AFRL-RV-PS-
TR-2021-0090

INTEGRATION OF CHIMERA GPS ANTI-SPOOFING METHODS WITH INERTIAL NAVIGATION SYSTEMS

Mark L. Psiaki

Kevin T. Crofton Department of Aerospace and Ocean Engineering
Virginia Tech
300 Turner Street NW, Suite 4200
Blacksburg, VA 24061-0203

31 May 2021

Final Report

AIR FORCE RESEARCH LABORATORY
Space Vehicles Directorate
3550 Aberdeen Ave SE
AIR FORCE MATERIEL COMMAND
KIRTLAND AIR FORCE BASE, NM 87117-5776

DTIC COPY

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research which is exempt from public affairs security and policy review in accordance with AFI 61-201, paragraph 2.3.5.1. This report is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RV-PS-TR-2021-0090 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//SIGNED//

Dr. Joanna C. Hinks
Program Manager/AFRL/RVB

//SIGNED//

For: Erin N. Pettyjohn, Chief
AFRL Geospace Technologies Division

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
31-05-2021	Final Report	30 Aug 2019 – 31 May 2021

4. TITLE AND SUBTITLE
Integration of CHIMERA GPS Anti-Spoofing Methods with Inertial Navigation Systems

5a. CONTRACT NUMBER
FA9453-19-1-0083

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
C6601F

6. AUTHOR(S)
Mark L. Psiaki

5d. PROJECT NUMBER
4846

5e. TASK NUMBER
EF133151

5f. WORK UNIT NUMBER
V1FQ

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Kevin T. Crofton Department of Aerospace and Ocean Engineering
Virginia Tech
300 Turner Street NW, Suite 4200
Blacksburg, VA 24061-0203

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory
Space Vehicles Directorate
3550 Aberdeen Avenue SE
Kirtland AFB, NM 87117-5776

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RVBYS

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-RV-PS-TR-2021-0090

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Tightly-coupled INS/GPS navigation algorithms have been developed to use GPS data that have latency due to authentication using the CHIMERA system. Such an algorithm is needed to produce non-delayed navigation results that are guaranteed, via CHIMERA, not to have been spoofed. The developed algorithms are modified versions of the standard tightly-coupled INS/GPS method in which the INS data define a Kalman filter's position, velocity, and attitude dynamics via the technique known as model replacement. The Kalman filter's measurements are the GPS observables pseudorange and carrier Doppler shift. The CHIMERA authentication latency is handled by running multiple filters or partial filters. One filter uses only the authenticated data up to a past authentication epoch and INS data that run up to the present. One or more other filters process unauthenticated data in preparation for future use. Some filters consider the possibility that authentication times are staggered in hopes of improving performance. The methods have been evaluated using real data collected from a general-aviation aircraft and using simulations of what INS of varying quality would have output had they been taken on the same flight. Results for the fast CHIMERA channel, which entails latencies of only 2 seconds, have accuracies commensurate with zero-latency stand-alone GPS regardless of the IMU quality, i.e., 2 m RMS position errors. Results for the slow CHIMERA channel are problematic. The slow channel entails latencies of 180 sec. For non-staggered slowchannel CHIMERA with a navigation-grade INS, RMS position error is 9.5 m, and peak position error is more than 20 m. Lower-grade INS produce much larger RMS position errors, on the order of 100 to 2400 m, when paired with slow-channel CHIMERA. Even the navigation-grade INS performance with slow CHIMERA is significantly worse than zero-latency stand-alone GPS. If three staggered CHIMERA authentication groups are used, then these results improve to 6 m RMS and 15 m peak position error if using a navigation-grade IMU. The staggered CHIMERA results for tactical-grade and MEMSgrade IMUs improve by larger amounts, but they are still significantly worse than standalone, zero-latency GPS.

15. SUBJECT TERMS
authentication, anti-spoofing, GPS/INS, Kalman filtering

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Dr. Joanna C. Hinks
Unclassified	Unclassified	Unclassified	Unlimited	48	**19b. TELEPHONE NUMBER** *(include area code)*

ACKNOWLEDGMENTS

This material is based on research sponsored by Air Force Research Laboratory under agreement number FA9453-19-1-0083. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.

DISCLAIMER

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Laboratory or the U.S. Government.

Integration of CHIMERA GPS Anti-Spoofing Methods with Inertial Navigation Systems

Final Report for AFRL Grant No. FA9453-19-1-0083

by Mark L. Psiaki, Principal Investigator
Kevin T. Crofton Department of Aerospace and Ocean Engineering
Virginia Tech, Blacksburg, Virginia 24061-0203

Report for period starting on Aug. 30, 2019 and ending on May 31, 2021

Abstract

Tightly-coupled INS/GPS navigation algorithms have been developed to use GPS data that have latency due to authentication using the CHIMERA system. Such an algorithm is needed to produce non-delayed navigation results that are guaranteed, via CHIMERA, not to have been spoofed. The developed algorithms are modified versions of the standard tightly-coupled INS/GPS method in which the INS data define a Kalman filter's position, velocity, and attitude dynamics via the technique known as model replacement. The Kalman filter's measurements are the GPS observables pseudorange and carrier Doppler shift. The CHIMERA authentication latency is handled by running multiple filters or partial filters. One filter uses only the authenticated data up to a past authentication epoch and INS data that run up to the present. One or more other filters process unauthenticated data in preparation for future use. Some filters consider the possibility that authentication times are staggered in hopes of improving performance. The methods have been evaluated using real data collected from a general-aviation aircraft and using simulations of what INS of varying quality would have output had they been taken on the same flight. Results for the fast CHIMERA channel, which entails latencies of only 2 seconds, have accuracies commensurate with zero-latency stand-alone GPS regardless of the IMU quality, i.e., 2 m RMS position errors. Results for the slow CHIMERA channel are problematic. The slow channel entails latencies of 180 sec. For non-staggered slow-channel CHIMERA with a navigation-grade INS, RMS position error is 9.5 m, and peak position error is more than 20 m. Lower-grade INS produce much larger RMS position errors, on the order of 100 to 2400 m, when paired with slow-channel CHIMERA. Even the navigation-grade INS performance with slow CHIMERA is significantly worse than zero-latency stand-alone GPS. If three staggered CHIMERA authentication groups are used, then these results improve to 6 m RMS and 15 m peak position error if using a navigation-grade IMU. The staggered CHIMERA results for tactical-grade and MEMS-grade IMUs improve by larger amounts, but they are still significantly worse than stand-alone, zero-latency GPS.

Work Accomplished During Grant Period

Three tasks were carried out under this grant. The first was to acquire GPS/INS data for use in evaluating the algorithms and to develop a method of simulating different types of INS other than the one used for the real data. The simulation effort involved the determination of a "truth" trajectory for the true aircraft, including an attitude trajectory, the differentiation of that trajectory to produce "truth" rate-gyro and accelerometer outputs, and the addition of various levels of white-noise, bias, and bias drift to the "truth" outputs in order to simulate varying INS qualities.

The second task was to develop and evaluate filtering methods for overcoming CHIMERA latency when all of the signals authenticate at the same time. One basic algorithm has been developed. It consists of two filters. One the filters implements pure INS propagation in order to propagate from the most recent CHIMERA authentication epoch to the current time. This propagation takes INS bias estimates into account. The second filter performs a brute-force re-filtering operation that goes back and reprocesses all of the data from the next most recent authentication epoch to the newest one, this time using both the INS and the authenticated GPS data. A slightly different strategy does effectively the same thing, except that the two filters run in parallel. The full-data filter passes its authenticated estimates and covariance information to the INS-only filter in order to reset the latter filter at each authentication time – provided that the CHIMERA authentication is successful.

The third task was to develop and evaluate filtering methods that allow for staggering of CHIMERA authentication times. Like the non-staggered case, there is one part of each method that performs pure INS propagation in order to propagate from the most recent CHIMERA authentication epoch to the current time. Various other filters or partial filters run in parallel and process subsets of the GPS data in preparation for anticipated future authentication.

Personnel

Two researchers were involved in this project: Michael Esswein, a Ph.D. student, and Mark Psiaki, a professor. Both were part of Virginia Tech's Kevin T. Crofton Department of Aerospace and Ocean Engineering during the period when they worked on this project.

Significant Results

Almost all of the results of this grant are presented in detail in Ref. 1. For details see that paper. A preprint of it is attached to this report as Appendix A. The results reported in that paper are briefly summarized below. After that summary there is another section that gives a result which is not given in the paper.

Optimal Filter Designs for INS/CHIMERA Navigation with Authentication Latency

Three filters have been designed and evaluated, and they are reported in Ref. 1. The first is a sort of brute-force filter that is suitable for use with CHIMERA when there is no staggering of authentication epochs, i.e., when all GPS signals are authenticated at the same time. It is documented in Section II of Ref. 1. It runs two tightly-coupled INS/GPS filters. One filter uses only INS data to propagate its attitude/position/velocity estimate from the most recent CHIMERA GPS authentication epoch time. This propagation uses the standard technique known as INS model replacement. The other filter also uses INS model replacement for its dynamic propagation, but it also processes measurement updates from the GPS pseudoranges and carrier Doppler shifts. One version of the algorithm runs the INS-only filter in real-time from the most recent successful authentication. It is initialized with the best estimates of the state based on all of the GPS and INS data up through the most recent authentication time. The second filter reprocesses the INS and GPS data in an after-the-fact mode after the next authentication time. It re-starts at the previous authentication time and runs up through the new authentication time. Its outputs are then used to initialize the INS-only filter for the upcoming authentication interval. A slightly modified version runs the second filter in parallel with the first one in order for it to be immediately ready to initialize the INS-only filter at the new authentication time.

The other two filters that have been designed and evaluated have been tailored to the case of CHIMERA authentication staggering. They are described in Section V and Appendices A and B of Ref. 1. Authentication staggering occurs when different subsets of the available GPS signals have different CHIMERA authentication epoch times. Such an approach can ameliorate the accuracy degradation that is inherent in slow CHIMERA authentication even when using a navigation-grade INS to compensate for the latency that arises from the 180 sec interval between authentications. Both of these algorithms use tightly-coupled INS/GPS techniques.

One of the staggered-authentication algorithms is similar to the method discussed above when there is no staggering. Its main difference lies in its need to use $(N + 1)$ filters if there are N separate authentication groups. One filter uses only INS data from the most recent authentication epoch. It will have been initialized at that epoch time based on the best estimate that uses all of the INS data up to that time and all of the GPS data that have been authenticated from all groups at authentication times up to and including that time. A second filter will include additional GPS data: that from the subset of signals whose next authentication is the nearest in the future. A third filter will use all of the data that the second filter uses plus additional GPS data from the subset of signals whose next authentication time is the second nearest in the future, etc. The final filter will use all of the GPS data. At each authentication time, the various filters pass state and covariance information from one to the other. All except the first and last filter undergo a change of authentication signal sets whose data they will process. The $(N + 1)$ filters can be run in parallel. Alternatively, the same calculations can be implemented using brute-force re-filtering of past data that have been newly authenticated along with past data that had been previously authenticated.

The other staggered-authentication algorithm runs one complete Kalman filter and N partial filters. Its one complete Kalman filter is identical to the INS-only filter of the preceding algorithm. The N partial filters keep track of various subsets of additional information from the as-yet-unauthenticated GPS data. When a new authentication time epoch is reached, the INS-only filter has its information fused with the partial filter whose data have been authenticated. That partial filter as then re-set to contain zero information and the algorithm continues to the next authentication epoch.

The advantage of this second filter for the staggered case is that it reduces the computational expense relative to the other staggered-case filter. The computational cost reduction comes from the fact that all N of the partial filters process smaller numbers of measurements than do any of the N unauthenticated filters of the preceding method. An assessment of the amount of computational cost savings has yet to be done.

Both of the new filters for the staggered authentication case have been encoded and tested. They both work well when processing the true and partially simulated INS/GPS data that have been considered in this project.

Accuracy Results of Filters

Section III of Ref. 1 reports accuracy results for the case of no authentication staggering. It reports results for the fast CHIMERA channel with 2 seconds between authentications and for the slow channel with 180 seconds between authentications. Section IV of Ref. 1 reports accuracy results for the slow CHIMERA channel with authentication staggering. It assumes that the visible GPS signals are split into three roughly equal staggered authentication groups with even spacing between their

authentication times. Thus, roughly one third of the available GPS signals get authenticated every minute, but the data interval over which any one group gets authenticated stretches 180 sec into the past.

The fast-channel CHIMERA results are documented by the table in Fig. 3 of Ref. 1 and by the error time history in Fig. 4. RMS position errors are on the order of 2 m for all 4 grades of IMU that have been considered. The 4 considered IMUs are a navigation-grade device, two tactical-grade devices, and a MEMS-grade device.

The slow-channel CHIMERA results without authentication staggering are documented by the table in Fig. 5 of Ref. 1 and by the error time history in Fig. 6. An RMS error of 9.5 m and a peak error of about 20 m occurs when using slow CHIMERA authentication with a navigation-grade IMU. If a tactical-grade IMU is used, the RMS error grows to between 100 and 340 m. A MEMS-grade IMU yields an RMS position error of 2400 m. Peak errors for the tactical-grade and MEMS-grade IMUs are commensurately larger.

The slow-channel CHIMERA results with authentication staggering are documented by the table in Fig. 8 of Ref. 1 and by the error time history in Fig. 9. An RMS error of 6.0 m and a peak error of about 15 m occurs with a navigation-grade IMU. The RMS error increases to 43.8 m with a tactical-grade IMU and to 173.9 m with a MEMS-grade IMU. Thus, authentication staggering improves performance, especially for a lower-grade IMUs.

An Alternative Signal Processing Algorithm that uses Partial Re-Filtering

The foregoing algorithms are well designed for the cases that they consider. Several of them have unresolved challenges, however, if any of the CHIMERA authentications should fail. If authentication fails but is expected to succeed in the future due to spoofing counter measures, then some of the filters, especially for the second staggered-case algorithm, will have to flush some of their information and start over from scratch. This could be costly in terms of computational effort. Therefore, another algorithm has been developed. It is a partial re-filtering algorithm that is agnostic to the question of whether or not authentication times have been staggered and whether or not some GPS data never gets authenticated. Its strategy is to go back in time whenever a new span of data gets authenticated and to perform the needed partial filtering operations that incorporate the new information into the filter state estimate, its covariance information, and the corresponding information for the smoothed process noise. The new algorithm has been written-up in the form of the body of a technical note. It is attached to this report as Appendix B.

This new algorithm has a similar computational time advantage as the partial filters algorithm that has been discussed above for the staggered authentication case: It too processes only a reduced number of measurements for the staggered case and, therefore, requires fewer computations than a full brute-force re-filtering calculation would require.

This alternative algorithm has not yet been encoded and tested.

Summary

This grant has supported efforts to develop algorithms that compensate for the latency of the CHIMERA authentication of GPS signals. Various forms of tightly-coupled INS/GPS navigation Kalman filters have been developed and tested to achieve this goal. They all share the property of processing INS data and authenticated GPS data up to the time of the most recent CHIMERA authentication. Going forward from that time, they

process only the INS data in order to achieve a zero-latency solution with reduced uncertainty. A number of different algorithms have been developed and tested. One applies only to a CHIMERA authentication scheme in which all signals are authenticated together at the same time. Other filters also apply to the case of staggered authentication, where different subsets of the GPS signals have different authentication times.

The algorithms have been tested using real flight data and partially simulated flight data. The fast CHIMERA authentication channel achieves good accuracy, on the order of 2 m RMS in position for all types of IMUs. The slow CHIMERA authentication channel leads to RMS errors on the order of 10 m and peak errors on the order of 20 m when using a navigation-grade IMU and when all signals are authenticated simultaneously. Performance deteriorates markedly for tactical-grade and MEMS-grade IMUs. Staggering of the authentication times can improve the performance of the filters with slow-channel CHIMERA authentication, but the best such performance, the performance when using a navigation-grade IMU, still has RMS position errors on the order of 6 m and peak errors on the order of 15 m.

References

1. Esswein, M.C. and Psiaki, M.L., "GPS Spoofing Resilience via NMA/Watermarks Authentication and IMU Prediction," *Proc. ION GNSS+ 2021*, Sept. 21-24, 2021, St. Louis, MO.

APPENDIX A

GPS Spoofing Resilience via NMA/Watermarks Authentication and IMU Prediction

LIST OF FIGURES

Page

GPS Spoofing Resilience via NMA/Watermarks Authentication and IMU Prediction

Michael C. Esswein and Mark L. Psiaki
Virginia Tech, Blacksburg, VA 24061, USA

BIOGRAPHIES

Michael C. Esswein is a Ph.D. student currently studying at Virginia Tech. He holds a B.S. in Aerospace Engineering from the University at Buffalo. Research interests include estimation methods, GNSS anti-spoofing, and orbital mechanics.

Mark L. Psiaki is Professor and Kevin T. Crofton Faculty Chair of Aerospace and Ocean Engineering at Virginia Tech. He is also Professor Emeritus of Mechanical and Aerospace Engineering at Cornell University. He holds a Ph.D. in Mechanical and Aerospace Engineering from Princeton University. He is a Fellow of both the ION and the AIAA. His research interests are in the areas of navigation, spacecraft attitude and orbit determination, remote sensing, and general methods for estimation, filtering, and detection.

Abstract

A tightly coupled GPS/IMU estimation algorithm is developed assuming that all received measurements must first be authenticated by CHIps MEssage Robust Authentication (CHIMERA). CHIMERA is designed to authenticate incoming GPS signals through two methods referred to as the fast and slow channels. This paper analyzes the accuracy of estimation algorithms for both of these channels when using an Inertial Measurement Unit (IMU) to compensate for authentication delay, and it considers the effects of different quality IMUs. This paper also introduces a concept of authentication staggering as a possible approach to improve location and attitude accuracy. The estimation algorithm is modified to account for authentication staggering and different possible estimation architectures are developed for this purpose. The results indicate that the fast channel produces typical GPS navigation accuracy for different quality IMUs while the slow channel has moderately degraded navigation accuracy even with a navigation-grade IMU and highly degraded accuracy with tactical- and MEMS-grade IMUs. Staggering the authentication times of the GPS satellites can be used to improve navigation accuracy for the slow channel.

I. Introduction

One of the main current threats to resilient GPS Position, Navigation, and Timing (PNT) is the possibility of a spoofing attack. There are a number of different anti-spoofing techniques that have been proposed in order to mitigate this threat. One such technique is known as Navigation Message Authentication (NMA). NMA typically use a public-key/private-key method within the GPS navigation message in order to authenticate the incoming signal.[1] A realization of an NMA technique for the L1C signal, proposed by Anderson et al.,[2] is known as Chips-Message Robust Authentication (CHIMERA). CHIMERA uses navigation message authentication along with spreading code puncturing. In doing so, CHIMERA attempts to combat spoofing attacks by tying the signal to its source using cryptographic methods.

The current design of CHIMERA can be described by the following operations. A user receives an incoming signal. The user stores raw samples that have been output by the RF front-end Analog to Digital Converter (ADC)

for the duration of an authentication interval. The user next authenticates an incoming digital signature using the users' public key. If the signature is determined to be authentic, then the marker key, which can be obtained multiple ways, such as through the navigation message, is run through an algorithm to create cipher texts that derive marker chip values and their placements. The newly derived markers chips are correlated with all the stored samples. If the correlation succeeds, then the signal is deemed authentic by CHIMERA.

CHIMERA has two "channels", the slow channel and the fast channel. Channels are methods by which a user can authenticate the incoming signals. The slow channel assumes that the authentication processes is fully contained within the incoming GPS signals. Therefore, this method obtains the marker keys solely through the digital signature within the navigation message. Using the slow channel, a receiver can verify that the incoming signal is authentic every three minutes. The fast channel uses a trusted source that is external to the incoming signal in order to obtain new marker keys and the digital signature at higher data rates. This requires the receiver to have an independent broadcast communication link from this external source. This alternate method greatly decreases the time needed to authenticate because it does not require waiting for the navigation message to be decoded. The time for authentication can be on the order of seconds.

Authentication staggering is a concept that will play an important role in the latter part of this paper. This concept refers to having the digital signature, which lies within the navigation message of different GPS satellites, decoded at different times. This means that the authentication of GPS observables from different GPS satellites may occur at different times. Not every GPS satellite must have a different authentication time than the others. GPS satellites that have the same authentication time are considered to be part of the same authentication group.

For the purposes of analysis, some assumptions must be made about CHIMERA. For all simulations in this paper, the slow channel will take 3 minutes to authenticate while the fast channel will take 2 seconds to authenticate. For the cases that use staggering, all authentication groups will be evenly spaced from each other. For cases without authentication staggering, authentication occurs at the same time for all GPS signals.

This paper makes four contributions to the use of CHIMERA for authentication of GPS signals. The first is incorporation of an tightly coupled GPS/INS navigation filter to handle the authentication delays produced by CHIMERA. The second contribution is the implementation of authentication staggering with CHIMERA in order to improve navigation accuracy and the development of a brute-force navigation filter that uses IMU data and staggered, delayed CHIMERA authentications. Note that the abbreviations IMU and INS are used interchangeably in this paper, the latter being shorthand for Inertial Navigation System. The third contribution is a pair of alternate estimation architectures that can be used to improve computation speed in the staggered CHIMERA case relative to the brute-force approach. The fourth contribution is an initial evaluation of the performance of these algorithms using real and simulated data.

This paper presents its contributions to IMU-aided GPS navigation with CHIMERA authentication delays in four main sections followed by a conclusions section and appendices. Section II details the tightly coupled GPS/IMU navigation filter that is used to account for CHIMERA authentication delays. This section includes state and measurement models as well as the SRIF implementation that is used for the estimation algorithm. Section III discusses the results that are achieved when applying this algorithm to the fast and slow channel for different quality real and simulated IMUs. This section's results are based on GPS data taken onboard an aircraft. Section IV details the concept of authentication staggering, develops a brute-force filter architecture for dealing with staggered authentication, and compares its results to the non-staggered results for the slow channel. Section V develops alternate architectures that can be used to process the staggered data differently than a brute-force approach. These alternate methods do not improve navigation accuracy, but, they may be used to improve computational speed. Section VI presents this paper's conclusions. Appendix A gives the detailed equations of a staggered authentication filter that uses multiple filters for the different staggered groups of GPS data. Appendix B gives the detailed equations for an alternative staggered authentication filter that uses multiple partial filters to accomplish the same calculations.

II. Tightly Coupled GPS/IMU Navigation Filter that Accounts for CHIMERA Delays

A tightly coupled GPS/IMU navigation filter is an estimator where the attitude, position, and velocity components of the state are propagated by IMU model replacement and the filter's measurements are the GPS observables. The state vector used by this estimator is,

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{q}_{IMU/ECEF} \\ \boldsymbol{r}_{ECEF} \\ \boldsymbol{v}_{ECEF} \\ \boldsymbol{b}_{rg} \\ \boldsymbol{b}_{acc} \\ c\,\delta_r \\ c\,\dot{\delta}_r \\ \bar{\boldsymbol{b}}_{rg} \\ \bar{\boldsymbol{b}}_{acc} \end{bmatrix} \tag{1}$$

where $\boldsymbol{q}_{IMU/ECEF}$ is a 4×1 vector for the quaternion that parameterizes the rotation from the Earth-Centered/Earth-Fixed (ECEF) WGS-84 coordinate frame to the IMU body-fixed coordinate frame, $\boldsymbol{r}_{ECEF}$ is the 3×1 position of the IMU accelerometer in the ECEF frame, $\boldsymbol{v}_{ECEF}$ is the corresponding 3×1 velocity of the IMU in the ECEF frame, $\boldsymbol{b}_{rg}$ & $\boldsymbol{b}_{acc}$ are the 3×1 in-run stability bias vectors of the IMU's rate gyro and accelerometer, respectively, $c\delta_r$ is the scalar range-equivalent receiver clock offset, $c\dot{\delta}_r$ is the scalar range-rate-equivalent receiver clock offset rate, and $\bar{\boldsymbol{b}}_{rg}$ & $\bar{\boldsymbol{b}}_{acc}$ are the 3×1 repeatability bias vectors of the IMU's rate gyro and accelerometer, respectively. This state vector has 24 elements.

A. State Propagation Model

The quaternion, position, and velocity state is propagated using IMU model replacement. This means that IMU measurement models are used to define their dynamic propagation equations.

The quaternion dynamic propagation model is takes the form:

$$\dot{\boldsymbol{q}}_{IMU/ECEF} = \frac{1}{2}\Omega\left[\tilde{\boldsymbol{\omega}}_{rg} - \boldsymbol{b}_{rg} - \bar{\boldsymbol{b}}_{rg} - A(\boldsymbol{q}_{IMU/ECEF})\boldsymbol{\omega}_{earth} - \boldsymbol{\nu}_{rg}\right]\boldsymbol{q}_{IMU/ECEF} \tag{2}$$

where $\tilde{\boldsymbol{\omega}}_{rg}$ is the raw rate-gyro output vector,

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix}$$

the Earth's rotation rate vector in ECEF coordinates is,

$$\boldsymbol{\omega}_{earth} = \begin{bmatrix} 0 \\ 0 \\ \omega_e \end{bmatrix}$$

with $\omega_e = 7.2921151467 \times 10^{-5}$ rad/sec being the nominal Earth rotation rate, $A(\boldsymbol{q})$ is the 3×3 orthonormal rotation matrix associated with the quaternion $\boldsymbol{q}$, and $\boldsymbol{\nu}_{rg}$ is the rate-gyro angular random walk white noise.

The position and velocity dynamic propagation models consist of the following pair of coupled equations:

$$\dot{\boldsymbol{r}}_{ECEF} = \boldsymbol{v}_{ECEF} \tag{3}$$

$$\begin{aligned}\dot{\boldsymbol{v}}_{ECEF} =& [A(\boldsymbol{q}_{IMU/ECEF})]^T(\tilde{\boldsymbol{a}}_{acc} - \boldsymbol{b}_{acc} - \bar{\boldsymbol{b}}_{acc} - \boldsymbol{\nu}_{acc}) + \boldsymbol{g}(\boldsymbol{r}_{ECEF}) - 2\boldsymbol{\omega}_{earth} \times \boldsymbol{v}_{ECEF} \\ & -\boldsymbol{\omega}_{earth} \times \boldsymbol{\omega}_{earth} \times \boldsymbol{r}_{ECEF}\end{aligned} \tag{4}$$

where $\tilde{\boldsymbol{a}}_{acc}$ is the raw IMU accelerometer output and $\boldsymbol{\nu}_{acc}$ is the accelerometer velocity random walk white noise.

The IMU bias models' discrete-time dynamic propagations take the forms:

$$\boldsymbol{b}_{rg/acc(k+1)} = e^{-\frac{\Delta t}{\tau}}\boldsymbol{b}_{rg/acck} + \boldsymbol{w}_{rg/acck} \tag{5}$$

$$\boldsymbol{b}_{rg/acc(k+1)} = \boldsymbol{b}_{rg/acck} \tag{6}$$

where τ is a Markov process time constant. $\Delta t = t_{k+1} - t_k$ is the discrete-time interval between the times t_k and t_{k+1} at which the in-run stability bias states $\boldsymbol{b}_{rg/acck}$ and $\boldsymbol{b}_{rg/acc(k+1)}$ apply, and $\boldsymbol{w}_{rg/acck}$ is the discrete-time white noise vector that drives the corresponding first-order Markov process. Note that the subscript $()_{rg/acc}$ is used here as short-hand to indicate that there is one such equation or quantity for the IMU rate gyro and another for the IMU accelerometer.

The receiver clock discrete-time dynamic propagation drift model takes the following form:

$$\begin{bmatrix} c\,\delta_{r(k+1)} \\ c\,\dot{\delta}_{r(k+1)} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} c\,\delta_{rk} \\ c\,\dot{\delta}_{rk} \end{bmatrix} + \begin{bmatrix} w_{clk} \\ w_{driftk} \end{bmatrix} \tag{7}$$

where w_{clk} is the discrete-time scalar white noise component that drives clock offset random walk and w_{driftk} is the discrete-time scalar white noise component that drives clock rate offset random walk.

A nonlinear discrete-time dynamic model for this system with process noise can then be written as:

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}_k(\boldsymbol{x}_k, \boldsymbol{w}_k, \boldsymbol{w}_{k+1}) \tag{8}$$

where,

$$\boldsymbol{f}_k(\boldsymbol{x}_k, \boldsymbol{w}_k, \boldsymbol{w}_{k+1}) = \begin{bmatrix} \boldsymbol{q}_{IMU/ECEF(k+1)} \\ \boldsymbol{r}_{ECEF(k+1)} \\ \boldsymbol{v}_{ECEF(k+1)} \\ \boldsymbol{b}_{rg(k+1)} \\ \boldsymbol{b}_{acc(k+1)} \\ c\,\delta_{r(k+1)} \\ c\,\dot{\delta}_{r(k+1)} \\ \bar{\boldsymbol{b}}_{rg(k+1)} \\ \bar{\boldsymbol{b}}_{acc(k+1)} \end{bmatrix} \tag{9}$$

where the vectors $\boldsymbol{w}_k$ and $\boldsymbol{w}_{k+1}$ are white process noise vectors at samples k and $k+1$. They include discrete-time approximations of the effects of the rate-gyro angular random walk continuous-time white noise $\boldsymbol{\nu}_{rg}$ and the accelerometer velocity random walk continuous-time white noise $\boldsymbol{\nu}_{acc}$. These continuous-time white noise vectors are approximated as discrete-time white noise sequences where the value at a given sample instant is subtracted from the corresponding IMU output at that sample time, and the resulting partially corrected IMU measurement is deemed to apply for one half of a sample interval before and after the corresponding IMU sample time. That is why the white noise vectors $\boldsymbol{w}_k$ and $\boldsymbol{w}_{k+1}$ are both required for the dynamic propagation from time t_k to time t_{k+1} rather than just $\boldsymbol{w}_k$. Of course, the discrete-time white noise vector $\boldsymbol{w}_k$ also contains the discrete-tme white noise drivers of the IMU in-run stability bias drifts that are modeled in Eq. (5) and of the receiver clock and clock rate drifts that are modeled in Eq. (7).

In order to determine the state at sample time t_{k+1}, Eqs. (2-4) must be integrated from time t_k to time t_{k+1}. One possible method to achieve this is to use implicit trapezoidal integration. The integration scheme for this method is shown below,

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \frac{1}{2}\Delta t\left[\dot{\boldsymbol{x}}(t_k, \boldsymbol{x}_k) + \dot{\boldsymbol{x}}(t_{k+1}, \boldsymbol{x}_{k+1})\right] \tag{10}$$

where Δt is the sample interval.

It is implicit because $\boldsymbol{x}_{k+1}$ appears on both sides of the equation. To find the $\boldsymbol{x}_{k+1}$ that satisfies this equation, Newton's method is typically used. This method implies that the IMU measurements at t_k and time t_{k+1} must be known to propagate.

Three useful Jacobian first partial derivatives of the dynamics are defined as follows:

$$F_k = \left.\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{x}_k}\right|_{(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})}$$
$$\Gamma_{k,k} = \left.\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{w}_k}\right|_{(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})}$$
$$\Gamma_{k,k+1} = \left.\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{w}_{k+1}}\right|_{(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})}$$

where $\boldsymbol{f}_k$ is a nonlinear function defined in Eq. (9).

B. Measurement Model

The GPS observables that are used by the filter are pseudorange and Doppler shift. The model for the pseudorange is,

$$P^j = \sqrt{(x^j - x)^2 + (y^j - y)^2 + (z^j - z)^2} + c(\delta_r - \delta^j) + c\,\delta^j_{iono} + c\,\delta^j_{na} + \nu^j_{pseudo} \tag{11}$$

where x, y, and z are the components of the user position vector $\vec{\boldsymbol{r}}_{ECEF}$ and where x^j, y^j, and z^j are the ECEF position components of GPS satellite j at the broadcast time of the signal, but are rotated into the ECEF frame that applies at the time of signal reception.

The Doppler shift can be approximated by numerically differentiating the Accumulated Delta Range (ADR) model, where the ADR model takes a similar form to the pseudorange:

$$ADR^j = \lambda\phi^j = \sqrt{(x^j - x)^2 + (y^j - y)^2 + (z^j - z)^2} + c(\delta_r - \delta^j) - c\,\delta^j_{iono} + c\,\delta^j_{na} + \nu^j_{adr} \tag{12}$$

Then the Doppler shift, D^j, can be modeled approximately by,

$$-\lambda D_k^j \approx \frac{ADR_{k,-2}^j - 8ADR_{k,-1}^j + 8ADR_{k,+1}^j - ADR_{k,+2}^j}{12\Delta_{FD}} \tag{13}$$

where,

$$ADR_{k,l}^j = ADR^j\{x_k + [l\Delta_{FD}/(1+\dot{\delta}_{rk})]\dot{x}_k\} \tag{14}$$

is a time-perturbed version of the ADR using the time perturbation $l\Delta_{FD}$ in erroneous receiver clock time, which translates into the time perturbation $[l\Delta_{FD}/(1+\dot{\delta}_{rk})]$ in true time. The interval Δ_{FD} is the finite difference interval that is used for this 5-point finite-difference approximation of the ADR time rate of change.

The pseudorange and carrier Doppler shift measurement times do not necessarily correspond to the IMU sample times. Typically they are determined at some time $\tilde{t}_k$ that lies between two IMU sample times: $t_k \leq \tilde{t}_k < t_{k+1}$. Therefore, a partial dynamic propagation from IMU sample time t_k to GPS measurement sample time $\tilde{t}_k$ is needed as part of the measurement model.

The nonlinear measurement model vector function can then be written as:

$$\boldsymbol{h}_k(\boldsymbol{x}_k, \boldsymbol{w}_k, \boldsymbol{w}_{k+1}) = \begin{bmatrix} P^1 \\ \vdots \\ P^N \\ -\lambda D^1 \\ \vdots \\ -\lambda D^N \end{bmatrix} \tag{15}$$

where N is the number of available GPS signals and where the presence of the process noise vectors $\boldsymbol{w}_k$ and $\boldsymbol{w}_{k+1}$ is required to accomplish the dynamic propagation from t_k to $\tilde{t}_k$.

This function is then used to define the nonlinear measurement model:

$$\boldsymbol{y}_{k+1} = \boldsymbol{h}_k(\boldsymbol{x}_k, \boldsymbol{w}_k, \boldsymbol{w}_{k+1}) + \boldsymbol{\nu}_{k+1} \tag{16}$$

where $\boldsymbol{\nu}_{k+1}$ is the measurement noise vector. It is assumed to be a zero-mean Gaussian random vector with covariance matrix $R_{\nu\nu(k+1)}^{-1}\ R_{\nu\nu(k+1)}^{-\mathrm{T}}$. Note that $()^{-\mathrm{T}}$ refers to the inverse of the transpose of the matrix in question.

Three useful Jacobian first partial derivatives of the measurement model vector function are,

$$\begin{aligned} H_{xk} &= \left.\frac{\partial \boldsymbol{h}_k}{\partial \boldsymbol{x}_k}\right|_{(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})} \\ H_{k,wk} &= \left.\frac{\partial \boldsymbol{h}_k}{\partial \boldsymbol{w}_k}\right|_{(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})} \\ H_{k,w(k+1)} &= \left.\frac{\partial \boldsymbol{h}_k}{\partial \boldsymbol{w}_{k+1}}\right|_{(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})} \end{aligned} \tag{17}$$

where $\boldsymbol{h}_k$ is the measurement model function that is described in this section.

C. SRIF Implementation

This section describes how state estimation would be implemented if there were no need to account for delays. These developments are included for reference purposes. The estimator used is an Extended Kalman Filter (EKF) that is implemented using Square Root Information Filter (SRIF) techniques.

A single sample interval of the filter starts with *a posteriori* estimates of the state vector and the process noise vector, $\hat{\boldsymbol{x}}_k$ and $\hat{\boldsymbol{w}}_k$, at sample time t_k along with an *a priori* estimate of the process noise vector, $\bar{\boldsymbol{w}}_{k+1}$, at sample time t_{k+1}. It also starts with a pair of coupled *a posteriori* square-root information equations that model the estimation error uncertainty in $\hat{\boldsymbol{x}}_k$ an $\hat{\boldsymbol{w}}_k$ along with an *a priori* square-root information equation for the uncertainty in $\bar{\boldsymbol{w}}_{k+1}$. Grouping them together into a single system of square-root information equations yields:

$$\begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}_{wwk} & 0 & \hat{R}_{wxk} \\ 0 & 0 & \hat{R}_{xxk} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1} \\ \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \end{bmatrix} = - \begin{bmatrix} \boldsymbol{v}_{\bar{\boldsymbol{w}}_{k+1}} \\ \boldsymbol{v}_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}_{\hat{\boldsymbol{x}}_k} \end{bmatrix} \tag{18}$$

where $\bar{R}_{ww(k+1)}$, $\hat{R}_{wwk}$, $\hat{R}_{wxk}$, and $\hat{R}_{wxk}$ are square-root information matrices of appropriate dimensions and $\boldsymbol{v}_{\bar{\boldsymbol{w}}_{k+1}}$, $\boldsymbol{v}_{\hat{\boldsymbol{w}}_k}$, and $\boldsymbol{v}_{\hat{\boldsymbol{x}}_k}$ are uncorrelated, zero-mean, identity-covariance Gaussian random vectors of appropriate dimensions. This equation implies that the *a priori* error covariance in $\bar{\boldsymbol{w}}_{k+1}$ is $\bar{R}^{-1}_{ww(k+1)}\bar{R}^{-\mathrm{T}}_{ww(k+1)}$. Similarly, the *a posteriori* error covariance of the vector $[\hat{\boldsymbol{w}}_k; \hat{\boldsymbol{x}}_k]$ is

$$P_{wxk} = \begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \end{bmatrix}^{-1} \begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \end{bmatrix}^{-\mathrm{T}} \tag{19}$$

The nonlinear state dynamic propagation model and the measurement model are used to determine EKF approximations to the *a priori* state and measurement at sample time t_{k+1}:

$$\begin{aligned} \bar{\boldsymbol{x}}_{k+1} &= \boldsymbol{f}_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1}) \\ \bar{\boldsymbol{y}}_{k+1} &= \boldsymbol{h}_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1}) \end{aligned} \tag{20}$$

These values and the Jacobians of the dynamics and measurement models are used to define the following EKF-type linearized dynamics and measurement model equations:

$$\begin{aligned} \boldsymbol{x}_{k+1} - \bar{\boldsymbol{x}}_{k+1} &= F_k(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + \Gamma_{k,k}(\boldsymbol{w}_k - \hat{\boldsymbol{w}}_k) + \Gamma_{k,k+1}(\boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1}) \\ \boldsymbol{y}_{k+1} - \bar{\boldsymbol{y}}_{k+1} &= H_{xk}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + H_{k,wk}(\boldsymbol{w}_k - \hat{\boldsymbol{w}}_k) + H_{k,w(k+1)}(\boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1}) + \boldsymbol{\nu}_{k+1} \end{aligned} \tag{21}$$

The EKF/SRIF calculations also use the square-root information equation for the measurement noise in $\boldsymbol{y}_{k+1}$. It is

$$R_{\nu\nu(k+1)}\boldsymbol{\nu}_{k+1} = -\boldsymbol{v}_{\nu(k+1)} \tag{22}$$

where $\boldsymbol{v}_{\nu(k+1)}$ is a zero-mean, identity-covariance Gaussian random vector.

The combined dynamic propagation and measurement update calculations of the SRIF implementation of the EKF start by stacking the square-root information equations in Eq. (18) on top of the square-root information equation in Eq. (22). Next, the second line of Eq. (21) is solved for $\boldsymbol{v}_{\nu(k+1)}$, the result is substituted the $\boldsymbol{v}_{\nu(k+1)}$ square-root information. Finally, the first line of Eq. (21) is solved for $\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$, and the result is substituted into the combined system of square-root information equations. The resulting system of equations is:

$$\begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}_{wwk} - \hat{R}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{wxk}F_k^{-1} \\ -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{xxk}F_k^{-1} \\ R_{\nu\nu(k+1)}(H_{k,wk} - H_{xk}F_k^{-1}\Gamma_{k,k}) & R_{\nu\nu(k+1)}(H_{k,w(k+1)} - H_{xk}F_k^{-1}\Gamma_{k,k+1}) & R_{\nu\nu(k+1)}H_{xk}F_k^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1} \\ \boldsymbol{x}_{k+1} - \bar{\boldsymbol{x}}_{k+1} \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 0 \\ R_{\nu\nu(k+1)}(\boldsymbol{y}_{k+1} - \bar{\boldsymbol{y}}_{k+1}) \end{bmatrix} - \begin{bmatrix} \boldsymbol{v}_{\bar{w}(k+1)} \\ \boldsymbol{v}_{\hat{w}k} \\ \boldsymbol{v}_{\hat{x}k} \\ \boldsymbol{v}_{\nu(k+1)} \end{bmatrix} \tag{23}$$

The combined SRIF dynamic propagation and measurement update performs the following orthonormal upper-triangular factorization of the large coefficient matrix on the left-hand side of the preceding equation:

$$T_k \begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & 0 & \hat{R}_{xx(k+1)} \\ 0 & 0 & 0 \end{bmatrix} =$$

$$\begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}_{wwk} - \hat{R}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{wxk}F_k^{-1} \\ -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{xxk}F_k^{-1} \\ R_{\nu\nu(k+1)}(H_{k,wk} - H_{xk}F_k^{-1}\Gamma_{k,k}) & R_{\nu\nu(k+1)}(H_{k,w(k+1)} - H_{xk}F_k^{-1}\Gamma_{k,k+1}) & R_{\nu\nu(k+1)}H_{xk}F_k^{-1} \end{bmatrix} \tag{24}$$

where the large block matrix on the right-hand side of this equation constitutes the input to the factorization and the matrices on the left-hand side constitute the outputs. The output T_k is a large orthonormal matrix of appropriate dimensions. The output matrices R^*_{wwk}, $\hat{R}_{ww(k+1)}$, and $\hat{R}_{xx(k+1)}$ are square, upper-triangular matrices, and the output matrices $R^*_{w(k)w(k+1)}$, $R^*_{w(k)x(k+1)}$, and $\hat{R}_{wx(k+1)}$ are general matrices. The orthonormal matrix T_k is transposed and used in the following equation:

$$\begin{bmatrix} \boldsymbol{z}^*_{wk} \\ \hat{\boldsymbol{z}}_{w(k+1)} \\ \Delta\hat{\boldsymbol{z}}_{x(k+1)} \\ \hat{\boldsymbol{z}}_{r(k+1)} \end{bmatrix} = T_k^T \begin{bmatrix} 0 \\ 0 \\ 0 \\ R_{\nu\nu(k+1)}(\boldsymbol{y}_{k+1} - \bar{\boldsymbol{y}}_{k+1}) \end{bmatrix} \tag{25}$$

in order to compute the various vectors on the left-hand side.

Next, increments to the state and process noise vector estimates at sample time t_{k+1} are computed as follows:

$$\delta\boldsymbol{x}_{k+1} = \hat{R}^{-1}_{xx(k+1)}\Delta\hat{\boldsymbol{z}}_{x(k+1)} \tag{26}$$

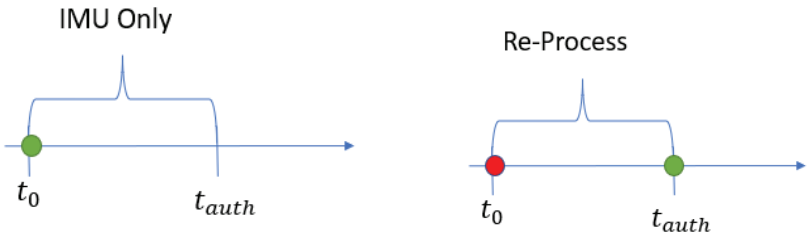
$$\delta\boldsymbol{w}_{k+1} = \hat{R}^{-1}_{ww(k+1)}(\hat{\boldsymbol{z}}_{w(k+1)} - \hat{R}_{wx(k+1)}\delta\boldsymbol{x}_{k+1}) \tag{27}$$

Finally the updated state and process noise estimates at time t_{k+1} are computed as follows:

$$\hat{\boldsymbol{x}}_{k+1} = \bar{\boldsymbol{x}}_{k+1} + \delta\boldsymbol{x}_{k+1} \tag{28}$$

$$\hat{\boldsymbol{w}}_{k+1} = \bar{\boldsymbol{w}}_{k+1} + \delta\boldsymbol{w}_{k+1} \tag{29}$$

The actual implementation of this SRIF incorporates the quaternion multiplicative error technique that is discussed in [3] and [4]. These techniques entail using a 3-element quaternion uncertainty vector because the magnitude of the unit-normalized attitude quaternion has no uncertainty. It is known to equal 1 exactly. The current derivations would become even more complicated than they already are if they included all of the details about how to incorporate quaternion multiplicative uncertainty. Therefore, this paper's filter derivations omit that complication. All of its results, however, use multiplicative quaternion uncertainty. The types of changes that occur in this



(a) Propagate from t_0 to t_{auth} (no measurements)

(b) Re-Propagate from t_0 to t_{auth} (all measurements)

Fig. 1: The two-part estimation process over a single authentication interval.

technique start with a 4×1 quaternion error such as $\boldsymbol{q}-\hat{\boldsymbol{q}}$ and project it down onto the 3-dimensional error space that is perpendicular to the estimate in $\hat{\boldsymbol{q}}$. These projections reduce the number of elements in a state estimation error vector such as $\boldsymbol{x}_k-\hat{\boldsymbol{x}}_k$ from 24 to 23, and the dimensions of the corresponding square-root information matrices and covariance matrices are also reduced from 24 to 23. The incorporation of the needed quaternion error projection techniques into this paper's methods are left as an exercise for the reader.

D. GPS/INS CHIMERA Algorithm

This subsection presents a method to incorporate CHIMERA authentication delays into the GPS/INS estimator described in the previous subsection. This section assumes that all observables will be authenticated at the same time. The incorporation of these delays into the estimator is done with a two step process. First, when no incoming observables can be authenticated, the estimator will simply propagate the state and square root information matrices without any measurement update. Second, once the observables are authenticated, the estimator will return to the nearest saved previous epoch state and rerun the estimation equations described in Subsection II.C only now including the measurement update from all previous observables up to the current time. Figure 1 shows this visually. In this example, the green dot indicates the current time. In subfigure (a), the estimator begins with authenticated estimates at time t_0 and runs without incorporating any measurements until it reaches time t_{auth}. Once t_{auth} is the current time, the next process shown in subfigure (b) begins. In this subfigure, the green dot is still the current time. The red dot indicates the saved epoch time, where the previously authenticated state and process noise estimates and their information matrices were stored for time t_0. The estimator will then use the saved authenticated quantities at the red dot and re-run the estimator to the current time using all measurements obtained during this interval. The updated state and information matrices at time t_{auth} will now become the new saved authenticated quantities for the next interval. This process will repeat for each future time interval that extends from the time of a successful authentication to the time of the next potential authentication.

The modification to the equations in Subsection II.C is relatively straightforward for the initial Fig. 1a portion of the algorithm when there is no authenticated data. During this interval, the final row of the large matrix on the left-hand side of Eq. (24) is omitted due to the lack of measurements. Equations (25)-(27) are omitted because all of the corresponding results are identically zero. Equations (28) and (29) are evaluated using $\delta\boldsymbol{x}_{k+1}=0$ and $\delta\boldsymbol{w}_{k+1}=0$. Note that these same measurement-update-less calculations are needed even for the Fig. 1b pass that uses authenticated data. They are needed for any IMU sample intervals when no GPS data are available. Given that an IMU may sample at 50-100 Hz while GPS data may be available at 1-20 Hz, many such update-less intervals exist even when working with authenticated data.

III. Analysis of the Fast and Slow CHIMERA Channels using a Variety of IMUs

In order to determine the viability of the tightly coupled GPS/IMU estimator, a number of factors must be considered. Two key factors are the CHIMERA authentication latency and the quality of the IMU used. A number of test cases will be compared to see how these factors affect navigation accuracy. The same GPS measurements will be used in every test case. These measurements come from an Inertial Labs GPS/IMU system that was recording GPS/IMU observables onboard an aircraft. The GPS sampling rate was set to 20 Hz, and the IMU sampling rate was set to 100 Hz. As only one IMU was recording real measurements on the aircraft, three other IMU's measurements were simulated. This was done by determining the body accelerations and the angular rates from interpolated truth data of the aircraft's position, velocity, and attitude. Then noise was applied to these measurements, based on three commercial IMUs of varying quality, to create simulated IMU measurements at the same times as the real measurements. Figure 2 shows the noise parameters for the three Honeywell IMUs that were simulated. The noise parameters of the real IMU data are not known, but seem to be close to the noise parameters of the tactical grade Honeywell IMU.

IMU/INS	Gyro Stability Bias (deg/hr)	Accel. Stability Bias (mg)	Gyro Repeatability Bias (deg/hr)	Accel. Repeatability Bias (mg)	ARW (deg/sqrt(hr))
HG9900 (Navigation Grade IMU)	.0006	.01	.003	.025	.002
HG1900 (Tactical Grade IMU)	1	.05	10	1	.06
HG1125 (MEMS IMU)	5	.3	120	1.5	.3

Fig. 2: Table of IMU Parameters.

Using the one set of real IMU measurements and the three sets of simulated IMU measurements, results were tabulated for both the fast channel and slow channel. Figure 3 shows the results for the different IMUs when using the fast channel. It can be seen that as the quality of the IMU decreases the RMS error increases slightly. However, even with the lowest quality IMU, the RMS position error never rises above 2 meters, and the RMS velocity error never rises above 0.25 meters/sec. For comparison, the GPS/IMU RMS errors without any CHIMERA authentication delays are approximately 1.47 meters for position and 0.03 meters/sec for velocity.

Figure 4 shows plots of the position error time history components in ECEF coordinates as well as the navigation filter's corresponding computed 1-sigma bounds for the case using IMU HG9900 for the fast channel. This case is a representative example of how the fast channel is able to maintain accuracy over time. In this case, position error never exceeds 2.5 meters in any direction. The fast channel is able to maintain close to nominal GPS navigation accuracy for all IMU qualities that were tested. This shows that incorporating an IMU is an effective method for handling CHIMERA authentications delays from the fast channel.

The 180 second authentication delays from the slow channel are much larger than the 2 second delays from the fast channel. Therefore, authentication via the slow CHIMERA channel presents a much more difficult test for the GPS/IMU system. Figure 5 shows the results for the different IMUs when using the slow channel. This figure shows that, even when using the highest quality IMU, the RMS position error is much larger than the nominal GPS navigation accuracy. It also shows that results get significantly worse when using lower quality IMUs. Figure 6 plots the position error component time histories in ECEF coordinates as well as their corresponding navigation

IMU/INS	RMS Position Error (m)	RMS Velocity Error (m/s)	RMS Quaternion Error
HG9900	1.488	0.031	1.2785e-4
HG1900	1.876	0.199	0.0026
HG1125	1.888	0.233	0.0228
Inertial Labs	1.722	0.189	0.0014

Fig. 3: Table of Fast Channel RMS Accuracy Results.

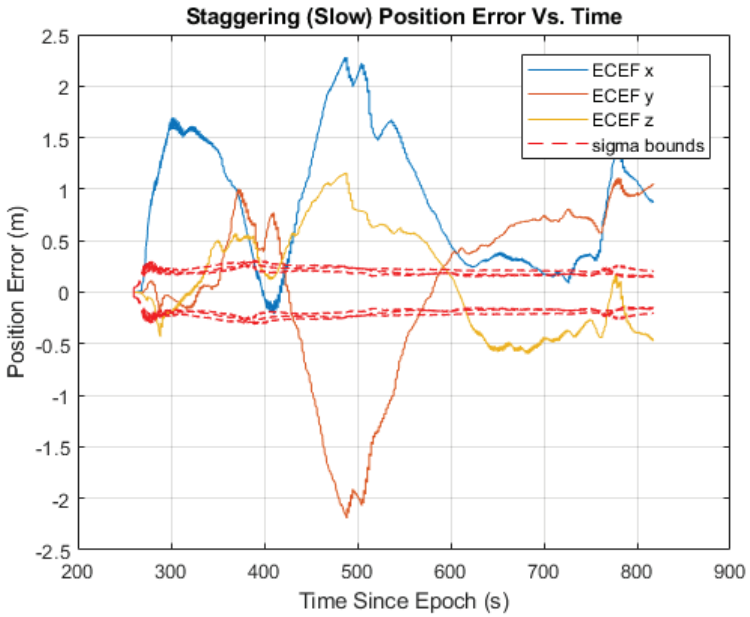


Fig. 4: Position error time histories and corresponding navigation filter computed 1-sigma values for fast channel.

filter computed 1-sigma bounds for the case that uses the IMU HG9900 for the slow channel. This figure not only shows the problem of high RMS error, but also highlights the fact that the worst-case error can reach over 20 meters in some instances even when the overall RMS error is much lower. This is due to the fact that the 3 minute authentication delay is too long to rely solely on the IMU.

IV. Authentication Staggering

One possible remedy to the problem of poor performance with slow channel CHIMERA authentication is to stagger the authentication intervals. Authentication staggering is the concept of separating the GPS satellites into different groups with each group having a different authentication time from the others. This does not change the

IMU/INS	RMS Position Error (m)	RMS Velocity Error (m/s)	RMS Quaternion Error
HG9900	9.45	0.14	1.122e-4
HG1900	339.05	6.15	0.007
HG1125	2,407.40	45.24	0.039
Inertial Labs	102.5759	1.4767	0.0020

Fig. 5: Table of Slow Channel RMS Accuracy Results.

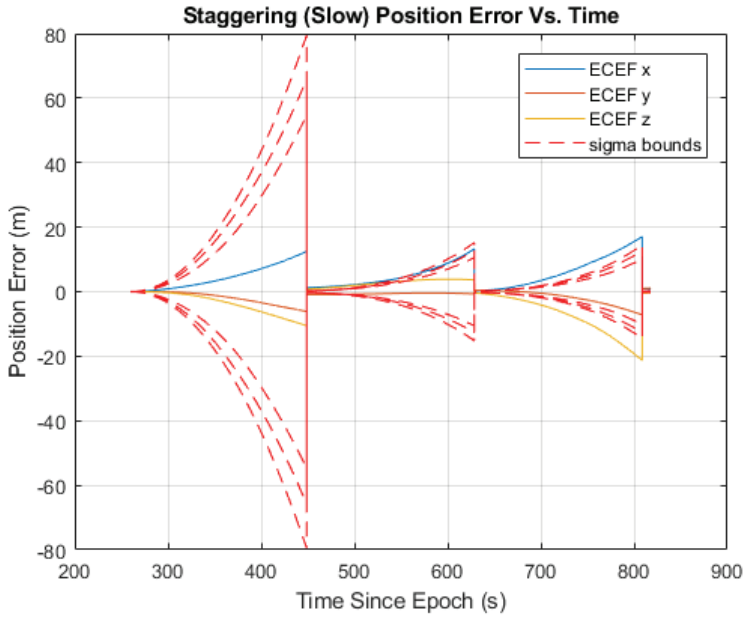


Fig. 6: Position error time histories and corresponding navigation filter computed 1-sigma values for the slow channel.

length of the authentication delay for any one signal because that is determined by the channel. However, it can reduce the maximum amount of IMU propagation time over which no satellites have been authenticated.

Figure 7 shows a timing diagram to help visualize this concept. This figure assumes that there are three authentication groups with each group denoted by a letter: A, B, or C. From this example, one can see that an authentication of some GPS observables will occur every minute. However, the interval between Group A's successive authentication times remains three minutes, the delay required by the slow channel. The same is true for the intervals between Group B authentication times and between Group C authentication times. The procedure for operating the navigation filter when including authentication staggering is similar to the procedure without

staggering. First, the state is updated solely using the IMU. Next, when an authentication time is reached, the interval is re-processed using the newly authenticated data. For authentication staggering, each one minute sub-interval must be processed $N + 1$ times, where N is the number of authentication groups, incorporating newly authenticated data each time.

Consider the sub-interval that extends from $t^C_{auth.}$ to the second $t^A_{auth.}$ in Fig. 7. The first processing of this one-minute interval occurs using IMU data only and starting from SRIF filter estimates based on all GPS data up through the first $t^A_{auth.}$ in the figure, based only on GPS data from authentication Groups B and C during the interval from the first $t^A_{auth.}$ to $t^B_{auth.}$, and based only GPS data from authentication Group C during the interval from $t^B_{auth.}$ to $t^C_{auth.}$. The second processing of the one-minute interval from from $t^C_{auth.}$ to the second $t^A_{auth.}$ occurs as part of a full 3-minutes of re-processing from the first $t^A_{auth.}$ to the second $t^A_{auth.}$ in Fig. 7. Again, processing starts with the filter state and covariance based on all the GPS data through the first $t^A_{auth.}$ It continues using all of the GPS data during its first minute of propagation and measurement update until it reaches $t^B_{auth.}$. During the interval from $t^B_{auth.}$ to $t^C_{auth.}$, it uses only GPS data from authentication Groups A and C. During the final interval from $t^C_{auth.}$ to the second $t^A_{auth.}$, it uses only GPS data from authentication Group A. The third and forth passes over the interval from $t^C_{auth.}$ to the second $t^A_{auth.}$ operate similarly, except that the third is part of a 3-minute propagation and update operation that extends from $t^B_{auth.}$ to a next $t^B_{auth.}$ that lies to the right of the figure's second $t^A_{auth.}$, and the forth is part of a 3-minute propagation and update operation that extends from $t^C_{auth.}$ to a next $t^C_{auth.}$ that lies even further to the right of the figure's second $t^A_{auth.}$. In fact, the procedure without staggering can be described as a special case of authentication staggering where N is equal to 1.

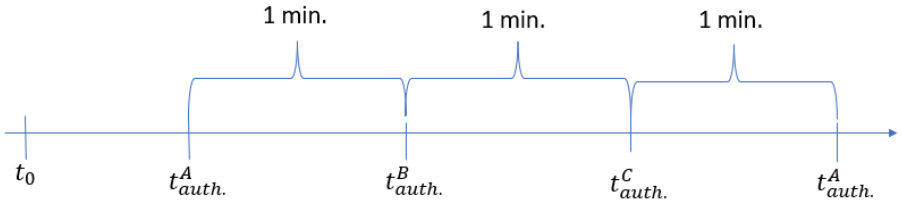


Fig. 7: CHIMERA staggering timing diagram.

The main idea behind staggering is that it is better to have some measurements sooner, rather than all measurements later. Figure 8 shows results for the same test cases as are covered in Fig. 5, except that these new results divide the visible GPS satellites into three authentication groups. One can see that, by comparing the two tables, the results improve when authentication staggering is introduced. The relative improvements in RMS position error for the tactical- and MEMS-grade IMUs are much larger than for the navigation-grade IMU. Figure 9 shows a plot of the position error component time histories in ECEF coordinates along with their corresponding computed navigation filter 1-sigma bounds for the case that uses the Honeywell IMU HG9900 for the slow channel with authentication staggering. Comparing Fig. 9 and Fig. 6, it can be seen that RMS error is reduced and the worst-case position error and position uncertainty is also reduced. This worst-case error difference is even more apparent when comparing authentication staggering and non authentication staggering for lower quality IMUs. The number of authentication groups used and the particular satellites in each group were arbitrarily chosen. Future work needs to be done to determine how much more effective an optimal grouping would be at improving navigation results.

V. Alternative Architectures

The preceding section's architecture for processing staggered authentication groups ensures that all data are processed appropriately. However, it uses a brute-force approach that constantly re-processes data over the same

IMU/INS	RMS Position Error (m)	RMS Velocity Error (m/s)
HG9900	5.981	0.099
HG1900	43.786	1.372
HG1125	173.907	7.795

Fig. 8: Table of Slow Channel RMS Accuracy Results with Staggering.

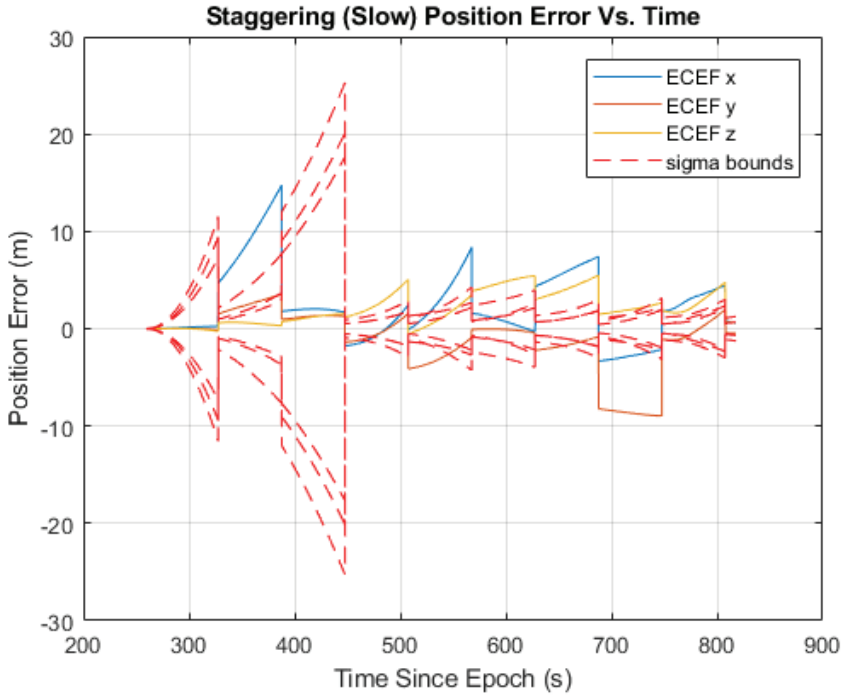


Fig. 9: Position error time histories and their corresponding navigation filter computed 1-sigma bounds when using the slow CHIMERA channel with staggering and a navigation-grade IMU.

interval. This section will discuss two alternative architectures that can be used to handle CHIMERA authentication delays and staggering in a more sophisticated manner.

A. Multiple Authentication Filter Staggering Architecture

The first of these two architectures will be referred to as the Multiple Authentication Filter (MAF) architecture. This architecture leverages the fact that each interval must be processed $N+1$ times. It implements $N+1$ filters in parallel in order to eliminate computation delays. Assuming two authentication groups, denoted by A and B, Fig. 10 shows a visual representation of which data groups each filter processes over multiple authentication intervals. Filter 1, which is the trusted filter whose outputs are used by the system, does not use GPS measurements. Filter

2 uses the measurements from only the upcoming authentication group. Filter 3 uses all available measurements from all groups. At each authentication time, Filter 2 passes its state estimate and square-root information matrix to Filter 1, and Filter 3 passes its state estimate and square-root information matrix to Filter 2. The detailed equations that implement this filter are presented in Appendix A. This pattern can be expanded to an arbitrary number of authentication groups. The details of how to do this for 3, 4, 5, etc. authentication groups are omitted for the sake of brevity. The interested reader should be able to work them out based in the foregoing pattern for 2 authentication groups and the equations in Appendix A.

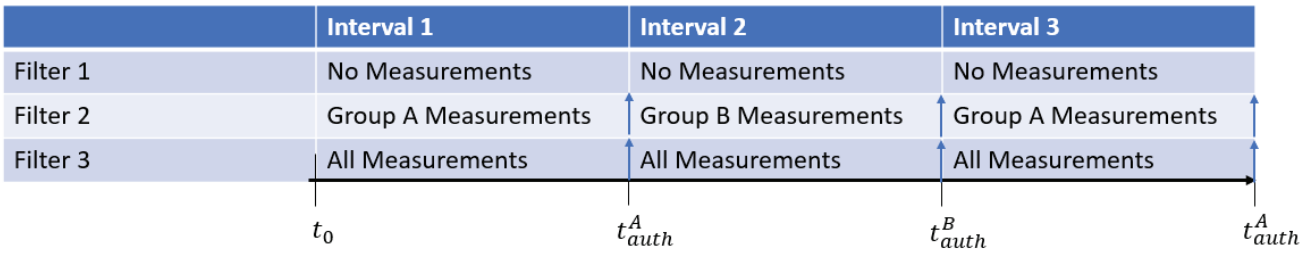


Fig. 10: Timing diagram for GPS data usage within the MAF architecture.

B. Multiple Authentication Partial Filter Staggering Architecture

The benefit of the previous method is its simplicity. It runs N+1 filters in parallel with different data sets applied to each filter and with estimates and uncertainties passed up the chain of filters at each authentication time. That architecture is still somewhat brute-force in that it runs N+1 full filters in parallel. The architecture presented in this subsection seeks to economize by re-using some common operations between the filters. This re-use comes at the expense of complexity, as the equations needed for this filter will demonstrate. The following example demonstrates how to implement this alternate architecture for three authentication groups. Note, however, that this architecture can be applied to any number of authentication groups.

Suppose the three staggered authentication groups are called Groups A, B, and C. At IMU sample time t_k, the SRIF version of this special EKF navigation filter starts with *a posteriori* estimates of the state vector and the process noise vector at sample time t_k, $\hat{\boldsymbol{x}}_k$ and $\hat{\boldsymbol{w}}_k$, that are based only on the data that have been fully authenticated up to this time. It also has an *a priori* estimate of the process noise vector at sample time t_{k+1}, $\bar{\boldsymbol{w}}_{k+1}$. The filter also has a pair of coupled *a posteriori* square-root information equations for $\boldsymbol{x}_k$ and $\boldsymbol{w}_k$ for its nominal authenticated filter and for each of the 3 authentication groups. In addition, it has an *a priori* square-root information equation for $\boldsymbol{w}_{k+1}$. Grouped together into a single large matrix-vector equation, these nine square-root information equations take the form:

$$\begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}^C_{wwk} & 0 & \hat{R}^C_{wxk} \\ \hat{R}^B_{wwk} & 0 & \hat{R}^B_{wxk} \\ \hat{R}^A_{wwk} & 0 & \hat{R}^A_{wxk} \\ \hat{R}_{wwk} & 0 & \hat{R}_{wxk} \\ 0 & 0 & \hat{R}^C_{xxk} \\ 0 & 0 & \hat{R}^B_{xxk} \\ 0 & 0 & \hat{R}^A_{xxk} \\ 0 & 0 & \hat{R}_{xxk} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1} \\ \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \end{bmatrix} = \begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^C_{wk} \\ \hat{\boldsymbol{z}}^B_{wk} \\ \hat{\boldsymbol{z}}^A_{wk} \\ 0 \\ \hat{\boldsymbol{z}}^C_{xk} \\ \hat{\boldsymbol{z}}^B_{xk} \\ \hat{\boldsymbol{z}}^A_{xk} \\ 0 \end{bmatrix} - \begin{bmatrix} \boldsymbol{v}_{\bar{\boldsymbol{w}}_{k+1}} \\ \boldsymbol{v}^C_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}^B_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}^A_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}^C_{\hat{\boldsymbol{x}}_k} \\ \boldsymbol{v}^B_{\hat{\boldsymbol{x}}_k} \\ \boldsymbol{v}^A_{\hat{\boldsymbol{x}}_k} \\ \boldsymbol{v}_{\hat{\boldsymbol{x}}_k} \end{bmatrix} \tag{30}$$

Suppose, without loss of generality, that authentication Group A is the one with the next new authentication time after t_k, that Group B has the next new authentication time after that of Group A, and that Group C has the latest

new authentication time. The first, fifth, and ninth rows of this system of equations are identical to the system in Eq. (18) and characterize the authenticated estimate at time t_k. The fourth and eighth lines of this equation, the ones corresponding to Group A, keep track of all of the new information from Group A GPS measurements that have been made since the most recent past Group A authentication time. The third and seventh lines correspond to Group B. They keep track of all of the new information from Group B GPS measurements since the most recent past Group B authentication time. The second and sixth lines correspond to Group C and keep track of all of the new information from Group C GPS measurements since the most recent past Group C authentication time. The vectors $\hat{\boldsymbol{z}}^A_{wk}$, $\hat{\boldsymbol{z}}^B_{wk}$, $\hat{\boldsymbol{z}}^C_{wk}$, $\hat{\boldsymbol{z}}^A_{xk}$, $\hat{\boldsymbol{z}}^B_{xk}$, and $\hat{\boldsymbol{z}}^C_{x}$ in this system of equations are known non-homogeneous terms that arise because of differences between the *a posteriori* estimates $\hat{\boldsymbol{x}}_k$ and $\hat{\boldsymbol{w}}_k$ and the estimates that would have been produced had the corresponding unauthenticated data been used to produce them. The terms $\boldsymbol{v}^A_{\hat{\boldsymbol{w}}_k}$, $\boldsymbol{v}^B_{\hat{\boldsymbol{w}}_k}$, $\boldsymbol{v}^C_{\hat{\boldsymbol{w}}_k}$, $\boldsymbol{v}^A_{\hat{\boldsymbol{x}}_k}$, $\boldsymbol{v}^B_{\hat{\boldsymbol{x}}_k}$, and $\boldsymbol{v}^C_{\hat{\boldsymbol{x}}_k}$ are uncorrelated, zero-mean, identity-covariance Gaussian random vectors that model the uncertainties in the corresponding information equations.

If there are no new GPS measurements in the IMU sample interval from t_k to t_{k+1}, then the filter performs only a dynamic propagation. The dynamic propagation is implemented using standard EKF/SRIF techniques. The first line of Eq. (20) is used to determine $\bar{\boldsymbol{x}}_{k+1}$. The first line of Eq. (21) is solved for $\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$. The result is substituted into the system of 9 square-root information equations. The resulting equations are defined by the coefficient matrix below and the equation that follows it. Note that the $\boldsymbol{\nu}_{prop}$ on the right-hand side of this equation equals the stacked vector $[\boldsymbol{v}_{\bar{\boldsymbol{w}}_{k+1}}; \boldsymbol{v}^C_{\hat{\boldsymbol{w}}_k}; \ldots; \boldsymbol{v}_{\hat{\boldsymbol{x}}_k}]$ that appears on the right-hand side of Eq. (30).

$$A_{prop} = \begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}^C_{wwk} - \hat{R}^C_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^C_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^C_{wxk}F_k^{-1} \\ \hat{R}^B_{wwk} - \hat{R}^B_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^B_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^B_{wxk}F_k^{-1} \\ \hat{R}^A_{wwk} - \hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{wxk}F_k^{-1} \\ \hat{R}_{wwk} - \hat{R}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{wxk}F_k^{-1} \\ -\hat{R}^C_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^C_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^C_{xxk}F_k^{-1} \\ -\hat{R}^B_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^B_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^B_{xxk}F_k^{-1} \\ -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{xxk}F_k^{-1} \\ -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{xxk}F_k^{-1} \end{bmatrix} \tag{31}$$

$$A_{prop} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1} \\ \boldsymbol{x}_{k+1} - \bar{\boldsymbol{x}}_{k+1} \end{bmatrix} = \begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^C_{wk} \\ \hat{\boldsymbol{z}}^B_{wk} \\ \hat{\boldsymbol{z}}^A_{wk} \\ 0 \\ \hat{\boldsymbol{z}}^C_{xk} \\ \hat{\boldsymbol{z}}^B_{xk} \\ \hat{\boldsymbol{z}}^A_{xk} \\ 0 \end{bmatrix} - \boldsymbol{\nu}_{prop} \tag{32}$$

The dynamic propagation performs a sequence of orthonormal/upper-triangular factorizations of subsets of rows of the coefficient matrix in this equation in order to complete its calculations. These operations are detailed in Appendix B. These operations produce a system of square-root information equations applicable at time t_{k+1} that is like the system in Eq. (30).

If any measurements occur in the IMU sample interval from time t_k to time t_{k+1}, then this filter performs a combined dynamic propagation and measurement update. Without loss of generality, suppose that measurements

are available from all three authentication groups. If any group lacks measurements, then the corresponding matrices and vectors in the following description have zero rows in them. The measurement update calculations require *a priori* estimates of the GPS measurement vectors for the three groups. Let them be:

$$\begin{aligned}\bar{\boldsymbol{y}}^A_{k+1} &= \boldsymbol{h}^A_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1}) \\ \bar{\boldsymbol{y}}^B_{k+1} &= \boldsymbol{h}^B_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1}) \\ \bar{\boldsymbol{y}}^C_{k+1} &= \boldsymbol{h}^C_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1})\end{aligned} \tag{33}$$

The functions $\boldsymbol{h}^A_k(,,)$, $\boldsymbol{h}^B_k(,,)$, and $\boldsymbol{h}^C_k(,,)$ are the GPS pseudorange and carrier Doppler shift measurement model functions for, respectively, authentication Groups A, B, and C.

The EKF-type measurement update calculations require linearized measurement models for the three groups. They take the form:

$$\begin{aligned}\boldsymbol{y}^A_{k+1} - \bar{\boldsymbol{y}}^A_{k+1} &= H^A_{xk}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + H^A_{k,wk}(\boldsymbol{w}_k - \hat{\boldsymbol{w}}_k) + H^A_{k,w(k+1)}(\boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1}) + \boldsymbol{\nu}^A_{k+1} \\ \boldsymbol{y}^B_{k+1} - \bar{\boldsymbol{y}}^B_{k+1} &= H^B_{xk}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + H^B_{k,wk}(\boldsymbol{w}_k - \hat{\boldsymbol{w}}_k) + H^B_{k,w(k+1)}(\boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1}) + \boldsymbol{\nu}^B_{k+1} \\ \boldsymbol{y}^C_{k+1} - \bar{\boldsymbol{y}}^C_{k+1} &= H^C_{xk}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) + H^C_{k,wk}(\boldsymbol{w}_k - \hat{\boldsymbol{w}}_k) + H^C_{k,w(k+1)}(\boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1}) + \boldsymbol{\nu}^C_{k+1}\end{aligned} \tag{34}$$

The vectors $\boldsymbol{y}^M_{k+1}$ for $M = A, B,$ and C are the actual measurement vectors for Groups A, B, and C. The matrices H^M_{xk}, $H^M_{k,wk}$, and $H^M_{k,w(k+1)}$ for $M = A, B,$ and C are the Jacobian first partial derivatives of the corresponding measurement model function $\boldsymbol{h}^M_k(,,)$ for $M = A, B,$ and C, similar to Eq. (17). The vectors $\boldsymbol{\nu}^M_{k+1}$ for $M = A, B,$ and C are measurement noise vectors. They are modeled as being zero-mean, Gaussian random vectors with measurement noise covariances $[R^M_{\nu\nu(k+1)}]^{-1}[R^M_{\nu\nu(k+1)}]^{-\mathrm{T}}$ for $M = A, B,$ and C.

The combined dynamic propagation and measurement update operations use the following measurement noise square root information equations of the three authentication groups:

$$\begin{aligned}R^A_{\nu\nu(k+1)}\boldsymbol{\nu}^A_{k+1} &= -\boldsymbol{v}^A_{\nu(k+1)} \\ R^B_{\nu\nu(k+1)}\boldsymbol{\nu}^B_{k+1} &= -\boldsymbol{v}^B_{\nu(k+1)} \\ R^C_{\nu\nu(k+1)}\boldsymbol{\nu}^C_{k+1} &= -\boldsymbol{v}^C_{\nu(k+1)}\end{aligned} \tag{35}$$

where $\boldsymbol{v}^M_{\nu(k+1)}$ for $M = A, B,$ and C are uncorrelated, zero-mean, identity-covariance Gaussian random error vectors.

The combined dynamic propagation and measurement update starts by solving the three equations in Eq. (34) for the noise vectors $\boldsymbol{\nu}^M_{k+1}$ for $M = A, B,$ and C, and it substitutes the results into the three square-root information equations in Eq. (35). Next, it solves the first equation in Eq. (21) for $\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$, and it substitutes the result into these three square-root information equations. Next, these three equations are appended to the bottom of the system of 9 square-root information equations that are modeled in Eq. (32). The resulting system of 12 square-root information equations is modeled by the following coefficient matrix and equation:

$$A_{meas} = \begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}^C_{wwk} - \hat{R}^C_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^C_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^C_{wxk}F_k^{-1} \\ \hat{R}^B_{wwk} - \hat{R}^B_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^B_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^B_{wxk}F_k^{-1} \\ \hat{R}^A_{wwk} - \hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{wxk}F_k^{-1} \\ \hat{R}_{wwk} - \hat{R}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{wxk}F_k^{-1} \\ -\hat{R}^C_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^C_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^C_{xxk}F_k^{-1} \\ -\hat{R}^B_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^B_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^B_{xxk}F_k^{-1} \\ -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{xxk}F_k^{-1} \\ -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{xxk}F_k^{-1} \\ R^C_{\nu\nu(k+1)}(H^C_{k,wk} - H^C_{xk}F_k^{-1}\Gamma_{k,k}) & R^C_{\nu\nu(k+1)}(H^C_{k,w(k+1)} - H^C_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^C_{\nu\nu(k+1)}H^C_{xk}F_k^{-1} \\ R^B_{\nu\nu(k+1)}(H^B_{k,wk} - H^B_{xk}F_k^{-1}\Gamma_{k,k}) & R^B_{\nu\nu(k+1)}(H^B_{k,w(k+1)} - H^B_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^B_{\nu\nu(k+1)}H^B_{xk}F_k^{-1} \\ R^A_{\nu\nu(k+1)}(H^A_{k,wk} - H^A_{xk}F_k^{-1}\Gamma_{k,k}) & R^A_{\nu\nu(k+1)}(H^A_{k,w(k+1)} - H^A_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^A_{\nu\nu(k+1)}H^A_{xk}F_k^{-1} \end{bmatrix} \tag{36}$$

$$A_{meas} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{w}_{k+1} - \bar{\boldsymbol{w}}_{k+1} \\ \boldsymbol{x}_{k+1} - \bar{\boldsymbol{x}}_{k+1} \end{bmatrix} = \begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^C_{wk} \\ \hat{\boldsymbol{z}}^B_{wk} \\ \hat{\boldsymbol{z}}^A_{wk} \\ 0 \\ \hat{\boldsymbol{z}}^C_{xk} \\ \hat{\boldsymbol{z}}^B_{xk} \\ \hat{\boldsymbol{z}}^A_{xk} \\ 0 \\ \boldsymbol{z}^C_{a(k+1)} \\ \boldsymbol{z}^B_{a(k+1)} \\ \boldsymbol{z}^A_{a(k+1)} \end{bmatrix} - \boldsymbol{\nu}_{meas} \tag{37}$$

where the vector $\boldsymbol{\nu}_{meas}$ is a zero-mean, identity-covariance Gaussian random vector that equals $\boldsymbol{\nu}_{prop}$ from Eq. (32) with $\boldsymbol{v}^A_{\nu(k+1)}$, $\boldsymbol{v}^B_{\nu(k+1)}$, and $\boldsymbol{v}^C_{\nu(k+1)}$ from Eq. (35) appended to the bottom of it, and where $\boldsymbol{z}^M_{a(k+1)} = R^M_{\nu\nu(k+1)}(\boldsymbol{y}^M_{k+1} - \bar{\boldsymbol{y}}^M_{k+1})$ for $M = A, B,$ and C.

The combined dynamic propagation and measurement update performs a sequence of orthonormal/upper-triangular factorizations of subsets of the rows of the coefficient matrix in this equation in order to complete its calculations. These operations are detailed in Appendix B. These operations produce a system of square-root information equations applicable at time t_{k+1} that is like the system in Eq. (30).

If the next authentication time for Group A has been reached at sample time t_{k+1}, then special operations are carried out to include the Group A information in the fully authenticated state estimate. These operations start with the combined square-root information equations at sample time t_{k+1} for the original authenticated estimate and for Group A:

$$\begin{bmatrix} \hat{R}^A_{ww(k+1)} & \hat{R}^A_{wx(k+1)} \\ 0 & \hat{R}^A_{xx(k+1)} \\ \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & \hat{R}_{xx(k+1)} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_{k+1} - \hat{\boldsymbol{w}}_{k+1} \\ \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{z}}^A_{w(k+1)} \\ \hat{\boldsymbol{z}}^A_{x(k+1)} \\ 0 \\ 0 \end{bmatrix} - \boldsymbol{\nu}_{auth} \tag{38}$$

where the vector $\boldsymbol{\nu}_{auth} = [\boldsymbol{v}^A_{\hat{\boldsymbol{w}}_k}; \boldsymbol{v}^A_{\hat{\boldsymbol{x}}_k}; \boldsymbol{v}_{\hat{\boldsymbol{w}}_k}; \boldsymbol{v}_{\hat{\boldsymbol{x}}_k}]$ is zero-mean, identity-covariance Gaussian random noise. An orthonormal/upper-triangular factorization of the left-hand coefficient matrix in this equation is performed in order to derive updates to the authenticated matrices $\hat{R}_{ww(k+1)}$, $\hat{R}_{wx(k+1)}$, and $\hat{R}_{xx(k+1)}$. Corresponding operations on the non-homogeneous terms on the right-hand side of this equation are used to develop Group-A authentication update increments to the current authenticated estimates $\hat{\boldsymbol{w}}_{k+1}$ and $\hat{\boldsymbol{x}}_{k+1}$. The details of these operations are given in Appendix B.

Note: All of these filters assume that all data eventually get authenticated. Should any data fail its authentication test, then the IMU-only filter would not undergo any authentication update. If there were hope that future GPS data would successfully pass its CHIMERA authentication test after a failure, then the various other filters or partial filters of the two techniques would need modifications in order to prepare them properly for this eventuality. The question of how to do this has been left as a subject of potential future study.

VI. Conclusions

The proposed CHIMERA system will be a useful tool for authenticating incoming GPS signals. However, this system creates a time delay between the time a GPS observable is received and the time it can be authenticated by CHIMERA. One proposed solution to the inherent navigation lag is to use a tightly coupled GPS/IMU navigation filter to dynamically propagate the state during this delay. The fast CHIMERA channel, with only a two second delay, is able to maintain nominal GPS navigation accuracy even when using low grade IMUs. The slow CHIMERA channel, with a 180 second delay, cannot maintain nominal navigation accuracy. RMS position errors on the order of 9.5 meters occur when using a navigation-grade IMU and slow CHIMERA authentication. The RMS errors can grow to thousands of meters when using a MEMS-grade IMU. Peak errors immediately before an authentication can be much larger than these RMS values.

A strategy for potentially improving on the slow-CHIMERA performance is to stagger the times that different GPS satellites have their signals authenticated. Results showed that this method improves the navigation accuracy when compared to non-staggered results. Future work will need to be done in order to determine if an optimal grouping of GPS satellites can further improve navigation accuracy.

Multiple filter architectures can be derived to handle the time delay produced by CHIMERA authentication and authentication staggering. This paper develops two alternate architectures that can be used and briefly discusses their computational benefits. Overall, a tightly coupled GPS/IMU system can be used to aid navigation in the presence of CHIMERA authentication delays.

Appendix A
Multiple Authentication Filter Equations

This section shows in detail the full equations for this architecture assuming 2 authentication groups.

Filter 1:

Filter 1 works with matrices and vectors that bear a superscript $()^1$ in order to designate its estimates and square-root information matrices. For example, its *a posteriori* state and process noise estimates at sample time t_k are, respectively, $\hat{\boldsymbol{x}}^1_k$ and $\hat{\boldsymbol{w}}^1_k$, and its *a priori* state estimate at sample time t_{k+1} is $\bar{\boldsymbol{x}}^1_{k+1}$. Its operations to transition from sample time t_k to sample time t_{k+1} follow those of Eqs. (20) and (24)-(29), except that the last lines of Eqs. (24) and (25) are omitted because there are no measurements associated with this fully authenticated filter. Therefore, the increments computed in Eqs. (26) and (27) are zero, and the *a priori* and *a posteriori* estimates at sample time t_{k+1} are identical in Eqs. (28) and (29).

Filter 1's calculations start by propagating the nonlinear dynamics to get the *a priori* estimate at sample time t_{k+1}:

$$\bar{\boldsymbol{x}}^1_{k+1} = \boldsymbol{f}_k(\hat{\boldsymbol{x}}^1_k, \hat{\boldsymbol{w}}^1_k, \bar{\boldsymbol{w}}_{k+1}) \tag{39}$$

Next, Filter 1 forms the large block matrix on the right-hand side of the following equation, and it performs an orthonormal/upper-triangular factorization of it in order to produce the output matrices on the left-hand side of this equation:

$$T_k \begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ 0 & \hat{R}^1_{ww(k+1)} & \hat{R}^1_{wx(k+1)} \\ 0 & 0 & \hat{R}^1_{xx(k+1)} \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}^1_{wwk} - \hat{R}^1_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^1_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^1_{wxk}F_k^{-1} \\ -\hat{R}^1_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^1_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^1_{xxk}F_k^{-1} \end{bmatrix} \tag{40}$$

Finally, the *a priori* state and process noise estimates at sample time t_{k+1} become the *a posteriori* estimates:

$$\hat{\boldsymbol{x}}^1_{k+1} = \bar{\boldsymbol{x}}^1_{k+1} \tag{41}$$

$$\hat{\boldsymbol{w}}^1_{k+1} = \bar{\boldsymbol{w}}_{k+1} \tag{42}$$

Filter 3:

Before defining Filter 2, it is helpful first to define Filter 3. For the case of two authentication groups, Filter 3 is the full measurement case. The highest numbered filter will always correspond to the full measurement case.

If there is no measurement, then the operations are like those of Filter 1. Except that its vectors and matrices bear the superscript $()^3$ rather than $()^1$, its operations repeat those given in Eqs. (39)-(42) for Filter 1.

If there are any measurements from either of the 2 authentication groups, then it uses the same procedure that is shown in the SRIF implementation given in Eqs. (20) and (24)-(29).

First, the *a priori* state estimate at sample time t_{k+1} is approximated as:

$$\bar{\boldsymbol{x}}^3_{k+1} = \boldsymbol{f}_k(\hat{\boldsymbol{x}}^3_k, \hat{\boldsymbol{w}}^3_k, \bar{\boldsymbol{w}}_{k+1}) \tag{43}$$

Next, the combined dynamic propagation and measurement update calculations are implemented as follows:

The matrix on the right-hand side of the following equation is factorized using an orthonormal/upper-triangular factorization to produce the output matrices on the left-hand side.

$$T_k \begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ 0 & \hat{R}^3_{ww(k+1)} & \hat{R}^3_{wx(k+1)} \\ 0 & 0 & \hat{R}^3_{xx(k+1)} \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}^3_{wwk} - \hat{R}^3_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^3_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^3_{wxk}F_k^{-1} \\ -\hat{R}^3_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^3_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^3_{xxk}F_k^{-1} \\ R_{\nu\nu(k+1)}(H_{k,wk} - H_{xk}F_k^{-1}\Gamma_{k,k}) & R_{\nu\nu(k+1)}(H_{k,w(k+1)} - H_{xk}F_k^{-1}\Gamma_{k,k+1}) & R_{\nu\nu(k+1)}H_{xk}F_k^{-1} \end{bmatrix} \tag{44}$$

The non-homogeneous vector on the extreme right-hand side of the following equation is transformed to produce the non-homogeneous terms on the left-hand side.

$$\begin{bmatrix} \boldsymbol{z}^*_{wk} \\ \hat{\boldsymbol{z}}^3_{w(k+1)} \\ \Delta\hat{\boldsymbol{z}}^3_{x(k+1)} \\ \hat{\boldsymbol{z}}^3_{r(k+1)} \end{bmatrix} = T_k^T \begin{bmatrix} 0 \\ 0 \\ 0 \\ R_{\nu\nu(k+1)}(\boldsymbol{y}_{k+1} - \bar{\boldsymbol{y}}_{k+1}) \end{bmatrix} \tag{45}$$

where $\bar{\boldsymbol{y}}_{k+1} = \boldsymbol{h}_k(\hat{\boldsymbol{x}}^3_k, \hat{\boldsymbol{w}}^3_k, \bar{\boldsymbol{w}}_{k+1})$.

Increments to the state and process noise estimates are computed as follows:

$$\delta\boldsymbol{x}^3_{k+1} = (\hat{R}^3_{xx(k+1)})^{-1}\Delta\hat{\boldsymbol{z}}^3_{x(k+1)} \tag{46}$$

$$\delta\boldsymbol{w}^3_{k+1} = (\hat{R}^3_{ww(k+1)})^{-1}(\hat{\boldsymbol{z}}^3_{w(k+1)} - \hat{R}^3_{wx(k+1)}\delta\boldsymbol{x}^3_{k+1}) \tag{47}$$

Finally, the *a posteriori* Filter-3 state and process-noise estimates at time t_{k+1} are computed:

$$\hat{\boldsymbol{x}}^3_{k+1} = \bar{\boldsymbol{x}}^3_{k+1} + \delta\boldsymbol{x}^3_{k+1} \tag{48}$$

$$\hat{\boldsymbol{w}}^3_{k+1} = \bar{\boldsymbol{w}}_{k+1} + \delta\boldsymbol{w}^3_{k+1} \tag{49}$$

Filter 2:

The no measurement case operations for Filter 2 are like those of Filter 1. Its operations repeat those given in Eqs. (39)-(42) for Filter 1, except that its vectors and matrices bear the superscript $()^2$ rather than $()^1$.

If there is a measurement, then Filter 2 performs a combined dynamic propagation and measurement update, like Filter 3, except that it only uses the measurements which correspond to the authentication group with the nearest future authentication time. These operations are the same as those given for Filter 3 in Eqs. (43)-(49). There are two differences. First, Filter 2's vectors and matrices bear the superscript $()^2$ rather than $()^3$. Second, the measurement equations used in the last lines of the right-hand sides of Eqs. (44) and (45) are different. They are only the subset of measurements from the authentication group with the nearest future authentication time.

When the next authentication occurs, the following operations are carried out in order to pass the newly authenticated information from Filter 2 to Filter 1 and in order to prepare Filter 2 for working only with data from the alternate authentication group, which is the group that will have the next authentication time:

$$\hat{R}^1_{xx(k+1)} = \hat{R}^2_{xx(k+1)} \tag{50}$$

$$\hat{R}^2_{xx(k+1)} = \hat{R}^3_{xx(k+1)} \tag{51}$$

$$\hat{R}^1_{wx(k+1)} = \hat{R}^2_{wx(k+1)} \tag{52}$$

$$\hat{R}^2_{wx(k+1)} = \hat{R}^3_{wx(k+1)} \tag{53}$$

$$\hat{R}^1_{ww(k+1)} = \hat{R}^2_{ww(k+1)} \tag{54}$$
$$\hat{R}^2_{ww(k+1)} = \hat{R}^3_{ww(k+1)} \tag{55}$$

$$\hat{\boldsymbol{x}}^1_{k+1} = \hat{\boldsymbol{x}}^2_{k+1} \tag{56}$$
$$\hat{\boldsymbol{x}}^2_{k+1} = \hat{\boldsymbol{x}}^3_{k+1} \tag{57}$$

$$\hat{\boldsymbol{w}}^1_{k+1} = \hat{\boldsymbol{w}}^2_{k+1} \tag{58}$$
$$\hat{\boldsymbol{w}}^2_{k+1} = \hat{\boldsymbol{w}}^3_{k+1} \tag{59}$$

Note that Filter 3 changes none of its vectors or matrices due to an authentication. This is true because Filter 3 always uses all of the data. Essentially, it is a filter that operates as though all of the data are always authenticated.

For each additional authentication group, this approach would need an additional filter. Given N authentication groups, Filter 1 would be the one that used none of the data, Filter 2 would use only the data for the authentication group with the nearest future authentication time, Filter 3 would use only the data from the two authentication groups with the two nearest future authentication times, etc. Finally, Filter N+1 would be the one that used all the data from all of the authentication groups. The equations implemented by the N+1 filters would be analogous to the equations that have been described above.

APPENDIX B
MULTIPLE AUTHENTICATION PARTIAL FILTER EQUATIONS

This section presents in detail the full equations for this architecture assuming 3 authentication groups.

If there are no measurements from any authentication groups, then the following operations are implemented to transition from sample time t_k to sample time t_{k+1}:

The process begins by computing the following approximation of the *a priori* state estimate at sample time t_{k+1}:

$$\bar{\boldsymbol{x}}_{k+1} = \boldsymbol{f}_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1}) \tag{60}$$

Starting with the large block coefficient matrix in Eq. (31), the first, fifth, and ninth rows are used to form the matrix on the right-hand side of the following equation. An orthonormal/upper-triangular factorization of the resulting matrix is then performed to determine the output matrices on the left-hand side of the following equation:

$$T_k \begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & 0 & \hat{R}_{xx(k+1)} \end{bmatrix} = \begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}_{wwk} - \hat{R}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{wxk}F_k^{-1} \\ -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{xxk}F_k^{-1} \end{bmatrix} \tag{61}$$

The three rows of the upper-triangular matrix on the left-hand side of the preceding equation and used to replace the first, fifth, and ninth rows of the large block matrix in Eq. (31) to yield the following partially transformed matrix:

$$\begin{bmatrix} R^{*}_{wwk} & R^{*}_{w(k)w(k+1)} & R^{*}_{w(k)x(k+1)} \\ \hat{R}^{C}_{wwk} - \hat{R}^{C}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{C}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{C}_{wxk}F_k^{-1} \\ \hat{R}^{B}_{wwk} - \hat{R}^{B}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{B}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{B}_{wxk}F_k^{-1} \\ \hat{R}^{A}_{wwk} - \hat{R}^{A}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{A}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{A}_{wxk}F_k^{-1} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ -\hat{R}^{C}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{C}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{C}_{xxk}F_k^{-1} \\ -\hat{R}^{B}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{B}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{B}_{xxk}F_k^{-1} \\ -\hat{R}^{A}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{A}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{A}_{xxk}F_k^{-1} \\ 0 & 0 & \hat{R}_{xx(k+1)} \end{bmatrix} \tag{62}$$

Next, the first, fourth, and eighth rows of the preceding matrix are used to form the matrix on the right-hand side of the following equation. An orthonormal/upper-triangular factorization of the resulting matrix is then performed to determine the output matrices on the left-hand side of the following equation:

$$T_k^A \begin{bmatrix} R^{*A}_{wwk} & R^{*A}_{w(k)w(k+1)} & R^{*A}_{w(k)x(k+1)} \\ 0 & \hat{R}^{A}_{ww(k+1)} & \hat{R}^{A}_{wx(k+1)} \\ 0 & 0 & \hat{R}^{A}_{xx(k+1)} \end{bmatrix} = \begin{bmatrix} R^{*}_{wwk} & R^{*}_{w(k)w(k+1)} & R^{*}_{w(k)x(k+1)} \\ \hat{R}^{A}_{wwk} - \hat{R}^{A}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{A}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{A}_{wxk}F_k^{-1} \\ -\hat{R}^{A}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^{A}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^{A}_{xxk}F_k^{-1} \end{bmatrix} \tag{63}$$

The three rows of the upper-triangular matrix on the left-hand side of the preceding equation are used to replace the first, fourth, and eighth rows of the large block matrix in Eq. (62) to yield yet another partially transformed matrix. An equation like Eq. (63) for Group B is formed based on the first, third, and seventh rows of this new partially transformed matrix, and the result is yet another partially transformed matrix. Finally, another equation like Eq. (63) for Group C is formed based on the first, second, and sixth lines of this next partially transformed matrix, and its outputs are used to form the following final transformed matrix:

$$\begin{bmatrix} R^{*C}_{wwk} & R^{*C}_{w(k)w(k+1)} & R^{*C}_{w(k)x(k+1)} \\ 0 & \hat{R}^{C}_{ww(k+1)} & \hat{R}^{C}_{wx(k+1)} \\ 0 & \hat{R}^{B}_{ww(k+1)} & \hat{R}^{B}_{wx(k+1)} \\ 0 & \hat{R}^{A}_{ww(k+1)} & \hat{R}^{A}_{wx(k+1)} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & 0 & \hat{R}^{C}_{xx(k+1)} \\ 0 & 0 & \hat{R}^{B}_{xx(k+1)} \\ 0 & 0 & \hat{R}^{A}_{xx(k+1)} \\ 0 & 0 & \hat{R}_{xx(k+1)} \end{bmatrix}$$

A corresponding set of transformations are applied to the following non-homogeneous vector that has been taken from Eq. (32):

$$\begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^{C}_{wk} \\ \hat{\boldsymbol{z}}^{B}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{wk} \\ 0 \\ \hat{\boldsymbol{z}}^{C}_{xk} \\ \hat{\boldsymbol{z}}^{B}_{xk} \\ \hat{\boldsymbol{z}}^{A}_{xk} \\ 0 \end{bmatrix}$$

There is no need to transform this vector using the transpose of the T_k matrix because its first, fifth, and ninth rows are all zero. The first operation applies the following transformation to the first, fourth, and eighth rows of the preceding vector:

$$\begin{bmatrix} \boldsymbol{z}^{*A}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{w(k+1)} \\ \hat{\boldsymbol{z}}^{A}_{x(k+1)} \end{bmatrix} = (T^{A}_{k})^{T} \begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^{A}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{xk} \end{bmatrix}$$

Substitution of the left-hand side results of this partial transformation into the first, fourth, and eighth rows of the original large vector yields the following partially transformed non-homogeneous term:

$$\begin{bmatrix} \boldsymbol{z}^{*A}_{wk} \\ \hat{\boldsymbol{z}}^{C}_{wk} \\ \hat{\boldsymbol{z}}^{B}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{w(k+1)} \\ 0 \\ \hat{\boldsymbol{z}}^{C}_{xk} \\ \hat{\boldsymbol{z}}^{B}_{xk} \\ \hat{\boldsymbol{z}}^{A}_{x(k+1)} \\ 0 \end{bmatrix}$$

Next, a similar partial transformation of the non-homogeneous vector's first, third, and seventh rows is implemented for Group B. Afterwards, a similar partial transformation of the resulting vector's first, second, and sixth rows is implemented for Group C. The final result is the following transformed non-homogeneous vector:

$$\begin{bmatrix} \boldsymbol{z}^{*C}_{wk} \\ \hat{\boldsymbol{z}}^{C}_{w(k+1)} \\ \hat{\boldsymbol{z}}^{B}_{w(k+1)} \\ \hat{\boldsymbol{z}}^{A}_{w(k+1)} \\ 0 \\ \hat{\boldsymbol{z}}^{C}_{x(k+1)} \\ \hat{\boldsymbol{z}}^{B}_{x(k+1)} \\ \hat{\boldsymbol{z}}^{A}_{x(k+1)} \\ 0 \end{bmatrix}$$

If there are measurements, then the combined dynamic propagation and measurement update operations start like the simple dynamic propagation operations. They compute the following approximation of the *a priori* state estimate at sample time t_{k+1}:

$$\bar{\boldsymbol{x}}_{k+1} = \boldsymbol{f}_k(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{w}}_k, \bar{\boldsymbol{w}}_{k+1}) \tag{64}$$

Starting with the large block coefficient matrix in Eq. (36), the first, fifth, and ninth rows are used to form the matrix on the right-hand side of the following equation. An orthonormal/upper-triangular factorization of the resulting matrix is then performed to determine the output matrices on the left-hand side of the following equation:

$$T_k \begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & 0 & \hat{R}_{xx(k+1)} \end{bmatrix} = \begin{bmatrix} 0 & \bar{R}_{ww(k+1)} & 0 \\ \hat{R}_{wwk} - \hat{R}_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{wxk}F_k^{-1} \\ -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}_{xxk}F_k^{-1} \end{bmatrix} \tag{65}$$

The three rows of the upper-triangular matrix on the left-hand side of the preceding equation and used to replace the first, fifth, and ninth rows of the large block matrix in Eq. (36) to yield the following partially transformed matrix:

$$\begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ \hat{R}^C_{wwk} - \hat{R}^C_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^C_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^C_{wxk}F_k^{-1} \\ \hat{R}^B_{wwk} - \hat{R}^B_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^B_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^B_{wxk}F_k^{-1} \\ \hat{R}^A_{wwk} - \hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{wxk}F_k^{-1} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ -\hat{R}^C_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^C_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^C_{xxk}F_k^{-1} \\ -\hat{R}^B_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^B_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^B_{xxk}F_k^{-1} \\ -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{xxk}F_k^{-1} \\ 0 & 0 & \hat{R}_{xx(k+1)} \\ R^C_{\nu\nu(k+1)}(H^C_{k,wk} - H^C_{xk}F_k^{-1}\Gamma_{k,k}) & R^C_{\nu\nu(k+1)}(H^C_{k,w(k+1)} - H^C_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^C_{\nu\nu(k+1)}H^C_{xk}F_k^{-1} \\ R^B_{\nu\nu(k+1)}(H^B_{k,wk} - H^B_{xk}F_k^{-1}\Gamma_{k,k}) & R^B_{\nu\nu(k+1)}(H^B_{k,w(k+1)} - H^B_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^B_{\nu\nu(k+1)}H^B_{xk}F_k^{-1} \\ R^A_{\nu\nu(k+1)}(H^A_{k,wk} - H^A_{xk}F_k^{-1}\Gamma_{k,k}) & R^A_{\nu\nu(k+1)}(H^A_{k,w(k+1)} - H^A_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^A_{\nu\nu(k+1)}H^A_{xk}F_k^{-1} \end{bmatrix} \tag{66}$$

A partial transformation process is then implemented for each authentication group starting with Group A and ending with Group C. The process for Group A is shown below. It takes the first, fourth, eighth, and twelfth rows of the large block matrix in Eq. (66) and uses them to form the matrix on the right-hand side of the following equation. An orthonormal/upper-triangular factorization of the resulting matrix is then performed to determine the output matrices on the left-hand side of the following equation:

$$T^A_k \begin{bmatrix} R^{*A}_{wwk} & R^{*A}_{w(k)w(k+1)} & R^{*A}_{w(k)x(k+1)} \\ 0 & \hat{R}^A_{ww(k+1)} & \hat{R}^A_{wx(k+1)} \\ 0 & 0 & \hat{R}^A_{xx(k+1)} \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} R^*_{wwk} & R^*_{w(k)w(k+1)} & R^*_{w(k)x(k+1)} \\ \hat{R}^A_{wwk} - \hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{wxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{wxk}F_k^{-1} \\ -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k} & -\hat{R}^A_{xxk}F_k^{-1}\Gamma_{k,k+1} & \hat{R}^A_{xxk}F_k^{-1} \\ R^A_{\nu\nu(k+1)}(H^A_{k,wk} - H^A_{xk}F_k^{-1}\Gamma_{k,k}) & R^A_{\nu\nu(k+1)}(H^A_{k,w(k+1)} - H^A_{xk}F_k^{-1}\Gamma_{k,k+1}) & R^A_{\nu\nu(k+1)}H^A_{xk}F_k^{-1} \end{bmatrix} \tag{67}$$

The four rows of the upper-triangular matrix on the left-hand side of the preceding equation are used to replace the first, fourth, eighth, and twelfth rows of the large block matrix in Eq. (66) to yield yet another partially transformed matrix. An equation like Eq. (67) for Group B is formed based on the first, third, seventh, and eleventh rows of this new partially transformed matrix, and the result is yet another partially transformed matrix. Finally, another equation like Eq. (67) for Group C is formed based on the first, second, sixth, and tenth rows of this next partially transformed matrix, and its outputs are used to form the following final transformed matrix:

$$\begin{bmatrix} R^{*C}_{wwk} & R^{*C}_{w(k)w(k+1)} & R^{*C}_{w(k)x(k+1)} \\ 0 & \hat{R}^{C}_{ww(k+1)} & \hat{R}^{C}_{wx(k+1)} \\ 0 & \hat{R}^{B}_{ww(k+1)} & \hat{R}^{B}_{wx(k+1)} \\ 0 & \hat{R}^{A}_{ww(k+1)} & \hat{R}^{A}_{wx(k+1)} \\ 0 & \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & 0 & \hat{R}^{C}_{xx(k+1)} \\ 0 & 0 & \hat{R}^{B}_{xx(k+1)} \\ 0 & 0 & \hat{R}^{A}_{xx(k+1)} \\ 0 & 0 & \hat{R}_{xx(k+1)} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{68}$$

Various non-homogeneous terms also need to be transformed. The needed operations start with the following non-homogeneous vector from Eq. (37):

$$\begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^{C}_{wk} \\ \hat{\boldsymbol{z}}^{B}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{wk} \\ 0 \\ \hat{\boldsymbol{z}}^{C}_{xk} \\ \hat{\boldsymbol{z}}^{B}_{xk} \\ \hat{\boldsymbol{z}}^{A}_{xk} \\ 0 \\ \boldsymbol{z}^{C}_{a(k+1)} \\ \boldsymbol{z}^{B}_{a(k+1)} \\ \boldsymbol{z}^{A}_{a(k+1)} \end{bmatrix}$$

There is no need to transform this vector using the transpose of the T_k matrix because its first, fifth, and ninth rows are all zero. The first operation applies the following transformation to the first, fourth, eighth, and twelfth rows of the preceding vector:

$$\begin{bmatrix} \boldsymbol{z}^{*A}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{w(k+1)} \\ \hat{\boldsymbol{z}}^{A}_{x(k+1)} \\ \hat{\boldsymbol{z}}^{A}_{r(k+1)} \end{bmatrix} = (T^{A}_{k})^{T} \begin{bmatrix} 0 \\ \hat{\boldsymbol{z}}^{A}_{wk} \\ \hat{\boldsymbol{z}}^{A}_{xk} \\ \boldsymbol{z}^{A}_{a(k+1)} \end{bmatrix}$$

Substitution of the results on the left-hand side of the preceding equation into the first, fourth, eighth, and twelfth rows of the original large non-homogeneous vector yields the following partially transformed vector:

$$\begin{bmatrix} \boldsymbol{z}_{wk}^{*A} \\ \hat{\boldsymbol{z}}_{wk}^{C} \\ \hat{\boldsymbol{z}}_{wk}^{B} \\ \hat{\boldsymbol{z}}_{w(k+1)}^{A} \\ 0 \\ \hat{\boldsymbol{z}}_{xk}^{C} \\ \hat{\boldsymbol{z}}_{xk}^{B} \\ \hat{\boldsymbol{z}}_{x(k+1)}^{A} \\ 0 \\ \boldsymbol{z}_{a(k+1)}^{C} \\ \boldsymbol{z}_{a(k+1)}^{B} \\ \hat{\boldsymbol{z}}_{r(k+1)}^{A} \end{bmatrix}$$

A similar transformation is applied for Group B to the first, third, seventh, and eleventh rows of the preceding non-homogeneous vector to produce yet another partially transformed non-homogeneous vector. Finally, a similar transformation is applied for Group C to the first, second, sixth, and tenth rows of the result in order to produce the final transformed non-homogeneous vector:

$$\begin{bmatrix} \boldsymbol{z}_{wk}^{*C} \\ \hat{\boldsymbol{z}}_{w(k+1)}^{C} \\ \hat{\boldsymbol{z}}_{w(k+1)}^{B} \\ \hat{\boldsymbol{z}}_{w(k+1)}^{A} \\ 0 \\ \hat{\boldsymbol{z}}_{x(k+1)}^{C} \\ \hat{\boldsymbol{z}}_{x(k+1)}^{B} \\ \hat{\boldsymbol{z}}_{x(k+1)}^{A} \\ 0 \\ \hat{\boldsymbol{z}}_{r(k+1)}^{C} \\ \hat{\boldsymbol{z}}_{r(k+1)}^{B} \\ \hat{\boldsymbol{z}}_{r(k+1)}^{A} \end{bmatrix} \tag{69}$$

Note that the final three vectors in this expression are residual error vectors.

After all of these calculations have been completed, and regardless of whether or not measurements have been processed, the following assignments are made:

$$\hat{\boldsymbol{x}}_{k+1} = \bar{\boldsymbol{x}}_{k+1}$$
$$\hat{\boldsymbol{w}}_{k+1} = \bar{\boldsymbol{w}}_{k+1}$$

and the last three rows, the residual error rows, are dropped from the large transformed coefficient matrix in Eq. (68) and from the large transformed non-homogeneous vector in Eq. (69).

Next, a query is made as to whether authentication Group A has received a new authentication for all of its data up through sample time t_{k+1}. If no authentication has occurred, then the algorithm proceeds to the next sample interval.

If Group A's GPS data have received an authentication up through sample time t_{k+1}, then the following operations are needed in order to incorporate that information into the trusted state estimate. The second and third columns of the fourth, fifth, eighth, and ninth rows of the matrix in Eq. (68) are used to form the matrix on the right-hand side of the following equation. An orthonormal/upper-triangular factorization of the resulting matrix is then performed to determine the output matrices on the left-hand side of the following equation:

$$T^A_{new(k)} \begin{bmatrix} \hat{R}^{auth}_{ww(k+1)} & \hat{R}^{auth}_{wx(k+1)} \\ 0 & \hat{R}^{auth}_{xx(k+1)} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \hat{R}^{A}_{ww(k+1)} & \hat{R}^{A}_{wx(k+1)} \\ 0 & \hat{R}^{A}_{xx(k+1)} \\ \hat{R}_{ww(k+1)} & \hat{R}_{wx(k+1)} \\ 0 & \hat{R}_{xx(k+1)} \end{bmatrix}$$

Next, the fourth, fifth, eighth, and ninth rows of the non-homogeneous vector in Eq. (69) are used to form the vector on the far right-hand side of the equation below, and the resulting vector is transformed as shown below to produce the outputs on the left-hand side of this equation:

$$\begin{bmatrix} \delta\hat{\boldsymbol{z}}^{auth}_{w(k+1)} \\ \delta\hat{\boldsymbol{z}}^{auth}_{x(k+1)} \\ \hat{\boldsymbol{z}}^{auth}_{r(k+1)} \end{bmatrix} = (T^A_{new(k)})^T \begin{bmatrix} \hat{\boldsymbol{z}}^{A}_{w(k+1)} \\ \hat{\boldsymbol{z}}^{A}_{x(k+1)} \\ 0 \\ 0 \end{bmatrix}$$

Note that the last entry of the vector on the left-hand side of this equation is an authentication residual vector.

Next, the following operations compute updates of the authenticated state and process noise estimates at sample time t_{k+1}:

$$\hat{\boldsymbol{x}}^{auth}_{k+1} = \hat{\boldsymbol{x}}_{k+1} + (\hat{R}^{auth}_{xx(k+1)})^{-1}\delta\hat{\boldsymbol{z}}^{auth}_{x(k+1)} \tag{70}$$

$$\hat{\boldsymbol{w}}^{auth}_{k+1} = \hat{\boldsymbol{w}}_{k+1} + (\hat{R}^{auth}_{ww(k+1)})^{-1}[\delta\hat{\boldsymbol{z}}^{auth}_{w(k+1)} - \hat{R}^{auth}_{wx(k+1)}(\hat{\boldsymbol{x}}^{auth}_{k+1} - \hat{\boldsymbol{x}}_{k+1})] \tag{71}$$

It is necessary to update the remaining non-homogeneous terms in the system's square-root information equations in order to account for the changes to the state and process noise estimates given in Eqs. (70) and (71). For the Group B equations, the needed modifications to the non-homogeneous terms are:

$$\hat{\boldsymbol{z}}^{B(new)}_{x(k+1)} = \hat{\boldsymbol{z}}^{B}_{x(k+1)} + \hat{R}^{B}_{xx(k+1)}(\hat{\boldsymbol{x}}_{k+1} - \hat{\boldsymbol{x}}^{auth}_{k+1}) \tag{72}$$

$$\hat{\boldsymbol{z}}^{B(new)}_{w(k+1)} = \hat{\boldsymbol{z}}^{B}_{w(k+1)} + \hat{R}^{B}_{ww(k+1)}(\hat{\boldsymbol{w}}_{k+1} - \hat{\boldsymbol{w}}^{auth}_{k+1}) + \hat{R}^{B}_{wx(k+1)}(\hat{\boldsymbol{x}}_{k+1} - \hat{\boldsymbol{x}}^{auth}_{k+1}) \tag{73}$$

The equations for the modifications to the Group C non-homogeneous terms are almost identical, except that $()^C$ superscripts replace $()^B$ superscripts.

Finally, a switch of authentication staggering groups must be implemented so that the new Group A is the one with the next authentication time. This will be the authentication group that had been labeled Group B. Similarly, the new Group B must be the one with the second future authentication time. This will be the authentication group that had been labeled Group C. The new Group C must be the one whose next authentication time lies furthest in the future. This will be the authentication group that had been labeled Group A. This latter group will have no measurement information in it yet because all of its information will have been incorporated into the fully authenticated filter that bears no group label.

The following matrix and vector substitutions are required in order to maintain consistency with the foregoing changes to the definitions of the three authentication groups:

$$\hat{R}^{A}_{xx(k+1)} = \hat{R}^{B}_{xx(k+1)}$$
$$\hat{R}^{B}_{xx(k+1)} = \hat{R}^{C}_{xx(k+1)}$$
$$\hat{R}^{C}_{xx(k+1)} = \mathbf{0}$$

$$\hat{\boldsymbol{z}}^{A}_{x(k+1)} = \hat{\boldsymbol{z}}^{B(new)}_{x(k+1)}$$
$$\hat{\boldsymbol{z}}^{B}_{x(k+1)} = \hat{\boldsymbol{z}}^{C(new)}_{x(k+1)}$$
$$\hat{\boldsymbol{z}}^{C}_{x(k+1)} = \mathbf{0}$$

$$\hat{R}^{A}_{wx(k+1)} = \hat{R}^{B}_{wx(k+1)}$$
$$\hat{R}^{B}_{wx(k+1)} = \hat{R}^{C}_{wx(k+1)}$$
$$\hat{R}^{C}_{wx(k+1)} = \mathbf{0}$$

$$\hat{R}^{A}_{ww(k+1)} = \hat{R}^{B}_{ww(k+1)}$$
$$\hat{R}^{B}_{ww(k+1)} = \hat{R}^{C}_{ww(k+1)}$$
$$\hat{R}^{C}_{ww(k+1)} = \mathbf{0}$$

$$\hat{\boldsymbol{z}}^{A}_{w(k+1)} = \hat{\boldsymbol{z}}^{B(new)}_{w(k+1)}$$
$$\hat{\boldsymbol{z}}^{B}_{w(k+1)} = \hat{\boldsymbol{z}}^{C(new)}_{w(k+1)}$$
$$\hat{\boldsymbol{z}}^{C}_{w(k+1)} = \mathbf{0}$$

$$\hat{\boldsymbol{w}}_{k+1} = \hat{\boldsymbol{w}}^{auth}_{k+1}$$
$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}^{auth}_{k+1}$$
$$\hat{R}_{ww(k+1)} = \hat{R}^{auth}_{ww(k+1)}$$
$$\hat{R}_{wx(k+1)} = \hat{R}^{auth}_{wx(k+1)}$$
$$\hat{R}_{xx(k+1)} = \hat{R}^{auth}_{xx(k+1)}$$

ACKNOWLEDGMENT

This work has been supported by the Air Force Research Laboratory through grant no. FA9453-19-1-0083. Joanna C. Hinks is the technical monitor.

REFERENCES

[1] A. Kerns, K. Wesson, and T. Humphreys, "A Blueprint for Civil GPS Navigation Message Authentication," in *Proc. IEEE/ION Plans Monterey, CA*, 2014, p. 262–269.

[2] J. Anderson, K. Carroll, N. DeVilbiss, J. Gillis, J. Hinks, B. O'Hanlon, J. Rushanan, L. Scott, and R. Yazdi, "Chips-Message Robust Authentication (Chimera) for GPS Civilian Signals," in *ION GNSS+ Portland, OR*, 2017, p. 2388–2416.

[3] E. Lefferts, F. Markley, and M. Shuster, "Kalman Filtering for Spacecraft Attitude Estimation," in *Journal of Guidance, Control, and Dynamics, Vol. 5, No. 5*, 1982, pp. 417–429.

[4] M. Mueller, M. Hehn, and R. D'Andrea, "Covariance Correction Step for Kalman Filtering with an Attitude," in *Journal of Guidance, Control, and Dynamics, Vol. 40, No. 9*, 2017, pp. 2301–2306.

[5] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: J. Wiley & Sons, 2001.

[6] G. Bierman, *Factorization Methods for Discrete Sequential Estimation*. New York: Academic Press, 1977.

[7] L. Scott, "Anti-Spoofing and Authenticated Signal Architectures for Civil Navigation Systems," in *Proc. ION GPS/GNSS Portland, OR*, 2003, pp. 1543–1552.

[8] T. Humphreys, B. Ledvina, M. Psiaki, B. O'Hanlon, and P. Kintner, Jr., "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer," in *Proceedings of the ION GNSS 2008*. Savannah, GA: ION, Sept. 2008, pp. 2314–2325.

[9] J. Farrell, *GNSS Aided Navigation & Tracking Inertially Augmented or Autonomous*. Baltimore: American Literary Press, 2007.

REPROCESSING OF PAST DATA

An algorithm can be developed that processes newly authenticated GPS data from past samples using partial Square-Root Information Filter (SRIF) calculations. These calculations use the standard linearizations of an Extended Kalman Filter (EKF). They use the same system models and notation as are used in [1].

Input Data for Reprocessing Calculations

Prior to the processing of past data that have been recently authenticated, suppose that the following three types of information are available: First, there is a coupled pair of square-root information equations that characterize the estimates of the sample-k process-noise vector $\boldsymbol{w}_k$ and state vector $\boldsymbol{x}_k$. They are:

$$\begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \end{bmatrix} = - \begin{bmatrix} \boldsymbol{v}_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}_{\hat{\boldsymbol{x}}_k} \end{bmatrix} \tag{1}$$

where $\hat{\boldsymbol{w}}_k$ is the *a posteriori* estimate of $\boldsymbol{w}_k$, $\hat{\boldsymbol{x}}_k$ is the *a posteriori* estimate of $\boldsymbol{x}_k$, $\boldsymbol{v}_{\hat{\boldsymbol{w}}_k}$ and $\boldsymbol{v}_{\hat{\boldsymbol{x}}_k}$ are uncorrelated, zero-mean, identity-covariance Gaussian random noise vectors, and $\hat{R}_{wwk}$, $\hat{R}_{wxk}$, and $\hat{R}_{xxk}$ are square-root information matrices. Sample k is the current sample of the Kalman filter, but these estimates only incorporate the GPS information that has been previously authenticated. There remain measurements at previous samples and at the present sample that have been authenticated recently, but they have not yet been incorporated into these filter estimates.

The vectors $\hat{\boldsymbol{w}}_k$ and $\hat{\boldsymbol{x}}_k$ and the matrices $\hat{R}_{wwk}$, $\hat{R}_{wxk}$, and $\hat{R}_{xxk}$ will have been determined by one of two means. Initially, before any data have been authenticated, they will have been determined via the IMU-only filtering pass that is described in [1]. If previous reprocessing due to authentication has occurred, as happens in the case of staggered authentications, then these vectors and matrices will have been determined during the previous authentication reprocessing, as described below.

The joint estimation error covariance for $\boldsymbol{w}_k$ and $\boldsymbol{x}_k$ is

$$P_{wxk} = E\{ \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k - \hat{\boldsymbol{w}}_k \\ \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \end{bmatrix}^{\mathrm{T}} \} = \begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \end{bmatrix}^{-1} \begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \end{bmatrix}^{-\mathrm{T}} \tag{2}$$

where the notation $()^{-\mathrm{T}}$ indicates the inverse of the transpose of the matrix in question. This covariance relationship is consistent with standard SRIF techniques.

The second piece of information available to the reprocessing algorithm is a set of smoothed square-root information equations for the past process-noise vectors $\boldsymbol{w}_j$ for $j = (k{-}m), \ldots, (k{-}1)$. The index decrement m equals the number of newly authenticated measurements, which are $\boldsymbol{y}_{j+1}$ for $j = (k{-}m), \ldots, (k{-}1)$. The smoothed square-root information equations are

$$R^*_{wwj}\boldsymbol{w}_j + R^*_{w(j)w(j+1)}\boldsymbol{w}_{j+1} + R^*_{w(j)x(j+1)}\boldsymbol{x}_{j+1} = \boldsymbol{z}^*_{wj} - \boldsymbol{v}_{\hat{\boldsymbol{w}}^*_j} \quad \text{for } j = (k{-}m), \ldots, (k{-}1) \tag{3}$$

The matrices R^*_{wwj}, $R^*_{w(j)w(j+1)}$, and $R^*_{w(j)x(j+1)}$ are 1-sample smoothed square-root information matrices, and $\boldsymbol{z}^*_{wj}$ is a corresponding known non-homogeneous term. The vector $\boldsymbol{v}_{\hat{\boldsymbol{w}}^*_j}$ is a zero-mean, identity-covariance Gaussian random vector.

Similar to the quantities in Eq. (1), these matrices and this vector will have been determined by one of two means. Initially, before any data have been authenticated, they will have been determined from the results of the IMU-only filtering pass that is described in [1]. The matrices R^*_{wwj},

$R^*_{w(j)w(j+1)}$, and $R^*_{w(j)x(j+1)}$ are direct outputs of that pass. The vector $\boldsymbol{z}^*_{wj}$ is computed as:

$$\boldsymbol{z}^*_{wj} = \boldsymbol{z}^{*ep}_{wj} + R^*_{wwj}\hat{\boldsymbol{w}}_j + R^*_{w(j)w(j+1)}\bar{\boldsymbol{w}}_{j+1} + R^*_{w(j)x(j+1)}\bar{\boldsymbol{x}}_{j+1} \tag{4}$$

where $\boldsymbol{z}^{*ep}_{wj}$ is the $\boldsymbol{z}^*_{wj}$ vector that gets calculated in the first line of Eq. (25) of [1]. It is given the extra superscript $()^{ep}$ here in order to distinguish it from the new $\boldsymbol{z}^*_{wj}$ that gets computed here. The vectors $\hat{\boldsymbol{w}}_j$ and $\bar{\boldsymbol{x}}_{j+1}$ are estimates that get generated during the IMU-only filtering pass of [1], and the vector $\bar{\boldsymbol{w}}_{j+1}$ is an *a priori* estimate which is available during that pass. If previous reprocessing due to authentication has occurred, then these matrices and this vector will have been determined during the previous authentication reprocessing, as described below.

The third piece of available information is the following set of linearized dynamic propagation equations for $j = (k-m), \ldots, (k-1)$:

$$\boldsymbol{x}_{j+1} = F_j\boldsymbol{x}_j + \Gamma_{j,j}\boldsymbol{w}_j + \Gamma_{j,j+1}\boldsymbol{w}_{j+1} + \breve{\boldsymbol{f}}_j \quad \text{for } j = (k-m), \ldots, (k-1) \tag{5}$$

This model is a linearization of the following nonlinear dynamics propagation model from Eq. (8) of [1]:

$$\boldsymbol{x}_{j+1} = \boldsymbol{f}_j(\boldsymbol{x}_j, \boldsymbol{w}_j, \boldsymbol{w}_{j+1}) \tag{6}$$

The three matrices in Eq. (5) are Jacobian matrices of the nonlinear function on the right-hand side of Eq. (6):

$$\begin{aligned} F_j &= \left.\frac{\partial \boldsymbol{f}_j}{\partial \boldsymbol{x}_j}\right|_{(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1})} \\ \Gamma_{j,j} &= \left.\frac{\partial \boldsymbol{f}_j}{\partial \boldsymbol{w}_j}\right|_{(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1})} \\ \Gamma_{j,j+1} &= \left.\frac{\partial \boldsymbol{f}_j}{\partial \boldsymbol{w}_{j+1}}\right|_{(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1})} \end{aligned} \tag{7}$$

where the vectors $\hat{\boldsymbol{x}}_j$ and $\hat{\boldsymbol{w}}_j$ are estimates that get generated during the IMU-only filtering pass of [1], and the vector $\bar{\boldsymbol{w}}_{j+1}$ is an *a priori* estimate which is available during that pass. The non-homogeneous vector $\breve{\boldsymbol{f}}_j$ in Eq. (5) is generated from the results of the Ref. [1] IMU-only filtering pass:

$$\breve{\boldsymbol{f}}_j = \bar{\boldsymbol{x}}_{j+1} - F_j\hat{\boldsymbol{x}}_j - \Gamma_{j,j}\hat{\boldsymbol{w}}_j - \Gamma_{j,j+1}\bar{\boldsymbol{w}}_{j+1} \tag{8}$$

where

$$\bar{\boldsymbol{x}}_{j+1} = \boldsymbol{f}_j(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1}) \tag{9}$$

from that same IMU-only forward filtering pass.

Nonlinear and Linearized Measurement Models of the Newly Authenticated Data

The reprocessing that incorporates the information from the newly authenticated measurements starts with the following nonlinear measurement model from Eq. (16) of [1]:

$$\boldsymbol{y}_{j+1} = \boldsymbol{h}_j(\boldsymbol{x}_j, \boldsymbol{w}_j, \boldsymbol{w}_{j+1}) + \boldsymbol{\nu}_{j+1} \tag{10}$$

where $\boldsymbol{\nu}_{j+1}$ is a zero-mean Gaussian random measurement noise vector. Its square-root information matrix is $R_{\nu\nu(j+1)}$, and its corresponding square-root information equation is

$$R_{\nu\nu(j+1)}\boldsymbol{\nu}_{j+1} = -\boldsymbol{v}_{\boldsymbol{\nu}_{j+1}} \tag{11}$$

where $\boldsymbol{v}_{\boldsymbol{\nu}_{j+1}}$ is a zero-mean, identity-covariance Gaussian random vector. Therefore, the covariance matrix of the measurement noise vector $\boldsymbol{\nu}_{j+1}$ is $R^{-1}_{\nu\nu(j+1)} R^{-\mathrm{T}}_{\nu\nu(j+1)}$.

The following linearized versions of the measurement models for all of the newly authenticated data are formed for use in the reprocessing calculations:

$$\boldsymbol{y}_{j+1} = H_{xj}\boldsymbol{x}_j + H_{j,wj}\boldsymbol{w}_j + H_{j,w(j+1)}\boldsymbol{w}_{j+1} + \breve{\boldsymbol{h}}_j + \boldsymbol{\nu}_{j+1} \quad \text{for } j = (k-m), \ldots, (k-1) \tag{12}$$

The three matrices on the right-hand side of Eq. (12) are Jacobian matrices of the nonlinear function on the right-hand side of Eq. (10):

$$\begin{aligned} H_{xj} &= \left.\frac{\partial \boldsymbol{h}_j}{\partial \boldsymbol{x}_j}\right|_{(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1})} \\ H_{j,wj} &= \left.\frac{\partial \boldsymbol{h}_j}{\partial \boldsymbol{w}_j}\right|_{(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1})} \\ H_{j,w(j+1)} &= \left.\frac{\partial \boldsymbol{h}_j}{\partial \boldsymbol{w}_{j+1}}\right|_{(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1})} \end{aligned} \tag{13}$$

where the estimates at which these Jacobians are evaluated come from the IMU-only pass of the filter in [1]. The non-homogeneous term in the linearized measurement model is determined via the calculation:

$$\breve{\boldsymbol{h}}_j = \boldsymbol{h}_j(\hat{\boldsymbol{x}}_j, \hat{\boldsymbol{w}}_j, \bar{\boldsymbol{w}}_{j+1}) - H_{xj}\hat{\boldsymbol{x}}_j - H_{j,wj}\hat{\boldsymbol{w}}_j - H_{j,w(j+1)}\bar{\boldsymbol{w}}_{j+1} \tag{14}$$

Storage of Cumulative Authenticated Information

The reprocessing algorithm needs to keep track of the information about $\boldsymbol{w}_j$ and $\boldsymbol{x}_j$ that is contained in the reprocessed data $\boldsymbol{y}_\ell$ for $\ell = (k-m+1), \ldots, j$. It stores this information in the following pair of coupled square-root information equations:

$$\begin{bmatrix} \tilde{R}_{wwj} & \tilde{R}_{wxj} \\ 0 & \tilde{R}_{xxj} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_j \\ \boldsymbol{x}_j \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{z}}_{wj} \\ \tilde{\boldsymbol{z}}_{xj} \end{bmatrix} - \begin{bmatrix} \tilde{\boldsymbol{v}}_{wj} \\ \tilde{\boldsymbol{v}}_{xj} \end{bmatrix} \tag{15}$$

where $\tilde{R}_{wwj}$, $\tilde{R}_{wxj}$, and $\tilde{R}_{xxj}$ are square-root information matrices, $\tilde{\boldsymbol{z}}_{wj}$ and $\tilde{\boldsymbol{z}}_{xj}$ are known non-homogeneous vectors, and $\tilde{\boldsymbol{v}}_{wj}$ and $\tilde{\boldsymbol{v}}_{xj}$ are uncorrelated, zero-mean, identity-covariance Gaussian random vectors.

The reprocessing algorithm starts with the following initialization of these quantities at sample $j = k - m$:

$$\begin{aligned} \tilde{R}_{wwj} &= 0 \\ \tilde{R}_{wxj} &= 0 \\ \tilde{R}_{xxj} &= 0 \\ \tilde{\boldsymbol{z}}_{wj} &= 0 \\ \tilde{\boldsymbol{z}}_{xj} &= 0 \end{aligned} \tag{16}$$

This initialization is consistent with the assumption that there is no new authenticated data from before measurement $\boldsymbol{y}_{k-m+1}$.

Recursive Reprocessing Operations

The reprocessing algorithm performs the following operations recursively for each sample starting from $j = k - m$ and working forward one sample at a time up through sample $j = k - 1$. The first operation of the recursion forms a coupled system of four square-root information equations by stacking the square-root information in Eq. (3), the coupled pair of square-root information equations in Eq. (15), and the square-root information equation in Eq. (11). Next, the measurement model in Eq. (12) is solved for $\boldsymbol{\nu}_{j+1}$, and the result is substituted into the system of square-root information equations. Lastly, the dynamics model in Eq. (5) is solved for $\boldsymbol{x}_j$, and this result is substituted into the system. The resulting system of square-root information equations takes the form:

$$\begin{bmatrix} R^*_{wwj} & R^*_{w(j)w(j+1)} & R^*_{w(j)x(j+1)} \\ (\tilde{R}_{wwj} - \tilde{R}_{wxj}F_j^{-1}\Gamma_{j,j}) & (-\tilde{R}_{wxj}F_j^{-1}\Gamma_{j,j+1}) & (\tilde{R}_{wxj}F_j^{-1}) \\ (-\tilde{R}_{xxj}F_j^{-1}\Gamma_{j,j}) & (-\tilde{R}_{xxj}F_j^{-1}\Gamma_{j,j+1}) & (\tilde{R}_{xxj}F_j^{-1}) \\ R_{\nu\nu(j+1)}(H_{j,wj} - H_{xj}F_j^{-1}\Gamma_{j,j}) & R_{\nu\nu(j+1)}(H_{j,w(j+1)} - H_{xj}F_j^{-1}\Gamma_{j,j+1}) & R_{\nu\nu(j+1)}(H_{xj}F_j^{-1}) \end{bmatrix} \times \begin{bmatrix} \boldsymbol{w}_j \\ \boldsymbol{w}_{j+1} \\ \boldsymbol{x}_{j+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{z}^*_{wj} \\ (\tilde{\boldsymbol{z}}_{wj} + \tilde{R}_{wxj}F_j^{-1}\breve{\boldsymbol{f}}_j) \\ (\tilde{\boldsymbol{z}}_{xj} + \tilde{R}_{xxj}F_j^{-1}\breve{\boldsymbol{f}}_j) \\ R_{\nu\nu(j+1)}(\boldsymbol{y}_{j+1} - \breve{\boldsymbol{h}}_j + H_{xj}F_j^{-1}\breve{\boldsymbol{f}}_j) \end{bmatrix} - \begin{bmatrix} \boldsymbol{v}_{\hat{\boldsymbol{w}}^*_j} \\ \tilde{\boldsymbol{v}}_{wj} \\ \tilde{\boldsymbol{v}}_{xj} \\ -\boldsymbol{v}_{\boldsymbol{\nu}_{j+1}} \end{bmatrix} \tag{17}$$

The algorithm performs an orthonormal/upper-triangular (QR) factorization of the large block matrix on the left-hand side of this equation. It is defined as follows:

$$\tilde{T}_j \begin{bmatrix} R^{*new}_{wwj} & R^{*new}_{w(j)w(j+1)} & R^{*new}_{w(j)x(j+1)} \\ 0 & \tilde{R}_{ww(j+1)} & \tilde{R}_{wx(j+1)} \\ 0 & 0 & \tilde{R}_{xx(j+1)} \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} R^*_{wwj} & R^*_{w(j)w(j+1)} & R^*_{w(j)x(j+1)} \\ (\tilde{R}_{wwj} - \tilde{R}_{wxj}F_j^{-1}\Gamma_{j,j}) & (-\tilde{R}_{wxj}F_j^{-1}\Gamma_{j,j+1}) & (\tilde{R}_{wxj}F_j^{-1}) \\ (-\tilde{R}_{xxj}F_j^{-1}\Gamma_{j,j}) & (-\tilde{R}_{xxj}F_j^{-1}\Gamma_{j,j+1}) & (\tilde{R}_{xxj}F_j^{-1}) \\ R_{\nu\nu(j+1)}(H_{j,wj} - H_{xj}F_j^{-1}\Gamma_{j,j}) & R_{\nu\nu(j+1)}(H_{j,w(j+1)} - H_{xj}F_j^{-1}\Gamma_{j,j+1}) & R_{\nu\nu(j+1)}(H_{xj}F_j^{-1}) \end{bmatrix} \tag{18}$$

where the large block matrix on the right-hand side of this equation constitutes the input to the QR factorization, and the matrices on the left-hand side constitute the outputs. The output matrix $\tilde{T}_j$ is orthonormal, the output matrices R^{*new}_{wwj}, $\tilde{R}_{ww(j+1)}$, and $\tilde{R}_{xx(j+1)}$ are square, upper-triangular square-root information matrices, and the output matrices $R^{*new}_{w(j)w(j+1)}$, $R^*_{w(j)x(j+1)}$, and $\tilde{R}_{wx(j+1)}$ are dense square-root information matrix components of appropriate dimensions.

Next, the algorithm performs the following transformation of the non-homogeneous term that is

the first term on the right-hand side of Eq. (17):

$$\begin{bmatrix} \boldsymbol{z}_{wj}^{*new} \\ \tilde{\boldsymbol{z}}_{w(j+1)} \\ \tilde{\boldsymbol{z}}_{x(j+1)} \\ \tilde{\boldsymbol{z}}_{r(j+1)} \end{bmatrix} = \tilde{T}_j^{\mathrm{T}} \begin{bmatrix} \boldsymbol{z}_{wj}^{*} \\ (\tilde{\boldsymbol{z}}_{wj} + \tilde{R}_{wxj} F_j^{-1} \breve{\boldsymbol{f}}_j) \\ (\tilde{\boldsymbol{z}}_{xj} + \tilde{R}_{xxj} F_j^{-1} \breve{\boldsymbol{f}}_j) \\ R_{\nu\nu(j+1)} (\boldsymbol{y}_{j+1} - \breve{\boldsymbol{h}}_j + H_{xj} F_j^{-1} \breve{\boldsymbol{f}}_j) \end{bmatrix} \tag{19}$$

The three new matrices from the top line of the left-hand side of Eq. (18), R_{wwj}^{*new}, $R_{w(j)w(j+1)}^{*new}$, and $R_{w(j)x(j+1)}^{*new}$ are used to replace the original matrices R_{wwj}^{*}, $R_{w(j)w(j+1)}^{*}$, and $R_{w(j)x(j+1)}^{*}$. Similarly, the new vector from the time line of the left-hand side of Eq. (19), $\boldsymbol{z}_{wj}^{*new}$, is used to replace $\boldsymbol{z}_{wj}^{*}$. These four replacements prepare the smoothed process-noise information equation for $\boldsymbol{w}_j$ in case a future authentication should require another reprocessing pass through sample j. This is a likely scenario in the case of staggered authentication measurement sets.

The three matrices $\tilde{R}_{ww(j+1)}$, $\tilde{R}_{wx(j+1)}$, and $\tilde{R}_{xx(j+1)}$ from the second and third lines of the left-hand side of Eq. (18) and the two vectors $\tilde{\boldsymbol{z}}_{w(j+1)}$ and $\tilde{\boldsymbol{z}}_{x(j+1)}$ from the second and third lines of the left-hand side of Eq. (19) constitute the information that is needed in order to recursively apply these operations to successive samples. The vector $\tilde{\boldsymbol{z}}_{r(j+1)}$ from the last line of the left-hand side of Eq. (19) is an authentication residuals vector. It gets discarded.

Fusion of Authenticated Past Data with Current Estimates

The reprocessing recursion terminates at the sample $j = k - 1$. Its final output is the following coupled system of two square-root information equations:

$$\begin{bmatrix} \tilde{R}_{wwk} & \tilde{R}_{wxk} \\ 0 & \tilde{R}_{xxk} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{x}_k \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{z}}_{wk} \\ \tilde{\boldsymbol{z}}_{xk} \end{bmatrix} - \begin{bmatrix} \tilde{\boldsymbol{v}}_{wk} \\ \tilde{\boldsymbol{v}}_{xk} \end{bmatrix} \tag{20}$$

The final authentication operation fuses this information with the information in the coupled *a posteriori* square-root information equations in Eq. (1). Combining the two sets of square-root information equations yields the system:

$$\begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \\ \tilde{R}_{wwk} & \tilde{R}_{wxk} \\ 0 & \tilde{R}_{xxk} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{x}_k \end{bmatrix} = \begin{bmatrix} (\hat{R}_{wwk} \hat{\boldsymbol{w}}_k + \hat{R}_{wxk} \hat{\boldsymbol{x}}_k) \\ (\hat{R}_{xxk} \hat{\boldsymbol{x}}_k) \\ \tilde{\boldsymbol{z}}_{wk} \\ \tilde{\boldsymbol{z}}_{xk} \end{bmatrix} - \begin{bmatrix} \boldsymbol{v}_{\hat{\boldsymbol{w}}_k} \\ \boldsymbol{v}_{\hat{\boldsymbol{x}}_k} \\ \tilde{\boldsymbol{v}}_{wk} \\ \tilde{\boldsymbol{v}}_{xk} \end{bmatrix} \tag{21}$$

The large coefficient matrix on the left-hand side of this equation gets QR-factorized as follows:

$$\breve{T}_k \begin{bmatrix} \hat{R}_{wwk}^{new} & \hat{R}_{wxk}^{new} \\ 0 & \hat{R}_{xxk}^{new} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \hat{R}_{wwk} & \hat{R}_{wxk} \\ 0 & \hat{R}_{xxk} \\ \tilde{R}_{wwk} & \tilde{R}_{wxk} \\ 0 & \tilde{R}_{xxk} \end{bmatrix} \tag{22}$$

The block matrix on the right-hand side of this equation is the input to the QR factorization, and the matrices on the left-hand side are the outputs. The matrix $\breve{T}_k$ is orthonormal. The square-root information matrices $\hat{R}_{wwk}^{new}$ and $\hat{R}_{xxk}^{new}$ are square and upper-triangular. The square-root information component $\hat{R}_{wxk}^{new}$ is a dense matrix of appropriate dimensions.

Next, the leading non-homogeneous vector on the right-hand side of Eq. (21) is transformed as follows:

$$\begin{bmatrix} \hat{\boldsymbol{z}}_{wk}^{new} \\ \hat{\boldsymbol{z}}_{xk}^{new} \\ \boldsymbol{z}_{rak} \\ \boldsymbol{z}_{rbk} \end{bmatrix} = \breve{T}_k^{\mathrm{T}} \begin{bmatrix} (\hat{R}_{wwk}\hat{\boldsymbol{w}}_k + \hat{R}_{wxk}\hat{\boldsymbol{x}}_k) \\ (\hat{R}_{xxk}\hat{\boldsymbol{x}}_k) \\ \tilde{\boldsymbol{z}}_{wk} \\ \tilde{\boldsymbol{z}}_{xk} \end{bmatrix} \tag{23}$$

Next, the new state and process-noise vector estimates are formed as follows:

$$\begin{aligned} \hat{\boldsymbol{x}}_k^{new} &= (\hat{R}_{xxk}^{new})^{-1}\hat{\boldsymbol{z}}_{xk}^{new} \\ \hat{\boldsymbol{w}}_k^{new} &= (\hat{R}_{wwk}^{new})^{-1}(\hat{\boldsymbol{z}}_{wk}^{new} - \hat{R}_{wxk}^{new}\hat{\boldsymbol{x}}_k^{new}) \end{aligned} \tag{24}$$

The lower two output vectors on the left-hand side of of Eq. (23), $\boldsymbol{z}_{rak}$ and $\boldsymbol{z}_{rbk}$, are residual error vectors of the authentication. They are discarded.

The final step of the reprocessing algorithm replaces the original *a posteriori* estimates $\hat{\boldsymbol{w}}_k$ and $\hat{\boldsymbol{x}}_k$ with their respective newly updated versions $\hat{\boldsymbol{w}}_k^{new}$ and $\hat{\boldsymbol{x}}_k^{new}$. Similarly, the original *a posteriori* square-root information matrices $\hat{R}_{wwk}$, $\hat{R}_{wxk}$, and $\hat{R}_{xxk}$ are replaced with their respective newly updated versions $\hat{R}_{wwk}^{new}$, $\hat{R}_{wxk}^{new}$, and $\hat{R}_{xxk}^{new}$.

REFERENCES

[1] M. Esswein and M. Psiaki, "GPS Spoofing Resilience via NMA/Watermarks Authentication and IMU Prediction," *Proceedings of the ION GNSS+ 2021*, St. Louis, MO, ION, Sept. 2021.

DISTRIBUTION LIST

DTIC/OCP
8725 John J. Kingman Rd, Suite 0944
Ft Belvoir, VA 22060-6218 1 cy

AFRL/RVIL
Kirtland AFB, NM 87117-5776 1 cy

Official Record Copy
AFRL/RVB/Dr. Joanna C.Hinks 1 cy

This page is intentionally left blank.