Global Positioning System Receiver Simulation for 
Integrated Inertial Navigation Systems *

Dan Simon and Hosny El-Sherief  
TRW Systems Integration Group, SB2-1651  
PO Box 1310  
San Bernardino, CA  92402-1310

Abstract

There is much interest in integrated navigation using the Global Positioning System and Inertial Measurement Units. This integration combines the long-term accuracy of GPS with the short-term accuracy of IMUs, thereby realizing the best of both worlds. There is thus a need for realistic simulations of integrated GPS/IMU navigation systems. This paper discusses such a simulation, and presents some simulation results. The specific application considered is missile navigation.

1 Introduction

In spite of much recent interest in integrated GPS/IMU navigation [4], to the authors' knowledge there is little published work on the derivation of GPS receiver design constraints which are based on integrated navigation requirements. As a result, many integrated navigation systems use GPS receivers which are either underdesigned or overdesigned. That is, the receivers provide either much better or much worse performance than required. There is therefore a need for realistic simulations of integrated GPS/IMU navigation, so that integrated navigation systems can be designed more intelligently and cost-effectively. This paper discusses a software-based simulator for integrated GPS/IMU navigation. The particular application considered is missile navigation.

GPS contains three major segments: a space segment, a control segment, and a user segment. The space segment consists of a constellation of semi-synchronous satellites which transmit composite signals at two L-band frequencies. The control segment consists of monitor stations which precisely compute the ephemerides and clock drifts of the satellites, and then uplink these corrections to the satellites. The user segment consists of a receiver which is designed to track the pseudorandom (PR) binary codes and carrier frequencies transmitted by the satellites, and hence determine user position and velocity [5].

The ranges between the satellites and the GPS receiver are obtained by tracking the PR codes, typically using a delay-lock loop (DLL) [11], which operates at a rate of 50 Hz. The range error depends on such things as the vehicle dynamics, the loop filter design, and the signal-to-noise ratio (SNR).

The receiver also measures the relative velocities between itself and the satellites. This is done by tracking the phase of the carrier, typically with a phase-locked loop (PLL) [7]. The delta range is determined by taking the phase at the end of the measurement interval and subtracting the phase at the beginning of the interval. The phase measurement is called the doppler range, and the difference between two doppler ranges is called a delta range. The tracking error depends on such things as the vehicle dynamics, the loop filter design, and the SNR.

GPS receivers can be designed to track the L1 (1.575 GHz) carrier, the L2 (1.228 GHz) carrier, or both. Tracking both frequencies allows for better correction of the ionospheric refraction error since ionospheric refraction is proportional to the square of the wavelength. The residual ionospheric refraction which remains uncorrected results in both PR code-derived range errors and carrier phase-derived doppler range errors [9]. In the single frequency case, ionospheric modelling helps reduce the ionospheric refraction error by 50 - 75%. But ionospheric modelling cannot be used to predict the change in the ionospheric refraction very accurately; so modelling does not reduce the doppler range error due to ionospheric refraction.

Most IMUs have relatively good short term accuracy – that is, they have very little high frequency error. On the other hand, GPS has relatively good
long term accuracy – that is, it has very little low frequency error. This combination makes integrated
GPS/IMU navigation a natural application of GPS, giving a user the best of both worlds [4]. Integrated
GPS/IMU navigation is performed using Kalman filter. First, the IMU and GPS are used separately
to compute the ranges and delta ranges between the user and the satellites. The differences between these
IMU-indicated and GPS-indicated quantities are the measurements which are used in the Kalman filter.
The filter states are the GPS and IMU instrumentation errors. The sensitivities of position and velocity
measurements to GPS and IMU errors are used to compute the measurement matrix of the Kalman filter.

Navigation performance can be quantified by Position and Velocity Spherical Error Probable (PSEP and
VSEP). SEP is defined as the 50th percentile probability radius of the navigation solution. That is, there is
a 50% probability that the vehicle position (velocity) is within PSEP (VSEP) of the navigation solution.

2 Simulation Overview

This section discusses the components of the integrated GPS/IMU navigation simulation. A simulation
can be set up by running IMUSENS and GPSSENS. These two programs need to be run only once for a
given trajectory and for a given IMU and GPS receiver. Then the following five programs are run n
times, where n is the number of Monte Carlo samples which the user deems appropriate: PERTURB,
GPSSIM, GPSOBS, KFILTER, and NAVTRAJ. Once these five programs have been run n times, the user is
in a position to obtain a statistical measure of the performance of the integrated GPS/IMU navigator. That
can be done by running SEPFCALC. These eight programs are discussed further below.

1 IMUSENS

IMUs typically consist of three accelerometers and three gyroscopes. The accelerometers and gyroscopes
output the computed vehicle position, velocity, and attitude (navigation solution). But IMU errors result
in an incorrect navigation solution.

The accelerometers measure acceleration, which is compensated for gravity and then integrated twice to
compute velocity and position. As an example of how error sensitivities can be computed for an accelerome-
ter, consider the following simplified error model.

\[ A_m = A(1 + S_a) + B_a \]  

where

\[ A_m, A = \text{measured and true acceleration} \]

\[ S_a = \text{accelerometer scale factor error} \]

\[ B_a = \text{accelerometer bias}. \]

A gyro measures angular rotation rate, which is integrated to compute attitude. The following equation is
a simplified error model for a gyro.

\[ \Omega_m = \Omega(1 + S_\phi) + D \]  

where

\[ \Omega_m, \Omega = \text{measured and true angular rate} \]

\[ S_\phi = \text{gyro scale factor error} \]

\[ D = \text{gyro drift}. \]

The IMU’s calculation of velocity and position is gov-
erned by the following differential equation.

\[ \frac{d^2 R_e}{dt^2} = G(R_e) + A_m \]  

where

\[ R_e = \text{IMU-computed position vector} \]

\[ G(z) = \text{gravity vector at position } z \]

The true velocity and position are governed by the differential
equation

\[ \frac{d^2 R}{dt^2} = G(R) + A \]  

where \( R \) and \( A \) are the true position and acceleration.
It can be shown [8] that by differencing Equations 3 and 4 and applying a small angle approximation, a set
of nine differential equations describing the propagation of navigation errors can be written as

\[ \frac{d}{dt} \left( \frac{\partial r}{\partial e_i} \right) = \frac{\partial v}{\partial e_i} \]  

\[ \frac{d}{dt} \left( \frac{\partial v}{\partial e_i} \right) = \frac{\partial G}{\partial R} \frac{\partial r}{\partial e_i} + A \times \left( \frac{\partial \phi}{\partial e_i} \right) + f_a \]  

\[ \frac{d}{dt} \left( \frac{\partial \phi}{\partial e_i} \right) = f_\phi. \]

In the above equations,

\[ f_a = \begin{cases} \frac{\partial A_m}{\partial e_i} & \text{accelerometer error propagation} \\ 0 & \text{gyro error propagation} \end{cases} \]

\[ f_\phi = \begin{cases} 0 & \text{accelerometer error propagation} \\ \frac{\partial \Omega_m}{\partial e_i} & \text{gyro error propagation} \end{cases} \]

\[ e_i = \text{i-th IMU error source} \]

\[ r, v = \text{IMU position and velocity errors} \]

\[ \phi = \text{IMU attitude error}. \]
Equations 5 are numerically integrated for each IMU error to obtain the sensitivities of the IMU navigation error to each IMU error.

If $f_a$ and $f_e$ are both set to zero and the initial conditions for the nine states of Equations 5 are set to unity in turn, the integration yields error sensitivities due to unit errors in initial position, velocity, and attitude. Once error sensitivities are obtained for each error source desired, $\frac{\delta r}{\delta e_1}$ and $\frac{\delta v}{\delta e_1}$ are used to form the time-varying measurement matrix which is used in the Kalman filter.

(2) GPSSENS
GPS errors can be categorized as either space segment errors or user segment errors. The space segment errors are quantities over which the user has little or no control, while the user segment errors are directly affected by the design of the receiver. Some of the errors are biases (essentially constant during the course of a mission), some of the errors are Gauss-Markov processes [3] with an associated time constant $\tau$, and other errors are modelled as white Gaussian noise.

<table>
<thead>
<tr>
<th>Error</th>
<th># of Terms</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satellite Errors</td>
<td>$3n$</td>
<td>Bias</td>
</tr>
<tr>
<td>Satellite Position</td>
<td>$3n$</td>
<td>Bias</td>
</tr>
<tr>
<td>Satellite Velocity</td>
<td>$n$</td>
<td>Bias</td>
</tr>
<tr>
<td>Clock Phase</td>
<td>$n$</td>
<td>Bias</td>
</tr>
<tr>
<td>Clock Frequency</td>
<td>$n$</td>
<td>Bias</td>
</tr>
<tr>
<td>DLL Tracking</td>
<td>$n$</td>
<td>WGN (*)</td>
</tr>
<tr>
<td>PLL Tracking</td>
<td>$n$</td>
<td>WGN (*)</td>
</tr>
<tr>
<td>Ionospheric Range</td>
<td>$n$</td>
<td>Bias</td>
</tr>
<tr>
<td>Ionospheric Doppler</td>
<td>$n$</td>
<td>Markov</td>
</tr>
<tr>
<td>Tropospheric Error</td>
<td>$n$</td>
<td>Markov</td>
</tr>
<tr>
<td>Antenna Phase</td>
<td>$n$</td>
<td>Markov</td>
</tr>
<tr>
<td>Global GPS Errors</td>
<td>$n$</td>
<td>Markov</td>
</tr>
<tr>
<td>Antenna Phase Center</td>
<td>$3$</td>
<td>Bias</td>
</tr>
<tr>
<td>GPS/IMU Clock Offset</td>
<td>$1$</td>
<td>Bias</td>
</tr>
</tbody>
</table>

Table 1: GPS Errors for a Typical Receiver Tracking $n$ Satellites. (*) WGN = White Gaussian Noise.

GPSSENS uses basic physical laws to generate the sensitivities of the GPS range and delta range measurement errors to GPS errors. Most of the errors are satellite-specific. But some of the GPS errors are global, which means that there is only one error for the entire GPS. Table 1 contains a list of the GPS errors whose sensitivities are generated by GPSSENS. See [10] for standard deviations, time constants, and mathematical expressions of the GPS error sensitivities.

(3) PERTURB
PERTURB is a relatively straightforward program. Its inputs are the IMUSENS-generated IMU error sensitivities, and a file of IMU error standard deviations. It calls a random number routine to generate IMU errors. It then multiplies the IMU errors by the time-varying IMU error sensitivities to form trajectory errors. These errors are then added to the true trajectory to form an IMU-indicated trajectory.

(4) GPSSIM
GPSSIM is the counterpart of PERTURB. First it reads the vehicle trajectory and generates satellite orbital information. It then forms true range and delta range measurements for each satellite which is tracked by the GPS receiver. The user specifies how many satellites can be tracked by the receiver. GPSSIM then reads the GPS error sensitivities and a file of GPS error standard deviations. It calls a random number routine to generate GPS errors. Some of the GPS errors are Markov processes, and are generated accordingly. GPSSIM then multiplies the errors by the time-varying GPS error sensitivities to form range and delta range errors. These errors are then added to the true ranges and delta ranges to form GPS-indicated ranges and delta ranges.

The receiver simulation employed in GPSSIM utilizes a 50 Hz DLL with a user-selectable bandwidth and phase margin. In addition, the user can specify either a 2nd or 3rd order DLL filter. The PLL simulation is either a 50 Hz Costas loop [6] or a 500 Hz Extended Kalman Filter [12]. If a Costas loop is used, the PLL is rate-aided by the IMU rate information (read from the output of PERTURB) to maintain phase lock during high dynamics, and the user can specify the bandwidth, the phase margin, and filter order (2nd or 3rd order). The user can also specify the signal-to-noise ratio (SNR) for both the DLL and the PLL.

(5) GPSOBS
GPSOBS is a program which transforms the IMU-indicated trajectory and the IMU error sensitivities into GPS coordinates. The IMU-indicated trajectory output from PERTURB, and the IMU error sensitivities output from IMUSENS, are in Cartesian coordinates. But the GPS-indicated measurements output from GPSSIM, and the GPS error sensitivities output from GPSSENS, are in vehicle-to-satellite coordinates. GPSOBS uses the vehicle-to-satellite geometries to transform the IMU-indicated trajectory and the IMU error sensitivities into range and delta range coordinates. This is in preparation for the Kalman filter.
(6) KFILTER

KFILTER implements a discrete-time Kalman filter to estimate the IMU and GPS errors for a given flight. KFILTER gets the IMU and GPS measurements from GPSOBS and GPSSENS. It gets the error sensitivities from GPSOBS and GPSSENS. The time-varying measurement matrix used in the filter is formed by differencing the IMU and GPS error sensitivities. The filter may be reduced order due to (simulated real-time) computer throughput constraints.

(7) NAVTRAJ

Once the Kalman filter generates estimates of the IMU errors, the estimates can be multiplied by the IMU error sensitivities (output from IMUSENS) to obtain an estimate of the IMU navigation error. The navigation error estimate is then added to the IMU-indicated trajectory to obtain an estimate of the true trajectory. This estimate is what a real time GPS/IMU navigation flight computer would compute as the vehicle trajectory. This estimate is then subtracted from the true trajectory to obtain a GPS/IMU integrated navigation trajectory error as a function of time.

(8) SEPCalc

After programs PERTURB, GPSSIM, GPSOBS, KFILTER, and NAVTRAJ have been run n times, the user is in a position to perform a statistical analysis of integrated navigation performance. This is done by taking the n outputs of NAVTRAJ and computing PSEP and VSEP. This is done by simply using the n position and velocity errors at each time point and computing the median.

3 Simulation Study

This section presents a GPS receiver study based on the tools discussed in the previous section. The study is based on a representative missile test trajectory. The missile has three solid rocket motors, each of which burn for approximately 60 seconds. The missile’s IMU consists of three accelerometers and three gyro’s, and contains 76 error terms – 10 initial condition errors, 33 accelerometer errors, and 33 gyro errors [2, 10]. The missile’s GPS receiver tracks six satellites and the flight computer derives an integrated GPS/IMU navigation solution once per second.

The receiver simulation (GPSSIM) was set up with a DLL bandwidth of 20 rad/sec and a phase margin of 45°. The simulated PLL was an IMU rate-aided 3rd order Costas loop with a bandwidth of 2 rad/sec and a phase margin of 45°. The PLL and DLL SNRs were both set to 30 dB-Hz. Fourteen GPS errors were simulated for each satellite, and four global GPS errors were simulated (see Table 1).

A full order Kalman filter would contain 164 states – 76 IMU error states, 4 global GPS errors, and 6 × 14 satellite-specific errors. It may not be feasible to run such a large filter in real time, so two reduced order filters were implemented (a lowest order filter with 47 states, and a medium order filter with 65 states). The most significant IMU and GPS errors were used as filter states. The variances of the IMU and GPS errors which were used as filter states were increased proportionately to compensate for the fact that most of the IMU and GPS errors were neglected in the Kalman filter [10]. The unmodelled errors are also compensated for by adding additional delta range measurement noise to the filter.

The simulation tools discussed in this paper were implemented in Fortran on a Sun SPARCstation 10. The required CPU time for each of the simulation components is as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>CPU Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMUSENS</td>
<td>345 s</td>
</tr>
<tr>
<td>GPSSENS</td>
<td>40 s</td>
</tr>
<tr>
<td>PERTURB</td>
<td>30 s</td>
</tr>
<tr>
<td>GPSSIM</td>
<td>28 s</td>
</tr>
<tr>
<td>GPSOBS</td>
<td>3 s</td>
</tr>
<tr>
<td>KFILTER</td>
<td>37 s (47 states)</td>
</tr>
<tr>
<td></td>
<td>67 s (65 states)</td>
</tr>
<tr>
<td>NAVTRAJ</td>
<td>8 s</td>
</tr>
<tr>
<td>SEPCalc</td>
<td>4 s</td>
</tr>
</tbody>
</table>

Each SEP data point was determined by conducting a Monte Carlo analysis of 100 samples. First, IMUSENS and GPSSENS were run once to generate IMU and GPS error sensitivities along the missile trajectory. Then PERTURB, GPSSIM, GPSOBS, KFILTER, and NAVTRAJ were run 100 times. The position and velocity errors for the 100 Monte Carlo samples were then used by SEPCalc to statistically compute SEP as a function of time. Since SEP at payload deployment determines weapon delivery accuracy, SEP at 184 seconds following launch was used as a scalar measure of integrated navigation performance.

Table 2 shows SEP as a function of filter order for single and dual frequency. It is seen that for a single frequency receiver, filter order does not play a large part in determining PSEP. For a dual frequency receiver, filter order has a larger influence on PSEP.

It has been shown [1] that weapon delivery accuracy depends more on velocity accuracy than position accuracy. Table 2 shows VSEP for a single frequency receiver as a function of filter order, and Figure 1 shows VSEP for a dual frequency receiver as a function of antenna phase error (parameterized by filter order).
The simulation results presented in this paper indicate that the choice between a single or dual frequency receiver is critical for navigation accuracy. If a dual frequency receiver is used, then antenna phase response accuracy is a critical parameter. If a single frequency receiver is used, antenna phase response accuracy is not so important because its contribution to the navigation error is swamped by the ionospheric error. Filter order is also a critical parameter.

The simulation demonstrated in this paper can be used to determine suitable GPS receiver designs for applications other than missile navigation. This approach allows the user to specify a GPS receiver design which provides enough accuracy, but which is not overly accurate and expensive. In addition, other GPS receiver parameters can be studied using the tools described here (e.g., filter rate, and the length of time during which the PLL loses lock on the carrier).

Table 2: Integrated GPS/IMU Spherical Error Probable at Payload Deployment

<table>
<thead>
<tr>
<th>Filter Order</th>
<th>Single Freq.</th>
<th>Dual Freq.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSEP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowest</td>
<td>26.9 feet</td>
<td>22.7 feet</td>
</tr>
<tr>
<td>Medium</td>
<td>26.4</td>
<td>20.1</td>
</tr>
<tr>
<td>Full</td>
<td>26.3</td>
<td>19.6</td>
</tr>
<tr>
<td>VSEP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowest</td>
<td>0.1323 feet/sec</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td>0.0818</td>
<td></td>
</tr>
<tr>
<td>Full</td>
<td>0.0751</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Integrated GPS/IMU Spherical Error Probable at Payload Deployment

It is seen from comparing Table 2 and Figure 1 that the use of a dual frequency receiver significantly reduces VSEP. In addition, the use of higher order filters significantly reduces VSEP. Interestingly, a single frequency receiver with a full order filter may outperform a dual frequency receiver with a low order filter.

4 Conclusion

A Monte Carlo simulation tool to investigate the effects of GPS receiver design on integrated navigation performance was described and demonstrated. The specific application considered was missile navigation. Performance was based on 100 Monte Carlo samples. The Kalman filter rate was 1 Hz, and six GPS satellites were tracked during the flight. Two reduced order filters were compared to the full order filter.

References