

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**HAREKETLİ BAZ İSTASYONU ARACILIĞI İLE PARÇACIK FİLTRESİ
KULLANILARAK ALINAN SİNYAL GÜÇ GÖSTERGESİ TABANLI
KONUMLAMA**

YÜKSEK LİSANS TEZİ

Şerife Senem KARAMAN

Elektrik Elektronik Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Tolga GİRİCİ

Anabilim Dalı : Herhangi Mühendislik, Bilim

Programı : Herhangi Program

TEMMUZ 2020

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

Şerife Senem KARAMAN

İMZA



ÖZET

Yüksek Lisans Tezi

HAREKETLİ BAZ İSTASYONLARI ARACILIĞI İLE PARÇACIK FİLTRESİ
KULLANILARAK ALINAN SİNYAL GÜÇ GÖSTERGESİ TABANLI

KONUMLAMA

Şerife Senem KARAMAN

TOBB Ekonomi ve Teknoloji Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Doç.Dr. Tolga GİRİCİ

Tarih: Temmuz 2020

İnsansız hava araçlarının (İHA) baz istasyonu olarak kullanımı son yılların popüler bir araştırma konusudur. İHA baz istasyonlarının konum ve yüksekliklerinin kontrol edilebilmesi büyük avantaj sağlamaktadır. İyi bir konumlandırma için kullanıcı konumlarının bilinmesi gerekmektedir. Özellikle kullanıcıların hareketli olduğu ve GPS benzeri bir konumlandırma aygıtı bulundurmadığı durumlar zorluk yaratmaktadır.

Bu çalışmada hareketli baz istasyonu kullanılarak yerde sabit kullanıcı ve hareketli kullanıcının Parçacık Filtresi yardımı ile sabit kullanıcı için konum kestirimi ve hareketli kullanıcı için ise takibi gerçekleştirilmiştir. Parçacık Filtresinde yer alan parametrelerin değiştirilmesi ile farklı senaryolar oluşturularak simülasyonları koşulmuş ve kestirimi iyileştiren parametreler araştırılmıştır. Optimal Hareketli Baz İstasyonu Yüksekliği, yeniden örnekleme yöntemi, parçacık sayısı ve Hareketli Baz İstasyonunun harcadığı toplam enerji değeri getiri götürü analizinde kullanılmıştır.

Anahtar Kelimeler: Zamanlama analizi, İstatiksel modelleme, Benzetim.

ABSTRACT

Master of Science

USE OF PARTICLE FILTERING IN RECEIVED SIGNAL STRENGTH
INDICATOR BASED
LOCALIZATION BY DRONE BASE STATIONS

Serife Senem Karaman

TOBB University of Economics and Technology
Institute of Natural and Applied Sciences
Master of Science Programme

Supervisor: Assoc. Prof. Tolga GİRİCİ

Date: July 2020

Drone Base Stations (DBSs) provide flexible deployment and line-of-sight coverage opportunities, which led to many use cases, such as broadband Internet, military, surveillance, agriculture etc. DBSs can optimize and adapt their positions based on user location information. Especially in GPS-denied tactical scenarios ground user location estimation is an important problem. In this work we investigate particle filter as a method of user position estimation. We utilize the recently proposed air to ground pathloss model for RSSI-based location estimation. We investigate different DBS trajectories and various resampling methods. Also, we investigate the tracking performance of the filter by comparing different simulation results including the total energy needed by DBS to track one target.

Keywords: Localization, drone base stations, UAV estimation, particle filter, tracking, simulation.

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren sayın hocam Tolga GİRİCİ'ye, kıymetli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Elektrik Elektronik Mühendislięi Bölümü öğretim üyelerine, gücünü her zaman yanımda hissedeyeğim babama, bu süreçte desteklerini benden esirgemeyen aileme çok teşekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ŞEKİL LİSTESİ	viii
ÇİZELGE LİSTESİ	ix
KISALTMALAR	x
SEMBOL LİSTESİ	xi
RESİM LİSTESİ	xiii
1 GİRİŞ	1
1.1 Tezin Amacı.....	1
1.2 Literatür Araştırması	1
1.3 Alınan Sinyal Güç Göstergesi.....	2
1.4 Parçacık Filtresi.....	3
1.4.1 Parçacık Filtresi Algoritması.....	5
1.4.2 Parçacık Filtresinin Matematiksel Türetimi.....	7
2 MATEMATİKSEL MODEL VE SAYISAL YÖNTEM	9
2.1 Matematiksel Model.....	9
2.2 Sayısal Yöntem	11
2.2.1 Olasılıksal Yol Kayıp Modeli	11
2.2.2 Parçacık Filtresi Algoritması.....	11
2.2.3 Yeniden Örnekleme Yöntemleri	13
2.2.3.1 Çok Terimli Örnekleme Yöntemi.....	13
2.2.3.2 Arta Kalan Örnekleme Yöntemi.....	15
2.2.3.3 Kademeli Yeniden Örnekleme Yöntemi.....	16
2.2.3.4 Sistematik Yeniden Örnekleme Yöntemi.....	17
2.2.4 Hareketli Baz İstasyonu Enerji Modeli.....	18
2.2.5 Çevresel Parametreler	20
3 PARÇACIK FİLTRESİ KULLANILARAK KONUM KESTİRİMİ	23
3.1 Benzetim Çalışmaları	23
3.2 Yerde Sabit Kullanıcının Konum Kestirimi	23
3.3 Yerde Hareketli Kullanıcının Konum Kestirimi	31
3.3.1 Takip Algoritması.....	32
3.3.1.1 Parçacık sayısının takip performansı ve toplam HBİ enerjisine etkisi	36
3.3.1.2 Dairesel hareket çapının takip performansı ve toplam HBİ enerjisine etkisi	37
3.3.1.3 Kullanıcı Zaman Değerinin Takip Performansı ve Toplam HBİ Enerjisine Etkisi	41
3.3.1.4 HBİ İleri Yönlü Hızının ve Toplam HBİ Enerjisine Etkisi.....	43
4 SONUÇ VE ÖNERİLER	44
KAYNAKLAR	46
ÖZGEÇMİŞ	50

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1: ASGG-Konum arası ilişki.	3
Şekil 1.2: Parçacık filtresi kullanılarak elde edilmiş “parçacık” gösterimi.....	5
Şekil 3.1: $T = 9$ değeri için farklı yörünge tipleri (a) grid, (b) dairesel, (c) dairesel ile verilen yörüngeye eş değer enerjili grid.	25
Şekil 3.2: Yeniden örnekleme tipi ve HBİ ölçüm alma sayısının ortalama hataya etkisi.	27
Şekil 3.3: Ölçüm alma sayısı ve yeniden örnekleme yöntemlerinin maksimum konum kestirim hatasına etkisi.	28
Şekil 3.4: HBİ yörünge değişimi ve ölçüm sayısı için HBİ toplam enerji (Joule) değişimi.	29
Şekil 3.5: HBİ yörünge değişimi ve ölçüm sayısı için ortalama hata değişimi.	30
Şekil 3.6: HBİ izdüşüm ve kullanıcı hareketi (a) Parçacık Sayısı 1000, HBİ Yüksekliği 500m, Kullanıcı Toplam Zamanı 500, Hata Değeri 85 (b) Parçacık Sayısı 1000, HBİ Yüksekliği 500, Kullanıcı Toplam Zamanı 500, Hata Değeri 148 (c) Parçacık Sayısı 1000, HBİ Yüksekliği 800m, Kullanıcı Toplam Zamanı 500, Hata Değeri 225 (d) Parçacık Sayısı 1000, HBİ Yüksekliği 100, Kullanıcı Toplam Zamanı 500, Hata Değeri 825 değerlerine sahip grafiklerdir.....	35
Şekil 3.7: Parçacık Sayısının Ortalama Hata ve HBİ Enerjisine ($\times 1000$) Etkisi.....	36
Şekil 3.8: Kullanıcı hareketine ait yarıçap değerinin ortalama hata ve HBİ enerjisine ($\times 1000$) etkisi (h değeri 500 m).	38
Şekil 3.9: Kullanıcı hareketine ait yarıçap değerinin ortalama hata ve HBİ enerjisine ($\times 1000$) etkisi (h değeri 1000 m).	40
Şekil 3.10: Yarıçapın (r) 1000 olduğu dairesel hareketin tamamlanma süresi (bitişzamanı) (a) 10, (b) 100 için	41
Şekil 3.11: Kullanıcıya ait Dairesel Hareketin Tamamlanma Zamanının Ortalama Hata ve HBİ Enerjisine ($\times 1000$) Etkisi.....	42
Şekil 3.12: HBİ İleri Yönlü Hız Değişiminin HBİ Enerjisine ($\times 1000$) Etkisi	43

ÇİZELGE LİSTESİ

Sayfa

Çizelge 1.1: Parçacık filtresi algoritması.	6
Çizelge 2.1: Tez kapsamında oluşturulan PF algoritması.	11
Çizelge 2.2: Çok terimli örnekleme kaynak kodu (MATLAB).	14
Çizelge 2.3: Arta kalan örnekleme kaynak kodu (MATLAB).	15
Çizelge 2.4: Kademeli yeniden örnekleme kaynak kodu (MATLAB).	16
Çizelge 2.5: Sistemik yeniden örnekleme kaynak kodu (MATLAB).	17
Çizelge 2.6: Enerji modeli parametreleri.	20
Çizelge 2.7: Şehir modeli çevresel parametreleri.	21
Çizelge 2.8: RF modeli çevresel parametreleri	22
Çizelge 3.1: Kanal ve yol kaybı değişkenleri (Şehir dışı çevre koşulu).	24
Çizelge 3.2: Ortalama hata ile parçacık sayısı (L), HBİ yüksekliği (h), HBİ yörüngesine ait yarıçap değeri.	25
Çizelge 3.3: Yeniden Örnekleme Tiplerine ait Hesaplama Zamanı	26
Çizelge 3.4: Takip algoritması değişkenleri (Şehir Dışı Çevre Koşulu).	31
Çizelge 3.5: Takip algoritması.	32

KISALTMALAR

ASGG	: Alınan Sinyal Güç Göstergesi
DBİ	: Dron Baz İstasyonu
GPS	: Global Positioning System
HBİ	: Hareketli Baz İstasyonu
ITU	: International Telecommunication Union
İHA	: İnsansız Hava Aracı
KKS	: Küresel Konumlama Sistemi
UVA	: Unmanned Aerial Vehicle
PF	: Parçacık Filtresi
RF	: Radyo Frekans
RSSI	: Received Signal Strenght Indicator

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

Açıklama

a	Çevresel parametre
A_d	DBİ aracının alanı
b	Çevresel parametre
bel	İnanç fonksiyonu
c	Ses hızı
C_{ds}	DBİ'nin sürtünme katsayısına bağlı sabit
d^{ud}	Kullanıcı ile DBİ pozisyonu arasındaki gerçek uzaklık
D_{max}	Kullanıcının içinde aldığı kare haritanın boyutu
E_T	DBİ toplam enerji değeri
f_c	Taşıyıcı frekans
g	Yer çekimi ivmesi
h^d	DBİ düşey ekseninde konumu
k_0	DBİ bıçak boyut sabiti
l_T	İki ardışık ölçüm noktası arasındaki uzaklık
L	Parçacık sayısı
m	DBİ ağırlık değeri
M	Toplamları gerçekleştirilecek ağırlıkların sayısı
n	Birbiçimli dağılım
N	Parçacık ağırlıkları sayısı
P_d	DBİ toplam güç değeri
p_{Los}	Görüş hattı içinde olma olasılığı
p_{NLos}	Görüş hattı dışında olma olasılığı
PL_{Los}	Görüş hattı içinde yol kaybı değeri
PL_{NLos}	Görüş hattı dışında yol kaybı değeri
r^{ud}	Kullanıcı ile DBİ arasındaki yatay uzaklık
R	DBİ yörüngesine ait yarıçap değeri
t	Zaman
T	DBİ ölçüm noktası sayısı
u	Kontrol fonksiyonu
v	DBİ ileri yönlü hız değeri
v_t	DBİ rotor kanadının doğrusal uç hızı
v_{ind}	DBİ ortalama indüklenen hız değeri

x	Durum fonksiyonu
x^d	DBİ –x ekseninde konumu
x^u	Kullanıcı–x ekseninde konumu
X	İnanç fonksiyonunu temsil eden parçacık seti
y^d	DBİ –y ekseninde konumu
y^u	Kullanıcı –y ekseninde konumu
z	Ölçüm fonksiyonu
Q	Ağırlıklar toplamı değeri
α	Yerleşik arazi alanının toplam arazi alanına oranı
β	Birim alan başına ortalama bina sayısı
γ	Bina yükseklik dağılımı ölçek parametresi
η_{LoS}	Görüş hattı içi yol kaybı değeri
η_{NLoS}	Görüş hattı dışı yol kaybı değeri
μ_{LoS}	Görüş hattı içi yol kaybı ortalaması
μ_{NLoS}	Görüş hattı dışı yol kaybı ortalaması
σ_{LoS}	Görüş hattı içi yol kaybı standart sapması
σ_{NLoS}	Görüş hattı dışı yol kaybı standart sapması
θ^{ud}	Kullanıcı ile DBİ pozisyonu arasındaki yükseliş açısı
ρ	Hava yoğunluğu

RESİM LİSTESİ

Sayfa



1 GİRİŞ

İnsansız Hava Aracı (İHA) olacak şekilde kullanılan Dron Baz İstasyonlarındaki (DBİ) gelişmeler son 10-15 yılda büyük ölçüde artmıştır. Yazılım, malzeme ve çip teknolojisindeki ilerlemeler DBİ kullanım kapasitesini genişletmiştir. Nesnelerin İnterneti, Güvenlik, Askeri ve Genişalan İnternet Erişim gibi alanlarda DBİ kullanımı görülmektedir. DBİ kullanımının temel kısıtı güç gereksinimidir.

Dron Baz İstasyonu (DBİ) kullanımı haberleşme mimarisine mobilizasyon yeteneği kazandırmakta ve böylece temel bir avantaj ortaya konulmaktadır. İşlem kapasitesi veya kapsama alanını maksimuma çıkaracak şekilde DBİ konumları optimize edilebilmektedir. DBİ kullanımının bir diğer avantajı da yükseklikleri sebebi ile kendisi ve kullanıcı arasında görüş hattı sağlama olasılığını arttırmasıdır. Bu avantajlar, İHA'ların kablosuz haberleşmede baz istasyonu olarak kullanılması konusunda yapılan çalışmaların popülaritesini son beş yılda arttırmıştır.

1.1 Tezin Amacı

DBİ'lerin kablosuz haberleşme mimarisinde kullanılması kapsamında gerçekleştirilen çalışmalar [1] ile verilen, yerden havaya kanal modelini temel almaktadır. Tezin amacı, parçacık filtresi kullanımının [1] ile verilen kanal modelinde uygulanarak sabit kullanıcı yer kesitimi ile hareketli kullanıcı için takip gerçekleştirilmesidir. Bunun için, optimal hareketli baz istasyonu yüksekliği, parçacık sayısı, iterasyon sayısı ve yörüngenin kestirim performansına etkisi incelenmiş, farklı örnekleme metodlarının etkisi de çalışılarak parçacık filtresinin kesitirim performansı arttırılmıştır.

1.2 LiteratürAraştırması

DBİ'lerin kablosuz haberleşme mimarisinde kullanılması kapsamında gerçekleştirilen çalışmalar [1] ile verilen, yerden havaya kanal modelini temel almaktadır. Aynı kapsamda yapılan diğer çalışmalarda ise maksimum kapsama alanı için optimal İHA yüksekliğinin incelenmesi [2], İHA'nın üç boyutlu olacak şekilde konumlandırılması

[3,4], maksimum veri hızı için gereken İHA sayısı ve konumuna yönelik optimizasyon çalışmaları [5] gerçekleştirilmiştir.

Yukarıda verilen çalışmaların hepsi yerdeki kullanıcıya ait konum bilgisinin bilindiğini varsaymaktadır. Hem İHA hem de yerdeki kullanıcılar üzerinde Küresel Konumlama Sistemi (KKS) bulunduğu durumlar için bu varsayım gerçekçi olmaktadır. Fakat, KKS'nin engellenme olasılığının bulunduğu askeri uygulamalarda veya yerdeki kullanıcı üstünde KKS sinyali olmadığı durumlarda ilgili kullanıcıya ait konumun belirlenmesi ihtiyacı doğmaktadır. Bu tip durumlar için, Alınan Sinyal Güç Göstergesi (ASGG) tabanlı konum kestirimi alternatif bir çözüm sunmaktadır. Hava- yer arası kanal modeli [1] ile yol kaybı hesaplanarak yerdeki kullanıcının yer kestirimi gerçekleştirilebilmektedir. [6] ve [7] ile verilen çalışmalarda ilk defa yol kaybının yer kestiriminde kullanılması ile İHA baz istasyonlarının konumları hesaplanmıştır.

RSSI-tabanlı konum kestirimi daha önce kapalı mekan kablosuz haberleşme ağları için düşünülmüştür [8]. [9] ile verilen çalışmada ise İHA kullanılacak şekilde kullanım genişletilmiş ve parçacık filtresi metodu yerdeki kullanıcının yerinin kestirimi için kullanılmış, yol kaybı modeli farklı ve daha basit olacak şekilde ele alınmıştır.

Literatüre 1993 yılında girmiş olan [10] parçacık filtresi doğrusal olmayan ve Gauss dağılımına uymayan problemlerin çözümünde kullanılan popüler bir sayısal kestirim metodudur. Parametrik olmayan bu filtre tekniğinde, ardıl durum sabit bir fonksiyon formuna dayanmamaktadır ve ardıl duruma ait olasılık yoğunluğu parçacıklar ile temsil edilmektedir. Parçacık filtresi metodunda [11], ardıl durum olasılık yoğunluğunun ardışık Monte Carlo kestirimi gerçekleştirilir. Bu nedenle Parçacık Filtresine literatürde Ardışık Monte Carlo da denilmektedir.

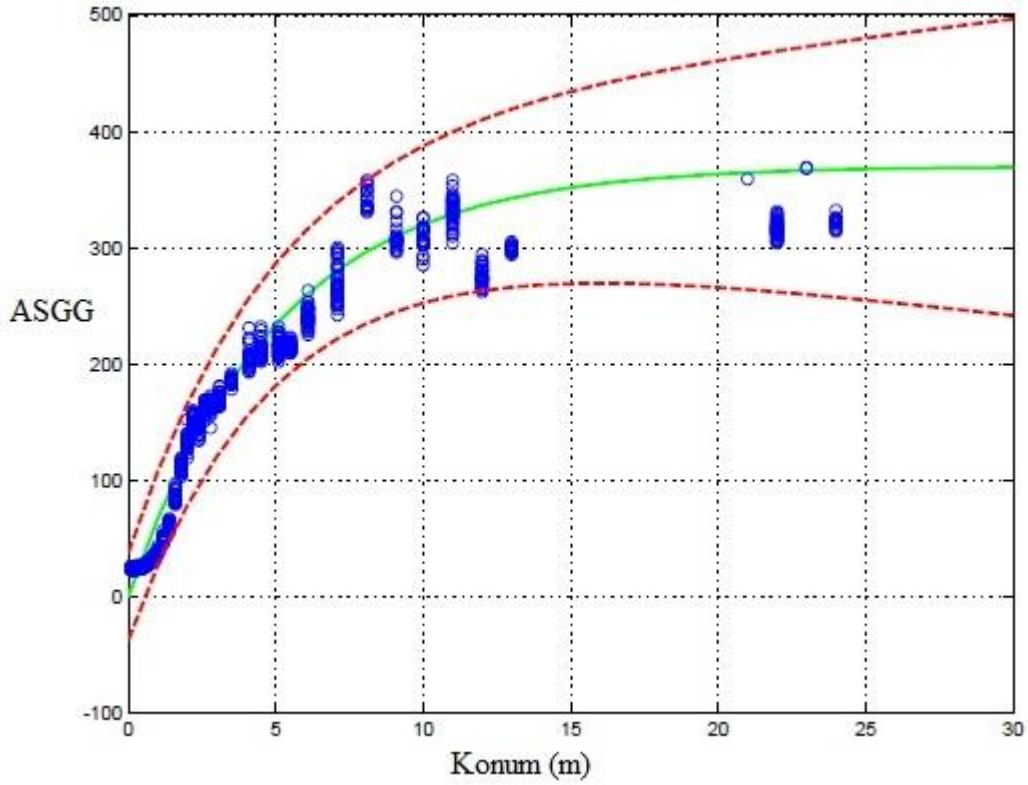
PF kullanmadan sadece ASGG tabanlı konumlama çalışmaları da literatürde bulunabilen çalışmalardır. [19] ile verilen çalışmada ASGG ve Pekiştirmeli Öğrenme (İng. *Reinforcement Learning*) yöntemleri kullanılarak İHA'nın otonom olarak yerdeki objelerin konumunu bulması sağlanmıştır.

1.3 Alınan Sinyal Güç Göstergesi

Konumlama problemi için literatürde bir çok başarılı çözüm yöntemi bulunmaktadır [12]. Bu yöntemler temelde uzaklığa bağlı konumlama ve uzaklığa bağlı olmayan

konumlama olmak üzere ikiye ayrılmaktadır. ASGG ölçümüne yönelik pratik uygulamaları içeren ayrıntılı bilgi [20] ve [21]'de ele alınmaktadır.

Uzaklığa bağlı konumlama algoritmaları daha fazla donanım ihtiyacı doğururken daha yüksek konum hassasiyeti vermektedir. Uzaklığa bağlı olmayan konumlama algoritmalarında ise donanım ihtiyacı daha azdır fakat daha düşük konumlama hassasiyetleri vardır. Alınan sinyal gücüne bağlı konumlama, uzaklığa bağlı bir konumlama yöntemidir. ASGG ile konum arası ilişki [9] numaralı kaynakta Şekil 1.1'de görüldüğü gibi ele alınmıştır.



Şekil 1.1: ASGG-Konum arası ilişki.

Şekil 1.1 ile verilen ASGG-Konum arasındaki ilişki formülize edilmiş ve belirli bir ASGG değerine sahip kullanıcının konum kesitiriminde kullanılmıştır.

1.4 Parçacık Filtresi

Parçacık Filtresi, parametrik olmayan filtreler sınıfında yer alır. Bu filtre türünde sonraki durum, her biri durum uzayına yaklaşık olarak karşılık gelen sonlu sayıda örnekler ile temsil edilmektedir. Örnekler rastgele olacak şekilde önceki dağılımdan

türetilmektedir. Durum uzayı tahmin kalitesi, sonraki dağılımı temsil etmek için kullanılan parçacık sayısına bağlıdır.

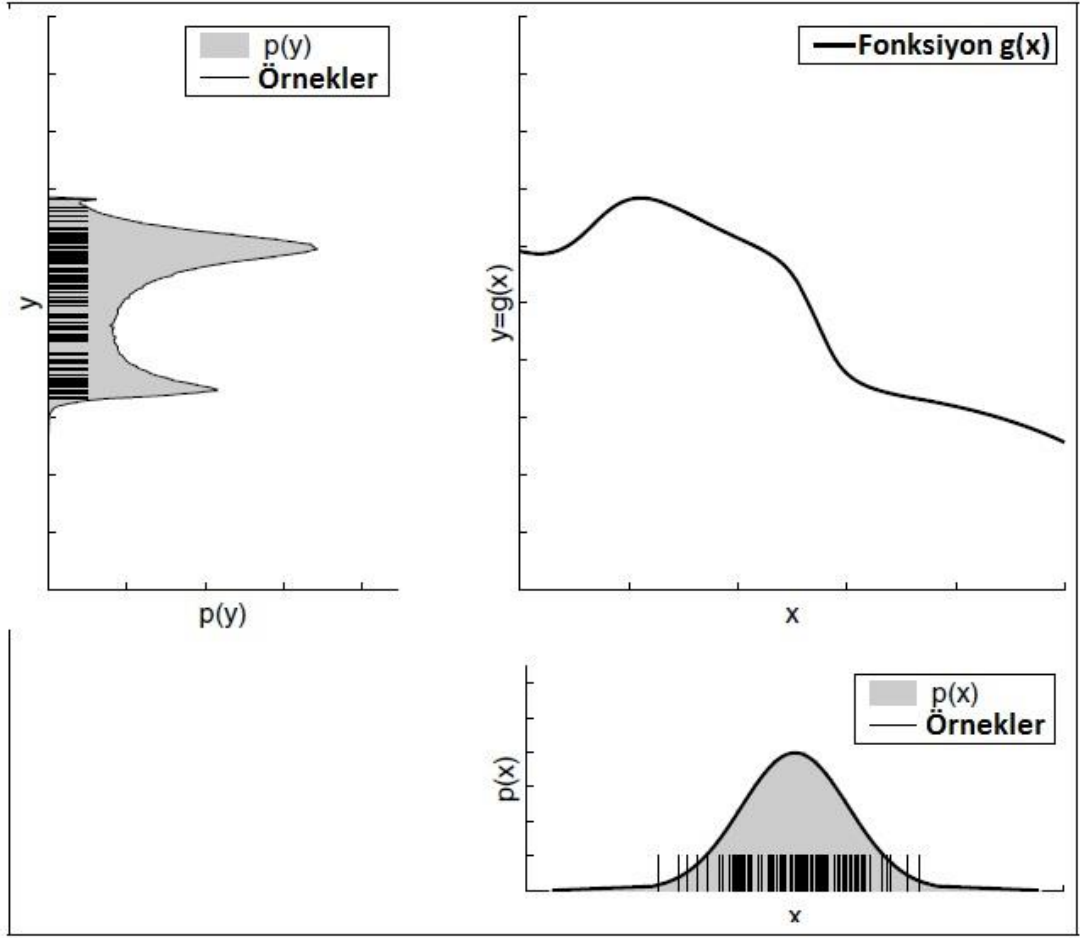
Parçacık filtresinin daha iyi anlaşılabilmesi için zaman içinde değişmeyen ikili Bayes filtresi tanımlamak gerekmektedir. Robotikte bazı problemler zaman içinde değişmeyen ikili durumdaki tahmin problemleri olarak formüle edilir. Bu problemler ikili Bayes filtresi olarak adlandırılmaktadır. Bir robot, ortamdaki sabit ikili durumu, bir dizi sensör ölçümünden tahmin ederse algılama problemleri ortaya çıkabilir. Örneğin, bir robot, kapı durumunun algılama sırasında değişmediği bir durumda, kapının açık veya kapalı olup olmadığını bilmek isteyebilir. Bu durumdaki tahmin problemleri Bayes filtresi ile formülize edilebilmektedir.

Filtre tarafından ölçülecek olan durum statik olduğunda, Eşitlik (1.1) ile verildiği gibi inanç fonksiyonunun dağılımı yalnızca sensör verilerinden alınan ölçümlerin bir fonksiyonu olmaktadır:

$$bel_t(x) = p(x|z_{1:t}, u_{1:t}) = p(x|z_{1:t}) \quad (1.1)$$

Durum, x veya $-x$ ile belirtilen iki olası değerden seçilir. Durum x belirtilirken zaman belirteci olan t 'nin kullanmaması, durumun zamanla değişmediğini yansıtmaktadır.

Sürekli fonksiyonların örnekleme tabanlı temsil edilmesinin avantajlarından biri rastgele değişkenlerin doğrusal olmayan dönüşümlerini modelleyebilmesidir. Şekil 1.1'de sağ alt grafikte Gauss rastgele değişkeni, X 'ten türetilmiş örnekler görülmektedir. Sağ üstte yer alan ve doğrusal olmayan $g(x)$ fonksiyonundan örnekler seçilmiştir. Sonuçta elde edilen örnekler ise rastgele değişken Y 'ye göre dağıtılmıştır.



Şekil 1.2: Parçacık filtresi kullanılarak elde edilmiş “parçacık” gösterimi.

Parçacık filtresinde, sonraki dağılımın örneklerine parçacıklar denilmekte ve aşağıda Eşitlik (1.1) ile ifade edilmektedir.

$$X_t := x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]} \quad (1.1)$$

Her parçacık $x_t^{[l]}$ ($1 \leq l \leq L$) durumun t zamanındaki örneğidir. L değeri X_t parçacık setinde yer alan parçacık sayısıdır. Pratikte, L parçacık sayısı yüksek bir değer almaktadır, örneğin $L = 1000$ gibi. Bazı uygulamalarda L değeri zamanın t bir fonksiyonudur.

1.4.1 Parçacık Filtresi Algoritması

Diğer tüm Bayes filtresi algoritmalarında yer aldığı gibi inanç fonksiyonu $bel(x_t)$, bir önceki zaman inanç fonksiyonundan $bel(x_{t-1})$ türetilmektedir. İnanç fonksiyonu $bel(x_t)$, X_t parçacık seti ile temsil edildiği için, parçacık filtresinde X_t parçacık seti bir önceki zamana ait parçacık setinden X_{t-1} türetilir. Parçacık filtresine ait genel

matematiksel model aşağıda Çizelge 1.1 içerisinde yer alan algoritma ile verilmektedir.

Çizelge 1.1: Parçacık filtresi algoritması.

```

1:   Algoritma Parçacık_filtresi ( $\mathbf{X}_{t-1}, \mathbf{u}_t, \mathbf{z}_t$ ):
2:        $\overline{\mathbf{X}}_t = \mathbf{X}_t = \emptyset$ 
3:       for  $l = 1$  to  $L$  do
4:           örnekle  $x_t^{[l]} \sim p(x_t | \mathbf{u}_t, x_{t-1}^{[l]})$ 
5:            $w_t^{[l]} = p(z_t | x_t^{[l]})$ 
6:            $\overline{\mathbf{X}}_t = \mathbf{X}_t + \langle x_t^{[l]}, w_t^{[l]} \rangle$ 
7:       endfor
8:       for  $l = 1$  to  $L$  do
9:           draw  $i$  with probability  $\propto w_t^{[i]}$ 
10:          ekle  $x_t^{[i]}$  to  $\mathbf{X}_t$ 
11:       endfor
12:       return  $\mathbf{X}_t$ 

```

Parçacık filtresi algoritmasının girdisinde \mathbf{X}_{t-1} ile belirtilen bir önceki zamana ait parçacık seti, \mathbf{u}_t ile belirtilen kontrol ve \mathbf{z}_t ile belirtilen ölçüm değerleri yer almaktadır. Algoritma ilk olarak inancı $\overline{bel}(x_t)$ temsil eden geçici bir parçacık seti $\overline{\mathbf{X}}_t$ oluşturmaktadır. Bunu, girdi parçacık setini \mathbf{X}_{t-1} oluşturan her parçacığı $x_{t-1}^{[l]}$ sistematik olarak işleyerek yapar. Daha sonra, bu parçacıkları, ardıl dağılıma $bel(x_t)$ yaklaşan \mathbf{X}_t kümesine dönüştürür.

Algoritmanın dördüncü satırında, $x_{t-1}^{[l]}$ parçacığına ve kontrol \mathbf{u}_t 'ye bağlı olarak t süresi için varsayımsal bir durum $x_t^{[l]}$ meydana getirilir. Elde edilen örnek, \mathbf{X}_{t-1} setinin içinde yer alan l .inci parçacığı temsil ettiğinden, l indeksi ile gösterilir. Bu adım, durum geçiş dağılımından $p(x_t | \mathbf{u}_t, x_{t-1})$ örnekleme içerir. Bu adımı uygulamak için, durum geçiş dağılımından örnek alabilmek gerekir. L iterasyondan sonra elde edilen parçacıklar kümesi, filtrenin $\overline{bel}(x_t)$ gösterimidir.

Algoritmanın beşinci satırında, her parçacık $x_t^{[l]}$ için $w_t^{[l]}$ ile belirtilen önem faktörü (önem ağırlığı veya ağırlık değeri) hesaplanır. Önem ağırlık faktörleri z_t ölçümünü parçacık setine dahil etmek için kullanılır. Dolayısıyla önem ağırlığı, $w_t^{[l]} = p(z_t | x_t^{[l]})$ ile verilen, parçacığın $x_t^{[l]}$ altındaki ölçüm z_t olasılığıdır. Bir parçacığın ağırlığını $w_t^{[l]}$ olarak yorumlarsak, ağırlıklı parçacıklar kümesi (yaklaşık olarak) Bayes filtresi ardılına $bel(x_t)$ oluşturur.

Parçacık filtresi algoritmasının en önemli bölümü, sekizden onbire kadar olan satırlar arasında gerçekleşen işlemlerden oluşmaktadır. Bu satırlarda *yeniden örnekleme* veya *önemli örnekleme* olarak adlandırılan yöntem uygulanır. Algoritma, geçici küme \bar{X}_t içerisindeki L tane parçacığı önem ağırlıkları dikkate alınarak yerine farklı parçacıklar koyar ve algoritma tarafından parçacıklara ait dağılım önem ağırlıkları dahil edilecek şekilde parçacıklar değiştirilir. Böylece L parçacıklarından oluşan bir parçacık kümesi aynı boyuttaki başka bir parçacık grubuna dönüştürülür. Yeniden örnekleme basamağından önce parçacıklar $\bar{bel}(x_t)$ 'ye göre dağıtılır, yeniden örnekleme basamağından sonra ise yaklaşık olarak $bel(x_t) = \eta p(z_t | x_t^{[l]}) \bar{bel}(x_t)$ 'ye göre dağıtılırlar.

1.4.2 Parçacık Filtresinin Matematiksel Türetimi

[17] ile verilen kaynaktan yararlanılarak matematiksel türetim gerçekleştirilmiştir. Parçacık filtresini matematiksel olarak türetebilmek için parçacıkların durum dizisi örnekleri olarak düşünülmesi gerekmektedir.

$$x_{0:t}^{[l]} = x_0^{[l]}, x_1^{[l]}, \dots, x_t^{[l]} \quad (1.2)$$

Parçacık filtresi algoritması parçacıkların üretildiği durum örneklerinin $x_{0:t-1}^{[l]}$ sırasını, parçacık $x_t^{[l]}$ ifadesine eklemektedir. Filtre, inanç $bel(x_t) = p(x_t | u_{1:t}, z_{1:t})$ yerine tüm ardıl durum dizisini hesaplamaktadır.

$$bel(x_{0:t}) = p(x_{0:t} | u_{1:t}, z_{1:t}) \quad (1.3)$$

Ardıl durum dizisi $bel(x_{0:t})$, $bel(x_t)$ fonksiyonundan analogik şekilde türetilerek bulunur. Eşitlik (1.4) ile aşağıdaki gibi verilmiştir,

$$\begin{aligned}
p(x_{0:t}|z_{1:t}, u_{1:t}) &= \eta p(z_t | x_{0:t}, z_{1:t-1}, u_{1:t}) p(x_{0:t} | z_{1:t-1}, u_{1:t}) \quad (1.4) \\
&= \eta p(z_t | x_t) p(x_{0:t} | z_{1:t-1}, u_{1:t}) \\
&= \eta p(z_t | x_t) p(x_t | x_{0:t-1}, z_{1:t-1}, u_{1:t}) p(x_{0:t-1} | z_{1:t-1}, u_{1:t}) \\
&= \eta p(z_t | x_t) p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1})
\end{aligned}$$

Bu eşitliklerde integral işareti bulunmaması, öncülde yer alan tüm durumların kapsandığı anlamına gelmektedir.

Öncül dağılımdan $p(x_0)$ parçacıklar türetilmektedir. Varsayalım ki $t - 1$.inci zamandaki parçacıklar $bel(x_{0:t-1})$ fonksiyonuna göre dağıtılmış olsun, bu fonksiyonun içindeki l .inci parçacık $x_{0:t-1}^{[l]}$ olarak gösterilmekte ve $x_t^{[l]}$ örneği, algoritmanın 4.adımında aşağıda Eşitlik (1.5)'te yer alan dağılıma göre türetilmektedir. Eşitlik (1.6) ise o örneğe ait ağırlık değerini göstermektedir.

$$p(x_t | x_{t-1}, u_t) bel(x_{0:t-1}) = p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1}) \quad (1.5)$$

$$\begin{aligned}
w_t^{[l]} &= \frac{\text{hedef dağılım}}{\text{önerilen dağılım}} \quad (1.6) \\
&= \\
&= \frac{\eta p(z_t | x_t) p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{1:t-1}, u_{1:t-1})}{p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{0:t-1}, u_{0:t-1})} \\
&= \eta p(z_t | x_t)
\end{aligned}$$

Eşitliklerde yer alan η sabitinin yeniden örnekleme ve ağırlıkların belirlenmesinde bir rolü yoktur. Çünkü yeniden örnekleme, önem ağırlıklarının olasılığa oranına göre hesaplanmaktadır. $w_t^{[l]}$ ile orantılı olasılıktaki parçacıkların yeniden örneklenmesi ile ortaya çıkan parçacıklar, önerilen dağılım ile önem ağırlıklarının $w_t^{[l]}$ çarpımına göre Eşitlik (1.7)'de verildiği gibi dağılım göstermektedir.

$$\eta w_t^{[l]} p(x_t | x_{t-1}, u_t) p(x_{0:t-1} | z_{0:t-1}, u_{0:t-1}) = bel(x_{0:t-1}) \quad (1.7)$$

2 MATEMATİKSEL MODEL VE SAYISAL YÖNTEM

2.1 Matematiksel Model

Sisteme ait matematiksel model olarak, tek bir Drone Baz İstasyonunun (DBİ) yerde bulunan tek kullanıcı yerinin kestirildiği durum ile yerde hareketli kullanıcının takibi olacak şekilde ele alınmıştır. Aynı kestirim metodu, çoklu kullanıcıların yerinin tespiti için de uygulanabilir. Tek kullanıcının yeri D_{max} boyutuna sahip bir kare alan içerisinde rastgele olacak şekilde konumlandırılmıştır. Konuma ve DBİ'ye ait pozisyon bilgileri sırası ile (x^u, y^u) ve (x^d, y^d, h^d) olarak gösterilmektedir. Kullanıcı ile DBİ arasındaki yatay uzaklık parametresi r^{ud} ile belirtilmiş ve $r^{ud} = \sqrt{(x^u - x^d)^2 + (y^u - y^d)^2}$, iki pozisyon arasındaki gerçek uzaklık ise d^{ud} olarak belirtilmiş ve $d^{ud} = \sqrt{(h^d)^2 + (r^{ud})^2}$ olacak şekilde hesaplanmıştır. İki pozisyon arasındaki yükseliş açısı ise $\theta^{ud} = \tan^{-1}\left(\frac{h^d}{r^{ud}}\right)$ olarak verilmiştir. [1] ve [2]'de yer alan olasılıksal yolkayıp modeli (İng., *probabilistic pathloss model*) varsayımı tezde kullanılan olasılıksal yol kayıp modelinde kullanılmıştır. Bu modele göre, kullanıcı konumu ile DBİ'nin birbirlerini görme veya birbirlerinin görüş hattı içinde olma (İng., *line-of-sight, LoS*) ve birbirlerinin görüş hattı dışında olma (İng., *non-line-of-sight, NLoS*) (birbirlerini görmeme) olasılıkları yükseliş açısı θ^{ud} değerine bağlıdır.

$$p_{LoS}^{ud}(\theta^{ud}) = \frac{1}{1+a\left(-b\left(\frac{180}{\pi} \tan^{-1}\left(\frac{h^d}{r^{ud}}\right)-a\right)\right)} \quad (2.1)$$

$$p_{NLoS}^{ud}(\theta^{ud}) = 1 - p_{LoS}^{ud} \quad (2.2)$$

Eşitlik (2.1) ve Eşitlik (2.2) denklemlerinde belirtilen a ve b değerleri çevresel parametrelerdir ve kullanıcı ile DBİ'nin konuşlandığı çevresel alana göre farklılık göstermektedir örneğin şehir, şehir çevresindeki yerleşim yeri veya kırsal alan gibi.

Kullanıcı ile HBİ'nin birbirlerini görme ve görmeme durumlarında, yol kaybı değeri ise (dB cinsinden) aşağıda Eşitlik (2.3) ve Eşitlik (2.4) denklemlerinde sırası ile verilmektedir.

$$PL_{LOS}^{ud} = 10\gamma \log\left(\frac{4\pi d^{ud} f_c}{c}\right) + \eta_{LOS}(\theta^{ud}) \quad (2.3)$$

$$PL_{NLOS}^{ud} = 10\gamma \log\left(\frac{4\pi d^{ud} f_c}{c}\right) + \eta_{NLOS}(\theta^{ud}) \quad (2.4)$$

Denklemlerde yer alan γ yol kaybı üssü, f_c taşıyıcı frekans (Hz), c ise (m/s) cinsinden ışık hızı değerleridir. η_{LOS} ve η_{NLOS} değerleri görüş hattı içinde ve görüş hattı dışında durumlar için sırası ile yol kaybı değerleridir. [1]'da türetilen örnekler, yükseliş açısı fonksiyonuna bağlı olacak şekilde Eşitlik (2.5) ve (2.6)'daki gibi rastgele Gaus dağılımına oturtulmuştur.

$$\eta_{LOS}(\theta^{ud}) \approx N\left(\mu_{LOS}, \sigma_{LOS}^2(\theta^{ud})\right) \quad (2.5)$$

$$\eta_{NLOS}(\theta^{ud}) \approx N\left(\mu_{NLOS}, \sigma_{NLOS}^2(\theta^{ud})\right) \quad (2.6)$$

Görüş hattında bulunma ve görüş hattında bulunmama durumunda yol kaybına ait ortalama ve standart sapma değerleri sırası ile μ_{LOS} ve μ_{NLOS} ile $\sigma_{LOS}(\theta^{ud})$ ve $\sigma_{NLOS}(\theta^{ud})$ 'dur. Standart sapma değeri ise aşağıda Eşitlik (2.7) ve Eşitlik (2.8) ile verilmiştir.

$$\sigma_{LOS}(\theta^{ud}) = \alpha_1 \exp(-\beta_1 \theta^{ud}) \quad (2.7)$$

$$\sigma_{NLOS}(\theta^{ud}) = \alpha_2 \exp(-\beta_2 \theta^{ud}) \quad (2.8)$$

Eşitlik (2.7) ve Eşitlik (2.8)'de yer alan $\alpha_1, \alpha_2, \beta_1$ ve β_2 parametreler koşulan senaryonun içinde bulunduğu ortama ait çevresel parametrelerdir.

Görüş hattı içinde ve görüş hattı dışında olma durumlarına ait yol kaybı ortalama değeri yükseliş açısı ve uzaklığa bağlı olacak şekilde sırası ile Eşitlik (2.9) ve Eşitlik (2.10) ile verilmiştir.

$$\overline{PL}_{LOS}(\theta^{ud}, d^{ud}) = 10\gamma \log\left(\frac{4\pi d^{ud} f_c}{c}\right) + \mu_{LOS} \quad (2.9)$$

$$\overline{PL}_{NLOS}(\theta^{ud}, d^{ud}) = 10\gamma \log\left(\frac{4\pi d^{ud} f_c}{c}\right) + \mu_{NLOS} \quad (2.10)$$

Yükseliş açısı ve uzaklık verildiğinde bulunacak yol kaybı olasılık yoğunluk fonksiyonu Eşitlik (2.11) ile formüle edilmiştir.

$$f_{PL}(pl|\theta^{ud}, d^{ud}) = \frac{p_{LOS}^{ud}(\theta^{ud})}{\sqrt{2\pi\sigma_{LOS}^2(\theta^{ud})}} e^{-\frac{(pl-\bar{P}_{LOS}(\theta^{ud}, d^{ud}))^2}{2\sigma_{LOS}^2(\theta^{ud})}} + \frac{p_{NLOS}^{ud}(\theta^{ud})}{\sqrt{2\pi\sigma_{NLOS}^2(\theta^{ud})}} e^{-\frac{(pl-\bar{P}_{NLOS}(\theta^{ud}, d^{ud}))^2}{2\sigma_{NLOS}^2(\theta^{ud})}} \quad (2.11)$$

2.2 Sayısal Yöntem

Çalışma kapsamında kullanılan temel denklemler olarak Olasılıksal Yol Kayıp Modeli, Parçacık Filtresi Algoritması, Yeniden Örnekleme Metodları ve Drone Baz İstasyonu Enerji Modeli kullanılmıştır.

2.2.1 Olasılıksal Yol Kayıp Modeli

Eşitlik (2.11) ile verilmiştir.

2.2.2 Parçacık Filtresi Algoritması

[9] ile verilen kaynakta belirtildiği gibi Parçacık Filtresi Hareketli Baz İstasyonu tarafından yerdeki sabit kullanıcıyı bulmak amacı ile kullanılmaktadır. Tez kapsamında oluşturulan PF algoritması aşağıda Çizelge 2.1’de verilmiştir.

Çizelge 2.1: Tez kapsamında oluşturulan PF algoritması.

- 1: L adet parçacık konumu üret: $(x^l, y^l), \forall l = 1, \dots, L, t = 0, w_l = \frac{1}{L}, \forall 1, \dots, L.$
- 2: Hesapla θ^{ld} ve $d^{ld}, \forall l = 1, \dots, L.$
- 3: $N_{th} = 0.1L$
- 4: **while** not converge **do**
- 5: $t = t + 1$
- 6: Yolkaybı ölçümü al: pl_t
- 7: Hesapla: $f_{PL}(pl_t|\theta^{ld}, d^{ld}), \forall l = 1, \dots, L.$
- 8: Ağırlıkları güncelle: $w_l = w_l \times f_{PL}(pl_t|\theta^{ld}, d^{ld}), \forall l = 1, \dots, L.$
- 9: Ağırlıkları normalize et: $w_l \rightarrow \frac{w_l}{\sum_{i=1}^L w_i}, \forall l = 1, \dots, L.$
- 10: Hesapla: $N_{eff} = \frac{1}{\sum_{l=1}^L w_l^2}.$
- 11: **if** $N_{eff} < N_{th}$ **then**

```

12:         Yeniden örnekleme yap:  $(x^l, y^l), \forall l = 1, \dots, L.$ 
13:         Ağırlıkları yenile:  $w_l = \frac{1}{L}, \forall l = 1, \dots, L.$ 
14:     endif
15:         Rastgele karıştır:  $(x^l, y^l) \rightarrow (x^l + rand \times \Delta_x, y^l + rand \times \Delta_y), \forall l = 1, \dots, L.$ 
16:     Hesapla  $\theta^{ld}$  ve  $d^{ld}, \forall l = 1, \dots, L.$ 
17:     Kestirimi güncelle:  $(\mu_x, \mu_y) = (\sum_{l=1}^L w_l x^l, \sum_{l=1}^L w_l y^l)$ 
18:     Varyansı güncelle:  $(\sigma_x, \sigma_y) = (\sum_{l=1}^L (x^l - \mu_x)^2 w_l, \sum_{l=1}^L (y^l - \mu_y)^2 w_l), \forall l = 1, \dots, L.$ 
19:     if  $(\sigma_x < \sigma_{th})$  and  $(\sigma_y < \sigma_{th})$  then
20:         converge.
21:     endif
22: end while
23: Return:  $(\mu_x, \mu_y)$ 

```

Parçacık filtresi boyutu D_{max} olan kare bir alanda L tane parçacık oluşturularak başlatılır. Parçacıklara ait ağırlıklar 1.satırda verildiği gibi $w_l = \frac{1}{L}$ olacak şekilde hesaplanır. Drone Baz İstasyonunun kendi lokasyonunu tam olarak bildiği varayılımı ile her bir parçacık ile DBİ arasındaki yükseliş açısı ve uzaklık 2.satırda sırası ile $d^{ud} = \sqrt{(h^d)^2 + (r^{ud})^2}$ ve $\theta^{ud} = \tanh^{-1}\left(\frac{h^d}{r^{ud}}\right)$ formülleriyle hesaplanır. Filtre için etkin parçacık sayısı N_{eff} değerinin belirli bir eşik değerden N_{th} daha az tutulmak istenir ve bu değere göre yeniden örnekleme gerçekleşmektedir. Parçacık sayısı eşik değeri kaynak [9] ve [10] ile verildiği gibi $N_{th} = 0.1L$ olacak şekilde alınmıştır (3.satır). Algoritmanın ana döngüsü satır 4 ile 22 arasında gerçekleşmektedir. Her adımda, DBİ yol kaybı ölçümü yapmakta (6.satır) ve her parçacık için ölçülen yol kaybı değerinin olasılığını hesaplamaktadır (7.satır). Ağırlık değerleri 8.satırda güncellenir ve 9.satırda normalize edilir. Etkin parçacık sayısı bulunur ve bu değer eşik değeri altında olursa yeniden örnekleme başlar (10.satır). İlerleyen bölümlerde anlatılacağı gibi, yeniden örnekleme temelde yüksek olasılıklı parçacıklar çoğaltılmasını sağlamaktadır. Bu sebeple, 15. ve 16.satırlarda verildiği gibi yüksek olasılıklı

parçacıklar rassal olarak yerleri biraz değiştirilmek sureti ile çoğaltılmaktadırlar. Yerde sabit kullanıcı yer tahmin değeri, parçacık lokasyonlarının ağırlıklandırılmış toplamıdır. Parçacıklara ait yer tahmin değeri (ortalama) ve varyans 17. ve 18.satırlarda verilmektedir. Eğer varyans değeri belirli bir eşik değeri altına düşerse algoritma sonlanmaktadır (19.satır).

Yerde konumu sürekli değişen kullanıcı için ise kullanıcıya ait bir hareket modeli belirlenmektedir. Tez kapsamında yerdeki kullanıcı için belirlenmiş hareket modeli dairesel yolda ilerleme şeklindedir. Dairesel yolun çap değeri ile kullanıcı hızı, takip algoritmasını etkileyen performans parametreleri olarak belirlenmiştir. Kullanıcının sabit durumdaki yer tahmininde HBİ üç farklı tip yörünge izlemişse de kendisine ait bir hareket modeli ve sabit hızı bulunmaktadır. Bu senaryoda ise HBİ'nin hareket modeli, en son güncellenen kullanıcı yeri olacak şekilde HBİ konumu olarak her iterasyonda belirlenmektedir. Kısaca HBİ'ye ait belirli bir hareket modeli bulunmamakta ve bir hareket noktasından başka bir hareket noktasına geçişinde hızı sabit veya değişken olarak ele alınmaktadır.

2.2.3 Yeniden Örnekleme Yöntemleri

Yeniden örneklemede, önceki ağırlık değerlerine göre parçacıklar yeniden çoğaltılmaktadırlar. Bu yöntemin temel amacı düşük ağırlıklı parçacıkların (örneğin düşük olasılık değerine sahip parçacıklar) elenmesini sağlamaktır. PF algoritmasında yeniden örnekleme yapılmazsa, algoritmada hızlı bir şekilde dejenerasyon olmaktadır yani ağırlığı yüksek olan bir kaç parçacık kalan tüm parçacıkları domine etmektedir. Bu da yapılan kestirimin yanlış olmasını ve kestirimlere ait varyans değerlerinin büyümesini sağlamaktadır. Sonuç olarak PF algoritmasının en önemli adımlarından birisi yeniden örneklemedir denebilir. Literatürde birden çok yeniden örnekleme metodu bulunmaktadır. Tez çalışmasında, farklı yeniden örnekleme metodları kullanarak, gerçekleştirilen kestirime ait lokalizasyon hatasının ortalamasındaki ve hatanın maksimum değerindeki değişiklik incelenmiştir. Bu kapsamda ele alınan yeniden örnekleme yöntemleri Çok Terimli (ing. *Multinomial*), Arta Kalan (ing. *Residual*), Kademeli (ing. *Stratified*) ve Sistemattir (ing. *Systematic*). Aşağıda incelenmiş olan yeniden örnekleme metodları ve yapılan performans çalışmaları yer almaktadır.

2.2.3.1 Çok Terimli Örnekleme Yöntemi

Çok Terimli yeniden örnekleme metodundaki temel düşünce, $(0,1]$ aralığında n_t^n şeklinde birbiriçimli dağılıma sahip N tane bağımsız rastgele değişken üretip, bu değişkenleri parçacık seçiminde kullanmaktır. N değeri ağırlıksayısı olarak ele alınır ve toplam parçacıkların sayısına eşit olabilir. n_t^n dağılımının limitleri $Q_t^{l-1} < n \leq Q_t^l$ olarak verilmektedir. Çok Terimli Örnekleme yöntemine basit rastgele metod da denilmektedir. Parçacıkların örnekleme rastgele olduğu için verilen parçacık için örnekleme sayısının üst ve alt limitleri sırası ile sıfır (örneklememiş) ve N_t (N_t defa örneklemiş) değerleridir. Bu durum, yeniden örnekleme parçacıkları için maksimum varyans getirmektedir.

Çok Terimli yeniden örnekleme metodunun hesaplama karmaşıklığı $O(NM)$ seviyesindedir. Bu metod hesaplamada yavaşlığa sebep olduğu için daha hızlı yöntemler geliştirilmiştir. M değeri toplamları gerçekleştirilecek ağırlıkların sayısıdır. Örneğin ikili arama ile m değerinin bulunması gerçekleştirilmiş ve hesaplama karmaşıklığı M değerinden $\log M$ değerine düşürülmüştür. Bir parçacığın örnekleme sayısına ait varyans değerinin düşürülmesi ise kademelendirme (ing. *Stratification*) ve deterministik örnekleme ile gerçekleştirilebilir.

Algoritmada ilk başta $Q_m^t = \sum_{l=1}^m w_m$ tanımlanır. Bu yöntemde, $n \sim U(0,1)$ dağılımına göre rastgele bir sayı üretilmiştir. Daha sonra $Q_t^{l-1} < u \leq Q_t^l$ aralığında olan w_l ağırlığı örnek olarak seçilmiş ve bu seçim toplam parçacık sayısı L defa tekrarlanmıştır.

Çizelge 2.2: Çok terimli örnekleme kaynak kodu (MATLAB).

```
weight = w./sum(w);
N = length(weight);
Q = cumsum(weight);
index = zeros(1, N);
p = 0;
while p < N
    p = p + 1;
    sampl = rand;
    j = 1;
    while Q(j) < sampl
        j = j + 1;
    end
    index(p) = j;
end
```


2.2.3.2 Arta Kalan Örnekleme Yöntemi

Arta Kalan Örnekleme metodu iki aşamadan oluşmaktadır. Birincisi, ağırlığı $1/N$ değerinden fazla olan parçacıkların deterministik şekilde tekrarlanması; ikincisi, geri kalan ağırlık değerleri için rastgele örnekleme yapılması. Bu yöntem kullanılarak örneklenmiş olan m 'ninci parçacık $N_t^m + R_t^m$ defa örneklenmiş olur, burada N_t^m ve R_t^m değerleri birinci ve ikinci aşamadan kalan tekrarlamaya sayılarıdır. N_t^m değeri $\lfloor Nw_t^m \rfloor$ değerine eşittir. Birinci aşamada tekrarlanan tüm parçacıkların sayısı $N_t = \sum_{m=1}^M N_t^{(m)}$, ikinci aşamada tekrarlanan tüm parçacıkların sayısı $R_t = N - N_t$ olarak bulunur. Arta Kalan ağırlık ise $\hat{w}_t^m = w_t^{(m)} - \frac{N_t^m}{N}$ ile bulunur.

İkinci aşamada, parçacıklar Arta Kalan ağırlıklarına göre ve Çok Terimli Yeniden Örnekleme metodu kullanılarak çekilirler. Arta Kalan yöntemde her iki döngüye ait hesaplama karmaşıklığı toplanarak, yöntemin hesaplama karmaşıklığı bulunur, $O(M) + O(R_t)$. Bu yöntem kullanılarak örneklenmiş olan m 'inci parçacığın kaç defa örneklendiğine ait üst ve alt limitler, $\lfloor Nw_t^m \rfloor$ ve $\lfloor Nw_t^m \rfloor + R_t$ ile hesaplanır.

Algoritmada, yeniden örnekleme yöntemine ait ilk aşamada ağırlığı $1/L$ değerinden büyük olan parçacıklar deterministik olarak yeniden örneklenmiştir. l parçacığı $N_l = \lfloor L \times w_l \rfloor$ defa yeniden örneklenmiş ve böylece $L - \sum_{l=1}^L N_l$ tane örnek oluşmuştur. İkinci aşamada ise örneklere ait her ağırlık değeri $w_l - \frac{N_l}{L}$ olacak şekilde güncellenmiştir. Yeniden örneklemenin ikinci aşamasında bu ağırlık değerleri ile Çok Terimli yeniden örnekleme gerçekleştirilmiştir.

Çizelge 2.3: Arta kalan örnekleme kaynak kodu (MATLAB).

```
weight = w./sum(w);
N = length(weight);
index=zeros(N,1);
Ns = floor(N.*weight);
R = sum(Ns);
p = 1;
j = 0;
while j < L
j = j + 1;
cnt = 1;
```

```

while cnt <= Ns(j)
    index(p) = j;
    p = p + 1;
    cnt = cnt + 1;
end
end
N_rdn = N - R;
Ws = (N*weight - Ns)/N_rdn;
Q = cumsum(Ws);
while(p <= N)
    sampl = rand;
    j = 1;
    while(Q(j) < sampl),
        j = j + 1;
    end
    index(p) = j;
    p = p + 1;
end

```

2.2.3.3 Kademeli Yeniden Örnekleme Yöntemi

Kademeli Yeniden Örnekleme tüm parçacıkları alt parçacık gruplarına (katmanlara) ayıran bir yöntemdir. Yöntem, $(0,1]$ aralığını N tane alt kademeye $\left(0, \frac{1}{N}\right] \cup \dots \cup \left(1 - \frac{1}{N}, 1\right]$ olacak şekilde ayırır. Rastgele sayılar $\{n_t^{(n)}\}_{n=1}^N$ birbirinden bağımsız olacak şekilde her bir alt parçacık grubundan seçilmektedir. Normalize edilmiş ağırlıklar toplamı ile dağılımının limitleri $Q_t^{(m-1)} < n_t^{(n)} \leq Q_t^{(m)}$ formülüne göre belirlenir.

Algoritmada bu yöntem uygulanırken ilk önce, $(0,1]$ aralığı L adet alt aralığa $\left(0, \frac{1}{L}\right] \cup \dots \cup \left(1 - \frac{1}{L}, 1\right]$ ayrılmıştır. Her alt aralık için, $n^n \sim U\left(\frac{n-1}{L}, \frac{n}{L}\right), n = 1, \dots, L$ olacak şekilde rastgele sayı üretilmiştir. Her bir n^n için seçilen parçacık ağırlığı $w_l, Q_t^{l-1} < n^n \leq Q_t^l$ aralığını sağlamaktadır.

Çizelge 2.4: Kademeli yeniden örnekleme kaynak kodu (MATLAB).

```

N = length(weight);
Q = cumsum(weight);
index = zeros(1, N);
G = linspace(0, 1-1/N, N) + rand(1, N)/N;
p = 1;

```

```

j = 1;
while(p <= N && j <= L)
    while Q(j) < G(p)
        j = j + 1;
    end
    index(p) = j;
    p = p + 1;
end

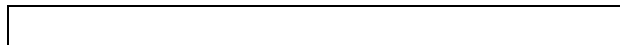
```

2.2.3.4 Sistematik Yeniden Örnekleme Yöntemi

Sistematik yeniden örnekleme metodunda da kademeli yeniden örneklemede olduğu gibi parçacıkların alt parçacıklara ayrılması yatmaktadır. Fakat kademeli yeniden örneklemeden farklı olarak, sistematik yeniden örneklemede $n_t^{(1)}$ rastgele değişkeni $(0, \frac{1}{N}]$ içerisinde seçilir, geri kalan u değişken ise deterministik olarak $n_t^{(1)} \sim U(0, \frac{1}{N}]$ $n_t^{(n)} = n_t^1 + \frac{n-1}{N}$, $n = 2, 3, \dots, N$ elde edilir.

Hem Kademeli hem de Sistematik Yeniden Örnekleme yöntemi hesaplama kompleksitesi $O(N)$ değerindedir. Sistematik yöntem, Kademeli yöntemle göre hesaplama yönünden daha etkilidir çünkü daha az sayıda rastgele sayı üretilmektedir. Sistematik yöntemde m 'inci parçacığın yeniden örnekleme sayısına ait alt ve üst limitler sırası ile $[Nw_t^m]$ ve $[Nw_t^m] + 1$ ile belirlenir. Limitler Kademeli Yeniden Örnekleme ile karşılaştırılınca farklılık göstermektedir. Kademeli Yeniden Örneklemede bu limitler sırası ile $([Nw_t^m] - 1, 0)$ ve $[Nw_t^m] + 2$ olarak ele alınır çünkü $\{n_t^{(n)}\}_{n=1}^N$ değişkenlerinin arası eşit değildir, bunun yerinde değişkenler arası fark $n = 2, 3, \dots, N$ için $\Delta n = n_t^n - n_t^{(n-1)}$ olarak bulur. Δu değerinin 0'a yaklaşması durumunda, çok düşük ağırlığa (sıfıra yakın ama Δu 'dan fazla) sahip parçacık iki defa yeniden örneklenebilir; Δn değerinin $2/N$ değerinden büyük olması durumunda ise ağırlığı $1/N$ ile Δn arasında olan parçacık atılabilir. Böylece, Sistematik Metod kullanılarak elde edilen yeniden örnekleme parçacığının varyansı Kademeli Metod kullanılarak elde edilen parçacıktan daha az olmaktadır.

Çizelge 2.5: Sistematik yeniden örnekleme kaynak kodu (MATLAB).



```

N = length(weight);
Q = cumsum(weight);
index = zeros(1, N);
G = linspace(0,1-1/N,N) + rand/N;
p = 1;
j = 1;
while(p<=N && j<=L),
    while Q(j) < G(p)
        j = j + 1;
    end
    index(p) = j;
    p = p + 1;
end

```

2.2.4 Hareketli Baz İstasyonu Enerji Modeli

HBI'ler limitli batarya ömrüne sahip ekipmanlardır, bu sebeple enerji etkin operasyon ihtiyaçları bulunmaktadır. Dronun batarya tüketim limiti olmasaydı, kendisinden beklenen işlevi optimizasyon ihtiyacı hissetmeden süreden bağımsız şekilde havada kalabilecek şekilde istendiği ölçüde yol kaybı ölçümü yapabiliirdi. Tez kapsamında İnsansız Hava Aracı enerji modeli olarak [13] ile önerilen model kullanılmaktadır, bu model [14] ile verilen çalışmada da kullanılmıştır.

İnsansız Hava Aracının harcadığı enerji temelde iki bileşenden oluşmaktadır: haberleşme kaynaklı enerji ve itki enerjisi. İtke enerjisi, İHA'nın hem havada durmasını hem de hareket yeteneğine sahip olmasını sağlar. Pratikte haberleşme için harcanan enerji itkiye harcanan enerji yanında ihmal edilebilmektedir, haberleşmeye harcanan enerji bir kaç watt mertebesindeyken, itki sağlamak için harcanan enerji yüzler watt mertebesinde [15]. Bu nedenle tez kapsamında haberleşme için harcanan enerji, İHA enerji modelinde hesaplamalara dahil edilmemiştir. Enerji hesabı kapsamında, İHA'nın temelde iki modu ele alınmıştır. Bunlar, *ileri uçuş* modu ve *asılı durma* modudur. Tüm uçuş boyunca aktif olması gereken *haberleşme* moduna yönelik harcanan enerji tez kapsamında ele alınmamıştır.

İleri uçuş modu, DBİ'nin bir ölçüm noktasından diğer ölçüm noktasına (İng., *wavepoint*) hareketidir, bu hareket düz bir hareket olma zorunda değildir. *Asılı durma* modunda ise DBİ yerdeki kullanıcıdan veri toplamakta yani ölçüm almaktadır. İlgili modların gerçekleştirilmesi amacı ile DBİ'nin harcadığı güç bulunurken, bıçak profili,

parazitik güç ve indüklenen güç değerleri hesaplanmaktadır. Bıçak profili, rotor kanadını döndürmek için gereken enerjidir. Parazitik enerji, yatay yönde sürüklenme kuvvetini yenmek için gereken enerjidir ve hızın küpü ile orantılıdır. DBİ'nin sürati yüksekken artmakta, asılı kalma modunda ise değeri sıfır olmaktadır. İndüklenen güç ise dronun yüksekliğini korumak için hava akışı sebebi ile oluşabilecek aerodinamik sürüklenme kuvvetine karşı ihtiyaç duyulan enerjidir. İndüklenen güç DBİ'nun havadaki hızı ile ters orantılı olarak değişmektedir. Asılı durma modunda, DBİ'nin yerden yüksekliğinin korunması için gereken kaldırma kuvveti ana rotorların dönüşü ile oluşturulmalıdır. Bu, az miktarda havanın önemli ölçüde hızlandırılması gerektiği anlamına gelmektedir. Bununla birlikte, DBİ *ileri uçuş* moduna geçtiğinde, rotorlardan daha yüksek kütle akışı olacak ve sonucunda bıçak diski artan aşağı doğru hava akışı sebebi ile daha fazla yükselecektir. Yerden yüksekliğin korunması için motorlara daha az hava ivmesi gerekir yani karşılığında daha az enerji harcanmış olur. Tezde ileri uçuş moduna geçildiğinde DBİ'nin küçük meyil açısına (İng., *tilt angle*) sahip olduğu ve değerinin beş dereceden az olduğu varsayılmıştır. Bu varsayımla, toplam güç ihtiyacı hızın bir fonksiyonu olarak aşağıda Eşitlik (2.12) ile verilmiştir.

$$P_d(v) = mgv_{ind} + \frac{1}{2}\rho v^3 C_{ds} + k_0 \left(1 + 3\frac{v^2}{v_t^2}\right) \quad (2.12)$$

Eşitlik (2.12)'de yer alan ve toplanan üç terim sırası ile indüklenen, parazitik ve bıçak profiline ait güç tüketimleridir. m HBI'nin kütlesi, g yer çekimi ivmesi, ρ hava yoğunluğu, v_t rotor kanadının doğrusal uç hızı, C_{ds} HBI'nin sürüklenme katsayısına bağlı ve k_0 ise bıçak boyutlarına bağlı olan sabit değerlerdir. Tez kapsamında değerler Çizelge 2.6 ile verildiği gibi alınmıştır. Ortalama indüklenen hız v_{ind} Eşitlik (2.15) ile verilmiştir.

$$v_{ind} = \sqrt{\frac{-v^2 + \sqrt{v^4 + \left(\frac{mg}{\rho A_d}\right)^2}}{2}} \quad (2.15)$$

Eşitlik (2.15)'te yer alan A_d insansız hava aracının alan değeridir. [14] ile verilen kaynakta DBİ'nin yol kaybı ölçümü yapmak ($v = 0$) için harcadığı süre 5 s olacak şekilde alınmıştır. Bu durumda harcanan güç $P_d(0) = k_0 + \sqrt{\frac{(mg)^3}{2\rho A_d}}$ ifadesi ile verildiği gibi bulunacaktır. Verilen parametreler için bir dronun 40 km/h hıza sahipken enerji harcamasının en az değerde olduğu bulunmuştur. Bu nedenle tez kapsamında

DBİ'nin ölçüm noktalarına düz bir çizgi üstünde olacak şekilde 40 km/s hız ile ilerlediği ve her ölçüm noktasında 5 s kaldığı varsayılmıştır. Değerlere ait parametreler aşağıda Çizelge 2.6'da verilmiştir.

Çizelge 2.6: Enerji modeli parametreleri.

Parametre	Açıklama	Değer
m	DBİ kütlesi [kg]	5
g	Standart yer çekimi ivmesi [m/sec^2]	10
ρ	Hava yoğunluğu	1.225
v_t	Rotor kanadının doğrusal uç hızı	100
C_{ds}	Sürtünme ve referans alan sabiti	0.4
k_0	Bıçak boyut sabiti	570
A_d	DBİ alanı [m^2]	0.25
v	DBİ ileri hızı [km/saat]	40
T	DBİ ölçüm noktası (sayısı)	4,9,16,25
l_T	İki ardışık ölçüm noktası arasındaki uzaklık [m]	-

DBİ'nin harcadığı enerjinin bulunması için, *ileri uçuş* modunda ve *asılı durma* modunda harcanılan zaman bilgisine ihtiyaç vardır. DBİ'nin gerçekleştirdiği ölçüm sayısının T olduğunu, yerden bir ölçüm aldıktan sonra diğer ölçüm noktasına gitmek için aldığı yolun ise l_T olduğu varsayılmıştır. Bu durumda DBİ'nin harcadığı toplam enerji dairesel yörünge için aşağıda Eşitlik (2.16) ile bulunmuştur.

$$E_T \cong \frac{T-1}{40} \times \sqrt{2} R \sqrt{1 - \cos\left(\frac{2\pi}{T}\right) P_d(40) + T \times P(0)} \quad (2.16)$$

Eşitlik (2.16)'da yer alan R değeri, DBİ'nin havadaki dairesel yörüngesine ait yarıçap değeridir. Toplam enerji E_T hesabında sırası ile ilk terim ileri yönlü enerji, ikinci terim ise asılı kalma enerjisidir.

DBİ'nin harcadığı toplam enerji grid yörünge için ise Eşitlik (2.17) ile verilmiştir.

$$E_T \cong \frac{T-1}{40} \times \frac{D_{maz}}{3} P_d(40) + T \times P(0) \quad (2.17)$$

2.2.5 Çevresel Parametreler

Tez çalışmasında araştırılan çevresel parametreler filtre performansını etkileyen parametreler arasında ele alınmaktadır. [16] ile verilen ve düşük irtifalı platformların

hava – yer arası yol kaybını modelleyen kaynakta sırası ile şehir içi, şehir dışı, kalabalık şehir içi ve yüksek binaların bulunduğu şehir yerleşimleri için radyo frekans modellemesi çalışılmıştır. Şehir içi modelleme, ITU-R (International Telecommunication Union) tarafından önerilen ve RF sinyali yayılımının genel şehir geometrisine bağlı olduğu standart üç parametre α_0 , β_0 ve γ_0 ile verilmektedir. Bu parametreler aşağıdaki gibi açıklanmıştır:

Parametre α_0 : Yerleşik arazi alanının toplam arazi alanına oranını temsil eder.

Parametre β_0 : Birim alan başına ortalama bina sayısını temsil eder (bina/km²).

Parametre γ_0 : Rayleigh olasılık yoğunluk fonksiyonuna göre bina yükseklik dağılımını tanımlayan ölçek parametresini temsil eder.

İlgili çalışma kapsamında şehir içine ait çevresel özellikler tipik bir Avrupa şehrini temsil edecek şekilde, şehir dışına ait çevresel özellikler kırsal bölgeler için, kalabalık şehir içine ait çevresel özellikler binaların birbirine yakın olduğu bazı şehir türleri için ve yüksek binaların bulunduğu şehir yerleşimleri ise gökdelenlerin bulunduğu şehirler olarak ele alınmıştır. Çizelge 2.7 ile verilen tabloda ITU-R tarafından seçilen şehir geometrisini tanımlayan parametreler verilmiştir.

Çizelge 2.7: Şehir modeli çevresel parametreleri.

Çevre Parametreleri	Parametre α_0	Parametre β_0	Parametre γ_0
Şehir İçi	0.1	750	8
Şehir Dışı	0.3	500	15
Yoğun Şehir İçi	0.5	300	20
Yüksek Bina Şehir İçi	0.5	300	50

Aynı kaynakta [16] şehir modeli çevresel parametrelerine ek olarak, çalışmanın devamında Radyo Frekans Modeli Parametreleri hesaplanmıştır. Tez çalışması kapsamında Radyo Frekans yayılımını etkileyen tüm çevresel parametrelerin filtrenin takip ve yer kestirimi performansına etkisi incelenmiştir. Hareketli Baz İstasyonu frekans değeri 2000 MHz olacak şekilde alınmıştır, bu nedenle [16] ile verilen kaynakta bu frekans değerine karşılık gelen RF modeli çevresel parametreler Çizelge 2.8’de verilen şekilde kullanılmıştır.

Çizelge 2.8: RF modeli çevresel parametreleri

2000 MHz				
	Şehir İçi	Şehir Dışı	Yoğun Şehir İçi	Yüksek Bina Şehir İçi
μ_1	0.1	1.0	1.6	2.3
μ_2	21	20	23	34
(a_1, b_1)	(11.25,0.06)	(10.39,0.05)	(8.96,0.04)	(7.37,0.03)
(a_2, b_2)	(32.17,0.03)	(29.6,0.03)	(35.97,0.04)	(37.08,0.03)
(c, d)	(0.76,0.06)	(0.6,0.11)	(0.36,0.21)	(0.05,0.61)

3 PARÇACIK FİLTRESİ KULLANILARAK KONUM KESTİRİMİ

3.1 Benzetim Çalışmaları

Literatürde parçacık filtresi kullanılarak gerçekleştirilen birçok konum kestirimi çalışmaları bulunmaktadır. Konum kestirimine ait matematiksel model ile birlikte bir sistem modeli oluşturulmuş ve parametreleri belirlenmiştir. Benzetim çalışmaları kapsamında parçacık filtresi algoritması temel alınarak oluşturulan matematiksel model MATLAB ortamında kodlanmıştır.

Parçacık Filtresinin konum kestirimine etki eden bir çok parametresi bulunmaktadır. Bu parametrelerin filtre performansına etkisi yani kullanıcının gerçek konumu ile kesitirilen konum arasındaki fark olan ortalama hata değeri tez kapsamında kontrol edilen parametre olmuştur.

Çalışmada ilk önce HBİ ile yerdeki sabit bir kullanıcının konum kestirimi yapılmaya çalışılmış, filtre parametrelerinin kestirim performansına etkisi incelenmiş ve Hareketli Baz İstasyonunun bu kestirimi gerçekleştirme sırasında harcadığı enerji belirlenerek kestirim hatası ile harcanan enerji arasında karşılaştırma yapılmıştır. Bu çalışmada Hareketli Baz İstasyonuna ait hareket modeli değiştirilerek farklı yörüngeler izlemesi sağlanmış, bu farklı yörüngelerin filtre performansına etkisi incelenmiştir.

Çalışmanın ikinci kısmında ise HBİ'nin yerde hareketli bir kullanıcının konumunu kesitirip, kullanıcının konum kestirimine ek olarak takip edilmes sağlanmıştır. Bu benzetim çalışmasında yerden baz istasyonuna herhangi bir geri bildirim yapılmamıştır. Böylece yerdeki kullanıcının ilk konumu dahil olmak üzere, ilerlediği yön ve sürat değeri de HBİ tarafından bilinmeyecek şekilde sadece RSSI değeri kullanılarak kullanıcının takip edilmesi sağlanmıştır.

3.2 Yerde Sabit Kullanıcının Konum Kesitirimi

Parçacık Filtresinde Eşitlik (2.1), (2.2), (2.3), (2.4), (2.7), (2.8), (2.9) ve (2.10) denklemleri ile verilmiş ve sırası ile görüş hattı içinde veya dışında olma olasılığı,

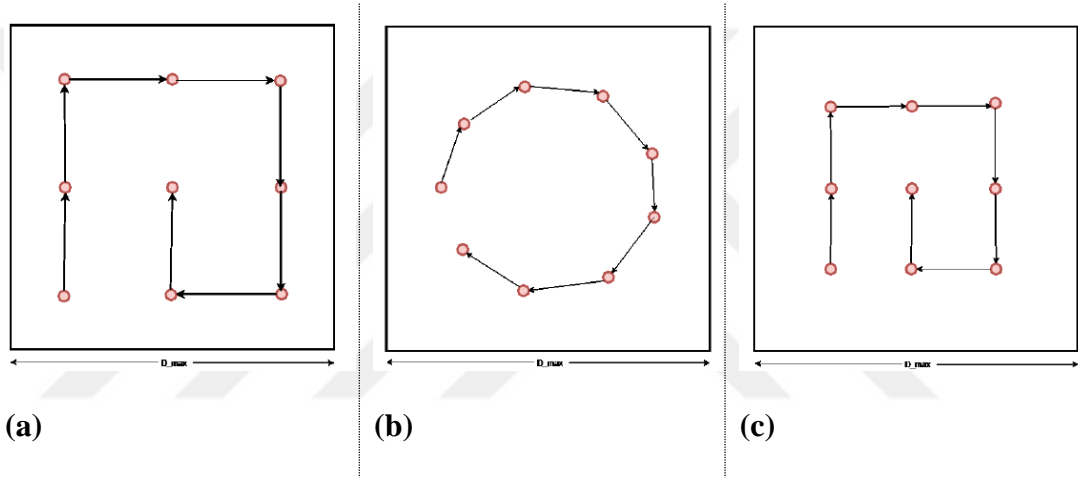
görüş hattı içinde veya dışında yol kaybı değeri, standart sapma değerleri, görüş hattı içinde veya dışında ortalama yol kayıp değerlerine ait ifadelerin bulunması için kullanılan ve şehir dışı çevre koşulunda sabit olan ve senaryoya göre değiştirilen parametreler aşağıda Çizelge 3.1 ile verilmiştir.

Çizelge 3.1: Kanal ve yol kaybı değişkenleri (Şehir dışı çevre koşulu).

Parametre	Tanım	Değer
f_c	Taşıyıcı frekans değeri	2 GHz
c	Işık hızı	3×10^8 m/s
D_{max}	Alan değeri	2000m
a, b	Çevresel Parametreler	9.61 , 0.16
α_1, β_1	Görüş Hattı parametresi	10.39 , 0.05
α_2, β_2	Görüş Hattı Dışı parametresi	29.6, 0.03
μ_{Los}, μ_{NLos}	Ortalama Yol Kaybı	1dB , 20dB
L	Parçacık Sayısı	Değişken
r	HBİ Yörünge yarıçap değeri	Değişken
T	HBİ Ölçüm Alma değeri	Değişken
h	HBİ Yerden Yükseklik değeri	Değişken
Yeniden Örnekleme Tipi	Çok Terimli (ing. <i>Multinomial</i>) Arta Kalan (ing. <i>Residual</i>) Kademeli (ing. <i>Stratified</i>) Sistemik (ing. <i>Systematic</i>).	Değişken

Yerde sabit kullanıcının konum kestiriminde yüksek performans elde edilmesi için parçacık filtresine ait değişkenlerin ortalama hataya etkisi incelenmiştir. Kullanıcının D_{max} içerisinde herhangi bir noktaya konuşturulması ile filtre başlatılmaktadır. Kullanıcı konumu rastgele şekilde 100 defa değiştirilmiş ve her konum için hata değeri hesaplanarak toplam 100 deneme (iterasyon) için ortalama hatanın karesi bulunmuştur. Hataların karelerinin ortalaması, yani tahmini değerler ile gerçek değer arasındaki ortalama kare farkı ölçülmüştür.

HBİ'nin havada izlediği yörünge için grid ve dairesel hareket modelleri incelenmiştir. HBİ'nin havada ölçüm sayısı T ile belirtilmiştir. HBİ'nin izlediği yörüngeye ait hareket modeli sırası ile aşağıda Şekil 3.1 ile belirtildiği gibidir.



Şekil 3.1: $T = 9$ değeri için farklı yörünge tipleri (a) grid, (b) dairesel, (c) dairesel ile verilen yörüngeye eş değer enerjili grid.

Sabit kullanıcı yer kesitimi için filtre parametreleri olan HBİ yüksekliği, HBİ yörüngesine ait yarıçap ve parçacık filtresi sayısının ortalama hata değerine etkisi Çizelge 3.2'de verilmiştir.

Çizelge 3.2: Ortalama hata ile parçacık sayısı (L), HBİ yüksekliği (h), HBİ yörüngesine ait yarıçap değeri.

$L = 1000, T = 12, h = 500, r = \dots$	$\frac{2000}{2}$	$\frac{2000}{3}$	$\frac{2000}{4}$	$\frac{2000}{5}$	$\frac{2000}{6}$	$\frac{2000}{7}$	$\frac{2000}{8}$
Ortalama hata ve r	165	133	159	183	242	261	336
$L = 1000, T = 12, r = 2000/3, h = \dots$	200	400	600	800	1000	1200	1400
Ortalama hata ve h	363	197	126	105	103	102	110

$h = 800, T = 12, r = 2000/3, L = \dots$	100	500	1000	1500	2000	2500	3000
Ortalama hata ve L	139	96	105	97	113	96	109

HBİ yörüngesine ait yarıçap değeri 2000/3 olduğu durumda en düşük ortalama hata değerini vermektedir. Bu benzetim gerçekleştirilirken h değeri 500 m olarak alınmış, ortalama en düşük hata değerini verecek şekilde ayarlanmamıştır.

HBİ yükseklik değerinin (h) ortalama hataya etkisi yorumlanırken D_{max} değeri temel alınmalıdır. D_{max} içerisinde 100 deneme için kullanıcı konumu rastgele belirlenmekte ve bu konumun kestirimine yönelik 100 deneme sonucu için ortalama hata değeri elde edilmektedir. HBİ yüksekliğinin D_{max} değerinin yarısına yakın olduğu durumda ortalama konum kestirim hatası en düşük değerini almaktadır. 800 m değerinde optimum kestirim gerçekleştiği söylenebilir. Benzetim gerçekleştirilirken HBİ yörünge yarıçap değeri bir önceki benzetim sonucunda en düşük ortalama hata değerini verecek şekilde ayarlanmıştır.

Parçacık sayısının ortalama hataya etkisinde ise hem HBİ yükseklik hem de yarıçap değerleri ortalama hatayı en çok düşüren değerler olarak belirlenmiş ve parçacık sayısı değeri 100 ile 3000 arasında değiştirilerek ortalama hataya etkisi incelenmiştir. Parçacık sayısının artırılması filtre performansını arttıracaktır çünkü tahmin için kullanılan ve ağırlıklandırılan olası konum sayısı artmaktadır. Fakat parçacık sayısı artırıldıkça benzetim zamanı uzamakta yani hesaplama kapasitesi ihtiyacı artmaktadır. Bu nedenle 500 parçacık sayısı değeri konum kestirimi için yeterli olarak ele alınmış ve optimum değer olarak belirlenmiştir.

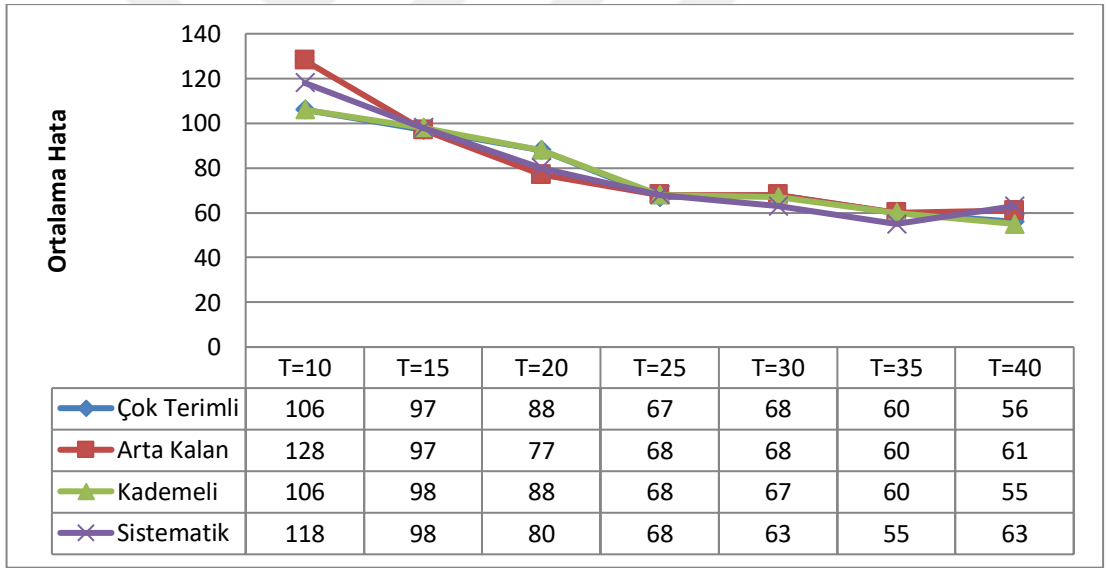
HBİ'nin ölçüm alma sıklığı arttırılarak, filtrenin yeniden örnekleme yöntemine etkisi incelenmiş ve ortalama konum kestirim hatasının değişimi Şekil 3.2'de verilmiştir. Yeniden örnekleme yöntemleri karşılaştırılırken ayrıca yeniden örnekleme sırasında hesaplama uzunluğunun belirlenmesi de sağlanarak, yöntemlerin daha sağlıklı karşılaştırılmasına çalışılmıştır. Bu sonuçlar da Çizelge 3.3 ile verilmiştir.

Çizelge 3.3: Yeniden Örnekleme Tiplerine ait Hesaplama Zamanı

Yeniden Örnekleme Tipi	T Değeri	Hesaplama Zamanı
Çok Terimli	40	87.3932
Arta Kalan	40	87.2250

Yeniden Örnekleme Tipi	T Değeri	Hesaplama Zamanı
Kademeli	40	87.1184
Sistemantik	40	86.9516

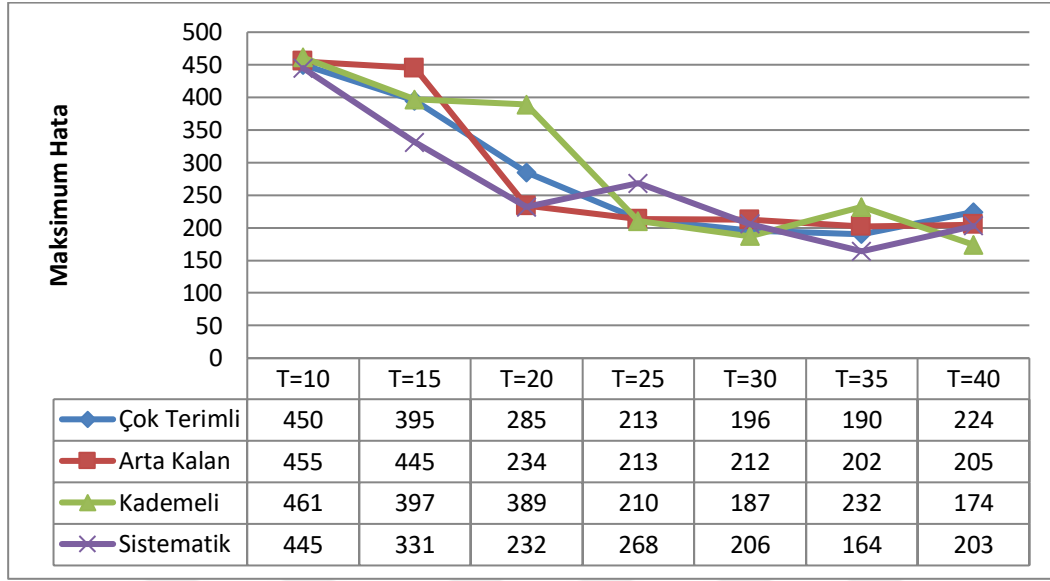
Çizelge 3.3'te yeniden örnekleme yöntemleri içerisinde hesaplama zamanı en düşük Sistemantik yeniden örnekleme yöntemi olduğu görülmektedir. Ortalama hata değerinin ölçüm alma sıklığı arttıkça tüm yeniden örnekleme tipleri için azalma eğilimi gösterdiği görülmüştür. Şekil 3.2'de görüldüğü üzere sistemantik yeniden örnekleme ise tüm örnekleme tipleri içinde ortalama konum kestirim hatasını en çok azaltan yöntemdir. Hesaplama zamanı ve konum kestirim hatasına olumlu etkisi sebebi ile devam eden benzetimlerde yeniden örnekleme tipi olarak Sistemantik kullanılmıştır.



Şekil 3.2: Yeniden örnekleme tipi ve HBİ ölçüm alma sayısının ortalama hataya etkisi.

Yeniden örnekleme tiplerinin ayrıca maksimum konum kestirim hatası belirlenerek, performansı en düşük yeniden örnekleme yöntemi anlaşılmasına çalışılmıştır. Şekil 3.3'te de görüldüğü gibi HBİ ölçüm sıklığı arttıkça ortalama konum kestirim hatasının azalmasına benzer şekilde, maksimum kestirim hatasında da azalma görülmüştür. HBİ'nin aldığı ölçüm sayısı arttıkça filtre hatasının azalması beklenen bir sonuçtur.

Maksimum hatayı en fazla arttıran yeniden örnekleme yöntemi Kademelidir. En iyi sonuç veren ise Sistematik yenien örnekleme yöntemidir.



Şekil 3.3: Ölçüm alma sayısı ve yeniden örnekleme yöntemlerinin maksimum konum kestirim hatasına etkisi.

HBİ enerjisinin modellenmesi için Bölüm 2.2.4'te yer alan enerji modeli kullanılmıştır. Enerji modelinde toplam harcanan enerji, HBİ'nin bir ölçüm noktasından diğer ölçüm noktasına geçişi için harcadığı enerji ile ölçüm alması için geçirdiği durma süresi boyunca harcadığı enerjinin toplamı olarak bulunmuştur. 100 deneme için HBİ hareketine ait ileri hız değeri 40 km/h, durma süresi sabit 5 s ve ileri hareketi sağlayan kanat hızı 100 olarak alınmıştır. Aşağıda HBİ dairesel ve grid yörüngelere ait değişen ölçüm sayısına göre HBİ enerjisi ve ortalama hata değeri verilmiştir.

Grid Yörünge: Şekil 3.1 (a) ile verildiği şekilde HBİ'nin yörünge düzlemi eşit gridlere bölünmüş ve HBİ ölçüm alma süresince durma noktası gridlerin orta noktası olacak şekilde belirlenmiştir. Toplam enerji değeri Eşitlik (3.1)'de verilmektedir.

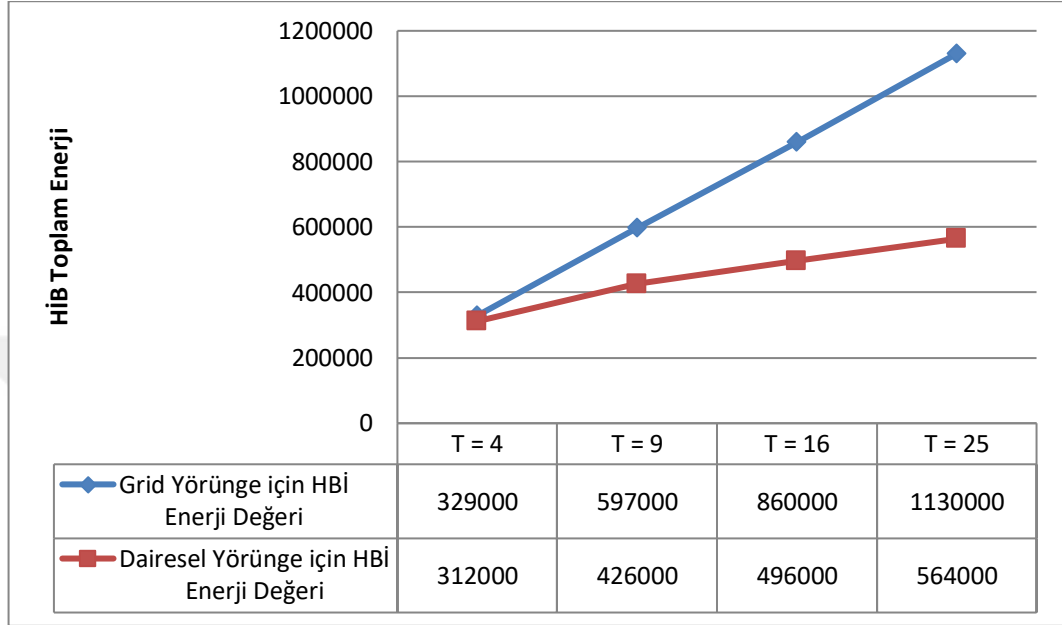
$$E_T \cong \frac{T-1}{40} x \frac{D_{max}}{3} P_d(40) + TxP(0) \quad (3.1)$$

Dairesel Yörünge: Şekil 3.1 (b) ile verildiği şekilde HBİ $\frac{D_{max}}{3}$ yarıçapına sahip bir daire yörüngesine sahiptir. Eşit T kadar bölünmüş ve bu değer kadar yol kaybı ölçümü yapan HBİ'nin toplam enerjisi Eşitlik (3.2) ile verilmektedir.

$$E_T \cong \frac{T-1}{40} x \sqrt{2}R \sqrt{1 - \cos(\frac{2\pi}{T})} P_d(40) + TxP(0) \quad (3.2)$$

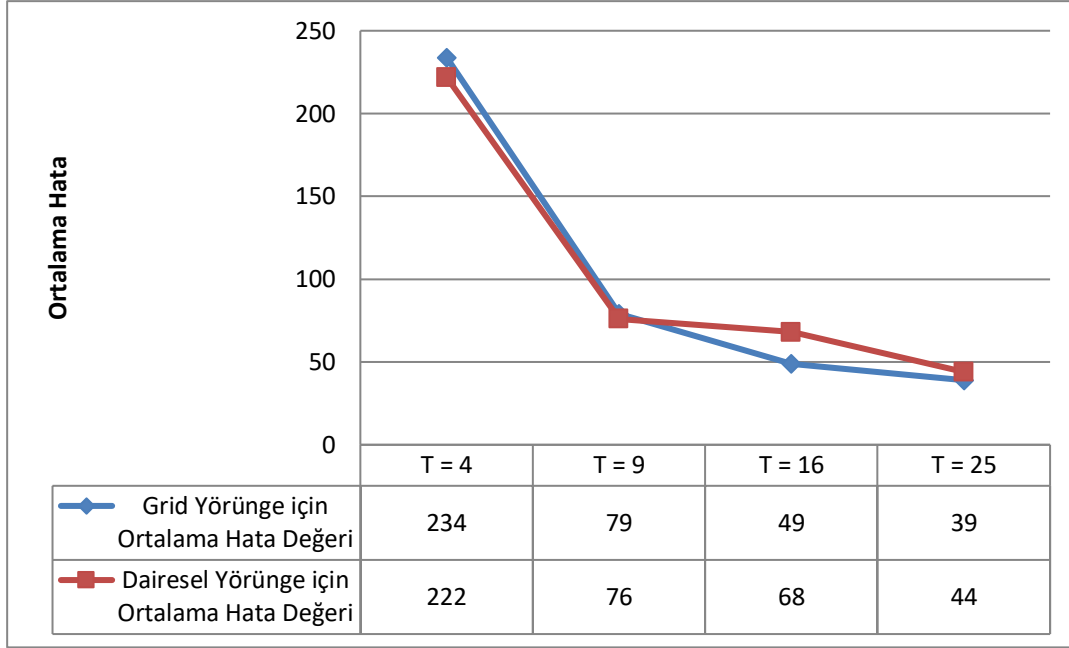
Dairesel yörüngenin yarıçap değeri R 'dir. Enerji hesabında sırası ile ilk terim ileri uçuş ikinci terim ise asılı durma sırasında harcanan enerji değerleridir.

Dairesel Yörünge: Şekil 3.1 (c) ile verilen grid yörüngede ölçüm noktaları birbirine yaklaşacak şekilde gridin küçüldüğü görülmektedir. Bu yörünge, Şekil 3.1 (b) ile verilen daireysel yörünge ile aynı enerjiyi harcamaktadır.



Şekil 3.4: HBI yörünge değişimi ve ölçüm sayısı için HBI toplam enerji (Joule) değişimi.

Şekil 3.4'te de görüldüğü gibi HBI'nin harcadığı toplam enerji alınan ölçüm sayısı arttıkça artmıştır. Bu sonuç bize sabit hızda hareket eden HBI'nin ölçüm alırken harcadığı enerjinin bir noktadan diğerine harcadığı enerjiden, yani toplam uçuş boyunca hareketinden kaynaklı olan enerjiden, fazla olduğunu vermektedir. Grid ve daireysel yörüngeler arasında ise HBI enerjisinin daireysel yörünge izlediği durumda ölçüm sayısı aynı kalsa dahi daha düşük enerji harcadığını göstermektedir. Bunun sebebi ise grid yörüngede bir ölçüm noktasından diğer ölçüm noktasına daha uzun süre uçuş yapılmasından yani mesafeden kaynaklıdır.



Şekil 3.5: HBİ yörünge değişimi ve ölçüm sayısı için ortalama hata değişimi.

Şekil 3.5'te de gösterildiği gibi ortalama hata değeri ise her iki yörünge için ölçüm sayısı arttıkça azalmıştır. Bu durum beklenen bir sonuçtur çünkü HBİ'nin sık ölçüm alarak kestirimi doğru yapma olasılığı artmaktadır. Sonuç olarak hem grid hem de dairesel yörüngede birbirine yakın ortalama hata değerleri çıkmaktadır, birinin diğerinden çok daha yüksek performans göstermemektedir.

Şekil 3.5 ve Şekil 3.6'nın yörüngeler kapsamında harcanan enerjilerin doğru karşılaştırılması için Şekil 3.1 (c) ile verilen grid yörünge değerlendirmeye alınmıştır. T=16 durumunda Şekil 3.1 (b) ve (c) ile verilen yörüngeler birbirine eşit ve 4.96×10^5 J enerji harcadığı hesaplanmıştır. Dairesel yörüngeye sahip HBİ'nin konum kestirim hatası 67.9 metre iken Grid yörüngeye sahip olduğu durumda ortalama hata değeri 84 metreye çıkmaktadır. Bu durum, aynı enerji harcama durumunda dairesel yörünge'nin konum kestiriminde daha yüksek performans gösterdiği şeklinde yorumlanabilir. Çünkü grid yörünge, harita üzerindeki alanı daha eşit şekilde gezmesine rağmen bunu yaparken daha uzun yol kat etmeyi gerektirmiştir, bu sebeple ileri uçuş için harcanan enerji daha yüksek çıkmıştır.

3.3 Yerde Hareketli Kullanıcının Konum Kestirimi

Benzetim çalışmalarının ikinci kısmında yerde hareketli bir kullanıcının parçacık filtresi ile konum kestirimi gerçekleştirilerek, HBİ'nin kullanıcıyı takip etmesi sağlanmaktadır. Bölüm 2.2.2 ile verilen parçacık filtresi algoritması yerdeki kullanıcıya bir hareket modeli eklenerek değiştirilmiş, HBİ'nin bir sonraki konumu ise kestirilen kullanıcı konumu olacak şekilde güncellenerek HBİ'nin hareketli kullanıcıyı takip etmesi sağlanmıştır. Yani HBİ'nin önceden belirlenmiş bir hareket modeli bulunmamaktadır.

Kullanıcının ilk benzetimde olduğu gibi ilk konumu rastgele seçilmiş ve bu noktadayken ilk kestirim gerçekleştirildikten sonra kullanıcı dairesel bir hareket modeline sahip olmuştur. Dairesel hareket modelinde yarıçap değeri kestirimi etkileyen parametrelerden biri olarak alınmıştır. Hareketli kullanıcının konum kestiriminde önemli olan diğer parametreler ise kullanıcı hızı, HBİ hızı ve parçacık sayısı olarak belirlenmiştir.

Parçacık filtresinin karakterizasyonu sabit kullanıcının konum kestiriminde araştırılmış ve filtrenin belirli parametreler için nasıl tepki verdiği sabit kullanıcı konumu senaryosunda incelenmiştir. Bu inceleme sonucunda bazı optimum parametreler elde edilmiştir. Yeniden örnekleme metodu olarak bir önceki senaryo için en optimum sonucu veren yöntem olan Sistematik yeniden örnekleme kullanılmıştır. Bu benzetimdeki amaç ise HBİ'nin kullanıcı konumunu bulduktan sonra düşük enerji ve ortalama hata değeri ile takibin gerçekleştirilmesini sağlayan filtre parametrelerinin elde edilmesidir.

Çizelge 3.4: Takip algoritması değişkenleri (Şehir Dışı Çevre Koşulu).

Parametre	Tanım	Değer
f_c	Taşıyıcı frekans değeri	2 GHz
c	Işık hızı	3×10^8 m/s
D_{max}	Alan değeri	2000m

Parametre	Tanım	Değer
a, b	Çevresel Parametreler	9.61 , 0.16
α_1, β_1	Görüş Hattı parametresi	10.39 , 0.05
α_2, β_2	Görüş Hattı Dışı parametresi	29.6, 0.03
μ_{Los}, μ_{NLos}	Ortalama Yol Kaybı	1dB , 20dB
v	HBİ İleri Yönlü Hız Değeri	Değişken
L	Parçacık Sayısı	Değişken
r	Kullanıcı hareketi yarıçap değeri	Değişken
$bitişzamanı$	Kullanıcı zaman değeri	Değişken
h	HBİ Yerden Yükseklik değeri	Değişken
Yeniden Örnekleme Tipi	Sistemik (ing. <i>Systematic</i>).	-

3.3.1 Takip Algoritması

Parçacık Filtresi algoritması temel alınarak oluşturulan takip algoritması, hareketli kullanıcının yerinin bulunmasını sağlamaktadır. Tez kapsamında oluşturulan takip algoritması aşağıda Çizelge 3.5 ile verilmiştir.

Çizelge 3.5: Takip algoritması.

<p>1: Kullanıcının ilk konumunu rastgele belirle</p> <p>2: HBİ ilk konumu belirle</p> <p>3: Parçacık Filtresi ile L adet parçacık konumu üret: $(x^l, y^l), \forall l = 1, \dots, L, t = 0, w_l = \frac{1}{L}, \forall 1, \dots, L.$</p> <p>2: Hesapla θ^{ld} ve $d^{ld}, \forall l = 1, \dots, L.$</p>
--

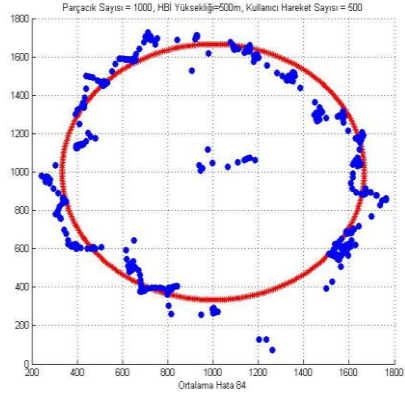
```

3:  $N_{th} = 0.1L$ 
4: while not converge do
5:    $t = t + 1$ 
6:   Yol kaybı ölçümü al:  $pl_t$ 
7:   Hesapla:  $f_{PL}(pl_t|\theta^{ld}, d^{ld}), \forall l = 1, \dots, L.$ 
8:   Ağırlıkları güncelle:  $w_l = w_l \times f_{PL}(pl_t|\theta^{ld}, d^{ld}), \forall l = 1, \dots, L.$ 
9:   Ağırlıkları normalize et:  $w_l \rightarrow \frac{w_l}{\sum_{i=1}^L w_i}, \forall l = 1, \dots, L.$ 
10:  Hesapla:  $N_{eff} = \frac{1}{\sum_{l=1}^L w_l^2}.$ 
11:  if  $N_{eff} < N_{th}$  then
12:    Yeniden örnekleme yap:  $(x^l, y^l), \forall l = 1, \dots, L.$ 
13:    Ağırlıkları yenile:  $w_l = \frac{1}{L}, \forall l = 1, \dots, L.$ 
14:  endif
15:  Rastgele karıştır:  $(x^l, y^l) \rightarrow (x^l + rand \times \Delta_x, y^l + rand \times \Delta_y), \forall l = 1, \dots, L.$ 
16:  Hesapla  $\theta^{ld}$  ve  $d^{ld}, \forall l = 1, \dots, L.$ 
17:  Kestirimi güncelle:  $(\mu_x, \mu_y) = (\sum_{l=1}^L w_l x^l, \sum_{l=1}^L w_l y^l)$ 
18:  Varyansı güncelle:  $(\sigma_x, \sigma_y) = (\sum_{l=1}^L (x^l - \mu_x)^2 w_l, \sum_{l=1}^L (y^l - \mu_y)^2 w_l), \forall l = 1, \dots, L.$ 
19:  if  $(\sigma_x < \sigma_{th})$  and  $(\sigma_y < \sigma_{th})$  then
20:    converge.
21:  endif
22:  Hareketli Baz İstasyonu konumu güncelle:  $(x_{HBi}, y_{HBi}) = (\mu_x, \mu_y)$ 
23:  Kullanıcı konumu güncelle
24:  Hareketli Baz İstasyonu enerjisini hesapla
26: end while
27: Ortalama hata hesapla
28: Return:  $(\mu_x, \mu_y)$ 

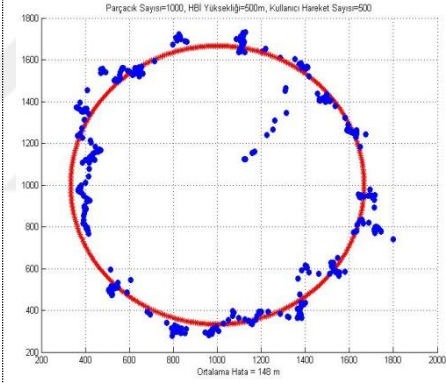
```

Parçacık Filtresinde kullanılan ve kestirimi etkileyen filtre parametreleri olarak parçacık sayısı, kullanıcı hızı ve kullanıcı hareket modeli ele alınmıştır. Filtre performansına bakılırken her seferinde Hareketli Baz İstasyonu tarafından harcanan toplam enerji de incelenmiştir. Ayrıca HBİ hızı da değiştirilerek senaryolar koşulmuş, optimum HBİ hızı belirlenmeye çalışılmıştır. HBİ enerji modelinden gelen ve HBİ'nin ölçüm aldığı süre boyunca harcadığı ve havada kalma süresi ile orantılı enerji değeri bu senaryolarda sıfır olacak şekilde alınmıştır. Yani HBİ'nin sadece durduğu süre boyunca ölçüm alması değil, sürekli ölçüm alacağı varsayılmış bu nedenle ölçüm almak için havada asılı kalmayacak şekilde enerji değeri hesaplanmıştır.

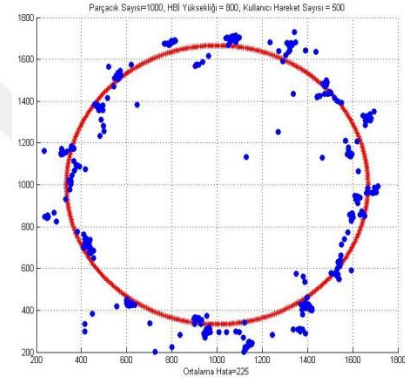
Filtre ile takip için ortalama hata değerine göre kullanıcıyı takip eden HBİ'nin havadaki hareketine ait izdüşümler çizdirilmiştir. Bu çizimlerde, takipte iyileşme olsa bile yani hata değerleri düşse bile izdüşüm çizimlerinde aynı oranda iyileşmeler görülmemekte, olsa bile göz ile ayırt edilememektedir. Sadece belirli hata değeri üstünde takibin olmadığı söylenebilmektedir. Takibin gerçekleşmediği ortalama hata değerine hem farklı senaryolarda hem de 100 üzeri benzetim koşuları sonucunda ulaşılmıştır. Takibin olmadığı hata değeri 300 m ve üzeri hata değeri olacak şekilde alınmıştır. Göz ile birbirine yakın hata değerleri için HBİ izdüşümleri yerine doğrudan hata değerlerine göre senaryolar incelenmiştir. Aşağıda Şekil 3.6 ile çeşitli ortalama hata değerlerinde HBİ izdüşümleri (mavi) ve kullanıcı hareketi (kırmızı) verilmiştir.



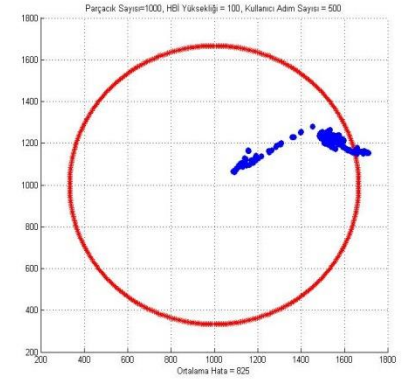
(a)



(b)



(c)



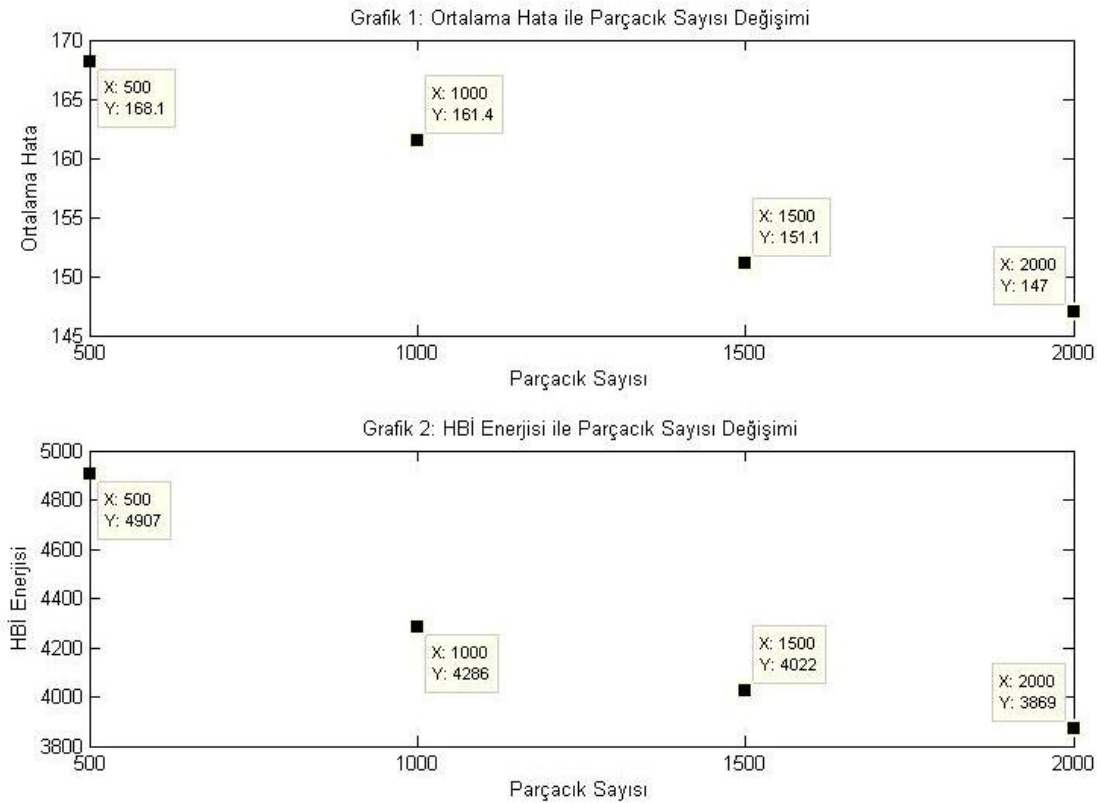
(d)

Şekil 3.6: HBİ izdüşüm ve kullanıcı hareketi (a) Parçacık Sayısı 1000, HBİ Yüksekliği 500m, Kullanıcı Toplam Zamanı 500, Hata Değeri 85 (b) Parçacık Sayısı 1000, HBİ Yüksekliği 500, Kullanıcı Toplam Zamanı 500, Hata Değeri 148 (c) Parçacık Sayısı 1000, HBİ Yüksekliği 800m, Kullanıcı Toplam Zamanı 500, Hata Değeri 225 (d) Parçacık Sayısı 1000, HBİ Yüksekliği 100, Kullanıcı Toplam Zamanı 500, Hata Değeri 825 değerlerine sahip grafiklerdir.

Benzetimler koşullarında sabit kullanıcıları durum ile aynı olacak şekilde birbirinden bağımsız 100 deneme için ortalama hata değeri hesaplanmıştır. Bu şekilde filtrenin belirli parametre değışikliklerinde belirli bir trendi izlenmesi sağlanmaya çalışılmıştır. Yeniden örnekleme metodu olarak sabit kullanıcıları senaryo için en iyi sonucu veren Sistematik yeniden örnekleme metodu kullanılmıştır. HBİ yüksekliği (h) tüm senaryolarda sabit tutulmuş ve 500 m olacak şekilde alınmıştır. Aşağıda sırası ile benzetim senaryolarında incelenen durumlar ve elde edilen ortalama hata ile buna eşlik eden enerji değeri arasındaki değışikliklere ait sonuçlar verilmiştir.

3.3.1.1 Parçacık sayısının takip performansı ve toplam HBİ enerjisine etkisi

Sabit kullanıcıları durumda filtre performansı ile parçacık sayısı arasındaki ilişki değerlendirilmiştir. İlgili senaryoda parçacık sayısının optimum değeri belirlenmeye çalışılmıştır. Hareketli kullanıcıları durumda ise parçacık sayısı daha geniş bir aralıkta taranarak birbirinden bağımsız 100 deneme için ortalama hataya ve HBİ enerjisine ($\times 1000$) etkisi incelenmiştir.

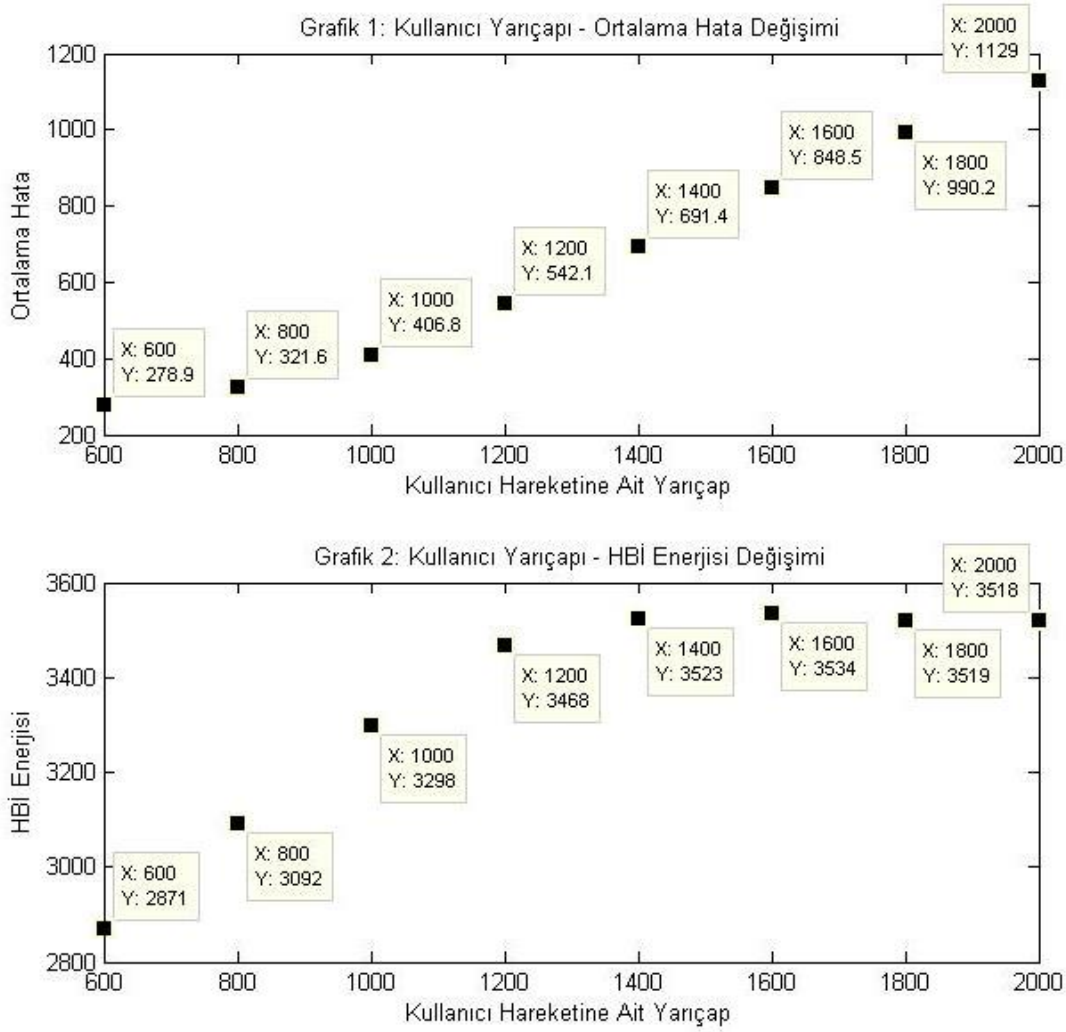


Şekil 3.7: Parçacık Sayısının Ortalama Hata ve HBİ Enerjisine ($\times 1000$) Etkisi

Şekil 3.7 Grafik 1’de parçacık sayısı (L) arttıkça fitrenin takip performansında artma olduğu ve takipte iyileşme gerçekleştiği görülmektedir. Parçacık sayısının (L) artması aynı şekilde harcanan enerjide de iyileşme meydana getirmiştir. Bunun sebebi temelde filtrenin yüksek parçacık sayısına sahip olduğu durumda daha iyi konum kesitiriminde bulunmasından kaynaklanmaktadır. Fakat parçacık sayısının artması işlem kapasitesini arttırdığı için benzetimlerin koşması daha uzun sürmüştür. Aslında bu HBİ’nin işlemci kapasitesini arttıran bir durum yaratmaktadır. Senaryolarda HBİ’nin ölçüm alma zamanı ayrıca belirtilmemiş veya sabit alınmamış, sürekli ölçüm aldığı varsayılmıştır. İşlemci kapasitesi ölçüm alma zamanı ile orantılıdır. Bu sebeple parçacık filtresinde parçacık sayısının artarak filtrenin kestirim yapma zamanında artış meydana getirmesinin, HBİ’nin ölçüm alma zamanına bir etkisi ve dolayısı ile HBİ havada kalma enerjisinde artma meydana getirmesi hesaba katılmamıştır. HBİ ileri yönlü hız değeri (v) sabit ve 40 km/h olacak şekilde alınmıştır. Senaryoda harcanan enerjiyi etkileyen en büyük parametre, takipten kaynaklı HBİ yer değiştirmesidir. HBİ yer değiştirmesinin büyük olduğu durumda ileri yönde harcanan enerji artmakta ve en yüksek değerini almaktadır. Sonuç olarak parçacık sayısındaki artış takibi kolaylaştırdığı yani ileri yönlü hareketi iyileştirmesi sebebi ile harcanan enerjide düşme meydana getirmektedir.

3.3.1.2 Dairesel hareket çapının takip performansı ve toplam HBİ enerjisine etkisi

Yerde kullanıcının dairesel hareketine ilişkin çap değeri arttırılmış ve sabit harita boyutu üzerinde daha geniş bir alanda gezen kullanıcı takibinin gerçekleştirilmesi senaryosu incelenmiştir. $4000 \times 4000 \text{ m}^2$ harita üzerinde dairesel harekete ait yarıçap değeri sırası ile 600 m , 800 m , 1000 m , 1200 m , 1400 m , 1600 m , 1800 m ve 2000 m olacak şekilde değiştirilmiştir. Senaryoda HBİ yüksekliği (h) 500 m , parçacık sayısı (L) 1000 , HBİ ileri yönlü hız değeri (v) 40 km/h , Kullanıcı zaman değeri (*bitişzamanı*) 500 olacak şekilde alınmıştır.



Şekil 3.8: Kullanıcı hareketine ait yarıçap değerinin ortalama hata ve HBİ enerjisine (x1000) etkisi (h değeri 500 m).

Şekil 3.8 ile verilen Grafik 1’de kullanıcı hareketine ait yarıçap değeri (r) arttıkça fitrenin takip performansında azalma olduğu ve belirli bir değerden sonra takibin gerçekleşmediği, gerçekleşse bile yüksek hata oranı ile gerçekleştiği görülmektedir. Kullanıcı hareketine ait yarıçap değerinin artması ile kullanıcı harita üzerinde daha geniş bir bölgede gezinmesi benzetilmeye çalışılmıştır. Böylece haritanın uç değerlerinde de kestirim yapılması sağlanmaya çalışılmıştır.

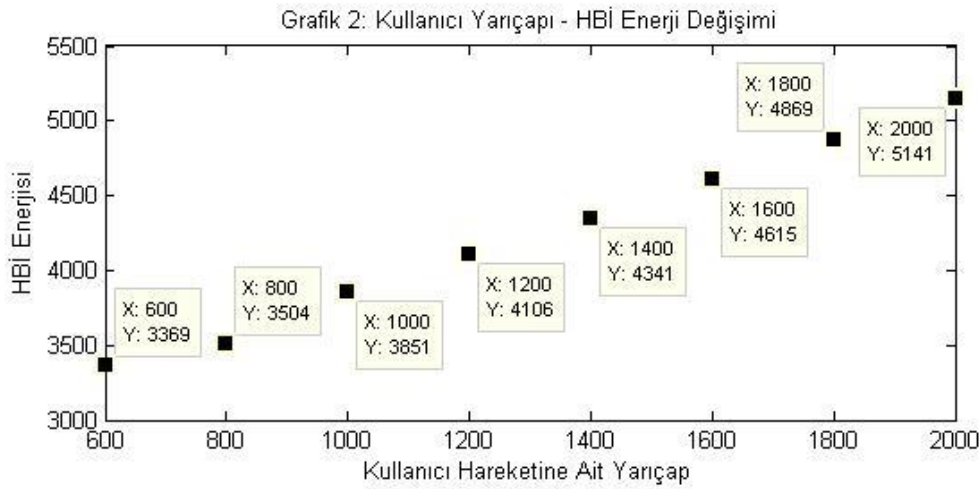
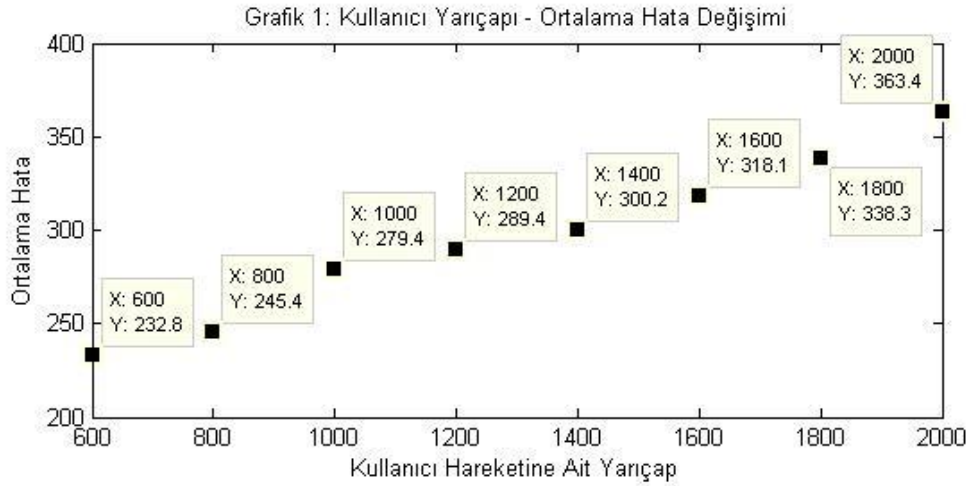
HBİ’nin takibi gerçekleştirilememesinin en temel sebeplerinden biri yerden yüksekliğinin (h) tüm senaryolar için sabit olmasıdır. HBİ yüksekliğinin (h), kullanıcı yarıçap değerine (r) yakın olduğu durumda en düşük ortalama hata değeri elde edilmiştir. Bunun sebebi, Yol Kaybı Modelinde kullanıcının HBİ’nin görüş hattı

içerisindeki kalma olasılığının pl_t yüksek olmasından kaynaklanmaktadır. HBİ ile kullanıcı arasındaki yükseliş açısı θ^{ld} arttığında ise bu değer azalmakta böylece HBİ'nin kullanıcının görüş hattı dışında olma olasılığı artarak, takip edememesi veya yüksek konumlama hatası ile takip etmesiyle sonuçlanmaktadır.

HBİ'nin takibi yüksek konumlama hatası veya takip edememesinin diğer sebebi ise parçacık sayısının sabit alınmasıdır. Parçacık sayısında artma olması filtrenin kestirim olasılığını arttırdığı için genel olarak filtrenin iyileşmesine etki ettiği önceki senaryolarda da tartışılmıştır.

Şekil 3.8 ile verilen Grafik 2'de ise kullanıcı hareketine ait yarıçap değeri (r) arttıkça HBİ'nin toplamda harcadığı enerjinin değişimi verilmektedir. Enerji değeri filtrenin takip performansı azaldıkça artmakta, takibin hiç yapılamadığı yani ortalama hata değerinin yaklaşık $600 m$ olduğu durumda ise enerji değerinde farklılık gözlenmemiştir. Bunun sebebi HBİ'nin takibi kaçırdığı için takip ettiği mesafeye kadar yüksek enerji harcadığı takibi kaçırdıktan sonra aynı yerde gezinmesi sebebi ile harcanan enerjiye katkıda bulunmadığı olarak yorumlanmıştır.

Bu senaryoda dairesel harekete ait yarıçap değeri (r) arttırıldığında $600m$ yarıçapa kadar yüksek ortalama hata değerleri ile takip gerçekleşmekte, bu değerden sonra ise takip gerçekleşmemektedir. HBİ yüksekliğinin arttırılarak takibin gerçekleştirilebilmesini sağlamak amaçlanmıştır. $4000 \times 4000 m^2$ harita üzerinde dairesel harekete ait yarıçap değeri sırası ile $600 m, 800 m, 1000 m, 1200 m, 1400 m, 1600 m, 1800 m$ ve $2000 m$ olacak şekilde değiştirilmiştir. Senaryoda HBİ yüksekliği (h) $1000 m$, parçacık sayısı (L) 1000 , HBİ ileri yönlü hız değeri (v) $40 km/h$, Kullanıcı zaman değeri (*bitişzamanı*) 500 olacak şekilde alınmıştır.



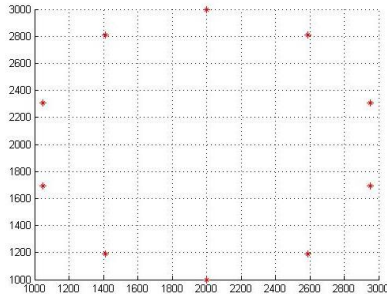
Şekil 3.9: Kullanıcı hareketine ait yarıçap değerinin ortalama hata ve HBI enerjisine (x1000) etkisi (h değeri 1000 m).

Şekil 3.9 ile verilen Grafik 1’de, h değeri 500 m olduğu durumdan çok daha yüksek performansta takip gerçekleştiği görülmektedir. HBI yükseklik değeri h arttığı için ASGG değerinde artma meydana gelmekte ve kullanıcının HBI’nin görüş hattı içerisinde kalma olasılığı artmaktadır. Bu sebeple takip performansının artması ve ortalama hata değerinde azalma gözlenmesi beklenen bir durumdur. Grafik 2’de ise harcanan toplam HBI değerinde h değeri 500 m olduğu duruma göre ortalama %17 değerinde artma meydana gelmiştir. Bölüm 2.2.4 ile verilen HBI enerji modelinde, yükseklikten kaynaklı hava yoğunluğunda meydana gelen değişikliğin indüklenen hız değerinde değişiklik yaratacağı belirtilmiştir. Yükseklik arttıkça hava yoğunluğu düşeceği için indüklenen hız değeri ve dolayısı ile harcanan güç değeri artacaktır. Bu sebeple HBI’nin harcadığı toplam enerjide de bir artış meydana gelecektir. HBI enerji

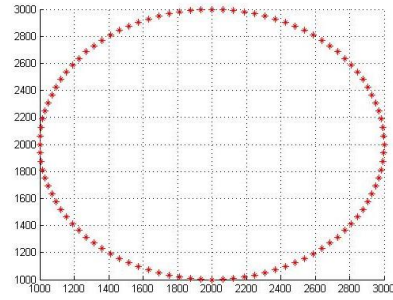
modelinde sürtünmeden kaynaklı enerji kaybı sadece ileri yönlü hız değiştiğinde değişmekte bunun dışında sabit olacak şekilde ele alınmaktadır. Daha gerçekçi bir modelde ise hava yoğunluğunun azalması ile sürtünme değeri azalacak ve yüksek irtifada sürtünmeden kaynaklı enerji kaybı daha az olacaktır. Fakat tez kapsamında kullanılan enerji modelinde bu durum ele alınmamıştır. Sonuç olarak benzetim sonuçları ile hem parçacık filtresi hem de enerji modeli ile tutarlı olduğu söylenebilir.

3.3.1.3 Kullanıcı Zaman Değerinin Takip Performansı ve Toplam HBI Enerjisine Etkisi

Kullanıcı zaman değeri (*bitişzamanı*) belirli yarıçap değerindeki dairesel izin kaç adımda biteceğini belirlemektedir. Yani, temelde yerdeki kullanıcının aldığı yolu belirlemektedir. Yerdeki kullanıcının belirli yarıçap değerindeki dairesel hareket izi, kullanıcının o hareketi tamamlama zamanına bölünür ve dairesel hareketin tamamlanma süresi değişmiş olur. Hareketin tamamlanma süresinin kısa tutulması, kullanıcının birim zamanda çok daha uzun mesafe kat etmesi anlamına gelmektedir. Aşağıda yer alan Şekil 3.9 ile verilen iki örnek sırası ile hareket tamamlanma süresi 10 ile 100'ü göstermektedir.



(a)

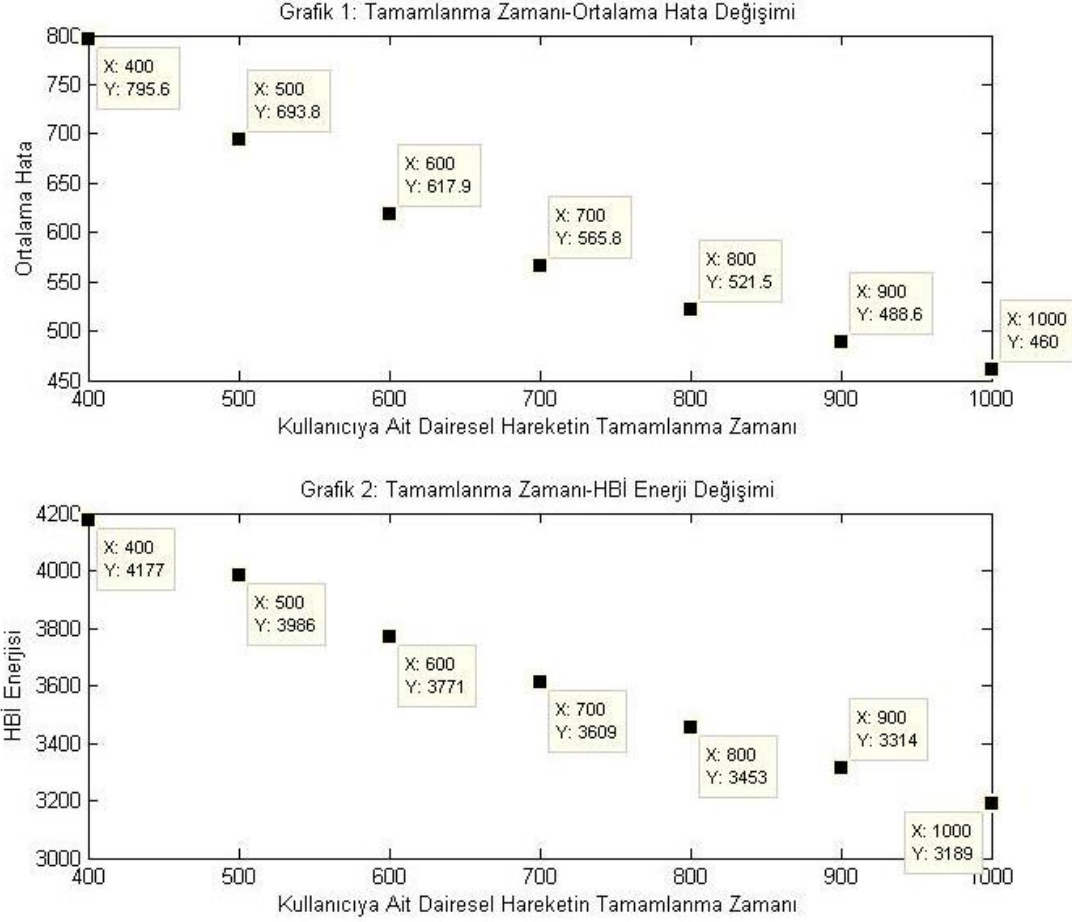


(b)

Şekil 3.10: Yarıçapın (r) 1000 olduğu dairesel hareketin tamamlanma süresi (bitişzamanı) (a) 10, (b) 100 için

Yerdeki kullanıcının dairesel hareketin tamamlanmasına yönelik zaman değeri artırılmış yani kullanıcının birim zamanda aldığı mesafe kısaltılarak takip performansı ve HBI enerjisindeki değişim 100 deneme için incelenmiştir. 4000×4000 m² harita üzerinde yarıçapı (r) 1000 m olan dairesel hareketin tamamlanma zamanı sırası ile

400,500,600,700,800,900,1000 olacak şekilde değiştirilmiştir. Senaryoda HBİ yüksekliği (h) 500 m, parçacık sayısı (L) 1000, HBİ ileri yönlü hız değeri (v) 40 km/h olacak şekilde belirlenmiştir.



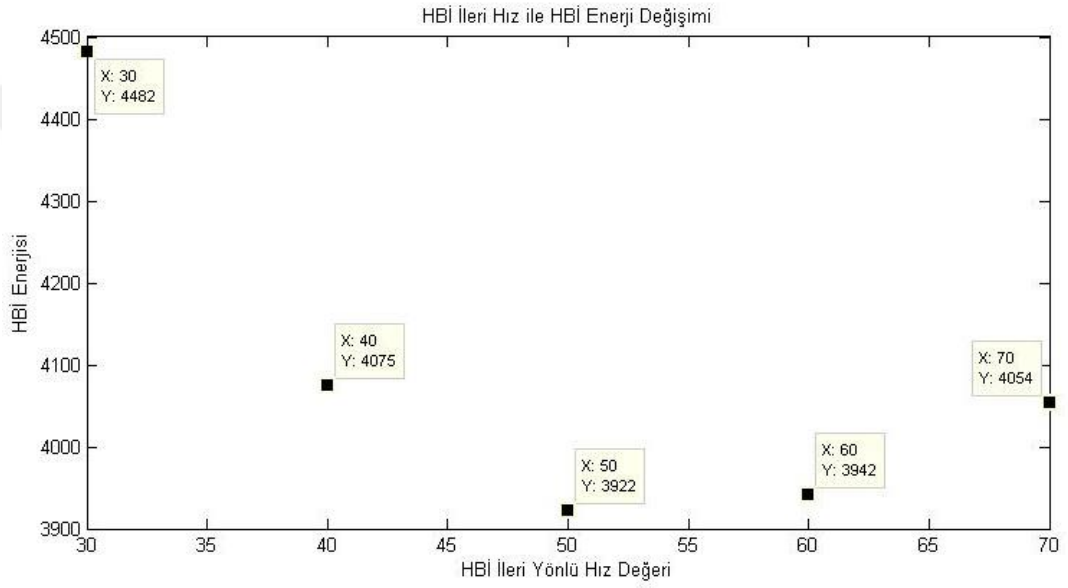
Şekil 3.11: Kullanıcıya ait Dairesel Hareketin Tamamlanma Zamanının Ortalama Hata ve HBİ Enerjisine (x1000) Etkisi

Şekil 3.11 ile verilen Grafik 1'de kullanıcı hareketine ait tamamlanma zamanı (*bitişzamanı*) arttıkça ortalama hata değeri azalarak fitrenin takip performansında artma meydana geldiği görülmüştür.

Düşük tamamlanma zamanı için HBİ'nin takibi gerçekleştirememesinin sebebi, yerde kullanıcıya ait ölçüm alınacak nokta sayısının az olmasından kaynaklanmaktadır. Parçacık filtresi ölçüm alamadan kullanıcının dairesel hareketi tamamlanmakta bu durumda HBİ ölçüm gerçekleştiremediği için de takibi gerçekleştirememektedir. Bu durum HBİ'nin takip etmeye çalışması kaynaklı çok fazla yer değiştirmesine sebep olmakta ve Şekil 3.10 ile verilen Grafik 2'de görüldüğü gibi ileri yönlü enerjisini arttırmaktadır.

3.3.1.4 HBİ İleri Yönlü Hızının ve Toplam HBİ Enerjisine Etkisi

HBİ enerji modeli, sabit kullanıcı konumunun belirlenmesine yönelik senaryolarda kullanılan model ile aynı olacak şekilde alınmıştır. HBİ enerjisinin, hareketli kullanıcı takibi senaryolarında sadece ileri yönlü hızdan oluştuğu önceki benzetimlerde belirtilmiştir. Bu benzetimde ise ileri yönlü hız değişiminin HBİ enerjisine etkisi 100 deneme için incelenmiştir. 4000x4000 m² harita üzerinde yarıçapı (r) 1000 m olan dairesel hareketin tamamlanma zamanı 500, HBİ yüksekliği (h) 500 m, parçacık sayısı (L) 1000, HBİ ileri yönlü hız değeri (v) ise 30 km/h, 40 km/h, 50 km/h , 60 km/h , 70 km/h olacak şekilde belirlenmiştir.



Şekil 3.12: HBİ İleri Yönlü Hız Değişiminin HBİ Enerjisine (x1000) Etkisi

Şekil 3.12 ile verildiği üzere ileri yönlü hız değeri 50 km/h olduğu durumda HBİ'nin harcadığı enerji değeri en düşük değerini almaktadır. Bunun sebebi HBİ için kullanılan enerji modelinden gelmektedir. Enerji modelinde en düşük ileri yönlü enerjinin 40 km/h ile elde edildiği belirtilmiştir fakat bu modelden farklı olarak asılı kalma süresi boyunca harcanan enerji değeri sıfır olarak alınmış ve HBİ'nin sürekli ölçüm yaptığı varsayılmıştır. Bu nedenle artı kalan enerjinin ileri yönlü hızda artış meydana getirmesi beklenen bir sonuçtur.

4 SONUÇ VE ÖNERİLER

Tez çalışması kapsamında Hareketli Baz İstasyonu aracılığı ile Parçacık Filtresi ve Alınan Sinyal Güç Göstergesi yöntemi kullanılarak sırası ile sabit bir kullanıcı ve hareketli bir kullanıcı için konum kestirimi gerçekleştirilip, hareketli kullanıcı için Hareketli Baz İstasyonunun kullanıcıyı takip etmesi sağlanmıştır. Parçacık Filtresi konum kestirimine ait algoritma oluşturularak, filtre için optimum parametrelerin belirlenmiştir. Takip algoritmasında ise yine filtre parametrelerinin konumlama hatasına etkisi incelenerek bir getiri-götürü analizi yapılmış ve hem sabit hem de hareketli kullanıcı konum kestirimi için optimum filtre parametreleri belirlenmiştir.

Hareketli Baz İstasyonlarının batarya ömrüne sahip cihazlar olması sebebi ile konum kestirimi ve takibin iyileştirilmesi çalışmalarında harcanan toplam enerji de incelenmiş, getiri-götürü analizine dahil edilmiştir.

Tez kapsamında yerdeki kullanıcıdan herhangi bir kontrol verisi filtreye girdi olmamaktadır. Bu nedenle kestirim hatası belirli bir değerden aşağıya düşürülememektedir. Özellikle takip algoritmasında kapalı çevrim bir kontrol döngüsü ile takip algoritmasına girdi yapılabilir ve böylece ortalama hata değeri iyileştirilebilir. Hareketli kullanıcıya ait gerçek kuzey ile yaptığı açı kontrol girdisinde kullanılabilir. Buna ek olarak, PF ile çözülen konumlama problemi üç boyutlu bir problem olarak ele alınsa da aslında HBI'nin yerden yüksekliği sabit olacak şekilde belirlenmiştir. Böylece algoritmanın çözümü aslında iki düzlemde yaptığı söylenebilir. Kısaca şehir dışında yer alan bölgelerde kullanıcıya ait konum belirlense ve takip gerçekleştirilse bile şehir içi veya yüksek binaların bulunduğu bir bölgede bu çözüm uygun olmayacaktır. HBI'nin yükseklik değeri değiştirilebilir şekilde belirlendiğinde çevresel parametrelere daha duyarlı bir filtre tasarımı yapılması mümkündür.

HBI'nin harcadığı toplam enerji hesabında sabit ve hareketli kullanıcılar için farklı enerji hesabı kullanılmış, sabit kullanıcıda HBI'nin asılı durarak ölçüm alma zamanı için harcadığı enerji hesaplanmış, hareketli kullanıcı durumunda ise sürekli ölçüm yaptığı varsayılmıştır. Sabit veya sürekli ölçümler yerine daha optimum bir çözüm olarak HBI'nin kestirim hatasını tutması ve belirli kestirim hatası üstünde yanlış ölçüm yaptığını anlaması sağlanabilir. Bu durumda daha fazla takip yapması engellenebilir ve daha fazla enerji harcaması önlenmiş olabilir. HBI'nin PF algortiması ile elde ettiği kestirimler için hareketli kullanıcının yerine yönelik ekstrapolasyon yapması

sağlanabilir. Böylece yüksek tahmin hatası aldığında, bir önceki en düşük tahmin hatası değeri için hesaplanan konuma ait ekstrapolasyon yapılması ve o noktaya kullanıcıdan önce giderek ve ölçüm alması sağlanabilir.

PF algoritması bu tez çalışması kapsamında tek kullanıcı senaryolu çözümlerde kullanılmıştır fakat PF'nin çoklu kullanıcıların yerlerinin belirlenmesi de gerçekleştirilebilecek çalışmalar arasındadır.



KAYNAKLAR

- [1] **Al-Hourani, A., Kandeepan, S., Jamalipour, A.,** (2014). Modeling air-to-ground path loss for low altitude platforms in urban environments, *Global Communications Conference (GLOBECOM) IEEE*, 2898–2904.
- [2] **Al-Hourani, A., Kandeepan S., Lardner S.,** (2014). Optimal lap altitude for maximum coverage, *IEEE Wireless Communications Letters*, 569–572, vol. 3, no. 6.
- [3] **Bor-Yaliniz, R. I., El-Keyi, A., Yanikomeroglu, H.,** (2016). Efficient 3-d placement of an aerial base station in next generation cellular networks, *IEEE International Conference in Communication*, 1–5.
- [4] **Alzenad, M., El-Keyi, A., Yanikomeroglu, H.,** (2017). 3d placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements, *IEEE Wireless Communications Letters*.
- [5] **Akarsu, A., Girici, T.,** (2017). Fairness aware multiple drone base station deployment, *IET Communications*, 12, 425-431.
- [6] **Akarsu, A., Girici, T.,** (2018). Failure aware deployment of drone base stations, *Telecommunication Forum*, 26th. IEEE, 1-5.
- [7] **Sallouha, A. C. H., Azari, M. M., Pollin, S.,** (2017). Aerial anchors positioning for reliable rss-based outdoor localization in urban environments, *IEEE Wireless Communications Letters*, 7, 376-379.
- [8] **Zanca, G. , Zorzi, F. , Zanella, A. , Zorzi, M.,** (2008). Experimental comparison of rssi-based localization algorithms for indoor wireless sensor networks, *in Proceedings of the workshop on Real-world wireless sensor networks*, 1–5.

- [9] **Caballero, I. M. F., Merino, L., Ollero, A.,** (2008). A particle filtering method for wireless sensor network localization with an aerial robot beacon, in *International Conference on Robotics and Automation.*, 596–601.
- [10] **Gordon, N.J., Salmond, D.J., Smith, A.F.M.,** (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEEE-Proceedings-F*, 140, 107–113.
- [11] **Arnaud, D., Nando de F., Neil G.,** (2001). Sequential Monte Carlo methods in practice, *Statistics for Engineering and Information Science.*
- [12] **Yu, G., Yu, F., Feng, L.,** (2007). A localization algorithm using a mobile anchor node under wireless channel, *International Conference on Robotics and Biomimetics*, 1104–1108.
- [13] **Filippone, A.,** Flight performance of fixed and rotary wing aircraft. Vol.1. (Sf. 150), Elsevier, (2006).
- [14] **Sallouha, H., Azari, M. M., Pollin, S.,** (2018). Energy-constrained UAV Trajectory Design for Ground Node Localization, *Global Telecommunications (GLOBECOM)*, Abu Dhabi, United Arab Emirates, United Arab Emirates,.9-13 Aralık.
- [15] **Zeng, Y., Zhang, R.,** (2017). Energy-efficient UAV Communication with Trajectory Optimization, *IEEE Transactions on Wireless Com.*, 16, 3747–3760.
- [16] **Al-Hourani, A., Kandeepan, S., Jamalipour, A.,** (2014). Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments, *IEEE Global Communications Conference*, Austin, Teksas, ABD, 8-12 Aralık.
- [17] **Thrun, S. Burgard, W.,** Probabilistic Robotics. Vol.1. (Sf. 85), MIT Press, (2005).
- [18] **Zeng, Y., Zhang, R.,** (2017). Energy-efficient UAV Communication with Trajectory Optimization, *IEEE Transactions on Wireless Com.*, 16, 3747–3760.

- [19] **Dariush, E., Sanaa, S., Pin-Han, H., Chadi, A.,** (2020). Autonomous UAV Trajectory for Localizing Ground Objects: A Reinforcement Learning Approach, *IEEE Transactions on Mobile Computing*, 1-1.
- [20] **Zanella, A., Sanaa, S., Pin-Han, H., Chadi, A.,** (2016). Best Practice in RSS Measurements and Ranging, *IEEE Communications Surveys & Tutorials*, 18, 2662 – 2686.
- [21] **Tsenka, S., Kerasiotis, F., Antonopoulos, C., Papadopoulos, G.,** (2016). Best RSS-based localization for wireless sensor networks in practice, *2014 9th International Symposium on Communication Systems, Networks & Digital Sign (CSNDSP)*, Manchester, İngiltere , 23-25 Haziran.





ÖZGEÇMİŞ

Ad-Soyad : Şerife Senem KARAMAN
Uyruğu : T.C.
Doğum Tarihi ve Yeri : 14.10.1984 , Ordu
E-posta : tosun.senem@gmail.com

ÖĞRENİM DURUMU:

- **Lisans** : 2012, TOBB Ekonomi ve Teknoloji Üniversitesi, Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği

MESLEKİ DENEYİM VE ÖDÜLLER:

Yıl	Yer	Görev
2012	Savunma Teknolojileri ve Ticaret A.Ş.Ankara	Sistem Mühendisi
2014	ROKETSAN A.Ş.Ankara	Sistem Mühendisi

YABANCI DİL: İngilizce

TEZDEN TÜRETİLEN YAYINLAR, SUNUMLAR VE PATENTLER:

- **Karaman, S. S., Akarsu, A., & Girici, T., 2019.** Use of Particle Filtering in RSSI-Based Localization by Drone Base Stations. *International Symposium on Networks, Computers and Communications (ISNCC)* (pp. 1-5). IEEE.