AFRL-RV-PS-
TR-2016-0158

EXTERNAL AIDING METHODS FOR IMU-BASED NAVIGATION

Kyle J. DeMars, et al.

University of Missouri System
Missouri University of Science and Technology
300 West 12 Street
Rolla, MO 65409-6506

26 November 2016

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 was cleared for public release by the PRS OPSEC Office and 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-2016-0158 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//SIGNED//	//SIGNED//
Stephen Forbes Program Manager, AFRL/RVBYE	Dr. Thomas R. Caudill, Acting Chief AFRL Battlespace Environment 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.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
26-11-2016	Final Report	27 Aug 2014 – 26 Aug 2016

4. TITLE AND SUBTITLE
External Aiding Methods for IMU-Based Navigation

5a. CONTRACT NUMBER
FA9453-14-1-0243

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62601F

6. AUTHOR(S)
Kyle J. DeMars, Christine L. Schmid, Samuel J. Haberberger, and Matthew J. Gualdoni

5d. PROJECT NUMBER
1010

5e. TASK NUMBER
PPM00009521

5f. WORK UNIT NUMBER
EF127203

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Missouri System
Missouri University of Science and Technology
300 West 12 Street
Rolla, MO 65409-6506

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/RVBYE

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-RV-PS-TR-2016-0158

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited. (OPS-17-13031 dtd 10 Mar 2017)

13. SUPPLEMENTARY NOTES

14. ABSTRACT
A primary aspect involved in the navigation of a wide variety of vehicles, such as ground vehicles and aircraft, is the use of inertial measurement units, which provide an internal measure of the linear acceleration and angular velocity of the vehicle in order to be used by a navigation system in the dead-reckoning of the vehicle's position and attitude. Due to the nature of sensors, the measurement of the linear acceleration and angular velocity are corrupted by several error sources, such as bias and noise, amongst others. Another key element involved in the navigation of vehicles is the use of external aiding via additional sensors, such as range-finders or star cameras, where the purpose of these sensors is to aid navigation by providing measurements relative to the external environment in which the vehicle is operating. In order to assess the coupled impact of these sensors, along with their inherent uncertainties, on the navigation accuracy for an arbitrary vehicle, a simulation and analysis tool is developed. The developed tool is capable of modeling different sensors in a variety of configurations and performing studies via Monte Carlo and linear covariance analyses to determine the achievable navigation accuracies of a prescribed sensor suite.

15. SUBJECT TERMS
Inertial Measurement Unit Modeling, IMU Modeling, Dead-Reckoning, Navigation, External Aiding, Monte Carlo Simulation, Linear Covariance Analysis

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Stephen Forbes
Unclassified	Unclassified	Unclassified	Unlimited	46	**19b. TELEPHONE NUMBER** *(include area code)*

This page is intentionally left blank.

TABLE OF CONTENTS

Section **Page**

LIST OF FIGURES

Figure **Page**

1.0 SUMMARY

A primary aspect involved in the navigation of a wide variety of vehicles, such as ground vehicles and aircraft, is the use of inertial measurement units. The purpose of the inertial measurement unit is to provide an internal measure of the linear acceleration and angular velocity of the vehicle in order to be used by a navigation system in the dead-reckoning of the vehicle's position and attitude. Due to the nature of sensors, the measurement of the linear acceleration and angular velocity are corrupted by several error sources, such as bias and noise, amongst others. Another key element involved in the navigation of vehicles is the use of external aiding via more conventional sensors, such as range-finders and star cameras. The purpose of these sensors is to aid navigation by providing measurements relative to the external environment in which the vehicle is operating. In order to assess the coupled impact of these sensors, along with their inherent uncertainties, on the navigation accuracy for an arbitrary vehicle, a simulation and analysis tool is developed that can model several inertial measurement unit mechanizations, model several sensors for the purpose of external aiding, simulate the error sources encountered in the acquisition of measurement data, emulate the navigation software, and perform a range of analyses via Monte Carlo simulation and linear covariance analysis. The developed tool is applied to a ground vehicle navigation scenario in order to assess the performance of different sensors and provide conclusions as to when each sensor should be used to achieve the best navigation performance.

2.0 INTRODUCTION

Inertial measurement unit (IMU) technology is used to drive a dominant element of navigation systems on vehicles ranging from submarines to ground vehicles, missiles, and spacecraft. The purpose of the IMU is to provide an internal measure of the changes in the linear and angular motions of the vehicle due to non-gravitational stimuli by using triads of single-axis accelerometers to measure the linear motion and gyros to measure the angular motion via measurement of the angular velocity. In doing so, the position, velocity, and attitude of the vehicle can be predicted from a prescribed starting condition along with a representation of the confidence in the prediction, i.e. an associated uncertainty.

Traditionally, IMU-based navigation is mechanized in one of two ways: gimbaled/gyro-stabilized or strapdown. In gimbaled mechanizations, the triad of accelerometers is placed on a "stable table" at the center of set of interconnected gimbals or at the center of a fluidic stabilization system. In the strapdown mechanization, the triad of accelerometers and gyros is fixed (strapped down) to the body of the vehicle, and there is no stable table. The different mechanizations effectively alter the way in which the triad orientation is handled: for gimbaled systems, mechanical tracking is used, whereas computational tracking is used for strapdown systems. Common to every system, the outputs of the single-axis sensors that make up the accelerometer and gyroscope are used in conjunction with the navigation system software to dead-reckon (i.e. using only the IMU) the state (position, velocity, and attitude) of the vehicle.

Another core element of vehicular navigation is the use of external sensor data, in conjunction with the IMU data, to improve knowledge of the position, velocity, and attitude of the vehicle. Data acquired from range finders, star trackers, and line-of-sight sensors, amongst others, can be used to provide refinements in the uncertainty with which the state of the vehicle can be determined.

Using the IMU to predict the position, velocity, and attitude of a vehicle inherently leads to uncertainty in the aforementioned localization variables. Even in the case where perfect initial conditions are known (i.e. there is no uncertainty in the position, velocity, and attitude), errors in the IMU parameterization will flow into the translational and rotational states, causing a growth

in the uncertainty on the prediction of the position, velocity, and attitude through a degradation of the associated confidence in these predictions. Using the external sensor data can mitigate this growth of uncertainty by adding information about the vehicle's state.

Employing external measurement data increases the system complexity and cost by introducing additional system components; however, if the system complexity growth is accompanied by improved performance of the navigation system, then operation of the vehicle can be considerably improved. This can lead to more reliable operation of autonomous or remotely operated vehicles and it can lead to more robust fault and failure monitoring for human operated vehicles. The trade between increased complexity and improved performance needs to be thoroughly analyzed and understood in order to assess whether or not a suite of external measurement senors is efficacious.

Analysis of IMU-driven, externally aided navigation systems can be accomplished in several ways. One of the most flexible approaches is to use techniques from Monte Carlo simulation and particle filtering. This approach allows for the utilization of highly complex systems in a black box configuration with minimal intrusion into the governing equations; however, this can be quite computationally complex and require significant computing resources. An alternative is to use linear covariance analysis to predict the mean and covariance that result from the implementation of a certain combination of an IMU and other external sensors. This method accounts for the errors in all of the parameters, but does require linearization to be performed in order to determine the resulting uncertainties. The benefit of linear covariance analysis over Monte Carlo analysis is that linear covariance analysis is computationally efficient and can be performed on many configurations of the internal/external sensor combinations very expediently.

In order to quantify the uncertainties involved with a specific navigation system that is comprised of internal and external sensors, SAIMUN (Simulation and Analysis of IMU-based Navigation) was developed. This tool utilizes a known truth trajectory, IMU parameterization of interest, prescribed suite of external aiding sensors, and initial conditions to calculate performance criteria. The tool makes use of Monte Carlo analysis and linear covariance analysis to propagate the position, velocity, and attitude uncertainty forward in time. Monte Carlo analysis is used to investigate the uncertainties for IMU-only navigation systems and to baseline performance in this case. Linear covariance analysis is used to compare to Monte Carlo analysis for IMU-only navigation and to perform externally aided navigation analysis as well.

SAIMUN was developed to consider two different IMU mechanizations: strapdown and space stabilized. The strapdown IMU mechanization relies on microelectromechanical systems (MEMS) technology for the accelerometer and gyroscope triads. The triads are stationary with respect to the IMU case; thus, the attitude is dead-reckoned numerically using the gyroscope data to provide the vehicle attitude. The space stabilized IMU mechanization relies on a mechanical platform to track an inertial reference frame. Additionally, SAIMUN was developed to consider three types of external measurements: range/range-rate measurements, stellar line-of-sight measurements, and bearing angles measurements. The objective of the tool is to provide an analysis of the feasible navigation accuracy given some combination of IMU and external sensors.

The remainder of this report is organized as follows. Section 3.1 provides an overview of the simulation environment that was developed. Section 3.2 details the modeling of the error sources which contribute to the IMU corruption and provides the overall models for the measured IMU outputs that are used in the strapdown and space stabilized implementations. Following is a detailed development of the dynamical models used for the two IMU configurations in Section 3.4. Section 3.5 describes the Monte Carlo analysis technique and its usefulness in validating other uncertainty analysis techniques. Section 3.6 outlines the three sensors used for external aiding including application and their dynamical models. The theory and method of implementation of the linear covariance analysis is presented in Section 3.7. Section 4.0 presents the results of a ground

vehicle navigation simulation in which two IMUs are considered and compared using the developed tool. Section 5.0 concludes the report with a discussion as to how the developed tool is used to select an appropriate IMU for the ground vehicle navigation simulation as well as a discussion on other capabilities that SAIMUN offers.

3.0 METHODS, ASSUMPTIONS, AND PROCEDURES

3.1 SAIMUN Overview

In order to analyze the effect of inertial measurement unit and external sensor performance on navigation systems, SAIMUN was developed. SAIMUN utilizes Monte Carlo sampling and linear covariance analysis to propagate the initial position, velocity, and attitude uncertainty forward in time, while accounting for the injection of uncertainty into the position, velocity, and attitude due to error corruption present in the IMU. Updates to the uncertainty are also considered when incorporating external sensor data using linear covariance analysis.

The effect of the following IMU error sources on the position, velocity, and attitude uncertainty is quantified in SAIMUN: startup bias, bias instability, random walk, scale factor error, axes misalignment, axes nonorthogonality, and quantization effects. The modeling of these IMU error parameters is outlined in Section 3.2. The location of the IMU with respect to the center of mass (CM) and the IMU timestep, which is the time between successive measurements of the acceleration and angular velocity provided by the IMU, are also specified. Finally, the mechanization type for the IMU, which can be

- strapdown or
- space stabilized

is given. The IMU mechanizations are further described in Section 3.4.

In addition to information about the IMU configuration and error sources, the tool also requires a tabulated true trajectory, which is provided as an input to SAIMUN to ensure that the tool can be applied to any vehicle, ranging from ground vehicles to aquatic vehicles, air vehicles, and spacecraft. The tabulated truth trajectory consists of the following information:

- inertial position vector of the CM of the vehicle,
- inertial velocity vector of the CM of the vehicle,
- inertial acceleration vector of the CM of the vehicle,
- attitude of the vehicle body frame, expressed as a quaternion and defined as a 3-2-1 rotation from the inertial frame,
- angular velocity of the vehicle body frame expressed in the vehicle body frame,
- angular acceleration of the vehicle body frame expressed in the vehicle body frame, and
- seconds elapsed since the beginning of the trajectory.

An initial epoch, which corresponds to the start of the tabulated truth trajectory and is specified as a Julian Date, is also provided to link the truth trajectory to a known reference time.

SAIMUN is also integrated with the ability to process measurements from external sources. For each sensor the following information is required:

- measurement frequency,
- measurement error parameters given as a covariance matrix,
- sensor position and attitude with respect to the IMU, and
- latency associated with each measurement.

The analysis and initial distribution inputs required are:

- Number of Monte Carlo sample trajectories. This parameter is a trade between computational effort required and the accuracy of the results.
- Covariance of the initial position, velocity, and attitude of the vehicle, corresponding to the initial epoch.

SAIMUN then implements linear covariance and/or Monte Carlo analysis to propagate the initial uncertainty forward in time, while potentially accounting for the presence of external sensor data, and outputs the following plots:

- position standard deviation over time,
- velocity standard deviation over time, and
- attitude standard deviation over time.

3.2 IMU Modeling

The acceleration and angular velocity measured by an IMU are corrupted by a variety of error sources. SAIMUN accounts for the most dominant sources of error by specifying a probability density function (pdf) for the error parameters to define each error source. The pdfs implemented in SAIMUN are

- the Dirac distribution, which is used to represent a constant parameter specified solely by the mean,
- the normal distribution, which represents a Gaussian distributed parameter that is solely defined by a mean and variance (or standard deviation), and
- the uniform distribution, which represents a parameter that is equiprobable between lower and upper bounds.

These distributions are graphically represented in Fig. (1).

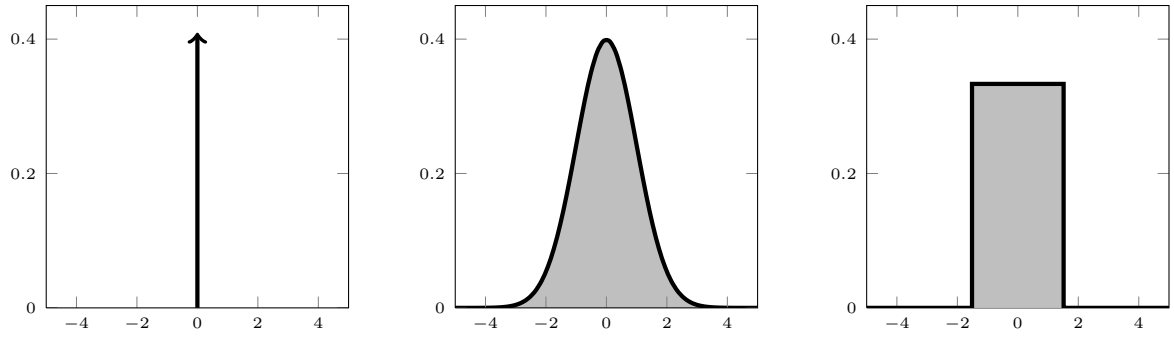
The IMU model accounting for error sources is given for the accelerometers and gyroscopes as

$$\boldsymbol{a}_{m,k} = \boldsymbol{a}_{\mathrm{Q},k}\Big((\boldsymbol{I} + \boldsymbol{N}_a + \boldsymbol{M}_a)\,(\boldsymbol{I} + \boldsymbol{S}_a)\,\big(\boldsymbol{T}_i^{\mathrm{IMU}}\boldsymbol{a}_{ng,k}^{i} + \boldsymbol{b}_{a,0} + \boldsymbol{b}_{a,k} + \boldsymbol{w}_{a,k}\big)\Big) \tag{1a}$$

$$\boldsymbol{\omega}_{m,k} = \boldsymbol{\omega}_{\mathrm{Q},k}\Big((\boldsymbol{I} + \boldsymbol{N}_g + \boldsymbol{M}_g)\,(\boldsymbol{I} + \boldsymbol{S}_g)\,\Big(\boldsymbol{\omega}_{p/i,k}^{\mathrm{IMU}} + \boldsymbol{b}_{g,0} + \boldsymbol{b}_{g,k} + \boldsymbol{w}_{g,k}\Big)\Big)\,, \tag{1b}$$

where

$\boldsymbol{a}_{ng,k}^{i}$ is the true non-gravitational inertial acceleration experienced by the IMU expressed in the inertial frame at t_k,



Zero-Mean Dirac Distribution | Standard Normal Distribution | Uniform Distribution

Figure 1. Three Distributions Implemented for Error Parameters in SAIMUN

$\boldsymbol{\omega}^{\mathbf{IMU}}_{p/i,k}$ is the true angular velocity of the IMU platform with respect to the inertial frame, expressed in the IMU frame at t_k,

$\boldsymbol{T}_i^{\mathrm{IMU}}$ is the rotation matrix representing the rotation from the inertial to the IMU frame,

$\boldsymbol{b}_{0,k}$ is the startup bias of the accelerometers,

$\boldsymbol{b}_{a,k}$ is the bias of the accelerometer at t_k, which changes due to bias instability,

$\boldsymbol{w}_{a,k}$ is the thermo-mechanical zero-mean white noise present in the accelerometers,

$\boldsymbol{S}_a$ is the scale factor error matrix of the accelerometers,

$\boldsymbol{M}_a$ is the axes misalignment matrix of the accelerometers,

$\boldsymbol{N}_a$ is the axes nonorthogonality matrix of the accelerometers, and

$\boldsymbol{a}_{\mathbf{Q},k}$ is the quantization effect caused by analog-to-digital conversion.

and similarly for the gyroscopes. For simplicity, let $\boldsymbol{a}^i_{ng,k}$ be $\boldsymbol{a}_k$ and $\boldsymbol{\omega}^{\mathrm{IMU}}_{p/i,k}$ be $\boldsymbol{\omega}_k$. The error sources are applied in the following order:

1. Walking bias and thermomechanical noise are applied first because they affect the sensor (accelerometer or gyroscope) regardless of how the sensor is mounted with respect to the defined IMU frame,
2. A scale factor error is applied next to account for errant voltages, circuitry, etc. in converting the sensor output to a value that can be quantized,
3. Axes nonorthogonality and misalignment errors are applied next to account for the mounting error between the sensors and the defined IMU frame,
4. Quantization error is applied last to emulate the Analog to Digital Conversion necessary for quantizing the sensor signal.

The mean of the startup bias of the sensor is included in the walking bias as it is assumed known from sensor testing.

The sources of error used by this model affect the gyroscopes identically and thus their presentation is omitted.

3.2.1 Startup Bias. The startup bias affecting the accelerometers, $\boldsymbol{b}_{a,0}$, is the average output of the accelerometers while they are not undergoing any acceleration [1]. For practical applications, the startup bias is quantified by accounting for and removing the accelerometer output due to gravity while the IMU remains motionless. Figure (2) shows an example of the startup bias of an accelerometer undergoing no acceleration. SAIMUN assumes that the statistics of the startup bias

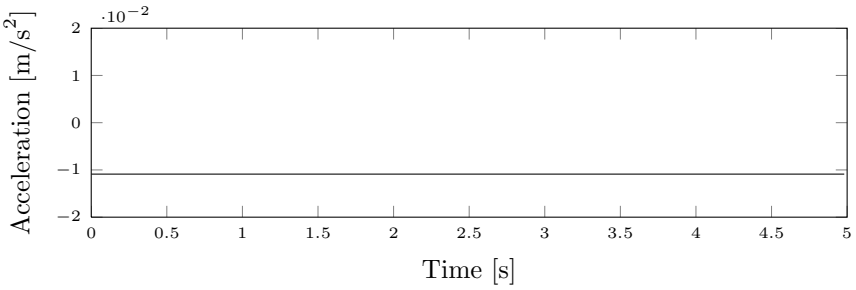


Figure 2. Accelerometer Output Corrupted by Startup Bias

are well known and can be approximated by one of the three supported error-parameter pdfs. The startup bias is then realized from the error-parameter pdf during virtual IMU initialization.

3.2.2 Bias Instability. The bias of an accelerometer tends to drift slowly over time and is known as a *walking* bias [1]. The walking bias affecting the accelerometers at t_k, $\boldsymbol{b}_{a,k}$, is the change in the bias after the IMU was powered on. This effect is modeled by a first-order random walk model, which causes a linear increase in the variance of the accelerometer bias with time. The discrete first-order random walk model is given by

$$\boldsymbol{b}_{a,k} = \boldsymbol{b}_{a,k-1} + \boldsymbol{w}_{\mathrm{BI},k}\,, \tag{2}$$

where $\boldsymbol{w}_{\mathrm{BI},k}$ is a zero-mean white noise process of standard deviation $\boldsymbol{\sigma}_{\mathrm{BI}}$. The initial standard deviation in the walking bias is zero, and the standard deviation at a later time, t^*_{BI}, is specified as $\boldsymbol{\sigma}^*_{\mathrm{BI}}$. Utilizing the zero initial condition and applying Eq. (2) recursively from t_0 to t_k yields the walking bias at t_k as

$$\boldsymbol{b}_{a,k} = \sum_{i=1}^{t_k/\Delta t} \boldsymbol{w}_{\mathrm{BI},k}\,, \tag{3}$$

where Δt is the timestep of the IMU. Noting that the variance of the summation of n independent random variables denoted by X_i is equivalent to the sum of the variances of n independent random variables, i.e.

$$\mathrm{Var}\left[\sum_{i=1}^{n} X_i\right] = \sum_{i=1}^{n} \mathrm{Var}\left[X_i\right]\,;$$

this yields

$$\mathrm{Var}\left[\boldsymbol{b}_{a,k}\right] = \sum_{i=1}^{t_k/\Delta t} \mathrm{Var}\left[\boldsymbol{w}_{\mathrm{BI},k}\right] = \frac{t_k}{\Delta t}\boldsymbol{\sigma}^2_{\mathrm{BI}}\,, \tag{4}$$

when applied to Eq. (3). Evaluating Eq. (4) at the boundary condition gives

$$\frac{t_k^*}{\Delta t}\boldsymbol{\sigma}_{\mathrm{BI}}^2 = \boldsymbol{\sigma}_{\mathrm{BI}}^{*\,2}\,. \tag{5}$$

The standard deviation of the zero-mean white noise process driving the random walk model, $\boldsymbol{\sigma}_{\mathrm{BI}}$, can now be found by manipulating Eq. (5) as

$$\boldsymbol{\sigma}_{\mathrm{BI}} = \sqrt{\frac{\Delta t}{t_k^*}}\boldsymbol{\sigma}_{\mathrm{BI}}^*\,. \tag{6}$$

Figure (3) shows an example accelerometer undergoing a random walk due to bias instability. SAIMUN inputs the boundary conditions $\boldsymbol{\sigma}_{\mathrm{BI}}^*$ and t_{BI}^*, the IMU timestep, Δt, and the desired error-

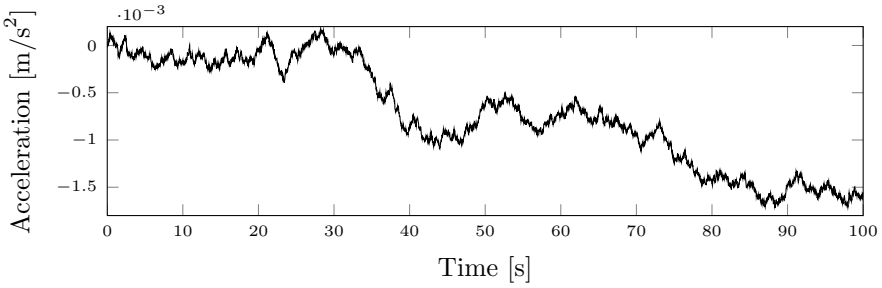


Figure 3. Accelerometer Output Corrupted by Bias Instability

parameter pdf-type for $\boldsymbol{w}_k$ and calculates the standard deviation of the underlying noise process driving the random walk model according to Eqs. (2)–(6). Each step in the random walk model is realized independently for each IMU.

3.2.3 Velocity Random Walk. When thermo-mechanical zero-mean white noise, $\boldsymbol{w}_{a,k}$, present in the accelerometers is integrated, a random walk is produced in the velocity state, and the standard deviation of the velocity grows proportionally to the square-root of time [1]. The variance of the white noise present in the accelerometers is described by the Velocity Random Walk (VRW) specification. The standard deviation of the zero-mean white noise is given as a function of the VRW specification as

$$\boldsymbol{\sigma}_{\mathrm{VRW}} = \frac{\mathrm{VRW}_{\mathrm{spec}}}{\sqrt{\Delta t}}\,.$$

Example accelerometer output corrupted by zero-mean white-noise is shown in Fig. (4). SAIMUN inputs the VRW specification and IMU timestep to calculate the standard deviation of the white noise process underlying the velocity random walk. This zero-mean white noise process is realized independently at each t_k for the virtual IMU.

3.2.4 Scale Factor Error. Errors in analog-to-digital converters and other circuitry can induce scale factor errors in the accelerometers, denoted by $\boldsymbol{S}_a$. For instance, errors may be incurred due to reference voltage errors caused by a variety of sources, including, among other things, temperature fluctuations and calibration errors. For illustration purposes, consider a simple example of a generic scale factor error, s, and assume that the nominal reference voltage for the analog-to-digital

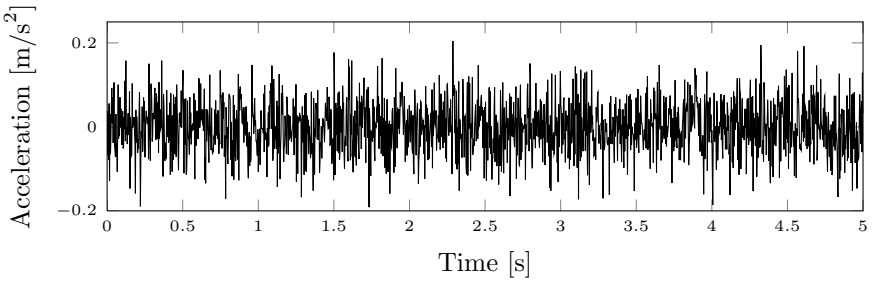


Figure 4. Accelerometer Output Corrupted by Thermo-Mechanical Zero-Mean White Noise

converter is $V_{\text{nom}} = 5$ V, while the actual reference voltage provided to the analog-to-digital converter is $V_{\text{act}} = 5.02$ V. The scale factor error of the signal being converted by the analog-to-digital converter is then given by

$$s = \frac{V_{\text{nom}} - V_{\text{act}}}{V_{\text{act}}} = \frac{5 \text{ V} - 5.02 \text{ V}}{5.02 \text{ V}} = -0.003984 = -0.3984\% \,.$$

Denoting the scale factor errors for each axis of the accelerometer triad by s_x, s_y, and s_z, the total accelerometer scale factor error may be expressed as

$$\boldsymbol{S}_a = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix} .$$

SAIMUN inputs an error-parameter pdf for s_x, s_y, s_z and realizes these parameters for the virtual IMU from this pdf.

3.2.5 Axes Misalignment Error. Errors in the manufacturing process and mounting of an IMU can lead to axes misalignment errors, denoted by $\boldsymbol{M}_a$. That is, the actual sensor axes are offset from the IMU case frame by a set of small rotations. This is readily seen by considering the term $\boldsymbol{I} + \boldsymbol{M}_a$ in Eq. (1a), which is of the form of a rotation matrix for small angles. Therefore, the actual sensed outputs of the IMU are not aligned perfectly with the modeled case frame of the IMU. The accelerometer misalignment errors are represented within $\boldsymbol{M}_a$ by m_x, m_y, and m_z, with each term corresponding to its respective sensor axis. The misalignment errors are expressed in matrix form as

$$\boldsymbol{M}_a = \begin{bmatrix} 0 & m_z & -m_y \\ -m_z & 0 & m_x \\ m_y & -m_x & 0 \end{bmatrix} .$$

SAIMUN inputs an error-parameter pdf for m_x, m_y, m_z and realizes these parameters for the virtual IMU from this pdf.

3.2.6 Axes Nonorthogonality Error. Errors in the manufacturing process of an IMU can lead to axes nonorthogonality errors, denoted by $\boldsymbol{N}_a$. If the sensor had no nonorthogonality errors, each axis of the sensor triad would be at perfect right angles to the other two axes. Therefore, the nonorthogonality errors account for small deviations from perfectly orthogonal axes and allow for the modeling of sensor outputs that effectively measure small overlaps between the individual

axes. The accelerometer axes nonorthogonality errors are represented within $\boldsymbol{N}_a$ by n_x, n_y, and n_z, with each term corresponding to its respective sensor axis. The nonorthogonality errors can be expressed in matrix form as

$$\boldsymbol{N}_a = \begin{bmatrix} 0 & n_z & n_y \\ n_z & 0 & n_x \\ n_y & n_x & 0 \end{bmatrix} .$$

SAIMUN inputs an error-parameter pdf for n_x, n_y, n_z and realizes these parameters for the virtual IMU from this pdf.

Example output of an accelerometer triad undergoing 1 m/s acceleration in the z direction with scale factor, axes misalignment, and axes nonorthogonality errors is shown in Fig. (5).

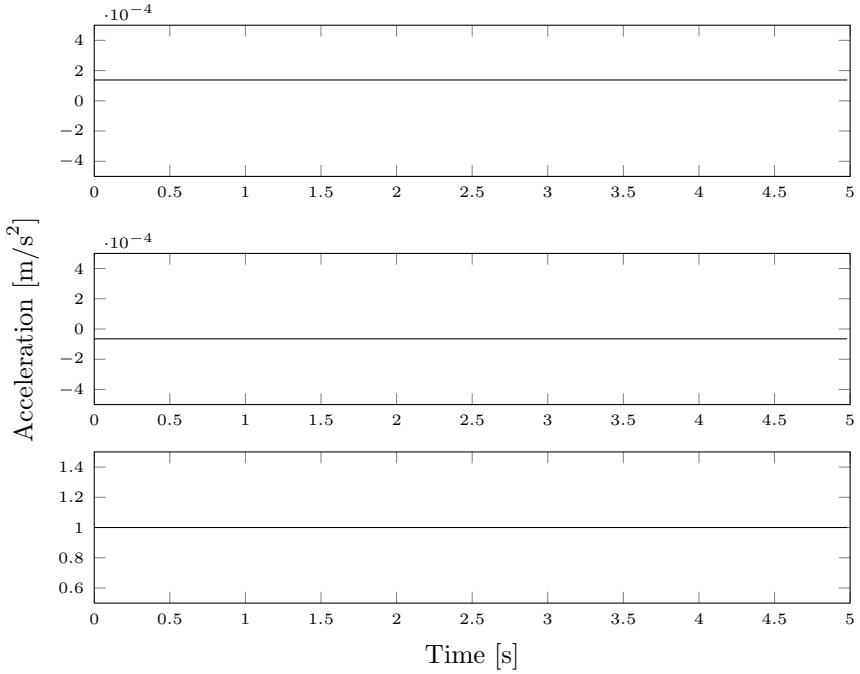


Figure 5. Three-Axis Accelerometer Output Corrupted by Scale Factor, Misalignment, and Nonorthogonality Errors

3.2.7 Quantization Error. Quantization error arises in IMUs due to the limitations of analog-to-digital signal conversion and is modeled such that the acceleration and angular velocity measurements are affected as

$$\boldsymbol{a}_{\mathrm{Q},k}(\boldsymbol{a}_k) = \mathrm{round}\left(\frac{2^{n-1}}{a_{\max}}\boldsymbol{a}_k\right)\frac{a_{\max}}{2^{n-1}} ,$$

where n is the bitrate of the analog to digital converter, and $a_{\max}$ is the maximum value able to be quantized by the analog to digital converter. It is noted that the quantization error is applied after all of the other errors have been applied, such that the output of the quantization error is the measured acceleration of Eq. (1a). Values of the bitrate and maximum value are both input into SAIMUN. The quantization error in angular velocity is calculated similarly. Figure (6) shows a quantized signal from an IMU with thermo-mechanical noise present.

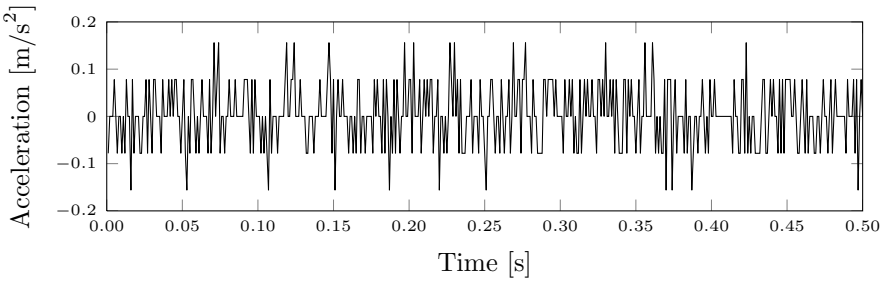


Figure 6. Accelerometer Output Corrupted by Quantization Error

3.3 IMU Inversion

In order to perform dead-reckoning navigation, the sensor errors must be removed from the non-gravitational acceleration and angular velocity. An expression for how these quantities are corrupted by the sensor are given by Eqs. (1). Solving this equation for the uncorrupted measurements requires an "inversion." The measured non-gravitational acceleration and angular velocity are modeled using Eqs. (1). For some vector $\boldsymbol{v} = [v_x \;\; v_y \;\; v_z]^T$, define the matrices $[\boldsymbol{v}\diagdown]$, $[\boldsymbol{v}\times]$, and $[\boldsymbol{v}*]$ to be

$$[\boldsymbol{v}\diagdown] = \begin{bmatrix} v_x & 0 & 0 \\ 0 & v_y & 0 \\ 0 & 0 & v_z \end{bmatrix}, \; [\boldsymbol{v}\times] = \begin{bmatrix} 0 & v_z & -v_y \\ -v_z & 0 & v_x \\ v_y & -v_x & 0 \end{bmatrix}, \text{ and } \; [\boldsymbol{v}*] = \begin{bmatrix} 0 & v_z & v_y \\ v_z & 0 & v_x \\ v_y & v_x & 0 \end{bmatrix}.$$

After omitting the effects due to quantization, the errors in the acceleration and angular velocity can be expressed as

$$\boldsymbol{a}_{m,k} = (\boldsymbol{I} + \boldsymbol{N}_a + \boldsymbol{M}_a)(\boldsymbol{I} + \boldsymbol{S}_a)(\boldsymbol{a}_k + \boldsymbol{b}_{a,k} + \boldsymbol{w}_{a,k}) \,, \tag{7}$$

where $\boldsymbol{a}_k = \boldsymbol{T}_i^{\text{IMU}} \boldsymbol{a}_k^i$ is used for compactness. Equation (7) may be solved for $\boldsymbol{a}_k$ in terms of the measured acceleration and the error sources to yield

$$\boldsymbol{a}_k = (\boldsymbol{I} + \boldsymbol{S}_a)^{-1} (\boldsymbol{I} + \boldsymbol{N}_a + \boldsymbol{M}_a)^{-1} \boldsymbol{a}_{m,k} - \boldsymbol{b}_{a,k} - \boldsymbol{w}_{a,k} \,.$$

Noting that $(\boldsymbol{I} + \boldsymbol{M}_a + \boldsymbol{N}_a)(\boldsymbol{I} + \boldsymbol{S}_a) \approx \boldsymbol{I} + \boldsymbol{\Lambda}_a$, applying the matrix inversion lemma, and simplifying the resulting expression, it follows that the true non-gravitational acceleration written in terms of the measured non-gravitational acceleration as

$$\boldsymbol{a}_k = \boldsymbol{a}_{m,k} + [\boldsymbol{a}_{m,k}\times]\boldsymbol{m}_a - [\boldsymbol{a}_{m,k}*]\boldsymbol{n}_a - [\boldsymbol{a}_{m,k}\diagdown]\boldsymbol{s}_a - \boldsymbol{b}_{a,k} - \boldsymbol{w}_{a,k} \,, \tag{8}$$

from which one may obtain an estimate of the true non-gravitational acceleration as $\hat{\boldsymbol{a}}_k = \mathrm{E}\{\boldsymbol{a}_k\}$, which gives

$$\hat{\boldsymbol{a}}_k = \boldsymbol{a}_{m,k} + [\boldsymbol{a}_{m,k}\times]\hat{\boldsymbol{m}}_a - [\boldsymbol{a}_{m,k}*]\hat{\boldsymbol{n}}_a - [\boldsymbol{a}_{m,k}\diagdown]\hat{\boldsymbol{s}}_a - \hat{\boldsymbol{b}}_{a,k} - \hat{\boldsymbol{w}}_{a,k} \,. \tag{9}$$

If all of the error sources are zero-mean, it follows from Eq. (9) that

$$\hat{\boldsymbol{a}}_k = \boldsymbol{a}_{m,k} \,.$$

Parallel results hold for expressing the true angular velocity in terms of the measured angular velocity. That is, following the same process used in arriving at Eqs. (8) and (9), it can be shown that the true and estimated angular velocities can be expressed as

$$\boldsymbol{\omega}_k = \boldsymbol{\omega}_{m,k} + [\boldsymbol{\omega}_{m,k}\times]\boldsymbol{m}_g - [\boldsymbol{\omega}_{m,k}*]\boldsymbol{n}_g - [\boldsymbol{\omega}_{m,k}\diagdown]\boldsymbol{s}_g - \boldsymbol{b}_{g,k} - \boldsymbol{w}_{g,k}$$
$$\hat{\boldsymbol{\omega}}_k = \boldsymbol{\omega}_{m,k} + [\boldsymbol{\omega}_{m,k}\times]\hat{\boldsymbol{m}}_g - [\boldsymbol{\omega}_{m,k}*]\hat{\boldsymbol{n}}_g - [\boldsymbol{\omega}_{m,k}\diagdown]\hat{\boldsymbol{s}}_g - \hat{\boldsymbol{b}}_{g,k} - \hat{\boldsymbol{w}}_{g,k}$$

where all of the error sources are now for the gyro instead of the accelerometer. As with the accelerometer, if all of the error sources are zero-mean, it follows that

$$\hat{\boldsymbol{\omega}}_k = \boldsymbol{\omega}_{m,k} \, .$$

Define the non-gravitational acceleration and angular velocity deviations to be the difference in the truth from the estimate as

$$\begin{aligned} \delta \boldsymbol{a}_k &= \boldsymbol{a}_k - \hat{\boldsymbol{a}}_k \\ \delta \boldsymbol{\omega}_k &= \boldsymbol{\omega}_k - \hat{\boldsymbol{\omega}}_k \, . \end{aligned}$$

Taking the difference between true and estimated expressions yields

$$\begin{aligned} \delta \boldsymbol{a}_k &= [\boldsymbol{a}_{m,k} \times] \delta \boldsymbol{m}_a - [\boldsymbol{a}_{m,k} *] \delta \boldsymbol{n}_a - [\boldsymbol{a}_{m,k} \diagdown] \delta \boldsymbol{s}_a - \delta \boldsymbol{b}_{a,k} - \delta \boldsymbol{\omega}_{a,k} \\ \delta \boldsymbol{\omega}_k &= [\boldsymbol{\omega}_{m,k} \times] \delta \boldsymbol{m}_g - [\boldsymbol{\omega}_{m,k} *] \delta \boldsymbol{n}_g - [\boldsymbol{\omega}_{m,k} \diagdown] \delta \boldsymbol{s}_g - \delta \boldsymbol{b}_{g,k} - \delta \boldsymbol{\omega}_{g,k} \, , \end{aligned}$$

where the deviations in the error parameters are defined similarly to acceleration and angular velocity.

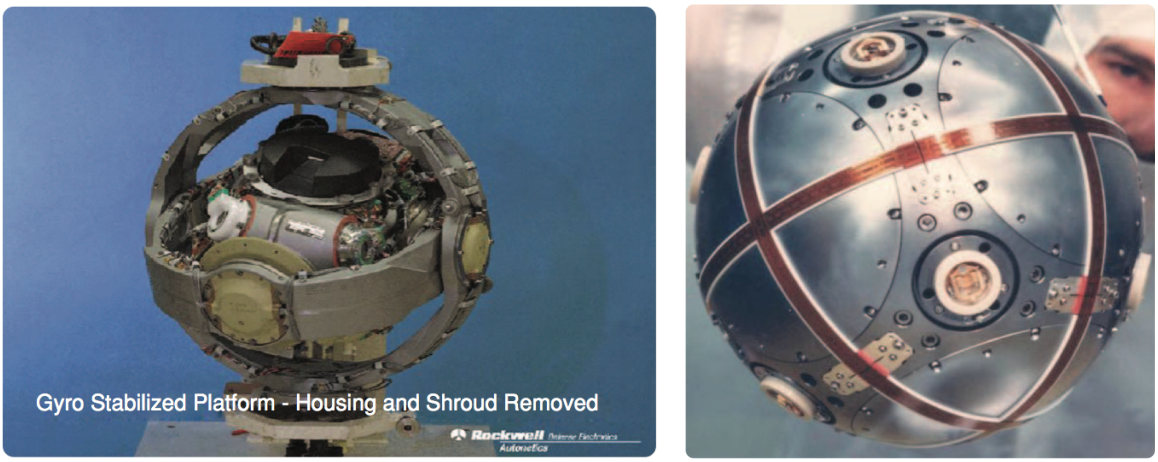
3.4 IMU Mechanizations

Traditionally, IMU-based navigation is mechanized in one of two ways: gimbaled/gyro-stabilized or strapdown. In the gimbaled mechanization, the triad of accelerometers is placed on a "stable table" at the center of set of interconnected gimbals. This enables the stable table, through the use of feedback tracking using the gyros, to track a predetermined attitude such that the accelerometers measure a certain, desirable resultant linear motion. Typically, the stable table is configured to track a north-east-down coordinate system, in which case the accelerometers measure the linear motion in these directions, or it is configured to track the inertial coordinate system (i.e. to remain fixed in space as the vehicle maneuvers).

The most basic of gimbal-based mechanizations employs three interconnected gimbal rings, such as is found on the Minuteman III intercontinental ballistic missile (see Fig. (7(a))). While this configuration provides the ability to actively control the orientation of the stable table through motors mounted on the gimbals, the use of three gimbals sometimes leads to gimbal lock and prevents the stable table from tracking all possible attitudes that may be required.

More advanced configurations of the stable table mounting design can be used to replace the original gimbaled system. This is accomplished by using four or more gimbals or by using a fluid stabilization system, which has the stable table floating in the center of a sphere filled with pressurized fluid. Whereas the attitude of the stable table is controlled using torquers and manipulation of the gimbals in the original design, the attitude of the stable table in the fluidic design is controlled via nozzle actuators which avoids any possible complications due to gimbal lock. One such implementation of the fluidic system is the Advanced Inertial Reference Sphere (AIRS) that was developed for the LGM-118A Peacekeeper (see Fig. (7(b))).

In the strapdown mechanization, the triads of accelerometers and gyros are fixed (strapped down) to the body of the vehicle, and there is no stable table. This means that where the stable table was used to mechanically track the inertial frame in order to track a predetermined attitude and sense accelerations in this rotating frame, the gyro outputs are now used to numerically track the orientation of the IMU (called the case frame) with respect to the inertial frame. To avoid any singularities present in three-parameter attitude representations such as Euler angles, the attitude quaternion is used and can be propagated forward in time using the sensed angular velocity from



(a) Minuteman III IMU (b) Peacekeeper IMU

Figure 7. Examples of Stable Table IMU Mechanizations [2]

the IMU. Additionally, the accelerations are now sensed in the case frame, and these must be transformed using the case-to-inertial attitude in order to predict the translational position and velocity of the vehicle.

Two inertial navigation system mechanizations are presented here: a strapdown system and a passively commanded inertially stable space stabilized gimbaled system. For each of the aforementioned mechanizations, the governing equations for the position, velocity, and attitude of the IMU are presented, and a set of discretized equations are developed. The discretized equations governing the position, velocity, and attitude of the IMU then are representative of the software and hardware that would be used in each inertial navigation mechanization.

In the following sections, the following remains consistent:

$\boldsymbol{r}$	position
$\boldsymbol{v}$	velocity
$\bar{\boldsymbol{q}}$	attitude expressed as a quaternion
$\boldsymbol{a}$	acceleration
$\boldsymbol{\omega}$	angular velocity
$\bar{\boldsymbol{\omega}}$	angular velocity expressed as a true quaternion
$\boldsymbol{T}_a^b$	rotation matrix from frame a to frame b

IMU	inertial measurement unit
CM	center-of-mass of the vehicle
p	platform frame of the IMU
b	body frame of the vehicle
s	sensor case frame
i	Earth-Centered Inertial frame
f	Earth-Centered Earth-Fixed
g	due to gravity
ng	not due to gravity
m	measured

and a subscript given in the form a/b denotes a with respect to b. For brevity, only the governing equations and the resulting discretized update equations are given in the following sections.

3.4.1 Strapdown Mechanization. The translational and rotational states of a strapdown IMU aboard a vehicle are governed by [3, 4]

$$\begin{aligned}\dot{\boldsymbol{r}}^i_{\text{IMU}}(t) &= \boldsymbol{v}^i_{\text{IMU}}(t)\\ \dot{\boldsymbol{v}}^i_{\text{IMU}}(t) &= \boldsymbol{a}^i_g(\boldsymbol{r}^i_{\text{IMU}}(t) + \boldsymbol{T}^i_b(t)\boldsymbol{r}^b_{\text{CM/IMU}}) + \boldsymbol{T}^i_b(t)\boldsymbol{a}^b_{ng}(t)\\ \dot{\bar{\boldsymbol{q}}}^b_i(t) &= \frac{1}{2}\bar{\boldsymbol{\omega}}^b_{b/i}(t) \otimes \bar{\boldsymbol{q}}^b_i(t)\,,\end{aligned}$$

For notational ease, let

$$\begin{array}{rclrclrcl} \boldsymbol{r}^i_{\text{IMU}}(t) & \to & \boldsymbol{r}(t)\,, & \boldsymbol{v}^i_{\text{IMU}}(t) & \to & \boldsymbol{v}(t)\,, & \boldsymbol{a}^i_g(\cdot) & \to & \boldsymbol{g}(\cdot)\,,\\ \boldsymbol{T}^i_b(t) & \to & \boldsymbol{T}^T(t)\,, & \boldsymbol{a}^b_{ng}(t) & \to & \boldsymbol{a}(t)\,, & \bar{\boldsymbol{q}}^b_i(t) & \to & \bar{\boldsymbol{q}}(t)\,,\\ \boldsymbol{\omega}^b_{b/i}(t) & \to & \boldsymbol{\omega}(t)\,, & \boldsymbol{r}^p_{\text{CM/IMU}} & \to & \boldsymbol{d}\,, & \text{and} \quad \boldsymbol{r}^i_{\text{IMU}}(t) + \boldsymbol{T}^i_b(t)\boldsymbol{r}^b_{\text{CM/IMU}} & \to & \boldsymbol{s}(t)\,. \end{array}$$

With these substitutions, the equations of motion may be written succinctly as

$$\begin{aligned}\dot{\boldsymbol{r}}(t) &= \boldsymbol{v}(t)\\ \dot{\boldsymbol{v}}(t) &= \boldsymbol{g}(\boldsymbol{s}(t)) + \boldsymbol{T}^T(t)\boldsymbol{a}(t)\\ \dot{\bar{\boldsymbol{q}}}(t) &= \frac{1}{2}\bar{\boldsymbol{\omega}}(t) \otimes \bar{\boldsymbol{q}}(t)\,.\end{aligned}$$

The discrete-time dead-reckoning equations for the position, velocity, and attitude of a vehicle with a strapdown IMU are given by:

$$\begin{aligned}\boldsymbol{r}_k &= \boldsymbol{r}_{k-1} + \boldsymbol{v}_{k-1}\Delta t_k + \frac{1}{2}\boldsymbol{g}_{k-1}\Delta t_k^2 + \frac{1}{2}\boldsymbol{T}^T_{k-1}\boldsymbol{a}_k\Delta t_k^2 \qquad (10a)\\ &\quad - \frac{1}{6}\left(\boldsymbol{G}_{k-1}\boldsymbol{T}^T_{k-1}[\boldsymbol{d}\times] + \boldsymbol{T}^T_{k-1}\left[\boldsymbol{a}_k\times\right]\right)\boldsymbol{\omega}_k\Delta t_k^3\\ \boldsymbol{v}_k &= \boldsymbol{v}_{k-1} + \boldsymbol{g}_{k-1}\Delta t_k + \boldsymbol{T}^T_{k-1}\boldsymbol{a}_k\Delta t_k - \frac{1}{2}\left(\boldsymbol{G}_{k-1}\boldsymbol{T}^T_{k-1}[\boldsymbol{d}\times] + \boldsymbol{T}^T_{k-1}\left[\boldsymbol{a}_k\times\right]\right)\boldsymbol{\omega}_k\Delta t_k^2 \qquad (10b)\\ \bar{\boldsymbol{q}}_k &= \bar{\boldsymbol{q}}(\boldsymbol{\omega}_k\Delta t_k) \otimes \bar{\boldsymbol{q}}_{k-1}\,, \qquad (10c)\end{aligned}$$

where $\boldsymbol{g}_{k-1}$ and $\boldsymbol{G}_{k-1}$ are the two-body gravitational acceleration experienced at the CM (not the IMU) and the associated Jacobian evaluated at the estimated position of the CM at time t_{k-1}.

3.4.2 Space Stabilized Mechanization. The space stabilized mechanization tracks an Earth-centered, inertially non-rotating coordinate frame. This reference frame is realized by a gyro-stabilized platform with triaxial control. Except for torques applied as compensation to account for anisoelastic effects, thermal and acceleration sensitivity, etc., the gyroscopes are uncommanded and the system is "free-floating."

Noting that the specific force is actually measured in the IMU platform frame, the translational equations governing the dynamical evolution of the position and attitude of the IMU can be expressed as a set of first-order differential equations given by

$$\begin{aligned}\dot{\boldsymbol{r}}^i_{\text{IMU}}(t) &= \boldsymbol{v}^i_{\text{IMU}}(t)\\ \dot{\boldsymbol{v}}^i_{\text{IMU}}(t) &= \boldsymbol{g}^i(\boldsymbol{r}^i_{\text{IMU}}(t)) + \boldsymbol{T}^i_p(t)\boldsymbol{f}^p(t)\\ \dot{\bar{\boldsymbol{q}}}^p_i(t) &= \frac{1}{2}\bar{\boldsymbol{\omega}}^p_{p/i}(t) \otimes \bar{\boldsymbol{q}}^p_i(t)\end{aligned}$$

For notational simplicity, let $\boldsymbol{r}^i_{\text{IMU}}(t) \to \boldsymbol{r}$, $\boldsymbol{v}^i_{\text{IMU}}(t) \to \boldsymbol{v}$, $\boldsymbol{\omega}^p_{p/i}(t) \to \boldsymbol{\omega}(t)$, $\bar{\boldsymbol{q}}^p_i(t) \to \bar{\boldsymbol{q}}(t)$, $\boldsymbol{T}^i_p(t) = \boldsymbol{T}^T(t)$, and $\boldsymbol{f}^p(t) = \boldsymbol{f}(t)$, such that the governing equations become

$$\begin{aligned}
\dot{\boldsymbol{r}}(t) &= \boldsymbol{v}(t) \\
\dot{\boldsymbol{v}}(t) &= \boldsymbol{g}(\boldsymbol{r}(t)) + \boldsymbol{T}^T(t)\boldsymbol{f}(t) \\
\dot{\bar{\boldsymbol{q}}}(t) &= \frac{1}{2}\bar{\boldsymbol{\omega}}(t) \otimes \bar{\boldsymbol{q}}(t)\,.
\end{aligned}$$

Discretization and integration yield the following discrete-time propagation equations:

$$\boldsymbol{r}_k = \boldsymbol{r}_{k-1} + \boldsymbol{v}_{k-1}\Delta t_k + \frac{1}{2}\left(\boldsymbol{T}^T_{k-1}\boldsymbol{f}_k + \boldsymbol{g}_{k-1}\right)\Delta t_k^2 \tag{11a}$$

$$\boldsymbol{v}_k = \boldsymbol{v}_{k-1} + \left(\boldsymbol{T}^T_{k-1}\boldsymbol{a}_k + \boldsymbol{g}_{k-1}\right)\Delta t_k \tag{11b}$$

$$\bar{\boldsymbol{q}}_k = \bar{\boldsymbol{q}}(\boldsymbol{\omega}_k\Delta t_k) \otimes \bar{\boldsymbol{q}}_{k-1}\,. \tag{11c}$$

3.5 Monte Carlo Analysis

In general, either of the mechanizations of the IMU can be represented by a nonlinear dynamical system. If the expressions for position, velocity, and attitude are collected into a state vector, $\boldsymbol{x}$, the general nonlinear dynamical system representing the forward evolution of the state for either of the IMU systems may be expressed as

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{a}_k, \boldsymbol{\omega}_k, t_k)\,, \tag{12}$$

where $\boldsymbol{a}_k$ and $\boldsymbol{\omega}_k$ represent the true vehicle non-gravitational acceleration and angular velocity, respectively. For the mechanizations presented in Section 3.4, $\boldsymbol{f}(\cdot)$ represents one of Eqs. (10) or Eqs. (11). Functionally, the relationship between the true and measured IMU outputs is given by

$$\boldsymbol{a}_{m,k} = \boldsymbol{a}(\boldsymbol{\phi}_k, t_k) \qquad \text{and} \qquad \boldsymbol{\omega}_{m,k} = \boldsymbol{\omega}(\boldsymbol{\psi}_k, t_k)\,, \tag{13}$$

where $\boldsymbol{\phi}_k$ and $\boldsymbol{\psi}_k$ represent the IMU model parameters, which also have governing evolutionary equations as

$$\boldsymbol{\phi}_k = \boldsymbol{g}(\boldsymbol{\phi}_k, t_k) \qquad \text{and} \qquad \boldsymbol{\psi}_k = \boldsymbol{h}(\boldsymbol{\psi}_k, t_k)\,. \tag{14}$$

In the preceding discussions, many possibilities for the forms of $\boldsymbol{g}$, $\boldsymbol{h}$, $\boldsymbol{\phi}$, and $\boldsymbol{\psi}$ have been given. For example, for the simple additive model where the true IMU outputs are corrupted by a constant bias $\boldsymbol{b}$ and a white-noise sequence $\boldsymbol{w}$, Eqs. (13) become

$$\boldsymbol{a}_{m,k} = \boldsymbol{a}_k + \boldsymbol{b}_a + \boldsymbol{w}_{a,k} \qquad \text{and} \qquad \boldsymbol{\omega}_{m,k} = \boldsymbol{\omega}_k + \boldsymbol{b}_g + \boldsymbol{w}_{g,k}\,,$$

and the parameters of the IMU error model are

$$\boldsymbol{\phi}_k = \begin{bmatrix} \boldsymbol{b}_a \\ \boldsymbol{w}_{a,k} \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{\psi}_k = \begin{bmatrix} \boldsymbol{b}_g \\ \boldsymbol{w}_{g,k} \end{bmatrix},$$

with the governing equations $\boldsymbol{\phi}_k = \boldsymbol{\phi}_{k-1}$ and $\boldsymbol{\psi}_k = \boldsymbol{\psi}_{k-1}$ (note that the white noise sequence does not require a formal time-wise evolution owing to the properties of white noise).

Given the description of the IMU-based dynamical system via Eq. (12), the dynamical description of the IMU model parameters via Eq. (14), and a statistical description of the IMU model parameters that appear in Eq. (13), the objective is to determine a statistical description of the

position, velocity, and attitude of the IMU at some later time. While there is no tractable exact solution for predicting these statistics, in general, several approaches exist for approximating the statistics. One method is the Monte Carlo process. The Monte Carlo process effectively performs a large number of simulations of the dynamical system, with each simulation drawing new samples from every statistical distribution involved in the system. As such, the accuracy of a Monte Carlo process is limited by the number of simulations considered, which is, in turn, limited by the computational time allotted. On the other hand, the Monte Carlo process provides a very flexible environment that is capable of handling a wide variety of dynamical systems and statistical distributions.

To illustrate the Monte Carlo process, consider the two dimensional transformation from polar to Cartesian coordinates, i.e.

$$\begin{bmatrix} x \\ y \end{bmatrix} = r \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}.$$

Given a distribution on $\boldsymbol{w} = [r \quad \theta]$, it is desired to determine distribution characteristics on $\boldsymbol{z} = [x \;\; y]$. First, assume that r and θ are independent and Gaussian with mean and covariance

$$\boldsymbol{m_w} = \begin{bmatrix} 1 \text{ [m]} \\ 60 \text{ [deg]} \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{P_w} = \begin{bmatrix} (0.02 \text{ [m]})^2 & 0 \\ 0 & (30 \text{ [deg]})^2 \end{bmatrix}.$$

Therefore, the polar to Cartesian conversion may be written as

$$\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{w}),$$

where $\boldsymbol{w}$ is the input and $\boldsymbol{z}$ is the associated output. The Monte Carlo simulation process is started by drawing a number of samples from the input vector, where the sampling is performed with respect to the probability density function which governs the characteristics of the input. Once the samples are drawn, the nonlinear mapping function, $\boldsymbol{f}$, is applied to each input sample to generate a set of output samples.

For the coordinate conversion problem, the Monte Carlo simulation is initialized by drawing 1×10^5 realizations of the random vector $\boldsymbol{w}$ from the distribution

$$p(\boldsymbol{w}; \boldsymbol{m_w}, \boldsymbol{P_w}) = |2\pi \boldsymbol{P_w}|^{-1/2} \exp\left\{ -\frac{1}{2} (\boldsymbol{w} - \boldsymbol{m_w})^T \boldsymbol{P_w}^{-1} (\boldsymbol{w} - \boldsymbol{m_w}) \right\}.$$

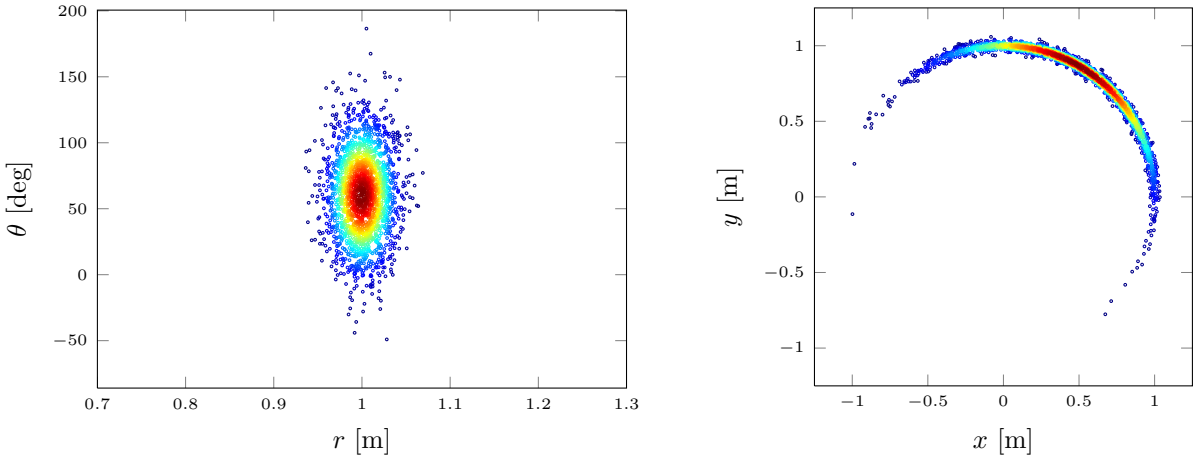
The set of samples drawn is shown in Fig. (8(a)). The nonlinear transformation is applied to each sample in the input space and the resulting set of Cartesian coordinates is shown in Fig. (8(b)). It is clear that the output space is not capable of being described by a Gaussian distribution; however, the Monte Carlo simulation technique can still be used to compute an accurate approximation of the mean and covariance via

$$\boldsymbol{m_z} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{z}_i \qquad \text{and} \qquad \boldsymbol{P_z} = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{z}_i - \boldsymbol{m_z})(\boldsymbol{z}_i - \boldsymbol{m_z})^T, \tag{15}$$

where

$$\boldsymbol{z}_i = \boldsymbol{f}(\boldsymbol{w}_i)$$

and $\boldsymbol{w}_i$ is the i^{th} Monte Carlo sample. The contours representing the Gaussian distribution computed from the preceding mean and covariance equations are shown on top of the set of output-space

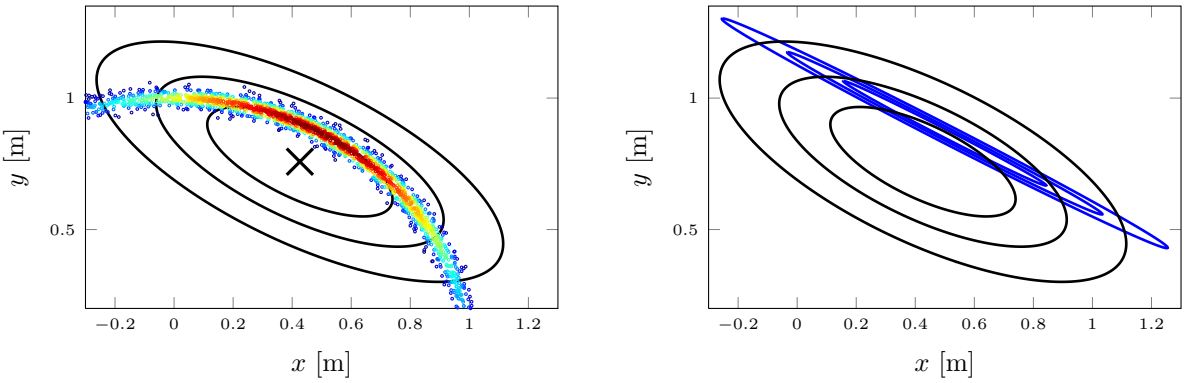


(a) Input Space Monte Carlo Samples

(b) Mapped Samples in the Output Space

Figure 8. Input and Output Samples from the Monte Carlo Simulation

Monte Carlo samples in Fig. (9(a)). Moreover, a linearization approach can be used to map the input mean and covariance into an approximate output mean and covariance. The comparison between the linear mean and covariance and the Monte Carlo mean and covariance is shown in Fig. (9(b)), wherein it is seen that the two differ a great deal. The only characteristic that is mostly captured by the linearization process is the orientation of the covariance matrix. The centering (i.e. the mean of the distribution) is not captured accurately, and the size of the mapped covariance matrix is not captured accurately.



(a) Monte Carlo Mean and Covariance

(b) Comparison between Linearization and Monte Carlo

Figure 9. Distribution Contours Obtained by the Mean and Covariance from the Monte Carlo Simulation and Linearization

To further demonstrate the power of Monte Carlo simulation, consider a modification of the preceding problem. In this case, it is still assumed that the radial and angular input coordinates are independent, but now the radial direction is taken to be Gaussian distributed and the angular direction is taken to be uniformly distributed. The mean and covariance are taken to be the same as before so that the input space has the same first and second central moments as in the preceding

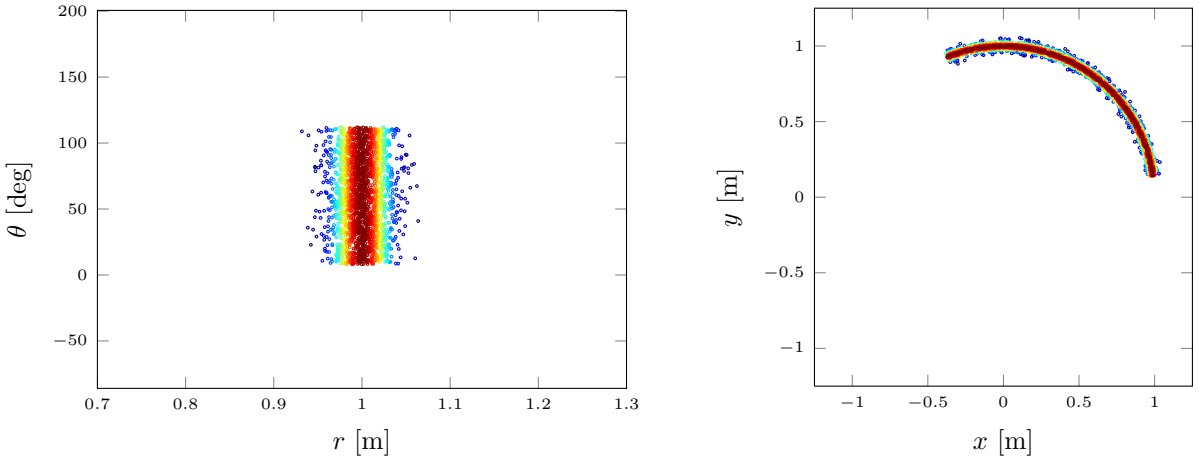
example. The probability density function for the input is now given by

$$p(\boldsymbol{w}; m_r, P_r, m_\theta, P_\theta) = |2\pi P_r|^{-1/2} \exp\left\{-\frac{(w_r - m_r)^2}{2P_r}\right\} \cdot \frac{1}{2\sqrt{3P_\theta}}$$
$$= \frac{1}{2\sqrt{6\pi P_r P_\theta}} \exp\left\{-\frac{(w_r - m_r)^2}{2P_r}\right\},$$

where $m_\theta - \sqrt{3P_\theta} \leq \theta \leq m_\theta + \sqrt{3P_\theta}$, such that $[m_\theta - \sqrt{3P_\theta}\ ,\ m_\theta + \sqrt{3P_\theta}]$ is the support set of the uniform angular distribution. From the original specifications for the transformation problem, the means and covariances required to evaluate the input-space distribution are

$$m_r = 1\ [\text{m}]\,, \quad m_\theta = 60\ [\text{deg}]\,, \quad P_r = (0.02\ [\text{m}])^2\,, \quad \text{and} \quad P_\theta = (30\ [\text{deg}])^2\,.$$

As before, the Monte Carlo simulation is initialized by drawing 1×10^5 realizations of the random vector $\boldsymbol{w}$ from the distribution $p(\boldsymbol{w}; m_r, P_r, m_\theta, P_\theta)$, and each sample is mapped into the output space via the nonlinear transformation. These samples are shown in Figs. (10(a)) and (10(b)), respectively. The mean and covariance is again computed from the Monte Carlo samples and



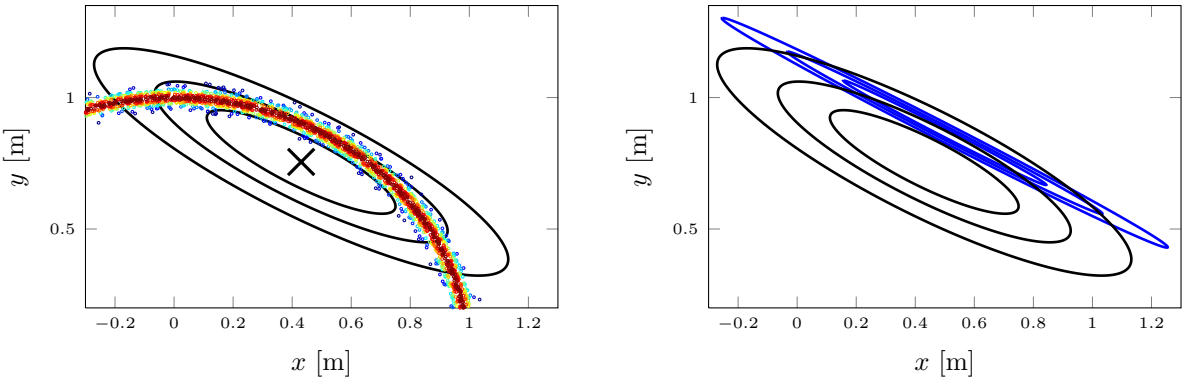
(a) Input Space Monte Carlo Samples

(b) Mapped Samples in the Output Space

Figure 10. Input and Output Samples from the Monte Carlo Simulation (modified problem)

shown along with the output-space samples in Fig. (11(a)). Finally, a linearization approach is used to map the input-space mean and covariance into an output-space mean and covariance. The comparison between the Monte Carlo mean and covariance and the linearization mean and covariance is shown in Fig. (11(b)).

While it is clear that the linearization process once again fails to capture the true mean and covariance, it is noted that the linearized mean and covariance depicted in Fig. (11(b)) are identical to the ones depicted in Fig. (9(b)). At the same time, however, the Monte Carlo means and covariances differ between these two simulations. This occurs because the Monte Carlo simulation has and uses knowledge of the entire distribution on both the input and output spaces whereas a linearization procedure works with only the first and second central moments of the distribution. Therefore if two input distributions have the same mean and covariance, and are subjected to the same nonlinear transformation, the linearized solutions will always have the same output mean and covariance.



(a) Monte Carlo mean and covariance

(b) Comparison between linearization and Monte Carlo

Figure 11. Distribution Contours Obtained by the Mean and Covariance from the Monte Carlo Simulation and Linearization (modified problem)

It is also worth noting that the Monte Carlo simulation enables analysis extending beyond the computation of the mean and covariance. For instance, higher-order statistical moments may be computed in a similar fashion to the mean and covariance, best case and worst case performance can be analyzed through analysis of the dispersions away from a predefined truth, or the radial distance encapsulating a given percentage of the sample points can be computed.

The problem of analyzing the performance of IMU-based navigation systems is characterized by complex, nonlinear functions where the input space is described by the IMU error sources, which may have fairly arbitrary statistical descriptions. Monte Carlo analysis naturally handles the characteristics of this problem and enables a more rich data analysis suite than is found in typical linearization-based analysis.

3.6 External Aiding

In this section, models for three different external sensors available within SAIMUN are described. The sensors presented are

1. range/range-rate,
2. stellar line-of-sight, and
3. bearing angles.

For each sensor considered, the measurement model is described for each of the previously described IMU mechanizations. Additionally, to facilitate data processing, the measurement Jacobian for each configuration is described.

3.6.1 Range/Range-Rate Measurements. The range ρ, by definition is the difference magnitude between two position vectors. For this case, the two position vectors will be the vehicle's position in a given frame $\boldsymbol{r}_s$, and the position vector of a transmission receiving ground station $\boldsymbol{r}_g$ in the same frame. $\boldsymbol{r}_g$ is assumed to be known.

Strapdown The range measurement, z_ρ, is modeled as the range between an onboard sensor and a known reference position that is corrupted by additive, zero-mean, white measurement noise; the corresponding model is

$$z_\rho = \sqrt{(\boldsymbol{r}^i_{s/g})^T(\boldsymbol{r}^i_{s/g})} + v_\rho \,,$$

where v_ρ is the measurement noise and $\boldsymbol{r}_{s/g}$ is the true position of the range sensor with respect to the known ground station, which is given by

$$\boldsymbol{r}^i_{s/g} = \boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{T}^i_b \boldsymbol{r}^b_{s/\text{IMU}} - \boldsymbol{T}^i_f \boldsymbol{r}^f_g \,.$$

A reference measurement may be determined by evaluating the measurement model at a reference state, leading to $\hat{z}_\rho$. The deviation between the true and reference measurements is then considered as

$$\delta z_\rho = z_\rho - \hat{z}_\rho \,.$$

This deviation is then linearized using a first-order Taylor series expansion, and the result may be expressed as

$$\delta z_\rho = \boldsymbol{H}_{\rho,r} \delta \boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{H}_{\rho,\theta} \delta \boldsymbol{\theta} + v_\rho \,,$$

where $\delta \boldsymbol{r}^i_{\text{IMU}}$ is the error between the true and reference IMU positions and $\delta\boldsymbol{\theta}$ is the attitude error, which is defined to be twice the vector part of the quaternion error. This illustrates that the range sensor measurement model has sensitivity associated with the position of the IMU and the attitude. This means that direct updates can be obtained for these states when considering range measurements from a known ground station.

Similar to the range measurement, the range-rate measurement produces a measurement of the range-rate between an onboard sensor and a known ground station. This is represented by $z_{\dot\rho}$, where

$$z_{\dot\rho} = \frac{(\boldsymbol{r}^i_{s/g})^T(\boldsymbol{v}^i_{s/g})}{\sqrt{(\boldsymbol{r}^i_{s/g})^T(\boldsymbol{r}^i_{s/g})}} + v_{\dot\rho} \,,$$

where $v_{\dot\rho}$ is zero mean, white, range-rate measurement noise, $\boldsymbol{r}_{s/g}$ is the position between the sensor and the ground station, and the velocity of the range-rate sensor with respect to the ground station are

$$\boldsymbol{v}^i_{s/g} = \boldsymbol{v}^i_{\text{IMU}} + \boldsymbol{T}^i_b [\boldsymbol{\omega}^b_{b/i}\times] \boldsymbol{r}^b_{s/\text{IMU}} - \boldsymbol{T}^i_f [\boldsymbol{\omega}^f_{f/i}\times] \boldsymbol{r}^f_g \,.$$

As with the range measurement, a reference measurement may be determined by evaluating the measurement model at a reference state, leading to $\hat{z}_{\dot\rho}$. The deviation between the true and reference measurements is then considered as

$$\delta z_{\dot\rho} = z_{\dot\rho} - \hat{z}_{\dot\rho} \,.$$

This deviation is then linearized using a first-order Taylor series expansion, and the result may be expressed as

$$\delta z_{\dot\rho} = \boldsymbol{H}_{\dot\rho,r} \delta \boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{H}_{\dot\rho,v} \delta \boldsymbol{v}^i_{\text{IMU}} + \boldsymbol{H}_{\dot\rho,\theta} \delta\boldsymbol{\theta} + \boldsymbol{H}_{\dot\rho,b_g} \delta \boldsymbol{b}_g + \boldsymbol{H}_{\dot\rho,m_g} \delta \boldsymbol{m}_g + \boldsymbol{H}_{\dot\rho,n_g} \delta \boldsymbol{n}_g + \boldsymbol{H}_{\dot\rho,s_g} \delta \boldsymbol{s}_g + v_{\dot\rho} \,,$$

where $\delta \boldsymbol{v}^i_{\text{IMU}}$ is the error between the true and reference IMU velocities, $\delta \boldsymbol{b}_g$ is the gyro bias error, $\delta \boldsymbol{m}_g$ is the gyro misalignment error, $\delta \boldsymbol{n}_g$ is the gyro nonorthogonality error, and $\delta \boldsymbol{s}_g$ is the gyro scale factor error. Thus, the range-rate measurement has sensitivity associated with the position of the IMU, the velocity of the IMU, the attitude of the vehicle, and the gyro parameters.

Space Stabilized The range measurement for the space stabilized case is nearly the same as for the strapdown case; therefore, the model can be written as

$$z_\rho = \sqrt{(\boldsymbol{r}^i_{s/g})^T(\boldsymbol{r}^i_{s/g})} + v_\rho \,,$$

where v_ρ is zero-mean, white, measurement noise. The term $\boldsymbol{r}_{s/g}$ still represents the position of the range sensor with respect to the known ground station, but is slightly changed from the strapdown case, such that is it now given by

$$\boldsymbol{r}^i_{s/g} = \boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{T}^i_p \boldsymbol{T}^p_b \boldsymbol{r}^b_{s/\text{IMU}} - \boldsymbol{T}^i_f \boldsymbol{r}^f_g$$

A reference measurement may be determined by evaluating the measurement model at a reference state, leading to $\hat{z}_\rho$. The deviation between the true and reference measurements is then considered as

$$\delta z_\rho = z_\rho - \hat{z}_\rho \,.$$

This deviation is then linearized using a first-order Taylor series expansion, and the result may be expressed as

$$\delta z_\rho = \boldsymbol{H}_{\rho,r}\delta\boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{H}_{\rho,\theta}\delta\boldsymbol{\theta} + v_\rho \,.$$

It is important to note that the attitude error, $\delta\boldsymbol{\theta}$, which is still defined to be twice the vector part of the quaternion error, now defines the attitude error of the platform, as opposed to of the body.

As with the space stabilized range measurement, the range-rate measurement model is largely unchanged by the IMU mechanization; therefore, the model may be written as

$$z_{\dot\rho} = \frac{(\boldsymbol{r}^i_{s/g})^T(\boldsymbol{v}^i_{s/g})}{\sqrt{(\boldsymbol{r}^i_{s/g})^T(\boldsymbol{r}^i_{s/g})}} + v_{\dot\rho} \,,$$

where $v_{\dot\rho}$ is zero mean, white, range-rate measurement noise, and the position and velocity relative to the ground station are

$$\begin{aligned}
\boldsymbol{r}^i_{s/g} &= \boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{T}^i_p \boldsymbol{T}^p_b \boldsymbol{r}^b_{s/\text{IMU}} - \boldsymbol{T}^i_f \boldsymbol{r}^f_g \\
\boldsymbol{v}^i_{s/g} &= \boldsymbol{v}^i_{\text{IMU}} + \Big(\boldsymbol{T}^i_p[\boldsymbol{\omega}^p_{p/i}\times]\boldsymbol{T}^p_b + \boldsymbol{T}^i_p\boldsymbol{T}^p_b[\boldsymbol{\omega}^b_{b/p}\times]\Big)\boldsymbol{r}^b_{s/imu} - \boldsymbol{T}^i_f[\boldsymbol{\omega}^f_{f/i}\times]\boldsymbol{r}^f_g \,.
\end{aligned}$$

The deviation between the true and reference measurements is then considered as

$$\delta z_\rho = z_{\dot\rho} - \hat{z}_{\dot\rho} \,,$$

where $\hat{z}_{\dot\rho}$ is the reference measurement. This deviation is then linearized using a first-order Taylor series expansion, and the result may be expressed as

$$\delta z_{\dot\rho} = \boldsymbol{H}_{\dot\rho,r}\delta\boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{H}_{\dot\rho,v}\delta\boldsymbol{v}^i_{\text{IMU}} + \boldsymbol{H}_{\dot\rho,\theta}\delta\boldsymbol{\theta} + \boldsymbol{H}_{\dot\rho,b_g}\delta\boldsymbol{b}_g + \boldsymbol{H}_{\dot\rho,m_g}\delta\boldsymbol{m}_g + \boldsymbol{H}_{\dot\rho,n_g}\delta\boldsymbol{n}_g + \boldsymbol{H}_{\dot\rho,s_g}\delta\boldsymbol{s}_g + v_{\dot\rho} \,.$$

As with the range measurement for the space stabilized IMU mechanization, the attitude error, $\delta\boldsymbol{\theta}$, represents the platform attitude error. It should also be noted that the exact terms appearing in the preceding first-order expansion are, in general, different than those found for the strapdown range-rate measurement.

Data Processing When considering both range and range-rate measurements, the combined measurement and reference measurement are

$$\boldsymbol{z}_{\rho,\dot{\rho}} = \begin{bmatrix} z_\rho \\ z_{\dot{\rho}} \end{bmatrix} \qquad \text{and} \qquad \hat{\boldsymbol{z}}_{\rho,\dot{\rho}} = \begin{bmatrix} \hat{z}_\rho \\ \hat{z}_{\dot{\rho}} \end{bmatrix}.$$

In order to process these measurements, it is required to determine the Jacobian of the combined measurement. The Jacobian for the combined measurements is simply given by the concatenation of the individual Jacobians, or

$$\boldsymbol{H}_{\rho,\dot{\rho}} = \begin{bmatrix} \boldsymbol{H}_{\rho,r} & \boldsymbol{0}_{3\times3} & \boldsymbol{H}_{\rho,\theta} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times12} \\ \boldsymbol{H}_{\dot{\rho},r} & \boldsymbol{H}_{\dot{\rho},v} & \boldsymbol{H}_{\dot{\rho},\theta} & \boldsymbol{H}_{\dot{\rho},b} & \boldsymbol{H}_{\dot{\rho},m} & \boldsymbol{H}_{\dot{\rho},n} & \boldsymbol{H}_{\dot{\rho},s} & \boldsymbol{0}_{3\times9} \end{bmatrix}.$$

3.6.2 Stellar Line-of-Sight Measurements. Star tracking measurement systems utilize a camera and the knowledge provided by star field databases to assist in attitude determination. For simplicity, the measurements considered here are unit vectors to each star in the star camera field of view. However, this solution suffers from the fact that the covariance matrix of a unit vector, as well as the addition used in the additive update, are not well defined. Two alternative approaches, focal plane angles (FPA) and quaternion estimation (QUEST), avoid such issues, and should be considered for future investigation.

Strapdown The model of the star camera measurement is given as

$$\boldsymbol{z}_{sc} = \boldsymbol{T}_b^{sc}\boldsymbol{T}_i^b\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr} + \boldsymbol{v}_{sc}^{sc}$$

where $\boldsymbol{T}_b^{sc}$ is the rotation matrix relating the body and star camera reference frames, $\boldsymbol{T}_i^b$ is the rotation matrix relating the inertial and body frames, $\boldsymbol{T}_{sr}^i$ is the rotation matrix relating the stellar reference and inertial frames, $\boldsymbol{v}_{sc}^{sc}$ is zero-mean, white, measurement noise, and $\boldsymbol{u}_{s/sc}^{sr}$ is the unit vector representation of the line-of-sight to a reference star, which can be expressed in terms of right ascension (α^{sr}) and declination (δ^{sr}) as

$$\boldsymbol{u}_{s/sc}^{sr} = \begin{bmatrix} \cos\delta^{sr}\cos\alpha^{sr} \\ \cos\delta^{sr}\sin\alpha^{sr} \\ \sin\delta^{sr} \end{bmatrix}.$$

Similarly, the reference star camera measurement can be written as

$$\hat{\boldsymbol{z}}_{sc}^{sc} = \boldsymbol{T}_b^{sc}\hat{\boldsymbol{T}}_i^b\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr}.$$

The deviation of the star camera measurement, which is the difference between the true and reference measurements, can be expressed, after a first-order Taylor series expansion, as

$$\delta\boldsymbol{z}_{sc}^{sc} = \boldsymbol{T}_b^{sc}\left[\hat{\boldsymbol{T}}_i^b\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr}\times\right]\delta\boldsymbol{\theta} + \boldsymbol{v}_{sc}^{sc},$$

where $\delta\boldsymbol{\theta}$ is the attitude error, and it is noted that this represents the attitude error for the body-fixed reference frame.

Space Stabilized The space stabilized mechanization model is nearly identical to the strapdown mechanization model, except that an extra rotation must be introduced to account for the appearance of the platform frame. Thus, the measurement generated by the star camera is modeled as

$$\boldsymbol{z}_{sc}^{sc} = \boldsymbol{T}_b^{sc}\boldsymbol{T}_p^b\boldsymbol{T}_i^p\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr} + \boldsymbol{v}_{sc}^{sc}\,,$$

and the reference measurement is found via

$$\hat{\boldsymbol{z}}_{sc}^{sc} = \boldsymbol{T}_b^{sc}\boldsymbol{T}_p^b\hat{\boldsymbol{T}}_i^p\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr}\,.$$

As with the strapdown case, the difference between the measurement and reference measurement is defined to be a measurement deviation, $\delta\boldsymbol{z}_{sc}^{sc}$, and this deviation is expanded in a first-order Taylor series to yield

$$\delta\boldsymbol{z}_{sc}^{sc} = \boldsymbol{T}_b^{sc}\boldsymbol{T}_p^b\big[\hat{\boldsymbol{T}}_i^p\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr}\times\big]\delta\boldsymbol{\theta} + \boldsymbol{v}_{sc}^{sc}\,,$$

where $\delta\boldsymbol{\theta}$ is the attitude error, and it is noted that this represents the attitude error for the platform-fixed reference frame.

Data Processing In order to process stellar line-of-sight measurements, it is required to determine the Jacobian of the measurement. The Jacobian for the line-of-sight measurements takes on the same form, independent of the IMU mechanization, such that it can be written as

$$\boldsymbol{H}_{sc} = \begin{bmatrix}\boldsymbol{0}_{3\times6} & \boldsymbol{H}_\theta & \boldsymbol{0}_{3\times24}\end{bmatrix}\,,$$

where

$$\boldsymbol{H}_\theta = \boldsymbol{T}_b^{sc}\big[\hat{\boldsymbol{T}}_i^b\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr}\times\big]$$

for the strapdown IMU mechanization and

$$\boldsymbol{H}_\theta = \boldsymbol{T}_b^{sc}\boldsymbol{T}_p^b\big[\hat{\boldsymbol{T}}_i^p\boldsymbol{T}_{sr}^i\boldsymbol{u}_{s/sc}^{sr}\times\big]$$

for the space stabilized IMU mechanization.

3.6.3 Bearing Angles Measurements. A coordinate system is defined such that the y axis points along the "nose" of the vehicle, the z axis points "up," and the x axis, which completes the triad, points "right.". The bearing angles measurements are measurements of the line-of-sight to a destination target, where the angles are taken to be the elevation and azimuth angles. The relative position of the destination target with respect to the bearing angles sensor is denoted by $\boldsymbol{\rho}_{t/s}$ (or simply $\boldsymbol{\rho}$, for ease of exposition), where the destination target is assumed to be known in the fixed frame. The elevation angle is measured on the interval $[-\pi/2,\ \pi/2)$ and is calculated as

$$z_\epsilon = \sin^{-1}(\rho_z/|\boldsymbol{\rho}|) + v_\epsilon\,,$$

where ρ_z is the z-component in $\boldsymbol{\rho}$ and v_ϵ is zero-mean, white, measurement noise. Care should be taken to avoid singularities as the target is approached when $|\boldsymbol{\rho}| = 0$. If so, then $z_\epsilon = 0$. The azimuth angle is measured on the interval $[0, 2\pi]$ and is found via

$$z_\alpha = \text{atan2}(\rho_y, \rho_x) + v_\alpha\,,$$

where atan2 is the two-argument arc-tangent function, ρ_x and ρ_y are the x and y components of the relative position, $\boldsymbol{\rho}$, and v_α is zero-mean, white, measurement noise.

Reference measurements can be determined by evaluating the elevation and azimuth angle models at a reference state. These reference measurements are then used to define measurement deviations as the difference between the measurement and its reference. A first-order Taylor series expansion for the deviations in the elevation and azimuth angles yields

$$\delta z_\epsilon = \frac{\partial \epsilon}{\partial \boldsymbol{\rho}} \delta \boldsymbol{\rho} + v_\epsilon \qquad \text{and} \qquad \delta z_\alpha = \frac{\partial \alpha}{\partial \boldsymbol{\rho}} \delta \boldsymbol{\rho} + v_\alpha \,,$$

where $(\partial \epsilon / \partial \boldsymbol{\rho})$ and $(\partial \alpha / \partial \boldsymbol{\rho})$ are simple derivatives of the trigonometric functions and $\delta \boldsymbol{\rho}$ is the error in the relative position vector with respect to the reference. Depending on the IMU mechanization, this error changes form.

Strapdown For the non-gimbaled mechanization, the relative position vector, in inertial coordinates, to the destination target from the sensor is given by

$$\boldsymbol{\rho}^i_{t/s} = \boldsymbol{T}^i_f \boldsymbol{r}^f_t - (\boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{T}^i_b \boldsymbol{r}^b_{s/\text{IMU}}) \,,$$

which can be used to determine the error in the relative position as

$$\delta \boldsymbol{\rho}^i_{t/s} = -\delta \boldsymbol{r}^i_{\text{IMU}} + \hat{\boldsymbol{T}}^i_b [\boldsymbol{r}^b_{s/\text{IMU}} \times] \delta \boldsymbol{\theta} \,,$$

where $\delta \boldsymbol{\theta}$ is the attitude error, which is the error of the body-fixed frame.

Space Stabilized With a gimbaled, space stabilized, platform, the relative position vector of the destination target with respect to the sensor requires the added transformation between the platform and the body frame. The result is given by

$$\boldsymbol{\rho}^i_{t/s} = \boldsymbol{T}^i_f \boldsymbol{r}^f_t - (\boldsymbol{r}^i_{\text{IMU}} + \boldsymbol{T}^i_p \boldsymbol{T}^p_b \boldsymbol{r}^b_{s/\text{IMU}}) \,.$$

Evaluating a reference relative position vector, determining the error, and performing a first-order Taylor series expansion yields

$$\delta \boldsymbol{\rho}^i_{t/s} = -\delta \boldsymbol{r}^i_{\text{IMU}} + \hat{\boldsymbol{T}}^i_p [\boldsymbol{T}^p_b \boldsymbol{r}^b_{s/\text{IMU}} \times] \delta \boldsymbol{\theta}$$

where $\delta \boldsymbol{\theta}$ is the attitude error, which is the error of the platform-fixed frame.

Data Processing In order to process the bearing, it is required to determine the Jacobian of the measurement, which takes on the same form, independent of the IMU mechanization, such that it can be written as

$$\boldsymbol{H}_{\epsilon,\alpha} = \begin{bmatrix} \boldsymbol{H}_{\epsilon,r} & \boldsymbol{0}_{1\times 3} & \boldsymbol{H}_{\epsilon,\theta} & \boldsymbol{0}_{1\times 24} \\ \boldsymbol{H}_{\alpha,r} & \boldsymbol{0}_{1\times 3} & \boldsymbol{H}_{\alpha,\theta} & \boldsymbol{0}_{1\times 24} \end{bmatrix} \,,$$

where

$$\boldsymbol{H}_{\epsilon,r} = -(\partial \epsilon / \partial \boldsymbol{\rho}) \qquad \text{and} \qquad \boldsymbol{H}_{\alpha,r} = -(\partial \alpha / \partial \boldsymbol{\rho})$$

for both of the IMU mechanizations,

$$\boldsymbol{H}_{\epsilon,\theta} = (\partial \epsilon / \partial \boldsymbol{\rho}) \hat{\boldsymbol{T}}^i_b [\boldsymbol{r}^b_{s/\text{IMU}} \times] \qquad \text{and} \qquad \boldsymbol{H}_{\alpha,r} = (\partial \alpha / \partial \boldsymbol{\rho}) \hat{\boldsymbol{T}}^i_b [\boldsymbol{r}^b_{s/\text{IMU}} \times]$$

for the strapdown IMU mechanization, and

$$\boldsymbol{H}_{\epsilon,\theta} = (\partial \epsilon / \partial \boldsymbol{\rho}) \hat{\boldsymbol{T}}^i_p [\boldsymbol{T}^p_b \boldsymbol{r}^b_{s/\text{IMU}} \times] \qquad \text{and} \qquad \boldsymbol{H}_{\alpha,r} = (\partial \alpha / \partial \boldsymbol{\rho}) \hat{\boldsymbol{T}}^i_p [\boldsymbol{T}^p_b \boldsymbol{r}^b_{s/\text{IMU}} \times]$$

for the space stabilized IMU mechanization.

3.7 Linear Covariance Analysis

Recall from Section 3.5 that either of the mechanizations of the IMU can be represented by a nonlinear dynamical system; provided that the position, velocity, and attitude are collected into a state vector, $\boldsymbol{x}$, the general nonlinear dynamical system representing the forward evolution of the state for either of the IMU systems is given by

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{a}_k, \boldsymbol{\omega}_k, t_k)\,, \tag{16}$$

where $\boldsymbol{a}_k$ and $\boldsymbol{\omega}_k$ represent the true vehicle non-gravitational acceleration and angular velocity, respectively. For the mechanizations presented in Section 3.4, $\boldsymbol{f}(\cdot)$ represents one of Eqs. (10) or Eqs. (11). In addition, the true vehicle non-gravitational acceleration and angular velocity are each corrupted by a set of errors ($\boldsymbol{\phi}_k$ for the non-gravitational acceleration and $\boldsymbol{\psi}_k$ for the angular velocity) to produce the measurements of the non-gravitational acceleration and angular velocity, which is functionally represented by Eq. (13). These parameters can also evolve in time, as dictated by Eqs. (14).

Given the dynamical systems models along with a representation of the statistics of the quantities involved, it is possible to investigate the evolution of the uncertainty for the position, velocity, and attitude of the vehicle. As discussed in Section 3.5, one method for investigating this uncertainty evolution is through the use of Monte Carlo simulation. The Monte Carlo method, while highly flexible, requires significant computational resources. An alternative method, which has the advantage of being less computationally demanding, is to use a Kalman filtering-based approach.

The particular Kalman filtering-based approach used here is known as linear covariance analysis. In linear covariance analysis, the nonlinear systems describing the evolution of the IMU-driven systems are linearized about a reference truth trajectory. This has the immense advantage of decoupling the uncertainty propagation from any state propagation method, which means that only the covariance needs to be propagated in order to determine the uncertainty of the position, velocity, and attitude of the vehicle. In order to incorporate the uncertainties present in the IMU error sources, an augmented state vector, $\boldsymbol{x}_{\text{aug}}$, is defined to be the standard state vector, $\boldsymbol{x}$, concatenated with all of the accelerometer and gyroscope error parameters except those that are characterized as white-noise sequences. All of the white-noise error parameters are collected together as the vector $\boldsymbol{w}_{\text{aug}}$. With these definitions, a new nonlinear dynamical system can be constructed for the augmented state vector, which is given by

$$\boldsymbol{x}_{\text{aug},k} = \boldsymbol{f}_{\text{aug}}(\boldsymbol{x}_{\text{aug},k-1}, \boldsymbol{w}_{\text{aug},k-1})\,. \tag{17}$$

where $\boldsymbol{x}_{\text{aug},k}$ and $\boldsymbol{x}_{\text{aug},k-1}$ are the (augmented) state at times t_k and t_{k-1} respectively.

This dynamical system describes the evolution of the position, velocity, and attitude of the vehicle, as well as the non-white-noise error parameters associated with the IMU. The nonlinear equation in Eq. (17) is then linearized about some reference truth, $\boldsymbol{x}_k^*$, similar to how the extended Kalman filter is linearized about the current state estimate. The resulting linearized dynamical system is used to construct the covariance propagation equation

$$\boldsymbol{P}_k = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1}\boldsymbol{F}_{k-1}^T + \boldsymbol{M}_{k-1}\boldsymbol{Q}_{k-1}\boldsymbol{M}_{k-1}^T\,, \tag{18}$$

where $\boldsymbol{P}_k$ represents the covariance at time t_k. In Eq. (18), $\boldsymbol{F}_{k-1}$ is the tangent linear dynamics evaluated along the reference state and nominal IMU model parameters, $\boldsymbol{M}_{k-1}$ is the Jacobian of the state with respect to the white-noise sequence parameters, and $\boldsymbol{Q}_{k-1}$ is the process noise covariance matrix that represents the statistics of the white-noise sequence. This covariance, represented by Eq. (18), represents the covariance of the full augmented state vector; to investigate and analyze

the uncertainties characterizing the position, velocity, and attitude of the vehicle, the corresponding elements can be extracted.

The linear covariance approach can also be used to investigate the effects of acquiring external sensor data, such as that from a GPS sensor, an altimeter, a range sensor, or a magnetometer, among others. External data is generally employed to reduce the uncertainty that is present in the prediction of the translational and rotational states of a vehicle. By combining external data with IMU data, lower uncertainties can be obtained; however, this means relying on data regarding the external environment, which can lead to sensitivities in data quality and reliability. The external data are, in general, taken to be of the form

$$\boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{x}_{\text{aug},k}) + \boldsymbol{v}_k \,,$$

where $\boldsymbol{v}_k$ represents the measurement noise of the sensor at time t_k, which is taken to be zero mean with covariance $\boldsymbol{R}_k$. When presented with this external measurement data, the linear covariance method updates the covariance through a blending of the *a priori* state information and the newly acquired external measurement data via

$$\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\,\boldsymbol{P}_k^-\,(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)^T + \boldsymbol{K}\boldsymbol{R}_k\boldsymbol{K}^T \,,$$

where $\boldsymbol{K}_k$ is the Kalman gain, which is given by

$$\boldsymbol{K}_k = \boldsymbol{P}_k^-\boldsymbol{H}_k^T\left(\boldsymbol{H}_k\boldsymbol{P}_k^-\boldsymbol{H}_k^T + \boldsymbol{R}_k\right)^{-1} .$$

Whereas the nonlinear dynamics function, $\boldsymbol{f}_{\text{aug}}(\cdot)$, and the tangent linear dynamics matrix, $\boldsymbol{F}_{k-1}$, are key in the propagation stage of the linear covariance method, the nonlinear measurement function, $\boldsymbol{h}(\cdot)$ and its associated tangent linear matrix, $\boldsymbol{H}_k$, are critical for the update stage of the linear covariance method.

4.0 RESULTS AND DISCUSSION

4.1 Trajectory Simulation

In order to provide the trajectory input to SAIMUN that is outlined in Section 3.1, a ground vehicle was used to log acceleration and angular velocity data using a VectorNav VN-100[1] IMU. The VN-100 logged data at 40 Hz during a trajectory that took approximately 41 minutes to complete. A smoother was implemented to decrease the effect of noise on the recorded acceleration and angular velocity data. The smoothed acceleration and angular velocity data were dead-reckoned from a known starting condition to provide the true trajectory for SAIMUN to use. The dead-reckoned trajectory is not perfectly representative of the trajectory followed by the vehicle; however, for the purposes of applying SAIMUN and analyzing IMU-based navigation performance, the dead-reckoned trajectory is taken to be the true trajectory.

4.2 IMU Selection

To demonstrate the utility of SAIMUN, two inertial measurement units are compared: the Lord Microstrain 3DM-GX3-15[2] and the Analog Devices ADIS16488[3]. Both IMUs utilize MEMS sensors

[1] http://www.vectornav.com/Downloads/Support/PB-12-0002.pdf, accessed 04/21/2014
[2] http://files.microstrain.com/3DM-GX3-15-Inertial-Measurement-Unit.pdf, accessed 04/21/2014
[3] http://www.analog.com/static/imported-files/data_sheets/ADIS16488.pdf, accessed 04/21/2014

to provide triaxial accelerometers, gyroscopes, magnetometers, and a pressure sensor. The IMUs are implemented to provide dead-reckoning navigation of a ground vehicle trajectory.

For this analysis, only the accelerometer and gyroscopic information is of interest. Table 1 shows the pertinent specifications for both of the chosen IMUs. These IMUs are interesting to compare due to the differences between the random walk and bias instability specifications. Table 1 shows that the ADIS16488 has a higher random walk specification and thus more white noise present in the signal. The 3DM-GX3-15, on the other hand, has a higher bias instability specification. The higher bias instability and lower random walk specifications of the 3DM-GX3-15 compared to the ADIS16488 imply that the position and attitude uncertainty associated with the 3DM-GX3-15 will grow more slowly than the ADIS16488 initially. Eventually, the position, velocity, and attitude uncertainty associated with the 3DM-GX3-15 will grow larger and surpass their counterparts associated with the ADIS16488 as time increases.

Table 1. IMU Specifications for the ADIS16488 and 3DM-GX3-15

	Specification	3DM-GX3-15	ADIS16488
Accelerometers	Analog-to-Digital Converter Bitrate	16	32
	Range	± 156.96 $\frac{\text{m}}{\text{s}}$	± 176.58 $\frac{\text{m}}{\text{s}}$
	Scale Factor Error	$\mathcal{U}(-500, 500)$ ppm	$\mathcal{U}(-5000, 5000)$ ppm
	Axes Nonorthogonality Error	$\mathcal{U}(-103, 103)$ arcsec	$\mathcal{U}(-126, 126)$ arcsec
	Axes Misalignment Error	$\mathcal{U}(-103, 103)$ arcsec	$\mathcal{U}(-3600, 3600)$ arcsec
	Velocity Random Walk	$0.0007848 \frac{\text{m/s}}{\sqrt{\text{hr}}}$	0.029 $\frac{\text{m/s}}{\sqrt{\text{hr}}}$
	Bias Instability	0.0003924 $\frac{\text{m}}{\text{s}}$	0.000981 $\frac{\text{m}}{\text{s}}$
	Bias Instability Time	100 s	100 s
	Startup Bias	$\mathcal{U}(-0.0196, 0.0196)$ $\frac{\text{m}}{\text{s}}$	$\mathcal{U}(-0.016, 0.016)$ $\frac{\text{m}}{\text{s}}$
Gyroscopes	Analog-to-Digital Converter Bitrate	16	32
	Range	± 600 $\frac{\text{deg}}{\text{s}}$	± 450 $\frac{\text{deg}}{\text{s}}$
	Scale Factor Error	$\mathcal{U}(-500, 500)$ ppm	$\mathcal{U}(-100, 100)$ ppm
	Axes Nonorthogonality Error	$\mathcal{U}(-103, 103)$ arcsec	$\mathcal{U}(-180, 180)$ arcsec
	Axes Misalignment Error	$\mathcal{U}(-103, 103)$ arcsec	$\mathcal{U}(-3600, 3600)$ arcsec
	Angular Random Walk	0.03 $\frac{\text{deg}}{\sqrt{\text{hr}}}$	0.3 $\frac{\text{deg}}{\sqrt{\text{hr}}}$
	Bias Instability	18 $\frac{\text{deg}}{\text{hr}}$	$6.25 \frac{\text{deg}}{\text{hr}}$
	Bias Instability Time	100 s	100 s
	Startup Bias	$\mathcal{U}(-0.25, 0.25)$ $\frac{\text{deg}}{\text{s}}$	$\mathcal{U}(-0.2, 0.2)$ $\frac{\text{deg}}{\text{s}}$

4.3 Performance Metrics

In order to assess the performance of an IMU for a given trajectory the standard deviation in the position, velocity, and attitude is utilized as the performance metric. Because the ADIS16488 and the 3DM-GX3-15 are both relatively low-end IMUs, two cases are considered: Case 1 analyzes only the first 500 seconds of the entire 41 minute trajectory while Case 2 analyzes the entire 41 minute trajectory.

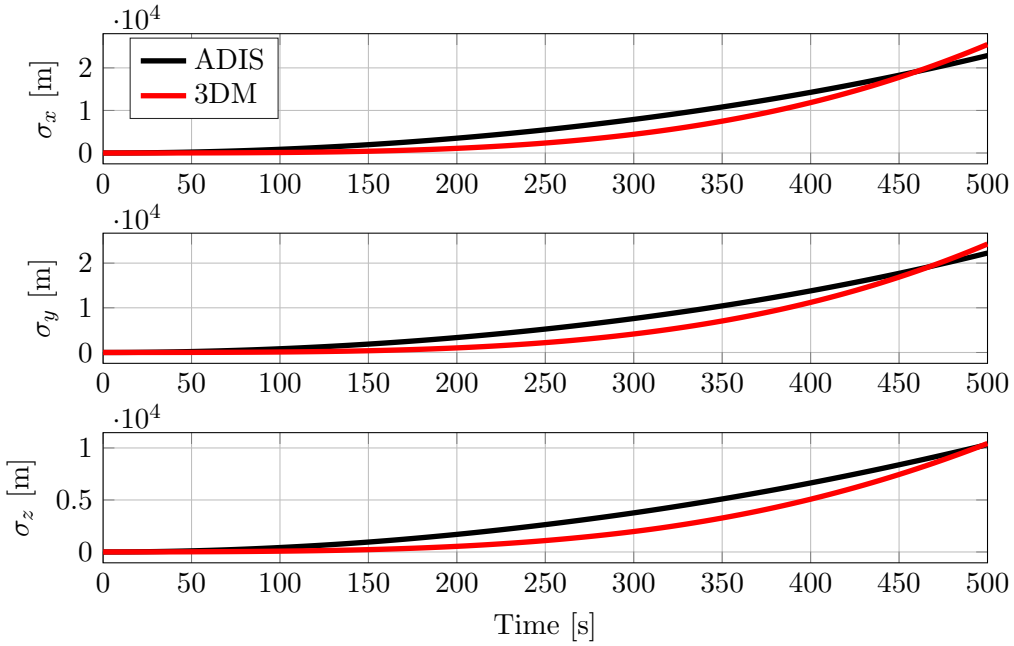


Figure 12. Position Standard Deviation as a Function of Time for the ADIS16488 and the 3DM-GX3-15 IMUs: Case 1

4.3.1 Standard Deviation in Position and Attitude. In order to quantify uncertainty in the position, velocity, and attitude states, the standard deviation, or square root of the second-central moment, is calculated and plotted as a function time based on the Monte Carlo samples or the linear covariance estimate. The position and attitude standard deviations computed for the ADIS16488 and 3DM-GX3-15 IMUs can be found in Figs. (12) and (13) respectively for the Case 1 trajectory with velocity omitted for brevity. Due to the similarity between the strapdown and space stabilized metrics, only the strapdown results are presented. As was stated in Section 4.2, the ADIS under-performs against the 3DM at the beginning of the run, but this switches within the Case 1 bounds. Through the rest of the run, the ADIS continues to outperform the 3DM. The Case 2 plots are omitted for the IMU comparison for conciseness; after the 500 second mark, the two IMUs continue to diverge.

To test the validity of linear covariance analysis, the standard deviation of the position, velocity, and attitude for the strapdown mechanization are plotted against the same metric computed via Monte Carlo simulation. The results for Case 2 are shown in Figs. (14)–(16). With the exception of the z component of the velocity in Fig. (15), the linear covariance and Monte Carlo analyses match very closely. The standard deviations observed from Case 1 are so nearly identical, the plots were not included.

Similarly to the strapdown mechanization, the space stabilized results from linear covariance and Monte Carlo match quite closely with the exception of the velocity z component. When scrutinized, the deviation of linear covariance from Monte Carlo is smaller when the space stabilized mechanization is implemented. This is to be expected due to the nature of the dynamics. Newtonian mechanics produce the simplest result when the acceleration components occur naturally in the inertial frame which is the intent of a space stabilized mechanization. Conversely, the accelerations obtained via a strapdown mechanization must be converted from the body to the estimated inertial frame.

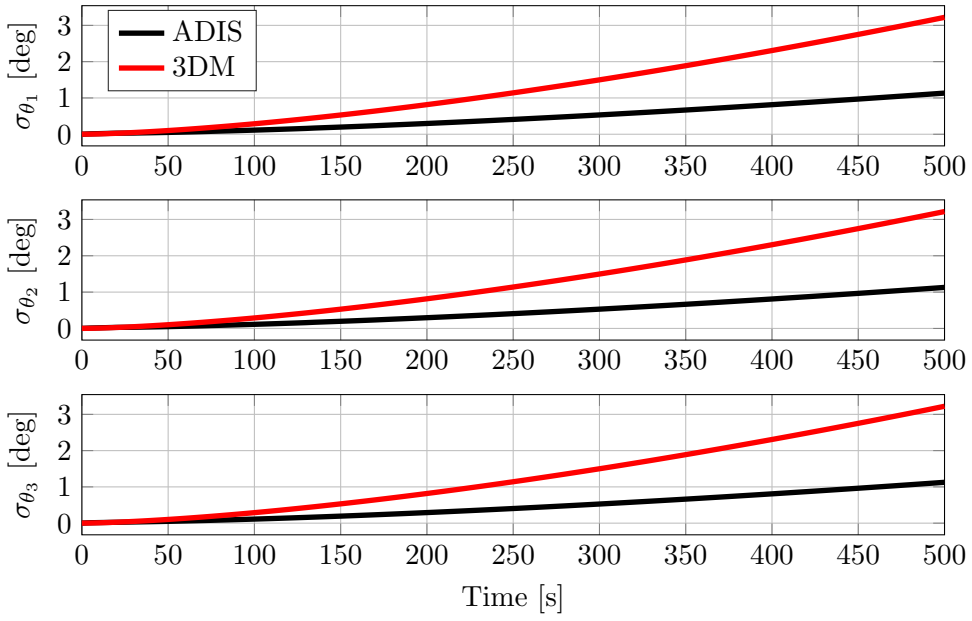


Figure 13. Attitude Standard Deviation as a Function of Time for the ADIS16488 and the 3DM-GX3-15 IMUs: Case 1

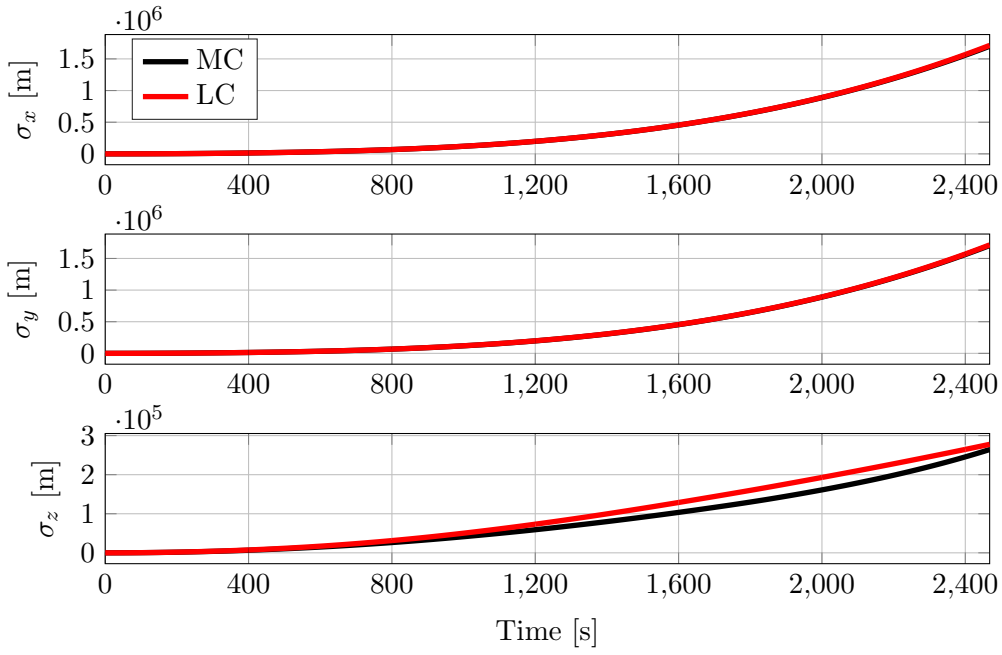


Figure 14. Position Standard Deviation for the ADIS16488 in a Strapdown Mechanization: Case 2

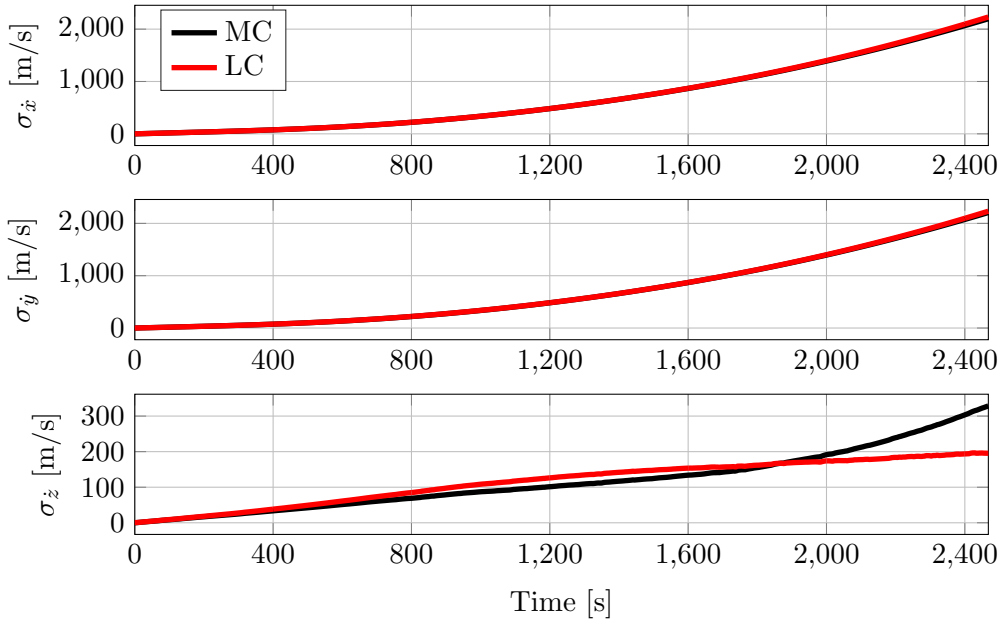


Figure 15. Velocity Standard Deviation for the ADIS16488 in a Strapdown Mechanization: Case 2

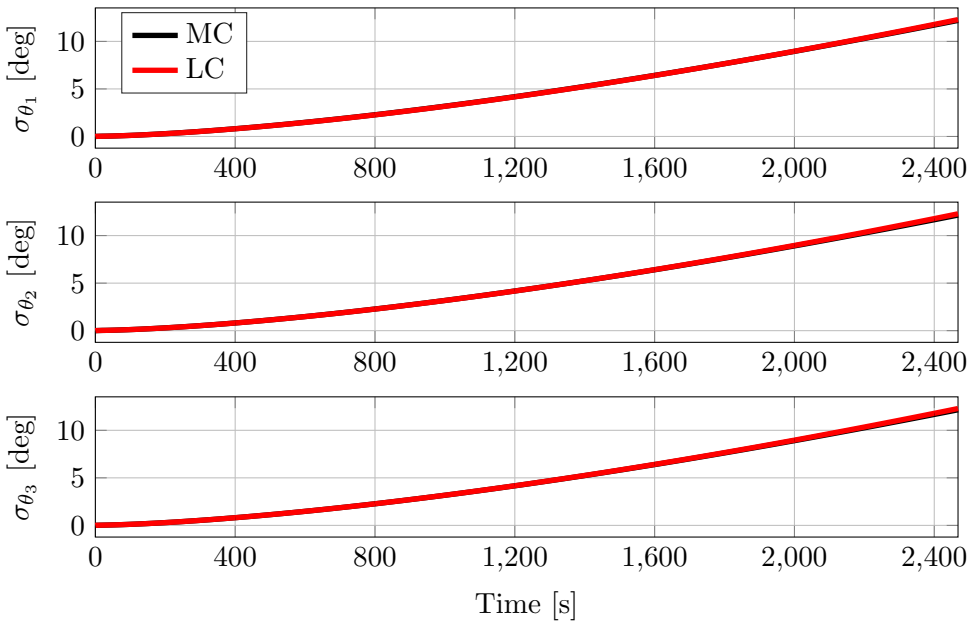


Figure 16. Attitude Standard Deviation for the ADIS16488 in a Strapdown Mechanization: Case 2

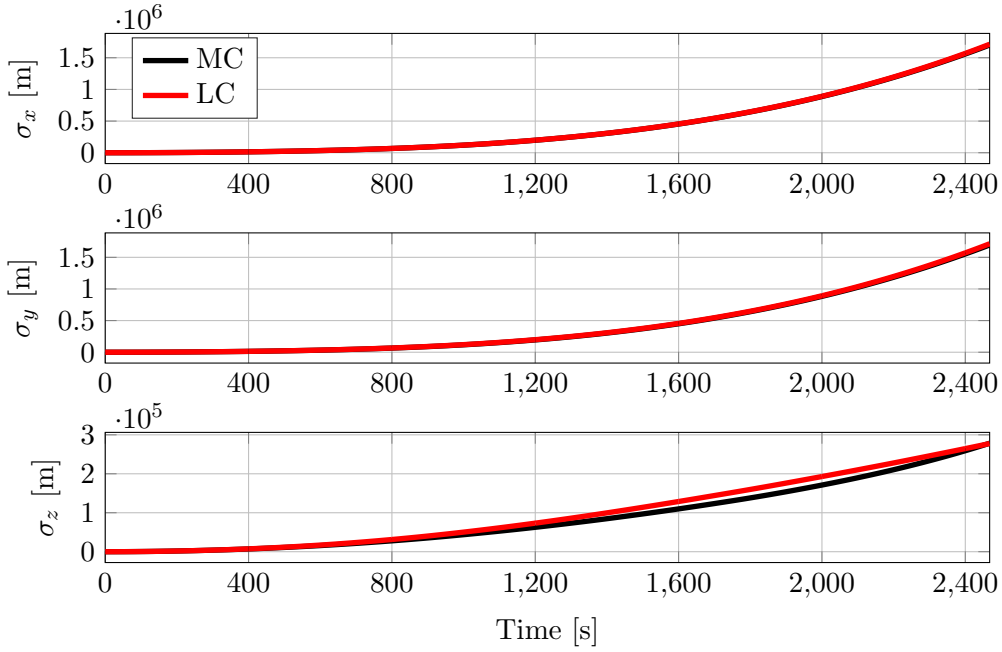


Figure 17. Position Standard Deviation for the ADIS16488 in a Space Stabilized Mechanization: Case 2

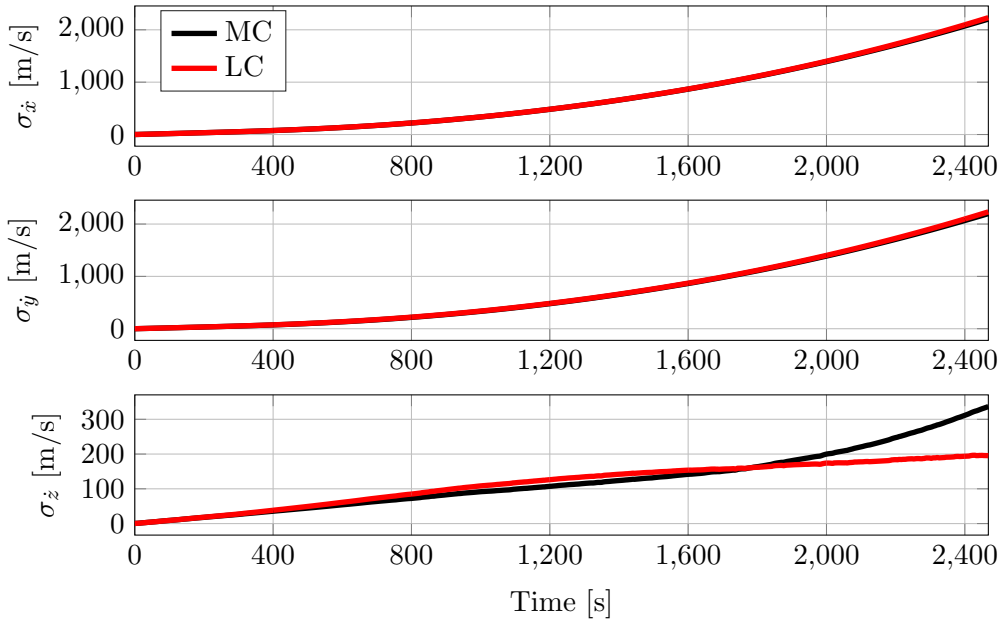


Figure 18. Velocity Standard Deviation for the ADIS16488 in a Space Stabilized Mechanization: Case 2

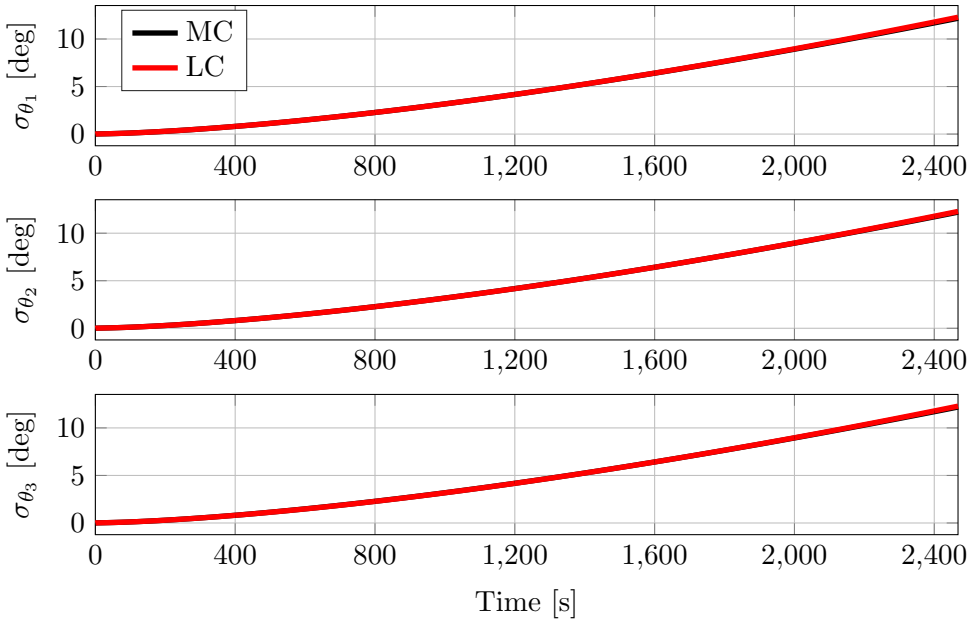


Figure 19. Attitude Standard Deviation for the ADIS16488 in a Space Stabilized Mechanization: Case 2

Table 2. External Aiding Configuration

Sensor	Start Time (s)	Finish Time (s)	Meas. Spacing (s)	Sensor Noise
Range/Range-Rate	0	150	10	0.1 m-m/s
Stellar Line-of-Sight	400	600	10	20 arcsec
Bearing Angles	800	1000	0.1	20 arcsec

4.3.2 Standard Deviation with External Aiding. The effects of the three external sensors (range/range-rate, stellar line-of-sight, and bearing angles) on the standard deviation growth for the space stabilized mechanization are presented in Figs. (20)–(22). The sensors are run according to the values given in Table 2. The figures show the ADIS with varying levels of external aiding: unaided, range/range-rate and stellar line-of-sight, and stellar-line-of-sight and bearing angles. It should be noted that the noise associated with each of the sensors is assumed to be zero-mean white noise.

Based on the standard deviation growth pattern, although the late stage bearing angles significantly reduce translational uncertainty, initial range/range-rate measurements are much more beneficial.

As a demonstration of the flexibility of the tool, Figs. (23)–(25) depict a hypothetical case study attempting to determine a combination of sensors that produces standard deviation growth similar to a "tactical" grade IMU. Shown is an unaided fictional tactical IMU (achieved by reducing error sources in the ADIS IMU), an unaided ADIS, and an aided ADIS. Based on this analysis, a user could determine that the use of the three presented external sensors could ensure the uncertainty growth associated with an ADIS IMU would be comparable to the tactical IMU. A user could then confidently assume that the purchase of a costly high-grade IMU would be unnecessary provided that the external sensor data is available via the additional sensors. These results are easily obtainable with minimal computational effort. Had a case study of this sort been attempted in a Monte

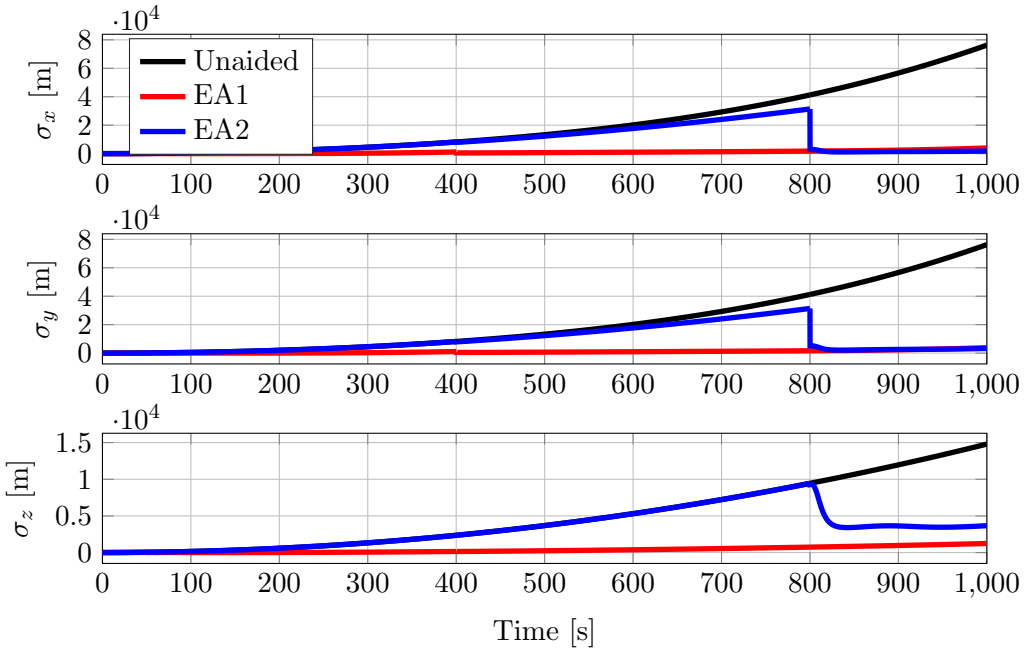


Figure 20. Position Standard Deviation from the ADIS16488 in a space stabilized mechanization with and without external aiding. EA1 utilizes range/range-rate and stellar line-of-sight measurements, while EA2 utilizes stellar line-of-sight and bearing angles.

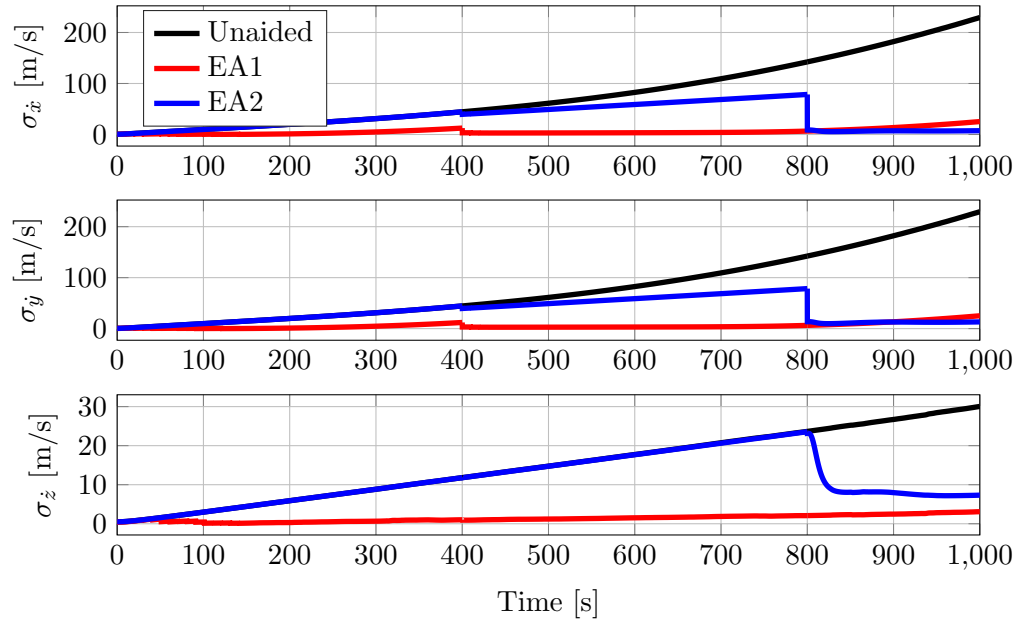


Figure 21. Velocity Standard Deviation from the ADIS16488 in a space stabilized mechanization with and without external aiding. EA1 utilizes range/range-rate and stellar line-of-sight measurements, while EA2 utilizes stellar line-of-sight and bearing angles.

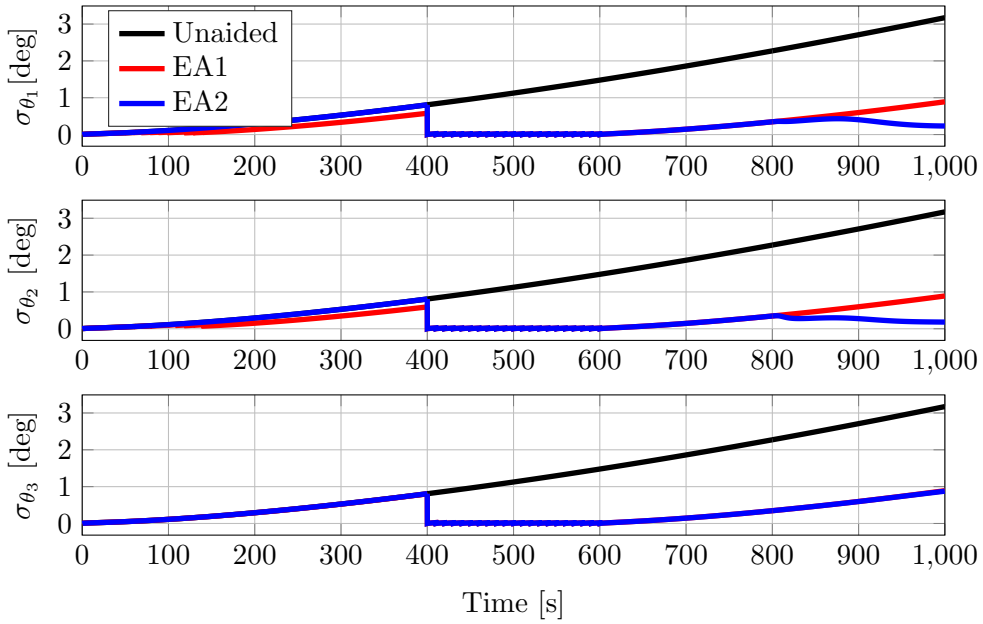


Figure 22. Attitude Standard Deviation from the ADIS16488 in a space stabilized mechanization with and without external aiding. EA1 utilizes range/range-rate and stellar line-of-sight measurements, while EA2 utilizes stellar line-of-sight and bearing angles.

Carlo simulation (with external aiding utilities), this process would have required a much larger computational burden.

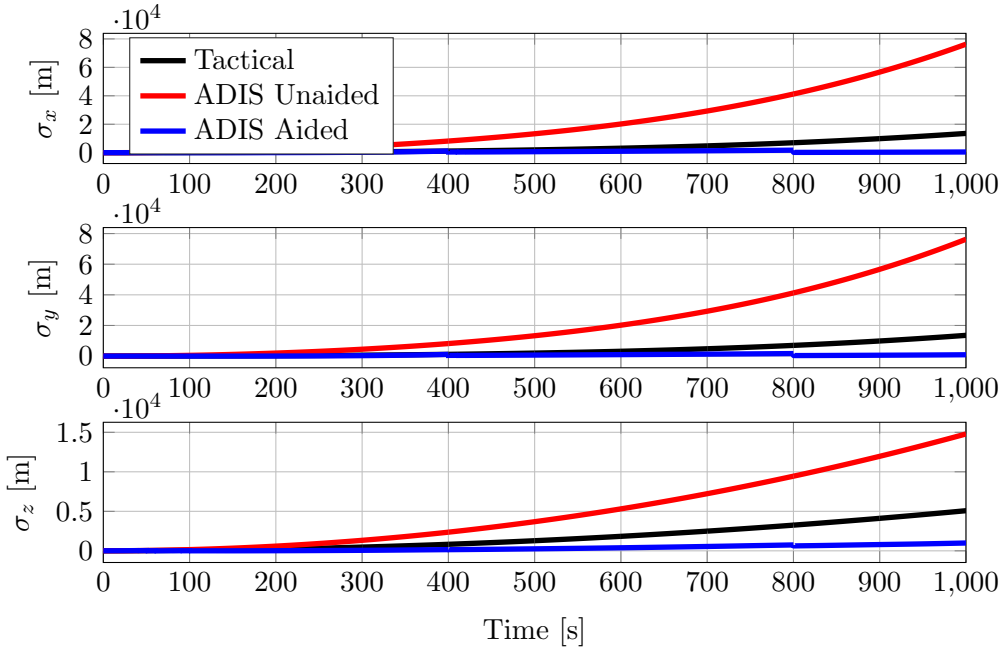


Figure 23. Position standard deviations resulting from an unaided tactical IMU and the ADIS16488 IMU with and without external aiding in a space stabilized mechanization. All three sensors listed in Table 2 are utilized.

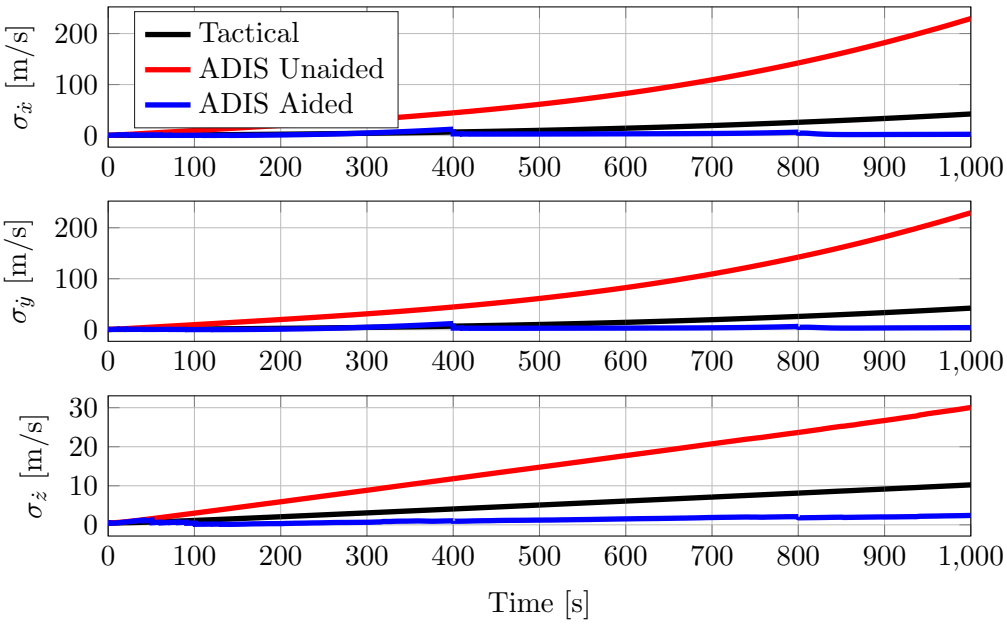


Figure 24. Velocity standard deviations resulting from an unaided tactical IMU and the ADIS16488 IMU with and without external aiding in a space stabilized mechanization. All three sensors listed in Table 2 are utilized.

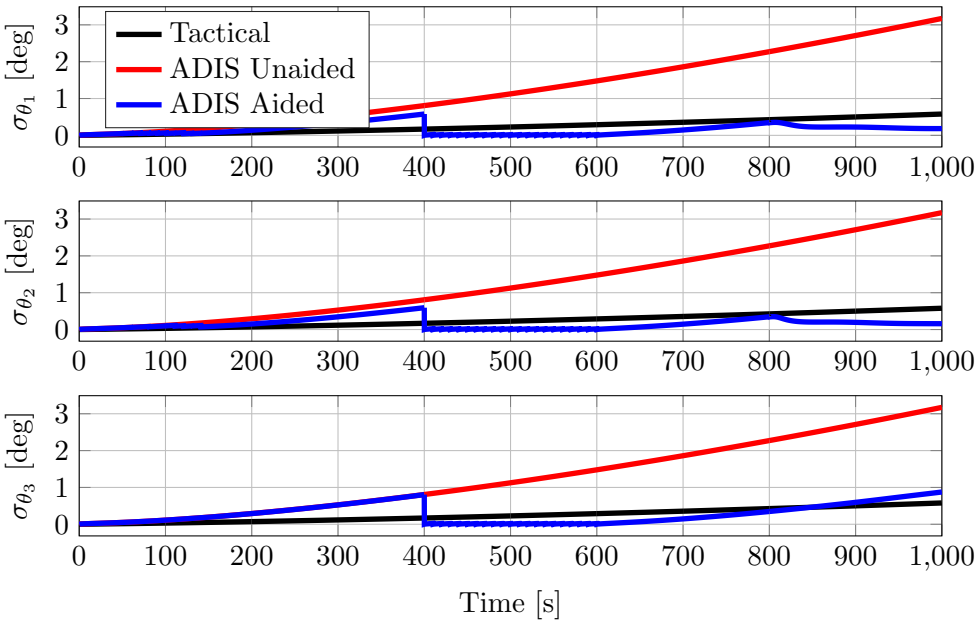


Figure 25. Attitude standard deviations resulting from an unaided tactical IMU and the ADIS16488 IMU with and without external aiding in a space stabilized mechanization. All three sensors listed in Table 2 are utilized.

5.0 CONCLUSIONS

Before choosing an IMU and external sensors to be implemented in a navigation system, SAIMUN can be used to analyze the performance of the full sensor suite along a given trajectory or trajectories. When comparing the Lord MicroStrain 3DM-GX3-15 and the Analog Devices ADIS16488 IMUs in the navigation of an example vehicle trajectory, the lower random walk specification of the 3DM-GX3-15 causes its associated uncertainty to grow at a slower rate during the beginning of the trajectory; however, the higher bias instability of the 3DM-GX3-15 causes its associated uncertainties to grow faster and overtake the uncertainties associated with the ADIS16488. For these IMUs, 425-500 seconds is observed to be the approximate window in which the performance of the IMUs is most similar, as quantified by the standard deviations. Before this window, the 3DM-GX3-15 provides a more accurate navigation solution. After this window, the ADIS16488 provides a more accurate navigation solution. SAIMUN allows this trade study to be performed before purchasing or testing any hardware and will allow for larger trade spaces to be investigated virtually, obtaining, building, or testing any physical IMUs.

SAIMUN enables both gimbal- and strapdown-based navigation systems to be analyzed virtually in the developed simulation environment. An IMU error parameterization and a trajectory are defined in the simulation environment along with a variety of external sensors, including their own error parameterizations. To provide validity of the unaided linear covariance analysis, both linear covariance and Monte Carlo analyses can be run in the same simulation. With linear covariance analysis verified via Monte Carlo analysis, the external aiding sensors can be added to the simulation and Monte Carlo can be removed.

Currently, SAIMUN is capable of rendering navigation solutions for gimbal- and strapdown-based navigation systems, while considering IMU error sources due to startup bias, bias instability, thermo-mechanical zero-mean white noise, scale factor errors, axes misalignment errors, axes nonorthogonality errors, and quantization effects caused by analog-to-digital conversion. Where appropriate, SAIMUN allows for a selection from multiple probabilistic representations of the IMU error sources, such as Dirac, Gaussian, and uniform distributions. The structure of SAIMUN supports future extensions, such as additional IMU mechanizations, IMU parameter errors, error source distributions, and sensors. On the analysis side of SAIMUN, this is also true; that is, analyses extending beyond those presented in this report can be incorporated. The simulation and analysis environment is designed to be modular such that these modifications are straightforward to implement within SAIMUN.

Multiple possibilities for external aiding are incorporated into SAIMUN, such as range/range-rate measurements, stellar line-of-sight measurements, and bearing angles measurements. SAIMUN facilitates analysis that combines both IMU and external aiding data to investigate the possible navigation accuracy of aided or unaided navigation solutions. In this way, SAIMUN can be used to determine if a lower cost IMU can be paired with a set of external sensors to achieve similar or better performance as a higher cost IMU. It is shown that combining all three external sensors with a MEMS IMU can, for the trajectory investigated, lead to similar navigation accuracies as would be achieved with a fictitious tactical grade IMU.

REFERENCES

[1] Woodman, Oliver J., *An Introduction to Inertial Navigation*, Tech. Rep. UCAM-CL-TR-696, University of Cambridge, Computer Laboratory, August 2007.

[2] Wang, H. G. and Williams, T. C., "Strategic Inertial Navigation Systems," *IEEE Control Systems Magazine*, Vol. **28**, No. 1, February 2008, pp. 65 – 85.

[3] DeMars, Kyle J., *Precision Navigation for Lunar Descent and Landing*, Master's thesis, The University of Texas at Austin, Austin, TX, May 2007.

[4] Zanetti, Renato, *Advanced Navigation Algorithms for Precision Landing*, Ph.D. thesis, The University of Texas at Austin, Austin, TX, December 2007.

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

AIRS	Advanced Inertial Reference Sphere
CM	Center of Mass
EKF	Extended Kalman Filter
FPA	Focal Plane Angles
IMU	Inertial Measurement Unit
MEMS	MicroElectricalMechanical Systems
ng	non-gravitational
pdf	Probability Density Function
QUEST	Quaternion Estimation
SAIMUN	Simulation and Analysis of IMU-based Navigation
VRW	Velocity Random Walk

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 2 cys

Official Record Copy
AFRL/RVBYE/Stephen Forbes 1 cy

This page is intentionally left blank.