Next Generation Precise Satellite Positioning

Analysis of advanced positioning techniques with multiple navigation satellite systems using Curtin's PPP-RTK User Platform

N. Treffers

Master Thesis
Geoscience and Remote Sensing

TU Delft
Curtin University

Next Generation Precise Satellite Positioning

Analysis of advanced positioning techniques with multiple navigation satellite systems using Curtin's PPP-RTK User Platform

by

N. Treffers

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday February 24, 2016 at 11:00 AM.

Student number:	1283510	
Project duration:	February 24, 2015 – February 24, 2016	
Thesis committee:	Prof. dr. ir. P. J. G. Teunissen,	TU Delft, Curtin University
	Dr. ir. A. A. Verhagen,	TU Delft
	Dr. R. C. Lindenbergh,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Preface

Writing this thesis marks the final step of the master track Geoscience and Remote Sensing, which is the final stretch of my time as a student at TU Delft. Though the courses in the master track did not focus only on satellite navigation, both my internship and thesis did. As a field of research it is larger than it seemed when I started. On one hand new techniques to use the satellite signals are still being developed and improved. On the other hand more and more satellites become available every year. Using all available data becomes more and more complex. Getting the theory to work in practice, like is done in the user platform that is tested here, is yet another step. Even then one has to consider that these tests are still limited. Having a software tool that works in the field, real-time and in poor conditions is a long step from here. In reality there will always be a scenario possible in which problems arise, no matter what software is used. Such is the nature of software.

But before anything can be put into use it must be tested. Providing a helping hand in testing the new techniques as implemented in the Curtin PPP-RTK User Platform has been a great experience for me. My sincere thanks go out to professor Teunissen and Curtin University for allowing me to work on this project in Perth for 7 months. During those 7 months I spent most of my days in the GNSS Research Centre at Curtin. My colleagues there helped by answering a lot of the questions I had. Because I focused on testing software I also got to occasionally annoy them by describing the bugs I found. For the patience when I found another problem and for the great discussions we had (both on GNSS topics and the world in general) I want to thank the entire team. I guess I will miss the Friday meetings. Being on the other side of the world for this long makes me glad to have had the support from family and friends, even though most of them were living on schedules completely different from mine. I was especially glad that my grandfather managed to hang in there while I was away.

As described previously there are a lot of directions of research still open in this field. With new techniques, satellites and fields of application the options will only grow. Whether in research or application I still hope to help in some progress in the future. This thesis represents my input in the GNSS field up to this point. As a final step for completing my time as a student: I hope the reader enjoys the contents.

Summary

For ongoing research on satellite navigation the GNSS research centre at Curtin University has developed a new software platform. The Curtin PPP-RTK user platform is a software prototype, developed in Matlab, that is capable of processing observation data from Global Navigation Satellite System (GNSS) receivers worldwide. The positioning methods that can be used are Precise Point Positioning (PPP) and Integer ambiguity resolution enabled Precise Point Positioning (PPP-RTK). The user platform supports observation data from all GNSS constellations except the Russian GLOball NAvigation Satellite System (GLONASS).

The contents of this thesis are the results of testing done on the new user platform. The first step in testing the software is selecting a large number of datasets. For testing the performance of the software, using only Global Positioning System (GPS), more than 700 datasets are processed. For the newer constellations data is used from the Multi-GNSS EXperiment (MGEX). As not as many receivers provide data, only around 200 datasets are used for the multi-GNSS scenarios. These datasets have been processed with different settings to test for bugs in the software. The problems encountered have been solved, but are not detailed in the report. The performance when processing these datasets, once any bugs have been corrected, is described.

The first performance analysis is done for PPP, using only GPS. Three factors are investigated. As a measure of precision the empirical standard deviation of the position estimates is used. The precision of the horizontal components is a few centimeters for the processed datasets. The up component is weaker, but is still under the 1 decimeter level. The precision is therefore as is expected for PPP using only GPS.

The estimated position is also compared to externally provided ground truth values, to ensure that there is no large bias in any of the three components. After processing all datasets the combined results indeed show distributions that are centered close to zero. The third factor for the performance is the convergence time. Though the time it takes for the solution to converge widely differs per scenario, the majority of the datasets converge within 30 minutes.

The next step is including more constellations, still using PPP. The Chinese BeiDou provides a slight increase in precision when used in combination with GPS. The increase in precision is highly dependent on the number of satellites and therefore, for BeiDou, on the geographical location of the receiver. The European Galileo constellation shows improvement globally, but the effect is minimal due to a limited number of satellites. The Japanese Quazi-Zenith Satellite System (QZSS) is tested, but the effects are small as the constellation only consists of a single satellite at this point.

The convergence time for combinations of constellations using PPP is investigated as well. The improvement for BeiDou is minimal, as the geostationary satellites do not greatly benefit the convergence of the solution. The limited number of Galileo satellites also has no large effect on the convergence time. The extra satellites must be in view at the start of the processed data to affect the convergence time. With only a few Galileo satellites this is not the case for most datasets.

The third testing step is PPP-RTK using only GPS. In the future the additional corrections required for PPP-RTK will be available from large, or global, networks. For testing the user platform the corrections used have been generated from only single stations. As the range between user and reference station is limited by satellites that are in common view, PPP-RTK is tested for different inter-station distances. For a selection of datasets forming pairs over 200, 500 and 1000 kilometers the convergence time is compared with and without ambiguity resolution. For all the distances the convergence time decreases significantly. After ambiguity resolution the majority of the datasets show convergence time under 10 minutes.

The last step is testing PPP-RTK using combinations of GPS, BeiDou and Galileo. The number of pairs that can be formed for multi-GNSS testing is limited, due to the restrictions on inter-station distances. Therefore datasets are used multiple time, forcing the software to start processing at different points in the datasets. Different segments of the datasets are selected, based on the number of satellites available at the start of processing.

When a large number of GPS satellites is available the solution, for most of the datasets, convergences in 10 minutes. Adding 3 Galileo satellites pushes the convergence time down to 5 minutes. Using GPS and BeiDou gives an improvement down to 2 minutes. Combining all three constellations pushes this down even further to 1 minute. This all is the case with sufficient satellites: approximately 10 for GPS and BeiDou and 3 for Galileo.

At a different starting point in the datasets, with fewer GPS satellites available, the convergence time is longer. Using only 6 GPS satellites leads to convergence times around 60 minutes. Adding the Galileo constellation is still beneficial as the convergence time is cut to a third. Using all available satellites in this less optimal scenario pushes the convergence time to 10 minutes, still indicating a large benefit for multi-GNSS PPP-RTK.

For both PPP and PPP-RTK the benefit of adding more constellations is shown. Since not all constellations are complete, this is only an indication of the expected performance once all satellites are available. For PPP-RTK the user platform is to be used for future research and testing of network corrections, when these become available. After all performed tests the user platform is deemed ready for its purpose: a testing tool to be used for further research on PPP and PPP-RTK making use of any GNSS available.

List of Abbreviations

CDMA	Code Division Multiple Access
CNES	Centre National d'Etudes Spatiales
DIA	Detection Identification and Adaptation
FDMA	Frequency Division Multiple Access
GIM	Global Ionosphere Maps
GLONASS	GLOball NAvigation Satellite System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IGS	International GNSS Service
IRNSS	Indian Regional Navigation Satellite System
ISB	Inter System Bias
LAMBDA	Least-squares Ambiguity Decorrelation Adjustment
LOM	Local Overall Model
MGEX	Multi-GNSS EXperiment
PCO	Phase Center Offset
PCV	Phase Center Variation
PPP	Precise Point Positioning
PPP-RTK	Integer ambiguity resolution enabled Precise Point Positioning
QZSS	Quazi-Zenith Satellite System
RINEX	Receiver INdependent EXchange
RTK	Real-Time Kinematic
SPP	Single Point Positioning
TEC	Total Electron Content
UTC	Universal Coordinated Time
ZTD	Zenith Tropospheric Delay

Contents

1 Introduction

In recent years the development of positioning techniques using satellites has been progressing in two directions. On one hand newer techniques have been devised that allow more precise and faster positioning using the long existing Global Positioning System (GPS). On the other hand more and more satellites are becoming available as new Global Navigation Satellite System (GNSS) constellations are being deployed. The Curtin GNSS Research Group has been developing its own in-house positioning software used for GNSS research on new positioning techniques while making use of all available constellations. This software is known as the Curtin PPP-RTK User Platform. The user platform uses a novel undifferenced approach for both Precise Point Positioning (PPP) and Integer ambiguity resolution enabled Precise Point Positioning (PPP-RTK). This approach has been developed as part of a research project that finished in 2015. The platform, being the implementation of this approach, was also part of the research project and is to be used for future research as well.

As the software has only recently been put together it has not been thoroughly tested yet. In order to make sure that the software remains functional under all circumstances large scale testing is required. Once any occurring problems have been handled the results that have been produced must be analyzed. This is to see how well the user platform performs and if the undifferenced model does indeed function as expected. The user platform is intended to be used for further research. By investigating the improvements that can be achieved by using the newer constellations a first step is made in using the platform for further research. This combines further testing of the user platform with analyzing the performance of the GNSS constellations. The three main research goals for this report are based on these observations.

The first part of the research is focused on the functionality of the software. Different input files are used by the software. This includes the observation data recorded by receivers contained in Receiver INdependent EXchange (RINEX) observation files. Additionally corrections for the satellite's orbit and on-board clock are used. This data is available in varying formats, leading to the first main research question: *What different types of observation and correction files can Curtin's PPP-RTK user platform successfully process?* The term 'successfully' is included as the external data files that are used can contain unexpected fields or miss information. These situations can cause the software to behave incorrectly. In this case modifications to the code have been made to capture such problems.

The second part of the research is centered on the performance of the software. For the different positioning techniques the performance is expected to be different. First is the precision of the results, indicating how stable a solution is over time. The accuracy is part of the analysis as high precision alone does not guarantee that the estimated position is actually correct. Lastly the time it takes to get to a precise solution is of importance. As the positioning techniques can be used to get more precise results over time this convergence time is also investigated. The second main research question is therefore: *What precision, accuracy and convergence time can be achieved under different conditions using the positioning techniques available in the user platform?* This question includes the aspects of defining what the precision, accuracy and convergence time actually are and how to measure them from the results produced. The results for these factors, using different constellations and positioning techniques, are also compared to literature. As long as comparison results are actually available from other sources for a tested scenario.

The third and last part is focused on new research done using the software. While testing with the newer constellations is also part of testing the functionality and performance of the software, these options are also quite new. As new satellites are still being launched the performance of PPP with multiple constellations must still be investigated. Even more so for PPP-RTK in the case of multi-GNSS. The third research question is: *How does the performance of PPP and PPP-RTK change if the newer GNSS constellations are used and what problems still exist when processing this multi-GNSS data?* Expecting only an increase in performance is not realistic at this stage as many of the newer constellations are still incomplete. Also problems exist that cannot yet be overcome, but these are addressed as well.

The datasets used for analysis and testing are all publicly available. Observation data is used from either the International GNSS Service (IGS) or from the receivers present at Curtin University. This combination provides a globally distributed set of receivers as well as a few local regions with larger numbers of receivers close together. For orbit and clock corrections either the combined products from IGS are used or the products provided by the individual analysis centers. As all these datasets are available on a daily basis, tests are performed for different years and different parts of the year. For the multi-GNSS testing more recent data is preferred as more satellites are available.

This thesis has been structured as follows. First a theoretical background is provided. This includes a description of the satellite constellations that are used and what signals these satellites provide. Secondly the background also describes what is measured and what is estimated using different positioning techniques. This is followed by a description of the test plan used to test the software's functionality and analyze the output that is generated.

The results from these test are separated into two parts for different positioning methods. First the results for standard PPP are elaborated on. This is first done for GPS and later expanded to the newer constellations. The second results section is dedicated to PPP-RTK. Again this is done first for only GPS and later for the others as well. Because the PPP-RTK technique is newer and has some limitations, this chapter also describes some encountered problems and restrictions that still apply at this time.

Following the results section are the conclusions that attempt to answer the research questions posed herein. After this follow the recommendations for further research based on issues encountered during testing and research question that remained partially unanswered after the testing and analysis. The appendices contain additional information referenced in other sections of the thesis, but also contain description of software problems that have been encountered and resolved. These problems have been part of answering some of the research questions posed, but as the issues no longer exist they are not a part of the main section of the report.

2

Global Navigation Satellite Systems

This chapter consists of background information on subjects used throughout the remainder of the report. First a short description is provided on all constellations used for testing the PPP-RTK user platform. This description contains the frequencies used by the satellites and the orbit types used in the constellations. Successively the types of observations available in the observation data are described.

The chapter continues on several disturbances that are present in the observation data. This list is not complete, but refers to those that are of importance in later chapters. Afterwards a description of the positioning techniques used by the software is provided. Finally also a short description of the user platform itself is provided.

2.1. GNSS Constellations

The number of GNSSs available is ever increasing. The following section provides a brief description of each of the constellations. As the specific signals transmitted by the satellites in each of these constellations are important for processing, these are also listed for each constellation.

2.1.1. GPS

Of all GNSSs currently in use, GPS has been operational the longest. The system has been fully operational since 1995. A fully operational constellation consists of a minimum of 27 satellites (before 2011 this was 24 [4]). With satellites being added and replaced the total number can go up to 32. All the satellites are in a circular orbit approximately 20,200 km above the Earth's surface. The satellites are divided over six orbit planes each with an inclination of 55 degrees. The distribution of the satellites in the orbit planes is chosen to provide global coverage, while having improved coverage in specific regions on the planet [4].

All current GPS satellites broadcast signals on at least two frequencies (L1 and L2). Newer satellites in the constellation also provide a new third frequency (L5). With the current planning all satellites will provide the third signal in 2020. Table 2.1 provides the available frequencies and corresponding wavelengths for GPS.

Signal	Band	frequency [MHz]	wavelength [cm]
L1	L1	1575.42	19.0
L2	L2	1227.60	24.4
L5	L5	1176.45	25.5

Table 2.1: Signals available for the GPS constellation. Bands, frequencies and wavelengths for each of the signals.

2.1.2. GLONASS

The Russian GLOball NAvigation Satellite System (GLONASS) is the only constellation next to GPS to have become fully operational. Currently 24 satellites are active in three different orbital planes [7]. The satellites orbit the Earth approximately 19,130 km above the surface. The GLONASS satellites use two different base frequencies, however every satellite may have a different small offset from one another. Usually two satellites in the constellation use exactly the same frequency, but these can never be in view of a user at the same time. This offset in frequency is used to distinguish the satellite by the user. This method in distinguishing between satellites is Frequency Division Multiple Access (FDMA). Other constellations use Code Division Multiple Access (CDMA) which distinguishes satellites by the code that is transmitted. With the current planning all Glonass satellites are expected to be broadcasting a new CDMA signal as well around the year 2020.

2.1.3. BeiDou

The Chinese GNSS BeiDou (previously named Compass) has not yet fully been completed. The constellation consists of satellites in three different orbit types. The first orbit type is a geostationary orbit. Because these satellites complete one revolution of the Earth in the same time as the planet fully rotates they are constantly covering the same area. This does mean the satellites are present over the equator, but the coverage is centered on the mainland of China. The second part of the constellation consists of satellites with an inclined geosynchronous orbit. These satellite have the same orbital period as the geostationary ones but have a non-zero degree inclination. In longitudinal direction these satellites hold more or less the same position while they do significantly change in latitude. The result is that these satellites do for part of the day fully cover the mainland of China. As a side effect the satellites have a similar ground track South of the equator, making them useful in Australia as well. Figure 2.1 depicts the ground tracks for these two satellite types.

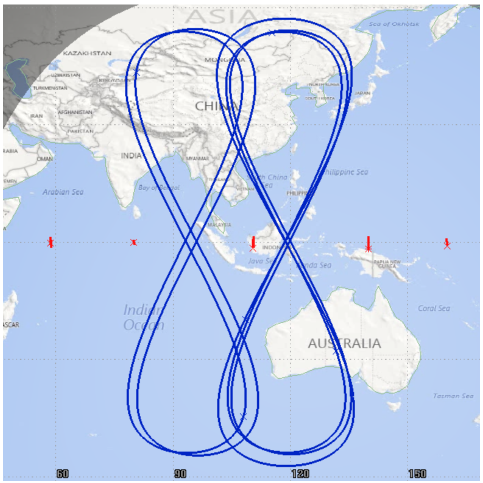


Figure 2.1: Ground tracks for BeiDou geostationary (red) and inclined geosynchronous orbits (blue). (Source: AGI Systems Toolkit)

The third orbit type is the normal medium Earth orbit. These satellite have orbit similar to GPS and GLONASS and therefore provide coverage over the entire world during the day. The orbital altitude for the medium Earth orbit satellites is 21,528 km. As the constellation is not yet completed, and especially the number of medium Earth orbit satellites is lacking, the system can only effectively be used in the region around China. The BeiDou satellites make use of three different frequencies, listed in table 2.2.

Signal	Band	frequency [MHz]	wavelength [cm]
B1	L2	1561.098	19.2
B2	L7	1207.14	24.8
B3	L6	1268.52	23.6

Table 2.2: Signals available for the BeiDou constellation. Bands, frequencies and wavelengths for each of the signals. [5]

2.1.4. GALILEO

The Galileo constellation is the European GNSS. The first test satellites were launched in 2005 and 2008, but have since then been decommissioned. From 2011 on the operational segment of the constellation is being deployed. As of December 2015 a total of 12 Galileo satellites are in orbit. The completed constellation will have a total of 30 satellites which includes spare satellites present for each orbit plane [10]. The orbits used are medium Earth orbits with an orbital altitude of 23,222 km [3]. Due to an anomaly during launch satellites 5 and 6 are not in this specified orbit. Despite this the two satellites have been activated and can be tracked by receivers. The full constellation is planned to be complete around 2020.

The satellites in the Galileo constellation transmit navigation signals on four frequencies. The list of frequencies and wavelengths for the signals are provided in table 2.3. Of special interest in this case are the E1 and E5a, which overlap with the GPS L1 and L5 frequencies. Also the E5b signal has an overlap with the BeiDou B2 signal. This is a useful fact that can be exploited for improved positioning performance. It is possible to treat the signals from different constellations on overlapping frequencies as if they all belong to the same constellation.

Signal	Band	frequency [MHz]	wavelength [cm]
E1	L1	1575.42	19.0
E5a	L5	1176.45	25.5
E5b	E5	1207.14	24.8
E6	L6	1278.75	23.4

Table 2.3: Signals available for the Galileo constellation. Bands, frequencies and wavelengths for each of the signals.

2.1.5. QZSS

The Quazi-Zenith Satellite System (QZSS) was developed specifically for use in Japan and is therefore a regional system. In cities with a lot of high-rise buildings visibility of low-elevation satellites is problematic. QZSS was commissioned specifically to assist with positioning in environments where only high elevation satellites can be used. The complete constellation will consist of at least three satellites in a highly elliptical orbit, but as of 2015 only a single satellite has been launched. The elliptical orbits are designed so that a satellite is visible at a high elevation above Japan for a large part of the day. The apogee of the orbit is therefore above Japan. Even though the satellite is furthest away from the Earth at this point it is also moving more slowly, which increases the duration of its stay above Japan. The lower segment of the orbit cause the satellites also to be visible south of the equator in for instance Australia. However as the satellite is passing through the perigee the movement is much faster. The ground track for the current QZSS satellite is shown in figure 2.2.

As the constellation is primarily designed to function in combination with GPS the signals are chosen to use similar bands. The available frequencies are provided in table 2.4. The fourth frequency specified is LEX, which experimental and in the future will not be publicly available.

Signal	Band	frequency [MHz]	wavelength [cm]
L1	L1	1575.42	19.0
L2	L2	1227.6	24.4
L5	L5	1176.45	25.5
LEX	E6	1278.75	23.4

Table 2.4: Signals available for the QZSS constellation. Bands, frequencies and wavelengths for each of the signals.

Figure 2.2: Ground track for the QZSS satellite. (Source: AGI Systems Toolkit)

2.1.6. IRNSS

The Indian Regional Navigation Satellite System (IRNSS) is the newest of the constellations currently operating. This regional system is designed to be used mainly in India. The designed constellation will, upon completion, consist of 7 satellites. Three of these will be in a geostationary orbit positioned close to India. The remaining four satellites are two pairs split over two inclined geosynchronous orbits. All 7 satellites will be visible continuously in India, but other countries can make use of some of the satellites during parts of the day as well. The satellites broadcast signal on only a single L-band frequency, the L5. This allows the use of this constellation in combination with other systems using this frequency (GPS, Galileo and QZSS). The satellites also transmit an S-band signal, but none of the widely used receivers can track this signal at this point. As of July 2015 four of the seven satellites have been successfully launched. The constellation is planned to be completed in 2016.

2.2. Observations

For each frequency used by a GNSS two quantities a receiver can observe are used. These observations concern the code being broadcast and the phase of the signal. The actual location to which these observation refer, the phase center of the antenna, is also of importance when using any observable.

2.2.1. Code observations

A satellite is continuously broadcasting a repeating signal on each of the frequencies in use. At the same time the receiver generates the same bit of code internally. This is then compared to the received signal allowing the time difference between the signal being broadcast and the signal being received to be computed. This time difference multiplied by the speed of light becomes the range. This range does not take into account several factors. A dominant factor is for instance the accuracy of the receiver clock. This factor cannot be neglected, as due to the speed of light a small inaccuracy causes a large change in the range, but is not corrected in the observation itself. For that reason the name pseudorange is used for these code observations.

The main code being broadcast is pseudorandom noise, but in addition to this also information is transmitted. Part of this information is the broadcast ephemeris, which indicates the approximate satellite position. This information is required to be able to estimate the position, without the support of external products.

2.2.2. Phase observations

Receivers are also capable of tracking the phase of the signal received. The precision of these measurements is a lot better than the code observations. The formal precision of the phase measurement is about a factor 100 better than the code measurements. The phase of the received signal is compared to an internally generated signal. The only information available is the difference in phase between these two signals. Nothing is known on the number of full cycles of the signal between the satellite and the receiver. However if the receiver keeps tracking the satellite the change in number of cycles between epochs can be determined. The number of total cycles indicated by the receiver remains arbitrary. Some receivers may start at zero allowing for both negative and positive phase observations (depending on satellite's movement after first being tracked). Other receivers can start at higher values to prevent negative values for the phase measurements. Since the number of cycles the signal has completed from transmission to reception is ambiguous none of these choices influence later results.

2.2.3. Phase center

For both the previously described observables it is important to specify the position that these observations actually refer to. If, for instance for one of the IGS reference stations, a value for the ground truth is provided, this ground truth refers to a geodetic benchmark that is fixed somewhere in the ground. The antenna used for this station will be set up with respect to this benchmark. In some cases the antenna is right on top of the benchmark, but most of the time there is some offset. The offset between the benchmark and the reference point of the antenna (usually the bottom of the antenna) is provided in the data stream from the reference station.

The observations recorded by the receiver do not refer to the reference point but to the phase center of the antenna. The exact location of the phase center with respect to the reference point of the antenna is something that has to be determined for each type of antenna. This information is available (again from IGS) for a lot of different antenna types. However the location of the phase center depends of several factors. For this reason the information is provided in two parts. The first is the Phase Center Offset (PCO). This describes the offset between the phase center and the reference point of the antenna, but this value depends on the frequency of the signal received. The PCO is defined at zenith, so when the signal is received from straight above the antenna. The receiver Phase Center Variation (PCV) describes the change of the phase center under different angles, both zenith and azimuth. For both angles the change is usually described per 5 degrees. Also the PCV depends on the frequency of the received signal.

For each type of antenna the PCOs and PCVs are completely different and must therefore be uniquely determined through testing. For this reason it is to be expected that this information is not available for every type of antenna. For some types only the PCO is provided, for some antenna types no information at all is given. If no information is provided a value of zero is assumed for the missing parts. Since the value of the PCO is on the centimeter level the results for data from an antenna with incomplete information cannot be expected to be of the highest quality.

As both the PCO and PCV values depend on the frequency the information is provided per frequency. However, only for the first two frequencies of GPS. For all other constellations, except GLONASS, there is no information available which decreases the expected performance. The antenna data files from IGS also provide information on the PCO and PCV of the satellites. This information has recently become available for all constellations currently in use [11].

2.3. Disturbances and corrections

As already indicated for the pseudorange, it is not free of disturbances. In fact both the phase and code observations are affected by a number of disturbing factors. Some of these disturbances can be modeled, others can be estimated from the observation data. Some effects are even small enough to simply be ignored. Describing all disturbance sources is not of use for the research done, for which reason this section describes the disturbances that are important in the context of the user software. More complete descriptions are available from Heroux and Kouba [6]. The disturbances and corrections described are the troposphere and ionosphere delays and the correction for phase wind-up. The effects that are not described are not minor, as for instance relativistic effects must be taken into account for any form of positioning, but the implementation in the software is straightforward.

2.3.1. Troposphere

The electromagnetic signals transmitted by the satellites travel at the speed of light in a vacuum. Within the atmosphere of the Earth the assumption of a vacuum cannot be made, hence the transmitted signals are delayed. When considering satellite positioning the atmosphere is split into two sections. The lower section is the neutral atmosphere. Neutral in this case refers to no significant presence of charged particles. Even though the stratosphere contains sufficient neutral elements to affect the satellite signal, the majority of the particles are found in the troposphere. For this reason the neutral atmosphere is referred to as the troposphere for the purpose of positioning [18].

The presence of neutral atoms and molecules in the troposphere causes a delay in the satellite signals. The important consideration for the troposphere is that due to the composition this delay is not frequency-dependent. The benefit is that the delay is therefore the same for all signals from a single satellite. What is estimated is the Zenith Tropospheric Delay (ZTD), the delay when a satellite would be straight above a receiver. This estimate is then applied to all satellite signals using a mapping function depending on the elevation (lower elevations lead to a longer path through the troposphere). Estimation of the ZTD is two-fold. The first is the Dry Tropospheric Delay which can be accurately modeled. A frequently used model is provided by Saastamoinen [15]. The second part of the tropospheric delay is the Wet Tropospheric Delay. This delay is caused by the presence of water (vapor and liquid) in the troposphere and therefore shows more fluctuation. This second part is estimated from the observation data. Because of the separation from the dry part, the wet part of the delay can be modeled as a random-walk process.

2.3.2. Ionosphere

In contrast to the troposphere the ionosphere has sufficient charged particles to affect the signals passing through. The electron density along the path of the signal is what directly influences the delay. This density is described by integrating the density over the signal path, leading to the Total Electron Content (TEC). Contrary to the tropospheric delay, the ionospheric delay does depend on the frequency of the signal. A first order approximation can be given as in equation 2.1[18], where ι represents the ionospheric delay and f the frequency of the signal. α is a scaling factor that does not depend on the frequency.

$$\iota = \frac{\alpha \cdot TEC}{f^2} \tag{2.1}$$

In contrast to the tropospheric delays the ionospheric delays are highly variable. Therefore the delays cannot be estimated beforehand without additional information. The only options are estimating the delay, cancelling the effects or providing external information on the TEC. The first two options are possible for a single receiver when using more than a single frequency. For the third option the ionospheric delays are provided externally, either from another receiver close-by or from an ionosphere map created for the region. With the ionospheric delays provided in this way it is also possible to use only single frequency observations, though performance depends heavily on the precision of the provided ionosphere information.

The effect of the ionospheric delay on the observations is not the same for code and phase observations. The code observation is delayed by the ionosphere, leading to a range estimate that is too large. The ionosphere has an opposite effect on the phase measurements. The delay advances the phase of the signal, leading to a range estimate that is too small. This effect is important when setting up the observation equations.

2.3.3. Phase wind-up

If a receiver and satellite are considered stationary there is no change in the phase measurements. When a satellite is moving away the change in range causes additional cycles in the phase measurements. However since the orientation of the satellite also changes this has to be taken into account. If a satellite completes a full rotation around the axis fixed to the antenna orientation a full cycle is added, or subtracted, from the phase measurement. This phase wind-up is something that has to be specifically modeled for precise positioning. This requires information on the position of the user, but also the position of the satellite and specifically the position of the phase center of the antenna of the satellite. Additionally the movement is different per satellite type within each GNSS constellation. Almost all satellites ensure that the antenna is pointing in nadir direction, in other words towards the Earth. In the meantime the satellite is constantly rotating in order to keep the solar panels pointed towards the Sun. Normally this movement is slow, yet in some occasions the position of the Sun relative to the satellites orbit is such that relatively quick turns are required [11]. All these cases must be taken into account when modeling the phase wind-up. Modeling the wind-up allows the phase observations to be corrected for the satellites rotations [6]. The user can apply this wind-up correction using the satellite position and the position of the Sun relative to the satellite. This takes into account that the antenna on the satellite always points nadir, while the solar panels are directed at the Sun.

2.4. Observation equations

The basis for any form of positioning are the observation equations, relating the observations to the estimable parameters. For the code and phase observations a complete description of the derivation of the observation equations has already been done [18]. For the purpose of testing the user software the entire derivation is not required. Of importance are the observations equations as implemented in the software. A simplified version of the code observation equation is given in equation 2.2 (based on [12]). Simplified in this case refers to neglected elements like measurements errors (for instance multipath). The terms included in the equation are the ones that are modeled, estimated or canceled during positioning.

$$E(p^s_{r,j}) = \rho^s_r + (dt_r + d_{r,j}) - (dt^s + d^s_{,j}) + \mu_j \iota^s_r + \psi^s_r \tau_r \tag{2.2}$$

In equation 2.2 $E(p^s_{r,j})$ represents the expectation of the code measurement. The superscript s is given multiple times and refers to the satellite which is the source of the measurement. The subscript r in the same way refers to the receiver. The subscript j is used when a parameter depends on the frequency of the signal. The first parameter on the right hand side of the equation is ρ. This represents the geometric distance between the satellite and the receiver, the information that is actually required for positioning. The presence of the subscript r and superscript s indicate that the parameter depends on both the satellite and the receiver. dt represents the clock error. Separate parameters are present for the receiver and the satellite as these two are independent. However both the receiver and the satellite suffer from internal clocks that are not completely stable. The parameter d represents the code hardware delay. This parameter is again present for both receiver and satellite, but is in this case also different for each frequency, leading to the addition of the subscript j. The next parameter is ι which represents the ionospheric delay between receiver and satellite on the first frequency. As indicated in equation 2.1 the first order estimate of the ionospheric delay has a direct relation with the used frequency. This means that the ionospheric delay on any frequency can be related to the delay on the first frequency. For this reason $\mu_j = \lambda_j^2/\lambda_1^2$ relates the ionospheric delay between the different frequencies. τ_r is the ZTD for a receiver. The mapping function ψ maps this zenith delay to the path between the satellite and the receiver. For the phase observation equation 2.3 is provided (based again on [12]).

$$E(\phi^s_{r,j}) = \rho^s_r + (dt_r + \delta_{r,j}) - (dt^s + \delta^s_{,j}) - \mu_j \iota^s_r + \psi^s_r \tau_r + \lambda_j M^s_{r,j} \tag{2.3}$$

Several terms are similar in equation 2.3 compared to the code observation equation. The clock error terms are the same as these are common for both code and phase measurements. δ represents the phase hardware delays for both receiver and satellite. These delays are different for the code and phase, but the effect is similar leading to the use of d and δ to depict them. The ionospheric delay term is similar but, as indicated when describing the ionospheric delays, the opposite sign is used. The additional term M consists of several parts and is therefore provided in equation 2.4.

$$M^s_{r,j} = \varphi_{r,j}(t_0) - \varphi^s_{,j}(t_0) + N^s_{r,j} \tag{2.4}$$

In this equation the terms φ represent the initial phase of the receiver and the satellite upon acquisition of the signal. N represents the ambiguity in the phase measurement between receiver and satellite (per frequency).

An important consideration is that this ambiguity is an integer value, representing a number of full cycles. The initial phases for both satellite and receiver cannot be distinguished from the phase hardware delays and are therefore from now lumped together with each other. These are now referred to as the receiver and satellite phase bias. For the code observations code hardware delays and code biases have the same interpretation, but to keep the similarity to the phase observations the second name is used.

2.5. Single Point Positioning

Single Point Positioning (SPP) is the most basic form of positioning that can be performed by the user platform. SPP makes use of only the code observations. A minimum of 4 satellites are required to estimate the receiver coordinates and the receiver clock error. Using only a single frequency is possible, however the ionospheric delays are neglected or a-priori corrected in this scenario. The precision in this case is very poor with standard deviations of several meters. Including the code observations from a second frequency allows the ionospheric delays to be estimated, improving the horizontal precision by 25% [16]. Still the precision remains above the meter level. It is possible to use either the broadcast ephemeris or precise orbit products from an external source. In both cases the solution is called SPP, but the higher precision of the precise orbit products improves the result compared to the broadcast orbits.

The use of SPP in the user software is mainly focused on getting an initial rough estimate for the receiver coordinates and determining the transmission time of the satellite signals. This last step allows the actual position of the satellites at time of transmission to be determined, which is required for any further precise positioning. With the approximate receiver coordinates and the fixed satellite coordinates known also the elevation and azimuth angles for the satellites can be determined. The satellite coordinates are 'fixed' as they are assumed to be known parameters without any uncertainty applied to them. The determination of the satellite elevation is important to remove satellites from further calculation if the elevation is below a specified threshold. This threshold is defined because observations from low elevation satellites are more prone to outliers due to, for instance, multipath. Also the signals from low elevation satellites have a long path through the atmosphere, decreasing the precision. The value for such a threshold can be varied for different scenarios. As for the precision of SPP, the effect of a receiver position that is off by a few meters has negligible effect on the estimation of the satellite elevations.

2.6. Precise Point Positioning

For PPP first precise corrections for the satellite orbits and the satellite clocks are required. The instability of the clocks on-board the satellites is large enough to adversely affect the position estimates. Compared to SPP now also the phase observation are used next to the code observations. The phase observation equations of course contain the unknown parameter of the phase ambiguity. While this unknown cannot be directly solved, a powerful constraint can be put on it. Assuming the receiver continuously tracks the satellites and there have been no cycle slips (or they have been corrected for), the ambiguities are constant over time. The result of this assumption is that every epoch with observations can be used to improve the estimate of the phase ambiguities. Note that this does not hold for the undifferenced ambiguities. In the observation of a single satellite for a single receiver the ambiguities contain the receiver phase bias, which is not always constant in time. However by differencing between two satellites this receiver phase bias is canceled. Therefore it is the single differenced phase ambiguity that is to be assumed constant. For this reason a single satellite is chosen as pivot satellite. This means that the observation from that satellite are subtracted from the observations of all the other satellites. Note that the phase ambiguity, contrary to the observations equation 2.3, is not integer valued. The cause is that the satellite phase bias is still present and cannot be individually estimated. In the case of multi-GNSS and overlapping frequencies one pivot satellite can be selected per frequency, instead of one pivot per constellation. This improves the strength of the model.

The weakness of PPP is that it only reaches its best performance after some time. The solution improves over time, seen by the convergence of the solution. This is directly related to the behaviour of the estimated ambiguities. Given enough time these converge to a constant value. Convergence of the solution is usually defined as the time it takes for the solution to reach a precision of less than 1 decimeter [2]. For a normal case using dual frequency GPS and a stationary receiver this can take approximately 30 minutes [2]. However a normal case is hard to define. The convergence time depends heavily on different factors. The number of satellites and their geometry is an important factor. A receiver tracking an average of 7 satellites is likely to have a longer convergence time than one with 10 satellites in view. Taking into account the case of multi-GNSS a receiver with even more satellites is sure to benefit. If all satellites in view of the receiver are visible with similar azimuth and elevation angles, the geometry is poor and is likely to lead to a longer convergence time. For PPP precise orbit and clock products are used to improve the precision of the solution. However many different products are available, with varying quality. Using high quality products (generated by post-processing) it is possible to get position estimates using PPP with a standard deviation of 2-3 cm for the East and North components. The Up component can reach a precision of 4-5 cm. In this case kinematic positioning is done, so the position of the received is not assumed to be constant. When doing static positioning the estimates keep improving by using a longer measurement time.

In the standard case PPP is done with dual frequency GPS. Any constellation can be used and with any number of frequencies. Using a single frequency is possible, however this requires additional information on the ionospheric delays, as these cannot be estimated separately. The results for such a scenario heavily depend on the quality of the ionosphere information that is used. If the ionosphere information is accurate it can also be used for multi-frequency PPP and will in this case speed up the convergence time.

2.7. PPP-RTK

The concept of PPP-RTK is an improvement to PPP. This improvement is taken from the relative positioning technique Real-Time Kinematic (RTK), hence the name. For RTK two receivers are used, which are close to each other. The closer two receivers are to each other, the more parameters can be ignored as they are exactly the same for both receivers. The important part of RTK that is applied to PPP-RTK is the nature of the ambiguities. In PPP the ambiguities are real valued. Even after differencing between two satellites the ambiguities have real values. The undifferenced real valued ambiguities consist of three factors. The number of cycles between the receiver and the satellite, a phase bias on the receiver side and a phase bias on the satellite side. The receiver phase bias is canceled out when differencing between two satellites. For RTK it is possible to difference between the two receivers. In this way the phase bias of the satellite is canceled out. Therefore the double differenced (between satellite and between receiver) ambiguities have to have integer nature.

In the case of PPP-RTK the user has the observation data of only a single receiver. The information of the satellite phase biases is provided to the user. By applying these biases to the observations of the user the ambiguities are already single differenced (between receiver) and the additional step of between satellite differencing ensures that the ambiguities are integer valued. The improvement of PPP-RTK with respect to RTK in this case is that the user side is not affected (or does not need to know) the source of the phase bias corrections. These can be generated from a single receiver (providing a scenario similar to RTK), but may also have been estimated from a small network of receivers or even a full global network of reference receivers.

At this point the choice for not including GLONASS in the user software can be made clear. Because all GLONASS satellites transmit signals on a slightly different frequency. The receiver phase bias is different for each frequency and therefore does not completely cancel out when differencing between two satellites. Therefore the double differenced ambiguities do not have integer nature in this case. Attempts have been made to resolve this by using a second pivot satellite [1], but use of this method is not always practical. GLONASS uses an offset in frequency that is multiplied by a value k unique for each satellite. The proposed method has the requirement that the two pivot satellite should have an offset in frequency equal to one step. Investigation of an observation dataset shows that the number of available pairs can often drop to zero, making this method problematic.

Applying the phase bias corrections affects the nature of the estimated ambiguities as they can now be considered double differenced and therefore have integer nature. The estimates are still real valued. However, the expected means to which these (double differenced) ambiguities are converging over time are now integer values. The benefit of this is that it becomes possible to predict the value to which an ambiguity is converging before is has approached that value. Fixing the estimated ambiguities to an integer value is called ambiguity resolution. The benefit of this is two-fold. Some noise that is present in the observations can be reduced by this process, leading to more precise precision estimates. The larger improvement is however in the main weakness of PPP, the convergence time. The time improvement is again difficult to quantify as it differs depending on the scenario. For the case of GPS-only, with no ionosphere constraints the convergence time can go below 20 minutes. For multi-GNSS this can be decreased close to 10 minutes. When adding in ionospheric delays computed from a nearby reference station this can be decreased even more. With sufficiently accurate ionospheric delays and assuming the receiver position to be fixed the convergence can be instantaneous (as in the first epoch). These results have all been attained with the user software itself.

The resolution of the ambiguities to integer is not trivial. Determining the best integer candidate for each of the double differenced ambiguities requires searching through many options. The computational burden becomes worse with more satellites and more available frequencies. In the case of multi-GNSS the number of ambiguities to be fixed becomes very large and the search space becomes problematic. Simple solutions, like simply rounding the ambiguities to its nearest integer value, are not a computational issue, but rounding is not optimal. The integer least-squares method is optimal, but is computationally intensive. To reduce the computational burden the user platform uses Least-squares Ambiguity Decorrelation Adjustment (LAMBDA) [17].

2.8. User Platform

The Curtin PPP-RTK user platform was developed by the Curtin GNSS Research Centre at Curtin University. There were multiple incentives for the development of the user platform. It was one of the deliverables for a project sponsored by CRCSI (CRC for Spatial Information). In this case project 1.01 (New carrier phase processing strategies for next generation GNSS positioning). The user platform is also to be used in another CRCSI project (1.19), for which it can be used to test new developments in network processing as a user. Apart from this the user platform is also built in a flexible way to allow for future research on network and user concepts.

The user platform is a software prototype in Matlab. The software supports both the standard PPP and the PPP-RTK positioning methods. The constellations that are supported by the software have been restricted to all that make use of CDMA, for which reason Glonass support is not provided at this stage.

The user platform makes use of an undifferenced approach ([13]). This approach uses undifferenced observation equations so all parameters are present in the model. This is in contrast to for instance using ionosphere free combinations, in which case the ionosphere terms disappears from the model. The undifferenced approach does require additional constraint to make the model full rank. With all parameters still present the benefit is that a dynamic model can be assigned to each of them.

For research purposes the user platform allows these dynamic models to be set by a user. A parameter like the estimated position of the receiver can be set to constant, a position with some noise, or a position that is completely unlinked in time. The same holds for the other parameters that are still present in the model due to the use of the undifferenced approach. The only exception are the carrier phase ambiguities, which are either constant or completely unlinked in time.

Both PPP and PPP-RTK make use of the undifferenced approach. For PPP-RTK the undifferenced approach has another benefit. When using the ionosphere free combination with more than two frequencies the number of options for the satellite phase biases become larger as different combinations of two frequencies are possible. When providing corrections for satellite phase biases from a network in the case of the undifferenced approach there is a single bias per frequency per satellite. Corrections provided in any form can be applied in the user platform, as long as the format is clearly described.

2.9. Stochastic model settings

For any positioning method used the stochastic model is important. Some input used by the user platform is assumed to contain some uncertainty while other input is taken as the true value. For both PPP and PPP-RTK the satellite orbit and clock products are treated as true values. The same holds for the satellite phase biases provided for PPP-RTK.

For ionospheric delays the setting is optional. Externally provided delays can be treated as having zero noise. This option can be used for very precise delays, but for many ionosphere models the products must be assumed to contain noise. The expected standard deviation for this noise can be set. If the noise level is set too high the convergence time and precision of the result gets poorer. If the noise level is set too low many observations are rejected. Eventually no solution is provided as the outlier detection only investigates the observations, not the ionospheric delays that have been provided.

For the code and phase observations uncertainty is present. For all observations a standard setting (0.3 m for code, 0.003 m for phase). This setting is used for all constellations. For large scale testing these settings are kept, even though for specific constellations and receivers different values can provide better results. If the code and phase precision is set too high, outlier detection will identify very few observations. Any small outliers go undetected. If the precision is set too low many observations are wrongly identified (false positives).

3

Testplan

To test the Curtin PPP-RTK user platform a large number of datasets have to be processed. During the development of the software only a small selection of datasets are used, but for full testing a larger number of datasets is required. The use of real observation data is likely to provide a large number of unexpected situations. The unexpected situations are the main reason for testing the functionality, as these should be handled without problems.

3.1. RINEX observation files

Observation data for the software is provided in RINEX files, either version 2 or 3. Even though these file formats are extensively described[14] the large number of sub-versions and freedom within the file definitions require the file reading functionality to be flexible. For testing purposes the main source of observations files is IGS. Figure 3.1 shows the distribution for a single day in 2014. These stations are part of the normal IGS network and therefore provide observation data for GPS and in some cases Glonass. For these observation files RINEX version 2 is used.

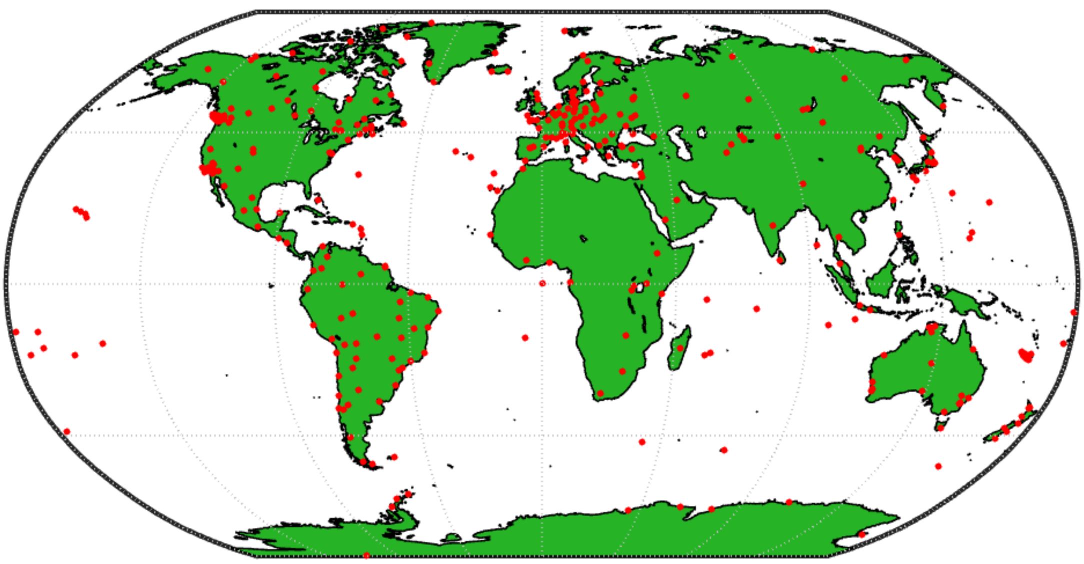


Figure 3.1: IGS station providing observation data on 14-245 (September 2nd 2014).

Related to IGS is the Multi-GNSS EXperiment (MGEX). For this project around a hundred stations provide data on a daily basis. Not every receiver tracks all constellations, but since the newer constellations are used the observation files in the MGEX network are RINEX version 3. Figure 3.2 shows all the MGEX stations providing data on a day in 2015. As an addition to the available datasets also the observation data from the receivers present at Curtin University are used. All these datasets can be used for any tests performed with the software and should present enough unexpected events for testing.

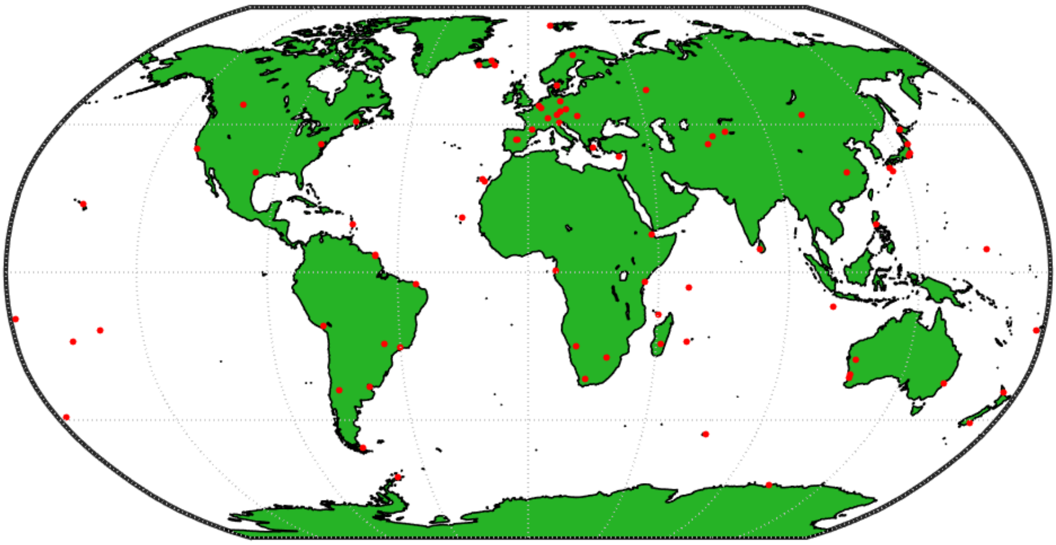


Figure 3.2: MGEX stations providing observation data on 15-168 (June 17th 2015).

3.2. Orbit and clock products

For precise positioning external data on the orbits of the satellites and on the clocks in the satellites is required. For both data files with corrections are provided in a predefined format. The orbit and clock products are produced by analysis centers all over the world. All these products are provided in the same format, but the data is different. The quality changes because of different processing methods. The highest quality orbits are the IGS final products. These are combined results from multiple analysis centers, but due to extensive post-processing are only available after several weeks. Products that are made available sooner (after a few hours, real-time or even predicted) are expected to be less precise. Products of different quality must still be processed by the user software, even though the quality of the results is not expected to be optimal.

For both the orbit and clock products the intervals at which corrections are provided is different from different sources. Orbit products usually have 15 minute intervals, but for instance Centre National d'Etudes Spatiales (CNES) provides data files with a 5 minute interval. The IGS final clock products are available with either 30 second or 5 minute interval, but the CNES real-time clocks have a 5 second interval. All these different versions of products must be correctly handled by the software.

The products mentioned before are all used for GPS. Some analysis centers also provide orbit and clock products for the other constellations. Again the quality differs between these products, as do the number of satellites in use for each of the products. Apart from the software having to handle the different situations this is also an important consideration when testing the performance of the software. When comparing the results of two scenarios both have to use the same products for a fair comparison.

3.3. PPP

When testing PPP using only GPS any day can be selected for all available IGS datasets to be processed. Since for PPP all products (orbit and clocks) can be used globally, all datasets can be processed in the same manner. In the case of multi-GNSS all MGEX datasets can be processed using any of the available orbit and clock products. The geographical location does become an important factor in these cases because the number of satellites available at any point of the Earth is different for the newer constellations.

Even though all datasets must be correctly processed the results from the tests are also used to identify problematic datasets. For testing the performance of the software some datasets are excluded, due to for instance hours of missing data. A problematic dataset is still processed without causing exceptions in the software, but the results can be bad.

For PPP the performance is tested by investigating different factors. The precision of the results by analyzing the stability of the solution after some time. The accuracy by comparing the computed position with an external solution for the position of the receivers. The time it takes the solution to converge is important as well. All these are also investigated as additional constellations are added to the solution.

3.4. PPP-RTK

The satellite specific corrections needed for PPP-RTK are not available for all constellations in the public domain. For that reason products are generated in-house at Curtin University. These products are limited in that these are at this stage still based on a single receiver. Because of this limitation processing global datasets is difficult. The user station must always be in range of the reference station (generating the products). This limitation is based on the number of satellites in view for both station and can therefore still be several thousand kilometers. However processing all global stations with one set of corrections is not possible.

The performance of PPP-RTK using the single station products is used as an indication of the performance when true network products are available in the future. When the station processed as user is close to the reference station the single station products can perform better than a network solution. A disturbance still present in the solution can cancel out between the two stations because the receiver-satellite geometry is the same for both stations. If the distance is large such a disturbance shows up in the solution. However the increased distance between the two stations also limits the number of satellites that can be used. In this case the single station product is expected to yield poorer performance for the user than a future network product.

Again all user stations must be correctly processed, even if there are no satellites in common view. For the analysis of the results the distance between the user and reference stations must be taken into account. For testing multi-GNSS not as many datasets are available restricting the number of pairs that can be formed for testing.

The two effects of ambiguity resolution done in PPP-RTK that are investigated are the increase in precision and the improved convergence time. When adding in more constellations these two factors are affected, which is analyzed. With the geographical restrictions for some constellations the number of datasets available for these tests is small.

3.5. Software output

The performance of the software is analyzed based on the output generated. After processing a dataset several plots are available. For one of the processed stations the plots generated by the user software are depicted. This station was processed in kinematic mode using standard PPP. Figure 3.3 is a summary that shows a combination of plots. The sub-plot on top is the estimated position in the East, North and up components. The legend of this plot shows the empirical standard deviation for the three components. The value μ indicates average difference between the estimated position and the ground truth provided by IGS. RMS is the root mean square computed from the estimated position and known ground truth. If no ground truth is provided only the standard deviation is shown. These statistics are only determined for the part of the plot on the right side of the ‘convergence’ line. This line is in this example set at four hours. Four hours is enough for any solution to converge, allowing for a fair comparison between different datasets. The second sub-plot shows the total number of satellites during the day. The third sub-plot indicates the values of the Local Overall Model (LOM) test and its critical value for every epoch.

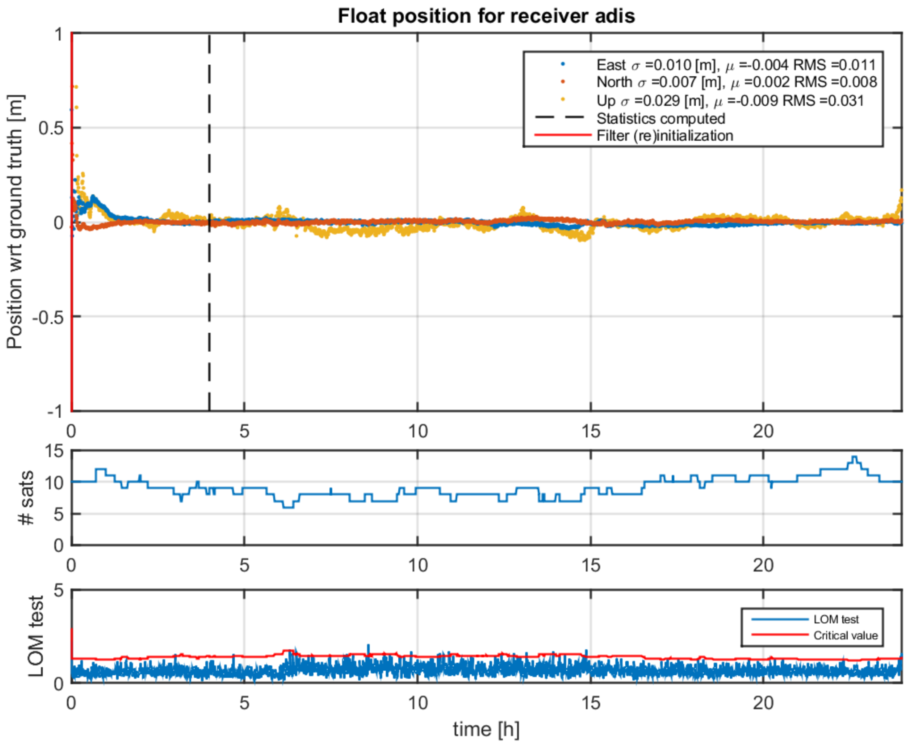


Figure 3.3: Summary plot for receiver ADIS. Standard PPP in kinematic mode.

Figure 3.4 depicts the North, East and up components separately. These plots show the behavior in each component more clearly. In the combined plot the North and East component are most of the time covered by the less stable up component. Because the precision of the up component is not as good as the horizontal components the scale is this plot is also slightly different for the up component. Figure 3.5 is a scatter plot of the horizontal components. Both figure 3.4 and 3.5 are different when PPP-RTK is used. In that case the solution after ambiguity fixing is shown as well. When processing a large number of datasets these individual plots are impractical. The information displayed can however still be extracted from the numerical results stored after a dataset has been processed. A statistic like the empirical standard deviation is then still determined in the same way as in the plots, so after a four hours.

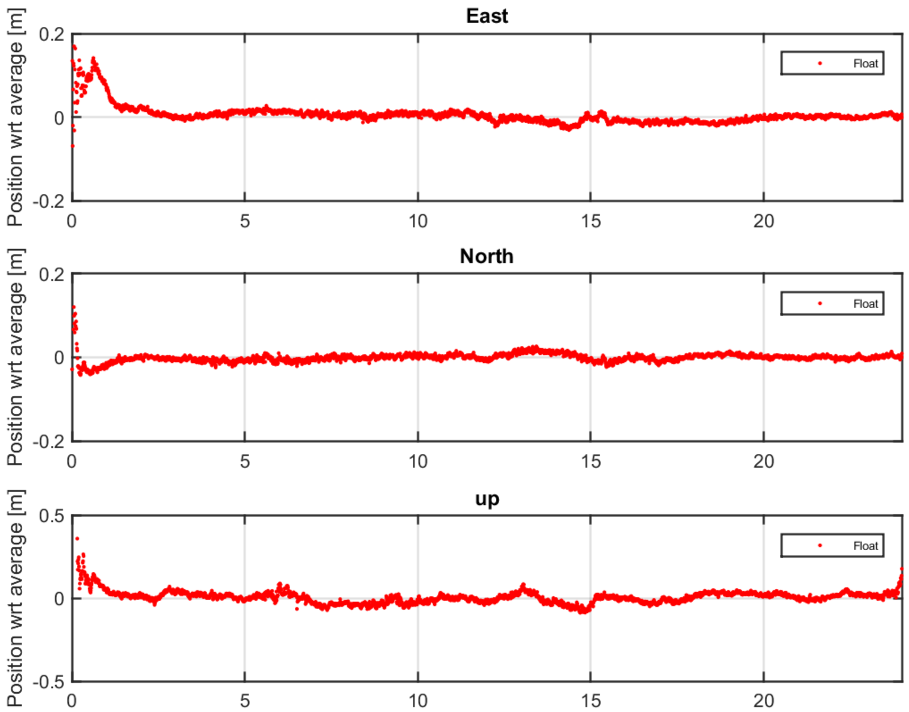


Figure 3.4: Position time series for receiver ADIS. Standard PPP in kinematic mode.

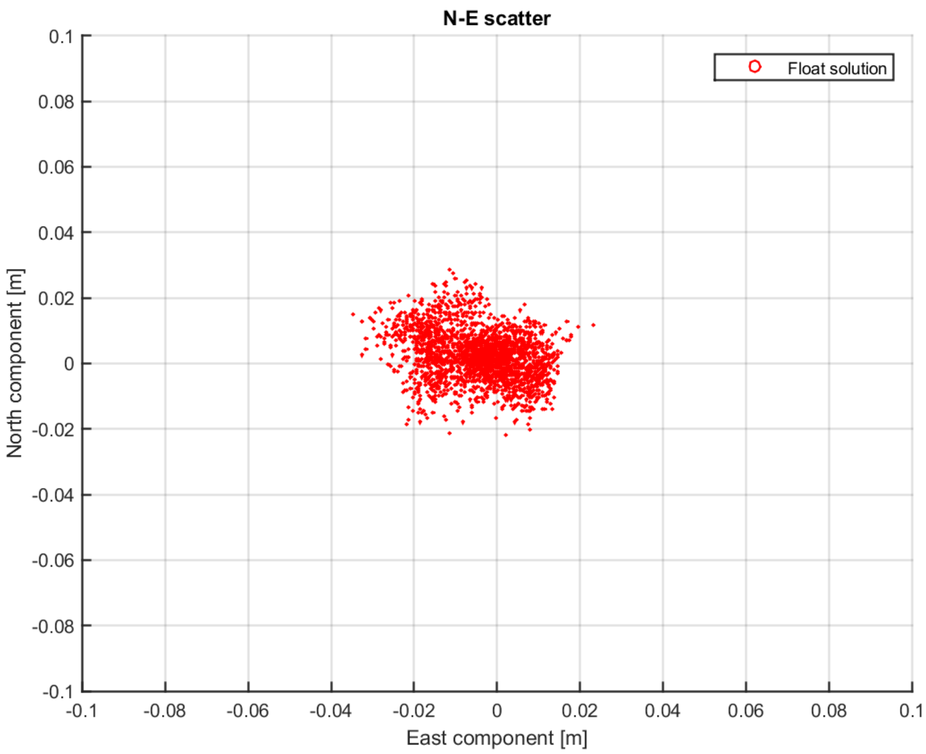


Figure 3.5: Horizontal scatter plot for receiver ADIS. Standard PPP in kinematic mode.

4

Test Results: Precise Point Positioning

This chapter describes the testing of the standard PPP functionality of the user platform. Problems in the software that have been encountered when processing specific datasets have been corrected. Examples of problems that have been resolved can be found in appendix D. After correcting for these issues the performance of the software in standard PPP has been analyzed. This testing has been split up into a section focusing on only GPS and a section focused on multi-GNSS. Multi-GNSS in this case refers to all constellations, including modernized GPS, and combinations of these constellations.

4.1. Standalone GPS

For testing standard PPP with only GPS, observation data from IGS is used. Two days are picked for this that are far apart, leading to small differences in the GPS constellation. The two days are December 14th 2013 (13-348) and September 2nd 2014 (14-245). These short notations for the year and the day of the year are used from now on. Both days are processed using two sets of orbit and clock products. Once with the IGS final products and once with the real-time products from CNES. The two options used for the receiver position are static (constant position) and kinematic (no dynamic model for the position). For kinematic positioning the empirical standard deviation of the estimated position is an indicator of the quality of the results. The static positioning results in a much lower empirical standard deviation for any dataset as the position estimate keeps improving over time. This setting also more clearly shows the convergence of PPP solution. Because the convergence of the solution is determined with respect to the position estimate after four hours, a more stable solution after these four hours is preferred. For some stations the ground truth location is available from IGS. This too can be compared with the results to see if there is no constant bias in all the solutions.

The reason for the comparison is not to determine whether or not the real-time products have poorer performance. The real-time products from CNES cannot be expected to provide the same quality as products that go through 3 weeks of post-processing. However, the problems in the orbit and clock products that can occur in real-time data are better to test the software's response to unexpected situations. If no large problems occur in the orbit and clock data the results should be acceptable, even though the better products are expected to yield far better results.

The observation files used for this analysis are all those that have not been rejected in the first stage of testing. Even though these rejected files can be processed the results are not very good. For instance the observation data must not have any data gaps. Some datasets have a couple of missing epochs, or even several hours. The results for these sets are poor, but that is not caused by the software. The same holds for datasets from receivers with reception problems. These are tested, but no matter what software is used the results are always poor. This does not mean that only the very good receivers remain, the final datasets do contain datasets of varying quality. A dataset is discarded only when the data is so poor that the solution has to converge again halfway through the day. For two days used for testing a significant amount of datasets remain. For 14-245 357 datasets remain. For December 13-348 409 datasets are used.

4.1.1. Dual frequency kinematic positioning

The first step is testing the software in kinematic mode. The only parameters with a dynamic model are the ambiguities. All other parameters are free to vary between epochs. Table 4.1 summarizes the results for the two days. Each day is processed using the two sets of orbit and clock products. Each dataset provides a full day of position estimates. As the receivers are stationary the standard deviation of this estimated position over the course of the day is used to describe the precision of the results. To allow the solution to converge the first four hours of the solution are not taken into account when determining this empirical standard deviation. When analyzing multiple datasets these standard deviations are then averaged for all datasets of the given day. These averages for the day in 2014 show a small difference in performance between the IGS and CNES products. Since the CNES product are real-time this decrease is performance is expected, but the effect is minimal. For the day in 2013 there are more problems. As indicated by the number of stations used for calculating the results quite some are missing for the tests with CNES products. The reason is that these sets gave exceptionally bad performance.

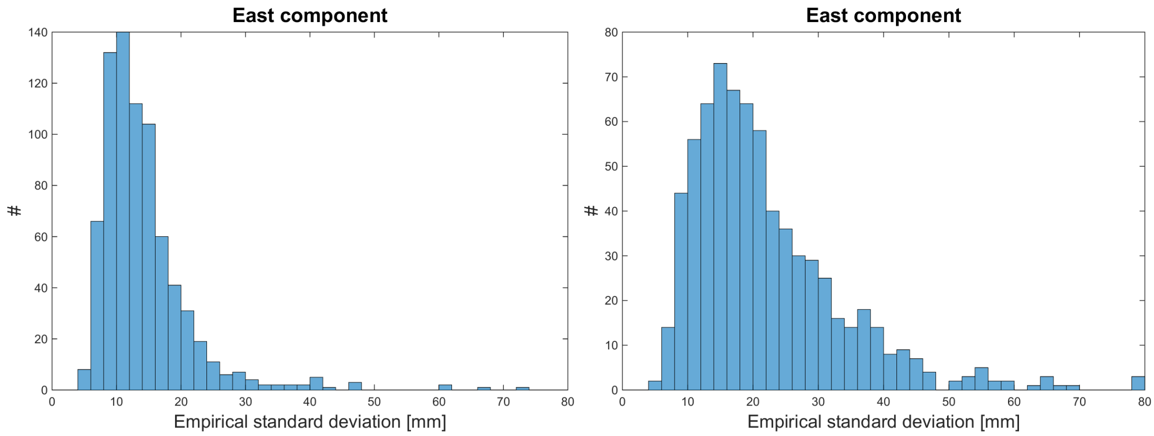
The IGS products contain orbit and clock data for almost all satellites for every epoch. The CNES products, because of the real-time nature, have gaps where data for some satellites is missing for several epochs. If the wrong satellites are affected at the wrong time the results for a specific station can be distorted, but a couple of these missing datapoints are not problematic. For the day in 2014 this problem is larger as there are no products available for several minutes after 4:30 in the afternoon (Universal Coordinated Time (UTC)). This disrupted the results for all stations. For this reason the standard deviations for that day, as shown in table 4.1, are only computed up to that disruption.

The poor performances when there are gaps in the data is expected for the kinematic mode. By introducing dynamic models for parameters other than the ambiguities the model becomes stronger and more problems can be captured. If, for instance, all satellites disappear for one epoch the solution has to converge again as if it is the start of the dataset. The standard deviations determined for such a dataset become very large. In these cases datasets are excluded from the analysis, like the 45 datasets on day 13-348.

Day	02/09/2014		14/12/2013	
Year - day of year	14-245		13-348	
Stations	357	357	406	361
Orbit & clock products	IGS final	CNES real-time	IGS final	CNES real-time
Mean σ_E [mm]	13.5	16.7	15.6	28.0
Mean σ_N [mm]	11.9	14.5	12.5	19.5
Mean σ_{up} [mm]	36.3	44.9	43.7	63.9

Table 4.1: Results for kinematic positioning for dual frequency GPS. Two processed days with two different sources for orbit and clock products. The average standard deviations are given for the North, East and up components.

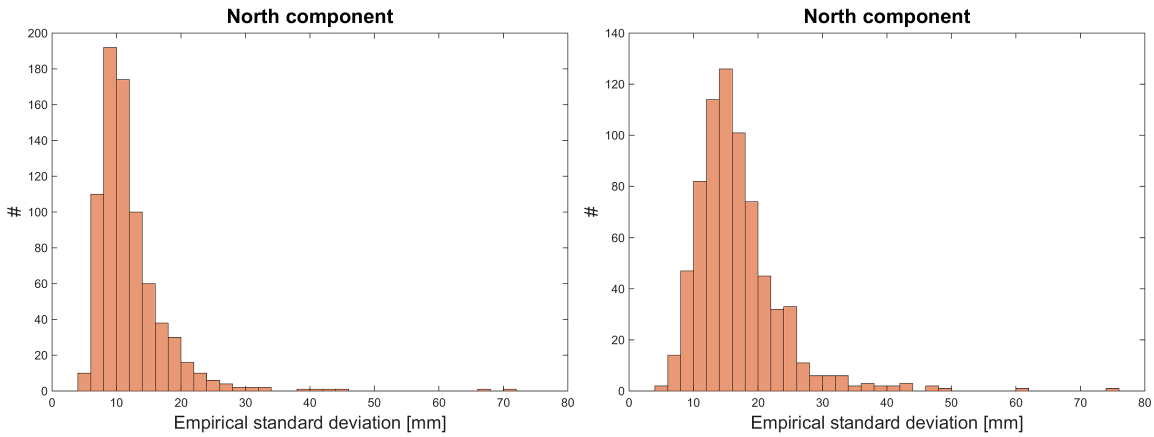
For both the sources of orbits and clock products the results from the two days are summed up. The histograms of the empirical standard deviation for both datasets are given in figures 4.1, 4.2 and 4.3. For each of these figures the histogram indicates that the results for the IGS products are more precise, as is also indicated by the earlier provided mean values. Especially in the case of the IGS products for the North and East components the distribution has a similar shape. The same rough shape can also be seen in the other histograms, but is less pronounced. The histograms appear similar to a χ^2 distribution. Normally distributed noise in the observations leads to a distribution of this shape for the standard deviation of the estimated position. For comparing to a χ^2 distribution the variance has to be used instead of the standard deviation. However testing whether the provided histograms actually match a χ^2 distribution is not useful as there are effects present in the datasets that do not follow any specific distribution. The receivers used for these computations are all over the world and not of the same type. Neither the receiver locations or the receiver types can be expected to follow any specific distribution. If the receiver type is a dominant factor in the quality of the positioning results the distribution shown in the histograms is expected not to be smooth at all. Since a clear distribution pattern is visible this can indicate that neither the location or the type of the receiver is a dominant factor in the results. Because of the shape of the histogram the mean precision provided for each of the datasets is a representative value for the results.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

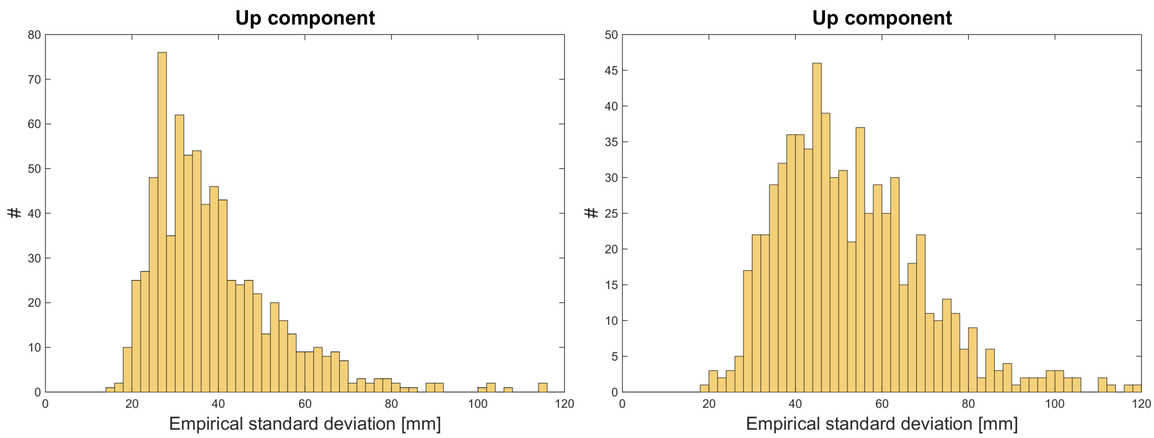
Figure 4.1: Histograms for empirical standard deviation of the East component for all station. Kinematic positioning is used.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure 4.2: Histograms for empirical standard deviation of the North component for all station. Kinematic positioning is used.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure 4.3: Histograms for empirical standard deviation of the up component for all station. Kinematic positioning is used.

For static PPP the precision for the East and North component is expected to provide precision of a few centimeters [2]. Static in this case refers to the receiver being stationary. Using the IGS orbit and clock products the user software indeed matches this performance. The weaker up component is expected to be around the decimeter level [2], which is the case using the IGS final products as well.

The convergence time of the PPP solution is an important aspect of the quality of the results. As previously indicated the criterion used for the convergence time is not always the same. The main goal of setting a criterion for the convergence time is to compare the results from different scenarios. Therefore the choice is made to use the numerical criterion from [2]. This criterion is an offset of 1 decimeter for East and North from the actual position. The actual position in this case is the mean of the estimated position computed after four hours. In this case the solution has converged for any scenario, with the exception of problematic data. Using ground truth is difficult as it is not available for all stations. It is possible to determine a convergence time for each single dataset by simply identifying the first epoch in which the estimated position is within the one decimeter range for both horizontal components. The problem is that, especially for kinematic positioning, there is a chance that one of the first epochs is already within this range, while later epochs the offset becomes larger again. Another option is to use a window. In that case the solution must remain within 1 decimeter of the actual position for a number of epochs. The problem then becomes the selection of the window length.

The convergence time for a single dataset is not of interest. For a larger number of datasets the average convergence time is investigated. Because of this the approach chosen to determine the convergence time takes all datasets into account (based on an example by Trimble [9]). For every epoch in every dataset the offset from the actual position is determined. For each epoch the list of offset values is sorted and three limits are determined. The limits are the 50, 68 and 90 % limits. In the case of 90-limit this means that 90% of the datasets have an offset from the actual position that is under this 90% value. The resulting convergence plots show three distinctive lines that can be used to compare between scenarios that do not have the same number of observation files. The percentages also diminish the effect of datasets with outliers. Even if there are a lot of problematic datasets this can be seen in the 90% curve, while the other are not affected.

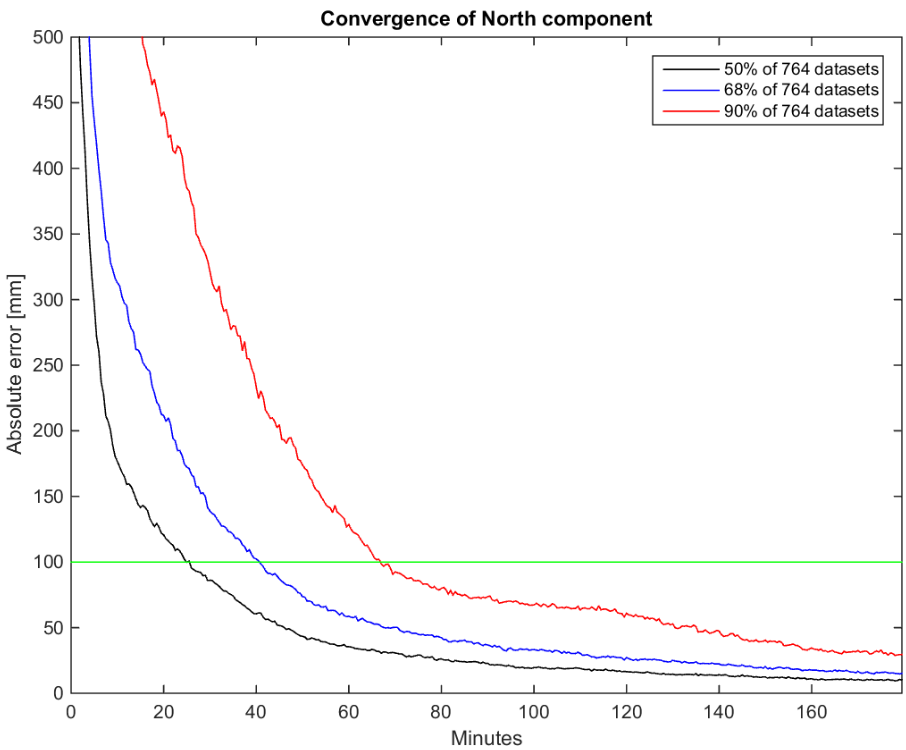


Figure 4.4: Offset between the North component and the empirical mean for all datasets using IGS final products. This represents kinematic positioning. The lines represent the limits under which 50, 68 and 90 percent of the datasets fall.

In this case all datasets from the two days using the IGS final products are combined and the convergence time is analyzed in the way described. The North components in shown in figure 4.4. Half of the datasets manage to reach the convergence criterion of 1 decimeter in under 30 minutes. The 68% limit is around 40 minutes, which is acceptable for kinematic positioning. The jump to the 90% is larger, indicating that there are quite some datasets that converge very slowly. For the East component the results is shown in figure 4.5. The performance of the East component is worse than the North component (the difference in performance is explained in section 5.1.1). The 50% limit for the convergence time is only reached after 50 minutes. Again the jump to the 90% line is large.

For both these figures the important consideration is that the model used is weak. There is only a dynamic model on the carrier phase ambiguities, all others are unlinked in time. Even though the 1 decimeter criterion is reached after a certain time the solution keeps improving after that. Even though it takes more than an hour for the 90% limit to reach the decimeter level after three hours the precision is already at 5 centimeter (both for the North and East components). For the computation of the empirical mean, which is used in computing the offset for every epoch, the first four hours have been skipped. The plots for the kinematic tests prove the necessity of this choice, as the solution keeps improving in the first few hours. The weakness of the model for the kinematic tests makes comparison with external sources not useful. This comparison is made for the static tests.

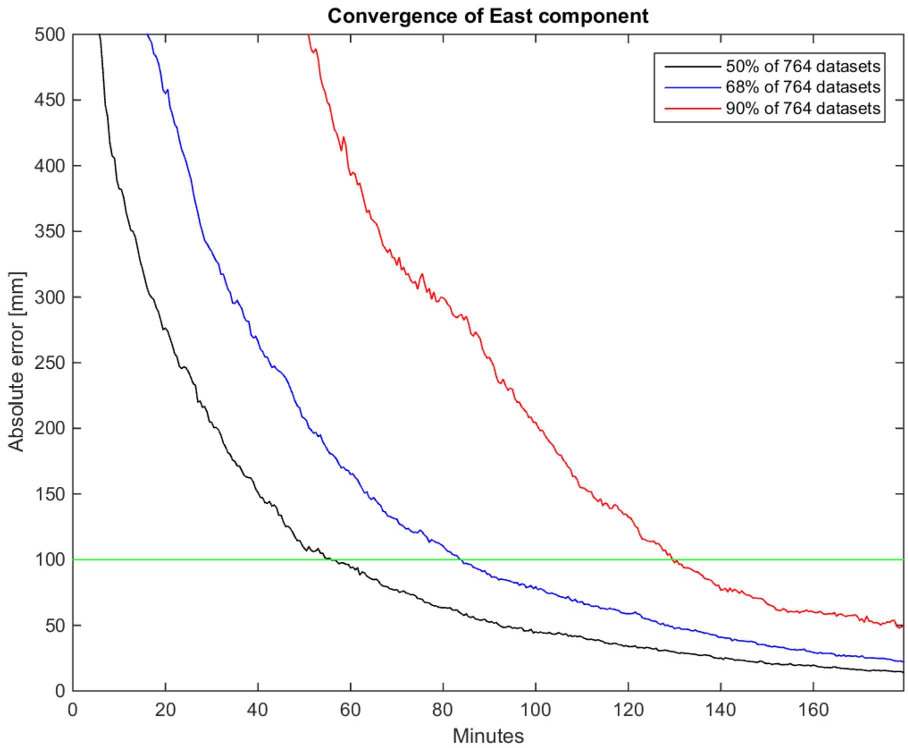


Figure 4.5: Offset between the East component and the empirical mean for all datasets using IGS final products. This represents kinematic positioning. The lines represent the limits under which 50, 68 and 90 percent of the datasets fall.

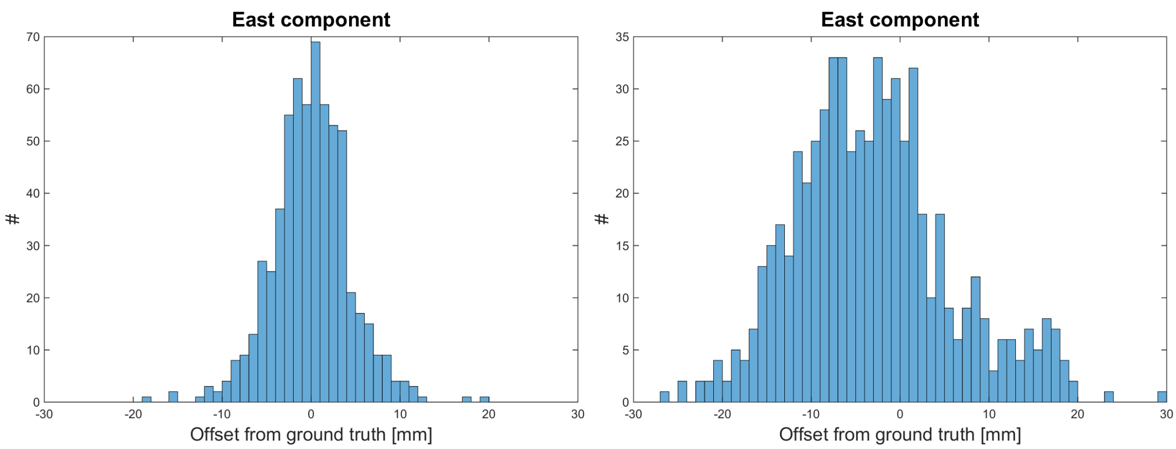
4.1.2. Dual frequency static positioning

The second step is processing the available datasets using static mode. In this case the model in strengthened by applying more constraints. The position of the receiver is assumed not to change. Due to the constraints the fluctuation of the estimated position is expected to be smaller. The results for all the datasets are provided in table 4.2. Since the estimated position converges with these settings it is now also useful to compare the position with an externally provided ground truth. This can only be done for the stations for which the ground truth is provided. If the data is available the ground truth is compared to the last hour of every dataset. The result should be very stable at this point, but relying on a single data point (for instance the very last epoch) is risky. The results for this comparison are also shown in table 4.2.

Day	02/09/2014		14/12/2013	
Year - day of year	14-245		13-348	
Stations	357	357	408	407
Orbit & clock products	IGS final	CNES real-time	IGS final	CNES real-time
Mean σ_E [mm]	2.6	4.2	3.2	5.7
Mean σ_N [mm]	1.4	1.9	1.4	2.4
Mean σ_{up} [mm]	5.2	6.5	5.8	8.2
Stations with known ground truth	316	316	306	305
Mean offset East [mm]	0.0	-1.0	0.0	-5.3
Mean offset North [mm]	0.3	-1.2	2.6	-1.4
Mean offset up [mm]	7.3	10.3	1.6	1.9

Table 4.2: Two processed days with two different sources for orbit and clock products. The average standard deviations are given for the North, East and up components. For the receiver for which the ground truth is known the mean offset between the estimated position and the ground truth is provided.

The empirical standard deviations of all three components is significantly lower than in the kinematic case. This holds for both the IGS and CNES orbit an clock products and also for both the days used for testing. The missing data for the CNES products still influences the results, but the effect is minimal. The constraint on the position is the largest factor is this. For the kinematic case there was no constraint on the estimated position at all, which means that the effect of problematic observations is immediate. For the static case a lot of epochs with problematic data are required for any significant change in the estimated position. Since the number of epochs with problematic orbit and clock products is limited the full day of data can be used for the static case. Because the empirical standard deviations are this small, histograms for these do not provide additional information.



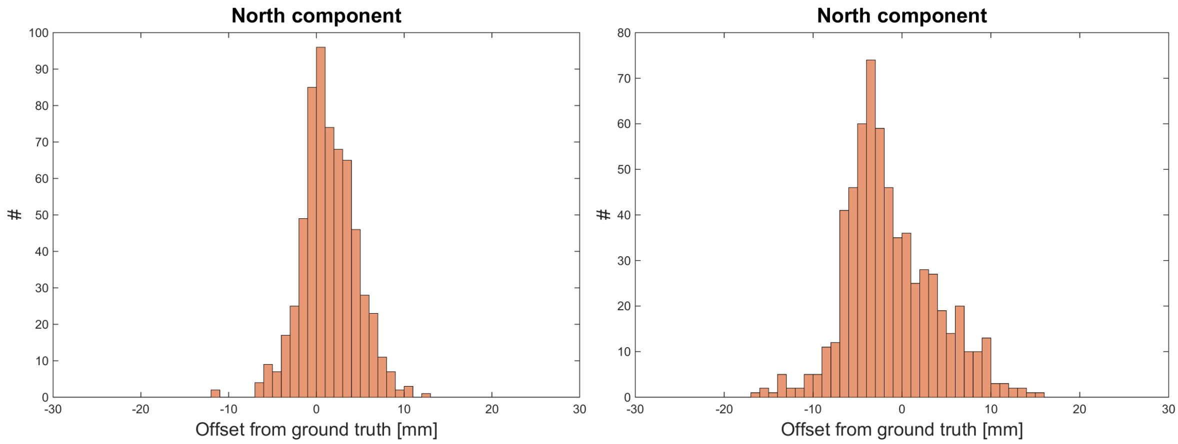
(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure 4.6: Histograms for offset between estimated position and ground truth in the East component. Static positioning is used.

Also in table 4.2 are the offsets with respect to the ground truth. For the North and East components there is little difference. This holds for both the IGS and CNES products and for both days. Only the East component of the day in 2013 using the orbit and clock from CNES shows a somewhat larger mean offset. However this is still below the centimeter level. Figures 4.6 and 4.7 provide the histograms for the difference between estimated position and ground truth. For CNES the histograms are in both cases significantly wider than the IGS counterparts. Therefore this slightly higher value in the range of 5 millimeter is not considered problematic. The histograms for the case of IGS products for the North and East components are centered around the zero offset.

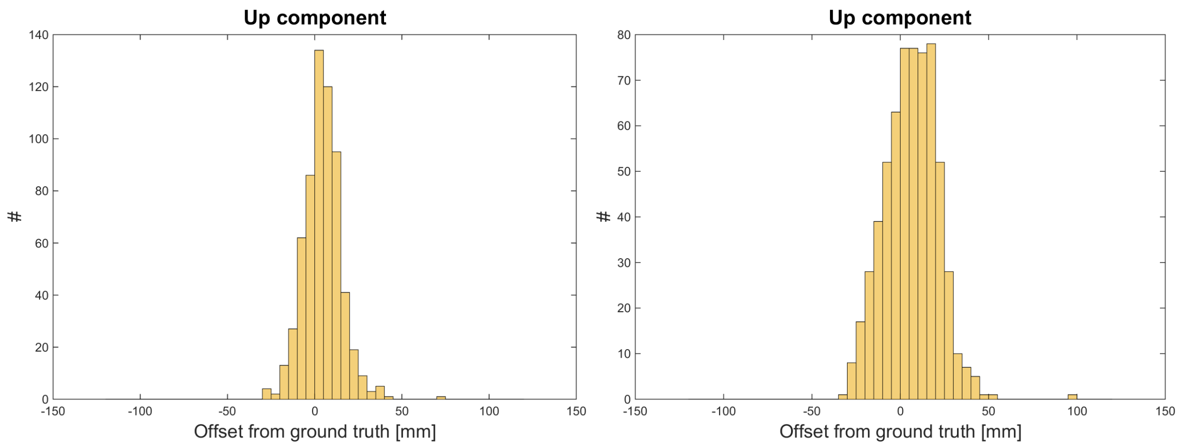
For the up component the mean offset is a little larger. However only for the case of the CNES orbit an clock products the value is slightly over a centimeter. More important however are the histograms. These are shown in figure 4.8 and indicate that, when taking all datasets into account, there is no large bias for the height component. For the results using the CNES products the histograms indicate a less clear distribution. The spread of the results is larger and more outliers are visible, both caused by the poorer quality real-time products. Appendix A shows the same table and histograms for a prior situation in which there still was a bias in the up component, but this was resolved.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure 4.7: Histograms for offset between estimated position and ground truth in the North component. Static positioning is used.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure 4.8: Histograms for offset between estimated position and ground truth in the up component. Static positioning is used.

To investigate the convergence time the same plots are given as for the kinematic case. For the East component the results are shown in figure 4.9. Again all datasets using the IGS final products are combined. When comparing this to the kinematic case (figure 4.4) a clear improvement. 68% of the datasets converge to decimetre level within 10 minutes (compared to 40 minutes in kinematic). 90% of the datasets still reach the criterion in 20 minutes, which is a significant step from the 70 minutes of the kinematic case. However the North component is expected to be the strongest of the two. The East component is shown in figure 4.10. 68% of the datasets converge to the criterion in slightly more than 30 minutes. 30 minutes is used as a benchmark in this case [2] that should be reached under 'normal conditions'. The 90% curve shows the problem with the definition of normal conditions. 90% of the datasets reach the convergence criterion after 70 minutes. This is better than the kinematic scenario in which this was 130 minutes. A lot of factors, that depend on the receiver type and the geographical location, play a role. It is not trivial to determine which of these datasets fall under the description of normal conditions. The convergence time shown for the 68% case does indicate that the user software is performing as expected for the static scenario, but further investigation of less optimal datasets is required.

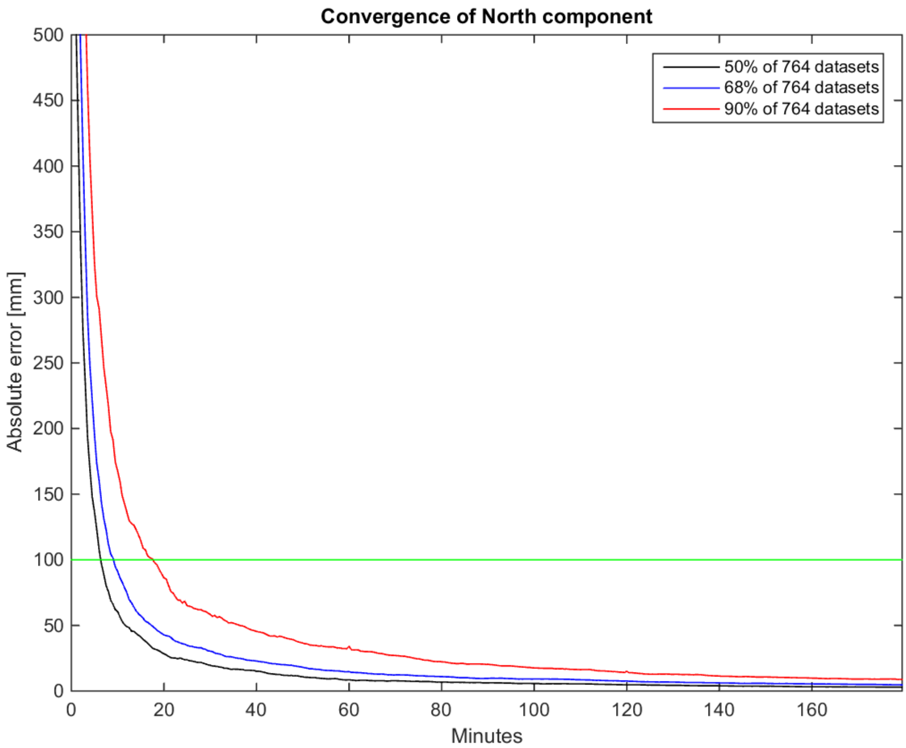


Figure 4.9: Offset between the North component and the empirical mean for all datasets using IGS final products. This represents static positioning. The lines represent the limits under which 50, 68 and 90 percent of the datasets fall.

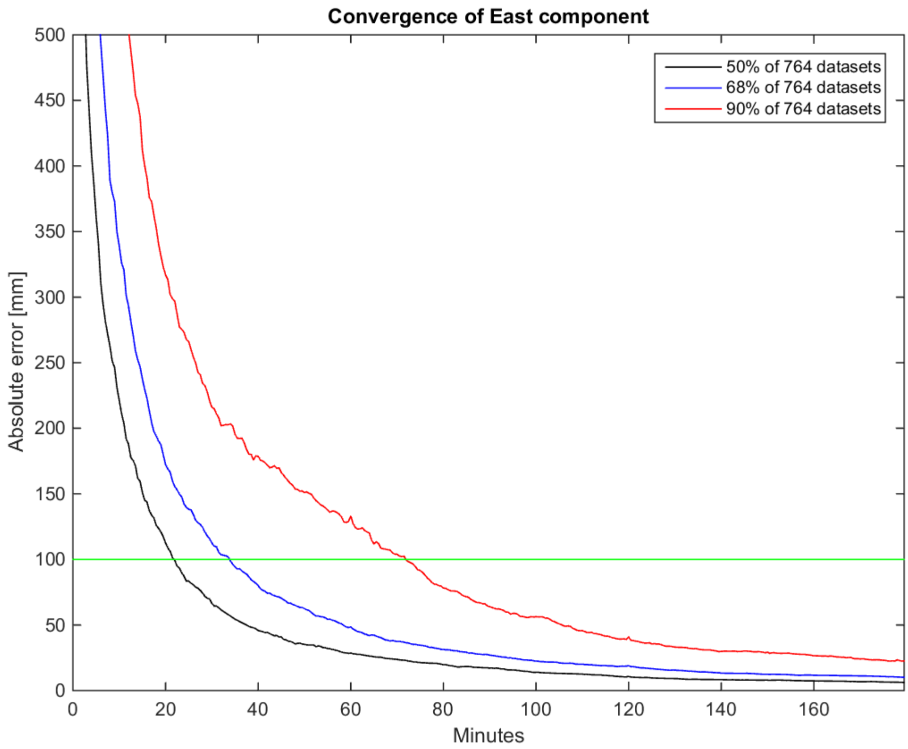


Figure 4.10: Offset between the East component and the empirical mean for all datasets using IGS final products. This represents static positioning. The lines represent the limits under which 50, 68 and 90 percent of the datasets fall.

4.1.3. Single frequency with GIM

It is not impossible to perform standard PPP with only a single frequency. However to do this additional information on the ionosphere is required. This can be done by using Global Ionosphere Maps (GIM) provided by IGS. These GIM are coarse maps of the ionosphere for the entire Earth for an entire day. With this information the model is now solvable. The results from only a single frequency are not expected to be as precise as the dual frequency results. It does depict a realistic scenario as most budget GPS receivers can only receive signals on a single frequency. Kinematic positioning is possible for this scenario, but the precision is poor. Static positioning can still give a solution. This is tested for both the days using GIM for both the days. In this case only the IGS orbit and clock products are used. The results are summarized in table 4.3.

The empirical standard deviations of the estimated position are higher for this scenario than for the dual frequency case. Taking into account the poor precision of the GIM the results are as expected. The precision for all three components is below the decimeter level. The standard deviations for the three components are higher than found by van Bree and Tiberius [19] (0.15 m for horizontal, 0.30 m for vertical), but in the same order of magnitude. All tests are done with the same settings for all receivers. Optimization of these settings per station can improve these results. The comparison with the ground truth shows good results for both the North and East components. There is a larger mean offset than for the dual frequency case, but taking into account the decreased precision these centimeter level variations are sufficiently accurate. There is still a problem for the height component as the offset there is above the decimeter level. The same vertical offset was also found for static positioning by van Bree et al. [20]. For a single frequency solution better results can be achieved, but for that case a more precise map of the ionosphere has to be provided.

4.1.4. Same station, multiple days

Processing stations all over the world provides a lot of different situations useful for analysis. As indicated previously some aspects influencing the solutions are however different for each of the datasets. Many differ-

Day	02/09/2014	14/12/2013
Year - day of year	14-245	13-348
Stations	357	403
Orbit & clock products	IGS final	IGS final
Mean σ_E [mm]	22.1	29.6
Mean σ_N [mm]	12.3	20.3
Mean σ_{up} [mm]	49.0	72.7
Stations with known ground truth	316	302
Mean offset East [mm]	6.8	-1.3
Mean offset North [mm]	4.2	14.9
Mean offset up [mm]	-100.2	-145.9

Table 4.3: Two days of data processed with single frequency observation data (L1). For static positioning the average standard deviation of the three components for all datasets is provided. For the stations that have an external ground truth available the offset from the ground truth is given as an average.

ent receivers and antennas are used for the ground stations, but also the satellite geometry during the day is widely different for the global set of stations. By using a single station but processing data for a full year, the receiver and the antenna are the same for every dataset. Local effects on the chosen station are very likely to repeat daily. Because the GPS satellites have ground tracks that almost repeat daily, the satellite geometry is expected to be very similar between different days.

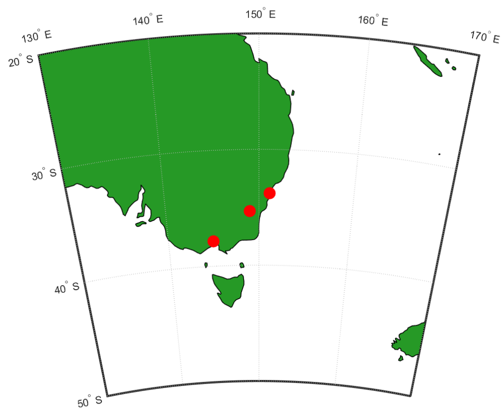


Figure 4.11: IGS stations selected for multiple day testing. From left to right: MOBS, STR2 and SYDN.

To test this, stations have to be chosen that have been operational for a longer period. The year 2014 was chosen for this, being the most recent year with full data. The stations selected must have observation data for, almost, the entire year. Secondly the antenna and receiver must not have been replaced during the period. Both these requirements can be checked via the stations logs. Three stations have been selected for the tests, all in Australia. The stations are *MOBS* in Melbourne, *STR2* in Canberra and *SYDN* in Sydney (figure 4.11 shows the selected stations). Choosing stations that are this close together allows the same stations to be used for testing PPP-RTK at a later stage.

The results for station STR2 immediately show a problem with analyzing the results for a large number of days. Figure 4.12 shows the results for a single day for STR2. While for large parts of the day the solution is stable, there are three epochs in which the solution has to converge again. The cause is loss-of-lock for all satellites in all three cases. With full kinematic processing the large jump in position cannot be avoided. There is no large problem as the solution quickly converges after the loss-of-lock, but when statistics on the precision are computed for such a day the standard deviations indicate poorer performance than is actually the case. Since this station shows results like this for many days during the year generating histograms for the standard deviations is not going to give a clear distribution.

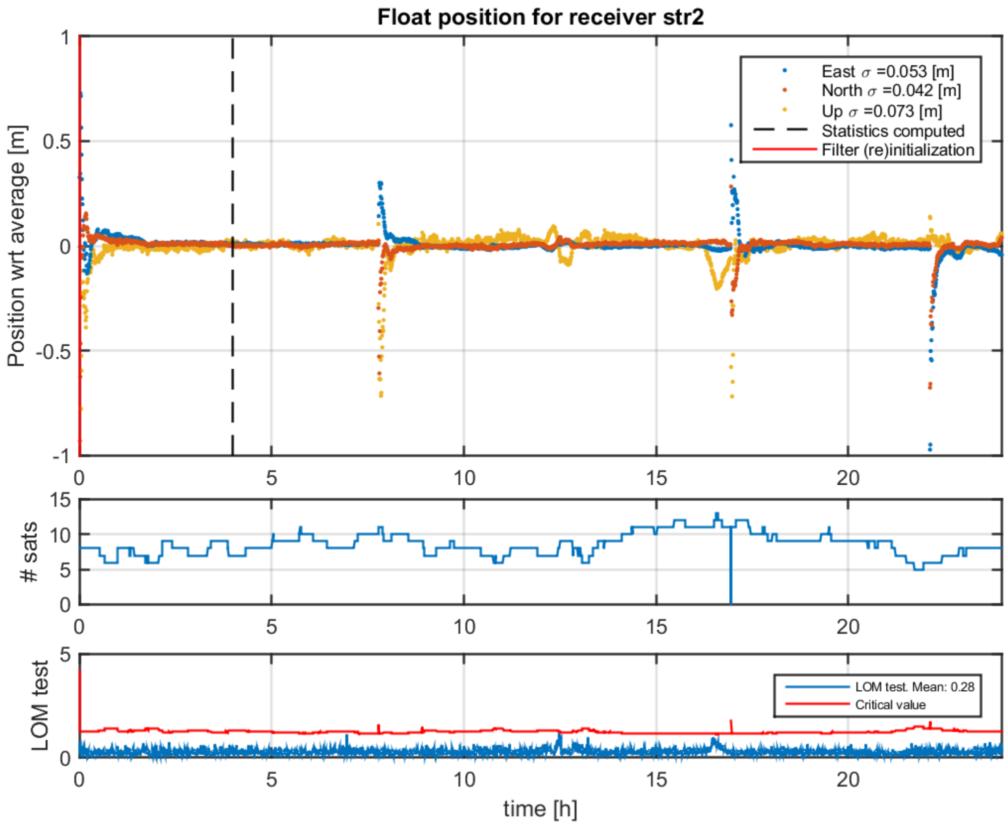
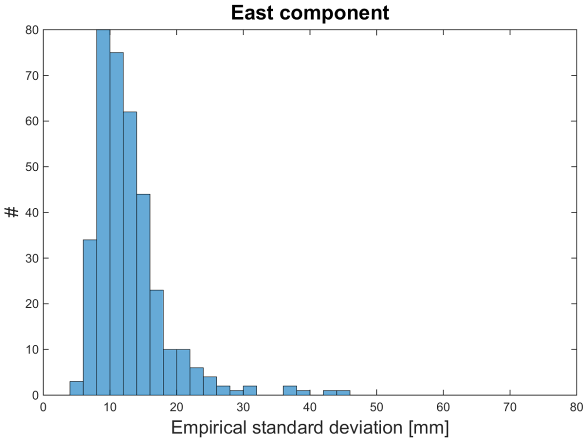


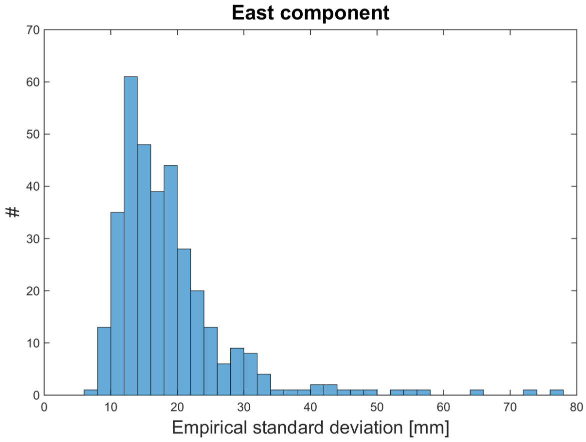
Figure 4.12: Summary of PPP results for station STR2 on day 112 of 2014.

For station MOBS and SYDN these histograms are provided in figures 4.13, 4.14 and 4.15. The histograms for all three components show that the station in Melbourne (MOBS) performs better. The histograms peak at lower standard deviations, but also contain less outliers. One reason for the different performance of the two station is that the receiver types are different (for the antenna the same type is used). For different receivers different settings for the formal precision of the code and phase measurements can be used to improve the results. However for all tests done these settings are kept the same. Modifying these settings can improve the results by detecting outliers that otherwise are ignored. On the other hand a too strict setting discards many observations, potentially leaving too few observations to reach a stable solution.

Even for the best of the two stations (MOBS) there are still days for which the datasets give poor results. Also some disturbances repeat for several days in the datasets around the same time. This is likely due to the GPS constellation repeating daily. These effect make it, as was the case for the global datasets, not useful to investigate the distribution of the histograms. When using the same receiver, antenna and geographical location, effects are still present that are not attributable to random noise in the observations.

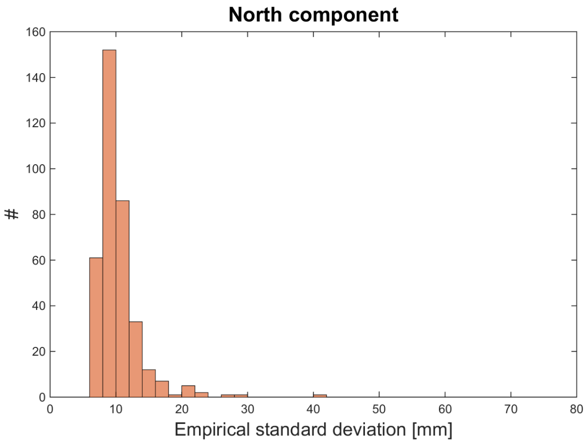


(a) IGS station MOBS.

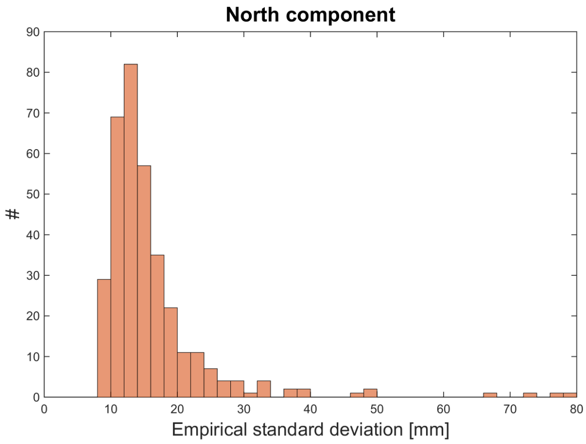


(b) IGS station SYDN.

Figure 4.13: Histograms for empirical standard deviation of the East component for all days. Kinematic positioning is used.

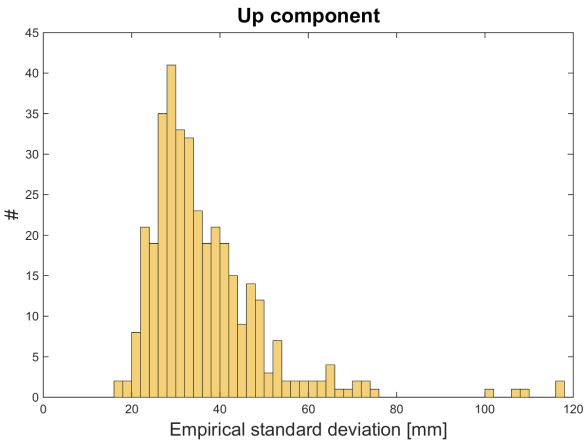


(a) IGS station MOBS.

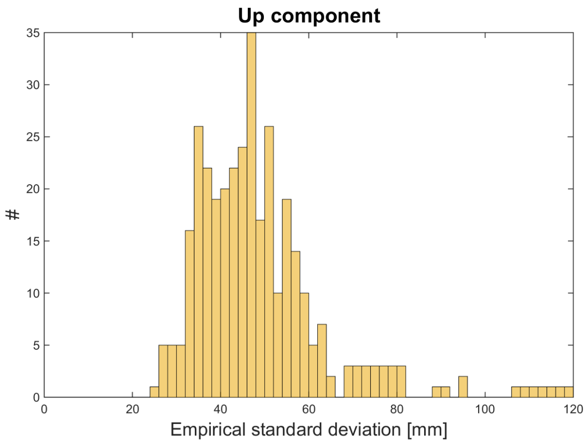


(b) IGS station SYDN.

Figure 4.14: Histograms for empirical standard deviation of the North component for all days. Kinematic positioning is used.



(a) IGS station MOBS.



(b) IGS station SYDN.

Figure 4.15: Histograms for empirical standard deviation of the Up component for all days. Kinematic positioning is used.

4.2. Multi-GNSS

The analysis of the Multi-GNSS PPP performance of the user software is based on two days of data. The first day is the 26th of November 2014 (14-330). The second day is June 17th 2015 (15-168). For both days some datasets were excluded because of gaps in the observation data. Processing these datasets did not result in problems, but the results deteriorated due to the data gaps, which cannot be controlled by the user software. For the remaining datasets comparison of the results is not straightforward. Not every RINEX observation file contains observations for each of the constellations, so for each comparison another subset from the observation data has to be chosen. Most observations files indicate in the header which constellations are present. In some cases, even though the header indicates the presence of a constellation, no satellites are tracked. One example is the Japanese station AIRA, which indicates the presence of BeiDou and Galileo, but no satellites from these constellations are ever tracked. The Swiss station ZIM3 tracks QZSS but the satellite is at such a low elevation that it is never used for positioning.

Complexity of the comparison is increased by the different available orbit and clock products. Three different sets of orbits and clock products are used for the analysis. The first are the IGS final products. These use the greatest number of fixed reference stations and are therefore expected to have the highest quality. The downside is that these products contain only information for GPS (Glonass is also present, but is not used in the user software). The second set of orbit and clock products are GBM, which are provided by GFZ in Potsdam. These products include GPS, BeiDou, Galileo and QZSS but the precision for GPS is poorer than the IGS final products. The main reason is that the number of fixed reference stations used for the computation of the orbits is different. The third set of products are the GRM products, the multi-GNSS products provided by CNES. These include orbits for GPS and Galileo.

The multi-GNSS tests with the new constellations are done for day 15-168, because a total of 7 Galileo satellites were available, compared to only 4 on day 14-330. Day 14-330 is used for testing modernized GPS.

4.2.1. GPS

When comparing all different scenarios care has to be taken for these comparisons to be fair. To ensure this, every comparison concerns a single change in settings. The benchmark is the case of positioning using GPS dual frequency and the IGS final orbit and clock products. This scenario can be tested on all available datasets, as all have at least dual frequency observation data for GPS. This first scenario is also used to exclude bad datasets. If the results do not have information for all epochs of the day in this scenario the observation data is not used for any later tests. All datasets that give poor performance are also investigated. Most of these cases represent a sudden reception issue for a number of epochs. Because the testing is done in kinematic mode this has an immediate detrimental effect on the solution. These datasets are likely to show this problem for all other kinematic tests and are therefore not useful for comparing the performance.

All multi-GNSS scenarios tested are affected by the lack of antenna PCO and PCV information on the side of the receiver. For the satellite side this information has recently become available [11]. The IGS provided calibrated value for the PCOs and PCVs for the GPS L1 and L2 frequencies. For observations on other frequency bands the software approximates the phase center corrections by taking either the values of L1 or L2, depending on which frequency is closest. To investigate what the effect is of these calibrated values they can be set to zero in the user software. The statistics derived from all datasets with and without phase center corrections are provided in table 4.4. As all datasets have GPS observations this comparison can be made for datasets from both days. The effect on the precision is immediately clear. All three component have standard deviation that is three times as large, compared to the benchmark. Setting the values for the PCO and PCV to zero is however more extreme than what is done for the new constellations. By mapping the phase center for, for instance, the L5 to the L2 band an approximation is made that is likely to be closer to the true value than the case of selecting a zero value. The problem here is that for some antennas this approximation is quite accurate while for other the difference is larger. Without actual calibrated data for the new frequencies this assumption is made from the comparison between the offsets for L1 and L2 for several antennas. For some the PCO can differ by only a few millimeters, while for other types the difference can be several centimeters. Since the quality of the solution is determined from the empirical mean of the estimated position only the difference between the phase centers is of importance. A comparison with a ground truth is this case is not practical as the vertical component of the PCO is large for any antenna. The results for this test show that it is important to keep in mind that a wrong assumption for the phase center corrections deteriorates the solution, but the exact effect is unknown as long as no calibration data is available for the newer signals.

Day	26/11/2014		17/6/2015	
Year - day of year	14-330		15-168	
Stations	85	85	105	105
Orbit & clock products	IGS Final			
PCO/PCV	Corrected	Ignored	Corrected	Ignored
Mean σ_E [mm]	15.7	44.6	15.2	44.1
Mean σ_N [mm]	14.3	32.9	12.2	29.9
Mean σ_{up} [mm]	47.4	118.7	36.3	114.8

Table 4.4: Results for kinematic positioning for dual frequency GPS. Two processed days with orbit and clock products from IGS, but with and without antenna PCV and PCO corrections. The average standard deviations are given for the North, East and up components.

The next comparison is the inclusion of the L5 signal being transmitted by the newest generation of GPS satellites. Additional signals are expected to improve the result, but processing the new data is difficult as only a few satellite broadcast the new signal. As with all newer signal no antenna calibration data is available for the receiver side, forcing the L5 to use the same PCOs and PCVs as the GPS L2. Again this scenario can be tested for all datasets on both days. The results are given in table 4.5. Is important that the number of stations used for the comparison change when compared to the dual-frequency GPS case. Even though all datasets have GPS observation, not all contain observations on the third frequency. For a fair comparison the datasets that do not have three frequency data are also excluded from the dual frequency benchmark. For both days there is a decrease in the precision when the L5 signal is taken into account. It is likely that one source of the decrease is the mapping of the L5 antenna PCOs and PCVs to the L2. With the L5 data for a single antenna it would be possible to determine the exact effects of this mapping. Without this information it is only possible to assume that this is the cause of the decreased precision, but it is not possible to exclude any other potential mis-modeling effects. For this reason any further tests are done using GPS dual frequency as a benchmark and not the triple frequency case.

Day	26/11/2014		17/6/2015	
Year - day of year	14-330		15-168	
Stations	82	82	101	101
Orbit & clock products	IGS Final			
Frequencies	L1/L2	L1/L2/L5	L1/L2	L1/L2/L5
Mean σ_E [mm]	15.6	19.7	13.6	22.3
Mean σ_N [mm]	14.2	17.4	11.4	17.2
Mean σ_{up} [mm]	47.7	57.5	34.2	56.6

Table 4.5: Results for kinematic positioning for dual and triple frequency GPS. Two processed days with orbit and clock products from IGS. The average standard deviations are given for the North, East and up components.

For later comparisons made between different GNSS constellations it is important to investigate the effect of the different quality for each of the orbit and clock products. For day 15-168 the performance of the GBM and GRM orbit and clock products is compared to the IGS final products. The results are provided in table 4.6. Both the multi-GNSS products give performance that is slightly worse than their IGS final counterparts. The performance of the GRM products is better than the GBM products. However the GBM product are still used for testing as these contain data for more constellations. The results for GPS only for these products is important for later comparison to scenarios for which more constellations are used.

4.2.2. BeiDou

For the analysis of the results when using the BeiDou constellation a number of different comparisons must be made. Stand-alone positioning is possible using BeiDou on day 15-168. Out of the 105 observation files available for day 15-168 a total of 68 indicate in the RINEX header that BeiDou observations are available. All of these files are processed without errors, but only 39 actually provide positions at a point of the day. From

Day	17/6/2015		
Year - day of year	15-168		
Stations	106	105	105
Orbit & clock products	IGS Final	GBM	GRM
Mean σ_E [mm]	13.6	19.3	15.5
Mean σ_N [mm]	11.4	15.9	12.8
Mean σ_{up} [mm]	34.2	43.8	37.9

Table 4.6: Results for kinematic positioning for dual frequency GPS. Two processed days with orbit and clock products from IGS, GFZ and CNES. The average standard deviations are given for the North, East and up components.

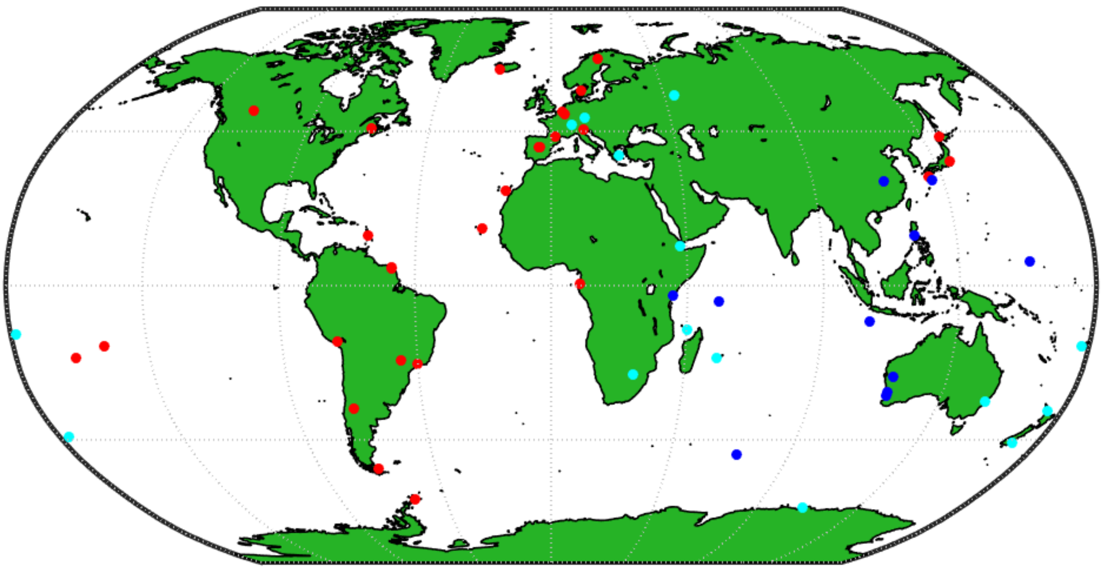


Figure 4.16: All MGEX stations on day 15-168 indicating in the RINEX header that BeiDou observations are available. In red the stations that provide no solution (29). In cyan the stations that do provide a solution, but with poor results (18). In blue the stations that provide a solution that is not problematic (21).

these 39 another 13 do not have enough satellites in view for the entire day to provide a stable solution. The stations that have BeiDou observations and the subset that give results are shown in figure 4.16.

The map clearly shows the importance of the geographical location of the receiver. The closer to China, the better the results. The only exceptions are the three stations in Japan. The reason for this performance can be directly seen in the RINEX observation file. These stations indicate in the header that BeiDou observations are available, but no satellite is ever tracked. The criterion for the stable solution is in this case set as a maximum standard deviation for the height component of 0.4 meter. This exact value is arbitrary, but any datasets with standard deviations higher than this have a shortage of satellites at some point of the day. Because there is no dynamic model on the position the solution diverges at this point. The choice of 0.4 meter shows the increase in performance when getting closer to the latitude of China. The statistics of the 21 datasets are not expected to be representative for the performance of BeiDou positioning as the geographical location is of such importance. However as an example a single station is investigated, in this case station CUT2 in Perth. This station has the best performance of the 21 stations when looking at the horizontal components. The performance when using GPS dual frequency is on the same level as the mean computed for all stations (in table 4.6). Table 4.7 shows the results for this station. Even for the best result for BeiDou stand-alone the performance is significantly worse than for GPS, especially the height component. At this stage it is not possible to investigate the influence of the three different segments of the constellation. If, for instance, the geostationary satellites are excluded insufficient satellites remain for stand-alone positioning.

Station	CUT2	CUT2
Products	GBM	GBM
GNSS	GPS	BeiDou
σ_E [mm]	15	25
σ_N [mm]	10	27
σ_U [mm]	32	193

Table 4.7: Standard deviations for three components for station CUT2, using Dual frequency GPS and dual frequency BeiDou

The previous test scenarios all used dual frequency BeiDou. Some MGEX stations track the third frequency signal as well. A total of 28 observation files indicate that the third frequency is tracked. Of these only 14 stations provided position results at some part of the day. 8 of these stations have a standard deviation of the vertical component that is smaller than 0.4 meter. The stations are shown in the map in figure 4.8. The results of these 8 stations can be directly compared. Since the satellites that are tracked are always exactly the same the differences cannot be due to elevation problems. The statistics are shown in table 4.8. The mean standard deviations for all components improve when the third frequency is used. With 8 samples this cannot be considered as proof that the third frequency is improving the results in all scenarios. When investigating the individual results and the improvement between dual and triple frequency only the North component always shows improvement. The other components show improvement for one station and a slight decrease in precision for others. There is no case in which all components are worse in the triple frequency case. However, because of the limited availability of the third frequency further testing is done using only the first two frequencies of BeiDou.

Stations	8	8
Products	GBM	GBM
Frequencies	B1/B2	B1/B2/B3
Mean σ_E [mm]	40.1	36.8
Mean σ_N [mm]	56.5	46.6
Mean σ_U [mm]	205.2	182.3

Table 4.8: Standard deviations for eight stations on 15-168 using dual and triple frequency BeiDou.

When combining BeiDou with GPS, all stations that track BeiDou provide positioning results. Out of the 68 datasets for day 15-168 that indicate the presence of BeiDou satellites in the RINEX header, 65 track the satellites at some point of the day. This scenario is tested both using kinematic and static positioning. The results are summarized in table 4.9. Even on the global level the addition of the BeiDou satellites degrades the precision of the results. This is especially clear for the kinematic positioning, where the effect is largest.

Stations	65	65	65	65	65	65
Products	GBM	GBM	GBM	GBM	GBM	GBM
GNSS	GPS	GPS+BeiDou	GPS+BeiDou*	GPS	GPS+BeiDou	GPS+BeiDou*
Setting	Kinematic	Kinematic	Kinematic	Static	Static	Static
Mean σ_E [mm]	16.9	24.6	13.8	3.1	3.5	3.1
Mean σ_N [mm]	16.9	26.0	15.7	1.4	1.5	1.4
Mean σ_U [mm]	42.3	66.2	41.2	5.5	6.2	5.6

Table 4.9: Standard deviations for three components for stations on day 14-330 using GPS and a combination of GPS and BeiDou. The third option (*) is the combination of GPS and BeiDou without the BeiDou geostationary satellites. The results are shown for both static and kinematic positioning.

The geostationary satellites in the BeiDou constellation are problematic for PPP. Estimation of the carrier phase ambiguities benefits from a change in receiver-satellite geometry over time. For the geostationary satellites the position with respect to a stationary receiver on the ground hardly changes. Estimation of the ambiguities is not adversely affected by this, but neither does it improve the estimation. The same lack of geometry change can be problematic for orbit determination, which can lead to a decreased precision when

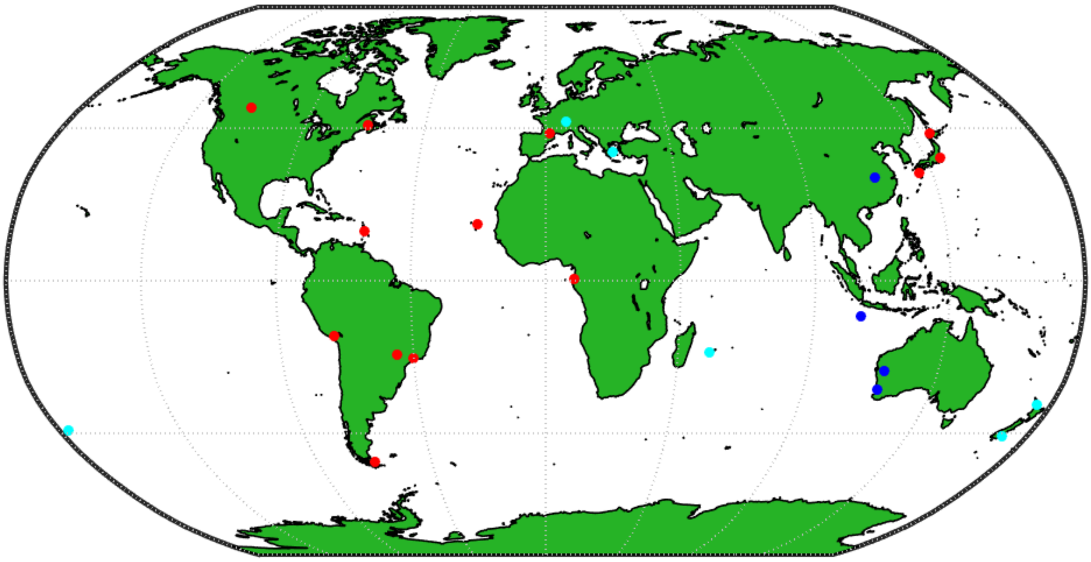
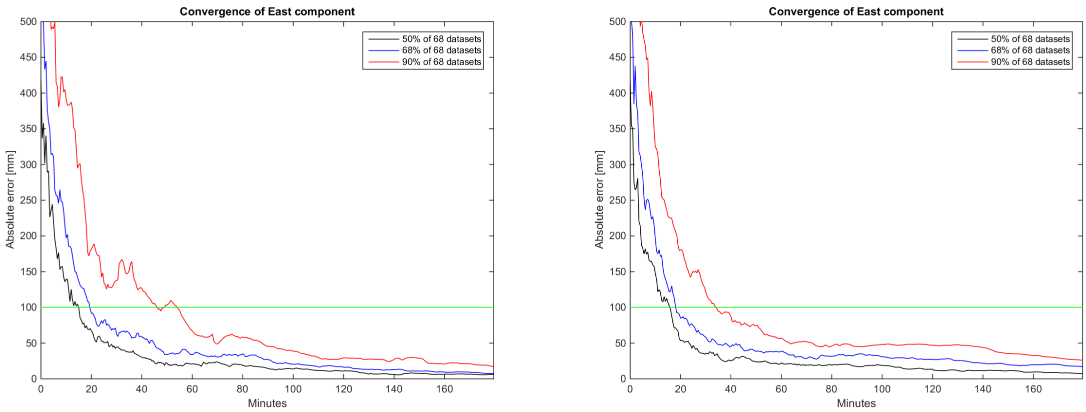


Figure 4.17: All MGEX stations on day 15-168 indicating in the RINEX header that triple frequency BeiDou observations are available. In red the stations that provide no solution (14). In cyan the stations that do provide a solution, but with poor results (6). In blue the stations that provide a solution that is not problematic (8).

the geostationary satellites are used. Thirdly the geostationary satellites have been shown to suffer from multipath effects on a larger scale than the other satellites in the constellation [21].

For these reasons another option is tested. This is the combination of GPS and BeiDou while excluding the five geostationary satellites. The results for this scenario, in both kinematic and static cases, are also given in table 4.9. With respect to the results for stand-alone GPS shown in table 4.9 there is now a slight improvement. The results when leaving out the geostationary satellites are significantly better than the first combination of GPS and BeiDou. Further investigation of this is possible when more BeiDou satellites become available. In that case excluding all the geostationary satellites is an option for stand-alone BeiDou, but at this time not enough satellites remain to test this.

The convergence times for stand-alone GPS and the combination of GPS and BeiDou are shown in figure 4.18. These convergence times are determined from the static solutions. Adding BeiDou improves the convergence of the datasets under the 90% line. There is little improvement for the other two limits, as these convergence times are already low. The 90% limit contains datasets that have a limited number of GPS satellite or poor satellites geometry, in which cases the benefit of the BeiDou satellites is greatest. Figure 4.19 shows the convergence times when the geostationary satellites are excluded. The lower two limits are largely unaffected. The 90% limit is better than the GPS case, but worse than the full GPS and BeiDou combination. Even though the geostationary satellites are problematic, excluding these leaves fewer satellites, counteracting the benefit of excluding the geostationary satellites. However taking into account the better precision it is advisable to exclude the geostationary satellites.



(a) GPS only

(b) GPS and BeiDou

Figure 4.18: Convergence time for the East component for 50, 68 and 90% of the datasets. Static positioning.

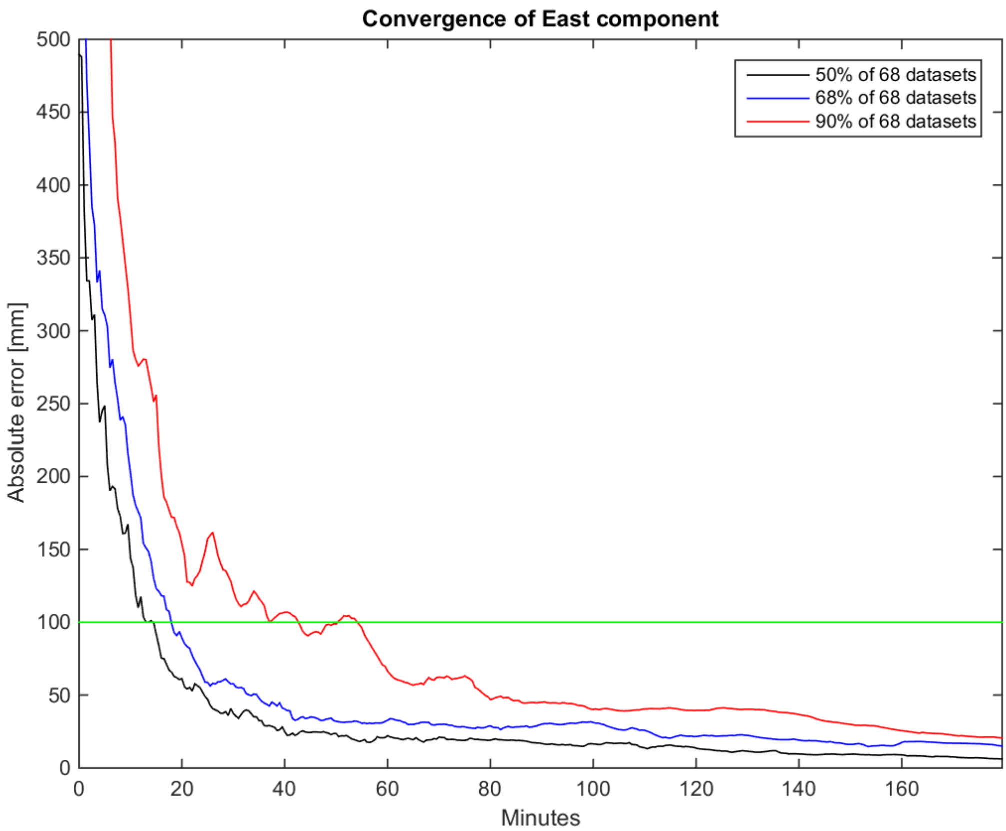


Figure 4.19: Convergence time for the East component for 50, 68 and 90% of the datasets using GPS and BeiDou, but excluding the BeiDou geostationary satellites

4.2.3. Galileo

For Galileo the analysis of the results is again based on day 15-168. For testing the software initially day 14-330 was used for Galileo as well and all issues with processing this data have been resolved. On that day a total of four Galileo satellite were in orbit and transmitting signals. Any increase in performance, when adding the Galileo constellation, is expected to be larger when more satellites are in use. A more recent day was therefore selected. In this case not the most recent day, as on for example day 15-200 (July 19th 2015) only five satellites were tracked by the receivers in the MGEX network. On the day 15-168 all satellites in orbit were transmitting and were tracked by the receivers. This leads to a total of seven available Galileo satellites for the whole planet. With these seven satellites it becomes possible for some receivers to do stand-alone positioning for a short period of time. With a duration this short it is useful to see whether the software is correctly processing the results, however the quality of the solution is not expected to be good.

As the Galileo constellation is designed to provide global coverage analyzing the results is more straightforward than the BeiDou case. All RINEX headers for the receivers that provided data for day 15-168 indicate that the Galileo constellation is being tracked. However, similar to the BeiDou case, three receivers in Japan never track any satellites during the whole day. After discarding these three observation files a total of 102 remain. All these observation files are processed using only GPS and using the combination of GPS and Galileo. Since both the GBM and GRM products provide correction for Galileo the tests can be done using both products. This processing is first done in kinematic mode. The mean standard deviations are computed and shown in table 4.10. The results are different for each of the products. When using the GBM products there is a slight improvement for each of the components. For the GRM products there is a significant decrease of precision for all three components.

Stations	102	102	102	102
Products	GBM	GBM	GRM	GRM
GNSS	GPS	GPS + Galileo	GPS	GPS + Galileo
Setting	Kinematic	Kinematic	Kinematic	Kinematic
Mean σ_E [mm]	19.9	18.4	16.3	20.1
Mean σ_N [mm]	16.6	15.7	13.4	14.9
Mean σ_U [mm]	45.8	45.0	38.9	45.9

Table 4.10: Standard deviations for three components for stations on day 15-168 using GPS and the combination of GPS and Galileo. The results are shown for kinematic positioning.

Here rises a problem for the comparison. It is clear that the performance of the GRM products, when only looking at GPS, is better. When more Galileo satellites are available the same comparison can be done for stand-alone Galileo. It is likely that the results would show that the performance of the GBM products is better for Galileo, while the GRM products provide poor results. This would explain the improvement for GBM and a decrease in precision for the GRM products when Galileo is added. The user platform allows for different products to be assigned for each constellation. For GPS the GRM products can be used, while for Galileo the GBM product are assigned. Under normal circumstances this is not advisable as both products are based on a different set of ground stations, potentially introducing a bias in the solution. The reason behind this bias is that any positioning done with PPP is in fact relative positioning, based on the network of stations generating the orbit and clock products. If this tracking network is not exactly using the same ground truth values for the stations, or uses different stations, the different GNSS constellations actually provide a slightly different position estimate. For testing the option of mixing the orbit and clock products is useful as the differences between the tracking stations used for the products are expected to be small. The results when the two products are mixed are shown in table 4.11. The results indeed indicate that the quality of the orbits and clocks for Galileo are poorer in the case of the GRM products. The East and North components, when GBM is used for Galileo, show the best performance of all combinations. The other mixed version, with GRM for Galileo, shows the worst performance of all options.

For static positioning the results are summarized in table 4.12. The differences in precision are very small for all three components. The effect of adding in the Galileo satellites is however always negative, if it is even visible. The largest effect is visible for the GBM products. With the precise estimates, due to the static positioning, the effect of the antenna PCOs and PCVs is expected to become more clear. For the Galileo signals these values are not exactly determined, but a bias of several millimeters is possible. For some antennas the

Stations	102	102
Products GPS	GRM	GBM
Products GAL	GBM	GRM
Setting	Kinematic	Kinematic
Mean σ_E [mm]	15.3	24.0
Mean σ_N [mm]	13.2	20.7
Mean σ_U [mm]	39.0	59.1

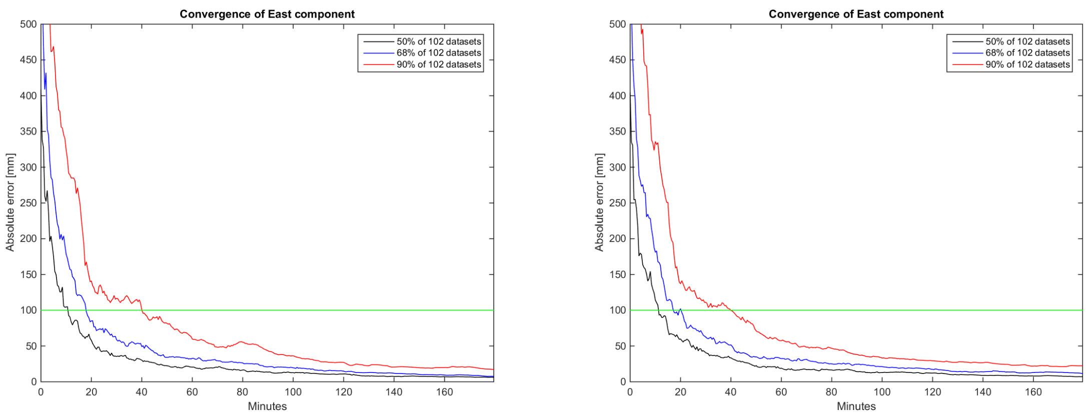
Table 4.11: Standard deviations for three components for stations on day 15-168 using GPS and the combination of GPS and Galileo. The orbit and clock products are mixed for performance comparison.

difference between the phase centers for GPS L1 and L2 can be 4 centimeters. While the mismatch caused by mapping the phase center for the Galileo signal to the nearest match (either L1 or L2) a small error can remain. For static positioning this is likely to generate more of an effect than for the more unstable kinematic solution. Yet again, without any calibrated antenna values for any frequency other than GPS L1 and L2, it is impossible to be sure that this is the cause.

The convergence of the East component, when using GBM products, is shown in figure 4.20. As expected with the small difference in the precision of the position estimates, the convergence time is not visibly affected by the additional satellites. While the standard deviation of the estimated position takes almost the entire day of data into account, the convergence plots do not. The convergence plots for stations all over the world is influenced by the geographical location, as the convergence of the solution depends on the satellites in view at the start of the day. With only seven Galileo satellites in use, the number of datasets that have enough Galileo satellites at the start of the day is minimal.

Stations	102	102	102	102
Products	GBM	GBM	GRM	GRM
GNSS	GPS	GPS + Galileo	GPS	GPS + Galileo
Setting	Static	Static	Static	Static
Mean σ_E [mm]	2.6	2.7	3.0	3.2
Mean σ_N [mm]	1.1	1.3	1.1	1.1
Mean σ_U [mm]	4.2	4.9	4.5	4.5

Table 4.12: Standard deviations for three components for stations on day 15-168 using GPS and the combination of GPS and Galileo. The results are shown for kinematic positioning.



(a) GPS only

(b) GPS and Galileo

Figure 4.20: Convergence time for the East component for 50, 68 and 90% of the datasets

4.2.4. Triple system

GPS, Galileo and BeiDou represent the constellations with the largest number of available satellites that can be processed by the software. Combining the three systems is therefore also of interest. In order to demonstrate this combination the datasets from day 15-168 are used. As orbit and clock products the GBM products are chosen. Due to the poorer performance the BeiDou geostationary satellites are excluded. The number of datasets that can be used for this test is determined by the number of RINEX observation files that contain observation data for BeiDou. This amounts to a total of 65 datasets for day 15-168. All these also have information for Galileo, so there is no decrease in number of testable sets for Galileo. Table 4.13 shows the mean standard deviations for all different combinations using the GBM orbit products in the case of kinematic positioning. The results here are as expected, having more satellites leads to more precise results. The only exception is the height component when only Galileo is added. Overall the improvement in precision is smallest in the height component for the other combinations as well. A potential cause is again the lack of accurate PCO and PCV data, which is most dominant in the vertical direction. Adding either BeiDou or Galileo shows a clear improvement and the combination improves even more.

Stations	65	65	65	65
Products	GBM	GBM	GBM	GBM
GNSS	GPS	GPS + Galileo	GPS + BeiDou	GPS + Galileo + BeiDou
Setting	Kinematic	Kinematic	Kinematic	Kinematic
Mean σ_E [mm]	16.9	15.9	13.8	13.3
Mean σ_N [mm]	16.9	14.7	15.7	13.3
Mean σ_U [mm]	42.3	42.5	41.2	40.7

Table 4.13: Standard deviations for three components for stations on day 15-168 using GPS and combination of GPS, Galileo and BeiDou. The results are shown for kinematic positioning.

The same test is repeated for the static case. The results for this are shown in table 4.14. No improvement is found in this case by adding the other constellations, with respect to GPS. Galileo in this case appears to have the largest effect on decreasing the precision.

Stations	65	65	65	65
Products	GBM	GBM	GBM	GBM
GNSS	GPS	GPS + Galileo	GPS + BeiDou	GPS + Galileo + BeiDou
Setting	Static	Static	Static	Static
Mean σ_E [mm]	3.1	3.4	3.1	3.4
Mean σ_N [mm]	1.4	1.5	1.4	1.5
Mean σ_U [mm]	5.6	6.1	5.6	5.9

Table 4.14: Standard deviations for three components for stations on day 15-168 using GPS and combination of GPS, Galileo and BeiDou. The results are shown for static positioning.

4.2.5. QZSS

The QZSS constellation at this stage only consists of a single satellite. Stand-alone positioning is impossible, but is not the intention for the constellation. The QZSS satellite shares the L1, L2 and L5 frequencies with GPS and is designed to function as a support for these satellites. Again not all receivers in the MGEX network track the constellation. Figure 4.21 shows all stations that indicate in the RINEX header that QZSS is tracked. In blue are the 35 stations that have the satellite in the results. The other 5 are indicated in red and either never track the satellite or at only very low elevations. For instance station ZIM3 in Switzerland only tracks the satellite for 11 epochs on the entire day.

The results when processing the datasets with and without QZSS are shown in table 4.15. The only difference visible is in the East component. However this difference appear larger than it actually is due to rounding. The actual difference between the two averaged standard deviations is less than 0.01 millimeter. Either way the difference is too small to be significant. The reason for this is the number of satellites in the QZSS constellation. A single satellite hardly affects the PPP result. The dominant contribution for a PPP solution (after

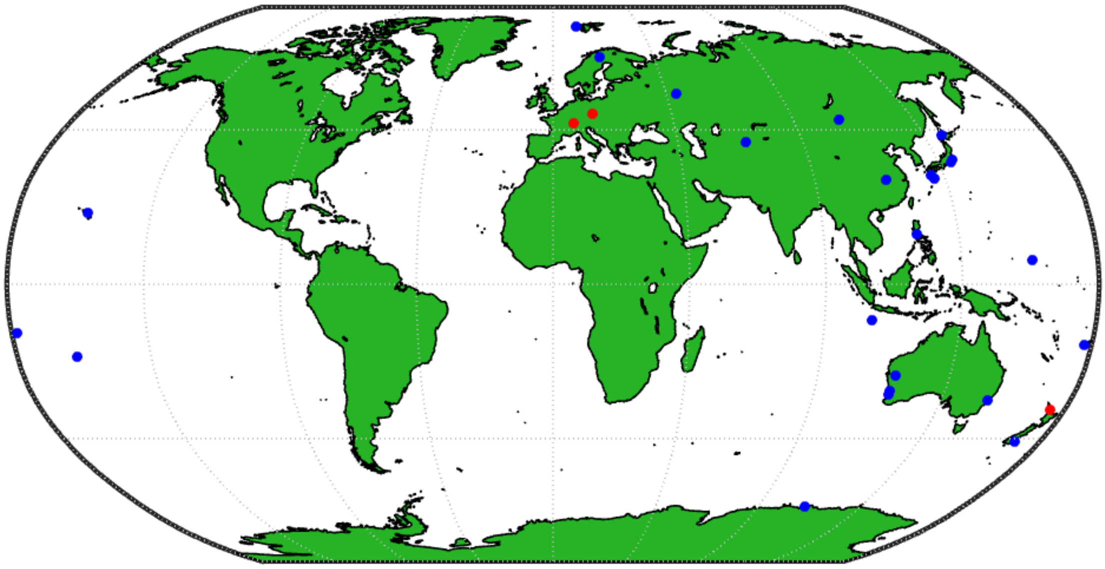


Figure 4.21: All MGEX stations on day 15-168 indicating in the RINEX header that QZSS observations are available. In red the stations that never track the satellite above the cut-off angle (5). In blue the stations that provide actually track the satellite at useful elevations during the day (35).

convergence) is provided by the phase measurements. Since the carrier phase ambiguities are differenced between satellites, nothing remains for QZSS. The code observations are still used, so in that sense QZSS has a small contribution. The results are not exactly equal therefore, but the difference is minute.

Stations	35	35
Products	GBM	GBM
GNSS	GPS	GPS + QZSS
Setting	Kinematic	Kinematic
Mean σ_E [mm]	17.5	17.6
Mean σ_N [mm]	21.9	21.9
Mean σ_U [mm]	48.0	48.0

Table 4.15: Standard deviations for three components for stations on day 15-168 using GPS and combination of GPS and QZSS. The results are shown for kinematic positioning.

As indicated the QZSS constellation does share frequencies with GPS. Another method of combining the systems in the software is available. In this case the constellations can share pivot satellites as long as common frequencies are used. The receiver tracks both constellations and there is a potential bias between the observations of both systems. This bias is referred to as the Inter System Bias (ISB). For the case of standard PPP this bias is fully absorbed into the carrier phase ambiguity, so the ISB can be assumed to be zero. When QZSS and GPS can use a common pivot satellite for differencing the QZSS phase observations can contribute to the solution. The one important assumption is that this bias must be constant. If this is not the case the ambiguities (single differenced between two constellations) are also not constant and therefore problems arise. For the 35 stations the results, when fixing the ISBs, are shown in table 4.16. These result are clearly different from the results in table 4.15. There is improvement for the East component, but decreased precision for North and Up. QZSS provides the only case in which the antenna PCO and PCV correction provide no problems. The frequencies used are the same as for GPS, so this is not the cause of the poorer results. The orbit products from GBM do indicate a lower quality for the QZSS orbits than for GPS. When these orbits become more accurate, the solution should improve for all components (with respect to stand-alone GPS). The changes in the results when fixing the ISB do indicate that this option in the software is handled correctly. The other source of the decreased precision of the solution might be due to the assumption of the ISB being constant. If this is not the case the decreased precision will still show up as the orbit products improve.

Stations	35
Products	GBM
GNSS	GPS + QZSS
Setting	Kinematic
Mean σ_E [mm]	16.5
Mean σ_N [mm]	22.5
Mean σ_U [mm]	50.3

Table 4.16: Standard deviations for three components for stations on day 15-168 using a combination of GPS and QZSS, while treating the QZSS satellite as part of the GPS constellation. The results are shown for kinematic positioning.

5

Test Results: PPP-RTK

The input required by the user platform for PPP-RTK are the precise orbits and the network products. The precise orbits used for processing should match those used by the network. The network products consist of satellite clock corrections, satellite phase biases and optionally ionospheric delays. At a later stage these corrections are actually provided by a network of stations. At this stage the network corrections are still generated from a single station. The user platform does not require information on the number of stations used to generate the network products, so for testing the functionality of the software this is of no influence. For testing the performance the single station products do have an effect. The satellites for which corrections are provided are limited to those in view of the station generating the network corrections (the reference station). If the station acting as user for the software is too far from the reference station, insufficient satellites are usable. Therefore there is a limit on the distance between the two stations. The user software is capable of acting as a network provider. In this case observation data is used to generate products from a observation file, instead of estimating the position. This functionality has been built-in for testing purposes, but is useful to test the PPP-RTK functionality. For all stations providing data on the chosen test day network products can be generated. These corrections can then be applied to any other stations in range allowing for a large number of datasets to test with.

5.1. GPS stand-alone: short baseline

The PPP-RTK functionality of the user platform, when using only GPS, is tested with data that has also been used for testing standard PPP. The main focus is on the observation data from IGS stations available on September 2nd 2014 (14-245). A division is made at this point for testing short and long baselines (distance between the user and reference stations). Over longer distances the number of satellites in view for both stations decreases, which also decreases the expected performance. For short distances it is also possible to make use of the estimates of the ionospheric delays. Having access to accurate ionosphere information also provides the possibility to test the user platform with only single frequency observations.

For day 14-245 a total of 357 stations provide observation data. For short baseline testing all combinations of these stations are made that have a maximum distance of 10 kilometers. The 10 kilometer limit is selected as at this distance information from the reference station on ionospheric delays can still be used. When there ionospheric delays are used even better performance is possible for PPP-RTK. A minimum distance of 1 meter is selected to prevent the same station to be used as both reference and user. Even though the software can process such a combination, the results are not useful for analysis. A total of 69 pairs of stations can be formed from the available observation files. This number should be even, as each station can be used both as reference and user. One station could not be used as reference. The capability of the user platform to generate single station corrections is only for testing and cannot capture problems for all scenarios. The remaining 69 station pairs are used to test the performance of PPP-RTK over short distances.

5.1.1. Dual frequency

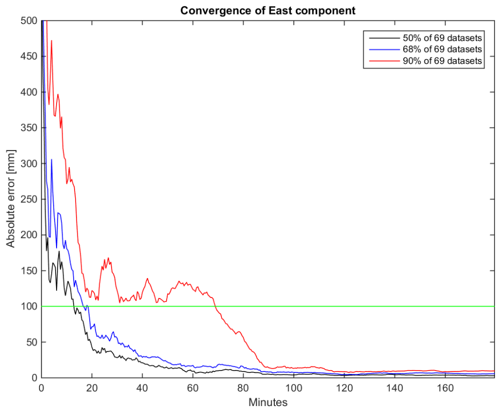
It is important to reiterate the two improvements that PPP-RTK can provide over standard PPP. First is improved precision by fixing the ambiguities. However, this comparison is difficult to make in these tests. The single station products provided from the reference station have a limitation: due to the method used for generating the products [8] the estimated satellite clocks and phase biases must always be used in combination. It is not possible to only use the clocks as these alone are too noisy. If this was possible a direct comparison between the standard PPP and the PPP-RTK results could be made. The difference in this case would be whether or not to apply the phase biases. Since this option is not possible due to the nature of the single station products, only the comparison between float and fixed results can be made. The float solution in this case concerns the estimated position with applied phase biases but without fixing the ambiguities to integers. The fixed solution represents the results when fixing the ambiguities to integer values. Between these two results an increase in precision for the fixed case is still expected, but this is not the precision increase between PPP-RTK and standard PPP. The second improvement expected by fixing the ambiguities to integer is decrease in convergence time. For the PPP-RTK float solution the ambiguities should converge to integer values over time. By resolving the ambiguities to integer this convergence is sped up, which should be clearly visible in the results.

The 69 datasets have been processed in kinematic mode. Only the ambiguities are assumed constant in time, while other parameters are unlinked in time. The resulting average standard deviations for the three component are shown in table 5.1. Both the ambiguity float and fixed results are shown. These float results are more precise than standard PPP, because the clock corrections are generated from a station within 10 kilometers of the user. Some problems, with for instance the orbit products, do not show up for short baselines. If a bias is present in both the estimation on the user and reference side this bias can cancel out because the path from receiver to satellite is similar for both stations. Therefore this bias does not affect the solution. There is improvement in the East and the North component visible after resolving the ambiguities, but the effect is small, due to the float solution already being precise. The up component does not show improvement, which is unexpected. Problems with fixing the ambiguities can occur, but an overall decrease in precision is problematic. Further investigation reveals that the decreased precision is caused by a total of 8 datasets. These individual sets show no improvement in any of the components after the integer ambiguities have been resolved. When excluding these 8 sets, which is shown in table 5.1 as well, there is improvement visible. In the up component it is very small, but no longer is there a decrease in precision. The reason for the poor performance of the 8 datasets is investigated at a later stage.

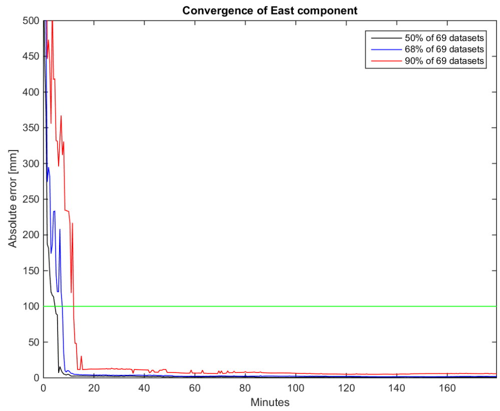
Day	02/09/2014		02/09/2014	
Year - day of year	14-245		14-245	
Stations	69	69	61	61
Ambiguities	Float	Fixed	Float	Fixed
Mean σ_E [mm]	9.5	7.8	9.7	5.3
Mean σ_N [mm]	9.2	8.7	9.2	7.5
Mean σ_{up} [mm]	25.3	29.8	24.8	24.7

Table 5.1: Results for kinematic positioning for dual frequency GPS. Average standard deviation for three components for all datasets and a selected subset of the data.

For static positioning the standard deviations, because these are determined after four hours, are not expected to change a lot after ambiguity fixing. Static positioning is used to investigate the convergence time of the solution with and without ambiguity resolution. The convergence time plots are given for both the ambiguity float and fixed solutions in figure 5.1. The East component is shown here. The improvement is large. Even the best 50% of the datasets show improvement as the time for that line to reach the decimeter level goes from 12 to 6 minutes. The 90% line shows even larger improvement. Some datasets have problems, as the convergence time indicated by the 90% curve shows that more than an hour is needed. However, fixing the ambiguities dramatically decreases the convergence time for the poorer datasets. The jumpy behavior of the curves in figure 5.1b is expected. Ambiguities are fixed every epoch, without considering wrong fixes. Once the ambiguities are correctly fixed there can be a sudden jump in the estimated position. Between two epochs a single dataset can make the change between 89% of the datasets having converged and 91%. As the curves are computed by sorting the datasets the order can change quite a bit between epochs. This causes the jump in the convergence plots to be quite sudden as well.



(a) Convergence for ambiguity float solution.

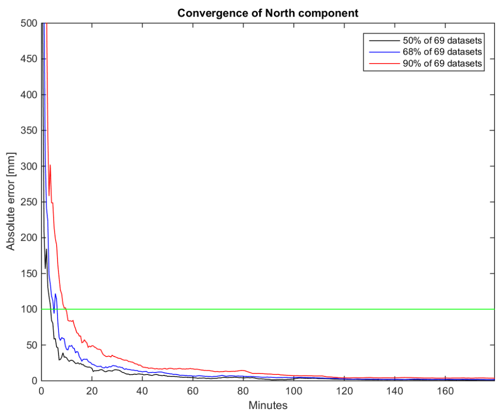


(b) Convergence for ambiguity fixed solution.

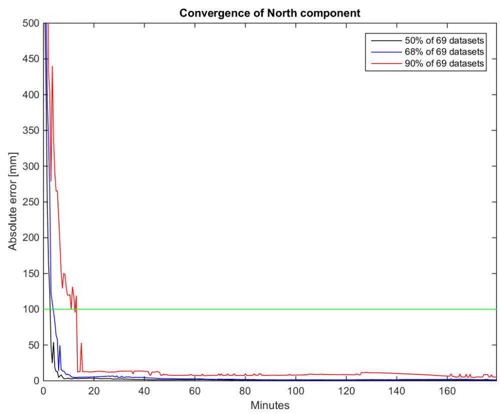
Figure 5.1: Convergence time plots for East component for short baseline PPP-RTK. Lines indicate 50, 68 and 90% of the datasets.

For the North component the convergence plots are shown in figure 5.2. For the 50% and 68% curves there is some improvement, but the effect is minimal. For the 90% curve there is a problem. The float solution shows no problematic datasets in the 90% curve, but after fixing the results degrade. Because the float solution for the North component is already better than for the East component, the datasets that had decreased performance after ambiguity resolution start to shown in the convergence plots. The reasons for the better performance of the North component, for the float solution, and the limited improvement after ambiguity fixing are in the changing satellite geometry:

The GPS satellites are in orbits with a 55 degree inclination. The velocity in North-South direction is therefore higher than the East-West direction. The geometry between the satellites and the receiver therefore changes quicker for the North component. Changing geometry is what is required for the carrier phase ambiguities to be estimated more quickly. For standard PPP and, in this case, the float solution the best performance is seen in the North component. Ambiguity resolution attempts to find the integer value to which the float ambiguity is converging, therefore it mimics speeding up the estimation of the carrier phase ambiguities. The largest benefit of can be found for the component that is weakest in standard PPP. Therefore the largest improvement is seen for the East component.



(a) Convergence for ambiguity float solution.



(b) Convergence for ambiguity fixed solution.

Figure 5.2: Convergence time plots for North component for short baseline PPP-RTK. Lines indicate 50, 68 and 90% of the datasets.

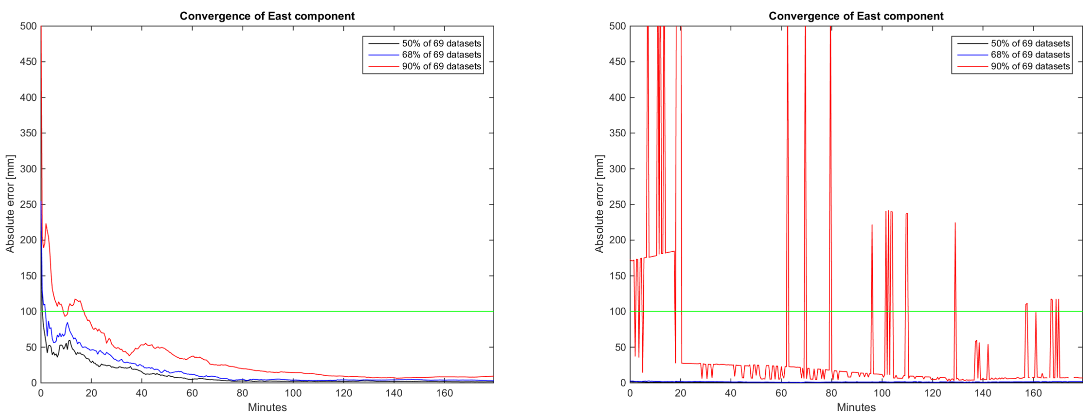
5.1.2. Dual frequency with Ionosphere corrections

When using two frequencies the ionospheric delays can be estimated. Since the single station product from the reference station also provides information on the ionospheric delay estimated at the reference station this information can be applied as well. By using the estimated ionospheric delay from the reference station the model is strengthened. This should improve the precision as the added information on the ionosphere reduces the number of parameters to estimate. For the convergence time the benefit is even greater, as the ionospheric delays are the main reason it takes the solution some time to converge. For the short baseline datasets the results are shown in table 5.2. When all 69 datasets are investigated again the precision deteriorates after ambiguity resolution. The effect is even more predominant than in the case without ionospheric corrections. The cause in this case again are the 8 datasets. These are the only sets in which the standard deviations are larger in the ambiguity fixed case, for all three components. After removing these 8 datasets, there is clear improvement after ambiguity resolution. Also, compared to table 5.1, the ambiguity float solution are more precise using the ionospheric data.

Day	02/09/2014		02/09/2014	
Year - day of year	14-245		14-245	
Stations	69	69	61	61
Ambiguities	Float	Fixed	Float	Fixed
Mean σ_E [mm]	7.3	9.6	6.9	2.7
Mean σ_N [mm]	6.0	7.1	5.6	3.5
Mean σ_{up} [mm]	21.9	35.1	19.8	13.6

Table 5.2: Results for kinematic positioning for dual frequency GPS using external ionosphere information. Average standard deviation for three components for all datasets and a selected subset of the data.

Using static positioning the convergence time is again determined. The convergence plots are shown in figure 5.3. The float solution already shows a very short convergence time for all curves. For the East component, 90% of the datasets reach the 1 decimeter criterion within 20 minutes. The results after fixing the ambiguities show the good and bad datasets. The 50 and 68% curves show that the solution instantaneously converges. With both the precise information on the ionosphere and the ambiguity resolution a very precise estimate of the position can be done immediately. With a distance of less than 10 kilometers between the stations this is expected. The 90% curve shows the bad results from the 8 stations with problematic ambiguity resolution. With the ionospheric delays no longer estimated by the user but externally provided the performance of these stations becomes even worse.



(a) Convergence for ambiguity float solution. (b) Convergence for ambiguity fixed solution.

Figure 5.3: Convergence time plots for East component for short baseline PPP-RTK, using ionospheric delays from the reference station. Lines indicate 50, 68 and 90% of the datasets. The jumps in the fixed solution are seen because with 8 out of 69 datasets having poor performance a single problematic dataset is always within the 90% limit.

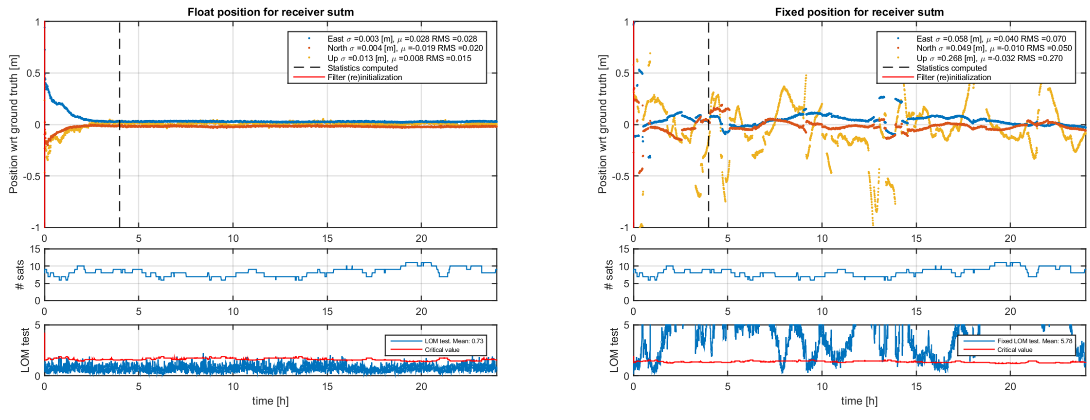
5.1.3. Single frequency with Ionosphere corrections

Previously it was shown that single frequency standard PPP was an option when ionospheric delays were externally provided. Since that is now also the case for short baselines to the reference station in the case of PPP-RTK the single frequency option is tested as well. With two frequency bands available the test can be done for each frequency. The results are summarized in table 5.3. For the L1 frequency the float solution is slightly less stable than the dual frequency counterpart. Fixing the ambiguities to integer values improves the precision for L1. For the L2 frequency the float solution is slightly poorer than the L1, however the performance of ambiguity resolution is again problematic. As before the problem is caused by the same subset of 8 datasets. However, the same datasets show no problems for L1. This indicates that even in the dual frequency case the cause of the poor fixing is linked to the L2 observations.

Day	02/09/2014		02/09/2014	
Year - day of year	14-245		14-245	
Stations	69	69	69	69
Ambiguities	Float	Fixed	Float	Fixed
Band	L1	L1	L2	L2
Mean σ_E [mm]	8.4	4.0	10.8	14.7
Mean σ_N [mm]	8.5	5.8	9.9	13.7
Mean σ_{up} [mm]	27.5	20.3	31.5	55.7

Table 5.3: Results for kinematic positioning for single frequency GPS using external ionosphere information. Average standard deviation for three components for all datasets using either the L1 or L2 band.

For one of the problematic stations the summarized results, for the float and fixed case, are shown in figure 5.4. The solution after fixing the ambiguities is completely corrupted. If the pair of stations shown in the figure is reversed, treating the other station as the user, the problem is still present. The likely cause of the problem is in the receivers and the RINEX 2 format. In the RINEX 2 format signals are described by two characters, for instance *L1*. This is in contrast to RINEX 3, where a third character is added for the method used to determine the observation. Different ways of tracking the signals introduce different biases, but as long as the same mode is used for all satellite this effect cancels by differencing between satellites.



(a) Ambiguity float solution.

(b) Ambiguity fixed solution.

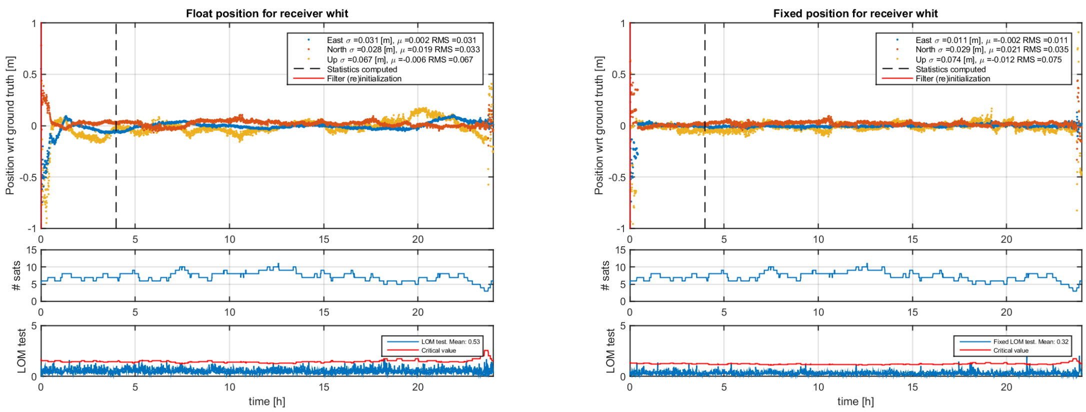
Figure 5.4: Summary plots for the float and fixed solution for station SUTM, using station SUTH as a reference station.

For the newer GPS satellite additional methods are available for tracking the second frequency. In RINEX 3 these different signals are stored as different signals. Upon converting the observation data to RINEX 2 this information might be, erroneously, lost. In this case all observations are treated as the same type. This bias has no effect on the float solution, as the carrier phase ambiguities still converge to a value. The only remaining effect is that after applying the phase biases the ambiguities do no longer converge to integer values. Therefore ambiguity resolution provides integer values that cannot be correct.

One way to prevent this issue receivers must be set to only use a single way of tracking satellites on a single frequency. A better solution is using the RINEX 3 format as these different tracking modes are distinguished in that format. For the user software it is impossible to solve the problem as outlier detection is ineffective. The DIA procedure cannot detect the bias on the L2 in either the network products or the observation data, as the bias is fully absorbed in the ambiguities. Only the integer nature of the ambiguities after applying the phase biases is violated.

5.2. GPS stand-alone: long baseline

Testing the PPP-RTK functionality of the user platform with larger distances between reference station and user is not straightforward. As previously described, over short distances biases are canceled. For this reason testing over larger distances is necessary. To test for biases that do not show for short baselines, the baseline should be as long a possible. However the satellites in view become problematic. With the single station products only the satellites in view for both receivers can be used and as soon as these drop, even for a short period, the solution can become unstable. When selecting a large value for the baseline, say 2500 kilometers, some user provide no results at all. Many provide some, very poor, results. But a few receivers still produce reasonable results. An example of this is shown in figure 5.5, where the solution is fine except for the last hour. Figure 5.6 shows the number of satellites for station WHIT when processed. For PPP all satellites present in the observation file can be used, while for PPP-RTK this number drops. The missing satellites are those that are not in view of the reference station (QUIN).



(a) Ambiguity float solution. (b) Ambiguity fixed solution.

Figure 5.5: Summary plots for the float and fixed solution for station WHIT, using station QUIN as a reference station.

Since results like these are only available for a few stations a baseline of 2500 kilometers is not useful for analysis. At a distance of 500 kilometers more station pairs provide results. This distance is long enough for problems in modeling to show up, while the number of satellites in common view is sufficient for positioning. For the observation data from day 14-245 all pairs of receiver with a distance between 450 and 550 kilometers are formed. This results in a total of 298 pairs. Some pairs have already been excluded due to problems with generating the single station products, which as stated before is only a testing functionality of the user platform.

The problem encountered with fixing ambiguities, because of the RINEX 2 limitations, in the short baseline scenario is also present for the longer baselines. If receivers have settings that introduce biases in the ambiguities, fixing these to integers provides poor results. These sets can be excluded on the basis that the fixed solution has higher empirical standard deviations, for all three components, than the float solution.

In kinematic mode the larger baseline immediately reveals a problem that is hidden in a short baseline scenario. A significant number of datasets for the 500 kilometer baseline have float solutions that are unstable. These solutions show large fluctuations and have very high empirical standard deviations. An example of such a solution is shown in figure 5.7. Fixing the ambiguities in this case has no benefit. These fixed solutions are still as unstable as the float counterparts.

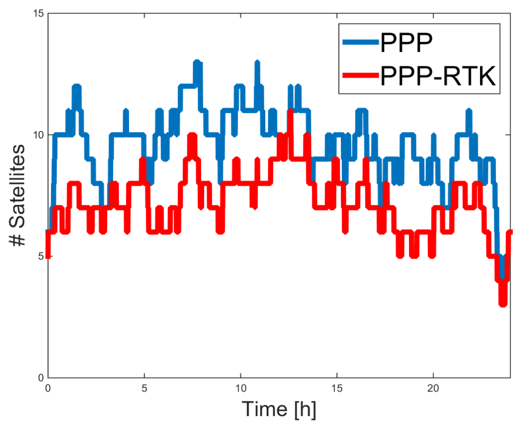
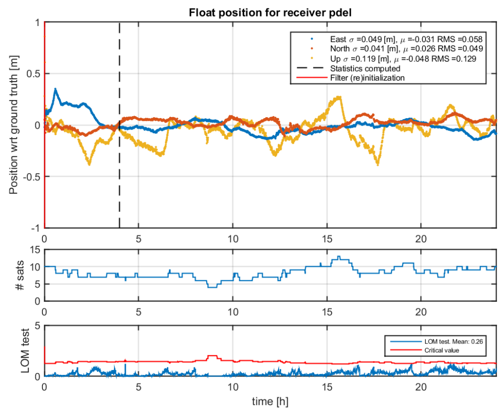
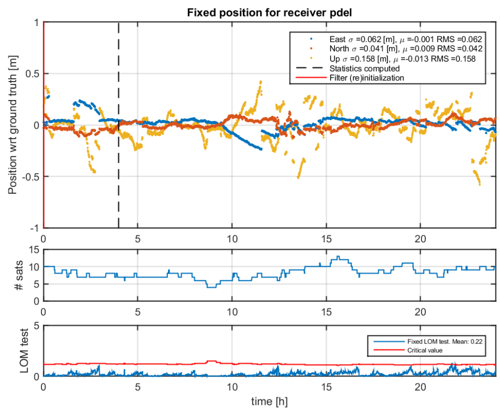


Figure 5.6: Difference in number of satellites used for station WHIT. In blue the satellites used for standard PPP. In red the number of satellites used for PPP-RTK when using a reference station (QUIN) that is 2500 km from the user.



(a) Ambiguity float solution.



(b) Ambiguity fixed solution.

Figure 5.7: Summary plots for the float and fixed solution for station PDEL, using station FLRS as a reference station.

The instability in the float solution is investigated at a later stage. For analyzing the effect of ambiguity resolution over longer baselines these unstable float solutions have to be excluded as well. This can be done on the basis of the empirical standard deviations for the float solution. These are particularly high for the scenarios in which the instability shows up. After excluding the unwanted datasets a total of 112 remain for the 500 kilometer baseline. The same process is repeated for approximate baselines of 200 and 1000 kilometers. The results for the datasets for the ambiguity float and fixed solutions are summarized in table 5.4.

The precision of the results for each of the three components decreases as the distance to the reference station increases. The larger the distance the greater the effect of any remaining biases. The decreased precision for the larger baselines can also be the cause of poorer satellite geometry. Only satellites in view for both receiver can be used, so fewer satellites remain. In the extreme case only satellites in between the user and reference station can be used. The remaining satellites are then all more or less in the same part of the sky, looking from the user side. Whether the deterioration is caused by a lack of satellites or a remaining bias cannot be distinguished from this data.

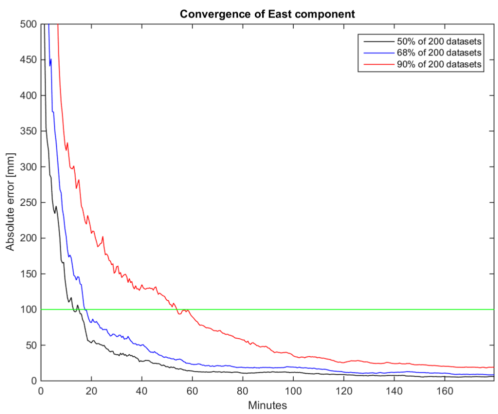
The purpose of these long baselines tests is to get an indication of the performance of the user software when true network products are available (see section 3.4). The performance of the single station products for longer baselines is expected to be worse than the performance when using global products. Fixing the ambiguities to integer values does improve the precision for all three baseline groups. For the longer baselines there is no benefit from canceling out some disturbances that would disappear for a short baseline. Therefore ambiguity resolution is also likely to benefit the positioning estimates when using network PPP-RTK products.

Day	02/09/2014		02/09/2014		02/09/2014	
Year - day of year	14-245		14-245		14-245	
Stations	200	200	112	112	75	75
Ambiguities	Float	Fixed	Float	Fixed	Float	Fixed
Baseline	200	200	500	500	1000	1000
Mean σ_E [mm]	16.5	8.1	22.8	11.5	25.7	13.2
Mean σ_N [mm]	15.1	10.8	21.7	14.9	23.4	17.8
Mean σ_{up} [mm]	42.8	36.2	51.4	47.7	57.4	50.6

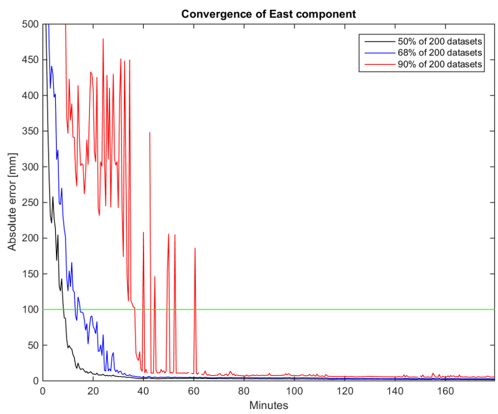
Table 5.4: Results for kinematic positioning for single frequency GPS using external ionosphere information. Average standard deviation for three components for all datasets using either the L1 of L2 band.

Potentially the benefit is even greater than for the larger baselines tested here, as using network products all satellite in view of the user station can be used.

For these longer baselines the convergence time is also investigated. This is done in static mode. The datasets that have been excluded for kinematic positioning are also skipped for the static case. The instability in the float solution is not as visible for static positioning, but the problem causing the instability is likely to affect the static solution as well. The convergence time of the East component, for the datasets with an approximate distance of 200 kilometers between user and reference, is shown in figure 5.8. Fixing the ambiguities to integer values has benefits for these datasets. For the best performing 50% of the datasets the convergence time drops to under 10 minutes. A small improvement is visible for the 68% curve as well, improving the convergence time by a few minutes. The 90% curve shows that some datasets have problems with ambiguity resolution.



(a) Ambiguity float solution.

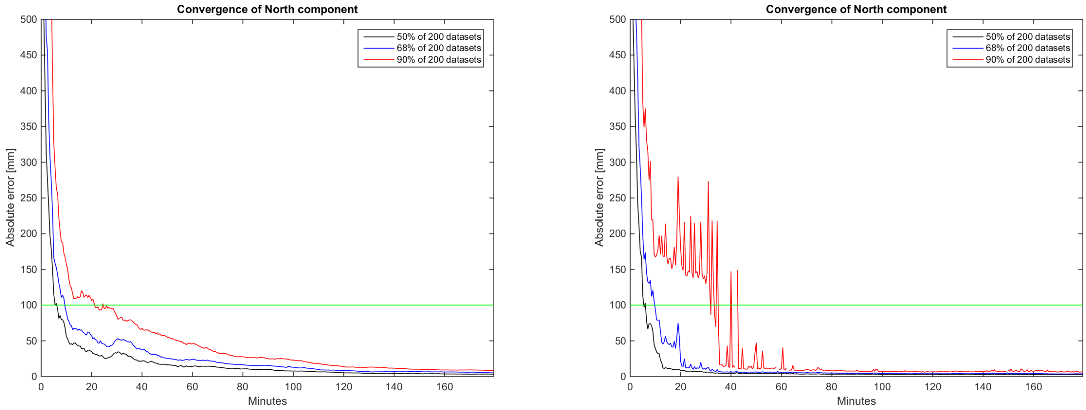


(b) Ambiguity fixed solution.

Figure 5.8: Convergence plots for the East component of PPP-RTK solutions with baselines of approximately 200 kilometers.

The convergence of the North component for the 200 kilometers baseline is shown in figure 5.9. Here the 90% curve indicates the presence of problematic datasets. The other two curves indicate the same convergence time for float and fixed solutions. The convergence for the North component is already quick, leaving little room for ambiguity resolution to improve the results for this component. This same effect is seen for the North component for larger baselines, which is why only the East component is shown after this.

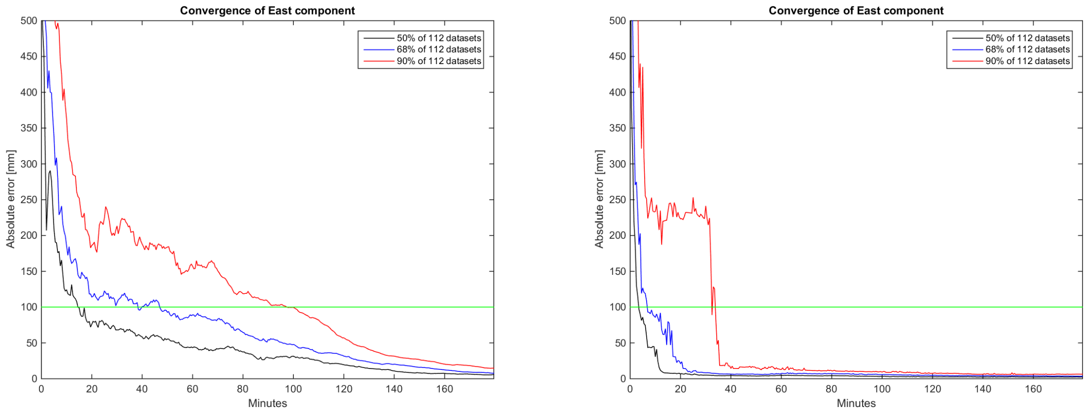
The convergence plots for the East component, using stations with distance around 500 kilometers, are shown in figure 5.10. Before ambiguity resolution the convergence time for the 50 and 68% curves is already higher than for the 200 kilometers case. Fixing the ambiguities benefits the convergence time. Even for the 68% curve the convergence time is less than 10 minutes. The 90% curve also improves from over 90 minutes to less than 40. The 90% curve is more stable than in the 200 kilometer baselines case. This indicates that not a lot of problematic datasets remain. The reason for this is that for longer distances the problematic behavior is easier to identify. If the previously shown problem with an unstable float solution is present in a dataset its



(a) Ambiguity float solution. (b) Ambiguity fixed solution.

Figure 5.9: Convergence plots for the North component of PPP-RTK solutions with baselines of approximately 200 kilometers.

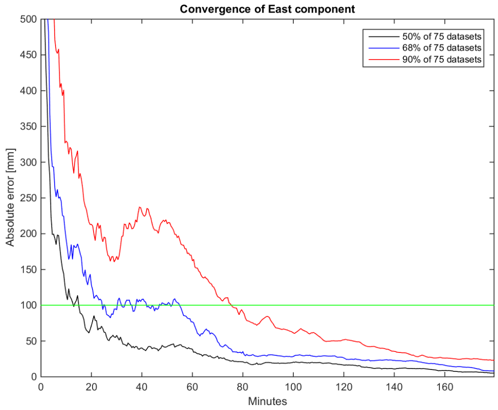
effects are large for the 500 and 1000 kilometer baselines, but still small in the 200 kilometer case. Fixing the ambiguities for these cases does lead to problems, resulting in the 90% curve as shown in figure 5.8b.



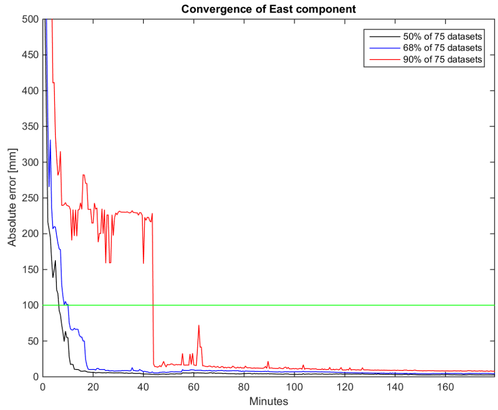
(a) Ambiguity float solution. (b) Ambiguity fixed solution.

Figure 5.10: Convergence plots for the East component of PPP-RTK solutions with baselines of approximately 500 kilometers.

For the 1000 kilometer baseline the convergence time plots for the East component are shown in figure 5.11. The same improvement is still visible. For all three curves the convergence time is approximately halved. For 68% of the datasets for the 500 and 1000 kilometer baselines the convergence time for the East component is close 10 minutes. The float solution in these cases shown convergence times between 20 and 40 minutes. Even over longer baselines the benefit of ambiguity resolution is visible. When the single station products are replaced with products from a network, the same performance in convergence time as seen for the longer baselines should be achieved. As more satellites are available for users, when the products include all satellites, these results might even improve further.



(a) Ambiguity float solution.



(b) Ambiguity fixed solution.

Figure 5.11: Convergence plots for the East component of PPP-RTK solutions with baselines of approximately 1000 kilometers.

5.3. Multi-GNSS

Using more constellations for PPP-RTK is possible with the user platform. The model used for PPP-RTK can be applied to all constellation using CDMA, leaving out only Glonass. Analysis of the PPP-RTK results using the more constellations is limited to combinations of GPS, BeiDou and Galileo. Observation data from QZSS can be processed by the user platform, but the single satellite does not benefit the solution. Similar to standard PPP it is possible to treat the QZSS satellite as being part of the GPS constellation. Doing this for PPP-RTK puts an additional requirement on the network corrections. Either the receiver used to generate the network products must be the same as the user receiver, or calibrated ISBs must be provided. The single station products used for these tests are too limited for this.

Both BeiDou and Galileo provide sufficient satellites to affect the PPP-RTK solution. The MGEX datasets used for multi-GNSS PPP are also used in this case. All 101 stations can be used to generate single station products, but the number of combinations that can be made is more limited than for the GPS stand-alone case. There are more limiting factors. Not all receivers in the MGEX network track the same constellations. After removing the station pairs that give unstable float solution not a lot of datasets remain.

The effect on the convergence time when more satellites are added is the main focus of this analysis. For BeiDou the number of satellites available depends on the geographical location. When analyzing multiple datasets the location of the receivers has a very large effect on the solution. Comparing between receiver pairs on different continents is therefore not very useful. Since the convergence time is only influenced by the number of satellites present during the start of processing Galileo also limits the analysis. A receiver that track a few Galileo satellites at the end of the day, but none at the start, cannot benefit from the added satellites.

To overcome this restriction on the datasets only a specific set of stations is used. MGEX station MRO1 in Western Australia is approximately 600 kilometers from Perth. This allows five of the receivers at Curtin University to be used, forming several pairs with 600 kilometer baselines. Station MRO1 tracks GPS, Galileo and BeiDou, as do the selected five receivers in Perth (CUT0, CUT2, CUTA, CUTB and CUTC).

The satellites available for these stations are shown in figure 5.12. The intervals are selected for these stations, based on the number of satellites available. Interval A start after roughly two hours, when a total of three Galileo satellites are available and ends before the fourth satellite rises. Interval B is centered around the single hour in which four Galileo satellites are available. Interval C has three Galileo satellites available, but the number of GPS satellites available at that time is lower compared to interval A.

The datasets are set up to be initialized during these intervals. In each case only four hours of data are processed, which is sufficient to determine the time needed for the solution to converge. The same observation data is used multiple times, with five minute delay between the different runs. This way full use is made of

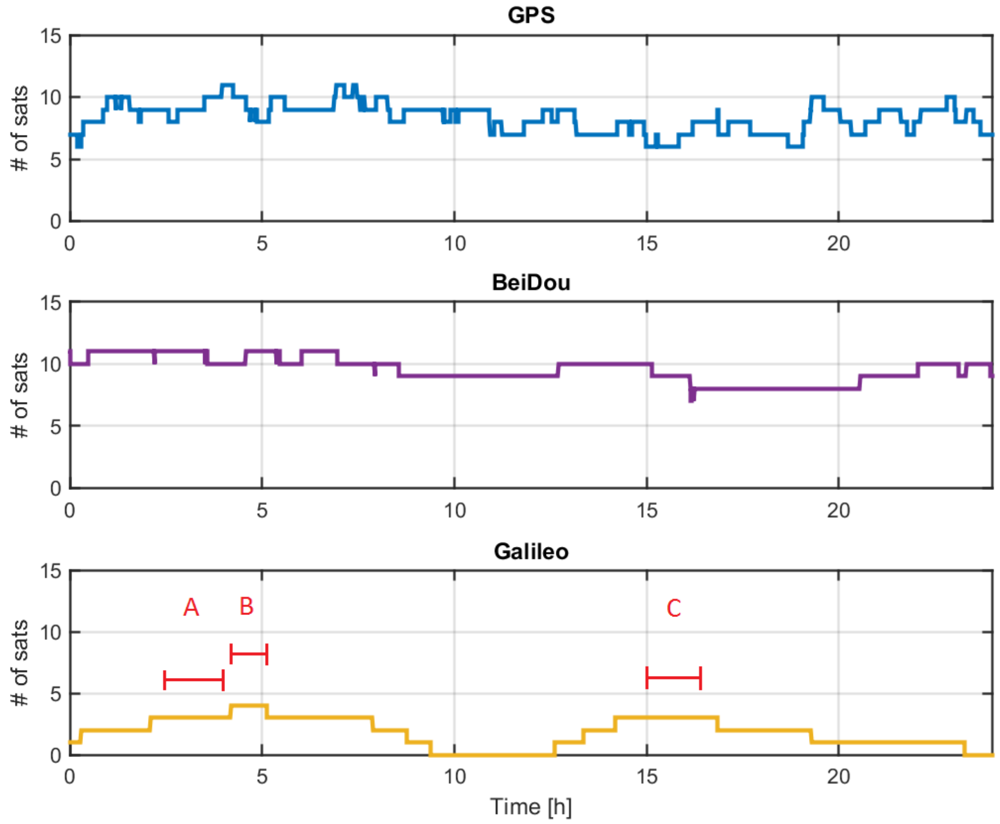


Figure 5.12: Satellites available for MGEX station MRO1 for day 15-168. In red three intervals (A,B and C) are shown with sufficient Galileo satellites.

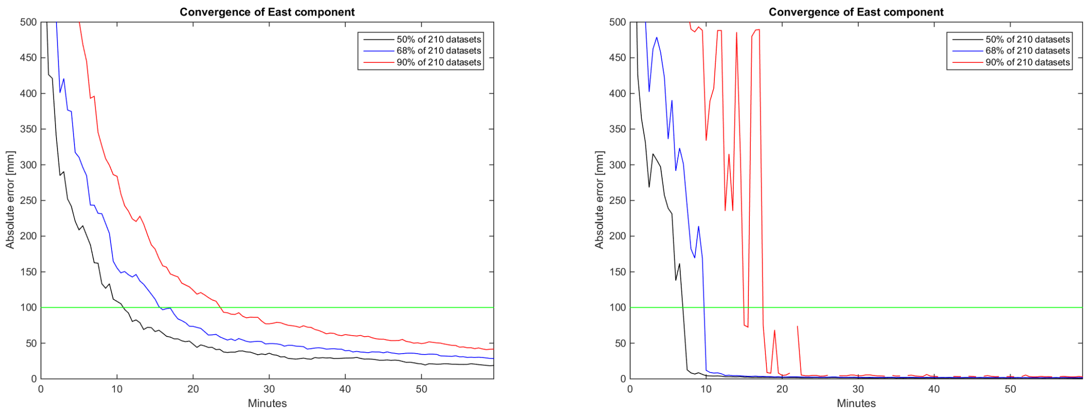
the available observation data as the hour of data with four Galileo satellites is processed several times. The selected intervals also have sufficient BeiDou satellites available.

5.3.1. Interval A: 3 Galileo Satellites

The first selected interval starts at epoch 260, which corresponds to 2:20 UTC. The interval ends at epoch 460, which is 3:50 UTC. For every 5 minutes in this interval a steering file configured to start. Each steering file lets the user platform process four hours of observation data. For interval A there are a total of 21 start points selected this way. The interval depicted in figure 5.12 is slightly larger as no steering files are configured to start in the last 20 minutes of the depicted interval. This way the number of available Galileo satellites is always three for all the tests during the convergence period.

A total of 10 station pairs are formed. Five stations in Perth act as reference stations with MRO1 as the user. The other way around the single station products generated from MRO1 are also applied to the 5 Perth stations. With 21 starting points there are a total of 210 result sets for interval A. These are processed using stand-alone GPS as a benchmark. This is compared to the combination of GPS and BeiDou, the combination of GPS and Galileo and the triple system combination.

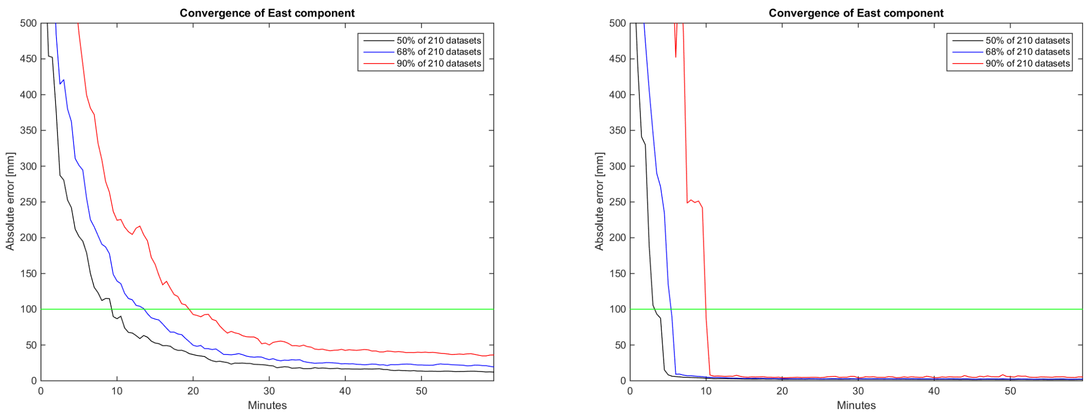
Figure 5.13 shows the convergence time of the East component when only GPS is used. The convergence time for the float solution is already short, during this interval between 8 and 9 GPS satellites are available. Therefore the convergence time for the best 50% of the results only slightly above 10 minutes. For all three depicted limits there is improvement when the ambiguities are fixed to integer values. For the 90% limits the results there are some problems with fixing visible, but the time to reach the 1 decimeter limit is better than the float counterpart.



(a) Convergence time for float solution.

(b) Convergence time for fixed solution.

Figure 5.13: Convergence time plot for the East component for interval A, using only the GPS constellation.



(a) Convergence time for float solution.

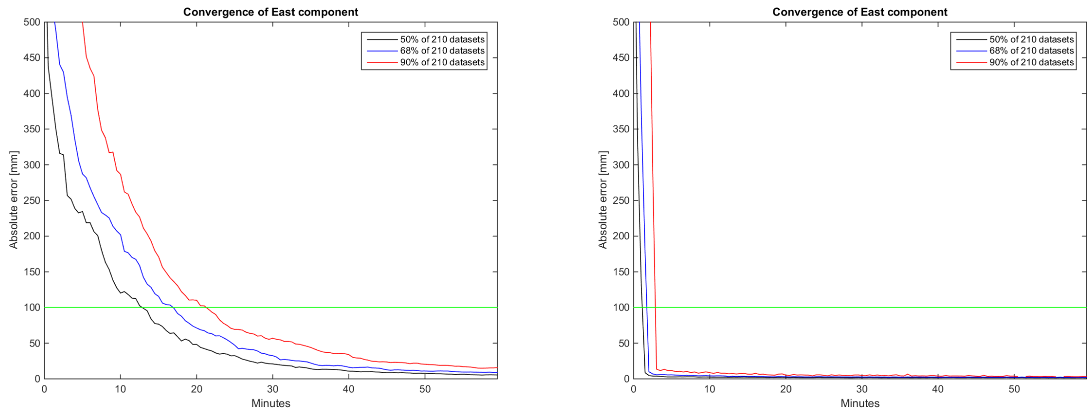
(b) Convergence time for fixed solution.

Figure 5.14: Convergence time plot for the East component for interval A, using the GPS and Galileo constellations.

The convergence plots when Galileo is added are shown in figure 5.14. With only three satellites for Galileo during the start of these datasets the effect on the float solution is minimal. The 68 and 90% limits improve by a few minutes. The fixed solution shows more improvement with respect to the GPS standalone counterpart. 68% of the datasets show a converged solution at 5 minutes. The performance increase after ambiguity fixing is cumulative: the float solution already improves with the Galileo satellites and the fixing improves the performance on top of this. Even the worst datasets converge in 10 minutes, compared to close to 20 for the GPS standalone datasets. With only three Galileo satellites aiding in the convergence of the solution this proves quite beneficial.

BeiDou has between 10 and 11 satellites available during the interval. The geographical location of the receivers used allows for all five geostationary satellites to be used for positioning. The geostationary satellites are used for the float solution, but the ambiguities for these satellites are not fixed by the software. This option is hard-coded in the user platform. The results for the combination of GPS and BeiDou are shown in figure 5.15. For the float solution the change in performance is similar to what was seen for standard PPP. The best 50% shows a convergence time that is poorer than the GPS stand-alone case. The 90% limit improves when BeiDou is added to the solution. Datasets with poorer performance benefit from additional satellites, while for the better datasets the convergence time is adversely affected by the geostationary satellites.

For the fixed solution the difference is clearly visible. With 10 or 11 BeiDou satellites, 5 or 6 satellites have ambiguities that are fixed to integer. For Galileo three satellites already made a difference. The 6 extra ones provided by BeiDou have an even larger impact. Even for the 90% limit the convergence time is below 4 minutes for this scenario.

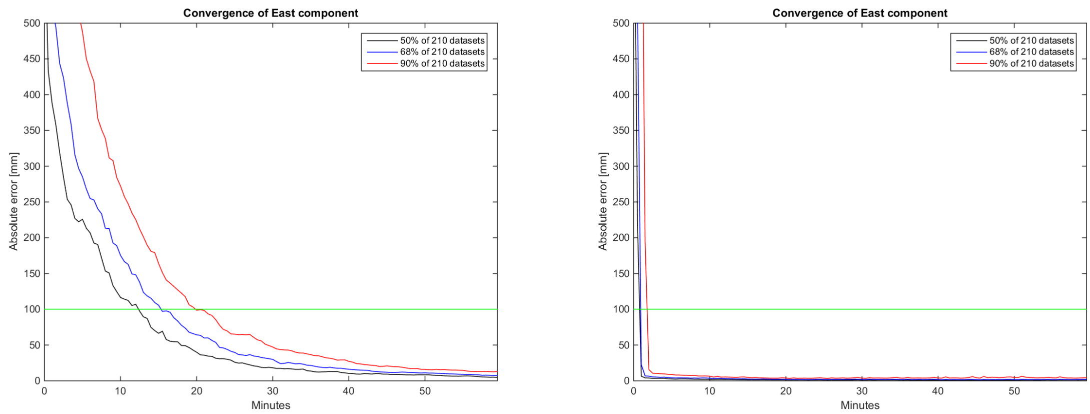


(a) Convergence time for float solution.

(b) Convergence time for fixed solution.

Figure 5.15: Convergence time plot for the East component for interval A, using the GPS and BeiDou constellations.

Even though there is little room for improvement the three constellations can also be combined. The results for this are shown in figure 5.16. A small improvement for the ambiguity fixed solution is still visible. The 50% limit shift to a single minute. Precise determination of the convergence time is not possible as the observation data only provides information every 30 seconds. Looking a the performance increase when the incomplete constellations are added, the improvement in the future should lead to convergence times given in seconds, not minutes. The interval selected is however one with particularly good satellite reception.



(a) Convergence time for float solution.

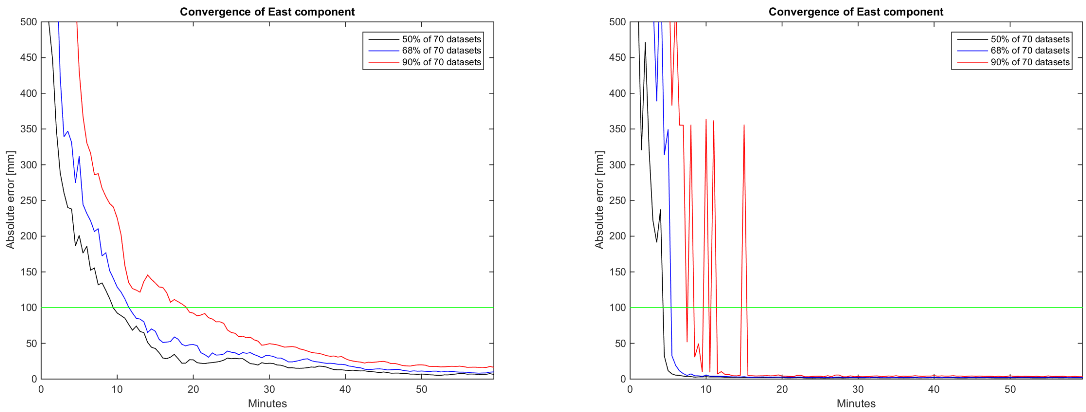
(b) Convergence time for fixed solution.

Figure 5.16: Convergence time plot for the East component for interval A, using the GPS, BeiDou and Galileo constellations.

5.3.2. Interval B: 4 Galileo Satellites

The second interval is shorter than the first. During almost an hour four Galileo satellites are in view for all receivers. The interval starts at epoch 510 which corresponds to 4:15 UTC. The last datasets are started at epoch 570, which is 4:45 UTC. This leaves a total of 70 datasets, with 10 station pairs.

Figure 5.17 shows the convergence plots for the GPS standalone scenario. The performance for both float and fixed solution is better for this interval, compared to the first one (figure 5.13). The improved performance is caused by an additional GPS satellite that is visible. During the convergence 10 GPS satellite are visible for all stations.

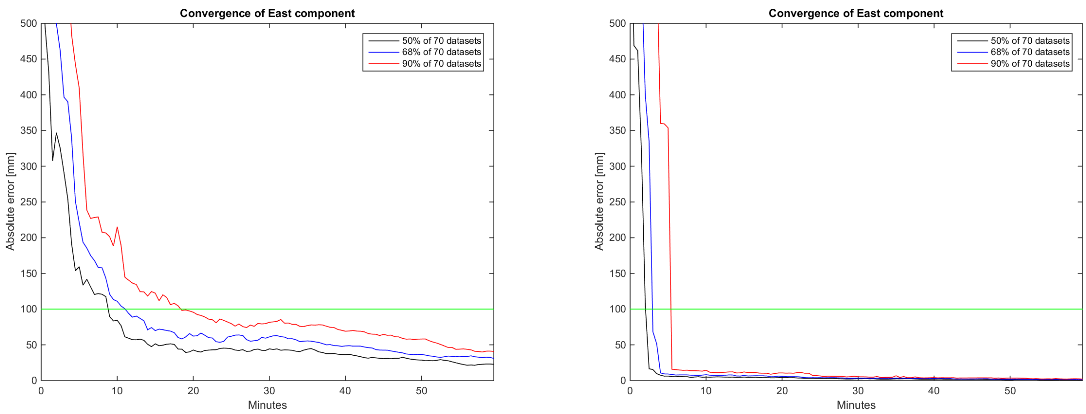


(a) Convergence time for float solution.

(b) Convergence time for fixed solution.

Figure 5.17: Convergence time plot for the East component for interval B, using only the GPS constellation.

The convergence plots after adding the four Galileo satellites are shown in figure 5.18. The float solutions improve slightly. The fixed solution loses the jumpy behavior from wrong fixes visible in the GPS stand-alone case. The effect on all three curves is very clear. With only four Galileo satellites the convergence time for 90% of the datasets is already pushed down to 5 minutes. When the Galileo constellation is completed, using the additional satellites is even more beneficial for PPP-RTK. For interval B BeiDou is not used as the number satellites is similar to the scenario for the first interval.



(a) Convergence time for float solution.

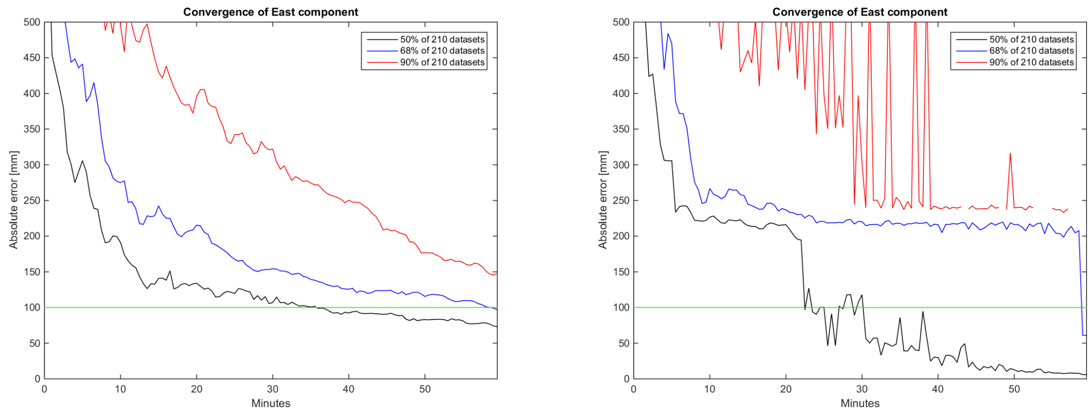
(b) Convergence time for fixed solution.

Figure 5.18: Convergence time plot for the East component for interval B, using the GPS and Galileo constellations.

5.3.3. Interval C: 6 GPS Satellites

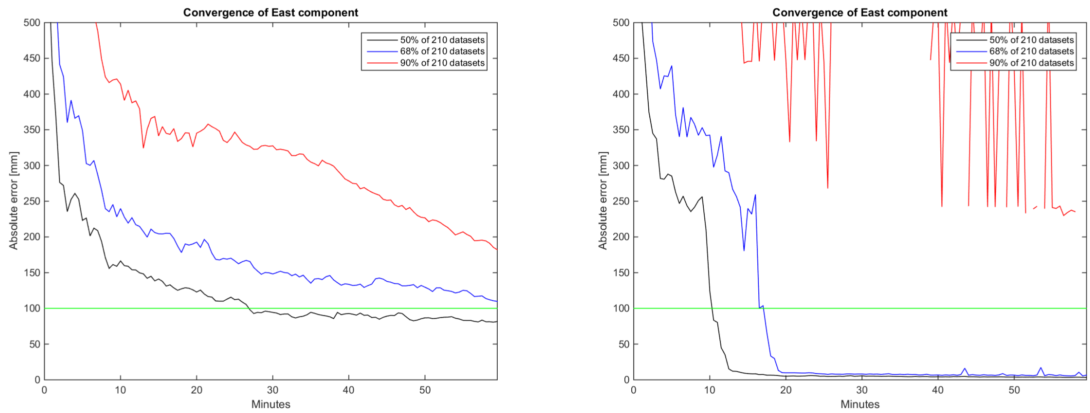
The third interval is later on the day. During interval C there are still three Galileo satellites visible, but the number of GPS satellites is lower than was the case for the first interval (see figure 5.12. The first datasets start at epoch 1800, which corresponds to 15:00 UTC. The last datasets start at epoch 2000, which is 16:40 UTC. The number of datasets is, similar to interval A, 210.

Figure 5.19 shows the convergence plots for the GPS stand-alone scenario. The performance is very poor. For the 50% limit the datasets converge after more than 30 minutes for the float solution. Ambiguity fixing improves the convergence time, but still this takes more than 25 minutes. For the 68% both float and fixed solutions take almost an hour to reach the 1 decimeter criterion. Having only 6 satellites available is not problematic once the solution has converged. Before convergence being so close to the minimum number of satellites is problematic.



(a) Convergence time for float solution. (b) Convergence time for fixed solution.

Figure 5.19: Convergence time plot for the East component for interval C, using only the GPS constellation.



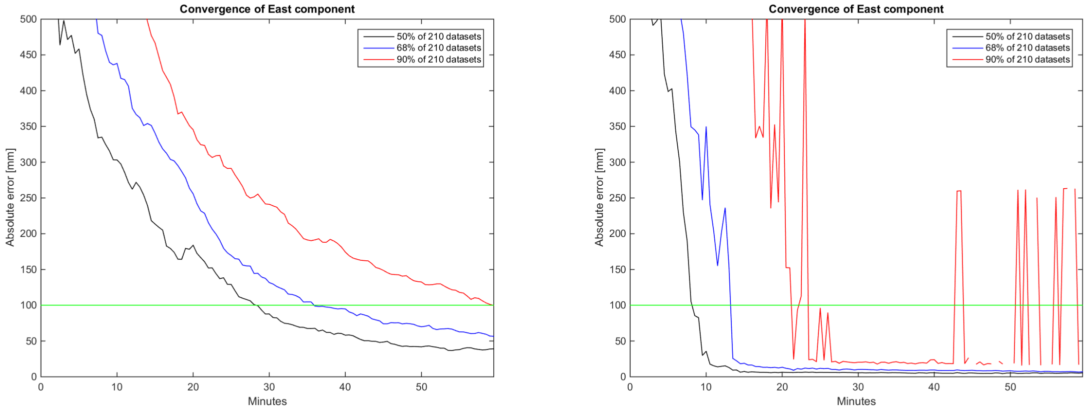
(a) Convergence time for float solution. (b) Convergence time for fixed solution.

Figure 5.20: Convergence time plot for the East component for interval C, using the GPS and Galileo constellations.

Adding the three available Galileo satellites to the solution results in figure 5.20. This provides no benefit to the float solution. The fixed solution improves for the 50 an 68% limits. The convergence time for these datasets becomes more acceptable to below 20 minutes. The problematic datasets in the 90% limit still remain problematic. These three satellites cannot provide enough support to resolve the problems for these datasets.

Figure 5.21 shows the convergence time when GPS and BeiDou are combined. The float solution already improves with the additional satellites, compared to GPS stand-alone. The lack of GPS satellites allows BeiDou to improve the solution. The improvement due to additional satellites is larger than the detrimental effect of the geostationary satellites. For the fixed solution there is improvement, but this is on the same level as the improvement for Galileo. For this interval there a fewer BeiDou satellites, only between 7 and 9. Of these 5 are always the geostationary satellites, leaving fewer ambiguities to be fixed.

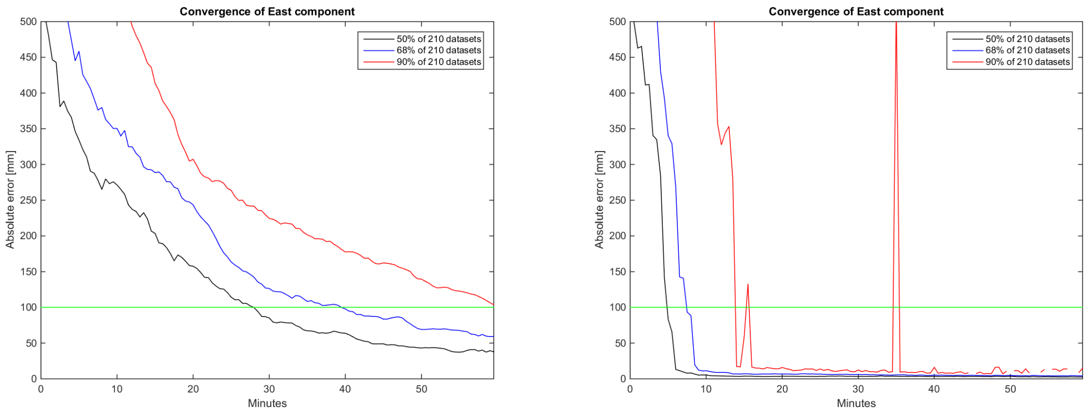
Figure 5.22 shows the full combination of the three constellations. This does indeed provide the best performance. Up to the 68% limit the datasets converge within 10 minutes. Even now not all problems in the datasets are resolved, as shown by the sudden jump in the 90% curve at 35 minutes. The performance for the best datasets is also not even close to the performance in interval A. This shows that the number of satellites is always the dominant factor. However, combining the constellations provides significant improvement for PPP-RTK. When more satellites for Galileo and BeiDou become available the performance for the best interval can be matched with a receiver in any geographical location. For now the improvement is only possible at certain times of the day (Galileo) or in certain parts of the world (BeiDou).



(a) Convergence time for float solution.

(b) Convergence time for fixed solution.

Figure 5.21: Convergence time plot for the East component for interval C, using the GPS and BeiDou constellations.



(a) Convergence time for float solution.

(b) Convergence time for fixed solution.

Figure 5.22: Convergence time plot for the East component for interval C, using the GPS, Galileo and BeiDou constellations.

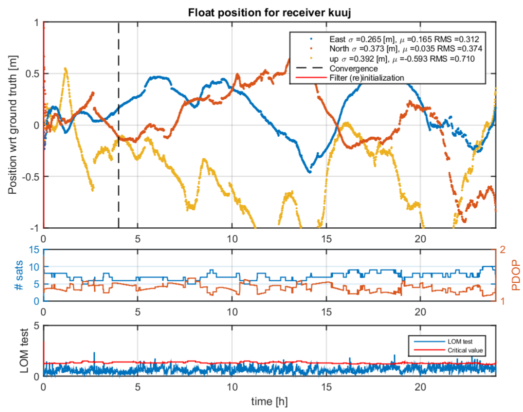
5.4. Float instability

The recurring problem of unstable float solutions is further addressed in this section. While the problem has not been solved, some further investigation is described. For different tests done with PPP-RTK this problems rises for some datasets. The problem is not linked to the RINEX version of the observation files. The GPS stand-alone tests are mostly done with RINEX 2, while the multi-GNSS tests are done with RINEX 3 observation files. In both cases the unstable float solutions show up in the results.

The first consideration is that the problem with the float solution does not depend on the user station's observation data. Figure 5.23 provides two examples for the same user station (KUUJ). The first example is the results when station THU2 is used as reference. This station is 2384 kilometers from the user, but still the float solution is stable. When the same user station is selected, but now the single station products from station GODE are used (baseline is 1803 kilometers). The solution is completely unstable. Similar results can be achieved with any user station when the GODE single station products are used. Only when a user station is close to the reference station the results remain stable.

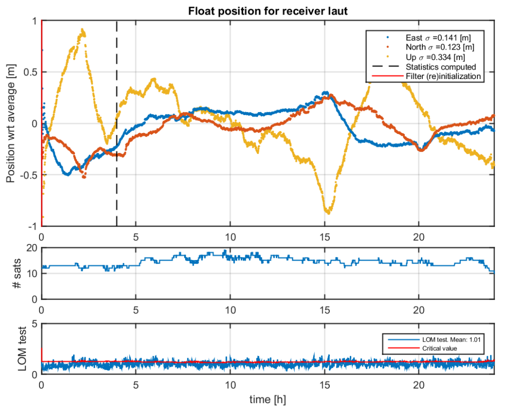
Figure 5.24 shows the results for station FTNA and LAUT on day 15-168. The results for both station are when the other station is used as a reference station. The distance between the two stations is 599 kilometers. The problem is clearly visible in the results of station LAUT, while the results of station FTNA are stable. The problem is therefore not caused by any kind of mismatch between two datasets. If such a case the problem

(a) Stable results with station THU2 as reference (2384 km).

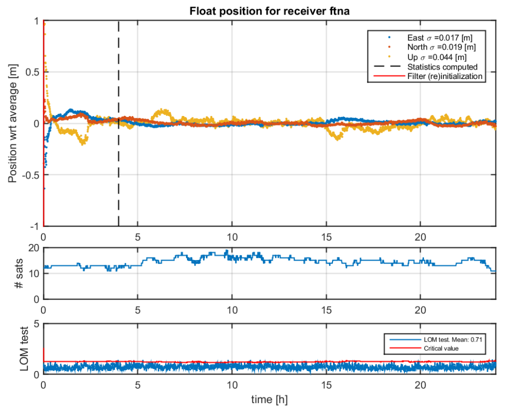


(b) Unstable results with station GODE as reference (1803 km).

Figure 5.23: Summary plots for user station KUUJ using two different reference stations to generate single station products.



(a) Unstable results for user LAUT with reference station FTNA (599 km).



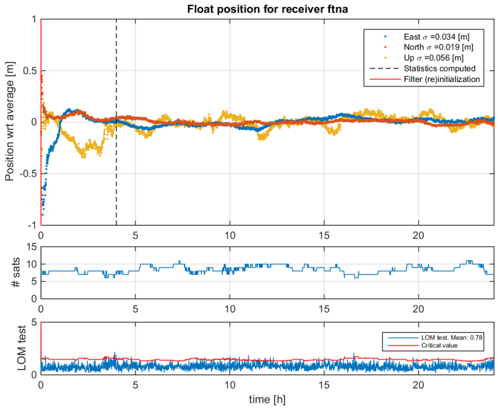
(b) Unstable results for user FTNA with reference station LAUT (599 km).

Figure 5.24: Summary plots for stations FTNA and LAUT, used once as reference and once to generate single station products.

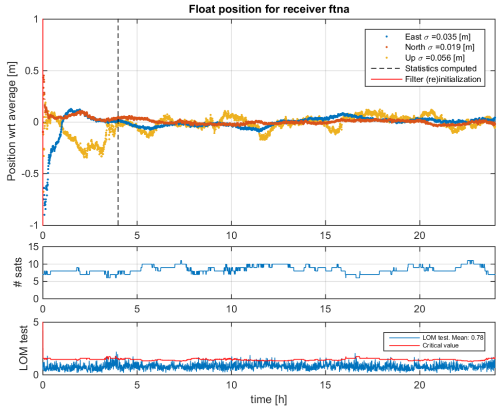
would exist in both results. In this case the problem is caused by the observation data of the reference station alone. The problem is also not caused by any specific constellation. For these results GPS, Galileo and BeiDou are used for positioning. All other combinations of the constellations also give unstable float solutions.

For generating the single station products only the orbit products and the observation data are used. One addition to the steering file for the user software to process is the fixed receiver position. As this is information that is not directly taken from the provided data, there is a risk that it causes the instability problem. For most tests the fixed location is determined by processing an observation in static mode and taking the average position of the last hour. As was shown when comparing standard PPP results to an external ground truth there can be a small offset (see section 4.1.2).

To test the effect of such an offset the datasets for stations FTNA and LAUT are used. As shown in figure 5.24b using station LAUT as reference provides results without the instability. The single station products for station LAUT are generated again, but after introducing an offset in its fixed position. First a shift of 1 centimeter in both North and East components is made. Section 4.1.2 indicates that an offset of 1 centimeter occurs often for datasets. For the second test this offset is pushed to 1 decimeter. If the position bias is causing the instability this should show clearly with a decimeter shift.



(a) Results when the position of the reference station has 1 cm bias in both East and North components.

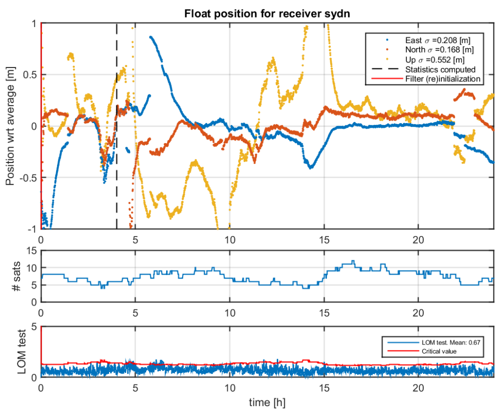


(b) Results when the position of the reference station has 1 dm bias in both East and North components.

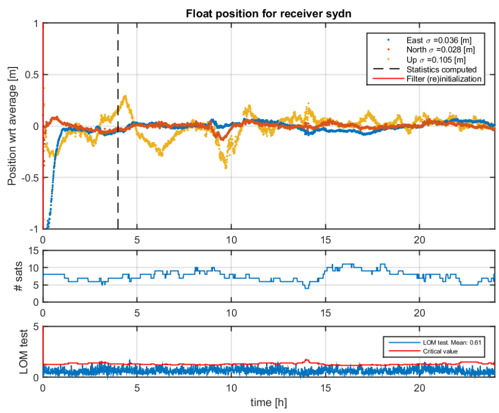
Figure 5.25: Summary plots station FTNA after a bias is applied to the position of reference station LAUT.

The results for the user station after applying the position shift for the reference station are shown in figure 5.25. The difference is not visible in the plots. The statistics show that the standard deviation for the East component increased by 1 millimeter for the case where the reference station is 1 decimeter shifted. An inaccurate position of the reference station does not cause the float instability. It does affect the solution for the user station, since the estimated position for the user contains the same bias as was introduced in the reference. So for accurate positioning on the user side an accurate position for the reference station is still a requirement.

The last potential problem to investigate is a dependency on receiver, antenna and the geographical location of the reference station. The three stations in the Eastern part of Australia that have been used for testing PPP over multiple days are also close enough for PPP-RTK. Stations MOBS (Melbourne) and SYDN (Sydney) are furthest apart. For every day an estimated position for the reference station is determined and afterwards single station product are generated.



(a) Day 14-100 (April 10th 2014).



(b) Day 14-101 (April 11th 2014).

Figure 5.26: Summary plots station SYDN using station MOBS as a reference station.

Figure 5.26 shows the results for two consecutive days processed using SYDN as the user and MOBS for generating the single station products. On day 100 the instability is clearly present, while on day 101 the result is stable. Since the same receiver and antenna are used on both days and there is no change in location, these three factors are not the cause of the float instability.

A cause for the instability has not been found by these tests, but some potential causes can already be excluded. The problems appears to rise on the side of generating the single station products. As this part of the user platform is basic functionality investigation of the cause is not continued further. Because the problem occurs for some reference stations, while other are not affected, it is likely that the source of the problem is easier to find when products are generated from a network instead of a single station. At that stage a single dataset creating the problem can be directly compared to those that show stable behavior.

6 Conclusion

The first goal has been to test the functionality of the user platform: *What different types of observation and correction files can Curtin's PPP-RTK user platform successfully process?* The best way to test for any bugs present within the software is to process as many datasets as possible. In total more than 60,000 datasets have been successfully processed during these tests. In order to have as many different scenarios as possible many combinations have been made. This includes processing observation data stored in RINEX 2 and 3 format, including many sub-versions of those formats. Different quality orbit and clock products have been used for testing as well. Throughout these tests a lot of problems occurred. Some bugs only showed in a single epoch, for one dataset out of 400 sets. Since these bugs have been identified and fixed, these have not been part of the main report, but have instead been partially described in the appendix. The user platform is now fully capable of processing observation data from RINEX observation files and orbit and clock products from SP3-files. No matter which set of constellations is chosen a result is produced. This does not guarantee a solution, as selecting a constellation that has insufficient satellites for positioning does not yield results for any part of the day. The software finishes processing in such a case, but simply produces no estimated position.

Not all problems have been resolved. When generating single station products for PPP-RTK using the user platform issues can still arise. This functionality has however only been included to be able to test PPP-RTK as a user. Problems in generating the products have been largely ignored as this functionality is not to be used for later research. Other software is expected to provide the corrections for PPP-RTK, either generated from a single station or from a network of receivers.

The second goal has been testing the performance of the software: *What precision, accuracy and convergence time can be achieved under different conditions using the positioning techniques available in the user platform?* Concerning accuracy the software produces results, using standard PPP, that are compared to externally provided ground truths. For all three components the results show a distribution around zero. This indicates that there is no large position bias in the solution for PPP using only GPS. Not every dataset provides a solution that exactly matches the external ground truth. The large number of test sets show a deviation of up to a centimeter for the East and North components and an offset of up to three centimeters for the weaker up component. This comparison to ground truth was not done for testing with more constellations, as these constellations suffer from a lack of information on antenna calibration. This causes a bias when comparing to a ground truth. Without additional calibration data for antennas this cannot be resolved.

The precision is determined per dataset. A full day of data is processed and all results after four hours are used to determine the empirical standard deviation of the position estimates. The first four hours are disregarded to ignore the impact of a dataset for which the solution takes a long time to converge.The precision of the results is hard to compare to literature. The achievable precision can be given for a 'normal scenario', but the definition of such a scenario is vague. The results depend heavily on the number of available satellites and the receiver-satellite geometry. For GPS kinematic PPP the horizontal components are well below a 30 millimeter standard deviation for the majority of the datasets. The up component is weaker but the majority of the datasets have a standard deviation below 60 millimeters. The histograms for these statistics show that better results are possible. The best datasets have horizontal standard deviations around 5 millimeters and up component below 20. These results only hold when the best orbit and clock products are used. Using poorer quality products reduces the precision of the solution.

The convergence time is determined only for batches of datasets. For every dataset the empirical mean is computed for the end of the observation data and used as the ground truth. For the start of the dataset the solution for each epoch is compared to this ground truth. For each epoch the datasets are sorted and three subsets are selected. The limits under which the best 50, 68 and 90% solutions fall are then plotted. The solution is assumed to have converged once these curves drop under a 10 centimeter offset from the computed mean. For most scenarios using only GPS the convergence time is less than 30 minutes. The convergence time can be significantly different for individual datasets as it depends heavily on the receiver-satellite geometry during the initial epochs.

The third goal was focused on multi-GNSS: *How does the performance of PPP and PPP-RTK change if the newer GNSS constellations are used and what problems still exist when processing this multi-GNSS data?* By expanding PPP tests to the newer constellations improvements can be found for the precision. First the results for GPS are reduced in quality because orbit and clock products are used that are also suitable for the newer constellations. The poorer results for GPS are then compared to the multi-GNSS results, taking into account that in the future the orbit and clock products will improve for all constellation. The Chinese BeiDou can on its own not match the precision of GPS. Even though for some regions more BeiDou satellites than GPS satellites are available, the precision is significantly poorer. When combining BeiDou and GPS some improvement can be found, especially for datasets with limited reception of GPS satellites. However the segment of BeiDou that consists of geostationary satellites appears not to benefit the solution in all cases. When more BeiDou satellites are available, stand-alone positioning without the geostationary satellites can be used to verify this. Currently using the geostationary satellites still benefits some scenarios as the mere presence of additional satellites overshadows the lack in performance of those geostationary satellites.

For Galileo the improvement is minimal for the PPP solution. This is due to the very limited number of available satellites. The presence of a small improvement or in other cases at least no decrease in precision, indicates that with more available satellites the use of this constellation is beneficial. This does depend on the quality of the orbit and clock products. Two sets have been tested, one with poor quality orbits. These decreased the precision, but with more ground stations tracking the Galileo satellites these products should only improve. QZSS has been tested, but the single satellite cannot make a large impact. Treating the single QZSS satellite as part of the GPS constellation does affect the solution, but the performance is poor. This is likely caused again by the quality of the orbit and clock corrections available for this single satellite.

The convergence time of the solution for PPP is an important parameter. For the case of a static solution with only GPS 50% of the datasets reaches the 1 decimeter criterion in 20 minutes. 68% of the datasets reaches the convergence criterion within 40 minutes. However the poorest datasets can take over an hour. This again shows a problem with the definition of a 'normal scenario' for which an expected convergence time is given. The convergence time depends heavily on the number of satellites and their geometry during the start of processing.

Additional constellations can be added to try and improve the convergence time. When adding BeiDou to GPS hardly any improvement is found. Only the datasets performing poorly with only GPS show a slight improvement in convergence time when BeiDou is added. Datasets with poor GPS reception can benefit from the BeiDou constellation, but the convergence time of datasets with sufficient GPS satellites does not benefit. For Galileo hardly any difference is found. With the limited number of satellites, the number of stations that have a significant number of Galileo satellites in view during start-up is very low. Only the convergence time of these station is affected, which does not show when computing the convergence time from a large number of datasets.

When testing the PPP-RTK performance of the software some limitations apply. In the future the user platform can be used to test positioning using PPP-RTK products computed from a network of stations. While the format of the input given to the user software is the same in that case, currently these corrections can only be generated from a single reference station. Any testing is done keeping in mind that in the future these products will be slightly different. For all PPP-RTK tests the solution is given for two cases. Before fixing the carrier phase ambiguities to integer (float solution) and after fixing (fixed solution).

If the reference station is less than 10 kilometers from the user ionospheric delays can also be used as input for the user station. In this case the results for the float solution are already good, with 90% of the float solutions converging in under 20 minutes. Because of the strength of the model, correctly fixing the ambiguities can be done immediately. In this case the convergence time is zero. Over longer distances the convergence time is worse. When the distance between the user and reference station is around 1000 thousand kilometers 50% of the datasets take 20 minutes to converge. 68% take just under an hour. Fixing the ambiguities shows the benefit here as in the fixed solution 68% convergences in 10 minutes.

These statistics only hold after some datasets are excluded. Limitations in the RINEX 2 format result in problems when fixing ambiguities for some datasets. A larger number of datasets have unstable solutions even for the float scenario. This is likely caused by the user platform functionality that generates the PPP-RTK corrections. Because this has not been fixed, these problematic datasets have been identified and excluded.

PPP-RTK has also been tested using the newer constellations. For testing the software functionality a lot of datasets have been processed, but for analysis of the result only a select number of datasets are used. BeiDou and Galileo have sufficient satellites in orbit for testing PPP-RTK, but since the convergence time is of interest the number of satellites during the start of processing is controlled. Datasets are selected to start at times when a significant number of Galileo satellites are present.

For these tests it was shown that any number of satellites from BeiDou and Galileo benefit the convergence of the solution. Even though the float solution is not affected by a lot, the fixed solution is. The improvement for the fixed solution is cumulative. It benefits from having additional ambiguities to fix and the slight improved convergence for the float solution. With sufficient GPS satellites available three Galileo satellites can cut the convergence time from 10 to 5 minutes. With additional satellite BeiDou can cut this to 3 minutes. Combining all three allows the majority of the datasets to converge within 2 minutes.

If the number of GPS satellites is limited the convergence times do not improve to this level. However the benefit is still significant. For such a case the convergence time for 68% of the datasets using only GPS can be close to an hour. Adding Galileo cuts this to 20 minutes, while BeiDou cuts this to 15. Adding all three constellations pushes the convergence time of the fixed solution under 10 minutes.

With all these different tests done the Curtin PPP-RTK user platform can now be used for research in the future. Though the performance in both PPP and PPP-RTK is still lacking in some scenarios, many of the problems are expected to be resolved once all constellations are complete. Further improvement is expected once the PPP-RTK corrections are available from a global network, but a task for the user platform is also to assist in creating these global corrections. Testing these corrections in the future is one of the reasons the software has been developed.

7 Recommendations

As software testing is never finished some open questions still remain. The recommendations presented here refer to question that remain unanswered for different reasons. The unstable float solution is a known issue that still has to be resolved. The effect of the approximation for receiver side PCOs requires very specific datasets to be tested. The remainder are recommendations for follow-up research.

7.1. Unstable float solution

When using the single station products for PPP-RTK some datasets still present the problem of an unstable float solution. Several potential sources for this problem have been investigated. The problem is not linked to the version of RINEX used, as is appears for any version. The same holds for the different constellations as the problems show up for any multi-GNSS combination. Permanent receiver specific problems are not the cause as the unstable solution occurs for some days, while other days are unaffected. Further research must be done to identify the source of the problems, which is likely located on the side of the reference station. When further research is done on generating PPP-RTK products from a network instead of a single receiver the source of the unstable solution is likely to reveal itself. The reason for this is that when one station has a problem, while the others do not, the mismatch in products becomes immediately clear.

7.2. Effect of PCO approximation

For multi-GNSS analysis a recurring problem is the lack of antenna calibration data available for the frequencies used by newer constellations. The phase center offsets and variations are only provided for the first two GPS frequencies. When a different frequency is used the phase center is approximated by taking the data from the closest of the two GPS frequencies. The effect of this approximation can vary a lot between different antenna types, as can be seen by the different phase centers for the two GPS frequencies. For some antennas this difference is a few millimeters, but others show a 4 centimeter difference between L1 and L2. Testing for the effect is difficult. Galileo uses frequencies with and without calibrated antenna data, making it useful to test the effect of the approximation. However the orbit products used for Galileo are less precise and this is likely to cloud the effect of the phase center approximation. It is possible to overcome the orbit precision problem with the user platform. When a user and reference station are selected that are close together many orbit inaccuracies cancel. Because of the short baseline, single frequency PPP-RTK becomes possible, using the ionosphere delays from the reference station. If single frequency PPP-RTK using both GPS and Galileo is performed, using the L1 frequency for both, there should be no problem visible in the results. As soon as GPS L1 is combined with the L5 signal from Galileo, the solution can possibly become problematic. This problem depends on the antenna of both reference and user station. If both use the same antenna type the effect should cancel. If two different antennas are used that have widely varying phase centers the solution can show problems, as the observations for both constellations actually refer to a different position. Testing this is still difficult as it requires a very specific setup of receivers. Two receivers with the same antenna type and a third with a completely different antenna. All three have to track Galileo and be as close a possible, to allow for single frequency PPP-RTK. Performing such a test is more useful when more Galileo satellites are available, but might in the future also be obsolete, when antenna calibration data for Galileo is available.

7.3. Observation data from moving receiver

All the datasets used for testing were acquired by receiver that were stationary. No testing with truly kinematic data, hence a receiver that was moving, was done. When making use of positioning techniques in the field it is likely that in some cases the receiver is moving around. The reason for skipping these kinematic datasets for this research is two-fold. The first is that comparing the results to literature is almost impossible. The performance of every kinematic datasets is likely to vary quite a bit. The quality of the results depends on the number of satellites and the reception. Both these factors are completely different for each dataset used. To compare the performance of the user platform to any other software, the exact same dataset would have to be used. The second problem is investigates the accuracy of the estimated position. For the station IGS stations there is always the option to compare the results to an externally provided ground truth. For a moving receiver there is no such ground truth available. A common approach is to have more one receiver attached to the same moving vehicle. This way the distance between two receivers is fixed and this distance can be computed from the results of the two receivers. The problem is that this distance can still contain a bias. An example: a problem occurs in the software that causes a bias of 2 meters for the North component. Such a bias is completely unacceptable, but if it is present for both receivers computing the distance between the two does not indicate an issue. For further research it is useful to come up with methods to test truly kinematic datasets while having a benchmark that can discern a bias as was described above. Additional kinematic tests, without comparing to other sources, is useful in any case. With the reception problems that are more likely for moving receivers (like passing under some trees) more potential bugs can show up that have to be fixed.

7.4. External PPP-RTK products

The corrections required for multi-GNSS PPP-RTK are not yet available in the public domain. For this reason all tests for PPP-RTK have been done using products generated at Curtin University. At this moment these products are still generated from a single receiver, but in the future will be generated from a network of receivers. However, for GPS the corrections required for PPP-RTK are available from third party sources. In order to compare the performance of different products these third party products should also be used as input. Implementation of such functionality is difficult. It requires a precise understanding of the provided corrections. If the corrections are incorrectly applied this affects the carrier phase ambiguities. The assumption is made that these ambiguities have integer nature after applying the provided phase bias corrections. If this is not the case ambiguity resolution still yields a result, but the resulting fixed solution is very poor. Some initial tests have been done that indeed showed this problematic behavior. The best method to investigate the problem with the third party corrections is to compare these to corrections that do provide good results. However, since the products currently available are the limited single-station products, this comparison cannot be easily made. It is advisable to investigate the third party products, but only after in-house network products are available for comparison.

7.5. Verification of integer ambiguities

The position estimate after the ambiguities have been fixed is not verified in any of the performed tests. Ambiguity resolution always provides a solution, but whether or not this solution is correct is a second step. After processing data from a stationary receiver the epochs in which the ambiguities are correctly fixed can be separated from the wrong fixes. This can be done because a correct value for the position is known. During processing the actual position cannot be used to verify the fixed ambiguities. Different methods exist to assess the confidence to be had in a fixed solution. These methods have been implemented in the user platform, but have been turned off for any testing described. In the optimal case these methods would discard any wrong fixes, while keeping all correct fixes. Testing these methods on a large number of datasets is recommended and can be done with the user platform. However analysis of these results, taking into account the different available methods and settings for these methods, is beyond the scope of the current research.

Bibliography

[1] S. Banville, P. Collins, and F. Lahaye. GLONASS ambiguity resolution of mixed receiver types without external calibration. *GPS Solutions*, 17(3):275–282, 2013. ISSN 1521-1886.

[2] S. Bisnath and Y. Gao. Current State of Precise Point Positioning and Future Prospects and Limitations. In *Observing our Changing Earth*, pages 615–623. Springer Berlin Heidelberg, 2009.

[3] ESA. http://www.esa.int/our_activities/navigation/the_future_-_galileo/launching_galileo, 2016. last accessed 11-1-2016.

[4] gps.gov. Official U.S. Government information about the Global Positioning System (GPS) and related topics. website, 2016. last accessed 11-1-2016.

[5] C. Han, Y. Yang, and Z. Cai. BeiDou Navigation Satellite System and its time scales. *Metrologia*, 48, 2011.

[6] P. Heroux and J. Kouba. GPS Precise Point Positioning Using IGS Orbit Products. *GPS Solutions*, 5:12–28, 2001.

[7] IAC. https://www.glonass-iac.ru/en/. website, Information-Analytical Centre, 2016. last accessed 11-1-2016.

[8] A. Khodabandeh and P. J. G. Teunissen. Array-based satellite phase bias sensing: theory and GPS/BeiDou/QZSS results. *Measurement Science and Technology*, 2014.

[9] R. Leandro, H. Landau, M. Nitschke, M. Glocker, S. Seeger, X.. Chen, A. Deking, M. BenTahar, F. Zhang, K. Ferguson, R. Stolz, N. Talbot, G. Lu, T. Allison, M. Brandl, V. Gomez, W. Cao, and A. Kipka. RTX Positioning: The Next Generation of cm-accurate Real-Time GNSS Positioning. In *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011)*, 2011.

[10] X. Li, M. Ge, X. Dai, X. Ren, M. Fritsche, J. Wickert, and H. Schuh. Accuracy and reliability of multi-GNSS real-time precise positioning: GPS, GLONASS, BeiDou and Galileo. *Journal of Geodesy*, 2015.

[11] O. Montenbruck, R. Schmid, F. Mercier, P. Steigenberger, C. Noll, R. Fatkulin, S. Kogure, and A. S. Ganeshan. GNSS satellite geometry and attitude models. *Advances in Space Research*, 2015.

[12] D. Odijk, P. J. G. Teunissen, and B. Zhang. Single-Frequency Integer Ambiguity Resolution Enabled GPS Precise Point Positioning. *Journal of Surveying Engineering*, 2012.

[13] D. Odijk, B. Zhang, A. Khodabandeh, R. Odolinksi, and P. J. G. Teunissen. On the estimability of parameters in undifferenced GNSS network and PPP-RTK user models by means of S-system theory. *Journal of Geodesy*, 2015.

[14] RWG. RINEX version 3.02. Format definition, RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104), 2013.

[15] J. Saastamoinen. Atmospheric correction for the troposphere and stratosphere in radio ranging of satellites. *Geophysics Monograph Series*, 15:247–251, 1972.

[16] C. Satirapod, C. Rizos, and J. Wang. GPS single point Positioning with SA off: How accurate can we get? *Survey Review*, 36:255–262, 2001.

[17] P. J. G. Teunissen. The Least-Squares Ambiguity Decorrelation Adjustment: A Method for Fast GPS Integer Ambiguity Estimation. *Journal of Geodesy*, 70(1-2):65–82, 1995.

[18] P. J. G. Teunissen and A. Kleusberg. *GPS for geodesy*. Springer-Verlag Berlin Heidelberg New York, 1998.

[19] R. J. P. van Bree and C. C. J. M. Tiberius. Real-time Single-frequency Precise Point Positioning: Accuracy Assessment. *GPS Solutions*, 16(2):259–266, 2012. ISSN 1521-1886.

[20] R. J. P. van Bree, P. J. Buist, C. C. J. M. Tiberius, B. van Arem, and V. L. Knoop. Lane Identification with Real Time Single Frequency Precise Point Positioning: A Kinematic Trial. In *ION GNSS proceedings 2011*, 2011.

[21] F. Zhang, H. He, B. Tang, F. Shen, and R. Chen. Analysis of Signal Characteristic and Positioning Performance Affected by Pseudorange Multipath for COMPASS. *China Satellite Navigation Conference (CSNC) 2013 Proceedings*, page 493–503, 2013. ISSN 1876-1119.

Appendices

A

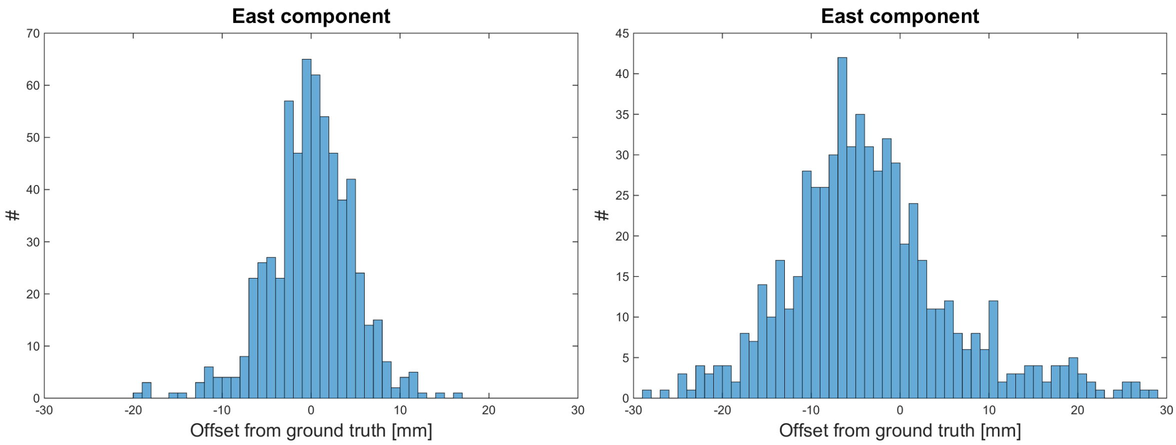
Bias in height

During the testing of standard PPP the empirical standard deviations and the offset from the provided ground truth were determined. The results are shown in chapter 4. For the initial tests a large bias was found for the up component. Table A.1 shows the results for the tests of standard PPP. The data was processed in static mode, so the receiver is assumed to be stationary. For all the datasets an offset of around 3 centimeters was found. The histograms for the East and North components (figures A.1 and A.2) show a distribution around zero. However the histogram for the up component (figure A.3) is problematic.

The cause of this bias was tracked down to the receiver PCV. The related receiver PCOs were already applied at this point as these cause a significant difference between the actual position of an antenna and the estimated position. The PCVs describe the change in this offset under different angles to the satellites. While this has no significant influence on the East and North components, for the up component the effect was up to 3 centimeters.

Day	02/09/2014		14/12/2013	
Year - day of year	14-245		13-348	
Stations	357	357	409	408
Orbit & clock products	IGS final	CNES real-time	IGS final	CNES real-time
Mean σ_E [mm]	3.1	4.6	5.1	6.9
Mean σ_N [mm]	1.6	2.2	2.1	2.5
Mean σ_{up} [mm]	7.1	8.3	9.7	10.2
Stations with known ground truth	316	316	306	306
Mean offset East [mm]	0.0	-1.4	-0.1	-5.2
Mean offset North [mm]	0.4	0.5	2.6	-1.5
Mean offset up [mm]	-31.8	-30.1	-36.7	-36.7

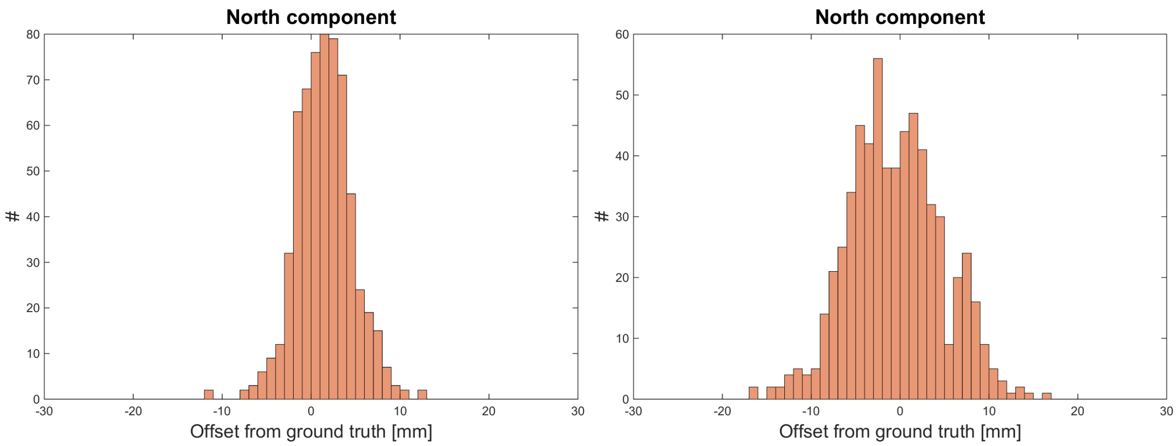
Table A.1: Two processed days with two different sources for orbit and clock products. The average standard deviations are given for the North, East and up components. For the receiver for which the ground truth is known the mean offset between the estimated position and the ground truth is provided.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

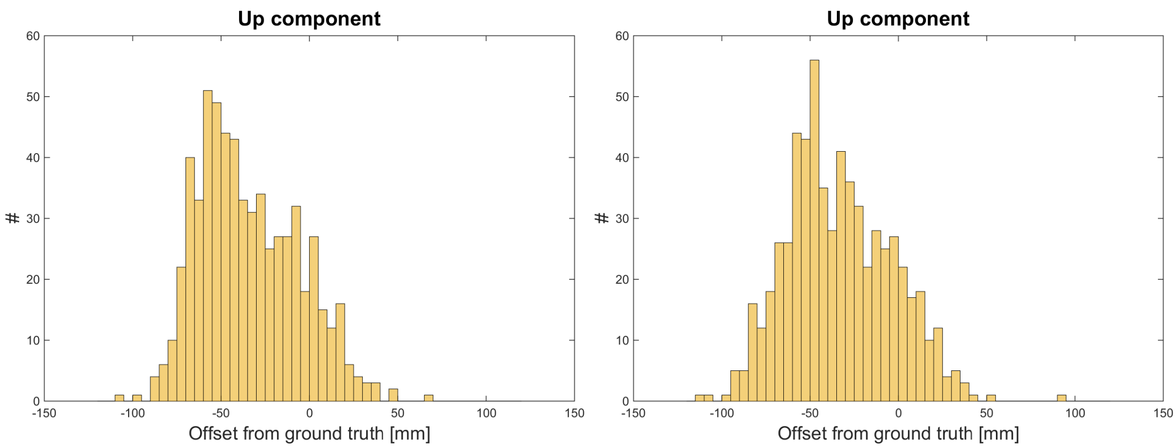
Figure A.1: Histograms for offset between estimated position and ground truth in the East component. Static positioning is used.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure A.2: Histograms for offset between estimated position and ground truth in the North component. Static positioning is used.



(a) IGS orbit and clock products.

(b) CNES orbit and clock products.

Figure A.3: Histograms for offset between estimated position and ground truth in the up component. Static positioning is used.

B

Software description

The Curtin PPP-RTK user platform has been developed in Matlab. It consists purely of routines written for Matlab, so no external software is used. The only external input for the platform are the settings (provided in a steering file), observations and corrections used. This section describes the main functionality of the user platform and the different ways the software can be configured.

B.1. Steering files

The only input when executing the main routine of the user platform (*ppprtk-user.m*) is the path to the so called steering file. This file contains all parameters required to process a given dataset. Some settings are necessary for the software to run (for example the location of the observation data), other settings have default values that are used if no value is provided. The first section of an example steering file is provided in figure B.1.

```
% The folder and files RINEX / SP3 files
folder       = ../../../data/        =% folder to save auxiliary files
output       = output/example        =% folder to save generated GNSS data structure
phcfile      = support/ant/igs08.atx =% GNSS satellite-receiver PCV and PCO file
satinfo      = support/ant/igs08.atx =% GNSS satellite information
pco_off      = 0                     =% Turn of PCO's for GPS
%-------------------------------------------------------------------------------
rnxfile.obs  = input/example/obsfile.15o    =% RINEX observation file
%-------------------------------------------------------------------------------
rnxfile.nav1 = input/example/navfile.15n    =% input GNSS navigation files
rnxfile.nav2 =                              =% input GNSS navigation files
prefile.orb1 = input/example/orbfile1.sp3   =% GNSS sp3 orbit file
prefile.orb2 = input/example/orbfile2.sp3   =% GNSS sp3 orbit file
prefile.clk1 = input/example/clkfile1.clk   =% GNSS sp3 clock file
prefile.clk2 = input/example/clkfile2.clk   =% GNSS sp3 clock file
%-------------------------------------------------------------------------------
```

Figure B.1: Steering file part 1

The first setting is the folder in which the data to be processed is located. This folder (here indicated by 'data') contains four standard sub-folders. The folder 'input' contains all the data specific to the current dataset to be processed. The folder 'support' contains files that are required but are not specific to the dataset currently being processed. The 'output' folder is used to store the results and potential images generated from these results. Not shown in figure B.1 is the fourth folder, 'steering', in which different steering files are stored. This folder structure is not required for the software to function but is a useful structure to organize the data.

Figure B.1 also shows the most important file required. This is the RINEX observation file containing the observations to be used for each epoch to be processed. Also the precise orbit and clock files can be provided. Optionally the navigation files can be given as well. All these file have a number as an indicator for which system this data is used. As the software is designed for multi-GNSS several constellations can be used simultaneously. Each system can use its own orbit and clock data files with these settings.

The second section of the example steering file is shown in figure B.2. The top of this section shows optional files that can be included. Among these are differential code biases and the single station PPP-RTK products. One of these last two is only required when performing ambiguity resolution. When using only a single frequency additional information in the ionosphere is required, which can be provided using GIM. The 'ionfile' option allows for including such a file. The latter part of the section displayed in figure B.2 deals with the Detection Identification and Adaptation (DIA) procedure used for outlier detection.

```
%--------------------------------------------------------------------------------
p1c1file     = input/example/P1C11234.DCB   =% P1-C1 DCB
p1p2file     = input/example/P1P21234.DCB   =% P1-P2 DCB
ionfile      = input/example/codg1234.15i   =% ionosphere delays file (GIM)
pparfile     = input/example/refppar.mat    =% Curtin single receiver product
ocnlfile     = input/example/1234_ocn.blq   =% Ocean loading file
%--------------------------------------------------------------------------------
IA           = 1      =% Specify whether (1) or not (0) the identification
%                        and adaptation procedures should be performed
alfa0        = 0.001  =% level of significance of 1-dimension statistical test
power        = 0.80   =% power of statistical test, i.e. probability of
%                        correctly rejecting null hypothesis
%--------------------------------------------------------------------------------
```

Figure B.2: Steering file part 2

The third section of the steering file is shown in figure B.3. Here the selection of the systems is done. The order given for 'ReqSys' determines the order of the systems for the other variables. This means that 'ReqObs1' lists the required observation types for the first selected GNSS. Different types of orbits can be selected as well. The observations types provided should match those that a available in the RINEX observation file. The next part allows for choosing a cut-off elevation for the satellites. Also the method of interpolating the sp3 orbits can be varied.

Following this are several settings for the way the observation file is read. The setting of 'DeltaT' controls the time interval to be used. This value is compared to the interval defined in the RINEX observation file. If the observation file has a smaller interval than the required one a number of epochs is skipped from the data in order to match the required time interval. If the observation file has an interval larger than the desired interval the observation file time interval is used instead. The variables 'k0' and 'kend' are used to set a first and last epoch to be processed in a given dataset. For testing purposes it is also possible to exclude specific satellite from being used for computation. Finally a ground-truth position can be provided. This information is used to determine the offset of the estimated position from the true location. This is only used for plotting the results when assessing the quality of these results.

The fourth section of the steering file contains the settings for the stochastic model. The example file is presented in figure B.4. The standard deviations for the phase and code observables have to be provided for each GNSS and each frequency. This example shows the fields for a three frequency first system and a four frequency second system. For the other parameters indicated the standard deviations control the behavior of these parameters within the generalized filter. The indicator 'Inf' is used by the software to set the use of this parameter without a dynamic model. This means that this parameter is completely free to vary between epochs. By setting the variable 'sq' to infinite the software processes the data in purely kinematic mode. When the variable is set to 0 the data is processed in a purely static mode.

The final part of the steering file is shown in figure B.5. The first setting controls the way the ionosphere is estimated. If there is ionosphere information available from a reference station or network this can be used. Otherwise the ionosphere is estimated by the software itself. If a phase bias file has been provided the option for ambiguity resolution can be turned on. When this is enabled the last setting in the file can be used to control the success and failures rates, as well as the ratio test.

```
%-------------------------------------------------------------------------------------
ReqSys       = GE     =% requested GNSS's ('G'=GPS,'R'=Glonass,'E'=Galileo,'C'=BDS)
ReqOrb       = PP     =% requested orbits per GNSS ('B'=broadcast, 'P'=precise)
ReqObs1      = C1C L1C C2W L2W C5Q L5Q =% requested observables from GPS Rinex files
ReqObs2      = C1C L1C C5Q L5Q C7Q L7Q =% requested observables from GPS Rinex files
%-------------------------------------------------------------------------------------
CutOff         = 10       =% cut-off elevation in degree
SP3order       = 9        =% lagrange interpolation order (in case of sp3 orbits)
MinSP3Accuracy = 10       =% minimum accuracy for SP3 orbit data to be used
%-------------------------------------------------------------------------------------
% The start and finish time and the sampling interval
DeltaT            = 30     =% desired sampling interval in sec
remsat            =        =% list of satellites to be removed a priori
remsat_epoch      =        =% Epochs for which the satellites are removed
k0                =        =% first epoch
kend              =        =% last epoch
kn                =        =% Number of epochs before integer ambiguities of
%                             newly risen satellite are estimated
%-------------------------------------------------------------------------------------
% Receiver ground truth
rec.xyz_gt        =    -2414152.1194 4907778.5829 -3270644.4709
%-------------------------------------------------------------------------------------
```

Figure B.3: Steering file part 3

B.2. Main routine - Pre-processing

The structure of the software is depicted in a graphical way in figure B.6. The first part of the routine *ppprtk-user.m* is the initialization. The first step is reading the steering file that has been previously described. From the information provided in the steering files all specified files are read into Matlab. The only exception is the RINEX observation file. This one is processed per epoch. The full orbit and clock files are read in full. The clock files can be quite large and take several minutes to be read. For this reason a separate Matlab variable file is saved containing the information from the clock file. Any subsequent processes using the same clock file can be sped up in this way. Files with phase bias corrections are also fully read at this stage. The correct information per epoch needed by the software is extracted at a later stage by the software.

B.3. Main routine - Iterative section

The central section of the routine *ppprtk-user.m* loops over the epochs of data provided. The first section (Data reading figure B.6) of the loop reads new information from the RINEX observation data. This is done by the routine *readREC.m*. The information extracted from the files depends on the GNSSs in use, as well as the observations requested per GNSS. The first time the observation file is accessed also the information from the file header is read. This contains the antenna offsets and the time interval used in the observation file. Next to the observation data that is provided also the information from the 'loss of lock' indicator is returned.

The next step is organizing the data (Data filtering figure B.6). From the phase bias corrections, if provided, the correct data for the epoch currently processed is selected. The same step is performed for the orbit and clock data. The routine *removesat.m* now checks the information for all satellites. If any crucial information for the satellite is not available or not suitable (for instance the orbit data) the satellite is removed from the dataset for the current epoch. In future epochs the satellite can still be used, but at this stage the information provided cannot be used for positioning.

After organizing the data the routine *geometryfree_DIA.m* is called. This routine is used to deal with very large outliers in the data. Any satellites showing very problematic observations are completely removed from the dataset for this epoch. This routine mainly deals with the large outliers that can be caused by a loss of lock that was not indicated by the RINEX observation file.

The following step in the main routine is the single point positioning. This is done by the routine *singpoint-pos.m* and serves two main purposes. The first is to provide an initial estimate of the position. This is later

```
%-----------------------------------------------------------------------------------
% Stochastic model settings for the specified phase, code, ionosphere and
% range observables and Kalman filter dynamic model settings
%-----------------------------------------------------------------------------------
sp{1}      = [0.003 0.003 0.003]        =% Std [m] of undifferenced phase observable
sp{2}      = [0.003 0.003 0.003 0.003]  =% Std [m] of undifferenced phase observable
sc{1}      = [0.30 0.30 0.30]           =% Std [m] of undifferenced code observable
sc{2}      = [0.30 0.30 0.30 0.30]      =% Std [m] of undifferenced code observable
si         = .5    =% Std [m] of undifferenced ionospheric delays on
%                     the first frequency
sq         = .1    =% Std [m] of system noise (velocity/acceleration as white noise)
sqt        = 0.001 =% Std [m] of tropospheric zenith delays
sqc        = Inf   =% Std [m] of receiver+satellite clocks
sqd        = Inf   =% Std [m] of phase+code RECEIVER instrumental delays
sqi        = 1     =% Std [m] of ionospheric delays
sqa        = 0     =% Std [m] of DD ambiguities
%-----------------------------------------------------------------------------------
```

Figure B.4: Steering file part 4

```
%-----------------------------------------------------------------------------------
% Ionosphere-float or ionosphere-weight settings
%-----------------------------------------------------------------------------------
iono       = 0      =% float(0) / weighting (1) / fixed (2)
ppprtk     = 0      =% PPP-RTK (1) or standard PPP (0)
pco_off    = 0      =% Turn of PCO's for GPS
%-----------------------------------------------------------------------------------
% Ambiguity resolution settings
%-----------------------------------------------------------------------------------
parfix     = 1      =% Perform full (0), partial (1), or
%                      widelane (2) ambiguity resolution
P0         = 0.999  =% Expected minimum success rate in case of partial fixing
rtmu       = 1      =% Specify critical value mu of Ratio test. If empty,
%                      model-driven Ratio Test with fixed failure rate is executed.
%                      If 1, Ratio Test is not carried out (integer solution always
%                      accepted in that case)
Pf         = 0.001  =% Fixed ambiguity failure rate (in case critical value mu
%                      of Ratio test empty (only 0.01 and 0.001 are allowed))
%-----------------------------------------------------------------------------------
```

Figure B.5: Steering file part 5

used by the main filter routine and is especially important if there is no dynamic model for the position (in this case no information is available on the position from a previous epoch). The single point positioning is also used to determine the positions of the satellites at the time of transmission, as well as the satellite clocks at that time. When the positions of the satellites are known the routine also checks the elevation of the satellites. If a satellite has a lower elevation than set for the minimum elevation it is discarded.

After this the information is passed to the generalized filter. This is done in the routine *genfilt.m*. The first step in the routine is comparing the available satellites in the current epoch with the ones in the previous epoch. Depending on the setting from the steering file any parameters can be set to have a dynamic model, or not. Any new satellites do not have information from a previous epoch even though a dynamic model is set. Since the observation equations used are not linear the generalized filter is applied in an iterative way. This iteration is performed by the routine *genfilt_iter.m*. Depending on the setting for the dynamic model of the receiver position either the estimated position from the previous epoch, or the position from the single point positioning is used. After the filtered solution has converged the solution is tested for outliers. This is done by the DIA procedure. If the procedure is successfully passed (either with or without any adaptation) the process continues and a state vector with the float solution for the position is provided. If for some reason the overall model test still is not passed after the DIA procedure the epoch is skipped. A prediction is added to the state vector from the previous epoch so it can be used in the next epoch. The epoch is also skipped in the same way if there are insufficient satellites to solve the model.

As indicated in the steering file ambiguity resolution is optional. If enabled it is performed at this stage. The procedure *ambres.m* handles the attempt at ambiguity resolution. Several different options for ambiguity resolution are possible but in every case the procedure returns an integer estimate for the phase ambiguities and a flag to indicate whether or not this estimate is acceptable. The acceptance of the integer ambiguities depends on the quality check indicated in the steering file. If the result is accepted a modified state vector with the fixed solution for the position is given as output.

At the end of the loop all data is stored to respective variables. The exact output depends on whether the processing was successful. When for instance the ambiguity resolution was not successful or not performed, the fixed position result consists of 'Not-a-Number' values.

B.4. Main routine - Post-processing

After all epochs in the RINEX observation file are processed the results from all epochs are organized. All the data is grouped and saved in a single Matlab file. This file is saved in the output folder as provided in the steering file. The saved results can be plotted afterwards, but optionally can also be directly plotted from the main routine. The plots that are generated differ on the type of processing done. For standard PPP the following is generated:

- Main information figure
  - Time series of estimated position is East, North and up component
  - Plot with the number of satellites
  - Results for the LOM test and its critical value
- Separate time series for the three components
- Number of satellites (per GNSS).
- Scatter plot in North and East components

If for the processed station a ground truth location is given the time series are provided with the respect to the ground truth. If no ground truth is provided the estimates are given with respect to an average value. In the case of PPP-RTK additional figures are generated. The first is similar to the main information figure, with the time series, number of satellites and LOM test information. In this case the position after ambiguity fixing is provided in the figure. Additionally the separate time series and scatter plot now also contain the ambiguity fixed results next to the float solution.

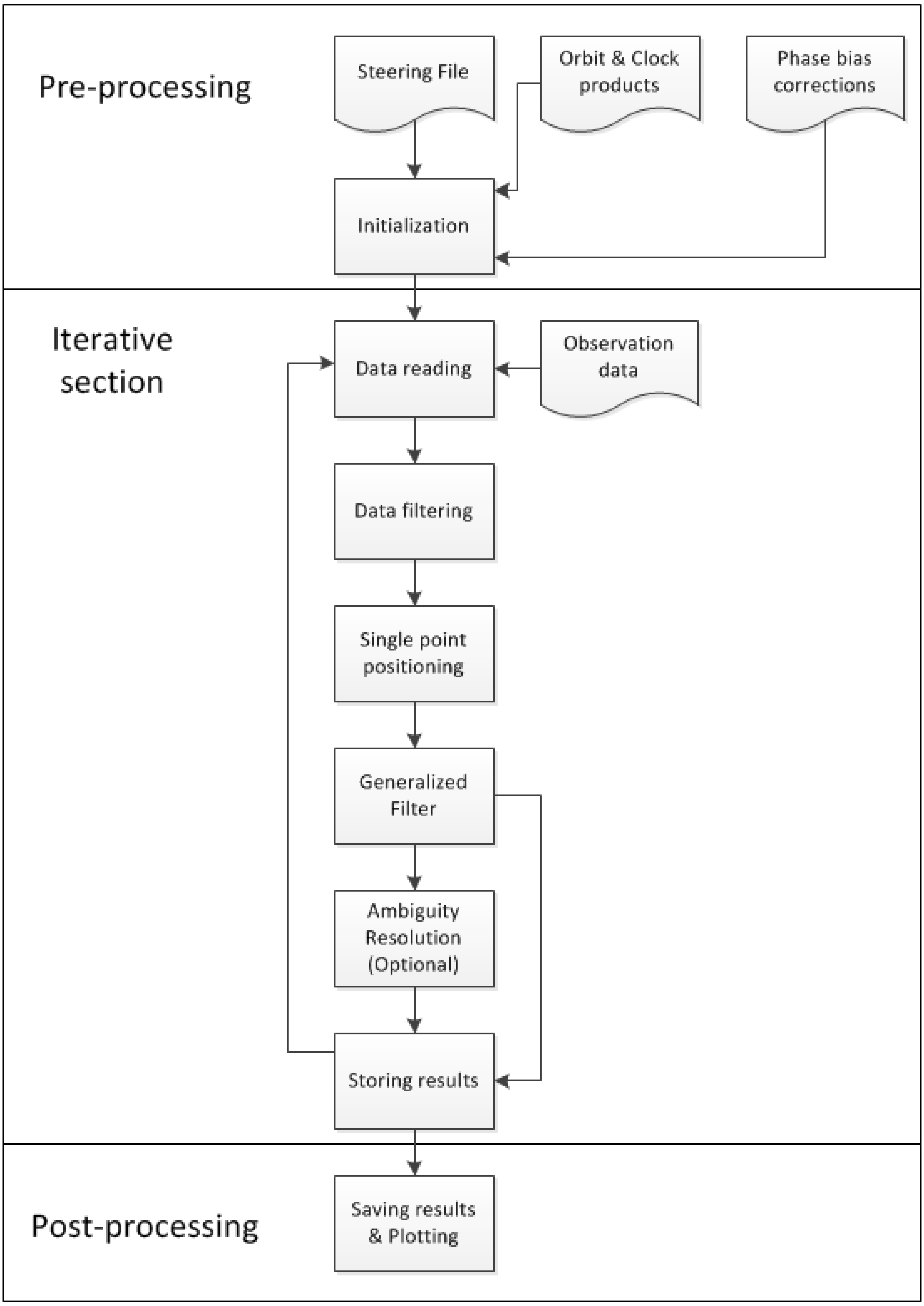


Figure B.6: Graphical representation of user software

C

Testing Tools

The only way to truly test the functionality and performance of the Curtin PPP-RTK user platform is to process large amounts of real observation data. For every scenario a steering file is required, indicating which observation data and products have to be used. Setting up a steering file does not take a lot of time, but in order to process a lot of datasets it is useful to automate this process.

C.1. Generate a steering file

When processing a large number of datasets usually all files are processed with the same settings. For instance, one wants to process a hundred observation files for one day with the same dynamic model and stochastic settings. The only difference between the steering files required for all these tests is the name of the user station and its observation file. It is practical to assume that the name of the station is also present in the filename chosen for saving the results. The Matlab function *generate_steering_file* is used to generate a steering file for a station. The first input is a source file, a steering file that is correctly configured for the chosen scenario. The only thing missing in this steering file are the setting different for each of the tests. In this example that would only be the station name. In the source file the station name is replaced by a marker. This marker can be anything, but it should be a unique combination of characters not present anywhere else in the steering file. The preferred marker for a user station is *SSSS*. The Matlab function takes as additional input a list of markers and a list of replacements. All markers specified in the steering source file are replaced by the replacements defined for each marker. In the simple scenario described the marker *SSSS* is present in the name of the output file and the name of the observation file. The last input for *generate_steering_file* is the filename of the new steering file to be written. This file shall be exactly the same as the source, with only the markers replaced.

This general example can be easily extended. Any number of markers can be provided to the function. A more extensive example is the case of multi-GNSS PPP-RTK. Because the observation types for each constellation can be different per steering file a marker can be specified for the observations for each system (ReqObs). The observation file is also different for each steering file as is the filename for the corrections being applied. The only important consideration is making sure each marker is unique, but since the length of the markers can be chosen this should not be a problem.

```
function generate_steering_file(fsource, markers, replacements, dest_file)
%GENERATE_STEERING_FILE Uses a base steering file with markers to generate
%a new file with specified variables
%   fsource - source steering file (with markers)
%   markers - cell array with markers specified
%   replacements - cell array with data to replace markers
%   dest_file - filename of newly generated steering file

    %Extension should be .txt

```

```
    if ~(strcmp(fsource(end-3:end),'.txt'))
        fsource = [fsource '.txt'];
    end

    if ~(strcmp(dest_file(end-3:end),'.txt'))
        dest_file = [dest_file '.txt'];
    end

    % Open source file and create new file
    source = fopen(fsource);
    target = fopen(dest_file,'w');

    while(~feof(source))
        line = fgetl(source);

        %looping through all provided markers
        for i = 1:length(markers)
            line = replace(line,markers{i},replacements{i});
        end
        fprintf(target,'%s\n',line);
    end

    fclose(target);
    fclose(source);
end

function [line] = replace(line, marker, replacement)
    % Replace all markers by their replacements in a line
    k = strfind(line,marker);
    for i = 1:length(k)
        newline = [line(1:k(i)-1) replacement line(k(i)+4:end)];
        line = newline;
    end

end
```

generate_steering_file.m

C.2. Selecting available observations

As indicated the observation types available in a RINEX observation file can be quite different for each station. Also there is a difference between the versions of RINEX used. The function *obs_select* allows the selection frequencies per constellation instead of defining the observation types. The available observation types are determined from the header of the observation file. A list of requested frequencies is provided to the function. The frequency are requested as a string with the first character representing the constellation. Any integers after the first character are treated as frequencies requested for this constellation. As an example *G125* returns the code and phase observation types for the three frequencies of GPS. If the L5 is not available in the RINEX header only the observation types for the frequencies that are available are returned. For RINEX version 2 only GPS is supported but the observation types corresponding to the requested frequencies are still provided if available (L5 returns nothing). If a constellation is not present an empty string is returned for that request. For some constellations more than one option for phase and code is available per frequency. In this case *obs_select* returns observation types based on an internal preference list. The function for selecting the observations based on frequency has also been implemented into the user software.

C.3. Approximate station location

When testing PPP-RTK corrections by a reference station are used. In most cases more than one correction file is available and therefore one is selected. As this selection is most of the time based on the distance between user and reference station it is useful to automate the look-up of this position. Most RINEX observation files contain an approximate receiver location in the header. The precision of this estimate is accurate enough for the purpose of determining the approximate baseline between user and reference station. The Matlab function *find_approx_loc* reads a RINEX observation file header to determine the approximate location. If none is specified it returns the XYZ position [0,0,0]. This can be used as an indication that the information is missing. Some observation files also specify [0,0,0] if the location is unknown. If this is the case processing the file using standard PPP is sufficient to get an approximate value for the receiver.

```
function [ pos_xyz ] = find_approx_loc( obs_file )
%FIND_APPROX_LOC Summary of this function goes here
%   Detailed explanation goes here

    fid = fopen(obs_file);
    pos_xyz = [0; 0; 0];

    while ~feof(fid)
        line = fgetl(fid);
        if ~isempty(strfind(line,'APPROX POSITION XYZ'))
            pos_xyz = sscanf(line,'%f %f %f');
        elseif ~isempty(strfind(line,'END OF HEADER'))
            break
        end

    end
    fclose(fid);
end
```

find_approx_loc.m

C.4. Generating multiple steering files

Generating a batch of steering files is done using a Matlab script named *gen_steer*. Different examples for this script can be given. For the simple scenario of multiple station on the same day only the observation files a different between steering files. In this case the script only needs to read all observation files from a specified folder and generate a steering file for each observation file. The orbit and clock products are the same for each station, so these only need to be specified in the source steering file.

```
clear;

ext = '.15o';
day = '1680';
source_file = 'steering_source.txt';
steer_folder = 'steering/';
obs_folder = '../obs-data/15168/';

%Search all observation files in designated folder.
obs_files = dir(obs_folder);

i = 1;
while i <= length(obs_files)
    if ~isempty(strfind(obs_files(i).name,ext))
        i = i + 1;
    else
        obs_files(i) = [];
    end
end

% Define markers in steering source file.
markers = {'SSSS','GGGG',};

for i = 1:length(obs_files)
    station = obs_files(i).name(1:4);
    obs = obs_select([obs_folder obs_files(i).name],{'G12'});
    if isempty(obs{1})
        continue;
    end
    replacements = {station,obs{1}};
    % Steering file to save
    target = [steer_folder 'steering_' station '.txt'];
```

```
    generate_steering_file(source_file,markers,replacements,target);
end
```

gen_steer.m basic

A more advanced case is processing the same station but for multiple days. In this case the filename for the observation file changes for every station. The same way the orbit and clock products advance by one day every dataset. Another problem are the P1/C1 DCBs, which are given on a monthly basis. It is possible to determine which files are required rigorously, but when generating steering files for testing this is not important. The example given works for any year, as long as a small modification is made for leap years.

```
clear;

ext = '.14o';
station = 'mobs';

source_file = 'steering_source.txt';
steer_folder = ['steering/' station '/'];
obs_folder = ['../obs-data/14000/' station '/'];

first_day = 1;
last_day = 365;
gps_day_start = 17732; %GPS day before 'first_day'

%Search all observation files in designated folder.
obs_files = dir(obs_folder);
OC1 = gps_day_start-1;
months = cumsum([31 28 31 30 31 30 31 31 30 31 30 31]);

% Define markers in steering source file.
markers = {'SSSS','DDDD','GGGG','PPPP','RRRR','QQQQ','MMMM'};

for day = first_day:last_day
    day_str = [num2str(day,'%.3i') '0'];

    % Cycle through orbit and clock products
    OC1 = OC1 + 1;
    if mod(OC1,10) == 7
        OC1 = OC1 + 3;
    end
    OC2 = OC1 + 1;
    if mod(OC2,10) == 7
        OC2 = OC2 + 3;
    end
    OC3 = OC2 + 1;
    if mod(OC3,10) == 7
        OC3 = OC3 + 3;
    end

    %Check if observation file for requested day is available.
    obs_file = [obs_folder station day_str ext];
    if ~exist(obs_file,'file')
        disp(['missing: ' obs_file]);
        continue;
    end
    obs = obs_select(obs_file,{'G12'});

    %P1C1 DCB
    P1C1dcb_rep = num2str(find(months >= day,1),'%.2i');
    %Orbit and clock files
    OC1str = num2str(OC1,'%.3i');
    OC2str = num2str(OC2,'%.3i');
    OC3str = num2str(OC3,'%.3i');
    replacements = {station,day_str,obs{1},OC1str,OC2str,OC3str,P1C1dcb_rep};
    target = [steer_folder 'steering_' station day_str '.txt'];
    generate_steering_file(source_file,markers,replacements,target);
end
```

gen_steer.m station for multiple days.

When selecting the correct reference station for PPP-RTK the situation becomes more complex. The approximate locations of the user and reference stations can be determined. In this case the folder with the correction from the reference stations is assumed to also contain the observation files so the location can be determined. One option is to select the closest reference station for each user, but a more complex situation is selecting a lower and upper limit for the baseline length. In this case all pairs that fall between these limits can be formed and a steering file is generated for each pair. For analysis it is useful to add the name of the reference station to the filename for the results file. Even the baseline length can be added with the correct markers and replacements.

A last example is the case of multi-GNSS. Because not all stations observe all constellations generating steering files is conditional. If not all requested constellation have observation types in the RINEX header no steering file is generated. This can be determined from the response of *obs_select* as it returns an empty field if no observation types for a constellation are found. Sometimes it is useful to test the number of observation types returned as problems may arise if only a single frequency is returned.

```
clear;

ext = '.15o';
day = '1680';
source_file = 'steering_source.txt';
steer_folder = 'steering/';
obs_folder = '../obs-data/15168/';

%Search all observation files in designated folder.
obs_files = dir(obs_folder);

i = 1;
while i <= length(obs_files)
    if ~isempty(strfind(obs_files(i).name,ext))
        i = i + 1;
    else
        obs_files(i) = [];
    end
end
% Define markers in steering source file.
markers = {'SSSS','GGGG','CCCC'};


for i = 1:length(obs_files)
    station = obs_files(i).name(1:4);

    obs = obs_select([obs_folder obs_files(i).name],{'G12','C27'});
    %Test whether or not observations are found
    if isempty(obs{1}) || isempty(obs{2})
        continue;
    end

    replacements = {station,obs{1},obs{2}};
    target = [steer_folder 'steering_' station '.txt'];
    generate_steering_file(source_file,markers,replacements,target);
end
```

gen_steer.m station for multiple constellations.

Many other test cases are possible for testing the software. For each case the script *gen_steer* has to be adapted to suit the scenario.

C.5. Processing a steering folder

After the generation of a number of steering files these have to be processed. The Matlab script *run_steering_folder* is used to process all steering files present in a folder. To speed up this process the steering files are processed in parallel. The more cores available on the computer in use the more 'workers' are assigned by Matlab to process the steering files. Every instance of the user software is independent so processing in this way is possible. The only requirement is that all steering files need a different filename to save the results, otherwise problems will arise.

When the steering files are processed all exceptions in the user software are captured. If the software fails for one steering file this ensures that the other files are still processed. If an exception occurs for a steering file the output of the exception is stored in the variable 'exceptions'. This can later be accessed to identify the problem. The variable 'processed' can contain three different values for each steering file. Not-a-Number indicates that the file in question has not yet been processed. An exception during processing leads to the value 0. The value 1 indicates that the file was successfully processed.

In the script the first thing to assign are the location of the user software and the location of the folder containing the steering files. The script can be run in different modes. Mode 0 processes only the steering files that have not yet been touched (hence have the value NaN in 'processed'). Mode 1 processes untouched files and files that previously gave an exception. This is used when one has some hope that a bug has been corrected. Mode 2 processes all files ignoring previous results. All modes except 2 also check the number of steering files and the length of 'processed' to make sure that one is not processing a different folder than in the previous run.

The variable 'batch-size' controls the number of steering files that are sent to the parallel loop at once. When the parallel loop is interrupted by the user the variable 'processed' is not always updated, so some files that were completed are still shown as untouched. By dividing the task into batches the variable 'processed' is sure to be updated after every batch. The setting 'burst' is used when first testing a set of steering files. If this is set to 1 only the first batch of files is processed. When problems are expected with a new set of steering files it is usually not practical to immediately process all datasets. A small subset can be sufficient.

This batch processing has two weaknesses. If the software pauses all workers get stuck and no further data is processed. Two causes for this happening are infinite loops and a left-over 'keyboard' command, which causes the software to wait for input.

```
clearvars -except processed exceptions
clc
close all

%-----------------------------------------------------------------------
%This script processes all steering files in a folder using the Curtin
%PPP-RTK user software. All files in the folder are processed. In the case
%of successful processing the steering file is flagged with 1 in the list
%'processed'. If an exceptions is triggered it is stored in the list
%'exceptions'. This also causes the steering file to be flagged with 0 in
%'processed'. The list 'processed' is not cleared, so it can be reused when
%the steering files were partially processed. The option 'MODE' can be used
%to either only process untouched files (MODE = 0) or process files that
%previously resulted in exceptions (MODE = 1). To process all: MODE = 2
%
%The 'batch_size' controls how many files are sent to the parallel
%for-loop. Higher values give a slight decrease in processing time, but
%only after completion of each batch the results are sure to have been
%saved to 'processed' and 'exceptions'. Setting BURST = 1 allows only the
%first batch to be processed, for testing purposes.

%Set full path to steering folder
steering_file = 'data/steering/';

%Set full path to user platform
ppprtk_folder = 'ppprtk_user\branches\Curtin_PPP_RTK_user\src\';
MODE = 2;
batch_size = Inf;
BURST = 0;
```

```
D = dir(steering_folder);

steering_files = {};

for i = 3:length(D)
    steering_files{i-2} = D(i).name;
end

%Protection against running a different folder by accident
if ~exist('processed') || MODE == 2
    processed = nan(length(steering_files),1);
elseif ~(length(processed) == length(steering_files))
    disp('List of processed files might not belong to current folder');
    keyboard;
end

exceptions = cell(length(steering_files),1);

%Switch to user platform folder
current_folder = cd;
cd(ppprtk_folder);

%Determine interval from batch size
interval = [0:batch_size:length(steering_files) length(steering_files)];

%Loop over batches
for j = 1:(length(interval)-1)

    %Call parallel loop
    parfor i = (interval(j)+1):interval(j+1)
        opt_flpath = fullfile([steering_folder steering_files{i}]);

        %conditional processing
        if ~(processed(i) >= MODE)
            try
                disp(['Processing ' steering_files{i}]);
                ppprtk_user(opt_flpath);
                processed(i) = 1;
            catch exception
                disp([steering_files{i} ' processing failed']);
                processed(i) = 0;
                msgString = getReport(exception)
                exceptions(i) = {msgString};
            end
        end
    end
    if BURST
        %Break after first loop
        break
    end

end

cd(current_folder)
```

run_steering_folder

C.6. Processing a single file

The Matlab script *run_single_file* is very simple. The only feature is that the variables ‘processed’ and ‘exceptions’ are not cleared. The *run_single_file* script is used to investigate a single steering file when problem occur. To keep track of what other exceptions occurred in a folder tested with *run_steering_folder* these two variables are kept.

C.7. Processing multiple steering folders

The script *run_steering_folders* is a modified version of the earlier described script. This does no longer save any exceptions but only processes steering files. It is possible to define multiple steering folders so these are processed one after another. If exceptions occur these are simply ignored so the other files are still processed. This is used to process a large number of files once no more bugs are expected.

```
clearvars -except processed exceptions
clc
close all

%-----------------------------------------------------------------------
%This script processes all steering files in several folders using the
%Curtin PPP-RTK user software. All files in the folders are processed.

%Specify all steering folders to be processed
steering_folder = {};
steering_folder{end+1} = 'data/steering_folder1/';
steering_folder{end+1} = 'data/steering_folder2/';
steering_folder{end+1} = 'data/steering_folder3/';

%Set full path to user platform
ppprtk_folder = 'ppprtk_user\branches\Curtin_PPP_RTK_user\src\';

current_folder = cd;
cd(ppprtk_folder);

for j = 1:length(steering_folder)
    %Loop through folders
    D = dir(steering_folder{j});
    disp(steering_folder{j});

    steering_files = {};

    for i = 3:length(D)
        steering_files{i-2} = D(i).name;
    end

    %Call parallel loop
    parfor i = 1: length(steering_files)
        opt_flpath = fullfile([steering_folder{j} steering_files{i}]);

        try
            disp(['Processing ' steering_files{i}]);
            ppprtk_user(opt_flpath);
        catch exception
            disp([steering_files{i} ' processing failed']);
        end
    end
end




cd(current_folder)
```

run_steering_folders

D

Testing

This chapter provides a description of the issues that have been uncovered during testing. The real observation data that is used is likely to provide a lot of unexpected situations or corner cases. These were most of the time not expected during the development of the software and are therefore not correctly handled. During the software development a select number of datasets have been used, but for testing many more are used. Some problems only show up in one epoch, in one datasets out of the hundreds tested. In this chapter these datasets are described, as well as the problems that occurred due to the exceptional situations. Also the method used to correct or by-pass the problem is given. In some of the cases the datasets are also selected to be kept for testing later on. Some of the problems are likely to surface again during the rest of the development. These saved tests can always be used by the developers to make sure the problems do not arise again at a later development stage.

D.1. Data gaps

The most frequently occurring problem in the datasets used are gaps in the data. When processing the data these gaps must not cause the software to stop working. However the quality of the results decreases with more missing epochs. For this reason datasets with small or large gaps are included when testing the functionality of the software. However for analysis of the quality of the results datasets with gaps are not taken into account as the decrease in performance cannot be controlled for these sets of data.

D.2. Faulty data

In some cases there are problems with observation files that cannot be correctly handled. The RINEX format defines the file format and still leaves quite some flexibility. However is some cases the information provided does not match the specified format. This is the case for the dataset from station *DOKD*. The following is an extract from the observation file.

```
 13 12 14  3  0  0.0000000  4  4
 13 12 14  3  0  0.0000000  0 14G02G04G05G10G12G13G17G23R01R02R08R22
                                R23R24
```

The value 4 (shown in red) is a flag indicating that the following lines do no contain data, but instead are comment lines. The second value 4 (indicated in blue) specifies the number of lines that are comments after this line. It is clear however that there are no comments. The software skips four lines and ends up with unexpected lines of data. As this flag is simply wrong there is reason to adapt the software to handle these kind of problems.

A second example of faulty data was found in a dataset from station *EUR2*. The order of observations is specified in the header of any RINEX file. This information is crucial for the user software. For this specific station the order of observation suddenly changed near the end of the day. While this does not cause the software to crash it is no longer possible to do positioning.

D.3. Receiver clock

The RINEX observation file provides the time of the observations. The time indicated is not directly used for positioning. The time can be indicated in milliseconds, but the timing requirements for positioning are way more strict than that. For this reason the receiver clock is estimated every epoch. The time indication in the RINEX file is different depending on receiver type and settings. Most datasets have time intervals that are exactly equal to the interval indicated in the header. In other cases the clock is allowed to drift. During the day a couple of milliseconds are added or subtracted from the indicated time. Either choice should be handled correctly by the software.

D.3.1. Station MAT1

The Italian IGS station *MAT1* has an exceptional way of handling the receiver clock drift. The time indicated in the RINEX file is not rounded to an integer value. During the day the pseudorange measurements continuously increase. At the start of the day the observations are in the range of 20.000 kilometers, which is expected with the orbit of the GPS satellites. Later on the day the pseudorange observations provide values of up to 80.000 kilometers. The user software indicated a shortage of satellites after the first couple of hours and therefore provided no position estimates for most of the day. The problem was an initial check on the observation data. A test was performed on the pseudorange observations for a satellite. If any of the observations indicated a range of more than 40.000 kilometers the satellite was discarded. This is a sensible first check on the observations, but does not take into account that this increase in pseudorange can be captured by the estimated receiver clock. By disabling the check on the pseudoranges the dataset could be processed. Now the problem with the receiver clock can be fully captured by the estimated parameter for the receiver clock.

D.3.2. Station GENO

The results for station GENO indicated that there were too few satellites for some epochs. The pattern was however very repetitive. Every whole hour the user software indicated that there were no satellites. The observation data did contain information but the satellites were all removed because the geometry-free DIA procedure, an initial rough check on the satellite data, indicated very large outliers. The problem here was again the time indicated in the observation file. During the day the clock drifted, adding a couple of milliseconds. At every hour the drift was corrected, do the indicated time was again an integer value. The geometry-free DIA cannot compensate for this change and therefore identifies this as a large outlier. With the geometry-free procedure disabled the change in the clock is again captured by the estimated receiver clock. Turning the geometry-free DIA procedure off is beneficial for this dataset, but for other datasets this procedure is still required. To allow for analysis of the quality of the results, the geometry-free DIA procedure is turned of for these specific datasets.

D.4. Loss of lock

Even though a RINEX observation file contains data for a specific interval, the receiver is actually tracking the satellites continuously. This is of importance for the phase measurements as only by continuous tracking the actual number of cycles difference between two epochs can be determined. Occasionally there may be outliers like cycle slips in the observations. These must be identified by outlier detection. If the satellite was not tracked continuously the receiver gives and indication of this. This is called 'Loss of Lock'. In two consecutive epochs there will be information for a satellite, yet the number of cycles indicated for the second epoch has no relation to the observation in the previous epoch. This change can also be seen as a cycle slip, however the size is significantly larger. The change is so large that it causes problems for the linearization in the filter. One way to capture these very large is using the geometry-free DIA procedure. This procedure detects the very large change for a single satellite between two epochs and thus removes the satellite for the second epoch. The next epoch the satellite rises again and is treated as a newly risen satellite, so new ambiguities are introduced.

This solution is not an optimal one. All observations from a single satellite are not used for a single epoch. The observations from these epochs are however not problematic, the link between the two epochs is. In most cases the loss of lock only occurs for a single satellite. If more than one, or most of the satellites, have the problem the geometry-free DIA procedure does not always yield good results. In figure D.1 the results for a station are shown. In this case the lock on all satellites is lost. Some satellites are removed, but not all. The result is that some ambiguities are kept, even though these are not correct. As seen in the figure the solution has trouble recovering from this problem.

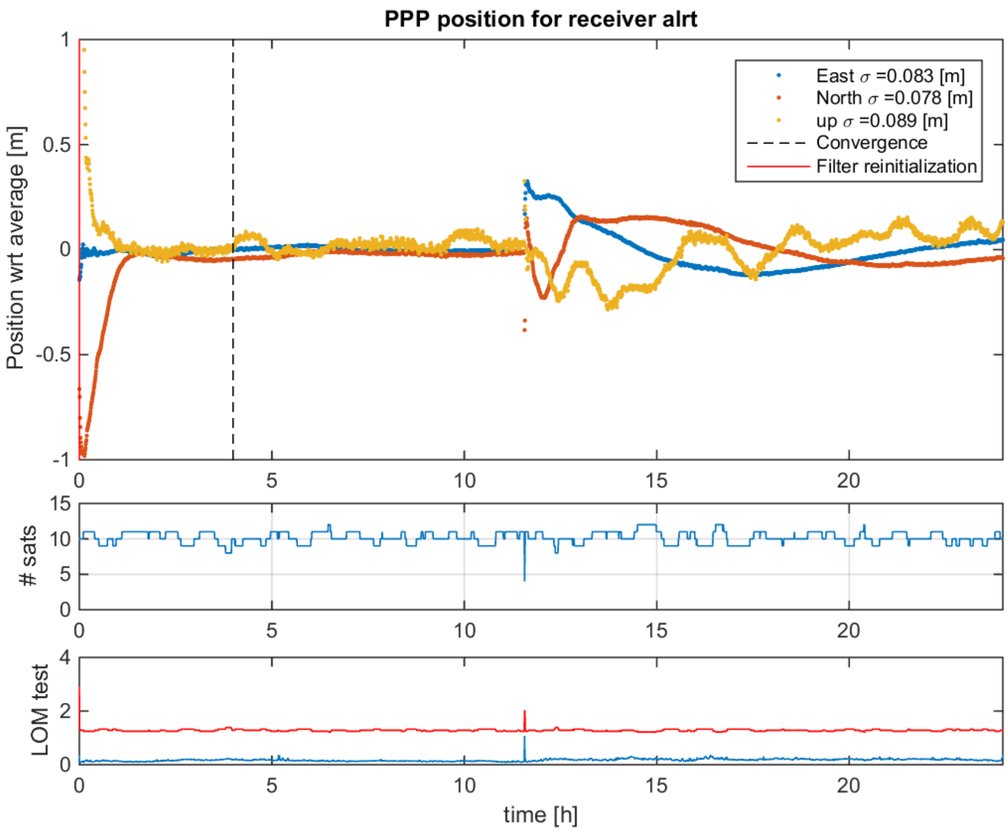


Figure D.1: Summary plot for station ALRT. Loss of lock on all satellites is not captured, resulting in an unstable solution.

The advantage for the loss of lock scenarios is that most of the time the receiver indicates a potential loss of lock. This is given as a flag in the RINEX file. Every epoch a list of satellites can be generated that have an indicated loss of lock since the previous epoch. An important consideration is that for some datasets a larger time interval is used than is actually present in the dataset. If the dataset has a 15 second interval, while the software requests 30 seconds, only every other epoch of information is used. The loss of lock indicators for the epochs in between should still be taken into account, even though the observations from these epochs are not used. If at any point between two epochs a loss of lock is encountered the software removes this satellite from the information of the previous epoch. This way the satellite is treated as a newly risen satellite, even though the satellite was used in the previous epoch. Compared to the geometry-free DIA method this allows the satellite to be used for an additional epoch. The results from the method are shown in figure D.2. There is still a problem in positioning, but since all ambiguities had to be reset and the positioning is purely kinematic this cannot be prevented. It is shows that after only a short period the results converge again, which was not the case when only the geometry-free DIA procedure was used.

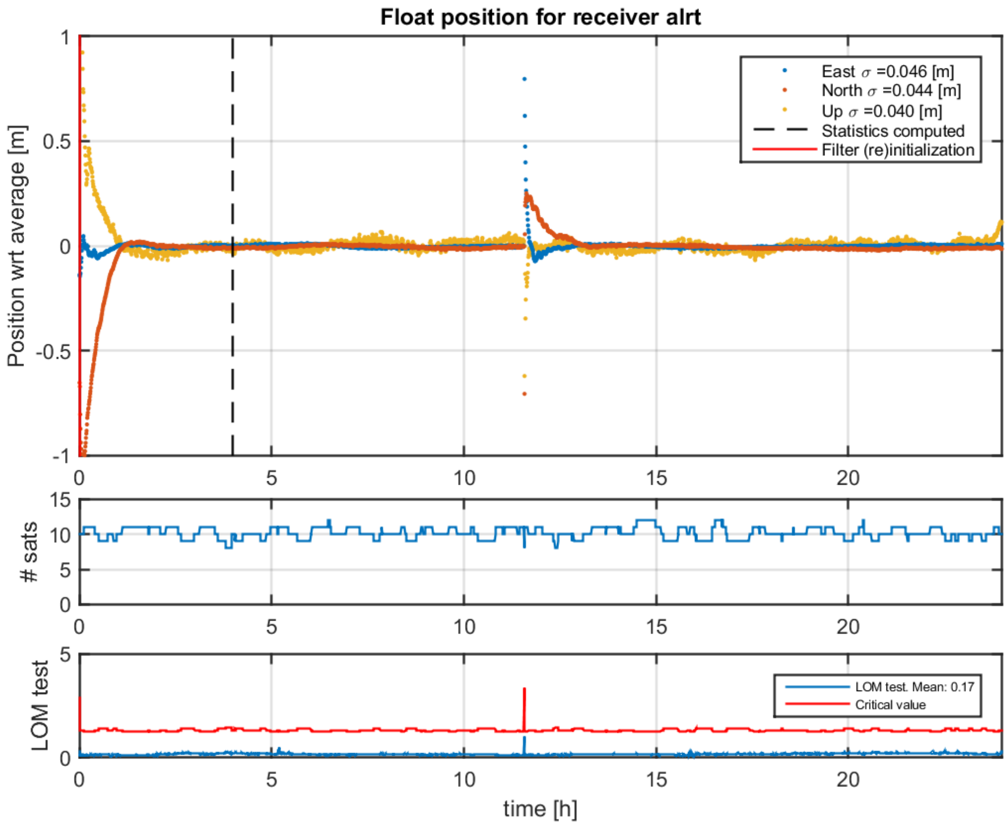


Figure D.2: Summary plot for station ALRT. Loss of lock on all satellites causes a short period where the solution converges again.

D.5. Pivot satellite

For both PPP and PPP-RTK one satellite per GNSS is selected as pivot in every epoch. This can be done in different ways. Every epoch a new satellite can be designated pivot, for instance the one with the highest elevation. Another option is to only change the pivot once the current pivot satellite is below the cut-off elevation or is no longer tracked. In both cases a change in pivot satellite required transformation of the phase ambiguities. In the Curtin user platform a pivot is chosen in the first epoch and kept for duration of the processing. Even when the pivot satellite sets the phase ambiguities all remain relative to the original pivot satellite.

During the testing of the datasets several scenarios were encountered that if not handled correctly caused large positioning errors. The first situation is the rising of a satellite that is already the pivot. Because the satellite has been out of view, either for a short of long period, the phase ambiguities from those previous epochs cannot be used. Even though it is physically the same satellite as the pivot it must be treated as a different newly rising satellite, with new values for the phase ambiguities. This is particularly a problem when a loss-of-lock is encountered on an active pivot satellite. In this case the satellite must immediately be treated as a new satellite. Keeping in mind that a loss-of-lock can easily result in change a in cycles for the phase observation of over 200,000. If this occurs for the pivot satellite this 'cycle-slip' enters into the ambiguities for all other satellites. As a result the DIA procedure has a problem. Normally a LOM test value of 2 might already be an indication of outliers. In this case this value has been shown to go up to 10^{13}.

It is also important to keep track of changing satellites even after the pivot satellite has set. If for some reason there are no common satellites between two epochs there is no longer a link to the initial pivot satellite. If this scenario occurs a new pivot satellite has to be selected. This hardly ever happens for GPS. However, for multi-GNSS scenarios this is quite common. As every GNSS uses its own pivot satellite and some of the constellations have only a few satellite those constellations have quite some epochs in which no satellite is in view. Positioning can continue, but keeping track of pivot satellites for each constellation must be done correctly.