Institute for
Communications
and Navigation

Low-cost GPS-based Compass with Reliable Ambiguity Resolution and Cycle Slip Correction

Jane Jean Kiam

Institute for Communications and Navigation

Prof. Dr. Christoph Günther

Supervised by Dr.-Ing. Patrick Henkel

September 16, 2013

Institute for Communications and Navigation
Univ.-Prof. Dr. sc. nat. Christoph Günther

Technische Universität München
Theresienstrasse 90
80333 Munich

Submitted for the degree *Master EI*
at the faculty *Elektrotechnik und Informationstechnik*
at *Technische Universität München (TUM)*

Email: jane.kiam@anavs.de

Layout by LaTeX 2_ε

Acknowledgements

Million thanks to Dr. Patrick Henkel, who spent an enormous amount of time to coach me and help me through the thesis. You have shown me creativity and patience. You have guided me. I would like to thank you for the recommendation for the scholarship of the *Centre de coopération universitaire franco-bavarois*.

Thanks to Juan Manuel Cárdenas, Naoya Oku and Philipp Berthold from Advanced Navigation Solutions (ANAVS). I have never enjoyed team work as much as with you.

Thanks to Prof. Dr. Christoph Günther, Zhibo Wen and Kaspar Giger from the Institute for Communications and Navigation at the Technische Universität München (TUM). You have taught me to love navigation.

Thanks to Télécom Bretagne, Brest, France. The Télécom Bretagne-TUM double-degree program has been a great opportunity and experience. You have educated me well.

Thanks to Andreas Wittmann. You make me happy.

Last, but not least, thanks to Teng Hoon Kiam, Wan Hwa Tan and Jane Yee Kiam. You have always stood by me.

Contents

LIST OF SYMBOLS, NOMENCLATURE, OR ABBREVIATIONS

0_{mn}	$m \times n$ matrix with entries 0
1_{mn}	$m \times n$ matrix with entries 1
$\bar{\rho}_r^k$	smoothed code measurement from receiver r with respect to satellite k
$\check{N}_{mn}^{kl}$	estimated fixed integer ambiguities of a double difference measurement
$\mathit{\Delta} N$	column vector of number of cycle slips of double difference measurements
$\mathit{\Delta} N_{\text{pred}}$	column vector of predicted number of cycle slips of double difference measurements
$\mathit{\Delta} N_{mn}^{kl}$	Number of cycle slips in the phase double difference measurement φ_{mn}^{kl}
η	noise associated with code measurement or with $\mathit{\Psi}$
$\hat{N}_{mn}^{kl}$	estimated float ambiguities of a double difference measurement
$\mathit{\Psi}$	a column vector stacking double difference phase and code measurements
$\mathit{\Psi}_{\text{fixed}}$	a column vector stacking fixed double difference phase and code measurements
ρ_r^k	code measurement from receiver r with respect to satellite k
ρ_{mn}^{kl}	a double difference code measurement between receiver m and n and between satellite k and l
$\tilde{\varphi}_{\text{dd,fixed}}$	notation used in the context of cycle slip correction. It denotes the fixed phase double difference measurement with cycle slip corrected.
$\tilde{H}_{\text{geo}}$	geometry matrix of φ_{dd} or ρ_{dd}
ε	noise associated with phase measurement
$\varphi_{\text{mn,fixed}}^{kl}$	a fixed double difference phase measurement between receiver m and n and between satellite k and l scaled with $\frac{\lambda}{2\pi}$ unless stated otherwise
$\varphi_{\text{dd,fixed}}$	a column vector stacking fixed double difference phase measurements. The phase measurements are scaled with with $\frac{\lambda}{2\pi}$ unless stated otherwise
φ_{dd}	a column vector stacking double difference phase measurements. The phase measurements are scaled with with $\frac{\lambda}{2\pi}$ unless stated otherwise
φ_{mn}^{kl}	a double difference phase measurement between receiver m and n and between satellite k and l scaled with $\frac{\lambda}{2\pi}$ unless stated otherwise
φ_r	a column vector containing carrier-phase measurements from different satellites scaled with $\frac{\lambda}{2\pi}$ unless stated otherwise
φ_r^k	carrier-phase measurement from receiver r with respect to satellite k scaled with $\frac{\lambda}{2\pi}$ unless stated otherwise
$\vec{b}_{mn}$	baseline vector pointing from receiver n to receiver m

$\vec{e}_r^k$	normalized line-of-sight vector directing from satellite k to receiver r
A	a mapping matrix of the double difference ambiguity parameters to the measurement Ψ
A_{cs}	a mapping matrix containing $\lambda/2$ as diagonal elements
A_φ	a mapping matrix of the double difference ambiguity parameters to the double difference phase measurement φ
b	simplified version of $\vec{b}_{mn}$ which does not state the order of the receivers
b_{L}	baseline vector in the local ENU frame of b_{mn}
H	concatenation of H_{geo} and A
H_{geo}	geometry matrix of Ψ
H_{L}	transformed $\tilde{H}_{\text{geo}}$, H, $\tilde{H}_{\text{geo}}$ in local ENU frame
I	identity matrix
m_ρ	multipath of code measurement
m_φ	multipath of phase measurement
N	column vector stacking double difference ambiguities
N_r^k	integer ambiguity of phase measurement received by receiver r with respect to satellite k
n_{ep}	number of epochs needed for initialization
R_{L}	the transformation matrix from ECEF-frame to local ENU-frame
t_n	time at epoch n
CSC	cycle slip correction
CSD	cycle slip detection
DD	double difference
ECEF	Earth-Centered, Earth-Fixed Cartesian coordinate system with the origin defined at the center of mass of the Earth
ENU	local East-North-Up coordinate frame
MAP	Maximum A Posteriori probability
PAD	position and attitude determination
WSSE	Weighted Sum Squared Error

1. Introduction

Attitude determination can be performed with inertial sensors, magnetometers and satellite navigation. Inertial sensors have three advantages: They are very robust, show a very low noise level, and provide measurements at a rate of 100 Hz already for low-cost devices. However, accelerometers and gyroscopes need an initialization and a careful calibration to remove the drift and scaling factors [1]. Magnetometers provide a heading information already without the need of an initialization. However, one has to be sufficiently far away from the magnetic poles. In Munich, the *magnetic declination* is only 2.4°. Moreover, magnetometers are extremely sensitive to ferromagnetic materials such that they can be hardly used in automotive applications [2].

Satellite navigation also enables precise attitude determination by performing differential carrier phase positioning between two or three GNSS receivers. However, carrier phases are periodic and integer ambiguities have to be resolved to fully benefit from the precise carrier phase measurements. In 1993, Teunissen invented the Least-squares Ambiguitiy Decorrelation Adjustmenet (LAMBDA) method to estimate the double difference carrier phase integer ambiguities [3]. The LAMBDA method is today widely used for RTK and differential carrier phase positioning over short baselines.

Henkel and Günther derived optimized multi-frequency linear combinations in [4] and [5], that increase the ambiguity discrimination and thereby simplify the integer ambiguity resolution. In [6] and [7], Teunissen developed a constrained LAMBDA method to increase the success rate by including *a priori* information on the baseline *length*. Henkel et al. further extended the constrained ambiguity fixing in [8] by using *soft* instead of *hard* a priori information (MAP estimation).

Carrier phase measurement can be tracked with millimeter to centimeter-level accuracy even with low-cost GPS receivers like u-box LEA 6T and is, thus, the key to accurate attitude (heading, pitch, roll angle) determination. Using a baseline vector derived from carrier phase double difference measurements to determine the heading of a vehicle offers more flexibility than a standalone GPS receiver: heading can be determined even when the vehicle is stationary. Furthermore, the true heading of the vehicle can be determined when the orientation of the vehicle deviates from its moving direction, which can help to detect for example the drift of a car.

However, there are three challenges to be considered with low-cost GPS receivers: The first challenge is code multipath, which is typically at least 10 m for low-cost GPS receivers even for open-sky conditions. This increased code multipath affects especially our integer ambiguity resolution, which typically takes 0.5 to 3 minutes. We exclude measurement epochs of increased multipath by carefully analyzing the code residuals. Once the ambiguities are fixed, we perform the coasting solely based on the carrier phases.

The second challenge affects cycle slips. While geodetic receivers also experience cycle slips, cycle slips do not occur so often and involve only a whole cycle slip. Low-cost receivers however experience cycle slips very frequently. Furthermore, cycle slips occur also as half cycle slips and affect multiple satellites simultaneously. We use an elevation mask of 20° as satellites of lower elevation are often affected by consecutive cycle slips. One option for cycle slip detection (CSD) and correction (CSC) would be to use an inertial sensor. In this thesis, we follow an alternative option, i.e. we use a Maximum A Posteriori Probability (MAP) estimator, which uses *a priori* information on the baseline (e.g. its length and the absolute vehicle velocity) for CSD and CSC.

The third aspect is caused by the oscillator, which typically shows clock offsets in the order of milliseconds and, thus, requires a correction for the satellite movement within the time of the differential receiver clock offset. Tab. 1.1 summarizes the critical aspects of low-cost GPS receivers and our approach to overcome these issues.

Challenge	Our solution
Code multipath: 10 m even in good environments	use of more epochs for ambiguity fixing, exclusion of measurement epochs, attitude derived solely from DD carrier phases after initial integer ambiguity resolution
Cycle slips: very frequent, also affecting multiple satellites simultaneously	$20°$ elevation mask, MAP estimator, which uses DD phase measurements and baseline a priori information
Oscillator: clock offset of milliseconds	correction of satellite movement within differential receiver clock offset

Table 1.1: Challenges of low-cost GPS receivers and our approach to overcome these issues.

In this thesis, the following algorithms were developed for ANAVS PAD system:

1) Extended synchronization correction
   - Transformation of synchronization correction in case of change of reference satellite
   - Precise extrapolation of synchronization correction over 50 epochs
2) Improved float solution: elimination of measurement epochs with high phase and code multipath
3) Cascaded cycle slip detection and correction:
   - Cycle slip detection and correction based on triple difference phase measurements for *low* dynamics, and MAP estimator with baseline a priori information derived from baseline length and code-based vehicle velocity vector *if* velocity is sufficiently high and smoothed code residuals are sufficiently low
   - A posteriori cycle slip detection and correction based on remaining phase residuals

The algorithms were tested during extensive measurement campaigns in Garching, Nymphenburg, Königsplatz, ESA/AZO, Starnberger See and Wolfsburg (VW). The last chapter of this work provides a thorough description of the complete attitude determination system. Detailed flow charts illustrate how the modules are ordered, how the parameters are transferred from one module to another and which algorithm is implemented in each module by referring the reader to a previous chapter/ section. With the explanation given in the last chapter, this thesis also serves as a handbook to those who wish to work on ANAVS PAD system.

2. Absolute Position Determination

A GPS receiver typically acquires signals from 7 to 12 visible satellites and tracks the code and carrier phases by a *Delay Locked Loop* (DLL) and a *Phase Locked Loop* (PLL). The DLL includes a correlator, a discriminator, a loop filter, a code numerically controlled oscillator (NCO), and a code generator to track the signal traveltime (and, thus, the pseudorange). Similarly, the PLL includes a phase NCO, which is driven by the difference of the measured phase and the oscillator generated phase. As the carrier phase is periodic with 2π, an integer ambiguity has to be resolved for each satellite to obtain a range information from the tracked phases.

We model the pseudorange as described by Henkel, Cardenas, Giorgi and Günther in [8] as:

$$\begin{aligned}\rho_r^k(t+\delta\tau_r) &= \|\vec{x}_r(t+\delta\tau_r) - \vec{x}^k(t+\delta\tau_r - \mathit{\Delta}\tau_r^k)\| + c\left(\delta\tau_r(t) - \delta\tau^k(t+\delta\tau_r - \mathit{\Delta}\tau_r^k)\right) \\ &\quad + I_r^k(t+\delta\tau_r) + T_r^k(t+\delta\tau_r) + m_{\rho_r}^k(t+\delta\tau_r) + b_r + b^k + \eta_r^k(t+\delta\tau_r), \qquad (2.1)\end{aligned}$$

with the receiver position $\vec{x}_r$, the satellite position $\vec{x}^k$, the speed of light c in vacuum, the slant ionospheric delay I_r^k varying between a few up to several tens of metres, the slant tropospheric delay T_r^k being 2 m in zenith direction, the code multipath error $m_{\rho_r}^k$, the receiver code bias b_r, the satellite code bias b^k and the code noise η_r^k. The measurement was explicitly modeled at time $t+\delta\tau_r$ instead of t to take the *satellite movement* during the receiver clock offset into account. All time variables involved are listed out, with the GPS system time t at the time of signal reception, the receiver clock offset $\delta\tau_r$, the satellite clock offset $\delta\tau^k$ and the delay $\mathit{\Delta}\tau_r^k$ between the transmission time and the received time.

The carrier phase measurement is similarly modeled with the ambiguities and phase biases as additional terms and the ionospheric error subtracted instead of added to the range, i.e.

$$\begin{aligned}\frac{\lambda}{2\pi}\varphi_r^k(t+\delta\tau_r) &= \|\vec{x}_r(t+\delta\tau_r) - \vec{x}^k(t+\delta\tau_r - \mathit{\Delta}\tau_r^k)\| + c\left(\delta\tau_r(t+\delta\tau_r) - \delta\tau^k(t+\delta\tau_r - \mathit{\Delta}\tau_r^k)\right) \\ &\quad - I_r^k(t+\delta\tau_r) + T_r^k(t+\delta\tau_r) + \lambda N_r^k + m_{\varphi_r}^k(t+\delta\tau_r) + \beta_r + \beta^k + \varepsilon_r^k(t+\delta\tau_r). \qquad (2.2)\end{aligned}$$

2.1 Absolute Positioning with Carrier Smoothed Code Measurements

The pseudorange measurement is provided by the *Delay Locked Loop* (DLL) and the phase measurement by the *Phase Locked Loop* (PLL). The noise performance of both tracking loops was derived by Günther in [9] and is given for the DLL by

$$\sigma_\rho = l_c \cdot \sqrt{\frac{d \cdot B_{\mathrm{DLL}}}{2P/N_0}(1 + \frac{2}{E/N_0(2-d)})}, \qquad (2.3)$$

and for the PLL by

$$\sigma_{\lambda\varphi} = \frac{\lambda}{2\pi} \cdot \sqrt{\frac{B_{\mathrm{PLL}}}{2P/N_0}(1 + \frac{1}{2E/N_0})}, \qquad (2.4)$$

where P is the signal power, E is the accumulated energy during correlation and N_0 is the noise power spectral density. B_{DLL} and B_{PLL} are respectively the bandwidths of the loops and d is the correlator-spacing in chips of the DLL. The ratio of the standard deviations of the code and phase tracking errors is dominated by the chip length and the carrier wavelength. As $l_{\mathrm{c}} = 300\ \mathrm{m}$ and $\lambda = 19.03\ \mathrm{cm}$, the phase can be tracked 1500 more accurately than the code measurements.

Carrier smoothing is a popular approach to reduce the code noise and multipath with the help of the low noise phase measurements without the need of an absolute carrier phase integer ambiguity resolution. Fig. 2.1 shows the functional diagram of carrier smoothing, where the code minus carrier phase measurements (i.e. code multipath and code noise, but also the ionospheric delay) are low-pass filtered.

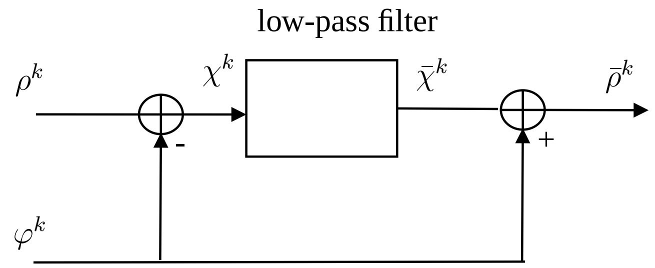


Figure 2.1: Hatch filter: The code minus phase measurements are low-pass filtered.

The carrier smoothed pseudorange measurement of satellite k at epoch t is given by

$$\bar{\rho}^k(t) = \frac{1}{\tau}\rho^k(t) + (1 - \frac{1}{\tau})\left(\bar{\rho}^k(t-1) + (\varphi^k(t) - \varphi^k(t-1))\right), \tag{2.5}$$

where τ is the smoothing time constant.

The carrier smoothing is typically initialised with $\bar{\rho}^k(1) = \rho^k(1)$. Note that carrier smoothing should be re-initialized always when cycle slips are detected but can not be reliably corrected.

Günther derived in [9] the variance of the smoothed code minus phase noise $\bar{\chi}$ for white Gaussian code and phase measurements as

$$\sigma^2_{\bar{\chi}} = \left(\frac{1}{2\tau - 1} + \frac{2\tau - 2}{2\tau - 1}(1 - \frac{1}{\tau})^{2(t-1)}\right) \cdot (\sigma^2_\varphi + \sigma^2_\rho), \tag{2.6}$$

where σ^2_φ and σ^2_ρ are the variances of the phase and code noises. The covariance between the smoothed code minus phase measurement and the phase measurement of the current epoch follows from Eq. (2.5) as

$$\sigma_{\bar{\chi}\varphi} = -\frac{1}{\tau}\sigma^2_\varphi, \tag{2.7}$$

Thus, the variance of the smoothed code measurement is obtained as

$$\begin{aligned} \sigma^2_{\bar{\rho}} &= \sigma^2_{\bar{\chi}} + 2\sigma_{\bar{\chi}\varphi} + \sigma^2_\varphi \\ &= \left(\frac{1}{2\tau - 1} + \frac{2\tau - 2}{2\tau - 1}(1 - \frac{1}{\tau})^{2(t-1)}\right) \cdot (\sigma^2_\varphi + \sigma^2_\rho) - 2\frac{1}{\tau}\sigma^2_\varphi + \sigma^2_\varphi. \end{aligned} \tag{2.8}$$

For large t, the variance of the smoothed code measurements converges to

$$\begin{aligned}\lim_{t\to\infty}(\sigma^2_{\bar{\rho}}) &= \frac{1}{2\tau-1}\cdot(\sigma^2_\varphi+\sigma^2_\rho)-2\frac{1}{\tau}\sigma^2_\varphi+\sigma^2_\varphi \\ &\sim \frac{1}{2\tau-1}\cdot\sigma^2_\rho+\sigma^2_\varphi. \end{aligned} \tag{2.9}$$

We choose $\tau = 600$ epochs, which results in a reduction of the standard deviation of the code noise by a factor of $\sqrt{2\tau-1} \approx 35$.

2.1.1 Iterative least-squares estimation of absolute receiver position and clock offset

The single-frequency carrier smoothed code measurements of K visible satellites are rearranged such that all known parameters (satellite position and clock offsets, atmospheric delays) are brought to the left side and the unknown parameters to the right side of the equation, i.e.

$$\begin{pmatrix} \bar{\rho}^1_r + \vec{e}^{\,1}_r\vec{x}^{\,1} + c\delta\tau^1 - \hat{I}^1_r - \hat{T}^1_r \\ \vdots \\ \bar{\rho}^K_r + \vec{e}^{\,K}_r\vec{x}^{\,K} + c\delta\tau^K - \hat{I}^K_r - \hat{T}^K_r \end{pmatrix} = H_{\bar{\rho}}\begin{pmatrix} \vec{x}_r \\ c\delta\tau_r \end{pmatrix} + \begin{pmatrix} \bar{\eta}^1 \\ \vdots \\ \bar{\eta}^K \end{pmatrix}, \tag{2.10}$$

where the ionospheric slant delays I^k_r could be partially corrected using the Klobuchar model or EGNOS corrections, and the tropospheric slant delays T^k_r are typically described through a blind model. The absolute receiver position and clock offsets are determined by least-squares estimation as

$$\begin{pmatrix} \hat{\vec{x}}_r \\ c\delta\hat{\tau}_r \end{pmatrix} = \min_{\vec{x}_r, c\delta\tau_r} \left\| \begin{pmatrix} \bar{\rho}^1_r + \vec{e}^{\,1}_r\vec{x}^{\,1} + c\delta\tau^1 - \hat{I}^1_r - \hat{T}^1_r \\ \vdots \\ \bar{\rho}^K_r + \vec{e}^{\,K}_r\vec{x}^{\,K} + c\delta\tau^K - \hat{I}^K_r - \hat{T}^K_r \end{pmatrix} - H_{\bar{\rho}}\begin{pmatrix} \vec{x}_r \\ c\delta\tau_r \end{pmatrix} \right\|^2_{\Sigma^{-1}_{\bar{\rho}}}. \tag{2.11}$$

As the geometry matrix $H_{\bar{\rho}}$ is itself dependent on the unknown receiver position, an iterative approach is required. The Gauss-Newton algorithm of (1) is used to determine the absolute receiver position and clock offset.

This algorithm can also be applied to determine the float solution of the baseline and ambiguity parameters as described in chapter 4.

Fig. 2.2 shows the track of a test drive in front of the ESA / AZO building in Oberpfaffenhofen, Germany. The track includes several turns and three sections with reversing at $97-104$ s, $140-160$ s and $180-185$ s. As the track was close to a high building, the code measurements were affected by substantial multipath. Even with severe multipath, we were still able to determine the absolute positions accurately.

Fig. 2.3 illustrates residuals of pure code and carrier-smoothed code measurement during absolute position estimation. We observe severe multipath of a few tens of meter in the case where pure code is used for absolute positioning. With carrier smoothing, multipath is substantially reduced. Remaining residuals of a few meters are due to unmodeled atmospheric delays and satellite biases.

Algorithm 1 Iterative least-squares solution

```
Input: $\bar{\rho}_r^k, \Sigma_{\bar{\rho}}^{-1}, \quad \vec{x}^{\,k}, \delta\tau^k \quad \forall k$
Output: $\hat{\vec{x}}_r, c\delta\hat{\tau}_r, H_{\bar{\rho}}$
$\hat{\vec{x}}_r^{(0)} = 0$    ▷ Initialization of receiver position and clock offset
$\delta\hat{\tau}_r^{(0)} = 0$
for $i = 1 \to 7$ do
    for $k = 1 \to K$ do
        $r_r^{(k,i)} = \bar{\rho}_r^k - \|\hat{\vec{x}}_r^{(i-1)} - \vec{x}^{\,k}\| - c(\delta\hat{\tau}_r^{(i-1)} - \delta\tau^k)$    ▷ Smoothed pseudorange residual
        $\vec{e}^{\,(k,i)} = \frac{\hat{\vec{x}}_r^{(i-1)} - \vec{x}^k}{\|\hat{\vec{x}}_r^{(i-1)} - \vec{x}^k\|}$
    end for
    $H_{\bar{\rho}}^{(i)} = \begin{pmatrix} (\vec{e}^{\,(1,i)})^{\mathrm{T}} & 1 \\ \vdots & \vdots \\ (\vec{e}^{\,(K,i)})^{\mathrm{T}} & 1 \end{pmatrix}$
    $\begin{pmatrix} \hat{\vec{x}}_r^{(i)} \\ c\delta\hat{\tau}_r^{(i)} \end{pmatrix} = \begin{pmatrix} \hat{\vec{x}}_r^{(i-1)} \\ c\delta\hat{\tau}_r^{(i-1)} \end{pmatrix} + \left((H_{\bar{\rho}}^{(i)})^{\mathrm{T}} \Sigma_{\bar{\rho}}^{-1} H_{\bar{\rho}}^{(i)}\right)^{-1} (H_{\bar{\rho}}^{(i)})^{\mathrm{T}} \Sigma_{\bar{\rho}}^{-1} \begin{pmatrix} r_r^{(1,i)} \\ \vdots \\ r_r^{(K,i)} \end{pmatrix}$
end for
```

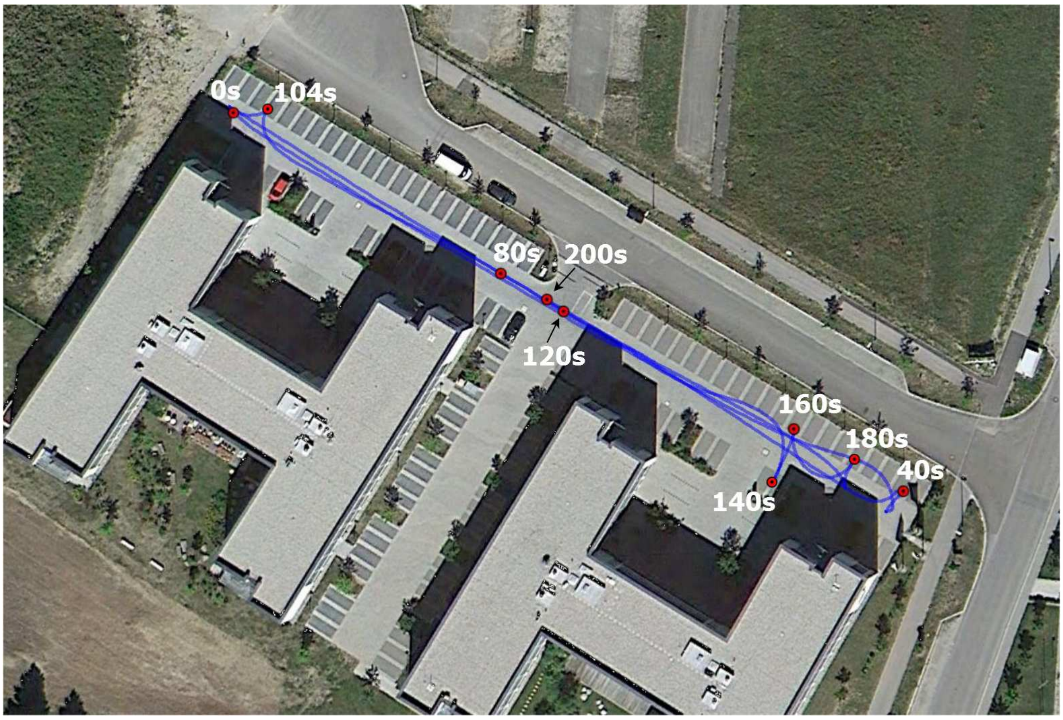


Figure 2.2: Track of car drive in front of the ESA/ AZO building in Oberpfaffenhofen. The track includes three reverse drives at 97 - 104 s, 140 - 160 s and 180 - 185 s.

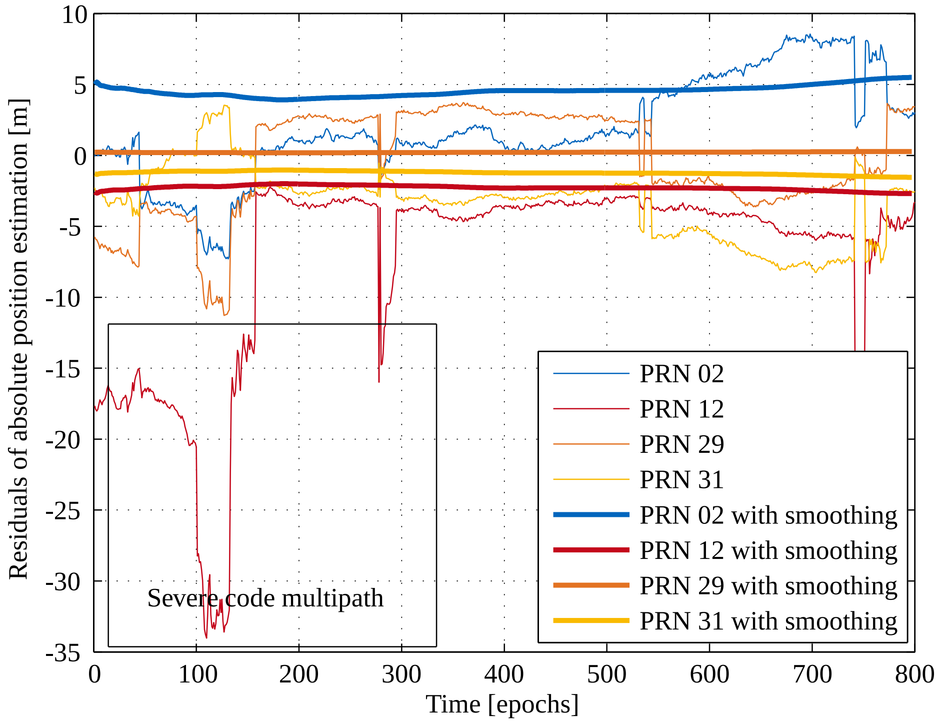


Figure 2.3: Residuals of absolute position estimation at ESA-AZO. Thicker lines show the residuals of smoothed code measurement during absolute position estimation while finer lines show the residuals of pure code measurement during absolute position estimation.

3. Attitude determination: measurement model

The relative position (and thereby the attitude) information is completely included in the receiver-satellite range in Eq. (2.1) and (2.2). All other terms are nuisance parameters that we wish to eliminate by differencing or to suppress by correction data. Correction data are provided for example by SAPOS[1], the IGS, the DGPS System and ASCOS. These services provide correction parameters which are valid over a local area since ionospheric and tropospheric delays are not sensitive to the variation in geographical position over a few meters. However, ambiguities in carrier phases are different for each receiver and, thus, cannot be eliminated by using a reference station. The resolution of integer ambiguities is required and our approach is explained in chapter 5.

3.1 Single difference carrier phase measurements

The single difference between the carrier phase measurement of satellite k observed by receivers 1 and 2 is given by

$$\begin{aligned} \frac{\lambda}{2\pi}\varphi_{12}^k = \frac{\lambda}{2\pi}(\varphi_1^k - \varphi_2^k) \quad &= \quad (r_1^k - r_2^k) - (I_1^k - I_2^k) + (T_1^k - T_2^k) + c(\delta\tau_1 - \delta\tau_2) \\ &\quad +\lambda(N_1^k - N_2^k) + (m_{\varphi,1}^k - m_{\varphi,2}^k) + (\beta_1 - \beta_2) + (\varepsilon_1^k - \varepsilon_2^k). \end{aligned} \tag{3.1}$$

Satellite clock offsets as well as satellite phase biases can be considered stable over the time of the differential receiver clock offset and, therefore, are eliminated by the differencing between two receivers. We simplify the notation by writing the differencing as $(\cdot)_{12}$, i.e.

$$\frac{\lambda}{2\pi}\varphi_{12}^k = r_{12}^k - I_{12}^k + T_{12}^k + c\delta\tau_{12} + \lambda N_{12}^k + m_{\varphi,12}^k + \beta_{12} + \varepsilon_{12}^k. \tag{3.2}$$

For attitude determination, the baseline length varies between 1 m (car application) and 20 m (ship application) and therefore, ionospheric and tropospheric delays are also cancelled by the differencing between two receivers. Hence, Eq. (3.2) can be further simplified to

$$\frac{\lambda}{2\pi}\varphi_{12}^k = r_{12}^k + c\delta\tau_{12} + \lambda N_{12}^k + m_{\varphi,12}^k + \beta_{12} + \varepsilon_{12}^k. \tag{3.3}$$

3.2 Double difference code and carrier phase measurements

We observe that the single difference model described in the previous section still contains the receiver clock offsets and receiver phase biases which can be eliminated in a similar manner by differencing measurements from two satellites. The double difference (DD) measurement between receiver 1 and 2 and satellite k and l follows from Eq. (3.3) and is given by

$$\begin{aligned} \frac{\lambda}{2\pi}(\varphi_{12}^k - \varphi_{12}^l) &= (r_{12}^k - r_{12}^l) - \lambda(N_{12}^k - N_{12}^l) + (m_{\varphi,12}^k - m_{\varphi,12}^l) + (\varepsilon_{12}^k - \varepsilon_{12}^l) && (3.4) \\ &= r_{12}^{kl} - \lambda N_{12}^{kl} + m_{\varphi,12}^{kl} + \varepsilon_{12}^{kl}. && (3.5) \end{aligned}$$

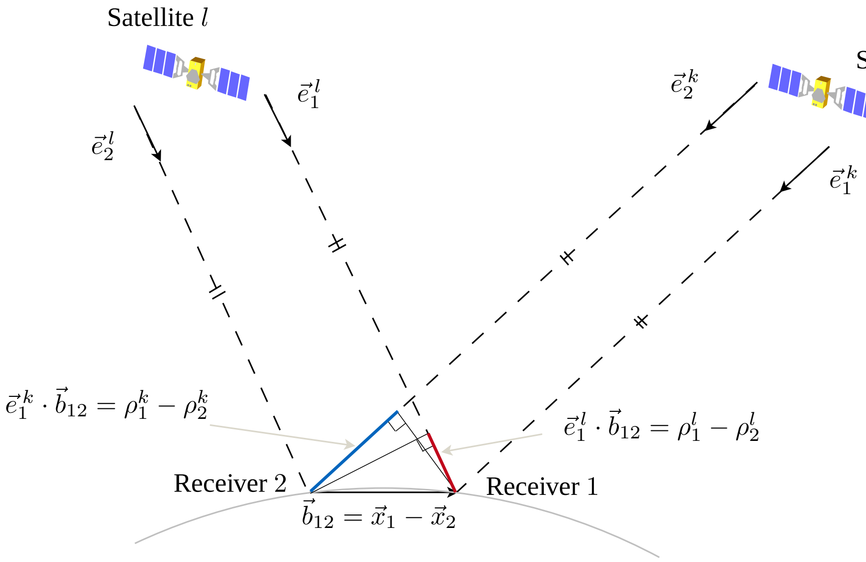

Satellite l
Satellite k
$\vec{e}_1^{\,l}$
$\vec{e}_2^{\,l}$
$\vec{e}_2^{\,k}$
$\vec{e}_1^{\,k}$
$\vec{e}_1^{\,k} \cdot \vec{b}_{12} = \rho_1^k - \rho_2^k$
$\vec{e}_1^{\,l} \cdot \vec{b}_{12} = \rho_1^l - \rho_2^l$
Receiver 2
Receiver 1
$\vec{b}_{12} = \vec{x}_1 - \vec{x}_2$

Figure 3.1: Double difference with short baseline

The main objective of this work is to determine the heading of a car or a ship in harsh environments. The heading can be easily obtained from the baseline[2] using trigonometric functions. Therefore, it is practical to introduce the baseline vector $\vec{b}_{12}$ into the double difference equation. Knowing that $\vec{b}_{12} = \vec{x}_1 - \vec{x}_2$, where $\vec{x}_r$ is the position vector of receiver r in the ECEF frame, and that for short baselines $\vec{e}_1^{\,k} \approx \vec{e}_2^{\,k}$, where $\vec{e}_r^{\,k}$ is the line-of-sight vector directing from satellite k to receiver r (see Fig. 3.1), we can therefore rewrite Eq. (3.4) as

$$\begin{aligned}\frac{\lambda}{2\pi}(\varphi_{12}^k - \varphi_{12}^l) =& (\|\vec{x}_1 - \vec{x}^{\,k}\| - \|\vec{x}_2 - \vec{x}^{\,k}\|) - (\|\vec{x}_1 - \vec{x}^{\,l}\| - \|\vec{x}_2 - \vec{x}^{\,l}\|) + \lambda N_{12}^{kl} + m_{\varphi,12}^{kl} + \varepsilon_{12}^{kl}\\ =& (\vec{e}_1^{\,k,\mathrm{T}} \cdot (\vec{x}_1 - \vec{x}^{\,k}) - \vec{e}_2^{\,k,\mathrm{T}} \cdot (\vec{x}_2 - \vec{x}^{\,k})) - (\vec{e}_1^{\,l,\mathrm{T}} \cdot (\vec{x}_1 - \vec{x}^{\,l}) - \vec{e}_2^{\,l,\mathrm{T}} \cdot (\vec{x}_2 - \vec{x}^{\,l}))\\ &+ c_{12}^{kl} + \lambda N_{12}^{kl} + m_{\varphi,12}^{kl} + \varepsilon_{12}^{kl}\\ \approx& (\vec{e}_1^{\,k} - \vec{e}_1^{\,l})^{\mathrm{T}} \cdot \vec{b}_{12} + c_{12}^{kl} + \lambda N_{12}^{kl} + m_{\varphi,12}^{kl} + \varepsilon_{12}^{kl}, \end{aligned} \tag{3.6}$$

where c_{12}^{kl} is a correction for the projected *satellite movement* within the time of the differential receiver clock offset, which was derived by Juan Cardenas in his master thesis [10] and will be explained later in this thesis. The same derivation can also be applied to the pseudorange, resulting in the following relationship:

$$\begin{aligned}\rho_{12}^k - \rho_{12}^l =& (\|\vec{x}_1 - \vec{x}^{\,k}\| - \|\vec{x}_2 - \vec{x}^{\,k}\|) - (\|\vec{x}_1 - \vec{x}^{\,l}\| - \|\vec{x}_2 - \vec{x}^{\,l}\|) + m_{\rho,12}^{kl} + \eta_{12}^{kl}\\ =& (\vec{e}_1^{\,k,\mathrm{T}} \cdot (\vec{x}_1 - \vec{x}^{\,k}) - \vec{e}_2^{\,k,\mathrm{T}} \cdot (\vec{x}_2 - \vec{x}^{\,k})) - (\vec{e}_1^{\,l,\mathrm{T}} \cdot (\vec{x}_1 - \vec{x}^{\,l}) - \vec{e}_2^{\,l,\mathrm{T}} \cdot (\vec{x}_2 - \vec{x}^{\,l}))\\ &+ c_{12}^{kl} + m_{\rho,12}^{kl} + \eta_{12}^{kl}\\ \approx& (\vec{e}_1^{\,k} - \vec{e}_1^{\,l})^{\mathrm{T}} \cdot \vec{b}_{12} + c_{12}^{kl} + m_{\rho,12}^{kl} + \eta_{12}^{kl}. \end{aligned} \tag{3.7}$$

Assuming that K satellites are visible at a certain epoch and that satellite 1 is chosen as reference satellite, we can write the $K-1$ respective double difference carrier phase measurements in matrix-

[1] Satellite Positioning Service of the German State Survey. It has around 200 stations nationwide and provides correction up to centimeter precision in real-time [9].

[2] Baseline is the vector linking the two receivers.

vector notation as:

$$\begin{pmatrix} \varphi_{12}^{21} - c_{12}^{21} \\ \vdots \\ \varphi_{12}^{K1} - c_{12}^{K1} \end{pmatrix} = \begin{pmatrix} (\vec{e}_1^{\,1} - \vec{e}_1^{\,2})^{\mathrm{T}} \\ \vdots \\ (\vec{e}_1^{\,1} - \vec{e}_1^{\,K})^{\mathrm{T}} \end{pmatrix} \vec{b}_{12} + \begin{pmatrix} \lambda N_{12}^{21} \\ \vdots \\ \lambda N_{12}^{K1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{12}^{21} \\ \vdots \\ \varepsilon_{12}^{K1} \end{pmatrix} . \tag{3.8}$$

Note that in Eq. (3.8), multipath is considered as part of measurement noise ε_{12}^{k1}. Besides, $\lambda/(2\pi)\varphi_{12}^{k1}$ was redefined as φ_{12}^{kl} to simplify the notation. Due to the presence of ambiguities, there are $3 + (K - 1)$ unknowns in the above matrix equation. Teunissen suggested in [11] to use phase measurements from multiple epochs to avoid having an under-determined system. However, Günther pointed out in [9] that these phase measurements have to be sufficiently spaced in time domain to ensure linear independence as multipath can be highly correlated between consecutive epochs while the line-of-sight unit vector varies little from one epoch to the other. In order to reduce the observation time needed to solve for baseline and fix the ambiguities, we use code measurements in addition to the phase measurements. Code measurements, although noisy, are useful in our case as they are not affected by ambiguities [9]. Stacking corrected double difference carrier phase and code measurements taken from n_{ep} epochs, we obtain the following system of equations:

$$\underbrace{\left(\begin{array}{c} \varphi_{12}^{21}(t_1) - c_{12}^{21}(t_1) \\ \vdots \\ \varphi_{12}^{K1}(t_1) - c_{12}^{K1}(t_1) \\ \hline \vdots \\ \hline \varphi_{12}^{21}(t_{n_{\mathrm{ep}}}) - c_{12}^{21}(t_{n_{\mathrm{ep}}}) \\ \vdots \\ \varphi_{12}^{K1}(t_{n_{\mathrm{ep}}}) - c_{12}^{K1}(t_{n_{\mathrm{ep}}}) \\ \hline \rho_{12}^{21}(t_1) - c_{12}^{21}(t_1) \\ \vdots \\ \rho_{12}^{K1}(t_1) - c_{12}^{K1}(t_1) \\ \hline \vdots \\ \hline \rho_{12}^{21}(t_{n_{\mathrm{ep}}}) - c_{12}^{21}(t_{n_{\mathrm{ep}}}) \\ \vdots \\ \rho_{12}^{K1}(t_{n_{\mathrm{ep}}}) - c_{12}^{K1}(t_{n_{\mathrm{ep}}}) \end{array}\right)}_{\Psi} = \underbrace{\left(\begin{array}{c} (\vec{e}_1^{\,1}(t_1) - \vec{e}_1^{\,2}(t_1))^{\mathrm{T}} \\ \vdots \\ (\vec{e}_1^{\,1}(t_1) - \vec{e}_1^{\,K}(t_1))^{\mathrm{T}} \\ \hline \vdots \\ \hline (\vec{e}_1^{\,1}(t_{n_{\mathrm{ep}}}) - \vec{e}_1^{\,2}(t_{n_{\mathrm{ep}}}))^{\mathrm{T}} \\ \vdots \\ (\vec{e}_1^{\,1}(t_{n_{\mathrm{ep}}}) - \vec{e}_1^{\,K}(t_{n_{\mathrm{ep}}}))^{\mathrm{T}} \\ \hline (\vec{e}_1^{\,1}(t_1) - \vec{e}_1^{\,2}(t_1))^{\mathrm{T}} \\ \vdots \\ (\vec{e}_1^{\,1}(t_1) - \vec{e}_1^{\,K}(t_1))^{\mathrm{T}} \\ \hline \vdots \\ \hline (\vec{e}_1^{\,1}(t_{n_{\mathrm{ep}}}) - \vec{e}_1^{\,2}(t_{n_{\mathrm{ep}}}))^{\mathrm{T}} \\ \vdots \\ (\vec{e}_1^{\,1}(t_{n_{\mathrm{ep}}}) - \vec{e}_1^{\,K}(t_{n_{\mathrm{ep}}}))^{\mathrm{T}} \end{array}\right)}_{H_{\mathrm{geo}}} \vec{b}_{12} + \lambda \underbrace{\left(\begin{array}{c} N_{12}^{21} \\ \vdots \\ N_{12}^{K1} \\ \hline \vdots \\ \hline N_{12}^{21} \\ \vdots \\ N_{12}^{K1} \\ \hline 0 \\ \vdots \\ 0 \\ \hline \vdots \\ \hline 0 \\ \vdots \\ 0 \end{array}\right)}_{AN} + \underbrace{\left(\begin{array}{c} \varepsilon_{12}^{21}(t_1) \\ \vdots \\ \varepsilon_{12}^{K1}(t_1) \\ \hline \vdots \\ \hline \varepsilon_{12}^{21}(t_{n_{\mathrm{ep}}}) \\ \vdots \\ \varepsilon_{12}^{K1}(t_{n_{\mathrm{ep}}}) \\ \hline \eta_{12}^{21}(t_1) \\ \vdots \\ \eta_{12}^{K1}(t_1) \\ \hline \vdots \\ \hline \eta_{12}^{21}(t_{n_{\mathrm{ep}}}) \\ \vdots \\ \eta_{12}^{K1}(t_{n_{\mathrm{ep}}}) \end{array}\right)}_{\eta} . \tag{3.9}$$

By introducing the differential geometry matrix H_{geo} and the mapping matrix A which maps the differential ambiguity to its measurement, we can further simplify the notation of Eq. (3.9) to

$$\Psi = H_{\mathrm{geo}}\vec{b}_{12} + AN + \eta = H\xi + \eta, \tag{3.10}$$

where

$$\xi = \begin{pmatrix} \vec{b}_{12} \\ N \end{pmatrix} \quad \text{and} \quad H = \begin{pmatrix} H_{\mathrm{geo}} & A \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} \lambda \cdot 1_{(t_{nep})\times(K-1)} & 0_{(t_{nep})\times(K-1)} \end{pmatrix}^{\mathrm{T}} .$$

3.2.1 Least-squares float solution

Eq. (3.9) is an over-determined system of equations. We determine a weighted least-squares float solution of $\vec{b}_{12}$ and N (see e.g. [12] and [13]). The positive-definite weighting matrix prioritizes good and penalizes noisy measurements. The weighting matrix is typically chosen as the inverse of the measurement covariance matrix Σ_{Ψ} as described by Misra and Enge in [13].

We analyze two approaches to determine the variances and co-variances of Σ_{Ψ}: The first one is based on the residuals of the polynomial fitting of the double difference measurements. The number of epochs and the order of the polynomial are chosen in such a way that

1) the *temporal variations* of the *double difference measurements* due to *receiver and satellite movements* can be described by the *polynomial*

2) the number of epochs is sufficiently large to reflect the impact of code multipath even in stationary conditions

The second approach is based on a model of the code noise standard deviations: It assumes that the code noise standard deviations are elevation dependant, and that the dependency follows an exponential function.

The least-squares float solution of Eq. (3.10) is given by

$$\begin{pmatrix} \vec{b}_{12} \\ N \end{pmatrix} = (H^{\mathrm{T}}WH)^{-1}H^{\mathrm{T}}W\Psi. \tag{3.11}$$

3.3 Attitude determination

In this section, we derive the 3D-attitude (heading and pitch) from the relative position between two GPS receivers. We focus on two receivers mounted on the roof of a car along its longitudinal axis, and with equal distances to both the left and right side of the car as shown in Fig. 3.2.

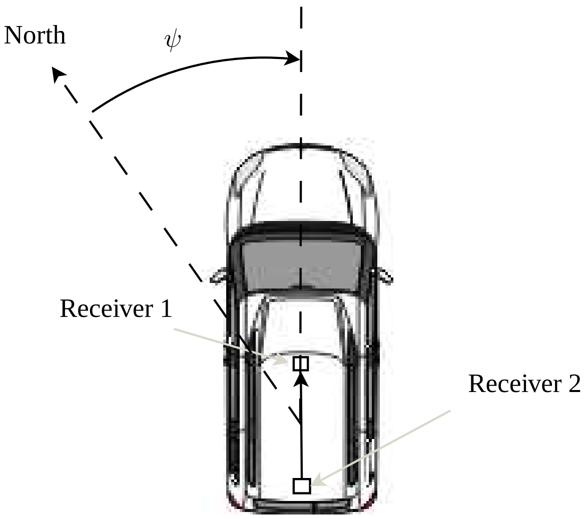


Figure 3.2: Heading determination of a car with two GNSS receivers

The distance between both receivers is a priori determined by a meter with an accuracy of approximately 1 cm and included in the ambiguity fixing and attitude determination. The height difference between both receivers is in general negligible and is constrained to zero in environments with an insufficient number of visible satellites.

3.3.1 Heading determination

We count the heading clock-wise on the East-North plane with $0°$ in Northern direction. The least-squares float solution of the baseline vector $\vec{b}_{12}$ in Eq. (3.11) is given in the ECEF (Earth-Centered, Earth-Fixed) frame. For heading determination, it is more practical to express the baseline vector in the ENU (East, North, Up) coordinate frame centered at the center of inertia of the car:

$$\vec{b}_{12,\mathrm{ENU}} = R_{\mathrm{L}}\,\vec{b}_{12} = (b_{\mathrm{E}}, b_{\mathrm{N}}, b_{\mathrm{U}})^{\mathrm{T}} \quad \text{with} \quad R_{\mathrm{L}} = R_1(\pi/2-\phi)R_3(\pi/2+\lambda), \tag{3.12}$$

where ϕ and λ are the latitude and longitude of the center of inertia of the car at the current epoch; R_1 and R_3 are the rotation matrices about the x-axis and respectively the z-axis in the ECEF coordinate frame. The heading is thus given by:

$$\psi = \arctan(b_{\mathrm{E}}/b_{\mathrm{N}}). \tag{3.13}$$

3.3.2 Pitch angle determination

The pitch angle (and thereby the slope of the road) can also be easily obtained from the baseline vector:

$$\theta = \arctan\left(\frac{b_{\mathrm{U}}}{\sqrt{b_{\mathrm{E}}^2 + b_{\mathrm{N}}^2}}\right). \tag{3.14}$$

4. Determination of least-squares float solution

In this chapter, we first review the iterative Gauss-Newton method for estimating the baseline and double difference ambiguities using the measurement model of Eq. (3.10). The float solution disregards the integer property of ambiguities and is subsequently fixed to an integer one as described in Chap. 5.

Besides the traditional float solution, this thesis also includes a new method to efficiently determine a baseline length constrained float solution. So far, an *iterative* solution was required for calculating the derivative of the baseline length constraint function with respect to Lagrange parameter. In this thesis, *a closed-form expression* is derived by exploiting a *property of the derivative of matrix inversions*, which does no longer require an iterative solution.

4.1 Synchronization of low-cost GPS receivers

Clock offsets of low-cost GPS receivers (e.g. u-blox, Skytraq) typically are in the order of milliseconds to seconds, which is 6-9 orders of magnitude larger than of geodetic receivers. The receiver clock offsets do not directly affect the double difference measurements as they are cancelled by the double differencing. However, there is an indirect affect: As satellites move with a speed of approximately 4 km/s, the satellite movement within the time of the receiver clock offset can vary between several metres (for u-blox receivers) up to several kilometres (for Skytraq receivers). Thus, there is a need for correcting the satellite movement within the time of the differential receiver clock offsets, i.e. the difference between the clock offsets of both receivers.

The most accurate model available today for double difference measurements of *low-cost* GPS receivers was developed by ANAVS and is described in [8]. It is given by

$$\rho_1^{kl}(t+\delta\tau_1) - \rho_2^{kl}(t+\delta\tau_2) \approx (\vec{e}^{\,kl}(t))^{\mathrm{T}}\vec{b}_{12}(t) + c_{12}^{kl}(t,\delta\tau_1,\delta\tau_2) + m_{12}^{kl}(t) + \eta_{12}^{kl}(t), \tag{4.1}$$

where m_{12}^{kl} is the differential code multipath and $c_{12}^{kl}(t,\delta\tau_1,\delta\tau_2)$ is the correction for the satellite movement within the time of the differential receiver clock offset $\delta\tau_1 - \delta\tau_2$, which was derived by Henkel and Cardenas in [8] and [10] as

$$\begin{aligned} c_{12}^{kl}(t+\delta\tau_1, t+\delta\tau_2) \;=\;& (\vec{e}_1^{\,k}(t+\delta\tau_1))^{\mathrm{T}}\left(\vec{x}_1(t+\delta\tau_1) - \vec{x}^{\,k}(t+\delta\tau_1-\varDelta\tau_1^k)\right)\\ & -(\vec{e}_1^{\,l}(t+\delta\tau_1))^{\mathrm{T}}\left(\vec{x}_1(t+\delta\tau_1) - \vec{x}^{\,l}(t+\delta\tau_1-\varDelta\tau_1^l)\right)\\ & -(\vec{e}_1^{\,k}(t+\delta\tau_2))^{\mathrm{T}}\left(\vec{x}_1(t+\delta\tau_2) - \vec{x}^{\,k}(t+\delta\tau_2-\varDelta\tau_2^k)\right)\\ & +(\vec{e}_1^{\,l}(t+\delta\tau_2))^{\mathrm{T}}\left(\vec{x}_1(t+\delta\tau_2) - \vec{x}^{\,l}(t+\delta\tau_2-\varDelta\tau_2^l)\right). \end{aligned} \tag{4.2}$$

It shall be stressed that the correction has millimetre accuracy despite the use of noisy code measurements. This high accuracy is achieved due to the differential nature of the correction, i.e. the satellite positions $\vec{x}^{\,k}(t+\delta\tau_1-\varDelta\tau_1^k)$ and $\vec{x}^{\,k}(t+\delta\tau_2-\varDelta\tau_2^k)$ are both derived solely from the measurements of the first receiver and the clock offset $\delta\tau_2$ of the second receiver. The noise in the clock estimate of the second receiver has still only a negligible impact on the correction, as 1 m noise in the clock estimate corresponds to a time of only $1/3\cdot 10^{-8}$ s during which the satellite

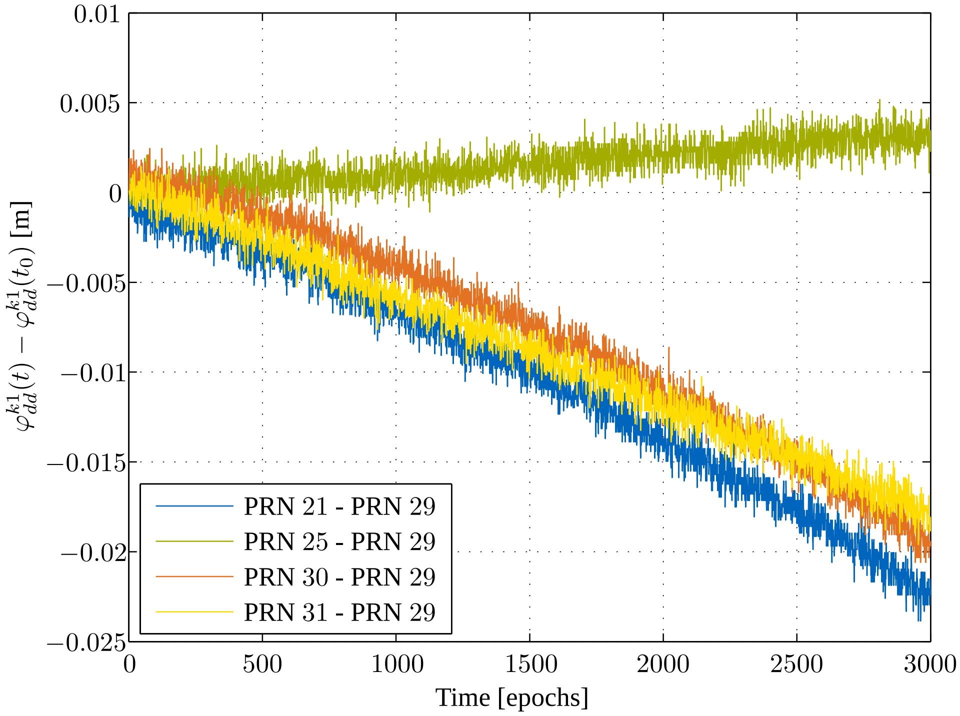


Figure 4.1: Double difference phase measurements of zero-baseline stationary receivers subtracted with DD measurement of the first epoch. The differential DD measurements demonstrate a linear drift.

movement can be neglected. The satellite positions $\vec{x}^k(t+\delta\tau_1 - \Delta\tau_1^k)$ are first calculated and then linear interpolated to $\vec{x}^k(t+\delta\tau_2 - \Delta\tau_2^k)$.

A similar model was suggested for the double difference carrier phase measurements, i.e.

$$\lambda/2\pi(\varphi_1^{kl}(t+\delta\tau_1) - \varphi_2^{kl}(t+\delta\tau_2)) \approx (\vec{e}^{\,kl}(t))^{\mathrm{T}}\vec{b}_{12}(t) + c_{12}^{kl}(t,\delta\tau_1,\delta\tau_2) + \lambda N_{12}^{kl} + \varepsilon_{12}^{kl}(t), \quad (4.3)$$

where the phase multipath has been mapped to the phase noise as it is typically less than 2 cm.

Fig. 4.1 shows the temporal change of the double difference synchronization correction relative to the first epoch for a zero-baseline test. As the baseline was by definition zero, the synchronization correction should be constant if the differential receiver clock offset is negligible. However, a drift is visible which could be derived directly from Eq. (4.2).

The synchronization correction term is quasi linear as one can observe from Fig. 4.1. In order to keep the computational complexity at a minimum, we determine the synchronization correction only every 50-th epoch (i.e. every 10 s), and *interpolate* $c_{12}^{kl}(t)$ in between. The coefficients $\hat{\alpha}_0$ and $\hat{\alpha}_1$ of the linear polynomial are found by least-squares estimation as

$$\begin{pmatrix} \hat{\alpha}_0 \\ \hat{\alpha}_1 \end{pmatrix} = \arg\min_{\alpha_0,\alpha_1} \sum_{t'=t-50}^{t-1} \| c_{12}^{kl}(t') - \sum_{p=0}^{1} \alpha_p \cdot t^p \|^2. \quad (4.4)$$

Data analysis showed that an *extrapolation* can also be performed over 50 epochs with an accuracy of 1 mm to 1 cm. Therefore, we determine the correction term every 50-th epoch and extrapolate it based on the last two analytically computed corrections $c_{12}^{kl}(t_1)$ and $c_{12}^{kl}(t_2)$, i.e.

$$c_{12}^{kl}(t) = c_{12}^{kl}(t_1) + \frac{c_{12}^{kl}(t_1) - c_{12}^{kl}(t_2)}{t_1 - t_2}(t - t_1), \quad \text{for} \quad t_2 < t_1 < t. \quad (4.5)$$

However, clock jumps and changes in the reference satellites require a careful adaption of the synchronization correction. In detail, the following adaption has been integrated:

- Interpolation of satellite positions $\vec{x}^k$ and $\vec{e}^k$ over time of differential receiver clock-offset
  - every 50 epochs (10 s)
  - additionally in case of clock jumps
  - additionally in case of change of reference satellite
- Analytical computation of correction
  - every 50 epochs (10 s)
  - additionally in case of clock jumps
  - additionally in case of change of reference satellite
  - remarks:
    - in all cases except of clock jumps: update of slope of correction
    - in all other cases: interpolation of correction based on most recent analytical correction and slope
- Change in reference satellite
  - transform correction
  - additionally interpolate transformed correction in case of loss lock (no signal)

4.2 Unconstrained least-squares float solution

This section briefly reviews the calculation of the float baseline/ ambiguity solution using the iterative Gauss-Newton algorithm. The sum of squared residuals is minimized, where *residuals* are defined as the difference between the *observed* and the *calculated* double difference (DD) measurements. The so-called *calculated* double difference measurements are derived directly from the receiver and satellite positions as described by Borre in [14]. We also include the *synchronisation correction* also for the *calculated* DD to be consistent with the synchronized DD measurements.

4.2.1 Unconstrained least-squares float solution

We subtract the synchronisation correction directly from the double difference measurements of Eq. (3.9). We now model the corrected double difference measurements as

$$\varPsi = H\xi + \begin{pmatrix} m_\varphi \\ m_\rho \end{pmatrix} + \begin{pmatrix} \varepsilon \\ \eta \end{pmatrix}, \tag{4.6}$$

where

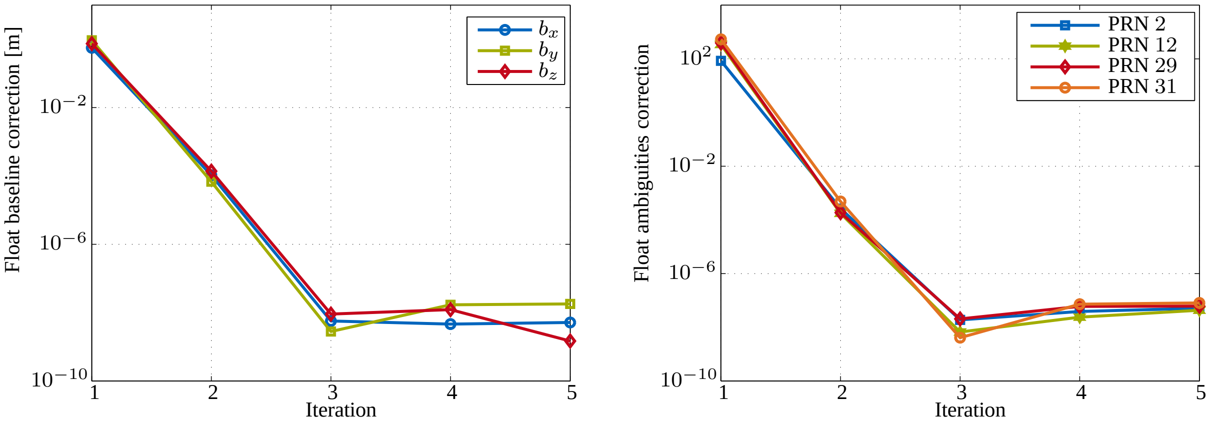
$$H = \begin{pmatrix} \tilde{H}_{\text{geo}} & A \\ \tilde{H}_{\text{geo}} & 0_{(K-1)\cdot t_{n_{\text{ep}}}\times(K-1)} \end{pmatrix} \quad \text{and} \quad \xi = \begin{pmatrix} b \\ N \end{pmatrix},$$

with the pure geometry matrix $\tilde{H}_{\text{geo}}$ including the normalized line-of-sight vectors, the mapping matrix $A = \lambda \cdot 1^{(K-1)\cdot t_{n_{\text{ep}}}\times(K-1)}$ dependent only on the wavelength λ, the baseline vector b in ECEF coordinates, the ambiguities N, the code and phase multipath m_φ and m_ρ, and the phase and code noises ε and η.

Note that the H matrix does not need to be updated during the iterations. The one obtained from the initial absolute position determination can be re-used as its accuracy is sufficient for relative positioning over short baselines. This is very beneficial from a computational point of view as the H matrix typically includes $t_{n_{\text{ep}}} = 800$ epochs to achieve a sufficiently accurate convergence of the float ambiguity solution. Measurement analysis has shown that there is no need to update the $3 \cdot K \cdot 800 = 2400 \cdot K$ parameters of the H matrix at each iteration, which is a substantial benefit for real-time implementations on a microprocessor.

The calculation of the synchronization correction also depends on the H-matrix. However, the H matrix of the absolute position determination has a sufficient accuracy such that there is also no need to update the synchronization correction after determining the float (or fixed) solution. Although H is already pre-determined, an *iterative* solution is still required for estimating the baseline and float ambiguities: The benefit of the iterations arises from the *calculated* double difference measurements, which are updated after updating the position estimate of the second receiver.

Our algorithm for calculating the weighted least-squares float solution with the synchronization correction is shown in diagram 2. The first receiver is chosen as reference receiver. Its position estimate is averaged while the receiver is stationary. The algorithm then uses the iterative Gauss-Newton method which requires an initialization of the unknown parameters. We initialize both baseline and ambiguities with zero (see *line 3* and *4*). This implies that the absolute position of receiver 2 is initialized with the position estimate of receiver 1 (*line 2*). In *lines 8* and *9*, receiver-satellite ranges are calculated based on the estimated receiver and satellite positions. The latter ones were obtained with the ephemeris data of the navigation message. Subsequently, *calculated* double differences (DD) are obtained using solely these calculated ranges (*line 14*) and the synchronization corrections. The calculated DD are then stacked in a vector (*line 17*).



(a) (b)

Figure 4.2: a) Convergence of baseline correction term; b) Convergence of float ambiguities correction term

In *line 18*, double difference residuals are determined. They are defined as the difference between the *calculated* and *measured* double differences. The residuals will then be used to determine the least-squares baseline and ambiguity corrections (*line 19*). The baseline and float ambiguity estimates are subsequently updated by adding (subtracting) the corrections to the estimates of the previous iteration (*lines 20* and *21*). Figure 4.2 shows the convergence of the float solution. Each of the first two iterations reduces the uncertainty of the float estimation by 4 orders of magnitude. At the third iteration, the correction terms are smaller than 10^{-5} m. Given the speed of convergence, a maximum of 5 iteration is definitely sufficient for our algorithm 2.

Algorithm 2 Iterative least-squares float solution with synchronization correction

```
Input: $\vec{x}_1(t), \vec{x}_1^{\,k}(t), \quad W, \quad H, \Delta\varphi_{12}^{k1}(t), \Psi, \quad \forall k, \forall t$
Output: $\vec{x}_{2,\text{ave}}, N$
$\vec{x}_{1,\text{ave}} = \mathrm{E}\left\{\vec{x}_1(t)\right\}$    ▷ Averaging receiver 1 absolute positions
$\vec{x}_{2,\text{ave}}^{(0)} = \vec{x}_{1,\text{ave}}$    ▷ Initialization of relative position and correction terms
$b_{\text{corr}} = 0_{3\times 1}$
$N_{\text{corr}} = 0_{(K-1)\times 1}$

for $i = 1 \rightarrow 5$ do    ▷ Newton iterations

    for $t = 1 \rightarrow t_{\max}$ do
        for $k = 1 \rightarrow K$ do
            $r_{1,\text{cal}}^{k}(t) = \|\vec{x}^{\,k}(t) - \vec{x}_{1,\text{ave}}\|$    ▷ Calculation of receiver-satellite ranges
            $r_{2,\text{cal}}^{k,(i)}(t) = \|\vec{x}^{\,k}(t) - \vec{x}_{2,\text{ave}}^{(i)}\|$
        end for
    end for

    for $t = 1 \rightarrow t_{\max}$ do    ▷ Calculation of double difference ranges
        for $k = 2 \rightarrow K$ do
            $r_{12,\text{cal}}^{k1,(i)}(t) = (r_{1,\text{cal}}^{k}(t) - r_{1,\text{cal}}^{1}(t)) - (r_{2,\text{cal}}^{k,(i)}(t) - r_{2,\text{cal}}^{1,(i)}(t)) - c_{12}^{k1}(t)$
        end for
    end for

    $r_{\text{dd,cal}}^{(i)} = \left( r_{12,\text{cal}}^{21,(i)}(1) \;\; \dots \;\; r_{12,\text{cal}}^{K1,(i)}(1) \;\middle|\; \dots \;\; \dots \;\middle|\; r_{12,\text{cal}}^{21,(i)}(t_{\max}) \;\; \dots \;\; r_{12,\text{cal}}^{K1,(i)}(t_{\max}) \right)^{\mathrm{T}}$
    $\begin{pmatrix} r_{\text{dd},\varphi}^{(i)} \\ r_{\text{dd},\rho}^{(i)} \end{pmatrix} = \Psi - 1_{2\times 1} \otimes r_{\text{dd,cal}}^{(i)}$    ▷ Calculation of phase and code residuals
    $\begin{pmatrix} b_{\text{corr}}^{(i)} \\ N_{\text{corr}}^{(i)} \end{pmatrix} = (H^{\mathrm{T}} W H)^{-1} H^{\mathrm{T}} W \begin{pmatrix} r_{\text{dd},\varphi}^{(i)} \\ r_{\text{dd},\rho}^{(i)} \end{pmatrix}$    ▷ Determination of correction terms
    $x_{2,\text{ave}}^{(i)} = x_{2,\text{ave}}^{(i-1)} - b_{\text{corr}}^{(i)}$    ▷ Update receiver 2 position and float ambiguities
    $N^{(i)} = N^{(i-1)} + N_{\text{corr}}^{(i)}$

end for
```

4.3 Covariance matrix

This section focuses on two methods to determine the covariance matrix of the tracked phase and code measurements. The first one is particular suitable if double difference measurements of two static receivers are available for a few hundred epochs. The second one is based on a stochastic model, which assumes that the standard deviations are elevation-dependent and that this dependency follows an exponential function. The inverse of the covariance matrix provides the weighting matrix W that we used in our weighted least-squares float solution (see *line 19*).

4.3.1 Estimation of noise statistics

We estimate the statistics of the double differences from a large number of measurements. Let us first take a closer look at Eq. (3.6) and its dependency on time for *static* receivers:

$$\varphi_{12}^{kl}(t) \approx (\vec{e}_1^{\,k}(t) - \vec{e}_2^{\,l}(t))^{\mathrm{T}} \cdot \vec{b}_{12} + \lambda N_{12}^{kl} + \varepsilon_{12}^{12}(t), \tag{4.7}$$

where the multipath has been included in the noise.

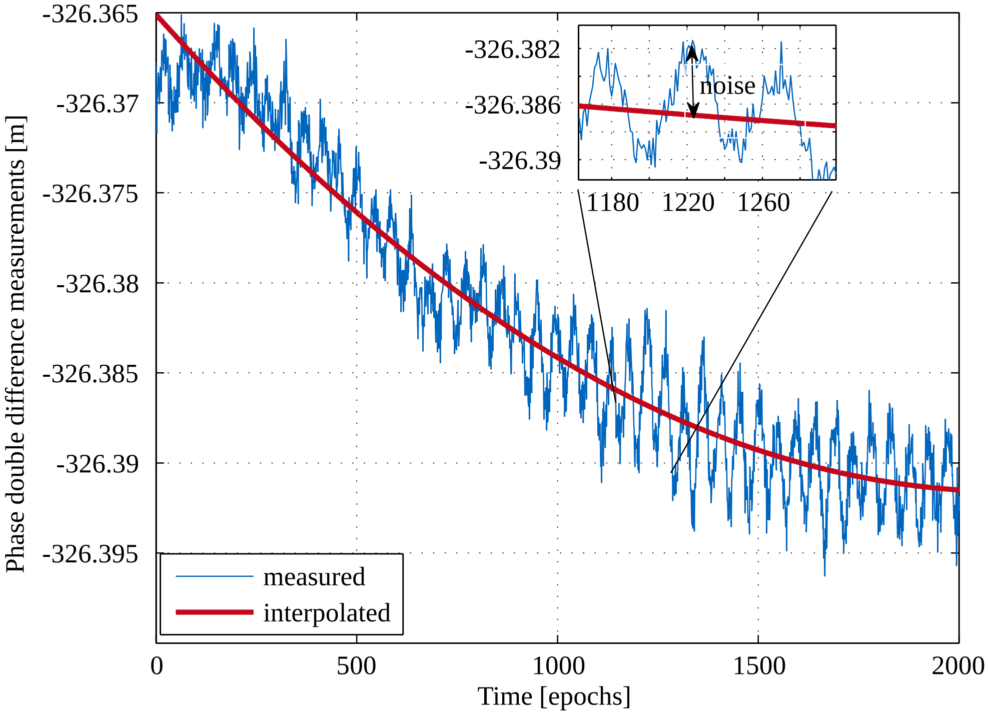


Figure 4.3: Carrier phase double difference measurement of a static receivers. The red curve represents the interpolated measurements using a second degree polynomial. Deviation of the measurements from the interpolation is considered as measurement noise and multipath.

The ambiguity is constant if no cycle slips occur. With static receivers, the baseline vector is constant but the satellite-receiver line of sight vectors still change over time. We use a polynomial to describe this time dependency. Measurement analysis has shown that a second order polynomial is sufficient to describe the dynamics of the projected satellite positions over up to 2000 epochs (see Fig. 4.3).

The coefficients of the second order polynomial are obtained by minimizing the squared difference between the double difference carrier phases and the polynomial, i.e.

$$\begin{pmatrix} \hat{\alpha}_0^{k1} \\ \hat{\alpha}_1^{k1} \\ \hat{\alpha}_2^{k1} \end{pmatrix} = \arg \min_{\alpha_0^{k1},\alpha_1^{k1},\alpha_2^{k1}} \sum_{t=1}^{t_{\max}} \left(\varphi_{12}^{k1}(t) - \sum_{p=0}^{2} \alpha_p^{1k} \cdot t^p \right)^2 . \tag{4.8}$$

An estimate of the variance of φ_{12}^{k1} then follows from

$$\sigma^2(\varphi_{12}^{k1}) \approx 1/t_{\max} \sum_{t=1}^{t_{\max}} \left(\varphi_{12}^{k1}(t) - \sum_{p=0}^{2} \hat{\alpha}_p^{1k} \cdot t^p \right)^2 . \tag{4.9}$$

Similarly, the covariance between the double differences φ_{12}^{k1} and φ_{12}^{l1} follows from

$$\sigma(\varphi_{12}^{k1}, \varphi_{12}^{l1}) \approx 1/t_{\max} \sum_{t=1}^{t_{\max}} \left(\varphi_{12}^{k1}(t) - \sum_{p=0}^{2} \hat{\alpha}_p^{1k} \cdot t^p \right) \left(\varphi_{12}^{l1}(t) - \sum_{p=0}^{2} \hat{\alpha}_p^{1l} \cdot t^p \right) . \tag{4.10}$$

It shall be stressed that the estimation of the noise statistics from the residuals of a least-squares polynomial fitting can only be performed for static receivers.

4.3.2 Estimation of noise/ multipath statistics with an exponential delay model

Signals from satellites of lower elevation are typically affected more by multipath than signals from satellites of higher elevation. McGraw et al. [15] carefully analyzed the noise statistics for Local Area Augmentation Systems (LAAS). They showed that the dependency of the noise standard deviations on the elevation angle can be well described by an exponential function. Henkel and Günther used this model to analyze the impact of code multipath on ambiguity fixing and derived a partial integer decorrelation for the optimum trade-off between variance reduction and bias amplification in [16].

The exponential multipath delay model can be expressed as follows:

$$\sigma_\rho^k(E^k) = \sigma_{\rho,0} \cdot \mathrm{e}^{-E^k/E_\rho} \quad \text{and} \quad \sigma_\varphi^k(E^k) = \sigma_{\varphi,0} \dot{} \mathrm{e}^{-E^k/E_\varphi},$$

where σ_ρ^k and σ_φ^k are the code and phase standard deviations respectively of satellite k with a certain elevation angle E^k; E_ρ and E_φ are the decay constants while $\sigma_{\rho,0}$ and $\sigma_{\varphi,0}$ denote the upper bounds of the exponential function since both the decay constant and the elevation are strictly positive.

By modeling the noise standard deviations at two elevation angles $\{E_{\text{low}}, E_{\text{up}}\}$ as $\{\sigma_{\rho,\text{low}}, \sigma_{\rho,\text{up}}\}$, we can derive the decay constant. We first take the ratio between both standard deviations

$$\frac{\sigma_{\rho,\text{low}}}{\sigma_{\rho,\text{up}}} = \mathrm{e}^{\frac{E_{\text{up}} - E_{\text{low}}}{E_\rho}}, \tag{4.11}$$

and then take the logarithm and solve for E_ρ:

$$E_\rho = \frac{E_{\text{up}} - E_{\text{low}}}{\ln\left(\sigma_{\rho,\text{low}}/\sigma_{\rho,\text{up}}\right)}. \tag{4.12}$$

Similarly, we derive the decay constant for the model of the phase noise standard deviation:

$$E_\varphi = \frac{E_{\text{up}} - E_{\text{low}}}{\ln\left(\sigma_{\varphi,\text{low}}/\sigma_{\varphi,\text{up}}\right)}. \tag{4.13}$$

Once the decay constants are determined, we can easily derive the upper bounds $\sigma_{\rho,0}$ and $\sigma_{\varphi,0}$ by evaluating Eq. (4.3.2) at E_{low}:

$$\sigma_{\rho,0} = \frac{\sigma_{\rho,\text{low}}}{\text{e}(-E_{\text{low}}/E_\rho)} \quad \text{and} \quad \sigma_{\varphi,0} = \frac{\sigma_{\varphi,\text{low}}}{\text{e}(-E_{\text{low}}/E_\varphi)}. \tag{4.14}$$

Note that the standard deviations determined here are for absolute (i.e. undifferenced) measurements. To obtain the standard deviations of double difference measurements, the individual standard deviations have to be added.

4.3.3 Elimination of epochs with high multipath

Temporally correlated code multipath typically requires a long observation period (we choose 800 epochs which corresponds to 2 min. 40 s) to obtain a sufficiently accurate float solution. In this long time period, temporal variations might take place in the environment even if both receivers are static. We can perform a selection of 'good' measurement epochs based on the *Weighted Sum of Squared Errors (WSSE)* of the code measurements (see appendix A1.2) after completing the baseline estimation of Algorithm 2:

$$\text{WSSE}_{r_{\text{dd},\rho}}(t) = \|r_{\text{dd},\rho}(t)\|^2_{W_\rho} \tag{4.15}$$

with $r_{\text{dd},\rho}(t)$ being the residual of the least-squares float solution.

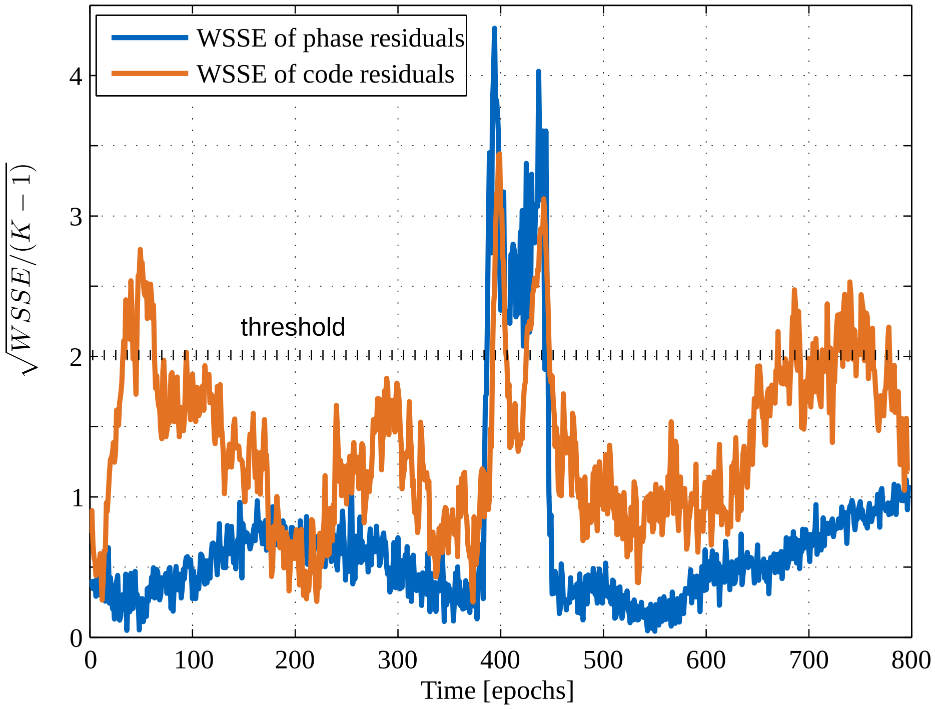


Figure 4.4: Normalized WSSE of phase and code residuals after unconstrained float solution

We use only code measurements from *the epochs* where the weighted sum of code residuals (of the float ambiguity/ baseline solution) is below a predefined threshold. As the WSSE also

Selection strategy	**Heading**	**Heading error**	**Integer estimates**		
			$N_{1,2}^{4,1}$	$N_{1,2}^{13,1}$	$N_{1,2}^{20,1}$
all epochs	-73.7	$+1.7°$	647	335	399
constrained WSSE of code residuals (689 out of 800 epochs)	-73.3	$+1.7°$	647	335	399
constrained WSSE of code residuals (689 out of 800 epochs) and phase residuals (738 out of 800 epochs)	-75.4	$-0.4°$	647	335	398

Table 4.1: Heading estimate and its error after ambiguity fixing with different selection strategies

increases with the number of visible satellites, we additionally normalize it by the number of satellites excluding the reference satellite. The epoch selection is thus based on

$$\frac{1}{K(t)-1}\mathrm{WSSE}_{r_{\mathrm{dd},\rho}}(t) < \mathrm{WSSE}_{\mathrm{th}}. \tag{4.16}$$

Figure 4.4 shows the square root of the normalized WSSE of phase residuals after solving for unconstrained float solution. Between epoch 380 and 450, the car trunk was left open and hence, introduced more noise in the receive signals is caused by reflection on the hard metal surface. This short-term disruption is reflected by the sudden jump in WSSE. The sudden increase in WSSE does not only occur to code measurements but also to phase measurements although carrier phase measurements are less sensitive to multipaths. Such phenomenon is predictable since weights attributed to phase measurements are larger than weights given to code measurements by two order of magnitude. Therefore, a selection of ‘good’ phase measurements shall also be performed:

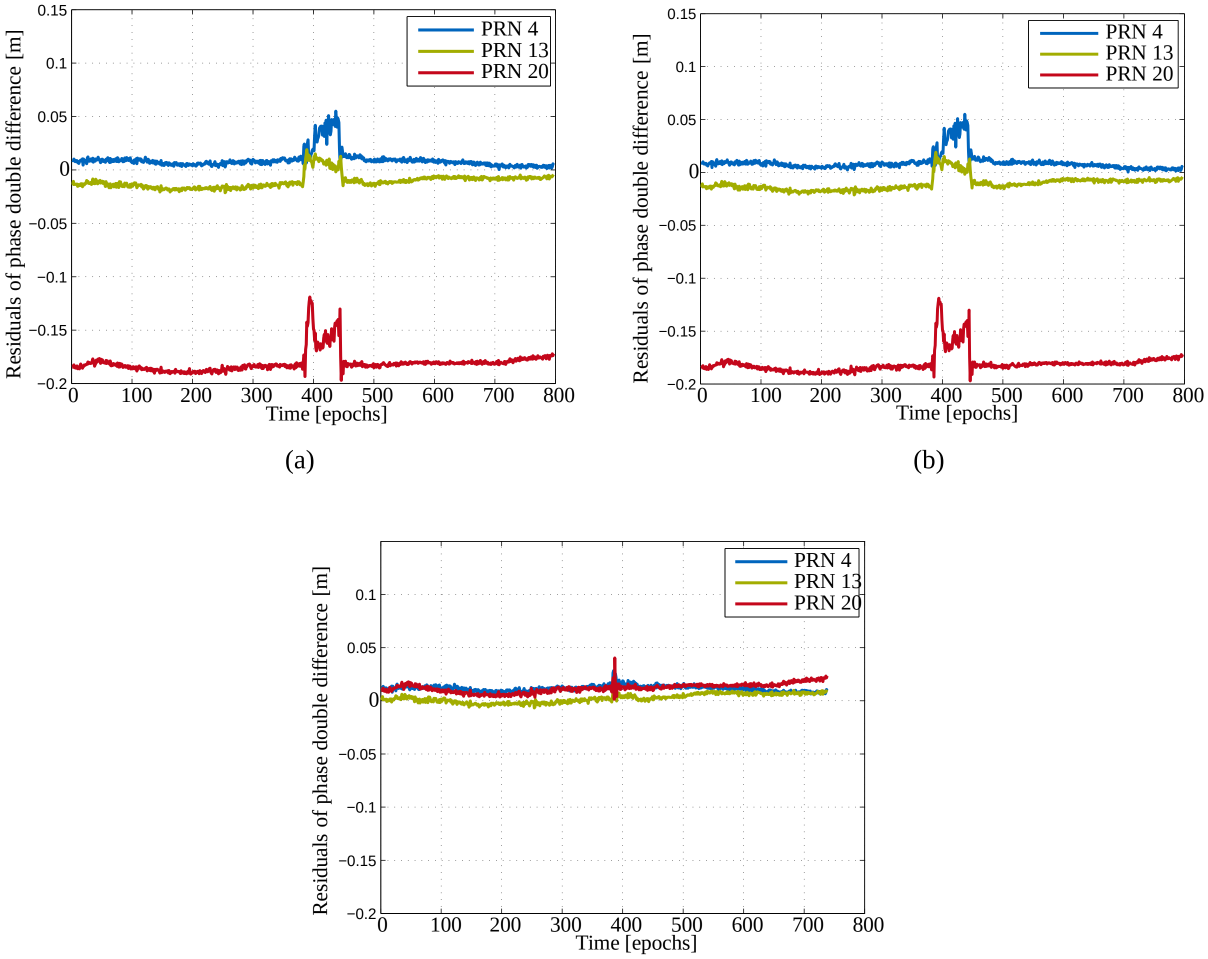
$$\frac{1}{K(t)-1}\mathrm{WSSE}_{r_{\mathrm{dd},\varphi}}(t) < \mathrm{WSSE}_{\mathrm{th}}. \tag{4.17}$$

Table 4.1 groups results obtained after resolving for the integer ambiguities with only the selected measurements. Complete methodology to arrive to the results shown is described in section 7.2.1. Elimination of only code measurements with considerable residuals (larger than $\mathrm{WSSE}_{\mathrm{th}} = 4$) does not affect the fixing decision. This is due to the comparably low weights given to code measurements (see section 4.3.1). An additional elimination of phase measurements with substantial noise is hence performed.

We observe that elimination of both noisy code and phase measurements improve the phase residuals after the fixing (see Fig 4.5).

4.4 Constrained float solution

The least-squares solution described previously solves Eq. (3.10) by estimating the baseline as well as the ambiguities which together, minimizes the squared noise. However, the solution might not be precise when the actual noise contained in the measurements is big or when the noise statistics are not correctly estimated. Error can be shifted from the ambiguities to the baseline or vice versa



(a)

(b)

(c)

Figure 4.5: Phase residuals with different selection strategy: a) Phase residuals after ambiguities fixing with constrained tree search. No epoch selection is performed; b) Phase residuals after ambiguities fixing with constrained tree search. Only code measurements with low WSSE are selected to perform the search. No selection on phase measurements; c) Phase residuals after ambiguities fixing with constrained tree search. Only code and phase measurements with low WSSE are selected to perform the search.

to minimize the noise during the estimation. However, given the configuration of our setting with a fix baseline, we can use the fix baseline length as a constraint to solve Eq. (3.10):

$$\Psi = H \underbrace{\begin{pmatrix} b \\ N \end{pmatrix}}_{\xi} + \varepsilon, \quad \text{with} \quad \|b\| = l_{ap}. \tag{4.18}$$

We use Lagrange multiplier to express the optimization (minimization) problem with a known constraint [8]. The problem can be formulated as below:

$$\Lambda(b, N, \mu) = \left\| \Psi - H \begin{pmatrix} b \\ N \end{pmatrix} \right\|^2_{\Sigma_\Psi^{-1}} + \mu(\|b\|^2 - l_{ap}^2). \tag{4.19}$$

By introducing a selection matrix $S = \left(I^{3\times 3}, 0^{3\times(K-1)}\right)^{\mathrm{T}}$, we can reformulate the optimization problem above as follows:

$$\begin{aligned}\varLambda(\xi,\mu) &= \|\varPsi - H\xi\|^2_{\varSigma_{\varPsi}^{-1}} + \mu(\|S\xi\|^2 - l_{\mathrm{ap}}^2)\\ &= (\varPsi - H\xi)^{\mathrm{T}}\varSigma_{\varPsi}^{-1}(\varPsi - H\xi) + \mu(\|S\xi\|^2 - l_{\mathrm{ap}}^2). \end{aligned} \tag{4.20}$$

The optimization problem is solved by finding staple points which correspond to when the partial derivative of the cost function $\varLambda$ is zero. If $\varSigma_{\varPsi}^{-1}$ is symmetric, we obtain:

$$\begin{aligned}\frac{\partial\varLambda(\xi,\mu)}{\partial\xi} = 0 &\Leftrightarrow (\varPsi - H\xi)^{\mathrm{T}}\cdot 2\varSigma_{\varPsi}^{-1}\cdot H = 2\mu\xi^{\mathrm{T}}S^{\mathrm{T}}S\\ &\Leftrightarrow (H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}H + \mu S^{\mathrm{T}}S)\xi = H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}\varPsi\\ &\Leftrightarrow \xi = (H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}H + \mu S^{\mathrm{T}}S)^{-1}H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}\varPsi. \end{aligned} \tag{4.21}$$

Note that for x a vector and A a matrix independent of x, the derivative of $x^{\mathrm{T}}Ax$ with respect to x is $x^{\mathrm{T}}(A + A^{\mathrm{T}})$.

Eq. (4.21) is then injected into the original equation of the constraint and we obtain

$$(H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}H + \mu S^{\mathrm{T}}S)^{-1}H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}\varPsi^{\mathrm{T}}S^{\mathrm{T}}S(H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}H + \mu S^{\mathrm{T}}S)^{-1}H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}\varPsi) - l_{\mathrm{ap}}^2 = 0. \tag{4.22}$$

Let $f(\mu)$ denote the entire left term. We observe that $f(\mu)$ can be rewritten as follows:

$$\begin{aligned}f(\mu) &= \|S(H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}H + \mu S^{\mathrm{T}}S)^{-1}H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}\varPsi)\|^2 - l_{\mathrm{ap}}^2 && (4.23)\\ &= \|S\xi(\mu)\|^2 - l_{\mathrm{ap}}^2 && (4.24)\\ &= (S\xi(\mu))^{\mathrm{T}}(S\xi(\mu)) - l_{\mathrm{ap}}^2. && (4.25)\end{aligned}$$

The above equation has unfortunately no close form solution; therefore, secant method is used to find the root with μ initialized to 0, which corresponds to the case of an unconstrained float solution. The $(n+1)$-th iteration is given by:

$$\mu^{(n+1)} = \mu^{(n)} - \left.\frac{f(\mu)}{f'(\mu)}\right|_{\mu=\mu^{(n)}}, \tag{4.26}$$

where

$$\begin{aligned}\frac{\partial}{\partial\mu}f(\mu) &= 2(S\xi)^{\mathrm{T}}S\frac{\partial}{\partial\mu}(\xi(\mu))\\ &= 2\xi^{\mathrm{T}}S^{\mathrm{T}}S\frac{\partial}{\partial\mu}(\xi(\mu)). \end{aligned} \tag{4.27}$$

Let $\tilde{\varLambda}(\mu) = H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}H + \mu S^{\mathrm{T}}S$, we can thus express $\xi(\mu)$ as follows:

$$\xi(\mu) = \tilde{\varLambda}^{-1}(\mu)H^{\mathrm{T}}\varSigma_{\varPsi}^{-1}\varPsi, \tag{4.28}$$

and the partial derivative of $\xi(\mu)$ is given by

$$\frac{\partial}{\partial\mu}(\xi(\mu)) = \frac{\partial}{\partial\mu}(\tilde{\Lambda}^{-1}(\mu)) \cdot H^{\mathrm{T}}\Sigma_{\Psi}^{-1}\Psi \tag{4.29}$$

$$= -\tilde{\Lambda}^{-1}(\mu)S^{\mathrm{T}}S\tilde{\Lambda}^{-1}(\mu)H^{\mathrm{T}}\Sigma_{\Psi}^{-1}\Psi. \tag{4.30}$$

Eq. 4.30 is obtained with the following:

$$I = \tilde{\Lambda}^{-1}(\mu) \cdot \tilde{\Lambda}(\mu) \Leftrightarrow 0 = \frac{\partial}{\partial\mu}(\tilde{\Lambda}^{-1}(\mu)) \cdot \tilde{\Lambda}(\mu) + \tilde{\Lambda}^{-1}(\mu) \cdot \frac{\partial}{\partial\mu}(\tilde{\Lambda}(\mu)) \tag{4.31}$$

$$\Leftrightarrow \frac{\partial}{\partial\mu}(\tilde{\Lambda}^{-1}(\mu)) = -\tilde{\Lambda}^{-1}(\mu)\underbrace{\frac{\partial}{\partial\mu}(\tilde{\Lambda}(\mu))}_{=S^{\mathrm{T}}S}\tilde{\Lambda}^{-1}(\mu). \tag{4.32}$$

The algorithm below describes step-by-step the secant method used to solve Eq. 4.23:

Algorithm 3 Secant method

```
$\mu_{\text{new}} = 0$
while $|\mu_{\text{new}} - \mu| < \Delta\mu_{\text{th}}$ do
    $\mu = \mu_{\text{new}}$
    $f(\mu) = \|S(H^{\mathrm{T}}\Sigma_{\Psi}^{-1}H + \mu S^{\mathrm{T}}S)^{-1}H^{\mathrm{T}}\Sigma_{\Psi}^{-1}\Psi)\|^2 - l_{\text{ap}}^2$
    $\tilde{\Lambda}(\mu) = H^{\mathrm{T}}\Sigma_{\Psi}^{-1}H + \mu S^{\mathrm{T}}S$
    $\frac{\partial}{\partial\mu}(\xi(\mu)) = -\tilde{\Lambda}^{-1}(\mu)S^{\mathrm{T}}S\tilde{\Lambda}^{-1}(\mu)H^{\mathrm{T}}\Sigma_{\Psi}^{-1}\Psi$
    $f'(\mu) = 2 \cdot \xi^{\mathrm{T}}(\mu)S^{\mathrm{T}}S\frac{\partial}{\partial\mu}(\xi(\mu))$
    $\mu_{\text{new}} = \mu^{(n)} - \frac{f(\mu)}{f'(\mu)}\Big|_{\mu=\mu^{(n)}}$
end while
```

4.4.1 Two dimensional baseline estimation

Fig. 4.6 shows the length convergence of baseline estimated with algorithm 3 performed using different measurement sets. ESA AZO is to simulate an urban environment as the measurements are taken in front of office buildings with concrete walls while Starnberger See is to simulate a suburban environment as the measurements are taken on a road which crosses a field. In both cases, measurements from 800 epochs were used to construct Ψ in Eq. 4.18 and the car remained stationary during the acquisition.

We observe that the length difference between the estimated baseline and the *a priori* baseline converges to less than 10^{-5} m after five iteration.

For on-road automobile use, the fixed baseline on the car roof (see Fig. 3.2) has a length of less than 2 m. Therefore, the relative height between both receivers in the local ENU frame can be neglected or assumed zero. With the assumption that the height component is known, we need fewer satellites to estimate the baseline, which is interesting in urban environment as visible satellites are rare. Fig. 4.7 illustrates length convergence of estimated baseline with algorithm 3 performed using the same measurement sets taken at ESA AZO as well as Starnberger See.

Similarly, after five iterations, the length difference between estimated baseline and the *a priori* baseline is less than 10^{-5}.

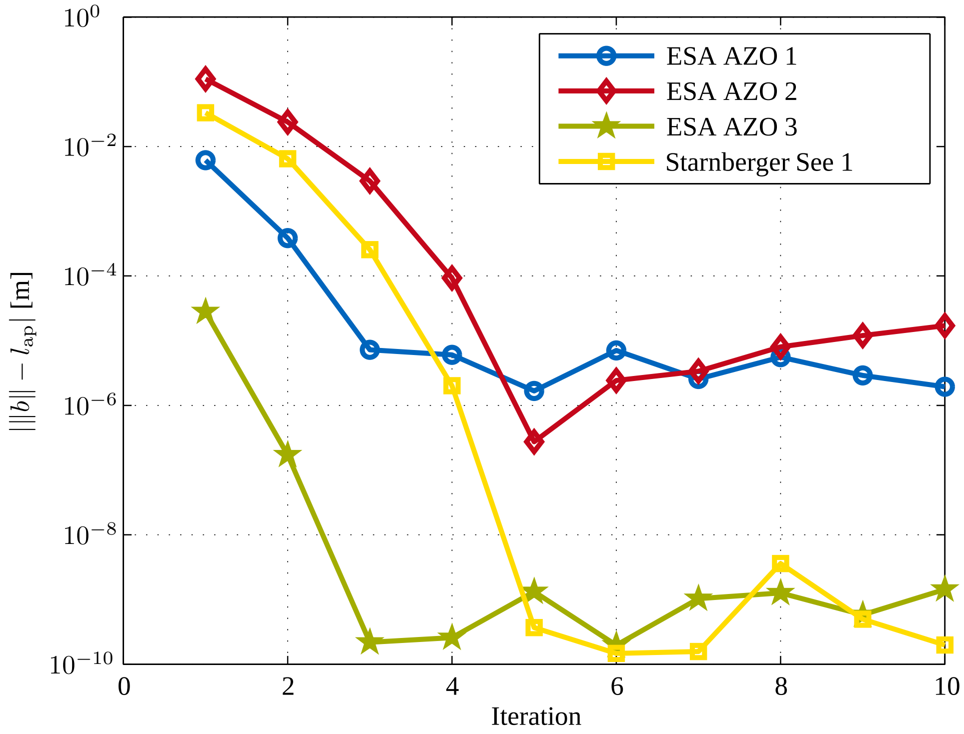


Figure 4.6: Convergence of 3-dimensional constrained float solution

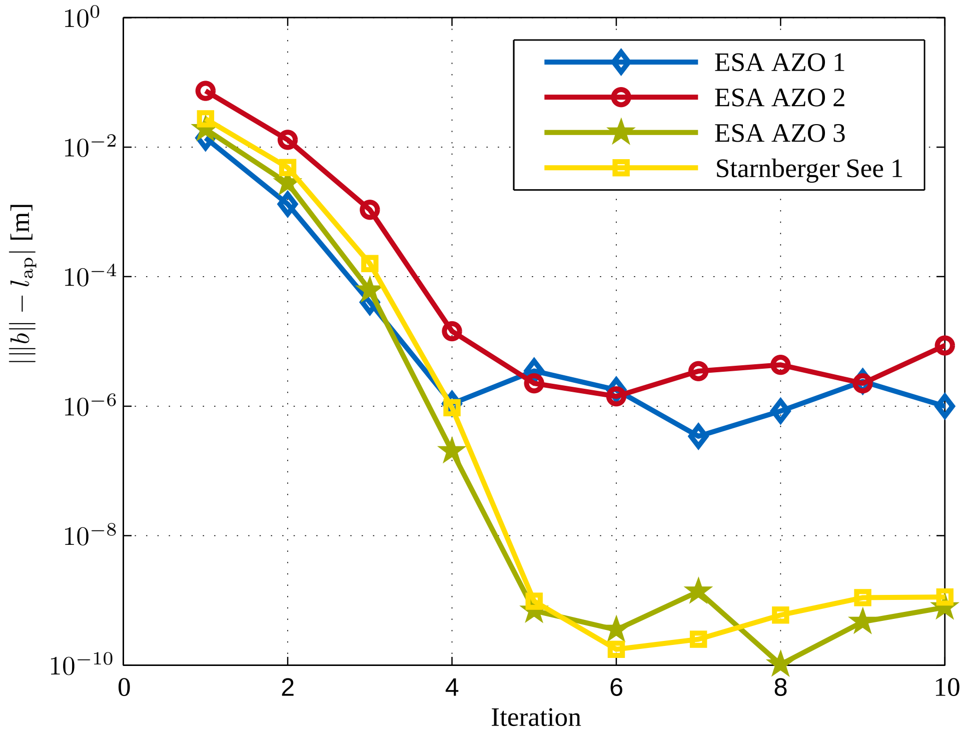


Figure 4.7: Convergence of 2-dimensional constrained float solution

5. Determination of fixed solution

This chapter focuses on the integer least-squares estimation of the baseline vector b and the double difference carrier phase integer ambiguities, i.e.

$$\min_{b \in \mathcal{R}^{2\times 1}, N \in \mathcal{Z}^{K-1\times 1}} \|\Psi - H_{\text{geo}} b + AN\|^2_{\Sigma_\Psi^{-1}}. \tag{5.1}$$

As we would like to reuse the float MAP estimate of the previous chapter, we have to perform a mapping from float to integer numbers. Blewitt [17] already proposed in 1989 a bootstrapping method, which performs conditional least-squares adjustment. However, bootstrapping is depending on the order of fixings and is not fully exploiting the complete correlation.

Today, Teunissen's Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method [3] is widely used. This chapter first reviews Teunissen's LAMBDA method as well as Henkel's MAP estimator. The latter one clearly outperforms both the unconstrained and constrained LAMBDA methods. The latter one uses an unconstrained search and then selects the integer candidates that comply with the baseline constraints. This implies the unnecessary calculation of numerous integer candidate vectors. We directly include baseline a priori information in the tree search, which leads to a shrinking of the search intervals.

We analyze the performance of the constrained search and compare it to the performance of the traditional LAMBDA search. Additionally, we also study the benefit of the use of an integer decorrelation with a Z-transformation for both the LAMBDA search and the constrained integer tree search.

5.1 LAMBDA method for integer ambiguity estimation

LAMBDA method implies four major steps. Sections 5.1.1 to 5.1.4 review these steps. A more detailed explanation can also be found in [18] and [9].

5.1.1 Float solution

The first step is performed by estimating b and N in $\mathbb{R}$. There are many ways to obtain a float solution for the underlined problem. In [18], Teunissen and Jonge suggested to use normal equations and Cholesky factorization. The Cholesky factor will also provide us a variance-covariance matrix of N which is required later in the fixed solution.

However, in our solution, we use iterative approaches described in the previous chapter (see Algo. 1 and 3) to obtain the float estimates of the baseline and the ambiguities. The estimates are noted as

$$\begin{bmatrix} \hat{b} \\ \hat{N} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \Sigma_{\hat{b}} & \Sigma_{\hat{b}\hat{N}} \\ \Sigma_{\hat{N}\hat{b}} & \Sigma_{\hat{N}} \end{bmatrix}, \tag{5.2}$$

where the latter one denotes the variance-covariance matrix of the float ambiguities.

5.1.2 Decorrelation

[11] shows that the correlation between ambiguities renders the integer search extremely inefficient. Teunissen observed a discontinuity in the spectrum of conditional standard deviations: the first ambiguity searches have much larger standard deviations followed by a sudden drop for the subsequent searches. This discontinuity is even more severe in the case of GPS double-difference ambiguities.

By decorrelating the float ambiguities, we can flatten the spectrum of conditional standard deviations. In other words, we eliminate the discontinuity by a decorrelation. Teunissen proposed in [11] to use the Z-transformation for decorrelation as not only it preserves the search space, it accepts integer entries. However, Z-transformation is not fully decorrelating the ambiguities but only reducing the correlation. Such trade-off is to ensure the integerness of the transformation [19].

The decorrelated ambiguities are obtained as,

$$\hat{N}_z = Z\hat{N}, \tag{5.3}$$

which could also be interpreted as a set of multi-satellite linear combinations. The variance-covariance matrix becomes an almost diagonal matrix and can be written with the triangular LDL^{T} decomposition as:

$$\begin{aligned}
\Sigma_{\Psi,z}^{-1} &= Z^{\mathrm{T}}\Sigma_{\Psi}^{-1}Z \\
&= Z^{\mathrm{T}}(LDL^{\mathrm{T}})^{-1}Z \\
&= Z^{\mathrm{T}}L^{\mathrm{T},-1}D^{-1}L^{-1}Z
\end{aligned} \tag{5.4}$$

In the following sections, ambiguities are always decorrelated. After fixing is accomplished, the decorrelation is reversed:

$$\check{N} = Z^{-1}\check{N}_z. \tag{5.5}$$

To increase readability, we omit the subscript “z" for decorrelated variables. Bear in mind that the ambiguities are always decorrelated during the integer search.

5.1.3 Discrete search

5.1.3.1 Problem separation

Since the ambiguities have to be solved in the integer space, a direct application of least-squares solution to obtain simultaneously the baseline and the ambiguities cannot be used. Teunissen [19] decomposed the error norm of Eq. 5.1 in order to separate the integer discrete search and the float baseline least-squares problem:

$$\|\Psi - H_{\mathrm{geo}}b - AN\|^2_{\Sigma_{\Psi}^{-1}} = \|\check{b}(N) - b\|^2_{\Sigma_{\check{b}(N)}^{-1}} + \|\hat{N} - N\|^2_{\Sigma_{\hat{N}}^{-1}} + \|P_{\bar{A}}^{\perp}P_{H_{\mathrm{geo}}}^{\perp}\Psi\|^2_{\Sigma_{\Psi}^{-1}}, \tag{5.6}$$

where $\check{b}(N)$ is the baseline estimated with fixed integer ambiguities, $P_{H_{\mathrm{geo}}}^{\perp}$ is the orthogonal projector on the space H_{geo} and $\bar{A} = P_{H_{\mathrm{geo}}}^{\perp}A$.

Günther derived the decomposition in [9]. The least-squares solution of the baseline is given by

$$\check{b}(N) = (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} (\varPsi - AN). \tag{5.7}$$

We can define a projection operator which projects into the hyperplane spanned by H_{geo}:

$$P_{H_{\text{geo}}} = H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1}. \tag{5.8}$$

With the above projector, we can also define its orthogonal projector with respect to the metric $\varSigma_{\varPsi}^{-1}$:

$$P_{H_{\text{geo}}}^{\perp} = I - P_{H_{\text{geo}}}, \tag{5.9}$$

such that $P_{H_{\text{geo}}}^{\perp} \varSigma_{\varPsi}^{-1} P_{H_{\text{geo}}} = 0$ and that $P_{H_{\text{geo}}}^{\perp} H_{\text{geo}} = 0$.

With these two projectors, we can express an arbitrary vector as a sum of two orthogonal vectors. Applying this concept to the error norm of Eq. 5.1, we obtain:

$$\begin{aligned} \|\varPsi - H_{\text{geo}} b - AN\|_{\varSigma_{\varPsi}^{-1}}^2 &= \|P_{H_{\text{geo}}}(\varPsi - H_{\text{geo}} b - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 + \|P_{H_{\text{geo}}}^{\perp}(\varPsi - H_{\text{geo}} b - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 \\ &= \|P_{H_{\text{geo}}}(\varPsi - AN) - H_{\text{geo}} b\|_{\varSigma_{\varPsi}^{-1}}^2 + \|P_{H_{\text{geo}}}^{\perp}(\varPsi - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 \end{aligned} \tag{5.10}$$

The second term of the above equation can be solved using a least-squares estimation, i.e.

$$\begin{aligned} \hat{N} &= \arg \min_{N \in \mathbb{Z}^{K-1}} \|P_{H_{\text{geo}}}^{\perp}(\varPsi - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 \\ &= \arg \min_{N \in \mathbb{Z}^{K-1}} \|P_{H_{\text{geo}}}^{\perp} \varPsi - \bar{A} N\|_{\varSigma_{\varPsi}^{-1}}^2 \\ &= (\bar{A}^{\text{T}} \varSigma_{\varPsi}^{-1} \bar{A})^{-1} \bar{A}^{\text{T}} \varSigma_{\varPsi}^{-1} (P_{H_{\text{geo}}}^{\perp} \varPsi), \end{aligned} \tag{5.11}$$

with $\bar{A} = P_{H_{\text{geo}}}^{\perp} A$ to simplify the notation. However, to further simplify the above equation, we will have to develop all variables with only geometry matrices and covariance matrix.

Let's first only develop $\bar{A}^{\text{T}} \varSigma_{\varPsi}^{-1} P_{H_{\text{L}}}^{\perp}$ in the above least-squares solution:

$$\begin{aligned} \bar{A}^{\text{T}} \varSigma_{\varPsi}^{-1} P_{H_{\text{geo}}}^{\perp} &= A^{\text{T}} P_{H_{\text{geo}}}^{\perp,\text{T}} \varSigma_{\varPsi}^{-1} P_{H_{\text{geo}}}^{\perp} \\ &= A^{\text{T}} (I - H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1})^{\text{T}} \varSigma_{\varPsi}^{-1} \\ &\quad (I - H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1}) \\ &= A^{\text{T}} (I - \varSigma_{\varPsi}^{-1} H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}}) \varSigma_{\varPsi}^{-1} \\ &\quad (I - H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1}) \\ &= A^{\text{T}} (\varSigma_{\varPsi}^{-1} - \varSigma_{\varPsi}^{-1} H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} \\ &\quad - \varSigma_{\varPsi}^{-1} H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} \\ &\quad + \varSigma_{\varPsi}^{-1} H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1}) \\ &= A^{\text{T}} \varSigma_{\varPsi}^{-1} (I - H_{\text{geo}} (H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1} H_{\text{geo}})^{-1} H_{\text{geo}}^{\text{T}} \varSigma_{\varPsi}^{-1}) \\ &= A^{\text{T}} \varSigma_{\varPsi}^{-1} P_{H_{\text{geo}}}^{\text{T}} \\ &= A^{\text{T}} P_{H_{\text{geo}}}^{\text{T}} \varSigma_{\varPsi}^{-1} \end{aligned}$$

$$= \bar{A}^{\mathrm{T}} \varSigma_{\varPsi}^{-1}. \tag{5.12}$$

The second last line is obtained using a general property of any arbitrary projection matrix P with the associated metric Q: $P^{\mathrm{T}}Q = QP$.

With the above simplification, we can therefore express $\hat{N}$ as follows:

$$\hat{N} = (\bar{A}^{\mathrm{T}} \varSigma_{\varPsi}^{-1} \bar{A})^{-1} \bar{A}^{\mathrm{T}} \varSigma_{\varPsi}^{-1} \varPsi. \tag{5.13}$$

Similar to the first decomposition step we made to obtain Eq 5.10, we define a projection operator which projects into the hyperplane spanned by $\bar{A}$:

$$P_{\bar{A}} = (\bar{A}^{\mathrm{T}} \varSigma_{\varPsi}^{-1} \bar{A})^{-1} \bar{A}^{\mathrm{T}} \varSigma_{\varPsi}^{-1}, \tag{5.14}$$

and by using the orthogonal projector $P_{\bar{A}}^{\perp}$ as well, we can decompose the second term of Eq. 5.10, i.e.

$$\|P_{H_{\mathrm{geo}}}^{\perp}(\varPsi - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 = \|P_{\bar{A}} P_{H_{\mathrm{geo}}}^{\perp}(\varPsi - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 + \|P_{\bar{A}}^{\mathrm{T}} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2. \tag{5.15}$$

If we try to solve N in the real-number space, the first term of the above equation can be made zero because $\varPsi$ is projected into the hyperplane spanned by $\bar{A}$. We can thus express $\varPsi = H_{\mathrm{geo}}b + A\hat{N}$, where $\hat{N}$ is determined with Eq. 5.13. This expression of $\varPsi$ allows the first term to be zero if $N \in \mathbb{R}^{K-1}$.

We can thus rewrite

$$\begin{aligned}
\|P_{H_{\mathrm{geo}}}^{\perp}(\varPsi - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 &= \|P_{\bar{A}} P_{H_{\mathrm{geo}}}^{\perp}(H_{\mathrm{geo}}b + A\hat{N} - AN)\|_{\varSigma_{\varPsi}^{-1}}^2 + \|P_{\bar{A}}^{\mathrm{T}} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2 \\
&= \|\bar{A}\hat{N} - \bar{A}N\|_{\varSigma_{\varPsi}^{-1}}^2 + \|P_{\bar{A}}^{\mathrm{T}} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2 \\
&= \|\hat{N} - N\|_{\varSigma_{N}^{-1}}^2 + \|P_{\bar{A}}^{\mathrm{T}} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2,
\end{aligned} \tag{5.16}$$

where $\varSigma_{\hat{N}}^{-1} = \bar{A}^{\mathrm{T}} \varSigma_{\varPsi}^{-1} \bar{A}$.

Eq. 5.16 provides us a decomposition of the second term of Eq. 5.10. The first term of Eq. 5.10 can also be simplified by using the fact that $\|P_{\bar{A}}^{\mathrm{T}} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2$ of Eq. 5.16 is an irreducible error independent from N. Therefore, we can first seek to minimize the second term of Eq. 5.10 by only looking at the error norm $\|\hat{N} - N\|_{\varSigma_{N}^{-1}}^2$. Then, we inject the resolved integer ambiguities $\check{N}$ into the first term of Eq. 5.10. The decomposition into three error norms becomes

$$\begin{aligned}
\|\varPsi - H_{\mathrm{geo}}b - AN\|_{\varSigma_{\varPsi}^{-1}}^2 &= \|P_{H_{\mathrm{geo}}}(\varPsi - AN) - H_{\mathrm{geo}}b\|_{\varSigma_{\varPsi}^{-1}}^2 + \|\hat{N} - N\|_{\varSigma_{\hat{N}}^{-1}}^2 + \|P_{\bar{A}}^{\perp} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2 \\
&= \|\check{b}(N) - b\|_{\varSigma_{\check{b}(N)}^{-1}}^2 + \|\hat{N} - N\|_{\varSigma_{\hat{N}}^{-1}}^2 + \|P_{\bar{A}}^{\perp} P_{H_{\mathrm{geo}}}^{\perp} \varPsi\|_{\varSigma_{\varPsi}^{-1}}^2,
\end{aligned} \tag{5.17}$$

with $\varSigma_{\check{b}(N)}^{-1} = H_{\mathrm{geo}}^{\mathrm{T}} \varSigma_{\varPsi}^{-1} H_{\mathrm{geo}}$.

The above decomposition allows us to separate real-valued and integer-valued minimization. With the decomposition, we can :

- First, determine the ambiguities in its integer space with respect to the metric $\varSigma_{\hat{N}}^{-1}$.
- Then, by using the fixed ambiguities, we determine the associated baseline in its real-number space with respect to the metric $\varSigma_{\check{b}(N)}^{-1}$ using a least-squares estimator.

5.1.3.2 Sequential conditional least-squares estimation

As mentioned above, we first try to minimize the following:

$$\min_{N} \|N - \hat{N}\|^2_{\Sigma_{\hat{N}}}, \quad \text{with} \quad N \in \mathbb{Z}^{K-1}. \tag{5.18}$$

If N is a real-number integer, taking the float solution $\hat{N}$ minimizes automatically the above. Since N is an integer vector, we have to perform a discrete search within an ellipsoid region χ^2. How χ^2 is defined for LAMBDA method can be found in [18].

The mathematical formulation of the integer search in χ^2 to minimize 5.18 is written as:

$$(N - \hat{N})^{\mathrm{T}} \Sigma^{-1}_{\hat{N}} (N - \hat{N}) \leq \chi^2. \tag{5.19}$$

Since $\Sigma^{-1}_{\hat{N}}$ is positive-definite, we can decompose $\Sigma^{-1}_{\hat{N}}$ into a product of diagonal matrix and triangular matrices [18]:

$$\Sigma^{-1}_{\hat{N}} = LDL^{\mathrm{T}} \tag{5.20}$$

where $L = \begin{pmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{(K-1)1} & l_{(K-1)2} & \cdots & l_{(K-1)(K-1)} \end{pmatrix}$ and $D = \begin{pmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_{K-1} \end{pmatrix}$.

Note that the $\Sigma^{-1}_{\hat{N}}$ can also be partitioned into $L^{\mathrm{T}}DL$.

On the one hand, the algebraic expansion of Eq. (5.19) with elements of L and D yields

$$\sum_{i=1}^{K-1} d_i[(N_i - \hat{N}_i) + \sum_{j=1}^{i-1} l_{ij}(N_j - \hat{N}_j)]^2 \leq \chi^2. \tag{5.21}$$

On the other hand, we can also interpret the discrete search of Eq. 5.19 in χ^2 statistically:

$$\sum_{i=1}^{K-1} \frac{(N_i - \hat{N}_{i|1,\ldots,i-1})^2}{\sigma^2_{\hat{N}_i|1,\ldots,i-1}} \leq \chi^2. \tag{5.22}$$

Ambiguities are often correlated to each other. Therefore, in the sequential search, we have to take into account the conditional adjustment on ambiguity parameters $\sigma^2_{\hat{N}_i|1,\ldots,i-1}$ which are already fixed.

The comparison of the algebraic expansion and the statistical formulation yields:

$$d_i^{-1} = \sigma^2_{\hat{N}_i|1,\ldots,i-1} \quad \text{and} \quad \hat{N}_{i|1,\ldots,i-1} = \hat{N}_i - \sum_{j=1}^{i-1} l_{ij}(N_j - \hat{N}_j). \tag{5.23}$$

Notice that if there is no correlation between different ambiguities, L is a diagonal, or more precisely, an identity matrix. Thus, in the absence of correlation, $\hat{N}_{i|1,\ldots,i-1} = \hat{N}_i$.

The sequential conditional adjustment can be done recursively by fixing first of all the first ambiguity and subsequently the following ambiguities by adjusting the search region. If N_i to $N_i - 1$ are already conditioned, the bound for the search of N_i is determined as follows:

$$((N_i - \hat{N}_i) + \sum_{j=1}^{i-1} l_{ij}(N_j - \hat{N}_j))^2 \leq \underbrace{\frac{\chi^2}{d_i} - \frac{1}{d_i}\sum_{l=1}^{i-1} d_l((N_l - \hat{N}_l) + \sum_{j=1}^{i-1} l_{lj}(N_j - \hat{N}_j))^2}_{\text{bound}_i}. \quad (5.24)$$

Hence, N_i is bounded by:

$$-\sqrt{\text{bound}_i} + \hat{N}_i - \sum_{j=1}^{i-1} l_{ij}(N_j - \hat{N}_j) \leq N_i \leq \sqrt{\text{bound}_i} + \hat{N}_i - \sum_{j=1}^{n} l_{ij}(N_j - \hat{N}_j). \quad (5.25)$$

All candidates of N_i within the search region is considered and N_i is subsequently fixed to the one which minimizes Eq. 5.18. The fixed N_i is denoted $\check{N}_i$.

5.1.4 Fixed baseline solution

After fixing the ambiguities, the baseline solution can be obtained directly using a linear minimum mean square error estimator [18]:

$$\check{b}(\check{N}) = \hat{b} - \Sigma_{\hat{b}\hat{N}}\Sigma_{\hat{N}}^{-1}(\hat{N} - \check{N}), \quad (5.26)$$

where variables with an inverse hat are determined after fixing.

The fixed baseline solution can also be determined using least-squares solution after fixing the double difference measurements, i.e. subtracting ambiguities from the measurement $\Psi_{\text{fixed}} = \Psi - A\check{N}$:

$$\check{b}(\check{N}) = (H_{\text{geo}}^{\text{T}}\Sigma_{\Psi}^{-1}H_{\text{geo}})^{-1}H_{\text{geo}}^{\text{T}}\Sigma_{\Psi}^{-1}\Psi_{\text{fixed}}. \quad (5.27)$$

5.2 Constrained integer tree search

LAMBDA method offers an efficient search which uses least-squares estimator to solve the ambiguities of a carrier-phase measurement. LAMBDA method sorts the ambiguity candidates based on the distance (in a metric sense governed by the measurement covariance matrix) to the ambiguity candidate determined in the float solution.

In the case where the distribution of the baseline vector is known, this *a priori* information on the baseline can be used with a *Maximum A Posteriori Probability (MAP)* estimator to estimate the ambiguities and determine the baseline. The fixing is improved as the determined integer ambiguities fulfill the baseline constraint. When only knowledge on the baseline length is available, a soft constraint can be imposed in the search [8].

5.2.1 Hard-constrained minimizer

We intend to solve Eq. (4.6). We assume in this subsection that *a priori* knowledge on the distribution of the baseline vector in the local ENU coordinates is available: $b_L \sim \mathcal{N}(b_{L,\text{ap}}, \Sigma_b)$. The determination of $b_{L,\text{ap}}$ must be as less dependent as possible on the measurement and preferably obtained from another source rather than measurements used in Ψ. $b_{L,\text{ap}}$ can be obtained either by using the Doppler shift or from two carrier smoothed absolute position estimates when the receivers are moving.

Since the *a priori* knowledge is known in the local ENU coordinates, it is thus more practical to rewrite Eq. (4.6) as

$$\Psi = H_{\mathrm{L}}\xi_{\mathrm{L}} + \eta \tag{5.28}$$

where

$$H_{\mathrm{L}} = \begin{pmatrix} H_{\mathrm{geo,L}} & A \end{pmatrix} \quad \text{with} \quad H_{\mathrm{geo,L}} = H_{\mathrm{geo}} R_{\mathrm{L}}^{-1}, \tag{5.29}$$

and

$$\xi_{\mathrm{L}} = \begin{pmatrix} b_{\mathrm{L}} \\ N \end{pmatrix} \quad \text{with} \quad b_{\mathrm{L}} = R_{\mathrm{L}} b. \tag{5.30}$$

In this section, we work with a baseline vector in the local ENU coordinate system as it is more practical to determine the heading. Note however that the concept still works with a baseline vector in the ECEF coordinate system.

The MAP estimator can be defined as follows [8] [20]:

$$\begin{pmatrix} \hat{b}_{\mathrm{L}} \\ \check{N} \end{pmatrix} = \arg \max_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{3\times 1} \\ N \in \mathbb{Z}^{K-1}}} P(\xi_{\mathrm{L}}|\Psi) \tag{5.31}$$

We can assume that we have *a priori* knowledge over the distribution of ξ_{L} by simply setting the mean of the ambiguities and the variance as infinity. This assumption is as good as having no *a priori* knowledge. Therefore, Eq. 5.28 can be reformulated and simplified using a monotonously decreasing logarithmic function:

$$\begin{pmatrix} \hat{b}_{\mathrm{L}} \\ \check{N} \end{pmatrix} = \arg \max_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{3\times 1} \\ N \in \mathbb{Z}^{K-1}}} P(b_{\mathrm{L}}, N|\Psi) \tag{5.32}$$

$$= \arg \max_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{3\times 1} \\ N \in \mathbb{Z}^{K-1}}} \frac{P(\Psi|b_{\mathrm{L}}, N)P(b_{\mathrm{L}})P(N)}{P(\Psi)} \tag{5.33}$$

$$= \arg \min_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{3\times 1} \\ N \in \mathbb{Z}^{K-1}}} -\log\left(\frac{P(\Psi|b_{\mathrm{L}}, N)P(b_{\mathrm{L}})P(N)}{P(\Psi)}\right) \tag{5.34}$$

$$= \arg \min_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{3\times 1} \\ N \in \mathbb{Z}^{K-1}}} -\log\left(P(\Psi|b_{\mathrm{L}}, N)P(b_{\mathrm{L}})P(N)\right) \tag{5.35}$$

$$= \arg \min_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{3\times 1} \\ N \in \mathbb{Z}^{K-1}}} \|\Psi - H_{\mathrm{L}}\xi\|^2_{\Sigma_{\Psi}^{-1}} + \|b_{\mathrm{L}} - b_{\mathrm{L},ap}\|^2_{\Sigma_{b_{\mathrm{L}}}^{-1}} \tag{5.36}$$

Searching for an argument to maximize the probability is the same as searching for an argument to maximize the logarithm of the probability due to the monotony of the logarithmic function. Taking the negative of the function inverses then the maximization to a minimization (see Eq. 5.34). Given the ambiguities and the baseline vector, carrier-phase measurement is assumed to be Gaussian distributed as the measurement follows a Gaussian distribution. The baseline vector is assumed to follow a Gaussian distribution. Therefore, minimizing the logarithm of the joint Gaussian probabilities is the same as minimizing the sum of the the squared norms because the density function of a normally distributed random vector with k elements $x \sim \mathcal{N}(\mu_x, \Sigma_x)$ is given by:

$$f_x(x_1, \cdots, x_k) = \frac{1}{\sqrt{(2\pi)^k |\Sigma_x|}} \exp\left(-\frac{1}{2}(x-\mu_x)^{\mathrm{T}} \Sigma_x^{-1}(x-\mu_x)\right). \tag{5.37}$$

However, $P(\Psi)$ no longer plays a role in the minimization as the probability is independent of ξ_{L} (see Eq. 5.35). This independency can be explained by

$$P(\Psi) = \int\limits_{\mathbb{R}^3} \sum_{N \in \mathbb{Z}^{N-1}} P(\Psi, b_{\mathrm{L}}, N) N \, \mathrm{d}b_{\mathrm{L}}. \tag{5.38}$$

Without any *a priori* knowledge on the ambiguities, the search can be extended to the whole natural number space, $\mathbb{Z}^{N-1}$, which is extremely inefficient and even infeasible. We can then limit the search within a defined volume like in LAMBDA method and the minimizer becomes

$$\min_{b_{\mathrm{L}}} \left(\|\Psi - H_{\mathrm{L}} b_{\mathrm{L}} - AN\|^2_{\Sigma_\Psi^{-1}} + \|b_{\mathrm{L}} - b_{\mathrm{L,ap}}\|^2_{\Sigma_{b_{\mathrm{L}}}^{-1}} \right) \leq \chi^2. \tag{5.39}$$

How χ^2 is determined will be explained in 5.2.4.

In this work, we do not use the hard-constrained minimizer as the *a priori* baseline vector is not available since the ambiguities have to be fixed while the car is stationary and the Doppler shift of u-blox LEA 6T is too noisy to be exploited for determining the b_{L}. The search using a hard-constrained minimizer is however similar to the tree search explained below for the soft-constrained minimizer.

5.2.2 Soft-constrained minimizer

In the case where no *a priori* knowledge on the baseline vector is available but only on the baseline length is known, we can modify the minimizer using a *Soft Length Constraint (SLC)* instead of a hard constraint on the complete vector:

$$\min_{b_{\mathrm{L}}} \left(\|\Psi - H_{\mathrm{L}} b_{\mathrm{L}} - AN\|^2_{\Sigma_\Psi^{-1}} + \frac{(\|b_{\mathrm{L}}\| - l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}} \right) \leq \chi^2, \tag{5.40}$$

where l_{ap} is the known baseline length and $\sigma^2_{l_{\mathrm{ap}}}$ is the variance of the baseline length.

LAMBDA integer search for ambiguities have a final goal to minimize Eq. 5.17. The third term of the error sum depends solely on the measurement. Therefore, it can be considered as irreducible noise. As mentioned earlier in section 5.1.3.1, the search consists of in the first place finding integer ambiguities which minimize the ambiguity error. The search for integer ambiguities is said to be performed partially as it uses a sequential conditional adjustment to take into account

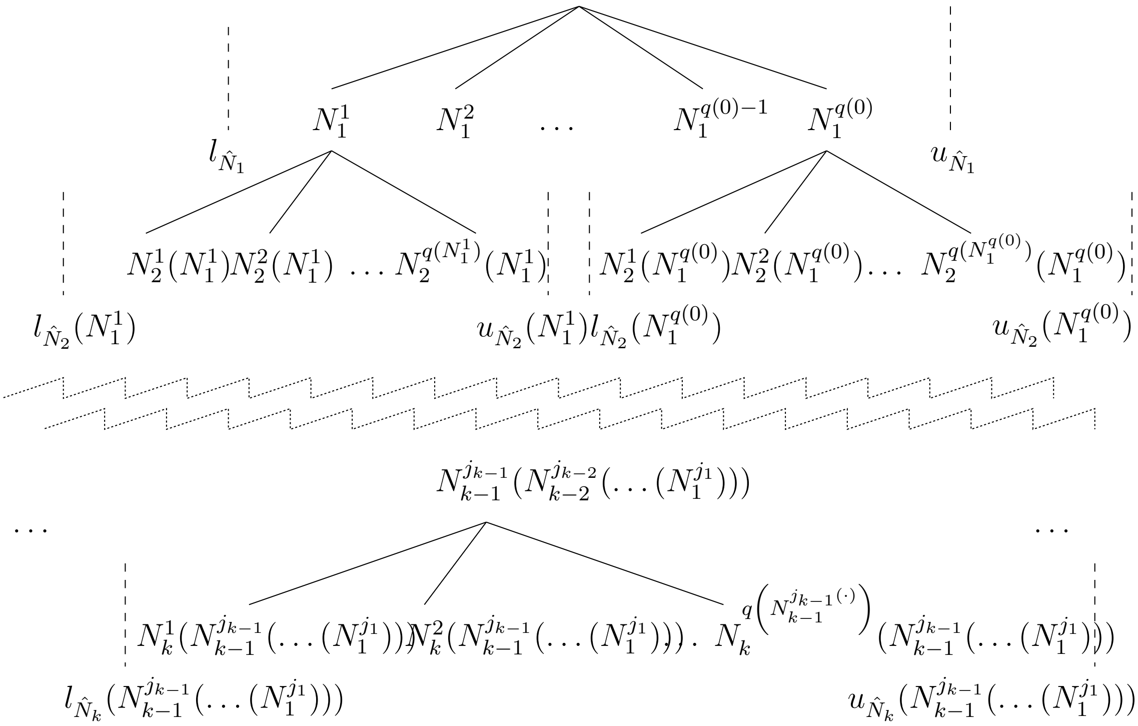

N_1^1
N_1^2
$\dots$
$N_1^{q(0)-1}$
$N_1^{q(0)}$
$l_{\hat{N}_1}$
$u_{\hat{N}_1}$
$N_2^1(N_1^1)N_2^2(N_1^1)\ \dots\ N_2^{q(N_1^1)}(N_1^1)$
$N_2^1(N_1^{q(0)})N_2^2(N_1^{q(0)})\dots\ N_2^{q(N_1^{q(0)})}(N_1^{q(0)})$
$l_{\hat{N}_2}(N_1^1)$
$u_{\hat{N}_2}(N_1^1)l_{\hat{N}_2}(N_1^{q(0)})$
$u_{\hat{N}_2}(N_1^{q(0)})$
$N_{k-1}^{j_{k-1}}(N_{k-2}^{j_{k-2}}(\dots(N_1^{j_1})))$
$\dots$
$\dots$
$N_k^1(N_{k-1}^{j_{k-1}}(\dots(N_1^{j_1})))N_k^2(N_{k-1}^{j_{k-1}}(\dots(N_1^{j_1})))\ .\ N_k^{q\left(N_{k-1}^{j_{k-1}(\cdot)}\right)}(N_{k-1}^{j_{k-1}}(\dots(N_1^{j_1})))$
$l_{\hat{N}_k}(N_{k-1}^{j_{k-1}}(\dots(N_1^{j_1})))$
$u_{\hat{N}_k}(N_{k-1}^{j_{k-1}}(\dots(N_1^{j_1})))$

Figure 5.1: Search tree for MAP/ SLC ambiguity resolution: Each row refers to the integer candidates of one double difference ambiguity. For each candidate, a lower and un upper bound are derived for the subsequent double difference integer ambiguity. If there is no integer ambiguity in the interval between the lower and upper bound, then this path is no longer considered. Otherwise, for each candidate inside the set, a lower and an upper bound are determined for the subsequent ambiguity.

the correlation between ambiguities. With the found ambiguities, we then solve the least-squares problem posed by the first quadratic error norm to determine the associated fixed baseline vector.

With a soft-constrained minimizer, the search will be slightly different. We still use the sequential conditional fixing. Therefore the search resembles a tree search but each layer of the tree search is conditioned by Eq. 5.40. With the latter, a lower and an upper bound are derived and only candidates within the bounds are considered. If at one layer no integer ambiguity is available between the bounds, the path is no longer considered. This method which aims to achieve a higher efficiency in the integer search was developed by Henkel et al. [8] [20].

If the sequential conditional adjustment is done for k ambiguities, the ambiguities can be subdivided into a set of integer valued and a set of real-valued ambiguities with the real-valued ambiguities grouped together with the baseline coordinates:

$$\tilde{N} = \begin{pmatrix} N^1 \\ \vdots \\ N^k \end{pmatrix} \in \mathbb{Z}^{k\times 1}, \quad \tilde{\xi}_\mathrm{L} = \begin{pmatrix} b_\mathrm{L} \\ N^{k+1} \\ \vdots \\ N^{K-1} \end{pmatrix} \in \mathbb{R}^{(3+K-1-k)\times 1}$$

The error decomposition becomes

$$\|\Psi - H_\mathrm{L}\xi_\mathrm{L} - AN\|^2_{\Sigma_\Psi^{-1}} = \|\hat{\tilde{N}} - \check{\tilde{N}}\|^2_{\Sigma^{-1}_{\hat{\tilde{N}}}} + \|\check{\tilde{\xi}}(\tilde{N}) - \hat{\tilde{\xi}}\|^2_{\Sigma^{-1}_{\check{\tilde{\xi}}(\tilde{N})}} + \|P^\perp_{\tilde{A}} P^\perp_{H_\mathrm{L}} \tilde{\Psi}\|^2_{\Sigma_\Psi^{-1}}, \tag{5.41}$$

where $\check{\tilde{\xi}}(\tilde{N})$ is the baseline found by using the ambiguities

$$\begin{pmatrix} \check{\tilde{N}} \\ \hat{N}_{k+1} \\ \vdots \\ \hat{N}_{K-1} \end{pmatrix} \tag{5.42}$$

concatenated with the real-valued ambiguities.

We can perform an LDL^{T} on the ambiguity error norm like in Eq. 5.20:

$$\|\hat{\tilde{N}} - \check{\tilde{N}}\|^2_{\Sigma^{-1}_{\hat{\tilde{N}}}} = \sum_{i=1}^{k} \frac{(\check{N}_i - \hat{N}_{i|1,\ldots,i-1})^2}{\sigma^2_{\hat{N}_{i|1,\ldots,i-1}}}, \tag{5.43}$$

with $\sigma^2_{\hat{N}_{i|1,\ldots,i-1}} = d_i^{-1}$ the conditional float ambiguity variance (see Eq. 5.23).

We now intend to fix the next ambiguity N_{k+1}. In other words, we are at $(k+1)$-th layer of the tree search (see Fig. 5.1). In order for an integer to be accepted as N_{k+1}, it has to fulfill the inequality below obtained by combining Eq. 5.43 and 5.40:

$$\begin{aligned} \frac{(N_{k+1} - \hat{N}_{k+1|1,\ldots,k})^2}{\sigma^2_{\hat{N}_{k+1|1,\ldots,k}}} \quad \leq \quad & \chi^2 - \|P^{\perp}_{\bar{A}} P^{\perp}_{H_{\mathrm{L}}} \Psi\|^2_{\Sigma^{-1}_{\Psi}} - \sum_{i=1}^{k} \frac{(\check{N}_i - \hat{N}_{i|1,\ldots,i-1})^2}{\sigma^2_{\hat{N}_{i|1,\ldots,i-1}}} \\ & - \min_{\tilde{\xi}} \left(\|\check{\tilde{\xi}}(\tilde{N}) - \hat{\tilde{\xi}}\|^2_{\Sigma^{-1}_{\check{\tilde{\xi}}(\tilde{N})}} + \frac{(\|S\check{\tilde{\xi}}(\tilde{N})\| - l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}} \right), \end{aligned} \tag{5.44}$$

where $S = (1_{3\times3}, 0_{3\times(K-1-k)})$.

The minimization of the last term can be done by using the float baseline vector hard-constrained float solution described in section 4.4. This length-constrained baseline concatenated with the real-valued ambiguities is denoted by $\check{\tilde{\xi}}_{\mathrm{opt}}(\tilde{N})$ and

$$\min_{\tilde{\xi}} \left(\|\check{\tilde{\xi}}(\tilde{N}) - \hat{\tilde{\xi}}\|^2_{\Sigma^{-1}_{\check{\tilde{\xi}}(\tilde{N})}} + \frac{(\|S\check{\tilde{\xi}}(\tilde{N})\| - l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}} \right) = \|\check{\tilde{\xi}}_{\mathrm{opt}}(\tilde{N}) - \hat{\tilde{\xi}}\|^2_{\Sigma^{-1}_{\check{\tilde{\xi}}(\tilde{N})}}. \tag{5.45}$$

Solving the inequality 5.44 gives us a lower and upper bound for N_{k+1} which respectively are expressed as:

$$\begin{aligned} l_{\hat{N}_{k+1}} &= \left\lfloor \hat{N}_{k+1|1,\ldots,k} - \sigma_{\hat{N}_{k+1|1,\ldots,k}} \sqrt{A_{k+1}(\tilde{N})} \right\rfloor \\ u_{\hat{N}_{k+1}} &= \left\lceil \hat{N}_{k+1|1,\ldots,k} + \sigma_{\hat{N}_{k+1|1,\ldots,k}} \sqrt{A_{k+1}(\tilde{N})} \right\rceil, \end{aligned} \tag{5.46}$$

with

$$A_{k+1}(\tilde{N}) \quad = \quad \chi^2 - \|P^{\perp}_{\bar{A}} P^{\perp}_{H} \tilde{\Psi}\|^2_{\Sigma^{-1}_{\Psi}} - \sum_{l=1}^{k-1} \frac{(N_l - \hat{N}_{l|1,\ldots,l-1})^2}{\sigma^2_{\hat{N}_{l|1,\ldots,l-1}}}$$

$$-\left(\|\check{\tilde{\xi}}_{\mathrm{opt}}(\tilde{N})-\hat{\tilde{\xi}}\|^2_{\Sigma^{-1}_{\check{\tilde{\xi}}(\tilde{N})}}+\frac{(\|S\check{\tilde{\xi}}_{\mathrm{opt}}(\tilde{N})\|-l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}}\right). \tag{5.47}$$

These upper and lower bound can be used to limit the search space for N^{k+1}. Integers within the bounds can be the candidates of $(k+1)$-th layer in the tree search.

We repeat the process and search for the next candidate. The overall search resembles a tree-search whereby only branches which survive at the most bottom layer are eligible candidates for integer ambiguities.

Finally, among all candidates, we select one which minimizes Eq. 5.40.

5.2.3 Evaluation of soft-constrained tree search

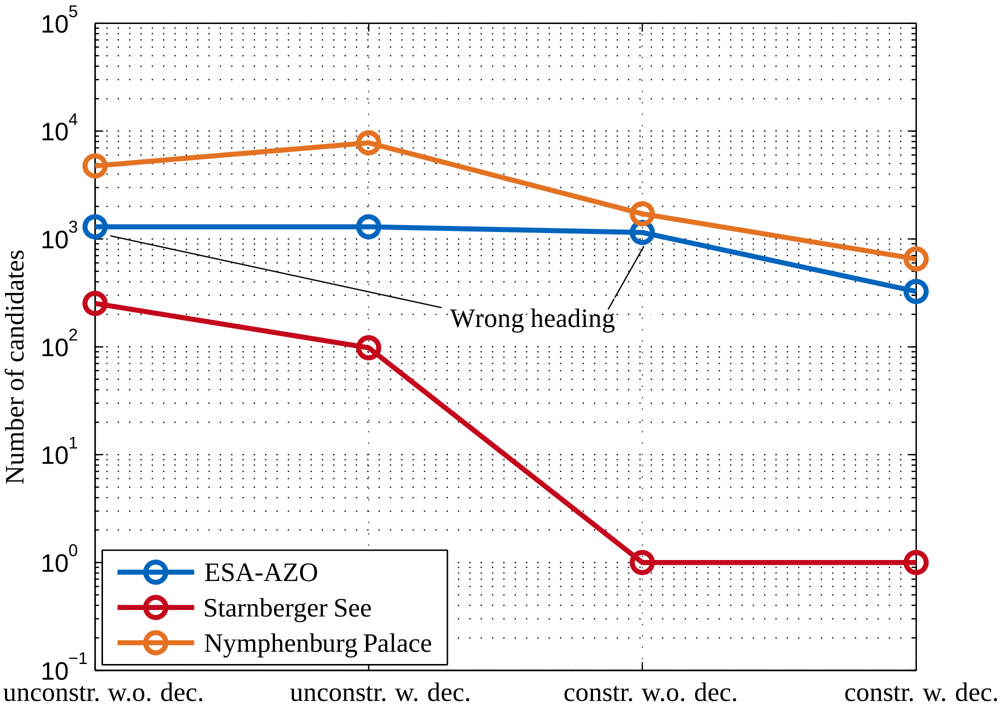


Figure 5.2: Comparison of constrained (constr.), unconstrained (unconstr.) integer tree search either with (w.) or without (w.o.) decorrelation (dec.)

By subtracting the last two terms from χ^2 in Eq. 5.47, the search space is substantially reduced at each layer compared to the unconstrained integer search tree.

Figure 5.2 compares the soft-constrained and the unconstrained integer ambiguity search using three different sets of data. ESA-AZO simulates an urban environment as the data was collected right in front of a building (see Fig. 2.2). Starnberger See simulates a suburban environment while Nymphenburg Palace simulates a relatively calm urban environment.

Note that all integer ambiguity resolutions are correct except for the ESA-AZO data set where decorrelation was not used. Whether the ambiguities are decorrelated or not, results show that the unconstrained integer search is less efficient in the sense that it finds more candidates given a search space χ^2. The search tree is thus more elaborated and more irrelevant branches are produced at each layer which can be eliminated by using a constrained integer search.

Besides, without decorrelating the ambiguities, the resolution of integer ambiguities also becomes less reliable as one can conclude from the wrong heading determination with the test drive at ESA-AZO.

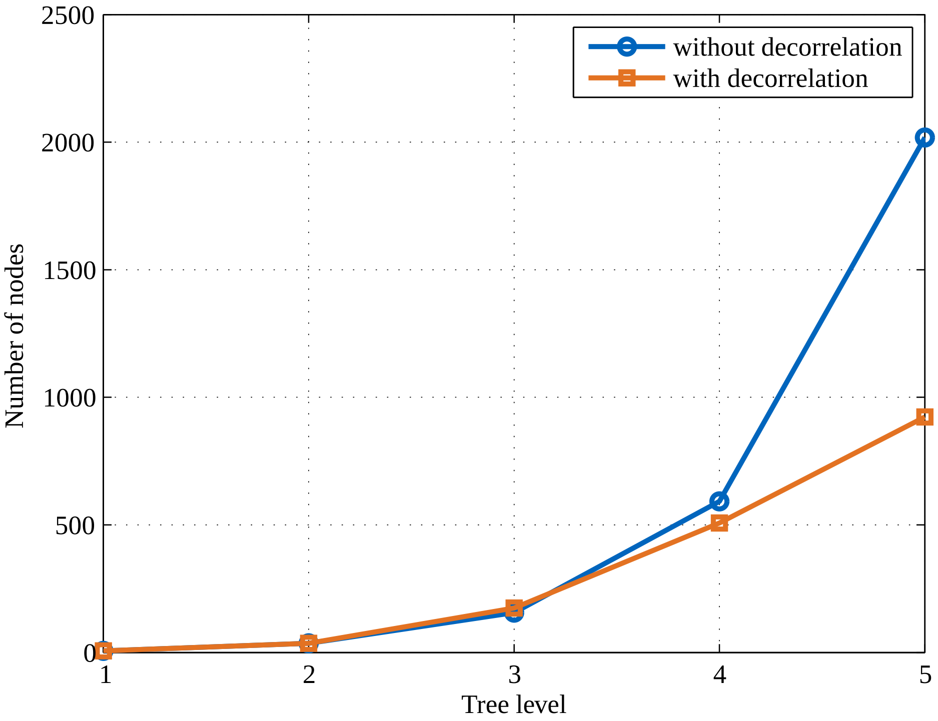


Figure 5.3: The flattening of standard deviation spectrum is reflected in the number of nodes at each layer of the soft-constrained integer tree search. The discontinuity is reduced with Z-transformation.

Figure 5.3 shows number of nodes found at each layer of the soft-constrained integer tree search. The sudden increase of the number of nodes can be observed in the case where no decorrelation was done while the number of nodes increases gradually in the case where ambiguities are decorrelated. The results conforms with Teunissen's observation in [11]: Without decorrelation with Z-transformation, the spectrum of conditional standard deviations is discontinued and is reflected by a sudden increase of nodes at the last few tree search layers.

Fig. 5.4 shows the double difference phase residuals during the initial 160 s after integer ambiguities are fixed. The car was standing during the measurement acquisition. The phase residuals of all four double differences are less than two centimeters over the complete period, which indicates a correct integer ambiguity resolution.

5.2.4 Search volume

The discrete integer search of ambiguities is conditioned by Eq. 5.39. The search space volume can be adjusted as wished upon condition that at least one integer candidate is contained within the search volume. Teunissen [18] had proposed various methods to define an appropriate search space, for instance by defining the number of candidates one wishes to obtain.

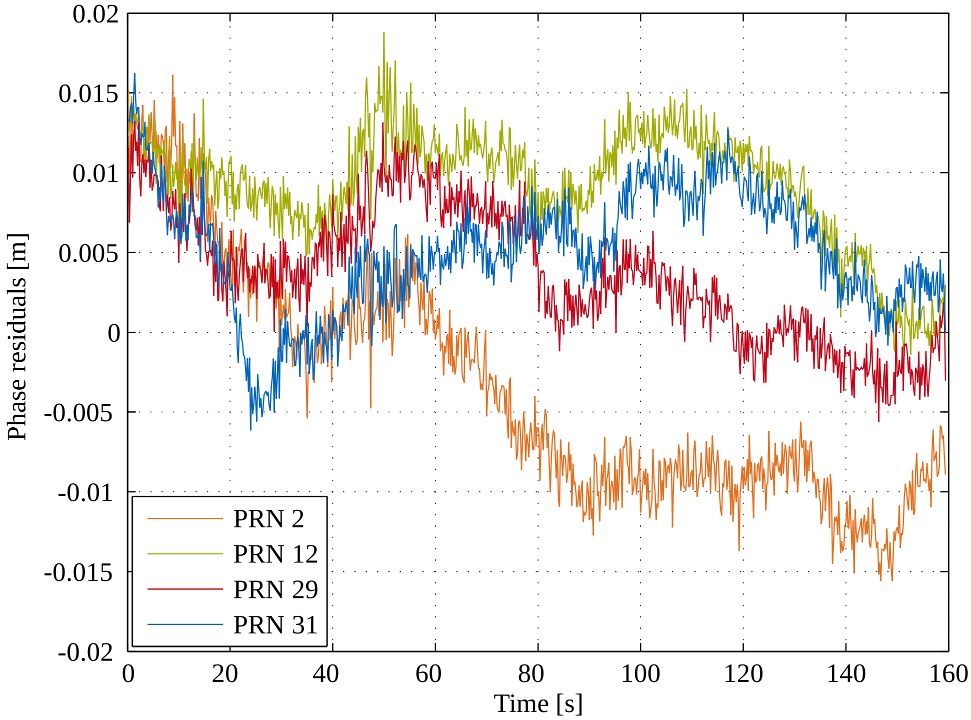


Figure 5.4: Phase residuals during initial integer ambiguity resolution.

Since the constrained integer tree search is to optimize the efficiency of the search, instead of defining the number of candidates that we wish to evaluate, we define the search space χ^2 in such a way that it contains *at least* one candidate and evaluates all ambiguity candidates found within the search space. Since the unconstrained tree search is less efficient compared with the constrained tree search (see Fig. 5.2), the search space defined using one of the candidates given by the unconstrained integer search ensures that at least one candidate will be found with the constrained integer search.

We take for instance the best candidate (i.e. the candidate nearest to the float ambiguities in a metric sense) given by the unconstrained integer search $\check{N}_{\text{unconstr}}$ to define the search space for constrained tree search χ^2_{constr}:

$$\chi^2_{\text{constr}} = \|\Psi - H_{\text{L}}\check{b}_{\text{L}}(\check{N}_{\text{unconstr}}) - A\check{N}_{\text{unconstr}}\|^2_{\Sigma_\Psi^{-1}} + \frac{(\|\check{b}_{\text{L}}(\check{N}_{\text{unconstr}})\| - l_{\text{ap}})^2}{\sigma^2_{l_{\text{ap}}}} \tag{5.48}$$

We know that we will obtain by the end of the constrained tree search at least one ambiguity candidate set, which is $\check{N}_{\text{unconstr}}$ itself. The above equation can be further simplified to only one term instead of a sum of two weighted quadratic residuals by determining the *hard-constrained float baseline solution* $\check{b}_{\text{L,opt}}(\check{N}_{\text{unconstr}})$. With the hard-constrained baseline, the second term which indicates the baseline *a priori* residuals becomes negligible and therefore, we can define the search volume with the following

$$\chi^2_{\text{constr}} = \|\Psi - H_{\text{L}}\check{b}_{\text{L,opt}}(\check{N}_{\text{unconstr}}) - A\check{N}_{\text{unconstr}}\|^2_{\Sigma_\Psi^{-1}}. \tag{5.49}$$

6. Cascaded cycle slip correction

This chapter provides a *cascaded* cycle slip detection and correction method for *low cost* GPS receivers. The cycle slip correction is a crucial point as

- $\lambda/2$ cycle slips occur due to undetected changes of the navigation bit;
- receivers occasionally track a reflection rather than the direct signal;
- multiple satellites might be affected simultaneously;
- phase changes due to high receiver dynamics have to be separated from cycle slips.

Challenging environments include narrow street canyons and alleys, where the direct signals might be shadowed. We strengthen the cycle slip detection and correction by taking the following information into account:

- fixed phase residuals of all double differences and their history;
- heading of previous epochs and its prediction to the current epoch;
- length of the baseline between both receivers;
- code-based absolute velocity of the vehicle.

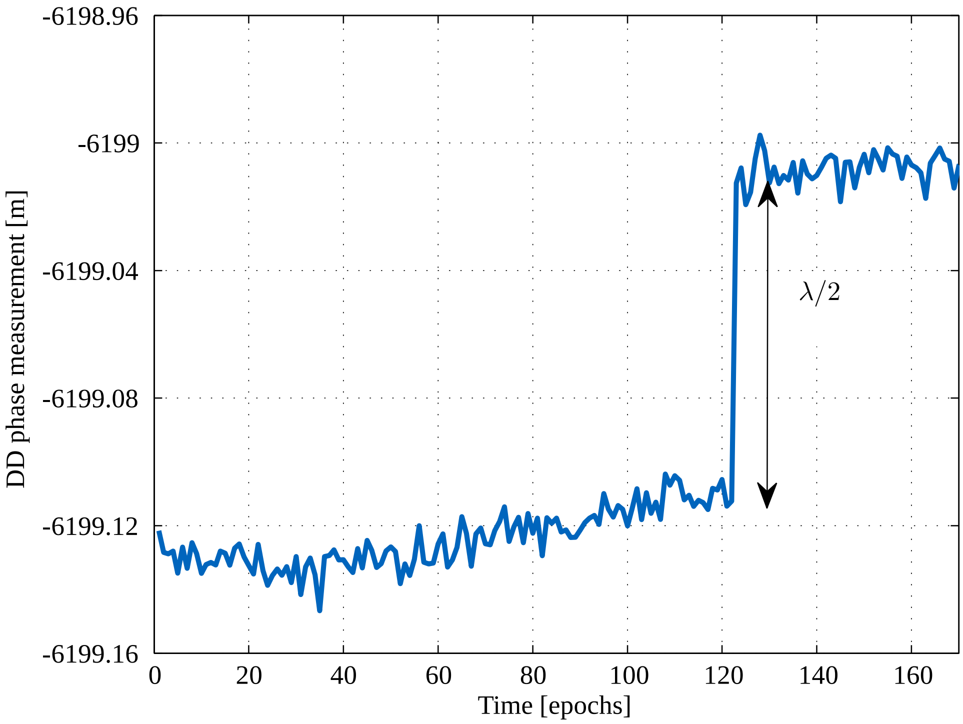


Figure 6.1: Phase double difference measurement of one satellite when the vehicle is stationary. At epoch 122, cycle slip happens and causes a jump in the measurement.

Fig. 6.1 shows the temporal evolution of the double carrier phase measurement for *stationary* receivers. Under normal conditions, the double difference phase measurements of stationary receivers only slightly drift due to the satellite movement. However, when cycle slips happen, sudden jumps can be observed in the phase measurement like in Fig. 6.1.

A cycle slip means a jump of a wavelength λ in the phase measurement scaled to length unit. For geodetic receivers, cycle slips are always an integer multiple of a wavelength. Unfortunately, with low-cost receivers, cycle slips can be also a multiple of $\lambda/2$, as shown in Fig. 6.1. The half cycle slip originates from an undetected bit transition in the navigation message, which leads to a jump by $\pi/2$.

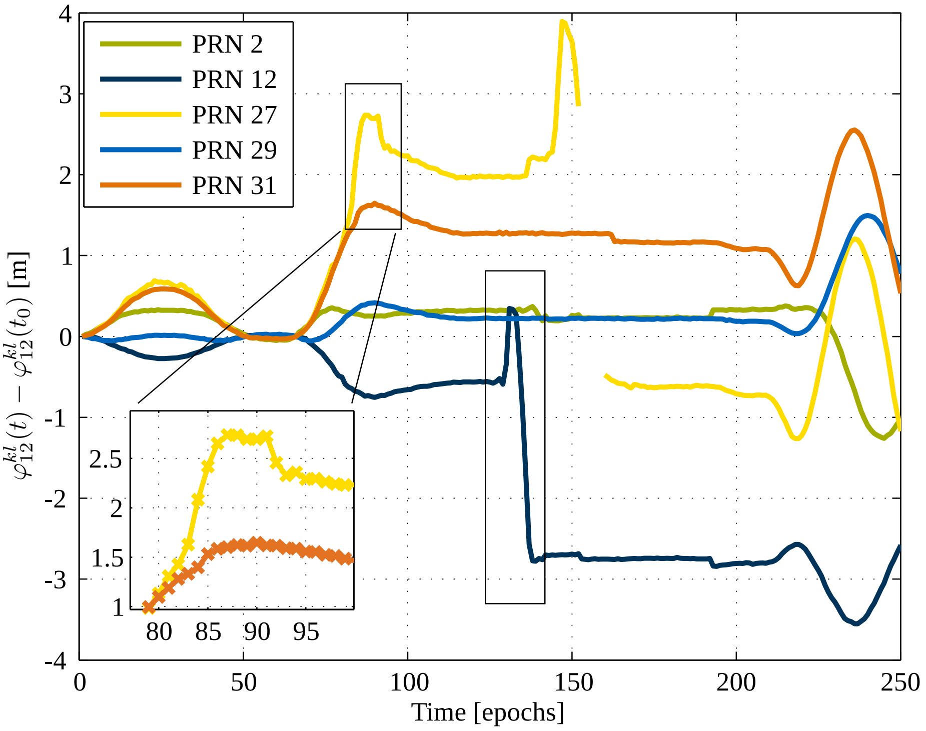


Figure 6.2: Phase double difference of visible satellites subtracted with the measurement of the first epoch. Measurements were acquired when the vehicle was moving.

Cycle slips cannot be detected in a single phase measurement due to satellite movement. However, it is possible to detect cycle slips in double difference measurements which are important in our case as heading is derived from double difference measurements. Fig. 6.1 shows a situation where cycle slips can be easily recognized and solved formally by taking the difference of two consecutive measurements. We however do not always have such easily identifiable jumps especially when the receivers are moving. Fig. 6.2 shows the double difference phase measurements of a car moving in an urban environment. The double differences vary substantially during the movement and make cycle slips harder to recognize with naked eyes and even more complicated to identify formally. Furthermore, cycle slips can affect simultaneously multiple satellites. Therefore, cross checking double difference phase measurements can also not help.

In the case of L1 band being used, a cycle slip of half a cycle can account for at least 9 cm of error in the measurement and therefore introduce an error of $\approx 0.1^\circ$ in the heading if we have a baseline of 1 m. The error can be increased in the case where cycle slips affect simultaneously several satellites. If left uncorrected, this error can be accumulative: When cycle slip happens at one epoch, measurements of the subsequent epochs will also be affected (see Fig. 6.1).

6.1 Phase measurement model with cycle slip considered

To adapt to the use of *low cost* GPS receiver, the double difference measurement model described in Eq. 3.6 will be refined in this chapter to take into account $\lambda/2$ cycle slips:

$$\varphi_{12}^{k} - \varphi_{12}^{l} \approx (\vec{e}_1^{\,k} - \vec{e}_1^{\,l})^{\mathrm{T}} \cdot \vec{b}_{12} + c_{12}^{kl} + \lambda N_{12}^{kl} + \frac{\lambda}{2}\mathit{\Delta} N_{12}^{kl} + m_{\varphi,12}^{kl} + \varepsilon_{12}^{kl}. \tag{6.1}$$

We define here the term “fixed double difference phase measurement", which is simply calculated by subtracting the resolved double difference ambiguities $\check{N}$ from the double difference phase measurement:

$$\varphi_{12,\mathrm{fixed}}^{kl} = \varphi_{12}^{kl} - \lambda \check{N}_{12}^{kl} \tag{6.2}$$

In the case where cycle slip happens, we have to take the number of cycle slips $\mathit{\Delta} N$ into account in the fixed double difference measurement. By mapping multipath to noise, the above equation becomes:

$$\varphi_{12,\mathrm{fixed}}^{kl} \approx (\vec{e}_2^{\,k} - \vec{e}_2^{\,l})^{\mathrm{T}} \cdot \vec{b}_{12} + \frac{\lambda}{2}\mathit{\Delta} N_{12}^{kl} + \varepsilon_{12}^{kl}. \tag{6.3}$$

By grouping double difference measurements obtained from different satellites with respect to the reference satellite ($\mathrm{PRN} = 1$), the above equation can then be written in matrix-vector representation:

$$\begin{aligned}\varphi_{\mathrm{dd,fixed}} &= \begin{bmatrix} \varphi_{12,\mathrm{fixed}}^{21} \\ \varphi_{12,\mathrm{fixed}}^{31} \\ \vdots \\ \varphi_{12,\mathrm{fixed}}^{K1} \end{bmatrix} \\ &= \tilde{H}_{\mathrm{geo}} b_{12} + A_{\mathrm{cs}} \mathit{\Delta} N + \varepsilon_{12}^{kl} && (6.4) \\ &= H_{\mathrm{L}} b_{\mathrm{L}} + A_{\mathrm{cs}} \mathit{\Delta} N + \varepsilon_{12}^{kl}, && (6.5)\end{aligned}$$

where A_{cs} is the mapping matrix to map the number of cycle slips to the phase measurement. Note that Eq. 6.5 only considers carrier phase measurement of visible satellites of the epoch in question. The last line is obtained by performing a ECEF to ENU frame transformation with the help of R_{L} given by Eq. 3.12:

$$\begin{aligned}\tilde{H}_{\mathrm{geo}} b_{12} &= \tilde{H}_{\mathrm{geo}} R_{\mathrm{L}}^{-1} R_{\mathrm{L}} b_{12} \\ &= H_{\mathrm{L}} b_{\mathrm{L}}. && (6.6)\end{aligned}$$

Working in the local ENU frame has its advantage: the Up-component can be constrained to zero in the case where movement is limited to a flat surface, which is often an acceptable assumption with on-road automotive. By doing so, we need fewer satellites to determine the baseline which is especially critical in urban environment. In the following, we will reason in the local ENU frame and eventually reduce the problem to *two dimensions* by assuming nullity for the Up-component ($b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}$).

6.2 Cycle slip correction with MAP

Blewitt proposed in [21] an approach using triple difference phase measurements to correct for cycle slips of a stationary receivers. The method becomes very soon insufficient when the receivers are moving at high dynamics. A quick turn of a vehicle can lead to a change in the double differences of $\pm\lambda/2$ or even of $\pm\lambda$, which might be misinterpreted as a cycle slip [8].

Lipp and Gu proposed in [22] a cycle slip detection and correction method using acceleration and angular rate measurements provided by inertial sensors. However, when inertial sensors are not available, reliable cycle slip correction can still be implemented using more sophisticated estimators.

6.2.1 MAP estimator with a priori baseline and ambiguity information

Instead of correcting for cycle slips satellite by satellite, we detect cycle slip by processing all visible satellites jointly and by analyzing the error norm derived with a Maximum A Posteriori Probability (MAP) estimator. Furthermore, since we are in a GNSS compass model configuration, we can also combine the length residual together with the MAP-based error norm to correct for cycle slips.

We intend to solve for the baseline vector in local ENU frame b_L in Eq. 6.5. The solution requires however phase measurements corrected with a known number of cycle slips $\varDelta N$. Subsequently, we can use a least-squares estimator to solve simultaneously $\varDelta N$ and b_L:

$$\min_{\substack{b_\mathrm{L}\in\mathbb{R}^{2\times 1}\\ \varDelta N\in\mathbb{Z}^{K-1}}} \|\varphi_{\mathrm{dd,fixed}} - (H_\mathrm{L} b_\mathrm{L} + A_\mathrm{cs}\varDelta N)\|^2_{\varSigma^{-1}_{\varphi_{\mathrm{dd,fixed}}}}. \tag{6.7}$$

[23] used a MAP estimator with an *a priori* knowledge on the baseline length to resolve for ambiguities. Section 5.2.2 also demonstrated the benefit of using constraint derived from an *a priori* knowledge on the baseline while performing an integer search (see Fig. 5.2). Embracing the same concept as Eq. 5.32, we can try to *resolve for $\varDelta N$ with a MAP estimator which exploits a priori information on the probability distribution of the baseline and the number of cycle slips*:

$$\begin{pmatrix} \hat{b}_\mathrm{L} \\ \varDelta\check{N} \end{pmatrix} = \arg \max_{\substack{b_\mathrm{L}\in\mathbb{R}^{2\times 1}\\ \varDelta N\in\mathbb{Z}^{K-1}}} P(b_\mathrm{L}, \varDelta N|\varphi_{\mathrm{dd,fixed}}). \tag{6.8}$$

With the same reasoning (Eq. 5.32 to 5.35), Bayes' rule infers that the MAP estimator can be expressed as a product of a likelihood term and a prior term. Together with the assumption that measurement noise is Gaussian distributed and that the *a priori* information is independent from the measurement and also follows a Gaussian distribution, we can express the MAP estimator as a sum of two least-squares estimators:

$$\begin{aligned}\begin{pmatrix} \hat{b}_\mathrm{L} \\ \varDelta\check{N} \end{pmatrix} &= \arg \max_{\substack{b_\mathrm{L}\in\mathbb{R}^{2\times 1}\\ \varDelta N\in\mathbb{Z}^{K-1}}} \frac{P(\varphi_{\mathrm{dd,fixed}}|b_\mathrm{L}, \varDelta N)P(b_\mathrm{L}, \varDelta N)}{P(\varphi_{\mathrm{dd,fixed}})} \\ &= \arg \max_{\substack{b_\mathrm{L}\in\mathbb{R}^{2\times 1}\\ \varDelta N\in\mathbb{Z}^{K-1}}} P(\varphi_{\mathrm{dd,fixed}}|b_\mathrm{L}, \varDelta N)P(b_\mathrm{L}, \varDelta N)\end{aligned}$$

$$= \arg \min_{\substack{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1} \\ \Delta N \in \mathbb{Z}^{K-1}}} \|\varphi_{\mathrm{dd,fixed}} - H_{\mathrm{L}} b_{\mathrm{L}} - A_{\mathrm{cs}} \Delta N\|^2_{\Sigma^{-1}_{\varphi_{\mathrm{dd,fixed}}}}$$

$$+ \left\| \begin{pmatrix} b_{\mathrm{L}} \\ \Delta N \end{pmatrix} - \begin{pmatrix} b_{\mathrm{L,ap}} \\ \Delta N_{\mathrm{ap}} \end{pmatrix} \right\|^2_{\Sigma^{-1}_{b_{\mathrm{L}},\Delta N}} . \tag{6.9}$$

The baseline *a priori* information can be derived using either an interpolation of the previously determined headings or a smoothed code based velocity. However, we do not have any *a priori* knowledge on the number of cycle slip. Furthermore, Eq. 6.9 involves an integer least-squares estimator as ΔN is an integer, which can lead to a complicated discrete search. Data analysis shows that the number of cycle slips are normally small integers. Therefore, the search can be limited to $\{-2, -1, 0, +1, +2\}$ from the predicted number of cycle slips which can be obtained by differencing the measured phase double difference and a polynomial fitted phase double difference. The next section proposes a cascaded cycle slip correction (cascaded CSC) which consists of first a *dynamic-based CSC* using a predicted number of cycle slips and a MAP estimator with only a priori information on the probability distribution of the baseline. Subsequently, with the estimated baseline, the cascaded CSC corrects for remaining cycle slips of the low-weighted measurements with an *a posteriori CSC* using the remaining phase residuals.

6.2.2 Dynamic-based cycle slip correction

Henkel and Kiam proposed a method to fully benefit from the MAP estimator by using a “pre-corrected" phase double difference and an *a priori* baseline information [20] when it is available:

$$\hat{b}_{\mathrm{L}} = \arg \max_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} P(b_{\mathrm{L}} | \tilde{\varphi}_{\mathrm{dd,fixed}}(\Delta N_{\mathrm{cand}})), \tag{6.10}$$

where $\tilde{\varphi}_{\mathrm{dd,fixed}}(\Delta N_{\mathrm{cand}})$ is the fixed phase double difference measurement corrected with a candidate of number of cycle slip ΔN_{cand} from a defined search space:

$$\tilde{\varphi}_{\mathrm{dd,fixed}}(\Delta N_{\mathrm{cand}}) = \varphi_{\mathrm{dd,fixed}} - A_{\mathrm{cs}} \Delta N_{\mathrm{cand}}. \tag{6.11}$$

How ΔN_{cand} is determined will be described thoroughly later. The general idea is to first calculate the difference between the measured phase double difference and the polynomial fitted phase double difference to determine the possible number of cycle slips. Other candidates are obtained in the neighborhood of $\{-2, -1, 0, +1, +2\}$ around the predicted number of cycle slips.

With Bayes’ rule and Gaussian assumption, we rewrite the MAP estimator as follows:

$$\begin{aligned} \hat{b}_{\mathrm{L}}(\Delta N_{\mathrm{cand}}) =& \arg \max_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} \frac{P(\tilde{\varphi}_{\mathrm{dd,fixed}}(\Delta N_{\mathrm{cand}}) | b_{\mathrm{L}}) P(b_{\mathrm{L}})}{P(\tilde{\varphi}_{\mathrm{dd,fixed}})} \\ =& \arg \max_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} P(\tilde{\varphi}_{\mathrm{dd,fixed}}(\Delta N_{\mathrm{cand}}) | b_{\mathrm{L}}) P(b_{\mathrm{L}}) \\ =& \arg \min_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} \|\tilde{\varphi}_{\mathrm{dd,fixed}}(\Delta N_{\mathrm{cand}}) - H_{\mathrm{L}} b_{\mathrm{L}}\|^2_{\Sigma^{-1}_{\tilde{\varphi}_{\mathrm{dd,fixed}}}} \\ & + \|b_{\mathrm{L,ap}} - b_{\mathrm{L}}\|^2_{\Sigma^{-1}_{b_{\mathrm{L}}}} . \end{aligned} \tag{6.12}$$

We further develop the above equation by assuming that the *a priori* baseline information and the phase measurements are uncorrelated. We obtain then:

$$\begin{aligned}\hat{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) =& \arg \min_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} (\tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) - H_{\mathrm{L}} b_{\mathrm{L}})^{\mathrm{T}} \mathit{\Sigma}^{-1}_{\tilde{\varphi}_{\mathrm{dd,fixed}}} (\tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) - H_{\mathrm{L}} b_{\mathrm{L}}) \\ &+ (b_{\mathrm{L,ap}} - b_{\mathrm{L}})^{\mathrm{T}} \mathit{\Sigma}^{-1}_{b_{\mathrm{L}}} (b_{\mathrm{L,ap}} - b_{\mathrm{L}})\end{aligned} \tag{6.13}$$

and by introducing the following notations

$$z_{\mathrm{MAP}} = \begin{pmatrix} \tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) \\ b_{\mathrm{L,ap}} \end{pmatrix} \quad \text{and} \quad H_{\mathrm{L,MAP}} = \begin{pmatrix} H_{\mathrm{L}} \\ I_2 \end{pmatrix}, \tag{6.14}$$

we can rewrite the minimization as

$$\begin{aligned}\hat{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) =& \arg \min_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} \begin{pmatrix} \tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) - H_{\mathrm{L}} b_{\mathrm{L}} \\ b_{\mathrm{L,ap}} - b_{\mathrm{L}} \end{pmatrix}^{\mathrm{T}} \underbrace{\begin{pmatrix} \mathit{\Sigma}^{-1}_{\tilde{\varphi}_{\mathrm{dd,fixed}}} & 0 \\ 0 & \mathit{\Sigma}^{-1}_{b_{\mathrm{L}}} \end{pmatrix}}_{\mathit{\Sigma}^{-1}_{z_{\mathrm{MAP}}}} \\ & \begin{pmatrix} \tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) - H_{\mathrm{L}} b_{\mathrm{L}} \\ b_{\mathrm{L,ap}} - b_{\mathrm{L}} \end{pmatrix} \\ =& \arg \min_{b_{\mathrm{L}} \in \mathbb{R}^{2\times 1}} \| z_{\mathrm{MAP}} - H_{\mathrm{L,MAP}} b_{\mathrm{L}} \|_{\mathit{\Sigma}^{-1}_{z_{\mathrm{MAP}}}} .\end{aligned} \tag{6.15}$$

With the found baselines and their associated number of cycle slip candidates, we select the best pair $\{\hat{b}_{\mathrm{L}}(\mathit{\Delta} N), \mathit{\Delta} \check{N}\}$ by using error norms derived from a MAP estimator 6.12:

$$\begin{aligned}\begin{pmatrix} \hat{b}_{\mathrm{L}}(\mathit{\Delta} N) \\ \mathit{\Delta} \check{N} \end{pmatrix} =& \arg \min_{\begin{pmatrix} \check{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) \\ \mathit{\Delta} N_{\mathrm{cand}} \end{pmatrix}} \left\| \tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) - H_{\mathrm{L}} \hat{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) \right\|^2_{\mathit{\Sigma}^{-1}_{\tilde{\varphi}_{\mathrm{dd,fixed}}}} \\ &+ \left\| b_{\mathrm{L,ap}} - \hat{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) \right\|^2_{\mathit{\Sigma}^{-1}_{b_{\mathrm{L}}}} .\end{aligned} \tag{6.16}$$

Nevertheless, we have preferred to use only the *a priori* baseline length information as the orientation of the *a priori* baseline is not reliable in our case. Hence, we choose the best candidate by minimizing the following:

$$\begin{aligned}\begin{pmatrix} \hat{b}_{\mathrm{L}}(\mathit{\Delta} N) \\ \mathit{\Delta} \check{N} \end{pmatrix} =& \arg \min_{\begin{pmatrix} \check{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) \\ \mathit{\Delta} N_{\mathrm{cand}} \end{pmatrix}} \left\| \tilde{\varphi}_{\mathrm{dd,fixed}}(\mathit{\Delta} N_{\mathrm{cand}}) - H_{\mathrm{L}} \hat{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}}) \right\|^2_{\mathit{\Sigma}^{-1}_{\tilde{\varphi}_{\mathrm{dd,fixed}}}} \\ &+ \frac{(\|\hat{b}_{\mathrm{L}}(\mathit{\Delta} N_{\mathrm{cand}})\| - l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}}.\end{aligned} \tag{6.17}$$

MAP estimator is extremely useful for determining cycle slips when the *a priori* baseline information is available so that the measurement is checked with an external reference. However, this is not always the case especially when *only* GNSS-based measurements are available.

We developed a *dynamic-based* CSC which adapts the determination of *a priori* baseline information according to the dynamics of the receivers so that Eq. 6.17 can be used for detecting and correcting cycle slips. Three main modes of correction were developed according to the dynamics of the receiver:

- *High dynamics* in the sense that receiver speed is high enough;
- *Low dynamics* in the sense that previous headings are almost constant;
- All other cases which are not qualified for the two dynamics above are classified as *medium dynamics*.

The following sections explain step by step how dynamic-based CSC is performed according to the dynamics of the vehicle.

6.2.2.1 Cycle slip correction at high dynamics

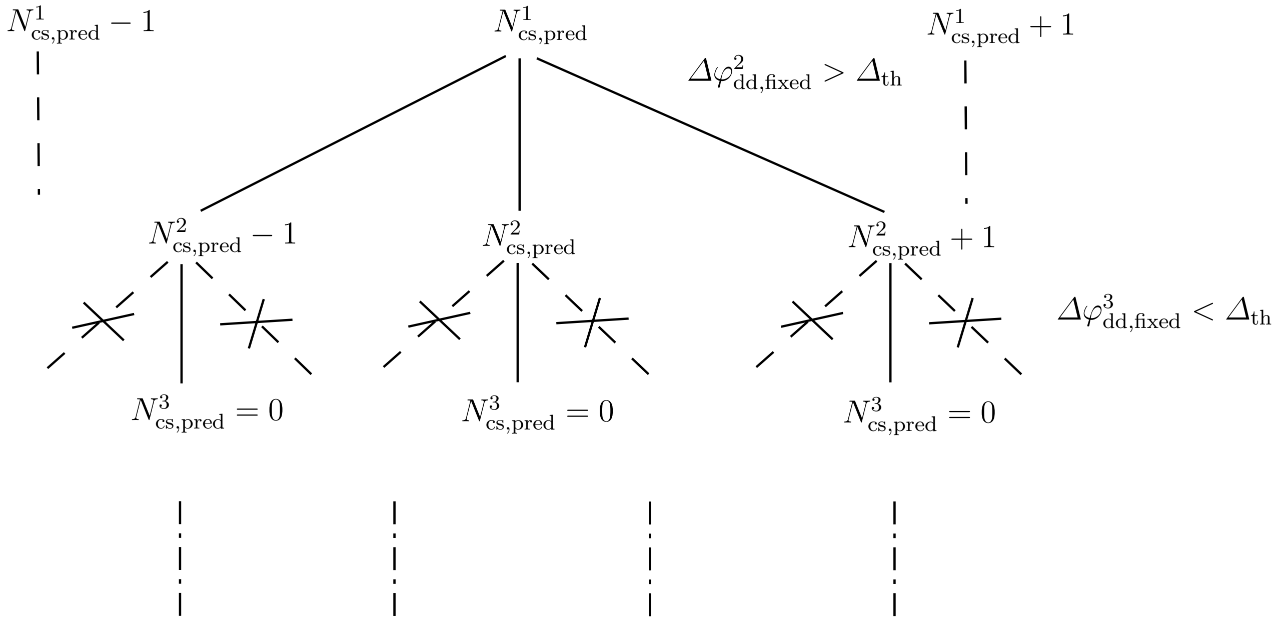

$N^1_{\text{cs,pred}} - 1$
$N^1_{\text{cs,pred}}$
$N^1_{\text{cs,pred}} + 1$
$\Delta\varphi^2_{\text{dd,fixed}} > \Delta_{\text{th}}$
$N^2_{\text{cs,pred}} - 1$
$N^2_{\text{cs,pred}}$
$N^2_{\text{cs,pred}} + 1$
$\Delta\varphi^3_{\text{dd,fixed}} < \Delta_{\text{th}}$
$N^3_{\text{cs,pred}} = 0$
$N^3_{\text{cs,pred}} = 0$
$N^3_{\text{cs,pred}} = 0$

Figure 6.3: Combination of all possible cycle slips: A branch which survives until the end layer are considered as a candidate of $\Delta N_{\text{cs,cand}}$.

Line 2 predicts the heading of the current epoch by extrapolating linearly.

Line 4 determines an a priori baseline information. When the vehicle is moving, the baseline fixed on the vehicle is parallel to the velocity. Therefore, the smoothed-code based velocity vector v can be exploited to find the *a priori* baseline in local ENU coordinate frame:

$$b_{\text{L,ap}} = \frac{v^{E,N,U}}{\|v^{E,N,U}\|} l_{\text{ap}}, \tag{6.18}$$

However, this approach to set the baseline is only legitimate when the speed is high enough due to the norm in the denominator. Note that in the algorithm, we limit ourselves to only the East and the North components since heading determination is the main goal in this work.

Algorithm 4 Cycle slip correction at high dynamics

```
Input: $\phi, l_{\mathrm{ap}}, \bar{v}^{\mathrm{E,N}}, \varSigma_{\bar{x}}^{-1}, R_{\mathrm{L}}, \varphi^{\mathrm{T}}_{\mathrm{dd,fixed}}, H_{\mathrm{L}}, \sigma_{l_{\mathrm{ap}}}$
Output: $\varDelta N_{\mathrm{acc}}, \varDelta N_{\mathrm{cs,sel}}$

$(\alpha_0, \alpha_1) = \arg\min_{\alpha_0,\alpha_1} \quad \sum_{n=-5}^{-1} \|\phi(t-nT) - \sum_{p=0}^{1} \alpha_p \cdot (t-nT)^p\|^2$
$\phi_{\mathrm{pred}} = \alpha_0 + \alpha_1(5T)$    ▷ extrapolate to predict current heading
while $t$ do
    $b_{\mathrm{L,ap}} = \frac{v^{\mathrm{E,N}}}{\|v^{\mathrm{E,N}}\|} l_{\mathrm{ap}}$    ▷ a priori baseline vector
    $\varSigma_{b_{\mathrm{L,ap}}} = R_{\mathrm{L}} \varSigma_{\bar{x}}^{-1} R_{\mathrm{L}}^{\mathrm{T}} \frac{l_{\mathrm{ap}}^2}{\|\bar{v}^{\mathrm{E,N}}\|^2}$    ▷ a priori baseline covariance matrix
    $z_{\mathrm{MAP}} = \begin{pmatrix} \varphi_{\mathrm{dd,fixed}} \\ b_{\mathrm{L,ap}} \end{pmatrix};$
    $\varSigma_{z_{\mathrm{MAP}}}^{-1} = \begin{pmatrix} \varSigma_{\varphi_{\mathrm{dd,fixed}}}^{-1} & 0 \\ 0 & \varSigma_{b_{\mathrm{L,ap}}}^{-1} \end{pmatrix}$    ▷ set up variables for MAP estimation
    $H_{\mathrm{L,MAP}} = \begin{pmatrix} H_{\mathrm{L}} \\ I_2 \end{pmatrix}$
    $A_{\mathrm{MAP}} = \begin{pmatrix} \frac{\lambda}{2} I_{K-1} \\ 0_{2\times(K-1)} \end{pmatrix}$
    $P_{H_{\mathrm{L,MAP}}^{\mathrm{T}}} = H_{\mathrm{L,MAP}} (H_{\mathrm{L,MAP}}^{\mathrm{T}} \varSigma_{z_{\mathrm{MAP}}}^{-1} H_{\mathrm{L,MAP}})^{-1} H_{\mathrm{L,MAP}}^{\mathrm{T}} \varSigma_{z_{\mathrm{MAP}}}^{-1}$    ▷ projection matrix
    $P_{H_{\mathrm{L,MAP}}^{\mathrm{T}}}^{\perp} = I_{K+2} - P_{H_{\mathrm{L,MAP}}^{\mathrm{T}}};$
    $b_{\mathrm{L,MAP}} = (H_{\mathrm{L,MAP}}^{\mathrm{T}} \varSigma_{z_{\mathrm{MAP}}}^{-1} H_{\mathrm{L,MAP}})^{-1} H_{\mathrm{L,MAP}}^{\mathrm{T}} \varSigma_{z_{\mathrm{MAP}}}^{-1} z_{\mathrm{MAP}}$
    $\phi_{\mathrm{MAP}} = \arctan(\frac{b_{\mathrm{MAP}}^{\mathrm{E}}}{b_{\mathrm{MAP}}^{\mathrm{N}}})$    ▷ predicted heading in degree without CSC
    if $\phi_{\mathrm{pred}} - \phi_{\mathrm{MAP}} > 2$ then
        $r_{\mathrm{MAP}} = P_{H_{\mathrm{L,MAP}}^{\mathrm{T}}}^{\perp} z_{\mathrm{MAP}}$    ▷ MAP estimation phase residuals before CSC
        $\mathrm{WSSE}_{\mathrm{MAP}} = r_{\mathrm{MAP}}^{\mathrm{T}} \varSigma_{z_{\mathrm{MAP}}}^{-1} r_{\mathrm{MAP}} + \frac{(\|b_{\mathrm{L,MAP}}\| - l_{\mathrm{ap}})^2}{\sigma_{l_{\mathrm{ap}}}^2}$    ▷ WSSE of phase residuals before CSC
        $\mathrm{WSSE}_{\mathrm{MAP,th}} = 2^2((K-1)+2)+2;$
        if $\mathrm{WSSE}_{\mathrm{MAP}} > \mathrm{WSSE}_{\mathrm{MAP,th}}$ then
            $\varDelta N_{\mathrm{pred}} = \left[\frac{\varphi_{\mathrm{dd,fixed}} - H_{\mathrm{L}} b_{\mathrm{L,ap}}}{\lambda/2}\right]$    ▷ predict number of cycle slips jointly for all satellites
            for $k = 2 \to K$ do    ▷ we assume satellite 1 is the reference satellite
                $(\beta_0, \beta_1) = \arg\min_{\beta_0,\beta_1} \quad \sum_{n=-5}^{-1} \|\varphi_{\mathrm{dd,fixed}}^{k}(t-nT) - \sum_{p=0}^{1} \beta_p \cdot (t-nT)^p\|^2$
                $\Delta\varphi^k_{\mathrm{dd,fixed}} = |\varphi^k_{\mathrm{dd,fixed}}(t) - (\beta_0 + \beta_1 t)|$
                **if** $\Delta\varphi^k_{\mathrm{dd,fixed}} > \Delta_{th}$ **then**
                    $\Delta N^k_{\mathrm{cand}} = \{\Delta N^{k1}_{\mathrm{pred}} - 1, \Delta N^{k1}_{\mathrm{pred}}, \Delta N^{k1}_{\mathrm{pred}} + 1\}$          ▷ a search in $\{-1, 0, +1\}$
                **else**
                    $\Delta N^k_{\mathrm{cs,cand}} = \{0\}$;
                **end if**
            **end for**
            Determine all combinations of $\Delta N^k_{\mathrm{cand}}$ and group in $\Delta N_{\mathrm{cs,cand}}$
            Determine number of candidates $n_{\mathrm{cand}}$
            **for** $c = 1 \to n_{\mathrm{cand}}$ **do**
                $r_{\mathrm{MAP,cand}}(c) = P^{\perp}_{H^{\mathrm{T}}_{\mathrm{L,MAP}}}(z_{\mathrm{MAP}} - \frac{\lambda}{2}\Delta N_{\mathrm{cs,cand}}(c))$
                $b_{\mathrm{L,cand}}(c) = (H^{\mathrm{T}}_{\mathrm{L,MAP}}\Sigma^{-1}_{z_{\mathrm{MAP}}}H_{\mathrm{L,MAP}})^{-1}H^{\mathrm{T}}_{\mathrm{L,MAP}}\Sigma^{-1}_{z_{\mathrm{MAP}}}(z_{\mathrm{MAP}} - \frac{\lambda}{2}\Delta N_{\mathrm{cs,cand}}(c))$;
                $\mathrm{WSSE}_{\mathrm{MAP,cand}}(c) = r_{\mathrm{MAP,cand}}(c)^{\mathrm{T}}\Sigma^{-1}_{z_{\mathrm{MAP}}}r_{\mathrm{MAP,cand}}(c) + \frac{(\|b_{\mathrm{L,cand}}(c)\| - l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}}$
            **end for**
        **end if**
        **if** $\min(\mathrm{WSSE}_{\mathrm{MAP,cand}}) > \mathrm{WSSE}_{\mathrm{MAP,th}}$ OR $N_{\mathrm{cand}} = 1$ **then**
            **for** $k = 1 \to K$ **do**                                                  ▷ extended search
                $(\beta_0, \beta_1) = \arg\min_{\beta_0,\beta_1} \quad \sum_{n=-5}^{-1} \|\varphi^k_{\mathrm{dd,fixed}}(t - nT) - \sum_{p=0}^{1} \beta_p \cdot (t - nT)^p\|^2$
                $\Delta\varphi^k_{\mathrm{dd,fixed}} = \varphi^k_{\mathrm{dd,fixed}}(t) - (\alpha_0 + \alpha_1 t)$
                **if** $\Delta\varphi^k_{\mathrm{dd,fixed}} > \Delta_{th}$ **then**                         ▷ exclude trustable satellites
                    $\Delta N^k_{\mathrm{cand}} = \{\Delta N^{k1}_{\mathrm{pred}} - 2, \Delta N^{k1}_{\mathrm{pred}} - 1, \Delta N^{k1}_{\mathrm{pred}}, \Delta N^{k1}_{\mathrm{pred}} + 1, \Delta N^{k1}_{\mathrm{pred}} + 2\}$ ▷ search
                **else**
                    $\Delta N^k_{\mathrm{cs,cand}} = 0$;
                **end if**
            **end for**
            Determine all combinations of $\Delta N^k_{\mathrm{cand}}$ and group in $\Delta N_{\mathrm{cs,cand}}$
            Determine number of candidates $n_{\mathrm{cand}}$
            **for** $c = 1 \to n_{\mathrm{cand}}$ **do**
                $r_{\mathrm{MAP,cand}}(c) = P^{\perp}_{H^{\mathrm{T}}_{\mathrm{L,MAP}}}(z_{\mathrm{MAP}} - \frac{\lambda}{2}\Delta N_{\mathrm{cs,cand}}(c))$
                $b_{\mathrm{L,cand}}(c) = (H^{\mathrm{T}}_{\mathrm{L,MAP}}\Sigma^{-1}_{z_{\mathrm{MAP}}}H_{\mathrm{L,MAP}})^{-1}H^{\mathrm{T}}_{\mathrm{L,MAP}}\Sigma^{-1}_{z_{\mathrm{MAP}}}(z_{\mathrm{MAP}} - \frac{\lambda}{2}\Delta N_{\mathrm{cs,cand}}(c))$
                $\mathrm{WSSE}_{\mathrm{MAP,cand}}(c) = r_{\mathrm{MAP,cand}}(c)^{\mathrm{T}}\Sigma^{-1}_{z_{\mathrm{MAP}}}r_{\mathrm{MAP,cand}}(c) + \frac{(\|b_{\mathrm{L,cand}}(c)\| - l_{\mathrm{ap}})^2}{\sigma^2_{l_{\mathrm{ap}}}}$
            **end for**
        **end if**
        Select cycle slip candidate $\Delta N_{\mathrm{cs,sel}}$ which minimizes the WSSE
        Determine index of the best candidate $\mathrm{ind\_sel}$ among all candidates
        **if** $\mathrm{WSSE}_{\mathrm{MAP,cand}}(\mathrm{ind\_sel}) < \mathrm{WSSE}_{\mathrm{MAP}} < \mathrm{WSSE}_{\mathrm{MAP,th}}$ **then**
            $\varphi_{\mathrm{dd,fixed}} = \varphi_{\mathrm{dd,fixed}} - \frac{\lambda}{2}\Delta N_{\mathrm{cs,sel}}$
            $\Delta N_{\mathrm{acc}} := \Delta N_{\mathrm{acc}} + \Delta N_{\mathrm{cs,sel}}$                 ▷ Increment accumulated cycle slips
        **else**
            Try cycle slip correction at medium dynamics
        **end if**
    **end if**
**end while**
```

Line 6 to 9 prepare variables needed for MAP estimation described in section 6.2.2.

Line 10 to 11 determine the projection matrix $P_{H^{\mathrm{T}}_{\mathrm{L,MAP}}}$ as well as the orthogonal projection matrix $P^{\perp}_{H^{\mathrm{T}}_{\mathrm{L,MAP}}}$ of the least-squares problem in Eq. 6.17.

Line 12 is determined here by assuming that there is no cycle slip. The calculated baseline vector is then used to determine the MAP heading ϕ_{MAP}. If the MAP heading is different from the extrapolated heading ϕ_{pred} determined in *Line 2* by 2°, a search for cycle slips is thus necessary. However, if the MAP heading is comparable to the extrapolated heading, no search for cycle slips is required. By introducing this condition, we can reduce the probability of false alarm, i.e. correcting for cycle slip while in reality, no cycle slip occurs.

Line 18 checks again if a search for cycle slip is really necessary. The search will only be performed if the MAP error norm $\mathrm{WSSE}_{\mathrm{MAP}}$ (see Eq. 6.17) is greater than a predefined threshold $\mathrm{WSSE}_{\mathrm{MAP,th}}$.

Line 19 is performed to predict jointly for all satellites the number of cycle slips by using the phase residuals after fixing the baseline to the a priori baseline. By refering to Eq. 6.5, we know that if the residuals are too big, the phase double difference measurements contain probably cycle slips. The number of probable cycle slips is calculated by dividing the residuals with $\lambda/2$.

Line 23 is again to avoid false alarm in the correction. It is performed individually for each satellite. If the difference between the extrapolated phase double difference and the measured phase double difference is small, no cycle slip search is required.

Line 24 to 30 and *42 to 48* search for candidates of number of cycle slips. The first search is limited to a smaller neighborhood around the predicted number of cycle slip $\varDelta N^{k1}_{\mathrm{pred}}$ and subsequently extended to a bigger neighborhood if the first search fails to provide a good cycle slip candidate. The search is performed as illustrated by Fig. 6.3.

Take the first search for example, each node at each layer of the search produces three new branches. All branches will survive if the condition at *Line 23* is fulfilled at the next layer. Or else, each node only produces one branch which is linked to a node that represents zero cycle slip.

Line 49 to 53 select the best candidate by using the sum of error norms in Eq. 6.17 derived with a MAP estimator. If no candidate at the first search produces $WSSE_{\mathrm{MAP,cand}}$ which is smaller than a desired threshold, an extended search will subsequently be performed. If the extended search fails again to find the right candidate, we will then use another cycle slip correction mode.

This approach however only works for movements like in our case where the direction of displacement is aligned with the baseline. In maritime cases, the orientation of the baseline is often affected by the water flow, which renders Eq. 6.18 invalid [8].

6.2.2.2 Cycle slip correction at medium dynamics

Although smoothed with carrier phase measurement, code measurement remains noisy to a certain extend that the error in the velocity calculation is comparable to the true velocity when the dynamics are low. Therefore, when the dynamics are not high enough, velocity is no longer usable to derive the *a priori* baseline vector.

Since we have relatively low dynamics, the yaw and pitch of the vehicle do not vary much: a linear fitting of the previous yaw and pitch angles can hence be used to predict the angles of the current epoch. Subsequently, with the predicted attitude of the current epoch, the baseline vector can be calculated with trivial trigonometric operations as below:

Algorithm 5 Cycle slip correction at medium dynamics

```
Input: $\phi, l_{\mathrm{ap}}, R_{\mathrm{L}}, \varphi^{\mathrm{T}}_{\mathrm{dd,fixed}}, H_{\mathrm{L}}, \sigma_{l_{\mathrm{ap}}}$
Output: $\mathit{\Delta} N_{\mathrm{acc}}, \mathit{\Delta} N_{\mathrm{cs,sel}}$
 1: while $t$ do
 2:     $\begin{pmatrix} \alpha_0 & \alpha_1 \end{pmatrix} = \arg\min_{\alpha_0,\alpha_1} \quad \sum_{n=-5}^{-1} \|\phi(t-nT) - \sum_{p=0}^{1} \alpha_p \cdot (t-nT)^p\|^2$
 3:     $\phi_{\mathrm{pred}} = \alpha_0 + \alpha_1(5T)$        ▷ extrapolate to predict current heading
 4:     $b_{\mathrm{L,ap}} = l_{\mathrm{ap}} \begin{pmatrix} \sin(\phi_{\mathrm{pred}}) \\ \cos(\phi_{\mathrm{pred}}) \end{pmatrix}$        ▷ determine a priori baseline
 5:     if $\phi(t-1) - \phi(t-5) > 5$ then
 6:         $\mathit{\Sigma}^{-1}_{b_{\mathrm{L,ap}}} = (l^2_{\mathrm{ap}}(\phi(t-1) - \phi(t-5))^2, I_2)^{-1};$
 7:     else
 8:         $\mathit{\Sigma}^{-1}_{b_{\mathrm{L,ap}}} = (l^2_{\mathrm{ap}}(\frac{5\pi}{180} I_2)^{-1};$
 9:     end if        ▷ determine baseline covariance matrix
10:     perform line 6 to 11 of algorithm 4
11:     $b_{\mathrm{MAP}} = (H^{\mathrm{T}}_{\mathrm{L,MAP}} \mathit{\Sigma}^{-1}_{z_{\mathrm{MAP}}} H_{\mathrm{L,MAP}})^{-1} H^{\mathrm{T}}_{\mathrm{L,MAP}} \mathit{\Sigma}^{-1}_{z_{\mathrm{MAP}}} z_{\mathrm{MAP}}$
12:     $\phi_{\mathrm{MAP}} = \arctan(\frac{b^E_{\mathrm{MAP}}}{b^N_{\mathrm{MAP}}})$
13:     if $\phi_{\mathrm{pred}} - \phi_{\mathrm{MAP}} > 2$ then        ▷ determine WSSE of MAP estimation
14:         perform line 15 to 17 of algorithm 4
15:     end if
16:     if $\mathrm{WSSE}_{\mathrm{MAP}} > \mathrm{WSSE}_{\mathrm{MAP,th}}$ then
17:         $\mathit{\Delta} N_{12,\mathrm{cs}} = \left[ \frac{\varphi_{\mathrm{dd,fixed}} - H_{\mathrm{L}} b_{\mathrm{L,ap}}}{\lambda/2} \right]$        ▷ predict number of cycle slips jointly for all satellites
18:         perform line 37 to 54 to search for cycle slip candidates and to select subsequently the best candidate
19:         if $\mathrm{WSSE}_{\mathrm{MAP,cand}}(\mathrm{ind\_sel}) < \mathrm{WSSE}_{\mathrm{MAP}}$ AND $\mathrm{WSSE}_{\mathrm{MAP,cand}}(\mathrm{ind\_sel}) < \mathrm{WSSE}_{\mathrm{MAP,th}}$ then
20:             $\varphi_{\mathrm{dd,fixed}} = \varphi_{\mathrm{dd,fixed}} - \frac{\lambda}{2} \mathit{\Delta} N_{\mathrm{cs,sel}};$
21:         else
22:             cycle slip correction fails
23:         end if
24:     end if
25: end while
```

$$b_{\mathrm{L,ap}} = \begin{bmatrix} \sin(\phi_{\mathrm{pred}})\cos(\theta_{\mathrm{pred}}) \\ \cos(\phi_{\mathrm{pred}})\cos(\theta_{\mathrm{pred}}) \\ \sin(\theta_{\mathrm{pred}}) \end{bmatrix} \cdot l_{\mathrm{ap}}, \tag{6.19}$$

with ϕ_{pred} and θ_{pred} obtained by extrapolating from the previous five epochs for the current epoch.

Besides the baseline vector, the rest of the algorithm resembles algorithm 4.

6.2.2.3 Cycle slip correction at stationary

For the case of a stationary vehicle, carrier phase measurement remains stable and only a drift due to satellite movement is observable over a long period. Hence, as described in [17], a triple difference approach can be used to detect cycle slips.

$$\varphi_{\mathrm{td}}^{k1}(t) = \varphi_{\mathrm{dd}}^{k1}(t) - \varphi_{\mathrm{dd}}^{k1}(t-1) \tag{6.20}$$

denotes the triple difference operation using the double difference phase measurement of the current epoch and from the previous epoch.

The number of cycle slips to be corrected is the rounded quotient of the triple difference and $\left[\frac{(\varphi_{\mathrm{td}}^{k,1}}{\lambda/2})\right]$. This approach is simple but very efficient when it comes to a cycle slip correction for cases like in figure 6.1.

6.2.2.4 Cycle slip correction at low dynamics

When the vehicle is coasting at very low dynamics, the phase measurements do not vary much and therefore we can adopt a triple difference method as in the case of stationary vehicle:

$$\varphi_{\mathrm{td}}^{k,1} = \varphi_{\mathrm{dd}}^{k,1}(t) - \varphi_{\mathrm{dd}}^{k,1}(t-1). \tag{6.21}$$

However, an approach which depends entirely on triple difference is not reliable. As the system has to work in a real-time manner, we predict the dynamic level by looking at the previous epochs. If the previous epochs have phase measurements which are stable, it is then very likely that the current epoch does not vary much. Such reasoning does not work in all cases. The low dynamics here are defined as such by looking only at the vehicle previous headings. If the vehicle coasts in a line at a high speed, a turn can cause a big change in heading and in phase measurements instantaneously. A naive approach which depends only on the triple difference will treat the turn as a cycle slip and lead to a wrong correction, causing the change in motion to happen later. Therefore, when the previous epochs reveal that the vehicle have very low dynamics, we use the baseline of the previous epoch as an *a priori* baseline but in order to not take the low dynamics (only in heading sense) for granted, we also take the velocity derived from the smoothed code measurement into account (see *Line 15* to *Line 21*).

Unlike in algorithm 4 and 5, the region of integer search is determined first. Since this cycle slip correction method depends on the triple difference if all triple differences are small, no further correction is necessary. Nevertheless, unlike in the case of a stationary vehicle, a simple rounding is not appropriate since the phase measurements can be slightly less stable when there is some dynamics. Therefore, we need to define a "gray zone" when $0.25 < |\frac{\varphi_{\mathrm{td}}^{k}}{\lambda/2} - \lfloor\frac{\varphi_{\mathrm{td}}^{k}}{\lambda/2}\rfloor| < 0.75$ (see *Line 4* to *9*).

Algorithm 6 Cycle slip correction at low dynamics

```
Input: $\phi, l_{\mathrm{ap}}, R_{\mathrm{L}}, \varphi^{\mathrm{T}}_{\mathrm{dd,fixed}}, H_{\mathrm{L}}, \sigma_{l_{\mathrm{ap}}}, b_{\mathrm{L,fixed}}(1:2, t-1), \mathit{\Sigma}_{\bar{x}}(t-1))$
Output: $\mathit{\Delta} N_{\mathrm{acc}}, \mathit{\Delta} N_{\mathrm{cs,sel}}$
while $t$ do
    $\varphi_{\mathrm{td,fixed}}(t) = \varphi_{\mathrm{dd,fixed}}(t) - \varphi_{\mathrm{dd,fixed}}(t)$;  ▷ triple difference between two consecutive epochs
    for $k = 2 \to K$ do
        if $|\frac{\varphi^k_{\mathrm{td}}}{\lambda/2} - \lfloor\frac{\varphi^k_{\mathrm{td}}}{\lambda/2}\rfloor| > 0.75$ then
            $\mathit{\Delta} N^k_{\mathrm{cs},cand} = \lceil\frac{\varphi^k_{\mathrm{td}}}{\lambda/2}\rceil$   ▷ determine individually cycle slip candidates for each satellite
        else if $|\frac{\varphi^k_{\mathrm{td}}}{\lambda/2} - \lfloor\frac{\varphi^k_{\mathrm{td}}}{\lambda/2}\rfloor| < 0.25$ then
            $\mathit{\Delta} N^k_{\mathrm{cs},cand} = \lfloor\frac{\varphi^k_{\mathrm{td}}}{\lambda/2}\rfloor$
        else
            $\mathit{\Delta} N^k_{\mathrm{cs},cand} = \{\lfloor\frac{\varphi^k_{\mathrm{td}}}{\lambda/2}\rfloor, 0, \lceil\frac{\varphi^k_{\mathrm{td}}}{\lambda/2}\rceil\}$
        end if
    end for
    determine all $N_{\mathrm{cand}}$ combinations of candidates $\mathit{\Delta} N_{\mathrm{cs}}$
    if $N_{\mathrm{cand}} > 0$ then
        $e_{b,\mathrm{head}} = \frac{b_{\mathrm{L,fixed}}(1:2,t-1)}{\|b_{\mathrm{L,fixed}}(1:2,t-1)\|}$   ▷ predict directive vector by using previous fixed baseline
        if $\|\bar{v}_{\mathrm{L}}(1:2,t)\| > 10$ then
            $e_{b,\mathrm{vel}} = \frac{\bar{v}_{\mathrm{L}}(1:2,t)*T}{\|\bar{v}_{\mathrm{L}}(1:2,t)*T\|}$
            $e_{b,\mathrm{ap}} = e_{b,\mathrm{head}} + e_{b,\mathrm{vel}}$   ▷ predict directive vector by using previous fixed baseline and current velocity vector
            $b_{\mathrm{L,ap}} = \frac{e_{b,\mathrm{ap}}}{\|e_{b,\mathrm{ap}}\|} l_{\mathrm{ap}}$   ▷ determine a priori baseline
        else
            $b_{\mathrm{L,ap}} = \frac{e_{b,\mathrm{head}}}{\|e_{b,\mathrm{head}}\|} l_{\mathrm{ap}}$   ▷ determine a priori baseline
        end if
        if $\|\bar{v}_{\mathrm{L}}(1:2,t)\| > 10$ then   ▷ determine covariance matrix of $b_{\mathrm{ap}}$
            $\mathit{\Sigma}_{b_{\mathrm{ap}}} = \mathit{\Sigma}_{b_{\mathrm{L,fixed}}}(t-1) + R_{\mathrm{L}}(\mathit{\Sigma}_{\bar{x}}(t) + \mathit{\Sigma}_{\bar{x}}(t-1))R_{\mathrm{L}}^{-1}$
        else
            $\mathit{\Sigma}_{b_{\mathrm{ap}}} = \mathit{\Sigma}_{b_{\mathrm{L,fixed}}}(t-1)$
        end if
        perform line 6 to 11 of algorithm 4 to prepare variables for MAP;
        perform line 49 to 53 of algorithm 4 to determine the WSSE of each candidate
        select cycle slip candidate $\mathit{\Delta} N_{\mathrm{cs,sel}}$ which minimizes the WSSE
        determine index of the best candidate $\mathrm{ind\_sel}$ among all candidates
        $\varphi_{\mathrm{dd,fixed}} = \varphi_{\mathrm{dd,fixed}} - \frac{\lambda}{2}\mathit{\Delta} N_{\mathrm{cs,sel}}$;
    end if
end while
```

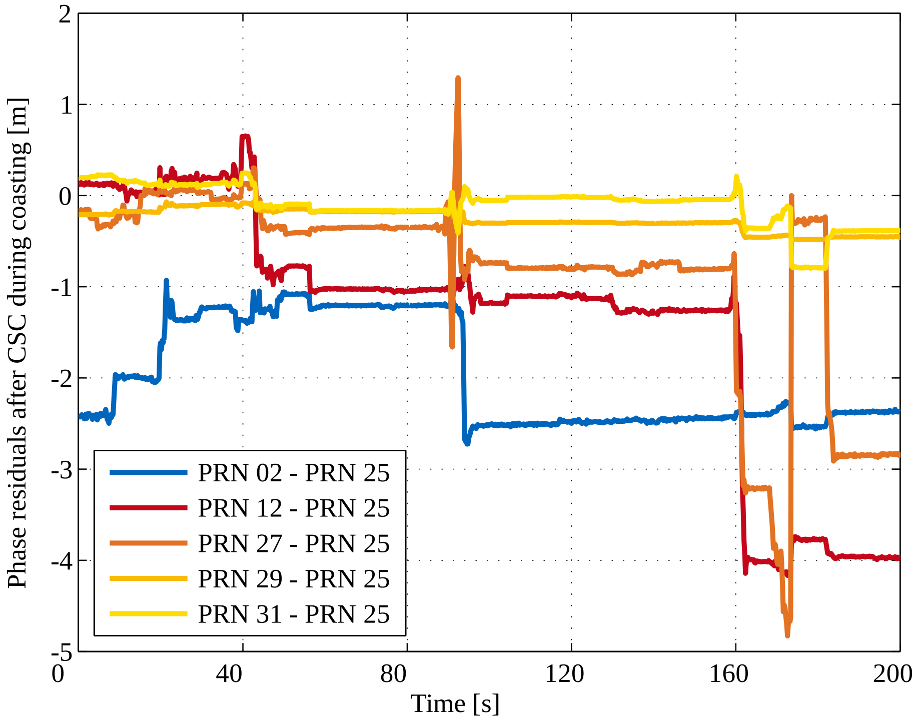


Figure 6.4: Test drive at at ESA-AZO: Corresponding phase residuals in the case where no cycle slip correction was used.

6.2.3 Validation of dynamic-based cascaded cycle slip correction

We carried out extensive tests on the algorithm. First, the ANAVS attitude determination system was performed without any cycle slip correction. Fig. 6.4 illustrates the phase residuals after fixing baseline. Without any cycle slip correction, the residuals ranges from 10 cm to 5 m, indicating that the least-squares estimation of the baseline does not fit well with the fixed phase DD measurements.

With the use of dynamic-based CSC, the residuals are substantially reduced as seen in Fig. 6.5. However, we still observe with some satellites (PRN 02 and PRN 29) phase residuals of meter level. These are indeed satellites of the lowest elevation angle among all visible satellites. PRN 02 has an elevation angle of 28° while PRN 27 has an elevation angle of 26°. Therefore, a selection based on the a weighted squared error (see Eq. 6.17) is penalizing for low elevated satellites as they have low weighting, which can be 100 times lower than the satellite with the highest elevation. Consequently, satellites with low weighting can either be neglected in the correction and be corrected wrongly. While a missed correction or a wrong correction does not affect much the instantaneous heading determination given the low weighting, the error continues to propagate and can even be accumulative since we often use information from the previous epochs to derive a priori information for the current epoch.

A missed or wrong correction of a low-weighted satellite can be corrected using the following equation:

$$\varphi_{\text{dd,fixed}} - A_{\text{cs}} \mathit{\Delta} N = H b_{\text{fixed}} + A_{\text{cs}} \mathit{\Delta} \tilde{N} + \varepsilon, \tag{6.22}$$

where b_{fixed} is the baseline calculated right after phase double difference is corrected with dynamic-based cycle slip correction, and $\tilde{N}$ is the remaining cycle slips.

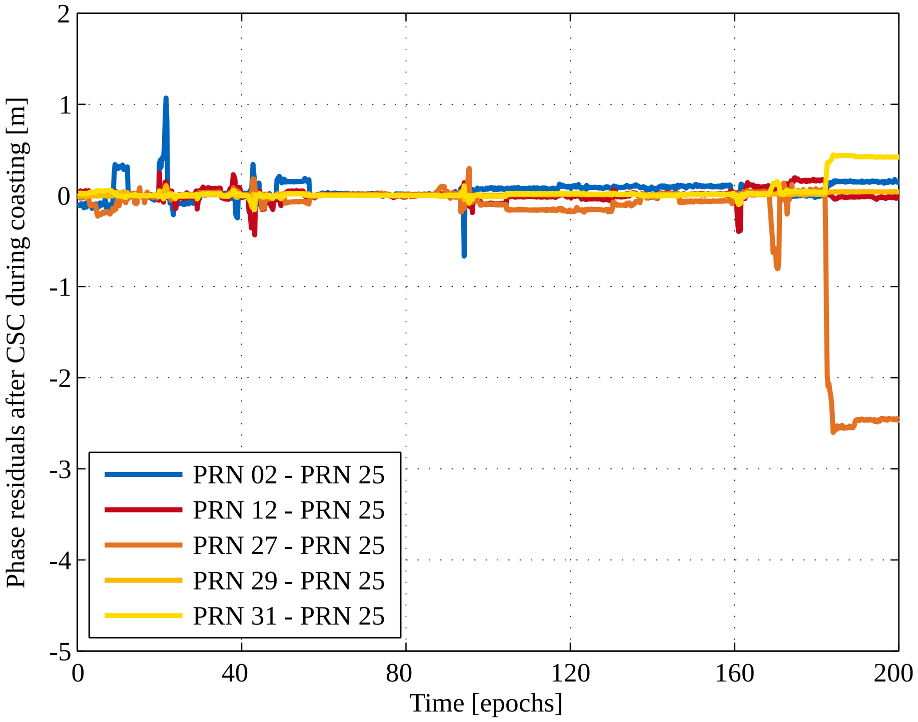


Figure 6.5: Test drive at at ESA-AZO: Corresponding phase residuals where dynamic-based cascaded cycle slip correction was in use.

$A_{\text{cs}}\Delta\tilde{N} + \varepsilon$ represents the residuals that we observe in Fig. 6.5. In the case where no cycle slip remains, the residuals must be small since they only represent the noise level in phase measurements. When the residuals become substantial, $\tilde{N}$ contains remaining cycle slips that have to be corrected.

The final corrected phase double difference is hence:

$$\tilde{\varphi}_{\text{dd,fixed}} = \varphi_{\text{dd,fixed}} - A_{\text{cs}}(\Delta N + \Delta\tilde{N}), \tag{6.23}$$

where ΔN is derived from the *dynamic-based CSC* and $\Delta\tilde{N}$ from the *a posteriori CSC*.

Cascaded cycle slip correction is more robust than only dynamic-based cycle slip correction as it does not only correct for satellites with high weighting with a WSSE derived with a MAP estimator but also performs a posterior correction for low-weighted satellites once the baseline is fixed to avoid propagative errors.

Fig. 6.6 shows the phase residuals of the same data set with the phase double differences corrected with cascaded CSC (first with the dynamic-based cycle slip correction and subsequently with a posterior correction). The residuals are below 5 cm, which shows that cycle slips are reliably corrected.

Fig. 6.8 shows heading obtained from sole dynamic-based CSC (blue curve) and cascaded CSC with a posteriori CSC in use after dynamic-based CSC (orange curve). Different correction modes of the dynamic-based cycle slip correction were use intermittently. Red markers indicate epochs where very low dynamics cycle slip correction mode was in use; green markers indicate epochs where medium dynamics cycle slip correction mode was in use; magenta markers indicate epochs where very high dynamics cycle slip correction mode was in use. The last section of the curves

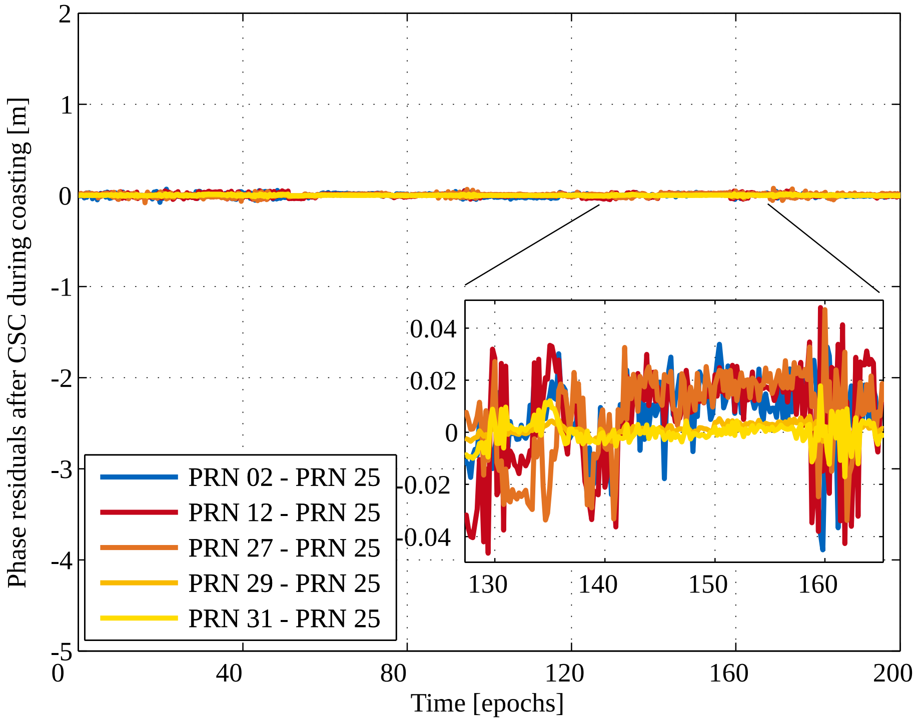


Figure 6.6: Test drive at at ESA-AZO: Corresponding phase residuals where dynamic-based cascaded cycle slip correction was coupled with a fixing posterior correction.

differ substantially; this difference is also easily observable from the residuals (see Fig. 6.5 and 6.6). Comparing the heading curve to the plotted route of the car (see Fig. 6.8), we notice that the heading estimated using only dynamic-based CSC is wrong while cascaded CSC is more reliable.

Another test drive was conducted in front of Volkswagen headquarter in Wolfsburg, Germany. Fig. 6.9 shows the track taken during the test and Fig. 6.10 shows the corresponding headings.

Fig. 6.11 illustrates the phase residuals of fixed solution during coasting obtained with only a dynamic-based cascaded cycle slip correction while Fig. 6.12 shows the phase residuals obtained using additionally a posterior cycle slip correction.

With the dynamic-based cycle slip correction, we observe that satellite PRN 32 is constantly bearing big residuals. It is indeed low-elevated. The substantial residual of this satellite persists or even worsen when left uncorrected. With a posterior cycle slip correction, satellite PRN 32 is corrected accordingly and the overall residuals are improved by at least a factor of 2.

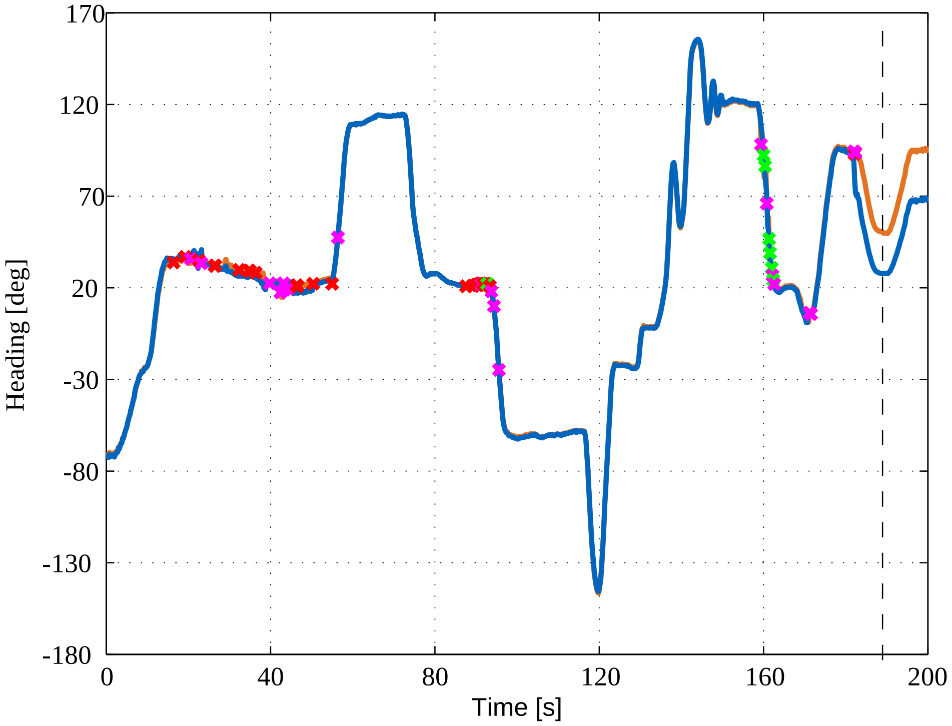


Figure 6.7: Heading determination of a test drive at ESA-AZO, Oberpfaffenhofen. Headings were obtained using the dynamic-based and subsequently a posteriori CSC. Different colors show that different cycle slip correction modes were used.

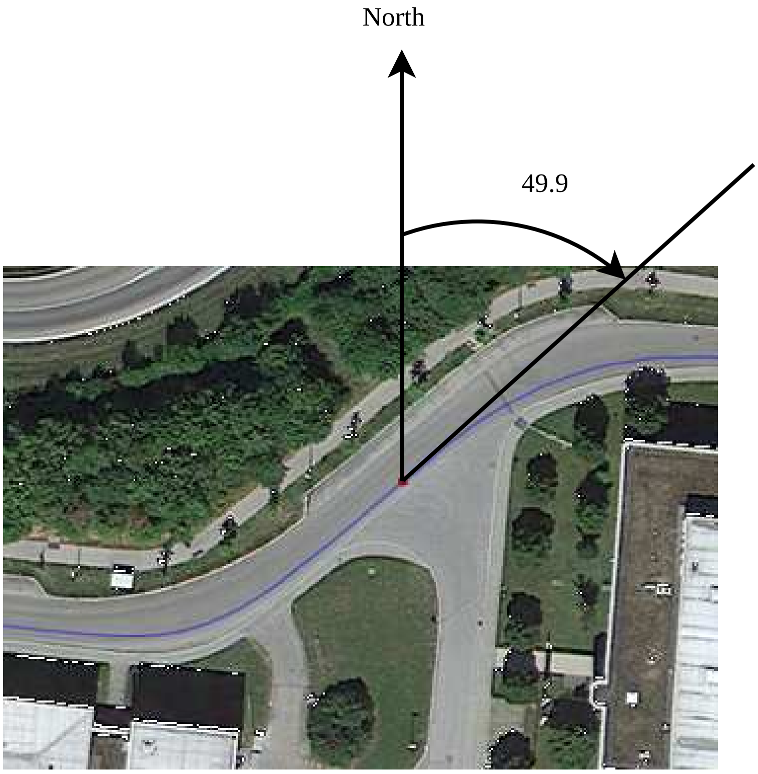


Figure 6.8: Heading at 189 s during the test at ESA-AZO, Oberpfaffenhofen

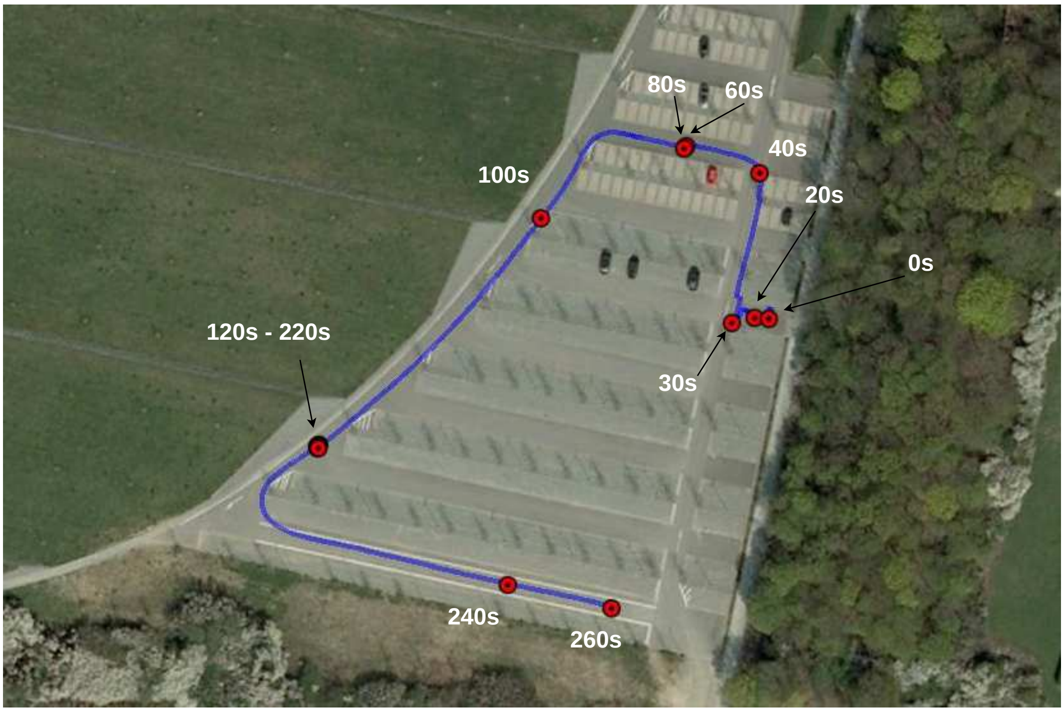


Figure 6.9: Test drive conducted at VW, Wolfsburg. After initialization, the car moved round the open area. At $60-80$ s $120-220$ s, the car was stationary.

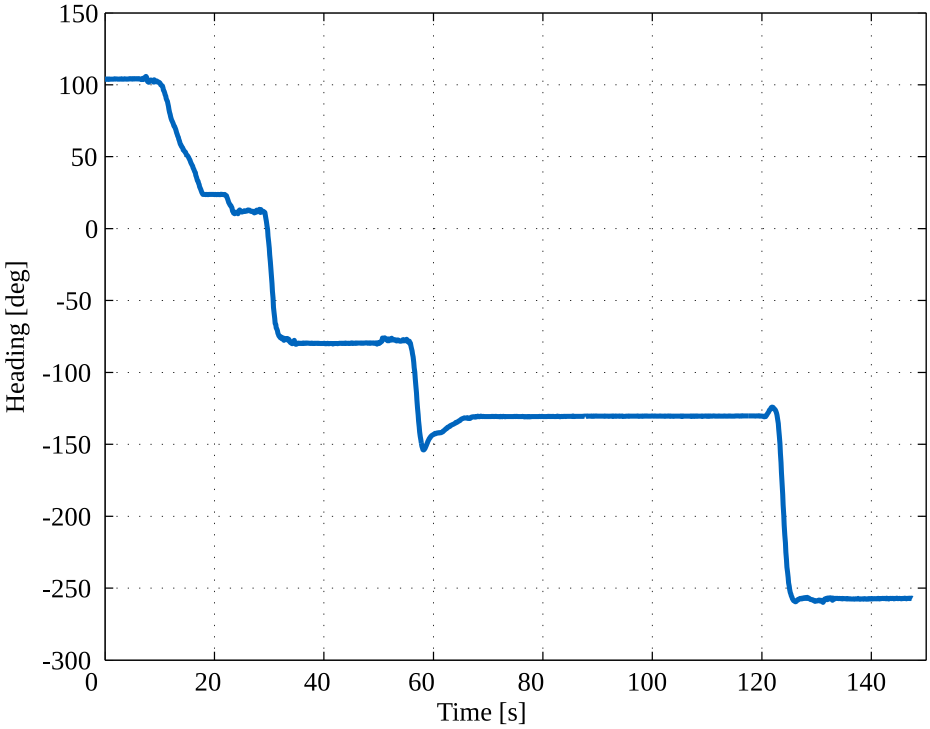


Figure 6.10: Headings obtained from the test drive at VW, Wolfsburg.

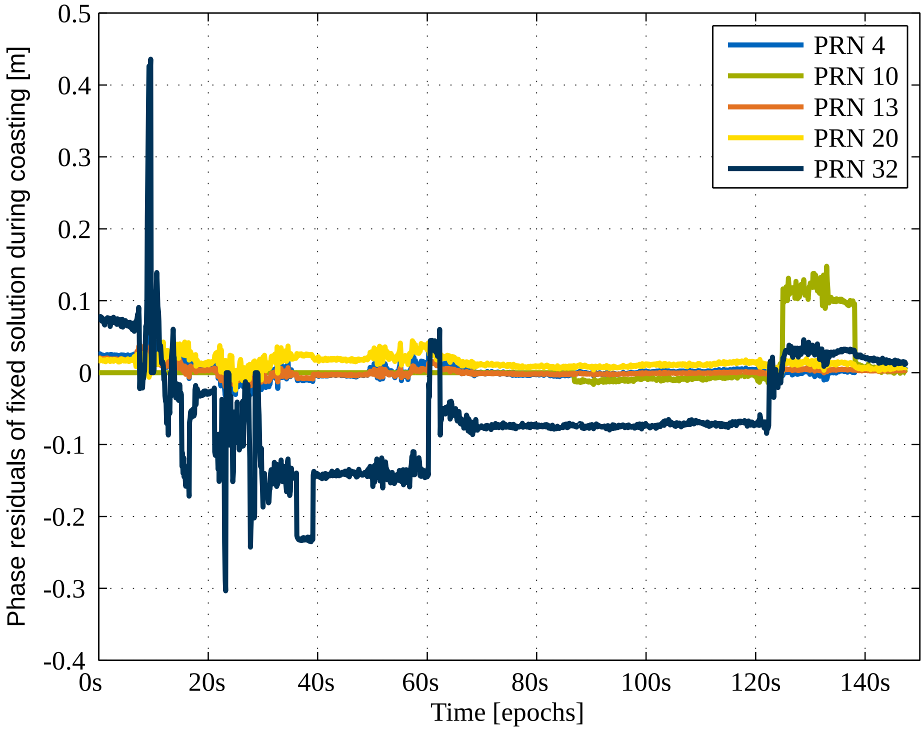


Figure 6.11: Test drive at VW Wolfsburg, : Corresponding phase residuals where dynamic-based cascaded cycle slip correction.

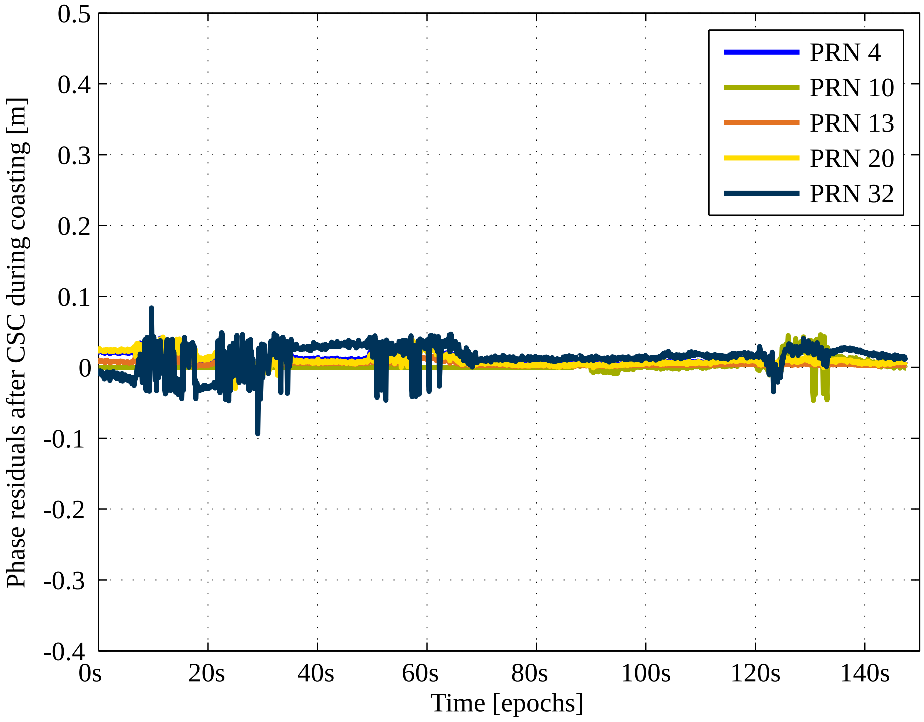


Figure 6.12: Test drive at VW, Wolfsburg: Corresponding phase residuals where dynamic-based cascaded cycle slip correction was coupled with a fixing posterior correction.

7. ANAVS PAD: Position and Attitude Determination research platform

Advanced Navigation Solutions (ANAVS)[1] is actively involved in absolute and relative position determination as well as attitude determination. This master thesis consists mainly of the attitude determination part or more precisely, heading determination. Algorithms and methods explained before are put into practice in the *Position and Attitude Determination, (PAD)* system. The system is tested extensively to provide precise headings in terrestrial navigation. However, the system can be further developed for maritime and aeronautical use.

The following sections will give a description of how the system is configured hardware-wise and how the different methods are structured in modules. The hardware part of the system is mainly conceived by Philipp Berthold and carefully developed by Philipp Berthold and Naoya Oku.

7.1 Hardware configuration

Figure 7.1 outlines the basic setup of the heading determination platform. An i7 processing unit and two u-blox LEA-6T receivers are integrated in a casing. Two receivers are needed because we are performing relative positioning. Two jacks on the rear-side of the casing are used to connect the GPS antennas to the receivers. The box can be powered with a variable voltage between 6 and 30 Volts. The high voltage input range is to ensure that the operation remains reliable when the power supply is connected to a running car battery. Such security measure also enables the box to be connected to high voltage board supply on a ship or a truck.

The box uses an Intel i7-CPU as in the research phase, the algorithms are implemented in MATLAB which requires high computing resources for a real-time system[2]. The box is passively cooled with heatpipes integrated in the aluminium casing.

The box can output results from the code to other electronic devices with a graphical display via a wireless local area network (WLAN) connection. However, MATLAB only allows a single-user connection. Philipp Berthold developed a network interface (Fig. 7.2) which acts as a router and outputs the results to up to 12 users.

[1] Located in Munich, Germany, ANAVS is created in October 2011 by Dr. Patrick Henkel, Prof. Christoph Günther, Sabine Schmitz, Peter Schmitz and Juan Cardenas.

[2] Since we are using 5Hz receivers, realtime requirements are met if each epoch is processed in less than 0.2 second.

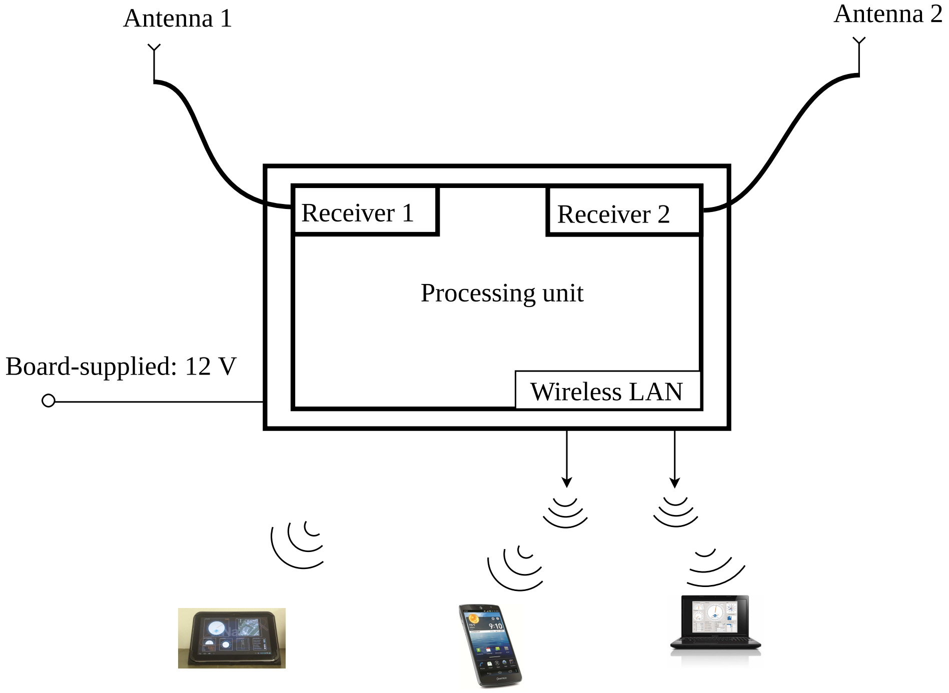


Figure 7.1: Network interface of ANAVS PAD system

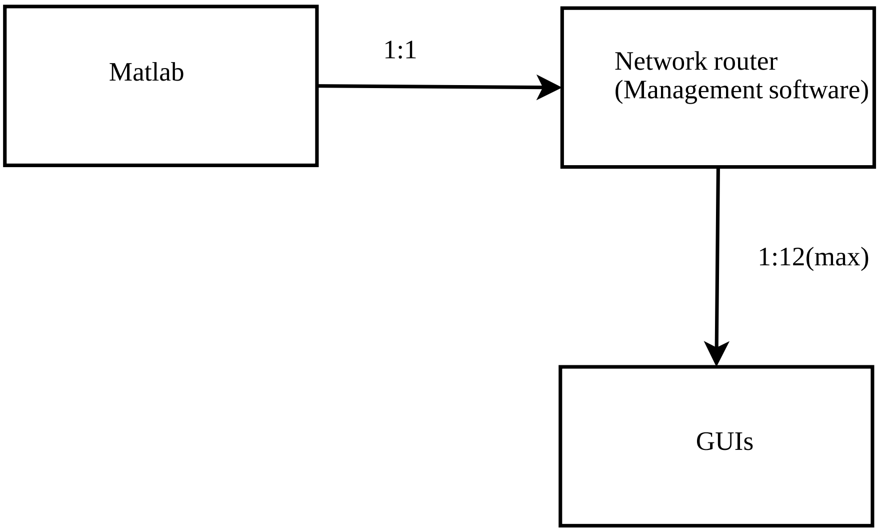


Figure 7.2: Network interface of ANAVS PAD system

7.2 Software architecture

The code is structured with modules. Each module contains an algorithm aiming to solve the same problem. Figure 7.3 depicts the general overview of the modular architecture.

First the signal received from satellites is processed in the u-blox receivers to output .ubx files (containing navigation and observation messages) which are later parsed to provide time (`week_1`,`week_2`, `tow_1`, `tow_2`), orbit data (`Orbit`) as well as code and phase measurements (`rho_1`, `rho_2`, `phi_1`, `phi_2`). These data are obtained for every epoch and are temporarily stored in two First In First Out (FIFO) buffers: `sampleFifo1` and `sampleFifo2`. The buffers then feed data epoch by epoch to the main body of the code.

Before using the phase measurements, we have to resolve the integer ambiguities (see eq. 2.2). In the case where cycle slip is absent and the satellites are continuously visible, the integer ambiguities are invariant. With this concept in mind, we separate the code into two main parts: *Initialization* and *Coasting*. The former intends to solve the ambiguities which can then be used in the latter to determine the baseline vector. Since the ambiguities are constant once they are resolved, no ambiguity resolution is required in the coasting phase. However, in the coasting phase, cycle slips can happen and have to be corrected to ensure the integrity of the phase measurements.

Due to the fact that we have two major modules which do not run simultaneously, it is practical to set different timelines for each module and a timeline for the whole process. `ep` denotes the epochs of the whole process while `t_ep` and `t_ep_meas` denote the independent timelines of the initialization and the coasting phase respectively (see Figure 7.4). `t_ep` counts from *1* to `N_ep_init` which indicates the number of epochs needed for initialization while `t_ep_meas` has no upper bound. Bear in mind that `t_ep_meas` $= 1$ when `ep` $=$ `N_ep_init` $+ 1$.

7.2.1 Initialization

This section explains what each module incorporated in the main *Initialization* module is performing. Figure 7.4-7.7 illustrate in the middle different modules with blue indicating that the module is a separated module from the `Initialization.m` while white indicates that the module is written in `Initialization.m`. On the left are all variables needed in the module and on the right are variables which are created or modified in the module. The list of output variables exclude intermediate variables which are used only within the module.

In the initialization phase, the receivers have to be stationary in order to eliminate any eventual cycle slips with a simple and reliable triple difference method described in section 6.2.2.3. With the cycle slips eliminated, we can assume that integer ambiguities are constant and proceed with a solution exploiting redundancy of measurements from different epochs.

7.2.1.1 Absolute Position Determination - Initialization

As described in Chapter 4, float ambiguities have to be determined before the search of an integer solution. Due to the noise level in the received signal caused by either the noisy environment or the low-cost receiver, we use a system of equation which exploits the redundancy (see eq. 3.9). Using more measurements can improve the efficiency of integer search [24]. A total number of `N_ep_init` epochs (in our case, 800 epochs) are collected and stored *column-wise* in variables bearing the same name ending with `*_mepoch` (eg. `rho_1_mepoch(1:32,t_ep)` = `rho_1`).

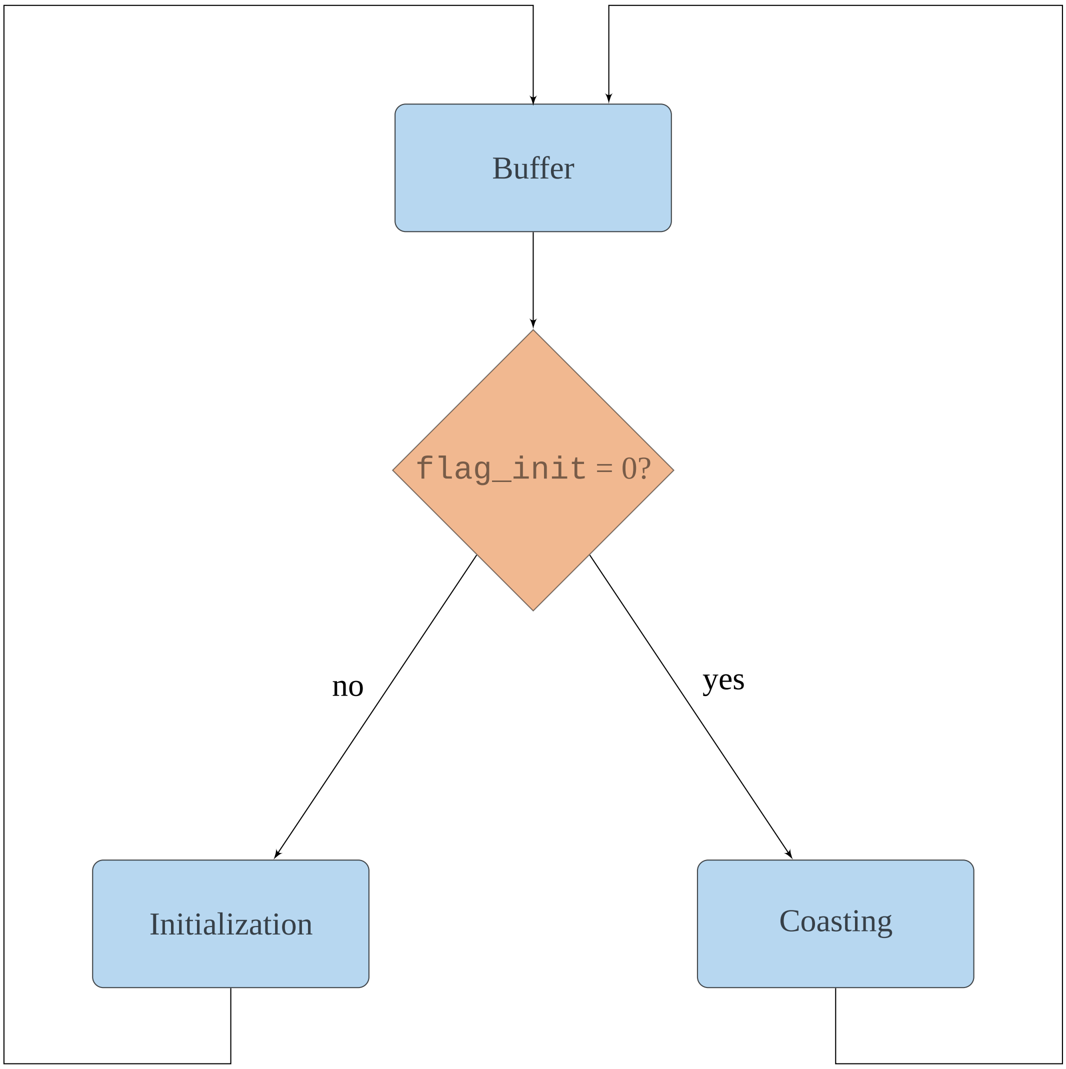


Figure 7.3: General architecture of modules

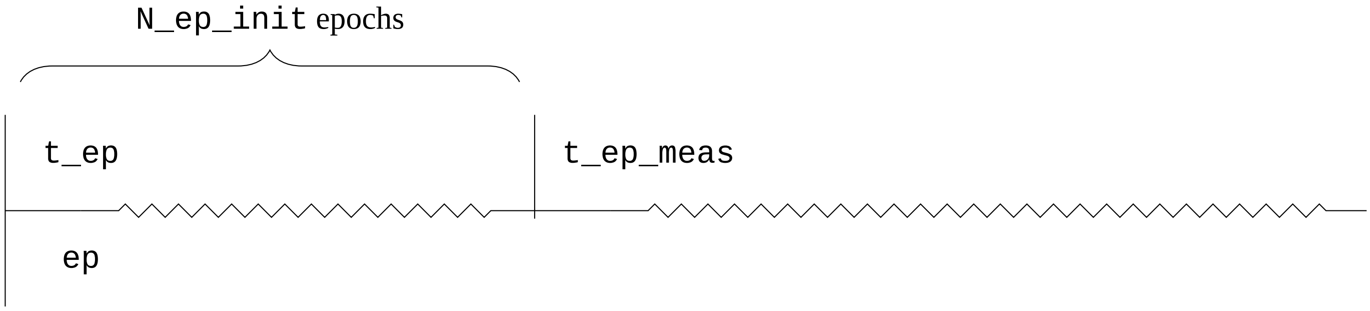


Figure 7.4: Timelines in the code

During the collection of N_ep_init epochs, the code-based absolute position of each receiver is determined with an iterative least-squares method like in algorithm 1. Nevertheless, the main goal here is to obtain the receiver clock offsets and the geometry matrices from the least-squares solution. A more precise code-based absolute positioning using carrier smoothing will be performed again at a later stage. Other data such as satellite position is also calculated using the orbit data.

Variables required for receiver clocks synchronization such as the geometry matrices, elevation angles and the satellite positions are stored *row-wise* in matrices which begin with E_32_*. Receiver clock offset for each receiver is also stored in RX_CLK_1 and RX_CLK_2.

With the absolute positions determined, latitudes and longitudes of each epoch are also stored *column-wise* in lat_*_mepoch and lon_*_mepoch.

t_ep is incremented by 1 after each epoch until N_ep_init.

7.2.1.2 Determination of available satellites and reference satellites

This module is not a separate module but incorporated in the *Initialization* given its relatively simple task. It searches for satellites which provide code and phase measurements throughout the whole N_ep_init epochs. The PRN identifiers of these satellites are stored under AvailPRN_init.

An elevation mask ele_mask is also applied to all satellites of AvailPRN_init. Satellites with an elevation lower than ele_mask will be excluded from AvailPRN_init. Finally, N_S number of satellites are available and the entire *Initialization* module will only be working with these satellites.

Satellite with the highes elevation angle normally suffer the least from multipaths. Therefore, we select the most elevated satellite PRN_ref to be the reference sastellite. PRN_ref_index denotes the reference satellite index in AvailPRN_init. AvailPRN_DD_init contains all satellites from AvailPRN_init excluding the reference satellite.

Single differences are determined with eq. 3.1 and stored *row-wise* in D_SD_PHI_m_s_srefR* or D_SD_RHO_s_srefR*. With eq. 3.4, double differences are determined and stored *row-wise* in D_DD_PHI_m and D_DD_RHO.

7.2.1.3 Synchronization Correction - Initialization

The problem that be intend to solve involve two receivers which are not synchronized. Each receiver has its own clock; in other words, the measurements of the same epoch are taken at different time instants. The offset between both receiver clocks is although only at millisecond level, however as satellites are moving at $4km/s$, such an offset can cause an error of $1m$ in range measurements. Henkel et al. [8][10] developed a synchronization method which takes the clock of one of the two receivers as the reference clock and interpolate measurements from the other receiver accordingly.

However, the synchronization is not done for each epoch but interpolated linearly instead. The double difference correction `D_PSEUDO_DD` to eliminate the differential clock offset is calculated using eq. 4.2. Phase and code double difference for each epoch are corrected with eq. 4.3 and stored *column-wise* in `phi_dd` and `rho_dd`.

7.2.1.4 Remove the first $1 +$ `max_int_degree` epochs

Due to the interpolations for synchronizing the receiver clocks, `D_PSEUDO_DD` is not calculated for the first $1 +$ `max_int_degree` epochs. Measurements for these epochs are hence not correctable. We have to neglect these epochs in variables with `*_mepoch` for the subsequent operations and start only with epoch `t_start_afix` $= 2 +$ `max_int_degree`.

At the same time, to keep homogeneity, we transpose the satellite position matrices, double difference correction as well as the geometry matrix for receiver one so that each epoch is stored *column-wise* in `x_s_ECEF_Rx1`, `x_s_ECEF_Rx2`, `pseudo_dd_init` and `H_1`.

In all following modules, we only use geometry matrix from receiver 1 as the normalized elements in the matrix do not vary much with a short baseline.

7.2.1.5 Cycle Slip Detection and Correction - Initialization

After synchronizing the receiver clocks, double difference cycle slips are corrected using a triple difference approach (see section 6.2.2.3). `csc_accumulated` is incremented with the detected cycle slips (already multiplied with $\lambda/2$. Since cycle slip affects all subsequent epochs as well (see figure 6.1, the accumulated cycle slips will be used to correct subsequent double difference measurements acquired.

7.2.1.6 Noise Statistics - Initialization

Before proceeding with baseline determination or ambiguity resolution, we have to define the statistical model of the system due to the random noise in eq. 3.9. We assume that the measurement noise is Gaussian distributed with zero-mean. The standard deviation is determined statistically using a polynomial fitting over double difference measurements of the `N_ep_init` epochs (see section 4.3.1. A statistical approach is preferred to a model as the receivers are stationary and therefore a second degree polynomial fitting is sufficient to match the satellite movement to the measurements (see figure 4.3). The standard deviations determined for each double difference measurement are used to fill the diagonal positions of the code and phase measurement covariance matrices (`Sigma_rho` and `Sigma_phi` respectively). The off-diagonal elements are theoretically difficult to determine. For practical use, we take $1/2 * \min(\sigma^k)$. `Sigma` take the code and phase covariance matrices as diagonal blocks to form the covariance matrix for measurement Ψ in eq. 3.9.

However, the noise statistics above are only determined for double difference measurements. We need section 4.3.2 to calculate standard deviations for code and phase measurements (`sigma_rho` and `sigma_phi`). These variables will be used in the next module.

7.2.1.7 Carrier Smoothing - Initialization

Although the absolute positions are already determined at an earlier stage when data from each epoch is collected, we perform again here an absolute position determination. This time however, is no longer using the pure code measurement but the smoothed code measurement. As explained before in section 2.1, the geometry change of the code and phase measurement should be identical. Therefore, smoothing the code measurement using a Hatch filter (see Fig. 2.1) can help to improve the precision of absolute positions.

`sigma_rho` and `sigma_phi` are used to determine the covariance matrix of the Hatch filter according to eq. 2.8. Absolute positions are calculated for each epoch (`x_1_smoothed` and `x_2_smoothed`). The middle of both receivers is taken as the absolute position of the vehicle `x_smoothed`. Its latitude and longitude are also calculated (`lat_smoothed` and `lon_smoothed`). The averaged absolute position of receiver 1 is also determined by taking the average over all epochs `x_Rx1_ave`.

This module also provides the continuity of the satellites `flag_cont`.

7.2.1.8 Stack double difference measurements in vectors

This module is incorporated in *Initialization* module itself. It stacks measurements from different epochs into vectors `phi_dd_multi_epoch` and `rho_dd_multi_epoch`.

7.2.1.9 Unconstrained Iterative Least Squares Float Solution

The final goal of the initialization phase is to determine the double difference integer ambiguities of the measurements. We use here the LAMBDA method (see section 5.1) which requires first the ambiguities to be solved in float form. As the PAD system is supposed to work under all possible circumstances including noisy urban environment, we exploit the redundancy by taking a system of eq. 3.9. By the end the this module, we will be able to determine a float solution for the baseline vector as well as the double difference ambiguities. However, this is not our goal here. Instead, the residuals `omc_phi_dd` and `omc_rho_dd` of code and phase measurement respectively are out main interest.

7.2.1.10 Selection Epochs Small Code Multipath

Since we have to wait for `N_ep_init` epochs before we can determine the ambiguities, during the waiting period, some irregularities can happen and result in a substantial multipath at certain epochs. The results can be distorted and redundancy will do us more harm than good. Therefore, by studying the residuals obtained from the float solution previously determined, we can eliminate irregular epochs in the code measurements as well as in the phase measurements. (see eq. 4.16 and 4.16). In the end, the "good" code measurements from epochs `less_code_multipath_epochs` and phase measurements from epochs `less_phase_multipath_epochs` will be used in the next step which consists of determining the final float solution.

`length_less_code_multipath` and `length_less_phase_multipath` are the number epochs from code and respectively phase measurements which survive the selection.

7.2.1.11 Unconstrained Iterative Least Squares Float Solution - Less Multipath

Likewise, the float solution is determined with algorithm 2 but using only selected measurements. The least-squares solution gives us an estimated position vector of the second receiver `x_Rx2_ave` with respect to the first receiver `x_Rx1_ave` as well as the float double difference integer ambiguities `N_acc`. The least-squares solution also provides us an important sub-product: the double difference ambiguity covariance matrix `Sigma_N_hat` which is used to define the metric of the search space in LAMBDA method.

The geometry matrix `H` provided by the iterative least-squares will also be used subsequently in other modules.

7.2.1.12 Intermediate absolute and relative positions

This module is incorporated in *Initialization* module itself. Latitude and longitude of the stationary vehicle are determined here. Using these latitude and longitude, we can established the transformation matrix `R_L` which transforms a position vector in the ECEF frame to a local ENU frame. The unconstrained baseline vector `b_hat_float` and its vector in the local ENU frame are calculated by subtracting the absolute positions. The float ambiguities `N_hat` is assigned with `N_acc` found in the above unconstrained least-squares method. The heading `head_float_unconstrained` is calculated with eq. 3.13.

The statistics `Sigma_b_N_hat`, `Sigma_N_hat` and `Sigma_b_hat` of unconstrained float solution are also calculated.

Code and phase measurements are eventually stacked in a column vector `Psi` as in eq. 3.9.

7.2.1.13 Constrained Float Solution with Tight Length Constrained

Float solution determined above does not exploit an *a priori* knowledge: the fix baseline length `b_length_apriori`. This module uses Lagrange optimization and Newton method (see section 4.4) to determine the baseline vector in local ENU-frame `b_hat_local_float_constrained` and the constrained float ambiguities `N_hat_constrained`. Eventually if the secant method in Lagrange optimization converges, the constrained float solution for ambiguities `N_hat` will be used rather than the unconstrained float solution. However, the unconstrained float solution cannot be skipped because we need the ambiguities covariance matrix `Sigma_N_hat` later in the integer search.

The geometry matrix `H` is transformed to be used with a baseline in the local ENU frame (i.e. $H_\mathrm{L} = H * R_\mathrm{L}^{-1}$). Its height component is truncated here as we only work with two dimensions.

7.2.1.14 Fixing LAMBDA Decorrelation and Search

We start the search with first of all LAMBDA method using `Sigma_N_hat` and `N_hat`. The search first executes a decorrelation for the above variables with a Z-transformation. We obtain the `Z` matrix and its inverse `Z_inv`, as well as the decorrelated covariance matrix `Q` and the float ambiguities `N_dec`. A search space volume is defined according to how many candidates we want the search to output. The search is then performed within the defined space volume and the candidates found are stored in `N_fixed` which is sorted according the squared norms `sqnorm_unconstrained` (see eq. 5.18 of section 5.1).

7.2.1.15 Fixing Constrained Search Tree Tight Length Constraint

LAMBDA method is efficient as a discrete integer search but however does not integrate the *a priori* knowledge (see section 5.2.2). Our configuration with a fix baseline length gives us an advantage which can be used to eliminate candidates which do not fulfill the baseline length constraint as in eq. 5.39. The sequential conditional least-squares principle can be considered as a tree search (see Fig. 5.1). Using a tight length constraint can help the search be more efficient as we eliminate invalid branches while going down the tree according to the inequality in 5.44 and candidates which survive through the whole search are stored in `N_fixed`.

`ncands` register the number of candidates found with a search space volume `Chi2_constrained` defined as described in section 5.2.4 using the residual of the best candidate found with LAMBDA method.

A back-transformation is performed on `N_fixed` using `Z_inv` to undo the decorrelation.

7.2.1.16 Constrained Fixed Solution For All Candidates

With integer ambiguities candidates found, we determine the baseline vector `b_check_unverified` with a tight length constraint as described in section 4.4. However, is the tight length constraint is not fulfilled (i.e $||b_{\mathrm{L}}|| - l_{\mathrm{ap}} >= 0.001$), the associated ambiguity candidate will be discarded. `b_check_fixed_constrained` takes in only those which fulfilled the length constraint and `ind_cand_rel` registers the indices of the candidates in the original list of candidates determined in the previous module. The WSSE of the phase residuals `WSSE_meas` are then calculated to sort the candidates.

With the verified baseline vector, the corresponding heading `head` is calculated with eq. 3.13.

7.2.1.17 Select the best candidate

The best integer ambiguities candidate is selected `N_fixed_selected` together with its associated baseline vector and the heading is calculated with eq. 3.13.

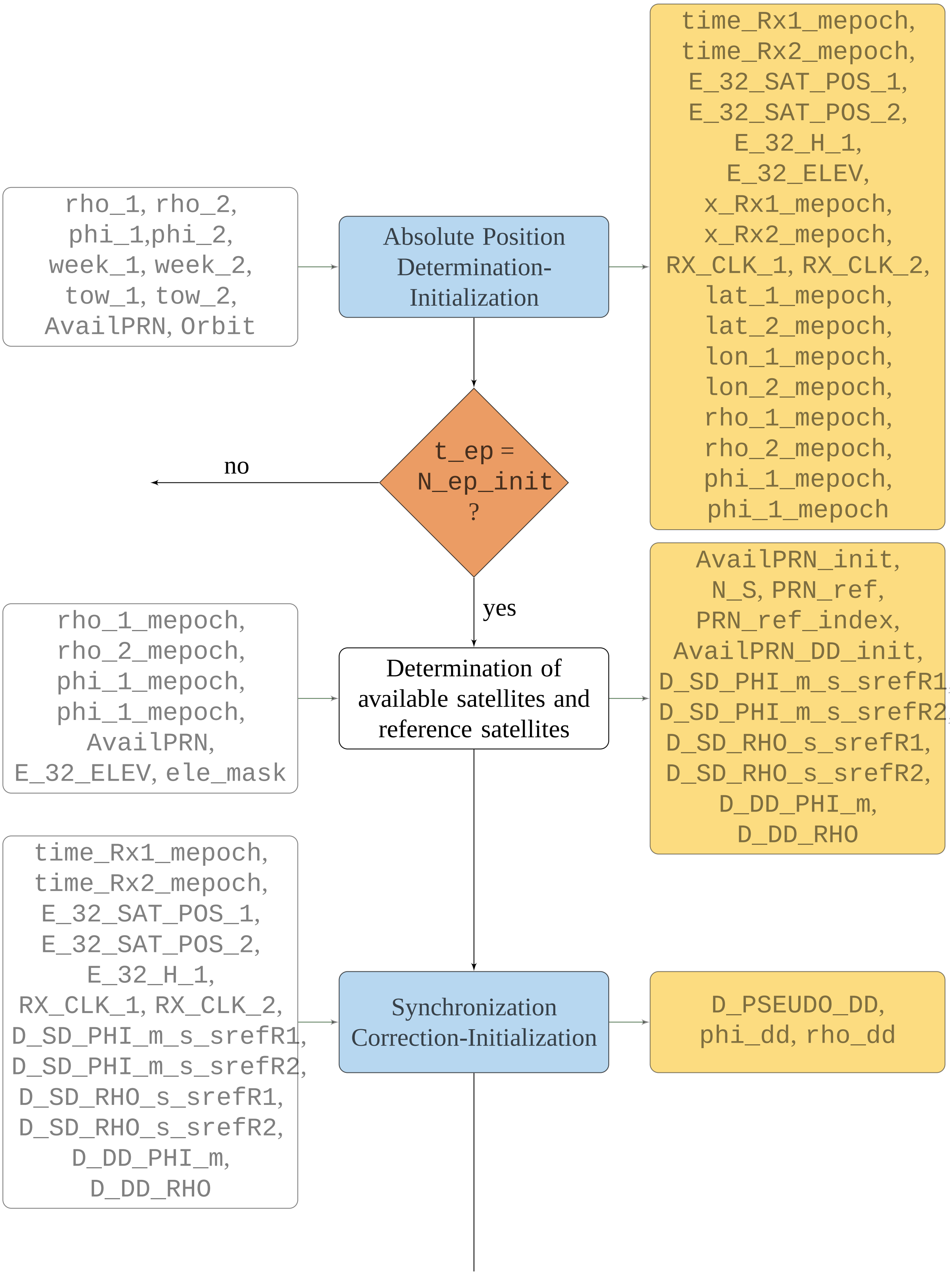


Figure 7.5: Initialization part-1

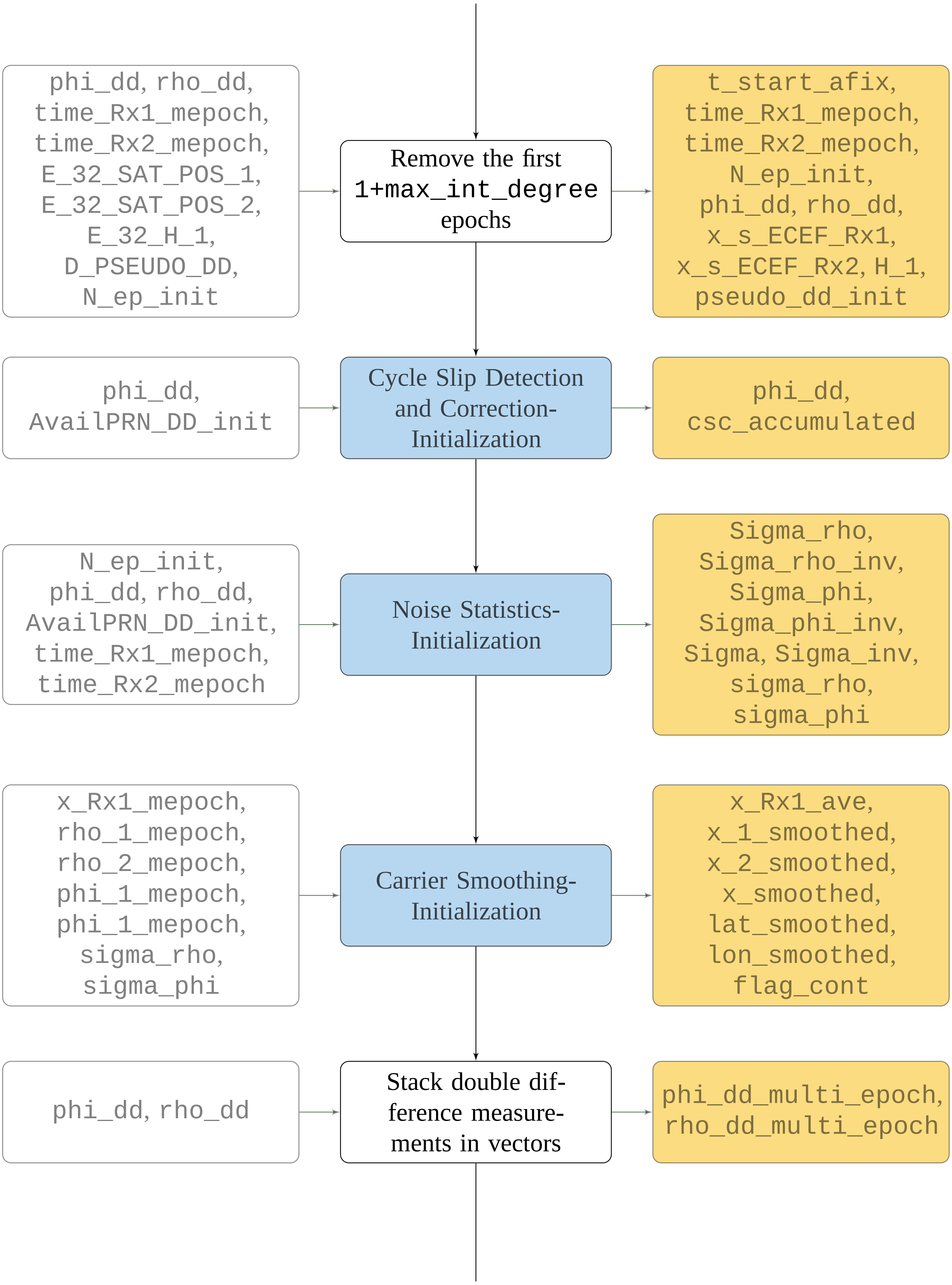


Figure 7.6: Initialization part-2

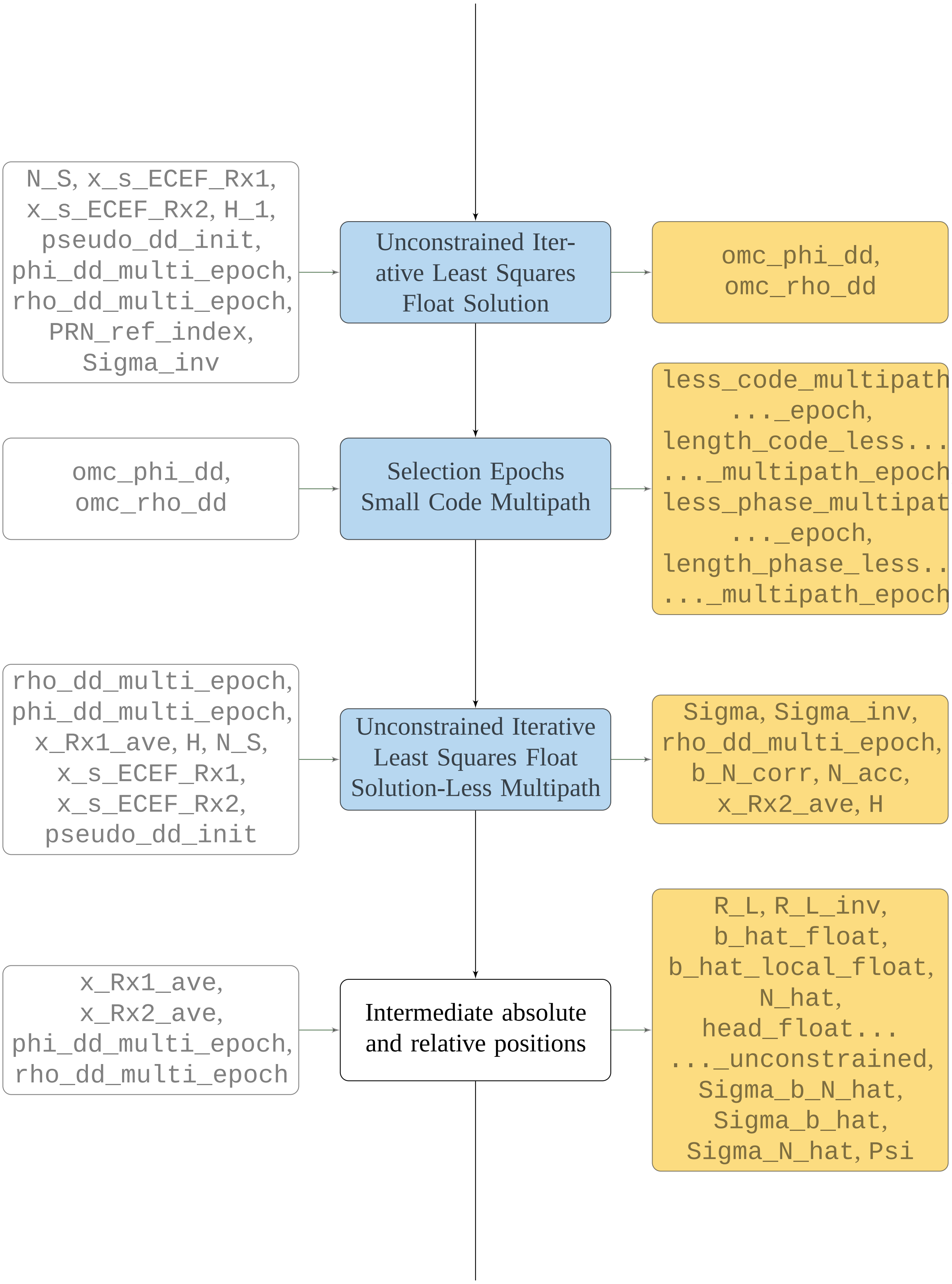


Figure 7.7: Initialization part-3

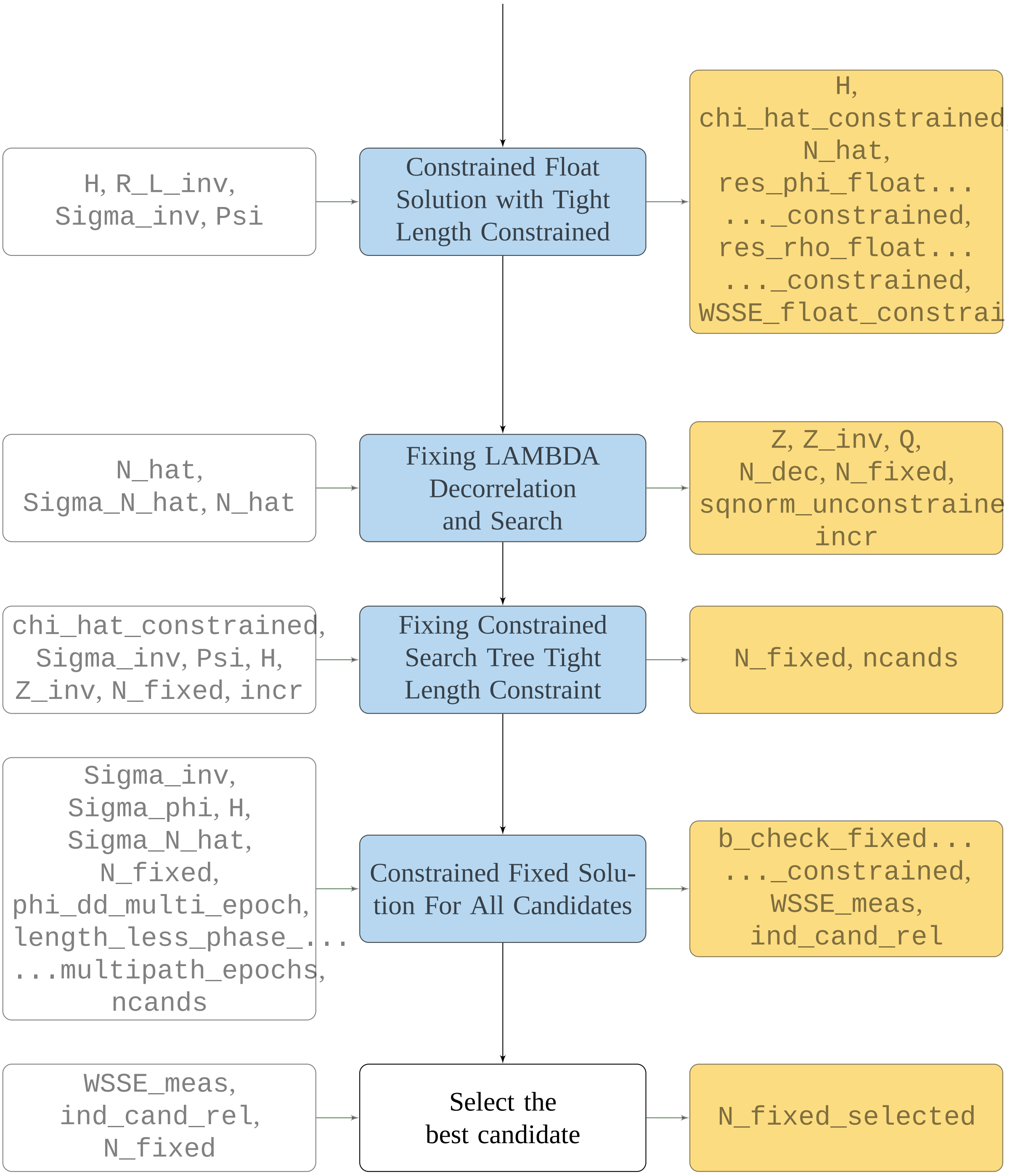

H, R_L_inv, Sigma_inv, Psi
Constrained Float Solution with Tight Length Constrained
H, chi_hat_constrained, N_hat, res_phi_float... ..._constrained, res_rho_float... ..._constrained, WSSE_float_constrai
N_hat, Sigma_N_hat, N_hat
Fixing LAMBDA Decorrelation and Search
Z, Z_inv, Q, N_dec, N_fixed, sqnorm_unconstraine incr
chi_hat_constrained, Sigma_inv, Psi, H, Z_inv, N_fixed, incr
Fixing Constrained Search Tree Tight Length Constraint
N_fixed, ncands
Sigma_inv, Sigma_phi, H, Sigma_N_hat, N_fixed, phi_dd_multi_epoch, length_less_phase_... ...multipath_epochs, ncands
Constrained Fixed Solution For All Candidates
b_check_fixed... ..._constrained, WSSE_meas, ind_cand_rel
WSSE_meas, ind_cand_rel, N_fixed
Select the best candidate
N_fixed_selected

Figure 7.8: Initialization part-4

7.2.2 Coasting

Once the receivers are initialized, any movement of the vehicle is allowed. With the determined ambiguities `N_fixed_selected`, we can now use the phase measurements. In the *coasting* phase, measurements are processed epoch by epoch in a real-time manner.

7.2.2.1 Absolute Position Determination

This module is incorporated in the *coasting* module itself. Similar to the first module in *Initialization*, the code measurement of the current epoch is used to determine the absolute positions (`x_Rx1` and `x_Rx2`), the receiver clock offsets (`dt_Rx1` and `dt_Rx2`), the geometry matrix for receiver 1 `H_1` and the elevation angle for available satellites `elev_1`.

Satelite positions are determined first with the orbit data and later corrected with the earth rotation: `xSat_Rx1_rot` and `xSat_Rx2_rot`. An elevation mask is applied to the satellites and `N_S` registers the number of available satellites `AvailPRN` which are elevated higher than the elevation mask.

7.2.2.2 Parameter Initialization-Coasting

This module is destined to structure the variables. The *Coasting* phase can be running continuously and therefore storing all data is not an option due to memory limitation. However, previous information is necessary for operations such as interpolation. We need at most `max_epochs` previous epochs to process the current epoch in the *Coasting* module (see interpolation in receiver clock synchronization). Therefore, we store the previous `max_epochs` epochs' measurements in the first `max_epochs` columns as well as the current epoch measurement in the last column of variables ending with `*_meas`. Older epochs are discarded from the system.

Satellite positions of the current epoch are also saved in `x_s_ECEF_Rx1_epoch` and `x_s_ECEF_Rx2_epoch` so that it is easier to mobilize the satellite position vectors later.

7.2.2.3 Reference Satellite-Coasting

While only satellites which are visible throughout `N_ep_init` epochs are used in the initialization phase, in the coasting phase, we use all visible satellite of the current epoch which are higher than the elevation mask `ele_mask`. Since the vehicle is moving and in an urban environment, the visible satellites become more variant. Therefore, the coasting phase must be designed to cope with such versatile use. The reference satellite `PRN_ref`has to be determined for each epoch individually. We use preferably the reference satellite from the previous epoch to avoid complex computation as long as the elevation of the old reference satellite does not fall below `ele_ref_sat_min`. Or else, we use a satellite with the highest elevation angle among all visible satellites `AvailPRN` as the reference. The reference satellite `PRN_ref` must also have been continuously visible for `max_epochs` so that it does not affect the linear interpolation used in the snychronization [10]. Its index in the available satellite vector `AvailPRN` is `PRN_ref_index`.

7.2.2.4 Determine double difference measurements

Double difference phase and code measurements (`phi_dd_meas` and `rho_dd_meas`) are determined with eq. 3.7 and 3.6 and are stored in the last column (`max_epochs`+1)-th column of `phi_dd_meas` and `rho_dd_meas`.

Accumulated cycle slips `csc_accumulated` are subtracted subtracted from the current phase double difference measurement `phi_dd_meas(:,max_epochs+1)`.

7.2.2.5 Synchronization Correction - Coasting

Similar to the beginning of the initialization phase, the receiver clocks are not synchronized which causes the measurements to be taken at slightly different time. Again, the clock of receiver 1 is used as the reference clock and measurements from receiver 2 are interpolated [10]. The correction terms are stored in the last column of `pseudo_dd_meas`; measurements from the last column of `phi_dd_meas` and `rho_dd_meas` (i.e the current phase and code measurements) are corrected with eq. 4.3.

7.2.2.6 Identification of new and old satellites

We first perform a check if all available measurements are still valid after the previous synchronization module. The fact that synchronization requires previous epochs to interpolate, satellites which are visible and higher than the elevation mask for the first time will not be properly correctly with the synchronization. Therefore, these satellites will also be excluded from the list of available satellites `AvailPRN`. After modifying the latter, we have to redefine `N_S` the number of available satellites and `PRN_ref_index`, index of the reference satellite in the list of available satellites.

Note that in the end, satellites which are in the `AvailPRN` are not only visible, but also elevated higher than the elevation mask and with correctable differential receiver clocks. With `AvailPRN`, we set up another list of satellites used in double difference `AvailPRN_DD`, which technically is a copy list of `AvailPRN` excluding the reference satellite.

Unlike *Initialization*, we do not always use the same satellites. In *Coasting*, we consider all visible satellites with a high enough elevation of the current epoch. Therefore at each epoch, we can have new satellites and lost satellites, which have to be treated differently when we correct the cycle slips. `PRN_lost` and`PRN_new` store lists of lost and new satellites respectively of the current epoch, with `PRN_lost_ind` the indices of the lost satellites in the list of available satellites of the previous epoch `AvailPRN_DD_last_epoch`, and `PRN_new_ind` indices of the new satellites in the list of available satellites of the current epoch `AvailPRN_DD`.

In this module, we also set up another list `AvailPRN_DD_intermediate`, which technically is the list of satellites used in double difference of the previous epoch `AvailPRN_DD_last_epoch`, with the lost satellites omitted (i.e. indices `PRN_lost_ind` are omitted). This list is for cycle slip correction as we can only correct for satellites available at the current epoch as well as in the previous epoch. A new satellite cannot be corrected as its integer ambiguity is not yet fixed. Therefore, if there are cycle slips in the measurement provided by the new satellite, it shall be considered as part of the ambiguity. Notice however that half a cycle slip in the new satellite will therefore not be taken into account as ambiguity fixing for the new satellite is only a multiple of a whole wavelength λ.

We reset the accumulated cycle slips `csc_accumulated` for lost and new satellites to zero.

7.2.2.7 Noise estimation

Unlike in the initialization phase, we do not estimate the measurement noise statistically as the satellites visibility is not continuous. A statistical approach will make noise estimation of newly tracked satellites impossible. One compromise is to take a noise model like explained in section 4.3.2 and [16] which determines the standard deviations (`sigma_rho` and `sigma_phi`) of each measurement base on the elevation angles `elev_1`.

Standard deviations for code measurements `sigma_rho` is used later in carrier smoothing. Since we intend to solve the baseline vector based solely on phase measurements, we determine here the double difference measurement covariance matrix `Sigma_phi` with only the phase measurement standard deviations. `sigma_phi` only provides standard deviation of phase measurement while Sigma_phi is the double difference measurement covariance matrix. We use hence a combination of standard deviation to form the variances and covariances:

$$\sigma_{\mathrm{dd}}^{k,2} = 2\sigma^{k,2} + 2\sigma^{\mathrm{ref},2} \tag{7.1}$$

$$\sigma_{\mathrm{dd}}^{kl} = 2\sigma^{\mathrm{ref},2}. \tag{7.2}$$

The weighting matrix of the double difference measurement Sigma_phi_inv is also determined to save computational effort later in the code.

7.2.2.8 Carrier Smoothing - Coasting

The receivers used in PAD system focus more on pseudorange measurements. When a clock jump is detected, which is always a $1ms$ jump, the receivers correct the jump for code measurements but not for phase measurements. We detect the jump by subtracting the current code measurement and the previous code measurement rho_1_prev. These jumps once detected, will be used to calculate the equivalent jumps for the case of phase measurements (phi_1_corr_acc and phi_2_corr_acc). The current phase measurements phi_1 and phi_2 are corrected. These jumps do not have to corrected for double difference measurements since the cancel off each other.

The next step consists of using carrier smoothing technique described in section 2.1 to determine the absolute position of the vehicle based on smoothed code measurements at the current epoch x_smoothed. The covariance matrix Sigma_x_smoothedof the absolute position is calculated with covariance matrices of both receivers absolute positions given while solving the least-squares problem.

res_1_smoothed is the residuals of each satellite code measurement after performing an iterative least-squares estimation of the absolution position of receiver 1.

7.2.2.9 Intermediate absolute positions and velocity determination

This module is incorporated in *Coasting* module itself. Using x_smoothed and x_smoothed_previous_epoch, we can calculate the speed of the vehicle v_meas(max_epochs+1. Note however that the calculate speed does not represent the current speed but can be approximated as the current speed when the time interval between two epochs is short.

Latitude lat_smoothed, longitude lon_smoothed are determined; together with these geodetic coordinates, we determine the transformation matrix R_L and its inverse R_L_inv. And with the transformation matrix, we can calculate the velocity v_meas_local_smoothed in the local ENU frame, and the heading of the vehicle based on the the direction of movement head_abs_pos. The rate of turn rate_of_turn_abs_pos is also calculated from the previous and current movement-based headings.

We determine here a threshold for the residuals of code-based absolute position res_1_smoothed_th and its associated WSSE.

The next steps consist of estimating the baseline. Therefore, by the end of this module, we prepare variables which we will need later in the iterative least-squares solution, namely the geometry matrix H and receiver 2 absolute position x_Rx2_meas(max_epochs+1) which is initialized with receiver 1 position x_Rx1_meas(max_epochs+1).

7.2.2.10 Cycle Slip Detection and Correction - Coasting

In order to use phase measurements for relative positioning, we have to first eliminate abnormal jumps in the measurements caused by cycle slips. While cycle slip correction is done relatively early in the initialization phase in order to have corrected phase measurements for statistical noise estimation, cycle slip correction is performed quite late in the coasting phase. The main reason is that while the vehicle is moving, the measurements are even noisier and cycle slips happen more often. As the satellite movement is no longer the sole movement concerned, a naive triple difference approach cannot be used here to detect and correct cycle slips. We need other *a priori* information as explained in chapter 6 in order to be able to use a MAP estimator to estimate the number of cycle slips. The complete cycle slip detection and correction is divided into several submodules as illustrated in figure 7.12. The different modes are detailed in chapter 6.

Before deciding which cycle slip mode to use, the module first prepares intermediate variables (i.e. `H_intermediate`, `Sigma_phi_intermediate`, `Sigma_phi_intermediate_inv`, `phi_dd_fixed_intermediate`) which take into account all visible satellites of current epoch used for double difference measurement `AvailPRN_DD`, while excluding new satellites `PRN_new_ind` in the list.

CSC_mode_1 Cycle slip correction at very low dynamics (see section 6.2.2.4). Mode 1 cycle slip correction will be selected (`flag_try_CSC_mode = 1`) when the phase-based heading `head_meas` of the previous 5 epochs do not vary much. By summing up the change in heading determined by phase measurements of the previous epochs is not enough as it does not take the current epoch into account. If the current epoch experiences a sudden turn, mode 1 cycle slip should also be excluded. Therefore, a linear fitting is done on code-based headings `head_abs_pos` from the previous 4 epochs as well as the current epoch. Comparing the gradients of the linear polynomial `head_abs_pos_fitted_complete` of the previous epoch and the current epoch will tell us if there is a sudden turn at the current epoch. Cycle slip detection and correction is performed as described in algorithm 6. Cycle slip candidate which, after determining the baseline with the corrected phase double difference measurements, bears the lowest residual WSSE `WSSE_CSC_td_cand` is the number of cycle slip detected and will be used to update `csc_accumulated`.

CSC_mode_2 Cycle slip correction at low dynamics. This mode is used when mode 3 fails to correct cycle slip or when conditions to use mode 1 or mode 3 are not fulfilled. The correction mode is performed according to algorithm 6.2.2.2.

CSC_mode_3 Cycle slip correction at high dynamics is performed when the vehicle is coasting at high speed `v_meas`. The code-based speed should also not be too noisy and `WSSE_res_1_smoothed` should be kept low. The first condition is necessary since the correction is dividing vectors with the speed. The last two conditions are to ensure that the code-based velocity is not too noisy as the *a priori* baseline knowledge `b_apriori` is derived using `v_meas_local_smoothed`.

7.2.2.11 Determine baseline and fix ambiguities of new satellites

With the corrected phase measurements, we determine the baseline vector `b_fixed_meas` and `b_loc_fixed_meas` with an iterative least-squares approach. With that, we also find an integer ambiguity candidate for the new satellites by minimizing the noise. However, we do not completely trust the candidate. We search around it and the candidate which has the smallest WSSE is the ambiguity to obtain the finally chosen ambiguity for new satellite `N_fixed_selected`.

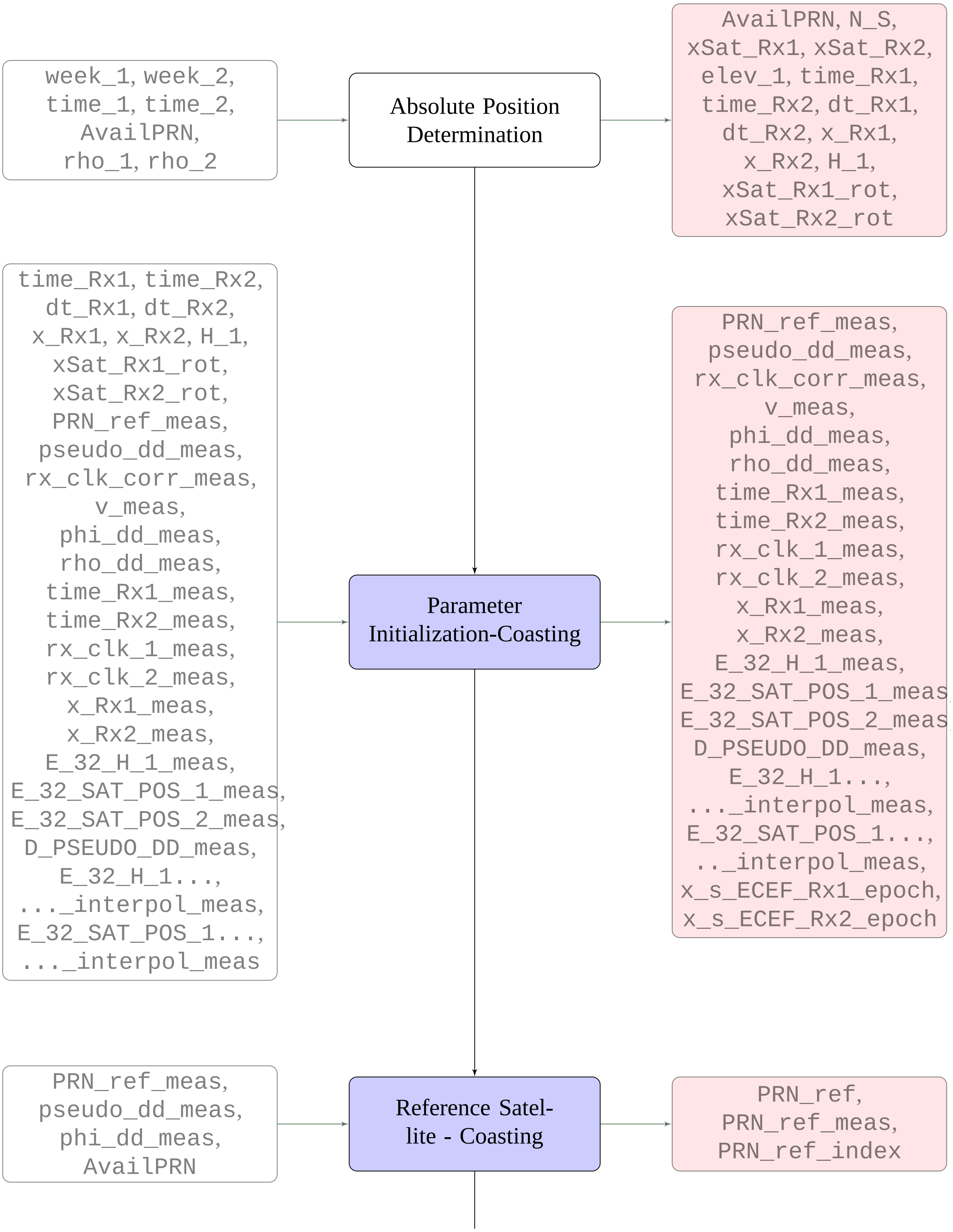


Figure 7.9: Coasting - Part 1

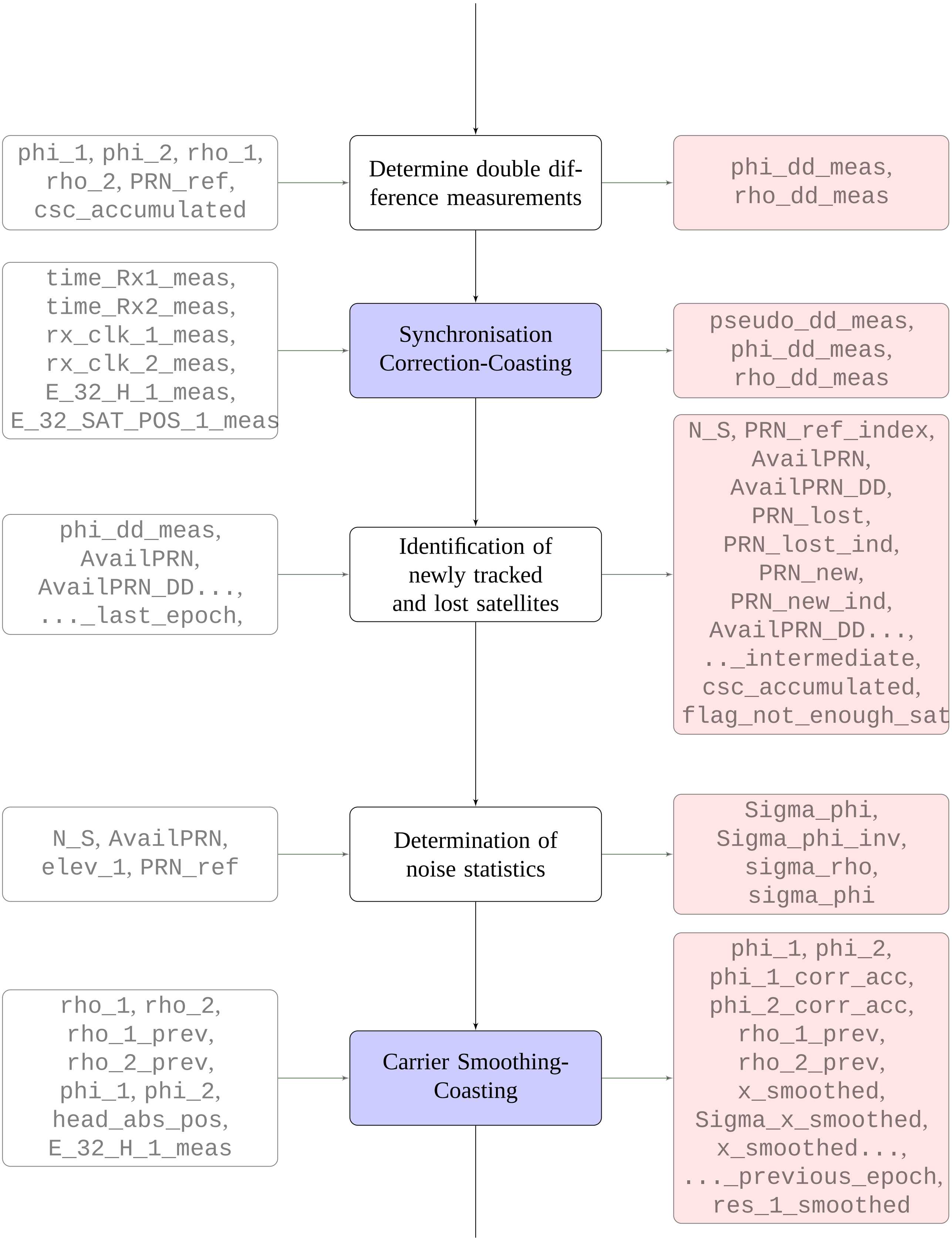

phi_1, phi_2, rho_1, rho_2, PRN_ref, csc_accumulated
Determine double difference measurements
phi_dd_meas, rho_dd_meas
time_Rx1_meas, time_Rx2_meas, rx_clk_1_meas, rx_clk_2_meas, E_32_H_1_meas, E_32_SAT_POS_1_meas
Synchronisation Correction-Coasting
pseudo_dd_meas, phi_dd_meas, rho_dd_meas
N_S, PRN_ref_index, AvailPRN, AvailPRN_DD, PRN_lost, PRN_lost_ind, PRN_new, PRN_new_ind, AvailPRN_DD..., .._intermediate, csc_accumulated, flag_not_enough_sat
phi_dd_meas, AvailPRN, AvailPRN_DD..., ..._last_epoch,
Identification of newly tracked and lost satellites
N_S, AvailPRN, elev_1, PRN_ref
Determination of noise statistics
Sigma_phi, Sigma_phi_inv, sigma_rho, sigma_phi
rho_1, rho_2, rho_1_prev, rho_2_prev, phi_1, phi_2, head_abs_pos, E_32_H_1_meas
Carrier Smoothing-Coasting
phi_1, phi_2, phi_1_corr_acc, phi_2_corr_acc, rho_1_prev, rho_2_prev, x_smoothed, Sigma_x_smoothed, x_smoothed..., ..._previous_epoch, res_1_smoothed

Figure 7.10: Coasting - Part 2

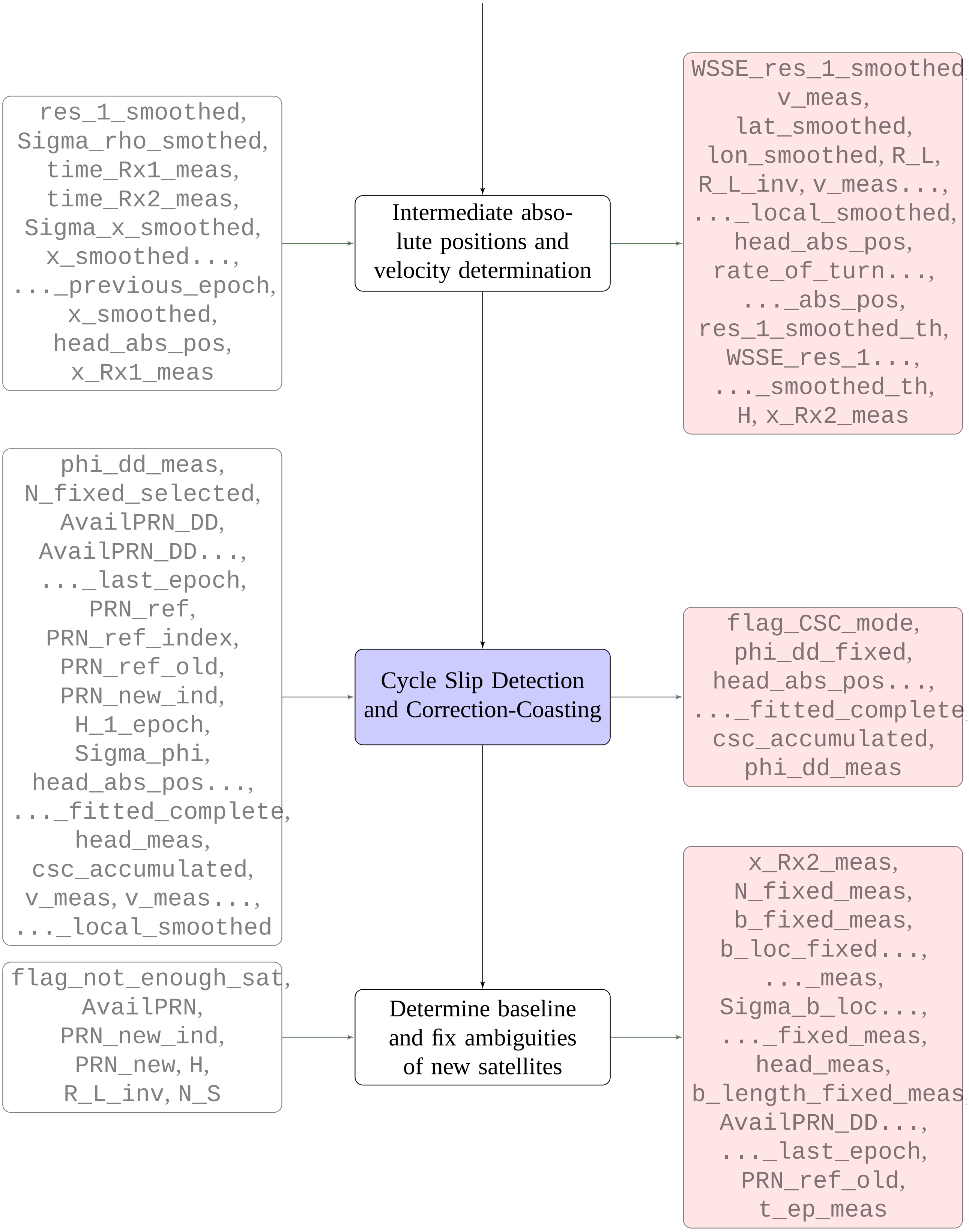

res_1_smoothed, Sigma_rho_smothed, time_Rx1_meas, time_Rx2_meas, Sigma_x_smoothed, x_smoothed..., ..._previous_epoch, x_smoothed, head_abs_pos, x_Rx1_meas
Intermediate absolute positions and velocity determination
WSSE_res_1_smoothed, v_meas, lat_smoothed, lon_smoothed, R_L, R_L_inv, v_meas..., ..._local_smoothed, head_abs_pos, rate_of_turn..., ..._abs_pos, res_1_smoothed_th, WSSE_res_1..., ..._smoothed_th, H, x_Rx2_meas
phi_dd_meas, N_fixed_selected, AvailPRN_DD, AvailPRN_DD..., ..._last_epoch, PRN_ref, PRN_ref_index, PRN_ref_old, PRN_new_ind, H_1_epoch, Sigma_phi, head_abs_pos..., ..._fitted_complete, head_meas, csc_accumulated, v_meas, v_meas..., ..._local_smoothed
Cycle Slip Detection and Correction-Coasting
flag_CSC_mode, phi_dd_fixed, head_abs_pos..., ..._fitted_complete, csc_accumulated, phi_dd_meas
flag_not_enough_sat, AvailPRN, PRN_new_ind, PRN_new, H, R_L_inv, N_S
Determine baseline and fix ambiguities of new satellites
x_Rx2_meas, N_fixed_meas, b_fixed_meas, b_loc_fixed..., ..._meas, Sigma_b_loc..., ..._fixed_meas, head_meas, b_length_fixed_meas, AvailPRN_DD..., ..._last_epoch, PRN_ref_old, t_ep_meas

Figure 7.11: Coasting - Part 3

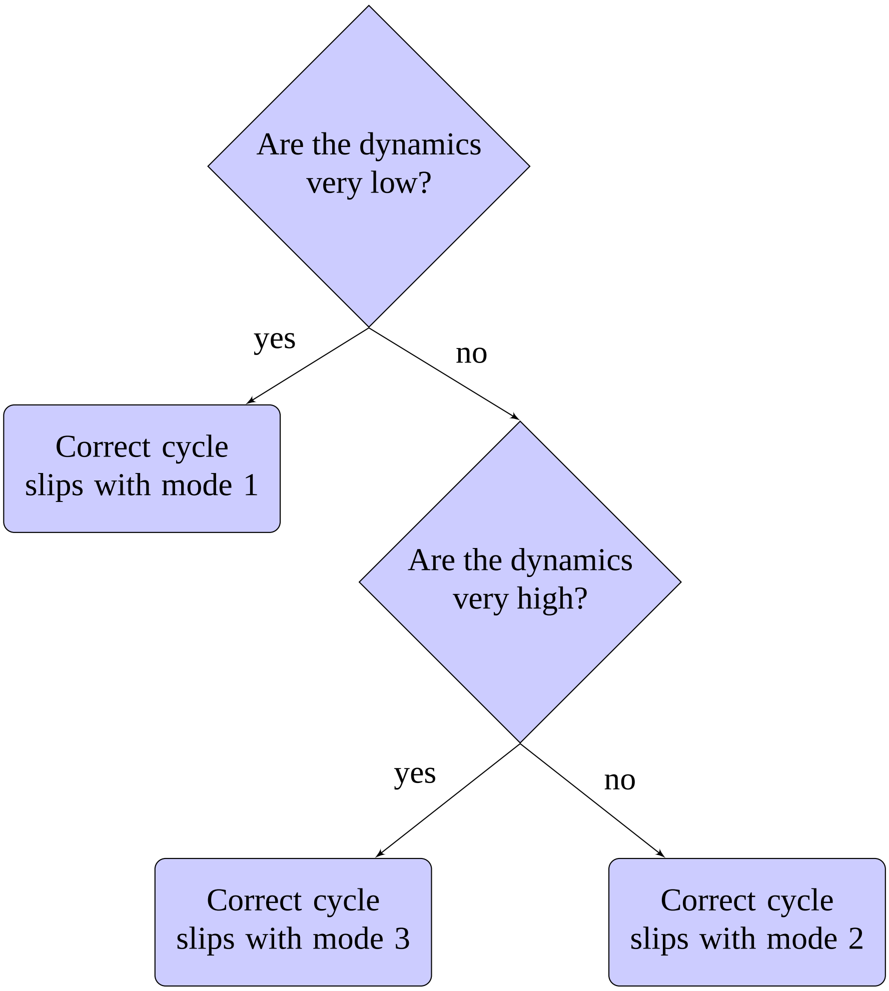

Are the dynamics very low?
yes
no
Correct cycle slips with mode 1
Are the dynamics very high?
yes
no
Correct cycle slips with mode 3
Correct cycle slips with mode 2

Figure 7.12: Cycle slip correction during coasting

7.3 Verification of ANAVS PAD system in heading determination

The ANAVS PAD system was verified in several test drives, where two low-cost receivers (u-blox LEA-6T) together with two single frequency low-cost patch antennas were in use. The patch antennas were mounted $1 - 1.5$ m apart on the roof of the car as shown in Fig. 3.2. Both antennas were aligned to the longitudinal axis of the car and had a negligible height difference.

7.3.1 Test Drive 1: Nymphenburg Palace in Munich

Figure 7.13 shows the track of the test drive conducted at Nymphenburg Palace in Munich. The integer ambiguities were resolved in the beginning with algorithm described in section 7.2.1 while the car was stationary. The orientation of the car was found without any movement. The track shown is subdivided into sections of 20 s.

Fig. 7.14 shows the course of the heading during the test drive at Nymphenburg Palace. The enlarged regions show that the noise of the heading estimate is in the order of only 0.1°. The abrupt heading changes at 70 s, 110 s and 140 s indicate u-turns or turns from one road into another road.

Fig. 7.15 shows the phase residuals of our MAP estimator for while the car is coasting along the track at Nymphenburg Palace. The phase residuals of the two satellites of highest elevation

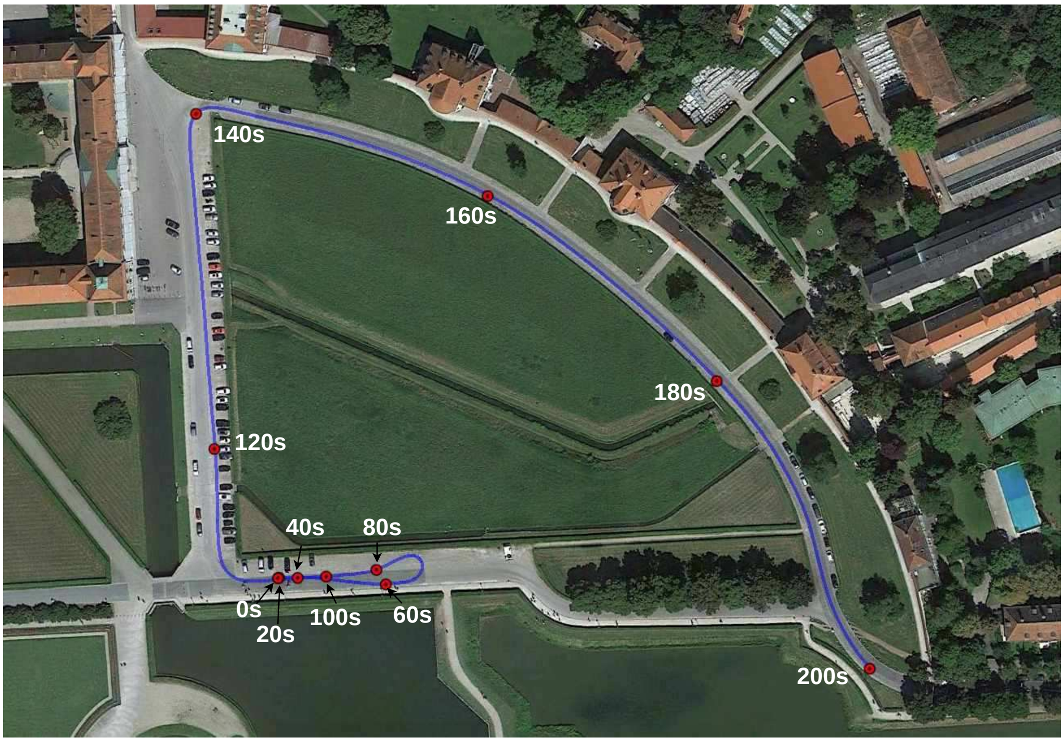


Figure 7.13: Track of car drive at Nymphenburg Palace. The integer ambiguities are resolved in the beginning with the car standing still. The subsequent track is subdivided into sections of 20 s.

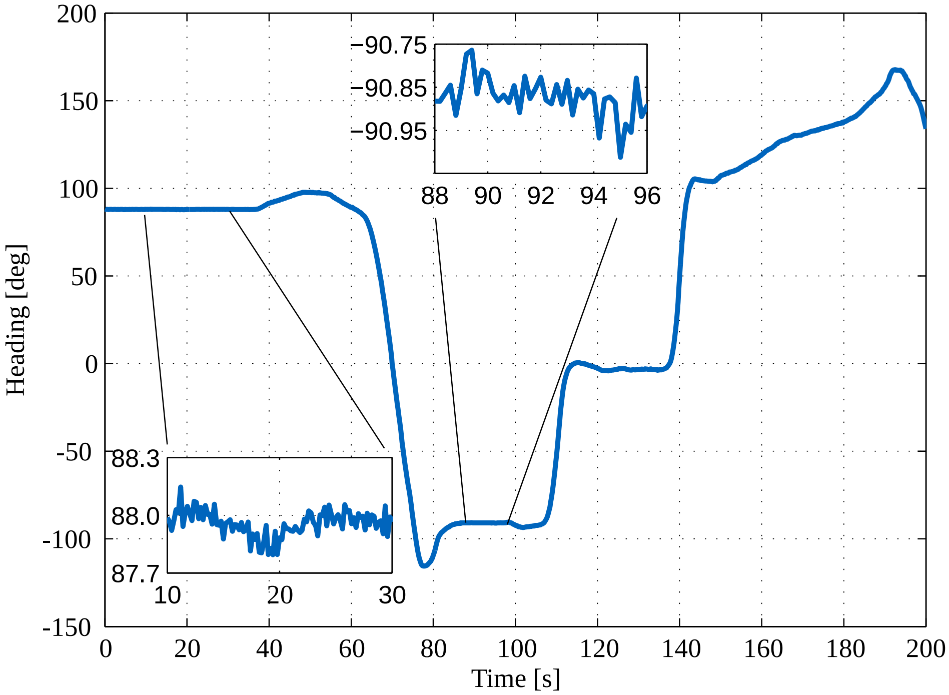


Figure 7.14: Heading of track at Nymphenburg Palace. The noise of the heading estimate is in the order of only 0.1°.

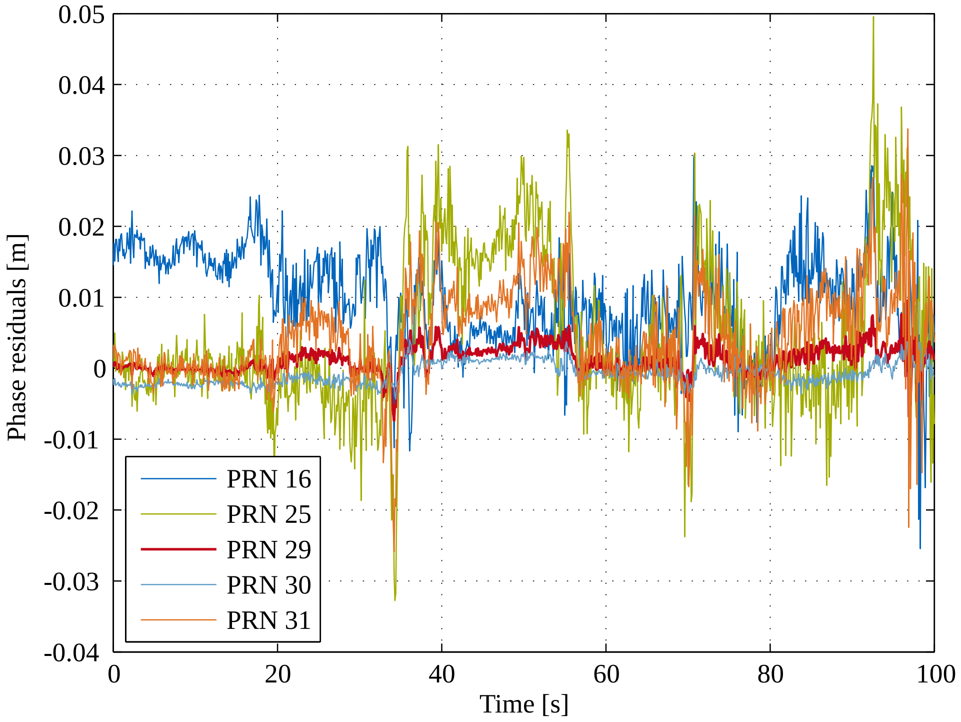


Figure 7.15: Phase residuals of fixed baseline solution for track at Nymphenburg: The phase residuals of all satellites are far below one wavelength. For the two satellites of highest elevation, the residuals are only a few millimeters.

(PRN 29, 30) are only of a few millimeters while the phase residuals of the other satellites are more affected by multipath but still remain unbiased and drift-free.

7.3.2 Test Drive 2: ESA AZO in Oberpfaffenhofen

We conducted another test drive in front of ESA/ AZO building in Oberpfaffenhofen. The track is shown in Fig. 2.2. The ambiguities were resolved resolved in the beginning while the car was standing in front of the building (see 0 s). Due to reflections from the concrete walls, the code measurements were affected by substantial multipath. Nevertheless, we still managed to resolve the integer ambiguities and determine the heading of the stationary car.

Fig. 7.16 shows the double difference phase residuals of the fixed MAP solution during the initial 160 s. The car was standing. The phase residuals of all four double differences are less than two centimeters over the complete period. This indicates a correct integer ambiguity resolution.

Fig. 7.17 shows the course of the heading as obtained by out MAP estimator. The estimated heading is varying only by a few degrees during the drive from one end to the other end of the road between 5 s and 35 s. The figure also shows three reversing sections in good agreement with Fig. 2.2.

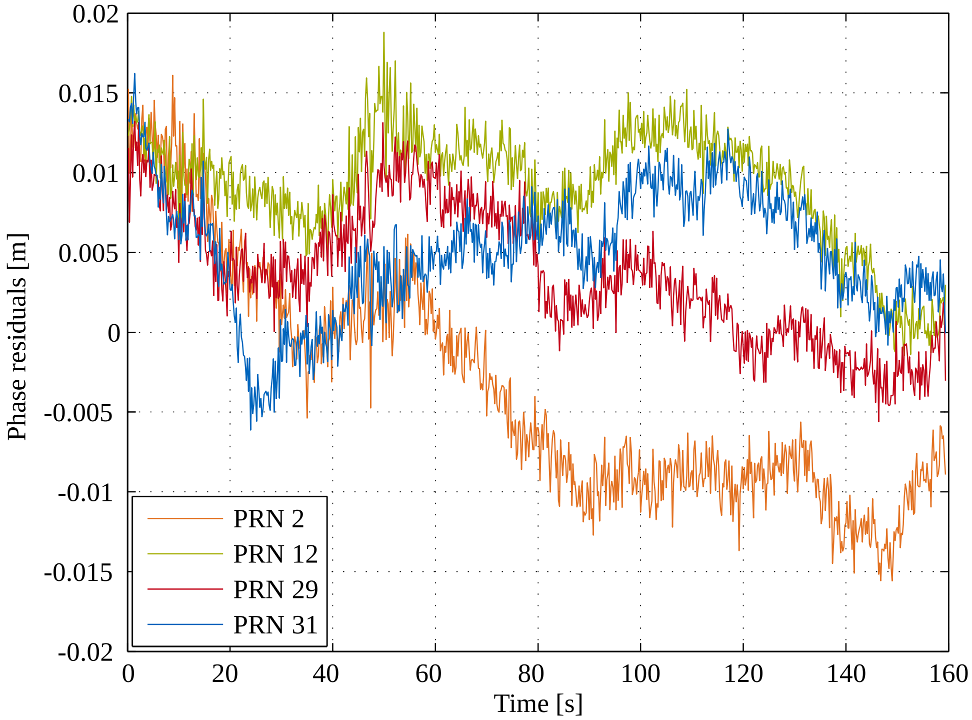


Figure 7.16: Phase residuals during initial integer ambiguity resolution.

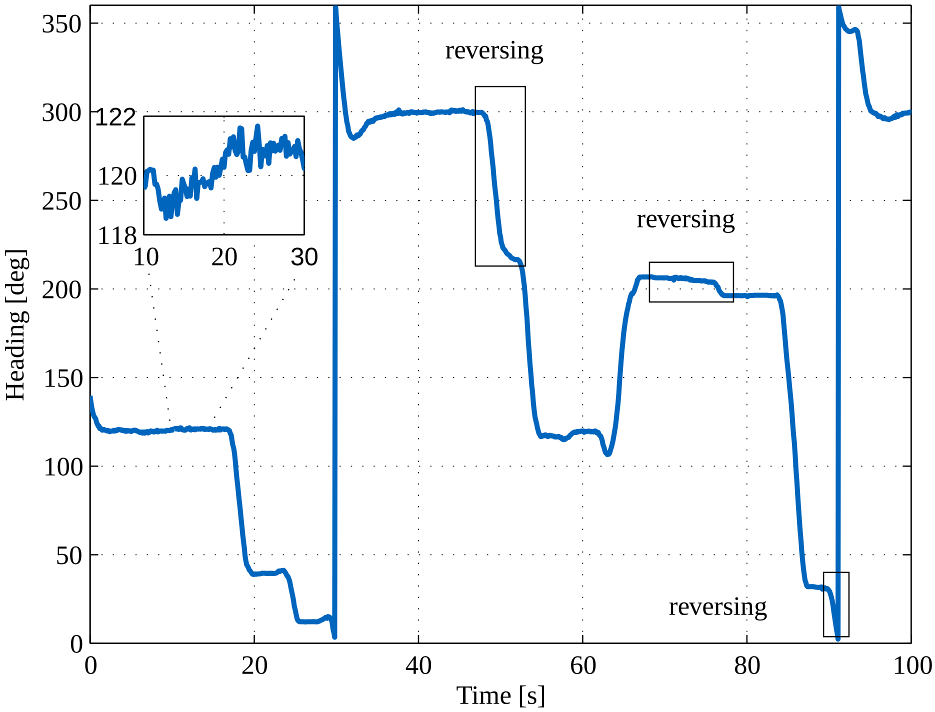


Figure 7.17: Heading of car during car drive in front of the ESA/ AZO building.

Appendix

A1. Weighted Sum of Squared Errors

A1.1 Cost function

One useful indication to know how precise if the estimated value without knowing the true value is by checking the measurement residuals. Residuals are very widely used, to test the deviation of the estimated and the measured values, for example in Kalman Filter. In the case where we have a linear relation between Y the measurement vector and X the parameter of interest that we wish to determine.

$$Y = AX + B, \tag{A1}$$

where A is the design matrix of the linear equation. A is deterministic but B is Gaussian distributed $B \sim \mathcal{N}(0, \varSigma)$.

We can determine X with an estimator and $\hat{X}$ denotes the estimated value. The residual or rather the cost function can be calculated by taking the difference between the measurement and the estimated value:

$$r = Y - A\hat{X}. \tag{A2}$$

Some measurement is better than others. Therefore, it is reasonable to incorporate a weighting matrix W in the problem. In the case of a least-squares estimator being used, the estimate takes the following expression [13]

$$\hat{X} = (A^{\mathrm{T}}WA)^{-1}A^{\mathrm{T}}WY, \tag{A3}$$

and the residual can be deduced as follows:

$$\begin{aligned} r &= Y - A\hat{X} \\ &= Y - A(A^{\mathrm{T}}WA)^{-1}A^{\mathrm{T}}WY \\ &= Y - PY \\ &= (I - P)Y, \end{aligned} \tag{A4}$$

where $P = A(A^{\mathrm{T}}WA)^{-1}A^{\mathrm{T}}W$ is a projector.

A1.2 WSSE, Weighted Sum Square Error

An approach to quantify how much the estimates deviate from the measurements is by weighting the residuals against the quality of each measurement. In order to ensure that any deviation, whether negative or positive value, is accumulated but not canceling each other, we can use the Weighted sum squared error (WSSE) to translate this deviation mathematically:

$$\mathrm{WSSE} = r^{\mathrm{T}} W r \tag{A5}$$

As the noise vector in satellite navigation is always assumed to be normally distributed with zero mean, the WSSE follows a central chi-square distribution [25].

List of Figures

List of Tables

Bibliography

[1] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed. American Institute of Aeronautics and Astronautics, 2009.

[2] C. K. T. S. Moafipoor, D. A. Grejner-Brzezinska, "Adaptive Calibration of a Magnetometer Compass for a Personal Navigation System ," in *IGNSS Symposium*, Sydney, Australia, December 2007.

[3] P. Teunissen, "Least-Squares Estimation of Integer GPS Ambiguities," in *General Meeting of the International Association of Geodesy*, Beijing, China, 1993.

[4] P. Henkel, "Reliable Carrier Phase Positioning," Ph.D. dissertation, Technische Universität München, Verlag Dr. Hut, ISBN 978-3-86853-716-1, July 2010.

[5] P. Henkel and C. Günther, "Three frequency linear combinations for Galileo," in *Proceedings of 4-th IEEE Workshop on Positioning, Navigation and Communication (WPNC)*, Hannover, Germany, March 2007, pp. 239–245.

[6] P. J. G. Teunissen, "The LAMBDA method for the GNSS Compass," *Artificial Satellites*, vol. 41, no. 3, pp. 89–103, 2006.

[7] ——, "Integer least-squares theory for the GNSS compass," *Journal of Geodesy*, vol. 84, pp. 433–447, 2010.

[8] P. Henkel, J. Cardenas, G. Giorgi, and C. Günther, "Precise attitude determination with low-cost satellite navigation receivers," 2013, *Submitted for publication in IEEE Transactions on Vehicular Technology*.

[9] C. Günther, *Lecture notes for satellite navigation*, 2013.

[10] J. M. Cardenas, "Master thesis," Ph.D. dissertation, Technische Universität München, March 2013.

[11] P. Teunissen, "On the GPS Double-Difference ambiguities and their partial search spaces," in *III Hotine-Maussi Symposium on Mathematical Geodesy*, L'Aquila, Italy, June 1994.

[12] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, 4th ed. Mc Graw Hill, 2002.

[13] P. Misra and P. Enge, *Global Positioning System*, 2nd ed. Ganga-Jamuna Press, 2006.

[14] K. Borre, "The Easy Suite - Matlab code for the GPS newcomer," *GPS Solutions*, vol. 7, no. 1, pp. 47–51, 2003.

[15] G. A. McGraw, T. Murphy, M. Brenner, S. Pullen, and A. J. V. Dierendonck, "Development of the LAAS Accuracy Models," in *ION GPS 2000*, Salt Lake City, USA, September 2000.

[16] P. Henkel and C. Günther, "Partial integer decorrelation: optimum trade-off between variance reduction and bias amplication," *Journal of Geodesy*, vol. 84, no. 1, pp. 51–63, January 2010.

[17] G. Blewitt, "Carrier-Phase Ambiguity Resolution for the Global Positioning System Applied to Geodetic Baselines up to 2000 km," *Journal of Geophysical Research*, vol. 94, no. B8, pp. 10.187–10.203, August 1989.

[18] P. J. G. Teunissen and C. Tiberius, "The LAMBDA method for integer ambiguity estimation: implementation aspects," *Publications of the Delft Geodetic Computing Centre*, August 1996.

[19] P. J. G. Teunissen, "The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation," *Journal of Geodesy*, vol. 70, no. 1-2, pp. 65–82, 1995.

[20] P. Henkel and J. J. Kiam, “Maximum A Posteriori Probability Estimation of Integer Ambiguities and Baseline,” in *IEEE Proceedings of Elmar Symposium*, Zadar, Croatia, September 2013.

[21] G. Blewitt, “An automatic editing algorithm for GPS data,” *Geophysical Research Letters*, vol. 17, no. 3, pp. 199–202, 1990.

[22] A. Lipp and X. Gu, “Cycle-slip detection and repair in integrated navigation systems,” in *Proceedings of the IEEE Position, Location and Navigation Symposium*, 1994, pp. 681–688.

[23] P. Henkel and P. Jurkowski, “Reliable Integer Ambiguity Resolution: Soft constraints on the baseline length and direction, and new multi-frequency code carrier linear combinations,” in *Internatiol Association of Geodesy Symposia*, 2011.

[24] P. Teunissen, “The invertible GPS ambiguity transformations,” *Manuscripta Geodaetica*, vol. 20, no. 1, pp. 489–497, 1995.

[25] S. C. Lo, B. B. Peterson, and P. K. Enge, “Proving the Integrity of the Weighted Sum Squared Error Loran Cycle Confidence Algorithm,” in *Navigation*, San Diego, USA, January 2007.