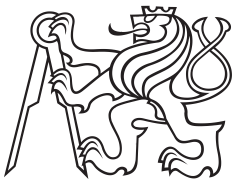
Master Thesis



Czech Technical University in Prague

F3

Faculty of Electrical Engineering
Department of Measurement

Signals of Opportunity Positioning Using LEO Satellites

Bc. Vojtěch Voska

Supervisor: Ing. Jiří Svatoň, Ph.D.
Programme: Aerospace Engineering
Field: Avionics
May 2024

MASTER‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Voska Vojtěch** Personal ID number: **474387**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Measurement**

Study program: **Aerospace Engineering**

Branch of study: **Avionics**

II. Master’s thesis details

Master’s thesis title in English:

Signals of opportunity positioning using LEO satellites

Master’s thesis title in Czech:

Využití signálů družic na nízké oběžné dráze k určování polohy

Guidelines:

The aim of the work is the study and experimental demonstration of SoP (Signals of Opportunity) positioning (by the Doppler shift method or another) using the signal reception of one of the communication satellite systems (e.g., Iridium, Starlink, Orbcomm, and others).
First, carry out an analysis of the existing methods described in the literature. Analysis of presumed parameters (precision, availability, other conditions, and measurement time) will be included. GNSS will be used as a performance benchmark. Based on the analysis, select the appropriate communication system(s) and signal processing method regarding the existing laboratory equipment (especially SDR receiver equipment - frequency range, bandwidth). Subsequently, demonstrate, using the selected system and method, signal reception, Doppler frequency measurement, and, with the help of known elements of the satellite orbit (e.g., TLE), the calculation of the stationary user's position. Matlab or Python scripts are assumed to be the work output. Finally, verify the functionality and compare achieved results with the existing literature and with conventional GNSS (from the perspective of measurement precision and availability).

Bibliography / sources:

1] Jardak, N.; Adam, R. Practical Use of Starlink Downlink Tones for Positioning. Sensors 2023, 23, 3234, https://doi.org/10.3390/s23063234
2] KHALIFE, Joe J.; KASSAS, Zaher M. Receiver design for Doppler positioning with LEO satellites. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, p. 5506-5510.
3] Farhangian, F.; Landry, R., Jr. Multi-Constellation Software-Defined Receiver for Doppler Positioning with LEO Satellites. Sensors 2020, 20, 5866.
4] H. Benzerrouk, Q. Nguyen, F. Xiaoxing, A. Amrhar, A. V. Nebylov and R. Landry, "Alternative PNT based on Iridium Next LEO Satellites Doppler/INS Integrated Navigation System," 2019 26th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS), St. Petersburg, Russia, 2019, pp. 1-10.
5] VEJRAŽKA, F., ŠEDIVÝ, P., Radiové systémy, soubor přednášek k předmětu, ZS 2012/2013.

Name and workplace of master's thesis supervisor:

Ing. Jiří Svatoň, Ph.D. Department of Radioelectronics FEE

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **05.09.2023** Deadline for master's thesis submission: **24.05.2024**

Assignment valid until:
by the end of summer semester 2024/2025

Ing. Jiří Svatoň, Ph.D. Supervisor's signature	Head of department's signature	prof. Mgr. Petr Páta, Ph.D. Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

.	
Date of assignment receipt	Student's signature

Acknowledgements

I would like to thank my family for all their support, patience and understanding.

I would also like to thank my friends for their advice during the times I felt stuck, and for their help in the tedious task of proofreading this thesis.

Last but not least, I would like to thank my supervisor for his support, advice, patience and quick replies.

Declaration

I, the author, hereby declare that I have completed this thesis independently without illegitimate assistance, and that I have stated all used literature in accordance with the Methodical Instruction no. 1/2009, concerning ethical principles of authoring a university thesis at Czech Technical University in Prague.

In Prague, 23 May 2024 .....

This thesis uses several Free and Open Source programs. Attributions per the respective program's license are within the thesis text, where appropriate, and all are listed in Appendix.

Abstract

This work is a study and an experimental demonstration of satellite-aided Signals of Opportunity (SoP) positioning using the Doppler shift method. This method of positioning uses the reception of satellite signals which were transmitted for other purposes than navigation to determine the position of the user by means of measuring the Doppler shift of the signal. The Doppler method is discussed and some Low Earth Orbit satellite communication systems potentially useful for SoP positioning are described. A survey of current research on the topic is conducted and the SoP systems developed therein are compared to dedicated navigation systems. A full-stack SoP navigation system, capable of signal capture, satellite position determination and static user position estimation is developed and described in detail. The system is then experimentally evaluated and its parameters analysed.

Keywords: Signals Of Opportunity Navigation, Doppler Positioning, Doppler Shift, Iridium, Orbcomm, Globalstar, Starlink, NAVSAT, Argos, GNSS, Phase-Shift Keying Modulation, Satellite Constellation, LEO Satellites, Software-Defined Radio

Supervisor: Ing. Jiří Svatoň, Ph.D.

Abstrakt

Tato práce je studií a experimentální demonstrací systému určení polohy na základě zpracování oportunních signálů Dopplerovskou metodou. Tato metoda určení polohy využívá příjmu signálů vyslaných za jiným účelem než navigace a měření jejich Dopplerovské frekvence. Dopplerovská metoda je popsána a některé satelitní systémy na nízké orbitě Země potenciálně použitelné pro oportunní navigaci jsou diskutovány. Je provedena rešerše soudobého výzkumu v této oblasti, a vyvinuté systémy jsou porovnány s dedikovanými satelitními navigačními systémy. Je detailně popsán vývoj plnohodnotného oportunního navigační systému, schopného přijmout a zpracovat signál, spočítat polohu vysílacího satelitu a odhadnou polohu statického uživatele. Tento systém je pak experimentálně prověřen a jeho parametry analyzovány.

Klíčová slova: navigace příležitostnými signály, Dopplerovská navigace, Dopplerův jev, Iridium, Orbcomm, Globalstar, Starlink, NAVSAT, Argos, GNSS, fázové klíčování, satelitní konstelace, LEO satelity, softwarově definované rádio

Překlad názvu: Využití signálů družic na nízké oběžné dráze k určování polohy

Contents

Figures

Tables

Chapter 1

Introduction

The modern world is increasingly reliant on navigation systems in all areas and industries, from transport through agriculture to security. Be it precision agriculture, geo-fencing, the guidance of aircraft, ships, autonomous vehicles, logistic coordination and tracking, emergency search and rescue, house arrest control, all kinds of military operations, and of course personal navigation, they all share the need for a quick, reliable and accurate positioning.

The need perhaps first arose in military context, and indeed the first navigation systems more sophisticated than a human with a map and a compass or a sextant were developed by the various militaries of the world in the 20$^{\text{th}}$ century, but the civilian sector soon caught up. Systems such as Loran C, Non Directional Beacon or VHF Omnidirectional Radio Range (VOR) were ground based radio navigation systems. From the middle of the century the access to space allowed for satellite-aided navigation systems, such as Transit, Dora, and then the Global Positioning System (GPS) and Global Navigation Satellite Systems (GNSS) in general.

Today, the GNSS provide the bulk of the navigation services for the applications outlined above. GNSS are commonly based on satellites in Middle Earth Orbit (MEO) around 20 000 km above the surface, although some systems utilise Geosynchronous Earth Orbits (GSO) too, with constellations of around 30 satellites. The large orbital height allows a relatively small constellation to have several satellites simultaneously visible from the surface of the Earth. That is required, as the navigation method used by the GNSS, the range method, requires at least four simultaneously visible satellites. However, it carries a significant drawback - the received signal is very weak, which not only requires complex signal processing for acquisition, but also makes the GNSS signals susceptible to jamming and spoofing[1].

In the transport context, perhaps more than in any other, the consequences of a spoofing or a jamming attack are significant, ranging from disruption of the flow of goods and people to a possible harm to human lives. The availability of powerful hardware and software for spoofing lowers the barriers potential adversaries face in an attack, whereas the need for the architecture of the GNSS to remain compatible makes adaptation difficult[7].

[1] *Jamming* means blocking out a signal with a stronger one, whereas *spoofing* means emulating the signal by another in order to deceive the receiver

However, there need not be malice involved - simple unavailability of the GNSS, poor signal quality or insufficient coverage may suffice to cause disruption. GNSS are run by governments and government-like entities, who may decide to limit the service e.g. in times of war, the GNSS signals are easily degraded by buildings or dense trees, can be jammed by radar, radios or electromagnetic interference and the coverage is somewhat geographically limited (e.g. GPS performance degrades in the polar regions[7, 8]).

Solutions were developed to mitigate the vulnerabilities in GNSS. Coupling with an Inertial Navigation System (INS) is one, physical layer authentication[9] or crowd-sourcing, i.e. the fusion of information from multiple sources, are some of the other[7]. However, another solution might be to rely on an independent navigation system, one with much stronger signals and much more diverse signal sources - Signals of Opportunity (SoP) navigation using communication satellites.

The access to space allowed for satellite communication systems to be deployed, and the industry is thriving today. There are multiple constellations, some with high tens of satellites, some with high thousands (see chapter 3 for details). As a result, the frequency space is filled by communication transmissions, which can be readily used for navigation.

In the context of this work, Signals of Opportunity positioning[2] is the process of finding the user position by the use of signals which were not designed nor transmitted for the purpose of positioning. In this work, only satellite SoP are considered, although SoP positioning can be performed over signals from any source.

This thesis studies and experimentally demonstrates the SoP positioning using the Doppler method.

1.1 The structure of the thesis

In the remainder of this chapter, SoP positioning is explained, its advantages and challenges are discussed and some use cases are presented.

In chapter 2, the theoretical foundation for the Doppler shift method is laid. Some ways of implementation of the method in the context of SoP positioning are discussed. A process of capturing a satellite signal is described, as is a way of determining a satellite position. Lastly, the coordinate frames of reference relevant in this work are described.

In chapter 3, some Low Earth Orbit satellite systems potentially usable for SoP navigation are described - the Iridium NEXT, Orbcomm, Globalstar and Starlink constellations. The constellation, orbital and signal parameters are listed and the signal types are described, with a particular focus on the signals with properties useful for SoP navigation. Lastly, a satellite visibility analysis is conducted for select systems.

[2]Within the thesis, *positioning* and *navigation* are used somewhat interchangeably. The system developed by this work deals exclusively with a static user, and is thus a *positioning* system, but other system referenced are capable of working in dynamic mode as well and the findings of this thesis are applicable to dynamic modes as well.

The chapter 4 is a survey of SoP systems in the current research. The methods used, signals exploited and results achieved are described. Additionally, two dedicated navigation systems which use the Doppler shift method are depicted, to provide a basis of comparison. The SoP systems are then compared against GNSS.

The chapter 5 describes in detail the development of a full stack navigation system. The implementation of satellite signal capture, demodulation and decoding is described. Satellite identification system is developed, and the algorithm for user position calculation is devised.

In the chapter 6, the developed navigation is tested and its parameters evaluated.

1.2 Signals of Opportunity navigation challenges

Due to the nature of the signals, SoP positioning presents several unique challenges.

Firstly, the signal transmission time is generally not known. This limitation rules out the application of the pseudo-range method of navigation, employed e.g. by GPS.

Secondly, the clocks onboard the satellites transmitting the SoP signals generally do not need to be as accurate as GNSS clocks. This means that the transmission frequency of SoP signals suffers from larger errors due to clock offset and drift. This also precludes using any sort of timestamp within the signal for the pseudo-range method, even if it was present and decoded.

Thirdly, the identity or position of the transmitting satellite are generally not included in the SoP signal message. This places additional requirements on an SoP navigation system and introduces additional errors.

Fourthly, the signals are not guaranteed to be available globally, and can be selectively turned off by the system operator.

Fifthly, the SoP do not generally contain any parameters which can be used to synchronise the receiver clock, correct the satellite orbital model or the ionospheric propagation model, as is the case for e.g. GPS[10, p. 22].

For these and other reasons, SoP navigation systems generally use the Doppler shift method (see chapter 4), which does not rely on message transmission time. The Doppler method (see chapter 2) does not rely on the signal reception time either, but the satellite position prediction likely does (see chapter 5 for a discussion about various implementations). As a result, the challenges mean that an SoP navigation system will likely never be as accurate as a GNSS.

1.3 Signals of Opportunity navigation advantages and possible use cases

The nature of the SoP also provides some significant advantages over GNSS.

Firstly, the signals mainly originate from LEO satellites, orbiting around 800 km above the surface (see chapter 3), which is around 20 to 30 times closer than the MEO GNSS satellites. This results in the SoP being much stronger than GNSS signals (at least by an order of magnitude, although some sources claim up to four orders of magnitude[11]), and therefore much more capable of penetrating tree cover or buildings. Indeed, the capability of obtaining a fix with SoP navigation, but not with GPS was demonstrated in [3].

Secondly, the signals originate from many different satellites, with different orbits and transmission frequencies, which results in frequency and direction-diverse signal set. This can be used to eliminate e.g. ionospheric refraction. It also makes jamming the signals very difficult.

Thirdly, the satellites are owned and operated by many entities, making the overall system more diverse and thus more resilient to outages.

The SoP navigation is not only useful as a standalone system, but it can be readily used to augment other systems, be it GNSS, or by fusion with INS or magnetometer etc.

These traits make the SoP readily usable as a backup or augmentation navigation system, especially in GNSS-denied conditions. For instance, it can be used as an emergency localisation system in aircraft, improving safety or aiding in emergency search and rescue. A case study may be the loss of Malaysia Flight MH370 and the subsequent search operation, hampered by a lack of location data about the aircraft[12].

Furthermore, the SoP navigation system can aid in navigation reliability and safety. In this form, the SoP system can function as a part of a spoofing-detection system, providing a second source of navigation information. Resistance to spoofing is vital for ships relying heavily on autonomous navigation, as GNSS spoofing is one of the dangers faced by mariners from sea pirates[13].

Chapter 2

Doppler frequency-based satellite-aided positioning

This chapter lays the groundwork for the system designed in the latter part of this work. The Doppler effect and the Doppler frequency-based method, henceforth referred to as the *Doppler method*, are described, with some implementations based on the reviewed literature. Furthermore, this chapter discusses the process of satellite signal capture, which is an essential part of satellite-aided navigation. A way to determine satellite position is presented, and frames of reference used for navigation in this work are described.

2.1 The Doppler effect

The Doppler effect is the change in the frequency of a wave as perceived by an observer who is moving relative to the source of the wave. The received frequency can be expressed as

$$f_R = \frac{c \pm v_R}{c \pm v_T} f_0 \tag{2.1}$$

where c is the speed of the wave in the medium, v_R and v_T are the respective scalar velocities of the receiver and transmitter relative to the medium (the direction depends on the sign convention), and f_0 is the transmitted frequency.

To simplify calculations, the Equation 2.1 can be simplified:

$$f_R = \frac{c + v_R}{c + v_T} f_0$$
$$\text{dividing by } c : f_R = \left(\frac{1 + \frac{v_R}{c}}{1 + \frac{v_T}{c}}\right) f_0$$

Assuming $c \gg v_T$ and therefore $\frac{v_T}{c} \ll 1$, a substitution using a truncated Taylor series expansion $\frac{1}{1+x} \approx (1 - x)$ can be made:

$$f_R = \left(\frac{1}{1+\frac{v_R}{c}}\right) \times \left(\frac{1}{1-\frac{v_T}{c}}\right) f_0$$
$$f_R = \left(1 - \frac{v_T}{c} + \frac{v_R}{c} - \frac{v_R v_T}{c^2}\right) f_0$$
$$\frac{v_R v_T}{c^2} \to 0 : f_R = \left(1 + \frac{v_R - v_T}{c}\right) f_0$$
$$\text{substituting } \Delta v = v_R - v_T : f_R = \left(1 + \frac{\Delta v}{c}\right) f_0 \tag{2.2}$$

Δv is the relative velocity of the receiver and transmitter, alternatively labelled as range rate $\dot{\rho}$. By convention it is positive if the range between the transmitter and receiver decreases.

The Doppler shift $f_D = f_R - f_0$ can be expressed from Equation 2.2 as:

$$f_D = \frac{\Delta v}{c} f_0 \tag{2.3}$$

In vector form, the range rate $\dot{\rho}$ can be expressed as the projection of the relative velocity vector $\vec{v_T} - \vec{v_R}$ on the relative position vector $\vec{r_T} - \vec{r_R}$:

$$\dot{\rho}_i = (\vec{V_T} - \vec{V_R}) \frac{\vec{r_T} - \vec{r_R}}{||\vec{r_T} - \vec{r_R}||} \tag{2.4}$$

Thus, for a stationary receiver ($\vec{v_R} = \vec{o}$), the Doppler frequency can be expressed as

$$f_D = f_0 \frac{1}{c} (\vec{v_T} - \vec{v_R}) \frac{\vec{r_T} - \vec{r_R}}{||\vec{r_T} - \vec{r_R}||} \tag{2.5}$$

The assumption of stationary receiver requires the positions to be in ECEF frame, as otherwise the receiver rotates with the Earth.

2.2 The Doppler method

The Doppler method is a method of radio-frequency navigation which calculates the user position from measurements of the Doppler shift of the received signal and from the knowledge of the transmitter position.

When a satellite passes over the surface of the Earth with a tangential velocity v, transmitting a signal on a frequency f_0 (see Figure 2.1), a user on the surface measures a Doppler-shifted frequency f_R (see Equation 2.2).

The relative velocity of the satellite and the user, v_T, is the component of the satellite velocity vector, which points in the direction of user. It is offset from the satellite velocity by the angle α. It can be expressed as

$$v_T = v \ \cos\alpha$$

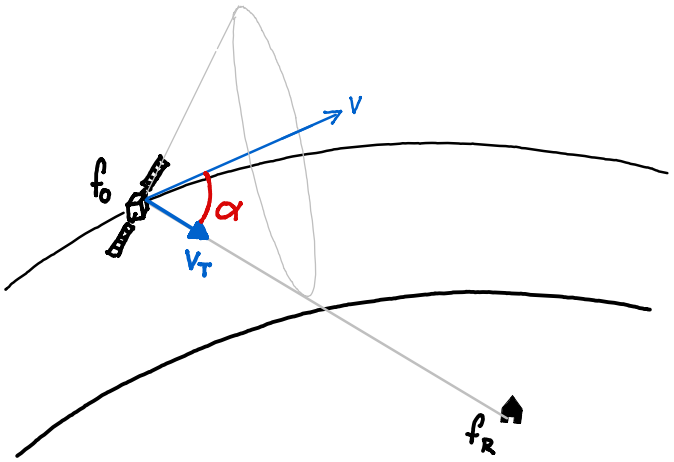


Figure 2.1: The Doppler method principle

Thus, the Doppler shift can be expressed as

$$f_D = f_R \frac{v}{c} \cos\alpha \tag{2.6}$$

$$\text{thus } \cos\alpha = \frac{f_D}{f_R} \frac{c}{v} \tag{2.7}$$

The user is located somewhere on an infinite cone (the *isodoppler cone*) with the centre at the position of the satellite at the time of transmission, the axis in the direction of v and an opening angle 2α.

If the measurement is carried multiple times, a *Doppler curve* can be obtained - a curve of Doppler shifts in time (t, f_D). The user is located at the intersections of the respective isodoppler cones. If the user is assumed to be on the surface, an intersection with a sphere representing the surface can be considered.

Furthermore, it is apparent from Equation 2.7, that if $f_D = 0$, $cos\alpha = 0$ and therefore $\alpha = 90°$ - the isodoppler curve at that point becomes a plane perpendicular to the satellite orbital velocity. When projected on the surface of the Earth, the plane appears as a line perpendicular to the satellite track.

It is important to note that for a pass of a single satellite, the Doppler method always yields at least two results (one actual and one shadow) symmetrical along the satellite ground track (see Figure 2.2 or Figure 5.8 for the symmetrical solution being found in practice). Additional satellite measurements or other means are necessary to resolve the symmetry problem and determine which position is the correct one.

2.3 Implementations the Doppler method

A review of existing work reveals two general ways of implementing the Doppler method. They are based on variations of Equation 2.5.

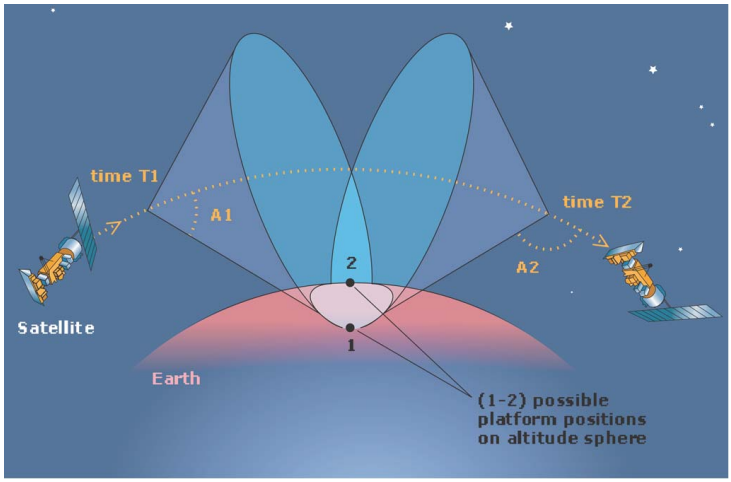


Figure 2.2: Illustration of the Doppler method symmetry problem[1]

2.3.1 The Doppler curve fitting method

The Doppler curve fitting method is based on measuring the frequency emitted by a satellite, thus obtaining a Doppler shift curve. This measured curve is then compared using the least-squares fit to the trial Doppler curves generated based on trial user position. The trial position is corrected until the best fit is found. The trial Doppler curve is calculated based on Equation 2.5. This method had been used by the Transit navigation system[14].

2.3.2 The Observation Model method

The second method is the most common in the reviewed research. It is based on creating an observation model and injecting Doppler measurements into an Extended Kalman Filter (EKF) to estimate the user location[15, 16, 12].

The range rate measurement of i^{th} satellite was usually modelled in the research as

$$\dot{\rho}_i = \frac{cf_{D_i}}{f_R} = (V_i - V)\frac{r_i - r_0}{||r_i - r_0||_2} + c\dot{\delta}t - c\dot{\delta}t^i + \sigma \tag{2.8}$$

where $c\dot{\delta}t$, $c\dot{\delta}t^i$ is the clock drift in $\mathrm{m\,s^{-1}}$ of the receiver and satellite, respectively, and σ is Gaussian white noise[15].

2.4 Maximum Doppler shift

An estimate of the maximum of Doppler shift which can occur in a captured signal frequency is a useful parameter for signal acquisition and for transmission channel identification, where it can serve as a limit.

The largest shift occurs when $\cos\alpha = 1$ in Equation 2.6, that is, when the satellite radius vector is perpendicular to that of the receiver. However, in this situation, the satellite in LEO is invisible to the receiver. More practically,

System	h (km)	f_R (MHz)	f_{Dmax} (kHz)
Iridium	781	1626	36.1
Orbcomm	715	138	3.1
Globalstar	1414	2500	48.9

Table 2.1: Maximum Doppler shift for select LEO satellites (calculated)

the largest Doppler shift occurs when the satellite is first (and last) visible to the receiver. For a ground-based receiver with no obstacle in sight, this happens when the satellite intersects a plane tangential to the Earth surface at the receiver position.

In this special case, the angle α is equal to the angle between the satellite and user radius vectors. Thus,

$$\cos\alpha = \frac{R_E}{R_E + h}$$

where R_E is the radius of the Earth and h is the orbital altitude of the satellite.

Orbital velocity for a circular orbit is approximately

$$v \approx \sqrt{\frac{\mu}{R_E + h}}$$

where μ is the standard gravitational parameter. Thus, substituting the equation above into Equation 2.6, the maximum Doppler shift depends only on the orbital altitude and transmission frequency:

$$f_{Dmax} \approx f_R \frac{1}{c} \sqrt{\frac{\mu}{R_E + h}} \frac{R_E}{R_E + h} \tag{2.9}$$

The calculated maximum Doppler shift for the satellites considered in the LEO system survey in chapter 3 is in Table 2.1.

2.5 Capturing a satellite signal

One of the basic principles of capturing a satellite signal burst, extensively used in the reviewed literature, is based on thresholding the signal power level.

First, the raw captured signal is amplified by a low noise amplifier (LNA) and is squared once or twice (2^{nd} or 4^{th} power) to increase the prominence of dominant peaks. Then, the Fast Fourier transform (FFT) or a Power Spectral Density (PSD) function is applied with a windowing function (e.g. Hamming or Blackman-Harris). The resultant spectrum is then compared to a threshold representing the noise level. Peaks whose power level is stronger than the threshold and whose frequency falls within the plausible range for the given signal type are considered to be indicative of a present signal. The peak rise and fall times are noted (usually in terms of sample number rather than seconds).

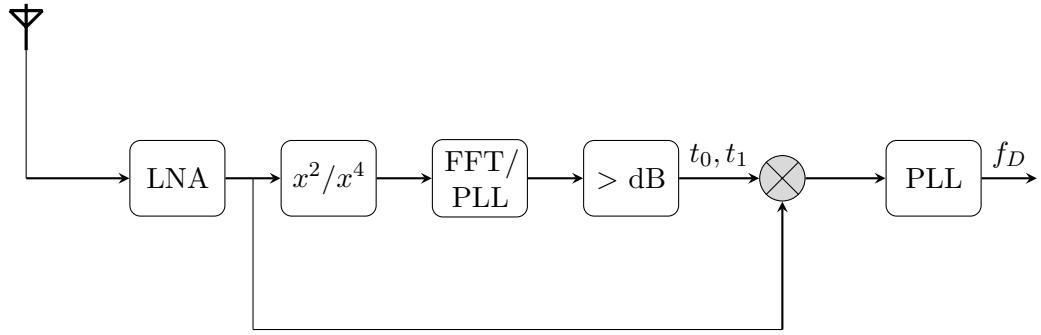


Figure 2.3: General process of capturing a satellite signal

Card #	Satellite Number	Class	International Designator (Year, Lch#, Piece)	Yr	Epoch Day of Year (plus fraction)	Mean motion derivative (rev/day /2)	Mean motion second derivative (rev/day2 /6)	Bstar (/ER)	Eph	Elem num	Chk Sum
1	16609	U	86017A	93	352.53502934	.00000789	00000-0	10529-3	0	342	

Card #	Inclination (deg)	Right Ascension of the Node (deg)	Eccentricity	Arg of Perigee (deg)	Mean Anomaly (deg)	Mean Motion (rev/day)	Epoch Rev	Chk
2	51.6190	133.3400	0005770	102.5680	257.5950	15.591140700447869		

Figure 2.4: Structure of a TLE set[2]

The samples within those times enter a Phase Lock Loop (PLL) to refine the frequency estimate. The PLL is composed of a Numerically Controlled Oscillator (NCO), integrator functions, phase detector and a loop filter. A typical setup is shown in the block diagram in Figure 2.3[16, 17, 18].

2.6 Calculating satellite position

The Doppler method requires the knowledge of the position of the transmitting satellite, which is then used as a reference. Unless the satellite itself transmits its position (which is the case for Iridium Ring Alert messages, albeit in a low resolution), the position needs to be calculated on the user side. To do this, the satellite orbit needs to be known and a propagation model used to calculate the satellite position at a given time.

A widely used approach utilises Two Line Element (TLE) sets, which contain orbital parameters for satellites, and the Simplified General Perturbations-4 (SGP4) model for propagation. TLEs are regularly updated by the North American Aerospace Defense Command (NORAD) and are publicly available via e.g. the CelesTrak project website[19]. An analysed example of TLE is shown in Figure 2.4. A TLE is valid for a given time, which is called an *epoch*.

The SGP4 model is an orbit propagation model developed by the U.S. Air Force in the 1960s and further refined e.g. by Vallado et al.[20]. It is compatible with TLEs and commonly used together. The perturbation calculations are simplified - the Earth's gravitational model is truncated, the atmospheric model is a static density field with exponential decay, and third-body influences are modelled only partially[21, p. 698]. The SGP4

model works in the TEME coordinate frame (see section 2.7).

The accuracy of the model was evaluated in [22]. It was concluded that for a LEO satellite, the position prediction errors of the SGP4 algorithm when the TLEs are one day old are on the order of 10^2 m, with maximum errors between 3 to 6 km, depending on the intensity of the Solar flux.

2.7 Frames of reference

Any position needs to be expressed in a frame of reference. For terrestrial calculations, it is advantageous to use an Earth-centred frame, which has its origin at some centre (e.g. gravitational, physical) of the Earth. There are two general types of Earth-centred frames: Earth-Centred Inertial (ECI), which are fixed relative to some position external to the Earth (usually the Earth's position at equinox) and thus do not rotate with the Earth (they are inertial), and Earth-Centred Earth-Fixed (ECEF), which are fixed to a point on Earth and thus do rotate with it (they are non-inertial).

For this work, three frames of reference are important: TEME (ECI), ITRS (ECEF) and WGS84 geodetic datum (ECEF).

The process of transforming between those systems is described e.g. in [21], and the transformations used in this work are all based on the equations and algorithms provided there: from TEME to ITRS [21, p. 231-233], from ITRS to geodetic [21, p. 174-179] and back [21, p. 146].

2.7.1 True Equator Mean Equinox

The True Equator Mean Equinox (TEME) reference frame is a Cartesian ECI frame. The z-axis points along the Celestial Ephemeris Pole[1] (CEP), while x-axis is located on the True Equator plane (a plane perpendicular to the CEP) and points in the direction of the Vernal Equinox without accounting for celestial nutation (the *mean* or *uniform* equinox)[21, p. 231-233]. The TEME frame is used by the SGP4 algorithm (see section 2.6).

2.7.2 International Terrestrial Reference System

The International Terrestrial Reference System (ITRS) is a standard Cartesian ECEF frame. The origin is at the centre of mass of the Earth and the axes are fixed to defining coordinates on Earth. As a result of tectonic movement, the system is periodically recalculated[21, p. 152]. All calculations within this work are done in ITRS.

2.7.3 World Geodetic System Geodetic Datum

The World Geodetic System-84 (WGS84), where 84 refers to the most recent version from 1984, is an ECEF frame maintained by the U.S. National Geospatial-Intelligence Agency. The WGS84 and ITRS agree on a level of

[1] The Earth rotation axis

centimetres[21, p. 203]. The geodetic datum (hereafter referred to as only the *geodetic frame*) is a latitude-longitude-altitude (ϕ, λ, h respectively) frame defined within WGS84. In this work, the estimated position is in the geodetic frame for clarity, however, for navigation calculations themselves the position is always converted to ITRS.

Chapter 3

Overview of LEO satellite systems suitable for SoP positioning

This chapter describes current LEO satellite constellations which are readily usable for SoP navigation, namely Iridium NEXT, Orbcommm, Globalstar and Starlink. For each system, the orbital parameters, constellation status, frequency ranges and communication schemes are described, in addition to select signal types deemed of particular importance for this work.

Parameters of the systems are summarised in Table 3.1.

3.1 Iridium NEXT

Iridium satellite constellation, owned and operated by Iridium Communications Inc., provides global communications coverage for satellite phones, pagers and other devices, as well as an interface to terrestrial communications networks via several Gateways.

The system became operational in 1998 and saw a total of 96 satellites launched, before the constellation was completely overhauled between 2017 and 2019, replacing all of the original satellites with Iridium NEXT satellites. Currently[1], no original satellite remains operational, however, 25 satellites remain defunct on orbit[23]. In this work, unless specifically noted, Iridium refers to Iridium NEXT.

The constellation consists of 66 satellites[2], 11 each in six orbital planes, spaced 30° apart. The altitude of the satellites is approx. 781 km, the inclination is 86.4° and the orbital period is approx. 100 min. For a ground user, one satellite is visible for about 7 min. The constellation covers the entire global surface[24].

Communication is carried over Ka-band for satellite-satellite links and satellite-gateway links, and over L-band (1616 to 1626.5 MHz) for satellite-user links. Each satellite provides 48 individual spot beams, sharing 240 traffic channels with a frequency re-use pattern[25]. At ground level with an Iridium antenna, the signal strength can be expected to be −90 to −110 dBm[26].

[1] As of October 2023

[2] Iridium 102-114, 116-123, 125-160, 163-168, 171-173, 180. All ranges are inclusive.

	Iridium	Orbcomm	Globalstar	Starlink
Satellites	66	30	48	>5000
Orb. planes	6 × 86.4°	4 × 47°	8 × 52°	>180
Altitude	781 km	715 km	1414 km	550 km
Visibility	7 min	7 min	14 min	4 min
Frequency range	1616 MHz - 1626.5 MHz	137 MHz - 138 MHz	2483.5 MHz - 2500 MHz	10.7 GHz - 12.7 GHz
Strength	−100 dBm	−118 dBm	not found	not found
Modulation	DE-QPSK	SD-QPSK	QPSK	16QAM

Table 3.1: Summary of *approximate* parameters of LEO satellite systems: number of active satellites, number of orbital planes and inclinations, altitudes, visibility for one satellite, frequency ranges, signal strengths at ground level and modulation used

The Iridium NEXT onboard clocks have a stability better than 10×10^{-9} and a drift less than $3\,\mathrm{Hz\,s^{-1}}$, which is sufficient for time transfers with accuracy in the order of 1 µs[27].

3.1.1 Iridium signals

Due to the scope of this work, only the satellite-user (L-band) signals are examined, as the Ka-band signals are neither globally transmitted nor aimed towards the Earth.

Iridium uses a combination of Space Division Multiple Access (SDMA), Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA) and Time Division Duplex (TDD) multiple access schemes[27]. For every spot beam, channels[3] are implemented using TDMA architecture based on TDD using a time frame[3]. The structure of a TDMA frame is illustrated in Figure 3.1.

In the 1626 to 1626.5 MHz band used by Iridium, there are 252 carriers with carrier spacing of 41.667 kHz grouped into 31 sub-bands of 8 and one of 4. There are 5 simplex carriers in the band with a spacing of 35 kHz, one of which is the Ring Alert, and the remaining four are messaging carriers for paging and acquisition. The polarisation of both the uplink and downlink band is right-hand circular. Iridium uses DE-QPSK modulation[28].

Paging

Paging signals enable users to receive ringing and paging messages during heavier atmospheric fading conditions and in buildings where attenuation is greater. There are five paging channels, one for alert (Ring Alert, detailed below) and five for transmitting paging messages to the receive-only terminals. The transmission duration does not exceed 20.32 ms[29].

[3] A channel is a specific FDMA frequency and TDMA timeslot[25]

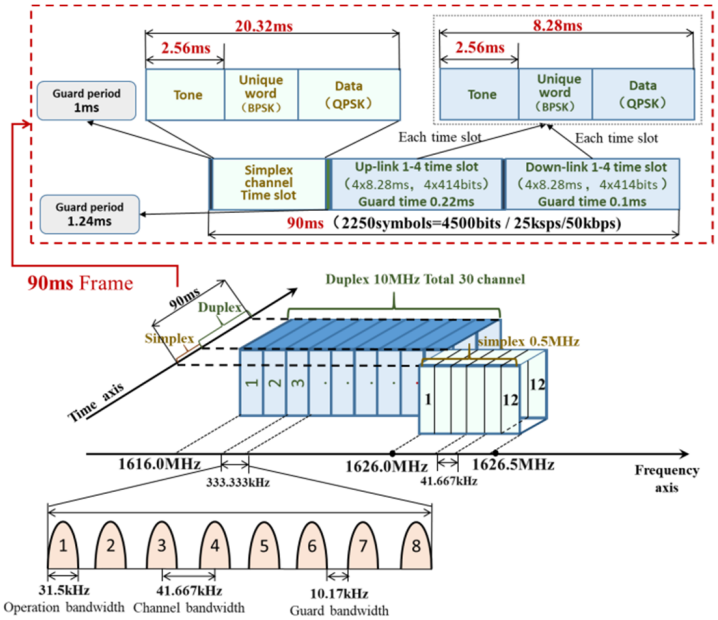


Figure 3.1: Iridium frequency allocation and frame structure[3]

Ring Alert

The Iridium Ring Alert signal is an unencrypted downlink-only simplex channel with a carrier frequency of 1626.2708 MHz[28]. Each of the 48 satellite beams transmits a Ring Alert message every 4.32 s, for a transmission period of 90 ms per one satellite[25], which can be used to keep track of a specific satellite[27].

The transmit power for the Ring Alert channel is higher than that of voice/data channel, so as to enable the mobile earth terminals to receive ring alerts even when their antennas are stowed[29].

The Ring Alert Message contains the following information[9]:

- Satellite ID - a numeric identifier (2 to 115) of the transmitting satellite, which notably does *not* correlate with the satellite NORAD identifier,
- Beam ID - a numeric identifier (0 to 47) of the transmitting beam,
- Latitude - the current *ground* latitude of the satellite in degrees, with two decimal places of precision, calculated by the satellite,
- Longitude - same format as Latitude and
- Satellite altitude - altitude above the surface in km.

Importantly, the inclusion of satellite ID in the Ring Alert message, combined with the fixed and known frequency of the Ring Alert channel and the absence of encryption, enables any receiver to quickly determine which satellite is transmitting. The Ring Alert signals are thus very useful for SoP navigation using Iridium.

3.2 Orbcomm

Orbcomm satellite constellation provides private internet access and machine-to-machine communication services primarily to industrial customers, such as transportation companies or manufacturers. The first launch of the system occurred in 1991. There were several generations of satellites, of which two amounted to significant numbers - the first production generation, referred to as Orbcomm Generation 1 (OG1), and the latest one, Orbcomm Generation 2 (OG2), which began launching in 2012. TLEs indicate that 60 satellites are currently[4] in orbit, of which 36 are active[5], 24 of OG1 and 12 of the OG2 configuration[30].

The Orbcomm satellites orbit in four orbital planes inclined at 47° at an altitude of 715 km. The orbital period is approx. 99 min and a satellite is visible to a ground observer for about 7 min[31, 11].

The Orbcomm communication system consist of the Space segment (satellites), Ground segment (Earth-based gateways and communication centres) and the Subscriber segment (Earth-based paying customers)[11]. Orbcomm satellites are also equipped to receive Automatic Identification Signals from maritime traffic, however this is not a navigation system, but rather a navigation information collection system[31].

3.2.1 Orbcomm signals

Downlink signals from the Orbcomm satellites to the subscriber and ground segments are in the VHF band, within 137 to 138 MHz. There are 12 channels for communication with the subscriber segment and one for the gateways[11].

Each satellite is assigned one of the 12 channels. Over the constellation, the frequencies are reused in a scheme such that signals of equal frequency never overlap. Frequencies of the channels are listed in Table 3.2. The bandwidth of all channels is 25 kHz, except the gateway channels, whose is doubled[6]. Importantly, this assignment provides a way of satellite identification, if a table of assigned frequencies is provided or constructed experimentally.

The downlink signals are modulated using Symmetric Differential Quadrature Phase Shift Keying (SD-QPSK). The transmission power varies from 10 to 40 W with effective isotropic radiated power (EIRP) of about 12 dBW. Ground signal strength is about −118 dBm. The polarisation is right-hand circular[11, 31].

In addition to the downlink communication signals, Orbcomm satellites are equipped with a 1 W unmodulated 400.1 MHz (UHF) beacon[6]. However, in [11] it was shown that this beacon was missing or turned off in the tracked satellites.

[4] As of October 2023

[5] Orbcomm 4-15, 18-21, 23, 27, 30, 31, 32 (partially), 34, 35 (OG1); 103, 107-110, 112-118 (OG2). All ranges are inclusive.

Channel	Frequency (MHz)
S-1	137.2000
S-2	137.2250
S-3	137.2500
S-4	137.4400
S-5	137.4600
S-6	137.6625
S-7	137.6875
S-8	137.7125
S-9	137.7375
S-10	137.8000
S-11	137.2875
S-12	137.3125
Gateway	137.5600

Table 3.2: Orbcomm downlink frequency channels[6]

3.3 Globalstar

Globalstar is a satellite communication system designed to provide voice and low-rate data communication to user terminals on the Earth. The satellites connect the users and ground-based gateways by retransmitting the user terminal signals[32, 33].

The constellation had two distinct generations. The first generation has been launched from 1998 and included 48 LEO satellites, none of which remain operational. The current[6], second, generation has 24 operational satellites[7], launched between 2010 and 2013. Neither the orbital configuration nor the signal spectrum has changed between the generations[34].

The satellites orbit at an altitude of 1400 km in 8 planes inclined at 52°. The orbital period is 114 min. Each Globalstar satellite is visible for around 14 min[33, 34].

The number of simultaneous visible satellites depends on latitude. For the original constellation and a latitude of 50°, a probability of seeing exactly one, two or three satellites simultaneously in a given moment is respectively 19 %, 71 % and 10 %[32]. For the second generation, the probabilities can be expected to be much lower.

The Globalstar satellites use a phased array antennas with 91 high power amplifiers, each providing an output power of 4 W (for a total of 364 W)[33]. Satellite fractional frequency error standard deviation is specified as 10^{-9}[32].

[6] As of October 2023
[7] Globalstar 074-097, inclusive.

3.4 Globalstar signals

The Globalstar constellation implements four types of signals - satellite-user downlink (S band), user-satellite uplink (L band), and satellite-gateway downlink and uplink (C band, 5 to 7 GHz). All transmissions use Code Division Multiple Access (CDMA)[35].

The user downlinks are in the frequency range from 2483.5 to 2500 MHz. Each satellite transmits in 16 beams, utilising the full allocated range of 16.5 MHz. The modulation is QPSK[33].

3.5 Starlink

Of the systems discussed, Starlink has by far the most satellites. However, due to the downlink frequency exceeding 10 GHz, it is not feasible to process Starlink signals in this work. Starlink is therefore discussed only briefly for the sake of completeness.

Starlink is an satellite internet and telephony constellation. It currently[8] features over 5000 satellites in over 180 orbital planes, inclined mostly at 52° or 97°. A constellation of up to 24000 - 42000 satellites is possible[15]. The satellites orbit at an altitude of around 550 km, with an orbital period of 91 min and single satellite visibility time of around 4 min.

The starlink downlink signal is in the 10.7 to 12.7 GHz band, with several channels of a 240 MHz bandwidth. Furthermore, several 1 MHz bandwidth tones were identified within the band by experimentation. The structure of the signal is not public. The reception of signals within this band usually requires a low-noise block down-converter and a parabolic antenna reflector[17].

3.6 Summary

Iridium NEXT, Orbcommm, Globalstar and Starlink were discussed in this chapter, but only the frequency ranges of Iridium NEXT, Orbcommm, and Globalstar fit within the bounds set by available antennas. Thus, only those three will be considered in this work.

Iridium NEXT constellation has the most satellites, while Globalstar has the most simultaneously visible satellites. At least one Iridium NEXT and Globalstar satellite is always visible, while for Orbcomm this is not the case. As many as 6, 6, and 13 Iridium, Orbcomm and Globalstar satellites respectively are visible simultaneously. On average, 2.2, 1.6, and 5.8 respectively are visible simultaneously. These values are summarised in Table 3.3 and visualised in Figure 3.2. Elevation angles for the visible satellites of one constellation (Iridium NEXT) are in Figure 3.3.

The visibility was simulated relative to the location of CTU FEE (50°6′11″N, 14°23′32″E, 225 m) over 36 hours starting on May 1st 2024 00:00 UTC. The

[8] As of October 2023

	Iridium	Orbcomm	Globalstar
Min	1	0	1
Avg	2.2	1.6	5.8
Max	6	6	13

Table 3.3: Number of simultaneously visible satellites for Iridium, Orbcomm and Globalstar over 36 hours

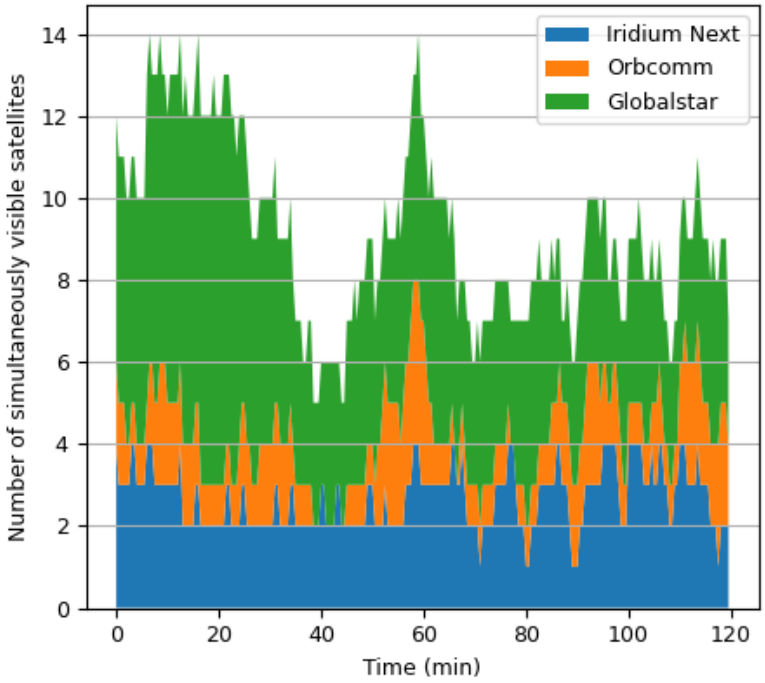


Figure 3.2: Number of simultaneously visible Iridium, Orbcomm and Globalstar satellites during 2 hours of a day

plots are 2-hour excerpts from the middle of the time window.

The signal power at the receiver location for Iridium and Orbcomm is −100 dBm and −118 dBm respectively, compared to the power of GPS signal of −128 dBm[10, p. 11] (difference of 630x, 10x respectively), which clearly illustrates the advantage of signal strength the SoP positioning systems enjoy over GNSS.

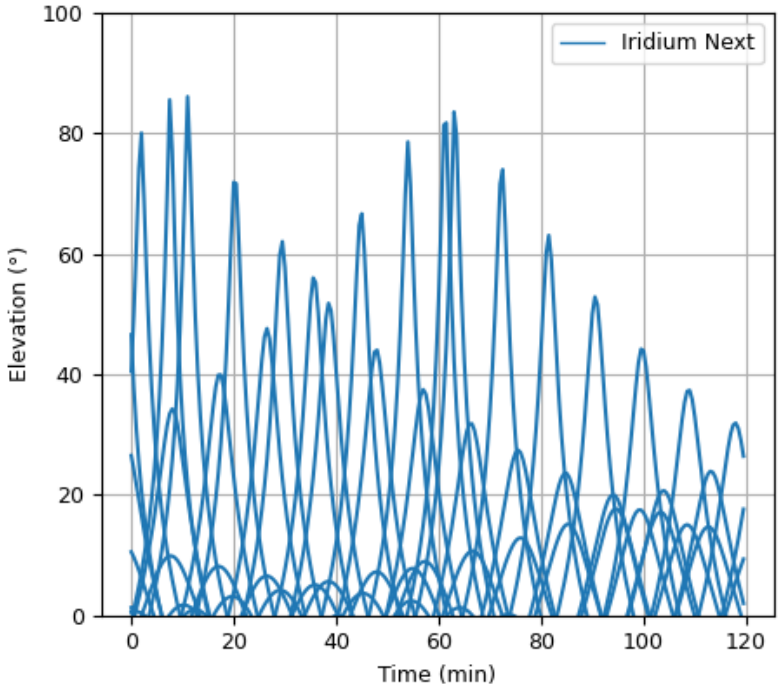


Figure 3.3: Elevation angles of Iridium satellites during during 2 hours of a day

Chapter 4

Survey of current SoP positioning systems

This chapter contains a survey of current research on the topic of SoP positioning. Firstly, two positioning systems based on the Doppler method are presented for comparison. Then, a survey of state-of-the art experimental SoP positioning systems is conducted, and for each research effort, the methods and results are summarised. Finally, some general conclusions are drawn and the results achieved by the surveyed systems are compared to dedicated positioning systems - the Transit and GNSS systems.

4.1 Doppler method positioning system

This section briefly discusses two dedicated positioning systems, the Transit and Argos, which use the Doppler method. It is meant to provide a basis of comparison when evaluating the SoP systems which is more appropriate than comparing against GNSS.

4.1.1 Transit

The Navy Navigation Satellite System, commonly referred to as Transit or NAVSAT, was the first operational satellite navigation system. The system was used to provide accurate position updates for the INS equipment aboard Polaris nuclear ballistic missile submarines. Later, civilian use cases such as fishing boat navigation emerged. The system was operational from 1964 to 1996. The system used the Doppler method for positioning, the details of which are described in subsection 2.3.1. The transmission frequencies were 150 MHz and 400 MHz (two to allow for compensation of ionospheric refraction) and the system also provided a data message used for clock synchronisation. The performance of the system exceeded that of all of the early large radio systems - the RMSE of a position fix of a static naval user was 40 to 100 m and the time to fix was 10 to 16 min, but could be as low as 2 min with lower accuracy[14].

4.1.2 Argos

Argos is a global satellite-based location and data collection system dedicated to studying and protecting Earth environment, operated through a multi-national research agency collaboration. It allows any mobile object, usually a tracked animal, equipped with a compatible transmitter terminal to be located across the world[36]. Unlike all other systems discussed, the actual position determination is done at the level of the satellite, which receives a signal (which may also contain sensor data) from the terminal and calculates its position.

Argos terminal messages are transmitted at a frequency of (401 650±30) kHz. Calculation of terminal position is done through the Doppler shift method, using a classical nonlinear least squares estimation or a multiple-model Kalman filter. The positioning accuracy typically ranges from the order of 10^1 km to less than 250 m, depending on the number of terminal messages received during a satellite pass[1]. Algorithms based on animal movement models are used to resolve the symmetrical problem of Doppler positioning[36].

4.2 Experimental SoP positioning systems

This section contains a survey of select experimental SoP positioning systems, each with a brief description of the implementation and experimental results. All of these system use the Doppler-shift method.

In [32], it is demonstrated using analysis and simulation that it is possible to obtain quick[1] positioning using Doppler frequency measurements of signals from one or two satellites of the Globalstar constellation (forward down-link carrier frequency of 2500 MHz). The error in the horizontal plane of such positioning is expected to be less than 9 km in 90 % of cases using one satellite. Using two satellites, the error is expected to be 1.4 km in 90 % of cases. This is without any advanced signal processing or data fusion.

In [26], a method of positioning is devised for use as a secondary aircraft navigation system in GPS-denied areas using Iridium NEXT satellites and an SDR. Circular Error Probable (CEP) of 0.2 to 2 km in dynamic mode is demonstrated without using significant computational resources. The system is based on receiving STL, Ring Alert and MSG Iridium NEXT signals on 1626.104 MHz, 1626.27 MHz and 1626.44 MHz, respectively, and then fusing the Doppler shift data with IMU data using Kalman filtering.

In [17], a Starlink-based system is designed, based on Doppler measurements of a tone band centred on 11 325 MHz. The system tracked the tone signal of up to six satellites by FLL-assisted-PLL. Extended Kalman filtering was used. The system accuracy was 375 m. The team determined that the most significant error sources were the receiver (user) clock drift, the user time error and the satellite position error. Notably, the satellite clock drift was not determined to be significant.

[1] Definition of "quick" is not provided, but it is stated that it is significantly less than 16 s

In [18], a system based on the fusion (using EKF) of Iridium Doppler shift measurements, INS and magnetometer data is proposed and simulated for a highly dynamic application. The simulated Iridium signal was centred on 1626.25 MHz. The arrival position error[2] in 95 % of cases was approximately 300 m for the best case, and 4 km for the intermediate case. The main difference between the two cases was the accuracy of orbital data.

In [11], a framework is presented which estimates the LEO satellite states along with the states of a navigating vehicle using Orbcomm satellite signals. The system fuses INS with Doppler measurements of a signal in VHF downlink band of 137 to 138 MHz using EKF. The system was tested using an UAV, which had GPS signal available for 90 s and then turned off for 30 s. The final position error was 8.8 m, compared with INS-only error of 31.7 m. Static positioning without the use of GNSS was not performed.

In [37], a framework employing EKF to estimate user position from Doppler measurement of multiple generic LEO satellites was proposed. Simulation showed a RMSE in position from approx. 170 m for 5 satellites over one minute to 11.5 m for 25 satellites over four minutes. Experimental run with 2 Orbcomm satellites (broadcasting at 137.3125 MHz and 137.25 MHz respectively) over one minute showed a RMSE of 360 m.

In [27], and Iridium-based system is proposed and tested. The system uses the Ring Alert and Primer message signals to capture Doppler shift data, which are then processed using Kalman filtering and the least squares method. The system accuracy can be as much as 22 m after 24 h of measurement under an open sky. For 30 min under open sky, the accuracy was shown to be 46 m. Furthermore, the system accuracy in GPS-hostile conditions (in a dense forest) was shown to be[3] 108 m, whereas GPS did not provide a solution.

In [16], two LEO constellations (Iridium NEXT and Orbcomm) are used to determine position. Three-dimensional RMSE achieved over 30 s using a single constellation (Orbcomm) is approx. 0.76 km, whereas using both constellations the RMSE is 0.22 to 0.18 km depending on the received signal type. Accuracy when measuring messaging bursts was better than for Ring Alert bursts. EKF was utilised in the signal processing. In the receiver architecture, each constellation has a dedicated acquisition chain, including separate antennas, amplifiers, filters and signal processing.

4.2.1 Summary

The research outlined above is summarised in Table 4.1. Most research focused on using Iridium NEXT satellites as signal source, followed by Orbcomm. The accuracy of position determination was mostly in the order of 10^2 to 10^3 m and improved significantly both with measurement time and with the number of measured satellites. In one case the performance was shown to exceed the one of existing GNSS systems. Furthermore, the surveyed research

[2] That is, the error in position after travelling along a predefined trajectory in the simulation

[3] The experiment time is not stated, but can be inferred to be about 30 min

Constellation	Signal	Mode	Accuracy (2D)	Exp. time	Source
Globalstar	2500 MHz	static	1.4 to 9 km*	few s	[32]
Iridium NEXT	STL, Ring Alert, MSG	dynamic	0.2 to 2 km	not found	[26]
Starlink	11 325 MHz	static	375 m	330 s	[17]
Generic, Orbcomm	137 MHz	static	11.5* to 360 m	1 to 4 min	[37]
Iridium NEXT	Ring Alert, Primer	static	46 to 108 m	30 min	[27]
Iridium NEXT, Orbcomm	MSG, Ring Alert	static	0.18 to 0.76 km	30 s	[16]

Note: Results from [18] and [11] are not in this table, as the research concerns SoP positioning coupled with INS in a dynamic application

Table 4.1: Summary of selected existing SoP positioning systems (* denotes a simulated result)

demonstrated the possibility of using SoP positioning both in static and dynamic applications. Notably, all of the surveyed experiments used Kalman filtering for fusion of multiple navigation system data or error estimation, albeit the actual implementation varied greatly.

4.3 Comparison to dedicated positioning systems

To compare the SoP positioning systems, a representative system needs to be created first. The considered parameters are the system positioning accuracy and the measurement time required to achieve a position fix (TTF, time to fix).

There are too few systems to use standard statistical methods, and the parameters of the systems are too spread apart to be represented by just one system. Furthermore, the systems with better accuracy generally take much longer to acquire a fix. Therefore, two systems were considered for comparison - "SoP A", which represents a more accurate albeit slower system, and "SoP B", which represents the opposite. The SoP A has an accuracy of 200 m and time to fix of 600 s, whereas the SoP B parameters are 700 m and 120 s, respectively.

The parameters of the different GNSS systems are quite similar at the level of SoP systems[38], therefore GPS will be used as a representative example. No navigation aiding, such as SBAS, will be considered, as this approximately matches the operation of SoP systems. However, even within GPS the system parameters differ greatly. Geodetic GPS receivers are accurate to a few centimetres, but require up to 30 min to obtain a fix with such accuracy[38]. On the other hand, commercial-grade receivers are accurate on the level of metres and require no more than few minutes to fix from a cold start[4][10, p.

[4]GPS cold start matches the operation of SoP systems

System	Accuracy (m)	× best	Time to Fix (s)	× best
GPS	**3**	1	**120**	1
Transit	70	23	780	6.5
SoP A	200	67	600	5
SoP B	700	233	**120**	1

Table 4.2: Comparison of SoP and dedicated positioning systems

33]. To represent GPS, a system with an accuracy of 3 m and time to fix of 120 s will be considered.

Additionally, the Transit system (70 m, 13 min) will be considered (see section 4.1). The Argos system will not be considered, as its parameters vary greatly and its operation is different.

The systems are summarised in Table 4.2. Clearly, GPS performs best in terms of accuracy. Interestingly though, the time of fix of the faster SoP system (SoP B) is similar to that of cold start GPS. Compared to Transit, the SoP systems are faster, and the difference in accuracy is much less, albeit still not in favour of the SoP systems.

The findings of this comparison are hardly surprising, as even the Doppler method requires time synchronisation, which is not provided by SoP signals, and neither are precise ephemerides to increase the accuracy of satellite orbit determination or any other navigation aids. Furthermore, the SoP systems are based on systems which were never intended for precision navigation, such as the TLEs, or indeed the communication satellites themselves - the clock accuracy and therefore the transmission frequency is not assured on a navigation level.

On the other hand, some research showed that the SoP system is capable of acquiring a fix in conditions when GPS is not; and the demonstrated parameters are sufficient for some applications, e.g. emergency location (see the discussion in section 1.3 for more details).

Chapter 5

Design of the SoP navigation system

The principal goal of this work is the experimental demonstration of SoP positioning using the satellite signals. This chapter describes the design and implementation of the navigation system, which performs SoP positioning using Iridium NEXT signals.

Iridium NEXT was chosen for the demonstration because of all the considered systems within the capability of the available radio equipment (i.e. Iridium NEXT, Orbcomm, Globalstar), Iridium NEXT is the best documented[1], the format of its frames is well known, it has the most satellites and the most current constellation, its signals fall within the L1 band, which results in readily available antennas and its channel structure is known and includes simplex broadcast channels of fixed frequency with unencrypted data.

In order to calculate the user position, the navigation system has to cover four steps, each of which is discussed in detail below:

1. Signals from satellites are captured,
2. data frames are extracted from the signal and decoded,
3. decoded frames are processed - transmitting satellite is identified, the satellite position at the transmission time is calculated and frequency curves for individual satellites are found,
4. initial location is estimated and the user position is calculated.

A note on nomenclature: Since this chapter discusses the design of the navigation system in detail, an explanation of the used nomenclature should be given. Within the context of this chapter and the navigation system, the *navigation system* refers to the application developed in this work and a *frame* is a data unit sent over a channel by a communication satellite, on the reception of which the navigation system is based. *Extracting* frames means acquiring information about a transmitted frame that do not go

[1]This is a subjective claim gained from research and interactions with the amateur radio community. However, the number of results for the system name on Google Scholar illustrates the situation: 354 000, 3800 and 15 600 respectively for "Iridium NEXT", "Orbcomm" and "Globalstar" as of April 2024

beyond its RF properties, such as the centre frequency, signal strength or time of arrival. *Demodulating* means converting captured samples into data bits, while *decoding* means converting the bits into meaningful (or human-readable) data. A frame has some properties, namely *reception time* t, a *centre frequency* f which is shifted from a *base frequency* of the channel f_B on which the frame was transmitted by Doppler shift f_D ($f_D = f - f_B$). Any value with the subscript U refers to the user (i.e. the position of the antenna of the navigation system), a subscript S or prefix `sat-` refers to the transmitting satellite. Unless otherwise noted, all positions denoted by x, y, z are in ECEF frame (namely the ITRS) and all positions denoted by ϕ, λ, h are in the geodetic frame (ϕ, λ, h = latitude (rad), longitude (rad), height (m)). In algorithms or conditions, subscript i refers to the i-th received frame. References to *Iridium* always mean *Iridium NEXT*.

5.1 Capturing signals

The first step in determining the user position is to capture signals transmitted by the satellites used for navigation. To achieve that, an antenna and a radio are needed.

5.1.1 Antenna setup

As discussed in subsection 3.1.1, Iridium NEXT transmits in the 1626 to 1626.5 MHz frequency range, which lies within the L-band. The signal strength on Earth is expected to be in the −90 to −110 dBm range[26], which is in the order of pW in absolute signal power.

Thanks to the relative proximity (in frequency terms) of the Iridium signals to those of GPS, a GPS antenna can be used. However, some active antennas, such as the NovAtel GPS-702[39], are severely attenuated in the Iridium signal band, and thus cannot be used.

The antenna used in this work is NovAtel GPS-704-X passive GNSS antenna. The 3 dB pass band of this antenna is 1.15 to 1.65 GHz with minimum gain at zenith of 6.0 dBic in the L1 band. Additionally, the antenna features NovAtel's patented Pinwheel technology for multi-path rejection and phase centre stability[40]. The antenna was chosen due to its suitable characteristics as well as its availability.

To acquire Iridium signals of sufficient strength, a low noise amplifier was found to be necessary. The amplifier used in this work is the Mini-Circuits ZX60-3018G-S coaxial amplifier with frequency range of 0.2 to 3 GHz, typical maximum output power of 12.8 dbm and gain of 20.60 dB for 1671 MHz[2][20].

5.1.2 Radio

The radio used in the navigation system is the ADALM-PLUTO (in short PlutoSDR) software defined radio development kit. It is a portable self-

[2]Closest value in the datasheet to the received frequency range.

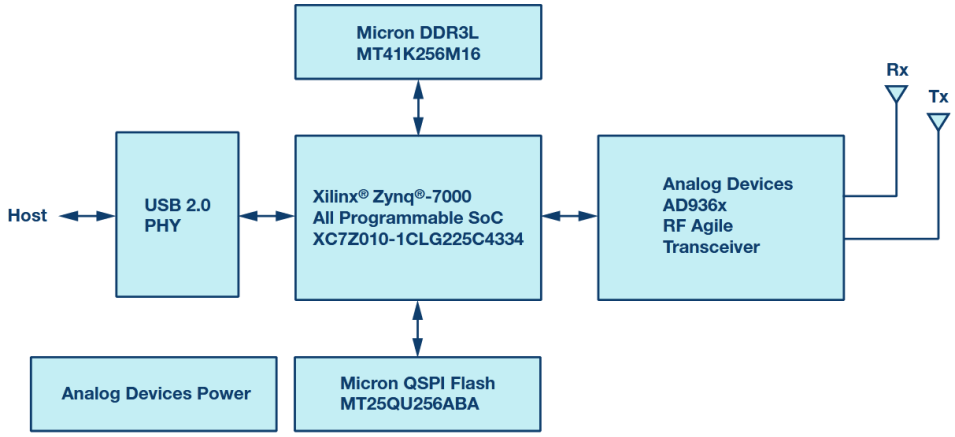


Figure 5.1: PlutoSDR block diagram[4]

contained RF-learning module, equipped the AD9363 RF Agile Transceiver. PlutoSDR has one TX and one RX channel with a RF range of 0.325 to 3.8 GHz. The maximum bandwidth is 20 MHz. The SDR is powered and communicates through USB2.0 with drivers available in Windows and application programming interface (API) in, among others, C, C++ and Python. The APIs are implemented in several development platforms, notably Matlab and GNURadio (via SoapySDR). The reference clock of the SDR has a nominal frequency of 40 MHz and an accuracy of $\pm 25 \times 10^{-6}$[4]. The block diagram of the SDR is in Figure 5.1.

Frequency offset and drift

The synthesizer blocks that generate all data clocks, sample clocks, and local oscillators inside the PlutoSDR transceiver are supplied by a reference clock, which is external to the transceiver, but it is integrated in PlutoSDR and connected to pin `XTALN` of the transceiver (see Figure 5.2)[5]. Due to frequency conversion, any error in the frequency of the reference clock will be multiplied in actual measurements. Given that the reference clock nominal frequency of 40 MHz is approximately 40 times lower than the measured band ($\approx$1600 MHz), 1 Hz offset in reference clock frequency will appear as a -40 Hz shift in measured frequency.

Measurements of satellite signals showed that the reference clock of the PlutoSDR suffers from significant frequency offset, which varies with time. As a result, the measured signal frequency differs from actual received frequency by up to 20 kHz. A significant cause of the change in the reference clock frequency are variations in temperature, both as a result of internal heating of the SDR components and outer environment changes, which cannot be avoided when measuring outdoors or over a long period of time.

An experiment was carried out, in which the PlutoSDR reference clock frequency was compared to a precise frequency generator (an atomic clock at the FEE CTU). After SDR startup, the reference clock frequency was

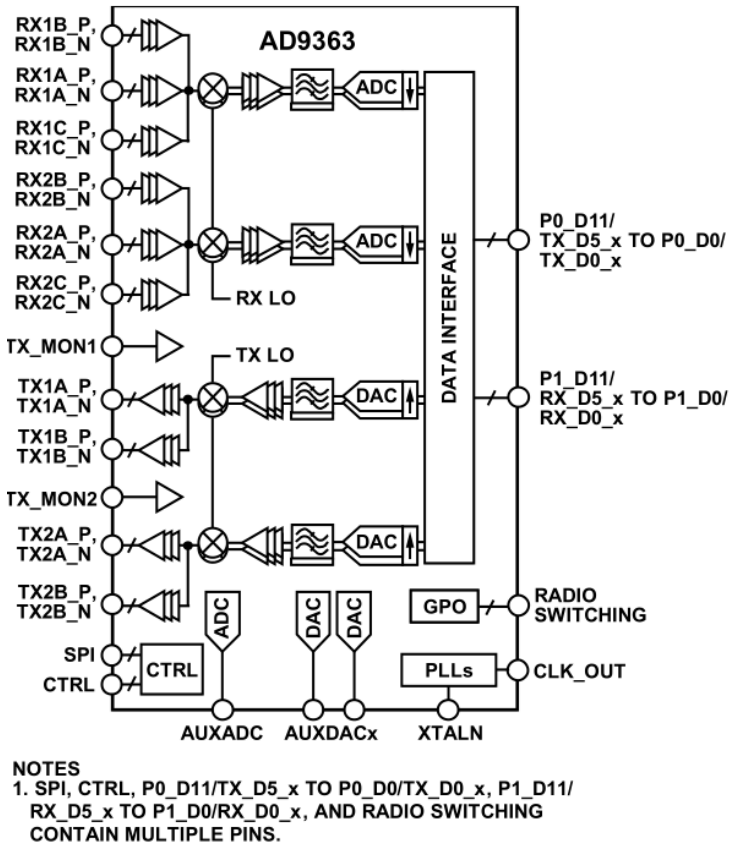


Figure 5.2: AD9363 functional diagram[5]

39 999 729 Hz (nominal frequency f_N is 40 000 000 Hz - a difference of −271 Hz or $-6.8 \times 10^{-6} \times f_N$). After 20 minutes of operation, the reference clock frequency was 39 999 571 Hz (−429 Hz or $-10.7 \times 10^{-6} \times f_N$). After this time, the frequency varied by ±50 Hz, but mostly stabilised. Any change in external environment, such as opening the laboratory window, caused significant variation in reference clock frequency.

As a result of this experiment, the PlutoSDR was covered in polystyrene case when measurements were taken, to somewhat stabilise its temperature. Additionally, the SDR was calibrated to account for some of the discrepancy in reference clock frequency, by setting the device to assume the reference clock frequency is 39 999 571 Hz.

This experiment, as well as the satellite signal measurements shows the navigation system needs to account for frequency offset (henceforth referred to as just *offset*) and the change of this offset (henceforth referred to as *drift*) in its calculations. Without precise external frequency source, such as an atomic clock (which is unfeasible) or a GPS-synchronised oscillator (which introduces a reliance on GNSS), precise measurements of absolute frequency are not possible. Using a better SDR with more precise or better temperature-compensated oscillator might help, but any system needs to handle unreliable frequency measurements by itself.

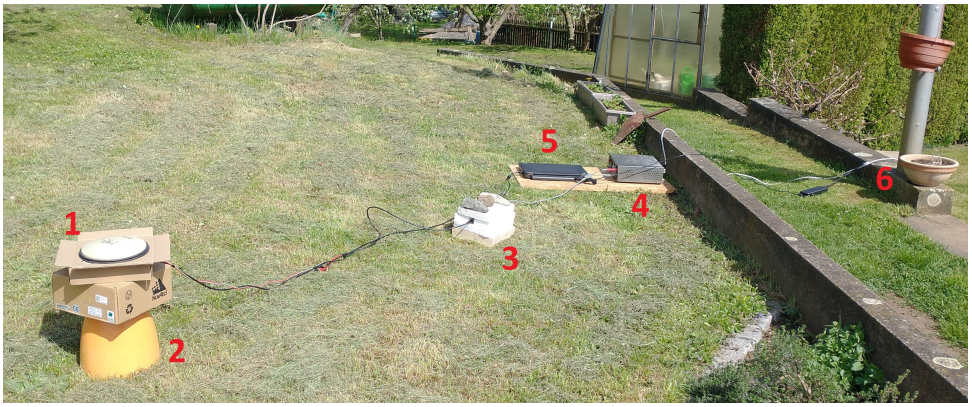


Figure 5.3: Measurement setup

5.1.3 Measurement setup

For successful signal capture, the antenna needs to have a wide and mostly unobstructed field of view. Furthermore, the radio should be appropriately protected from rapid changes in environment temperature (such as wind), and adequate power supply must be provided to the radio.

An example of the measurement setup used in this work is photographed in Figure 5.3. It consist of the GPS-704-X antenna with the amplifier (1), an antenna stand (2), the PlutoSDR in polystyrene case (3), a 12 VDC laboratory power supply (4) powering the amplifier, a Lenovo T440p laptop (5) running the measurement software and a power supply (6). The measurement setup changed very little during the course of the work.

The view of the antenna is unobstructed above approximately 15° of elevation, with the exception of north direction, which is slightly obstructed by a tree. However, as is apparent in the data, the field of view is overall satisfactory (see e.g. Figure 5.5).

5.2 Frame extraction and decoding

One of the methods of acquiring and tracking a satellite signal has been described in section 2.5. However, that refers only to acquiring the signal itself and measuring its frequency. It is highly advantageous to also demodulate and decode the data within the signal. Firstly, some frame types are associated with a specific frequency or a range of frequencies, which greatly simplifies determining the frequency on which the frames were transmitted, and thus their Doppler shift. Secondly, some frame types contain satellite identifiers, which are useful for determining which satellite is transmitting. Thirdly, some frame types contain the position of the transmitting satellite, which may be used as an initial location for the navigation algorithm.

Extracting, demodulating and decoding frames from satellite signal is a specialised task, which is beyond the scope of this work. Therefore, two already existing programs were used - `gr-iridium`[41] and `iridium-toolkit`[42].

Due to the lack of public documentation, the programs were developed by reverse-engineering Iridium phone and pager firmware and by trial and error over two years[43].

The `gr-iridium` package is a module for GNURadio, which handles capturing and demodulating frames sent by Iridium satellites. It works with generic SDRs and outputs decoded Iridium frames as text output in console. Provided it is supplied with proper configuration, the package works from command line and is Windows-compatible, with compiled binaries available in Conda package manager. It requires GNURadio 3.10. The principle of operation is described below.

The `iridium-toolkit` is a Python application capable of decoding demodulated Iridium frames captured by `gr-iridium`. It can work from command line, is Windows-compatible and does not require compilation nor it relies on GNURadio. It can decode the data in several types of Iridium frames by matching against a known message format.

Both of these packages have already been used in research, such as in [9] to capture Iridium Ring Alert messages over a long time.

Process used by `gr-iridium` to capture and decode Iridium frames

Below is a description of the process the `gr-iridium` uses to capture and demodulate Iridium frames. Since the documentation of the package is sparse, the information below was inferred from code. References to specific sections of code are in footnotes where appropriate[3].

The core of the package is a GNURadio flowgraph (see Figure 5.4[4]). Within this GNURadio environment, IQ samples are passed along the solid arrows into the blue nodes of the function blocks and Protocol Data Unit (PDU) messages are passed along the dotted arrows into the grey nodes. The process of extracting and demodulating frames is sequential - it starts in the `Soapy Custom Source` and ends in the `iridium_frame_printer`.

The **`Soapy Custom Source`** block handles communication with the PlutoSDR. It sends out raw IQ samples for further processing.

The **`iridium_burst_tagger`** block identifies Iridium bursts in the captured data and "tags" them by creating metadata with references to burst start time (in terms of samples), magnitude, internal ID, centre frequency, rough carrier frequency offset (CFO) etc. First, the block squares the received signals twice[5]. Then it performs FFT on the received samples with as fixed length window of the "Blackman-Harris" type (scaling factor 0.42, equivalent noise bandwidth 1.72[6])[44]. After that it removes peaks around burst by the use of a burst mask, extracts the remaining peaks, updates the mask to match the new burst and passes the samples along with the tag to block output.

[3]The source code is available at `https://github.com/muccc/gr-iridium/blob/v1.0.0/`

[4]The flowgraph was created by the author, it is not actually present in graphical form in the package

[5]gr-iridium/lib/iridium_burst_tagger_impl.cc, line 527

[6]gr-iridium/lib/iridium_burst_tagger_impl.cc, line 133

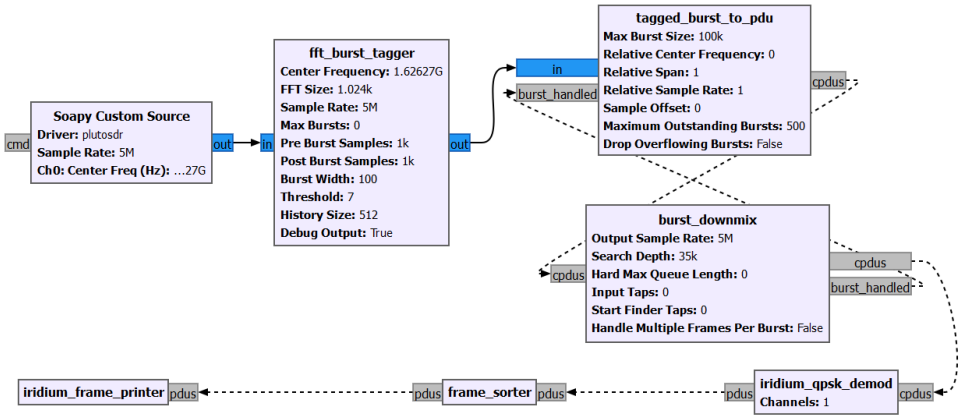


Figure 5.4: `gr-iridium` flowgraph

The **`tagged_burst_to_pdu`** block reads the tags and uses them to pack the raw samples belonging to a burst with the metadata from the tag into a standardised message format (a PDU), which is then passed into block output.

The **`burst_downmix`** operates on PDUs. For each PDU, it shifts the centre frequency by the rough estimate of CFO[7], applies a low pass filter, and decimates the signal. Then it searches for start of the burst by analysing magnitude of the filtered signal. Then it searches for the unique BPSK synchronisation word in the frame (see subsection 3.1.1 and Figure 3.1 for signal structure details) to acquire a fine estimation of CFO[8], which is again used to shift the signal. After that, the signal is filtered and a correlation function is used to find the start of the unique word and to determine the direction of the burst (uplink or downlink)[9]. Lastly, it determines the exact frame size, appends the PDU with the new information and passes it to the block output.

The **`iridium_qpsk_demod`** block demodulates the frame data. It decimates the signal to one sample per symbol, applies a PLL to remove remaining frequency or phase offset[10]. Then it performs a QPSK demodulation, checks the unique word and passes the modified PDU to the block output.

The **`frame_sorter`** block sorts the PDU by time of arrival (the processing is multi-threaded - ordering in time is not assured), and the **`iridium_frame_printer`** block prints the block into console or a file.

5.3 Data processing

After the capture, demodulation and decoding of Iridium frames, further processing is needed to create data suitable for navigation calculation.

[7] gr-iridium/lib/burst_downmix_impl.cc, line 804
[8] gr-iridium/lib/burst_downmix_impl.cc, line 520
[9] gr-iridium/lib/burst_downmix_impl.cc, line 612
[10] gr-iridium/lib/iridium_qpsk_demod_impl.cc, line 358

5.3.1 Parsing frames

The output of the `iridium-toolkit` is a text file, one line per frame containing the extracted information, which needs to be parsed for further processing. The output from `iridium-toolkit` is of the form: (1) frame type (e.g. IRA = Iridium Ring Alert); (2) Unix timestamp of the start of the recording[11]; (3) seconds from the start of the recording; (4) centre frequency of the frame; (5) demodulator confidence in percent; (6) signal parameters for the frame[12]; (7) frame data length; (8) direction of the frame (UP = uplink, DL = downlink); (9) frame-type-specific data. An example of a received frame is below.

```
1    2              3              4           5    6
IRA: p-289693-e000 000025108.8066 1626314368  91%  -29.47|
                7   8  9
-090.82|30.79 107 DL sat:067 beam:27 xyz=(+1390,+0092, ...
```

The frame-type-specific data for some frame types, namely IRA (Iridium Ring Alert, see subsection 3.1.1 for details) and IBC (a frame of the duplex channels), contain a satellite identifier, e.g. `sat:067`. However, this is a system internal ID - it is not public and has no relation whatsoever to the IDs published by NORAD[43]. Additionally, the IRA frames contain information about the satellite position in geodetic frame (`pos` in degrees latitude/longitude, `alt` in km), e.g.

```
pos=(+38.82/+003.79) alt=797
```

If the `alt` is high (generally above 700 km), the position refers to the actual satellite, if it is low, the position refers to the location on Earth where the satellite beam is hitting the surface[43]. Some nonsensical values were also observed and filtered out by only accepting frames with 700 km $<$ `alt` $<$ 900 km where satellite position information was extracted. The latitude and longitude is given with two significant digits of precision, which, for the latitude of Prague corresponds to accuracy of 1.11 km and 0.71 km respectively - not suitable for precise navigation. The process by which Iridium satellites calculate this position is not public, but it can be assumed it has similar errors as the SGP4 algorithm.

The final navigation system is capable of working with IRA and IBC frames. From these, the timestamps (2 and 3), frequency (4) and satellite ID (part of 9) are extracted for further usage where navigation is concerned, and also the satellite position (part of 9) is extracted from IRA frames in other applications.

5.3.2 Synchronisation of transmitter and receiver time

In the measurement setup used in this work, the timestamps provided by the SDR were incorrect. The reason for this was not found. An alternative way to obtain the start time, and a way to provide a limited receiver-transmitter

[11] In the setup in this work, this timestamp is incorrect

[12] Signal level (dB), Noise level (dB), SNR (dB)

time synchronisation, was found in the IBC frames. Occasionally, the IBC frames contain a timestamp (e.g. `time:2024-05-05T15:45:04.61Z`), which refers to the beginning of the slot in which said frame was sent. As the slot number is also included in the frame, it is possible to convert the timestamp to the time of frame reception[43]. This was used in this work to calculate the time of recording start and to correct the receiver time.

5.3.3 Determining frame transmission frequency

To calculate the Doppler shift of the received frames, it is necessary to identify the channel - and thus the channel base frequency - on which the frame was transmitted. For IRA frames, this is simple - the transmission frequency of Ring Alert frames is set at 1626.2708 MHz.

For the other frames, the transmitting channel can be identified as the closest centre frequency. The channel spacing, which is higher than the maximum Doppler shift for line-of-sight communication assures that there will be one and only one fit. However, this approach proved unreliable, as higher Doppler shifts were encountered (even in simplex channels where identification is certain).

An additional identification mechanism was thus implemented, where the received frames were sorted by the satellite ID, and a the channel was selected by the above method while observing that the relative Doppler shift to the base frequency among the frames received from a single satellite should not change rapidly:

$$\frac{f_{D_i}}{f_{B_i}} \approx \frac{f_{D_{i-1}}}{f_{B_{i-1}}}$$

However, this still did not prove sufficient and the incorrectly classified channels were a source of significant distortion. Therefore, the system uses only the IRA frames for navigation.

To filer out frames received with erroneous frequency, a thresholding filter was applied (f_T is the threshold of Doppler shift, equal to the channel spacing of $\approx 41.7\,\text{kHz}$):

$$f_{D_i} < f_T$$

5.3.4 Identifying the transmitting satellite

The satellite ID with an IRA or an IBC message identifies a satellite within the Iridium ecosystem. However, the rules by which an ID is assigned to a specific satellite are neither public nor apparent. Put simply, the ID within an Iridium frame has no relation to any other public ID of a satellite, such as those assigned by NORAD and used in TLEs. Additionally, as experiments have shown (see below), while NORAD IDs are constant in time, Iridium IDs are not. Therefore, to determine which Iridium frame ID corresponds to which NORAD ID, a mapping table needs to be constructed.

The creation of the mapping table (Iridium frame ID to NORAD ID) in this work takes advantage of the satellite position information contained within an IRA frame. This can be compared to the positions of all active Iridium satellites predicted by the SGP4 algorithm for the same time. The satellite with the closest predicted position to the position in the frame can be assumed to be the one that transmitted the frame. The algorithm (in pseudocode without vector optimisation) is in alg. 1, an example of the table is in Table 5.1.

Algorithm 1 Iridium satellite identification

```
satellites = download TLEs for Iridium
frames = load decoded IRA frames
saved_distances = list
frame_ID = table

for frame in frames:
    parse (RX_time, frame_ID, frame_position) from frame

    for sat_ID in satellites:
        predicted_position = run SGP4 for sat_ID at RX_time
        distance = predicted_position - frame_position
        save distance and sat_ID into saved_distances

    closest_ID = find satellite_ID for
                    minimum distance in saved_distances
    save closest_ID to frame_ID row in saved_IDs

for frame_ID in saved_IDs:
    most_common_ID = find most common ID in frame_ID row
    Iridium ID frame_ID is NORAD ID most_common_ID
```

This process is fairly computationally demanding and not completely error-free. However, when done over several hundred frames with a follow-up application of statistical methods, the satellites can be identified fairly certainly. Uncertainly identified satellites were filtered out by maintaining that the number of frames per satellite is larger than `MIN-FRAME-OCCURRENCE`.

Using a large dataset for mapping IDs

Assuming the frame IDs remain constantly attached to the same satellite, it would be advantageous to use a large dataset spanning several days or weeks to identify all of the satellites within the Iridium constellation. Such dataset can be found in [9]. The data come from Doha, Quatar, and are split into two parts. The first part, which was used for this work, spans 30 days between August 12$^{\text{th}}$ and September 10$^{\text{th}}$, 2020 and includes 2 060 289 IRA frames.

IRA ID	Most common NORAD ID	Count	2nd most common NORAD ID	Count	Confidence
036	122	8715	169	414	95%
038	113	8774			100%
039	148	7158			100%
040	153	7300	124	334	95%
...	...	...	...		
069	136	8930			100%
071	167	9314	170	190	98%
072	173	9110	175	16	100%
073	133	8807	169	340	96%
...	...	...	...		
104	116	8878			100%
107	142	7343	124	182	98%
109	130	8870			100%
110	114	9186			100%

Table 5.1: Example of ID mapping table from large dataset

IRA ID	NORAD ID Aug./Sep. 2020	Jan. 2024	Mar. 2024
046	158	156	156
049	163	159	159
057	119	**169**	**139**
...	...	...	...
073	133	100	100
078	107	**179**	**128**
079	100	129	129

Table 5.2: Example of ID mapping table changing in time

An excerpt from the mapping table is in Table 5.1.

However, a comparison of the table created from the large dataset to a table created from data measured as part of this work in March 2024 reveals the IDs are not in fact constant. Thus, an out-of-date, albeit large dataset cannot be used to create the table. Furthermore, some IDs were observed to change even during this work (Jan. to Mar. 2024). Therefore, a mapping table needs to be generated for each new data.

5.3.5 Identifying a satellite without decoding frames

Not all frame types include satellite position information or a satellite ID. Furthermore, not all satellite constellations transmit readily decodable signals in the first place. Thus, a mechanism for satellite identification without relying on frame decoding is needed.

One such mechanism (see alg. 2) is based on selecting satellites based on plausibility of the measured Doppler shift. First, a rough user location needs

to be known. Second, the constellation of the satellite needs to be determined. Thirdly, the channel needs to be identified to determine the Doppler shift. Then, the positions of all the satellites within the constellation are predicted using the SGP4 algorithm. Satellites with negative elevation angles, which cannot plausibly be received, are removed. For all the remaining satellites, the expected Doppler shift for a given channel is calculated. The satellite with the closest estimated Doppler shift is then selected. Note that because in this situation there is no way to distinguish frames belonging to one (albeit unknown) satellite, statistical methods cannot be used.

This method requires prior estimate of user location, therefore it cannot be used alone. On the other hand, it requires no demodulation or decoding, and is thus far more generic. It would be suitable for extending the navigation system to other constellations than Iridium, where the IRA-based method would provide the initial position estimate. Since the navigation system uses IRA frames, this method was not implemented.

Algorithm 2 Generic satellite identification

```
user_position = estimate of user location
satellites = download TLEs for constellation
frame = load a received frame
parse (RX_time, doppler_frequency) from frame

for sat_ID in satellites:
    predicted_position = run SGP4 for sat_ID at RX_time
    elevation = calculate elevation for
                    predicted_position and user_position
    if elevation < 0:
        continue the loop with the next sat_ID
    sat_doppler_freq = calculate doppler shift for
                            predicted_position and user_position
    save sat_doppler_freq and sat_ID

closest_ID = find sat_ID for
        min(abs(sat_doppler_freq - doppler_frequency) in
        saved data
```

5.3.6 Predicting satellite position

The final step in the data processing for navigation is to predict the position and velocity vectors of the satellite that transmitted each frame. To do this, TLEs for the satellite need to be downloaded, a prediction algorithm run and the resulting vectors converted to a proper frame of reference. This process is also commonly used in other parts of the navigation system (e.g. for satellite identification).

Up-to-date TLEs can be downloaded from e.g. CelesTrak[19]. For this purpose, a script was written which first checks if up-to-date TLE files for a given constellation can be found locally from a previous download, and if not, downloads TLEs from the server. In either case, it parses the TLEs, calculates the age of the TLE set and returns the TLEs as a dictionary. The script can be supplied with a path to a directory, in which case it loads the TLEs from there. In case data were collected but TLEs for them are no longer available, such as for the data from [9], a Special Request can be made on CelesTrak. The required TLEs are then sent by email. As these TLEs are packaged in files per-satellite rather than per-constellation, a script was created which performs the conversion and automatically selects a most up-to-date TLE set for a given time.

For estimation of satellite position, the widely used SGP4 algorithm was selected (see section 2.6 for details). In Python, this algorithm (in the version from [2]) was implemented e.g. in the `python-spg4` package[45], which was used in this work. If provided with a Julian date or an array thereof, the algorithm returns an array of position and velocity vectors in the TEME frame (see section 2.7).

To convert from the TEME frame to ITRS (ECEF), as well as to handle all other frame-of-reference conversions, the package `astropy`[46] was used. This package was also used to handle operations with times and dates, such as converting Gregorian dates to Julian dates.

Lastly, the data were packed into a two dimensional array for further calculations. The array is arranged so the "rows" (first index) are the data processed from frames, and the "columns" (second index) are

- absolute time of frame reception (Unix time i.e. seconds),
- centre frequency of frame (Hz),
- base frequency of frame channel (Hz),
- NORAD satellite ID (-),
- (3x) ITRS position vector (x, y, z; km),
- (3x) ITRS velocity vector (x, y, z; $\mathrm{km\,s^{-1}}$).

5.4 Calculating user position

For calculating the user position from the processed data, the Doppler-curve-fitting method was implemented (see subsection 2.3.1 for its theoretical basis). This method was chosen because it is transparent and thus suited for demonstration-type work, shows well the influences of error sources on the final position estimate, and can be developed iteratively and therefore allows for exploration of this specific navigation problem.

In principle, the method attempts to find a state vector $\vec{s}$, so that the metric function $F(c_M, c_E)$, where c_M is the measured Doppler curve and c_E

is a trial Doppler curve based on $\vec{s}$, is minimal. Simply, the method tries to estimate a state vector, whose corresponding Doppler curve provides the *best* match for the measured Doppler curve. The form of the vector $\vec{s}$, the specific metric and the method of creating the trial Doppler curve c_E, as well as the implementation of the search are described in detail below.

5.4.1 Estimating initial position

There are several ways to estimate an initial position for the method. In testing, the navigation system was shown to successfully locate the user while the initial position was well over 1000 km away from the final position, the estimate may thus be only very rough. On the other hand, different implementations of the method may require more precise estimate.

Firstly, external knowledge may be used, e.g. if the user knows they are in the Czech Republic, its geographic centre may be used.

Secondly, the position of any received satellite may be used, as those can be assumed to be relatively close. As apparent in Figure 5.5, the satellites are generally located within 2500 km, with the majority of captured frames coming from less than 1000 km away. An average of the satellite positions can be used to obtain a better estimate (see the figure). However, the reliability of this method depends on the field of view of the antenna being similar in all directions - if, for instance, the antenna is shielded from the north, the resultant position will be significantly shifted southward.

Thirdly, as described in subsection 2.3.1, the user is (in an ideal case) located on a line perpendicular to the satellite track, intersecting the satellite track at the position where the Doppler shift of the satellite signal is zero ($r_{f_D=0}$). It follows that the position on the satellite track which is closest to the user is $r_{f_D=0}$. Obtaining this position is straightforward as long as the measured Doppler curve for a given satellite passes through zero. The received data are split into curves per satellite and per channel, that is, a curve is formed by all observations with constant f_B and satellite ID. Furthermore, curves with longer gap between two frames than `MAX-TIME-GAP` seconds and split. This was necessary for very long captures, where a single satellite is captured on two or three passes. Finally, curves with fewer than `MIN-CURVE-LENGTH` frames are discarded. For each curve the point of $f_D = 0$ is interpolated. Positions from all successfully interpolated curves are then averaged to obtain the initial position estimate.

The navigation system in this work uses the third method, with the second method as backup if no curves were interpolated. For further navigation calculations, the curves are joined together again to form a single curve.

5.4.2 Trial Doppler curve generation

The Doppler frequency observed by a static user of a frame transmitted by a dynamic satellite is (see Equation 2.5)

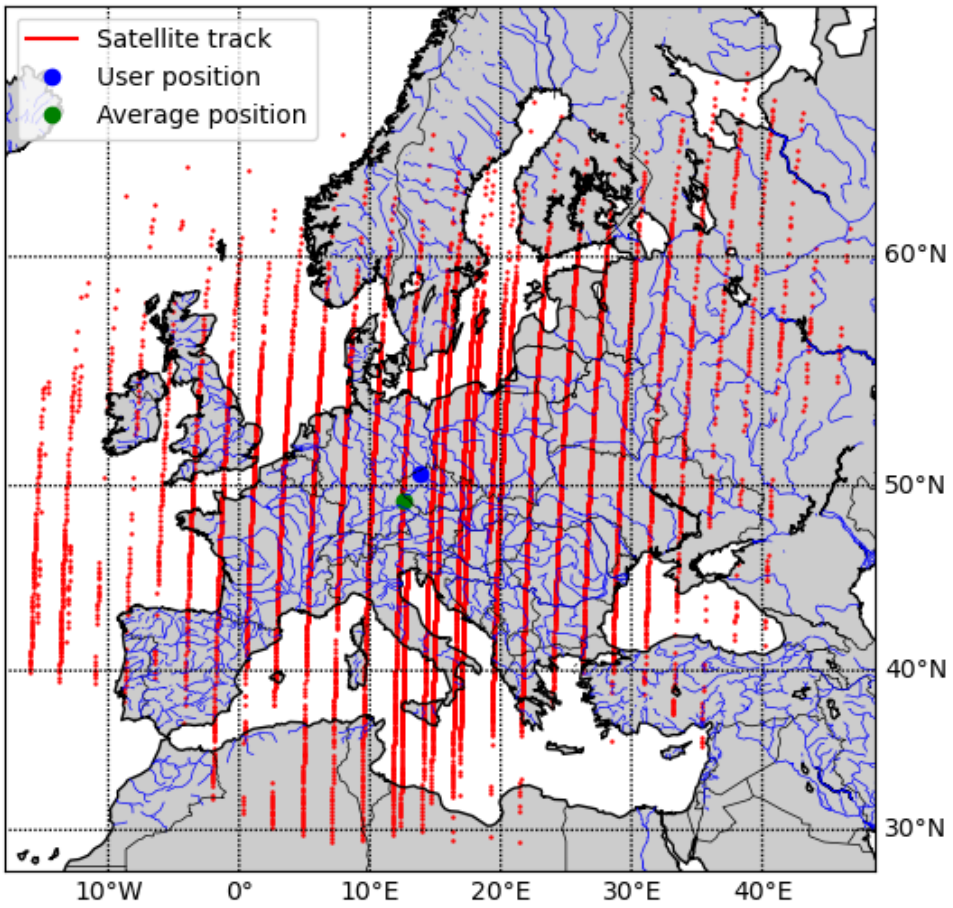


Figure 5.5: Example of tracks of captured satellites

$$f_D = \frac{1}{c}\dot{\rho} f_B \tag{5.1}$$

where the range rate $\dot{\rho}$ can be expressed as

$$\dot{\rho}_i = v_S \frac{r_S - r_U}{||r_S - r_U||_2} \tag{5.2}$$

where the satellite and user position vectors, r_S and r_U respectively, as well as the satellite velocity vector v_S, are expressed in ECEF frame.

Additionally, as mentioned in subsubsection 5.1.2, the received frequency is shifted by some amount, which is not constant. This frequency shift is modelled as a constant offset f_O (Hz) and time-variable drift.

Experimentation has showed that the drift is not linear nor it can be accurately modelled as a polynomial. However, it was also found that within a shorter time window of about 30 min, the drift can be approximated linearly. As a result, the drift is modelled as $f_{drift} = \dot{f}_O$, where $\dot{f}_O$ is the rate of change of the offset ($\mathrm{Hz\,s^{-1}}$).

Combining Equations 5.1 and 5.2 with offset and drift yields

$$f_D = \frac{1}{c} f_B v_S \frac{r_S - r_U}{||r_S - r_U||_2} + f_O + \dot{f}_O t \tag{5.3}$$

Inputting the time t, frequency f_B and vectors r_S and v_S into Equation 5.3 for each frame of the measured curve C_M at the trial user position r_U outputs the trial Doppler curve C_T, where both curves are functions of frequency by time.

Therefore, the navigation system needs to estimate the user position, the offset and the drift. For simplicity of algorithm debugging, and to ensure the user position estimate remains plausible (i.e. not beneath or high above Earth surface), the position is given in geodetic frame. The state vector is

$$\vec{s} = (\phi_U, \lambda_U, h_U, f_O, \dot{f}_O)$$

5.4.3 Metric

To evaluate how good is the trial position r_U, the trial curve and the measured curve are compared. To do this, a metric function $F(C_M, C_T)$ which for two curves yields a scalar value representing their similarity is needed.

The metric can take advantage of the fact that both C_M and C_T share the same time axis, i.e. both represent Doppler frequencies at the same times.

For this work, the metric e is given as the sum of squares of the differences in frequency of both curves:

$$e = \sum_{i=1}^{n} (C_{M_i} - C_{T_i})^2 \tag{5.4}$$

where n is the length of the two curves. The lesser the e, the more similar the curves are.

The process, which generates a trial curve for a given state vector $\vec{s}$ and compares it to the measured curve in order to output the metric e, is described in alg. 3.

5.4.4 Searching for the metric function minimum

Two approaches to find the metric function minimum, i.e. the best estimate of user position, were implemented. The first is iterative - it moves the trial user position in the direction with the best improvement in metric. The second is grid-search based - the algorithm searches through a matrix of possible state vectors and picks the one with the lowest metric.

The iterative algorithm is fairly fast and can converge even when the initial location is very far from the final estimate. On the other hand, it is theoretically prone to convergence into local minima. Conversely, the grid search algorithm is very slow and requires the initial estimate of the user position to be fairly close[13], however, it is resilient to local minima. In the final system, only the iterative algorithm was used.

[13]While the algorithm can search an arbitrarily large area, it would take a *very* long time

Algorithm 3 Evaluation of trial Doppler curve

```
def check_trial_curve(lat_u, lon_u, alt_u, off_u, dft_u,
                      measured_data, r_s, v_s):
    curve_len = lenght measured_data
    # measured curve = [times, doppler_frequencies]
    measured_curve, f_B = extract from measured_data
    start_time = min(measured_curve[:, 0])
    c = 299 792 458 m/s  # speed of light

    trial_curve = empty array of shape (curve_len, 2)
    trial_curve[:, 0] = measured_curve[:, 0]  # copy time axis
    r_u = convert (lat_u, lon_u, alt_u) to ECEF

    # range rate (scalar mult. and norm are along asix 0)
    Ro_dot = v_s * (r_s - r_u) / ||r_s - r_u||
    # doppler frequency
    f_D = Ro_dot * f_b / C
    # drift - substract start of measurement
    f_dft = (trial_curve[:, 0] - start_time) * dft_u
    # trial curve = doppler frequency + offset + drift
    trial_curve[:, 1] = f_d + off_u + f_drift

    # Calculate metric
    var = sum((measured_curve[:, 1] - trial_curve[:, 1]) **2)

    return var
```

The iterative search

The iterative-search algorithm takes a current best estimate of the trial state vector $\vec{s_0}$ and calculates the metric e_0. Then, for each member $\vec{s_{0k}}$ of the state vector $\vec{s}$ (e.g. longitude or drift), it moves the trial state vector by a corresponding step Δ_k in both directions ($+\Delta_k$, $-\Delta_k$) and calculates the metric (e_{k+}, e_{k-}), whilst keeping the other members of the state vector constant. Then it compares the new metric with the original one (e_0) to obtain the best direction δ_k in which to move the state vector:

$$\delta_k = \text{sign}(\max(0, e_0 - e_{k+}) - \max(0, e_0 - e_{k-}))$$

If $\delta_k \neq 0$, the metric decreased in one or directions, and member $\vec{s_{1k}}$ of the state vector for the next iteration is thus modified by adding or subtracting the step Δ_k. If $\delta_k = 0$, the member is kept constant, but the step size is decreased by $\dot{\Delta}$, down to step limit $\Delta_{k_{MIN}}$:

$$
\begin{aligned}
&\text{If } \delta_k \neq 0: \\
&\quad \vec{s_1}_k = \vec{s_0}_k + \delta_k \dot{\Delta}_k \\
&\quad \Delta_{k_1} = \Delta_k \\
&\text{If } \delta_k = 0: \\
&\quad \vec{s_1}_k = \vec{s_0}_k \\
&\quad \Delta_{k_1} = \min(\dot{\Delta} \cdot \Delta_k, \Delta_{k_{MIN}})
\end{aligned}
$$

Variable step size, where the δ_k is used as the coefficient for the step size instead of the $\dot{\Delta}$, was tried, but it was found that this creates an insignificant decrease in the number of iterations compared to problems related to overstepping. Thus, fixed step size was used.

After checking all members of the state vector, the algorithm continues in another iteration with $\vec{s_0} = \vec{s_1}$ and $\Delta_k = \Delta_{k_1}$ for all k. The algorithm ends either when the number of iteration exceeds `ITERATION-LIMIT`, or when it converges. The convergence criterion is all step sizes are at their respective limit and moving in no direction improves the metric:

$$\text{convergence} = \text{all}_k(\Delta_k = \Delta_{k_{MIN}}) \ \ \& \ \ \text{all}_k(\delta_k = 0)$$

Because the algorithm was shown to be prone to cycles, a cycle detector was implemented, which looks at the last two sets of δ_k. If their member-wise sum equals the zero vector, a cycle is detected and the step size of all members with $\delta_k \neq 0$ is forcibly decreased.

The number of trial curve checks is a useful metric to evaluate the algorithm speed. The algorithm performs $n + 1$ trial curve checks per iteration, where n is the size of $\vec{s}$. In testing, about 200 to 400 iterations, therefore 1200 to 1600 searches ($n = 6$), were most common.

While the algorithm is not protected against convergence into a local minimum, this was not observed during testing very often, and the local minima that were observed were within hundreds of meters of the global minimum showed by the grid search method (see below).

An example of the algorithm search is in Figure 5.6, with the final estimated and measured curves in Figure 5.7.

The grid search

The grid search algorithm performs the trial curve checking for a set of combinations of $\vec{s}$ and picks the one with the best metric. The process is then repeated M times with decreasing grid spacing.

First, an n-dimensional grid, where n is the length of the state vector $\vec{s}$, is constructed. Each axis belongs to one member k of the state vector, and contains g_k values spaced apart by step Δ_{k_m} ($m \in M$), centred on $\vec{s_0}_k$. Then, for each point of the grid, the trial curve checking is performed. Finally, the lowest metric with the corresponding state vector $\vec{s_1}$ is found.

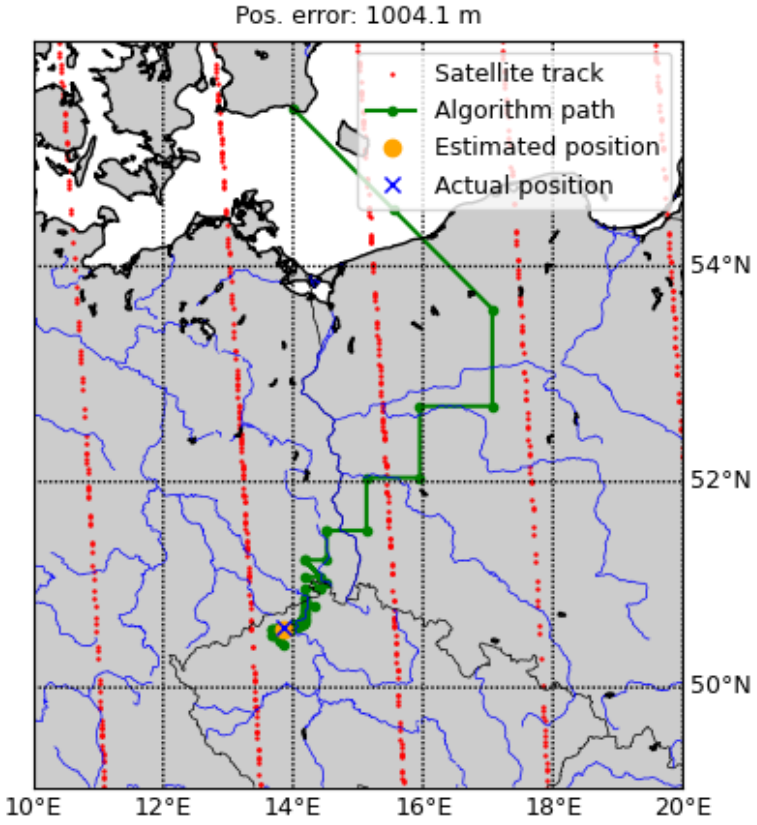


Figure 5.6: An example of the iterative curve fitting algorithm search

Once the $\vec{s_1}$ is found, m is incremented and a new grid is constructed based on $\vec{s_1}$ and Δ_{k_m}, which is chosen so $\Delta k_m \approx \Delta_{k_{m-1}}/g_k$ - this way, the new grid fills one cell of the previous grid. Once the step sizes (and thus the resolutions) are sufficiently small, the search is completed.

The step sizes are fixed as parameters in the form of a list of size M. The grid size g_k is fixed for each state vector member in all iterations. If the step size in a given iteration is set to zero, the member is considered fixed in the previous value.

The algorithm performs $M \cdot \Pi_1^n g_k$ trial curve checks in total. If this algorithm was to be used on its own, a sufficiently large step in horizontal coordinate would have to be in the order of 100 km for $g = 10$, and likewise for the other state vector members. Thus, a total of $5 \cdot (10 \cdot 10 \cdot 10 \cdot 5 \cdot 5) = 125\,000$ was sometimes needed, making this approach very slow. On the other hand, convergence to the global minium is assured, as long as the the actual user position lays within the first (largest) grid, which places stricter requirements on the estimation of the initial position than the iterative method.

To alleviate this shortcoming, a variation was considered. If the best-fitting state vector for a given grid was located at the edge of the grid, the grid would be extended in that direction. As this algorithm ended up not being

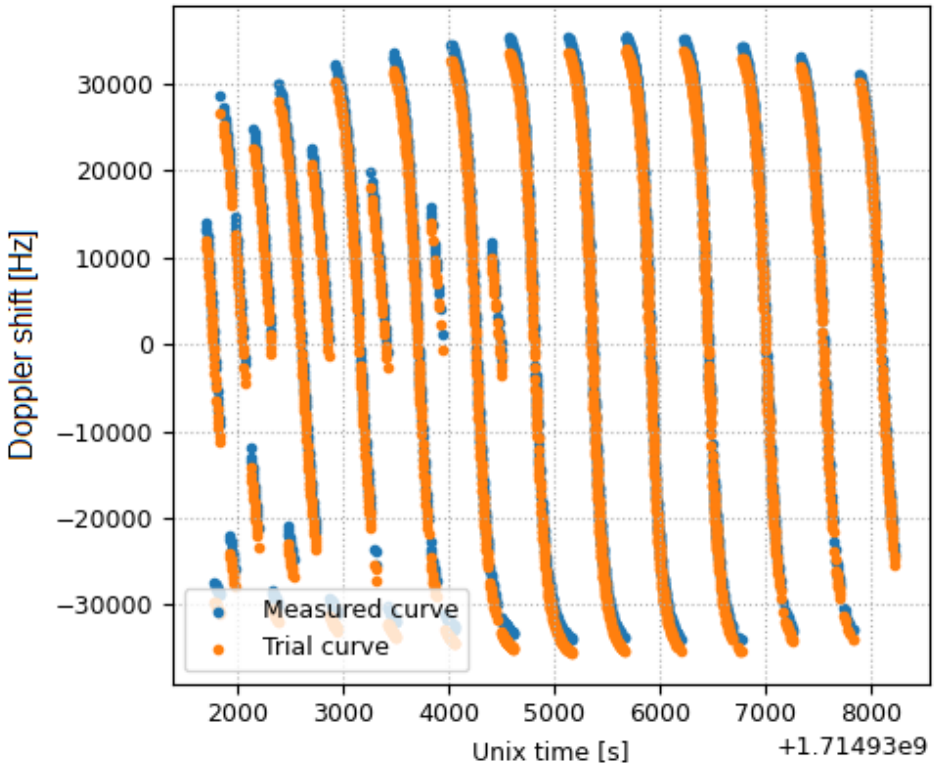


Figure 5.7: Doppler curves from final estimation and measurement

used on its own, this was not implemented.

One possible application for this algorithm is as a final step to the iterative search, which solves the problem of convergence into a local minimum close to the actual user position. However, as any meaningful search would still require 10^4 trial curve checks, significantly slowing down the overall calculation, for no observed benefit, this was not used.

5.5 Navigation system implementation and structure

The navigation system is written in Python programming language. This language was selected for four reasons. Firstly, it is widely used in Academia, it contains many scientific packages and supports all the functions required by this work. Secondly, Python supports GNURadio, which greatly enabled the signal processing part of this work. Thirdly, the author is a professional programmer working in Python, and is thus very familiar with the language. And finally, the author believes that Free and Open Source software should be the go-to choice for Academia, and writing the application in Python means that anybody can access it without financial burden.

The application is written in accordance with Python coding standard, makes use of Python packages for some operations and implements its own

functions for others. The code is encapsulated, uses classes where appropriate and is documented. Version control software (Git) was used extensively in development.

The code of the navigation system is structured to enable operation which is independent of the specific satellite system being received. Data are passed between parts of the program in standardised structures and the navigation calculations are generic. Thus, the system should work with minimal changes (once the radio support is implemented) also with e.g. Orbcomm or Globalstar.

The organisation of the project is shown in Figure 5.9

5.5.1 Additional tools created as part of this work

Some additional tools were created to aid in analysis, development and testing of the navigation system created in this work. They all fit within the `/app` section.

`generate_test_data`. This script creates simulated data, which was very useful in development of the navigation algorithms as well as in analysis of the frequency of the received frames. Plotting the received frames (t, f) against the simulated data shows most clearly the frequency offset and drift induced by the local SDR oscillator errors. The process of creation of the simulated data is essentially equal to the one used in trial curve checking (see Equation 5.3), with preset position. Data points are generated for each satellite present in the measured data, in one second increments for the entire duration of the measured data. An elevation filter is applied to remove signals form satellites which rare beyond the horizon, and only signals with Doppler shift of less than $\pm 36\,\mathrm{kHz}$ are kept.

`predict_satellite_passes`. This application predicts which satellites will be visible in the near future. It first provides a summary of all visible satellites, with exact times and azimuths of rise and fall and maximum elevation. Then it each 2 seconds prints all the visible satellites right now, with azimuth and elevation. This was useful in the early stages of developing the system to capture the satellite signals.

`read_time`. This scripts handles time synchronisation on the PlutoSDR, as the device was shown to not keep time properly. It connects over serial port, logs in and sets the device time to current system time using the `date` command. This is done when the system time seconds counter changes, to achieve maximum accuracy (since the `date` command accepts only whole seconds).

`convert_tles_from_special_request`. This is a simple script which converts TLE files from CelesTrak special request (one file per satellite with all TLEs for a given period) into the format used for current TLEs by CelesTrak (only one file containing one TLE set each per satellite). For each satellite, the closest TLE set for a given date and time is picked. This was useful for

retroactive download of TLEs (when the author forgot to download them during data capture).

`find_norad_ids_in_sat08_data.` This script was used to identify all the satellites in the large dataset from [9], as described in subsubsection 5.3.4.

`noise_analysis.` This script was used to evaluate the noise resilience of the system. Gaussian noise of increasing noise standard deviation and mean value was added to simulated data and then the navigation algorithm was run, to get an idea about the influence of noise on the precision of the navigation system in the design process.

5.6 Summary

The navigation system is a Python application capable of calculating the user position from received satellite signal, without any prior knowledge of the user position. It is built to be independent of the actual satellite system used to navigate, and to be used with generic SDR and antenna setup. However, the support of only Iridium and PlutoSDR are actually implemented and verified.

First, the system receives, demodulates and decodes Iridium frames, using GNURadio with `gr-iridium` and the `iridium-toolkit` for the signal and frame processing.

Next, the navigation system processes the received frames. It identifies on which channel was a frame transmitted. It is capable of working with any channel type which includes satellite identifier, but only IRA frames were used in its final form. The system downloads up-to-date TLEs from the internet and uses them for SGP4-algorithm predictions. The system matches Iridium internal IDs with NORAD IDs, and predicts the position of the transmitting satellite at frame reception time.

Then, it uses the processed data to estimate the initial position for the navigation calculations. To do this, it interpolates the moment of zero-Doppler-shift for each Doppler curve, and places the initial estimate at the position where the moment occurs, averaged over all interpolated curves.

Finally, the system calculates the user position using the Doppler-curve-fitting method in the iterative optimisation algorithm. For each trial position, the metric for the measured and trial curves is calculated, and the increase or decrease of the metric in all directions is evaluated. The trial position is then moved by a set step in the direction of the largest decrease of the metric, until it no longer decreases in any directions or the iteration limit is exceeded.

The performance of the navigation system is examined in chapter 6.

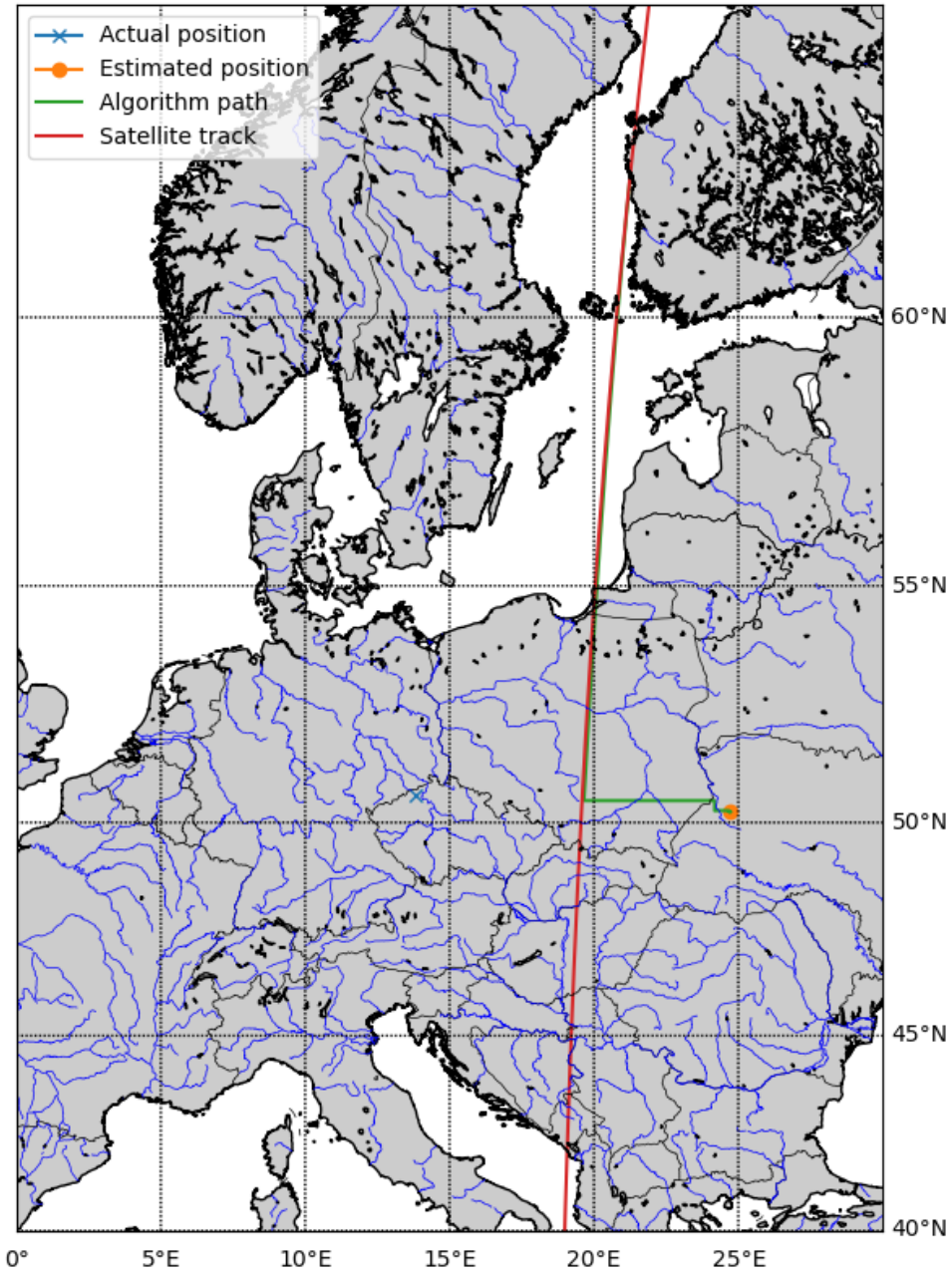


Figure 5.8: An example of the algorithm finding the symmetrical solution

```
+--Data - folder with saved data from measurements
+--External - folder for external packages, excluded in Git
| +--iridium-toolkit - the iridium-toolkit package source code
|--src - navigation system source code
 +--app - executable files
 | +--data_analysis - data analysis and visualisation scripts
 | +--development - other scripts used in development
 +--config - configuration files, parameters of algorithms etc.
 +--navigation - data processing, algorithms for navigation
 +--radio  - tools of processing frames for specific satellite
constellations
 +--satellites - scripts working with satellites, TLEs and SGP4
 |--utils - utilities for debugging, plotting and logging
```

Figure 5.9: Navigation system project structure

Chapter 6

Experimental demonstration of SoP positioning

This chapter describes the evaluation of the navigation system, the data used in testing and the navigation system performance. First, the evaluation data are discussed, as they provide a useful insight into the radio environment of Iridium signals. Second, evaluation methods are explained and results presented. Then, the navigation system is compared to the existing SoP systems. Finally, the details of the system performance are discussed and some explanations offered.

6.1 Navigation data

Ten data sets were collected for the purpose of navigation system testing and validation (*validation* data, prefix `val`), in addition to data which were used during development (*experimental* data, prefix `exp`). An example of the data is in Figure 6.1. All data are present in the digital appendix.

6.1.1 Data parameters

Select parameters of each data set are in Table 6.1. These include the time of the start of the collection (UTC), duration and length (in data points, i.e. processed IRA frames), the number of satellites whose transmitted frames are present in the data set, mean age of the TLEs, and mean signal parameters - the signal level, the level of the ambient noise, and Signal to Noise Ratio (SNR).

Detailed parameters of the validation data are in Table 6.6. Those are presented as minimum, mean, and maximum values across all data sets. In addition to the parameters mentioned above, the table includes the number of received frames per satellite in the data set, visibility time of satellites, the latitude and longitude of the satellite at the time of frame reception, the extent of satellite coverage, which refers to the distance to the furthest satellite in a given geographical direction (North, South, East, West), the count of some frame types in the data sets, the confidence of the demodulator in the demodulated bits, and the measured Doppler shift.

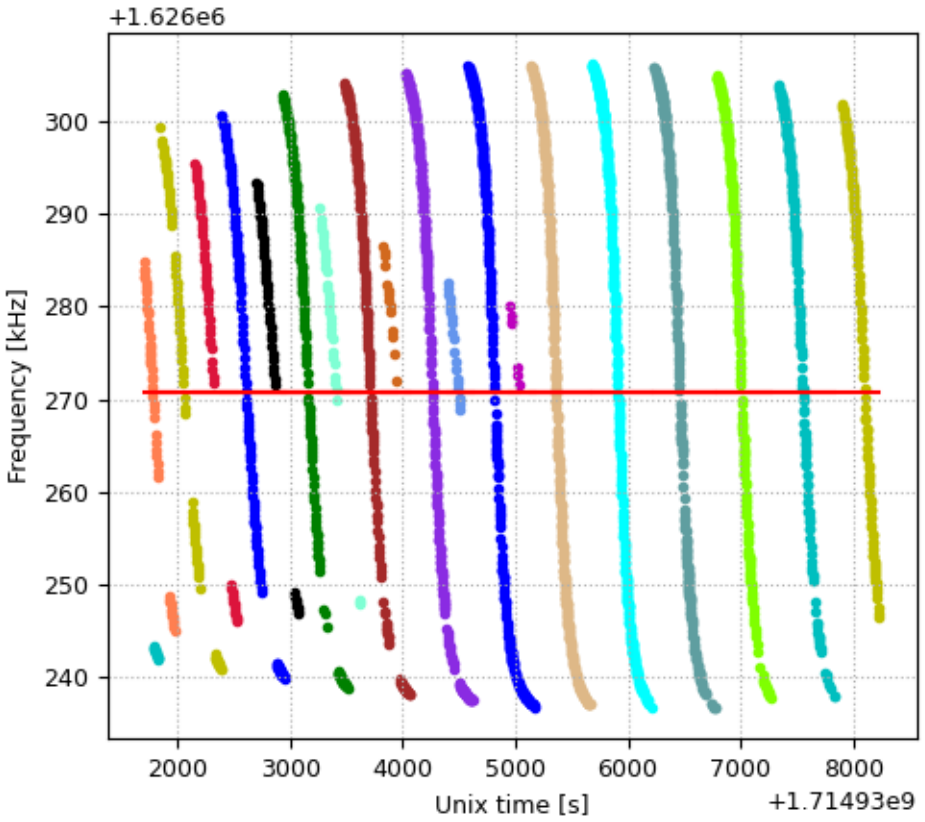


Figure 6.1: Example data (`val02`), colour-coded for different satellites, with the IRA transmission frequency shown in red

Of particular importance is the TLE age, as the errors in predicted satellite position can reach several kilometres when the TLE set is one day old (see section 2.6). In the validation data, the mean TLE age is 16 hours and ages up to 40 hours were observed. This also demonstrates that the TLE publishing is not regular.

The mean satellite visibility time is around 7 min, which corresponds to the theoretical visibility time for Iridium from Table 3.1. The maximum visibility time of over an hour is due to one satellite being captured on multiple passes during one measurement.

Interestingly, the maximum Doppler shift of 37.7 kHz is larger than the calculated maximum of 36 kHz from Table 2.1. This suggest that a limited reception from beyond the horizon is possible. The mean Doppler shift is positive, which suggests the mean direction of the captured frames is biased to the south, as most of the Iridium constellation orbits (from Earth-based observer perspective) from the south to the north. This is easily explained by the measurement site geographic layout (there is a tree and then a hill right to the north).

ID	Start time	Duration	Length	Sat.	TLE age	Signal param.*
val01	2024-05-05 15:11:42	01:06:35	1766	15	21:04:31	-22.3|-83.5|28.9
val02	2024-05-05 17:55:00	01:48:49	3301	18	27:36:32	-22.2|-83.2|28.8
val03	2024-05-05 19:45:47	01:21:15	1474	8	26:12:40	-21.8|-82.4|28.5
val04	2024-05-06 15:20:49	00:40:58	1017	10	02:21:15	-22.5|-82.8|28.3
val05	2024-05-06 18:30:19	00:55:05	1955	10	05:45:37	-23.5|-84.6|28.9
val06	2024-05-13 12:22:27	01:06:39	4141	18	14:11:58	-25.0|-86.3|29.1
val07	2024-05-13 14:45:22	01:06:37	2404	15	15:43:39	-24.4|-85.7|29.0
val08	2024-05-13 16:58:01	00:39:51	2958	6	18:59:35	-25.8|-86.7|28.7
val09	2024-05-13 18:49:33	00:58:20	2005	14	20:35:00	-24.0|-85.2|29.0
val10	2024-05-14 12:43:10	01:23:19	4420	13	16:50:59	-24.3|-85.7|29.2

*Expressed as Signal level (dB) | Noise level (dB) | SNR (dB)

Table 6.1: Captured data overview

6.1.2 Data collection

The validation data were all collected during the first half of May 2024, in four batches on May 5th, 6th, 13th and 14th, using the measurement setup described in chapter 5. The desired duration of a single dataset was 1.5 hours, but this was not always possible due to the weather, as the measurement setup is not rainproof. Generally, the validation data were collected from the afternoon to late evening[1], under clear to thickly clouded sky.

Each data set was trimmed to remove the frequency variations caused by temperature fluctuations in the SDR after startup. The NORAD IDs of the satellites captured within the data were estimated and used to construct the NORAD ID to Iridium ID mapping table (see Table 6.2 for the version valid for the validation data). The data were then processed by the navigation system to produce user position estimates. The parameters of the navigation system, mentioned throughout chapter 5, are summarised in Table 6.3 along with the values used in the validation data processing.

[1]Note that the times in Table 6.1 are in UTC and local time during measurement is CEST, so 2 hours ahead.

100: 73	120: 38	139: 57	155: 25
102: 112	121: 42	140: 39	156: 46
103: 103	122: 44	141: 51	157: 6
104: 110	123: 48	142: 82	158: 18
106: 114	125: 69	143: 43	159: 49
107: 115	126: 71	144: 74	160: 90
108: 2	128: 78	145: 5	163: 3
109: 4	129: 79	146: 107	164: 13
110: 9	130: 85	147: 7	165: 23
111: 16	131: 87	148: 77	166: 96
112: 17	132: 88	149: 30	167: 67
113: 24	133: 89	150: 40	168: 68
114: 26	134: 92	151: 111	171: 81
116: 28	135: 93	152: 22	172: 72
117: 29	136: 99	153: 8	173: 65
118: 33	137: 104	154: 94	180: 50
119: 36	138: 109		

Table 6.2: Iridium NORAD ID to IRA ID mapping table (X:Y, meaning Iridium X is `sat-id` Y in IRA frames), valid for the measured data

Step size	Initial	Final	Value limit	
Latitude	100×10^3	1		m
Longitude	100×10^3	1		m
Altitude	10,	1	0 to 3000	m
Offset	3000	1		Hz
Drift	0.1	0.001		Hz/s
`MIN-CURVE-LENGTH`	10 frames			
`MAX-TIME-GAP`	60 s			
`ITERATION-LIMIT`	500			
`MIN-FRAME-OCCURRENCE`	10 frames			

Table 6.3: Parameters of the navigation systems

6.2 Navigation system performance

The main parameters of the performance of the navigation system are

1. Accuracy or Bias, i.e. the distance from the estimated user position to the actual user position. Both 2D (horizontal) and 3D (accounting for altitude) accuracy is presented, but in further analysis (e.g. accuracy as a function of measurement time) only horizontal accuracy is considered.

2. Precision or Standard Deviation, i.e. the deviation of the estimated user position from the mean estimate in the horizontal plane only. Three metrics were considered - all of the data, best 95 % and best 50 % of the data.

ID	Lat. (°)	Lon. (°)	Alt (m)	Offset (Hz)	Drift (Hz/s)	2D err. (m)	3D err. (m)
val01	50.571	13.839	222	2004	0.131	2124	2124
val02	50.581	13.846	300	2140	-0.173	1000	1004
val03	50.580	13.843	0	990	-0.131	1063	1083
val04	50.580	13.839	0	774	0.256	1155	1173
val05	50.568	13.859	1060	908	-0.085	2674	2807
val06	50.578	13.835	608	5004	0.127	1395	1452
val07	50.580	13.839	928	5412	0.013	1137	1347
val08	50.577	13.819	212	4941	-0.217	2237	2237
val09	50.562	13.837	724	4105	-0.311	3102	3145
val10	50.571	13.858	692	4421	0.022	2367	2417

Table 6.4: Navigation system estimates

Data	Mean lat. (°)	Mean lon. (°)	Bias (m)	Std. dev. (m)
All				1611
50%	50.575	13.842	1669	572
95%				1467

Table 6.5: Accuracy and precision

3. Time To Fix (TTF), i.e. the measurement time required to achieve an estimate. As the system is capable of estimating user position from very few frames, but such estimation have enormous errors, the TTF is defined as the time at which the estimated position accuracy begins to approach the eventual accuracy of the whole data set.

4. Number of satellites required to produce an estimate. Same concerns apply here as for the TTF.

The user location estimate produced by the navigation system for each data set, along with the 2D and 3D errors, is in Table 6.4.

6.2.1 Accuracy and precision

The accuracy of the navigation system (see Table 6.5) was calculated from the mean of all the position estimates, and it is 1669 m. The precision of all data is 1611 m, of the best 50 % it is 572 m, and of the best 95 % it is 1467 m. The estimates are visualised in Figure 6.2. The corresponding probability distribution function is in Figure 6.3, where the blue line represents the 2D error relative to the actual position, and the red line represents the 2D error relative to the mean estimate.

All of the estimates are to the south of the actual position - in fact, the mean error in the E-W direction is 144 m eastwards (as opposed to the N-S directional error of 1662 m southwards). Furthermore, the grouping of the estimates around the mean is more tight in the N-S direction than in the E-W direction (2090 m and 2836 m, respectively).

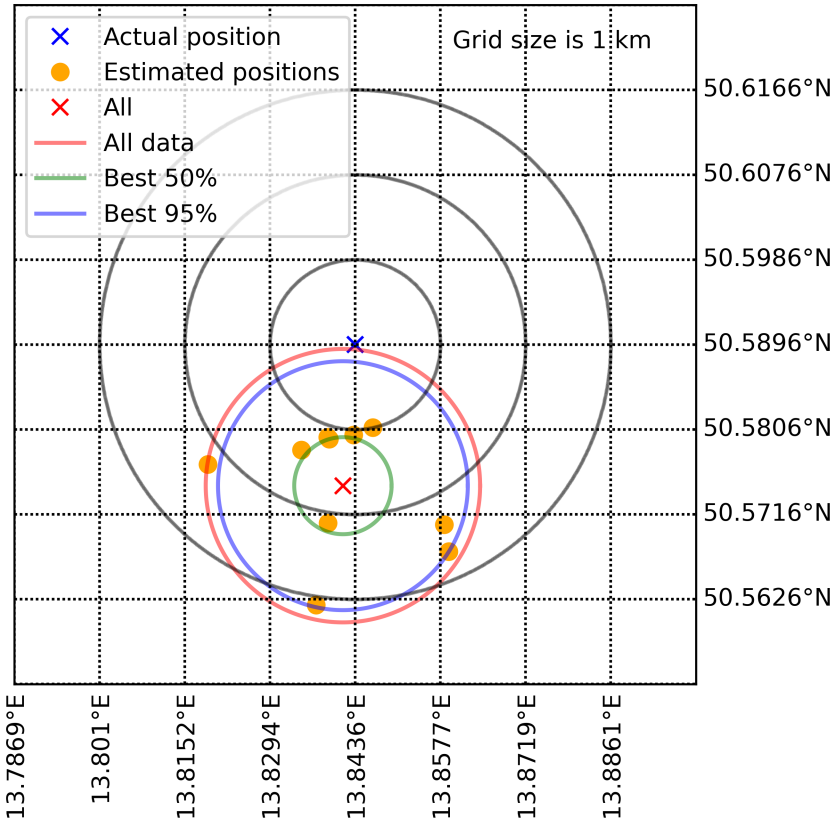


Figure 6.2: Estimated user positions from the validation data

6.2.2 Accuracy and measurement time

The accuracy of the navigation system as a function of measurement time is shown in Figure 6.4. It was calculated by splitting the validation data into 5-minute chunks and solving the user position using the cumulative data. The blue curve is the mean across the validation data, with error bars representing the minimum and maximum values in the data set. The datasets shorter than one hour were excluded. The orange line is the frame count in time (mean across validation data). The relationship between measurement time and frame count is linear. Based on Figure 6.4, the time to fix of the navigation system can be said to be 25 to 30 minutes.

6.2.3 Accuracy and number of satellites

The accuracy of the navigation system as a function of received satellites is shown in Figure 6.5. Similarly to the measurement time function, it was calculated by splitting the validation data into chunks by satellite, in the descending order of received frames, and solving the user position using the cumulative data. The plot is laid out similarly to Figure 6.4. The relationship between the number of received satellites time and frame count is not linear,

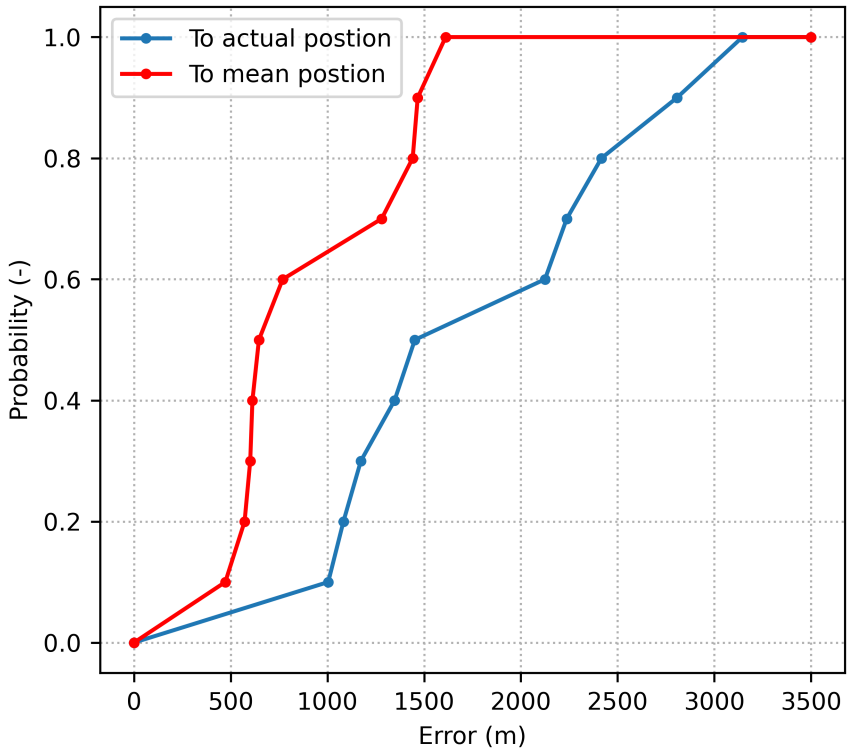


Figure 6.3: Accuracy and precision cumulative distribution function

however, which is due to the sorting of the chunks by size. Based on Figure 6.5, it can be said the that the system can reliably function with Doppler curves from as few as two satellites.

6.3 Discussion of the results

As seen in Figure 6.2, the distribution of the user position estimates is significantly biased south and slightly west. In fact, the bias roughly follows the direction of the satellite track (see e.g. Figure 5.5). This is a systematic error of the method, the cause of which is not known to the author.

It may arise due to incorrect timing, as during development, it was shown that an incorrectly set time of start of recording would move the estimate in the direction of the satellite tracks, by approximately 5 km northward for each added second. Another explanation might be incorrectly estimated frequency offset, or combination of both factors. It is important to note that signal flight time is not compensated, which may account for some of the discrepancy.

As mentioned previously, the distribution of the user position estimates is more tight in the N-S direction (approx. parallel to the satellite tracks) than

in the E-W direction (approx. perpendicular to the satellite tracks), which is inconsistent with the reviewed literature (e.g. [37, 3]), where the distribution was significantly broader in the direction perpendicular to the satellite track. This may hint at a possible advantage of the curve fitting method, however, the results presented here are neither numerous nor distinctly distributed enough to prove this.

In both the measurement-time and satellite-number accuracy functions, the accuracy generally improves with measurement time or satellite number, respectively. However, the accuracy can also decrease with a step in meas. time or the number of satellites. This may be due to the added data being of lower quality (e.g. in `val03` there is a gap caused by a drop in communication), or by the added data having a different drift, which the linear estimator cannot compensate.

Comparison of the measurement time and satellite number accuracy functions reveals the system works much better over a small set of complete Doppler curves than over longer, but incomplete data. This is not surprising, as the method chosen was designed to work over curves rather than data points. Theoretical minimum TTF is therefore around 10 minutes, as this is the time required for two consecutive satellites to pass over the user (based on the mean satellite visibility time).

The overall accuracy of the system is lowered by several factors. Firstly, the the satellite positions and velocities were calculated by SGP4 propagation from TLE sets, which can have substantial errors. However, this cannot be avoided unless a dedicated satellite tracking system is developed. It seems that the errors in the TLE-SGP4 positions are somewhat random, as the estimates with old TLE sets are more accurate than the satellite position estimates, as suggested by the analysis in section 2.6.

Secondly, ionospheric and tropospheric delays were neglected, as was the time of flight of the signal. Furthermore, the satellite clock was assumed to be perfectly accurate, which is likely not the case.

Thirdly, the time of reception of the satellite signal may not be accurate, as it was not provided by the SDR but rather by synchronisation with Iridium IBC frames. There were occasional step changes in the estimated recording start time of hundreds of milliseconds throughout the recordings, which may suggest a timing inaccuracy on the part of either the SDR or the Iridium signals themselves. Note that this method of time synchronisation was not found in literature by the author nor it could verified in this work.

Most important source of inaccuracy is thought to be the SDR frequency offset and drift. This can be somewhat compensated e.g. by introducing a temperature-coupled compensator, or by better estimation of drift by the navigation system, as the linear compensation proved somewhat insufficient.

6.4 Comparison to existing SoP positioning systems

To compare the navigation system (NS) against the existing SoP positioning systems (reviewed in chapter 4), the two model SoP systems from section 4.3 are used: SoP A and SoP B. The SoP A has an accuracy of 200 m and time to fix of 10 min, whereas the SoP B has an accuracy of 700 m and TTF of 2 min (see Table 4.2). Note that some of the systems used to construct these example systems utilise sensor fusion with e.g. INS or altimeter. The NS has an accuracy of 1669 m and TTF of 30 min. Precision or accuracy as a function of number of satellites is not known for the SoP positioning systems from the literature, and therefore cannot be compared.

The navigation system developed in this work is less accurate than the SoP systems from the literature. This is partially due to the existing systems generally using more complex error estimation, especially the estimation of the receiver (and in some cases transmitter) frequency drift. Another factor might be the issues with timing, which were not present in the reviewed systems.

The NS generally requires more time to acquire a fix than the reviewed systems, which is an attribute of the curve-fitting calculation method. The TTF could be significantly lowered if more constellations were used, as this would enable simultaneous complete Doppler curves.

Finally, unlike the existing systems, the NS is a full-stack system capable of performing all the required tasks, which introduces many more avenues for errors, such as satellite or channel misidentification, and makes the identification and correction of these errors more difficult due to the complexity of the system.

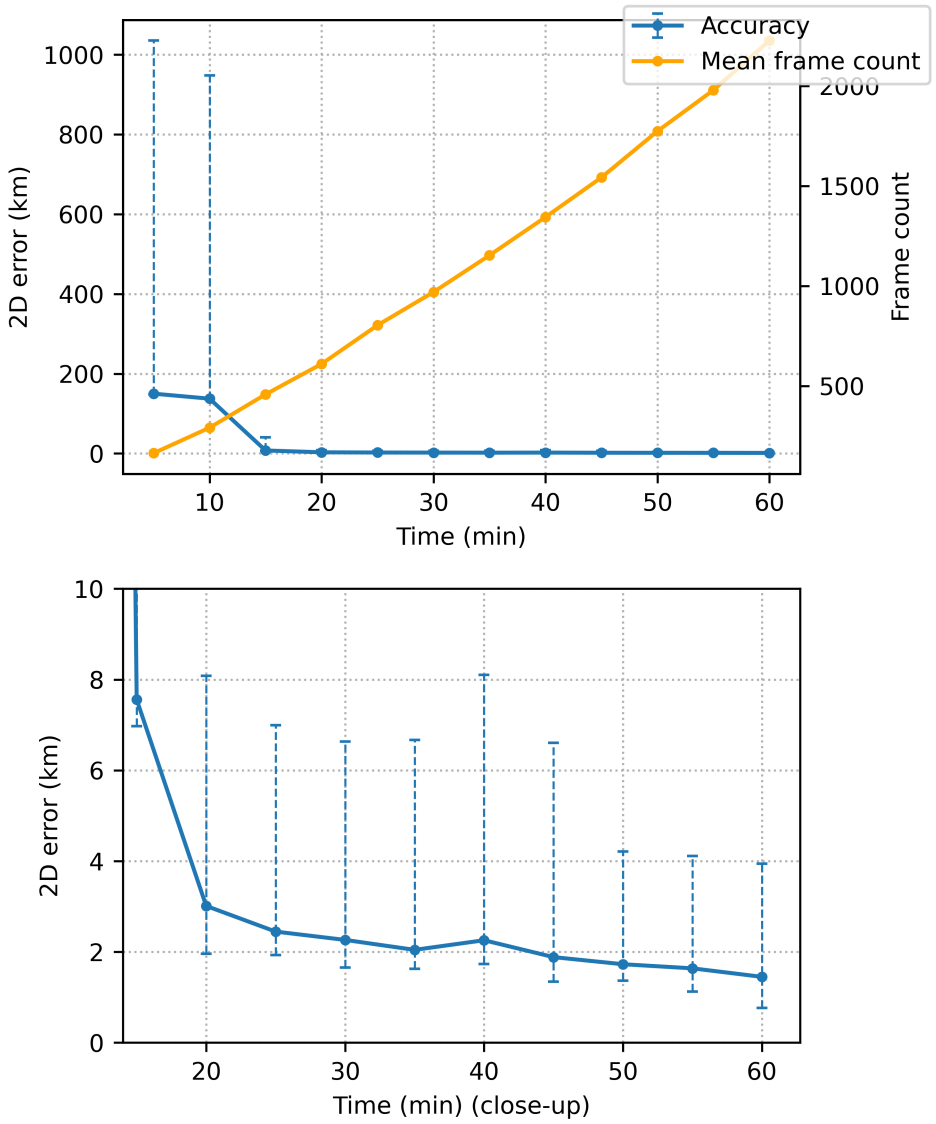


Figure 6.4: Accuracy as a function of measurement time

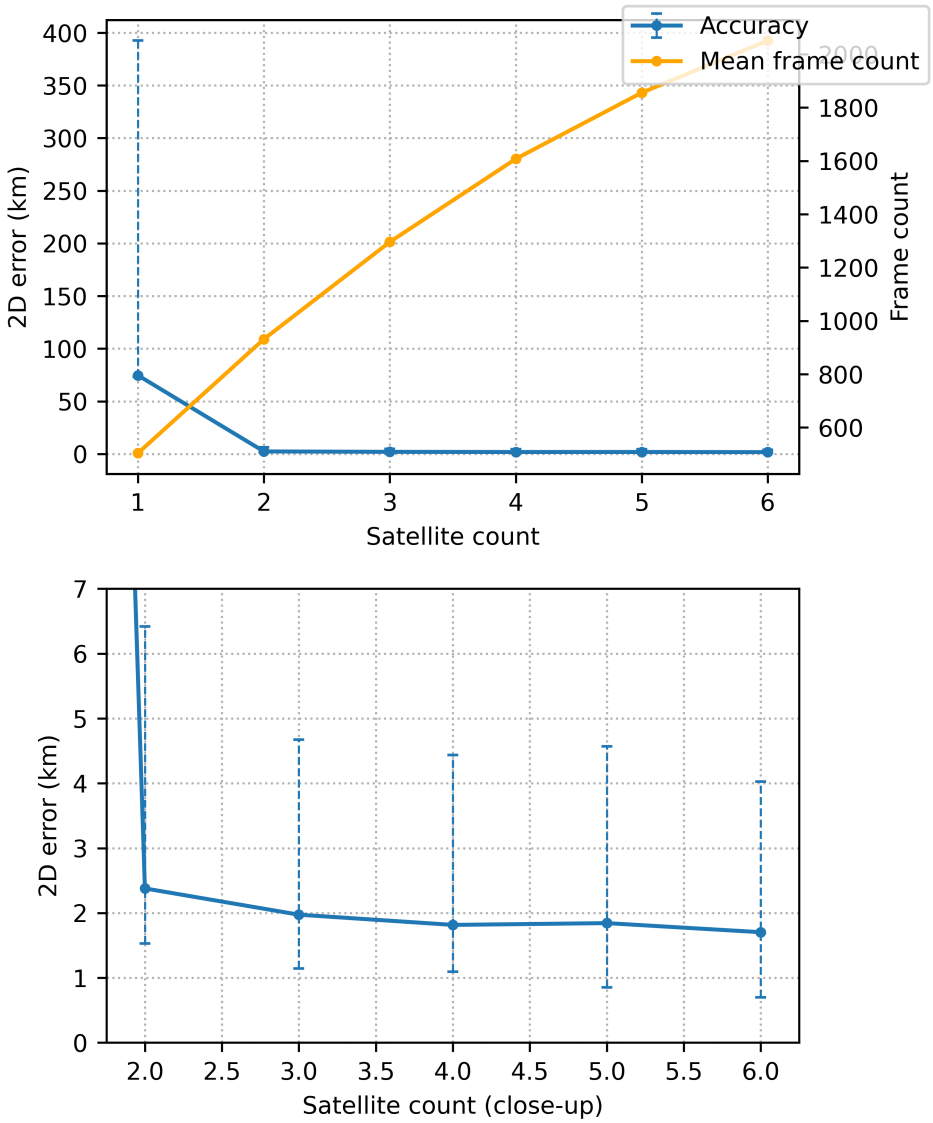


Figure 6.5: Accuracy as a function of number of received satellites

		Min	Mean	Max	Unit
Duration		00:39:51	01:06:45	01:48:49	hh:mm:ss
Data length		1017	2544	4420	frames
Frames per sat.	min	7	27	120	
	mean	102	215	493	
	max	245	481	903	
Detected sat.		6	13	18	
Satellite visibility time	min	00:00:51	00:01:49	00:03:39	
	mean	00:05:13	00:07:43	00:17:42	
	max	00:08:57	00:19:21	01:46:34	
TLE age	min	01:00:14	10:07:32	14:46:07	
	mean	02:21:15	16:56:11	27:36:32	
	max	03:22:10	24:23:58	40:09:30	
Satellite latitude	min	28.84	29.20	30.31	°
	mean	46.72	49.21	52.53	°
	max	17.38	29.22	33.00	°
Satellite longitude	min	-20.33	-12.76	-4.16	°
	mean	6.64	10.19	13.32	°
	max	63.80	64.48	66.72	°
Extent of satellite coverage	N	1471	1547	1796	km
	S	2252	2375	2414	km
	E	250	1086	1352	km
	W	1271	1873	2398	km
Frame count	IRA	1018	2599	4443	frames
	IBC	1018	2599	4443	frames
	All	43013	84489	159055	frames
Demodulator confidence	mean	96.7	97.0	97.3	%
Doppler shift	min	-35.15	-32.44	-29.52	kHz
	mean	-3.54	3.83	9.08	kHz
	max	34.10	36.55	39.73	kHz

Table 6.6: Data parameters (minimum, mean, maximum values)

Chapter 7
Conclusion

7.1 Summary

This thesis studied and demonstrated the Signals of Opportunity (SoP) positioning using the Doppler-shift method and LEO satellites, which is a way of determining user position by evaluating the Doppler shift of satellite signals not intended for navigation. The advantages of SoP positioning were high strength of the signals and their diversity in source, direction, frequency, timing and control. These strengths also presented several challenges, namely the unknown signal transmission time, less accurate and stable transmitter clocks, the inability to synchronise the receiver and transmitter time, as well as a general lack of documentation.

The SoP positioning systems reviewed in this work were based on the Doppler shift method, which is discussed in chapter 2. In chapter 3, the parameters and signals of some LEO satellite systems - namely Iridium NEXT, Orbcomm and Globalstar - were described. The constellations were made of 30 to 66 active satellites transmitting QPSK-modulated signals in the L, VHF and S band, respectively. A simulation confirmed that at least one satellite from each constellation was visible at almost all times.

Some of the current state-of-the-art SoP navigation systems were surveyed in chapter 4. Generally, the systems relied on purpose built receivers coupled with an EKF. Most of the systems were based on using the Iridium NEXT, with some using a combination of constellations. Both static and dynamic applications were reviewed. Signal fusion with INS, altimeter or magnetometer data was common. The accuracy of position determination was mostly in the order of 10^2 to 10^3 m, with a measurement time of 30 s to 30 min. In one case, the SoP positioning system provided a solution where GPS did not (due to foliage). In terms of accuracy, all of the reviewed systems were outclassed by GPS and other purpose-built navigation systems, however, the accuracy may have already sufficed for some applications. Additionally, some of the systems were as fast as a cold-start GPS. While the reviewed SoP systems did not compare evenly to GNSS, their performance highlighted the potential of SoP positioning.

The SoP positioning was demonstrated in this work by building a full-stack SoP positioning system, capable of capturing signals from Iridium NEXT

satellites, identifying the transmitting satellite and calculating its position, estimating an initial location without any external input, and calculating the user position. The system, described in chapter 5, was written in Python in a modular fashion, designed to be independent of the actual satellite system used to navigate, and to be used with generic SDR and antenna setup. Since it was a demonstrator, modularity and readability were key design principles, as this enables further research and expansion.

In chapter 6, the positioning system was shown to have a 2D accuracy of 1669 m and a precision of 1467 m (best 95 % of data). The time to achieve a sufficiently accurate position fix was 25 to 30 minutes, but a time of 10 minutes was possible. The system could reliably function with Doppler curves from as few as two satellites.

The navigation system developed in this work was less accurate than the SoP systems from the literature. This was partially due to the existing systems generally using more complex error estimation, especially the estimation of the receiver (and in some cases transmitter) frequency drift. The navigation system generally required more time to acquire a fix than the reviewed systems.

7.2 Future work

In the future, the capabilities of the navigation system can be extended, and the system was designed to accommodate such extensions. A custom Iridium receiver can be built, perhaps one optimised only for navigation, which discards or delegates signal demodulation and decoding in favour of more accurate frequency measurements, and which better compensates for offset and drift. The navigation system can also be extended to support additional constellations, such as Orbcomm or Globalstar or even Starlink. Other methods of the user position estimation may be developed, such as the EKF-based observation model method.

The major error sources in the navigation systems are the receiver and satellite frequency drift, the TLE-SGP4 prediction inaccuracies and the issues with determining exact frame reception time. Lesser error sources are the lack of compensation for signal travel time and ionospheric delays. These may also be addressed in the future, for example by developing a better offset and drift compensation or by using multi-frequency measurements to compensate for ionospheric delays. A custom receiver may also alleviate some of these issues.

The main contribution of the designed positioning system demonstrator is that it deals with all the steps of SoP positioning, highlighting the challenges which are not apparent from the literature, and providing a complete framework for further development. More broadly, this work provides a study of the process of SoP positioning and a survey of the current state-of-the-art. Hopefully, it will aid in future research of alternative positioning methods, which may address some of the many challenges of satellite navigation.

Appendix A

List of Abbreviations

API	Application Programming Interface
CDMA	Code Division Multiple Access
CEP	Circular Error Probable
CEST	Central European Summer Time
CFO	Carrier Frequency Offset
ECEF	Earth Centred Earth Fixed
ECI	Earth Centred Inertial
EIRP	Effective Isotropic Radiated Power
EKF	Extended Kalman Filter
FDMA	Frequency Division Multiple Access
FFT	Fast Fourrier Transform
FLL	Frequency Lock Loop
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning System
GSO	Geosynchronous Earth Orbit
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IQ sample	Complex sample or Quadrature sample
IRA	Iridium Ring Alert
ITRS	International Terrestrial Reference System
LEO	Low Earth Orbit
LNA	Low Noise Amplifier
MEO	Middle Earth Orbit
NAVSAT	Navy Navigation Satellite Systém
NCO	Numerically Controlled Oscillator
NORAD	North American Aerospace Defense Command

OG1	Orbcomm Generation 1
PDU	Protocol Data Unit
PLL	Phase Lock Loop
PSD	Power Spectral Density
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RF	Radio Frequency
RMSE	Root Mean Square Error
RTK	Real Time Kinematics
RX	Receiver, Receiving
SDMA	Space Division Multiple Access
SDR	Software Defined Radio
SGP4	Simplified General Perturbations 4
SK	Binary Phase Shift Keying
SNR	Signal to Noise Ratio
SW	Software
SoP	Signals of Opportunity
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TEME	True Equator Mean Equinox
TLE	Two Line Elements
TTF	Time To Fix
TX	Transmitter, Transmitting
UAV	Unmanned Aerial Vehicle
UHF	Ultra High Frequency
UTC	Universal Coordinated Time
VHF	Very High Frequency
VOR	VHF Omnidirectional Radio Range
WGS84	World Geodetic System 1984

Appendix B

Data and Code

The data collected as part of this thesis and the source code of the navigation system constitute the digital appendix to the thesis. The files were submitted along with the digital thesis document (16 zip files packed separately, each containing one dataset, and one zip file with the source code) and the files are available on GitHub (two zip files, one with experimental data and one with validation data, and one zip file with the source code).

The source code includes a *readme* file with installation and usage instructions.

On GitHub, the data and the source code can be downloaded from the Release page of the personal repository of the author:

`https://github.com/voskavoj/DP/releases/tag/submission`

The data are in several folders, each of which contains the following files:

- `output.bits` - these are the demodulated frames
- `decoded.txt` - these are the decoded frames
- `Iridium.txt` - these are the TLEs for Iridium valid at the time of recording
- `log.txt` - this is the log from the data capture
- `saved_nav_data.pickle` - these are the processed data ready for navigation calculations
- `test_nav_data.pickle` - these are simulated data used for algorithm tuning (not always present)
- `start_time.txt` - this file includes the timestamp of the recording start

Appendix C

List of Free and Open Source software used

Below is a list of Free and Open Source software used in the navigation system developed in this work, with a link to the respective project's website and a comment about its purpose within the navigation system. The author would like to thank the authors of all of the packages, and the Open Source community as a whole, for keeping their project free of charge and available. The list is sorted alphabetically.

- Basemap (`https://matplotlib.org/basemap/stable/api/basemap_api.html`) for plotting
- GNURadio (`https://www.gnuradio.org/`) for signal capture
- Python (`https://www.python.org/`) and the Python Standard Library as the primary programming language
- SoapySDR (`https://github.com/pothosware/SoapySDR/wiki`) for PlutoSDR support
- astropy (`https://www.astropy.org`) for frames of reference transformations and operations with time
- geopy (`https://geopy.readthedocs.io/en/stable/`) for latitude and longitude distance calculations
- gr-iridium (`https://github.com/muccc/gr-iridium`) for Iridium signal capture and demodulation
- iridium-toolkit (`https://github.com/muccc/iridium-toolkit`) for Iridium frames decoding
- matplotlib (`https://matplotlib.org/`) for plotting
- numpy (`https://numpy.org/`) for vector mathematics
- requests (`https://docs.python-requests.org/en/latest/index.html`) for downloading TLEs
- serial (`https://pyserial.readthedocs.io/en/latest/pyserial.html`) for communicating with PlutoSDR
- sgp4 (`https://pypi.org/project/sgp4/`) for SGP4 predictions

Appendix D

Bibliography

[1] Rémy Lopez, Jean-Pierre Malardé, François Royer, and Philippe Gaspar. Improving argos doppler location using multiple-model kalman filtering. *IEEE transactions on geoscience and remote sensing*, 52(8):4744–4755, 2013.

[2] David Vallado, Paul Crawford, Ricahrd Hujsak, and TS Kelso. Revisiting spacetrack report# 3, rev 3. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 6753, 2006.

[3] Zizhong Tan, Honglei Qin, Li Cong, and Chao Zhao. Positioning using iridium satellite signals of opportunity in weak signal environment. *Electronics*, 9(1):37, 2019.

[4] Analog Devices. *ADALM-PLUTO*, 2017. Datasheet. v1.2.

[5] Analog Devices. *AD9363 RF Agile Transceiver*, 2016. Datasheet. Rev. D.

[6] Sarah Reid. Orbcomm system overview. Technical Report A80TD0008 rev. G, Orbcomm LLC., Dulles, Virginia, USA, December 2001.

[7] Gabriele Oligeri, Savio Sciancalepore, and Roberto Di Pietro. Gnss spoofing detection via opportunistic iridium signals. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 42–52, 2020.

[8] Anna BO Jensen and Jean-Paul Sicard. Challenges for positioning and navigation in the arctic. *Coordinates*, 6(10):10–13, 2010.

[9] Gabriele Oligeri, Savio Sciancalepore, and Roberto Di Pietro. Physical-layer data of iridium satellites broadcast messages. *Data in Brief*, 46, 2023.

[10] Frank Stephen Tromp Van Diggelen. *A-gps: Assisted gps, gnss, and sbas*. Artech house, 2009.

[11] Joshua Morales, Joe Khalife, and Zaher M Kassas. Simultaneous tracking of orbcomm leo satellites and inertial navigation system aiding

using doppler measurements. In *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, pages 1–6. IEEE, 2019.

[12] Hamza Benzerrouk, Quang Nguyen, Fang Xiaoxing, Hamza Rasaee, Rene Jr Landry, et al. Leo satellites based doppler positioning using distributed nonlinear estimation. *IFAC-PapersOnLine*, 52(12):496–501, 2019.

[13] John Grady. Threats to merchant ships growing; mariners face pirates, lethal drones. `https://news.usni.org/2021/08/05/threats-to-merchant-ships-growing-mariners-face-pirates-lethal-drones`, Aug 2021.

[14] Thomas A. Stansell. The transit - navigation satellite system. Technical Report R-5933A, Magnavox, USA, June 1983.

[15] Chaoqun Yang, Bo Zang, Bowen Gu, Liang Zhang, Chuanjin Dai, Lulan Long, Zongchang Zhang, Linlin Ding, and Hongbing Ji. Doppler positioning of dynamic targets with unknown leo satellite signals. *Electronics*, 12(11):2392, 2023.

[16] Farzan Farhangian and René Landry Jr. Multi-constellation software-defined receiver for doppler positioning with leo satellites. *Sensors*, 20(20):5866, 2020.

[17] Nabil Jardak and Ronan Adam. Practical use of starlink downlink tones for positioning. *Sensors*, 23(6):3234, 2023.

[18] Nabil Jardak and Quentin Jault. The potential of leo satellite-based opportunistic navigation for high dynamic applications. *Sensors*, 22(7):2541, 2022.

[19] T. S. Kelso. [online] `https://celestrak.org/`, 1985.

[20] Mini-Circuits. *ZX60-3018G-S*, 2 2019. Datasheet. Rev. D.

[21] David A. Vallado and Wayne D. McClain. *Fundamentals of astrodynamics and applications*. Microcosm Press, 4 edition, 2013.

[22] Wei Dong and Zhao Chang-yin. An accuracy analysis of the sgp4/sdp4 model. *Chinese Astronomy and Astrophysics*, 34(1):69–76, 2010.

[23] Rod Sladen. Iridium constellation status. [online] `http://www.rod.sladen.org.uk/iridium.htm`, Jun 2020. In memoriam. Accessed on 02.10.2023.

[24] Robert J Kerczewski, Mike Meza, and Om Gupta. Application of the iridium satellite system to aeronautical communications. In *14th Ka and Broadband Communications Conference*, number E-16727 in series, 2008.

[25] Dan Veeneman. Iridium. [online] `http://www.decodesystems.com/iridium.html`, November 2021. Accessed on 05.10.2023.

[26] Hamza Benzerrouk, Quang Nguyen, Fang Xiaoxing, Abdessamad Amrhar, Alexander V Nebylov, and Rene Landry. Alternative pnt based on iridium next leo satellites doppler/ins integrated navigation system. In *2019 26th Saint Petersburg international conference on integrated navigation systems (ICINS)*, pages 1–10. IEEE, 2019.

[27] Zizhong Tan, Honglei Qin, Li Cong, and Chao Zhao. New method for positioning using iridium satellite signals of opportunity. *IEEE access*, 7:83412–83423, 2019.

[28] Frank Buntschuh. Iridium next engineering statement. Technical report, Iridium Communications Inc., McLean, Virginira, USA, December 2013.

[29] Donald H. Gips. FCC Order and Authorization (DA 96-1789). Technical Report DA 96-1789, Federal Communications Commission, Washington D.C., USA, October 1996.

[30] Orbcomm Inc. Orbcomm network data. [online] `https://www.orbcomm.com/resources/network-data`, November 2023. Accessed on 5.11.2023.

[31] Dan Veeneman. Orbcomm. [online] `http://www.decodesystems.com/orbcomm.html`, January 2019. Accessed on 5.11.2023.

[32] Nadav Levanon. Quick position determination using 1 or 2 leo satellites. *IEEE Transactions on Aerospace and electronic Systems*, 34(3):736–754, 1998.

[33] Edward Hirshfield. The globalstar system breakthroughs in efficiency in microwave and signal processing technology. In *Professional Program Proceedings. ELECTRO'96*, pages 147–152. IEEE, 1996.

[34] Globalstar Inc. Globalstar overview 2017. [online] `https://www.globalstar.com/Globalstar/media/Globalstar/Downloads/Spectrum/GlobalstarOverviewPresentation.pdf`, 2017. Accessed on 11.11.2023.

[35] Dan Veeneman. Globalstar. [online] `http://www.decodesystems.com/globalstar.html`, May 2018. Accessed on 7.11.2023.

[36] Collecte Localisation Satellites. Argos user's manual. Technical report, CLS, June 2016. [online] `https://www.argos-system.org/wp-content/uploads/2023/01/CLS-Argos-System-User-Manual.pdf`.

[37] Joe J Khalife and Zaher M Kassas. Receiver design for doppler positioning with leo satellites. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5506–5510. IEEE, 2019.

[38] Changsheng Cai, Yang Gao, Lin Pan, and Jianjun Zhu. Precise point positioning with quad-constellations: Gps, beidou, glonass and galileo. *Advances in space research*, 56(1):133–143, 2015.

[39] Novatel. *GPS/GLONASS/GALILEO receiver GNS 702*, 4 2019. Datasheet. v1.4.

[40] Novatel. *GPS-704-X*, 11 2015. Datasheet. v5.

[41] Schneider and Sec. gr-iridium. [online] `https://github.com/muccc/gr-iridium`, 2022. Version 1.0.0.

[42] Schneider and Stefan "Sec" Zehl. iridium-toolkit. [online] `https://github.com/muccc/iridium-toolkit`, 2024. Commit 47377de.

[43] Schneider and Stefan "Sec" Zehl. Iridium satellite hacking. HOPE XI 2016, Conference recording, [online] `https://www.youtube.com/watch?v=cvKaC4pNvck`, 2016.

[44] Michael Cerna and Audrey F. Harvey. *The Fundamentals of FFT-Based Signal Analysis and Measurement*. National Instruments, 2000. Application Note 041.

[45] Brandon Rhodes. python-sgp4. [online] `https://github.com/brandon-rhodes/python-sgp4`, 2024. Commit 8416d16.

[46] Astropy Collaboration. The Astropy Project: Sustaining and Growing a Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core Package. *The Astrophysical Journal*, 935(2):167, August 2022.