GNSS Spoofing Mitigation in the Position Domain

Fabian Rothmaier, Yu-Hsuan Chen, Sherman Lo, Todd Walter, *Stanford University*

BIOGRAPHY

Fabian Rothmaier is a PhD candidate at the GPS Laboratory at Stanford University. He received his B. Engr. degree from the University of Applied Sciences Bremen, Germany in 2015 and his M. Sc. degree from Stanford University in 2017.

Yu-Hsuan Chen is a research associate at the Stanford GPS Laboratory. He received his Ph.D. in electrical engineering from National Cheng Kung University, Taiwan.

Sherman Lo is a senior research engineer at the Stanford GPS Laboratory. He received his Ph.D. in Aeronautics and Astronautics from Stanford University in 2002. He has and continues to work on navigation robustness and safety, often supporting the FAA. He has conducted research on Loran, alternative navigation, SBAS, ARAIM, GNSS for railways and automobile. He also works on spoof and interference mitigation for navigation. He has published over 100 research papers and articles.

Todd Walter received his Ph.D. in Applied Physics from Stanford University in 1993. He is a Research Professor in the Department of Aeronautics and Astronautics at Stanford University. His research focuses on implementing high-integrity air navigation systems. He has received the ION Thurlow and Kepler awards. He is also a fellow of the ION and has served as its president.

ABSTRACT

In this paper we present an alternative approach to spoofing mitigation. We leverage the fact that under most attack modes, both the authentic and spoofed signals are received by the victim. Once an attack is detected by conventional spoofing detection means, the receiver scans the vicinity of each satellite signal's correlation peak for secondary peaks. In the meantime, the navigation continues in dead reckoning mode, based on other sensors and the user's dynamics model. A decision about which signals to trust is then cast in the position domain.

Once sufficient secondary peaks are detected, several navigation solutions are created from combinations of main and secondary peaks. We show approaches to drastically reduce the number of considered solutions at every epoch. Through recursive Bayesian estimation of the likelihood of each set of peaks we determine unlikely sets that can be ignored in future epochs. We further reduce the computational load by sharing computations among peak combinations.

As an example, we apply the procedure to the TEXBAT dataset. We demonstrate a detection of secondary peaks even in the scenario where the spoofer has the largest power advantage and recover the spoofed and authentic navigation solutions in all cases. We finally discuss how to select the authentic solution among the two using the results in the position domain, leveraging measurements from auxiliary sensors such as an IMU.

I. INTRODUCTION

GNSS has become the foundation of the Position, Navigation and Time (PNT) solution of many safety of life applications. This success story has been fueled by the accuracy, availability, continuity and high levels of integrity offered by satellite navigation systems. However, the quality of the service is nowadays threatened by increasing levels of interference.

Among the many types of interference, this paper considers interference through counterfeit GNSS signals. These signals are broadcasted by a malicious actor attempting to fool the victim's GNSS receiver. This attack on satellite navigation systems is known as spoofing. Significant work has been done in the field of GNSS spoofing detection, and promising results have been shown for defenses against different threat scenarios. Overviews of attack modes and common defense strategies are given by [1], [2] and [3].

Much less work has been done on recovering the authentic position solution once a spoofing attack has been detected.

The concept of excluding incorrect signals from the navigation solution dates back to at least the Fault Detection and Exclusion (FDE) algorithms of Receiver Autonomous Integrity Monitoring (RAIM). If at least six satellite signals are visible, FDE can identify and exclude an erroneous signal. But RAIM is not designed to protect against intentional interference and currently only covers faults on a single satellite. The same limitation applies to the estimation scheme presented in [4]. Advanced RAIM (ARAIM) will go further by considering multiple faults and constellations [5].

Several techniques focus on suppressing the spoofed signals assuming they are broadcasted from one direction. The idea is to eliminate the malicious signals from the received RF pattern, such that once again only authentic signals are visible to the receiver. This has been achieved by steering a spatial null towards the largest power source using an array of multiple antennas [6], [7]. Alternatively, the correlation in Doppler variation caused by antenna movement can be leveraged to identify

and exclude satellite signals coming from the same direction [8].
Several signal processing techniques have been explored that eliminate signals without the use of multiple or moving antennas. [9], [10] and [11] eliminate one signal per PRN for example by a projection the signals onto their nullspace or by superpositioning the opposite signal. All three approaches however make strong assumptions on which signal per PRN is the authentic and which one is the spoofed signal.

In this paper we present an approach that does not eliminate any signals, but tracks up to two correlation peaks per PRN. During a spoofing attack, we track and decode both the authentic and spoofed signal for each satellite. We consider all possible navigation solutions from the different combinations of peaks and track the likelihood of each solution by considering each solution's pseudorange residuals. Two solutions emerge as most probable, namely the authentic and spoofed navigation solution. We cast the decision about which solution to trust in the position domain with a Multi-Hypothesis Kalman Filter.
The approach does not require multiple antennas capable of beamstearing or a movable antenna. It further makes no assumptions about the spoofed signal being stronger than the authentic or the receiver being locked to the authentic signal first.

The remaining paper is divided into three sections plus a summary and conclusion. In the next section we cover the detection and tracking of a secondary correlation peak. We consider computational challenges and limitations e.g. in the spoofer's power advantage.
In the ensuing section we discuss the efficient computation of the up to 2^N different position solutions from N PRNs with two peaks each. We present a Bayesian filter architecture tracking the likelihood of each computed position solution. We describe the filter architecture used to differentiate between authentic and spoofed solution.
Finally we apply the discussed approach to the TEXBAT dataset [12]. We demonstrate detection of secondary peaks and successful tracking of both the spoofed and authentic position solution in two scenarios and analyze ways to tell the two apart.

II. SECONDARY PEAK DETECTION

To begin, we discuss the detection of a second signal in the ambiguity function of a given PRN. This technique is well established for the detection of spoofing signals, it has for example been employed in [13] and [14].

1. Signal Model

Following the notation in [15] and omitting noise for notational brevity, we express the satellite signal at time t after the receiver front end as

$$s(t) = \sqrt{C}D(t-\tau)x(t-\tau)\cos\left(2\pi(f_{IF}+f_D)t+\delta\theta\right) \tag{1}$$

For the signal power C, navigation data bit $D=\{-1;1\}$, code phase τ, spread spectrum code x, intermediate frequency f_{IF}, Doppler shift f_D and carrier phase offset $\delta\theta$. During the acquisition process, the receiver estimates τ for position and time, f_D for velocity and clock drift, and $\delta\theta$ for precise positioning. This process at the heart of GNSS signal processing is detailed in various standard texts such as [15]. As part of this procedure, the receiver multiplies the received signal with what is called the inphase and quadrature reference signals. A pair of correlators further multiplies each result with a local replica of the signal $x(t-\hat{\tau})$, where $\hat{\tau}$ is the receiver's code phase estimate, and averages the output over the correlation time T_{CO}. The outputs of inphase and quadrature correlators S_I and S_Q are then given by [15]

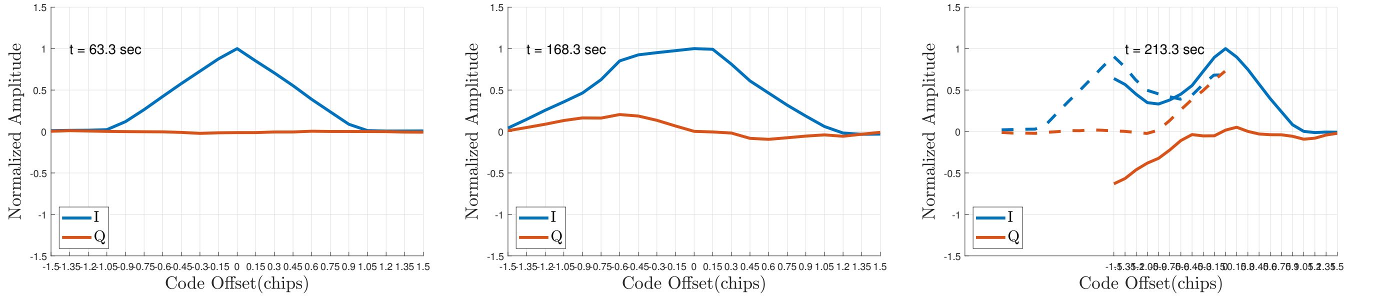
$$\begin{aligned} S_I(\Delta\tau,\Delta f_D,\Delta\theta) &= \frac{\sqrt{C}D}{T_{CO}}\int_0^{T_{CO}} x(t-\tau)x(t-\hat{\tau})\cos\left(2\pi\Delta f_D t+\Delta\theta\right)\,dt \\ S_Q(\Delta\tau,\Delta f_D,\Delta\theta) &= \frac{\sqrt{C}D}{T_{CO}}\int_0^{T_{CO}} x(t-\tau)x(t-\hat{\tau})\sin\left(2\pi\Delta f_D t+\Delta\theta\right)\,dt \end{aligned} \tag{2}$$

with the errors in code delay $\Delta\tau=\tau-\hat{\tau}$, Doppler $\Delta f_D = f_D-\hat{f}_D$ and carrier phase $\Delta\theta=\delta\theta-\hat{\theta}$. Inphase and quadrature signals are often expressed together as real and imaginary parts of the complex correlator output $\tilde{S}$.

$$\begin{aligned} \tilde{S} &= S_I + jS_Q \\ &= \sqrt{C}D\exp\left(j\Delta\theta\right)\tilde{R}(\Delta\tau,\Delta f_D) \\ \tilde{R}(\Delta\tau,\Delta f_D) &= \frac{1}{T_{CO}}\int_0^{T_{CO}} x(t-\tau)x(t-\hat{\tau})\exp\left(j2\pi\Delta f_D t\right)dt \end{aligned} \tag{3}$$

With the ambiguity function $\tilde{R}(\Delta\tau, \Delta f_D)$. For $\Delta f_D = 0$ it equals the correlation function, values around $\Delta\tau = 0$ then make up the characteristic correlation triangle. If the signal is tracked perfectly, that is $\Delta\tau = \Delta f_D = \Delta\theta = 0$, we have $S_I = \sqrt{C}D$ and $S_Q = 0$.

So far we have only considered a single signal. This is an acceptable simplification under ideal conditions. GNSS signal codes are engineered for low cross correlation, such that a signal from satellite i does not noticeably excite the ambiguity function $\tilde{R}$ when correlated with the reference signal of another satellite j.
In the presence of multipath or spoofing however, two or multiple signals with the same spreading code x but different delays τ and possibly Doppler f_D are received by the front end. As long as the receiver's analog to digital converter has sufficient dynamic range to capture all received signals, these result in elevated values of $\tilde{R}$ around each signal's delay and Doppler, represented in a superposition of signals in Eq. (3). The half-width of the correlation triangle is one chip, equal to around $300m$ in the pseudorange domain. Signals with the same spreading code and spaced less than one chip apart result in a distorted correlation peak. Signals spaced further than that result in multiple correlation peaks. We illustrate this circumstance in Figure 1. It shows the inphase (I) and quadrature (Q) correlation values between ± 1.5 chips around the prompt correlator before, during and after the lift-off of the spoofing attack of TEXBAT scenario $ds3$ [12]. During this type of attack, the spoofer initially transmits signals with the same code delay and thereby range information as the authentic signal, overpowers the authentic signals, and then gradually changes τ. We will discuss the TEXBAT attack scenarios in more detail in Section IV. For now we focus only on the effect on the correlation function. Before and during the early stage of the attack, the receiver's correlation peak shows no distortion as depicted in Figure 1a. Once the attacker changes τ, the correlation peak gets distorted by the superposition of correlation values (Figure 1b). This distortion is visible both in the inphase correlation due to the altered τ as well as the quadrature correlation due to the difference in Doppler and carrier phase between authentic and spoofed signal. Once the signals are spaced more than approximately one chip, a secondary peak is detected and tracked with a second set of 21 correlator pairs. This can be seen in Figure 1c.



(a) Early phase, spoofer transmits replica of the authentic signal. No distortion visible.

(b) Lift-off phase, spoofer alters his τ and lures the victim off course. Distortion is visible in both I and Q channel.

(c) After lift-off. Two distinct but distorted peaks are visible. The secondary peak is indicated by dashed lines.

Figure 1: Correlator values during three stages of a lift-off spoofing attack. Plots based on 21 correlator pairs spaced evenly between ± 1.5 chips around the prompt correlator of identified peaks. All values are normalized by the prompt inphase correlator of the main peak.

2. Tracking Multiple Peaks

Common GNSS receivers usually track one correlation peak per PRN, often with only three correlator pairs referred to as "early", "prompt" and "late". Various techniques exist to detect this peak, from sequential search in codephase and Doppler domain, parallel searches using multiple correlator pairs or Fast Fourier Transform (FFT) techniques [16, 17]. Once the largest peak in the search space or a peak of sufficient strength is detected, this search is usually stopped and signal acquisition complete. Reasons to not search for further peaks can range from conserving energy to simply not considering the option that there might be more than one peak. Continuous monitoring of the correlation function to identify a spoofing attack through the presence of a second correlation peak has been proposed, with challenges and capabilities in an aviation context outlined by [14].

The goal of this paper however is not only to detect spoofing, but to retrieve the authentic navigation signal despite the presence of an attack and continue safe navigation. To this end, we want to track multiple correlation peaks per PRN independently.
We use a Software Defined Radio (SDR) to assign two channels to each PRN, each with 21 evenly spaced correlator pairs between ± 1.5 chips around the prompt. One channel, from here on labeled the "main" channel, tracks the correlation peak by minimizing $\Delta\tau$, Δf_D and $\Delta\theta$ as would a normal receiver. The other, "secondary" channel scans for other peaks. At least two operational modes could be imagined, depending on the resources of the user and the given threat scenario.

1. If computational and especially energy resources are no concern, the second channel can continuously search a large space of $\{\tau; f_D; \delta\theta\}$. This continuous search can detect signals even despite large differences in pseudorange between

authentic and spoofed signal. This effort should be made at receiver startup and after a loss of signal to for increased robustness against spoofed conditions at start up and jam-then-spoof attacks.

2. Computationally less expensive is a local search around the tracked correlation peak, within a limited range of $\Delta\tau$, Δf_D and $\Delta\theta$. To further limit the energy consumption, this search can be limited to be only executed once a spoofing alarm has been raised by other means. This approaches provides robustness to so-called carry-off attacks, where the spoofer initially transmits correct pseudoranges and only slowly lures the victim off course.

For this study we followed the second approach. While the first channel is tracking a signal, the secondary channels of the SDR are in acquisition mode trying to detect secondary peaks. The search is conducted for code phases more than ± 0.4 chips from the main peak at the same Doppler frequency. Any peak with an amplitude greater than the acquisition threshold used for the main peak is detected and tracked, unless it is less than 0.4 chips from the main peak in which case the acquisition is re-initiated. It is important to note that for each PRN both peaks are tracked and decoded independently, separate sets of measurements are recorded.

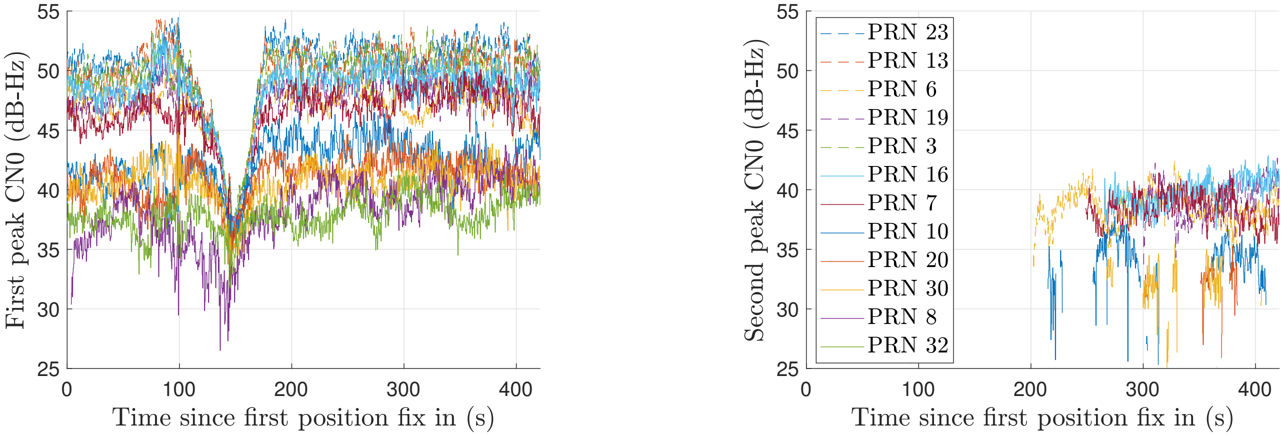
While this technique has proven powerful against attacks such as TEXBAT as we will see in Section IV, several scenarios can be imagined that would be challenging to mitigate.

- Tracking both the spoofed and authentic signal requires them to be at similar power levels. Spoofing attacks with a significant power advantage, attacks that jam or increase the noise floor and especially sophisticated attacks that null out the authentic signal prohibit this technique. In Section IV we show successful detection of secondary peaks during TEXBAT scenario $ds2$, an attack with a $10dB$ power advantage. We do however detect fewer secondary signals than during attacks with lower power advantage, the scenario can therefore likely be considered a limit case. We show the Carrier to Noise Ratio $C/N0$ during scenario $ds2$ of the 12 main peaks and 6 detected secondary peaks in Figure 2. Note the significantly lower $C/N0$ of the secondary peaks, corresponding to the $10dB$ power advantage of the malicious signals.

  These results were obtained with an SDR with 16 bit resolution analog to digital conversion. A receiver with smaller resolution would likely only be able to track signals with smaller power differences.

- Two peaks can only be identified and correctly tracked if they are sufficiently spaced. Secondary signals - caused by multipath or a spoofing attack - that are close in code phase to the authentic signal can significantly distort the peak without creating a second distinct peak. During all test scenarios, peaks were only successfully detected once they are around 1 chip spaced from the main peak. 1 chips corresponds to about $300m$ in pseudorange, distorted peaks created by signals spaced less than 1 chip therefore can lead to significantly erroneous pseudoranges before a secondary peak is detected successfully. This issue can be partially mitigated through various Signal Quality Monitoring (SQM) approaches such as the ones presented in [18] that flag a satellite as unusable if the peak is distorted.

- If a spoofing signal is present during receiver startup or if jamming is exercised before the onset of an attack, the spoofing and authentic signals can be spaced far apart, with significantly different resulting navigation solutions. Whatever peak the receiver selects as main navigation signal, the secondary signal would likely not be detected by a local search. This issue is easily mitigated if the entire $\{\tau; f_D; \delta\theta\}$ space is searched for multiple peaks during every acquisition process.



(a) $C/N0$ of main peaks.

(b) $C/N0$ of secondary peaks.

Figure 2: $C/N0$ of the first and detected secondary peaks during TEXBAT scenario $ds2$. We note the significantly lower $C/N0$ of the secondary peaks, caused by a $10dB$ power advantage of the spoofing signals. The legend in Figure 2b is valid for both plots.

In this section, have discussed the employment of a second receiver channel to track a secondary correlation peak. It is decoded independently of the main correlation peak, resulting in up to two pseudorange measurements for each PRN. So far we have made no assumption about which measurement is authentic, and which one is spoofed. In the ensuing sections we will uncover the authentic and spoofed position solutions from these pseudoranges while limiting the computational complexity and finally explore application examples.

III. IDENTIFICATION OF FEASIBLE NAVIGATION SOLUTIONS

The signal processing described in the previous section is a two-sided sword. It enables us to track and decode the authentic satellite signals despite the presence of even high powered spoofing attacks. But it leaves us with up to 2 pseudorange measurements per PRN, without a hint about which one to trust. With N satellites in view with 2 pseudoranges each, we now have the choice between 2^N different combinations to be used in the computation of the navigation solution. In this section we will discuss how to narrow the selection down to the few most probable combinations while again keeping the computational burden low.

The procedure described in the following paragraphs as well as the original search for secondary peaks is started once a spoofing attack is detected. The alarm could be raised by a detector monitoring signal power and distortion of the autocorrelation function as the ones presented in [19, 20].

1. The Figure of Merit

We follow the notation in [15] of the linearized measurement equation. For N satellites in view, we have

$$y = Gx + \epsilon \quad \text{with} \quad \epsilon \sim N(0, W^{-1}) \tag{4}$$

where $y \in \mathbb{R}^N$ are the pseudorange measurements minus the expected range, $G \in \mathbb{R}^{N \times p}$ is the geometry matrix, $x \in \mathbb{R}^p$ is the state, $\epsilon \in \mathbb{R}^N$ is the nominal measurement noise and $W \in \mathbb{R}^{N \times N}$ is the noise information matrix (the inverse of the covariance matrix). For the single constellation case we have $p = 4$ for three position variables and one clock bias state.

As we outlined in Section II.2, two peaks are only detected if the authentic and false pseudoranges are at least 0.4 chips or around $120m$ apart. Out of the 2^N peak combinations, all but two will further consist of a mix of authentic and spoofed signals. Due to the significant difference in pseudorange between two peaks, a mix of authentic and spoofed range measurements is likely inconsistent. One way of detecting such inconsistencies is through the χ^2-statistic [21].

$$t = y^T(W - WG(G^TWG)^{-1}G^TW)y \tag{5}$$

The χ^2-statistic will be elevated if the pseudoranges are inconsistent. Out of the 2^N measurement combinations, two will likely show a significantly smaller residual χ^2-statistic. These are the set of all nominal and all spoofed satellites, theoretically consisting of exactly opposite peak selections.

A brute force approach could compute the position solution and χ^2-statistic for all 2^N combinations and simply select the two with the lowest values of t. This has two drawbacks. One is the obvious computational burden of solving the navigation equation up to several thousand times at every epoch. Even when leveraging the relationship between combinations through rank one updates in solving the navigation equation (4) this might not be suitable for a receiver to be done in real time.

Secondly, while it is intuitively sensible that only two combinations with small residuals exist, and that those two represent the authentic and spoofed solution, we have no proof for this claim. At the very least, there could be geometry constellations that allow for a mixture of authentic and spoofed signals to result in small residuals. Considering only the two combinations with the smallest χ^2-statistic might in such cases result in the wrong result.

To better compare peak combinations we follow a probabilistic approach. Instead of a combination's χ^2-statistic t, we consider its likelihood. Under nominal conditions, t follows a central χ^2 distribution with $N - p$ degrees of freedom [21].

$$t \sim \chi^2_{N-p} \tag{6}$$

We define a combination of peaks in the vector $c \in \mathbb{R}^{N \times 1}$. We can compute the probability of a vector of measurements being consistent for a combination c by evaluating the probability density function (pdf) of the central χ^2 distribution

$$p(y|c) = \frac{1}{2^{(N-p)/2}\Gamma((N-p)/2)} t^{(N-p)/2-1} e^{-t/2} \tag{7}$$

where Γ is the gamma distribution and t is computed using Eq. (5). Using Bayes Rule and for a given prior probability $p(c)$, we can compute the posterior probability of a peak combination c being consistent.

$$p(c|y) = \frac{p(y|c)p(c)}{\sum_{c \in C} p(y|c)p(c)} \tag{8}$$

where C is the set of all 2^N combinations.

2. The Sequential Estimation Approach

We have now defined a probabilistic approach determining the probability of each combination of peaks. But we have left two questions unanswered. We have yet to reduce the number of combinations that we need to consider at every epoch to limit the computational burden of the approach. We further require prior probabilities of each combination for the Bayesian update in Eq. (8). We solve both questions with a sequential approach known as a Histogram Filter [22].

Similar to a Kalman Filter, a Histogram Filter alternates between time updates and measurement updates. We initialize the prior uniformly over all combinations. At each epoch the measurement update is given by Equation (8). During the time update, the vector of all posterior probabilities at epoch k is multiplied by the transition matrix T before it serves as new prior at $k+1$. In the transition matrix, we want to capture how likely peak combinations change from one epoch to another. We choose to represent the probability of one peak being spoofed instead of authentic or vice versa by the hyper-parameter λ. Assuming peaks behave independently, the probability of one peak combination c_i transitioning to another peak combination c_j is then given by λ^m, where m is the number of different peak assignments. We now represent a peak combination c of N peaks as an $N \times 1$ vector identifying which peak is used for each PRN. The likelihood of one combination c_i transitioning to combination c_j is then given by

$$\lambda^{\sum_{n=1}^{N} c_i^{(n)} \neq c_j^{(n)}} \tag{9}$$

where $c_i^{(n)}$ is the nth element of the vector c_i. We further want to encode the assumption that combinations of opposite peak selections should be equally likely, as they represent the spoofed and authentic solution. We therefore describe the similarity of two peak combinations c_i and c_j as the minimum between the number of equal peaks and the number of opposite peaks. We summarize our notation in the definition of the transition matrix $T \in \mathbb{R}^{2^N \times 2^N}$ as

$$T_{ij} = \frac{\lambda^{\min(\sum_{n=1}^{N} c_i^{(n)} \neq c_j^{(n)}, \sum_{n=1}^{N} c_i^{(n)} = c_j^{(n)})}}{\sum_i \lambda^{\min(\sum_{n=1}^{N} c_i^{(n)} \neq c_j^{(n)}, \sum_{n=1}^{N} c_i^{(n)} = c_j^{(n)})}} \tag{10}$$

where the numerator ensures the necessary normalization of the transition matrix.
The posterior probability of a peak combination c_i at epoch k is then given by

$$\begin{aligned} p(c_i|y_{1:k}) &= \frac{p(y_k|c_i)p(c_i| \text{ previous epochs})}{\sum_{c_i \in C} p(y_k|c_i)p(c_i| \text{ previous epochs})} \\ &= \frac{p(y_k|c_i)\sum_{j=1}^{N} T_{ij} p(c_j|y_{1:k-1})}{\sum_{c_i \in C} p(y_k|c_i)\sum_{j=1}^{N} T_{ij} p(c_j|y_{1:k-1})} \end{aligned} \tag{11}$$

where the subscript $y_{1:k}$ indicates all measurements between epochs 1 and k.

This formulation of T, just like the choice of using a Histogram filter as well as the following simplifications for computational simplicity represent only one possible approach to this problem. They are a heuristic that has worked well in our experience, but without any performance guarantee.

The Histogram filter tracks the probability of each combination given the measurement residuals of the current and past epochs. It is straight forward to now reduce the number of considered combinations by tracking only the combinations with the largest probabilities and setting all other probabilities to 0. This pruning can be done by eliminating combinations with probabilities below a certain threshold, by considering only a certain number of the most probable combinations, or a combination thereof. This step is done before every measurement update.

This essentially turns the Histogram Filter into a greedy, sequential search of the most probable peak combinations. The transition step ensures local exploration of the space of combinations, the measurement update evaluates the objective function and the pruning causes the greedy nature of the algorithm. A more classical greedy search for sets of satellites with consistent pseudoranges has been proposed by [23] with good results but without tracking probabilities. If computational capabilities allow, a greedy search or L1 minimization as in [23] can be employed at each epoch before computing the measurement update using Equation (8).

We summarize the sequential estimation procedure in Figure 3.

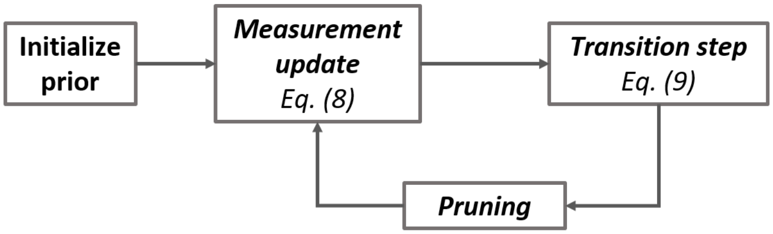


Figure 3: Schematic of the Histogram Filter architecture tracking the probability of each combination. The key equations of measurement update and transition step are outlined.

3. Tracking the Authentic Solution

So far, we have discussed identifying consistent sets of pseudoranges among the measurements obtained off the main and secondary peak of each PRN. But we have not yet gained any ground towards identifying the authentic solution. Thanks to the work we've done in the previous sections, we can now cast this decision in the position domain.

The idea is to use a (possibly Extended) Kalman Filter ((E)KF) to track our state. A common solution is to couple a six degree of freedom Inertial Measurement Unit (IMU) with the GNSS. While no spoofing alarm is raised, we estimate the IMU's bias terms for an accurate model of the accelerometer and gyro error terms. This fairly standard approach is detailed in many texts, we follow the error-state implementation in [24]. When a spoofing alarm is raised, we stop using the GNSS measurements but instead propagate the filter state using the IMU and any other sensors we might have available.

Once secondary peaks have allowed for multiple consistent position solutions, each could be used to update the (E)KF. This poses a standard multi target tracking problem with unknown data association [22]. Every position solution would have to be tracked in a separate (E)KF, the resulting state solution at every epoch can be described as a Gaussian Mixture Model. At every epoch, each (E)KF is split into many new filters: one for each new consistent position solution. It is easy to see the exponential increase in computational complexity that makes this Multi-Hypothesis-(E)KF unrealistic.

When exactly to restart using GNSS navigation solutions is once again a heuristic. We offer a criterion that has lead to good results in Section IV.

To reduce the computational complexity, we once again prune the tree of considered options. Instead of branching into many new filters at every epoch, we only track the most likely. The likelihood of a solution is made up of two components. One part is the likelihood of the solution being a consistent solution calculated by the Histogram Filter described in Section III.2. The second part is the likelihood of that solution being the measurement update of the filter tracking the authentic solution. We calculate by evaluating the normalized distance between the (E)KF's guess and the navigation solution in question. Both are multivariate Normal distributions, the likelihood is therefore given by

$$p(x_i^{(k)}|\mu^{(k-)};\Sigma^{(k-)}) = (2\pi)^{-\frac{d}{2}}\det(\Sigma^{(k-)}+P_i^{(k)})^{-\frac{1}{2}}\exp\Big((x_i^{(k)}-\mu^{(k-)})^T(\Sigma^{(k-)}+P_i^{(k)})^{-1}(x_i^{(k)}-\mu^{(k-)})\Big) \tag{12}$$

where $\Sigma^{(k-)}$ is the $d \times d$ (E)KF state covariance matrix prior to the measurement update at epoch k, $P_i^{(k)}$ is the covariance of the ith consistent solution, $x_i^{(k)}$ is the ith consistent navigation solution and $\mu^{(k-)}$ is the (E)KF state mean prior to the measurement update.

We can now finally identify the measurement that is most likely to contain the authentic solution at epoch k by solving

$$i_{auth}^{(k)} = \arg\max_i \Big(p(x_i^{(k)}|\mu^{(k-)};\Sigma^{(k-)})p(c_i|y_{1:k})\Big) \tag{13}$$

After the coasting phase, we update the filter with the over time most consistent (Eq. (11)) solution that is most likely to be the (E)KF update (Eq. (12)). The operational principle is sketched out in Figure 4.

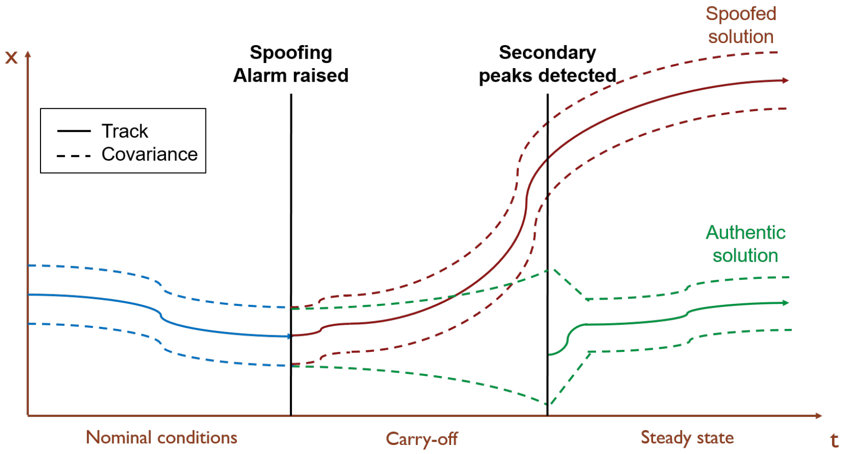


Figure 4: Conceptual depiction of the (E)KF architecture to identify the authentic solution. Once a spoofing alarm is raised, the filter coasts on non-GNSS sensors. We omit the track line in the figure during this phase to emphasize the absence of GNSS measurement updates. The covariance increases during this phase depending on the auxiliary sensors available. Once consistent navigation solutions based on secondary peaks are detected, the GNSS navigation is resumed with the most consistent, most probable solution.

Equations (12) and (13) describe how to update one (E)KF. If computational capacities allow, multiple (E)KFs can be run in parallel tracking e.g. a selection of the most likely trajectories. In Section IV we will run two filters, one tracking the authentic and one tracking the spoofed solution.

The fairly general formulation in this section is no coincidence. The (E)KF formulation depends heavily on the application and sensors at the engineer's disposal and we will explore an example in Section IV.

4. Practical Considerations

We have now defined a histogram filter architecture that tracks the probability of each of the 2^N peak combinations based on each combination's χ^2 statistic. We have introduced pruning heuristics to reduce the number of considered combinations to a tractable amount while the filter converges to the few combinations resulting in low residuals. Before exploring examples in the next section, we want to consider a few practical aspects of the algorithm.

1. The effect of a pull-off type attack on the correlation peak is depicted in Figure 1. We note that before two distinct peaks are visible, only a single, distorted peak is visible. This distortion degrades tracking performance. As we can tell from Eq. (3), tracking of the inphase signal results in our estimate of the code delay $\Delta\tau$ and thereby pseudorange. A poorly tracked peak in the inphase channel as it is visible in Figure 1b therefore causes an inaccurate pseudorange measurement, which in turn will be inconsistent with other, accurate pseudoranges. This results in an incorrect navigation solution and an elevated χ^2 statistic.
   One conclusion from this insight is to consider, if no second peak is detected for a specific PRN, navigation solutions without that PRN. This does not increase the size of the set of possible combinations C. We merely changed the options for each satellite signal from "main peak or secondary peak" to "main peak or (secondary peak if detected, otherwise no peak)".
   This has two desirable effects. One is that we effectively let the filter exclude PRNs with a single, distorted peak and can still find consistent sets despite the presence of signals with erroneous pseudoranges. Secondly we can handle an intermittent appearance of a secondary peak. If the spoofed and authentic signal are around 0.4 chips apart, the receiver might only sometimes be able to detect two distinct peaks. The likelihood of any set including that secondary peak is not affected by the peaks temporary absence as the filter instead considers the just as consistent set of peaks without that peak.

2. We discovered in the previous point that a distorted inphase correlation peak results in an inaccurate pseudorange. A simplistic measure of the peak's distortion is the Delta metric, the difference between early and late correlator with equal spacing from the prompt [25]. Under nominal conditions it is approximately zero mean Normally distributed [26]. To avoid erroneous pseudoranges disturbing the filter's convergence, we only consider pseudoranges from correlation peaks with a sufficiently small Delta metric. The threshold above which to ignore pseudoranges is another hyper-parameter.

3. A challenge in designing SQM based anti-spoofing techniques is posed by multipath. Both spoofing attacks and multipath can cause a distortion of the autocorrelation function, and both can cause a distortion of the user's navigation solution.

Means to differentiate between the two are offered, among others, by [27], [28]. Particularly relevant for the work presented in this manuscript is the underlying fact that both spoofing and multipath result in the presence of a second GNSS signal. A significant difference between the two types of interference is that secondary signals created by a spoofer result in a coherent set of pseudoranges whereas signals caused by multipath generally do not. The measurement residual based approach presented in this section would therefore determine navigation solutions based on multipath signals to be inconsistent. Even though this remains to be tested in future work, this suggests that the approach presented in this paper could be used to mitigate certain multipath scenarios.

4. Should the employed spoofing detection approach erroneously raise an alert because of multipath, two outcomes are possible. Either we do not identify a subset of consistent, line-of-sight pseudorange measurements, for example because the multipath persist for only a short time or the vast majority of signals is affected. Then the described (E)KF remains in the "Carry-off" phase of Figure 4 and coasts until the effect is over.
   The other option is that a significant subset of pseudoranges is not affected by multipath and remains consistent. The filter architecture identifies these measurements and resumes "trusted" satellite navigation as in the "Steady state" phase of Figure 4.

IV. APPLICATION EXAMPLES

In this section we now want to test the presented approach against examples from the TEXBAT dataset. The dataset is described extensively in [12]. In our further analysis we will now limit ourselves to scenarios $ds4$ and $ds6$. $ds4$ is a so-called position push attack with around $1dB$ power advantage that introduces a $600m$ offset in the ECEF z-direction for a static receiver. Scenario $ds6$ is very similar to $ds4$, but for the case of a dynamic receiver.

1. Spoofing Detection

A countless number of spoofing defenses has been developed and tested against the TEXBAT dataset, among others by the authors of [29], [30], [31], [20], [19]. Leveraged tell-tale signs include the received power, various SQM metrics, pseudorange residuals, or combinations thereof.
In this work we employ the χ^2 statistic of pseudorange residuals known from RAIM [21] as well as the power and distortion metrics defined in [20]. For each metric we cast an individual decision in a Neyman-Pearson (NP) detector [32]. We raise a spoofing alarm if any of the three detectors alert.
Any of the cited spoofing detection means could be employed for the purpose of this paper. As long as an alarm is raised before any offset is introduced in the navigation solution, the precise detection mean is outside the scope of this work.

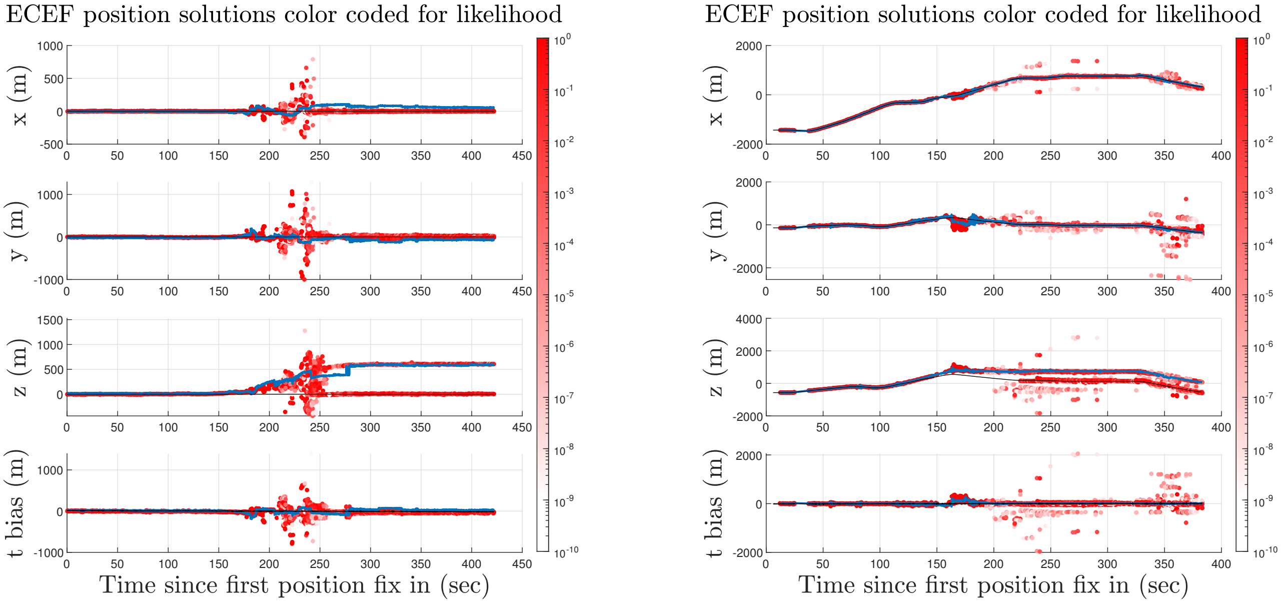
2. The most probable Navigation Solutions

In the previous sections we have introduced various hyper parameters mainly for computational tractability. To generate the results in this section we set $\lambda = 0.01$ and prune all sets with likelihood below 10^{-4}. To confirm the success of these parameters, we test the algorithm against TEXBAT scenarios $ds4$ and $ds6$. We choose these two scenarios as they represent so-called position push attacks, where the spoofer introduces an offset in the ECEF position solution (in contrast to offsets in the time solution). In our analysis, these scenarios were more difficult to track successfully during the coasting phase. With the analysis in this section we want to confirm the identification of the two consistent navigation solutions with the promised low computational burden as well as the successful continuation of satellite navigation once secondary peaks are available.

We can examine the result with respect to the first goal in Figure 5. The figures show the computed navigation solutions at each epoch in red, shaded depending on likelihood. The black solid line indicates the truth, the blue line is the solution obtained by our SDR without employing any spoofing defense. In scenario $ds4$, the spoofer introduces a $600m$ offset in the z-direction that can be observed in Figure 5a with respect to the otherwise stationary receiver. After a transition period, the SDR tracks the spoofed solution solution in the ECEF z-axis but with offsets in x, y direction. The Histogram Filter in the meantime tracks both the spoofed and the authentic solutions precisely: we observe red dots on the black line as well as on the blue, with a preference for the black. We can conclude that the authentic solution results in smaller residuals than the spoofed.

Scenario $ds6$ is similar to $ds4$, but with a dynamic user platform. We show the results of the dynamic scenario in Figure 5b. We observe very similar results as for the static scenario in Figure 5a. After a turbulent transition phase, we track both the authentic and spoofed solution. The results do show an increased amount of noise, likely due to a more obstructed sky view compared to the static scenario. The SDR has again been fooled and is following the spoofed solution.

After confirming that both consistent sets of peaks were identified and tracked by the Histogram Filter, we explore the computational burden of the approach. To avoid computing 2^N least squares solutions in real time for N observed satellites, we employ the pruning heuristics discussed in Section III.2. We show the result in Figure 6 for scenarios $ds4$ and $ds6$ by plotting the number of sets for which a least squares solution was computed, as well as the number of peak combinations possible

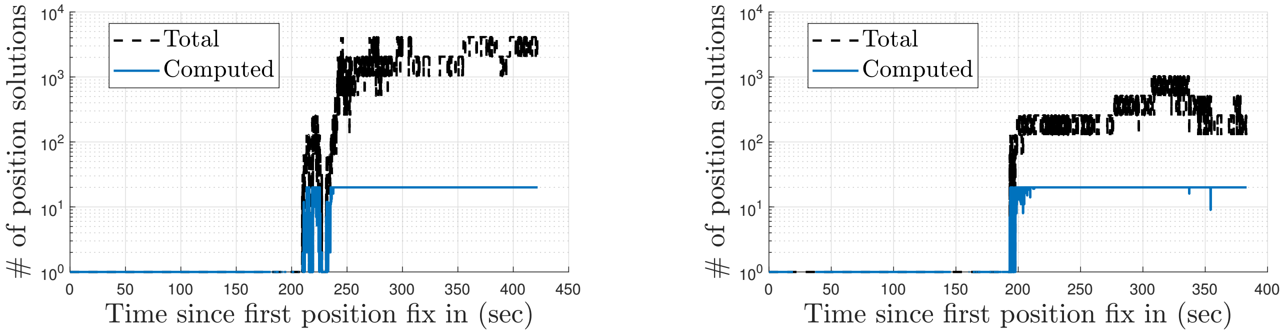


(a) Scenario $ds4$.

(b) Scenario $ds6$.

Figure 5: ECEF position solutions in meter during TEXBAT scenarios $ds4$ and $ds6$. Red dots show position solutions of the most probable solutions at each epoch, color coded for their likelihood as determined by the Histogram Filter architecture. The solid black line represents the truth. The blue solid line is the solution obtained if all main peaks are used, representing a receiver without spoofing defense.

at every epoch. We can see the number of possible combinations vary between several hundred and 4096, depending on the number of tracked satellite signals. Up to the appearance of the first secondary peak around $210sec$ ($ds4$) and $190sec$ ($ds6$) we only compute a navigation solution for a single set of peaks. Once secondary peaks are visible, we can observe a reduction in sets of around 99% compared to the total number, causing us to use a $\log$ scale in the figure. The total number of least squares navigations solutions to be computed stays below 20 at all times during the two scenarios. In fact, we have purposefully chosen to only consider at most the 20 most probable sets at every epoch to demonstrate how few computations are necessary for a successful result.



(a) Scenario $ds4$.

(b) Scenario $ds6$.

Figure 6: The number of least squares solutions computed at each epoch as well as the total number of peak combinations possible at every epoch during scenarios $ds4$ and $ds6$. Depending on the number of tracked satellite signals, the total number varies between several hundred and 4096 combinations. Once secondary peaks are visible, the pruning heuristics reduces the total number by up to 99% with never more than 20 computed solutions.

3. Tracking of the authentic Solution

We can see in Figure 5 that we successfully identify sets of peaks that result in the authentic and the spoofed solution. We now present an EKF architecture coupling an IMU with the GNSS to track the antenna's state. We follow the error-state implementation in [24]. The filter's state $x \in \mathbb{R}^{17\times1}$ is given by

$$x = \begin{bmatrix} \Psi \\ \dot{r} \\ r \\ b_a \\ b_g \\ b_c \\ \dot{b}_c \end{bmatrix} \tag{14}$$

x contains the attitude in Euler angles $\Psi \in \mathbb{R}^{3\times1}$, the ECEF position and velocity $r \in \mathbb{R}^{3\times1}$ and $\dot{r} \in \mathbb{R}^{3\times1}$, accelerometer and gyro bias terms $b_a \in \mathbb{R}^{3\times1}$ and $b_g \in \mathbb{R}^{3\times1}$ as well as the clock bias and clock drift terms b_c and $\dot{b}_c$.

TEXBAT offers a clean version of the recordings, depicted in black in Figure 5. We consider this our "truth" and generate IMU measurements from a smoothed version of the clean data. We model the IMU measurements as

$$\begin{aligned} y_a(t) &= \ddot{r}(t) + b_a(t) + v_a \\ y_g(t) &= \dot{\Psi}(t) + b_g(t) + v_g \end{aligned} \quad \text{where} \quad \begin{aligned} v_a &\sim N(0, Q_a) \\ v_g &\sim N(0, Q_g) \end{aligned} \tag{15}$$

$\ddot{r}(t)$ and $\dot{\Psi}(t)$ are obtained from the clean data. The bias terms are described by first-order Gauss-Markov processes.

$$\begin{aligned} \dot{b}_a(t) &= w_a \\ \dot{b}_g(t) &= w_g \end{aligned} \quad \text{where} \quad \begin{aligned} w_a &\sim N(0, Q_{b_a}) \\ w_g &\sim N(0, Q_{b_g}) \end{aligned} \tag{16}$$

In the presented results we simulate a tactical grade IMU characterized by the root Power Spectral Densities (PSD)s in Table 1. The noise covariance matrices are diagonal, the variance terms are calculated by

$$\sigma_w^2 = \frac{S_w}{\tau} \tag{17}$$

where τ is the integration time interval and S_w is the PSD of the process.

Table 1: Tactical grade IMU model characteristics. S_a, S_g, S_{b_a} and S_{b_g} represent the PSDs of v_a, v_g, w_a and w_g respectively.

$\sqrt{S_a}$	$\sqrt{S_g}$	$\sqrt{S_{b_a}}$	$\sqrt{S_{b_g}}$
$100\mu g/\sqrt{Hz}$	$0.1°/\sqrt{h}$	$100\mu g/\sqrt{Hz}$	$2 * 10^{-6} rad\ s^{-0.5}$

We resume navigation with GNSS signals once at least 5 secondary signals are available. In the tested single constellation scenarios, this lets us evaluate the likelihood of the residual χ^2 statistic in Eq. (7) of a navigation solution using only secondary signals. We further want to resume satellite navigation only once we have reasonably converged to two consistent navigation solutions (authentic and spoofed). We therefore only resume using GNSS once the second most likely set has a likelihood of at least 0.01.

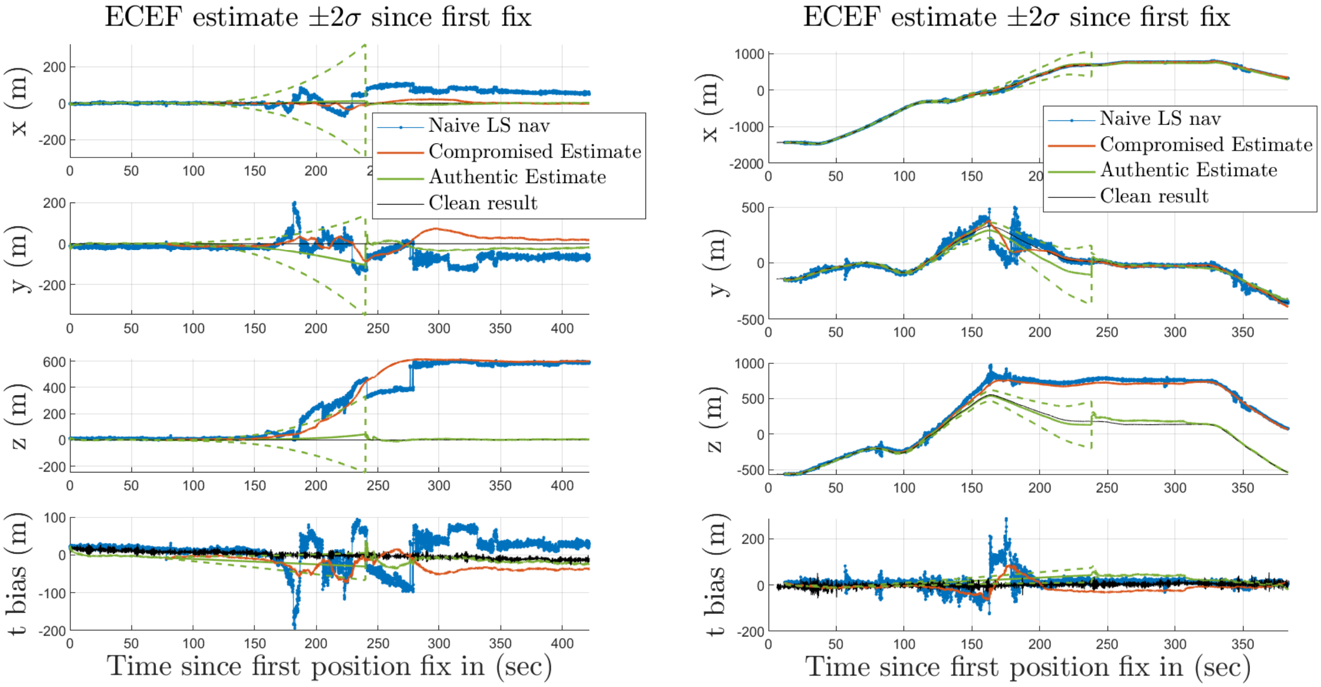
In Figure 7 we display results for both scenarios $ds4$ and $ds6$. The figures contain the smoothed clean "truth" solution in black, and the least squares navigation solution obtained by the SDR in blue. We further display the estimates of an EKF that continuously uses the GNSS measurements in red, as well as the EKF that takes full advantage of the work described in this paper and coasts on the IMU measurements while a spoofing alarm is active in green. It only resumes using GNSS measurements once the above mentioned criteria are fulfilled.

During $ds4$ the SDR solution (blue) does not represent the solution intended by the spoofer as is visible in the remaining offset in the ECEF x and y solution in Figure 7a. Due to its small power advantage, the attack did not capture all tracking loops of the receiver. The least squares solution is obtained from a mix of authentic and spoofed pseudoranges. The continuously tracking EKF (red) uses the most likely navigation solution based on the results presented in Figure 5 and therefore mostly follows the consistent solution intended by the spoofer. Small oscillations in the estimate visible in the x and y direction are likely due to the disagreement between IMU and spoofed GNSS measurements. In a deeper analysis this would show in large IMU bias estimates b_a and b_g.

The trusted, "authentic" EKF (green) shows the expected increase in position error and covariance during the coasting phase, but immediately tracks the correct solution once sufficient secondary peaks are visible. Errors in the estimate are almost all reflected in the inflated covariance. In the position domain, the effect of the spoofing attacks is not more than a $100sec$ lasting outage followed by multipath-like minor navigation errors.

The result for scenario $ds6$ is very similar. The spoofer now successfully captures all tracking loops resulting in a consistent solution. The continuously tracking, compromised EKF (red line) follows the spoofed GNSS signals closely. The trusted EKF (green line) once again coasts during the pull-off phase and tracks the authentic solution once secondary peaks are visible.

The results in Figure 7 further show why we chose the position push scenarios. With a well estimated clock bias and drift, we are able to maintain a fairly precise estimate of the time bias during the coasting phase. Differentiating between the authentic solution and a $600m$ offset in the time solution would not have been a major challenge.



(a) Scenario $ds4$.

(b) Scenario $ds6$.

Figure 7: Navigation solution from clean data (black, our "truth"), a least squares solution using all main peaks (blue), an EKF tracking continuously (red) and an EKF coasting during times that a spoofing alarm is active (green). Dashed lines represent the $\pm 2\sigma$ bounds on the EKF estimates. The continuously tracking solution is compromised, it follows the spoofed signals. We can see the increase in position and clock estimate uncertainty during the coasting phase of the "authentic" solution which successfully tracks the true solution.

The results presented in this subsection are heavily dependent on the position error and covariance at the end of the coasting phase. Is it too large, we can no longer successfully differentiate between the authentic and spoofed solution. In our simulations, using a consumer grade IMU for example resulted in too large covariance values. Using additional sensors such as magnetometers or obtaining forward velocity measurements from a pitot-static system on an aircraft or a car odometer on the other hand can greatly improve coasting performance. This could be further improved by adding vehicle specific motion constraints.
Any position uncertainty based on accelerometer measurements and attitude estimates will eventually grow at least quadratic with time, in our implementation using rate gyros it even grows with the 3rd power in time [24]. The duration of the pull-off between the first spoofing alarm and the appearance of secondary peaks therefore is critical to the procedure's success.

V. SUMMARY AND CONCLUSION

We have presented an approach to mitigate the effect of a spoofing attack. We assign two receiver channels to each PRN to track and decode both the authentic and spoofed signal. We identify consistent position solutions among the set of possible signal combinations. We then use a Multi-Hypothesis (E)KF for dead reckoning navigation during signal pull-off and to identify the authentic solution once at least 5 PRNs show two signals. We evaluate the procedure against static and dynamic position push scenarios from the TEXBAT dataset and successfully mitigate each attack using a tactical grade IMU to coast through the pull-off period. During all attacks the position error never exceeds a few 10s of meters.

The presented algorithm contains several hyper-parameters and heuristics mainly to reduce the computational load. The chosen parameters work very well in the presented scenarios, but should be validated against a wider range of scenarios and with real data.

Several of the presented techniques are heuristics. Further future work includes exploring alternatives, such as a tightly coupled EKF using pseudoranges directly instead of GNSS navigation solutions and a sensitivity study for the used hyper-parameters.

ACKNOWLEDGMENTS

The authors would like to thank Kazuma Gunning for his advice on GNSS-IMU filtering. His navsu Matlab toolbox further was a tremendously helpful resource during this project.

REFERENCES

[1] A. Jafarnia-Jahromi, A. Broumandan, J. Nielsen, and G. Lachapelle, "GPS vulnerability to spoofing threats and a review of antispoofing techniques," *International Journal of Navigation and Observation*, 2012.

[2] C. Günther, "A Survey of Spoofing and Counter-Measures," *Navigation, Journal of the Institute of Navigation*, 2014.

[3] M. L. Psiaki and T. E. Humphreys, "GNSS Spoofing and Detection," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1258–1270, 2016.

[4] Y. Oshman and M. Koifman, "Robust navigation using the global positioning system in the presence of spoofing," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 1, pp. 95–104, 2006.

[5] J. Blanch, T. Walter, P. Enge, Y. Lee, B. Pervan, M. Rippl, A. Spletter, and V. Kropp, "Baseline advanced RAIM user algorithm and possible improvements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 1, pp. 713–732, 2014.

[6] S. Daneshmand, A. Jafarnia-Jahromi, A. Broumandan, and G. Lachapelle, "A low-complexity GPS anti-spoofing method using a multi-antenna array," in *25th International Technical Meeting of the Satellite Division of the Institute of Navigation 2012, ION GNSS 2012*, vol. 2, Nachville, TN, 2012, pp. 1233–1243.

[7] A. Konovaltsev, M. Cuntz, C. Haettich, and M. Meurer, "Autonomous Spoofing Detection and Mitigation in a GNSS Receiver with an Adaptive Antenna Array," in *26th International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS+ 2013*. Nashville, TN: The Institute of Navigation, 2013, pp. 2937–2948.

[8] A. Broumandan, A. Jafarnia-Jahromi, and G. Lachapelle, "Spoofing detection, classification and cancelation (SDCC) receiver architecture for a moving GNSS receiver," *GPS Solutions*, vol. 19, no. 3, pp. 475–487, 2015.

[9] S. Han, L. Chen, W. Meng, and C. Li, "Improve the Security of GNSS Receivers Through Spoofing Mitigation," *IEEE Access*, vol. 5, pp. 21 057–21 069, 2017.

[10] F. Wang, H. Li, and M. Lu, "GNSS spoofing detection and mitigation based on maximum likelihood estimation," *Sensors (Switzerland)*, vol. 17, no. 7, 2017.

[11] Y. Guo, L. Miao, and X. Zhang, "Spoofing detection and mitigation in a multi-correlator gps receiver based on the maximum likelihood principle," *Sensors (Switzerland)*, vol. 19, no. 1, p. 17, 2019.

[12] T. E. Humphreys, J. A. Bhatti, D. Shepard, and K. Wesson, "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 2012, ION GNSS 2012*, Nashville, TN, 2012, pp. 3569–3583.

[13] L. Scott, "Anti-Spoofing and Authenticated Signal Architectures for Civil Navigation Systems," *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*, 2003.

[14] C. Hegarty, B. W. O'Hanlon, A. Odeh, K. Shallberg, and J. Flake, "Spoofing detection in GNSS receivers through cross-ambiguity function monitoring," in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS+ 2019*. Miami, Florida: Institute of Navigation, 2019, pp. 920–942.

[15] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements, and Performance; Revised Second Edition*, 2nd ed. Lincoln, Massachusetts: Ganga-Jamuna Press, 2011.

[16] B. Peterson, R. J. Hartnett, R. Fiedler, and A. Nebrich, “Frequency domain techniques for fast GPS acquisition and interference detection/rejection,” *Navigation, Journal of the Institute of Navigation*, 1996.

[17] D. J. Van Nee and A. J. Coenen, “New fast GPS code-acquisition technique using FFT,” *Electronics Letters*, 1991.

[18] E. G. Manfredini, “Signal processing techniques for GNSS anti-spoofing algorithms,” Ph.D. dissertation, POLITECNICO DI TORINO, 2017.

[19] E. G. Manfredini, D. M. Akos, Y.-H. Chen, S. Lo, T. Walter, and P. Enge, “Effective GPS Spoofing Detection Utilizing Metrics from Commercial Receivers,” in *Proceedings of the 2018 International Technical Meeting of The Institute of Navigation*, 2018, pp. 672–689.

[20] K. D. Wesson, J. N. Gross, T. E. Humphreys, and B. L. Evans, “GNSS Signal Authentication Via Power and Distortion Monitoring,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 2, pp. 739–754, 2017.

[21] M. Joerger, F. C. Chan, and B. Pervan, “Solution separation versus residual-based RAIM,” *Navigation, Journal of the Institute of Navigation*, vol. 61, no. 4, pp. 273–291, 2014.

[22] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. Cambridge, Massachusetts: The MIT Press, 2005.

[23] J. Blanch, T. Walter, and P. Enge, “Fast multiple fault exclusion with a large number of measurements,” in *Institute of Navigation International Technical Meeting 2015, ITM 2015*, 2015, pp. 696–701.

[24] P. D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 1st ed. Boston, NY: Artech House, 2008.

[25] R. E. Phelts, “Multicorrelator Techniques for Robust Mitigation of Threats To GPS Signal Quality,” Ph.D. dissertation, Stanford University, 2001.

[26] C. Sun, J. W. Cheong, A. G. Dempster, H. Zhao, and W. Feng, “GNSS spoofing detection by means of signal quality monitoring (SQM) metric combinations,” *IEEE Access*, vol. 6, pp. 66 428–66 441, 2018.

[27] A. Broumandan, A. Jafarnia-Jahromi, G. Lachapelle, and R. T. Ioannides, “An approach to discriminate GNSS spoofing from multipath fading,” in *2016 8th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC 2016*, 2016, pp. 1–10.

[28] C. Hegarty, A. Odeh, K. Shallberg, K. D. Wesson, T. Walter, and K. Alexander, “Spoofing Detection for Airborne GNSS Equipment,” in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2018)*, Miami, Florida, 2018, pp. 1350–1368.

[29] M. Troglia Gamba, M. D. Truong, B. Motella, E. Falletti, and T. H. Ta, “Hypothesis testing methods to detect spoofing attacks: a test against the TEXBAT datasets,” *GPS Solutions*, vol. 21, no. 2, pp. 577–589, 2017.

[30] E. G. Manfredini, F. Dovis, and B. Motella, “Validation of a signal quality monitoring technique over a set of spoofed scenarios,” in *2014 7th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC 2014 - Proceedings*. Noordwijk, Netherlands: IEEE, 2015, pp. 1–7.

[31] A. Broumandan, R. Siddakatte, and G. Lachapelle, “Feature article: An approach to detect GNSS spoofing,” *IEEE Aerospace and Electronic Systems Magazine*, pp. 64–75, 2017.

[32] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I*. New York: John Wiley & Sons, Inc., 2001.