Received: 15 September 2021 | Revised: 14 October 2022 | Accepted: 17 December 2022

DOI: 10.33012/navi.579

ORIGINAL ARTICLE

PPP/PPP-RTK Message Authentication

Ignacio Fernandez-Hernandez[*++] | **Rui Hirokawa**[+] | **Vincent Rijmen**[++**] | **Yusuke Aikawa**[+]

**European Commission, Belgium;*
[+]Mitsubishi Electric Corporation, Japan;
[++]KU Leuven, Belgium;
***University of Bergen, Norway*

Correspondence
Ignacio Fernandez-Hernandez
European Commission – DG DEFIS.C.2.
Av. d'Auderghem 45
1049 Brussels, Belgium
Email: ignacio.fernandez-hernandez@ec.europa.eu

Abstract

This paper analyzes candidate schemes for PPP/PPP-RTK (precise point positioning/real-time kinematic) data authentication. Asymmetric schemes are proposed based on existing standards and compatible with GNSS messages. Post-quantum cryptographic signatures are also reviewed and discussed. Two schemes are selected for analysis: digital signature (DS) based on ECDSA, and delayed disclosure (DD) based on a hybrid scheme using the TESLA protocol. Each of them is described in detail for both Galileo high-accuracy service and QZSS centimeter-level accuracy service. The performance of the schemes in terms of time to receive the corrections message and increase in the age of data (ΔAOD) is analyzed. The analysis is complemented by a review of the CPU consumption at receiver level.

Keywords

Galileo, message authentication, PPP, PPP-RTK, QZSS, spoofing

1 | INTRODUCTION

GNSS message and signal authentication has been discussed for almost two decades (Kerns et al., 2014; Scott, 2003; Wesson et al., 2012; Wullems et al., 2005). Nowadays, satellite navigation systems are gradually introducing these services, starting with Galileo open service navigation message authentication (OSNMA; Fernandez-Hernandez et al., 2016) already publicly broadcast with its signal in space. GPS will also introduce message and signal authentication soon in the experimental NTS-3 satellite (Anderson et al., 2017).

In parallel, Japan's Quasi-Zenith Satellite System (QZSS), Europe's Galileo, and China's BeiDou System (BDS) are providing and testing precise point positioning–real-time kinematic (PPP/PPP-RTK) corrections (Cabinet Office, 2021; Fernandez-Hernandez et al., 2022; Liu et al., 2020). However, in the current specifications, high-accuracy messages are not authenticated. Users may find themselves initially authenticating traditional navigation messages, but switching to high accuracy and using new unauthenticated data messages, remaining unprotected or with a false impression of security based on a partial protection.

This paper looks at the problem of PPP/PPP-RTK authentication, with a focus on PPP/PPP-RTK message authentication, i.e., ensuring that the origin of the data is the system and not another source (in addition to message integrity). The problem of high-accuracy data authentication is not trivial. Unlike standard broadcast

ephemeris, which can be constant for hours, high-accuracy corrections need to be fresh, no older than a few tens of seconds, and adding authentication latency may degrade performance.

On one hand, there is extensive literature on GNSS authentication, but on the other, very few references are available on PPP/PPP-RTK message authentication, to the knowledge of the authors. Hirokawa and Fujita (2019) analyzed the problem of authenticating PPP-RTK messages with a focus on Timed Efficient Stream Loss-Tolerant Authentication (TESLA; Perrig et al., 2002) for QZSS. Fernandez-Hernandez et al. (2015) further implemented TESLA for the first-ever signal-in-space test of high accuracy and authentication for Galileo, but without any optimization. Yet, highly secure, accurate, and autonomous location services remain an objective of the GNSS community.

The article first presents the rationale, design considerations, and performance indicators of PPP/PPP-RTK authentication. Then, we present and justify data authentication proposals for both Galileo High Accuracy Service (HAS) and QZSS Centimeter-Level Augmentation Service (CLAS). The proposals are then evaluated according to different reception conditions from open sky to hard urban. Finally, the paper finalizes with the conclusions of the analysis.

2 | WHY PPP/PPP-RTK AUTHENTICATION?

A typical satnav receiver needs to receive coherently forged signals from several satellites in order to compute a coherently false position. On the contrary, users of augmentation or correction messages, such as PPP/PPP-RTK or Satellite-Based Augmentation System (SBAS), are subject to a single point of failure, as a single message is providing corrections to many satellites. For this reason, other such systems are already developing data authentication solutions (Neish, 2020). In the case of PPP, one could argue that corrections are in the few-decimeter level. On occasion, however, corrections provide wider ranges to cope with higher errors. For example, QZSS CLAS orbit errors range in the message around ±26 m (Cabinet Office, 2021), and Galileo HAS orbit errors range around ±16 m (Fernandez-Hernandez et al., 2022). Such error magnitudes can cause a hazardous event for many applications. The same applies to clock and code bias corrections.

The importance of signal authentication becomes higher when the signal is used as a ranging source. GPS's Chimera (Anderson et al., 2017) can be used as a reference on this topic. PPP/PPP-RTK signals, such as those used with QZSS or HAS, can be used for both data provision and ranging purposes, but we focus here on their use as a data channel providing corrections applied to other GNSS measurements.

The GNSS broadcast data corrected by the PPP/PPP-RTK provider should be authenticated, too, for a full data-authenticated position. This is compatible with Galileo OSNMA and possibly GPS in the future. We also consider this feature relevant, but out of the main scope of this work.

3 | DESIGN CONSIDERATIONS

This section presents design considerations for the accommodation of an authentication scheme in PPP/PPP-RTK messages as well as cryptographic functions and parameters.

3.1 | Authentication Message Considerations

Message authentication adds bandwidth overhead, so we focus on proposals that can be accommodated in the current schemes—at least QZSS CLAS and Galileo HAS. We start from the premise that an ideal authentication scheme is one that provides the required security level and does not degrade user performance before adding authentication. For high-accuracy services, we focus on accuracy and availability, but in order to facilitate analysis, we use delta age of data (**Δ**AOD) and reception time, referred to here as time to retrieve data (TTRD).

The AOD measures the time elapsed between when the input for correction is available (e.g., the measurements to estimate the orbits or clocks) and when the orbit and clock corrections are applied in the receiver. The AOD depends on system latencies (measurement reception at the orbit and clock estimator, correction computation, uplink to satellites) that we assume are not affected by authentication, i.e., the system infrastructure has enough capabilities to almost-instantaneously perform any cryptographic operations. We, therefore, focus on increases in the AOD due to the transmission of the message from the satellite to the receiver and refer to it as ΔAOD. Notice that ΔAOD increases due to authentication, but ΔAOD does not *only* represent the increase due to authentication. The AOD-accuracy relationship is particularly important for clock corrections, as characterized in Hirokawa and Fernandez-Hernandez (2020) for GPS, Galileo, GLONASS, and BDS clocks, where it is shown that some satellites may have a clock error above a decimeter if the correction is one-minute old. We calculate ΔAOD on a per-message basis, i.e., if the receiver needs to receive more than one message (e.g., clocks and orbits separately) for a full position, ΔAOD is calculated for each of them:

$$\Delta AOD_j = t_{app,j} - t_{tx,j} \tag{1}$$

where $t_{app,j}$ is the time of the application of message j at the receiver, and $t_{tx,j}$ is the message j transmission start time by the satellite. We select application time and not reception time, as in the case of delayed disclosure, the correction may be received but not applied until it is authenticated. For digital signatures, we do not make this distinction, and consider the application time as the time when both the message and the signature are available.

TTRD measures the time it takes to receive all the corrections, from one to multiple messages, allowing the receiver to compute a corrected position:

$$TTRD = max(TTRD_j) \tag{2}$$

where $j = 1,...,N$, and N is the total number of messages to be received. When message authentication is added, TTRD is equivalent to the time to first authenticated fix (TTFAF; Caparra, et al., 2016) or time to first authentication (Kerns et al., 2014), with multiple messages possibly transmitted and received in parallel. For the case of HAS, two messages (one for orbits, mask, etc. and one for clocks) are received (N=2). This allows for the transmission of a short clock message that can be applied almost instantaneously, without waiting to receive other, more slowly varying information such as orbit corrections or satellite biases. For the case of CLAS, a message with the local atmospheric corrections is also included (N=3).

ΔAOD and TTRD are illustrated in Figure 1 for two HAS messages: *slow*, which uses 80% of the bandwidth, and *fast*, which uses 20% of the bandwidth. Two satellite streams are represented in two rows of pages, where each cell represents one

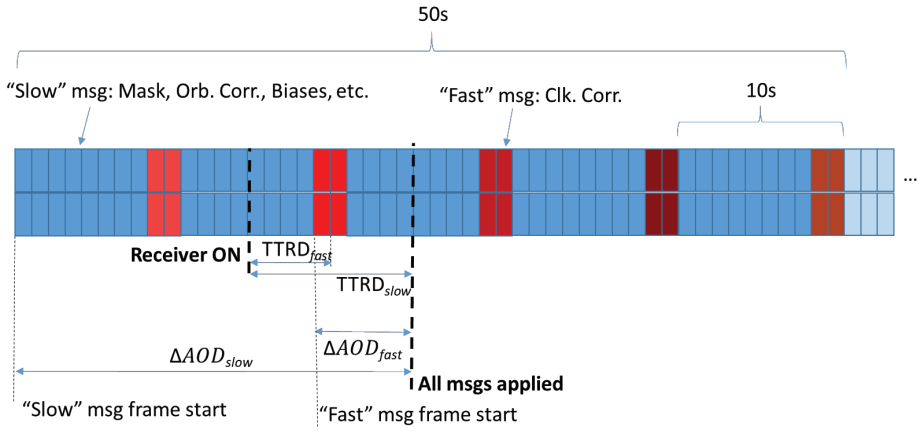

FIGURE 1 **Δ**AOD and TTRD for two messages (N=2): slow (blue) and fast (different shades of red/brown) on Galileo HAS message stream from two satellites.

page. A slow message is transmitted during 50 seconds, and then changes. The start of the fast and slow messages is represented by the thin dotted lines labeled *Fast/Slow message frame start*. The messages are 16 (slow) and two (fast) pages long—both have to be received—and each page counts for reception, thanks to the HAS High-Parity Vertical Reed Solomon (HPVRS) encoding scheme used (Fernandez-Hernandez et al., 2020a). The receiver starts up at a random point (*Receiver ON*), then receives eight slow message pages (four per satellite), then four fast message pages (two per satellite), and then it continues receiving the remaining eight slow message pages to complete the 16-page slow message and apply all corrections at the second bold dashed line (*All messages applied*). In this example, the time to retrieve the fast message was 5 seconds (TTRD_{fast} = 5 s), the time to retrieve the slow message was 10 seconds (TTRD_{slow} = 10s), and the ΔAOD for the slow and fast cases were 24 and 6 seconds, respectively, as shown in the figure. Note that, thanks to the HPVRS scheme, the receiver startup point (*Receiver ON*) may occur after the start of the slow message transmission, yet the message can be retrieved. This property also holds if the message includes authentication pages, as shown later.

ΔAOD and TTRD are correlated, but they are both analyzed for the sake of completeness. In addition, we look at the receiver computational complexity to verify authentication.

3.2 | Cryptographic Considerations

We propose some design considerations for the cryptographic protocol in this section:

1. **One-way asymmetry**: The authentication mechanism must be one-way and asymmetric, as opposed to symmetric authentication, based on a secret key in the possession of both the system and the receiver. This is the case of all other satnav civil authentication schemes. This asymmetry can be obtained by using digital signatures that can be verified using public information only, but not generated without the private key, as for elliptic curve cryptographic signatures like the Elliptic Curve Digital Signature Algorithm (ECDSA) or EC-Schnorr, or by time-delayed asymmetry, where a message authentication code (MAC) is transmitted and, after some delay, the key to generate it. The latter

is the case of the abovementioned TESLA protocol. TESLA is primarily based on symmetric cryptography for the generation of the authentication code but achieves asymmetric properties thanks to the delayed disclosure of the key.

2. **Long-term security, short-term validity**: The underlying cryptography must be considered secure for the long term, understanding *long term* to be 20–30 years. While updates are not discarded, satnav infrastructure has long lead times on the system side of several years. Also, some users may have restrictions to update their firmware or software with new cryptographic functions. For this reason, we focus a priori on 128-bit security levels (SL) for the cryptographic keys and, when a solution deviates from that, it is highlighted and justified. Note that, in contrast, some of the data authenticated is very short-lived: clock corrections degrade in a few tens of seconds. However, this should not be a driver for cryptographic parameters: First, the target of the authentication scheme is that no data is spoofed, no matter if it is short-lived or not. Second, the relevant attacks depend on the cryptoperiod of the secret key (i.e., the TESLA seed key or the ECDSA private key), which can last several years, rather than the validity of the MACs or the digital signatures. For TESLA, in addition to SL=128, we consider 40-bit MACs to be sufficient given that an attacker has a spoofing success probability of 2^{-40} per attempt, and the attack is detected when unsuccessful. The sensitivity of TESLA MACs and key lengths to data spoofing is analyzed in detail in Fernandez-Hernandez et al. (2021).
3. **Standards**: To the extent possible, we focus on message authentication solutions based on current standards to facilitate implementation in the receiver and widespread use. When this is not possible, we propose standards under development or proposals in the literature that, while not standardized, we consider plausible. Note that the TESLA protocol is standardized as part of the International Standards Organization (ISO) lightweight protocol standard (ISO, 2018a) and also proposed in several Internet Engineering Task Force (IETF) documents, such as Perrig et al. (2005) or Fries and Tschofenig (2006).
4. **Post-Quantum Cryptography**: One of the challenges of designing long-lifetime cryptosystems nowadays is the quantum threat. The National Institute of Standards and Technology (NIST) is developing post-quantum cryptography standards (NIST, 2020b). Three finalists were selected: Dilithium, Falcon, and Rainbow. Out of these, Rainbow seemed to be the best candidate for GNSS as it offers the shortest signatures, down to 528 bits. However, new attacks are being discovered (Beullens, 2020, 2022) so what is considered a secure signature size is not yet fixed. Table 1 compares the three NIST finalists, recent candidate Short Quaternion and Isogeny Signature (SQISign) showing promise, and with a comparatively short signature, a Great Multivariate Short Signature (GeMSS), which is not a NIST finalist but was preselected in previous rounds. It proposes the shortest signature of all (282 bits for SL=128) and has been proposed for GPS's NTS3 experimental satellite as its signature fits best with the GPS L1C message structure (Hinks et al., 2021).

The field of post-quantum cryptography evolves very quickly. New attacks are being discovered and old attacks are being improved by new insights on a monthly basis. These developments necessitate updates in the security parameters of the designs which, in turn, impact the key lengths and signature sizes. At the time of writing this paper, it seems premature to advocate for post-quantum algorithms in a GNSS specification. Also, a challenge of some schemes such as Rainbow and

TABLE 1
Three NIST Finalists for Postquantum Cryptography Digital Signatures: SQISign and GeMSS

SCHEME	REF.	SIGNATURE (bits)	PUBLIC KEY (kBytes)
Rainbow (parameters submitted to NIST, SL 112)	Ding and Schmidt (2005)	528	158
Rainbow (modified parameters, SL 128)	Beullens (2020)	568	203
Falcon (SL 128)	Fouque et al. (2020)	5328	0.88
Dilithium3	Bai et al. (2021)	26344	1.91
SQSISign	Feo et al. (2020)	1632	0.0625
GeMSS	Casanova et al. (2017)	282	444.69

GeMSS is the long public key. This may have a high impact in over-the-air rekeying (OTAR) if the new public keys are transmitted in the clear. This challenge can be overcome by having public keys already pre-stored in the receiver, encrypted with a short, symmetric key that is transmitted at the time of application (Caparra et al., 2017). In the sequel, we will focus the performance analysis on pre-quantum schemes, and assume that at least some post-quantum schemes will be close in performance to ECDSA. Further details about post-quantum schemes for GNSS can be found in Neish et al. (2019b).

4 | MESSAGE AUTHENTICATION PROPOSALS

We propose two main candidates: one based on delayed disclosure through TESLA and one based on ECDSA. For delayed disclosure, the protocol stores the data, D_i, and a MAC, $M_{i,tx}$, transmitted in parallel. Then, after some delay, its associated key, k_{i+1}, is disclosed and the receiver generates M_i and compares it with $M_{i,tx}$ as follows:

$$M_i = mac(k_{i+1}, D_i)$$
$$M_i == M_{i,tx} \rightarrow data\,authenticated \tag{3}$$

where $==$ is the *is equal to* conditional test operator. We assume a one-way function for the key generation and authentication as follows:

$$k_i = f(k_{i+1}, i, \alpha) = trunc(h(k_{i+1} \,|\, i \,|\, \alpha), l) \tag{4}$$

where k_i is the key generated from k_{i+1} through the one-way function, $f(.)$, and then disclosed in reverse order (0,..., i , $i+1$...); α is a salt to avoid precomputation attacks; $trunc(.,l)$ is the truncation function to l bits, where l is the key size; $h(.)$ is a hash function, which includes the disclosure time, $t_{GNSS,i}$, or an equivalent counter i; and | is the concatenation operator. Note that the TESLA protocol requires the receiver to be loosely synchronized with an external time source (Fernandez-Hernandez et al., 2020b).

Regarding the ECDSA, the digital signature can be modeled as follows:

$$B_i = v\left(S_{i,tx}, h\left(D_i(t_{GNSS,i})\right), K_{publ}\right)$$
$$B_i == true \rightarrow data\ authenticated \tag{5}$$

where $S_{i,tx}$ is the digital signature transmitted for D_i, i.e., the (r, s) pair for the ECDSA; K_{publ} is the public key; $v(.)$ is the signature verification function, and B_i is a Boolean variable with the result of the verification of signature i. In order to avoid data replay attacks, the data stream, D_i, needs to include the message time, $t_{GNSS,i}$, as otherwise previous data could be replayed.

As the main performance drivers rely on how the schemes are implemented in the message, the next two subsections propose concrete schemes for both Galileo HAS and QZSS CLAS.

4.1 | PPP Message Authentication Proposals: Galileo HAS

Galileo HAS defines flexible correction messages in terms of update rate and content. However, we focus on a basic standard configuration where there is a slow update rate message, including the satellite mask, orbits, and biases, updated every 50 s, and a fast update rate message, including the clocks, updated every 10 s, as shown in Figure 1. We will call the slow message with the satellite mask, orbit corrections, and biases *MT1-orb*, and the fast message with the satellite clocks *MT1-clk*. MT1-clk occupies two pages out of 10, and the rest is devoted to MT1-orb. Each HAS page is transmitted every second and has 448 bits—24 bits for the header and 424 bits for the body.

Table 2 presents the size of the MT1-orb and MT1-clk messages for a certain configuration, based on the Galileo HAS Interface Control Document (ICD; EU, 2022). The MT1-orb fits in 16 pages and the MT1-clk fits in two pages for corrections of Galileo and GPS, including four code and phase biases for Galileo and two code and phase biases for GPS.

We define two authentication proposals: one with digital signatures (DSs) based on current standards and one with a hybrid approach including a digital signature and TESLA delayed disclosure (DD). The proposals focus on the design choices that are more relevant for user performance. Aspects like key-authenticating-keys and OTAR are presented but not treated in detail, and will be looked at more closely in future work.

Digital Signature: This proposal is based on the ECDSA (NIST, 2013), although other signatures are possible, including Rainbow, as abovementioned, or EC-Schnorr (ISO, 2018b; Schnorr, 1989). We assume a 4xSL (security level) digital signature size. The message includes one digital signature for each MT1-orb and one for each MT1-clk. For the MT1-orb, one 424-bit page would be sufficient, as it accommodates 424 + 188 (spare) = 612 bits, enough for a 512-bit signature (SL=128 bits) of a P-256/SHA-256 ECDSA signature. However, this would only leave 100 bits every 50 s (i.e., 2 bps) for OTAR/key management (e.g., signed root key of the key chain). As a design choice, we propose two pages for a total of 1,036 bits, as shown in Figure 2. This allows a margin for OTAR and other fields.

For the MT1-clk authentication, having one extra page would give 424 + 80 (spare) = 504 bits. This is not sufficient for a 512-bit signature. However, given that

TABLE 2
Galileo HAS Message Size

GENERAL PARAMETERS		MESSAGE SIZE	MT1-ORB	MT1-CLK
Number of systems (GPS, Galileo)	2	Bits	6596	768
Number of satellites (Galileo/GPS)	24/32	Pages	16	2
Number of code and phase biases (Galileo/GPS)	4/2	Spare bits	188	80

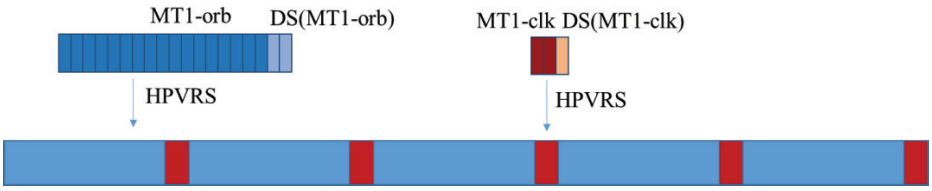

FIGURE 2 Galileo HAS message with digital signature (DS) authentication, including HPVRS encoding; OTAR is included in the two DS pages of the MT1-orb.

the MT1-clk is only two pages long, adding another two pages for authentication implies a 100% overhead. Then, a shorter signature of 448 bits from the ECDSA P-224/SHA-224 is chosen, providing SL=112, which might be sufficient. This requires two different public keys: one for the MT1-clk and one for the MT1-orb where, if needed, the MT1-clk 112-bit key can be updated more often. Currently, 112-bit security is the minimum acceptable security level value through 2030, per NIST (2020a), recommending 128 bits onward. If a minimum SL=128 has to be used, HAS flexibility allows for the provision of a clock subset message instead of a full clock set message, and updates the most stable clocks with the MT1-orb message. An optimized DS scheme could also just transmit one signature in the MT1-clk, signing both MT1-orb and MT1-clk, and leaving the MT1-orb unchanged. This approach is left for further work.

In summary, the HAS-DS approach adds two pages to the MT1-orb and one page to the MT1-clk, for a total of 18 and three pages, respectively, as shown in Figure 2.

Delayed Disclosure (Hybrid): This approach introduces a TESLA-like configuration. It is not obvious how to benefit from the advantages of TESLA for HAS, where most of the data is provided in a single message, transmitted by all satellites in view, for which a digital signature can be provided at a low overhead. The delayed disclosure is not ideal as, if the data is not used until authenticated, its **Δ**AOD increases. In order to overcome this problem, we apply an approach where, under certain constraints, some data can be used *before* it is authenticated. This approach is also under analysis for SBAS, where some messages need to be instantaneously applied (Neish et al., 2019a). For PPP, we can assume the following check is performed prior to using a clock correction:

$$\delta C_i^s - \delta C_{i-1}^s \leq K \cdot \sigma_{allan,s}(t_i - t_{i-1}) \tag{6}$$

where δC_i^s is the clock correction (a scalar) at time t_i for satellite s, K is a constant depending on the tolerable false alert probability (PFA), and $\sigma_{allan,s}$ is the Allan deviation for period $t_i - t_{i-1}$. For example, $K = 3.5$ would allow a PFA below 0.02%, assuming a normal clock error distribution. For a period of 10 seconds and $\sigma_{allan,s}(10s) = 1$ cm, if the delta correction were above 3.5 cm, it would be discarded and the satellite set to not monitored at the receiver. Note also that, as few-cm errors cannot lead to a successful spoofing attack for most applications, large K values could be used. Also, to simplify the process, a generic σ_{allan} for all satellites, including the worst case, could be used as well. In that case, one could assume a constant threshold for all corrections:

$$\delta C_i - \delta C_{i-1} \leq THR(PFA, \sigma_{allan}, \tau_{clk}) \tag{7}$$

where τ_{clk} is the clock update rate. In our case, and based on the use-then-authenticate approach, the delayed-disclosure protocol we propose is as follows: the MT1-orb is still signed with the DS. Then, the MT1-clk is authenticated via a TESLA chain,

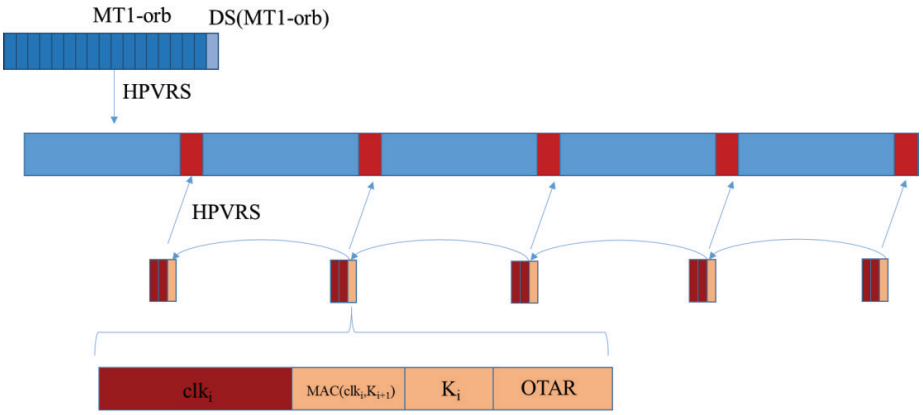

FIGURE 3 Galileo HAS message with delayed-disclosure (DD) authentication: MT1-clk is authenticated through TESLA and MT1-orb is authenticated through DS, which includes HPVRS encoding.

as shown in Figure 3. In our scenario, there are 504 bits available: 80 spare bits and 424 from a new page. They can be used for the MAC, the TESLA key, and key management data, as they are more than sufficient for MACs of up to 40 bits and keys up to 128 bits (for a total of 168 bits), which we consider to be secure for GNSS-TESLA. As OTAR mainly goes in the MT1-clk, we allow MT1-orb authentication in one page instead of two pages, as in the DS case. We assume the TESLA OTAR field includes the digital signature of the TESLA root key, and that all satellites use the same TESLA key chain.

While our configuration cannot accommodate the 168 TESLA authentication bits into the spare 80 bits, other configurations might reallocate the clock corrections (e.g., through clock subset corrections, as mentioned above) to make them fit in, and then OTAR data should be placed in the MT1-orb message. We did not pursue DD optimizations in our work because DS performance seems good enough, as is shown later.

4.2 | PPP/PPP-RTK Message Authentication Proposals: QZSS CLAS

QZSS CLAS also defines multiple correction messages named compact State Space Representation (SSR) based on Radio Technical Commission for Maritime services (RTCM) SSR (Cabinet Office, 2021). There are slow update rate messages including the satellite and signal mask, orbits, user range accuracy (URA), signal biases, and local atmospheric corrections (ionospheric and tropospheric delay) updated every 30 seconds, as well as fast update rate messages, including satellite clocks, that are updated every 5 seconds, as shown in Figure 4. For the sake of consistency with the previous section, we will call the slow message with the satellite mask, orbit corrections, and biases *MT-orb,* with an update rate of 30 seconds. The message with the local atmospheric corrections is named *MT-loc*, with an update rate of 30 seconds, and the message with the satellite clocks and an update rate of 5 seconds is *MT-clk*, per Figure 4. Each QZSS L6 message is transmitted every second and has 2,000 bits—49 bits for the header, 256 bits for Reed-Solomon error correction data, and 1,695 bits for the body (Cabinet Office, 2021). QZSS CLAS defines the subframe with an update rate of 5 seconds as having 8,475 bits, and includes multiple compact SSR messages. The major-frame with an update rate of 30 seconds includes six subframes.

TABLE 3
QZSS CLAS Message Size

GENERAL PARAMETERS		MESSAGE SIZE	MT-ORB	MT-CLK	MT-LOC
Number of systems (GPS, Galileo, QZSS)	3	Bits	3,202	337	5,575
Number of satellites (GPS/Galileo/QZSS)	9/8/3	Messages	2	1	4
Number of code and phase biases (GPS/Galileo/QZSS)	3/3/3	Spare bits	-	-	-

Table 3 presents the size of the MT-orb, MT-clk, and MT-loc messages according to the QZSS CLAS ICD (Cabinet Office, 2021). The parameters chosen for this example came from nationwide PPP-RTK services such as QZSS CLAS.

MT-orb requires 3,202 bits, which fits in the first subframe, and MT-clk requires 337 bits for each subframe, including corrections for 20 satellites (nine from GPS, eight from Galileo, and three from QZSS), and three code and phase biases for GPS, Galileo, and QZSS. The MT-loc atmospheric corrections of 5,575 bits include 12 areas with a total of 212 grid points.

As for Galileo HAS, we define two authentication schemes: digital signature (DS) and delayed disclosure (DD):

Digital Signature: This proposal is based on the ECDSA (NIST, 2013) as Galileo HAS. The change to the message consists in just adding a digital signature for each subframe including MT-orb, MT-clk, and MT-loc. In this paper, we assume that SL=128 is enough, and the 512-bit signature of a P-256/SHA-256 ECDSA is applied. For the data for OTAR/key management, we assume 804 bits, which is comparable to the Galileo HAS DS case. In QZSS CLAS, each 5-second subframe includes both slow corrections (such as orbit and/or atmospheric correction) and fast corrections (including clock), and we do not apply the shortened P-224 signature as for Galileo HAS. In summary, the CLAS-DS approach adds a DS for MT-orb, MT-clk, and MT-loc at the end of the first subframe, a DS for MT-clk and MT-loc at the end of the next four subframes, and a DS for MT-clk and MT-loc with OTAR and other fields at the end of the last subframe, respectively, as shown in Figure 5. The six digital signatures with a 512-bit length with OTAR and other fields require 3,876 bits for 30 seconds, or 129.2 bps. For the case of nationwide PPP-RTK, it consumes 63.3 bps to include the correction data for a satellite; the required data rate for authentication (129.2 bps) is approximately the same as the correction data for two satellites.

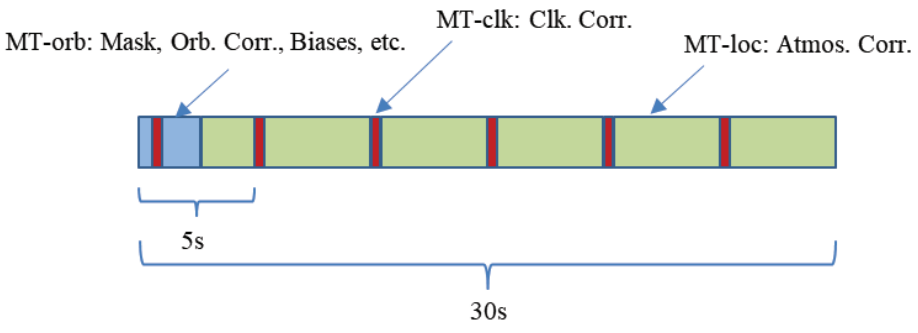

FIGURE 4 QZSS CLAS message configuration page stream from one satellite: The blue parts represent MT-orb, the green parts represent MT-loc, and red parts represent MT-clk.

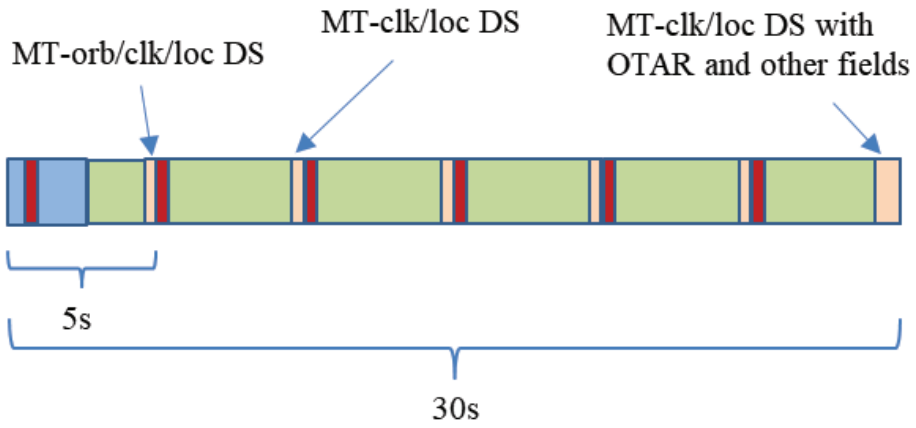

FIGURE 5 QZSS CLAS message with digital signature (DS) authentication: OTAR is included in the last DS with MT-clk and MT-loc.

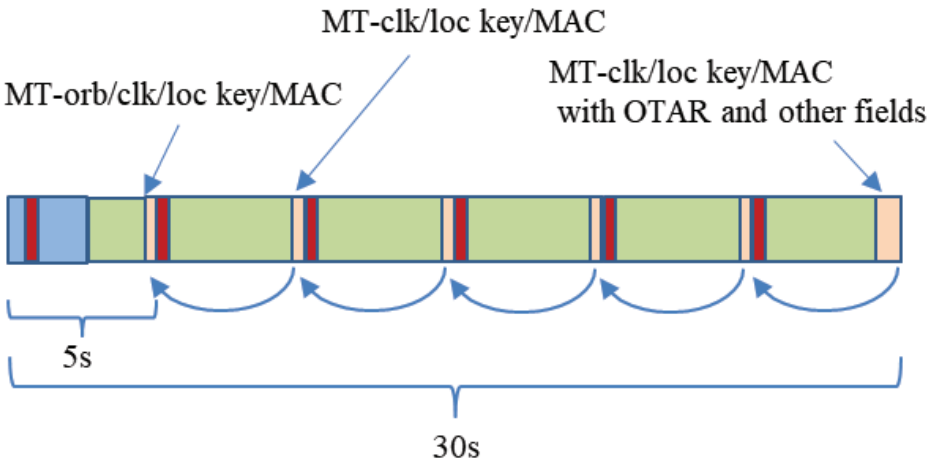

FIGURE 6 QZSS CLAS message with delayed disclosure (DD) authentication: Each subframe is authenticated through TESLA and OTAR is included in the last subframe.

Delayed Disclosure: This approach introduces a TESLA-like configuration based on the use-then-authenticate approach, such as that used for Galileo HAS. Each subframe is authenticated via a TESLA chain, as shown in Figure 6. MACs of up to 40 bits and keys of up to 128 bits are included for Galileo HAS, and 804 bits are assigned for OTAR with key management for DSs. The required data for authentication is 1,812 bits or 60.4 bps—approximately the same as the correction data for one satellite.

4.3 | Summary

A summary of the proposals is provided in Table 4, including the security bits, authentication bits, number of extra pages (HAS) or subframes (CLAS) per authentication, the authentication bit distribution in signatures, keys, MACs, and the rest (OTAR/spare), as well as the OTAR/spare bandwidth available.

TABLE 4
Authentication Proposals Summary.

	Security bits	Auth. bits	Pages /sub-frames data + auth.	Auth. bit distribution	OTAR/ spare BW	Comments
HAS-DS	128/112	Orb: 512 Clk: 448	Orb: 16 + 2 Clk : 2 + 1 [pages]	Orb: 512b DS + 524b OTAR/spare Clk: 448b DS + 56b OTAR/spare	16.1 bps	Both messages signed with DS. Clk DS reduced from 512 to 448 bits to fit in one page (+ spare).
HAS-DD	128	Orb: 512 Clk: 168	Orb: 16 + 1 Clk : 2 + 1 [pages]	Orb: 512b DS + 100b OTAR/spare Clk: 128b Key + 40b MAC + 336b OTAR/spare	35.6 bps	Hybrid scheme: MT1-orb signed with DS. MT1-clk based on 10-second-delay TESLA, use-then-authenticate. Assumption: 40-bit MACs, 128-bit keys. Loose time sync required in receiver. TESLA root key signature included in OTAR.
CLAS-DS	128	512	1 [subframe]	SF1-5: 512b DS SF6: 512b DS + 804b OTAR/spare	26.8 bps	Messages in subframe signed with DS.
CLAS-DD	128	168	1 [subframe]	SF1-5: 128b Key + 40b MAC, SF6: 128b Key + 40b MAC + 804b OTAR/spare	26.8 bps	Messages in sub-frame are authenticated with 5-second delay TESLA, use-then-authenticate. Assumption: 40-bit MACs, 128-bit keys. Loose time sync required in receiver. TESLA root key signature included in OTAR.

TABLE 5
Scenarios

	Open sky		Urban		Hard urban	
	Sats	PER	Sats	PER	Sats	PER
Galileo	4	0.005	4	0.01,0.05, 0.1,0.2	2	0.1, 0.2
QZSS	3	0.005	2	0.05, 0.1	2	0.1, 0.2

5 | TEST RESULTS

5.1 | Message Authentication Scenarios

We define three scenarios to test the impact of authentication in **Δ**AOD and TTRD. The scenarios are defined as a function of the number of transmitting satellites (of the PPP/PPP-RTK data) and page error rate (PER). The PERs are based on a land-mobile satellite model using experimental measurements described in Arndt et al. (2012), and then simplified for Galileo using conservative PER bounds as described in Fernandez-Hernandez et al. (2017). As the purpose of the analysis is to compare different schemes, we considered the model, even if simple, to be good enough for our purpose with one exception: The model did not take into account

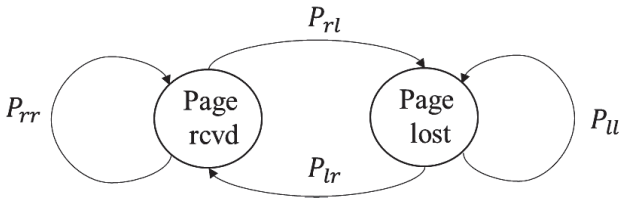

FIGURE 7 Markov chain burst model diagram

the previous state to determine the PER, which was just randomly allocated. In order to take into account error bursts, which occur often in environments such as those under study, a Markov chain for the PER determination should be added. We heuristically assumed a factor of three, i.e., after a message is lost, the PER would be augmented by three (e.g., from 20% to 60%), which is a way to model bursts for low PER values. The model in question is based on Fuller et al. (2000) and depicted in Figure 7.

The Markov chain probabilities are related to the PERs as follows:

$$\begin{aligned} &\text{Received page state} : P_{rl} = \text{PER} \;;\; P_{rr} = 1 - \text{PER} \\ &\text{Lost page state} \quad : P_{ll} = \text{PER} \cdot B \;;\; P_{lr} = 1 - \text{PER} \cdot B \end{aligned} \tag{8}$$

where PER takes the values of Table 5, and B is the Markov chain burst factor ($B = 3$). For each scenario (*open sky* [OS], *urban* [U], and *hard urban* [HU]), PER values were statically assigned per satellite according to Table 5 (e.g., every hard urban Monte-Carlo run would use two satellite-ground channels with PER values of 0.1 and 0.2). Notice that the PER value should be associated with certain signal power reception conditions and depends on the coding schemes implemented for each system: QZSS uses Reed-Solomon (RS) (Cabinet Office, 2021), and Galileo HAS uses HPVRS at the message level and a convolutional code at the page level. Also, the conditions presented are pessimistic, particularly the hard urban case for QZSS, given that at least one QZSS satellite is expected to be visible at the zenith in Japan. However, as our purpose is just to assess the relative impact of authentication under various conditions, we consider the model to be fit for its purpose—simple and broad enough.

The performance indicators TTRD and **Δ**AOD are calculated as follows: For each instance, a receiver in a given scenario (OS/U/HU) starts at a random point and collects messages until all corrections are received. The time elapsed is measured as TTRD. In addition, the **Δ**AOD values of both the clock (fast) and orbit/bias (slow) corrections (and local, for QZSS) are measured. A Monte-Carlo simulation with 10,000 instances is run for each scenario and authentication scheme combination in order to derive statistically meaningful results.

5.2 | Other Schemes for Comparison

In addition to the DS and DD schemes, the scenario models a scheme without authentication as a reference, as well as a scheme named *Full Delay Disclosure* (FDD). The purpose of the FDD scheme is to illustrate the effect of a standard TESLA implementation (i.e., without the use-then-authenticate approach). As the reader may anticipate, this increases the TTRD and **Δ**AOD due to the disclosure time, especially in the case of HAS, but we provide it as a reference in the test results. For HAS, FDD is based on one extra page for the orbit message (i.e., 17

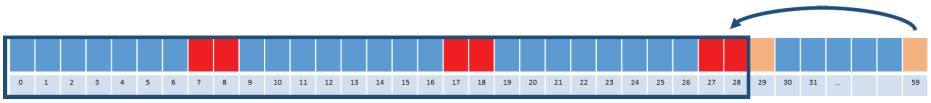

FIGURE 8 Full Delay Disclosure scheme for HAS based on a TESLA chain with a loose time sync requirement of 30 seconds, as well as MT1-Orb and MT1-Clk update rates of 30 s and 10 s, respectively

pages in total), with 30-s/10-s update rates for the orbit/clock messages, respectively, following a standard TESLA approach in which each message contains its MAC (not shown in the figure), and the next orbit message at the next 30-s interval contains the related key. The HAS scheme is shown in Figure 8. The CLAS case is similar, but with a DD of only 5 seconds.

5.3 | TTRD and ΔAOD Results

The TTRD and **Δ**AOD results for Galileo HAS are shown in Table 6 for the average and 95th percentile. The results show that, on average, the DS and DD schemes performed very closely to one another, and with only a small degradation with respect to the reference case, of about 1–2 seconds in open sky and urban environments, and up to 5 seconds in hard urban environments for the TTRD. The **Δ**AOD was degraded by approximately 2 seconds. This degradation seems affordable, especially for decimeter-level applications. The full probability cumulative distribution functions (CDFs) for Galileo HAS can be seen on the right side of Figure 9, while the left side shows the TTRD for all instances of all scenarios and schemes from which the statistics were derived. One can appreciate a slight degradation of DS with respect to DD, particularly for the hard urban case.

Taking into account the similar performance of DS and DD, and that DD requires some assumptions on the data processing (use-then-authenticate) and loose time synchronization, DS seems to be a better choice. The main factor that can lean the design closer toward DD is computational cost, which is generally lower for

TABLE 6
Average and 95th Percentile TTRD and ΔAOD [s] for Galileo HAS

	Average								
	OPEN SKY			**URBAN**			**HARD URBAN**		
	TTRD	**Δ**AOD-clk	**Δ**AOD-orb	TTRD	**Δ**AOD-clk	**Δ**AOD-orb	TTRD	**Δ**AOD-clk	**Δ**AOD-orb
Ref (no auth)	5.61	2.07	22.31	6.32	2.79	22.68	14.07	4.78	26.41
GAL-DS	6.62	4.10	24.30	7.22	4.61	24.35	18.40	6.09	28.38
GAL-DD	6.64	4.16	24.21	6.96	4.37	24.68	17.48	5.97	28.16
GAL-FDD	50.50	32.00	59.00	51.33	32.00	59.00	58.92	32.49	59.00
	95th percentile								
	OPEN SKY			**URBAN**			**HARD URBAN**		
	TTRD	**Δ**AOD-clk	**Δ**AOD-orb	TTRD	**Δ**AOD-clk	**Δ**AOD–orb	TTRD	**Δ**AOD-clk	**Δ**AOD-orb
Ref (no auth)	8.00	6.00	44.00	8.00	7.00	45.00	24.00	9.00	47.00
GAL-DS	9.00	8.00	47.00	10.00	9.00	47.00	30.00	9.00	48.00
GAL-DD	9.00	8.00	47.00	10.00	8.00	47.00	29.00	9.00	47.00
GAL-FDD	64.00	32.00	59.00	65.00	32.00	59.00	73.00	32.00	59.00

symmetric operations, like MAC computations, than for asymmetric ones, like the ECDSA. In the following subsection, we analyze the differences in computational cost based on existing recent literature. The figure also shows a staircase shape in the FDD's CDF for HAS. The reason for this is that the TTRD ends when the TESLA key is received (usually in 60 s), leading to a uniform TTRD probability distribution for each random start time and a staircase shape in the CDF. The FDD approach also causes a delay of at least 30 seconds for HAS with respect to the DS and DD cases.

The TTRD and **Δ**AOD for QZSS CLAS results are also shown in Table 7 for the average and 95th percentile. For QZSS CLAS, the results show that, on average, the DS and DD schemes performed very closely to one another in open sky and urban

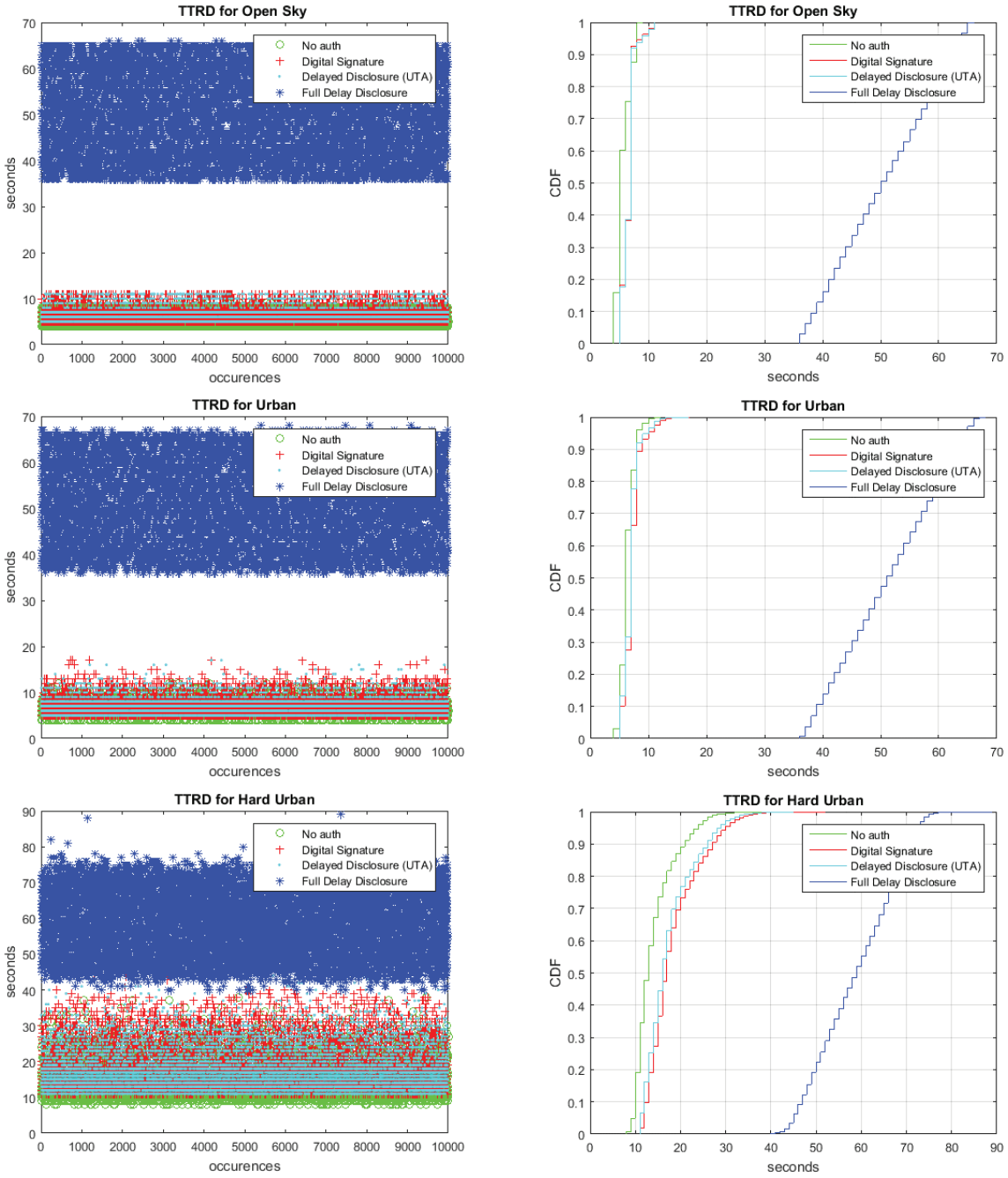

FIGURE 9 Monte-Carlo simulations (left) and CDFs (right) for the TTRD including conditions of no authentication, DS, DD, and FDD, for the open sky, urban, and hard urban cases for Galileo HAS

TABLE 7
Average and 95th Percentile TTRD and ΔAOD [s] for QZSS CLAS.

	Average											
	OPEN SKY				**URBAN**				**HARD URBAN**			
	TTRD	ΔAOD -clk	ΔAOD -orb	ΔAOD -loc	TTRD	ΔAOD -clk	ΔAOD -orb	ΔAOD -loc	TTRD	ΔAOD -clk	ΔAOD -orb	ΔAOD -loc
Ref (no auth)	21.54	2.39	6.94	8.07	21.63	2.39	6.95	8.08	26.71	2.41	7.37	8.72
QZS-DS	23.13	3.98	8.53	9.20	23.22	3.98	8.54	9.21	28.51	4.04	9.16	9.67
QZS-DD	23.13	3.98	8.53	9.20	23.22	3.98	8.54	9.21	28.51	4.04	9.16	9.67
QZS-FDD	28.87	3.90	14.26	9.31	29.04	3.90	14.28	9.31	40.39	3.94	15.47	9.97
	95th percentile											
	OPEN SKY				**URBAN**				**HARD URBAN**			
	TTRD	ΔAOD -clk	ΔAOD -orb	ΔAOD -loc	TTRD	ΔAOD -clk	ΔAOD -orb	ΔAOD -loc	TTRD	ΔAOD -clk	ΔAOD -orb	ΔAOD -loc
Ref (no auth)	30.00	4.00	21.00	27.00	31.00	4.00	21.00	27.00	56.00	4.00	23.00	27.00
QZS-DS	32.00	4.00	24.00	24.00	33.00	4.00	24.00	24.00	56.00	4.00	24.00	24.00
QZS-DD	32.00	4.00	24.00	24.00	33.00	4.00	24.00	24.00	58.00	4.00	24.00	24.00
QZS-FDD	38.00	4.00	28.00	24.00	38.00	4.00	58.00	24.00	82.00	4.00	34.00	29.00

environments as they did for Galileo HAS, where the degradation with respect to the reference case was about 1–2 seconds. The performance was degraded for the hard urban case for the TTRD, where the degradation was up to 11 seconds. The ΔAOD was degraded by approximately 2 seconds, as it had for Galileo HAS. Because of the changes in the message framing, the QZSS CLAS case shows a slight improvement in ΔAOD-loc at 95% due to authentication, but a slight degradation in most other parameters and cases. The full probability CDFs can be seen on the right side of Figure 10, while the left side shows the TTRD of all instances for all scenarios and schemes from which the statistics were derived. The FDD approach also caused a delay, with respect to the DS and DD cases, of at least 5 seconds which is the time it takes to receive the TESLA key in the next subframe.

5.4 | Computational Cost

The computational cost of ECDSA and TESLA authentication has been analyzed in Cancela et al. (2019). Table 8 shows the averaged CPU time for several devices, including portable ones, for the ECDSA and Hash-based Message Authentication Code (HMAC)-SHA256, which is used for OSNMA MACs and is representative of the delayed disclosure (DD) schemes here. The table shows that, with the selected processors, both operations are affordable, although as expected, the hash-based message authentication code (HMAC) operation was significantly lighter. GNSS receiver processors may have a lower clock rate, though, on the order of 100–200 MHz. Curran and Hanley (2019) show an average processing time of 49–26 ms for ECDSA-P224 and 84–180 MHz for Advanced RISC (reduced instruction set computer) Machine (ARM) processors, which could be representative of GNSS receivers. Also, Troglia et al. (2021) presents a detailed profiling of OSNMA crypto functions in ARM-based platforms, showing results in the same range.

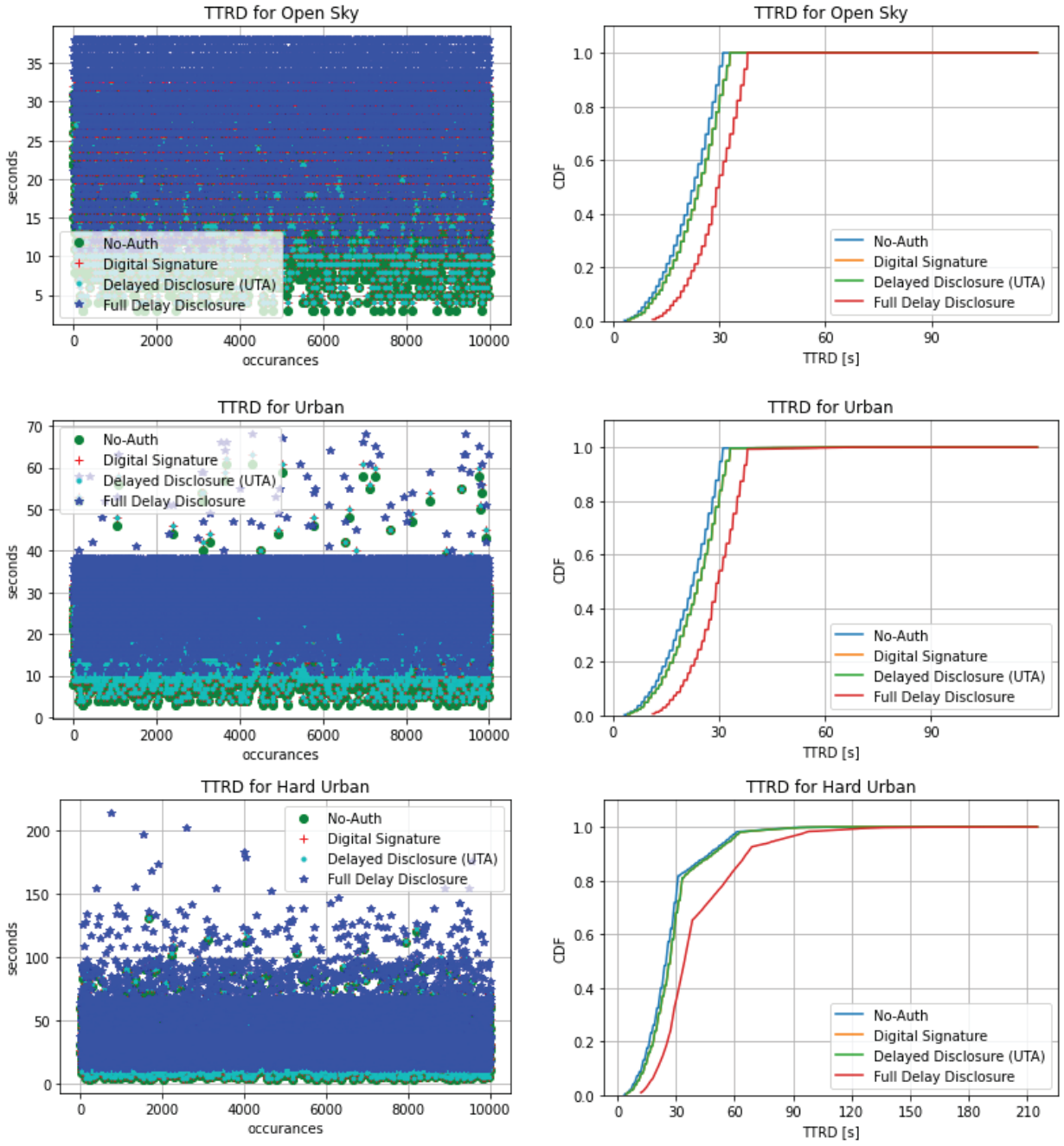

FIGURE 10 Monte-Carlo simulations (left) and CDFs (right) for the TTRD including conditions of no authentication, DS, DD, and FDD, for the open sky, urban, and hard urban cases for QZSS CLAS

Based on these figures, we assume that a high accuracy receiver can afford to perform ECDSA digital signature verifications at least once every 5–10 seconds without a significant increase in CPU load. In fact, Troglia et al. (2021) shows a computational cost of a standard per-second position, velocity, and time (PVT) computation, excluding carrier-phase processing and ambiguity estimation, for a 1.5-GHz ARM processor on a Raspberry device, of approximately 17 ms, while the ECDSA-P256 verification cost is below 1 ms. Note also that PPP/PPP-RTK receivers need to perform multi-frequency, multi-GNSS tracking including carrier-phase processing and possibly integer ambiguity resolution.

TABLE 8
Computational Cost for ECDSA and HMAC (TESLA) Cryptographic Operations (Cancela et al., 2019)

	CPU	ECDSA P256 [ms]	HMAC-SHA256 [ms]
Computer	i5-2400 CPU @ 3.10GHz	2.526	0.278
Samsung Galaxy S6	Octa-core (4x2.1 GHz Cortex-A57 & 4x1.5 GHz Cortex-A53)	8.745	0.693
Xiaomi MI 5S	Quad-core (2x2.15 GHz Kryo & 2x1.6 GHz Kryo)	9.235	0.656
Samsung Galaxy Tab A	Quad-core 1.2 GHz	27.868	1.881
LG G4	Hexa-core (4x1.4 GHz Cortex-A53 & 2x1.8 GHz Cortex-A57)	12.143	1.454

To complement the analysis of pre-quantum schemes, and leaving aside the recent attacks mentioned and their potential impact, Rainbow performs well compared to other postquantum schemes. According to the Rainbow submission documents, we have the following timings on an ARM Cortex-M4 processor running at 16 MHz: signature generation of 47 ms and signature verification of 15 ms. Therefore, the Rainbow signature verification cost also seems affordable for PPP/PPP-RTK receivers.

6 | CONCLUSION

PPP/PPP-RTK data authentication can avoid coherent position spoofing by modifying the PPP/PPP-RTK correction message, which may constitute a single point of failure for high accuracy receivers. The data authentication asymmetric schemes proposed must be long-term cryptographically secure, based on current standards if possible, and lightweight enough to be accommodated for in GNSS messages. Two schemes were selected for analysis: DS, based on ECDSA-P256 for 512-bit signatures (P-224 for 448-bit signatures in one case), and DD, based on a hybrid delayed-disclosure protocol using TESLA with the same key chain from all satellites. DS and DD were implemented for both Galileo HAS, transmitting PPP corrections for GPS and Galileo, and QZSS CLAS, transmitting PPP-RTK corrections from QZSS inclined geo-synchronous orbit (IGSO) satellites, which include local ionospheric and tropospheric corrections.

The performance of the schemes in terms of absolute time to receive the corrections message (TTRD) and increase in the age of data (ΔAOD) was analyzed, including urban scenarios leading to frequent reception errors. The results show that DS degradation is small, below 2 seconds in most conditions. DD degradation is also small, but under the condition that the receiver can use the clock corrections before they are authenticated. With this purpose, a test based on a-priori Allan variances and the clock update period is proposed.

Postquantum signatures were also discussed, in particular, those preselected by the NIST. Out of those, Rainbow signatures can yield a performance close to the ECDSA. However, it is premature to suggest a postquantum signature at this stage, due to the new attacks regularly discovered and variations in their security parameters. GNSS PPP/PPP-RTK providers might propose mixed schemes with both prequantum (ECDSA) and postquantum (e.g., Rainbow) implementations as a risk-mitigation measure.

The analysis Is complemented by a"revi'w of the receiver CPU consumption for the cryptographic operations. DD operations are less CPU-intensive but, in exchange, they require loose time synchronization and use-then-authenticate logic. On the contrary, asymmetric schemes are more CPU intensive but they seem affordable for PPP/PPP-RTK receivers.

DISCLAIMER

The content of this article does not necessarily reflect the official position the authors' organizations. Responsibility for the information and views set out in this article lies entirely with the authors.

REFERENCES

Anderson, J. M., Carroll, K. L., DeVilbiss, N. P., Gillis, J. T., Hinks, J. C., O'Hanlon, B. W., Rushanan, J.J., Scott, L., & Yazdi, R. A. (2017). Chips-message robust authentication (Chimera) for GPS civilian signals. *Proc. of the 30th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2017),* Portland, OR. https://doi.org/10.33012/2017.15206

Arndt, D., Heyn, T., Konig, J., Ihlow, A., Heuberger, A., Prieto-Cerdeira, R., & Eberlein, E. (2012). Extended two-state narrowband LMS propagation model for S-Band. *Proc. of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB),* Seoul, Korea. https://doi.org/10.1109/BMSB.2012.6264301

Bai, S., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., & Stehlé, D. (2021). *CRYSTALS-Dilithium: algorithm specifications and supporting documents* (Version 3.1). https://pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf

Beullens, W. (2020). *Improved cryptanalysis of UOV and Rainbow* [Report 2020/1343]. Cryptology ePrint Archive. https://eprint.iacr.org/2020/1343

Beullens, W. (2022). *Breaking Rainbow takes a weekend on a laptop* [Report 2022/214]. Cryptology ePrint Archive. https://eprint.iacr.org/2022/214.pdf

Cabinet Office (2021*). Quasi-Zenith Satellite System interface specification Centimeter Level Augmentation Service* [Report IS-QZSS-L6-004]. https://qzss.go.jp/en/technical/download/pdf/ps-is-qzss/is-qzss-l6-004.pdf

Cancela, S., Calle, J. D., & Fernandez-Hernandez, I. (2019). CPU consumption analysis of TESLA-based navigation message authentication. *2019 European Navigation Conference (ENC)*, Warsaw, Poland. https://doi.org/10.1109/EURONAV.2019.8714171

Caparra, G., Ceccato, S., Sturaro, S., & Laurenti, N. (2017). A key management architecture for GNSS open service navigation message authentication. *2017 European Navigation Conference (ENC)*, Lausanne, Switzerland. https://doi.org/10.1109/EURONAV.2017.7954220

Caparra, G., Wullems, C., Ceccato, S., Sturaro, S., Laurenti, N., Pozzobon, O., Ioannides, R.T., Crisci, M. (2016). Design drivers for navigation message authentication schemes for GNSS systems. *InsideGNSS*, 64-73.

Casanova, A., Faugere, J. -C., Macario-Rat, G., Patarin, J., Perret, L., & Ryckeghem, J. (2017). *GeMSS: a great multivariate short signature*. https://www-polsys.lip6.fr/Links/NIST/GeMSS_specification.pdf

Curran, J., & Hanley, N. (2019). On the energy and computational cost of message authentication schemes for GNSS. *IEEE Aerospace and Electronic Systems Magazine*, *34*(1), 40–53. https://doi.org/10.1109/MAES.2019.180078

Ding, J., & Schmidt, D. (2005). Rainbow, a new multivariable polynomial signature scheme. In J. Ioannidis, A. Keromytis, & M. Yung (Eds.), *Applied cryptography and network security* (pp. 164–175). Springer-Verlag. https://doi.org/10.1007/11496137_12

European Union (EU). (2022). Galileo High Accuracy Service signal-in-space Interface Control Document [Report HAS SIS ICD, Issue 1.0]. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_HAS_SIS_ICD_v1.0.pdf

Feo, L. D., Kohel, D., Leroux, A., Petit, C., & Wesolowski, B. (2020). *SQISign: compact post-quantum signatures from quaternions and isogenies.* Cryptology ePrint Archive. https://eprint.iacr.org/2020/1240.pdf

Fernandez-Hernandez, I., Ashur, T., & Rijmen, V. (2021). Analysis and recommendations for MAC and Key lengths in delayed disclosure GNSS authentication protocols. *IEEE Transactions on Aerospace and Electronic Systems*, *57*(3), 1827–1839. https://doi.org/10.1109/TAES.2021.3053129

Fernandez-Hernandez, I., Calle, J. D., Cancela, S., Pozzobon, O., Sarto, C., & Simón, J. (2017). Packet transmission through navigation satellites: a preliminary analysis using Monte Carlo simulations. *2017 European Navigation Conference (ENC)*, Lausanne, Switzerland. https://doi.org/10.1109/EURONAV.2017.7954221

Fernandez-Hernandez, I., Chamorro-Moreno, A., Cancela-Diaz, S., Calle-Calle, J. D., Zoccarato, P., Blonski, D., Senni, T., de Blas, F. J., Hernandez, C., Simon, J., & Mozo, A. (2022). Galileo high accuracy service: initial definition and performance. *GPS Solutions, 26*(65). https://doi.org/10.1007/s10291-022-01247-x

Fernandez-Hernandez, I., Rijmen, V., Seco-Granados, G., Simon, J., Rodriguez, I., & Calle, J. D. (2016). A navigation message authentication proposal for the Galileo open service. *NAVIGATION*, *63*(1), 85–102. https://doi.org/10.1002/navi.125

Fernandez-Hernandez, I., Rodríguez, I., Tobías, G., Calle, J. D., Carbonell, E., Seco-Granados, G., Simon, J., & Blasi, R. (2015). Testing GNSS high accuracy and authentication–Galileo's commercial service. *Inside GNSS*, 37–48.

Fernandez-Hernandez, I., Senni, T., Borio, D., & Vecchione, G. (2020a). High-Parity vertical Reed Solomon codes for long GNSS high accuracy messages. *NAVIGATION,* 67(2), 365–378.. https://doi.org/10.1002/navi.357

Fernandez-Hernandez, I., Walter, T., Neish, A., & O'Driscoll, C. (2020b). Independent time synchronization for resilient GNSS receivers. *Proc. of the 2020 International Technical Meeting of the Institute of Navigation*, San Diego, CA, 964–978. https://doi.org/10.33012/2020.17190

Fouque, P. -A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., & Zhang, Z. (2020). *Falcon: fast-Fourier lattice-based compact signatures over NTRU* (Specification v1.2). https://falcon-sign.info/falcon.pdf

Fries, S., & Tschofenig, H. (2006). *RFC-4442: Bootstrapping Timed Efficient Stream Loss-Tolerant Authentication (TESLA).* IETF. https://doi.org/10.17487/RFC4442

Fuller, R., Walter, T., & Enge, P. (2000). Burst mode message loss effects on WAAS availability. *Proc. of the 13th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS 2000)*, Salt Lake City, UT, 230–241. https://www.ion.org/publications/abstract.cfm?articleID=1408

Hinks, J., Gillis, J. T., Loveridge, P., Myer, G., Rushanan, J. J., & Stoyanov, S. (2021). Signal and data authentication experiments on NTS-3. *Proc. of the 34th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2021),* St. Louis, MO, 3621–3641. https://doi.org/10.33012/2021.17964

Hirokawa, R., & Fernandez-Hernandez, I. (2020). Open format specifications for PPP/PPP-RTK services: overview and interoperability assessment. *Proc. of the 33rd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2020)*, 1268–1290. https://doi.org/10.33012/2020.17620

Hirokawa, R., & Fujita, S. (2019). A message authentication proposal for satellite-based nationwide PPP-RTK correction service. *Proc. of 32th International Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2019)*, Miami, FL, 1798–1811. https://doi.org/10.33012/2019.17085

International Standards Organization (ISO). (2018a). *Information technology–security techniques–lightweight cryptography–Part 7: Broadcast authentication protocols* [Report ISO/IEC 29192-7]. https://www.iso.org/obp/ui#iso:std:iso-iec:29192:-7:dis:ed-1:v1:en

International Standards Organization (ISO). (2018b). *IT Security techniques–digital signatures with appendix–Part 3: discrete logarithm based mechanisms* [Report ISO/IEC 14888-3:2018]. https://www.iso.org/obp/ui/#iso:std:iso-iec:14888:-3:ed-4:v1:en

Kerns, A. J., Wesson, K. D., & Humphreys, T. E. (2014). A blueprint for civil GPS navigation message authentication. *2014 IEEE/ION Position, Location and Navigation Symposium*, Monterey, CA. https://doi.org/10.1109/PLANS.2014.6851385

Liu, C., Gao, W., Liu, T., Wang, D., Yao, Z., Gao, Y., Nie, X., Wang, W., Li, D., Zhang, W., Wang, D., & Rao, Y. (2020). Design and implementation of a BDS precise point positioning service. *NAVIGATION*, *67*(4), 875–891. https://doi.org/10.1002/navi.392

Neish, A. M. (2020). *Establishing trust through authentication in satellite based augmentation systems* [Doctoral dissertation, Stanford University]. https://web.stanford.edu/group/scpnt/gpslab/pubs/theses/Neish-Thesis-Final.pdf

Neish, A., Walter, T., & Enge, P. (2019a). Quantum-resistant authentication algorithms for satellite-based augmentation systems. *NAVIGATION*, *66*(1), 199–209. https://doi.org/10.1002/navi.287

Neish, A., Walter, T., & Powell, J. D. (2019b). SBAS data authentication: a concept of operations. *Proc. of the 32nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2019)*, Miami, FL, 1812–1823. https://doi.org/10.33012/2019.17086

National Institute of Standards and Technology (NIST). (2013). *Digital signature standard* (DSS) [Report FIPS PUB 186-4]. https://csrc.nist.gov/publications/detail/fips/186/4/final

National Institute of Standards and Technology (NIST). (2020a). *Recommendation for key management: part 1 – general* [Report 800-57 Part 1 Rev. 5]. https://csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-5/final

National Institute of Standards and Technology (NIST). (2020b). *Post-quantum cryptography (PQC)*. NIST. https://csrc.nist.gov/projects/post-quantum-cryptography

Perrig, A., Canetti, R., Tygar, J. D., & Song, D. (2002). The TESLA broadcast authentication protocol. *CryptoBytes, 5*(2), 2–13. https://www.readkong.com/page/the-tesla-broadcast-authentication-protocol-5530155

Perrig, A., Song, D., Canetti, R., Tygar, J. D., & Briscoe, B. (2005). *Timed Efficient Stream Loss-Tolerant Authentication (TESLA): multicast source authentication transform introduction* [Report RFC 4082]. Datatracker. https://datatracker.ietf.org/doc/rfc4082/

Schnorr, C. P. (1989). Efficient identification and signatures for smart cards. In G. Brassard (Ed.), *Advances in cryptology–CRYPTO '89 Proceedings* (Vol. 435, pp. 239–252). Springer. https://doi.org/10.1007/0-387-34805-0_22

Scott, L. (2003). Anti-spoofing & authenticated signal architectures for civil navigation systems. *Proc. of the 16th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS/GNSS 2003),* Portland, OR, 1543–1552. https://www.ion.org/publications/abstract.cfm?articleID=5339

Troglia Gamba, M., Nicola, M., & Motella, B. (2021). Computational load analysis of a Galileo OSNMA-ready receiver for ARM-based embedded platforms. *Sensors, 21*(2), 467. https://doi.org/10.3390/s21020467

Wesson, K., Rothlisberger, M., & Humphreys, T. (2012). Practical cryptographic civil GPS signal authentication. *NAVIGATION, 59*(3), 177–193. https://doi.org/10.1002/navi.14

Wullems, C., Pozzobon, O., & Kubik, K. (2005). Signal authentication and integrity schemes for next generation global navigation satellite systems. *2005 European Navigation Conference (ENC-GNSS)*, Munich, Germany. https://www.researchgate.net/profile/Kurt-Kubik/publication/265821209_Signal_Authentication_and_Integrity_Schemes_for_Next_Generation_Global_Navigation_Satellite_Systems/links/54b61e6b0cf28ebe92e7a784/Signal-Authentication-and-Integrity-Schemes-for-Next-Generation-Global-Navigation-Satellite-Systems.pdf

How to cite this article: Fernandez-Hernandez, I., Hirokawa, R., Rijmen, V., & Aikawa, Y. (2023). PPP/PPP-RTK message authentication. *NAVIGATION, 70*(2). https://doi.org/10.33012/navi.579