Semiassisted Signal Authentication for Galileo: Proof of Concept and Results

IGNACIO FERNANDEZ-HERNANDEZ
European Commission, Brussels, Belgium

JÓN WINKEL
RHEA, Wavre, Belgium

CILLIAN O'DRISCOLL
COD Consulting, Cork, Ireland

SIMÓN CANCELA
GMV Aerospace and Defence, Madrid, Spain

RAFAEL TERRIS-GALLEGO, Member, IEEE
JOSE A. LÓPEZ-SALCEDO, Senior Member, IEEE
GONZALO SECO-GRANADOS, Fellow, IEEE
Universitat Autònoma de Barcelona, Barcelona, Spain

ANDREA DALLA CHIARA
CARLO SARTO
Qascom S.r.l., Bassano del Grappa, Italy

DANIEL BLONSKI
European Space Agency, Noordwijk, The Netherlands

JAVIER DE BLAS
European Union Agency for the Space Program, Prague, Czech Republic

Manuscript received 29 August 2022; revised 28 November 2022; accepted 26 January 2023. Date of publication 9 February 2023; date of current version 9 August 2023.

DOI. No. 10.1109/TAES.2023.3243587

Refereeing of this contribution was handled by J. Gross. *(Corresponding author: Ignacio Fernandez-Hernandez.)*

Authors' addresses: Ignacio Fernandez-Hernandez is with the European Commission, 1049 Brussels, Belgium, E-mail: (ignacio.fernandez-hernandez@ec.europa.eu); Jón Winkel is with RHEA, Munich, Germany, E-mail: (j.winkel@rheagroup.com); Cillian O'Driscoll is with COD Consulting, Cork, Ireland, E-mail: (cillian@codc.ie); Simón Cancela is with GMV Aerospace and Defence, 28760 Madrid, Spain, E-mail: (scancela@gmv.com); Rafael Terris-Gallego, Jose A. López-Salcedo, and Gonzalo Seco-Granados are with the Universitat Autònoma de Barcelona, 08193 Barcelona, Spain, E-mail: (rafael.terris@uab.cat; jose.salcedo@uab.cat; gonzalo.seco@uab.cat); Andrea Dalla Chiara and Carlo Sarto are with Qascom S.r.l., 36061 Bassano del Grappa, Italy, E-mail: (andrea.dallachiara@qascom.it; carlo.sarto@qascom.it); Daniel Blonski is with the European Space Agency, 2201 Noordwijk, The Netherlands, E-mail: (daniel.blonski@esa.int); Javier de Blas is with the European Union Agency for the Space Program, 170 00 Prague, Czech Republic, E-mail: (franciscojavier.deblasfernandez@euspa.europa.eu). *(Corresponding author: Ignacio Fernandez-Hernandez.)*

0018-9251 © 2023 IEEE

A semiassisted global navigation satellite systems (GNSS) authentication concept is proposed, with two main advantages. First, it can be implemented in a GNSS without modifications to the signal plan, provided that the GNSS can transmit an encrypted signal and an open signal with unpredictable and verifiable bits regularly. Second, it can work in a receiver for up to several days without any assistance or the possession of a private key, at the expense of some authentication latency. In this concept, parts of the encrypted signal to be transmitted in the future are re-encrypted, published, and downloaded in the receiver. Some seconds after transmission, the satellite openly broadcasts the decryption key, and the receiver performs the *a-posteriori* correlation with the encrypted signal. The concept is particularized for the Galileo system as Galileo assisted commercial authentication service, based on OSNMA in E1B for the decryption keys and E6C for the encrypted signal. An end-to-end procedure for measurement authentication is proposed and tested, showing its viability.

I. INTRODUCTION

Global navigation satellite systems (GNSS) location is based on two main inputs: the satellite positions and time data and the pseudorange measurements obtained at the user receiver. GNSS data authentication is already in the signal in space for Galileo through open service navigation message authentication (OSNMA) [1] and possibly for other GNSS in the future. Pseudoranges are more difficult to authenticate, as they are generated in the receiver. However, encrypting the spreading code sequence used to compute them provides an additional level of protection. The reason is that direct sequence – spread spectrum (DS–SS) techniques bury the signals below the noise floor in the receiver, so an adversary cannot easily estimate the encrypted chips. That is why spreading code total or partial encryption is proposed as the most effective way to authenticate GNSS signals and measurements [2]. Other methods can use signal unpredictability from NMA cryptographic bits or symbols [3], vestigial signal [4] or power-distortion monitoring [5], [6], or consistency checks with inertial sensors or other sources [7], [8]. While spreading code authentication may still be vulnerable to a delayed rebroadcast of the signal, or *meaconing,* its use in combination with the methods just mentioned, particularly [4], [5], [6], can make it highly robust against most possible attacks.

Civil spreading code authentication (SCA) was proposed almost two decades ago [9] and can be implemented in many ways, many of which are reviewed in [10]. In standalone mode, i.e., without any assistance, SCA is based on spreading code watermarks correlated *a-posteriori*, when a decrypting key is disclosed, as proposed in [11] and more recently in [12]. In assisted mode, the receiver can capture an encrypted signal snapshot and have it authenticated in a server, which may [13] or may not [14] know the decryption key. In another approach, the receiver receives from the server *a-posteriori* the correlation replica [15]. In both cases, the receiver needs continuous assistance from a server. Both standalone and assisted methods are under test for GPS Navigation Technology Satellite (NTS)-3's CHIMERA [16] and have also been proposed for Galileo [17], which is analyzing them for its second-generation signals.

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

Meanwhile, the proposed scheme, here named semiassisted authentication, is intended for Galileo first generation but can be extrapolated to other GNSS. In this scheme, the GNSS transmits unpredictable and authenticatable bits and an encrypted signal, e.g., a military or commercial one. The purpose is to facilitate system implementation and maximize receiver autonomy, as it will be explained later. Once Galileo confirmed the provision of signal authentication [18], this concept was proposed to alleviate receiver requirements yet providing SCA based on the already existing signal plan [19]. This concept is applied to Galileo through the E1B OSNMA and the E6C pilot signal, currently open [20] but soon to be encrypted.

After this introduction, we present the general concept. Then, we particularize it for Galileo, including cryptographic operations, definition of the E6C snapshot, signal processing, and signal authentication metrics. We then present test results and finalize with the conclusions.

II. GENERAL CONCEPT

As previously mentioned, our main motivation is to provide SCA by a GNSS system that transmits a spreading code-encrypted signal and NMA, such as Galileo, without changing the signal plan, satellite payload, or key management scheme. Currently, all GNSS transmit encrypted signals: GPS's military P(Y) and M-code signals, Galileo PRS (public regulated service) [21] and commercial service (CS) [22] E6C component, and so on. Galileo already provides NMA, but other GNSS may too do so in the future. Note that SCA designs from scratch, without these constraints, may be more optimal, such as [16] or [17]. If the signals cannot be upgraded but the key management scheme can, then [12] can be a good option.

The service operator must know *a priori* the key used for the encrypted signal spreading code, and the future NMA keys transmitted in an open signal. These NMA keys are unpredictable to the user. For example, a TESLA chain like OSNMA's [23] fits this purpose. With these two elements, the operator re-encrypts parts of the encrypted signal with the NMA cryptographic material to be later disclosed, to create the so-called *re-encrypted code sequences* (RECS) and publish them in advance.

The receiver downloads the RECSs, with their start and end times, for the desired autonomy period. At, or around, those times, the receiver records a signal snapshot that should contain the encrypted signal, then waits for the NMA key to be disclosed, decrypts the RECS to obtain the original encrypted code sequence (ECS), and correlates *a-posteriori* the signal snapshot with the ECS. If the presence of the ECS is detected, under certain conditions and hypotheses the receiver can assume, with a high probability, that the signal is authentic. A key assumption is that the receiver has a loose synchronization time source, sufficient to prevent an adversary to replay the ECS after the NMA key is disclosed. This requirement is already in place for Galileo OSNMA.

The concept is depicted in Fig. 1. Fig. 1(a) shows an encrypting key K_{encr} generating a keystream. The keystream

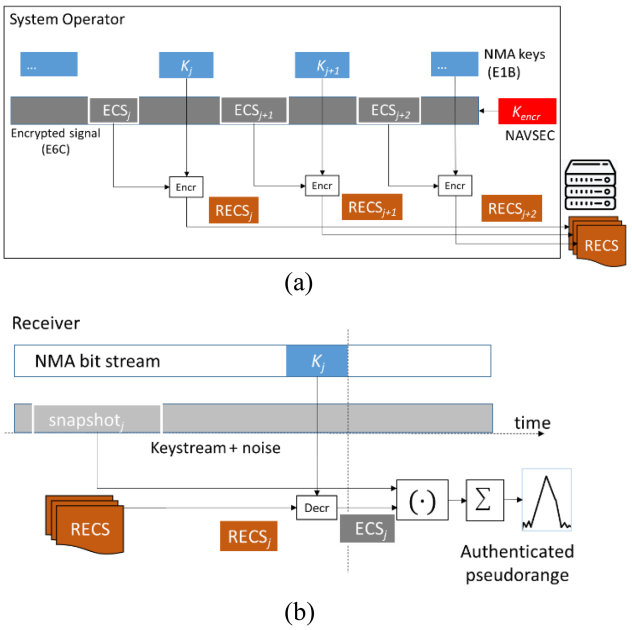


Fig. 1. Semiassisted spreading code authentication (a) system side and (b) receiver side.

will be multiplied with the E6-C spreading codes. Some sequences of the encrypted signal, denoted ECS_j, ECS_{j+1} …, are re-encrypted with NMA keys K_j, K_{j+1} …, forming RECS_j, RECS_{j+1} …; then these RECS, together with their associated time tags and other metadata, are published in a server, for a certain period that can be of several days or even weeks, depending on the system operation and key management. In Fig. 1(b), the receiver stores a signal snapshot containing the ECS, waits until the related NMA key (K_j) is disclosed, decrypts RECS_j with a hashed version of the key (K_j'), and performs the correlation, obtaining the de-spreading gain if the signal is found. We must highlight that this is a proposal that provides a confidence level on the pseudorange authentication but, in order to increase this confidence level, further receiver measures must be implemented. These may include power level checks such as automatic gain control (AGC) monitoring [24], dedicated signal correlation tests such as [25], or other checks [4], [5]. While relevant, these checks are left out of the scope of this work.

III. PARTICULARIZATION FOR GALILEO ACAS

The semiassisted authentication concept is currently under implementation and testing as Galileo Assisted Commercial Authentication Service, or ACAS. An early version was also prototyped in the EC NACSET project [26]. ACAS is based on two already existing Galileo signals: the Galileo E1B [27] including OSNMA [28], and the Galileo E6C pilot signal [20] providing the encrypted signal. This signal is currently open but will soon be encrypted. An implementation of the Galileo ACAS concept is depicted in Fig. 2, with the following steps:

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

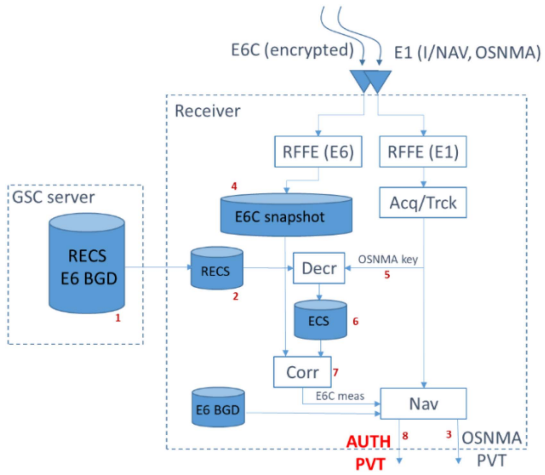


Fig. 2. Semiassisted CAS scheme, including the steps to deliver the service.

1) The server, e.g., the European GNSS Service Centre, publishes the E6C RECS files and authenticated satellite broadcast group delays (BGDs) between E1 and E6, as this information is not in the broadcast I/NAV message and therefore not authenticated by OSNMA. Note that Galileo HAS (high accuracy service) provides these BGDs but they are unauthenticated at the moment.
2) The receiver downloads the RECS/BGD from the server.
3) The receiver starts up, receives through the RFFE (radio frequency front end) and acquires the Galileo E1B signal, and obtains data-authenticated PVT.
4) At the time RECS are expected, the receiver records and stores a snapshot of E6C samples.
5) After some time, the RECS-related OSNMA key is received.
6) For each satellite, the RECS is decrypted with an OSNMA-based key, obtaining the ECS.
7) The ECS and the E6C snapshot are correlated, to compute an E6C pseudorange.
8) E6C pseudoranges together with I/NAV authenticated data and E6 BGDs are used for measurement and eventually position authentication.

A. RECS and BGD Files

The RECS file starts with a header that defines the start time (t_{start} as per Fig. 3), duration, satellite ID for which RECS are provided, initial offset between t_{start} and the first RECS (δ_{RECS}), time between RECS (τ_{RECS}), number of chips per RECS (N_{chips}), delay between the RECS I/NAV subframe (or OSNMA period) and the OSNMA key (D_K, where $D_K = 1$ means OSNMA key in the next I/NAV subframe), and $\Delta\tau_{\mathrm{MAX}}$, or RECS maximum random delay, which is a parameter to optionally randomize the position of each RECS. RECS parameters are depicted in Fig. 3, where $\mathrm{RECS}^{k}_{j,i}$ represents the ith RECS of satellite k to be decrypted using OSNMA key j. The top plot shows no time randomization ($\Delta\tau_{\mathrm{MAX}} = 0$), and the bottom plot shows

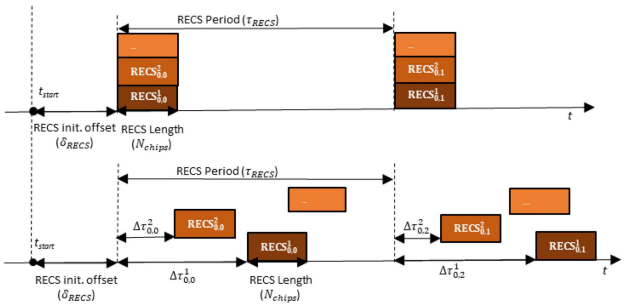


Fig. 3. Main parameters of RECS files, without (top) and with (bottom) randomization.

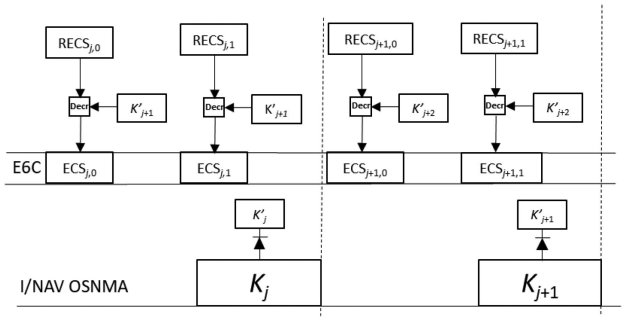


Fig. 4. RECS decryption process ($D_K = 1$ and two RECS per 30-s OSNMA period).

randomization in the RECS start ($\Delta\tau \leq \Delta\tau_{\mathrm{MAX}} \neq 0$). The receiver can determine $\Delta\tau$ only after the OSNMA key of the related period is disclosed.

Regarding the BGD files, the system can predict the E1 I/NAV-E6C BGDs and provide them in separate files. The BGDs are very stable and in nominal conditions can be predicted for several days.

B. Cryptographic Operations

The main cryptographic operations required at the receiver are the generation of the RECS decryption key, the generation of the randomization parameter, and the RECS decryption. Further details about cryptographic operations and parameters, including OSNMA, can be found in [28] and [29]. The RECS decryption process for several OSNMA blocks is depicted in Fig. 4.

Once the OSNMA key K_j belonging to OSNMA block j is received and verified by the OSNMA keychain, the RECS decryption key K'_j is generated as follows:

$$K'_j = \mathrm{SHA256}\left(K_j\right) \tag{1}$$

where SHA256() is the hash function SHA-256 as per [30]. As SHA256 is a one-way function, this allows RECS decryption only when the OSNMA-related key is disclosed, but, at the same time, the OSNMA keychain itself is not used at the moment of RECS encryption.

RECS time randomization avoids all RECS to be placed at the same time in the signal. It is bounded by the parameter $\Delta\tau_{\mathrm{MAX}}$, the maximum RECS random delay. If $\Delta\tau_{\mathrm{MAX}} \neq 0$, RECS locations are delayed by a random time offset $\Delta\tau$ between zero and $\Delta\tau_{\mathrm{MAX}}$. For example, if $\Delta\tau_{\mathrm{MAX}} = 3$,

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

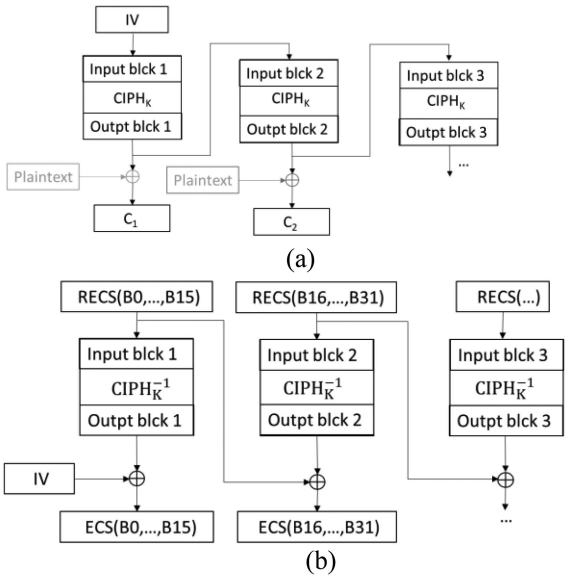


Fig. 5. Generation of C for (a) RECS randomization and (b) ECS. RECS/ECS are represented in 16-byte arrays. The plaintext in (a) is depicted in grey as it is set to zero.

RECS can be located with a delay which is a multiple of 8 ms (or two E1B/C codes), i.e., 0, 8, 16, or 24 ms for $\Delta\tau = 0, 1, 2$, or 3, respectively, as per (5). As there is only one OSNMA key for every 30-s period, but more offsets may be required for each satellite per RECS period, we first use the AES cypher, as shown in Fig. 5, to generate a sufficiently large cyphertext from which the random time offsets are extracted. In particular, the cyphertext is generated as follows:

$$(\mathrm{C}_1, .., \mathrm{C}_\mathrm{N}) = \mathrm{AES256}_{OFB}\left(K'_{j+D_K}, 0,\ \mathrm{IV}\right) \tag{2}$$

where $(\mathrm{C}_1, .., \mathrm{C}_\mathrm{N})$ is the cyphertext block array from which the $\Delta\tau$'s are generated, consisting of N 128-bit blocks; $\mathrm{AES256}_{OFB}(a, b, \mathrm{IV})$ is the AES cipher in OFB (output feedback) mode and configured for 256-bit keys (AES-256), where a is the key, b is the plaintext, and IV is the initialization vector (IV), as per [31]. The IV used for $\mathrm{AES256}_{OFB}$ is generated as follows:

$$\mathrm{IV} = trunc\left(128, \mathrm{SHA256}\left(GST_{SF,j}\right)\right) \tag{3}$$

where $trunc(n, p)$ is the truncation function that retains the n MSBs of the input p, and $GST_{SF,j}$ is the 32-bit GST associated to the OSNMA key K_j as per [27]. Then, the cyphertext block array is allocated to the random time offsets $\Delta\tau$s as follows:

$$\begin{aligned} C_1 &= \left[B_1^1, B_1^2, \ldots, B_1^{16}\right], C_2 = \left[B_1^{17}, B_1^{18}, \ldots, B_1^{32}\right], C_3 \\ &= \left[B_1^{33}, \ldots\right], \\ C_4 &= \left[B_2^1, B_2^2, \ldots, B_2^{16}\right], C_5 = \left[B_2^{17}, B_2^{18}, \ldots, B_2^{32}\right], C_6 \\ &= \left[B_2^{33}, \ldots\right], \\ &\ldots \end{aligned} \tag{4}$$

where B_i^k is a byte defining $\Delta\tau_i^k$, expressed as an integer number of 8-ms periods, as follows:

$$\Delta\tau_i^k = B_i^k \mod (\Delta\tau_{MAX} + 1) \tag{5}$$

where mod is the *modulo* operator; $\Delta\tau_{MAX}$ is described in the RECS file; i is the RECS position in 30-s period j (j omitted from (6) for simplicity); and k is the SVID index, defining the satellite, and where only values from 1 to 36 are currently used. For example, if $\Delta\tau_{MAX} = 3$ and $B_2^1 = 5$, then $\Delta\tau_2^1 = 1$ (8 ms), which relates to the second RECS of the period for SVID = 1. Fig. 5(a) shows how to generate C based on AES in OFB mode. The figure is based on the AES modes of operation as described in [31].

Finally, the ECS decryption process, particularized for the decryption key K'_{j+D_K}, is performed as follows, for every satellite k:

$$\mathrm{ECS}_{j,i}^k = \mathrm{AES256}_{CBC}^{-1}\left(K'_{j+D_K}, \mathrm{RECS}_{j,i}^k,\ \mathrm{IV}\right) \tag{6}$$

where $\mathrm{ECS}_{j,i}^k$ is the decrypted sequence and $\mathrm{AES256}_{CBC}^{-1}(a, b, \mathrm{IV})$ is the inverse cipher AES, configured for 256-bit keys (AES-256) in CBC mode to decrypt a plaintext b with 256-bit key a, where IV is defined in (3). As abovementioned, the j index defines the decrypting key, and the i index allows for more than one RECS per decrypting key. Note that $\mathrm{RECS}_{j,i}^k$ must be an integer number of blocks of 128 bits, as required by the proposed AES implementation [32], configured to operate in CBC mode. Fig. 5(b) shows how to generate the ECS based on AES in CBC mode. Note that the AES standard and operation modes as per [31] and [32] also define how the AES cypher can generate the RECS in the first place, as per Fig. 1(a).

C. Snapshot Interval Determination

This section and the following ones present an implementation of the ACAS signal correlation process. The ACAS signal authentication presented here is a two-step approach: first, the signal in E6 is detected. Then, upon successful detection, the range offset between E1 and E6 is estimated. The signal detection approach proposed is essentially a handover from the E1 tracking process, reducing the signal search in E6 to a handful of correlations. This is justified since an authentic signal must be present simultaneously in E1 and E6. The general correlation approach is discussed in [33] and the range authentication step is developed in the following.

E6C snapshot definition – We assume the E1 signal with OSNMA is trustable *a priori* and verify this hypothesis with the E6C ECS correlation. Therefore, at the time of the capturing the snapshot, the receiver is synchronized with GST through E1. Then, for a certain satellite k and ECS_j^k, associate with a GST second GST_j, the *a priori* E6 snapshot start time $t_{start,j}^k$ in the receiver is defined as follows:

$$t_{start,j}^k = \mathrm{GST}_j + \delta_{\mathrm{RECS}} + \tau_{prop}^k - \delta t_{sat,E1}^k + \delta t_{rx,E1} + \delta_{E1,E6}^k \tag{7}$$

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

where δ_{RECS} is the RECS offset, τ^k_{prop} is the estimated propagation time of the satellite (typically between 77 and 97 ms for Galileo), $\delta t^k_{sat,E1}$ is the satellite clock offset at the time of the snapshot in the E1 signal (which is known at the receiver), $\delta t_{rx,E1}$ is the receiver clock bias (which is also estimated by the receiver, based on an E1 signal solution), and $\delta^k_{E1,E6}$ is the estimated time bias between the E1 and E6 signals (detailed later). $\delta^k_{E1,E6}$ ensures that the ECS is fully contained in the snapshot. Note that having multiple ECS per OSNMA block (i.e., $i > 1$) has benefits, such as allowing to detect when a spoofing attack started. For now, we drop the index i for simplicity but without loss of generality; the proposal can be understood as for the case of $i = 1$, i.e., one ECS per OSNMA key, but it can be generalized to $i > 1$. Another possibility is to partition the ECS and spread them throughout the block, as proposed in [33]. Note also that our definition of $t^k_{start,j}$ neglects the errors in the estimations of its related parameters, which may be in the order of a few meters and therefore usually below one E6C chip (∼60 m). Then, the snapshot end time $t^k_{end,j}$ can be defined as follows:

$$t^k_{end,j} = t^k_{start,j} + \Delta\tau_{MAX} + \frac{N_{chips}}{R_c} \tag{8}$$

where $\Delta\tau_{MAX}$ is the maximum random offset, N_{chips} is the RECS length in chips, and R_c is the chip rate ($R_c = 5.115 \cdot 10^6$ chips per second for E6C). A snapshot that covers the time interval $t^k_{start,j} \le t < t^k_{end,j}$ usually covers all chips in the RECS for satellite k in epoch j. If the receiver captures only one snapshot for all satellites $k = 1 \ldots \kappa$, then the snapshot needs to be broadened by some tens of ms, as defined by $t^{all}_{start,j}$ and $t^{all}_{end,j}$:

$$t^{all}_{start,j} = \text{GST}_j + \delta_{\text{RECS}} + \min(\boldsymbol{\tau}^{all}_{sat,prop}) + \delta t_{rx,E1} \tag{9}$$

$$t^{all}_{end,j} = \text{GST}_j + \delta_{\text{RECS}} + \max\left(\boldsymbol{\tau}^{all}_{sat,prop}\right) + \delta t_{rx,E1} + \Delta\tau_{MAX} + \frac{N_{chips}}{R_c} \tag{10}$$

where max() and min() are the maximum and minimum operators and $\boldsymbol{\tau}^{all}_{sat,prop}$ is a vector with propagation times and signal time offsets

$$\boldsymbol{\tau}^{all}_{sat,prop} = (\tau^1_{prop} - \delta t^1_{sat,E1} + \delta^1_{E1,E6}, \ldots, \tau^\kappa_{prop} - \delta t^\kappa_{sat,E1} + \delta^\kappa_{E1,E6}) \tag{11}$$

This guarantees that the snapshot starts at the earliest-to-arrive satellite and finishes at the end of the ECS of the latest satellite, considering both propagation, satellite clocks and biases. When K'_{j+D_K} is determined, the random times Delta taukj are calculated, allowing the snapshot of samples for each satellite $\Delta\tau^k_j$ to be narrowed as follows:

$$t^k_{acq-start,j} = t^k_{start,j} + \Delta\tau^k_j \tag{12}$$

$$t^k_{acq-end,j} = t^k_{acq-start,j} + \frac{N_{chips}}{R_c}. \tag{13}$$

E1-E6 time and frequency offsets – The term $\delta^k_{E1,E6}$ models the time bias estimation between the E1 and E6 pseudoranges. It allows estimating the E6 pseudorange from the E1 measurement. It is modeled as follows:

$$\delta^k_{E1,E6} = \widehat{BGD}^k_{sat,E1,E6} + \delta I_{E1,E6} + \widehat{HWB}_{rx,E1,E6} \tag{14}$$

$\widehat{BGD}^k_{sat,E1,E6}$ is the *estimate* (denoted as $\widehat{(\cdot)}$) of the satellite bias, or BGD between E1B/C and E6C signals transmitted by satellite k. $\widehat{BGD}^k_{sat,E1,E6}$ is an input from the BGD file, as shown in Fig. 2. Then,

$$\delta t^k_{sat,E6} = \delta t^k_{sat,E1} - \widehat{BGD}^k_{sat,E1,E6} \tag{15}$$

where $\delta t^k_{sat,E6}$ is the satellite time offset as observed in the E6 signal. Galileo E1-E6 BGDs are usually of the order of up to a few meters and are expected to be stable, allowing the correlation process.

$\widehat{HWB}_{rx,E1,E6}$ is the receiver hardware bias between E1B/C and E6B/C. $\widehat{HWB}_{rx,E1,E6}$ should be calibrated in the receiver and estimated *a priori*. If the E6 sample-grabbing hardware is different than the E1B/C tracking one, hardware biases may be higher than those in standard multifrequency receivers, as shown in [34].

Finally, $\delta I_{E1,E6}$ is an offset due to the frequency dispersive effect of the ionosphere [35]. The total ionospheric error in E1, before correction, can be defined as follows:

$$I_{E1} \approx \frac{40.3\text{TEC}}{f_1^2} = \hat{I}_{E1} + \epsilon_{I,E1} \tag{16}$$

where TEC is the total electron content, $\hat{I}_{E1}$ is the corrected part from the ionospheric model, and $\epsilon_{I,E1}$ is the noncorrected part. We assume that the ionospheric error is estimated from a given ionospheric model, such as Galileo's NequickG [36] or GPS's Klobuchar [37], which corrects a certain percentage of the ionospheric error. For example, the Klobuchar model is claimed to correct about 50% of the ionospheric error in average ionosphere conditions and NequickG about 70% (RMS, root mean square). NequickG is preferred as its coefficients are authenticated by Galileo OSNMA. Then, the term $\delta I_{E1,E6}$ needs to account for the fact that, due to the frequency dispersion in the ionosphere, the error in E6 is higher,

$$I_{E6} \approx \frac{40.3\text{TEC}}{f_6^2},$$

so $\hat{I}_{E6}$ must be increased accordingly. In the absence of a more refined model, a user can estimate $\delta I_{E1,E6}$ as follows, where $f_1 = 1575.42$ MHz and $f_6 = 1278.75$ MHz, for E1 and E6, respectively:

$$\hat{I}_{E6} = \hat{I}_{E1} + \delta I_{E1,E6} = \hat{I}_{E1} \cdot \frac{f_1^2}{f_6^2} \tag{17}$$

$$\delta I_{E1,E6} = \hat{I}_{E1}\left(\frac{f_1^2}{f_6^2} - 1\right). \tag{18}$$

D. ECS Correlation and Pseudorange Computation

For the correlation process, the receiver needs to align the ECS replica in time and frequency. The carrier frequency of the signal as processed by the receiver is mainly affected by the relative satellite-receiver velocity (Doppler effect) and the receiver clock oscillator drift. These effects add a

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

different frequency offset in the E1 and E6 bands, as the frequency carriers have different wavelengths. The frequency offset for E6 is estimated as follows:

$$\widehat{\Delta f}^{k}_{j,E6} = \Delta f^{k}_{j,E1} \cdot \frac{\lambda_1}{\lambda_6} \quad \Delta f^{k}_{j,E1} \cdot \frac{f_6}{f_1} \tag{19}$$

where $\Delta f^{k}_{j,E1}$ is the frequency offset in E1, and λ_1 and λ_6 are the carrier wavelengths for E1 and E6, respectively. The frequency offset can be also expressed for convenience as $\widehat{\Delta f}^{k}_{j,E6} = \Delta f^{k}_{j,E1} + \Delta f^{k}_{E1,E6}$, where $\Delta f^{k}_{E1,E6} = \Delta f^{k}_{E1}(1 - \frac{\lambda_1}{\lambda_6})$.

Regarding the E6 time offset, the receiver can use as an input the code phase and frequency estimations from E1, $\tau^{k}_{j,E1}$ and $f^{k}_{j,E1}$, respectively, for a given satellite k and instant identified by j and, where $\tau^{k}_{j,E1}$can be defined as the start of the 4-ms E1B spreading code, which is synchronized with the start of the ECS sequence in E6C, $\hat{\tau}^{k}_{j,E6}$, except for $\delta^{k}_{E1,E6}$:

$$\hat{\tau}^{k}_{j,E6} = \tau^{k}_{j,E1} + \delta^{k}_{E1,E6} \tag{20}$$

with $\hat{\tau}^{k}_{j,E6}$ and $\widehat{\Delta f}^{k}_{j,E6}$ the receiver can acquire the ECS in snapshot mode by correlating the ECS replica modulated with $\widehat{\Delta f}^{k}_{j,E6}$ with the snapshot. If the E6C encrypted signal is present in the snapshot, the receiver pseudorange and Doppler frequency measurements in E6C, synchronized with E1B except for noise and biases, can be verified for the position authentication.

Note that, while the frequency errors may be negligible assuming the same clock reference, in the time domain, several correlations may be needed due to ionosphere and multipath errors, as shown in the next section. Another practical aspect for consideration is that the receiver may not generate a pseudorange measurement exactly synchronized with $\tau^{k}_{j,E1}$, for example, E1 if measurements are generated at a 50-Hz rate and the E1 measurement does not coincide with the ECS. In this case, the receiver can estimate $\tau^{k}_{j,E1}$ based on the closest measurement and the known range rate. For periodic codes, cross-correlations can become a problem when integrating coherently over multiple periods (e.g., GPS L1 C/A code). As the encrypted sequences are essentially infinitely long random codes, their cross-correlation properties can be modeled as a Gaussian distribution of variance N_{chips} and, thus, may occasionally have worse cross-correlation properties than periodic ones. However, since in ACAS there is no accumulation over multiple consecutive periods, the cross-correlation can be shown to be negligible.

E. Measurement and Position Authentication

There are several ways by which the authentication verification can be performed. Aspects like the E6 versus E1 processing gain, or the difference in E6 and E1-based positions, can be assessed. For simplicity, here we propose a model based on the difference between $\tau^{k}_{j,E6}$ and $\hat{\tau}^{k}_{j,E6}$ compared to an upper bound of the estimated measurement error, as already outlined in [38]. For a given ECS^{k}_{j} associated to measurement $\tau^{k}_{j,E6}$, we define a boolean variable ξ^{k}_{j} and perform the following verification:

$$\left|\tau^{k}_{j,E6} - \hat{\tau}^{k}_{j,E6}\right| \le \gamma_{auth} \rightarrow \xi^{k}_{j} = 1 \text{ (meas. auth.)}$$

$$\left|\tau^{k}_{j,E6} - \hat{\tau}^{k}_{j,E6}\right| > \gamma_{auth} \rightarrow \xi^{k}_{j} = 0 \text{ (meas. not auth.)} \tag{21}$$

$$\prod_{k=1}^{\kappa} \xi^{k}_{j} = 1 \rightarrow E1\ I/NAV_j \text{ position authenticated.} \tag{22}$$

We define the authentication threshold γ_{auth} with the assumption that, in the absence of attacks, i.e., in nominal conditions, the error contributions follow (or can be bounded by) zero-mean normal distributions, and therefore the overall error of the test variable also follow (or can be bounded by) a zero-mean normal distribution characterized by a standard deviation σ_{auth}. Then,

$$\gamma_{auth} = \kappa\sigma_{auth} \tag{23}$$

where the value of κ can be defined for the desired level of confidence (e.g., $\kappa = 2$ for a 95% or $\kappa = 3$ for a 99.7% confidence interval, respectively), on a per-user basis. Integrity faults will fall under the "measurement not authenticated" ["*meas. not auth.*" in (21)] case, and discernment between integrity faults and spoofing is left out of scope of this work. In order to define σ_{auth} we need to model the *a priori* error contributions of all components of $|\tau^{k}_{j,E6} - \hat{\tau}^{k}_{j,E6}|$, which can be expressed as a sum of independent error contributions that we here model as Gaussian, zero-mean, and independent. In particular, multipath in E1 and E6 have different envelopes and are modeled independently, although they may be dependent, and also nonzero mean for short durations of time. The selected variance should be large enough to account for this, as well as the environment. With these assumptions, $|\tau^{k}_{j,E6} - \hat{\tau}^{k}_{j,E6}|$ can be modeled as a zero-mean Gaussian distribution with variance σ^2_{auth}, $N(0,\ \sigma^2_{auth})$, where

$$\sigma^2_{auth} = \sigma^2_{HWB_{rx,E1,E6}} + \sigma^2_{BGD^{k}_{E1,E6}} + \sigma^2_{I,E1}\left(\frac{f_1^2}{f_6^2} - 1\right) + \sigma^2_{MP,E6} + \sigma^2_{MP,E1} + \sigma^2_{n,E6} + \sigma^2_{n,E1}\,. \tag{24}$$

The Appendix provides a proof of (24), under the zero-mean Gaussian assumption. A proposal for measurement error variance characterization is also presented in [38].

IV. SIMULATION RESULTS

This section presents some simulation results of the signal authentication concept described in the previous sections. They complement other results focusing on other ACAS signal processing aspects in other references [39], [40]. The simulation covers both the signal detection as well as the range estimation part.

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

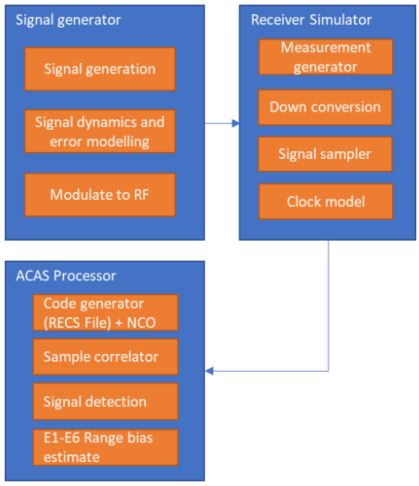


Fig. 6. ACAS simulator architecture.

A. Simulator Description and Configuration

A C++/MATLAB simulator has been developed to test ACAS. It assumes that the RFFE subsystems for E1 and E6 are driven by the same clock and that the sampling instants between frequencies are also synchronized. In a real ACAS receiver, the RFFE subsystems must be calibratable, in the sense that any digitally produced delays (filters, decimation, etc.) can be modeled in parameter $\widehat{HWB}_{rx,E1,E6}$. The residual errors between the two RF paths in the receiver ($\sigma^2_{HWB_{rx,E1,E6}}$) as well as the group delays in the satellites ($\sigma^2_{BGD^k_{E1,E6}}$) are assumed to be negligible after application of the BGDs.

The ACAS simulator is depicted in Fig. 6 and consists of three main modules: the signal generator, the receiver simulation, and the ACAS processor. The signal generator produces a signal that is consistent with the line-of-sight (LoS) dynamics and other error sources of the simulation. Based on the LoS and other relevant range errors, the receiver simulator generates E1B observables (pseudoranges and carrier phases) together with E6C sample batches. The errors modeled for the sample batches are, first, thermal noise as an additive white Gaussian noise (AWGN) added to each sample ($\sigma^2_{n,E6}$), consistent with the carrier-to-noise density ratio (C/N_0), which is an input to the simulation; and second, the relative offset between E1 and E6, based on the error of the E1 tracking loop ($\sigma^2_{n,E1}$) and the differential errors between E1 and E6 (mainly the delta ionosphere, based on $\sigma^2_{I,E1}$, and multipath, $\sigma^2_{MP,E6}$ and $\sigma^2_{MP,E1}$). The receiver time is simulated using a two-state clock model depending on h_o and h_{-2} with values $7.115{\cdot}10^{-24}$ s and $4.311{\cdot}10^{-21}$ Hz, respectively [41]. Finally, the ACAS processor uses the E1 measurements to directly process the signal on E6 and perform a coherent correlation over the sample batch and the ECS.

The E6 signal is detected by identifying the correlator value with the highest magnitude and, if above the threshold, which is defined *a priori* for a certain false alarm probability, signal detection is declared. The neighboring correlators (early and late) are used to form a linear interpolation of the *s-curve*, the root of which yields the range bias between E1 and E6 (snapshot mode), which relates to $|\tau^k_{j,E6} - \hat{\tau}^k_{j,E6}|$ as per (20) and (21). Since the receiver knows with good accuracy where to expect the E6 signal, both in delay and frequency, from the E1 measurements, the detection is done with just a few correlators. For the measurement authentication detection threshold γ_{auth}, in our proof-of-concept we have taken some simple assumptions which are presented and justified in Table I. These are just approximations to validate the method simulation results versus the theory. In an ACAS receiver, the threshold should ideally be set using all information available, such as current receiver configuration (bandwidth, tracking loop parameters), C/N_0, and other metrics depending on the receiver capability.

Out of the many simulations performed, we select the results of two runs, *Run 1* and *Run 2*, configured as per Table II, and with a simulation duration of 20 000 s, a sample batch rate of 1 Hz, and a complex sample rate 10 MHz. $P_{FA,\,signal}$ is the false alarm E6C signal acquisition probability, and $P_{MD,\,signal}$ is the theoretically expected misdetection probability associated to $P_{FA,\,signal}$ for the test.

TABLE I
Measurement Error Contributions

Parameter	Value	Justification
$\sigma_{HWB_{rx,E1,E6}}$	0 m	Assumed to be negligible thanks to receiver calibration.
$\sigma_{BGD_{E1,E6}}$	0 m	Assumed to be negligible thanks to BGD file
$\sigma_{I,E1}$	3.5 m	Preliminary estimation based on maximum RMS daily error after NequickG correction during 2013–2014 as per [36]. Also used in [38]
$\sigma_{MP,E1}$	3 m	Based on [38] and [42] as a preliminary estimation. C/N_0, elevation, dynamics, or environment not considered yet
$\sigma_{MP,E6}$	0.6 m	Preliminary approximation, based on $\sigma_{MP,E1}$ taking into account E1 and E6 chip rates
$\sigma_{n,E1}$	0.4 m	Based on [43], using C/N_0=37 dBHz, RF bandwidth 20 MHz, correlator spacing 0.1 chip, and DLL loop bandwidth of 1 Hz
$\sigma_{n,E6}$	2.5 m	E6 snapshot correlation does not benefit from filtering by a DLL in tracking mode and is therefore noisier. It corresponds to a DLL with a loop bandwidth of 125 Hz and 8 ms integration time
σ_{auth}	4.72 m	As per (24)

TABLE II
Simulation Parameters

	Run 1	Run 2
C/N_0 *(E1,E6)*	36 dBHz	37 dBHz
ECS	8 ms	8 ms
$P_{FA,signal}$	10^{-4}	10^{-5}
$P_{MD,signal}$	$8{\cdot}10^{-5}$	$2{\cdot}10^{-5}$
κ	3	4
γ_{auth}	14.16 m	18.88m
γ_{auth} conf. interval	99.7%	99.994%

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

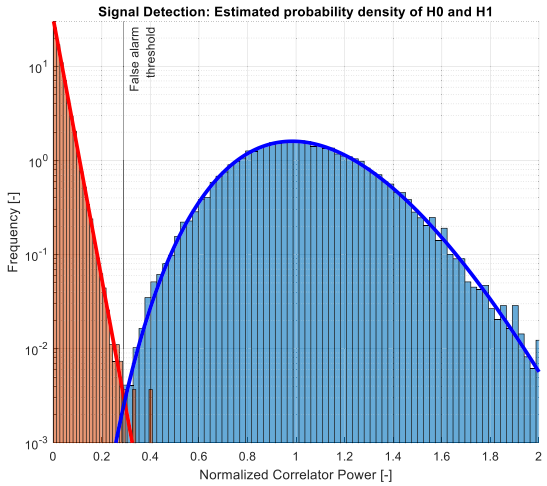


Fig. 7. Histograms for E6C signal detection (Run 1). Left: H0, E6 signal not present. Right: H1: E6C signal present. The theoretical expected curves are shown as solid lines.

TABLE III
E6C Signal Detection Simulation (Sim.) Results Versus Theory

	E6C signal not present (spoofing)		E6C signal present	
Signal detection	Signal detected	Signal not detected	Signal not detected	Signal detected
Spoofing logic	Spoofing misdetection	Spoofing detected	Spoofing false alarm	Nominal case
	$P_{FA,signal}$	1-$P_{FA,signal}$	$P_{MD,signal}$	1-$P_{MD,signal}$
Run 1 Theory	0.01%	99.99%	0.008%	99.992%
Run 1 Sim.	0.01%	99.99%	0.01%	99.99%
Run 2 Theory	0.001%	99.999%	0.002%	99.998%
Run 2 Sim.	0%	100%	0.005%	99.995%

B. Test Results

Signal detection – Histograms and numerical results from the simulations of the two hypotheses (signal present and absent) are presented in Fig. 7 and Table III, showing a good match between theory and results. We assume that E6C absence means that the receiver is spoofed (i.e., system events leading to E6C correlation absence are excluded). Then, a *signal false alarm* (i.e., the signal is not present, but it is detected), means a *spoofing misdetection*. Likewise, a *signal misdetection* (i.e., signal present but not detected) means a potential *spoofing false alarm.* This logic is clarified in Table III. The attempt of signal detection in the absence of the signal is done by correlating against an absent code. Note that an ACAS receiver can monitor this metric and the corresponding threshold. Also note that, as this test is per LoS, it may be acceptable to temporarily remove a LoS from the E1 PVT solution from an E6 misdetected signal, until it can be verified, in order to reduce spoofing false alarms. Also, spoofing alarms could be raised following a "tiered" approach, whereby an alarm is raised following two consecutive checks where the E6C is not detected.

Range estimate and measurement authentication – After the signal has been detected, the range bias between the E6 measurement $\tau^k_{j,E6}$ and the E1-based estimate $\hat{\tau}^k_{j,E6}$ is calculated. The simulator runs 21 correlators in parallel, separated by 3 m (i.e., 1/10 of an E6C chip) to correlate

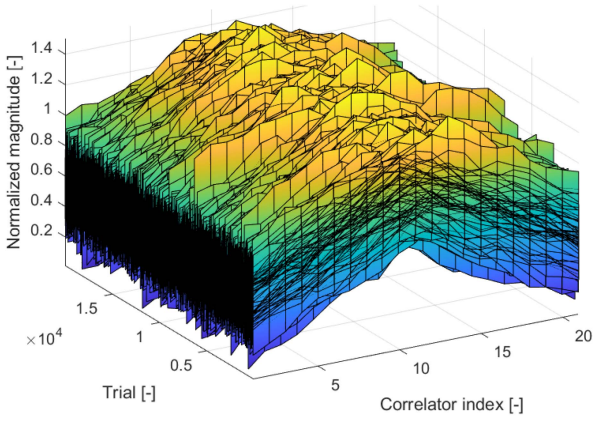


Fig. 8. "Waterfall" of 20 000 batches.

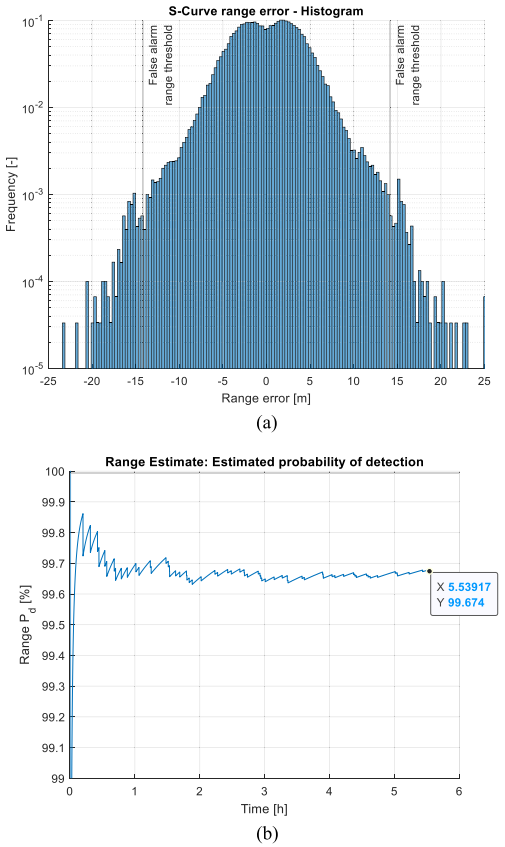


Fig. 9. Range estimates from Run 1. (a) Histogram of the range error. The false alarm threshold γ_{auth} is defined for K = 3. (b) Probability of detection.

the ECS against the snapshot samples. Due to the accurate frequency estimate from the observables on E1, no frequency search is needed. Fig. 8 shows the result of a "waterfall" of all 20 000 batches for Run 1. The results show how the correlator peaks are affected by the different error contributions.

The measured versus estimated E6 ranges ($\tau^k_{j,E6} - \hat{\tau}^k_{j,E6}$) also for Run 1 are shown in Fig. 9(a). The values generally follow a normal distribution except for some events lumped into several distinct peaks. This mismatch, which may affect the slight difference between probabilities abovementioned, may be due to the simple S-curve root estimation explained

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
RECS Storage Needs for Different Use Cases

Number of satellites	24	10	24
RECS period [s]	30	30	120
Hours of autonomy	12	1	120
Correlation time [ms]	1	8	8
Total storage [Mbytes]	**22.18**	**6.25**	**450.14**

above: in some cases, the correlator with the maximum value is not the most accurate one and the root-finding is biased. Also, note that the E6 snapshot range estimator only involves a few operations and in particular no iterations. Of course, there exist more elaborate methods, but we wanted to show that ACAS processing is possible and gives meaningful results with very low complexity. In Fig. 9(b) the result of the verification in (21) is shown for Run 1. The curve converges to approximately a probability of measurement authentication of 99.67%, where the expected value is 99.73% ($\kappa = 3$). Note that the run takes some observations to converge to the theoretical probability of detection, but this probability is constant along the run.

C. Further Considerations

The ACAS measurement authentication check can be complemented with power measures (e.g., Automatic Gain Control, AGC) or vestigial ECS during the uncertainty time in the receiver, as further elaborated in [39]. Also, an end-to-end logic when the measurement is not authenticated needs to be developed. Depending on other factors, this may be considered as an indicator of signal spoofing, or just degraded conditions. In any case, when the measurements are authenticated, which hopefully will be most of the time, and under the assumptions stated, this can be used to ascertain the authenticity of the receiver position, which is the final goal of GNSS authentication. Also, further error characterization with real measurements is advisable to tune the signal detection and measurement authentication thresholds to optimal values.

Another open point is the receiver protection against attacks during the 30-s period between authentications. A receiver may extrapolate the authenticated solution by using dead reckoning from inertial sensors, or clock/user dynamic models, e.g., as proposed in [44]. These topics will be further researched in follow-up of this work.

Finally, the receiver data storage requirement for a certain autonomy period is also relevant. This requirement depends on the RECS length, the number of satellites stored, and the RECS period. For example, a receiver requiring a 1-h autonomy with an authenticated position fix (RECS period) every 30 s, RECS for 10 satellites, and 8 ms per correlation, would require only 6.25 Mbytes of space, as shown in Table IV, which also includes other examples. Note that, for short periods, the receiver can only download RECS for a subset of satellites. In any case, the data storage requirements for semiassisted authentication seem affordable for current devices in the market.

V. CONCLUSION

Semiassisted signal authentication can be implemented in a GNSS without modifications to the signal plan, provided that the GNSS transmits an encrypted signal and an open signal with unpredictable and verifiable bits. It can also work in a receiver with several days of autonomy and without the possession of a private key, at the expense of some authentication latency. The concept is based on taking parts of an encrypted signal, or ECS to be transmitted in the future, re-encrypting them (into RECS) and publishing them for download in the receiver. The receiver stores a snapshot with the ECS and, some seconds later, receives the key allowing to decrypt the RECS and perform the *a-posteriori* ECS correlation.

Semiassisted authentication is particularized as Galileo ACAS, based on OSNMA in E1B for the decryption keys and E6C for the encrypted signal. We describe ACAS in detail, including the RECS input files and the receiver cryptographic operations. We propose a measurement authentication process based on the comparison of E1 and E6C measurements from the ECS correlation with a threshold based on the *a priori* variances of the error contributions, although more refined methods are possible.

The ACAS concept is tested in the simulator with very low demands on resources: only a handful of correlations in E6C are necessary due to the accurate handover from the E1 observables. C/N_0 values of 36–37 dBHz, ECS sizes of 8 ms, and measurement error models from the literature are used. The results are consistent with the theory, with a spoofing misdetection probability below 10^{-4} and a measurement authentication probability around 99.7%, although these values are configurable. The receiver storage requirements for the RECS are also outlined, showing ACAS viability with today's receivers.

Further work will include developing the end-to-end ACAS receiver logic and testing the method with real measurements.

APPENDIX

The standard deviation of σ_{auth} is the statistical distribution of $\tau^k_{E6} - \hat{\tau}^k_{E6}$. It is modeled as follows for $f_1 = 1575.42$ MHz and $f_6 = 1278.75$ MHz. Note that if generalized to any frequencies f_a and f_b, the frequency coefficient $(\frac{f_a^2}{f_b^2} - 1)$ must be expressed as an absolute value $|\frac{f_a^2}{f_b^2} - 1|$:

$$\begin{aligned}\tau^k_{E6} - \hat{\tau}^k_{E6} &= \tau^k_{E6} - \tau^k_{E1} - \delta^k_{E1,E6} = HWB_{rx,E1,E6} \\ &- \widehat{HWB}_{rx,E1,E6} + BGD^k_{sat,E1,E6} - \widehat{BGD}^k_{sat,E1,E6} \\ &+ \epsilon_{I,E1}\left(\frac{f_1^2}{f_6^2} - 1\right) \\ &+ \epsilon_{MP,E6} - \epsilon_{MP,E1} + \epsilon_{n,E6} - \epsilon_{n,E1} = \epsilon_{HWB_{rx,E1,E6}} + \epsilon_{BGD^k_{E1,E6}} \\ &+ \epsilon_{I,E1}\left(\frac{f_1^2}{f_6^2} - 1\right) + \epsilon_{MP,E6} - \epsilon_{MP,E1} + \epsilon_{n,E6} - \epsilon_{n,E1}.\end{aligned}$$

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

ACKNOWLEDGMENT

The authors would like to acknowledge the ESA/EUSPA/JRC teams involved in the internal review of the concept, the NACSET and PAULA teams involved in its early prototyping, and V. Rijmen, G. Caparra, and F. Ardizzon for the work and discussions on the topic. The content of this article does not necessarily reflect the official position of the authors' organizations. Responsibility for the information and views set out in this article lies entirely with the authors. Future ACAS publications, reference algorithms, standards, or user scenarios may differ from the algorithms or techniques proposed in this work.

REFERENCES

[1] EUSPA, 2021. [Online]. Available: https://www.euspa.europa.eu/newsroom/news/tests-galileo-osnma-underway

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

[3] 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.

[4] C. Hegarty, B. O'Hanlon, A. Odeh, K. Shallberg, and J. Flake, "Spoofing detection in GNSS receivers through cross-ambiguity function monitoring," in *Proc. 32nd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2019, pp. 920–942.

[5] 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.

[6] J. N. Gross, C. Kilic, and T. E. Humphreys, "Maximum-likelihood power-distortion monitoring for GNSS-signal authentication," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 1, pp. 469–475, Feb. 2019.

[7] F. P. Rothmaier, "Statistical inference for safe and continuous navigation in the presence of GNSS spoofing," Ph.D. dissertation, Stanford Univ., Stanford, CA, USA, 2021.

[8] Ç. Tanıl, 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.

[9] L. Scott, "Anti-spoofing & authenticated signal architectures for civil navigation systems," in *Proc. 16th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2003, pp. 1543–1552.

[10] D. Margaria, B. Motella, M. Anghileri, J.-J. Floch, I. Fernandez-Hernandez, and M. Paonni, "Signal structure-based authentication for civil GNSSs: Recent solutions and perspectives," *IEEE Signal Process. Mag.*, vol. 34, no. 5, pp. 27–37, Sep. 2017.

[11] O. Pozzobon, "Keeping the spoofs out - signal authentication services for future GNSS," in *Proc. InsideGNSS*, 2011, pp. 48–55.

[12] J. Anderson, S. Lo, and T. Walter, "Cryptographic ranging authentication with TESLA, rapid re-keying, and a PRF," in *Proc. Int. Tech. Meeting Inst. Navigation*, 2022, pp. 43–55.

[13] O. Pozzobon et al., "Open GNSS signal authentication based on the Galileo commercial service (CS)," in *Proc. 26th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2013, pp. 2759–2768.

[14] S. Lo, D. D. Lorenzo, P. Enge, D. Akos, and P. Bradley, "Signal authentication: A secure civil GNSS for today," *InsideGNSS*, vol. 4, no. 5, pp. 30–39, 2009.

[15] O. Pozzobon, L. Canzian, M. Danieletto, and A. Dalla Chiara, "Anti-spoofing and open GNSS signal authentication with signal authentication sequences," in *Proc. 5th ESA Workshop Satell. Navigation Technol. Eur. Workshop GNSS Signals Signal Process.*, 2010, pp. 1–6.

[16] J. M. Anderson et al., "Chips-message robust authentication (chimera) for GPS civilian signals," in *Proc. 30th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2017, pp. 2388–2416.

[17] D. S. Maier and T. Pany, "Assessment of a low power offset BPSK component for spreading code authentication," *J. Positioning, Navigation Timing*, vol. 9, no. 2, pp. 43–50, 2020, doi: 10.11003/JPNT.2020.9.2.43.

[18] European Commission, *Commission Implementing Decision (EU) 2017/224 of 8 February 2017 (CS Implementing Act)*, 2017.

[19] I. Fernandez-Hernandez, "Method and system for radionavigation authentication," Patent EP3491420B1, Jan. 11, 2017.

[20] European Union, "E6-B/C signal-in-space technical note, issue 1, January 2019," 2019. [Online]. Available: https://www.gsc-europa.eu/sites/default/files/sites/all/files/E6BC_SIS_Technical_Note.pdf

[21] EUSPA, "Public regulated service," Accessed: 2022. [Online]. Available: https://www.gsc-europa.eu/galileo/services/public-regulated-service

[22] I. Fernandez-Hernandez et al., "Testing GNSS high accuracy and authentication - Galileo's commercial service," *Inside GNSS*, vol. 11, no. 1, pp. 37–48, Jan./Feb. 2015.

[23] I. Fernandez-Hernandez, V. Rijmen, G. Seco-Granados, J. Simon, I. Rodriguez, and J. D. Calle, "A navigation message authentication proposal for the Galileo open service," *J. Inst. Navigation*, vol. 63, pp. 85–102, 2016.

[24] D. M. Akos, "GPS/GNSS spoofing detection via automatic gain control (AGC)," *Navigation*, vol. 59, no. 4, pp. 281–290, 2012.

[25] C. O'Driscoll, T. Scuccato, A. DallaChiara, T. Pany, M. A. Diez, and M. S. Hameed, "The attack agnostic defence: A spoofing detection metric for secure spreading sequences," in *Proc. NAVITEC*, 2022, p. 64.

[26] S. Cancela et al., "Field testing of GNSS user protection techniques," in *Proc. 32nd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2019, pp. 1824–1840.

[27] European Union, "OSSISICD: Open service signal-in-space interface control document, issue 2.0," 2021.

[28] European Union, "Galileo open service navigation message authentication (OSNMA) user ICD for the test phase - issue 1.0," Nov. 2021.

[29] 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.

[30] National Institute of Standards and Technology, "FIPS PUB 180-4: Secure hash standard (SHS)," 2012.

[31] National Institute of Standards and Technology, "NIST special publication 800-38A - recommendation for block cipher modes of operation," NIST, 2001.

[32] National Institute of Standards and Technology, "Federal information processing standards publication 197 (FIPS-197) specification for the advanced encryption standard (AES)," NIST, 2001.

[33] R. X. Kor, P. A. Iannucci, L. Narula, and T. E. Humphreys, "A proposal for securing terrestrial radio-navigation systems," in *Proc. 33rd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2020, pp. 3313–3325.

[34] C. O'Driscoll and G. Caparra, "Nautilus: An embedded navigation authentication testbed," in *Proc. 34th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2021, pp. 3698–3710.

[35] M. M. Hoque and N. Jakowski, "Ionospheric propagation effects on GNSS signals and new correction approaches," in *Proc. Glob. Navigation Satell. Syst.: Signal, Theory, Appl.*, 2012, vol. 10, Paper 30090.

[36] European Union, "Ionospheric correction algorithm for Galileo single frequency users v1.2," 2016. [Online]. Available: https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_Ionospheric_Model.pdf

[37] J. A. Klobuchar, "Ionospheric time-delay algorithm for single-frequency GPS users," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-23, no. 3, pp. 325–331, May 1987.

[38] F. Ardizzon, G. Caparra, I. Fernandez-Hernandez, and C. O'Driscoll, "A blueprint for multi-frequency and multi-constellation PVT assurance," in *Proc. NAVITEC*, 2022, p. 101.

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

[39] R. Terris-Gallego, I. Fernandez-Hernandez, J. A. Lopez-Salcedo, and G. Seco-Granados, "Guidelines for Galileo assisted commercial authentication service implementation," in *Proc. Int. Conf. Localization GNSS*, 2022, pp. 1–7.
[40] R. Terris-Gallego, I. Fernandez-Hernandez, J. A. López-Salcedo, and G. Seco-Granados, "Operating modes and performance evaluation of Galileo assisted commercial authentication service," in *Proc. 35th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2022, pp. 3393–3407.
[41] J. McNeill, S. Razavi, K. Vedula, and D. Richard Brown, "Experimental characterization and modeling of low-cost oscillators for improved carrier phase synchronization," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf.*, 2017, pp. 1–6.
[42] A. Simsky, D. Mertens, J. M. Sleewaegen, M. Hollreiser, and M. Crisci, "Experimental results for the multipath performance of Galileo signals transmitted by GIOVE-A satellite," *Int. J. Navigation Observ.*, vol. 2008, 2008, Art. no. 416380, doi: 10.1155/2008/416380.
[43] J. W. Betz, "Design and performance of code tracking for the GPS M-code," in *Proc. Int. Tech. Meeting Satell. Division Inst. Navigation*, 2000, pp. 2140–2150.
[44] T. Mina, A. Kanhere, A. Shetty, and G. Gao, "GPS spoofing-resilient filtering with chimera and self-contained odometry," in *Proc. 35th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2022, pp. 3768–3782.

Ignacio Fernandez-Hernandez received the Electronic Engineering degree in electronic engineering from ICAI, Madrid, Spain, in 2001, the M.B.A. degree in business administration from LBS, London, U.K., in 2011, and the Ph.D. degree in electronic systems from Aalborg University, Aalborg, Denmark, in 2015, for his research on snapshot positioning and GNSS authentication.

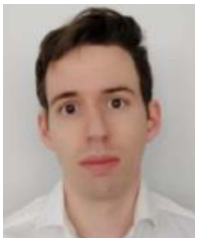
He works for the European Commission DG DEFIS, where he has led the definition of authentication and high-accuracy services for Galileo. In 2017, he was a Visiting Scholar with the GPS Laboratory, Department of Aeronautics and Astronautics, Stanford University, and is currently a Visiting Professor with KU Leuven, Leuven, Belgium.

Jón Winkel received the Diploma in physics from the University in Hamburg, Hamburg, Germany, in 1994 and the Ph.D. degree in geodesy from the University FAF Munich, Munich, Germany, in 2003.

He has more than 25 years of experience in GNSS, with more than 80 publications in journals and conferences. After 17 years, as the Head of the receiver department with IFEN, he joined the Vehicle Motion and Positioning Sensor (VMPS) team with the Chassis Systems Control Department, Bosch GmbH, in 2018 as an Expert on GNSS. In 2019, he was on the payload team developing the G2G Galileo satellites with Airbus. Since 2022, he has been an Advisor to the EC on GNSS, focusing on authentication services.

Cillian O'Driscoll received the Ph.D. degree in electrical and electronic engineering from University College Cork, Cork, Ireland, in 2006.

Since 2001, he has been active in satellite navigation research and development, with more than 70 publications in international conferences and journals. He previously worked as a Research Engineer with the University of Calgary, Canada, and with the European Commission contributing to the EU satellite navigation system Galileo. He is currently an independent consultant, working with both the EC and the European Space Agency (ESA) among other clients.



Simón Cancela received the M.Sc. degree in advanced mathematics from the Universidad Complutense de Madrid, Madrid, Spain, 2014.

In 2015, he joined GMV and has been involved in projects related to the design and development of GNSS authentication, antispoofing algorithms, and high accuracy positioning.

Rafael Terris-Gallego (Member, IEEE) received the M.Sc. degrees in telecommunication engineering from the Universitat Politecnica de Catalunya (UPC), Barcelona, Spain, and in digital communications systems from Telecom Bretagne (IMT Atlantique), Brest, France, both in 2001.

From 2001 to 2004, he worked with Wavecom France (Sierra Wireless) as a Researcher in mobile communications. From 2005 to 2015, he worked with Indra Barcelona as an Engineer in satellite communications projects, most of them funded by the European Space Agency (ESA). He is currently an Adjunct Lecturer with the Universitat Autonoma de Barcelona (UAB), a Teaching Consultant with the Universitat Oberta de Catalunya (UOC), and a Researcher in GNSS with the Institute of Space Studies of Catalonia (IEEC).

Jose A. López-Salcedo (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in telecommunications engineering from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2001 and 2007, respectively.

Since 2006, he has been with the Universitat Autonoma de Barcelona, Bellaterra, Spain, where he is currently a Professor, and also affiliated with the Institute of Spatial Studies of Catalonia (IEEC). Since 2019, he has been the Secretary/Treasurer of the IEEE AESS Spain Chapter, and since 2020, a Member of the Editorial Committee of the Korean Institute of Positioning, Navigation and Timing (IPNT). His research interests lie on the field of GNSS and LEO-PNT signal processing.

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

From 2002 to 2005, he worked with the European Space Agency on the design of the Galileo system and receivers. He is currently a Professor with the Universitat Autonoma de Barcelona, Bellaterra, Spain, and also with the Institute of Spatial Studies of Catalonia. He is the Co-Founder of Loctio, a start-up providing low-energy GNSS positioning solutions for IoT. Since 2019, he has been the President of the IEEE AESS Spain Chapter. His research interests include GNSS signal processing and 5G localization.

Andrea Dalla Chiara received the M.Sc. degree in microelectronics from the University of Padova, Padua, Italy, in 2006, and the Ph.D. degree in information technology with focus on instrumentation and measurements and RF interference, Universitá degli Studi di Padova, Padova, Italy, in 2010.

He is currently leading the Simulation Tools Division in Qascom S.r.l., formerly developer of GNSS/SBAS simulators and receivers with focus on authentication.

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.

Carlo Sarto is currently the Head of Security Engineering with Qascom S.r.l., Bassano del Grappa, Italy. He has more than 15 years of experience in GNSS cybersecurity. He is currently responsible for the coordination of engineering activities related to security of navigation and communication satellite systems at ground, satellite, and user segment.

Daniel Blonski is a Navigation System Performance Engineer with ESTEC, ESA, where he is contributing to the development of the European Navigation Systems as a Member of the ESA Directorate of Navigation.

Javier de Blas received the M.Sc. degree in aeronautical engineering and in airport systems from the Technical University of Madrid (UPM), Madrid, Spain, in 2004.

He has more than 16 years of experience in the aerospace industry. He currently works with EUSPA as a Galileo Commercial and High Accuracy Service Manager.

Authorized licensed use limited to: Universitat Autonoma De Barcelona. Downloaded on August 17,2024 at 15:01:58 UTC from IEEE Xplore. Restrictions apply.