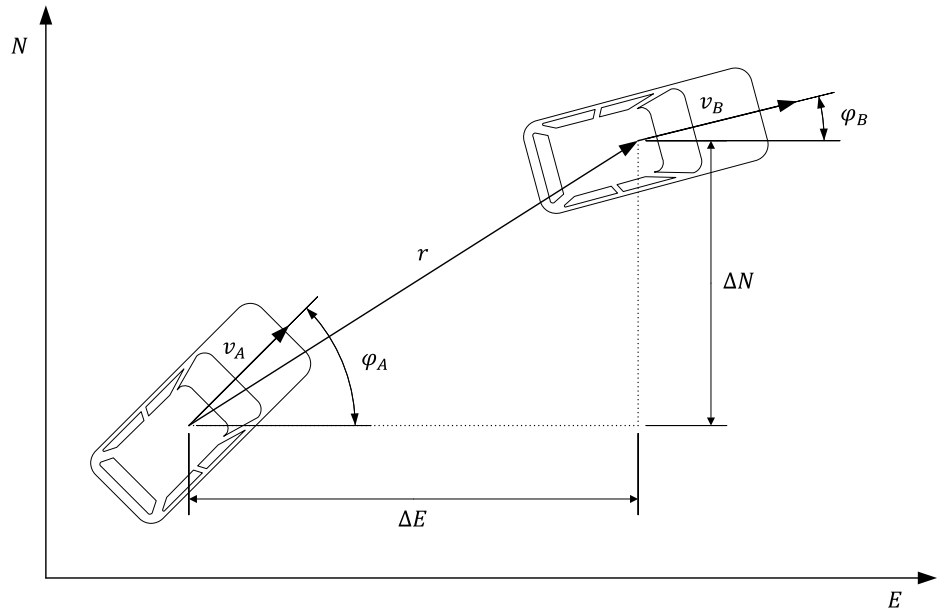
CHALMERS



Low cost relative GNSS positioning with IMU integration

Master's Thesis in Systems, Control and Mechatronics

Viktor Elisson

Gabriel Gässler

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2014
Master's Thesis EX030/2014

Master's Thesis EX030/2014

Low cost relative GNSS positioning with IMU integration

VIKTOR ELISSON
GABRIEL GÄSSLER

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2014

Low cost relative GNSS positioning with IMU integration
VIKTOR ELISSON
GABRIEL GÄSSLER

Master's Thesis EX030/2014
Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-722 1000

Cover:
Relative positioning of two vehicles with the same notations as applied in the report.

Chalmers University of Technology
Göteborg, Sweden 2014

Abstract

In recent years, automotive industry has put much effort into developing intelligent active safety systems to assist the driver. Many such systems depend on relative positioning of vehicles in close proximity. Currently laser, radar and camera sensors are used in mass production while Global Navigation Satellite Systems (GNSS) are often applied as a reference during verification. This thesis evaluates methods for low cost, single frequency GNSS relative positioning which can be applied in verification of active safety systems. Since GNSS sensors operating in the same area are affected by similar errors, the relative accuracy is higher than the absolute for short distances. One of the proposed methods consists of two separate loosely coupled GNSS and inertial measurement unit (IMU) filters for absolute positioning, sharing only the absolute position of each vehicle in order to calculate the distance between them. The second method utilizes a tightly coupled filter where GNSS raw data and IMU measurements of two vehicles are used in a single filter which estimates a relative baseline rather than absolute positions. Both proposed implementations are applying nonlinear Kalman filtering and smoothing. The tightly coupled approach provides superior results in most scenarios. Because the algorithms can be implemented without smoothing and the IMU sensors already exist in modern vehicles, the proposed low cost system can serve as a basis for a real time implementation to support active safety functions.

Keywords: GNSS, GPS, IMU, Relative positioning, RTK, Sensor fusion, Kalman filtering, Smoothing

Acknowledgments

We would like to thank Volvo Car Corporation for giving us the opportunity to carry out this thesis. Thanks to all members of the Driver Assistance and Active Safety Functions Analysis and Verification group for their support, especially our supervisor Hans Bäckström. Special acknowledgments go to Henrik Lind at Volvo Car Corporation for his valuable suggestions as well as our examiner Lennart Svensson for his guidance through the project and Jan Johansson for his advice and support in GNSS related topics, both at Chalmers University of Technology.

Gothenburg, May 2014
Viktor Elisson, Gabriel Gässler

Nomenclature

C/N_0 Carrier-To-Noise Density Ratio

A2R Absolute To Relative

CAN Controller Area Network

CDMA Code Division Multiple Access

DOF Degree Of Freedom

ECEF Earth-Centered Earth-Fixed

EKF Extended Kalman Filter

ENU East, North, Up

ERTS Extended Rauch-Tung-Striebel Smoother

FDMA Frequency Division Multiple Access

GNSS Global Navigation Satellite System

IGS International GNSS Service

IMU Inertial Measurement Unit

INS Inertial Navigation System

LAMBDA Least-squares AMBiguity Decorrelation Adjustment

NNSS Navy Navigation Satellite System

PPP Precise Point Positioning

RHCP Right Hand Circular Polarized

RINEX Receiver Independent Exchange Format

RMSE Root Mean Square Error

RTK Real Time Kinematic

RTS Rauch-Tung-Striebel Smoother

SD Standard Deviation

TAI International Atomic Time

UKF Unscented Kalman Filter

URTS Unscented Rauch-Tung-Striebel Smoother

UTC Coordinated Universal Time

UTM Universal Transverse Mercator

WGS World Geodedic System

CONTENTS

1 Introduction

Over the previous years, vehicles have faced a rapid change towards intelligent driver assistant systems such as electronic stability control, autonomous emergency braking, adaptive cruise control, and lane keep assist. As an example of the trend, Volvo Cars is running a pilot project, Drive Me, which in year 2017 aims in having autonomous cars driving inside Gothenburg. EURO NCAPs rating plan [18] states that active safety functions will have a growing impact on the rating. Therefore car manufacturers also strive for developing new safety functions as a sales argument.

Development of such technologies requires in-depth testing and for verifying the functionality and optimizations in automotive systems many test drives have to be performed. The data collected during these tests is often recorded and post-processed later on. For verification it is helpful to know the position of the test vehicles. In order to interpret the data it has to be treated differently if a vehicle is driving up- or downhill, doing curves, switching lanes, being in rural or urban areas. Often test vehicles drive together in close proximity to strengthen the data by undergoing the similar traffic circumstances. This is also useful for testing safety functions where several cars interact with each other. Therefore not only the absolute position of each vehicle is of importance, but even more the relative position. Laser, radar, and camera sensors are today used in active safety functions for measuring the relative distance while Global Navigation Satellite Systems (GNSS) are used as a reference during verification.

In industry, GNSS relative positioning of moving objects is most commonly used against a stationary reference which is called base station and knows its position very accurate. This allows very accurate absolute and relative positioning of moving objects in the area close to a base station. Systems providing such a functionality are very expensive and setting up a base station also takes time. Concerning that relative accuracy is often more important than absolute in active safety purposes there is a demand for low cost system for relative positioning between moving objects without a base station to the expense of high absolute accuracy.

Modern vehicles are already equipped with sensors such as odometer, gyroscopes, and accelerometers providing a better short term accuracy than GNSS, which has its strength in long term accuracy. A sensor fusion algorithm can be used to aid the GNSS with measures from these sensors at no additional cost.

This work focuses on methods to achieve good relative post-processed positions of two vehicles based on test drive records. The results can also be extended to real-time processing purposes such as the fast growing active safety functionalities.

1.1 Purpose

The purpose of this thesis is to develop an accurate positioning solution using a combination of GNSS and in-vehicle sensors with particular focus on relative positioning. The system should be affordable to be put in a large scale of test vehicles. Current systems available at the market suited for this purpose are very expensive.

1.2 Objective

The main restrictions are on cost and accuracy. The desired price per unit should be below 300 €. The relative accuracy to other cars has to be less than 0.5 m. It should also be possible to get a fair absolute accuracy, the limit here is 3 m. All accuracy limits are given horizontally.

1.3 Available data

The two main data sources utilized in this thesis are GNSS sensors and the Volvo test cars Controller Area Network (CAN) data. The used CAN data includes sensor measurements such as the yaw rate, the steering angle, and the estimated speed of the vehicle. Most of this data is provided with a frequency of 50 Hz. For verification purposes also the radar range to a target vehicle was taken into account. Reference drives are also done in comparison to the high cost, dual frequency receiver RT-Range, which also takes advantage of a high precision IMU and a base station.

The low cost GNSS sensors which are applied provide GNSS raw data access. This data consists of the range between satellite and receiver, the phase of the carrier signal, the Doppler shift, and the carrier-to-noise density ratio (C/N_0) for each observed satellite with 1 Hz update rate. In post processing also a third data source providing some corrections for GNSS data is employed.

Two different types of single frequency GNSS receivers are tested and compared. One of them being an u-blox LEA-6T receiver and the other a NVS NV08C-CSM receiver. Both can track GPS and geostationary navigation satellites, the NVS receiver is also capable of tracking GLONASS and Galileo, the Russian and European counterparts to GPS.

1.4 Thesis outline

Chapters 2 to 4 present the theory on which all the further chapters are based on. Chapter 2 focuses on GNSS and explains observable signals, error sources and positioning methods whereas the vehicle sensors are referred to in Chapter 3. The last theory chapter presents sensor fusion based on Kalman filtering and smoothing. Especially methods for fusing earlier described sensors are described. Chapter 5 presents the dynamical model and two proposed sensor fusion algorithms. First a loosely coupled filter for absolute positioning is developed. The second one is a tightly coupled filter based on previous work which is also presented here. In particular, this thesis contributes in investigating smoothing for both presented algorithms. The performance of both implementations in different driving scenarios such as country roads and freeways are demonstrated in Chapter 6. The absolute positioning algorithm is also tested in dead reckoning environments where smoothing is very beneficial. Chapter 7 and 8 summarize the thesis and give an outlook to possible future improvements.

2 Global Navigation Satellite System

GNSS can be tracked back till the 1960s. In this time US scientists discovered that they could track the Sputnik satellite by using the Doppler shift [13]. They were able to estimate the satellites orbit by observing the frequency changes of the transmitted signals. The possibility to track artificial objects in space by ground stations led to the idea to also track objects on the earth with the help of satellites in space. The first GNSS was the US Navy Navigation Satellite System (NNSS), also known as Transit which was later on replaced by GPS. Four years after the start of the first GPS prototype satellite in 1978 the Soviet Unions counterpart called GLONASS also started launching first satellites [13]. Other systems like the European Galileo or the Chinese BEIDOU/COMPASS were started decades later.

The basic principle behind all these systems is that the satellites transmit signals to receivers on earth based on which they calculate their position as shown in Figure 2.1. The transmitted data contains the current position of the satellite, the time when the data package was sent and some other data, which will be described later on.

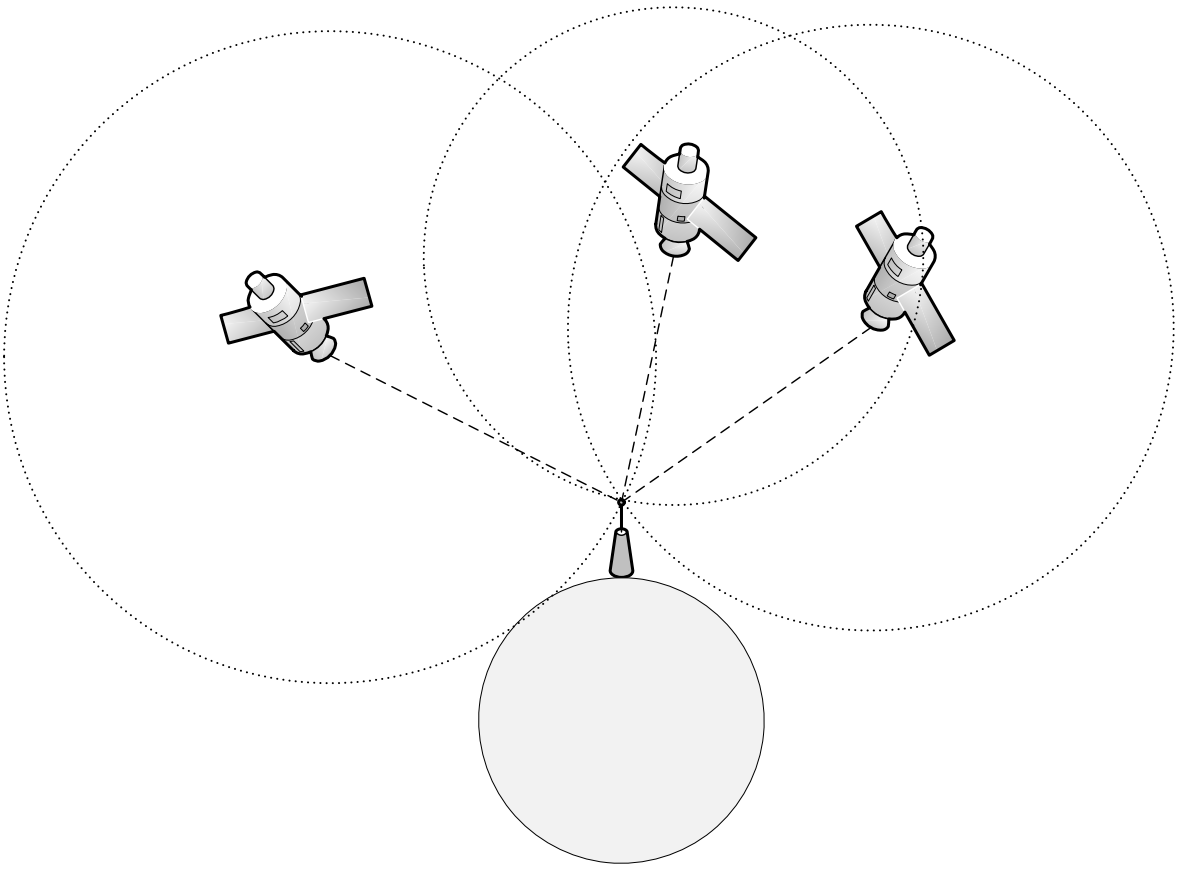


Figure 2.1. Principle of satellite positioning in 2D

In order to determine its position in 3D, a receiver needs at least data sent by four satellites at the same time. This is because it needs to solve for four unknowns, i.e. the position in three dimensions and the receiver time offset. The receivers own clock is much too inaccurate in order to use it for positioning since time errors lead to distance errors by multiplying them approximately with the speed of light. As a result, GNSS are very dependent on accurate timing, which leads to the issue that even though the satellites use atomic clocks, their time signals cannot be treated as perfect reference. This holds at least if a high accuracy position shall be obtained and will be discussed in Section 2.2.

The GPS system uses the L band frequencies 1575.42 MHz, 1227.6 MHz and recently also 1176.45 MHz for transmission of data [5]. They are called L1, L2, and L5. The single frequency receivers used for this thesis only support L1, therefore all further considerations hold for L1 if not stated differently.

Other GNSS make use of different frequencies and in case of GLONASS also other transmission technologies. GLONASS uses Frequency Division Multiple Access (FDMA) with each satellite using slightly different frequencies but is about to also send Code Division Multiple Access (CDMA) signals with all satellites of the system using the same frequencies. Then all GNSS will support CDMA which makes it easier to receive all of them within a single receiver [21].

2.1 Observables

A typical set of observables for a single channel GNSS receiver are the so-called pseudorange (P) to the satellite, the Doppler shift (D), the C/N_0 (S), and for more advanced receivers also the carrier phase measurements (L). The pseudorange is the time difference between the time when the data was sent by the satellite and when it was received multiplied by the speed of light. Utilizing the carrier signals phase as observable allows very precise positioning. The principle of carrier phase positioning is depicted in Figure 2.2.

The receiver is able to measure the phase φ_k of the carrier signal for each time epoch k and provides the carrier phase L_k as observable by utilizing the following equation

$$L_k = \lambda(\varphi_k + M) + L_0 \tag{2.1}$$

The carrier phase observable L_k at each time step is the sum of the current phase φ_k and all full cycles since the observation started M times the wavelength λ as well as of a constant term L_0. The initial term L_0 is set by the receiver and will be neglected from now on as done in [30]. The accumulated phase since the start of the measurement $\varphi_k + M$ will be called ϕ. Error terms are ignored in Equation (2.1), the most important of them are included in the more detailed Equation (2.3).

Since the carrier phase observable is only a relative measurement it is also needed

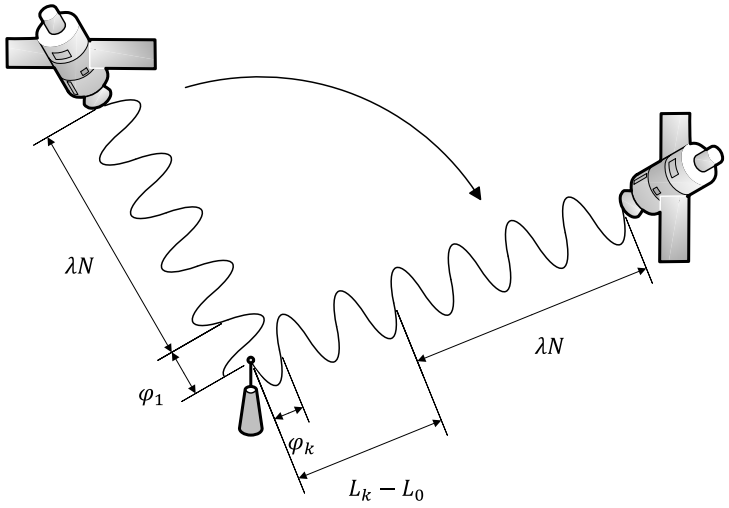


Figure 2.2. Carrier phase measurements

to calculate the initial number of full signal waves between sender and receiver. If this absolute reference, called phase ambiguity parameter (N), can be solved to its integer value, then in theory under absence of other errors, distance accuracies down to less than 1% of the wavelength can be achieved [5]. For example, L1 has a wavelength of approximately 20 cm, so distance accuracies of less than 2 mm are possible in ideal cases. GPS also transmits an encrypted military signal, but since this is not sent on L1 and therefore cannot be observed with the receivers used for this thesis, thus it will not be described in more detail.

Apart from these typical observables, the satellites also transmit data in their so-called navigation message. This includes information like the health status of the satellite, rough correction terms for their clock and orbit, elevation and azimuth angle. The need of correction terms is shown in the next section.

2.2 Error sources

The most important GNSS error sources are described in this section on the example of GPS. Other GNSS suffer mostly from the same errors. Table 2.1 provides an overview of the GPS errors for single frequency receivers and their quantitative 1σ impact on pseudorange measurements.

The resulting user equivalent range error (UERE) of 5.3 m leads to a position error of approximately 10 m for good measuring environments [14]. The far biggest error source, selective availability, is not mentioned in this table because it is not in effect anymore since May 2000. It was used by the US GPS authorities to artificially worsen the civil GPS signal by increasing the satellite clock 1σ error to approximately 20 m [14].

Significant error sources in the Space segment are satellite clock instabilities and

Segment Source	Error Source	GPS 1σ Error
Space	Ephemeris data	2.1 m
	Satellite clock	2.1 m
User	Ionospheric delay	4.0 m
	Tropospheric delay	0.7 m
	Multipath	1.4 m
	Receiver noise and resolution	0.5 m
Sum	User equivalent range error, RMS	5.3 m

Table 2.1. Error sources in GPS [14]

incorrect ephemeris data. Even though each of the satellites has several atomic clocks they are not perfectly accurate and also tend to drift away over longer time periods. As mentioned earlier, time errors can roughly be multiplied by the speed of light to get the resulting distance errors. To avoid even larger positioning errors due to clock drift and also orbit perturbations, the GPS ground segment transmits correction signals to the satellites. The error correction terms predicted by the ground segment for the satellites are not only used for correcting the signals sent by the satellites themselves, but are also partly included in the navigation message of the satellite to the receiver. These predictions improve the overall accuracy, but they still leave a comparably large error as presented in Table 2.1. The main reason for this is that the bytes reserved for correction data in the navigation message are very limited and the satellites can therefore not transmit the orbit correction terms at full accuracy. This is one of the major GPS design drawbacks which the upcoming Galileo system should be much less affected by due to more accurate navigation messages [5].

Due to the very high time accuracy requirements it is necessary to take relativistic effects into account. For example clocks in space run at a different rate than they would do on earth. In order to be observed with a rate of 10.23 MHz by the receivers, satellite clocks run at 10.22999999543 MHz [15]. The user does not have to take care of this potential error source, however there are other relativistic error sources which have to be taken into account in order to achieve a high accuracy. The biggest of these error sources is the slight eccentricity of the satellite orbit. This does not only cause varying distances from the satellites to the earths gravitation center but also causes changes in the satellite clock frequency, depending on this distance. According to [8] this can cause an pseudorange error up to a maximum of 21 m, which equals to a time error of 70 ns.

Other important error sources are atmospheric effects on the signal propagation by adding delay. In the upper atmosphere, in GNSS context often referred to as ionosphere, the signals get disturbed by loaded particles. This is mostly due to solar radiation and therefore this error is much larger during daytime than nighttime. In the lower atmosphere, referred to as troposphere, the weather also influences the signal propagation. The major error here is caused by water vapor. There is no clear border between both atmospheric segments and they can even overlap.

These ionospheric and tropospheric definitions do not exactly match the geophysical definitions but they were chosen anyway since they are most commonly used in GNSS context [5].

Errors are also introduced by receiver noise and resolution limits as well as multipath. Both given reference values hold for high cost receivers and a good measurement environment, which does not apply to most of the data collected for this thesis. Multipath is caused by signal reflections on buildings and other obstacles. Figure 2.3 shows a scenario in which the direct signal is shadowed by a tree and the antenna only receives reflected signals. Most antenna designs include two methods in order

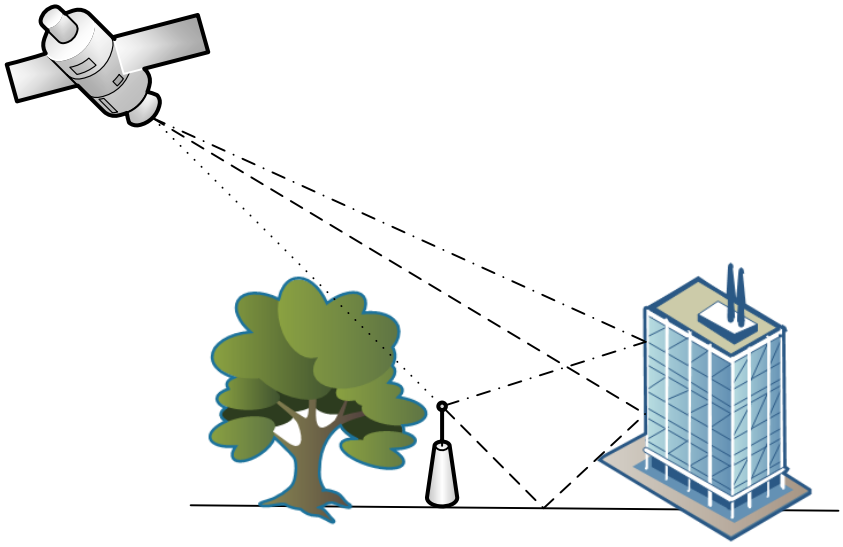


Figure 2.3. Multipath

to reduce multipath effects [5]. One of these is to shape the gain pattern for reducing the gain on low elevation signals. This is based on the fact that most of the multipath signals arrive from angles near and below the horizon as shown in Figure 2.3. The second method takes advantage of the right hand circular polarization (RHCP) of the GPS signal. Reflections on smooth surfaces change the signals polarization. So in theory an ideal GPS antenna only takes into account signals which are reflected an even amount of times. But reflections on rough surfaces cause randomly polarized signals. And in addition antennas are not ideal and cannot obtain total rejection of non-RHCP signals, especially low-price single feed antennas provide poor rejection characteristics. A typical 3 dB attenuation for multipath signals can be gained by taking advantage of the polarization [16]. Multipath errors can be reduced further by mounting the antenna on radio frequency absorbing material.

If the direct signal is not shadowed and not weaker than multipath signals the carrier-phase errors cannot exceed 90° which corresponds to 5 cm. However, pseudorange based errors caused by multipath can be larger than 100 m [16].

It is possible to estimate the multipath effect on all available satellite signals at each time epoch. One approach is the inclusion of omnidirectional (infrared) cameras,

which can be used to detect obstacles and is proposed in [10]. This enables simulations determining if the calculated signal path is blocked by obstacles. But since omnidirectional cameras are no standard configuration in cars, this approach is not employable for low cost solutions. Another strategy takes advantage of available map and elevation data and to fuse them into a 3D model instead of using a camera as presented in [12]. This strategy can be very beneficial in static urban areas but it heavily depends on reliable elevation data. It is not applicable on areas where no elevation data is available or the data is outdated, for example due to new buildings. This strategy also provides no help if the multipath is caused by trees, which is one of the main multipath sources in the test drives done for this thesis.

Some of the described error sources like space segment errors and receiver limits can be considered to be largely time and space invariant noise. As described previously, the ionospheric delay heavily depends on the daytime. This also holds for the tropospheric delay, but on a shorter time scale. Multipath is also to an extent time variant since the satellite constellation changes over time and along with that the shadowed and non-shadowed paths between satellite and receiver. But on average multipath is much more space variant since open areas should cause nearly no multipath error while in obstructed areas even some centimeters difference in position can decide upon a signal being shadowed or not. The tropospheric delay can be treated as constant in close proximity whereas this assumption hold for ionospheric delay even over large areas. For relative positioning, especially space variant errors are important since time variant errors can be mostly canceled out.

2.3 Measurement model and positioning methods

A basic model for the most important error sources in pseudorange is given by

$$P_k^p = \rho_k^p + (\Delta t_k - \Delta t^p)c + I_k^p + A_k^p \tag{2.2}$$

and for carrier phase measurements by

$$L_k^p = \lambda\phi_k^p = \rho_k^p + (\Delta t_k - \Delta t^p)c - I_k^p + A_k^p + \lambda N_k^p \tag{2.3}$$

Derivations for Equations (2.2) and (2.3) are presented in [30]. In this model P_k^p represents the measured pseudorange, ϕ_k^p the measured accumulated carrier phase, Δt_k is the error in the receiver clock, Δt^p the error in the satellite clock, I_k^p and A_k^p ionospheric and tropospheric errors, λ the wavelength of the signal, N_k^p the integer ambiguity parameter, c the speed of light in vacuum, and ρ_k^p the true distance with p being a index for satellites and k for receivers. The model is given noiseless, of course the observations also include other additive noise terms like multipath or receiver noise. This model is the basis of most of the positioning methods.

The measured pseudoranges can be used as follows in order to calculate the absolute

position of the receiver [8]

$$\begin{aligned} P^1 &= \sqrt{(x^1 - x)^2 + (y^1 - y)^2 + (z^1 - z)^2} + (t_k - t^1)c \\ P^2 &= \sqrt{(x^2 - x)^2 + (y^2 - y)^2 + (z^2 - z)^2} + (t_k - t^2)c \\ P^3 &= \sqrt{(x^3 - x)^2 + (y^3 - y)^2 + (z^3 - z)^2} + (t_k - t^3)c \\ P^4 &= \sqrt{(x^4 - x)^2 + (y^4 - y)^2 + (z^4 - z)^2} + (t_k - t^4)c \end{aligned} \tag{2.4}$$

The terms x^p, y^p, z^p, and t^p are stated in the satellites navigation messages, so there are as many equations as unknowns ($x, y, z,$ and t_k) left. These equations can be solved using e.g., least squares or an Extended Kalman filter. Introducing more satellites to the system of equations enables more accurate estimation of the position.

The following sections describe methods to remove or reduce the mentioned error terms.

2.3.1 Real Time Kinematic

Real Time Kinematic (RTK) is a positioning method based on carrier phase measurements and a single reference station. A GSM modem can be used to connect a GNSS RTK receiver to the reference station. Reference stations know their own position very accurate due to long term measurements and also calculate their position based on current GNSS measurements. With this information a GNSS receiver can calculate the distance vector, called baseline, to a reference stations position based on current GNSS measurements. This baseline is then added to the precise position of the reference station to determine the receivers position. This method cancels out common errors, which receiver and reference station are both affected by, and therefore improves the position accuracy. The downside of RTK is that it only works well if receiver and reference station are located close to each other. This is required because the share of common errors in the total error is decreasing with increasing distance. As a rule of thumb the distance should not be longer than 20 km [5]. Alongside with RTK, differencing methods are often used. The two most common out of these are also used in this work and hence described below.

Single Differencing

The singe differencing method can be applied if two ground stations A and B both receive signals from the same single satellite p as shown in Figure 2.4. Then based on Equations (2.2) and (2.3), the satellite clock error Δt^p can be canceled out in pseudorange and carrier phase respectively [24]

$$\begin{aligned} \Delta P^p_{AB} &= P^p_A - P^p_B \\ &= \Delta\rho^p_{AB} + \Delta t_{AB}c + \Delta I^p_{AB} + \Delta A^p_{AB} \end{aligned} \tag{2.5}$$

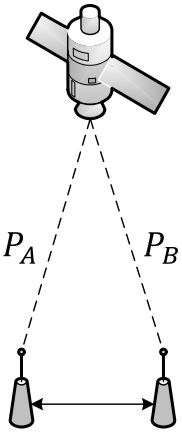


Figure 2.4. Single Differencing

$$\begin{aligned}\Delta L^p_{AB} &= L^p_A - L^p_B \\ &= \Delta\rho^p_{AB} + \Delta t_{AB}c - \Delta I^p_{AB} + \Delta A^p_{AB} + \lambda\Delta N_{AB}\end{aligned} \tag{2.6}$$

If both ground stations are closely located to each other, the terms ΔI^p_{AB} and ΔA^p_{AB} are significantly smaller than I^p_{AB} and A^p_{AB}.

Double Differencing

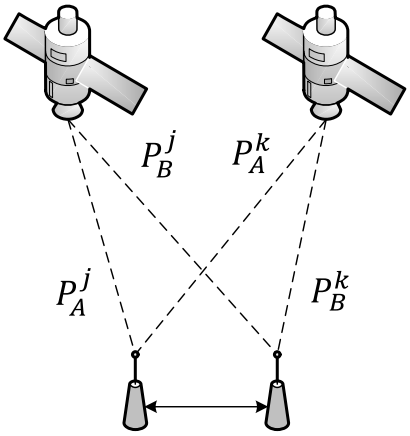


Figure 2.5. Double Differencing

A pair of single differences as given in Equations (2.5) and (2.6) can be differenced further in order to cancel out satellite and receiver clock errors of two satellites j and k and two stations A and B as illustrated in Figure 2.5. This forms the double differences [24]

$$\begin{aligned}\nabla\Delta P^{jk}_{AB} &= \Delta P^j_{AB} - \Delta P^k_{AB} \\ &= \nabla\Delta\rho^{jk}_{AB} + \nabla\Delta I^{jk}_{AB} + \nabla\Delta A^{jk}_{AB}\end{aligned} \tag{2.7}$$

$$\begin{aligned}\nabla\Delta L_{AB}^{jk} &= \Delta L_{AB}^{j} - \Delta L_{AB}^{k} \\ &= \nabla\Delta\rho_{AB}^{jk} - \nabla\Delta I_{AB}^{jk} + \nabla\Delta A_{AB}^{jk} + \nabla\Delta\lambda N_{AB}^{jk}\end{aligned} \tag{2.8}$$

In double differencing often one satellite is chosen as reference and the differences are calculated between this satellite and all the others. Single and double differences can also be used in a different order or between time epochs. Sometimes triple differences are applied by differencing Equations (2.7) and (2.8) between two time epochs.

2.3.2 Network Real Time Kinematic

Network RTK eliminates the major downside of RTK by using a network of reference stations. There are many of these networks worldwide, mostly covering one specific country. One example is the SWEPOS network for Sweden, operating about 300 reference stations in early 2014. By using network based solutions like SWEPOS the costumers do not have to find a suitable reference station on their own and it is even possible to get accurate corrections if there is no reference station nearby. This is possible because network based solutions can interpolate the corrections terms between several stations to the approximate position of the costumer [5].

2.3.3 Precise Point Positioning

Precise Point Positioning (PPP) does not take advantage of reference stations, but of precise absolute correction terms [5]. These correction terms can include satellite clock corrections, orbit corrections, ionospheric and tropospheric correction and are provided by organizations such as the International GNSS Service (IGS)[2]. These correction terms are calculated using data provided by hundreds of stations worldwide. The big difference between RTK and PPP is that RTK uses the knowledge of how the errors affect positions nearby and does corrections relative to these stations while PPP calculates the position using absolute error terms. A major upside of PPP compared to RTK is that it does not require reference stations nearby or reference networks covering the requested position. The downsides are that PPP requires much more sophisticated error modeling in order to take the correction terms into account and that it is very difficult to use PPP in real time since data from from all over the world on stations operated by a plenty of different authorities is required. Nevertheless IGS started a PPP real time service in 2012 with the intention to make in public available, but this has not happened by early 2014. However everyone can apply for free access to this service.

2.4 Reference systems

The application of GNSS requires standardized time and coordinate reference systems. The most important of these are described in this section.

2.4.1 Time reference systems

The most commonly known time reference system is Coordinated Universal Time (UTC), on which the timezones used all over the world are based. But there also exist other time reference systems like the International Atomic Time (TAI), which is calculated as the weighted average on hundreds of atomic clocks worldwide. The time difference between UTC and TAI is the total amount of leap seconds plus the initial offset of 10 seconds between both systems before leap-seconds were introduced in 1972. As of early 2014, UTC is 35 seconds behind TAI since at this time a total amount of 25 leap seconds were introduced. The GPS time is in between UTC and TAI. It does not take leap seconds into account and was synchronized with UTC on the 6th of January 1980 [5]. This point in time also marks the start of GPS time. Since the difference between UTC and TAI was 19 seconds at this date, GPS time will always be 19 seconds behind TAI. So by early 2014 the difference between GPS time and UTC is 16 seconds. This has to be taken care of when analyzing GPS observations, which are always given in GPS time. Each GNSS uses its own time scale but for processing, times are often given in GPS time.

2.4.2 Coordinate systems

In order to calculate positions on the earth's surface, the position has to be somehow transformed from a celestial to a terrestrial coordinate system. A often applied geodetic system is called World Geodedic System 84 (WGS 84), which is used by GPS [17]. This system is used by GPS and takes the earth eccentricity into account. However, for some calculations it is sufficient to use a simpler terrestrial coordinate system such as the earth-centered earth-fixed (ECEF) or local east, north, up (ENU) coordinates. The latter describes a position projection to a plane which is tangential to the earth in a reference point. This is valid for short distances on the earth surface, but introduces errors, which grow rapidly with increasing distance to the reference point. A often used projected coordinate system based on this principle is the Universal Transverse Mercator (UTM) system. It divides the earth up into sixty reference points and their surrounding areas [5]. Since these zones are fixed this can introduce huge errors when calculating distances between points belonging to different zones. This method was not used in this work since all of the measurements were taken in Gothenburg or close to it and a UTM zone border splits the city up into two zones. That can cause problems using UTM, even though the zones overlap by one degree of longitude.

2.5 RTKLIB

RTKLIB [25] is an open source GNSS computing toolbox. It contains tools for logging, converting, plotting, streaming, downloading, and processing GNSS data. It supports several positioning algorithms, amongst which are also PPP and RTK.

In this work RTKLIB is used for logging and converting the logs from proprietary receiver formats into the standardized Receiver Independent Exchange Format (RINEX) as well as for post-processing. For the latter correction data from IGS for satellite clocks, orbit errors and ionospheric delay is deployed.

3 Vehicle sensors

Modern vehicles are equipped with several sensors aiding functions like electronic stability control. These sensors can also be utilized for navigation purposes.

3.1 Inertial Measurement Unit

An Inertial Measurement Unit (IMU) contains gyroscopes, accelerometers, and the necessary electronics to function the sensors. It often contains a microprocessor in order to compensate for some bias terms in the sensors [27]. Common IMUs usually have either three or six degrees of freedom (DOF). A three DOF IMU consists of a one-axis gyroscope and a two-axis accelerometer while in a six DOF IMU both sensors have three axis.

3.1.1 Gyroscope

Gyroscopes are normally used for measuring angular velocity. Traditionally this has been done by measuring the inertial properties of a spinning mass. Nowadays Micro-Machined Electromechanical Systems (MEMS) gyroscopes are used for automotive purposes because of their low price, small form factor, and high reliability. They are based on exploiting the Coriolis effect of a vibrating mass [27].

3.1.2 Accelerometer

As the name indicates accelerometers measure acceleration, this also includes gravitational forces. Traditionally accelerometers have been a spring-damper-mass system where the acceleration is determined by the displacement of the mass. MEMS accelerometers are now the most common in the automotive industry [27].

3.2 Odometer

The odometer measures the traveled distance by counting the number of wheel ticks. In automotive applications this is done by having a toothed wheel mounted to the rotating shaft. A hall sensor detects by variations in the magnetic field each time the tooth passes [3]. The vehicle speed can then be calculated by knowing the wheel radius and the number of counted wheel ticks during a certain period of time.

4 Sensor fusion

Sensor fusion is a term used to describe methods that combines data from different sensors to achieve a more accurate and reliable estimate of a quantity than one sensor alone could provide [3]. The sensors do not have to measure the same physical quantity but can be related in a model of the observed system. Having prior information of how the states are connected in a state space model and the noise, different versions of the Kalman filter can be applied for sensor fusion.

4.1 State space modeling

To form a framework for the Kalman filter, state space modeling needs to be explained. In a state space representation, a mathematical model of differential equations describing a physical systems inputs, states, and outputs is used.

4.1.1 Continuous time model

A general form of a state space model is described as follows

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \tilde{\mathbf{f}}(\mathbf{x}(t), \mathbf{u}(t)) + \mathbf{w}(t) \\ \mathbf{y}(t) &= \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) + \mathbf{v}(t) \end{aligned} \tag{4.1}$$

where $\mathbf{x}$ is the state vector, $\mathbf{u}$ is the input vector, $\mathbf{y}$ is the measurement vector, $\mathbf{w}$ is the process noise vector, and $\mathbf{v}$ is the measurement noise vector. The process model $\mathbf{f}$ contains first order differential equations relating the states and inputs. The measurement model $\mathbf{h}$ relates the measurements to the states.

For the linear case, Equation (4.1) can be written in matrix form as

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \tilde{\mathbf{A}}\mathbf{x}(t) + \tilde{\mathbf{B}}\mathbf{u}(t) + \mathbf{w}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{v}(t) \end{aligned} \tag{4.2}$$

4.1.2 Discrete time model

In order to handle the state space model from Equations (4.1) and (4.2) in a computer, the equations need to be discretized. This can be done by approximating the derivatives with a finite difference approximation as

$$\dot{x}(t) \approx \frac{x_{k+1} - x_k}{T} \tag{4.3}$$

where T is the sampling time.

Discretizing the continuous nonlinear model from Equation (4.1) gives

$$\begin{aligned}\mathbf{x}_k &= \mathbf{x}_{k-1} + T\tilde{\mathbf{f}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + T\mathbf{w}_{k-1} \\ &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + T\mathbf{w}_{k-1} \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k\end{aligned} \tag{4.4}$$

and the discretized linear model from Equation (4.2) becomes

$$\begin{aligned}\mathbf{x}_k &= (\mathbf{I} + T\tilde{\mathbf{A}})\mathbf{x}_{k-1} + T\tilde{\mathbf{B}}\mathbf{u}_{k-1} + T\mathbf{w}_{k-1} \\ &= \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + T\mathbf{w}_{k-1} \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k\end{aligned} \tag{4.5}$$

4.2 Kalman filter

The widely used Kalman filter was derived around 1960 [7] and is the optimal estimator under certain assumptions on the data.

Given the state space in Equation (4.5) with gaussian distributed noise, $\mathbf{w} \sim N(\mathbf{0}, \mathbf{Q})$ and $\mathbf{v} \sim N(\mathbf{0}, \mathbf{R})$ and assuming that it is conditional according to

$$\begin{aligned}\mathbf{x}_k &\sim p(\mathbf{x}_k|\mathbf{x}_{k-1}) \\ \mathbf{y}_k &\sim p(\mathbf{y}_k|\mathbf{x}_k)\end{aligned} \tag{4.6}$$

Equation (4.6) is assumed to be Markovian.

The Kalman filter applies Bayes rule

$$p(\mathbf{x}_k|\mathbf{y}_k) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k)}{p(\mathbf{y}_k)} \tag{4.7}$$

Using Equation (4.7) the derivations ends up in the two-step recursive equations
Prediction

$$\begin{aligned}\mathbf{x}_k^- &= \mathbf{A}_{k-1}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} \\ \mathbf{P}_k^- &= \mathbf{A}_{k-1}\mathbf{P}_{k-1}\mathbf{A}_{k-1}^{\mathrm{T}} + \mathbf{Q}_{k-1}\end{aligned} \tag{4.8}$$

Update

$$\begin{aligned}\mathbf{K}_k &= \mathbf{P}_k^-\mathbf{C}_k^{\mathrm{T}}[\mathbf{C}_k\mathbf{P}_k^-\mathbf{C}_k + \mathbf{R}_k]^{-1} \\ \mathbf{x}_k &= \mathbf{x}_k^- + \mathbf{K}_k(\mathbf{y} - \mathbf{C}_k\mathbf{x}_k^-) \\ \mathbf{P}_k &= \mathbf{P}_k^- - \mathbf{K}_k\mathbf{C}_k\mathbf{P}_k^-\end{aligned} \tag{4.9}$$

The prediction step incorporates prior information about the state which consists of the posterior densities from the previous time step and the dynamical model. In the update step also the measurement model, connecting the states to the observed quantities, is taken into account and the posterior densities for the current time step are calculated.

Since this filter is limited to the linear case and the process model for a car is nonlinear, there is a need for nonlinear extensions to the Kalman filter. The drawback is that optimality cannot be guaranteed with the nonlinear versions.

4.2.1 Extended Kalman filter

The Extended Kalman filter (EKF) is the most well used estimator for nonlinear systems. The EKF starts from the more general state space model in Equation (4.4) with Gaussian distributed noise assumption and linearizes the model in each time step with a Taylor expansion. This report treats the first order expansions but it is possible to apply higher order approximations. The equations are

Prediction

$$\begin{aligned} \mathbf{x}_k^- &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \\ \mathbf{P}_k^- &= \mathbf{F}_\mathbf{x}(\mathbf{x}_{k-1})\mathbf{P}_{k-1}\mathbf{F}_\mathbf{x}^\mathrm{T}(\mathbf{x}_{k-1}) + \mathbf{Q}_{k-1} \end{aligned} \tag{4.10}$$

Update

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k^-\mathbf{H}_\mathbf{x}^\mathrm{T}(\mathbf{x}_k^-)[\mathbf{H}_\mathbf{x}(\mathbf{x}_k^-)\mathbf{P}_k^-\mathbf{H}_\mathbf{x}^\mathrm{T}(\mathbf{x}_k^-) + \mathbf{R}_k]^{-1} \\ \mathbf{x}_k &= \mathbf{x}_k^- + \mathbf{K}_k[\mathbf{y}_k - \mathbf{H}_\mathbf{x}(\mathbf{x}_k^-)] \\ \mathbf{P}_k &= \mathbf{P}_k^- - \mathbf{K}_k\mathbf{H}_\mathbf{x}(\mathbf{x}_k^-)\mathbf{P}_k^- \end{aligned} \tag{4.11}$$

with the partial derivatives

$$\mathbf{F}_\mathbf{x} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}|_{x=x_k, u=u_k} \tag{4.12}$$

$$\mathbf{H}_\mathbf{x} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}|_{x=x_k, u=u_k} \tag{4.13}$$

4.2.2 Unscented Kalman filter

In 1997 the Unscented Kalman Filter (UKF) was proposed as an alternative to the EKF [6]. The filter is based on function evaluations rather than linearization like the EKF. This is numerically more stable since derivatives might contain division by zero and they can also be hard or even impossible to calculate analytically. The UKF is also able to handle higher order nonlinearities more accurately than the EKF [6].

The UKF is based on the Unscented transform which it uses to approximate the mean and covariance of the transformed variable instead of approximating the nonlinear function as the EKF tries to do. The Unscented transform generates a fixed number of sigma points around a variable x with the same mean and covariance as x. The sigma points are propagated through the nonlinear model and the mean and covariance of the transformed variable x are then estimated.

After initializing $\mathbf{P}_0$ and $\mathbf{x}_0$, the filter algorithm for the model in Equation (4.4) as described by [23] is

Prediction
Generate the sigma points

$$\begin{aligned}\mathcal{X}_{k-1}^{(0)} &= \mathbf{x}_{k-1} \\ \mathcal{X}_{k-1}^{(i)} &= \mathbf{x}_{k-1} + \sqrt{n+\lambda} + \left[\sqrt{\mathbf{P}_{k-1}}\right]_i, \\ \mathcal{X}_{k-1}^{(i+n)} &= \mathbf{x}_{k-1} - \sqrt{n+\lambda} + \left[\sqrt{\mathbf{P}_{k-1}}\right]_i, \quad i = 1, ..., n\end{aligned} \tag{4.14}$$

with

$$\begin{aligned}\lambda &= \alpha^2(n+\kappa) - n \\ \sqrt{\mathbf{P}}\sqrt{\mathbf{P}}^{\mathrm{T}} &= \mathbf{P}\end{aligned} \tag{4.15}$$

Compute the process model transformation of the sigma points

$$\mathcal{X}_k^{(i)} = \mathbf{f}(\mathcal{X}_{k-1}^{(i)}), \quad i = 0, ..., 2n \tag{4.16}$$

Compute the predicted mean and covariance of the transformed sigma points

$$\begin{aligned}\mathbf{x}_k^- &= \sum_{i=0}^{2n} W_i^{(m)} \hat{\mathcal{X}}_k^{(i)}, \\ \mathbf{P}_k^- &= \sum_{i=0}^{2n} W_i^{(c)} (\hat{\mathcal{X}}_k^{(i)} - \mathbf{x}_k^-)(\hat{\mathcal{X}}_k^{(i)} - \mathbf{x}_k^-)^{\mathrm{T}} + Q_{k-1}\end{aligned} \tag{4.17}$$

with the weights

$$\begin{aligned}W_0^{(m)} &= \frac{\lambda}{n+\lambda} \\ W_0^{(c)} &= \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta) \\ W_i^{(m)} &= \frac{1}{2(n+\lambda)}, \quad i = 1, ..., 2n \\ W_i^{(c)} &= \frac{1}{2(n+\lambda)}, \quad i = 1, ..., 2n\end{aligned} \tag{4.18}$$

Update
Generate the sigma points

$$\begin{aligned}\mathcal{X}_k^{-(0)} &= \mathbf{x}_k^- \\ \mathcal{X}_k^{-(i)} &= \mathbf{x}_k^- + \sqrt{n+\lambda} + \left[\sqrt{\mathbf{P}_k^-}\right]_i, \\ \mathcal{X}_k^{-(i+n)} &= \mathbf{x}_k^- - \sqrt{n+\lambda} + \left[\sqrt{\mathbf{P}_k^-}\right]_i, \quad i = 1, ..., n\end{aligned} \tag{4.19}$$

Compute the measurement model transformation of the sigma points

$$\hat{\mathcal{Y}}_k^{(i)} = \mathbf{h}(\mathcal{X}_k^{-(i)}), \quad i = 0, ..., 2n \tag{4.20}$$

Compute the predicted mean $\boldsymbol{\mu}_k$, the predicted covariance $\mathbf{S}_k$, and the predicted cross-covariance $\mathbf{C}_k$ between the state and the measurement.

$$\begin{aligned}\boldsymbol{\mu}_k &= \sum_{i=0}^{2n} W_i^{(m)} \hat{\mathcal{Y}}_k^{(i)} \\ \mathbf{S}_k &= \sum_{i=0}^{2n} W_i^{(c)} (\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)(\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)^T + R_k \\ \mathbf{C}_k &= \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{X}_k^{-(i)} - \mathbf{x}_k)(\hat{\mathcal{Y}}_k^{(i)} - \boldsymbol{\mu}_k)^T\end{aligned} \tag{4.21}$$

The filter equations similar to Equation (4.9) are described as

$$\begin{aligned}\mathbf{K}_k &= \mathbf{C}_k \mathbf{S}_k^{-1} \\ \mathbf{x}_k &= \mathbf{x}_k^- + \mathbf{K}_k(\mathbf{y}_k - \boldsymbol{\mu}_k) \\ \mathbf{P}_k &= \mathbf{P}_k^- - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T\end{aligned} \tag{4.22}$$

The algorithm runs Equations (4.14) - (4.22) as new measurements arrive. According to [29] the tuning parameter α controls the spread of sigma points (usual choice 10^{-3}), κ is a scaling parameter which is usually 0, and β describes prior knowledge about the distribution of the state. For Gaussian distributions $\beta = 2$.

4.3 Rauch-Tung-Striebel smoother

Filters cannot only be run forward in time, but also backwards, then they are called smoothers. This can be especially helpful for post-processing purposes where not only the past, but also the future of the measurements is known. The smoother equivalent to the Kalman filter is called Rauch-Tung-Striebel Smoother (RTS) and was presented by H. E. Rauch, F. Tung, and C. T. Striebel in 1965 [20]. While the Kalman filter output is conditional on the current and past measurements, the RTS is conditional on the whole measurement set.

The backward recursive equations for the linear RTS are

$$\begin{aligned}\mathbf{x}_{k+1}^- &= \mathbf{A}_k \mathbf{x}_k + \mathbf{B}\mathbf{u}_k \\ \mathbf{P}_{\mathbf{k+1}}^- &= \mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^{\mathrm{T}} + \mathbf{Q}_k \\ \mathbf{G}_k &= \mathbf{P}_k \mathbf{A}_k^{\mathrm{T}} (\mathbf{P}_{k+1}^-)^{-1} \\ \mathbf{x}_k^s &= \mathbf{x}_k + \mathbf{G}_k(\mathbf{x}_{k+1}^s - \mathbf{x}_{k+1}^-) \\ \mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k(\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-)\mathbf{G}_k^{\mathrm{T}}\end{aligned} \tag{4.23}$$

The first two equations are equivalent to the Kalman filter prediction Equations (4.8). The state $\mathbf{x}_k$ and covariance $\mathbf{P}_k$ are taken from the corresponding time steps in the Kalman filter. A proof of these equations is shown in [23].

4.3.1 Extended Rauch-Tung-Striebel smoother

The Rauch-Tung-Striebel smoother can also be formulated for the extended Kalman filter. Referring to [23] for the derivations, the equations for the first order extended Rauch-Tung-Striebel smoother (ERTS) are

$$\begin{aligned}
\mathbf{x}_{k+1}^{-} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\
\mathbf{P}_{k+1}^{-} &= \mathbf{F}_{\mathbf{x}}(\mathbf{x}_k)\mathbf{P}_k{\mathbf{F}_{\mathbf{x}}}^{\mathrm{T}}(\mathbf{x}_k) + \mathbf{Q}_k \\
\mathbf{G}_k &= \mathbf{P}_k{\mathbf{F}_{\mathbf{x}}}^{\mathrm{T}}(\mathbf{x}_k)(\mathbf{P}_{k+1}^{-})^{-1} \\
\mathbf{x}_k^s &= \mathbf{x}_k + \mathbf{G}_k(\mathbf{x}_{k+1}^s - \mathbf{x}_{k+1}^{-}) \\
\mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k(\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^{-})\mathbf{G}_k^{\mathrm{T}}
\end{aligned} \tag{4.24}$$

with $\mathbf{x}_k$, $\mathbf{f}$, $\mathbf{F}_{\mathbf{x}}$, and $\mathbf{P}_k$ taken from the EKF. The recursion starts with $\mathbf{x}_T^s = \mathbf{x}_T$ and $\mathbf{P}_T^s = \mathbf{P}_T$ and runs for $k = T-1, ..., 0$.

4.3.2 Unscented Rauch-Tung-Striebel smoother

The corresponding Unscented Rauch-Tung-Striebel smoother (URTS) was derived by Simo Särkkä [22] and is described as

Generate the sigma points

$$\begin{aligned}
\mathcal{X}_k^{(0)} &= \mathbf{x}_k \\
\mathcal{X}_k^{(i)} &= \mathbf{x}_k + \sqrt{n+\lambda} + \left[\sqrt{\mathbf{P}_k}\right]_i, \\
\mathcal{X}_k^{(i)} &= \mathbf{x}_k - \sqrt{n+\lambda} + \left[\sqrt{\mathbf{P}_k}\right]_i, \quad i = 1, ..., n
\end{aligned} \tag{4.25}$$

with

$$\begin{aligned}
\lambda &= \alpha^2(n+\kappa) - n \\
\sqrt{\mathbf{P}}\sqrt{\mathbf{P}}^{\mathrm{T}} &= \mathbf{P}
\end{aligned} \tag{4.26}$$

Compute the process model transformation of the sigma points

$$\mathcal{X}_{k+1}^{(i)} = \mathbf{f}(\mathcal{X}_k^{(i)}), \quad i = 0, ..., 2n \tag{4.27}$$

Compute the predicted mean $\mathbf{x}_{k+1}^{-}$, the predicted covariance $\mathbf{P}_{k+1}$, and the cross-covariance $\mathbf{D}_{k+1}$ with the weights $W_i^{(m)}$ and $W_i^{(c)}$ taken from Equation (4.18)

$$\begin{aligned}
\mathbf{x}_{k+1}^{-} &= \sum_{i=0}^{2n} W_i^{(m)} \hat{\mathcal{X}}_{k+1}^{(i)} \\
\mathbf{P}_{k+1}^{-} &= \sum_{i=0}^{2n} W_i^{(c)} (\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{x}_{k+1}^{-})(\hat{\mathcal{X}}_{k+1}^{(i)} - \mathbf{x}_{k+1}^{-})^{\mathrm{T}} + Q_k \\
\mathbf{D}_{k+1} &= \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{X}_k^{(i)} - \mathbf{x}_k)(\hat{\mathcal{X}}_k^{(i)} - \mathbf{x}_k^{-})^{\mathrm{T}}
\end{aligned} \tag{4.28}$$

Compute the smoother gain $\mathbf{G}_k$, the mean $\mathbf{x}_k$, and the covariance $\mathbf{P}_k^s$

$$\begin{aligned}\mathbf{G}_k &= \mathbf{D}_{k+1}(\mathbf{P}_{\mathbf{k+1}}^-)^{-1} \\ \mathbf{x}_k^s &= \mathbf{x}_k + \mathbf{G}_k(\mathbf{x}_{k+1}^s - \mathbf{m}_{k+1}^-) \\ \mathbf{P}_k^s &= \mathbf{P}_k + \mathbf{G}_k(\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-)\mathbf{G}_k^{\mathrm{T}}\end{aligned} \tag{4.29}$$

The algorithm runs backwards in time for $k = T-1, ..., 0$ initialized with $\mathbf{x}_T^s = \mathbf{x}_T$ and $\mathbf{P}_T^s = \mathbf{P}_T$ from the last filtering step. $\mathbf{x}_k$ and $\mathbf{P}_k$ are taken from the filtering step in Equation (4.22).

4.4 GNSS/INS integration

GNSS provides measurements with a long term absolute accuracy, typically with a low update rate at 1 Hz. The signals can be blocked or have poor quality in environments where the sky is not clearly visible causing unreliable or even loss of positioning. An inertial navigation system (INS), including gyroscopes, accelerometers, and odometers in contrary has good short term accuracy but an over time growing error due to integration of drift and noise. The INS has a high update rate, usually more than 50 Hz and a high availability. A combination of these systems have the possibility of providing a high bandwidth with both short and long term accuracy [11].

There exist different methods for fusing GNSS and INS data with each having its own pros and cons. In the following sections different Kalman filtering methods are reviewed in order to provide an insight of the alternatives.

4.4.1 Loosely coupled integration

In a loosely coupled filter, the GNSS receiver is treated as a sensor measuring the position [27]. This enables the application of any GNSS receiver providing a position as input to the sensor fusion filter. The implementation is simple and stable. The drawback is that a minimum of 4 satellites are needed to get a 3D position measurement from the GNSS sensor. Another problem to consider is that having two cascaded filters can cause the measurement noise of the position being correlated. The Kalman filter is based on a white noise assumption, i.e., uncorrelated noise, of the measurements which could then be false. A picture illustrating the data flow in a loosely coupled filter is depicted in Figure 4.1. The feedback of the integrated navigation solution corrects the INS calculated position. The acquisition aiding from the INS to the GNSS tracking loops reduces the bandwidth of the system which improves the signal tracking in noisy environments [27].

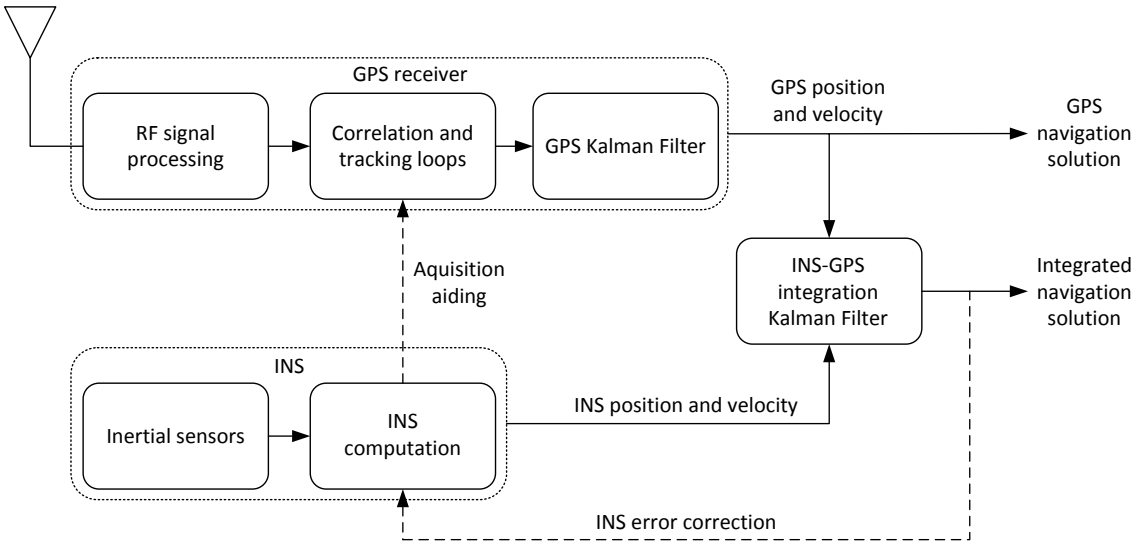


Figure 4.1. Loosely coupled integration [27]

4.4.2 Tightly coupled integration

In a tightly coupled filter the measurements from the GNSS receiver and IMU data are integrated at a deeper level into one single filter which overcomes the problem with colored noise as in the loosely coupled [27]. Here the pseudorange and carrier phase measurements from the GNSS receiver are integrated with IMU in the same filter as depicted in Figure. 4.2. This gives the advantage that less than 4 satellites can aid the filter which is especially good in urban areas. The disadvantage is that it is a more complex system to implement and more hardware specific using the low level information. There also exists a method called ultra-tight coupling [27] where the integration is done already in the tracking loops of the satellite signals inside the receiver. This provides possible improvements of an increased signal to noise ratio, multipath resistance, and faster acquisition of blocked signals but is a highly complex system to implement.

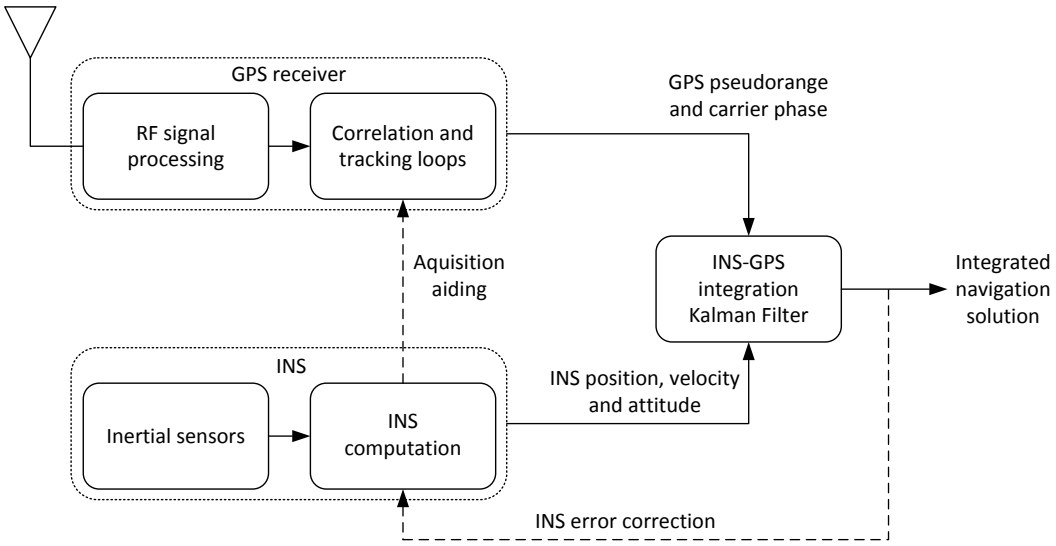


Figure 4.2. Tightly coupled integration [27]

5 Modeling and filter implementation

This chapter describes the motion and sensor models followed by the specific filter implementations. Two different filter implementations for calculating the relative distance are presented, both according the methods described in Sections 4.4.1 and 4.4.2. The first one being a loosely coupled filter for absolute positioning where the absolute positions of two cars can then be used in order to calculate the baseline. The second described implementation is a tightly coupled filter for directly calculating the baseline from the IMU and raw GNSS data.

All calculations are done in a local ENU frame as described in Section 2.4.2. Angles and distances used in this chapter are shown in Figure 5.1. The origin of the coordinate system equals the initial position of one of the receivers. According to this figure also the lateral and longitudinal distance components between two vehicles can be calculated. The derivation ends up in

$$\begin{aligned} r &= \sqrt{\Delta N^2 + \Delta E^2} \\ \theta &= \arctan 2 \frac{\Delta Y}{\Delta X} \\ \Delta\text{Lat} &= r \cos\left(\frac{\pi}{2} - \varphi + \theta\right) \\ \Delta\text{Lon} &= r \sin\left(\frac{\pi}{2} - \varphi + \theta\right) \end{aligned} \tag{5.1}$$

where φ is the heading according to the global coordinate system and θ is the angle between two receivers.

5.1 Motion models

The movement of a particle in a 2D-plane with a heading angle and velocity can be described with a coordinated turn model [3]. The coordinates turn model can be described by either Cartesian or polar velocity. The polar velocity model is chosen since it directly includes the heading angle as a state which is of interest for calculating lateral and longitudinal distances. It is also convenient to use the absolute velocity because this is the measured variable, instead of the components as in a Cartesian model. The model is described as

$$\begin{aligned} \dot{E}(t) &= v(t)\cos(\varphi(t)) \\ \dot{N}(t) &= v(t)\sin(\varphi(t)) \\ \dot{v}(t) &= 0 \\ \dot{\varphi}(t) &= \dot{\varphi}(t) \\ \ddot{\varphi}(t) &= 0 \end{aligned} \tag{5.2}$$

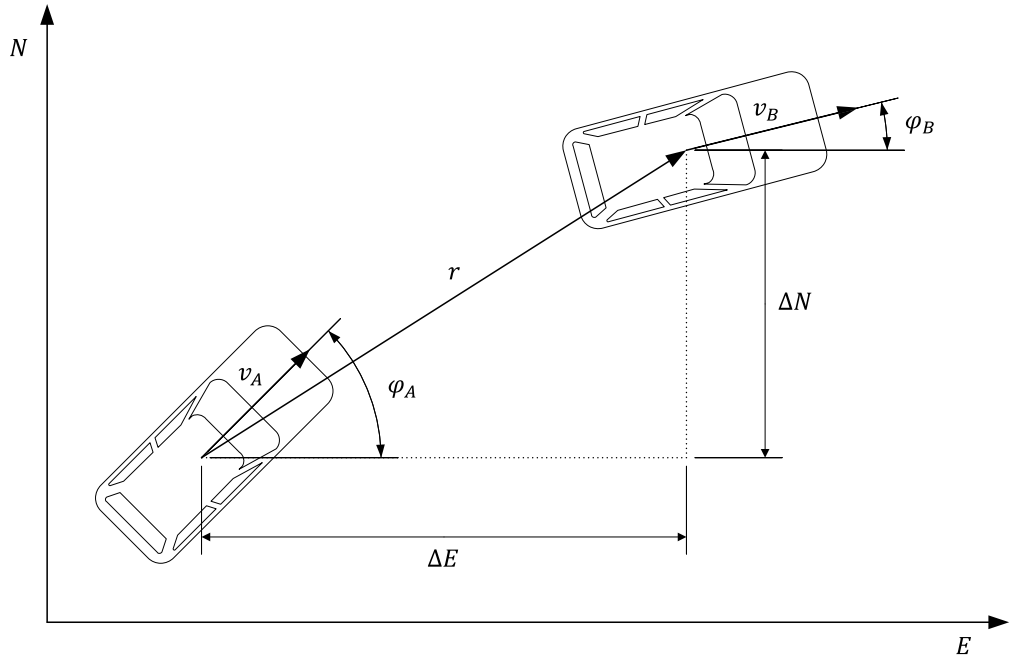


Figure 5.1. Lateral and longitudinal distance between two cars

where E and N are the position related to the center of gravity (COG), v is the velocity and φ is the heading angle. Discretization of Equation (5.2) using Equation (4.4) gives

$$\begin{aligned} E_{k+1} &= E_k + T v_k \cos(\varphi_k) \\ N_{k+1} &= N_k + T v_k \sin(\varphi_k) \\ v_{k+1} &= v_k \\ \varphi_{k+1} &= \varphi_k + T\dot{\varphi}_k \\ \dot{\varphi}_{k+1} &= \dot{\varphi}_k \end{aligned} \tag{5.3}$$

5.1.1 Bicycle model

To make use of the steering angle, a kinematic bicycle model (Figure 5.2) of the car is treated.

Referring to [19] for the derivations with a rear wheel angle fixed to zero, the vehicle slip angle β is described as

$$\beta = \tan^{-1}\left(\frac{l_r \tan\delta}{l_f + l_r}\right) \tag{5.4}$$

where δ is the front wheel angle and l_f, l_r are the lengths depicted in Figure 5.2. Including the steering angle as an input to the system in Equation (5.3) results in

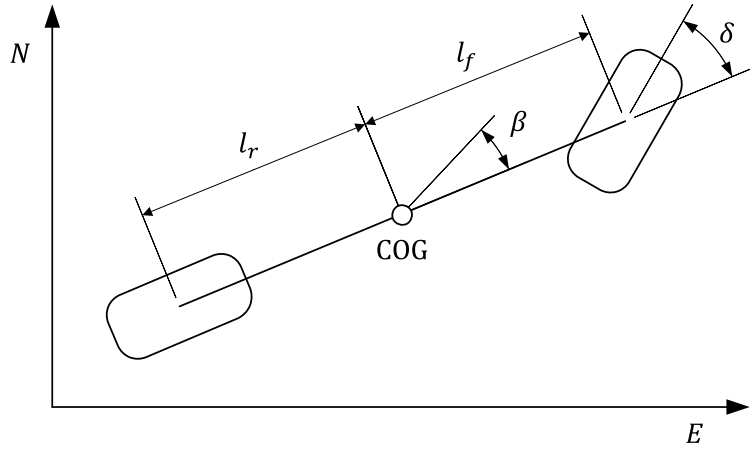


Figure 5.2. Bicycle model

the model

$$\begin{aligned} E_{k+1} &= E_k + T v_k \cos(\varphi_k + \beta_k) \\ N_{k+1} &= N_k + T v_k \sin(\varphi_k + \beta_k) \\ v_{k+1} &= v_k \\ \varphi_{k+1} &= \varphi_k + T\dot{\varphi}_k \\ \dot{\varphi}_{k+1} &= \dot{\varphi}_k \end{aligned} \tag{5.5}$$

5.2 Sensor model

The sensor model is based on the measurements from GNSS and in-vehicle sensors. The GNSS receiver outputs latitude and longitude coordinates which are given in a global coordinate system while the output of the in-vehicle sensors are in a local coordinate system fixed to the car. The GNSS coordinates are therefore transformed to ECEF and then to ENU as described in Section 2.4.2.

The RMS accuracy for NVS is specified to 2.5 m and for u-blox to more than 2.5 m. RTKLIB outputs the covariances of the calculated position which serves as a quality indicator of the position measurement. After testing of different weighting schemes, the following formula is chosen for the east and north coordinates

$$\begin{aligned} \sigma_E^2 &= 2^2 + \sigma_{E,RTKLIB}^2 \\ \sigma_N^2 &= 2^2 + \sigma_{N,RTKLIB}^2 \end{aligned} \tag{5.6}$$

The noise characteristics of the velocity sensor cannot be analyzed during stationary conditions. Due to the fact that it does not show much noise the standard deviation is estimated to a low value of 0.001 m/s.

The yaw rate sensor's standard deviation calculated from stationary measurements is $0.1\,^\circ/s$.

The heading angle is calculated from the current and previous position as

$$\varphi = \arctan 2 \left(\frac{N_k - N_{k-1}}{E_k - E_{k-1}} \right) \tag{5.7}$$

This measure is highly affected by noise in the position and is of that reason given a standard deviation of 90 °. As can be seen in Equation (5.5), the heading angle is updated using yaw rate estimates in each step and these measurements function as a corrector for the heading according to the global coordinate system.

The raw GNSS measurements' standard deviations are weighted according to the qualitative C/N_0 measure. In [8], a closed form solution weighting scheme based on internal receiver parameters is proposed. These parameters are not available and therefore a weighting scheme used in [1] entirely based on C/N_0 is applied according to

$$\sigma^2(C/N_0) = \frac{10^{\frac{C/N_{0,zenith}}{10}}}{10^{\frac{C/N_0}{10}}} \sigma^2_{zenith} \tag{5.8}$$

where $C/N_{0,zenith}$ is the C/N_0 value in zenith and σ^2_{zenith} the standard deviation in zenith. $C/N_{0,zenith}$ is estimated from observations to 50 dBHz. The standard deviation in zenith of carrier phase measurements were set to 0.008 m and pseudorange to 0.75 m. In addition all measurements having a C/N_0 less than 25 dBHz are disregarded.

5.3 Absolute to Relative

The implemented loosely coupled filter is depicted in Figure 5.3 and is slightly modified compared to the one described in Section 4.4.1. The feedback from the INS to the tracking loop is removed since no such possibility exists in the used hardware. Because of that, the improvements of the signal tracking in noisy environments are lost. The position is calculated with the open source software RTKLIB which processes the GNSS data together with the accurate correction terms provided by IGS. It applies PPP as described in Section 2.3.3, the RTKLIB settings are attached in Appendix C. The RTKLIB position estimates, IMU and odometer output form the complete measurement model which for the fully observed case is equal to the state vector

$$\mathbf{y} = \mathbf{x} = \begin{bmatrix} E & N & v & \varphi & \dot{\varphi} \end{bmatrix}^{\mathrm{T}} \tag{5.9}$$

The process model in Equation (5.5) together with the measurement model in Equation (5.9) forms the sensor fusion algorithm. The steering angle is treated as an input. Due to the nonlinearities both EKF and UKF with their corresponding smoothers are evaluated in the implementation. The parameters α, β and κ given in in Section 4.2.2 are applied to the UKF implementation. The process covariance matrix is tuned to be

$$\mathbf{Q} = \mathrm{diag}\left(\begin{bmatrix} 0.1 & 0.1 & 0.01 & (0.2\pi/180)^2 & (0.1\pi/180)^2 \end{bmatrix} \right) \tag{5.10}$$

and the measurement covariance matrix with values motivated from Section 5.2 is formed as

$$\mathbf{R} = \text{diag}\left(\begin{bmatrix}\sigma_E^2 & \sigma_N^2 & 0.001 & (90\pi/180)^2 & (0.1\pi/180)^2\end{bmatrix}\right) \tag{5.11}$$

The covariance matrix $\mathbf{P}$ is initialized to ten times the identity matrix.

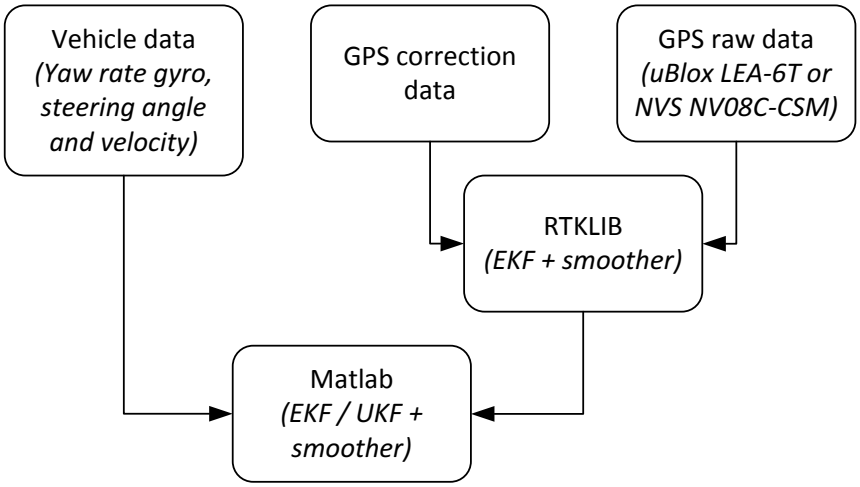


Figure 5.3. Data flow for absolute positioning algorithm

Since the GNSS measurements arrive at 1 Hz and in-vehicle sensors at 50 Hz, the Kalman filter running at 50 Hz has to be updated with a different measurement vector depending on the available information. Operating the filter at 50 Hz thus gives the possibility of calculating a position output at 50 Hz while the position input only updates with 1 Hz.

RTKLIB outputs the standard deviations along with the calculated positions. These are taken into account to modify the measurement covariance to trust positions with high uncertainty less. The heading angle calculation is very sensitive to low speed driving since it needs two positions to calculate the heading. If they are close, measurement noise has a high impact and consequently the heading angle cannot be updated at low speed. These conditions in combination with the multirate filter can be summarized in the pseudocode presented in Algorithm 1.

Having two vehicles running the loosely coupled filter, the distance can be calculated as

$$d = \sqrt{(E_1 - E_2)^2 + (N_1 - N_2)^2} \tag{5.12}$$

This algorithm will further on be referred to as A2R (Absolute to Relative) together with the name of the applied filter or smoother e.g., EKF.

if $\sigma_{GNSS} > \sigma_{Max}$ **or** $\overline{GNSS}$ **then**
  $y = \begin{bmatrix} v & \dot{\varphi} \end{bmatrix}^{\mathrm{T}}$
  $R = \mathrm{diag}(\sigma_v^2, \sigma_{\dot{\varphi}}^2)$
else if $v < v_{min}$ **then**
  $y = \begin{bmatrix} E & N & v & \dot{\varphi} \end{bmatrix}^{\mathrm{T}}$
  $R = \mathrm{diag}(\sigma_X^2, \sigma_Y^2, \sigma_v^2, \sigma_{\dot{\varphi}}^2)$
else
  $y = \begin{bmatrix} E & N & v & \varphi & \dot{\varphi} \end{bmatrix}^{\mathrm{T}}$
  $R = \mathrm{diag}(\sigma_X^2, \sigma_Y^2, \sigma_v^2, \sigma_{\varphi}^2, \sigma_{\dot{\varphi}}^2)$
end if

Algorithm 1: Measurement update

5.4 Relative positioning

Relative positioning can also be performed based on the single and double difference equations as presented in Section 2.3.1. The baseline between two receivers can then be calculated by projecting the difference onto the unit vectors pointing to the each visible satellite.

5.4.1 Related work

Techniques calculating the baseline for moving receivers mounted on e.g., vehicles are presented in [1], [4] and [28]. These approaches are based on Equations (2.6) and (2.8) with some slight modifications. To find a very accurate baseline, the carrier phase measurements have to be used. In [4], the double difference is taken over two consecutive epochs which removes the need of carrier phase ambiguity resolution. The drawback is that this algorithm needs to know the initial position of the receivers and the error accumulation over time.

Both [1] and [28] use a Kalman filter for estimating floating point values of the differenced ambiguity parameter. To achieve a good accuracy they should be fixed to integer values.

In [28], the single differenced ambiguities are estimated with a Kalman filter. The ambiguities are then double differenced and an attempt is made to fix them to integers with the Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method. If the LAMBDA method succeeds in finding the integer values a very accurate baseline can be determined. In [1], the Kalman filter approach is also used with the double differenced measurements as states in the filter.

Utilizing the method presented in [28], the single difference Equations (2.5) and (2.6) for j satellites expressed in matrix form, a relation from the single differenced measurements $\Delta P \in \mathbb{R}^{j\times 1}$ and $\Delta L \in \mathbb{R}^{j\times 1}$ to the ambiguities $\Delta N \in \mathbb{R}^{j\times 1}$, baseline

components $\Delta r_{AB}^{ECEF} \in \mathbb{R}^{3\times1}$ and clock offset $c\delta t_{AB}$ is found as

$$\begin{bmatrix} \Delta P \\ \Delta L \end{bmatrix} = \begin{bmatrix} G & 1 \\ G & 1 \end{bmatrix} \begin{bmatrix} \Delta r_{AB}^{ECEF} \\ c\delta t_{AB} \end{bmatrix} + \begin{bmatrix} 0 \\ \lambda I \end{bmatrix} \Delta N \tag{5.13}$$

where $G \in \mathbb{R}^{j\times3}$ is the unit vector matrix pointing from the receiver to each satellite as

$$G = \begin{bmatrix} u_x^1 & u_y^1 & u_z^1 \\ \vdots & \vdots & \vdots \\ u_x^j & u_y^j & u_z^j \end{bmatrix} \tag{5.14}$$

Lambda method

The LAMBDA method is estimating the integer GNSS ambiguities [26]. It processes on the float estimated differential satellite phase ambiguity parameters as well as the corresponding covariances. The LAMBDA method enables much more stable carrier phase position solutions by stabilizing the ambiguities but it needs fairly accurate input data. This was mostly not the case in the measurements taken for this thesis.

5.4.2 Proposed implementation

The proposed algorithm consists of three steps, determining a floating point estimate of the ambiguity parameter, attempting to fix the integer ambiguity, and correcting the baseline if an integer ambiguity could be fixed. The filter is implemented as an Extended Kalman filter and smoother.

The proposed state vector $\mathbf{x} \in \mathbb{R}^{j+8\times1}$ is defined as

$$\mathbf{x} = \begin{bmatrix} \varphi_A \\ \varphi_B \\ \dot{\varphi}_A \\ \dot{\varphi}_B \\ \Delta r_{AB} \\ c\delta t_{AB} \\ \Delta N \end{bmatrix} \tag{5.15}$$

and all states from Δr_{AB} on are initialized by solving Equation (5.13) using least squares. φ and $\dot{\varphi}$ are the heading angles and turn rates of the two cars A and B in the ENU reference coordinate system and initialized to 0. $\Delta r_{AB} \in \mathbb{R}^{3\times1}$ is the baseline between both cars in the three ENU components and $c\delta t_{AB}$ is the estimates differential clock drift. The clock drift cannot be neglected since the clocks of the applied GNSS receivers are much less stable than it would be the case with professional receivers, this is discussed in Section 6.4. ΔN contains the estimated

differential satellite phase ambiguity parameters for each selected satellite. This implies that the size of the state vector is changing over time depending on the number of observed satellites j.

Not having a fixed size state vector is proposed in order to speed up computations since this approach does not require to reserve 32 states for all the GPS satellites or even 56 states if GLONASS is additionally included. Typically not more than a third of all available satellites are observed at one time epoch. Since all the corresponding estimated differential satellite phase ambiguity parameters as well as the entries in the covariance matrix should be reset anyway if the connection to satellites is lost this approach enables a large reduction in dimensionality without dropping any valuable information. However this requires to transform the state $\mathbf{x}$ and covariance matrix, $\mathbf{P} \in \mathbb{R}^{j+8 \times j+8}$ each time the set of observed satellites changes but the resources spent on this transformations are by magnitudes smaller than the ones which are saved by reducing the dimensionality.

The initial covariance matrix is set to be

$$\mathbf{P} = \operatorname{diag}\left(\left[\pi, \pi,\ 1,\ 1,\ 4,\ 4,\ 4,\ 20000000,\ 3_1 \cdots 3_j\right]\right) \tag{5.16}$$

In order to calculate single differences on the phase ambiguity only satellites observed by both receivers are taken into account. In addition the satellites have to fulfill a C/N_0 threshold of 25 dBHz and an elevation angle of at least 5 degrees.

The motion model for the relative base vector is derived by the taking the difference between two single vehicle models from Equation (5.3). The phase ambiguities are kept constant as they do not change over epochs. This forms the nonlinear state prediction equation

$$\mathbf{x}_{k+1} = \begin{bmatrix} \varphi_{A,k} + T\dot{\varphi}_{A,k} \\ \varphi_{B,k} + T\dot{\varphi}_{B,k} \\ \dot{\varphi}_{A,k} \\ \dot{\varphi}_{B,k} \\ \Delta E_{AB,k} + T[v_{A,k}\cos(\varphi_{A,k}) - v_{B,k}\cos(\varphi_{B,k})] \\ \Delta N_{AB,k} + T[v_{A,k}\sin(\varphi_{A,k}) - v_{B,k}\sin(\varphi_{B,k})] \\ \Delta U_{AB,k} \\ c\delta t_{AB,k} \\ \Delta N_k \end{bmatrix} \tag{5.17}$$

The linearized state transition matrix consists of an upper part $\mathbf{F}_{upper} \in \mathbb{R}^{8\times 8}$

$$\mathbf{F}_{upper,k} = \begin{bmatrix} 1 & 0 & T & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -Tv_A\sin(\varphi_{A,k}) & Tv_B\sin(\varphi_{B,k}) & 0 & 0 & 1 & 0 & 0 & 0 \\ Tv_A\cos(\varphi_{A,k}) & -Tv_B\cos(\varphi_{B,k}) & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.18}$$

and a lower part, which is an identity matrix with the size equal to the number of observed satellites at that epoch. They form $\mathbf{F} \in \mathbb{R}^{j+8 \times j+8}$ as

$$\mathbf{F} = \text{diag}\left(\left[\mathbf{F}_{upper},\ \mathbf{I}_{j\times j}\right]\right) \tag{5.19}$$

The time constant T is the filter sampling time. The corresponding process covariance matrix $\mathbf{Q} \in \mathbb{R}^{j+8 \times j+8}$ was tuned to be

$$\begin{aligned}\mathbf{Q} = \text{diag}\Big(\big[&0.5^2,\ 0.5^2,\ (0.1\pi/180)^2,\ (0.1\pi/180)^2,\ 0.25^2,\\ &0.25^2,\ 0.5^2,\ 20000,\ 0.01_1^2 \cdots 0.01_j^2\big]\Big)\end{aligned} \tag{5.20}$$

The measurement model including all available measurements is described as

$$\mathbf{y_k} = \begin{bmatrix}\varphi_{A,k}\\ \varphi_{B,k}\\ \dot{\varphi}_{A,k}\\ \dot{\varphi}_{B,k}\\ \Delta P_k\\ \Delta L_k\end{bmatrix} = \begin{bmatrix}\varphi_{A,k}\\ \varphi_{B,k}\\ \dot{\varphi}_{A,k}\\ \dot{\varphi}_{B,k}\\ GR\Delta r_{AB,k} + c\delta t_{AB,k}\\ GR\Delta r_{AB,k} + c\delta t_{AB,k} + \Delta N_k\end{bmatrix} \tag{5.21}$$

with R being the transformation matrix from ENU to ECEF.

The multirate measurement updates are handled in a similar way as in the absolute to relative Algorithm 1. The first case of Algorithm 2 in Appendix A is executed when a GNSS signal is available but the position fixes for both cars are assumed to be bad. The second case in in effect if no GNSS measurements are available or both are assumed to be bad. In the third and fourth case one of the two cars receives a valid GNSS fix and in the last case all measurements are taken into account. Conditions for a GNSS fix to be treated as not fully reliable are

- The standard deviations for a PPP position fix exceed a threshold
- The height component changes too much between two samples
- The velocity is too low
- The CAN speed measurement differs too much from the estimated speed according to GNSS measurements

If one of these conditions is fulfilled for a GNSS measurement the corresponding heading angle will not be updated using the position fixes but only by using the IMU data.

Cycle slip detection

In case that the satellite signal is blocked, the carrier phase tracking loses count of some number of cycles. This means that the determined phase ambiguity is not valid anymore since the carrier phase is relative measurement from the time it started counting. In [4] it is proposed to use the doppler shift to predict the next carrier phase measurement. This gives the equation

$$\hat{\phi}_{k+1} = \phi_k + D_k \tag{5.22}$$

If $\hat{\phi}$ differs with more than a given threshold from the measured ϕ, it is regarded as a cycle slip and the corresponding ambiguity state as well as covariance are reinitialized. An additional method used in [9] was also implemented. This method also detects problems related to the pseudorange and resulted in more stable solutions. The equation is defined as

$$dN = \left((\Delta P_k - \Delta L_k) - (\Delta P_{k-1} - \Delta L_{k-1})\right)/\lambda \tag{5.23}$$

and if dN exceeds a threshold, the measurement is treated as cycle slip affected.

5.5 Implementation aspects

Some further aspects have to be taken into consideration in order to implement the presented algorithms.

5.5.1 Data synchronization

The GNSS receiver provides timestamped data with the GPS time format. The CAN data only contains a timestamp related to the time of when logging was started. Therefore the data needs to be synced in time and there is no common measured variable. But the GNSS position data can be utilized to calculate the speed, which is also contained as a variable in the CAN data. By calculating the cross-covariance between these velocity signals the time offset can be found.

The signals at the CAN-bus are provided at an over time stable frequency. However some jitter can be observed in the data and since the timestamps are available the complete measurement vectors of speed and yaw rate can be resampled uniformly.

5.5.2 Detecting signal quality

As mentioned earlier not all available satellite signals are necessarily taken into account by the algorithms. There are many indicators, which satellites should be excluded and which ones should not. In the algorithms some hard and some soft

criteria are used. Hard ones are elevation angle and C/N_0 thresholds. If the elevation angle above earth surface of a satellite or the C/N_0 is too low the corresponding satellite is excluded independent of all other criteria since the signal is not reliable enough. The same applies to satellites which send a message that their data is completely unhealthy at the moment. All the observed satellites which are left after the hard exclusion are sorted according to soft criteria. Such criteria can be how long a specific satellite was observed in total in order to use common satellites over ideally the whole observation time. Other criteria can be a good distribution of the satellites over the whole sky to achieve a good geometry or high C/N_0.

These general criteria determine which satellite signals are inputted to the filter algorithms 1 and 2. But it is up to the algorithms to decide if they treat the signals as fully reliable, partly reliable or completely unreliable.

6 Results

This chapter treats the performance of the developed algorithms for absolute and relative positioning compared in various environments using statistical methods. All drives were done using Volvo V40 if not stated differently.

Statistical measures for the resulting offsets between the positioning algorithms and the reference systems are the root mean square error (RMSE), standard deviation (SD), and the 68th as well as the 95th percentile. The percentiles are always taken from the absolute values of the errors, i.e., a 68th percentile of x m states that 68% of the data is closer to the true value than x m. If a normal distribution with a mean close to zero would be given, the 68th absolute percentile would be close to the standard deviation.

6.1 Absolute positioning

This section presents the performance of the absolute algorithms presented in Section 5.3. Due to the lack of reference data their results are mostly compared to the street positions according to Google earth. Comparisons of the collected data to the positions in Google earth led to the conclusion that Google earth provides a fairly accurate reference, which also holds in curves and even through tunnels. It has to be taken into account that the images are mostly not taken from a zenith angle and therefore buildings sometimes cover parts of streets.

6.1.1 Test Track

It was possible to perform one short reference test using a high precision reference system, the RT Range System from Oxford Technical Solutions. This is a RTK system using a basestation nearby. Two drives with this system were done at the same time, one with a Volvo XC60 and one with a XC90. These drives are marked as (1) and (2) in Table 6.1. On each car one of these systems consisting of a dual frequency GPS receiver and a very accurate IMU was installed. The absolute accuracy of RT Range is specified to 2 cm and regarded as ground truth.

The reference drive took place at a Volvo test track for country roads as shown in Figure 6.1 on which two laps were driven in a total of 8:21 minutes. The results are shown in Table 6.1.

The measurements were performed with three GNSS antennas in a row on each car roof. The baseline between the reference antenna and the u-blox antenna was approximately 30 cm and reference to NVS 50 cm. Therefore all the results for u-

Figure 6.1. Test Track with north-up map orientation

Algorithm	Receiver	RMSE [m]		SD [m]		68th [m]		95th [m]	
		1	2	1	2	1	2	1	2
EKF	u-blox	1.318	1.503	0.451	0.543	1.496	1.616	2.005	2.387
UKF	u-blox	1.314	1.506	0.444	0.542	1.479	1.628	1.989	2.382
ERTS	u-blox	1.240	1.357	0.374	0.455	1.422	1.439	1.798	2.055
URTS	u-blox	1.240	1.359	0.374	0.458	1.423	1.442	1.791	2.065
EKF	NVS	1.448	1.744	0.470	0.701	1.585	1.854	2.146	2.593
UKF	NVS	1.446	1.750	0.466	0.707	1.589	1.867	2.136	2.598
ERTS	NVS	1.404	1.542	0.420	0.534	1.515	1.761	2.027	2.238
URTS	NVS	1.403	1.540	0.419	0.533	1.511	1.762	2.026	2.234

Table 6.1. Absolute positioning baseline

blox and NVS receivers except the standard deviation cannot be compared to each other directly. Also both drives and therefore the effect of the different IMUs on the algorithms cannot be compared directly due to low sample quantity. Noticeable results which can be derived from Table 6.1 for absolute positioning are

- The results of EKF and UKF as well ERTS and URTS are very similar.
- The smoothed results are always better than the filtered ones, as it can be expected.
- The u-blox receiver provides better standard deviation values. It also seems

to deliver better positions since RMSE, 68th, and 95th percentile are mostly lower than the values for the NVS receivers minus the difference in real baseline (20 cm).

6.1.2 City

This test drive shows how massive multipath effects in urban areas are handled by the absolute positioning algorithms and the u-blox receiver. In Figure 6.2 five different results based on the same measurement are plotted:

- Red: The GPS measurements using correction data from IGS processed with PPP filtering and smoothing in RTKLIB. This is used as basis for all further processing.
- Dark blue: The filtered GPS measurements fused with IMU data in an EKF.
- Bright blue: The filtered GPS measurements fused with IMU data in an UKF.
- Dark green: The filtered GPS measurements fused with IMU data in an ERTS.
- Bright green: The filtered GPS measurements fused with IMU data in an URTS.

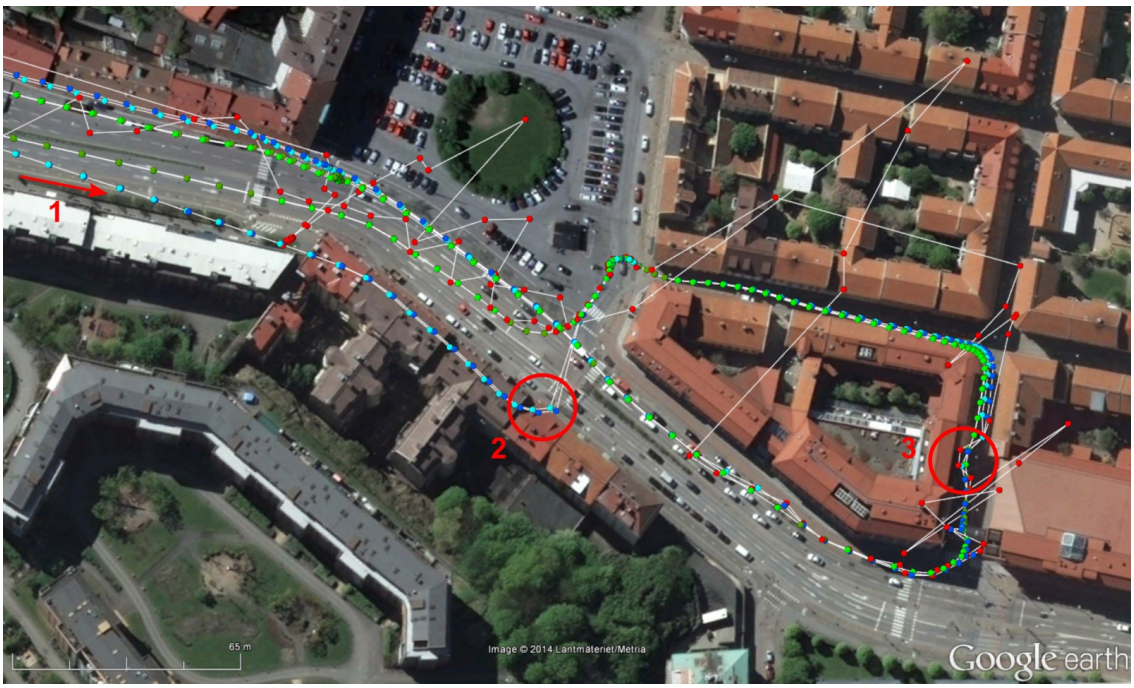


Figure 6.2. City track with west-up map orientation

Without the IGS correction data there would have been nearly no GPS fix in this area. The red track shows that the u-blox receiver has major problems in narrow

urban canyons and also does not perform too good on wide roads if there are big buildings nearby. It can also be observed that EKF and UKF perform nearly identical under these conditions. Furthermore, coming from the lower lanes on the left side of Figure 6.2 (1) there are no GPS fixes which are treated as valid. Therefore both filtered solutions in blue drift away to the right side of the street until a new valid GPS fix arrives to the filter and the position gets corrected (2). This drift is mostly due to an inaccurate initial heading angle. In contrast to that, both smoothed solutions keep the lane nearly perfectly. On the right part of the figure there are some GPS samples which are close to the real position of the car and therefore treated as valid even though they are still off by some meters. This causes an erroneous turn in all the solutions (3).

6.1.3 Dead Reckoning

In urban canyons there are still some valid GPS fixes to support the positioning algorithm. In contrast to that, solutions in tunnels have to fully rely on the dead reckoning abilities of the used algorithms. For this test the 1.6 km long Götatunneln in the center of Gothenburg was chosen and the GPS measurements before and after the tunnel were taken with a u-blox receiver. The results are depicted in Figure 6.3 with colors as used in Figure 6.2.

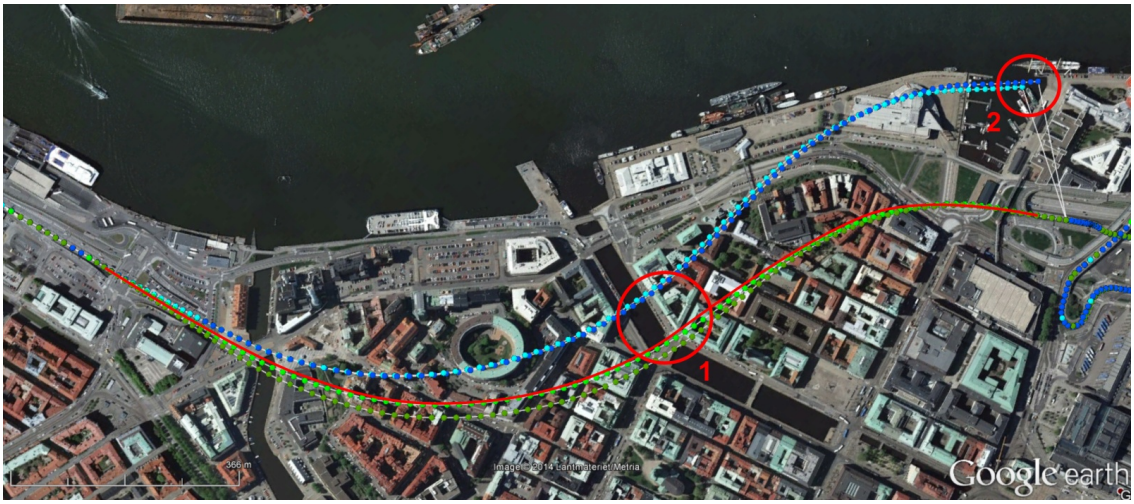


Figure 6.3. Dead reckoning track with north-west-up map orientation

The red track is the location of the tunnel according to Google earth. It can be assumed that this data is based on a high performance dead reckoning system which should be very accurate and therefore is a valid reference. As Figure 6.3 demonstrates, the smoothed solutions are by magnitudes better that the filtered ones because they can take advantage of data from both ends of the tunnel in order to aid the solution. Therefore not only the offset at the end but also in the mid of the tunnel is compared in Table 6.2.

Algorithm	Offset mid of tunnel (1)	Offset end of tunnel (2)
EKF	74 m	214 m
UKF	78 m	207 m
ERTS	21 m	<0.5 m
URTS	7 m	<0.5 m

Table 6.2. Dead reckoning (Götatunneln)

According to Table 6.2 the difference between EKF and UKF themselves is very small. But once combined with a corresponding smoother the UKF provides better results than the EKF.

Test drives in other tunnels led to comparable results. One example is the 2.1 km long Lundbytunneln, as shown in Table 6.3. The results are taken during two drives through the tunnel with different cars. Drive 1 was done eastwards with a Volvo S60 and drive 2 westwards in a Volvo V40.

Algorithm	Offset mid of tunnel		Offset end of tunnel	
	Drive 1	Drive 2	Drive 1	Drive 2
EKF	33 m	34 m	215 m	234 m
UKF	30 m	31 m	207 m	229 m
ERTS	20 m	20 m	<0.5 m	<0.5 m
URTS	19 m	13 m	<0.5 m	<0.5 m

Table 6.3. Dead reckoning (Lundbytunneln)

Despite the different preconditions like a different driving directions and IMUs, the results of both drives in Lundbytunneln are very similar. Compared with Götatunneln the filters perform better in the mid of the tunnel which might be caused by the different curve shapes of both tunnels.

6.2 Relative positioning

All relative positioning drives include a reference measurement to estimate the performance of the algorithms. The default reference value is the distance between both cars according to the chasing vehicles built-in radar sensor. This sensor delivers a resolution of 0.1m, but the accuracy is usually worse. Radar measurements in general are not very good references because the radar does not measure the distance between the cars center points but from the front of the hunter to the closest part of the target vehicle. The difference between this measured distance and the desired center-to-center distance is not constant if the cars drive curves as shown in Figure 5.1.

All drives were done with adaptive cruise control activated in the chasing vehicle which controls the distance between the cars to a set value consisting of a fixed offset

and a part proportional to the velocity. For all measurements the baseline profiles are given and due to adaptive cruise control they can also be roughly translated to the corresponding velocities. The complete speed profiles are attached in Appendix B. In contrast to the absolute positioning no city test drives are presented in this section. The reason is that there was no usable reference over a reasonable period of time because the radar lost target in tight curves and roundabouts.

6.2.1 Country Roads

This dataset was collected on a country road passing through several forests and small villages as shown in Figure 6.4. The weather was cloudy, but not rainy. The main challenge on this dataset are areas with bad GPS reception in forests with trees close to the road as well as rather tight curves which cause problems to the radar.

Figure 6.4. Country road track with west-up map orientation

The cars were driving between 0 and 80 km/h, their baseline is depicted in Figure 6.5 and the statistical results in Table 6.4.

The large baseline errors around second 200 in Figure 6.6 are caused by poor GPS reception due to trees close to both sides of the road. Some samples around second 500 were excluded from the analysis since the radar tracked the wrong target at that time.

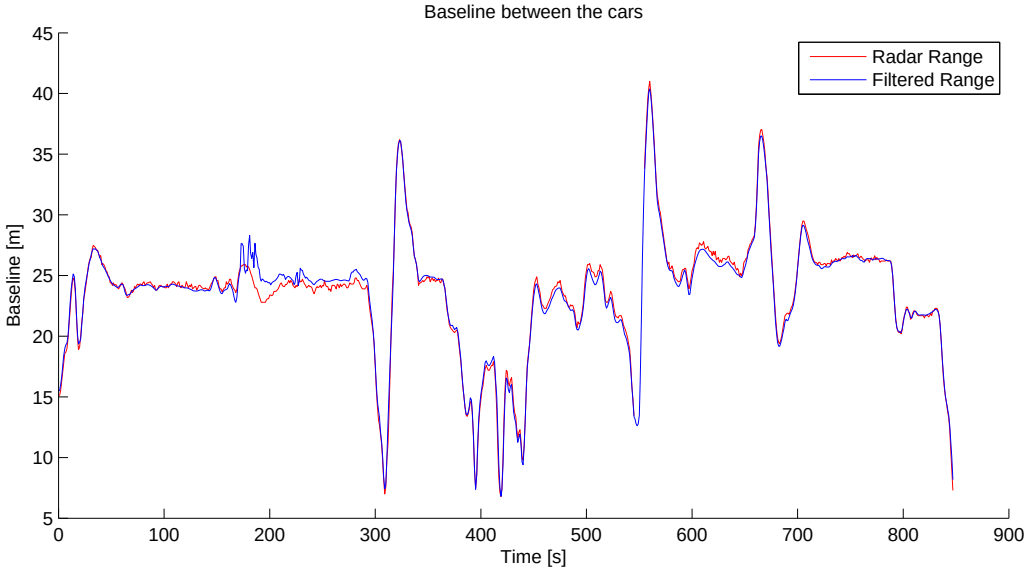


Figure 6.5. Radar range vs. filtered range of the relative algorithm in Country Roads

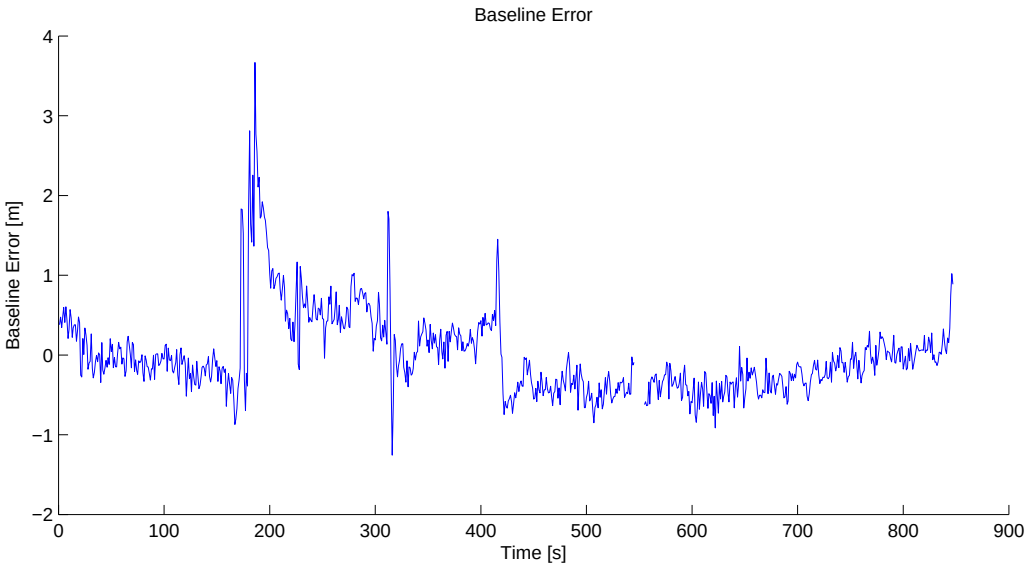


Figure 6.6. Baseline error of the relative algorithm in Country Roads

Algorithm	Receiver	RMSE [m]	SD [m]	68th [m]	95th [m]
Relative	u-blox	0.514	0.515	0.418	0.900
EKF A2R	u-blox	0.537	0.506	0.457	1.075
UKF A2R	u-blox	0.537	0.507	0.456	1.074
ERTS A2R	u-blox	0.468	0.444	0.419	0.931
URTS A2R	u-blox	0.468	0.443	0.419	0.927
EKF A2R	NVS	0.922	0.751	0.660	1.866
UKF A2R	NVS	0.872	0.698	0.661	1.796
ERTS A2R	NVS	0.571	0.426	0.573	1.035
URTS A2R	NVS	0.567	0.412	0.575	1.026

Table 6.4. Country Road (14:06 min)

6.2.2 Freeway

The data was collected heading north at road E6 as illustrated in Figure 6.7. The weather was cloudy, but not rainy. The vehicle speed varied between 90-110 km/h with both cars overtaking at some occasions. Figure 6.8 shows the baseline during the drive.

Figure 6.7. Freeway track with west-up map orientation

As Table 6.5 shows, the relative algorithm clearly outperforms the A2R algorithm in areas having a good GNSS reception like on freeways. Figure 6.9 depicts the baseline error between the radar range and the relative algorithm. Some extreme values might be caused by radar errors. The rest of the error consists of noise and a drifting offset. The latter part of that might be caused by radar target drift or filtering effects.

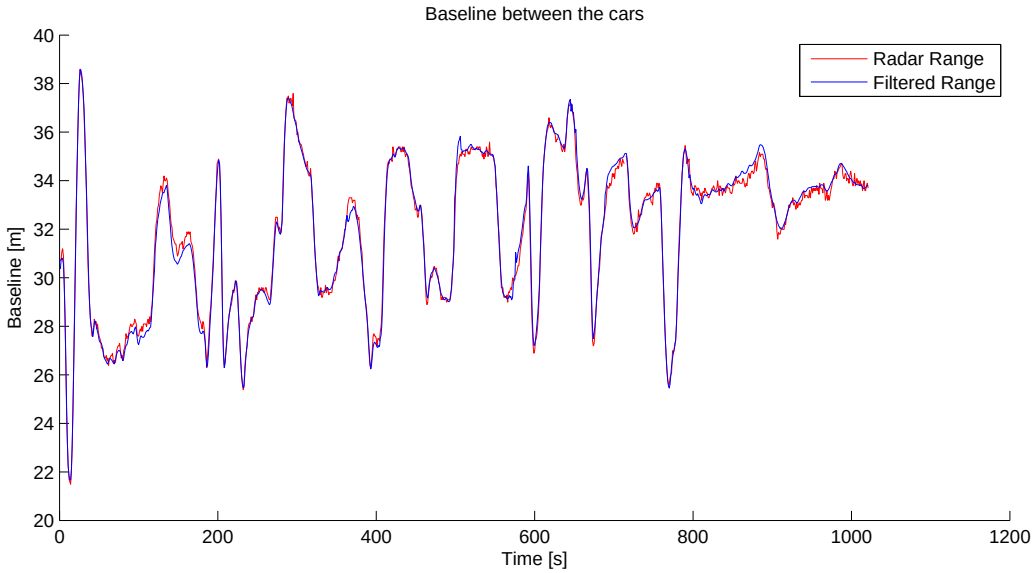


Figure 6.8. Radar range vs. filtered range of the relative algorithm in Freeway

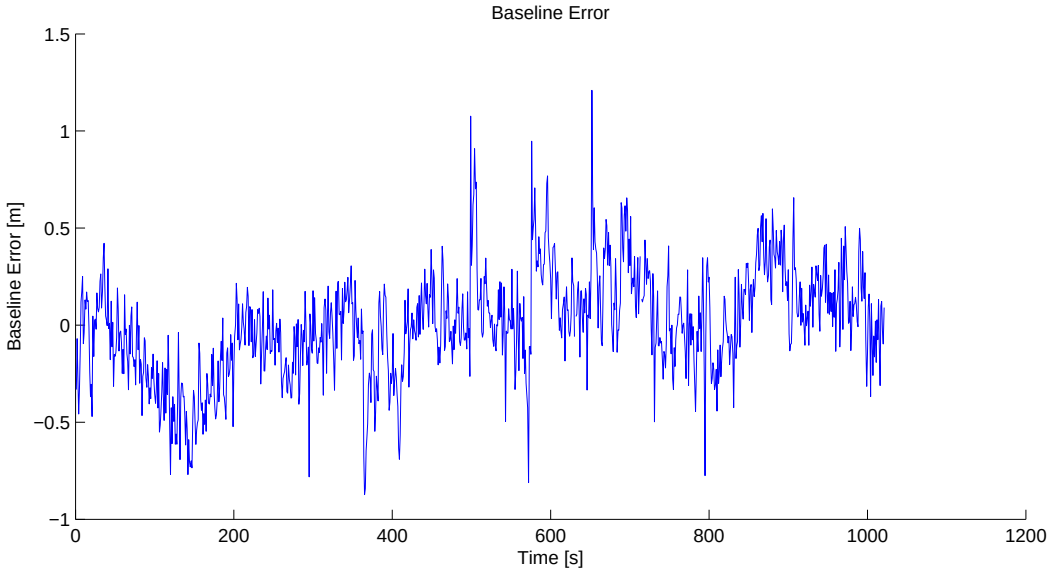


Figure 6.9. Baseline error of the relative algorithm in Freeway

Algorithm	Receiver	RMSE [m]	SD [m]	68th [m]	95th [m]
Relative	u-blox	0.273	0.273	0.257	0.538
EKF A2R	u-blox	0.630	0.622	0.483	1.395
UKF A2R	u-blox	0.632	0.624	0.481	1.411
ERTS A2R	u-blox	0.428	0.405	0.373	0.846
URTS A2R	u-blox	0.429	0.406	0.375	0.849
EKF A2R	NVS	1.070	1.061	0.481	1.126
UKF A2R	NVS	1.061	1.052	0.480	1.127
ERTS A2R	NVS	0.438	0.433	0.363	0.734
URTS A2R	NVS	0.440	0.434	0.364	0.731

Table 6.5. Freeway (17:00 min)

6.2.3 Test Track

As described previously the radar is not a perfect reference system, but it had to be used since no more accurate system was available most of the time. However it was possible to perform one short reference test using a high precision reference system, the RT Range System from Oxford Technical Solutions as described in Section 6.1.1. This RTK system can not only be used in a basestation-rover setup, but also in a rover-rover setup, measuring the distance between two moving objects. The baseline accuracy is specified to 3 cm on a 200 m baseline. These drives were done with a Volvo XC60 and a XC90.

The reference drive took place at the Volvo test track for country roads shown in Figure 6.1. Two laps were driven. In difference to all the other test drives this one was not performed using adaptive cruise control, which led to a more dynamic baseline. The resulting baseline is depicted in Figure 6.10. Table 6.6 presents the statistical results for this reference test.

Algorithm	Receiver	RMSE [m]	SD [m]	68th [m]	95th [m]
Relative	u-blox	0.359	0.359	0.360	0.676
EKF A2R	u-blox	0.552	0.552	0.502	1.118
UKF A2R	u-blox	0.552	0.552	0.504	1.114
ERTS A2R	u-blox	0.541	0.529	0.439	1.187
URTS A2R	u-blox	0.542	0.529	0.439	1.187
EKF A2R	NVS	0.611	0.600	0.513	1.291
UKF A2R	NVS	0.609	0.597	0.502	1.280
ERTS A2R	NVS	0.446	0.446	0.386	0.915
URTS A2R	NVS	0.445	0.445	0.384	0.917

Table 6.6. Test Track (8:21 min)

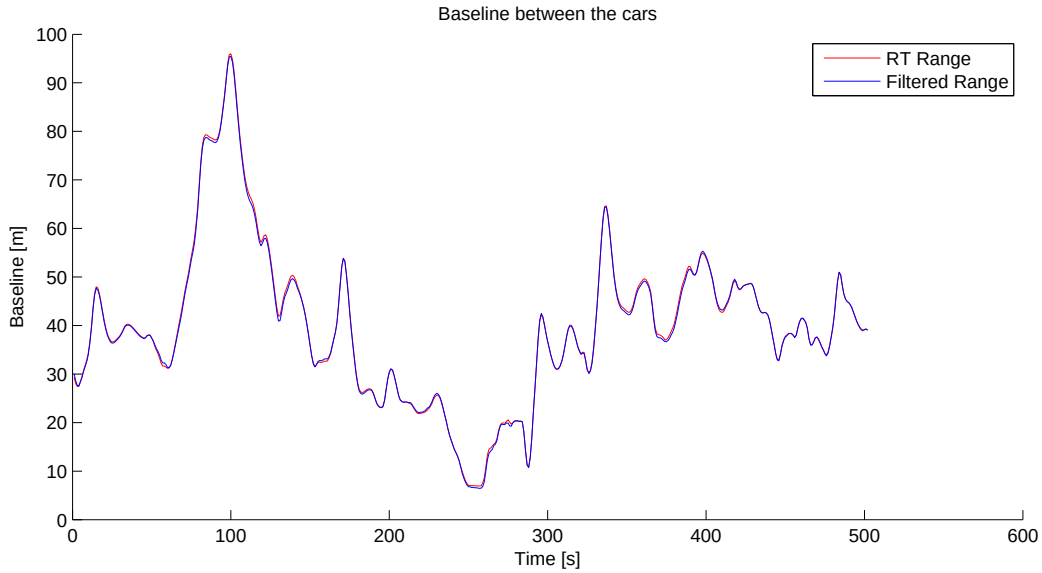


Figure 6.10. RT Range vs. filtered range of the relative algorithm in Test Track

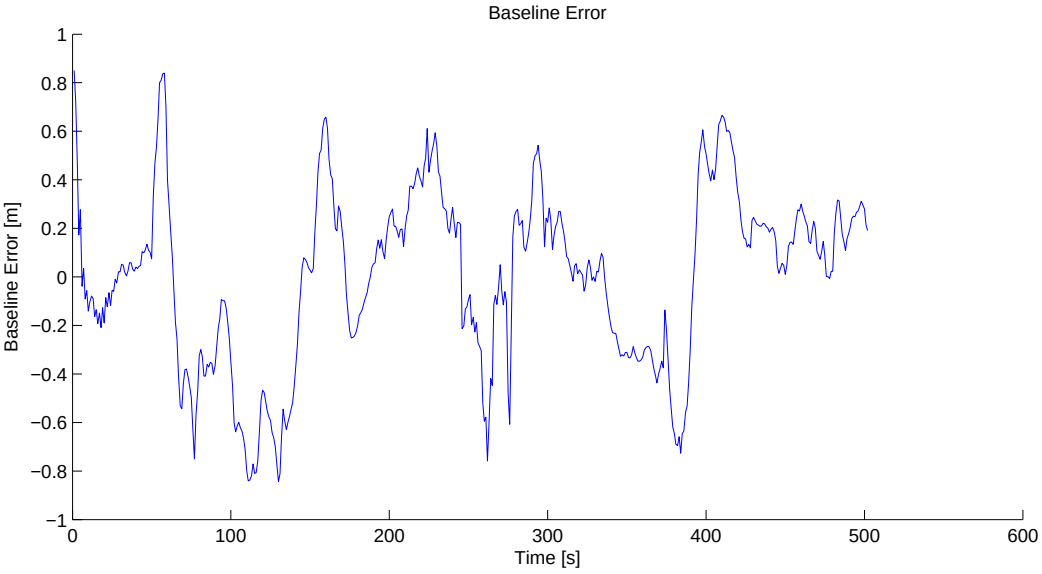


Figure 6.11. Baseline error of the relative algorithm in Test Track

6.3 Smoother performance

The implemented smoothers improved the results for all of the measurements, absolute as well as relative and for both receivers. However the advantages due to the smoothing were not evenly distributed over all measurements, they are presented in in Table 6.7.

Some noteworthy results can be drawn from Table 6.7

Algorithm	Receiver	RMSE		SD		68th		95th	
		ERTS	URTS	ERTS	URTS	ERTS	URTS	ERTS	URTS
Absolute 1	u-blox	6%	6%	17%	16%	5%	4%	10%	10%
Absolute 2	u-blox	10%	10%	16%	16%	11%	11%	14%	13%
A2R Country	u-blox	13%	13%	12%	13%	8%	8%	13%	14%
A2R Freeway	u-blox	32%	32%	35%	35%	23%	22%	39%	40%
A2R Test Track	u-blox	2%	2%	4%	4%	13%	13%	-6%	-7%
Absolute 1	NVS	3%	3%	11%	10%	4%	5%	6%	5%
Absolute 2	NVS	12%	12%	24%	25%	5%	6%	14%	14%
A2R Country	NVS	38%	35%	43%	41%	13%	13%	45%	43%
A2R Freeway	NVS	59%	59%	59%	59%	24%	24%	35%	35%
A2R Test Track	NVS	27%	27%	26%	25%	25%	24%	29%	28%

Table 6.7. Error decrease due to smoothing compared to the corresponding filters

1. Out of all the given parameters the SD takes the biggest advantage of smoothing.

2. EKF and UKF gain about the same amount of accuracy by applying the corresponding smoothers.

3. The NVS-results benefit more in the RMSE than in the percentiles compared to the u-blox results

4. The NVS-results take much more advantage of smoothing than the u-blox results.

Point 1 can be expected since the SD measures the smoothness of a dataset. Number 2 could have also been predicted since the EKF and UKF are delivering very similar results and therefore also the corresponding smoothed results should be close. Point 3 can be verified by the observation that the filtered NVS result contains more single outliers than the u-blox result. Due to squaring this has a large effect on the RMSE, but a little on the percentiles. Many of these outliers are removed by the smoothers. A possible assumption for number 4 could be the observation that the filtered results for the u-blox receivers are usually more accurate than the ones for the NVS receivers. So the NVS results have a higher potential for improvement. But the smoothed results for the NVS receivers are usually better than the ones for the u-blox receivers. This contradicts that result number 4 is only caused by a higher potential. In fact it is mainly caused by a similar reason than number 3 - the filtered results for the NVS receivers have more single outliers inside and outside the 68th and 95th percentile, which get removed by the smoother. This is depicted in Figure 6.12, which is taken from the start of the Test Track measurement comparing the EKF and ERTS results for both receivers.

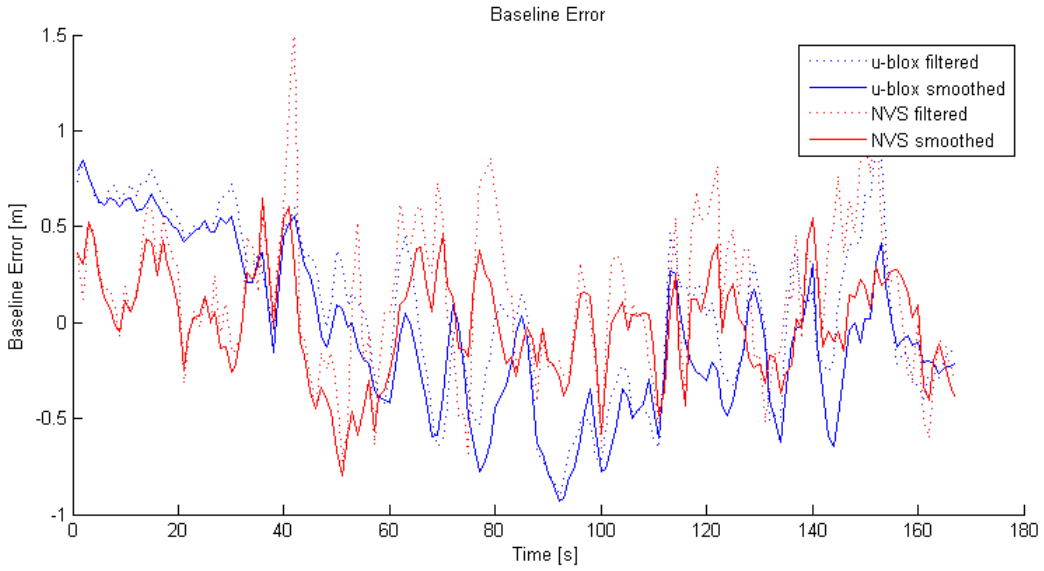


Figure 6.12. A2R baseline error comparison

Smoothing can contribute the most in areas with missing GNSS fixes and even more bad ones as below bridges, close to trees or buildings and in tunnels. This effect cannot be extracted from Table 6.7 and an extreme example is provided in the dead reckoning drives in Section 6.1.3.

6.4 Receiver clock drift

Even though the receiver clocks are steered to GPS time, they are not perfectly in sync with the atomic clocks on the satellites and drift away over time. If the time drift exceeds a certain threshold, the receiver time jumps by one millisecond. This effect has to be considered since a time difference of this magnitude equals to a pseudorange of 300 km.

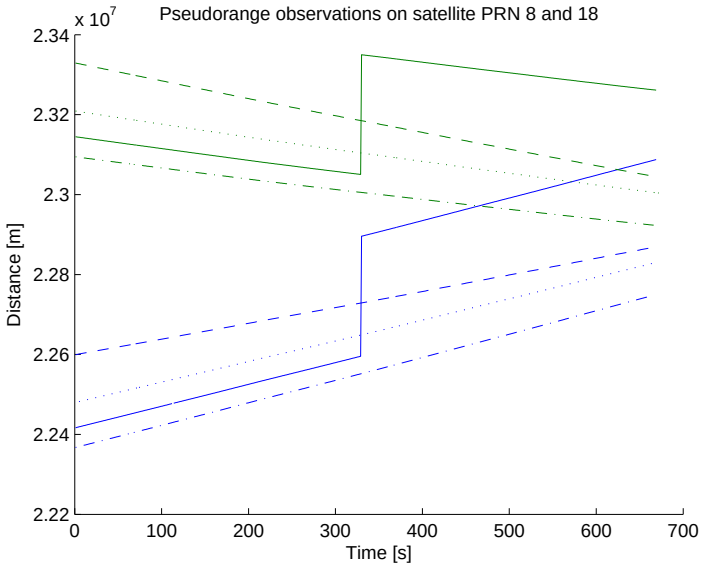


Figure 6.13. Clock drift comparison

Figure 6.13 shows an example of clock drift and time slips. Here the observations of four identical receivers having the same configuration and identical antennas mounted on the same driving car are plotted with different line styles. For simplicity only two satellites, plotted in blue and green, are included in the plot but the qualitative observations hold for all satellites. The first 300 seconds show that the measured pseudoranges differ up to more than 200 km (0.7 ms) between the receivers due to timing inaccuracies. The observed pseudorange jump in second 330 is not in one of the receivers having the lowest or highest pseudoranges and therefore the biggest drift as it might be assumed. This leads to the assumption that the receivers are also not able to estimate accurate how large their clock drift is. A comparison of the receivers plotted dashed and dashed-dotted gives a good insight into how much the receivers clocks can be normally expected to drift over time. Over a period of 600 s their difference in pseudorange to each other changes by approximately 100

km. This equals to a drift against each other of 170 m (560 ns) per second.

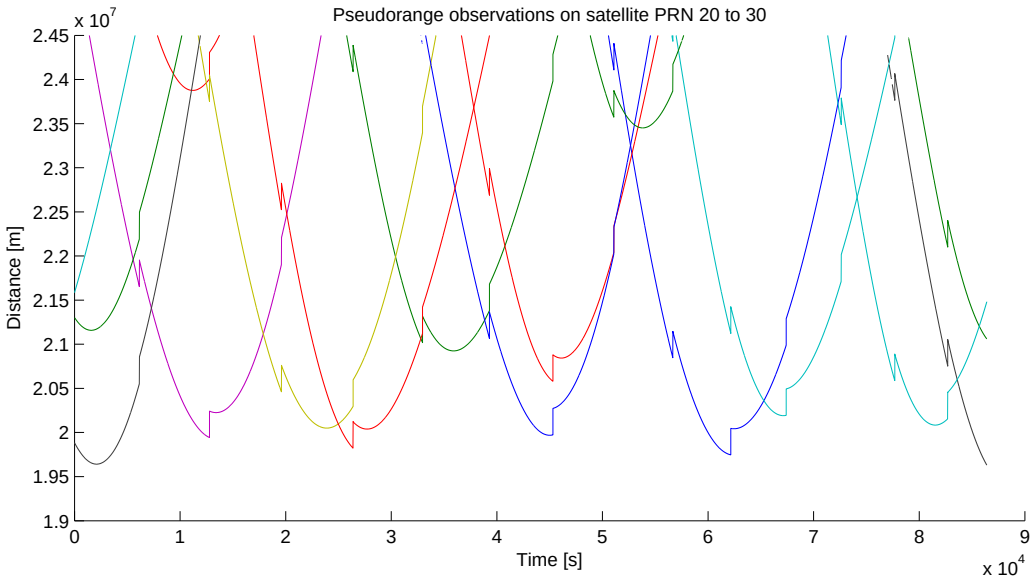


Figure 6.14. Time slip for long term observations

Figure 6.14 depicts a 24 h stationary measurement of 11 satellites with a single receiver. It shows that time slips are not rare events but can happen regularly with the used receivers. It also leads to the assumption that the drift is not constant over time since the jumps are not equally distributed over time. This is why a clock drift state is included in the filters in order to handle these effects.

It is to mention that according to the measurements done for this thesis the used u-blox receivers suffer much more from clock drift and time slips than the ones from NVS. Figure 6.13 and 6.14 both only show u-blox receivers. And it seems also like not all u-blox receivers are drifting equally much. One receiver causes about 15 time slips over 24 h as depicted in Figure 6.14, another only one having the same measurement conditions.

7 Discussion

For absolute positioning the goal of three meter accuracy was achieved in both RMSE and 68th as well as 95th percentile in the results of the test track. This result was also verified during testing by comparing against lane markers in both driving scenarios, freeways and country roads. There is an error of about 1.4 m left in the absolute position which is partly canceled out by the A2R algorithm. This indicates that even thought accurate clock corrections, satellite orbits, and atmospheric compensations were applied with data provided by IGS, they were not able to cancel all corresponding errors. In urban areas with very few and also inaccurate GNSS fixes, the algorithms were able to trace where the vehicle has driven. However accuracy down to the required 3 meters can not be guaranteed in areas with poor GNSS quality. The benefits of applying smoothing are especially visible in dead reckoning since the smoother connects the measurements also backwards in time and therefore corrects some of the incorrect heading angles.

Looking at the different evaluated methods for nonlinear filtering, the Unscented Kalman filter and smoother showed better performance in dead reckoning. This is also supported by the presented theory since it propagates the nonlinearities better. When GNSS data was available, no advantages were observed with the Unscented versions which could depend on the fact that the nonlinearity error never propagates long enough. It was also observed that the Unscented versions are easier to implement than the Extended versions due to the lack of partial derivatives. For a real time application with limited processing power it has to be considered that the UKF implementation required a significantly higher computational effort. A possible improvement for the absolute positioning would be the use of a tightly coupled filter. Due to the similar results in A2R for both nonlinear Kalman filters only one of both was implemented in the relative algorithm. The extended version was chosen because of less computational effort.

Models taking advantage of accelerometer measurements were also tested. However the longitudinal acceleration measurement was much more noisy than the velocity measurement and was therefore not taken into account. A tested model taking the lateral acceleration into account did not provide a better performance than the model described in this thesis.

For relative positioning, the A2R algorithm did not fulfill the 0.5 m accuracy constraint for one of both receivers on the test track and country road. One reason might be that this algorithm cannot ensure the use of the same satellites in both receivers and therefore cannot get the relative accuracy benefits of being in the same area. The tightly coupled relative algorithm fulfills the requirements of 0.5 m on the test track and freeway scenarios where it outperformed the A2R algorithm. For the country road there are problems with several multipath inducted cycle slips causing reinitializations of the filter, which degrades the accuracy. There exist methods for, instead of reinitializing, repairing cycle slips which possibly could provide better

results. One method for single frequency receivers is described in [1].

The major remaining drawback of the relative algorithm is that it requires more visible satellites than the A2R algorithm due to differencing. This degrades the accuracy in areas with bad GNSS reception even more and can be observed by comparing the results for freeway and country road. On the freeway with good reception conditions the relative algorithm was clearly superior which is not the case in the country road. It has to be taken into consideration that all measurements were taken in Sweden where GPS reception is rather bad compared to lower latitude locations due to low GPS satellite elevation angles. For application in areas better suited to the GPS constellation or by taking advantage of several GNSS this drawback can be mitigated.

The relative algorithm required considerably higher effort in tuning and finding working thresholds in order to make it stable. There are thresholds for cycle slip detection and satellite selection, weighting schemes based on C/N_0, the LAMBDA method, varying size of the state vector, etc., which increases the complexity. Therefore it is more difficult to make the algorithm compatible with other receivers and in this project the NVS receiver arrived late. This is the reason why no results could be delivered for the relative algorithm and this receiver. The A2R algorithm is more plug-and-play-like which is a big advantage.

The receiver clock drift and jumps, described in Section 6.4, were one of the challenges using the raw data. A time slip causes all ambiguity estimates to be invalid. That results in a necessary reinitialization of the filter.

The proposed algorithms are well suited for a real time implementation. An important aspect to consider is the amount of information, which has to be shared between both cars. In the current setup, the relative algorithm requires three 50 Hz signals of each vehicle and GNSS raw data which includes four measurements for every visible satellite at 1 Hz. In A2R there would only be the calculated position at the requested update rate to be transmitted.

An approach was taken to include GLONASS satellites in addition to GPS in the A2R algorithm. But several problems were encountered. One is that taking GLONASS into account in RTKLIB requires different clock and orbit correction files. These combined files for GPS and GLONASS are usually published by IGS with more than one week delay, whereas the GPS-only corrections are provided in advance for the orbit predictions and with about one day delay for the clock corrections. Secondly, results based on different correction files are not directly comparable. However the main problem is that including GLONASS satellites often led to exactly the same results as GPS-only. It is probable that RTKLIB is not capable of deep coupling of GPS and GLONASS and instead calculates two separate positions, of which an weighted average is taken. But without deep coupling the advantages of using several GNSS are very limited. Therefore no GLONASS results are taken into account in Chapter 6.

8 Conclusion and outlook

It was possible to fulfill the constraints on this thesis work on accuracy as well as on the price. The absolute position accuracy on the test track given Table 6.1 states that the RMSE as well as the 68th percentile were both approximately at 1.5 m for ERTS and URTS on both receivers, which is much better than the demanded 3 m. Even the 95th percentile was always better than 3 m.

The relative accuracy can be assumed to be better than the asked 0.5 m since the RMSE and 68th percentile were always below this barrier with one exception: in the country roads measurement the RMSE was 0.014 m above the threshold. But since this test was done against the comparably inaccurate radar it can be assumed that the error introduced by the radar was more than 0.014 m.

The deployed GPS receivers are evaluation kits at for approximately 300 € each. For usage on a large scale the positioning modules can be bought for below 150 € (u-blox LEA-6T at the u-blox shop, price for a single piece) or even below 40 € (NVS NV08C-CSM at a big electronic component distributor, price for a single piece). Even with integration and antenna cost the price constraint will be hold.

It can be advantageous to take into account several GNSS instead of just GPS and some receivers like the applied model from NVS are already able to track satellites of several GNSS at the same time. Galileo, COMPASS, and the new generation of GLONASS satellites can be used together with GPS in the presented relative algorithm with only slight modifications in the code. This can be especially helpful in areas where only few satellites are visible such as in high latitude regions or in urban canyons. However it is difficult to use GPS and current FDMA only GLONASS satellites in the same filter due to differences in the observables and there were only very few Galileo and COMPASS satellites available while performing the measurements for this thesis. A possible solution to include also the current generation GLONASS satellites is presented in [1].

Currently nearly only sports utility vehicles (SUV) have 6 DOF IMUs since pitch and roll rate as well as vertical acceleration measurements are important inputs to their stability systems. But since prices on these units drop it can be expected that they will replace 3 DOF IMUs also in other vehicles. This can improve the results in dead reckoning, especially when also moving in vertical direction, which is often the case for tunnels.

In the last years the amount of companies competing in the low-cost single frequency GNSS receiver market has increased and prices have dropped. It might be possible that some of these companies start selling low-cost receivers which can handle multiple frequencies in the next years. Having measurements on two or even more frequencies makes multipath detection and elimination much more reliable. Since multipath is the main problem in short baseline relative positioning the accuracy

can be improved a lot using multiple frequencies in multipath prone environments.

The authors concluding opinion about the evaluated algorithms is that utilizing the raw GNSS measurements in a single filter has the highest potential for relative positioning which the results also point towards. Observing the current developments in technology, it can be assumed that more accurate results can be achieved in the near future using the presented methods.

Bibliography

[1] S. Carcanague. *Low-cost GPS/GLONASS Precise Positioning in Constrained Environment*. PhD thesis, Institut National Polytechnique de Toulouse, Toulouse, 2013.

[2] J. M. Dow, R. E. Neilan, and C. Rizos. The International GNSS Service in a changing landscape of Global Navigation Satellite Systems. *Journal of Geodesy*, 83:191198, DOI: 10.1007/s00190-008-0300-3, 2009.

[3] F. Gustafsson. *Statistical Sensor Fusion*, volume 2:1. Studentlitteratur AB, Lund, 2012.

[4] W. Hedgecock, M. Maroti, J. Sallai, P. Volgyesi, and A. Ledeczi. High-accuracy differential tracking of low-cost GPS receivers. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 221–234. ACM, 2013.

[5] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle. *GNSS Global Navigation Satellite Systems*. Springer-Verlag Wien, 2008.

[6] S. J. Julier and J. K. Uhlman. A New Extension of the Kalman Filter to Nonlinear Systems. *Oxford Press*, 1997.

[7] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME-Journal of Basic Engineering*, 82 (Series D):35–45, 1960.

[8] E. D. Kaplan and C. J. Hegarty. *Understanding GPS: Principles and Applications*. ARTECH HOUSE, 2006.

[9] S. M. Martin. Closely Coupled GPS/INS Relative Positioning For Automated Vehicle Convoys. Master's thesis, Auburn University, 2011.

[10] J. Meguro, T. Murata, J. Takiguchi, Y. Amano, and T. Hashizume. GPS Multipath Mitigation for Urban Area Using Omnidirectional Infrared Camera. *Intelligent Transportation Systems, IEEE Transactions on*, 10(1):22–30, March 2009.

[11] A. Noureldin, T. B. Karamat, and J. Georgy. *Fundamentals of Inertial Navigation, Satellite-based Positioning and their Integration*. Springer Berlin Heidelberg, 2013.

[12] M. Obst, S. Bauer, P. Reisdorf, and G. Wanielik. Multipath detection with 3D digital maps for robust multi-constellation GNSS/INS vehicle localization in urban areas. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 184–190, June 2012.

[13] B. W. Parkinson, J. J. Spilker, P. Axelrad, and P. Enge. *Global Positioning System: Theoy and Applications*, volume 1, chapter 1: GPS Fundamentals by B. W. Parkinson, pages 3–28. American Institue of Aeronautics and Astronautics, 1996.

[14] B. W. Parkinson, J. J. Spilker, P. Axelrad, and P. Enge. *Global Positioning System: Theoy and Applications*, volume 1, chapter 11: GPS Error Analysis by B. W. Parkinson, pages 469–484. American Institue of Aeronautics and Astronautics, 1996.

[15] B. W. Parkinson, J. J. Spilker, P. Axelrad, and P. Enge. *Global Positioning System: Theoy and Applications*, volume 1, chapter 18: Introduction to Relativistic Effects on the Global Positioning System by N. Ashby and J. J. Spilker, pages 623–698. American Institue of Aeronautics and Astronautics, 1996.

[16] B. W. Parkinson, J. J. Spilker, P. Axelrad, and P. Enge. *Global Positioning System: Theoy and Applications*, volume 1, chapter 14: Multipath Effects by M. S. Braasch, pages 547–568. American Institue of Aeronautics and Astronautics, 1996.

[17] B. W. Parkinson, J. J. Spilker, P. Axelrad, and P. Enge. *Global Positioning System: Theoy and Applications*, volume 1, chapter 9: GPS and the Global Navigation Satellite System (GLONASS) by P. Daly and P. N. Misra, pages 243–274. American Institue of Aeronautics and Astronautics, 1996.

[18] European New Car Assessment Programme. Euro NCAP Rating Review. Technical report, European New Car Assessment Programme, 2012.

[19] R. Rajamani. *Vehicle Dynamics and Control*. Springer US, New York, 2006.

[20] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum Likelihood Estimates of Linear Dynamic Systems. *AIAA Journal*, 3(8):1445–1450, 1965.

[21] S. Revnivykh. GLONASS Status and Modernization. Central Research Institute of Machine Building Federal Space Agency, Moscow, 2011.

[22] S. Särkkä. Unscented Rauch-Tung-Striebel Smoother. *Automatic Control, IEEE Transactions on*, 53(3):845–849, April 2008.

[23] S. Särkkä. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[24] G. Seeber. *Satellite Geodesy: foundations, methods, and applications*. de Gruyter, 1993.

[25] T. Takasu. RTKLIB: An Open Source Program Package for GNSS Positioning, 2013.

[26] 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.

[27] D. H. Titterton and J. L. Weston. *Strapdown inertial navigation technology*, volume 17. IEE, Stevenage, 2004.

[28] W. Travis, S. Martin, and D. M. Bevly. Automated short distance vehicle following using a dynamic base RTK system. *International Journal of Vehicle Autonomous Systems*, 9(1):126–141, 2011.

[29] E. A. Wan and R. van der Merwe. The unscented Kalman filter for nonlinear estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pages 153–158, 2000.

[30] D. E. Wells, N. Beck, D. Delikaraoglu, A. Kleusberg, E. J. Krakiwsky, G. Lachapelle, R. B. Langley, M. Nakiboglu, K. P. Schwarz, J. M. Tranquilla, and P. Vaníček. *Guide to GPS Positioning*. Canadian GPS Associates, 1986.

Appendix A

Measurment selection

if $GNSS$ **and** $[(\sigma_{GNSS,A} > \sigma_{max}$ **or** $\Delta z_{GNSS,A} > z_{max})$ **and** $(\sigma_{GNSS,B} > \sigma_{max}$ **or** $\Delta z_{GNSS,B} > z_{max})$ **or** $(v_A < v_{min}$ **and** $v_B < v_{min})$ **or** $(\left|v_{CAN,A} - v_{GNSS,A}\right| > v_{diff,max}$ **and** $\left|v_{CAN,B} - v_{GNSS,B}\right| > v_{diff,max})]$ **then**

$y = \begin{bmatrix} \dot{\varphi}_A & \dot{\varphi}_B & \Delta P & \Delta L \end{bmatrix}^{\mathrm{T}}$

$R = \mathrm{diag}(\sigma^2_{\dot{\varphi}}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\rho_A} + \sigma^2_{\rho_B}, \sigma^2_{\phi_A} + \sigma^2_{\phi_B})$

else if $\overline{GNSS}$ **or** $[(\sigma_{GNSS,A} > \sigma_{max}$ **or** $\Delta z_{GNSS,A} > z_{max})$ **and** $(\sigma_{GNSS,B} > \sigma_{max}$ **or** $\Delta z_{GNSS,B} > z_{max})$ **or** $(v_A < v_{min}$ **and** $v_B < v_{min})$ **or** $(\left|v_{CAN,A} - v_{GNSS,A}\right| > v_{diff,max}$ **and** $\left|v_{CAN,B} - v_{GNSS,B}\right| > v_{diff,max})]$ **then**

$y = \begin{bmatrix} \dot{\varphi}_A & \dot{\varphi}_B \end{bmatrix}^{\mathrm{T}}$

$R = \mathrm{diag}(\sigma^2_{\dot{\varphi}}, \sigma^2_{\dot{\varphi}})$

else if $\sigma_{GNSS,A} > \sigma_{max}$ **or** $\Delta z_{GNSS,A} > z_{max}$ **or** $v_A < v_{min}$ **or** $\left|v_{CAN,A} - v_{GNSS,A}\right| > v_{diff,max}$ **then**

$y = \begin{bmatrix} \varphi_B & \dot{\varphi}_A & \dot{\varphi}_B & \Delta P & \Delta L \end{bmatrix}^{\mathrm{T}}$

$R = \mathrm{diag}(\sigma^2_{\varphi}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\rho_A} + \sigma^2_{\rho_B}, \sigma^2_{\phi_A} + \sigma^2_{\phi_B})$

else if $\sigma_{GNSS,B} > \sigma_{max}$ **or** $\Delta z_{GNSS,B} > z_{max}$ **or** $v_B < v_{min}$ **or** $\left|v_{CAN,B} - v_{GNSS,B}\right| > v_{diff,max}$ **then**

$y = \begin{bmatrix} \varphi_A & \dot{\varphi}_A & \dot{\varphi}_B & \Delta P & \Delta L \end{bmatrix}^{\mathrm{T}}$

$R = \mathrm{diag}(\sigma^2_{\varphi}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\rho_A} + \sigma^2_{\rho_B}, \sigma^2_{\phi_A} + \sigma^2_{\phi_B})$

else

$y = \begin{bmatrix} \varphi_A & \varphi_B & \dot{\varphi}_A & \dot{\varphi}_B & \Delta P & \Delta L \end{bmatrix}^{\mathrm{T}}$

$R = \mathrm{diag}(\sigma^2_{\varphi}, \sigma^2_{\varphi}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\dot{\varphi}}, \sigma^2_{\rho_A} + \sigma^2_{\rho_B}, \sigma^2_{\phi_A} + \sigma^2_{\phi_B})$

end if

Algorithm 2: Measurement update

Appendix B

Speed Profiles

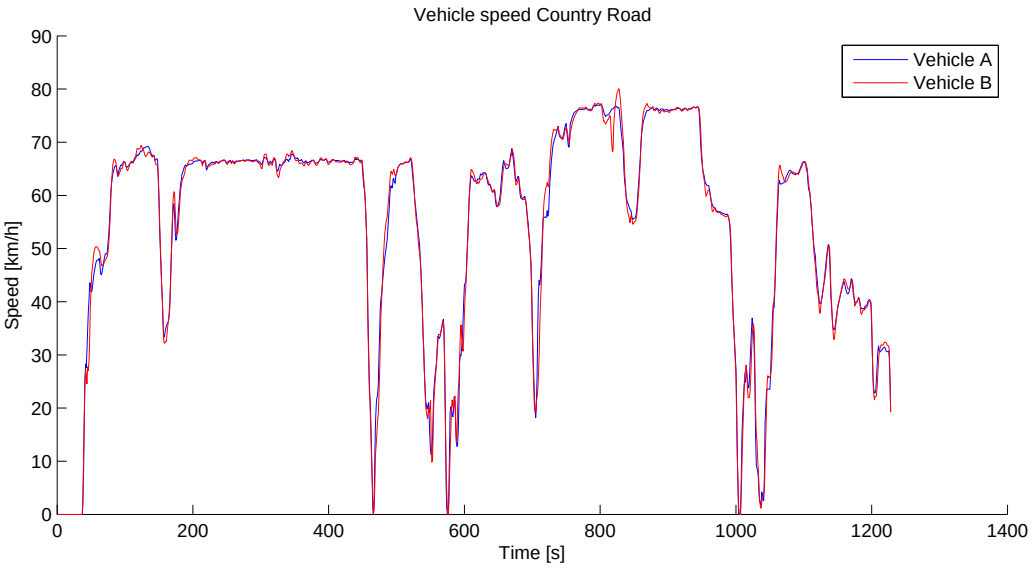


Figure B.1. Vehicle speed for the Country Road measurement

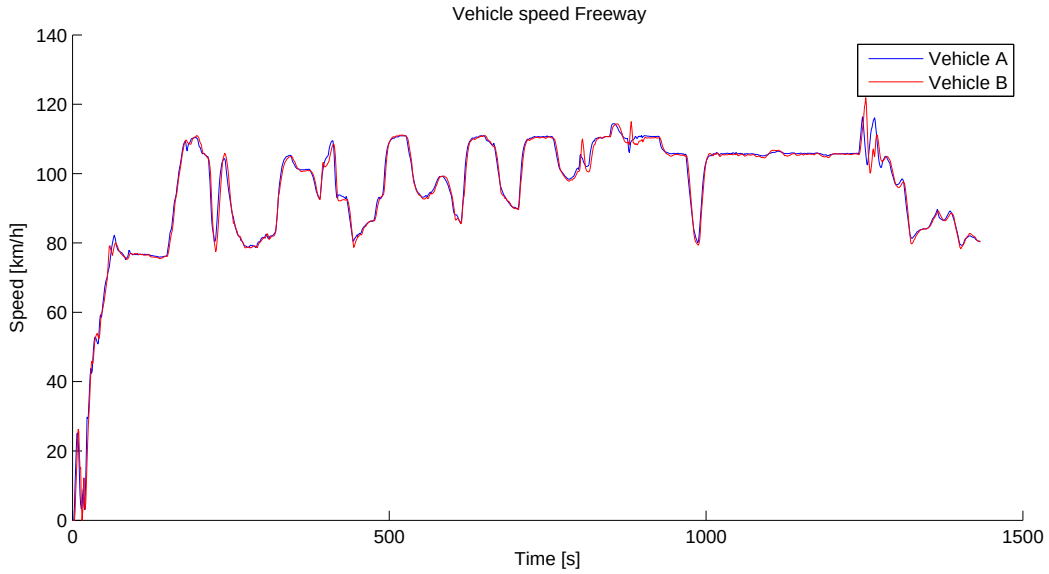


Figure B.2. Vehicle speed for the Freeway measurement

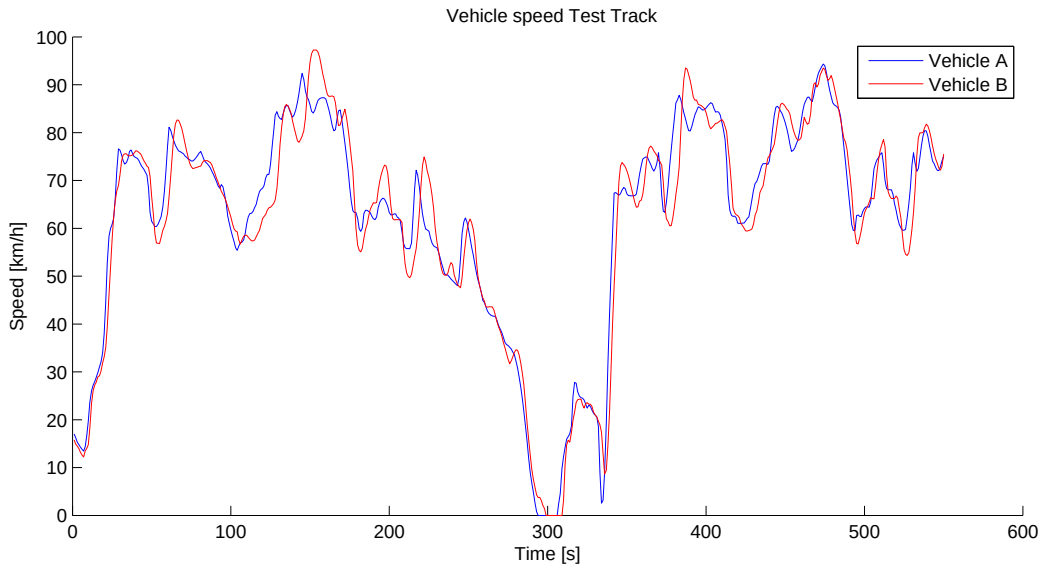


Figure B.3. Vehicle speed for the Test Track measurement

Appendix C

RTKLIB settings

```
# rtkpost options (2014/04/22 08:13:44, v.2.4.2)

pos1-posmode       =ppp-kine   # (0:single,1:dgps,2:kinematic,3:static,4:movingbase,
                                 5:fixed,6:ppp-kine,7:ppp-static)
pos1-frequency     =l1         # (1:l1,2:l1+l2,3:l1+l2+l5,4:l1+l2+l5+l6,5:l1+l2+l5+l6+l7)
pos1-soltype       =combined   # (0:forward,1:backward,2:combined)
pos1-elmask        =15         # (deg)
pos1-snrmask_r     =on         # (0:off,1:on)
pos1-snrmask_b     =off        # (0:off,1:on)
pos1-snrmask_L1    =40,40,40,40,40,40,40,40,40
pos1-snrmask_L2    =0,0,0,0,0,0,0,0,0
pos1-snrmask_L5    =0,0,0,0,0,0,0,0,0
pos1-dynamics      =on         # (0:off,1:on)
pos1-tidecorr      =on         # (0:off,1:on)
pos1-ionoopt       =ionex-tec  # (0:off,1:brdc,2:sbas,3:dual-freq,4:est-stec,5:ionex-tec,
                                 6:qzs-brdc,7:qzs-lex,8:vtec_sf,9:vtec_ef,10:gtec)
pos1-tropopt       =est-ztd    # (0:off,1:saas,2:sbas,3:est-ztd,4:est-ztdgrad)
pos1-sateph        =precise    # (0:brdc,1:precise,2:brdc+sbas,3:brdc+ssrapc,4:brdc+ssrcom)
pos1-posopt1       =on         # (0:off,1:on)
pos1-posopt2       =on         # (0:off,1:on)
pos1-posopt3       =on         # (0:off,1:on)
pos1-posopt4       =on         # (0:off,1:on)
pos1-posopt5       =on         # (0:off,1:on)
pos1-exclsats      =           # (prn ...)
pos1-navsys        =3          # (1:gps+2:sbas+4:glo+8:gal+16:qzs+32:comp)
pos2-armode        =continuous # (0:off,1:continuous,2:instantaneous,3:fix-and-hold,4:ppp-ar)
pos2-gloarmode     =on         # (0:off,1:on,2:autocal)
pos2-arthres       =3
pos2-arlockcnt     =0
pos2-arelmask      =0          # (deg)
pos2-arminfix      =10
pos2-elmaskhold    =0          # (deg)
pos2-aroutcnt      =5
pos2-maxage        =30         # (s)
pos2-syncsol       =off        # (0:off,1:on)
pos2-slipthres     =0.05       # (m)
pos2-rejionno      =30         # (m)
pos2-rejgdop       =30
pos2-niter         =1
```

```
pos2-baselen       =0          # (m)
pos2-basesig       =0          # (m)
out-solformat      =llh        # (0:llh,1:xyz,2:enu,3:nmea)
out-outhead        =on         # (0:off,1:on)
out-outopt         =on         # (0:off,1:on)
out-timesys        =gpst       # (0:gpst,1:utc,2:jst)
out-timeform       =hms        # (0:tow,1:hms)
out-timendec       =3
out-degform        =deg        # (0:deg,1:dms)
out-fieldsep       =
out-height         =ellipsoidal # (0:ellipsoidal,1:geodetic)
out-geoid          =internal   # (0:internal,1:egm96,2:egm08_2.5,3:egm08_1,4:gsi2000)
out-solstatic      =all        # (0:all,1:single)
out-nmeaintv1      =0          # (s)
out-nmeaintv2      =0          # (s)
out-outstat        =off        # (0:off,1:state,2:residual)
stats-eratio1      =100
stats-eratio2      =100
stats-errphase     =0.003      # (m)
stats-errphaseel   =0.003      # (m)
stats-errphasebl   =0          # (m/10km)
stats-errdoppler   =10         # (Hz)
stats-stdbias      =30         # (m)
stats-stdiono      =0.03       # (m)
stats-stdtrop      =0.3        # (m)
stats-prnaccelh    =10         # (m/s^2)
stats-prnaccelv    =10         # (m/s^2)
stats-prnbias      =0.0001     # (m)
stats-prniono      =0.001      # (m)
stats-prntrop      =0.0001     # (m)
stats-clkstab      =5e-12      # (s/s)
ant1-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant1-pos1          =90         # (deg|m)
ant1-pos2          =0          # (deg|m)
ant1-pos3          =-6335367.6285 # (m|m)
ant1-anttype       =
ant1-antdele       =0          # (m)
ant1-antdeln       =0          # (m)
ant1-antdelu       =0          # (m)
ant2-postype       =llh        # (0:llh,1:xyz,2:single,3:posfile,4:rinexhead,5:rtcm)
ant2-pos1          =90         # (deg|m)
ant2-pos2          =0          # (deg|m)
ant2-pos3          =-6335367.6285 # (m|m)
ant2-anttype       =
ant2-antdele       =0          # (m)
ant2-antdeln       =0          # (m)
ant2-antdelu       =0          # (m)
misc-timeinterp    =off        # (0:off,1:on)
misc-sbasatsel     =0          # (0:all)
misc-rnxopt1       =
misc-rnxopt2       =
file-satantfile    =
file-rcvantfile    =
file-staposfile    =
file-geoidfile     =
```

```
file-ionofile      = 'insert corresponding IONEX file'
file-dcbfile       =
file-eopfile       =
file-blqfile       =
file-tempdir       =
file-geexefile     =
file-solstatfile   =
file-tracefile     =
```