Technische Fakultät
Albert-Ludwigs-Universität, Freiburg

Lehrstuhl für Kommunikationsysteme
Prof. Dr. Gerhard Schneider

Master thesis

Mobile Assisted GPS Localization in GSM Networks

September 3, 2012

Refik Hadžialić

Supervised by
M.Sc. Konrad Meier
M.Sc. Dennis Wehrle

First Examiner
Prof. Dr. Gerhard Schneider

Second Examiner
Prof. Dr. Christian Schindelhauer

Erklärung

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen/Hilfsmittel verwendet habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, bereits für eine andere Prüfung angefertigt wurde.

Ort, Datum
(Place, Date)

Unterschrift
(Signature)

Acknowledgment

I would like to thank my supervisors Konrad Meier and Dennis Wehrle for their help and encouragement during the thesis. Besides the help from my supervisors I would like to thank my family and friends who supported me through my master's degree, and the entire Communication Systems department for their support, free coffee and to Prof. Dr. Gerhard Schneider for making all the required hardware available. I would like to thank Prof. Dr. Christian Schindelhauer for writing the recommendation letter so that the DAAD could extend my scholarship and stay in Freiburg. I would like to thank my friend Mirza Hamza from the Telecommunications department on the faculty of Electrical Engineering in Sarajevo for proofreading the AGPS chapter. I would like to thank Sebastian Schmelzer for his LaTeX tips, Michael Neves Pereira and Jonathan Bauer for lending me their cell phones to test my localization system as well as Johan Latocha for patiently explaining words I did not understand in the German language and for showing me Inkscape. I would like to thank Richard Zahoransky for the helpful discussions about various GSM topics. Thanks to Holger Hans Peter Freyther as well, who gave me tips on how to modify OpenBSC to make an independent data channel interface with a cell phone. Things which have not been done before are always intellectually seductive and this kept me motivated and working during the tough periods.

Abstract

Contents

1. Introduction

1.1. Motivation

Recent developments in the field of physics, chemistry and electronics have led to the cost-efficient manufacturing of diverse, compact single chip integrated solutions. As a consequence of this rapid development it became possible to integrate a GPS receiver into almost every cell phone without drastically increasing the price, physical size or weight of the cell phone. An example of this would be the GPS receiver chip inside the Apple iPhone 3GS. Its cost was estimated to be around $2.25 USD [9]. It is important to note that the number of wireless connections has increased as well; in 2011 there were 6 billion mobile connections worldwide [34]. In the following European countries, Germany, France, Spain, Italy and UK, 44% of all GSM users own a smart phone, whereas in the US and Canada this number is slightly higher, 46% [26]. By the statistics of the Blur group, 47% of all the cell phones on the world will be smart phones by 2015 [12].

An emerging new market of Location-Based Services (LBS) has resulted and since then the telecommunication and marketing industry have undergone rapid change. In 2009, 63 million users owned LBS-capable phones. This number is expected to grow in 2012 to 468 million users worldwide [12]. As social networks like Facebook, Twitter or Foursquare (a location-based social network) expand, it has become a trend for the users to share their location with their friends [12]. It has been reported that LBS represents a bonanza opportunity for new startup companies and global industry analysts project by 2015 a global market worth $21 USD billion (≈ €17.142 billion) [57]. New ideas and algorithms for tracking, navigation solutions, safety, security, searching for local business and payments shall emerge from the use of LBS technology [57]. LBS have already been used for tracking people with dementia and Alzheimer's disease, as reported in a study performed by researchers at the University of Siegen [60]. The Enhanced 911 (E911), an emergency service in the US for linking emergency callers with the appropriate service (police, firefighters and emergency room), is regulated by the US Federal Communication Commission (FCC), which sets the standards for all telecommunication providers, including the precision with which

callers' locations are tracked [78]. Europe's emergency service has similar standards [51]. Next generation networks, Long Term Evolution (LTE) 4G networks, have been designed from the start to have LBS capabilities integrated in the system and have better LBS performance as well as higher accuracy compared to GSM networks [75]. In the introductory chapter, some of the most common positioning techniques in wireless networks shall be presented and analysed, including Cell-ID, Time-of-Arrival, Angle-of-Arrival and GPS positioning. The author shall then describe the goals of his thesis. In this thesis the author shall provide the theoretical and practical knowledge required for building a localization system of mobile users inside of a 2G GSM network by taking advantage of the already-existing AGPS receivers inside of smart phones.

1.2. Positioning techniques

In this section, the current technologies for estimating the position of a mobile user shall be presented and their working principle. When the GSM network was designed, its primary goal was to enable wireless full duplex telephone service [40]. Over the past decade the GSM and its derivative networks became more popular and mature compared to the initial GSM standard, so the demands grew for new services such as Internet connectivity and LBS. Emergency services wanted to be able to localize mobile users in emergency situations like snow avalanches or other non-typical daily emergency situations [51]. This demand led to the devolopment of various approaches that differ in complexity and in the degree of accuracy of position fixes. However, the user positioning was limited by existing technology standards, and any improvement would require extremely expensive cost modifications to the existing network infrastructures. Several different ideas have been put forward to localize mobile users while avoiding these potential problems. They shall be presented in the following sections. First, however, it is important to distinguish between three different approaches to positioning mobile users, handset-based, network-based, and hybrid-based. With handset-based techniques, the handset itself tries to estimate its position on its own using the available information. In the network-based approach, the network makes all the required measurements and the handset itself is passive. The last, hybrid-based, approach uses resources from the handset and network together; both are active participants in the position estimation process. For the purposes of this thesis, the term "Mobile Station" will be used to designate the user one intends to locate. A few different methods, varying by their complexity and precision, shall be presented, in order of their complexity.

1.2.1. Cell-ID

The cell-identification method is the simplest known GSM positioning method [53, Chapter 8]. By knowing the geographical location of the Base Transceiver Station (BTS), one can roughly estimate the position of the Mobile Station (MS) [30, Chapter 4]. It is important to build maps where the BTS signal can be received and where the border *handover* points are located. Handover is the process of switching from one BTS to another where the signal reception strength is higher than on the currently-connected BTS. The basic principle is shown in figure 1.1. The BTS are divided into geographical regions[1] by their signal coverage. The MS is in the region of the currently-connected BTS and it could be at any point inside of the hexagon. Every

[1] Usually they are represented as hexagons but it could take any other geometric shape.

BTS has a unique identifier code name and hence can be distinguished from other BTS's.

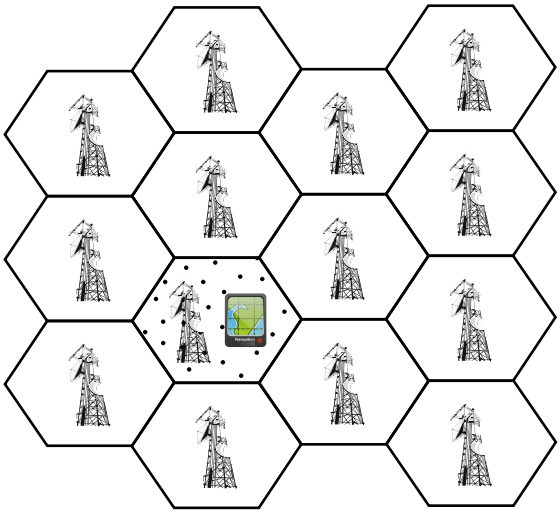

Figure 1.1.: Cell-ID position estimation technique where a mobile user can be connected to only one BTS.

Using this method even higher accuracies can be achieved than the known shape of signal reception [53, Chapter 8], provided that the *timing advance* (TA) value is known. The TA is the rough prediction of the *round trip time* (RTT), time required for a data packet to be received and acknowledged by the MS. Using this measure a rough circle can be made between the BTS and the bordering points of the Cell-ID region since TA multiplied by the speed of light produces the radius distance of the circle. To obtain the TA value a connection between the MS and the BTS has to exist or a silent call can be made where the GSM subscriber does not even notice that he/she is being called since there is no ringing or any other sign that an idle connection is being performed on the MS [3, Chapter 4]. If there are more antennas than one, then the MS location can be even more precisely specified. This can still be inaccurate, however, because of multipath signal reflections. In urban environments it is usually the case that there is no optical line of sight between the BTS and MS, so while the signal propagates from the BTS to the MS and vice versa it may be reflected by buildings or other objects which add extra propagation time (extra range to the distance). The accuracy of this method is typically in a range of 200 m [85]. This method can be seen both as a handset- and network-based position estimation technique, due to the fact that the user may run his/her own application on the cell phone or it can be applied by the network operator himself. This estimation technique does not require the MS to be a smart phone; it works with any type of cell phone.

1.2.2. Received Signal Strength

The Received Signal Strength (RSS) position estimation technique, as the name states, uses the signal strength measurement reports to localize the MS. RSS measurement reports in GSM networks are transmitted from the MS to the BTS and they are used to determine if the handover process should be triggered or not [84].

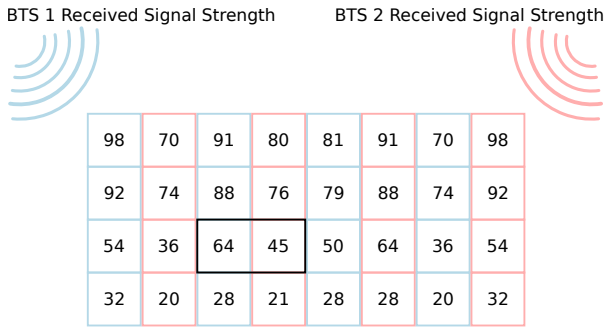

Figure 1.2.: Basic idea of the RSS estimation technique. One rectangle location is represented by two RSS measurements for two BTS, blue indicates BTS1 and red indicates BTS2.

This method requires mapping the location blocks of the covered areas with RSS before it can be used [84]. The basic idea can be seen in figure 1.2, where one location region is represented by two differently colored (blue and red) RSS measurements inside of one rectangle. By knowing the RSS in advance one can estimate the probability that the MS is located in the black rectangle. Since it is difficult to measure the RSS at every point, the map is interpolated with RSS using the Voronoin interpolation to calculate the expected RSS at places were the measurement have not been conducted [84]. In the next step the probability distribution, the Bayes Theorem, is utilised to estimate the location of the MS by computing probabilities for all the points in the map. The precision of this method is limited to the amount of BTS in range and the manufacturer design of the RF front end in the cell phone. However, this method can be applied on any cell phone and does not require a smart phone. It is a network based estimation technique.

1.2.3. E-OTD and UL-TDOA

E-OTD and UL-TDOA are two similar positioning techniques; both use the time difference of signal arrival and for this reason have been grouped together. E-OTD stands for Enhanced Observed Time Difference. This technique requires the GSM

network to be clock-synchronized. The clock synchronization of the GSM network can be achieved with a Location Measurement Unit (LMU) [28]. LMU's provide the precise time to the BTS's by having an atomic clock synchronized with the BTS on a seperate location from the BTS or by providing a special GPS device at the BTS' location that can provide the precise time [28]. The clock synchronization of the MS and the BTS is required because the E-OTD technique takes advantage of measuring signal propagation time. A data signal with precise up-to-date time information is transmitted from three or more spatially distinct BTS's at the same time and then propagation time is measured on the MS (all these BTS's must be detecable by the MS itself) [59]. Once the difference in time is known between when the signal was transmitted and when it was received, it is easy to estimate the relative position to the BTS's with hyperbolic trilateration [59] [3, Chapter 4]. In order to estimate the absolute position, one must first know the absolute location of the BTS's. The basic idea can be seen in figure 1.3. E-OTD requires the cell phone to be equiped

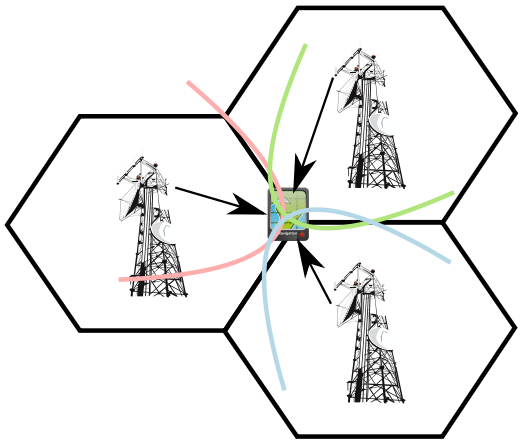

Figure 1.3.: Basic idea of the E-OTD positioning technique. Current time information is transmitted from 3 different BTS's at the same time. Then the MS observes the difference of time when the information arrive and using trilateration technique calculates the relative position of the MS.

with firmware to perform these measurements but does not require new or external hardware. The accuracy of this method lies in the range between 50-200 m, depending on the location of the MS [55]. This method is can still be susceptible to the multipath signal problem, however. E-OTD is a handset-based position estimation technique.

UL-TDOA (Up-Link Time Difference of Arrival) is a similar localization technique as E-OTD [55]. The basic difference between UL-TDOA and E-OTD is that the signal propagation time is observed on the BTS's and not on the MS itself. To estimate the position of the MS, the BTS responsible for the MS forces the MS to request a

handover to two or more BTS's nearby. The MS sends a handover burst signal and the neighboring BTS's measure the waiting time between the handover request signal itself and the transmitted burst from the MS. Using the observed time difference, the BTS's can compute the location of the MS. It is important to note that this position estimation technique takes place while there is an active call on the MS or the BTS makes a silent call to the MS where the mobile user is not aware of being tracked [55]. This technique is slightly less accure than E-OTD; the accuracy lies between 50-300 m [59]. The unsynchronized operation of the GSM network makes these two techniques impossible without clock synchronization. One microsecond error would produce an error of around 300 m. The advantage of UL-TDOA over E-OTD lies in the fact that no extra software modifications have to be made to the cell phone and this technique works on every cell phone. UL-TDOA is a network-based position estimation technique.

1.2.4. Assisted-GPS

Another positioning technique is Assisted-GPS (AGPS). It has recently gained popularity because of the great number of smart phones with an embedded AGPS receiver and the introduction of 3G/4G networks. These networks are clock-synchronized since high-bandwidth wireless services require synchronous working operation [23] [68]. AGPS receivers can decrease the waiting time required to estimate the position if the "exact" time is known [21, Chapter 4]. It works by exploiting the existing navigation satellite network. In the event where mobile users are in urban environments, the GPS satellite signals are blocked by the buildings. Further analyses showed that the received signals arrive at the cell phone with errors because of multiple propagation reflection and are often hardly distinguishable from noise [21, Chapter 2]. The power of received signals on a GPS receiver is in the range of 100 attowatts[2] when the GPS receiver is outdoors. The signal strength becomes even smaller by a factor of 10-1000 if the user is between tall buildings or indoors [21, Chapter 2]. All these factors affect the acquisition of GPS signals and make the correct reception of GPS signals unrealisable and impractical. Instead of searching manually for the GPS satellites and waiting for the orbiting parameters to arrive from the satellites, which are required to estimate the position, information about the orbiting GPS satellites is transmitted over an existing GSM network infrastructure. This provides the AGPS receiver additional data to track weak signals. The theoretical foundation of how GPS and AGPS receivers estimate the position is addressed in more detail in chapter 2. This method does not work on every cell phone as do the aforementioned

[2] 1 attowatt = $10^{-16}W$. The reception quality depends on the receiver's antenna and RF front-end design as well.

methods. It requires the cell phones to be equiped with an AGPS receiver. From this point on, cell phones with an AGPS receiver shall be refered to as smart phones since they have another potential use aside from the default communication application. The AGPS position estimation technique is a hybrid-based technique because the position is estimated with the help of the handset, that estimates the position, and the network provider since it delivers the required data for faster acquisition time.

1.2.5. Other techniques

The previously-mentioned localization techniques are not the only existing methods but are the standardized ones. In this section, two more techniques shall be briefly described: Angle-of-Arrival and Google Maps' WiFi tagging.

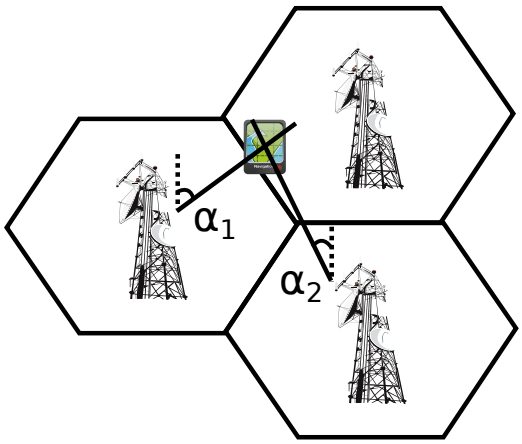

Figure 1.4.: Basic idea of the Angle-of-Arrival positioning technique. The angle of the reception signal on the BTS antenna is measured. By knowing at least two angles on two BTS's, it is possible to interpolate the intersection point where the MS is located.

Angle-of-Arrival (AOA) is a localization technique that exploits a geometric fact that by knowing at least two angles from two known points, i.e. BTS's, it is possible to construct the third triangle point (intersection point). The intersection point represents the location of the MS. The angle is derived by a burst signal transmitted from the MS and the time difference of arrival for different elements of the burst signal. Once the angle is computed, it is straightforward to find the intersection point. This technique requires the BTS's to be synchronized with LMU's and to be in line of sight with the BTS's, otherwise this method shall develiver poor position results. It belongs to the group of network based position estimation techniques.

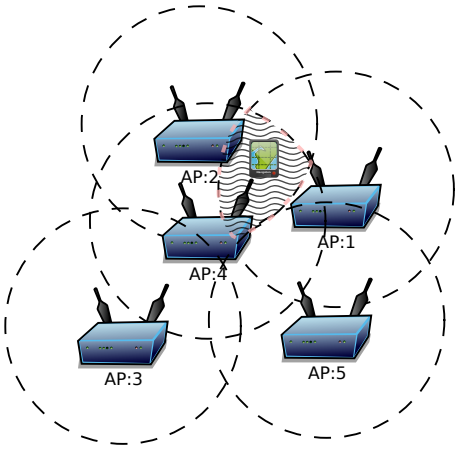

Figure 1.5.: Wireless Access Point tagging. The MS could be located anywhere where all three access points are visible, this area has a wavy background and is between access points 1, 2 and 4.

Another technique gaining in popularity is used by Google Maps to identify the position of the user by simply tagging an area with all visible wireless access points [31]. Since each access point has a unique MAC address it is not hard to identify them while driving through urban areas with a WiFi scanning device. The basic idea is depicted in figure 1.5, where the MS in this particular example is located where access points 1, 2 and 4 are visible at the same time stamp. This technique works efficiently indoors as well as outdoors in cities since ranges of 801.11 b/g wireless networks are not more than 30-150 m, though the new standard 801.11 n has a wider coverage area. A simple overview of all the discussed techniques is given in table 1.1.

Table 1.1.: Overview of the localization techniques.

Method	Sync.	Advantage&Disadvantage	Accuracy	Type
Cell-ID	No	Works on any cell phone, Imprecise	Anywhere in cell	Network
Cell-ID + TA	No	Works on any cell phone, Imprecise but better than Cell-ID alone	Anywhere in cell but with a radius	Network
RSS	No	Works on any cell phone, Depends on cell phone model and environment	≈ 300 m	Network
E-OTD	Yes	Works on most new cell phone models, Expensive because LMU	$\approx 50 - 200$ m	Handset
UL-TDOA	Yes	Works on any cell phone Expensive because LMU	$\approx 50 - 300$ m	Network
AGPS	Yes/No	Works on some cell phones with AGPS receivers, Very precise	$\approx 5 - 20$ m	Hybrid
AOA	Yes	Works on any cell phone, Expensive because LMU	Depends if MS is in line of sight	Network
Google maps with WiFi	No	Requires a smart phone with Google maps and Wireless 801.11 b/g/n, Does not work outside of cities or missing & unknown WiFi signal	$\approx 5 - 30$ m	Handset with aid of Network

1.3. Goals of the thesis

In this thesis the author shall provide theoretical and practical background knowledge required for building a localization system of mobile users inside of a 2G GSM network by taking the advantage of AGPS receivers inside of smart phones.

The thesis is divided into three parts. The first is a theoretical introduction to GSM and GPS systems as well as the protocol required for the positioning of mobile users. The second part provides more details on the software implementation and the hardware used in this work. The last section is a discussion of the achieved results in the test environment and the author's conclusions.

Chapters 2 and 3 will provide a theoretical introduction of GPS and AGPS receivers as well as of the GSM operational principles for understanding the basic functioning principles of the entire positioning system. The theoretical concepts of GPS receivers will be analysed and discussed in depth since they provide evidence for the advantages and limitations of this method. These two chapters will provide an explanation for the achieved and observed results in this thesis. Once the GPS and GSM working principles have been explained, the author shall proceed with introducing the reader to the Radio Resource Location Protocol (RRLP), responsible for transmission of assistance data and obtaining the position of the mobile user. More details on RRLP will be provided in chapter 4.

In chapter 5, the reader will be introduced to the software development and implementation process. More details on the hardware connections and set up shall be provided in chapter 6.

In chapter 7 test results and the test environment will be presented. Chapter 8 will provide a summary of the entire system.

The appendix contains details for configuring the entire system and for obtaining the same results. This thesis includes a USB stick with the source code developed during the work on this thesis.

2. GPS & Assisted-GPS

What use is knowledge if there is no understanding?

(Stobaeus)

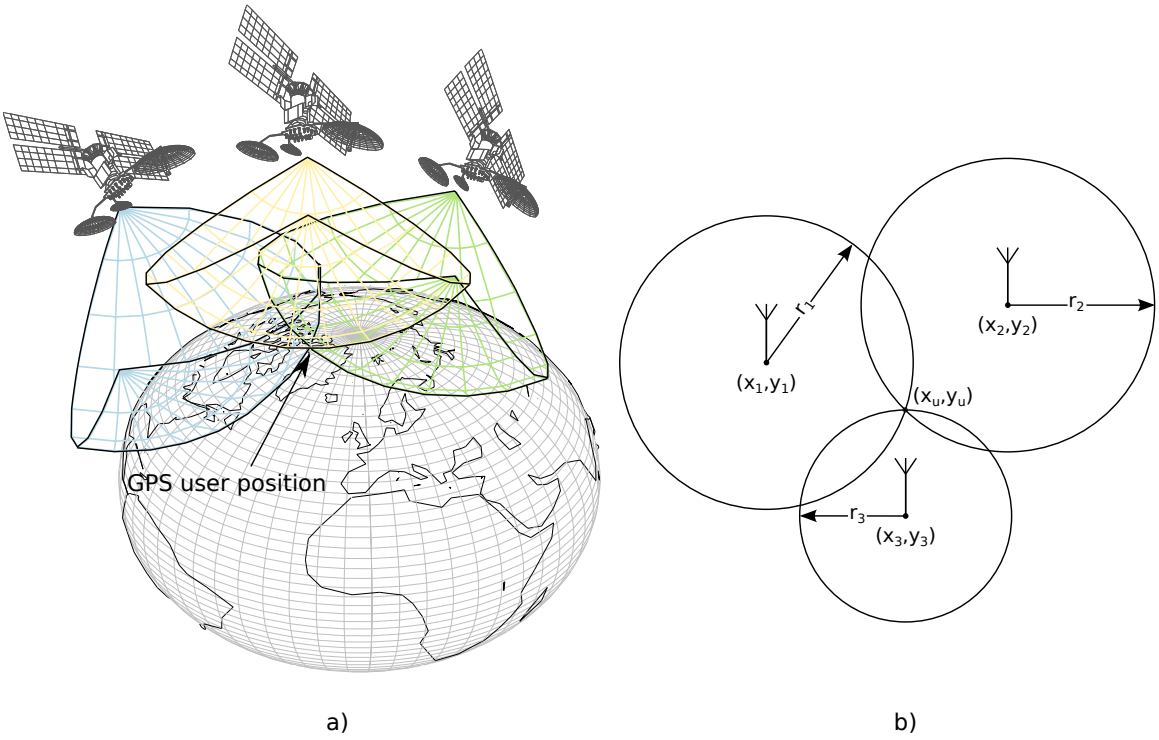

Figure 2.1.: GPS Simple working principle, a) example in 3D space with spheres b) example in 2D space with circles.

In the new global economy age, GPS positioning has become of important value for various services and businesses. It has been growing at a rate of 30% in the past few years and the application market is expected to be worth €240 milliard by 2020 only in Europe [32]. The goal of this chapter is to bring more details and insights of how GPS receivers work. The chapter is divided in few sections that explain how the data are modulated before transmission, demodulated on the receiver, how the

search space works, how the target user position is estimated and the errors that can influence the overall working of the system.

In this paragraph the general idea shall be given of how GPS works and how the position is estimated. Before all the details are revealed in the following sections, it is important to understand the basic principle of GPS navigation. GPS positioning works by using the principle of *trilateration*. Distances from the satellites to the GPS receiver are measured and from these distances receiver's position is estimated. The distances are estimated by measuring the signal propagation time between the satellites and the receiver, this position estimation technique is also known as time-of-arrival (TOA) method. Once sufficient amount of measurements from different satellites were generated, the position of the receiver can be approximated. It is important to understand that the positions from the satellites need to be known and same location reference system has to be used. The general principle of this idea can be seen in figure 2.1, picture (a) represents the idea with spheres in 3D space and picture (b) the same idea but in 2D space. The blue, yellow and green wireframes below the GPS satellites represent the spheres for a given range, between the satellite and the estimated position of the GPS user for the given satellite. By intersecting all the three spheres, the position of the user is estimated. In the next sections this general idea shall be developed in more details, in an step by step approach, and the ideas shall be verified using the appropriate mathematical models.

2.1. GPS data and signal modulation

The aim of this section is to give the reader an overview of the transmitted GPS data and to understand what type of processing takes place on the GPS satellite itself. As discussed in the paragraph earlier, to estimate the position of the GPS receiver, it is important to know the position of the satellite at the moment of signal transmission. Prior to releasing the data in the athmosphere, they need to be modulated in order for the GPS receiver to receive and demodulate them.

Each one of the GPS satellites transmits the same type of information. The transmitted data are called *frames* [13]. One frame of data can be seen in figure 2.2. Every of the 25 transmitted frames can be divided into five subframes of 300 bits length [21]. The data in the frames are called *navigation data* because using them the GPS receiver can estimate user's position. Each subframe can be divided into

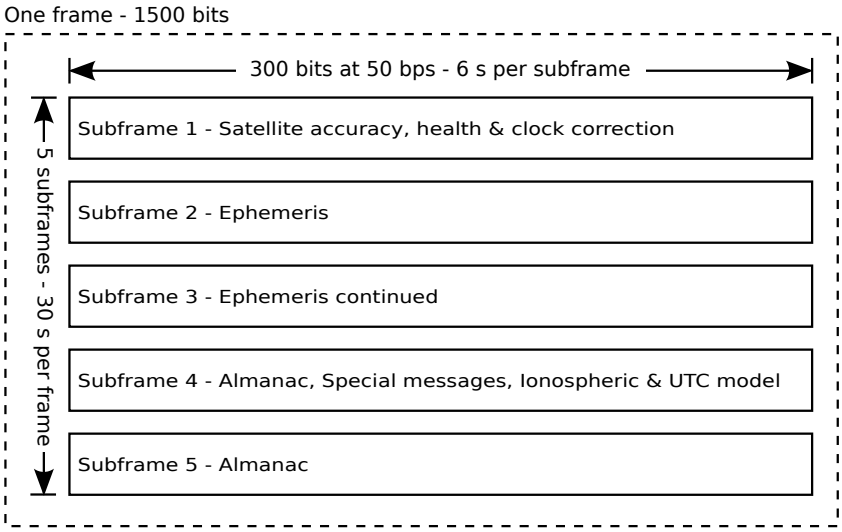

Figure 2.2.: One frame of 1500 bits on L1 frequency carrier. Image courtesy of [37].

three fields of data, as shown in figure 2.3, telemetry (TLM), handover (HOW) word and rest of the data (navigation data). TLM is the first word of the subframe and consists of a unique preamble used to synchronize and identify the subframes [13]. HOW is the second word of the subframe and consists of the *GPS system time* and subframe ID [13]. GPS system time is the time the atomic clock on the satellite generates at the moment of subframe broadcast and it acts as a time stamp [2]. The third segment of the subframe, indicated as rest of data in figure 2.3, consists of the navigation data. The first subframe includes data about the satellite accuracy and health as well as parameters used for the clock corrections on the receiver side. More

details on these parameters shall be given in section 2.2. Subframe two and three are made of *ephemeris data*. Ephemeris information are precise parameters for predicting the precise orbital position of the GPS satellite. Using ephemeris data for the specific system time stamp and the equations given in appendix section C the GPS receiver can precisely estimate the position (x_s, y_s, z_s) of the satellite. The first three subframes are satellite dependent and do not change in the transmitted 25 frames aside from the system time stamp [1]. Fourth and fifth subframes include *almanac*

Figure 2.3.: Subframes always start with telemetry and handover words

data, low-precision clock corrections, ionospheric model and UTC time calculation parameters. Almanac information are rough coarse parameters for predicting the orbital position of the GPS satellites. These low-precision parameters are used by the GPS receiver to estimate the rough position of the GPS satellites and to reduce the searching space for the GPS satellite transmission frequencies[1] and obtaining the precise ephemeris data. Ionospheric model and UTC time calculation parameters are required by the GPS receiver to refine the calculation of delays through the ionosphere [13]. The reason why there are 25 frames is because of the last two subframes, four and five. Subframes four and five have data which cycle through the 25 frames, i.e. almanac data are transmitted for all the 32 GPS satellites[2] in case the receiver found only one satellite and once it collected all almanac data, it can search for other visible GPS satellites. These 25 frames create a masterframe. Once the 25 frames have been transmitted, the process is repeated again.

The data are modulated using the binary phase shift keying (BPSK) technique. The newly modulated signal is the *L1* signal and it is emitted from the satellite directed antennas towards Earth [1]. The BPSK technique works by changing the phase of the carrier signal for 180° at the moment of bit toggle (flipping) in the data [1] [13]. Basic principle of this technique can be seen in figure 2.4. The carrier wave for GPS BPSK modulation is centered at a frequency of 1575.42 MHz [13]. These signals travel an average distance of 20200 km from the satellite to the GPS receiver and are affected by various sources of noise. BPSK modulation is mostly used for satellite links because of its simplicity and immunity to noise and signal intereference for the price of transfering data at low speed rates [41, Chapter 1]. The demodulation process of L1 shall be discussed and analysed seperately in section 2.2.1.

[1] Although all satellites transmit on the same one frequency, when the signals are received on Earth, they have a different frequency from the transmitted one. This shall be further explained in more details in the following sections 2.2.1, 2.2.2 and 2.2.3.

[2] 24 satellites are used in the GPS system, the rest is used in case one of the 24 fails.

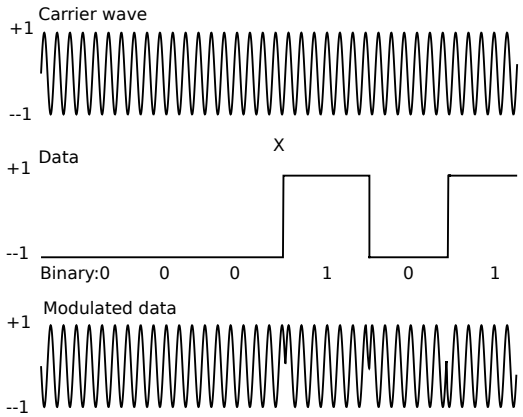

Figure 2.4.: BPSK Modulation - First signal is the carrier wave, and it is multiplied (mixed) with the second signal, which are the data to be transmitted. The resulting signal at the output of the satellite antenna is the third one.

However, before the raw navigation data enter the BPSK modulation process, they are XORed with pseudo random noise (PRN) sequences for different satellites (each satellite owns a unique PRN sequence) [13]. PRN sequences are used to identify which satellite signal is being decoded, transmission of the data on the same frequency as well as to enable the distance measuring mechanism between the satellite and the GPS receiver. Equivalent PRN sequence is generated on the GPS receiver and it is compared with the received PRN sequence which is delayed (shifted) due to the distance. This delay multiplied with the speed of light yields the distance between the satellite and the GPS receiver. PRN sequences have similar autocorrelation properties as noise, when it is shifted in time domain it has a low correlation value whereas when it is matched with exact image of itself it produces a high correlation peak [10, Chapter 3]. This property is used for identifying the satellites and for finding the exact phase shift. The second important property of PRN sequences is the property of orthogonality. This proberty enables the reception of different data on the same frequency, also known as code division multiple access (CDMA). It is important to note that the PRN sequences must have a higher frequency than the data, i.e. the bit duration of a PRN sequence is much shorter than of the data [10, Chapter 3]. Single bits in PRN sequences are called *chips* and the complete sequence as *code* [10, Chapter 3]. This newly generated signal is called direct sequence spread spectrum (DSSS) [10, Chapter 3]. In GPS terminology it is named as Code/Acquisition (C/A) code. C/A code is feed into the BPSK modulation process, where it is mixed with the carrier wave and producing the L1 signal. More details shall be given in the C/A demodulation section 2.2.2. Transmission speed of the navigation message is 50 bps, therefore the reception of a complete masterframe requires around $\approx$ 12.5 minutes,

i.e. $(1500\,\text{bitsperframe} \cdot 25\,\text{frames})/(50\,\text{bps} \cdot 60\,\text{s})$.

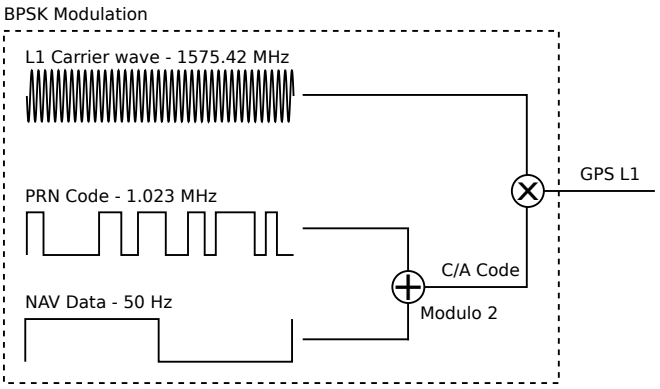

Figure 2.5.: Modulation of the GPS signal L1. Image courtesy of [37].

The described GPS navigation data modulation can be seen in figure 2.5 and it can be represented in form of equation (2.1) [35], where $D(t)$ are the navigation data at the moment t, $C(t)$ is the PRN chip at the moment t, $cos(2\pi f_c + \varphi_{SV})$ is the generated carrier wave with frequency f_c and phase φ_{GPS}, P is output power of the transmitter amplifier.

$$S(t) = PD(t)C(t)cos(2\pi f_c + \varphi_{GPS}) \tag{2.1}$$

The equation 2.1 shall be rewritten as given in 2.2. It represents the same equation but at the GPS receiver after traveling $\approx 20200\,km$, where $d_{C/A}$ is the C/A data and $n(t)$ is the random noise at moment t influenced by various factors that influence electromagnetic waves.

$$S(t) = \sqrt{\frac{P}{2}} d_{C/A} cos(2\pi f_c + \varphi_{GPS}) + n(t) \tag{2.2}$$

The GPS satellites are positioned in orbits so that at every moment at any spot on Earth, at least four satellites are visible (a spot can be considered as a mountain peak since in the cities GPS signals are blocked by buildings). In the next section, more details shall be revealed on the process of demodulating the GPS L1 signal and acquiring the correct time and position.

2.2. GPS signal acquisition and demodulation

GPS satellites[3] orbiting our planet, at a distance of approximately $20200\,km$, are equiped with precise atomic clocks [21, Chapter 2.7]. These atomic clocks are calibrated and maintained on a daily basis by the U.S. Air Force [33]. The time the atomic clock generate, refered earlier as GPS system time, denoted as t_{SV}, is generated as a time stamp at the moment of the subframe broadcast [2]. In addition to the broadcast time, subframe 1 contains parameters to account for the deterministic clock errors embedded in the broadcasted GPS system time stamp. These errors can be characterized as bias, drift and aging errors [2]. The correct broadcast time, denoted as t, can be estimated using the model given in equation (2.3) [2]. In equation (2.4), where the GPS receiver is required to calculate the satellite clock offset, denoted as Δt_{SV}, a number of unknown terms can be seen. These terms are encapsulated inside of the transmitted frames. The polynomial coefficients: a_{f0} - *clock offset*, a_{f1} - *fractional frequency offset*, a_{f2} - *fractional frequency drift*; and t_{0c} - *reference epoch* are encapsulated inside of subframe 1. The only remaining unknown term left in equation (2.4) is the *relativistic correction term*, denoted as Δt_r. Δt_r can be evaluated by applying the equation given in (2.5). F is a constant calculated from the given parameters in (C.0.7) and (C.0.8), whereas e, $\sqrt{A}$ and E_k are orbit parameters encapsulated in subframes 2 and 3 [2].

$$t = t_{SV} - \Delta t_{SV} \tag{2.3}$$

$$\Delta t_{SV} = a_{f0} + a_{f1}(t_{SV} - t_{oc}) + a_{f2}(t_{SV} - t_{oc})^2 + \Delta t_r \tag{2.4}$$

$$\Delta t_r = F e \sqrt{A} \sin E_k \tag{2.5}$$

$$F = \frac{-2\sqrt{\mu_e}}{c^2} = -4.442807633 \cdot 10^{-10} \frac{s}{\sqrt{m}} \tag{2.6}$$

Nevertheless, the broadcast satellite time information is not sufficient to estimate the precise time at the moment of the signal arrival. Even though the signal arrives in approximately[4] $77\,ms$, the precision of the atomic clock is in the range of 10 ns [21, Chapter 2]. Undoubtedly the signal propagation (travel) time, denoted as t_{prop}, has to be taken into account. In that case, the exact time at the moment of arrival is known, denoted as t_{exact} and is given in equation (2.7). Propagation time is computed by measuring the phase shift of the C/A signal, more details shall be

[3] Ssatellites are named as space vehicles in GPS terminology and the abrevation SV is used in the equation notations to denote a parameter related to the satellite itself.

[4] Propagation time depends on user and GPS satellite position.

given in sections 2.2.2 and 2.3. More importantly, t_{exact} time shall be later used to synchronize various time dependent systems like the GSM, LTE, GNSS or other communication and ranging systems.

$$t_{exact} = t_{prop} + t \tag{2.7}$$

2.2.1. Carrier wave demodulation

In order to calculate the signal propagation time between the satellite and the receiver, the internal sine wave synthesizer inside of the receiver has to be synchronized with the carrier sine wave generator of the GPS satellite [70]. In other words, the identical carrier wave replica has to be generated on the receiver as on the satellite [14]. However, the received signal is not the equivalent of the transmitted signal. Due to the nature of the Doppler effect[5] and wave propagation, the transmitted signal arrives phase disordered at the receiver [70]. This phase disorder is a consequence of the relationship between the instantaneous frequency and instantaneous phase according to equations (2.8) and (2.9).

$$f(t) = \frac{1}{2\pi}\frac{\partial}{\partial t}\phi(t) \tag{2.8}$$

$$\phi(t) = 2\pi \int_{-\infty}^{t} f(\tau)d\tau \tag{2.9}$$

Considering that the GPS satellites orbit the Earth with a speed of around $3.9\,km/s$, the Earth rotates around its axis and the target user with the GPS receiver may move as well, the Doppler effect is unavoidable. The observed phase at the receiver antenna, denoted as φ_o, can be described using the equation given in (2.10), where φ_{GPS} represents the known satellite carrier wave phase, $\delta\varphi_{SV}$ the clock instabilities on the GPS satellite, φ_a the phase shift error caused by propagation delays in the ionosphere and troposphere respectively, $\delta\varphi_{DE}$ the phase shift caused by the Doppler effect and $\delta\varphi_w$ is the wideband noise phase shift.

$$\varphi_o = \varphi_{GPS} + \delta\varphi_{SV} + \varphi_a + \delta\varphi_{DE} + \delta\varphi_w \tag{2.10}$$

The task of the demodulation process is to generate a replica carrier wave with the matching phase shift and mix it with the incoming signal. In the ideal case the

[5] Doppler effect is a phenomenon that happens as a result of relative motion of the two bodies, transmitter and receiver, towards or away from each other and causes frequency shift of the electromagnetic wave [83, Chapter 4].

observed phase on the antenna and the generated phase on the receiver, denoted as φ_{rec}, cancel each other out, that is to say, equation (2.11) equals zero.

$$\Delta\varphi = \varphi_o - \varphi_{rec} \tag{2.11}$$

The circuit responsible for generating the same carrier wave is the phase locked loop (PLL). The PLL circuit is a feedback loop that modifies the synthesized wave parameters such that $\Delta\varphi \approx 0$, a phase shift is shown in figure 2.6.

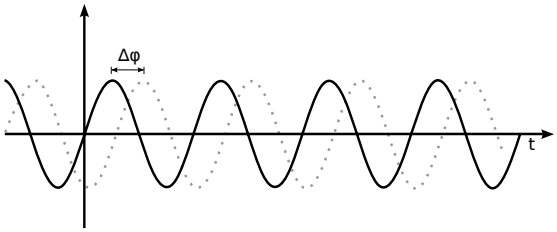

Figure 2.6.: Two equivalent carrier waves with the same frequency but different phase shift

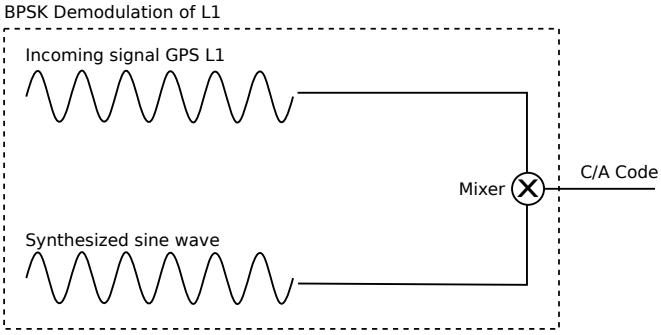

Figure 2.7.: Demodulation of the L1 GPS signal

The reason for this is straightforward to understand by looking at the multiplication of two sine waves. The GPS L1 signal demodulator at the receiver is depicted in figure 2.7, the incoming signal L1 is multiplied with the synthesized sine wave (multiplication is the function of a mixer, denoted as $\otimes$ in figure 2.7). For the purpose of easier analysis, cosine waves shall be used istead of sine waves, the difference between them is only in the phase shift, as denoted in equation (2.12).

$$\sin(\pm x) = \cos\left(\frac{\pi}{2} \pm x\right) \tag{2.12}$$

Multiplication of two cosine waves, as in equation (2.13), can be derived by adding $\cos(A + B)$ and $\cos(A - B)$ together, as respectively given in equations (2.14) and

(2.15).

$$\cos(A)\cdot\cos(B) = \frac{1}{2}\cos(A-B) + \frac{1}{2}\cos(A+B) \tag{2.13}$$

$$\cos(A+B) = \cos(A)\cos(B) - \sin(A)\sin(B) \tag{2.14}$$

$$\cos(A-B) = \cos(A)\cos(B) + \sin(A)\sin(B) \tag{2.15}$$

The incoming GPS L1 signal with a frequency f_1, given in figure 2.7, can be written as $d_{C/A}\cos(\omega_1 t)$, a similar form is given in equation (2.2), where $\omega_1 = 2\pi f_1$ is the angle frequency and $d_{C/A}$ is the C/A data (navigation message modulated with the PRN code), $d_{C/A} = d_{PRN} \oplus d_{NAV}$. If equation (2.13) is rewritten with the received GPS signal L1 and synthesized wave with frequency f_2, the equation results the one given in (2.16)

$$d_{C/A}\cdot\cos(\omega_1 t)\cos(\omega_2 t) = \frac{1}{2}d_{C/A}\cdot\cos(\omega_1 t-\omega_2 t) + \frac{1}{2}d_{C/A}\cos(\omega_1 t+\omega_2 t) \tag{2.16}$$

This leaves the resulting signal with two frequency terms, a low frequency term $(\omega_1 t - \omega_2 t)$ and a high frequency term $(\omega_1 t + \omega_2 t)$, the t can be taken in front of the bracket as it is a common multiplier. The high frequency term, $(\omega_1 + \omega_2)$, can be filtered out using a low-pass filter[6]. Ideally, the difference of the angle frequencies is zero, as in equation (2.17), since $\cos(\Delta\omega) = \cos(0) = 1$ and the remaining left signal is only the C/A code multiplied with the DC term (zero frequency producing a constant voltage) leaving only $\frac{1}{2}d_{C/A}$.

$$\Delta\omega = \omega_1 - \omega_2 = 0 \tag{2.17}$$

However, if the frequencies do not match, $f_1 \neq f_2$, then the output signal $\frac{1}{2}d_{C/A}$ shall be modified by the residual frequency $f_1 - f_2$, and subsequently this shall change the demodulated C/A output (also known as phase shift). Under those circumstances the correlator shall be unable to match the C/A code with the correct PRN code. An illustration of this phenomenon is depicted in figure 2.8.

[6] A low-pass filter passes low frequency signals and attenuates high frequency signals. In other words, signals higher than the specified cutoff frequency of the low-pass filter, are cut off by reducing their amplitudes.

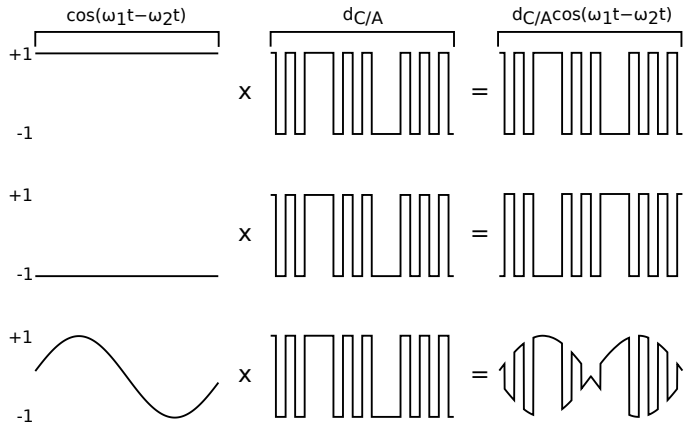

Figure 2.8.: Effects of the low frequency term on the demodulated output C/A wave on the GPS receiver (the explanations and figures are from top to bottom). If the synthesized frequency is correct, $f_1 = f_2$, the low frequency term becomes a DC term and does not modify the output $d_{C/A}$ wave (first figure). If the frequency matches but the phase not, in this case the phase is shifted for π, then $d_{C/A}$ is inverted (second figure). If the phase shifts with time, then the amplitude and phase of $d_{C/A}$ shall vary as well (third figure). Image courtesy of [21].

2.2.2. C/A wave demodulation

As a result of the previous step, one can continue with the demodulation of the C/A wave. Demodulating the C/A wave with the PRN code shall result in the time and navigation data. Each tracked GPS satellite signal is demodulated seperately using the same PRN code, code chipping rate and carrier frequency-phase (which was determined above) for the given satellite [25, Chapter 4]. The PRN codes for each GPS satellite is well defined and known by the GPS receiver. The receiver has to generate the equivalent PRN code with matching code chipping rate (phase) of the transmitted C/A code, this is depicted in figure 2.9 [25, Chapter 5]. This phase shift is again a consequence of the Doppler effect described in section 2.2.1. For the particular example, the matching phase shift was achieved with the second replica PRN code, with a phase shift of $\tau = 0$ but there could be a case with any other value of τ, $\tau \in [0, 1022]$. Implementation of the PRN code synthesizer depends on the GPS receiver manufacturer but it is usually implemented as a linear feedback shift registers (LFSR) that produces an output according to a predefined function $f(\tau)$. This function, $f(\tau)$, generates an PRN code, that is delayed in phase by τ, where τ is a multiple of the chipping rate period $T_c = 977.5\,ns$. The chipping period T_c can be derived from equation (2.18). The amount of time required to find a matching PRN code shift, τ, on the receiverr is proportional to the amount of parallely working

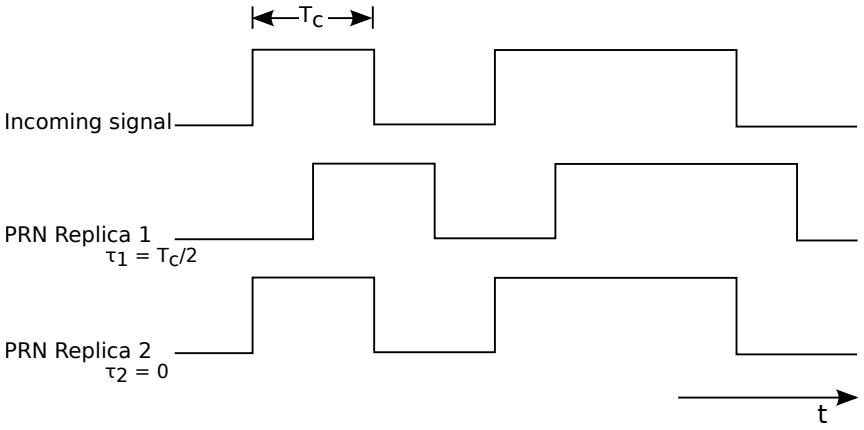

Figure 2.9.: Comparison between the original C/A code generated on the GPS satellite with two synthesized PRN codes with a different phase shift on the receiver. Image courtesy of [25].

LFSRs on the system [10, Chapter 3]. Clearly with more LFSRs the required time for finding the matching phase shift increases.

$$T_c = \frac{1}{f_{PRN}} = \frac{1}{1.023 \cdot 10^6 \text{Hz}} \tag{2.18}$$

To determine whether the synthesized PRN code, matches the incoming C/A code of the received satellite signal, known correlation properties of PRN codes are used, as described in section 2.1. Since the PRN code is modeled as a sequence of +1's and -1's, the autocorrelation of a signal is at its maximum if it is in phase, i.e. summing up the sequence products yields the absolute maximum value for the case where each bit from one signal matches the bit from the other signal. As an illustration of the idea, an example is given in figure 2.10. The cross-correlation of the incoming C/A code with the first synthesized PRN code produces a result of $-3 = (+1) \cdot (-1) + (-1) \cdot (+1) + (+1) \cdot (-1) + (+1) \cdot (+1) + (-1) \cdot (+1)$, whereas the cross-correlation of the incoming C/A code and the second synthesized PRN code yields a result of $+5 = (+1) \cdot (+1) + (-1) \cdot (-1) + (+1) \cdot (+1) + (+1) \cdot (+1) + (-1) \cdot (-1)$.

The same principle applies to the transmitted C/A and generated PRN code sequences in the GPS receiver. Thus, this can be modeled using the equation given in (2.19), where $G_i(t)$ is the C/A code[7] as a function of time t, for the GPS satellite i; $T_{C/A}$ is the C/A chipping period of $977.5\,ns$ and τ is the phase shift in the auto-correlation function [25, Chapter 4].

[7] PRN generated code for GPS satellites is called Gold code sequences since they were first discovered by Dr. Robert Gold.

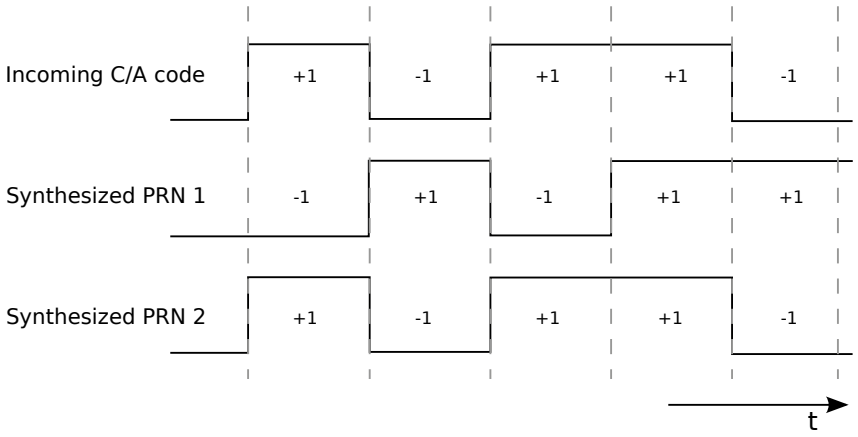

Figure 2.10.: Cross-correlation on three different signals. Image courtesy of [25].

$$R_i(t) = \frac{1}{1023 \cdot T_{C/A}} \int_{t=0}^{1022} G_i(t) G_i(t+\tau) d\tau \tag{2.19}$$

Another correlation property of the PRN codes is used, the fact that in the ideal case the cross-correlation of two different PRN codes yields a result of zero. The ideal case of PRN code can be modeled as in equation (2.20),

$$R_{ij}(\tau) = \int_{-\infty}^{+\infty} PRN_i(t) PRN_j(t+\tau) d\tau = 0 \tag{2.20}$$

where PRN_i is the PRN code waveform for GPS satellite i and PRN_j is the PRN code waveform for every other GPS satellite other than i, $i \neq j$ [25, Chapter 4]. Equation (2.20) "states that the PRN waveform of satellite i does not correlate with PRN waveform of any other satellite j for any phase shift τ" [25, Chapter 4]. Without the property given in (2.20), the GPS receiver would not be able to smoothly differentiate between different GPS satellite signals. Once the phase shift, τ, has been found, the C/A code is modulated (XORed) with it. The resulting binary code shall be the navigation message. The implementation problem of finding correct C/A and carrier wave demodulation shall be further explained in the following section 2.2.3.

2.2.3. Implementation of the 2D search space problem

In the following paragraphs an introduction shall be given on the implementation problems of the previously discussed concepts. As it can be seen, from subsections 2.2.1 and 2.2.2, decoding the GPS navigation message is a 2D search space problem for each GPS satellite signal acquisition. The 2D search space is limited by well

known physical properties of the GNSS system such as the motion speed of GPS satellites (and the receiver) as well as the frequency oscillator on the receiver.

GPS satellites move toward or away from the GPS receiver with a speed of 800 m/s [21, Chapter 3]. The Doppler effect on the frequency of the satellite can be estimated using equation (2.21), where f_e is the emitting frequency (L1), v_{SV} is the speed of the satellite towards (away from) the receiver and c is the speed of light.

$$f_{DE} = f_e \frac{v_{SV}}{c} \tag{2.21}$$

By inserting the appropriate values in equation (2.21) yields a result of ≈ 4.2 kHz, for 800 m/s and ≈ −4.2 kHz (if the satellite moves away from the GPS receiver then the speed is taken as negative). This makes a total range of ≈ 8.4 kHz. The Doppler effect of the GPS receiver motion can be ignored since for each 1 km/h of movement, it affects the frequency range for ≈ 1.46 Hz.

On the other hand, the frequency offset induced by the reference oscillator in the GPS receiver can not be ignored. Function of the reference oscillator is to give the GPS receiver the clock pulse required for all the computations and comparisons. The frequency search space is "additionaly affected for 1.575 kHz of unknown frequency offset for each 1 ppm (*parts per million*) of the unknown receiver oscillator offset" [21, Chapter 3]. The reference oscillators in GPS receivers have typically an offset of $\pm 0.5, \pm 1, \pm 2, \pm 3,$ or $\pm\, 5$ ppm [17], [21, Chapter 3], the standard in smart phone design has been set to ±2.5 ppm [67]. In the worst case this makes the unknown frequency to be in range of 10 kHz – 25 kHz.

A typical receiver searches in frequency bands (bins) of several hundred Hz [35]. Commonly used frequency bin size is 500 Hz, therefore there are about 20-50 bins to search (10000 Hz/500 Hz = 20) [21, Chapter 3]. The frequency search bin (band) size is a function of the desired peak magnitude loss (signal to noise ratio) due to the frequency mismatch and integration time period. Larger frequency bands mean a smaller number of bins to search but a greater correlation peak magnitude loss, i.e. with larger frequency bands it becomes harder to identify the correlation peaks described in sections 2.1 and 2.2.2. The frequency search bin size can be estimated using the frequency mimsmatch loss *sinc* function given in equation (2.22) [54], [21, Chapter 6], where Δf is the frequency mismatch in Hz, in other words it represents the difference between the received signal frequency and the synthesized carrier frequency on the receiver; and T_{ci} is the coherent integration time (usually $0.5\,ms$ according to [54] and [21, Chapter 3] but depends on the implementation).

$$D_F = \left| \frac{\sin(\pi \Delta f T_{ci})}{\pi \Delta f T_{ci}} \right| \tag{2.22}$$

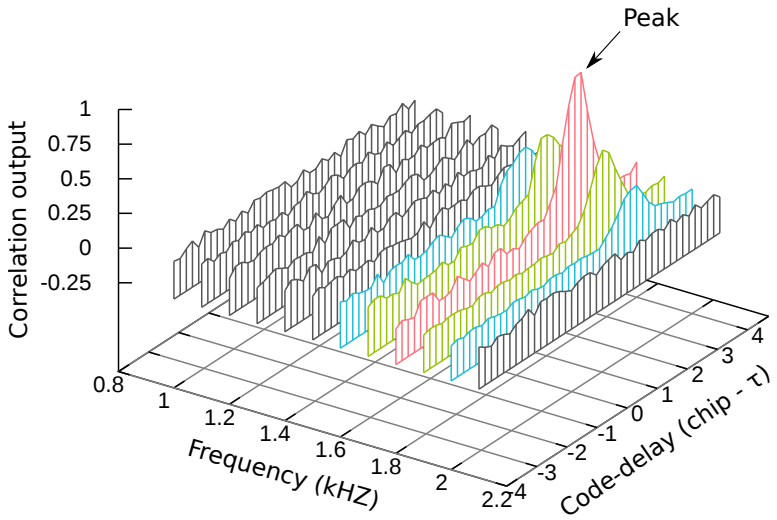

Figure 2.11.: Segment of the frequency/code delay search space for a single GPS satellite. Image courtesy of [21].

The frequency mimsmatch loss sinc function, D_F, is evaluated in dB, therefore for a loss of $\approx 0.98\,\text{dB}$, the frequency mismatch ought to be $\Delta f = 250\,\text{Hz}$, due to the fact that the maximum loss shall occur when the frequency is differing by 1/2 of the bin spacing. That is to say, for a bin space of 500 Hz, it is 250 Hz.

"The total range of possible GPS code delays is $1\,ms$. This is because the GPS C/A PRN code is $1\,ms$ long, and then it repeats. The PRN code chipping rate is 1.023 MHz, and there are 1023 chips in the complete $1\,ms$ epoch" [21, Chapter 3].

For the purpose of better understanding, a segment of the frequency/code delay search space is shown in figure 2.11. The peak implies the correct frequency and code delay have been found. In figure 2.11 smaller frequency bins have been used so that the concept becomes understandable to the reader.

The speed of searching the 2D search space (finding the peak) depends on the complexity and strategy of the implemented algorithm [13, Chapter 6]. In the worst case, there are in total 102300 conbinations in the search space, this can be derived from equation (2.23), visually shown in figure 2.12.

$$\text{Search Space} = 50\,(\text{bins}) \cdot 1023\,(\text{C/A codes}) \cdot 2\,(\text{Phases per C/A chip}) \tag{2.23}$$

The common strategy is to start searching from the middle frequency bins and to jump up and down until the entire search space has been exhausted (first 500 Hz, second -500 Hz, then in the 1000 Hz bin and then in the -1000 Hz bin), as shown in

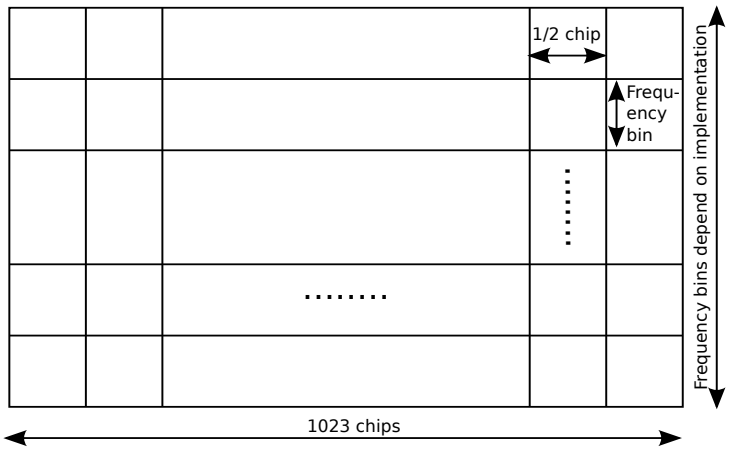

Figure 2.12.: The total search space.

figure 2.13 [21, Chapter 3]. This procedure is performed when no extra information are known by the receiver (almanac data are missing), i.e. first time the GPS receiver is turned on. It is known under the name of cold start.

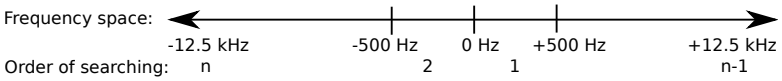

Figure 2.13.: Idea of the frequency searching algorithm.

There are three different working modes when it comes to searching for the GPS satellites. If no information are known, when some information are known and when almost all information are known. These three modes are known as *cold* (as discussed earlier), *warm* and *hot* start. They differ from each other by the amount of known information by the GPS receiver. Cold start indicates the GPS receiver has no almanac, ephemeris, oscillator offset and time data. In order to track the satellites faster next time the GPS receiver is started, it stores the previously mentioned data (last known almanac, ephemeris, oscillator offset, time and position data) in its electrically erasable programmable read only memory (EEPROM). This new type of start, is known as a warm start, provided that the data in the receivers' EEPROM are not older than 180 days and its real time clock counter was constantly updated. In this case, the receiver uses the previously saved information to estimate the position of the satellites, therefore the Doppler effects can be roughly estimated. As a consequence of the known Doppler effect, the frequency bin where to start the search first is known this time [21, Chapter 3]. Hot start works in the same manner, only the ephemeris data and time data are precisely known (time ought to be known in accuracy of submilliseconds).

2.3. Distance and position estimation

In this section the focus is set on distance and position estimation inside of the GPS receiver. GPS system, as discussed earlier, takes advantage of the TOA ranging concept to determine user position. Time is measured how long it takes for a signal to arrive from a known location. In figure 2.14, an example concept can be seen,

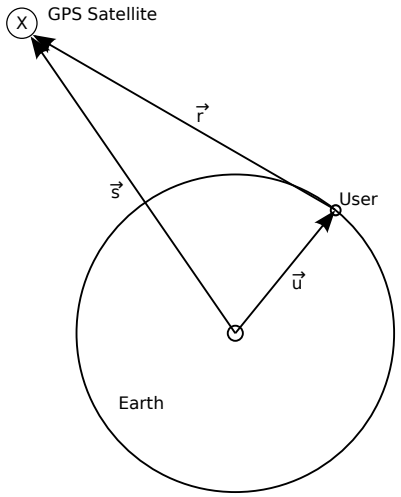

Figure 2.14.: Basic distance estimation principle for one satellite. Image courtesy of [25].

where $\vec{u} = (x_u, y_u, z_u)$ represents the unknown GPS user position vector with respect to Earth-Centered, Earth-Fixed[8] (ECEF) coordinate system, $\vec{r}$ is the distance vector from the satellite to the user and $\vec{s} = (x_s, y_s, z_s)$ represents the GPS satellite position with respect to ECEF at a timepoint. Vector $\vec{s}$ is computed from ephemeris data broadcasted by the satellite. The distance vector $\vec{r}$, distance between the satellite and user, can be computed using equation (2.24) and its magnitude is given in equation (2.25).

$$\vec{r} = \vec{s} - \vec{u} \tag{2.24}$$

$$r = \|s - u\| \tag{2.25}$$

The geometric distance of r is computed by measuring the signal propagation time, this is illustrated in figure 2.15 and it was discussed in section 2.2.2. The PRN code generated on the GPS satellite at time t_1 arrives at the time t_2, the difference between these two time stamps, Δt, represents the propagation time. By multiplying the propagation time, Δt, with the speed of light, c, the geometric distance r is computed, as given in equation (2.26).

[8] ECEF is a Cartesian coordinate system where the point $(0, 0, 0)$ is defined as the center of mass of the Earth [15].

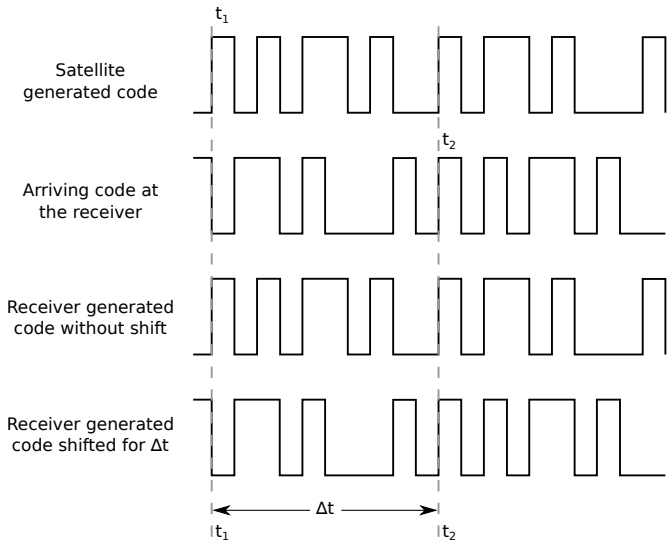

Figure 2.15.: Estimating the distance by phase shift $\Delta t = t_2 - t_1 = \tau$. Image courtesy of [25].

$$r = c\Delta t \tag{2.26}$$

Since the clocks are not synchronized, as described in sections 2.2 and 2.2.3, clock error offsets have to be added to the geometric distance r. This new distance is called *pseudorange*, ρ, because the range is determined using the difference of two nonsynchronized clocks (one on the GPS satellite and the other one on the receiver) that generate PRN codes[9]. Pseudorange is calculated as given in equation (2.27), where t_u is the advance of the receiver clock with respect to the system time[10] and δt is the offset of the satellite clock from the system time [25].

$$\rho = r + c(t_u - \delta t) \tag{2.27}$$

Equation (2.25) can be rewritten as (2.28) with respect to equation (2.27).

$$\rho - c(t_u - \delta t) = \|s - u\| \tag{2.28}$$

Offset of the satellite clock from the system time, δt, is updated from Earth, as discussed in 2.2 and for that reason it can be removed for sake of simplicity, i.e. it is not an unknown term anymore, then the eqaution (2.28) can be rewritten as (2.29).

$$\rho - ct_u = \|s - u\| \tag{2.29}$$

[9]pseudo - Not genuine; sham; not perfect.
[10]System time is the exact time on Earth and it is the most precise time known!

In order to estimate the user (GPS receiver) position, advance of the receiver clock with respect to the system time, t_u, has to be found, in other words equation (2.30) has to be solved, where i is the index of visible satellites at the moment of signal reception [25].

$$\rho_i = \|s_i - u\| + ct_u \tag{2.30}$$

The estimated position of the user, $\vec{u} = (x_u, y_u, z_u)$, is a three dimensional vector and as stated above the clock offset, t_u, is unknown as well. This four dimensional space requires to have at least four pseudorange equations (2.30) to find all the four unknown terms. As a result of this fact, at least four satellites have to be visible at the same time to estimate the position of the target user. Equation given in (2.30) takes the form in (2.31) because the coordinate system is Cartesian and ρ_i is nothing else but Euclidean distance where $i = 1, 2, ..., n$ such that $n \geq 4$ and $\vec{s_i} = (x_i, y_i, z_i)$ is the satellite position estimated from the ephemeris data.

$$\rho_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2} + ct_u \tag{2.31}$$

Undoubtedly, the given equation in (2.31) is a nonlinear equation[11]. It is not straightforward to find explicit solutions of nonlinear equations, it is more difficult than compared to linear equations. There are different techniques to solve sets of nonlinear equations [25, Chapter 7] but in this work the linearization method[12] shall be presented to find the unknown terms (x_u, y_u, z_u, t_u), i.e. out of an approximate position and clock offset the true user position and the true clock offset shall be calculated.

$$\rho_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2} + ct_u = f(x_u, y_u, z_u, t_u) \tag{2.32}$$

Let the equation (2.31) for pseudoranges, be rewritten as a function f of four unknown terms x_u, y_u, z_u and t_u, as given in (2.32) [25, Chapter 2]. Suppose that the approximation of the position and the clock offset are known, denoted as $\hat{x_u}$, $\hat{y_u}$, $\hat{z_u}$ and $\hat{t_u}$, then equation (2.32) can be rewritten as an approximate pseudorange (2.33).

$$\hat{\rho}_i = \sqrt{(x_i - \hat{x_u})^2 + (y_i - \hat{y_u})^2 + (z_i - \hat{z_u})^2} + c\hat{t_u} = f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u}) \tag{2.33}$$

In other words, the unknown true position terms x_u, y_u, z_u and the clock offset term t_u, of the GPS receiver, shall be expressed by the approximate values and an incremental component as shown in equation (2.34) [25].

$$\begin{aligned} x_u &= \hat{x_u} + \Delta x_u \\ y_u &= \hat{y_u} + \Delta y_u \\ z_u &= \hat{z_u} + \Delta z_u \\ t_u &= \hat{t_u} + \Delta t_u \end{aligned} \tag{2.34}$$

[11] Nonlinear equations, also known as polynomial equations, are equations that can not satisfy both of the linearity properties: additivity $f(x + y) = f(x) + f(y)$ and homogeneity $f(\alpha x) = \alpha f(x)$, $\alpha \in \mathbb{R}$ [63].

[12] Linear approximation is a technique where a function is approximated using a linear function.

By inserting the terms from (2.34) into equation (2.32), a new equation is derived as in (2.35).

$$f(x_u, y_u, z_u, t_u) = f(\hat{x_u} + \Delta x_u, \hat{y_u} + \Delta y_u, \hat{z_u} + \Delta z, \hat{t_u} + \Delta t_u) \tag{2.35}$$

In the next step the pseudorange function shall be approximated using Taylor series[13] (linearization of the nonlinear equation). Taylor series for a function $f(x)$ is given in equation (2.36), where as a approaches x the estimation error shall be smaller and smaller, i.e. $f(x) = f(a)$ when $x = a$. The approximation error depends on Taylor polynomial degree (the amount of terms or taken derivatives of the function) and how far away the point a is from x [77, Chapter 11.9]. The basic idea of the principle can be seen in figure 2.16.

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \ldots \tag{2.36}$$

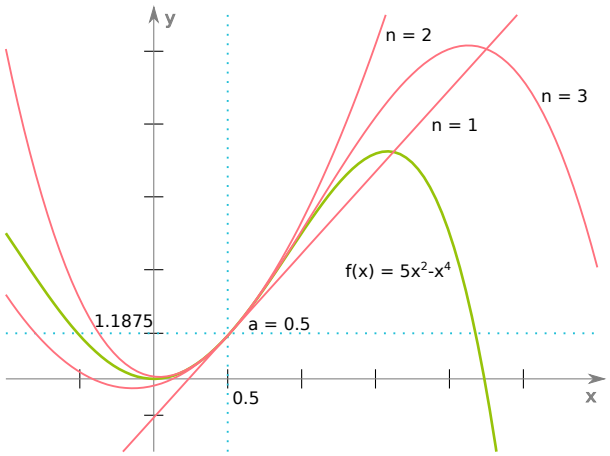

Figure 2.16.: Taylor series approximation for a point $a = 0.5$ where n is the Taylor polynomial degree.

Due to the four unknown terms, Taylor series for multivariables have to be used. The general formula is given in equation (2.37), where vector $\mathbf{x} \in \mathbb{R}^n$ denotes n variables, ∇ (nabla) is the Del[14] operator given in (2.38) and $\mathbf{a}$ is the linearization point of interest [42].

$$f(\mathbf{x}) \approx f(\mathbf{a}) + \nabla f|_{\mathbf{x}=\mathbf{a}} \cdot (x-a) \tag{2.37}$$

$$\nabla^T = \left[\frac{\partial}{\partial x_1} \ldots \frac{\partial}{\partial x_n}\right] \tag{2.38}$$

[13] Taylor series "is a representation of a function as an infinite sum of terms that are calculated from the values of the function's derivatives at a single point" [77, Chapter 11].

[14] Del, ∇, is the vector differential operator.

One can note that in equation (2.37) the Taylor series polynomial is of the first degree. This is because of one reason, it linearizes the approximation of the function $f(\mathbf{x})$ at point $\mathbf{a}$ and as a consequence it removes the nonlinearities [25] [77, Chapter 11.10], as seen in figure 2.16, for $n = 1$ the resulting function is linear. In the previously described step, one would calculate a hyperplane tangent to a point a in a n-Dimensional space. By inserting equation (2.35) in equation (2.37), it yields equation (2.39) where $\mathbf{x} = (x_u, y_u, z_u, t_u)$ and $\mathbf{a} = (\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})$.

$$\begin{aligned} &f(\hat{x_u} + \Delta x_u, \hat{y_u} + \Delta y_u, \hat{z_u} + \Delta z, \hat{t_u} + \Delta t_u) \approx f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u}) \\ &+\frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{x_u}} \Delta x_u + \frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{y_u}} \Delta y_u \\ &+\frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{z_u}} \Delta z_u + \frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{t_u}} \Delta t_u \end{aligned} \tag{2.39}$$

The terms from equation (2.39) are solved individually in equations (2.40) where $\sqrt{(x_i - \hat{x_u})^2 + (y_i - \hat{y_u})^2 + (z_i - \hat{z_u})^2}$ has been subsituted with $\hat{r_i}$.

$$\begin{aligned} \frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{x_u}} &= \frac{1}{2} \frac{-2(x_i - \hat{x_u})}{\sqrt{(x_i - \hat{x_u})^2 + (y_i - \hat{y_u})^2 + (z_i - \hat{z_u})^2}} = -\frac{x_i - \hat{x_u}}{\hat{r_i}} \\ \frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{y_u}} &= \frac{1}{2} \frac{-2(y_i - \hat{y_u})}{\sqrt{(x_i - \hat{x_u})^2 + (y_i - \hat{y_u})^2 + (z_i - \hat{z_u})^2}} = -\frac{y_i - \hat{y_u}}{\hat{r_i}} \\ \frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{z_u}} &= \frac{1}{2} \frac{-2(z_i - \hat{z_u})}{\sqrt{(x_i - \hat{x_u})^2 + (y_i - \hat{y_u})^2 + (z_i - \hat{z_u})^2}} = -\frac{z_i - \hat{z_u}}{\hat{r_i}} \\ \frac{\partial f(\hat{x_u}, \hat{y_u}, \hat{z_u}, \hat{t_u})}{\partial \hat{t_u}} &= c \end{aligned} \tag{2.40}$$

Then by substituting the equation terms from (2.40), (2.32) and (2.33) into (2.39), the resulting equation is given in (2.41).

$$\rho_i = \hat{\rho_i} - \frac{x_i - \hat{x_u}}{\hat{r_i}} \Delta x_u - \frac{y_i - \hat{y_u}}{\hat{r_i}} \Delta y_u - \frac{z_i - \hat{z_u}}{\hat{r_i}} \Delta z_u + c\Delta t_u \tag{2.41}$$

At this step, by solving equation (2.39), the linearization of the nonlinear equations is completed.

$$\hat{\rho_i} - \rho_i = \frac{x_i - \hat{x_u}}{\hat{r_i}} \Delta x_u + \frac{y_i - \hat{y_u}}{\hat{r_i}} \Delta y_u + \frac{z_i - \hat{z_u}}{\hat{r_i}} \Delta z_u - c\Delta t_u \tag{2.42}$$

$$\Delta\rho = \hat{\rho_i} - \rho_i \tag{2.43}$$

$$\alpha_{xi} = \frac{x_i - \hat{x_u}}{\hat{r_i}} \qquad \alpha_{yi} = \frac{y_i - \hat{y_u}}{\hat{r_i}} \qquad \alpha_{zi} = \frac{z_i - \hat{z_u}}{\hat{r_i}} \tag{2.44}$$

By rearanging the equation (2.41) one derives equation (2.42). And then by substituting the terms in (2.43) and (2.44) into (2.42), the equation resembles the one given in (2.45).

$$\Delta\rho_i = \alpha_{xi}\Delta x_u + \alpha_{yi}\Delta y_u + \alpha_{zi}\Delta z_u - c\Delta t_u \tag{2.45}$$

There are four unknowns, Δx_u, Δy_u, Δz_u and Δt_u, in equation (2.45). By solving this set of linear equations, which shall result in finding Δx_u, Δy_u, Δz_u and Δt_u, the GPS receiver position (x_u, y_u, z_u) and clock offset t_u is computed by replacing the same into equations in (2.34). Equation (2.45) can be rewritten for four satellites in the matrix form as in (2.46).

$$\Delta\boldsymbol{\rho} = \boldsymbol{\alpha}\Delta\boldsymbol{x} \tag{2.46}$$

$$\Delta\boldsymbol{\rho} = \begin{bmatrix}\Delta\rho_1\\ \Delta\rho_2\\ \Delta\rho_3\\ \Delta\rho_4\end{bmatrix} \qquad \boldsymbol{\alpha} = \begin{bmatrix}\alpha_{x1} & \alpha_{y1} & \alpha_{z1} & 1\\ \alpha_{x2} & \alpha_{y2} & \alpha_{z2} & 1\\ \alpha_{x3} & \alpha_{y3} & \alpha_{z3} & 1\\ \alpha_{x4} & \alpha_{y4} & \alpha_{z4} & 1\end{bmatrix} \qquad \Delta\boldsymbol{x} = \begin{bmatrix}\Delta x_u\\ \Delta y_u\\ \Delta z_u\\ -\Delta c t_u\end{bmatrix} \tag{2.47}$$

Finally, by multiplying both left sides[15] of the equation (2.46) with the inverse term of $\boldsymbol{\alpha}$, it yields the result of the unknown terms, as given in equation (2.49).

$$\boldsymbol{\alpha}^{-1}\Delta\boldsymbol{\rho} = \boldsymbol{\alpha}^{-1}\boldsymbol{\alpha}\Delta\boldsymbol{x} \tag{2.48}$$

$$\Delta\boldsymbol{x} = \boldsymbol{\alpha}^{-1}\Delta\boldsymbol{\rho} \tag{2.49}$$

Linearization is repeated in a loop, where in the next round the approximate positions are set to the just derived position values, that is, $\hat{x_u} = x_u$, $\hat{y_u} = y_u$, $\hat{z_u} = z_u$ and $\hat{t_u} = t_u$. This process is repeated until the approximated positions converge to their final values. It is not necessarily required that the initial positions are very accurate and the results are usually obtained by 4-5 itterations [50]. Risks exist that the solutions shall still be corrupted but there are different error avoiding mechanisms to solve these problems, like minimizing the error contribution using more than four satellite measurements [50] [25, Chapter 7].

[15] Matrix multiplication is not communitative, $\mathbf{AB} \neq \mathbf{BA}$.

2.4. Assisted GPS in Wireless networks

In the following paragraphs Assisted GPS (AGPS) shall be presented and how it works. AGPS receivers work on the equivalent idea as warm/hot start on GPS receivers. Instead of loading the recently saved data from the EEPROM, an external information transfer medium is used to deliver the equivalent type of information that are known at a warm/hot start [73], [22], [7]. In this work, the external transfer medium is air and the information are transferred using electromagnetic waves. The existing GSM interface was utilised for the purpose of delivering the data to the smart phone with an AGPS receiver. The basic scenario can be seen in figure 2.17.

The BTS station is connected to the global navigation satellite system (GNSS) server, which is directly connected to the GPS reference station. The GPS reference station delivers the GNSS server exact time stamps, approximate location, satellite health as well as clock corrections, ionospheric and UTC model, almanac and ephemeris data [7].

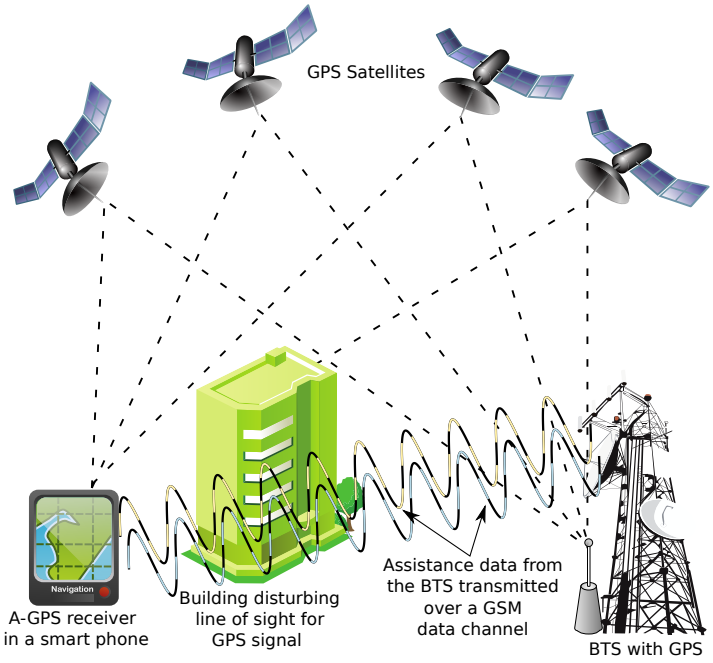

Figure 2.17.: Basic AGPS principle

Time stamp is not used in GSM networks since it can be off by several seconds and would require additional equipment for synchronizing the network [7], [22]. However in CDMA networks the time stamp is accurate to within $100\,\mu s$ [7]. Approximate

location is typically taken to be the location of the BTS from which the target AGPS receiver acquires the assistance data. Ephemeris and navigation data obtained by the AGPS receiver in the smart phone help it to estimate the positions of the GPS satellites. This method can greatly enhance the sensitivity of the receiver especially in urban environments [7].

Conventional GPS receivers require at least up to extra 18 to 30 s to receive and decode the navigation data and to generate a location fix [7]. The bit error rate associated with gathering and decoding data dramatically decreases since the acquired signals can be attenuated by 10 to 20 dB indoors [7] of the nominal −130 dB on a 3 dBi "linearly polarized user receiving antenna[16] (located near ground) at worst normal orientation" [2].

A simplified AGPS algorithm given in [7] shall be presented here. This algorithm benefits in speed the more assistance data is present. As the first satellites are tracked, the AGPS algorithm has an estimation of the feasible region where the target AGPS user might be located. Consequently, this feasible region shall shrink until the location has been fully estimated [7].

(i) Visible satellites and their positions are identified and computed out of the delivered ephemeris and time data.

(ii) For each visible satellite SV_i, the code phase, τ_i, is estimated.

(iii) Pseudoranges are calculated for each visible satellite SV_i.

(iv) Trilaterate the position out of the pseudoranges ρ_i.

Although the AGPS algorithms can be seen as a set of equations with more unknown terms being known. It is straightforward to solve a set of equations when all the terms are known. However, without assistance information which provide additional information to the GPS receiver, it takes more time to obtain (decode) assistance data from the satellite message. Numerous AGPS algorithms exist, some do not require the exact time component and navigation data to be present in the assistance data [8].

2.5. Error estimation

[16] 3 dBi antenna indicates an antenna with a gain of 3 dB with respect to an isotropic (omnidirectional) antenna [21, Chapter 2].

3. GSM

In the past two decades we have been witness to an increasing development of wireless communication technologies, one of the most rapidly developing fields of engineering. Global System for Mobile Communications[1] (GSM) networks played a major role in wide-spreading wireless voice services in every corner of the planet [34]. According to the GSM Association (GSMA) in 2011 there have been 6 billion registered wireless connections world wide [34]. In this chapter more details shall be given on the second generation GSM network which was employed in this work for delivering GPS assistance data to cell phones. More information shall be provided on the general working principles of GSM and how a Standalone Dedicated Control Channel (SDCCH) is initialized to deliver data to cell phones.

[1] First time when the standard was developed, GSM meant *Groupe Spéciale Mobile* [38]

3.1. Overview of the Air interface

In this section the reader shall be provided with principles how the GSM network operates. The main task of GSM networks was to enable wireless voice transmission between GSM and other GSM/telephone users inside of switched networks. It was not designed to be used with data services which are a necessity in today's standards. GSM networks are worldwide spread and work on different frequency spectrums depending on the country where the networks are employed. The reason why different frequencies are used is because of intereference with different wireless systems and used telecommunication standards. Particularly in Germany, the Federal Network Agency (German: *Bundesnetzagentur*) is the responsible organisation for assigning different frequencies to GSM operators since these frequencies belong to the group of licensed frequencies and are not allowed to be used by everyone. In Germany the used frequency bands are GSM900, EGSM900 and GSM1800, their frequency ranges can be seen in table 3.1 [58]. These frequency bands are divided into 200 KHz channels, for a frequency band range of 25/35 MHz there are 124/175 operating channels. This technique is called Frequency Division Multiple Access (FDMA) and supports using parallely more frequency channels inside of the same covered area with GSM RF signal. FDMA is employed when the frequency bandwidth is limited like in the GSM networks. By utilising FDMA the network throughput is used more efficiently since different users can send or receive information at different frequency slots instead of waiting for their turn. These frequency channels have a unique identifier number. They are named as Absolute Radio Frequency Channel Numbers (ARFCN). The basic idea of FDMA inside of the frequency spectrum GSM900 for GSM can be seen in figure 3.1, ARFCN numbers are assigned according to the frequencies which are employed. It is important to distinguish uplink and downlink frequencies. Uplink frequency is used when the cell phone transmits data to the network operator, whereas downlink from the network operator to the cell phone. GSM is a full duplex communication system, at the same time the cell phone or the network operator can send and receive data. Although the equivalent ARFCN number is used for uplink and downlink, the frequencies are shifted 45 MHz in EGSM900/GSM900 and 95 MHz in GSM1800 as it can be seen in figure 3.1 for GSM900.

Table 3.1.: GSM operating frequencies in Germany

Frequency band	Uplink frequency (MHz)	Downlink frequency (MHz)	Channel number
GSM900	890 - 915	935 - 960	1 - 124
EGSM900	880 - 915	925 - 960	0, 1 - 124, 975 - 1023
GSM1800	1710 - 1785	1805 - 1880	512 - 885

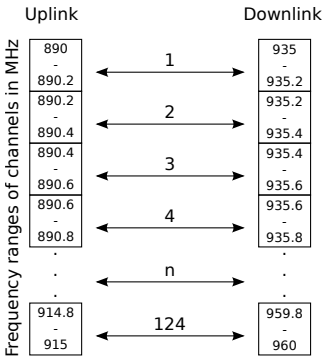

Figure 3.1.: Frequency ranges of uplink and downlink channels in the GSM900 band. Each box represents a frequency band (channel). Image courtesy of [58] and [82].

Aside from using different frequency channels, each frequency channel is split up into eight time slots. This technique of dividing a frequency into time slots is named Time Division Multiple Access (TDMA). TDMA allows several users to share the same frequency channel but in different time slots. Using this technique the voice throughput is better utilised and a broader amount of users can be served at the "same" time, i.e. the capacity of parallely speaking GSM users is increased. TDMA was employed because the voice could be compressed with Linear Predictive Coding (LPC) without the human noting a difference in the call quality [18]. By taking advantage of LPC, instead of the 64 kbps required for transmission of voice it was possible to compress the voice without losing much of the call quality into 8 kbps for half rate and 16 kbps for full rate[2]. Since new wireless services are data oriented and the networks become packet networks this type of modulating data had to be changed, 3G and 4G networks use different frequency ranges and technique to modulate and demodulate data.

TDMA applied on the FDMA technique constrains the GSM air interface to be of 2D structure. The idea of employing TDMA on FDMA in the GSM900 band can be seen in figure 3.2. Each time slot duration is ≈577 μs, all 8 time slots have a period of ≈ 4.615 ms [82] [38]. By applying this technique each GSM user can send data inside of the assigned time slot without disturbing users on different time slots.

Eight time slots in GSM are called a TDMA frame. Each time slot in GSM is known as a physical channel, on the physical channels are built up the logical

[2]Human speech has a frequency bandwidth between 0 and 4000 Hz [19]. Human voice is by its nature analog and requires to be converted into a digital stream of ones and zeros. By Nyquist-Shannon sampling theorem the sampling frequency must be at least two times greater than the sampled frequency and with an 8 bit Analog to Digital Converter (ADC) this defines the 64 kbps required to transfer voice ($2 \cdot 4000Hz \cdot 8 = 64000$).

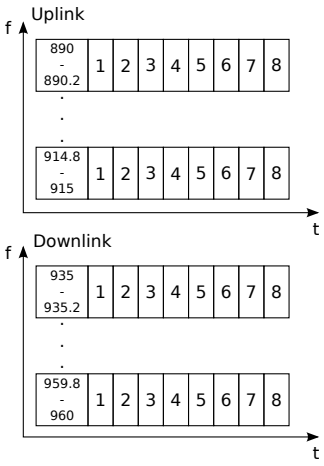

Figure 3.2.: Each frequency channel is split into 8 time slots. More GSM users can be served at the "same" time. Image courtesy of [38].

channels. Logical channels have a predefined pattern of time slot they are assigned. Logical channels can be divided in two groups, traffic channels (TCH) and signalling channels (SCH). User payload data like speech and message data are transmitted in the TCH channels whereas control data for control, synchronization and management of the GSM network are transmitted through the SCH channels [24, Chapter 4].

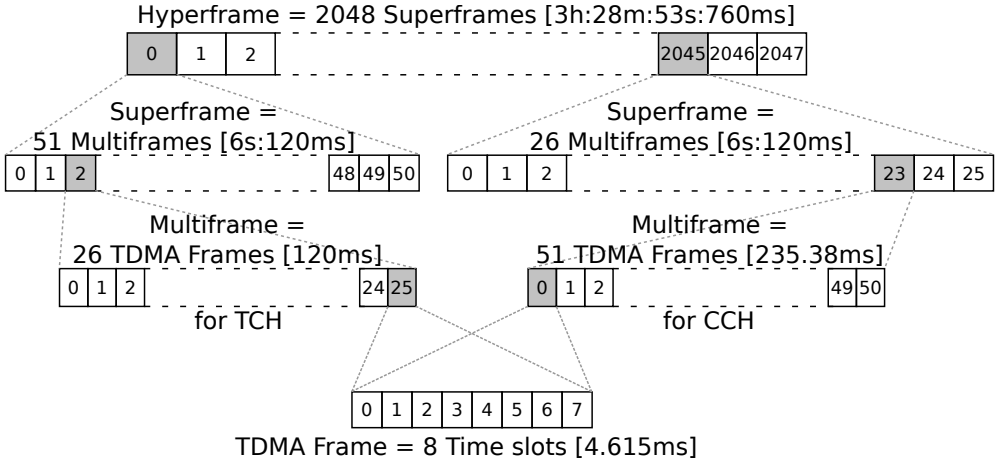

Figure 3.3.: Hierarchy of the GSM frames. Image courtesy of [38].

Every TDMA frame is assigned a unique integer number which is then repeated and reassigned every 3h:28m:53s:760ms, also known as *hyperframe* [38, Chapter 7]. In the hierarchy pyramid, a layer lower of the hyperframe is the *superframe*. There are two types of superframes, consisting of two types of *multiframes*, differing in their length [38, Chapter 7]. The relations can be seen in figure 3.3 with their duration periods.

The multiframe with 26 TDMA frames carries only traffic channels and associated control channels. The other multiframe type, with 51 TDMA frames carries solely signalling data. This hierarchy constrain was defined due to internal synchronization of the GSM network and cyphering between the MS and the Base Transceiver Station (BTS) [38, Chapter 7].

3.2. GSM Network structure

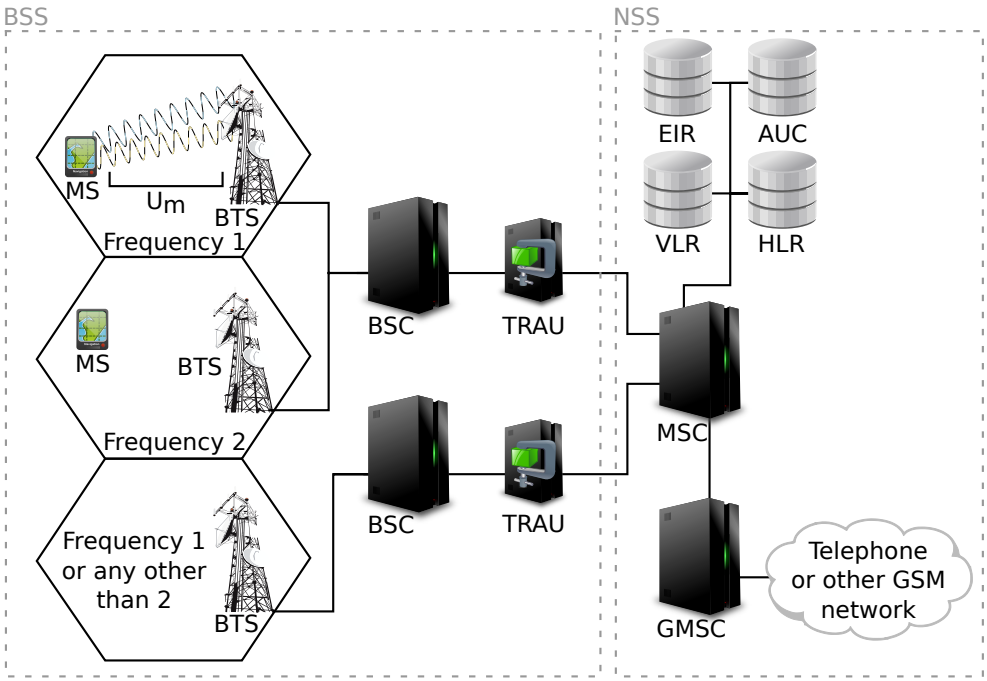

Figure 3.4.: Basic GSM network block diagram. Image courtesy of [58] and [82].

BTS is the first hardware unit the cell phone is communicating with over the air interface and provides a "physical" connection with the cell phone [38, Chapter 3]. This physical connection between the BTS and the cell phone is the U_m *interface*, as shown in figure 3.4. A BTS can serve up to seven users on one frequency in full duplex mode since one out of eight time slot is used for broadcasting of signalling and system information, transmitted in the broadcast control channel (BCCH). By sectorizing BTSs with different frequencies the number of seven mobile users can be increased. BTS consists of a RF tranceiver, internal clock and modulator/demodulator. The function of the RF transceiver is to enable the reception and transmission on the uplink and downlink channel for the cell frequency where the BTS is located[3]. The main function of the internal clock is to supply the BTS with a frequency such that the internal circuits can produce the TDMA frames. The internal clock has to be sufficiently accure for the GSM network to work, an accuracy of at least ±5 ppm (parts per million) [82]. If the GSM network is synchronized, this internal clock is not employed but an external clock generator signal from an atomic clock, this is required for some of the position localization techniques, as described in section

[3]Cell is the area covered with GSM signal and from which a cell phone can communicate with a BTS.

1.2.3. Modulator/demodulator main function is the modulation and demodulation of the received and transmitted signals. The transmission from the cell phone to the BTS is shifted for 3 time slots compared to the reception of the signal from the BTS[4] [38, Chapter 7] [58] [86, Chapter 4].

One or more BTSs are connected to the Base Station Controller (BSC). The main task of the BSC is to control the radio resources of the connected BTSs such as assigning radio channels to different BTS, frequency hopping in case of an handover and controlling the power levels within channel [86, Chapter 4] [58] [38, Chapter 3]. BSC is connected to the Transcoding Rate and Adaptation Unit (TRAU). This builds the Base Station Subsystem (BSS), as it can be seen in figure 3.4, on left side inside of the gray dashed line rectangle. Inside of the BSS, TRAU is responsibe for compressing and decompressing speech between the cell phone and a speech signal from the other side, from 64 kbps to 16 or 8 kbps depending if it is a full or half rate channel.

The next subsystem block is the Network Switching Subsystem (NSS), as it can be seen on figure 3.4, on right side inside of the gray dashed line rectangle. The main task of NSS is to connect the GSM with other telephony networks (GSM networks from other providers or the Public Switched Telephone Network) [86, Chapter 4]. It consists of Mobile Switching Center (MSC), Gateway Mobile Switching Center (GMSC) and databases.

MSC's main function is to route incoming and outgoing calls between the moving mobile users, "the assignment of user channels toward the BSS" [38, Chapter 4] [58]. GMSC is a type of MSC for external networks, GSM networks from other providers or telephone networks are routed through the GMSC[38, Chapter 4].

There are four databases: Home Location Register (HLR), Visitor Location Register (VLR), Authentication Center (AUC) and Equipment Identity Register (EIR). HLR database stores data about the GSM subscribers of a network provider. The data that can be found in HLR: the unique International Mobile Subscriber Identity (IMSI) - that is stored on the SIM card of a mobile user; usage statistics; subscriber's number (MSISDN) and the current location of the mobile user acquired by knowing the location of the BSC controlling the BTS that provides at the current moment the GSM air interface to the mobile user [58]. VLR serves as a temporary data storage of important parts of HLR data (not all data known for the particular user) of all the visiting mobile subscribers served by the current MSC. i.e. if a MS from its home MSC enters an area covered by the newly entered MSC, its VLR will request some of the HLR data from the HLR database of the MSC where the MS is registered [82]

[4]Timing advance factor is added to the three time slots.

[86, Chapter 4]. AUC contains confidential keys for each mobile subscriber required for encrypting the data before they are transmitted to the MS from the BTS [24, Chapter 3]. The keys located in AUC are also required for the MS to register in the network [58]. EIR is an optional database but contains data about approved types of mobile equipment (not stolen cell phones), black listed cell phones (they are identified by their International Mobile Equipment Identity number which is unique for every manufactured cell phone) and cell phones which ought to be tracked if they register [38, Chapter 4].

3.3. Logical channels and the SDCCH channel

In this section more details will be given on logical channels and the procedure to initialize (open) an SDCCH channel (Standalone Dedicated Control Channel). As stated in section 3.2, logical channels can be divided in two groups, traffic channels (TCH) and signalling channels (SCH). The former are employed for transfering payload data like speech and message data and the latter for managing and synchronizing the GSM network [24, Chapter 4]. Traffic and signalling channels can be split up by their usage, as given in tables 3.2 and 3.3.

Table 3.2.: Traffic channels on the Air interface

Channel name	Abbreviation	Function	Direction
Traffic channel full rate	TCH/F	Full rate traffic transmission	MS↔BSS
Traffic channel half rate	TCH/H	Half rate traffic transmission	MS↔BSS

Table 3.3.: Control channels on the Air interface

Channel name	Abbreviation	Function	Direction
Frequency correction channel	FCCH	Frequency correction for oscillator on MS	MS←BSS
Synchronization channel	SCH	Synchronization information (TDMA frame number to know current location in hyperframe)	MS←BSS
Broadcast common control channel	BCCH	Broadcast information about current BTS and its neighbouring cells	MS←BSS
Access grant channel	AGCH	Required to assign the MS an SDCCH or TCH channels	MS←BSS
Paging channel	PCH	Paging request is sent out when MS has incoming traffic (phone call, SMS, etc.)	MS←BSS
Cell broadcast channel	CBCH	Required to broadcast a message to all MS inside of a MSC (e.q. weather forecast)	MS←BSS
Standalone dedicated control channel	SDCCH	Exchange of signalling information between MS and BTS when no TCH is active	MS↔BSS
Slow associated control channel	SACCH	Transmission of signalling data during an active TCH connection (signal strength and sync. data)	MS↔BSS
Fast associated control channel	FACCH	Transmission of signalling data during an active connection but used only if necessary (e.q. handover)	MS↔BSS
Random access channel	RACH	Request from MS to BTS for a communication channel (e.g. a phone call from MS)	MS→BSS

The protocol scenario occuring in this work can be seen in figure 3.5 [36]. In order for the assistance data to be delivered to the MS, an SDCCH channel has to be initialized. This occurs in the following procedure, the BTS where the MS has

been lastly active or idle broadcast a paging request (PCH channel) to the selected MS. After the MS obtains the paging request, the MS shall try to send a random access request (RACH channel) using the Slotted Aloha protocol. Another MS could transmit a random access request in the same time slot allowing collisions to occur. In case there was no collision, if the BTS successfully received the random access request and at the moment of reception has a free SDCCH channels, it will immediatelly reserve an SDCCH channel and send the MS an assignment request (AGCH channel) back. After the MS obtains the assignment request, the SDCCH channel is initialized and data can be transferred in both directions, in this case assistance data to the MS and acknowledgements, errors or the position from the MS back to the BTS. In the case if all SDCCH channels are reserved, the network will queue an SDCCH request for later assignment or it may send an assignment reject. While the SDCCH channel connection has been established assistance data can be transmitted to the MS. While an active SDCCH conecction exists, the MS will receive and transmit radio link control messages (signal strength and synchronization data) on the SACCH channel [36].

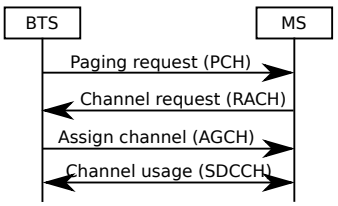

Figure 3.5.: Initializing an successful SDCCH channel. Image courtesy of [36].

4. Radio Resource Location Protocol

This chapter shall focus on the Radio Resource Location Protocol (RRLP) and a description how it works inside of the GSM network shall be given. RRLP is a protocol from the family of Location Services (LCS) which were not part of the initial GSM standard. It is a widely used protocol in other cellular networks like UMTS, it was later introduced to the GSM system as well [3]. It was developed by the request of government and rescue organizations to fulfill the wireless enhanced 911 standard in the US, each mobile user had to be located within a range of 300 m in 95% of cases and within 100 m in 67% of cases [29].

The standard supports three positioning mechanisms: E-OTD, UL-TDOA and AGPS [3]. The LCS process can be divided into two seperate stages, signal measurements and position estimation from the derived data in the previous stage. In this chapter the description shall be given on how to make an RRLP request, how to send assistance data and then more information shall be given on its response.

4.1. RRLP Request

In this section the RRLP protocol and its request shall be reviewed in more detail. RRLP represents the connection/protocol between the Serving Mobile Location Center (SMLC) and the standalone handset, in this case the MS [37, Chapter 5]. The SMLC node contains the functionaly to support location services for the GSM network [3]. SMLCs primary function is to manage the overall coordination and scheduling of resources required to perform the localization of the MS and it is located on the Base Station Controller (BSC) [3]. SMLC controls the LMU's as well but since in this work no LMU were available this part shall be skipped as well as the description of E-OTD and UL-TDOA localization.

Before an attempt is made, of requesting the SMLC to initialize an RRLP request, an SDCCH connection channel has to be initialized to the MS, this connection can not be seen by the MS user[1].

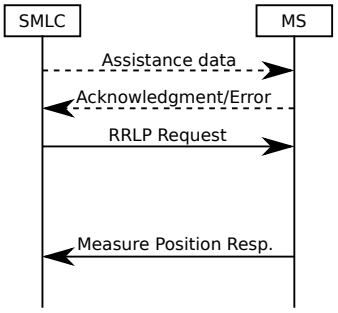

Figure 4.1.: RRLP Request protocol. Assistance data can be sent before the request is made. If the assistance data are sent, their reception acknowledgement is sent as a response from the MS. Image courtesy of [37] and [4].

Data sent inside of a protocol are called Protocol Data Unit (PDU). On different layer levels PDU's may take a different shape and size because of the encapsulation or splitting [52] [76]. In RRLP, the PDU's sent from the SMLC ought to be not greater than 244 bytes[2], although the standard defines that larger packets shall be split in lower layers, in this work the rule of 244 bytes has been obeyed and each PDU packet is not greater than 211 bytes [4]. In the RRLP standard terms, the messages are entitled as *components* and fields in the messages (components) are labelled as *information elements* (IE) [4]. The SMLC may send only the request

[1] However, it is possible to take into consideration that something is going on the cell phone if the MSs battery is drained faster because an active RF connection drains the battery faster than a passive MS connected to the GSM network.

[2] Bytes of 8 bits!

for the position of the MS or it may assist the MS with assistance data required to estimate the position (in case of an AGPS request, these data may be ephemeris, almanac, accurate timing data and similar data that help to estimate the position in a shorter period of time). The RRLP protocol is shown in figure 4.1. Dashed lines represent optionally transmitted data like assistance data according to which an acknowledgement or error shall be produced. Once the MS obtains the RRLP request, after a period of processing time it shall respond to the SMLC with the position of the MS or with an error IE indicating what assistance data are missing or why it can not return the position [4] [5]. In the response component IE it is exactly indicated what type of data ought to be sent to the MS so it can complete the RRLP request and give back its position. To save bandwidth space in the communication between the SMLC and MS, it can be proceeded in such a manner that first the RRLP request is sent out for the position estimation and then if the MS requires some of the assistance data, it shall send a request for those data back to the SMLC and then the SMLC can send the required data and expect an successful response from the MS. However, in this work the author had a different approach in that sense, that first all the RRLP assistance data were sent and then the RRLP position request. This way, sending all assistance data, was choosen over the other idea of waiting for the MS response because in OpenBSC it was not possible to access directly the response data without querying the database directly. Since this system is a real time system, waiting for the database to respond may have corrupted the state machine of the GSM network and this would led to the malfunction and eventually failure of the complete network!

The structure of the RRLP messages (requests, assistance data and response) is well defined using Abstract Syntax Notation One (ASN.1) in the technical specifications 3GPP 04.31 and ETSI TS 144 031 [5] [27]. ASN.1 is a conventional notation for denoting the abstract syntax of data used inside of protocols or data structures [72, Chapter 8] [46]. In other words, using ASN.1 it is possible to describe data in an indepedent representation of programming languages in which a protocol is implemented. In this section, only some of the mostly important and used parts of the RRLP protocol inside of the thesis shall be presented, more details can be found in the technical specifications [5] [27]. Structure of the RRLP message encoding for transmission can be seen in listing 4.1. Further details on some of the unknown terms are given in listings 4.2 and 4.3. An example how to build an RRLP request packet shall be given, then it shall be encoded using Packed Encoding Rules (PER). PER is one of the telecommunication standards used for encoding and decoding messages inside of protocols specified in the ASN.1 notation [47].

Listing 4.1: **Structure of the RRLP message in ASN.1**

```
RRLP-Messages
-- { RRLP-messages }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN
IMPORTS
  MsrPosition-Req, MsrPosition-Rsp, AssistanceData,
  ProtocolError
FROM
  RRLP-Components -- { RRLP-Components }
;

PDU ::= SEQUENCE {
  referenceNumber INTEGER (0..7),
  component RRLP-Component
}

RRLP-Component ::= CHOICE {
  msrPositionReq MsrPosition-Req,
  msrPositionRsp MsrPosition-Rsp,
  assistanceData AssistanceData,
  assistanceDataAck NULL,
  protocolError ProtocolError,
  ...,
  posCapabilityReq PosCapability-Req,
  posCapabilityRsp PosCapability-Rsp
}
END
```

Listing 4.2: **Structure of the RRLP request in ASN.1**

```
-- Measurement Position request component

MsrPosition-Req ::= SEQUENCE {
  positionInstruct PositionInstruct,
  referenceAssistData ReferenceAssistData OPTIONAL,
  msrAssistData MsrAssistData OPTIONAL,
  systemInfoAssistData SystemInfoAssistData OPTIONAL,
  gps-AssistData GPS-AssistData OPTIONAL,
  extensionContainer ExtensionContainer OPTIONAL,
  ...,
  -- Release 98 extension element
  rel98-MsrPosition-Req-extension Rel98-MsrPosition-Req-Extension OPTIONAL
}
```

PER is intended for use in circumstances where minimizing the size of the representation of values is the major concern in the choice of encoding rules [47]. In other words, it compresses the data in the PDU packets by limiting the bit field length to the minimal amount of bits required to define the minimal and maximal variable values defined in the standard. There are two variations of PER, aligned and nonaligned [47]. In the RRLP protocol the nonaligned type of PER is used. The major difference between aligned and nonaligned PER lies in the fact that some data structures are

aligned on octet boundaries in aligned PER, i.e. there are some wasted padding bits which are set to zero if not used according to the size of the packets.

Before proceeding with an example, summary for the used ASN.1 type elements shall be provided otherwise it is not possible to proceed with an example RRLP request. A type of **SEQUENCE** is used to reference a "fixed, ordered list of types (some of which may be declared to be optional); each value of the sequence type is an ordered list of values, one from each component type" [47] where the IE (fields) of **OPTIONAL** type do not need to be included and are not mandatory. Variables defined by **CHOICE** are used to reference "a list of distinct types; each value of the choice type is derived from the value of one of the component types" [47], i.e. only one element is selected from the list and the elements of the list are defined according to their variable type. Variables of type **ENUMERATED** are "simple types whose values are given distinct identifiers as part of the type notation" [47], these types are used to distinguish a choice by identifying it with an incremented number from the previous element where the first element is of value zero. Variables defined by **INTEGER** are of the "simple type with distinguished values which are the positive and negative whole numbers, including zero (as a single value)" [47].

At this point the meaning of RRLP data elements marked in red, blue and orange from listing 4.1 shall be given. To construct an RRLP PDU sequence (packet) these fields need to be known: *referenceNumber* and *RRLP-Component*. **referenceNumber** specifies the reference number of the request and is used for the purpose of identifying the response from the MS. It can take any value between 0 and 7, in PER enocoding this requires at least a three bit representation since with three bits, eight different values can be represented ($2^3 = 8$). **component** is of the type RRLP-Component, which is a CHOICE list. RRLP-Component is used for defining what type of information the packet shall include (assistance data, request, response, error, etc.). For this particular example one chooses **msrPositionReq** that is of type MsrPosition-Req (marked in orange in listing 4.1), with this information the MS shall know that its position is requested. MsrPosition-Req is a SEQUENCE, consisting out of one mandatory and few optional IE elements. One choice shall be only considered, **PositionInstruct**, the rest is used later for the assistance data and what type of information is included inside of the PDU message. **PositionInstruct** consists of five elements but four are mandatory: *methodType*, *positionMethod*, *measureResponseTime* and *useMultipleSets*. These four elements are the most compact representation of an inquiry for the MS to differentiate between all the possible position measurements it could perform, how long (time duration) it is allowed to measure the position and how many positions it should perform and return in its response.

methodType defines where the position estimation calculation ought to be executed, shall it take place solely on the MS (*msBased*), solely on the server[3] (*msAssisted*), or one is prefered over the other depending if the MS can execute the prefered one (*msBasedPref* or *msAssistedPref*). The uncertainty of the accuracy of the estimated position is only optional if the choosen method is *msAssisted*, otherwise it must be included in the message.

Listing 4.3: **Structure of the data types from RRLP request in ASN.1**

```
-- Position instructions
PositionInstruct ::= SEQUENCE {
  -- Method type
  methodType MethodType,
  positionMethod PositionMethod,
  measureResponseTime MeasureResponseTime,
  useMultipleSets UseMultipleSets,
  environmentCharacter EnvironmentCharacter OPTIONAL
}

--
MethodType ::= CHOICE {
  msAssisted AccuracyOpt, -- accuracy is optional
  msBased Accuracy, -- accuracy is mandatory
  msBasedPref Accuracy, -- accuracy is mandatory
  msAssistedPref Accuracy -- accuracy is mandatory
}

-- Accuracy of the location estimation
AccuracyOpt ::= SEQUENCE {
  accuracy Accuracy OPTIONAL
}

-- The values of this field are defined in 3GPP TS 03.32 (Uncertainty code)
Accuracy ::= INTEGER (0..127)

-- Position Method
PositionMethod ::= ENUMERATED {
  eotd (0),
  gps (1),
  gpsOrEOTD (2)
}

-- Measurement request response time
MeasureResponseTime ::= INTEGER (0..7)

-- useMultiple Sets, FFS!
UseMultipleSets ::= ENUMERATED {
  multipleSets (0), -- multiple sets are allowed
  oneSet (1) -- sending of multiple is not allowed
}
```

$$r = 10((1.1)^K - 1) \tag{4.1}$$

$$MeasureResponseTimeBitValue = \frac{ln(N)}{ln(2)} \tag{4.2}$$

[3] With server the BTS location is ment!

This uncertainty of the accuracy, is an integer number, that defines how certain the accuracy of the returned position ought to be. It can be calculated using the equation (4.1), where K is the seven bit integer number and r is the accuracy uncertainty in meters [6]. The next three parameters to be defined are the position estimation technique (GPS, E-OTD or one of the two prefered by the MS), the position measurement time and how many measurements the MS ought to report back to SMLC. Since in this thesis the author exploits the AGPS method, GPS is choosen for **PositionMethod**. **MeasureResponseTime** is a three bit integer value that corresponds to the time the MS is allowed to perform the position estimation and to send a respose back to SMLC. Otherwise, if it takes longer the MS than the specified time period, it shall disconnect the SDCCH channel without responding back. It can be calculated using the equation given in (4.2), where N is the number of seconds the MS is allowed to perform the position estimation.

After the ASN.1 parameters of an RRLP request have been understood, they can be choosen and set according to the position measurement request which the network operator wants to perform. In the next step the RRLP request can be constructed and encoded using PER. To construct the RRLP request query from the above given ASN.1 specifications is straightforward. The choosen values are only concatenated into a binary string. A simple RRLP request in PER encoded form is shown in figure 4.2 and the previous conversion process might become more clear, different variables have been colored with distinguishable colors to its neighbor variables so that it is easy to recognize different variables. The five red zeros define what type of data shall be included in the current RRLP packet. This becomes more understandable by looking at the listing 4.4. After the concationation it can be converted to the desired notation system (binary, hexadecimal, etc.). In this particular example the RRLP request to be sent using the RRLP protocol in hexadecimal notation is: **400178F8**. This message is transmitted to the MS via the opened SDCCH channel. However, before sending this request the assistance data can be sent. In the folowing section 4.2 more details of how assistance data are sent shall be presented.

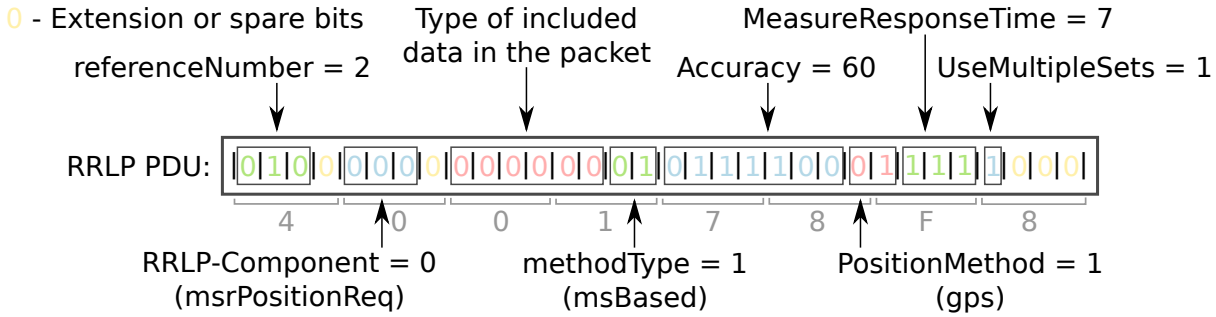

Figure 4.2.: An example RRLP request. Constructing a binary RRLP request in PER from ASN.1. Yellow zero bits are extension markers or spare bits. Image courtesy of [37].

Listing 4.4: Encoding an RRLP request from ASN.1 to PER

```
             RRLP Message:
40 010.....   referenceNumber = 2
              component:
   ...0....    Extension of RRLP-Component = 0 :Absent
   ....000.    RRLP-Component = 0 :msrPositionReq
                MsrPosition-Req:
   .......0     Extension of MsrPosition-Req = 0 :Absent
01 0.......     referenceAssistData = 0 :Absent
   .0......     msrAssistData = 0 :Absent
   ..0.....     systemInfoAssistData = 0 :Absent
   ...0....     gps-AssistData = 0 :Absent
   ....0...     extensionContainer = 0 :Absent
                PositionInstruct:
   .....0..      environmentCharacter = 0 :Absent
                 MethodType:
   ......01      MethodType = 1 :msBased
                  Accuracy:
78 0111100.       Accuracy = 60
                 PositionMethod:
   .......0      PositionMethod = 1 :gps
F8 1.......
                 MeasureResponseTime:
   .111....      MeasureResponseTime = 7
                 UseMultipleSets:
   ....1...      UseMultipleSets = 1 :oneSet
   .....000  Spare Bits = 000b
```

4.2. RRLP Assistance data

Assistance data are of the most important value when it comes to RRLP response time. If the assistance data are present, the response time ought to be shorter since the AGPS receiver knows the orbital information of the satellites and the exact time which allows the AGPS to find immediatelly the Doppler frequency and phase shift of the visible GPS satellite. In the assistance data packets, same as in the request packet, one has to specify what type of assistance information are included in the RRLP assistance packets. In this thesis, as assistance data only the almanac, ephemeris, UTC model, ionospheric model and reference location are transmitted to the MS. There are also other assistance data like differential GPS corrections (DGPS), real time integrity, acquisition assistance and reference time but none of these were available to the author, so they were avoided. The reasons for this decision were explained earlier as cost and complexity issues.

DGPS corrections give additional accuracy to the GPS measurement reports between 1 – 3 m [49] and they are rarely used [37, Chapter 4]. Real time integrity is a list of satellites the AGPS receiver ought not use because these satellites have an integrity problems [37, Chapter 4]. This helps the AGPS receiver to know why assistance data for satellites with errors are not included in the ephemeris or almanac and to be aware not to use instead the old cached data [37, Chapter 4]. Reference time can be considered as the "most valuable" data for the MS because it provides "exact time" to the AGPS receiver. This helps the AGPS receiver to better estimate the phase shift required to detect GPS signal which in return allows the AGPS receiver better integration period τ to detect weaker signals in cities or buildings (see section 2.2 and page 23 for better understanding) [37, Chapter 4]. By knowing the reference time, it is straightforward for the AGPS receiver to predict the TLM and HOW starting words of the transmitted GPS packets from the satellites (refer to section 2.1). Reference time data can include the relationship between the GSM network and GPS time, as well as the GSM frame number to help the MS to synchronize with the BTS (earlier this was described as time synchronized GSM networks which is required for methods like E-OTD) [37, Chapter 4]. Acquisition assistance data, as the name itself says provides the AGPS receiver directly with acquisition data. Acquisition data are the Doppler frequencies and phase shift precalculated on the BTS for the MS. If this type of data is provided to the AGPS receiver, it does not require to compute and search for the given data from the provided time on its on [37, Chapter 4]. This would speed up the process of getting a position and would help weak signals to be detected which in return would minimize the reception errors.

More information on the assistance data transmitted within the RRLP protocol in this work shall be presented here. As listed above, almanac, ephemeris, UTC

model, ionospheric model and reference location are transmitted to the MS. Reference location is the location of the BTS and provides the MS with an approximate location which can be used for the position determination in equations given in section 2.3. Furthermore, this limits the search space in time and frequency domain for satellites to lock on, since if the AGPS receiver has access to these data it can not expect to see satellites which send signals on the opposite side of the Earth [37, Chapter 4]. With the reference location, one sends also the altitude and uncertainty of the included location data so that the AGPS receiver inside the MS can determine and limit the time and frequency search space even further. The ionospheric model includes data for correcting errors introduced by the radio wave transmission through the ionosphere [37, Chapter 4]. These data are not satellite dependent therefore they are not sent for each satellite seperately but once and they are valid for all satellites [37, Chapter 4]. Navigation data in RRLP terminology are the ephemeris data. The transmitted assistance data can be seen in the following tables 4.1, 4.2, 4.3 and 4.4, on the following pages 59, 60 and 61. How other data are encoded shall be given in the implementation chapter, chapter 5.

The packets are constructed in the same manner as RRLP requests with a slight difference of selecting different RRLP components and including assistance data. In this particular example, only a packet with the reference location shall be presented, a "complete" 211 bytes PDU packet constructed by author's software would require at least four pages to be shown. Instead of RRLP request (*msrPositionReq*) in **RRLP-Component** one has to choose assistance data (*assistanceData*) (for the purpose of better understanding in this listening different colors have been used, this particular difference was bolded in listing 4.6). Afterwords one needs to specify what type of assistance the packet includes, in this case it is GPS assistance data (*gps-AssistData*, marked in red color in listing 4.6). GPS assistance data were described in the two previous paragraphs and therefore shall be omitted here. They shall be only listed in the order as specified in the RRLP standard for GPS assistance data, listing 4.5: reference time, reference location, DGPS corrections, navigation model, ionospheric model, UTC model, almanac, acquisition assistance and real time integrity (all marked with blue color in listing 4.6). The assistance data one wants to include in the RRLP packet have to be selected previously. Selecting is straightforward and one only is required to set the appropriate bit to one (1=included in the packet, 0=not included in the packet). Since in this example only the reference location is transmitted inside the RRLP PDU packet, the *refLocation* bit is set to one. Once the variables have been set, the assistance data have to follow the given order as in listing 4.5. The top variable data (*referenceTime*) would follow as first and bottom variable (*realTimeIntegrity*) would be the last to be included in the RRLP assistance PDU packet. The reference location has to be converted into an ellipsoid point with altitude and uncertainty ellipsoid as described in the standard [6] under section *7.3.6*,

as shown in figure 4.3.

Listing 4.5: **Structure of data types of GPS assistance data in ASN.1**

```
-- Control header of the GPS assistance data
ControlHeader ::= SEQUENCE {

  -- Field type Present information
  referenceTime ReferenceTime OPTIONAL,
  refLocation RefLocation OPTIONAL,
  dgpsCorrections DGPSCorrections OPTIONAL,
  navigationModel NavigationModel OPTIONAL,
  ionosphericModel IonosphericModel OPTIONAL,
  utcModel UTCModel OPTIONAL,
  almanac Almanac OPTIONAL,
  acquisAssist AcquisAssist OPTIONAL,
  realTimeIntegrity SeqOf-BadSatelliteSet OPTIONAL
}
```

The reference location consists of longitude, latitude, altitude, uncertainty semi-major, uncertainty semi-minor, orientation of major axis, uncertainty of altitude and confidence level. **S** is sign of the latitude, it is set to zero if it is North and one if it is South. **D** is the altitude direction, it is set to zero if the altitude that follows is height and to one if it is depth. Uncertainty semi-major and uncertainty semi-minor are uncertainties for longitude and latitude. Orientation of major axis is the orientation angle of the BTS between the major axis and North pole in degrees. These terms are depicted in figure 4.4 by showing the World Geodetic System 1984 (WGS84). The latitude, longitude and altitude need to be encoded into a format recognized by the RRLP standard. This is straightforward and can be proceeded using the equations shown in (4.3), where φ is the latitude and λ is the longitude value in decimal degrees. Longitude is encoded as second compliment binary number [6]. The altitude is encoded as it is where one bit increments represent one meter incerements. The uncertainties for latitude, longitude and altitude are encoded using the equation given in (4.4), where r is the uncertainty in meters for latitude and longitude, and h is the uncertainty in meters for altitude of the BTS. Both values, U_L and U_A, are 7 bit numbers in the range between 0 and 127. Orientation of major axis is not used in this work so it was set to zero. Confidence describes the level by which the sent BTS reference position is known to be correct. The confidence is a 7 bit number but ought to take values between 0 and 100 since it represents the percentage. In this work it was set to zero, i.e. no information is available about the confidence for our reference location. It did not change the output behaviour from the MS.

If the reference location is included in the RRLP assistance packet, it is important to specify the octet length of the reference location. The length of the reference location of an ellipsoid point with altitude and uncertainty ellipsoid is of length 14

8	7	6	5	4	3	2	1
1	0	0	1	Spare bits			
S							
Degrees of latitude							
Degrees of longitude							
D							
Altitude							
0	Uncertainty semi-major						
0	Uncertainty semi-minor						
Orientation of major axis							
0	Uncertainty Altitude						
0	Confidence						

Figure 4.3.: Reference location is a 14 octet stream built according to the given rule as specified in the standard [6] under section *7.3.6*. Image courtesy of [6].

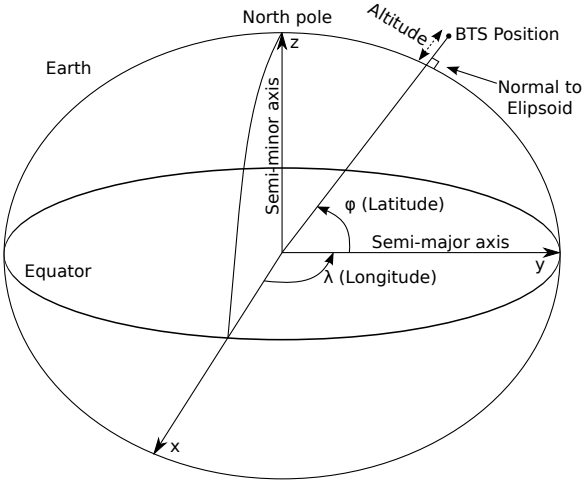

Figure 4.4.: World Geodetic System 1984. Image courtesy of [37].

octets, as it can be seen in figure 4.3 (amount of rows), it is written as 13 octets in

the RRLP PDU packet. It is always specified as one number less since at least one octet has to be included in the reference location. There are other reference location standards inside of the RRLP protocol. This way the RRLP protocol knows where the data end and where new data may start if they are included. What type of reference location is include is defined by the first four bits of the reference location, in this case it is 1001, as it can be seen in figure 4.3. This is an additional mechanism for error control, if the numbers do not match when the transmitted binary data have been decoded then the MS can return an error and ask for retransmission of the data. Information related to the reference location in the example listing 4.6 are marked with orange color. Once the assistance data have been transmitted to the MS, it shall respond back with an acknowledgement or error depending if the data were correctly received and parsed by the MS. The acknowledgement shall have the same reference number as the assistance packet. This can be seen as well in figure 4.1. In the next section more details shall be given on the RRLP response from the MS.

$$\begin{aligned} Lat &= \frac{2^{23}}{90} \cdot |\varphi| \quad \Longleftarrow \quad \text{Latitude} \\ Long &= \frac{2^{24}}{360} \cdot \lambda \quad \Longleftarrow \quad \text{Longitude} \end{aligned} \tag{4.3}$$

$$\begin{aligned} U_L &= \left\lceil \frac{ln(\frac{r}{10}+1)}{ln(1.1)} \right\rceil \Big| U_A \in [0,127] \quad \Longleftarrow \quad \text{Uncertainty for latitude and longitude} \\ U_A &= \left\lceil \frac{ln(\frac{h}{45}+1)}{ln(1.025)} \right\rceil \Big| U_A \in [0,127] \quad \Longleftarrow \quad \text{Uncertainty for altitude} \end{aligned} \tag{4.4}$$

Table 4.1.: GPS UTC Model content

Field (IE)	Description
A_1	Drift coefficient of GPS time scale relative to UTC time scale
A_0	Bias coefficient of GPS time scale relative to UTC time scale
t_{ot}	Time data reference time of week
Δt_{LS}	Current or past leap second count
WN_0	Time data reference week number
WN_{LSF}	Leap second reference week number
DN	Leap second reference day number
Δt_{LSF}	Current of future leap second count

Table 4.2.: Navigation message (ephemeris) content

Field (IE)	Description
Satellite ID	This is the satellite ID that is in the range of 0 to 63. PRN=SatelliteID + 1
Satellite status	This is an indicator of whether this is a new or existing satellite and whether the navigation model is new or the same.
C/A or P on L2	Code(s) on L2 channel
URA Index	User range accuracy
SV Health	Satellite health
IODC	Issue of data, clock
L2 P Data flag	
SF 1 Reserved	
T_{GD}	Estimated group delay differential
t_{oc}	Apparent clock correction
a_{f2}	Apparent clock correction
a_{f1}	Apparent clock correction
a_{f0}	Apparent clock correction
C_{rs}	Ampltitude of the sine harmonic correction term to the orbit radius (meters)
Δn	Mean motion difference from computed value (semicircles/second)
M_0	Mean anomaly at reference time (semicircles)
C_{uc}	Ampltitude of the cosine harmonic correction term to the argument of latitude (radians)
e	Eccentricity
C_{us}	Amplitude of the sine harmonic correction term to the argument of latitude (radians)
$A^{1/2}$	Square root of semi-major axis (meters)
t_{oe}	Reference time ephemeris
Fit Interval Flag	
AODO	Age of data offset
C_{ic}	Amplitude of the cosine harmonic correction term to the angle of inclination (radians)
Ω_0	Longitude of ascending node of orbit plane at weekly epoch (semicircles)
C_{is}	Amplitude of the cosine harmonic correction term to the angle of inclination (radians)
i_0	Inclination angle at reference time (semicircles)
C_{rc}	Amplitude of the cosine harmonic correction term to the orbit radius (meters)
ω	Argument of perigee (semicircles)
OMEGAdot	Rate of right ascension (semicircles/second)
Idot	Rate of inclination angle (semicircles/second)

Table 4.3.: Almanac message content

Field (IE)	Description
SatelliteID	This is the satellite ID that is in the range of 0 to 63. PRN=SatelliteID + 1
SV Health	Satellite health (e.g. 000 means the satellite is fully operational)
e	"Eccentricity shows the amount of the orbit deviation from circular (orbit). It is the distance between the foci divided by the length of the semi-major axis" [81]
TOA	Time of applicability, reference time for orbit and clock parameters (seconds). "The number of seconds in the orbit when the almanac data were generated" [81]
OI	Orbital inclination (radians). The angle to which the SV orbit meets the equator [81]
RORA	Rate or right ascension (radians/second). "Rate of change of the angle of right ascension as defined in the Right Ascension mnemonic" [81]
$A^{1/2}$	Square root of semi-major axis (meters$^{1/2}$). " This is defined as the measurement from the center of the orbit to either the point of apogee or the point of perigee" [81]
Ω_0	Right Ascension at Week (radians). Longitude of ascending node of orbit plane at weekly epoch
ω	Argument of perigee (semicircles). "An angular measurement along the orbital path measured from the ascending node to the point of perigee, measured in the direction of the SV's motion" [81]
M_0	Mean anomaly (radians)
a_{f0}	Satellite clock bias (seconds). Satellite clock error at reference time
a_{f1}	Satellite clock drift (seconds per second). Satellite clock error rate
Week	Week number since the last reset (i.e. since year 1980 modulo 1024 weeks)

Table 4.4.: GPS Ionosphere Model content

Field (IE)	Description
α_0	Coefficient 0 of vertical delay
α_1	Coefficient 1 of vertical delay
α_2	Coefficient 2 of vertical delay
α_3	Coefficient 3 of vertical delay
β_0	Coefficient 0 of period of the model
β_1	Coefficient 1 of period of the model
β_2	Coefficient 2 of period of the model
β_3	Coefficient 3 of period of the model

Listing 4.6: **Encoding reference location from ASN.1 to PER**

```
             RRLP Message:
44 010.....   referenceNumber = 2
              component(RRLP-Component):
   ...0....    Extension of RRLP-Component = 0 :Absent
   ....010.    RRLP-Component = 2 :assistanceData
                AssistanceData:
   .......0      Extension of AssistanceData = 0 :Absent
11 0.......      referenceAssistData = 0 :Absent
   .0......      msrAssistData = 0 :Absent
   ..0.....      systemInfoAssistData = 0 :Absent
   ...1....      gps-AssistData = 1 :Present
   ....0...      moreAssDataToBeSent = 0 :Absent
   .....0..      extensionContainer = 0 :Absent
                 GPS-AssistData:
                  ControlHeader:
   ......0.        referenceTime = 0 :Absent
   .......1        refLocation = 1 :Present
00 0.......        dgpsCorrections = 0 :Absent
   .0......        navigationModel = 0 :Absent
   ..0.....        ionosphericModel = 0 :Absent
   ...0....        utcModel = 0 :Absent
   ....0...        almanac = 0 :Absent
   .....0..        acquisAssist = 0 :Absent
   ......0.        realTimeIntegrity = 0 :Absent
                   RefLocation:
                    threeDLocation(Ext-GeographicalInformation):
   .......0          Ext-GeographicalInformation length(octet) = 13 :13 + 1 = 14
D9 1101....
   ....1001          Ext-GeographicalInformation = 904445940594B200000707000700h
04 00000100
44 01000100
59 01011001
40 01000000
59 01011001
4B 01001011
20 00100000
00 00000000
00 00000000
70 01110000
70 01110000
00 00000000
70 01110000
00 0000....

   ....0000  Spare Bits = 0000b
```

4.3. RRLP Response

In this section the RRLP response from the MS shall be analysed. The RRLP response is constructed in the same manner as the RRLP request and assistance data by following ASN.1 rules precisely specified in the RRLP standard. RRLP response is produced by the MS itself. It may include the estimated position, data for estimating the position on the BTS (if MS assisted was choosen as the prefered method) or errors indicating that some of the previously stated assistance data are missing. Missing data and errors are specified inside of the RRLP response. The response data shall be PER encoded and require to be decoded into the ASN.1 notation. In listing 4.7 an example of an RRLP response with an error can be seen. The location error bit is set if the location of the MS is not present within the message (marked in red). The MS may sometimes supply more information on the error if the MS knows this information (newer models support this). In case it does support more information, it shall set an optional IE *additionalAssistanceData* bit (marked in cyan).

Listing 4.7: **Decoding an error RRLP response from Samsung Galaxy S3**

```
             RRLP Message:
42 010.....   referenceNumber = 2
              component(RRLP-Component):
   ...0....    Extension of RRLP-Component = 0 :Absent
   ....001.    RRLP-Component = 1 :msrPositionRsp
                MsrPosition-Rsp:
   .......0     Extension of MsrPosition-Rsp = 0 :Absent
04 0.......     multipleSets = 0 :Absent
   .0......     referenceIdentity = 0 :Absent
   ..0.....     otd-MeasureInfo = 0 :Absent
   ...0....     locationInfo = 0 :Absent
   ....0...     gps-MeasureInfo = 0 :Absent
   .....1..     locationError = 1 :Present
   ......0.     extensionContainer = 0 :Absent
                LocationError:
   .......0      Extension of LocationError = 0 :Absent
99 1.......      additionalAssistanceData = 1 :Present
                 LocErrorReason:
   .0......       Extension of LocErrorReason = 0 :Absent
   ..0110..       LocErrorReason = 6 :gpsAssDataMissing
                 AdditionalAssistanceData:
   ......0.      Extension of AdditionalAssistanceData = 0 :Absent
   .......1      gpsAssistanceData = 1 :Present
0B 0.......      extensionContainer = 0 :Absent
                 GPSAssistanceData:
   .000101.       GPSAssistanceData length(octet) = 5 :5 + 1 = 6
   .......1       GPSAssistanceData = E80000000000h
D0 11010000
00 00000000
00 00000000
00 00000000
00 00000000
00 0000000.
   .......0  Spare Bits = 0b
```

This is followed with a more detailed explanation of the error that not sufficient assistance data were present, *LocErrorReason* (marked with blue color). There are other possible location error reasons as well and they are listed in listing 4.8. Depending on the MS model, it can even further specify what kind of GPS assistance data are missing. This shall be well specified by setting the IE *gpsAssistanceData* bit, this is shown in listing 4.7 (marked with magenta color). If this bit is set, the length of the IE for requested missing assistance data shall be exactly specified as well as what assistance data are missing (marked in orange color).

Listing 4.8: **Possible location error reasons**

```
LocErrorReason ::= ENUMERATED {
  unDefined (0),
  notEnoughBTSs (1),
  notEnoughSats (2),
  eotdLocCalAssDataMissing (3),
  eotdAssDataMissing (4),
  gpsLocCalAssDataMissing (5),
  gpsAssDataMissing (6),
  methodNotSupported (7),
  notProcessed (8),
  refBTSForGPSNotServingBTS (9),
  refBTSForEOTDNotServingBTS (10),
  notEnoughGANSSSats (11),
  ganssAssDataMissing (12),
  refBTSForGANSSNotServingBTS (13)
}
```

8	7	6	5	4	3	2	1
A	B	C	D	E	F	G	H
0	0	0	0	0	K	J	I

Figure 4.5.: Requested AGPS assistance data to be delivered

The first two bytes of the IE *GPSAssistanceData* contain the information for requested assistance AGPS data (marked in orange color). They can be seen in figure 4.5 [5]. If one of these bits from A to K is set, the MS requires more assistance data. The meaning of the bits in figure 4.5 is given in table 4.5. In this particular example, the first two bytes are: **E800**, indicating acquisition assistance, reference time, reference location and the navigation model are requsted by the MS as assistance data. The next RRLP response example, shown in listing 4.10, is a response with a successfully estimated position! A successful or erroneous position response is of RRLP measurement responses type, this can be seein in listing 4.10 (bolded, in listing 4.7 it is bolded as well). It can not be distinguished by analysing the first byte of the response stream! In the second byte, two mutually exclusive IE contain the information if the response contains the location information or not, *locationInfo* bit must be set and *locationError* must be unset (both marked in red color in

listing 4.10). If the IE *locationInfo* bit is one and *locationError* bit zero, then the position of the MS is included in the response. Aside from the position information, the time when the position measurement was performed is included as well however, only the least significant bits in the range of miliseconds. The most significant bits shall be derived by the SMLC using the GSM frame number, included in the IE *refFrame*. *refFrame* contains the GSM frame number as observed by the MS without the TA factor taken into account [5]! The time of miliseconds can be found in the IE *gpsTOW*. The included time is not in UTC format and would require additional conversions. The elements of *locationInfo* can be seen in listing 4.9. The IE *fixType* contains the information if the performed measurement was 2D or 3D.

Table 4.5.: Requested AGPS assistance data bit meaning

Bit (IE)	Description
A	Acquisition assistance requested
B	Reference time requested
C	Reference location requested
D	DGPS corrections requested
E	Navigation model requested
F	Ionospheric model requested
G	UTC model requested
H	Almanac data requested
I	Real time integrity requested
J	Ephemeris extension requested
K	Ephemeris extension check requested

Listing 4.9: **Structure of data types of location info data in ASN.1**

```
-- Location information IE
LocationInfo ::= SEQUENCE {
  refFrame INTEGER (0..65535), -- Reference Frame number
  -- If refFrame is within (42432..65535), it shall be ignored by the receiver
  -- in that case the MS should provide GPS TOW if available
  gpsTOW INTEGER (0..14399999) OPTIONAL, -- GPS TOW
  fixType FixType,
  -- Note that applicable range for refFrame is 0 - 42431
  -- Possible shapes carried in posEstimate are
  -- ellipsoid point, ellipsoid point with uncertainty circle,
  -- ellipsoid point with uncertainty ellipse,
  -- ellipsoid point with altitude and uncertainty ellipsoid
  posEstimate Ext-GeographicalInformation
}

FixType ::= INTEGER {
  twoDFix (0),
  threeDFix (1)
} (0..1)
```

The position information is extracted with the inverse process as it was specified for the reference location. Equations to return from the bit format to decimal degrees are given in equation (4.5). In the next chapter, more details shall be given on the implementation of the complete system.

Listing 4.10: **Decoding a successful RRLP response from iPhone 3GS**

```
            RRLP Message:
42 010.....  referenceNumber = 2
             component(RRLP-Component):
   ...0....   Extension of RRLP-Component = 0 :Absent
   ....001.   RRLP-Component = 1 :msrPositionRsp
               MsrPosition-Rsp:
   .......0    Extension of MsrPosition-Rsp = 0 :Absent
11 0.......    multipleSets = 0 :Absent
   .0......    referenceIdentity = 0 :Absent
   ..0.....    otd-MeasureInfo = 0 :Absent
   ...1....    locationInfo = 1 :Present
   ....0...    gps-MeasureInfo = 0 :Absent
   .....0..    locationError = 0 :Absent
   ......0.    extensionContainer = 0 :Absent
               LocationInfo:
   .......1     gpsTOW = 1 :Present
FF 11111111     refFrame = 65535
FF 11111111
61 01100001     gpsTOW = 6399000
A4 10100100
18 00011000
               FixType:
B6 1.......     FixType = 1 :threeDFix
               posEstimate(Ext-GeographicalInformation):
   .01101..     Ext-GeographicalInformation length(octet) = 13 :13 + 1 = 14
   ......10     Ext-GeographicalInformation = 904445840594A6016316114F1D44h
41 01000001
11 00010001
16 00010110
10 00010000
16 00010110
52 01010010
98 10011000
05 00000101
8C 10001100
58 01011000
45 01000101
3C 00111100
75 01110101
10 000100..
   ......00  Spare Bits = 00b
```

$$\varphi = \frac{90}{2^{23}} \cdot Lat \quad \Longleftarrow \quad \text{Latitude in decimal degrees}$$
$$\lambda = \frac{360}{2^{24}} \cdot Long \quad \Longleftarrow \quad \text{Longitude in decimal degrees} \tag{4.5}$$

5. Implementation

The aim of this chapter is to give the reader a review of the employed hardware and the software implementation. The main idea of author's approach to the problem is discussed in this chapter. The implementation can be divided into two stages. The first stage being the inital phase of the thesis where the initial system has been set up to perform RRLP tests. The second stage can be divided into two implemantation parts. The first part of the second stage consists of the development of the application that generates RRLP assistance data. The second part of the second stage consists of modifying the existing open source GSM software and implementing the procedures for creating a data channel between the BTS and MS. This channel was deployed for the transmission of assistance data to the MS and for obtaining the response from the MS.

5.1. Initial phase

Traditionally all radio communication systems are hard wired and the hardware is developed to do only one fixed function as the nanoBTS, to serve as a BTS. nanoBTS is a dedicated BTS hardware, used to set up the GSM network with OpenBSC (more details on the nanoBTS can be found in the hardware description). However, at the start of the thesis, the author had no access to the nanoBTS. On the other hand, instead of the nanoBTS a software defined radio (SDR) platform was available and used to emulate the GSM network. SDR is a hardware platfrom that enables the development and test of different radio communication systems and protocols using software that modifies the function of the hardware. In other words, the hardware may perform different functions in the range of its specified limitations. Those limitations can be the frequency on which the SDR can transmit and receive radio waves, the speed of sampling a radio wave signal and other properties. The basic idea is to use the fast performace of a CPU from the computer to do the software signal processing while the SDR hardware itself performs only the physical radio communication like emitting and receiving radio waves. Alternatively to the dedicated hardware, SDRs can be programmed to perform various functions e.q. an FM radio, a GPS receiver,

GSM and etc., all of them employing different modulation/demodulation procedures and frequency spectrums [11] [69]. Theoretically "anything" can be built using an SDR platform that is within the domain of the SDR hardware.

The exploited SDR platform in this thesis was the Universal Software Radio Peripheral (USRP) by Ettus Research. USRP had already a GSM and RRLP software implementation. The GSM network software used on USRP was OpenBTS, a Linux application written in C++ employing the SDR platform to provide a GSM air interface [69]. Once the system has been successfully configured and set in operation it was followed by tests to verify if it was operating correctly. Initially, the system was tested with 2G cell phones (Nokia 3310 and Siemens M50) and its correctness was verified. While the system was tested with smart phones, a strange behaviour could be noticed. Sometimes the smart phones (*iPhones* 3*GS* and 4) could not detect existance of the GSM network at all, i.e. the network could not be found in the search menu where all GSM networks in range are shown. The reason for this strange phenomenon may be found in the unstable operation of the cheap clock oscillator. Although the clock unstability issue can not be confirmed by the author due to the missing hardware equipment to measure the actual frequency and its deviation. Nevertheless, these results were consistent with the results of the OpenBTS developers with similar clock issues[1]. As previously stated in the GSM chapter, the clock oscillator for the BTS is not allowed to deviate more than ±5 ppm (parts per million). This finding, that older cell phones like Nokia 3310 and Siemens M50 have rather less problems connecting to the GSM network than the newer cell phones suggest that newer generation cell phones are not robust and resistant to the timing deviation issues. Meanwhile the RRLP module was downloaded and installed. The module was written by Kurtis Heimerl in two different programming languages, Erlang and Common Gateway Interface (CGI)[2]. Once the RRLP module was configured and installed, the new GMS system configuration was examined. The first observation and finding was that not a single smart phone could connect to the GSM network. In the log files it could be seen a time out was triggered by OpenBTS. This timeout was triggered while the smart phones tried to get a position fix after the RRLP request was delivered to the MS. This result may be explained by analysing at what stage in the protocol the RRLP request was sent. The RRLP request was immediatelly sent after the paging request has been obtained by the MS. This evidence justifies the time out behaviour. Once the option for sending RRLP requests while the paging is in progress was disabled, this problem was solved! Next step was to manually send the RRLP requests from the OpenBTS terminal to smart phones. Contrary to expectations, the smart phones sometimes received the RRLP request as an SMS message

[1] GSM not detecting station, USRP1, FA-SY1, WBX, DBS http://www.ruby-forum.com/topic/1876696

[2] Kurtis Heimerl's code can be found on https://github.com/ttsou/RRLP

and did not provide any response. In the case where the smart phones did not receive the RRLP request as an SMS message, still no response was produced. One of the consequences of such behaviour was that the RRLP could not be tested inside of this set up because the system itself was unstable and had an unpredictive behaviour. The conducted tests with OpenBTS thus lead to a logical decision to employ dedicated BTS hardware with a tested and calibrated clock oscillator only for GSM. On the other hand, the Erlang RRLP module was a starting point to understand the RRLP protocol. The generated assistance data packets by the module were used for comparison and a template to build author's RRLP assistance data generator. The nanoBTS is operated by OpenBSC which is explained in the following section.

5.2. OpenBSC and its original RRLP implementation

OpenBSC is an open source implementation of a GSM network software by Osmocom. It was developed for experimentation and security research of the GSM networks [66]. OpenBSC is "implementing the minimal necessary parts to build a small, self-contained GSM network" [65]. This self-contained GSM network consists of following functional components: Base Station Controller (BSC), Mobile Switching Center (MSC), Home Location Register (HLR),Authentication Center (AUC), Visitor Location Register (VLR) and Equipment Identity Register (EIR). OpenBSC was written in C and operates on Linux. OpenBSC binds to the BTS using the Abis or Abis/IP interface. At the moment OpenBSC supports Voice calls, SMS, handovers, support for multiple BTS and other features not of the interest for this work. OpenBSC has an implemented module for transmitting RRLP requests however without assistance data. This module was tested but without successfully obtaining a position from the MS. While the tests have been performed, no results were obtained due to a watchdog time out produced by OpenBSC. In order to send an RRLP request in OpenBSC, a silent SMS would be sent to the cell phone followed by the RRLP request. Silent SMS is the equivalent of a normal SMS but without notifying the user of its reception [74]. When the silent SMS is received on the cell phone, the message content is not displayed to the user neither is it stored in the SMS inbox. In other words, its arrival remains completely unknown to the user to whom it was sent [74]. An acknowledgement is sent back to the GSM network operator that the MS received the silent SMS. The watchdog timer in OpenBSC has been triggered because the acknowledgement was not received within a certain time limit while the MS was attempting to obtain a GPS position. To overcome this problem another approach had to be taken by the author to send RRLP assistance data with position requests. This shall be further analysed and explained in more details in the following sections.

5.3. RRLP assistance data generator

At the point of working on this thesis, two different RRLP implementations on two different hardware platforms have been examined without successfully obtaining a GPS localization. The next step in the attempt to obtain GPS positions from MS was to gain better understanding of the RRLP protocol and RRLP assistance data. The RRLP assistance data generated by Heimerl's application did not produce valid assistance data. In order to publish the RRLP assistance data generator as open source to be wider extended or ported to another programming language, it was required to be written in a programming language understandale by wider audience. It was sound to write the RRLP assistance data generator in C++ because OpenBSC was written in C and OpenBTS in C++. The main tasks here can be split up down to: verify the existance and age of assistance data download of the assistance data, conversion of the data, verification of their correctness, construction of RRLP packets according to the ASN.1 standard, conversion of it to PER and at last saving in the hexadecimal form in a text file.

In the almanac and ephemeris files, downloaded from NASA and Trimble, assistance data were stored for 32 different GPS satellites. Contrary to expectations after the generated RRLP packets have been analysed, Heimerl's code produced RRLP assistance data packets with only valid data for one satellite but duplicated 32 times. At this stage, it was important to have a fully working RRLP assistance data generator. This generator would be subsequently used to examine the RRLP protocol once OpenBSC was modified to open a data channel for transmitting the assistance data with the RRLP position request. Heimerl's code, written in Erlang, had to be fully understood to build a working replica RRLP assistance data generator. It was used as a template with the specified RRLP conversion standard itself. The downloaded assistance data were in formats known as Receiver Independent Exchange format (RINEX) for ephemeris data and Yuma for almanac data. RINEX is a format to exchange raw satellite navigation data from GPS receivers that have an ability to output them. Although almanac data can be in RINEX form as well, the almanac data found online were in the Yuma format. The read files with assistance data had to be first properly parsed and then converted into the format specified in the standard by ETSI TS 144 031 [27]. The data included in the ephemeris file contained also UTC model, and the ionospheric model. Other operational data were included in the configuration file like the reference location data, more details can be found in the software appendix configuration section A.3. Once all data have been successfully parsed and converted, they had to be verified to be in the specified range as in the standard. Afterwards the converted and verified assistance data were combined into binary series of data according to the RRLP standard as described in chapter

4. If the assistance data packet size in binary format was not divisible by eight then additional padding zeros were added until this condition was satisfied. Better comprehension of the RRLP assistance data generator can be gained by looking at the flowchart in figure 5.1.

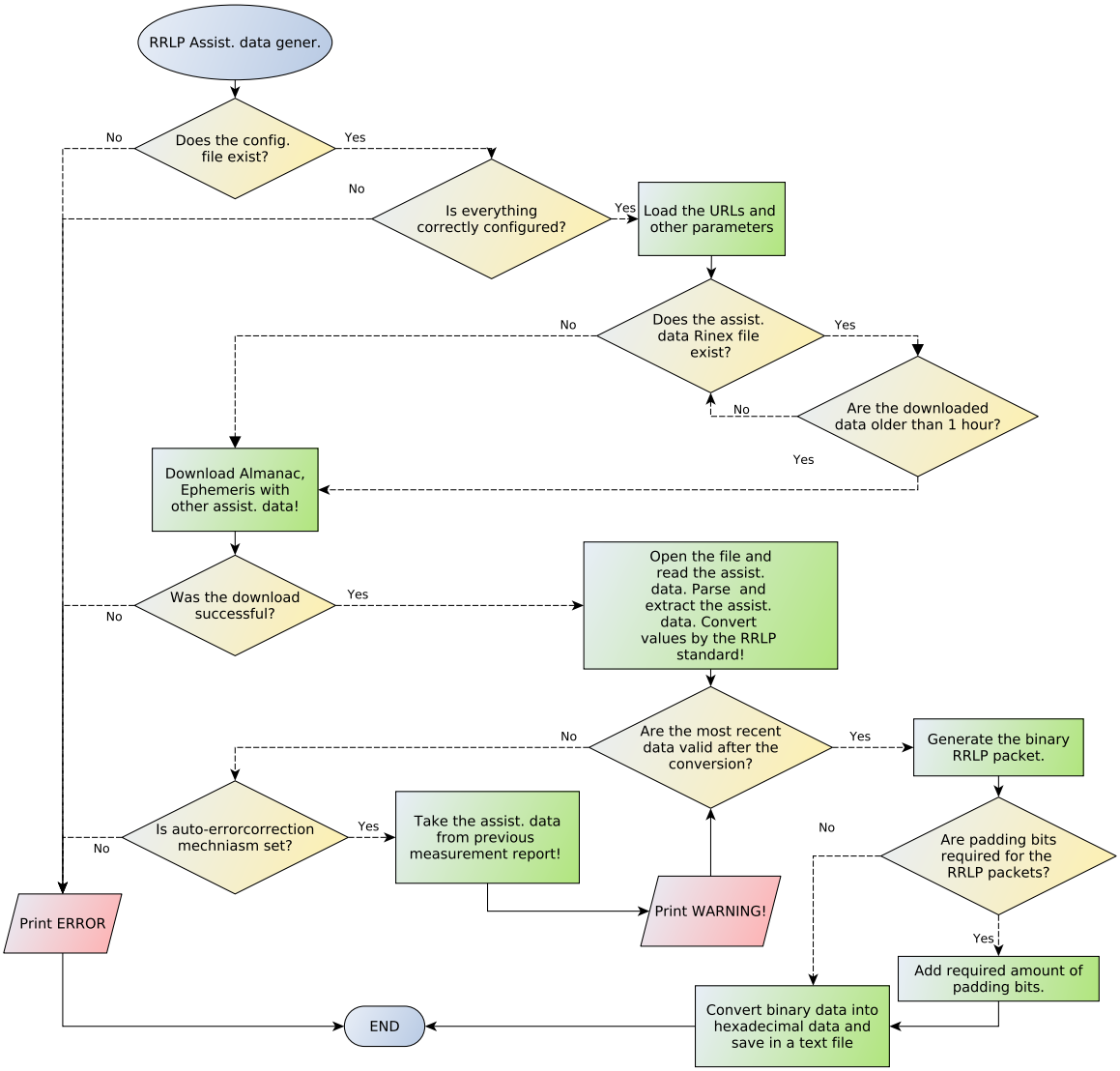

Figure 5.1.: Flowchart for the RRLP assistance data generators

Since the ephemeris data refresh every two hours, the latest generated data are appended at the end of the file for the current reading[3]. It was common that the ephemeris assistance data contained errors which were detected in the data range

[3]This is performed by Trimble, whose ephemeris data were used.

verification step. To avoid disruption in operation of the written software, once data out of range were detected they are immediatelly substituted with data for the same satellite but with two hours older ephemeris data. If the AGPS receiver in the MS uses the ephemeris data from that particular satellite then the distance estimation is affected and may contain errors! This problem is well known and confirmed by different studies [39] [48]. A solution to this problem is proposed in the future work section 7.3. Once the assistance data have been generated, converted to hexadecimal notation and saved to a text file they can be used by OpenBSC to be transmitted to the MS. The decision to save the data to a text file, instead of storing to the database, was made because OpenBSC is a real-time system. If the database does not respond OpenBSC' real-time functionality might be lost and the system will malfunction. Since the text file is small and accessed only when an RRLP request is queued, it is faster than initializing the database driver, opening a socket connection to the database, making request queries to the database, obtaining the result and closing the socket connection. At this step the assistance data are ready to be opened by OpenBSC and sent to the MS.

5.4. Creating a data channel in OpenBSC

To avoid the watchdog time out triggered by OpenBSC when the RRLP requests have been sent originally the solution was to open a data channel. The original idea was to open a data channel with a silent SMS and then to send the RRLP request. The opening of a data channel (SDCCH) and what is on going in OpenBSC can be split up into 4 stages: adding a Virtual Teletype (VTY) interface to execute an RRLP request, open a physical data channel between the BTS and MS, opening the text file with assistance data and sending the data as well as parsing the RRLP packets, waiting for the response back and disconnecting. This will be explained in an sequential order of execution.

The GSM network operator can connect to the VTY interface of OpenBSC using Telnet on port 4242 to issue commands and administer the GSM network. In order to send RRLP requests from the VTY command interface, RRLP execution command had to be integrated. Function that integrates that command is named **`subscriber_silent_rrlp_start()`** and is in the file *vty_interface_layer3.c*. Once the operator executes on of the four implemented commands: RRLP request, RRLP request with almanac data, RRLP request with ephemeris and other assistance data or to end the execution of the RRLP request. For the first three commands, the next executed function is **`gsm_rrlp_operation_start()`** in the file *silent_call.c*. Any of the three commands will iniate the opening of an

SDCCH channel (what in **gsm_rrlp_operation_start()** function takes place was already explained in 3.3). In the case, when the GSM operator can not wait for the RRLP request answer or wants to stop the execution of the RRLP request, he can issue a command in the VTY interface to stop it, which will execute **gsm_rrlp_operation_stop()** in the file *silent_ call.c*. If the channel was successfully opened the function **send_rrlp_req()** will be executed that is responsible for opening the text file with assistance data, copying the assistance data into the RRLP data structure and transmitting it to the MS. The data structure is shown in 5.1. The structure contains the length of the data packets, *lengthOfPacket* as well as the content of the packets, *packetContent* which never exceeds more than 211 bytes.

Listing 5.1: **Data structure containing the RRLP assistance data.**

```
struct rrlpPacket {
  int lengthOfPacket;
  uint8_u packetContent[211];
}
```

After the RRLP assistance data have been successfully loaded into the RRLP data structure, they are transmitted to the MS. In listing 5.2 the function required to send assistance data is shown. The first arguments passed to the function provides the pointer to the opened SDCCH data channel connection. It is followed by the Application Protocol Data Unit identifier set to `0x00`. The third argument is the size of the transmitted packet. The last passed argument to the function is the starting address of the location where the packet content is located. Since there is more than one RRLP assistance packet to be transmitted, this function gets executed a few times in a loop defined by the packet counter variable *packNum*. The responses sent back by the MS are obtained by the BTS and stored in the HLR database table *ApduBlobs* in PER notation. A small utility was programmed to connect to the database, display the acquired data and the decoded position if it is contained in the RRLP response.

Listing 5.2: **Function required to transmit assistance data.**

```
int response = gsm48_send_rr_app_info(conn, 0x00, ↙
    ↳ AlmanacPackets[packNum].lengthOfPacket, AlmanacPacket[packNum].packetContent)
```

5. Implementation

6. Hardware

In the following chapter the author shall introduce the reader to the hardware components used in the thesis. The hardware components shall be presented according to their importance of building an operational and functional GSM network with GPS localization capabilities. Firstly the nanoBTS shall be introduced since it is the main hardware component used for building a basic GSM network infrastructure. Then a short insight into the used GPS receiver shall be given. Additionally the mobile stations used for testing of the system shall be reviewed. Finally, a hardware connection diagram shall be given.

6.1. GSM BTS - nanoBTS

In recent years, there has been an increasing interest in deployment of private cellular networks in remote areas or for research which lead to the devolopment of diverse "low-cost" GSM hardware solutions. According to ip.access[1], the manufacturer of nanoBTS, their hardware product is deployed for coverage of "hard-to-reach places; in-buildings; remote areas; marine and aviation; and public spaces". A nanoBTS with its plastic cover can be seen in Figure 6.1. Our University GSM network consists of three nanoBTS stations. The deployed nanoBTS in author's thesis works in the 1800 MHz frequency range, for which the University of Freiburg had obtained a licence from the Federal Network Agency (German: *Bundesnetzagentur*). The transmission frequencies range between 1805-1880 MHz, with 200 kHz channel spacing and maximal output power of +13 dBm (≈20 mW), whereas the receiving frequencies lie in the range between 1710-1785 MHz and same channel spacing as for transmission of 200 kHz [44].

Check the output powere 20 dBm

Add the Abis over IP protocol

The nanoBTS is equiped with an internal 0 dBi (nominal) omni-directional antenna. However, two external antennas sized 30x36 mm, one for transmission (TX) and the other one for reception (RX) of radio waves were used to extend the coverage area. These antennas are connected via the SMA connectors. By using an RF

[1] http://www.ipaccess.com

Figure 6.1.: nanoBTS with its plastic cover. Image courtesy of ip.access ltd

Check for what NWL is

amplifier and larger antennas, for these frequency ranges, the covered area with the GSM signal reception can be increased. For the gain estimation and radiation angle of the used antennas the measurement equipment was missing and therefore was not conducted and described in this work.

At the bottom of the nanoBTS there are 5 ports, as seen in Figure 6.2. The ports from left to right are: voltage supply, ethernet cable with power supply, USB port, TIB-IN and TIB-OUT. In the next paragraph a brief overview of each port shall be given.

The left most port is the power supply port used for supplying the nanoBTS with 48 V DC and is optionally used depending on the cable configuration. In author's hardware configuration the power supply port is not used. The following port is for the ethernet connection with 48 V DC power supply. This port is connected to a power supply that is supplied with the nanoBTS. It extends the ethernet connection with 48 V DC for the normal operation mode of the nanoBTS which is in the range between 38-50 V DC. The power consumption of the nanoBTS is 13 W. More details on how to interconnect the cables shall be given in section 6.3. In the middle of the five port region, the mini USB port can be found. It is used by the manufacturer to write the firmware software to the nanoBTS. The last two ports are the TIB-IN and TIB-OUT port[2]. These two ports are used if the GSM network operator requires more than 11 channels to increase the overall capacity of the network. "Up to 4 nanoBTS can be combined into a multiple TRX cell, increasing the number of supported users per TRX by up to 200%. The TIB-OUT from the Master TRX must be connected to the TIB-IN of the slave TRX. This in turn has its TIB-OUT connected to the next TRX in the chain" [43]. The multiple TRX cell configuration

[2]TIB stands for Timing Interface Bus

Figure 6.2.: nanoBTS with two external antennas and five connection ports

shall not be further discussed in this work since the purpose of the work was not to boost the capacity of a GSM network but implementation and testing of the RRLP protocol.

To determine the working state of the nanoBTS, an indicator status LED is located on the left side of the five ports region. After the nanoBTS is connected to the power suplly with the ethernet cable, it shall change its color and blink speed according to the state it is in. The states can be seen in the Table given in 6.1 [45].

One of the key limitations of gathering more technical data and the critical aspect of this description lies in the fact, that nanoBTS is not an open source hardware platform and ip.access does not offer more details on their product. The lack of systematic hardware analysis can be seen as a major drawback of working with the nanoBTS hardware. However, the given technical data are sufficient for reproducing and conducting the RRLP tests described in this thesis.

6.2. GPS Receiver - NL-402U

In the next paragraphs the used GPS device shall be described. In contrast to the earlier described hardware, nanoBTS, which the University of Freiburg already owned, the budget for the GPS receiver was limited and the Navilock NL-402U was bought considering only the single criterion, the price. The Navilock NL-402U GPS receiver is based on the u-blox UBX-G5000 single chipset and is a one chip solution [80]. It can be seen on Figure 6.3 with its passive ceramic patch antenna. 1575,42 MHz is the operating frequency of the receiver which corresponds to the L1 civil frequencies and Coarse/Acquisition (C/A) code. The GPS chipset consists of 50 channels, each channel tracks the transmission from a single satellite [25]. It is important to note, the number of channels inside a GPS receiver interrelates with the amount of time required to obtain the first fix. Receiver tracking sensitivity is -160 dBm (10^{-16} mW). The GPS receiver communicates with the computer ovet the USB port. Although the GPS receiver uses an USB interface, on the computer it emulates 2 UART ports, which are serial communication interfaces.

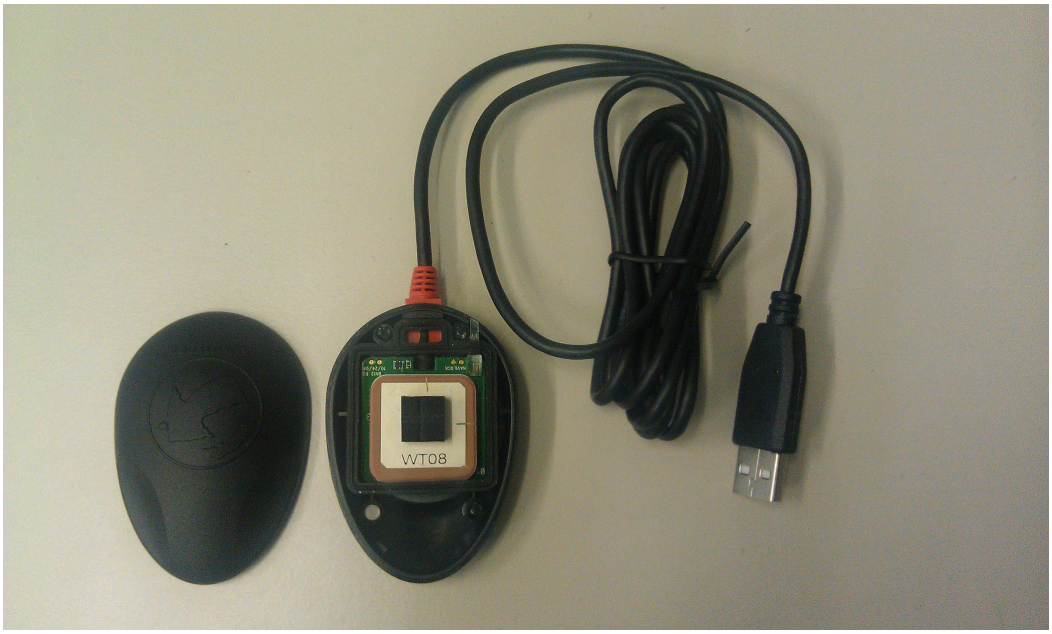

Figure 6.3.: Navilock NL-402U, opened up with the antenna and USB cable

6.3. Cable configuration

In the next section, the author shall focus on properly connecting the hardware. At least 4 ethernet cables with RJ45 connectors, on both sides, were required and one switch or hub connected to the internet. One should take notice of the cabling between the nanoBTS and the ethernet switch or hub, since wrong cabling with the power supply unit (PSU) could damage one of the devices. In Figure 6.4, the junction points are label according to the used configuration setting. The ethernet cables between the switch/hub, PSU and nanoBTS should not be longer than 100 m [45].

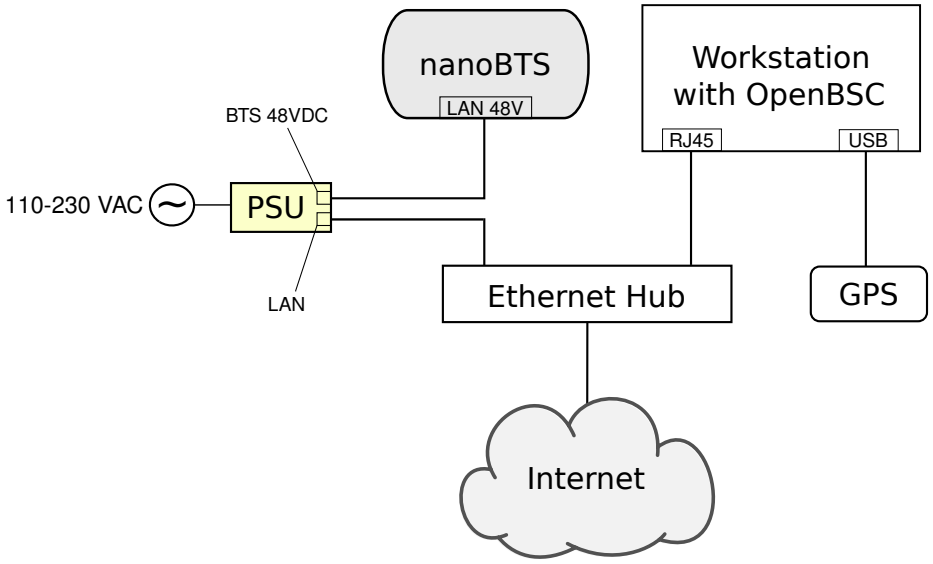

Figure 6.4.: Cable connections, showing interconnection diagram

Table 6.1.: Indicator LED status on the nanoBTS

State	Color & Pattern	When	Precedence
Self-test failure	Red - Steady	In boot or application code when a power on self-test fails	1 (High)
Unspecified failure	Red - Steady	On software fatal errors	2
No ethernet	Orange - Slow flash	Ethernet disconnected	3
Factory reset	Red - Fast blink	Dongle detected at start up and the factory defaults have been applied	4
Not configured	Alternating Red/ Green Fast flash	The unit has not been configured	5
Downloading code	Orange - Fast flash	Code download procedure is in progress	6
Establishing XML	Orange - Slow blink	A management link has not yet been established but is needed for the TRX to become operational. Specifically: for a master a Primary OML or Secondary OML is not yet established; for a slave an IML to its master or a Secondary OML is not yet established.	7
Self-test	Orange - Steady	From power on until end of backhaul power on self-test	8
NWL-test	Green - Fast flash	OML established, NWL test in progress	9
OCXO Calibration	Alternating Green/ Orange - Slow blink	The unit is in the fast calibrating state [SYNC]	10
Not transmitting	Green - Slow flash	The radio carrier is not being transmitted	11
Operational	Green - Steady	Default condition if none of the above apply	12 (Low)

7. Results

One of the most important parts of this thesis are the results that shall be presented in this chapter. Tests will be explained and how the results were obtained. Analysis of the results by the time required to perform a localization of a GSM user and the geographical dislocation error using Google maps are going to be discussed. Smart phones used for the test are going to be introduced and followed by the location of the tests. After the results have been provided, a section with criticism demonstrates all the obstacles that may have appeared while the tests have been performed and why some of the results may be biased. The criticism section is a vital part of this thesis, aside from the given theoretical and mathematical perspective of how AGPS works and why lack of time synchronization inside GSM can be of critical value to correctly evaluate the results. It gives an additional insight into the complete operation of the built localization system in this thesis.

7.1. Tests & Results

Although the main goal was to develop only a working positioning system, tests have been performed mostly inside of a closed building, inside of three different rooms in the mathematics computer pool of the University of Freiburg (German: *Mathematik Rechenzentrum*) and outside of the computer pool.

7.1.1. Smart phones tested

The requirement for a cell phone to be taken into account for testing was its classification as a smart phone (having at least an AGPS receiver) and its availability (person's good will to share their smart phone for the purpose of testing). The following eleven models have satisfied the criterion and have been used to perform the tests, as given in table 7.1.

Table 7.1.: Smart phone models used for testing in the thesis.

Cell phone	Manufacturer & Country
Defy	Motorola, USA
iPhone 4	Apple, USA
iPhone 3GS	Apple, USA
G1	Google, USA
Galaxy S2	Samsung, South Korea
Galaxy S3	Samsung, South Korea
Galaxy Nexus i9250	Samsung, South Korea
E71	Nokia, Finland
N95	Nokia, Finland
Desire S	HTC, Taiwan
Blade	ZTE, P.R. of China

7.1.2. Performed tests

As stated in section 7.1, tests have been performed outside and inside of the computer pool building. Three different test modes were tried out, first only an RRLP request without any assistance data was sent. The second test included an RRLP request with almanac, UTC model, ionospheric model and reference location data. The last test was an RRLP request with almanac, ephemeris, UTC model, ionospheric model and reference location data. The stated requests have been sent in an reverse order, to observe if the smart phones can actually make an usage of the assistance data. The RRLP requests were manually sent from the telnet interface from OpenBSC after they have been implemented by the author. Results delivered by the MS were stored in the database and the following analysis is based on them and on the time out results which were not stored in the database but notices by the author.

The first tests took place on the ground-floor of the computer pool, in figure 7.1 depicted with a green dot as Test room 1. The smart phones were horizontally lying

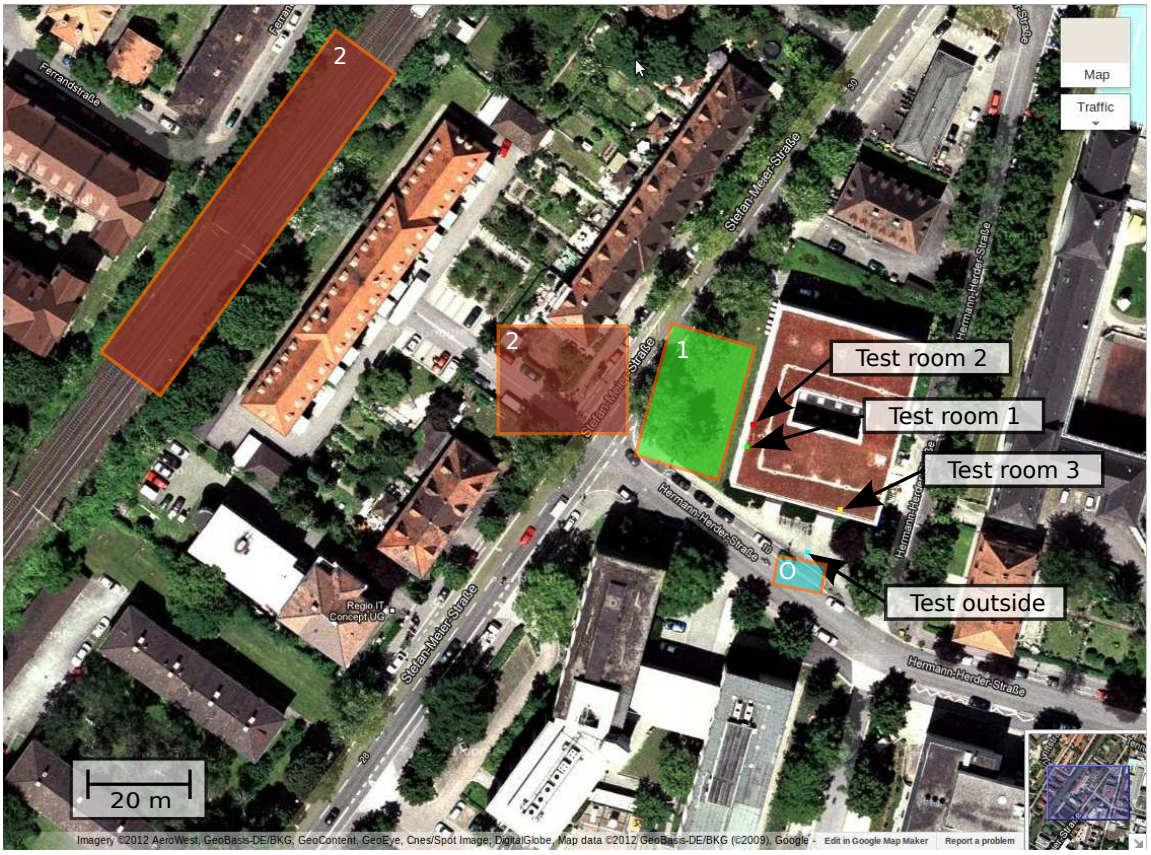

Figure 7.1.: Test rooms as well as the results delivered by the smart phones. Image courtesy of Google Maps.

on the table 50*cm* away from the window in the first test and in the second test vertically parallel to the window. The results of this two smart phone position tests in the room showed that smart phone position did not make any influence on the test results. The delivered position coordinates by the smart phones in all performed tests were in range of the green rectangle labeled with a white one. According to Google Earth, Test room 1 has the following coordinates: latitude 48°0'13.21"N and longitude 7°50'53.53"E. The results were 5-20 m away from the real position according to Google Earth. The smart phones that provided these results were the *iPhone* 3*GS* and *G*1. The *iPhone* 3*GS* sent only a response when all assistance data (almanac, ephemeris, UTC model, ionospheric model and reference location data) have been delivered whereas the *G*1 only when the assistance data without ephemeris data were delivered or by only sending an RRLP request without any assistance data. It is apparent from these facts that the *iPhone* 3*GS* had used the assistance data to estimate its position otherwise it would send it is position back also when only an

(a) Smart phones on the window.

(b) Smart phones on the table.

Figure 7.2.: Test room 2 with the positions of the smart phones

RRLP position request was sent. Interestingly, the $G1$ did not deliver any results when the ephemeris data have been delivered to it. These findings suggest that the AGPS receiver in $G1$ may not know how to employ the ephemeris data because it is one of the first "real" smart phones on the market. Later on it can be seen other smart phones that are even older than the $G1$ can not employ any of the assistance data.

The second tests took place in Test room 2. Test room 2 is located in the basement of the computer pool, with the following coordinates: latitude 48°0'13.12"N and longitude 7°50'53.50"E. The fact tests took place in the basement adds an additional obstacle to the AGPS receiver in the MS, the signal strength of GPS signals is even weaker. The GPS signal strength was measured with an external GPS receiver connected to the computer and by its acquisition time it could be easily observed that the time required to track the satellites was a few times longer than in the Test room 1. The results from Test room 2 are most valuable because they have provided additional evidence that even without line of sight it is possible to receive weak GPS signals and approximate the position. The obtained results from the tests in Test room 2 can be seen in figure 7.1, two red rectangles labeled with a two in the left upper corner. It is somewhat surprising that different cell phone models delivered different positions (two different rectangle ranges) at different times of the day. This finding suggests that not the equal number of satellites were visible at the different time points when the tests have been performed. By observing the results in figure 7.1, it is straightforward to see deviation of the estimated positions by comparing the tests performed in Test room 1 and 2. Albeit these two test rooms are geographically not far away from each other, the major difference is in their

altitude and GPS signal strength reception. The estimated position deviation from the real position was greater with the GPS signal reception quality and with reduced satellites visibility. The majority of the performed tests in this work were performed in this room. Smart phones have been tested on the table which is $3m$ away from the windows and on the window itself, as shown in figure 7.2. These small changes in position of the smart phone did not make any difference in the resulting estimated position. The smart phones tested in Test room 2 were all the listed ones in table 7.1. One unanticipated finding was that the "newer generation" smart phones did not deliver their position in any case but rather requested more assistance data like *iPhone* 4, *Galaxy S*2 and *Galaxy S*3. There are two possible explanations for this result. This might be because the newer AGPS devices require more assistance data by relying on the network providers to have synchronized GSM systems or it is an security protection of the smart phone user. No information on the AGPS receiver chipset in the smart phones could be found online and author's given statements have to be considered with ambiguity. Another contrary to expectations, were the results with two Nokia "smart phone" models $E71$ and $N95$, results were only delivered when an RRLP request was sent without any assistance data. Although it was stated in their specifications both are equiped with an AGPS receiver RRLP requests with assistance data did not produce any output from these smart phones [62] [61]. The reason for this is not clear but it might be due to the fact these are older models in comparison to other smart phones in the tests. At the time point when they were released by Nokia the firmware for the phones might not have been fully evolved and developed since both models are from the same company.

The third test took place in Test room 3, as shown in figure 7.1 by the yellow dot. The third test room has the following coordinates: latitude 48°0'12.26"N and longitude 7°50'54.45"E. The smart phones were placed on the windows. In this room the smart phones tested did not deliver any positions but only errors about missing assistance data and time outs. While no MS delivered its position, the room was tested if a GPS position can be obtained with an external GPS receiver. The test was successful and after 12 minutes the position was obtained but the signal strength was weak according to the delivered GPS output. An implication emerging from this finding may be related to the GPS receiver high sensitivity (-160 dBm 10^{-16} mW) and an active patch antenna with a size of 2x2 cm [80]. The GPS receivers employed in smart phones have to be small in size to fit into the device and use a passive antenna because active antennas have an additional power consumption [79].

To determine if the delivered results are taken from the cache or some other memory in the smart phone, the same test have been performed in the basement hallway where no windows exist just after the smart phones delivered successfully their position in Test room 2. This test did not deliver any position but only time outs or errors

containing information that no satellites are visible. This test provided and confirmed that the smart phones are always performing a postion estimation at the moment when an RRLP request is sent to the MS.

The last test has been performed outside of the computer pool building. This test was conducted to confirm the argument that precision of the estimated position is related to the received GPS signal strength and number of visible satellites. As it can be seen in figure 7.1, the tests were performed at the blue dot and around it and the estimated positions were 1-5 m off of the real position. The estimated positions are shown in a blue rectangle with an O in the left corner of it.

Table 7.2.: Smart phone RRLP test results

Cell phone model	RRLP(E)	RRLP(A)	RRLP	Type of error (or missing data)
Defy	No	No	No	No response (time out)
iPhone 4	No	No	No	Reference time, Navigation Model, Reference Location
iPhone 3GS	Yes	Yes	No	/
G1	No	Yes	Sometimes	/
Galaxy S2	No	No	No	Acquisition Assistance
Galaxy S3	No	No	No	Reference Location, Reference Time, Acquisition Assistance, Navigation Model
Galaxy Nexus i9250	No	No	No	Did not respond, only ACKs
E71	No	No	Yes	/
N95	No	No	Yes	/
Desire S	Yes	No	No	/
Blade	No	No	Yes	/

In table 7.2 the list of all results is shown. The abbreviations used in the table are explained in this paragraph. RRLP indicates the MS has delivered its position only when an RRLP position request without any assistance data has been sent. RRLP(E) indicates the MS has delivered its position only when an RRLP request contained almanac, ephemeris, UTC model, ionospheric model and reference location data has been sent. RRLP(A) indicated the MS has delivered its position only when an RRLP request contained almanac, UTC model, ionospheric model and reference location data has been sent (the difference from RRLP(E) is in the fact that no ephemeris data are included). If there is no error description then the stated data were requested to be delivered to the MS. If the position was delivered the position estimation by the smart phones took never longet than 3 minutes. The waiting time period of 3 minutes for the result is not discouraging provided that the assistance data like reference time (exact time, explained in section 4.2) and acquisition assistance data (phase and Doppler

effect frequency required by the AGPS, explained in section 4.2) were not delivered to the MS. It is important to mention the strange behaviour by *Galaxy Nexus i*9250, the MS responded only with acknowledgements while the assistance data have been sent but after the reception it immediatelly closed the SDCCH channel. The *Blade* closed the SDCCH channel after 4 transmitted assistance packets for the RRLP(E) test. The *Defy* by Motorola did not produce any output at all and behaved like a 2G cell phone without a GPS receiver. To eliminate any doubts and suspicion if the SDCCH channel was properly working and not producing the time outs, 2G phones (Nokia 3310 and Siemens M50) have been used to perform tests. An SDCCH channel has been initialized 10 times at different days with the 2G phones and left open for 10 minutes. The 2G phones reported the signal strengths of the initialized channel according to the defined standard. This provides a proof for the cases where the time out appeared, it was not produced by the BTS but rather by the smart phone. One important remark related to the tests ought to be mentioned. While the *iPhone* 3*GS* and *G*1 provided the results in the tests, sometimes it was the case they did not deliver the results the first time the RRLP request was executed but an time out. Second time the same RRLP request was sent, the smart phones delivered their positions. This unexpected behaviour raises a suspicion that the smart phones do not behave according to the RRLP standard where it is well defined how much time they have to perform the localization. However, the previous statement ought to be considered with some uncertainty since it can not be proved without access to the firmware of the smart phones. The combination of described findings in this chapter and in table 7.2 provides some support for the premise that the RRLP standard is not yet a fully implemented standard by all the manufacturers that claim AGPS functionality. Does it depend on the AGPS chipset or RRLP itself, remains an open question.

7.2. Criticism of performed tests

Perhaps the most serious weakness of the presented results is that the author had no access to the firmware of the MS while the tests have been performed. This would allow the author to see what type and how the assistance data are employed by the AGPS in the MS. If access could be gained, to the internal operation of the AGPS receiver, all doubts and bias about the deduced hypotheses could be eliminated. The whole system represents a black box where an input is delivered and an output is expected. Another drawback was the lack of hardware information about the hardware inside of the MS (AGPS receivers and antennas). This does not allow an exact comparison between different cell phone models and if they can acquire any GPS signal in weak signal strength conditions.

Difficulties arise in assessment and comparing the results in this thesis with other relevant studies due to the lack of any research studies compleyed using the equivalent hardware and type of assistance data. In the relevant studies different hardware test equipment is used while this thesis was carried out without that test equipment [16]. In addition, no research has been found that surveyed the amount of time required to get a position response from a MS where only almanac, ephemeris, UTC model, ionospheric model and reference location data have been delivered to the MS.

Another limitation of the evaluated results lies in the fact that it has only been applied to the stated cell phones and it could not be tested with all possible models. The tests suffer from a major drawback as real time movement of satellites, the tests could not be conducted parallely but rather in serial manner in time. In other words a satellite visible at the moment while the first test is being performed may not be visible the second time when the test is executed. The tests were performed in the morning 10:00-12:00 and in the afternoon 16:00-19:00, with morning being the period of day when it was difficult even for the external GPS device to track the satellites. GPS signal strength is a vital measurement information, where the signal levels are lower than a predefined acquisition sensitivity even assistance data can not help. Antenna polarization and the position of the cell phone matter as well.

Correctness of assistance data in almanac and ephemeris data can not be verified. The author had to rely and trust NASA and Trimble as sources although errors were confirmed by different studies in [39] [48]. Errors can be confirmed by the author in ephemeris data as well (URA values were out of range specified by the standard). These errors were not continual but appeared occasionally and these errors were inside of the assistance data provided by NASA.

7.3. Future work

The system could be extended with a GPS device that delivers raw GPS data instead of using the data provided by NASA and Trimble. Obtained data by the GPS could be compared to the data provided by NASA or other GPS observation stations and verified for errors. By having more redundant sources of same information, mistakes in the output could be eliminated. More sources of redundant data could indicate the correctness of assistance data.

Another idea to extend this work would be to use an LMU and provide the reference time and acquisition assistance data. This feature would enhance the complete system. However, one ought to understand it would require great changes in the OpenBSC source code and interoperability between the BTS and LMU.

Additionally a position tracking system could be built. By adding a timer that will execute an RRLP request every few minutes. The successfully estimated positions could be connected into a path and displayed where the GSM user spends his time. Along the described method, a machine learning algorithm could be developed to predict the movement of GSM users [56].

8. Summary and discussion

This thesis has investigated how difficult it is to integrate mobile assisted GPS localization in GSM Networks. The aim of this work was set out to implement the “first” working open source RRLP implementation in GSM networks, as well as to determine and evaluate the limits of this localization technique. The research performed in this work has shown, it is not convoluted and burdensome to estimate the position of GSM users even inside of buildings granted that required assistance data can be provided by the GSM network operator. The findings of this study indicate that GSM users of smart phones can be tracked accurately and precisely without their knowledge. An implication of these evidence suggest it would be not complicated for German law enforcement agencies to employ this precise surveillance technique to follow suspects. According to the German Interior Minister Hans-Peter Friedrich, in 2011 silent SMS were employed with the UL-TDOA technique to track down suspects [20] [71]. The law is unclear if silent SMS can be considered communication taking into consideration no information are sent to the GSM user and is a gray area from the legal point of view [71]. "The state found that it was not one, since there is no content. This is useful, because if it is not a communication, it does not fall under the framework of the inviolability of telecommunications described in Article 10 of the German Constitution." said Mathias Monroy from Heise Online [71]. The development of a working RRLP application and obtained results from this work enhance the understanding of AGPS receivers and may be further used to better understand how the assistance data influence the obtained results. Finally, a number of important limitations in the obtained results need to be considered. Not all assistance data were available and the tests have been performed at different time points of the day. The amount of tested cell phones was not representative enough. However, this work has thrown up some questions in need of further investigation but it is only a tip of the iceberg! A future study investigating if further assistance data are provided to the cell phones would be very interesting. The produced RRLP software and obtained results may be used to develop new strategies aimed at protecting privacy of cell phone users.

8. Summary and discussion

Dictionary of acronyms

- *ARFCN* - Absolute Radio Frequency Channel Number - The channel number specifies the physical frequency channel used for transmission and reception of radio waves inside of an BTS covered area.
- *BTS* - Base Transceiver Station -
- *DC* - Direct Current
- *GNSS* - Global Navigation Satellite System - A satellite navigation system that allows a specialized receive to determine its location on Earth.
- *LED* - Light Emitting Diode - A diode that emitts light.
- *IP Address* - .
- *PCB* - Printed Circuit Board - The board where electronic components are soldered onto and wired through conductive tracks.
- *RRLP* - Radio Resource Location Protocol - The employed protocol in GSM, UMTS and other wireless networks for providing and exchange of geolocation information.
- *SMA* - SubMiniature version A - SMA is a connector used for interconnecting coaxial cables or PCB electronics that work in the frequency range between 0-18 GHz.
- *TIB* - Time Interface Bus - The TIB is used to provide the synchronization of the clock, frequency and frame number between the nanoBTS when operating in a single 2-4 BTS configuration.
- *TRX* -
- *UART* - Universal Asynchronous Receiver Transmitter - A serial communication interface used by computers or other peripheral devices to communicate.
- *UMTS* - Universal Mobile Telecommunications System - Third generation mobile network based on the GSM standards.

Write what an IP address is

Appendix

A. Installation and configuration guide

In order to evaluate the localization system, it is required to install OpenBSC and to modify the proper source files and compile the system. The aim of this section is to describe that process in such detail that the presented material is sufficient to reproduce equivalent or similar results. The guide was successfully tested out on the following operating systems: Ubuntu 10.04 LTS 64 bit and Ubuntu 12.04 LTS 64 bit. A self-bootable test USB system is supplied with the thesis and it can be evaluated without executing the given steps. There is a marking difference between text given in light and dark grey background color, the first ought to be typed in into the terminal window or it may be an output produced by an application, whereas the later emphasizes a file modification case.

A.1. Installation of OpenBSC

In order to compile OpenBSC it is required to install the following precompiled packages[1]:

- libdbi0
- libdbi0-dev
- libdbd-sqlite3
- libortp-dev
- build-essential
- libtool
- autoconf
- automake
- git-core
- pkg-config

[1] If more details are required for the installation process a guide can be found at [64].

Before installing the required packages and libraries, to keep the installation process clean and free of modifying other files, the author will create a new directory.

```
mkdir gsm_localization
cd gsm_localization
```

By executing the following instructions the required libraries will be installed.

```
sudo apt-get install libdbi0-dev libdbd-sqlite3 build-essential
sudo apt-get install libtool autoconf automake git-core
sudo apt-get install pkg-config libortp-dev
```

After the packages were installed, *libosmocore* library must be downloaded, compiled and installed. By executing the following instructions:

```
git clone git://git.osmocom.org/libosmocore.git
cd libosmocore
autoreconf -fi
./configure
make
sudo make install
sudo ldconfig
cd ..
```

In the next step *libosmo-abis* will be installed.

```
git clone git://git.osmocom.org/libosmo-abis.git
cd libosmo-abis
autoreconf -fi
./configure
make
sudo make install
sudo ldconfig
cd ..
```

After the previous steps have finished successfully, the author will proceed with downloading, compiling and installing OpenBSC.

```
git clone git://git.osmocom.org/openbsc.git
cd openbsc/openbsc
autoreconf -i
sudo export PKG_CONFIG_PATH=/usr/local/lib/pkgconfig
./configure
make
```

At this point, OpenBSC should be successfully compiled.

A.2. Configuring nanoBTS for OpenBSC

To enable the nanoBTS and OpenBSC to be fully operational, the last configuration steps have to be made. It is necessary to inform the nanoBTS of the IP address of the server that is running OpenBSC since it must connect to OpenBSC. We need to find a free ARFCN channel where our system is expected to operate[2].

To find the ID and the IP address of the nanoBTS it is required to start *ipaccess-find*[3].

```
cd ~/gsm_localization/openbsc/openbsc/src/ipaccess
./ipaccess-find
```

ipaccess-find will produce an output similar to the one given:

```
Trying to find ip.access BTS by broadcast UDP...
MAC_Address='00:02:95:00:61:70'  IP_Address='132.230.4.63'
Unit_ID='1801/0/0'  Location_1=''  Location_2='BTS_NBT131G'
Equipment_Version='165g029_73'
Software_Version='168a352_v142b30d0'
Unit_Name='nbts-00-02-95-00-61-70'
Serial_Number='00110533'
```

In the next step, the nanoBTS is informed of the OpenBSC IP address by typing the following commands (the first IP address belongs to the server running OpenBSC and the second to the nanoBTS):

```
cd ~/gsm_localization/openbsc/openbsc/src/ipaccess
./ipaccess-config -o 132.230.4.65 132.230.4.63 -r
```

It is required to create the directory where the configuration file will be located and to modify the configuration file.

```
sudo mkdir /usr/local/lcr
cd ~/gsm_localization/openbsc/openbsc/doc/
cd examples/osmo-nitb/nanobts
sudo cp openbsc.cfg /usr/local/lcr
sudo vim /usr/local/lcr/openbsc.cfg
```

[2]A licence has to be obtained from the Federal Network Agency (German: *Bundesnetzagentur*), otherwise it is ilegal and may be considered as a criminal act.

[3]The nanoBTS ought to be blinking in orange color before starting *ipaccess-find*.

A free ARFCN channel can be found using a spectrum analyzer and by setting the frequency range to the GSM frequency band. One has to slide through the frequencies shown on the X-axis, and by looking at the Y-axis with appropriate frequency resolution[4], where the received power is represented[5]. By patiently observing the Y-axis it can be easily seen on the X-axis which channels are taken by other GSM service providers and which are free. The chosen channel ought to be peak free. Once a free frequency channel has been found, it is necessary to instruct the nanoBTS to operate in that frequency range. The line, numbered 58, has to be modified with the correct free ARFCN channel,in this case 877.

```
arfcn 877
```

The ARFCN channel value can be calculated using the given formula in (A.2.1), where f_{start} is the starting frequency of the uplink bandwitdh for DCS1800, f_{CB} is the channel bandwidth and *Offset* is the offset[6].

$$f_{up}(\text{ARFCN}) = f_{start} + f_{CB} \cdot (\text{ARFCN} - \text{Offset})$$

$$where \begin{cases} f_{start} &= 1710.2\,\text{MHz} \\ f_{CB} &= 200\,\text{kHz} \\ \text{Offset} &= 512 \end{cases} \tag{A.2.1}$$

On line numbered 53, the last configuration file modification has to be made for the final configuration of the OpenBSC software. The Unit ID from the output above has to be set[7].

```
ip.access unit_id 1801 0
```

At this point the nanoBTS and OpenBSC configuration is done.

[4] The frequency resolution must be set to $f_{CB} = 200\,\text{kHz}$ or higher values for faster movement in the frequency spectrum.

[5] Dependent of the manufacturer and settings of the spectrum analyzer, it can show signal amplitude, magnitude and power.

[6] A table with frequency channels can be found at the following URL: https://gsm.ks.uni-freiburg.de/arfcn.php

[7] Indentation has to match the one of the configuration file.

A.3. Installation and configuration of GNSS assistance software

To install the RRLP software that generates GNSS assistance data several libraries are required to be installed, *cURL*[8], *libconfig* and *SQLite*. *cURL* was used for the purpose of safely downloading GNSS data from the Navigation Center of the US Coast Guard and Trimble server. *libconfig* library is used for reading in the configuration file, this way compiling of the software whenever one changes the settings was avoided. The *SQLite* library was employed to access the database used by OpenBSC to store the respondence data from the mobile stations.

```
cd ~/gsm_localization
sudo apt-get install libsqlite3-dev
wget http://curl.haxx.se/download/curl-7.25.0.tar.gz
wget http://www.hyperrealm.com/libconfig/libconfig-1.4.8.tar.gz
tar -xvzf curl-7.25.0.tar.gz
tar -xvzf libconfig-1.4.8.tar.gz
cd curl-7.25.0
make
sudo make install
cd ..
cd libconfig-1.4.8/
./configure
make
sudo make install
```

Once the libraries have been successfully installed, the user may proceed with the configuration and compiling the GNSS assistance software, which is the key software produced in this thesis. The configuration file can be found in the same directory as the RRLP modules under the name: "gnssrrlp.cfg". The sample configuration file is already preconfigured for the location of "Angewandte Mathematik und Rechenzentrum" building. Latitude and longitude of the BTS are expressed in decimal degrees and are bounded by $\pm 90°$and $\pm 180°$respectively. Positive latitudes are north of the equator, whereas negative are south of the equator. It is alike for longitude coordinates, positive longitudes are east of Prime Meridian and negative are west of the Prime Meridian. If the position in decimal degrees of the BTS is unknown, it is straightforward to derive them using the formula given in (A.3.1), where D are

[8] It may happen that the given download URLs are wrong and in the meantime have changed, but one can easily find the latest versions on http://curl.haxx.se/ and http://www.hyperrealm.com/libconfig/

degrees, M are minutes and S are seconds[9].

Describe other parameters as well.

$$DD = D + \frac{M}{60} + \frac{S}{3600} \tag{A.3.1}$$

The altitude may be left as it is, set to 0, since it is not used in the current measurement technique[10].

```
// An example configuration file for the GNSS RRLP software.
name = "Configuration for GNSS and RRLP";

// Change the settings if required:
settings =
{
  config = ( {
    ephemeris_url  = "ftp://ftp.trimble.com/pub/eph/CurRnxN.nav";
    almanac_url = "http://www.navcen.uscg.gov/ ↙
       ↳ ?pageName=currentAlmanac&format=yuma";
    latitude_of_BTS  = 48.003601;
    longitude_of_BTS    = 7.848056;
    altitude_of_BTS  = 0.0;
    uncertainty_of_lat_long  = 7;
    uncertainty_of_alt  = 7;
    confidence_level    = 0;
    ephemeris_repair  = false;
    use_reference_time  = false;
    extra_seconds_to_add  = 7;
    timezone_of_BTS  = 1;
    time_to_refresh_ephem = 1;
    time_to_refresh_alm = 1 ; } );
};
```

CHECK IF THIS IS CORRECT

The uncertainty of the latitude and longitude correctness can be described using equation (A.3.2) [6]. The uncertainty of r is expressed in meters, it defines how accurate is the specified location of the BTS. In the configuration file, K is set to 7, which corresponds to $r = 9.4872$ m. Instead of using the integer parameter K as the known variable, the equation (A.3.2) can be rewritten as in (A.3.3), where we can

[9] An online converter of the Federal Communication Commission can be used as well to convert from degrees, minutes and seconds to decimal degrees and vice versa http://transition.fcc.gov/mb/audio/bickel/DDDMMSS-decimal.html

[10] If the value is set to zero, it is important to set it to 0.0 because *libconfig* would otherwise convert it to an integer however it is a floating point number.

get the integer value K for a previously selected r.

$$r = C((1+x)^K - 1)$$

$$where \begin{cases} C &= 10 \\ x &= 0.1 \\ K &\in [0,127] \cap \mathbb{N}_0 \end{cases} \tag{A.3.2}$$

$$K = \left\lceil \frac{ln(\frac{r}{C}+1)}{ln(1+x)} \right\rceil$$

$$where \begin{cases} C &= 10 \\ x &= 0.1 \\ r &\in [0,1800]\,\text{km} \end{cases} \tag{A.3.3}$$

A set of uncertainties r is given in table A.3.1 for various integer values of K.

Value of K	Value of uncertainty r
0	0 m
1	1 m
2	2.1 m
3	3.3 m
-	-
20	57.3 m
-	-
60	3.0348 km
-	-
100	137.8 km
-	-

Table A.3.1.: Example uncertainties (latitude and longitude) for various integer values of K

Altitude uncertainty can be described using the same Binomial expansion method, as given in (A.3.4), however with altered constant values [6]. The altitude uncertainty ranges between 0 m and 990.5 m ($h \in [0, 990.5]\,\text{m}$). Although the same constant name K is used, it describes the altitude uncertainty, (A.3.5).

$$h = C((1+x)^K - 1)$$

$$where \begin{cases} C &= 45 \\ x &= 0.025 \\ K &\in [0,127] \wedge \|K\| \end{cases} \tag{A.3.4}$$

$$K = \left\lceil \frac{ln(\frac{h}{C}+1)}{ln(1+x)} \right\rceil$$

$$where \begin{cases} C &= 45 \\ x &= 0.025 \\ h &\in [0,990.5]\,\mathrm{m} \end{cases} \tag{A.3.5}$$

A set of uncertainties h is given in table A.3.2 for various integer values of K.

Value of K	Value of uncertainty h
0	0 m
1	1.13 m
2	2.28 m
3	3.46 m
-	-
20	28.74 m
-	-
60	152.99 m
-	-
100	486.62 m
-	-

Table A.3.2.: Example uncertainties (altitude) for various integer values of K

Confidence level is the next parameter in the configuration file that needs to be set. It can take any integer value between 0 and 127. The confidence level defines the percentage of the confidence that the target entity, the GSM user one wants to locate, is within the geometric shape defined earlier. A value of 0 and between 100 and 127, may be interpreted as "no information" [6]. The reason why the values are not limited to 100 is because of the nature of binary numbers and that 2^6 bits is not sufficient to represent the number 100, but rather requires one bit more.

Confidence level is followed by the ephemeris repair option. Ephemeris repair is a variable of the boolean type, it can take two different values *true* or *false*. Ephemeris

data may contain errors or miss some satellite information [48] [39] and the ephemeris repair function, if set to true, will take data of the previous measurement report. This introduces an error as well.

To increase the speed of measurement report, reference time can be used to provide extra information for the A-GPS in the MS of target entity. This field is of boolean type, if set to true, reference time is included in the sent packets.

see how much the reference time can deviate from current time

Since the sent packets are not transmitted in real time but put on a stack and then sent to the MS, a time delay exists. A solution to this problem is to add extra seconds to the reference time being sent. In order to assess the amount of extra seconds to add, the GSM operator is required experimentally to verify his/her findings. .

The reference time being sent to the MS is Coordinated Universal Time (UTC). The GPS device receives UTC time from the satellites and adjusts the computer time. To set the correct time, time zone offset of the BTS ought to be set correctly.

Finally, the refresh time of downloading new almanac and ephemeris data has to be set. The variable uses the hour unit, how often the data are being downloaded. If the data are used from a local GNSS station, refresh time of the ephemeris data should be set to every 30 minutes or 0.5 hours. The almanac data are valid for up to 180 days [1] but are updated usually every day[11] [33].

[11] Almanac update times can be found here: http://www.navcen.uscg.gov/?pageName=currentNanus&format=txt

B. Sourcecode

Example:

```
#include <stdio.h>

int main(void)
{
    printf("Hallo Welt!\n");
    return 0;
}
```

C. GPS Constants and equations

$$
\begin{aligned}
A &= (\sqrt{A})^2 \\
n_0 &= \sqrt{\frac{\mu}{A^3}} \\
t_k &= t - t_{oe} \\
n &= n_0 + \Delta n \\
M_k &= M_0 + nt_k \\
M_k &= E_k - e\sin E_k \\
v_k &= \tan^{-1}\left(\frac{\sin v_k}{\cos v_k}\right) = \tan^{-1}\left(\frac{\frac{\sqrt{1-e^2}\sin E_k}{1-e\cos E_k}}{\frac{\cos E_k - e}{1-e\cos E_k}}\right) \\
v_k &= \tan^{-1}\left(\frac{\sin v_k}{\cos v_k}\right) = \tan^{-1}\left(\frac{\sqrt{1-e^2}\sin E_k/(1-e\cos E_k)}{(\cos E_k - e)/(1-e\cos E_k)}\right) = \tan^{-1}\left(\frac{\sqrt{1-e^2}\sin E_k}{\cos E_k - e}\right) \\
E_k &= \cos^{-1}\left(\frac{e+\cos v_k}{1+e\cos v_k}\right) \\
\Phi_k &= v_k + \omega \\
\delta u_k &= c_{us}\sin 2\Phi_k + C_{us}\cos 2\Phi_k \qquad \text{(C.0.6)} \\
\delta r_k &= c_{rc}\cos 2\Phi_k + C_{rs}\sin 2\Phi_k \\
\delta i_k &= c_{ic}\cos 2\Phi_k + C_{is}\sin 2\Phi_k \\
u_k &= \Phi_k + \delta u_k \\
r_k &= A(1 - e\cos E_k) + \delta r_k \\
i_k &= i_0 + \delta i_k + (IDOT)t_k \\
x_k^{'} &= r_k\cos u_k \\
y_k^{'} &= r_k\sin u_k \\
\Omega_k &= \Omega_0 + (\Omega - \Omega_e)t_k - \Omega_e t_{oe} \\
x &= x_k^{'}\cos\Omega_k - y_k^{'}\cos i_k\sin\Omega_k \\
y &= x_k^{'}\sin\Omega_k - y_k^{'}\cos i_k\cos\Omega_k \\
z &= y_k^{'}\sin i_k
\end{aligned}
$$

$$
\mu_e = 3.986004418\cdot 10^{14}\frac{m^3}{s^2} \quad \Longleftarrow \quad \text{Geocentric gravitational constant} \qquad \text{(C.0.7)}
$$

$$c = 2.99792458 \cdot 10^8 \frac{m}{s} \quad \Longleftarrow \quad \text{speed of light} \tag{C.0.8}$$

List of Tables

List of Figures

List of Figures

Bibliography

[1] Navstar GPS User Equipment Introduction. Online, Sept. 1996. URL http://www.navcen.uscg.gov/pubs/gps/gpsuser/gpsuser.pdf.

[2] Interface Specification IS-GPS-200. Online, June 2010. URL http://www.losangeles.af.mil/shared/media/document/AFD-100813-045.pdf.

[3] 3GPP. Location Services (LCS); Functional description; Stage 2. TS 03.71 V7.11.0, 3rd Generation Partnership Project (3GPP), June 2006. URL http://www.quintillion.co.jp/3GPP/Specs/GSM_GERAN/0371-7b0.pdf.

[4] 3GPP. Location Services (LCS); Mobile Station (MS) - Serving Mobile Location Centre (SMLC) Radio Resource LCS Protocol (RRLP). TS 04.31 V8.18.0, 3rd Generation Partnership Project (3GPP), June 2007. URL http://www.quintillion.co.jp/3GPP/Specs/GSM_GERAN/0431-8i0.pdf.

[5] 3GPP. Location Services (LCS); Base Station System Application Part LCS Extension (BSSAP-LE) (Release 8). TS 49.031 V8.1.0, 3rd Generation Partnership Project (3GPP), Dec. 2008. URL http://www.quintillion.co.jp/3GPP/Specs/GSM_GERAN/49031-810.pdf.

[6] 3GPP-Coordinates. 3GPP TS 23.032 V6.0.0 (2004-12), 3rd Generation Partnership Project; Technical Specification Group Core Network; Universal Geographical Area Description (GAD) (Release 6). Technical report, Dec. 2004.

[7] N. Agarwal, J. Basch, P. Beckmann, P. Bharti, S. Bloebaum, S. Casadei, A. Chou, P. Enge, W. Fong, N. Hathi, W. Mann, A. Sahai, J. Stone, J. Tsitsiklis, and B. Van Roy. Algorithms for GPS operation indoors and downtown. *GPS Solutions*, 6:149–160, 2002. ISSN 1080-5370. 10.1007/s10291-002-0028-0.

[8] D. Akopian and J. Syrjarinne. A network aided iterated LS method for GPS positioning and time recovery without navigation message decoding. In *Position Location and Navigation Symposium, 2002 IEEE*, pages 77–84, 2002. doi: 10.1109/PLANS.2002.998892.

[9] Andrew Rassweiler. iPhone 3GS Carries $178.96 BOM and Manufacturing Cost, iSuppli Teardown Reveals. http://www.isuppli.com/Teardowns/News/Pages/iPhone-3G-S-Carries-178-96-BOM-and-Manufacturing-Cost-iSuppli-Teardown-Reveals.aspx. [Online; accessed 14-July-2012].

[10] A. Bensky. *Wireless positioning technologies and applications*. Artech House, Boston, Mass, 2008. ISBN 1596931302.

[11] E. Blossom. Exploring gnu radio. http://www.gnu.org/software/gnuradio/doc/exploring-gnuradio.html#fm-receiver, 2004. [Online; accessed 30-August-2012].

[12] blur group marketing. Trends and Statistics in Location Based Services. http://blur-marketing.com/blog/trends-and-statistics-in-location-based-services/. [Online; accessed 7-July-2012].

[13] K. Borre. *A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach (Applied and Numerical Harmonic Analysis)*. Birkhäuser Boston, 2006. ISBN 9780817643904.

[14] M. Braasch and A. van Dierendonck. GPS receiver architectures and measurements. *Proceedings of the IEEE*, 87(1):48–64, jan 1999. ISSN 0018-9219. doi: 10.1109/5.736341.

[15] J. R. Clynch. Earth Coordinates. http://www.gmat.unsw.edu.au/snap/gps/clynch_pdfs/coorddef.pdf, 2006. [Online; accessed 27-June-2012].

[16] S. Communication. A-gps over the air test method: Business and technology implications. http://www.spirentfederal.com/gps/documents/Spirent_A-GPS_OTA_whitepaper.pdf, 2009. [Online; accessed 20-August-2012].

[17] CORP, DAISHINKU. Development of Miniature High-Precision SMD TCXO for GPS. Technical report, DAISHINKU CORP. 1389 Shinzaike, Hiraoka-cho, Kakogawa, Hyogo 675-0194 Japan, 2008. URL http://www.kds.info/html/products/new_product/4567115_en.htm.

[18] L. Deng and D. O'Shaughnessy. *Speech Processing: A Dynamic and Optimization-Oriented Approach (Signal Processing and Communications)*. CRC Press, 2003. ISBN 0824740408.

[19] Department of Physics and Astronomy, Georgia State University. Maximum Sensitivity Region of Human Hearing. http://hyperphysics.phy-astr.

gsu.edu/hbase/sound/maxsens.html. [Online; accessed 13-August-2012].

[20] D. D. der mksult GmbH. Polizei setzt häufig stille sms zur ortung verdächtiger ein. https://www.unwatched.org/EDRigram_10.2_Polizei_setzt_haeufig_Stille_SMS_zur_Ortung_Verdaechtiger_ein, 2012. [Online; accessed 2-September-2012].

[21] V. Diggelen. *A-GPS assisted GPS, GNSS, and SBAS*. Artech House, Boston, 2009. ISBN 1596933747.

[22] G. Djuknic and R. Richton. Geolocation and assisted GPS. *Computer*, 34(2): 123 –125, feb 2001. ISSN 0018-9162. doi: 10.1109/2.901174.

[23] Dominik Schneuwly. The Synchronization of 3G UMTS Networks. http://www.oscilloquartz.com/file/pdf/Ap17UMTS-screen.pdf. [Online; accessed 14-July-2012].

[24] J. Eberspächer, H.-J. Vögel, C. Bettstetter, and C. Hartmann. *GSM - Architecture, Protocols and Services*. Wiley, 2009. ISBN 0470030704.

[25] C. H. Elliott D. Kaplan. *Understanding GPS: principles and applications*. Artech House, Boston, 2006. ISBN 1580538940.

[26] Email Marketing Reports. Smartphone statistics and market share. http://www.email-marketing-reports.com/wireless-mobile/smartphone-statistics.htm. [Online; accessed 7-July-2012].

[27] ETSI. Location Services (LCS); Mobile Station (MS) - Serving Mobile Location Centre (SMLC) Radio Resource LCS Protocol (RRLP). TS 144 031V8, European Telecommunications Standards Institute (ETSI), June 2007. URL http://www.etsi.org/deliver/etsi_ts/144000_144099/144031/07.05.00_60/ts_144031v070500p.pdf.

[28] ETSI. Universal Mobile Telecommunications System (UMTS); Location Measurement Unit (LMU) performance specification. TS 125.111, European Telecommunications Standards Institute (ETSI), Apr. 2008. URL http://www.etsi.org/deliver/etsi_ts/125100_125199/125111/07.01.00_60/ts_125111v070100p.pdf.

[29] FCC. Wireless E911 Location Accuracy Requirements; E911 Requirements for IP-Enabled Service Providers. https://www.federalregister.gov/articles/2010/11/02/2010-27579/wireless-e911-location-accuracy-requirements-e911-requirements-for-ip-enabled-service-providers. [Online; accessed 19-July-2012].

[30] J. Figueiras and S. Frattasi. *Mobile Positioning and Tracking: From Conventional to Cooperative Techniques*. Wiley, 2010. ISBN 0470694513.

[31] Google. Location-based services. http://support.google.com/maps/bin/answer.py?hl=en&answer=1725632. [Online; accessed 18-July-2012].

[32] GPS World. European Commission Report on Galileo Estimates $ 1 Trillion in Europe Depends on SatNav. http://www.gpsworld.com/gnss-system/news/european-commission-report-galileo-estimates-1-trillion-europe-depends-satnav-10950. [Online; accessed 27-June-2012].

[33] J. G. Grimes. Global positioning system standard positioning service performance standard. Online, Sept. 2008. URL http://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf.

[34] GSMA. Brief History of GSM & the GSMA. http://www.gsma.com/aboutus/history/. [Online; accessed 7-July-2012].

[35] X. Guan, D. Hu, and J. Chen. Design and implementation of the acquisition circuit in software GPS receiver. In *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pages 4 pp.–4, nov. 2005. doi: 10.1109/MTAS.2005.243823.

[36] T. Halonen, J. Romero, and J. Melero. *GSM, GPRS and EDGE Performance: Evolution Toward 3G/UMTS*. John Wiley & Sons, 2002. ISBN 0470844574.

[37] N. Harper. *Server-side GPS and assisted-GPS in Java*. Artech House, Boston, 2010. ISBN 9781607839859.

[38] G. Heine. *GSM Networks: Protocols, Terminology and Implementation (Artech House Mobile Communications)*. Artech House Publishers, 1998. ISBN 0890064717.

[39] L. Heng, G. X. Gao, T. Walter, and P. Enge. GPS Ephemeris Error Screening and Results for 2006-2009. *ION Institute of Navigation Global Navigation Satellite Systems Conference*, 2010.

[40] D. Hogrefe. Mobile Communication - Wireless Telecommunication Systems - University of Goettingen. http://medien.e-learning.uni-goettingen.de/daten/990092/20111/2726/15-pdf/04_Wireless-Telecommunication_I_V2.pdf, 2012. [Online; accessed 5-April-2012].

[41] M. Ibnkahla. *Signal Processing for Mobile Communications Handbook*. CRC Press, 2004. ISBN 084931657X.

[42] P. A. Iglesias. Linearization. http://www.ece.jhu.edu/~pi/Courses/454/NotesA.pdf. [Online; accessed 27-June-2012].

[43] ip.access ltd. GSM-over-IP picocells for in-building coverage and capacity, 2005. URL http://www.hexazona.com/nexwave/docs/ipaccess/nanoBTS;1800-1900.pdf.

[44] ip.access ltd. The world's most deployed picocell. http://www.ipaccess.com/en/nanoGSM-picocell, 2007. [Online; accessed 3-April-2012].

[45] ip.access ltd. nanoBTS Installation Manual, 2009. URL http://subversion.assembla.com/svn/bxpgfKRFar3O9EeJe5afGb/PP/ipaccess/NGSM_INST_300_nanoBTS_Install_v3_0.pdf.

[46] ITU. Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation. TS ITU-T X.680, International Telecommunication Union (ITU), July 2002. URL http://www.itu.int/ITU-T/studygroups/com17/languages/X.680-0207.pdf.

[47] ITU. Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER). TS ITU-T X.691, International Telecommunication Union (ITU), July 2002. URL http://www.itu.int/ITU-T/studygroups/com17/languages/X.691-0207.pdf.

[48] D. C. Jefferson and Y. E. Bar-Sever. Accuracy and Consistency of Broadcast GPS Ephemeris Data. *ION Institute of Navigation International Technical Meeting*.

[49] C. Kee and B. Parkinson. Wide area differential gps (wadgps): future navigation system. *Aerospace and Electronic Systems, IEEE Transactions on*, 32(2):795 – 808, april 1996. ISSN 0018-9251. doi: 10.1109/7.489522.

[50] P. A. Kline. *Atomic Clock Augmentation For Receivers Using the Global Positioning System*. PhD thesis. URL http://scholar.lib.vt.edu/theses/available/etd-112516142975720/.

[51] K. W. Kolodziej and J. Hjelm. *Local Positioning Systems: LBS Applications and Services*. CRC Press, 2006. ISBN 0849333490.

[52] C. Kozierok. *The TCP/IP guide : a comprehensive, illustrated Internet protocols reference*. No Starch Press, San Francisco, 2005. ISBN 159327047X.

[53] A. Küpper. *Location-Based Services: Fundamentals and Operation*. Wiley, 2005. ISBN 0470092319.

[54] C. Ma, G. Lachapelle, and M. E. Cannon. Implementation of a Software GPS Receiver. In *Proceedings of ION GNSS 2004 (Session A3), Long Beach, CA*, sep. 2004. URL http://plan.geomatics.ucalgary.ca/papers/04gnss_ion_cmaetal.pdf.

[55] A. Malik. *RTLS for dummies*. For Dummies John Wiley distributor, Hoboken, N.J. Chichester, 2009. ISBN 9780470398685.

[56] S. MALM and L. OSBORNE. Mobile phone companies can predict future movements of users by building a profile of their lifestyle. http://www.dailymail.co.uk/sciencetech/article-2190531/Mobile-phone-companies-predict-future-movements-users-building-profile-lifestyle.html. [Online; accessed 29-August-2012].

[57] Martin Zwilling, Forbes Magazine. Location-Based Services are a Bonanza for Startups. http://www.forbes.com/sites/martinzwilling/2011/01/31/location-based-services-are-a-bonanza-for-startups/. [Online; accessed 7-July-2012].

[58] K. Meier. Netzwerkmonitor fuer die Ortung in GSM-Netzen. Master's thesis, University of Freiburg, 2010.

[59] Motorola. Overview of 2G LCS Technnologies and Standards. Technical Report LCS-010019, 3GPP TSG SA2 LCS Workshop, Jan. 2001. URL ftp://ftp.3gpp.org/workshop/Archive/0101LCS/Docs/PDF/LCS-010019.pdf.

[60] C. Müller, L. Wan, and D. Hrg. Dealing with wandering: a case study on caregivers' attitudes towards privacy and autonomy when reflecting the use of lbs. In *Proceedings of the 16th ACM international conference on Supporting group work*, GROUP '10, pages 75–84, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0387-3. doi: 10.1145/1880071.1880082. URL http://doi.acm.org/10.1145/1880071.1880082.

[61] Nokia. Detailed specifications for the Nokia N95. http://www.nokia.com/us-en/products/phone/nokia-n95/specifications/, 2006. [Online; accessed 29-August-2012].

[62] Nokia. Detailed specifications for the Nokia E71. http://www.nokia.com/in-en/products/phone/e71/specifications/, 2008. [Online; accessed 29-August-2012].

[63] T. Ogunfunmi. *Adaptive Nonlinear System Identification: The Volterra and Wiener Model Approaches (Signals and Communication Technology)*. Springer, 2007. ISBN 0387263284.

[64] osmocom. OpenBSC build guide. Web. URL http://openbsc.osmocom.org/trac/wiki/Building_OpenBSC. [Online; accessed 22-May-2012].

[65] Osmocom. Openbsc. http://openbsc.osmocom.org/trac/, 2012. [Online; accessed 30-August-2012].

[66] Osmocom. Openbsc - openbsc. http://openbsc.osmocom.org/trac/wiki/OpenBSC, 2012. [Online; accessed 30-August-2012].

[67] PERICOM. Choice of TCXO for GPS Design. Technical report, Pericom Semiconductor Corporation, 3545 North First St., San Jose, CA 95134, USA, 2008. URL http://www.pericom.com/pdf/applications/AN335.pdf.

[68] Phil Mann. Timing synchronization for 3G wireless. http://www.eetasia.com/ART_8800354031_590626_NT_14db7f7f.HTM. [Online; accessed 14-July-2012].

[69] RangeNetworks. Openbts. http://wush.net/trac/rangepublic, 2012. [Online; accessed 30-August-2012].

[70] A. Razavi, D. Gebre-Egziabher, and D. Akos. Carrier loop architectures for tracking weak GPS signals. *Aerospace and Electronic Systems, IEEE Transactions on*, 44(2):697–710, april 2008. ISSN 0018-9251. doi: 10.1109/TAES.2008.4560215.

[71] E. D. Rights. Police frequently uses silent sms to locate suspects. http://www.edri.org/edrigram/number10.2/silent-sms-tracking-suspects, 2012. [Online; accessed 2-September-2012].

[72] R. Sharp. *Principles of protocol design*. Springer, Berlin, 2008. ISBN 3540775404.

[73] S. Soliman, S. Glazko, and P. Agashe. GPS receiver sensitivity enhancement in wireless applications. In *Technologies for Wireless Applications, 1999. Digest. 1999 IEEE MTT-S Symposium on*, pages 181–186, feb 1999. doi: 10.1109/MTTTWA.1999.755159.

[74] F. SOYEZ. Getting the message? police track phones with silent sms. http://owni.eu/2012/01/27/silent-sms-germany-france-surveillance-deveryware/, 2012. [Online; accessed 1-September-2012].

[75] Spirent Communications. Spirent Expands Leadership in Testing E911 and Location Based Services for LTE Networks. http://www.spirent.com/About-Us/News_Room/Press-Releases/2012/2012_02_21_Spirent_LBS_LTE_Testing. [Online; accessed 7-July-2012].

[76] W. Stevens. *TCP/IP illustrated*. Addison-Wesley Pub. Co, Reading, Mass, 1994. ISBN 9780201633467.

[77] J. Stewart. *Calculus*. Brooks Cole, 2011. ISBN 0538497815.

[78] The Register, UK News magazine. FCC to fine network operators who can't find customers. http://www.theregister.co.uk/2007/08/31/e911_fine/. [Online; accessed 7-July-2012].

[79] u-blox AG. GPS Antennas: RF Design Considerations for u-blox GPS Receivers. https://www.u-blox.com/images/downloads/Product_Docs/GPS_Antennas_ApplicationNote%28GPS-X-08014%29.pdf, 2009. [Online; accessed 29-August-2012].

[80] u-blox AG. UBX-G5010, G5000/G0010. http://www.texim-europe.com/promotion/560/ubx-g5010%20datasheet_te.pdf, 2009. [Online; accessed 5-April-2012].

[81] u-blox AG. The GPS Dictionary. http://www.u-blox.com/images/stories/the_gps_dictionary.pdf, 2010. [Online; accessed 1-August-2012].

[82] D. Wehrle. Open Source IMSI-Catcher. Master's thesis, University of Freiburg, 2010.

[83] G. Xu. *GPS: Theory, Algorithms and Applications*. Springer, 2007. ISBN 3540727140.

[84] R. M. Zahoransky. Localization in GSM Mobile Radio Networks. Master's thesis, University of Freiburg, 2011.

[85] V. Zeimpekis, G. M. Giaglis, and G. Lekakos. A taxonomy of indoor and outdoor positioning techniques for mobile location services. *SIGecom Exch.*, 3(4):19–27, Dec. 2002. ISSN 1551-9031. doi: 10.1145/844351.844355. URL http://doi.acm.org/10.1145/844351.844355.

[86] J. Zhang and G. de la Roche. *Femtocells: Technologies and Deployment*. Wiley, 2010. ISBN 0470742984.