

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**GPS KONUM BELİRLEME ALGORİTMALARININ
UYGULANMASI**

**YÜKSEK LİSANS TEZİ
Müh. Saygın TOPATAN**

Anabilim Dalı: Elektronik ve Haberleşme Mühendisliği

Programı: Telekomünikasyon Mühendisliği

Tez Danışmanı: Doç.Dr. Selçuk PAKER

ARALIK 2008

**GPS KONUM BELİRLEME ALGORİTMALARININ
UYGULANMASI**

**YÜKSEK LİSANS TEZİ
Müh. Saygın TOPATAN
(504051331)**

**Tezin Enstitüye Verildiği Tarih : 3 Aralık 2008
Tezin Savunulduğu Tarih : 21 Ekim 2008**

Tez Danışmanı : Doç.Dr. Selçuk PAKER

Diğer Jüri Üyeleri Prof. Dr. Osman PALAMUTÇUOĞULLARI

Yrd. Doç.Dr. Berk ÜSTÜNDAĞ

ARALIK 2008

ÖNSÖZ

Bu tez çalışmam sürecinde bana gösterdikleri sabır ve verdikleri moral için meslektaşlarım Gültekin Erden ve Zekeriya Aydın'a, çalışmamda kullandığım kaynaklara erişimde yardımları için Erdem Aytaç'a, sevgili arkadaşlarım Kadir Uğur Mert ve Soner Soydinç'e, yüksek lisans eğitimim boyunca gösterdiği anlayış için çalıştığım şirket adına Altuğ B. Altıntaş'a, tüm hayatım boyunca sürekli desteklerini esirgemeyen ailem Emel Şenyaşar, Direnç Özgün Topatan ve Hayati Topatan'a, hayatıma güzel bir anlam kattığı için Ekin Aytaç'a, tüm hocalarıma en içten teşekkürlerimi sunarım.

Pratik yol göstericiliği, ilgisi, ve anlayışıyla bu tezi gerçekleştirmemi sağlayan Doç. Dr. Selçuk Parker'e çok teşekkür ederim.

Aralık 2008

Saygın Topatan
Elektronik ve Hab. Müh.

İÇİNDEKİLER

KISALTMALAR	iv
TABLO LİSTESİ	v
ŞEKİL LİSTESİ	vi
ÖZET	vii
SUMMARY	ix
1. GİRİŞ	1
1.1. GPS Sistemi ve Çalışmanın Amacı	1
2. GPS SİSTEMİ	3
2.1. GPS Sisteminin Bölümleri	3
2.1.1. Uzay bölümü	3
2.1.2. Kontrol bölümü	6
2.1.3. Kullanıcı bölümü	7
2.2. GPS İşareti	7
2.2.1. Navigasyon verisi	7
2.2.1.1. Telemetri ve el değiştirme kelimeleri	8
2.2.1.2. Navigasyon mesajındaki veri	8
2.2.2. Yayılım dizisi	9
2.3. GPS İşaret Yapısı	9
2.3.1. C/A kodu	10
2.3.2. Altın serinin üretilmesi:	11
3. HATA KAYNAKLARI	14
3.1. Uydu Saat Hatası	14
3.2. Yörünge Tahmin Hatası	14
3.3. Görelilik Etkileri	14
3.4. Atmosferik Etkiler	15
3.1.1. İyonosfer etkileri	15
3.1.2. Troposferik gecikme	15
3.5. Kullanıcı Gürültüsü ve Çözünürlük	16
3.6. Çok Yol Etkisi	17
4. GPS KONUM BELİRLEME ALGORİTMALARI	19
4.1. İteratif En Küçük Kareler Yöntemi	19
4.2. Diferansiyel GPS Konumlama Teknikleri	22

4.2.1. Koda dayalı algoritma	23
4.2.2. Taşıyıcı fazına bağlı algoritma	25
4.3. Genel Kinematik GPS Teknikleri	26
4.4. Kapalı Form GPS Konumlama Algoritması	28
4.4.1. Kapalı form çözüm	29
4.4.2. Kalman güncelleştirme algoritması	34
5. VERİ TOPLAMA VE TEST DÜZENİ	37
5.1. Toplanan Verinin İncelenmesi	38
5.1.1. GPS ham verisi ve kanalların izlenme sonuçları	38
5.1.2. Yörünge verileri	42
5.1.3. GPS uydu konumları	43
5.2. Toplanan Veriden Kullanıcı Konumunun Hesaplanması	44
6. SAYISAL SONUÇLAR VE DEĞERLENDİRME	47
KAYNAKLAR	49
ÖZGEÇMİŞ	50

KISALTMALAR

GPS	: Global Positioning System
SVN	: Space Vehicle Number
PRN	: Pseudo Random Noise
NAVSTAR	: Navigation System with Time and Ranging
UHF	: Ultra High Frequency
TLM	: Telemetry
HOV	: Handover
TOW	: Time of Week
C/A	: Coarse Acquisition
Mhz	: Mega Hertz
BPSK	: Binary Phase Shift Keying
DLL	: Delay Lock Loop
PLL	: Phase Lock Loop
ECEF	: Earth Centered, Earth Fixed
ILS	: Iterative Least Squares
DGPS	: Differential GPS

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 2.1 Blok I uyduları fırlatılma tarihleri	4
Tablo 2.2 Blok II uyduları fırlatılma tarihleri	5
Tablo 2.3 Blok IIA uyduları fırlatılma tarihleri	5
Tablo 2.4 Blok IIR uyduları fırlatılma tarihleri	5
Tablo 2.5 Blok IIR-M uyduları fırlatılma tarihleri	6
Tablo 2.6 C/A kodu faz atanması	13
Tablo 4.1 DGPS düzeltmesinden önce ve sonra tipik uydu hataları	24
Tablo 5.1 GPS uydularından elde edilen yörünge bilgileri	42
Tablo 5.2 GPS uydularından elde edilen yörünge bilgi açıklamaları	42
Tablo 5.3 Uyduların (PRN:2 – PRN:4) X,Y,Z koordinatları ve PR (Sözde Mesafe) Ölçümleri (m. 10^5)	43
Tablo 5.4 Uyduların (PRN:9 – PRN:12) X,Y,Z koordinatları ve PR (Sözde Mesafe) Ölçümleri (m. 10^5)	43
Tablo 5.5 Uyduların (PRN:15 – PRN:19) X,Y,Z koordinatları ve PR (Sözde Mesafe) Ölçümleri (m. 10^5)	43
Tablo 6.1 Kapalı form konumlama algoritması ve iteratif en küçük kareler yönteminin sonuçlarının karşılaştırılması	47

ŞEKİL LİSTESİ

	<u>Sayfa No</u>	
Şekil 2.1	GPS sisteminin bölümleri	3
Şekil 2.2	GPS uydularının yerleşimi	4
Şekil 2.3	Navigasyon veri yapısı	8
Şekil 2.4	GPS sinyalinin uyduda üretilmesi	9
Şekil 2.5	L1 işaretinin yapısı.	10
Şekil 2.6	Bir uydu için BPSK modüleli L1 taşıyıcı dalgası ve C/A kodu ve navigasyon verisi	11
Şekil 2.7	C/A kod üretici.	12
Şekil 3.1	Atmosfer katmanları	16
Şekil 3.2	Multipath etkisi .	17
Şekil 4.1	Genel DGPS uygulaması.	23
Şekil 5.1	Veri toplama amacıyla kullanılan GPS alıcısı	37
Şekil 5.2	SiGe 4110L fonksiyonel blok diyagramı .	38
Şekil 5.3	Ham GPS veri bilgisi	39
Şekil 5.4	PRN 2 GPS uydusu verileri izleme sonucu	40
Şekil 5.5	PRN 4 GPS uydusu verileri izleme sonucu	40
Şekil 5.6	PRN 9 GPS uydusu verileri izleme sonucu	41
Şekil 5.7	PRN 12 GPS uydusu verileri izleme sonucu	41

GPS KONUM BELİRLEME ALGORİTMALARININ UYGULANMASI

ÖZET

GPS kullanıcısının konumunu bulmak için günümüzde yaygın olarak kullanılan metod, sözde mesafe denklemlerini lineerleştirmek, ve kullanıcı tarafından sağlanan başlangıç noktası tahminini kullanarak, iteratif en küçük kareler (ILS) yöntemiyle kullanıcı konumunu hesaplamaya dayalıdır. Dünyaya yakın konumları belirlemede, dünyanın merkezinin iteratif algoritmaların başlangıç noktası olarak seçilmesi, iyi bir seçimdir; çözüm gerçek konuma yakınsar.

Son yıllarda, bu konuda yapılan çalışmalar, nonlinear GPS sözde mesafe ölçümlerini çözmek için, iteratif olmayan kapalı form çözümler önermektedir.

Bu çalışmanın amacı, kullanılan iteratif GPS konum belirleme algoritmalarından daha iyi başarımlar sunan algoritmaları incelemek, ve uygulamaktır. Bu algoritmalar, genel olarak, kullanılan iteratif çözümlerin yerine, kapalı form çözümler sunan algoritmalarlardır. Algoritmaların başarımlarının yapılabilmesi için bir test ortamı gerekir, ve bu test ortamı da ham GPS verisini yakalayıp, bilgisayarda ikili bir dosya olarak kaydedebilen bir anten sayesinde sağlanmıştır. Bu antenle örnek uydu verileri, çeşitli senaryolar (çok yol (multipath) hatalı, hareketli) gözetilerek toplanmıştır. Kapalı form algoritmaları uygulayabilmek için, anten ile toplanan veriyi işleyen, ve iteratif yolla konum bulan MATLAB kaynak kodu kullanılmıştır. Bu kod ile geliştirilen algoritmaların ihtiyaç duyduğu sözde mesafe ölçümleri elde edilmiş, çalışma kapsamında geliştirilen kod ile konum hesaplamaları yapılmıştır.

İncelenen kapalı form çözüm, ($n \geq 6$) uydunun varlığını gerektirmektedir, ve GPS probleminde 4 bilinmeyen çözülmesi gerektiği gerçeği göz önünde bulundurulduğunda, artık belirtilmiş (over determined) bir sistemle çalışmaktadır. Sözde mesafe ölçümleri, gürültülü kabul edilmektedir. Ölçümlerin stokastik tabiatı, GPS sözde

mesafe denklemlerine yansıtılmıştır, ve stokastik olarak modellenen ölçümlerle konum bulma, stokastik kestirim problemine dönüşmüştür. Doğru stokastik kestirim modeli ve kestirimi kullanıldığında kullanıcı konumu yanında ölçüm gürültü gücü kestirimi de sunan bir GPS çözümü elde edilmiştir.

IMPLEMENTATION OF GPS POSITION DETERMINATION ALGORITHMS

SUMMARY

The conventional method for determining GPS user position is linearizing the pseudorange measurements, and by using the user provided initial point, calculating the user position with Iterative Least Squares (ILS) method. Picking the earth center, as the initial point for iterative algorithm is a good choice, for near earth navigation; the solution converges to the real position.

In recent years, studies provide non iterative closed form solutions to the nonlinear pseudorange measurements.

The aim of this thesis, is analyzing and implementing the algorithms, offering a better performance, and more accurate results than the conventional iterative algorithms. These algorithms are providing closed form solutions, instead of iterative solutions. In order to test these algorithms, a test environment is required, and an antenna, which is capable of capturing raw GPS satellite data, and saving it as a binary file on a computer, is used. Example test data is captured with this antenna, regarding various scenarios (multipath error, movement). For implementing the closed form algorithms, a MATLAB source code is used, which processes captured data, calculates pseudorange measurements, and finds the user position iteratively. Closed form algorithms need pseudorange measurements, and these are obtained, using this code. Then the code is developed, to make position estimation calculations, using the closed form solution.

Examined closed form solution requires ($n \geq 6$) satellites visible to user, and considering the need to solve 4 unknowns in GPS problem, an overdetermined system is treated. Pseudorange measurements are recognized to be corrupted by noise. The stochastic nature of measurements is reflected to measurement equations, and estimating position with stochastically modeled measurements, the problem turns out to be a

stochastic estimation problem. Using the right stochastic estimation model, this results in a solution, which provides measurement noise intensity estimate, with user position.

1. GİRİŞ

1.1. GPS Sistemi ve Çalışmanın Amacı

GPS sistemi, herhangi bir yerde bulunan kullanıcının konumunu belirleyen, uydu tabanlı 3 boyutlu navigasyon sistemidir. Kullanıcıda bulunan GPS alıcısı, herhangi bir anda dünya çevresinde kendi yörüngelerinde bulunan 24 uydudan en az 4 tanesinin sürekli yaydığı sinyalleri alarak, sinyalin varış zamanına dayanan mesafe kestirimi yapar. Bu mesafe kestirimi sözde mesafe (pseudorange) olarak adlandırılır, çünkü alınan GPS işaretinde hatalar mevcuttur. Günümüzde kullanılan konum belirleme yöntemi, nonlinear sözde mesafe denklemlerini lineerleştirir ve önceden tanımlı başlangıç konum noktasından yararlanarak, iteratif yolla kullanıcı konumunu belirlemeye çalışır.

GPS sistemi ile alıcının konumunu belirleme işlemi, 4 bilinmeyenli bir denklemi çözmektir. Bu bilinmeyenler, kartezyen konum koordinatları x , y , z ve GPS saat hatasıdır. Bu çözüme ulaşmak için kullanıcı en az 4 tane uydunun görüşünde olmalıdır.

Bu çalışmanın amacı, kullanılan iteratif GPS konum belirleme algoritmalarından daha iyi başarımlar sunan algoritmaları incelemek ve uygulamaktır. Bu algoritmalar, genel olarak, kullanılan iteratif çözümlerin yerine, kapalı form çözümler sunan algoritmalarlardır. Algoritmaların başarımlarının testinin yapılabilmesi için bir test ortamı gerekir ve bu test ortamı da ham GPS verisini yakalayıp, bilgisayarda ikili bir dosya olarak kaydedebilen bir anten sayesinde sağlanmıştır. Bahsedilen algoritmaları uygulayabilmek için, anten ile toplanan veriyi işleyen ve iteratif yolla konum bulan MATLAB kaynak kodu [1] kullanılmıştır.

Çalışmanın ikinci bölümü, genel GPS sisteminin bölümlerini ve işleyişini incelemekte, test düzeneğinde kullanılan antenin ve kullanılan yazılımın topladığı verilerin işlenmesinde önemli olan GPS işaretinin yapısı, GPS Navigasyon verisi gibi konular incelenmiştir.

Çalışmanın üçüncü bölümde, GPS başarımını ve sözde mesafe ölçümlerini etkileyen atmosfer etkileri, saat hatası, multipath gibi hata kaynaklarına yer verilmiştir.

Çalışmanın dördüncü bölümünde, GPS Konum belirleme algoritmalarına yer verilmiştir. Tekil kullanıcı için düzenlenen iteratif metod, kullanıcı dışında önceden konumu bilinen bir referans alıcıya dayanan diferansiyel GPS sistemi, kullanıcı hızı ivmesi gibi belirlemelerde yüksek başarıma sahip genel kinematik teknikleri ve bu çalışmanın konusu olan kapalı form GPS konumlama algoritmaları açıklanmıştır.

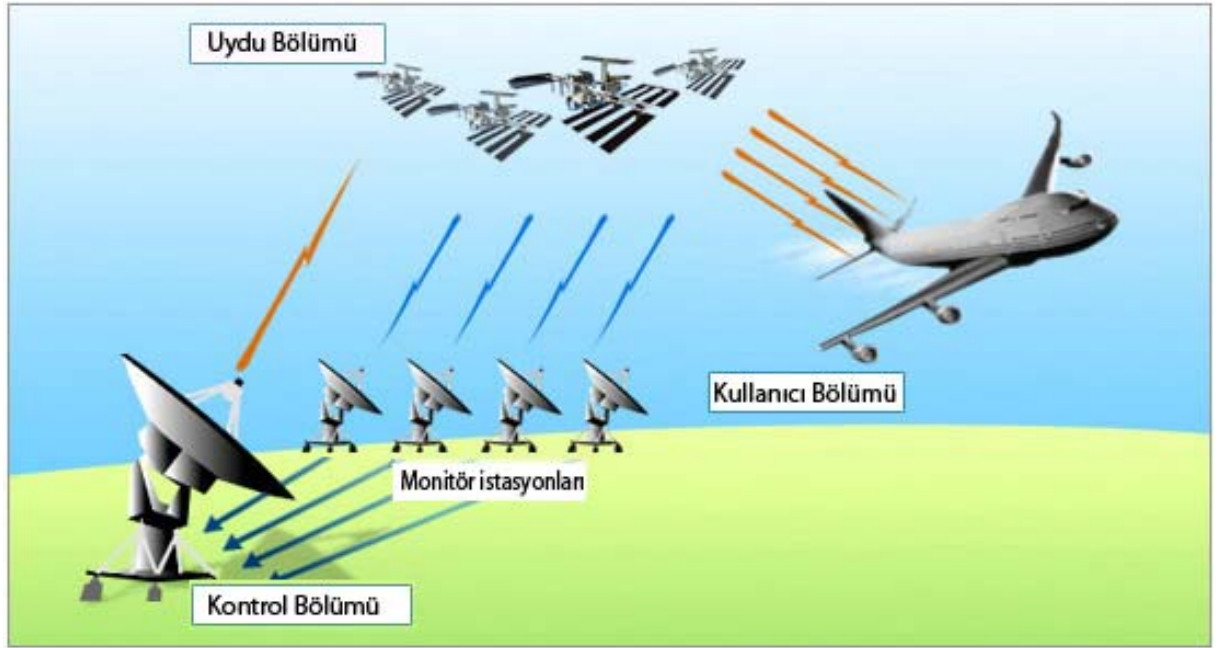
Çalışmanın beşinci bölümü, uygulanan GPS konum belirleme algoritmalarının test edilmesi için kurulan veri oluşturma düzeni açıklanmıştır. Farklı verilerle testi gerçekleştirmek için farklı hata kaynaklarının etkisindeki bölgelerde veri toplanmıştır.

Çalışmanın altıncı bölümünde, uygulanan kapalı form GPS konum belirleme algoritmasının sayısal sonuçları ve başarımı incelenmiştir.

2. GPS SİSTEMİ

2.1 GPS Sisteminin Bölümleri

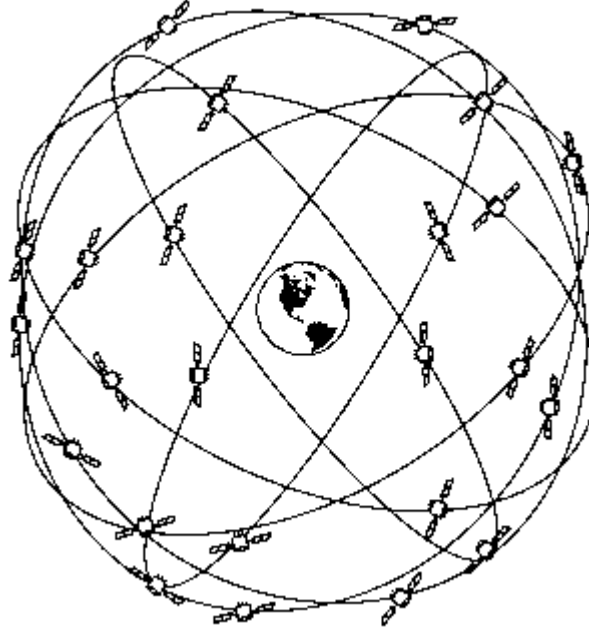
Uzay Bölümü, Kontrol Bölümü ve Kullanıcı Bölümü olmak üzere 3 ayrı bölümden oluşmaktadır.



Şekil 2.1 :GPS sisteminin bölümleri

2.1.1. Uzay bölümü

GPS uyduları yer yüzeyinden yaklaşık 20200 km. yükseklikte, 6 orbital yüzeyinde, her yüzeyde 4 uydunun olmak üzere, 24 tanedir. Yörüngeleri dairesel şekilde ekvator civarında birbirlerine eşit mesafede ve yine birbirlerine 60° lik açı ile yerleşirler. Bu şekilde, tüm dünyanın her anda 4 ile 8 arasında uydunun kapsama altına alınması sağlanır [2].



Şekil 2.2 : GPS uydularının yerleşimi

GPS uyduları, radyo alıcı-vericileri, atomik saatler, bilgisayarlar ve sistemi işletmek için çeşitli yardımcı ekipmanlar bulundurlar. Her uydu uzaydaki konumunu içeren mesaj yayınlr ve bundan faydalanarak kullanıcılar kendi konumunu belirleyebilirler.

GPS uydularının fırlatılma tarihlerine ve özelliklerine göre bir sınıflandırma yapılmıştır. Buna göre, 1978-1985 yılları arasında atılan uydular Blok I, 1989-1990 arasında atılanlar Blok II, 1990-1996 arası Blok IIA, 1997-2004 arasında fırlatılanlar Blok IIR, 2005'ten beri fırlatılanlar Blok IIR-M olarak sınıflandırılmaktadır. Tablo (2.1 - 2.5) bu uyduların fırlatılma tarihlerini ve SVN, PRN numaralarını göstermektedir.

Tablo 2.1 : Blok I uyduları fırlatılma tarihleri

SVN	Tarih	Uydu	SVN	Tarih	Uydu
01	1978-02-22	NAVSTAR 1	06	1980-04-26	NAVSTAR 6
02	1978-05-13	NAVSTAR 2	07	1981-14-09	NAVSTAR 7
03	1978-10-06	NAVSTAR 3	08	1983-07-14	NAVSTAR 8
04	1978-04-01	NAVSTAR 4	09	1984-06-13	NAVSTAR 9
05	1980-02-09	NAVSTAR 5	10	1984-09-08	NAVSTAR 10
11	1985-10-09	NAVSTAR 11			

Tablo 2.2: Blok II uyduları fırlatılma tarihleri

SVN	Tarih	Uydu	SVN	Tarih	Uydu
14	1989-04-05	NAVSTAR II-1	17	1989-10-02	NAVSTAR II-5
13	1989-06-10	NAVSTAR II-2	18	1990-01-24	NAVSTAR II-6
16	1989-08-18	NAVSTAR II-3	20	1990-03-26	NAVSTAR II-7
19	1989-10-21	NAVSTAR II-4	21	1990-08-02	NAVSTAR II-8
17	1989-10-02	NAVSTAR II-5			

Tablo 2.3: Blok IIA uyduları fırlatılma tarihleri

SVN	Tarih	Uydu	PRN	SVN	Tarih	Uydu	PRN
23	1990-11-26	GPS IIA-1	32	31	1993-03-29	GPS IIA-10	
24	1991-07-04	GPS IIA-2	24	37	1993-05-13	GPS IIA-11	
25	1992-02-23	GPS IIA-3	25	39	1993-06-26	GPS IIA-12	09
28	1992-04-10	GPS IIA-4		35	1993-08-30	GPS IIA-13	05
26	1992-07-07	GPS IIA-5	26	34	1993-10-28	GPS IIA-14	04
27	1992-09-09	GPS IIA-6	27	36	1994-03-10	GPS IIA-15	06
32	1992-11-22	GPS IIA-7		33	1996-03-28	GPS IIA-16	03
29	1992-14.49	GPS IIA-8		40	1996-07-16	GPS IIA-17	10
22	1993-02-03	GPS IIA-9		30	1996-09-12	GPS IIA-18	30

Tablo 2.4: Blok IIR uyduları fırlatılma tarihleri

SVN	Tarih	Uydu	PRN	SVN	Tarih	Uydu	SVN	PRN
42	1997-01-17	GPS IIR-1		42	2003-01-29	GPS IIR-8	56	16
43	1997-07-23	GPS IIR-2	13	43	2003-03-31	GPS IIR-9	45	21
46	1999-10-07	GPS IIR-3	11	46	2003-12-21	GPS IIR-10	47	22
51	2000-05-11	GPS IIR-4	20	51	2004-03-20	GPS IIR-11	59	19
44	2000-07-16	GPS IIR-5	28	44	2004-06-23	GPS IIR-12	60	23
41	2000-11-10	GPS IIR-6	14	41	2004-11-06	GPS IIR-13	61	02

Tablo 2.5: Blok IIR-M uyduları fırlatılma tarihleri

Tarih	Uydu	SVN	PRN
2005-09-26	GPS IIR-M-1	53	17
2006-09-25	GPS IIR-M-2	52	31
2006-11-17	GPS IIR-M-3	58	12
2007-10-17	GPS IIR-M-4	55	15
2007-14.52	GPS IIR-M-5	57	29
2008-03-15	GPS IIR-M-6	48	07

2.1.2. Kontrol bölümü

Operasyonel Kontrol Sistem, ana kontrol merkezi, monitor istasyonları ve yer kontrol istasyonlarından oluşur. Bu sistemin görevi uyduların yörünge tahmini ve zaman tayini için izini sürmek ve uydulara veri mesajı yüklemektir. Üç bölümden oluşmaktadır, bunlar ana kontrol istasyonu, monitör istasyonu ve yer kontrol istasyonudur.

Ana Kontrol İstasyonu, monitör istasyonlarından veri toplayarak, uydu yörüngelerini ve saat parametrelerini hesaplar. Bu sonuçları yer istasyonlarına uydulara yüklenmesi amacıyla gönderir. Uydu kontrolü ve sistem işletimi de bu istasyonun sorumluluğudur.

Monitör İstasyonu, dakik atomik saat standardı sağlar ve görünürdeki tüm uyduların uzaklıklarını belirler. Bu uzaklıklar, her 1,5 saniyede bir hesaplanır, iyonosferik ve meteorolojik veri ile düzeltilip 15 dakikalık veri oluşturulurularak ana kontrol istasyonuna gönderilir[2].

Yer Kontrol İstasyonu, bir çok yer anteni barındırırlar, uydular ile haberleşme linkleri teşkil ederler. Ana Kontrol istasyonlarında hesaplanan uydu konumları, saat bilgileri GPS uydularına S bandı kullanılarak yüklenir. Eğer bir yer kontrol istasyonu deaktive olursa, uydular önceden saklanmış navigasyon verileri ile konum tahmini yapabilirler[3].

2.1.3. Kullanıcı bölümü

GPS alıcısıdır. L bandından aldığı uydu sinyalinin demodüle eder, işler ve uyguladığı düzeltmelerle uyduların mesafe kestirimini yapar. Bu mesafeleri kullanarak, kendi konumunu, hızını belirler.

2.2. GPS İşareti

GPS İşareti UHF bandında (500 Mhz – 3 Ghz) iki frekansta taşınır. Bu frekanslar, temel frekans olan

$f_0 = 10,23\text{MHz}$ den türetilmiştir.

$$f_{L1} = 154f_0 = 1575.42\text{MHz} \quad (2.1)$$

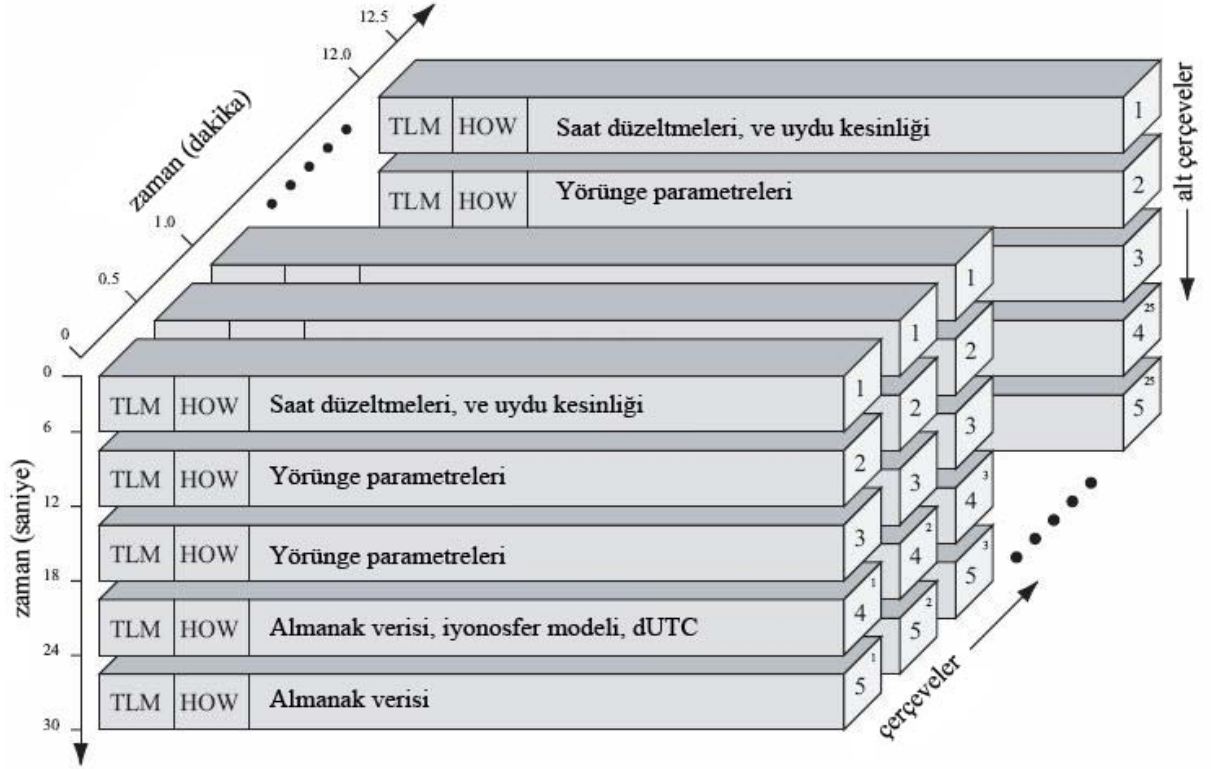
$$f_{L2} = 120f_0 = 1227.60\text{MHz} \quad (2.2)$$

GPS İşaretinin, taşıyıcı dışında navigasyon verisi ve yayılım dizisi kısımları vardır.

2.2.1. Navigasyon Verisi

Navigasyon verisi uydu yörüngesi hakkında bilgi içerir. Bu bilgi tüm uydulara yer istasyonunda GPS Kontrol Bölümünde yüklenmiştir. Navigasyon verisi L1 frekansında iletilir ve 50 bps.'lık veri hızına sahiptir.

Navigasyon verisi, 1500 bit uzunluğunda büyük bir çerçevenin içerisinde 5 tane 300 er bit uzunluğunda altçerçeve den oluşur. Her altçerçeve, 30 ar bitlik 10 kelime içerir. 1, 2 ve 3 numaralı altçerçeveler, her çerçevede tekrar edilir. 4 ve 5 numaralı altçerçevelerin aynı yapıda fakat farklı veriden oluşan 25 farklı versiyonu vardır. 50 bps. lik veri hızıyla bir altçerçevenin iletimi 6 saniye, bir çerçevenin iletimi 30 saniye ve bütün navigasyon mesajının iletimi 12,5 dakika sürmektedir[1].



Şekil 2.3: Navigasyon veri yapısı

2.2.1.1. Telemetri ve el değiştirme kelimeleri

10 kelimelik altçerçeveler, her zaman iki özel kelime çifti ile başlar: Telemetri (TLM) ve El Değiştirme Kelimesi (Handover, HOV).

TLM, her altçerçevenin ilk kelimesidir ve 6 saniyede bir tekrar edilir. Çerçeve senkronizasyonu için kullanılan 8 bitlik başlık ardından 16 tane rezerve bit ve parite bitleri vardır.

HOV kelimesi, 17 bitlik haftanın zamanı (TOW) bilgisi içerir. İki bayrak, kullanıcıya antispoofing gibi bilgiler sağlar ve diğer 3 bit altçerçevenin, o çerçevedeki 5 altçerçeveden hangisi olduğu bilgisi, kimliğidir.

2.2.1.2. Navigasyon mesajındaki veri

TLM ve HOV kelimeleri dışında her altçerçevede 8 ayrı veri kelimesi daha yer alır.

1. Altçerçeve: İlk altçerçeve saat bilgisi taşır. Bu bilgi alıcı tarafında uydu ile arasındaki mesafeyi hesaplamakta kullanılan, mesajın ne zaman yayımlandığı bilgisini taşır. Ayrıca bu altçerçevede, verinin güvenilir olup olmadığını bildiren veri bulunur.

2. ve 3. Altçerçeve: Uydu yörünge verisi taşınır. Yörünge verisi uydu yörüngesiyle ilgilidir ve uydu konumunu belirlemek için kullanılır.

4. ve 5. Altçerçeve: Bu iki altçerçeve, her 12,5 dakikada bir tekrarlanır ve almanak verisi içerir. Bu almanak verisi, yörünge ve düşük duyarlıklı zaman verilerinden oluşur. Ayrıca, her uydu kendisi için sadece yörünge verisi gönderirken, almanak verisini tüm GPS uyduları için gönderir.

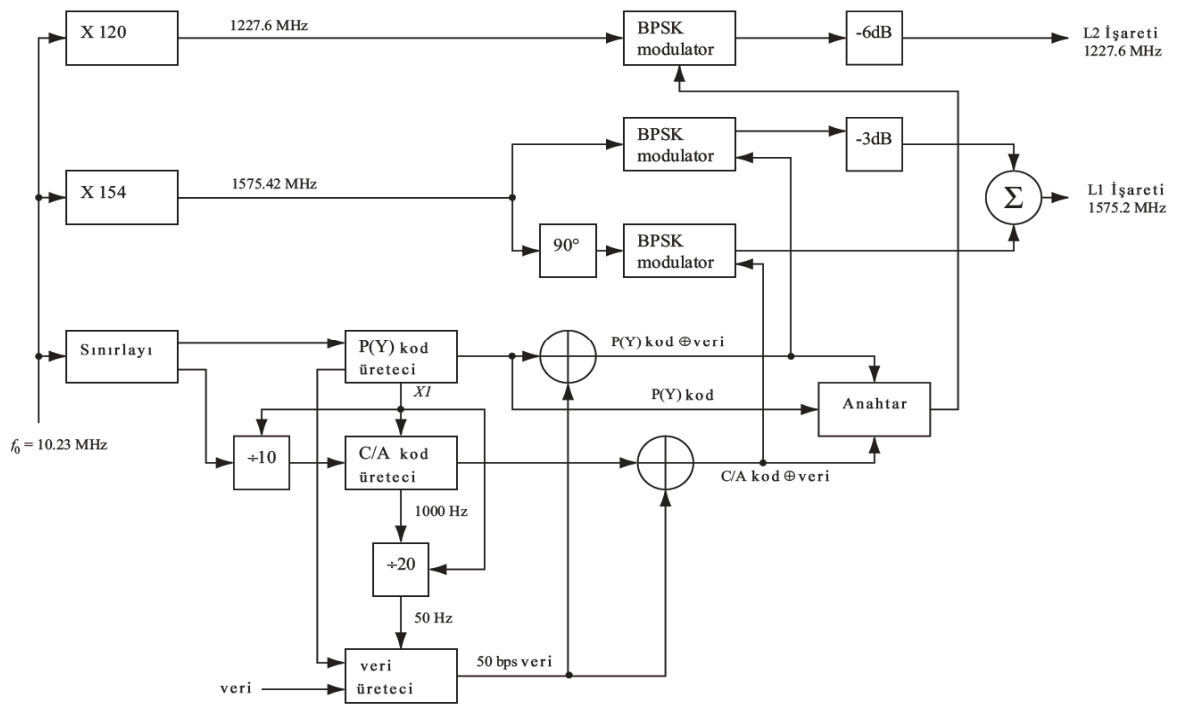
2.2.2 Yayılım dizisi

Her uydunun iki eşsiz yayılım dizisi (spreading sequence) veya kodu vardır.

Coarse acquisition code (C/A): 1023 çiplik bir dizidir. Burada çip, bit yerine geçen bir deyimdir, kodun bilgi taşımadığını vurgulamak için kullanılır. Her ms. de bir kez kod tekrarlanır, dolayısıyla 1,023 MHz lik çip hızı vardır.

Encrypted Precision Code (P(Y)): Her GPS haftasının (cumartesiye pazara bağlayan gece) başında tekrarlanır. Yaklaşık $2,35 \cdot 10^4$ çip uzunluğunda bir koddur. Askeri amaçlarla kullanılır ve standart konumlama servisinin bir parçası değildir.

2.3 GPS İşaret Yapısı



Şekil 2.4: GPS Sinyalinin uyduda üretilmesi

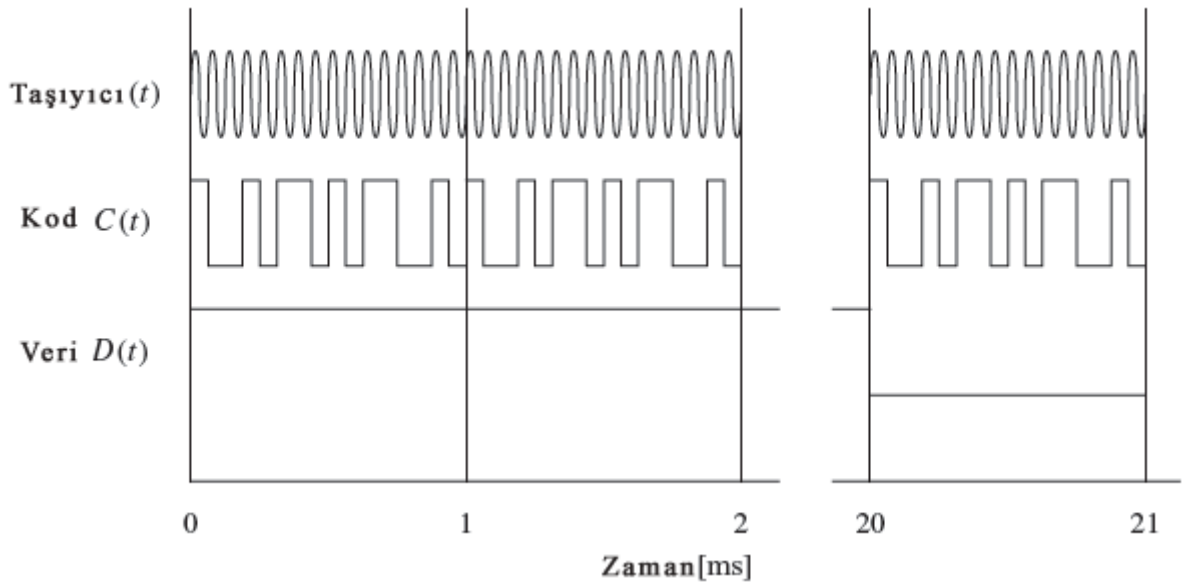
Şekil 2.4’de GPS sinyali oluşturulurken, 10,23 Mhz frekanslı saat işareti kullanılır. Sırasıyla 154 ve 120 ile çarpılarak, L1 ve L2 taşıyıcı işaretleri oluşturulur. En alta görülen veri jeneratörü, navigasyon verisini üretir. Veri jeneratörü ve kod jeneratörü, P(Y) kod jeneratörü tarafından sağlanan X_1 işareti sayesinde senkronize olurlar. Kod üretildikten sonra navigasyon verisiyle modulo 2 toplayıcılar üzerinden toplanırlar.

Navigasyon verisi ile C/A kodu ve P(Y) kodunun toplanmasıyla oluşan işaretler BPSK ile modüle edilirler. Burada iki kod arasında 90° lik faz farkı vardır. Uydu k ‘dan iletilen veri şu şekilde gösterilebilir[1]:

$$s^k(t) = \sqrt{2P_C}(C^k(t) \oplus D^k(t)) \cos(2\pi f_{L1}t) + \sqrt{2P_{PL1}}(P^k(t) \oplus D^k(t)) \sin(2\pi f_{L1}t) + \sqrt{2P_{PL2}}(P^k(t) \oplus D^k(t)) \sin(2\pi f_{L2}t) \quad (2.3)$$

Burada P_C , P_{L1} ve P_{L2} C/A ve P kodlu sinyallerin gücü, C^k k. uyduya atanan kod dizisi, P^k k. uyduya atanan P(Y) kod dizisi D^k ise navigasyon veri dizisi, f_{L1} ve f_{L2} L1 ve L2 nin taşıyıcı frekanslarıdır.

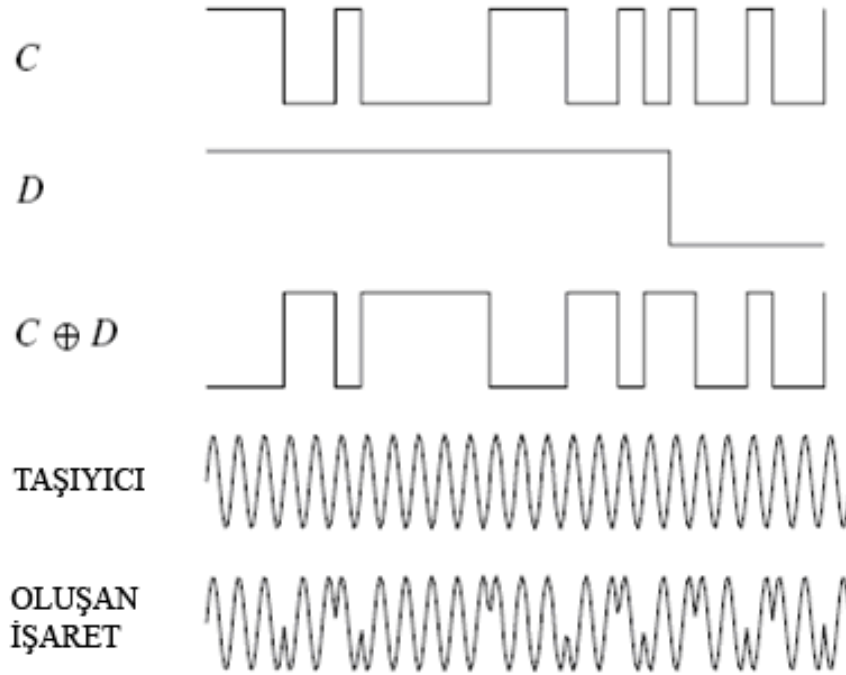
Şekil 2.5 de L1 frekansındaki sinyalin 3 parçası görülüyor. C/A kodu her milisaniyede bir kendini tekrar eder ve navigasyon biti 20 ms. sürer. Yani, her navigasyon biti için işaret 20 tam C/A kodu içerir[1].



Şekil 2.5 :L1 İşaretinin yapısı

Şekil 2.5’de Altın Kod C, navigasyon verisi D , $C \oplus D$ (modulo 2 toplanmış hali) ve taşıyıcı işaret görülmektedir. Son işaret BPSK ile, taşıyıcının fazının çip değişim

anında 180° döndürülmesi ile, elde edilir. Navigasyon verisinde bit geçişi olduğunda da, Şekil 2.6'dan gözlenebileceği üzere, sonuç işaretinin fazı yine 180° döner.



Şekil 2.6 : Bir uydu için BPSK modüleli L1 taşıyıcı dalgası ile C/A kodu ve navigasyon verisi

2.3.1. C/A kodu

C/A kod dizileri, Gold Codes (Gold, 1967) ailesine üye kodlardır. Bunlar ayrıca sözde rastgele gürültü dizileri (Pseudo Random Noise Sequences, PRN) olarak da anılırlar.

Uydular tarafından iletilen PRN kodları benzer gürültü özelliklerinde deterministik dizilerdir. Her C/A kodu LFSR (Linear Feedback Shift Register) ile üretilirler. LFSR $N = 2^n - 1$ elemanlı, maksimum uzunluklu dizi üretir. Gold code ise iki maksimum uzunluklu dizinin toplamıdır. GPS C/A'nın kullandığı $n=10$ olup, $p(t)$ dizisi her milisaniyede tekrar ettiğinden çip uzunluğu $1\text{ms} / 1023 = 977,5 \text{ ns} \approx 1\mu\text{s}$ olur, boşluk veya havada yayılırken 300 m.lik bir uzunluk demektir. C/A kodunun otokorelasyon fonksiyonu şu şekilde yazılabilir:

$$r_p(\tau) = \frac{1}{NT_c} \int_0^{NT_c} p(t)p(t+\tau)dt \quad (2.4)$$

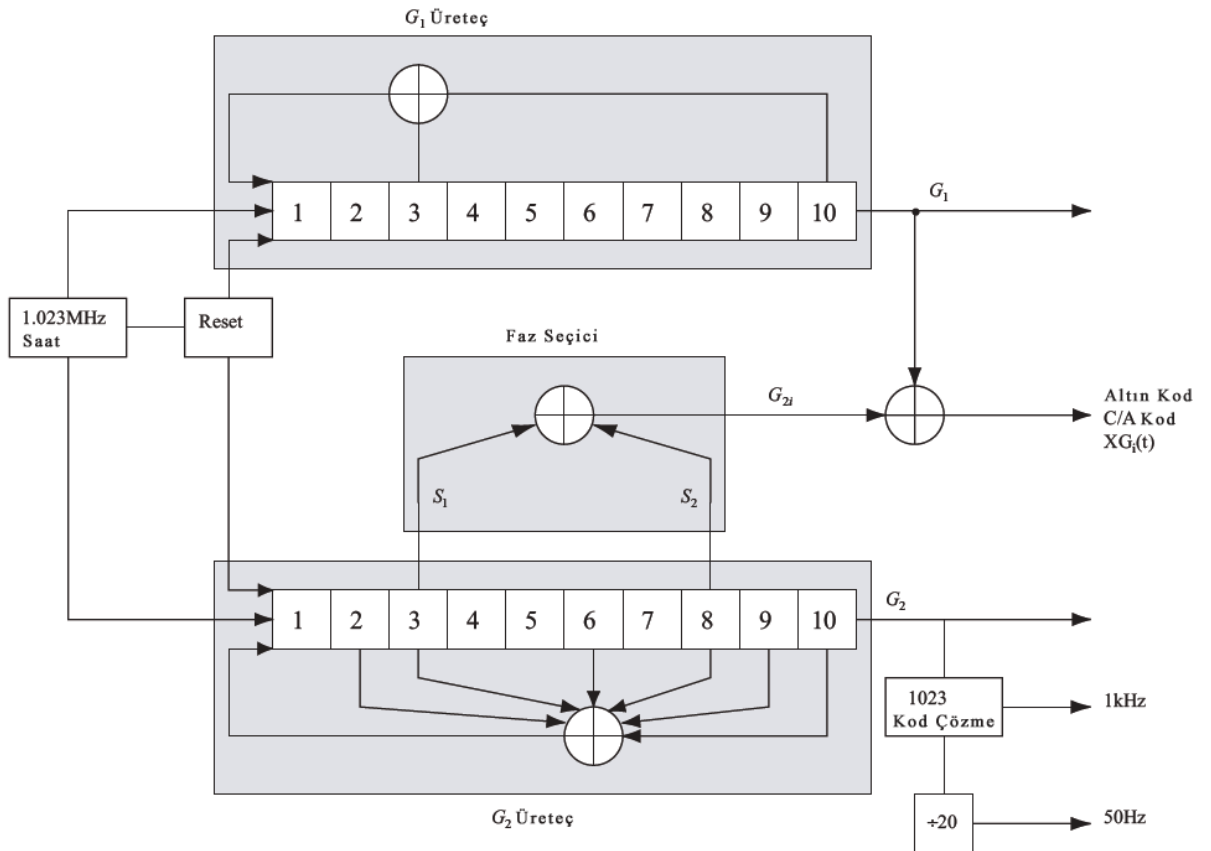
Seride rastgele dağılmış 512 tane 1 , 511 tane de 0 olmalıdır. Bu şekilde oluşmuş bir dizi tamamen deterministiktir, yani bu seri rastgele değil, sözde rastgele (pseudorandom) dir. Korelasyon aralığının dışında $p(t)$ ' nin Otokorelasyon fonksiyonu $-1/N$ olmaktadır. C/A kodu için sabit terim $-1/N = -1/1023$ dür.

2.3.2. Altın serinin üretilmesi:

C/A kod üretici G_1 ve G_2 olarak anılan iki tane kaydırmalı kaydediciye (shift register) sahiptir. Her birinin 10 hücresi vardır ve 1023' lük dizi üretirler. Üretilen iki dizi birbiriyle mod-2 de toplanır ve 1023 çiplik C/A kodu üretilir. Her 1023 üncü periyotta, kaydırmalı kaydediciler 1' lenerek ilk durumuna döndürülürler ve kod yeniden başlar. G_1 ve G_2 'nin mimarilerini açıklayan polinomlar:

$$G_1 = 1 + X^3 + X^{10} \quad (2.5)$$

$$G_2 = 1 + X^2 + X^3 + X^6 + X^8 + X^9 + X^{10} \quad (2.6)$$



Şekil 2.7: C/A kod üretici

Uyduların C/A kodlarını farklı üretebilmek için, iki kaydırmalı kaydedicinin çıktıları özel bir usulde birleştirilir. G1 kaydedicisi, çıktısını her zaman değişikliğe uğramadan sunar; G2 kaydedicisi iki durumunu mod 2 toplayıcıdan geçirerek çıktıyı oluşturur. Mod 2 toplayıcıya gidecek durumların (state) seçimi faz seçimi adını alır. Tablo 2.6 da her C/A kodu için faz seçim kombinasyonları görülmektedir.

Tablo 2.6: C/A kodu faz atanması

Uydu Kimlik Numarası	GPS PRN İşaret Numarası	Kod Faz Seçimi G2	Kod Gecikme Çipleri	ilk Oktal 10 çip
1	1	$2 \oplus 6$	5	1440
2	2	$3 \oplus 7$	6	1620
3	3	$4 \oplus 8$	7	1710
4	4	$5 \oplus 9$	8	1744
5	5	$1 \oplus 9$	17	1133
6	6	$2 \oplus 10$	18	1455
7	7	$1 \oplus 8$	139	1131
8	8	$2 \oplus 9$	140	1454
9	9	$3 \oplus 10$	141	1626
10	10	$2 \oplus 3$	251	1504
11	11	$3 \oplus 4$	252	1642
12	12	$5 \oplus 6$	254	1750
13	13	$6 \oplus 7$	255	1764
14	14	$7 \oplus 8$	256	1772
15	15	$8 \oplus 9$	257	1775
16	16	$9 \oplus 10$	258	1776
17	17	$1 \oplus 4$	469	1156
18	18	$2 \oplus 5$	470	1467
19	19	$3 \oplus 6$	471	1633
20	20	$4 \oplus 7$	472	1715
21	21	$5 \oplus 8$	473	1746
22	22	$6 \oplus 9$	474	1763
23	23	$1 \oplus 3$	509	1063
24	24	$4 \oplus 6$	512	1706
25	25	$5 \oplus 7$	513	1743
26	26	$6 \oplus 8$	514	1761
27	27	$7 \oplus 9$	515	1770
28	28	$8 \oplus 10$	516	1774
29	29	$1 \oplus 6$	859	1127
30	30	$2 \oplus 7$	860	1453
31	31	$3 \oplus 8$	861	1625
32	32	$4 \oplus 9$	862	1712
—	33	$5 \oplus 10$	863	1745
—	34	$4 \oplus 10$	950	1713
—	35	$1 \oplus 7$	947	1134
—	36	$2 \oplus 8$	948	1456
—	37	$4 \oplus 10$	950	1713

3.HATA KAYNAKLARI

GPS sisteminde alıcının konumunu veya hızını doğru belirlemesi, GPS sistem zamanına senkronize olması, çeşitli faktörlerin karmaşık etkileşimlerine bağlıdır.

3.1. Uydu Saat Hatası

GPS uydularında yayınlanan işaret üretimi dahil, tüm araçüstü zamanlama operasyonlarını idare eden atomik saat vardır. Bu saatler son derece istikrarlı olmalarına rağmen 1 msn. ye kadar GPS zamanından sapabilir [3]. Bu 1 msn.lik sapma 300 km.lik sözde mesafe hatasına neden olabilir. Ana Kontrol İstasyonu bu sapmayı belirler ve navigasyon mesajının tekrar yayınlanması için saat düzeltme parametreleri iletir.

3.2.Yörünge Tahmin Hatası

Tüm uydular için, navigasyon mesajında diğer parametrelerle birlikte yayınlanması için, optimal yörünge tahminleri hesaplanarak uydulara yüklenir. Bu hesaplamalar, rezidüel hata içerir[3].

3.3.Görelilik Etkileri

Einstein'ın görelilik kuramı, sözde mesafe ölçümünde etkindir. İşaret kaynağı (GPS uydusu) veya işaret alıcısı (kullanıcı) hareketliyken, özel görelilik kuramı; işaret kaynağı ve işaret alıcısı farklı yerçekimi ortamlarında olduğunda genel görelilik kuramı dikkate alınmalıdır[2].

Uydu saati, her iki görelilik kuramından da etkilenmektedir. Bu etkiyi gidermek amacıyla uydu saati frekansı, 10,2299999545 MHz. e ayarlanmıştır. Deniz seviyesindeki kullanıcı tarafında gözlenen frekans hiç bir düzeltmeye gerek kalmaksızın 10,23 Mhz. olur [3].

Dünyanın çekim alanı, uydu yörüngesinde göreliliğe dayalı olarak düzensizliğe neden olur[2].

Sinyal iletimi esnasında, dünyanın dönmesine bağlı olarak Sagnac etkisi olarak bilinen göreliliğe dayalı bir etki oluşur [3]. Uydu işaretinin iletiminde propagasyon zamanına bağlı olarak, yerküredeki saat, referans noktaya göre sınırlı rotasyona uğrar.

Son olarak, GPS işareti dünyanın çekim alanına bağlı olarak uzay zaman kıvrılmasına uğrar.

3.4. Atmosferik Etkiler

Bir dalganın, bir ortamdaki propogasyon hızı, ortamın kırılma katsayısı ile ifade edilir. Eğer ortamda, dalganın propagasyon hızı dalganın frekansının bir fonksiyonu ile ifade edilirse, o ortam için saçıcıdır denir. Saçıcı bir ortamda, işaretin taşıyıcı fazının hızı, işaret bilgisini taşıyan dalganınkinden farklıdır. GPS dalgası atmosferde ilerlerken, iyonosfer ve troposferden de geçer, ve bu ortamların saçıcılık özellikleri farklı olduğundan, işarete olan etkileri farklıdır.

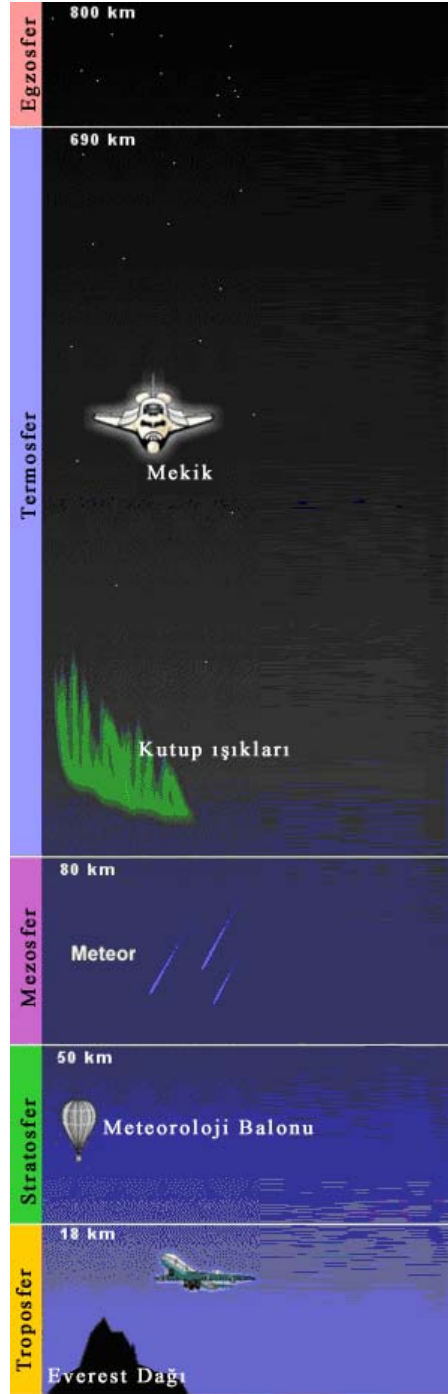
3.4.1. İyonosfer etkileri

İyonosfer, yerküreden 70 km. ile 1000 km. yükseklikte olan saçıcı bir ortamdır. Bu bölgede, güneşten gelen ultraviyole ışınlar iyonlaşarak gaz molekülü parçaları oluştur ve serbest elektronlar salınır. Bu serbest elektronlar, yayılan GPS işaretleri de dahil olmak üzere, tüm elektromanyetik dalga propagasyonunu etkiler [3]. İyonosfer etkisi, iyonosfer saçıcı bir ortam olduğundan, frekans bağımlıdır. GPS sistemi birkaç farklı frekans üzerinden çalışacak şekilde tasarlanmıştır ve bu özellik kullanılarak, iyonosfer etkileri ölçülebilir ve düzeltilebilir[4].

3.4.2. Troposferik gecikme

Troposfer, atmosferin en alt kısmıdır. İyonosferin aksine troposfer, 15 Ghz. e kadar olan, yani GPS taşıyıcı frekansları için saçıcı olmayan bir ortamdır. GPS işareti üzerindeki troposfer etkileri, çalışma frekansından bağımsızdır. Elektromanyetik işaretler, troposferdeki nötr atom ve moleküllerden etkilenir. Bu etkilere, troposferik gecikme veya troposferik kırılma denir. Troposferik gecikme, zenith yönünde yaklaşık 2 m.dir, ancak zenith açısının artmasıyla artar. Birkaç derecelik düşük uydu elevasyonu durumunda GPS işaretinin troposferik gecikmesi birkaç metreden fazla olabilir[4].

Troposferik etkiler, sıcaklık, basınç, nem oranı, GPS anteninin konumu gibi etkenlere bağlıdır.



Şekil 3.1: Atmosfer katmanları

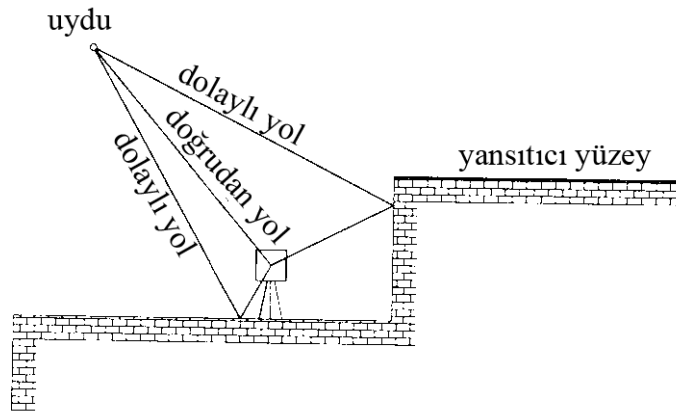
3.5. Kullanıcı Gürültüsü ve Çözünürlük

Ölçüm hataları, alıcıdaki izleme döngülerinden de kaynaklanabilir. Gecikme kilit döngüsü (DLL, Delay Lock Loop) bakımından, sözde mesafe ölçüm hatalarından

baskın olanları, termal gürültü jitteri ve dinamik stres hatasıdır[3]. İkincil önemli hata kaynakları, donanım ve yazılım çözünürlüğü ve osilatör stabilitesidir.

3.6. Çok yol Etkisi (Multipath)

Alıcı ölçümlerindeki en büyük hata kaynaklarından birisi de çok yol etkisidir. İşaretin, alıcıya çevredeki nesnelere, veya yer üzerinden yansıtılarak birden fazla yol üzerinden ulaşması bu etkiyi doğurur.



Şekil 3.2: Multipath etkisi

Şekil 3.1.' de görüldüğü gibi, sinyal alıcıya bir doğrudan ve iki tane dolaylı yoldan, ulaşmaktadır. Bunun sonucu olarak, alınan işaretlerin faz farkları, katettikleri yol uzunluklarıyla doğru orantılı olacaktır. Çok farklı geometrik durumlar olduğundan dolayı, multipath etkisinin genel bir modeli yoktur; ama multipath etkisini tahmin edebilmek için, L1 Ve L2 kodlarının kombinasyonu ve taşıyıcı faz ölçümleri kullanılır. Burada temel prensip, troposfer, saat hataları ve görelilik etkilerinin kodu ve taşıyıcı işareti aynı miktarda etkileyeceği gerçeğidir. Bu, elbette frekansa bağlı olan iyonosfer hatası için ve multipath etkisi için geçerli sayılmaz [2].

Tranquilla ve Carr (1990/91) sözde mesafe ölçümündeki multipath hatalarını 3 sınıfa ayırmışlardır:

- Yaygın olarak dağıtılmış alandan saçılarak yayılması : Örneğin işaretin metalik bir ortamdan geçmesi.

- Antenin civarındaki tanımlı nesnelere veya yansıtıcı yüzeylerden yansıması.
- Çok düşük frekanstaki dalgalanma: Genelde su yüzeyinden yansıma için kullanılır.

Geometriye bağılı olarak söylenebilir ki, düşük elevasyonlu uydulardan gelen işaretler, yüksek elevasyonlu uydulardan gelen işaretlere göre multipath etkisine daha hassastırlar [3].

4. GPS KONUM BELİRLEME ALGORİTMALARI

Kullanıcı konumunu 3 boyutta (x_u, y_u, z_u) ve t_u (alıcı saatinin GPS sistem saatine göre olan ofseti) belirlemek için en az 4 uydudan sözde mesafe ölçümleri alınır. Bu sözde mesafe ölçümleri şu denklem ile elde edilir:

$$p_i = \|s_i - u\| + ct_u \quad (4.1)$$

Burada i 1'den 4'e kadar uydu sayısını, p sözde mesafe ölçümünü, s , uydunun ECEF mesafesini, u , kullanıcının ECEF mesafesini, c ışık hızını göstermektedir. Bu denklem, aradığımız bilinmeyenler olan x_u, y_u, z_u, t_u 'ya göre şu şekilde genişletilebilir:

$$p_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2} + ct_u \quad (4.2)$$

Bu non lineer denklemi çözmek için birçok teknik geliştirilmiştir. Bunlar kapalı form (closed-form) çözümler, Kalman filtresi ve iteratif en küçük kareler yöntemleridir.

4.1. İteratif En Küçük Kareler Yöntemi (Iterative Least Squares)

Kullanıcının konumu bilinmediğinden, tahmini sözde mesafeler oluşturmak için, tahmini kullanıcı konumu ($\hat{x}, \hat{y}, \hat{z}$) kullanılır.

$$p_i = \sqrt{(x_i - \hat{x}_u)^2 + (y_i - \hat{y}_u)^2 + (z_i - \hat{z}_u)^2} + c\hat{t}_u \quad (4.3)$$

Gerçek konum ile, tahmini konum arasındaki ilişki hata ile beraber şu şekilde yazılabilir:

$$x_u = \hat{x}_u + \Delta x_u \quad (4.4)$$

Sözde mesafe denklemleri, lineerleştirilmek için birinci dereceden Taylor serisine açılarak, şu yaklaşıklık elde edilir:

$$\Delta p = H\Delta x \quad (4.5)$$

Matris gösterimi de şu şekilde olur:

$$\Delta p = \begin{bmatrix} \Delta p_1 \\ \cdot \\ \cdot \\ \Delta p_n \end{bmatrix} \quad H = \begin{bmatrix} a_{x1} & a_{y1} & a_{z1} & -1 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ a_{xn} & a_{yn} & a_{zn} & -1 \end{bmatrix} \quad \Delta x = \begin{bmatrix} \Delta x_u \\ \cdot \\ \cdot \\ \Delta ct_u \end{bmatrix} \quad (4.5a)$$

Δp_i elemanları şu şekilde tanımlanır:

$$\Delta p_i = \hat{p}_i - p_i \quad (4.6)$$

$$a_{xi} = \frac{x_i - \hat{x}_u}{\hat{r}_i}, \quad a_{yi} = \frac{y_i - \hat{y}_u}{\hat{r}_i}, \quad a_{zi} = \frac{z_i - \hat{z}_u}{\hat{r}_i} \quad (4.6a)$$

$$r_i = \sqrt{(x_i - \hat{x}_u)^2 + (y_i - \hat{y}_u)^2 + (z_i - \hat{z}_u)^2} \quad (4.6b)$$

(4.5) denkleminin çözümü:

$$\Delta x = H^{-1} \Delta p \quad (4.7)$$

Δx için bulunan değerler (4.4) denkleminde kullanılarak kullanıcı konumu elde edilir.

Burada üç olası durum göz önüne alınmalıdır. Eğer dörtten az sayıda uydunun sözde mesafesi elde ise, kullanıcı konumu belirlenemez, çünkü Δx çözülemez. Eğer tam olarak dört ayrı uydudan sözde mesafe varsa tek bir çözüm olacaktır. Fakat, 4'ten fazla uydu görünür durumda ise, ihtiyacın üzerinde bir lineer sistem elde edilir ve bu sistemle Δx i bulmak için mükemmel bir çözüm sunulamaz. Bu noktada, en küçük kareler yöntemi kullanılır.

En küçük kareler yöntemi ile şu çözüm elde edilebilir:

$$\Delta x = (H^T H)^{-1} H^T \Delta p \quad (4.8)$$

Ağırlıklandırılmış en küçük kareler yöntemi ile şu çözüm elde edilir:

$$\Delta x = (H^T C_p^{-1} H)^{-1} H^T \Delta p \quad (4.9)$$

Ağırlıklandırılmış en küçük kareler yöntemi, sözde mesafe ölçümleri farklı hata istatistiklerine sahipken veya sözde mesafe ölçümleri ilintili iken kullanılabilir. C_p matrisi, ölçüm hata kovaryans matrisidir (diyagonal terimler ölçüm hata varyansları, diğer terimler ölçüm hataları arasındaki korelasyondur). $C_p = I$ (birim matris) durumunda, ağırlıklandırılmamış durumla, bu ağırlıklandırılmış çözüm özdeştir.

Dörtten fazla uydu bulunması durumunda, Δx için tam bir çözüm yoktur. Ancak ölçüm rezidüelleri, v , ölçümlere uygulanırsa :

$$\Delta p = H\Delta x + v \quad (4.10a)$$

Veya,

$$v = \Delta p - H\Delta x \quad (4.10b)$$

Uydu saat düzeltmeleri uygulanırsa:

$$p_{corrected} = p + c\Delta t_{sv} \quad (4.11)$$

Burada,

$p_{corrected}$ = Uydu saat hatası düzeltilmiş sözde mesafe

p = Orjinal ham sözde mesafe

Δt_{sv} = Uydu saat düzeltmesi $a_{f0} + a_{f1}(t - t_0) + a_{f2}(t - t_0)^2 + \Delta t_r$

$a_{f0}, a_{f1}, a_{f2}, t_0$ = Navigasyon mesajından gelen uydu saat hatası düzeltme parametreleri

Δt_r = Görelilik düzeltmesi = $F e \sqrt{a} \sin(E_k)$

F = sabit = $-4.442807633 \times 10^{-10} \text{ sec}/(\text{meter})^{1/2}$

e = Navigasyon mesajından gelen eksantriklik

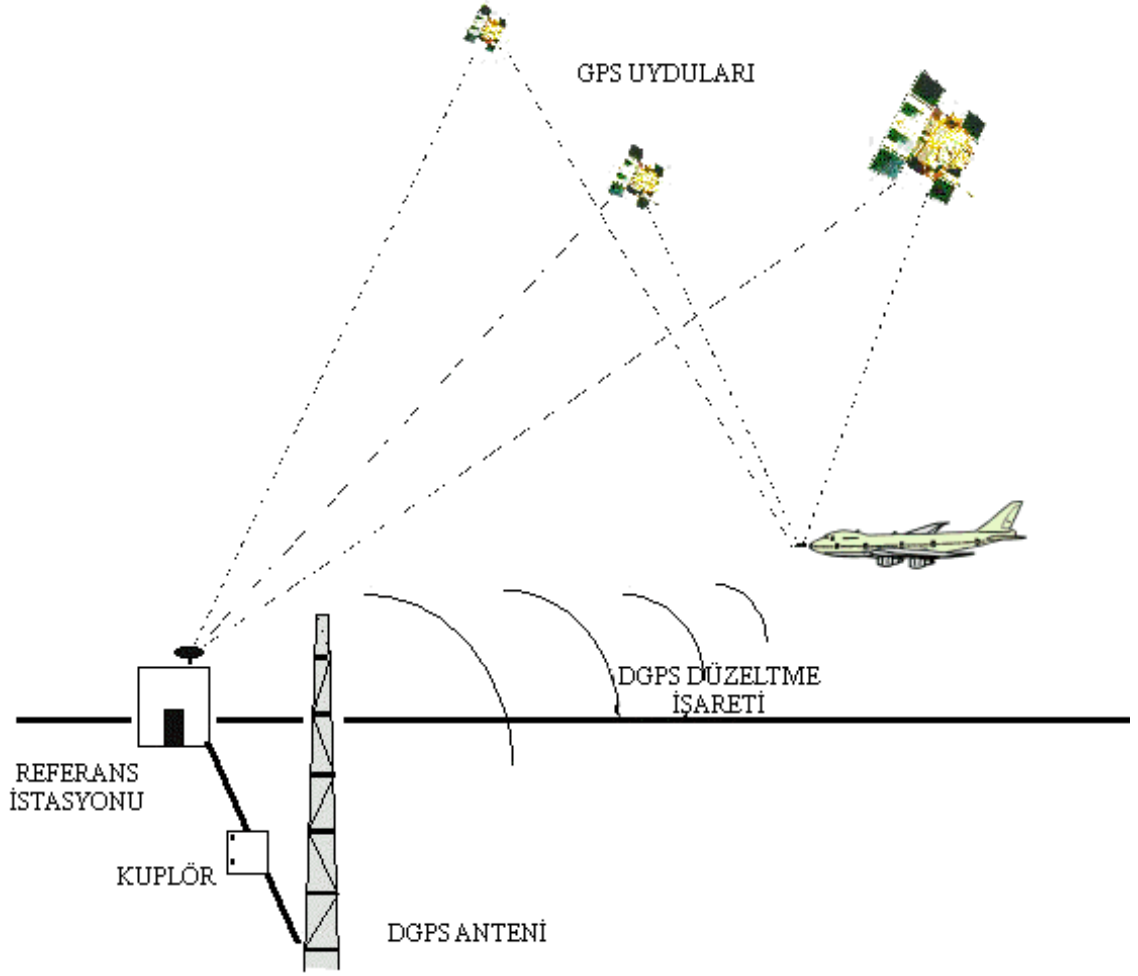
\sqrt{a} = Navigasyon mesajından gelen yarım ana eksenin karekökü

E_k = Uydu konum hesaplamasından gelen eksantrik anomalisi

İteratif en küçük kareler yöntemiyle konum belirleme tekniği hızlı ve güvenilir olmasına rağmen, bulunan sonuçlar yaklaşık 30-50 m. hata ile olmaktadır [3,5].

4.2. Diferansiyel GPS Konumlama Teknikleri

DGPS'teki temel fikir, iki yada daha çok alıcının aynı uydudan aldığı sinyal sayesinde, ortak hataları gidermektir. Bu alıcılardan biri referans alıcıdır ve konumu bilinir. Diğer alıcılar da, kullanıcılardır ve kullanıcılar referans istasyonun görüş alanı içindedirler. Referans istasyon, diğer tüm alıcılar gibi GPS sözde mesafe ölçümleri yapar, fakat referans istasyon kendi kesin konumunu bildiğinden, ölçümlerdeki sapmaları belirleyebilir. Referans istasyonun görüşündeki tüm uydular için, bu sapmalar sözde mesafe ölçümlerinden uydu-istasyon geometrik uzaklığı çıkartılarak hesaplanır. Bu sapmalar, sözde mesafe ölçüm süreci içinde oluşan (örn. İyonosfer, troposfer gecikmeleri, alıcı gürültüsü) ve alıcı saatinin GPS sistem saatine göre sapma hatalarıdır. Gerçek zamanlı uygulamalarda referans istasyon bu sapmaları, diferansiyel düzeltmeleri, kapsama alanındaki tüm kullanıcılara iletir. Kullanıcılar kendi konum belirleme doğruluklarını artırmak için bu düzeltmeleri kullanırlar. Kullanıcı ve referans istasyon ortak hatalara sahip oldukları için ve kullanıcı tarafından bu hatalar giderilebildiği için, DGPS iyileştirilmiş doğrulukta bir başarımlar sağlar. Tipik olarak konum belirleme hatası 10m.den az olmaktadır[3].



Şekil 4.1: Genel DGPS uygulaması

DGPS' te genellikle iki teknik kullanılır, kod ve taşıyıcı fazına dayalı algoritmalarıdır.

4.2.1. Koda dayalı algoritma

Referans alıcının (m) bilinen konumu (x_m, y_m, z_m) olsun ve i 'nci uydunun bildirilen yörünge bilgisi (x_i, y_i, z_i) olsun. Referans istasyondan i 'nci uyduya olan geometrik uzaklık şu şekilde bulunur:

$$R_m^i = \sqrt{(x_i - x_m)^2 + (y_i - y_m)^2 + (z_i - z_m)^2} \quad (4.12)$$

Referans alıcı, i 'nci uydunun için sözde mesafe ölçümünü şu şekilde yapabilir:

$$p_m^i = R_m^i + \varepsilon_{m,uzay} + \varepsilon_{m,kontrol} + \varepsilon_{m,kullanici} + ct_m \quad (4.13)$$

Burada $\varepsilon_{m,uzay}$, $\varepsilon_{m,kontrol}$ ve $\varepsilon_{m,kullanici}$ sırasıyla uzay, kontrol ve kullanıcı bölümlerinin etkisiyle oluşmuş sözde mesafe hataları; t_m referans alıcının saatinin GPS sistem

saatinden sapmasını göstermektedir. Bu hatalar Tablo 4.1'de özetlenerek gösterilmektedir. Referans alıcı diferansiyel düzeltmeyi oluşturmak için, i 'nci uydunun sözde mesafe ölçümünden p_m^i ve R_m^i geometrik uzaklığı ayırır;

$$\Delta p_m^i = p_m^i - R_m^i = \varepsilon_{m,uzay} + \varepsilon_{m,kontrol} + \varepsilon_{m,kullanici} + ct_m \quad (4.14)$$

Kullanıcı bu düzeltme teriminden yararlanarak kendi sözde mesafe ölçümünü oluşturur:

$$p_u^i - \Delta p_m^i = R_u^i + \varepsilon_{m,uzay} + \varepsilon_{m,kontrol} + \varepsilon_{m,kullanici} + \varepsilon_{m,uzay} + ct_m - (\varepsilon_{m,uzay} + \varepsilon_{m,kontrol} + \varepsilon_{m,kullanici} + ct_m) \quad (4.15)$$

Eğer kullanıcının alıcısı referans alıcıya yakın konumdaysa, kullanıcının alıcısındaki sözde mesafe hata bileşenleri, referans alıcınıninkilerle neredeyse aynı olacaktır. Multipath, kullanıcı gürültüsü gibi etkenler, bu durum için istisnadır ve her iki alıcı için de ortak hata değillerdir. Bu yüzden kullanıcıdaki düzeltilmiş sözde mesafe şu şekilde elde edilir:

$$p_{u,düzeltilmiş} = R_u^i = \varepsilon_u' + ct_{toplam} \quad (4.16)$$

Burada ε_u' rezidüel kullanıcı bölümü hatası (multipath, vs..) ve t_{toplam} birleşmiş saat sapmalarıdır ($t_u - t_m$).

Tablo 4.1 : DGPS düzeltmesinden önce ve sonra tipik uydu hataları

Kaynak Bölüm	Hata Kaynağı	GPS 1 σ Hata (m)	DGPS Düzeltmesi Sonrası GPS 1 σ Hata (m)
Uzay	Uydu saat stabilitesi	3.0	0
	Uydu düzensizliği	1.0	0
	Uydu erişilebilirliği	32.3	0
	Diğer (termal, radyasyon, vb.)	0.5	0
Kontrol	Yörünge tahmin hatası	4.2	0
	Diğer (düzenleyici roket permormansı, vb..)	0.9	0

Tablo 4.1(devam) : DGPS düzeltmesinden önce ve sonra tipik uydu hataları

Kullanıcı	Iyonosfer gecikmesi	5.0	0
	Troposfer gecikmesi	1.5	0
	Alıcı gürültüsü ve çözünürlüğü	1.5	2.1
	Çok yol etkisi	2.5	2.5
	Diğer	0.5	0.5
Sistem UERE	Toplam (rms)	33.3	3.3

4.2.2. Taşıyıcı fazına bağlı algoritma

Alınan sinyalin frekansındaki kaymanın (Doppler kayması) büyüklüğünü ölçerek daha kesin konum bilgisi elde edilebilir. Tipik olarak taşıyıcı frekansındaki kayma GPS uydusunun

kullanıcıya göre bağlı hareketinden kaynaklanır, L1 ve L2 taşıyıcılarında $\Delta f = \pm 5000 \text{ Hz}$ Doppler frekans kayması oluşur [3].

$$\Delta f = f_R - f_T \quad (4.17)$$

Burada f_R , kullanıcıda alınan işaretin frekansı (Hz) ; f_T ise bilinen iletim frekansdır (Hz). Taşıyıcı faz ölçümleri, $\Phi(t)$, Doppler ölçümlerinin $\Delta f_{ölçülen}$ zaman üzerinde integre edilmesiyle elde edilebilir:

$$\Phi(t) = \int_{t_0}^t \Delta f_{ölçülen}(t) dt + \phi(t_0) \quad (4.18)$$

Başlangıç taşıyıcı fazının tamsayı kısmına “taşıyıcı fazı tamsayı belirsizliği” denir. Kullanıcı uydu takip ettiğinde, alıcı sadece taşıyıcı çevrimlerini saydığından bu tamsayı vardır. En kesin uzaklık ölçümünü yapabilmek için, bu tamsayı belirsizliğinin çözümü çok önemlidir ve çözümü için birçok teknikten yararlanılır. En popüler olanları en küçük kareler iterasyonu ve LAMDA metodlarıdır [5].

4.3. Genel Kinematik GPS Teknikleri

Bazı basit alıcılarda kullanıcı hızı, tahmin edilen konumun zamana göre türevi alınarak hesaplanır:

$$\dot{u} = \frac{du}{dt} \approx \frac{u(t_2) - u(t_1)}{t_2 - t_1} \quad (4.19)$$

Genellikle bu yaklaşım zayıf sonuçlar doğurur; ancak kullanıcı hızı belirli zaman aralığında sabit ise, kabul edilebilir.

Birçok alıcı Doppler ölçümlerini işleyerek verimli bir şekilde uydunun Doppler frekansını tahmin edebilir. Uydu hız vektörü, yörünge verisi ve alıcıdaki yörünge modeli ile hesaplanır [3,5]. Alıcı antende alınan frekans, klasik Doppler denklemiyle şöyle bulunur: (görelilik etkilerini ihmal ederek)

$$f_R = f_T \left(1 - \frac{v_r \cdot a}{c}\right) \quad (4.20)$$

f_T = iletilen uydu sinyali frekansı

v_r = uydu – kullanıcı bağıl hız vektörü

a = kullanıcıdan uyduya doğrultuyu gösteren birim vektör

c = ışık hızı

Nokta çarpım, uydu vektörü a_r 'ye anlık açık görüş boyunca olan uydu-kullanıcı bağıl hız vektörünün radyal parçasını temsil etmektedir. v_r 'nin miktarı hız farkı olarak veriliyor:

$$v_r = v - u \quad (4.21)$$

v , uydunun hızı ve u , kullanıcının hızıdır (her ikisinde bilinen ECEF frame'ine referans eder). Bundan dolayı, bağıl hareketten kaynaklanan Doppler kayması şu bağıntıyı getirir:

$$\Delta f = f_R - f_T = -f_T \frac{(v - u) \cdot a}{c} \quad (4.23)$$

Ölçülen Doppler frekansından(Δf), kullanıcının hızını(u) hesaplamının farklı teknikleri vardır[3].

j 'nci uydü için, (4.23) denklemin şü hale gelir:

$$f_{R_j} = f_{T_j} \left\{ 1 - \frac{1}{c} [(v_j - u) \cdot a_j] \right\} \quad (4.24)$$

Düzeltilmiş uydü frekansını,

$$f_{T_j} = f_0 + \Delta f_{T_j} \quad (4.25)$$

olarak verilir. Burada f_0 , nominal iletilen frekans ve Δf_{T_j} , navigasyon mesaj güncellemesinden belirlenen düzeltmedir.

Frekansın ölçülen değeri, frekans kaymasından dolayı, hatalıdır. Bu kayma, kullanıcı zamanındaki sapma ve GPS zamanı ile ilgilidir.

$$f_{R_j} = f_j (1 + \dot{\delta}_u) \quad (4.26)$$

Denkleminde, kullanıcı saati hızlı ise $\dot{\delta}_u$ pozitif alınır. Cebirsel işlemler ile,

$$\frac{c(f_j - f_{T_j})}{f_{T_j}} + v_{xj} a_{xj} + v_{yj} a_{yj} + v_{zj} a_{zj} = \dot{x}_u a_{xj} + \dot{y}_u a_{yj} + \dot{z}_u a_{zj} - \frac{cf_j \dot{\delta}_u}{f_{T_j}} \quad (4.27)$$

olarak yazılır. Burada v_j, a_j j'nci uydunun hızı ve ivmesidir. $v_u = (\dot{x}_u, \dot{y}_u, \dot{z}_u)$ 'da kullanıcı hızıdır. Denklemi daha da sadeleştirirsek, $f_j / f_{T_j} = 1$ olduğundan

$$d_j = v_{xj} a_{xj} + v_{yj} a_{yj} + v_{zj} a_{zj} \quad (4.28)$$

Dört tane bilinmeyen, dört uydudan alınan ölçümlerle ve lineer denklem kümesiyle çözülebilir.

$$d = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} \quad H = \begin{bmatrix} a_{x1} & a_{y1} & a_{z1} & 1 \\ a_{x2} & a_{y2} & a_{z2} & 1 \\ a_{x3} & a_{y3} & a_{z3} & 1 \\ a_{x4} & a_{y4} & a_{z4} & 1 \end{bmatrix} \quad g = \begin{bmatrix} \cdot \\ x_u \\ \cdot \\ y_u \\ \cdot \\ z_u \\ \cdot \\ -c \dot{\delta}_u \end{bmatrix} \quad (4.29)$$

$$d = Hg \quad (4.29a)$$

genel formuyla yazılan bu bağıntı;

$$g = H^{-1}d \quad (4.29b)$$

ile çözülebilir. Belirtilen çözüm, kullanıcı hızını belirlemek için, ölçüm gürültüsü veya multipath hataları ile bozulmuş ölçümleri kullanmaktadır. Kalman filtresi metodu burada kullanılırsa daha yumuşatılmış bir navigasyon çözümü elde edilebilir.

Kalman filtresi tekniği, gürültü istatistiklerine ve mevcut ölçümlere bağlı olarak kullanıcı pozisyonu, hızı ve saat kaymasının optimum tahminlerini yapan özyinelemeli bir algoritmadır[3]. Kalman filtresi 4 uydudan az sayıda uydu kullanımına izin verir ve ölçüm gürültüsünün etkilerini ağırlıklandırmak için durum tahminlerini ayarlar.

4.4. Kapalı Form GPS Konumlama Algoritması

GPS kullanıcısının konumunu bulmak için şu anda kullanılan metod, sözde mesafe denklemlerini lineerleştirmek ve kullanıcı tarafından sağlanan başlangıç noktası tahminini kullanarak, iteratif yolla hesaplamaktır. Dünyaya yakın konumları belirlemede, dünyanın merkezi, iteratif algoritmaların başlangıç noktası olarak seçilmesi, iyi bir seçimdir, çözüm gerçek konuma yakınsar [5-7].

Son yıllarda, bu konuda yapılan çalışmalar, nonlinear GPS sözde mesafe ölçümlerini çözmek için, iteratif olmayan kapalı form çözümler önermektedir [5, 8-12].

Bu çalışmada incelenen algoritma, [5] nolu kaynakta önerilen algoritmadır. Bu algoritma, sözde mesafe ölçümlerinin kapalı form çözümünü içeren ve sözde mesafe ölçüm gürültüsü altında, konum tahminlerinin istatistiklerine odaklanan bir algoritmadır.

İncelenen kapalı form çözüm, ($n \geq 5$) uydunun varlığını gerektirmektedir ve GPS probleminde 4 bilinmeyen çözülmesi gerektiği gerçeği göz önünde bulundurulduğunda, artık belirtilmiş (over determined) bir sistemle çalışmaktadır. Sözde mesafe ölçümleri, gürültülü kabul edilmektedir. Ölçümlerin stokastik tabiatı, GPS sözde mesafe denklemlerine yansıtılmıştır ve stokastik olarak modellenen ölçümlerle konum bulma, stokastik kestirim problemine dönüşmüştür. Doğru stokastik kestirim modeli ve kestirimi kullanıldığında kullanıcı konumu yanında ölçüm gürültü gücü kestirimi de sunan bir GPS çözümü elde edilmiştir.

4.4.1. Kapalı form çözüm

Rezidüel hatalar, tek bir rastgele değişken V altında toplanırsa,

$$R_i = \sqrt{(u_x - x_i)^2 + (u_y - y_i)^2 + (u_z - z_i)^2} + b + v_i \quad (4.30)$$

yazılabilir. Bu denklem şu şekilde tekrar yazılabilir:

$$(u_x - x_i)^2 + (u_y - y_i)^2 + (u_z - z_i)^2 = (R_i - b - v_i)^2 \quad (4.31)$$

(4.31) genişletildiğinde

$$\begin{aligned} u_x^2 + u_y^2 + u_z^2 - b^2 - 2x_i u_x - 2y_i u_y - 2z_i u_z + 2R_i b \\ = R_i^2 - x_i^2 - y_i^2 - z_i^2 - 2R_i v_i + 2b v_i + v_i^2 \end{aligned} \quad (4.32)$$

elde edilir.

(4.32) nin ilk dört terimi, n tane denklemde de ortak olan, bilinmeyenlerdir. Nonlineer terimleri fark alarak elemek mümkündür. B'inci denklemi kalan (n-1) denklemden çıkarırsak, (n-1) denklemi, bilinmeyen değişkenleri lineer şekilde yazabiliriz.

$$\begin{aligned} (x_n - x_i)u_x + (y_n - y_i)u_y + (z_n - z_i)u_z + (R_i - R_n)b \\ = \frac{1}{2}(R_i^2 - R_n^2 + x_n^2 - x_i^2 + y_n^2 - y_i^2 + z_n^2 - z_i^2) \\ + R_n v_n - R_i v_i - b v_n + b v_i + \frac{1}{2}(v_i^2 - v_n^2) \end{aligned} \quad (4.33)$$

(4.33) denklemini matris formunda yazdığımızda,

$$\vec{Z} = H\vec{u} + \vec{V} \quad (4.34)$$

burada \vec{Z} ölçüm vektörüdür.

$$\vec{Z} = \begin{bmatrix} Z_1 \\ Z_2 \\ \bullet \\ \bullet \\ \bullet \\ Z_{n-1} \end{bmatrix} \quad (4.35)$$

\vec{Z} 'nin elemanları şu şekilde ifade edilir:

$$Z_i = \frac{1}{2}(R_i^2 - R_n^2 + x_n^2 - x_i^2 + y_n^2 - y_i^2 + z_n^2 - z_i^2) \quad (4.36)$$

H, (n-1) x 4 lük regresör matristir.

$$H = \begin{bmatrix} (x_n - x_1) & (y_n - y_1) & (z_n - z_1) & (R_1 - R_n) \\ (x_n - x_2) & (y_n - y_2) & (z_n - z_2) & (R_2 - R_n) \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ (x_n - x_{n-1}) & (y_n - y_{n-1}) & (z_n - z_{n-1}) & (R_{n-1} - R_n) \end{bmatrix} \quad (4.37)$$

\vec{u} , bilinmeyenlerin vektörüdür $\vec{u} = [u_x, u_y, u_z, b]^T$

Son olarak, \vec{V} de (n-1) hata vektörüdür. $V = [V_1, V_2, \dots, V_{n-1}]^T$

$$V_i = R_n v_n - R_i v_i - b(v_n - v_i) + \frac{1}{2}(v_i^2 - v_n^2), i = 1, 2, \dots, n-1 \quad (4.38)$$

\vec{u} nun kestirimini yapabilmek için, denklem hata vektörü \vec{V} nin istatistiğini, sözde mesafe ölçüm gürültüsünün bilinen istatistiklerinden çıkarılmalıdır.

$$v_i \in N(0, \sigma^2), i = 1, 2, \dots, n \quad (4.39)$$

olduđuna göre,

$$\begin{aligned} E\{v_i\} &= 0 \\ E\{v_i^2\} &= \sigma^2 \quad i = 1, 2, \dots, n-1 \\ E\{v_i v_j\} &= 0 (i \neq j) \end{aligned} \quad (4.40)$$

yazılabilir. Buradan, hata vektörü \vec{V} nin istatistiđi hesaplanabilir:

$$\begin{aligned} E\{V_i\} &= 0 \\ E\{V_i V_j\} &= \begin{cases} \sigma^4 + \sigma^2 [(R_n - b)^2 + (R_i - b)^2] & i = j \text{ için} \\ \frac{\sigma^4}{2} + \sigma^2 (R_n - b)^2 & i \neq j \text{ için} \end{cases} \end{aligned} \quad (4.41)$$

Kovaryans matris $P_v = E(\vec{V} \vec{V}^T)$ tri diyagonal formda ifade edilebilir:

$$P_v = \sigma^2 \begin{bmatrix} d_1 & c & \bullet & \bullet & c \\ c & d_2 & c & \bullet & \bullet \\ \bullet & \bullet & d_3 & \bullet & \bullet \\ \bullet & c & c & \bullet & c \\ c & \bullet & \bullet & \bullet & d_{n-1} \end{bmatrix}_{(n-1) \times (n-1)} \quad (4.42)$$

Burada c,

$$c = \frac{\sigma^2}{2} + (R_n - b)^2 \quad (4.42a)$$

bađıntısıyla verilmektedir. (4.42) matrisindeki diyagonal elemanlar :

$$d_i = \sigma^2 + (R_n - b)^2 + (R_i - b)^2 \quad i = 1, 2, \dots, n-1 \text{ için} \quad (4.42b)$$

(4.34) deki lineer regresyon, \vec{u} vektörünün bilinmeyen parametreleri kestirimini yapmak için kullanılır. Amaç, kestirim hatasını minimize eden $\hat{\vec{u}}$ yu elde etmektir. En küçük varyans parametresi kestirimi:

$$\hat{\vec{u}} = (H^T P_v^{-1} H)^{-1} H^T P_v^{-1} \vec{Z} \quad (4.43)$$

(4.43) de bulunan kestirimin bir özelliği de σ , sözde mesafe ölçüm hata varyansına bağlı olmamasıdır. (4.45) denklemini hata kovaryansı $P_{\hat{u}}$ 'yi \tilde{R} 'nin skaler bir değerle çarpımı olarak ifade eder. (4.43) denkleminde $P_{\hat{u}}$ 'nin skaler çarpımı gidecektir ve minimum varyans parametre kestirimi şu şekilde yazılabilir:

$$\hat{u} = (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \tilde{Z} \quad (4.44)$$

(4.44) 'den kestirim hatasının kovaryansı

$$\begin{aligned} P_{\hat{u}} &= E \left\{ (\hat{u} - \bar{u}) (\hat{u} - \bar{u})^T \right\} \\ &= \sigma^2 c (H^T \tilde{R}^{-1} H)^{-1} \end{aligned} \quad (4.45)$$

Kestirim çözümünün tersine kovaryans $P_{\hat{u}}, \sigma$ 'ya bağlıdır. Bundan dolayı, hata kovaryansını kestirebilmek için, σ ya bilinmeli, ya da kestirilebilmelidir.

(4.34) yi (4.44) nin içinde yazdığımızda

$$\begin{aligned} \hat{u} &= (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} (H\bar{u} + \tilde{V}) \\ &= \bar{u} + (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \tilde{V} \end{aligned} \quad (4.46)$$

(4.46) 'yi sonuç fark denkleminde yazdığımızda şu bağıntı ortaya çıkar:

$$\begin{aligned} \tilde{Z} &= \tilde{Z} - H\hat{u} = H\bar{u} + \tilde{V} - H(\bar{u} + (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \tilde{V}) \\ &= (I_{n-1} - H(H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1}) \tilde{V} \end{aligned} \quad (4.47)$$

$M \equiv I_{n-1} - H(H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1}$ şeklinde bir matris tanımlarsak, fark denklemini şu şekilde ifade edilebilir.

$$\tilde{Z} = M\tilde{V} \quad (4.48)$$

Ağırlıklandırılmış sonuç fark denklemini,

$$\tilde{Z} \equiv R_e^{-1} \tilde{Z} \quad (4.49)$$

Burada R_e^{-1} Cholesky ayrışmasından $\tilde{R} : \tilde{R} = R_e^T \tilde{R}^{-1}$ elde edilir. R_e (n-1) x (n-1) boyutlarında olduğundan, büyük bir matris tersini alma işleminden kaçınmak için, R_e^{-1} , \tilde{R}^{-1} 'in fonksiyonu olarak yazılmıştır.

$$R_e^{-1} = R_e^T \tilde{R}^{-1} \quad (4.50)$$

Skaler değeri hesaplamak için şu bağıntıyı kullanalım

$$\tilde{\mathbf{Z}}^T \tilde{\mathbf{Z}} = \tilde{\mathbf{Z}}^T (R_e^{-1})^T R_e^{-1} \tilde{\mathbf{Z}} = \tilde{\mathbf{Z}}^T \tilde{R}^{-1} \tilde{\mathbf{Z}} = \tilde{\mathbf{V}}^T M^T \tilde{R}^{-1} M \tilde{\mathbf{V}}$$

Gösterilebilir ki,

$$E\{\tilde{\mathbf{Z}}^T \tilde{\mathbf{Z}}\} = c \sigma^2 \text{Tr}(M) \quad (4.51)$$

ve

$$\text{Tr}(M) = n - 5 \quad (4.52)$$

Buradan,

$$E\{\tilde{\mathbf{Z}}^T \tilde{\mathbf{Z}}\} = c \sigma^2 (n - 5) = \left(\frac{\sigma^4}{2} + \sigma^2 (R_n - b) \right) (n - 5) \quad (4.53)$$

(4.53) denklemi düzenlendiğinde σ^2 cinsinden ikinci derece bir denklem çıkar. Bu denklemi çözmek için şu bağıntı kullanılırsa,

$$\tilde{\mathbf{Z}}^T \tilde{\mathbf{Z}} = \tilde{\mathbf{Z}}^T \tilde{R}^{-1} \tilde{\mathbf{Z}} \quad (4.54)$$

Ölçüm gürültü gücünün, σ^2 , veri yönelimli kestirimi çıkar:

$$\hat{\sigma}^2 \approx -(R_n - \hat{b}) + \sqrt{(R_n + \hat{b})^2 + \frac{2(\tilde{\mathbf{Z}}^T \tilde{R}^{-1} \tilde{\mathbf{Z}})}{n - 5}} \quad (4.55)$$

Türetilmiş lineer regresyon bağıntısı (4.34) 'de (n-1) denklemden oluşmaktadır ve \bar{u} 'daki 4 bilinmeyen kestirimini yapılabilmesi için (n-1) = 4 'ün sağlanmasını gerektirmektedir. Buradan hareketle, (4.44) deki çözümü oluşturmak için, en azından

5 uydunun görünür olması gerekmektedir. Bir fazladan uydu bilgisi de, hata kovaryansı kestirimi $P_{\bar{u}}$ 'nin, veri yönelimli tahminini yapabilmek için gereklidir. En azından 6 uydunun görünür olması, \bar{u} 'daki 4 parametrenin ve kestirim hata kovaryansının başlangıç tahminini yapılabilmesi için gereklidir. Eğer sözde mesafe ölçüm gürültü şiddeti biliniyorsa, 5 uydunun görünür olması yeterli olacaktır.

\bar{u} parametrelerinin kestirimi, başlangıçta n ölçüm olmasına rağmen, sadece n-1 denkleme dayandırılmıştır. Parametre kestirimini elde etmek için n bağıntının da kullanılması gereklidir. Algoritmanın ikinci adımında, n'inci sözde mesafe bağıntısını kullanarak, Kalman filtresi benzeri bir yaklaşımla ölçüm güncellemesi yapmaktadır.

4.4.2. Kalman güncelleştirme algoritması

Kalman güncelleştirme algoritması klasik Kalman filtresine benzerlik gösterir; fakat önceki tahmin değerlerini güncelleştirmek için kullanılan yeni ölçüm değerleri, önceki ölçüm değerleriyle ilişkili olduğundan klasik Kalman filtresinden farklıdır. Klasik Kalman filtresi önceki tahmin ve yeni ölçümün arasında korelasyon olmasını kabul etmez. Kalman benzeri güncelleştirme denklemini oluşturmak için yeni ölçümdeki gürültü (v_n) ve önceki tahmindeki gürültü arasındaki korelasyonu bilmek gereklidir.

Z_n , skaler ölçümü gösterir,

$$Z_n = R_n + \frac{(u_{x0} - x_n)x_n + (u_{y0} - y_n)y_n + (u_{z0} - z_n)z_n}{\sqrt{(u_{x0} - x_n)^2 + (u_{y0} - y_n)^2 + (u_{z0} - z_n)^2}} \quad (4.56)$$

h , (4 x 1) regresör vektördür,

$$h \equiv \begin{bmatrix} \frac{(u_{x0} - x_n)}{\sqrt{(u_{x0} - x_n)^2 + (u_{y0} - y_n)^2 + (u_{z0} - z_n)^2}} \\ \frac{(u_{y0} - y_n)}{\sqrt{(u_{x0} - x_n)^2 + (u_{y0} - y_n)^2 + (u_{z0} - z_n)^2}} \\ \frac{(u_{z0} - z_n)}{\sqrt{(u_{x0} - x_n)^2 + (u_{y0} - y_n)^2 + (u_{z0} - z_n)^2}} \\ 1 \end{bmatrix} \quad (4.57)$$

Burada kullanılan birincil kullanıcı konumu tahmini (u_{x0}, u_{y0}, u_{z0}) 4.1.1 de anlatılan kapalı form algoritmasından gelmektedir.

$$u_{x0} = \hat{u}_x, u_{y0} = \hat{u}_y, u_{z0} = \hat{u}_z$$

Tahmini GPS çözümünden, gerçek GPS parametre vektörü bir rastlantı değişkeni olarak tanımlanabilir.

$$\vec{u} = \hat{\vec{u}} + \vec{W} \quad (4.58)$$

v_n ve \vec{W} arasındaki korelasyon şu şekilde tanımlanabilir

$$p \equiv E\{\vec{W}v_n\} = E\{v_n\vec{W}\}$$

v_n ve \vec{W} arasındaki ilişkiyi belirlemek için konum vektörü şu şekilde yazılabilir

$$\vec{u} = (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \vec{Z} - (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \vec{V} \quad (4.59)$$

Buradan \vec{W} için \vec{V} türünden şu bağıntı yazılabilir:

$$\vec{W} = (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \vec{V} \quad (4.60)$$

\vec{V} ve v_n elemanları arasındaki varyans bağıntısı şu şekilde yazılabilir;

$$E\{\vec{V}v_n\} = \sigma^2 (R_n - b) \begin{bmatrix} 1 \\ 1 \\ \bullet \\ \bullet \\ \bullet \\ 1 \end{bmatrix}_{(n-1) \times 1} \quad (4.61)$$

(4.60) ve (4.61) bağıntıları kullanılarak v_n ve \vec{W} arasındaki kovaryans elde edilir:

$$p = \sigma^2 (R_n - b) (H^T \tilde{R}^{-1} H)^{-1} H^T \tilde{R}^{-1} \begin{bmatrix} 1 \\ 1 \\ \bullet \\ \bullet \\ \bullet \\ 1 \end{bmatrix}_{(n-1) \times 1}$$

Önceki \hat{u} ve $P_{\hat{u}}$ hesaplarını ve bu değerlerin Kalman ile güncellenmiş hallerini ayırmak için aşağıdaki notasyon kullanılacaktır:

- \hat{u}^- ve $P_{\hat{u}}^-$ güncelleştirmeden önceki konum vektörü ve hata vektörü tahminleridir.
- \hat{u}^+ ve $P_{\hat{u}}^+$ güncelleştirmeden sonraki konum vektörü ve hata vektörüdür.

Kalman benzeri güncelleştirme için kullanılan bağıntılar şöyledir:

$$\hat{u}^+ = \hat{u}^- + K (Z_n - h^T \hat{u}^-) \quad (4.62)$$

$$P_{\hat{u}}^+ = \{I - [(1 - p^T h)K + p]h^T\} Y \quad (4.63)$$

Bu bağıntılarda geçen Y ara terimi, güncelleştirme öncesi kovaryans matrisidir:

$$Y = P_{\hat{u}}^- + \frac{h^T P_{\hat{u}}^- h - 1}{(1 - p^T h)^2} p p^T + \frac{1}{1 - p^T h} (P_{\hat{u}}^- h p^T + p h^T P_{\hat{u}}^-) \quad (4.64)$$

K ise, Kalman filtre kazancı olarak şöyledir:

$$K = \frac{1}{1 - p^T h} \left(\frac{1}{1 + h^T Y h} Y h - p \right) \quad (4.65)$$

(4.62) (4.65) arası bağıntılar algoritmanın MATLAB implementasyonunda kullanılmıştır. Kalman güncelleştirme algoritması kapalı form GPS çözüm tahminini direkt ve nonrekürsif bir şekilde arındırmak üzere kurgulanmıştır. Ancak (4.34) deki ölçüm denklemi lineer olmadığından, Kalman güncelleştirmesi ile yapılacak bu işlem sürecinin iteratif bir şekilde sonucun önceden belirlenmiş hata payı içerisinde kalana kadar devam etmesi uygun olur.

5. VERİ TOPLAMA VE TEST DÜZENİ

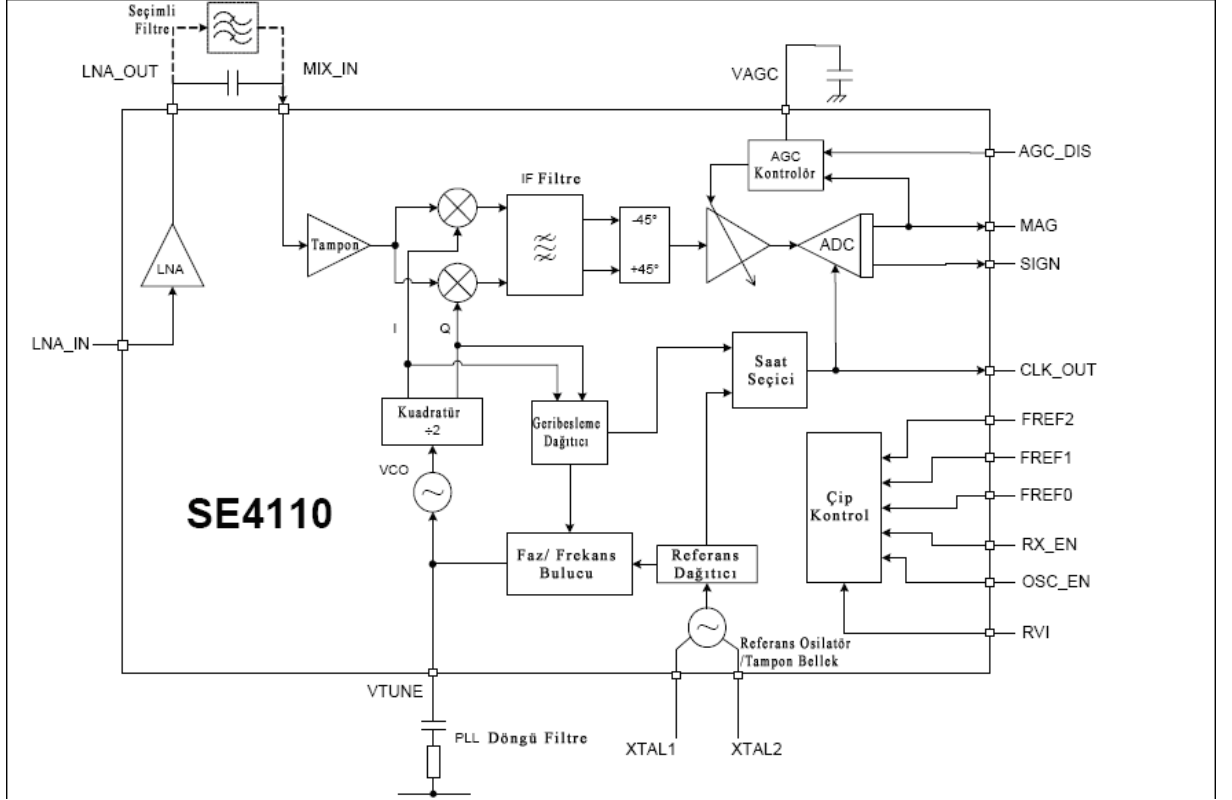
GPS Konum Belirleme algoritmalarını test etmek için University of Colorado Aerospace Department ve SiGe Semiconductor işbirliğinde üretilen GPS alıcısı, SE4110L kullanılmıştır. Bu alıcı L1 bandındaki ham GPS veri akımını toplar, analog dijital çevrimi yaparak, ikili formatta bilgisayara aktarır. Cihaz ile birlikte windows işletim sistemi için hazırlanmış, bilgisayarın cihazla konuşmasını sağlayan kütüphaneler gelmektedir. Cihazla birlikte ayrıca maksimum 38.4 saniyelik GPS verisi toplayabilen bir konsol uygulaması gelmektedir. Danish GPS Center’da geliştirilmiş başka bir uygulamayla daha büyük veriler toplanabilmektedir.



Şekil 5.1: Veri toplama amacıyla kullanılan GPS alıcısı

Bu uygulamalarla GPS verisi toplamanın süre kısıtı, sabit diskin tampon belleğinin sınırlı olmasından kaynaklanmaktadır. Bazı durumlarda, bu kısıt veri toplama işinin yarıda kesilmesine yol açmaktadır. Veri toplama uygulamasında sağlanan bir seçenek ile veri doğrudan hafızada tamponlanıp, sonradan sabit diske yazıldığında bu sorun aşılmaktadır.

İkili GPS verilerini işlemek için bir çatı olarak MATLAB kaynak kodu kullanılmaktadır[1]. GPS sinyalinin işlendiği tüm evreler ayrı ayrı ele alınmaktadır ve bu alanlarda yapılacak geliştirmeler için uygundur.



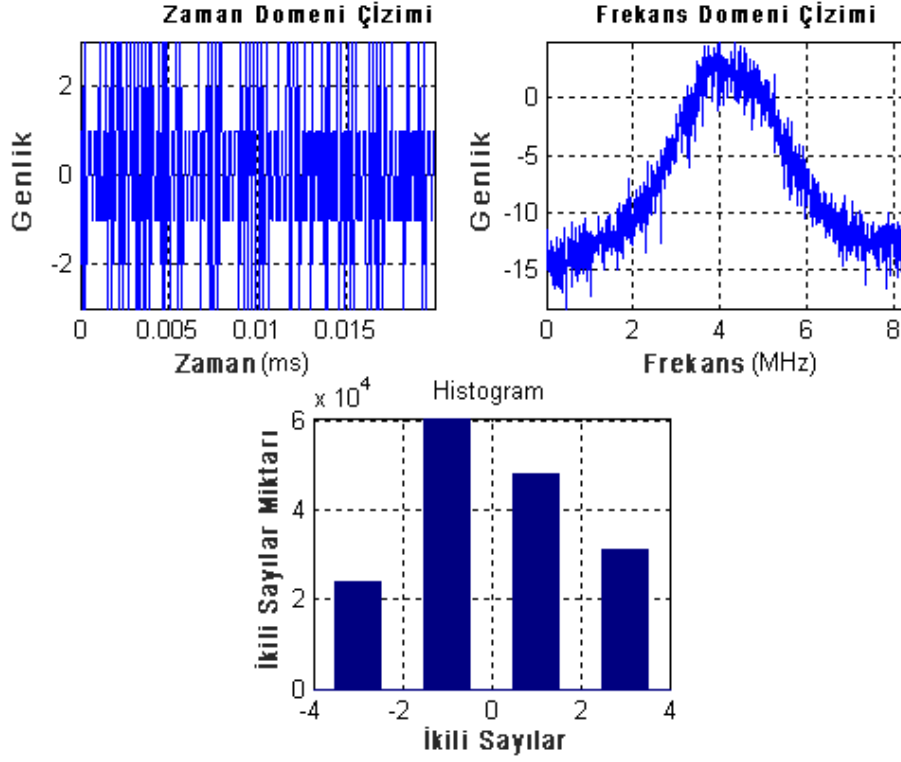
Şekil 5.2: SiGe 4110L fonksiyonel blok diyagramı

5.1 Toplanan Verinin İncelenmesi

GPS alıcısıyla İstanbul – Akatlar bölgesinde veri toplanmıştır. Veri bulutsuz havada, açık görüş sahasında toplanmış olup toplam 6 GPS uydusundan (PRN: 2 – 4 - 9 – 12 – 15 - 17) yeterli sinyal seviyesi alınmış ve bu GPS uydularından toplanan veriler incelenmiştir. Veriler GPS alıcısından toplanan ham verinin incelenmesi için özel olarak MATLAB ortamında geliştirilmiş program yardımıyla incelenmiştir[1].

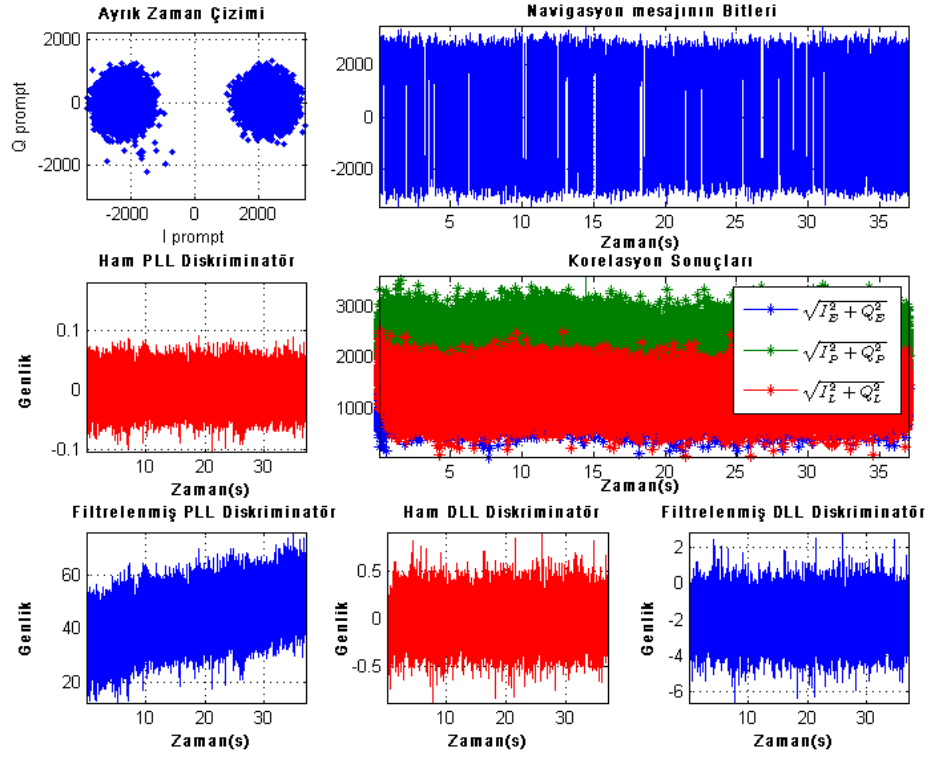
5.1.1 GPS ham verisi ve kanalların izlenme sonuçları

Şekil 5.1’de toplanan GPS Ham verisi ile ilgili bilgiler yer almaktadır. GPS işaretinin zaman domeninde, frekans domeninde ve analog dijital dönüşümü sonucu histogram olarak gösterilmiştir.

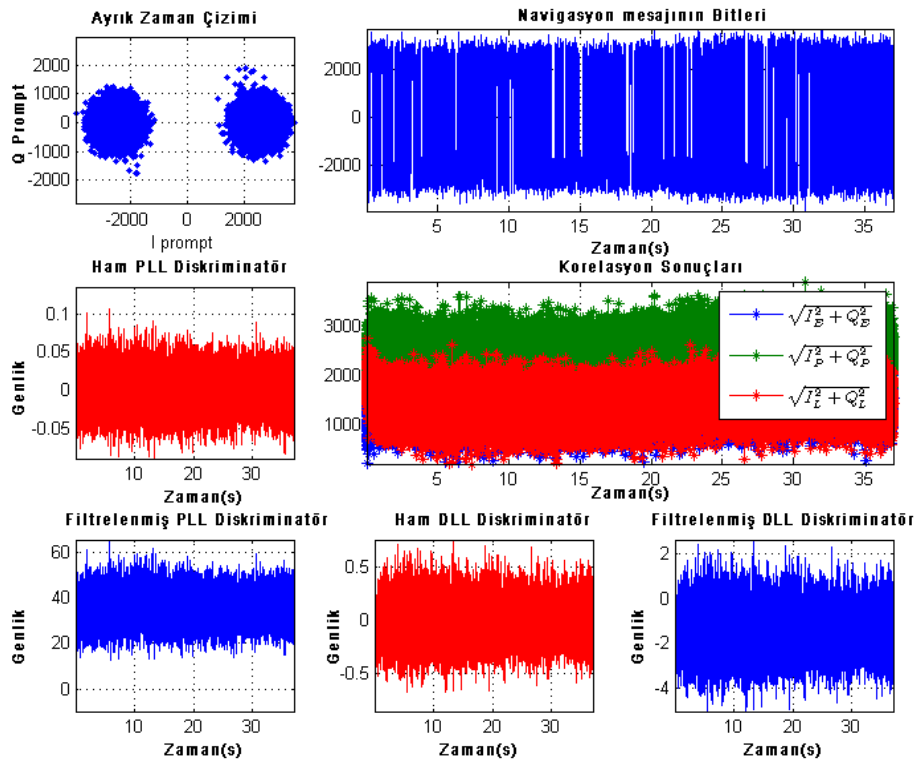


Şekil 5.3 : Ham GPS veri bilgisi

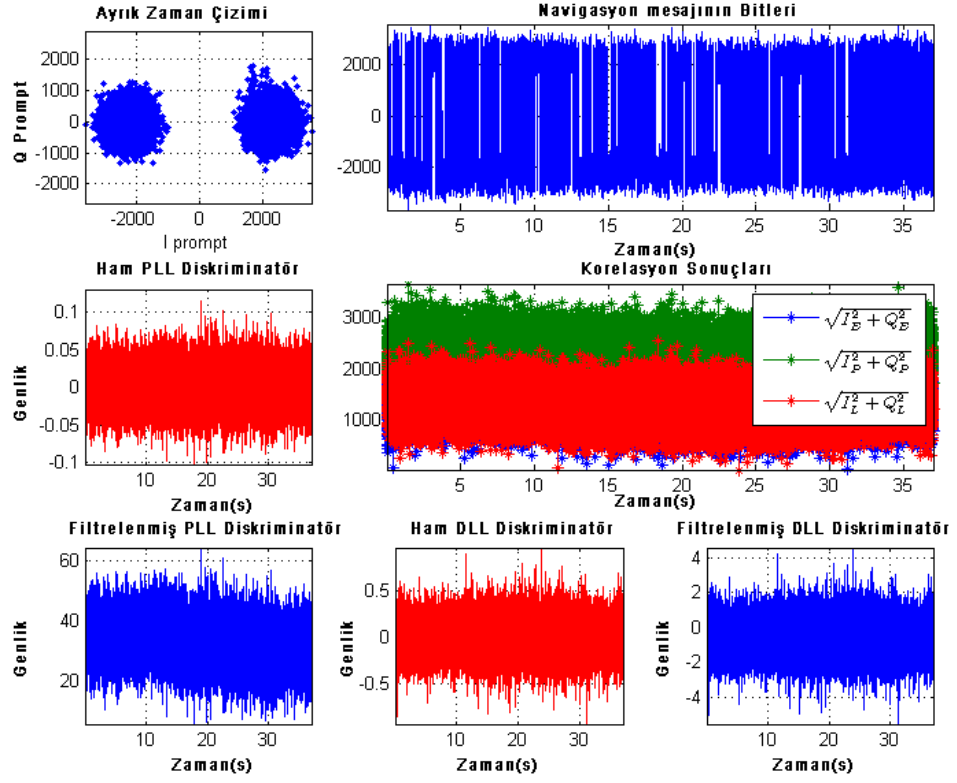
Şekil 5.4' den Şekil 5.7' e kadar olan şekiller saptanan GPS uyduların (kanallar) ile ilgili verilerdir. İzleme (tracking) adımı sonucu ortaya çıkar. İzleme adımında bir önceki (acquisition) adımda elde edilen frekans ve kod fazı parametrelerinin düzeltilmesi ve her uydudan navigasyon verisinin demodüle edilmesi işlemleri yapılır. Şekillerde kanalların ayrık zamanda saçılımı, Ham ve filtrelenmiş DLL-PLL diskriminatörleri, navigasyon mesajı bitleri grafikleri ve korelatörlerin korelasyon sonuçları yer almaktadır [1].



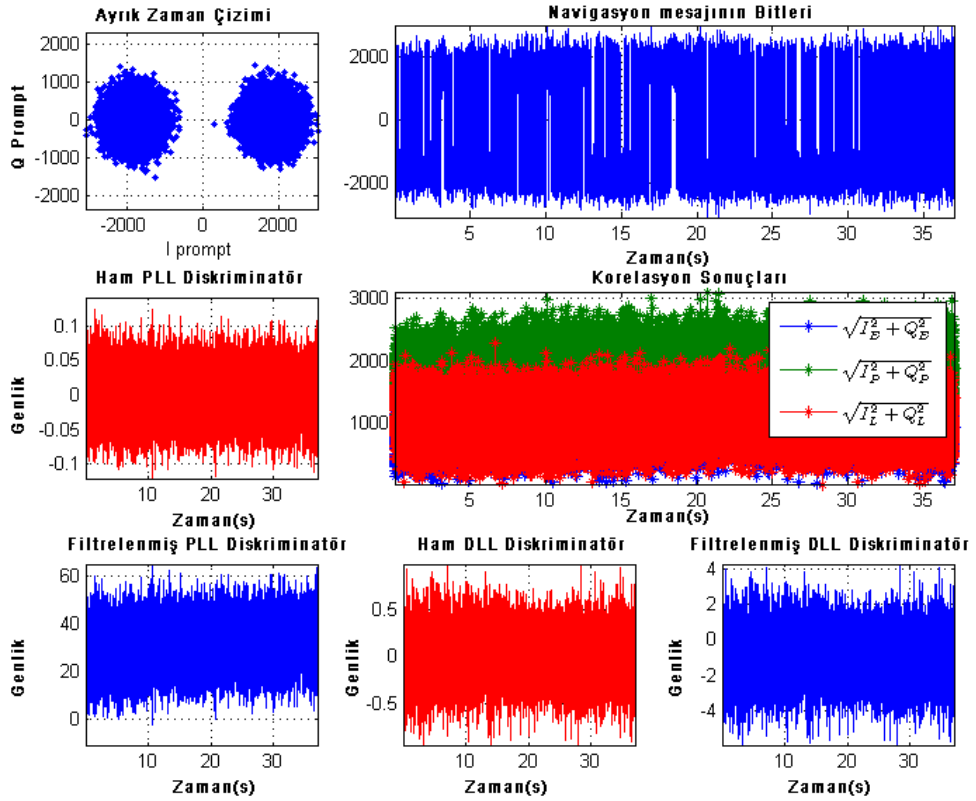
Şekil 5.4: PRN 2 GPS Uydusu verileri izleme sonucu



Şekil 5.5 : PRN 4 GPS Uydusu verileri izleme sonucu



Şekil 5.6 : PRN 9 GPS Uydusu verileri izleme sonucu



Şekil 5.7 : PRN 12 GPS Uydusu verileri izleme sonucu

5.1.2 Yörünge Verileri

GPS uydularından alınan navigasyon mesajından elde edilen yörünge verileri Tablo 5.1' de gösterilmiştir.

Tablo 5.1: GPS uydularından elde edilen yörünge bilgileri

	PRN : 2	PRN : 4	PRN : 9	PRN : 12	PRN : 15	PRN : 17
M_0	-2,5843	0,2232498	0,913404	2,167165	-1,37009	-1,614348
Δn	5,1181e-09	5,2202e-09	4,786e-09	4,4212e-09	4,57411e-09	4,356e-09
e	0,008751	0,0082804	0,0201623	0,003148	0,000861837	0,003812
$(A)^{1/2}$	5153,766	5153,5736	5153,658	5153,5317	5153,6996	5153,5905
$(\text{OMEGA})_0$	-2,05379	-2,0350739	1,0621402	2,1486125	0,0387485	-3,076903
i_0	0,942553	0,9418224	0,9726257	0,9646723	0,95772	0,960085
ω	2,637872	0,3967484	1,441004	-0,8603	-1,18951	-2,785141
OMEGADOT	8,2603e-09	-8,501e-09	-8,661e-09	-7,949e-09	-8,0639e-09	-7,76e-09
IDOT	4,3573e-11	1,4643e-11	2,5001e-12	-2,675e-10	2,47153e-10	4,0e-10
C_{uc}	9,8906e-07	7,1898e-07	3,7252e-08	-5,288e-06	4,7311e-06	2,458e-07
C_{us}	5,2172e-06	5,0105e-06	2,7529e-06	7,6238e-06	8,2701e-06	1,019e-05
C_{rc}	267,71875	270,40625	337,34375	232,90625	221,468	185,812
C_{rs}	15,1875	11,40625	4,6875	-102,1875	89,9687	2,0312
C_{ic}	1,0430e-07	1,1920e-07	-3,073e-07	5,774e-08	-5,2154e-08	7,450e-09
C_{is}	9,313e-08	1,1920e-07	-4,097e-08	1,8626e-08	-1,4907e-08	-1,06e-07
t_{oe}	388800	388800	388800	388784	388800	388800
IODE	45	74	15	11	7	61

Tablo 5.2'de Tablo 5.1'de verilen yörünge verilerinin açıklamaları verilmiştir[1].

Tablo 5.2: GPS uydularından elde edilen yörünge bilgi açıklamaları

M_0	Referans zamanda ortalama anomali (2^{-31} yarı-daire)
Δn	Hesaplanan değer ve hareketin ortalama farkı (2^{-43} yarı-daire/sn.)
e	Dışmerkezlilik, eksantriklik (2^{-33} birimsiz)
$(A)^{1/2}$	Yarım ana eksenin karekökü (2^{-19} metre ^{1/2})
$(\text{OMEGA})_0$	Yörünge ekseninin yükselen düğümünün haftalık boylamı (2^{-31} yarı-daire)
i_0	Referans zamanda eğiklik açısı (2^{-31} yarı-daire)
ω	Yerberi argümanı (perigee) (2^{-31} yarı-daire)
OMEGADOT	Bahar açısı (Right ascension) oranı (2^{-43} yarı-daire/sn)
IDOT	Eğiklik açısı oranı (2^{-29} radyan)
C_{uc}	Enlem argümanının kosinüs harmonik düzeltme teriminin genliği (2^{-29} radyan)
C_{us}	Enlem argümanının sinüs harmonik düzeltme teriminin genliği (2^{-29} radyan)
C_{rc}	Yörünge yarıçapının kosinüs harmonik düzeltme teriminin genliği (2^{-5} metre)
C_{rs}	Yörünge yarıçapının sinüs harmonik düzeltme teriminin genliği (2^{-5} metre)
C_{ic}	Eğiklik açısının kosinüs harmonik düzeltme teriminin genliği (2^{-29} radyan)
C_{is}	Eğiklik açısının sinüs harmonik düzeltme teriminin genliği (2^{-29} radyan)
t_{oe}	Yörünge için referans zaman (2^4 sn)
IODE	Verinin yayın kimliği (8 bit uzunluğunda yörünge verisinin kimliğini belirten sayıdır.)

5.1.3 GPS uydu konumları

Elde edilen navigasyon mesajından GPS uydularına olan uzaklık ve GPS uydularının konumları bulunmuştur. Aşağıdaki tablolarda yapılan ölçümün ilk 10 saniyesi için belirlenen veriler, uyduların X,Y,Z konumları ve alıcı ile aralarında olan sözde mesafe ölçümlerini göstermektedir.

Tablo 5.3: Uyduların (PRN:2 – PRN:4) X,Y,Z koordinatları ve PR (Sözde Mesafe) ölçümleri (m. 10^5)

t (sn)	PRN:2 X	PRN:2 Y	PRN:2 Z	PRN:2 PR	PRN:4 X	PRN:4 Y	PRN:4 Z	PRN:4 PR
1	2,687782	155,096	216,1649	229,3251374	139,0349	214,1847	-81,0961	253,4855
2	2,660962	155,0907	216,1689	229,2570028	139,0356	214,195	-81,0673	253,4065
3	2,634141	155,0855	216,1728	229,1888681	139,0363	214,2054	-81,0385	253,3276
4	2,607319	155,0802	216,1768	229,1207335	139,037	214,2158	-81,0096	253,2486
5	2,580497	155,0749	216,1807	229,0525988	139,0377	214,2262	-80,9808	253,1697
6	2,553674	155,0696	216,1847	228,984281	139,0384	214,2366	-80,952	253,0908
7	2,526849	155,0644	216,1886	228,9161464	139,0392	214,247	-80,9231	253,0118
8	2,500024	155,0591	216,1925	228,8480117	139,0399	214,2574	-80,8943	252,9329
9	2,473199	155,0538	216,1964	228,7798771	139,0406	214,2677	-80,8655	252,8539
10	2,446372	155,0486	216,2004	228,7115592	139,0413	214,2781	-80,8366	252,7752

Tablo 5.4: Uyduların (PRN:9 – PRN:12) X,Y,Z koordinatları ve PR (Sözde Mesafe) ölçümleri (m. 10^5)

t (sn)	PRN:9 X	PRN:9 Y	PRN:9 Z	PRN:9 PR	PRN:12 X	PRN:12 Y	PRN:12 Z	PRN:12 PR
1	161,6663	38,0842	201,1626	207,1147	56,02262	254,53	38,47033	239,4109
2	161,6751	38,10966	201,1512	207,04	56,0175	254,5264	38,50139	239,3336
3	161,6839	38,13513	201,1397	206,9651	56,01238	254,5227	38,53244	239,2563
4	161,6927	38,16059	201,1283	206,8902	56,00725	254,5191	38,56349	239,179
5	161,7016	38,18605	201,1168	206,8152	56,00212	254,5154	38,59454	239,1017
6	161,7104	38,21151	201,1054	206,7405	55,997	254,5118	38,62559	239,0244
7	161,7192	38,23696	201,0939	206,6656	55,99187	254,5081	38,65664	238,9471
8	161,728	38,26242	201,0825	206,5907	55,98674	254,5045	38,68769	238,87
9	161,7369	38,28787	201,071	206,5158	55,9816	254,5008	38,71873	238,7927
10	161,7457	38,31332	201,0596	206,4409	55,97647	254,4971	38,74978	238,7154

Tablo 5.5: Uyduların (PRN:15 – PRN:19) X,Y,Z koordinatları ve PR (Sözde Mesafe) ölçümleri (m. 10^5)

t (sn)	PRN:15 X	PRN:15 Y	PRN:15 Z	PRN:15 PR	PRN:19 X	PRN:19 Y	PRN:19 Z	PRN:19 PR
1	254,7279	68,54369	-31,4908	237,1883	199,2161	-50,675	168,3909	223,3397
2	254,7235	68,54581	-31,5221	237,1216	199,2034	-50,6578	168,4113	223,2635
3	254,719	68,54792	-31,5535	237,0549	199,1908	-50,6406	168,4316	223,1875
4	254,7145	68,55004	-31,5848	236,9883	199,1781	-50,6234	168,4519	223,1113
5	254,71	68,55215	-31,6162	236,9218	199,1654	-50,6063	168,4722	223,0353
6	254,7055	68,55427	-31,6475	236,8551	199,1528	-50,5891	168,4926	222,9591

Tablo 5.5: Uyduların (PRN:15 – PRN:19) X,Y,Z koordinatları ve PR (sözde mesafe) ölçümleri (m. 10^5)

7	254,701	68,55638	-31,6788	236,7884	199,1401	-50,5719	168,5129	222,8831
8	254,6965	68,55849	-31,7102	236,7218	199,1274	-50,5547	168,5332	222,8069
9	254,692	68,56061	-31,7415	236,6551	199,1148	-50,5375	168,5535	222,7309
10	254,6875	68,56272	-31,7728	236,5884	199,1021	-50,5203	168,5738	222,6549

5.2. Toplanan Veriden Kullanıcı Konumunun Hesaplanması

GPS alıcısıyla toplanan veriden 6 tane GPS uydusunun konumu ve sözde mesafeleri kullanılarak kullanıcı konumu belirlemek için aşağıda verilen MATLAB kodu yazılmıştır. 4.4.1 ve 4.4.2 adımlarında açıklanan ve iki adımdan oluşan hesaplamanın ilk adımı kapalı formda kullanıcı konum vektörü ve hata kovaryansı tahmini yapmakta olup, ikinci adımı ise Kalman benzeri bir filtreyle birinci adımda elde edilen sonuçları güncellemeye yöneliktir. Birinci adımın MATLAB implementasyonu şu şekildedir.

```
function [ userPosEstimate, estErrCov, H, R_inverse, pseudoRanges, satellitePos ] =
closedForm( satellitePos, pseudoRanges, currMeasNr )
%CLOSEDFORM Calculates user Position, and of Error Covariance Estimation using input
parameters as
%satellite Positions and PseudoRanges of satellites.
%% Constants
%%The constants for the code below
% n is the number of satellites
n = size(satellitePos,2);
% lastTimeFrame finds the latest pseudoRange measurement, since the
% SatellitePos is the latest result
% currMeasNr = size(pseudoRanges,2);
%% Rearrange satellitePositions and PseudoRanges,as nth be the furthest
% Find the furthest satellite, and make its measurements as "n th" element
% of the pseudorange vector. Also arrange the satellitePos matrix.
maxPseudoRange=0;
for i = 1 : n
    if (pseudoRanges(i,1) ~= Inf) && (pseudoRanges(i,1) > maxPseudoRange)
        maxPseudoRange = pseudoRanges(i,1);
        indexOfTheFurthestSatellite = i;
    end
end
if(indexOfTheFurthestSatellite ~= n)
    for j = 1 : 3
        Temp = satellitePos(j,n);
        satellitePos(j,n) = satellitePos(j,indexOfTheFurthestSatellite) ;
        satellitePos(j,indexOfTheFurthestSatellite) = Temp;
    end
end
for i = 1 : currMeasNr
    Temp = pseudoRanges(n,i);
    pseudoRanges(n,i)=pseudoRanges(indexOfTheFurthestSatellite,i);
    pseudoRanges(indexOfTheFurthestSatellite,i) = Temp;
```



```

    end
end
%% H vector is the (n-1)*4 regressor matrix (column*row)
H = zeros( (n-1) , 4 );
% H vectors first 3 columns
for j = 1 : (n-1) ,
    for i = 1 : 3 ,
        H(j,i) = ( satellitePos(i,n) - satellitePos(i,j) );
    end
end
% H vectors 4 th column
for j = 1 : (n-1) ,
    H(j,4) = ( pseudoRanges(j, currMeasNr)- pseudoRanges(n, currMeasNr) );
end
%% Z is the measurement vector
Z = zeros( (n-1),1 );
for j = 1 : (n-1) ,
    Z(j,1) = 0.5 * ( pseudoRanges(j, currMeasNr).^2 - pseudoRanges(n, currMeasNr).^2 + ...
        satellitePos(1, n).^2 - satellitePos(1, j).^2 + ...
        satellitePos(2, n).^2 - satellitePos(2, j).^2 + ...
        satellitePos(3, n).^2 - satellitePos(3, j).^2 );
end
%%
R_straight = ones(n-1);
for i = 1 : n-1 ;
    R_straight(i,i) = 1+ (pseudoRanges(i, currMeasNr)/pseudoRanges(n, currMeasNr)).^2 ;
end
R_inverse = inv(R_straight);
% u is the user position vector estimate
userPosEstimate = inv(H' * R_inverse * H)* H' * R_inverse * Z;
sigmaSquare = -1 * (pseudoRanges(n, currMeasNr) - userPosEstimate(4) )+
((pseudoRanges(n, currMeasNr) - userPosEstimate(4)).^2 + (2 * (Z' * R_inverse * Z) / (n-
5)))^(1/2);
c = sigmaSquare/2 + (pseudoRanges(n, currMeasNr) - userPosEstimate(4)).^2;
estErrCov = c * sigmaSquare * inv(H' * R_inverse * H);

```

- İkinci adımın implementasyonu ise şu şekildedir:

```

function [ userPositionPositive, estErrCovPositive] = kalmanFilter( userPositionNegative,
estErrCovNegative, H, R, pseudoRanges, satellitePos)
%kalmanFilter updates the user Position Estimate, and Error Covariance Estimations found
with closed Form Solution.
% updates the User Position Estimate, and Error Covariance Estimations using Kalma-like
filter.
% Inputs are User Position Estimation, Error Covariance Estimations,
% satellite Positions, PseudoRanges of Satellites, and H, R matrixes
% which are constructed in the Closed Form Step.
% The number of satellites in view
n = size(satellitePos,2);
% vn pseudoRange Measurement vector is calculated regarding the initial user
% position estimate and clock error estimate.
vn = zeros(n , 1);
userPositionPositive = userPositionNegative;
estErrCovPositive = estErrCovNegative;

```

```

for i = 1 : n
    vn(i) = pseudoRanges(i) - ( userPositionPositive(1) - satellitePos(1,i)).^2 + ...
        (userPositionPositive(2) - satellitePos(2,i)).^2 + ...
        (userPositionPositive(3) - satellitePos(3,i)).^2).^1/2 - ...
        userPositionPositive(4);
end
% V is the (n-1) error vector
V = zeros(n-1 , 1);
for i = 1 : (n-1)
    V(i) = pseudoRanges(n) * vn(n) - pseudoRanges(i) * vn(i) - userPositionPositive(4) *
(vn(n) - vn(i))...
        + 1/2 * ( vn(i).^2 - vn(n).^2 );
end
% M Vector to calculate Z
M = eye(n-1) - H * inv( H' * R * H ) * H' * R;
% Z return difference vector
Z = M * V;
% variance of error
varyans = userPositionPositive(4) - pseudoRanges(n) + ( pseudoRanges(n) -
userPositionPositive(4)).^2 + 2*( Z' * R * Z)/(n-5) ).^(1/2);
% tempUserSatelliteLength is the temp value. It is needed on more than one points, so in
order
% not to calculate many times (for performance), it is stored in
% memory
tempUserSatelliteLength = ((userPositionPositive(1) - pseudoRanges(1)).^2 +
(userPositionPositive(2) - pseudoRanges(2)).^2 + (userPositionPositive(3) -
pseudoRanges(3)).^2).^1/2);
Zn = pseudoRanges(n)+ ...
(...
    ((userPositionPositive(1) - pseudoRanges(1))*pseudoRanges(1) +
(userPositionPositive(2) - pseudoRanges(2))*pseudoRanges(2) + (userPositionPositive(3) -
pseudoRanges(3))*pseudoRanges(3))...
    /...
    tempUserSatelliteLength...
);
% h is the regressor vector
h(1,1) = ( userPositionPositive(1) - pseudoRanges(1) ) / tempUserSatelliteLength;
h(2,1) = ( userPositionPositive(2) - pseudoRanges(2) ) / tempUserSatelliteLength;
h(3,1) = ( userPositionPositive(3) - pseudoRanges(3) ) / tempUserSatelliteLength;
h(4,1) = 1;
p = varyans * (pseudoRanges(n) - userPositionPositive(4))*inv( H' * R * H ) * H' * R *
ones(n-1,1);
% Y is the intermediate variable , preupdate covariance matrix
Y = estErrCovPositive + ( ( h' * estErrCovPositive * h - 1 )/( 1 - p'*h).^2 ) * p * p' + (
estErrCovPositive * h * p' + p * h' * estErrCovPositive ) / (1 - p'*h);
% K is the Kalman Filter gain
K = ( 1 / (1 - p'*h) ) * ( Y * h - p ) / ( 1 + h' * Y * h ) ;
% Iteration is needed to update the user Position and error covariance
% values. 3 iterations are sufficient regarding to the tests made.
for iterate = 1 : 3
    userPositionPositive = userPositionPositive + K * ( Zn - h' * userPositionPositive);
    estErrCovPositive = ( eye(4) - ( (1 - p' * h) * K + p) * h' ) * Y;
end

```

6. SAYISAL SONUÇLAR VE DEĞERLENDİRME

İki farklı konumdan, biri İstanbul Akatlar bölgesinden, diğeri İTÜ Elektrik-Elektronik fakültesinden olmak üzere toplanan GPS verisi ile kapalı form konumlama algoritmasının başarımı incelenmiştir ve iteratif en küçük kareler yöntemiyle karşılaştırılmıştır. Elde edilen sonuçlar, ölçümün yapıldığı konumun gerçek koordinatları üzerinden değerlendirilip sonuçların enlem ve boylam düzleminde hataları bulunmuştur.

Tablo 6.1: Kapalı form konumlama algoritması ve iteratif en küçük kareler yönteminin sonuçlarının karşılaştırılması

	Akatlar	Hata ($\sqrt{\Delta X^2 + \Delta Y^2}$)	Maslak	Hata ($\sqrt{\Delta X^2 + \Delta Y^2}$)
Kapalı Form 1. Adım	Enlem: 41 derece 4 dk. 49.6003 sn. Boylam: 29 derece 1 dk. 39.4817 sn. Yükseklik: 53.8564 m.	17.467 m.	Enlem: 41 derece 6 dk. 15.8929 sn. Boylam: 29 derece 1 dk. 30.4342 sn. Yükseklik: 245.273 m.	21.719 m.
Kapalı Form 2. Adım (Kalman)	Enlem: 41 derece 4 dk. 49.6031 sn. Boylam: 29 derece 1 dk. 39.4785 sn. Yükseklik: 52.7878 m.	17.439 m.	Enlem: 41 derece 6 dk. 15.9064 sn. Boylam: 29 derece 1 dk. 30.4405 sn. Yükseklik: 243.5867 m.	21.644 m.
İteratif En Küçük Kareler	Enlem: 41 derece 4 dk. 48.9669 sn. Boylam: 29 derece 1 dk. 37.4702 sn. Yükseklik: 21.5 m.	23.4 m.	Enlem: 41 derece 6 dk. 15.4685 sn. Boylam: 29 derece 1 dk. 30.0686 sn. Yükseklik: 3.6 m.	23.349 m.

Tablo 6.1'den görüldüğü gibi, kapalı form konumlama algoritması konum bulma açısından daha iyi başarımlar sunmaktadır.

Kapalı form konumlama algoritması, iterasyon gerektirmediğinden, ILS'e göre daha az hesaplama yüküne ihtiyaç duyar. Tablo 6.2 toplanan verilerle ILS ve kapalı form konumlama algoritmasının çalışma zamanını karşılaştırmaktadır. Karşılaştırma yapılan ILS algoritması [1] nolu kaynakta belirtilen implementasyondur. Çalışma zamanı testi P4 2.8 Ghz. mikroişlemcili bir bilgisayarda yapılmıştır, her iki algoritma da onar kez farklı verilerle çalıştırılıp, çıkan çalışma zamanı sonuçlarının ortalaması alınmıştır.

Tablo 6.2: Kapalı form konumlama algoritması ve iteratif en küçük kareler yönteminin çalışma zamanı karşılaştırması

Kapalı Form 1. Adım	Kapalı Form 2. Adım (Kalman)	Kapalı Form (Toplam)	İteratif En Küçük Kareler
0.23696 msn.	0.88239 msn.	1.11935 msn.	7.52 msn.

ILS'in kullanıcı konumu belirlemek için 4 tane uydudan yeterli sinyal seviyesi alması gerekiyken, kapalı form konumlama algoritmasında en az 6 uduya ihtiyaç vardır.

Kapalı Form Konumlama algoritmasını öneren kaynakta Kalman benzeri filtre ile yapılan güncelleştirmenin yapılan denemelerde 2-3 iterasyon sonunda doyuma ulaştığı, dolayısıyla 3 iterasyonun uygulamanın yeterli doğrulukta sonuç elde etmek için yeterli olduğu önerilmektedir[5]. Ancak bu çalışma kapsamında yapılan denemelerde ilk Kalman güncelleştirmesinden sonraki iterasyonlarda elde edilen kullanıcı konumunun değişmediği görülmüştür.

KAYNAKLAR

- [1] **Borre K., Akos D.M., Bertelsen N., Rinder P., Jensen S.H.**, 2007. A Software Defined GPS and Galileo Receiver, Birkhäuser, Boston.
- [2] **Hofmann-Wellenhof, B., Lichtenegger, H. and Collins, J.**, 2001. Global Positioning System Theory and Practice, Springer-Verlag Wien, New York
- [3] **Kaplan, E.D.**, 1996. Understanding GPS: Principles and Applications, Artech House, London..
- [4] **Xu, G.**, 2003. GPS Theory, Algorithms and Applications, Springer-Verlag, Berlin Heidelberg
- [5] **Nguyen, T.Q.**, 2007. Efficient GPS Position Determination Algorithms, *PhD Thesis*, Air Force Institute of Technology, Ohio.
- [6] **Nardi, S.**, 1998. Improved Mathematical Modeling For GPS Based Navigation, *MS Thesis*, Air Force Institute of Technology, Ohio.
- [7] **Carlos, C.R.**, 1999. An Efficient GPS Position Determination Algorithm, *MS Thesis*, Air Force Institute of Technology, Ohio.
- [8] **Leva, J. L.**, 1996. An Alternative Closed-Form Solution to the GPS Pseudo Range Equations, *IEEE Transaction on Aerospace and Electronic Systems*, Vol. **32(4)**, 1430-1439.
- [9] **Krause, L. O.**, 1987. A Direct Solution to GPS Type Navigation Equations, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. **23(2)**, 223-232.
- [10] **Abel, J. and Chaffee J.**, 1991. Existence and Uniqueness of GPS Solutions, *IEEE Transactions on Aerospace and Electronic Systems*, **27(6)**, 952-956.
- [11] **Chaffee, J. and Abel J.**, 1994. On the Exact Solutions of Pseudorange Equations, *IEEE Transactions on Aerospace and Electronic Systems*, **30(4)**, 1021-1030.
- [12] **Hoshen, J.**, 1996. The GPS Equations and the Problem of Apollonius, *IEEE Transactions on Aerospace and Electronic Systems*, **32(3)**, 1116-1124.

ÖZGEÇMİŞ

1982 Eskişehir doğumlu olan Saygın Topatan ilk ve orta öğrenimini Eskişehir’de tamamlamıştır. 2000 yılında Eskişehir Fatih Fen Lisesi’ni bitirerek İstanbul Teknik Üniversitesi, Elektrik Elektronik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü’nde lisans eğitimine başlamıştır. 2005 yılında İstanbul Teknik Üniversitesinde Telekomünikasyon Mühendisliği programında Yüksek Lisans eğitimine başlamıştır. Bilgi Teknolojileri ve yazılım geliştirme alanında çeşitli şirketlerde çalışmıştır. Halen mühendislik mesleğini sürdürmektedir.