

Closed-Loop GNSS/INS Simulation Chain with RTK-Accuracy for Sensor Fusion Algorithm Verification

Andreas Schütz, Mohamed Bochkati, Daniel Maier, Thomas Pany, *Universität der Bundeswehr München*

BIOGRAPHY (IES)

Andreas Schütz is a research associate at the Institute of Space Technology and Space Applications (ISTA) at the Universität der Bundeswehr, München. His research focuses on multi sensor fusion, mainly GNSS/INS/LiDAR, precise GNSS positioning techniques and positioning integrity. He holds a Bachelor's and Master's degree in Geodesy and Geoinformation from Technical University Munich.

Mohamed Bochkati is a research associate at the Institute of Space Technology and Space Applications (ISTA) at the Universität der Bundeswehr, München. His research focuses on Deeply Coupled GNSS/INS integration and novel calibration techniques for MEMS-IMUs. Before he joined the University of the Bundeswehr, he was a research associate at the Institute of Geodesy (ife) at the Leibniz University Hannover, where his research activities include quantum inertial navigation systems and GNSS receiver clock modeling. He holds a Bachelor's and Master's degree in Geodesy and Geoinformation from Technical University Munich.

Daniel S. Maier received a bachelor in Physics in 2015 and a master in Applied and Engineering Physics in 2017 from the Technical University of Munich (TUM), Germany. Since 2017 he has been a research associate at the Institute of Space Technology and Space Applications of the Universität der Bundeswehr München. His current research interests include GNSS signal generation, signal authentication, spectral signal compatibility, Spoofing detection and antenna array processing.

Prof. Thomas Pany is with the Universität der Bundeswehr München at Space Systems Research Center (FZ-Space) where he leads the satellite navigation unit LRT 9.2 of the Institute of Space Technology and Space Applications (ISTA). He teaches navigation focusing on GNSS, sensors fusion and aerospace applications. Within LRT 9.2 a good dozen of full-time researchers investigate GNSS system and signal design, GNSS transceivers and high-integrity multi-sensor navigation (inertial, LiDAR) and is also developing a modular UAV-based GNSS test bed. ISTA also develops the MuSNAT GNSS software receiver and recently focuses on Smartphone positioning and GNSS/5G integration. He has a PhD from the Graz University of Technology (sub auspiciis) and worked in the GNSS industry for seven years. He authored around 200 publications including one monography and received five best presentation awards from the US institute of navigation. Thomas Pany also organizes the Munich Satellite Navigation Summit.

ABSTRACT

For future navigation applications like autonomous driving, air mobility or indoor navigation, GNSS only will not be sufficient to fulfill the challenging requirements these applications are demanding. Only GNSS in combination with other sensor systems can face these challenges ahead. To be able to develop such complex multi-sensor systems, it is crucial to first establish a continuous, robust and flexible procedure to test and verify algorithms and their implementation at any stage of their development. Originating from a pure software receiver, the Multi-Sensor Navigation Analysis Tool (MuSNAT) [1], developed at the Institute of Space Technology and Space Applications (ISTA) at the Universität der Bundeswehr München, has evolved into a versatile navigation platform using various advanced GNSS positioning techniques and sensors such as LiDAR and IMU. Using previously developed GNSS and INS simulation tools within the scope of this paper, we demonstrate how we at ISTA realized our closed-loop testing procedure and successfully generated and married synthetic GNSS and IMU data in an RTK/INS loose-coupling scenario. We successfully show the overall quality of the simulation and sensor fusion using the coincidence of a-posteriori INS biases and the initially simulated ones, in a high precision scenario, as well as the ambiguity fixing ratio of the GNSS phase measurements.

INTRODUCTION

Modern algorithm development, not only in the context of navigation but in a wide spectrum of fields, cannot meet its need of test data merely by traditional data collection. The most prominent example might be the gathering of training data for image processing using convolutional neural networks (CNN). But this statement also holds for GNSS and inertial data, as well as for LiDAR measurements, especially in the case of a representative number of datasets for edge or corner-cases being needed. Two main reasons may be accounted for that: Firstly, data collection is extremely time consuming. Secondly, due to the hardware-dependent nature of GNSS and inertial data, a wide variety of sensor characteristics needs to be considered within each observed test case. This, however, requires costly equipment and precise measurement synchronization at hardware level. So it might be worthwhile to consider a different approach, namely simulating observations, given the history and previous work done at ISTA, especially with respect to the MuSNAT Transceiver [2] and the Closed-Loop INS Simulation Tool [3]. The following will introduce the simulation tools and the navigation engine, the strategy on how these tools were combined to become the ISTA Satellite and Inertial Measurement Simulation System (SIMSS), and the quality of the synthetic integrated processing.

CLOSED-LOOP INS SIMULATOR

The Closed-Loop INS Simulation environment represents a framework to develop novel strapdown and IMU calibration algorithms, where the 3-dimensional reference motion, i.e. position, velocity and attitude are well known in advance. So, instead of spending several days in planning, calibrating and collecting IMU (+GNSS) data, such tool allows to generate these observations offline in the office and use them in post-processing, usually in conjunction with GNSS. To realize this, first, the error-free IMU observation, i.e. the specific force f_{ib}^b and rotation rates ω_{ib}^b in the x, y and z-direction should be computed in the body-frame (indicated by the superscript “b”). These observations require as input the position, velocity, and attitude expressed in the navigation frame (n-frame) representing the in the north-east-down realization (NED). This step is called the inverse strapdown mechanization equation (ISME), because the process of treating the IMU-observation is inverted w.r.t. the conventional way where the user obtains these sensor readings corrupted by noise and other constant errors, such as Bias or scale factor (SF).

Afterwards, both accelerometer and gyroscope signals undergoes different integration steps with different initial parameters in different stages, such as initial attitude obtained by static alignment or using start position information P_0 , to output at the outcome of the strapdown process, position, velocity and attitude information. The validation of this simulator has been established while comparing the input reference trajectory with the computed, unperfect, trajectory (see Figure 1). The simulator can achieve for different kinematic seniors the sub-meter level accuracy after 1 hour of integration time [3].

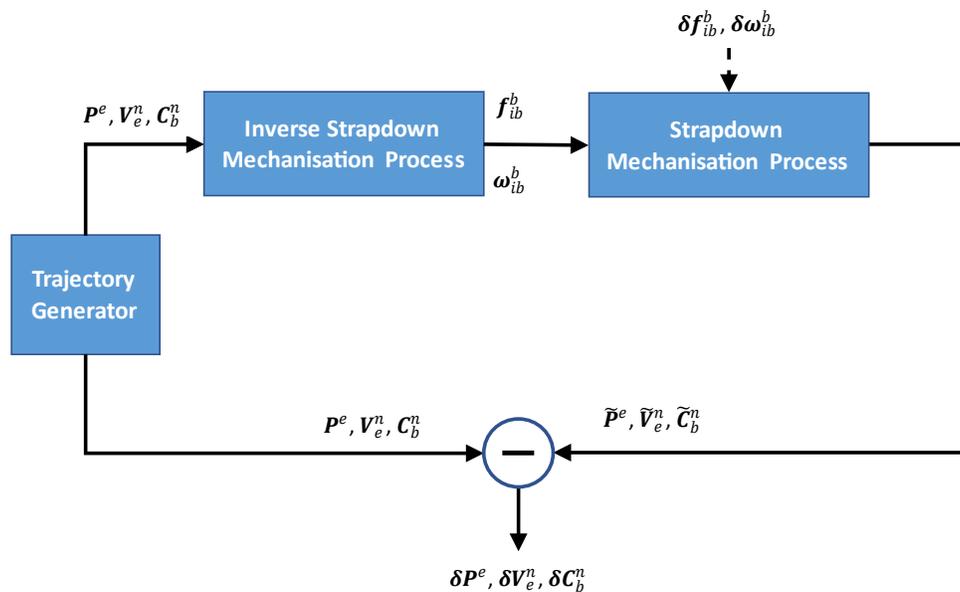


Figure 1: Closed-Loop IMU Simulation with Input-Output validation, adapted from [3]

In this contribution only the IMU-Observation generator (left part in Figure 1), i.e. the inverse strapdown mechanization process, will be used to generate IMU observations with different error parameters in order to be integrated with the GNSS-Observations using the MuSNAT-RTK-IMU-KF. Similarly, the same input reference trajectory can be used from the MuSNAT-Transceiver to simulate along the giving path the appropriate IF-Samples. Consequently, having the same trajectory and therefore the same time source guarantees a perfect absolute time stamping (UTC) between all fused observations. Therefore, only the Simulator part that deals with IMU-Observation will be described in the next subsection.

Generation of the IMU Observations

The specific force of an IMU can be expressed by the following general equation

$$\mathbf{f}_{ib}^b = \mathbf{C}_n^b [\dot{\mathbf{v}}_{eb}^n + (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{v}_{eb}^n - \mathbf{g}^n] + \delta \mathbf{f}_{ib}^b \quad (1)$$

Where $\dot{\mathbf{v}}_{eb}^n$ is the change of velocity expressed in the navigation frame, the second term represent the Coriolis term that includes both earth rotation rate with respect to the inertial frame (2) and the angular rate of the navigation frame with respect to the Earth frame, also called transport rates (3). The last term within the brackets is local gravity vector of the earth (4) which can be computed using e.g. the Somigliana formula [4] and usually expressed as a 3d-vector with only one component in the z-direction. The transition from n-frame to the body-frame, where usually both acceleration and rotation sensing take place, a rotation matrix, \mathbf{C}_n^b can be employed. The last term $\delta \mathbf{f}_{ib}^b$ on the right side is the error model related to the specific force that contains both deterministic and stochastic errors.

$$\boldsymbol{\omega}_{ie}^n = \mathbf{C}_e^n \boldsymbol{\omega}_{ie}^e = [\omega_e \cos \varphi \quad 0 \quad -\omega_e \sin \varphi]^T \quad (2)$$

$$\boldsymbol{\omega}_{en}^n = [\dot{\lambda} \cos \varphi \quad -\dot{\varphi} \quad -\dot{\lambda} \cos \varphi]^T = \left[\frac{v_E}{R_n + h} \quad -\frac{v_N}{R_m + h} \quad -\frac{v_E \tan \varphi}{R_n + h} \right]^T \quad (3)$$

$$\mathbf{g}^n = [0 \quad 0 \quad -g]^T \quad (4)$$

As depicted in Figure 1, these observations can be generated by selecting a user defined trajectory. This includes the information about the position in geodetic coordinates (φ, λ, h) , the velocity along the east, north, and down directions (v_N, v_E, v_D) , and the attitude angles as roll, pitch and yaw (ϕ, θ, ψ) .

The generation procedure of the reference trajectory is first based on a set of segments with user defined dynamics, such as straight line, accelerations, decelerations, small turns, U-turn, constant velocities, static periods as well as varying attitude, and their combinations. To control these shapes the forward acceleration in the b-frame, $\mathbf{a}^b = [a_x^b \quad 0 \quad 0]^T$, as well as the attitude angle rates $(\dot{\phi}, \dot{\theta}, \dot{\psi})$. To obtain the full trajectory, inertial integration parameter, such as initial position $\varphi_0, \lambda_0, h_0$ as well as initial orientation ϕ_0, θ_0, ψ_0 are mandatory. On the other hand, the velocity in the b-frame should be propagated over time through the equation of motion $\mathbf{v} = \mathbf{v}_0 + \mathbf{a}\Delta t$. This information allows to compute the velocity and it's first derivative as indicated in (5) and (6)

$$\mathbf{v}_{eb}^n = \begin{bmatrix} v_N \\ v_E \\ v_D \end{bmatrix} = \mathbf{C}_b^n \mathbf{v}_{eb}^b = \mathbf{C}_b^n \begin{bmatrix} v_x^b \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} v_x^b \cos \theta \cos \psi \\ v_x^b \cos \theta \sin \psi \\ -v_x^b \sin \theta \end{bmatrix} \quad (5)$$

Taking advantage of the so-called non-holonomic constraints (NHC) [5] where the physical assumption is made for a vehicle which doesn't jump off the ground or slides on the ground and therefore the velocity of the vehicle in the plane perpendicular to the forward direction is almost zero. Consequently, only the forward velocity v_x^b is necessary to calculate \mathbf{v}_{eb}^n and thus reducing v_y^b, v_z^b to zero. The change of velocity can be obtained from the first derivative $\dot{\mathbf{v}}_{eb}^n$ with respect to time.

$$\dot{\mathbf{v}}_{eb}^n = \frac{d\mathbf{v}_{eb}^n}{dt} = \frac{d}{dt} \begin{bmatrix} v_x^b \cos \theta \cos \psi \\ v_x^b \cos \theta \sin \psi \\ -v_x^b \sin \theta \end{bmatrix} = \begin{bmatrix} \dot{v}_x^b \cos \theta \cos \psi - v_x^b \dot{\theta} \sin \theta \cos \psi - v_x^b \dot{\psi} \cos \theta \sin \psi \\ \dot{v}_x^b \cos \theta \sin \psi - v_x^b \dot{\theta} \sin \theta \sin \psi + v_x^b \dot{\psi} \cos \theta \cos \psi \\ -\dot{v}_x^b \sin \theta - v_x^b \dot{\theta} \cos \theta \end{bmatrix} \quad (6)$$

Where \dot{v}_x^b denotes the acceleration in the b-frame ($\dot{v}_x^b = a$).

To obtain the position vector \mathbf{P}^e in the earth-fixed-earth-centered frame (e-frame) as $\mathbf{P}^e = [\varphi \ \lambda \ h]^T$, the velocity in the north, east and down direction is needed, which can be derived from (5). These parameters can be expressed in term the rate of change of the position, therefore

$$\dot{\varphi} = \frac{v_N}{R_m + h}, \dot{\lambda} = \frac{v_E}{(R_n + h)\cos\varphi}, \dot{h} = -v_D \quad (7)$$

Assuming, a constant sampling interval Δt ($= t_{k+1} - t_k$) using for instance, the trapezoidal rule of numerical integration [4] the position vector can be computed as follows

$$\begin{aligned} \varphi_{k+1} &= \varphi_k + \frac{1}{2} \frac{v_{N,k} + v_{N,k+1}}{R_m + h_k} \Delta t \\ \lambda_{k+1} &= \lambda_k + \frac{1}{2} \frac{v_{E,k} + v_{E,k+1}}{(R_n + h_k)\cos\varphi_k} \Delta t \\ h_{k+1} &= h_k - \frac{1}{2} (v_{D,k} + v_{D,k+1}) \Delta t \end{aligned} \quad (8)$$

Since the rotation between coordinate frames can be expressed as a sum of rotations, the output of the gyroscope triad can be written as

$$\boldsymbol{\omega}_{ib}^b = \mathbf{C}_n^b(\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) + \boldsymbol{\omega}_{nb}^b + \delta\boldsymbol{\omega}_{ib}^b \quad (9)$$

This equation includes, in addition to both earth rotation and transport rates, the impact of the body rates, i.e. $\boldsymbol{\omega}_{nb}^b$ which is related to the attitude angle rates, $\dot{\phi}, \dot{\theta}, \dot{\psi}$, by the following equation [4]:

$$\boldsymbol{\omega}_{nb}^b = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{C}_1(\phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{C}_1(\phi)\mathbf{C}_2(\theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (10)$$

similar to the accelerometer observation model in (1), the mathematical model of the gyroscope observation is also affected by errors that can be generally described by $\delta\boldsymbol{\omega}_{ib}^b$

MUSNAT TRANSCEIVER

A software receiver has usually a satellite signal tracking unit which uses code, frequency and phase tracking loops to estimate signal time delay τ , signal doppler shift f_d and the carrier phase Φ respectively the pseudorange ρ with its derivatives $\dot{\rho}$ and $\ddot{\rho}$. These parameters are subsequently forwarded to the navigation processor unit where the tracking parameter of all satellites are used to estimate the receiver position, velocity and time (PVT) and that's it.

The MuSNAT Transceiver concept, however, is based on the well-known software receiver vector tracking mode [6]. A software receiver in vector tracking mode has an additional feedback loop between the navigation processor unit and the satellite signal tracking loops, illustrated in Figure 2 (left). In the feedback loop propagate the currently estimated PVT into the future and uses this information in combination with the known satellite ephemeris to estimate the upcoming line-of-side (LOS) signal conditions (ρ , $\dot{\rho}$ and $\ddot{\rho}$). These future signal properties can be used to steer the numerical controlled oscillator (NCO) which governs the replicate signal generation inside the tracking loops.

To transform the software receiver with vector tracking into a GNSS signal generator two main modifications are necessary. First, the PVT feedback loop is replaced by the injection of a predefined user trajectory containing the desired PVT information for which the GNSS signals shall be generated. Secondly, the signal replicas inside the tracking loops are multiplied by a predefined amplitude and a corresponding navigation symbol and subsequently all complete satellite signals are added and saved as output GNSS signal. The modifications are shown in Figure 2 (right). A detailed description of the conversion of the MuSNAT software receiver to a software transceiver with the option to generate GNSS signals on intermediate frequency or in baseband are given in [2] with a proof of concept by generating Galileo OS E1 signals with navigation message authentication (OSNMA). An accuracy assessment for the archived code generation accuracy and the achievable positioning accuracy with the theoretical limits was done

in [7]. The results show that the code tracking performance reaches close to the theoretical limit and the positioning is bias free and the positioning variance is also close to the theoretical limits. The validation of the bias free phase generation and if the phase consistency among the generated satellite signals is adequate for phase-based positioning and RTK is subject of this work.

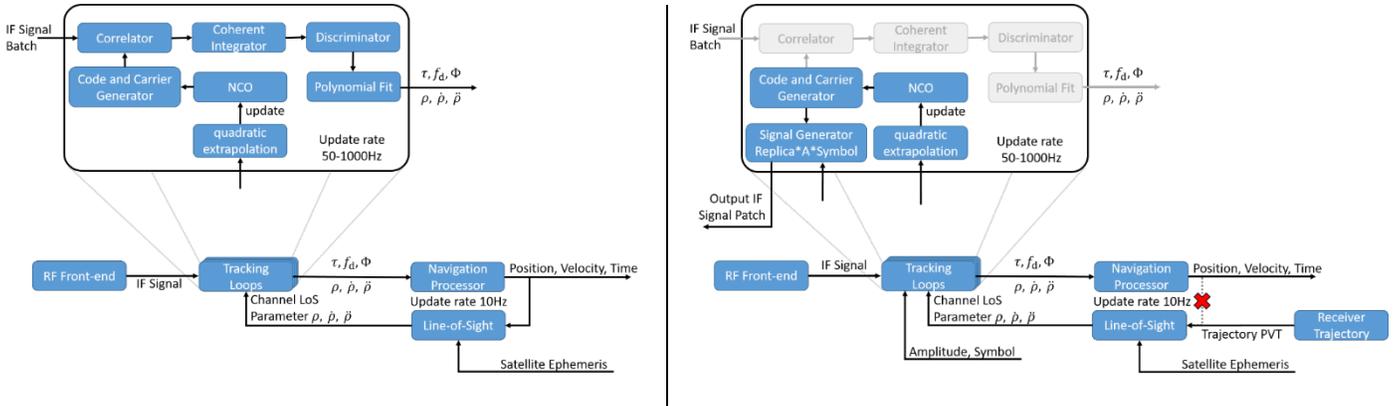


Figure 2: Flowchart of the basic concept of the GNSS vector tracking (left) and the adaptations for the signal generation (right). [7]

GNSS/INS NAVIGATION ENGINE

The MuSNAT, among others, provides a loosely-coupled GNSS/INS error-state-EKF estimating 15 states, namely the three components of position, velocity, attitude, gyroscope bias and acceleration bias. It is designed to incorporate both GNSS and LiDAR measurement updates, from which only GNSS, respectively RTK-fixed, solutions are accepted in this very scenario to ensure a high quality of measurement updates and to use RTK accuracy of a few cm within the measurement accuracy matrix of the KF. Therefore, false fixes and acceptance of erroneous data becomes visible immediately upon usage in the filter. The update validation steps consist of variance-covariance and propagation residual checks in addition to the GNSS positioning status query. The processing flow of the KF is depicted in Figure 3 and for a more detailed description and performance assessment using real data under harsh conditions the reader is referred to [8]. For RTK-processing, RTKLib version 2.4.3 b29 is accessed through an interface in the MuSNAT.

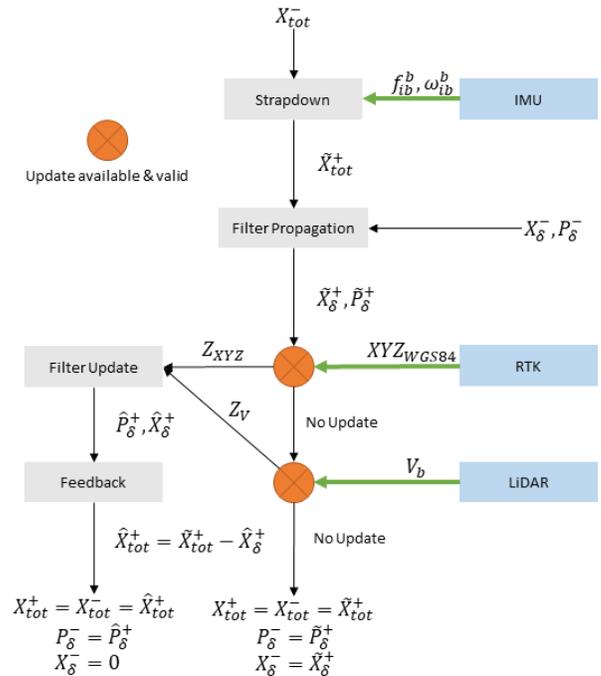


Figure 3: GNSS/INS EKF Flowchart

COMBINATION OF GNSS AND IMU SIMULATION

The very beginning of the simulation lies within the IMU simulator. A set of segments is defined, each with its specific dynamic (i.e. velocity, acceleration/deceleration, turn rate) and the duration for which this dynamic state is held. This creates an error-free dead-reckoning trajectory which is then connected to a starting point in a given coordinate frame, in our case WGS84 as it has to be combined with GNSS measurements. This trajectory is then fed into the MuSNAT transceiver as user trajectory to generate the IF-sample data. As a prerequisite though, in order to be used for RTK in the MuSNAT, the simulated GNSS data has to be available in RINEX format. This has two reasons: Base station data processing for RTK within the MuSNAT is currently only supported in two modes: Real-time RTCM or RINEX post-processing. Although rover data can be processed directly from IF-

sample data, it is quite time-extensive to post-process IF-samples compared to RINEX files, especially when experimenting with different modes or settings in the positioning, which require multiple iterative software executions. Also, there comes no downside with using RINEX files regarding the verification of the tracking features of the simulated signal, as shown by the RINEX file generation process (Figure 5). Since RINEX files are generated in the MuSNAT by tracking the simulated IF-samples and logging the observations, the acquisition and tracking are validated implicitly by storing correct RINEX data. So, the described interactions between the IMU Simulator and the MuSNAT Transceiver add up to a completely new software package, which we baptized SIMSS (Satellite and Inertial Measurement Simulation System).

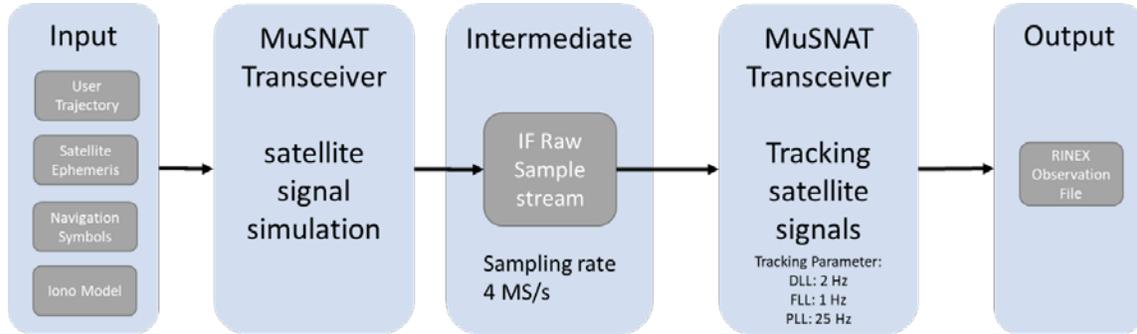


Figure 4: MuSNAT IF-sample to RINEX conversion pipeline

SIMULATION AND SENSOR FUSION VERIFICATION SETUP & RESULTS

Figure 5 shows the high-level flowchart of the SIMSS verification chain. To validate the SIMSS as well as the sensor fusion algorithm, the experiment was specifically designed for precision evaluation. Figure 6 shows the synthetic trajectory and the skyplot of the artificial satellites in view. No map information was considered in generating this trajectory, its design is driven to achieve the best possible observability for each difficult to observe state, i.e. inertial biases and heading. It therefore consists of multiple 8-shapes providing accelerations in vehicle x- and y-direction, observability in Z (respective down) direction is given through the gravitational acceleration. The trajectory starts with a stationary phase of one minute before they dynamic part. It has to be noted that the GNSS/INS engine uses a kinematic alignment, meaning that no static leveling and heading determination are performed during the stationary beginning. Finally, the virtual base station is located at the origin of the user trajectory.

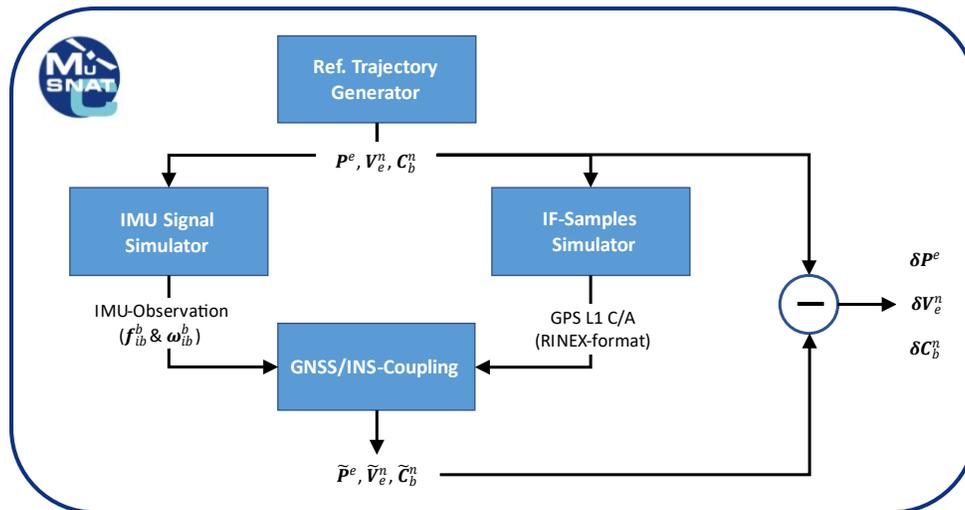


Figure 5: MuSNAT SIMSS

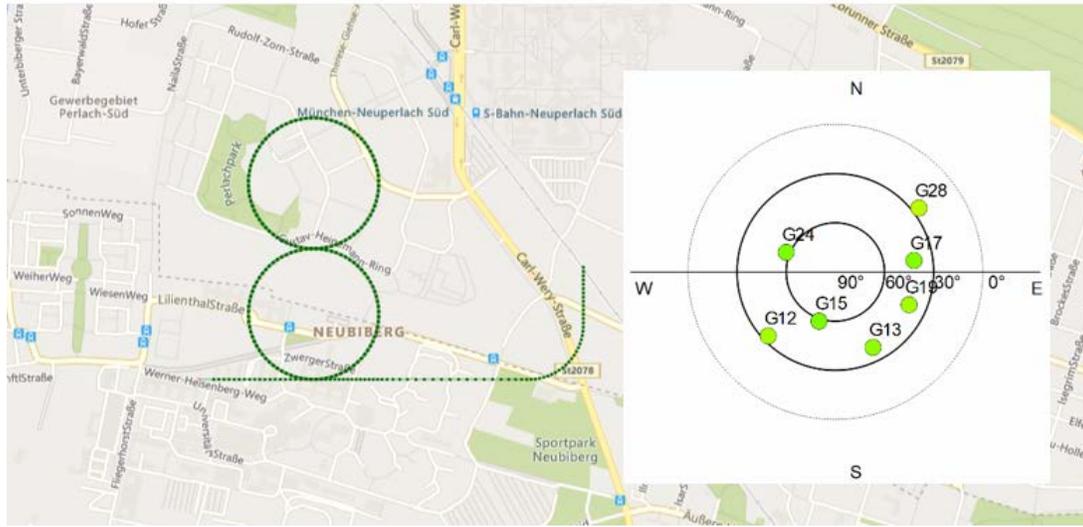


Figure 4: Simulated trajectory (left) and skyplot of the artificial satellites in view (right)

With respect to sensor measurements, a reduced and simple set of GNSS observations (namely GPS L1 only) with no multipath and low noise, in combination with extremely small biases and noise on the inertial unit was decided upon. This way, potential errors in the GNSS signal simulation and inertial data simulation cannot be disguised. Additionally, the GNSS signal has to be precise and accurate enough to enable the sensor fusion algorithm to recover the inertial biases. Tables 1 - 4 show the key settings of the simulations and processing. In the IMU simulation part (see Tab. 1), all three axes are assumed to be affected by the same error sources.

In order to avoid filter updates with reduced accuracy from float solutions, only RTK-fixed solutions are accepted as KF updates. This way again, potential errors in the simulation are discovered by the RTKLib being unable to fix the ambiguities, and with the stringent fixed position accuracy, the IMU is forced on the GNSS trajectory. Upon the biases converging to their initially simulated value, it is proven that the simulations and sensor fusion are, indeed, correct.

Table 1: Simulation parameters used by the IMU signal simulator

IMU Simulation Parameter	Value	Unit
Sampling rates	200	Hz
GPS week number	2083	-
Duration	15	minutes
Gyro bias error, all axis	1	deg/h
Gyro ARW (white noise), all axis	0.15	deg/sqrt(h)
Acc bias error, all axis	100	μg
Acc VRW (white noise) , all axis	50	$\mu\text{g}/\text{sqrt}(\text{Hz})$

Table 2: Simulation parameters used by the GNSS signal simulation

GNSS Simulation Parameter for IF sample generation	Value	Unit
GPS Satellites	12, 13, 15, 17, 19, 24, 28	PRN
Simulated C/N0	40, 42, 44, 43, 40, 44, 38	dB-Hz
Noise	Adaptive White Gaussian Noise	-
Signal	GPS L1 C/A	-
Sampling rate	4M	S/s

Sample format	Real 8bit	Bit
Measurement rate (update rate)	10	Hz

Table 3: MuSNAT GPS tracking parameters

GPS tracking parameters for RINEX generation	Value	Unit
Coherent integration time	20	ms
DLL bandwidth	2	Hz
DLL filter order	2	-
FLL bandwidth	1	Hz
FLL filter order	1	-
PLL bandwidth	25	Hz
PLL filter order	2	-
Measurement rate	1	s
Navigation module	SPP (least-squares)	-

Table 4: MuSNAT GNSS/INS processing parameters

GNSS/INS Processing Parameter	Value	Unit
Elevation mask	15	deg
Ambiguity Fixing threshold	3	-
Gyroscope process noise	0.001	(deg/sec)/sqrt(Hz)
Accelerometer process noise	50	$\mu\text{g}/\text{sqrt}(\text{Hz})$
GNSS measurement accuracy horizontal	0.03	m
GNSS measurement accuracy vertical	0.05	m

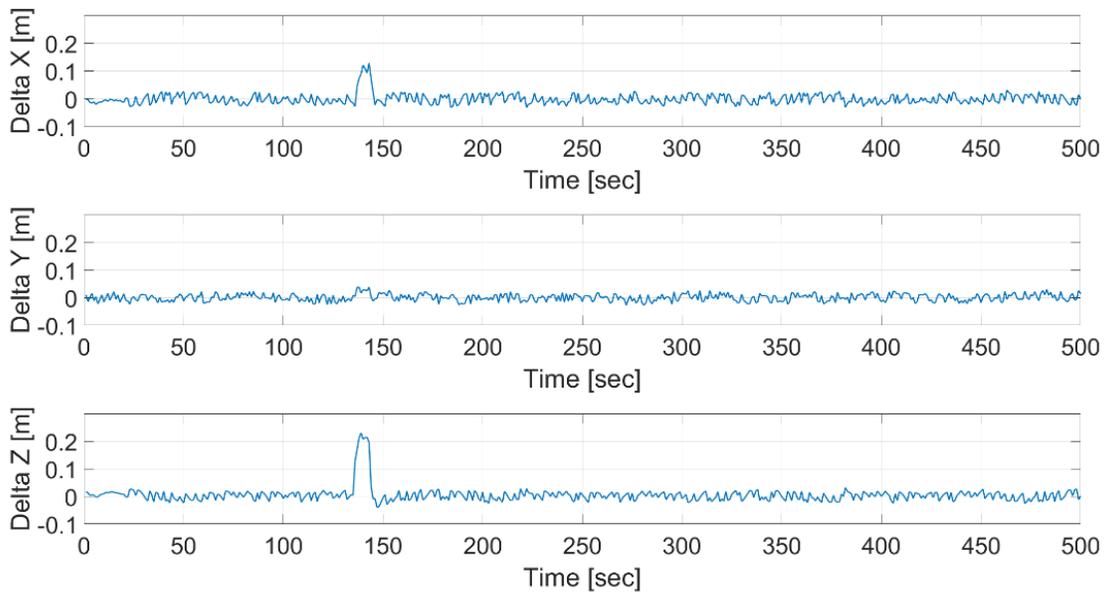


Figure 5: Difference between reference trajectory and EKF solution

Figure 7 depicts the difference between the ground truth and estimated trajectory, ground truth being the output from the reference trajectory generator previously introduced. Apart from few epochs around second 140 the difference lies within the 3 centimeter level (as used as GNSS measurement accuracy within the GNSS/INS filter) with zero mean. The offset at second 140 appears to be

a false fix by the RTKLib, as it mainly shows in X and Z (which are highly correlated at central European longitude) and has a magnitude of approximately one GPS L1 wavelength.

Figure 8 displays the ambiguity resolution ratio output by the RTKLib. After a convergence phase of approximately one minute, consisting mainly of float (orange) and SPP (red) solutions, RTKLib is able to continuously fix the phase ambiguities apart from one epoch, with ratios beyond 10 for the vast majority of the trajectory (this is considered an open sky scenario). The single float epoch, however, coincides with the recovery of the actual trajectory after several false fixes as seen in the estimated trajectory versus ground truth. Due to the alignment method in the EKF, namely the kinematic alignment, filter estimation commences after several continuous RTK-fixed epochs exceeding a certain velocity threshold. The SPP and float solutions in the visible in the first 100 epochs in Figure 8 occur during the initial static phase and therefore do not affect the filter solution (since it is not yet initialized at this point).

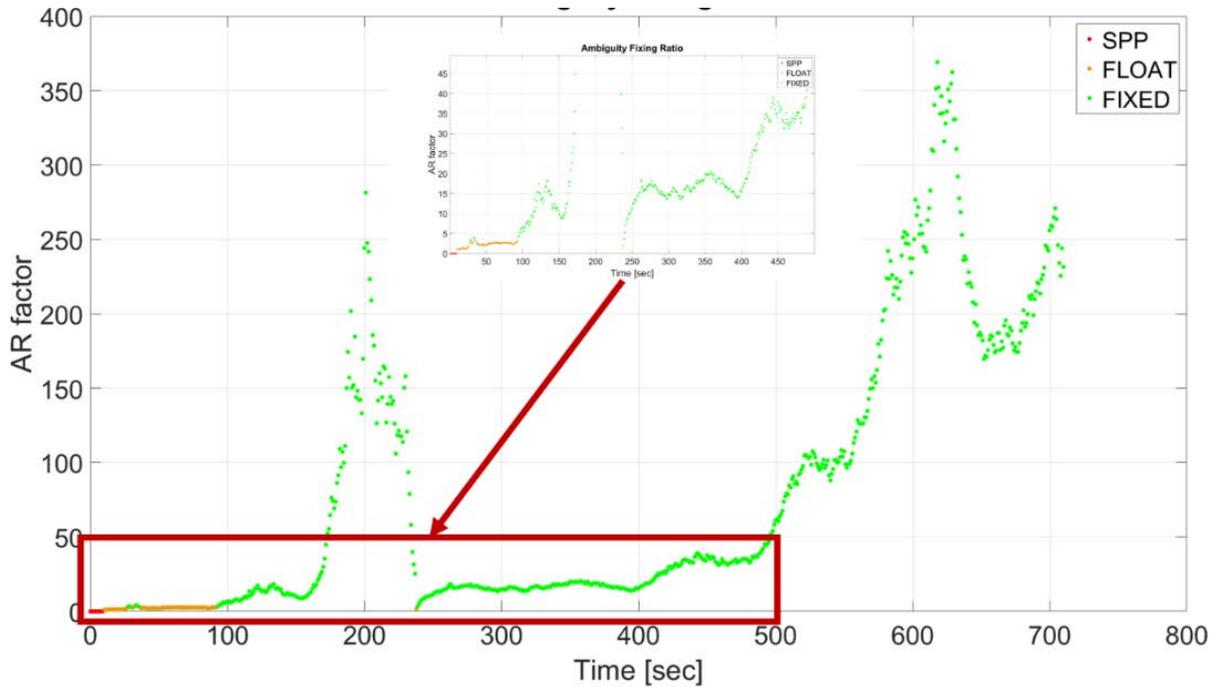


Figure 6: Ambiguity resolution ratio from RTKLib

As main quality criterion we select the convergence of the estimated biases of the gyroscope and accelerometer as depicted in Figure 9 and 10, where in both graphics the line in light green states the actual bias that was added to the measurements. With the initial convergence behavior looking promising, the false fixes around epoch 140 (referring to Figures 9 and 10), as presumed, cause the bias estimates to diverge briefly. Hence the filter is able to recover and converge to the actual values in the end. The accelerometer bias in X does not fully converge until epoch 540. This is due to the constant velocity in X during the “8” shape and the therefore poor observability of this state. After leaving the “8”, the vehicle accelerates and the filter is ultimately able to estimate the bias correctly. This was not possible prior, since the acceleration in X took place at the very beginning of the trajectory and convergence was interrupted at epoch 140.

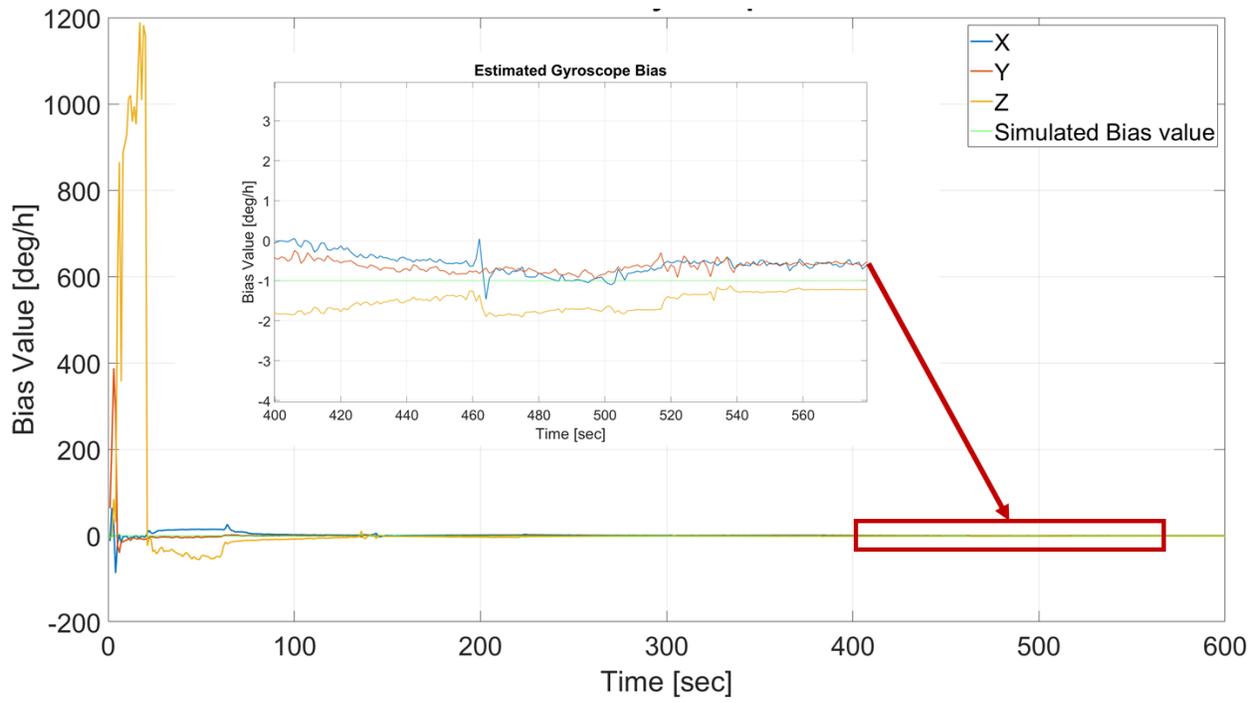


Figure 7: EKF estimated gyroscope biases

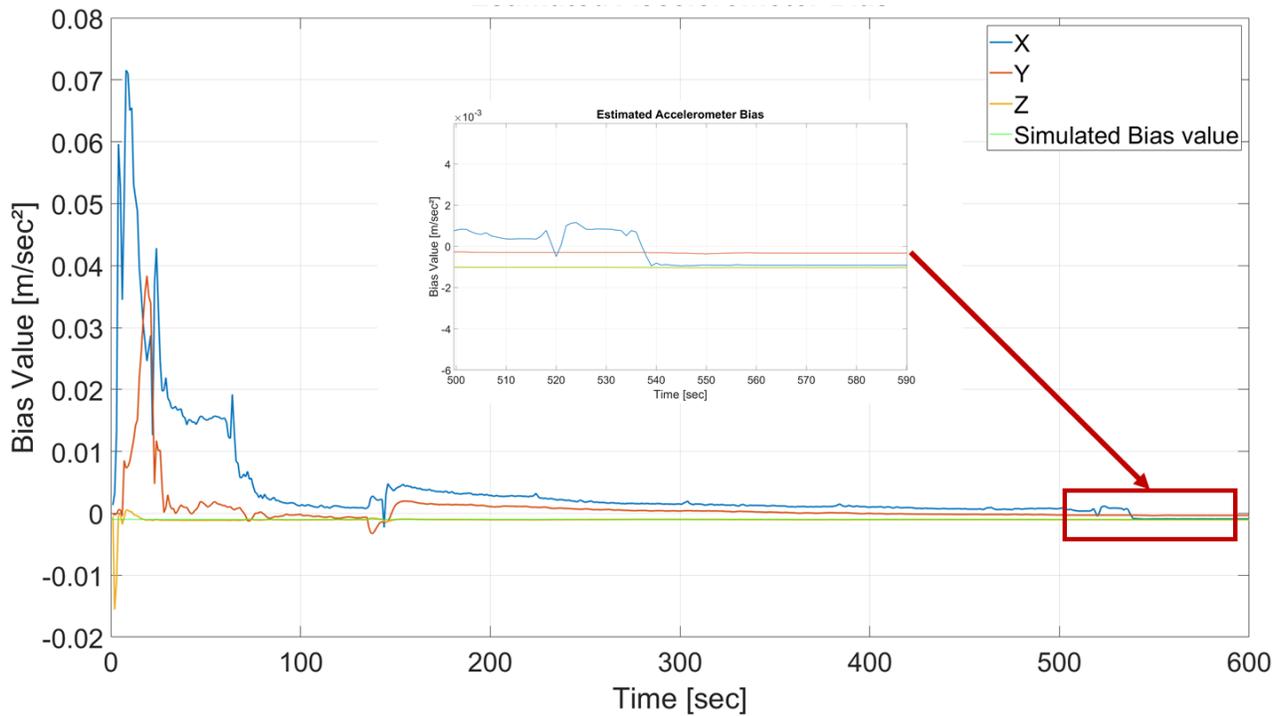


Figure 8: EKF estimated accelerometer biases

CONCLUSION & OUTLOOK

In this publication we have successfully demonstrated the ISTA GNSS Transceiver's capability to synthesize RTK-processable phase data at IF and RINEX level, and its compatibility with the Closed-Loop INS simulator. Upon combining both these tools, the ISTA SIMSS was established and its produced data was verified by processing with the MuSNAT-internal RTK-INS loosely coupled sensor fusion. At this point, simulations options regarding GNSS are limited to different signal and GNSS selections under open sky conditions. However, multipath and occlusion are under consideration. Another future endeavor is the potential integration of simulated LiDAR data, which implicitly includes models of the surroundings and the environment. This would state a feasible way of introducing information for the GNSS simulation regarding obstructions and NLOS signals and is currently under evaluation. Overall, considering the Closed-Loop INS Simulator's capability of providing IMU data of any sensor grade, the ISTA SIMSS is underway to become a valuable, multi-purpose dataset generation engine, with a wide application range especially within the scope of autonomous vehicle navigation and smartphone-based navigation.

ACKNOWLEDGMENTS

The results presented in this work were developed within the projects "OPTimal Assistierte, hoch Automatisierte, Autonome und kooperative Fahrzeugnavigation und Lokalisation (short: OPA3L, grant no. 50NA1910)" and "Integration of authentication features in open service (OS) satellite navigation signals (short: SatNavAuth, grant no. 50NA1703)" funded by the German Federal Ministry for Economic Affairs and Energy (BMWi) and administered by the Project Management Agency for Aeronautics Research of the German Space Agency (DLR) in Bonn, Germany.

REFERENCES

1. Pany, T., Dötterböck, D., Gomez-Martinez, H., Hamed, M. Subhan, Hörkner, F., Kraus, T., Maier, D., Sanchez-Morales, D., Schütz, A., Klima, P., Ebert, D., "The Multi-Sensor Navigation Analysis Tool (MuSNAT) – Architecture, LiDAR, GPU/CPU GNSS Signal Processing," *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019)*, Miami, Florida, September 2019, pp. 4087-4115. <https://doi.org/10.33012/2019.17128>
2. Maier, D. S., Frankl, K. and Pany, T., "The GNSS-Transceiver: Using Vector-tracking Approach to Convert a GNSS Receiver to a Simulator; Implementation and Verification for Signal Authentication," *Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018)*, Miami, Florida, September 2018, pp. 4231-4244. <https://doi.org/10.33012/2018.16083>
3. Bochkati, M., Schön, S., Schlippert, D., Schubert, C., Rasel, E., "Could cold atom interferometry sensors be the future inertial sensors? -First simulation results", *In Proceedings of the Inertial Sensors and Systems (ISS)*, Karlsruhe, Germany, 19–20 September 2017; pp. 1–20.
4. Jekeli, C., "Inertial Navigation Systems with Geodetic Applications", *Walter de Gruyter GmbH & Co.KG*, 2001.
5. El-Sheimy, N., "The Potential of Partial IMUs for land Vehicle Navigation", *InsideGNSS, Issue Spring 2008*, pp. 16-25. URL: <https://insidegnss.com/wp-content/uploads/2018/01/spring08-016-025.pdf>
6. Pany, T., Bernd E., "Use of a vector delay lock loop receiver for GNSS signal power analysis in bad signal conditions," *Position, Location, And Navigation Symposium, 2006 IEEE/ION*. IEEE, 2006.
7. Maier, D. S., Pany, T., "GNSS Transceiver: Simulation Accuracy Assessment of Using a Vector-Tracking Receiver as RF Constellation Simulator," *InsideGNSS, Issue September/October 2019*, pp. 28-37. URL: <https://insidegnss.com/gnss-transceiver-simulation-accuracy-assessment-of-using-a-vector-tracking-receiver-as-rf-constellation-simulator/>
8. A. Schütz, D. E. Sánchez-Morales and T. Pany, "Precise positioning through a loosely-coupled sensor fusion of GNSS-RTK, INS and LiDAR for autonomous driving," *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Portland, OR, USA, 2020, pp. 219-225, doi: 10.1109/PLANS46316.2020.9109934.