Chinese Journal of Electronics
Vol.32, No.3, May 2023

Necessary Condition for the Success of Synchronous GNSS Spoofing

WANG Yiwei, KOU Yanhong, and HUANG Zhigang

(*School of Electronics and Information Engineering, Beihang University, Beijing 100083, China*)

Abstract — A synchronous GNSS generator spoofer aims at directly taking over the tracking loops of the receiver with the lowest possible spoofing to signal ratio (SSR) without forcing it to lose lock. This paper investigates the factors that affect spoofing success and their relationships. The necessary conditions for successful spoofing are obtained by deriving the code tracking error in the presence of spoofing and analyzing the effects of SSR, spoofing synchronization errors, and receiver settings on the S-curve ambiguity and code tracking trajectory. The minimum SSRs for a successful spoofing calculated from the theoretical formulation agree with Monte Carlo simulations at digital intermediate frequency signal level within 1 dB when the spoofer pulls the code phase in the same direction as the code phase synchronization error, and the required SSRs can be much lower when pulling in the opposite direction. The maximum spoofing code phase error for a successful spoofing is tested by using TEXBAT datasets, which coincides with the theoretical results within 0.1 chip. This study reveals the mechanism of covert spoofing and can play a constructive role in the future development of spoofing and anti-spoofing methods.

Key words — Spoofing, Anti-spoofing, Synchronization, Spoofing to signal ratio (SSR), Tracking loop.

I. Introduction

With the explosive growth of the unmanned aerial vehicle (UAV) market, various flying without approval or out-of-control accidents of industrial-grade and consumer-grade UAVs have emerged, which has brought great safety risks to other aircraft, personnel and property in the increasingly complex airspace environment. The flight control of a UAV usually relies on the Global Navigation Satellite System (GNSS) information. Therefore, spoofing the target GNSS receiver is an effective way to expel the illegally flying UAV.

According to the complexity of the spoofer, GNSS spoofing can be classified as simplistic, intermediate or sophisticated [1]. A simplistic spoofing is not synchronized with the real satellite signals tracked by the target receiver, so it has to force the receiver to lose lock and capture it by using an overwhelming power advantage at the risk of being easily detected [2]. The success rate of the simplistic spoofing is decided by the probability of receiver losing lock under the suppressing jamming and the probability of acquiring and locking on the spoofing signal. On the contrary, the intermediate and sophisticated spoofing aims to keep the target receiver from losing lock by transmitting a synchronized spoofing signal at a much lower power during the intrusion, which is usually based on a good estimate of the position, velocity, and the authentic signal state of the receiver. In this sense, the intermediate and sophisticated spoofing can also be called the synchronous spoofing. The synchronous spoofer usually increases its transmission power gradually to achieve an expected spoofing to signal ratio (SSR), which is a covert process and is difficult to be detected [3]–[5]. In addition, the strategy proposed in [6] can also be classified as a synchronous type of spoofing, which intrudes the target with an intentional code phase bias at the beginning of the spoofing without gradually increasing the power.

A typical synchronous spoofing intrusion process includes scenarios such as initial synchronization, power increasing, and code phase pull-off [1], [4]. In the real world, inevitable synchronization errors, variations in the SSR, code phase and carrier phase of the spoofing signal, and the target receiver settings impact the spoofing success rate [7], especially when appropriate anti-spoofing signal processing methods are employed by the target receiver. For example, GNSS signal quality monitoring (SQM) approaches can be applied for spoofing

Manuscript Received Aug. 26, 2021; Accepted July 6, 2022. This work was supported by the National Natural Science Foundation of China (61271197).
 DOI:10.23919/cje.2021.00.307

detection [8], [9]. The spoofing signal can also be detected by monitoring the signal power variation during the intrusion [10], [11]. The hypothesis testing of the cross-correlation function (CCF) symmetry is another approach to detect spoofing signal, as the CCF can be distorted if the spoofing code phase error is obvious [12]. Overall, the spoofing signal can be detected by its influence on the receiver [13], [14].

The effectiveness of anti-spoofing algorithms depends on how the defender understands the impact of the intrusion signal. Therefore, it is necessary to analyze how the spoofing signal controls the target receiver, what are the conditions for successful spoofing, and what are the influential factors, especially when the spoofing attack employs more subtle power and less detectable pull-off strategy. A smaller SSR means that the target is less possible to be alarmed during the intrusion [11], [15]. However, the spoofer also needs to synchronize the spoofing signal with the authentic signal well, otherwise the spoofing success rate cannot be guaranteed. The SSR lower limit required for spoofing success is analyzed in [3]. However, the results only involve the situation when the pull-off direction is the same as the initial code synchronization error. The behaviors of the code tracking loop (CTL) at different intrusion stage and the influence of the CTL settings have not been fully demonstrated. As indicated in [13], spoofing and multipath have something in common. The methodology used in multipath effect analysis can be used to analyze spoofing intrusion [16], [17]. Overall, the spoofing signal will change the shape of the S-curve and introduces a lock point bias [18]. Unlike the multipath effect, however, the power of the spoofing signal can exceed that of the authentic signal, and the arrival time of the spoofing signal can be either ahead or behind of the authentic signal. Therefore, the potential influence of the spoofing signal on the S-curve is much more complicated than that of multipath.

This paper investigates the factors that determine the success of the synchronous spoofing which takes over the target receiver CTL. The main contributions are as follows: 1) We indicate that the required SSR can be much lower if the pull-off direction is opposite to the initial code synchronization error; 2) For the spoofing scenario where the pull-off is in the same direction as the code error, the necessary condition for successful spoofing is derived, which corresponds to the unambiguity of the S-curve before the pull-off stage, along with a wide pull-in range on the spoofing signal side when the pull-off causes two lock points to occur; 3) The minimum SSR required to achieve a 99% spoofing success rate and the effects of the spoofing synchronization error, the target receiver correlator spacing and CTL bandwidth, and the signal carrier-to-noise ratio is theoretically analyzed and numerically simulated. The results provide a reference for the design of spoofers and anti-spoofing receivers.

The rest of the paper is organized as follows. Section II provides the signal model and the code tracking errors introduced by spoofing. Section III analyzes the CTL behavior caused by spoofing. Section IV reveals the factors that influence spoofing success. In Section V, the minimum SSR for a 99% spoofing success rate under different synchronization errors, receiver settings, and carrier-to-noise ratios (CNRs) are presented by simulation. A real data experiment with GPS L1 C/A signal from TEXBAT datasets is also conducted [19], [20], which verifies the theoretical results. The conclusions and discussion are given in Section IV.

II. Model of Code Tracking Error with Spoofing

The spoofing signal will cause code tracking error to the target receiver. Modelling this tracking error is essential to reveal the impact of the spoofing signal on the CTL. This section establishes the model of the received signal with spoofing, derives the expressions of the CTL discriminator output and the code tracking error, and analyzes the error envelope characteristics and the S-curve ambiguity that affects the spoofing success.

1. CTL discriminator output

The incoming signal is given as follows by a sum of authentic signal, the spoofing signal, and the noise.

$$\begin{aligned} r(t) =& s_a(t) + s_s(t) + n(t) \\ =& A \times \left\{ \begin{array}{l} C(t-\tau_0)D(t-\tau_0)\cos(2\pi f_0 t + \theta_0) \\ +\alpha_s C(t-\tau_0-\tau_s)D_s(t-\tau_0-\tau_s) \\ \times \cos\left[2\pi(f_0+f_s)t + \theta_0 + \theta_s\right] \end{array} \right\} \\ &+ n(t) \end{aligned} \tag{1}$$

where $C(t)$ is the pseudo random noise (PRN) code and $D(t)$ is the navigation data. The amplitude, code delay, carrier frequency, and phase of the authentic signal are denoted as A, τ_0, f_0, and θ_0. The amplitude ratio, code phase offset, carrier frequency offset, and carrier phase offset of the spoofing signal relative to the authentic signal are represented as A_s, τ_s, f_s, and θ_s. $n(t)$ is the additive noise which follows Gaussian distribution. For simplicity, $A = 1$ is assumed and the impact of navigation data is not considered. The SSR is defined as SSR=$20\log_{10}(\alpha_s)$ with the unit dB. Different from the multipath signal, $\alpha_s \geq 1$ usually holds when the spoofing signal reaches its maximum power, and τ_s can be either positive or negative in real scenery. To discriminate the code phase error, the local replicas with different code phase shift $\delta = -d, 0, d$ need to be generated to correl-

ated with the signal. The complex CCF after the carrier demodulation and code de-spreading can be expressed as

$$\begin{aligned}&\mathrm{CCF}(\varepsilon+\delta)\\&=\frac{1}{T_{coh}}\int_{T_0}^{T_0+T_{coh}}\left[\begin{array}{l}r(t)\mathrm{e}^{\mathrm{j}(2\pi\widehat{f}_0 t+\widehat{\theta}_0)}\\\times C(t-\widehat{\tau}_0-\varepsilon-\delta)\end{array}\right]\mathrm{d}\varepsilon\\&=R(\varepsilon+\delta)\mathrm{sinc}(f_e T_{coh})\mathrm{e}^{\mathrm{j}\theta_e}+\alpha R(\varepsilon+\delta-\tau_s)\\&\quad\times\mathrm{sinc}\left[(f_e-f_s)T_{coh}\right]\mathrm{e}^{\mathrm{j}(\theta_e-\theta_s)}+N_x\\&=I_x+\mathrm{j}Q_x\end{aligned}\tag{2}$$

where the real and imaginary parts of CCF with different delays are donated as I_x and Q_x. The starting time is denoted as T_0, and the integration period is denoted as T_{coh}; $\widehat{\tau}_0$, $\widehat{f}_0$, and $\widehat{\theta}_0$ are the local estimates of the code phase, carrier frequency, and carrier phase; $\varepsilon=\tau_e=\widehat{\tau}_0-\tau_0$, $f_e=\widehat{f}_0-f_0$, and $\theta_e=\widehat{\theta}_0-\theta_0$ are the estimation errors; N_x is the noise term. If $f_s\gg 0$, the carrier tracking loop can get into an unsteady state by processing two signals with different frequencies simultaneously. Hence, the frequency synchronization error is assumed to be small enough, and the energy loss caused by sinc function in (2) is negligible. Therefore, the outputs of these branches can be expressed as

$$\begin{aligned}\left[\begin{array}{l}I_E\\I_P\\I_L\end{array}\right]=&\left[\begin{array}{ll}R(\varepsilon-d)&\alpha_s R(\varepsilon-\tau_s-d)\\R(\varepsilon)&\alpha_s R(\varepsilon-\tau_s)\\R(\varepsilon+d)&\alpha_s R(\varepsilon-\tau_s+d)\end{array}\right]\\&\times\left[\begin{array}{l}\cos(\theta_e)\\\cos(\theta_e-\theta_s)\end{array}\right]+\left[\begin{array}{l}N_{IE}\\N_{IP}\\N_{IL}\end{array}\right]\end{aligned}\tag{3}$$

$$\begin{aligned}\left[\begin{array}{l}Q_E\\Q_P\\Q_L\end{array}\right]=&\left[\begin{array}{ll}R(\varepsilon-d)&\alpha_s R(\varepsilon-\tau_s-d)\\R(\varepsilon)&\alpha_s R(\varepsilon-\tau_s)\\R(\varepsilon+d)&\alpha_s R(\varepsilon-\tau_s+d)\end{array}\right]\\&\times\left[\begin{array}{l}\sin(\theta_e)\\\sin(\theta_e-\theta_s)\end{array}\right]+\left[\begin{array}{l}N_{QE}\\N_{QP}\\N_{QL}\end{array}\right]\end{aligned}\tag{4}$$

where N_{IE}, N_{IP}, N_{IL}, N_{QE}, N_{QP} and N_{QL} are the noises after coherent integration. The noise terms are omitted for simplicity in the following theoretical analysis. In this way, the S-curve can be obtained from the CCF as

$$S(\varepsilon)=\frac{e}{\lambda}=\frac{\left(I_E^2+Q_E^2\right)^k-\left(I_L^2+Q_L^2\right)^k}{\left(I_E^2+Q_E^2\right)^k+\left(I_L^2+Q_L^2\right)^k}\tag{5}$$

where e is the S-curve with the normalization factor λ. k determines the discriminator type with $k=0.5$ and $k=1$ corresponding to the normalized non-coherent envelope discriminator and normalized non-coherent power discriminator. The output of the discriminator with spoofing signal is given by

$$\begin{aligned}e=&\left(I_E^2+Q_E^2\right)^k-\left(I_L^2+Q_L^2\right)^k\\=&\left[\begin{array}{l}R^2(\varepsilon-d)+\alpha_s^2R^2(\varepsilon-\tau_s-d)\\+2\alpha_s\cos(\theta_s)R(\varepsilon-d)R(\varepsilon-\tau_s-d)\end{array}\right]^k\\&-\left[\begin{array}{l}R^2(\varepsilon+d)+\alpha_s^2R^2(\varepsilon-\tau_s+d)\\+2\alpha_s\cos(\theta_s)R(\varepsilon+d)R(\varepsilon-\tau_s+d)\end{array}\right]^k\end{aligned}\tag{6}$$

Let $e=0$, thus ε can be estimated by solving equation (6). The solution defines the zero-crossing point bias of the S-curve, which drives the CTL to lock on the incoming signal [21].

$$\widehat{\varepsilon}=\arg\min_{\varepsilon}|e|\tag{7}$$

The solution of (7) is the same when $k=0.5$ and $k=1$. As demonstrated in Fig.1, typically a synchronous spoofing intrusion process can be divided into several stages. After the spoofing signal gets aligned with the authentic signal, the spoofing signal begins to increase its power to the expected SSR from a low SSR, as shown in Fig.1(a). Then, the spoofing signal might keep relatively static with the authentic signal for a while, as shown in Fig.1(b). Next, the spoofing signal begin to lift off the CTL from the authentic signal, as shown in Fig.1(c). Finally, the CTL may be captured by the spoofing signal, and the spoofer can provide false navigation message to the victim [22], as shown in Fig.1(d).

During the intrusion, the shapes of the CCF and the S-curve change continuously, especially when the spoofing signal has a significant synchronization error with the authentic signal. The spoofing signal may even introduce an additional lock point and a boundary to the S-curve. Each lock point of the S-curve is a steady-state tracking point for CTL. The offset of the zero-crossing point caused by the pollution of the spoofing signal can be seen as the code tracking error. A boundary is also a zero-crossing but with the following features: 1) It cannot provide a steady-state tracking for the CTL, and the CTL will be pulled away when getting close to it; 2) The CTL can hardly exceed it when getting close to it, unless the discriminator rushes towards it at a very high speed of code phase shift due to signal dynamics, interference, or noises; 3) It divides the pull-in area of the S-curve into two parts, which are governed by the lock point on the authentic signal side and on the spoofing signal side, respectively.

2. Code tracking error

The code tracking bias ε can be solved referring to the multipath analysis method in references [16] and [18]. However, since the synchronization error of the spoofing signal may be positive or negative, and the amplitude ratio may be larger or smaller than 1, the cases of the S-curve zero-crossings are more complicated under

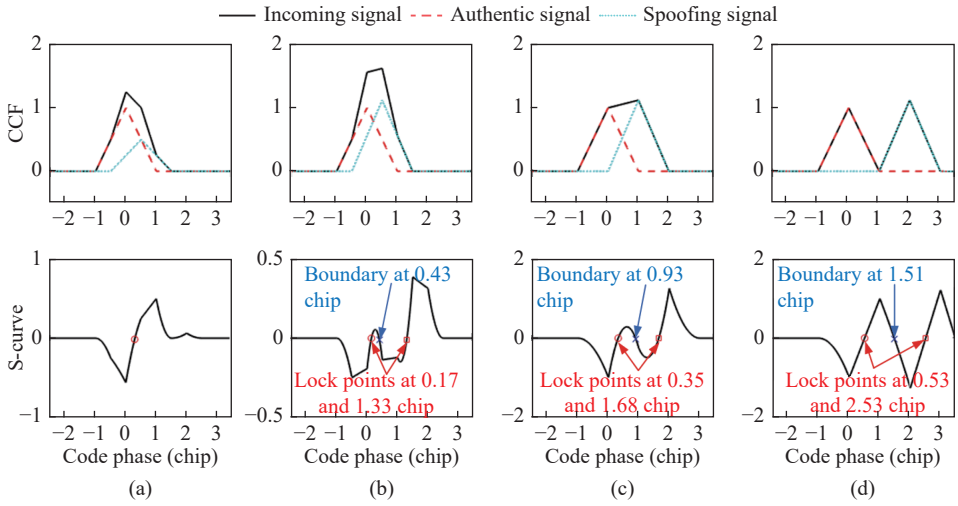

Fig. 1. CCF and S-curve under synchronous GNSS spoofing intrusion with expected SSR = 1 dB and an initial code phase synchronization error = 0.5 chip: (a) Power increasing when SSR reaches −6 dB; (b) Keeping static with SSR = 1 dB; (c) Pull-off when code phase reaches 1 chip; (d) Capturing and controlling with spoofing code phase = 2 chip.

spoofing than under multipath. Taking the BPSK-modulated GNSS signal as an example, the value of the early (E) or late (L) correlator of the authentic signal is a piece-wise function of ε and d, as listed in Table 1.

Table 1. Correlator outputs of E and L correlators

Range of ε	Early correlator	Late correlator
$1-d \le \varepsilon < 1+d$	$1-\varepsilon+d$	0
$d \le \varepsilon < 1-d$	$1-\varepsilon+d$	$1-\varepsilon-d$
$-d \le \varepsilon < d$	$1+\varepsilon-d$	$1-\varepsilon-d$
$d-1 \le \varepsilon < -d$	$1+\varepsilon-d$	$1+\varepsilon+d$
$-d-1 \le \varepsilon < d-1$	0	$1+\varepsilon+d$

The correlation of the spoofing signal is also a piece-wise function of $\varepsilon - \tau_s$ with a form similar to Table 1. Hence, as classified in Fig.2, when the spoofing signal is mixed with the authentic signal, the combination of ε and $\varepsilon - \tau_s$ impacting the S-curve can be divided into 25 cases.

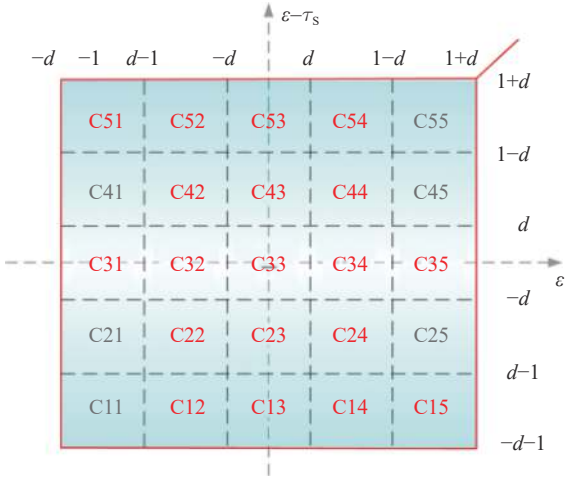

Fig. 2. Potential cases of the spoofing code phase synchronization error τ_s and the CTL estimation error ε.

By substituting the early and late branches' values defined in (3) and (4) into (6) and (7), the equation can be written in a unified form as

$$a_{Ck_1k_2}\varepsilon^2_{Ck_1k_2} + b_{Ck_1k_2}\varepsilon_{Ck_1k_2} + c_{Ck_1k_2} = 0, \quad k_1, k_2 = 1, 2, \ldots, 5 \tag{8}$$

If $a_{Ck_1k_2} = 0$, there is only one solution expressed as follows corresponding to this linear equation.

$$\varepsilon_{Ck_1k_2} = -\frac{c_{Ck_1k_2}}{b_{Ck_1k_2}} \tag{9}$$

Otherwise, there can be two solutions expressed as follows by solving this quadratic equation.

$$\varepsilon_{Ck_1k_2} = \frac{-b_{Ck_1k_2} \pm \sqrt{\Delta_{Ck_1k_2}}}{2a_{Ck_1k_2}}, \quad \Delta_{Ck_1k_2} = b^2_{Ck_1k_2} - 4a_{Ck_1k_2}c_{Ck_1k_2} > 0 \tag{10}$$

The solution is denoted as $Ck_1k_2^+$ if "+" is selected in (10). Given a set of $d, \alpha, \theta_s, \tau_s$, the possible cases may not be unique, and there may be 1 or 3 effective solutions of ε. Each effective solution corresponds to a zero-crossing point of the S-curve, and each zero-crossing point with a positive slope is a potential CTL lock point, such a condition can translate to $S(\varepsilon-\sigma) < 0 \cap S(\varepsilon+\sigma) > 0$, where $\sigma > 0$ but approaches to 0. If only one solution is found to be effective, it must be the unique CTL lock point. Otherwise, there will be 3 effective solutions with 2 potential lock points and 1 boundary point between the two pull-in ranges corresponding to these 2 lock points. A boundary is a zero-crossing point with a negative slope, that is $S(\varepsilon-\sigma) > 0 \cap S(\varepsilon+\sigma) < 0$.

Here, we try to check all possible spoofing signal states and identify the meaning of each effective solu-

tion with the assistance of computer enumeration. Such enumeration has been used to obtain the closed-form expression of the multipath error envelope (MEE) [16], [18], [23]. However, the analysis is much more complicated for the spoofing cases, as the 3 possible solutions with 2 lock points and 1 boundary imply the ambiguity of the S-curve and thus the diversity of the CTL behavior.

To traverse all possible states of parameter combinations, the spoofing signal parameters SSR, τ_s and θ_s are set to scan within the ranges of $[-20, 20]$ dB, $[-2, 2]$ chip, and $[-\pi, \pi]$ rad, with a step of 0.01 dB, 0.01 chip, and 0.01 rad, respectively. The EML spacing is also set to scan from 0.05 to 0.5 chip with a step of 0.05 chip.

All the solutions with SSR ≥ 0 dB can be classified after the enumeration as: 1) C11, C21, C25, C41, C45, and C55 do not have effective solution among the 25 potential cases, whereas every other case yields one or two effective solutions; 2) The solutions denoted as $C31^+$, $C32^+$, C33, $C34^+$, and $C35^+$ correspond to the lock point which always appear on the spoofing signal side, and $C12^-$, $C13^+$, $C54^-$, and $C53^+$ correspond to another lock point on the authentic signal side; 3) When two lock points appear simultaneously, there must be a boundary dividing the two pull-in ranges, and $C13^-$, $C14^-$, $C15^-$, C22, $C23^-$, C24, $C32^-$, $C34^-$, $C35^-$, C44, $C43^-$, C42, $C52^-$, and $C51^-$ are found to correspond to this boundary. The solutions of all the cases with $\varepsilon > d-1$ and $\varepsilon - \tau_s \leq d$ are listed in the Appendix A, while C31, C42, C43, C44, and C51, C52, C53, C54 are the symmetrical cases of C35, C24, C23, C22, and C15, C14, C13, C12. For example, the solution C51 can be obtained according to C15 by simply replacing τ_s with $-\tau_s$ in the solution of C15.

The spoofing success is related to the ambiguity of the CCF, and the condition of this ambiguity is that one of $C12^-$, $C13^+$, $C54^-$, and $C53^+$ is an effective solution, which can be expressed as formula (11).

$$\{\kappa_{C12^-} \cup \kappa_{C13^+} \cup \kappa_{C54^-} \cup \kappa_{C53^+}\} \tag{11}$$

For example, the condition $\kappa_{C12^-} = 1$ translates to the two conditions shown in formula (12): i) The discriminant function Δ_{C12} is larger than 0; ii) The solution falls within its supposed range to be effective.

$$\kappa_{C12^-} = \begin{cases} 1, & \Delta_{C12} = b_{C12}^2 - 4a_{C12}c_{C12} > 0, \\ & d-1 \leq \varepsilon_{C12^-} < -d, \\ & -1-d \leq \varepsilon_{C12^-} - \tau_s < d-1 \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

If condition (11) is satisfied in the first two stages, there will be 3 effective solutions with 2 CTL lock points and 1 boundary, implying that the SSR is not high enough to compensate for the large synchronization errors of the spoofing signal, and the spoofing will probably fail. Otherwise, there is only one lock point on the S-curve, meaning that the synchronization error in the initial stage is not large and the spoofing has a chance to succeed. Nevertheless, with the code phase divergence between two signals getting larger during the pull-off stage, condition (11) will be satisfied with another lock point appearing on the authentic signal side, which also represents the separation of the two signals.

Similar to the MEE of the one-path specular multipath model, the spoofing error envelopes (SEE) can envelope the CTL tracking errors caused by spoofing. However, the SEE analysis is more complicated as the spoofing may introduce multiple zero-crossing points. Fig.3 depicts the SEEs of the two potential lock points and the boundary, with a spoofing code phase offset scanning from -1.5 chip to 1.5 chip, an SSR of 3 dB, a spoofing carrier phase offset $\theta_s = 0$ rad (solid line) and π rad (dashed line) for the upper and lower envelopes, an EML spacing of 0.5 chip, 0.3 chip, and 0.2 chip respectively. The corresponding case numbers are illustrated in Fig.4. It can be seen that the tracking bias ε is associated with the CTL state, which has the ambiguity un-

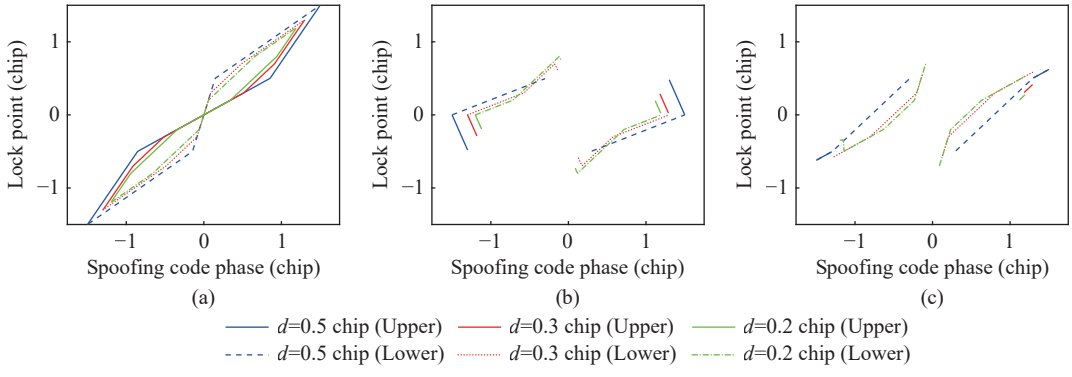

Fig. 3. SEEs with an SSR = 3 dB; a spoofing carrier phase offset $\theta_s = 0$ rad (solid line) and π rad (dashed line) for the upper and lower envelopes; an EML spacing d = 0.5, 0.3, and 0.2 chip. (a) The lock point on the spoofing signal side; (b) The boundary; (c) The lock point on the authentic signal side.

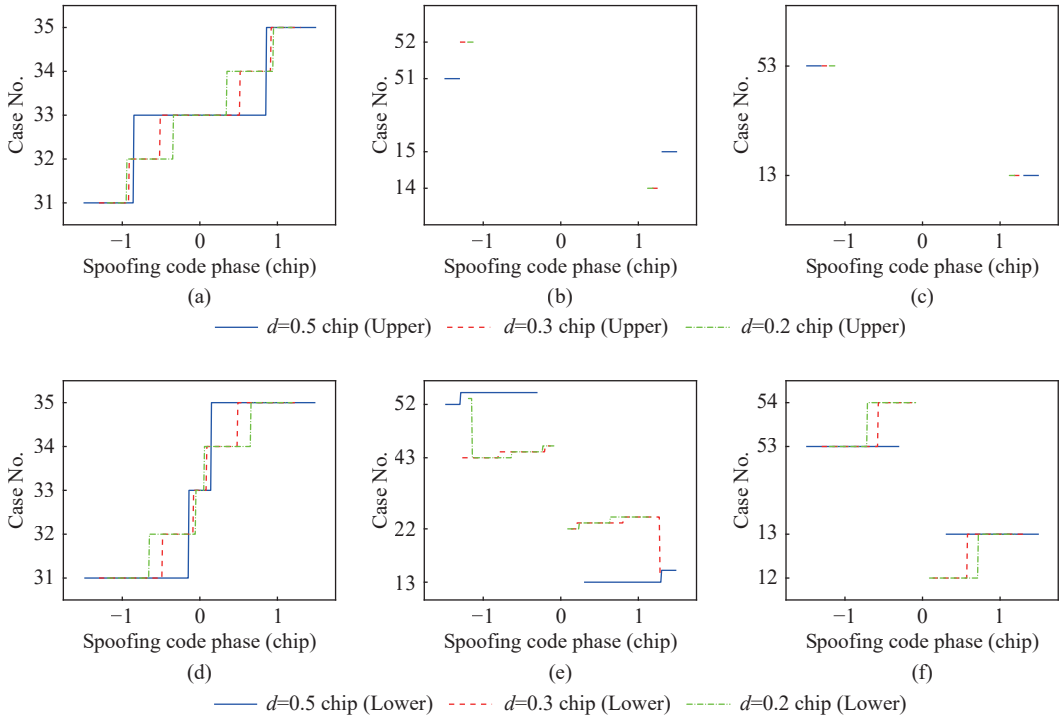

Fig. 4. Case numbers corresponding to Fig.3(a) and (d) Case numbers for the lock point on the spoofing signal side (Fig.3(a) upper and lower, respectively); (b) and (e) Case numbers for the boundary (Fig.3(b) upper and lower, respectively); (c) and (f) Case numbers for the lock point on the authentic signal side (Fig.3(c) upper and lower, respectively).

der certain sets of parameters.

Since different modulations and bandwidths are adopted by different GNSS signals, we use chip as the unit of the code phase. Under the condition of same other factors, the code phase synchronization error in meters for a BPSK(10) signal should be 1/10 as that for a BPSK(1) signal. When it comes to the binary offset carrier (BOC) modulation, however, the situation is slightly different. For most unambiguous processing techniques in the target receiver, the analysis method in this paper is still applicable as the effects of the correlation side-peaks have been eliminated. Occasionally, if the receiver neglects the false lock risk caused by the correlation side-peaks and track the main-peak directly, the code phase synchronization error must be within the range of the main-peak (e.g., $-1/3 < \tau_s < 1/3$ chip for BOC(1,1)), otherwise the spoofing may fail. Nevertheless, the SEE models of a BOC signal can be analyzed in a same way as a BPSK signal with a same CCF main-peak width. Limited to the length, we only focus on BPSK in the rest of the paper.

III. Impact of Spoofing on Code Tracking Trajectory

The overall trend of the code tracking trajectory can be predicted by analyzing the S-curve and its changes. As indicated in [6], if the code phase synchronization error of the spoofing signal satisfies $\tau_s \geq 0$ with a positive pull-off direction, the behavior of the CTL is different from the situation with $\tau_s < 0$. Hence, we discuss the impacts in these two situations respectively.

1. Pull-off in the same direction as the code synchronization error

Assuming the initial code phase synchronization error $\tau_s \geq 0$ and the subsequent pull-off direction of the spoofing signal is positive, the flow diagram of the spoofing intrusion process and the resulting CTL behavior can be described by Fig.5.

The symmetric situation with $\tau_s < 0$ and a negative pull-off direction will have the same consequences. When the spoofing signal begins to increase its power, the CTL adjusts itself to keep up with the moving lock point. If the synchronization error comes with an insufficient SSR, the condition defined in (11) is satisfied. In this circumstance, the CTL is handled by the lock point on the authentic signal side and can hardly cross the boundary of the two pull-in ranges in the following stages, and thus the spoofing tends to fail, unless there is a sudden dynamic change or interference in the following stages of spoofing. If condition (11) is not satisfied, the spoofing signal can take control of the CTL after power-increasing. The spoofing signal begins to adjust its code phase offset to leave away from the authentic signal. If condition (11) is never satisfied during the pull-off stage until two signals completely separate, the spoofing tends to succeed. However, condition (11)

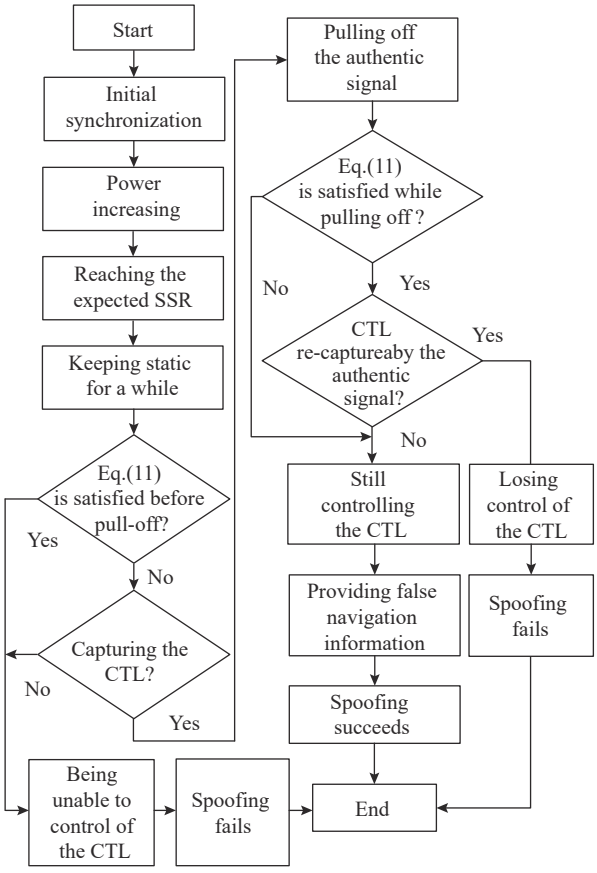

Fig. 5. Spoofing intrusion process with positive synchronization error and positive pull-off direction.

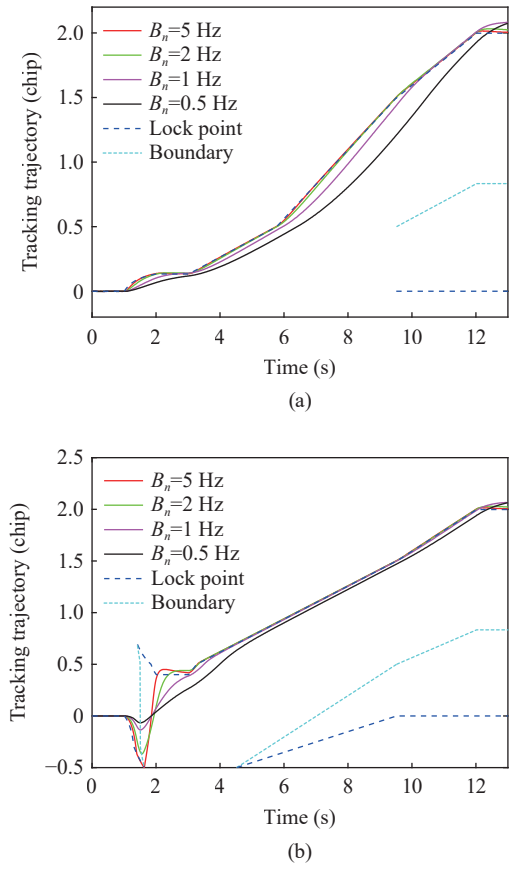

Fig. 6. Code tracking trajectory for the initial carrier phase error (a) 0 rad and (b) π rad, with EML spacing = 0.5 chip, SSR = 6 dB, initial code phase synchronization error = 0.2 chip.

is possible to be satisfied during the pull-off stage, at which moment 2 lock points appear simultaneously on the S-curve. This is a critical moment for the spoofing, as the CTL will be either controlled by the spoofing signal or re-captured by the authentic signal with an extending pull-in range, which leads to success or failure of the spoofing respectively.

Simulations based on our software signal simulator and receiver at the digital intermediate frequency (IF) signal level are presented to exemplify the above analysis. The GPS L1 PRN3 signal is simulated with C/N_0=45 dB-Hz. The receiver CTL is set with an EML spacing $d = 0.5$ chip, an integration period $T_{coh} = 5$ ms, and loop bandwidths 0.5, 1, 2, and 5 Hz. The spoofing process is illustrated in Fig.6, with an initial code phase synchronization error of τ_s =0.2 chip.

The spoofing signal starts to increase its power at 1 s and reaches an SSR of 6 dB at the end of 2 s. The pull-off stage begins at 3 s and finishes at 12 s. The changing phase offset of locally generated code relative to the incoming code is denoted as the CTL "tracking trajectory" and plotted. For the zero carrier phase synchronization error case (θ_s =0 rad) in Fig.6 (a), the overall trajectory coincides well with the unique lock point on the spoofing signal side, and the spoofing finally succeeds as the discriminator is misled to 2 chips away from the authentic code phase. However, when the CTL bandwidth gets narrower, the pull-off becomes more hysteretic and difficult due to the reduced CTL dynamic tolerance. During the pull-off stage, condition (11) is never satisfied until 9.5 s when two signals completely separate with each other. In this condition, the discriminator cannot be re-captured by the authentic signal and spoofing success can be well ensured. For the $\theta_s = \pi$ rad case in Fig.6 (b), during the power increasing stage, the tracking trajectory first goes to the negative direction, and then turns back to the original position meaning that the synchronization errors has been compensated by SSR as the S-curve is unambiguous before the pull-off stage, and the spoofing can eventually succeed with a high possibility. When the expected SSR degrades to 4 dB, the results are demonstrated in Fig.7.

The trajectories in Fig.7 (a) show that the condition (11) is satisfied at 6 s, which is the critical moment when 2 lock points appear on the S-curve. The distance between 2 lock points in this moment is 0.25 chip only, and the CTL is re-captured by the authentic

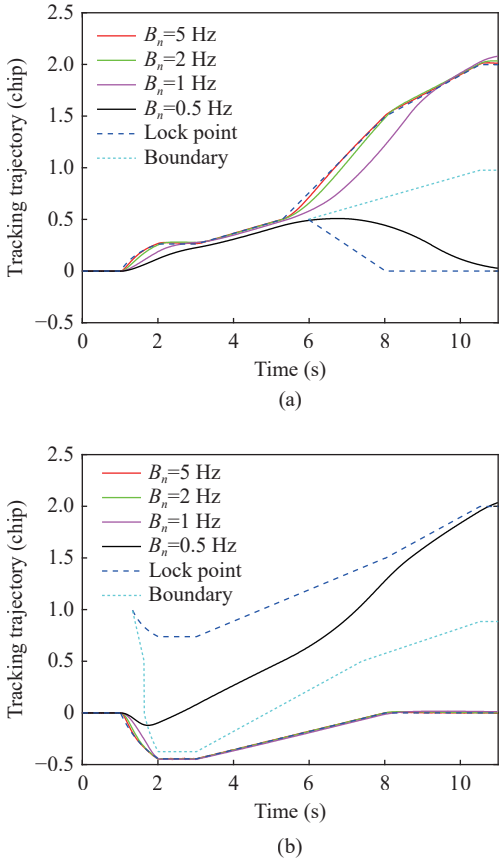

Fig. 7. Code tracking trajectory for the initial carrier phase error (a) 0 rad and (b) π rad, with the EML spacing = 0.5 chip, SSR = 4 dB, initial code phase synchronization error = 0.2 chip.

signal if $B_n = 0.5$ Hz, whilst the CTL is still be controlled by the spoofing signal with 1, 2, and 5 Hz. Thus, the expected SSR = 4 dB cannot fully ensure spoofing success in this condition. As demonstrated in Fig.7(b), when the carrier phase synchronization error is $\theta_s = \pi$ rad, the S-curve is ambiguous with 2 lock points after the power increasing stage. When it comes to the pull-off stage, the CTL is still handled by the authentic signal, and the spoofing is hard to succeed. It can be seen that the spoofing with CTL bandwidth $B_n = 1$, 2, or 5 Hz fails in the end. Only when $B_n = 0.5$ Hz, the spoofing can succeed by a fluke. This is because the CTL bandwidth is so narrow that the CTL cannot keep up with the moving lock point on the authentic signal side in the power increasing stage, the CTL can be captured by the spoofing signal in the subsequent pull-off stage. However, this phenomenon is so particular that it only occurs when the power increasing speed is high and the target CTL dynamic tolerance is not enough.

2. Pull-off in the opposite direction as the code synchronization error

The flow diagram with a negative code phase synchronization error $\tau_s < 0$ and a positive pull-off direction is given in Fig.8.

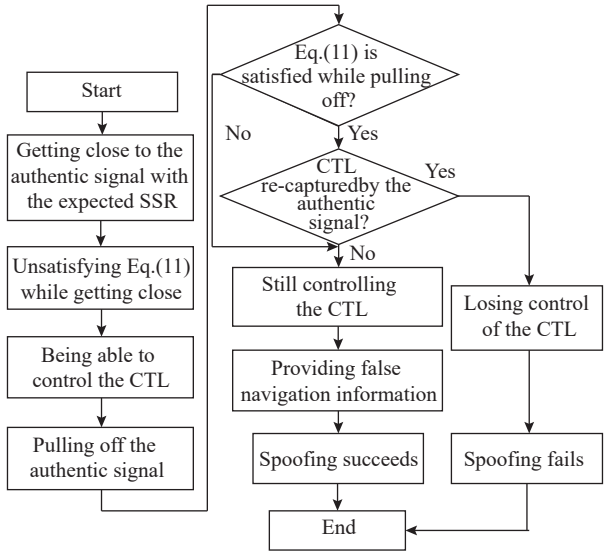

Fig. 8. Spoofing intrusion process with negative synchronization error and positive pull-off direction.

In this situation, the CTL will definitely be controlled by the spoofing signal when it gets close to the authentic signal, thus the ambiguity problem is not related to the spoofing success rate. Though the CTL still suffers the risk of being re-captured by the authentic signal when condition (11) is satisfied during the pull-off stage, the CTL has been handled by the spoofing signal for a longer time compared with its counterpart with $\tau_s \geq 0$. Thus, it is more tightly controlled by the spoofing signal, and the spoofing is more likely to succeed with $\tau_s < 0$. The examples are demonstrated in Fig.9. The SSR is set to 4 dB and the code phase synchronization error is $\tau_s = -0.5$ chip. It can be seen that all the intrusions succeed in this simulation.

IV. Factors Influencing Spoofing

Based on the above analysis, we can deduce that the relationship among SSR, synchronization error, pull-off direction, and target receiver CTL settings play an important role in spoofing success. The factors that impact the spoofing success rate can be summarized as follows.

1. S-curve ambiguity and width of the pull-in range on the spoofing signal side

Assuming $\tau_s \geq 0$, the most essential factor influencing spoofing success is whether the S-curve is ambiguous at the end of the power increasing stage, which can be expressed as the opposition condition of (11)

$$\{\bar{\kappa}_{\mathrm{C12}-} \cap \bar{\kappa}_{\mathrm{C13}+} \cap \bar{\kappa}_{\mathrm{C54}-} \cap \bar{\kappa}_{\mathrm{C53}+}\} \tag{13}$$

For the case that the subsequent pull-off stage is in the same direction as the code synchronization error, such as pull-off in the positive direction when $\tau_s \geq 0$ or

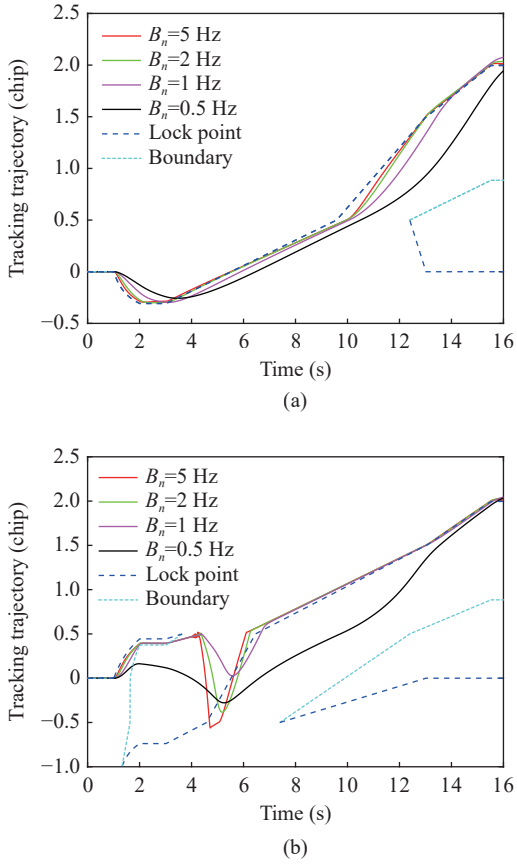

Fig. 9. Code tracking trajectory for the initial carrier phase error (a) 0 rad and (b) π rad, with the EML spacing = 0.5 chip, SSR = 4 dB, initial code phase synchronization error = −0.5 chip.

pulling in the negative direction when $\tau_s < 0$, condition (13) is a necessary condition for successful spoofing. On the contrary, for the case that the subsequent pull-off direction is opposite to the direction of the code synchronization error, such as pull-off in the positive direction when $\tau_s < 0$ or in the negative direction when $\tau_s \geq 0$, the ambiguity problem does not influence spoofing success anymore, and the spoofing success rate will be significantly improved.

According to the model established in Section II, when the S-curve becomes ambiguous with two lock points, the widths of the pull-in ranges on the spoofing signal side and the authentic signal side are plotted in Fig.10 and Fig.11, respectively. If the S-curve has only one lock point, the corresponding position is filled with blank.

The dividing line between blank area and the filled area represents the moment when 2 lock points begin to appear on the S-curve simultaneously. By observing the change of the dividing line, it can be seen that the increase of the code phase synchronization error makes the ambiguity problem more likely to occur. For example, if the SSR is 2 dB and $\theta_s = 0$, the ambiguity problem occurs when the code phase synchronization error $\tau_s \geq 1.25$ chip. That is to say, the spoofing can succeed when $\tau_s < 1.25$ chip as the S-curve keeps unambiguous before the pull-off stage. In addition, the increase of the carrier phase synchronization error also makes the ambiguity problem easier to happen. For example, if the SSR is 2 dB and $\theta_s = \pi$ rad, the ambiguity problem occurs when $\tau_s = 0.23$ chip, which makes a huge difference in comparison with the case when $\theta_s = 0$ rad. Moreover, a stronger SSR can compensate for synchronization errors to reduce the probability of an ambiguous S-curve. For example, if the SSR increases to 4 dB under $\theta_s = 0$ rad, the code phase synchronization error that an unambiguous S-curve can withstand is 1.41 chip.

The spoofing can still fail if the discriminator is recaptured by the authentic signal in the pull-off stage even though condition (13) is satisfied during the power increasing stage. For example, even though Fig.7(a) may succeed with a reduced SSR compared with Fig.6(a), the spoofing success rate will be significantly reduced when there is interference or sudden change, or the target CTL dynamic tolerance does not meet expectation. Furthermore, when condition (11) is just satisfied, the width of the pull-in range on the spoofing signal side dominates this distance, whereas the width on the authentic signal side approaches 0 chip at that moment. For example, it can be seen from Fig.10 (a) and Fig.11(a) that, if SSR=0 dB, $\theta_s = 0$ rad, and the spoofing code phase synchronization error $\tau_s < 1$ chip, the ambiguity phenomenon does not appear before the pull-off stage. When condition (11) is just satisfied during the pull-off stage at $\tau_s = 1$ chip, the widths of the pull-in ranges between the two lock points approach 0 chip, and the spoofing signal is easily re-captured by the authentic signal. The values are 0.51 chip and 0.82 chip respectively when SSR= 2 dB and 4 dB in Fig.10(a), which ensures that the discriminator is difficult to be re-captured. This indicates that a stronger SSR can reduce the risk of the failure during the pull-off stage in the actual environment with noise and interference.

2. Frequency misalignment

The minimum SSR for a successful intrusion with $f_s \neq 0$ Hz is also analyzed using real GPS satellite signal. The code phase synchronization error τ_s is set to be positive, while the carrier phase synchronization error θ_s is random, which is in line with the level of a real spoofer. The overall results under different f_s coincide with the case where there is no frequency error but $\theta_s = \pi/2$ rad. An example with SSR=6 dB is given in Fig.12. It can be seen that the tracking trajectories and the zero-crossings under the condition of $f_s = 2$ Hz and $\theta_s = 0$ rad are compared with whose under the condition of

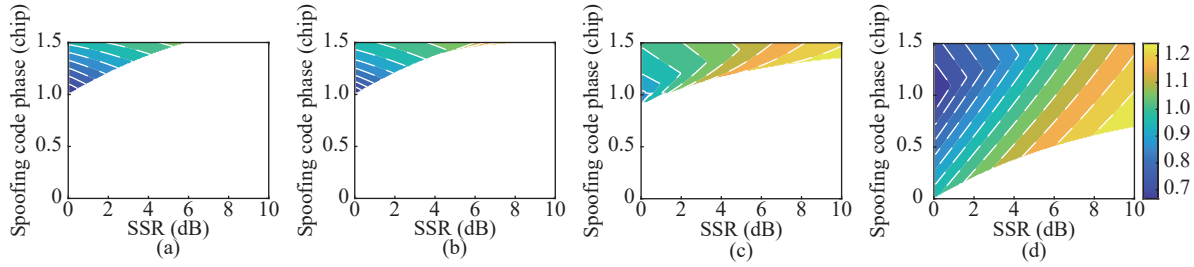

Fig. 10. The width (in chips) of the pull-in range on the spoofing signal side with EML spacing = 0.5 chip and carrier phase error = (a) 0; (b) $\pi/3$; (c) $2\pi/3$; (d) π rad, with white dash lines representing contour lines.

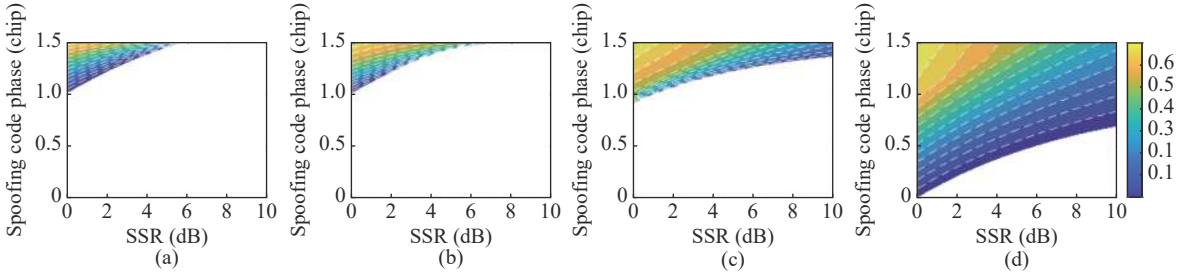

Fig. 11. The width (in chips) of the pull-in range on the authentic signal side with EML spacing = 0.5 chip and carrier phase error = (a) 0; (b) $\pi/3$; (c) $2\pi/3$; (d) π rad, with white dash lines representing contour lines.

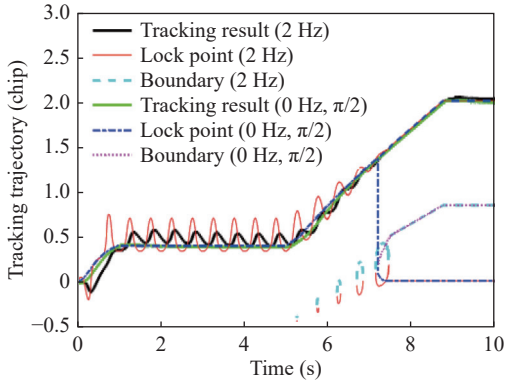

Fig. 12. Code tracking trajectory with EML spacing = 0.5 chip, SSR = 6 dB, code phase synchronization error = 0.2 chip, and frequency error = 2 Hz, compared with the case with frequency error = 0 Hz and carrier phase synchronization error = $\pi/2$ rad.

$f_s = 0$ Hz and $\theta_s = \pi/2$ rad. In this simulation, the spoofing signal keeps static for 1–5 s, and the code tracking trajectory oscillates at the frequency $f_s = 2$ Hz during this period. In this case, the loss of the sinc function after coherent integration is still negligible, but S-curve oscillation effect is more frequent. The carrier phase of the spoofing signal can change back and forth between 0 and 2π rad, and the effect of the carrier phase is greatly reduced compared to the case of $f_s = 1$ Hz. As the spoofing signal leaves the authentic signal from 5 s to 9.5 s, the oscillation amplitude gradually attenuates during this period. The overall tracking trajectories in these two cases coincide with each other, which verifies the above analysis. Therefore, the spoofing with frequency misalignment can be approximated to the case of frequency alignment with the same SSR and $\theta_s = \pi/2$ rad.

V. Experiments with Simulations and Real Data

First, Monte Carlo numerical experiments are conducted to estimate the minimum SSR required to achieve a certain spoofing success rate. Since the carrier phase of the simulated signal is easy to control, the influence of the synchronization error and the carrier-to-noise ratio is analyzed by simulating the authentic and spoofing GPS PRN 3 L1 C/A signals using our software IF signal simulator in Section V.1 and Section V.2. For the frequency misalignment experiment, the real GPS PRN 31 L1 C/A signal collected at 9 am, March 31st, 2021 is used as the authentic signal in Section V.3. After the authentic signal state parameters are estimated, the corresponding spoofing signal is generated by our software IF signal simulator and mixed with the authentic signal. All the above signals are processed by our software GPS receiver. Each experiment is conducted with 100 runs for each case of parameter setting. Given the experimental parameters shown in Table 2, the SSR for 99% successful spoofing runs and 1% failed run out of 100 runs can be regarded as the minimum SSR to achieve a 99% spoofing success rate. We also verify our analyses using the TEXBAT datasets in Section V.4, which contains 7 synchronous spoofing scenarios (Scenarios 2–8) of GPS L1 C/A signals. The description of the TEXBAT datasets can be found in [19] and [20].

Table 2. Experimental parameters

Parameter	Setting	Parameter	Setting
Sampling rate	160 MHz	DLL bandwidth	1, 2 Hz
Integration cycle	5 ms	PLL bandwidth	15 Hz
Signal	GPS L1 C/A	Power-increasing rate	0.25 dB/s
PRN	3, 31	Pulling-off rate	0.5 chip/s
C/N_0	34–48 dB-Hz	EML spacing	0.3, 0.5 chip

1. Effect of synchronization error

The SSR changes from 0 dB to 10 dB with a step of 0.05 dB to search for the minimum SSR for each code phase synchronization error setting. The minimum SSRs that achieve a 99% spoofing success rate and the minimum SSRs for the target CTL captured by the spoofing signal before the pull-off stage are demonstrated in Fig.13. The theoretical results calculated according to (13) are also plotted with dash lines for comparison. Overall, the trend of the analytical calculation results coincides with the simulation results. In addition, the minimum SSR required for the 99% spoofing success rate is slightly higher than the SSR of the target CTL captured by the spoofing signal, which can be explained by the analysis in Section IV.2, that is, the discriminator is more likely to be re-captured during the pull-off stage. As the carrier phase error increases, the minimum SSR for a successful spoofing increases dramatically, especially when the carrier phase error exceeds $\theta_s = \pi/2$ rad. If the carrier phase error is 0 rad, the minimum SSR is 0.86 dB for a code synchronization error of 0.5 chip, whereas this value reaches 5.2 dB if the carrier phase error is π rad. Furthermore, the dynamic tolerance of the target receiver also affects the spoofing success rate. For example, comparing Fig.13(a) and Fig.13(b), it can be seen that the minimum SSR increases if the EML spacing narrows down from 0.5 chip to 0.3 chip. This is because the dynamic tolerance of the CTL decreases as the EML spacing decreases, and the pull-off becomes more difficult. A similar conclusion can be obtained when the victim receiver adopts a smaller CTL bandwidth, for example, Fig.13 (a) and Fig.13(c).

The minimum SSR are shown in Fig.14 under a negative spoofing code phase synchronization error (and a positive code phase pull-off direction). It can be seen

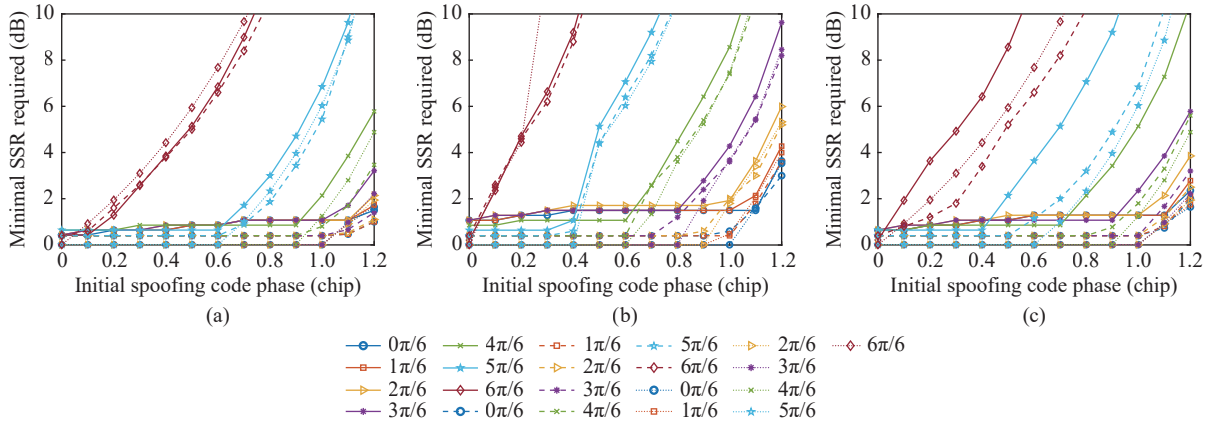

Fig. 13. Minimum SSR for a 99% spoofing success rate (solid lines with circular marks), the CTL captured by the spoofing signal before the pull-off stage (dash lines with squared marks), and the theoretical results according to (13) (dotted line without marks), under different code phase synchronization errors ($\geq$0) and carrier phase errors, with (a) EML spacing = 0.5 chip and CTL bandwidth = 2 Hz; (b) EML spacing = 0.3 chip and CTL bandwidth = 2 Hz; and (c) EML spacing = 0.5 chip and CTL bandwidth = 1 Hz.

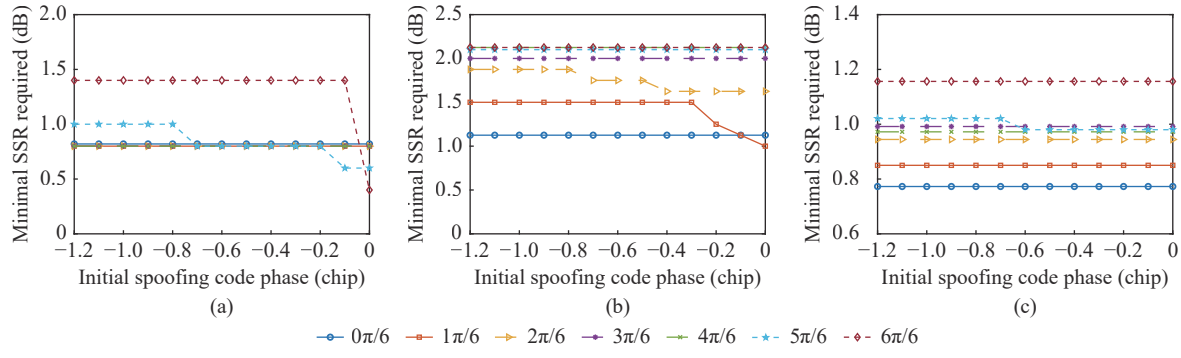

Fig. 14. Minimum SSR for a 99% spoofing success rate under different code phase synchronization errors (<0) and carrier phase errors, with (a) EML spacing = 0.5 chip and CTL bandwidth = 1 Hz, (b) EML spacing = 0.3 chip and CTL = 1 Hz, (c) EML spacing = 0.5 chip and CTL bandwidth = 2 Hz.

that the required minimum SSR is much lower than the cases where the code phase synchronization errors are positive, and the values of the synchronization error do not affect the minimum SSR significantly. Since the spoofing signal will inevitably sweep over the authentic signal during the pull-off stage, the synchronization error does not play an important role in these cases. When the carrier phase error θ_s sweeps from 0 to π rad, the minimal SSR required increases as the width of the pull-in range on the spoofing signal side at the critical moment shrinks, and the total amplitude of the CCF is greatly counteracted when θ_s approaches to π rad. By comparing the results shown in Fig.14(a) and Fig.14(c), it can be seen when the CTL bandwidth comes to 2 Hz, the spoofing is potential to succeed with SSR less than 1 dB and the effect caused by carrier phase error reduces. Overall, the previous analysis is verified in this simulation.

2. Effect of C/N_0

The minimum SSR required to achieve a 99% spoofing success rate with $\tau_s = 0.5$ chip and $\tau_s = -0.5$ chip under different C/N_0 are shown in Fig.15.

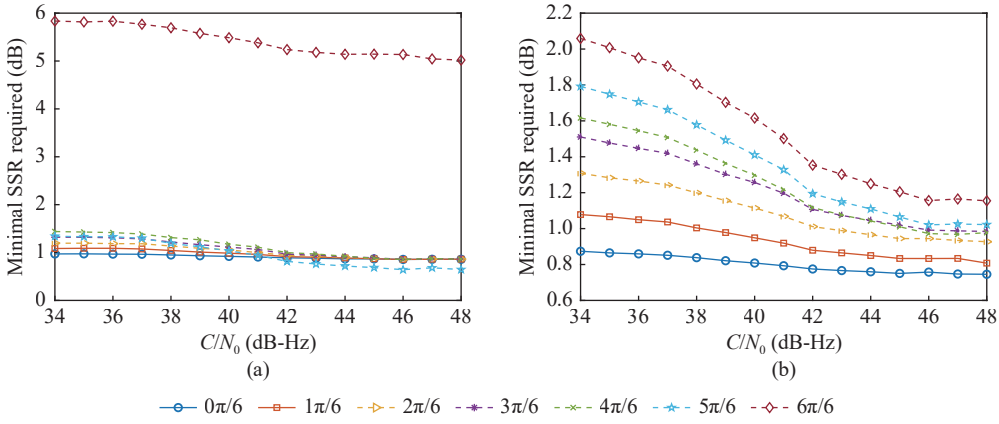

Fig. 15. Minimum SSR for a 99% spoofing success rate under different carrier phase errors and C/N_0, with EML spacing = 0.5 chip, CTL bandwidth = 2 Hz, and code phase synchronization error = (a) 0.5 chip, (b) −0.5 chip.

It can be seen that the minimum SSR required for a successful spoofing get higher as C/N_0 decreases. However, the difference between the required SSRs with C/N_0= 34 dB-Hz and 48 dB-Hz is less than 1 dB. Hence, it can be concluded that the noise brings some randomness and instability to the spoofing effect. However, it does not change the overall trend of the required SSR significantly.

3. Effect of frequency misalignment

The minimum SSR for a successful intrusion with $f_s \neq 0$ Hz is also analyzed using real GPS satellite signal. The code phase synchronization error τ_s is set to be positive, while the initial carrier phase synchronization error θ_s is random, which is in line with the level of a real spoofer. As demonstrated in Fig.16, the overall results under different f_s coincide with the case where there is no frequency error but $\theta_s = \pi/2$ rad. When f_s reaches 5 Hz, the required minimum SSR approaches to its counterpart with no frequency error but $\theta_s = \pi/2$ rad. In this case, the loss of the sinc function after coherent integration is still negligible, but S-curve oscillation effect is more frequent. The carrier phase of the spoofing signal can change back and forth between 0 and 2π rad more frequently, and the effect of the initial carrier phase is greatly reduced compared to the case of $f_s = 1$ Hz. Overall, the analysis in the above section is verified.

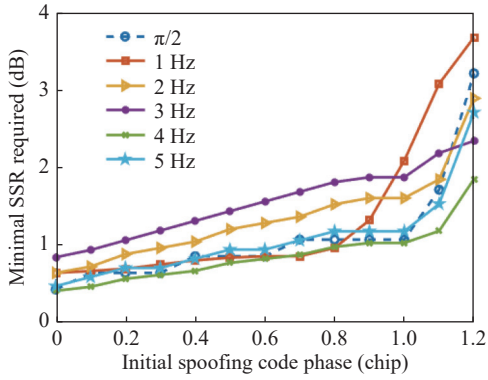

Fig. 16. Minimum SSR for a 99% spoofing success rate under different code phase synchronization errors (≥ 0) and carrier frequency errors, with EML spacing = 0.5 chip, CTL bandwidth = 2 Hz, carrier tracking loop bandwidth = 15 Hz.

4. Verification with real data

The SSR for Scenarios 2–6 in TEXBAT are 10, 1.3, 0.4, 9.9, 0.8 respectively [19], and the maximum value of the SSR is 6 dB for Scenarios 7 and 8, with a ramping amplitude during the pull-off [20]. We select PRN 3 for Scenarios 2–4, 7, and 8, and PRN 15 for Scenarios 5 and 6 to process the data. Although there is no initial code phase synchronization error in TEXBAT, differ-

ent start time in the pull-off stage can reflect different code phase error in the experiment. We start the tracking loops from the authentic side with a controllable start time (with a step of 0.02 chip approximately), thus the maximum code phase synchronization error for the success of spoofing can be searched out. There is more or less frequency misalignment in Scenarios 2–6, and the ramping amplitudes in Scenarios 7 and 8 can introduce a PLL fluctuation similar to the influence of the frequency misalignment. Therefore, the results derived from (13) with $\theta_s = \pi/2$ rad coincide with the experiment results. As shown in Fig.17 , the differences between theoretical results and experimental results are within 0.1 chip, which verifies the correctness of the necessary condition defined in (13).

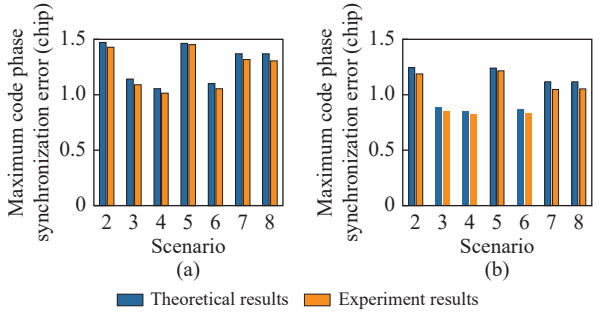

Fig. 17. Theoretical and experimental results of maximum code phase synchronization errors for successful spoofing. (a) EML spacing = 0.5 chip and CTL bandwidth = 1 Hz, (b) EML spacing = 0.3 chip and CTL bandwidth = 1 Hz.

VI. Conclusions

By establishing the model of the target receiver CTL tracking error in typical spoofing scenarios and analyzing the process of spoofing signal taking over the CTL, this study reveals the factors that determines the success of spoofing, which indicates that:

First, if the pull-off is in the same direction as the code synchronization error, a high enough SSR is necessary to compensate the synchronization error to satisfy (13) in order to avoid two possible lock points before the pull-off stage. The minimum SSRs for successful spoofing calculated from (13) agree with the simulation results within 1 dB as long as the spoofing pull-off speed is within the CTL dynamic tolerance. The overall effect of the frequency synchronization error can be approximately equivalent to the case with a same SSR, $\theta_s = \pi/2$ rad and no frequency error. The experimental results of maximum code phase synchronization error with the TEXBAT datasets agree with (13) within 0.1 chip, which verifies the correctness of the proposed necessary condition.

Second, if the pull-off direction is opposite to the code synchronization error, the spoofing intrusion can sweep over the code phase of the authentic signal being tracked so that the spoofing is more likely to succeed, as the necessary condition defined in (13) must be fulfilled during the pull-off. In this case, the initial code synchronization error has little effects, and the SSR required to achieve a spoofing success rate of 99% can be as low as 1 dB, if EML spacing = 0.5 chip, CTL bandwidth = 2 Hz, and carrier phase synchronization error $\theta_s < \pi/2$ rad.

Future study will focus on the development of spoofing and anti-spoofing methods under different spoofing strategies.

Appendix A

The closed-form solutions of the zero-crossing points of the S-curve with SSR ≥ 0, and $\varepsilon - \tau_s < d$ are listed as follows.

C12:

$$\varepsilon_{\mathrm{C12}} = \frac{-b_{\mathrm{C12}} \pm \sqrt{b_{\mathrm{C12}}^2 - 4a_{\mathrm{C12}}c_{\mathrm{C12}}}}{2a_{\mathrm{C12}}}, \quad \text{s.t.} \begin{cases} d-1 \leq \varepsilon < -d \\ -1-d \leq \varepsilon - \tau_s < d-1 \end{cases} \tag{A-1}$$

where

$$\begin{aligned} a_{\mathrm{C12}} &= \alpha^2 + 2\alpha\cos(\theta_s) \\ b_{\mathrm{C12}} &= 4d + 2\alpha^2(1+d-\tau_s) + 2\alpha\cos(\theta_s)\left[2+2d-\tau_s\right] \\ c_{\mathrm{C12}} &= 4d + \alpha^2(1-\tau_s+d)^2 + 2\alpha\cos(\theta_s)(1+d)(1+d-\tau_s) \end{aligned}$$

C13:

$$\varepsilon_{\mathrm{C13}} = \frac{-b_{\mathrm{C13}} \pm \sqrt{b_{\mathrm{C13}}^2 - 4a_{\mathrm{C13}}c_{\mathrm{C13}}}}{2a_{\mathrm{C13}}}, \quad \text{s.t.} \begin{cases} -d \leq \varepsilon < d \\ -1-d \leq \varepsilon - \tau_s < d-1 \end{cases} \tag{A-2}$$

where

$$\begin{aligned} a_{\mathrm{C13}} &= -\alpha^2 + 2\alpha\cos(\theta_s) \\ b_{\mathrm{C13}} &= 2\left[2(1-d) - \alpha^2(1+d-\tau_s) - \alpha\cos(\theta_s)(\tau_s - 2d)\right] \\ c_{\mathrm{C13}} &= -\alpha^2(1+d^2+\tau_s^2+2d-2\tau_s-2d\tau_s) \\ &\quad - 2\alpha\cos(\theta_s)(1-\tau_s-d^2+d\tau_s) \end{aligned}$$

C14:

$$\varepsilon_{\mathrm{C14}} = \frac{-b_{\mathrm{C14}} \pm \sqrt{b_{\mathrm{C14}}^2 - 4a_{\mathrm{C14}}c_{\mathrm{C14}}}}{2a_{\mathrm{C14}}}, \quad \text{s.t.} \begin{cases} d \leq \varepsilon < 1-d \\ -1-d \leq \varepsilon - \tau_s < d-1 \end{cases} \tag{A-3}$$

where

$$\begin{aligned} a_{\mathrm{C14}} &= -\alpha^2 + 2\alpha\cos(\theta_s) \\ b_{\mathrm{C14}} &= -2\left[2d + \alpha^2(1+d-\tau_s) + \alpha\cos(\theta_s)(-2d+\tau_s)\right] \\ c_{\mathrm{C14}} &= 4d - \alpha^2(1+d^2+\tau_s^2+2d-2\tau_s-2d\tau_s) \\ &\quad - 2\alpha\cos(\theta_s)(1-\tau_s-d^2+d\tau_s) \end{aligned}$$

C15:

$$\varepsilon_{\mathrm{C15}} = \frac{-b_{\mathrm{C15}} \pm \sqrt{b_{\mathrm{C15}}^2 - 4a_{\mathrm{C15}}c_{\mathrm{C15}}}}{2a_{\mathrm{C15}}}, \quad \text{s.t.} \begin{cases} 1-d \le \varepsilon < 1+d \\ -1-d \le \varepsilon - \tau_s < d-1 \end{cases} \tag{A-4}$$

where

$$a_{\mathrm{C15}} = 1-\alpha^2$$
$$b_{\mathrm{C15}} = -2-2d-2\alpha^2(1+d-\tau_s)$$
$$c_{\mathrm{C15}} = 1+2d+d^2-\alpha^2(1+d^2+\tau_s^2+2d-2\tau_s-2d\tau_s)$$

C22:

$$\varepsilon_{\mathrm{C22}} = \frac{1+\alpha^2-\alpha^2\tau_s-\alpha\cos(\theta_s)(-2+\tau_s)}{-1-\alpha^2-2\alpha\cos(\theta_s)}, \quad \text{s.t.} \begin{cases} d-1 \le \varepsilon < -d \\ d-1 \le \varepsilon - \tau_s < -d \end{cases} \tag{A-5}$$

C23:

$$\varepsilon_{\mathrm{C23}} = \frac{-b_{\mathrm{C23}} \pm \sqrt{b_{\mathrm{C23}}^2 - 4a_{\mathrm{C23}}c_{\mathrm{C23}}}}{2a_{\mathrm{C23}}}, \quad \text{s.t.} \begin{cases} -d \le \varepsilon < d \\ d-1 \le \varepsilon - \tau_s < -d \end{cases} \tag{A-6}$$

where

$$a_{\mathrm{C23}} = \alpha\cos(\theta_s)$$
$$b_{\mathrm{C23}} = 1-d-\alpha^2 d-\alpha\cos(\theta_s)\tau_s+\alpha\cos(\theta_s)$$
$$c_{\mathrm{C23}} = -\alpha^2 d+\alpha^2\tau_s d+\alpha d^2\cos(\theta_s)-\alpha d\cos(\theta_s)$$

C24:

$$\varepsilon_{\mathrm{C24}} = \frac{1-\alpha^2+\tau_s\alpha^2-\alpha\cos(\theta_s)\tau_s}{1+\alpha^2-2\alpha\cos(\theta_s)}, \quad \text{s.t.} \begin{cases} d \le \varepsilon < 1-d \\ d-1 \le \varepsilon - \tau_s < -d \end{cases} \tag{A-7}$$

C32:

$$\varepsilon_{\mathrm{C32}} = \frac{-b_{\mathrm{C32}} \pm \sqrt{b_{\mathrm{C32}}^2 - 4a_{\mathrm{C32}}c_{\mathrm{C32}}}}{2a_{\mathrm{C32}}}, \quad \text{s.t.} \begin{cases} d-1 \le \varepsilon < -d \\ -d \le \varepsilon - \tau_s < d \end{cases} \tag{A-8}$$

where

$$a_{\mathrm{C32}} = \alpha\cos(\theta_s)$$
$$b_{\mathrm{C32}} = -d+\alpha^2(1-d)+\alpha(1-\tau_s)\cos(\theta_s)$$
$$c_{\mathrm{C32}} = -d-\alpha^2(1-d)\tau_s-\alpha(d-d^2+\tau_s)\cos(\theta_s)$$

C33:

$$\varepsilon_{\mathrm{C33}} = \frac{\left[\alpha^2+\alpha\cos(\theta_s)\right]\tau_s}{(1+\alpha^2)+2\alpha\cos(\theta_s)}, \quad \text{s.t.} \begin{cases} -d \le \varepsilon < d \\ -d \le \varepsilon - \tau_s < d \end{cases} \tag{A-9}$$

C34:

$$\varepsilon_{\mathrm{C34}} = \frac{-b_{\mathrm{C34}} \pm \sqrt{b_{\mathrm{C34}}^2 - 4a_{\mathrm{C34}}c_{\mathrm{C34}}}}{2a_{\mathrm{C34}}}, \quad \text{s.t.} \begin{cases} d \le \varepsilon < 1-d \\ -d \le \varepsilon - \tau_s < d \end{cases} \tag{A-10}$$

where

$$a_{\mathrm{C34}} = -\alpha\cos(\theta_s)$$
$$b_{\mathrm{C34}} = -d+\alpha^2(1-d)+\alpha(1+\tau_s)\cos(\theta_s)$$
$$c_{\mathrm{C34}} = d-\alpha^2(1-d)\tau_s+\alpha(d-d^2-\tau_s)\cos(\theta_s)$$

C35:

$$\varepsilon_{\mathrm{C35}} = \frac{-b_{\mathrm{C35}} \pm \sqrt{b_{\mathrm{C35}}^2 - 4a_{\mathrm{C35}}c_{\mathrm{C35}}}}{2a_{\mathrm{C35}}}, \quad \text{s.t.} \begin{cases} 1-d \le \varepsilon < 1+d \\ -d \le \varepsilon - \tau_s < d \end{cases} \tag{A-11}$$

where

$$a_{\mathrm{C35}} = 1-2\alpha\cos(\theta_s)$$
$$b_{\mathrm{C35}} = -2-2d+4\alpha^2(1-d)+2\alpha\cos(\theta_s)(2d+\tau_s)$$
$$c_{\mathrm{C35}} = 1+2d+d^2-4\alpha^2(1-d)\tau_s + 2\alpha\cos(\theta_s)(1-\tau_s-d^2-d\tau_s)$$

References

[1] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, *et al.*, "Assessing the spoofing threat: Development of a portable GPS civilian spoofer," in *Proceedings of the 21st International Technical Meeting of the Satellite Division of the Institute of Navigation*, Savannah, GA, USA, pp.2314–2325, 2008.

[2] P. Papadimitratos and A. Jovanovic, "Protection and fundamental vulnerability of GNSS," in *Proceedings of IEEE International Workshop on Satellite and Space Communications*, Toulouse, France, pp.167–171, 2008.

[3] M. Zhou, H. Li, and M. Q. Lu, "Calculation of the lower limit of the spoofing-signal ratio for a GNSS receiver-spoofer," *EURASIP Journal on Wireless Communications and Networking*, vol.2018, article no.44, 2018.

[4] A. M. Khan, N. Iqbal, A. A. Khan, *et al.*, "Detection of intermediate spoofing attack on global navigation satellite system receiver through slope based metrics," *The Journal of Navigation*, vol.73, no.5, pp.1052–1068, 2020.

[5] J. X. Tu, X. Q. Zhan, M. L. Chen, *et al.*, "GNSS intermediate spoofing detection via dual-peak in frequency domain and relative velocity residuals," *IET Radar, Sonar & Navigation*, vol.14, no.3, pp.439–447, 2020.

[6] Y. J. Gao, Z. W. Lv, and L. D. Zhang, "Asynchronous lift-off spoofing on satellite navigation receivers in the signal tracking stage," *IEEE Sensors Journal*, vol.20, no.15, pp.8604–8613, 2020.

[7] M. Zhou, Y. Liu, L. Xie, *et al.*, "Performance analysis of spoofing signal ratio for receiver-spoofer," in *Proceedings of the 2017 International Technical Meeting of The Institute of Navigation*, Monterey, CA, USA, pp.898–911, 2017.

[8] K. Benachenhou and M. L. Bencheikh, "Detection of global positioning system spoofing using fusion of signal quality monitoring metrics," *Computers & Electrical Engineering*, vol.92, article no.107159, 2021.

[9] W. Y. Gao, H. Li, and M. Q. Lu, "Multi-channel joint signal quality monitor method for detecting GNSS time syn-

chronization attacks, " in *Proceedings of the 34th International Technical Meeting of the Satellite Division of the Institute of Navigation*, St. Louis, MO, USA, pp.4274–4287, 2021.

[10] S. Jeong and J. Lee, "Synthesis algorithm for effective detection of GNSS spoofing attacks," *International Journal of Aeronautical and Space Sciences*, vol.21, no.1, pp.251–264, 2020.

[11] D. Miralles, A. Bornot, P. Rouquette, *et al.*, "An assessment of GPS spoofing detection via radio power and signal quality monitoring for aviation safety operations," *IEEE Intelligent Transportation Systems Magazine*, vol.12, no.3, pp.136–146, 2020.

[12] Y. W. Wang, Y. H. Kou, Y. Zhao, *et al.*, "Detection of synchronous spoofing on a GNSS receiver using weighed double ratio metrics," *GPS Solutions*, vol.26, no.3, article no.articleno.91, 2022.

[13] F. Wang, H. K. Zhao, Y. X. Xu, *et al.* , "GNSS spoofing detection and elimination based on multipath mitigation technology, " in *Proceedings of 2020 China Satellite Navigation Conference (CSNC): Volume III*, J. D. Sun, C. F. Yang, J. Xie, *et al.*, Eds., Springer, Singapore, pp.665–677, 2020.

[14] Y. Wang, J. M. Hao, W. P. Liu, *et al.*, "Dynamic evaluation of GNSS spoofing and jamming efficacy based on game theory," *IEEE Access*, vol.8, pp.13845–13857, 2020.

[15] L. He, H. Li, W. Y. Li, *et al.* , "Neural network based C/N0 abnormity detection method for GPS anti-spoofing," in *Proceedings of the 2016 International Technical Meeting of The Institute of Navigation*, Monterey, CA, USA, pp.716–725, 2016.

[16] L. Y. Liu and M. Amin, "Tracking performance of the coherent and noncoherent discriminators in strong multipath," *IEEE 2007 9th International Symposium on Signal Processing and Its Applications (ISSPA)*, Sharjah, United Arab Emirates, pp.1–4, 2007.

[17] L. Cheng, J. Chen, and M. G. Gan, "Multipath error analysis of carrier tracking loop in GPS receiver," in *Proceedings of the 29th Chinese Control Conference*, Beijing, China, pp.4137–4141, 2010.

[18] J. Chen, L. Cheng, and M. G. Gan, "Modeling of GPS code and carrier tracking error in multipath," *Chinese Journal of Electronics*, vol.21, no.1, pp.78–84, 2012.

[19] T. E. Humphreys, J. A. Bhatti, D. Shepard, *et al.*, "The Texas spoofing test battery: toward a standard for evaluating GPS signal authentication techniques," in *Proceedings of the 25th International Technical Meeting of the Satellite Division of the Institute of Navigation*, Nashville, TN, USA, pp.3569–3583, 2012.

[20] T. Humphreys, "TEXBAT data sets 7 and 8," Available at: *http://radionavlab.ae.utexas.edu/datastore/texbat/texbat_ds7_and_ds8.pdf*, 2021.

[21] R. E. Phelts and D. M. Akos, "Effects of signal deformations on modernized GNSS signals," *Journal of Global Positioning Systems*, vol.5, no.1-2, pp.2–10, 2006.

[22] E. Shafiee, M. R. Mosavi, and M. Moazedi, "Detection of spoofing attack using machine learning based on multi-layer neural network in single-frequency GPS receivers," *The Journal of Navigation*, vol.71, no.1, pp.169–188, 2018.

[23] L. Cheng, J. Chen, and G. Xie, "Model and simulation of multipath error in DLL for GPS receiver," *Chinese Journal of Electronics*, vol.23, no.3, pp.508–515, 2014.

WANG Yiwei received the B.S. and M.S. degrees from School of Telecommunications Engineering from Xidian University, China, in 2015 and 2018, respectively. He is currently a Ph.D. student in the School of Electronics and Information Engineering at Beihang University, China. His research interests include GNSS signal processing and high-performance receivers design. (Email: ywwang@buaa.edu.cn)

KOU Yanhong (corresponding author) received the Ph.D. degree from Beihang University in 2006. She is an Associate Professor in the School of Electronics and Information Engineering, Beihang University, Beijing, China. Her research interests include high-performance GNSS receivers and simulators, signal processing, and satellite communication. (Email: kouy@buaa.edu.cn)

HUANG Zhigang received the Ph.D. degree from Beihang University in 2004. He is a Full Professor in the School of Electronics and Information Engineering at Beihang University. His research interests include wireless indoor positioning and integrity algorithms.
(Email: baahzg@163.com)