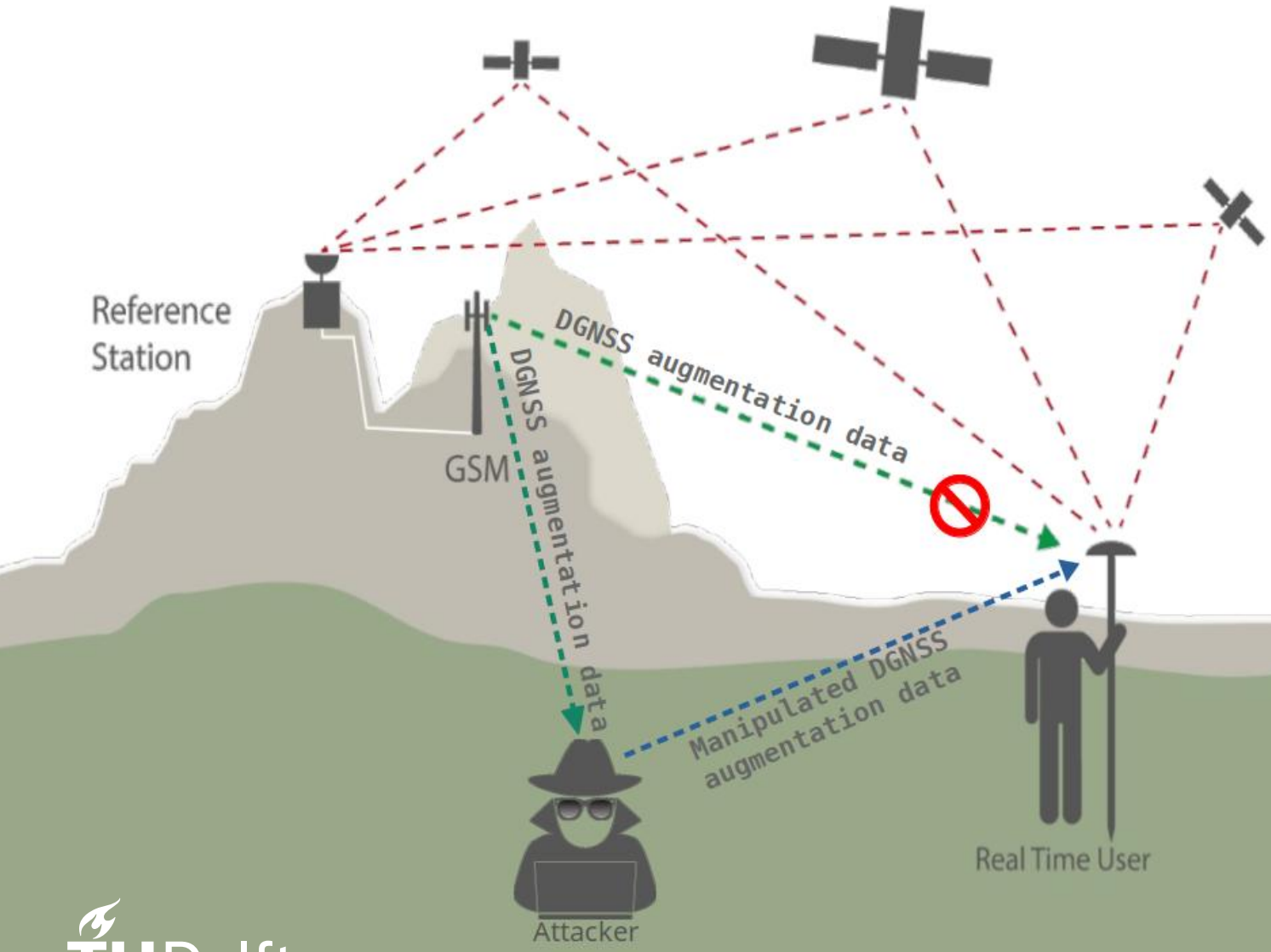
RTK-GNSS augmentation data spoofing

P.M. van Tol



TU Delft

RTK-GNSS augmentation data spoofing

by

P.M. van Tol

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday August 31, 2020 at 14:00

Student number:	4624165	
Project duration:	March 1, 2019 – August 31, 2020	
Thesis committee:	Dr. ir. C.C.J.M. Tiberius,	TU Delft, supervisor
	Prof. dr. ir. P.J.G. Teunissen,	TU Delft
	Prof. dr. ir. P.H.A.J.M. van Gelder,	TU Delft

TU Delft

Preface

This report is written as a final assignment for graduation from the master “Geoscience and Remote Sensing” at the University of Technology in Delft. After graduation as hydrographic surveyor I started in Delft with this master. During the programme it became clear that positioning is an import aspect, not only for surveying, but also in everyday life. When Mr. Tiberius pointed out the subject about augmentation data manipulation I was very interested. After written this report I am glad that I have chosen this topic.

The topic is interesting since it is often is questioned how precise one can position using a GNSS system. It is common to use a differential setup to achieve the highest precision, but it is not often questioned how safe this setup is. In a time where internet security and privacy are hot topics, it is interesting to see that the security and validity of the received augmentation data is not questioned.

While writing this report it is assumed that the reader has basic knowledge of satellite navigation positioning. The report is written with an audience of colleague master students in mind.

I would like to thank Dr. P.R. Zimmermann, who is part of a community of researchers that work together to address cybersecurity challenges, for his explanation about different types of cyber security. It helped me to understand the basics which where needed to understand the threat of a potential hack on the augmentation data. Thanks also to Jean-Paul Henry from the company 06-GPS for an extensive explantation about the implementation of RTK and network-RTK. This helped to understand the practical implementation for users that want to use RTK. My sincere appreciation for Dr. Tomoji Takasu for his quick and valuable answer on an issue with the RTKlib implementation of differential processing.

The completion of this report without the work of Dr.ir. Hans van der Marel, with his many Matlab scripts, would be a lot harder.

I would like to extend my deepest gratitude to Prof.dr.ir. Peter Teunissen for his help in understanding and clear explanations about the broad subject of GNSS and for his feedback during the writing process. Also I would like to express my deepest appreciation to Dr.ir. Christian Tiberius for all his help as a daily supervisor and his patience with answering all my questions.

In the end I would thank my family and friends for their support and encouragement during this project.

Finally, all I can do is hope that you will enjoy reading the report as much as I enjoyed writing it.

P.M. van Tol
Harderwijk, August 2020

Summary

The use of Global Navigation Satellites Systems is increasing rapidly. More and more applications use positioning and/or timing information form a Global Navigation Satellite System (GNSS). Also more and more people and applications rely on high-precision positioning based on GNSS. The high-precision solution of GNSS is achieved with the use of example augmentation data. For example real-time kinematic (RTK)-GNSS enables centimetre-level positioning. Commonly the augmentation data is sent with the use of internet. At the moment an unsecure internet link is used to sent this augmentation data from the reference station to the user.

The aim of this study was to find out if it is possible to manipulate the augmentation data for DGNSS using a cyber attack without being detected, and what the consequences could be for the final estimated parameters of interest. The parameters of interest can be the position and/or the timing.

The manipulation of the augmentation data is also described as spoofing of the augmentation data. GNSS spoofing is not new, but until now the spoofing attacks concentrate on the GNSS radio signals. This is different with what is done in this study, where the spoofing of the augmentation data itself is analysed.

The research first focusses on the data link between the reference station and the receiver. The augmentation data is sent using the Networked Transport of RTCM via Internet Protocol (NTRIP). What is found is that this is an unsecure connection. For an attacker it is possible to use a man-in-the middle attack, where the augmentation data is sent from the reference station, via the hacker, to the user. The data is not encrypted and therefore it is possible for the hacker to see and alter the data. This is shown in Figure 1.

Based on a man-in-the-middle attack this study found that it is possible to manipulate the DGNSS augmentation data, without detection. The model that is used to manipulate the augmentation data is based on a Single Point Positioning model. As long as the manipulation is in the range of the design matrix (A) of the used model, it is not detectable. This means that the manipulation only contributes to the so called influential bias and not, or minimal, to the testable bias. As the name suggest, the result of this manipulation is that the final solution is manipulated due to the effect in the influential bias, and without detection since the testable bias is not changed.

GNSS processing is based on non-linear observation equations. This means that those models are linearised before the final solution is estimated based on the least squares estimation. The effect of this non-linearity is minimal, but it means that a (very) small part of the manipulation contributes to the testable bias. This study points out that this small increase of the testable bias is insignificant when the observations are tested based on an overall model test and the w-test.

The manipulation is defined in the parameter space, $\tilde{\mathbf{x}}$. The manipulation in observation space, which when added to the augmentation data, is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$, where A is the design matrix. This means that the manipulation is in the range space of A ($\tilde{\mathbf{y}} \in \mathcal{R}(A)$). The difference between the GNSS solutions, one that is based on the original observations $\hat{\mathbf{x}}$ and one solution that is based on the manipulated observations $\hat{\tilde{\mathbf{x}}}$, is equal to the input manipulation $\tilde{\mathbf{x}}$. This means that the consequence of a possible manipulation of the augmentation data is that the GNSS solution of the user can exactly be manipulated by the hacker.

The limitation of the manipulation is the Single Point Positioning (SPP) solution of the receiver. The observations of the user are not manipulated by the augmentation data, and therefore the user observations can be used to check if there is a significant difference between the SPP and the RTK solution. Also the position of the reference station is known. This can be used to check if there is not a too large difference between the SPP solution and the known position of the reference station. The maximum length of the manipulation $\tilde{\mathbf{x}}$ is approximately 2 to 3 meter.

The conclusion of this study is that it is possible to spoof the augmentation data when NTRIP is used to sent the augmentation data. Furthermore, the consequence of augmentation data spoofing is that it can be exactly manipulated by the hacker, based on a certain direction and distance, as long as the magnitude of the manipulation is in the order of 2 to 3 meter.

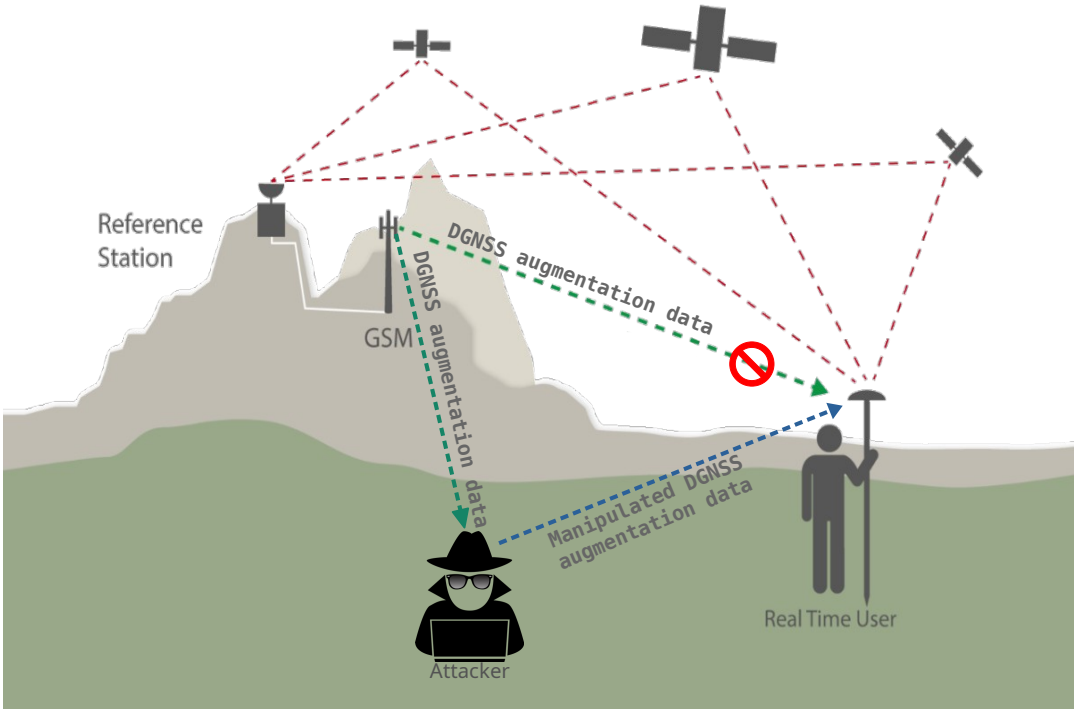


Figure 1: Schematic of DGNSS augmentation data spoofing.

Contents

List of Figures ix

List of Tables xi

1 Introduction 1
- 1.1 Jamming and spoofing . . . 2
- 1.2 Report structure . . . 4

2 Global Navigation Satellite System 5
- 2.1 GNSS measurement setup . . . 6
  - 2.1.1 Single-point positioning . . . 7
  - 2.1.2 Differential-GNSS code measurement . . . 7
  - 2.1.3 Differential-GNSS phase measurement . . . 8
  - 2.1.4 Timing . . . 9
- 2.2 Observation equations . . . 9
  - 2.2.1 Code measurement . . . 10
  - 2.2.2 Phase measurement . . . 10
- 2.3 Mathematical models . . . 11
  - 2.3.1 Single point . . . 11
  - 2.3.2 Differential GNSS . . . 12
  - 2.3.3 Timing . . . 15

3 Data formats and communication protocols used for GNSS 17
- 3.1 Data formats . . . 17
  - 3.1.1 Radio Technical Commission for Maritime Services (RTCM) SC-104 data format . . . 17
  - 3.1.2 Receiver INdependent EXchange (RINEX) format . . . 19
- 3.2 Communication protocols . . . 20
  - 3.2.1 Transmission Control Protocol (TCP) . . . 20
  - 3.2.2 Hypertext Transfer Protocol . . . 20
  - 3.2.3 Networked Transport of RTCM via Internet Protocol (NTRIP) . . . 21
- 3.3 Man-in-the-middle attack and countermeasures . . . 21
  - 3.3.1 Man-in-the-middle attack . . . 21
  - 3.3.2 Data interception . . . 22
- 3.4 Countermeasures . . . 23

4 Data manipulation theory 25
- 4.1 Parameter estimation . . . 25
  - 4.1.1 Best Linear Unbiased Estimator . . . 25
  - 4.1.2 Statistical testing . . . 27
- 4.2 Data manipulation . . . 30
- 4.3 2D example non-linear model . . . 30
- 4.4 Differential model . . . 38

5 Implementation of GNSS augmentation data manipulation 41
- 5.1 The manipulation procedure . . . 41
- 5.2 Linearisation position . . . 43
- 5.3 Single point positioning accuracy assessment . . . 44

6 GNSS manipulation results **47**
6.1 Base station manipulation . . . 48
6.2 Single point positioning . . . 50
6.3 D-GNSS . . . 51
6.4 RTK . . . 53
6.4.1 Default RTKlib configuration . . . 54
6.5 Timing . . . 55
6.6 Arbitrary manipulation . . . 56
6.7 Case study “Car experiment A13” . . . 58

7 Conclusion **63**
7.1 Recommendations . . . 63
7.2 Augmentation data spoofing detection . . . 64
7.2.1 Single epoch . . . 64

A Taylor series for square root **67**
A.1 Taylor series . . . 67
A.2 Taylor series root function . . . 68
A.2.1 Ratio test . . . 68

B Configuration files RTKlib **71**
B.1 Single Point Positioning . . . 71
B.2 Differential Positioning . . . 73
B.2.1 Pseudo range differences . . . 73
B.2.2 Carrier phase differences . . . 74

Bibliography **77**

List of Figures

1 Schematic of DGNSS augmentation data spoofing. vi

2.1 Pseudorange and phase observation. The image shows the carrier wave (blue), the PRN code sequence (orange) and the modulated carrier wave (green), the drawing is not to scale. The satellite is shown as a black cross and the receiver is the red triangle. 6
2.2 DGNSS overview [5] 7

3.1 Schematic of a Man-in-the-middle attack, after [15] 22

4.1 Schematic of the Gauss-Newton iteration. The iteration stops when $||\Delta\hat{\mathbf{x}}_{i+1}||^2_{Q^{-1}_{\hat{x}\hat{x}}} \leq \delta$ where δ is the threshold value. The value of δ can be chosen in the order of $\mathcal{O}(1e-10)$ or lower. The final value of $||\Delta\hat{\mathbf{x}}_{i+1}||^2_{Q^{-1}_{\hat{x}\hat{x}}}$ tells nothing about the accuracy of $\hat{\mathbf{x}}_{i+1}$, only that $L_F(\hat{\mathbf{x}}_{i+1}) \approx F(\hat{\mathbf{x}}_{i+1})$. 27
4.2 The values for the test-statistics for both the unmanipulated and the manipulated cases. In both cases and for both tests (overall model test and w-test) the test-statistic does not exceeds the critical value (the dotted line). The values can be found in Table 4.1 32
4.3 The BLUE estimation for both the original example on the left and the manipulated example on the right. The standard deviation of the observables is 0.02 meter. The manipulation in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. 33
4.4 The values for the test-statistics for both the unmanipulated and the manipulated cases. The null hypothesis is rejected since the test-statistic exceeds the critical value. The values can be found in Table 4.2 for H_0 and Table 4.3. 35
4.5 The observations and the manipulated observations for the new simulation with $m = 1080$ points equally distributed around $(0, 0)$ with a distance of 1000 meter. The manipulation is $\tilde{\mathbf{x}} = [1, 2]^T$. 36
4.6 A measurement setup with 1080 known points equally distributed around (0,0). The initial linearisation position $\mathbf{x}_0 = [0.5, 1]^T$ and the manipulation vector is $\tilde{\mathbf{x}} = [1, 2]^T$. . . . 37
4.7 A measurement setup with 1080 known points equally distributed around $(0, 0)$. The scaled clock bias is also estimated in meters. 37
4.8 A measurement setup with 1080 known points equally distributed around $(0, 0)$. The scaled clock bias is also estimated in meters. The initial guess $\mathbf{x}_0 = [0.5, 1, 0]^T$ and the manipulation vector is $\tilde{\mathbf{x}} = [2, 1, 0]^T$. 38
4.9 The measurement setup with 1080 known points, equally distributed around (0,0), is used to calculate the w-test and the influence of the manipulation. The noise is simulated using a normal distribution with a zero mean and a standard deviation of 0.02 meter. . . 39

5.1 Augmentation data spoofing procedure 42
5.2 The overall model test for the manipulation vector based on different design matrices. The overall model test is for one epoch, for 9 observations. The critical value is $T_{q=9-4=5} = 11.1$. The maximum value for the overall model test is 3.99. The used standard deviation per observation is 0.003 meter. 44
5.3 Histogram of the standard deviation in NEU (left) and in ECEF (right). 46
5.4 The SPP solution $\hat{\mathbf{x}}$ compared with the known position $\mathbf{x}$. On the left the difference between the SPP solution and the known position, $\Delta\hat{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$, is shown. On the right the overall model test is shown, based on the hypothesis $H_0 : \mathbf{x} = \hat{\mathbf{x}}$. A standard deviation of 1.5 meter is used, corresponding to the RTKlib value. 46

6.1 Location of base station and rover in Delft. 47
6.2 The manipulation per satellite ($\tilde{\mathbf{y}}$) and a skyplot of all observed satellites. 48

6.3 Difference between position based on the manipulated and the original RINEX file, ($\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}$). On the left a histogram is shown with the differences in X and Y. The expectation value of the difference between the solutions based on the original and the manipulated data should be the manipulation vector. 50
6.4 Effect of the augmentation data manipulation. The correct baseline ($\mathbf{x}_{1,2}$) is coloured green. The manipulation vector ($\tilde{\mathbf{x}}$) is coloured red, and has a north–east direction. The manipulated baseline ($\tilde{\mathbf{x}}_{1,2} = \mathbf{x}_{1,2} - \tilde{\mathbf{x}}$) is is coloured blue. The manipulated baseline is a baseline from a fake reference position, the white circle, to the rover. The manipulated baseline in combination with the reference position $\mathbf{x}_1$ result in a manipulated rover position ($\tilde{\mathbf{x}}_2$), the red filled circle. The manipulated rover positioning is manipulated in the opposite direction of the manipulation vector. 51
6.5 Scatter plot of the position of the user based on DGNSS-code positioning. 52
6.6 The estimated manipulation vector in two different plots 52
6.7 Scatter plots with the RTK solution. 53
6.8 The resulting manipulation vector in two different plots. 53
6.9 The RTKlib solution in RTK mode with the default configuration. 54
6.10 Difference of the sum of squared residual based on the original and manipulated observations. Left the pseudorange range residuals are used, and for the figure on the right the carrier–phase observations are used. 55
6.11 RTKlib solution with a manipulation in the receiver clock bias. 56
6.12 The RTK result of the original and the manipulated data. The manipulation is only in the clock bias. 56
6.13 The Single-point positioning (SPP) solution after an arbitrary manipulation of the observations. 57
6.14 The Real Time Kinematic (RTK) solution of a random manipulation. 58
6.15 The GNSS solution status for the original and arbitrary manipulated data. What can be seen is that there are almost no fixed solutions when the data is arbitrarily manipulated. 58
6.16 Antenna set-up on the roof of the test vehicle [37]. The used antenna for this study is in the front left position. 59
6.17 Car experiment A13 [18] data on top of a satellite image, using the UTM31N projection (EPSG:32631). The manipulation vector is approximately perpendicular to the highway, with an azimuth of $75°$ and with a length of 2.24 and 7 meter. The green dots represents the original data (the car taking the right most lane), the red dots represents the manipulation of 2.24m and the blue dots represents the manipulation of 7.00m. The manipulation of the rover position is in the opposite direction of the manipulation vector, see Figure 6.4. 59
6.18 Results for the manipulation of the A13 dataset. The upper two figures shows the GNSS mode for a manipulation length of 2.24 meter and 7.00 meter respectively. The bottom two plots shows the resulting manipulation vector for the x, y and z components, and the amount of used satellites. 61

7.1 Detection threshold for one epoch and the mean variance matrix. The area where the spoofing is not detected, with an alpha level of 0.05, is shown in black. 65

List of Tables

4.1 The used positions and simulated observations for the 2D example. The manipulation vector $\tilde{\mathbf{y}}$ is defined as $\tilde{\mathbf{y}} = A_1\tilde{\mathbf{x}}$ (A_1 is the first design matrix during the iteration). $\mathbf{x}$ denotes the true position (and clock bias in meters). The used manipulation vector in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The standard deviation of the observables is 0.02 meter. Both the overall model test and the w-test are not rejected, thus H_0 is assumed to be true, even when the data is manipulated. 32

4.2 The used positions and simulated observations for the 2D example. The manipulation vector $\tilde{\mathbf{y}}$ is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$. And the used manipulation vector in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The standard deviation of the observables is 0.02 meter. There is an outlier of 0.2 meter in the third measurement. The overall model test is in both cases rejected. The w-test shows the biggest bias in the third measurement and is also rejected. The test-statistics above the critical value are showed in red. The test-statistics and the final estimates are based on the H_0 hypothesis. 34

4.3 The used positions and simulated observations for the 2D example. The manipulation vector $\tilde{\mathbf{y}}$ is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$. And the used manipulation vector in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The standard deviation of the observables is 0.02 meter. There is an outlier of 0.2 meter in the third measurement. The overall model test and the w-tests are in both cases accepted. The test-statistics and the final estimates are based on the H_a hypothesis. The bias ∇ is added to the model and thus also estimated. The third measurement is grayed since this measurement does not contribute to parameters of interest anymore, but it is used to estimate ∇. 35

5.1 Covariance matrix from 24 hour observation, 30 seconds interval, marker point Delft–16. The Single-point positioning (SPP) solution is calculated in Earth Centerd Earth Fixed (ECEF) and in NEU (North, East, Up). 45

6.1 SPP manipulation with different input biases (first four columns). The coordinates (x, y and z) are given in meters. The clock biases are in nano seconds. After the input manipulation vector, the resulting biases are shown, next the normalized bias is shown. The normalized bias shows that there are no resulting biases as result of a larger manipulation. The last two columns shows the norm of the input bias and of the estimated bias. 51

6.2 DGNSS-code manipulation with different input biases (first four columns). The coordinates (x, y and z) are given in meters. The clock biases are described in nano seconds. After the input manipulation vector, the resulting biases are shown, next the normalized bias is shown. The last two columns shows the norm of the input bias and of the resulting bias. 52

6.3 RTK augmentation data manipulation with different input biases (first four columns). The coordinates (x, y and z) are given in meters. The clock biases are described in nano seconds. After the input manipulation vector, the resulting biases are shown, next the normalized bias is shown. The last two columns shows the norm of the input bias and of the estimated bias. 54

6.4 The manipulation vector and the estimated manipulation vector in ECEF for the A13 dataset. 60

1 Introduction

In the near future more and more organizations and people will rely on high-precision positioning using Global Navigation Satellite System (GNSS) solutions [29], but can it be assumed that this is safe for critical applications? GNSS is used for positioning, navigation and timing. The reason that high-precision positioning will be more often used is because there are more and more autonomous vehicles, like drones and cars, and the demand for more high-precision positioning is not limited to those applications. For example high-precision GNSS solutions are used for surveying, the precise clock of GNSS satellites can be used for timing and GNSS solutions can be used for agriculture activities, also autonomously.

An approach that is often used to increase the accuracy of Global Navigation Satellite System (GNSS) solutions is the use of differential GNSS. To use differential GNSS, extra external information is needed compared to the stand alone solution or a single point GNSS solution. This extra information is called the augmentation data and is needed for a more accurate GNSS solution. The augmentation data are determined using a known point near the user. In short, the augmentation data contain corrections to improve the precision of the GNSS solution. Before the user can use the augmentation data, the augmentation data need to be sent to the user. This can be done using radio signals, but it is also possible to send the augmentation data over the internet. In the beginning of the Real Time Kinematic (RTK) era it was common for each user to use his or her own reference station and set up a radio link for communication between the receiver and the reference station. With the availability of mobile internet the more often used method lately is to send the augmentation data using the mobile network.

The fact that real time high-precision positioning becomes more important raises also the question about the vulnerabilities of differential GNSS. One of the vulnerabilities is how the augmentation data are sent to the user. In this case the use of real time high-precision positioning means the use of RTK. GNSS-RTK uses relative positioning to position a rover (the user) relative to a reference station, using a baseline. When absolute coordinates are required, the coordinates of the static reference station are used in combination with the baseline.

As described in [22] GNSS in general is vulnerable for different kind of threats. One of those threats is jamming. A GNSS jammer sends radio signals at the same frequency as GNSS satellites and overrules the satellites' signals so that the original signals are no longer received by the receiver. Another threat is spoofing, which is used to mislead the target GNSS receiver, by intentionally transmitting fake GNSS radio signals to the target.The receiver still calculates positions, but with incorrect observations. This can be done for example by delaying the original GNSS signals, or manipulating the data that is sent from the satellites, the so-called navigation message. Both jamming and spoofing as described here are done on the radio signals. In [22] it is also described that sometimes it is possible to detect spoofing. Most of the research that is done about spoofing is on the GNSS radio signals.

With the commonly used communication for the augmentation data, over the internet, there is another threat, namely an attack on the augmentation data. This means that the observations done by the receiver of the user are not changed, but the augmentation data is not valid anymore. This is called a cyber attack on the augmentation data.

In the past it has been shown that radio signal spoofing can be successful. One of the examples in which spoofing was successful was at a super yacht in 2013 [16]. Other reports about spoofing are

about spoofing in the Black-Sea, and another occurrence of spoofing is reported in Iran [2, 23]. Spoofing attacks can be dangerous when GNSS is used for modern warfare, like the use of autonomous drones, but also when GNSS solutions are used for civil applications, like timing for telecommunication and power plants. There are also various methods determined to detect signal spoofing, which are mentioned in [22], [27] and [28]. Most of those methods focus on the physical aspect of radio signals, but there are other methods [27] that describes for example the use of external sensors and a Kalman filtering to detect a spoofing attack. The shortcoming of the use of for example IMU's (Inertial Measurement Unit) is that the accuracy degrades with time.

Another aspect of spoofing, like the attack with the super yacht [16], can also be applied to car navigation as described in [39]. This paper is about car navigation and how to manipulate the GNSS position for navigation, but when high-precision positioning is used for autonomous vehicles then it is also possible to perform a cyber attack by manipulating the augmentation data. When GNSS solutions are used for self driving cars this manipulation does not have to be very large to create extremely dangerous situation.

The augmentation data spoofing involves cyber-attacks, as long as the augmentation data is sent over the internet. This can be a DDoS attack, so that no augmentation data is received. This can be compared with signal jamming. No signal, or in the case of a cyber-attack, no augmentation data is received. Those kind of attacks are easily detectable. Another method is to manipulate the augmentation data, which can be compared with spoofing of the radio signal. Those kinds of attacks are harder and take more effort to detect. Such an attack can be done using for example a man-in-the-middle attack. At the moment the augmentation data are typically sent to the user via an unsecure connection. This makes it possible to see and alter the data with relatively little effort.

When more and more applications require high-precision positioning it means that it becomes more important to understand the possibilities to perform such an attack, how this probably will be done and its potential impact. The threat of those kinds of cyber attacks is that the only thing that one will need is a computer connected to the internet. This is different from radio spoofing as described earlier, where someone has to use radio transmitters of relatively high quality to perform a spoofing attack. Therefore there are a lot more people able to carry out such a cyber attack, compared with the relatively difficult signal spoofing.

At this moment it is not clear if it is possible to perform a cyber attack on the augmentation data without being detected. It is unclear whether such an attack can be detected by the Receiver Autonomous Integrity Monitoring (RAIM), for instance that observations, corrected by the (manipulated) augmentation data are rejected by the Fault Detection and Exclusion (FDE) procedure.

The objective of this research is to *determine if one can manipulate the augmentation data for differential GNSS using a cyber attack without being detected, and what the consequences are for the final estimated parameters of interest*. To find out if this is possible first the mathematical model used by the user receiver should be identified. This model is used to manipulate the parameters of interest without increasing the so-called testable bias. The next step is to analyse the impact on the final result (can one intentionally manoeuvre the user off track) and the possible effect on the solution quality. Finally it should be determined to what extent a cyber attack is possible for the used data-communication and what data are available for an attacker.

The scope of this graduation project is limited on account of cyber attacks and counter measures. This thesis presents an introduction for cyber attacks and possible existing solutions. During this project the GNSS system is limited to Global Positioning System (GPS).

To complete this graduation research different tools were used. A literature study is performed to find the information that is already available about this subject. The opportunity to interview experts in the fields of GNSS and cyber security is used. To implement and test the ideas in practice post processing software, RTKlib, is used. The post processing software was used with such settings that it is comparable with real time data processing. Further software packages such as Matlab and Qgis are used to manipulate and analyse the data.

1.1. Jamming and spoofing

GNSS signals are publicly available and have very low power once they arrive at the Earth's surface, the power varies between -163 dB W and -152dB W. It is thus possible for everybody to know what a GNSS signal looks like since it is not encrypted. It is possible to create your own data and fake a

GNSS signal so that a nearby receiver observes this fake signal as if it was a legitimate signal from a satellite.

Jamming and spoofing are two terms used often closely together. When an attacker wants to deny the victim the access of valid GNSS solutions both jamming and spoofing can be used. By jamming all GNSS radio signals are blocked and the victim is not able to calculate GNSS solutions anymore. Spoofing is the act of faking the GNSS solution so that the observations used by the victim are not trustworthy anymore, but without the victim knowing this.

People can use spoofing with bad intentions, but it is also used by governments. For example spoofing can be used as a countermeasure for drones [2]. Most of the time the purpose of spoofing is to let the user believe that the calculated position is correct and reliable, while in reality it is not.

Another example of spoofing is from [16], where researches gained hostile control of a 65 meter super yacht in the Mediterranean Sea. There they were able to take over control of the yacht, while the crew could compare their GNSS position with other onboard systems, like the compass.

A different report describes a spoofing attack in road navigation [39]. They were able to navigate the victims to locations 1 kilometre away from their original destination. As mentioned in their report, this only works as long as the driver has no knowledge of the area in general.

There are several levels of spoofing. The first method is not actually spoofing, it is called jamming. Jamming sends a signal at the same frequency as GNSS, roughly between 1.2 and 1.6 GHz, so that the receiver is not able to receive the original (weak) signals from the satellite. This is not a spoofing method, because the user knows that the GNSS signals are jammed, since it is impossible to calculate a position without measurements. Jamming can be used in combination with spoofing. When a receiver is first jammed, it loses all its connections with the satellites. The satellite signals have to be locked again, but then the spoofed signal will replace the signals from the satellites.

Spoofing is about transmitting generated or received GNSS signals, while the victim is able to calculate a position. This is the basis of the previous examples. One way to spoof a receiver is to use meaconing. Meaconing uses the received signal and repeat the same signal with its own transmitter. The intensity of the transmitted signal is higher than the intensity of the real signals from the satellites. Therefore the real signal is not measured anymore, but the spoofed signal is measured. When this fake signal is used to calculate the position and velocity the result will be the position and the velocity of the meaconor's receiver antenna. Meaconing can only be used for receivers positioned on a unknown location. For example a static receiver placed on a known position, the receiver can compare the known position with the calculated position.

Another option is to simulate the GNSS signal based on the known satellite locations. This makes it possible to use your own defined bias for the spoofing, instead of the repeat signal.

There are also some countermeasures for spoofing. Next to better hardware, there are also some software solutions. One can use absolute power monitoring, which compares the received signal strength and the expected signal strength. One problem with this is that the signal strength can vary due to atmospheric and solar interference. Doppler peak monitoring can also detect spoofing. The received satellite frequency changes due to the velocity of the satellite. Difference in the Doppler shift between the real and simulated constellation can be an indicator that the receiver is spoofed.

So far the spoofing of the GNSS radio signals are discussed. Another way of spoofing is on the augmentation data. The augmentation data spoofing requires a different approach than radio spoofing, for example no radio transmitter is used to fake the GNSS radio signals, but a computer is used to hack the data link between the reference station and the user. Also the countermeasures have to be completely different when one tries to detect augmentation data spoofing. Augmentation data spoofing was mentioned in [30], but was not discussed in detail.

Augmentation data is also considered as a tool to detect GNSS radio spoofing [21]. The assumption that is made in that paper is that the augmentation data is reliable and is not spoofed, which is not always true.

For a DGNSS system it is possible to spoof the final user position on three locations. It is possible to use radio spoofing near the reference station or near the user. This is the more traditional way of spoofing, by faking the GNSS radio signals. The other possibilities is spoofing on the augmentation data. In this study only the latter, spoofing of the augmentation data, is considered. Therefore it is assumed that there is no radio spoofing near the user. Radio spoofing near the reference station is less of an issue, since it does not matter where the reference observations are spoofed, the result is equal, namely manipulated reference observations.

1.2. Report structure

This report is structured as follows. In chapter 2 an explanation about GNSS is given. This chapter contains the different models for the different GNSS modes that are used. Chapter 3 contains information about the used data formats, communication protocols and information about different possible cyber attacks. The general theory about augmentation data manipulation can be found in chapter 4. The implementation of augmentation data manipulation is described in chapter 5 and the results are represented in chapter 6.

2

Global Navigation Satellite System

Global Navigation Satellite System (GNSS) is a positioning, navigation and timing (PNT) system that is based on distance measurements to satellites. GNSS is sometimes confused with Global Positioning System (GPS), but GPS is the American version of GNSS. There are other worldwide systems like GLONASS (Russian), Galileo (European) and Beidou (China). Other systems like NAVIC (India) also use distance measurements to satellites, but those are regional systems. Distance measurements are done based on travel time observations. GNSS can be used for absolute and relative positioning, it can be used static and kinematic, it is available 24 h per day and does work under all weather conditions.

The GNSS system that is used for this report is GPS, therefore the characteristics and principles of GPS are used and they are similar for the other GNSS systems although there may be small differences.

GPS uses at least 24 satellites, divided over six orbits. The orbital planes have an approximate inclination of 55° and the average distance from the Earth's surface to the GPS satellite is 20.240km. GPS is developed by the United States Department of Defence and the U.S. Air Force maintains, develops and operates the system. This is different compared to Galileo which is a civil system, while GPS is a military system. GPS consists of three segments, namely the user segment, space segment and control segment. The user segment are all users that use GPS for positioning, navigation or timing, the space segment are all satellites, and the control segment describes the operator and maintainer of GPS. From the control segment corrections are sent to GPS satellites when needed and they regulate the system so GPS is up and running.

Since GNSS is based on time it is necessary to have accurate time measurements, therefore GPS satellites are equipped with multiple atom clocks. The atomic clocks that are used in GPS satellites do have a drift, this drift is at most 10^{-13} per 24 hours. This is equal to 9 nano second per day ($1\text{e}{-13} \cdot 24\text{h} \cdot 60\text{min} \cdot 60\text{s} = 8.64\text{e}{-9}\,\text{s}\,\text{d}^{-1}$ (second per day)), in distance this is $9\text{ns} \cdot c = 3\text{m}$ where c represent speed of light ($\approx$ $3\text{e}8\,\text{m}\,\text{s}^{-1}$). Each satellite transmits a continuous signal. The base frequency of GPS is 10.23 MHz, the three carrier waves are based on this frequency. The carrier wave of L1 is $f_1 = 154 \times 10.23\text{MHz} = 1575.42\text{MHz}$, the carrier wave of L2 is $f_2 = 120 \times 10.23\text{MHz} = 1227.60\text{MHz}$ and the carrier wave of L5 is $f_5 = 115 \times 10.23\text{MHz} = 1176.45\text{MHz}$. The names L1, L2 and L5 refer to the different bands on which GPS sends its signals, in the part of the so called L-band.

To send digital codes, the Pseudo-Random Noise (PRN) code, a bi-phase modulation is used. Each satellite has its own unique PRN code. Observations are done on the carrier signal, this is called the phase measurement. It is also possible to measure the pseudorange based on the digital code, e.g. the C/A code on L1. The chiprate of the C/A code is 1.023MHz, therefore one bit consists of $\frac{154 \times 10.23}{1.023} = 1540$ radio waves. The total code length is 1023 bits, this is thus equal to $1023 \cdot 1540 = 157540$ cycles, for L1 this means a distance of $157540 \cdot \frac{c}{f1} \approx 300$km.

Figure 2.1 shows the Pseudo-Random Noise (PRN) code, carrier wave and the modulated signal. The modulated signal is a combination between the PRN and the carrier wave. The PRN code is unique for each satellite. It is possible to measure distance based on the PRN code with a precision of ≈3m. The GPS receiver creates a replica of the PRN code of a specific satellite, while tracking the received PRN code. The receiver correlates the received signal with the replica. This is called the tracking loop and it is a continuous process. Based on this tracking loop it is possible to determine the travel time of

the signal.

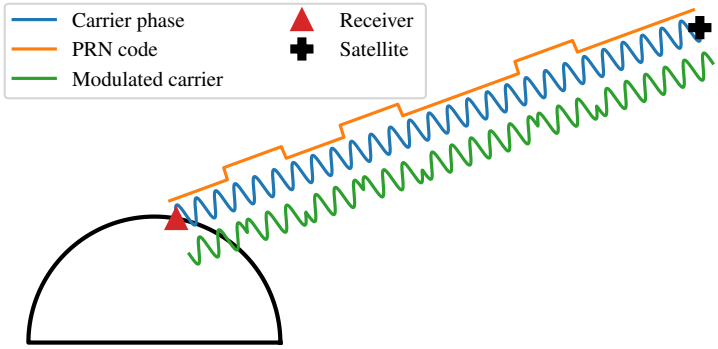


Figure 2.1: Pseudorange and phase observation. The image shows the carrier wave (blue), the PRN code sequence (orange) and the modulated carrier wave (green), the drawing is not to scale. The satellite is shown as a black cross and the receiver is the red triangle.

GPS transmits also the navigation message. The navigation message is used to be able to calculate the positions of the satellites. The navigation message contains the clock parameters, ephemeris data, UTC/ionosphere parameters and the almanac. The clock parameters and ephemeris data is sent every 30 seconds. The UTC/ionosphere parameters and the almanac are defined in subframe 4 and 5, but consist of 25 pages. This means that the whole almanac is send each 12.5 minute.

The distance between the receiver and the satellites are measured. The satellite positions are known. Based on the known positions of the satellites and the measured distances the receiver position and clock bias is calculated.

The accuracy of the final position solution is a result, among other things, of the satellite geometry. In general a result of the geometry is that the vertical accuracy is poorer than the horizontal accuracy by about a factor of 1.5.

The accuracy of the position is not only dependent on the geometry but also on the observation quality. Observations can contain other biases. There are measurement biases like multipath and atmospheric biases. The GNSS system can also contain biases like the satellite position and the satellite clock. The system biases are small enough to be in the noise of a Single-point positioning (SPP) position and therefore neglected. For high precision positioning those errors are eliminated or accounted for to the degree that the effect is again negligible for those applications.

Multipath however is an error that is not easy to eliminate. When the GNSS signal is received directly and indirectly it is called multipath. Multipath can for example occur due to reflection of the signal on near by structures. In some cases multipath can be detected due to an overdetermined mathematical model. The best one can do is identify the signal which has multipath and account for that observation when calculation the position. A good antenna helps to prevent multipath. Another method to prevent multipath to use a location with a clear sky and no structures which could cause reflection of the radio-signal towards the receiver antenna.

Systematic errors are for example wrong orbit parameters or satellite clock drift. A user cannot change the system itself and is not able to send orbital correction to satellites, it is the job of the control segment to ensure that the satellites are in orbit, or that they correct their orbit, so that GNSS is always available. One option is to use precise orbits. When precise orbits are used, the orbits are determined based on reference stations on Earth. The orbit parameters from the navigation message are then neglected.

2.1. GNSS measurement setup

There are different possible GNSS modes. As said it is possible to use GNSS for absolute positioning, for example Single-point positioning (SPP) , or relative positioning, for example Real Time Kinematic (RTK). It is also possible to use the GNSS observations for timing, which makes it possible to use the high precision clock of the satellites.

2.1.1. Single-point positioning

Single-point positioning (SPP) is the most common used GNSS mode. SPP is an absolute positioning mode, based on the PRN code measurement. The PRN code can be measured with a precision of 3 meter. The average user range error (URE) is ≤ 7.8m [12]. SPP can be used in real time. GNSS is based on distance measurement between one receiver and multiple satellites.

The distance measurements are done using the travel time of the signal from the satellite to the receiver. The satellite transmits the PRN signal to the receiver. The receiver creates a duplicate of the PRN signal, and then correlates it with the received signal. By changing the time of the replicated PRN code it is possible to determine the travel time. Each measurement of travel time describes a sphere of possible locations for the receiver centered on the known satellite position. When the travel time is scaled with the speed of light the distance between the satellite and the receiver is known in meters.

Combining multiple PRN code observations makes it is possible to find one location where multiple spheres intersect. This is true when all clocks, from the satellites and the receiver, are synchronous. Since this is not the case, not only the user receiver position is unknown but also the timing is. This means that there are four unknown parameters and thus at least four satellites are needed to compute a position. For SPP the satellite clock biases are calculated based on the navigation message. In practice there may be a small deviations between the calculated and the true satellite position and clock drift, but this is neglected since other error sources, for example the calculated (based on models) for atmospheric signal delays and the measurement precision, will have a larger impact on the final estimate.

The satellite positions are treated as known and the positions are calculated with the ephemeris from the navigation data. The orbits of the satellites are described in the navigation message. Based on the time of transmission of the signal it is possible to calculate the satellite position with a precision of a few meters. For SPP the calculated satellite positions do have a high enough precision to be treated as known positions.

SPP is used for example for smartphones, car navigation and so on. It is the least precise mode of GNSS and mostly used for applications which do not require high positioning. If higher positioning accuracy is required one can use a differential setup.

2.1.2. Differential-GNSS code measurement

Differential Global Navigation Satellite System (DGNSS) or Differential Global Positioning System (DGPS), is based on multiple observation positions. DGNSS uses a fixed base-station on a known position. This base-station observes the same satellites as the receiver, but the location of the base-station is known with high precision, for example a result of a long time series of SPP or an earlier survey. A schematic of DGNSS is shown in Figure 2.2.

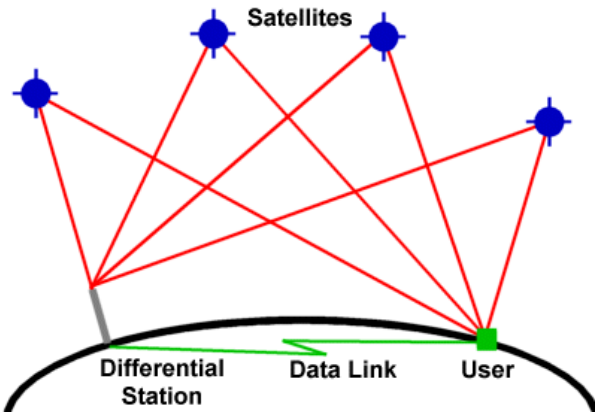


Figure 2.2: DGNSS overview [5]

The atmospheric errors are different for different positions, therefore the closer the receiver is to the base-station the more accurate the differential corrections are.

The principle of differential GNSS is that the systematic biases in the GNSS system are the same for both the reference station and the user receiver. The position of the reference station is known,

therefore it is possible to calculate the differential corrections based on the position or pseudoranges.

The differential corrections can be expressed in terms of pseudoranges or in terms of position. When the corrections are based on the position, the user can correct its final $[x, y, z]^T$ position with the calculated correction (position shift) of the reference station. The other option is to calculate a differential correction of range per pseudorange. To do this the calculated geometric distance between satellite and receiver is subtracted from the observed pseudorange at the reference station. The results are then by approximation the systematic biases per satellite, like the atmospheric effects and the satellite clock bias. When pseudorange corrections are used, the pseudorange of the user is corrected before the user position is calculated.

Pseudorange corrections are preferred over position corrections. The pseudorange corrections are per satellite and the user can chose to use different models, for example to use only certain satellites. The position correction is based on a model, for example including all satellites, therefore the user should use the same model to generate the same systematic biases. Systematic biases can also result from the used receiver. Position corrections are also influenced by the used hardware and to eliminate those errors the user should use the same hardware configuration. Sometimes not exactly the same satellites are visible on both the reference and user position, or at one of the locations there is multipath or another bias. This is the reason that more often pseudorange corrections are used instead of position corrections.

The differential corrections are sent to the user using dedicated radio signals or internet. The data that contains the differential corrections is called the augmentation data.

It is not necessary for each user to have its own base station, there are also regional Satellite-based Augmentation Systems (SBAS). Those systems will send the augmentation data to the user. Examples of different SBAS systems are WAAS (U.S.), EGNOS (Europe), MSAS (Japan), GAGAN (India) and SDMC (Russia) [14]

2.1.3. Differential-GNSS phase measurement

Differential-GNSS based on the fractional measurement of the phase of the carrier wave is often called Real Time Kinematic (RTK). The carrier phase observation is much more precise compared to the PRN code observation. The difficulty with the phase measurement is that there is an ambiguity in the amount of cycles that precedes the current cycle. The wave length of the L1 band of GPS is $^{(3e8 m s^{-1})}/_{(1575.42 MHz)} = 19$cm. This means that it is impossible to tell how many cycles there are between the receiver and the satellite. The measured fractional phase is a real number ($\mathbb{R}$), i.e. with decimals, but it is known that the measured phase is preceded by N cycles, where N is an integer ($\mathbb{Z}$).

RTK is based on relative positioning, with short baselines. Therefore two receivers are used, one static which is the reference receiver or base station and one rover. The rover can be kinematic but it is also possible to use a static rover.

The first observation of the fractional phase typically has a value between 0 and 1, i.e. $\mathbb{R} \in [0, 1]$. The carrier phase measurement is a continuous measurement. After the first observation, at t_0, the receiver is able to observe the zero-crossings of the carrier phase. This means that the carrier phase observation can be larger than 1. The extra unknown is the ambiguity from the first observation (t_0), after this observation the zero-crossings are accounted for.

To find the integer value of N a least squares adjustment is used. First the ambiguity is a floating number. Based on a first, ordinary least squares adjustment the ambiguity is estimated. Based on the estimate of the unknown ambiguities an integer is estimated. This new integer value for the ambiguities is used in a next least squares estimation. It can be hard to find the integer solution for N, one can use the Least-squares AMBiguity Decorrelation Adjustment (LAMBDA).

The float solution for RTK is the relative solution where the ambiguity is a real number instead of an integer. This is always the first step when using RTK. It is possible to get from the real valued ambiguity solution to an integer solution, using the LAMBDA method, which is then the fixed solution.

The fixed solution for RTK is the differential GNSS solution with an integer ambiguity. Each satellite signal has its own ambiguity. The first step is to use a float solution for a first estimate of the ambiguities. The second step is to find a solution with a constraint on the ambiguities, namely that it is known that the ambiguities have to be integer. This is the solution where the ambiguities are real numbered. In short the estimated float solution for the ambiguities are compared with integer grid points in a (hyper)ellipsoidal. The grid point nearest to the float solution is assumed to be the final integer solution. The ambiguities are constant over time, as long as the signal is un-interrupted. When there is a loss off

lock, the ambiguity is different and therefore has to be estimated again. In the past the ambiguity fixing was a time consuming process, but with LAMBDA most of the time the ambiguities are fixed within minutes.

2.1.4. Timing

GNSS can also be used for timing applications. The receiver itself has a normal clock, with an unknown clock bias. Based on the observations also the receiver clock bias is estimated. This makes it possible to have relatively high precision timing at the receiver, while the receiver clock itself is a quartz oscillator. The result is that the time bias is < 40ns. Another option is to use a relative setup. It is important to note that when a relative setup is used, the estimated clock bias is also relative to the reference station. This can be used when two separate stations have to be synchronized with each other.

2.2. Observation equations

There are two observation equations for GNSS applications. As said both measurements have their (dis)advantages. First the code observation is discussed, thereafter the phase observation.

For both methods the basis of the observable is the same, namely a time difference between the time of reception and the time of transmission of the signal.

$$\tau_r^s(t) = t_r(t) - t^s(t - \tau_r^s(t)) \tag{2.1}$$

Where $\tau_r^s(t)$ is the travel time between the receiver and the satellite, $t_r(t)$ is the time of reception based on the receiver time and $t^s(t)$ is the time of transmission in satellite time.

The travel time ($\tau_r^s(t)$) is time dependent since both the satellite and the rover are moving with respect to a real terrestrial reference frame. If there are no other error sources, like for example the clock bias in the receiver, the time $\tau_r^s(t) \cdot c$ is the geometric distance between the satellite and the receiver. It is import to note that the time of reception is at time t, while the time of transmission is at another unknown time, $t - \tau$ (τ seconds earlier as the signal needs τ seconds to travel from the satellite to the receiver). For further usage of the variable t, t represents GPS-time.

Astronomical time (or Universal Time (UT1)) is determined by the Earth rotation with reference to the stars. UT1 shows time deviation due to small variations in the Earth rotation, since time is closely linked with the longitude. Universal Time Coordinated (UTC) is used as stable time reference in daily live, which is linked with International Atomic Time (TAI). TAI is a weighted average of 400 atomic clocks. UTC follows UT1 with steps, so that the absolute difference between UT1 and UTC is always smaller than 0.9s ($|\text{UT1} - \text{UTC}| \leq 0.9$s). The drift of UT1 is circa 0.5~0.7s yr^{-1}. UTC is corrected with a leap second when the difference may exceed this limit. The link with TAI makes sure that UTC is stable, therefore between two corrections, the leap second, the difference between UTC and TAI is constant. GPS time is introduced 6th January 1980. The difference between TAI and GPS-time (GPST) is constant: $\text{TAI} - \text{GPST} = 19$s. GPST is also a constant time system, like TAI. This means that the difference between UTC and GPST will increase over time due to the drift in UT1.

GPS-time is included in the navigation message based on Week Number (WN) and Time Of Week (TOW). GPST is calculated using the WN, the number of weeks since 6th January 1980, and the TOW, time since start of the week (previous Sunday 0:00h).

$$604800 \cdot \text{WN} + \text{TOW} = \text{GPST}$$

For the observation equations the position of the satellites are used. To calculate the position of the satellite at $t - \tau$, therefore the travel time should be used, see Equation (2.1). Since this is unknown, the travel time is approximated for the first iteration.

Another effect that has influence on the geometric range is the rotation of the Earth during the travel-time of the signal from satellite to receiver on Earth, sometimes called the Sagnac effect. One way to correct for this effect is to rotate the satellites positions with the rotation of the Earth. For example both the satellites and the (Earth Centered Earth Fixed) ECEF system can be rotated back to the start of the week (TOW$= 0$). Then the geometric range can be calculated in this pseudo inertial frame. After the geometric range is calculated the satellite positions and the approximated receiver position can be returned back to the original reference ECEF frame at the time of the observation.

2.2.1. Code measurement

The code measurement is based on the PRN code. The observation equation is stated in below in Equation (2.2).

$$P_r^s(t) = \rho_r^s(t) + c\delta t_r(t) - c\delta t^s\,(t - \tau_r^s(t)) + I_r^s + T_r^s \tag{2.2}$$

Where $P_r^s(t)$ is the pseudo range, $\rho_r^s(t)$ is the geometric distance, c is the speed of light, $\delta t_r(t)$ is the receiver clock bias, $\delta t^s(t)$ is the satellite clock bias, I_r^s is the ionospheric influence and T_r^s is the bias due to the troposphere. The notation (t) indicates that the variable changes over time. The subscript r indicates the receiver and the superscript s indicates the satellite.

The pseudorange measurement ($P_r^s(t)$) is measured, while the geometric range ($\rho_r^s(t)$) is what is needed to determine the final position. If all clocks are in sync and there is no atmosphere it is still not possible to perfectly measure the geometric range since the instrument also has instrumental noise. The instrumental noise of GPS on the PRN code is in the order of 1% of the chip length, thus in the order of 3 meter. Further the satellite positions are treated as known locations, while in reality the positions are known with an accuracy in the order of meters. It is possible to use precise orbits, but in general this is not needed since the other error sources have larger impact on the final accuracy.

What can be seen is that for each satellite a new variable is introduced, namely the satellite clock bias. For SPP the satellite clock bias is not estimated, but retrieved from the navigation message. The atmospheric influence can be based on a model and when enough satellites are available some parameters for the model can be estimated. It is also possible to neglect the atmospheric effects and only estimate the position and the clock bias of the receiver, this is done when SPP is used.

When the atmosphere is included in the model it is often separated between the ionosphere and the troposphere, as can be seen in Equation (2.2).

2.2.2. Phase measurement

Another observable is the carrier wave of the signal instead of the PRN code. This measurement is based on the carrier wave of the signal since the observable is the phase of the carrier wave. The phase represents a measure of distance though its wavelength.

$$\Phi_r^s = \Phi_r(t) - \Phi^s(t - \tau_r^s(t)) + N_r^s \tag{2.3}$$

Where Φ_r^s is the difference in phase between the transmitted signal and the received signal, Φ_r is the received phase, Φ^s is the transmitted phase and N_r^s is the integer ambiguity. The terms $\Phi^s(t-\tau_r^s(t))$ and $\Phi_r(t)$ are defined as:

$$\Phi^s(t - \tau_r^s(t)) = \Phi^s(t_0) + f(t - \tau_r^s(t) - t_0) + f(\delta t^s(t - \tau_r^s(t)) - \delta t^s(t_0)) \tag{2.4}$$

$$\Phi_r(t) = \Phi_r(t_0) + f(t - t_0) + f(\delta t_r(t) - \delta t_r(t_0)) \tag{2.5}$$

The time t_0 is the start time of the observation time series. The ambiguity is solved with respect to this time, therefore the final phase measurement increases over time and does not stay between $[0 - 1]$. Due to the continuous behaviour of the measurement it is possible to count how many cycles have passed since t_0. The transmitted phase is calculated in the term $f(t-\tau_r^s(t)-t_0)$ and the received phase with $f(t - t_0)$, f is the frequency of the signal. The last part of the equations is the introduced error due to clock errors of the satellite and the receiver.

Substituting Equation (2.4) and Equation (2.5) in Equation (2.3) results in:

$$\Phi_r^s(t) = f\tau_r^s(t) + f\delta t_r(t) - f\delta t^s(t - \tau_r^s(t)) + A_r^s \tag{2.6}$$

The phase observation is in cycles. Equation (2.6) is divided in two parts. The first part is dependent of time and will change each new observation. The last part A_r^s is constant for un-interrupted tracking of the satellite since t_0. The constant part A_r^s is defined as:

$$A_r^s = N_r^s + \Phi_r(t_0) - f\delta t_r(t_0) - \Phi^s(t_0) + f\delta t^s(t_0) \tag{2.7}$$

Note that A_r^s is not per definition integer, the integer part is N_r^s. The phase observation is then multiplied by the wave length λ, so that the observation is also expressed in meters. Then the ionosphere and troposphere are added to complete the observation equation:

$$p_r^s(t) = \rho_r^s(t) + c\delta t_r(t) - c\delta t^s(t - \tau_r^s(t)) + \lambda A_r^s - I_r^s + T_r^s \tag{2.8}$$

λ is the wavelength for a specific frequency (f). When multiple frequencies are measured it is noted with subscript i, e.g. λ_i and f_i.

What can be seen is that the final equations for the observables of the PRN-code and the carrier wave are almost similar. Note the difference in sign for the ionospheric component. The ionosphere delays the signal of the PRN code, while the opposite happens for the carrier wave.

2.3. Mathematical models

Based on the observation equations different models can be used based on different applications of GNSS.

2.3.1. Single point

Single-point positioning (SPP) is based on pseudo ranges. The observations are based on Equation (2.2). First consider a functional model where the atmospheric influences are neglected. The result will be a function of the geometric range $\rho_r^s(t)$ and the clock bias of the receiver. The clock biases of the satellites are calculated based on the clock parameters from the navigation data.

The used functional model is then:

$$P_r^s(t) + c\delta t^s(t - \tau_r^s(t)) = \rho_r^s(t) + c\delta t_r(t) \tag{2.9}$$

Based on the first-order Taylor polynomial the model is linearised so that the Least Squares solution can be used.

$$f(x, y, z) = f(x_0 + \Delta x, y_0 + \Delta y, z_0 + \Delta z) \tag{2.10}$$

$$= f(x_0, y_0, z_0) + \frac{\partial f(x_0, y_0, z_0)}{\partial x_0}\Delta x + \frac{\partial f(x_0, y_0, z_0)}{\partial y_0}\Delta y + \frac{\partial f(x_0, y_0, z_0)}{\partial z_0}\Delta z \tag{2.11}$$

$$\rho_r^s(t) = \rho_{r,0}^s(t) - \frac{x^s(t - \tau_r^s(t)) - x_{r,0}(t)}{\rho_{r,0}^s(t)}\Delta x_r(t) - \frac{y^s(t - \tau_r^s(t)) - y_{r,0}(t)}{\rho_{r,0}^s(t)}\Delta y_r(t) - \frac{z^s(t - \tau_r^s(t)) - z_{r,0}(t)}{\rho_{r,0}^s(t)}\Delta z_r(t) \tag{2.12}$$

Where $\rho_{r,0}^s$ is:

$$\rho_{r,0}^s(t) = \left\{\left(x^s(t - \tau_r^s(t)) - x_{r,0}(t)\right)^2 + \left(y^s(t - \tau_r^s(t)) - y_{r,0}(t)\right)^2 + \left(z^s(t - \tau_r^s(t)) - z_{r,0}(t)\right)^2\right\}^{\frac{1}{2}} \tag{2.13}$$

To clarify the notation the variable t will be removed from further notations, but it should be noted that all parameters are dependent on time. Substituting Equation (2.12) in Equation (2.9) gives:

$$P_r^s + c\delta t^s - \rho_{r,0}^s - c\delta t_{r,0} = -\frac{x^s - x_{r,0}}{\rho_{r,0}^s}\Delta x_r - \frac{y^s - y_{r,0}}{\rho_{r,0}^s}\Delta y_r - \frac{z^s - z_{r,0}}{\rho_{r,0}^s}\Delta z_r + c\Delta\delta t_r \tag{2.14}$$

This results in a system of equations which can be written in the form of ($\mathrm{E}(\Delta\underline{\mathbf{y}}) = A\Delta\mathbf{x}$). Where $P_r^s + c\delta t^s - \rho_{r,0}^s - c\delta t_{r,0} = \Delta y^s$.

$$\mathrm{E}(\underbrace{\begin{bmatrix}\Delta y^1\\ \Delta y^2\\ \vdots\\ \Delta y^s\end{bmatrix}}_{\Delta\underline{\mathbf{y}}}) = \underbrace{\begin{bmatrix}-\frac{x^1-x_{r,0}}{\rho_{r,0}^1} & -\frac{y^1-y_{r,0}}{\rho_{r,0}^1} & -\frac{z^1-z_{r,0}}{\rho_{r,0}^1} & 1\\ -\frac{x^2-x_{r,0}}{\rho_{r,0}^2} & -\frac{y^2-y_{r,0}}{\rho_{r,0}^2} & -\frac{z^2-z_{r,0}}{\rho_{r,0}^2} & 1\\ \vdots & \vdots & \vdots & \vdots\\ -\frac{x^s-x_{r,0}}{\rho_{r,0}^s} & -\frac{y^s-y_{r,0}}{\rho_{r,0}^s} & -\frac{z^s-z_{r,0}}{\rho_{r,0}^s} & 1\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}\Delta x_r\\ \Delta y_r\\ \Delta z_r\\ c\Delta\delta t_r\end{bmatrix}}_{\Delta\mathbf{x}} \tag{2.15}$$

Finally it is possible to calculate the position when $m \geq 4$. When $m = 4$ there are enough observations to solve for the four parameters of interest. Each observation extra will add extra information to the model, which makes it possible to also check the observations for possible outliers. The Best Linear Unbiased Estimator (BLUE) is shown below in Equation (2.16).

$$\Delta\underline{\hat{\mathbf{x}}} = \left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \Delta\underline{\mathbf{y}} \tag{2.16}$$

Where Q_{yy} is the variance matrix of the observables. Often a diagonal matrix is used with a-priori defined variance.

$$Q_{yy} = \sigma^2 I_{m\times m} \tag{2.17}$$

The Dilution Of Precision is determined based on the design matrix of the model, based on $Q_x = (A^T A)^{-1}$. The DOP value is thus only a measure of the geometric strength and tells, apart from the scale factor, nothing about the precision. The variance matrix of the solutions is defined as:

$$Q_{\hat{x}\hat{x}} = \left(A^T Q_{yy}^{-1} A\right)^{-1} \tag{2.18}$$

If the a-priori variance matrix of the observables Q_{yy} is defined as described in Equation (2.17) then $Q_{\hat{x}\hat{x}} = \sigma^2 \left(A^T A\right)^{-1}$.

The receiver and satellite position in vector notation are: $\mathbf{x}_r = [x_r, y_r, z_r]^T$ and $\mathbf{x}^s = [x^s, y^s, z^s]^T$. The geometric range can then be given as: $\rho_r^s = |\mathbf{x}^s - \mathbf{x}_r|$. Each row of the design matrix can then also be written as:

$$\left[-\frac{x^s - x_{r,0}}{|\mathbf{x}^s - \mathbf{x}_{r,0}|} \quad -\frac{y^s - y_{r,0}}{|\mathbf{x}^s - \mathbf{x}_{r,0}|} \quad -\frac{z^s - z_{r,0}}{|\mathbf{x}^s - \mathbf{x}_{r,0}|} \quad 1\right] = \left[-\frac{\mathbf{x}^s - \mathbf{x}_{r,0}}{|\mathbf{x}^s - \mathbf{x}_{r,0}|} \quad 1\right] = \left[-\mathbf{u}_r^{s\,T} \quad 1\right] \tag{2.19}$$

Where $\mathbf{u}_r^{s\,T}$ is the unit direction vector from the receiver towards the satellite.

To find the final position an iteration is used. The receiver clock bias is also used to calculate the transmission time $t^s(t - \tau_r^s) = t_r - \delta t_r - \tau_r^s$, Equation (2.1). The receiver clock bias is included in the estimation process, and therefore after a solution is estimated, the new clock bias should be used. Since the functional model is a non-linear model, an iteration loop is used to try to converge the final estimate until it is smaller than a certain threshold value $||\Delta\mathbf{x}||_{Q_{\hat{x}\hat{x}}^{-1}} < \delta$. Due to the large distances between receiver and the satellites, in practice this will converge to a minimum and thus the $\Delta\mathbf{x}$ will go to zero.

2.3.2. Differential GNSS

Differential-GNSS (DGNSS) can be used to improve the accuracy of the estimation. DGNSS is based on a setup with multiple receivers, which simultaneously take pseudorange observations. In this case we limit us to a two receiver set-up.

The reference station is placed on a known position. The other receiver, the rover, has an unknown position. The reference station is denoted as $(\cdot)_1$ and the rover is denoted as $(\cdot)_2$.

The effect of many biases, which had a negative effect on the SPP solution, will be small or even negligible when DGNSS is used on sufficient short distances.

The satellite clock bias is eliminated, even if the travel time to the same satellite differs for the reference receiver and the rover ($\tau_1 \neq \tau_2$). The drift of the satellite clock is $9\text{ns/d} = 0.1\text{ps/s}$, which is in meters $3\text{e}8 \cdot 1\text{e}{-13} = 3\text{e}{-5}\text{m/s}$. The maximum difference in travel time is 19ms, for when the elevation for one receiver is 0° and for the other receiver 90° [34, chapter 21.4]. This means that the difference in satellite clock bias is maximum $3\text{e}{-5} \cdot 19\text{e}{-3} = 5.7\text{e}{-7}\text{meter}$ and this is a much smaller bias compared to the precision of the phase and code observations.

Differential-GNSS code measurement

DGNSS using code measurements is based on the observation Equation (2.2).

$$\mathrm{E}\{P_1^s(t)\} - \rho_1^s(t) = c\delta t_1(t) - c\delta t^s\left(t - \tau_1^s(t)\right) + I_1^s + T_1^s \tag{2.20}$$

$$\mathrm{E}\{P_2^s(t)\} = \rho_2^s(t) + c\delta t_2(t) - c\delta t^s\left(t - \tau_2^s(t)\right) + I_2^s + T_2^s \tag{2.21}$$

Subtracting Equation (2.20) from Equation (2.21) yields:

$$\mathrm{E}\{\bar{P}_2^s(t)\} = \mathrm{E}\{P_2^s(t)\} - \mathrm{E}\{P_1^s(t)\} + \rho_1^s = \rho_2^s(t) + c\delta t_{1,2} + I_{1,2}^s + T_{1,2}^s \tag{2.22}$$

In the first equation the geometric distance ρ_1^s is known since the reference station is placed on a known position. The bar $(\bar{\cdot})$ is used to show that it is a corrected pseudorange.

The atmospheric effects are distance dependent. The closer the reference station and the rover are together, the smaller the resulting atmospheric bias. Based on the distance between the reference station and the rover it can be chosen to use a (simplified) model to model the atmospheric effects or to neglect the differential atmospheric effects.

Due to the subtraction of Equation (2.21) and Equation (2.20) the newly introduced parameter, the relative clock bias, replaces the receiver clock bias. The relative clock bias is defined as $\delta t_{1,2} = \delta t_2 - \delta t_1$.

Another effect is that the observation noise is increased, since two observations are added together. If a simplified variance matrix is used, $Q_{yy} = \sigma^2 I_{m\times m}$, then the new variance of the corrected pseudoranges is defined as:

$$Q_{\bar{y}\bar{y}} = 2\sigma^2 I_{m\times m} \tag{2.23}$$

The corrected pseudoranges can be used to calculate the position of the rover. The model looks like the model that is used for SPP, Equation (2.15). The difference is that the pseudoranges are corrected and that the variance matrix is changed. The system with linearised observation equations is as follow:

$$\mathrm{E}(\begin{bmatrix}\Delta\bar{\mathbf{y}}\\ \underline{\ }_2\end{bmatrix}) = \begin{bmatrix}-(\mathbf{u}_2^1)^T & 1\\ \vdots & \vdots\\ -(\mathbf{u}_2^s)^T & 1\end{bmatrix}\begin{bmatrix}\Delta\mathbf{x}_2\\ c\Delta\delta t_{1,2}\end{bmatrix} \tag{2.24}$$

Where $\Delta\bar{\mathbf{y}}_{\underline{\ }2}$ is the observed minus computed corrected code observation:

$$\Delta\bar{\mathbf{y}}_2 = [\bar{P}_2^1 - \bar{P}_{2,0}^1, \dots, \bar{P}_2^s - \bar{P}_{2,0}^s]^T \tag{2.25}$$

Where $\bar{P}_{2,0}^s$ is the computed observation calculated with the initial guesses for the unknown parameters of Equation (2.22).

$$\bar{P}_{2,0}^s = P_2^s - P_1^s + \rho_1^s - \rho_{2,0}^s - c\delta t_{1,2,0} \tag{2.26}$$

Differential-GNSS mathematical model derivation

Next the derivation of the single differenced (SD) model is shown. The SD model results in a baseline. This means that the position is estimated relative to the reference station. This model can be used both for code and phase observations.

The linearised differenced model for two receivers is shown in Equation (2.27).

$$\mathrm{E}(\begin{bmatrix}\Delta\mathbf{y}_{\underline{\ }1}\\ \Delta\mathbf{y}_{\underline{\ }2}\end{bmatrix}) = \begin{bmatrix}-(\mathbf{u}_1^1)^T & & 1 & & -1 & & \\ \vdots & & \vdots & & & \ddots & \\ -(\mathbf{u}_1^s)^T & & 1 & & & & -1\\ & -(\mathbf{u}_2^1)^T & & 1 & -1 & & \\ & \vdots & & \vdots & & \ddots & \\ & -(\mathbf{u}_2^s)^T & & 1 & & & -1\end{bmatrix}\begin{bmatrix}\Delta\mathbf{x}_1\\ \Delta\mathbf{x}_2\\ c\Delta\delta t_1\\ c\Delta\delta t_2\\ c\Delta\delta t^1\\ \vdots\\ c\Delta\delta t^s\end{bmatrix} \tag{2.27}$$

Where $\left[\Delta\mathbf{y}_{\underline{\ }1}, \Delta\mathbf{y}_{\underline{\ }2}\right]^T$ is the observed minus computed vector of the observations of receiver 1 and 2.

In the linearised model the satellite coordinates are assumed to be known. The satellite clock biases are also included in the model. This is different compared to the SPP solution where the satellite clock biases are calculated based on the sent navigation message. The differential atmospheric delays are assumed to be absent and the atmospheric delays are captured as signal delay in the satellite clock bias $c\delta t^s$. The differential atmospheric biases are small compared to the absolute value, but increase when the distance between the reference station and the rover increase.

The following square $2m \times 2m$ full rank transformation matrix is applied to the observables [38].

$$T = \begin{bmatrix}I_m & \\ -I_m & I_m\end{bmatrix}$$

The transformation matrix is applied in the form $\mathbf{y} = A\mathbf{x} \rightarrow T\mathbf{y} = TA\mathbf{x}$, the resulting model of linearised observation equations is shown below.

$$\mathrm{E}\left(\begin{bmatrix} \Delta\underline{y}^1_{-1} \\ \vdots \\ \Delta\underline{y}^s_{-1} \\ \Delta\underline{y}^1_{-1,2} \\ \vdots \\ \Delta\underline{y}^s_{-1,2} \end{bmatrix}\right) = \begin{bmatrix} -(\mathbf{u}_1^1)^T & & 1 & & -1 & & \\ \vdots & & \vdots & & & \ddots & \\ -(\mathbf{u}_1^s)^T & & 1 & & & & -1 \\ (\mathbf{u}_1^1)^T & -(\mathbf{u}_2^1)^T & -1 & 1 & & & \\ & \vdots & \vdots & & & & \\ (\mathbf{u}_1^s)^T & -(\mathbf{u}_2^s)^T & -1 & 1 & & & \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \\ c\Delta\delta t_1 \\ c\Delta\delta t_2 \\ c\Delta\delta t^1 \\ \vdots \\ c\Delta\delta t^s \end{bmatrix} \tag{2.28}$$

The resulting observation vector after this transformation shows observation differences between reference receiver and the rover $\Delta\underline{\mathbf{y}}_{-1,2} = \Delta\underline{\mathbf{y}}_{-2} - \Delta\underline{\mathbf{y}}_{-1}$ for the lower m observations. Those lower m observations are the single differences. The upper m observations are the only m observations related to the satellite clock biases and therefore they can be considered to determine only the satellite clock biases. The upper m observations can be left out since the satellite clock biases are not of interest for the position.

The model can be reparameterized to split the coordinates in coordinates differences [38].

$$\underline{\mathbf{y}} = A\mathbf{x} = ADD^{-1}\mathbf{x}$$

The new parameters ($D^{-1}\mathbf{x}$) are shown on the right in Equation (2.29).

$$\underbrace{\begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \\ \Delta\delta t_1 \\ \Delta\delta t_2 \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} I_3 & & & \\ I_3 & I_3 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_{1,2} \\ \Delta\delta t_1 \\ \Delta\delta t_2 \end{bmatrix}}_{D^{-1}\mathbf{x}} \tag{2.29}$$

The coordinate differences are defined in $\Delta\mathbf{x}_{1,2} = \Delta\mathbf{x}_2 - \Delta\mathbf{x}_1$

$$\mathrm{E}\left(\begin{bmatrix} \Delta\underline{y}^1_{-1,2} \\ \vdots \\ \Delta\underline{y}^s_{-1,2} \end{bmatrix}\right) = \underbrace{\begin{bmatrix} (\mathbf{u}_1^1 - \mathbf{u}_2^1)^T & -(\mathbf{u}_2^1)^T & -1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ (\mathbf{u}_1^s - \mathbf{u}_2^s)^T & -(\mathbf{u}_2^s)^T & -1 & 1 \end{bmatrix}}_{AD} \underbrace{\begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_{1,2} \\ \Delta\delta t_1 \\ \Delta\delta t_2 \end{bmatrix}}_{D^{-1}\Delta\mathbf{x}} \tag{2.30}$$

In this model the position coordinates of the reference station are also estimated in $\Delta\mathbf{x}_1$. The coefficients in the first three columns, which are related to the reference position, are much smaller than the coefficients for the coordinate differences. A near rank deficiency occurs, to deal with this in GNSS models the coordinates of $\mathbf{x}_1$ are kept fixed, thus $\Delta\mathbf{x}_1 = 0$.

The estimation of the clock biases $\Delta\delta t_1$ and $\Delta\delta t_2$ are linearly dependent (the last two columns in the matrix) and thus it is not possible to solve both unknowns at the same time. Therefore the clock bias of the reference station δt_1 is kept fixed, thus $\Delta\delta t_1 = 0$.

The final linearised model is thus:

$$\mathrm{E}\left(\begin{bmatrix} \Delta\underline{y}^1_{-1,2} \\ \vdots \\ \Delta\underline{y}^s_{-1,2} \end{bmatrix}\right) = \begin{bmatrix} -(\mathbf{u}_2^1)^T & 1 \\ \vdots & \vdots \\ -(\mathbf{u}_2^s)^T & 1 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_{1,2} \\ \Delta\delta t_{1,2} \end{bmatrix} \tag{2.31}$$

Note that the design matrix A for the rover is the same as the design matrix A of the SPP model (Equation (2.15) and in vector form Equation (2.19)). The difference with the model described above, with the pseudorange corrections (see Equation (2.24)), is that this model estimates a position relative to the reference station. The previous model estimates an absolute position for the user receiver based on the corrected pseudorange observations.

Differential-GNSS phase measurement

Based on the derivation for the differential model it is possible to create a same model for the carrier phase observations. The DGNSS using carrier-phase observations, this is based on observation equation Equation (2.8).

$$\mathrm{E}\{p_1^s(t)\} - \rho_1^s(t) = c\delta t_1(t) - c\delta t^2(t - \tau_1^s(t)) + \lambda A_1^s - I_1^s + T_1^2 \tag{2.32}$$

$$\mathrm{E}\{p_2^s(t)\} = \rho_2^s(t) + c\delta t_2(t) - c\delta t^2(t - \tau_2^s(t)) + \lambda A_2^s - I_2^s + T_2^2 \tag{2.33}$$

Subtracting Equation (2.32) from Equation (2.33) yields:

$$\mathrm{E}\{\bar{p}_2^s(t)\} = \mathrm{E}\{p_2^s(t)\} - \mathrm{E}\{p_1^2(t)\} + \rho_1^s = \rho_2^s(t) + c\delta t_{1,2}(t) + \lambda A_{1,2}^s - I_{1,2}^s + T_{1,2}^2 \tag{2.34}$$

Where the single difference ambiguity $A_{1,2}^s$ is based on Equation (2.7)

$$A_{1,2}^s = A_2^s - A_1^s = N_{1,2}^s + \Phi_{1,2}(t_0) - f\delta t_{1,2}(t_0) \tag{2.35}$$

The bar $(\bar{\cdot})$ notation is used to show that it is a single differenced observation.

The ambiguities are unknown, but they remain the same over time as long as tracking of the satellite is not disturbed.

The clock bias $\delta_{1,2}$ and the ambiguity are relative to the reference station for the single differenced model. For this differenced model the variance matrix is changed again (see Equation (2.23)). Under normal conditions for small base-lines the differential ionosphere and troposphere delays can be neglected for DGNSS.

The final linearised model is shown in Equation (2.36), based on the derivation from subsection 2.3.2

$$\mathrm{E}(\left[\underline{\Delta\bar{\mathbf{y}}}_2\right]) = \begin{bmatrix} -\mathbf{u}_2^1 & 1 & \lambda & & \\ \vdots & \vdots & & \ddots & \\ -\mathbf{u}_2^s & 1 & & & \lambda \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_{1,2} \\ c\Delta\delta t_{1,2} \\ A_{1,2}^1 \\ \vdots \\ A_{1,2}^s \end{bmatrix} \tag{2.36}$$

Where $\underline{\Delta\bar{\mathbf{y}}}_2$ is the observed minus computed corrected phase observation.

$$\Delta\bar{\mathbf{y}}_2 = \left[\bar{p}_2^1 - \bar{p}_{2,0}^1, \ldots, \bar{p}_2^s - \bar{p}_{2,0}^s\right] \tag{2.37}$$

The ambiguities in Equation (2.36) shows that there are m, the number of the used satellites, extra unknowns. In practice a double differenced model is used so that the ambiguities are integer and the relative clock bias is eliminated. This is a linear transformation and the final result of a single differenced model or a double differenced model are equal for $\hat{\mathbf{x}}$.

2.3.3. Timing

GNSS can also be used for absolute timing applications. For precise timing applications a receiver is typically located on a known position. The parameter of interest is solely the clock bias of the receiver; δt_r.

$$\mathrm{E}(\begin{bmatrix} \underline{\Delta y}_r^1 \\ \vdots \\ \underline{\Delta y}_r^s \end{bmatrix}) = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \left[c\delta t_r\right] \tag{2.38}$$

Where $\Delta y_r^s = y_r^s - \rho_r^s + c\delta t^s$, and the atmospheric effects are modelled or neglected.

While the model is already a linear model, iteration is still necessary due to calculate the satellite positions on the correct time $t = t_r - \delta t_r$.

One can also synchronize two separate stations, based on relative positioning. In this case it is assumed that both the reference station and now also the rover station are at known positions. The estimated clock bias in that case s the relative clock bias $\delta t_{1,2}$, see the model below.

$$\mathrm{E}(\begin{bmatrix} \underline{\Delta y}_{1,2}^1 \\ \vdots \\ \underline{\Delta y}_{1,2}^s \end{bmatrix}) = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \left[c\delta t_{1,2}\right] \tag{2.39}$$

3

Data formats and communication protocols used for GNSS

In GNSS applications data are stored in specific formats. For real time differential GNSS data from another point, the reference station, are needed and are sent using specific protocols. Before one is able to manipulate the augmentation data one should understand how the data are stored and what kind of data are sent between a server and the client.

In this chapter an overview is presented with the most common data formats and communication protocols which are used for GNSS. If the used communication protocol is known it is possible to perform a man-in-the-middle (MITM) attack, this is described after the communication protocols. At the end of the chapter there is a part about some countermeasures against a MITM attack.

3.1. Data formats

A data format contains specific rules about how the data are formatted and for example how it is stored. Augmentation data are sent to the rover using communication protocols in a standard data format. In general all receivers can use the standard protocols and formats. The standard data format for augmentation data are the Radio Technical Commission for Maritime Services (RTCM) SC-104 data protocol. Another often used data format is the Receiver INdependent EXchange (RINEX) format. In practice the difference between RTCM and RINEX is that RTCM is used for real time positioning and RINEX is typically used for post processing. A manufacturer can also create its own format, this is also done for example by Trimble which uses the Compact Measurement Record (CMR) CMR+ and the newest CMRx formats.

3.1.1. Radio Technical Commission for Maritime Services (RTCM) SC-104 data format

Radio Technical Commission for Maritime Services (RTCM) is a non-profit scientific, professional and educational organization. The RTCM SC-104 is a data format for Differential GNSS Services of the RTCM [34, Chapter A]. The RTCM SC-104 format supports differential, real-time kinematic and precise GNSS formats. The latest RTCM version is 3.3 and is preferred when working with RTK [10], but in practice version 2 is still in use [1]. In this report the scope is limited to RTCM3.x.

The RTCM SC-104 format consists of different message types. In the format it is specified to use the binary protocol to store data, which means that it is unreadable for humans. The use of a binary protocol results in less data that has to be transferred to the client or the user. The RTCM SC-104 messages includes among other the observations and site metadata. The site metadata includes station coordinates and receiver and antenna descriptions [6]. CMR is believed to be a more efficient alternative for RTCM, this is said by the creators of CMR (Trimble) [13]. Since RTCM version 3 it is not clear if the CMR+ is still better than the free to use RTCM. The newer version CRMx is proprietary by Trmble and not free to use. The reason to start with CMR was to limit the bandwidth in comparison with RTCM. Since RTCM version 3, which is an open standard and supporting GPS, GLONASS and Galileo, it is a more logical choice than a proprietary format like CMRx [31].

RTCM has different kinds of message types, the Common Message Types, the State Space Representation (SSR) Message Types, Multiple Signal Messages (MSM) Message Types and Private Messages [11]. For the manipulation of the augmentation data the Common Message Types and the MSM message types are of importance. For a full list of all possible messages in the RTCM data format one has to purchase a copy of the current RTCM standard[1].

Common Message Types

As the name suggests are the Common Message Types the most common used messages. The way the message types are defined are in line with the previous versions of the RTCM format. With the Common Message Types it is possible to send corrections for the GPS and GLONASS systems. This means that it is not possible to use those messages in combination with the newer Galileo system.

For the manipulation the observations of the reference station need to be altered. Message 1004 "`Extended L1&L2 GPS RTK Observables`" is used to send L1/L2 and SNR data. Message 1012 "`Extended L1&L2 GLONASS RTK Observables`" is the message field that is used for the other GNSS, namely GLONASS. For the manipulation message 1004 and 1012 are thus of interest when the purpose is to manipulate the augmentation data.

Other message fields that can be used by an attacker are message fields 1005, 1006 or 1032. Message field 1005 "`Stationary RTK Reference Station ARP`", message field 1006 "`Stationary RTK Reference Station ARP plus Antenna Height`" and message field 1032 "`Physical Reference Station Position`" contains information about the position of the reference station.

Message field 1013 "`System Parameters`" contains information about the information stream. For example it states which message fields are sent and at which rates. Commonly the observables are sent with a frequency of 1Hz, but the reference station data are commonly sent at a lower frequency, like every 10 ~ 30 seconds [11].

Multiple Signal Messages Message Types

Before MSM became available it was impossible to send all corrections using the RTCM3 format. The format has support for some GPS and GLONASS bands. It was not possible to extend the format for use of other bands (L5 for GPS) or for other GNSS, like Galileo [17].

Reference data can be defined based on corrections or on the observations of the reference receiver. In RTCM 2 the reference corrections $\Delta\mathbf{y}$ are defined as the computed $\mathbf{y}_c$ minus the measured $\mathbf{y}$ range: $\Delta\mathbf{y} = \mathbf{y}_c - \mathbf{y}$. For RTCM 3 GNSS experts decided to provide the receiver observables as raw augmentation data.

The biggest advantage to use MSM is that it is possible the adapt for new GNSS (like Galileo) and new signals (like the L5 band for GPS). Previously one signal was used as the primary observable, for example the L1 pseudo range. When another signal, like L2, is used also the L1 pseudo range should be sent since it is the primary observable. The data for the other signals are defined based on this primary signal.

Conventional RTK requires pseudo range and phase range observations. For RTK MSM3 and MSM4 can be used to send those observations to the rover. The MSM can be used to sent different signals with different frequencies. For example for RTK a smoothed pseudo range can be transmitted at 1 Hz, but the phase range can be transmitted at 10Hz. The decoupling of the different signals saves bandwidth on the data link.

Nowadays the newer MSM format is already in use. It is recommend to choose between the legacy RTCM messages or the MSM, since it is possible that sending both messages cause problems for RTK users [9].

For an attacker not all information is useful. Most important is to be able to manipulate the augmentation data, is to know which fields are used. To know which fields are used to sent the (raw) observation, the attacker can look in message field 1013. The manipulation is based on the design matrix of the user, which is defined based on the user position and a specific model. That information is unknown, but it is assumed that for short baseline RTK the reference station is in the middle of or near the area where the user will use its RTK position. Therefore message field 1005 or 1006 is interesting since those are used to state the position of the reference station. When the MSM are used MSM3 and MSM4 contains the RTK information from the reference station.

[1] https://rtcm.myshopify.com/collections/differential-global…

3.1.2. Receiver INdependent EXchange (RINEX) format

The Receiver INdependent EXchange (RINEX) format stores raw GNSS observations. In practice RINEX is used to store the observations to process the data at a later time. At a later time sometimes it is possible to get satellite position corrections, so that those positions are more accurate. It is also possible to combine multiple RINEX files for example for differential positioning. The advantage is that there is no data link needed for this differential solution. The disadvantage is that the position is not calculated in real time. The RINEX format can be used for different GNSS systems, like GPS and GLONASS.

The latest Receiver INdependent EXchange (RINEX) version is 3.04 (released 23 November 2018), but RINEX 2.11 is still in use. The RINEX 3 standard is now jointly maintained by the International GNSS Service (IGS) RINEX Working Group and the RTCM Special Committee 104. The RINEX data are ASCII text with predefined field widths for data and labels. The RINEX format has a standard for the observation and the navigation data.

For this report RINEX version 2 and 2.11 are used, therefore the description given in this report is applicable to RINEX version 2. The information about the RINEX format is from [19].

RINEX files have a recommended filename structure. When the recommended structure is used, the filename contains information about the station name, date, which session on that date and the file type. There are five file types, namely: `o` observation files, `n` navigation files, `m` meteorological data files, `g` GLONASS observation files and `h` geostationary GPS payload navigation message files. In this case only the navigation and observations files are used. The filename looks like: `ssssdddf.yyt`. The first four characters `s` are used to identify the station, the next three characters `d` is the day of the year, `f` is the file sequence number, `yy` is the year and `t` is the file type. When the file sequence number is zero, it means that the file contains all data for that day.

Each RINEX file starts with a header. The header contains global information for the entire file. In the header column 61 till 80 contains the record descriptions. The first record has to be the “RINEX VERSION / TYPE”, for example:

```
     2.11           OBSERVATION DATA    GPS(GPS)            RINEX VERSION / TYPE
```

The “WAVELENGTH FACT L1/2” record must precede all records defining wavelength factors for individual satellites. And last the “# OF SATELLITES” record must be followed by the corresponding number of “PRN / # OF OBS” records. For the other records it does not matter in which order they are stored. The last record of the header should be the “END OF HEADER” record.

After the header, the observations are stored. Each epoch starts with an “EPOCH/SAT” record. This contains the date and time, epoch flag, number of satellites used in current epoch, list of used satellites and optional receiver clock offset. After this first record, then the observations are stored. Each observation type has a field of 14 characters, with three decimals. After each observation type there is a loss of lock indicator and an integer for the signal strength. An example of one epoch with 6 observation types for 4 satellites is shown below:

```
 19  2 18 12 43  7.0000000  0  4G13G15G17G24
  21503685.61717     7066.85917     1431.18558  21503683.35958       48.250
        34.500
  20643373.06317   -13100.58617    -3749.24758  20643371.88358       50.000
        40.000
  23144554.63316   -44221.78516    -9025.32357  23144551.96157       47.000
        29.250
  21192194.28917   -48742.49217   -11205.30758  21192194.00458       48.750
        39.500
```

The observation is done February 18th 2019, at 12:43:07.0000000. The epoch flag is “0”, which means “OK”. The number of satellites is 4, only GPS satellites. The used satellites are G13, G15, G17, G24.

The navigation file contains the information to calculate the satellite positions. The navigation file contains parameters per satellite. The parameters per satellite are usually updated every two hours [3]. When RTK with a short baseline is used, only one receiver has to store the navigation messages. When a short baseline is used the rover and the reference station will see the same satellites, therefore there is no need to for both the receivers to save a navigation file.

3.2. Communication protocols

Communication protocols specify how the data are transferred from for example a server to a client. Augmentation data are send from a server to a client with a specific protocol. It is possible to sent data using a radio link or over the internet.

Nowadays it is more common to use internet to send the augmentation data, especially when network RTK is used. The protocol that is used when the RTCM SC-104 data are sent over internet is called Networked Transport of RTCM via Internet Protocol (NTRIP) [4].

3.2.1. Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP) is a standard protocol used for the communication between two applications. To use TCP the applications should both be in an Internet Protocol (IP) network. TCP checks if all data packages are received. This means that an application that sends data over a TCP connection does not only send the data in using the network, but also checks if the data that are sent are received correctly. This makes TCP reliable, but due to the checking also relative slow. It is time consuming to check if the data are received correctly and relatively long latency can occur with a TCP connection.

First a connection is established between two applications and when there is a connection the communication can start. To start a connection a three way handshake is used. The first step in the three way handshake is the client which sends a Synchronized Sequence Number (SYN). The server sends the SYN and an Acknowledgement (ACK) back. The last step is the client which sends only the ACK back.

The TCP is part of the Internet Protocol Suite, and is almost always used in combination with the IP. There are four layers used for the Internet Protocol (IP), namely the application layer, the transport layer, the internet layer and the data link layer. The first layer, the application layer, is the layer which has an interaction with the application that requires network communication. Between the application layer and the transport layer Hyptertext Transfer Protocol (HTTP) is used. The transport layer divides the data in packages and the transport layer regulates the ACK with the client or server. Between the transport layer and the internet layer the TCP is used. The last layer is the data link layer. The data link layer handles the hardware.

3.2.2. Hypertext Transfer Protocol

Hyptertext Transfer Protocol (HTTP) is used for data communication on the world wide web (internet). The HTTP is used in the application layer. Normally the HTTP is used for normal browsing on the internet. A web browser normally uses HTTP requests for the data needed to show a webpage.

HTTP is not a secure communication protocol. Another protocol that can be used is Hyptertext Transfer Protocol Secure (HTTPS). HTTPS is the same protocol but it uses Transport Layer Security (TLS), or previous known as Secure Sockets Layer (SSL), for encryption. A HTTPS requests is thus based on HTTP, but the requests and responses are encrypted.

The TLS is used to encrypt the HTTP connection between a client and a server. The latest version of TLS is version 1.3; this version is from 2018. The latest implementations makes sure that the data are encrypted up to 100 years, or until quantum computers come available. TLS is used to exchange information for sensitive applications, for example credit card numbers, social security numbers and banking.

TLS can be used to make the connection between the client and a server encrypted, but also to authenticate the identity of the server. The authentication is a first step in securing the data connection. At the start of a TLS connection the client and server have to exchange the session keys. When the session keys are shared, then the data are encrypted using those keys. This means that when the handshake, the start of a connection, is successfully, then the data are encrypted and protected against eavesdropping. During the handshake the authentication is used to ensure that the data are really from the server using the server certificate. TLS uses different key exchange algorithms. One of the algorithms is the Diffie-Hellman handshake.

Diffie-Hellman

Diffie-Hellman is used for the key exchange. Assume that there are two parties involved, the client and the server. The goal is to setup an encrypted connection, but to create an encrypted connection the two parties can use a set of keys which are only known to them. One method to share two keys over a

public network is the Diffie-Hellman key exchange. The “internet” is in this case a public network, since anyone can listen or read a connection between a client and a server [8].

First there are two numbers which can be seen over the public network, namely the generator g and a prime number n. The server and the client have their own private numbers, a and b. Both the client and the server add their private keys and send that to each other. One number is then $g^a \mod n$ and the other is $g^b \mod n$. When they both add their own private keys they get both the same numbers: $g^{ab} \mod n = g^{ba} \mod n$. The part $\mod n$ makes sure that the number never is bigger than n.

The Diffie-Hellman key exchange method shows the importance of authentication. When no authentication is used, a man-in-the-middle is also able to respond to the messages from the client and the server. If an authentication method is used both the client and the server will know that the response is from an attacker instead of the assumed server or client.

3.2.3. Networked Transport of RTCM via Internet Protocol (NTRIP)

The NTRIP protocol is based on the Hyptertext Transfer Protocol (HTTP) streaming standard. NTRIP is used to send the RTCM SC-104 messages to the rover or the user. The NTRIP software consists of three different elements, namely the NtripClient, NtripServer and the NtripCaster. NTRIP uses TCP/IP for streaming over mobile networks.

The RTCM message can be sent in different ways, for example by radio or over internet using NTRIP. NTRIP is designed so it is possible to receive the augmentation data, but it does nothing with the data itself.The NtripClient and the NtripServer acts as clients in the network and the NtripCaster is the HTTP server.

The reference station(s) are the NtripSource(s). The data goes from the reference station(s) to the NtripServer. The NtripServer(s) send the data to one NtripCaster. It is thus possible that the NtripCaster receives data from multiple reference stations. The NtripClient gets the data of a NtripSource from the NtripCaster.

All data are sent using the extended HTTP (NTRIP) , which is thus unencrypted. To access the NtripCaster, it is possible to use a login and password before you are able to access the NtripCaster data. The client password is based on the HTTP Basic Authentication Scheme, or the Digest Authentication Scheme. The latter uses a hash to confirm the identity of the client. Thereafter the connection is unencrypted and thus unsecure.

The NtripServer sends its data to the NtripCaster. This connection is also unsecure, since no encryption or authentication is used. The documentation also states: “the connection between the client and the server can be regarded as a trusted carrier” [4].

3.3. Man-in-the-middle attack and countermeasures

3.3.1. Man-in-the-middle attack

The man-in-the-middle (MITM) attack is an attack where the attacker is in-between a connection on the internet, where two parties are communicating with each other. The MITM attack can be used to access information that is sent between two parties, for example the client and the server. During a MITM attack it is also possible that the attacker alters the data. When an unsecure connection is used it is possible for the attacker to see all information and alter it without being detected. When authentication and/or encryption is used then it is much harder, or even almost impossible, to alter the data without detection. The MITM is often used for two users in the same network, often in (free) public networks like train stations or coffee shops. It does not matter how the client is connected (wired or wireless) to the network.

When a connection is not secure it is easier to perform a MITM attack. When a secure connection is used the MITM should be in place before the session keys are shared between client and server. In case of NTRIP, the connection is unsecure.

Figure 3.1 shows the schematics of a MITM attack. It is possible that the attacker does nothing with the data except monitoring the data traffic. When a MITM attack is successful it is not known by either the client or the server that there is an attacker in between their connection. It is possible that a MITM attack is performed, but when the data are encrypted the attacker sees only rubbish and can do nothing with the data that are sent over that connection. For NTRIP this means that the authentication part, username and password, is useless for the attacker. This means that the username and password are hidden for the attacker. The data stream itself is not encrypted so after the client has sent its (secret)

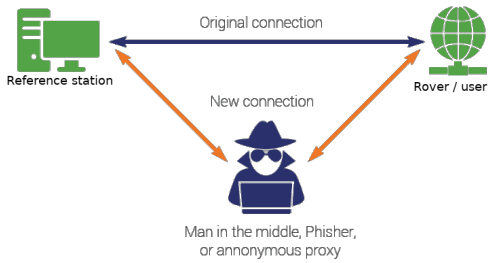


Figure 3.1: Schematic of a Man-in-the-middle attack, after [15]

credentials to the server the server starts transmitting unencrypted data to the client. For the attacker it is possible to see and manipulate the data sent between server and client and vice versa.

There are different methods to perform a MITM attack, the most challenging part is the data interception itself.

3.3.2. Data interception

To be able to be a man-in-the-middle, one should first intercept the data from the client and the server.

Address Resolution Protocol spoofing

The Address Resolution Protocol (ARP) is a protocol used to link an IP address with a device which has its own unique MAC address. When one wants to send data to another device it has to know its IP address, like a telephone number, which can be found using the Address Resolution Protocol. When a sender wants to know the MAC address corresponding to an IP address, it sends an ARP request. The destination node should reply. When an attacker is in the network it is possible that the attacker replies, and thus that the attackers MAC and IP address are used instead of the intended MAC and IP address of the target. This has to be done to the victim and the gateway in the network. ARP does not have any authentication, therefore it is not known by the victim or the gateway that they do not talk directly with each other, but that there is an attacker in-between the connection [26].

The ARP is the basis of the MITM attack, but it is only possible to do this when the attacker and the victim are inside the same network. If a network is trusted, because it is known who is in the network, it is not likely that an attacker is within the same network. For example a home or office network can be trusted since it is managed by yourself, but a public network is not trusted since everybody can access that public network.

Border Gateway Protocol hijacking

The Border Gateway Protocol (BGP) is used to find a path to a specific IP address, or the server. The different Border Gateway Protocol (BGP) routers store routing tables between different Autonomous Systems. An Autonomous System is a network managed by one organization. The BGP connects different large networks, so they can access each other. Using BGP hijacking it is possible to reroute the internet traffic through other paths. This makes it possible to perform for example a man-in-the-middle attack. To perform a BGP attack control of a BGP router is needed. Therefore it is much more difficult to perform a Border Gateway Protocol hijacking than to spoof the Address Resolution Protocol.

Man-in-the-middle attack on mobile networks

NTRIP is used often in combination with mobile internet instead of a wired connection. A mobile connection makes it possible to use a moving RTK receiver. Universal Mobile Telecommunication Standard (UMTS) is one of multiple mobile technologies. It is possible to perform a man-in-the-middle attack on an UMTS connection [25]. The main vulnerability in UMTS is its predecessor Global System for Mobile communications (GSM) which is still in use for example when UMTS is not available.

Even nowadays, with the start of the 5G era, GSM is still in use. Since 3G (UMTS) authentication is used in the system, but when the newer technologies are not available, a cell phone will use GSM. This means that it is possible to perform a man-in-the-middle attack on a mobile network, but it will be

very hard because the victim has first to use GSM which means that the mobile connection must be disturbed.

Cell-Site Simulator
Another device, also used by the government [24], is a Cell-Site Simulator (CSS). A CSS can be used passive and active. When CSS is used in passive mode the simulator does not transmit anything. The passive mode can be used to track mobile connections based on the International Mobile Subscriber Identity (IMSI).

When CSS is used in active mode the CSS broadcasts signals so that mobile devices connect to the CSS instead of the legitimate cell sites. This can be compared with GNSS spoofing on the radio signals. When a mobile connection is created between the CSS and the user, the CSS creates a connection with the actual cell site. The CSS is thus the “man in the middle” and has access to the data that are sent using the mobile connection.

3.4. Countermeasures

The first step to prevent a MITM attack is to use at least some encryption. There is not one solution against all MITM attacks. All kinds of internet applications try to protect themself against those attacks.

An encryption based on TLS is a fist step to prevent the MITM attack. Since the NTRIP is already based on HTTP, the transition to HTTPS is not that big. The difference is that one should have an authorized certificate to be able to share session keys including authentication during the key exchange.

Most augmentation data providers already use a username and password combination before the user is able to receive the augmentation data. The authentication in NTRIP already uses HTTPS, so it should be secure. Maybe it is possible to create a session key and exchange the key over this secure connection.

There are two reasons why a MITM attack is undesirable. First for the user it can be problematic when wrong or no augmentation data are received. The final position can be wrong and depending on the application this can have serious consequences. Furthermore for the provider of the augmentation data. Often the augmentation data are sent by a company which wants to make money of it. If it is possible to receive the augmentation data that is meant for someone near you, the MITM attacker has access to free augmentation data.

4

Data manipulation theory

The first step for augmentation data manipulation is the man-in-the-middle (MITM) attack. In the case a man-in-the-middle attack is accomplished, then the augmentation data can be changed. The challenge is to change the augmentation data in such a way that it cannot be detected by the user. It can be assumed that a user, or the software in the receiver, not only uses the data to estimate the parameters of interest, but that also statistical testing is involved to detect potential outliers in the data. In the end the augmentation data should be manipulated in such a way that it is not detected by the used statistical tests.

First the general idea of parameter estimation and testing is discussed. Thereafter the theoretical approach of undetected data manipulation is introduced [35]. This is followed by two examples using different mathematical models, based on the test outcomes.

4.1. Parameter estimation

Parameter estimation is used when it is impossible to measure the parameter(s) of interest directly. Often something else can be measured and can be used to determine the parameter(s) of interest. It is thus necessary to know how the observables (the measurements) and the parameter(s) of interest, are linked to each other. This relationship is defined in the used model. The parameters of interest for example for GNSS can be the position and receiver clock bias ($[x, y, z, \delta t]$). The position is not measured directly, but time differences are measured and converted to pseudo ranges by receiver. The parameters of interest are estimated based on the observed pseudo ranges.

4.1.1. Best Linear Unbiased Estimator

For parameter estimation of GNSS models the Best Linear Unbiased Estimator (BLUE) is used. The BLUE is a special case of the Weighted Least Square Estimator (WLSE), namely the case where the weight matrix is the inverse variance matrix of the observables. The BLUE is defined as:

$$\underline{\hat{\mathbf{x}}} = \left(A^T Q_{\underline{y}\underline{y}}^{-1} A\right)^{-1} A^T Q_{\underline{y}\underline{y}}^{-1} \underline{\mathbf{y}} \tag{4.1}$$

Where $\underline{\hat{\mathbf{x}}}$ is a $[n \times 1]$ vector, $\underline{\hat{\mathbf{x}}}$ is the estimator for the parameters of interest. The vector $\underline{\mathbf{y}}$ is a $[m \times 1]$ vector containing the observables. Matrix A is the design matrix. The design matrix A describes the linear relationship between the parameters of interest and the observables ($\mathbf{y} = A\mathbf{x}$). It is possible to have more observables than parameters ($m > n$). If the observables do not have a random behaviour, then the system $\mathbf{y} = A\mathbf{x}$ is consistent. When the observables do have a random behavior then it is unlikely that the system of equations is consistent. When dealing with random variables the model becomes $\underline{\mathbf{y}} = A\mathbf{x} + \underline{\mathbf{e}}$. The vector $\underline{\mathbf{e}}$ contains the residuals, which are also random variables. The estimated residuals are defined as $\underline{\hat{\mathbf{e}}} = \underline{\mathbf{y}} - A\underline{\hat{\mathbf{x}}}$. When a non-linear model is used ($\mathbf{y} = F(\mathbf{x})$), The estimated residuals are defined as $\underline{\hat{\mathbf{e}}} = \underline{\mathbf{y}} - \underline{\hat{\mathbf{y}}}$ where $\underline{\hat{\mathbf{y}}} = F(\hat{\mathbf{x}})$.

The underline under a variable is used to describe a random variable. For example $\underline{y} \sim \mathcal{N}(\mu, \sigma^2)$ is a normally distributed random variable with mean μ and variance σ^2. When a variable is noted as y it

is a random draw from the random variable $\underline{y}$. The bold notation is used for vectors, i.e. $\underline{\mathbf{y}} = \begin{bmatrix} \underline{y_1} \cdots \underline{y_m} \end{bmatrix}^T$ and $\mathbf{y} = [y_1 \cdots y_m]^T$.

The observables are assumed to be normally distributed ($\mathcal{N}$) with a known variance matrix ($\mathrm{D}(\underline{\mathbf{y}})$). The mathematics model is given in Equation (4.2).

$$\mathrm{E}(\underline{\mathbf{y}}) = A\mathbf{x} \qquad \mathrm{D}(\underline{\mathbf{y}}) = Q_{yy} \tag{4.2}$$

Where $E(\cdot)$ is the mathematical expectation operator or the mean, and $D(\cdot)$ is the mathematical dispersion operator or the variance [36, Chapter 2.4]. The mathematical definition of the mean and variance is given in Equation (4.3).

$$\mathrm{E}(\underline{x}) = \int_{-\infty}^{\infty} x f_{\underline{x}}(x) dx \qquad \mathrm{D}(\underline{x}) = \mathrm{E}\left(\left(\underline{x} - \mathrm{E}(\underline{x})\right)^2\right) \tag{4.3}$$

Where $f_{\underline{x}}(x)$ is the probability density function of $\underline{x}$. The expected values for a vector is the expected value for each element. The variance is then given as a square matrix with the standard deviation of each element squared on the diagonal.

$$\mathrm{E}(\underline{\mathbf{y}}) = [\mathrm{E}(\underline{y}_1), \mathrm{E}(\underline{y}_2), \ldots, \mathrm{E}(\underline{y}_m)]^T \qquad \mathrm{D}(\underline{\mathbf{y}}) = \mathrm{E}\left((\underline{\mathbf{y}} - \mathrm{E}(\underline{\mathbf{y}}))(\underline{\mathbf{y}} - \mathrm{E}(\underline{\mathbf{y}}))^T\right) \tag{4.4}$$

In this case the observables are normally distributed with a known variance matrix, i.e.

$$\underline{\mathbf{y}} \sim \mathcal{N}\left(\mathrm{E}(\underline{\mathbf{y}}), \mathrm{D}(\underline{\mathbf{y}})\right)$$

Since the variance matrix of the observables is known it is also possible to define the variance matrix of the estimator. This means that the variance of the estimator is based on a-priori knowledge about the statistical information of the observables and the design matrix.

$$\mathrm{E}(\underline{\hat{\mathbf{x}}}) = \mathbf{x} \qquad \mathrm{D}(\underline{\hat{\mathbf{x}}}) = Q_{\hat{x}\hat{x}} = \left(A^T Q_{yy}^{-1} A\right)^{-1} \tag{4.5}$$

This shows that the expected values for the residuals are zero, see Equation (4.6).

$$\mathrm{E}(\underline{\hat{\mathbf{e}}}) = \mathrm{E}(\underline{\mathbf{y}}) - A\,\mathrm{E}(\underline{\hat{\mathbf{x}}}) = 0 \qquad \mathrm{D}(\underline{\hat{\mathbf{e}}}) = Q_{\hat{e}\hat{e}} = Q_{yy} - Q_{\hat{y}\hat{y}} \tag{4.6}$$

The variance for the estimated observables differ compared to the direct (independently) measured observables. The estimated observables are described below in Equation (4.7).

$$\underline{\hat{\mathbf{y}}} = A\underline{\hat{\mathbf{x}}} \qquad \mathrm{D}(\underline{\hat{\mathbf{y}}}) = Q_{\hat{y}\hat{y}} = AQ_{\hat{x}\hat{x}}A^T = A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T \tag{4.7}$$

It is possible that the a-priori model shows no correlation between the measurements, but that the estimated observables show correlation in their variance matrix.

Non-linear models The BLUE method is based on linear models. It is possible to have a non-linear model, for example the GNSS models, and still use the BLUE for parameter estimation. To use the BLUE linearisation is used to transform the non linear model to a linear model. The linearisation is based on the first two terms of the Taylor expansion. To linearise a non-linear function $F(\mathbf{x})$ in $\mathbf{x}_0$, Equation (4.8) can be used.

$$L_F(\mathbf{x}) = F(\mathbf{x}_0) + \nabla F(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) \tag{4.8}$$

For linearisation an approximate value for $\mathbf{x}_0$ has to be chosen, so that the linearisation can be done in that point. The value of $\mathbf{x}_0$ should be approximately equal to $\mathbf{x}$, therefore an educated guess is made for $\mathbf{x}_0$. Using the BLUE for a non-linear model it is often used in combination with Gauss-Newton iteration, see Figure 4.1 for a schematic of the Gauss-Newton iteration [36].

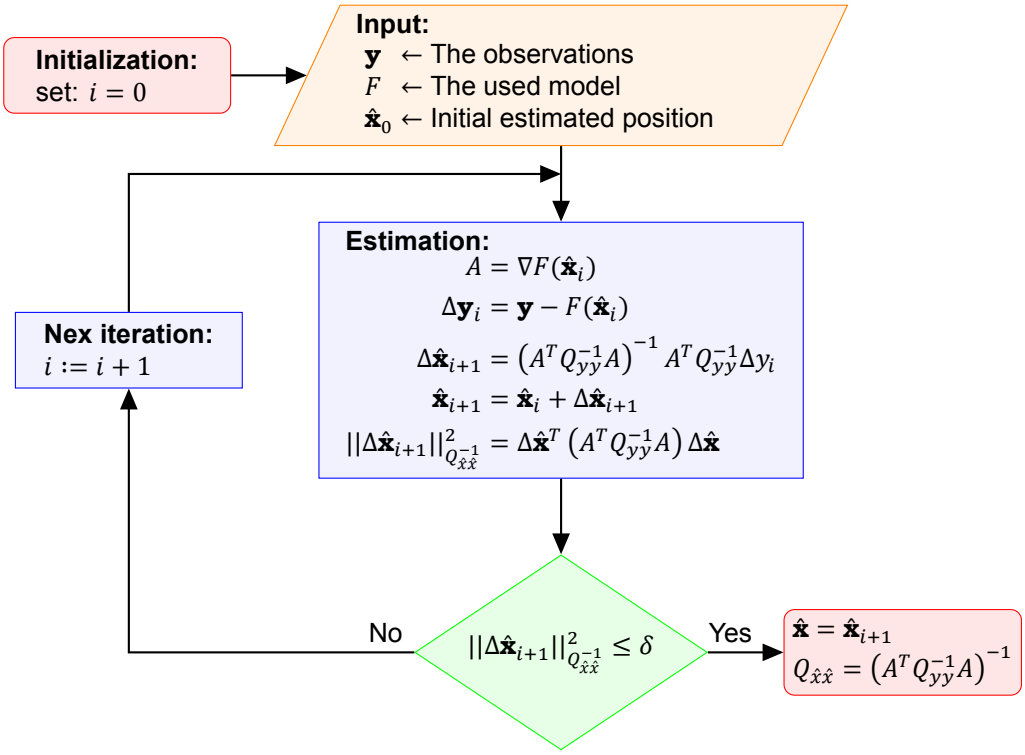


Figure 4.1: Schematic of the Gauss-Newton iteration. The iteration stops when $||\Delta\hat{\mathbf{x}}_{i+1}||^2_{Q^{-1}_{\hat{x}\hat{x}}} \leq \delta$ where δ is the threshold value. The value of δ can be chosen in the order of $\mathcal{O}(1e-10)$ or lower. The final value of $||\Delta\hat{\mathbf{x}}_{i+1}||^2_{Q^{-1}_{\hat{x}\hat{x}}}$ tells nothing about the accuracy of $\hat{\mathbf{x}}_{i+1}$, only that $L_F(\hat{\mathbf{x}}_{i+1}) \approx F(\hat{\mathbf{x}}_{i+1})$.

When $\mathbf{x} = \mathbf{x}_0$ it can be seen that the linearisation is equal to the function value in $\mathbf{x}$, $L_F(\mathbf{x}) = F(\mathbf{x}_0) + \nabla F(\mathbf{x}_0) \cdot 0 = F(\mathbf{x})$ and for $\mathbf{x}$ not too far away from $\mathbf{x}_0$, we have $L_F(\mathbf{x}) \approx F(\mathbf{x})$. When the BLUE is used for such a model the observables are $\mathrm{E}(\underline{\mathbf{y}}) = F(\mathbf{x})$, and the unknown parameter(s) of interest is vector $\mathbf{x}$.

When $\mathbf{x}$ is a $[n \times 1]$ vector containing n parameters, the Jacobian $J(F) = \nabla F(\mathbf{x}_0)$ is the design matrix (A). To solve the BLUE Equation (4.8) can be rewritten:

$$\mathbf{y} - F(\mathbf{x}_0) = \nabla F(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) \tag{4.9}$$

The vector with calculated values is noted as $\mathbf{y}_0 = F(\mathbf{x}_0)$ and the unknowns $(\mathbf{x} - \mathbf{x}_0) = \Delta\mathbf{x}$. When the calculated observables $(\mathbf{y}_0)$ in the initial location is almost the same as the observables then $\Delta\mathbf{x}$ will also be almost zero. The final parameter of interest is defined as $\mathbf{x} = \mathbf{x}_0 + \Delta\mathbf{x}$. In the final form it looks like:

$$\Delta\mathbf{y} = \nabla F(\mathbf{x}_0)\Delta\mathbf{x} \tag{4.10}$$

If the algorithm converges, then the estimator $\hat{\mathbf{x}}_{i+1}$ is a (local) minimum of the least square norm, $(\arg\min_{\mathbf{x}} ||\mathbf{y} - F(\mathbf{x})||^2_{Q_{yy}^{-1}})$ [36].

4.1.2. Statistical testing

For parameter estimation one also needs to test the solution to check if the assumed model was correct and if there were no errors in the observations. This is also called “detection and validation”. When the BLUE is used it is possible to use the a-priori information, $(D(\underline{\mathbf{y}}))$, for testing.

In general a hypothesis, denoted with a H, is tested against another hypothesis. The initial hypothesis, or the one that is believed to be true, is called the null hypothesis H_0. Another hypothesis can be called the alternative hypothesis H_a. When there are multiple alternative hypotheses then the subscript is a number H_1. The hypothesis, see Equation (4.11), is a conjecture about the probability density function (PDF) of a random variable [36].

$$H : \underline{\mathbf{y}} \sim f_{\underline{\mathbf{y}}}(\mathbf{y}|\mathbf{x}) \tag{4.11}$$

A sample of the observable $\underline{\mathbf{y}}$ is used to determine if the null hypothesis should be rejected or not. The test is specified by the critical region K. The area that is not covered by K is the region where the null hypothesis is not rejected. Region K^c is the complement of the region K.

Hypothesis testing Hypothesis testing is the basis of both the overall model test and the w-test. The overall model test is used to check if the used mathematical model should be rejected or can be accepted based on the observations. The w-test is used to test individual observations for outliers. Outliers, or biases, in the observations are denoted with the vector $\mathbf{b}_y$. The orthogonal decomposition of bias $\mathbf{b}_y$ is an influential bias $\mathbf{b}_{\hat{y}}$ and a testable bias $\mathbf{b}_{\hat{e}}$, i.e. $\mathbf{b}_y = \mathbf{b}_{\hat{y}} + \mathbf{b}_{\hat{e}}$ [34, chapter 24]. The influential bias is spanned by the column space of matrix A, $\mathbf{b}_{\hat{y}} \in \mathcal{R}(A)$. This means that $\mathbf{b}_{\hat{y}}$ is the orthogonal projection of $\mathbf{b}_y$ onto A. The influential bias translates into a bias in the final estimate $\hat{\mathbf{x}}$. The testable bias is in the orthogonal complement of the column space of matrix A, i.e. $\mathbf{b}_{\hat{e}} \in \mathcal{R}^{\perp}(A)$ (the left null space). This means that if the bias $\mathbf{b}_y$ changes only in the testable bias space ($\mathcal{R}^{\perp}(A)$), the final estimate for $\hat{\mathbf{x}}$ stays the same. This also works the other way around, if $\mathbf{b}_y$ only changes in the influential space ($\mathcal{R}(A)$), then the estimate $\hat{\mathbf{x}}$ changes but the residuals stay the same. Note that A is based on a linear model. When a non-linear model is used in combination with the Gauss-Newton iteration the design matrix will change each iteration step. This means that the column space $\mathcal{R}(A)$ and the orthogonal complement of A will also change during the iteration.

The used test method is the Generalized Likelihood Ratio (GLR), see Equation (4.12) [36, chapter 7].

$$\text{reject } H_0 \text{ if } \frac{\max_{x\in\Phi_0} f_{\underline{y}}(y|x)}{\max_{x\in\Phi} f_{\underline{y}}(y|x)} < a \tag{4.12}$$

The random variable $\underline{y}$ is described by a PDF $f_{\underline{y}}$. For each value y the PDF has a value based on this value and the parameters describing this PDF, in this case defined as x. This can be for example the mean and the variance of the PDF. In the numerator the space of x is limited to the null hypothesis. What this function does in words is determine the maximum value for the PDF based on y and with $x \in \Phi_0$. The same is done in the denominator, but this time x is not longer limited to the subspace of $x \in \Phi_0$. When the maximum value for the PDF lies in the region described by the null hypothesis the ratio will be high, with the extreme case that the ratio will be 1. This means that the null hypothesis is not rejected [36].

Because the GLR is used, only two hypotheses are tested at one time. When the observables are normally distributed the GLR can be written in another form, namely:

$$\text{reject } H_0 \text{ if } \hat{e}_0^T Q_{yy}^{-1} \hat{e}_0 - \hat{e}_a^T Q_{yy}^{-1} \hat{e}_a > k_\alpha \tag{4.13}$$

The subscripts $\dot{()}_0$ and $\dot{()}_a$ are used to define the null hypothesis or the alternative hypothesis. The estimated residuals $\hat{e}$ are based on the model which corresponds with the hypothesis, the model defined for the null hypothesis or the alternative model. The estimated residuals are defined as $\hat{e} = y - \hat{y}$.

The threshold values for a or k_α are coupled, $k_\alpha = \ln\left(\frac{1}{a^2}\right)$. This last form is also called the test statistic $T_q = \hat{e}_0^T Q_{yy}^{-1} \hat{e}_0 - \hat{e}_a^T Q_{yy}^{-1} \hat{e}_a$, where q stands for the degree of freedom. The test statistic is Chi-squared distributed $\chi^2(q)$ under H_0.

In general there are thus two hypotheses used at a time. For example the null hypothesis states that there is no bias, this is described in the left column, see Equation (4.14). Note that the subscript $_0$ is used to define the used hypothesis and not the initial guess $\mathbf{x}_0$. The alternative hypothesis is that there is a bias, $\mathbf{b}$, the right column in Equation (4.14).

$$\begin{aligned}
\underline{\hat{\mathbf{x}}}_0 &\overset{H_0}{\sim} \mathcal{N}(\mathbf{x}, Q_{\hat{x}\hat{x}}), & \underline{\hat{\mathbf{x}}}_0 &\overset{H_a}{\sim} \mathcal{N}(\mathbf{x} + \mathbf{b}_{\hat{x}}, Q_{\hat{x}\hat{x}}) \\
\underline{\hat{\mathbf{y}}}_0 &\overset{H_0}{\sim} \mathcal{N}(A\mathbf{x}, Q_{\hat{y}\hat{y}}), & \underline{\hat{\mathbf{y}}}_0 &\overset{H_a}{\sim} \mathcal{N}(A\mathbf{x} + \mathbf{b}_{\hat{y}}, Q_{\hat{y}\hat{y}}) \\
\underline{\hat{\mathbf{e}}}_0 &\overset{H_0}{\sim} \mathcal{N}(0, Q_{\hat{e}\hat{e}}), & \underline{\hat{\mathbf{e}}}_0 &\overset{H_a}{\sim} \mathcal{N}(\mathbf{b}_{\hat{e}}, Q_{\hat{e}\hat{e}})
\end{aligned} \tag{4.14}$$

When in reality the observation contains a bias, but the null hypothesis is not rejected, it is called a missed detection. A Missed Detection (MD) can result in an influential bias.If that happens then the estimate $\hat{\mathbf{x}}$ will be biased. It is also possible that the null hypothesis is rejected while in reality there was no outlier in the observations, this is called a False Alarm (FA). When the alternative hypothesis is used instead of the null hypothesis the final estimate will still be unbiased.

The null hypothesis is shown in Equation (4.15) and the alternative hypothesis is shown in Equation (4.16).

$$H_0 : \mathrm{E}\left(\underline{\mathbf{y}}\right) = A\mathbf{x} \tag{4.15}$$

$$H_a : \mathrm{E}\left(\underline{\mathbf{y}}\right) = A\mathbf{x} + C_y\nabla \tag{4.16}$$

With $[m \times n]$ design matrix A, $[m \times 1]$ vector of observables $\underline{\mathbf{y}}$, $[n \times 1]$ vector of parameters $\mathbf{x}$, $[m \times q]$ matrix for the alternative hypothesis C_y and $[q \times 1]$ vector containing the bias ∇. The bias for the alternative hypothesis is modelled as $\mathbf{b}_y = C_y\nabla$. It is assumed that rank $\left(A, C_y\right) = n + q$. The parameter space ($\Phi$) describes the possible outcomes for the parameters, i.e. $\Phi = \{x \in \mathbb{R}^n, \nabla \in \mathbb{R}^q\}$. In the case of the null hypothesis Φ_0 the additional parameters are expected to be zero, i.e. $\Phi_0 = \{x \in \mathbb{R}^n, \nabla = 0\}$.

If the alternative hypothesis is true, the expectation for the residuals using the null hypothesis is thus not zero anymore, but the bias. However if the bias is detected, the alternative model should be used, which takes the bias into account. Note however that the null hypothesis is only rejected if the testable bias causes the test-statistic to be larger than the threshold k_α in terms of k_α. Therefore it is possible to detect a relatively small bias if most of the bias lies in the orthogonal complement of A, or it is possible to miss a relatively big bias if the bias is mostly in the column space of A.

Overall model test The overall model test is a special case of the Generalized Likelihood Ratio (GLR) test. The alternative hypothesis contains $q = m - n$ extra parameters. Therefore a consistent model of equation originates and thus there are no residuals when the alternative hypothesis is used. The size of the new matrix $\left[A, C_y\right]$ is thus $[m \times m]$. For the overall model test the test statistic Equation (4.13) is used $T_{q=m-n}$. Since $\hat{e}_a = 0$ there is no need to calculate the estimators with both the null and the alternative hypothesis. To perform an overall model test it is thus sufficient to only perform the BLUE with the model of the null hypothesis.

$$T_{q=m-n} = \hat{e}_0^T Q_{yy}^{-1} \hat{e}_0 \tag{4.17}$$

w-test The w-test is a test where the alternative hypothesis matrix C_y is a vector, this means that $q = 1$ and ∇ has size $[1 \times 1]$. Only one parameter is thus added in this alternative hypothesis. Adding only one optional bias as an alternative to the null hypothesis makes it is possible to check if there is a single observation that contains a significant bias. The w-test uses m alternative hypotheses, one for each observation. The C_y vector is filled with zeros and has a 1 at place i, with $i = 1, \ldots, m$. The first alternative hypothesis has a vector that looks like: $C_{y_1} = [1, 0, \cdots, 0]^T$, the second alternative hypothesis has a vector that looks like: $C_{y_2} = [0, 1, 0, \cdots, 0]^T$ and so on.

The test statistic for the w-test is as follow:

$$T_{q=1} = \frac{\left(\mathbf{c}_y^T Q_{yy}^{-1} \hat{\underline{\mathbf{e}}}\right)^2}{\mathbf{c}_y^T Q_{yy}^{-1} Q_{\hat{e}_0 \hat{e}_0} Q_{yy}^{-1} \mathbf{c}_y} \tag{4.18}$$

Here the test statistic is Chi-squared distributed. It is also possible to use a standard normal distribution for the w-test, then $\sqrt{T_{q=1}} = w$. The w-test is then a two-sided test.

For both the overall model test and the w-test it holds that when the test statistic exceeds a critical value k_α the null hypothesis is rejected. In general the w-test is only used when the overall model test is rejected. When the overall model test is rejected it means that there is something wrong with the used model, or with one of the observations. The w-test is then used to identify which observable has the largest test statistic. The next step, the adaptation step, can be to use the corresponding alternative model.

It is not per definition that when the overall model test is rejected that one of the observables contains a blunder. It is also possible that the a-priori information is too optimistic, thus that the variance of the observations is set too small.

4.2. Data manipulation

Based on the information about how the parameters are estimated it is possible to design a method to manipulate the data in such a way that it is impossible to detect based on the statistical tests described before. The GNSS models are linearised and after the linearisation the BLUE is applied. At the end the statistical tests are used to determine if the correct model was used and that there were no outliers in the data.

Based on the BLUE it is possible to manipulate the observables $\underline{\mathbf{y}}$ in such a way that it is not detected by the statistical tests. To do so the introduced bias should lie in the column space of A so that the introduced bias only contributes to the influential bias and contributes nothing to the testable bias.

The manipulation vector is noted as $\tilde{\mathbf{y}}$, which means that the new vector (noted with a bar) $\underline{\bar{\mathbf{y}}} = \underline{\mathbf{y}} + \tilde{\mathbf{y}}$. This new vector is the combination of the random observables and the manipulation vector. As said above the manipulation vector should introduce only an influential bias and thus $\tilde{\mathbf{y}} \in \mathcal{R}(A)$. This means that the whole bias $\tilde{\mathbf{y}} = \mathbf{b}_y$ and that $\tilde{\mathbf{y}} \notin \mathcal{R}^{\perp}$. This means that the mapping of the manipulation vector on this space is also zero, i.e. $\left(\mathbf{I}_m - A\left(\left(A^T Q_{yy}^{-1} A\right)\right)^{-1} A^T Q_{yy}^{-1}\right)\tilde{\mathbf{y}} = \mathbf{0}$ [chapter 24][34].

First the manipulation vector is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}}$ is the manipulation in parameter space. Using the A matrix to define the manipulation vector it means that $\tilde{\mathbf{y}} \in \mathcal{R}(A)$ since $A\mathbf{x} \in \mathcal{R}(A)$. The new estimate is noted with both a hat and a bar to clarify that a manipulated observation vector $\bar{\mathbf{y}}$ is used for the parameter estimation. It is shown in Equation (4.19) that the final residuals, which are used for the statistical tests, are the same when the observables are manipulated with this manipulation vector.

$$
\begin{aligned}
\hat{\bar{\mathbf{e}}} &= \bar{\mathbf{y}} - \hat{\bar{\mathbf{y}}} \\
&= \bar{\mathbf{y}} - A\hat{\bar{\mathbf{x}}} \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \bar{\mathbf{y}} \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \left(\mathbf{y} + \tilde{\mathbf{y}}\right) \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \left(\mathbf{y} + A\tilde{\mathbf{x}}\right) \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \mathbf{y} - AI\tilde{\mathbf{x}} \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \mathbf{y} - \tilde{\mathbf{y}} \\
&= \left(\mathbf{y} + \tilde{\mathbf{y}}\right) - A\hat{\mathbf{x}} - \tilde{\mathbf{y}} \\
&= \mathbf{y} - A\hat{\mathbf{x}} = \hat{\mathbf{e}}
\end{aligned}
\tag{4.19}
$$

What is shown above is that the residual vector is equal for the BLUE result using the original observations and the BLUE result using the manipulated observations.

It is thus possible to find a manipulation vector $\tilde{\mathbf{y}}$ which is not detectable by the statistical tests, since the vector adds nothing to the testable bias. This is based on the linear model and that the A matrix is known. In practice however it most likely not known which A matrix a user uses, which is dependent on the chosen GNSS model and the location of the user. However it is very likely that some kind of iteration is implemented in the software to determine the final position. Due to the iteration the A matrix does change. The manipulation can be done only one time, which means that even if the used model and thus the A matrix is known it is possible to create a testable bias of zero only for the first iteration. After the first iteration a (small) part of the manipulation vector will be most likely in the null space of the new A matrix.

4.3. 2D example non-linear model

A 2D example is used to show the theory using a practical example. The example is about 2D positioning using m known points (s_m). This can be compared with GNSS positioning in 3D where the satellites are assumed to be known points in space. The points are assumed to be deterministic. The position of

the receiver $\mathbf{x}_r$ has to be estimated. In this case the position vector $\mathbf{x}_r$ contains three elements, namely the x_1 and x_2 coordinates and an extra parameter for the receiver clock offset scaled by the speed of light $\delta = c\delta t_r$, $\mathbf{x}_r = [x_{1,r}, x_{2,r}, \delta]$. This offset is an error which is the same for all measurements. This can be compared with a receiver clock bias in GNSS, except in this case the bias is also in meters like the other two elements of the position vector.

The used observables $\underline{\mathbf{y}}$ are the measured distances between the known points and the receiver. Those are used to determine the position of the receiver $\mathbf{x}$. Therefore $\underline{\mathbf{y}}$ is a $[m \times 1]$ vector containing m distances to m unique known points. A distance measurement is defined including the offset (δ), see Equation (4.20).

$$F(\mathbf{x}_r) = \begin{bmatrix} f_1(\mathbf{x}_r) \\ \vdots \\ f_m(\mathbf{x}_r) \end{bmatrix} = \begin{bmatrix} \sqrt{(x_{1,r} - x_1^1)^2 + (x_{2,r} - x_2^1)^2} + \delta \\ \vdots \\ \sqrt{(x_{1,r} - x_1^m)^2 + (x_{2,r} - x_2^m)^2} + \delta \end{bmatrix} \tag{4.20}$$

The design matrix A is the linearisation of the function $\mathbf{y} = F(\mathbf{x})$ in position $\mathbf{x}_0$. $F(\mathbf{x})$ is based on Equation (4.20).

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_{1,0}} & \frac{\partial f_1}{\partial x_{2,0}} & \frac{\partial f_1}{\partial \delta_0} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial x_{1,0}} & \frac{\partial f_m}{\partial x_{2,0}} & \frac{\partial f_m}{\partial \delta_0} \end{bmatrix} = \begin{bmatrix} \frac{x_{1,0} - x_1^1}{f_1} & \frac{x_{2,0} - x_2^1}{f_1} & 1 \\ \vdots & \vdots & \vdots \\ \frac{x_{1,0} - x_1^m}{f_m} & \frac{x_{2,0} - x_2^m}{f_m} & 1 \end{bmatrix}$$

Where $x_{1,0}$ and $x_{2,0}$ are the first two elements of the $\mathbf{x}_0$ vector. The superscript indicates which known point x_1^i and x_2^i with $i = 1, \ldots, m$ was used for the corresponding observable.

The measurements are defined as:

$$\mathbf{y} = \begin{bmatrix} f_1(\mathbf{x}_r) \\ \vdots \\ f_m(\mathbf{x}_r) \end{bmatrix} + \boldsymbol{\varepsilon} \qquad Q_{yy} = \begin{bmatrix} \sigma^2 & & 0 \\ & \ddots & \\ 0 & & \sigma^2 \end{bmatrix}$$

Where $\underline{\boldsymbol{\varepsilon}} = \left[\underline{\varepsilon_1}, \ldots, \underline{\varepsilon_m}\right]^T$ is a vector with random noise, $\underline{\varepsilon_i} \sim N(0, 0.02^2)$ for $i = 1, \ldots, m$. This is used to simulate the measurement noise. The measurement values below, for vector $\mathbf{y}$, result from first computing the true measurement value based on the known positions P_i with $i = 1, \ldots, m$ and the true values for the unknown parameters, $\mathbf{x}$, and then adding a sample from the random noise $\boldsymbol{\varepsilon}$.

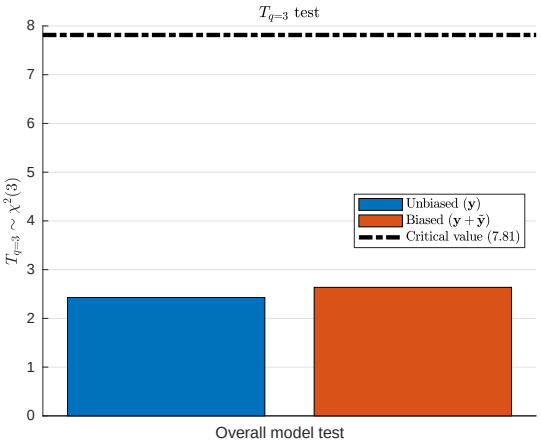
Based on the model described above the position is estimated using the BLUE and the Gauss-Newton iteration. First the results are shown when the observation is not manipulated and next the result is shown where the observation vector is perturbed. The first example is shown in Table 4.1. Figure 4.2 shows the results of the statistical tests and Figure 4.3 shows a 2D figure with the measurement setup and the resulting estimations for $\hat{\mathbf{x}}$ and $\hat{\tilde{\mathbf{x}}}$.

The standard deviation of the observables is 0.02 meter. The table shows the estimated parameter for a case without ($\hat{\mathbf{x}}$) and with ($\hat{\tilde{\mathbf{x}}}$) manipulated observations. To create the manipulation vector $\tilde{\mathbf{y}}$ a bias in parameter space is introduced, $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The manipulation vector is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$. The table shows that the difference between $\hat{\mathbf{x}}$ and $\hat{\tilde{\mathbf{x}}}$ is almost as the chosen bias $\tilde{\mathbf{x}}$, namely $\hat{\tilde{\mathbf{x}}} - \hat{\mathbf{x}} = [0.96, 2.17, 0.13]^T$. This is not exactly the same as $\tilde{\mathbf{x}}$, because the Gauss-Newton iteration is used. The bias is only introduced using the design matrix A which is defined using the linearisation in $\mathbf{x}_0$, but during the iteration the A matrix does change slightly and therefore the introduced bias is not only in $\mathcal{R}(A)$ anymore, but also partly in $\mathcal{R}^\perp(A)$.

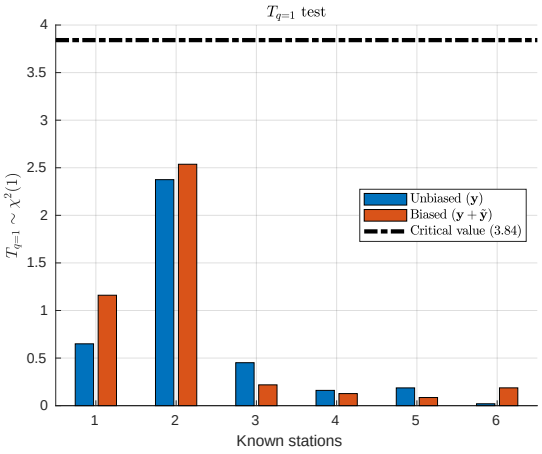
The overall model test is normally the first statistical test used to check if the estimated parameter can be correct. In this case an alpha level of 0.05 is used. The model under H_0 has 6 observations (m) and 3 parameters (n) thus 3 degrees of freedom, $q = m - n$. The corresponding critical value with the given alpha level and the $\chi^2(3, 0)$ distribution is 7.8147. In both cases the test-statistics are much smaller than this critical value thus the null hypothesis is not rejected, see Equation (4.13).

	x[m]	y[m]	δ[m]	$\mathbf{y}$[m]	$\bar{\mathbf{y}}$[m]	$\tilde{\mathbf{y}}$[m]	$T_{q=m-n}$	$T_{q=1}$ for $\hat{\mathbf{x}}$	$T_{q=1}$ for $\hat{\tilde{\mathbf{x}}}$
P_1	60	120		125.29	123.98	-1.31		0.650	1.160
P_2	35	110		131.27	130.38	-0.89		2.375	2.537
P_3	20	90		128.05	127.59	-0.45		0.451	0.218
P_4	5	40		118.83	119.36	0.53		0.160	0.127
P_5	15	20		105.46	106.36	0.90		0.187	0.085
P_6	140	120		111.79	109.61	-2.18		0.020	0.187
$\mathbf{x}_0$	115	15	0						
$\mathbf{x}$	120	10	0						
$\hat{\mathbf{x}}$	120.02	9.96	-0.05				2.428		
$\hat{\tilde{\mathbf{x}}}$	120.99	12.12	0.08				2.637		

Table 4.1: The used positions and simulated observations for the 2D example. The manipulation vector $\tilde{\mathbf{y}}$ is defined as $\tilde{\mathbf{y}} = A_1\tilde{\mathbf{x}}$ (A_1 is the first design matrix during the iteration). $\mathbf{x}$ denotes the true position (and clock bias in meters). The used manipulation vector in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The standard deviation of the observables is 0.02 meter. Both the overall model test and the w-test are not rejected, thus H_0 is assumed to be true, even when the data is manipulated.



(a) Overall model test $T_{q=3}$. The blue bar is the unmanipulated test-statistic ($T_{q=3}(\hat{\mathbf{e}})$) and the orange bar is the manipulated test-statistic ($T_{q=3}(\hat{\tilde{\mathbf{e}}})$)

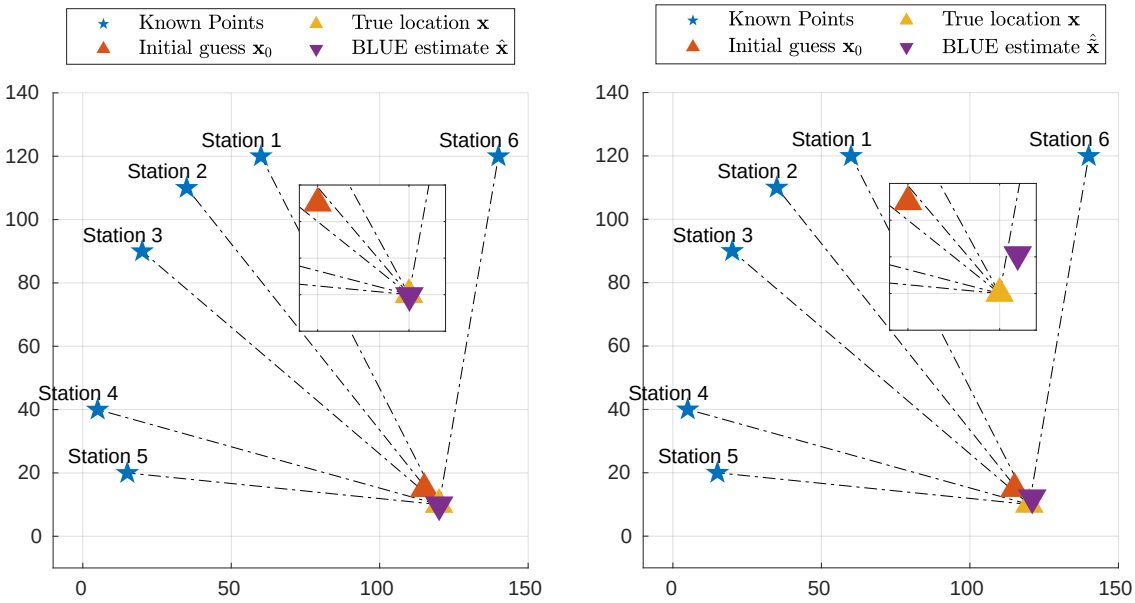


(b) W-test $T_{q=1}$, one test-statistic per observation. The blue bars are the unmanipulated test-statistics ($T_{q=1}(\hat{\mathbf{e}})$) and the orange bars are the manipulated test-statistics ($T_{q=1}(\hat{\tilde{\mathbf{e}}})$).

Figure 4.2: The values for the test-statistics for both the unmanipulated and the manipulated cases. In both cases and for both tests (overall model test and w-test) the test-statistic does not exceeds the critical value (the dotted line). The values can be found in Table 4.1

The critical value for the w-test is defined with the same alpha level (0.05). The critical value for the w-test with the $\chi^2(1, 0)$ distribution is 3.8415. None of the observations exceeds this threshold therefore the null hypothesis is not rejected. As said it is also possible to take the square root of the w-test so that the critical value is in the same units as the observations. Then the test-statistic is standard normal distributed, $\mathcal{N}(0, 1)$ and the w-test becomes a two-sided test.

In Figure 4.3 the results are shown in 2D, thus the scaled clock bias is not shown. The left figure shows the result without manipulation of the observations. On the right the observations are manipulated with $A\tilde{\mathbf{x}}$. This can also be seen since the final estimate $\hat{\tilde{\mathbf{x}}}$ does not lie on top of the true location of $\mathbf{x}$.



(a) BLUE estimation without manipulation ($\hat{\mathbf{x}}$). Overall model test accepted.

(b) BLUE estimation with manipulation ($\hat{\tilde{\mathbf{x}}}$). Overall model test accepted.

Figure 4.3: The BLUE estimation for both the original example on the left and the manipulated example on the right. The standard deviation of the observables is 0.02 meter. The manipulation in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$.

What can be concluded from this example is that in this specific case the manipulation vector did not contribute enough to the testable bias to be detected. In a perfect case the values for the test-statistics for the overall model test and the w-test should be the same in the manipulated and the original case. The introduced bias in parameter space is relatively large compared to distances between the known points and the receiver location $\mathbf{x}$. The length of the final manipulation in parameter space is $|\hat{\tilde{\mathbf{x}}}| = |\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}| = 2.38$. The average distance between the location $\mathbf{x}$ and the known points is 120 meter. Therefore the length of the manipulation vector is approximately 2 percent of the average distance. When this method is used for GNSS an average distance between Earth and the GNSS satellites is 20.200 km. The corresponding manipulation vector would then be approximately 400 km. Normally RTK-GNSS does not work over such large distances therefore it can be assumed that the difference in test-statistics resulting from the non-linear behaviour of the BLUE is far less than in this example. When the manipulation of augmentation data is used for GNSS, the user can also use its own stand alone position, using solely the measurements collected by its own receiver, thus independent of the (manipulated) augmentation data. The stand alone precision of GNSS is in order of meters. Therefore the stand alone precision is the limitation of the length of $|\tilde{\mathbf{x}}|$ and not the possibility of detection by the used statistical tests.

When one uses linearisation an error can be defined as $\varepsilon = |f(x) - L_{f(x_0)}(x)|$. It is possible to create a set of values $x \in \Omega$ for which holds that $\varepsilon < \delta$, where δ is a constant. The linearisation error of a root function can be written using an Taylor expansion in point x_0, see Equation (4.21), the derivation can be found in Appendix A. The root function is used since distance measurements are based on root functions (see Equation (4.20)). The series start at $n = 2$ since the first two terms of the Taylor expansion of $f(x)$ are the same as the linearisation.

$$\varepsilon = \sum_{n=2}^{\infty} (-1)^{n-1} \frac{\prod_{n=2}^{n} (2n-3)}{2^n} x_0^{(\frac{1}{2}-n)} \frac{(x-x_0)^n}{n!} \tag{4.21}$$

What can be seen is that when $x_0 \to \infty$ then $\varepsilon \to 0$. This means that when x_0 increases the area (Ω) with a maximum error (δ) also increases. When dealing with GNSS the distances between a satellite and the user is relatively large. This means that the initial guess $\mathbf{x}_0$ is relatively close to $\mathbf{x}$ so $|\mathbf{x} - \mathbf{x}_0|$ will be relatively small. The sphere describing equal distance from the satellite to Earth is locally almost

linear, and the sphere is by approximation a plane. This means that the design matrix A does not change that much during the Gauss-Newton iteration. Locally means here a few kilometers.

Example with an outlier The next example shows the same set-up with almost the same observables. This time there is an outlier in the third measurement of 0.2 meter, ten times the standard deviation. The manipulation vector $\tilde{\mathbf{y}}$ stays the same since the A is not influenced by the bias in the observations. The values are shown in Table 4.2.

	x[m]	y[m]	δ[m]	$\mathbf{y}$[m]	$\bar{\mathbf{y}}$[m]	$\tilde{\mathbf{y}}$[m]	$T_{q=m-n}$	$T_{q=1}$ for $\hat{\mathbf{x}}$	$T_{q=1}$ for $\hat{\tilde{\mathbf{x}}}$
P_1	60	120		125.29	123.98	-1.31		19.179	21.622
P_2	35	110		131.27	130.38	-0.89		5.465	5.185
P_3	20	90		128.25	127.79	-0.45		59.078	62.453
P_4	5	40		118.83	119.36	0.53		5.083	4.902
P_5	15	20		105.46	106.36	0.90		0.149	0.293
P_6	140	120		111.79	109.61	-2.18		31.129	34.098
$\mathbf{x}_0$	115	15	0						
$\mathbf{x}$	120	10	0						
$\hat{\mathbf{x}}$	120.16	9.80	-0.19				61.138		
$\hat{\tilde{\mathbf{x}}}$	121.12	11.98	-0.06				64.870		

Table 4.2: The used positions and simulated observations for the 2D example. The manipulation vector $\tilde{\mathbf{y}}$ is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$. And the used manipulation vector in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The standard deviation of the observables is 0.02 meter. There is an outlier of 0.2 meter in the third measurement. The overall model test is in both cases rejected. The w-test shows the biggest bias in the third measurement and is also rejected. The test-statistics above the critical value are showed in red. The test-statistics and the final estimates are based on the H_0 hypothesis.

The results are also shown in Figure 4.4. In the figures it can be seen that there is still no significant difference between the test-statistic from the original example or the one which used the manipulated data.

There is an outlier in the third observation, which is also detected according the overall model test and the w-test. Therefore the null hypothesis is rejected and thus the alternative model is used next. The manipulation vector is still based on the original A matrix which is used for H_0.

There are two options for the implementation of the alternative hypothesis. It is possible to remove the third observation from the set of observations or it is possible to add a parameter that should be estimated in the model. For the final result it makes no difference since only the third observation contributes to the new parameter. In this case it is chosen to estimate the bias in the third measurement to see how well the bias is detected.

As said the manipulation vector $\tilde{\mathbf{y}}$ stays the same and is based on the design matrix with three columns. The new design matrix has four columns since the bias is added as unknown parameter.

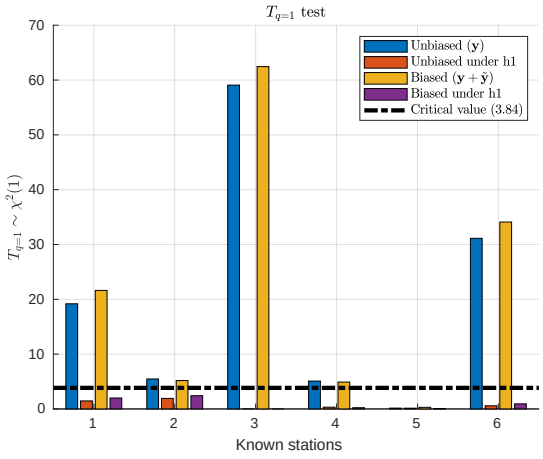
In Table 4.3 the results with the alternative hypothesis are shown, accommodating an outlier in the third observation. The w-test-statistics for the third measurement is zero since all the “error” in this measurement is captured by the estimate of ∇. This is the case because the third measurement is the only measurement that is used to estimate ∇.

In this case the design matrix is significantly changed compared to the start when the influential bias was defined. Still the manipulation is not detected by the statistical tests.

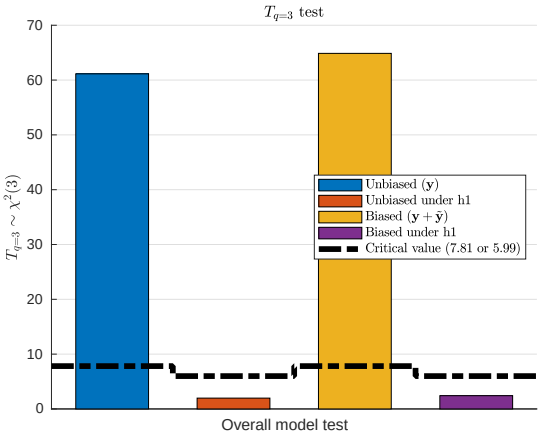
The new test-statistics are also shown in Figure 4.4. The critical value for the overall model test is changed between H_0 and H_a since the degree of freedom has decreased with one.

	x[m]	y[m]	δ[m]	∇	$\mathbf{y}$[m]	$\bar{\mathbf{y}}$[m]	$\tilde{\mathbf{y}}$[m]	$T_{q=m-n}$	$T_{q=1}$ for $\hat{\mathbf{x}}$	$T_{q=1}$ for $\hat{\hat{\mathbf{x}}}$
P_1	60	120			125.29	123.98	-1.31		1.461	1.993
P_2	35	110			131.27	130.38	-0.89		1.922	2.410
P_3	20	90			128.25	127.79	-0.45		0.000	0.000
P_4	5	40			118.83	119.36	0.53		0.318	0.222
P_5	15	20			105.46	106.36	0.90		0.135	0.061
P_6	140	120			111.79	109.61	-2.18		0.572	0.924
$\mathbf{x}_0$	115	15	0	0						
$\mathbf{x}$	120	10	0	0.2						
$\hat{\mathbf{x}}$	120.04	9.94	-0.06	0.18				1.976		
$\hat{\hat{\mathbf{x}}}$	120.99	12.12	0.07	0.19				2.418		

Table 4.3: The used positions and simulated observations for the 2D example. The manipulation vector $\tilde{\mathbf{y}}$ is defined as $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$. And the used manipulation vector in parameter space is $\tilde{\mathbf{x}} = [1, 2, 0]^T$. The standard deviation of the observables is 0.02 meter. There is an outlier of 0.2 meter in the third measurement. The overall model test and the w-tests are in both cases accepted. The test-statistics and the final estimates are based on the H_a hypothesis. The bias ∇ is added to the model and thus also estimated. The third measurement is grayed since this measurement does not contribute to parameters of interest anymore, but it is used to estimate ∇.



(a) W-test $T_{q=1}$, one test-statistic per observation. The blue bars are the unmanipulated test-statistics ($T_{q=1}(\hat{\mathbf{e}})$) based on H_0. The yellow bars are the manipulated test-statistics ($T_{q=1}(\hat{\hat{\mathbf{e}}})$) also based on H_0. The orange and purple bars are the w-test-statistics based on H_a.



(b) Overall model test $T_{q=3}$ and $T_{q=2}$. The blue bar is the unmanipulated test-statistic ($T_{q=3}(\hat{\mathbf{e}})$) and the yellow bar is the manipulated test-statistic ($T_{q=3}(\hat{\hat{\mathbf{e}}})$) both based on H_0. The orange and purple bars are the unmanipulated and manipulated test-statistics based on H_a ($T_{q=2}(\hat{\mathbf{e}})$ and $T_{q=2}(\hat{\hat{\mathbf{e}}})$). The critical value is changed between H_0 and H_a.

Figure 4.4: The values for the test-statistics for both the unmanipulated and the manipulated cases. The null hypothesis is rejected since the test-statistic exceeds the critical value. The values can be found in Table 4.2 for H_0 and Table 4.3.

Correlation

What is shown is that the test statistic of the w-test differs slightly when the unmanipulated solution and manipulated solution are compared. The size of the difference in the w-test statistic correlates with the direction of the manipulation direction. The difference between the w-test statistic is given as $\Delta T_{q=1} = T_{q=1} - \bar{T}_{q=1}$, where $T_{q=1}$ is the w-test statistic based on the unmanipulated solution and $\bar{T}_{q=1}$ is the w-test statistic for the manipulated solution. When for each observation this w-test is calculated, it is combined in one vector $\mathbf{T}_{q=1}$. Each element i of $\mathbf{T}_{q=1}$ corresponds with the w-test for observation i in $\mathbf{y}$.

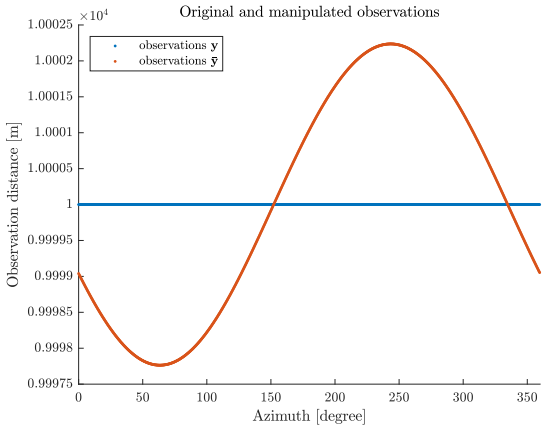
When the manipulation is only in the influential bias then $\Delta\mathbf{T}_{q=1} = \mathbf{0}$. As shown before this is not exactly the case, since the design matrix A slightly change during the estimation of $\hat{\mathbf{x}}$. The change in the linearisation point is the reason that $\Delta\mathbf{T}_{q=1} \neq \mathbf{0}$. The linearisation of a sphere in point $\mathbf{x}_0$ results in a plane with a normal vector which is parallel to the unit direction vector of the linearisation point towards the known point, for example a satellite. During the iteration the linearisation point changes.

When the three points, the known point P, the true location $\mathbf{x}$ and the linearisation point $\mathbf{x}_0$ are all on one line then there is no difference in the linearisation between point $\mathbf{x}_0$ and $\mathbf{x}$, since the direction from $\mathbf{x}$ to P is the same as the direction from $\mathbf{x}_0$ to P.

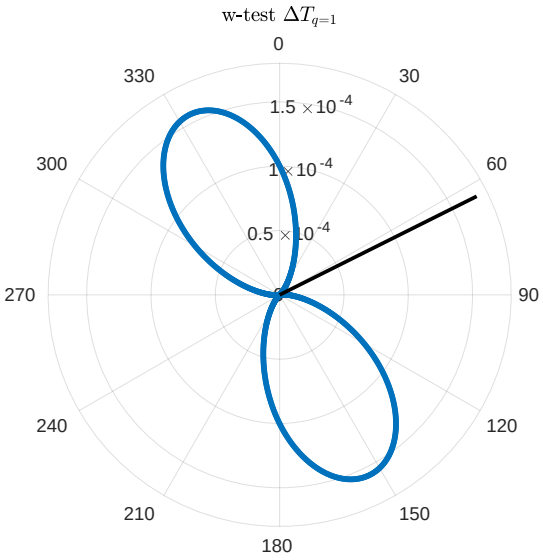
This knowledge can be applied to the manipulation vector $\tilde{x}$. When the observations are manipulated then the final estimate is no longer $\mathbf{x}$ but $\bar{\mathbf{x}}$. This means that the final A matrix is also defined in $\bar{\mathbf{x}}$. The linearisation of a known point,which is on one line with $\mathbf{x}$ and $\bar{\mathbf{x}}$, has the same value for both $\mathbf{x}$ and $\bar{\mathbf{x}}$.

To show this effect another measurement setup is simulated. For this simulation 1080 known points are used. The true location is $\mathbf{x} = [0, 0]^T$. First the scaled clock bias $c\delta t$ is not used in the model. The known points are equally distributed around the known point, with a radius of 1000 meter. The initial position is equal to the true location, $\mathbf{x}_0 = \mathbf{x}$. The manipulation is $\bar{\mathbf{x}} = [2, 1]^T$. In total there are thus $m = 1080$ observations, and 1080 different w-tests. Since the known points are distributed around $\mathbf{x}$, there are two points which are on one line with $\mathbf{x}$ and $\bar{\mathbf{x}}$. The linearisation of those two points in $\mathbf{x}$ and $\bar{\mathbf{x}}$ are equal, and therefore the w-test should be zero. A standard deviation of $\sigma = 0.05$ meter is used for to calculate the w-test. The simulated observations does not contain noise.

Below in Figure 4.5 the observations $\mathbf{y}$ and the manipulated observations $\bar{\mathbf{y}}$ are shown. What can be seen is that there are two points where $\mathbf{y} = \bar{\mathbf{y}}$. The observations are plotted against the azimuth from $\mathbf{x}$ towards the known points. The known points where $\mathbf{y} = \bar{\mathbf{y}}$ have an azimuth of 153 and 334 degree. The manipulation $\bar{\mathbf{x}}$ has an azimuth of 63.43 degree, and since $\mathbf{x} = [0, 0]^T$ this is also the direction between $\mathbf{x}$ and $\bar{\mathbf{x}}$. In the next plot the test statistic is shown for the w-test. The direction of the manipulation is shown with a black line. The magnitude of the manipulation is not to scale.



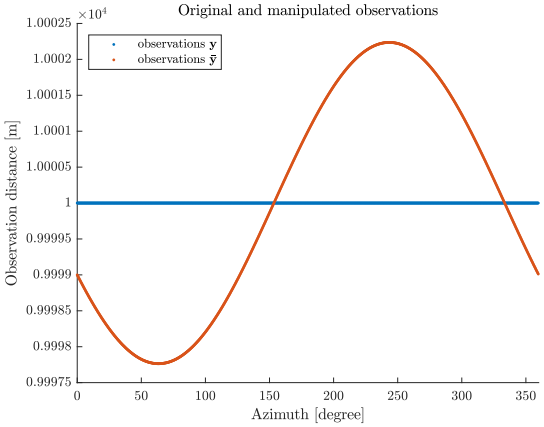
(a) Both the original and the manipulated observations, $\mathbf{y}$ and $\bar{\mathbf{y}}$ respectively.



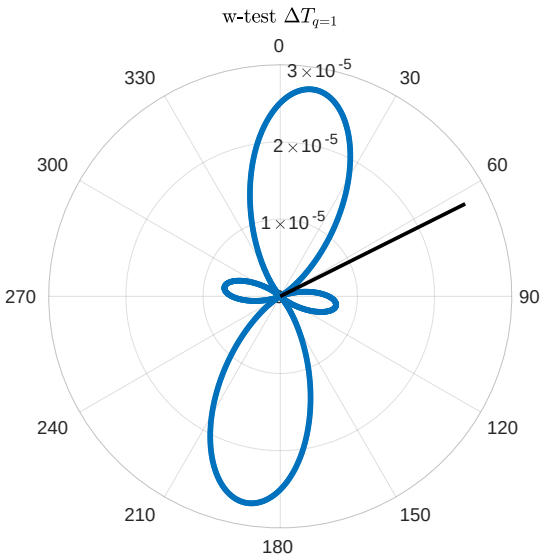
(b) The absolute difference of the w-test based on the original and the manipulated solution. The direction of the manipulation vector $\tilde{\mathbf{x}}$ is shown with the balck line, magnitude not to scale.

Figure 4.5: The observations and the manipulated observations for the new simulation with $m = 1080$ points equally distributed around $(0, 0)$ with a distance of 1000 meter. The manipulation is $\bar{\mathbf{x}} = [1, 2]^T$.

Before it was assumed that the first linearisation is done at the true location $\mathbf{x}$. It is also possible that $\mathbf{x}_0$ is at another position. When the direction from $\mathbf{x} \rightarrow \mathbf{x}_0$ is known then it is still possible that a part of the design matrix does not change. For the next case the manipulation direction is not equal to the direction from $\mathbf{x}$ to $\mathbf{x}_0$. In this case the initial position $\mathbf{x}_0 = [0.5, 1]^T$.



(a) Original $\mathbf{y}$ and manipulated $\bar{\mathbf{y}}$ observations for $m = 1080$ known points.

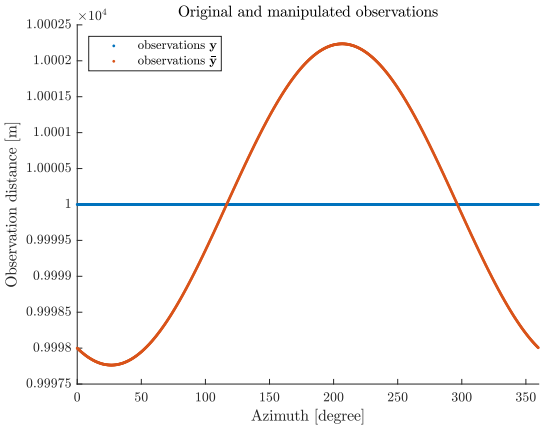


(b) The absolute difference of the w-test between the original and the manipulated solution. The direction of the manipulation vecotr is shown with a black line, magnitude is not to scale.

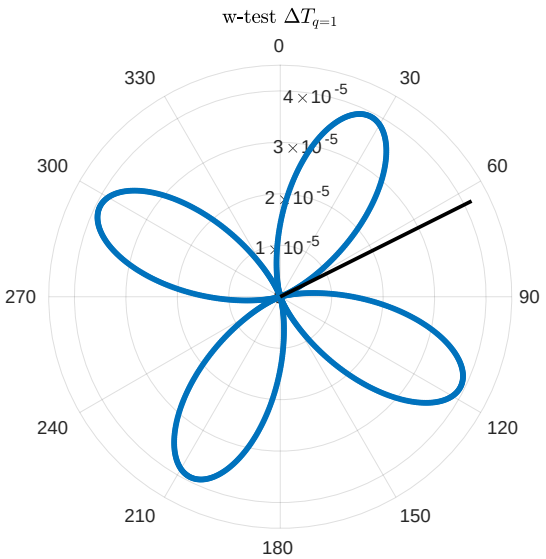
Figure 4.6: A measurement setup with 1080 known points equally distributed around (0,0). The initial linearisation position $\mathbf{x}_0 = [0.5, 1]^T$ and the manipulation vector is $\tilde{\mathbf{x}} = [1, 2]^T$.

Figure 4.6 shows the result when a different initial guess is used. This shows that the largest difference is not longer perpendicular to the direction of the manipulation vector. Exactly in the direction of the manipulation vector the difference is still zero.

The next step is to include the scaled clock bias $c\delta t$ in meters. This also changes the design matrix A, since there is an extra column filled with ones. The effect is that all observations contribute equally to the estimation of the scaled clock bias.



(a) Original $\mathbf{y}$ and manipulated $\bar{\mathbf{y}}$ observations for $m = 1080$ known points.

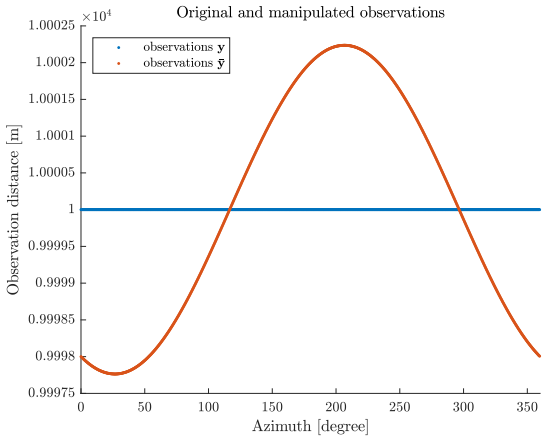


(b) The absolute difference of the w-test between the orgianl and the manipulated solution. The direction of the manipulation vector is shown with a black line, magnitude is not to scale.

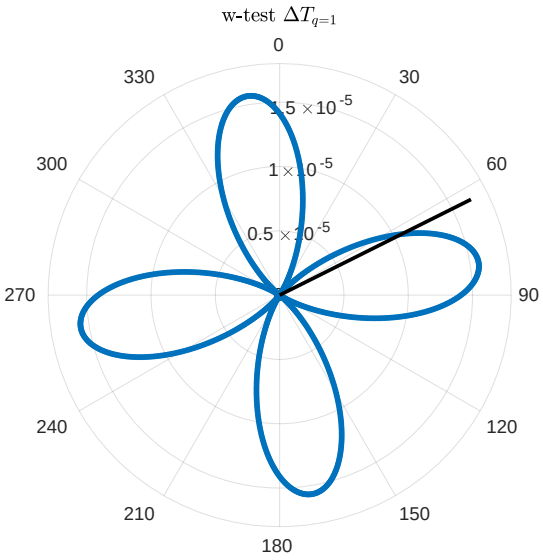
Figure 4.7: A measurement setup with 1080 known points equally distributed around $(0, 0)$. The scaled clock bias is also estimated in meters.

What can be seen in Figure 4.7 is that the estimation of the receiver clock bias has influence on the final difference of the results of the w-test. Before the largest difference was almost perpendicular to the manipulation vector. With the influence of the clock estimation a star pattern is formed. Now both in the direction of the manipulation vector and perpendicular to the manipulation vector the difference in the w-test is zero.

Next the clock bias estimation is combined with a new initial position $\mathbf{x}_0 = [0.5, 1]^T$. The result is shown in Figure 4.6.



(a) Original $\mathbf{y}$ and manipulated $\bar{\mathbf{y}}$ observations for $m = 1080$ known points.



(b) The absolute difference of the w-test between the origianl and the manipulated solution. The direction of the manipulation vector is shown with a black line, magnitude is not to scale.

Figure 4.8: A measurement setup with 1080 known points equally distributed around $(0, 0)$. The scaled clock bias is also estimated in meters. The initial guess $\mathbf{x}_0 = [0.5, 1, 0]^T$ and the manipulation vector is $\tilde{\mathbf{x}} = [2, 1, 0]^T$.

What can be seen in Figure 4.8 is that the influence of another initial guess is that the star pattern is rotated. The direction of the manipulation vector does not longer mean that the difference of the w-test is zero.

For the last example noise is added to the observations $\mathbf{y}$. In this case it is a combination of a manipulation of $\tilde{\mathbf{x}} = [1, 2, 0]^T$, a initial guess of $\mathbf{x}_0 = [0.5, 1, 0]$ and noise in the measurements. This is shown in Figure 4.9.

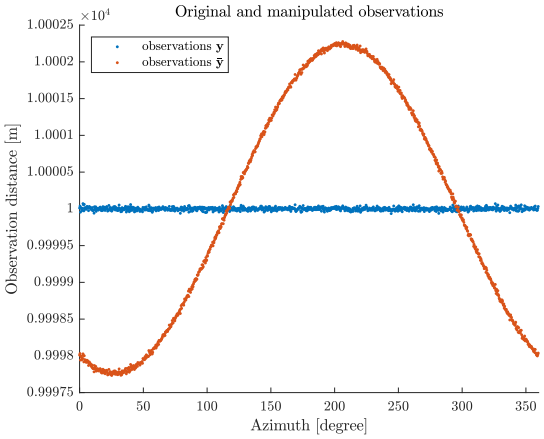
Figure 4.9 shows the correlation of the w-test differences with noisy observations. As a result the difference of the w-test is still correlated with the direction of the manipulation vector.

This shows that there are small differences between the w-test based on the original solution and the w-test based on the manipulated solution. The main reason for this difference is the non-linear model that is used for the position estimation.

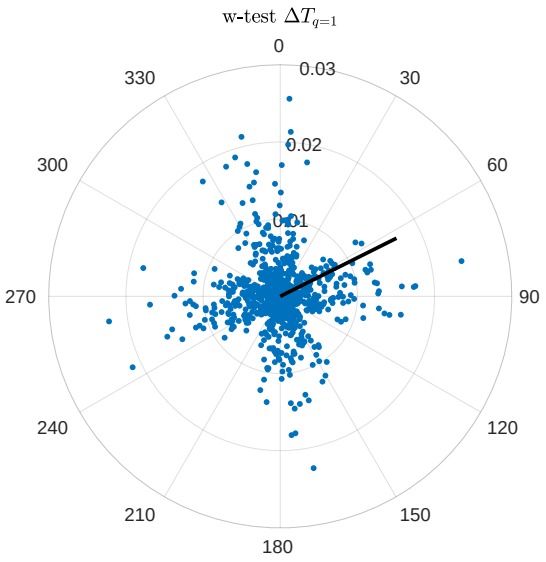
It is shown that it is possible to create an undetectable bias for a linear system. For a non-linear model is it also possible to use the parameter estimation theory to determine an undetectable bias for the observations. Due to the Gauss-Newton iteration the introduced bias contains both an influential and testable bias. The influential bias is still larger than the testable bias, and in the shown examples the manipulation was not detected.

4.4. Differential model

So far a linear model is used where all data are manipulated, the examples concern absolute stand-alone positioning. In a differential setup there are two stations, the user receiver and the reference station. When the goal is to manipulate only the augmentation data, this means that there are observations which are not altered. In this case the observations of the user receiver are not manipulated,



(a) Original $\mathbf{y}$ and manipulated $\bar{\mathbf{y}}$ observations for $m = 1080$ known points.



(b) The absolute difference of the w-test between the origianl and the manipulated solution. The direction of the manipulation vector is shown with a black line, magnitude is not to scale.

Figure 4.9: The measurement setup with 1080 known points, equally distributed around (0,0), is used to calculate the w-test and the influence of the manipulation. The noise is simulated using a normal distribution with a zero mean and a standard deviation of 0.02 meter.

but only those from the reference station.

First the situation can be described as shown in Equation (4.22). Note that this is in principle the same model as shown for differential GNSS in Equation (2.27).

$$\mathrm{E}\left(\begin{bmatrix}\underline{\mathbf{y}}_1\\ \underline{\mathbf{y}}_2\end{bmatrix}\right) = \begin{bmatrix}A_1 & & C\\ & A_2 & C\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_c\end{bmatrix} \tag{4.22}$$

The two different stations are shown with subscript $\dot{()}_1$ and $\dot{()}_2$. Each station has its own parameters, like the position and a clock offset, represented by $\mathbf{x}_1$ and $\mathbf{x}_2$. Both stations also have shared parameters like the clock biases of the satellites, represented by $\mathbf{x}_c$.

The model that can be used for RTK is a differential model, which is shown in Equation (4.23). This is the same model used by GNSS, see Equation (2.31).

$$\mathrm{E}\left(\left[\underline{\mathbf{y}}_2 - \underline{\mathbf{y}}_1\right]\right) = \left[A_{1,2}\right]\left[\mathbf{x}_2\right] \tag{4.23}$$

When manipulating the observations for this model actually only one of the two observations is manipulated based on the design matrix, in this case A_2. When only this differenced model is used then the whole $\mathbf{y}_{1,2}$ vector is manipulated through $\mathbf{y}_1$. If no other model is used then it is hard to detect a bias which is mostly in the range space of A. It is however possible to use the observations $\mathbf{y}_2$ which are manipulated through an cyber attack on the augmentation data.

For differential positioning not only the observations are used, but also the known position of the reference station. This makes it also possible for the receiver to use only part of the data to check if the data are valid. One method is to calculate the reference station position and compare this with the known position, see Equation (4.24).

$$\mathrm{E}\left(\left[\underline{\mathbf{y}}_1\right]\right) = \left[A_1\right]\left[\mathbf{x}_1\right] \tag{4.24}$$

Another method is to use the trusted measurements of the receiver, $\underline{\mathbf{y}}_2$, to calculate a stand-alone position and compare this with the differential position Equation (4.25).

$$\mathrm{E}(\left[\underline{\mathbf{y}}_2\right]) = \left[A_2\right]\left[\mathbf{x}_2\right] \tag{4.25}$$

The differential position is more precise than a stand-alone position. The stand-alone solution based on the protected observations ($\mathbf{y}_2$) should be used to compare the two results, taking into account this difference in precision.

5

Implementation of GNSS augmentation data manipulation

This chapter describes the implementation of the GNSS augmentation data manipulation. The manipulation is based on an assumed SPP model, since the exact model of the target is unknown. In this chapter it is determined if the assumed model is by approximation equal to the model that is used by the target. After that it is tested what the effect is when the model is based on a wrong initial location, the linearisation error. The size of the manipulation is limited by the SPP solution of the user based on the manipulated observations. Based on a dataset of 24 hours the accuracy of the SPP solution is calculated.

5.1. The manipulation procedure

Below in Figure 5.1 a flowchart of the manipulation procedure is shown.

The input that is different for each manipulation is the manipulation in parameter space and the initial guess for the reference station. The data of the reference station are available and used to create the model or the design matrix A. A RTK model uses the receiver position for the design matrix, which means that another model is used for the manipulation. The reason is that the position of the receiver is unknown when traditional RTK is used.

For each epoch the new A matrix is determined. To determine the A matrix the satellite positions are calculated based on the observations of the reference station. Based on the satellite positions and the position of the reference station the unit direction vectors are calculated, which are used in the design matrix A.

After the iteration is finished, the A matrix is defined for that epoch. Based on the A matrix the manipulation for the observations is calculated based on the method as described before, $\tilde{\mathbf{y}} = A\tilde{\mathbf{x}}$. This manipulation is added to the original epoch observations $\bar{\mathbf{y}} = \mathbf{y} + \tilde{\mathbf{y}}$.

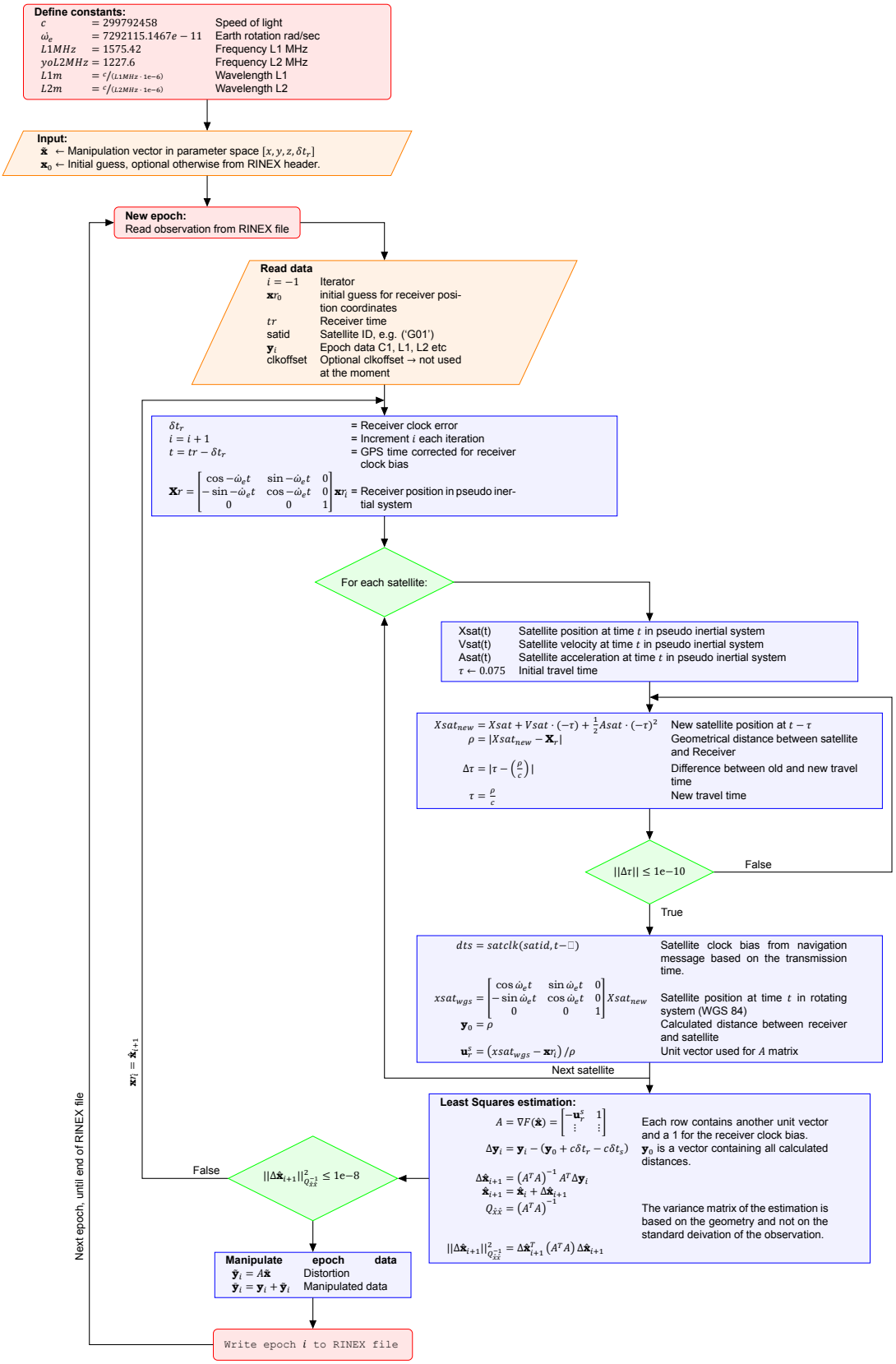


Figure 5.1: Augmentation data spoofing procedure

5.2. Linearisation position

The linearisation point, the initial guess $\mathbf{x}_0$, is used to determine the design matrix A. Ideally the same model, and thus design matrix, will be used by both the attacker and the user. To calculate the design matrix, A, the user position is used. Part of the design matrix are the unit direction vector between the receiver and the satellites. The attacker does not now the position of the user is, and it is possible that the user is moving. Often the position of the reference station is sent in the RTCM message. If this is not the case it is possible to use the augmentation data to estimate the reference station position based on a SPP solution. The attack has thus knowledge about the position of the reference station.

In this study the position of the reference station is used to calculate the design matrix and the observation manipulation vector. The attacker uses another point of the linearisation, and thus the model that is used by the attacker is different than the model that is used by the user. The impact of the difference in the linearisation position it can be defined as the contribution to the overall model test.

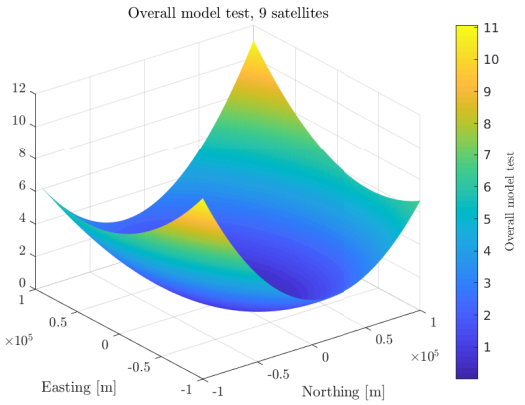
To test this setup only the observation manipulation vector is used, $\tilde{\mathbf{y}}$, since that vector is the impact on the

One epoch of the reference station is used to define the A matrix on a known position. Only the observation manipulation vector is considered at the moment, so it is clear what the contribution of this vector is to the overall model test. Before it is proofed that the contribution to the testable bias in a linear model is zero, see Equation (4.19). In this case the design matrix used by the hacker, B, is not equal to the design matrix that is used for the manipulation, A. In this case the part of the observation manipulation vector $\tilde{\mathbf{y}} = B\tilde{\mathbf{x}}$ that is not in the column space of A contributes to the testable bias. The calculation of the residual $\tilde{\hat{\mathbf{x}}}$ based on the manipulated observations $\bar{\mathbf{y}}$ is shown in Equation (5.1).

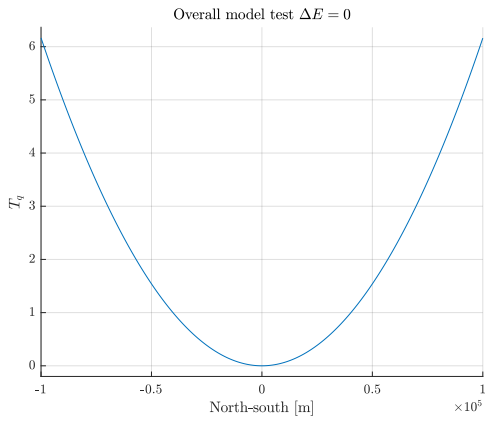
$$
\begin{aligned}
\hat{\bar{\mathbf{e}}} &= \bar{\mathbf{y}} - A\hat{\bar{\mathbf{x}}} \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \bar{\mathbf{y}} \\
&= \bar{\mathbf{y}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} (\mathbf{y} + B\tilde{\mathbf{x}}) \\
&= (\mathbf{y} + B\tilde{\mathbf{x}}) - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} (\mathbf{y} + B\tilde{\mathbf{x}}) \\
&= \underbrace{\mathbf{y} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} \mathbf{y}}_{\text{Unmanipulated BLUE residual}} + \underbrace{B\tilde{\mathbf{x}} - A\left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} B\tilde{\mathbf{x}}}_{\text{Manipulated contribution to the residual}}
\end{aligned}
\tag{5.1}
$$

The unmanipulated BLUE residual is not dependent on the manipulation, therefore only the last part, the manipulation contribution to the residuals, is considered. Note that when $B = A$, the residual as result of the manipulation is zero.

The linearisation position for B is changed with respect to the linearisation position of A. The position is changed with steps of 1000 meter, from -100 km till 100 km in both north–south and east–west direction. To know the contribution to the overall model test some parameters for the test have to be defined. The used epoch has 9 satellites, thus there are $m = 9$ observations. There are four unknowns, the position and the receiver clock bias. The degree of freedom is $df = 9 - 4 = 5$. The standard deviation of the observation is set to 0.003 meter, the observation precision of the carrier–phase measurement [20].



(a) Overall model test for one epoch. The design matrix (A) is changed in north and east direction.



(b) An cross section in north-south for the overall model test. The east-west offset was zero.

Figure 5.2: The overall model test for the manipulation vector based on different design matrices. The overall model test is for one epoch, for 9 observations. The critical value is $T_{q=9-4=5} = 11.1$. The maximum value for the overall model test is 3.99. The used standard deviation per observation is 0.003 meter.

Figure 5.2 shows the overall model test for the manipulation vector. An alpha level op 0.05 is used. The threshold value is $T_{q=5} = 11.0705$, and the maximum value for the overall model test is 11.0710. The used manipulation vector is $\tilde{\mathbf{x}}$. The model is rejected twice due to a wrong linearisation point. The manipulation vector is rejected at the corners of the possible linearisation positions. The location of the two points which where rejected is north–east and south–west, with a distance of 141.42km.

$$\begin{aligned} \hat{\mathbf{e}} &= B\tilde{\mathbf{x}} - A\left((A^T Q_{yy}^{-1} A) A^T Q_{yy}^{-1} B\tilde{\mathbf{x}}\right) \\ T_{q=5} &= \hat{\mathbf{e}}^T Q_{yy}^{-1} \hat{\mathbf{e}} \end{aligned} \tag{5.2}$$

Note that this shows only the part of the manipulation vector in the testable bias. The observation noise from the original observations, $\mathbf{y}$, is not taken into account.

Based on this result it can be concluded that it indeed does not matter if the used design matrix was exactly at the users position. In this case the distance between the receiver and the simulated reference station was 100km in all directions. Normally RTK is only used over distances up to 30km, therefore the bias introduced by a slightly different A matrix is minimal, therefore the base–station position can be used for the linearisation instead of using the actual user–receiver position.

5.3. Single point positioning accuracy assessment

The result of augmentation data spoofing is a manipulated DGNSS solution. The manipulation is in the observations of the reference station. The user also has its own, unaltered, observations. The user–receiver can check if there is a significant difference between the (unmanipulated) SPP solution and the DGNSS solution. In this section the general accuracy is analysed based on a data record of 24 hour.

Normally the 1 sigma accuracy of single point positioning is < 10 meter (1 sigma) [34, Table 21.7]. The empirical data for this case study shows something different. To determine the precision for SPP solutions using RTKlib another dataset is used. The data is collected for 24 hours, on February 19th 2020, with an interval of 30 seconds in Delft, the marker name is Delft–16. The used receiver is a TPS ODYSSEY_E, and the used antenna is TRM29659.00. In this case not only GPS is observed, but also GLONASS. This results in more observed satellites and thus more observations. The RTKlib software has as output option the standard deviation in meters, based on the geometry of the system.

Normally the accuracy of the system given in the specifications, i.e. the accuracy of SPP, a negative scenario is used so that the accuracy is reached even with less favourable conditions. In case of undetected data manipulation the most negative scenario is an accurate SPP solution. Therefore the precision, which is given as specification of the system, cannot be used because it should be assumed that often the system works better than the given specifications. The precision will be used to determine

the maximum length of the manipulation vector. To determine the precision a dataset of 24 hours is used. The value that will be used is the smallest standard deviation from the RTKlib solution. The standard deviation is dependent on the amount of satellites and the geometry of the system. The same constellation occurs each 11h 58m 02s (12h sidereal time), therefore all possible constellations are observed in the 24 hour observation record. The minimum calculated standard deviation from RTKlib is then used as a value for the precision of the SPP solution.

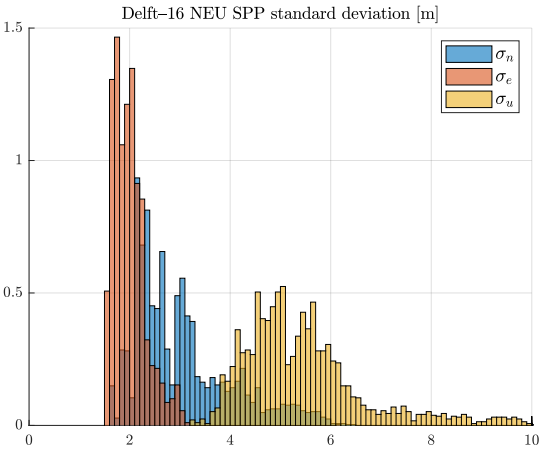
Below in Table 5.1 the standard deviations of the RTKlib solution is shown. The standard deviation is given in meters. The same data is used for the Figure 5.3.

	Mean	Median	Standard deviation	Minimal	Maximal
σ_x	4.0212 m	3.5953 m	1.2849 m	2.3962 m	9.5371 m
σ_y	2.0156 m	1.9930 m	0.3272 m	1.5486 m	6.1370 m
σ_z	4.7871 m	4.4751 m	1.3456 m	2.6614 m	31.2213 m
σ_n	3.0811 m	2.7470 m	1.1733 m	1.6134 m	20.4951 m
σ_e	1.9979 m	1.9640 m	0.3266 m	1.5332 m	6.1582 m
σ_u	5.4770 m	5.2163 m	1.3238 m	3.2091 m	25.0471 m

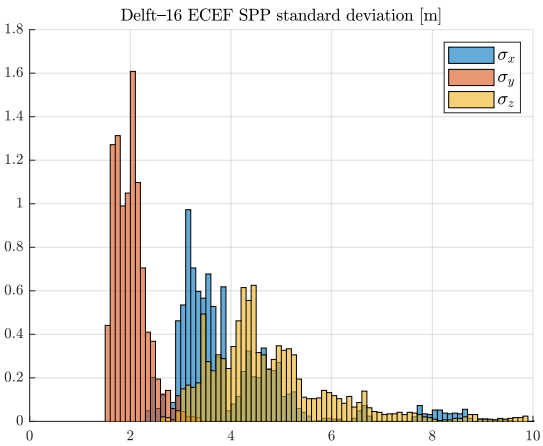
Table 5.1: Covariance matrix from 24 hour observation, 30 seconds interval, marker point Delft–16. The Single-point positioning (SPP) solution is calculated in Earth Centerd Earth Fixed (ECEF) and in NEU (North, East, Up).

It is possible to calculate an ellipsoid, based on a certain confidence level, per epoch based on the geometry of the system. This geometry is dependent on the positions of the satellites and the receiver. This can be of interest when it is desired to get the maximum length of the manipulation vector. However in this case the covariance is not important, because the orientation of the ellipsoid changes over time. It is chosen to use the minimal standard deviation for the ECEF system. For the solution in NEU the standard deviation for the horizontal (North, East) is set to the minimal standard deviation of the horizontal components, the standard deviation of the Up component is treated separately. In general it is known that the standard deviation of the Up component is roughly twice as large as the horizontal standard deviation. This is a result of the satellites which can be “around” the receiver in a local horizontal plane, but this is impossible in the vertical direction due to the Earth’s surface. For a solution in ECEF it is also true that one of the axes of the confidence ellipsoid is twice as large as the other axes, but it is not certain where those axes are since it is dependent on the position of the satellites.

In figure Figure 5.3 the histogram of the calculated standard deviation, based on the geometry of the satellites, is shown. For the NEU results the minimum standard deviation is for $\sqrt{\sigma_n}$ and $\sqrt{\sigma_e}$. This minimal standard deviation is 1.5m. The minimal standard deviation in Up direction is 3.2m. As said for the ECEF solution only one value for the a-priori standard deviation will be used. The minimal standard deviation is also 1.5m. It is assumed that the user uses a significance level of 0.05 for testing. This means that by approximation the elements of the manipulation vector cannot be larger than $1.96\sigma \approx 2.9$m in horizontal direction for NEU and in all directions for ECEF.



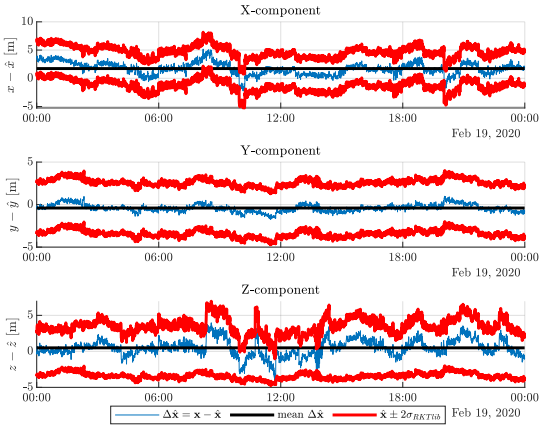
(a) Standard deviation in North East Up.



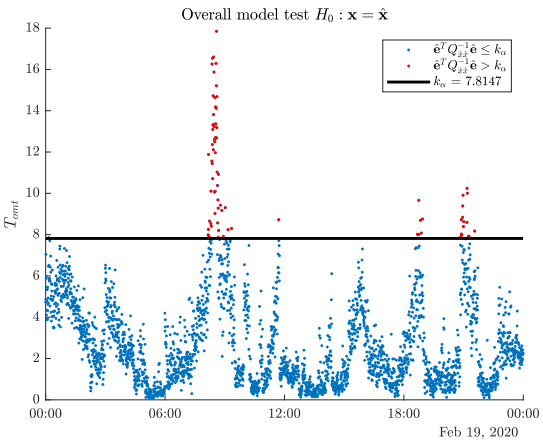
(b) Standard deviation in Earth Centerd Earth Fixed (ECEF).

Figure 5.3: Histogram of the standard deviation in NEU (left) and in ECEF (right).

The defined precision is based on the RTKlib output. It is also possible to analyse the SPP solution compared to a known point. The result is shown in Figure 5.4.



(a) Difference between known position and SPP solution, $\Delta\hat{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$



(b) The overall model test with an alpha level of 0.05. Degree of freedom is 3.

Figure 5.4: The SPP solution $\hat{\mathbf{x}}$ compared with the known position $\mathbf{x}$. On the left the difference between the SPP solution and the known position, $\Delta\hat{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$, is shown. On the right the overall model test is shown, based on the hypothesis $H_0 : \mathbf{x} = \hat{\mathbf{x}}$. A standard deviation of 1.5 meter is used, corresponding to the RTKlib value.

The data is not manipulated. Figure 5.4a shows the difference in position between the known position and the SPP solution, the confidence interval is shown in red, with a confidence level of 0.95. What can be seen is that the true position is not always inside this confidence interval.

The overall model test, shown in Figure 5.4b, shows that at multiple epochs the SPP solution is rejected. 2.67% of the observations are rejected according the overall model test with an alpha level of 0.05 and based on the standard deviation from the RTKlib output.

A value of 1.5 meter for the standard deviation seems correct, based on the actual difference between the SPP solution and the known position.

6

GNSS manipulation results

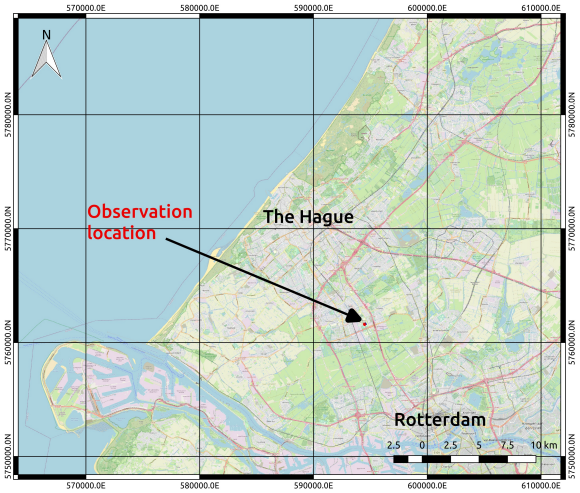
In this chapter the results of the manipulation of GNSS augmentation data, based on post processing, are discussed. At the start the SPP solution for the base station is shown. After the SPP solution, both differential, code and phase, timing solutions are discussed. At the end an example is shown for data that are measured on a highway near Delft, the A13. The data are processed using RTKlib [7] with different configurations.

The dataset that is used to analyse the different GNSS modes is collected at the TU Delft campus. The used receiver was a Trimble R7 with a Trimble Zephyr Geodetic (TRM41249) antenna. These data are used for short baseline Real Time Kinematic (RTK) measurements. The duration of the experiment is approximately 45 minutes. The maximum distance between the reference station and the rover is 100 meter. Only GPS satellites are observed.

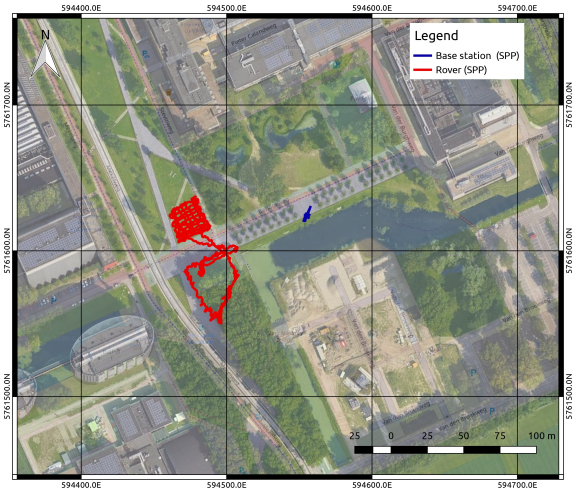
The base station position is known:

```
Base station position ECEF ETRS89
X = 3923753.203 m
Y =  300351.476 m
Z = 5002645.104 m
```

The observations are done in Delft, see Figure 6.1.



(a) Observation location at the TU Delft campus. Underlying map is from openstreetmap in UTM31N (EPSG:32631).



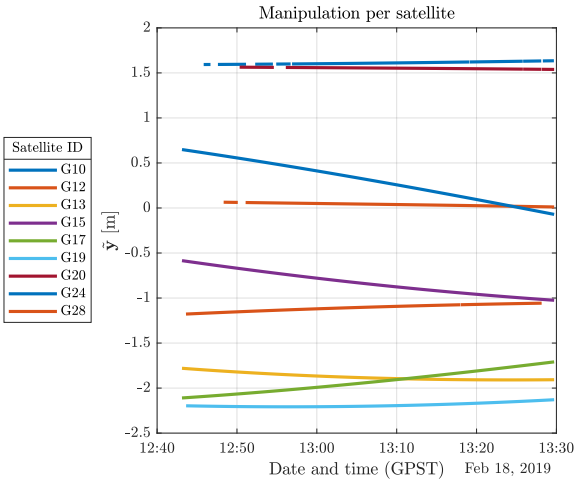
(b) Detailed overview observation location. The base station is shown in blue and the rover is shown in red. Underlying satellite image is from Google Maps.

Figure 6.1: Location of base station and rover in Delft.

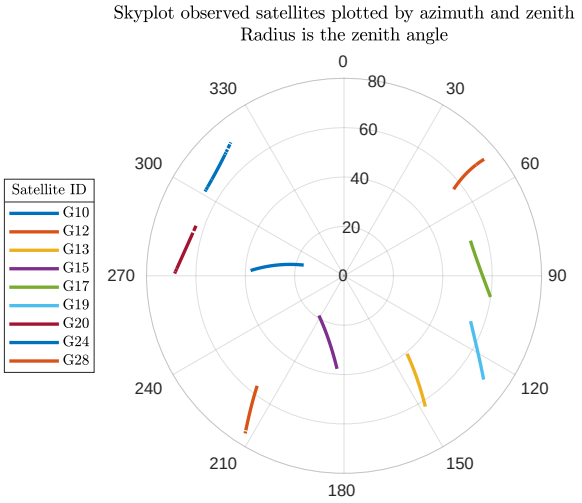
6.1. Base station manipulation

In this study post processing software are used. For SPP there are two data files, the observations and the navigation file. For relative positioning, like DGNSS and RTK, the observations of the base station are also used. Only the observations at the base station can be manipulated with a MITM attack. The data are treated epoch per epoch, so no information about a next epoch is used. This simulates a real time attack.

The same data are used for the different GNSS modes. The manipulation is done only on the RINEX file of the base station. Different manipulation vectors can be used, as long as the length of the vector is not larger than 2.9 meter (based on a significance level of 0.05). The manipulation per observation is calculated and added to the original observation. For a manipulation vector of $\tilde{\mathbf{x}} = [1, 2, 0, 0]$ the manipulation per satellite is shown in Figure 6.2a and Figure 6.2b shows the satellite positions in a skyplot.



(a) The manipulation per observation as function of time epoch, for a manipulation vector $\tilde{\mathbf{x}} = [1, 2, 0, 0]^T$. At most there are 9 satellites observed at a time.



(b) The observed satellites shown based on the azimuth and the zenith angle. The zenith angle is shown in radial direction, a zenith angle of ≈ 90° means that the satellite is on the horizon.

Figure 6.2: The manipulation per satellite ($\tilde{\mathbf{y}}$) and a skyplot of all observed satellites.

What can be seen is that a manipulation vector with a length of 2.24 meter $\left(|\tilde{\mathbf{x}}| = \sqrt{2^2 + 1^2 + 0^2 + 0^2}\right)$ in parameter space translates into the manipulations in observation space in the same order of length, in this case between -2.5 till 2 meter. Due to the geometry of the system some observations are increased and some are decreased by the manipulation. The manipulation per satellite is not constant because the satellite's position changes over time.

For one epoch the manipulation is given:

```
Epoch: 19  2 18 13  0  0.0000000  0  9G10G12G13G15G17G19G20G24G28
Manipulation per satellite:
G10    1.601966e+00 m
G12    5.003856e-02 m
G13   -1.866024e+00 m
G15   -7.794110e-01 m
G17   -1.991660e+00 m
G19   -2.206903e+00 m
G20    1.558494e+00 m
G24    4.118419e-01 m
G28   -1.119744e+00 m
```

This manipulation vector is added to the existing observations in the RINEX file. In this case the observed GPS satellite has four observations of interest, namely two code observations (C1 and P2) and the carrier wave observations (L1 and L2). The location of the observations are given in the header of the RINEX file. In this case the order is: C1, L1, L2, P2, S1 and S2. The last two observation types (S1 and S2) are the Signal-to-Noise-Ratio's and not of interest for this manipulation.

```
Header line with observation types:
     6    C1    L1    L2    P2    S1    S2                  # / TYPES OF OBSERV

ORIGINAL EPOCH
 19  2 18 13  0  0.0000000  0  9G10G12G13G15G17G19G20G24G28
  23581297.484 5  -1045387.234 5   -295079.03546  23581298.33646        39.250
        20.000
  23567281.945 6  -3176142.285 6  -2179662.91846  23567280.88746        46.000
        22.000
  21955963.977 6    808378.809 6    625828.94947  21955960.59047        46.250
        28.750
  20791723.766 7   -808936.648 7   -623881.91048  20791721.62948        50.250
        36.750
  22798695.242 6  -3437140.367 6  -2652857.30947  22798694.12147        45.000
        28.500
  23192741.891 5  -5251577.629 5  -4033258.58246  23192738.36746        40.750
        25.500
  22914137.469 1   -337175.656 1                                        33.000

  20758533.195 7  -3903065.660 7  -3014574.02048  20758533.29348        51.250
        38.750
  23388364.438 5    867734.473 5    672489.80146  23388362.36346        41.500
        23.000

MANIPULATED EPOCH
 19  2 18 13  0  0.0000000  0  9G10G12G13G15G17G19G20G24G28
  23581299.086 5  -1045378.816 5   -295072.47546  23581299.93846        39.250
        20.000
  23567281.995 6  -3176142.022 6  -2179662.71346  23567280.93746        46.000
        22.000
  21955962.111 6    808369.003 6    625821.30847  21955958.72447        46.250
        28.750
  20791722.987 7   -808940.744 7   -623885.10248  20791720.85048        50.250
        36.750
  22798693.250 6  -3437150.833 6  -2652865.46547  22798692.12947        45.000
        28.500
  23192739.684 5  -5251589.226 5  -4033267.61946  23192736.16046        40.750
        25.500
  22914139.027 1   -337167.466 1                                        33.000

  20758533.607 7  -3903063.496 7  -3014572.33448  20758533.70548        51.250
        38.750
  23388363.318 5    867728.589 5    672485.21646  23388361.24346        41.500
        23.000

       DIFFERENCE ORIGINAL - MANIPULATED (only C1 L1 L2 P2)
 19  2 18 13  0  0.0000000  0  9G10G12G13G15G17G19G20G24G28
       -1.602 5        -8.418 5        -6.56046        -1.60246
       -0.050 6        -0.263 6        -0.20546        -0.05046
        1.866 6         9.806 6         7.64147         1.86647
        0.779 7         4.096 7         3.19248         0.77948
        1.992 6        10.466 6         8.15647         1.99247
        2.207 5        11.597 5         9.03746         2.20746
       -1.558 1        -8.190 1
       -0.412 7        -2.164 7        -1.68648        -0.41248
        1.120 5         5.884 5         4.58546         1.12046
```

Above the manipulation of the RINEX observation file is shown. The observations units of L1 and L2 are in cycles of the carrier signal. When the carrier phase exceeds the fixed format, the observation has to be clipped into the valid interval. This should be considered when the carrier phase observation is altered.

The manipulation for both pseudorange measurements (C1 and P1) is the same since both are given in meters; $\tilde{\mathbf{y}} = \tilde{\mathbf{y}}_{C1} = \tilde{\mathbf{y}}_{P2}$. The manipulation for the carrier phase measurement is scaled with the wavelength; $\tilde{\mathbf{y}}_{L1} = \frac{\tilde{\mathbf{y}}}{\lambda_{L1}}$ and $\tilde{\mathbf{y}}_{L2} = \frac{\tilde{\mathbf{y}}}{\lambda_{L2}}$

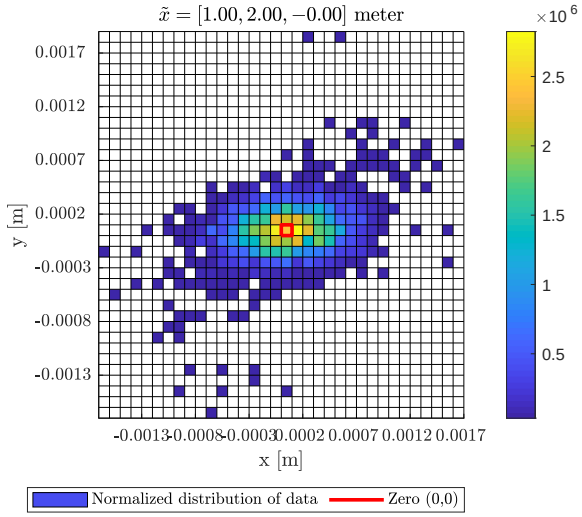
The original and manipulated data of the base station are used for the three different positioning modes. Different manipulation vectors are considered, but the manipulation $\tilde{\mathbf{x}} = [1, 2, 0, 0]$ is used for most of the analyses. The reason is that the size of the vector is still within limits of the SPP solution accuracy, but it can be significant for applications which require high precision positioning. A smaller bias, in a extreme case in order of millimetres, is less likely to be detected but it is also less significant for the user and therefore a less interesting case.

For the manipulation the known position of the base station is used. During a MITM attack it is likely that the position of the base station is also sent in the RTCM message, but if this is not the case a SPP is also a good indicator about the position of the base station, especially after multiple epochs (assuming that the base station is at a fixed location).

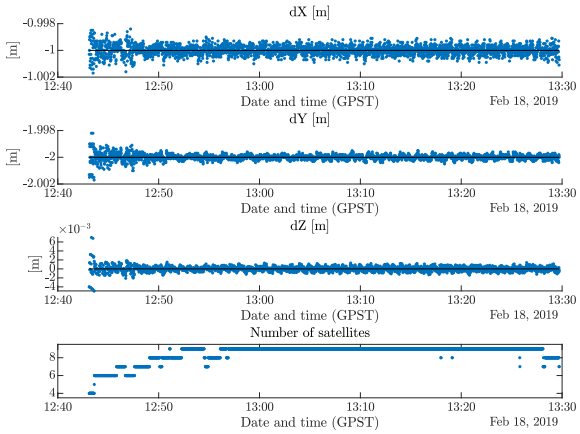
6.2. Single point positioning

The single point solution of the reference station is produced with RTKlib. The configuration file can be found in section B.1. The base station is stationary.

In Figure 6.3 the Single-point positioning (SPP) solution is shown. The expected value when the manipulated data are subtracted from the original data are the negative manipulation vector, $\mathrm{E}(\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) = -\tilde{\mathbf{x}}$. In 6.3a the known manipulation vector is subtracted from the estimated manipulation vector, i.e. $\tilde{\mathbf{x}} + \mathrm{E}(\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) = 0$. The colour scale is such that the histogram is an empirical a Probability Density Function. It can be seen that most data are at point zero (0,0). This is as expected. In theory all values should equal zero, but this is not the case. The observation noise is eliminated since it is assumed that the residuals remain the same before and after manipulation. The reason of the spread, less than 2 millimetre, can be a result of the rounding off errors in the RINEX file. The observations and the manipulation are real numbers, but in the RINEX file the maximum number of decimals is three.



(a) Histogram of data manipulation. The positions resulting from manipulated data are subtracted from positions resulting from orignial data, and the known manipulation is subtracted ($\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}} - \tilde{\mathbf{x}}$). The values are calculated as an empirical probability density function; $\frac{n_i}{N \cdot w_i}$. Where n_i is the number of sampels in bin i, N is the total number of samples and $w_i = dx \cdot dy$ is the bin width.



(b) Difference between the original data and manipulated data. The top three subplots show the difference in X, Y and Z. The last subplot shows the amount satellites that is being observed.

Figure 6.3: Difference between position based on the manipulated and the original RINEX file, ($\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}$). On the left a histogram is shown with the differences in X and Y. The expectation value of the difference between the solutions based on the original and the manipulated data should be the manipulation vector.

Other manipulation vectors are also used and shown in Table 6.1. The effect resulting from the manipulation is almost exactly the same as the input manipulation vector. Different sizes of manipulation vectors in the same direction are used. The same direction is used so that the geometry is equal between different manipulations. This makes it possible to analyse the effect of different manipulation sizes. Further one perpendicular vector ($[-2, 1, 0, 0]$) is used to show that the errors in the resulting manipulation is not direction dependent. The last vector is the one in z direction. This shows that a manipulation in one direction also works fine.

$\tilde{\mathbf{x}}$				$\hat{\mathbf{x}}$				$\frac{\hat{\mathbf{x}}}{\lvert\hat{\mathbf{x}}\rvert}$				$\lvert\tilde{\mathbf{x}}\rvert$	$\lvert\hat{\mathbf{x}}\rvert$
x	y	z	δt_r	x	y	z	δt_r	x	y	z	δt_r		
0.1000	0.2000	0	0	−0.1000	−0.2000	0.0000	0.0001	−0.4471	−0.8945	0.0002	0.0006	0.2236	0.2236
1	2	0	0	−1.0000	−2.0000	0.0000	0.0000	−0.4472	−0.8944	0.0000	0.0000	2.2361	2.2361
10	20	0	0	−10.0000	−20.0000	−0.0000	0.0000	−0.4472	−0.8944	−0.0000	0.0000	22.3607	22.3607
100	200	0	0	−100.0005	−199.9997	−0.0009	0.0007	−0.4472	−0.8944	−0.0000	0.0000	223.6068	223.6068
−2	1	0	0	2.0000	−1.0000	0.0000	−0.0000	0.8944	−0.4472	0.0000	−0.0000	2.2361	2.2361
0	0	2	0	−0.0000	−0.0000	−2.0000	0.0000	−0.0000	−0.0000	−1.0000	0.0000	2	2.0000

Table 6.1: SPP manipulation with different input biases (first four columns). The coordinates (x, y and z) are given in meters. The clock biases are in nano seconds. After the input manipulation vector, the resulting biases are shown, next the normalized bias is shown. The normalized bias shows that there are no resulting biases as result of a larger manipulation. The last two columns shows the norm of the input bias and of the estimated bias.

It can thus be concluded that the Single-point positioning (SPP) manipulation is successful and that the behaviour of the manipulated observations is as expected.

6.3. D-GNSS

The same data as before are used for the DGNSS solution, though now the data of the user receiver are added. The same manipulation vectors are used as for the SPP solution. It was not possible to process the largest manipulation vector ($\tilde{\mathbf{x}} = [100, 200, 0, 0]$) in RTKlib. As a sanity check RTKlib apparently checks whether the observations of the base station correspond to the given base station position, at least within tens of meters. The used RTKlib configuration file is show in B.2.1.

For manipulation with differential positioning there are two options, either the position of the reference station is manipulated or the observations are manipulated so that the manipulated observations corresponds to another position than the known reference station position. The reference position is known by the user. Therefore it is assumed that when the RTCM message contains another, manipulated, position for the reference station, the software will notify the user. The other option left is thus to manipulate the observations as if they belong to another reference position. After the manipulation the observations should correspond to another position (near the actual base station position), so that the resulting baseline is from this fake position to the rover position. This effect is shown in Figure 6.4.

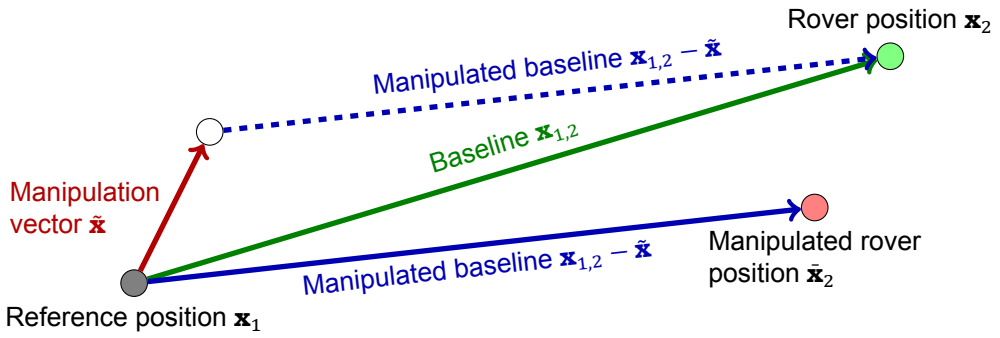


Figure 6.4: Effect of the augmentation data manipulation. The correct baseline ($\mathbf{x}_{1,2}$) is coloured green. The manipulation vector ($\tilde{\mathbf{x}}$) is coloured red, and has a north–east direction. The manipulated baseline ($\tilde{\mathbf{x}}_{1,2} = \mathbf{x}_{1,2} - \tilde{\mathbf{x}}$) is is coloured blue. The manipulated baseline is a baseline from a fake reference position, the white circle, to the rover. The manipulated baseline in combination with the reference position $\mathbf{x}_1$ result in a manipulated rover position ($\tilde{\mathbf{x}}_2$), the red filled circle. The manipulated rover positioning is manipulated in the opposite direction of the manipulation vector.

A scatter plot of the solution is shown in Figure 6.5. The used manipulation vector is $\tilde{\mathbf{x}} = [1, 2, 0, 0]$. The resulting position, based on the manipulated augmentation data from the base station, is in opposite direction of the manipulation vector.

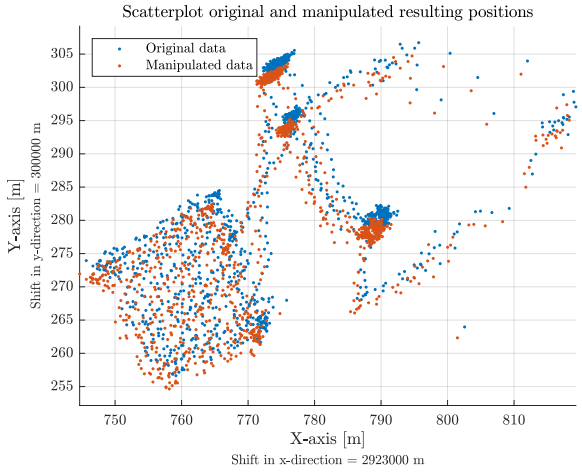
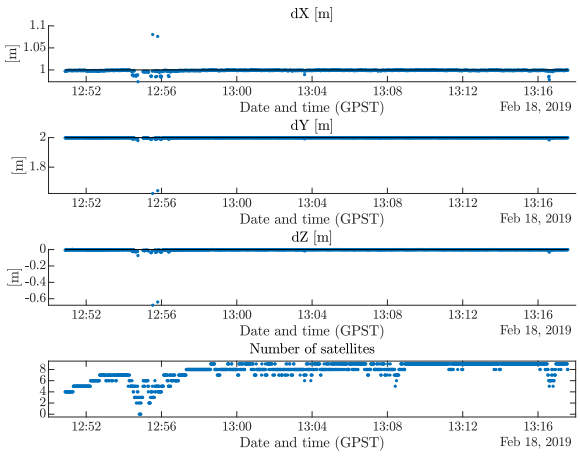
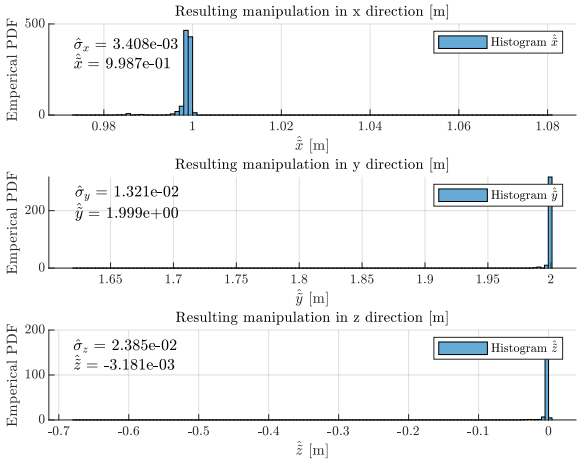


Figure 6.5: Scatter plot of the position of the user based on DGNSS-code positioning.

In Figure 6.6a the resulting manipulation vector per epoch is shown. In the other figure, Figure 6.6b, the histogram is shown of the same resulting manipulation. For every epoch the GNSS mode was equal for both the original and manipulated data. This means that there is no influence of the manipulation on the GNSS mode.



(a) Difference between the position resulting from the original and manipulated data, $\hat{\mathbf{x}} - \tilde{\hat{\mathbf{x}}}$. The upper three plots shows the difference in meters in x, y and z direcion respectively. The last plot shows the number of satellite for each epoch.



(b) Histogram of the resulting manipulation in x, y and z direction.

Figure 6.6: The estimated manipulation vector in two different plots

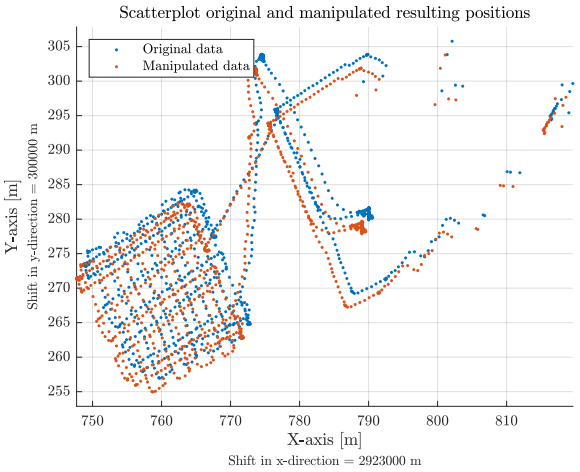
For DGNSS also some other manipulation vectors are used. The values can be found in Table 6.2. All manipulations are successful within the detection threshold.

$\tilde{\mathbf{x}}$				$\hat{\mathbf{x}}$				$\frac{\tilde{\hat{\mathbf{x}}}}{\lvert\tilde{\hat{\mathbf{x}}}\rvert}$				$\lvert\tilde{\mathbf{x}}\rvert$	$\lvert\tilde{\hat{\mathbf{x}}}\rvert$
x	y	z	δt_r	x	y	z	δt_r	x	y	z	δt_r		
0.1000	0.2000	0	0	0.0998	0.1993	−0.0015	0	0.4476	0.8942	−0.0066	0	0.2236	0.2229
1	2	0	0	0.9980	1.9934	−0.0144	0	0.4477	0.8942	−0.0065	0	2.2361	2.2294
10	20	0	0	9.9756	19.9308	−0.1558	0	0.4476	0.8942	−0.0070	0	22.3607	22.2884
0	0	2	0	−0.0069	−0.0109	1.9659	0	−0.0035	−0.0056	1.0000	0	2	1.9660
−2	1	0	0	−1.9909	0.9941	0.0014	0	−0.8947	0.4467	0.0006	0	2.2361	2.2252

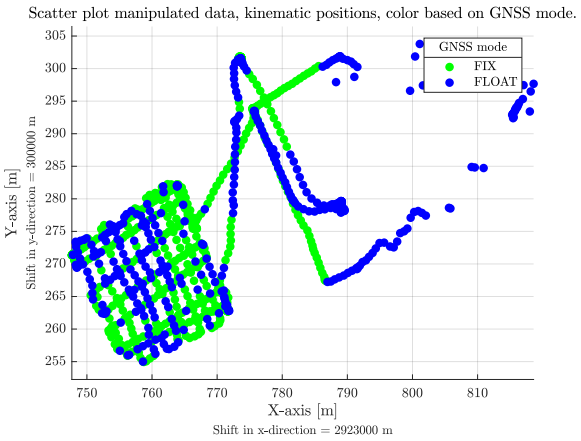
Table 6.2: DGNSS-code manipulation with different input biases (first four columns). The coordinates (x, y and z) are given in meters. The clock biases are described in nano seconds. After the input manipulation vector, the resulting biases are shown, next the normalized bias is shown. The last two columns shows the norm of the input bias and of the resulting bias.

6.4. RTK

The last position mode is RTK. The configuration file for RTKlib can be found in subsection B.2.2. The general scatter plot of the solution based on both the original and the manipulated data is shown in Figure 6.7a. Again a manipulation vector of $\tilde{\mathbf{x}} = [1, 2, 0, 0]$ is used. In Figure 6.7b the GNSS mode is shown.



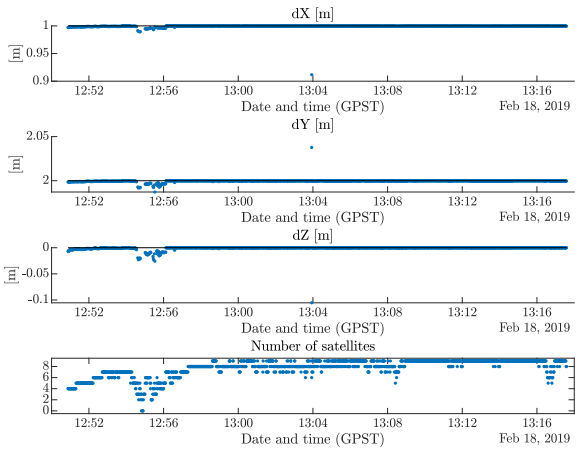
(a) Scatter plot of the rover.



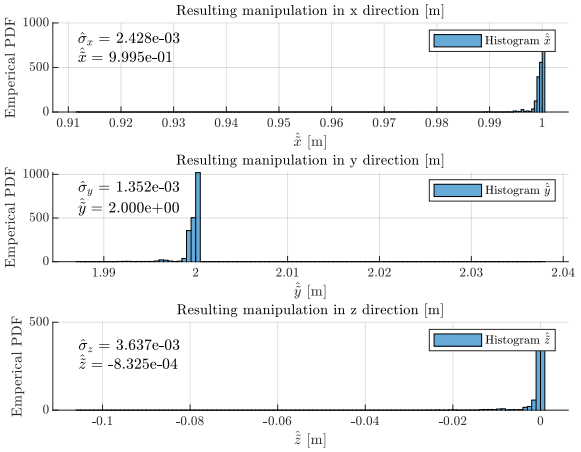
(b) Scatter plot of the rover. The colors indicate the the GNSS mode.

Figure 6.7: Scatter plots with the RTK solution.

The results of the manipulation of the augmentation data, are shown in Figure 6.8. This time the manipulation did change the GNSS mode, though only for one epoch. For the manipulation vector perpendicular to this manipulation vector, but with the same length; $\tilde{\mathbf{x}} = [-2, 1, 0, 0]$ there does not occur a single change in the GNSS mode. The influence of the manipulation on the final GNSS mode depends on the magnitude of the manipulation. For example a vector of length 0.2236 meter there is no difference in GNSS mode, but another vector with length 22.3607 meter has four times a different mode.



(a) Difference between the solution based on the original and the manipulated data, $\hat{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}$ is shown in the upper three plots. The upper three plots shows the difference in meters in x, y and z direcion respectively. The last plot shows the number of used satellites for each epoch.



(b) Histogram of the difference in x, y and z direction.

Figure 6.8: The resulting manipulation vector in two different plots.

Below in Table 6.3 the results for the RTK solution are shown. The table shows that the resulting

difference of the user position is equal to the input manipulation in the order of millimetres.

$\tilde{\mathbf{x}}$				$\hat{\mathbf{x}}$				$\frac{\hat{\mathbf{x}}}{\lvert\hat{\mathbf{x}}\rvert}$				$\lvert\tilde{\mathbf{x}}\rvert$	$\lvert\hat{\mathbf{x}}\rvert$
x	y	z	δt_r	x	y	z	δt_r	x	y	z	δt_r		
0.1000	0.2000	0	0	0.0999	0.1994	−0.0013	0	0.4477	0.8942	−0.0057	0	0.2236	0.2230
1	2	0	0	0.9988	1.9945	−0.0119	0	0.4478	0.8941	−0.0053	0	2.2361	2.2307
10	20	0	0	9.9757	19.9385	−0.1478	0	0.4474	0.8943	−0.0066	0	22.3607	22.2953
0	0	2	0	−0.0056	−0.0090	1.9718	0	−0.0028	0.0046	1.0000	0	2	1.9718
−2	1	0	0	−1.9918	0.9943	0.0012	0	−0.8947	0.4466	0.0005	0	2.2361	2.2262

Table 6.3: RTK augmentation data manipulation with different input biases (first four columns). The coordinates (x, y and z) are given in meters. The clock biases are described in nano seconds. After the input manipulation vector, the resulting biases are shown, next the normalized bias is shown. The last two columns shows the norm of the input bias and of the estimated bias.

6.4.1. Default RTKlib configuration

In the previous examples the configuration that is used in RTKlib was not the default configuration. The atmosphere models were not used. The addition of extra information or estimation for the atmosphere corrections does change the used model.This was done so that the simplified model that is used for the manipulation and the used model in RTKlib were more equal. Another option, the Receiver Autonomous Integrity Monitoring (RAIM) and Fault Detection and Exclusion (FDE) were not used. Those options are used to detect possible outliers in the observations. When the manipulation is successful, the manipulation is not detected by those function. Now it is time to see if the manipulation, based on a simplified model, also results in a successful manipulation of the RTKlib solution when the default options are used.

What can be seen is that the manipulation is still successful, see Figure 6.9.

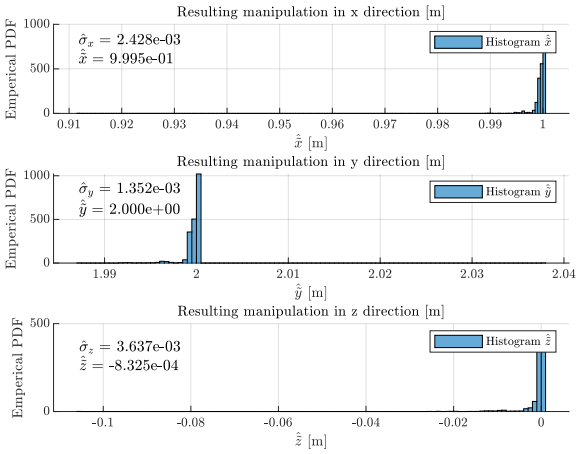


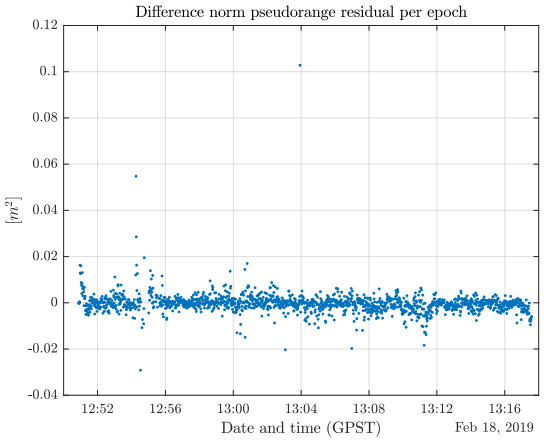
Figure 6.9: The RTKlib solution in RTK mode with the default configuration.

Another indicator of a successful manipulation are the residuals. The residuals of the least-square estimation are in the testable bias space. A successful manipulation is when the contribution of the manipulation is only in the influential bias, and not in the testable bias. Therefore, when the residuals are not significantly changed, the manipulation is successful.

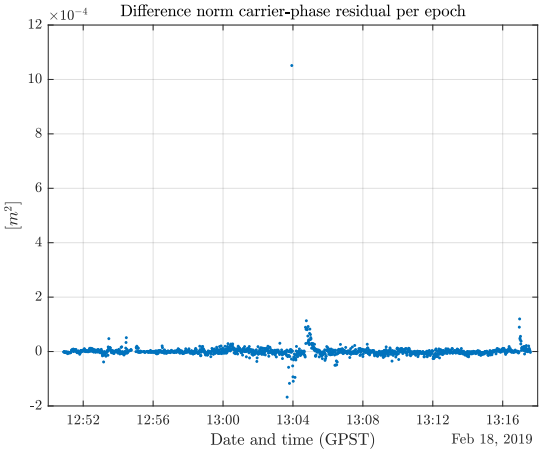
The mean residual for the pseudorange observation based on the original data is -0.0427 meter, with a standard deviation of 1.1252 meter. The average difference for the residual based on the original and the manipulated data is $-3.0533\text{e}{-4}$ meter, with a standard deviation of 0.0018 meter.

The mean residual for the carrier-phase observation based on the original data is $3.0842\text{e}{-4}$ meter, with a standard deviation of 0.0170 meter. The average difference for the residuals based on the original and the manipulated data is $-1.9482\text{e}{-4}$ with a standard deviation of $3.0535\text{e}{-4}$.

To compare the residuals per epoch, the sum of the squared residuals per epoch is used. The sum of the squared residuals based on the manipulations is subtracted from the sum of the squared residuals based on the original observations. This is done for both the pseudorange and the carrier-phase observation. The flaw in this comparison is that when there are more satellites used, thus $m >$ then the sum of the squared error is expected to be higher.



(a) Difference of the sum of squared residuals based on the original and manipulated code observation.



(b) Difference of the sum of squared residuals based on the original and manipulated carrier–phase observation.

Figure 6.10: Difference of the sum of squared residual based on the original and manipulated observations. Left the pseudorange range residuals are used, and for the figure on the right the carrier–phase observations are used.

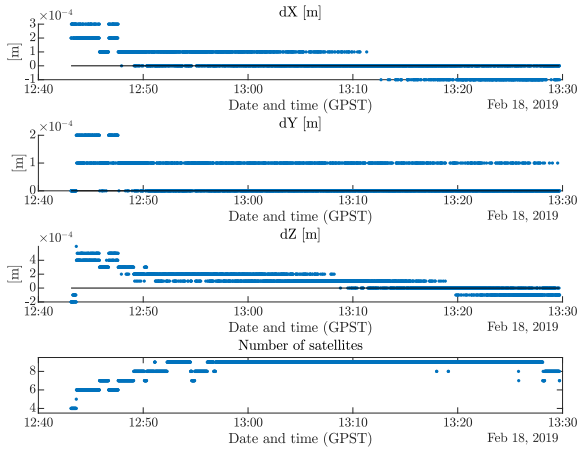
The residuals are not significantly changed based, therefore it is concluded that the manipulation is successful.

6.5. Timing

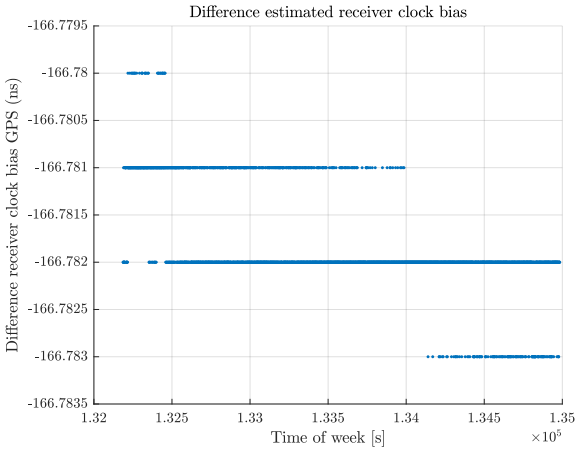
In addition to positioning, GNSS can also be used for timing applications. When this is done the parameter of interest is the receiver clock bias. This bias can be used to correct the receiver clock to synchronize it with GNSS time. In this case only GPS satellites are used, which means that the time is synchronized with GPS time.

For timing applications it is also possible to use differential GNSS, but the parameter that is estimated is not longer the receiver clock bias, but the relative clock bias δt_{12}, see Equation (2.31).

In this case a bias of 50 meter is used. A manipulation of 50 meter is equal to 166.78ns, $\frac{50}{c} = 1.6678\text{e}{-7}\text{s}$. The SPP solution, where all four parameters (x, y, z and δt_1) are estimated, is shown in Figure 6.11. The manipulation is only in the clock bias, therefore it is expected that the change in position is zero.



(a) Difference in x, y and z solution between the original and manipulated data. The manipulation is on the receiver clock bias. The difference in the final solution (x, y and z) is in the order of tenths of millimeters, see the scale of the y-axes which are 1e−4m.

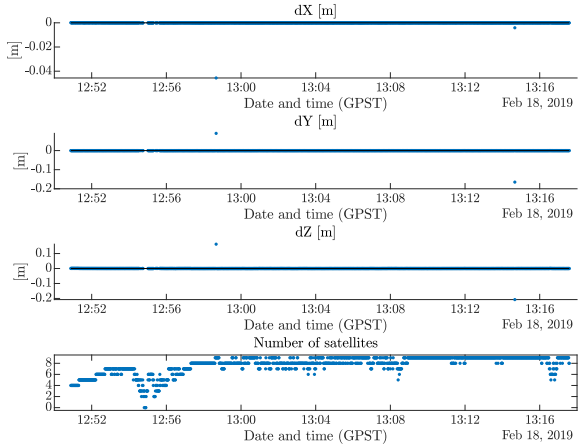


(b) The difference in estimated receiver clock bias δt based the original and manipulated data. The input manipulation in clock bias is $\frac{50}{c} \approx 166.78204$ns, where c is the speed of light. The result is given in nano seconds with three descimals, which explains the jumps in the data.

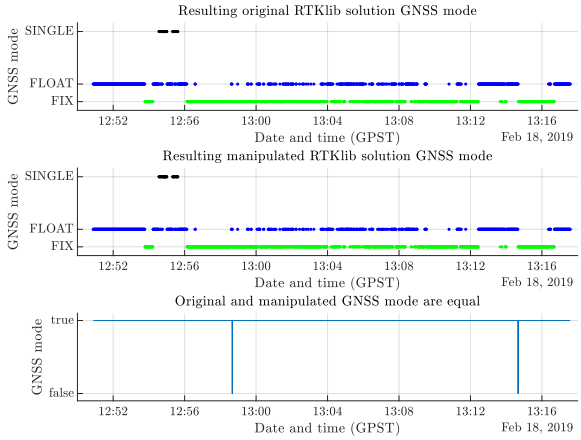
Figure 6.11: RTKlib solution with a manipulation in the receiver clock bias.

What can be seen in Figure 6.11a is that the manipulation has no effect on the final x, y and z solution. The difference between receiver clock bias based on the original and the manipulated data is shown in Figure 6.11b. It shows that the difference is almost exactly the input manipulation. The results are in nano seconds and given with three decimals. Therefore the jumps shown in this graph are a result of the numerical rounding errors of the result by RTKlib.

In differential mode there is no output for the relative clock bias by RTKlib. The positioning results are shown in Figure 6.12. What can be seen is that there is almost no effect in the x, y and z parameters. For two epochs the solution status of the manipulated data was different compared with the original data. The fact that it is shown that for the SPP solution there is a difference in estimated receiver clock bias it can be concluded that the effect of the manipulation is in the estimated relative clock bias.



(a) The difference between the original and the manipulated data.



(b) The solution status of the original and the manipulated data.

Figure 6.12: The RTK result of the original and the manipulated data. The manipulation is only in the clock bias.

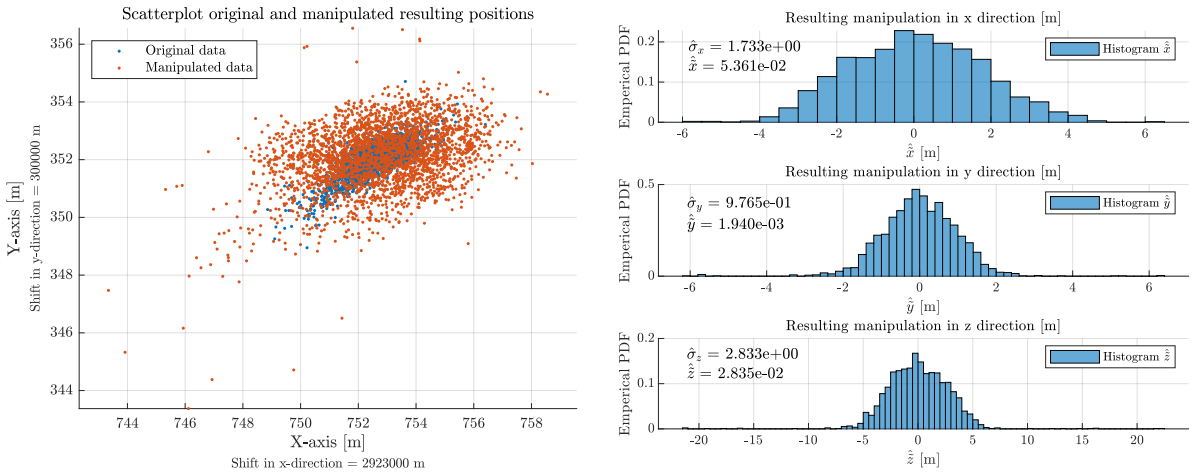
6.6. Arbitrary manipulation

So far it is shown that it is possible to manipulate the augmentation data so that the final parameters of interest are manipulated in a certain direction with a given length; $\tilde{\mathbf{x}} = [1, 2, 0, 0]^T$. Now it is time to show what happens when the observation data is manipulated in a arbitrary way.

To keep in the same order of length of the manipulation the following approach is used. First the manipulation vector is calculated based on the procedure as described before. Each epoch has m observations, and there are thus m manipulation values. Those values are arbitrary assigned to the original observations. This means that each manipulation value is used once, but not necessarily to the corresponding satellite.

This will show the effect of what happens when the observation manipulation vector has the same length as before, but the new manipulation vector is not in the range space of A, $\tilde{\mathbf{y}} \notin \mathcal{R}(A)$, due to arbitrary assigned position of each element of vector $A\tilde{\mathbf{x}}$, which is in the range space of A.

To analyse the effect of the random manipulation the SPP solution of the reference station is shown in Figure 6.13. What can be seen is that the estimated behaviour has a random distribution around zero. The point cloud in the scatterplot shown in Figure 6.13a shows a large spread.



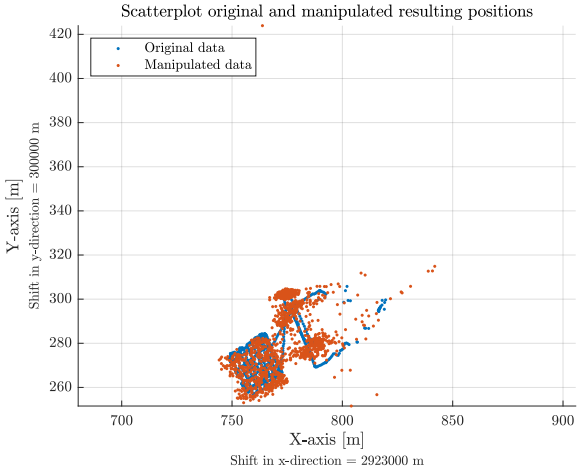
(a) The single point solution of the reference station. The blue dots shows the solution based on the original data, the orange dots shows the solution based on the manipulated data.

(b) The difference between the original and manipulated single point solution of the reference station per x, y and z component shown in an a histogram.

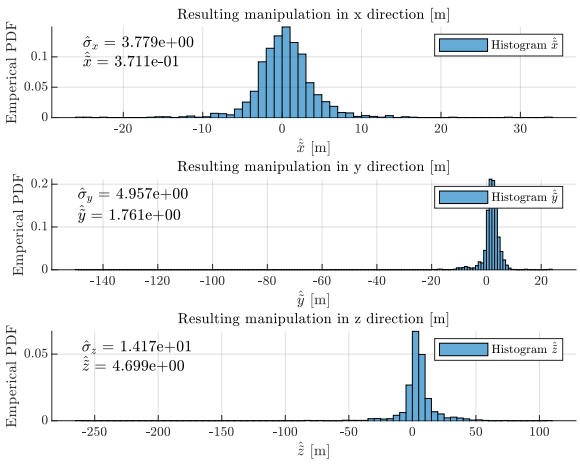
Figure 6.13: The Single-point positioning (SPP) solution after an arbitrary manipulation of the observations.

From the SPP results of the reference station it can already be said that the resulting manipulation for differential positioning will also be random distributed. What remains unknown is the effect on the RTK solution with such data. It can be possible that a random manipulation is successful, which means that the manipulation is not detected, but that the attacker cannot “steer” the manipulation in a desired direction.

The RTK solution is shown in Figure 6.14. This result shows indeed a random behaviour in the resulting manipulation. The mean manipulation for the x, y and z solution is not centered around zero.



(a) The solution, based on the orginal data and the arbitrary manipulated data, in x and y direction.



(b) The histogram for x, y and z direction.

Figure 6.14: The Real Time Kinematic (RTK) solution of a random manipulation.

In Figure 6.15 the RTK status is shown. What can be seen is that there is almost all the time a float solution. This means that the software is not able to find the ambiguities in this dataset with arbitrary manipulated observations.

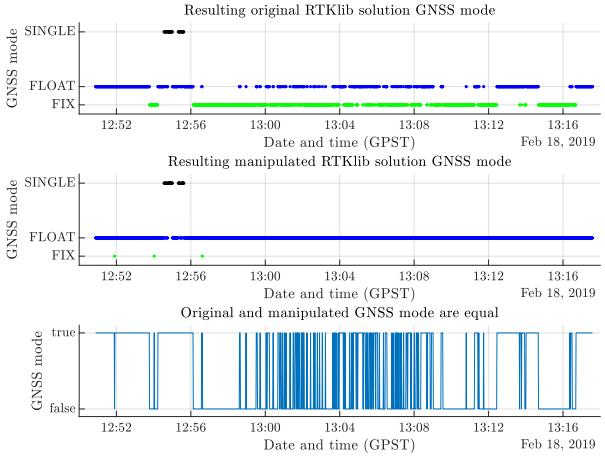


Figure 6.15: The GNSS solution status for the original and arbitrary manipulated data. What can be seen is that there are almost no fixed solutions when the data is arbitrarily manipulated.

The manipulation in combination with a corresponding model, which is also used by the user, is not only necessary for the direction and magnitude of the manipulation, but also to ensure that the receiver is able to obtain a fixed solution in RTK mode. The resulting solution after an arbitrarily manipulation, the spread of the solution larger than the expected spread of a SPP model. The sum of the squared mean residuals is 11.99, while the critical value for an overall model test on the position is 7.8147, from the chi–square distribution with 3 degrees of freedom and an alpha level of 0.05. Therefore, this manipulation can easily be detected when the user also calculates and checks the reference position based on the SPP solution from the augmentation data.

6.7. Case study “Car experiment A13”

The dataset from [37] is used to show the manipulation in combination with a real life GNSS application that can be used (in the near future) for example for self driving cars. The used reference receiver is a Trimble NetR9 and the reference antenna is Leica AR25.R3, and the used rover receiver is a Trimble R7-a with a Trimble Zephyr Geodetic L1/L2 antenna type 41249–00.

Figure 6.16: Antenna set-up on the roof of the test vehicle [37]. The used antenna for this study is in the front left position.

The data were collected by a driving car on a highway in the Netherlands, the A13, over 5.5 kilometer. The sampling rate was 10Hz. A part of the data is shown in Figure 6.17. For this study three runs are used, two times from south to north and once from north to south. The total duration is approximately 15 minutes. This segment already shows the possibilities for data manipulation for high precision applications. In this case two manipulation vectors are used, both in the same direction. One manipulation vector has the same length as before, 2.24m and the other manipulation vector has a length of 7.00m. The manipulation vectors are approximately perpendicular to the highway, with an azimuth of 75°. The manipulation vector is defined in North, East, Up (NEU) and afterwards transformed to ECEF coordinates. Based on the desired length (l) and azimuth (α) the manipulation vector can be calculated in NEU, see Equation (6.1). Based on the reference station position this vector is transformed to ECEF.

$$[\text{North} , \text{East} , \text{Up}]^T = l \cdot [\cos\alpha, \sin\alpha, 0]^T \tag{6.1}$$

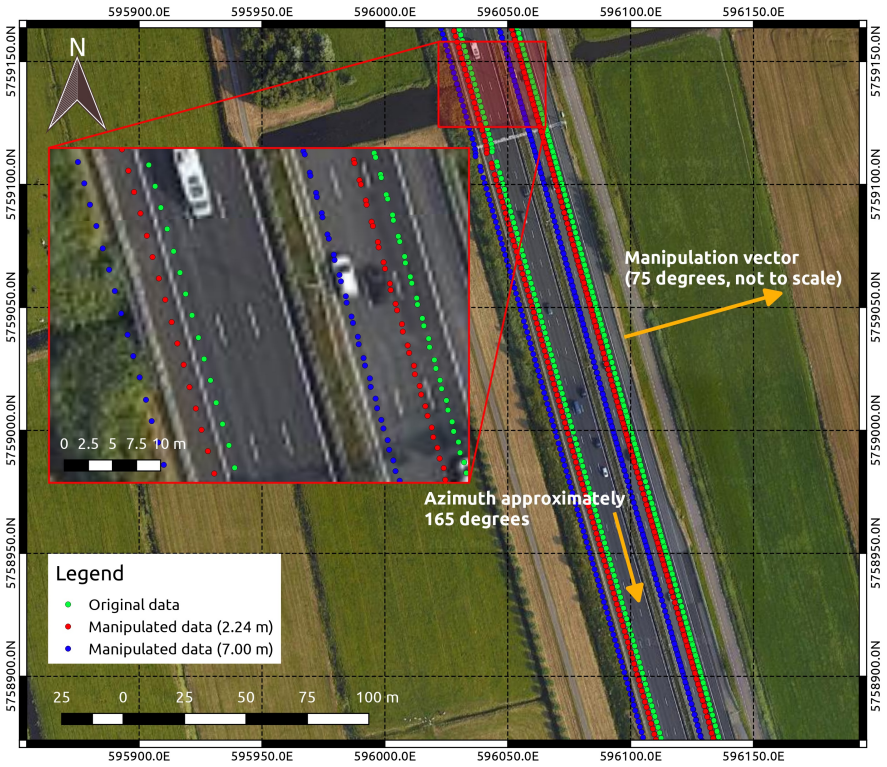


Figure 6.17: Car experiment A13 [18] data on top of a satellite image, using the UTM31N projection (EPSG:32631). The manipulation vector is approximately perpendicular to the highway, with an azimuth of 75° and with a length of 2.24 and 7 meter. The green dots represents the original data (the car taking the right most lane), the red dots represents the manipulation of 2.24m and the blue dots represents the manipulation of 7.00m. The manipulation of the rover position is in the opposite direction of the manipulation vector, see Figure 6.4.

For the post processing of the data the default RTKlib configuration for RTK is used. The most important difference is that the Receiver Autonomous Integrity Monitoring (RAIM) and Fault Detection and Exclusion (FDE) feature is enabled. This means that when the sum of the squared errors of the residuals is over a threshold a satellite can be excluded [32].

What can be seen in the overview in Figure 6.17 is that the manipulation is significant compared to the lane width. In the Netherlands the lane width on a high way is 3.5 meter [18]. The reason for choosing a manipulation vector length of 2.24 meter is because this is the same size as used before. The length of 7 meter is used to show that a larger manipulation vector is also possible using the default settings of RTKlib. Also the 7 meter is equal to two lanes on a Dutch highway. The manipulation is approximately perpendicular to the highway, therefore the manipulation with a length of 7 meter is exactly 2 lanes. Note that the observations of the reference station are manipulated with an azimuth of 75°, which means that the final rover position is manipulated with a direction of 225°, see Figure 6.4.

The manipulation vector and the resulting manipulation vector is shown in Table 6.4. What can be seen is that both vectors are correct estimated and thus that the manipulation is successful.

$\lvert\tilde{\mathbf{x}}\rvert$	$\tilde{\mathbf{x}}$			$\hat{\tilde{\mathbf{x}}}$		
	x	y	z	x	y	z
2.2361	−0.6199	2.1187	0.3564	−0.6197	2.1184	0.3564
7.0000	−1.9405	6.6325	1.1158	−1.9398	6.6319	1.1159

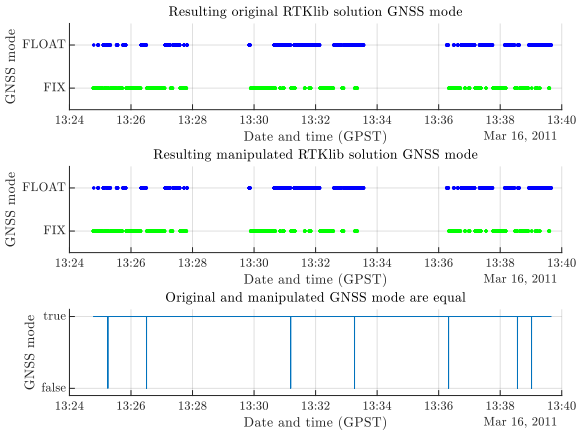
Table 6.4: The manipulation vector and the estimated manipulation vector in ECEF for the A13 dataset.

Figure 6.18 shows the result in x, y and z direction. This figures also shows the GNSS mode as function of time. What can be seen is that due to the manipulation at several epochs the solution status is float instead of fix. This occurs more often with a larger manipulation vector.

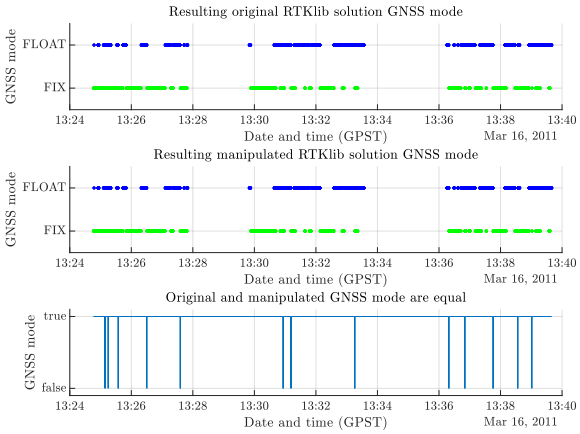
The estimated manipulation vector is is not always estimated correct, see Figure 6.18c and Figure 6.18d. This error occurs when the solution has another solution status, see Figure 6.18a and Figure 6.18b. What can be seen is that when a larger manipulation vector is used, more often the solution mode (fixed or a float solution) is different compared with the solution based on the original observations. Even with a larger manipulation, like the used 7 meter, there was always a differential solution. This means that the solution is always manipulated.

The process to get a fixed solution is a stochastic process. When considering the manipulation of 7 meter, there are 16 occurrences where the original data and the manipulated data does not have the same GNSS mode. The total number of epochs is 6131. The original data has 3139 fixed solutions and 2992 float solutions. This means that 51.20% of the solutions is fixed. The manipulated data has 3149 fixed solutions and 2982 float solutions. This means that 51.36% of the solutions is fixed. It can thus be concluded that some differences in the solution status does not mean that the manipulation is unsuccessful. In this case the manipulated data has even more fixed solutions than the original data.

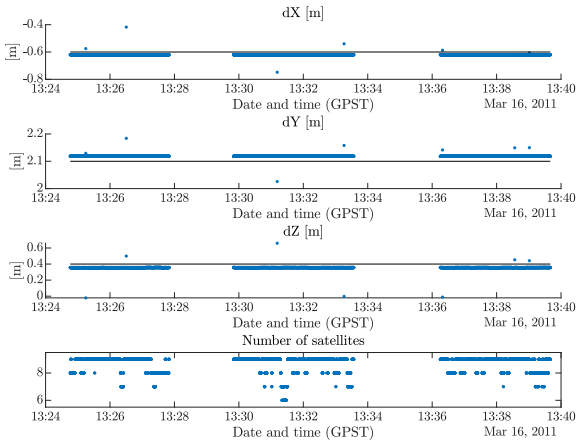
Given that a lane has a width of 3.5 meter, a manipulation of a few meters, as shown above, can be the cause for dangerous situations. High precision GNSS solutions are used for applications like self driving cars. A self driving car can use the precise GNSS position to determine if the car should move to another lane. For example,see Figure 6.17, the manipulated position of the north to south route (on the left side of the highway) is in the verge of the road. To correct for this (wrong) position the car should steer to the left, which means it is actually steering onto another lane, which may be occupied. In the other direction, from south to north (on the right side of the highway), the manipulated position is to the left on another lane. A autonomous car may decide to go to the right lane assuming that there are is no other traffic. To accomplish that, the car should steer to the right, which means in reality the car is steering into the verge of the road.



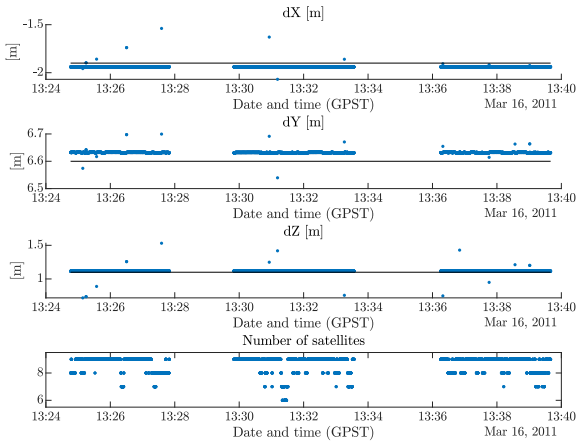
(a) The GNSS mode for the manipulation vector of length 2.24 meter. The bottom plot shows if the GNSS modes for the original data and the manipulated data are equal.



(b) The GNSS mode for the manipulation vector of length 7.00 meter. The bottom plot shows if the GNSS modes for the original data and the manipulated data are equal.



(c) The difference between the solution, based on the original and the manipulated data, is shown in the upper three plots. The used manipulation vector is $[0.62, -2.12, -0.36, 0]^T$ meter. The fourth plot shows the GNSS mode. The bottom plot shows the number of used satellites.



(d) The difference between the solution, based on the original and the manipulated data, is shown in the upper three plots. The used manipulation vector is $[1.94, -6.63, -1.12, 0]^T$ meter. The fourth plot shows the GNSS mode. The bottom plot shows the number of used satellites.

Figure 6.18: Results for the manipulation of the A13 dataset. The upper two figures shows the GNSS mode for a manipulation length of 2.24 meter and 7.00 meter respectively. The bottom two plots shows the resulting manipulation vector for the x, y and z components, and the amount of used satellites.

7

Conclusion

The purpose of this study was to analyse whether it is possible to manipulate GNSS augmentation data without being detected as well as the consequences for the estimated parameters. This study shows that it is possible to spoof the augmentation data without raising detection.

The first step is that an attacker has access to the GNSS augmentation data, before the attacker is able to spoof the data. A hacker has access to the data, since the protocol that is used to send the augmentation data, is based on HTTP. This protocol is not encrypted and messages, which are sent to the user, have no authentication.

Based on the data that are intercepted it is possible to create a manipulation vector which is present almost only in the influential bias and not in the testable bias. This means that such a manipulation is not detected by the user by the means of statistical testing like the overall model test or the w-test.

In terms of the solution of the target user receiver, the manipulation is successful. This means that the user receiver solution can be manipulated fully according to the intentions of the attacker. It is possible to manipulate each parameter of interest, position coordinates ($x\ y\ z$) and receiver clock offset (δt), separately or in combination.

In the last example it is shown that a manipulation is possible of augmentation data used for lane identification on a highway. For lane identification high precision positions are necessary, which were not trustworthy anylonger after the GNSS augmentation data were manipulated.

The possibilities to manipulate the augmentation data are not endless. Only a part of the data are manipulated, namely the augmentation data or the data from the reference station, but the observations done by the user receiver are protected. This means that the user can use those protected observations in combination with a Single-point positioning (SPP) solution to check if the difference between the DGNSs and the SPP solution is significant. In practice it means that the size of the manipulation is limited to the precision of the SPP solution of the user.

7.1. Recommendations

After it is shown that the manipulation of the augmentation data is a potential threat, there are also some recommendations. There are different solutions to solve the vulnerabilities, namely from an ICT point of view and a mathematical data processing point of view.

For newer devices a newer protocol would be a good solution. The protocol that is used at the moment, HTTP, is labelled as insecure in a browser (for example Google Chrome). Therefore a first step would be to move from HTTP to HTTPS. This does not mean that HTTPS is 100% safe, but it is a first step and it is definitely harder to hack than pure HTTP. One thing that is important to ensure is compatibility with older firmware.

Another option is to use a second encrypted link which is solely used for authentication. It is possible to use a hash function in combination with the augmentation data. The user is able to check its own calculated hash based on the received data (from the unsecure link) and check if it is the same hash as calculated by the reference station. Older hardware will then be compatible, since the “old” unsecure link is not changed.

For GNSS software it would be good to be aware of the possibility that the augmentation data that are received are not authentic.

The first thing software can do is to use unmanipulated observations, namely the observations of the user receiver, and compare a single point solution solution with the differential solution, based on augmentation data that may have been manipulated. This is a solution that can detect large manipulations, that is, manipulations larger than the accuracy of the single point position solution.

Based on the augmentation data, the position of the reference station can also be calculated (also in SPP mode). This will also detect large manipulations since the position of the reference station is known. If augmentation data are available for a longer time it can be possible to see a trend in the data. For example if the mean position of the reference station over 24 hours is not close to the known position, this can be an indication that the augmentation data are manipulated. This is however not a real-time solution.

7.2. Augmentation data spoofing detection

A brief introduction is given to a possible method that can be used for detecting augmentation data spoofing. The general idea is shown, and thereafter an example is given where the augmentation data is tested using the known position of the reference station and the SPP solution of the reference station based on the augmentation data.

When assuming that an attack only has access to the augmentation data then not all observations are manipulated. Using the observations of the user receiver it is possible to detect augmentation data spoofing to a certain extent.

It is assumed that the attacker only has access to the augmentation data and that there is no radio spoofing on the rover location. This means that the observations of the rover are unbiased. The reference station position is assumed to be known and static. It is also possible that the reference station is known based on the information in the RTCM message, which the attacker is able to manipulate. It is assumed that the software in the rover does not change the known position of the reference station without notifying the user. The only reason one would automatically update the reference station position is when a virtual reference station (VRS) is used. A VRS is often used for network RTK. For now traditionally RTK short-baseline positioning is used, which means that there is one known reference station and one rover.

The precision of pseudorange observations is 3m and the precision of carrier phase observations is 3mm, thus 1000 times more precise. For RTK positing there are four sets of observation equations. Namely the carrier phase and the code observation of the reference station, respectively $\underline{\mathbf{\Phi}}_1$ and $\underline{\mathbf{p}}_1$, and the carrier phase and the code observation of the rover, respectively $\underline{\mathbf{\Phi}}_2$ and $\underline{\mathbf{p}}_2$. The reference station is denoted with subscript $(\cdot)_1$ and the user receiver is denoted with subscript $(\cdot)_2$.

The first two methods which are discussed use a single epoch approach. After that two multiple epoch approaches are described.

7.2.1. Single epoch

The single epoch solution is based on 1 epoch at a time. Information from previous epochs is not used.

The general idea is to test if the difference between two positions is significant. For the reference station the estimated position is compared with the known position of the reference station. For the rover the difference between the estimated RTK position and the SPP position of the rover is compared.

Single point solution on reference station

The single point solution for the reference station is based on the augmentation data. This data may have been manipulated. The position of the reference station is known.

The null hypothesis for the reference station is:

$$H_0 : \mathbf{x} = \hat{\mathbf{x}}_1$$

When there is no manipulation then the single point position solution does not differ significant from the known position of the reference station. The expectation is thus:

$$\mathrm{E}(\mathbf{x}_1 - \underline{\hat{\mathbf{x}}}_1) = 0 \tag{7.1}$$

The overall model test for the positions is shown below.

$$T_{q=3} = (\mathbf{x} - \underset{\sim}{\hat{\mathbf{x}}}_1)^T Q_{\hat{x}\hat{x}}^{-1} (\mathbf{x} - \underset{\sim}{\hat{\mathbf{x}}}_1) \tag{7.2}$$

The variance $Q_{\hat{x}\hat{x}}^{-1}$ is a result of the used model for the SPP solution. This test only test the position and the estimated receiver clock bias is not used. To be able to test the all observations it is possible to eliminate the receiver clock bias using a linear combination of the observations.

$$M\,\mathrm{E}(\mathbf{p}_1) = MA\mathbf{x} \tag{7.3}$$

Where M denotes a $[m-1 \times m]$ matrix.

$$M = \begin{bmatrix} -1 & 1 & & \\ -1 & & 1 & \\ \vdots & & & \ddots \\ -1 & & & 1 \end{bmatrix} \tag{7.4}$$

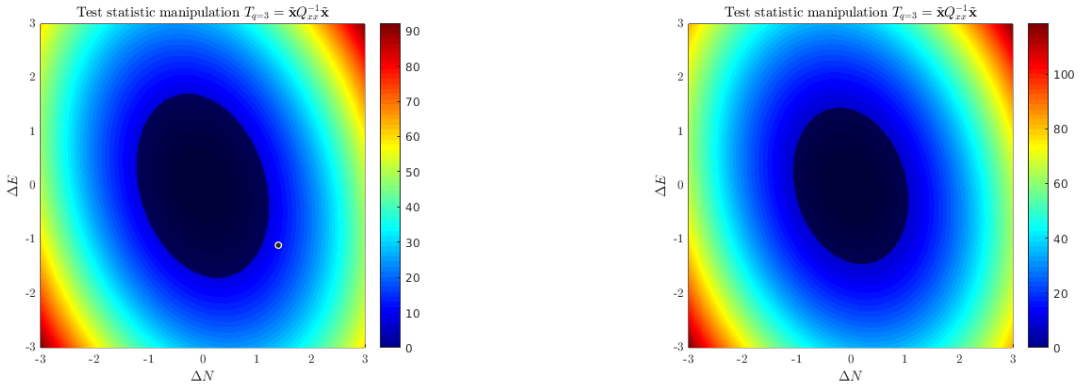
The variance of the observations is given as unit matrix scaled by the variance of the code observation, $\sigma_p = 3$ meter. The propagation of the variance:

$$Q_{SD} = M\sigma_p^2 \mathbf{I}_{[m\times m]} M^T = \sigma_p^2 \begin{bmatrix} 2 & 1 & & \cdots & 1 \\ 1 & 2 & \ddots & & \vdots \\ & \ddots & \ddots & & \\ \vdots & & & 2 & 1 \\ 1 & \cdots & & 1 & 2 \end{bmatrix} \tag{7.5}$$

Due to the cancellation of the receiver clock bias the variance matrix is now a full matrix instead of diagonal matrix. This means that the new observations are correlated with each other.

The difference matrix operator is also applied on the design matrix A, MA. The result is that the last column of A, which is filled with ones for the receiver clock bias, are all zeros. Therefore the last column is removed from the design matrix and the receiver clock bias is no longer estimated. The new design matrix, MA, has $m-1$ row, where m denotes the number of observations.

Example The known position of the reference station is used to determine which sample sizes are possible to detect. The detection is dependent on the geometry of the observation, A, and the precision of the observation. Three examples are shown. In one example the bias is calculated based on one epoch. In the other case the mean variance matrix of RTKlib is used for $Q_{\hat{x}\hat{x}}$. The effect is that the area of no detection is more circular, because the observation geometry is also averaged.



(a) The overall model test based on a varaince matrix from 1 epoch

(b) The overall model test based on a mean variance matrix from, based on approximate 45 minutes.

Figure 7.1: Detection threshold for one epoch and the mean variance matrix. The area where the spoofing is not detected, with an alpha level of 0.05, is shown in black.

The manipulation vector inside the greyish area is not detected based on an alpha level of 0.05. According to this data a manipulation in the order of meters can be detected.

Single point solution for rover
What is shown above can also be done at the rover. The rover can use its own, unmanipulated, observations to calculate the SPP solution. This solution is then tested against the RTK solution. The difference with the testing using the reference station is that both the RTK solution and the SPP solution has a certain precision. For the propagation of variances the two variance matrices, one from the RTK solution and one from the SPP solution, are added together. The variance matrix for the SPP solution is defined above.

The RTK solution combines all observations, $\mathbf{p}_1$, $\mathbf{\Phi}_1$, $\mathbf{p}_2$ and $\mathbf{\Phi}_2$. The observation equation that are used are given in [33, Equation (1)], see below.

$$D^T\mathbf{p}_j(i) = D^T A_i\mathbf{x} \tag{7.6}$$
$$D^T\mathbf{\Phi}_j(i) = D^T A_i\mathbf{x} + \lambda_j\mathbf{a}_j \tag{7.7}$$

Where j denotes the used frequency, $i = 1, \dots, k$ the epoch number, A_i the design matrix at epoch i and $\mathbf{a}_j$ contain the unknown ambiguities. $\mathbf{p}_j$ and $\mathbf{\Phi}_j$ contain the single differenced code and phase observations. The matrix D^T is the $[m-1 \times m]$ difference matrix operator. This matrix is used to go from single-differenced model to a double-differenced model. For now only the floated baseline is used.

The variance matrix for both the code and the phase measurement is given as: [33, Equation (6)]

$$Q_{\hat{x}\hat{x}}(\mathbf{\Phi}) = \frac{1}{(\alpha_1+\alpha_2)}\left[\sum_{i=1}^{k}(A_i-\bar{A})^T P(A_i-\bar{A})\right]^{-1} \tag{7.8}$$
$$Q_{\hat{x}\hat{x}}(\mathbf{p}) = \frac{1}{(\beta_1+\beta_2)}\left[\sum_{i=1}^{k}A_i^T P A_i\right]^{-1} \tag{7.9}$$
$$Q_{\hat{x}\hat{x}}(\mathbf{\Phi},\mathbf{p}) = [Q_{\hat{x}\hat{x}}(\mathbf{p}) + Q_{\hat{x}\hat{x}}(\mathbf{\Phi})]^{-1} \tag{7.10}$$

Where $\bar{A} = \frac{1}{k}\sum_{i=1}^{k}A_i$ and $P = D(D^TD)^{-1}D^T$. And α_1 and α_2 are the weights for the phase observations of the reference receiver and the rover receiver, β_1 and β_2 are the weights for the code observation of the reference receiver and the rover receiver. k is the amount of epochs, in this case $k = 1$.

Based on this function the variance matrix of the RTK solution is also known. The variance matrix of the solution for RTK can be denoted as $Q_{2,rtk} = Q_{\hat{x}\hat{x}}(\mathbf{\Phi},\mathbf{p})$. The final test is then:

$$T_{q=3} = (\bar{\mathbf{x}}_{2,rtk} - \underline{\hat{\mathbf{x}}}_{2,spp})^T(Q_{2,spp} + Q_{1,rtk})^{-1}(\bar{\mathbf{x}}_{2,rtk} - \underline{\hat{\mathbf{x}}}_{2,spp}) \tag{7.11}$$

The variance matrix is larger than when only the SPP solution is used, which was the case for the reference station. The advantage of this test is that also the phase observations are used.

Multiple epoch Based on Equation (7.10) it is possible to calculate the variance matrix for multiple epochs. What can be seen is that when only one code observation is used, i.e. $\beta_2 = 0$, the solutions precision increase each epoch. The standard deviation for the solution decrease per epoch with a factor $\frac{1}{\sqrt{k}}$. This can be too optimistic because the observations may contain non-Gaussian distributions and time-correlated biases.

What is shown is that it is possible to detect the augmentation spoofing, when the used manipulation vector was significant. Further research should reveal if multiple epoch solutions, fixed RTK solutions and the addition of the phase observation to the SPP should would make it possible to detect smaller biases than shown before.

A

Taylor series for square root

The Taylor series is used for the linearisation. It is tried to describe the behaviour of the Taylor series of the square root function.

A.1. Taylor series

The Taylor series is defined as:

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0)^1 + \frac{f''(x_0)}{2!}(x - x_0)^2 + \frac{f'''(x_0)}{3!}(x - x_0)^3 + \cdots \tag{A.1}$$

$$= \sum_{n=0}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \tag{A.2}$$

Where $f^{(n)}(x_0)$ is the nth derivative of f evaluated in point x_0. This holds as long as $|x - x_0| < R$ where R is the region of convergence.

A series is called convergent if $\lim_{n\to\infty} s_n = s$ is a real number. $s_n = \sum_{i=1}^{n} a_i$ is the partial sum of the series. If the series is convergent then s is the called the sum of the series. This series is convergent only when the limit $\lim_{n\to\infty} a_n = 0$.

A power series is given as:

$$\sum_{n=0}^{\infty} c_n(x - x_0)^n \tag{A.3}$$

Where c_n denotes a constant.

There are three possibilities:

1. The series converges only when $x = x_0$
2. The series converges for all x
3. There is a positive number R such that the series converges if $|x - x_0| < R$ and diverges if $|x - x_0| > R$

One can use the ratio test to test where the series converges.

$$\lim_{n\to\infty} \left| \frac{a_{n+1}}{a_n} \right| = L < 1 \tag{A.4}$$

The nth-degree Taylor polynomial of f at x_0 is given as T_n, this is the Taylor series up to the nth derivative.

On the interval $|x - x_0| < R$ then in general:

$$f(x) = \lim_{n\to\infty} T_n(x) \tag{A.5}$$

This means that there is a remainder term if $n \nrightarrow \infty$.

$$R_n(x) = f(x) - T_n(x) \tag{A.6}$$

A.2. Taylor series root function

In this case the function f is the root function $\sqrt{\cdot}$. The function is continuous for all positive numbers.

The first Taylor polynomial for the root function is defined as:

$$T_1(x) = \sqrt{x_0} + \frac{1}{2}x_0^{-0.5}(x - x_0) \tag{A.7}$$

$$= \sqrt{x_0} + \frac{1}{2\sqrt{x_0}}(x - x_0) \tag{A.8}$$

The remainder term is then:

$$R_1(x) = \sqrt{x} - T_1(x) \tag{A.9}$$

$$= T_\infty - T_1 \tag{A.10}$$

Therefore the remainder is also the Taylor series without the first two terms:

$$R_1 = \sum_{n=2}^{\infty} \frac{f^{(n)}(x - x_0)^n}{n!} \tag{A.11}$$

The first few terms are defined as:

$$f^0 = x^{0.5} \tag{A.12}$$

$$f^1 = 0.5x^{-0.5} \tag{A.13}$$

$$f^2 = -0.25^2 x^{-1.5} \tag{A.14}$$

$$f^3 = 0.375x^{-2.5} \tag{A.15}$$

$$f^4 = -0.9375x^{-3.5} \tag{A.16}$$

$$f^5 = 3.28125x^{-4.5} \tag{A.17}$$

$$f^6 = -14.765625x^{-5.5} \tag{A.18}$$

The first part of the derivative, the numerical value in front of x, can be written as c_n:

$$c_n = (-1)^{n+1} \frac{\prod_{n=2}^{n} 2(2n-3)}{2^n} \qquad \text{for } \{n \in \mathbb{Z} \mid n \geq 2\} \tag{A.19}$$

The last part of the derivative can also be written using $x^{\frac{1}{2}-n}$ so that the final derivative is:

$$f^n = c_n x^{\frac{1}{2}-n} \tag{A.20}$$

Finally this can be used to calculate and analyse the behaviour of the remainder R_1:

$$R_1(x) = \sum_{n=2}^{\infty} (-1)^{n+1} \frac{\prod_{n=2}^{n} 2(2n-3)}{2^n} x_0^{\frac{1}{2}-n} \frac{(x - x_0)^n}{n!} \tag{A.21}$$

A.2.1. Ratio test

The ratio test is used to check if the series is convergent, or to find the region of convergence. The first part $(-1)^{n+1}$ result in a negative value for the ratio test, since either a_n or a_{n+1} is negative. The ratio test is absolute so this part of the a_n can be neglected.

$$\left|\frac{a_{n+1}}{a_n}\right| = \left|\frac{\frac{\prod_{n=2}^{n+1} 2(2n-3)}{2^{n+1}} x_0^{\frac{1}{2}-n-1} \frac{(x-x_0)^{n+1}}{(n+1)!}}{\frac{\prod_{n=2}^{n} 2(2n-3)}{2^{n}} x_0^{\frac{1}{2}-n} \frac{(x-x_0)^{n}}{n!}}\right| \tag{A.22}$$

$$= \left|\frac{2^n}{2^{n+1}} \frac{\prod_{n=2}^{n+1} 2(2n-3)}{\prod_{n=2}^{n} 2(2n-3)} \frac{x_0^{\frac{1}{2}-n-1}}{x_0^{\frac{1}{2}-n}} \frac{(x-x_0)^{n+1}}{(x-x_0)^n} \frac{n!}{(n+1)!}\right| \tag{A.23}$$

$$= \left|\frac{1}{2} 2(2(n+1)-3) \frac{x-x_0}{x_0} \frac{1}{(n+1)}\right| \tag{A.24}$$

$$= \left|\frac{(2n-1)(x-x_0)}{x_0(n+1)}\right| \tag{A.25}$$

To find the region the final part is to find where $\lim_{n\to\infty} \left|\frac{(2n-1)(x-x_0)}{x_0(n+1)}\right| < 1$.

Using the rule that $\lim_{x\to b}[f(x)g(x)] = \lim_{x\to b} f(x) \cdot \lim_{x\to b} g(x)$

$$\lim_{n\to\infty} \left|\frac{(2n-1)(x-x_0)}{x_0(n+1)}\right| = \lim_{n\to\infty} \left|\frac{x-x_0}{x_0}\right| \lim_{n\to\infty} \left|\frac{2n-1}{n+1}\right| \tag{A.26}$$

$|2n-1| = 2n-1$ and $|n+1| = n+1$ since $n \to \infty$.

$$= \lim_{n\to\infty} \left|\frac{x-x_0}{x_0}\right| \lim_{n\to\infty} \frac{2-\frac{1}{n}}{1+\frac{1}{n}} \tag{A.27}$$

Using the form $\lim_{x\to b}\left(\frac{f(x)}{g(x)}\right) = \frac{\lim_{x\to b} f(x)}{\lim_{x\to b} g(x)}$, given that $\lim_{x\to b} g(x) \neq 0$. Also $\lim_{n\to\infty} \left|\frac{x-x_0}{x_0}\right| = \left|\frac{x-x_0}{x_0}\right|$ since there is no n inside the limit.

$$= \left|\frac{x-x_0}{x}\right| \frac{\lim_{n\to\infty} 2-\frac{1}{n}}{\lim_{n\to\infty} 1+\frac{1}{n}} \tag{A.28}$$

$$= \left|\frac{x-x_0}{x}\right| \left(\frac{2}{1} + \frac{\lim_{n\to\infty} \frac{-1}{n}}{\lim_{n\to\infty} \frac{1}{n}}\right) \tag{A.29}$$

Note that $\lim_{n\to\infty} \frac{1}{n} = 0$ and that $\lim_{n\to\infty} \frac{1}{n} = \lim_{n\to\infty} \frac{-1}{n}$

(A.30)

$$= \left|\frac{x-x_0}{x}\right| 2 \tag{A.31}$$

The convergence region is defined as $\frac{|x-x_0|}{|x_0|} 2 < 1$.

$$1 > \frac{|x-x_0|}{|x_0|} 2 \tag{A.32}$$

$$\frac{1}{2} > \left|\frac{x-x_0}{x_0}\right| \tag{A.33}$$

$$\frac{1}{2} > \left|\frac{x}{x_0} - 1\right| \tag{A.34}$$

$$-\frac{1}{2} > \frac{x}{x_0} - 1 > \frac{1}{2} \tag{A.35}$$

$$\frac{1}{2} > \frac{x}{x_0} > \frac{3}{2} \tag{A.36}$$

$$\frac{x_0}{2} > x > \frac{3x_0}{2} \tag{A.37}$$

Therefore the point of interest (x) has to be in between $\frac{1}{2}x_0$ and $1.5x_0$ so that the absolute difference $|x - x_0| < 0.5x_0$. This is the region of convergence for the remainder function, equation (A.21).

When the root is used for the estimation of y with a functional model like $y = \sqrt{x}$, the least squares can be used. Since it is a non-linear model, a Gauss-Newton iteration can be used to find the smallest $\Delta\hat{x}$ which corresponds for the measured y. Even if the start point is far off, the difference between the calculated y_0 in x_0 will have this relationship:

$$y - y_0 > 0 \implies x - x_0 > 0 \tag{A.38}$$

$$y - y_0 < 0 \implies x - x_0 < 0 \tag{A.39}$$

$$y - y_0 = 0 \implies x - x_0 = 0 \tag{A.40}$$

For a fixed value of $|x - x_0|$ it can be seen that as long as the linearization point is large (x_0) the remainder term is smaller compared to a small x_0. Therefore it is possible to invert the linearization for small values of x_0 so that the linearization still "fast" converges.

Assume a value for $x \in (0, 1)$, then there is value for $10^{-i} = x$ where $i = -\log_{10} x \implies i > 0$. This also implies that $10^i > 0$. Then it follows that $\sqrt{x} = \sqrt{10^{-i}} = \sqrt{\frac{1}{10^i}} = \frac{\sqrt{1}}{\sqrt{10^i}}$. The value $\sqrt{10^i} > 0$ and is easier to calculated, especially when $x \to 0$ Note that $10^i = 10^{-\log_{10} x} = \frac{1}{10^{\log_{10} x}} = \frac{1}{x}$. Therefore $\sqrt{x} = \frac{1}{\sqrt{\frac{1}{x}}}$. As long $x \in (0, 1)$, the value of $\frac{1}{x} > 0$.

The value of x_0 is thus larger when the inverted value for x_0 is used. Also the convergence region increases when a larger value for x_0 is used.

B

Configuration files RTKlib

In this appendix the used RTKlib configuration can be found. Those contain the settings as they were used to calculate the GNSS solutions which are based on RTKlib.

B.1. Single Point Positioning

rtklib_spp.config

```
# rtkpost_qt options (2020/03/15 23:35:39, v.2.4.3 b8)

pos1-posmode       =single     # (0:single,1:dgps,2:kinematic,3:static,4:movingbase,5:fixed,
                               # 6:ppp-kine,7:ppp-static,8:ppp-fixed)
pos1-frequency     =l1+l2      # (1:l1,2:l1+l2,3:l1+l2+l5,4:l1+l5)
pos1-soltype       =forward    # (0:forward,1:backward,2:combined)
pos1-elmask        =15         # (deg)
pos1-snrmask_r     =off        # (0:off,1:on)
pos1-snrmask_b     =off        # (0:off,1:on)
pos1-snrmask_L1    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L2    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L5    =0,0,0,0,0,0,0,0,0
pos1-dynamics      =off        # (0:off,1:on)
pos1-tidecorr      =off        # (0:off,1:on,2:otl)
pos1-ionoopt       =off        # (0:off,1:brdc,2:sbas,3:dual-freq,4:est-stec,5:ionex-tec,
                               # 6:qzs-brdc,7:qzs-lex,8:stec)
pos1-tropopt       =off        # (0:off,1:saas,2:sbas,3:est-ztd,4:est-ztdgrad,5:ztd)
pos1-sateph        =brdc       # (0:brdc,1:precise,2:brdc+sbas,3:brdc+ssrapc,4:brdc+ssrcom)
pos1-posopt1       =off        # (0:off,1:on)
pos1-posopt2       =off        # (0:off,1:on)
pos1-posopt3       =off        # (0:off,1:on,2:precise)
pos1-posopt4       =off        # (0:off,1:on)
pos1-posopt5       =off        # (0:off,1:on)
pos1-posopt6       =off        # (0:off,1:on)
pos1-exclsats      =           # (prn ...)
pos1-navsys        =1          # (1:gps+2:sbas+4:glo+8:gal+16:qzs+32:comp)
pos2-armode        =continuous # (0:off,1:continuous,2:instantaneous,3:fix-and-hold)
pos2-gloarmode     =on         # (0:off,1:on,2:autocal)
pos2-bdsarmode     =on         # (0:off,1:on)
pos2-arthres       =3
pos2-arthres1      =0.9999
pos2-arthres2      =0.25
pos2-arthres3      =0.1
pos2-arthres4      =0.05
pos2-arlockcnt     =0
pos2-arelmask      =0          # (deg)
pos2-arminfix      =10
pos2-armaxiter     =1
pos2-elmaskhold    =0          # (deg)
pos2-aroutcnt      =5
pos2-maxage        =30         # (s)
pos2-syncsol       =off        # (0:off,1:on)
```

```
pos2-slipthres     =0.05       # (m)
pos2-rejionno      =30         # (m)
pos2-rejgdop       =30
pos2-niter         =1
pos2-baselen       =0          # (m)
pos2-basesig       =0          # (m)
out-solformat      =xyz        # (0:llh,1:xyz,2:enu,3:nmea)
out-outhead        =on         # (0:off,1:on)
out-outopt         =on         # (0:off,1:on)
out-timesys        =gpst       # (0:gpst,1:utc,2:jst)
out-timeform       =hms        # (0:tow,1:hms)
out-timendec       =3
out-degform        =deg        # (0:deg,1:dms)
out-fieldsep       =
out-height         =ellipsoidal # (0:ellipsoidal,1:geodetic)
out-geoid          =internal   # (0:internal,1:egm96,2:egm08_2.5,3:egm08_1,4:gsi2000)
out-solstatic      =all        # (0:all,1:single)
out-nmeaintv1      =0          # (s)
out-nmeaintv2      =0          # (s)
out-outstat        =residual   # (0:off,1:state,2:residual)
stats-eratio1      =100
stats-eratio2      =100
stats-errphase     =0.003      # (m)
stats-errphaseel   =0.003      # (m)
stats-errphasebl   =0          # (m/10km)
stats-errdoppler   =10         # (Hz)
stats-stdbias      =30         # (m)
stats-stdiono      =0.03       # (m)
stats-stdtrop      =0.3        # (m)
stats-prnaccelh    =10         # (m/s^2)
stats-prnaccelv    =10         # (m/s^2)
stats-prnbias      =0.0001     # (m)
stats-prniono      =0.001      # (m)
stats-prntrop      =0.0001     # (m)
stats-prnpos       =0          # (m)
stats-clkstab      =5e-12      # (s/s)
ant1-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant1-pos1          =90         # (deg|m)
ant1-pos2          =0          # (deg|m)
ant1-pos3          =-6335367.62849036 # (m|m)
ant1-anttype       =
ant1-antdele       =0          # (m)
ant1-antdeln       =0          # (m)
ant1-antdelu       =0          # (m)
ant2-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant2-pos1          =90         # (deg|m)
ant2-pos2          =0          # (deg|m)
ant2-pos3          =-6335367.62849036 # (m|m)
ant2-anttype       =
ant2-antdele       =0          # (m)
ant2-antdeln       =0          # (m)
ant2-antdelu       =0          # (m)
misc-timeinterp    =off        # (0:off,1:on)
misc-sbasatsel     =0          # (0:all)
misc-rnxopt1       =
misc-rnxopt2       =
misc-pppopt        =
file-satantfile    =
file-rcvantfile    =
file-staposfile    =
file-geoidfile     =
file-ionofile      =
file-dcbfile       =
file-eopfile       =
file-blqfile       =
file-tempdir       =
file-geexefile     =
file-solstatfile   =
file-tracefile     =
```

B.2. Differential Positioning

B.2.1. Pseudo range differences

rtklib_dgps.config

```
# rtkpost_qt options (2020/03/16 01:18:30, v.2.4.3 b8)

pos1-posmode       =dgps       # (0:single,1:dgps,2:kinematic,3:static,4:movingbase,5:fixed,
                               # 6:ppp-kine,7:ppp-static,8:ppp-fixed)
pos1-frequency     =l1+l2      # (1:l1,2:l1+l2,3:l1+l2+l5,4:l1+l5)
pos1-soltype       =forward    # (0:forward,1:backward,2:combined)
pos1-elmask        =15         # (deg)
pos1-snrmask_r     =off        # (0:off,1:on)
pos1-snrmask_b     =off        # (0:off,1:on)
pos1-snrmask_L1    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L2    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L5    =0,0,0,0,0,0,0,0,0
pos1-dynamics      =off        # (0:off,1:on)
pos1-tidecorr      =off        # (0:off,1:on,2:otl)
pos1-ionoopt       =off        # (0:off,1:brdc,2:sbas,3:dual-freq,4:est-stec,5:ionex-tec,
                               # 6:qzs-brdc,7:qzs-lex,8:stec)
pos1-tropopt       =off        # (0:off,1:saas,2:sbas,3:est-ztd,4:est-ztdgrad,5:ztd)
pos1-sateph        =brdc       # (0:brdc,1:precise,2:brdc+sbas,3:brdc+ssrapc,4:brdc+ssrcom)
pos1-posopt1       =off        # (0:off,1:on)
pos1-posopt2       =off        # (0:off,1:on)
pos1-posopt3       =off        # (0:off,1:on,2:precise)
pos1-posopt4       =off        # (0:off,1:on)
pos1-posopt5       =off        # (0:off,1:on)
pos1-posopt6       =off        # (0:off,1:on)
pos1-exclsats      =           # (prn ...)
pos1-navsys        =1          # (1:gps+2:sbas+4:glo+8:gal+16:qzs+32:comp)
pos2-armode        =continuous # (0:off,1:continuous,2:instantaneous,3:fix-and-hold)
pos2-gloarmode     =on         # (0:off,1:on,2:autocal)
pos2-bdsarmode     =on         # (0:off,1:on)
pos2-arthres       =3
pos2-arthres1      =0.9999
pos2-arthres2      =0.25
pos2-arthres3      =0.1
pos2-arthres4      =0.05
pos2-arlockcnt     =0
pos2-arelmask      =0          # (deg)
pos2-arminfix      =10
pos2-armaxiter     =1
pos2-elmaskhold    =0          # (deg)
pos2-aroutcnt      =5
pos2-maxage        =30         # (s)
pos2-syncsol       =off        # (0:off,1:on)
pos2-slipthres     =0.05       # (m)
pos2-rejionno      =30         # (m)
pos2-rejgdop       =30
pos2-niter         =1
pos2-baselen       =0          # (m)
pos2-basesig       =0          # (m)
out-solformat      =xyz        # (0:llh,1:xyz,2:enu,3:nmea)
out-outhead        =on         # (0:off,1:on)
out-outopt         =on         # (0:off,1:on)
out-timesys        =gpst       # (0:gpst,1:utc,2:jst)
out-timeform       =hms        # (0:tow,1:hms)
out-timendec       =3
out-degform        =deg        # (0:deg,1:dms)
out-fieldsep       =
out-height         =ellipsoidal # (0:ellipsoidal,1:geodetic)
out-geoid          =internal   # (0:internal,1:egm96,2:egm08_2.5,3:egm08_1,4:gsi2000)
out-solstatic      =all        # (0:all,1:single)
out-nmeaintv1      =0          # (s)
out-nmeaintv2      =0          # (s)
out-outstat        =residual   # (0:off,1:state,2:residual)
stats-eratio1      =100
stats-eratio2      =100
stats-errphase     =0.003      # (m)
```

```
stats-errphaseel   =0.003      # (m)
stats-errphasebl   =0          # (m/10km)
stats-errdoppler   =10         # (Hz)
stats-stdbias      =30         # (m)
stats-stdiono      =0.03       # (m)
stats-stdtrop      =0.3        # (m)
stats-prnaccelh    =10         # (m/s^2)
stats-prnaccelv    =10         # (m/s^2)
stats-prnbias      =0.0001     # (m)
stats-prniono      =0.001      # (m)
stats-prntrop      =0.0001     # (m)
stats-prnpos       =0          # (m)
stats-clkstab      =5e-12      # (s/s)
ant1-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant1-pos1          =90         # (deg|m)
ant1-pos2          =0          # (deg|m)
ant1-pos3          =-6335367.62849036 # (m|m)
ant1-anttype       =
ant1-antdele       =0          # (m)
ant1-antdeln       =0          # (m)
ant1-antdelu       =0          # (m)
ant2-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant2-pos1          =51.9972020218877 # (deg|m)
ant2-pos2          =4.37728299290437 # (deg|m)
ant2-pos3          =42.4305072547868 # (m|m)
ant2-anttype       =
ant2-antdele       =0          # (m)
ant2-antdeln       =0          # (m)
ant2-antdelu       =0          # (m)
misc-timeinterp    =off        # (0:off,1:on)
misc-sbasatsel     =0          # (0:all)
misc-rnxopt1       =
misc-rnxopt2       =
misc-pppopt        =
file-satantfile    =
file-rcvantfile    =
file-staposfile    =
file-geoidfile     =
file-ionofile      =
file-dcbfile       =
file-eopfile       =
file-blqfile       =
file-tempdir       =
file-geexefile     =
file-solstatfile   =
file-tracefile     =
```

B.2.2. Carrier phase differences

rtklib_kinematic.conf

```
# rtkpost_qt options (2020/05/25 11:54:00, v.2.4.3 b8)

pos1-posmode       =kinematic  # (0:single,1:dgps,2:kinematic,3:static,4:movingbase,5:fixed,
                               # 6:ppp-kine,7:ppp-static,8:ppp-fixed)
pos1-frequency     =l1+l2      # (1:l1,2:l1+l2,3:l1+l2+l5,4:l1+l5)
pos1-soltype       =forward    # (0:forward,1:backward,2:combined)
pos1-elmask        =15         # (deg)
pos1-snrmask_r     =off        # (0:off,1:on)
pos1-snrmask_b     =off        # (0:off,1:on)
pos1-snrmask_L1    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L2    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L5    =0,0,0,0,0,0,0,0,0
pos1-dynamics      =off        # (0:off,1:on)
pos1-tidecorr      =off        # (0:off,1:on,2:otl)
pos1-ionoopt       =off        # (0:off,1:brdc,2:sbas,3:dual-freq,4:est-stec,5:ionex-tec,
                               # 6:qzs-brdc,7:qzs-lex,8:stec)
pos1-tropopt       =off        # (0:off,1:saas,2:sbas,3:est-ztd,4:est-ztdgrad,5:ztd)
pos1-sateph        =brdc       # (0:brdc,1:precise,2:brdc+sbas,3:brdc+ssrapc,4:brdc+ssrcom)
```

```
pos1-posopt1       =off        # (0:off,1:on)
pos1-posopt2       =off        # (0:off,1:on)
pos1-posopt3       =off        # (0:off,1:on,2:precise)
pos1-posopt4       =off        # (0:off,1:on)
pos1-posopt5       =off        # (0:off,1:on)
pos1-posopt6       =off        # (0:off,1:on)
pos1-exclsats      =           # (prn ...)
pos1-navsys        =1          # (1:gps+2:sbas+4:glo+8:gal+16:qzs+32:comp)
pos2-armode        =continuous # (0:off,1:continuous,2:instantaneous,3:fix-and-hold)
pos2-gloarmode     =on         # (0:off,1:on,2:autocal)
pos2-bdsarmode     =on         # (0:off,1:on)
pos2-arthres       =3
pos2-arthres1      =0.9999
pos2-arthres2      =0.25
pos2-arthres3      =0.1
pos2-arthres4      =0.05
pos2-arlockcnt     =0
pos2-arelmask      =0          # (deg)
pos2-arminfix      =10
pos2-armaxiter     =1
pos2-elmaskhold    =0          # (deg)
pos2-aroutcnt      =5
pos2-maxage        =30         # (s)
pos2-syncsol       =off        # (0:off,1:on)
pos2-slipthres     =0.05       # (m)
pos2-rejionno      =30         # (m)
pos2-rejgdop       =30
pos2-niter         =1
pos2-baselen       =0          # (m)
pos2-basesig       =0          # (m)
out-solformat      =xyz        # (0:llh,1:xyz,2:enu,3:nmea)
out-outhead        =on         # (0:off,1:on)
out-outopt         =on         # (0:off,1:on)
out-timesys        =gpst       # (0:gpst,1:utc,2:jst)
out-timeform       =hms        # (0:tow,1:hms)
out-timendec       =3
out-degform        =deg        # (0:deg,1:dms)
out-fieldsep       =
out-height         =ellipsoidal # (0:ellipsoidal,1:geodetic)
out-geoid          =internal   # (0:internal,1:egm96,2:egm08_2.5,3:egm08_1,4:gsi2000)
out-solstatic      =all        # (0:all,1:single)
out-nmeaintv1      =0          # (s)
out-nmeaintv2      =0          # (s)
out-outstat        =residual   # (0:off,1:state,2:residual)
stats-eratio1      =100
stats-eratio2      =100
stats-errphase     =0.003      # (m)
stats-errphaseel   =0.003      # (m)
stats-errphasebl   =0          # (m/10km)
stats-errdoppler   =10         # (Hz)
stats-stdbias      =30         # (m)
stats-stdiono      =0.03       # (m)
stats-stdtrop      =0.3        # (m)
stats-prnaccelh    =10         # (m/s^2)
stats-prnaccelv    =10         # (m/s^2)
stats-prnbias      =0.0001     # (m)
stats-prniono      =0.001      # (m)
stats-prntrop      =0.0001     # (m)
stats-prnpos       =0          # (m)
stats-clkstab      =5e-12      # (s/s)
ant1-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant1-pos1          =90         # (deg|m)
ant1-pos2          =0          # (deg|m)
ant1-pos3          =-6335367.6285 # (m|m)
ant1-anttype       =
ant1-antdele       =0          # (m)
ant1-antdeln       =0          # (m)
ant1-antdelu       =0          # (m)
ant2-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant2-pos1          =51.997202022 # (deg|m)
ant2-pos2          =4.377282993 # (deg|m)
```

```
ant2-pos3          =42.4304999988526 # (m|m)
ant2-anttype       =
ant2-antdele       =0          # (m)
ant2-antdeln       =0          # (m)
ant2-antdelu       =0          # (m)
misc-timeinterp    =off        # (0:off,1:on)
misc-sbasatsel     =0          # (0:all)
misc-rnxopt1       =
misc-rnxopt2       =
misc-pppopt        =
file-satantfile    =
file-rcvantfile    =
file-staposfile    =
file-geoidfile     =
file-ionofile      =
file-dcbfile       =
file-eopfile       =
file-blqfile       =
file-tempdir       =
file-geexefile     =
file-solstatfile   =
file-tracefile     =
```

Bibliography

[1] RTK (GNSS) 1-2 cm | 06-GPS. URL https://www.06-gps.nl/rtk-gnss-1-2-cm/. Library Catalog: www.06-gps.nl.

[2] Above Us Only Stars. URL https://www.c4reports.org/aboveusonlystars. Library Catalog: www.c4reports.org.

[3] GPS Navigation Message - Navipedia. URL https://gssc.esa.int/navipedia/index.php/GPS_Navigation_Message.

[4] *Networked Transport of RTCM via Internet Protocol (Ntrip)* , 1 edition. [Online; accessed March-2020].

[5] Beacon Company of Egypt - Egypt Marine DGPS, . URL http://www.beacon-egypt.com/dgps.htm.

[6] DGNSS Standards - Navipedia, . URL https://gssc.esa.int/navipedia/index.php/DGNSS_Standards.

[7] RTKLIB: An Open Source Program Package for GNSS Positioning, . URL http://www.rtklib.com/.

[8] Secret Key Exchange (Diffie-Hellman) - Computerphile, . URL https://www.youtube.com/watch?v=NmM9HA2MQGI.

[9] RTCMV3 Standard Logs. URL https://docs.novatel.com/OEM7/Content/Logs/RTCMV3_Standard_Logs.htm?TocPath=Logs%7CAll%20Logs%7CGNSS%20Logs%7C_____155.

[10] RTCM 2 Message List - SNIP Support, . URL https://www.use-snip.com/kb/knowledge-base/rtcm-2-message-list/.

[11] An RTCM 3 message cheat sheet, . URL https://www.use-snip.com/kb/knowledge-base/an-rtcm-message-cheat-sheet/. Library Catalog: www.use-snip.com.

[12] GPS.gov: GPS Accuracy. URL https://www.gps.gov/systems/gps/performance/accuracy/.

[13] Glossary Trimble website. URL https://www.trimble.com/ec_receiverhelp/v4.15/en/Glossary.htm.

[14] What is SBAS?, March 2016. URL https://www.gsa.europa.eu/european-gnss/what-gnss/what-sbas. Library Catalog: www.gsa.europa.eu.

[15] Executing a Man-in-the-Middle Attack in just 15 Minutes - Hashed Out, November 2018. URL https://www.thesslstore.com/blog/man-in-the-middle-attack-2/. Library Catalog: www.thesslstore.com Section: Hashing Out Cyber Security.

[16] Jahshan Bhatti and Todd E Humphreys. Hostile control of ships via false GPS signals: Demonstration and detection. *NAVIGATION: Journal of the Institute of Navigation*, 64(1):51–66, 2017.

[17] Alexey Boriskin, Dmitry Kozlov, and Gleb Zyryanov. The RTCM Multiple Signal Messages: A New Step in GNSS Data Standardization. In *Proceedings of the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012)*, pages 2947–2955, 2012.

[18] Roel Bree and C.C.J.M. Tiberius. Real-time single-frequency precise point positioning: Accuracy assessment. *GPS Solutions*, 16:259–266, 04 2011. doi: 10.1007/s10291-011-0228-6.

[19] Werner Gurtner. RINEX: The Receiver Independent Exchange Format Version 2.10. 2001.

[20] GJ Husti. *Global Positioning System*. Delft University Press, 2000.

[21] Seongkyun Jeong, Minchan Kim, and Jiyun Lee. CUSUM-based GNSS Spoofing Detection Method for Users of GNSS Augmentation System. *International Journal of Aeronautical and Space Sciences*, 21(2):513–523, June 2020. ISSN 2093-274X, 2093-2480. doi: 10.1007/s42405-020-00272-9. URL http://link.springer.com/10.1007/s42405-020-00272-9.

[22] Steven Lewis, Logan Maynard, C. Edward Chow, and Dennis Akos. Secure GPS data for critical infrastructure and key resources: Cross-layered integrity processing and alerting service. *Navigation*, 65(3):389–403, 2018. doi: 10.1002/navi.251. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/navi.251.

[23] Barend Lubbers. GNSS Spoofingen en Detectie, 12 2019. URL http://www.navnin.nl/new/wp-content/uploads/2020/01/2019_HSB_2_Workshop-Spoofing-by-Barend-Lubbers-KIM.pdf.

[24] Alejandro N. Mayorkas. Department Policy Regarding the Use of Cell-Site Simulator Technology. *POLICY DIRECTIVE 047-02*, 2015.

[25] Ulrike Meyer and Susanne Wetzel. A man-in-the-middle attack on UMTS. In *Proceedings of the 2004 ACM workshop on Wireless security - WiSe '04*, page 90, Philadelphia, PA, USA, 2004. ACM Press. ISBN 978-1-58113-925-9. doi: 10.1145/1023646.1023662. URL http://portal.acm.org/citation.cfm?doid=1023646.1023662.

[26] Gopi Nath Nayak and Shefalika Ghosh Samaddar. Different flavours of Man-In-The-Middle attack, consequences and feasible solutions. In *2010 3rd International Conference on Computer Science and Information Technology*, pages 491–495, Chengdu, China, July 2010. IEEE. ISBN 978-1-4244-5537-9. doi: 10.1109/ICCSIT.2010.5563900. URL http://ieeexplore.ieee.org/document/5563900/.

[27] Panagiotis Papadimitratos and Aleksandar Jovanovic. GNSS-based positioning: Attacks and countermeasures. In *MILCOM 2008-2008 IEEE Military Communications Conference*, pages 1–7. IEEE, 2008.

[28] Mark L. Psiaki and Todd E. Humphreys. GNSS Spoofing and Detection. *Proceedings of the IEEE*, 104(6):1258–1270, June 2016. ISSN 0018-9219, 1558-2256. doi: 10.1109/JPROC.2016.2526658. URL http://ieeexplore.ieee.org/document/7445815/.

[29] Yasith Ramawickrama, Jude Vijayanga, Rohan Dharmarathne, and Hiruni Wijesooriya. The Future of GNSS in the Next Ten years. page 23, 2016.

[30] Logan Scott and LS Consulting. Anti-Spoofing & Authenticated Signal Architectures for Civil Navigation Systems. page 10.

[31] Mike Strutt. RTCM3.x or CMR+ – What should I use?

[32] T. Takasu. *RTKLIB ver. 2.4.2 Manual*. RTKlib, 2 edition.

[33] P J G Teunissen. A canonical theory for short GPS baselines. page 17.

[34] Peter Teunissen and Oliver Montenbruck. *Springer handbook of global navigation satellite systems*. Springer, 2017.

[35] P.J.G. Teunissen. To Be or Not To Be Foold, That Is The Question: Some Theoretical Considerations, 2019. [Internal technical note].

[36] P.J.G. Teunissen, D.G. Simons, C.C.J.M. Tiberius, and Faculty of Aerospace Engineering Delft University of Technology. *Probability and Observation Theory;AE2-E01: Exercises*. TU Delft, 2008. URL https://books.google.nl/books?id=E1pBnQEACAAJ.

[37] C.C.J.M. Tiberius, R.J.P. van Bree, and P.J. Buist. Staying in lane: real-time single-frequency ppp on the road. *Proceedings of inside GNSS (November/December)*, pages 48–53, 2011.

[38] Christian C.J.M. Tiberius. Recursive data processing for kinematic gps surveying. *Publications on Geodesy, 45*, 1998. doi: 90-804147-1-9. URL http://resolver.tudelft.nl/uuid:f218e8c4-25d2-4793-8762-7500998b22a4.

[39] Kexiong Curtis Zeng, Yuanchao Shu, Shinan Liu, Yanzhi Dou, and Yaling Yang. A practical GPS location spoofing attack in road navigation scenario. In *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications*, pages 85–90. ACM, 2017.