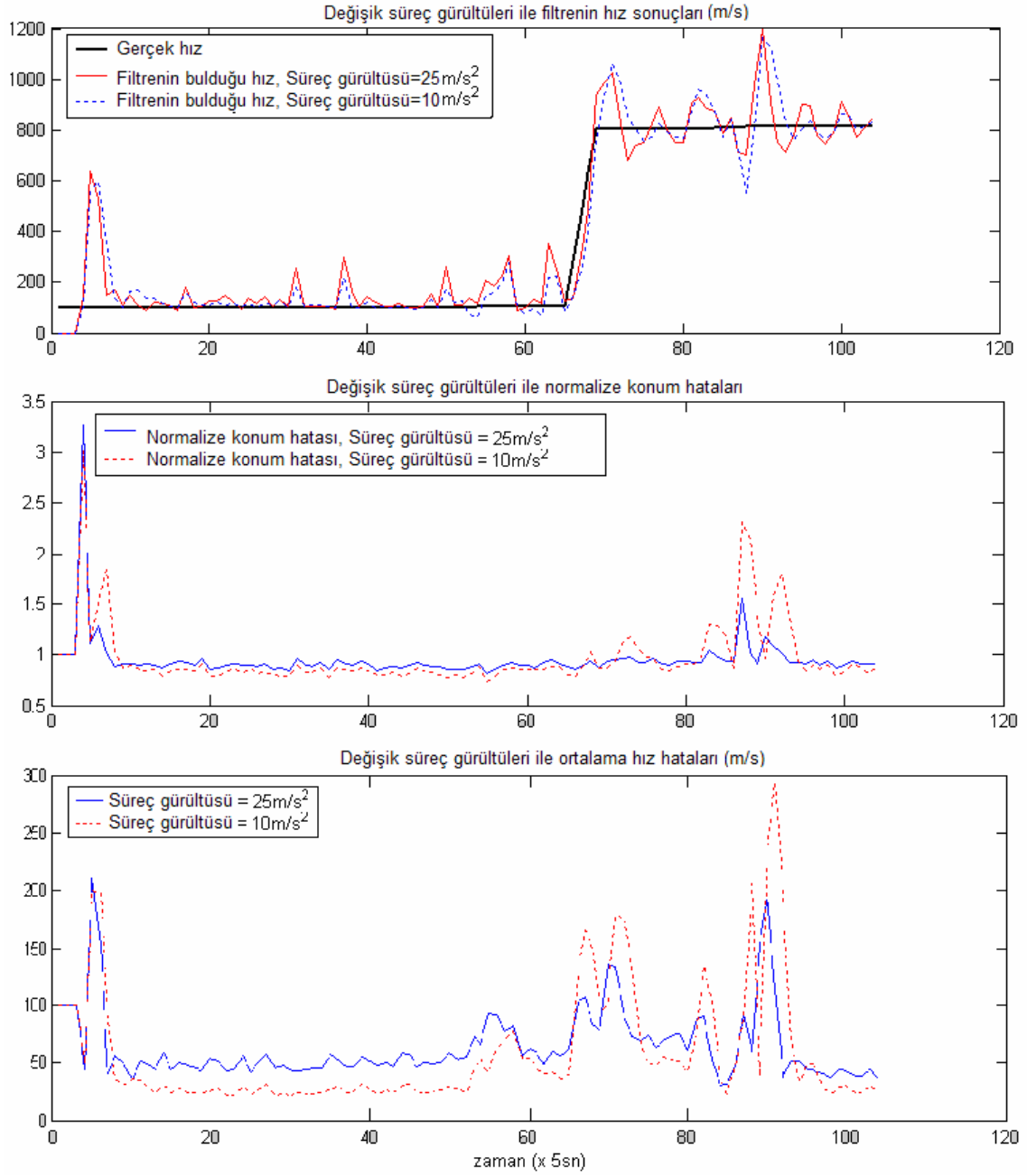
$$\text{Ölçüm Matrisi: } H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Kovaryans matris, durum değişim matrisi ve ölçüm gürültüsü değerleri bölüm 2.3.1'deki gibi alındığında Kalman filtresinin genel senaryo'daki başarımları Şekil 2.24'deki gibi oluşmaktadır. Yine diğer parametreler sabit kalmak koşuluyla süreç gürültü kovaryansı artırıldığında oluşan hız değerleri, ortalama hız hataları ve normalize konum hataları Şekil 2.25'te karşılaştırmalı olarak verilmektedir.

Beklendiği gibi süreç gürültüsünü artırmak manevralı anlarda başarımları artırmış ancak manevra yapılmayan anlarda başarımları azaltmıştır.



Şekil 2.24 Kalman filtresinin genel senaryoda takip başarımları



Şekil 2.25 Kalman filtresinin değişik süreç gürültüleri ile genel senaryoda takip başarımlarını karşılaştırmaları

3. ÇOK FİLTRELİ MODEL

Çok filtreli model birden çok filtreyi aynı anda çalıştıran ve bütün filtrelerin sonucunu kullanarak ortak bir sonuç üreten modeldir. Çok filtreli modellerde en iyi bilinen en yeni yaklaşım IMM filtresidir. Çok filtreli model olarak bu çalışmada IMM filtresi ve IMM filtresinin değiştirilmiş yapıları incelenecektir.

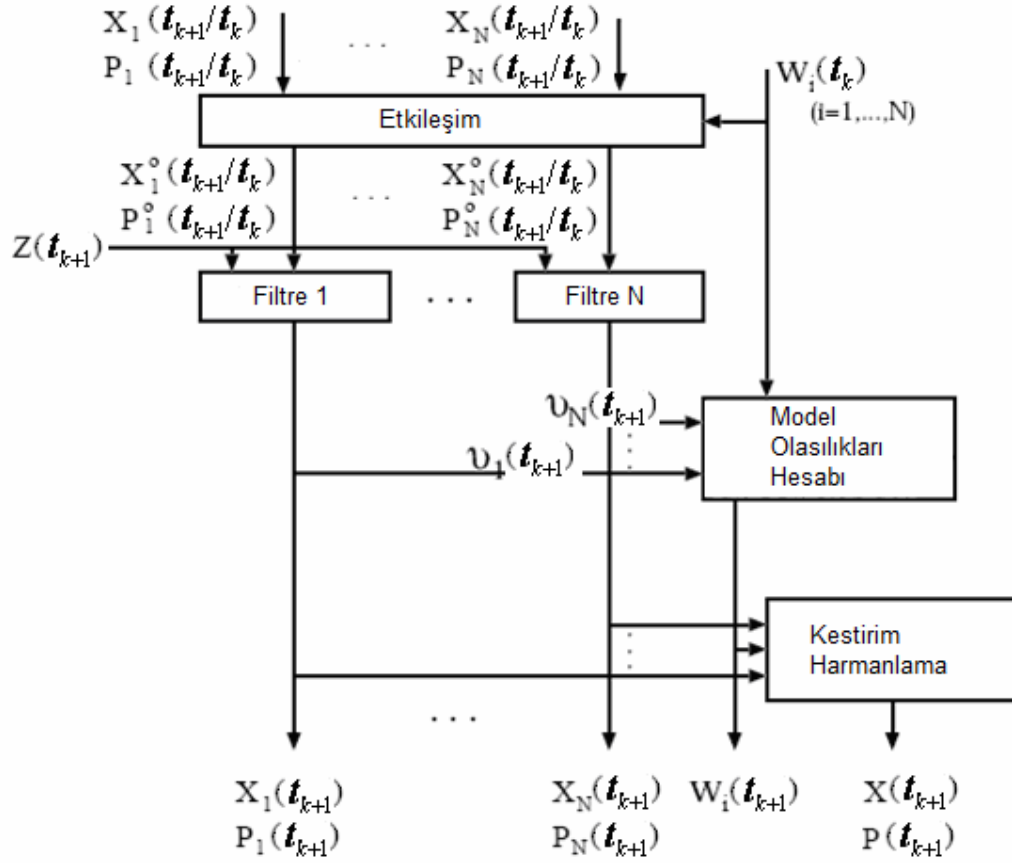
3.1 IMM (Etkileşimli Çoklu Model) Filtresi

Etkileşimli çoklu model, birden fazla filtreyi aynı anda kullanan, olasılıksal bir mantıkla filtreler arasında geçiş yapan, ve sonuç olarak bütün filtrelerin kendi sonuçlarını filtrelerin ağırlıklı ortalamasıyla çarpıp genel bir filtre sonucu üreten bir modeldir. Bar-Shalom, Y., ve Li, X. R.,(1993) ve Averbuch, A. vd. (1991) IMM'in hedef takip sistemleri arasında en iyi metodlardan biri olduğunu belirtmişlerdir.

Bu algoritma, manevra yapan hedefi takip etmek için belli sayıda filtre kullanır ve her filtrenin ürettiği durum vektörlerini harmanlayıp ortak bir sonuç üretir. Her döngünün başlangıcında en sonki başarılı filtrelerin tahminleri ağırlıklı olarak kullanılır. Her modelin ağırlığı o modelin yaptığı tahmin başarısıyla doğru orantılıdır ve bu değer benzerlik (likelihood) fonksiyonu ile bulunur.

Şekil 3.1, IMM algoritmasının blok diyagramını göstermektedir.

Burada şekildeki $\mathbf{X}_i(t_k)$, $\mathbf{P}_i(t_k)$ ve $\mathbf{W}_i$ sırayla i.nci filtrenin durum kestirimi, durum kestirim kovaryansı ve olasılığını göstermektedir. Algoritmanın işleyişi dört ana başlıkta verilebilir: tahminlerin etkileşimi, filtreleme, filtre olasılığı hesaplama ve kestirim harmanlama.



Şekil 3.1 IMM Algoritması

1) Tahminlerin Etkileşimi

j 'inci filtre için tahminlerin etkileşimi (3.1)'deki bağıntıda verilmiştir:

$$X_j^o(t_{k+1}/t_k) = \sum_{i=1}^N \frac{P_{ij} W_i(t_k)}{\sum_{m=1}^N P_{mj} W_m(t_k)} X_i(t_{k+1}/t_k) \quad (3.1)$$

Burada

$X_i(t_{k+1}/t_k)$: i 'inci filtrenin k anında $k+1$ anı için yapmış olduğu durum tahmini,

$W_i(t_k)$: i 'inci filtrenin olasılığı,

P_{ij} : i 'den j 'ye geçiş olasılığı,

N : filtre sayısıdır,

Etkileşim sonucu j' inci filtrenin süreç hata kovaryansı $P_j^o(t_{k+1}/t_k)$:

$$P_j^o(t_{k+1}/t_k) = \sum_{i=1}^N \left\{ \frac{P_{ij}W_i(t_k)}{\sum_{m=1}^N P_{mj}W_m(t_k)} \left[P_i(t_{k+1}/t_k) + (X_i(t_{k+1}/t_k) - X_i^o(t_{k+1}/t_k))(X_i(t_{k+1}/t_k) - X_i^o(t_{k+1}/t_k))^T \right] \right\} \quad (3.2)$$

Bar-Shalom, Y., ve Li, X, R.,(1993) sistemde M tane filtre olduğu durumda, filtre geçiş olasılıklarını aşağıdaki gibi kullanmışlardır.

$$p_{ij} = \begin{cases} A & i = j \\ \frac{1-A}{M-1} & i \neq j \end{cases} \quad (3.3)$$

Eğer A büyük seçilirse filtre geçiş olasılığı düşük kalır. Bu da filtre modelinde değişimlere çok izin verilmediği sonucunu çıkarır. A küçük seçilirse filtre geçiş olasılıkları büyük olacaktır. Bu durumda değişen dinamiklere karşı filtreler arası daha kolay geçiş yapılabilecektir. Bar-Shalom, Y., ve Li, X, R.,(1993) A'nın 0.8 ile 0.98 arası seçilmesinin uygun olacağını belirtmişlerdir.

Örneğin A = 0.8, filtre sayısı M = 3 alınırsa, filtre geçiş olasılıkları:

$P_{11} = 0.8$, $P_{12} = 0.1$, and $P_{13} = 0.1$ olacaktır.

Model olasılıkları için de başlangıç için aşağıdaki gibi bir yöntem kullanılmaktadır.

$$w_i = \begin{cases} 0.6 & i = j \\ \frac{0.4}{M-1} & i \neq j \end{cases} \quad (3.4)$$

2) Filtreleme

Filtreleme için referans olarak Kalman filtreleri seçilmiştir. Bunun yanında farklı filtreler de kullanılabilir.

Durum tahmini:

$$x(t_{k+1}/t_k) = Fx(t_k/t_k) \quad (3.5)$$

Yenilenme:

$$v(t_{k+1}) = z(t_{k+1}) - Hx(t_{k+1}/t_k) \quad (3.6)$$

Durum hata kovaryansı:

$$P(t_{k+1}/t_k) = FP(t_k/t_k)F^T + GQG^T \quad (3.7)$$

Yenilenme kovaryansı:

$$S(t_{k+1}) = HP(t_{k+1}/t_k)H^T + R \quad (3.8)$$

Filtre kazancı:

$$K(t_{k+1}) = P(t_{k+1}/t_k)H^T S(t_{k+1})^{-1} \quad (3.9)$$

Kovaryans tahmini:

$$P(t_{k+1}/t_{k+1}) = [1 - K(t_{k+1})H]P(t_{k+1}/t_k) \quad (3.10)$$

Durum kestirimi:

$$x(t_{k+1}/t_{k+1}) = x(t_{k+1}/t_k) + K(t_{k+1})v(t_{k+1}) \quad (3.11)$$

3) Model Olasılık Hesabı

j ' inci filtrenin benzetim (likelihood) fonksiyonu, yenilenme miktarı $v(t_{k+1})$ ve ilgili yenilenme kovaryansı $S(t_{k+1})$ ile bulunur:

$$\Lambda_j(t_k) = P[Z(t_k) | M_j(t_k), Z^{k-1}] \quad (3.12)$$

$$\Lambda_j(t_k) = \frac{1}{\sqrt{(2\pi)^n |S_j(t_k)|}} e^{\left(-\frac{1}{2}v_j^T(t_k)S_j^{-1}v_j(t_k)\right)} \quad (3.13)$$

Filtre olasılıkları benzetim fonksiyonlarından bulunur:

$$w_j(t_k) = \frac{\Lambda_j(t_k) \sum_{l=1}^n p_{lj} w_l(t_{k-1})}{\sum_{i=1}^n \sum_{l=1}^n \Lambda_i(t_k) p_{li} w_l(t_{k-1})} \quad (3.14)$$

4) Kestirim Harmanlama

Her filtrenin kestirimi filtrelerin olasılıkları ile ağırlandırılarak toplu kestirim yapılır. Harmanlanmış kestirim IMM filtresinin sonucu olur:

$$X(t_k / t_k) = \sum_{j=1}^n w_j(t_k) X_j(t_k / t_k) \quad (3.15)$$

$$P(t_k / t_k) = \sum_{j=1}^n w_j(t_k) \left[P_j(t_k) + (X_j(t_k / t_k) - X(t_k / t_k))(X_j(t_k / t_k) - X(t_k / t_k))^T \right] \quad (3.16)$$

IMM filtresi diğer tekli filtrelere nazaran manevra anlarında daha yumuşak geçiş yapar. Bu geçişler sonuçlarda devamlılığı getirir. Filtre sayısının çokluğu da muhtemel manevralara daha çok hazırlıklı olmayı sağlar. Ancak filtre sayısı oranında da hesap yükü getirmektedir. Uygulamaların genelinde üç filtre yeterli bulunmaktadır. Biri manevra yapmayan hedefler için modellenirken diğerleri farklı ivmeli manevraları kataracak şekilde modellenir.

3.1.1 IMM Filtresinin Genel Senaryo Test Sonuçları

Etkileşimli çoklu model aşağıda değişik parametreler ile test edilmiştir:

Test 1:

Bu testte literatürde genel senaryo genel olarak kabul edilen ve kullanılan parametreler ile test edilmektedir:

Filtre sayısı: $M = 3$,

$$\text{Geçiş olasılığı: } P_{ij} = \begin{cases} 0.8, & i = j \\ 0.1, & i \neq j \end{cases} P_{ij} = \begin{cases} 0.8 & i = j \\ 0.1 & i \neq j \end{cases}$$

$$\text{Başlangıç model olasılıkları: } W_i = \begin{cases} 0.6, & i = 1 \\ 0.2, & i > 1 \end{cases} W_{ij} = \begin{cases} 0.6 & i = 1 \\ 0.2 & i > 1 \end{cases}$$

Üç filtre aşağıdaki gibi seçilmiştir:

Filtre 1: İki boyutlu filtredir. Hedef hareketi konum ve hız değerleriyle modellenir. Böylece konum hatası hız tahminindeki hatalardan kaynaklanacaktır. En fazla 10 m/s hız hatasının yapıldığı ve bu hatanın olasılığının homojen dağıldığı kabul edilir:

$$\text{Kovaryans matrisi } Q = \begin{bmatrix} C_{XX} & C_{XY} & C_{XV_x} & C_{XV_y} & C_{XA_x} & C_{XA_y} \\ C_{YX} & C_{YY} & C_{YV_x} & C_{YV_y} & C_{YA_x} & C_{YA_y} \\ C_{V_xX} & C_{V_xY} & C_{V_xV_x} & C_{V_xV_y} & C_{V_xA_x} & C_{V_xA_y} \\ C_{V_yX} & C_{V_yY} & C_{V_yV_x} & C_{V_yV_y} & C_{V_yA_x} & C_{V_yA_y} \\ C_{A_xX} & C_{A_xY} & C_{A_xV_x} & C_{A_xV_y} & C_{A_xA_x} & C_{A_xA_y} \\ C_{A_yX} & C_{A_yY} & C_{A_yV_x} & C_{A_yV_y} & C_{A_yA_x} & C_{A_yA_y} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 33.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 33.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Bu filtre manevra yapmayan hedefler için tasarlanmıştır.

Filtre 2: İkinci filtre üç boyutludur. Hedef hareketi hız, konum ve ivme ile modellenir böylece konum ve hız hataları ivmedeki hatadan kaynaklandığı varsayılır. En fazla $\pm 10\text{m/s}^2$ homojen dağılmış ivme hatasını kotarabilecek şekilde tasarlanmıştır:

$$\text{Kovaryans matrisi: } Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 33.3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 33.3 \end{bmatrix}$$

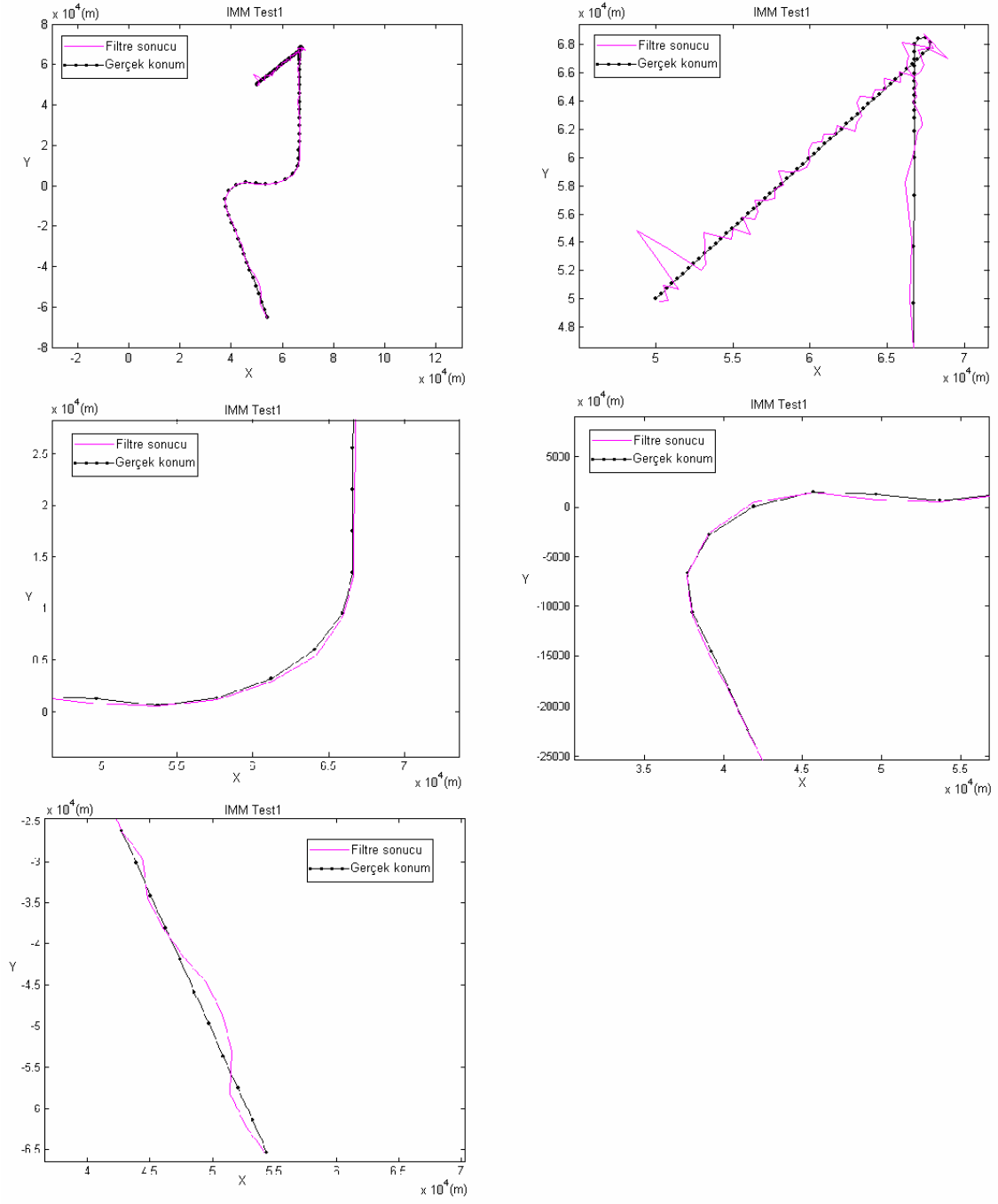
Bu filtre düşük ivmeli manevra yapan hedefler için tasarlanmıştır.

Filtre 3: Filtre 2'deki gibi üç boyutlu filtredir. En fazla $\pm 30\text{m/s}^2$ homojen dağılımlı ivme hatasını kotarabilecek şekilde tasarlanmıştır:

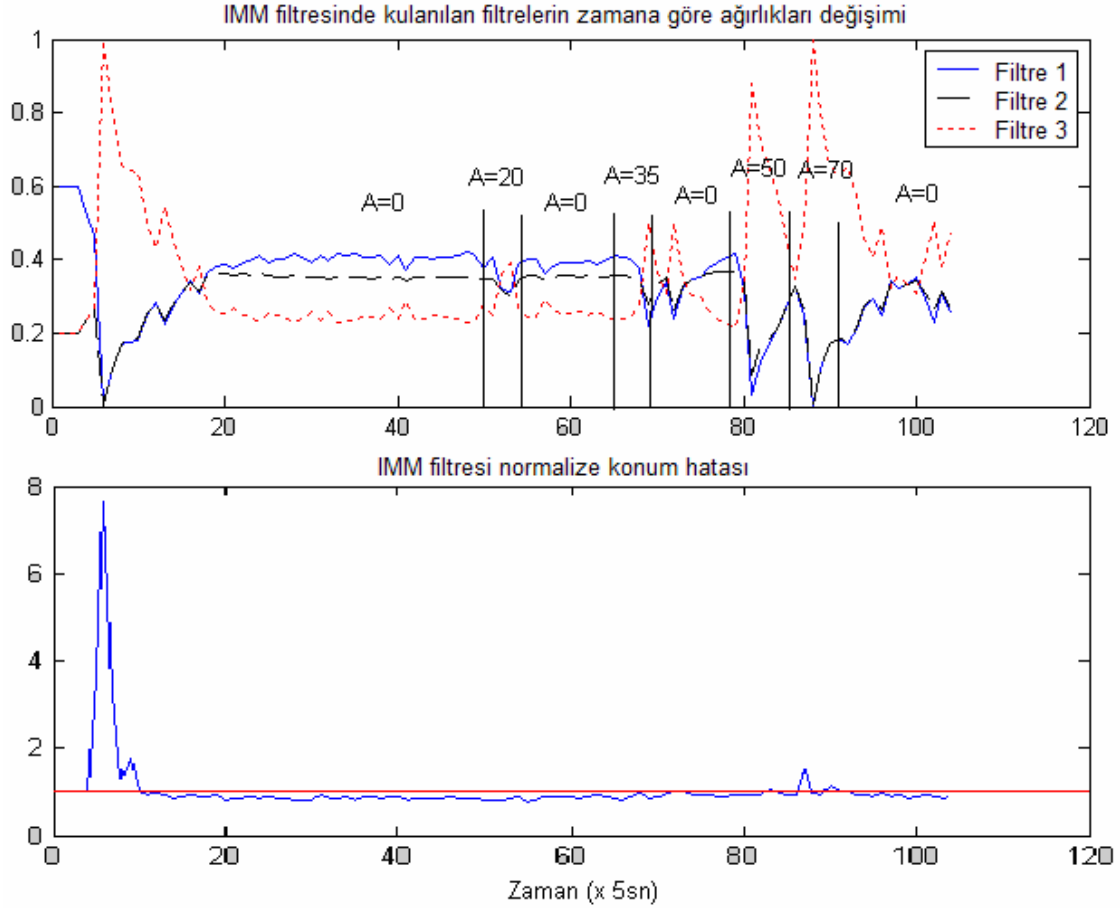
$$\text{Kovaryans matrisi: } Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 300 & 0 \\ 0 & 0 & 0 & 0 & 0 & 300 \end{bmatrix}$$

Bu filtre yüksek manevralı hedefleri takip etmek için tasarlanmıştır.

Seçilen filtreler ile IMM filtresi test edildiğinde başarımları Şekil 3.2 ve Şekil 3.3'te görülmektedir. Şekil 3.3'te görüldüğü gibi IMM filtresi, hedef manevra yapmadığı anlarda birinci filtreye (1–49 ölçüm arası), yüksek manevra yaptığı anlarda (80–104 arası) ise üçüncü filtreye daha çok ağırlık vermektedir.



Şekil 3.2 IMM filtresinin genel senaryo üzerinde görsel test sonucu



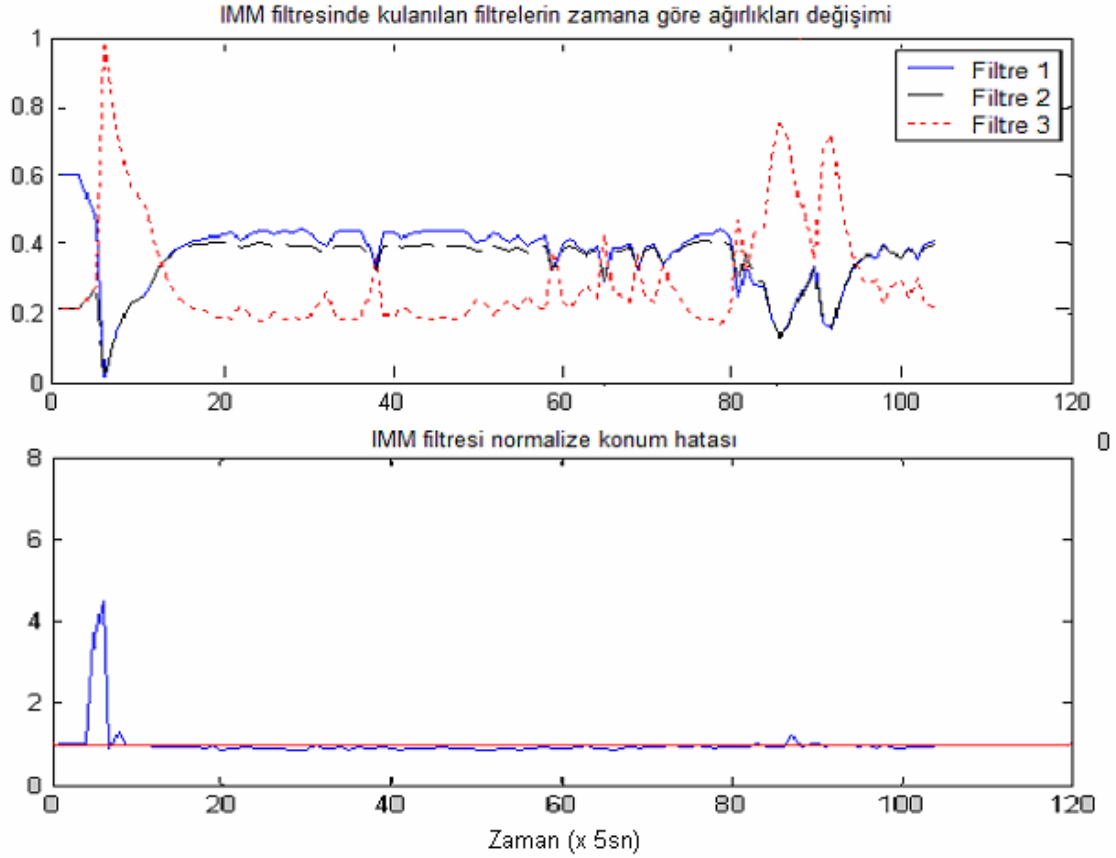
Şekil 3.3 Test 1: IMM filtresinde alt filtre ağırlıkları değişimi ve normalize konum hatası başarımı

Test 2:

Bu testte bir filtrenin gereğinden çok büyük kovaryans matrisine sahip olması durumunda IMM filtresinin davranışı incelenmektedir.

Diğer bütün parametreleri test 1'deki gibi seçilmiş, üçüncü filtrenin kovaryans matrisi ise çok büyütülmüştür örneğin en çok $\pm 70\text{m/s}^2$ homojen dağılımlı ivme hatasını kataracak şekilde seçilmiştir.

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1633 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1633 \end{bmatrix}$$



Şekil 3.4 Test 2: IMM filtresinde alt filtre ağırlıkları değişimi ve normalize konum hatası başarımı

Şekil 3.4'te IMM'in hedef kinematiklerine uygun davrandığı yani düzgün doğrusal anda Filtre 1'i manevra anında filtre 3'ü kullandığı görülmektedir. Bu durum Test 1'deki davranışla aynıdır. Filtre 3'ün kovaryans matrisi sadece filtre 3 kullanıldığı anda devreye girmiştir. Şekil 3.4'te görüldüğü gibi normalize konum hatası 87. ölçümde Test 1'dekinden daha azdır.

Bununla birlikte senaryonun bütününde filtre 3 toplam kestirim sonucuna katkıda bulunmaktadır. Manevra yapılmadığı zamanlarda filtre 3'ün büyük kovaryans matrisi ile ürettiği sonuç toplam kestirime eklendiği için toplam sonuç kötüleşmiştir. Normalize konum hatası bu testte test 1'e göre daha az başarılıdır.

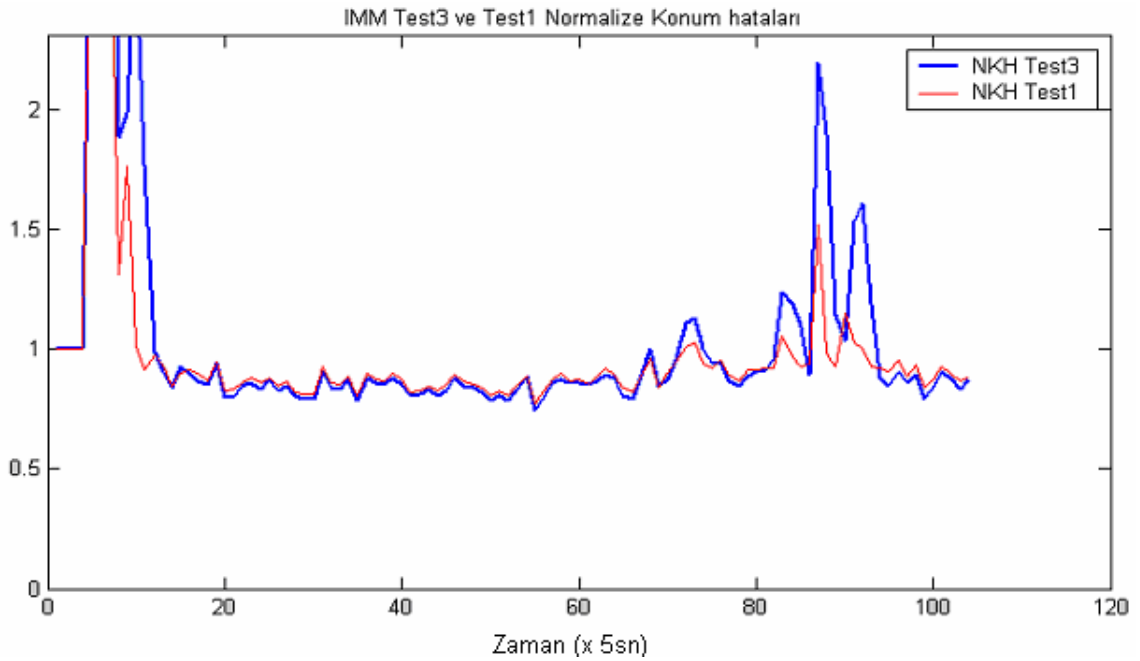
Test 3:

Test 3'te filtre geiş matrisini daha iyi seçmenin filtreye etkisi incelenmektedir. Genel hava araçlarının davranışını incelendiğinde genelde düzgün doğrusal hareket ettikleri ve az sıklıkla manevra yaptıkları görülür. Böylece filtre 1'den diğer filtreler geiş olasılığı düşük seçilmelidir. Benzer şekilde yüksek manevranın çok devam ettirilmeyeceği ve en kısa zamanda düzgün doğrusal harekete geçileceği düşünülebilir. Yüksek manevrada hareket etmek pilotlarca daha az dayanılabilir bir durumdur ve bu hareket anlarında düzgün doğrusal harekete geme olasılığı manevrayı azaltma olasılığından daha yüksektir.

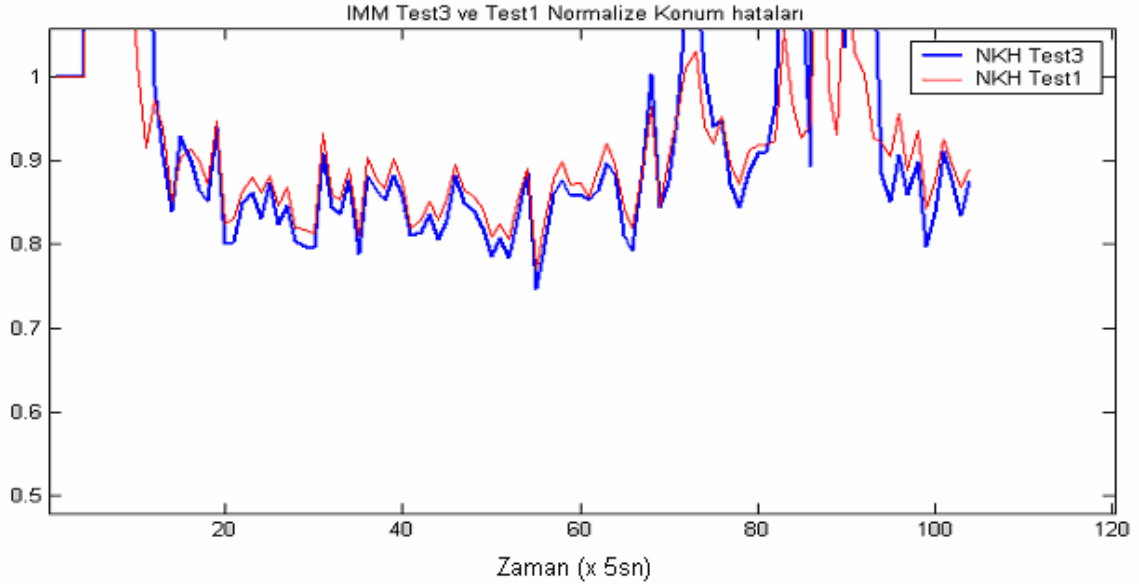
Geiş matrisini aşağıdaki gibi seçmek daha gerçekçi olacaktır:

$$P_{ij} = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.2 & 0.1 \\ 0.5 & 0.3 & 0.2 \\ 0.7 & 0.2 & 0.1 \end{bmatrix}$$

Diğer tüm parametreler test 1'deki ile aynı kabul edilmektedir. Şekil 3.5 ve Şekil 3.6'da bu testte önerilen geiş parametreleri ile hesaplanmış NKH ile Test 1'deki standart önerilen geiş parametreleri ile hesaplanmış NKH değerleri karşılaştırılmıştır:



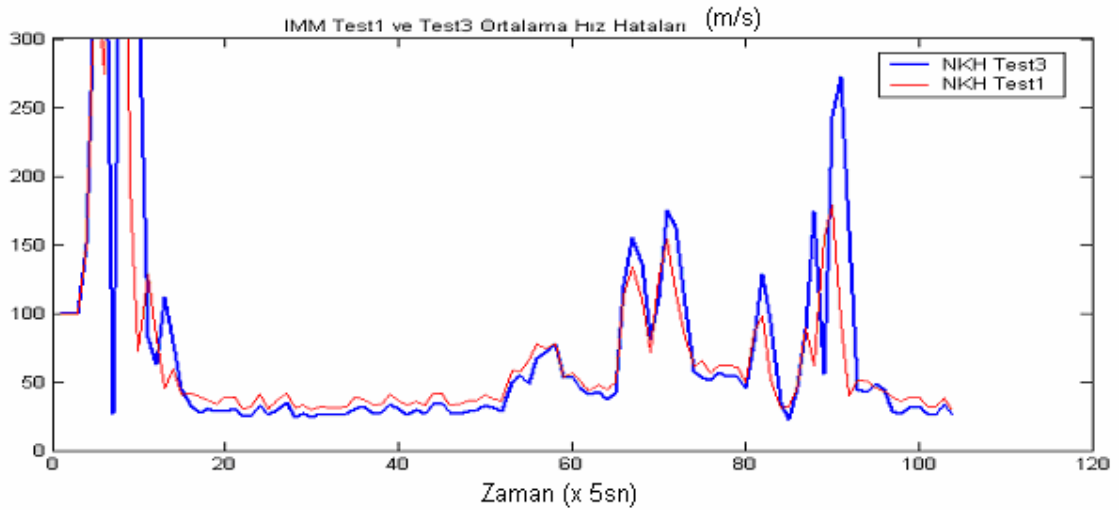
Şekil 3.5 Test 1 ile Test 3'ün normalize konum hataları karşılaştırması



Şekil 3.6 Test 1 ile Test 3'ün normalize konum hataları karşılaştırması-2

Yeni geçiş parametreleri IMM filtresinde filtre 1'i (düzgün doğrusal hareket modeli) daha çok kullanmaya zorlamış, bunun sonucu olarak filtre 1'in sonuç üzerindeki etkisi artmıştır. Şekil 3.5'de cismin düzgün doğrusal hareket ettiği ilk 55 örnekte yeni parametrelerin kazandırdığı iyileşme görülmektedir. Şekil 3.6'da ise manevra anında (67-95 arası) eski parametrelerin daha iyi sonuç verdiği görülebilir.

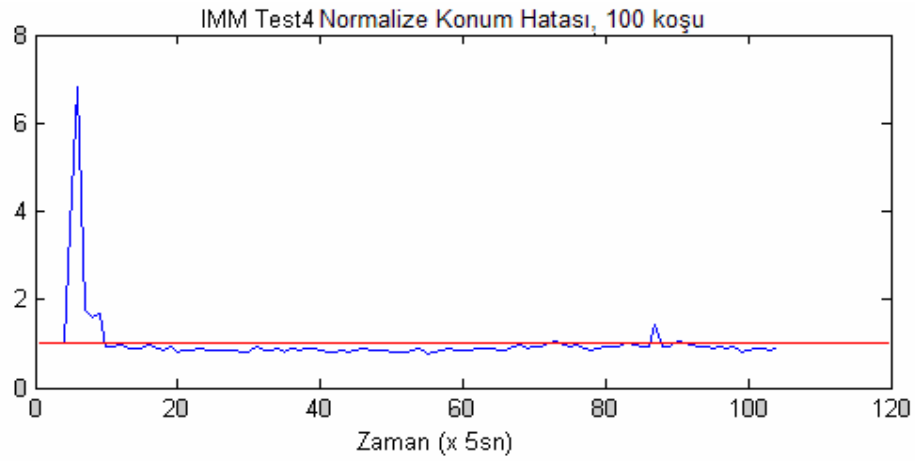
Ortalama hız hatasında da benzer sonuçlar Şekil 3.7'de görülmektedir.



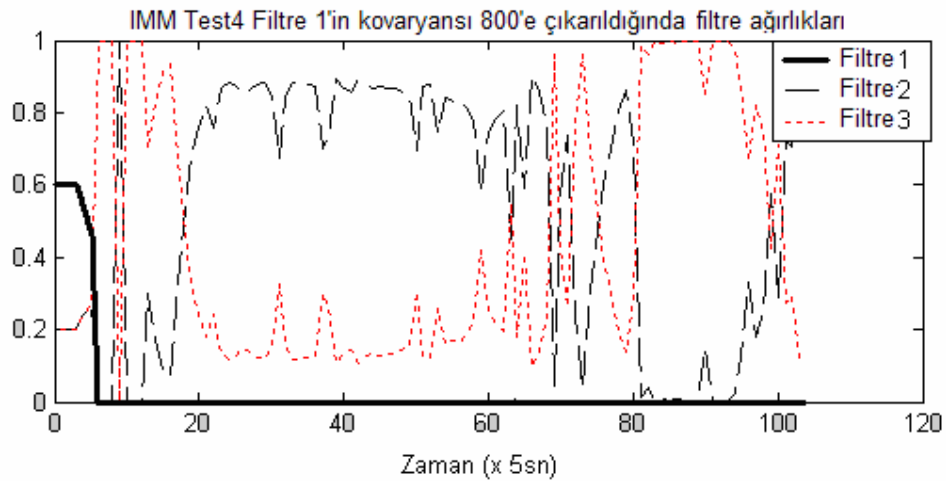
Şekil 3.7 Test 1 ile Test 3'ün ortalama hız hataları karşılaştırması

Test4 (NIMM: Etkileşimsiz Çoklu Model Yaklaşımı):

Bu testte tüm parametreler test 1’deki gibi tutulup etkileşimli çoklu modelin etkileşim kısmı çıkarılmaktadır. Böylece etkileşimsiz çoklu model tanımı yapılmaktadır. Bu filtreye NIMM veya IMM-2 denilebilir. Etkileşim kısmı çıkarıldığında bütün filtreler işleyiş olarak birbirlerinden tamamen bağımsız çalışmakta böylece filtre sonuçları filtrelerin ağırlıkları oranında harmanlanarak NIMM filtresi sonucunu üretmektedir. Şekil 3.8’de NIMM filtresinin normalize konum hatası verilmektedir. Şekil 3.9’da da filtrelerin olasılıklarının değişimi görülmektedir.



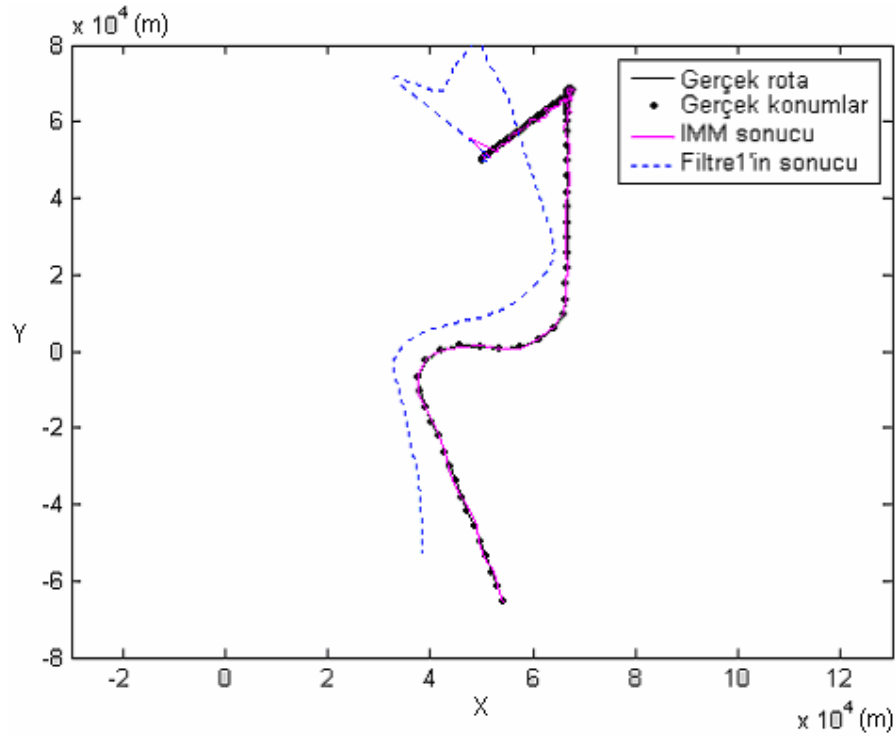
Şekil 3.8 IMM Test4 NKH sonucu



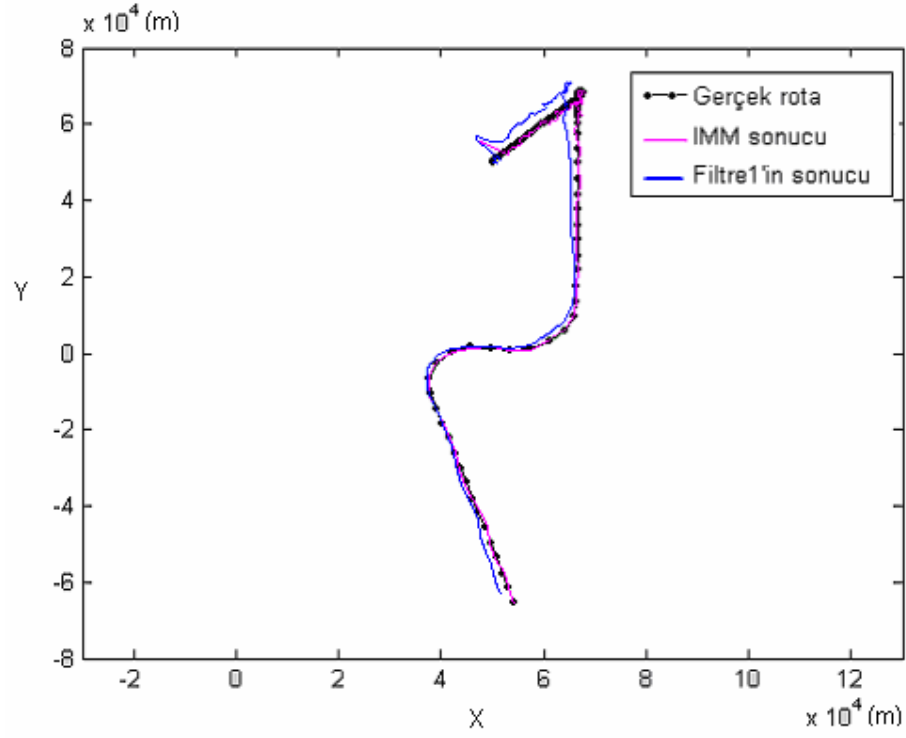
Şekil 3.9 IMM Test4 filtre olasılıkları

Şekil 3.9’da filtre 1’in bağımsız olarak çalıştığına ağırlığı sıfıra indiği görülmektedir. Bunun

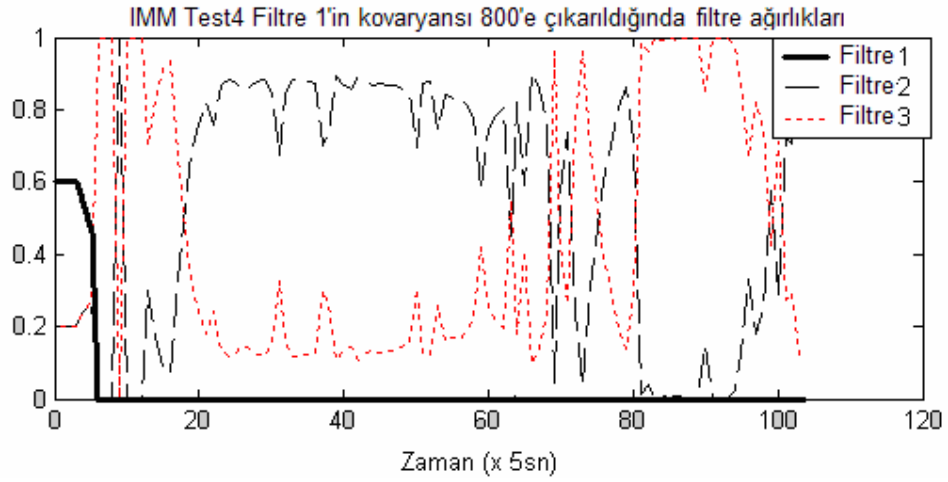
sebebini anlamak için filtre 1'in bağımsız sonuçlarını incelemek gerekir. Şekil 3.10'da filtre 1'in tek başına bağımsız olarak ürettiği sonuçların gerçek değerden ne kadar uzakta olduğu görülmektedir. Bu durumu düzeltmek için kovaryans matrisi büyütme çözüm olabilir. Şekil 3.11 ve Şekil 3.12'de filtre 1'in kovaryans matrisinin 800'e çıkarıldığı durumdaki sonuç görülmektedir. Nispeten iyileşme olsa bile filtre 1 gerçek konum değerlerinden uzakta sonuçlar üretmektedir. Bu durumun sebebi olarak filtre 1'in iki boyutlu olması, manevra karşısında kendini düzeltme yeteneğinin olmaması gösterilebilir. Filtre dengeye oturana kadar geçen sürede oluşan salınımlardan dolayı başlangıçtaki hatalı karar filtre 1'i ilk 50 örnekte bile başarısız kılmıştır. Filtre 1'in genel senaryoda cismin düzgün doğrusal hareket ettiği ilk 50 örnekte beklenen faydası görülmemektedir. Bu seçimin yanlış olduğu görülmektedir.



Şekil 3.10 IMM Test4'te filtre 1'in bağımsız ürettiği sonuç



Şekil 3.11 IMM Test4'te kovaryans matrisi artırıldığında filtre 1'in bağımsız ürettiği sonuç

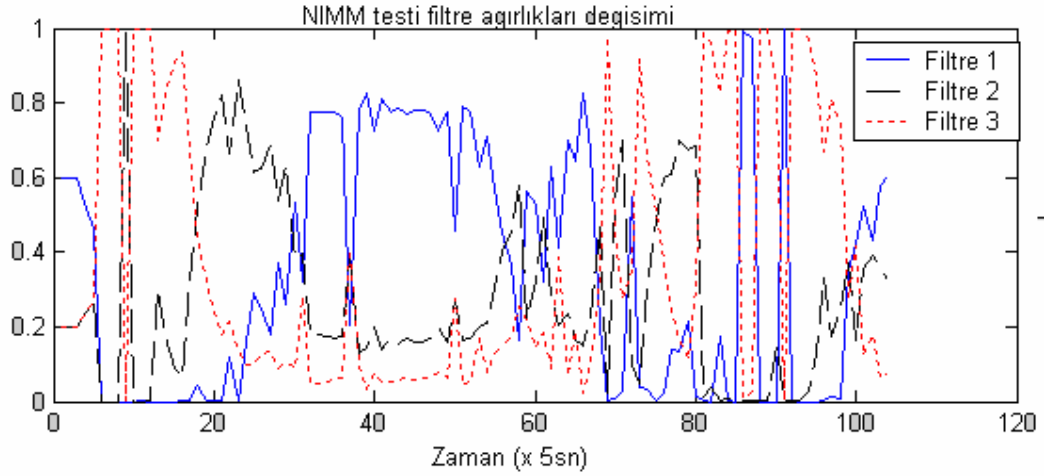


Şekil 3.12 IMM Test4'te filtre 1'in kovaryans matrisi artırıldığında filtre olasılıkları

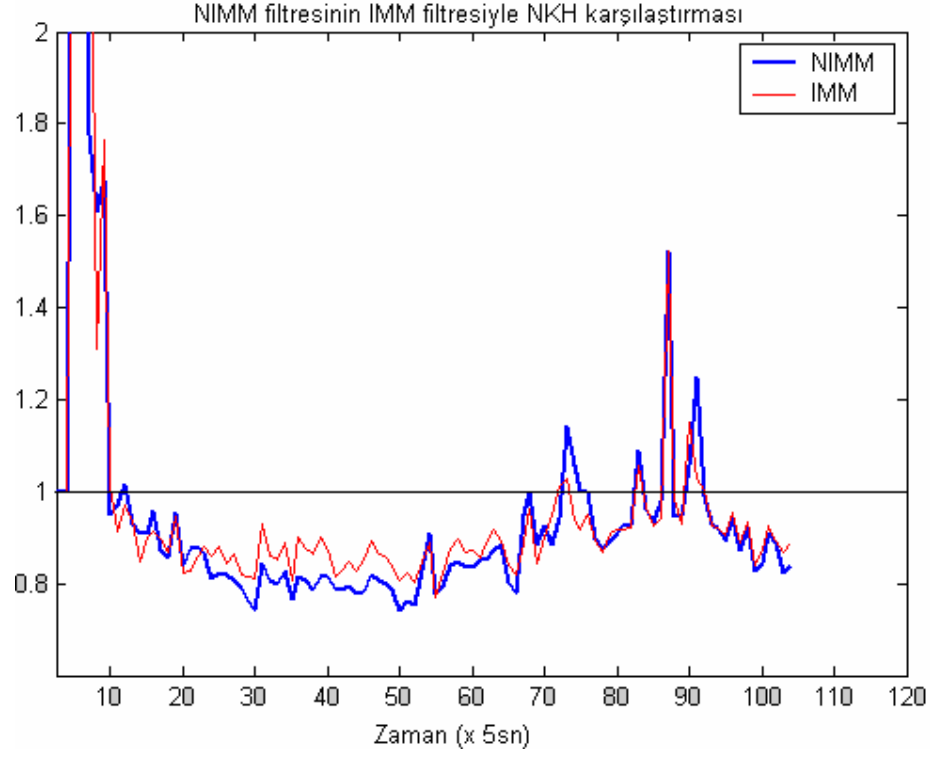
NIMM filtresinde filtre 1 için yine 3 boyutlu ancak küçük kovaryanslı bir filtre seçilmiştir. Diğer parametreler test 1'deki gibi kullanılmaktadır. Filtre 1 için aşağıdaki kovaryans matrisi seçilmiştir:

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

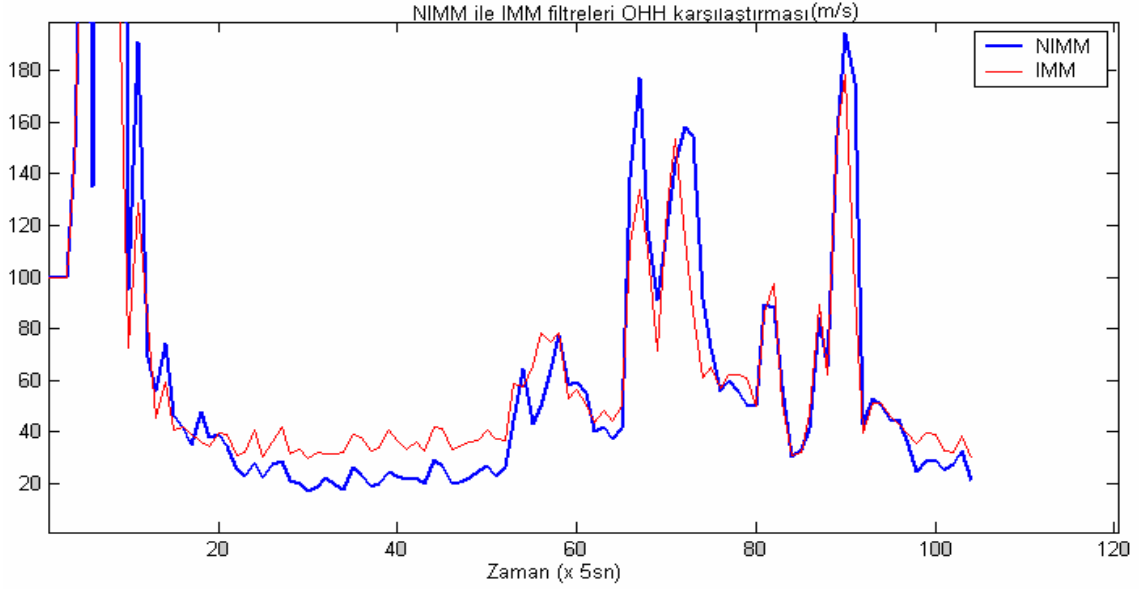
Filtre ağırlıkları değişimi Şekil 3.13'te görülmektedir. NIMM filtresinin IMM filtresine göre hem NKH hem de OHH grafiklerinde (Şekil 3.14 ve Şekil 3.15) daha başarılı sonuçlar ürettiği görülmektedir. NIMM filtresi bu başarısını alt filtrelerini birbirlerinden bağımsız olarak çalıştırmasından elde etmektedir. Bir filtrenin başarısızlığı diğer filtreyi etkilememektedir. Başarı oranına göre manevra anlarında filtrelerin ağırlığı değişmekte böylece NIMM sonucu olarak her zaman başarılı filtrelerin sonucu verilmektedir.



Şekil 3.13 NIMM filtresinde alt filtrelerin manevraya bağımlı olarak ağırlıklarının değişimi

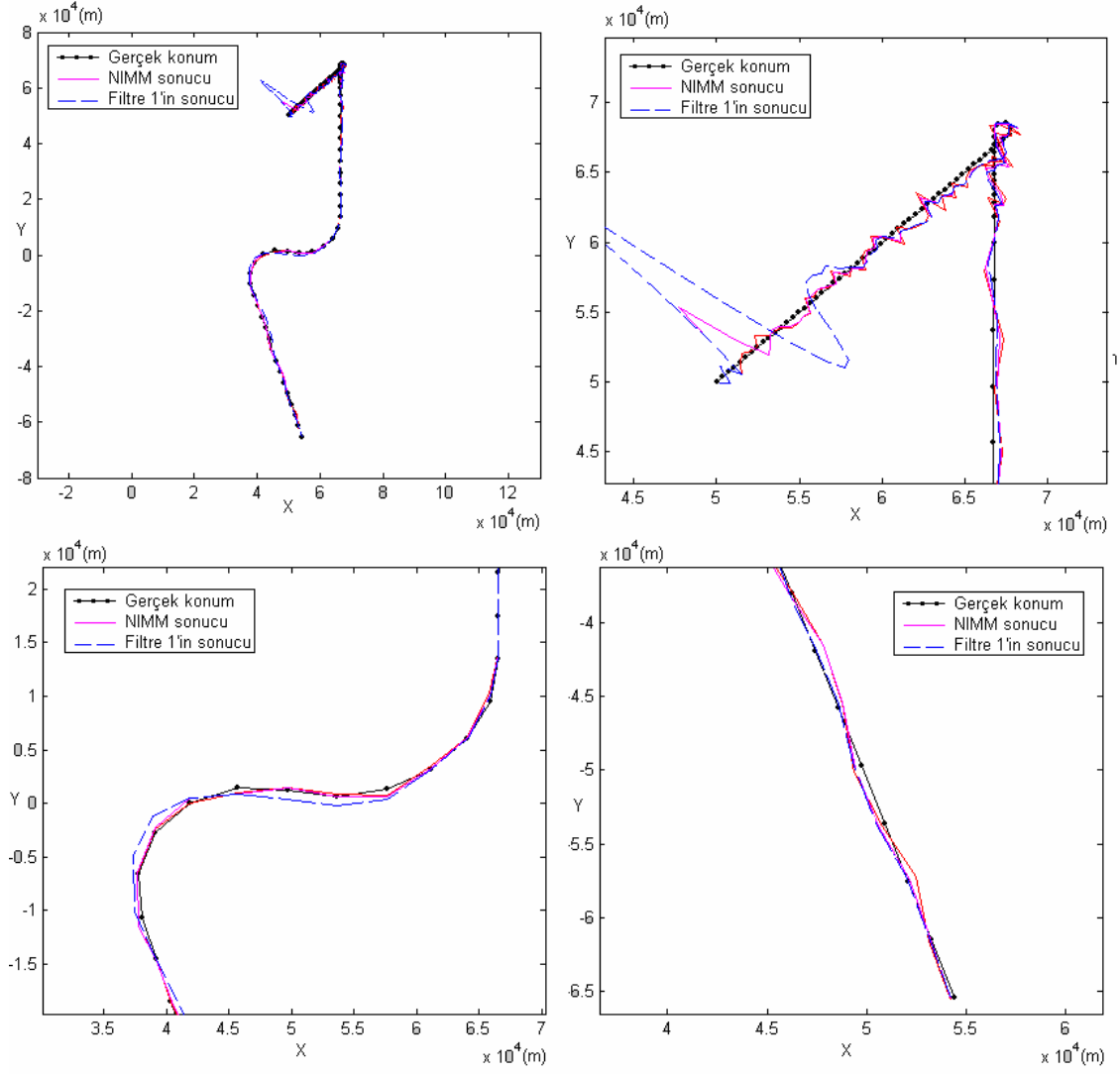


Şekil 3.14 NIMM filtresi ile IMM filtresinin NKH karşılaştırması



Şekil 3.15 NIMM filtresi ile IMM filtresinin OHH karşılaştırması

Filtre 1'i üç boyutlu seçmenin sonuçları Şekil 3.16'da görülmektedir. Filtre 1 küçük kovaryansı sayesinde düzgün doğrusal hareketlerde başarılı olmuş, üçüncü boyutu sayesinde de başlangıç hatalarını kapatabilmiştir.



Şekil 3.16 Filtre 1'in NIMM filtresinde bağımsız katkısı

4. YENİ BİR YAKLAŞIM: ŞABLON FİLTRESİ

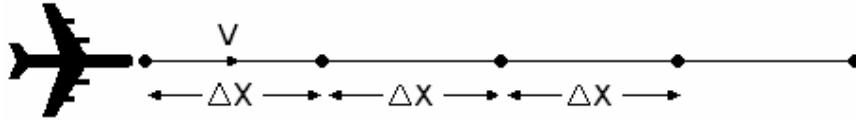
Literatürde genel takip amaçlı birçok filtre tasarlanmıştır. Bunlardan en bilinenleri olarak Kalman ve sonrasında IMM filtreleri bölüm 2 ve bölüm 3'te incelenmiştir. Bu filtrelerin özelliği, belirlenen sistem dinamikleri ve kovaryans matrisler ölçüsünde geleceğe yönelik en iyi MMSE tahmini yapabilmeleridir. Ancak kovaryans matrisinin nasıl seçilmesi gerektiği konusu önemli bir zorluk getirmektedir. Bunun yanında bu filtrelere cismin ne gibi değişik manevralar yapabileceği öğretilemediği için bu filtreler manevraları önceden bilemezler. Sadece kovaryans matrisi ölçüsünde yapılan manevrayı yakalayıp yakalayamayacağı öngörülebilir.

Bu çalışmada özellikle savaş uçaklarını takip etmeye yönelik en zorlayıcı manevralar üzerinde filtreler test edildi. O halde savaş uçaklarının yapabileceği manevraları önceden bilen bir filtre, yapılan manevralara hazırlıklı olur ve de en iyi şekilde harekete oturacak sonuçlar üretir. Bu filterde bir uçağın yapabileceği 3 değişik manevrayı bilen ve yapılan manevraları eldeki mevcut şablona bakarak karar veren bir mekanizma gerçekleştirilmiştir. Bu filtrenin önceki filtrelerden tek farkı takip edeceği cismin manevralarına karşı önceden eğitilmiş olması ve hareketleri şablona oturtarak kolayca karar verebilmesidir.

Tezin başlangıcında uçağın sadece x,y eksenlerinde hareket edeceğini ve de dolayısıyla takibin de bu iki boyut üzerinde yapılacağı belirtilmişti. İki boyut üzerinde mevcut hareketler aşağıdaki şekilde sınıflandırılabilir:

1. Düzgün Doğrusal Hareket (DDH)

Düzgün doğrusal harekette cisim sabit hızda ve bir yönde hareket eder. Hızı ve yönü değişmez. Bu hareket, Şekil 4.1'de görülmektedir:



Şekil 4.1 Düzgün doğrusal hareket

Kinematikler şu şekilde hesaplanır:

$$\text{Hız (V)} = \frac{\Delta X}{\Delta T} \quad (4.1)$$

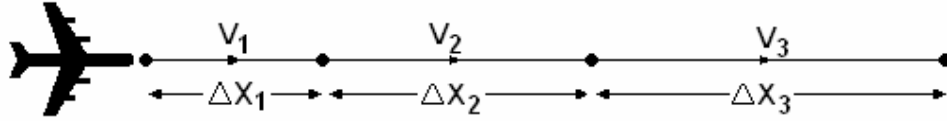
ΔT : iki nokta arası geçen süre (örnekleme aralığı)

$$\text{İvme (A)} = \Delta V = 0$$

(4.2)

2. Düzgün Doğruda Hızlanan Hareket (DDHH)

Bu hareket şeklinde cisim doğrusal olarak ilerlerken hızı eşit miktarlarda sürekli artıyor. Bu hareket Şekil 4.2’de görülmektedir:



Şekil 4.2 Düzgün doğrua hızlanan hareket

Kinematikler şu şekildedir:

$$\text{İvme} = A$$

Hız:

$$V_1 = V_1$$

$$V_2 = V_1 + A \cdot \Delta T$$

$$V_3 = V_1 + 2.A.\Delta T \quad (4.3)$$

...

Konum:

$$\Delta X_1 = V_1 \cdot \Delta T$$

$$\Delta X_2 = V_2 \cdot \Delta T = V_1 \cdot \Delta T + A.(\Delta T)^2$$

$$\Delta X_3 = V_3 \cdot \Delta T = V_1 \cdot \Delta T + 2.A.(\Delta T)^2 \quad (4.4)$$

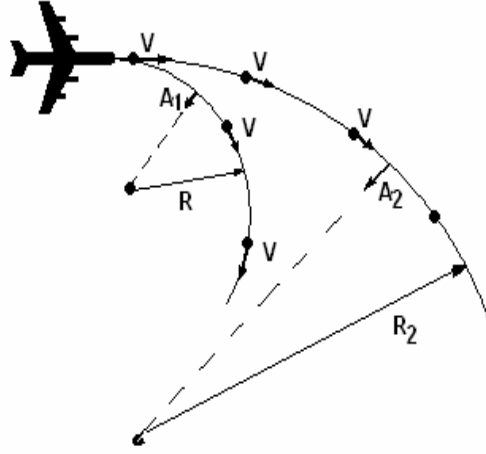
...

3. Dönme Hareketi (DH)

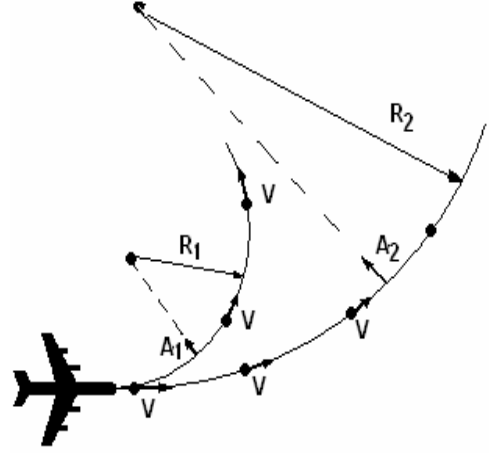
Dönme hareketinde cisim ilerleme yönünün 90° sağına veya soluna doğru dönüş yapar. Bu sırada hızda değişiklik olmaz. Dönüş tam bir daire ya da tam bir dairenin bir yayı olacak kadar sürdürülebilir:

Örnek dönüşler, Şekil 4.3’te görülmektedir:

Sağa dönüş



Sola dönüş



Şekil 4.3 Dönme hareketleri

Bu harekette kinematikler şu şekildedir:

Hız(V): Büyüklüğü sabit, yönü hareketin yönünde.

İvme(A): Büyüklüğü sabit, yönü her zaman dönüş çemberinin merkezine doğru.

$$A = \frac{V^2}{R} \quad (4.5)$$

Konum: Çemberin yayı üzerinde bulunacak şekilde eşit aralıklarla hareket eder.

Şekil 4.4'te bir hedefin dönerken çember üzerinde bulunacağı konumlar gösterilmiştir.

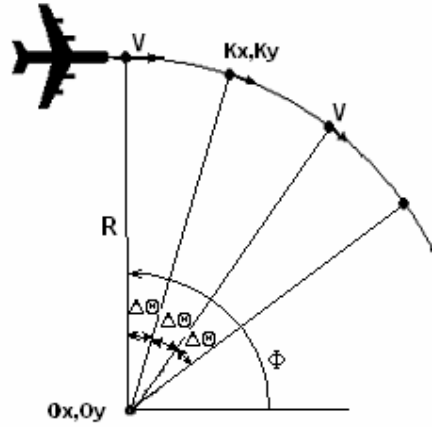
Ox, Oy: Dönüş çemberinin merkezi,

Kx, Ky: Konumu

temsil ederse:

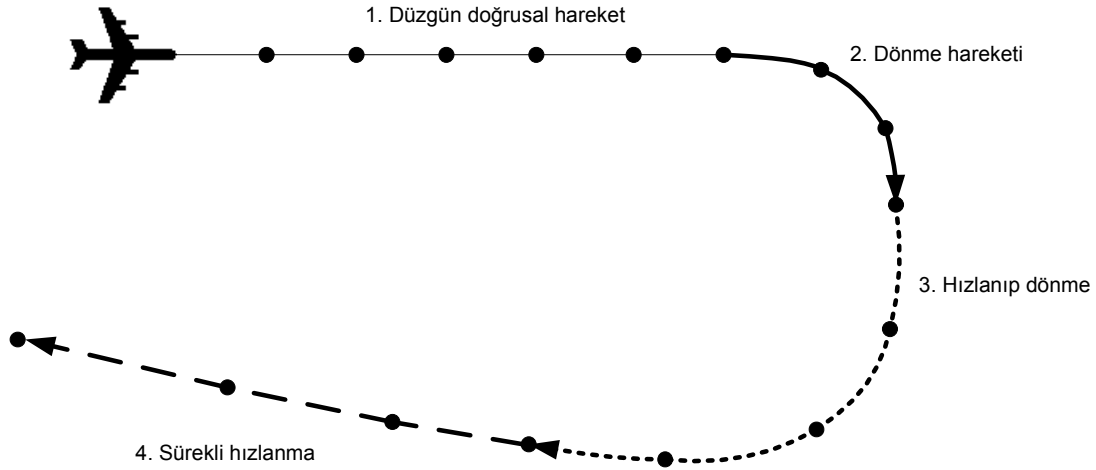
$$Kx(n) = Ox + R \cdot \cos(\Phi \pm n \cdot \Delta\Theta)$$

$$Ky(n) = Oy + R \cdot \sin(\Phi \pm n \cdot \Delta\Theta) \quad (4.6)$$



Şekil 4.4 Dönme hareketi yapan hedefin bulunacağı konumlar

Bir uçağın yapacağı tüm hareketler yukarıdaki üç çeşit hareketle temsil edilebilir. Şekil 4.5'te görüldüğü gibi her hareket aslında üç temel hareketten biridir:

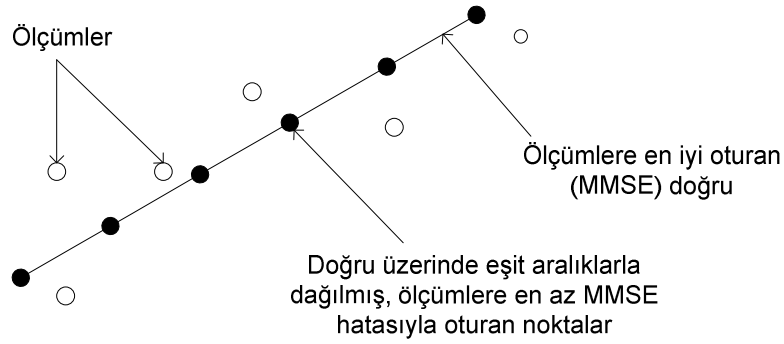


Şekil 4.5 Bir hava hedefinin hareketlerinin üç ana hareket çeşidi ile temsil edilebileceğini gösteren şekil

Dönme hareketi yaparken hızını veya ivmesini değiştiren cisim aslında yeni bir dönme hareketi yapmaktadır. O halde bu üç hareketi tek tek takip eden üç filtre bir modelde aynı anda çalıştırılabilir ve de ölçümlere en yakın sonucu veren filtre modelin çıktısı olur. Bu modelde alınan ölçümler üzerine sırayla sabit aralıklı doğru, artan aralıklı doğru ve de sabit aralıklı çember oturtulup en küçük ortalama karesel hata (MMSE) veren model seçilir ve bu modele göre kinematikler hesaplanır.

1. Ölçümlere Sabit Aralıklı Doğru Oturtma (SADO)

Gerçekte düz bir doğru üzerinde giden bir cismin gürültülü ölçüm sonucu şekli ve de bu şekil üzerine doğru oturtmak istediğimizde oluşacak şekil aşağıdaki gibi gerçekleşir (Şekil 4.6):



Şekil 4.6 Ölçümlere sabit aralıklı doğru oturtma örneği

En küçük ortalama karesel hata (MMSE) veren noktaların hesabı aşağıdaki gibi yapılır:

Xon, Yon: N. ölçümün x ve y değerleri,

Xkn, Ykn: N. karar verilen noktanın x ve y değerleri olmak üzere:

E: Toplam karesel hata

Karar verilen noktalar:

(Xk1 , Yk1)

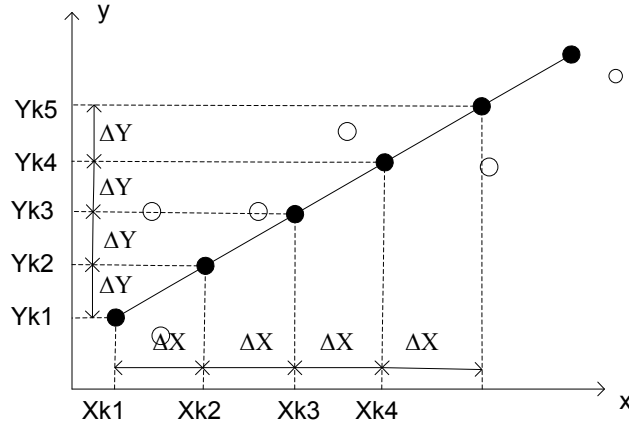
(Xk2 , Yk2) = (Xk1 + ΔX , Yk1 + ΔY)

(Xk3 , Yk3) = (Xk1 + 2.ΔX , Yk1 + 2.ΔY)

...

(Xkn , Ykn) = (Xk1 + (n-1).ΔX , Yk1 + (n-1).ΔY) (4.7)

Karar verilen noktalar Şekil 4.7’de gösterilmektedir.



Şekil 4.7 Sabit aralıklı doğru oturtma yönteminde karar verilen noktaların x ve y ekseninde iz düşümü

$$E = (X_{o1} - X_{k1})^2 + (Y_{o1} - Y_{k1})^2 + \dots + (X_{on} - X_{kn})^2 + (Y_{on} - Y_{kn})^2$$

$$E = (X_{o1} - X_{k1})^2 + \dots + (X_{on} - X_{kn})^2 + (Y_{o1} - Y_{k1})^2 + \dots + (Y_{on} - Y_{kn})^2$$

$$E_x = (X_{o1} - X_{k1})^2 + \dots + (X_{on} - X_{k1} - (n-1) \cdot \Delta X)^2 \quad (4.8)$$

$$E_y = (Y_{o1} - Y_{k1})^2 + \dots + (Y_{on} - Y_{k1} - (n-1) \cdot \Delta Y)^2 \quad (4.9)$$

$$E = E_x + E_y \quad (4.10)$$

$$E = \sum_{i=1}^n (X_{oi} - X_{k1} - (i-1)\Delta X)^2 + \sum_{i=1}^n (Y_{oi} - Y_{k1} - (i-1)\Delta Y)^2 \quad (4.11)$$

Hataı küçültmek için türev alınır ve sıfıra eşitlenir:

$$\frac{\partial E}{\partial X_{k1}} = \sum_{i=1}^n 2(X_{oi} - X_{k1} - (i-1)\Delta X) \cdot -1 = 0 \quad (4.12)$$

$$\sum_{i=1}^n X_{oi} - \frac{n \cdot (n-1)}{2} \Delta X = n \cdot X_{k1}$$

$$X_{k1} = \frac{1}{n} \sum_{i=1}^n X_{oi} - \frac{n-1}{2} \Delta X$$

$$\boxed{X_{k1} = \overline{X_o} - \frac{(n-1)}{2} \Delta X} \quad (4.13)$$

$$\frac{\partial E}{\partial \Delta X} = \sum_{i=1}^n 2(X_{oi} - X_{k1} - (i-1)\Delta X) \cdot -(i-1) = 0 \quad (4.14)$$

$$\sum_{i=1}^n (i-1)X_{oi} - X_{k1} \sum_{i=1}^n (i-1) - \Delta X \cdot \sum_{i=1}^n (i-1)^2 = 0$$

4.13'te bulunan X_{k1} değeri kullanılırsa:

$$\Delta X = \frac{12}{n.(n^2 - 1)} \sum_{i=1}^n (i-1)X_{oi} - \frac{6}{n+1} \overline{X_o} \quad (4.15)$$

X_{k1} ve ΔX kullanılarak X_{k2} , X_{k3} , ... değerleri 4.7'deki gibi bulunur.

Benzer şekilde Y_{k1} , Y_{k2} , ... Y_{kn} değerleri de aşağıdaki bağıntıdan elde edilir:

$$\Delta Y = \frac{12}{n.(n^2 - 1)} \sum_{i=1}^n (i-1)Y_{oi} - \frac{6}{n+1} \overline{Y_o} \quad (4.16)$$

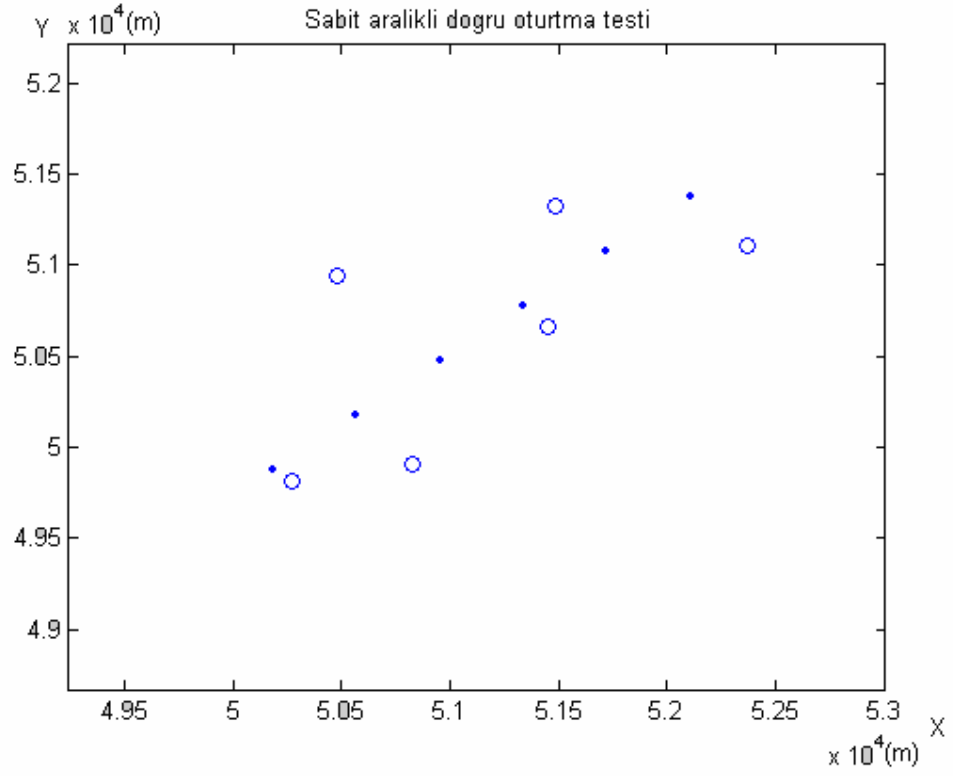
$$Y_{k1} = \overline{Y_o} - \frac{(n-1)}{2} \Delta Y \quad (4.17)$$

$$Y_{kn} = Y_{k1} + (n-1)\Delta Y \quad (4.18)$$

Bu değerler üzerinden ortalama hata:

$$\text{Ortalama Hata} = \frac{1}{n} . (\sqrt{Ex} + \sqrt{Ey}) \quad (4.19)$$

Şekil 4.8'de örnek sabit aralıklı doğru oturtma algoritmasının Matlab'da test sonuçları gösterilmektedir:



Şekil 4.8 Örnek verilere sabit aralıklı doğru oturtma algoritmasının Matlab’de test sonucu

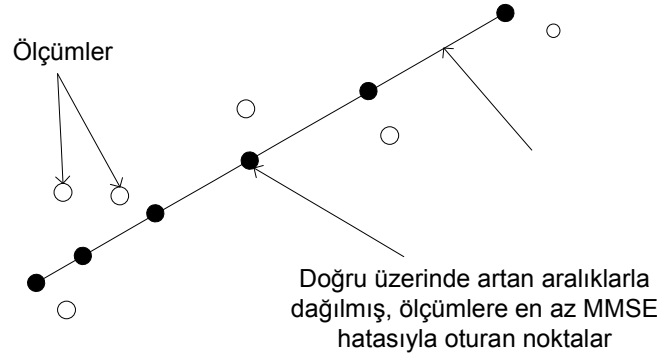
Bu teknikle cismin hız ve ivme değerleri de kolayca (4.20)’deki gibi bulunur:

$$V_i = \frac{\Delta X}{\Delta T}, \quad i = 1, 2, \dots \quad (4.20)$$

$$A_i = 0, \quad i = 1, 2, \dots$$

2. Ölçümlere Artan Aralıklı Doğru Oturtma (AADO)

Düzgün doğrudan artan hızla hareket eden cisim için ikinci bir filtre (doğru oturtma) gerekmektedir. Bu cisimden alınan ölçümler ile bu ölçümlere en iyi (MMSE) oturan noktalar Şekil 4.9'daki gibi gösterilebilir:



Şekil 4.9 Ölçümlere artan aralıklı doğru oturtma örneği

Bu noktaların hesabı aşağıdaki gibi yapılır:

$$X_1 = X_1$$

$$X_2 = X_1 + V.\Delta T + 0,5.A.\Delta T^2$$

$$X_3 = X_1 + V.(2.\Delta T) + 0,5.A.(2.\Delta T)^2$$

...

$$X_n = X_1 + V.((n-1).\Delta T) + 0,5.A.((n-1).\Delta T)^2 \quad (4.22)$$

$$a = V.\Delta T,$$

$$b = 0,5.A.(\Delta T)^2 \text{ alınırsa}$$

$$X_1 = X_1$$

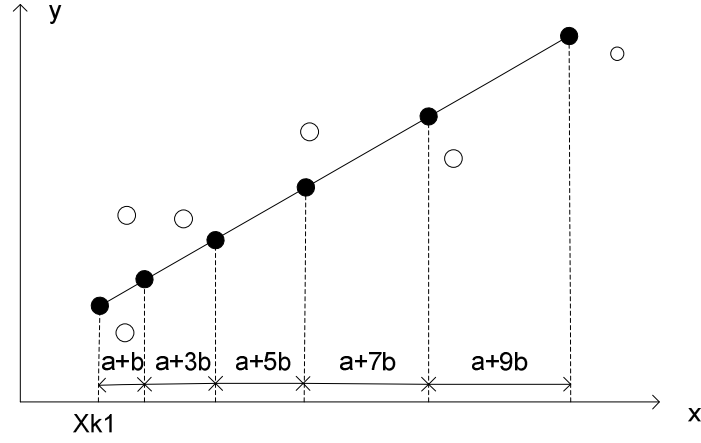
$$X_2 = X_1 + a + b$$

$$X_3 = X_1 + 2.a + 4.b$$

...

$$X_n = X_1 + (n-1).a + (n-1)^2.b \quad (4.23)$$

Bu noktalar Şekil 4.10'da gösterilmektedir.



Şekil 4.10 Artan aralıklı doğru oturtma yönteminde karar verilen noktaların x ekseninde iz düşümü

$$E = (X_{o1} - X_{k1})^2 + (Y_{o1} - Y_{k1})^2 + \dots + (X_{on} - X_{kn})^2 + (Y_{on} - Y_{kn})^2 \quad (4.24)$$

$$E_x = (X_{o1} - X_{k1})^2 + \dots + (X_{on} - X_{k1} - (n-1).a_x - (n-1)^2.b_x)^2 \quad (4.25)$$

Benzer şekilde

$$E_y = (Y_{o1} - Y_{k1})^2 + \dots + (Y_{on} - Y_{k1} - (n-1).a_y - (n-1)^2.b_y)^2 \quad (4.26)$$

$$E = E_x + E_y \quad (4.27)$$

$$E = \sum_{i=1}^n (X_{oi} - X_{k1} - (i-1)a_x - (i-1)^2 b_x)^2 + \sum_{i=1}^n (Y_{oi} - Y_{k1} - (i-1)a_y - (i-1)^2 b_y)^2 \quad (4.28)$$

X_{k1} , Y_{k1} , a_x , b_x , a_y , b_y değerlerini MMSE'yi karşılayacak şekilde bulunmalıdır:

$$\begin{aligned}
\frac{\partial E}{\partial Xk_1} &= 0 \\
\frac{\partial E}{\partial a_x} &= 0 \\
\frac{\partial E}{\partial b_x} &= 0 \\
\frac{\partial E}{\partial Yk_1} &= 0 \\
\frac{\partial E}{\partial a_y} &= 0 \\
\frac{\partial E}{\partial b_y} &= 0
\end{aligned} \tag{4.29}$$

Alınan türevler sonucunda aşağıdaki denklemler bulunur:

$$Xk_1 + \frac{(n-1)}{2}a_x + \frac{(n-1)(2n-1)}{6}b_x = \overline{Xo} \tag{4.30}$$

$$Xk_1 + \frac{2n-1}{3}a_x + \frac{n(n-1)}{2}b_x = \frac{2\sum_{i=1}^n (i-1)X_i}{n(n-1)} \tag{4.31}$$

$$\frac{n(n-1)(2n-1)}{6}Xk_1 + \left(\frac{n(n-1)}{2}\right)^2 a_x + \left(\sum_{i=1}^{n-1} i^4\right)b_x = \sum_{i=1}^n (i-1)^2 Xo_i \tag{4.32}$$

Denklemlerdeki katsayıları harflerle temsil edilirse:

$$\begin{aligned}
k_{11} &= 1 \\
k_{12} &= \frac{(n-1)}{2} \\
k_{13} &= \frac{(n-1)(2n-1)}{6} \\
k_{21} &= 1 \\
k_{22} &= \frac{2n-1}{3} \\
k_{23} &= \frac{n(n-1)}{2} \\
k_{31} &= \frac{n(n-1)(2n-1)}{6} \\
k_{32} &= k_{23}^2 \\
k_{33} &= \sum_{i=1}^{n-1} i^4 \\
l_1 &= \overline{Xo} \\
l_2 &= \frac{2 \sum_{i=1}^n (i-1) X_i}{n(n-1)} \\
l_3 &= \sum_{i=1}^n (i-1)^2 X_{oi}
\end{aligned} \tag{4.81}$$

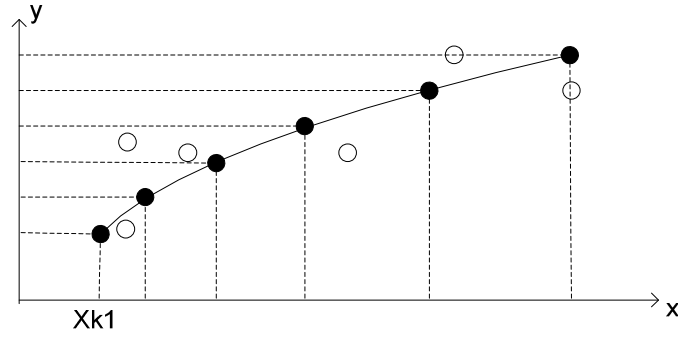
$$\begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \begin{bmatrix} Xk_1 \\ a_x \\ b_x \end{bmatrix} = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} \Rightarrow \begin{bmatrix} Xk_1 \\ a_x \\ b_x \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}^{-1} \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} \tag{4.82}$$

Benzer şekilde Yk_1 , a_y , b_y de bulunur. y değerleri için

$$\begin{aligned}
l_1 &= \overline{Yo} \\
l_2 &= \frac{2 \sum_{i=1}^n (i-1) Y_{oi}}{n(n-1)} \\
l_3 &= \sum_{i=1}^n (i-1)^2 Y_{oi}
\end{aligned} \tag{4.83}$$

kullanılır.

Yukarıdaki denklemlerde görüleceği gibi Xk ve Yk değerleri birbirlerinden bağımsız olduğu için Xk ve Yk bir doğru oluşturmayabilir. Bu da istenilen doğruya hızlanma hareketini temsil etmemiş olur. Örneğin yukarıdaki denklemler Şekil 4.11'deki gibi sonuç üretebilir:



Şekil 4.11 Ölçümlere artan aralıklı doğru oturtma yönteminde doğrusallığın sağlanamayabileceğini gösteren şekil

Bu durumda alınan ölçümlerin önce bir doğru üzerinde iz düşümünü aldıktan sonra iz düşümlerin yukarıdaki denklemlerden geçmesi daha doğru olur.

Ölçümlere en iyi oturan doğru ve bu doğru üzerindeki iz düşümler şöyle hesaplanır:

1. Ölçümlerin $y=ax+b$ doğrusu üzerinde iz düşümleri bulunur:

$$E_1 = \sum_{i=1}^n (y_i - ax_i - b)^2 \quad (4.84)$$

$$\frac{\partial E}{\partial a} = 0 \rightarrow a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i = \sum_{i=1}^n y_i x_i \quad (4.85)$$

$$\frac{\partial E}{\partial b} = 0 \rightarrow a \sum_{i=1}^n x_i + b.n = \sum_{i=1}^n y_i \quad (4.86)$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum x_i^2 & \sum x_i \\ \sum x_i & n \end{bmatrix}^{-1} \cdot \begin{bmatrix} \sum y_i x_i \\ \sum y_i \end{bmatrix} \quad (4.87)$$

İz düşüm noktaları:

$$\begin{aligned} \hat{X}_{O_i} &= X_{O_i} \\ \hat{Y}_{O_i} &= a\hat{X}_{O_i} + b \end{aligned} \quad (4.88)$$

2. Ölçümlerin $x=ay+b$ doğrusu üzerinde iz düşümleri bulunur:

$$E_2 = \sum_{i=1}^n (x_i - ay_i - b)^2 \quad (4.89)$$

$$\frac{\partial E}{\partial a} = 0 \rightarrow a \sum_{i=1}^n y_i^2 + b \sum_{i=1}^n y_i = \sum_{i=1}^n y_i x_i \quad (4.90)$$

$$\frac{\partial E}{\partial b} = 0 \rightarrow a \sum_{i=1}^n y_i + b.n = \sum_{i=1}^n x_i \quad (4.91)$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum y_i^2 & \sum y_i \\ \sum y_i & n \end{bmatrix}^{-1} \begin{bmatrix} \sum y_i x_i \\ \sum x_i \end{bmatrix} \quad (4.92)$$

İz düşüm noktaları:

$$\begin{aligned} \hat{Y}_{o_i} &= Y_{o_i} \\ \hat{X}_{o_i} &= a\hat{Y}_{o_i} + b \end{aligned} \quad (4.93)$$

3. En az hatayı veren iz düşüm modeli uygulanır

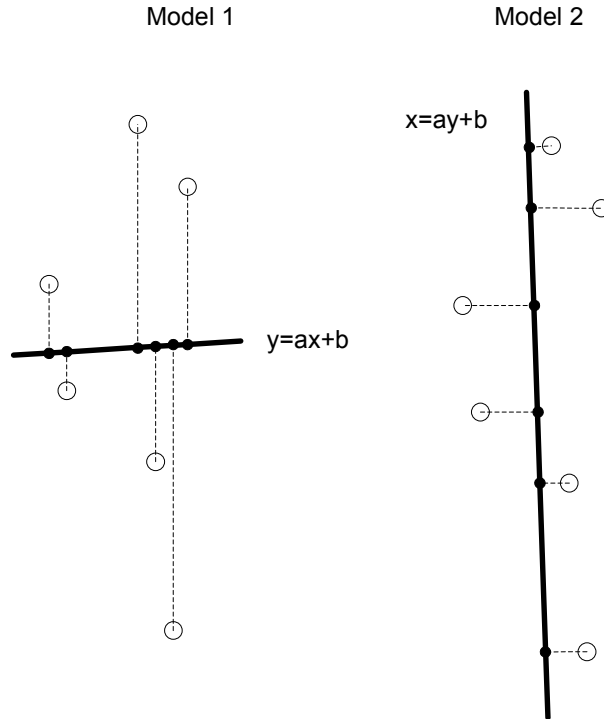
eğer $E_1 > E_2$

İkinci iz düşüm modelini kullan

değilse

Birinci iz düşüm modelini kullan

Eğer cisim $x=a$ doğrusunda ilerliyorsa yukarıdaki ikili modelin faydası daha iyi görülebilir:

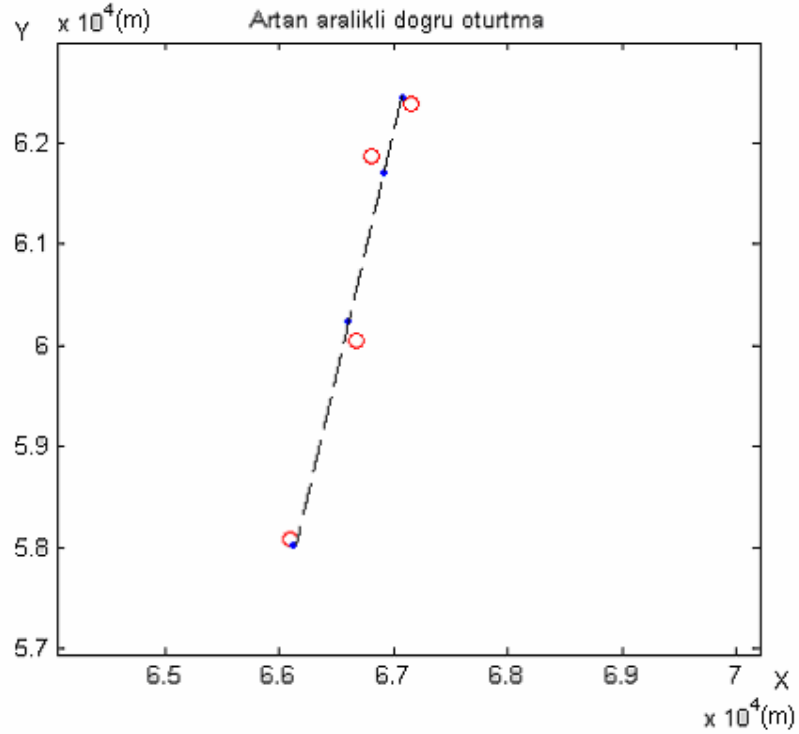


Şekil 4.12 $x=a$ doğrusuna yakın hareket eden hedefe doğru oturtma modelleri

Şekil 4.12’de de görüldüğü gibi cismin hareketine uygun doğrulaştırma modeli Model 2’dir.

O halde özetle artan aralıklı doğru oturtma filtresinde önce ölçümlerin bir doğru üzerine izdüşümü yukarıdaki gibi bulunur daha sonra izdüşüm değerleri üzerinden X_k , Y_k , ve a, b değerleri bulunur.

Filtrenin Matlab’de testi Şekil 4.13’te verilmiştir.



Şekil 4.13 Matlab’de örnek verilere artan aralıklı doğru oturtma algoritması test sonucu

Filtrenin ürettiği X_{k1} , Y_{k1} , a, b değerlerinden yola çıkılarak konum, hız ve ivme değerleri ve ortalama hata şöyle bulunur:

$$X_i = X_{k1} + (i-1).a_x + (i-1)^2.b_x, \quad i = 1, 2, \dots, n \quad (4.94)$$

$$Y_i = Y_{k1} + (i-1).a_y + (i-1)^2.b_y, \quad i = 1, 2, \dots, n \quad (4.95)$$

$$A_{x_i} = \frac{2.b_x}{(\Delta T)^2}, \quad i = 1, 2, \dots, n \quad (4.96)$$

$$A_{y_i} = \frac{2.b_y}{(\Delta T)^2}, \quad i = 1, 2, \dots, n$$

$$Vx_i = \frac{a_x}{\Delta T} + b_x(i-1).\Delta T, \quad i = 1, 2, \dots, n \quad (4.97)$$

$$Vy_i = \frac{a_y}{\Delta T} + b_y(i-1).\Delta T, \quad i = 1, 2, \dots, n$$

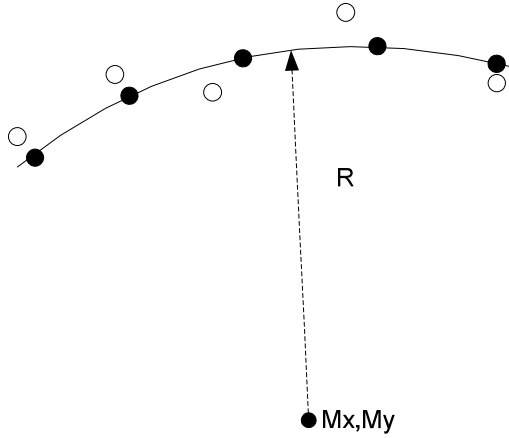
$$Ex = (Xo1 - Xk1)^2 + \dots + (Xon - Xk1 - (n-1). \Delta X)^2 \quad (4.98)$$

$$Ey = (Yo1 - Yk1)^2 + \dots + (Yon - Yk1 - (n-1). \Delta Y)^2 \quad (4.99)$$

$$\text{Ortalama hata} = \frac{1}{n} . (\sqrt{Ex} + \sqrt{Ey}) \quad (4.100)$$

4. Ölçümlere Sabit Aralıklı Çember/Yay Oturtma (SAÇO)

Çember oturtma problemi daha önceki doğru oturtma probleminden daha zor olmaktadır. Cisim tam bir çember hareketi yapmak zorunda değildir. Bu durumda en uygun çember yayı bulunmalı ve bu yay üzerinde en küçük karesel hata (MMSE)'yı veren eşit aralıklı noktalar belirlenmelidir. Şekil 4.14'te örnek ölçümler ile uygun çember ve bu çember üzerindeki uygun (MMSE) noktalar gösterilmiştir.



Şekil 4.14 Ölçümlere sabit aralıklı çember oturtma örneği

4.1 Çember Oturtma

Çember oturtma algoritması aşağıdaki gibi çalışmaktadır:

1. Merkez noktası olarak başlangıçta ölçümlerin ortalaması alınır:

$$\begin{aligned} Mx &= \overline{Xo}, \\ My &= \overline{Yo} \end{aligned} \quad (4.101)$$

2. Yarıçap (R) merkeze göre en az karesel hatayı verecek şekilde hesaplanır:

$$\begin{aligned} E &= \sum_{i=1}^n \left((Xo_i - Mx)^2 + (Yo_i - My)^2 - R^2 \right)^2 \text{ min olması için} \\ R^2 &= \frac{1}{n} \sum_{i=1}^n \left((Xo_i - Mx)^2 + (Yo_i - My)^2 \right) \end{aligned} \quad (4.102)$$

3. Merkez ve yarıçap değerlerine göre hata(E) 2. adımdaki gibi bulunur

4. Merkez noktası etrafında d mesafe uzaklıkta +x, -x, +y, -y yönlerinde dört aday nokta üzerine merkez kaydırılarak yeni yarıçap(R_{+x} , R_{-x} , R_{+y} , R_{-y}) ve hata (E_{+x} , E_{-x} , E_{+y} , E_{-y})

değerleri 2'deki gibi bulunur.

5. Yeni hata değerleri ile mevcut hatadan hangisi en küçükse merkez noktası o noktaya kalıcı olarak kaydırılır. Yeni hataların hepsi mevcut hatadan büyükse merkez noktası eski yerinde kalır ve d mesafesi yarıya indirilir.

6. Belirli bir döngü sayısına veya d mesafesi çok küçük bir değere inene kadar 4 ve 5 tekrarlanır.

Yukarıdaki algoritmanın bulduğu en iyi çember Matlab'de denenmiştir.

Jmax : Maksimum döngü sayısı

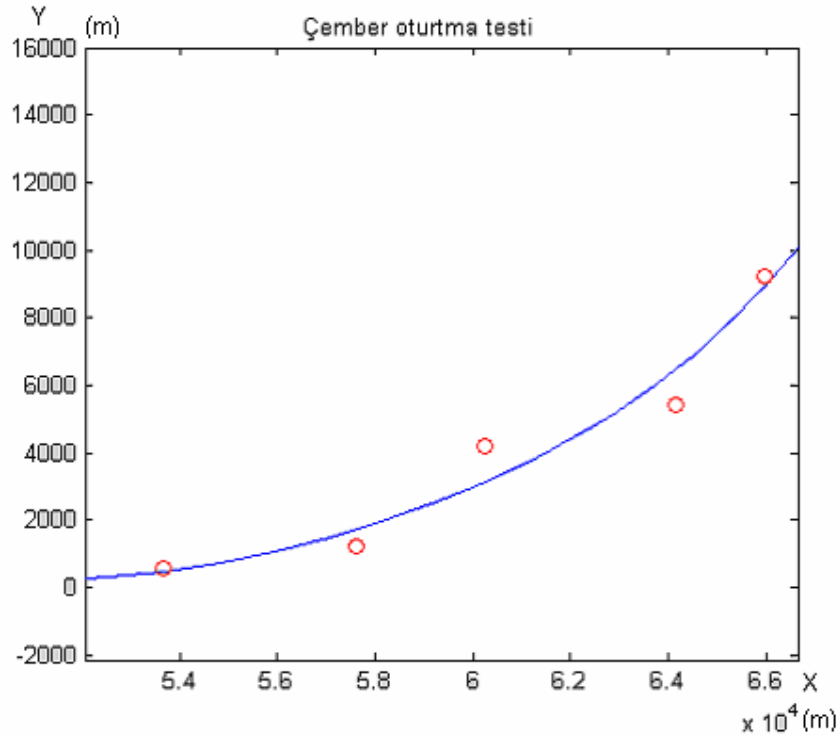
d : Merkezi kaydırma mesafesi

dmin : Minimum kaydırma değeri olmak üzere

Jmax = 100,

d = 4000 (metre)

dmin = 10 (metre) ile Şekil 4.15'teki sonuç bulunmuştur:



Şekil 4.15 Matlab'de ölçümlere çember oturtma algoritması test sonucu

3.2 Çember üzerinde eşit aralıklı MMSE noktaları bulmak:

Çember üzerindeki noktalar Şekil 4.4’de gösterildiği gibi tanımlanabilir:

$$Kx_1 = Mx + R.\cos(\Phi)$$

$$Kx_2 = Mx + R.\cos(\Phi+\Delta\Theta)$$

$$Kx_3 = Mx + R.\cos(\Phi+2.\Delta\Theta)$$

...

$$Ky_1 = My + R.\sin(\Phi)$$

$$Ky_2 = My + R.\sin(\Phi+\Delta\Theta)$$

$$Ky_3 = My + R.\sin(\Phi+2.\Delta\Theta)$$

...

Burada görüldüğü gibi noktalar yarıçap ve açı düzleminde temsil edildiğinde açı doğrusal olarak artıyor/azalıyor. O zaman bu düzlemde ölçümlerin de çember üzerindeki iz düşümünü alırsak MMSE noktaları sabit aralıklı doğru oturtma filtresinden geçirilerek bulunabilir:

Algoritma:

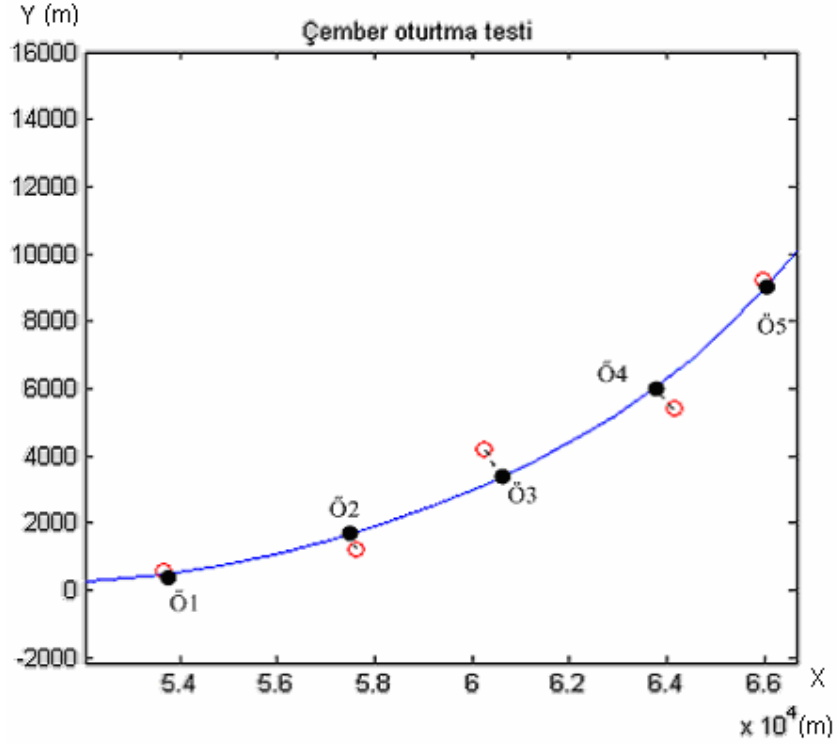
1. Ölçümlerin çember üzerindeki iz düşümünü bul:

$$\Phi_{O_i} = \tan^{-1}(Ox_i - Mx, Oy_i - My) \quad (4.103)$$

$$\ddot{O}x_i = Mx + R.\cos(\Phi_{O_i}) \quad (4.104)$$

$$\ddot{O}y_i = My + R.\sin(\Phi_{O_i}) \quad (4.105)$$

Bu noktalar Şekil 4.16’da gösterilmiştir.



Şekil 4.16 Ölçümlerin çember üzerindeki iz düşümü

2. İzdüşümlerin açılarını SADO filtresinden geçirerek gerçekte olmaları gereken açıları Şekil 4.17'deki gibi bul:



Şekil 4.17 İzdüşüm açılarının düzeltilmesi

Not: Ölçümlerden alınan açılar 180° veya 0° etrafında ise, örneğin 160, 175, 190, 210..., 180° 'den büyük açılar küçük açılarla olan devamlılığı sağlanmalıdır. Eğer yapılan hesaplamada 190 ve 210 yerine -170 ve -150 kullanılırsa açılar SADO filtresine girmeden önce devamlılığı bozulmuş olur ki bu da hatalara yol açar. Yani 160, 175, 190, 210..., serisinde $\Delta\Theta = 16^\circ$ çıkacakken seri 160, 175, -170, -150 ile SADO filtresinde $\Delta\Theta = 120^\circ$ gibi hatalı sonuç üretir. 0° etrafında da bozulma olabilir. Örneğin 20, 15, 5, -10, -15 olan seri 20, 15, 5, 350, 345 olarak alınırsa benzer hatayı üretir. Devamlılığı sağlamak için bir çözüm olarak açılar $(-180, +180)$ arasında alınıp aşağıdaki gibi bir algorithmadan geçirilmelidir:

i: 2'den n'ye kadar

eğer ($\Phi_{O_i} - \Phi_{O_{i-1}} < -180$) doğru ise

$$\Phi_{O_i} = \Phi_{O_i} + 360$$

eğer ($\Phi_{O_i} - \Phi_{O_{i-1}} > 180$) doğru ise

$$\Phi_{O_i} = \Phi_{O_i} - 360$$

3. Karar verilen noktaları, hız ve ivmeyi hesapla:

$$\boxed{X_{k_i} = M_x + R \cdot \cos(\Phi_{k_i})} \quad (4.106)$$

$$\boxed{Y_{k_i} = M_y + R \cdot \sin(\Phi_{k_i})} \quad (4.107)$$

$$V_i = \frac{R \cdot \Delta \Theta}{\Delta T}, \quad i = 1, 2, \dots, n \quad (4.108)$$

$$A_i = \frac{V_i^2}{R}, \quad i = 1, 2, \dots, n$$

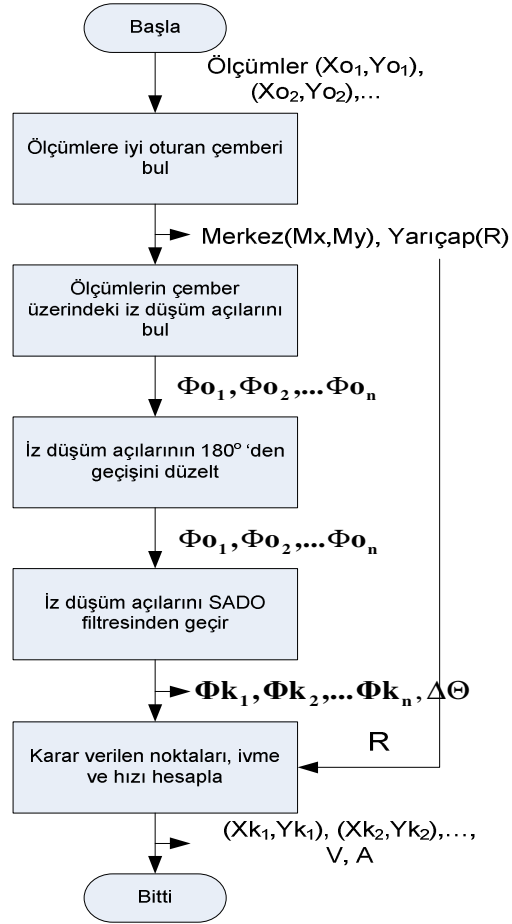
5. Toplam ortalama hatayı hesapla

$$E_x = (X_{O1} - X_{k1})^2 + \dots + (X_{On} - X_{k1} - (n-1) \cdot \Delta X)^2 \quad (4.109)$$

$$E_y = (Y_{O1} - Y_{k1})^2 + \dots + (Y_{On} - Y_{k1} - (n-1) \cdot \Delta Y)^2 \quad (4.110)$$

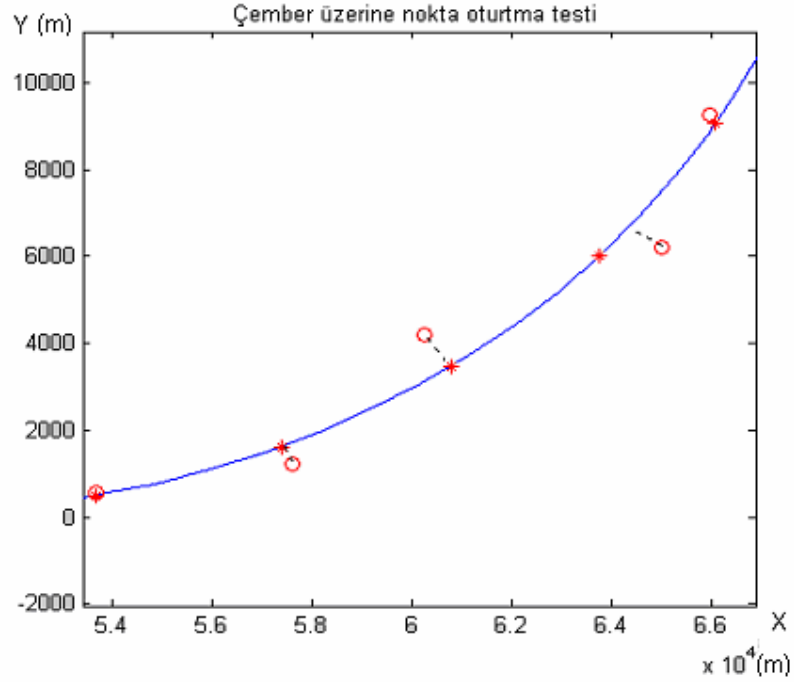
$$\text{Ortalama hata} = \frac{1}{n} \cdot (\sqrt{E_x} + \sqrt{E_y}) \quad (4.111)$$

Genel akışı Şekil 4.18'de verilmiştir.



Şekil 4.18 SAÇO filtresi akış diyagramı

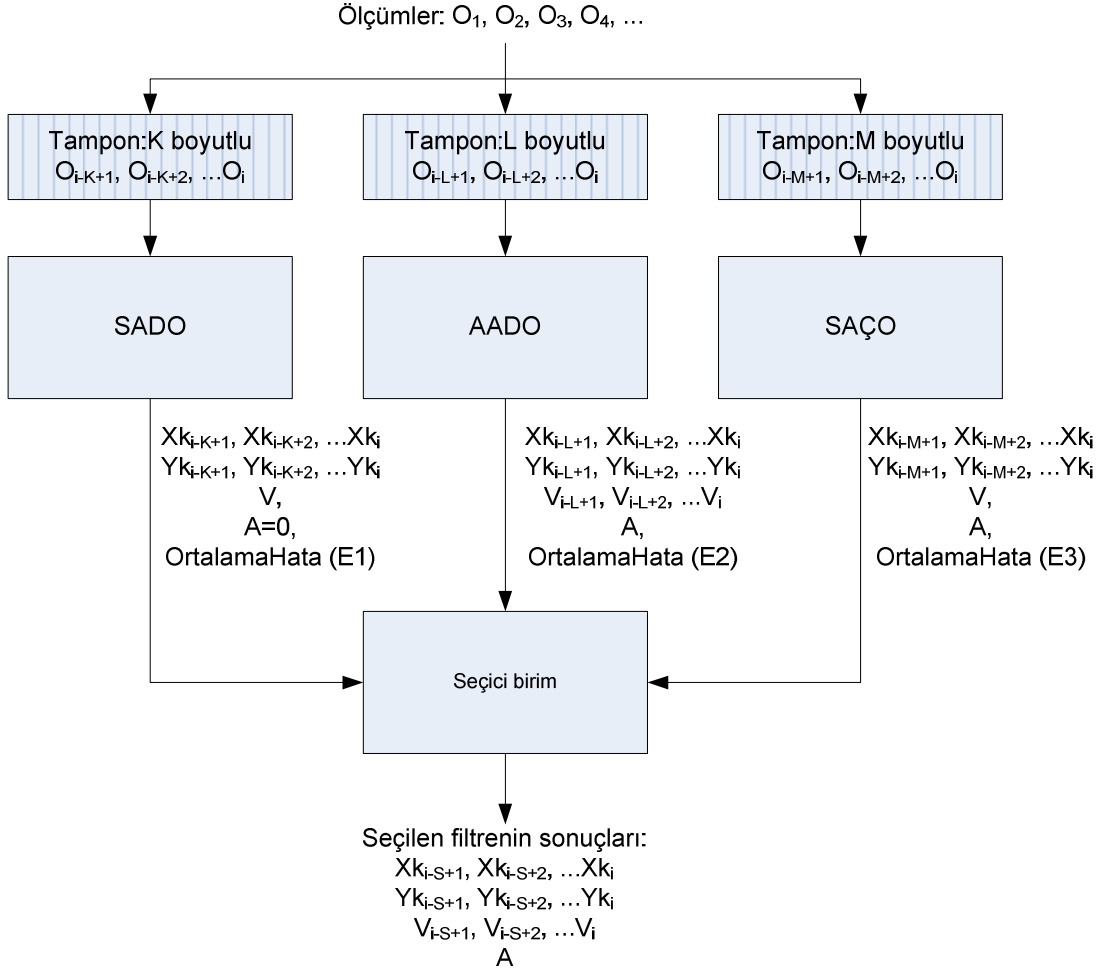
Filtrenin Matlab'deki sonucu, Şekil 4.19'da verilmiştir.



Şekil 4.19 SAÇO filtresinin örnek verilerle Matlab’de test sonucu

Şablon Filtresi Genel Akışı

Yukarıdaki üç ana filtrenin, cismin yapacağı bütün hareketleri takip edebileceği görülmüştü. Şablon filtresinde bu üç filtreyi, NIMM (etkileşimsiz çoklu model) filtresinde olduğu gibi paralel çalıştırıp, en az hata veren filtrenin sonucu Şablon filtresinin sonucu olarak verilebilir. Şablon filtresinin genel akışı Şekil 4.20’de verilmiştir.



Şekil 4.20 Şablon filtresi akış diyagramı

Tampon boyutu seçme:

Her filtre için ayrı tampon kullanılır. Tampon içindeki ölçüm verileri üzerinde filtreler çalıştırılır. Tampon boyutu olarak farklı değerler kullanılabilir. Bütün filtreler için ideal boyut olarak 5 alınabilir. Tampon boyutu en az 3 olmalıdır. Düzgün doğrusal ilerleyen cisim için tampon boyutu artırıldıkça SADO daha güzel sonuçlar verir. Düzgün doğrusal hareket sona erdiğinde tampon boyutu başlangıçtaki en küçük değere düşürülmelidir. Burada da ayrıca düzgün doğrusal hareketi kontrol eden bir mekanizma kullanılmalıdır. Bu işlem için basitçe üretilen hız değerleri kullanılmaktadır. Üretilen hız değerinde değişme başladığında düzgün doğrusal hareketin bittiği yeni bir ivmeli hareketin başladığı anlaşılır, böylece tampon boyutu başlangıçtaki en küçük değere düşürülür.

Seçici birim tasarımı:

Seçici birim üç filtrenin ortalama hata değerlerine bakarak hangi filtrenin sonucunu Şablon

filtresinin genel sonucu olarak vereceğine karar verir. Bu işlem için en basit mantık en küçük hata değerini veren filtrenin sonucunu kullanmaktır. Yapılan denemeler sonucunda bu mantık çoğunlukla başarılı sonuç vermekle birlikte bazı durumlarda hatalı sonuç üretmiştir. Örneğin aslında bir doğru üzerinde ilerleyen cismin gürültülü ölçümleri 4-5 eleman üzerinden bakıldığında bir yay oluşturabilir. Bu da doğrusal giden bir cismin Şablon filtresi sonucunu bölük parçalara ayırabilir. Bu küçük sapmaları önlemek için aşağıdaki gibi bir mantık izlenmiştir:

Başlangıçta seçilen filtre = Filtre 1 (SADO)

eğer ($\text{enküçük}hata_{i-2}=E1$) ve ($\text{enküçük}hata_{i-1}=E1$) ve ($\text{enküçük}hata_i=E1$) ise

seçilen filtre = Filtre 1 (SADO)

eğer ($\text{enküçük}hata_{i-2}=E2$) ve ($\text{enküçük}hata_{i-1}=E2$) ve ($\text{enküçük}hata_i=E2$) ise

seçilen filtre = Filtre 2 (AADO)

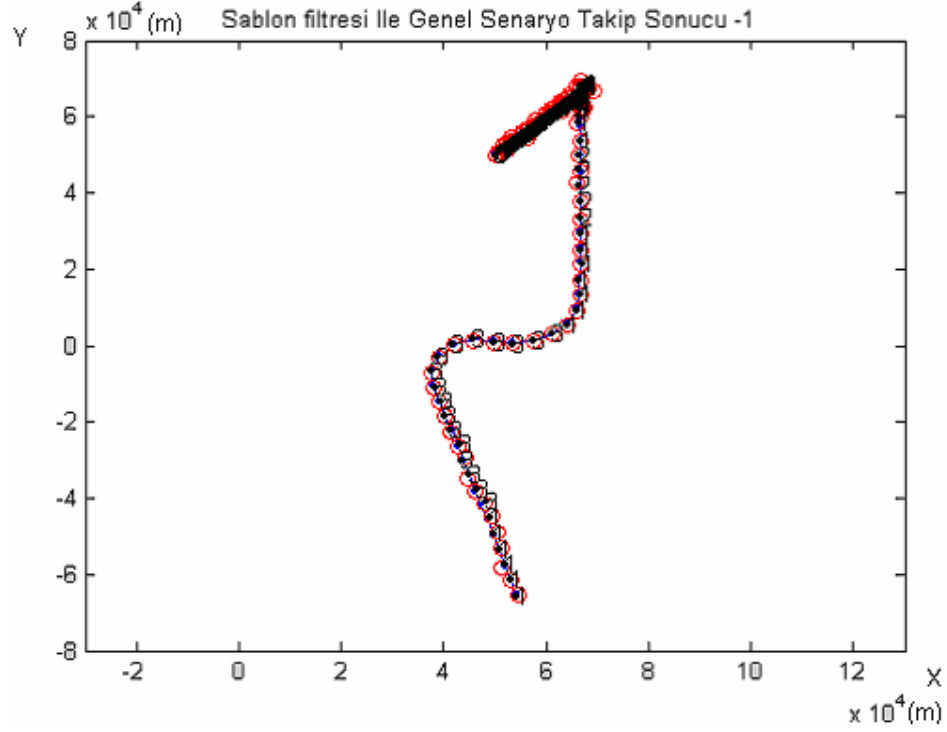
eğer ($\text{enküçük}hata_{i-2}=E3$) ve ($\text{enküçük}hata_{i-1}=E3$) ve ($\text{enküçük}hata_i=E3$) ise

seçilen filtre = Filtre 3 (SAÇO)

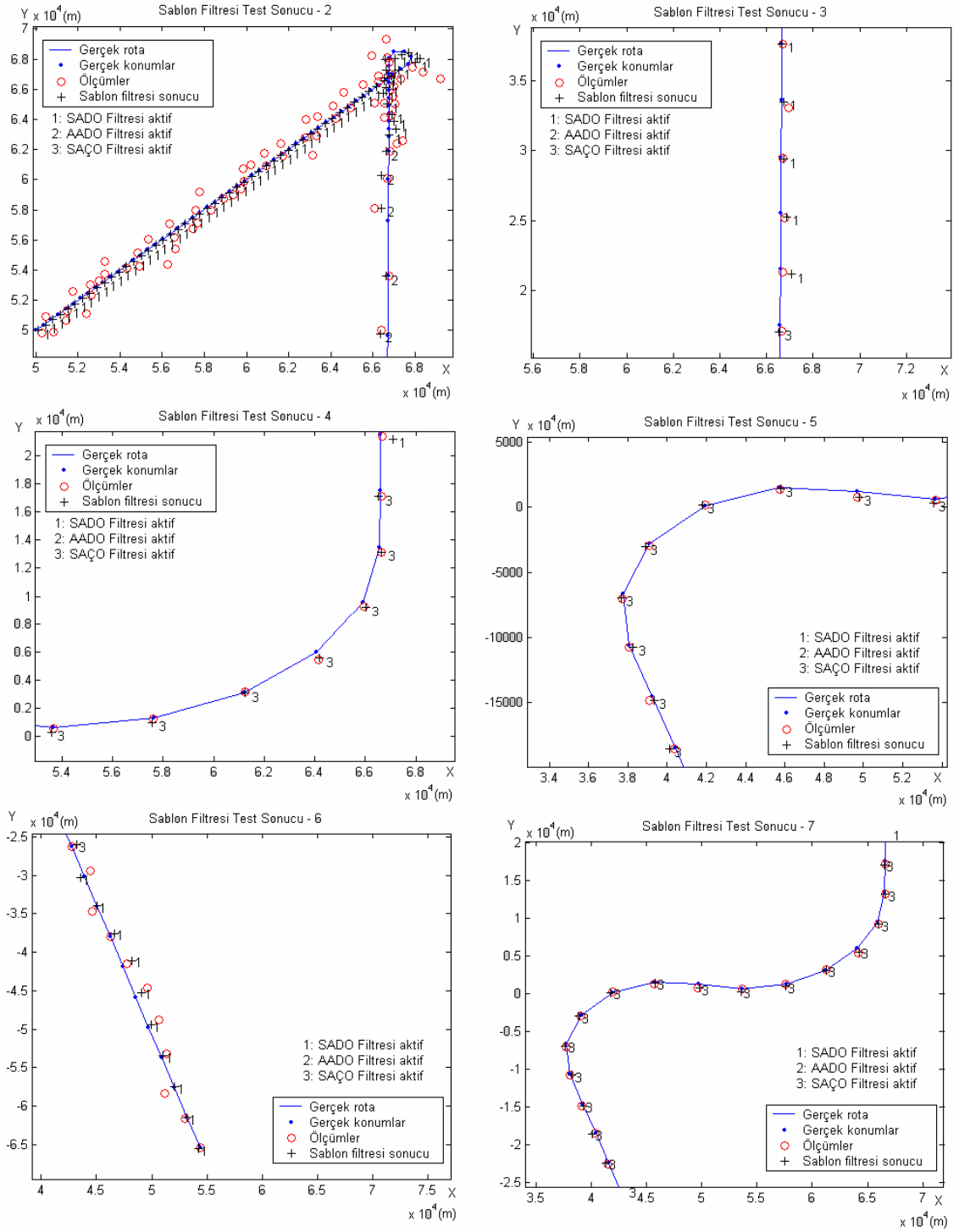
Kısaca üç defa üst üste en az hatalı sonuç üreten filtre aktif filtre olarak seçiliyor. Bu mantık sonucunda hatalı filtre seçimi oldukça azalmıştır.

4.1.1 Şablon filtresinin genel senaryo test sonuçları

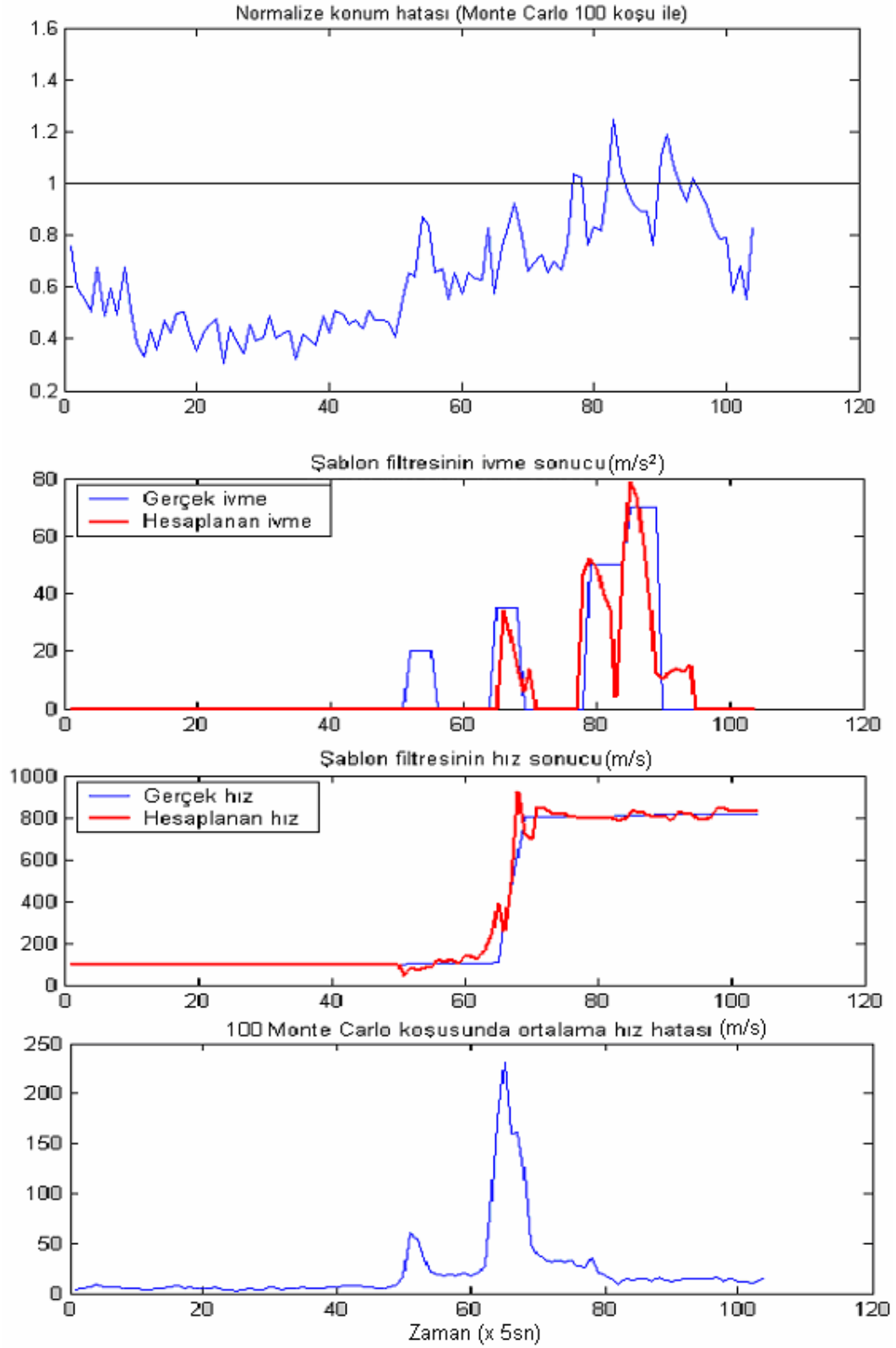
Diğer bütün filtrelerde olduğu gibi Şablon filtresinin genel senaryo üzerindeki test sonuçları analiz edilmiştir. Şekil 4.21, Şekil 4.22 ve Şekil 4.23'te Şablon filtresinin başarımları verilmektedir. Şekil 4.21'de genel olarak filtrenin ürettiği motif gerçeğe oldukça yakın olduğu görülmektedir. Aşağıdaki şekilde "o"lar ölçümleri, siyah noktalar ise filtrenin ürettiği konum değerlerini göstermektedir.



Şekil 4.21 Şablon filtresinin genel senaryoda başarımları - 1



Şekil 4.22 Şablon filtresinin genel senaryodaki başarımı - 2



Şekil 4.23 Şablon filtresi kinematik sonuçlarının analizi

Şekil 4.21, Şekil 4.22 ve Şekil 4.23'te görüldüğü üzere şablon filtresi gerçek değerlere oldukça yakın sonuç vermekle birlikte motif olarak gerçek harekete çok yakın sonuç üretmektedir. Düzgün doğrusal harekete doğru oturma tekniği sonucunda düzgün doğrusal hareket eden bir cisim için Şablon filtresi de düzgün doğrusal noktalar üretiyor. Diğer filtrelerde bu anlarda üretilen sonuçların doğruya yakın olabilmesi için kovaryansı oldukça küçültmek zorunda kalıyorduk ki yine de tam düzgün doğru elde edemiyorduk. Benzer şekilde manevra anlarında diğer filtreler x ve y eksenlerinde dönme hareketini x ve y eksenlerinde ayrı ayrı ivmeli hareket olarak modellemek zorunda kalırken Şablon filtresi dönme hareketini direk sonuca götürecek çember oturtma yöntemiyle gerçekleştirmektedir. Bu yetenekler sayesinde Şablon filtresi gerçeğe en yakın sonuçlar üretmektedir.

Şablon filtresi bu yeteneğini “hazır ve bilinen” hareket modellerine sahip olmasından alır. Önceki filtreler gürültülü ölçümlerden yola çıkarak manevra anında ivme hesabı yapıp bir sonraki değeri tahmin etmeye çalışırken Şablon filtresi geçmiş ölçümleri de hesaba katarak yapılan harekete uygun doğruyu oturtmak suretiyle en iyi konum değerlerini üretip bu doğrudan hız ve ivme hesabı yapmaktadır.

Şablon filtresinin diğer bir avantajı her ölçüm alındığında sadece son ölçüme göre değil, geçmiş ölçümleri de dahil edecek şekilde topluca bakıp daha sağlıklı kestirim yapması ve böylece geçmişten son ölçüme kadar bütün konum ve kinematik değerleri yeniden güncelleyebilmesidir. Yani kısaca geçmiş değerleri düzeltme böylece geçmişe bakarak mevcut anı daha doğru analiz etme yeteneğine sahiptir.

Şablon filtresi diğer filtrelere göre anlaşılabilirliği ve modifikasyonu daha kolaydır. Diğer filtrelerde kovaryans matrisi bilgisi gerekmektedir. Kovaryans matrisinin baştan uygun seçilmemesi veya büyük/küçük seçilmesine göre sonuç etkilenmektedir. Bir senaryoda iyi sonuç üreten kovaryans matrisi başka senaryoda küçük veya aşırı büyük kalabilir. Oysa şablon filtresinde kovaryans hesabı yapmaya gerek yoktur.

Şablon filtresinin diğer filtrelerle rakamsal olarak karşılaştırma sonuçları da aşağıda verilmiştir.

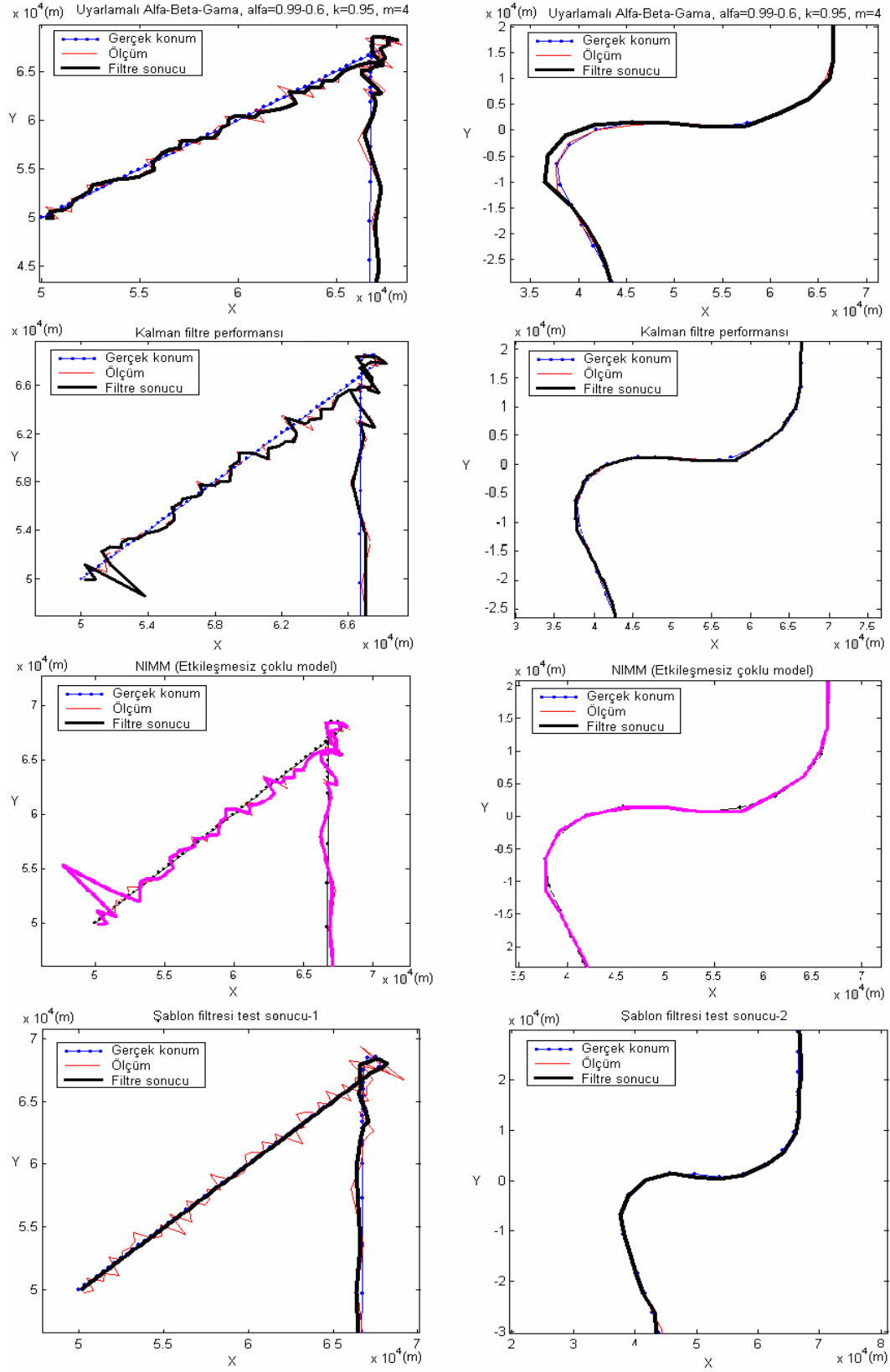
4.1.2 Şablon filtresinin diğer filtrelerle karşılaştırılması

Şablon filtresi diğer filtrelerle aşağıdaki dört ana başlıkta karşılaştırılmaktadır:

1. Motife göre karşılaştırma

Şekil 4.24'teki figürlerden de görülebileceği gibi Şablon filtresinin oluşturduğu motif diğer

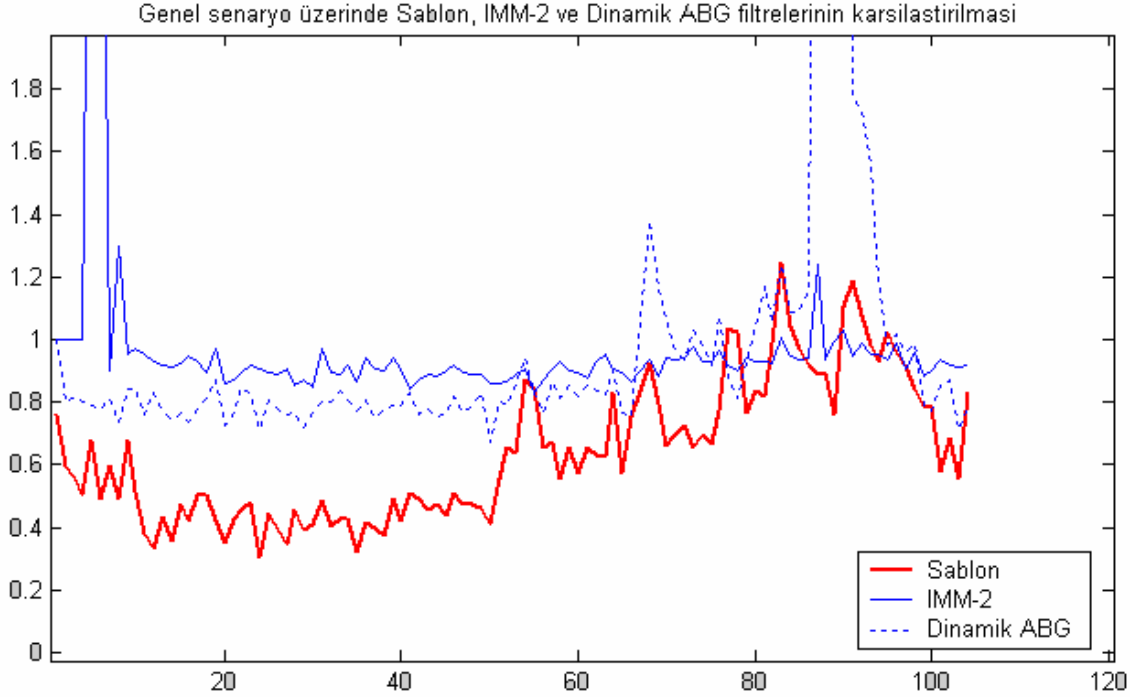
temel filtreleere göre gerçeğe çok daha yakın ve yumuşaktır. Diğer filtrelerdeki gibi zigzaglar yoktur. Şablon filtresinin bu açıdan farkı ve avantajı oldukça açık bir şekilde görülmektedir. Özellikle uzun bir süre boyunca düzgün doğrusal giden cisimler için ürettiği sonuç diğer filtreleere göre oldukça başarılı ve neredeyse sıfır hatalıdır. Şablon filtresinin ürettiği motif, hedefin rotasını belirlemek açısından daha düzgün sonuçlar vermektedir.



Şekil 4.24 Şablon filtresinin motif başarısı açısından diğer filtrelerle karşılaştırılması

2. Normalize konum hatasına göre karşılaştırma

Daha önce incelemiş olduğumuz en başarılı filtreler ile şablon filtresinin normalize konum hataları karşılaştırılmıştır. Alfa-Beta, Alfa-Beta-Gama ve Uyarlamalı Alfa-Beta-Gama grubunda en iyi performansı Uyarlamalı Alfa-Beta-Gama; Kalman ve IMM filtreleri grubunda ise en iyi performansı IMM-2 (NIMM: etkileşmesiz çoklu model) gösterdiği için karşılaştırmada bu filtreler kullanılmıştır.



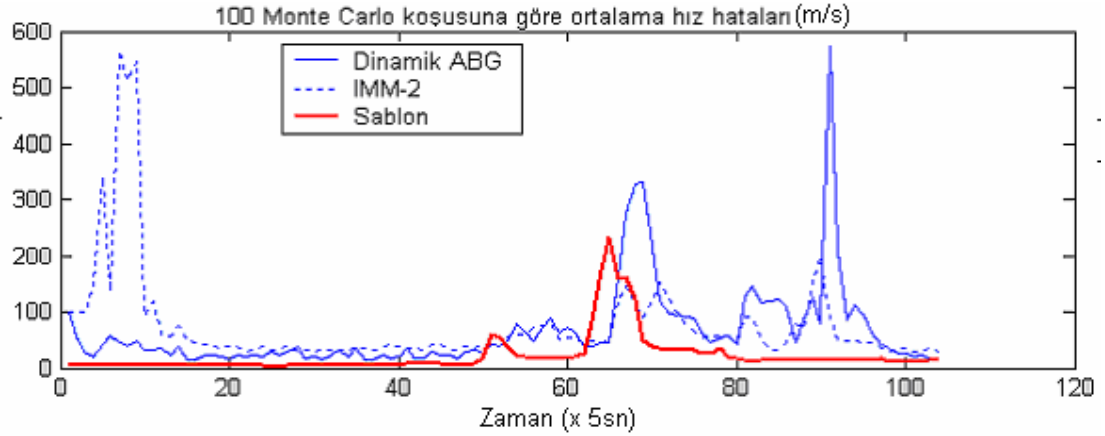
Şekil 4.25 Genel senaryo ile test edilen Şablon, IMM-2 ve Uyarlamalı(Dinamik) Alfa-Beta-Gama filtrelerinin NKH karşılaştırması

Şekil 4.25'te görüldüğü gibi şablon filtresinin başarısı diğer iki filtreye göre çok daha fazladır. Özellikle gürültünün en etkili olduğu düzgün doğrusal hareket bölgesinde (1 ile 50nci örnek arası) başarının farkı çok daha belli olmaktadır.

3. Ortalama hız hatasına göre karşılaştırma

Ortalama hız hatasına göre Şablon filtresinin IMM-2 ve Uyarlamalı(Dinamik) ABG (Alfa-Beta-Gama) filtrelerinin karşılaştırması Şekil 4.26'da verilmiştir. Şekilde de görüldüğü gibi Şablon filtresi ortalama hız hatası değerinde de diğer filtrelerden daha başarılıdır. Özellikle düzgün doğrusal hareket anında Şablon filtresi sıfıra yakın hata üretmiştir. Bununla birlikte diğer filtrelerin ürettiği hız değerinde belli bir ortalama değer etrafında oynamalar vardır. Oysa Şablon filtresinin ürettiği hız değerinde sabitlik vardır. Hız

değerinde değişimler operatörler için rahatsızlık verdiği bilinmektedir. Bununla birlikte hız değerinde oynamalar hedefi tanımak için hedef kinematiklerini kullanan sistemlerde hataya yol açmaktadır. Şablon filtresinin ürettiği hız değerlerinde oynama olmadığı için Şablon filtresi bu açıdan da avantajlıdır.



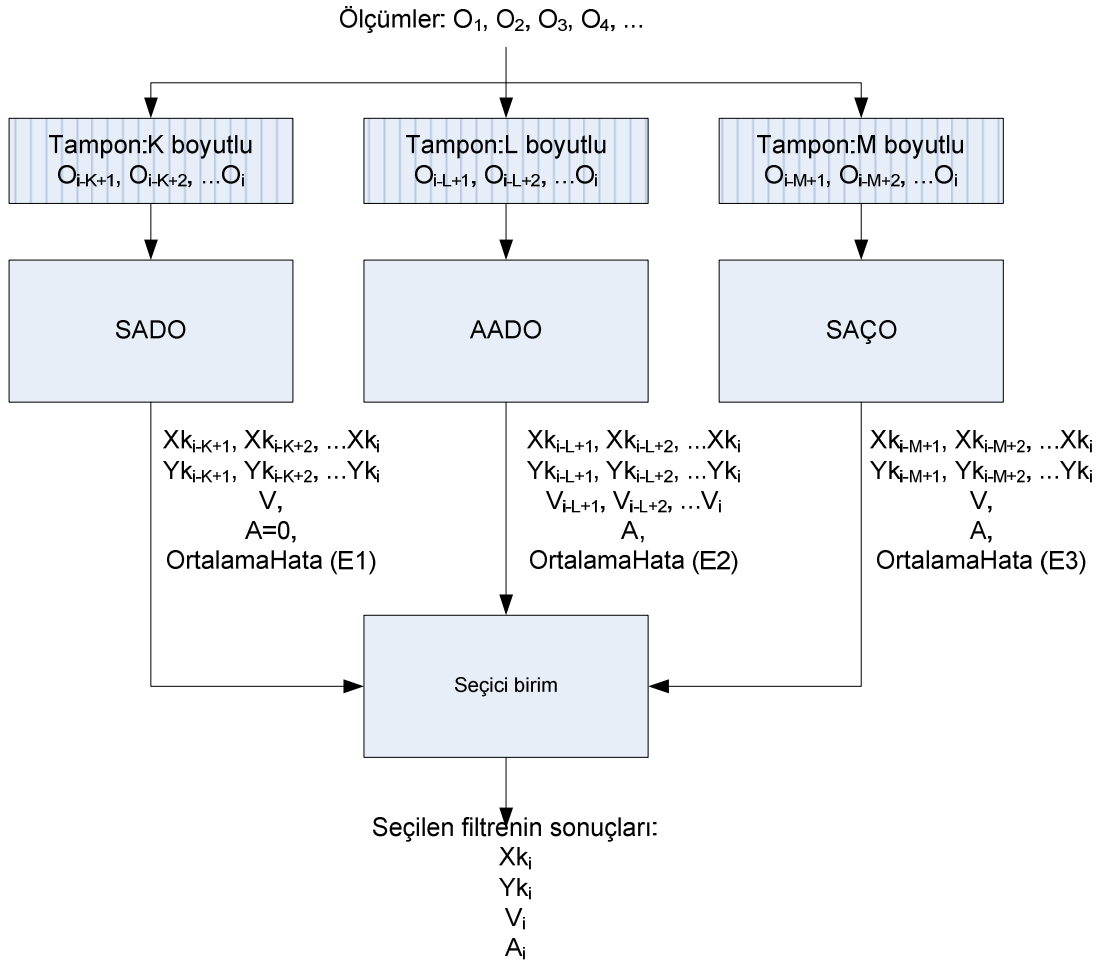
Şekil 4.26 Genel senaryo ile test edilen Şablon, IMM-2 ve Dinamik ABG filtrelerinin ortalama hız hataları karşılaştırması

4.1.3 Gerçek Zamanlı Şablon Filtresi

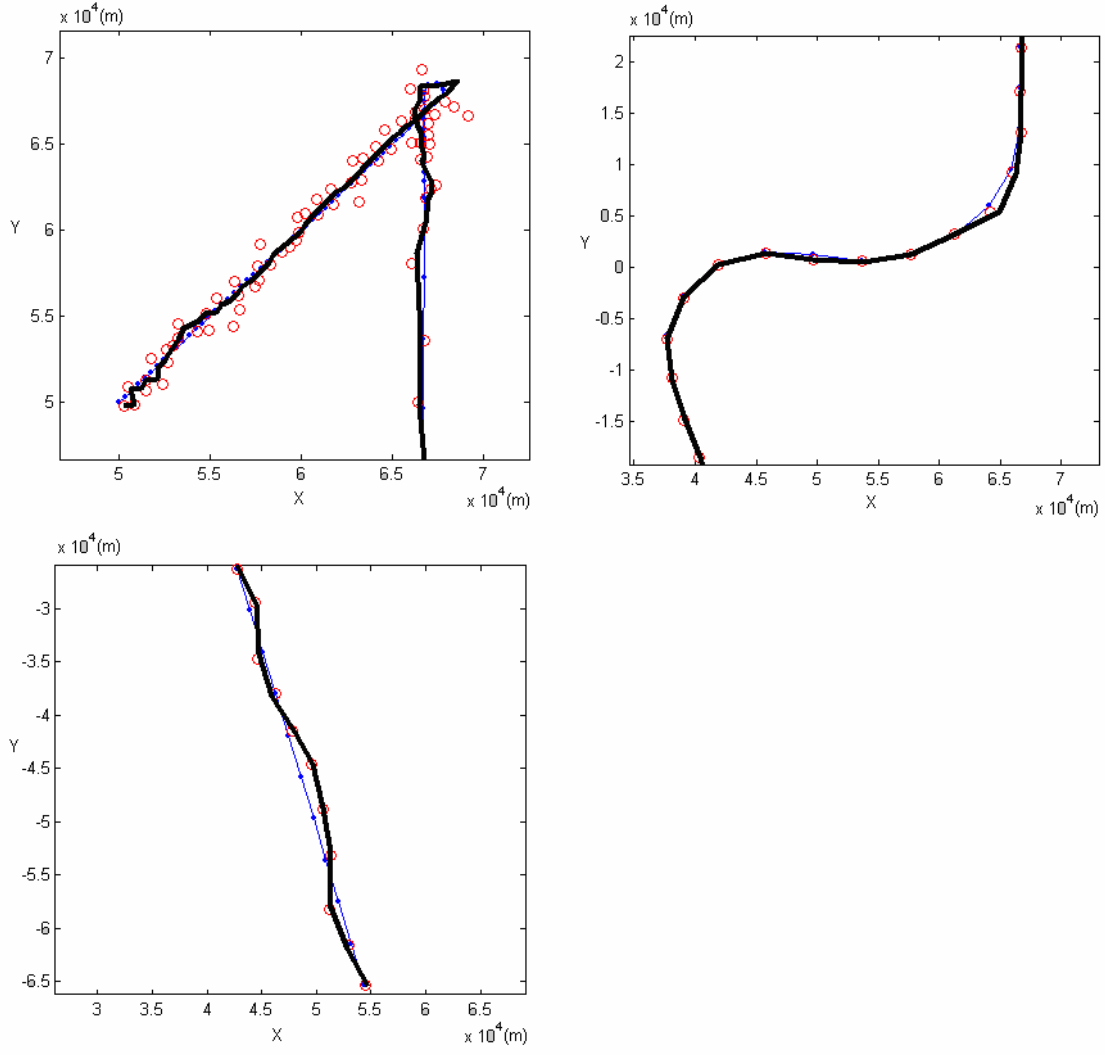
Şablon filtresinin en büyük avantajlarından biri de geçmişi düzeltme yeteneğiydi. Eğer bu filtrenin kullanılacağı sistem geçmişi düzeltmek istemiyorsa veya sadece bulunduğu andaki filtre sonuçları önemli ise filtreden geçmişi düzeltme kısmı çıkarılmalıdır ve filtrenin mevcut anda ürettiği değerler filtre çıktısı olmalıdır. Filtrenin akış diyagramı Şekil 4.27’de verilmiştir.

Gerçek zamanlı şablon filtresinde geçmişi düzeltme avantajı olmadığı için filtrenin başarısı ürettiği anlık sonuçlara bağlıdır. Bu durumda filtre hareket çeşidinde değişimlere hızlı cevap vermelidir. Bir önceki geçmiş düzeltmeli filtrede kullanılan tampon boyu daha çok veri üzerinden daha doğru sonuç üretme için daha büyük tutulmuştu. Gerçek zamanlı filtrede bu boy küçültülmelidir. Seçilebilecek en küçük kullanılabilir tampon değeri 4’tür. Daha küçüğü filtrede dengesizliklere yol açabilir. Bu filtrede ivmeli hareketler için (SAÇO ve AADO filtreleri) tampon olarak 4 kullanılacaktır.

Bu durumda filtre değerlendirme sonuçları tekrar ele alınabilir. Gerçek zamanlı şablon filtresinin başarımlı figürleri Şekil 4.28’de verilmiştir.

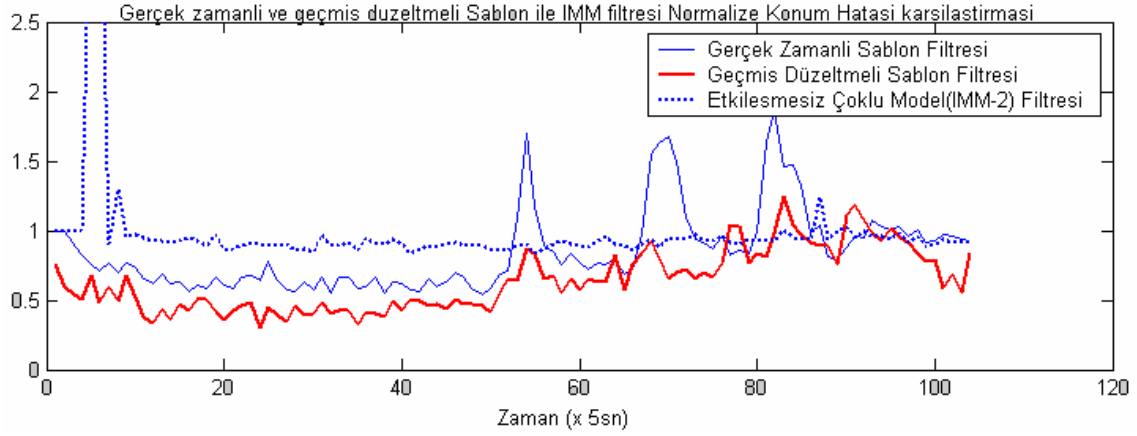


Şekil 4.27 Gerçek Zamanlı Şablon Filtresinin akış diyagramı

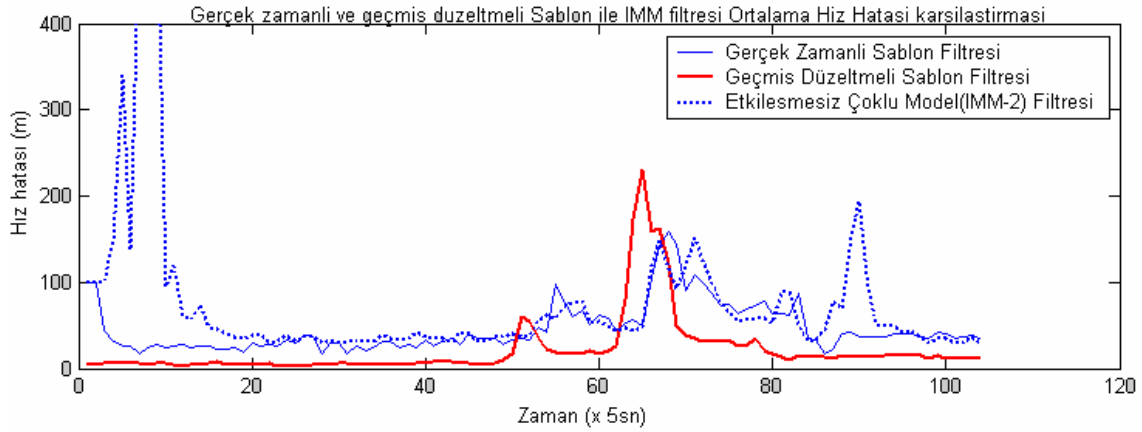


Şekil 4.28 Genel senaryo ile test edildiğinde Gerçek Zamanlı Şablon Filtresi'nin motif başarısı

Şekil 4.29 ve Şekil 4.30'da gerçek zamanlı şablon filtresinin normal şablon filtresi ve iyileştirilmiş IMM (NIMM) filtreleri ile normalize konum hatasına göre başarı karşılaştırılması yapılmaktadır



Şekil 4.29 Genel senaryo ile test edilen Gerçek Zamanlı Şablon, Geçmiş Düzeltmeli Şablon ve IMM-2(NIMM) filtrelerinin NKH karşılaştırması



Şekil 4.30 Genel senaryo ile test edilen Gerçek Zamanlı Şablon, Geçmiş Düzeltmeli Şablon ve IMM-2 filtrelerinin NKH karşılaştırması

Görüldüğü gibi gerçek zamanlı şablon filtresi geçmiş düzeltme avantajına sahip olmadan bile IMM filtresinden toplamda daha başarılıdır.

5. SONUÇLAR ve ÖNERİLER

Hava hedeflerini gözetleme sistemleri, kargaşa ortamında radarın ölçtüğü gürültülü ölçümleri kullanarak bu ölçümlerden hedefler oluşturmakta ve hedeflerin kinematik bilgilerini hesaplamaktadır. Bunun için radar ölçümlerinden hedef tespiti yapma, ilk defa gözlemlenen tespitlerden iz başlatma, tespitleri mevcut hedeflerle ilişkilendirme, ilişkilendirilmiş hedefleri filtreleme ve kinematiklerini belirleme ve iz sonlandırma gibi karmaşık bir yığın işlemler yapılır. Bahsedilen her bir işlemsel birim için ayrı detaylı çalışmalar yürütülmektedir. Bu çalışmada alt işlevsel birimlerden en karmaşık olanı yani filtreleme ve kinematik belirleme birimi üzerinde durulmuş, derinlemesine analizler yapılmıştır. Literatürde bilinen filtreler teorik olarak anlatılmış, avantaj ve dezavantajları ortaya konmuştur. Bütün filtreleri daha sağlıklı değerlendirmek için gerçek dünya koşullarına uygun en zorlayıcı manevralar yapan bir hedef benzetimi yapılmış ve yine gerçek verilere uygun olacak şekilde modellenen bir hava tarama radarı ile bu hedef tespit edilmiştir. Hedefe ait ölçümler 100 defa rasgele ölçüm gürültüleri ile alınmış ve filtreler bu ölçümlerle 100 Monte Carlo koşusuna tutulmuş ve başarımları ayrı ayrı analiz edilip karşılaştırılmıştır.

Test verisi oluşturmak için kullanılan radar modeli ve çalışma parametreleri, günümüzün savaş gemilerinde hala aktif bir şekilde kullanılan ANSPS-49 hava tarama radarına oldukça yakındır. Bu radar en iyi 5 saniyede bir gözlem alabilmektedir. Daha sık gözlem almak takibin başarımlarını artırır. Ancak mevcut koşullar düşünüldüğünde bir savaş uçağının yapacağı yüksek manevraları en fazla 30 saniye devam ettirdiği bu sebeple manevra anına ait en fazla 6 ölçüm alınabildiği görülmektedir. Daha yüksek manevralarda bu sayı daha da azdır. Manevrayı tespit edebilmek için kullanılan manevra dedektörleri en az 3 gözleme ihtiyaç duymaktadır. Bu da manevra anında teorik olarak başarılı olduğu ispat edilen filtreler için bu yeteneklerini kullanabilecekleri sadece 3 gözlem olacağı anlamına gelmektedir. Bu çalışmada özellikle filtrelerin başarımları test edilirken ve karşılaştırma yapılırken bu durum gözönüne alınmıştır.

En basit filtrelerden Alfa-Beta ve Alfa-Beta-Gama filtreleri, kullanımları kolay ve işlem gücü gereksinimleri düşük olduğu için hedef takip sistemlerinde sıkça kullanılmaktadır. Bu çalışmada Alfa-Beta ve Alfa-Beta-Gama filtrelerinin karşılaştırması yapılmış teorik olarak başarılı ve başarısız olduğu koşullar irdelenmiş, gerçek dünya koşullarına uygun benzetim ortamında başarımları ortaya konmuştur. Ayrıca Alfa-Beta ve Alfa-Beta-Gama filtrelerinin başarımlarını artırmak için bu filtreleri değişen manevra koşullarına göre parametrelerini düzenleyen bir dinamiklik getirilmiştir. Bu sayede bu filtrelerin parametreleri otomatik olarak

değişebilmektedir. İşlem gücünde fazla bir yük getirmeyen bu mekanizma ile başarıları artırılmıştır. Teorik olarak manevra anlarında daha başarılı sonuçlar ürettiği ispat edilen Alfa-Beta-Gama filtresinin yukarıda bahsedilen kısıtlamalardan dolayı (ölçüm frekansı sebebiyle manevra anına ait az ölçüm alma) aslında bütün senaryo düşünüldüğünde daha başarısız olduğu gösterilmiştir. Buna sebep olarak da manevrayı öğrenene kadar geçen sürede hedefin manevrayı bitirdiği ve düzgün doğrusal hareketine döndüğü, bu sefer de filtrenin düzgün doğrusal koşullara geri dönmek için geçen sürede hatalar oluşacağı böylece manevrayı öğrendikten sonra bu bilgiyi kullanacak zaman kalmadığı için bu avantajının aslında dezavantaja döndüğü gösterilebilir. Yapılan analizlerde 5 saniyede bir ölçüm alındığı durumlarda sırayla en az başarılı olandan en çok başarılı olana doğru Alfa-Beta-Gama, Uyarlamalı Alfa-Beta-Gama, Alfa-Beta, Uyarlamalı Alfa-Beta filtreleri gösterilebilir. Daha yüksek frekansta örneğin 1 saniyede 1 ölçüm alınabildiği durumlarda üç boyutlu filtrelerin avantajı ortaya çıkacak ve sıralama Alfa-Beta, Alfa-Beta-Gama, Uyarlamalı Alfa-Beta, Uyarlamalı Alfa-Beta-Gama şeklinde gerçekleşecektir.

Değişken parametrelili filtrelerden en iyi bilinen Kalman filtresi de bu çalışmada incelenmiştir. Kalman filtresi teorik olarak ortalama en küçük karesel hatalı (MMSE) sonuç üretmektedir. Bu açıdan sabit katsayılı filtrelerden (Alfa-Beta ve Alfa-Beta-Gama gibi) daha üstündür. Dezavantajı ise sabit katsayılı filtrelerden daha çok işlem gücü gerektirmesidir. Ayrıca Kalman filtresi, hedef hareket modeline uygun kovaryans matris bilgisine ihtiyaç duymaktadır. Bu çalışmada örnek senaryo üzerinde kovaryans matrisinin nasıl seçilebileceği ve bu seçim yapılırken nelerin gözönünde bulundurulması gerektiği anlatılmıştır. Hedef takip sisteminde tek filtre kullanılacaksa bu durumda seçilen kovaryans matris hedefin yapabileceği bütün olası manevralar gözönüne alınarak seçilmelidir. Kovaryans matris yapılabilecek en büyük manevrayı 2-3 gözlemde kapatabilecek büyüklükte seçilmelidir.

Uyarlamalı Kalman filtresi yapılan manevrayı öğrendikten sonra kovaryans matrisinde gerekli düzeltmeyi otomatik olarak yapabilmektedir. Bu çalışmada Uyarlamalı Kalman filtresine girilmemiştir çünkü yukarıdaki sebeplerden dolayı manevra öğrenilene kadar geçen sürede manevra bitirilmiş olmakta böylece bu mekanizmaların avantajlarından faydalanılamamaktadır. Bunun yerine çoklu modelleri kullanan IMM filtresi test edilmiştir. IMM filtresinde değişik manevra koşullarına uygun kovaryans matrislerine sahip birden çok filtre aynı anda çalıştırılmaktadır. Hedefin hareketine en yakın kovaryans matrisine sahip filtre en başarılı sonuç üretecek IMM filtresi de bu filtrenin sonucunu kullanacaktır. Bu filtrenin en önemli avantajı yapılan manevralara hazır filtrelerin olması böylece manevrayı

öğrenmek için zaman kaybı yaşanmamasıdır. IMM filtresinde kullanılan filtre sayısı artırıldıkça hedefin hareketine daha yakın filtreler devreye girebilecek böylece başarı artacaktır. Ancak bunun dezavantajı, filtre sayısı ile orantılı olarak artan işlemci gücü gereksinimidir. Günümüzün teknolojisinde artık bu işlemci hızlarına ulaşmak mümkündür. Bar-Shalom'un yaptığı çalışmalarda IMM filtresinin en iyi sonuç verdiği gösterilmiştir. Bu çalışmada da IMM filtresinin alt filtreleri olarak Kalman filtresi kullanılmış ve başarısı diğer filtrelerle karşılaştırılmıştır. Bu çalışmada IMM filtresinin iç mekanizması derinlemesine gösterilmiştir. Ayrıca filtrede iyileştirmeler de yapılmıştır. Literatürde kullanılan parametrelerin dışına çıkılarak gerçek uygulamada daha başarılı olacak parametreler önerilmiş ve test edilerek başarıları ortaya konmuştur. Bununla birlikte IMM filtresinde etkileşim kısmı çıkarılarak yeni bir filtre IMM-2 veya NIMM (Non-interacting Multiple Model) filtreleri oluşturulmuş ve bu filtrelerin başarısı test edilmiştir. NIMM filtresi üretilen senaryonun bütünü düşünüldüğünde daha başarılı sonuçlar vermiştir.

Son olarak bu çalışmada yepyeni bir filtre, Şablon filtresi, tasarlanmış ve tanıtılmıştır. Yığın işleme tekniğine dayalı bu filtre, geleneksel filtrelerden farklı olarak sadece son gözleme göre değil her ölçümde geçmiş ölçümleri de hesaba katarak kestirim yapmaktadır. Bu teknik insan gözüyle yapılan karar verme mekanizmasıyla benzerlik gösterir. Operatör gözle takip ettiği hedefin bir sonraki turda bulunacağı konumu geçmiş konumlara ve hedefin yönelimine bakarak yapar. Dolayısıyla geçmişteki gözlemlerden faydalanır. Şablon filtresi de geçmiş gözlemler ile mevcut gözlemi bir şablona oturtmaya çalışır. Kullanılan üç farklı şablon, düzgün doğrusal hareket şablonu, bir doğrudan hızlanan hareket şablonu ve dönme hareketi şablonu ile hangi tür hareket yapıldığını belirlemektedir. Diğer filtrelerden farklı olarak şablon filtresi geçmişini düzeltebilmektedir. Şablon filtresi, yapılan testler sonucunda mevcut bilinen bütün filtrelerden çok daha başarılı sonuçlar vermiştir. Kovaryans matrisi gerektirmemesi, iyileştirmelere açık olması, ürettiği konum ve hız değerlerinde zigzaglar olmaması, ürettiği hız, konum ve ivme değerleri açısından en az hatalı sonuç üretmesi, yapılan hareketi en iyi modelleyen filtre olması, geçmişini düzeltebilmesi açısından diğer filtrelerle göre oldukça fazla üstünlük sağlamıştır. Tek dezavantajı olarak gerektirdiği işlem gücü gösterilebilir.

Geçmişini düzeltme yeteneğinin kullanılmadığı durumlarda kullanılabilecek Gerçek Zamanlı Şablon filtresi de bu çalışma kapsamında tanıtılmıştır. Geçmişini düzeltme kısmı çıkarılmış filtre sadece alınan son ölçümü şablon üzerinde düzeltebilmektedir. Bu durumda bile diğer filtrelerden daha başarılı sonuçlar üretmiştir.

İleride yapılacak alıřmalar kapsamında řablon filtresinde kullanılan btn řablonların boyunu otomatik ayarlayacak bir mekanizma tasarlanabilir. Bu sayede řablon filtresinin bařarısı daha da artırılabilir.

Radar lmnde hibir gzlemin kaırılmadıėı ve kargařa ortamında hatalı lmlerin yapılmadıėı kabul edilmiřti. İleriye ynelik olarak bu konularda gereėe yakın bir modelleme yapılarak kargařa ortamı benzetilebilir ve radarın belli oranda gzlem kaırması saėlanarak gerek kořullara daha yakın bir test ortamı oluřturulabilir. Bu test ortamında filtrelerin karřılařtırması ve bařarımları daha saėlıklı analiz edilebilir.

KAYNAKLAR

- Averbuch, A., Itzikowitz, S. ve Kapon T., (1991), "Radar Target Tracking Viterbi versus IMM", IEEE Transactions on Aerospace and Electronic Systems, AES-27, May 1991, pp 550-562.
- Bar-Shalom, Y. ve Li, X. R., (1993), "Estimation and Tracking: Principles, Techniques and Software", Artech House,
- Bar-Shalom, Y., Chang, K.C., ve Blom, H.A.P., (1989), "Tracking a Maneuvering Target Using Input Estimation Versus The Interacting Multiple Model algorithm", IEEE Trans. Aerospace and Electronic Systems, AES-25, March 1989, pp 296-300,
- Blackman, S.S., (1986), "Multiple-Target Tracking with Radar Applications", Dedham, MA: Artech House,
- Efe, M., (1998), "Adaptive Approaches to Maneuvering Target Tracking", Doctor of Philosophy, The University of Sussex, UK.
- Grewal, M.S. ve Andrews, A.P., (1993), "Kalman Filtering: Theory and Practice", Prentice Hall,
- Kalata, P.R., (1984), "The Tracking Index: A Generalized Parameter for α - β and α - β - γ Target Trackers", IEEE Transaction on Aerospace and Electronic systems, AES-20, March 1984, pp. 174-182,
- Kalman, R. E., (1960), "A New Approach to Linear Filtering and Prediction Problems", Journal of Basic Engineering, Vol 82, No 1, March, pp. 35-46,
- Lana, A., (2001), "Kalman Filtresi Ve Olasılıksal Veri İlişkilendirme Yöntemlerini Kullanan Çoklu Hedef İzleme Algoritmaları", Master of Science, İstanbul Teknik Üniversitesi, İstanbul.
- Li, X. R. ve Bar-Shalom, Y., (1993), "Design of an Interacting Multiple Model Algorithm For Air Traffic Control Tracking", IEEE Transactions on Control Systems Technologies, special issue on Air Traffic Control, Vol 1, No 3, September 1993, pp 186-194,

ÖZGEÇMİŞ

Doğum tarihi 18.06.1979

Doğum yeri Mersin

Lise 1993-1996 Mersin Fen Lisesi

Lisans 1996-2001 Boğaziçi Üniversitesi Mühendislik Fak.
Elektrik Elektronik Mühendisliği Bölümü

Yüksek Lisans 2005-2007 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektronik ve Haberleşme Müh. Anabilim Dalı,
Haberleşme Programı

Çalıştığı kurumlar

2002-2005 Tübitak MAM Bilişim Teknolojileri Enstitüsü

2005-Devam ediyor NETAŞ