OST
Eastern Switzerland
University of Applied Sciences

Master Project 1

Integration of a GNSS RTK Module into a ROS2 Based Mobile Robot Platform

Autumn Semester 2025

Version: 1.0

Institute:	ISF Institute for Intelligent Systems and Smart Farming	**Student:**	Chris Stolz chris.stolz@ost.ch
Examinator:	Prof. Dr. Dejan Šeatović dejan.seatovic@ost.ch	**Matriculation Nr.:**	17-179-359
Supervisor:	Luis Meier luis.meier@ost.ch	**Degree Program:**	MSE Mechatronics & Automation

7 January 2026

OST
Eastern Switzerland
University of Applied Sciences

Contents

Abstract

Definition of Task:
This MSE project investigates the integration and verification of a GNSS Real-Time Kinematic (RTK) positioning system into a ROS2- based mobile robotic platform. The task involves integrating the GNSS RTK module into the Husarion Panther unmanned ground vehicle and incorporating it into the ROS2 software stack. The system is intended to acquire GNSS RTK data along with diagnostic metrics to enable reliable outdoor localisation. The project further aims to evaluate the system performance under realistic operating conditions, focusing on horizontal, vertical, and three-dimensional positioning accuracy, RTK correction service availability, and overall system reliability.

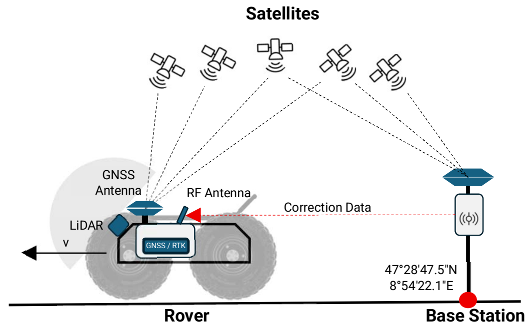


Fig. 1: Visualization of High Level Project Scope

Results:
The GNSS RTK module and LiDAR are mounted on a central bracket, ensuring unobstructed sky visibility for the antenna and a clear LiDAR field of view. The bracket allows the precise antenna positioning and offers the possibility to mount a reflecting prism for verification. The interface hatch houses the Raspberry Pi 5 and the GNSS module, provides IP54-rated protection, functions as a heatsink for the electronics, and supports modular connections for the LiDAR and antennas. This design ensures robust operation, minimizes EMC interference, and requires only two permanent cables connected to the robot system. With the RTK correction service, the system achieved centimeter-level horizontal positioning accuracy (RMSE 7.2 mm to 9.5 mm). Vertical errors were more variable, resulting in three-dimensional RMSE of 31.1 mm to 41.6 mm. RTK availability varied from 58.2 % to 96.2 %, directly affecting positioning stability and continuity.

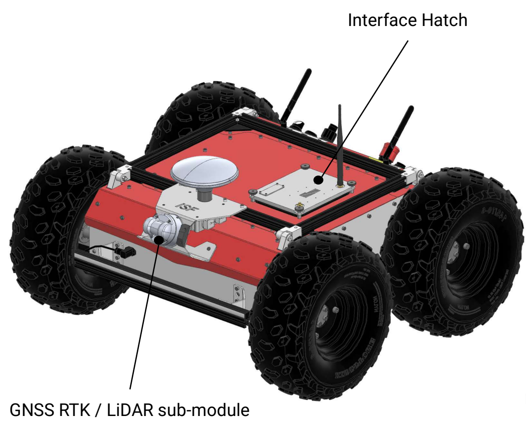


Fig. 2: UGV with integrated GNSS RTK / LiDAR sub-module

Conclusion:
The results confirm that GNSS RTK systems can provide high-accuracy localization suitable for mobile robotic applications when correction data is reliably available. However, the observed sensitivity to correction service continuity, satellite geometry, and evaluation methodology highlights the need for robust data validation and handling strategies within the robotic localization stack. Continuous monitoring of diagnostic metrics, such as RTK fix status and DOP values, is essential to ensure safe and reliable operation. This work provides a validated basis for integrating GNSS RTK into mobile robotic platforms and identifies key aspects for improving robustness and reliability in real-world deployments.

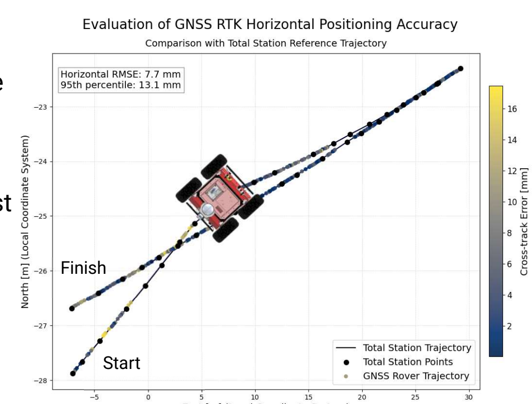


Fig. 3: Horizontal Position Accuracy Evaluation

Introduction

Problem Statement

Depth cameras, RGB cameras, IMU, and LiDAR sensors offer relative positioning through dead reckoning and SLAM (Simultaneous Localization and Mapping), but they suffer from accumulated drift and error over time due to a missing ground truth measurement. Conventional SLAM systems rely on distinctive landmarks, such as buildings or walls, for accurate localization, which can pose significant challenges in wide open areas due to their limited measurement range. This is particularly problematic in autonomous agricultural robotics applications with wide open areas such as cornfields.

SLAM-based systems also face a range of practical limitations. These include significant computational and memory demands, limited measurement range, sensor sensitivity to dust or contamination, calibration challenges, and vulnerability to shocks and vibrations. However, the issue of high computational load has been greatly mitigated by recent advances in highly efficient Graphics Processing Units (GPUs), which enable faster (parallel) data processing and improved system performance. LiDAR and depth cameras equipped with infrared projectors, which are designed to improve low-light performance, often require active cooling when operating in hot environments.

For near field mapping (NFM) and global mapping and localization in unstructured environments the SLAM system is indispensable and therefore widely adopted in the field of autonomous robotics. A specialized sensor fusion algorithm is used to combine all sensor data into a single map and pose estimation, which is then used to localize and navigate the robot within its environment. Near-field mapping data can be fed into an obstacle avoidance algorithm to successfully avoid obstacles by recalculating an optimized trajectory around them. In Fig. 4 you can see a typical architecture which utilizes depth cameras and a LiDAR for near field mapping and mapping of the environment. The individual sensors are placed to maximize the field of view (FoV).

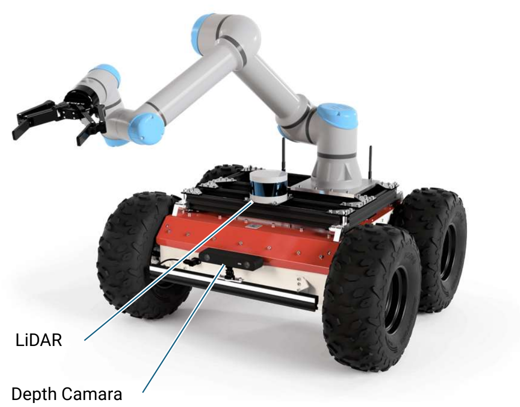


Fig. 4: Husarion Panther UGV with manipulation arm [8]

Integrating a GNSS RTK (Global Navigation Satellite System Real-Time Kinematics) module as a “ground truth” source, allows for continuous correction of absolute positioning, significantly reducing drift and improving long-term localization stability of a SLAM system. By providing centimeter-level accuracy, GNSS RTK data serves as a robust anchor for fusing information from RGB cameras, depth cameras, IMUs, and LiDAR. This enhances the precision of mapping, obstacle detection, and trajectory estimation. With this high-accuracy reference, SLAM map optimization becomes more straightforward, computational demands are lowered, and the robot’s global position can be determined with greater confidence. When combined with conventional SLAM systems, GNSS RTK addresses their inherent limitations, delivering an extra layer of reliability, precision and accuracy for localization and mapping. This synergy is especially valuable in critical applications with wide open areas, where precise and accurate localization is essential.

Objectives

The main objective of this project is to integrate a GNSS RTK module into a ROS2-based robotics system to enable precise positioning in a semi-structured outdoor environment. To achieve this, the following specific objectives are defined:

- Achieve reliable communication between the GNSS RTK module and ROS2
- Implement ROS2 nodes into an existing ROS2 software stack which can process and publish GNSS RTK positioning data
- Validate the integrated GNSS RTK module in terms of position accuracy and reliability
- Provide documentation and reusable software components

Project Scope

The scope of this project focuses on the integration and validation of a GNSS RTK module within a ROS2 based robotic platform for outdoor localization. Specifically, the project covers the following scope:

- Integration and configuration of a GNSS RTK module into the ROS2 robotics system
- Design and implementation of a compact sub-module to house the GNSS antenna, GNSS receiver and a LiDAR sensor
- Validation of system performance in a semi-structured outdoor environment
- Optional hardware and software integration of the LiDAR sensor

Out of Scope:

- Advanced multi-sensor fusion
- Development of high-level navigation or path-planning algorithms

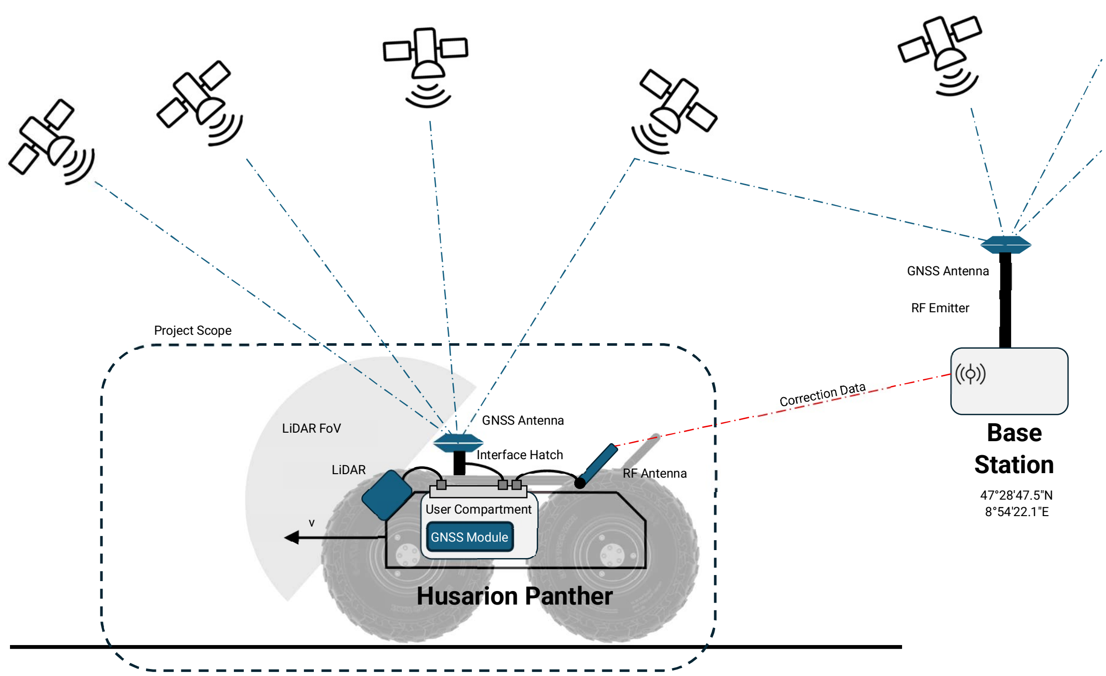


Fig. 5. Visualization of High Level Project Scope

Documentation Structure

The documentation for this project is organized to ensure clarity and traceability of all aspects of this work. The three main components of the project documentation are this document, a code repository on GitLab, and a supplementary appendix folder.

The structure is designed such that the three components complement each other:

- The main documentation presents the conceptual and methodological framework, detailing scope, system design, implementation, testing, and results
- The GitLab repository provides the implementation and codebase, including all nodes, configuration files, and scripts developed during the project. A readme.md file is available in the repository which proves further details about the repository.
- The appendix folder contains supplementary materials, such as datasheets, manuals, EBOM, and raw data, supporting the validation of the results presented in this document

Cross-references throughout this document guide the reader to the relevant code or supplementary material as needed.

Background and Literature Review

Global Navigation Satellite Systems (GNSS)

GNSS, or Global Navigation Satellite System, includes a range of satellite-based positioning systems, such as GPS (United States), GLONASS (Russia), Galileo (Europe), BeiDou (China), as well as regional solutions like QZSS (Japan).

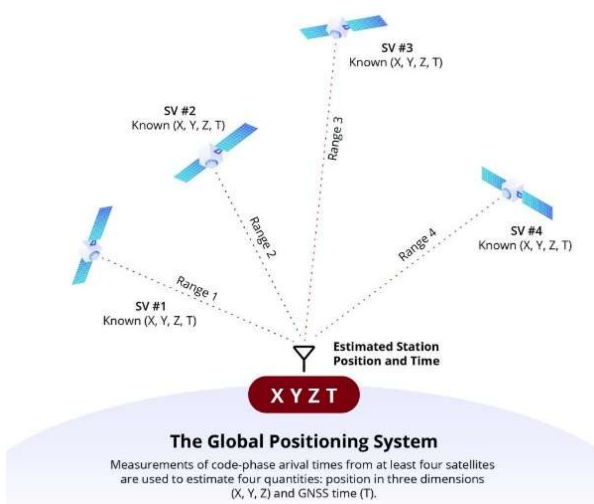


Fig. 6. GNSS Positioning Concept [1]

These systems enable users to determine precise location, velocity, and time anywhere on earth by receiving signals transmitted from satellites in orbit. Each satellite sends out uniquely coded, time-stamped signals, which are captured and interpreted by GNSS receivers. Through trilateration, the receiver calculates its position by measuring the time it takes for each signal to travel from the satellite to the receiver's antenna. Since satellite clocks are precisely synchronized, the receiver can accurately estimate the signal's travel time. Multiplying this time by the speed of light gives a distance measurement in meters from each satellite to the receiver, allowing for highly accurate geospatial positioning. [1]

Using timestamps and satellite coordinates (X, Y, Z) from multiple satellites, a GNSS receiver measures signal time delays to determine its location. For positioning, a minimum of four satellites is required. Typically, this provides accuracy within 3 to 5 meters, which is standard for modern smartphones and mid-range navigation devices. Achieving greater accuracy is limited by several technical factors that affect both the satellite signals and the receiver's processing capabilities. These limitations are discussed in the following section. [1]

Errors on GNSS Signals

The following figure visualizes the possible errors on a GNSS signal. Detailed explanation of the individual root causes and possible mitigations methods can be found in the section below.

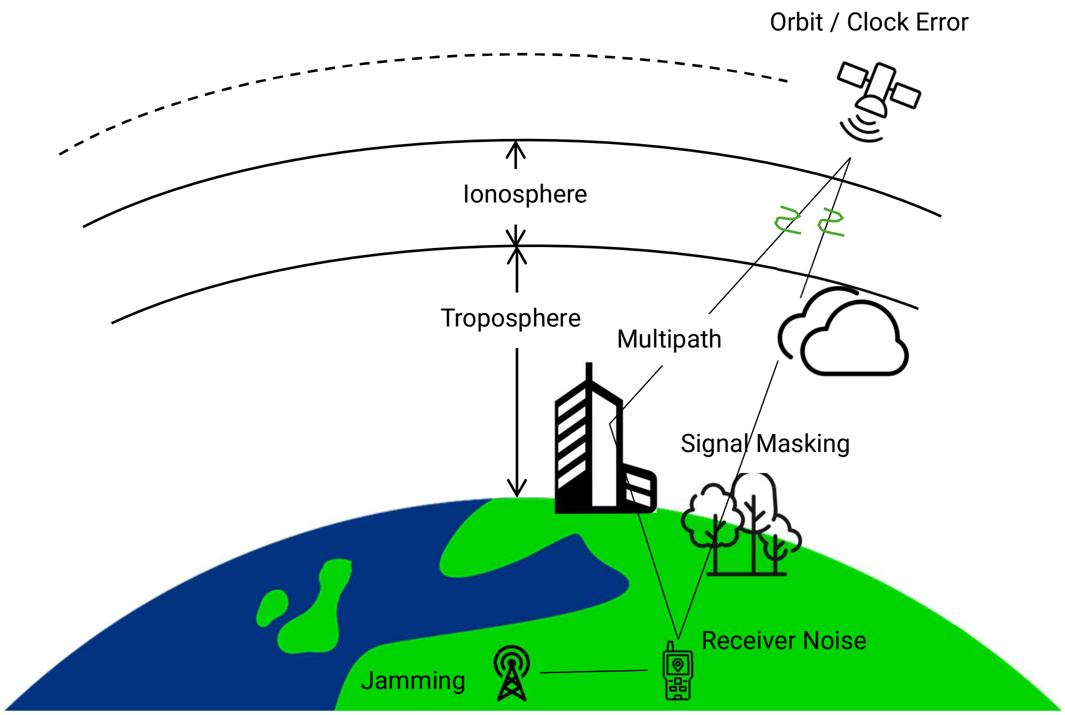


Fig. 7. GNSS Signal Errors (adapted from [2])

Orbit error:	The satellite's orbit can be influenced by various gravitational forces, including those from the earth, moon, and sun, which may cause slight deviations from its predicted trajectory. Another contributing factor to orbit errors is atmospheric friction. Although satellites operate at high altitudes, even minimal atmospheric drag can negatively impact their trajectory over time. Although continual monitoring and regular updates to satellite trajectories help to reduce orbit errors, these inaccuracies can never be fully eliminated. [2]
Clock error:	Each GNSS satellite uses multiple atomic clocks that, despite their stability, can experience minor drifts or biases over time from environmental factors, aging, and inherent limitations. Even an error of just a few nanoseconds in the satellite's clock can lead to position inaccuracies of several meters, since calculating distance relies on the precise measurement of how long it takes the signal to travel from satellite in orbit to the receiver on earth. [2]

Ionospheric Delay:	The ionospheric highly depends on factors such as geographic location, time of day, and solar activity. The ionosphere acts as a dispersive medium, causing the delay to depend on the frequency of the GNSS signal. Therefore high frequency signals experience less delay than low frequency signal. Dual-frequency GNSS receivers can use this property to measure the ionospheric effect and effectively correct for this source of error. [2]
Tropospheric Delay:	Tropospheric delay affects GNSS signals as they pass through the lowest layer of the atmosphere, causing them to slow down. This delay has two components: the dry delay and the wet delay. The dry delay is mainly caused by air molecules and is relatively stable and predictable. The wet delay is caused by water vapor, which is highly variable and harder to correct.[2] Unlike the ionospheric delay, tropospheric delay is non-dispersive, meaning it affects GPS signals equally at all frequencies (L1, L2, L5). As a result, dual-frequency correction methods that are effective for ionospheric delays do not work for tropospheric delays as a mitigation method. Instead, tropospheric delay is mitigated using empirical or physical models that incorporate atmospheric pressure, temperature, and humidity data. After applying these models, the remaining errors are typically reduced to less than 20 centimeters. [2]
Multipath:	Multipath errors occur when satellites signals bounce off surfaces like buildings, water, or the ground before reaching the receiver, rather than following a straight path. These reflected signals can interfere with the direct signal, leading to inaccuracies in position calculations. [2] To minimize multipath effects, specialized antennas, such as choke ring or ground plane designs, are often used. Additionally, advanced signal processing techniques can help filter out reflected signals by analysing their strength or angle of arrival. [2] Multipath errors are common in urban environments, where tall buildings and dense structures create complex signal reflections. [2]
Receiver Noise & Jamming:	Receiver noise and jamming errors occur when GNSS signals are disrupted by either internal electronic interference or external sources, including intentional jamming. These disturbances can significantly degrade the quality of the received signals, resulting in reduced accuracy and reliability of positioning data. [2] The impact of elevated noise is apparent in the reduced precision of signal correlation and tracking loops. This leads to less accurate measurements for both pseudo range and carrier phase. This then increases the time required to obtain a reliable position fix and lowers overall positioning accuracy. [2] Sources of receiver noise may include the internal electronics of the receiver and electromagnetic interference from nearby devices within the system. Understanding and mitigating these errors is crucial for achieving high-precision GNSS results [2]

GNSS RTK Principles

GNSS RTK (Global Navigation Satellite System Real-Time Kinematics) is a high-precision positioning technique that improves positioning accuracy of standard satellite navigation from meters down to centimeters. This is achieved by correcting errors in satellite signals which occurred due to several error sources, including satellite orbit and clock errors, as well as ionospheric and tropospheric effects. Combined, these error sources contribute to inaccurate positioning. RTK addresses this by using real-time correction data from a reference station whose exact coordinates are known. Therefore, the reference station is able to calculate the errors and send this information as correction data to the rover. [3]

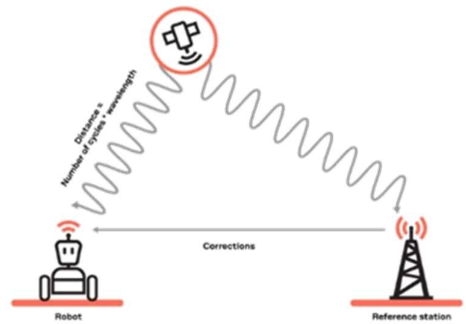


Fig. 8. RTK System with Reference Station [3]

There are two other main methods to implement correction:

GNSS with RTN (Real-Time Network) is an extension of the RTK method, in which a network of physically distributed GNSS reference stations is connected through a central control unit. Position errors are calculated based on all reference stations in the network and then sent to the user's rover in real time. This architecture extends coverage beyond a single RTK reference station, enhancing reliability and precision. This is especially valuable in areas with sparse station placement or varying environmental conditions. RTN typically provides centimeter-level accuracy similar to RTK, but over larger geographic areas and with improved robustness. [3]

GNSS with RTX (Real-Time eXtended) utilizes a global or continental network of reference stations to deliver precise orbit, clock, and bias corrections via the internet or satellite, eliminating the need for a local base station. These corrections support real-time absolute positioning with centimeter-level accuracy anywhere in the world, enabling operation even in remote regions where RTK or RTN networks are not available. RTX is characterized by the use of precise point positioning (PPP) and advanced algorithms for rapid initialization and high accuracy, benefiting from a global reference infrastructure rather than a local one. [3]

System Requirements & Architecture

Requirements

The requirements for this project are documented in a dedicated System Requirements Document, which is accessible in the appendix project folder. It contains 13 high-level system requirements, divided into functional, performance, and environmental requirements. In addition to these requirements, the document also specifies the relevant design constraints for the project.

Hardware System Architecture

The following section provides an overview of the hardware system architecture. It describes the individual components, explains how they are connected, and outlines their functions within the system.

Hardware Design

Fig. 9 shows the top-level assembly with the integrated GNSS / LiDAR sub-module and the corresponding interface hatch. This design and assembly structure provides modularity and simplifies the assembly of the entire system.

The GNSS antenna is placed centrally on top of the robot platform to maximize the "Sky View." This positioning ensures the antenna is completely unobstructed by the robot's structure or other equipment, allowing strong, direct satellite signal reception.

The LiDAR is also mounted centrally on the same bracket as the GNSS antenna. The tilt angle of 25° (relative to the base frame) enables the robot to detect obstacles and map its environment using the 180° spherical field of view of the Unitree LiDAR L2 4D.

The exact positions (relative to the body link) of the GNSS antenna and LiDAR are provided in a separate document. This document is available in Appendix Folder 3, "URDF Parameters."

All CAD files, incl. manufacturing data and pictures are available in the Appendix Folder 2, "CAD."
The bill of material can be extracted from the top-level assembly.

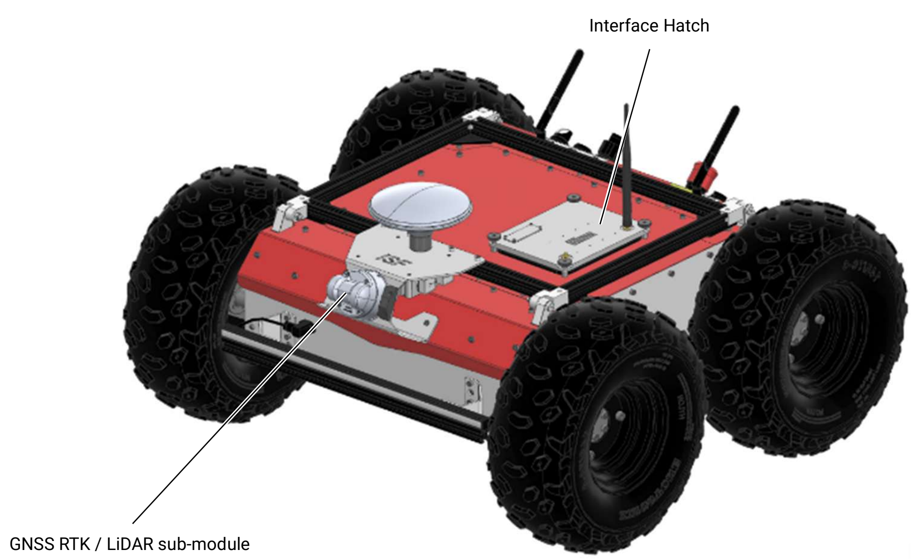


Fig. 9. Top-Level Assembly with integrated GNSS RTK / LiDAR sub-module

The interface hatch houses all enclosure connectors withing a single cap and features the same hole pattern as the original Husarion cap. This design allows the hatch to be mounted to the platform using only four screws, providing maximum flexibility and modularity. The antenna SMA connectors are IP67-rated to protect the Panther's internal electronics. A flat gasket seals the interface between the interface hatch and the user compartment enclosure. The interface hatch also includes an optional connector interface for the LiDAR.

The interface hatch (Fig. 10) serves multiple functions: it provides ingress protection for the user compartment, acts as a mounting bracket for the Raspberry Pi 5 and GNSS module, supports the dipole radio antenna, and functions as a heatsink for both the GNSS and radio modules.

This design ensures that the sensitive GNSS board is separated from high-power electronics and components that transmit high-frequency signals, minimizing EMC issues. In addition to this advantage, the design offers maximum flexibility, as only two cables are permanently connected to the robot platform, the power cable for the Raspberry Pi and the LAN cable from the switch to the Pi. This makes it easier to install or remove the entire module with minimal effort if needed.

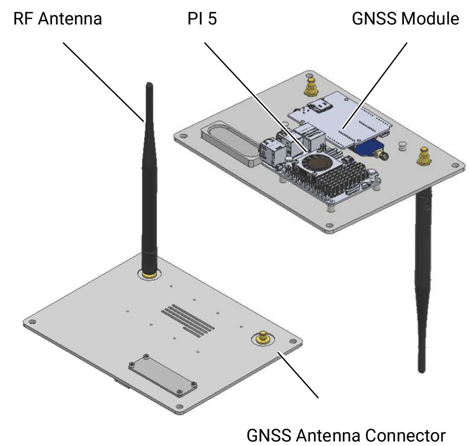


Fig. 10. Interface hatch with mounted components

The GNSS antenna is mounted on the same bracket as the LiDAR without any obstructions. The mounting adapter is a machined part which allows a precise positioning of the Antenna resulting in a total antenna offset ARP position tolerance of ± 0.5mm (without considering ITEM Profile tolerance chain). The adapter can also be used to mount the prism for the verification test (see Fig. 20 on page 21).

Below the sheet metal bracket, a terminal box for the LiDAR cable is mounted. It enables connection of the LiDAR's off-the-shelf connectors within an IP54-rated enclosure. Off-the-shelf cable feed-through glands are used, which do not require removing the bulky connectors due to their unique design and use of an elastomeric sealing element. The terminal box also helps reduce vibrations caused by the loose tolerance fits of the profile frame and the spinning LiDAR sensor.

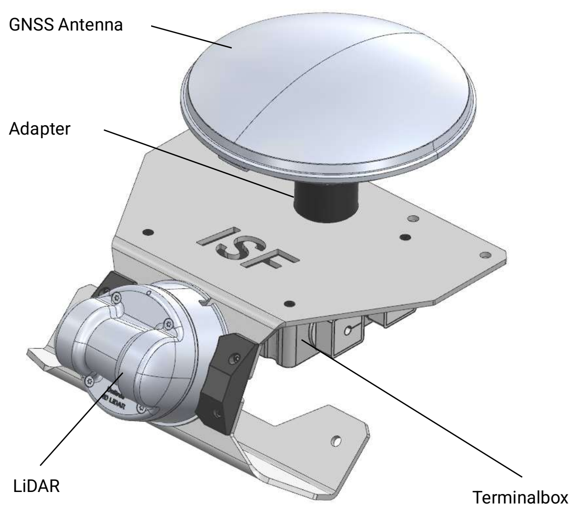


Fig. 11. GNSS RTK / LiDAR sub-module

Interconnection Diagram

The interconnection diagram illustrates the various subsystem modules and their connections within the system. Mechanical interfaces are not shown for clarity. A description of the main hardware components and their purpose can be found in the next chapter “Hardware Main Components” on page 14.

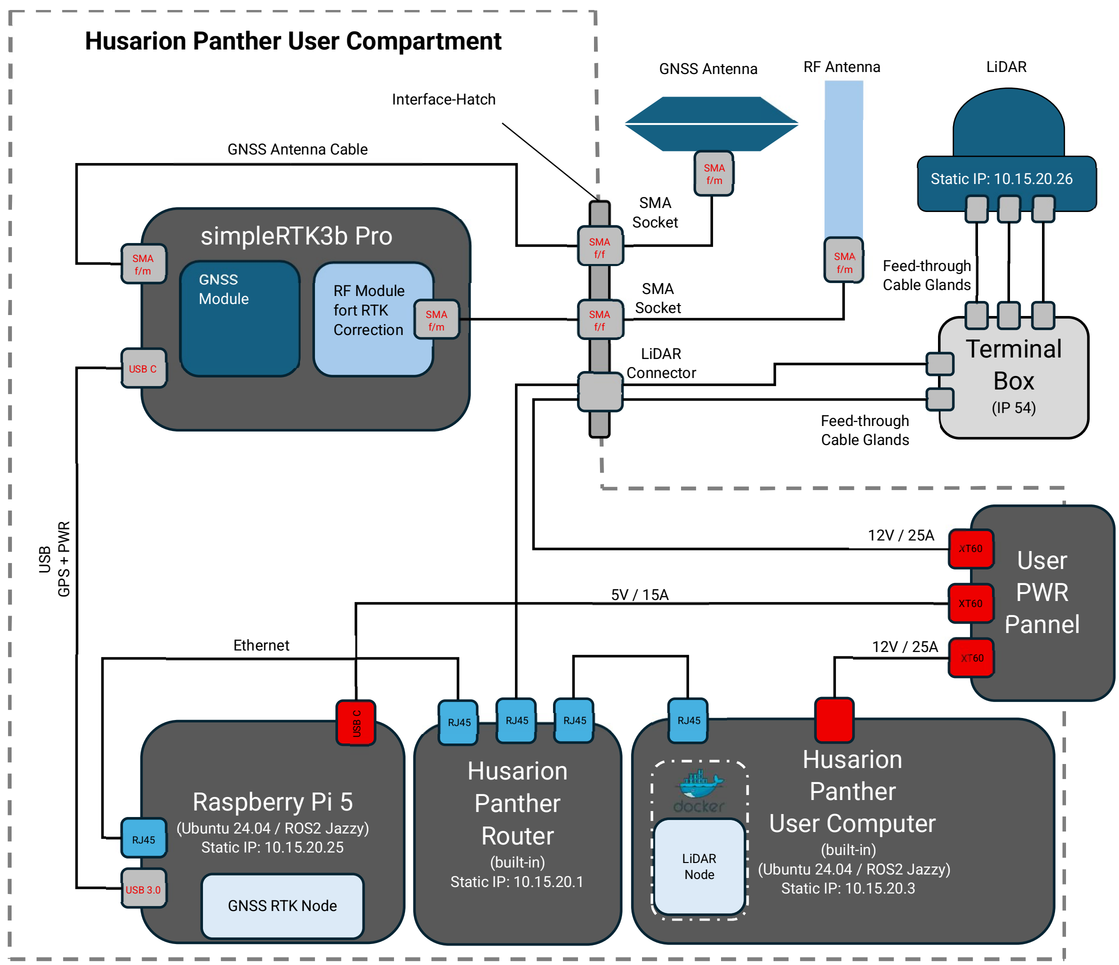


Fig. 12. Interconnection Diagram

Power:

- The Raspberry Pi 5 is powered via the user power pannel (5 V)
- The GNSS module receives power through a USB-C cable connected to the Pi
- The LiDAR sensor is directly powered by the 12 V user power panel

Communication:

- GNSS data is transmitted via the USB-C connection to the Raspberry Pi. The Pi runs the GNSS ROS 2 node and connects to the network, where it publishes it’s topics
- The LiDAR sensor connects directly to the built-in router and communicates over Ethernet. The corresponding ROS 2 node runs on the user computer inside a Docker container

Hardware Main Components

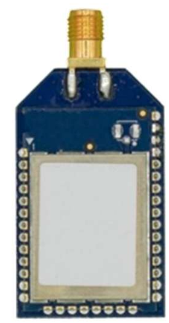
The following section provides an overview of the hardware off the shelf components used for this project with their main specifications.

Detailed specs of each component can be found in the

- Appendix Folder “Data Sheets”

GNSS Module		
Description:	A multi-band GNSS receiver module that provides centimeter-level RTK positioning accuracy. It connects to the ROS2 driver node via USB on the Raspberry Pi 5. Localization precision: • < 1cm with base station • < 0.6m standalone Update Rate: • 1 – 100Hz Multifrequency and multiconstellation: • GPS: L1C/A L1PY L2C L2P L5 • GLONASS: L1CA L2CA L2P L3 CDMA • Galileo: E1 E5a E5b E5 AltBloc E6 • BeiDou: B1I B1C B2a B2I B3 • QZSS: L1C/A L2C L5 • Navic: L5 • SBAS: WAAS, EGNOS, MSAS, GAGAN, SDCM (L1 L5) Interfaces: • USB • UART • XBee Start-up times: • Cold start: < 45s • Warm start: < 20s • Re-aquisition < 1s	 *Fig. 13. simpleRTK3B Pro [10]*
Purpose:	Provides centimeter-level positioning data for the Panther platform, enabling accurate localization and navigation.	

GNSS Antenna		
Description:	External precision antenna covering L1/L2/L5 frequency bands, factory-calibrated for optimal phase center stability. Supported positioning signal bands: • GPS: L1, L2, L5, L6 • GLONASS: G1, G2, G3 • BeiDou: B1, B2, B3 • Galileo: E1, E5, E6 Phase center error: • ±1mm	 *Fig. 14. Tripleband Antenna [11]*
Purpose:	Ensures reliable satellite signal reception and minimizes multipath distortion, supporting high-precision GNSS measurements.	

Radio Module		
Description:	Long-range transceiver capable of sending and receiving RTK correction data using serial or USB connection. Interfaces: • UART • RS232 Frequency: • 863-870 MHz Range in line sight: • Urban: 2.5 km • Rural: 5 km Antenna connector: • SMA female	 *Fig. 15. Radio Module [12]*
Purpose:	Transmits correction data between rover and base station to maintain RTK precision in real time.	

Raspberry Pi 5 (Ubuntu 24.04 / ROS 2 Jazzy)		
Description:	Computing unit running GNSS ROS2 driver node and communication interfaces. Connected to the Panther's control network via Lan cable. Processor: • 64-bit quad-core Arm Cortex-A76 (2.4GHz) RAM: • LPDDR4X-4267 SDRAM 8GB Wi-Fi: • Dual-band 802.11ac Wi-Fi Interfaces: • 2 × USB 3.0 • 2 × USB 2.0 ports • Gigabit Ethernet • 5V/5A DC power via USB-C	 *Fig. 16. Raspberry Pi 5 [13]*
Purpose:	Manages GNSS data acquisition and publishes GNSS topics in ROS2.	

Dipole Radio Antenna		
Description:	External antenna tuned for the radio module's operating frequency, designed for omnidirectional coverage and robust signal performance. Frequency range: • 863-927 MHz Gain: • 2.2 dBi Connector: • SMA Male	 *Fig. 17: Dipole Antenna for Radio [14]*
Purpose:	Maximizes radio link range and ensures stable data transmission for GNSS correction communication.	

LiDAR		
Description:	A compact LiDAR that provides 360° 3D point clouds with intensity for mapping, navigation, and obstacle avoidance. Field of View: • Ultra-wide-angle scanning (360° x 96°) Measurement range: • 30m @ 90% reflectivity Communication Interface: • Ethernet UDP/TTL • UART Blind Spot: • 0.05 [m] Dimension: • 75x75x65 [mm] Points/s • 64000	 *Fig. 18. Unitree LiDAR [15]*
Purpose:	Used as a main perception sensor to build maps, localize the robot, and detect obstacles for autonomous navigation	

Software Architecture

This section provides an overview of how the GNSS RTK module and the LiDAR sensor are integrated into the ROS2 software stack of the Husarion Panther robot platform. The Panther's onboard computer(s) run Ubuntu 24.04 and the ROS2 Jazzy distribution is used.

Integration

For the Sepentrio GNSS module Mosaic-X5 exists a ROS2 (Jazzy) compatible driver on Github:

- https://github.com/septentrio-gnss/septentrio_gnss_driver

This driver runs on a separate computer (Raspberry Pi 5) that uses the same Linux distribution as the onboard computer(s). The Pi 5 is connected to the same network and has a static IP addresses. ROS2 is used as middleware, allowing the localization nodes or any other subscribers to subscribe to the topics published by the GNSS driver node. The LiDAR sensor also has a static IP and is directly connected to the router.

Overview of IP addresses:

- Router: 10.15.20.1
- In-built computer (Pi 4): 10.15.20.2
- In-built user computer (intel nuc): 10.15.20.3
- GNSS_RTK computer (Pi 5): 10.15.20.25
- LiDAR: 10.15.20.26

Due to the limited time budget of this project, a full LiDAR software integration and verification could not be completed. One contributing factor is that the available LiDAR driver from Unitree is only compatible with an older ROS2 distribution (Foxy), which increases the complexity of integration with the selected ROS2 Jazzy environment. An initial attempt to operate the LiDAR on a native ROS2 Jazzy host system was nevertheless successful using a Docker-based approach. The LiDAR has the following static IP: 10.15.20.26 and was successfully powered by the robot system. Further documentation regarding the LiDAR software integration is not provided. The repository with the corresponding dockerfile can be shared on request but is not documented and still in a draft status.

Fig. 19 illustrates how the GNSS RTK module was integrated into the existing ROS2 software stack developed by Husarion. The upper part of the diagram, is available in Husarion's documentation for the Panther platform and provides a high-level overview of the individual nodes, plugins, objects, topics, services, and data flows within the system.

The lower part of the diagram shows how the Septentrio GNSS driver interacts with the rest of the software stack. It receives GNSS measurements and RTK correction data as inputs and publishes multiple output topics using standardized message types, such as NavSatFix and GPSFix. Table 1 on page 19 provides a high level overview of the most relevant in and outputs with their corresponding message types.

As illustrated in the diagram, the output data from the Septentrio GNSS driver is published via the ROS2 middleware. However, the existing software stack, particularly the localization node, does not actively subscribe to these topics or use the data for enhanced localization. The fusion of GNSS data with other sensor inputs was outside the scope of this project and would be subject to another project.

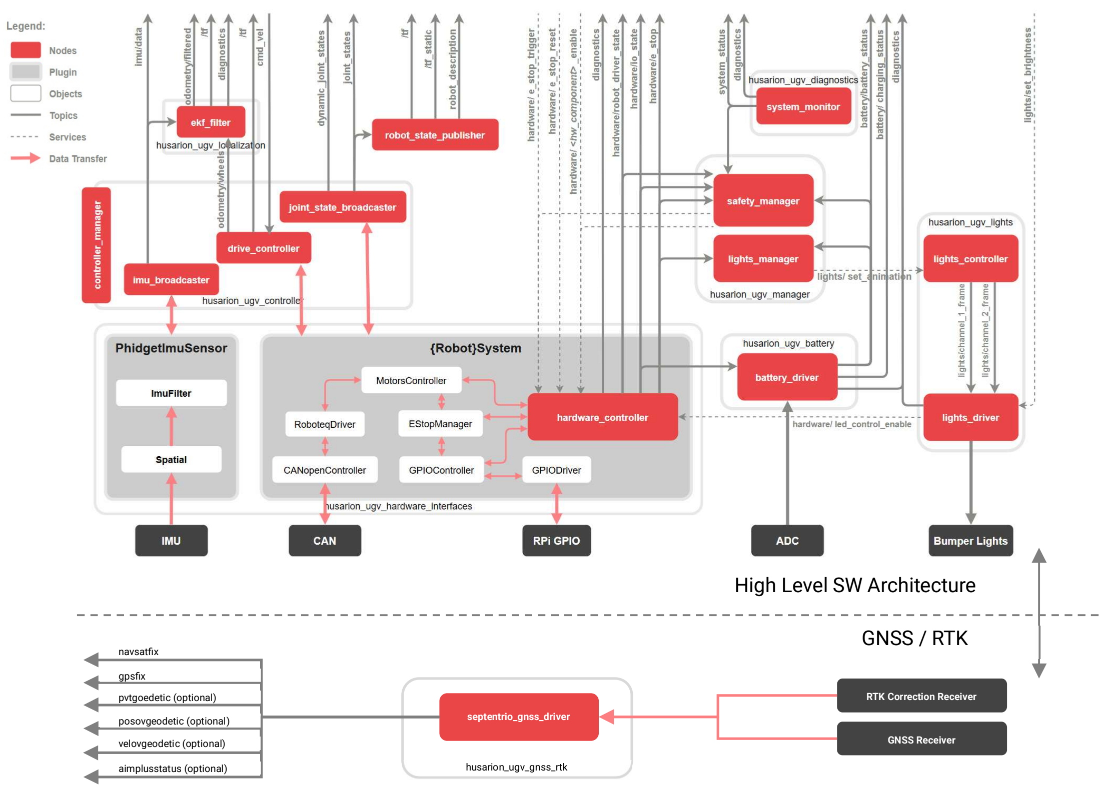


Fig. 19. Husarion Panther Software Architecture with GNSS RTK module (adapted from [8])

Raspberry Pi 5 Set-up

The development platform chosen for hosting the ROS2 GNSS RTK node is a Raspberry Pi 5, due to its compact form factor, low power consumption, and sufficient processing capability for GNSS RTK data processing.

- **Operating System:**
  A 64 GB SD card was used to flash the Ubuntu 24.04 image on the Pi.
  For the flashing of the image the Raspberry Pi Imager was used.

- **User Configuration and Remote Acccess:**
  User Name: husarion
  Password: GNSS_RTK

  In order to enable remote access SSH was enabled

- **ROS2:**
  Distribution: Jazzy

- **Workspace and Repository Setup:**
  ROS2 workspace was initialized and the required repository containg the Septentrio GNSS driver was clonded from GitHub. The workspace was then built using standard ROS2 build tools.

- **Automated Node Launch:**
  To ensure the GNSS RTK node starts automatically after boot, a shell script was created to launch the ROS2 node. This script was integrated into a systemd service file, allowing the node to start as a service whenever the Raspberry Pi is powered on.

- **Network Configuration:**
  The static IP 10.15.20.25 was assigned to the Pi to ensure reliable network connectivity, which is crucial for GNSS data communication. The preffered connection interface was set to LAN to reduce lateny and improve stabiltiy compared to wireless connections

GNSS RTK Set-up

The GNSS RTK module selected for this project is the Mosaic-X5 from Septentrio, chosen for its high performance (see specifications in Chapter "Hardware Main Components" on page 14), availability of ROS2 driver nodes, and, most importantly, its intuitive and straightforward configuration via the integrated web interface. This web interface distinguishes the Mosaic-X5 from other GNSS modules, such as those offered by u-blox, and was the key factor influencing the component selection.

Setup Process:

1. **Connect to Septentrio web interface**
   a. Connect the GNSS antenna to your receiver. Make sure the antenna has a good view of the sky for testing functionality.
   b. Connect the receiver to your PC via the USB port labelled as POWER+GPS *.
   c. Open a web browser and type in 192.168.3.1

*When you connect this product to a PC for the first time, the PC may not recognize it. You will only see a new Hard Disk in your computer. Open it and install the Septentrio driver. After installation, disconnect and reconnect again, your PC will recognize the receiver. This only need to be done once. On Linux systems like Ubuntu, usually native drivers like CDC-ACM suffice and no special USB driver installation is required.

2. **ConFig. GNSS receiver**
    a. Go to “Communication”, “Serial Port” and set Baud rate of COM1 and COM2 to 115’200 bps and press ok and save configuration
    b. Go to “GNSS”, “Position” and
        i. Select “Rover” mode
        ii. enable RTK Correction
    Press ok and save configuration

A detailed configuration instruction is provided by “SparkFun” and available here:
https://docs.sparkfun.com/SparkFun_GNSS_mosaic-X5/web_interface/#web-interface

GNSS Driver Node

The GNSS driver node fulfills four main functional roles:

1. **Data Acquisition:**
    - Establishes and maintains a communication channel with the receiver
    - Subscribes to Septentrio Binary Format (SBF) and optionally NMEA message streams
2. **Message Parsing and Translation:**
    - Decodes raw GNSS packets (SBF blocks) into structured internal representations
    - Transforms decoded data into ROS-native message types
    - Handles various message categories such as PVT (position/velocity/time), satellite status, base-station corrections, INS outputs, and diagnostics
3. **Configuration and Control:**
    - Applies startup configuration to the receiver (logging rate, output formats, correction sources, antenna configuration, etc.)
4. **Diagnostics and Time Synchronization**
    - Monitors receiver health, fix quality, RTK status, and message integrity
    - Integrates receiver time information into the ROS time system, offering support for PPS pin timing or GNSS-based clock synchronization if available

Septentrio offers a ROS2 compatible driver node for the mosaic x5 GNSS receiver.
GitHub repository link: https://github.com/septentrio-gnss/septentrio_gnss_driver.git

Input / Output Interfaces:

The following table provides an overview of the most relevant in-and-outputs of the ROS2 node:

I/O	Topic / Interface	Message Type	Description
Input	Serial stream	SBF (Septentrio Binary Format)	Raw GNSS and RTK correction data
Input	RF correction	RTCM	Real-time kinematic correction input
Output	/navsatfix	sensor_msgs/NavSatFix	Standard GPS fix (latitude, longitude, altitude)
Output	/gpsfix	gps_common/GPSFix	Extended fix message with DOP, satellite count
Output	/diagnostics	diagnostic_msgs/DiagnosticArray	System health info
Output	Etc.		

Table 1. Input & Outputs of GNSS Driver Node

Configuration and Launch files

Both the configuration and launch files are available in the following GitLab repository:
https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration

Two configuration files are provided in the config folder. The "gnss.yaml" file is intended for use when the GNSS module operates as a base station, while the "rover.yaml" file serves as the configuration for the rover setup. This structure allows for efficient switching between the two use cases. For this project the "rover.yaml" file is used.

```yaml
# Configuration Settings for the Rover
# GNSSParameters
polling_period:
  pvt: 100
  rest: 500
multi_antenna: false
publish:
  # For both GNSS and INS Rxs
  navsatfix: true
  gpsfix: true
  pvtgeodetic: true
  poscovgeodetic: true
  velcovgeodetic: true
```

- **Polling Period:**
  The "polling_period" parameter specifies the query rates for GNSS data, where pvt denotes the position-velocity-time solution polled every 100 ms, and rest covers all remaining GNSS messages polled every 500 ms.

- **Multi Antenna:**
  Was set to false since only one antenna is used.

- **Topics:**
  The topics listed above are published by the ROS2 node and are available to the localization and navigation nodes. Additional topics, such as diagnostics, can be enabled in the configuration file by setting the corresponding flag.

  - gpsfix:
    - Publishes common GNSS data
    - RTK Status
    - Number of satellites used
    - Number of satellites visible
    - Dilution of Precision (DOP) values
  - navsatfix:
    - Publishes sebsir_msgs/NavSatFix
    - Gives latitude, longitude, altitude + status flag
    - Standard ROS GNSS message → useful for localization or logging

Testing and Verification

Methodology Overview

Experimental Setup

Fig. 20 and Fig. 21 illustrate the experimental test setup. The use of a Leica TS16 robotic total station was recommended to improve the robustness and accuracy of the verification measurements (Prof. Dr. Dejan Šeatović, personal communication, Oct. 2025). To measure the distance between the total station and the rover, a reflecting prism is mounted along the same vertical axis as the GNSS receiver antenna. The combination of the prism and the total station's automated tracking capabilities allows continuous tracking of the robot, thereby enabling a dynamic and reliable measurement process.

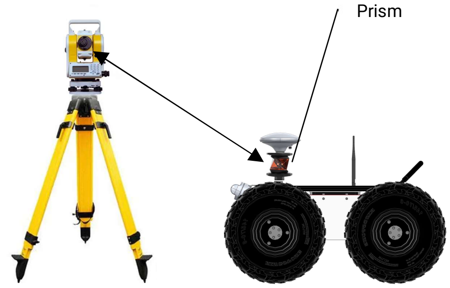


Fig. 20. Measurement Setup

Set-up Process:

1. Setup base station at known position and ensure correction data is send / received
2. Select position of total station which guarantees unobstructed view on the rover's measurement trajectory
3. Align the total station vertically and horizontally.
4. Measure the height of the total station
5. Create reference X-Axis for local coordinate system
6. Select correct reflecting prism type to consider vertical offset

The RTK base station comes already calibrated and is provided by the institute. The measurement trajectory was also defined by the institute and is shown in Fig. 21 in the top right corner.

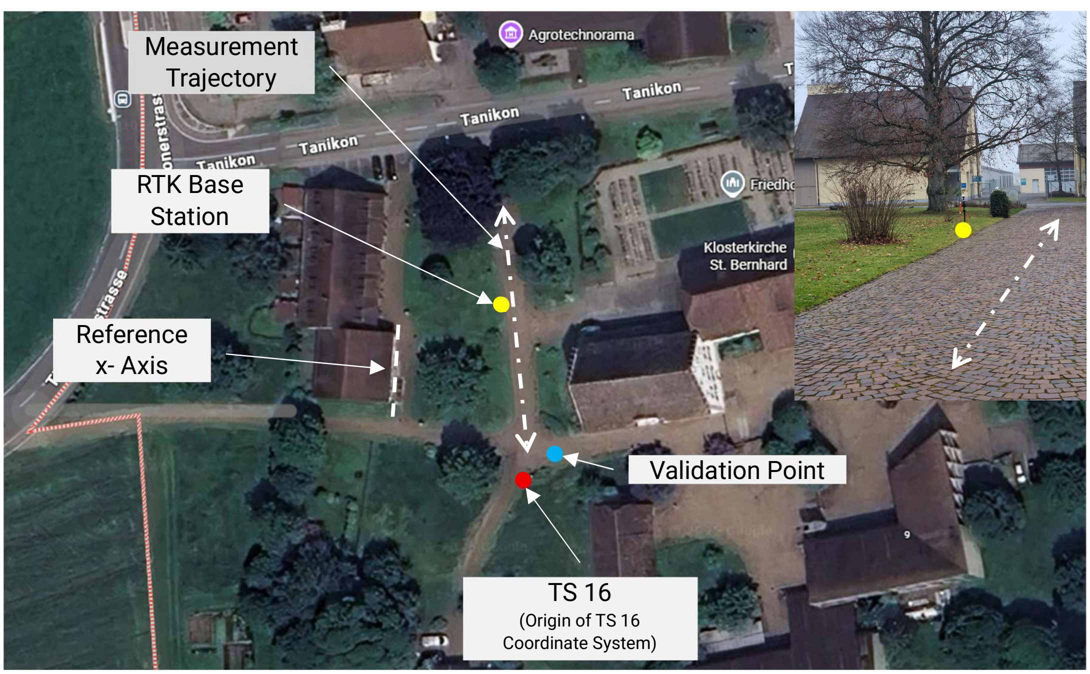


Fig. 21. Verification Setup / Environment

Test Conditions

The verification test was conducted on December 4, between 09:50 and 10:30 am, in Tänikon. The measurement location is shown in Fig. 21 on page 23. The GNSS antenna had a clear view of the sky, with no nearby buildings that could cause significant multipath effects. Although a large tree was present at the end of the measurement trajectory (see top right corner in Fig. 21), it had lost all of its leaves and was therefore considered unlikely to significantly affect the measurements.

Weather conditions were suboptimal, with fog and an ambient temperature of approximately 8 °C. However, the environmental conditions remained stable throughout the entire testing period.

During the test, pedestrians frequently passed the robot at close distances.

Prior to and throughout the test, the Dilution of Precision (DOP) values, as well as the number of visible and used satellites, were continuously monitored in a separate terminal to ensure compliance with the conditions specified in the system requirements document.

Test Cases

Two test cases were defined to verify the performance of the GNSS RTK module. The continuous-motion test case was specified by the institute, while the stop-and-go test case was included as an optional extension to evaluate the system's performance under static measurement conditions. In both cases, the trajectory was not strictly predefined or programmed and was determined by inputs from the user via the remote control. However, the goal was to follow a straight-line trajectory of approximately 70 meters, starting and ending at roughly the same location without major direction changes or corrections.

Continuous-motion test case:
In the continuous-motion test case (see Fig. 22), the robot moves continuously along the measurement trajectory at a constant speed of approximately 1 m/s until it reaches the reversal point where the robot briefly stops and reverses its direction without performing a full 180° rotation. Reference measurements with the total station are acquired while the robot is in motion at intervals of approximately 2–3 seconds, resulting in a dense and continuous set of trajectory data. The robot logs GNSS data continuously at a frequency of 10 Hz.

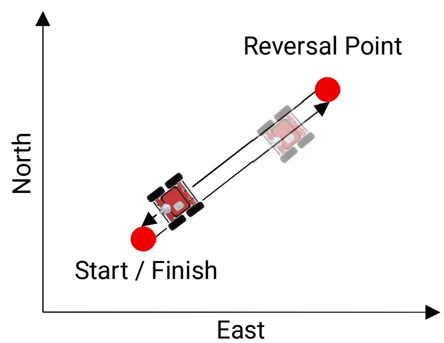


Fig. 22. Visualization of test case 1

Stop-and-go test case:
In the stop-and-go test case, the robot switches between motion and stationary phases. The robot moves for approximately 5 seconds, then stops to perform a measurement using the total station reference system. This sequence is repeated at successive measurement locations until the end of the trajectory is reached. As in the continuous-motion case, the robot reverses direction without performing a full 180° rotation at the reversal point. Also, in this test case the robot loggs GNSS data continously at 10 Hz.

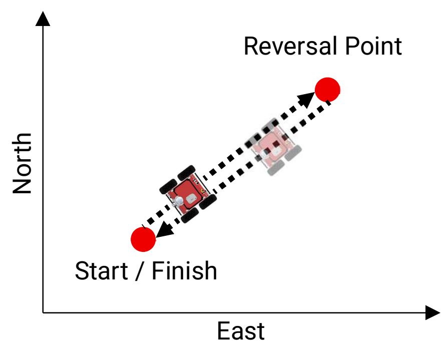


Fig. 23. Visualization of test case 2

A total of six measurement series were conducted. Three series were performed using the continuous-motion approach (Test002–Test004), and three series were performed using the stop-and-go approach (Test005–Test008).

Measurement Procedure

After the setup was completed as described on page 21, the actual measurement procedure was carried out as follows:

1. DOP values were checked to ensure they were within an acceptable range.
2. The number of used and visible satellites was verified to be sufficient.
3. The start time of the test was recorded.
4. The first total station measurement point ID was recorded.
5. It was checked if the reflecting prism was locked by the total station.
6. A new data log was started on the robot using the command provided below.
7. The first measurement with the total station was taken. The robot was then moved toward the reversal point, and measurements were collected approximately every 2–3 seconds with the total station. Alternatively, for the stop-and-go method, the robot was moved toward the reversal point, and measurements were taken with the total station during the stationary phases.
8. The robot reversed direction without turning around.
9. Step 7 was repeated until the finish point was reached.
10. The end time of the test was recorded.
11. The last total station measurement point ID was recorded.
12. The data log on the robot was stopped using the command provided below.

Total Station Data Log:
Each measurement point was recorded manually and saved in the corresponding project folder on the SSD. The measurement data was subsequently exported as a raw text file.

Robot Data Log:
The robot data was logged on the onboard computer via SSH. To ensure continuous data logging even in scenarios where the wireless connection between the robot and the user computer was interrupted, the following ROS2 command was used:

```
nohup ros2 bag record /topic1 /topic2 -o bag_$(date +%Y%m%d_%H%M%S) &
```

- nohup → keeps logging after SSH disconnect
- ros2 bag record → default rosbag behaviour
- -o bag_YYYYMMDD_HHMMSS → timestamped output directory
- & → runs in background
- No extra logs, no redirection, no scripting

```
pkill -SIGINT -f "ros2 bag record"
```

- Stop logging

Evaluation Metrics

To assess the performance of the GNSS RTK module, a set of evaluation metrics was defined, comparing its measurements to reference data collected with a Leica TS16 robotics total station. The metrics were selected to quantify accuracy and reliability under both static and dynamic conditions.

Position Accuracy:
Position accuracy describes how closely the GNSS RTK measurements match the reference points measured with the total station. Accuracy was evaluated separately for horizontal and vertical components.

- **Horizontal Accuracy:**
  The horizontal accuracy measures the deviation in the XY plane between the GNSS RTK measurement and the reference total station measurement.

$$E_{2D,i} = \sqrt{\left(X_{\mathrm{GNSS},i} - X_{\mathrm{TS},i}\right)^2 + \left(Y_{\mathrm{GNSS},i} - Y_{\mathrm{TS},i}\right)^2} = \sqrt{(\Delta X_i)^2 + (\Delta Y_i)^2}$$

Equation 1. Horizontal Error

$$\mathrm{RMSE}_{2D} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[(\Delta X_i)^2 + (\Delta Y_i)^2\right]}$$

Equation 2. Horizontal RMSE

- **Vertical Accuracy:**
  The deviation in height between GNSS and total station measurements was assessed using the same statistical indicators.

$$\Delta Z_i = Z_{\mathrm{GNSS},i} - Z_{\mathrm{TS},i}$$

Equation 3. Vertical Error

$$\mathrm{RMSE}_{Z} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\Delta Z_i)^2}$$

Equation 4. Vertical RMSE

- **3D Positioning Accuracy:**
  The 3D positioning accuracy can be calculated as a combination of horizontal and vertical error. The same statistical indicators are used.

$$E_{3D,i} = \sqrt{\left(X_{\text{GNSS},i} - X_{\text{TS},i}\right)^2 + \left(Y_{\text{GNSS},i} - Y_{\text{TS},i}\right)^2 + \left(Z_{\text{GNSS},i} - Z_{\text{TS},i}\right)^2} = \sqrt{(\Delta X_i)^2 + (\Delta Y_i)^2 + (\Delta Z_i)^2}$$

Equation 5. 3D Error

$$RMSE_{3D} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}[(\Delta X_i)^2 + (\Delta Y_i)^2 + (\Delta Z_i)^2]}$$

Equation 6. 3D RMSE Error

Reliability and Availability:
Reliability metrics capture the GNSS RTK module's ability to provide valid position solutions under various conditions:

- Percentage of time with a fixed RTK solution
- Dilution of Precision (DOP) values

Positioning Error Evaluation Methods

Method 1: Path-Based Orthogonal Error Method
In the path-based orthogonal error method, each GNSS position is evaluated relative to a reference trajectory measured by the robotic total station TS16 from Leica. The reference points measured by the total station are represented as a continuous two-dimensional trajectory using a line string representation. For each GNSS coordinate point, the closest position along the total station reference trajectory is determined by orthogonal projection. A linear interpolation is considered unproblematic, even though the number of total station coordinate points is sparse, because the measurement trajectory is straight. This assumption holds as long as there are no significant direction changes during the measurement, especially between two total station measurement points.

The corresponding height value is obtained through linear interpolation of the total station elevations along the cumulative path distance. Horizontal, vertical, and three-dimensional positional errors are then calculated as the differences between the GNSS points and their projected reference positions. Root Mean Square Error (RMSE) values are computed over all GNSS points to quantify overall positioning accuracy.

This method ensures that GNSS errors are evaluated relative to the true continuous trajectory rather than discrete reference points, providing a more accurate assessment.

Pros:

- Accurate cross-track error measurement
- Smooth height comparison even if total station coordinate points are sparse
- Reflects true deviation from the trajectory

Cons:

- Requires a continuous trajectory representation
- Slightly more computationally complex

The script used for the evaluation with this method is available in the GitLab repository:

- https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration/gnss_rtk_verification/src/gnss_rtk_vs_ts16_path_based_orthogonal_error.ipynb

Method 2: Nearest-Neighbour Method:
In the nearest-neighbour method, each coordinate point measured by the total station is directly compared to the closest GNSS coordinate point in three-dimensional space. Horizontal, vertical, and three-dimensional errors are computed as the direct differences between the two coordinate points. RMSE values are subsequently calculated overall total station coordinate points.

Unlike the path-based method, this approach does not account for path continuity and is therefore sensitive to the spatial density of the total station reference points. This can potentially lead to overestimated errors.

It should also be noted that this method may produce overly optimistic results if used alone in the stop-and-go test case (see Section Test Cases on page 22). During a 5-second static phase, the GNSS module provides 50 measurements at a sampling frequency of 10 Hz, from which only the best-matching point is used for the error calculation.

Pros:
- Simple to implement
- Works well when total station points are dense

Cons:
- Overestimates errors if total station points are sparse
- Does not account for path continuity
- Works only for stop-and-go test case

The script used for the evaluation with this method is available in the GitLab repository:
- https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration/gnss_rtk_verification/src/gnss_rtk_vs_ts16_nearest_neighbour.ipynb

Due to their different characteristics, the two methods are expected to yield different results. The path-based orthogonal error method is better suited for the continuous-motion test case, whereas the nearest-neighbor method is more appropriate for the stop-and-go test case. For measurement series 5, the nearest-neighbor method is additionally applied alongside the path-based orthogonal error method as a sanity check.

Expected Outcomes

The following expectations were formulated prior to the data analysis, based on GNSS-RTK characteristics, DOP values, satellite information, and the test condition and environment:

- **High-quality GNSS data:**
  Based on the monitoring of DOP values and satellite information, the collected rover data is anticipated to be of high quality. Consequently, the differences between the total station coordinates and the GNSS-RTK data points are expected to be at the centimeter level.

- **Reliable RTK correction service:**
  It is expected that the RTK correction service provides sufficiently continuous RTK correction data throughout the measurement series to support high-accuracy positioning. Short interruptions in RTK correction service may occur due to environmental or communication-related factors, but long periods without RTK correction service are not anticipated under the given test conditions.

- **Reliable reference measurements:**
  The total station's automated tracking system is anticipated to maintain continuous lock on the rover, providing a reliable reference dataset.

- **Improved accuracy with stop-and-go method:**
  Higher positioning accuracy is expected in the stop-and-go method for several reasons. During stop-and-go measurements, the rover remains stationary while data is collected,

effectively eliminating motion-induced errors. Additionally, this method reduces noise compared to continuous motion, where shocks and vibrations caused by traversing the gravel surfaces, can affect antenna measurements.

- **Horizontal vs. vertical accuracy:**
  Horizontal positioning is expected to be more accurate than vertical positioning, in line with typical GNSS performance.

- **Effect of satellite geometry:**
  During periods of favorable satellite geometry, indicated by low PDOP or GDOP values, the accuracy is expected to improve further.

- **Environmental factors:**
  Fog or moderate temperature fluctuations are not expected to significantly affect GNSS-RTK measurements. However, occasional noise may arise from external disturbances, such as passing pedestrians or other unpredictable environmental effects.

Overall, these expectations provide a framework for assessing the GNSS-RTK module's positioning performance and for comparing its measurements against the high-precision reference trajectory provided by the total station.

Data Analysis

Data Pre-processing

Coordinate Transformation

To validate the performance of the GNSS RTK module, the coordinate data logged by the rover and the Leica TS16 robotic total station must be expressed in the same coordinate system. The data points logged by the rover are in the WGS84 (World Geodetic System 1984) coordinate system, whereas the total station uses its own local coordinate system, which was established during the setup.

Therefore, the rover data must be transformed in two steps: first into the Swiss coordinate system CH1903+, and subsequently into the local coordinate system of the total station.

WGS84 to CH1903+:
For this transformation, the tool provided by Swisstopo can be utilized. It enables the upload of CSV files containing WGS84 coordinates and converts them into CH1903+ coordinates, or vice versa.
Swisstopo tool: https://www.swisstopo.admin.ch/en/coordinate-conversion-reframe

The following two Figures show the input and output settings used for the transformation:

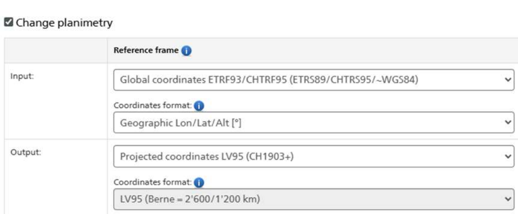


Fig. 24. Input Settings

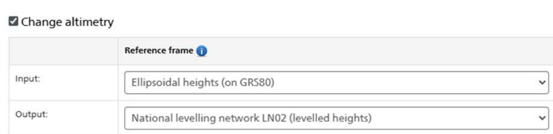


Fig. 25. Output Settings

The tool provided by Swisstopo requires a CSV file in a specific structure. Therefore, a script was written to take the raw CSV files from the different test cases and reformat them into the required format.

The script in the GitLab repository:

- https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration/gnss_rtk_verification/src/raw_2_swisstopo.py

CH1903+ to local coordinate system:
The transformation between the CH1903+ coordinate system and the local coordinate system was performed using a Helmert transformation. This method was chosen because it accounts simultaneously for translation, rotation, and scale differences between the two coordinate systems [5]. The implementation follows the Umeyama algorithm, which provides a closed-form, least-squares optimal solution, ensuring the best possible alignment of all coordinate points in both data sets (GNSS Coordinates and Total Station Coordinates) [6].

ID	GNSS Coordinates (CH1903+ Frame)			Total Station (Locale Frame)		
	X_{GNSS}	Y_{GNSS}	Z_{GNSS}	X_{TS}	Y_{TS}	Z_{TS}
TP0166	2710628.2520	1259735.8350	542.6712	-7.048	-27.875	-2.653
TP0167	2710627.6160	1259739.1430	542.6745	-3.751	-27.099	-2.691
TP0168	2710627.1530	1259741.7590	542.5968	-1.175	-26.506	-2.753
TP0169	2710626.7090	1259744.7220	542.5587	1.762	-25.923	-2.816
TP0170	2710626.2910	1259748.3760	542.4845	5.392	-25.333	-2.873
TP0171	2710625.8150	1259752.4640	542.3897	9.459	-24.67	-2.957
TP0172	2710625.3940	1259755.8690	542.3586	12.841	-24.086	-2.994
TP0173	2710624.9390	1259759.4930	542.2947	16.434	-23.455	-3.061
TP0174	2710624.2800	1259764.5110	542.2118	21.426	-22.565	-3.142
TP0175	2710623.7770	1259768.7250	542.1563	25.601	-21.854	-3.206
TP0177	2710624.0550	1259766.5240	542.1759	23.414	-22.236	-3.17
TP0178	2710624.4320	1259763.4030	542.2175	20.319	-22.765	-3.127
TP0179	2710624.8810	1259759.5700	542.2951	16.511	-23.395	-3.064
TP0180	2710625.2390	1259756.2370	542.3459	13.188	-23.905	-3.004
TP0181	2710625.6280	1259752.1280	542.3947	9.104	-24.491	-2.956
TP0182	2710625.9870	1259747.7890	542.4844	4.792	-25.058	-2.874
TP0183	2710626.1930	1259744.3350	542.5236	1.354	-25.426	-2.829
TP0184	2710626.3560	1259741.4950	542.6019	-1.465	-25.72	-2.765
TP0185	2710626.5050	1259738.9180	542.6493	-4.039	-25.991	-2.714

Table 2 Reference Points used for CH1903+ to local coordinate system transformation

The reference points used to derive the Helmert transformation parameters were extracted from measurement series 5 (Test005). In this series, a stop-and-go method was used, which is evident in Fig. 26 as the stair-step patterns in longitude and latitude. Total station measurements were conducted while the robot remained stationary to minimize errors caused by vibrations, reflections, or other disturbances.

The following filtering criteria were applied to get the reference points in Table 2 on page 28:

- gpsfix.status.status = 19 (RTK Fixed Integer)
- gpsfix.altitude > 589

These criteria ensured that only reliable, realistic and consistent measurements were retained for the subsequent coordinate transformation. The large positional jump observed in the second half of the measurement series could not be conclusively explained and was therefore excluded by the applied filtering rules.

The following figure presents the time-synchronized GPSFix raw data, including longitude, latitude, DOP values, RTK correction status, and satellite information. This plot allows a quick visual inspection of the dataset. Out of the eight measurement series conducted, series 5 (Test005) was considered the best for extracting the reference points (Table 2) needed for the CH1903+ to local coordinate transformation due to the distinctive steps, which allowed matching coordinate points of both datasets without the need for precise calculation of time offsets.

The script to inspect the raw data is available in the GitLab repository:

- https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration/gnss_rtk_verification/src/raw_data_preview.ipynb

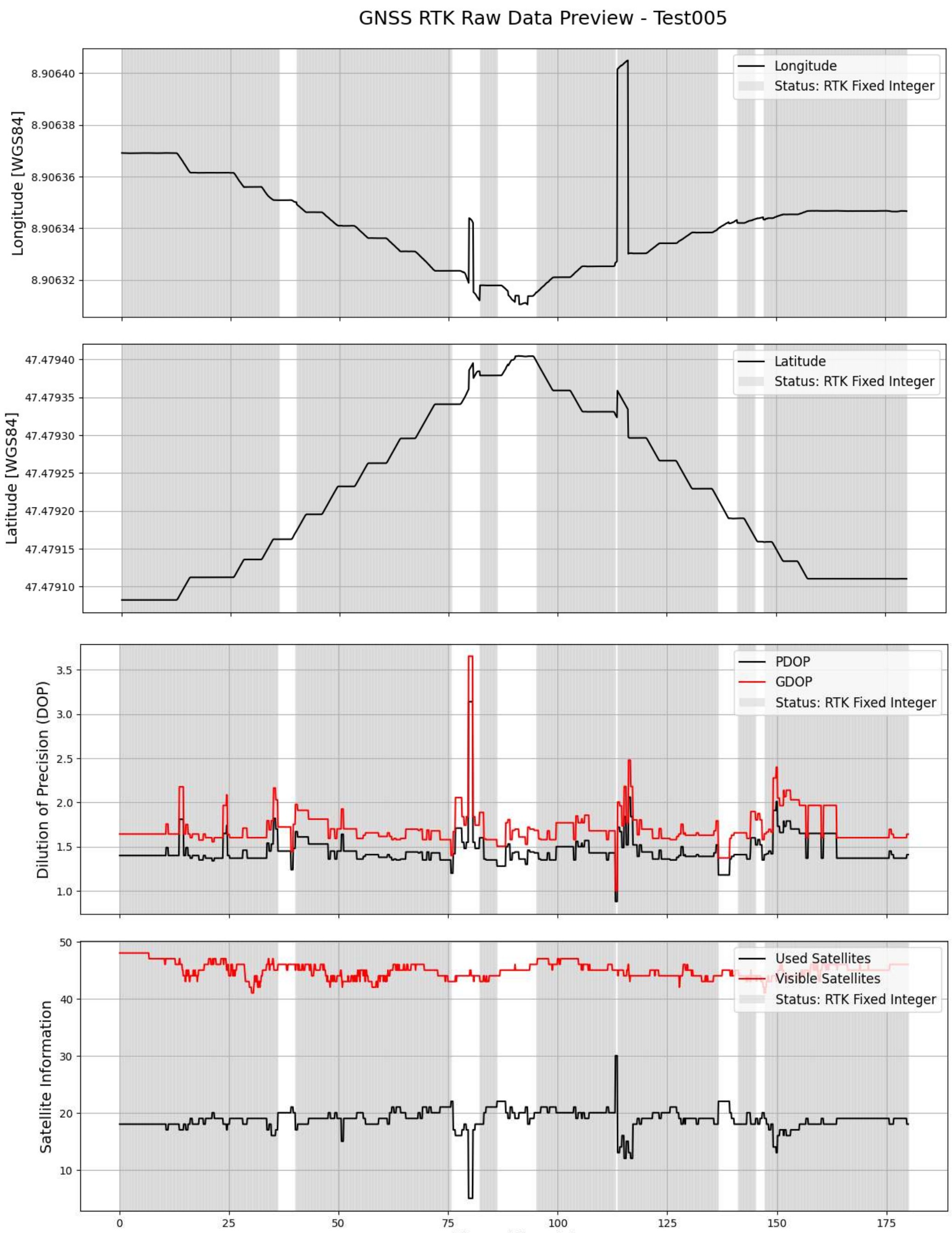


Fig. 26. Rover Coordinates (WSG84), DOP and Satellite Information - Test005

After the initial iteration of the Umeyama algorithm, the three reference points with the largest resulting residual magnitudes were removed from the reference dataset to improve the overall quality of the transformation. Fig. 27 shows the 3D Residuals Histogram including the three largest outliers.

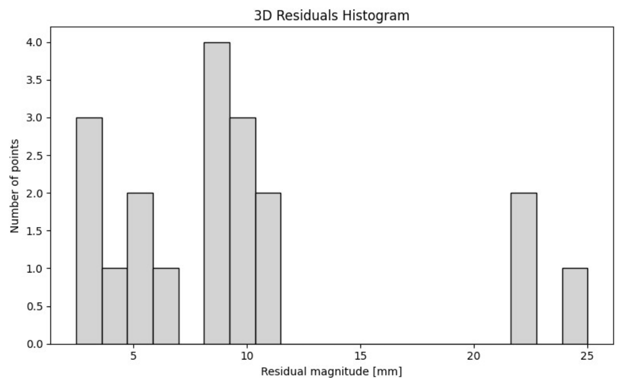


Fig. 27. 3D Residuals Histogram with outliers

The resulting Helmert Parameters:

- **Scale Factor s:**
  - $s = 0.9999415428186998$

- **Rotation Matrix R:**
  - $$R = \begin{bmatrix} 0.04753373 & 0.99886712 & -0.00223875 \\ -0.99877665 & 0.04755977 & 0.01353806 \\ 0.0136292 & 0.0015925 & 0.99990585 \end{bmatrix}$$

- **Translation Vector t:**
  - $$t = \begin{bmatrix} -1387079.71555932 \\ 2647209.46263466 \\ -39492.79468717 \end{bmatrix}$$

- **Fit Quality:**
  - $$rms = \begin{bmatrix} 0.00499467 \\ 0.00315058 \\ 0.00507683 \end{bmatrix}$$

The calculated scale factor s indicates that the local coordinate system is slightly smaller. A transformation without scaling would correspond to a scale factor of s = 1.

The root mean square (RMS) errors per axis are approximately 5 mm, 3 mm, and 5 mm showing that the point-to-point fit achieves millimeter-level accuracy. Consequently, the derived rotation matrix and translation vector are suitable for transforming the rover coordinates from the CH1903+ reference frame to the local coordinate frame of the total station.

The corresponding python script to perform the calculation of the transformation parameters is available in the GitLab repository:

- https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration/gnss_rtk_verification/src/coordinate_transform.py

The corresponding python script to perform the coordinate transformation with the calculated Helmert parameters is available in the GitLab repository:

- https://gitlab.ost.ch/isf/studienarbeiten/panther-gnss-integration/gnss_rtk_verification/src/ch1903plus_2_ts16.py

Filtering

The following two Figures present the unfiltered ROS2 GPSFix topic for measurement series 3 and 4 (Test003 and Test004). The data include longitude, latitude, altitude, position covariance, DOP values, satellite information, and RTK status. For improved readability, the plots are also provided in their original resolution in in the Appendix Folder 6, “Raw Data Plots”. The raw data from both the rover and the total station are available in the GitLab repository or in the Appendix Folder 5, “Verification Data”.

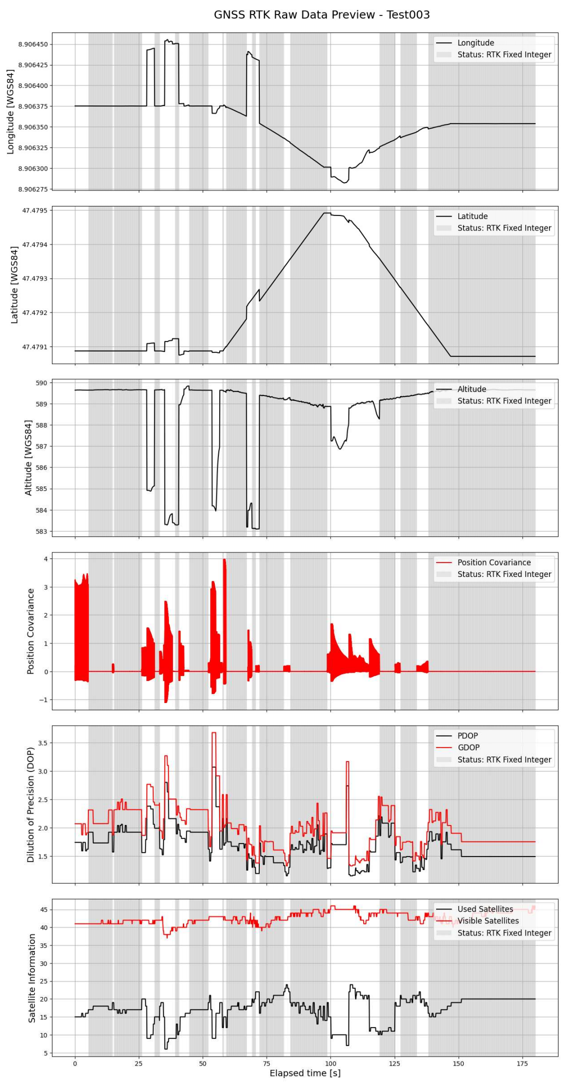


Fig. 28. Test003 Raw Data Preview

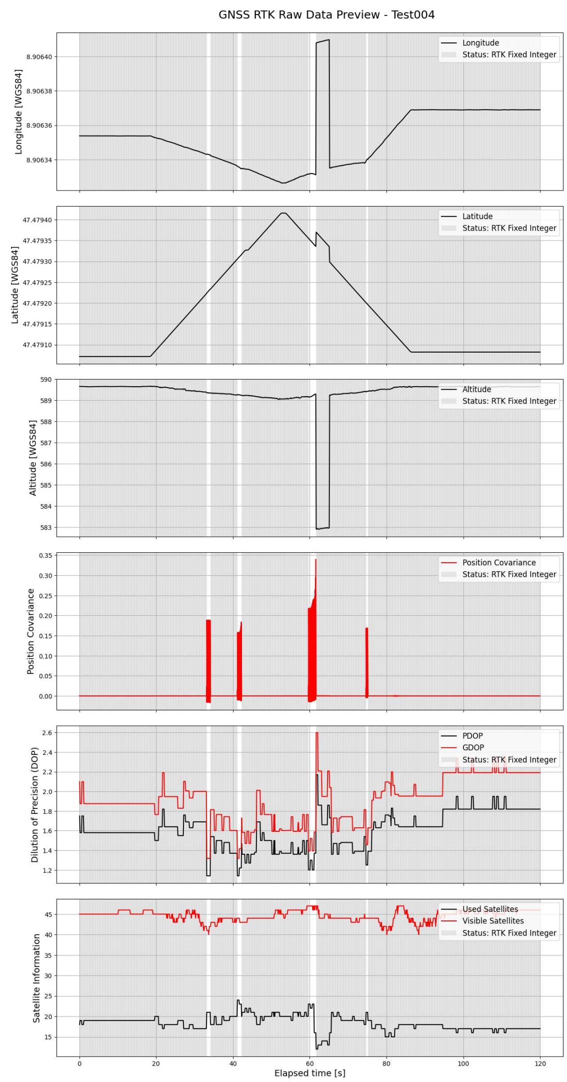


Fig. 29. Test004 Raw Data Preview

As shown in the figures above, the position data values are not consistently smooth and display abrupt deviations caused by various factors. These deviations may result from missing RTK correction data, external noise, multipath effects, or other error sources. Therefore, only coordinate data points obtained while RTK corrections were active were considered for the position accuracy evaluation. This filter was not applied for the evaluation of the reliability and availability performance metric.

The remaining measurement series (Test006, Test007, and Test008) show very poor RTK coverage and significant noise (see Appendix Folder 6, “Raw Data Plots”). During these tests, several pedestrians moved in close proximity to the robot, which may have contributed to the degraded data quality and were therefore not considered for the evaluation of the position accuracy of the GNSS RTK module.

In addition to the deviations mentioned above, the mismatch between the linearly interpolated total station trajectory and the GNSS rover trajectory presents another issue. This mismatch occurs because the total station measurement points are sparse. As a result, the interpolation can cause effects such as corner cutting, as shown in the zoomed in section in Fig. 30. This problem mainly occurs in sections where the operator had to actively change direction and could not follow a straight line due to lateral drift of the robot.

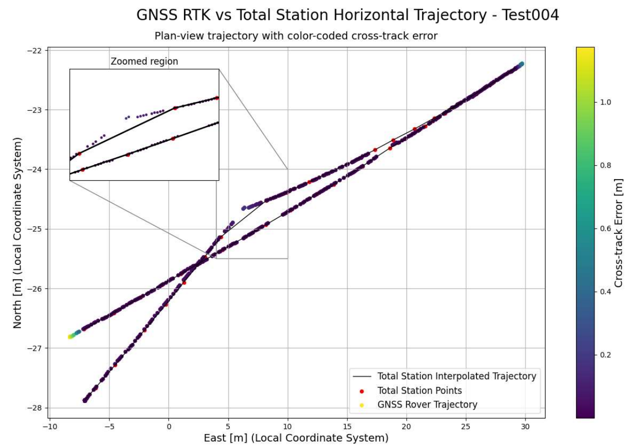


Fig. 30. Example of trajectory mismatch

A similar mismatch is also visible at the start and end of the trajectory (see Fig. 30). However, the cause is different. In these cases, GNSS data points recorded before the actual start and after the finish of the run must be removed in order to obtain a meaningful positioning accuracy metric. If not considered correctly in the evaluation process this invalid data points can lead to large cross-track errors that are not caused by GNSS positioning inaccuracies.

To remove these invalid data points, a threshold value of 17.5 mm for cross-track error was defined and applied to filter the data sets. The resulting trajectory for test series 4 (Test004) is visualized in Fig. 31.

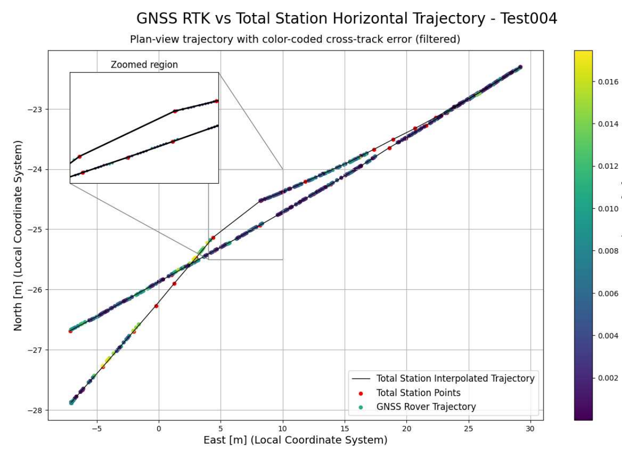


Fig. 31. Example trajectory with removed invalid data points

Results

Horizontal Position Accuracy

This section presents the horizontal position accuracy results of the GNSS RTK module, evaluated against the reference trajectory measured with the Leica TS16 total station. The comparison is based on a plan-view trajectory overlay and a statistical analysis of the horizontal position errors.

Fig. 32 and Fig. 33 illustrate the horizontal trajectories of the GNSS RTK module overlaid with the corresponding reference trajectory measured by the total station for the measurement series 3 and 4 (Test003 and Test004). The total station is shown as a linearly interpolated reference trajectory, with discrete total station measurement points also visible. A total of 224 validation points were obtained for measurement series 3 and 544 points for the measurement series 4. The figures further illustrate that the data sets were pre-processed to include only valid and relevant data points causing multiple gaps in the GNSS rover trajectory (see zoomed region in Fig. 33). The pre-processing procedure, including the applied filter criteria and their justification, are described in the previous section “Data Pre-processing” on page 27.

In both test cases, the GNSS rover trajectory follows the overall shape of the reference trajectory. Both Figures indicating very good horizontal positioning accuracy.

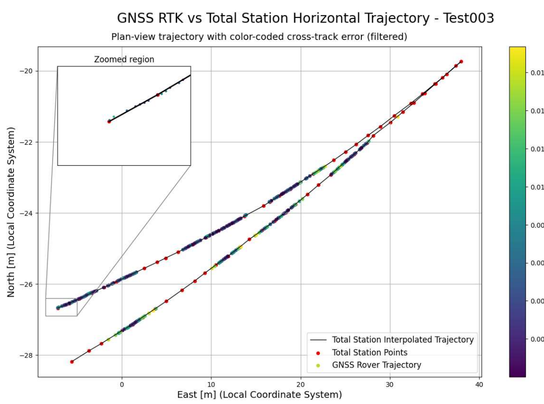


Fig. 32. 2D Traj. Overlay Test003

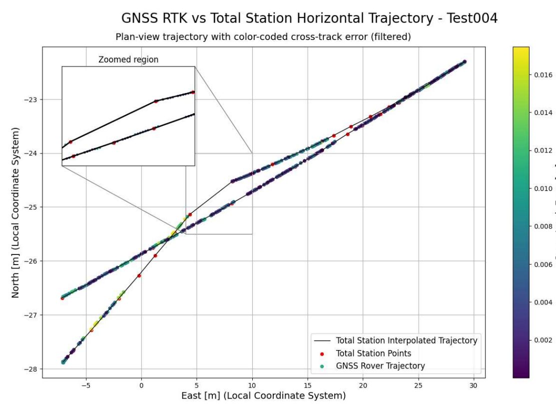


Fig. 33. 2D Traj. Overlay Test004

The error histograms show the distribution of the horizontal position errors for both measurement series. For both datasets, the horizontal positioning accuracy is within the centimeter range. The horizontal RMSE for measurement series 3 is 7.2 mm, whereas for measurement series 4 the RMSE is 7.7 mm. The 95th percentile values are 14.2 mm for measurement series 3 and 13.1 mm for measurement series 4.

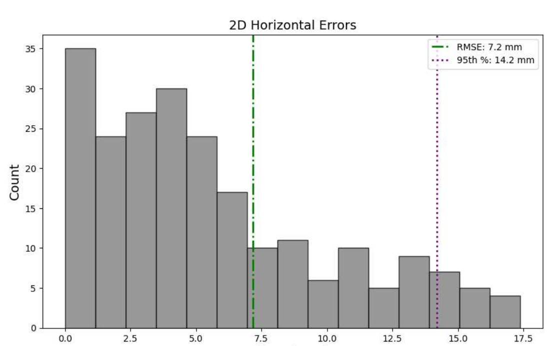


Fig. 34. 2D Error Histogram Test003

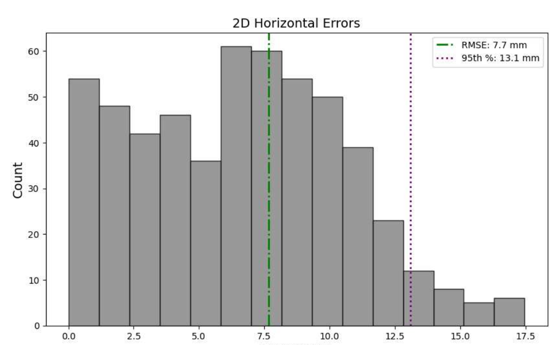


Fig. 35. 2D Error Histogram Test004

Vertical Position Accuracy

This section presents the vertical position accuracy results of the GNSS RTK module, evaluated against the reference trajectory measured with the Leica TS16 total station. The comparison is based on a trajectory overlay in the vertical domain and a statistical analysis of the vertical position errors.

The following two figures illustrate the vertical trajectories of the GNSS RTK module overlaid with the corresponding reference trajectories measured by the total station for measurement series 3 and 4 (Test003 and Test004). The reference trajectory is shown as a linearly interpolated total station trajectory, with discrete total station measurement points also indicated. The GNSS rover trajectory is plotted along the distance moved along the trajectory.

In both measurement series, the GNSS rover trajectory follows the overall shape of the reference trajectory in the vertical dimension with a clear but not always constant offset.

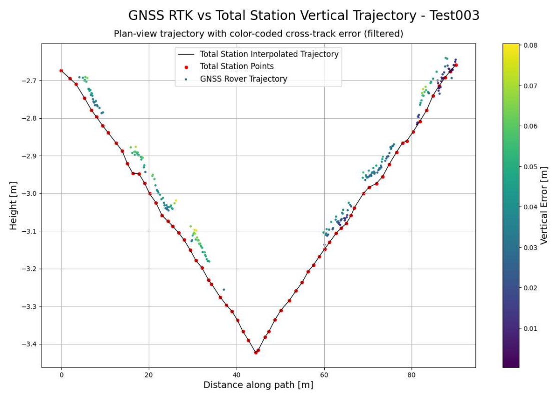


Fig. 36. Vertical Traj. Overlay Test003

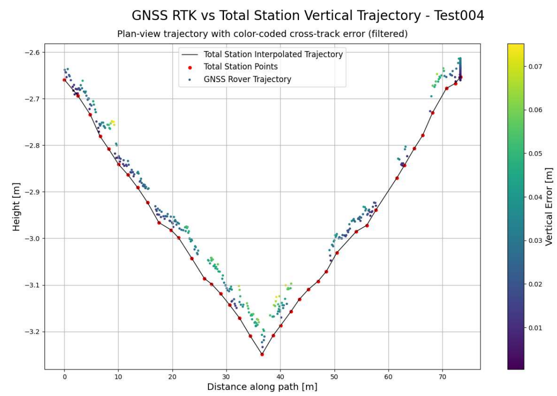


Fig. 37. Vertical Traj. Overlay Test004

The vertical error histograms for measurement series 3 and 4 are shown in Fig. 38 and Fig. 39. The vertical position error is defined as the height difference between the GNSS rover position and the corresponding interpolated reference position. The histograms illustrate the distribution of vertical errors across all rover positions for both measurement series.

For measurement series 3, the vertical root mean square error (RMSE) is 41.0 mm, and the 95th percentile of the vertical error distribution is 64.8 mm. For measurement series 4, the vertical RMSE is 30.2 mm, with a 95th percentile value of 55.4 mm.

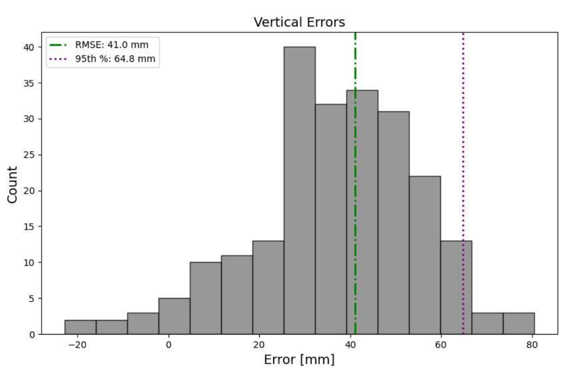


Fig. 38. Vertical Error Histogram Test003

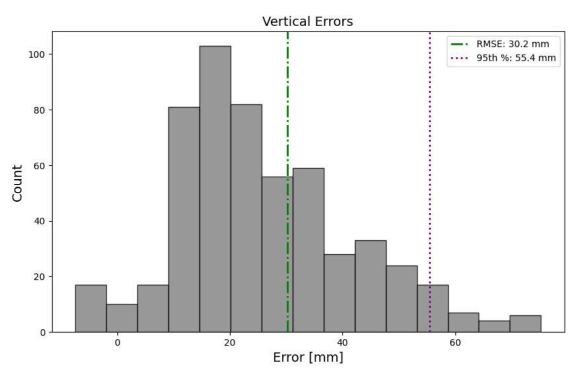


Fig. 39. Vertical Error Histogram Test004

3D Position Accuracy

This section presents the three-dimensional position error results of the GNSS RTK module, evaluated with respect to the reference trajectory measured by the Leica TS16 total station. The 3D position error is defined as the Euclidean distance between the GNSS rover position and the corresponding reference position, combining the horizontal and vertical error components.

Fig. 40 and Fig. 41 show the histograms of the 3D position errors for measurement series 3 and measurement series 4 (Test003 and Test004). The histograms illustrate the distribution of 3D position errors across all evaluated rover positions for both measurement series.

For measurement series 3, the 3D root mean square error (RMSE) is 41.6 mm, and the 95th percentile of the 3D error distribution is 65.2 mm. For measurement series 4, the 3D RMSE is 31.1 mm, with a corresponding 95th percentile value of 55.7 mm.

The histograms show the spread of 3D position errors over the full set of rover observations, with error magnitudes extending to approximately 80 mm for measurement series 3 and approximately 75 mm for measurement series 4.

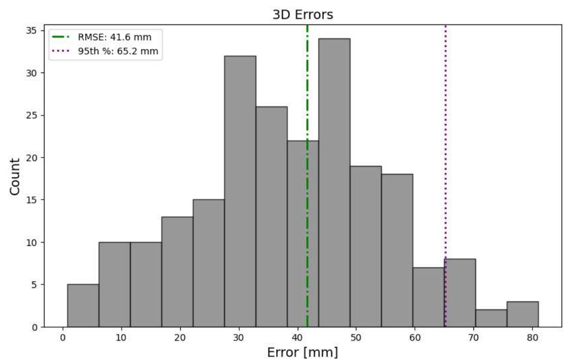


Fig. 40. 3D Error Histogram Test003

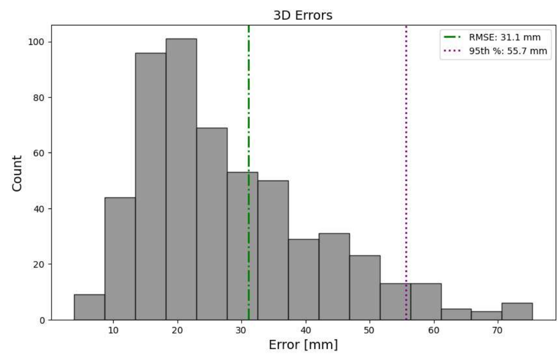


Fig. 41. 3D Error Histogram Test004

Methodology Comparison

This section presents a comparison of the positioning accuracy results obtained using two different error evaluation methodologies for measurement series 5 (Test005). For this measurement series the stop-and-go test case was applied. The comparison is based on the resulting RMSE values for the horizontal, vertical, and three-dimensional (3D) position errors.

The first approach, referred to as Method 1: Path-Based Orthogonal Error Method, evaluates the GNSS rover positions by projecting them orthogonally onto an interpolated reference trajectory and computing the corresponding horizontal, vertical, and 3D errors. Both methods are explained in the section "Positioning Error Evaluation Methods" on page 25.

The RMSE values obtained with both methodologies are summarized for the horizontal, vertical, and 3D position components in the following table and form the basis for the subsequent discussion of methodological differences.

Method	Horizontal RMSE [mm]	Vertical RMSE [mm]	3D RMSE [mm]
Path-Based Orthogonal Error	8.45	20.12	21.82
Nearest-Neighbour	9.54	10.18	13.95

Table 3. Methodology Comparison

Reliability and Availability

Satellite Coverage & Dilution of Precision

This section presents the results of satellite coverage for the conducted measurement series 4 and 8. These two measurement series were selected because they represent, respectively, a best-case and a worst-case scenario in terms of satellite coverage and dilution of precision values. The results of the remaining measurements series are available in the in the Appendix Folder 6, "Raw Data Plots"

Best-Case Scenario:
In the best-case scenario, the data was collected across 1,200 measurement points. The position dilution of precision (PDOP) values were consistently low, with a mean of 1.60 and a median of 1.62, indicating high positional accuracy. The geometric dilution of precision (GDOP) values followed a similar trend, with a mean of 1.90 and a median of 1.92. The standard deviations were small (0.17 for PDOP and 0.23 for GDOP), reflecting a minimal variability in geometric positioning.

The satellite availability was high during this measurement series, with an average of 44.63 satellites visible and 18.32 satellites used. The number of satellites remained relatively stable, with minimum and maximum values of 40 - 47 visible satellites and 12 - 24 used satellites, respectively. Overall, these results demonstrate optimal satellite geometry and reliable positioning accuracy.

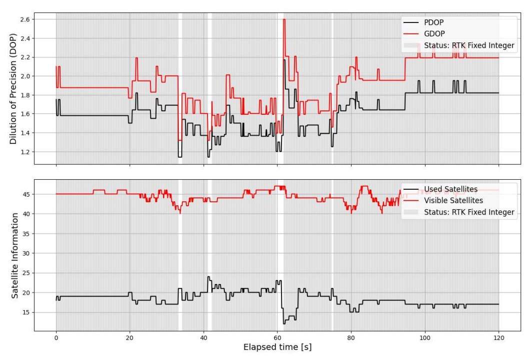


Fig. 42. DOP values and Sat. Information for Test004

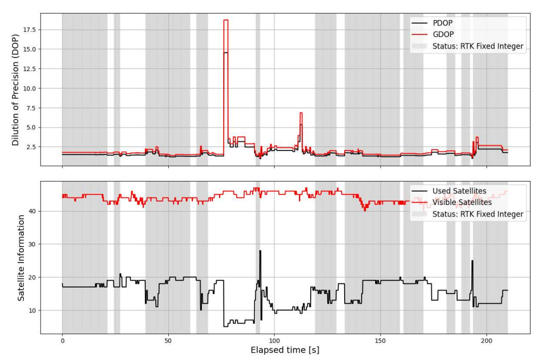


Fig. 43. DOP values and Sat. Information for Test008

Topic	count	mean	median	std	min	max
PDOP	1200	1.60	1.62	0.17	1.14	2.17
GDOP	1200	1.90	1.92	0.23	1.31	2.59
Satellites visible	1200	44.63	45.00	1.31	40.00	47.00
Satellites used	1200	18.32	18.00	1.77	12.00	24.00

Table 4. Statistic Parameters Test004

Topic	count	mean	median	std	min	max
PDOP	2100	1.77	1.50	1.34	0.97	14.55
GDOP	2100	2.11	1.77	1.74	1.12	18.72
Satellites visible	2100	44.07	44.00	1.34	40.00	47.00
Satellites used	2100	15.20	16.00	3.98	5.00	28.00

Table 5. Statistic Parameters Test008

Worst-Case Scenario:
In the worst-case scenario, the data was evaluated across 2,100 observations, both PDOP and GDOP values showed greater variability and higher maximum values, indicating periods of reduced positional accuracy. The mean PDOP increased to 1.77, with a median of 1.50, and the standard deviation rose up to 1.34. Similarly, GDOP values averaged 2.11 (median 1.77) with a standard deviation of 1.74. The maximum observed PDOP and GDOP reached 14.55 and 18.72, respectively, reflecting occasional instances of poor satellite geometry.

Satellite visibility remained relatively consistent, with a mean of 44.07 satellites visible and 15.20 satellites used. However, the number of satellites used show a higher variability (standard deviation of 3.98), ranging from a minimum of 5 to a maximum of 28 satellites. These results indicate that while satellite availability was generally stable, fluctuations in satellite usage and geometry occasionally led to significantly degraded positioning precision.

RTK Correction Service Coverage

This section presents the results of the RTK correction service coverage for the conducted measurement series. The analysis is based on the duration during which an RTK fixed solution was available relative to the total measurement duration, as well as the longest continuous time interval without an RTK fixed solution.

For the best-case scenario, represented by measurement series 4 (Test004), an RTK fixed solution was available for 115.30 s out of a total measurement duration of 119.90 s, corresponding to an RTK fixed coverage of 96.16 %. The longest continuous interval without an RTK fixed solution was 2.00 s.

For the worst-case scenario, represented by measurement series 8 (Test008), the RTK fixed solution duration was 122.20 s over a total measurement duration of 210.00 s, resulting in an RTK fixed coverage of 58.22 %. The longest continuous duration without RTK fixed correction was 26.00 s.

Table 6 summarizes the RTK correction service coverage results for measurement series 3 to 8 (Test003 to Test008). For each measurement series, the total measurement duration, the cumulative duration with RTK fixed solution, the corresponding RTK fixed coverage, and the longest duration without RTK fixed correction are reported.

Measurement Series	RTK Fixed Duration [s]	Total Duration [s]	RTK Fixed Coverage [%]	Longest Duration without RTK correction (fixed) [s]
#3 (Test003)	119.50	179.90	66.43	20.50
#4 (Test004)	115.30	119.90	96.16	2.00
#5 (Test005)	153.40	179.90	85.27	9.00
#6 (Test006)	211.61	269.91	78.40	13.50
#7 (Test007)	160.60	239.90	66.894	16.00
#8 (Test008)	122.20	210.00	58.22	26.00

Table 6. Statistic RTK coverage parameters for all measurement series

In addition to the quantitative results presented above, a trend can be visually identified across the measurement series. By comparing the raw plots of all series, it appears that the availability of RTK corrections may be location-dependent. Specifically, 7 out of 8 measurement series show missing RTK corrections near the furthest east–north locations. Fig. 44 illustrates this trend, with the relevant sections highlighted in red. This suggests that certain geographic positions may experience reduced RTK correction coverage, which could impact positioning accuracy in those areas.

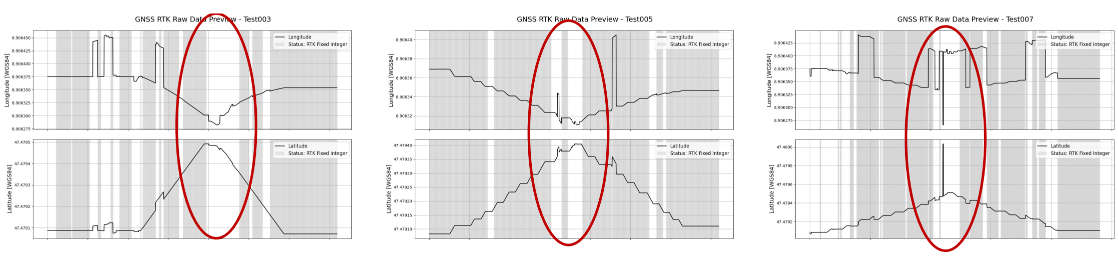


Fig. 44. Measurement series with interrupted RTK service at maxima point

Discussion

Interpretation of Results

Horizontal Position Accuracy:
The horizontal positioning results demonstrate that the GNSS RTK module can achieve centimeter-level accuracy across the evaluated measurement series (Test003, Test004 and Test005) when RTK correction data is available. The low RMSE values and narrow error distributions indicate that the horizontal positioning performance is stable under these conditions.

The close agreement between the GNSS RTK trajectories and the total station reference trajectories suggests that the applied RTK corrections effectively mitigate satellite-related and atmospheric errors in the horizontal plane. The similarity of the 95th percentile values across the different tests further indicates that larger deviations occur infrequently and do not dominate the overall accuracy performance.

These findings confirm that, under the tested conditions, the GNSS RTK system provides reliable horizontal positioning data suitable for applications requiring high horizontal positioning accuracy.

Vertical Position Accuracy:
In contrast to the horizontal component, the vertical positioning results obtained using method 1 show larger error magnitudes and a broader error distribution. The errors are approximately three to four times higher, resulting in a vertical positioning accuracy of about 40 - 50 mm. The observed differences between measurement series indicate that vertical positioning accuracy is more depending on environmental and measurement conditions than the horizontal position accuracy. Despite this increased variability, the vertical RMSE values remain within a 10cm range, demonstrating that the GNSS RTK module maintains a usable level of height accuracy throughout the experiments.

When the stop-and-go test case was applied (Test005) and the position errors were evaluated using Method 2, the vertical errors fell within the same order of magnitude as the horizontal errors. This comparatively low error level is mainly achieved due to the evaluation approach. In method 2, only the best-matching coordinate of the GNSS RTK module is selected for direct deviation comparison. Consequently, for each static phase (approximately 5s) of the stop-and-go test, the algorithm selects the best-matching point from a dataset of roughly 50 GNSS measurements. This behaviour is also evident in Fig. 45, where stacked GNSS RTK data points are visible at each total station measurement instance.

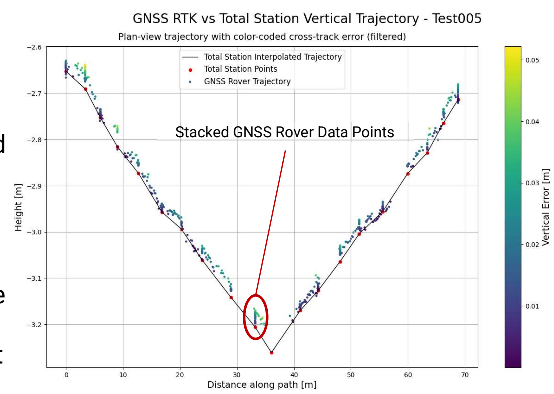


Fig. 45. Accumulated coordinate points

The results highlight the importance of evaluating vertical positioning separately, as vertical errors contribute significantly to the overall 3D positioning accuracy. The differences also demonstrate that vertical accuracy metrics are method-dependent and should be interpreted within the context of the applied evaluation approach.

Three-Dimensional Position Accuracy:
The three-dimensional positioning accuracy reflects the combined effects of horizontal and vertical error components. The 3D RMSE values closely follow the magnitude of the vertical errors, indicating that the vertical component is the dominant contributor to the total positioning error. This relationship and the results of the error evaluation method comparison confirm that improvements in vertical positioning accuracy would have a direct and significant impact on the overall 3D position accuracy. The 3D error distributions show that while most position estimates remain within a few centimeters, larger deviations occur in cases where vertical errors increase. Overall, the 3D accuracy results

demonstrate that the GNSS RTK module delivers consistent spatial positioning performance, with error characteristics largely contributed by the vertical dimension.

Methodology Comparison Results:
The comparison between the Path-Based Orthogonal Error Method (method 1) and the Nearest-Neighbour Method (method 2) reveals that the chosen evaluation methodology has a measurable impact on the vertical positioning accuracy metrics and therefore also on the 3D position accuracy. The Path-Based Orthogonal Error Method results in lower horizontal RMSE values, reflecting its ability to account for the continuous nature of the trajectories and reduce sensitivity to temporal or spatial sampling differences. Conversely, the Nearest-Neighbour Method yields lower vertical and 3D RMSE values. The reason for this is explained in the section above on page on page 38 and illustrated in Fig. 45.

Satellite Coverage:
The satellite coverage results demonstrate that sufficient satellite availability was generally maintained across all measurement series. However, variations in satellite usage had measurable effects on positioning performance. Even when the number of visible satellites remains steady, the subset of satellites effectively contributing to positioning can fluctuate due to differences in satellite geometry and measurement quality, such as signal elevation and range errors. Consequently, PDOP and GDOP values can vary even with a stable visible satellite count [17]–[19]. These findings underline the importance of satellite geometry and constellation availability as fundamental factors influencing GNSS RTK positioning accuracy.

RTK Correction Service Coverage:
The RTK correction service coverage analysis reveals notable variability between measurement series, ranging from near-continuous availability to extended interruptions of RTK fixed solutions. The best-case scenario demonstrates that, under favorable conditions, the correction service can provide nearly uninterrupted RTK fixed positioning. In contrast, the worst-case scenario highlights the impact of correction outages on positioning continuity, with prolonged periods without RTK fixed solutions. These interruptions directly affect the achievable positioning accuracy and reliability, particularly for applications requiring continuous high-precision localization. The visually identified location dependency of RTK coverage could be caused by the RTK base station or the nearby tree or the base station. The results indicate that RTK correction service availability is a critical factor for consistent positioning performance and must be considered alongside intrinsic GNSS accuracy metrics when evaluating system suitability.

Overall Interpretation:
Considering all results together, they show that the GNSS RTK module delivers high horizontal accuracy and reliable three-dimensional positioning under favorable conditions, while vertical accuracy and correction service availability represent the primary limiting factors. The analysis further demonstrates that both environmental conditions and methodological choices influence the reported accuracy metrics. These findings provide a comprehensive basis for assessing the applicability of the GNSS RTK system for mobile robotic platforms and inform potential strategies for system optimization and future improvements.

Limitations

Despite the comprehensive verification of the integrated GNSS RTK module, several limitations must be acknowledged that affect the interpretation of the results.

Reference System and Interpolation:
The total station measurements serve as the reference trajectory for accuracy evaluation. While the Leica TS16 system provides high-precision positioning, the reference trajectory is derived from discrete measurement points and subsequently interpolated. As a result, the reference path represents an approximation of the true rover trajectory, particularly in regions with curvature or changes in motion dynamics. This limitation affects all error metrics derived from the interpolated reference trajectory.

Dependence on RTK Correction Availability:
The reported positioning accuracy is strongly dependent on the availability of RTK correction data and the presence of an RTK fixed solution. Periods without RTK fixed status were completely excluded from accuracy evaluation. Consequently, the presented accuracy metrics primarily reflect system performance under favourable correction conditions and do not represent positioning behaviour during extended correction outages.

Error Evaluation Methodology:
Two different error evaluation methodologies were applied in this work, namely a path-based orthogonal error method and a Nearest-Neighbour Method. The results show that the choice of methodology has a measurable impact on the reported accuracy metrics. Therefore, the obtained RMSE values are not universally comparable to results derived using alternative evaluation approaches, and methodological consistency is required when comparing positioning performance across studies.

Filtering of Outliers:
To reduce the influence of large deviations and measurement artifacts, defined error filtering criteria were applied to the GNSS rover data. While this approach improves the robustness of the statistical analysis, it may reduce the representation of rare but potentially relevant error events. As a result, the reported accuracy metrics emphasize typical positioning performance rather than worst-case behaviour.

Limited Environmental Diversity:
The measurement campaigns were conducted within a limited set of environmental conditions and trajectories. Factors such as dense urban environments, severe multipath conditions, dynamic obstructions, or extreme satellite constellation geometries were not systematically evaluated. Consequently, the results may not be directly transferable to different operational environments.

Vertical Accuracy Sensitivity:
The vertical positioning results exhibit greater variability compared to the horizontal component. This behaviour is consistent with known GNSS characteristics but also highlights a limitation in achieving uniform accuracy across all spatial dimensions. As the vertical error contributes significantly to the overall 3D positioning accuracy, applications with strict height accuracy requirements may require additional sensor fusion or correction strategies.

These limitations provide context for the presented results and identify areas for further investigation and system improvement.

Comparison with Expectations

This section compares the experimentally obtained results with the expected outcomes formulated prior to data analysis. The expectations were derived from known GNSS RTK characteristics, satellite geometry indicators, and the test condition and environment.

The expectation of high-quality GNSS RTK data, reflected by centimeter-level differences between the GNSS rover positions and the total station reference, is largely confirmed by the results. When an RTK fixed solution was available, the horizontal positioning accuracy consistently remained within the centimeter range, supported by low RMSE values and narrow error distributions. This confirms that the monitored DOP values and satellite information provided a reliable indication of data quality during the experiments.

The expected continuity of the RTK correction service is only partially confirmed by the experimental results. While some measurement series showed near-continuous RTK fixed availability, others showed extended interruptions in correction service. These variations directly influenced the stability and reliability of the positioning results, particularly for applications requiring continuous high-precision localization. The observed behaviour highlights the importance of considering RTK correction availability as a key performance factor alongside positioning accuracy.

The reliability of the reference measurements was also confirmed. The total station's automated tracking system maintained continuous lock on the rover, enabling the generation of a consistent reference trajectory without any data gaps. This supports its suitability as a high-precision reference for GNSS RTK performance evaluation.

The expectation of improved accuracy using the stop-and-go measurement method is partially supported by the results. The analysis shows that the primary differences between the various approaches result from the error evaluation method. However, the stop-and-go measurement method, when combined with the nearest-neighbour error evaluation, provides more accurate 3d positioning results due to the longer sampling time at each position. This makes it particularly suitable for collecting the dataset required for the coordinate transformation.

As anticipated, horizontal positioning accuracy exceeded vertical accuracy across all evaluated measurement series. The vertical error component exhibited larger magnitudes and greater variability, consistent with typical GNSS performance characteristics and its dominant influence on the three-dimensional positioning error.

The influence of satellite geometry on positioning accuracy is also reflected in the results. Periods with favorable satellite configurations and stable RTK fixed solutions were associated with consistent positioning performance, while interruptions in RTK corrections led to increased variability.

Environmental factors such as fog and moderate temperature variations did not lead to systematic degradation of GNSS RTK performance, as expected. However, occasional localized deviations suggest that transient disturbances, such as nearby moving objects, can introduce short-term noise in the positioning data.

Overall, the experimental results largely confirm the predefined expectations and validate the suitability of the chosen test design. Deviations from the expected behaviour provide additional insight into real-world system limitations rather than contradicting the initial assumptions.

Conclusion and Recommendations

This project demonstrated that the implemented GNSS RTK module is suitable for deployment on a mobile robotic platform, such as the Husarion Panther. The system provides reliable localization data under favorable conditions, and diagnostic metrics, including covariance estimates and dilution of precision (DOP) values, were shown to be valuable indicators for assessing data quality and positioning accuracy.

The experimental results confirm that high horizontal positioning accuracy can be achieved when an RTK fixed solution is available. However, the verification of the module also revealed that GNSS RTK data cannot be directly integrated into a robotic localization and navigation stack by simply subscribing to the corresponding data topics. Instead, a more sophisticated real-time data handling and validation approach is required to ensure that only plausible and accurate GNSS measurements are used. This requirement is particularly relevant in safety-critical environments, where inaccurate or unstable positioning estimates may pose risks to people, infrastructure, or equipment.

The analysis further highlights the sensitivity of GNSS RTK performance to environmental conditions and correction service availability. Interruptions in RTK corrections and variations in satellite visibility directly affect positioning reliability, especially in the vertical and three-dimensional components. These findings emphasize the importance of robust system design that accounts for intermittent data quality rather than assuming continuous high-precision availability.

Based on the results and identified limitations, several directions for future work are recommended. First, further investigation of dynamic environmental disturbances is suggested to improve system robustness. Scenarios involving pedestrians, vehicles, or other moving objects passing close to the rover or the RTK base station could provide valuable insight into the system’s behaviour under realistic operational conditions. The knowledge gained from such experiments could be used to refine data validation and filtering algorithms.

Second, an assessment of location-dependent GNSS RTK data quality is recommended. A suitable test scenario would involve the robot navigating randomly through environments with varying characteristics, such as narrow corridors with tall buildings, areas near tree lines, or locations in close proximity to high-voltage power lines. Continuous recording of RTK correction status and diagnostic metrics would enable the generation of a spatial “heat map” of positioning quality, supporting the identification of areas with degraded performance and informing deployment strategies. Such a test could provide valuable insights for real-world deployments, where accurate localization is critical for safety.

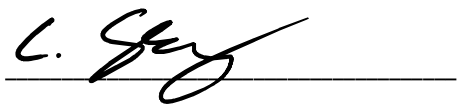
In addition to methodological improvements, several hardware and software enhancements are recommended. Running the GNSS RTK node directly on the robot’s onboard user computer would reduce system complexity, save space and weight, lower energy consumption, and free an additional interface on the network router. Furthermore, replacing the current 3D-printed terminal box with a machined aluminum enclosure would improve mechanical robustness and ensure full ingress protection, thereby increasing system reliability for long-term outdoor operation.

Overall, the results of this work provide a solid foundation for the integration of GNSS RTK systems into mobile robotic platforms and outline clear pathways for improving robustness, reliability, and operational safety in future developments.

Authorship Declaration

I hereby declare that this work has been carried out independently and without any unauthorized assistance. I have used only the sources and tools that I have explicitly cited. All ideas, data, or text taken from external sources—whether directly or indirectly—have been clearly acknowledged. This work has not been submitted, either in whole or in part, to any other examination authority, institution, or publisher, and has not been published previously. I am aware of and have adhered to the ethical standards applicable to academic research and writing.

Place, Date Wil, 30th December 2025

Signature 

Chris Stolz

OST
Eastern Switzerland
University of Applied Sciences

Appendix

References

[1] Calian Advanced Technologies, “GNSS Positioning Techniques,” Calian, Accessed: Oct. 10, 2025. [Online]. Available: https://www.calian.com/advanced-technologies/gnss/information-support/gnss-positioning-techniques/

[2] The 3rd Dimension, “Land surveying 101 - How accurate is RTK GPS,” YouTube, Dec. 26, 2023. [Online]. Available: https://www.youtube.com/watch?v=ENWCF81DgR8&t=445s. [Accessed: Sep. 12, 2025].

[3] u-blox AG, “Real-Time Kinematic (RTK) and GNSS corrections,” u-blox, Mar. 12, 2025. [Online]. Available: https://www.u-blox.com/en/technologies/rtk-real-time-kinematic. [Accessed: Sep. 01, 2025].

[4] Septentrio-GNSS, septentrio_gnss_driver, GitHub, 2025. [Online]. Available: https://github.com/septentrio-gnss/septentrio_gnss_driver (accessed: Aug. 10, 2025).

[5] “Helmert transformation,” Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Helmert_transformation. [Accessed: Dec. 9, 2025].

[6] “Kabsch algorithm,” Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Kabsch_algorithm. [Accessed: Dec. 9, 2025].

[7] Geodätisches Institut, Universität Karlsruhe (TH), «*GPS und Integration von GPS in bestehende geodätische Netze*», Sonderheft 38. Jahrgang der Mitteilungen des Deutschen Vereins für Vermessungswesen (DVW), Karlsruhe, Germany, 1993.

[8] Husarion, “Panther-5 robot,” Husarion. [Online]. Available: https://husarion.com/assets/images/panther-5-medium-0187b508c426f99ab3af82150c89dd75.png. Accessed: Sep. 10, 2025.

[9] Calian Advanced Technologies, “GNSS Positioning Techniques,” Calian, 2025. [Online]. Available: https://www.calian.com/advanced-technologies/gnss/information-support/gnss-positioning-techniques/. [Accessed: Oct. 10, 2025].

[10] “SimpleRTK3B-X5,” ArduSimple. [Online]. Available: https://www.ardusimple.com/product/simplertk3b-x5/ (accessed Nov. 30, 2025).

[11] “AS-ANT2B-SUR-L1L2-25SMA-00,” ArduSimple (Mouser). [Online]. Available: https://www.mouser.ch/ProductDetail/ArduSimple/AS-ANT2B-SUR-L1L2-25SMA-00?qs=vHuUswq2%252BswBsit1qMDm7Q%3D%3D … (accessed Nov. 30, 2025).

[12] “Radio Module – Extra Long Range,” ArduSimple. [Online]. Available: https://www.ardusimple.com/product/radio-module-extra-long-range/ (accessed Nov. 30, 2025).

[13] “Raspberry Pi 5 8GB Entwicklungsboard Kit,” Digitec. [Online]. Available: https://www.digitec.ch/de/s1/product/raspberry-pi-5-8gb-entwicklungsboard-kit-38955607 (accessed Dec. 1, 2025).

[14] “Dipole antenna for LR/XLR radio,” ArduSimple. [Online]. Available: https://www.ardusimple.com/product/dipole-antenna-for-lr-xlr/ (accessed Dec. 1, 2025).

[15] “Unitree 4D LiDAR L2,” Unitree. [Online]. Available: https://shop.unitree.com/products/unitree-4d-lidar-l2?srsltid=AfmBOoq5UWZ5ddLfMnpJvlEv8zV5t_tJhSRDaQ4rXFaDikO26ZymYpCW (accessed Dec. 1, 2025).

[16] “Team,” ISF Institute for Intelligent Systems and Smart Farming, OST. [Online]. Available: https://www.ost.ch/en/research-and-consulting-services/technology/system-technology/isf-institute-for-intelligent-systems-and-smart-farming/team (accessed Dec. 1, 2025).

[17] M. Tahsin, J. Taher, and M. S. Islam, “Analysis of DOP and its preciseness in GNSS position estimation,” in Proc. Int. Conf. Electrical Engineering and Information Communication Technology (ICEEICT), Dhaka, Bangladesh, May 2015, pp. 1-6. [Online]. Available: https://ieeexplore.ieee.org/document/7307445, doi: 10.1109/ICEEICT.2015.7307445.

[18] L. Siqueira and J. F. Galera, “Weighted Dilution of Precision (WPDOP) with consideration of range errors and elevation of GNSS satellites during high ionospheric activity,” in Proc. 34th Int. Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+), St. Louis, MO, USA, Sep. 2021, pp. 1596-1605. [Online]. Available: https://www.ion.org/publications/abstract.cfm?articleID=17890, doi: 10.33012/2021.17890.

[19] D. H. Won, J. S. Kim, and S. J. Lee, “Weighted DOP with consideration on elevation-dependent range errors of GNSS satellites,” IEEE Trans. Instrum. Meas., vol. 61, no. 12, pp. 3241-3250, Dec. 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6237526, doi: 10.1109/TIM.2012.2205512.

Tools used

The following tools were used in the preparation, configuration, and editing of this project: Visual Studio Code, NX Siemens, OpenAI ChatGPT, GitHub, GitLab, Microsoft Teams, Microsoft Word, Swisstopo reframe tool, Septentrio configuration tool

Project Management

Stakeholder Overview

The following section provides an overview of the project's stakeholders and their responsibilities and interests.

Fig. 46: Prof. Dr. Dejan Šeatović [16]

Contact & Personal Information:	
Full name:	Prof. Dr. Dejan Šeatović
Position / Responsibility:	Head of ISF / Advisor
Phone:	+41 58 257 47 04
Mail:	dejan.seatovic@ost.ch

Responsibilities:

- Project sponsor
  - Definition of scope and deliverables
- Examination

Interests:

- Successful project according to scope and time line definition.

Fig. 47: Luis Meier [16]

Contact & Personal Information:	
Full Name:	Luis Meier
Position:	Sr. Development Engineer at ISF
Phone:	+41 58 257 42 14
Mail:	luis.meier@ost.ch

Responsibilities:

- Review / Approval / Release
- Consulting / Support
- Supervision
- Examination

Interests:

- Enable MSE Student

Fig. 48: Chris Stolz

Contact & Personal Information:	
Full Name:	Chris Stolz
Position:	MSE Student
Phone:	+41 78 622 26 54
Mail:	chris.stolz@ost.ch

Responsibilities:

- Project lead
- Development and implementation
- Verification
- Documentation

Interests:

- On-time delivery of project deliverables
- Knowledge acquisition in the field of ROS2
- Knowledge acquisition in the field of GNSS RTK
- Knowledge acquisition in systems engineering

Project Timeline and Milestones:

The following figure shows the project plan with the most relevant milestones. In order to successfully develop, implement and verify the individual components.

The so called V-Model was applied for this project. The V-Model can be used for guidance and ensures that all system requirements are verified.

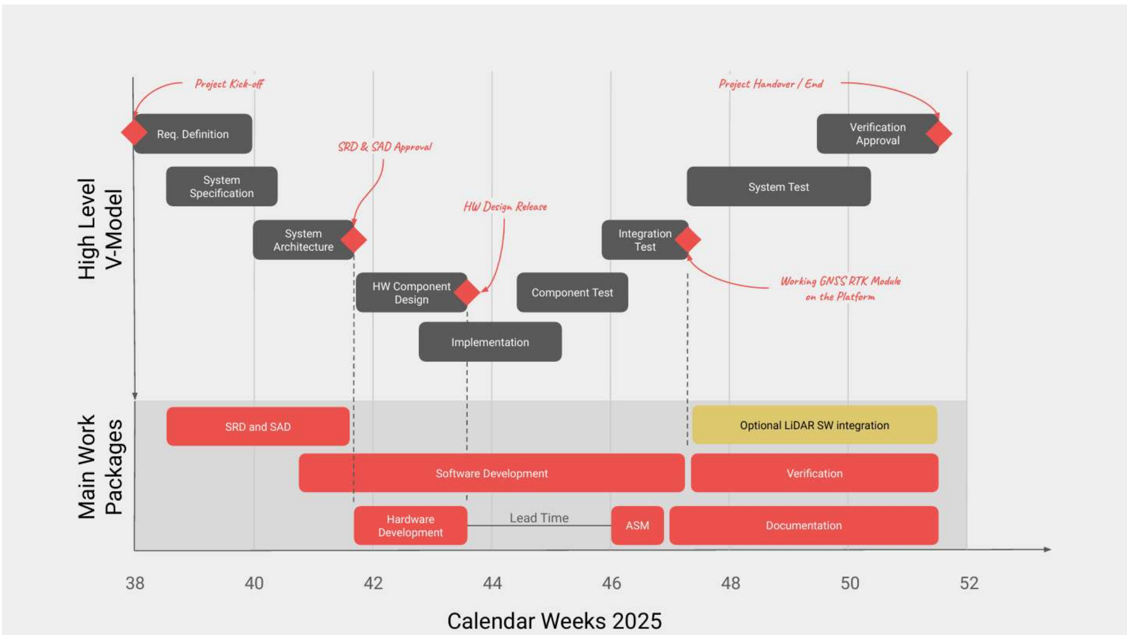


Fig. 49: Project Plan

Appendix Folder

1. System Requirements Document
2. CAD
3. Data Sheets
4. URDF Parameters
5. Verification Data
6. Raw Data Plots
7. Meeting Notes