Improving Galileo OSNMA Time to First Authenticated Fix

ALEIX GALAN-FIGUERAS , Graduate Student Member, IEEE
Katholieke Universiteit Leuven, Leuven, Belgium

IGNACIO FERNANDEZ-HERNANDEZ
Katholieke Universiteit Leuven, Leuven, Belgium
European Commission, Brussels, Belgium

WIM DE WILDE
Septentrio NV, Leuven, Belgium

SOFIE POLLIN , Senior Member, IEEE
Katholieke Universiteit Leuven, Leuven, Belgium

GONZALO SECO-GRANADOS , Fellow, IEEE
Universitat Autònoma de Barcelona, Barcelona, Spain
Institute of Space Studies of Catalonia (IEEC), Castelldefels, Spain

Galileo is the first global navigation satellite system to authenticate its civilian signals through the open service navigation message authentication (OSNMA) protocol. However, OSNMA adds a delay in the time to obtain a first position and time fix, the so-called time to first authentication fix (TTFAF). Reducing the TTFAF as much as possible is crucial to integrate the technology seamlessly into existing products. In cases where the receiver already has cryptographic data available, the so-called *hot start* mode and focus of this article, currently available implementations achieve an average TTFAF of around 100 s in ideal environments. In this work, we explore TTFAF optimizations available to general OSNMA-capable receivers and to receivers with tighter time synchronization than that required by the OSNMA receiver guidelines. We dissect the TTFAF process, describe optimizations, and benchmark them in three distinct scenarios (open-sky, soft urban, and hard urban) using recorded real data. Moreover, we also evaluate these optimizations using a synthetic scenario from the official OSNMA test vectors. The first block of optimizations centers on extracting as much information as possible from broken subframes by processing them at the page level and combining redundant data from multiple satellites. The second block of optimizations aims to reconstruct missing navigation data through the intelligent use of fields in authentication tags that belong to the same subframe as the authentication key. Combining both optimization ideas improves the TTFAF substantially for all considered scenarios. We obtain an average TTFAF of 60.9 s for the test vectors and 68.8 s for the open-sky scenario, with a lowest TTFAF of 44.0 s in both cases. Similarly, the urban scenarios show a drastic reduction in the average TTFAF between the nonoptimized and optimized cases. These optimizations have been made available as part of the open-source OSNMAlib library on GitHub.

Received 18 March 2024; revised 27 September 2024 and 4 April 2025; accepted 6 May 2025. Date of publication 14 May 2025; date of current version 13 October 2025.

DOI. No. 10.1109/TAES.2025.3570273

Refereeing of this contribution was handled by J. Blanch.

The work of Aleix Galan-Figueras was supported in part by the Research Foundation Flanders Frank de Winne PhD Fellowship under Grant 1SH9424N.

Authors' addresses: Aleix Galan-Figueras and Sofie Pollin are with the Department of Electrical Engineering, Katholieke Universiteit Leuven, 3001 Leuven, Belgium, E-mail: (aleix.galan@kuleuven.be; sofie.pollin@kuleuven.be); Ignacio Fernandez-Hernandez is with the Department of Electrical Engineering, Katholieke Universiteit Leuven, 3001 Leuven, Belgium, and also with European Commission, 1049 Brussels, Belgium, E-mail: (ignacio.fernandez-hernandez@kuleuven.be); Wim De Wilde is with Septentrio NV, 3001 Leuven, Belgium, E-mail: (wim.dewilde@septentrio.com); Gonzalo Seco-Granados is with the Department of Telecommunication Engineering, Universitat Autònoma de Barcelona, 08193 Barcelona, Spain, and also with the Institute of Space Studies of Catalonia (IEEC), 08860 Castelldefels, Spain, E-mail: (gonzalo.seco@uab.cat), *(Corresponding author: Aleix Galan-Figueras.)*

I. INTRODUCTION

Global navigation satellite system (GNSS) signals are vulnerable to interference, including the transmission of false GNSS-like signals, or spoofing. The addition of cryptographic information to civil GNSS signals was proposed decades ago as a way to detect spoofing [1], but its implementation has taken time. Meanwhile, several receiver-based antispoofing methods, such as signal power monitoring [2] or inertial systems [3], have been proposed. Finally, GNSS signals are gradually starting to provide cryptographic information.

Cryptographic techniques exploit the spoofer's ignorance of the cryptographic material when forging a signal. They can be applied to the spreading codes [4], [5] or to the navigation data bits, which is known as navigation message authentication (NMA). Although, in theory, a signal can still be replayed [6], NMA facilitates the detection of such attacks [7] and provides very good protection against other common attack methods.

Galileo, the European GNSS, is the first GNSS to provide authentication for its civil signals, implementing its own NMA-based protocol called open service navigation message authentication (OSNMA). This protocol is the one used in the research presented in this article. It was proposed in the last decade [8], has been transmitted over the past few years, and is expected to be launched operationally imminently [9].

When adding OSNMA, receivers should not experience degradation in accuracy or availability [10]. However, the time to first fix (TTFF) will be impacted. This is mainly because OSNMA is based on timed efficient stream loss-tolerant authentication (TESLA) [11], a delayed disclosure protocol, adapted to the GNSS. The data and tags act as bit commitment, and the commitment is revealed later with the transmission of the symmetric TESLA key. A characteristic of delayed disclosure protocols is the requirement of an external loose time reference, and that they allow one to use symmetric encryption algorithms. The symmetric encryption tags and keys are usually shorter than the signatures from a asymmetric encryption system, but their transmission increases the time to first authenticated fix (TTFAF) with respect to the TTFF [12]. For Galileo, the TTFF has been typically in the order of 30–60 s, although some recent improvements (the so-called *I/NAV improvements*) in the navigation message will bring it to even lower values [13].

Specifically, we will focus on *hot start* TTFAF, where the cryptographic information required to bootstrap the receiver is already known. Hereinafter, we will refer to TTFAF as hot start TTFAF. The OSNMA impact on the TTFAF has been previously analyzed in the literature. In [14], an average TTFAF reaches down to approximately 150 s including I/NAV improvements and 170 s excluding them. In [10], the lowest case comparable to this work achieves 127 s. In [9], a lowest case of 120 s is achieved, and in [15], the TTFAF of 90 s is achieved. It is normal that these values vary, as they depend on receiver implementation, which was not optimized to reduce TTFAF. We believe that TTFAF optimization is relevant for potentially many OSNMA users and is the focus of this article. We propose several strategies to reduce OSNMA TTFAF down to 44 s in the lowest case and test them in different environments.

To implement the proposed optimizations, we used OSNMAlib [16], an open-source library that implements the OSNMA protocol. We developed this library in 2022 and have maintained it since then. As the library is written in Python, it is easy to modify and extend for research purposes, even though it might not be suitable for embedded purposes.

OSNMAlib is not a receiver by itself; therefore, it needs a GNSS receiver to track satellites and decode the navigation data bits. For that purpose, we used Septentrio GNSS receivers (mosaic-X5 [17] and PolaRx5TR [18]) to collect all the necessary data, whose logging format is already integrated into our library.

The main contributions of this article can be summarized as follows.

1) We propose two ways to improve the TTFAF: page-level processing and cutoff point issue of data (COP-IOD) optimization. The first approach, initially designed by Damy et al. [19], is to extract partial information from broken subframes. The second idea goes even further and allows the reconstruction of missing navigation data by the innovative use of new OSNMA fields to improve the TTFAF significantly.
2) We validate these optimizations in three relevant scenarios using real data. The scenarios are diverse (open-sky, soft urban, and hard urban) to show that the two proposed methods are very complementary, and both ideas are needed to enable robust gains in all scenarios. We also evaluate the ideas using the official OSNMA test vectors.
3) We analyze the OSNMA cross-authentication algorithm and the implications it has in the TTFAF when leveraging on the COP-IOD optimization.
4) We provide an open-source implementation of the methods described in this article in the OSNMAlib library.

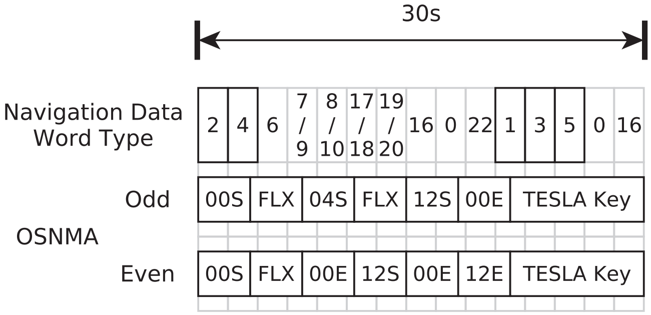


Fig. 1. Operational configuration of Galileo navigation data and OSNMA data for one subframe. Some WTs alternate between even and odd subframes. The WTs with bold borders are used in the ADKD0 authentication. In this representation, the authentication tag size is 40 bits and the TESLA key size is 128 bits.

The rest of this article is organized as follows. Section II provides a general description of the OSNMA protocol and a brief summary of the OSNMAlib library. In Section III, the hot start TTFAF process and the proposed optimizations are detailed. Section IV elaborates on security considerations and theoretical improvement of the optimizations. In Section V, the test scenarios used are described, and then, the test results are presented and discussed in Section VI. Finally, Section VII concludes this article and presents further improvement ideas.

II. Galileo I/NAV, OSNMA and OSNMAlib

A. Galileo I/NAV and OSNMA

Galileo OSNMA is transmitted in the I/NAV message, E1-B signal component [20]. The E1-B I/NAV message is composed of 30-s subframes of 15 2-s pages, each page including a word type (WT). WTs 1–5 contain the satellite ephemerides, ionosphere model, and health flags, and WTs 6–10 include time parameters (the latter shared with almanacs). There are other WTs, including only almanacs (WTs 7–9) and spare words (WT 0). As part of the I/NAV improvements mentioned, Galileo has recently added new WTs: WT 16 with a reduced ephemerides and WTs 17–20 with page recovery through Reed–Solomon, which can be useful for OSNMA, but we leave outside of our analysis for now. The WT order inside a subframe is represented in Fig. 1.

OSNMA is inserted in Galileo's E1-B page in a 40-bit field, which is transmitted every 2 s. As mentioned, OSNMA uses the TESLA protocol, with some variations and features, such as key chain sharing across transmitting satellites and cross authentication. The OSNMA 40-bit field is divided into the so-called Header and Root Key (HKROOT) section, of 8 bits, and the Message Authentication Codes and Key (MACK) section, of 32 bits. In this work, we focus on the latter, which is the most relevant one for the hot start TTFAF.

In the MACK section, six truncated message authentication code (MACs), or *tags*, are transmitted, preceding a key that authenticates the tags in the previous subframe (see Fig. 1) . Each tag has 40 bits, and it incorporates a 16-bit *tag-info* section, which encodes the satellite number the tag applies to and the type of authentication. At the moment, there are three types of tags, defined by the so-called authentication data and key delay (ADKD) parameter. ADKD0 and ADKD12 authenticate WTs 1–5, but ADKD12 with a key transmitted 5 min later to relax the receiver loose sync requirement, and ADKD4 authenticates the time (WTs 6 and 10).

Due to system limitations, not all satellites can transmit OSNMA data at the same time. We refer to a satellite transmitting OSNMA as *connected* and a satellite not transmitting OSNMA as *disconnected*. To solve this limitation, OSNMA transmits cross-authentication tags that enable the authentication of disconnected satellites. The ADKD0 cross-authentication tag positions are named `00E` in Fig. 1. There are also flex positions (`FLX`), whose tag type is not predefined and needs to be verified at runtime, which are currently only used for ADKD0 cross-authentication tags. Therefore, there are three ADKD0 cross-authentication tags on each subframe.

For further details, a broad explanation of OSNMA is provided in [21] and the full OSNMA specification can be found in the OSNMA signal-in-space interface control document (SIS ICD) [22].

B. OSNMA Time Synchronization

For OSNMA to work securely, the receiver must know its synchronization accuracy with respect to Galileo System Time (GST). This requirement comes from the use of the TESLA protocol: when a TESLA chain key is disclosed, all the previous cryptographic material can be trivially forged. This implies that the receiver must have collected all the navigation data and associated tags before the appropriate TESLA chain key is revealed by the system.

The time synchronization requirement for OSNMA is defined as T_L and set to 30 s: the time between the last bit of a tag and the first bit of the TESLA chain key authenticating it. A receiver that is not able to guarantee this T_L cannot use ADKD0 or ADKD4 tags. However, it may use ADKD12 tags if time synchronization is better than $T_L + 300$ s.

A receiver has several strategies for obtaining time synchronization with respect to the GST. If the receiver has no previous time information, it can retrieve the time from an external clock or from a secure Network Time Protocol connection. In both cases, the time needs to be converted to estimate the GST and take appropriate measures to handle the associated uncertainty. If the receiver already has time estimation and maintains it using an internal clock, the stability of the clock should be taken into account when verifying the time synchronization requirement. Further details and detailed procedures can be found in [23] and [24].

Not complying with the synchronization requirement allows for arbitrary forgery attacks. For a normal OSNMA usage, T_L is set as 30 s. However, we will use tighter time synchronizations of 17 and 25 s for some of the optimizations described in this work.

C. OSNMAlib

OSNMAlib [16], [25] is an open-source library written in Python that implements the OSNMA protocol. The library can be integrated into existing receivers and applications to incorporate NMA into the position velocity and time (PVT) calculation. It can read the Galileo I/NAV pages from an input, store the navigation and authentication data, perform the verification operations, and report the status. The library supports cold start, warm start, and hot start procedures.

The input required for OSNMAlib to work is the navigation data bits from Galileo E1-B I/NAV message as nominal page, the GST of the page transmission, and the Satellite Vehicle ID (SVID) to which the navigation data bits belong. Currently, OSNMAlib has the following input modules.

1) *Septentrio SBF:* Postprocess files or live data in real time from a Septentrio receiver in Septentrio Binary Format (SBF) if it contains the GALRawINAV block.
2) *u-blox UBX:* Postprocess files or live data in real time from a u-blox receiver in UBX format if it contains the UBX-RXM-SFRBX message.
3) *GNSS-SDR:* Process the output of the GNSS software-defined receiver project [26] from a user datagram protocol (UDP) socket.
4) *Galmon network:* Connect to the Galmon network [27] to process aggregated data from multiple receivers.
5) *Android GnssLogger App:* Postprocess the log files generated by the GnssLogger app for Android smartphones.

The library reports the OSNMA data received, the verification events, and the authenticated navigation data in chronological order. These logs also indicate when the receiver has enough authenticated data to calculate the first authenticated fix, together with the time elapsed since it started to process information. This logging option can be used to obtain the TTFAF value under different protocol configurations. Finally, the library also has a status logging every subframe in JSON format, which is useful for seeing the general state of OSNMA and extracting statistics about the scenario being processed. The status logging is used in the OSNMAlib web page to display live information of the OSNMA protocol [28].

III. PROPOSED TTFAF OPTIMIZATIONS

For standard (unassisted) TTFF, the user needs to acquire and track signals, and decode the ephemerides (WTs 1–5) from at least four satellites, and time (WTs 6 and 10) from at least one. For TTFAF, the receiver also needs to receive the tags authenticating each of the above, and a

TESLA key in the next subframe. Therefore, a delay is introduced.

For simplicity, we use the shorthand TTFAF to refer to TTFAF *hot start*, i.e., when only the authentication tags and one TESLA key are needed, and the receiver has the cryptographic information to authenticate the key with a so-called root key already in its possession. The root key is expected to last for several months; hence, it can be loaded to the receiver or reused from a previous execution. This is the standard operation mode and the focus of our article.

Another start state is *warm start*, where the receiver does not have the root key stored, but it has the public key needed to authenticate it. The receiver, then, needs to first retrieve the root key from the navigation data. The last start state is *cold start*, where the receiver only has in its possession the Merkle Tree root hash needed to authenticate the public key. The public key is expected to last for several years and is transmitted every 6 h.

The *warm start* and *cold start* states are out of the scope of this article since their TTFAF is bounded by other constraints. Nonetheless, the optimizations we describe can still be applied retroactively to the navigation and OSNMA data stored by the receiver once it retrieves a root key and is able to interpret it.

A. Page-Level Tag and Key Processing

At first glance, it may seem that OSNMA works at a per-satellite subframe level. The HKROOT is transmitted in numbered blocks that last one full subframe, and these subframe blocks need to be reordered to reconstruct the full message from multiple satellites. On the MACK side, a TESLA key is transmitted on every subframe to authenticate the tags of the previous subframe, and the tag order inside a subframe must be verified.

However, to optimize the performance of OSNMA, a more granular approach should be taken. A Galileo subframe lasts 30 s and comprises 15 pages of 2 s each. Discarding all well-received pages of a subframe because the receiver missed one of them is not the most optimal method. The intelligent use of these pages in challenging environments was first proposed in [19] and [29] and can lead to the recovery of more OSNMA tags and lower TTFAF values. The page-level processing implementation used in this work is similar to the one already described in [19] but is evaluated using the current OSNMA configuration, which includes flex tags that change the optimization's performance. Moreover, our postprocessing technique with a complete OSNMA receiver (further described in Section VI) allows us to obtain fine-graded TTFAF values that take into consideration all the current nuances of the OSNMA protocol with respect to navigation data.

The page-level processing technique consists of two ideas. The first idea is to extract tag sections from correctly received pages of partially corrupted subframes. For the secure use of OSNMA, the tags' order within a subframe must still be verified using the MAC lookup table or the MAC sequence value for the flex tags. Yet no flex tag may, in principle, be used in a subframe if the MAC sequence value or any other flex tag is missing. Consequently, a clear downside of having multiple flex tags in a MAC lookup table configuration is that this optimization will lose efficacy.

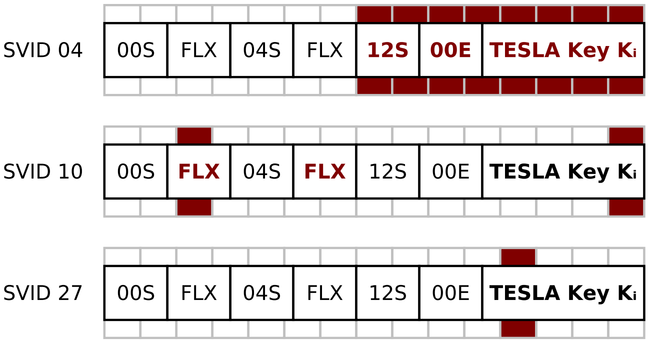


Fig. 2. Missing a page (colored in red) affects part of the cryptographic data of the subframe, but the rest is still valid. Missing one flex tag means that all flex tags are missed because their position cannot be verified. The TESLA key is the same for all satellites, so the receiver can reconstruct it by combining pages.

The second idea is to reconstruct the TESLA key by exploiting the diversity in the transmission. During a strong fading and poor visibility scenario, the receiver may not be able to fully retrieve the TESLA key from any satellite in view during one subframe. Nonetheless, that does not mean the TESLA key of that subframe is lost. Since all Galileo satellites transmit the same key during the same subframe, it may be possible to reconstruct the key using correctly received pages from different satellites.

In Fig. 2, we show an example of how page-level processing helps extract valid cryptographic data. The tag sequence and key and tag sizes correspond to the OSNMA parameters transmitted during the OSNMA operational phase, illustrated in Fig. 1. The figure depicts a subframe where satellite 04 moves out of sight, and the receiver misses the last few pages of the subframe. Nevertheless, the first four tags are perfectly useful. Satellite 10 misses a page corresponding to a flex tag, which affects the other flex tag, but the other four tags are valid. Both Satellite 10 and Satellite 27 miss a page of the key, so the subframe ends without any key fully received. However, the optimization is able to reconstruct the key because the satellites missed a different page.

Naturally, these optimizations are especially useful in scenarios with interference or fading where satellites are frequently out of sight. In a perfect open-sky scenario, only the low-elevation satellites entering or leaving the tracking horizon may have incomplete subframes.

B. Issue of Data (IOD) Navigation Data Link

Verifying the ADKD0 tags involves retrieving navigation data, followed by the corresponding tag for these data in the subsequent subframe, and finally acquiring the TESLA key used for generating the tag in the third subframe. With this approach, a TTFAF of 90 s can be achieved as the lowest time, but if the receiver misses the first pages of the first

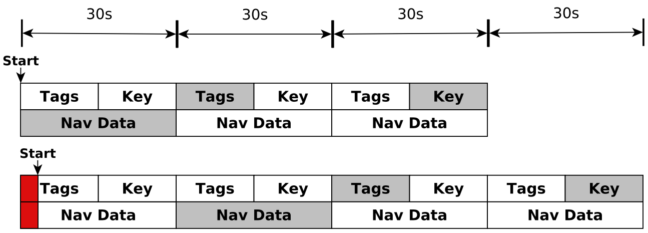


Fig. 3. Depiction of the OSNMA navigation data authentication process without any optimization for one satellite. The top row indicates the OSNMA data received and the bottom the navigation data; the gray elements are used together to authenticate the navigation data. If the receiver does not start aligned with a subframe, it has to wait until the next.

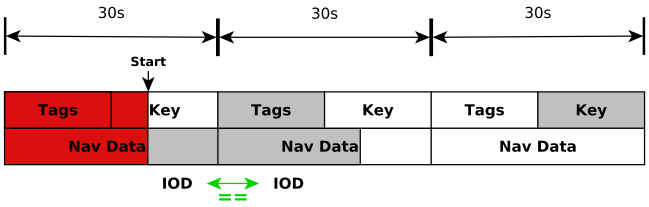


Fig. 4. Depiction of the authentication data process for one satellite if the IOD of the data from the two subframes is the same. In this case, the missed navigation data can be retrieved from the next subframe.

subframe, it has to wait until the next one to start with the process, hence delaying the TTFAF to a maximum of 119 s. Fig. 3 exemplifies these two cases for a single satellite. The top row indicates the OSNMA data, and the bottom row is the navigation data; both are transmitted in parallel. For any case between the lowest and highest values, Fig. 8 shows the TTFAF values depending on where the receiver starts in a subframe.

However, the ephemerides authenticated in ADKD0 change at a low rate and may be transmitted identically in several subframes. The data of multiple subframes can, therefore, be aggregated for authentication as long as they are the same. As discussed in the previous OSNMAlib paper [25] and the OSNMA receiver guidelines [24], one way to reconstruct the navigation data from different subframes unambiguously is to use the IOD value transmitted in the I/NAV words, except WT 5, which does not have an IOD. Hence, it must be assigned based on the IOD of other words of the subframe.

With this optimization, the lowest case occurs when the receiver starts processing navigation data immediately before WT 3 because it is the latest word containing the subframe IOD. WT 3 is transmitted 8 s before the end of the subframe, and we will have to wait for another subframe for the tags and an additional one for the key. Therefore, the lowest TTFAF is 60 s. If the navigation data do not change, the worst case occurs when the receiver starts immediately after WT 3 with a TTFAF of 97 s. A general example of this optimization is shown in Fig. 4, and Fig. 8 shows the TTFAF values for the IOD optimization as a function of subframe offset.

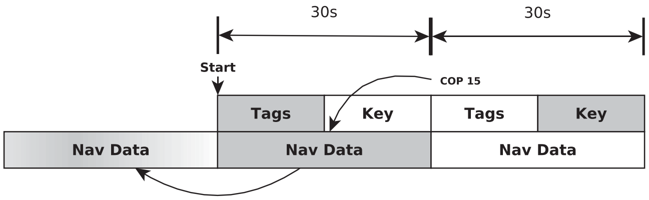


Fig. 5. If there is a tag in the key subframe that authenticates the navigation data with a COP higher than 1, the data received in the tags subframe are the same as the previous subframe and can be used to authenticate the first tag.

C. Cutoff Point (COP) Tag-Data Link

Originally, every tag included a 4-bit truncated IOD to link the tag with the data [30]. However, the unpredictability of the IOD evolution in the system could lead to failed authentications if not appropriately handled. After some years in which the field was defined as “Reserved,” the last OSNMA specification has replaced this field by the 4-bit COP field [22]. The COP indicates for how many subframes, the navigation data authenticated with the tag have not changed. A value of 1 means that the authentication tag can only use navigation data from the previous subframe. A value of 15 (the maximum possible) indicates that the authentication tag can be verified using navigation data from the 15 previous subframes.

Although the original intention for the COP is to link the tag transmitting it with data from the previous subframes, we propose to use it to link other tags with the same data. With the traditional OSNMA approach, the receiver can never use the tags of the first subframe because the data transmitted in the previous subframe are unknown. However, this is the exact information given by the COP. If the navigation data have not changed, the COP of the tags in the key subframe will be greater than 1, indicating that the navigation data in the tags subframe are the same as in the prior subframe. Therefore, we can unambiguously link the tags received in the first subframe with the data of the first subframe (see Fig. 5).

Nevertheless, for this optimization to work, the receiver must get one tag in two consecutive subframes for the same navigation data. The tag received in the first subframe is used to authenticate the navigation data when the key is disclosed in the second subframe. The COP of the tag received in the second subframe is used to verify that the data received in the first subframe can be linked with the first subframe’s tag.

By using the COP value, it may seem that the previously discussed IOD optimization is no longer beneficial. However, both can be merged for even better TTFAF results. The same IOD logic to link navigation data from two subframes can be combined with the information provided by the COP, as depicted in Fig. 6. The IOD links the navigation data from two subframes, and the COP shifts that data to the previous subframe, linking it with the tag.

The operations performed by a receiver implementing the COP and IOD optimization are as follows.

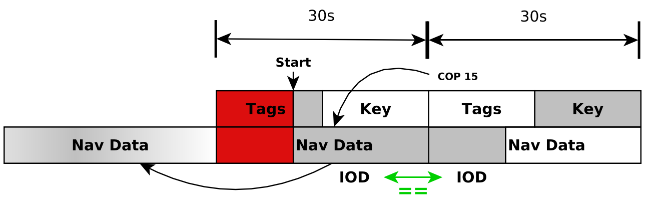


Fig. 6. IOD value can be used to bind navigation data of different subframes, and the COP value can be used to ensure the link between the navigation data and the authentication tag.

1) The receiver powers up in the middle of subframe SF_j, in time to get WTs 1, 3, and 5 from all Galileo satellites in view.
2) At the end of SF_j, the receiver has also extracted a few cross-authentication tags from connected satellites. These tags authenticate navigation data transmitted at the previous subframe (SF_{j-1}), data that the receiver missed because it was not powered ON.
3) During the next subframe (SF_{j+1}), the receiver gets all the WTs from all satellites in view. For each satellite, if the IOD of these WTs is the same as the IOD of the WTs received at SF_j, the partial navigation data received at SF_j can be fully reconstructed.
4) Then, the receiver looks at the COP value of the authentication tags extracted during SF_{j+1}. If the COP value is greater than 1, it means that the reconstructed data for SF_j is the same as the navigation data transmitted at SF_{j-1} for the satellites targeted by the tags.
5) At this moment, the receiver knows that the navigation data transmitted at SF_{j-1}, has the tags to authenticate them (received at SF_j), and has the TESLA key to verify them (received at SF_{j+1}). Therefore, it can proceed with navigation data verification.

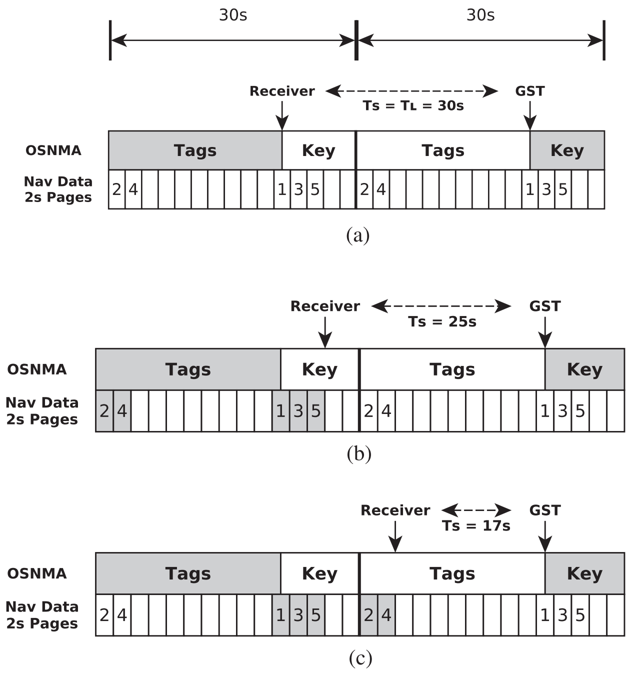
Combining the COP and the IOD, we obtain, in the lowest-case scenario, a TTFAF of 44 s on the even subframes or 46 s on the odd subframes. The position of the last cross-authentication tag in the tag sequence (see Fig. 1) defines the lowest possible TTFAF. If the navigation data do not change, the worst TTFAF is 73 s, when the receiver starts just after the last cross-authenticating tag.

We note that this optimization enables an acceptable forgery discussed in Section IV-C.

IV. FURTHER CONSIDERATIONS

A. Tighter Time Synchronization Requirement for Optimizations

The proposed optimizations in Sections III-B and III-C require time synchronization with respect to GST lower than T_L to work in a secure way. The OSNMA receiver must get all the authentication tags and navigation data before the corresponding TESLA chain key is disclosed to the system. With the optimizations, we use navigation data words transmitted closer to the TESLA key than T_L, thus requiring tighter time synchronization. We define the time synchronization parameter T_S as the maximum time synchronization the receiver can guarantee, independently of the method used to calculate it.



(a)

(b)

(c)

Fig. 7. Calculation of the maximum time synchronization (T_S) value for each optimization. The downward arrows indicate the time as perceived by the receiver and the GST. The darker color indicates the elements used for the authentication. (a) No optimization. Navigation data used from the previous subframe. (b) IOD optimization. (c) COP and IOD optimization.

The different T_S values are graphically shown in Fig. 7 for one satellite. The time as perceived for the receiver and the GST time are indicated as downward arrows, and the material used for the authenticated is indicated in gray color.

For the IOD optimization described in Section III-B, the receiver must be synchronized with the GST with a T_S of 25 s to achieve maximum performance. The value of 25 s corresponds to the time between the last bit of the last relevant navigation data word for ADKD0 (WT 5) transmitted in the tag subframe and the first bit of the TESLA key [see Fig. 7(b)]. The IOD optimization would work with a time synchronization of T_L, but it could only link WTs 2 and 4 with the navigation data of the previous subframe.

When using both the COP and the IOD to optimize the TTFAF, as described in Section III-B, T_S needed is 17 s. This value is the time between the last bit of the last relevant navigation data word for ADKD0 transmitted in the key subframe and the first bit of the TESLA key [see Fig. 7(c)]. Although with a T_S of 1 s, it would also be possible to use WT 1, we decided to discard the case because the WT is transmitted simultaneously as the key.

Aside from the not-optimized case, the rest of T_S calculations depend on the key size, the tag size, and the number of tags transmitted on each subframe. For these results, we used the configuration transmitted during the data recording for this article on 3 December 2023, which is the same configuration used for the operational phase of OSNMA (see Fig. 1).

Note that the navigation word order in the examples is extracted from the Galileo OS SIS ICD I/NAV Nominal subframe Structure for the E1-B signal [20], which is only indicative. Also, a multifrequency receiver capable of decoding the I/NAV stream from the E5b-I signal would get different values for the TTFAF and T_S.

A receiver implementing these optimizations must take into account its T_S and enable optimizations accordingly. In the case of OSNMAlib, the user may specify a time synchronization value different than T_L, and the library will only use the tags and optimizations that are cryptographically secure for the value. Currently, the library does not support to change T_S after it starts to run; hence, the receiver must ensure a lower value than the specified during the whole execution.

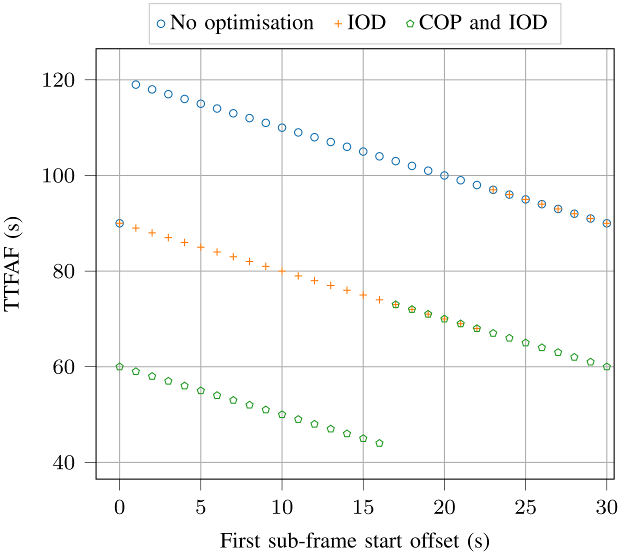


Fig. 8. Theoretical TTFAF values in an ideal scenario for the cases without optimization, with the IOD optimization, and with the COP-IOD optimization. The start time of the receiver within a subframe determines how long it will wait to get the first authenticated fix. The subframe start offset is relative to the first E1-B subframe from which the receiver starts decoding navigation data.

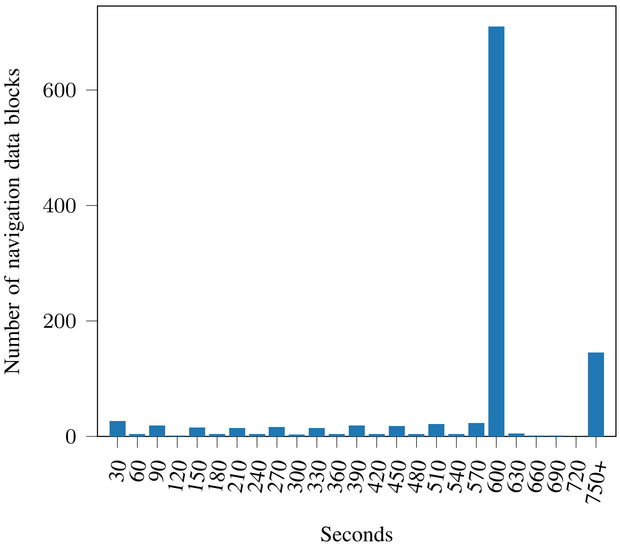


Fig. 9. Transmission of the same navigation data from each satellite in view during a 24 h recording. Most of the time, the same navigation data are transmitted for more than 600 s (20 subframes).

B. Optimization Theoretical Improvement

The page-level tag and key processing optimization (see Section III-A) is scenario specific, and its performance improvement will be determined by which pages the receiver misses. However, the TTFAF improvement of the tag-data link optimizations (see Sections III-B and III-C) can be analyzed from a theoretical point of view.

For this exercise, we will analyze the theoretical TTFAF value depending on the start time of the receiver inside a subframe for three cases: the basic OSNMA without any optimization, the IOD optimization, which is already state of the art, and our newly proposed COP and IOD optimization. We will consider a single-frequency receiver (E1-B only) in an ideal open-sky scenario with four satellites in view, no pages lost, and no change in the navigation data.

The results are shown in Fig. 8 with the TTFAF value for the described optimizations as a function of offset of the first E1-B subframe for which the receiver starts getting navigation data.

However, since the effectiveness of the optimizations is linked to the navigation data remaining the same between subframes, we have empirically analyzed how often the navigation data change for each satellite. For this purpose, we have used 24 hours of data from an open-sky receiver and calculated the duration of each block of navigation data (identified by the same IOD). The results shown in Fig. 9 clearly indicate that the majority of the time the navigation data get updated after 600 s or more, which is 20 subframes.

For a more fine-graded approach, we have calculated the probability of the IOD optimizations working on any given subframe for a satellite and for a receiver. The distinction is that, while the navigation data may change for a satellite, the optimization will still work if it remains the same for at least four of them. Nonetheless, we have observed that the navigation data usually change simultaneously for several satellites.

After analyzing 24 h of data recordings, the probability of the IOD optimizations working for a satellite on any given subframe is 96.28%, and the probability of an OSNMA receiver being able to use the optimization on any given subframe is 97.78% (see Table I).

C. Acceptable Forgeries

There is a security consideration worth discussing with the COP link optimization described in Section III-C: by using the COP value of the tags in the key subframe, we use unauthenticated information.

The receiver will accept 30-s-old forged data in the event of a change in navigation data if the adversary modifies the COP value of the current subframe tag and transmits the previous subframe data. In this case, the authentication will

TABLE I
IOD Optimization Success Rate for a Given Subframe From a 24-h Recording

	All Satellites	OSNMA Receiver
Total Epochs	29239	2880
Non-optimized Epochs	1087	64
Optimization Success (%)	**96.28**	**97.78**

The optimizations do not work if the navigation data IOD changes. For the OSNMA receiver case, the optimization has to work for at least four satellites.

pass because the receiver reconstructs the navigation data block using correct data for the received tag (the attacker cannot forge a tag), but the applicability of the data will be 30 s off because it was not transmitted during that subframe.

However, this forging does not represent a risk in itself because, according to the Galileo System Definition Document, the navigation data have a validity of 4 h without degrading the system performance [31]. Moreover, if there was no attack, the receiver would not be able to authenticate any data because the optimization does not work when the navigation data change. The adversary is allowing the receiver to have an authenticated fix it would otherwise not have.

Nevertheless, the forging is detected later: when the tag containing the modified COP value is authenticated. If no pages are lost, this happens at the end of the next subframe (i.e., 30 s later). The adversary could try to jam the receiver and not allow it to get more navigation and OSNMA data, hence hiding the attack. However, the navigation data transmitted for the attack are valid for navigation during 4 h: the attacker has not modified the contents of the navigation data (else it would not pass the tag verification), but has only retransmitted data from the previous subframe. Hence, the receiver could use them without added risks for as long as the data are valid.

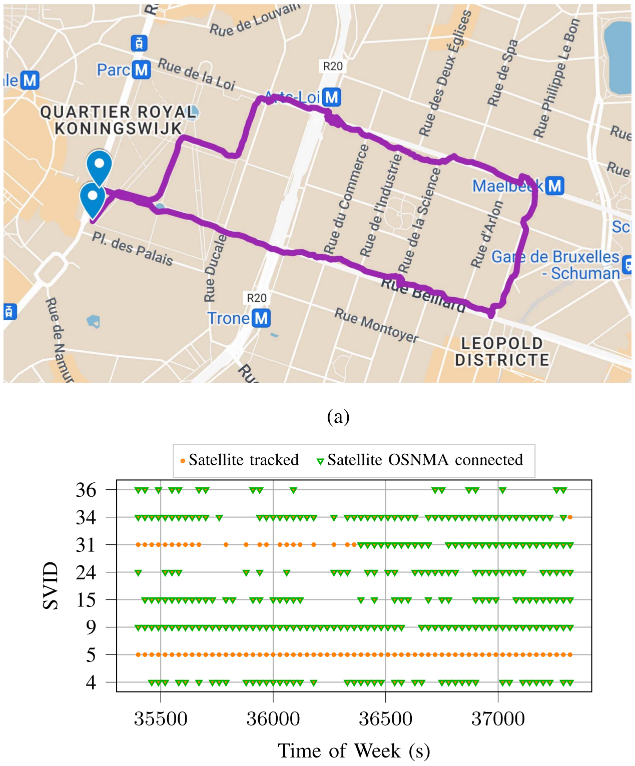
Another method to avoid the 30 s misalignment on the data applicability time would be to always relate the data to the first subframe in which a word is received, and not the second. Therefore, in the case of accepting the forged data, the validity time would start in the first subframe (where the data were actually transmitted by the system), and not on the second subframe (where the data were transmitted by the adversary).

As a final note, to perform this forging attack, the adversary must be able to replay the real Galileo signal and modify the navigation data fast enough to not fall behind the receiver's time synchronization. In such a scenario, general antireplay techniques, such as the use of partial correlations in the tracking loops [32], can also be applied to prevent forging. Finally, the attack can only be performed on the start-up of the protocol or after long interrupts, not during continuous authentications.

V. Scenarios

To evaluate the performance of the discussed optimizations, we recorded Galileo data in three relevant scenarios: a dynamic Hard Urban scenario, a dynamic Soft Urban scenario, and a static Open-Sky scenario. In addition, we have also processed configuration 2 of the official OSNMA test vectors [24] because it contains the same tag sequence as the live transmitted data. For the dynamic recordings, we used a Septentrio mosaic-X5 with firmware version 4.14.0. For the static Open-Sky scenario, we used a Septentrio PolaRx5TR with firmware version 5.5.0.

The data recordings are saved in SBF. This format contains the *GalRawINAV* block with all the information needed to postprocess the files with OSNMAlib (Galileo I/NAV message bits, SVID, and receiver GST). The recordings, containing all the GNSS logged information and format definition, are available in [33].



(a)

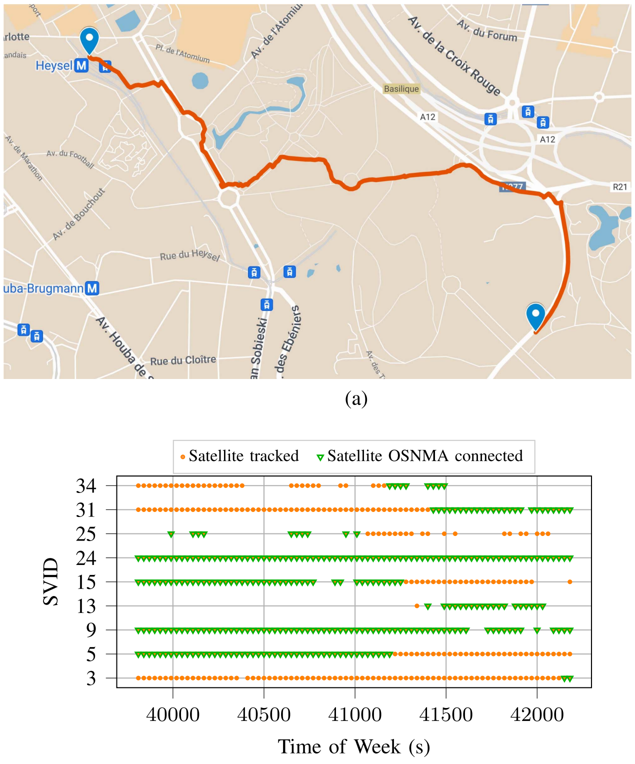
(b)

Fig. 10. Hard urban scenario recording of a walk in the European District of Brussels on 3 December 2023 from 09:50:00 to 10:22:30 UTC. (a) Trajectory followed. (b) Galileo satellites tracked and OSNMA connected.

A. Hard Urban Scenario: Brussels, European District

This scenario is a walk in the European District of Brussels, Belgium, on 3 December 2023, from 09:50:00 to 10:22:30 UTC, or GST 1267 35400 to 1267 37350. The trajectory [see Fig. 10(a)] starts at the Parc de Bruxelles and quickly heads to the urban canyon of Rue Belliart, Rue de Trèves, and Rue de la Loi. Finally, it returns to the park and ends close to the start location.

During the trajectory, the receiver got navigation data from eight different satellites [see Fig. 10(b)]. Only SVID 5 did not transmit OSNMA during the scenario; SVID 31 was initially disconnected but started transmitting OSNMA at half the scenario duration. The tracking is generally very



(a)

(b)

Fig. 11. Soft Urban scenario recording of a walk around the Atomium of Brussels on 3 December 2023, from 11:03:24 to 11:43:53 UTC. (a) Trajectory followed. (b) Galileo satellites tracked and OSNMA connected.

volatile, as it corresponds with a hard urban scenario, with several entirely lost subframes.

B. Soft Urban Scenario: Brussels, Atomium and Laeken Parks

This scenario is a walk around the Atomium and surrounding parks in Brussels, Belgium, on 3 December 2023, from 11:03:24 to 11:43:53 UTC, or GST 1267 39804 to 1267 42233. The trajectory [see Fig. 11(a)] walks close to the Atomium, enters Osseghem Park, and finally surrounds Laeken Park.

The receiver got navigation data from nine satellites during the trajectory [see Fig. 11(b)]. The number of satellites connected and disconnected is very balanced during the whole scenario, although there is a lot of change in which specific satellites transmit OSNMA.

C. Open-Sky: Leuven, Septentrio Offices

This scenario is a static recording of 60 min from the Septentrio Offices in Leuven, Belgium, on 20 December 2023, from 15:00:00 to 16:00:00 UTC, or GST 1269 313200 to 1269 316800.

The satellite visibility of this recording is excellent, as expected in an open-sky situation. A total of 11 satellites are received during the scenario, although the SVID 31 moves under the tracking horizon a few minutes in the recording (see Fig. 12). All satellites move from connected to disconnected and vice versa during the recording, but there are always at least four disconnected.

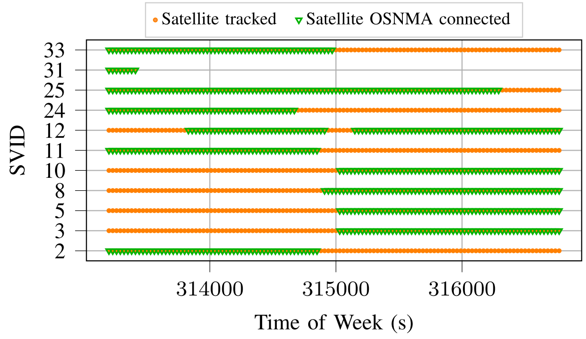


Fig. 12. Galileo satellites tracked and OSNMA connected in the Open-Sky static recording of 60 min from the Septentrio offices in Leuven, Belgium, on 20 December 2023; from 15:00:00 to 16:00:00 UTC.

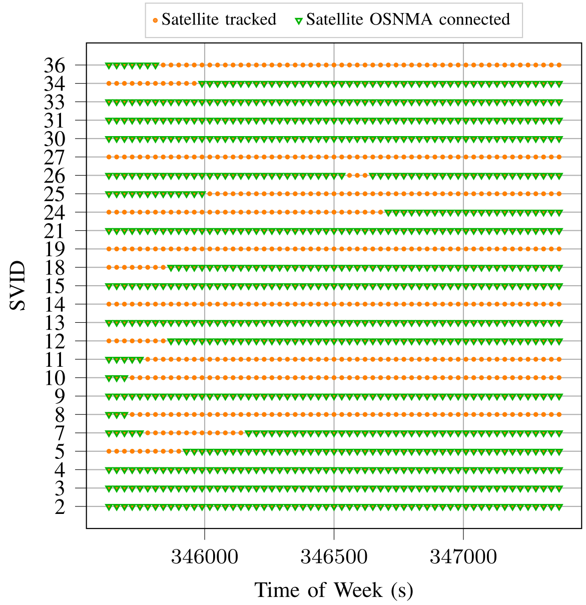


Fig. 13. Galileo satellites tracked and OSNMA connected in Configuration 2 of the test vectors from the OSNMA receiver guidelines [24]. It is a synthetic scenario with all Galileo satellites visible.

D. Test Vectors: Configuration 2

The OSNMA receiver guidelines [24] contain several test vectors to validate the implementation of the OSNMA protocol. The test vector titled “Configuration 2” contains OSNMA data with the same structure as the operational live data described in Fig. 1, so it is helpful to test and compare the optimizations. We have run the first 30 min of this test vector, simulating from July 26 at 23:59:43 to July 27 at 00:29:43 UTC, or GST 1248 345601 to 1248 347401. These test vectors must be formatted correctly to run in OSNMAlib because they are not chronologically sorted in their original format.

A particular characteristic of the test vectors is that they contain data from 25 Galileo satellites, which is impossible in a live recording (see Fig. 13). Moreover, they emulate a

TABLE II
TTFAF Metrics Using the IOD Data Link Optimization With a T_S of 30 s

Optimization	**Lowest (s)**	**Average (s)**	**P95 (s)**
Test Vectors	70.0	84.5	98.0
Open-Sky	70.0	84.5	98.0
Soft Urban	70.0	127.5	248.0
Hard Urban	70.0	266.1	427.0

perfect reception with no pages lost. Therefore, while we cannot directly extrapolate the results to a real scenario, they are useful to validate if the tag-data link optimizations work.

VI. Test Results

We implemented the optimizations in OSNMAlib in a flexible way so that they can be turned ON or OFF at choice. To obtain multiple TTFAF values from the continuous recordings, we replayed the logs in OSNMAlib but started to process them each time 1 s later. With this technique, we can emulate a receiver powering up at any moment of the recording and obtain all the TTFAF values needed to evaluate the optimizations. Therefore, the number of data points is directly the number of seconds on each scenario.

We decided to group the described optimizations into three accumulative groups to visualize their effects easily.

1) *Standard OSNMA:* It uses the IOD optimization to regenerate navigation data and the default T_S set to T_L (30 s). While a standard OSNMA may not include the IOD optimization, it is briefly described in the OSNMA ICD, was present in the first version of OSNMAlib, and is already used in other state-of-the-art implementations. Hence, we use this configuration as a baseline.
2) *Page-level processing and tighter time synchronization:* It uses the IOD optimization, a T_S of 25 s to use the IOD optimization at its full potential, and the page-level processing technique to extract valid navigation data from broken subframes.
3) *COP and IOD, with page-level processing and tighter time synchronization:* It uses the COP-IOD optimization to regenerate and propagate navigation data, a T_S of 17 s to use completely the COP optimization, and page-level processing.

The results are presented in a cumulative distribution function (CDF) for each scenario to provide a global view of the optimization performance in Fig. 14. In addition, in Fig. 15, we present the minimum TTFAF value obtained in each subframe to evaluate how the optimizations improve the TTFAF at different time periods. Finally, for each of the three tested optimization combinations, we have chosen the lowest, average, and percentile 95 values as relevant TTFAF metrics and displayed them in Tables II–IV.

TABLE III
TTFAF Metrics Using the IOD Data Link Optimization With a T_S of 25 s and Page-Level Processing

Optimization	**Lowest (s)**	**Average (s)**	**P95 (s)**
Test Vectors	68.0	82.5	96.0
Open-Sky	68.0	82.5	96.0
Soft Urban	68.0	94.1	137.0
Hard Urban	68.0	151.1	318.5

TABLE IV
TTFAF Metrics Using the COP-IOD Tag-Data Link Optimization With a T_S of 17 s and Page-Level Processing

Optimization	**Lowest (s)**	**Average (s)**	**P95 (s)**
Test Vectors	44.0	60.9	75.0
Open-Sky	44.0	68.8	87.0
Soft Urban	54.0	87.5	129.0
Hard Urban	60.0	146.1	305.0

A. Page-Level Tag and Key Processing

The page-level processing optimization works as expected: it improves the TTFAF in cases where Galileo I/NAV pages are lost. The two urban scenarios show a clear improvement between the case with page-level processing and the case without it (see Fig. 14). Due to the buildings and trees, nearly any satellite has dropped pages at some point, and the optimization extracts all it can from the left pages. For example, in the Hard Urban scenario, nearly 80% of the TTFAF values are lower than 200 s when using page-level processing, while the TTFAF increases to 360 s for the case without this optimization. Unsurprisingly, the improvement is more significant in the Hard Urban scenario than in the Soft Urban case, where fewer pages are lost.

When looking at the minimum TTFAF value per subframe (see Fig. 15) for the same urban scenarios, the effect of the harsh environment is displayed in the form of time spikes. In some cases, the page processing optimization follows the same spike as the not-optimized case but with slightly lower values. However, when this does not happen, the improvement is substantial (for example, around time of week 37 000 in the Hard Urban scenario).

In the Open-Sky scenario and test vectors, the 2-s improvement observed in the minimum TTFAF per subframe and in the displacement of the CDF is due to the reduction of T_S to 25 s and not to the page-level processing. Reducing T_S allows linking navigation data of two subframes using the IOD of WT 3 instead of WT 1. WT 3 is transmitted 2 s after WT 1, hence the improvement of 2 s in the minimum TTFAF when reducing T_S to 25 s.

The ineffectiveness of page-level processing for the test vectors is expected: the synthetic nature of the scenario implies that no pages are lost. In the open-sky scenario, we recorded that no satellite loses pages relevant to OSNMA, which is a possible situation. Nevertheless, note that this may differ in other open-sky scenarios, where some low-elevation satellites might lose pages.

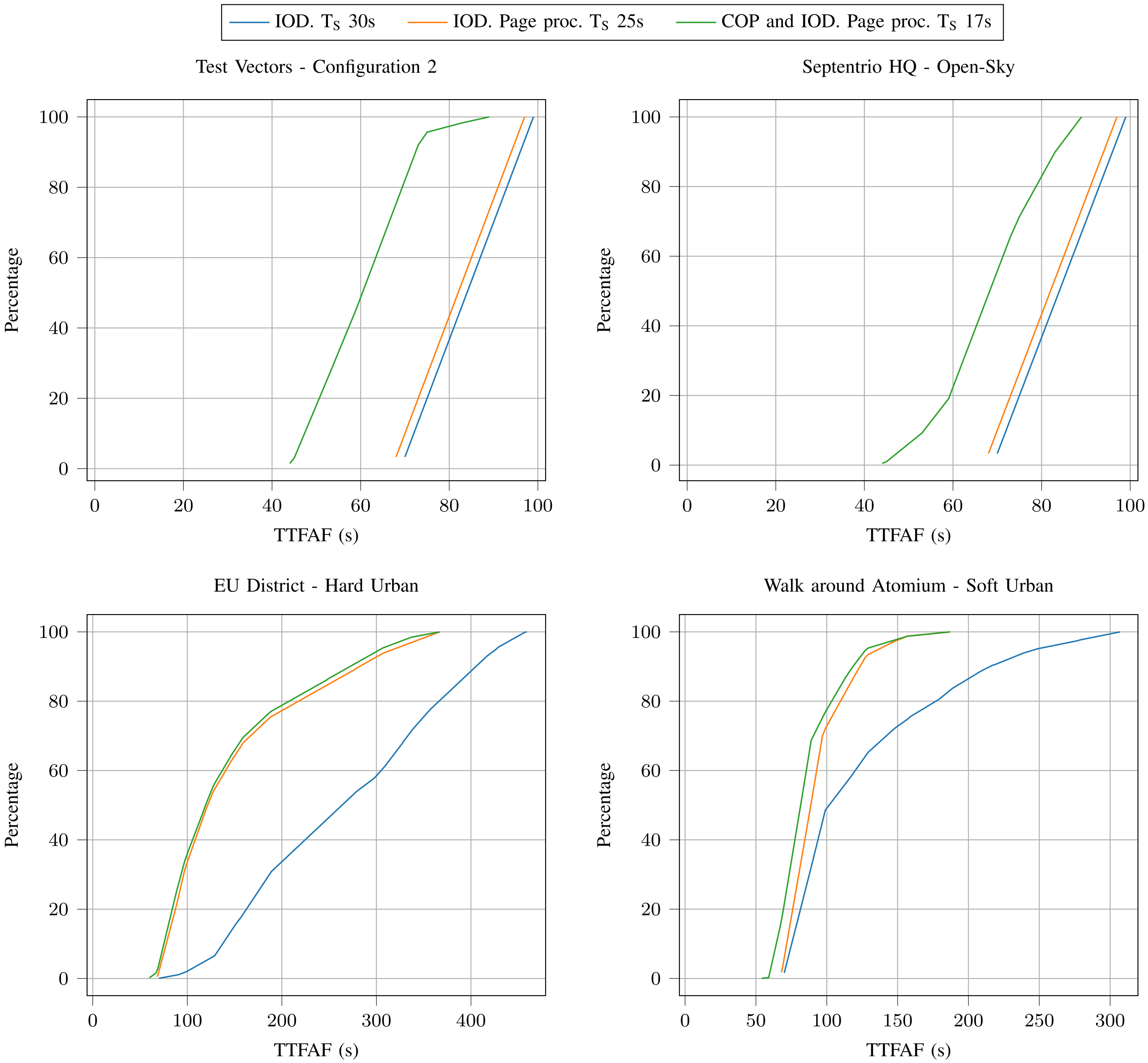


Fig. 14. Page-level processing optimization improves the TTFAF on the scenarios where pages are lost, such as the urban scenarios. The COP-IOD optimization improves as expected the TTFAF only in the scenarios where a lot of satellites are visible, while it struggles to bring any benefit in the urban scenarios.

B. COP-IOD Tag-Data Link

The COP-IOD tag-data link optimization struggles to yield any improvement in the urban scenarios (see Fig. 14). The essential requirement of obtaining two tags for the same satellite and navigation data in two consecutive subframes is hardly met due to the fading characteristic of these environments. Also, the reduced number of satellites in view makes this requirement even harder to fulfill. Still, it improves slightly more in the Soft Urban than in the Hard Urban scenario.

However, the optimization works according to the theory in the test vectors, improving the TTFAF very substantially. Some cases worse than expected can be seen as subframes with a minimum TTFAF of 60 s in Fig. 15 because the test vectors contain subframes with change of navigation data. When sufficient navigation data changes, the optimization cannot make assumptions based on the COP value for all tags, degrading the TTFAF. Despite that, it is always better than the cases with only the IOD optimization. Moreover, we can see how the lowest case for each subframe alternates between 44 and 46 s, determined by whether the tag sequence is for the odd or even subframe (see in Fig. 1 the position of the last cross-authentication tag `E00`).

Strangely, in the Open-Sky scenario, the COP-IOD optimization does not seem to work as well as theorized, even when tracking ten satellites for most of the time. The results are good; the improvement, when compared with the IOD optimization only values, is clear and huge, but we are in several subframes far away from the 44–46 s mark. In

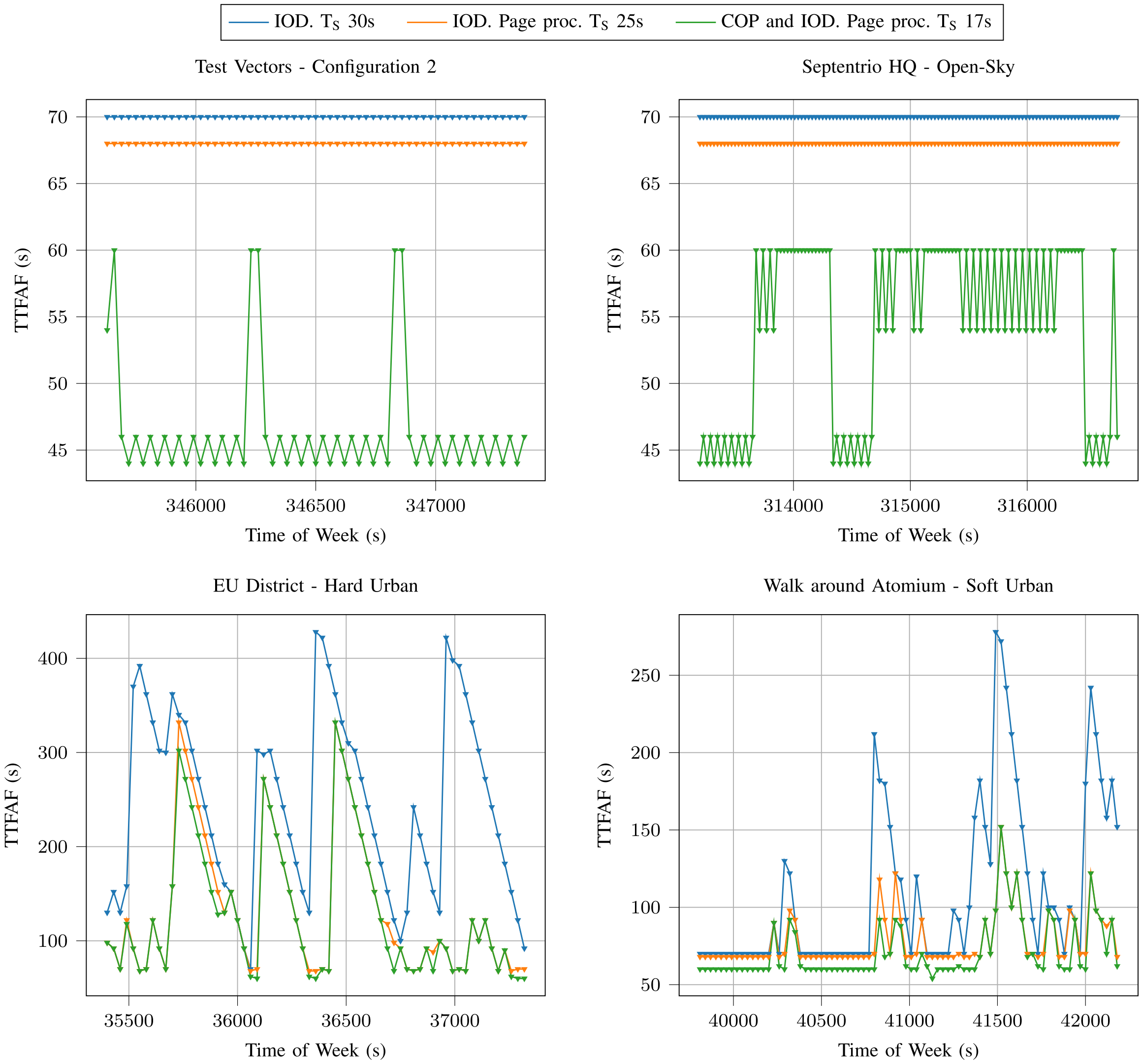


Fig. 15. Minimum TTFAF value obtained on each subframe. The theoretical minimum value with the COP-IOD optimization is 44 or 46 s. This value is only consistently achieved in the test vectors (except for the subframes with change of navigation data) and in some subframes of the Open-Sky scenario. The COP-IOD minimum value is never reached in the urban scenarios due to the high number of lost pages. However, for the same reason, the page-level processing optimization substantially improves the TTFAF values in the urban scenarios.

addition, we can see how the minimum TTFAF value for the subframes is discrete: 60, 54, 46, and 44 s. These values are directly linked to the position of the ADKD0 tags in the tag sequence, described in Fig. 1.

When a receiver implementing the COP-IOD optimization starts aligned with the beginning of the subframe, it receives four ADKD0 tags on that subframe from each connected satellite. If four of these tags are repeated in the next subframe, a TTFAF of 60 s can be obtained. This case is very likely with ten satellites in view for the Open-Sky scenario. Thus, we do not see any subframe with a minimum TTFAF greater than 60 s.

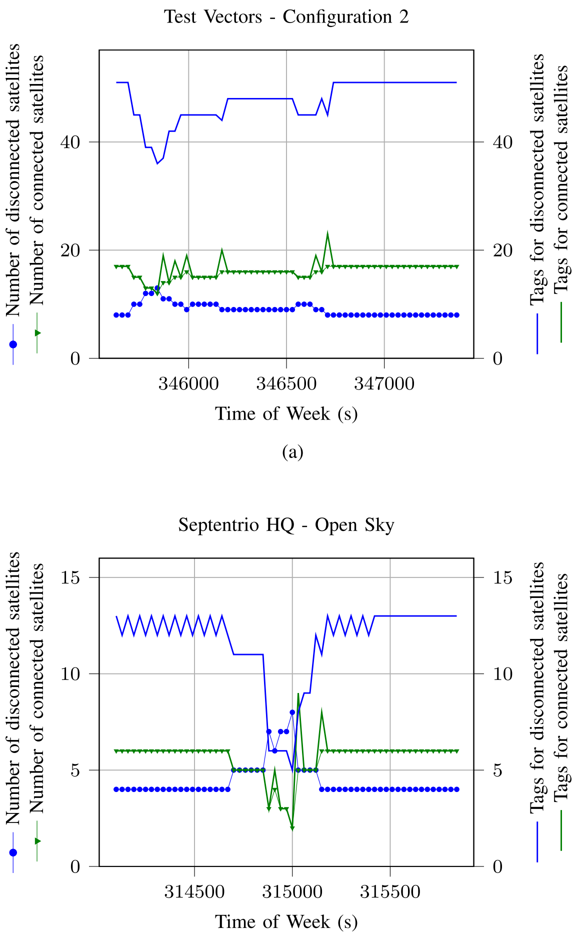
If the receiver starts later within the subframe and misses the first tag, it also loses the ability to authenticate the flex tag positions, effectively losing all flex tags. Therefore, it can only use the ADKD0 tags indicated with `00E` in Fig. 1. The discrete TTFAF values for the Open-Sky scenario in Fig. 15 are obtained when the receiver starts just before this ADKD0 tags.

Despite the identified shortcomings, the combination of IOD and COP tag-data link with page-level processing and a T_S of 17 s always gives the best results regardless of the circumstance (see Tables II–IV).

C. OSNMA Cross-Authentication Algorithm

The reason why, even in an open-sky scenario, the COP optimization is not working as well as expected lies in the OSNMA cross-authentication algorithm. Currently, OSNMA only transmits cross-authentication tags for disconnected satellites, and this behavior creates an imbalance in the number of ADKD0 tags a satellite receives during a subframe conditioned by its connection status.

If a satellite is connected, it will only receive one ADKD0 tag for the whole subframe: the self-authenticating



(a)

(b)

Fig. 16. On the left y-axis, the number of connected and disconnected satellites. On the right y-axis, the number of authentication tags received per subframe for connected and disconnected satellites. (a) shows the test vectors scenario, and (b) the open sky scenario. A connected satellite only gets one tag per subframe, while a disconnected satellite gets up to five tags per subframe on average.

tag, which is always transmitted at the first position of the tag sequence (see 00S in Fig. 1). However, if the satellite is disconnected, it will get multiple cross-authenticating ADKD0 tags from connected satellites. These tags are transmitted in the cross-authentication positions, which are currently three per subframe (see 00E and FLX in Fig. 1).

The tag unbalance becomes apparent when examining the number of tags received for connected and disconnected satellites in the test vectors and the Open-Sky scenarios (see Fig. 16). The number of tags per subframe for connected satellites is always the same as the number of connected satellites (hence, one tag per satellite). Yet, there are some subframes where there is more than one tag per satellite: when a previously disconnected satellite joins the OSNMA transmission. In those cases, because the tags are always transmitted for data in the previous subframe, the system still transmits tags for the satellite's data before the satellite starts to transmit OSNMA.

On the other hand, the number of tags received for disconnected satellites is up to five times the number of disconnected satellites in the test vectors [see Fig. 16(a)]. The ratio is a bit lower for the Open-Sky scenario [see Fig. 16(b)] because not all satellites are in view, so some tags are lost. In either case, the tags received for disconnected satellites are much more than those for connected satellites.

Another point of discussion is which disconnected satellites are selected for the cross-authentication positions. In the present OSNMA configuration, the connected satellites transmit every subframe one tag for the closest and second closest disconnected satellites, and one tag that alternates between the third and fourth closest disconnected satellites [34].

D. Cross-Authentication Algorithm Impact on the COP-IOD Optimization

Cross-authenticating only the disconnected satellites might maximize all-in-view satellite authentication with few connected satellites, but it hampers the performance of the COP-IOD optimization. Moreover, the more satellites become connected, the less tags are transmitted for satellites in view.

For example, in a seemingly good scenario with four connected satellites in view, an OSNMA receiver with the COP-IOD optimization will only be able to achieve a lowest TTFAF of 60 s, with an average of 74.5 s. This is because the only ADKD0 tag the satellites are getting is transmitted in the first position of the sequence, so if the receiver starts 2 s after the beginning of the subframe, it is sure not to receive any tag for that satellite for the rest 28 s of the subframe [see Fig. 17(a)]. However, it will still get better TTFAF values than using only the IOD optimization, with an average of 82.5 s, or no tag-data link optimization, with an average of 104.5 s.

With six satellites in view and only two of them connected, the COP-IOD optimization obtains better TTFAF values than with four connected satellites. Because the cross-authentication tag positions are situated later in the subframe, a receiver can start processing later and still receive tags for satellites in view [see Fig. 17(b)]. In this scenario, and assuming that the tags are transmitted in the last two positions, the lowest possible TTFAF is 54 s, with an average of 71.5 s. However, with the current OSNMA configuration (see Fig. 1), this scenario can only happen in the odd subframes where there are two 00E positions. In the even subframes, the FLX positions cannot be used when the receiver misses the first tag 00S.

It is not until we have eight satellites in view, four connected and four disconnected, that the COP-IOD optimization works as well as theorized. In this scenario, the four connected satellites transmit cross-authentication tags in the last position of the sequence for the four disconnected satellites [see Fig. 17(c)]. Thus, a receiver can start much later in the subframe and still receive tags for satellites in view. In this situation, the lowest possible TTFAF is of 44.0 s, with an average of 59.5 s. Paradoxically, the receiver will obtain the authenticated fix using satellites that are not transmitting OSNMA.

The discussion about the TTFAF in this section assumes that the navigation data does not change, which is true in

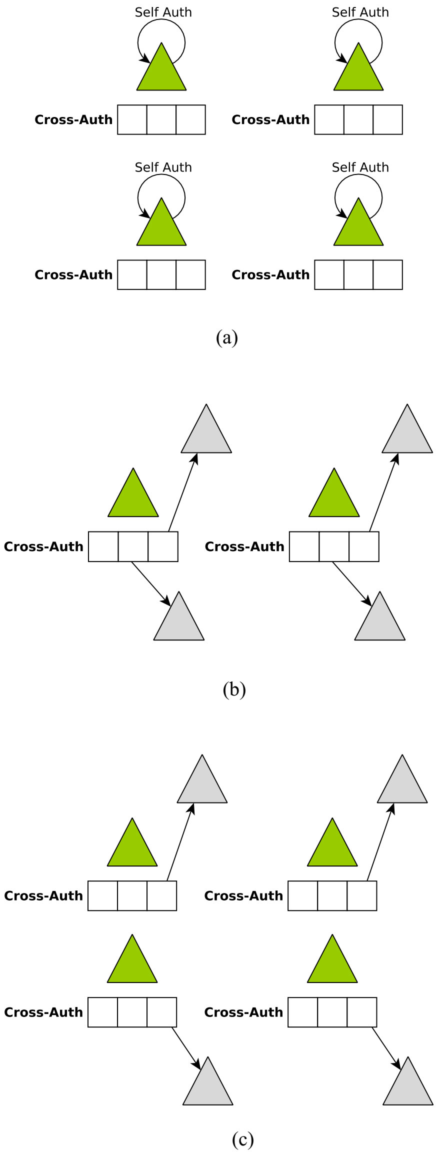


Fig. 17. COP-IOD optimization performance on multiple relevant scenarios. The triangle shapes represent satellites in view, with the green color for connected and the gray for disconnected. The arrows indicate for which satellite are the authentication tags issued. (a) With four connected satellites in view, they all self-authenticate using the first tag of the subframe. The COP-IOD optimization obtains a lowest TTFAF of 60 s. (b) With two connected and four disconnected satellites in view, the connected satellites cross-authenticate the disconnected. These tags are transmitted later in the subframe, allowing for a lowest TTFAF of 54 s. (c) With four connected and four disconnected satellites in view, the connected satellites cross-authenticate the disconnected with a tag in the last position. Hence, the lowest TTFAF can be of 44 s.

97.78% of the cases (see Table I). It also assumes that the cross-authentication tags are transmitted in an optimal sequence from the receiver perspective, which is scenario specific. Therefore, the values are a lower bound. However, it illustrates how, by enabling the cross-authentication of connected satellites, the performance of the protocol could increase, requiring less satellites in view to obtain an authenticated fix. Transmitting the self-authentication tag `00S` in the last position of the sequence could also improve the performance by allowing the receivers to start later in the subframe and still authenticate the flex tag positions.

VII. Conclusion

In this article, two concrete ideas have been proposed to improve the TTFAF: page-level processing and COP-IOD optimization. The analysis of the proposed optimizations over three distinct scenarios (Open-Sky, Hard Urban, and Soft Urban) and the test vectors shows how the TTFAF can be greatly improved by treating the navigation data received optimally. Moreover, both methods are proven to be complementary when examined in diverse environments.

The page-level processing for authentication tags and TESLA keys is extremely effective for the urban scenarios, improving the average TTFAF from 127.5 to 94.1 s in the Soft Urban scenario and from 266.1 to 151.1 s in the Hard Urban scenario. Due to the low satellite visibility and fading, the COP-IOD optimization only marginally improves the average TTFAF for the Soft and Hard Urban scenarios, obtaining 87.5 and 146.1 s, respectively.

However, the opposite occurs for the test vectors and the Open-Sky scenario: the page-level processing does not improve the TTFAF, but the COP-IOD optimization reduces it substantially. In both cases, the lack of missed pages inhibits page-level processing gains. Nonetheless, the COP-IOD optimization benefits from the good satellite visibility of the Open-Sky scenario and the ample number of satellites present in the test vectors. By using this last optimization, the average TTFAF improves from 82.5 to 60.9 s for the test vectors and 68.8 s for the Open-Sky scenario. The improvement for the lowest TTFAF value is even more impressive, from 68.0 to 44.0 s in both cases.

The COP-IOD optimization does not work entirely as expected in the Open-Sky scenario due to the cross-authentication algorithm followed by OSNMA. The algorithm never sends cross-authentication tags for satellites transmitting OSNMA, which generates an imbalance in the number of tags received for each satellite. This behavior adds extra constraints in the minimum number of satellites in view for the COP-IOD optimization to reach lower TTFAF values consistently.

To further improve the OSNMA metrics, it could be useful to implement a multifrequency library that also uses the I/NAV messages transmitted at E5b. Moreover, Galileo has recently implemented four new WTs that allow the recovery of missed clock and ephemeris pages using Reed–Solomon encoding, which can significantly improve the performance of OSNMA in urban scenarios [35].

Acknowledgment

The authors would like to thank Sibren De Bast for his insights and comments.

References

[1] L. Scott, "Anti-spoofing & authenticated signal architectures for civil navigation systems," in *Proc. Int. Tech. Meeting Satell. Division Inst. Navigat.*, 2003, pp. 1543–1552.

[2] K. D. Wesson, J. N. Gross, T. E. Humphreys, and B. L. Evans, “GNSS signal authentication via power and distortion monitoring,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 2, pp. 739–754, Apr. 2018.

[3] Ç. Tanil, S. Khanafseh, M. Joerger, and B. Pervan, “An INS monitor to detect GNSS spoofers capable of tracking vehicle position,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 1, pp. 131–143, Feb. 2018.

[4] J. M. Anderson et al., “Chips-message robust authentication (Chimera) for GPS civilian signals,” in *Proc. Int. Tech. Meeting Satell. Division Inst. Navigat.*, 2017, pp. 2388–2416.

[5] I. Fernandez-Hernandez et al., “Semi-assisted signal authentication for Galileo: Proof of concept and results,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 4, pp. 4393–4404, Aug. 2023.

[6] K. Zhang, E. G. Larsson, and P. Papadimitratos, “Protecting GNSS open service navigation message authentication against distance-decreasing attacks,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 2, pp. 1224–1240, Apr. 2021.

[7] T. E. Humphreys, “Detection strategy for cryptographic GNSS anti-spoofing,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 2, pp. 1073–1090, Apr. 2013.

[8] I. Fernandez-Hernandez, V. Rijmen, G. Seco-Granados, J. Simon, I. Rodríguez, and J. D. Calle, “A navigation message authentication proposal for the Galileo open service,” *NAVIGAT.: J. Inst. Navigat.*, vol. 63, no. 1, pp. 85–102, 2016.

[9] M. Götzelmann, E. Köller, I. Viciano-Semper, D. Oskam, E. Gkougkas, and J. Simon, “Galileo open service navigation message authentication: Preparation phase and drivers for future service provision,” *Navigation: J. Inst. Navigat.*, vol. 70, no. 3, 2023, Art. no. navi.572.

[10] L. Musumeci et al., “OSNMA user performance assessment at ESA/ESTEC—System qualifications tools and methodologies,” in *Proc. Int. Tech. Meeting Satell. Division Inst. Navigat.*, 2023, pp. 538–556.

[11] A. Perrig, J. Tygar, A. Perrig, and J. Tygar, “TESLA broadcast authentication,” in *Secure Broadcast Communication: In Wired and Wireless Networks*. Boston, MA, USA: Springer, 2003, pp. 29–53.

[12] I. Fernandez-Hernandez, T. Ashur, and V. Rijmen, “Analysis and recommendations for MAC and key lengths in delayed disclosure GNSS authentication protocols,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 3, pp. 1827–1839, Jun. 2021.

[13] M. Paonni et al., “Improving the performance of Galileo E1-OS by optimizing the I/NAV navigation message,” in *Proc. Int. Tech. Meeting Satell. Division Inst. Navigat.*, 2019, pp. 1134–1146.

[14] L. Cucchi, S. Damy, M. Paonni, M. Nicola, and B. Motella, “Receiver testing for the Galileo E1 OSNMA and I/NAV improvements,” in *Proc. Int. Tech. Meeting Satell. Division Inst. Navigat.*, 2022, pp. 808–819.

[15] T. Hammarberg, J. M. V. García, J. N. Alanko, and M. Z. H. Bhuiyan, “An experimental performance assessment of Galileo OSNMA,” *Sensors*, vol. 24, no. 2, 2024, Art. no. 404.

[16] A. Galan, I. Fernandez-Hernandez, and G. Seco-Granados, “OSNMAlib. GitHub repository,” 2024. [Online]. Available: https://github.com/Algafix/OSNMA

[17] Septentrio N.V., “Mosaic-X5 GNSS receiver module,” 2024. Accessed: 28 Feb. 2024. [Online]. Available: https://www.septentrio.com/en/products/gps/gnss-receiver-modules/mosaic-x5

[18] Septentrio N.V., “PolaRx5TR GNSS receiver,” 2024. Accessed: 28 Feb. 2024. [Online]. Available: https://www.septentrio.com/en/products/gps/gnss-reference-receivers/polarx-5tr

[19] S. Damy, L. Cucchi, and M. Paonni, “Performance assessment of Galileo OSNMA data retrieval strategies,” in *Proc. Satell. Navigat. Technol.*, 2022, pp. 5–7.

[20] *GNSS European (Galileo) Open Service, Signal-in-Space ICD, Issue 2.1*, European Union, Brussels, Belgium, Nov. 2023.

[21] I. Fernandez-Hernandez, S. Damy, M. Susi, I. Martini, and J.́ Ó. Winkel, “Galileo authentication and high accuracy: Getting to the truth,” Inside GNSS, Feb. 2023. Accessed: 5 Mar. 2024. [Online]. Available: https://insidegnss.com/galileo-authentication-and-high-accuracy-getting-to-the-truth/

[22] *GNSS European (Galileo) Open Service, Galileo OSNMA SIS ICD, Issue 1.1*, European Union, Brussels, Belgium, Oct. 2023.

[23] I. Fernandez-Hernandez, T. Walter, A. Neish, and C. O’driscoll, “Independent time synchronization for resilient GNSS receivers,” in *Proc. Int. Tech. Meeting Inst. Navigat.*, 2020, pp. 964–978.

[24] *GNSS European (Galileo) OPEN SERVIce, OSNMA Receiver Galileo Guidelines, Issue 1.3*, European Union, Brussels, Belgium, Jan. 2024.

[25] A. Galan, I. Fernandez-Hernandez, L. Cucchi, and G. Seco-Granados, “OSNMAlib: An open Python library for Galileo OSNMA,” in *Proc. Workshop Satell. Navigat. Technol.*, 2022, pp. 1–12.

[26] C. Fernández-Prades, “GNSS-SDR. CTTC. Open-source GNSS software-defined receiver,” 2024. [Online]. Available: https://gnss-sdr.org

[27] B. Hubert, “Galmon network. GitHub repository,” 2024. [Online]. Available: https://github.com/berthubert/galmon

[28] A. Galan-Figueras, C. Iñiguez, I. Fernandez-Hernandez, S. Pollin, and G. Seco-Granados, “Improving OSNMAlib: New Formats, Features, and Monitoring Capabilities,” *IEEE J. Indoor Seamless Position. Navigation*, vol. 3, pp. 117–127, 2025.

[29] S. Damy, L. Cucchi, and M. Paonni, “Impact of OSNMA configurations, operations and user’s strategies on receiver performances,” in *Proc. 35th Int. Tech. Meeting Satell. Division Inst. Navigat.*, 2022, pp. 820–827.

[30] I. Fernández et al., *Galileo Navigation Message Authentication Specification for Signal-in-Space Testing—v1.0*. Brussels, Belgium: Eur. Commission, 2016.

[31] *GNSS European (Galileo) Open Service, Service Definition Document, Issue 1.3*, European Union, Brussels, Belgium, Nov. 2023.

[32] G. Seco-Granados, D. Gómez-Casco, J. A. López-Salcedo, and I. Fernández-Hernández, “Detection of replay attacks to GNSS based on partial correlations and authentication data unpredictability,” *GPS Solutions*, vol. 25, 2021, Art. no. 33, doi: 10.1007/s10291-020-01049-z.

[33] A. Galan, I. Fernandez-Hernandez, and W. De Wilde, “GNSS recordings for Galileo OSNMA evaluation,” *IEEE Dataport*, 2024, doi: 10.21227/a0nm-kn45.

[34] A. Galan, C. O’Driscoll, I. Fernandez-Hernandez, and S. Pollin, “Sensitivity analysis of Galileo OSNMA cross-authentication sequences,” *Eng. Proc.*, vol. 88, no. 1, 2025, Art. no. 12.

[35] S. Damy, L. Cucchi, B. Motella, and M. Paonni, “Increasing OSNMA performance with Galileo I/NAV improvements: Tests in degraded reception conditions,” in *Proc. Int. Tech. Meeting Inst. Navigat.*, 2024, pp. 390–402.

Aleix Galan-Figueras (Student Member, IEEE) received the B.Sc. degree in computer engineering and the B.Sc. degree in telecommunication systems engineering from the Universitat Autonoma de Barcelona (UAB), Bellaterra, Spain, in 2020, and the M.Sc. degree in cybersecurity from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2022. He is currently working toward the Ph.D. degree in global navigation satellite system (GNSS) security and resilience with the WaveCoRE Research Group, Department of Electrical Engineering, Katholieke Universiteit Leuven, Leuven, Belgium.

During his studies, he participated in an Erasmus exchange with Katholieke Universiteit Leuven and Septentrio NV, Leuven, where he worked on his master’s thesis. During his master’s studies, he worked on a European Commission funded project with UAB to develop an open-source library for the Galileo open service galileo message authentication protocol. Then, he worked for two years in the industry with Septentrio NV on the topics of GNSS spoofing detection and software-defined radio devices.

Mr. Galan-Figueras received a Ph.D. Fellowship from the Research Foundation Flanders in 2023.

Ignacio Fernandez-Hernandez received the Ph.D. degree in electronic systems from Aalborg University, Aalborg, Denmark, in 2015.

He is currently with the European Commission, Brussels, Belgium, where has led the design and development of Galileo high accuracy and authentication services over the past few years. He also chairs the EU-US Resilience and EU Authentication and High Accuracy Working Groups. He is also an Engineer with the Higher Technical School of Engineering, Universidad Pontificia de Comillas, Madrid, Spain.

Wim De Wilde received the master's degree in electronic engineering from the University of Ghent, Ghent, Belgium, in 1999.

He then joined Alcatel Bell's research team as a System Engineer in wireline communications. In 2002, he joined Septentrio NV, Leuven, Belgium. With Septentrio NV, he has been involved in numerous global navigation satellite system receiver designs, with focus on the RF and digital signal processing section. He is currently a Team Leader of the OEM Platform Team, Septentrio NV.

Sofie Pollin (Senior Member, IEEE) received the engineering science degree and the Ph.D. (hons.) degree from KU Leuven, Leuven, in 2002 and 2006, respectively.

She is currently a Full Professor with KU Leuven focusing on wireless communication systems. She had a research position with UC Berkeley from 2006 to 2008, as a BAEF and Marie Curie Fellow. From 2008 to 2012, she was a Senior Researcher at imec, where she is currently still a Principal Member of technical staff. Her research centers around wireless networks that require networks that are ever more dense, heterogeneous, battery-powered, and spectrum-constrained. Her research interests include cell-free networks, integrated communication and sensing, and nonterrestrial networks.

Dr. Pollin is a Member of the Executive Editorial Committee for IEEE Transactions on Wireless Communications and an Associate Editor for IEEE Transactions on Mobile Computing. She is the publication and special issue Officer for the Aerial Communications Emerging Technology Initiative (AC-ETI). She was a TPC Co-Chair of the 4th and 5th IEEE Joint Communication and Sensing (JC&S) Symposium and of the 2024 EuCNC, symposium Co-Chair of Globecom 2021 (Communication Theory), Globecom 2022 (SAC AC), ICC 2024 (Communication Theory), PIMRC 2024, WCNC 2022, DySPAN 2015, and was involved in the organization of ICC 2020, DySPAN 2017, ISWCS 2015, CCNC 2016, and ACM Mobicom 2023. She was the recipient of ICC 2024 and EuCNC2024 best paper awards.

Gonzalo Seco-Granados (Fellow, IEEE) received the Ph.D. degree in telecommunications engineering from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2000, and the M.B.A. degree in business administration from IESE Business School, Barcelona, in 2002.

From 2002 to 2005, he was a Member of the European Space Agency, Noordwijk, The Netherlands, where he contributed to the design of the Galileo system. He is currently a Professor with the Department of Telecommunications, Universitat Autònoma de Barcelona, Barcelona, where he was the Coordinator of the Telecommunications Engineering Degree from 2007 to 2010 and the Vice Dean of the Engineering School from 2011 to 2019. He is also with the Institute of Space Studies of Catalonia, Barcelona, and is an ICREA Academia Fellow. In 2015, 2019, and 2022, he was a Fulbright Visiting Scholar with the University of California, Irvine, CA, USA. He is a co-founder of Loctio, a start-up offering low-energy global navigation satellite system (GNSS) positioning solutions for Internet of Things. His research interests include signal processing and signal/receiver design for GNSS, low earth orbit positioning, navigation, and timing, beyond 5G-integrated communications, localization, and sensing.

Dr. Seco-Granados received the 2021 IEEE Signal Processing Society's Best Paper Award. He was a Member of the IEEE Signal Processing Society Sensor Array and Multichannel Technical Committee from 2019 to 2024. He has been a Member of the EURASIP Signal Processing for Multisensor Systems Technical Committee since 2022. He was the President of the Spanish Chapter of the IEEE Aerospace and Electronic Systems Society from 2018 to 2025.