GPS-aided inertial navigation performance evaluation using the Extended Kalman Filter

Carvalho, F. M. R.*; Bruno, M. G. S.*; Carvalho, R. T**
* ITA – Instituto Tecnológico de Aeronáutica – Praça Marechal Eduardo Gomes, 50 - Vila das Acácias, CEP 12.228-900 - São José dos Campos – SP – Brasil
** IEAv – Instituto de Estudos Avançados – Rodovia dos Tamoios, km 5,5 – Putim, Cep - 12.228-840 - São Jose dos Campos – SP – Brasil

***Abstract* — We discuss the performance of a GPS-aided inertial navigation system that employs the Extended Kalman Filter (EKF) and GPS position measurements for estimating and correcting position errors on the inertial navigation system readings. A proper error model to account for position, velocity and attitude errors is described and validate. It also estimates the bias drift of both accelerometers and gyrometers, and GPS clock bias. Results on the EKF performance are presented for different flight trajectories.**

***Key words* — Inertial Navigation, GPS, Kalman Filter**

I. INTRODUCTION

Strap-down inertial navigation systems (INS) have a set of tri-axes mutually orthogonal accelerometers. Their axes are usually aligned to a similar set of gyrometers, all assembled to the vehicle body. Measurements of acceleration in this vehicle reference frame can be computed at the Lab reference frame by axes rotation transforms using their relative angular data provided by the integration of the gyro signals. A sequence of double integration on the lab-referred accelerations can finally produce vehicle position in this frame. The INS position estimate is susceptible to errors which may grow in time, due to the successive integrations performed on sensor signals showing bias drift and noise. Contributions to the position error variance, due to gyro bias drift, may grow proportional to t^3, since that goes through three successive integrations, whereas contributions from accelerometer error variance may grow proportional to t^2. So a very precise set of gyros and accelerometers are necessary for long flights if the navigation is solely based on the INS system. An alternative method for estimating position is the Global Positioning System (GPS). The different sources of GPS position errors are well-known, and their statistical properties are well-modeled in the literature [1]-[2]. In contrast to INS errors, GPS errors have predictable statistical properties in time (even though minor error contributions due to varying atmosphere travel time and multipath induced errors are intrinsically environment sensitive). In this sense, INS and GPS measurements have a complementary nature, so GPS estimates may be used from time to time to reset INS accumulated position errors. Stochastic estimators, such as those based on the Kalman filter, can be successfully implemented if a good navigation error model is derived, and if the INS and GPS error statistical properties are properly described and quantified.

For the INS, a linear time-varying error model can be derived from the navigation equations by differentiation. The state variables in that model are the errors on position, velocity and attitude angles, plus INS and GPS sensor biases.

Fernanda M. R. de Carvalho, fer_mribeiro@yahoo.com.br, Tel +55-12-3947-6806; Marcelo G. S. Bruno, bruno@ ita.br, Tel +55-12-3947-6906; Ricardo T. de Carvalho, ricc_33@yahoo.com.br, Tel. +55-12-3947-5654.

However, GPS data are related to position errors by a non-linear model, which leads to an Extended Kalman Filter (EKF) implementation. Adding GPS own error sources to the model completes a set of equations to be the EKF basis. Fig.1 shows such system: inertial sensors feed the flight computer, which calculates the vehicle position $[\lambda\ \phi\ h]_{INS}$ through the navigation equations, where λ, ϕ and h are respectively the vehicle latitude, longitude and height above ground. The EKF filter takes on that data to run an error model that estimates the error state vector, and adds the GPS measurements to update that estimation, taking into account the assumed statistical properties of both INS and GPS errors for the covariance matrix update. Its output can correct the previous $[\lambda\ \phi\ h]_{INS}$ data.

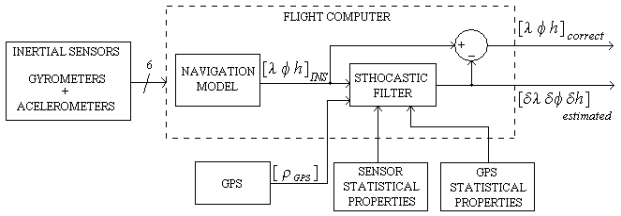

Fig. 1: Diagram showing the blocks for GPS-aided INS navigation.

Due to its potential to enhance navigation position estimation, such approach and its alternatives have been of great interest [3]-[4], in different topologies. The EKF performance though has to be evaluated on different levels of sensor accuracy and on different trajectories with their own peculiarities. In this work, we report on the error model and focus on the EKF performance for such GPS-aided navigation implementation. A complete account of INS errors, including bias errors in both gyro and accelerometer measurements, are considered, as well as the GPS intrinsic errors. The error model is described and validated. The results are finally presented and discussed.

II. ERROR MODELING AND VALIDATION

Fig. 2 shows the reference frames of interest. Here we use the notation largely employed in the literature on the variables for navigation model [3]. The vector variables taken in the vehicle reference frame will have a *b* index, where the vehicle axes are usually named roll, pitch and yaw axes, and uses the index *R*, *P* and *Y* respectively. Also is shown the ECEF reference frame (that uses the index *e*), positioned at Earth´s center and rotating with Earth at Ω_T rate. The Lab frame is where position is computed. Usually a Lab fixed on Earth´s surface is a common pick. However computations on gravity vectors and transformations are more elaborate there than it is if primarily we choose a NED reference system as an intermediary reference frame. Later a final transformation can lead to the position in the Lab frame

of choice. NED is a North-East-Down frame centered at a point on the WGS-84 ellipsoid, which represents Earth´s surface. That point is a projection of the vehicle center of mass on the ellipsoid, so this frame is moving with the vehicle. The index *n* denotes a NED variable, and *N*, *E* and *D* are the indices for its components.

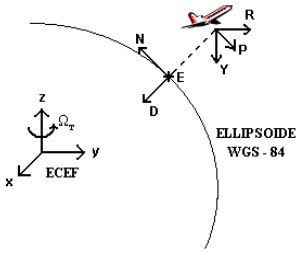

Fig.2: Reference frames used in the navigation problem.

The usefulness of this NED choice is first due to the fact that the computed velocity vector $\boldsymbol{v^n} = [v_N\ v_E\ v_D]^T$ is easily related to the general position vector $\boldsymbol{r} = [\lambda\ \phi\ h]^T$, our primary interest. Here h is the height from the ellipsoid surface. Secondly, the gravity vector $\boldsymbol{g^n} = [0\ 0\ g(h)]^T$ is easily represented. The navigation equation for the NED is

$$\frac{\boldsymbol{dv^n}}{\boldsymbol{dt}} = \boldsymbol{R_{b2n}}.\boldsymbol{a^b} + \boldsymbol{g^n} - (\Omega^{\boldsymbol{n}}_{\boldsymbol{en}} + \boldsymbol{2}\Omega^{\boldsymbol{n}}_{\boldsymbol{ie}})\boldsymbol{v^n} \qquad (1)$$

where $\boldsymbol{a^b} = [a_R\ a_P\ a_Y]^T$ has the specific forces measured by the vehicle accelerometers, and $\boldsymbol{R_{b2n}}$ is an orthonormal 3x3 rotation matrix from the vehicle RPY frame to the NED frame. It is the solution of the differential equation

$$\dot{\boldsymbol{R}}_{\boldsymbol{b2n}} = \boldsymbol{R_{b2n}}.\boldsymbol{\Omega^b_{ib}} - (\boldsymbol{\Omega^n_{ie}} + \boldsymbol{\Omega^n_{en}}). \qquad (2)$$

Here the gyro skew-symmetric 3x3 matrix $\boldsymbol{\Omega^b_{ib}}$ is derived from the rotation rate measurement vector $\boldsymbol{\omega^b_{ib}} = [\omega_R\ \omega_P\ \omega_Y]^T$. The skew-symmetric matrices $\boldsymbol{\Omega^n_{ie}}$ and $\boldsymbol{\Omega^n_{en}}$ are derived respectively from the vectors

$$\boldsymbol{\omega^n_{ie}} = [\Omega_T \cos(\lambda) \quad 0 \quad -\Omega_T \sin(\lambda)]^T \qquad (3)$$

$$\boldsymbol{\omega^n_{en}} = \left[\frac{v_E}{(R_\phi + h)} \quad \frac{-v_N}{(R_\lambda + h)} \quad \frac{-v_E \tan(\lambda)}{(R_\phi + h)}\right]^T. \qquad (4)$$

The first vector $\boldsymbol{\omega^n_{ie}}$ reflects the Earth´s rotation rate Ω_T component at latitude λ, and the second $\boldsymbol{\omega^n_{en}}$ accounts for the movement of the vehicle projection point on the WGS-84 surface, and it is a function of

$$R_\lambda = \frac{a^2 b^2}{(a^2 \cos(\lambda)^2 + b^2 \sin(\lambda)^2)^{3/2}} \qquad (5)$$

and

$$R_\phi = \frac{a^2}{(a^2 \cos(\lambda)^2 + b^2 \sin(\lambda)^2)^{1/2}} \qquad (6)$$

They are latitude-dependent curvature parameters of the WGS-84 ellipsoid. Its major and minor semi-axes are $a \cong 6378.1$ km and $b \cong 6356.8$ km. Now we may proceed to determine the model for navigation errors derived from the differentiation of (1). Let $\boldsymbol{X}$ be a 17x1 error vector defined by

$$\boldsymbol{X} = \left[\boldsymbol{\delta r}\ \boldsymbol{\delta v^n}\ \boldsymbol{\rho^n}\ \boldsymbol{\delta ba^b}\ \ \boldsymbol{\delta b\omega^b}\ \ \boldsymbol{\delta_{GPS}}\right]^T \qquad (7)$$

where the vectors inside are as follows: $\boldsymbol{\delta r} = [\delta\lambda\ \delta\phi\ \delta h]^T$ has the position errors to be estimated and subtracted from the actual INS measurement; $\boldsymbol{\delta v^n} = [\delta v_N\ \delta v_E\ \delta v_D]^T$ has the velocity errors; $\boldsymbol{\rho^n} = [\rho_N\ \rho_E\ \rho_D]^T$ has the attitude errors, where $\boldsymbol{\rho^n}$ is the vector related to a skew-symmetric matrix $\boldsymbol{P^n}$, representing a small axes rotation on the calculated $\boldsymbol{R_{b2n}}$ from its actual value $\overline{\boldsymbol{R_{b2n}}}$, due to gyro errors

$$\boldsymbol{R_{b2n}} - \overline{\boldsymbol{R_{b2n}}} = \boldsymbol{P^n}.\overline{\boldsymbol{R_{b2n}}} \qquad (8)$$

We complete $\boldsymbol{X}$ with $\boldsymbol{\delta ba^b} = [\delta ba_R\ \delta ba_P\ \delta ba_Y]^T$ and $\boldsymbol{\delta b\omega^b} = [\delta b\omega_R\ \delta b\omega_P\ \delta b\omega_Y]^T$, the accelerometer and gyro bias drift errors, both modeled as 1st order Markov processes ($\boldsymbol{\delta ba^b}$ shown below, $\boldsymbol{\delta b\omega^b}$ has similar form and parameters)

$$\dot{\boldsymbol{\delta ba^b}} = -\boldsymbol{\tau_a}^{-1}.\boldsymbol{\delta ba^b} + \boldsymbol{\varepsilon_a} \qquad (9)$$

where $\boldsymbol{\tau_a}^{-1} = \begin{bmatrix} \frac{1}{\tau_{aR}} & 0 & 0 \\ 0 & \frac{1}{\tau_{aP}} & 0 \\ 0 & 0 & \frac{1}{\tau_{aY}} \end{bmatrix}$ and $\boldsymbol{\varepsilon_a} = \begin{bmatrix} \varepsilon_{aR} \\ \varepsilon_{aP} \\ \varepsilon_{aY} \end{bmatrix}$.

$\varepsilon_{aR}, \varepsilon_{aP}\ and\ \varepsilon_{aY}$ are zero mean Gaussian noises $\mathcal{N}(0, \sigma^2_{\varepsilon a})$, and $\tau_{aR}, \tau_{aP}, \tau_{aY}$ are parameters which define if the sensor bias is more Gaussian noise in nature or more Brownian-like. Higher values for them will lead to the second case, whereas lower values will lead to the first one. This accounts for different sensor technologies. For example, medium grade fiber-optic gyros show typically a more Brownian bias nature, and accelerometers a more Gaussian bias. Finally, $\boldsymbol{\delta_{GPS}} = [\delta t_2\ x_1]$ has the receiver clock bias δt_2. It has to be accounted for making better use of GPS updates. A classical clock bias model [1] based on the phase noise ω_φ and clock frequency noise ω_f is

$$\begin{bmatrix} \dot{\delta t_2} \\ \dot{x_1} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.\begin{bmatrix} \delta t_2 \\ x_1 \end{bmatrix} + \begin{bmatrix} \omega_\varphi \\ \omega_f \end{bmatrix} \qquad (10)$$

In fact, δt_2 is often obtained from satellite-vehicle distance equations, because GPS does not use three but four satellites for that, which yields four equations for four unknowns: three position variables, *x*, *y* and *z*, in the ECEF frame, and δt_2. Defining $\boldsymbol{\rho_{GPS}} = [\rho_1\ \rho_2\ \rho_3\ \rho_4]^T$ the actual vector measured by the GPS, and ρ_i the pseudo-distance variables, we have

$$\rho_i = \sqrt{(x - X_i)^2 + (y - Y_i)^2 + (z - Z_i)^2} + c.\delta t_2 + \alpha_i, \quad i = 1,2,3,4 \qquad (11)$$

Here $\alpha_i = \mathcal{N}(0, \sigma^2_{\alpha_i})$ are the noise resulting from the contributions of all noise effects that can be modeled as Gaussian noise on the regular GPS measurement: error in the broadcast ephemeris data, multipath error, clock bias of satellite clock, atmospheric delay fluctuations (ionosphere and troposphere) and the larger contributor, the deliberate corruption of the satellite signals under the policy of selective availability (SA). Separate estimates of position error standard deviation due to those different effects are known [2,3]. By adding them all results in a total standard deviation of about 30 meters on *x, y, z* measurements. Typical values of the standard deviation σ_{α_i} are from 2 to 8 times smaller for the pseudo-distance ρ_i measurements, depending on satellite positions and line-of-sight to the vehicle [4]. Since ECEF position $[x\ y\ z]^T$ are related to $[\lambda\ \phi\ h]^T$ by

$$x = \left(\frac{a^2}{\sqrt{a^2cos^2(\lambda)+b^2sin^2(\lambda)}} + h\right).\cos(\lambda).\cos(\phi)$$
$$y = \left(\frac{a^2}{\sqrt{a^2cos^2(\lambda)+b^2sin^2(\lambda)}} + h\right).\cos(\lambda).\sin(\phi)$$
$$z = \left(\frac{b^2}{\sqrt{a^2cos^2(\lambda)+b^2sin^2(\lambda)}} + h\right).\cos(\lambda).\cos(\phi), \quad (12)$$

we can express the pseudo-distance ρ_i as a function of $[\lambda\ \phi\ h]^T$ via (11), which gives us a non-linear relationship between the GPS data and the position variables. This can be linearized by the standard procedure on the EKF using Jacobian, so that $\boldsymbol{\rho_{GPS}}$ becomes a linear function of the unknown state vector $\boldsymbol{X}$, used for the update part of the EKF:

$$\boldsymbol{Y} = \boldsymbol{\rho_{GPS}} \cong \boldsymbol{\mathcal{H}}.\boldsymbol{X} + \boldsymbol{\alpha}. \quad (13)$$

$\boldsymbol{\mathcal{H}}$ is a 4x17 matrix and $\boldsymbol{\alpha} = [\alpha_1\ \alpha_2\ \alpha_3\ \alpha_3]^T$ is the GPS Gaussian noise vector as in (11).

By differentiation of (1) with respect to all relevant variables $\lambda, \phi, h, v_N, v_E, v_D,$ and attitude angle deviations ρ_N, ρ_E, ρ_D, one get to an error dynamic model for the 17x1 state vector $\boldsymbol{X} = \left[\boldsymbol{\delta r}\ \boldsymbol{\delta v^n}\ \boldsymbol{\rho^n}\ \boldsymbol{\delta ba^b}\ \boldsymbol{\delta b\omega^b}\ \boldsymbol{\delta_{GPS}}\right]^T$:

$$\dot{\boldsymbol{X}} = \boldsymbol{A}(t).\boldsymbol{X} + \boldsymbol{B}(t) \quad (14)$$

where $\boldsymbol{A}$ is a 17x17 time-variant matrix whose elements are functions of $\lambda, \phi,$ and h, v_N, v_E and v_D, Earth rotation rate Ω_T, gravity $g(h)$, curvature parameters R_λ and R_ϕ, the $\boldsymbol{R_{b2n}}$ components, the gyro and accelerometer data, and also the τ bias parameters of (9). $\boldsymbol{A}$ can be expressed in terms of sub-matrices $\boldsymbol{C}, \boldsymbol{S}$ and $\boldsymbol{T}$ (here $\emptyset$ is a matrix full of zeros) **:**

$$\boldsymbol{A}(t) = \begin{bmatrix}[\boldsymbol{C}]_{9x9} & [\boldsymbol{S}]_{9x8} \\ [\emptyset]_{8x9} & [\boldsymbol{T}]_{8x8}\end{bmatrix} \quad (15)$$

Expressions for elements of $\boldsymbol{C}$ are found in [3]. $\boldsymbol{S}$ and $\boldsymbol{T}$ are

$$\boldsymbol{S} = \begin{bmatrix}[\emptyset]_{3x3} & [\emptyset]_{3x3} & [\emptyset]_{3x2} \\ \boldsymbol{R_{b2n}}.\boldsymbol{M_a} & [\emptyset]_{3x3} & [\emptyset]_{3x2} \\ [\emptyset]_{3x3} & \boldsymbol{R_{b2n}}.\boldsymbol{M_g} & [\emptyset]_{3x2}\end{bmatrix}$$

$$\boldsymbol{T} = \begin{bmatrix}-\boldsymbol{\tau_a}^{-1} & [\emptyset]_{3x3} & [\emptyset]_{3x2} \\ [\emptyset]_{3x3} & -\boldsymbol{\tau_g}^{-1} & [\emptyset]_{3x2} \\ [\emptyset]_{2x3} & [\emptyset]_{2x3} & \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\end{bmatrix} \quad (16)$$

where $\boldsymbol{M_a}$ and $\boldsymbol{M_g}$ are close-to-identity matrices to represent the eventual small angular axes misalignment with respect to the desired mutually orthogonal set-up, respectively, of the three accelerometers and the three gyros sets. $\boldsymbol{B}(t)$ is a 17x1 vector of Gaussian zero-mean noises given by

$$\boldsymbol{B}(t) = \left[[\emptyset]_{1x3}\ [\emptyset]_{1x3}\ [\emptyset]_{1x3}\ \boldsymbol{\varepsilon_a}\ \boldsymbol{\varepsilon_g}\ \left[\omega_\varphi\ \omega_f\right]\right]^T \quad (17)$$

Equations (13) and (14) are the basis for the EKF filter to be used in its discrete linearized form:

State vector $\hat{\boldsymbol{x}}_{\boldsymbol{k/k}}$ estimation:

$$\hat{\boldsymbol{x}}_{\boldsymbol{k/k-1}} = \boldsymbol{F_{k-1}}.\hat{\boldsymbol{x}}_{\boldsymbol{k-1/k-1}}$$

$$\hat{\boldsymbol{x}}_{\boldsymbol{k/k}} = \hat{\boldsymbol{x}}_{\boldsymbol{k/k-1}} + \boldsymbol{K_k}.\left(\boldsymbol{Y_k} - \boldsymbol{\mathcal{H}_k}.\hat{\boldsymbol{x}}_{\boldsymbol{k/k-1}}\right) \quad (18)$$

Covariance matrix $\boldsymbol{\Pi_{k/k}}$ estimation:

$$\boldsymbol{\Pi_{k/k-1}} = \boldsymbol{F_{k-1}}.\boldsymbol{\Pi_{k-1/k-1}}.\boldsymbol{F_{k-1}^T} + \boldsymbol{Q_{k-1}}$$

$$\boldsymbol{\Pi_{k/k}} = (\boldsymbol{I} - \boldsymbol{K_k}.\boldsymbol{\mathcal{H}_k}).\boldsymbol{\Pi_{k/k-1}} \quad (19)$$

Kalman gain $\boldsymbol{K_k}$:

$$\boldsymbol{K_k} = \boldsymbol{\Pi_{k/k-1}}.\boldsymbol{\mathcal{H}_k^T}.\left(\boldsymbol{\mathcal{H}_k}.\boldsymbol{\Pi_{k/k-1}}.\boldsymbol{\mathcal{H}_k^T} + \boldsymbol{R_k}\right)^{-1} \quad (20)$$

The index k is at time $k.\delta t$, where δt is the time step at each iteration. $\boldsymbol{Y_k}$ comes from GPS data, $\boldsymbol{Q_k}$ is a diagonal 17x17 matrix containing the variances of the elements in vector $\boldsymbol{B}$ associated to $\boldsymbol{X}$, and $\boldsymbol{R_k}$ is the 4x4 diagonal matrix containing the variances of α_i, associated to GPS errors. In the above equations, $\boldsymbol{F_k}$ is the value of the matrix $\boldsymbol{I} + \delta t.\boldsymbol{A}(k.\delta t)$, which comes from Euler linearization of (14). There are however more elaborated alternative methods to obtain the discrete representation of (14), at higher processing cost, which is not reported here.

III. VALIDATION OF THE ERROR MODEL

In order to proceed on EKF performance evaluation, one have first to test how accurate the error model is, since it drives the EKF estimation. This is accomplished by the scheme shown in Fig. 3.

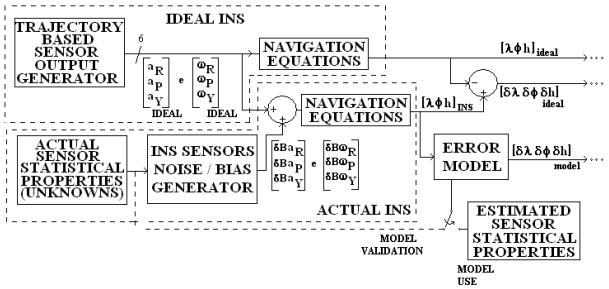

Fig.3: Block diagram to test model accuracy.

Here the diagram corresponds to a simulation carried on the Matlab/Simulink software environment. First, a trajectory is selected, and based on that, values of $[a_R\ a_P\ a_Y]^T$ and $[\omega_R\ \omega_P\ \omega_Y]^T$ are generated each iteration step. These are assumed to be the measurements made by the ideal sensors. A separate bias generator, in accordance to (9), and based on actual sensor statistical properties, outputs the bias to be added up to the ideal data, and produces a second set of acceleration and rotation rates, now simulating the actual sensor samples. These two data sets, the ideal and the actual one then feeds a navigation model, based on (1) and (2). Computation yields the ideal position $[\lambda\ \phi\ h]^T$, and also what the real sensors would indicate, here called $[\lambda\ \phi\ h]_{INS}^T$. For evaluation purposes, the unknown ideal position serves to see how well the filter tracks position. The difference $[\delta\lambda\ \delta\phi\ \delta h]^T = [\lambda\ \phi\ h]_{INS}^T - [\lambda\ \phi\ h]^T$ is the ideal position error, and should be compared to the independently generated $[\delta\lambda\ \delta\phi\ \delta h]_{model}^T$, based on (14) through (17). A condition is that the used seeds for the pseudo-random noise signals should be the same on the model and in the bias generator, done in Fig.3 by the switch to the left. Later, in the EKF performance tests, the seeds are of course not the same, and

the switch is turned right. Such arrangement is similar for velocities and attitude angles as well. Just for the sake of generality, let x be any one of those nine variables. A measure of how well the error model corresponds to reality in the simulation is evaluated by the figure of merit Δx, in dB, which is a normalized RMS deviation between the ideal error δx and the model estimated error $\widetilde{\delta x}$:

$$\Delta x(dB) = 10.\log_{10}\left(\frac{\sqrt{\frac{1}{(t_f - t_0)}\int_{t_0}^{t_f}(\widetilde{\delta x} - \delta x)^2\,dt}}{\sqrt{\frac{1}{(t_f - t_0)}\int_{t_0}^{t_f}(\delta x)^2\,dt}}\right) \quad (21)$$

Tests were run for different trajectories and initial conditions. For example, for trajectory 4, explained later, the results are shown in the Table below, where the agreement is so that the deviation is around 0.1% in the worst cases. There the integral is performed between a time interval corresponding to 40000 steps. However, in some trajectories, where the variable is to assume very small values, such a trajectory tangent to the ellipsoid ($h = 0$ all times), or else in trajectories where frame transforms have singularities, such in crossing of Earth´s poles, for longitude ϕ, the model may exhibit higher discrepancies. Also, for a huge number of time steps linearization may accumulate computational errors.

TABLE 1 - SAMPLES OF MODEL VALIDATION FIGURES

$\Delta\lambda$ (dB)	-40	Δh (dB)	-45	Δv_E (dB)	-28
$\Delta\phi$ (dB)	-29	Δv_N (dB)	-38	Δv_D (dB)	-42

IV. FILTER IMPLEMENTATION

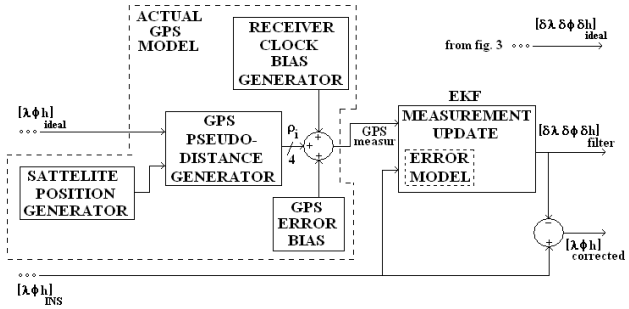

Fig.4: Block diagram for filter implementation and test

Fig. 4 shows the basic diagram for implementation of the EKF performance tests. There we implement a satellite position (X_i, Y_i, Z_i) algorithm to output the 24 satellite positions and also choose the four best satellites with respect to line-of-sight to the vehicle at every time step. The satellites are displaced at six orbital planes, equally spaced in roughly circular orbits, running at orbital radius of about 26560 km. Orbital time is around 11.967 hours. The actual GPS pseudo-distance data $\boldsymbol{Y_k}$ is generated by picking the ideal position $[\lambda\ \phi\ h]^T$ of Fig. 3, plus the signal degradation caused by clock bias drift and Gaussian noises, independently generated, and previously discussed. The EKF estimated error vector $\widehat{\boldsymbol{x}}_{\boldsymbol{k/k}}$ provides correction for the position $[\lambda\ \phi\ h]_{INS}^T$ integrated from INS measurements. Since for test purposes we have the ideal position recorded earlier on, the corrected position can be compared to it, in the sense of (21), evaluating the deviations of filter estimations. We can also monitor how the figure of merit changes along the time steps. Since Δx is normalized, its eventual value decay indicates convergence of the filter estimation as well its consistency in the sense discussed in [5]. And also the average value of $\widetilde{\delta x} - \delta x$ indicates how unbiased the filter implementation is.

IV. RESULTS

Filter performance must be checked on different flight trajectories in order to account for peculiarities of each one. In order to accomplish that we selected four different trajectories among those tested as to characterize its estimation power. Fig. 5 shows graphically those trajectories: In Traj.1 the vehicle goes exactly at the ellipsoid surface ($h = 0$ all the way), starting at $\lambda = 0°$ and $\phi = 90°$. In Traj.2 the vehicle is launched vertically from the same previous spot. In Traj.3 it is launched on an initially tangent line pointing North. And finally in Traj.4 it describes an irregular trajectory where non-zero rotation rates and accelerations combine as to simulate vehicle maneuvering.

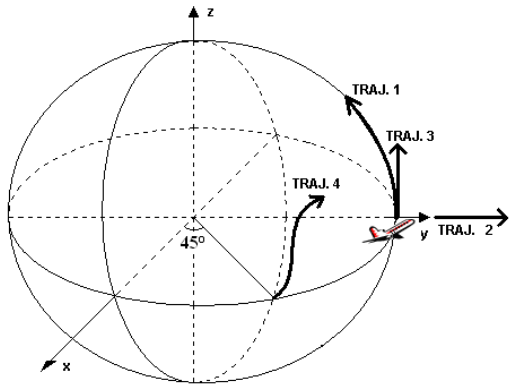

Fig.5: Trajectories employed in EKF evaluation.

Common to all trajectories are: time step $\delta t = 10^{-4}\ s$, gyro bias (all 3 gyros) have $\tau_\omega = 10^4$ and $\sigma_{\varepsilon\omega}^2 = 10^{-8}$ (corresponds to a bias drift of ≈ 1 deg/hr); accelerometer bias (all 3 of them) have $\tau_a = 10^{-3}$ and $\sigma_{\varepsilon a}^2 = 10^{-2}$ (corresponds to a bias drift of $\approx 5\ \mu g$); misalignment matrices $\boldsymbol{M_a}$ and $\boldsymbol{M_g}$ are identity; GPS clock bias variances are typical of temperature-compensated crystal clocks; initial attitude is such that RPY vehicle axes are aligned to NED axes; and GPS error variance $\sigma_{\alpha_i}^2$ was 25 m^2. Table 2 shows the initial conditions and the ideal acceleration and rotation rates employed to generate each of them.

TABLE 2: DATA FOR EACH TESTED TRAJECTORY

	Initial $[v_N\ v_E\ v_D]^T$ (m/s)	$[a_R\ a_P\ a_Y]^T$ m/s^2	$[\omega_R\ \omega_P\ \omega_Y]^T$ rad/s
Traj. 1	$[10^6\ 0\ \ 7.89]^T$	generated file	generated file
Traj. 2	$[0\ 0\ 0]^T$	$[0\ 0\ -10^3]^T$	$[0\ 0\ 0]^T$
Traj. 3	$[0\ 0\ 0]^T$	$[10^3\ 0\ -15]^T$	$[0\ 0\ 0]^T$
Traj. 4	$[100\ 0\ -200]^T$	$[10^3\ 1\ -10^2]^T$	$[0.1\ -0.4\ 0]^T$

Below Fig. 6 shows the EKF error estimation plot for Traj.4 as compared to the actual error plot, for latitude λ, on 1.1×10^6 time steps. Superposition of these plots in one plot would produce superimposed traces which would have made difficult to visually separate the plots. There we see that the estimation tracks the ideal result. This can be quantified by observing Fig. 7, which shows how the RMS deviation $\Delta\lambda$

(dB) evolves. There each point is the normalized RMS deviation integrated over each 5000 time steps interval.

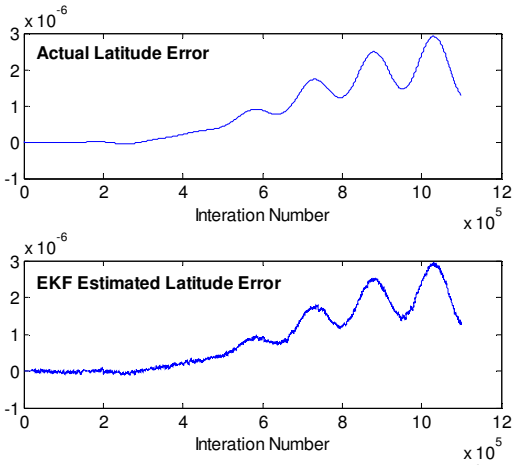

Fig.6: Comparison between actual and EKF estimate of λ.

We see initially that it decays towards a -20 dB level. The filter is unbiased and average of $\widetilde{\delta\lambda} - \delta\lambda$ over larger intervals tends to zero. And $\frac{1}{(t_f - t_0)}\int_{t_0}^{t_f}\left(\widetilde{\delta\lambda} - \delta\lambda\right)^2 dt$ tends to the variance pointed by the first diagonal element in the covariance matrix $\mathbf{\Pi_{k/k}}$, stating filter convergence.

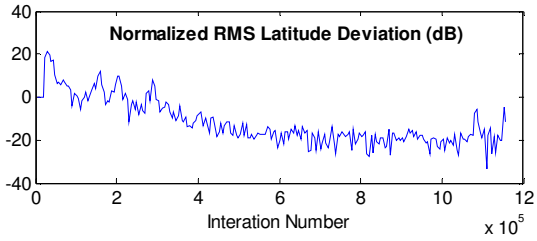

Fig. 7: Filter convergence as represented by Δλ in dB .

Variances for velocities are consistently higher. There is convergence, however to higher RMS deviations. Figs. 8 and 9 show the plots for the north velocity v_N to illustrate that. And Table 4 summarizes the results in all four trajectories for position and velocity errors. It shows the average RMS deviation observed at a larger time interval from the 6×10^5 time step to the final simulation time.

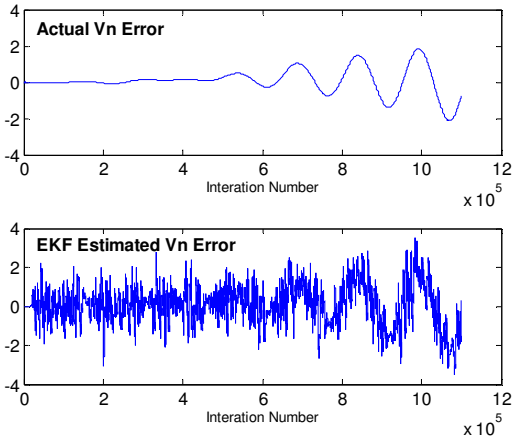

Fig. 8: Comparison between actual and EKF estimate of v_N.

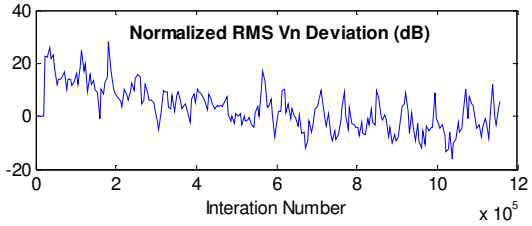

Fig. 9: Filter convergence as represented by Δv_N in dB.

TABLE 3 – FILTER CONVERGENCE DATA

	Traj.1	Traj. 2	Traj. 3	Traj. 4
Δλ (dB)	-33.1	-34.5	-29.1	-19.2
Δϕ (dB)	-35.5	-38.0	-24.7	-27.5
Δh (dB)	-6.5	-14.3	-23.0	-19.9
Δv_N (dB)	-0.4	-14.8	-10.6	-2.3
Δv_E (dB)	-1.4	-15.8	-5.4	-4.2
Δv_D (dB)	-0.5	-9.2	-11.1	-2.5

EKF tracking performance of error curves is to be also verified in specific situations of the trajectories. For example, Fig. 10 shows EKF filter tracking of the ideal longitude error $\delta\phi$ plot in Traj.1, when the vehicle passes through the Earth´s North pole. There, transformation between variables from NED to λ, ϕ, h leads to higher errors due to singularities at the exact North pole spot, however the filter does keep track of the actual error in this situation.

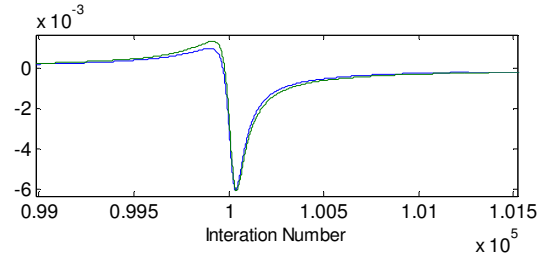

Fig.10: Detail of longitude filter tracking in Traj.1.

IV. CONCLUSIONS

EKF filter shows consistent convergence in several instances in this evaluation. Overall performance shown herein can now be compared to other filter implementations, such the Unscented Kalman Filter (UKF), and particle filters. For velocity and attitude errors, the covariance elements are larger, and a DSP filter post-processing may be applied. For real-time implementations though care must be taken to avoid filter bias due to processing delays, since corrections are to the most up-to-date sample. Furthermore, multi-rate systems, where different sample times of the INS and GPS applies, or even when GPS update is shortly lost, needs extrapolation of GPS data from past samples. We think the EKF approach implemented is suitable to tackle those issues.

BIBLIOGRAPHY

[1] M. S. Grewal, L. R. Weill and A. P. Andrews, "GPS Inertial Navigation And Integration", *John Wiley & Sons Inc.*, c2001.
[2] "Navstar Global Positioning System Surveying", *Engineer Manual*, 1110-1-1003, Department of the Army, US Army Corps of Engineers, Washington DC, July 2003.
[3] J. A. Farrel, and M. Barth , "The Global Positioning System and Inertial Navigation", *McGraw-Hill*, 1998.
[4] V. G. Dikshit, "An investigation of architectures for integration of stand-alone INS and GPS Navigation Systems", *Master Thesis*, July 2006
[5] Y. Bar-Shalom, X-R Li, "Estimation and Tracking: Principles, Techniques, and Software", *Artech House, Boston London*, 1993.