Improving GNSS spoofing awareness in smartphones via statistical processing of raw measurements

Received 20 January 2023; revised 1 March 2023; accepted 19 March 2023. Date of publication 27 March 2023; date of current version 6 April 2023.

Digital Object Identifier 10.1109/OJCOMS.2023.3260905

Improving GNSS Spoofing Awareness in Smartphones via Statistical Processing of Raw Measurements

AKMAL RUSTAMOV (Student Member IEEE), ALEX MINETTO (Member, IEEE), AND FABIO DOVIS (Member, IEEE)

Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Turin, Italy

CORRESPONDING AUTHOR: A. RUSTAMOV (e-mail: akmal.rustamov@polito.it)

The work of Alex Minetto was supported by the Programma Operativo Nazionale (PON) Ricerca e Innovazione of the Italian Ministry of University and Research (MUR) under Contract 32-G-13427-5 DM 1062/2021.

ABSTRACT Due to the low received power of Global Navigation Satellite Signals (GNSS), the performance of GNSS receivers can be disrupted by anthropogenic radio frequency interferences, with intentional jamming and spoofing activities being among the most critical threats. It is reported in the literature that modern, GNSS-equipped Android smartphones are generally resistant to simplistic spoofing, and many recent contributions support such a biased belief. In this paper, we present the results of a test campaign designed to further stress the resilience of such devices to simplistic spoofing attacks and highlight their actual vulnerability. We then propose an effective spoofing detection technique, that exploits the spatial and temporal correlation of the counterfeit signals by leveraging the statistical analysis of raw GNSS measurements. By not requiring access to the low signal processing level of the GNSS receiver, the proposed solution applies to any device embedding a GNSS receiver that provides raw GNSS measurements, such as current Android smartphones. Vulnerability analysis and validation of the proposed technique were conducted in a controlled environment by transmitting realistic, counterfeit Global Positioning System L1/CA navigation signals to a variety of Android smartphones embedding also different GNSS chipsets. We show that, under proper conditions, the devices were vulnerable to the attacks and that the effects were visible through their raw measurements, i.e., Carrier-to-noise ratio (C/N_0), pseudo-range measurements, and position estimates. In particular, the study demonstrates that cross-correlation between the C/N_0 time series provided by each device for different GNSS satellites increases under spoofing conditions, thus constituting an effective metric to detect the attack within a few seconds.

INDEX TERMS Radio frequency interference, simplistic spoofing, global navigation satellite system (GNSS), GNSS receiver, smartphone, GNSS raw measurements.

I. INTRODUCTION

WITH the rapid development of positioning and navigation technology based on the Global Navigation Satellite System (GNSS), mass-market applications have significantly increased with a demand for more accurate and reliable Positioning, Navigation and Timing (PNT) services. According to the latest market analysis issued by International Data Corporation, global smartphone shipments increased by 1.6 % in 2022 compared to last year [1]. Moreover, the European Agency for the Space Programme market report 2022 forecasts that, by 2031, more than 10 billion GNSS devices will be in use across the world and currently global smartphone and wearable sales contribute to roughly 91% of global shipments [2]. In May 2016, Google announced the disclosure of raw GNSS measurements starting with Android™ 7 [3]. For the first time, developers could access carrier and code measurements, internal clock information and decoded navigation messages from

mass-market devices. To date, raw GNSS measurements support is mandatory for devices running Android™ 10 (API level 29) or higher. Around 82 % of available Android™ phones currently support raw measurements data [3]. This data include internal clock measurements like the time of signal reception, clock drift, clock discontinuities, etc., and the GNSS receiver measurements such as the received GNSS satellite time, i.e., Time of Week (TOW), Doppler frequency, carrier phase measurements as well as constellation status, and further navigation data. The full description of the available raw measurements can be found in [3]. More recently, the Google Service Framework™ has provided Automatic Gain Control (AGC) measurements through updated Android™ classes, with the release of Android™ API 9.0. However, not all the GNSS chipsets are fully compliant with all those measurements and the quality of such data may vary from device to device [4]. Their use can lead to improved GNSS performance by opening the door to more advanced processing techniques previously reserved only to high-end GNSS receivers. These benefits have been demonstrated for code-based, aided, differential and precise point positioning [4], [5], [6]. Furthermore, collaborative navigation approaches leveraged raw measurements to offer a naive collaborative distancing technique [7] and a GNSS-only enhancement of Position, Velocity, Timing (PVT) estimation accuracy [8]. Although the standalone position computed from raw GNSS data may not be as accurate as the ones obtained through an onboard sensor and network integration (i.e., Google Fused Location Provider), the use of raw GNSS data and ad-hoc implemented algorithms can improve the solution with respect to unaided, standalone GNSS solutions. However, positioning improvement is not the only possible exploitation of raw measurements. Since they provide, to a certain extent, an insight of the processing taking place inside the chipset, they can be used to analyse the effects of radio-frequency (RF) impairments affecting the GNSS received signals, as well as to compute new metrics for the quality assessment of the associated output solution [9]. As an example, it is known that due to the weakness of GNSS signals, GNSS receiver performance can be easily disrupted by anthropogenic interferences, with jamming and spoofing activities being critical threats in this context. Swept-frequency and frequency-modulated jamming are typical intentional Radio Frequency Interference (RFI) that can be emitted by personal privacy devices with a carrier frequency that varies across GNSS bands. Spoofing, on the other hand, refers to the transmission of counterfeit, yet plausible GNSS signals with the intent of inducing false PVT estimates at a victim's receiver. Countermeasures for jamming and spoofing threats have been extensively discussed and proposed in the literature [10], and, as far as spoofing is concerned and targeted by our test campaign, a brief review will be covered in this article. As recalled later on, many of the proposed spoofing detection techniques require the implementation of sophisticated algorithms that need to have access to the low-level signal processing stages of the GNSS receiver in order to be effective against simplistic to advanced spoofing attacks [11], [12], [13], [14], [15], [16].

A classical technique for spoofing detection, based on the carrier-to-noise ratio (C/N_0), is proposed in [13], where the measured C/N_0 of received GNSS signals are compared to a known or expected value. The C/N_0 measures the strength of the carrier signal compared to the background noise. If C/N_0 is significantly lower than the expected value, it could indicate that an unspecified RFI is ongoing, and the signal should be discarded. However, if a spoofing signal is unspread, it can show a C/N_0 value within a nominal range, even though jamming or spoofing is performed. Detection of such malicious actions can be difficult in this case, as the C/N_0 measurement may not show any abnormalities. In such cases, other techniques operating over frequency or time domains, or the use of integrity messages from augmentation systems, can be used in conjunction with C/N_0 monitoring to provide a more robust spoofing detection. Some of these techniques can detect spoofing even when the spoofer uses the same codes, frequencies, and power levels as the legitimate signal [17].

However, such techniques might not be usable in systems and platforms embedding a GNSS chipset for which only a limited number of outputs is available to the user or to the higher application layers, such as in smartphones and several other mass-market devices. For this reason, it is of interest to devise interference detection and classification techniques, based only on the observation and the statistical processing of common outputs of GNSS receivers, and in particular of the raw measurements provided by Android™ smartphones as well as by a growing number of low-cost receivers, as reported in [11].

In this paper, taking the smartphone platform as a reference case study, we propose and assess the performance of a technique for the detection of single-antenna spoofing attacks. The proposed solution exploits the spatial and temporal correlation of the spoofing signals, and it is validated through an experimental campaign based on the analysis of the correlation of the raw output data provided by various Android™ smartphones. Therefore, the present article aims at

- analyzing the effect of single-antenna, simplistic spoofing attacks on the raw GNSS measurements provided by different smartphones embedding various GNSS chipsets
- identifying the most suitable time series of raw GNSS measurements and a suitable figure of merit allowing for a prompt and robust detection of the attacks
- defining a methodology for the analysis of the raw data of interest and of the associated figure of merit towards an effective detection of the attack
- experimentally assess the proposed technique through data collections retrieved from an on-field test campaign including 18 devices

The rest of the paper is organised as follows: Section II recalls reference studies and fundamental aspects of spoofing

attacks against GNSS receivers. Section III describes the setup for the vulnerability analysis and the effects on raw GNSS measurements and position estimation. Section IV introduces the methodology for a spoofing detection strategy exclusively based on raw GNSS data. A performance assessment is then presented in Section V and, eventually, conclusions are drawn in Section VI.

II. BACKGROUND AND RELATED WORKS

Recent studies have proposed various techniques for investigating the impact of spoofing attacks on GNSS receivers. Such an effort has been fundamental to enhance security and reliability in GNSS-based applications.

A GNSS satellite simulator was utilized for the first time in a GNSS spoofing experiment in [16]. In [18] researchers investigated the practical aspects of a satellite lock takeover, where a victim receives spoofed signals after first being locked on to legitimate Global Positioning System (GPS) signals. In [19], researchers at the University of Texas (Austin) developed a portable low-cost GPS intermediate spoofer. A successful spoofing attack was carried out on a commercial super yacht using intermediate spoofing techniques, thus highlighting the potential threat against civil PNT systems. Various techniques have since been developed to evaluate the active resilience of GNSS devices and to mitigate the risk of spoofing attacks for improved GNSS security in [20].

Early countermeasures were proposed in [21] to counteract simplistic spoofing attacks. Later advances in the field fostered the development of advanced spoofing detection and mitigation techniques at various stages of signal processing in GNSS receivers [22]. While high-end GNSS receivers now implement spoofing alert systems at their application layers, Android™ smartphones do not provide proper warnings to the user yet, and effective spoofing attacks may stealthily hinder their PNT capabilities. Indeed, upcoming GPS Chips-Message Robust Authentication (Chimera) [23] and Galileo Open Service Navigation Message Authentication [24] services allow receivers to be resilient against counterfeit signals at the cost of implementing the respective authentication algorithms.

At the present, it is worth examining the potential effects of intentional interference on mass-market GNSS-equipped devices, as well as assessing the resilience of their embedded receivers and their capability to live detect ongoing attacks with reasonable latency. Some demonstrations of spoofing against Google's Android™ Operating System (OS) were presented in [25] with realistic spoofing and fake Google Maps™ integration. This work demonstrated that spoofing might impact the device's navigation unit with obvious cascading effects on popular location-based services. In [26], Unicorn Team demonstrated the risk of spoofing attacks by recording legitimate GPS signal through an Ettus Research USRP™ B210 and replaying them by an Software Defined Radio (SDR) platform, i.e., BladeRF™, to effectively fool PVT solutions in a smartphone. The attack succeeded in highlighting smartphone vulnerability, as demonstrated by a PVT logging app installed on the phone. In [27] a simple spoofing methodology was demonstrated as being capable of fooling the navigation solution of a smartphone using SDR and complementary equipment. In [28], inertial navigation sensors such as magnetometer, accelerometer, and barometer were used for triggering possible spoofing event detection in smartphones. By exploiting the availability of GNSS raw measurements, in [29], [30], the impact of spoofing attacks against mobile phones were analysed and specific techniques were suggested to enhance security such as the use of cheap accelerometers together with the monitoring of raw GNSS measurements. The possibility to compare or combine metrics to better identifies spoofing and meaconing attacks was also investigated in [31]. In the study, GNSS anti-spoofing defense were proposed based upon a cooperative positioning approach leveraging the exchange of raw GNSS measurements. The results allowed the identification of possible metrics to be monitored to identify malicious attacks against the positioning and navigation systems in mass-market connected devices. In [32] researchers provided a mobile application for detecting GNSS jamming and spoofing. The application used four different methods to detect attacks: comparing the GNSS and network locations, checking the Android™ mock location flag, comparing the GNSS and system times, and observing the AGC and carrier to noise density ratio (C/N_0) signal metrics. In [33] the authors looked at AGC measurements from multiple smartphone models which have different GNSS chipsets, assesse their behavior under RFI, and point out the current limitations, and improvements that would assist in its usage as a GNSS RFI indicator. The Signal Quality Monitoring Technique, a spoofing-detection methodology, was presented in [34] for realistic spoofing scenarios. This solution is based on the quality of the correlation of the incoming signal and the receiver's local replica and on the cooperative use of a pair of extra correlators to find vestiges of the signal. In [35], the authors suggested how the National Marine Electronics Association messages provided by the GNSS receivers can be utilized to detect instances of spoofing and identify suspicious, potentially spoofed satellite signals. Authors in [36], by appropriately evaluating key measurements supplied in the raw GNSS engine and using software tools native to the Android™ OS, demonstrated the efficiency of Android™ smartphones in reporting interference occurrences as per the Standardization of GNSS Threat Reporting and Receiver Testing through International Knowledge Exchange, Experimentation and Exploitation Threat Monitoring and Reporting standard. A further research work attempted to simulate cost-effective and realistic spoofing attacks by evaluating their impact on output raw measurements [44]. The main limitation of the study was its narrow focus on only a few GNSS chipsets integrated into consumer devices, which may not be representative of the broader range of devices available in the market.

Moreover, it was observed that the Android™ smartphones under the test were generally resilient. A recent work proposes a combined jamming and spoofing detection technique based on AGC and C/N_0 observations [45]. The proposed strategy leverages two theoretical assumptions from [46], [47]

1) if AGC value decreases and C/N_0 decreases, jamming is likely.
2) if AGC value decreases and C/N_0 is relatively constant, spoofing is more likely than jamming.

Despite offering a relatively simple detection algorithm, it has to be remarked that, for many devices, AGC measurements may be an unreliable metric due to the rough resolution of the values and the fact that it is not even provided as an output by some GNSS chipsets. Depending on the different Android™ versions, AGC values are not granted for old devices and, generally speaking, they are not as reliable as other types of raw GNSS measurements, as highlighted in [45]. Furthermore, standalone C/N_0 time series have to be compared with a pre-defined threshold that may be not straightforward to be determined. Therefore, in order to provide a more robust detection, the algorithm is hybridized in an Android™ application named GNSS Alarm in charge to concurrently

- compare GNSS estimated location and estimates from other location providers (e.g., network)
- check for the Android™ mock location flag
- compare the GNSS and Android™ system times.

In this work, starting from the results presented in the existing literature, an extended investigation is performed, testing a wider variety of Android™ smartphones and spoofing attacks in different scenarios via an extensive test campaign, that unveils the actual, unsolved vulnerability of the smartphones against these threats. Differently from the approaches recalled in this literature review, our technique aims at

- leveraging a single data source, by looking for a unique figure of merit that only relies on GNSS data with no need for external location providers or access to O.S. flags
- detecting spoofing on a signal-basis and not only as an aggregated flag (constellation/band), thus providing a more detailed view of the attack (if required). It is worth remarking that spoofing attacks may affect only a subset of available satellites and AGC data cannot provide such a detailed view
- relying on a threshold that is independent from the magnitude of the data under analysis and does not require further normalization or runtime updates.

A. SPOOFING MODELING AND CLASSIFICATIONS

Unstructured RFI such as jamming disturbance can significantly impair the receiver by disrupting its operational capabilities at the early signal processing stages. On the other hand, spoofing disturbances act stealthily, as the receiver operation is typically not interrupted from a user standpoint. Spoofing methodologies are mostly classified on the basis of the time-coherence of the spoofing signals and their legitimate counterparts. The difficulty in performing coherent attacks also determines the practical feasibility and associated risk of such threats [37]. Furthermore, the possibility of detection from a receiver's standpoint may rise a further level of classification [10]. In [38], researchers classified spoofing attacks using a multilayered model, distinguishing between development architectures, acquisition strategy, control strategy, and application. This allowed them to assess the risks and strategies of operational spoofers with prevention. Depending on the features of the spoofing and the complexity of the attack, it is possible to classify these disturbances into three categories: *simplistic*, *intermediate* and *sophisticated* spoofing attacks. They are recalled hereafter for the sake of completeness [10], [16], [37].

1) SIMPLISTIC OR ASYNCHRONOUS SPOOFING

These attacks are characterized as the incoherent transmission of counterfeit GNSS signals over a pre-determined bandwidth aiming at forcing victim receivers to estimate a fake PVT solution. The lack of synchronisation between spoofers and GNSS timescale can be often used to detect ongoing attacks [45]. This class of spoofer can be also built by using a signal simulator that re-transmits counterfeit signals by means of mass-market SDR components [39].

2) INTERMEDIATE OR SYNCHRONOUS SPOOFING

This attack foresees a spoofer architecture embedding a GNSS, built-in receiver that acquire and tracks legitimate GNSS signals in order to coherently generate their counterfeit counterparts. By receiving real-time GNSS signals and estimating the main parameters of interest (i.e., code phase offset and Doppler shift) the spoofer can perform real-time signal transmission of the counterfeit signal by modifying these parameters on its need. A downside of an intermediate spoofing attack is that, in order to be effective, some a-priori information about the victim receiver must be known. To successfully mislead the target PVT estimate, different factors must be known, except in the case of a self-spoofing scenario in which the spoofer and the victim receiver may be co-located. Some implementations of intermediate spoofing scenarios of GPS signals to exploit a modified software-defined receiver integrated with the front-end, are presented in [10].

3) SOPHISTICATED OR MULTI-ANTENNA SYNCHRONOUS SPOOFING

This attack is also referred to as *nulling attack* and it aims at transmitting a disruptive interference signal along with counterfeit, spoofing signals. The use of multiple transmitters increases the effectiveness of the attack against physical detection methods based, for instance, on the angle of arrival. Sophisticated spoofing is the most insidious technique as

it takes control of the target receiver without being typically detected. As described in [40], the malicious action leveraged a soft-take-over through a time-synchronised transmission. It starts with a low level of power which is increased slowly till the receiver has acquired and started to track the spoofed signals. In [13], research conducted sophisticated spoofing scenarios in a multi-layered processing architecture. However, this type of spoofing uses multiple antennas to broadcast GNSS signals, thus overcoming state-of-the-art anti-spoofing countermeasures. Practically, this threat is rarely deployed due to its high cost and complexity and is typically not affordable without advanced expertise.

Due to its affordable cost and practicability, simplistic spoofing is the target threat we experimentally addressed in the current study. However, in the following, single-antenna attacks are modelled that cover, in principle, both simplistic and intermediate spoofing scenarios.

B. LEGITIMATE AND COUNTERFEIT SIGNALS MODELING

In absence of interferences, the GNSS signal received at the antenna can be modeled as the sum of N_s independent satellites' signals

$$x_{f_c}(t) = \sum_{i=1}^{N_s}\Big[\sqrt{P_{R,i}}D_i(t-\tau_i)C_i(t-\tau_i) \times \cos\big(2\pi\left(f_c + f_{d,i}(t)\right)t + \triangle\theta_i\big)\Big] + n(t) \quad (1)$$

where $P_{R,i}$ is the received signal power, $D_i(t)$ is the navigation data stream, $C_i(t)$ is the pseudo-random code sequence, f_c is the carrier frequency shifted by the observed Doppler shift $f_{d,i}$, τ_i is the propagation delay and $\triangle\theta_i$ is the phase offset. Eventually, $n(t)$ is the thermal noise contribution. The received power of each signal, P_{R_i}, reflects the unique properties of the propagation path it covered between transmitting and receiving antennas.

1) MODELLING OF SPOOFING GNSS SIGNALS

In order to fake a GNSS receiver, a spoofer must replicate all the components of the navigation signals defined in (1), such as its spreading code, Radio Frequency (RF) carrier, and the navigation data symbols of the selected constellation. A simplistic GNSS spoofer generates and transmits GNSS-like signals. However, it cannot keep phase and time coherence w.r.t. to the legitimate signals without an external time and frequency sources. The generated counterfeit signals have a similar structure to the legitimate signals, however, they may differ in terms of Doppler and phase shifts of both code and carrier. Furthermore, different power levels are usually observed at the receiver location for spoofing and legitimate signals, respectively. Advanced attacks may calibrate the signal power to be similar enough to the received power of each legitimate GNSS signal. However, such a calibration would require accurate knowledge of the attacker-to-victim range, thus of the victim's location, as typically addressed by sophisticated spoofing actions. In the following, spoofing GNSS signals will be identified through the apex $(\cdot)^{(S)}$. A simplistic spoofer will generate N_{sp} counterfeit signals characterized by code delay $\tau_i^{(S)}$, carrier phase $\triangle\theta_i^{(S)}$, Doppler shift $f_{d,i}^{(S)}(t)$ and data bit stream $\hat{D}_i$ for $i = 1, 2, 3, \ldots, N_{sp}$. A further Doppler shift, $f_d^{(S)}(t)$ may be introduced by the relative kinematics of the transmitting and receiving antennas and is assumed equal to zero when both are static or carried on the same moving rigid body. The expression for the sum of N_{sp} single frequency, single constellation spoofed signals is

$$x_{f_c}^{(S)}(t) = \sum_{i=1}^{N_{sp}}\Big[\sqrt{P_{R,i}^{(S)}}\hat{D}_i\left(t-\tau_i^{(S)}\right)C_i\left(t-\tau_i^{(S)}\right) \times \cos\Big(2\pi\left(f_c + f_{d,i}^{(S)}(t) + f_d^{(S)}(t)\right)t + \triangle\theta_i^{(S)}\Big)\Big] + n(t) \quad (2)$$

where the received power at the antenna, $P_{R,i}^{(S)}$, reflects the different amplitude attributed to each signal to simulate different path losses, and the $\hat{D}_i$ highlights possible differences w.r.t. the legitimate navigation message data stream foreseen in (1). The value of $P_{R,i}^{(S)}$, as for the real signals, may actually change over time, but differently from the case of (1), such variations depend on

- the misalignment of transmitting and receiving antennas as well as any changes in their relative heading during the spoofing attack
- the fading effects introduced by the terrestrial channel and mostly due to multipath which is very relevant when the attacker is at the same altitude of the victim receiver.

When a GNSS receiver is under a spoofing attack, it receives both authentic and spoofed signals, and additive thermal noise affects their sum. Therefore, the total signal at the victim receiver's front-end is modeled as

$$x_{tot}(t) = \bar{x}_{f_c}(t) + \bar{x}_{f_c}^{(S)}(t) + n(t) \quad (3)$$

where the notation $\bar{x}$ indicates the noiseless, legitimate and spoofing signals derived from (1) and (2) by neglecting the respective noise terms. Without any lack of generality, f_c will be referred hereafter as to GPS L1/CA center frequency, i.e., 1575.42 MHz.

2) IDENTIFICATION OF SPOOFING GNSS SIGNALS

From a geometrical and physical standpoint, legitimate GNSS signals propagate through different channels. The free space path loss mainly contributes to the differences in the received power observed for each satellite. Multipath-related constructive and destructive interferences may be responsible for fluctuations in the received power, being conditioned by the elevation at which each satellite is observed. Differently from the authentic GNSS signals, all the spoofing signals travel through the same propagation path from the spoofer's to the receiver's antenna, thus experiencing a common, yet

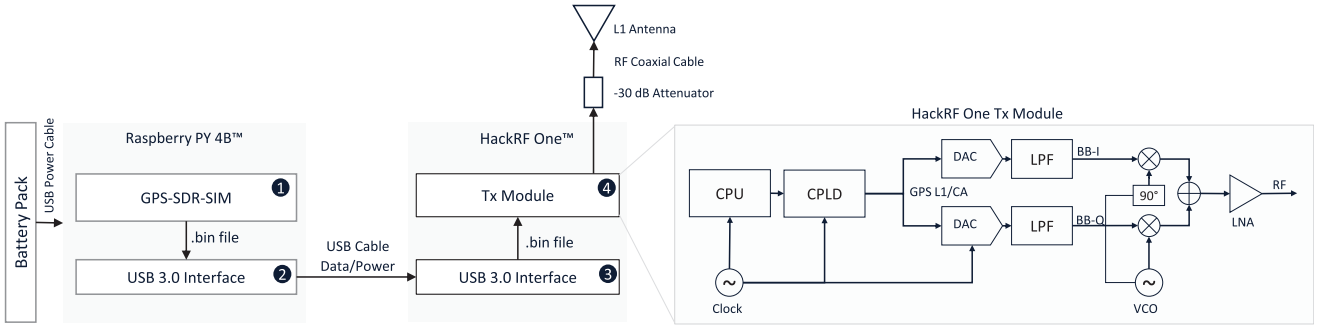


FIGURE 1. High-level diagram of the low-cost portable spoofer (left). The transmitter architecture (right) produces I/Q modulated GPS L1/CA signals transmitted at L1 center frequency $f_c = 1575.42$ MHz, through the digital-to-analog conversion and up-mixing of baseband signal samples (BB-I, BB-Q).

unique physical channel. The aforementioned power variations will then reflect in a similar way on each generated satellite's signal by introducing strong spatial and temporal correlation. In the following preliminary analysis we look for such correlations through the time series of the GNSS raw measurements to identify the spoofing attacks.

III. PRELIMINARY ANALYSIS

Through the assessment of the smartphone vulnerability to simplistic spoofing attacks, the following analysis aims at identifying the set of observables, among the available GNSS raw measurements, being suitable to the design of the proposed spoofing detection method.

A. EXPERIMENTAL SETUP AND TEST PROCEDURE

To perform the vulnerability analysis, we developed a low-cost portable spoofer based on a Great Scott Gadgets™ HackRF One™ platform [39] and a Raspberry™ PI 4B. A high-level diagram of the system is shown in Figure 1. The HackRF One™ is a low-cost, open-source SDR front-end allowing fast and accurate RF signal transmission from binary files (.bin). Such files contain the numerical samples of Intermediate Frequency (IF) or baseband signals. The device is a transceiver supporting center frequencies from 1 MHz to 6 GHz with tunable transmitting power and channel bandwidth. The software used to numerically generate the spoofed GPS signal is the GPS-SDR-SIM [41], an open GPS L1 C/A signal generator toolbox distributed with an MIT license [42]. The attack was planned to simulate a static position, and all the visible satellites belonging to the GPS constellations and their signals were transmitted by means of the SDR transceiver. An optional 10 MHz reference, i.e., Oven Controlled Crystal (Xtal) Oscillator was connected to the front-end to discipline the signal generation. The power supply was provided through a mass-market, 10000 mAh battery pack compliant to the supply specification of the Raspberry™ PI 4B. The HackRF One™ can be in turn supplied by the Raspberry™ PI through its USB 3.0 interface. The spoofing attack was performed through the portable spoofer according to the following procedure:

1) *Numerical Counterfeit signal generation*. The coordinates of the fake static location were chosen and configured to the GPS-SDR-SIM. The software also requires in input the daily GPS broadcast ephemeris (i.e., RINEX v2 brdc file). Once such inputs are provided, it generates the simulated pseudorange and Doppler shifts for the GPS satellites in view. Such a simulated data is used to produce a binary file with In-phase/Quadrature (IQ) samples of the complex baseband GNSS signal, ready to be reproduced by the SDR front-end (i.e., HackRF One™), according to the block diagram in Figure 1.
2) *A.bin file transmission*. The.bin file is read by the HackRF One™ through the USB interface of the Raspberry™ PI 4B.
3) *Digital to analogue conversion*. The transmitting module of the front-end (HackRF One™) is in charge to perform the digital-to-analog conversion by mixing the baseband signal provided at step 2 to the carrier frequency generated through the VCO (i.e., GPS L1 C/A), thus, transmitting I/Q modulated GNSS signals in L1/CA band. A block diagram is provided on the right side of Figure 1 that shows its architecture.
4) *RF signal transmission*. After baseband signal samples are generated, HackRF One™ mixes them to the carrier frequency, f_c, and transmits the RF signal through the antenna of the SDR platform. Specifically, the transmission command is used to spread the IF or baseband samples using HackRF One™), at L1/CA center frequency $f_c = 1575.42$ MHz.

1) DEVICES UNDER TEST

A variety of Android™ smartphones with single and multi-frequency GNSS chipsets were chosen to test the effects of the simplistic spoofing attacks performed through the aforementioned portable spoofer. The list of devices under test is reported in Table 1. These devices are all equipped with Google Android™ OS and the GNSS Logger Android™ application provided by Google™ was installed for the procurement of GNSS raw measurements. Additionally, their PVT solutions were logged through the Android™ National Marine Electronics Association (NMEA) Tools

TABLE 1. Android™ devices under test and embedded GNSS chipsets with supported frequency bands.

Model	System on chip (SOC)	GPS Bands
Samsung A30	Qualcomm Exynos 7904 Octa	L1
Samsung A32	Qualcomm Snapdragon 720G	L1
Samsung S6	Qualcomm Exynos 7420 Octa	L1
Samsung A20	Qualcomm Exynos 7884 Octa	L1
Samsung Note 8	Qualcomm Snapdragon 835	L1
Samsung Note 20	Qualcomm Exynos 990	L1
Samsung A21s	Qualcomm Snapdragon 720G	L1
Samsung Note 22	Qualcomm Snapdragon 8	L1
Samsung A72	Qualcomm Snapdragon 720G	L1
Xiaomi Redmi 6	Mediatek MT6762 Helio P22	L1
Xiaomi Redmi 6 Pro	Qualcomm Snapdragon 636	L1
Xiaomi Redmi 8	Qualcomm Snapdragon 845	L1+L5
Xiaomi Redmi 8 Pro	Qualcomm Snapdragon 845	L1+L5
Xiaomi Note 9	MediaTek Helio G85	L1+L5
Xiaomi Note 11	Qualcomm Snapdragon 680 4G	L1+L5
Xiaomi 12x	Qualcomm Snapdragon 870	L1+L5
Huawei Y9	Octacore HiSilicon Kirin 710	L1
Honor X8	Qualcomm Snapdragon 680	L1

application, which provides the GNSS standalone position of the smartphone in standard NMEA format [43].

2) TEST METHODOLOGY

Experiments on smartphones were carried out in a dedicated test campaign. Each test foresaw 600 s data collections for two complementary scenarios, in controlled environmental conditions. The range of the spoofer was kept at about 3 m, and, in order to prevent any RFI disturbances beyond the range of the experimental setup, a 30 dB attenuator was applied at the coaxial cable to reduce transmitting signal power levels and limit the spoofer coverage. The actual locations of the smartphones, i.e., the test site, were at N 45°3'52.4711" E 7°39'42.7179", 2022-06-14 at 14:50 UTC +00:00 while the portable spoofer broadcast spoofing signals over GPS L1 band with a fake location at N 45°09'28.5" E 7°34'47.9", 2022-06-14 at 14:00 UTC +00:00 which was approximately 12 km away from the test location. Based on the results of the previous test campaigns [31], [44], we modified the test settings to achieve the most vulnerable conditions under which an Android™ smartphone could be spoofed by a simplistic attack. To this aim, the test has been executed in both normal and pilot/airplane modes. Normal operational modes in smartphone foresees wireless data connectivity that allows smartphones to download updated GNSS ephemeris. Such a data can be used for cross-checking the time-consistency of the received navigation message in rudimentary anti-spoofing techniques. However, in Section III-B it has been shown that no relevant differences have been observed by operating the devices under test in the two modes. For a conservative approach, all the results presented in this article have been obtained in normal operational modes. For an exhaustive analysis, we reproduced two simplistic spoofing scenarios.

1) In Scenario 1, the devices received real GNSS signals for 150 s, then transmitted spoofed signals for the remaining $T = 350$ s. During the initial timespan of 150 s, the devices kept tracking legitimate GNSS signals only.
2) In Scenario 2, the spoofer transmitted counterfeit signals for 350 s, then the devices switched to track live GNSS signals upon spoofing interruption.

TABLE 2. GPS PRNs identifiers distinguished between available and counterfeit satellites signals during the non-spoofed and spoofed scenarios.

GPS satellites subset[1]	GPS L1/CA PRN
Real (legitimate signals)	1, 3, 8, 10, 14, 21, 22, 32
Counterfeit (spoofing signals)	1, 3, 8, 10, 14, 21, 22, 27, 32
Common	1, 3, 8, 10, 14, 21, 22, 32

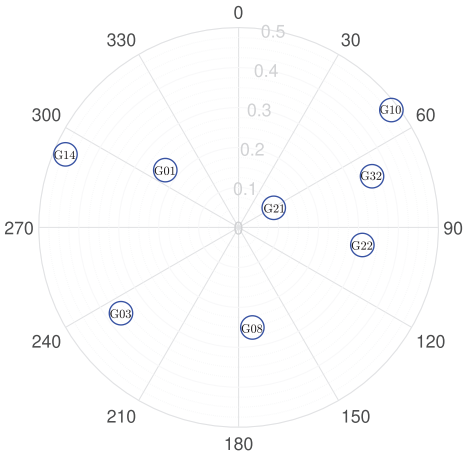
[1] The relative geometry of the constellations as seen by the receiver under test is shown in skyplots of Figure 2

The set of visible real satellite signals during the test campaign is reported in Table 2 for Scenarios 1 and 2. During the spoofing attacks, the set of counterfeit signals were generated to force the GNSS receiver to estimate a faked PVT solution. Counterfeit and real satellite signals are distinguished in Table 2. As it can be seen, such a set includes i) satellite signals that would be broadcasted by satellites that are not actually visible to the receivers at the time and location of the tests, and ii) satellite signals that are already being tracked by the receivers, for which the spoofing signal has to replace the real signals under tracking. The overall satellite skyplot of the visible satellites and of the counterfeit constellation generated during the tests are shown in Figure 2a and Figure 2b, respectively. The skyplots depict azimuth and elevation angles of the satellites w.r.t. the user location. For static users, they highlight the difference in the observed scenarios in terms of relative geometry of the satellites w.r.t. the receiver location.

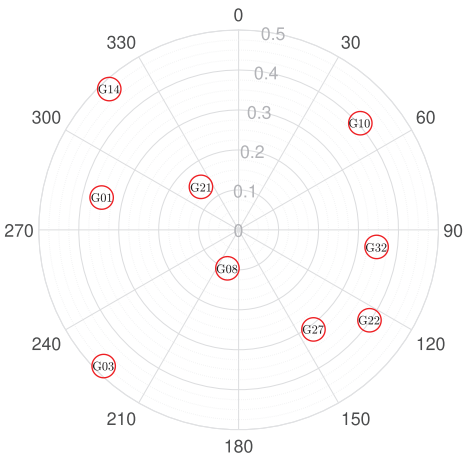
3) RAW GNSS MEASUREMENTS OF INTEREST

Among all the available raw GNSS measurements, the data fields of interest for the investigations pursued within this study are:

- *Automatic Gain Control (AGC).* The AGC implementation in a smartphone acts as a variable gain amplifier adjusting the power of the incoming signal. Changes in the value are typical indicators of power fluctuations of the input signal in a given frequency band. AGC value and its variations affect all the received GNSS signals. Independent effects on each signal cannot be inferred from such data.
- *Carrier-to-noise Density Ratio* (C/N_0): The C/N_0 measures the power density of the useful GNSS signal w.r.t. the noise floor power density and has a direct relationship with the signal strength as well as the accuracy of



(a) Real satellites subset



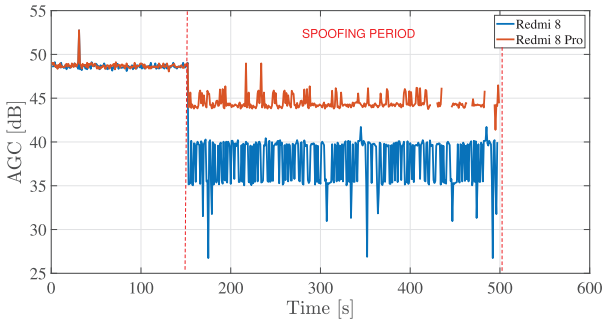
(b) Counterfeit satellites subset

FIGURE 2. Skyplot showing the constellation geometry of the real and counterfeit satellites for the respective legitimate and spoofed signals.

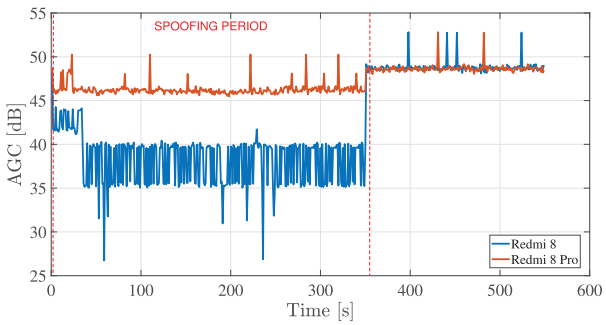
the associated measurements. In fact, strong received signals return high C/N_0 values, typically leading to better signal tracking and PVT determination. Abrupt variations to it can indicate the presence of interference while an unnaturally high value could also indicate the presence of a counterfeit satellite signal. C/N_0 is estimated by each tracking channel independently, therefore data is available for each received signal.

- *Pseudorange and Pseudorange rate* (PrM): The pseudorange is a measurement of the distance between the user and satellite, affected by the clock offset of the receiver clock w.r.t. the satellite timescale. The observation of the behavior of the pseudorange and of its variation over time (i.e., pseudorange rate), allows to inspect the effect of the spoofing signals (when they are tracked by the receiver) and to motivate any impact of the interference on the subsequent PVT solution.

For the sake of completeness, the output position estimates from the computed PVT solutions have been also investigated to provide evidence of the vulnerability of the devices under test to the simplistic spoofing attack. It is worth recalling



(a) Scenario 1



(b) Scenario 2

FIGURE 3. Comparison of samples AGC time series between two Android™ devices under the proposed spoofing scenarios.

that misleading PVT solutions are indeed the usual objective of such deliberate malicious actions.

B. SPOOFING EFFECTS ON SMARTPHONES RAW MEASUREMENTS

In this section, we analyze the effects of the designed spoofing test on the raw GNSS measurements of interest as well as on the position estimates provided by the devices under the test scenarios 1 and 2 described in Section IV. The effects of the spoofing signals are hereafter reported by means of measurements time series. The discussion follows the signal processing flow of a conventional GNSS receiver architecture, i.e., from the AGC to the PVT computation.

1) AUTOMATIC GAIN CONTROL (AGC)

Figure 3 plots the AGC value (in dB) observed under the test Scenarios 1 and 2 for the devices under test. From test Scenario 1 (Figure 3a), it can be seen that the effect of turning on the spoofer is similar to what in-band jamming or interference would cause. Due to the presence of an in-band, powerful signals, the receiver reduces the amplification of the incoming signals. By collaterally attenuating the legitimate GNSS signals, it creates the conditions for the acquisition of counterfeit signals. In the same figure, the AGC amplification dramatically drops from 48 to 45 dB for Redmi 8 Pro of the smartphone, and down to about 40 dB for the Redmi 8 once the spoofing is turned on at time $t = 150$ s. In the Scenario 2, being the spoofing signals broadcast with the

same power level as in Scenario 1, the initial values of the AGC amplification are similar as during the spoofing period in Scenario 1. When spoofing is ended at time $t = 350\,\text{s}$ during the test Scenario 2, the AGC increases back to its initial level, as it can be seen in Figure 3b.

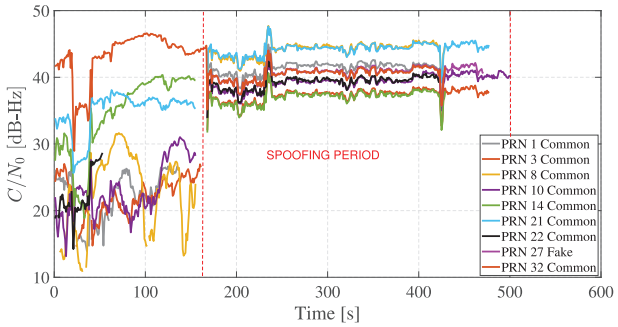
The jump in the AGC value for the Redmi8 device could indeed be due to a loss of lock on authentic signals and subsequent reacquisition and relock on spoofing signals. The strength and persistence of the spoofing signal could also be a factor in determining whether there is a gap in the measurements output or not. If the spoofing signal is strong and persistent enough, it could cause the GNSS receiver to lose lock for a longer period, resulting in a gap in GNSS output. On the other hand, if the spoofing signal is weaker or less persistent, the GNSS receiver may be able to maintain a lock on authentic signals and produce continuous output, even in the presence of the spoofing signal. Hardware and software differences in the GNSS receiver could also play a role in determining the response to spoofing signals. Different receivers may have different sensitivities, filtering capabilities, or other features that affect their ability to be resilient to spoofing attacks.

This variation brings evidence of the presence of the spoofing signal, and detection techniques based on the observation of the AGC level has been indeed proposed [33], [45], [48].

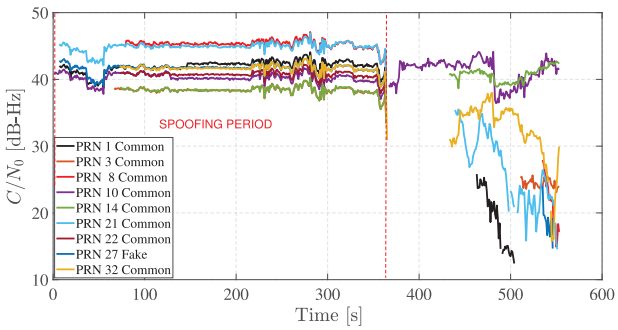
However, by itself, the AGC variation cannot be sufficient to declare the presence of a spoofing signal, but it only allows for raising an alert. Unconventional AGC behaviors may indeed subtend jamming attacks; therefore, the AGC variation has to be cross-checked together with the spectral distortion of the input signal or the C/N_0 value of each channel [49], or further independent metrics.

2) CARRIER-TO-NOISE DENSITY RATIO (C/N_0)

Figure 4 shows GNSS receiver's C/N_0 values estimated for GPS signals for the entire test duration by the Xiaomi RedMi 8 smartphone under test. Similar behaviors can be reported for all the other devices under test without any lack of generality. As it can be seen, in Figure 4a, under non-spoofing conditions (i.e., up to time $t = 150\,\text{s}$, the estimated C/N_0 assumes values between 15 dB-Hz and 45 dB-Hz depending on the strength of the received signal. The individual trends reflect different propagation conditions of each signal, being the transmitting satellites observed at different elevations and azimuth angles with respect to the receiver. Satellites at different elevations have different distances from the user, thus inducing different received signal power. Furthermore, received signals from low-elevation satellites may be degraded by multipath due to the presence of buildings and other obstacles. When the spoofer is turned on at time $t = 150\,\text{s}$, it can be seen in Figure 4a that it acts as generic RFI over the L1 frequency band by disturbing the reception of the legitimate signals up to their loss-of-lock. In parallel, the attack forces the GNSS receiver to acquire and lock on the spoofing signals. However, concerning the counterfeit signals, their received power is higher,



(a) Scenario 1



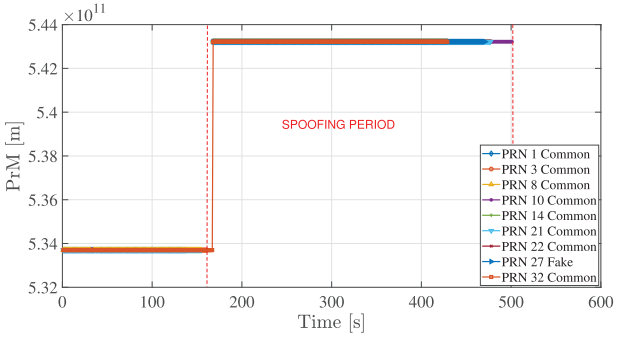
(b) Scenario 2

FIGURE 4. Comparison between sample C/N_0 time series between the proposed spoofing scenarios.

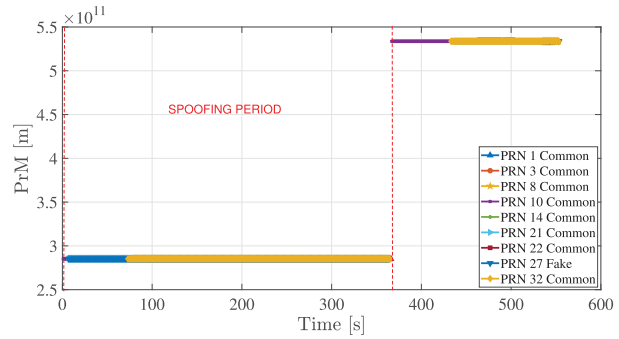
thus leading to a raise of the C/N_0 value. The most relevant observation is the similarity of the behavior over time of the C/N_0, where despite variability in the range 35-45 dB-Hz, there is a remarkable common trend in time. This can be explained considering that the spoofer is emulating a satellite scenario generating signals with different power levels in order to mimic the different distances of the satellites. However, at the same time, all the signals are generated through the same transmitting hardware and are subject to the very same propagation conditions, as discussed through the signal modeling in Section II. Similar remarks can be done by observing Figure 4b for the Scenario 2. It also shows similar trends during the spoofing period and a larger variability and diversity of such trends after the spoofer is turned off. These observations suggest that the correlation between the raw measurements, could be evidence of the spoofing presence and it is the basis of the spoofing detection technique presented and discussed in Section IV.

3) PSEUDORANGE MEASUREMENTS (PRM)

Figure 5 compares the pseudoranges value between all PRNs during the entire test period for the smartphone RedMi 8. When the simplistic spoofer is turned on at time $t = 150\,s$ the spoofed signals are tracked and the pseudorange value is altered accordingly. Some weaker real signal suffers from the in-band jamming effect and is not tracked anymore, such that their pseudorange is not provided during the spoofing period, as it can be seen in Figure 6. As for the common

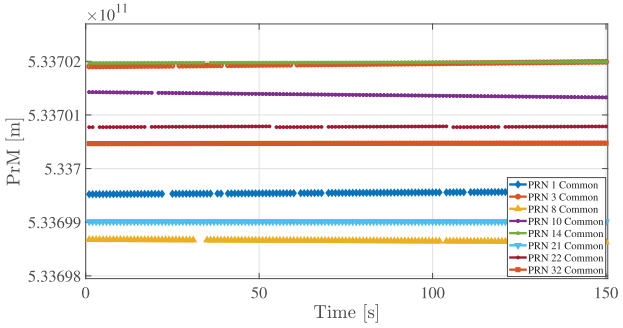


(a) Scenario 1

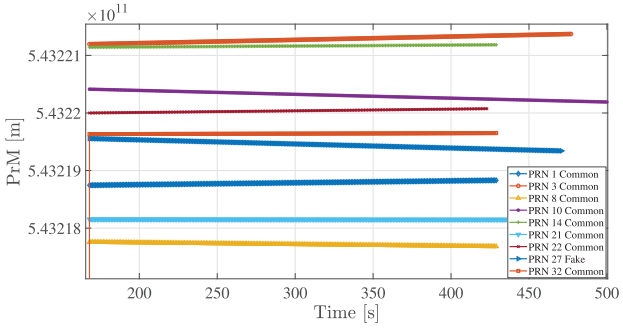


(b) Scenario 2

FIGURE 5. Comparison of PrM time series between the proposed spoofing scenarios.

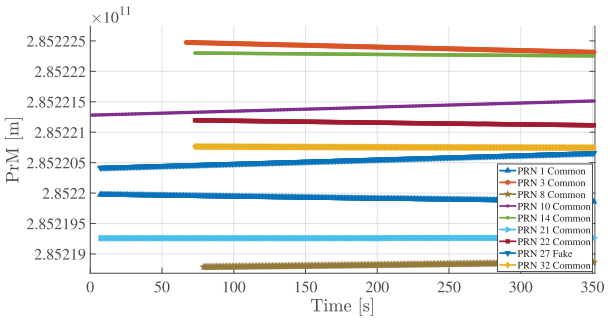


(a)

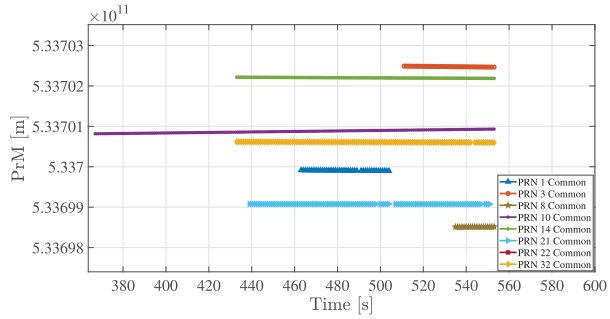


(b)

FIGURE 6. Zoom of the PrM values during test for the Scenario 1: Non spoofing period a) vs. Spoofed period b).



(a) Measurements during spoofing period



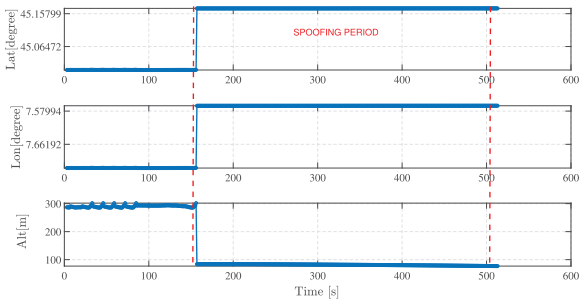
(b) Measurements during non-spoofing period

FIGURE 7. Zoom of the PrM values during the test under Scenario 2.

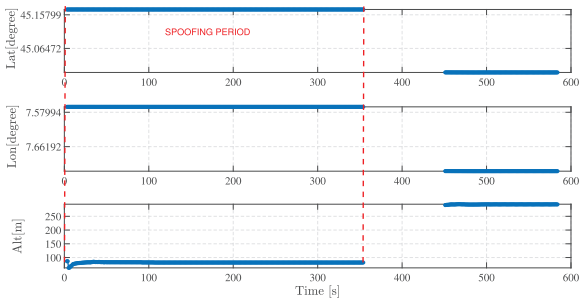
PRNs, the higher power of the spoofed signals forces the receiver to lose the lock and relock on the new signal, as shown by the jump in the pseudorange value. A dual effect can be noticed in Figure 7 when the spoofer is turned off. An interesting finding is that when the signal is switched from real to spoofing, there is a high jump in the pseudoranges that cannot be attributed to the spoofed location. Indeed, the counterfeit location set in these tests would not justify such a large variation. The reason for this is the different user clock biases estimated in spoofing presence. The observed anomalous magnitude of the pseudorange measurements is attributed to an altered estimate of the signal's time-of-flight due to the outdated TOW carried by spoofing signals. In the experiments, the estimation of the pseudorange measurements is based on such a TOW and on the local time at the receiver. When spoofing occurs, local time is not shifted accordingly to the TOW of the spoofed signals and the resulting time of flight becomes higher than expected in nominal conditions.

4) POSITION ESTIMATION

As a cross-check of the spoofing vulnerability, according to the NMEA stream it was observed that both the time and locations of all the smartphones under investigation were successfully spoofed. Figure 8a and Figure 8b represent the shifts in latitude, longitude and altitude reported in the NMEA log files in both the test Scenarios 1 and 2 for the Xiaomi RedMI8 device. The vertical dotted lines in Figure 8 delimit the timespan corresponding to the spoofing period. It can be seen in Figure 9 that two different positions were estimated during the test. The solution is shifted from the real



(a) Scenario 1



(b) Scenario 2

FIGURE 8. Effect of spoofing on Geodetic coordinates (PVT solutions). GPS spoofing: Test Scenario 1 a) vs. Test Scenario 2 b).

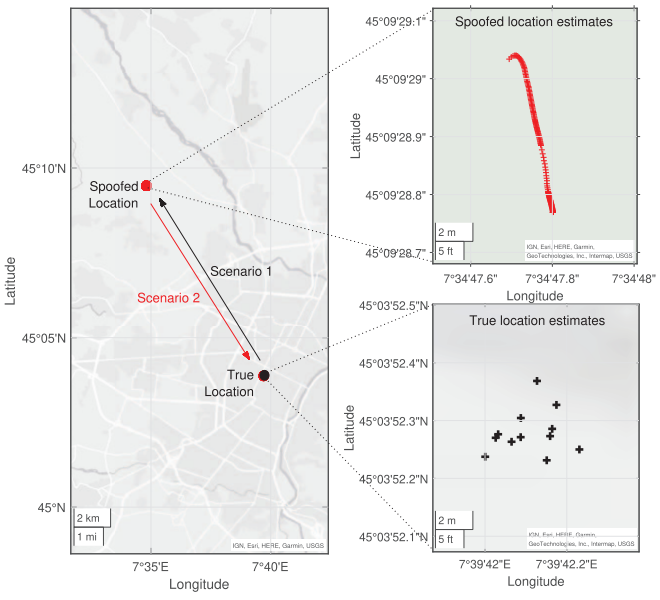


FIGURE 9. Sample effect of a simplistic spoofing attack on the position estimation of the Xiaomi Redmi 8 under test. The arrows in the left subplot indicate the transitions of the estimated locations in the different experimental scenarios.

to the fake location a few seconds after the spoofing starts (Scenario 1), and vice versa when after its end (Scenario 2). It can be noted that in test Scenario 2, Figure 8b shows a discontinuity of operation between $t = 350\,\text{s}$ and $t = 450\,\text{s}$ on the estimated latitude, longitude and altitude coordinates. This is due to the misalignment between real and simulated timescales and also depends on the outdated GPS ephemeris employed for the spoofing signal generation.

IV. METHODOLOGY FOR SPOOFING DETECTION

One of the main observations of the previously described test campaign is that spoofing induces significant variations in the estimated C/N_0 and PrM data, also showing a similar trend over time of these time series for different satellite signals [50]. Based on the measurement model described in Section II-B along with these observations, we introduce a methodology for the analysis of possible correlations between the raw data time series obtained for different tracked signals and observed within a common time window, T_n. In the following analysis, we will focus on the C/N_0 time series as the target data. In fact, the proposed methodology forgets about the physical meaning of the measurements, treating the observed data as time series of noisy raw GNSS measurements that have to be considered as realizations of non-stationary, stochastic processes.

A. SPOOFING DETECTION STRATEGY THROUGH PEARSON CORRELATION COEFFICIENTS

In order to provide a quantitive analysis of the similarity between data series along the experiment time, the pairwise *cross-correlation function* between two-time series X_A and X_B can be computed

$$R_{XY}(t_1, t_2) = E\big[X_A(t_1)X_B^*(t_2)\big] \tag{4}$$

where $E(\cdot)$ represents the mean operator, t_1 and t_2 identifies two generic time instants, and the $(\cdot)^*$ indicates the complex conjugate of the argument. By subtracting the respective mean to each series in (4), we obtain the *cross-covariance function*

$$C_{XY}(t_1, t_2) = E\Big[\big(X_A(t_1) - \mu_{X_A}(t_1)\big) \times \big(X_B(t_2) - \mu_{X_B}(t_2)^*\big)\Big]. \tag{5}$$

In the proposed applications short observation windows are expected to be monitored, thus long-term trends that typically characterize the investigated quantities can be neglected with no lack of validity. In light of this, wide-sense stationarity of the processes can be assumed and (5) is modified as

$$C_{XY}(\tau) = E\Big[\big(X_A(t) - \mu_{X_A}(t)\big) \times \big(X_B(t+\tau) - \mu_{X_B}(t+\tau)\big)^*\Big] \tag{6}$$

where $\tau = t_1 - t_2$, and denotes the independence on the choice of t_1 and t_2 of the cross-covariance, i.e., the cross-covariance of a Wide-Sense Stationary (WSS) process. In order to obtain a scale-free metric of the correlation between the time series, a normalization is introduced in (6), by defining

$$\rho_{X_A,X_B}(\tau) = \frac{C_{XY}(\tau)}{\sigma_{X_A}\sigma_{X_B}} \tag{7}$$

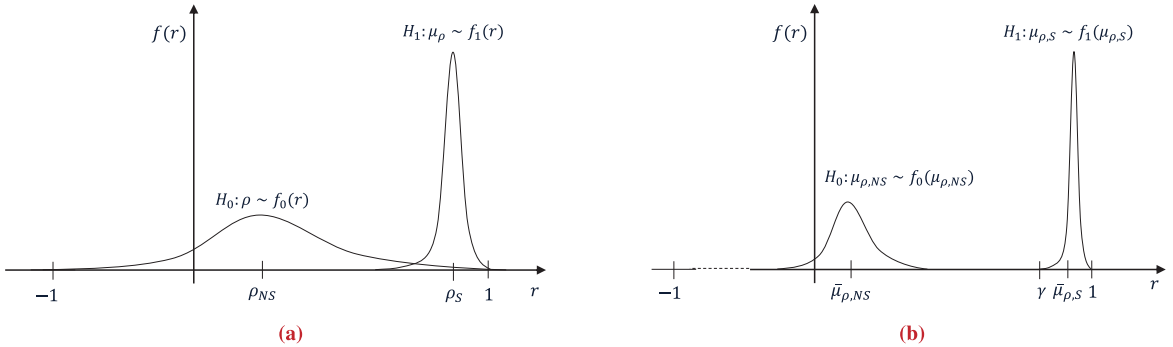


FIGURE 10. Pictorial view of the Neyman-Pearson Criterion applied to spoofing detection based on the Pearson correlation coefficients of C/N_0 time series. PDFs of single Pearson's correlation coefficients (a) and of aggregated data, i.e., average Pearson correlation coefficient (b).

that is known as *Pearson correlation function*. The maximum value assumed by (7) corresponds to the well-known *Pearson correlation coefficient* [51], [52], and is computed as

$$\rho_{X_A,X_B} = \max\left\{\rho_{X_A,X_B}(\tau)\right\} = \frac{\mathrm{cov}(X_A, X_B)}{\sigma_a \sigma_b}. \tag{8}$$

In case X_A and X_B are identical time series (i.e., they are of the same length and assume the same values), the maximum of (7) is located at $\tau = 0$. Furthermore, the Pearson correlation coefficient assumes values in the range $(-1, 1)$. $\rho_{X_A,X_B} = 1$ highlights a perfect positive relationship, while $\rho_{X_A,X_B} = -1$ denotes a perfect negative relationship, and $\rho_{X_A,X_B} = 0$ indicates the absence of a linear relationship between the random variables. Pearson correlation coefficient also relates to the slope of the linear regression between the time series. Therefore, it is exploited in the proposed solution to track temporal and spatial correlation that characterizes the spoofing signals.

B. IMPLEMENTATION OF PEARSON COEFFICIENTS ESTIMATION

In absence of repeatability conditions, multiple realizations of the random processes of interest are not available in the target application. Therefore, sample means and standard deviations in (6) must be estimated. By further assuming ergodicity of the observed data, the time average can be considered in place of the sample means for the estimation of μ_{X_A}, and μ_{X_B}, thus $C_{XY}(\tau)$ can be estimated through (6). Similarly, standard deviations computed on time can be used in place of their statistical counterpart. The proposed implementation acts on pairs of input time series by a) computing their time average, b) independently subtracting them to each, c) performing the discrete cross-correlation, d) normalizing the value by the product of their standard deviation, and eventually extracting its maximum, i.e., the Pearson Correlation Coefficient. An estimate of (8) is hence provided through the *approximated Pearson correlation coefficient*

$$\hat{\rho}_{a,b} = \frac{\sum_{t=1}^{Tn}(a_t - \overline{a})\left(b_t - \overline{b}\right)}{\sqrt{\sum_{t=1}^{Tn}(a_t - \overline{a})^2}\sqrt{\sum_{t=1}^{Tn}\left(b_t - \overline{b}\right)^2}} \tag{9}$$

where T_n is the window size, a_t, b_t are the time series samples observed at the t-th instant, and $\overline{a}, \overline{b}$ are the sample means. The proposed approximated correlation coefficient, $\hat{\rho}_{ab}$, can be computed for all the available pairs to verify their pairwise correlation. It is worth remarking that the estimation accuracy of $\hat{\rho}_{ab}$ can depend on the size of the observation window, T_n. Short windows may lead to misleading correlation information but long windows introduce a considerable latency for the collection of the data samples. A $K \times K$ symmetric, correlation matrix is eventually populated with the estimated correlation coefficients for each pair of tracked GNSS signals (9)

$$P = \begin{bmatrix} 1 & \hat{\rho}_{1,2} & \hat{\rho}_{1,3} & \cdots & \hat{\rho}_{1,K} \\ \hat{\rho}_{2,1} & 1 & \hat{\rho}_{2,3} & \cdots & \hat{\rho}_{2,K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{\rho}_{K,1} & \hat{\rho}_{K,2} & \hat{\rho}_{K,3} & \cdots & 1 \end{bmatrix} \tag{10}$$

and aggregated metrics can be used to build a decision logic over the whole set of tracked signals, such as the *average cross-correlation coefficient* computed on the lower triangular matrix

$$\mu_\rho^{(K)} = \frac{2}{K(k-1)} \sum_{a,b} \hat{\rho}_{a,b} \quad \forall a > b, \; a \in (1, K) \tag{11}$$

where a and b defines rows and columns indices, respectively.

C. DECISION LOGIC

The proposed decision logic for the detection of simplistic spoofing is hence based on a binary decision rule. Two hypotheses are tested in the context of a classical Neyman-Pearson decision problem, as depicted in Figure 10

- H_0: only legitimate GNSS signals are received and tracked. Under such a hypothesis, μ_ρ is typically low due to a poor cross-correlation among the C/N_0 time series, and its distribution follows a given PDF, $f_0(r)$.
- H_1: legitimate and spoofing GNSS signals are concurrently received and spoofing signals are tracked in place of the legitimate ones. Under this hypothesis, μ_ρ is

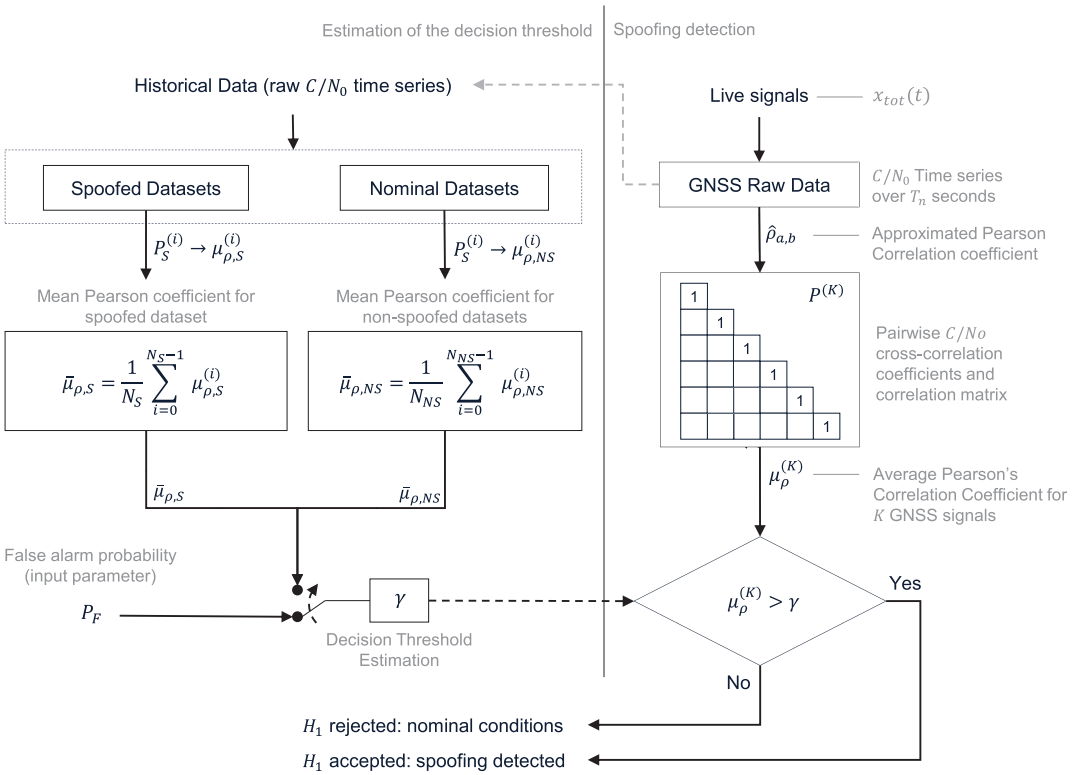


FIGURE 11. Decision logic scheme for the proposed spoofing detection strategy. The decision threshold can be established by fixing the false alarm probability, P_F, or heuristically derived by means of historical datasets.

expected to be as closer to 1 as many spoofed satellites are tracked by the receiver, with a PDF distributed according to $f_1(r)$.

Analytic expressions for $f_0(r)$ and $f_1(r)$ can be approximated through the series

$$f(r) \simeq \frac{2^{n-3}\left(1-\rho^2\right)^{\frac{n-1}{2}}\left(1-R^2\right)^{\frac{n}{2}-2}}{\pi\,\Gamma(n-2)} \times \sum_{k=0}^{\infty}\left[\Gamma\left(\frac{n-1+k}{2}\right)\right]^2\frac{(2R\rho)^k}{k!} \tag{12}$$

where r is a variable defined in the range $(-1, 1)$, $\Gamma(\cdot)$ is the Gamma function, n is the number of Pearson's correlation samples in a given experiment, and ρ is the known level of correlation [53]. The PDF described in (12) is a skew distribution with a skewness factor that increases with ρ. However, according to the Central Limit Theorem, (12) can be transformed through the Fisher transformation

$$z = \frac{1}{2}\ln\left(\frac{1+r}{1-r}\right) = \tanh^{-1}(r) \tag{13}$$

such that (12) approaches a normal distribution as n increase, with standard deviation

$$\sigma_z = \frac{1}{\sqrt{n-3}}. \tag{14}$$

This step is depicted by the plots of Figure 10. Single correlation coefficient will define the distribution of Figure 10a while averaging multiple coefficients will shift the decision problem to the Gaussian-like distributions defined through (13) and centered at the average correlation coefficient, as in Figure 10b. In order to establish a decision threshold, γ, we express the probability of a false alarm as

$$P_F \simeq \int_{\gamma'}^{1} f_0'(z)\,dz = \alpha \tag{15}$$

where $f_0'(z)$ is the transformed PDF according to (13), and α is the design parameter for the decision logic. The threshold γ' is computed by fixing the probability of a false alarm, α, as

$$\gamma' = Q^{-1}(\alpha) \tag{16}$$

where $Q(\cdot)$ is the Marcum Q-function and γ' must be reverted to γ by inverting (13). As an example, by fixing a probability of false alarm of 1.5 % through $\alpha = 0.015$, we obtain $\gamma' \simeq 2.17$. This value corresponds to a threshold $\gamma \simeq 0.5$ for the original PDF (12) of Figure 10b. $\gamma \simeq 0.5$ is hence the value that will be utilized in the experimental validation of our technique to establish the detection. Such a threshold will be also cross-validated by means of experimental datasets in Section V.

1) IMPLEMENTATION OF THE DECISION LOGIC

A block diagram for the implementation of the proposed decision logic is provided in Figure 11. The algorithm aims at

1) determining the *correlation threshold*, γ, under $\mathcal{H}_0$ and $\mathcal{H}_1$ hypothesis. γ can be estimated by fixing the false alarm probability.
2) comparing the current mean correlation coefficient $\mu_\rho^{(K)}$ estimated through real-time data over a window of T_W s, with the threshold γ.
3) deciding for spoofing or non-spoofing conditions within the observed time window by accepting or rejecting $\mathcal{H}_1$ according to the Neyman-Pearson criterion.

For a large set of training datasets, the decision threshold, γ can be heuristically selected by identifying the average correlation coefficients observed under the aforementioned hypothesis. Figure 11 highlights both theoretical and empirical threshold estimation on the right side of the diagram.

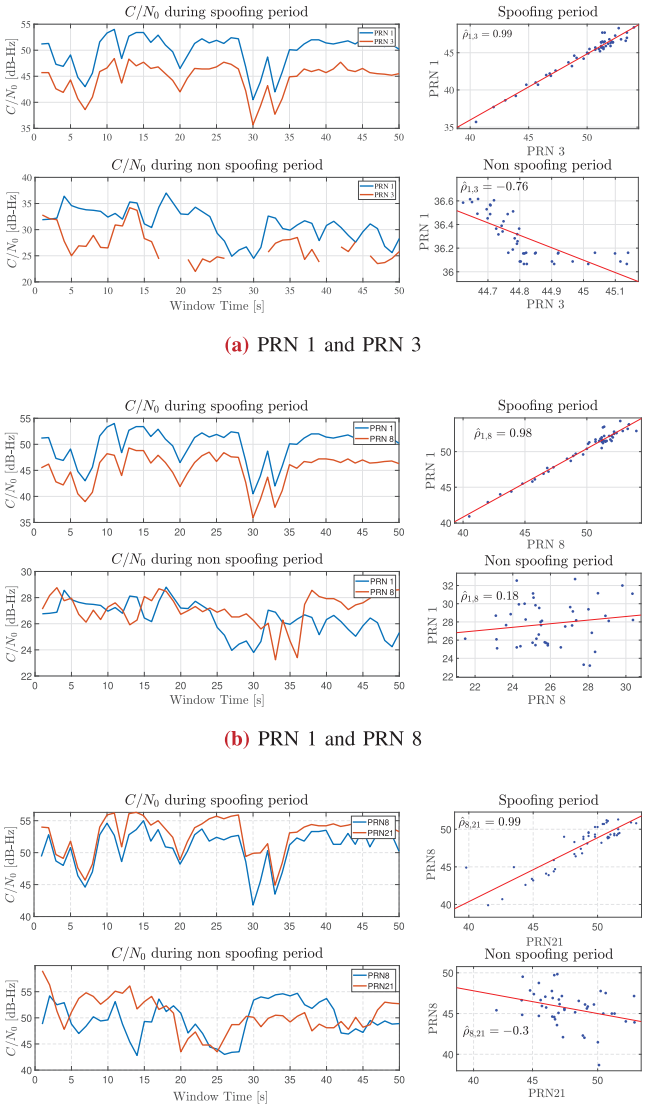
V. RESULTS

The performance of the proposed spoofing detection algorithm was assessed through the analysis of the pairwise cross-correlation between C/N_0 time series for each datasets. For the sake of clarity, in the following, GPS L1/CA signals are referred through the corresponding PRN code.

A. LINEAR CORRELATION OF C/N_0 TIME SERIES

As an example, the cross-correlation results of the different PRN's C/N_0 values are shown in Figure 12, by considering an observation window of 50 s. In detail, Figure 12a shows the C/N_0 values of GPS PRN 1 and PRN 3 for Xiaomi Redmi 8 being tested between the spoofing (top) and non spoofing (bottom) periods. As one can see, the C/N_0 of PRN 1 and PRN 3 are limited in the range 35-55 dB-Hz within the spoofing period. A remarkable difference is instead visible in non-spoofing conditions in which C/N_0 assumes values in the range 20-40 dB-Hz with a slow decreasing trend and sporadic discontinuities at low C/N_0 values. Their correlation in both spoofed and non-spoofed cases was verified by plotting the linear regression among the time series and evaluating their approximated Pearson correlation coefficients (9).

A higher linear correlation is evident in the upper plot, for which spoofing action induced a correlation value of $\hat{\rho}_{1,3} = 0.99$. Data discontinuities and dissimilar trends of the non-spoofed C/N_0 time series provide instead a poor correlation of $\hat{\rho}_{1,3} = -0.76$. Where zero values indicate undefined numeric results. The comparison of the two observation windows returns unambiguous classifications for spoofing and non-spoofing period. Similar outcomes can be observed in Figure 12b, and Figure 12c, in which 50 s observation windows return strong linear correlation of spoofed time series with correlation coefficients $\hat{\rho}_{1,8} = 0.98$ and $\hat{\rho}_{8,21} = 0.99$, respectively. Non-spoofing observations return a poor correlation coefficient in 12b while a larger value is visible in Figure 12c. In the latter, linear correlation appears stronger for higher values of C/N_0 but the overall correlation coefficient is still remarkably lower than the spoofing period. It is worth remarking that single pairwise observation cannot be considered as reliable inputs for the decision logic described in Section IV, and aggregated metrics, i.e., the average cross correlation coefficient defined in (11), must be used instead.



(a) PRN 1 and PRN 3

(b) PRN 1 and PRN 8

(c) PRN 8 and PRN 21

FIGURE 12. Comparison of C/N_0 time series for different PRNs during the spoofing and non spoofing period (left subplots), and associated linear correlation plot (right subplots). Discontinuities in the time series are mapped to null values in the correlation plots. Observation window of duration $T_n = 50$ s.

B. CORRELATION MATRIX AND AVERAGE PEARSON COEFFICIENTS

The patterns of the correlation matrices of 18 datasets have been evaluated to better understand the behaviour of the proposed indices. Figure 13, illustrates the results of Pearson correlation coefficients for the sample datasets D1, D2 and D8, showing each element of the matrix equation (10) obtained considering the C/N_0 measurements associated to each PRN. The numerical value indicating the rows and the columns of the matrix corresponding to a given PRN. Results

TABLE 3. Mean Pearson correlation coefficients and average correlation increment evaluated under $T_n = 50$ s for each dataset under non-spoofed and spoofed conditions.

Dataset ID	Device under test (Smartphone model)	Average Pearson correlation coefficient on legitimate signals ($\bar{\mu}_{\rho,NS}$)	Average Pearson correlation coefficient on spoofing signals ($\bar{\mu}_{\rho,S}$)	Average Pearson correlation coefficient increment on spoofing attack ($\delta\bar{\mu}_{\rho}$)
D1*	Samsung A30	0.069	0.977	92.94%
D2*	Samsung A32	0.042	0.988	95.75%
D3	Samsung S6	0.015	0.963	98.44%
D4	Samsung A20	0.040	0.967	95.86%
D5	Samsung Note 8	0.008	0.969	99.17%
D6	Samsung Note 20	0.144	0.974	85.21%
D7	Samsung A21s	0.104	0.969	89.27%
D8*	Samsung Note 22	0.204	0.986	79.31%
D9	Samsung A72	0.069	0.967	92.86%
D10	Xiaomi Redmi 6	0.042	0.988	95.75%
D11	Xiaomi Redmi 6 Pro	0.132	0.986	86.61%
D12	Xiaomi Redmi 8	0.001	0.970	99.89%
D13	Xiaomi Redmi 8 Pro	0.023	0.961	97.60%
D14	Xiaomi Note 9	0.078	0.967	91.93%
D15	Xiaomi Note 11	0.002	0.976	99.80%
D16	Xiaomi 12x	0.050	0.970	94.85%
D17	Huawei Y9	0.051	0.968	94.73%
D18	Honor X8	0.002	0.990	99.80%

*Reference datasets presented in Section V through the plots of the correlation matrices.

are presented using a heatmap based on a color code. To highlight the overall correlation increment due to spoofing attacks, we further defined an average Pearson correlation increment as

$$\delta\bar{\mu}_{\rho} = 100\left(\frac{\bar{\mu}_{\rho,S} - \bar{\mu}_{\rho,NS}}{\bar{\mu}_{\rho,S}}\right) \tag{17}$$

A summary of the results of all the experiments is given in Table 3. It can be seen that the Pearson correlation increment varies depending on the dataset but is always experimentally verified during the spoofing period for all the devices under test. As an example, we discuss hereafter the results of three representative datasets. As it can be observed, in the D1 dataset Figure 13b, one of the highest correlations corresponding to the pair PRN 1 and PRN 3 ($\hat{\rho}_{1,3} = 0.99$). Similarly, PRN 1 and PRN 3 were highly correlated in the D2 and D8 dataset, as shown in Figure 13d ($\hat{\rho}_{1,3} = 0.98$) and Figure 13f ($\hat{\rho}_{1,3} = 0.99$). In both the datasets, the C/N_0 time series increases the value of the Pearson coefficient of $\delta\bar{\mu}_{\rho} = 92.94\,\%$, $\delta\bar{\mu}_{\rho} = 95.75\,\%$ and $\delta\bar{\mu}_{\rho} = 79.31\,\%$ during the spoofing time period, with respect to the non-spoofed one where lower or negative correlation coefficients, $\hat{\rho}_{1,3} = -0.76$, $\hat{\rho}_{1,3} = 0.18$ and $\hat{\rho}_{1,3} = -0.30$ are observed in both Figure 13a, Figure 13c and Figure 13e populations. The characteristics of the remaining datasets are reported in Table 3, which summarizes the remarkable difference between correlation coefficients under spoofed and non-spoofed periods. The complete analysis of all the datasets confirms how a significant increment on the correlation between the time series can be observed for all the pairs of the PRNs. In light of this the mean of the Pearson coefficient defined in equation (11) is a suitable metric for the detection of a single-antenna spoofing attack based exclusively on the observation of raw GNSS measurements.

1) OBSERVATION WINDOW LENGTH AND DETECTION LATENCY

The results presented in Section V-A have been obtained for a pre-defined length of the observation window, i.e., $T_n = 50$ s. In order to identify a minimum latency for the detection of a possible spoofing attack through the proposed technique, the average Pearson correlation coefficients, $\mu_{\rho}^{(k)}$, have been evaluated for different window lengths in the range of 5 s to 400 s. Figure 14 shows the behaviour of $\mu_{\rho}^{(k)}$ and of the estimated threshold, γ, by varying the length of the observation windows for all the devices under test. In the interval between 5 s and 50 s we reported the coefficients with a step of 5 s. As it can be observed, the observation windows can be both shortened and extended without any remarkable impact on the performance of the proposed method. Shorter windows reduce the latency of the decision logic as well as the data buffering requirement but the output coefficients may turn unreliable as in the case of Samsung A32 in Scenario 2 (see, Figure 14b). Increasing T_n leads of course to an increased latency being also prone to the variability of the environment within the timespan. These

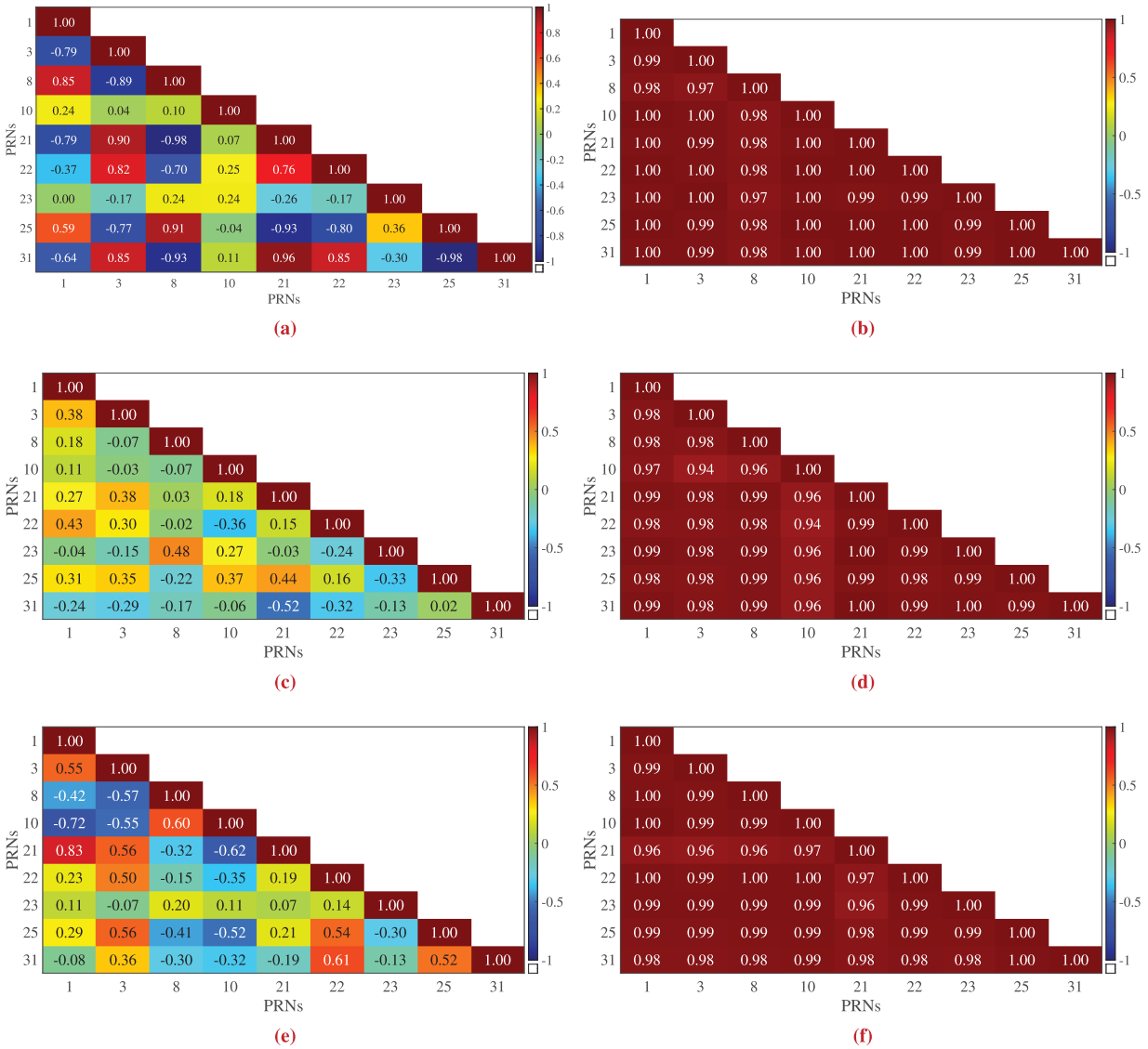
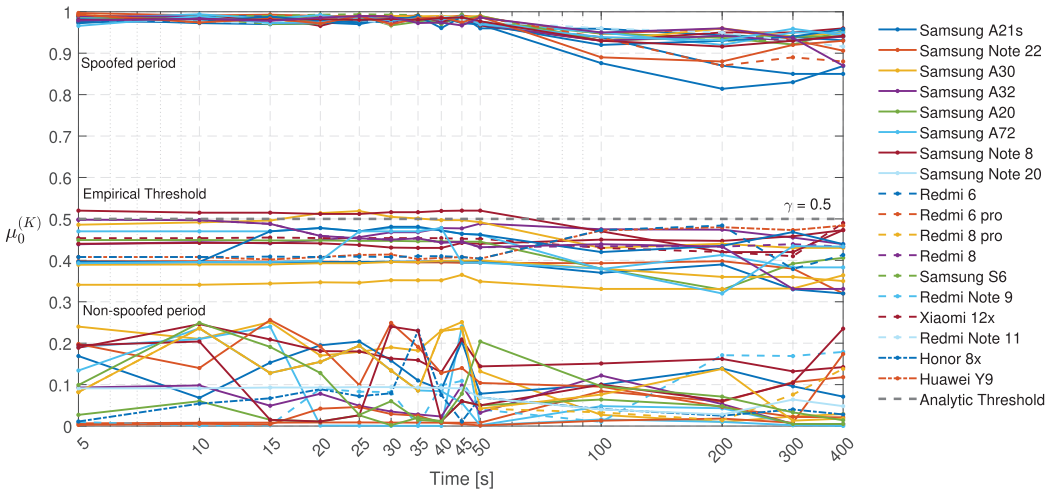


FIGURE 13. Sample heatmaps showing Pearson correlation coefficients between available PRNs according to (10). Results were computed over an observation window of duration $T_n = 50$ s during the spoofing (a, c, e) and non spoofing periods (b, d, f) for the Scenario 1.

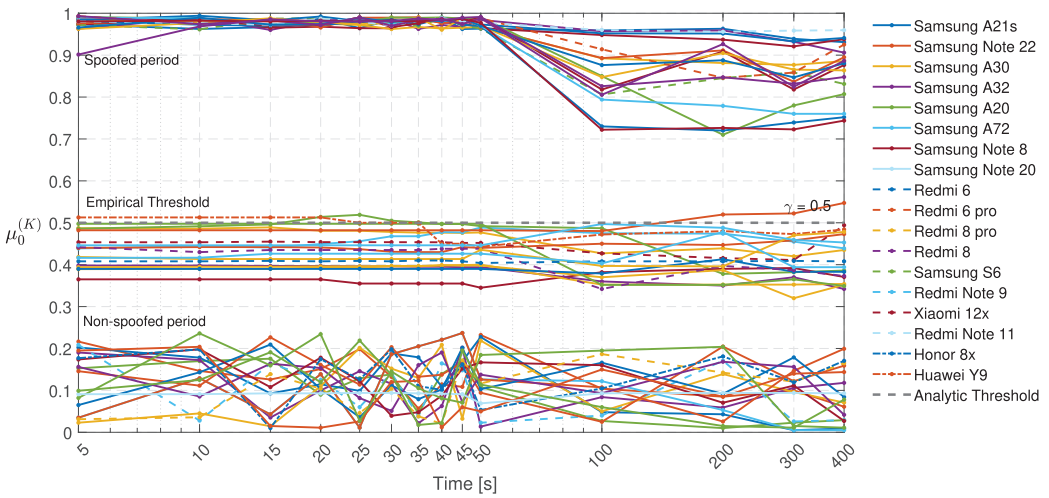
remarks justify the observation window of $T_n = 50$ s that have been hence considered in this work. According to this analysis, it is a valuable and safe trade-off between latency and reliable correlation coefficients.

VI. CONCLUSION

In this paper, starting from the analysis of the effects that single-antenna, simplistic spoofing has on the GNSS receivers embedded in a variety of Android™ smartphones, a spoofer detection technique based on the processing of raw measurement was proposed. The most relevant observation is that raw GNSS measurements, i.e., Carrier-to-noise ratio C/N_0 and pseudorange measurements, show a considerable correlation of the output time series. The estimation of the C/N_0 for spoofed signals is indeed sensitive to the spatial and temporal correlation introduced by the spoofer transmission of multiple signals over a single propagation channel. Such a peculiar feature of single-antenna spoofing attacks constitutes a considerable difference w.r.t. the received legitimate GNSS signals. It has been shown how estimating an average Pearson correlation coefficient considering all the PRNs pairs provides a suitable metric for detecting the attack. Furthermore, the analysis of such coefficients for the different devices under test showed that the performance does not depend on the target device and the observation window of the C/N_0 time series. Since the input needed for the proposed method are time-series of typical raw GNSS measurements, i.e., as C/N_0 values, future works will investigate the applicability of the proposed technique to other classes of GNSS devices, exploring different conditions of the attacks.



(a) Scenario 1



(b) Scenario 2

FIGURE 14. **Mean Pearson correlation coefficients computed for all the devices under test in Scenario 1 (top) and Scenario 2 (bottom), by varying the duration of the observation time window in 5-400 s. Analytic threshold, $\gamma = 0.5$, is compared to the empirical threshold computed upon historical data. The x-axis is shown in logarithmic scale for readibility reasons.**

REFERENCES

[1] "International Data Corporation (IDC) Market Research." Accessed: Jan. 15, 2023. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=US48936022

[2] "The EUSPA Earth Observation (EO) and Global Navigation Satellite System (GNSS) Market Report." Accessed: Jan. 15, 2023. [Online]. Available: https://www.euspa.europa.eu/2022-market-report

[3] Google LLC. "Raw GNSS Measurements." Accessed: Jan. 1, 2023. [Online]. Available: https://developer.android.com/

[4] N. Gogoi, A. Minetto, and F. Dovis, "On the cooperative ranging between Android smartphones sharing raw GNSS measurements," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5, doi: 10.1109/VTCFall.2019.8891320.

[5] M. Bakuła, M. Uradziński, and K. Krasuski, "Performance of GPS smartphone positioning with the use of P(L1) vs. P(L5) pseudo-range measurements," *Remote Sens.*, vol. 14, no. 4, p. 929, 2022, doi: 10.3390/rs14040929.

[6] G. Lachapelle and P. Gratton, "GNSS precise point positioning with Android smartphones and comparison with high performance receivers," in *Proc. IEEE Int. Conf. Signal Inf. Data Process. (ICSIDP)*, Dec. 2019, pp. 1–9, doi: 10.1109/ICSIDP47821.2019.9173062.

[7] A. Minetto, A. Nardin, and F. Dovis, "Modelling and experimental assessment of inter-personal distancing based on shared GNSS observables," *Sensors*, vol. 21, no. 8, p. 2588, 2021, doi: 10.3390/s21082588.

[8] A. Minetto, M. C. Bello, and F. Dovis, "DGNSS cooperative positioning in mobile smart devices: A proof of concept," *IEEE Trans. Veh. Technol.*, vol. 71, no. 4, pp. 3480–3494, Apr. 2022, doi: 10.1109/TVT.2022.3148538.

[9] N. Gogoi, A. Minetto, N. Linty, and F. Dovis, "A controlled-environment quality assessment of Android GNSS raw measurements," *Electronics*, vol. 8, no. 1, p. 5, 2019, doi: 10.3390/electronics8010005.

[10] F. Dovis, *GNSS Interference Threats and Countermeasures*. London, U.K.: Artech House, 2015.

[11] J. Li, X. Zhu, M. Ouyang, W. Li, Z. Chen, and Q. Fu, "GNSS spoofing jamming detection based on generative adversarial network," *IEEE Sensors J.*, vol. 21, no. 20, pp. 22823–22832, Oct. 2021, doi: 10.1109/JSEN.2021.3105404.

[12] Y. Liu, S. Li, Q. Fu, Z. Liu, and Q. Zhou, "Analysis of Kalman filter innovation-based GNSS spoofing detection method for INS/GNSS integrated navigation system," *IEEE Sensors J.*, vol. 19, no. 13, pp. 5167–5178, Jul. 2019, doi: 10.1109/JSEN.2019.2902178.

[13] A. Jafarnia-Jahromi, A. Broumandan, J. Nielsen, and G. Lachapelle, "GPS vulnerability to spoofing threats and a review of antispoofing techniques," *Int. J. Navig. Observ.*, vol. 2012, Jul. 2012, Art. no. 127072, doi: 10.1155/2012/127072.

[14] T. E. Humphreys, "Detection strategy for cryptographic GNSS anti-spoofing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 2, pp. 1073–1090, Apr. 2013, doi: 10.1109/TAES.2013.6494400.

[15] W. Zhou, Z. Lv, X. Deng, and Y. Ke, "A new induced GNSS spoofing detection method based on weighted second-order central moment," *IEEE Sensors J.*, vol. 22, no. 12, pp. 12064–12078, Jun. 2022, doi: 10.1109/JSEN.2022.3174019.

[16] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. O'Hanlon, and P. M. Kintner, "Assessing the spoofing threat: Development of a portable GPS civilian spoofer," in *Proc. 21st Int. Techn. Meeting Satellite Div. Inst. Navig. (ION GNSS)*, Jan. 2008, pp. 2314–2325.

[17] A. Broumandan, J. Nielsen, and G. Lachapelle, "GPS spoofer countermeasure effectiveness based on using signal strength noise powerand C/No observables," *Int. J. Satellite Commun. Netw.*, vol. 30, pp. 181–191, Jul. 2012.

[18] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful GPS spoofing attacks," in *Proc. 18th ACM Conf. Comput. Commun. Security*, 2011, pp. 75–86, doi: 10.1145/2046707.2046719.

[19] J. S. Warner and R. G. Johnston, "A simple demonstration that the global positioning system (GPS) is vulnerable to spoofing simple demonstration that the global positioning system (GPS) is vulnerable to spoofing," *J. Security Admin.*, vol. 25, no. 2, pp. 19–27, 2002.

[20] J. Bhatti and T. Humphreys, "Hostile control of ships via false GPS signals: Demonstration and detection," *Navigation*, vol. 64, pp. 51–66, Mar. 2017, doi: 10.1002/navi.183.

[21] J. Warner and R. Johnston, "GPS spoofing countermeasures," *Homeland Security J.*, vol. 25, p. 2, Jan. 2003.

[22] K. Songala, S. Ammana, H. Ramachandruni, and D. Achanta, "Simplistic spoofing of GPS enabled smartphone," in *Proc. IEEE Int. Women Eng. (WIE) Conf. Elect. Comput. Eng. (WIECON-ECE)*, Dec. 2020, pp. 460–463. doi: 10.1109/WIECON-ECE52138.2020.9397980.

[23] J. M. Anderson et al., "Chips-message robust authentication (Chimera) for GPS civilian signals," in *Proc. 30th Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GNSS+)*, 2017, pp. 2388–2416.

[24] I. Fernandez-Hernandez, G. Vecchione, and F. Diaz-Pulido, "Galileo authentication: A programme and policy perspective," in *Proc. 69th Int. Astronaut. Congr. IAC*, 2018, pp. 1–7.

[25] K. C. Zeng et al., "All your GPS are belong to us: Towards stealthy manipulation of road navigation systems," in *Proc. 27th USENIX Security Symp. (USENIX Security)*, Aug. 2018, pp. 1527–1544. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/zeng

[26] L. Huang and Q. Yang. "GPS Spoofing: Low-Cost GPS Emulator, DEF CON 23." Accessed: Jan. 15, 2023. [Online]. Available: https://defcon.org/html/defcon-23/dc-23-index.html

[27] S. C. Lo, Y. Chen, T. G. R. Reid, A. E. Perkins, T. Walter, and P. K. Enge, "KeyNote: The benefits of low cost accelerometers for GNSS anti-spoofing," in *Proc. ION Pac. PNT Meeting*, May 2017, pp. 775–796, doi: 10.33012/2017.15109.

[28] D. K. Lee, M. Petit, D. Miralles, S. Lo, and D. Akos, "Analysis of raw GNSS measurements derived navigation solutions from mobile devices with inertial sensors," in *Proc. 32nd Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GNSS+)*, Miami, FL, USA, 2019, pp. 3812–3831.

[29] D. Miralles, N. Levigne, D. M. Akos, J. Blanch, and S. C. Lo, "Android raw GNSS measurements as the new anti-spoofing and anti-jamming solution," in *Proc. 31st Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GNSS+)*, 2018, pp. 1–2.

[30] S. Ceccato, F. Formaggio, G. Caparra, N. Laurenti, and S. Tomasin, "Exploiting side-information for resilient GNSS positioning in mobile phones," in *Proc. IEEE/ION Position Location Navig. Symp. (PLANS)*, 2018, pp. 1515–1524.

[31] A. Rustamov, N. Gogoi, A. Minetto, and F. Dovis, "GNSS anti-spoofing defense based on cooperative positioning," in *Proc. 33rd Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GNSS+)*, Oct. 2020, pp. 3326–3337, doi: 10.33012/2020.17565.

[32] N. Spens, D.-K. Lee, and D. Akos, "An application for detecting GNSS jamming and spoofing," in *Proc. 33rd Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GNSS+)*, Sep. 2021, pp. 1981–1988, doi: 10.33012/2021.18027.

[33] D. K. Lee, N. Spens, B. Gattis, and D. Akos, "AGC on Android devices for GNSS," in *Proc. Int. Techn. Meeting Inst. Navig.*, Feb. 2021, pp. 33–41, doi: 10.33012/2021.17823.

[34] E. G. Manfredini, F. Dovis, and B. Motella, "Validation of a signal quality monitoring technique over a set of spoofed scenarios," in *Proc. 7th ESA Workshop Satellite Navig. Technol. Eur. Workshop GNSS Signals Signal Process. (NAVITEC)*, 2014, pp. 1–7, doi: 10.1109/NAVITEC.2014.7045136.

[35] D.-K. Lee et al., "Detection of GNSS spoofing using NMEA messages," in *Proc. Eur. Navig. Conf. (ENC)*, 2020, pp. 1–10, doi: 10.23919/ENC48637.2020.9317470.

[36] D. Miralles, M. Moghadam, and D. Akos, "GNSS threat monitoring and reporting with the Android raw GNSS measurements and STRIKE3," in *Proc. 32nd Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GNSS+)*, Oct. 2019, pp. 275–289, doi: 10.33012/2019.16984.

[37] M. L. Psiaki and T. E. Humphreys, "GNSS spoofing and detection," *Proc. IEEE*, vol. 104, no. 6, pp. 1258–1270, Jun. 2016.

[38] J. R. van der Merwe, X. Zubizarreta, I. Lukcin, A. Rügamer, and W. Felber, "Classification of spoofing attack types," in *Proc. Eur. Navig. Conf. (ENC)*, 2018, pp. 91–99, doi: 10.1109/EURONAV.2018.8433227.

[39] Great Scott Gadgets. "HackRF One." Accessed: Jan. 15, 2023. [Online]. Available: https://greatscottgadgets.com/hackrf/

[40] C. G. Günther, "A survey of spoofing and counter-measures," *Annu. Navig.*, vol. 61, no. 3, pp. 159–177, 2014.

[41] "Software-Defined GPS Signal Simulator." Accessed: Jan. 15, 2023. [Online]. Available: https://github.com/osqzss/gps-sdr-sim

[42] "MIT Licence." Accessed: Jan. 15, 2023. [Online]. Available: https://opensource.org/licenses/mit-license.php

[43] P. Ho. "NMEA Tools. (Version 2.7.35) [Mobile App]." 2013. [Online]. Available: https://play.google.com/store/apps/details?id=com.peterhohsy.nmeatools

[44] A. Rustamov, N. Gogoi, A. Minetto, and F. Dovis, "Assessment of the vulnerability to spoofing attacks of GNSS receivers integrated in consumer devices," in *Proc. Int. Conf. Localization GNSS (ICL-GNSS)*, 2020, pp. 1–6, doi: 10.1109/ICL-GNSS49876.2020.9115489.

[45] N. Spens, D.-K. Lee, F. Nedelkov, and D. Akos, "Detecting GNSS jamming and spoofing on Android devices," *J. Inst. Navig.*, vol. 69, no. 3, p. 537, 2022. [Online]. Available: https://navi.ion.org/content/69/3/navi.537

[46] D. Lee, N. Spens, B. Gattis, and D. Akos, "AGC on Android devices for GNSS," in *Proc. Int. Tech. Meeting Inst. Navig.*, Feb. 2021, pp. 33–41. [Online]. Available: https://doi.org/10.33012/2021.17823

[47] E. Manfredini, D. Akos, Y. Chen, S. Lo, T. Walter, and P. Enge, "Effective GPS spoofing detection utilizing metrics from commercial receivers," in *Proc. Int. Tech. Meeting Inst. Navig.*, Feb. 2018, pp. 33–41. [Online]. Available: https://doi.org/10.33012/2018.15595

[48] F. Bastide, D. Akos, C. Macabiau, and B. Roturier, "Automatic gain control (AGC) as an interference assessment tool," in *Proc. 16th Int. Tech. Meeting Satellite Div. Inst. Navig. (ION GPS/GNSS)*, Sep. 2003, pp. 2042–2053.

[49] M. Beatrice, S. Simone, M. Davide, and D. Fabio, "A method to assess robustness of GPS C/A code in presence of CW interferences," *Int. J. Navig. Observ.*, vol. 2010, Jul. 2010, Art. no. 294525, doi: 10.1155/2010/294525.

[50] D. Miralles, D. M. Akos, D.-K. Lee, A. Konovaltsev, L. Kurz, and S. Lo, "Robust satellite navigation in the Android operating system using the android raw GNSS measurements engine and location providers," in *Proc. Eur. Navig. Conf. (ENC)*, 2020, pp. 1–12, doi: 10.23919/ENC48637.2020.9317434.

[51] R. A. Fisher, "Statistical methods for research workers," in *Breakthroughs in Statistics* (Springer Series in Statistics), S. Kotz and N. L. Johnson, Eds. New York, NY, USA: Springer, 1992, pp. 66–70. [Online]. Available: https://doi.org/10.1007/978-1-4612-4380-9_6

[52] M. G. Kendall, *The Advanced Theory of Statistics. Vols. 1*. London, U.K.: Charles Griffin, 1948.

[53] R. J. Muirhead, *Aspects of Multivariate Statistical Theory*. Hoboken, NJ, USA: Wiley, 2009.

AKMAL RUSTAMOV (Student Member, IEEE) was born in Samarkand, Uzbekistan, in 1991. He received the B.Sc. and M.Sc. degrees in mechanical engineering from the Politecnico di Torino, Turin, Italy, where he is currently pursuing the Ph.D. degree with the Department of Electronics and Telecommunications. His research is focused on the implementation and resilience test of GNSS positioning systems for road applications. He has been involved in teaching the assistant part of the course "Signal Analysis and Processing" with the Polytechnic University of Turin, Tashkent, since 2020.

ALEX MINETTO (Member, IEEE) was born in Pinerolo, Italy, in 1990. He received the B.Sc. and M.Sc. degrees in telecommunications engineering from the Politecnico di Torino, Turin, Italy, and the Ph.D. degree in electrical, electronics and communications engineering in 2020. He joined the Department of Electronics and Telecommunications, Politecnico di Torino in 2022 as a Researcher and an Assistant Professor. His current research interests cover navigation signal design and processing, advanced Bayesian estimation applied to positioning and navigation technologies and applied Global navigation satellite system to space weather and space PNT.

FABIO DOVIS (Member, IEEE) was born in Bruino, Italy, in 1970. He received the M.Sc. and Ph.D. degrees from the Politecnico di Torino, Turin, Italy, in 1996 and 2000, respectively. He joined the Department of Electronics and Telecommunications, Politecnico di Torino as an Assistant Professor in 2004. Since 2021, he has been a Full Professor with the Department of Electronics and Telecommunications, where he coordinates the Navigation Signal Analysis and Simulation Research Group. He has a relevant experience in European projects in satellite navigation as well as cooperation with industries and research institutions. His research interests cover the design of GPS and Galileo receivers and advanced signal processing for interference and multipath detection and mitigation, as well as ionospheric monitoring. He serves as a member of the IEEE Aerospace and Electronics Systems Society Navigation Systems Panel.

Open Access funding provided by 'Politecnico di Torino' within the CRUI CARE Agreement