



Enabling the Galileo high accuracy service with open-source software: integration of HASlib and RTKLIB

Fabricio S. Prol¹ · Martti Kirkko-Jaakkola¹ · Oliver Horst² · Tuomo Malkamäki¹ · M. Zahidul H. Bhuiyan¹ · Sanna Kaasalainen¹ · Ignacio Fernández-Hernández³

Received: 28 September 2023 / Accepted: 13 January 2024 / Published online: 13 February 2024
© The Author(s) 2024

Abstract

The Galileo high accuracy service (HAS) is a free-of-charge service designed to deliver decimeter-level accuracy in real-time precise point positioning (PPP) applications using global navigation satellite systems (GNSS). With the intention of facilitating the use of HAS corrections with open access tools, we present the open-source library named HASlib and its integration with another open-source library named RTKLIB. HASlib decodes the Reed–Solomon encoded Galileo E6 navigation data pages and outputs the corrections in commonly used formats. This enables the utilization of HAS with conventional GNSS receivers and PPP engines that lack native support for HAS formats. For instance, the outputs from HASlib enable the use of HAS corrections in RTKLIB. In order to validate this integration, we demonstrate that HAS can allow decimeter-level accuracy using only free-of-charge services and tools. We have obtained a 3D root mean square error below 20 cm (1 sigma) after a convergence time of 10–90 min in Finland. This accuracy has overcome classical real-time solutions with broadcast and satellite-based augmentation system (SBAS) data by one order of magnitude. Compared to post-processed multi-GNSS PPP, HAS corrections required longer convergence times, given the real-time nature. Furthermore, our assessment revealed that the longer convergence time, compared to prior literature, was attributed to RTKLIB filtering procedure and geometry deficiencies in high latitudes. Nevertheless, once convergence was attained, a sub-decimeter level of accuracy was observed in both horizontal and vertical components. These findings highlight the effectiveness of Galileo HAS, HASlib, and RTKLIB as powerful tools for providing open-access to real-time PPP solutions.

Keywords GNSS · Galileo · Open-source · Precise point positioning

Introduction

GNSS receivers must apply corrections to compensate for deterministic errors, such as satellite orbit and clock, to reach decimeter-level positioning accuracy. This is typically performed in real time by connecting to a third-party receiver or a real-time kinematic (RTK) network, allowing relative positioning with differential GNSS (DGNSS) techniques. Those techniques require a separate data link (e.g., cellular connectivity) and often a service subscription. For

instance, the international GNSS service (IGS) provides real-time access to precise orbit and clock corrections via the Internet (Elsobeiey and Al-Harbi 2016). To eliminate the need for a separate data link, classic broadcasted ephemeris and satellite-based augmentation system (SBAS) messages are viable real-time correction options. However, the performance guarantees of SBAS and classical broadcast messages exceed one meter, as demonstrated by Ali et al. (2012), Montenbruck et al. (2015), Pan et al. (2017), and Park et al. (2020). As an alternative to broadcast and SBAS messages, GNSS corrections are also delivered by commercial services through geostationary (GEO) satellites, requiring a service subscription.

In recent years, the Galileo High Accuracy Service (HAS) has been introduced as a free-of-charge service. It is designed to enable stand-alone GNSS receivers with no separate data links to achieve 20 cm and 40 cm accuracy in horizontal and vertical positions (95%), respectively (European

✉ Fabricio S. Prol
fabricio.dossantosprol@nls.fi

¹ Department of Navigation and Positioning, Finnish Geospatial Research Institute, Espoo, Finland

² Kinexon Industries, Munich, Germany

³ European Commission, Brussels, Belgium

Union 2023). To achieve this goal, Galileo HAS broadcasts real-time correction data for Galileo and Global Positioning System (GPS) on the E6 signal. The signal is not encrypted, implying that any E6-capable receiver can benefit from the HAS free-of-charge. Alternatively, if the user terminal has internet connectivity, Galileo HAS also provides an internet-based service. Early Galileo HAS tests have shown that the service can already offer the expected service accuracy when Galileo is combined with GPS. Fernández-Hernández et al. (2022) conducted the first assessment of the performance of Galileo HAS. Their findings demonstrated that the system provides extensive coverage while achieving a few-centimeter broadcast correction accuracy. Later, Naciri et al. (2023) obtained a positioning accuracy better than 20 cm (95% confidence level interval) in kinematic mode. Martini et al. (2023) also carried out a positioning performance assessment, assuming a static case and a road environment with obstructions. Results of the static case showed an accuracy better than 3 cm and 8 cm (1-sigma) in horizontal and vertical components, respectively. The dynamic test revealed a horizontal positioning error of 26.7 cm (1-sigma) in environments with degraded visibility, which was four times superior to that of a code-based solution. The benefits of Galileo HAS are also extended to precise orbit determination (POD) of satellites in low earth orbit (LEO). Hauschild et al. (2022) made significant improvements in 3D orbit determination of Sentinel-6A by incorporating HAS corrections into their near-real-time processing. They achieved an orbit determination error of approximately 7–9 cm, which is an advancement compared to the method relying solely on broadcast ephemerides corrections, which yielded a less accurate accuracy of 17–20 cm.

The Galileo HAS corrections are designed to have global validity for precise point positioning (PPP) assisted by state space representation (SSR) corrections. Despite Galileo HAS is centered on PPP, the corrections can be implemented in single-point positioning (Angrisano et al. 2023) or incorporated into differential techniques. However, as opposed to DGNSS, where Radio Technical Commission for Maritime Services (RTCM) standards are widely supported, PPP does not have a globally adopted standard for the provision of HAS correction data. The standardization of the RTCM SSR format has been pending for several years. Alternatives have been proposed in the meantime; however, many of them are very similar to RTCM SSR. Galileo HAS corrections are

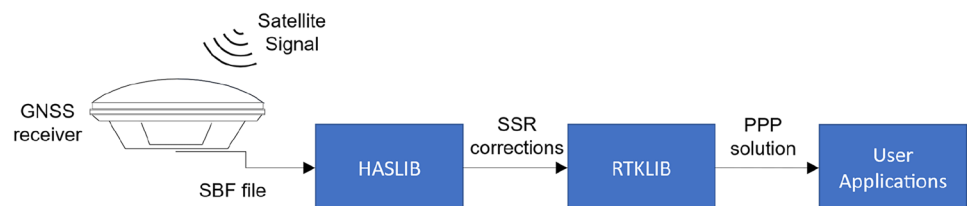
provided in their own format, reminiscent of the compact SSR format (Hirokawa et al. 2016). Furthermore, they are encoded into so-called high-parity vertical Reed Solomon (HPVRS) codes for optimizing the HAS message reception from multiple satellites (Fernández-Hernández et al. 2020). As a result, legacy GNSS receivers and PPP processing software that solely depend on GNSS data links are unlikely to be able to benefit from the HAS without a software upgrade.

To facilitate the use of Galileo HAS with legacy receivers and software, the HASlib package was developed by Horst (2021), as a contribution to the European Commission's Precise and Authentic User Location Analysis (PAULA) project (Horst et al. 2022). While some HAS decoders have been developed to analyze HAS results, for example, in Hauschild et al. (2022), open-source alternatives to HASlib have appeared only recently (Borio et al. 2023). HASlib is an open-source application that decodes the HAS HPVRS-encoded navigation data frames and converts the corrections to either RTCM-SSR or IGS-SSR. The HAS corrections generated from HASlib can be input to any GNSS receiver or PPP processing software that supports RTCM-SSR or IGS-SSR correction data streams, even if they do not natively support HAS. We validate the implemented package by showing a proof-of-concept result of processing Galileo HAS corrections with the open-source RTKLIB package. RTKLIB (Takasu 2013) was selected as it can provide real-time PPP solutions when aided by RTCM 3 messages. We aim to show the potential of Galileo HAS, HASlib, and RTKLIB to provide free-of-charge real-time solutions with high accuracy. In this regard, the next section describes the context of the open-source services. The proof-of-concept test is shown in Results section. The final conclusions are then outlined.

Open-source HAS decoder and PPP engine

Sub-meter GNSS positioning requires the availability of a service to provide correction data. Such data can be expressed in various formats, requiring a processing engine supporting the provided correction data. The adopted positioning services, data formats and used open-source library are presented in this section. Figure 1 shows the workflow to obtain the PPP solutions. GNSS receivers gathered data in the Septentrio binary (SBF) file format. Subsequently, the

Fig. 1 End-to-end architecture to access PPP solutions based on Galileo HAS and open-source software



collected data were processed through HASLib, which transforms the input file into SSR corrections. These SSR corrections are transmitted with a recognized standard by RTKLIB and serve as the foundation for computing PPP solutions to various forthcoming user applications. We utilized SBF files to construct a post-processing dataset, simplifying our investigation and facilitating the creation of a supplementary document. However, in real-time applications, HASLib must be applied with GNSS data streams through transmission control protocol (TCP) or serial ports. When applied in real time, our setup demonstrated the capability to generate an end-to-end result in around 2 s per epoch.

Open-source HAS decoder package

Three main principles drove the design of HASLib. First, the implementation must not consume too many computational resources to allow real-time operation on various platforms. Second, it should be portable, not dependent on commercial

software or libraries, as well as easy to use and integrate into existing projects and workflows. Third, the library should be straightforward to configure or modify. To achieve this, the library was designed for easy adoption by emphasizing modularity, code readability and easy extension through a clear interface. The modular structure enables a straightforward reuse of the whole library or its parts in the future. Different interfacing methods were developed for input and output data: streams, files, and TCP connection. The input data can be read from streams and files in SBF and binary exchange format (BINEX) (UNAVCO 2020). The output modes support two of the most meaningful representations, RTCM-SSR (RTCM 2013) or IGS-SSR (IGS 2020). To accommodate future needs, adding new input and output formats is also designed to be as straightforward as possible.

The top-level architecture of the decoder library is illustrated in Fig. 2, followed by Table 1, which describes the most important HASlib classes and their functionalities. The HASlib decoder is publicly available on the GitHub platform

Fig. 2 HASlib high-level architecture and data flow

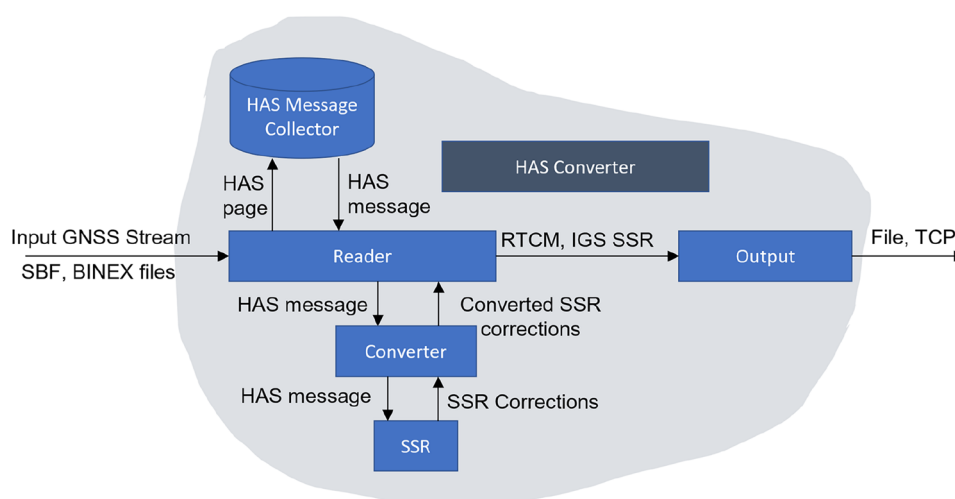


Table 1 Description of the most important HASlib classes and their functionalities

Class	Description
HAS converter	The HAS converter class is an all-encompassing interface. This library consolidates access to all the other classes and provides a clean and simple user interface to most of the required functions and subroutines
Reader	The Reader class handles the input streams and parsing data for the processing routines. SBF and BINEX format data can be read either from file, serial port, or TCP server
HAS message collector	The HAS message collection class is used to temporarily store the received HAS message pages. This class is also responsible for the sorting and handling of HAS messages, including storing and parsing of received masks and decoding the messages when enough pages have been acquired
SSR	The SSR class is a quasi-universal structure allowing the extension of the library to support SSR formats. This class enables storing all SSR information, such as clock and orbit corrections, satellite signal biases
Converter	The converter class is used to convert HAS messages to other SSR formats
Output	The Output class is used to format the converted messages for different output types. Each output type has its own class. A class designed specifically for the interfacing to the PPPWizard (Laurichesse and Privat 2015) library is included. In the case of RTKLIB, one can use the RTCM 3 output files

(National Land Survey of Finland 2022) and released under the European Union Public License (EUPL) with a copyright owned by the European Union.

As part of the HAS Message Collector class, an important step of HASlib is the optimization of the time needed to download data. To this end, HASlib exploits the availability of several satellites by decoding the messages in a Reed–Solomon scheme. The HAS corrections are received from several Galileo satellites simultaneously at a rate of 448 bits per second and per satellite (European Union 2023). The Reed–Solomon encoding distributes the information content of 32 data pages over a total of 255 redundant encoded pages. The original message length, k , is dependent on the amount of information transmitted in the HAS message. Each page consists of 424 bits of information. In order to decode a single message consisting of $k \leq 32$ pages, the receiver needs to receive any k different encoded pages to recover the information. In other words, the amount of information to be downloaded remains constant. Note that in practice, HAS messages can be shorter than 32 pages, in which the remaining pages are filled with zeros. If the messages are transmitted in the optimal way, this scheme enables an efficient dissemination of data (Senni et al. 2022).

Reed–Solomon encoding (Reed and Solomon 1960) is a linear operation in a Galois field. Consequently, both encoding and decoding operations can be expressed as matrix multiplication inside this field. Let c represent the 32-element message vector. Then, it can be encoded with the generator matrix \mathbf{G} to the code vector $\mathbf{\Gamma}$ of length 255 as (European Union 2023):

$$\mathbf{\Gamma} = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{31} \\ \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_{222} \end{bmatrix} = \begin{bmatrix} c \\ \gamma \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ \mathbf{P} \end{bmatrix} c = \mathbf{G} c \quad (1)$$

where the generator matrix \mathbf{G} is composed of blocks \mathbf{I} and \mathbf{P} . The top block \mathbf{I} is an identity matrix of size 32×32 , implying that the first 32 elements of the 255-page sequence are the original HAS message pages in clear text. The bottom block \mathbf{P} is a dense matrix with size 223×32 , and it corresponds to the remaining 223 parity pages.

This formulation makes the decoder straightforward to implement without the need for specialized algorithms. First, the received encoded pages are collected according to the page identifier which can be found in the clear-text page header. Then, once the necessary number k of different pages has been received, the message bytes are vertically grouped into words, and the decoding matrix \mathbf{G}^{-} is formed as a $k \times k$

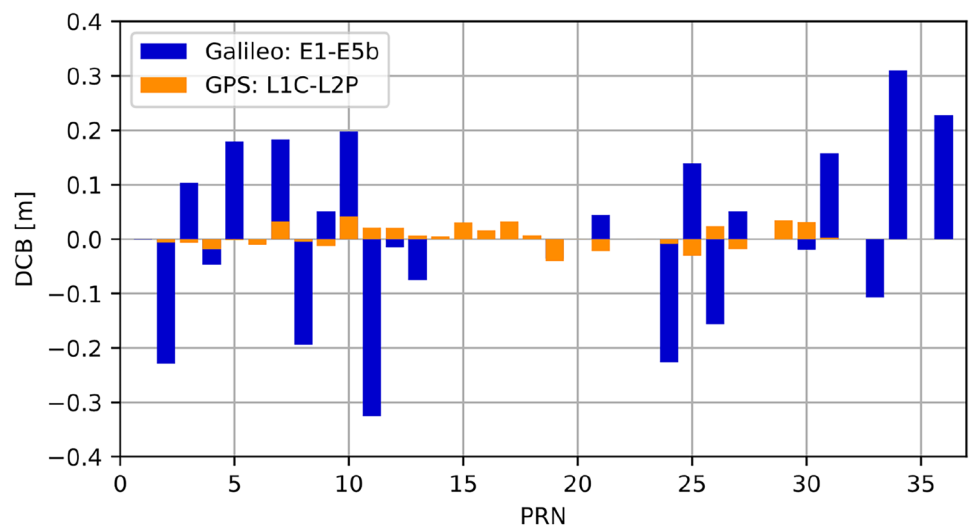
matrix with the corresponding rows of \mathbf{G} in the same order. The obtained matrix \mathbf{G}^{-} is invertible, and it is applicable to each vertical word; consequently, one matrix inversion is sufficient to decode the entire message. Reed–Solomon encoding and decoding, including the definition of the generator matrix \mathbf{G} , is explained in more detail in the interface control document (European Union 2023).

Open-source PPP software package

There are diverse options for PPP processing, ranging from receiver built-in features to commercial and open-source software. One of the most popular open-source packages for precise GNSS processing is RTKLIB. Its first release took place in 2006 and it is still being developed, allowing PPP processing with a cm-level accuracy (Prol et al. 2018). The current stable RTKLIB version (v. 2.4.3 b34) only supports the finalized RTCM SSR messages, implying that there is no support for PPP with the Galileo HAS without internet access. HASlib is used to cover this gap, enabling the conversion of Galileo HAS messages to RTCM 3. In this scheme, the satellite orbit and clock corrections provided by Galileo HAS messages are employed by RTKLIB. However, RTKLIB currently lacks the capability to incorporate the transmitted code biases. Therefore, we are not utilizing the HAS code biases in the RTKLIB PPP results. Instead, we let the code biases be absorbed mainly by the carrier phase ambiguity term and receiver clock. Details of how to employ the absolute code biases are shown in Villiger et al. (2019). As for the phase biases, Galileo HAS is not providing them yet, so they cannot be used in our PPP solution, and therefore no integer ambiguity resolution can be performed. For more details, Schaer et al. (2021) illustrate the potential advantages of incorporating phase bias in PPP implementations.

To exemplify the magnitude of the corrections not employed by RTKLIB, Fig. 3 shows the differential code bias (DCB) values computed based on Galileo HAS messages, which requires the code bias corrections by default. Note that these unapplied code biases, while up to some decimeters, do not introduce a similar error in the PPP solution as they are compensated by the filter as part of the float carrier phase ambiguity term and the receiver clock. Therefore, they do not lead to a commensurate error in the position solution. To substantiate this point, we refer to Villiger et al. (2019), who emphasize the impracticality of distinguishing between clock corrections and the associated bias terms. Moreover, the IGS real-time working group (IGS 2020) acknowledges that biases may be excluded in PPP with SSR messages because they cannot be differentiated from phase ambiguities in certain situations. This limitation is applicable only to cases where ambiguity resolution is not intended and for short durations, as the inter-signal biases (ISB) might not remain constant over time. Additionally,

Fig. 3 DCB values related to the signal biases of the ionosphere-free combination determined from the Galileo HAS messages on February 8, 2023, from 13:00 to 16:00 UT



from our perspective, code biases are necessary for positioning if the user processes measurements different from those of the network. For example, code biases are necessary if the user processes 1C/2P while the network uses 1W/2W. However, if both the network and the user use the same measurements, code biases are not needed for positioning, although they may be necessary for timing. Given these considerations, appropriate configurations are necessary in RTKLIB to overcome the absence of code bias corrections. This includes the estimation of float ambiguities and utilizing identical measurements used in satellite clock and signal bias estimation.

Table 2 presents the configurations employed in RTKLIB. As these options are specifically tailored for real-time applications, they effectively reflect the potential of PPP in field campaigns. Users can rely on these configurations to obtain accurate and reliable results during their operations with Galileo HAS, HASlib, and RTKLIB. However, it is essential to note that the authors have not applied any specific tuning to enhance the PPP algorithm. Consequently, this study showcases the inherent accuracy of RTKLIB, which may not fully reflect the accuracy of HAS corrections and expected convergence time, given that the RTKLIB software is not specifically developed to use Galileo HAS. RTKLIB utilizes the extended Kalman filter (EKF) with undifferenced observations to compute the solutions in PPP modes. Despite utilizing a static station, a kinematic mode is selected to evaluate the convergence time with a dynamic PPP solution. In the static mode, only three unknowns are defined for the coordinates, whereas the kinematic mode incorporates three unknowns with each epoch. This approach enables a more precise assessment of the time required to achieve a stable solution for most user applications.

The L1/L2 frequencies are used for GPS, while for Galileo, the E1/E5b frequencies are employed. The two

Table 2 RTKLIB configuration settings for real-time PPP aided by Galileo HAS corrections

Feature	RTKLIB options
Positioning mode	PPP kinematic
Frequencies	L1 + L2/E1 + E5b
Filter type	Forward
Elevation mask	7°
Earth tides	Solid dynamics
Ionosphere	Iono-Free LC
Troposphere	Estimated ZTD
Satellite ephemeris	Broadcast + SSR APC
Clock corrections	Broadcast + SSR APC
Receiver PCV	ANTEX file IGS14
Phase wind-up	On (Wu et al. 1993)
Code bias	Not supported
Phase bias	Not supported
Integer ambiguity resolution	Off
Constellation	GPS/Galileo
Observation sampling	1 s

frequencies enable the ionospheric-free (IF) linear combination (LC). The estimation of zenith tropospheric delay (ZTD) is performed during the PPP process aided by the Saastamoinen model (1973) and Niell mapping function (Niell 1996). Considering that the visibility is reduced at the high latitudes in Finland, a low cut-off angle of 7° is defined. The used antenna phase center (APC) models are based on the IGS corrections released in the antenna exchange format (ANTEX) files. To account for the Galileo HAS corrections, the satellite orbit and clock are estimated using the broadcast ephemeris and corrected by the HAS-based SSR messages referring to the APC location. No strategies are selected for ambiguity solutions as code bias corrections are not applied.

Results

The PPP assessment is conducted based on Galileo HAS data collected by raw GNSS measurements and HAS navigation data frames on five representative days between February 8 and 13, 2023. The data collection was conducted by a commercial off-the-shelf Septentrio PolaRx 5 receiver in Finland. The collected data in SBF format were processed by HASlib and converted to RTCM 3 messages. The functionality of the open-source services is validated by parsing the output of HASlib to the RTKLIB package. The interfaces are then applicable to generate accurate PPP results in real time.

Four configurations in RTKLIB are evaluated and compared. Table 3 summarizes all the configurations. These configurations were defined to represent the positioning modes and constellations that users will most likely apply in real-case experiments. They were applied to two stations, named rooftop and MPAS (Fig. 4). The rooftop station is in the premises of the Finnish Geospatial Research Institute (FGI) in Espoo, Finland (60.2° N, 24.8° E). The MPAS station is a monitoring station situated 6.2 km from the rooftop station. The station is positioned on top of a building near Helsinki city center within an urban environment characterized by numerous surrounding buildings and structures. The reference location of both stations was

determined by the AUSPOS online GPS processing service v.2.3 (Geoscience Australia 2021) with final IGS corrections and daily GNSS data.

Figure 5 illustrates the results in the rooftop station with an arc of three hours on February 8, 2023 (13:09–16:10 UTC). We did not include the PPP-IGS results in this analysis, as they exhibit comparable accuracy to PPP-HAS at the scale of Fig. 5. A dedicated comparison between the two methods is provided subsequently for enhanced visualization. By visual inspection in Fig. 5, we have identified that enabling HAS corrections allows to overcome other exclusively GNSS-based solutions by an order of magnitude. SPP-BRDC and SPP-SBS provided a meter-level accuracy, while PPP-HAS allowed centimetric accuracy. The PPP-HAS, therefore, demonstrates a clear advantage over typical positioning strategies that lack external data links beyond the GNSS-provided information. It must be noticed that the first minutes of PPP-HAS are related to the convergence time required for a stable solution. Furthermore, we have made the dataset and a comprehensive guideline for replicating this experiment available in the Supplementary document.

To further explore the capabilities of Galileo HAS in the rooftop station, Fig. 6 shows a comparison of the real-time PPP HAS solution (PPP-HAS-GPS/GAL) against a post-processed PPP solution, which includes final and accurate PPP corrections processed by IGS in a multi-GNSS

Table 3 Positioning modes employed for comparison against PPP with Galileo HAS messages

Positioning mode	Details
SPP-BRDC-GPS/GAL/GLO/BDS	Single point positioning (SPP) is applied using the broadcast ephemeris data for all GNSS constellations with the Klobuchar (1987) model for the ionosphere. Single-frequency data are used together with least-squares, representing the default GNSS solution
SPP-SBS-GPS	SPP is applied using the broadcast ephemeris data along with SBAS corrections for GPS
PPP-HAS-GPS/GAL	PPP aided by Galileo HAS messages is applied with no external data link other than GPS and Galileo. Dual-frequency data are used together with a Kalman filter, representing the default Galileo-HAS solution
PPP-IGS-GPS/GAL/GLO/BDS	PPP is applied using the final corrections provided by IGS for all GNSS constellations. Dual-frequency data are used together with a Kalman filter, representing a reference for comparison

The two first modes are convenient for positioning in various GNSS applications as they allow positioning solutions without additional corrections or real-time data streams. The last option is included as a PPP solution based on the most accurate and up-to-date information available in post-processed mode. The configuration used with the positioning mode ‘PPP-HAS-GPS/GAL’ is equal to that of Table 2

Fig. 4 Environment surrounding the rooftop and MPAS stations

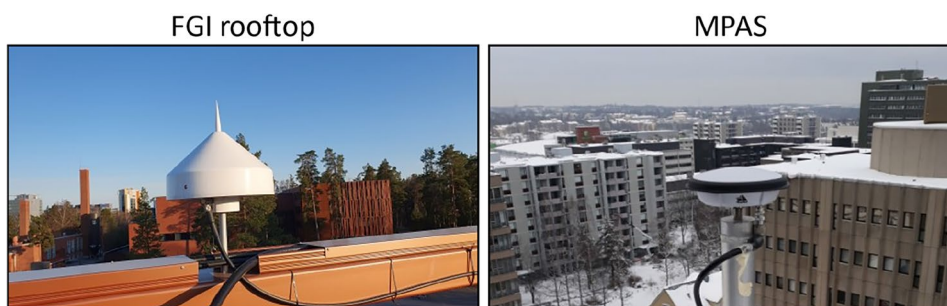


Fig. 5 Comparison of positioning solutions using real-time GNSS corrections broadcasted by the GNSS satellites. BRDC denotes the classic broadcast messages provided by GPS, Galileo, GLONASS, and Beidou, while SBS represents SBAS messages for GPS. PPP solutions were exclusively enabled by HAS corrections

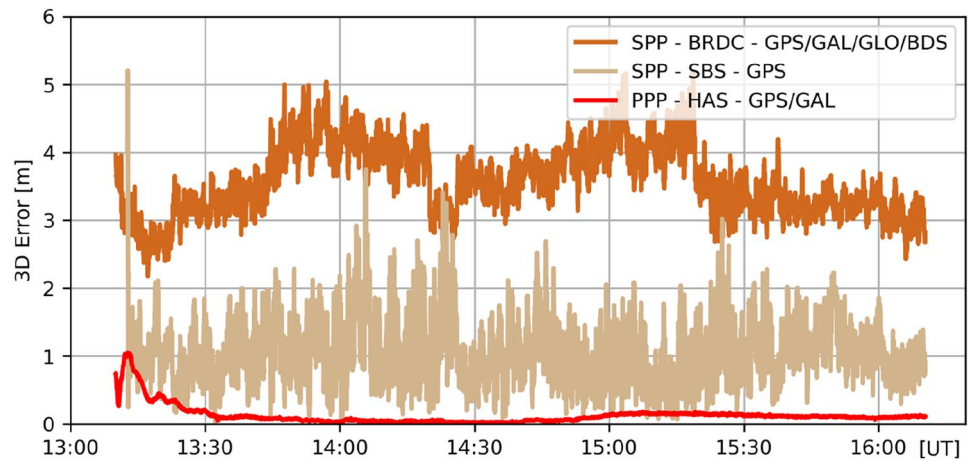
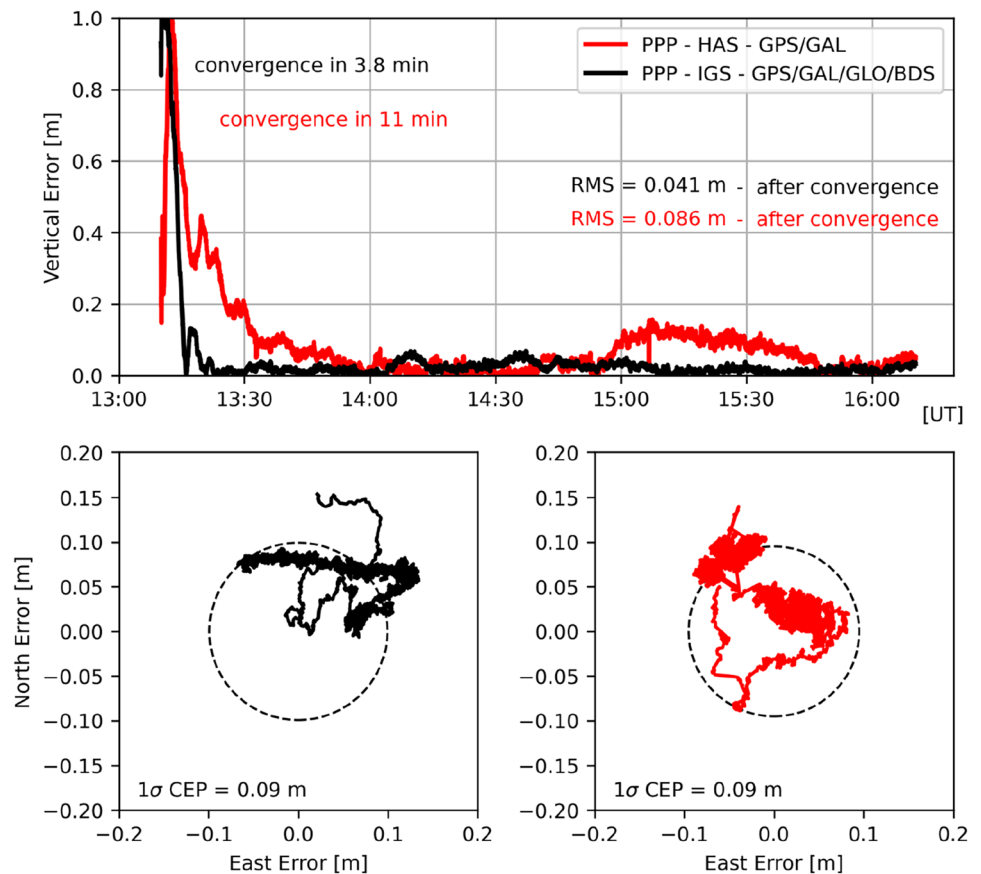


Fig. 6 Vertical errors (top panel) and horizontal errors (bottom panels) of the rooftop station. PPP was carried out with two different correction sources: HAS corrections (shown in red) and final IGS corrections (shown in black). HAS corrections have been derived directly from the GNSS receivers in real time and processed with GPS and Galileo systems. IGS corrections, which include final SP3 products from GPS, Galileo, GLONASS, and Beidou, have been obtained days after data collection. The PPP processing strategies utilized both open-source codes from HASlib and RTKlib



environment (PPP-IGS-GPS/GAL/GLO/BDS). Again, these results are related to arcs of three hours on February 8, 2023. We have adopted the horizontal and vertical convergence times as the duration needed to achieve and stabilize the solution below 20 cm and 40 cm for horizontal and vertical accuracy, respectively. These threshold values align with the study conducted by Naciri et al. (2023) and the Galileo target performance metrics established by EUSPA (2020). A convergence time of around 11 min was obtained using

Galileo HAS corrections, while IGS final products allowed a 3.8-min convergence time. We assume the convergence criterion is met when the horizontal and vertical components cross the predefined threshold values.

The top panel in Fig. 6 shows the vertical error, while the bottom panels show the horizontal error. The RMS was computed considering all values after the convergence time. In the vertical component, the Galileo HAS corrections allowed an RMS error (1-sigma) of 8.6 cm after convergence time,

while the post-processed PPP enabled 4 cm accuracy. In this particular case, the predominant factor influencing convergence time was the vertical component. In the horizontal

component, both solutions have shown a 1-sigma Circular Error Probable (CEP) of 9 cm after the convergence time, with an overall bias to the north direction. A bias in the horizontal component is expected within the post-processed PPP as the GNSS receiver is located at a high latitude (over 60°N in geographical latitude) and the GNSS coverage does not fully cover the Northern region, as shown in the sky plot example of Fig. 7. Although the largest errors in the HAS corrections are affecting the vertical component, we observe a comparable and biased level of accuracy in the horizontal component.

Figure 8 illustrates the PPP results obtained at the MPAS station. This example serves to reinforce the notion that the performance achieved at the rooftop station extends to other test cases. As demonstrated, the accuracy of PPP HAS remains around 10 cm, which is similar, but slightly inferior, to post-processed PPP. Additionally, this example requires a longer convergence time. The solution consistently hovers around the convergence time threshold (below 40 cm) for several minutes, nearly reaching 40 cm within the first 10 min. However, only after 20 min did the accuracy surpass the threshold. Similarly, the rooftop station also experiences resistance, with the vertical error remaining close to 40 cm for a brief period. However, the rooftop station accuracy reaches below 40 cm two times faster. The primary factor contributing to the lower accuracy in MPAS is the

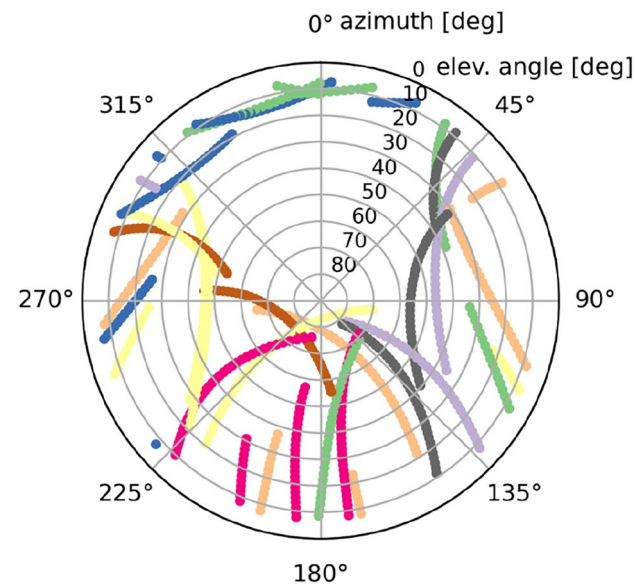
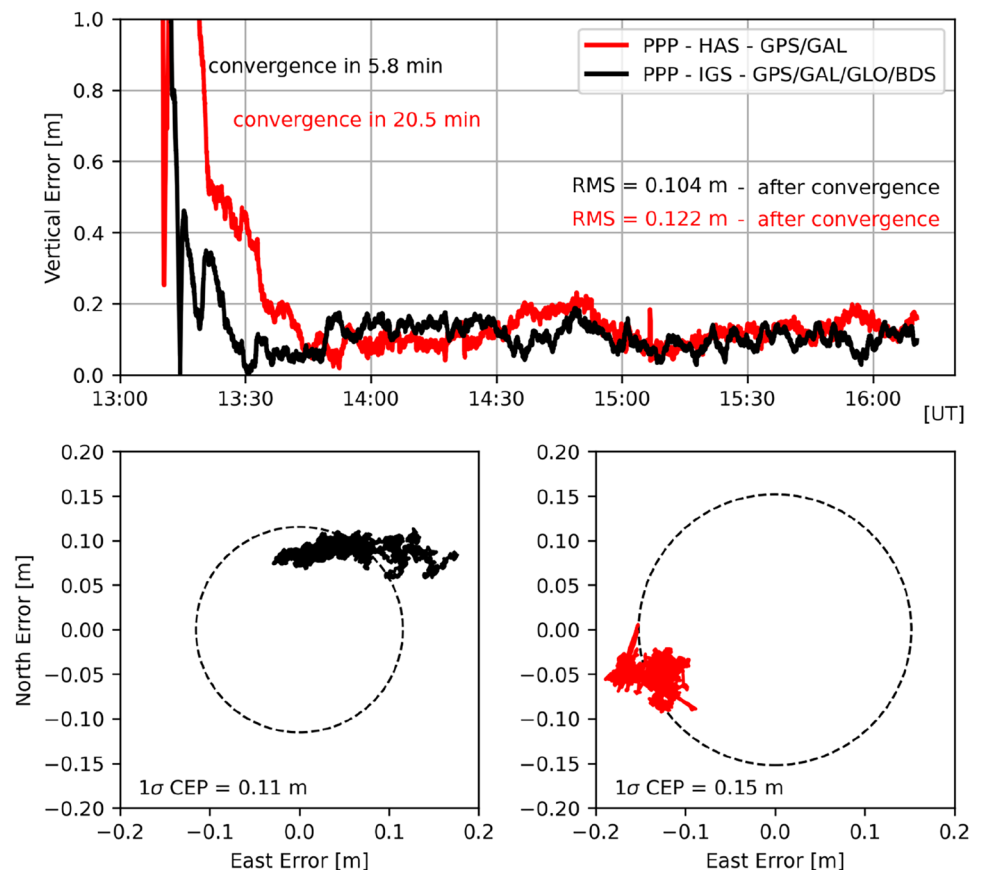


Fig. 7 GPS and Galileo polar coordinates observed by the rooftop station collected during a three-hour data arc on February 8, 2023

Fig. 8 Same as Fig. 6, with the exception that it represents the MPAS station, located 6.2 km away from the rooftop station



challenging urban environment, which is highly susceptible to signal multipath and obstruction. Both solutions (HAS PPP and post-processed PPP) present worse accuracy in MPAS when compared to the rooftop.

Figure 9 shows the rooftop results when 32 arcs of three hours are formed on February 9–12, 2023. Again, we show results based on PPP with HAS corrections and PPP with final IGS corrections. Both methods are capable of providing positioning accuracy within the expected thresholds below 20 cm and 40 cm for horizontal and vertical components, respectively. However, in this case, the convergence time was considerably larger. Although the vertical component could converge to below 40 cm within an average time span of 5 min, the horizontal component took an average of

01:30 h to converge to below 20 cm with Galileo HAS and 15 min with IGS corrections.

Figure 10 (bottom panels) illustrates the accuracy of HAS corrections in the cases analyzed in Fig. 9. The top panels of Fig. 10 display the accuracy of HAS corrections used to generate Fig. 6, surpassing the precision of Fig. 9. In Fig. 10 (bottom panel), HAS corrections exhibit an accuracy level of approximately ± 0.25 m, with an average error centered at -0.5 cm and a standard deviation of ± 0.22 m (2 sigma). In contrast, the top panel shows an average error centered at zero and a standard deviation of ± 0.19 m (2 sigma). This suggests that more accurate HAS corrections improve PPP accuracy and reduce convergence time. The performance of the HAS corrections

Fig. 9 PPP error of RTKLIB aided by HASlib at the rooftop station. The PPP solution is computed using several three-hour arcs spanning the period from February 9–12, 2023. The shadowed colored areas in the figure represent the standard deviation

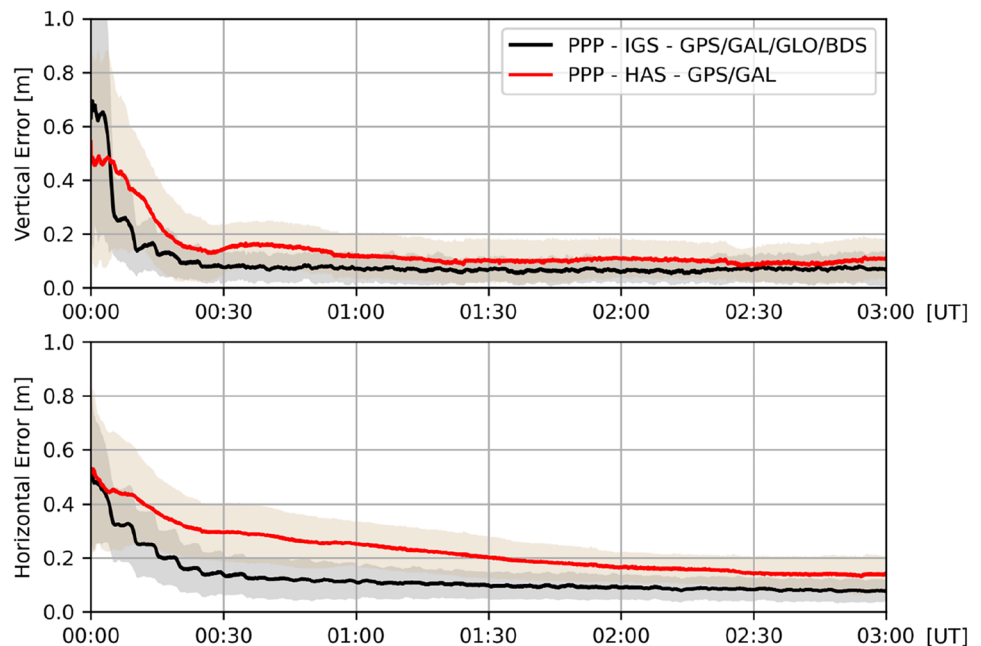


Fig. 10 Accuracy of orbital and clock data from the HAS corrections in relation to IGS final orbit files in the 3-h period of February 8, 2023 (top panels) and from February 9–12, 2023 (bottom panels). Each color represents a different satellite. The histogram is obtained merging both orbit and clock errors in bins of 1 cm

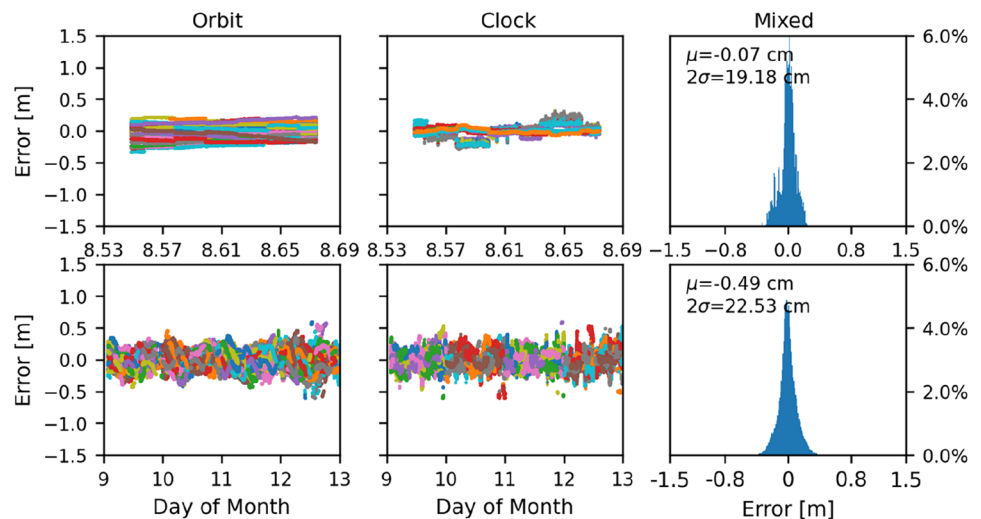


Table 4 Absolute mean (μ), standard deviation (σ), and RMS of the horizontal and vertical errors obtained using each positioning mode in the rooftop station

Positioning mode	Horizontal $\mu \pm \sigma$ (m)	Vertical $\mu \pm \sigma$ (m)	Horizontal RMS 95% (m)	Vertical RMS 95% (m)
SPP-BRDC	3.898 ± 0.406	3.265 ± 0.957	3.919	3.402
SPP-SBS	0.497 ± 0.270	0.909 ± 0.774	0.565	1.193
PPP-HAS	0.107 ± 0.028	0.095 ± 0.061	0.121	0.155
PPP-IGS	0.095 ± 0.017	0.058 ± 0.025	0.096	0.063

around ± 0.25 m compares well against prior research (Fernández-Hernández et al. 2022; Naciri et al. 2023). However, the convergence time in Fig. 9 is longer than reported in literature. Martini et al. (2024) achieved a 15-min overall convergence time in static mode using GPS and Galileo data, while Naciri et al. (2023) reported a 7.5-min convergence time in PPP-HAS with kinematic mode at 33 IGS stations using GPS and Galileo observations. To justify the prolonged convergence time, we attribute it to the fact that the PPP engine was designed to use HAS corrections specifically in these previous studies, whereas RTKLIB was initially developed for broader and more generic applications. Additionally, our solution faced challenges in maintaining accuracy below the 20 cm horizontal threshold in Finland due to geometric deficiencies. Positioning components with the lowest observational redundancy absorbed a significant portion of the orbital errors, contributing to the prolonged convergence time in the horizontal component.

Table 4 summarizes the average accuracy achieved in all days analyzed. The evaluation performance is presented in terms of absolute bias, standard deviation, and RMS. In each processing mode, the statistics were calculated based on results obtained after the PPP-HAS solution reached convergence. Like the analysis in Fig. 5, the SPP processing strategy with broadcast ephemeris presents the worst solution, with an absolute error of 3 m and a standard deviation of about 0.7 m in both horizontal and vertical components. It is interesting to observe that SPP with SBAS corrections exhibits a similar standard deviation but achieves a better average error, around 0.7 m. This outcome highlights the effectiveness of SBAS solution in mitigating systematic errors and maintaining the solution accuracy within the level of the code measurement noise. As for the PPP results, centimeter-level accuracy is achieved when utilizing both HAS and post-processed corrections after the convergence time. An important aspect accentuated by these findings is the relatively higher standard deviation observed in the PPP with HAS corrections, especially in the vertical component, as compared to post-processed PPP.

Summary and conclusions

We presented an integration of HASlib and RTKLIB to provide open-source access to Galileo HAS. The HASlib has been written in Python and it decodes the HPVRS-encoded HAS correction messages and outputs them in either the RTCM 3 or the IGS SSR format. In this first release, the supported input formats are SBF and BINEX. The library architecture was designed to be easily extensible to other input and output formats, with particular interest in RTCM 3, which is adopted by RTKLIB.

The experimental results validate that the HASlib interfaces are compatible with the open-source RTKLIB package, enabling the HAS corrections to improve the positioning precision by one order of magnitude over classical real-time solutions with broadcasted and satellite-based augmentation data. In comparison to post-processed multi-GNSS PPP, a comparable level of accuracy was achieved in the horizontal component, with longer convergence time and worse accuracy in the vertical component, but still under 20 cm (1-sigma). In total, HASlib makes it possible to use Galileo HAS with receivers and software that do not support HAS outside of the box. Nevertheless, our investigation has identified a longer convergence time in RTKLIB for enabling PPP solutions, primarily attributed to the software's filtering together with geometry-related issues in high latitudes. More extensive testing and verification of the library is foreseen, especially considering future developments of RTKLIB and Galileo HAS.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s10291-024-01617-7>.

Acknowledgements This work was conducted under the Precise and Authentic User Location Analysis (PAULA) project, funded by the European Commission DG-DEFIS under contract DEFIS/2020/OP/0002. This work was also supported by the INdoor Navigation from CUBesAT Technology (INCUBATE) project through the Technology Industries of Finland Centennial Foundation and the Jane and Aatos Erkko Foundation (JAES).

Author contributions FSP and MK-J conceptualized and designed the study, and were responsible for conducting the experiments. FSP primarily authored the main manuscript text, prepared Figures, and

conducted the data analysis. MK-J, OH, and TM contributed to the development of HASLib. IF-H, TM, and MZHB provided supervision and oversight of the study's findings. SK and IF-H secured funding for the project. All authors reviewed and approved the final manuscript.

Funding Open Access funding provided by National Land Survey of Finland.

Data availability The data from the rooftop station on February 08, 2023, can be found in the supplementary document. The GNSS data for the MPAS station are publicly accessible on the FINPOS data center website (<https://finpos.nls.fi/>). The remaining GNSS data are available from the corresponding author upon request.

Declarations

Conflict of interest The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ali K, Pini M, Dosis F (2012) Measured performance of the application of EGNOS in the road traffic sector. *GPS Solut* 16:135–145. <https://doi.org/10.1007/s10291-011-0253-5>
- Angrisano A, Ascione S, Cappello G, Gioia C, Gaglione S (2023) Application of “Galileo high accuracy service” on single-point positioning. *Sensors* 23:4223. <https://doi.org/10.3390/s23094223>
- Geoscience Australia (2021) Auspos online GPS processing service. <https://gnss.ga.gov.au/auspos>. Accessed 30 Jun 2022
- Borio D, Susi M, Gioia C (2023) GHASP: a Galileo HAS parser. *GPS Solut* 27:197. <https://doi.org/10.1007/s10291-023-01529-y>
- Elsobeiey M, Al-Harbi S (2016) Performance of real-time Precise Point Positioning using IGS real-time service. *GPS Solut* 20:565–571. <https://doi.org/10.1007/s10291-015-0467-z>
- European Union (2023) Galileo high accuracy service: service definition document (HAS SDD). Issue 1.0. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo-HAS-SDD_v1.0.pdf
- EUSPA (2020) Galileo high accuracy service (HAS) info note. European GNSS Agency. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_HAS_Info_Note.pdf
- Fernández-Hernández I, Senni T, Borio D, Vecchione G (2020) High-parity vertical Reed-Solomon codes for long GNSS high-accuracy messages. *Navigation* 67(2):365–378. <https://doi.org/10.1002/navi.357>
- Fernández-Hernández I, Chamorro-Moreno A, Cancela-Díaz S, Calle-Calle JD, Zoccarato P, Blonski D, Senni T, Blas FJ, Hernández C, Simón J, Mozo A (2022) Galileo high accuracy service: initial definition and performance. *GPS Solut* 26:65. <https://doi.org/10.1007/s10291-022-01247-x>
- European Union (2021) Galileo High Accuracy Service signal-in-space interface control document (HAS SIS ICD). Issue 1.0. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_HAS_SIS_ICD_v1.0.pdf
- Hauschild A, Montenbruck O, Steigenberger P, Martini I, Fernández-Hernández I (2022) Orbit determination of Sentinel-6A using the Galileo high accuracy service test signal. *GPS Solut* 26:120. <https://doi.org/10.1007/s10291-022-01312-5>
- Hirokawa R, Sato Y, Fujita S, Miya M (2016) Compact SSR messages with integrity information for satellite based PPP-RTK service. In: *Proceedings of ION GNSS 2016*, Institute of Navigation, Portland, Oregon, USA, September 12–16, pp 3372–3376
- Horst O, Kirkko-Jaakkola M, Malkamäki T, Kaasalainen S, Fernández-Hernández I, Moreno AC, Díaz SC (2022). HASlib: an open-source decoder for the galileo high accuracy service. In: *Proceedings of the ION GNSS 2022*, Institute of Navigation, Denver, Colorado, USA, September 19–23, pp 2625–2633
- Horst O (2021) Implementation of an open-source software suite for the Galileo High Accuracy Service. M.Sc. (Tech.) thesis, Aalto University, School of Electrical Engineering, Espoo, Finland
- IGS (2020) IGS state space representation (SSR) format. https://files.igs.org/pub/data/format/igs_ssr_v1.pdf. Accessed 29 Jun 2023
- Klobuchar JA (1987) Ionospheric time-delay algorithm for single-frequency GPS users. *IEEE Trans Aerosp Electron Syst* AES-23:325–331. <https://doi.org/10.1109/TAES.1987.310829>
- Laurichesse D, Privat A (2015) An open-source PPP client implementation for the CNES PPP-WIZARD demonstrator. In: *Proceedings of the ION GNSS 2015*, Institute of Navigation, Tampa, Florida, USA, September 14–18, pp 2780–2789
- Martini I, Susi M, Cucchi L, Fernández-Hernández I (2024) Galileo high accuracy service performance and anomaly mitigation capabilities. *GPS Solut* 28:25. <https://doi.org/10.1007/s10291-023-01555-w>
- Montenbruck O, Steigenberger P, Hauschild A (2015) Broadcast versus precise ephemerides: a multi-GNSS perspective. *GPS Solut* 19:321–333. <https://doi.org/10.1007/s10291-014-0390-8>
- Naciri N, Yi D, Bisnath S, Blas FJ, Capua R (2023) Assessment of Galileo High Accuracy Service (HAS) test signals and preliminary positioning performance. *GPS Solut* 27:73. <https://doi.org/10.1007/s10291-023-01410-y>
- National Land Survey of Finland (2022) HASlib. <https://github.com/nlsfi/HASlib>
- Niell AE (1996) Global mapping functions for the atmosphere delay at radio wavelengths. *J Geophys Res* 101:3227–3246. <https://doi.org/10.1029/95JB03048>
- Pan L, Cai C, Santerre R, Zhang X (2017) Performance evaluation of single-frequency point positioning with GPS, GLONASS, BeiDou and Galileo. *Surv Rev* 49:197–205. <https://doi.org/10.1080/00396265.2016.1151628>
- Park KW, Park J-I, Park C (2020) Efficient methods of utilizing multi-SBAS corrections in multi-GNSS positioning. *Sensors* 20:256. <https://doi.org/10.3390/s20010256>
- Prol FS, Camargo PO, Monico JFG, Muella MTAH (2018) Assessment of a TEC calibration procedure by single-frequency PPP. *GPS Solut* 22:35. <https://doi.org/10.1007/s10291-018-0701-6>
- Reed IS, Solomon G (1960) Polynomial codes over certain finite fields. *J Soc Ind Appl Math* 8(2):300–304. <https://doi.org/10.1137/0108018>
- RTCM (2013) Differential GNSS (Global Navigation Satellite Systems) services—version 3. RTCM Standard 10403.2, Radio Technical Commission for Maritime Services, Arlington, VA, with Amendments 1 and 2
- Saastamoinen J (1973) Contributions to the theory of atmospheric refraction, part II: refraction corrections in satellite geodesy. *Bull Geod* 107:13–34. <https://doi.org/10.1007/BF02522083>
- Schaer S, Villiger A, Arnold D, Dach R, Prange L, Jäggi A (2021) The CODE ambiguity-fixed clock and phase bias analysis

products: generation, properties, and performance. *J Geod* 95:81. <https://doi.org/10.1007/s00190-021-01521-9>

Senni T, Fernández-Hernández I, Cancela S (2022) Evaluation of the Hungarian algorithm for optimal transmission of the Galileo HAS message from multiple satellites. In: *Proceedings of the international conference on localization and GNSS*, Tampere, Finland

Takasu T (2013) RTKLIB: an open source program package for GNSS positioning. <http://www.rtklib.com/>. Accessed 27 June 2022

UNAVCO (2020) BINEX: binary exchange format. <https://www.unavco.org/data/gps-gnss/data-formats/binex/binex.html>. Accessed 28 June 2022

Villiger A, Schaer S, Dach R, Prange L, Sušnik A, Jäggi A (2019) Determination of GNSS pseudo-absolute code biases and their long-term combination. *J Geod* 93:1487–1500. <https://doi.org/10.1007/s00190-019-01262-w>

Wu JT, Wu SC, Hajj GA, Bertiger WI, Lichten SM (1993) Effects of antenna orientation on GPS carrier phase. *Manuscr Geodaet* 18:91–98

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Fabricio S. Prol received the PhD degree in cartographic sciences from the São Paulo State University, São Paulo, Brazil, with a focus on geodetic remote sensing and geodetic positioning in 2019. He worked as a Research Scientist in the German Aerospace Center (DLR) from 2019 until 2021. He has been working with the Finnish Geospatial Research Institute, Helsinki, Finland, since 2021. His research interests include ionospheric modeling, GNSS positioning, LEO-PNT navigation, and data

assimilation.



Martti Kirkko-Jaakkola received his M.Sc. and D.Sc. (Tech.) degrees from Tampere University of Technology, Finland, in 2008 and 2013, respectively. From 2013 to 2020, he worked full time at FGI as a Senior Research Scientist and Research Manager. As of 2021, his main occupation is with Nordic Inertial Oy, Akaa, Finland, with a secondary affiliation with the FGI. His research interests include low-cost GNSS and inertial navigation systems.



Oliver Horst received his M.Sc. degree in Information and Communication Technologies (ICT) with a major in autonomous systems and robotics from Aalto University, Espoo, Finland, in 2021. During the same year, at the Finnish Geospatial Institute in Helsinki, Finland, he focused on the development of HASlib. Since 2022, he has been working on industrial applications of indoor positioning systems at Kinexon, Munich, Germany.



Tuomo Malkamäki received his M.Sc. and Lic.Sc. (Tech.) degrees from Aalto University, Finland, in 2010 and 2013, respectively. Since 2015, he has worked as a Research Scientist and Research Group Manager at FGI. His research interests include navigation and positioning technology and sensors suitable for GNSS-degraded environments.



M. Zahidul H. Bhuiyan received a PhD degree in communications engineering from the Tampere University of Technology (TUT), Tampere, Finland, in 2007. He is a Research Professor at the Department of Navigation and Positioning, Finnish Geospatial Research Institute, Helsinki, Finland. He actively teaches GNSS-related courses with Finnish universities and other training schools. He has also worked as a Technical Expert with the EU Agency for the Space Program in H2020

project reviewing and proposal evaluation. His research interests include multi-GNSS receiver development, PNT robustness and resilience, and seamless positioning.



Sanna Kaasalainen received a PhD in astronomy from the University of Helsinki, Helsinki, Finland, in 2003. She is currently a Professor and Head of the Department of Navigation and Positioning, FGI of the National Land Survey, Helsinki, Finland. Her research interests include resilient positioning, navigation and timing, situational awareness, and optical sensors. She also has research experience in LIDAR remote sensing, sensor development, and astronomy.



Ignacio Fernández-Hernández received the Electronic Engineering degree in electronic engineering from ICAI, Madrid, Spain, in 2001, the M.B.A. degree in business administration from LBS, London, U.K., in 2011, and the PhD degree in electronic systems from Aalborg University, Aalborg, Denmark, in 2015, for his research on snapshot positioning and GNSS authentication. He works for the European Commission DG DEFIS, where he has led the definition of authentication and high-accuracy services for Galileo. In 2017, he was a Visiting Scholar with the GPS Laboratory, Department of Aeronautics and Astronautics, Stanford University, and is currently a Visiting Professor with KU Leuven, Leuven, Belgium.