Combinatorial Watermarking for GNSS Signal Authentication

Jason Anderson, Sherman Lo, Todd Walter

Abstract

Watermarking Signal Authentication can establish trust in satellite navigation signals by cryptographically perturbing the spreading code. In previous work, we propose and derive probability distributions that predict receiver statistics for use in scheme design. In this work, we design a SBAS watermarking signal authentication scheme that can afford 32-bit authentication security over a 6-second time interval, matching the cadence of SBAS navigation message authentication proposals and providing 6-12 second time to authentication. The scheme is immediately extendable to other GNSS services, provided the receiver is observing SBAS signals or is network connected. We design with parameters to match worst-case operating conditions and minimum receiver hardware. We validate our models via Monte Carlo simulation and experiments with real WAAS observation and a software-defined radio to support our scheme proposals.

I. INTRODUCTION

Receivers can use Watermarking Signal Authentication to establish trust in the satellite navigation ranging signals [Scott, 2003]. Several proposals and studies exist to augment GNSS signals with this technique, including for the Global Positioning Service (GPS) [Anderson et al., 2017, Hinks et al., 2021], and for the Wide Area Augmentation Service (WAAS) [O'Hanlon et al., 2022]. These techniques work together with Navigation Message Authentication to establish trust in transmitted navigation data. In this work, we design a Combinatorial Watermarking scheme for WAAS.

In our previous work, we created Combinatorial Watermarking [Anderson et al., 2023b]. Combinatorial watermarking refers to how the watermark is constructed. A combination of chips from the spreading code is pseudorandomly selected to be inverted to form the watermark. With this construction, the scheme can be flexible and provide a fixed degradation. We can parameterize the duty cycle of the watermark, apply the same security analysis to different signals, and compute in the probability distribution of induced measurements on the receiver. From our derivations, we can design the scheme and the radio together.

Combinatorial watermarking poses almost no additional burden on the data bandwidth of the current signal if Timed Efficient Stream Loss-tolerant Authentication (TESLA) is used [Anderson et al., 2022a]. The watermarks derive from a one-way function from TESLA distributions. The one-way function uses a series of Hash-based Message Authentication Code (HMAC) calls as a Key-Derivation Function (KDF) to generate pseudorandom data. The pseudorandom data is used to select the particular chips inverted in the watermark. By using our construction, the watermark can be derived from information already broadcast and observation of the watermark does not break the security of other aspects of the signal (e.g., Navigation Message Authentication).

We designed this signal authentication scheme to accommodate potential Federal Aviation Administration (FAA) and receiver concerns in this work. For instance, the FAA may desire to minimize the inverted duty cycle to ensure the fidelity of the signal at the WAAS service volume boundaries. Or they may desire to specify a signal authentication time-to-alert requirement. Receiver manufacturers may want to minimize the memory hardware requirements necessary to make an authentication determination confidently. The scheme is also designed to accommodate low-cost receivers.

We provide results for a representative breadth of signal conditions and radios. Our radio specifications are representative of radios currently used in commercial aviation. We verify our predictions with Monte Carlo simulation and by observing a mirrored situation with real WAAS signals with a software-defined radio. The mirrored situation is as follows. In a real scheme, the receiver uses the normal replica to track a watermarked signal. In our mirrored situation, the receiver uses a watermarked replica to track the normal signal. The resulting observables are the real observables negated under certain conditions.

From the results, we validate the computed probability distribution predictions of our mathematical derivations and the potential of receivers to provide this service as specified. We find a scheme that minimally affects current signals while providing a helpful signal authentication service. Should the FAA desire to augment WAAS with signal authentication, this work will provide a preliminary scheme design study. Moreover, aircraft would have some signal authentication security on their vertical protection levels derived from WAAS ranging.

1. SBAS Message Authentication

An SBAS authentication scheme is currently being considered based on TESLA [Anderson et al., 2023a]. The scheme exploits the data-efficient and loss-tolerant properties to allow message authentication without materially affecting performance require-

Table 1: A Table defining the variable notation for Combinatorial Watermarking.

Variable	Definition
n	The number of chips in a single watermark. For SBAS, $n = 1023$.
r	The number of chips inverted in a single watermark. For SBAS, r can vary to meet specific design concerns.
s	The number of chips an adversary may elect to invert when attempting to spoof a receiver. s may be any integer from 0 to $n/2$.
$\mathcal{H}(n, r, s)$	The Hypergeometric Distribution. An adversary engaged in a spoofing attack generating false signals with s randomly selected chips inverted will guess $h \sim \mathcal{H}(n, r, s)$ correctly for any one watermark.
R	The spreading code replica. R^w refers to the watermarked replica. R_{-} refers to the reversed replica, which is convolved with the signal to enact correlation.
H	The number of individual watermarks over R.
F	The sampling rate of the radio measuring the receiver observable used to determine the authenticity of a watermarked signal. For SBAS, this should be greater than 2 MHz.
T	The coherent integration of a single watermark measurement. In this work, we use T = 1ms.
P	The power of the signal within a receiver radio immediately proceeding correlation.
σ^2	The noise power within a receiver radio immediately proceeding correlation.
$\mathcal{N}$	The normal distribution.
S	The signal measured over time T at sampling rate F immediately proceeding correlation.
$Y, \mathcal{Y}$	The receiver observable statistic and its distribution, respectively.
$y = g(h \mid n, r, s)$	The linear function g transforms the support of $h \sim \mathcal{H}$ into the support of the radio observable Y.

ments[CITE TODD]. The scheme requires adding two message types (MT) to the schedule. For L1, these are MT20 and MT21. For L5, these are MT50 and MT51.

MT20 and MT50 deliver the TESLA delayed key distributions and HMACs of the normal SBAS messages. These messages occur at a rigid frequency of one every six seconds, except during SBAS alerts. MT21 and MT51 deliver the key maintenance information for the TESLA instance. These messages fit in the remaining empty message slots of the message scheduler.

Within the MT20 and MT50, the TESLA delayed distribution is called a Hash Point in [Anderson et al., 2023a], referring to the potential of complicated geometries to derive pseudorandom data. A KDF, such as HMAC-KDF (HKDF), can generate additional pseudorandom data that can be used for watermarking. Provided the derived pseudorandom data is used before generating the Hash Point is released, that derived data can be used to provide watermarks with authentication security.

2. Combinatorial watermarking

In this section, we provide an introduction to Combinatorial Watermarking. We refer to our previous work for additional details, including the mathematical and cryptographic derivations [Anderson et al., 2023b]. For the reader's convenience, Table 1 includes the variable notation definitions.

Our Combinatorial Watermarking construction ensures that the selection of chips is uniform and one-way. Uniformity ensures that the selection probability of any chip is unbiased. The one-way cryptographic pseudorandom construction admits no efficient algorithm to predict the underlying randomness. Together, Combinatorial Watermarking admits no efficient algorithm to predict the inverted chips in a watermark before observation.

To determine the authenticity of a watermarked signal, the receiver must construct a radio observable. In our previous work, we suggested the receiver statistic in Figure 1 because we can compute the probability distributions of the statistic output in nominal and spoofing conditions. After the delay prescribed by TESLA, the watermark seed is delivered to the receiver. The receiver can compute R^w from the watermark seed to compute Y. From our previous work, with our derivation of g to Equation (1) and our judicious selection of K in Equation (2), we arrive at Equations (3) and (4) for the distribution $\mathcal{Y}$ under the authentic and

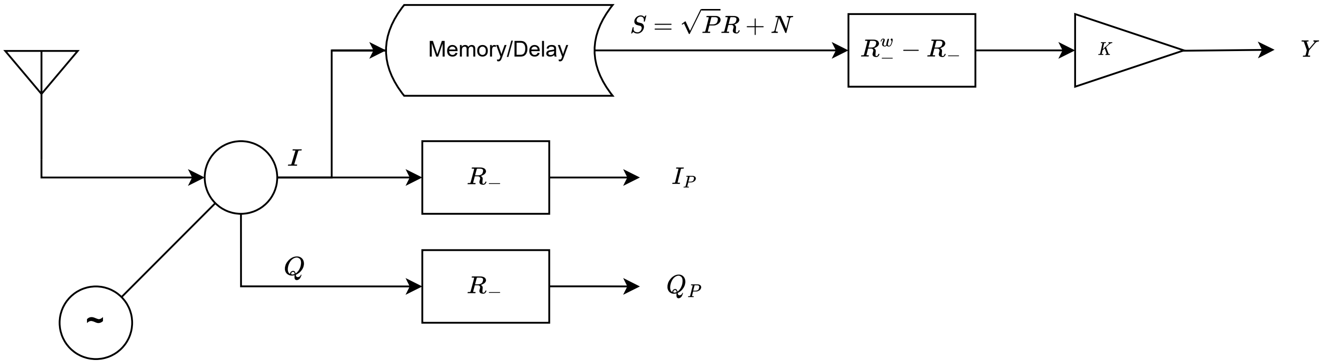

Figure 1: A radio diagram that includes the typical GNSS tracking loop and a new branch used to determine the authenticity of a watermarked signal. The delay results from the receiver's inability to know R^w until after the corresponding TESLA Hash Point distribution. After that distribution, the receiver may compute Y and dump the stored memory.

spoofing hypotheses.

$$y = g(h, n, r, s) = \frac{1}{n}\left(2h - r\right) \tag{1}$$

$$K = \frac{1}{2}\frac{1}{\sqrt{P}}\frac{1}{||R||_1} \tag{2}$$

$$\mathcal{Y} \mid \text{authentic}, H = \mathcal{N}\left(\frac{r}{n}, \frac{r}{n}\frac{\sigma^2}{P}\frac{1}{FTH}\right) \tag{3}$$

$$\mathcal{Y} \mid \text{spoof}, H = \sum_{H} g(\mathcal{H}(n, r, s), n, r, s) + \mathcal{N}\left(0, \frac{r}{n}\frac{\sigma^2}{P}\frac{1}{FT}\right) \tag{4}$$

$$\text{PDF}_{\mathcal{Y}|\text{spoof, } H}(y) = \text{PDF}_{\mathcal{H}(n,r,s)}(g^{-1}(y, n, r, s))^{*H} * \text{PDF}_{\mathcal{N}\left(0, \frac{r}{n}\frac{\sigma^2}{P}\frac{1}{FTH}\right)}(y) \tag{5}$$

From the distributions from Equation (3) and (4) (the PDF of Equation (4) is Equation (5)), one can select a boundary on Y and the scheme parameters to achieve desired probabilities of missed detection and false alarm.

II. WATERMARKING DESIGN FOR SBAS

While we have derived the math to determine the probability of missed detection and false alarm for a watermark and the accompanying receiver, technical and political externalities influence numerous complicated parameter trades. Many of these externalities remain unknown as a watermarking scheme approaches implementation. This limits the rigor of our approach to presenting a scheme, making arguments on its parameter selection, and meeting reasonable and proper requirements.

Section II.1 describes using cryptography so that any watermarking scheme has authentication security under TESLA. Section II.2 provides two parameter selection suggestions and our reasoning behind them. Section II.3 includes a study when perturbing our parameter selection and the effect on the performance indicators.

1. Cryptographic Construction

To exploit TESLA's data efficiency property to authenticate yet another object in the signal, we must introduce another branch of the Hash Path. In the schemes of Section II.2, there is a unique watermark for each spreading code symbol. Therefore, we must branch the TESLA Hash Path an additional 1000 times per second via HKDF.

In [Anderson et al., 2023a], Equation (1) describes how the provider generates a TESLA Hash Path p_i^P for authentication, and Equation (2) describes how to generate arbitrary numbers of signing keys for authentication items. In this work, Equation (6) extends this procedure for an SBAS watermark.

$$p^P_{t_j,w} = \text{HMAC}(p^P_i, \text{“Watermark Seed”}||\text{PRN}||\text{Frequency}||t_j||t_{\text{millisecond}}) \tag{6}$$

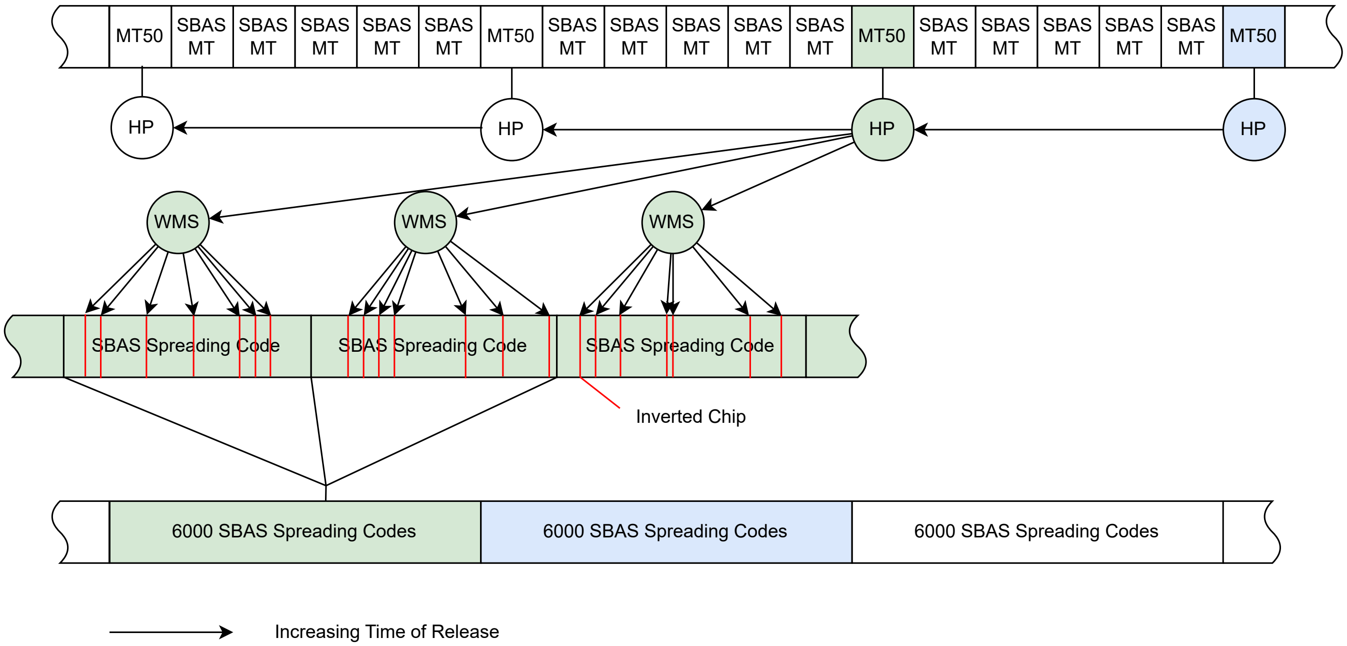

Figure 2: A conceptual diagram of an SBAS watermark requiring little or no additional bandwidth on the data. HP stands for Hash Point, following the terminology for an SBAS TESLA proposal [Anderson et al., 2023a]. WMS stands for Watermark Seed. Equation (6) determines a WMS from a HP. The colors indicate which watermarks are derived from which HP each six seconds, following the distribution from MT50. Items to the right are released after items on the left. Each arrow represents a cryptographic one-way operation.

Equation (6) uses HKDF to generate all the necessary pseudorandom bits for watermark seeds. The HMAC key field within Equation (6) is the applicable Hash Point and provides the 128-bit authentication security on the operation provided the disclosure delay adherence of TESLA. The HMAC message field within Equation (6) ensures that each branched watermark seed $p^P_{j,w}$ is cryptographically independent. Here, P indicates the specific TESLA Hash Path, t_j indexes time in seconds, and w indexes the millisecond (we note that P here does not relate to signal power as this was adapted from the notation of [Anderson et al., 2023a]). Each watermark will be cryptographically unique, provided the message field is unique. Within the message field, we include the subject "Watermark Seed" to differentiate the subject from other derived pseudorandom bits (e.g., keys signing other portions of the SBAS message scheme). Equation (6) concatenates bits presenting the PRN, frequency, and time to the millisecond to ensure unique watermarks for every satellite, frequency, and SBAS symbol, respectively. In [Anderson et al., 2023b], Algorithms (1) and (2) describe how to take watermark seed $p^P_{j,w}$ and compute the appropriate SBAS spreading code chips to invert.

From a hardware perspective, the hashing required per watermark is about four hashes per SBAS spreading code or a hashing rate of about 4000 per second per signal. We note that the construction of Equation (6) allows for the process to be done in parallel, per watermark, and as described in later sections, a receiver may not need to observe every watermark to achieve the needed security level. Moreover, our laptop hardware-accelerated tests of OpenSSL exceed this requirement by several orders of magnitude.

Given the TESLA distribution cadence every six seconds, a watermark seed distribution cadence of six seconds is convenient and the best possible in-band. Figure 2 provides a conceptual diagram of the temporal and geometric relations of the SBAS watermark and the TESLA distributions. Like with the SBAS data authentication, the minimum time to authentication ranges from 6-12 seconds, depending on the segment's relation to the 6-second TESLA distribution cadence. The minimum of 6 seconds follows a TESLA time synchronization requirement of 6 seconds [Anderson et al., 2022b]. In Figure 2, the green and blue colors correspond to what TESLA Hash Point applies to which sections of spreading code. Each arrow is a cryptographic one-way operation, where the arrows from Hash Point (HP) to Watermark Seeds (WMS) correspond to Equation (6) and never collide due to the unique HMAC message field and the HKDF-provided security.

2. Parameter Selection

From the construction in Section II.1 and Algorithms (1) and (2) of [Anderson et al., 2023b], watermarks can derive whatever cryptographic pseudorandom bits are required. We now have abstracted the cryptographic construction considerations in favor of the selection of parameters in Table 1. For most of the parameters, we rely on judicious selection based on conservative

Table 2: The selection of r for two suggested schemes with two different assumptions on the worst operating condition of *any* receiver at 32-bit security. Figure 3 shows the missed detection and false alarm probabilities over a decision boundary on Y from Equations (3) and (4).

Worst Operating Assumption	$\frac{P}{\sigma^2} = 2\frac{C}{N_0}\frac{1}{F}$	r	Power Loss
C/N$_0$ = 30 dB-Hz	$1 \cdot 10^{-3}$	15	-0.129 dB
C/N$_0$ = 25 dB-Hz	$3.1 \cdot 10^{-4}$	50	-0.447 dB

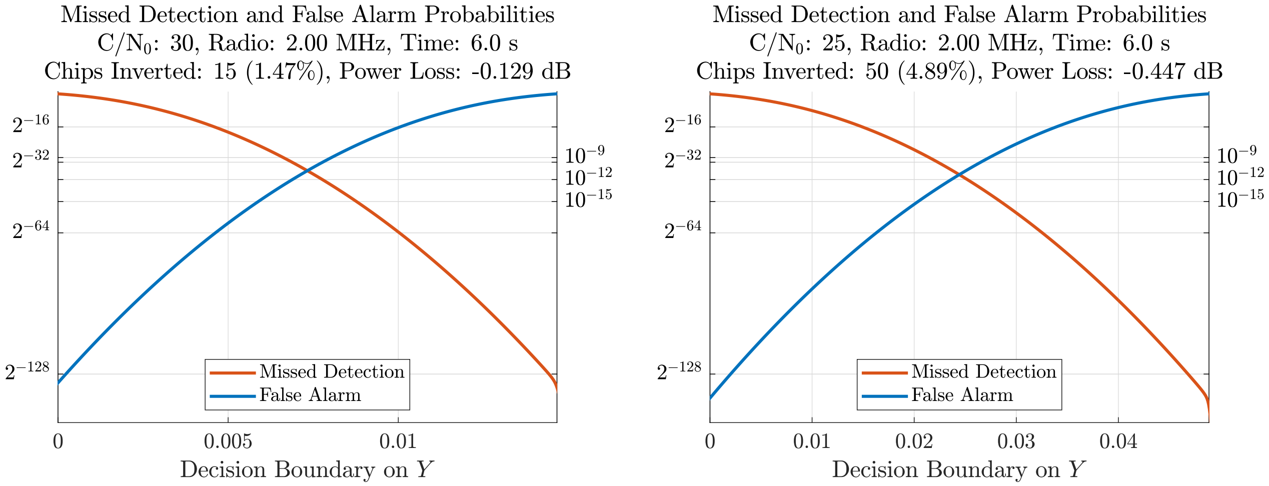

Figure 3: Probabilities of missed detection and false alarm over the selection on the decision boundary on Y for both of the schemes in Table 2 from Equations (3) and (4).

assumptions of the receiver and the receiver's observations and based on security and performance requirements.

First, let us consider performance requirements. We elect to maintain the same requirement as the SBAS data authentication for time to authentication because of the convenience afforded by the 6-second in-band TESLA distribution cadence. From the afforded cadence, we assume that a receiver may use 6 seconds of data to make an authentication determination to inform a *worst-case* design. We refer to Section II.3 for receivers with a better sense of their operation conditions. For probabilities of missed detection and false alarm, we follow [Neish et al., 2020]. A missed-detection probability of 10^{-9} is bounded above by 2^{-32}, corresponding to a watermark that provides 32-bit security over 6 seconds, matching the security level afforded to the data messages. A false-alarm probability of 10^{-9} yields an acceptable false alarm expectation about once every other century.

Second, let us consider some assumptions to fix some of our parameters. We elect to have a unique watermark for each SBAS spreading code and to allow any chips to be inverted because we can, and the scheme would be simple to specify and explain. Therefore, $n = 1023$, but this could be adjusted to meet concerns after repeating the analysis below (e.g., having a watermark span of ten spreading codes). The selection of r will be reserved until last, and we suggest that an implementing system allow that parameter to be changed with notice to receivers. Pursuant to making conservative assumptions about receiver operating conditions, our analysis assumes the sampling frequency $F = 2$ MHz. This selection fulfills our desire for our analysis to be valid for *any* receiver since $F = 2$ Mhz is just under the Nyquist Frequency. Perturbing F among a reasonable radio range from 2 to 20 MHz produces the least significant change to performance indicators among the other scheme parameters, so we keep $F = 2$ MHz in this analysis. Our selection of a coherent integration time of $T = 1$ ms is commensurate with aviation receivers under high dynamic conditions: a receiver will need to determine authenticity among the 6000 measurements each 6 seconds, and therefore, $H = 6000$. To determine noise assumptions for P and σ^2, we now judiciously branch to provide two suggested schemes. Our suggestions are as follows. We assume that a receiver would not attempt to track a signal below a C/N$_0$ (1) 30 dB-Hz or (2) 25 dB-Hz. From there, we determine r in Table 2 that meet our requirements via an iterative approach and Equations (3) and (5) assuming a halfway decision boundary on Y. While it would be possible to select a different boundary on Y to accommodate vastly different missed detection and false alarm probability requirements, we use a halfway decision boundary on Y to afford approximately equivalent missed detection and false alarm probability requirements for simplicity. Figure 3 presents this trade off for both schemes presented in Table 2 and shows that the a boundary of $y = \frac{r}{2n}$ meets our probability requirements.

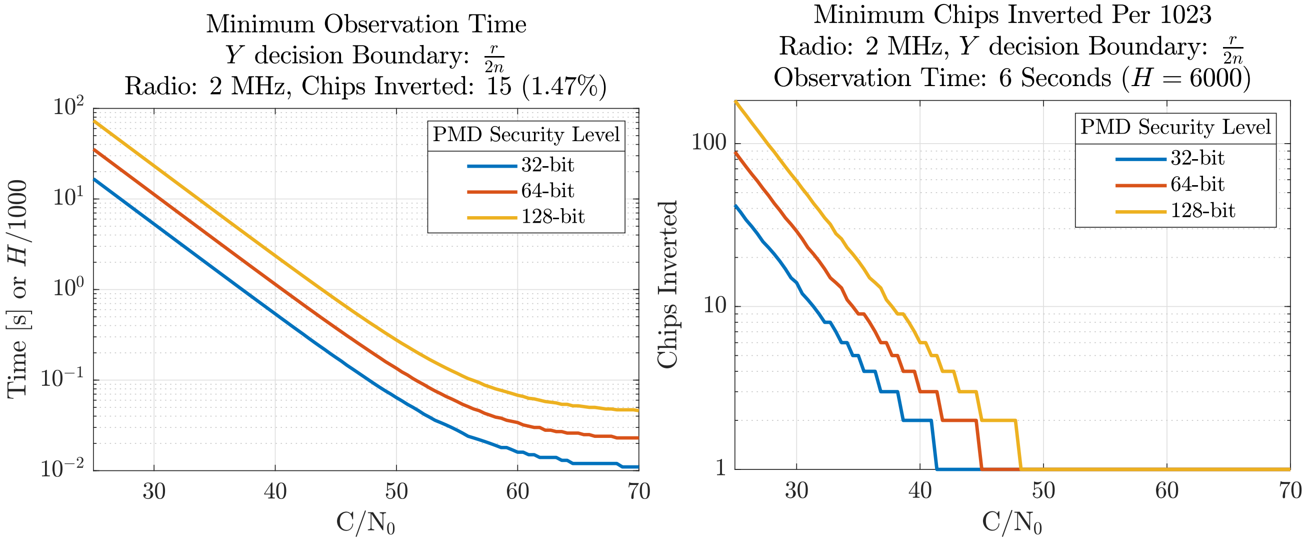

Figure 4: In the left figure, we use a binary search on the upper-tails of Equation (5) to determine the length of an observation required for a security level given a trusted lower-bound C/N$_0$ indicator. We note that the 32-bit security line is just under 6-seconds at 30 C/N$_0$, matching Figure 3 left with 15 chips inverted. In the right figure, we use binary search on the upper-tails of Equation (5) to determine the minimum number of chips inverted required for a security level given a trusted lower-bound C/N$_0$ indicator. We note that the 32-bit security line is just under 6 seconds (left) and 15 chips (right) at 30 C/N$_0$, showing how Figure 3 relates to Table 2.

3. Perturbing Noise Model Parameters

We present proposed parameter selections in Section II.2. Our selection of low C/N$_0$ ensured that the scheme's security was enforced in *any* reasonable condition. However, this lends another question of whether a receiver with a better lower-bound confidence of C/N$_0$ can use a more favorable scheme. That receiver must *trust* a lower bound on C/N$_0$, which we do not address in this work. Provided a higher trusted lower-bound C/N$_0$, the receiver would be able to determine signal authenticity with considerably fewer data and fewer chips inverted. Figure 4 presents this trade.

In Figure 4 left, we observe the relationship between C/N$_0$ and the amount of observation required to determine authenticity. The x-axis is C/N$_0$. The y-axis is the length of aggregate observation required, assuming individual observations of Y over 1 ms coherent integrations. To compute these curves, we used a binary search to compute the H from Equation (5) that yielded an upper-tail area above $y > \frac{r}{2n}$ less than the security level. At C/N$_0$ = 30, we observe between 5 and 6 seconds, or between 5000 and 6000 1ms observations, matching Figure 3 left. We observe that in order to have a higher security level than 32 bits, the receiver must observe over a longer period of time or have a higher trusted lower-bound C/N$_0$ measurement or requirement. Moreover, the log-log relationship breaks down as time approaches 10^{-1}, corresponding to under 100 observations and when there are not enough repeated convolution from Equation (5) to observe Central Limit Theorem effects.

In Figure 4 right, we observe the relationship between C/N$_0$ and the number of chip inversions in the scheme. The x-axis is C/N$_0$. The y-axis is the minimum number of chip inversions among the 1023 per millisecond to meet a security level. To compute these curves, we used binary search to compute r from Equation (5) that yielded an upper-tail area $y > \frac{r}{2n}$ less than the security level assuming six seconds of observation. Increasing the security level requires more chip inversions or a higher trusted lower bound on C/N$_0$. However, once the signal strength is sufficiently high, two chips are sufficient for each authentication security level within the figure.

In Figure 4, we assumed a decision boundary at $\frac{r}{2n}$. This corresponds approximately to where the missed detection and false alarm probabilities are the same. However, repeating this analysis with a different boundary on Y would be possible. This would be appropriate when the false alarm requirement is not as strong as the missed detection requirement, like when a scheme requires 128-bit security but a 10^{-9} false alarm probability.

Figure 5 presents the tradeoff between chips inverted, minimum observation time, and C/N$_0$, given a fixed security level. As the number of chips inverted increases, the minimum observation time decreases, or a lower C/N$_0$ can be tolerated. As with the other figures, the curves were computed via binary search on Equation (5).

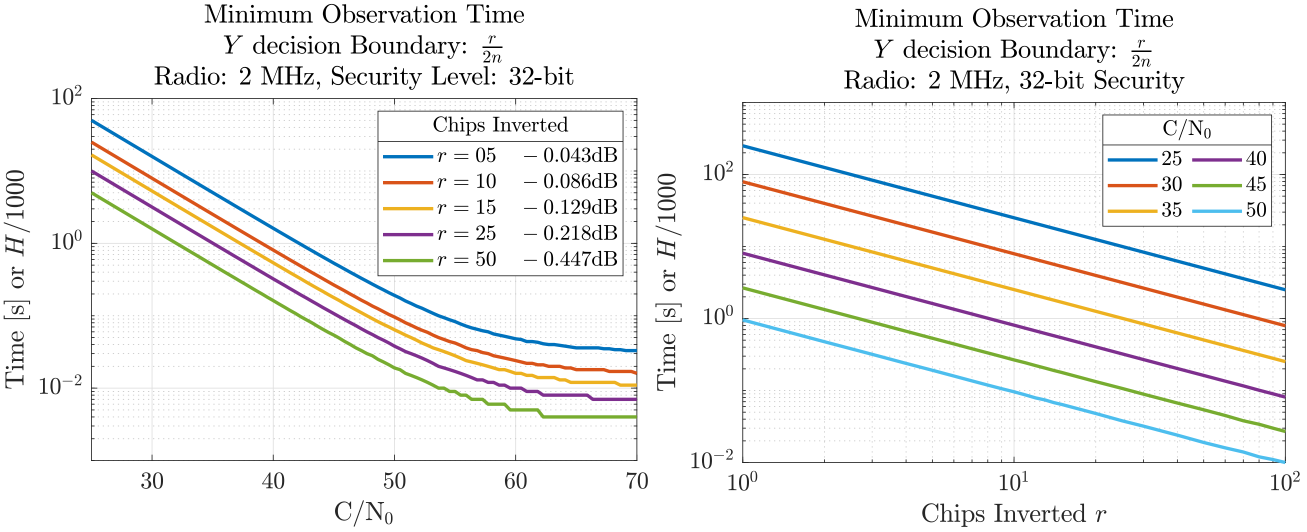

Figure 5: In both figures, we use a binary search of the upper-tails of Equation (5) to determine the length of an observation required for 32-bit security given the number of chips inverted and C/N$_0$. In the left figure, the isolines result form the chips inverted; whereas, in the right figure, the isolines form constant C/N$_0$.

III. MODEL VALIDATION

To validate the probability distributions from Equations (3) and (5), we devise (1) a set of Monte Carlo experiments and (2) an experiment with a WAAS software-defined radio observing WAAS signals.

1. Monte Carlo Validation

For our Monte Carlo validation, we built software that simulates $\mathcal{Y}$ under spoofing and authentic conditions. For each Monte Carlo experiment, each with a different simulated C/N$_0$, the following was repeated 10^6 times. The simulation generates 6000 WAAS PRN 131 spreading codes and randomly inverts $r = 15$ chips in each spreading code. To simulate an adversary-generated spreading code with the best $s = 511$ spoofing strategy, 6000 random sequences of 1023 bits are also generated. The original spreading code, watermarked spreading code, and adversary-generated spreading code was then resampled to have 2000 samples corresponding to 1 ms at $F = 2$ MHz to create R, R^w, R^a, respectively. Power and noise corresponding to the experiment C/N$_0$ value were added to R^w and R^a to complete the simulated signals S^{auth} and S^{spoof}. Then, we computed the average $y|\text{auth} = K \cdot (R^w - R) \cdot S^{\text{auth}}$ and $y|\text{spoof} = K \cdot (R^w - R) \cdot S^{\text{spoof}}$ among the 6000 to form a single experimental result. A single experiment with 10^6 trials, each with 6000 spreading codes, corresponding to $6 \cdot 10^9$ ms or about 70 days of spreading code, took about 6 hours real time on a Matlab instance with 16 parallel computing workers.

Among the 10^6 trials, the number of missed detections and false alarms divided by the number of experiments forms an unbiased estimator of the probabilities of missed detection and false alarms, respectively. Given the nature of the estimator, we can apply the Central Limit Theorem to form confidence bounds on the estimated probabilities. Table 3 provides the Monte Carlo results with 3-sigma (99.7%) confidence bounds. In each case, the predicted probabilities from Equations (3) and (5) fall within the Monte Carlo results. Our selection of C/N$_0$ for experiments was to ensure some observation of rare events and show the efficacy of our model in support of designs assuming a trusted, worst-case C/N$_0$. In the case with C/N$_0$ of 30, we return to the first scheme of Table 2, which has adverse-event probabilities less than 10^{-9}, where expect no missed detections or false alarms at 10^6 trials.

2. WAAS SDR

For our experiment, we construct a mirrored problem that can incorporate measurements of WAAS from [Anderson et al., 2023b]. At the time of writing, no signals incorporated watermarks. In a real scheme, the signal spreading code is watermarked, and a receiver tracking loop uses the original spreading code. In our mirrored experiment, the signal spreading code is the original spreading code, and the receiver tracking loop uses a watermarked spreading code [Bernabeu et al., 2022]. The result is that we can experimentally measure a negated Y with current signals.

To simulate the spoofed situation, we had the Y filter use two watermarked replicas. The first was one random watermarking function application to the original spreading code to simulate the authentic case. The second was an additional random

Table 3: Results from multiple Monte Carlo simulations to validate the models of Equations (3) and (5) under $n = 1023$, $r = 15$, $s = 511$, $F = 2$ MHz, $T = 1$ ms, $H = 6000$. The confidence intervals reported are the 3-sigma (99.7%) intervals after applying the Central Limit Thoerem using the mean and standard deviation divided by the square root of the number of experiments. With C/N$_0$ of 30, we return to the first scheme suggested in Table 2 with adverse event probabilities less than the 2^{-32} and 10^{-9} requirements. And given that we simulated 10^6 times, we expect not to observe any missed detections or false alarms.

C/N$_0$	Monte Carlo PFA	Equation (3) PFA	Monte Carlo PMD	Equation (5) PMD
30	None Observed	$1.65 \cdot 10^{-11}$	None Observed	$1.67 \cdot 10^{-11}$
25	0.00011 ± 0.00003	0.00010	0.00010 ± 0.00003	0.00009
20	0.0180 ± 0.0004	0.0180	0.0178 ± 0.0004	0.0178
15	0.1199 ± 0.0010	0.1191	0.1188 ± 0.0010	0.1187
10	0.2533 ± 0.0013	0.2532	0.2536 ± 0.0013	0.2536

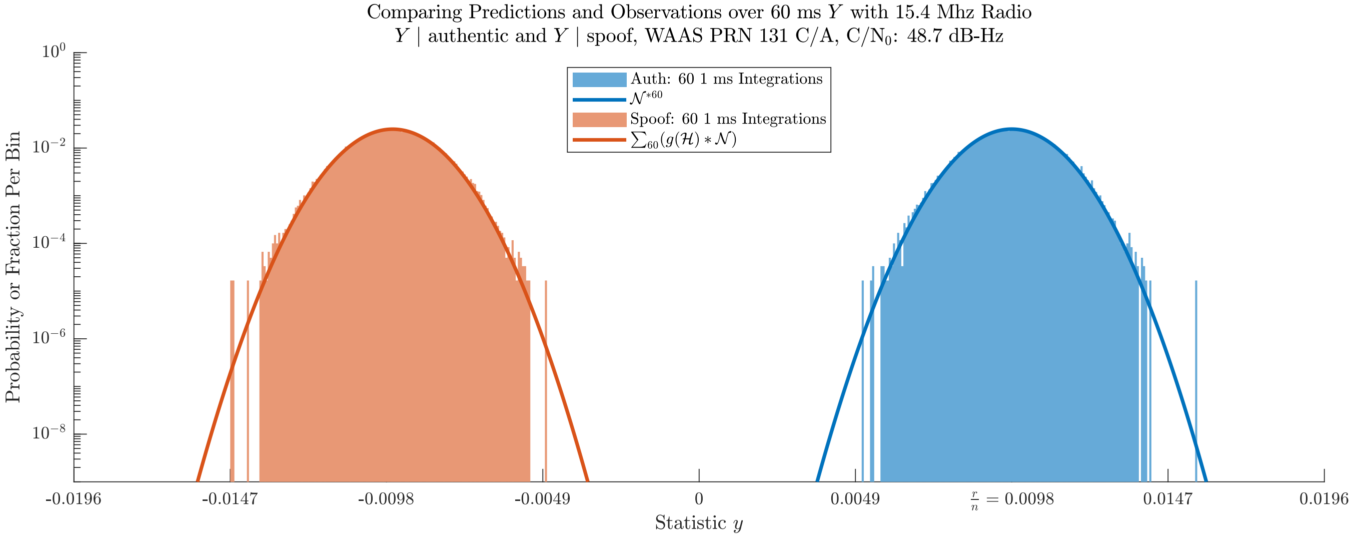

Figure 6: A comparison of our predicted distributions with post-process WAAS PRN 131 C/A observations made with a 15.4 MHz software-defined radio based on the mirrored experiment discussed in Section III.2. Over a 1-hour timer period, we measure 60 ms Y. Each Y included 60 1-ms coherent integrations via a converged tracking loop. Each millisecond contains its own watermark. The predicted distributions are computed via repeated convolution. In this scenario, the adversary and provider elected to invert $r = s = 10$ chips.

watermarking function application of the first watermarked replica to simulate the spoofing case. Provided both watermark function applications had the same number of inversions (i.e., $r = s$), we have the negated Y for the spoofing situation. In our results for the spoofing and authentic scenarios, we compensate for the mirrored situation by negating K within the filter.

Our data was taken on July 12, 2023, at 15.4 MHz with a USRP N310 over one hour. We used a value of $H = 60$ rather than the proposed $H = 6000$ to have more samples for comparison against our model and a better histogram. Figure 6 provides a comparison histogram of the two scenarios and demonstrates that our model can predict the center and spread well with actual radio data.

IV. CONCLUSION

In this work, we use our watermark model to suggest an SBAS signal authentication scheme compatible with an SBAS TESLA scheme. We devise two SBAS signal authentication schemes capable of 32-bit authentication security, 6 to 12 second time to authentication, and little degradation to the existing signal. Among the two schemes suggested, the difference lies with the trusted lower bound on the C/N$_0$ and the number of chips inverted per spreading code. We use our models to perturb our suggested scheme parameters to show trends that aid a watermarking signal authentication scheme parameter selection. Moreover, we validate our model using simulation and WAAS experimentation to establish confidence our methods. While SBAS presents a convenient in-band TESLA distribution option, the scheme can be applied to GNSS more broadly provided the receiver is network connected to receive the 6-second distributions.

REFERENCES

[Anderson et al., 2023a] Anderson, J., Lo, S., Neish, A., and Walter, T. (2023a). Authentication of Satellite-Based Augmentation Systems with Over-the-Air Rekeying Schemes. *NAVIGATION: Journal of the Institute of Navigation*, 70(3).

[Anderson et al., 2022a] Anderson, J., Lo, S., and Walter, T. (2022a). Efficient and Secure Use of Cryptography for Watermarked Signal Authentication. In *Proceedings of the 2022 International Technical Meeting of The Institute of Navigation*, pages 68–82.

[Anderson et al., 2022b] Anderson, J., Lo, S., and Walter, T. (2022b). Time Synchronization for TESLA-based GNSS-enabled Systems. In *Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022)*, pages 3408–3417.

[Anderson et al., 2023b] Anderson, J., Lo, S., and Walter, T. (2023b). Authentication Security of Combinatorial Watermarking for GNSS Signal Authentication. In *Proceedings of the 36th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2023)*, pages 495–509.

[Anderson et al., 2017] Anderson, J. M., Carroll, K. L., DeVilbiss, N. P., Gillis, J. T., Hinks, J. C., O'Hanlon, B. W., Rushanan, J. J., Scott, L., and Yazdi, R. A. (2017). Chips-message robust authentication (chimera) for GPS civilian signals. pages 2388 – 2416.

[Bernabeu et al., 2022] Bernabeu, J., Palafox, F., Li, Y., and Akos, D. M. (2022). A collection of SDRs for global navigation satellite systems (GNSS). In *Proceedings of the 2022 International Technical Meeting of The Institute of Navigation*, pages 906–919.

[Hinks et al., 2021] Hinks, J., Gillis, J. T., Loveridge, P., Miller, S., Myer, G., Rushanan, J. J., and Stoyanov, S. (2021). Signal and Data Authentication Experiments on NTS-3. pages 3621–3641.

[Neish et al., 2020] Neish, A., Walter, T., Powell, J., and of Aeronautics & Astronautics, S. U. D. (2020). *Establishing Trust Through Authentication in Satellite Based Augmentation Systems*. Stanford University.

[O'Hanlon et al., 2022] O'Hanlon, B., Rushanan, J. J., Hegarty, C., Anderson, J., Walter, T., and Lo, S. (2022). SBAS Signal Authentication. In *Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022)*, pages 3369–3377.

[Scott, 2003] Scott, L. (2003). Anti-Spoofing & Authenticated Signal Architectures for Civil Navigation Systems. In *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*, pages 1543 – 1552.