Civilian GPS Spoofing Detection based on Dual-Receiver Correlation of Military Signals

by Mark L. Psiaki, Brady W. O'Hanlon
Cornell University, Ithaca, N.Y. 14853-7501, U.S.A.

Jahshan A. Bhatti, Daniel P. Shepard, and Todd E. Humphreys
The University of Texas at Austin, Austin, Texas 78712-0235, U.S.A.

BIOGRAPHIES

Mark L. Psiaki is a Professor of Mechanical and Aerospace Engineering. He received a B.A. in Physics and M.A. and Ph.D. degrees in Mechanical and Aerospace Engineering from Princeton University. His research interests are in the areas of GNSS technology and applications, spacecraft attitude and orbit determination, and general estimation, filtering, and detection.

Brady W. O'Hanlon is a graduate student in the School of Electrical and Computer Engineering. He received a B.S. in Electrical and Computer Engineering from Cornell University. His interests are in the areas of GNSS technology and applications, GNSS security, and space weather.

Jahshan A. Bhatti is pursuing a Ph.D. in the Department of Aerospace Engineering and Engineering Mechanics. He received a B.S. and an M.S. in Aerospace Engineering from the University of Texas at Austin. He is a member of the UT Radionavigation Laboratory. His research interests are in the development of small satellites, software-defined radio applications, space weather, and GNSS security and integrity.

Daniel P. Shepard is pursuing a Ph.D. in the Department of Aerospace Engineering and Engineering Mechanics. He received a B.S. in Aerospace Engineering from the University of Texas at Austin. He currently works in the UT Radionavigation Lab. His research interests are in GNSS security, estimation and filtering, and guidance, navigation, and control.

Todd E. Humphreys is an Assistant Professor of Aerospace Engineering and Engineering Mechanics and Director of the UT Radionavigation Laboratory. He received a B.S. and an M.S. in Electrical and Computer Engineering from Utah State University and a Ph.D. in Aerospace Engineering from Cornell University. His research interests are in estimation and filtering, GNSS technology, GNSS-based study of the ionosphere and neutral atmosphere, and GNSS security and integrity.

ABSTRACT

Cross-correlations of unknown encrypted signals between two civilian GNSS receivers are used to detect spoofing of known open-source signals. This type of detection algorithm is the strongest known defense against sophisticated spoofing attacks if the defended receiver has only one antenna. The attack strategy of concern starts by overlaying false GNSS radio-navigation signals exactly on top of the true signals. The false signals increase in power, lift the receiver tracking loops off of the true signals, and then drag the tracking loops and the navigation solution to erroneous, but consistent results. This paper develops codeless and semi-codeless spoofing detection methods for use in inexpensive, narrow-band civilian GNSS receivers. Detailed algorithms and analyses are developed that use the encrypted military P(Y) code on the L1 GPS frequency in order to defend the open-source civilian C/A code. The new detection techniques are similar to methods used in civilian dual-frequency GPS receivers to track the P(Y) code on L2 by cross-correlating it with P(Y) on L1. Successful detection of actual spoofing attacks is demonstrated by off-line processing of digitally recorded RF data. The codeless technique can detect attacks using 1.2 sec of correlation, and the semi-codeless technique requires correlation intervals of 0.2 sec or less. This technique has been demonstrated in a narrow-band receiver with a 2.5 MHz bandwidth RF front-end that attenuates the P(Y) code by 5.5 dB.

INTRODUCTION

The vulnerability of unencrypted civilian GNSS signals to spoofing has long been known. The U.S. Department of Transportation has noted the vulnerability of GPS to spoofing [1]. Spoofing is the intentional broadcast of false signals that, in a user receiver, appear to be true signals. Spoofing of GNSS signals can cause a user receiver to determine a location that is far different from its true position, to compute erroneous corrections to its receiver clock, or to make both errors simultaneously [2,3,4,5,6,7].

Preprint from ION GNSS 2011

The spoofing attack described in Refs. 5 and 6 is hard to detect. It synthesizes spoofing signals for multiple satellites in a way that initially overlays them on top of the true signals. Next, it slowly pulls the victim receiver away from truth time and location in a self-consistent way. Typical Receiver Autonomous Integrity Monitoring (RAIM) methods for spoofing detection will fail to detect such an attack because they look for signal inconsistencies at the navigation level, which are not present in this scenario.

New RAIM methods are being developed to try to detect this type of attack at the tracking-loop/discriminator/correlator level [8,9,10]. These detection algorithms are complex and may be difficult to implement robustly. If such algorithms are to succeed, typically they must achieve detection at the moment of signal drag-off, which degrades their robustness.

Several other approaches have been proposed to detect this type of spoofing attack. These methods include cross-correlation of encrypted signals between secure and defended receivers [11,12], the use of multiple antennas [13], and methods that rely on inertial measuring devices and high-stability clocks. Other proposed methods would require changes to the navigation data message to provide Navigation Message Authentication (NMA) [3,14], or some sort of partial encryption of spreading codes [3,7]. NMA techniques may need to be implemented in conjunction with algorithms that detect dynamic estimation-and-replay spoofing of the NMA authentication bits [15].

The cross-correlation method of Refs. 11 and 12 has several advantages over the other methods. It does not require an extra GPS antenna or an IMU. It does require a communication link from a secure receiver so that parts of the two receivers' signals can be cross-correlated. The NMA method and methods based on new encrypted portions of the spreading code have the disadvantage of needing to change aspects of the broadcast signal. Presumably NMA could be implemented as an extension of the modern GPS civil navigation (CNAV) messaging format. The NMA approach would have a longer latency, taking up to 5 minutes to authenticate a signal, versus latency on the order of one second or less if using the cross-correlation method. Because of these advantages, the remainder of this paper focuses on the cross-correlation spoofing detection method.

The cross-correlation method relies on encrypted signals that are broadcast on the same frequency as the open-source signal that is being tracked for navigation purposes. For example, a GPS civilian receiver might track and use the unencrypted civilian pseudo-random number (PRN) codes such as the C/A code on the L1 frequency or the new L2C code on the L2 frequency. These frequencies also carry the encrypted military P(Y) PRN codes and, on newer satellites, the encrypted military binary offset carrier (BOC) M-codes. The civilian PRN codes can be spoofed using the technique of Refs. 5 and 6 or related techniques because the spoofer has prior knowledge of the codes. The spoofing detection methods proposed in Refs. 11 and 12 use the known carrier-phase and code-phase relationships between the tracked civilian codes and the encrypted military codes. These methods correlate the parts of the signal known to contain the encrypted military codes between two receivers. One receiver is presumed to reside in a secure location so that it has the correct encrypted code in the expected location. The spoofing detection algorithm correlates this part of the signal from the secure receiver with the same part of the signal from the other receiver, the potential spoofing victim. If the correlation is large enough, by an appropriate statistical measure, then the null-hypothesis of no spoofing is accepted. Otherwise, a spoofing alert is issued for the signal.

This strategy and the relationship of the open-source and encrypted signals is illustrated in Fig. 1 for the C/A and P(Y) signals on the GPS L1 frequency. The signals in the secure reference receiver are depicted in the left-hand plot, with the vertical blue curve depicting the C/A PRN code signal and the horizontal red/green curve depicting the P(Y) PRN code. Time increases along the second horizontal axis. The right-hand plot shows the same sections of these two signals in the second receiver, the potential victim for which spoofing detection must be performed. The use of orthogonal axes represents the fact that the C/A and P(Y) codes are modulated onto the carrier signal in phase quadrature. The strategy of Refs. 11 and 12 is to track the blue C/A signals in each receiver and to use the knowledge of these signals' phase and timing relationships to the P(Y) code in order to strip off the green part of the received P(Y) code in each receiver. Although this green signal is not known by either receiver a priori and although its received version is noisy, a correlation between these two green segments will produce a large statistic only if the correct P(Y) code is present in both receivers. This will be true only if the defended receiver is not being spoofed.

Reference 11 tested the un-spoofed case for this method. It showed a significant inter-receiver correlation of the baseband-mixed signal that was in phase quadrature with the GPS L1 C/A code. Thus, it verified lack of spoofing based on the encrypted L1 P(Y) signal. That effort did not perform a statistical analysis of the proper detection threshold for a spoofing alert, nor did it test the method under an actual spoofing attack. Its correlation calculations, which were based on batch laboratory data collection and analysis techniques, amounted to a proof-of-concept implementation. They required an expensive code offset timing search between the baseband quadrature signals of the two receivers. Further refinements are needed in order to develop a practical

operational system.

The effort of Ref. 12 sought to remedy several of these short-comings. It presents a statistical analysis of spoofing detection thresholds. In addition, it attempted to develop a system that could function in real-time. Its approach to real-time detection was to stream raw RF samples directly from the secure receiver to the potential victim receiver via the internet. The defended receiver, the potential victim of spoofing, was a software radio receiver. It had the real-time capacity to track signals both from its own antenna and in the streamed RF data that originated from the secure antenna. It also had the capacity to do the necessary correlation calculations of the quadrature baseband signals from the two data streams.

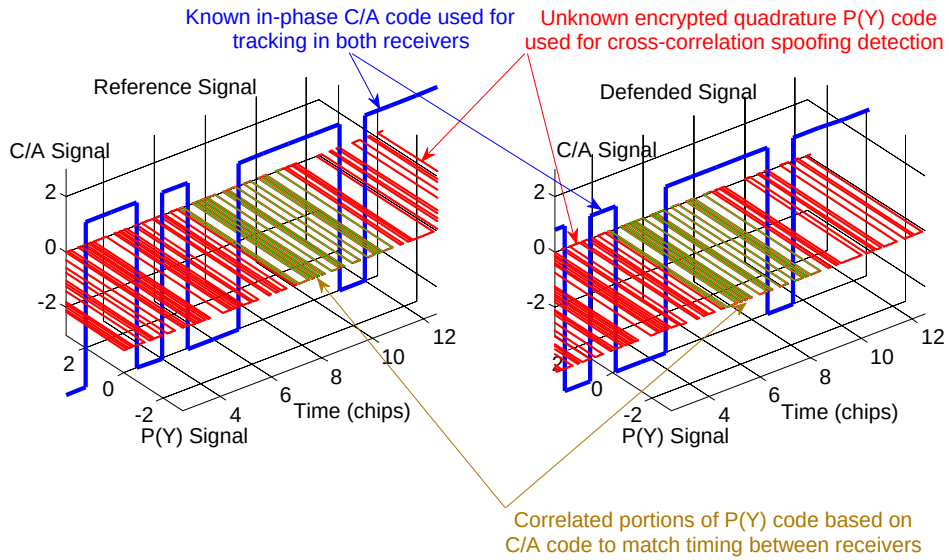

Fig. 1. Relationship of known open-source C/A signal and encrypted P(Y) signal on two receivers.

A significant contribution of Ref. 12 is an analysis which shows that the P(Y) code can be used for practical spoofing detection even in a narrow-band C/A-code receiver, i.e., one with an RF front-end bandwidth of only 1.9 MHz. Reference 11 implies the need for a wide-band RF front-end for this type of approach. A 1.9 MHz narrow-band receiver attenuates the P(Y) code by 6.9 dB and greatly distorts it, but there is still enough vestigial signal to achieve reasonable detection power for reasonable cross-correlation intervals. Unfortunately, Ref. 12 failed to achieve successful spoofing detection results due to bugs in its real-time software radio inter-receiver correlation calculations.

A further improvement to the cross-correlation method of Refs. 11 and 12 may be possible. This is true if the encrypted signal has a structure that allows a narrowing of its bandwidth in the two receivers prior to cross correlation. This narrowing of the bandwidths increases and signal-to-noise ratio (SNR), which decreases any possible squaring loss. Squaring loss is the loss of SNR that occurs when multiplying together two noisy signals. For GPS P(Y) code, signal bandwidth can be reduced by estimating the unknown W encryption chips that transform the open-source P code into the encrypted P(Y) code. This technique is used regularly in semi-codeless reception of the P(Y) signal on the L2 frequency in civilian dual-frequency GPS receivers [16,17,18]. If the encrypted signal in question is the GPS M code, then bandwidth reduction can be achieved by mixing out the known BOC signal to leave only the unknown 5.115 MHz spreading code chips, which then must be estimated in a manner analogous to W-chip estimation.

It is well known that semi-codeless dual-frequency civilian GPS techniques have an improved processing gain in comparison to codeless dual-frequency techniques. In the present context, the semi-codeless cross-correlation takes place between encrypted signals from two receivers that are both receiving the same frequency, e.g., the GPS L1 frequency. Although this differs from cross-correlation in a single receiver between the L1 and L2 frequencies, the potential for improved processing gain is the same. This improved gain should yield a better probability of detection for a given false alarm rate and correlation interval. In order to realize this processing gain, it is necessary to know the timing of the unknown encrypted chips relative to the tracked open-source signal. For the P(Y) code, the timing of the unknown W-chips relative to the P code and relative to the C/A code has been addressed by several researchers [16,18,19]. For M code, the timing of the BOC relative to the spreading codes and relative to the C/A code may have to be deduced by a one-time set of observations from a high-gain antenna.

This paper makes three principal contributions. First, it implements the codeless spoofing detection test of Ref. 12 and provides the first demonstrations of its effectiveness in detecting a sophisticated spoofing attack as defined in Refs. 5 and 6. It does this using recorded RF front-end data from two receivers in off-line MATLAB calculations. The RF front-ends have bandwidths of only 2.4 and 2.6 MHz. Therefore, this demonstration confirms the hypothesis of Ref. 12 that narrow-band receivers have

sufficient vestigial P(Y) code for purposes of spoofing detection.

The second contribution is the development and test of a semi-codeless form of P(Y)-code-based spoofing detection. Its method works even for a narrow-band receiver whose RF front-end filter passes less than 30% of the P(Y) signal's power. The receiver produces a very distorted P(Y) code that poses a challenge to semi-codeless signal processing. The test of this second detection method also uses off-line MATLAB calculations that operate on recorded RF data. It demonstrates that the semi-codeless method can detect a spoofing attack of the type in Refs. 5 and 6 with greater efficiency than the codeless method.

The last contribution is a theoretical comparison between the spoofing detection power of the codeless and semi-codeless techniques for a narrow-band L1 receiver. This comparison demonstrates a significant improvement on processing gain for the semi-codeless method.

Techniques similar to this paper's semi-codeless W-chips spoofing detection algorithm could be developed for the M-code. Such developments would require a much wider bandwidth RF front-end. They would also require a modified algorithm to remove the BOC signal, rather than P-code chips, before estimating the chips of the 5.115 MHz PRN spreading code. Another important aspect of an M-code-based system is the need to have Doppler separation or directional antennas at the secure reference station. Otherwise, it would be impossible to completely separate the M codes of different satellites. Any needed antenna directionality could be provided by a phased-array antenna with independent beam directions for each channel.

This paper does not attempt to devise any strategy in the event that a spoofing attack has been detected. Rather, its only goal is to inform the defended receiver whether or not its tracked open-source signals are reliable.

The remainder of this paper consists of 5 sections plus conclusions. Section II presents a mathematical model of the L1 C/A and P(Y) signals and of quadrature baseband mixing. These two signals are, respectively, the example open-source and encrypted signals that are considered throughout this paper. Section III reviews and explains the codeless spoofing detection method. Section IV develops the semi-codeless spoofing detection method based on W-chip estimation for the P(Y) code. This section also compares the detection power of the codeless and semi-codeless techniques. Section V presents test results for the two spoofing detection methods. Section VI discusses the possibility that modified spoofing attack strategies might provide tougher challenges to these methods, and it discusses possible responses to such challenges. Section VII presents this paper's conclusions.

II. MATHEMATICAL MODELS OF SIGNALS AND PRE-PROCESSING

A. Received Signal Models

The spoofing detection analysis starts with models of the received signals at the outputs of the RF front-ends of 2 receivers. These signals take the form:

$$y_{ai} = A_{ca} C_f(t_{ai}) cos[\omega_{IF} t_{ai} + \phi_a(t_{ai})] + A_{pa} P_{Yf}(t_{ai}) sin[\omega_{IF} t_{ai} + \phi_a(t_{ai})] + n_{ai} \quad (1a)$$

$$y_{bi} = A_{cb} C_f(t_{bi}) cos[\omega_{IF} t_{bi} + \phi_b(t_{bi})] + A_{pb} P_{Yf}(t_{bi}) sin[\omega_{IF} t_{bi} + \phi_b(t_{bi})] + n_{bi} \quad (1b)$$

where y_{ai} is the sample output by Receiver A's RF front-end at Receiver Clock A sample time t_{ai} and where y_{bi} is the sample output by Receiver B's RF front-end at Receiver Clock B sample time t_{bi}. Receiver A is assumed to be the secure reference receiver. Receiver B is the potential victim of a spoofing attack, the receiver for which spoofing detection must be performed.

The function $C_f(t)$ is the product of the C/A code and the 50 Hz navigation data bits as distorted and attenuated by the filter in the RF front-end. The function $P_{Yf}(t)$ is the distorted and attenuated product of the received P(Y) code and the navigation data bits. In the present analysis, these functions are presumed to be the same in both receivers. The semi-codeless analysis of Section IV will relax this assumption. Nominally these functions would be either +1 or -1 at all times due to the BPSK nature of the PRN codes and the navigation data, and their powers would equal 1. The RF front-end filters distort these signals so that they can take on different values than +/-1, and the filtering lowers their powers to values less than 1. Referring to Fig. 1, $C_f(t)$ is represented by the blue curves, and $P_{Yf}(t)$ is represented by the red/green curves, except that the figure does not depict distortion or attenuation. These functions' phase quadrature relationship in Eqs. (1a) and (1b) is illustrated in the figure by their being plotted along orthogonal axes.

The received C/A code amplitudes for the two receivers are, respectively, A_{ca} and A_{cb}. The corresponding received P(Y) amplitudes are A_{pa} and A_{pb}. Subsequent analyses in this paper assume that the P(Y) amplitudes can be deduced from the C/A amplitudes. This calculation takes the form:

$$A_p = A_c 10^{0.4/20} \sqrt{L_p} \quad (2)$$

where L_p is the power loss factor of the broadcast P(Y) code relative to the broadcast C/A code for the satellite in question. Typically $10 log_{10}(L_p)$ equals approximately -3 dB [20]. The 0.4 dB term in the exponent of Eq. (2) compensates for the fact that L_p is defined in the +/-10.23 MHz bandwidth centered at L1, which contains only the main lobe of the P(Y) power spectral density but 18

additional side-lobes of the C/A spectral density. The "a" and "b" subscripts have been omitted from Eq. (2) because it applies to both pairs of amplitudes for both receivers using the identical loss factor L_p.

The frequency ω_{IF} is the nominal intermediate frequency. It is the frequency to which the nominal carrier at ω_{L1} = 2πx1575.42x10^6 rad/sec gets mixed by the RF front-end.

The functions $\phi_a(t)$ and $\phi_b(t)$ are the beat carrier phase time histories of the signals at Receivers A and B, respectively. They have the opposite sign to the usual definition of beat carrier phase in the GPS literature. Their time derivatives equal the received carrier Doppler shifts.

The quantities n_{ai} and n_{bi} are the receiver noise terms. They are assumed to be discrete-time Gaussian white-noise with statistics:

$$E\{n_{ai}\} = 0,\ E\{n_{ai}^2\} = \sigma_{RFa}^2,\ E\{n_{ai}n_{aj}\} = 0 \ \text{ for all } i \neq j \tag{3a}$$

$$E\{n_{bi}\} = 0,\ E\{n_{bi}^2\} = \sigma_{RFb}^2,\ E\{n_{bi}n_{bj}\} = 0 \ \text{ for all } i \neq j \tag{3b}$$

$$E\{n_{ai}n_{bj}\} = 0 \ \text{ for all } i, j \tag{3c}$$

B. C/A-Code and Carrier Tracking and Quadrature Baseband Mixing

The spoofing detection algorithms of this paper presume that the reference and defended receivers are able to acquire and track the C/A code signals in Eqs. (1a) and (1b). A Delay-Lock Loop (DLL) is presumed to track the C/A PRN code in order to determine the start/stop times in $C_f(t)$. Suppose that these times are τ_{ak} and τ_{bk} at the end of the (k-1)st C/A code period and the start of the k^{th} C/A code period, as measured at Receivers A and B using their respective clocks. The tracking algorithms also use a Phase-Lock Loop (PLL) in order to determine the estimated beat carrier phase time histories $\hat{\phi}_a(t)$ and $\hat{\phi}_b(t)$.

The PLL uses feedback from a carrier-phase discriminator. The discriminator is computed from the following prompt in-phase and quadrature accumulations for the k^{th} code period:

$$I_k = \sum_{i=i_k}^{i_k+N_k-1} y_i C[(t_i - \tau_k)(1+\hat{\omega}_{Dk}/\omega_{L1})]\times \cos[\omega_{IF}t_i + \hat{\phi}_k + \hat{\omega}_{Dk}(t_i - \tau_k)] \tag{4a}$$

$$Q_k = \sum_{i=i_k}^{i_k+N_k-1} y_i C[(t_i - \tau_k)(1+\hat{\omega}_{Dk}/\omega_{L1})]\times \sin[\omega_{IF}t_i + \hat{\phi}_k + \hat{\omega}_{Dk}(t_i - \tau_k)] \tag{4b}$$

where the "a" and "b" subscripts have been omitted because the accumulation processing is similar in both receivers. The sample index i_k is the first sample of the k^{th} code period, i.e., the first sample such that $\tau_k \leq t_i$. The number N_k is the total number of samples in the code period so that the terminal index i_k+N_k-1 is the last sample of the code period, that is, the last sample such that $t_i < \tau_{k+1}$. The function $C[t]$ is the +1/-1-valued C/A PRN code without RF filter effects. The frequency $\hat{\omega}_{Dk}$ is the PLL's carrier Doppler shift estimate for the k^{th} code period, and the phase $\hat{\phi}_k$ is the estimated beat carrier phase at the code period start time τ_k.

Quadrature baseband mixing is used in order to isolate the P(Y)-code part of the signal. The quadrature baseband mixed signals for the k^{th} C/A code period are computed as follows:

$$y_{qi} = y_i\{I_k \sin[\omega_{IF}t_i + \hat{\phi}_k + \hat{\omega}_{Dk}(t_i - \tau_k)] - Q_k \cos[\omega_{IF}t_i + \hat{\phi}_k + \hat{\omega}_{Dk}(t_i - \tau_k)]\} / \sqrt{I_k^2 + Q_k^2} \quad \text{for } i = i_k, \ldots, (i_k+N_k-1) \tag{5}$$

where y_{qi} is the quadrature baseband mixed signal that corresponds to the original sample y_i. This mixing formula uses both the estimated carrier-phase time history from the PLL and the in-phase and quadrature accumulations. If the PLL has settled, then the quadrature accumulation Q_k will nominally be zero, and this formula will approximate simple multiplication by the quadrature $\sin[\omega_{IF}t_i+...]$ signal. Equation (5) is used in place of this simple multiplication because it compensates for the effects of navigation data bit signs and for PLL tracking errors. The latter compensation assumes that the noise effects on I_k and Q_k are negligible.

Again, the "a" and "b" subscripts have been omitted from Eq. (5). In later analyses, the quadrature baseband-mixed samples of the two receivers must be distinguished from each other. They will be designated as y_{qai} and y_{qbi}. They are computed as in Eq. (5), except that the quantities y_i, I_k, Q_k, t_i, $\hat{\phi}_k$, $\hat{\omega}_{Dk}$, τ_k, i_k, and N_k are modified to include an "a" or "b" subscript, depending on whether y_{qai} or y_{qbi} is being calculated.

Equation (5) provides a recipe for computing the quadrature baseband-mixed signal in each receiver. It is helpful also to have a model of this signal for each receiver. A model can be derived by substitution of the signal model in Eq. (1a) or (1b) into Eq. (5) and by assuming that the true beat carrier phase time history is accurately represented by $\hat{\phi}_k + \hat{\omega}_{Dk}(t_i - \tau_k)$ -$atan2(Q_k, I_k)$. The function $atan2(\ ,\ )$ is the usual 2-argument arctangent function. The resulting models for the two receivers take the form:

$$y_{qai} = \tfrac{1}{2} A_{pa} P_{Yf}(t_{ai}) + n_{qai} \quad (6a)$$

$$y_{qbi} = \tfrac{1}{2} A_{pb} P_{Yf}(t_{bi}) + n_{qbi} \quad (6b)$$

where the quadrature baseband noise terms n_{qai} and n_{qbi} have the statistics

$$E\{n_{qai}\} = 0, E\{n_{qai}^2\} = \tfrac{1}{2}\sigma_{RFa}^2, E\{n_{qai} n_{qaj}\} = 0 \text{ for all } i \neq j \quad (7a)$$

$$E\{n_{qbi}\} = 0, E\{n_{qbi}^2\} = \tfrac{1}{2}\sigma_{RFb}^2, E\{n_{qbi} n_{qbj}\} = 0 \text{ for all } i \neq j \quad (7b)$$

$$E\{n_{qai} n_{qbj}\} = 0 \text{ for all } i, j \quad (7c)$$

The models in Eqs. (6a) and (6b) ignore the parts of the signals in Eqs. (1a) and (1b) that get mixed to vicinity the frequency $2\omega_{IF}$ by the operations in Eq. (5). This is reasonable because the neglected high-frequency signals will not affect the subsequent baseband processing.

These quadrature models neglect the effect of 50 Hz navigation data bits on the mixing recipe in Eq. (5). The neglected sign effects will be identical for both receivers. The goal of this effort is to cross-correlate the two quadrature signals. Therefore, neglected data bit signs will not have an impact because they will cancel each other in all cross-correlation calculations.

C. Modeling W Encryption Chips and RF Filter Distortion of the P(Y) Code

The P(Y) code can be modeled as the product of the known P code [20] multiplied by unknown W encryption chips. This model takes the form

$$P_Y(t) = P(t)W(t) \quad (8)$$

where $P_Y(t)$ is the +/-1-valued encrypted P(Y) code, $P(t)$ is the +/-1-valued known P code, and $W(t)$ is the +/-1-valued unknown time history of encryption chips. The $W(t)$ encryption chips have an average chipping rate of 480 KHz. The PRN code time history $P(t)$ and the encryption chip time history $W(t)$ have known code phase relationships between the times when their chip sign transitions can occur and between the times when the chip sign transitions can occur on the corresponding C/A code $C(t)$.

The timing of the $W(t)$ chips is discussed in Refs. 16, 18, and 19. The following description is based on Ref. 19 and on unpublished results that were obtained during the study which is reported in that work. The $W(t)$ chip timing is directly linked to that of the X1A code, which is a generator code that is used to form the known $P(t)$ code [20]. The X1A code chips at 10.23 MHz and repeats every 4092 chips, i.e., every 400 μsec. Each chip interval of the X1A code is aligned with a chip interval of the $P(t)$ code. Every 4092 chips of X1A code is broken down into L_w equal sets of chip periods. Each of these $4092/L_w$ chip periods is broken down into M_w W-chip periods of duration I_w P-code chips followed by N_w W-chip periods of duration J_w P-code chips. Thus, $L_w(I_w M_w + J_w N_w) = 4092$. The exact values of L_w, M_w, N_w, I_w, and J_w have been determined, but have not been fully published. References 16 and 18 erroneously report that $L_w = 1$. Reference 19 reports that L_w is greater than 1. Published information indicates that the two durations of the W chips, I_w and J_w, are about 20 P chips and that the average W chipping rate $[L_w(M_w+N_w)/4092]$x10230 KHz = 480 KHz.

The filtered version of the P(Y) code that appears in Eqs. (1a), (1b), (6a), and (6b) can be modeled as follows:

$$P_{Yf}(t) = \sum_{j=-\infty}^{\infty} w_j P_{fwj}(t) \quad (9)$$

where w_j is the j^{th} +/-1-valued W chip and where $P_{fwj}(t)$ is the attenuated and distorted version of the 20 or so P chips that correspond to the j^{th} W chip.

The w_j chip values cannot be known a priori in a civilian receiver, but the functions $P_{fwj}(t)$ can be determined based on the known P code, the known W-chip timing, and the modeled effects of the RF front-end filter. Suppose that the unfiltered version of $P_{fwj}(t)$ takes the form:

$$P_{wj}(t) = \sum_{i=i_{wj}}^{i_{wj}+I_{wj}-1} p_i \Pi_{T_p}[t - (i - i_{wj})T_p - \tau_{wj}] \quad (10)$$

where p_i is the known +1/-1 value of the i^{th} P-code chip of the given GPS week, T_p is the P-code chip period, i_{wj} is the index of the initial P-code chip of the j^{th} W chip as measured from the start of the GPS week, I_{wj} is the total number of P-code chips in the j^{th} W chip, and τ_{wj} is the start time of the j^{th} W chip and of the $(i_{wj})^{\text{th}}$ P chip. The function $\Pi_T(t)$ is the usual rectangular support function, which is equal to one over the interval $0 \leq t < T$ and zero elsewhere. The P-code chip period is nominally $T_p = 1/(10.23\text{x}10^6)$ sec, but it will vary if there is a non-zero code Doppler shift.

The filtered version of these same P-code chips takes the form

$$P_{fwj}(t) = \sum_{i=i_{wj}}^{i_{wj}+I_{wj}-1} p_i \Psi[t - (i - i_{wj})T_p - \tau_{wj}] \quad (11)$$

where $\Psi(t)$ is the filtered version of the rectangular support function $\Pi_T(t)$:

$$\Psi(t) = \begin{cases} 0 & t \leq 0 \\ \int_{t-T_{hmax}}^{t} h_{RF}(t-\tau)\Pi_{T_p}(\tau)d\tau & 0 < t \leq (T_p + T_{hmax}) \\ 0 & (T_p + T_{hmax}) < t \end{cases} \quad (12)$$

In this formula, $h_{RF}(t)$ is the real part of the envelop impulse response function of the receiver's RF filter. This

function can be determined using off-line system identification techniques [21]. Equation (12) assumes that $h_{RF}(t)$ is a finite impulse response with zero response T_{hmax} sec after the impulse. This is a reasonable approximation for a large enough T_{hmax}, and it is consistent with the system identification assumptions of Ref. 21.

Figure 2 plots five examples of the unfiltered and filtered sets of P-code chips that are associated with 5 different W chips. The upper graph plots the unfiltered time histories $P_{wj}(t)$ for j = 1, ..., 5, and the lower plot shows the corresponding filtered $P_{fwj}(t)$ time histories. Each W chip in this example spans I_{wj} = 20 P-code chips. The lower plots have been generated using the $h_{RF}(t)$ filter impulse response function associated with one of the RF front-ends that has been used to generate results for Section V. The plots for the other receiver's RF front-end would be similar. It is obvious from Fig. 2 that this narrow-band RF filter causes significant power attenuation and distortion in the P(Y) signal. These accurate models of the attenuation and distortion are important to the development of this paper's spoofing detection algorithms.

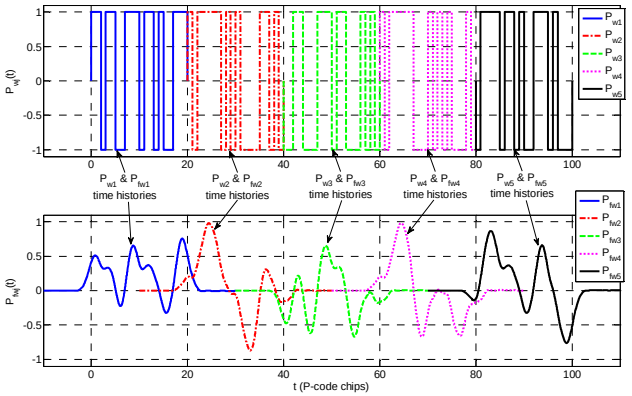

Fig. 2. Wide-band (top) and filtered (bottom) P-code chips of 5 successive W encryption chips (filtering performed by a 2.5 MHz wide narrow-band RF front-end; filter delay removed from bottom plot).

D. P(Y) Code and C/A Code Power Loss in the RF Front-End Filter

The filter impulse response function can be used to determine the P(Y) signal's power loss in the narrow-band RF front-end. This calculation starts by computing the envelop filter's frequency response

$$H_{RF}(j\omega) = \int_0^{T_{hmax}} h_{RF}(t) e^{-j\omega t} dt \qquad (13)$$

where $j = (-1)^{1/2}$ in this formula. The square of the absolute value of this function multiplies the unfiltered P(Y) code's normalized power spectral density

$$S_{py}(\omega) = \left[\frac{\sin(\omega T_p/2)}{(\omega T_p/2)} \right]^2 \qquad (14)$$

in order to yield the corresponding filtered power spectral density. The ratio of the integrals of the filtered and unfiltered power spectral densities gives the power loss through the filter. It is

$$L_{fpy} = \frac{\int_{-2\pi/T_p}^{2\pi/T_p} |H_{RF}(j\omega)|^2 S_{py}(\omega) d\omega}{\int_{-2\pi/T_p}^{2\pi/T_p} S_{py}(\omega) d\omega} \qquad (15)$$

Recall that T_p in these formulas is the P-code chipping period. Thus, these integrals are performed over the main lobe of the P(Y) signal, i.e., over the range -10.23 MHz to +10.23 MHz.

Another power loss factor is that of the C/A code. It is important because the spoofing detection calculations need to know P(Y) code power or amplitude, and they infer it from C/A code amplitude using calculations like those in Eq. (2). The C/A code loss factor must account for two effects. One is the loss in the RF front-end filter, and the other is the loss associated with the accumulation calculations in Eqs. (4a) and (4b). The latter loss arises from the use of the unfiltered C/A code $C[t]$ in the accumulation recipes. The total power loss of the C/A code at the output of the $[I_k,Q_k]$ accumulation process is:

$$L_{fca} = \left[\max_{\tau} \int_0^{t_{max}} h_{RF}(t) s_{ca}(t-\tau) dt \right]^2 \qquad (16)$$

where $s_{ca}(t)$ is the symmetric autocorrelation function of the unfiltered C/A code. The result of the integration in Eq. (16) is the cross-correlation between the filtered and unfiltered versions of the C/A code. Its maximum value is less than 1, but it approaches 1 as the filter bandwidth increases [21].

III. CODELESS SPOOFING DETECTION TECHNIQUE

This section develops an implementation of the codeless spoofing detection algorithm of Refs. 11 and 12. A significant amount of this material is taken from Ref. 12, but the notation has been changed in a number of places in order to conform with the models in Section II of the present paper. In addition to the notation changes, the developments of the present section include implementation details that are not found elsewhere.

A. Computation of the Raw Codeless Spoofing Detection Statistic

The raw codeless spoofing detection statistic is the sum of products of quadrature samples from Receivers A and B.

In other words, it is the sum of products of Eq. (6a) samples and Eq. (6b) samples. Before forming products, however, it is necessary to map sample times in the two receivers to identical values as measured relative to their respective tracked C/A codes. This inter-receiver time mapping relies on the DLL estimates of the C/A code start/stop times, τ_{a1}, τ_{a2}, ..., τ_{ak}, τ_{ak+1}, ... and τ_{b1}, τ_{b2}, ..., τ_{bk}, τ_{bk+1}, ...

Suppose, in addition, that there is a known differential relative timing offset between the filtered P(Y) code and the DLL estimate of the filtered C/A code. This offset is denoted by δt_{ab}, and it represents a difference between the two receivers. It is a measure of the amount by which the filtered P(Y) code in Receiver B is delayed relative to that receiver's DLL-generated C/A code replica when compared to the filtered P(Y) code in Receiver A. Nominally, one would expect this differential timing offset to be zero or nearly so. A non-zero value is allowed in the present analysis in order to make it more general and to facilitate an experimental study of the magnitude of this delay.

Suppose that the correlation calculation seeks the correct quadrature sample from Receiver B to correlate with sample y_{qai} from Receiver A, which was sampled at Receiver A clock time t_{ai}. Suppose that the delayed sample time ($t_{ai}+\delta t_{ab}$) lies in the Receiver A DLL's estimate of the reception interval of the k^{th} C/A PRN code period. That is, suppose that $\tau_{ak} \le (t_{ai}+\delta t_{ab}) < \tau_{ak+1}$. Then the first step in the correlation process is to compute the corresponding time according to Receiver B's clock. Using linear interpolation between DLL code start/stop times, it is:

$$\tilde{t}_{bi} = \tau_{bk} + \left(\frac{\tau_{bk+1} - \tau_{bk}}{\tau_{ak+1} - \tau_{ak}} \right) (t_{ai} + \delta t_{ab} - \tau_{ak}) \tag{17}$$

This Receiver B time estimate can be used to interpolate between Receiver B quadrature samples from Eq. (6b) in order to synthesize the "sample" of the Receiver-B quadrature signal that corresponds to the Receiver-A sample y_{qai}. Suppose that the interpolated time $\tilde{t}_{bi}$ from Eq. (17) lies between Receiver-B RF sample times t_{bj} and t_{bj+1}. Then the synthesized quadrature sample of Receiver B is the linearly interpolated value:

$$\tilde{y}_{qbi} = y_{qbj} + \left(\frac{y_{qbj+1} - y_{qbj}}{t_{bj+1} - t_{bj}} \right) (\tilde{t}_{bi} - t_{bj}) \tag{18}$$

The Receiver-A quadrature samples from Eq. (6a) and the synthesized Receiver-B quadrature samples from Eq. (18) are multiplied together and summed in order to form the un-normalized codeless spoofing detection statistic:

$$\gamma_{ul} = \sum_{i=i_l}^{i_l+M-1} y_{qai} \tilde{y}_{qbi} \tag{19}$$

The index i_l in this formula is the initial sample of the correlation accumulation interval, and M is the total number of samples used in each accumulation. This l^{th} un-normalized spoofing detection statistic spans a data interval of length $T_{corr} = M\Delta t$ sec, where $\Delta t = t_{ai+1} - t_{ai}$ is the RF front-end sample period. The mid-point of this interval is

$$t_{cl} = t_{ai_l} + \frac{(M-1)\Delta t}{2} \tag{20}$$

according to the Receiver-A clock.

B. Hypothesis Test for Spoofing based on a Normalized Codeless Detection Statistic

The spoofing detection statistic in Eq. (19) has significantly different properties depending on whether or not the C/A code signal tracked by Receiver B is a spoofed signal. If the signal is not spoofed, then the synthesized $\tilde{y}_{qbi}$ quadrature sample is assumed to be modeled by Eq. (6b). If the signal is spoofed, however, then the P(Y) code is presumed to be absent from the quadrature channel of Receiver B. In this case, Eq. (6b) is modified to setting the P(Y)-code amplitude to $A_{pb} = 0$.

Under the hypothesis of spoofing, hypothesis H_1, the mean and variance of the spoofing detection statistic γ_{ul} are

$$\begin{aligned} \bar{\gamma}_{u|H1} &= E\{\gamma_{ul} \mid H_1\} \\ &= \sum_{i=i_l}^{i_l+M-1} [\tfrac{1}{2} A_{pa} P_{Yf}(t_{ai}) + E\{n_{qai}\}] E\{\tilde{n}_{qbi}\} \\ &= 0 \end{aligned} \tag{21a}$$

$$\begin{aligned} \sigma^2_{\gamma u|H1} &= E\{\gamma^2_{ul} \mid H_1\} \\ &= E\left\{ \left[\sum_{i=i_l}^{i_l+M-1} [\tfrac{1}{2} A_{pa} P_{Yf}(t_{ai}) + n_{qai}] \tilde{n}_{qbi} \right]^2 \right\} \\ &= \sum_{i=i_l}^{i_l+M-1} [\tfrac{1}{4} A^2_{pa} P^2_{Yf}(t_{ai}) + \tfrac{1}{2} \sigma^2_{RFa}] \tfrac{1}{2} \sigma^2_{RFb} \\ &= \frac{M}{4} \sigma^2_{RFb} [\sigma^2_{RFa} + \tfrac{1}{2} A^2_{pa} \bar{P}^2_{Yf}] \\ &= \frac{M}{4} \sigma^2_{RFa} \sigma^2_{RFb} [1 + 2\Delta t (C/N_0)_{pya}] \end{aligned} \tag{21b}$$

where $\tilde{n}_{qbi}$ is the noise in the synthesized quadrature sample $\tilde{y}_{qbi}$, which is assumed to obey the same statistics as n_{qbi} in Eqs. (7b) and (7c). The quantity $\bar{P}^2_{Yf}$ is the mean value of P^2_{Yf}, i.e., it is the power of the distorted P(Y) code at the output of the RF front-end filter. The quantity $(C/N_0)_{pya} = A^2_{pa} \bar{P}^2_{Yf} / (4\sigma^2_{RFa} \Delta t)$ is the filtered P(Y)-code carrier-to-noise ratio in Receiver A. The derivations in Eqs. (21a) and (21b) depend on the assumptions that $E\{\tilde{n}_{qbi} \tilde{n}_{qbj}\} = 0$ for all (i,j) such that

$i \neq j$ and that $E\{n_{qai}\tilde{n}_{qbj}\} = 0$ for all (i,j).

Under the hypothesis of no spoofing, hypothesis H_0, the mean and variance of γ_{ul} are

$$\begin{aligned}\bar{\gamma}_{u|H0} &= E\{\gamma_{ul} \mid H_0\} \\ &= \sum_{i=i_l}^{i_l+M-1} [\tfrac{1}{2} A_{pa} P_{Yf}(t_{ai}) + E\{n_{qai}\}] \times \\ &\qquad [\tfrac{1}{2} A_{pb} P_{Yf}(\tilde{t}_{bi}) + E\{\tilde{n}_{qbi}\}] \\ &= \frac{M}{4} A_{pa} A_{pb} \bar{P}_{Yf}^2 \\ &= M\sigma_{RFa}\sigma_{RFb}\Delta t \sqrt{(C/N_0)_{pya}(C/N_0)_{pyb}} \end{aligned} \quad (22a)$$

$$\begin{aligned}\sigma^2_{\gamma u|H0} &= E\{\gamma_{ul}^2 \mid H_0\} - \bar{\gamma}^2_{u|H0} \\ &= E\Big\{\Big[\sum_{i=i_l}^{i_l+M-1} [\tfrac{1}{2} A_{pa} P_{Yf}(t_{ai}) + n_{qai}] \times \\ &\qquad [\tfrac{1}{2} A_{pb} P_{Yf}(\tilde{t}_{bi}) + \tilde{n}_{qbi}]\Big]^2\Big\} - \bar{\gamma}^2_{u|H0} \\ &= \left[\tfrac{1}{4} A_{pa} A_{pb} \sum_{i=i_l}^{i_l+M-1} P_{Yf}(t_{ai}) P_{Yf}(\tilde{t}_{bi})\right]^2 \\ &\qquad + \tfrac{1}{8} A_{pa}^2 \sigma_{RFb}^2 \sum_{i=i_l}^{i_l+M-1} P_{Yf}^2(t_{ai}) \\ &\qquad + \tfrac{1}{8} A_{pb}^2 \sigma_{RFa}^2 \sum_{i=i_l}^{i_l+M-1} P_{Yf}^2(\tilde{t}_{bi}) \\ &\qquad + \frac{M}{4} \sigma_{RFa}^2 \sigma_{RFb}^2 - \bar{\gamma}^2_{u|H0} \\ &= \left(\frac{M^2}{16} A_{pa}^2 A_{pb}^2 [\bar{P}_{Yf}^2]^2 - \bar{\gamma}^2_{u|H0}\right) \\ &\qquad + \frac{M}{8} [A_{pa}^2 \sigma_{RFb}^2 + A_{pb}^2 \sigma_{RFa}^2] \bar{P}_{Yf}^2 \\ &\qquad + \frac{M}{4} \sigma_{RFa}^2 \sigma_{RFb}^2 \\ &= \frac{M}{4} \sigma_{RFa}^2 \sigma_{RFb}^2 \{1 + 2\Delta t[(C/N_0)_{pya} \\ &\qquad + (C/N_0)_{pyb}]\} \end{aligned} \quad (22b)$$

where $(C/N_0)_{pyb} = A_{pb}^2 \bar{P}_{Yf}^2 / (4\sigma_{RFb}^2 \Delta t)$ is the P(Y)-code carrier-to-noise ratio in Receiver B.

The derivations in Eqs. (22a) and (22b) assume that the mean value of the product $P_{Yf}(t_{ai}) P_{Yf}(\tilde{t}_{bi})$ also equals $\bar{P}_{Yf}^2$. This is reasonable when the RF front-end filters are similar because the Receiver A time t_{ai} and the Receiver B time $\tilde{t}_{bi}$ are the same times relative to their respective P(Y) codes by virtue of the construction of $\tilde{t}_{bi}$ in Eq. (17). Of course, a stricter use of notation would have created slightly different function names for $P_{Yf}(t)$ in the two receivers in order to allow them to take on the same value at the different input time arguments t_{ai} and $\tilde{t}_{bi}$.

The carrier-to-noise ratios $(C/N_0)_{pya}$ and $(C/N_0)_{pyb}$ in the final forms of Eqs. (21b)-(22b) are used in place of terms involving $A_{pa}^2 \bar{P}_{Yf}^2$ and $A_{pb}^2 \bar{P}_{Yf}^2$. This convention is adopted because it is convenient to deduce the carrier-to-noise ratios. The determination of $(C/N_0)_{pya}$ and $(C/N_0)_{pyb}$ begins with a determination of the corresponding C/A-code carrier-to-noise ratios. Given a time history of prompt accumulations I_k and Q_k for the C/A code, the calculation starts by determining the mean amplitude of the accumulation vector $[I_k; Q_k]$ and the noise variance in each of this vector's components:

$$A_{IQ} = (\bar{z}^2 - \sigma_z^2)^{1/4} \quad (23a)$$

$$\sigma_{IQ}^2 = 0.5(\bar{z} - \sqrt{\bar{z}^2 - \sigma_z^2}) \quad (23b)$$

where $\bar{z}$ and σ_z^2 are, respectively, the mean and variance of the accumulation power:

$$\bar{z} = E\{I_k^2 + Q_k^2\} = \frac{1}{K} \sum_{k=1}^{K} (I_k^2 + Q_k^2) \quad (24a)$$

$$\sigma_z^2 = E\{[I_k^2 + Q_k^2]^2\} - \bar{z}^2 = \frac{1}{K} \sum_{k=1}^{K} (I_k^2 + Q_k^2)^2 - \bar{z}^2 \quad (24b)$$

As a side benefit, the accumulation variance in Eq. (23b) can be used to estimate the effective variance of the noise in the raw RF samples:

$$\sigma_{RF}^2 = \frac{2}{\bar{N}_{accum}} \sigma_{IQ}^2 \quad (25)$$

where $\bar{N}_{accum} = (N_1 + N_2 + \ldots + N_K)/K$ is the average number of samples in an accumulation. The value of this variance for each receiver is needed in Eqs. (21b) to (22b).

The C/A-code carrier-to-noise ratio is computed from the accumulation amplitude and variance in Eqs. (23a) and (23b). Given the accumulation interval $T_{accum} = \Delta t \, \bar{N}_{accum}$, the carrier-to-noise ratio is:

$$(C/N_0)_c = \frac{A_{IQ}^2}{2\sigma_{IQ}^2 T_{accum}} \quad (26)$$

Given the C/A-code carrier-to-noise ratio, the P(Y) code carrier-to-noise ratio can be computed. This calculation considers the effects of filter loss and distortion, as per Eqs. (15) and (16), and the transmitted power decrement of the P(Y) code in comparison to the C/A code, as per Eq. (2). The resulting formula is

$$(C/N_0)_{py} = L_{fpy} L_p \left[\frac{10^{-0.04/10} (C/N_0)_c}{L_{fca}}\right] \quad (27)$$

The power of 10 in this equation adjusts for the fact that the L_{fca} loss calculation in Eq. (16) presumes an infinite bandwidth of the transmitted C/A code instead of the actual 20.46 MHz bandwidth. The term in square

brackets on the right-hand side of this equation is what the received C/A-code carrier-to-noise ratio would have been had there been no loss in the filter or in the prompt accumulation calculations.

The formulas in Eqs. (23a)-(27) apply to Receivers A and B. The usual "*a*" and "*b*" subscripts can be added to each of the quantities in order to denote the receiver to which it applies.

Typically the variance results in Eqs. (23b), (24b), and (25) are computed only once when the receiver is operating on a quiescent signal with very little actual amplitude fluctuation. These quantities tend to remain constant over time due to the actions of the RF front-end's automatic gain control.

The signal power quantities in Eqs. (23a) and (24a) and the associated carrier-to-noise ratios in Eqs. (26) and (27) are typically re-computed continually. One might re-compute them for each spoofing detection accumulation interval. This approach enables the spoofing detection test to adapt to the time variations in signal power that typically occur.

Before developing the spoofing test, it is helpful to normalize the test statistic. A suitable normalization is to divide γ_{ul} by its standard deviation under the spoofed hypothesis H_1, $\sigma_{\gamma u|H1}$. This produces the normalized spoofing test statistic:

$$\gamma_l = \frac{\gamma_{ul}}{\sigma_{\gamma u|H1}} = \frac{\sum_{i=i_l}^{i_l+M-1} y_{qai}\tilde{y}_{qbi}}{\sigma_{RFa}\sigma_{RFb}\sqrt{\frac{M}{4}[1+2\Delta t(C/N_0)_{pya}]}} \quad (28)$$

The results in Eqs. (21a)-(22b) can be used to compute the means and standard deviations of this statistic under the respective hypotheses of spoofing on Receiver B, H_1, and no spoofing, H_0. These quantities are:

$$\bar{\gamma}_{H1} = 0 \quad (29a)$$

$$\sigma_{\gamma|H1} = 1 \quad (29b)$$

$$\bar{\gamma}_{H0} = 2\Delta t\sqrt{\frac{M(C/N_0)_{pya}(C/N_0)_{pyb}}{1+2\Delta t(C/N_0)_{pya}}} \quad (29c)$$

$$\sigma_{\gamma|H0} = \sqrt{\frac{1+2\Delta t[(C/N_0)_{pya}+(C/N_0)_{pyb}]}{1+2\Delta t(C/N_0)_{pya}}} \quad (29d)$$

The means and variances in Eqs. (29a)-(29d) can be used to design and analyze a sensible spoofing detection test. The necessary derivations require knowledge of the spoofed and un-spoofed probability density functions $p(\gamma_l|H_1)$ and $p(\gamma_l|H_0)$. The exact formulas for these functions are complicated because they involve products of the Gaussian noise terms n_{qai} and $\tilde{n}_{qbi}$. Fortunately, the randomness in γ_l is the result of many such product terms. Therefore, the central limit theorem can be invoked in order to model these two probability density functions as Gaussian distributions.

Given the Gaussian assumption and given the allowable false-alarm probability α_{FA}, the spoofing detection threshold γ_{th} can be computed by solving the following equation:

$$\alpha_{FA} = \int_{-\infty}^{\gamma_{th}} p(\gamma_l|H_0)d\gamma_l = \frac{1}{\sqrt{2\pi}\sigma_{\gamma|H0}}\int_{-\infty}^{\gamma_{th}} exp\{-\frac{(\gamma_l-\bar{\gamma}_{H0})^2}{2\sigma^2_{\gamma|H0}}\}d\gamma_l \quad (30)$$

This threshold is used to determine whether the signal in Receiver B is being spoofed according to the following rule: If $\gamma_l \geq \gamma_{th}$, then accept the H_0 hypothesis that there is no spoofing, but if $\gamma_l < \gamma_{th}$, accept the H_1 hypothesis that there is spoofing. This threshold and spoofing test lead to the following probability of a successful detection:

$$\mathcal{P}_{detect} = \int_{-\infty}^{\gamma_{th}} p(\gamma_l|H_1)d\gamma_l = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\gamma_{th}} exp\{-0.5\gamma_l^2\}d\gamma_l = 1-\mathcal{P}_{misdet} \quad (31)$$

Note that the H_0 un-spoofed hypothesis is somewhat unusual: It has a non-zero mean that is calculated by factoring down the measured C/A carrier-to-noise ratio in order to estimate the P(Y) carrier-to-noise ratio. It is important to use the proper calculation of the C/A carrier-to-noise ratio in Eqs. (23a)-(26) and the proper attenuation to get the P(Y) carrier-to-noise ratio in Eq. (27). Errors in these calculations will cause errors in the un-spoofed expected value $\bar{\gamma}_{H0}$ and in the spoofing detection threshold γ_{th}. These errors will cause the detection test to have a different false-alarm probability and a different probability of detection than are given in Eqs. (30) and (31).

The analysis of this section assumes that the noise in the quadrature baseband-mixed signal is purely white noise. This assumption is violated to some extent in any real receiver. For the receivers considered in the present study, their departures from the white-noise assumption do not appear to be large enough to have a significant impact on the spoofing detection results. If the non-whiteness of the noise were an issue, then it would be straight-forward to adapt the foregoing analysis appropriately. This adaptation is omitted for the sake of brevity.

The detection statistic γ_l would be the optimal Neyman-Pearson detection statistic [22] if the noise in Receiver A were negligible and if the prediction of the P(Y) carrier-to-noise ratios for the two receivers were exact. In that

case, the Receiver-A quadrature signal would yield a perfect scaled replica of the encrypted P(Y) code. One could use this replica and the P(Y) amplitudes on Receivers A and B in order to derive the joint probability density functions for y_{qbi} for $i = i_l, ..., i_l+M-1$ under the two hypotheses. One could demonstrate a monotonic, one-to-one correspondence between the ratio of these two probability density functions and the γ_l test statistic. This correspondence would prove the optimality of the γ_l statistic. The use of a sub-optimal test statistic is necessitated by the receivers' imperfect knowledge of the P(Y) signal.

C. Potential for Cross-Talk between Channels

There is a potential for the P(Y) code or even the C/A code of another GPS signal to affect the spoofing detection statistic γ_l in Eq. (28). This can happen if the Doppler shifts and code delays of the other GPS signal line up in a certain way with those of the signal for which spoofing detection is being performed. The necessary Doppler alignment to cause interference is that of a zero-valued or nearly zero-valued Doppler double difference between the two receivers and the two signals. That is, if the carrier Doppler shift difference between the two GPS signals is the same at both the reference receiver and the defended receiver, then there is a potential interference. This difference must be smaller than the correlation accumulation frequency $1/T_{corr}$. Otherwise, the averaging action of the accumulation in Eq. (28) will attenuate the interference.

An additional requirement for interference between two signals is that their double-differenced PRN code phase be zero or nearly zero. That is, the C/A code period start/stop time difference between the two signals for the reference receiver must equal this same difference for the defended receiver. If this code-phase double difference is less than the correlation time of the filtered P(Y) code, then un-intended cross-correlations of the P(Y) code of the other signal can appear in the γ_l spoofing detection statistic of Eq. (28). Similarly, if this code-phase double difference is less than a C/A code PRN chip length, then un-intended cross-correlations of the other signal's C/A code can appear in γ_l. The C/A code of the second signal could affect the P(Y) cross-correlation of the signal in question because the second C/A code could lie nearly in phase quadrature with the C/A code of the original signal.

This type of interference was noted in the study of codeless cross-correlation spoofing detection found in Ref. 12. In that study, the two receivers were both located in Ithaca, NY. Given this close proximity, the carrier Doppler shift double differences and the code phase double differences were likely to be small, and interference was likely to occur.

Under normal conditions, it is unlikely that two signals will interfere due to small double differences in Doppler shift and code phase. Large double differences will normally be caused by the necessary receiver separation between the secure reference receiver and the defended receiver. If both double differences are small, however, then this fact will be noticeable from the C/A code tracking, and the spoofing detection calculations for the signals in question must be ignored or modified. Otherwise, the computed γ_l can be much larger than expected, much smaller than expected, or even negative [12]. These possibilities arise because additional non-zero correlations of the second signal can add constructively or destructively to alter the mean value of γ_l.

It is possible to reduce or even eliminate this type of interference at the reference station. The necessary infrastructure would be a high-gain antenna system with independently steerable beams, such as could be provided by a phased array. Given sufficient gain, the interference effects of other signals on γ_l would be negligible even with zero-valued double differences of Doppler shift and code phase.

IV. SEMI-CODELESS SPOOFING DETECTION TECHNIQUE

The semi-codeless spoofing detection technique attempts to improve the power of the spoofing detector by employing additional a priori knowledge about the P(Y) code. This a priori knowledge is the fact that P(Y) is generated by mixing the known P code with the unknown W encryption chips, as described in Subsection II.C.

A. "Hard" W Chip Estimates

The heart of the semi-codeless spoofing detection method is an estimator for the unknown +1/-1 values of the w_j chips in Eq. (9). For a given interval of interest, the estimates are formed by solving the following batch least-squares estimation problem:

find: $w_1, w_2, w_3, ..., w_K$ (32a)

to minimize:

$$J(w_1,...,w_K) = \frac{1}{2}\sum_{i=1}^{M}[y_{qi} - \frac{1}{2}A_p \sum_{j=1}^{K} w_j P_{fwj}(t_i)]^2 \quad (32b)$$

subject to:

w_j = -1 or +1 for $j = 1, ..., K$ (32c)

The cost function in Eq. (32b) is half the sum of the squared errors in M instances of Eq. (6a) or (6b), depending on the receiver in question. This cost formula uses the W-chips model of the $P_{Yf}(t)$ function in Eq. (9) in order to frame the problem explicitly in terms of unknown W chips. Note that Eqs. (32a)-(32c) do not include "a" or "b" subscripts. These have been omitted because this estimation problem applies equally to both receivers. For the same set of W-chips, however, there is an independent W-chip estimation problem based on each

receiver's independent quadrature samples and on each receiver's $P_{fwj}(t)$ as dictated by the impulse response of its RF front-end's band-pass filter.

The sample indices i and the W-chip indices j in this problem are defined somewhat arbitrarily in order to simplify the estimation problem statement. In practice, the range of the RF sample index i might be different from 1 to M, and the W-chip index range might be different from 1 to K. The first RF sample, sample $i = 1$ at time t_1 in the problem above, should be the initial sample in the P(Y) code interval associated with the initial W-chip function $P_{fw1}(t)$, as per Fig. 2. Similarly, the last RF sample, sample $i = M$ at time t_M, should be the final sample associated with the final W-chip function $P_{fwK}(t)$.

The number of W chips estimated in a given batch optimization, K, is arbitrary. A sensible choice would set K equal to the number of W chips that were used to calculate a single semi-codeless spoofing detection statistic. Thus, if a 0.2 sec correlation were used for each independent detection statistic, then a sensible choice of K would be (0.2sec)x(480,000 W-chips/sec) = 96,000 W-chips.

The cost function in Eq. (32b) presumes that the timing of the received, RF-filtered W chips is known exactly according to the receiver clock. That is, the functions $P_{fwj}(t)$ are presumed to be known with t measured in receiver clock time. This knowledge depends on the working of the C/A-code DLL, on knowledge of the relative delays between the filtered C/A-code chips and the filtered P-code chips that comprise $P_{fwj}(t)$, and on knowledge of the nominal W-chip timing relative the nominal C/A code timing at the transmitter. The latter knowledge is well defined by the GPS Interface Specification [20] in conjunction with the W-chip timing studies associated with Refs. 17 and 19. The other two pieces of information are effectively defined by the way that the RF filter impulse response function, as estimated by the system identification procedures of Ref. 21, is defined relative to the C/A code DLL tracking point. If the same DLL is used for spoofing detection as was used for the RF filter system identification and if the same PRN code is being tracked as one of the codes that was used for system identification, then this relative timing should be well known. In other situations, a calibration must be made of this relative timing.

This relative timing calibration is the semi-codeless equivalent of the codeless timing offset δt_{ab}. Recall that this latter offset is used in Eq. (17) of the codeless spoofing detection calculations. In the semi-codeless case, each receiver has its own independent relative timing of the W chips and their associated filtered P chips relative to the C/A-code DLL tracking point. The effects of changes in the semi-codeless relative timing assumptions have been examined experimentally, and results from this study are reported in Section V.

The P(Y) code amplitude A_p in Eq. (32b) must be deduced from the C/A code amplitude in order to define the estimation problem. The formula for this amplitude is

$$A_p = \frac{2}{\overline{N}_{accum}} \sqrt{\frac{L_p}{L_{fca}}} 10^{0.4/20} A_{IQ} \tag{33}$$

Recall that A_{IQ} is the C/A-code prompt accumulation amplitude from Eq. (23a). This equation is the semi-codeless equivalent of codeless Eq. (27). The factor $2/\overline{N}_{accum}$ transforms from accumulation amplitude to carrier amplitude. The square-root in Eq. (33) arises because this is an amplitude equation rather than a power equation. The tern L_{fpy} is missing from Eq. (33) because this component of power loss is modeled not by the carrier amplitude A_p but by the filtered time histories $P_{fwj}(t)$. The power of 10 in Eq. (33) accounts for the differing losses of the C/A code power and the P(Y) code power in the +/-10.23 MHz bandwidth in comparison to their infinite-bandwidth powers.

Optimization of the cost function in Eq. (32b) is performed iteratively. The ad hoc iteration strategy relies on the following fact: The overlaps of the neighboring non-zero portions of the $P_{fwj}(t)$ functions are small relative to the time spans over which they have appreciable non-zero values. This fact is evident in the bottom plot of Fig. 2. Therefore, reasonable first-cut estimates of the W-chips are:

$$\hat{w}_j = sign[\sum_{i=i_{minj}}^{i_{maxj}} P_{fwj}(t_i) y_{qi}] \quad \text{for } j = 1, ..., K \tag{34}$$

where i_{minj} and i_{maxj} are, respectively, the minimum and maximum sample indices i for which $P_{fwj}(t_i)$ is appreciably different from zero. The *sign*[] function in Eq. (34) returns a +1 for a positive or zero input argument and a -1 for a negative input argument.

If there were no non-zero overlaps between neighboring $P_{fwj}(t)$ functions, then the W-chip estimates in Eq. (34) would be optimal. In the presence of minor overlaps, the following heuristic iteration should converge to the optimal solution

$$\hat{w}_j = sign\Big(\sum_{i=i_{minj}}^{i_{maxj}} P_{fwj}(t_i)\{y_{qi} - \tfrac{1}{2} A_p[\sum_{l=j-L}^{j-1} \hat{w}_l^{old} P_{fwl}(t_i) + \sum_{l=j+1}^{j+L} \hat{w}_l^{old} P_{fwl}(t_i)]\}\Big) \quad \text{for } j = 1, ..., K \tag{35}$$

where $\hat{w}_l^{old}$ for $l = 1, ..., K$ are the chip estimates from the previous iteration and where L is the number of neighboring chips on each side of w_j whose $P_{fwl}(t)$

functions have appreciable non-zero overlap with the $P_{fwj}(t)$ function. For the cases considered in this paper, L = 2 has been selected as a conservative estimate. The value L = 1 probably would have sufficed, and it would have saved computation time. Note: for j values less than L+1 or greater than K-L, one or the other of the summations over l in Eq. (35) must be truncated appropriately.

The iterative optimization starts by generating W-chip estimates using Eq. (34). Next, it updates its W-chip estimates using Eq. (35). It repeats the evaluations in Eq. (35) using the $\hat{w}_j$ values from the previous iteration as its $\hat{w}_l^{old}$ values for each new iteration. It terminates the iterative process when $\hat{w}_j = \hat{w}_j^{old}$ for all j = 1, ..., K.

The rate of convergence of the iterations is dependent on the receiver design through its RF filter's distorting effect on the $P_{fwj}(t)$ functions. Experience with this algorithm's convergence properties has been gained for the receivers used in the present study. The algorithm always terminated in 7 or fewer iterations, and the average number of iterations was less than 5. The required number of iterations should decrease for a higher bandwidth RF front-end.

The *sign*[] function in Eqs. (34) and (35) ensures that the resulting W-chip estimates are either +1-valued or -1-valued. Thus, the constraint in Eq. (32c) is enforced by this heuristic optimization procedure. Because of these constraints, the resulting W-chip estimates are termed "hard" estimates.

B. Probability of W-Chip Correctness and "Soft" W-Chip Estimates

One could directly correlate the "hard" +1/-1 W-chip estimates of the previous sub-section between Receivers A and B in order to form a spoofing detection statistic. It is well known, however, that "hard" W-chip estimates are not optimal when performing semi-codeless tracking of the L2 signal in dual-frequency receivers [17]. Therefore, it seems wise to develop "soft" W-chip estimates in the hopes of improving the detection power of the semi-codeless statistic.

Reasonable "soft" W-chip estimates can be derived based on the probabilities of correctness of the corresponding "hard" estimates. Assuming that the random errors in Eqs. (6a) and (6b) are described by the statistics in Eqs. (7a)-(7c), the probability that $\hat{w}_j$ is correct is approximately

$$\mathcal{P}\{\hat{w}_j\} = \left(1 + exp\{2[J(\hat{w}_1,...,\hat{w}_{j-1},\hat{w}_j,\hat{w}_{j+1},...,\hat{w}_K) - J(\hat{w}_1,...,\hat{w}_{j-1},-\hat{w}_j,\hat{w}_{j+1},...,\hat{w}_K)]/\sigma_{RF}^2\}\right)^{-1} \quad \text{for } j = 1, ..., K \tag{36}$$

This probability formula is based on the assumption that the normalized cost function $2J(w_1,...,w_K)/\sigma_{RF}^2$ is the negative log likelihood of the W chips and, up to a constant offset, that it is also the negative logarithm of the a posteriori probability of the W chips. The log-likelihood assumption is consistent with Eqs. (6a)-(7b). The equivalence between the log-likelihood function and the log a posteriori probability is a consequence of Bayes' formula and the uniform prior assumption that both possible W-chip values are equally likely [22].

The only difference between the two cost terms in the exponential in Eq. (36) is the sign of the estimate of w_j. The value $\hat{w}_j$ produces a lower cost than does the value $-\hat{w}_j$. Therefore, the exponent in Eq. (36) is guaranteed to be negative, and $\mathcal{P}\{\hat{w}_j\}$ is guaranteed to be greater than 0.5.

The probability formula in Eq. (36) is inexact for the situation of significant non-zero overlap between neighboring $P_{fwj}(t)$ functions. In this situation, the exact formula would involve a normalized sum over all possible combinations of +1/-1 values of the chips other than $\hat{w}_j$. Given the small overlaps, however, the approximation in Eq. (36) is reasonable. In the event of larger overlaps, an appropriate alternative to Eq. (36) could be developed. It has been omitted due to the lack of a perceived need and for the sake of brevity.

The RF sample variance from Eq. (25) can be used in Eq. (36), but an alternate value can be deduced directly from the optimal value of the Eq.-(32b) cost. That value is

$$(\sigma_{RF}^2)_{alt} = \frac{4J(\hat{w}_1,...,\hat{w}_K)}{M} \tag{37}$$

This formula is consistent with the statistical assumptions of Eqs. (6a)-(7c). This alternative form of σ_{RF}^2 has been used to generate results in the present paper. Although never very much different than the value computed in Eq. (25), this alternate value appears to be a more reliable indicator of the random noise levels in Eqs. (6a) and (6b).

The probability of a correct $\hat{w}_j$ estimate from Eq. (36) can be used to compute the probability that the correct estimate is w_j = +1. This latter probability is

$$\mathcal{P}_{j(+)} = \tfrac{1}{2} + \hat{w}_j[\mathcal{P}\{\hat{w}_j\} - \tfrac{1}{2}] \tag{38}$$

The "soft" estimate of w_j used in this paper is its conditional expectation value based on the receiver's quadrature baseband data, y_{q1}, y_{q2}, y_{q3}, ..., y_{qi}, ... It is:

$$\hat{w}_{sj} = E\{w_j \mid y_{q1}, y_{q2}, y_{q3}, \ldots\} = 2\mathcal{P}_{j(+)} - 1 \tag{39}$$

This estimate is guaranteed lie in the range -1 $\le \hat{w}_{sj} \le$ 1.

It takes on the end-point values if $\mathcal{P}_{j(+)}$ takes on the value 0 or 1. It equals 0 if $\mathcal{P}_{j(+)}$ = 0.5. Thus, $\hat{w}_{sj}$ is a reasonable "soft" estimate of w_j.

As in the case of codeless spoofing detection, the foregoing optimal estimation algorithm and analyses ignore the fact that the noise in the quadrature baseband-mixed signal is not purely white noise. If the violation of the whiteness assumption were significant, then the derivations of this subsection and the previous subsection would have to be modified in order to account for the noise correlations. The primary change would be to modify the cost function in Eq. (32b) to include cross-products between Eq.-(6) error terms with sample indices i that were near each other. The resulting modified cost would remain the negative log likelihood of the W chips after rescaling by $2/\sigma_{RF}^2$. These modifications to the W-chips estimation algorithm and the associated analyses appear to be unnecessary for the receivers considered in this paper, and they have been omitted for the sake of brevity.

C. Spoofing Detection via Correlation of "Soft" W-Chip Estimates

The soft W-chip estimates can be used to develop a spoofing test statistic that is Neyman-Pearson optimal in the limit of weak signals in Receiver B. The derivation of this test statistic assumes the availability of $\mathcal{P}_{aj(+)}$, reference Receiver A's calculated probability that chip w_j is positive. The derivation also assumes that the filtered P-code functions corresponding to any two distinct W-chips, say w_j and w_l, have insignificant overlap of their non-zero parts so that they are orthogonal. In other words, it assumes that

$$\sum_{i=1}^{M} P_{fwj}(t_{bi})P_{fwl}(t_{bi}) = 0 \quad \text{for all } (j,l) \text{ such that } j \neq l \quad (40)$$

This assumption allows a re-scaled version of the cost function in Eq. (32b) for Receiver B to be re-written in the form:

$$\begin{aligned}\tilde{J}_b(w_1,...,w_K) &= 2J_b(w_1,...,w_K)/\sigma_{RFb}^2 \\ &= \tilde{J}_{b0} - \sum_{j=1}^{K}\mu_{bj}w_j + \tfrac{1}{2}\sum_{j=1}^{K}\xi_{bj}^2 \end{aligned} \quad (41)$$

where the newly defined quantities on the right-hand side of Eq. (41) are

$$\tilde{J}_{b0} = \frac{1}{\sigma_{RFb}^2}\sum_{j=1}^{K} y_{qbi}^2 \quad (42a)$$

$$\mu_{bj} = \frac{A_{pb}}{\sigma_{RFb}^2}\sum_{i=i_{bminj}}^{i_{bmaxj}} P_{fwj}(t_{bi})y_{qbi} \quad \text{for } j = 1, ..., K \quad (42b)$$

$$\xi_{bj}^2 = \frac{A_{pb}^2}{2\sigma_{RFb}^2}\sum_{i=i_{bminj}}^{i_{bmaxj}} P_{fwj}^2(t_{bi}) \quad \text{for } j = 1, ..., K \quad (42c)$$

The re-scaled cost function in Eq. (41) is the negative log probability density of the Receiver B quadrature baseband mixed measurements y_{qb1}, ..., y_{qbM}. The corresponding probability density is conditioned on the W-chip values w_1, ..., w_K and on the P(Y)-code amplitude A_{pb}. Therefore, conditional probability density functions of the measurements for the un-spoofed and spoofed cases can be derived by using Eq. (41) along with the known W-chip probabilities from Receiver A. For the un-spoofed hypothesis, H_0, this conditional density function is

$$\begin{aligned} p(y_{qb1},\ldots,y_{qbM} \mid H_0) = \\ c\exp(-\tilde{J}_{b0} - \tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\times \\ \prod_{j=1}^{K}\{\mathcal{P}_{aj(+)}\exp(\mu_{bj}) \\ +[1-\mathcal{P}_{aj(+)}]\exp(-\mu_{bj})\} \end{aligned} \quad (43)$$

where c is a normalizing constant. The nominal spoofed hypothesis, H_1, assumes that there is nothing besides receiver noise on the quadrature channel. This hypothesis is the equivalent of using a Receiver-B P(Y) amplitude of A_{pb} = 0. The measurement probability density for this case is

$$p(y_{qb1},\ldots,y_{qbM} \mid H_1) = c\exp(-\tilde{J}_{b0}) \quad (44)$$

The Neyman Pearson spoofing detection test statistic in this case is

$$\begin{aligned}\gamma_{sopt} &= \frac{p(y_{qb1},\ldots,y_{qbM} \mid H_0)}{p(y_{qb1},\ldots,y_{qbM} \mid H_1)} \\ &= \exp(-\tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\times \\ &\quad \prod_{j=1}^{K}\{\mathcal{P}_{aj(+)}\exp(\mu_{bj}) + [1-\mathcal{P}_{aj(+)}]\exp(-\mu_{bj})\} \end{aligned} \quad (45)$$

If this statistic lies above a certain threshold value, then the un-spoofed hypothesis H_0 is accepted. Otherwise, the spoofed hypothesis H_1 is accepted, and a spoofing alert is issued.

Using algebra and hyperbolic trigonometry, one can derive equivalent forms of γ_{sopt} that are useful for analysis:

$$\begin{aligned}\gamma_{sopt} &= \exp(-\tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\prod_{j=1}^{K}\{\cosh(\mu_{bj}) \\ &\qquad +[2\mathcal{P}_{aj(+)}-1]\sinh(\mu_{bj})\} \\ &= \exp(-\tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\prod_{j=1}^{K}[\cosh(\mu_{bj}) \end{aligned}$$

$$+\hat{w}_{saj}\sinh(\mu_{bj})]$$

$$= \exp(-\tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\prod_{j=1}^{K}\frac{[1+\hat{w}_{saj}\tanh(\mu_{bj})]}{\sqrt{1-\tanh^2(\mu_{bj})}}$$

$$= \exp(-\tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\prod_{j=1}^{K}\frac{[1+\hat{w}_{saj}\hat{w}_{sbj}]}{\sqrt{1-\hat{w}_{sbj}^2}} \qquad (46)$$

The transition from the first to the second line of Eq. (46) makes use of the "soft" W-chip formula in Eq. (39) as applied for Receiver A. The transition from the second-to-last line to the last line of Eq. (46) uses the fact that

$$\hat{w}_{sbj} = 2\mathcal{P}_{bj(+)} - 1$$

$$= 2\{1+\exp[\tilde{J}_b(\hat{w}_1,\ldots,\hat{w}_{j-1},+1,\hat{w}_{j+1},\ldots,\hat{w}_K) - \tilde{J}_b(\hat{w}_1,\ldots,\hat{w}_{j-1},-1,\hat{w}_{j+1},\ldots,\hat{w}_K)]\}^{-1} - 1$$

$$= 2\left(\frac{1}{1+e^{-2\mu_{bj}}}\right)-1$$

$$= \frac{1-e^{-2\mu_{bj}}}{1+e^{-2\mu_{bj}}}$$

$$= \frac{e^{\mu_{bj}}-e^{-\mu_{bj}}}{e^{\mu_{bj}}+e^{-\mu_{bj}}}$$

$$= \tanh(\mu_{bj}) \qquad (47)$$

An equivalent detection statistic is

$$\gamma_{salt} = \log(\gamma_{sopt}) + \tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2$$

$$= \sum_{j=1}^{K}[\log(1+\hat{w}_{saj}\hat{w}_{sbj}) - \tfrac{1}{2}\log(1-\hat{w}_{sbj}^2)] \qquad (48)$$

One can derive a low-power approximation for Receiver B by expanding this statistic in a Taylor series about the values $\hat{w}_{sbj} = 0$. The result, to second order in $\hat{w}_{sbj}$, is

$$\gamma_{salt} \cong \sum_{j=1}^{K}[\hat{w}_{saj}\hat{w}_{sbj} + \tfrac{1}{2}(1-\hat{w}_{saj}^2)\hat{w}_{sbj}^2] \qquad (49)$$

Before proceeding further, it is worthwhile to examine this spoofing detection statistic. The first term in its sum is the product of the "soft" W-chip estimates from the two receivers. This whole analysis has been developed with the goal of deriving a detection statistic of this form. The second term, however, looks mostly for power in the P(Y) signal of Receiver B. If Receiver A has a very high carrier-to-noise ratio, then each of its $\hat{w}_{saj}$ estimates will be very near +1 or -1. In this case, the second term on the right-hand side of Eq. (49) will be insignificant because $(1-\hat{w}_{saj}^2)$ will be nearly zero. In other cases, however, the second term will indicate that there is a valid un-spoofed signal whenever it detects appreciable P(Y) signal power in Receiver B.

The proposed spoofing detection statistic in Eq. (49) leads to a dangerous situation. Suppose that a spoofer were to put P(Y) pseudo-code on the quadrature channel with randomly chosen false w_j chips. In this case, the first term in Eq. (49) would contribute zero to the spoofing statistic, on average, but the second term could contribute a significant positive component, possibly enough to cause a missed detection.

This situation can be resolved by considering an alternate spoofing hypothesis, H_1'. The spoofer generates a false P(Y) code with randomly chosen +1/-1 values for its W chips. Suppose that it assigns equal probabilities to the potential w_j values +1 and -1. The probability density of the Receiver B quadrature samples conditioned on spoofing hypothesis H_1' then becomes

$$p(y_{qb1},\ldots,y_{qbM}\mid H_1') = c\exp(-\tilde{J}_{b0} - \tfrac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\times \prod_{j=1}^{K}\cosh(\mu_{bj}) \qquad (50)$$

and the optimal spoofing detection statistic becomes

$$(\gamma_{sopt})' = \prod_{j=1}^{K}[1+\hat{w}_{saj}\hat{w}_{sbj}] \qquad (51)$$

Its logarithmic approximation to second order in $\hat{w}_{sbj}$ becomes

$$(\gamma_{salt})' = \log[(\gamma_{sopt})'] \cong \sum_{j=1}^{K}\hat{w}_{saj}\hat{w}_{sbj}(1-\tfrac{1}{2}\hat{w}_{saj}\hat{w}_{sbj}) \qquad (52)$$

This statistic does not suffer from the problem of too much probability of a missed detection because of false P(Y) code on the spoofed quadrature signal.

In order to simplify further analysis, the spoofing detection statistic is truncated to retain only the first-order terms in $\hat{w}_{sbj}$. Thus, the final statistic chosen for semi-codeless "soft" W-chips spoofing detection is

$$\gamma_s = \sum_{j=1}^{K}\hat{w}_{saj}\hat{w}_{sbj} \qquad (53)$$

This spoofing detection statistic is optimal in the limit of very low P(Y) power in defended Receiver B. It is optimal for any level of P(Y) power in reference Receiver A. At higher P(Y) powers in Receiver B, this statistic is sub-optimal. As for locally most powerful detection tests, one can normally tolerate sub-optimality in the case of a stronger signal.

This first-order truncation of the logarithmic Taylor series avoids the possibility that false W chips could defeat the detection. Nevertheless, the foregoing analysis of the

false W-chips case has been worthwhile. It suggests additional analyses that should be carried out in the future. The needed analyses would explore whether there are sophisticated spoofing attack scenarios which could overcome the defenses that are developed in the present paper.

D. Probability Analysis and Detection Threshold Design for the "Soft" W-Chips Test Statistic

At the point of spoofing detection, the random variability of the γ_s spoofing detection statistic in Eq. (53) comes entirely from the variability of the $\hat{w}_{sbj}$ soft W-chip estimates. Although the $\hat{w}_{saj}$ estimates will have been affected by random noise in reference Receiver A, these will be known quantities at the time of spoofing detection in Receiver B. The randomness in $\hat{w}_{sbj}$ can be characterized by using its formula in terms of μ_{bj}, Eq. (47), coupled with the probability density of μ_{bj}.

The randomness in μ_{bj} comes from the random noise in the y_{qbi} quadrature baseband mixed samples, as per Eq. (42b). Therefore, μ_{bj} follows a Gaussian distribution in each of the 3 possible situations: a) the signal is not being spoofed and the true w_j is +1, b) the signal is not being spoofed and the true w_j is -1, or c) the signal is being spoofed so that y_{qbi} contains only random noise. The mean and variance of μ_{bj} can be deduced in each of these situations by substituting the models in Eqs. (6b) and (9) into Eq. (42b) and by taking appropriate expectations based on the statistical models in Eq. (7b). Given these means and variances and given the w_j = +1 chip probabilities from reference Receiver A, the un-spoofed and spoofed conditional probability densities for μ_{bj} are:

$$p(\mu_{bj} \mid H_0) = \frac{1}{\sqrt{2\pi}\xi_{bj}} \{ \mathcal{P}_{aj(+)} e^{-(\mu_{bj} - \xi_{bj}^2)/(2\xi_{bj}^2)} + [1 - \mathcal{P}_{aj(+)}] e^{-(\mu_{bj} + \xi_{bj}^2)/(2\xi_{bj}^2)} \} \quad (54a)$$

$$p(\mu_{bj} \mid H_1) = \frac{1}{\sqrt{2\pi}\xi_{bj}} e^{-\mu_{bj}^2/(2\xi_{bj}^2)} \quad (54b)$$

In preparation for computing the means and variances of the spoofing detection statistic γ_s under the two hypotheses, one can compute the corresponding expectation values of $\hat{w}_{sbj}$ and $\hat{w}_{sbj}^2$. These are computed by using the model for $\hat{w}_{sbj}$ in terms of μ_{bj} in Eq. (47) and the probability density functions in Eqs. (54a) and (54b). The results are:

$$E\{\hat{w}_{sbj} \mid H_0\} = \int_{-\infty}^{\infty} tanh(\mu_{bj}) p(\mu_{bj} \mid H_0) d\mu_{bj} = \hat{w}_{saj} q(\xi_{bj}) \quad (55a)$$

$$E\{\hat{w}_{sbj}^2 \mid H_0\} = \int_{-\infty}^{\infty} tanh^2(\mu_{bj}) p(\mu_{bj} \mid H_0) d\mu_{bj} = q(\xi_{bj}) \quad (55b)$$

$$E\{\hat{w}_{sbj} \mid H_1\} = \int_{-\infty}^{\infty} tanh(\mu_{bj}) p(\mu_{bj} \mid H_1) d\mu_{bj} = 0 \quad (55c)$$

$$E\{\hat{w}_{sbj}^2 \mid H_1\} = \int_{-\infty}^{\infty} tanh^2(\mu_{bj}) p(\mu_{bj} \mid H_1) d\mu_{bj} = r(\xi_{bj}) \quad (55d)$$

where the two special functions $q(\xi)$ and $r(\xi)$ are defined as follows:

$$q(\xi) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} tanh[\xi(\xi + \eta)] exp\{-0.5\eta^2\} d\eta \quad (56a)$$

$$r(\xi) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} tanh^2(\xi\eta) exp\{-0.5\eta^2\} d\eta \quad (56b)$$

The analyses that derive Eqs. (55a)-(56b) involve changes of dummy integration variables from μ_{bj} to η. They require two proofs involving modifications to the integral in Eq. (56a). One proof demonstrates that a change from $(\xi+\eta)$ to $(-\xi+\eta)$ in the *tanh* function argument of the integrand causes the resulting formula to yield $-q(\xi)$. The other proof demonstrates that a squaring of the *tanh* function causes the resulting formula to remain equal to $q(\xi)$.

The special functions defined in Eqs. (56a) and (56b) can be evaluated approximately via numerical integration. For purposes of this paper, they have been pre-computed off-line on an extensive grid of ξ points. Afterwards, their pre-computed values have been used in an interpolation scheme in order to provide a quick, practical means of evaluating them.

The results in Eqs. (55a)-(55d) can be used to compute the corresponding means and variances of the spoofing statistic γ_s. These calculations rely in the assumption that any two "soft" W-chip estimates $\hat{w}_{sbj}$ and $\hat{w}_{sbl}$ for $j \neq l$ are sampled from independent distributions.

$$\bar{\gamma}_{s|H0} = E\{\gamma_s \mid H_0\} = \sum_{j=1}^{K} \hat{w}_{saj}^2 q(\xi_{bj}) \quad (57a)$$

$$\sigma_{\gamma s|H0}^2 = E\{\gamma_s^2 \mid H_0\} - \bar{\gamma}_{s|H0}^2 = \sum_{j=1}^{K} \hat{w}_{saj}^2 q(\xi_{bj})[1 - \hat{w}_{saj}^2 q(\xi_{bj})] \quad (57b)$$

$$\bar{\gamma}_{s|H1} = E\{\gamma_s \mid H_1\} = 0 \quad (57c)$$

$$\sigma_{\gamma s|H1}^2 = E\{\gamma_s^2 \mid H_1\} = \sum_{j=1}^{K} \hat{w}_{saj}^2 r(\xi_{bj}) \quad (57d)$$

As in the codeless case, the means and variances in Eqs. (57a)-(57d) are used to design and analyze the spoofing detection test. This involves the probability density functions $p(\gamma_s|H_0)$ and $p(\gamma_s|H_1)$. These are non-Gaussian due to the non-Gaussian nature of the $\hat{w}_{sbj}$ distributions. Given that many such random quantities affect the correlation statistic summation in Eq. (53), however, the central limit theorem can be invoked in order to argue that Gaussian approximations of these two distributions are reasonable.

Given the false-alarm probability α_{FA} and the Gaussian assumption, the spoofing detection threshold γ_{sth} is the solution of the following equation:

$$\alpha_{FA} = \int_{-\infty}^{\gamma_{sth}} p(\gamma_s|H_0)d\gamma_s = \frac{1}{\sqrt{2\pi}\sigma_{\gamma s|H0}} \int_{-\infty}^{\gamma_{sth}} exp\{-\frac{(\gamma_s - \bar{\gamma}_{s|H0})^2}{2\sigma^2_{\gamma s|H0}}\}d\gamma_s \quad (58)$$

If $\gamma_s \geq \gamma_{sth}$, then hypothesis H_0 is accepted: there is no spoofing. If $\gamma_s < \gamma_{sth}$, then hypothesis H_1 is accepted, and a spoofing alert is issued. The probability of successful spoofing detection is:

$$\begin{aligned} \mathcal{P}_{detect} &= \int_{-\infty}^{\gamma_{sth}} p(\gamma_s|H_1)d\gamma_s \\ &= \frac{1}{\sqrt{2\pi}\sigma_{\gamma s|H1}} \int_{-\infty}^{\gamma_{sth}} exp\{-\frac{\gamma_s^2}{2\sigma^2_{\gamma s|H1}}\}d\gamma_s \\ &= 1 - \mathcal{P}_{misdet} \end{aligned} \quad (59)$$

As in the codeless case, the un-spoofed H_0 hypothesis is atypical because the test statistic's mean value is non-zero. The accuracy of the mean valued given in Eq. (57a) is dependent on the accuracy of the modeled P(Y)-code carrier amplitudes in the two receivers, A_{pa} and A_{pb} as computed using Eq. (33). Any errors in these calculations will lead to errors in the predicted mean $\bar{\gamma}_{s|H0}$ and in the detection threshold γ_{sth}. Any such errors will distort the false-alarm probability and the probability of detection away from the values given in Eqs. (58) and (59).

E. Offline Analysis of "Soft" W-Chips Spoofing Detection

The results in Eqs. (57a)-(59) are suitable for on-line spoofing detection using the "soft" W-chips technique, but not for off-line analysis. In an off-line analysis, the required $\hat{w}_{saj}$ estimates from reference Receiver A are not available, nor are the ξ_{bj} values available from defended Receiver B.

An offline analysis replaces actual values of $\hat{w}_{saj}$ with a statistical model of their distribution. This model uses the *tanh* formula in Eq. (47) to represent $\hat{w}_{saj}$ as a function of μ_{aj}. Along the lines of Eq. (54a), this analysis models the probability distribution of μ_{aj} under two conditions as follows:

$$p(\mu_{aj} \mid w_j = +1) = \frac{1}{\sqrt{2\pi}\xi_{aj}} e^{-(\mu_{aj} - \xi_{aj}^2)/(2\xi_{aj}^2)} \quad (60a)$$

$$p(\mu_{aj} \mid w_j = -1) = \frac{1}{\sqrt{2\pi}\xi_{aj}} e^{-(\mu_{aj} + \xi_{aj}^2)/(2\xi_{aj}^2)} \quad (60b)$$

A similar statistical model applies to $\hat{w}_{sbj}$ and μ_{bj}, except that the μ_{bj} probability densities also must be conditioned on the un-spoofed and spoofed hypotheses. For the un-spoofed hypothesis, the two conditional probabilities are

$$p(\mu_{bj} \mid w_j = +1, H_0) = \frac{1}{\sqrt{2\pi}\xi_{bj}} e^{-(\mu_{bj} - \xi_{bj}^2)/(2\xi_{bj}^2)} \quad (61a)$$

$$p(\mu_{bj} \mid w_j = -1, H_0) = \frac{1}{\sqrt{2\pi}\xi_{bj}} e^{-(\mu_{bj} + \xi_{bj}^2)/(2\xi_{bj}^2)} \quad (61b)$$

In the case of spoofing, Eq. (54b) still gives the correct μ_{bj} probability density.

Expected values of ξ_{aj} and ξ_{bj} can be computed based on the respective P(Y) carrier-to-noise ratios of the two receivers.

$$\bar{\xi}_a = \sqrt{\frac{2(C/N_0)_{pya}}{f_{wchip}}} \quad (62a)$$

$$\bar{\xi}_b = \sqrt{\frac{2(C/N_0)_{pyb}}{f_{wchip}}} \quad (62b)$$

where f_{wchip} = 480,000 Hz is the nominal mean chipping rate of the W chips.

Given the forgoing statistical models, the means and variances in Eqs. (57a)-(57d) can be re-computed a priori. This computation accounts for the correlation between μ_{aj} and μ_{bj} that is caused by their both being conditioned on the same w_j chip value for any given case. After a lengthy derivation, the results are

$$\bar{\gamma}_{s|H0} = Kq(\bar{\xi}_a)q(\bar{\xi}_b) \quad (63a)$$

$$\sigma^2_{\gamma s|H0} = Kq(\bar{\xi}_a)q(\bar{\xi}_b)[1 - q(\bar{\xi}_a)q(\bar{\xi}_b)] \quad (63b)$$

$$\bar{\gamma}_{s|H1} = E\{\gamma_s \mid H_1\} = 0 \quad (63c)$$

$$\sigma^2_{\gamma s|H1} = Kq(\bar{\xi}_a)r(\bar{\xi}_b) \quad (63d)$$

The number of W chips in any given correlation statistic

is $K = T_{corr} f_{wchip}$, where T_{corr} is the duration of the correlation interval.

The computed a priori statistics in Eqs. (63a)-(63d) can be used in Eqs. (58) and (59) in order to analyze the expected performance of a proposed spoofing detection test. The analysis starts with expected P(Y)-code carrier-to-noise ratios in the two receivers $(C/N_0)_{pya}$ and $(C/N_0)_{pyb}$. These can be computed based on expected received power levels and on the RF filter losses that are modeled by Eq. (15). These carrier-to-noise ratios and the designed correlation interval T_{corr} serve as inputs to the calculations in Eqs. (62a)-(63d). Next, the results of these calculations and the designed false-alarm probability α_{FA} are input to the computations in Eqs. (58) and (59). The final results are the expected spoofing detection threshold and the corresponding probability of detection.

Analogous calculations can be carried out for the codeless spoofing detection test by using Eqs. (29a)-(31). These computations must use the following number of samples in order to define a codeless statistic of equivalent duration: $M = T_{corr}/\Delta t$.

Figure 3 compares the power of codeless and semi-codeless spoofing detection for correlation intervals T_{corr} that range along the horizontal axis from 0.01 sec to 10 sec. Both curves assume P(Y)-code carrier-to-noise ratios of 35 dB-Hz in both receivers, and both curves use a false-alarm probability of $\alpha_{FA} = 0.0001$ (0.01%). Also plotted are points for two actual cases associated with the results in Section V, one for PRN 13 plotted with triangles and one for PRN 17 plotted with squares. Both of these actual cases have carrier-to-noise ratios larger than 35 dB-Hz, despite the use of 2.4-2.6 MHz filter bandwidths in the RF front. Therefore, the carrier-to-noise ratios assumed for the two curves are somewhat conservative.

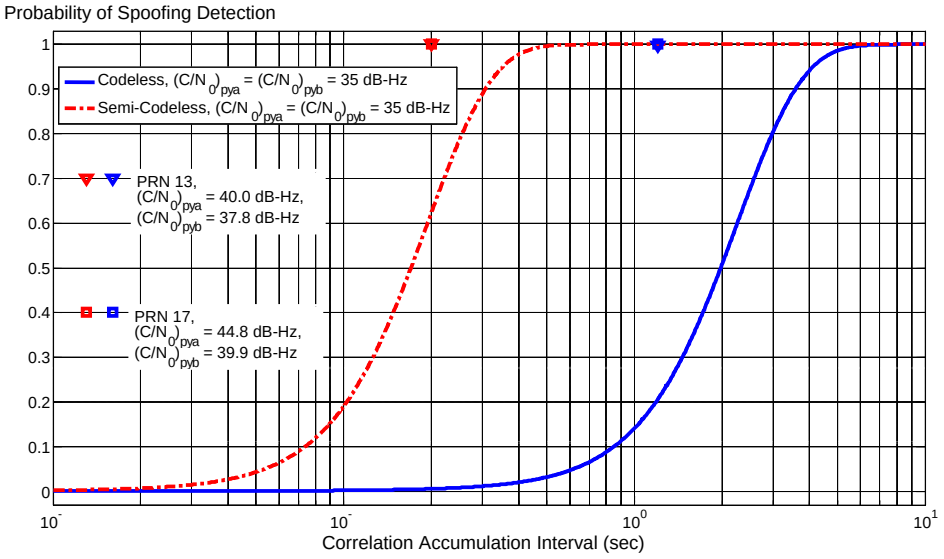

Fig. 3. Comparison of codeless and semi-codeless spoofing detection power as functions of correlation interval for a false alarm probability of 0.01%.

Figure 3 indicates that the codeless technique should be able to detect spoofing reliably ($\mathcal{P}_{detect} = 0.986$) with a cross-correlation interval of 5 sec. This same detection probability can be achieved by the semi-codeless technique when using intervals of only 0.43 sec.

Thus, the semi-codeless method is more than 10 times as efficient as the codeless method in its use of data to achieve a given power of detection. This efficiency has two advantages. First, spoofing detection can occur with a lower latency. Second, the required communication bandwidth between Receivers A and B can be reduced for a given frequency of calculation of spoofing detection test statistics. A further bandwidth reduction occurs for the semi-codeless technique because it only has to send data at the 480 KHz W-chips chipping rate. The codeless technique must send data at the RF sample frequency, $1/\Delta t$ Hz. This latter frequency can be on the order of 6 MHz or higher. Of course, each W-chip estimate might entail the transmission of more bits than each raw quadrature RF sample. This means that the bandwidth requirement of the codeless technique is not quite so high relative to the semi-codeless technique as is indicated by a simple comparison of sampling and chipping rates.

The data points on Fig. 3 for PRNs 13 and 17 indicate that practical spoofing detection can be performed with correlation intervals of 1.2 sec for the codeless technique (blue triangle and square) and 0.2 sec for the semi-codeless technique (red triangle and square). Given the very high probabilities of detection for these data points, it would be possible to reduce the cross-correlation intervals and still retain adequate detection power. These results are surprising and encouraging: Despite using RF front-ends that attenuate the P(Y) code power by 5.4 to 5.6 dB, despite the corresponding marked distortion of the P(Y) code, and despite the squaring losses in the detection calculations, practical spoofing detection can be performed using this technique.

Another surprising point of Fig. 3 is that the semi-codeless technique offers an order of magnitude improvement in the spoofing detection efficiency. Semi-

codeless techniques derive their advantage from a narrowing of signal bandwidth and a corresponding increase of the processing gain prior to squaring. Given that the signal bandwidth is already very narrow at the output of the RF filter, it was not obvious at the outset of this study that a further bandwidth reduction in the semi-codeless matched filter would afford any additional processing gain.

The semi-codeless technique has one significant disadvantage: It requires much more signal processing. First, the distorted $P_{fwj}(t)$ P-chip functions must be computed for each RF sample of each W-chip. These function evaluations are complicated and must be performed on-line. Additionally, the W-chip optimization problem in Eqs. (32a)-(32c) must be solved on-line. A good area for future research would be to seek efficient means of carrying out these complex calculations in real-time.

F. Alternate Semi-Codeless Spoofing Detection Statistic

There is an alternative semi-codeless detection statistic. It is an optimal Neyman-Pearson statistic in the limit of very high carrier-to-noise ratio at reference Receiver A. It is also optimal in the limit of very low carrier-to-noise ratio at defended Receiver B.

If $(C/N_0)_{pya}$ is very large, then the $\hat{w}_{saj}$ values will be either +1 or -1, and the corresponding probabilities $\mathcal{P}_{aj(+)}$ in Eq. (45) will be either 1 or 0. $(C/N_0)_{pya}$ values that approximate this limit can be achieved by using a high-gain antenna at reference Receiver A or by appropriate averaging of imprecise results from many secure reference receivers.

Given the high $(C/N_0)_{pya}$ limit, the analysis in Eqs. (43)-(48) can be modified to show that

$$\gamma_{sopt} = exp(-\frac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2)\prod_{j=1}^{K}exp(\hat{w}_{saj}\mu_{bj}) \tag{64}$$

and that

$$\gamma_{salt} = log(\gamma_{sopt}) + \frac{1}{2}\sum_{l=1}^{K}\xi_{bl}^2 = \sum_{j=1}^{K}\hat{w}_{saj}\mu_{bj} \tag{65}$$

An alternate analysis demonstrates that this same approximate is valid in the low $(C/N_0)_{pyb}$ limit. One starts with the γ_s approximation of Eq. (53), which is valid in this limit. If one substitutes the formula $\hat{w}_{sbj} = tanh(\mu_{bj})$ from Eq. (47) into Eq. (53) and if one uses the first-order approximation $tanh(\mu_{bj}) \cong \mu_{bj}$, then one arrives at the same formula for γ_{salt} as appears on the extreme right-hand side of Eq. (65). This *tanh*() approximation is valid for $|\mu_{bj}| << 1$. The limit $|\mu_{bj}| << 1$ holds true if $(C/N_0)_{pyb}$ is small. This latter result is implied by the two μ_{bj} mean values from Eq. (54a), $\pm\xi_{bj}^2$, if one considers the relationship of these mean values to the carrier-to-noise ratio, as given in Eq. (62b).

The accumulation statistic in Eq. (65) can be re-cast into a more conventional form by using Eq. (42b) to eliminate μ_{bj}:

$$\begin{aligned}\gamma_{salt} &= \sum_{j=1}^{K}\hat{w}_{saj}[\frac{A_{pb}}{\sigma_{RFb}^2}\sum_{i=i_{bminj}}^{i_{bmaxj}}P_{fwj}(t_{bi})y_{qbi}] \\ &= \frac{A_{pb}}{\sigma_{RFb}^2}\sum_{i=1}^{M}y_{qbi}[\sum_{j=1}^{K}\hat{w}_{saj}P_{fwj}(t_{bi})] \\ &= \frac{A_{pb}}{\sigma_{RFb}^2}\sum_{i=1}^{M}y_{qbi}\hat{P}_{Yfa}(t_{bi})\end{aligned} \tag{66}$$

The second line of Eq. (66) has been derived from the first line by extending the limits of the second summation on the first line all the way from $i = 1$ to $i = M$. Afterwards, the second line cay be derived by a simple interchange of the order of summation. This extension of the summation indices is allowable because the filtered P-chips function $P_{fwj}(t_{bi})$ has already been assumed to be zero-valued for $i < i_{bminj}$ and for $i_{bmaxj} < i$. The last line of Eq. (66) is derived by making the following definition of the approximate filtered P(Y) code:

$$\hat{P}_{Yfa}(t) = \sum_{j=1}^{K}\hat{w}_{saj}P_{fwj}(t) \tag{67}$$

That is, $\hat{P}_{Yfa}(t)$ is the approximation of the filtered P(Y) code which is based on assuming that the Receiver A soft W-chip estimates $\hat{w}_{saj}$ are the true W-chip values.

The detection statistic in Eq. (66) has several useful features. It is the optimal statistic in the limit of exact values in $\hat{w}_{saj}$, as has been stated. Thus, the detection statistic in Eq. (66) takes the standard matched filter form: mix the data with the known signal time history. Fortunately, this is also an approximately optimal form even when the expected signal time history is not known exactly due to residual uncertainty in the $\hat{w}_{saj}$ estimates. A second convenient feature of Eq. (66) is that it avoids the need to compute optimal $\hat{w}_{sbj}$ estimates. That is, it avoids the need to solve the optimal estimation problem in Eqs. (32a)-(32c) for Receiver B. This can represent a considerable computational savings.

There are two negative aspects of using the detection statistic in Eq. (66). They both concern loss of the symmetry between the receivers that had been present in the statistic formula in Eq. (53). This loss of symmetry implies that different calculations must be performed in the reference and defended receivers. Furthermore, if the detection is not done in the defended receiver, then it must transmit to the detector the raw quadrature baseband

samples y_{qbi} instead of the soft W-chip estimates $\hat{w}_{sbj}$. This change would likely require an increased communication bandwidth.

Probabilistic analysis of the detection statistic in Eq. (66) yields its means and variances under the un-spoofed H_0 and spoofed H_1 hypotheses:

$$\bar{\gamma}_{salt|H0} = E\{\gamma_{salt} \mid H_0\} = \sum_{j=1}^{K} \hat{w}_{saj}^2 \xi_{bj}^2 \tag{68a}$$

$$\sigma_{\gamma salt|H0}^2 = E\{\gamma_{salt}^2 \mid H_0\} - \bar{\gamma}_{salt|H0}^2 = \sum_{j=1}^{K} \hat{w}_{saj}^2 \xi_{bj}^2 [1 - \hat{w}_{saj}^2 \xi_{bj}^2 + \xi_{bj}^2] \tag{68b}$$

$$\bar{\gamma}_{salt|H1} = E\{\gamma_{salt} \mid H_1\} = 0 \tag{68c}$$

$$\sigma_{\gamma salt|H1}^2 = E\{\gamma_{salt}^2 \mid H_1\} = \sum_{j=1}^{K} \hat{w}_{saj}^2 \xi_{bj}^2 \tag{68d}$$

It is straightforward to show that these means and variances of the detection statistic γ_{salt} approach those in Eqs. (57a)-(57d), those of γ_s, in the limit of small $(C/N_0)_{pyb}$. This is true because small $(C/N_0)_{pyb}$ implies small ξ_{bj}^2, as per Eq. (62b). The special functions in Eqs. (56a) and (56b) take on the approximate 2nd-order Taylor series forms $q(\xi) \cong \xi^2$ and $r(\xi) \cong \xi^2$ for $\xi^2 << 1$. Substitution of these approximations into Eqs. (57a)-(57d) yields equivalent expressions to Eqs. (68a)-(68d) out to 1st order in ξ_{bj}^2, which proves the equivalence in the low-power limit. Note, however, that the expressions in Eqs. (57a)-(57d) differ from those in Eqs. (68a)-(68d) starting in their 2nd order terms in ξ_{bj}^2.

Off-line predictive results similar to Eqs. (63a)-(63d) could be developed for the alternate detection statistic γ_{salt}. They have been omitted for the sake of brevity.

V. EXPERIMENTAL SPOOFING DETECTION RESULTS

A. Cases Considered

This paper's spoofing detection algorithms have been implemented and tested on actual data. The algorithms run in MATLAB software receiver code that operates on recorded RF data in an off-line mode. The RF data have been collected simultaneously from reference Receiver A operating in Ithaca, NY and from defended Receiver B operating in Austin, TX. Both receivers were connected to roof-mounted patch antennas.

The RF front-ends of the 2 receivers have 3 dB bandwidths of 2.4 MHz (Ithaca) and 2.6 MHz (Austin). The former front-end attenuates the P(Y) signal power by 5.6 dB, and the latter by 5.4 dB.

In a first test, the Austin receiver was not subjected to a spoofing attack. The first test was conducted in Sept. 2010. In a second type of test, the Austin receiver was attacked using the spoofer that is described in Refs. 5 and 6. Various versions of the second test were conducted in Sept. 2010 and in July 2011. Results for the second type of test will be reported only for the July 2011 data because that data set employed the most sophisticated version of the spoofer.

The spoofing attack was carried out by combining the signal from the spoofer with the signal from the Austin, TX roof-mounted patch antenna. This combining operation was carried out electronically before input to the RF front-end of the defended receiver. This approach avoided violation of FCC regulations because the spoofing signal was never broadcast. The spoofer also had access to the signal from a roof-mounted antenna, as required by the spoofer design of Refs. 5 and 6. It used this data to lay the spoofed signal exactly on top of the true signal during the initial attack. This attack profile allowed the victim receiver to continue tracking C/A code without interruption and seemingly without problems during the attack.

A special spoofing protocol has been used for the July 2011 spoofed case. The initial 60 seconds of data have no spoofing. The spoofer turns on at about 60 seconds, but it keeps its spoofed C/A code exactly on top of the true C/A code for about the first 60 seconds of spoofing. During this initial period, there is zero carrier Doppler shift of the spoofed signal relative to the true signal. During this phase, the spoofing detection algorithm will still see the true P(Y) code on the quadrature channel unless the spoofed C/A code has exactly a 90 deg phase offset from the true C/A code. In this situation, however, the true P(Y) code will not have the correct amplitude relationship to the spoofed C/A code because the latter will have a higher amplitude than the true C/A code in order to take control of the receiver's tracking loops. At about 120 second into the spoofing run, i.e., about 60 seconds after the onset of the attack, the spoofer starts to move the spoofed C/A code phase away from the true code. This process is necessary if the spoofer wants to deceive the receiver about its position or time. The receiver's C/A-code tracking loops are dragged away from the true C/A code by the spoofed signal during this latter phase of the attack. This causes the P(Y) code to disappear from the quadrature channel of the victim receiver, and the spoofing detection test statistic should drop to a mean of zero at this point of the attack.

Only a subset of the visible GPS satellites had their C/A PRN codes spoofed in the attack. There were 9 signals present in the data, but only 6 of them were spoofed.

B. Performance of Codeless Spoofing Detection

Results for the codeless spoofing detection test are shown in Figs. 4 and 5. Figure 4 corresponds to an un-spoofed case. It plots the detection statistic γ (solid blue curve),

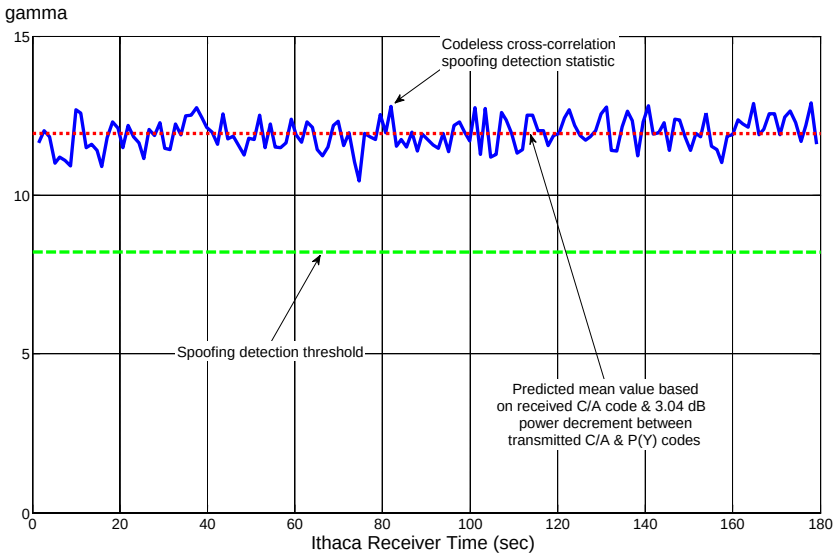

Fig. 4. Codeless spoofing detection statistic time history for PRN 17, un-spoofed case (T_{corr} = 1.2 sec, α_{FA} = 0.0001).

the statistic's predicted mean value $\bar{\gamma}_{H0}$ (dotted red curve), and the 0.01% false-alarm spoofing detection threshold γ_{th} (dashed green curve). The γ statistic has been computed using the cross-correlation interval T_{corr} = 1.2 sec. These curves apply to PRN 17, a typical tracked signal. The mean and threshold values have been computed based on the assumption that the P(Y) code is transmitted with a power level that is $10log_{10}(L_p)$ = -3.04 dB down from that of the C/A code. This is the value that causes $\bar{\gamma}_{H0}$ to equal the mean of γ -- note the correspondence between the level of the dotted red curve and the mean value of the solid blue curve. This case demonstrates the efficacy of the spoofing detection test: It clearly recognizes that this signal is not being spoofed. It also demonstrates the reasonableness of the statistical signal modeling that went into deriving the mean value $\bar{\gamma}_{H0}$ and the detection threshold γ_{th}.

Figure 5 demonstrates the codeless detection method's performance during a spoofing attack. Again, this figure plots time histories of the detection statistic γ, its predicted mean value $\bar{\gamma}_{H0}$, and the corresponding 0.01% false-alarm spoofing detection threshold γ_{th}, all calculated using 1.2 sec cross-correlation intervals. These quantities are plotted for two signals: PRN 13, which undergoes a spoofing attack starting at t = 60 sec, and PRN 23, which remains un-spoofed for the duration of the test. Unlike Fig. 4, the $\bar{\gamma}_{H0}(t)$ and $\gamma_{th}(t)$ time histories fluctuate because their levels are computed based on time-varying averages of the two receivers' C/A-code carrier-to-noise ratios. Each average is taken over the corresponding spoofing detection cross-correlation interval. The C/A to P(Y) transmitted power loss factors that have been used to produce these $\bar{\gamma}_{H0}(t)$ and $\gamma_{th}(t)$ plots are $10log_{10}(L_p)$ = -3.93 dB for PRN 13 and $10log_{10}(L_p)$ = -3.80 dB for PRN 23. These values have been chosen to make the $\bar{\gamma}_{H0}(t)$ curves lie close to the $\gamma(t)$ curves during the un-spoofed first 60 seconds of this case.

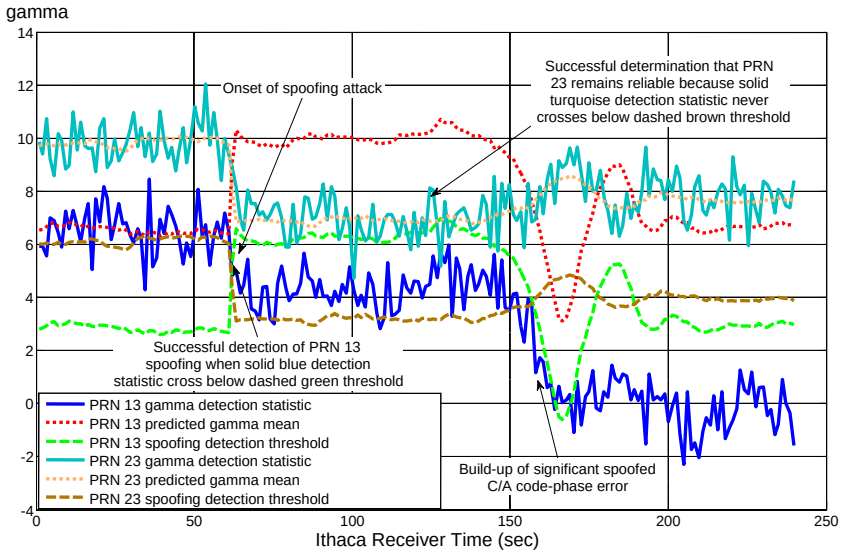

Fig. 5. Codeless spoofing detection statistic time histories for spoofed PRN 13 and un-spoofed PRN 23 (T_{corr} = 1.2 sec, α_{FA} = 0.0001).

Figure 5 shows clear responses at the time of the initial attack and further response changes as the attack progresses to carry the tracking loops away from the true signal. The spoofing detector correctly identifies the fact that PRN 13 is spoofed starting at t = 60 sec and that PRN 23 is never spoofed. PRN 13's solid blue spoofing detection statistic drops below its dashed green detection threshold and remains below that value except for a short interval from t = 164 to 169 sec. During this latter interval, the detection fails briefly because the detection power falls to low levels. This happens because the spoofed and true C/A codes briefly interfere with each other to produce a short, sharp power fade on that signal.

PRN 23, on the other hand, never generates a spoofing (false) alarm. Its solid turquoise detection statistic never drops below its corresponding dashed brown detection threshold.

It is interesting to note the behavior of spoofed PRN 13's detection statistic during the two phases of the attack. During the interval from t = 60 sec to t = 150 sec, the spoofed signal exactly overlays the true signal. The detection statistic drops a small amount, but not to a mean value of 0. The residual non-zero mean value is the result of the P(Y) code still being present, though not with the same amplitude as before the attack. One of the reasons for the amplitude reduction is the larger overall power entering the RF front-end at the onset of the attack. The spoofing signals must have higher power than the true signals in order to capture the receiver's tracking loops. This extra power affects the RF front-end's Automatic Gain Control (AGC), causing it to lower the gain. This lowered gain translates into a lowered received power of the true P(Y) code in Receiver B. This lower power reduces the value of the detection statistic. A second possible reason for the drop in the statistic during the middle interval is that the spoofed C/A code phase probably does not match the true C/A code phase. Therefore, the quadrature baseband mixing will not exactly capture the P(Y) code, thus reducing the detection statistic's amplitude. In an extreme situation, the detection statistic could take on a negative mean value during this phase. Starting at about t = 150 sec, the spoofer drags the receiver away from the true C/A code. It also drags the quadrature channel away from the true P(Y) code, and the spoofing detection statistic drops to a mean value of zero, as expected.

One might think that the spoofing detection test would not detect the attack until the last phase, when the spoofer drags the receiver away from the true C/A code phase. In fact, the detection is successful at the very outset of the attack. This happens because the spoofing detection threshold rises suddenly: Note the sudden jump of the green dashed curve at t = 60 sec. This rise is caused by the increased C/A code power of the combined spoofed plus true signal during this phase of the attack. This rise is sufficient to cause the spoofing alarm to be issued. Note, however, that there could be situations for well executed attacks where the spoofing attack would not be detected until the last phase, the phase of C/A code drag-off. Such a situation is acceptable because a spoofing attack with the spoofed C/A code exactly aligned to the true code represents a benign event.

The detection statistic for un-spoofed PRN 23, the solid turquoise curve, undergoes a sudden drop at the onset of the attack at t = 60 sec. This occurs because the receiver lowers its AGC gain in response to the extra power of the spoofing signals. The effect on an un-spoofed signal is to lower its C/A and P(Y) power, and this lowering of power is what causes the spoofing detection statistic for PRN 23 to decrease suddenly. One might think that this sudden decrease would give rise to a false spoofing alarm. This does not happen because the spoofing detection threshold for PRN 23, the dashed brown curve in Fig. 5, drops at the same time. It drops because it is keyed to the PRN 23 C/A signal power, which also drops in response to the AGC adjustment. Thus, the connection between the C/A-code signal power and the design of the spoofing detection threshold is important to the proper operation of this test.

The results in Fig. 5 might tempt one to suggest a simpler method of detecting the spoofing attack: Look for sudden changes of the AGC and of the C/A code power. If the AGC gain suddenly drops while the C/A power suddenly rises for some of the channels, then declare a spoofing attack. Additionally, small transient carrier phase glitches in the PLL tracking performance are evident on some of the spoofed channels at the onset of the spoofing attack. One might be tempted to look for such glitches and use them to detect a spoofing attack. Unfortunately, these detection methods can be defeated by slowly ramping up the power of the spoofed signals at the beginning of the attack. A slow attack was not used here only because the authors wanted to minimize the amount of data that needed to be tracked using offline MATLAB software receiver code. Such code runs very slowly, and its use on long data sets can be time-consuming.

In addition, the proposed simple detection scheme would work only if applied at or very near the initial moment of the spoofing attack. If the attack were not detected at its onset, then the simple detection methods would fail. This paper's cross-correlation-based detection methods function well during all phases of an attack.

The results in Fig. 5 and related results for other data sets represent the first successful spoofing detections using a single-antenna system at the defended receiver when attacked by the sophisticated spoofer of Refs. 5 and 6. The only other successful detection used a multi-antenna system [13]. This also represents the first successful detection of an actual spoofing attack using the cross-correlation method of Refs. 11 and 12. This demonstration is important because it proves that the vestigial P(Y) code in a narrow-band receiver can form the basis of a powerful spoofing detection test.

The detection powers in all 3 cases associated with Figs. 4 and 5 remain above 0.995, except for PRN 13 during the short interval from t = 160 to 174 sec. As already mentioned, this short anomaly is caused by a drop in the C/A code amplitude due to transient interference between the true and spoofed signals. During steady-state spoofing, no such interference would occur due to the temporal separation between the two codes. The nominally high probabilities of detection indicate that the

T_{corr} = 1.2 sec cross-correlation intervals are more than sufficient for a powerful test. They probably could be shortened significantly.

Two additional spoofed signals have been processed for the case associated with Fig. 5, those of PRN 03 and PRN 16. They both required P(Y) transmitted power decrements of $10log_{10}(L_p)$ = -3.37 dB in order to achieve good agreement between $\gamma(t)$ and $\bar{\gamma}_{H0}(t)$ during the initial un-spoofed phase. Spoofing detection worked well for these two signals, similar to the results for PRN 13 in Fig. 5.

C. Performance of Semi-Codeless Spoofing Detection

Figures 6 and 7 present results for the semi-codeless spoofing detection method. They plot $\gamma_s(t)$ spoofing detection time histories along with predicted mean-values $\bar{\gamma}_{s|H0}(t)$ and detection thresholds $\gamma_{sth}(t)$. Both figures use spoofing detection intervals of T_{corr} = 0.2 sec and false-alarm probabilities of 0.01%. The case shown in Fig. 6 is for PRN 17 not being spoofed. It re-processes the same RF data as have been used to generate the codeless detection results in Fig. 4. Figure 7 corresponds to a case in which PRN 13 is spoofed starting at t = 60 sec. It re-processes the same RF data that apply to the PRN 13 curves in Fig. 5. The salient feature of Figs. 6 and 7 is the ability of the semi-codeless method to distinguish between un-spoofed and spoofed signals as reliably as does the codeless method, but using cross-correlation intervals that are only 1/6th as long.

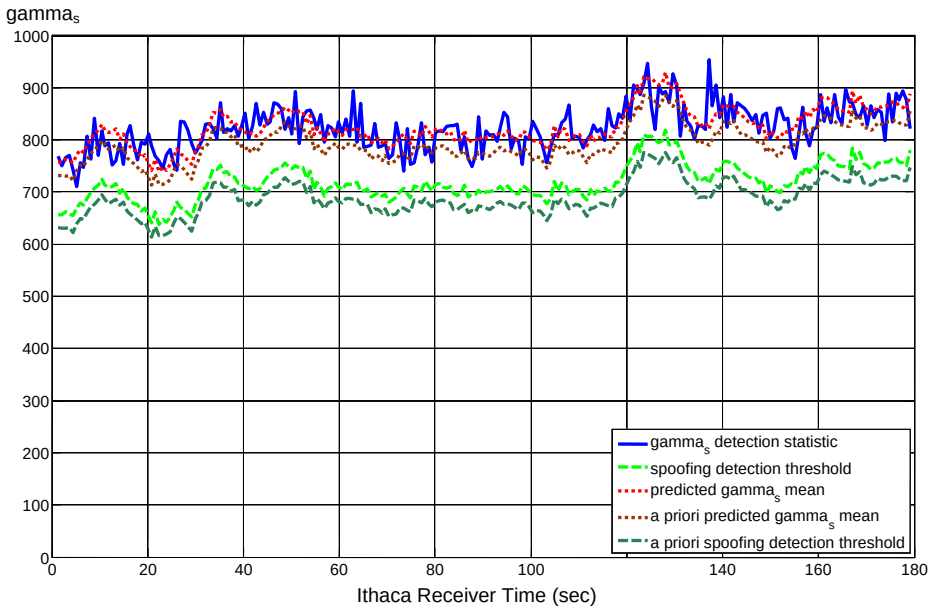

Fig. 6. Semi-codeless spoofing detection statistic and related time histories for PRN 17, un-spoofed case (T_{corr} = 0.2 sec, α_{FA} = 0.0001).

An important feature of these curves is the accuracy with which the predicted mean values in $\bar{\gamma}_{s|H0}(t)$ track the actual spoofing detection statistic $\gamma_s(t)$ whenever the signal is not being spoofed. These mean values rely on the power decrement factors L_p for the transmitted P(Y) code relative to the transmitted C/A code. The same power loss factors have been used in this analysis as have been used in the codeless analysis, $10log_{10}(L_p)$ = -3.04 dB for PRN 17 and $10log_{10}(L_p)$ = -3.93 dB for PRN 13. These factors were "tuned" in the codeless case in order to get $\bar{\gamma}_{H0}(t)$ to track $\gamma(t)$. No additional tuning has been used for the semi-codeless case. The reasonable correspondence of $\bar{\gamma}_{s|H0}(t)$ to $\gamma_s(t)$ in the semi-codeless case provides an independent confirmation that the original tunings are reflective of the actual relationship between the transmitted C/A and P(Y) power levels.

Note in Fig. 7 that the spoofing detection statistic hardly changes during the first portion of the attack, from t = 60 to 140 sec. The mean of the solid blue curve stays roughly constant, but there is a moderate increase in its variance. Apparently, this initial part of the attack leaves Receiver B with an ability to obtain reasonably accurate "soft" W-chip estimates; otherwise, the detection statistic would have decreased significantly. This is reasonable given that the true P(Y) code remains in the correct location relative to the spoofed C/A code during this interval. Despite this rough constancy of the spoofing detection statistic, the test still detects the attack by virtue of the sudden rise in its detection threshold at the onset -- see the dashed green curve. As in Fig. 5, this rise is caused by the increased power of the spoofed C/A code. The detection algorithm thinks that the true C/A code's power has risen and, therefore, that the true P(Y) code's power must have risen with it. It concludes that the accuracies of its Receiver-B "soft" W-chip estimates should have increased and, therefore, that the spoofing detection statistic's mean value should have increased. These increases cause the sudden rise in the dashed green detection threshold. The expected increase in $\gamma_s(t)$ fails to materialize, $\gamma_{sth}(t)$ crosses above $\gamma_s(t)$, and a spoofing alert is issued.

Figures 6 and 7 each have two new curves that do not appear in Figs. 4 and 5. These are the dotted brown a priori

predictions of $\bar{\gamma}_{s|H0}(t)$ and the dashed olive-green a priori thresholds $\gamma_{sth}(t)$. These values have been computed using the offline analysis formulas from Subsection IV.E. These formulas take as inputs the P(Y)-code carrier-to-noise ratios as deduced by scaling down the C/A-code carrier-to-noise ratios that are deduced from the C/A-code prompt accumulations. These curves have been plotted as a means of checking whether the analyses of Subsection IV.E have any basis in reality. There is a reasonable level of correspondence between these a priori $\bar{\gamma}_{s|H0}(t)$ and $\gamma_{sth}(t)$ curves and those that have been calculated based on the Receiver-A "soft" W-chip estimates $\hat{w}_{saj}$. The agreement is especially good in Fig. 7. Therefore, the a priori analyses of Subsection IV.E are reasonable.

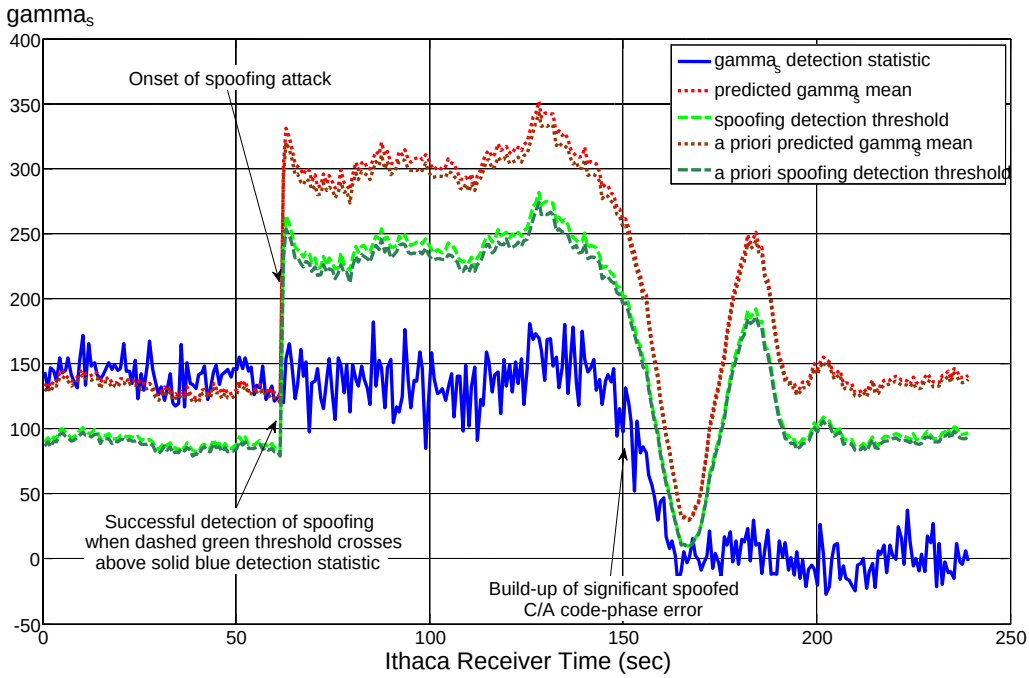

Fig. 7. PRN 13 semi-codeless spoofing detection statistic and related time histories during a spoofing attack (T_{corr} = 0.2 sec, α_{FA} = 0.0001).

The probabilities of detection associated with the semi-codeless cases in Figs. 6 and 7 are very high. The lowest value is 0.964, and that value occurs only briefly for PRN 13 during the interval from t = 164 to 170 sec. Recall from the discussion of Fig. 5 that the lower detection power during this interval occurs because of interference-induced loss of C/A signal power. This lowered detection power corresponds to a brief 0.35 sec interval of a missed spoofing detection in Fig. 7 at t = 168 sec. At all other times, the probability of detection is above 0.9999. Therefore, it would be possible to reduce the correlation interval significantly below T_{corr} = 0.2 sec and still retain adequate detection power.

Similar results have been obtained for the spoofed signals from PRN 03 and 16 for the same attack that yielded Fig. 7. Thus, the results for PRN 13 represent typical performance of the semi-codeless algorithm.

D. Investigation of the Effects of Relative Time Offsets between the C/A and P(Y) Codes

A study has been made of the effect on codeless spoofing detection of varying the differential relative time parameter δt_{ab}. Recall from Subsection III.A that this is a differential between Receivers A and B of the timing of the received, filtered P(Y) code relative to the tracked C/A code. Variations of this offset, as propagated through Eqs. (17) and (18), have been assessed in order to determine how they affect the mean cross-correlation amplitude. The correct value of δt_{ab} should give the peak amplitude.

All studies to date show that the peak cross-correlation amplitude occurs at δt_{ab} = 0 for the receivers and tracking loops that have been considered. The precision of this finding is significantly better than 0.025 C/A code chips (24 nsec). Given that the two receivers' RF front-ends and tracking software were identical to within manufacturing tolerance, this result is not surprising.

If there were significant differences between the receiver RF front-ends, the DLL discriminators, or the DLL tracking loops, then this result might change. In any application of codeless spoofing detection to a new receiver design, this issue should be investigated. If necessary, the optimal value of δt_{ab} should be determined, recorded, and applied as a calibration parameter during regular codeless cross-correlation calculations.

A similar timing issue has been investigated for the semi-codeless method. In this situation, it is a question of the timing of the RF-filtered C/A code, as measured by the DLL and its discriminator, relative to the filtered P(Y) code as predicted by the system-ID RF filter impulse response function. The optimal relative timing should give the "best" W-chip estimates, and by extension, the best detection power. In this context, the "best" estimates

are those that yield the lowest optimized value of the estimation cost function in Eq. (32b).

There is reason to believe that the proper relative timing would be known a priori in the present set of tests. This is true because these tests used the same receiver, the same RF front-end, the same C/A-code DLL, and the same discriminator to support the W-chips estimation process as had been used in the system-ID calculations for the RF filter [21]. Nevertheless, a study has been made of different timing offsets and their effects on the optimal values of the cost function in Eq. (32b). The study has produced puzzling results to date. Therefore, more work on this issue is deemed necessary.

If there is a timing bias between the C/A code DLL and the W-chips' filtered P(Y)-code functions, then this bias should be measurable as part of a calibration procedure. Therefore, a calibration test should carried out as part of any application of semi-codeless spoofing detection to a new receiver design. The relative timing result of the test should be retained and applied as a calibration factor during regular operation. Unfortunately, the inconclusive state of the investigation of this issue leaves open the question about how best to perform the requisite calibration.

VI. VULNERABILITY TO ALTERNATE METHODS OF SPOOFING ATTACK

This paper's two spoofing detection tests have been developed by using the methods of statistical hypothesis testing. They develop a test statistic that distinguishes between two precisely defined hypotheses. The null hypothesis is that the P(Y) code signal is present in quadrature with the C/A code in the defended receiver and that it has a well defined amplitude ratio relative to the C/A code. This is the un-spoofed hypothesis. The spoofed hypothesis presumes that there is no signal on the quadrature channel.

As has already been mentioned in Subsection IV.C, alternate forms of spoofing may be applied. If the spoofer suspects that this paper's cross-correlation algorithms are being used, then it may elect to do something different than leaving no signal on the quadrature channel. As already mentioned, the spoofer may elect to put pseudo P(Y) code on the quadrature channel. This possibility has been considered, and this consideration has steered the semi-codeless detection test away from choosing a statistic that would look partly for W-chip power in the defended receiver.

Another possibility for attack is a Security Code Estimation and Replay (SCER) attack [15]. This type of attack actively seeks to estimate the W chips on-line, and it uses its imprecise W-chip estimates in an attempt to spoof the true P(Y) code. This type of attack will dilute the spoofing detection power of a cross-correlation method in direct proportion to the percentage of its correct W-chip estimates. Of course, a large dilution can only be achieved by a high-gain antenna system. If the number of correctly estimated W chips in the spoofer were not too large and if the cross-correlation spoofing detection algorithm had enough power, then this type of attack would be detected. An effective SCER spoofer would have to estimate most of the W chips correctly, which would be expensive in terms of the needed antenna gain.

Alternatively, an SCER attack might try to compensate for mis-estimation of a significant fraction of the W chips by turning up the power of the spoofed P(Y) code. This strategy might thwart the codeless cross-correlation detection test of Section III or the alternate semi-codeless test of Subsection IV.F. The semi-codeless test developed in Subsections IV.A to IV.D, however, could detect this attack mode by looking at the distribution of its $\hat{w}_{sbj}$ estimates. Too many of them would be too near +1 or -1 for the given C/A-code carrier-to-noise ratio.

Furthermore, an SCER spoofer might need to induce a delay of the spoofed C/A code relative to the true C/A code in order to gain time to form its W-chip estimates. The necessary delaying action might be noticeable in the defended receiver at the onset of the attack.

There are other possible attack types. The spoofer might try to locate a second spoofer near the secure receiver. If both spoofers used a common false P(Y) code, then they would defeat this technique. A defense against such an attack would be to distribute an array of secure receivers over a large area and to connect them in a network that aggregated their $\hat{w}_{saj}$ chip estimates. If there were enough secure receivers and if their locations were kept secret, then it would be unlikely that enough of them could be discovered and spoofed in a way that would defeat the detection system. Reference stations could employ phased-array antennas with independently steerable beams in order to ensure their security. They could use beam steering to attenuate the signal of any spoofer that was not directly on their line-of-sight vector to a given GPS satellite.

A meaconing attack could also defeat this method. This technique receives and replays the entire GNSS spectrum with some unavoidable delay [15]. This type of attack can event defeat a secure military receiver if the replayed bandwidth is wide enough to contain the P(Y) or M codes. A sophisticated meaconing attack might use differential delays for different signals, which it could implement by using a phased array with independently steerable beams for signal reception prior to replay. This type of attack, however, would be very expensive. A

simple meaconing attack with only one delay for all signals would cause the spoofed receiver to determine a location equal to the spoofer's location, which could prove dangerous for the spoofer. Also, a victim receiver with a very stable oscillator might detect the attack because of the necessary delay.

Other types of spoofing attacks might be mounted against this paper's cross-correlation detection methods. Perhaps a problematic attack would be to raise the noise floor on the quadrature channel instead of putting estimated or false P(Y) code there. The analysis of all such attack scenarios and the performance of this paper's detectors under threat of such attacks is beyond this paper's scope. Several preliminary analyses of this subject suggest that this paper's spoofing detection techniques, especially its semi-codeless technique, would perform well under many attack scenarios if the power of detection were sufficiently close to 1 for the simple attack scenario discussed in this paper.

VII. SUMMARY AND CONCLUSIONS

Two spoofing detection methods have been developed for open-source/civilian GNSS signals. They rely on the presence of an encrypted/military signal on the same transmitted frequency and on knowledge of the timing and carrier-phase relationship of the encrypted signal to the open-source signal. The open-source signal is tracked in a secure reference receiver and in a defended receiver that might be the victim of a spoofing attack. The open-source tracking data are used to isolate the part of the received signal that is encrypted. The encrypted parts of the signals from the two receivers are cross-correlated after being brought together via a communications link. This use of cross-correlation obviates the need for a priori knowledge of the PRN code of the encrypted signal. If a high cross-correlation statistic is obtained, then no spoofing has been detected because this large value indicates the presence of the encrypted signal in both receivers. If the cross-correlation statistic is too low, then a spoofing alert is issued. The low cross-correlation is likely due to the absence of the encrypted part of the received signal in the defended receiver. The only explanation for this absence is that the tracked open-source signal is a false spoofing signal.

Codeless and semi-codeless cross-correlation spoofing detection tests have been developed, analyzed, and tested. The analyses provide an ability to choose spoofing detection thresholds based on hypothesis testing theory and an ability to predict detection power. The thresholds are dependent on the received power of the open-source signal and on the known power of the encrypted signal relative to the open-source signal. Semi-codeless techniques offer significant processing gain in comparison to codeless techniques. When applied using the encrypted GPS P(Y) code, the semi-codeless spoofing detection technique is similar to the semi-codeless methods that are used in civilian dual-frequency GPS receivers.

The new techniques have been applied to detect actual GPS spoofing attacks using recorded RF data and off-line signal processing. The codeless technique has successfully detected spoofing of the GPS L1 C/A code by cross-correlating the military P(Y) code over accumulation intervals of 1.2 sec. The semi-codeless technique has succeeded when using cross-correlation intervals of only 0.2 sec. It is likely that reductions of these intervals could be tolerated while maintaining a high detection power.

A surprising aspect of these results is that they have been obtained using low-gain patch antennas and narrow-band receivers. Each receiver's RF front-end had a 2.5 MHz wide filter and a 5.714 MHz sampling rate. These front-ends attenuate the P(Y) code by 5.5 dB and drastically distort its chips. Nevertheless, sufficient P(Y) power remains for successful spoofing detection based on short cross-correlation intervals.

REFERENCES

1. Anon., "Vulnerability Assessment of the Transportation Infrastructure Relying on the Global Positioning System," Tech. Report, John A. Volpe National Transportation Systems Center, 2001.
2. Warner, J.S., and Johnston, R.G., “A Simple Demonstration that the Global Positioning System (GPS) is Vulnerable to Spoofing,” *Journal of Security Administration*, 2003.
3. Scott, L., "Anti-Spoofing and Authenticated Signal Architectures for Civil Navigation Systems," *Proc. ION GPS/GNSS 2003*, Sept. 9-11, 2003, Portland, OR, pp. 1543-1552.
4. Scott, L., "Location Assurance," *GPS World*, Vol. 18, No. 7, July 2007, pp. 14-18.
5. Humphreys, T.E., Ledvina, B.M., Psiaki, M.L., O'Hanlon, B.W., and Kintner, P.M., Jr., "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer," *Proc. ION GNSS 2008*, Sept. 16-19, 2008, Savannah, GA, pp. 2314-2325.
6. Humphreys, T.E., Kintner, P.M., Jr., Psiaki, M.L., Ledvina, B.M., and O'Hanlon, B.W., "Assessing the Spoofing Threat," *GPS World*, Vol. 20, No. 1, Jan. 2009, pp. 28-38.
7. Pozzobon, O., "Keeping the Spoofs Out, Signal Authentication Services for Future GNSS," *Inside GNSS*, Vol. 6, No. 3, May/June 2011, pp. 48-55.
8. Dovis, F., Chen, X., Cavaleri, A., and Pini, M., "Detection of Spoofing Threats by Means of Signal Parameters Estimation," *Proc. ION GNSS 2011*, Sept. 20-23, 2011, Portland, OR.
9. Cavaleri, A., Pini, M., Lo Presti, L., Fantino, M., and Ugazio, S., "Signal Quality Monitoring Applied to

Spoofing Detection," *Proc. ION GNSS 2011*, Sept. 20-23, 2011, Portland, OR.
10. Wesson, K., Shepard, D., Bhatti, J., and Humphreys, T., "An Evaluation of the Vestigial Signal Defense for Civil GPS Anti-Spoofing," *Proc. ION GNSS 2011*, Sept. 20-23, 2011, Portland, OR.
11. Lo, S., De Lorenzo, D., Enge, P., Akos, D., and Bradley, P., "Signal Authentication, A Secure Civil GNSS for Today," *Inside GNSS*, Vol. 4, No. 5, Sept./Oct. 2009, pp. 30-39.
12. O'Hanlon, B.W., Psiaki, M.L., Humphreys, T.E., and Bhatti, J.A., "Real-Time Spoofing Detection in a Narrow-Band Civil GPS Receiver," *Proc. ION GNSS 2010*, Sept. 21-24, 2010, Portland, OR, pp. 2211-2220.
13. Montgomery, P.Y., Humphreys, T.E., and Ledvina, B.M., "A Multi-Antenna Defense: Receiver-Autonomous GPS Spoofing Detection," *Inside GNSS*, Vol. 4, No. 2, March/April 2009, pp. 40–46.
14. Wesson, K., Rothlisberger, M., and Humphreys, T., "Practical Cryptographic Civil GPS Signal Authentication," submitted to *Navigation*, in review, 2011 (Available at http://radionavlab.ae.utexas.edu/images/stories/files/papers/nma.pdf).
15. Humphreys, T.E, "Detection Strategy for Cryptographic GNSS Anti-Spoofing," submitted to *IEEE Transactions on Aerospace and Electronic Systems*, in review, 2011 (Available at http://radionavlab.ae.utexas.edu/images/stories/files/papers/ds.pdf).
16. Lorenz, R.G., Helkey, R.J., and Abadi, K.K., "Global Positioning System Receiver Digital Processing Technique," U.S. Patent No. 5,134,407, July 1992.
17. Woo, K.T., "Optimum Semicodeless Carrier-Phase Tracking of L2," *Navigation*, Vol. 47, No. 2, 2000, pp. 82-99.
18. Woo, R.K.T., Quan, J.O., and Cheng, U., "System and Method for Demodulating Global Positioning System Signals," U.S. Patent No. 6,125,135, Sept. 2000.
19. Psiaki, M.L., Powell, S.P., Jung, H., and Kintner, P.M., Jr., "Design and Practical Implementation of Multi-Frequency RF Front Ends Using Direct RF Sampling", *Proc. ION GPS/GNSS 2003*, Sept. 9-12, 2003, Portland, OR, pp. 90-102.
20. Anon., "Global Positioning System Wing (GPSW) Systems Engineering and Integration Interface Specification," IS-GPS-200E, Science Applications International Corporation, El Segundo, CA, June 2010.
21. Psiaki, M.L., and O'Hanlon, B.W., "System Identification of a GNSS Receiver's RF Filter Impulse Response Function," *Proc. ION GNSS 2011*, Sept. 20-23, 2011, Portland, OR.
22. Bar-Shalom, Y., Li, X.-R., and Kirubarajan, T., *Estimation with Applications to Tracking and Navigation*, J. Wiley & Sons, (New York, 2001), pp. 48-49, 72-74.