# Integrated INS/GPS Navigation System

Tareg Mahmoud and Bambang Riyanto Trilaksono

School of Electrical Engineering and Informatics, Institut Teknologi Bandung,
Bandung, Indonesia
Tareg_8@yahoo.com, Briyanto@lskk.ee.itb.ac.id

*Abstract:* Navigation equipment specifications differs in the update rate, accuracy, budget, reliability, size and mass. In some applications in order to meet navigation system requirement, a dead reckoning equipment i.e. Inertial Navigation System INS is could be integrated with one or many position fixing equipment, i.e. Global Navigation Satellite System GNSS. INS and GPS have different benefits and drawbacks, and they complement each other when integrating them to provide a navigation solution with higher bandwidth, and long-term and short-term accuracy. This research investigates the performance of an integrated system GPS/INS(MEMS) when changing the algorithm update rate, and compare between different integration algorithm namely loosely and tightly integration.

*Keywords*: INS, GPS, Loosely couple, tightly couple

## 1. Introduction

Concise Oxford Dictionary defines navigation as ‘‘any of several methods of determining or planning a ship’s or aircraft’s position and course by geometry, astronomy, radio signals, etc.” A navigation system may be used in different plant (aircraft, ship, car, etc.) and in different ways (automatically or manually); For example, dynamical motion of a car vehicle is relatively lower than a transport aircraft in terms of velocity and angular motion. The navigation system in the car is used by human driver however in a transport aircraft an autopilot need a navigational signal with a determined specification, and technical requirement, including reliability and safety requirement. A military application also has special requirement such as immunity from jamming. The navigation system requirements in terms update rate, accuracy, budget, reliability, security, size and mass differ based on the plant and usage.

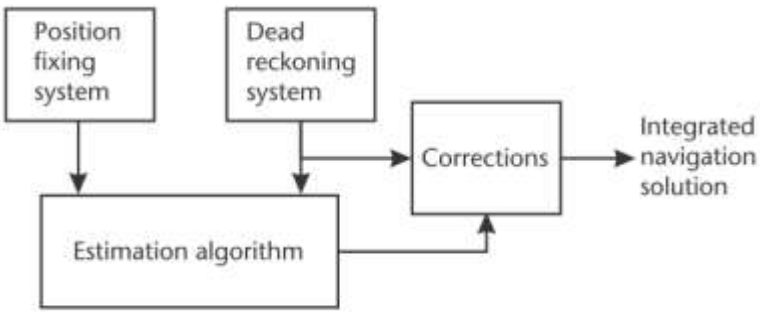



Figure 1. Typical integrated navigation architecture [1]

There are two categories of navigation systems based on technique it uses. The first is based on position fixing technique to determine position then the other navigational data could be obtained. Position fixing techniques uses bearing and/or distance from known objects to determine location, terrestrial land marks, map and magnetic compass, laser, radio signal and radar are used to determine range and/or bearing in position fixing navigation techniques. The second category of navigation system is known as dead reckoning, in which a device is used to acquire the change in position and/or velocity and integrate it, given that initial position is known.

Received: May 18th, 2017. Accepted: September 23rd, 2018
DOI: 10.15676/ijeei.2018.10.3.6

Device like odometer, underwater turbine, Doppler radar, accelerometer and gyro are used as measurement device in dead reckoning navigation system.

For some application the two navigation techniques are integrated to give one navigational solution, a typical integration architecture is shown in figure 1, where the output of position fixing device is corrected to give the integrated navigation solution, a dead reckoning device is added with Kalman filter to provide the necessary corrections.

A common example of integrating two navigation system is utilizing the Global Navigation Satellite System GNSS (position fixing navigation system) to aid a dead reckoning system i.e. Inertial Navigation System INS. The INS have advantages over GNSS like errors over short-term position are relatively small, update rate higher, and not subjection to radio jamming. In other hand GNSS have an advantage over INS like Long-term position errors do not degrade with time, lower cost, and GNSS do not require another means to initialize navigational solution as dead reckoning navigation system. Moreover, the error characteristics of position fixing and dead reckoning differ which allow the use powerful error estimation algorithms as the Kalman filter. So the integrated navigational system will inherent the good feature of GNSS and INS, and have more accurate navigational solation.

For decades, INS and GNSS integration has been investigated, loosely and tightly integration methodology which utilize Kalman filter are successfully implemented for INS/GNSS integration [3,4,5,6,7,8,9]. Kalman filter are considered as powerful tool for estimation and data fusion, (optimum in statistical sense) given that measurement model and its stochastic are known [10].

However, in the rapid growth of embedded system technology, new embedded system feature has been developed to increase the computational speed of algorithm (recently, ARM developed cortex-M4 processor with floating point feature); as the computational speed affect the Kalman filter performance [1], only few studies show the effect of algorithm update rate on the performance of INS/GPS integration. Moreover, as the INS/GPS coupling algorithm complicity differ [2], only few studies compare the performance of different integration algorithm which help in choosing appropriate algorithm for a certain application. This research investigates the performance of an integrated system GPS/INS(MEMS) implemented on embedded microcontroller when changing the algorithm update rate, and compare between different integration algorithm namely loosely and tightly integration.

In this paper, three different navigation method has been simulated and compared namely loosely coupled INS/GPS, tightly coupled INS/GPS, and INS. Moreover, the loosely coupled INS/GPS integration algorithm has been tested using Cortex-M4 for different update rate, results have been provided and discussed.

Outline of this paper is as follows. Inertial measurement unit error model is presented in section 2. Description of inertial navigation unit is discussed in section 3, followed by satellite navigation model and solution in section 4. In section 5, INS/GPS integration, loosely and tightly integration architecture is discussed along with Kalman filter algorithm. Simulation for navigation method is presented in section 6, followed by loosely couple integration testing in section 7. Finally, conclusion is written in section 8.

## 2. Inertial Measurement Unit Error model

Inertial sensors contain accelerometers which measure the specific force $\mathbf{f}_{ib}^{b}$, and gyroscopes (gyros) which measures the angular rate $\boldsymbol{\omega}_{ib}^{b}$.

The inertial measurement in this research consider MEMS technology. MEMS sensors are more light and small compared to the conventional mechanical gyro, in addition it exhibits greater shock tolerance; However, it gives poor performance. MEMS principally uses quartz and silicon sensor combined in a single silicon wafer.

As a measurement sensor an accelerometer and gyro are subjected bias errors, scale factor errors, and cross-coupling errors, in addition to the random noise. Moreover, angular rate-acceleration cross-sensitivity and higher order errors and may also occur.

Following sensor calibration and compensation, the IMU accelerometer and gyro biases, could be denoted by the vectors $\boldsymbol{b_a} = (b_{a,x}, b_{a,y}, b_{a,z})$ and $\boldsymbol{b_g} = (b_{g,x}, b_{g,y}, b_{g,z})$, respectively. Accelerometer and gyro scale factor errors could be denoted by the vectors $\boldsymbol{s_a} = (s_{a,x}, s_{a,y}, s_{a,z})$ and $\boldsymbol{s_g} = (s_{g,x}, s_{g,y}, s_{g,z})$, respectively.

Another type of error found in accelerometer and gyro is the cross-coupling errors, which are resulted from the misalignment between the orthogonal axis of the body frame and the sensor sensitive axis. The notation $m_{a,\alpha\beta}$, denote the cross-coupling coefficient of $\beta$-axis specific force sensed by the $\alpha$-axis accelerometer, and $m_{g,\alpha\beta}$, is used denotes the coefficient of $\beta$-axis angular rate which are sensed by the $\alpha$-axis gyro.

The following matrix express the scale factor error and cross-coupling error, when the accelerometer and gyro triad are nominally orthogonal to each other:

$$\boldsymbol{M_a} = \begin{pmatrix} s_{a,x} & m_{a,xy} & m_{a,xz} \\ m_{a,yx} & s_{a,y} & m_{a,yz} \\ m_{a,zx} & m_{a,zy} & s_{a,z} \end{pmatrix} \quad (1)$$

$$\boldsymbol{M_g} = \begin{pmatrix} s_{g,x} & m_{g,xy} & m_{g,xz} \\ m_{g,yx} & s_{g,y} & m_{g,yz} \\ m_{g,zx} & m_{g,zy} & s_{g,z} \end{pmatrix} \quad (2)$$

The vectors $\boldsymbol{w_a} = (w_{a,x}, w_{a,y}, w_{a,z})$ denote the random noise of IMU accelerometer, and $\boldsymbol{w_g} = (w_{g,x}, w_{g,y}, w_{g,z})$, denote the random noise of IMU gyro. The equation below shows to what extend the error contribute to the IMU output:

$$\tilde{\boldsymbol{f}}_{ib}^{b} = \boldsymbol{b}_a + (\boldsymbol{I}_3 + \boldsymbol{M}_a)\boldsymbol{f}_{ib}^{b} + \boldsymbol{w}_a \quad (3)$$

$$\tilde{\boldsymbol{\omega}}_{ib}^{b} = \boldsymbol{b}_g + (\boldsymbol{I}_3 + \boldsymbol{M}_g)\boldsymbol{\omega}_{ib}^{b} + \boldsymbol{G}_g\boldsymbol{f}_{ib}^{b} + \boldsymbol{w}_g \quad (4)$$

Where $\tilde{\boldsymbol{f}}_{ib}^{b}$ is IMU-output specific force, and $\tilde{\boldsymbol{\omega}}_{ib}^{b}$ are the and angular rate vectors, $\boldsymbol{f}_{ib}^{b}$ and $\boldsymbol{\omega}_{ib}^{b}$ are the true quantity, and $\boldsymbol{I}_3$ represents the identity matrix.

## 3. Inertial Navigation Unit

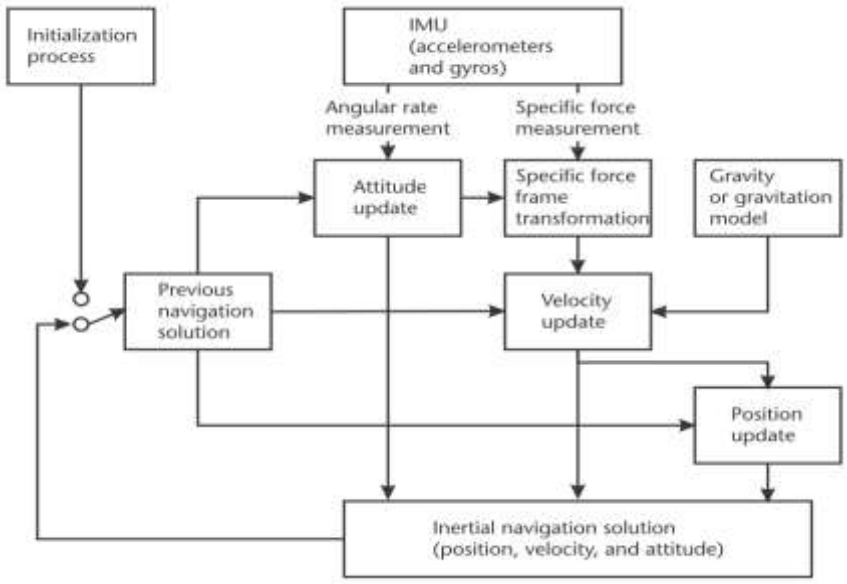



Figure 2. Schematic of an inertial navigation processor [1]

The inertial navigation system (INS), sometimes called inertial navigation unit (INU) categorized as a dead reckoning navigation system. INS as shown in figure 2 compromised from

an inertial measurement unit and a navigation processor. The navigation processor basically integrate the IMU output to produce a navigation solution. To compute a navigation solution four steps should be followed: attitude update, then transformation of the specific-force about the resolving axes, followed by velocity update, and finally position update. Moreover, the gravitational model is included for transforming the specific force to acceleration

Equation (5) to (8) shows how to use the measurement of the angular-rate and specific-force in the interval from $t$ to $t+\tau_i$ to update the Earth-referenced attitude, velocity, and position. The above four steps are described below. The suffixes (−) denote the value at the beginning of processing cycle at time $t$, and the suffixes (+) denote the value at the end of processing cycle $t+\tau_i$.

For Attitude Update the following equation apply [1]

$$\boldsymbol{C}_b^e(+) \approx \boldsymbol{C}_b^e(-)\left(\boldsymbol{I}_3 + \boldsymbol{\Omega}_{ib}^b \tau_i\right) - \boldsymbol{\Omega}_{ie}^e \boldsymbol{C}_b^e(-)\tau_i \quad (5)$$

Where, $\boldsymbol{C}_b^e$ is the Earth-frame coordinate transformation matrix, and $\boldsymbol{\Omega}_{ie}^e$ is the angular rate vector skew-symmetric matrix. `

And for Specific-Force Frame Transformation the following equation are used

$$\boldsymbol{f}_{ib}^e(t) = \boldsymbol{C}_b^e(t)\boldsymbol{f}_{ib}^b(t) \approx \frac{1}{2}(\boldsymbol{C}_b^e(-) + \boldsymbol{C}_b^e(+))\boldsymbol{f}_{ib}^b \quad (6)$$

To give Velocity Update the following equation is

$$\boldsymbol{v}_{eb}^e(+) \approx \boldsymbol{v}_{eb}^e(-) + (\boldsymbol{f}_{ib}^e + \boldsymbol{g}_b^e(\boldsymbol{r}_{eb}^e(-)) - 2\boldsymbol{\Omega}_{ie}^e \boldsymbol{v}_{eb}^e(-))\tau_i \quad (7)$$

Where, $\boldsymbol{g}_b^e$ represent the acceleration due to gravity acting on the body resolved in ECEF frame. For Position Update the following equation is

$$\boldsymbol{r}_{eb}^e(+) = \boldsymbol{r}_{eb}^e(-) + (\boldsymbol{v}_{eb}^e(-) + \boldsymbol{v}_{eb}^e(+))\frac{\tau_i}{2} \quad (8)$$

Where, $\boldsymbol{v}_{eb}^e$ and $\boldsymbol{r}_{eb}^e$ represent the velocity of the body in relation to the ECEF frame, respectively,

## 4. Satellite Navigation Systems model and solution

Table 1. GPS Satellite Orbit Ephemeris Parameters [2]

| *symbol* | *Description* | *Resolution(LSB)* |
|---|---|---|
| $t_{oe}$ | Reference time of the ephemeris | 16 seconds |
| $M_o$ | Mean anomaly at the reference time | $1.46 \times 10^{-9}$ rad |
| $e_o$ | Eccentricity of the orbit | $1.16 \times 10^{-10}$ (unit less) |
| $a^{1/2}$ | Square root of the semi-major axis | $1.91 \times 10^{-6}$ $m^{-0.5}$ |
| $\Omega_0$ | Right ascension of ascending node of orbital plane at the weekly epoch | $1.46 \times 10^{-9}$ rad |
| $i_0$ | Inclination angle at the reference time | $1.46 \times 10^{-9}$ rad |
| $\Omega$ | Argument of perigree | $1.46 \times 10^{-9}$ rad |
| $\Delta_n$ | Mean motion difference from computed value | $3.57 \times 10^{-13}$ rad $s^{-1}$ |
| $\dot{\Omega}_d$ | Rate of change of longitude of the ascending node at the reference time | $3.57 \times 10^{-13}$ rad $s^{-1}$ |
| $\dot{i}_d$ | Rate of inclination | $3.57 \times 10^{-13}$ rad $s^{-1}$ |
| $C_{uc}$ | Amplitude of the cosine harmonic correction term to the argument of latitude | $1.86 \times 10^{-9}$ rad |
| $C_{us}$ | Amplitude of the sine harmonic correction term to the argument of latitude | $1.86 \times 10^{-9}$ rad |
| $C_{rc}$ | Amplitude of the cosine harmonic correction term to the orbit radius | 0.0313 m |
| $C_{rs}$ | Amplitude of the sine harmonic correction term to the orbit radius | 0.0313 m |
| $C_{ic}$ | Amplitude of the cosine harmonic correction term to the angle of inclination | $1.86 \times 10^{-9}$ rad |
| $C_{is}$ | Amplitude of the sine harmonic correction term to the angle of inclination | $1.86 \times 10^{-9}$ rad |

Global navigation satellite systems GNSS is a fixing point navigation system, which provide the user with a three-dimensional positioning solution by passive ranging using radio signals transmitted by satellites orbiting around the earth.

There are many GNSS system orbiting around the earth, one of them is (NAVSTAR) Global Positioning System (GPS), which belong to the U.S. government.

In order to determine satellite position and velocity, GPS transmit ephemeris (satellite orbital data, 16 quasi-Keplerian parameters which are listed in Table 1, including the resolution in terms of the least significant bit, this is applicable to the legacy GPS navigation data message). The ephemeris parameters describe the orbit during the interval of time (at least 1 h) from which the parameters are transmitted.

The center of the satellite's antennas ECEF coordinates are calculated using a variation of the equations shown in Table 2.

The true range between satellite and user antenna $\rho_T$ could be obtained by measuring the time between the satellite signal transmission $\boldsymbol{t_{st}}$ , and the user's antenna time signal arrival $\boldsymbol{t_{sa}}$ and divided by the speed of light. The user equipment obtains pseudo-range measurements because of multiple inaccuracies and uncertainties, user-equivalent range error (UERE) represent the uncertainty of each pseudo-range measurement.

For satellite *j* the pseudo-range and pseudo-range rate which are measured by the user equipment are given by [1]:

$$\rho_T = \rho Tj + \delta\rho ij + \delta\rho tj - \delta\rho sj + \delta\rho rc \quad (9)$$

$$\dot{\rho}_{Rj} = \dot{\rho}_{Tj} + \delta\dot{\rho}_{ij} + \delta\dot{\rho}_{tj} - \delta\dot{\rho}_{sj} + \delta\dot{\rho}_{rc} \quad (10)$$

Equation (9) and (10) are used to model and simulate user equipment measurement.

Table 2. Algorithm for computing satellite position and velocity [2]

| | |
|---|---|
| $\mu = 3.986005 \times 10^{14} m^3/s^2$ | WGS84 value of earth's universal gravitational parameter |
| $\dot{\Omega}_e = 7.292115167 \times 10^{-5}$ rad/s | WGS84 value of earth's rotation rate |
| $a = (\sqrt{a})^2$ | Semi major axis |
| $n_0 = \sqrt{\mu / a^3}$ | Computed mean motion, rad/s |
| $t_k = t - t_{oe}^a$ | Time from ephemeris reference epoch |
| $n = n_0 + \Delta_n$ | Corrected mean motion |
| $M_k = M_0 + nt_k$ | Mean anomaly |
| $M_k = E_k - e\,sinE_k$ | Kepler's equation for eccentric anomaly |
| $f_k = \cos^{-1}(\frac{cosE_k - e}{1 - e\,cosE_k})$ | True anomaly from cosine |
| $f_k = \sin^{-1}(\frac{\sqrt{1-e^2}\sin E_k}{1 - e^2\cos E_k})$ | True anomaly from sine |
| $E_k = \cos^{-1}(\frac{e + cosf_k}{1 + e\,cosf_k})$ | Eccentric anomaly from cosine |
| $\phi_k = f_k + \omega$ | Argument of latitude |
| $\delta\mu_k = C_{\mu c}\cos 2\phi_k + C_{\mu s}\sin 2\phi_k$ | Second-harmonic correction to argument of latitude |
| $\delta g_k = C_{rc}\cos 2\phi_k + C_{rs}\sin 2\phi_k$ | Second-harmonic correction to radius |
| $\delta g_k = C_{ic}\cos 2\phi_k + C_{is}\sin 2\phi_k$ | Second-harmonic correction to inclination |
| $\mu_k = \phi_k + \delta\mu_k$ | Corrected argument of latitude |
| $r_k = a(1 - e\,cosE_k) + \delta r_k$ | Corrected radius |
| $i_k = i_0 - i_k + (IDOT)t_k$ | Corrected inclination |
| $x'_k = r_k\cos\mu_k$ | X coordinate in orbit plane |
| $y'_k = r_k\sin\mu_k$ | Y coordinate in orbit plane |
| $\Omega_k = \Omega_0 + (\Omega + \dot{\Omega}_e)t_k - \dot{\Omega}_e t_{oe}$ | Corrected longitude of ascending node |
| $x_k = x'_k\cos\Omega_k - y'_k\cos i_k\sin\Omega_k$ | ECEF X coordinate |
| $y_k = x'_k\sin\Omega_k - y'_k\cos i_k\cos\Omega_k$ | ECEF Y coordinate |
| $z_k = y'_k\sin i_k$ | ECEF Z coordinate |
| $v_{k,x} = \dot{x}'_k\cos\Omega_k - y'_k\cos i_k\sin\Omega_k + \dot{i}_k$ $y'_k\sin i_k\sin\Omega_k - \dot{\Omega}_d\,y_k$ | ECEF X velocity |
| $v_{k,y} = \dot{x}'_k\sin\Omega_k + y'_k\cos i_k\cos\Omega_k - \dot{i}_k$ $y'_k\sin i_k\cos\Omega_k + \dot{\Omega}_d\,y_k$ | ECEF Y velocity |
| $v_{k,z} = \dot{y}'_k\sin i_k + \dot{i}_k\,y'_k\cos i_k$ | ECEF Z velocity |

Where $\delta\rho_{ij}$ and $\delta\rho_{tj}$ are, the ionosphere and troposphere propagation errors, respectively. $\delta\rho_{sj}$ is the range error due to the satellite clock, $\delta\rho_{rc}$ is the range error due to the receiver clock, and $\delta\dot{\rho}_{ij}$, $\delta\dot{\rho}_{tj}$, $\delta\dot{\rho}_{sj}$ and $\delta\dot{\rho}_{rc}$ are their range-rate counterparts.

A set of the pseudo-range measurements cannot be used to easily used to drive the position solution in an analytical way. Therefore, the equations are linearized by performing a Taylor expansion about a predicted user position, $\mathbf{r}_{ea}^{eP}$, and clock offset $\delta\rho_{rc}^{P}$, the least-squares solution is used. For the ECEF frame [1]:

$$\begin{pmatrix} \hat{\mathbf{r}}_{eb}^{e}\,(t_{sa}) \\ \delta\hat{\rho}_{rc}\,(t_{sa}) \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{ea}^{eP} \\ \delta\rho_{rc}^{P} \end{pmatrix} + \,(\mathbf{G}_e^{\mathrm{T}}\,\mathbf{G}_e)^{-1}\mathbf{G}_e^{\mathrm{T}} \begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^{P} \\ \tilde{\rho}_{C2} - \rho_{C2}^{P} \\ \vdots \\ \tilde{\rho}_{Cn} - \rho_{Cn}^{P} \end{pmatrix} \tag{11}$$

$$\begin{pmatrix} \hat{\mathbf{v}}_{eb}^{e}\,(t_{sa}) \\ \delta\hat{\rho}_{rc}\,(t_{sa}) \end{pmatrix} = \begin{pmatrix} \mathbf{v}_{ea}^{eP} \\ \delta\dot{\rho}_{rc}^{P} \end{pmatrix} + \,(\mathbf{G}_e^{\mathrm{T}}\,\mathbf{G}_e)^{-1}\mathbf{G}_e^{\mathrm{T}} \begin{pmatrix} \tilde{\rho}_{C1} - \dot{\rho}_{C1}^{P} \\ \tilde{\rho}_{C2} - \dot{\rho}_{C2}^{P} \\ \vdots \\ \tilde{\rho}_{Cn} - \dot{\rho}_{Cn}^{P} \end{pmatrix} \tag{12}$$

Where

$$\rho_{rc}^{P} = \sqrt{\left[\hat{\mathbf{r}}_{esj}^{e}(\tilde{t}_{st,j}) - \mathbf{r}_{ea}^{eP}\right]^{\mathrm{T}}\left[\hat{\mathbf{r}}_{esj}^{e}(\tilde{t}_{st,j}) - \mathbf{r}_{ea}^{eP}\right]} + \delta\rho_{rc}^{P} + \,\delta\rho_{ie,j} \tag{13}$$

$$\dot{\rho}_{rc}^{P} = \mathbf{u}_{as,j}^{eP}\left[\hat{\mathbf{v}}_{esj}^{e}(\tilde{t}_{st,j}) - \mathbf{v}_{ea}^{eP}\right] + \delta\rho_{rc}^{P} + \,\delta\rho_{ie,j} \tag{14}$$

And the geometry matrix, G is obtained as follow.
Measurement matrix, H, is

$$\mathbf{H} = \left[\begin{matrix} \frac{\partial\rho_1}{\partial x_{ea}^{e}} & \frac{\partial\rho_1}{\partial y_{ea}^{e}} & \frac{\partial\rho_1}{\partial z_{ea}^{e}} & \frac{\partial\rho_1}{\partial\rho_{rc}} \\ \frac{\partial\rho_2}{\partial x_{ea}^{e}} & \frac{\partial\rho_2}{\partial z_{ea}^{e}} & \frac{\partial\rho_2}{\partial z_{ea}^{e}} & \frac{\partial\rho_2}{\partial\rho_{rc}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial\rho_n}{\partial x_{ea}^{e}} & \frac{\partial\rho_n}{\partial y_{ea}^{e}} & \frac{\partial\rho_n}{\partial z_{ea}^{e}} & \frac{\partial\rho_n}{\partial\rho_{rc}} \end{matrix}\right]\Bigg|_{\mathbf{r}_{ia}^{i}=\mathbf{r}_{ia}^{iP}} \tag{15}$$

Differentiating (15) with respect to the user position and clock offset result

$$\mathbf{H} = \begin{pmatrix} -u_{as,1,x}^{eP} & -u_{as,1,y}^{eP} & -u_{as,1,z}^{eP} & 1 \\ -u_{as,2,x}^{eP} & -u_{as,2,y}^{eP} & -u_{as,2,z}^{eP} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -u_{as,n,x}^{eP} & -u_{as,n,y}^{eP} & -u_{as,n,z}^{eP} & 1 \end{pmatrix} = \mathbf{G}_e \tag{16}$$

**5. INS/GPS integration**

Here, two types of integration architecture are discussed loosely coupled, and tightly coupled. The integration algorithm compares the outputs of GPS user equipment with the inertial navigation solution, then estimates corrections to the inertial position, velocity, attitude solution, acceleration bias and gyro bias. Kalman filter is used as an estimation algorithm. The obtained inertial navigation solution after correction represent the integrated navigation solution. In situation of GPS signal loss this architecture grantee producing integrated navigation solution.

The INS/GNSS integrated navigation architecture could be classified as an open loop and closed loop. As shown in figure 3 the open loop architecture integration algorithm estimates the error in the INS measurement and provide the solution. While in the closed loop architecture the INS drive the final solution and utilizing the algorithm output to drive the final navigation solution.

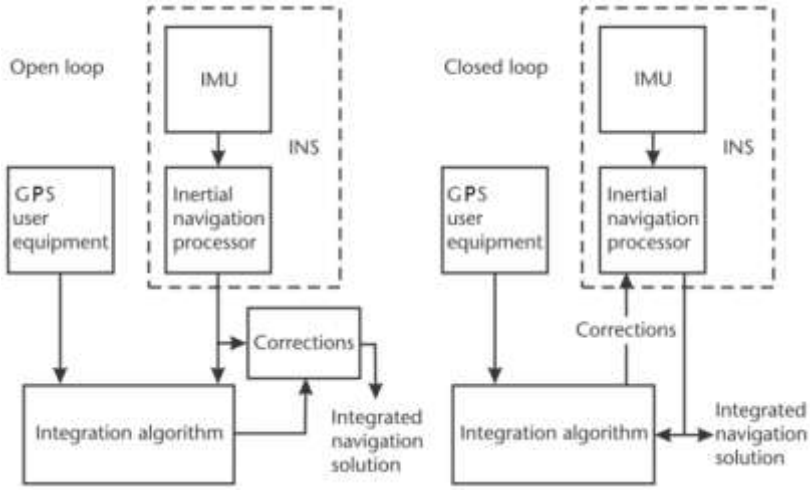



Figure 3. Open- and closed-loop INS correction architectures

*A. Kalman Filter Algorithm*

The Kalman filter can be described as a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance—when some presumed conditions are met.

The Kalman Filter basic technique was invented by R. E. Kalman in 1960 and since has been developed further by numerous authors. The Kalman filter make an estimation to a process by using a form of feedback control: the process state is estimated at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two phase: time update phase and measurement update phase. The current state and error covariance estimates fall in time update phase, which are responsible for projecting forward (in time) to obtain the a priori estimates for the next time step. While, the measurement update equations are responsible for the feedback—i.e. for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

The true state vector, x(t), at time, t, of any Kalman filter is described by the following dynamic model [1]:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}_s(t) \quad (17)$$

Where $\mathbf{w}_s(t)$ is the system noise vector, F(t) is the system matrix and G(t) is the system noise distribution matrix.

The expectation value of the true state vector, x(t), is the estimated state vector, $\hat{\mathbf{x}}(t)$. The expectation value of the system noise vector $\mathbf{w}_s(t)$, is zero, as the noise is assumed to be of zero mean. F(t) and G(t) are assumed to be known functions.

The discrete and continuous forms of the Kalman filter are equivalent, with $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}(t)$ and $\hat{\mathbf{x}}_{k-1} = \hat{\mathbf{x}}(t - \boldsymbol{\tau}_s)$.

Kalman filter measurement vector, z(t), is modeled as a linear function of the true state vector, x(t), and the white noise sources, $\mathbf{w}_m(t)$. Thus,

$$\mathbf{z}(t) = \mathbf{H}(t)\mathbf{x}(t) + \mathbf{w}_m(t) \quad (18)$$

Where H(t) is the measurement matrix and is obtained from the properties of the system.

The Kalman filter algorithm steps are described below:

1. Transition matrix calculation $\Phi_{k-1}$ :

Transition matrix shows the state vector changes with time as a function of the dynamics of the system modeled by the Kalman filter.

$$\Phi_{k-1} \approx \exp(\boldsymbol{F}_{k-1}\boldsymbol{\tau}_s) \quad (19)$$

The transition matrix is usually computed as a power-series expansion of the system matrix, F, and propagation interval $\boldsymbol{\tau}_s$.

2. System noise covariance matrix calculation, $Q_{k-1}$:
   System noise covariance matrix shows how the uncertainties of the state estimates increase with time as a result of noise sources in the Kalman filter system model (such as unmeasured dynamics and instrument noise).
   Usually the system noise covariance matrix form as diagonal and constant matrix.
3. State vector estimate propagation from $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}$ to $\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{+}$:
   The time-propagated state estimates are denoted $\hat{\boldsymbol{x}}\boldsymbol{k}-$, its counterparts following the measurement update is denoted $\hat{\boldsymbol{x}}\boldsymbol{k}+$, The subscript k is used to denote the iteration. The state vector estimate is propagated through time using:
   $$\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-} = \Phi_{k-1}\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{+} \quad (20)$$
4. Error covariance matrix propagation from $\hat{\boldsymbol{P}}_{\boldsymbol{k-1}}^{+}$ to $\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{-}$ :
   The time-propagated covariance matrix is denoted $\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{-}$, its counterparts following the measurement update is denoted $\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{+}$ , and the subscript *k* is used to denote the iteration. The covariance matrix is propagated through time using:
   $$\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{-} = \Phi_{\boldsymbol{k-1}}\hat{\boldsymbol{P}}_{\boldsymbol{k-1}}^{+}\Phi_{\boldsymbol{k-1}}^{\mathrm{T}} + \boldsymbol{Q}_{\boldsymbol{k-1}} \quad (21)$$
5. Calculate the measurement matrix $H_k$;
   The measurement matrix defines how the measurement vector varies with the state vector. In navigation, $H_k$ is a function of the user kinematics and/or GPS satellite geometry.
6. Calculate the measurement noise covariance matrix, $R_k$:
   The measurement noise covariance matrix, $R_k$, could be assume constant or could be modeled as a function of kinematics or signal-to-noise measurements.
7. Kalman gain matrix calculation $\boldsymbol{K}_k$ :
   Kalman gain determine the weighting of the measurement information in updating the state estimates. The Kalman gain is a function of the ratio between uncertainty of the true measurement, $z_k$, and the uncertainty of the measurement predicted from the state estimates, $H_k\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}$. The Kalman gain matrix is
   $$\boldsymbol{K}_k = \boldsymbol{P}_k^{-}\boldsymbol{H}_k^{\mathrm{T}}(\boldsymbol{H}_k\boldsymbol{P}_k^{-}\boldsymbol{H}_k^{\mathrm{T}} + \boldsymbol{R}_k)^{-1} \quad (22)$$
8. Formulate the measurement, $z_k$:
   In some applications, the measurement vector is presented in the system modeled by the Kalman filter. In other applications, $z_k$ should be calculated as a function of other system parameters.
9. State vector estimate Update from $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}$ to $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{+}$ :
   The state vector is updated with the measurement vector using
   $$\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{+} = \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-} + \boldsymbol{K}_k(\boldsymbol{z}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}) \quad (23)$$
10. Error covariance matrix update from $\boldsymbol{P}_{\boldsymbol{k}}^{-}$ to $\boldsymbol{P}_{\boldsymbol{k}}^{+}$:
    The error covariance matrix is updated with
    $$\boldsymbol{P}_{\boldsymbol{k}}^{+} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\,\boldsymbol{P}_{\boldsymbol{k}}^{-} \quad (24)$$

For INS/GPS there are interaction between the INS and GPS state in the measurement model, while in system model there is no interaction Therefore, the system, transition, and system noise covariance matrices could be expressed as [1]:

$$\boldsymbol{F} = \begin{pmatrix} \boldsymbol{F}_{\boldsymbol{INS}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{F}_{\boldsymbol{GPS}} \end{pmatrix},$$

$$\Phi = \begin{pmatrix} \Phi_{\boldsymbol{INS}} & \boldsymbol{0} \\ \boldsymbol{0} & \Phi_{\boldsymbol{GPS}} \end{pmatrix}, \boldsymbol{Q} = \begin{pmatrix} \boldsymbol{Q}_{\boldsymbol{INS}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_{\boldsymbol{GPS}} \end{pmatrix} \quad (25)$$

And

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{x}_{\boldsymbol{INS}} \\ \boldsymbol{x}_{\boldsymbol{GPS}} \end{pmatrix} \quad (26)$$

The state vector for INS resolved in the ECEF frame, Kalman filter estimated error in the navigation solution:

$$\boldsymbol{x}_{INS}^{e} = \begin{pmatrix} \delta_{\psi eb}^{e} \\ \delta \boldsymbol{v}_{eb}^{e} \\ \delta \boldsymbol{r}_{eb}^{e} \\ \boldsymbol{b}_{a} \\ \boldsymbol{b}_{\mathrm{g}} \end{pmatrix} \tag{27}$$

And, the system matrix is

$$\boldsymbol{F}_{INS}^{e} = \begin{pmatrix} -\boldsymbol{\Omega}_{ie}^{e} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \widehat{\boldsymbol{C}}_{b}^{e} \\ \boldsymbol{F}_{21}^{e} & -2\boldsymbol{\Omega}_{ie}^{e} & \boldsymbol{F}_{23}^{e} & \widehat{\boldsymbol{C}}_{b}^{e} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & \boldsymbol{I}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} \end{pmatrix} \tag{28}$$

Where;

$$\boldsymbol{F}_{21}^{e} = [-(\widehat{\boldsymbol{C}}_{b}^{e}\widehat{\boldsymbol{f}}_{ib}^{b})\wedge] \tag{29}$$

$$\boldsymbol{F}_{23}^{e} = \frac{2\mathrm{g}_{0}(\hat{L}_{b})}{r_{eS}^{e}(\hat{L}_{b})|\hat{L}_{b}|} \frac{\hat{\boldsymbol{r}}_{eb}^{e}}{|\hat{\boldsymbol{r}}_{eb}^{e}|^{2}} \hat{\boldsymbol{r}}_{eb}^{e\ \mathrm{T}} \tag{30}$$

The INS system noise covariance matrix, $\boldsymbol{Q}_{INS}$, with assumption than the 15 states are estimated as defined by:

$$\boldsymbol{Q}_{INS} = \begin{pmatrix} n_{\mathrm{rg}}^{2}\boldsymbol{I}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & n_{ra}^{2}\boldsymbol{I}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & \boldsymbol{I}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & n_{bad}^{2}\boldsymbol{I}_{3} & \boldsymbol{0}_{3} \\ \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & \boldsymbol{0}_{3} & n_{bgd}^{2}\boldsymbol{I}_{3} \end{pmatrix} \boldsymbol{\tau}_{s} \tag{31}$$

Where $n_{\mathrm{rg}}^{2}$ , $n_{ra}^{2}$ , $n_{bad}^{2}$ , and $n_{bgd}^{2}$ are the power spectral densities of, respectively, the gyro random noise, accelerometer random noise, accelerometer bias variation, and gyro bias variation, with the assumption that all gyros and accelerometers possess noise with equal characteristics.

Consider a measurement $\tilde{m}_{\mathrm{G}}$, acquired from GNSS user equipment and a prediction of that measurement $\tilde{m}_{I}$ , acquired from the raw inertial navigation solution (and the GNSS navigation data message, where appropriate). Using the Kalman filter state vector an estimates of the errors in these measurements $\delta\tilde{m}_{\mathrm{G}}$ and $\delta\tilde{m}_{I}$ , can then be obtained. There are two methods in which these can legitimately be constructed into a Kalman filter measurement, z, and estimate, in turn, these are $z_{G} = \tilde{m}_{\mathrm{G}} - \tilde{m}_{I}$ , $\hat{z}_{k}^{-} = \delta\tilde{m}_{\mathrm{G}} - \delta\tilde{m}_{I}$ (32)

And

$z_{G} = \tilde{m}_{\mathrm{G}}$ , $\hat{z}_{G}^{-} = \tilde{m}_{I} + \delta\tilde{m}_{\mathrm{G}} - \delta\tilde{m}_{I}$ (33)

Here the closed-loop correction of the INS is considered, the predicted measurement is derived from the corrected inertial navigation solution and becomes $\tilde{m}_{I}$ .

*B. Loosely coupled algorithm*

In loosely coupled INS/GPS integration, for INS errors estimation, the GPS user equipment's position and velocity solution are used, in which, the position and/or velocity from the GPS navigation solution is considered as a measurement in the Kalman filter integration. The integrated navigation solution is the taken from INS navigation solution (closed loop architecture), corrected with the Kalman filter estimates of its errors. Figure 5.3 shows a loosely coupled INS/GPS integration architecture.

The measurement innovation vector could be defined as the difference between the GPS and corrected inertial position and velocity solutions, in case that INS and GPS antenna are apart the lever arm from the INS to the GPS antenna is to be defined as $\mathbf{I}_{\mathrm{ba}}^{\mathrm{b}}$,. The coordinate frames for the measurement innovation should match those for the state vector. Thus [1],

$$\delta\boldsymbol{z}_{\mathrm{G},k}^{e-} = \begin{pmatrix} \hat{\boldsymbol{r}}_{eaG}^{e} - \hat{\boldsymbol{r}}_{eb}^{e} - \widehat{\boldsymbol{C}}_{b}^{e}\,\boldsymbol{I}_{ba}^{b} \\ \hat{\boldsymbol{v}}_{eaG}^{e} - \hat{\boldsymbol{v}}_{eb}^{e} - \widehat{\boldsymbol{C}}_{b}^{e}(\hat{\boldsymbol{\omega}}_{ib}^{b}\wedge \boldsymbol{I}_{ba}^{b}) + \boldsymbol{\Omega}_{ie}^{e}\widehat{\boldsymbol{C}}_{b}^{e}\,\boldsymbol{I}_{ba}^{b} \end{pmatrix}_{k} \tag{34}$$

Where the subscript k represents the measurement update iteration; *e* denotes the ECEF frame implementations; and the subscript G denotes GPS.
The loosely coupled measurement matrix for ECEF-frame is

$$\boldsymbol{H}^{e}_{\mathrm{G},k} = \begin{pmatrix} \boldsymbol{H}^{e}_{\mathrm{r1}} & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{H}^{e}_{\mathrm{v1}} & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \boldsymbol{H}^{e}_{\mathrm{v5}} \end{pmatrix}_k \tag{35}$$

Where

$$\boldsymbol{H}^{e}_{\mathrm{r1}} = [(\widehat{\boldsymbol{C}}^{e}_{b}\boldsymbol{I}^{b}_{ba})\wedge] \tag{36}$$

$$\boldsymbol{H}^{e}_{\mathrm{v1}} = [\{\widehat{\boldsymbol{C}}^{e}_{b}(\widehat{\boldsymbol{\omega}}^{b}_{ib}\wedge \boldsymbol{I}^{b}_{ba}) + \boldsymbol{\Omega}^{e}_{ie}\widehat{\boldsymbol{C}}^{e}_{b}\,\boldsymbol{I}^{b}_{ba}\}\wedge] \tag{37}$$

$$\boldsymbol{H}^{e}_{\mathrm{v5}} = \widehat{\boldsymbol{C}}^{e}_{b}[\boldsymbol{I}^{b}_{ba}\wedge] \tag{38}$$

*C. Tightly coupled algorithm*

The GNSS ranging processor's pseudo-range and pseudo-range-rate measurements are used in tightly coupled INS/GNSS integration, which are obtained from code and carrier tracking, respectively. Figure. 5.4 shows a tightly coupled INS/GPS integration architecture.
The GPS Kalman filter is subsumed into the INS/ GPS integration filter. The Kalman filter take the pseudo-range and pseudo-range rates from the GPS ranging processor as measurements input, and use them later to estimate the errors in the INS and GPS systems.
The integrated navigation solution forms the corrected inertial navigation solution (closed loop architecture).

The measurement innovation vector is resulted from the differences between the GNSS measured pseudo-range and pseudo-range rates, in addition to the values predicted from the corrected inertial navigation solution at the same time of validity, estimated receiver clock offset and drift, and navigation-data-indicated satellite positions and velocities. Thus [1]

$$\delta\boldsymbol{z}^{-}_{\mathrm{G},k} = \begin{pmatrix} \delta\boldsymbol{z}^{-}_{\rho,k} \\ \delta\boldsymbol{z}^{-}_{r,k} \end{pmatrix}, \quad \begin{matrix} \delta\boldsymbol{z}^{-}_{\rho,k} = (\tilde{\rho}_{\mathrm{C1}} - \hat{\rho}^{-}_{\mathrm{C1}}, \tilde{\rho}_{\mathrm{C2}} - \hat{\rho}^{-}_{\mathrm{C2}}, \ldots \tilde{\rho}_{\mathrm{C}n} - \hat{\rho}^{-}_{\mathrm{C}n})_k \\ \delta\boldsymbol{z}^{-}_{r,k} = (\tilde{\dot{\rho}}_{\mathrm{C1}} - \hat{\dot{\rho}}^{-}_{\mathrm{C1}}, \tilde{\dot{\rho}}_{\mathrm{C2}} - \hat{\dot{\rho}}^{-}_{\mathrm{C2}}, \ldots \tilde{\dot{\rho}}_{\mathrm{C}n} - \hat{\dot{\rho}}^{-}_{\mathrm{C}n})_k \end{matrix} \tag{39}$$

The position and velocity of the user antenna could be derived from the inertial navigation solution, thus:

$$\hat{\mathbf{r}}^{e}_{ea} = \hat{\mathbf{r}}^{e}_{eb} + \hat{\mathbf{c}}^{e}_{b}\boldsymbol{I}^{b}_{ba} \tag{40}$$

$$\hat{\mathbf{v}}^{e}_{ea} = \hat{\mathbf{v}}^{e}_{eb} + \widehat{\boldsymbol{C}}^{e}_{b}(\widehat{\boldsymbol{\omega}}^{b}_{ib}\wedge \boldsymbol{I}^{b}_{ba}) + \boldsymbol{\Omega}^{e}_{ie}\widehat{\boldsymbol{C}}^{e}_{b}\,\boldsymbol{I}^{b}_{ba} \tag{41}$$

The state vector in tightly coupled integration, are compromised from inertial states, receiver clock offset, and drift. As follow:

$$\mathbf{x}^{e} = \begin{pmatrix} \mathbf{x}^{e}_{INS} \\ \delta\boldsymbol{\rho}_{\ rc} \\ \delta\dot{\boldsymbol{\rho}}_{rc} \end{pmatrix} \tag{42}$$

The measurement matrix could be assembled as follow:

$$\boldsymbol{H}^{e}_{\mathrm{G},k} = \begin{pmatrix} \frac{\partial \boldsymbol{z}_{\rho}}{\partial \delta\psi^{e}_{eb}} & \mathbf{0}_{n,3} & \frac{\partial \boldsymbol{z}_{\rho}}{\partial \delta\boldsymbol{r}^{e}_{eb}} & \mathbf{0}_{n,3} & \frac{\partial \boldsymbol{z}_{\rho}}{\partial \delta\boldsymbol{\rho}_{rc}} & \mathbf{0}_{n,1} \\ \frac{\partial \boldsymbol{z}_{r}}{\partial \delta\psi^{e}_{eb}} & \frac{\partial \boldsymbol{z}_{r}}{\partial \delta\boldsymbol{v}^{e}_{eb}} & \frac{\partial \boldsymbol{z}_{r}}{\partial \delta\boldsymbol{r}^{e}_{eb}} & \mathbf{0}_{n,3} & \frac{\partial \boldsymbol{z}_{\rho}}{\partial \boldsymbol{b}_{\mathrm{g}}} & \mathbf{0}_{n,1} & \frac{\partial \boldsymbol{z}_{r}}{\partial \delta\dot{\boldsymbol{\rho}}_{rc}} \end{pmatrix}_{\boldsymbol{x}=\hat{\boldsymbol{x}}_k} \tag{43}$$

The differentials may be calculated analytically or numerically by perturbing the state estimates and calculating the change in estimate pseudo-range and pseudo-range rate. The dependence of the measurement innovations on the attitude error and of the pseudo-range-rate measurements on the position and gyro errors is weak, so a common approximation to the analytical solution is

$$\mathbf{H}_{G,k}^{e} = \left(\begin{array}{ccccccc} \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{u}_{as,1}^{e\,\mathrm{T}} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{u}_{as,2}^{e\,\mathrm{T}} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{u}_{as,n}^{e\,\mathrm{T}} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \hline \mathbf{0}_{1,3} & \mathbf{u}_{as,1}^{e\,\mathrm{T}} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \mathbf{0}_{1,3} & \mathbf{u}_{as,2}^{e\,\mathrm{T}} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{1,3} & \mathbf{u}_{as,n}^{e\,\mathrm{T}} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \end{array}\right)_{\mathbf{x}=\dot{\mathbf{x}}_{\mathbf{k}}} \tag{44}$$

## 6. Simulation

Three Simulink files were developed to compare different navigation method, INS, loosely coupled INS/GPS integration, and tightly coupled INS/GPS integration. In each file aircraft trajectory has been used to generate IMU and GPS measurement. Figure. 4 shows the input trajectory used in the simulation.

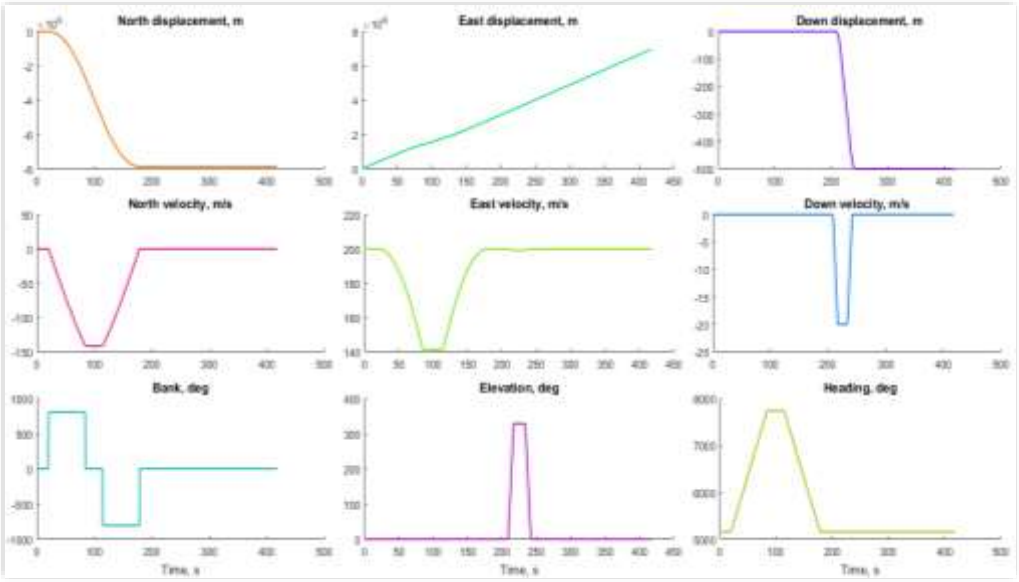



Figure 4. Aircraft 418 second input profile

Tactical grade IMU model has been built in Simulink for simulating specific force and angular rate measurement for the input trajectory, Figure 5 shows the true specific force of body frame with respect to ECEF frame, resolved along body-frame axes $\boldsymbol{f}_{eb}^{b}$.

Figure 6 shows the true angular rate of body frame with respect to ECEF frame, resolved about body-frame axes $\boldsymbol{\omega}_{eb}^{b}$.

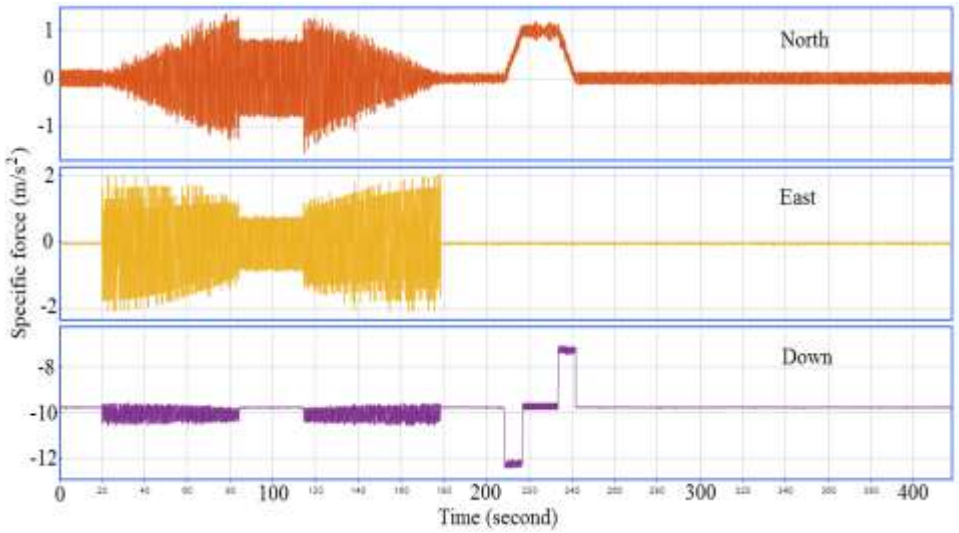



Figure 5. Simulation output of specific force measurement

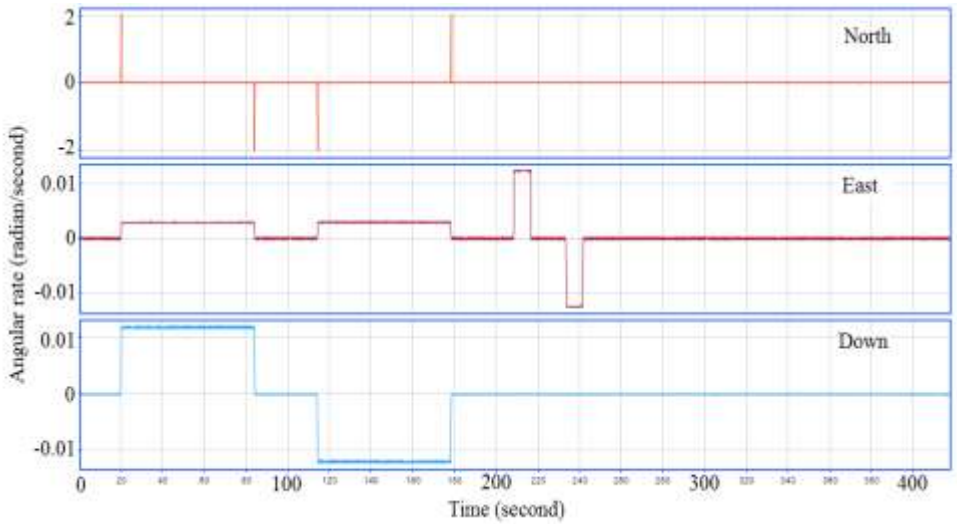



Figure 6. Simulation output of angular rate measurement

In the INS simulation file, no GPS aiding is used to obtain the navigation solution. Figure 7 shows flow chart of the INS Simulink file, Figure 8 shows the Simulink file. The GPS measurement is used only to get the initial position and velocity.

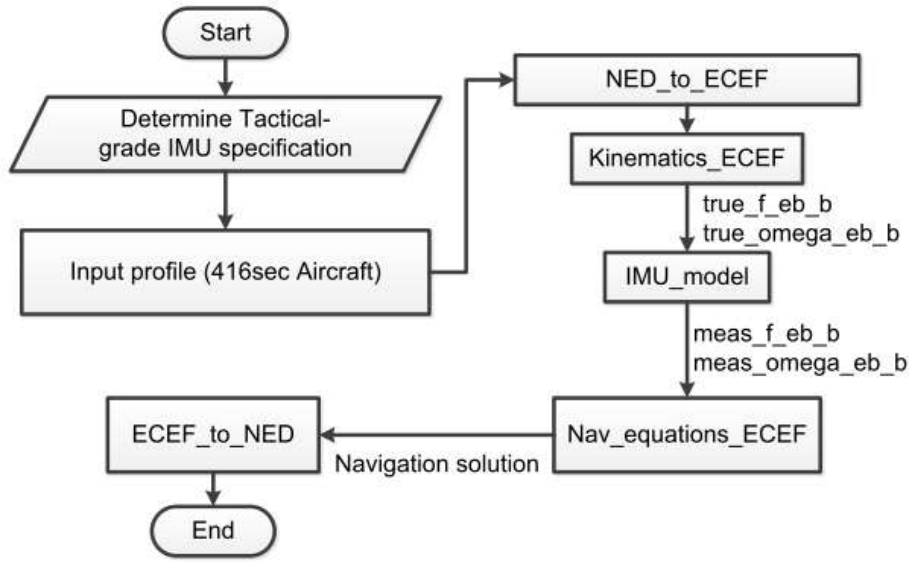



Figure 7. INS Simulation flow chart

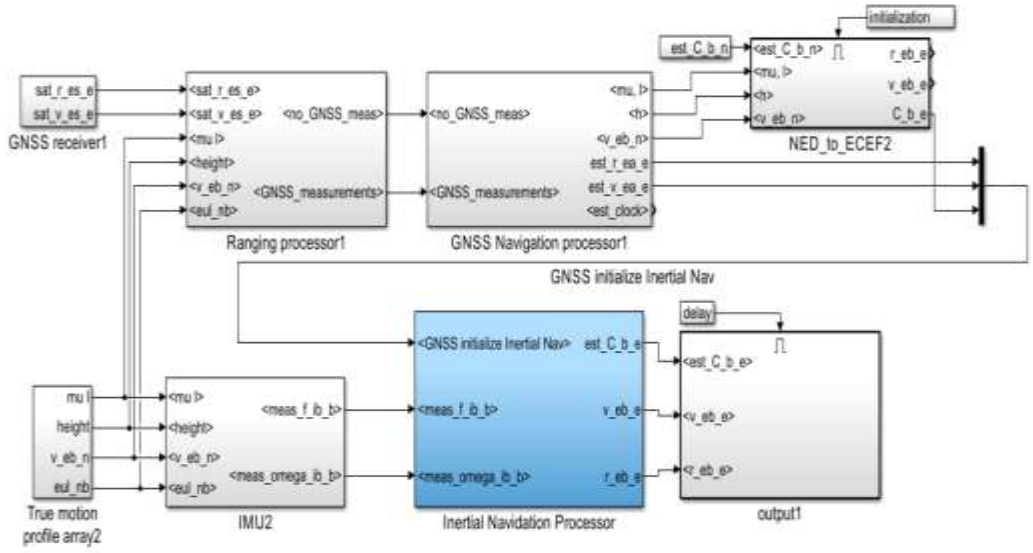



Figure 8. INS Simulink file

The loosely coupled INS/GPS integration simulation file algorithm is shown in figure 8, closed loop integration architecture is used, the INS measurement update rate is 100Hz while GPS measurement update rate is 2Hz, discrete solver were applied.

Simulink file flow chart is shown in Figure 9, it's configured to be applicable for Matlab Coder c-code generation, and Figure 10 shows the Simulink file.

In the simulation file closed loop integration architecture is used, the INS measurement update rate is 100Hz while GPS measurement update rate is 2Hz, discrete solver was applied.

Simulink file flow chart is shown in figure 11, it's configured to be applicable for Matlab Coder C-code generation, and figure 12 shows the Simulink file.

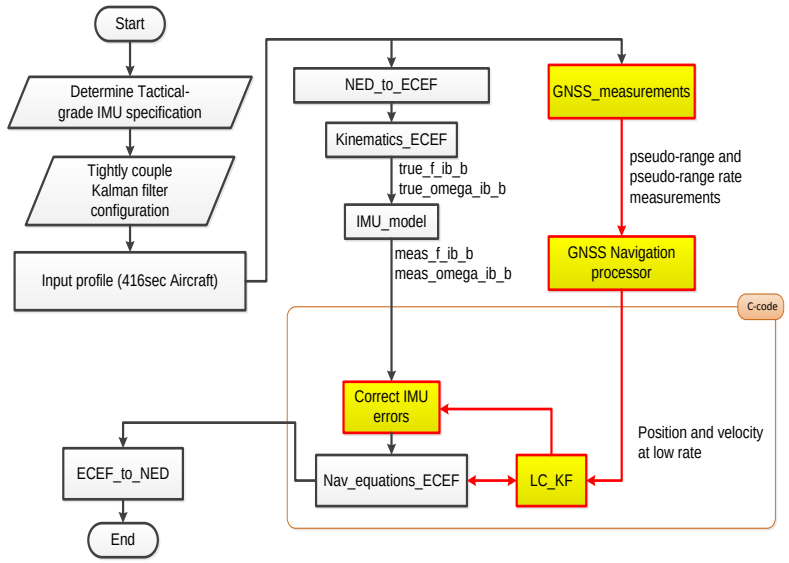



Figure 9. INS/GPS loosely couple integration Simulink file flow chart

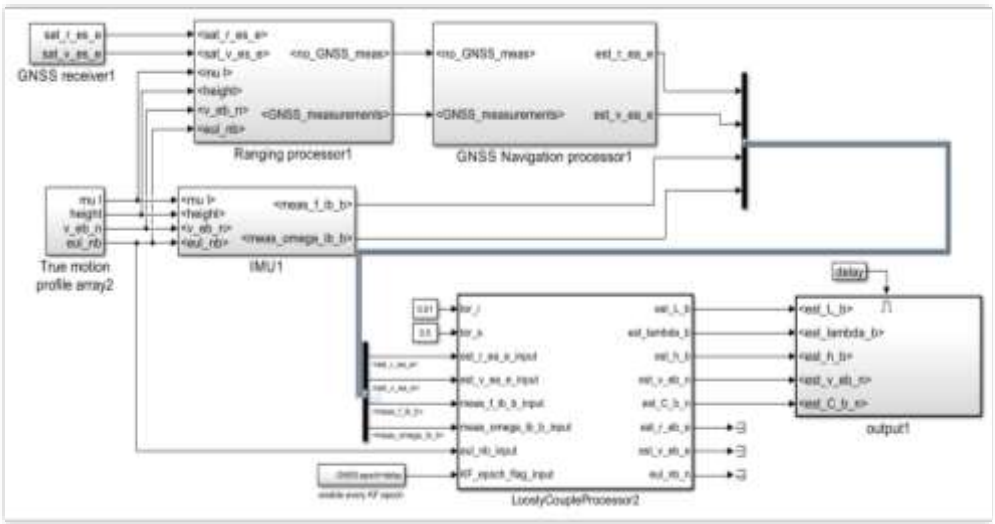



Figure 10. INS/GPS loosely couple integration Simulink file

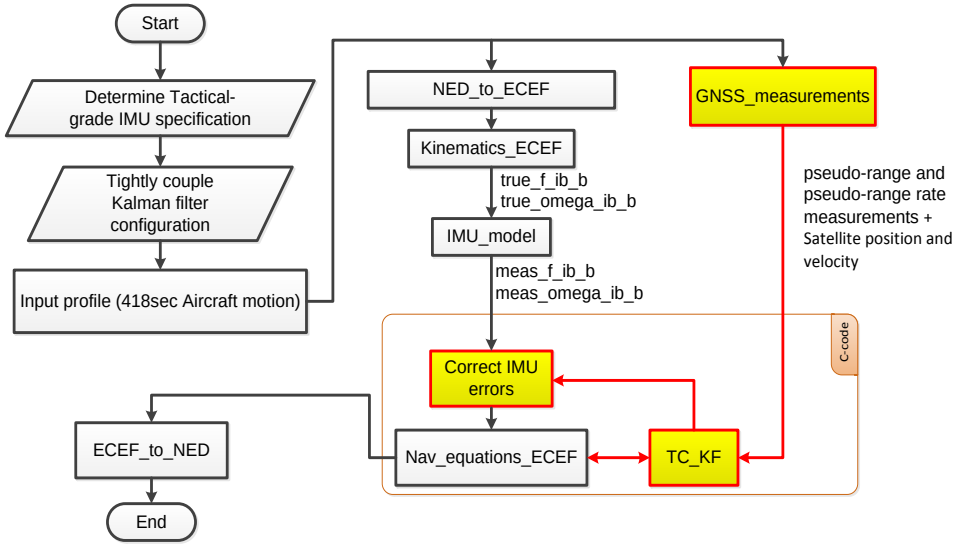



Figure 11. INS/GPS tightly couple integration Simulink flow chart

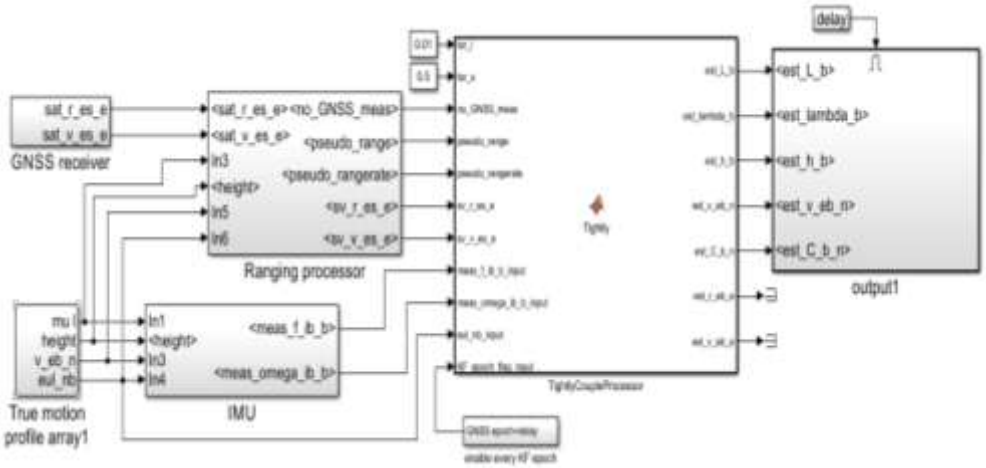



Figure 12. INS/GPS tightly couple integration Simulink file

*Simulation Result*

For the three Simulink file described previously (INS, INS/GPS loosely coupled integration, INS/GPS tightly coupled integration), 418 second simulation is done for aircraft trajectory shown in Figure 6. The absolute error in navigation solution (position and velocity in NED frame) in logarithmic to the base of 10 is shown in Figures 13, 14, 15, 16, 17 and 18.

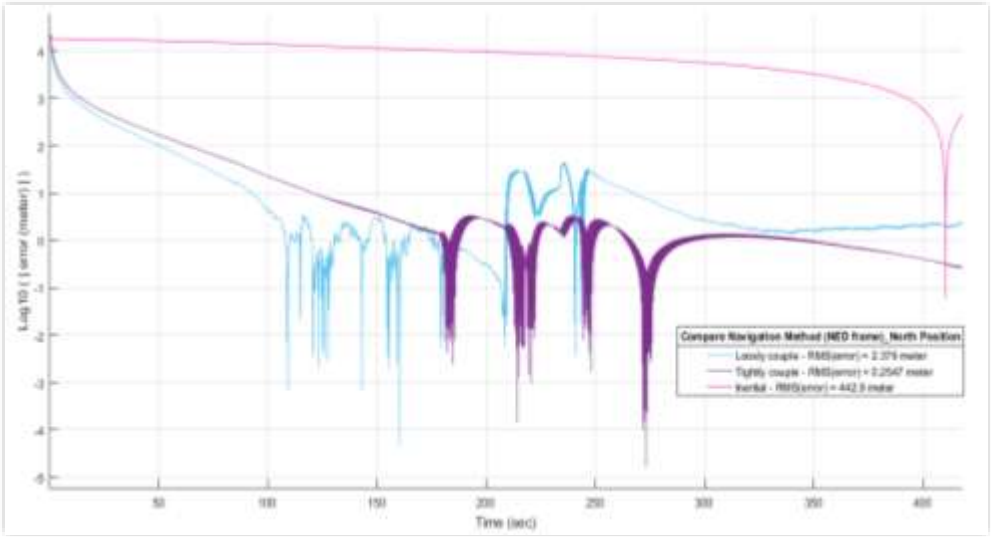



Figure 13. Compare the three navigation method - north position estimation

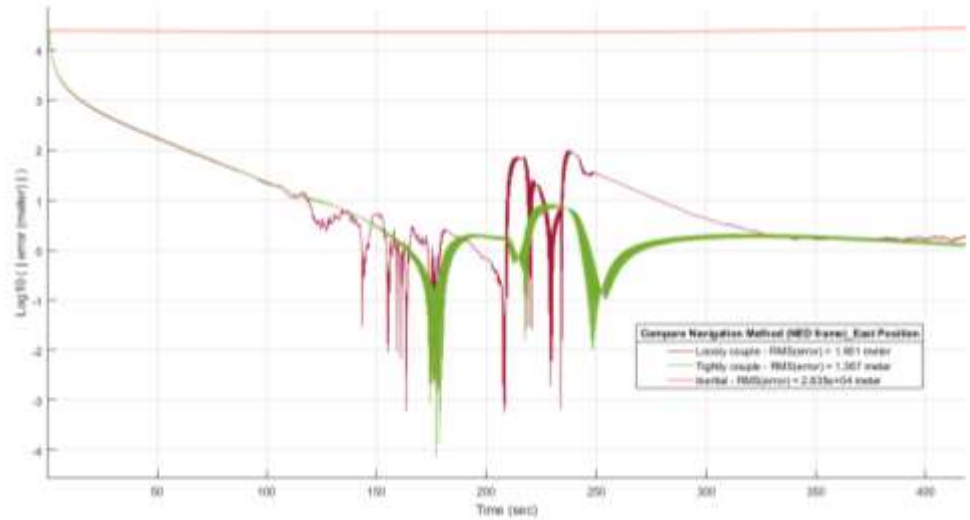



Figure 14. Compare the three navigation method - east position estimation

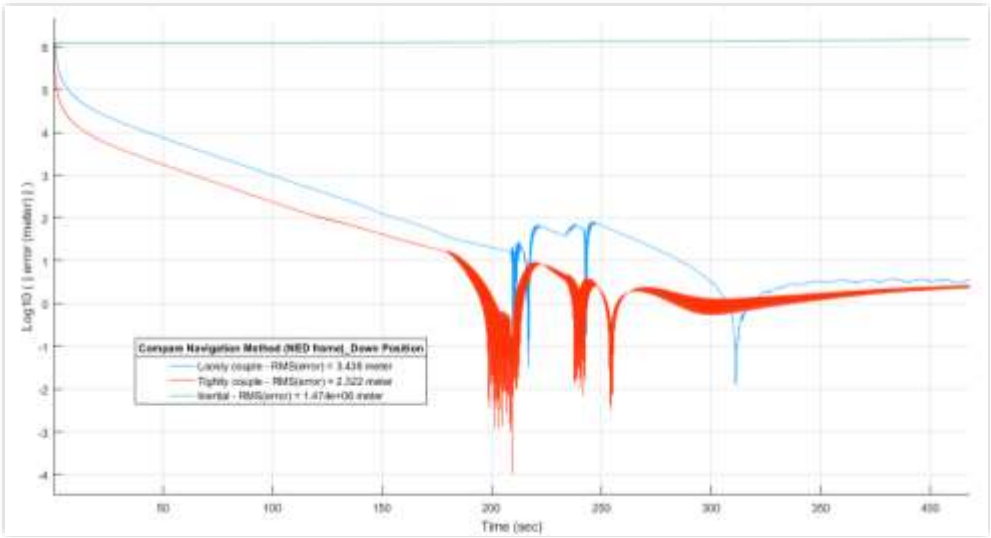



Figure 15. Compare the three navigation method - down position estimation

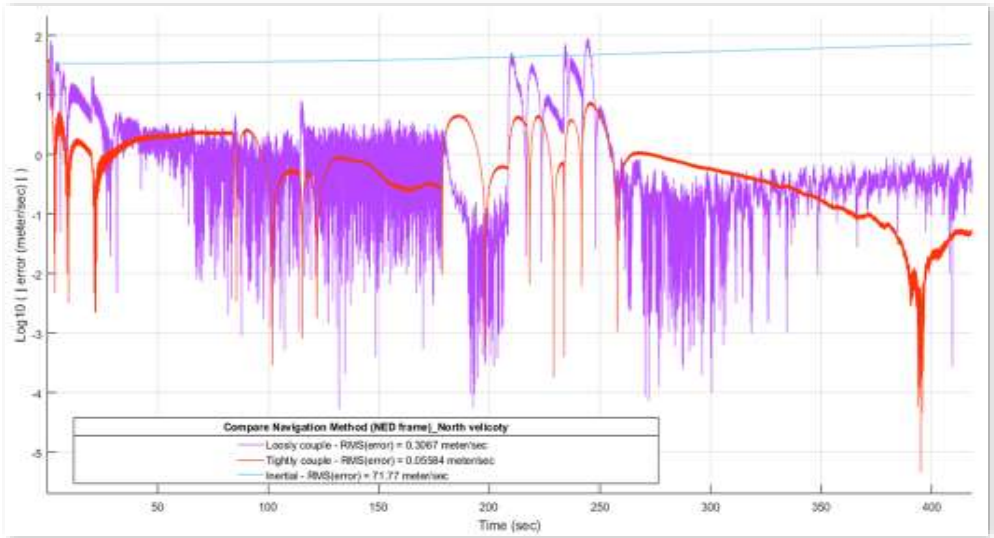



Figure 16. Compare the three navigation method - north velocity estimation

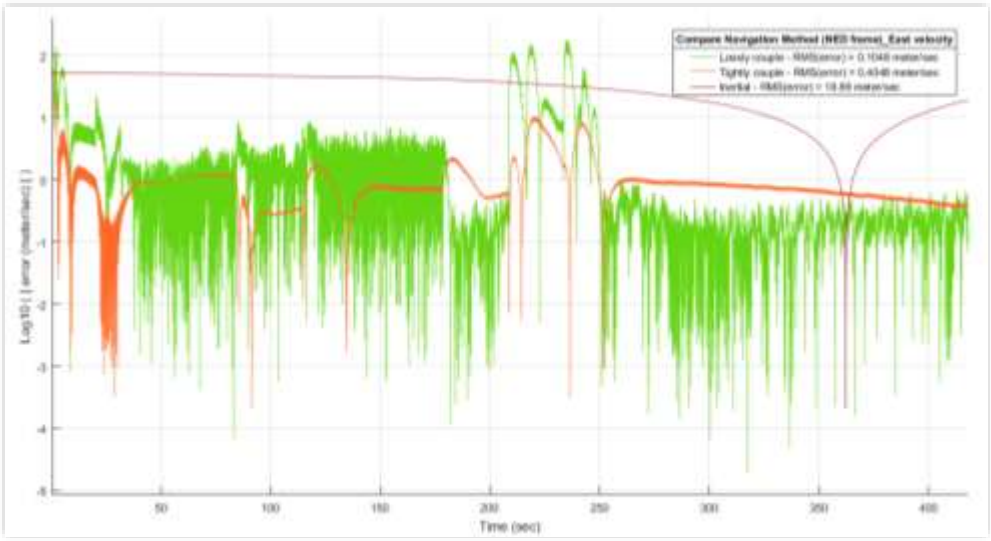



Figure 17. Compare the three navigation method - east velocity estimation

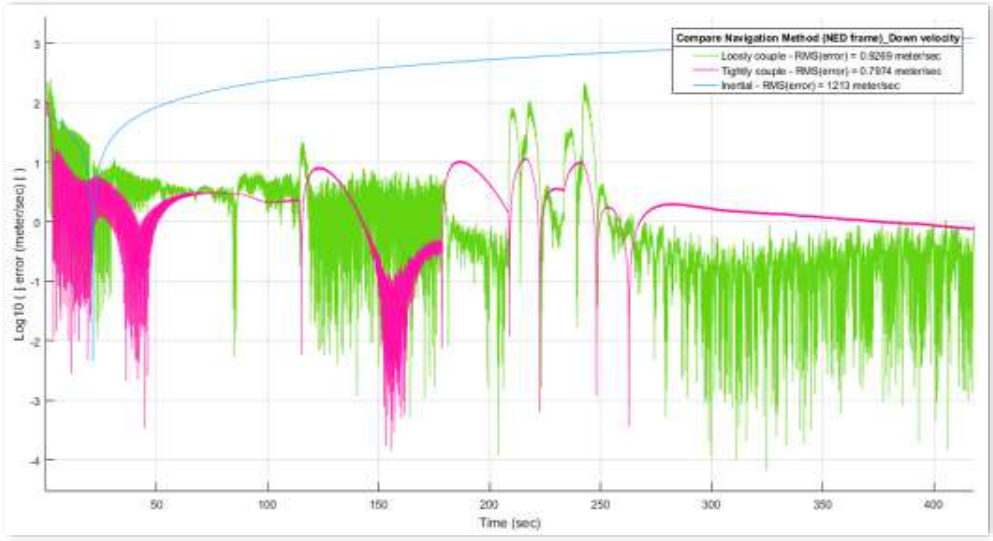



Figure 18. Compare the three navigation method - down velocity estimation

Error root-mean-square value (RMS) for NED frame position and velocity navigation solution, has been calculated and shown in table 3.

Table 3. The RMS error in position and velocity for the NED frame

| *Error RMS of* | INS | Loosely INS/GPS | Tightly INS/GPS |
|---|---|---|---|
| *North-position(meter)* | 442.9 | 2.379 | 0.2547 |
| *East-position(meter)* | 28,350 | 1.901 | 1.367 |
| *Down-position(meter)* | 1,474,000 | 3.438 | 2.322 |
| *North-velocity(meter/second)* | 71.77 | 0.3067 | 0.05584 |
| *East-velocity(meter/second)* | 18.89 | 0.1048 | 0.4048 |
| *Down-velocity(meter/second)* | 1213 | 0.9269 | 0.7974 |

## 7. Testing

An experiment has been done to test INS/GPS loosely coupled algorithm with different update rate, Figure 19 shows the experiment hardware and connection, ublox-6 GPS receiver has been used and configured to output ECEF position and velocity navigation solution in every 0.5 second.

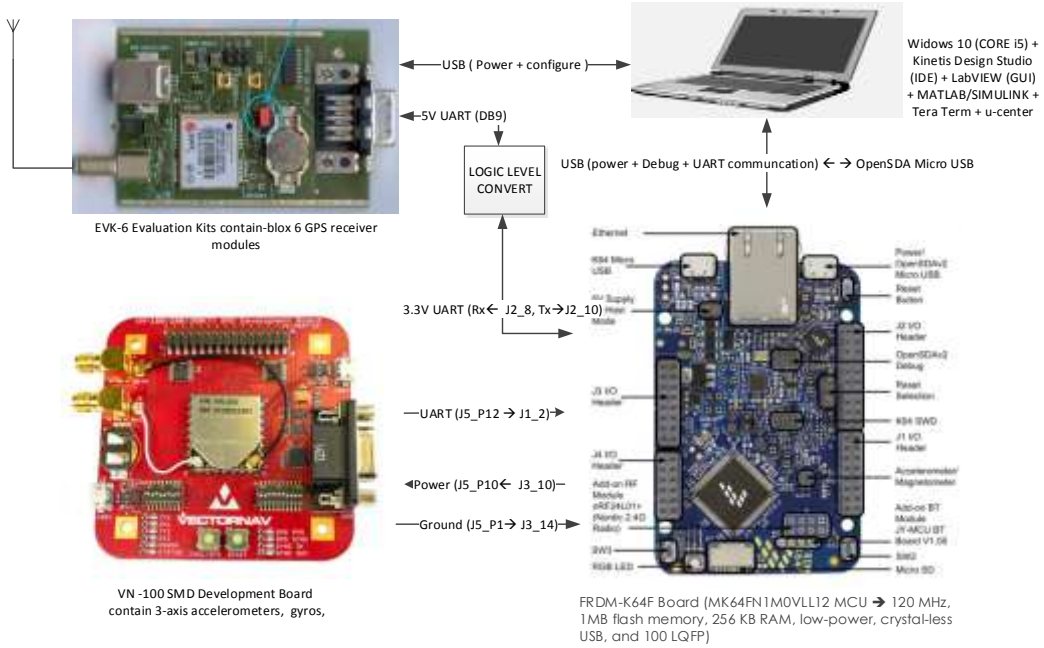



Figure 19. experiment hardware description diagram

For specific force and angular rate measurement VN-100 IMU is used and configured to output measurement every 0.02 seconds. FRDM-K64F (Cortex-M4 processor) board were programmed using C-ANSI code and "Kinetis Design Studio" Integrated Development Environment IDE to acquire data from the VN-100 IMU and the ublox-6 GPS receiver, then the FRDM-K64F is programmed to process the measurement and apply the loosely coupled integration algorithm, in which it applies the INS navigation equation with different update rate 5Hz, 25Hz, and 45Hz, and the Kalman filter apply correction at 2Hz update rate. Then the FRDM-K64F is programmed to output the navigation solution according to the INS navigation equation update rate, the test algorithm is shown in Figure20. A laptop was used to acquire the navigation solution, display, and log the data.

*Test Result*

The INS/GPS loosely coupled integration algorithm where tested three times each test with different INS equation update rate (5Hz, 25Hz, and 45Hz) each test last for 300 second, no movement were applied (stationary testing) so the ECEF velocity is zero, that enable to calculate the velocity error. Figure 21 shows the velocity error RMS value for each update rate, and Figure 22 shows the estimated velocity of the body in ECEF coordinate frame where Y-axes represent velocity (meter/second), x-axis represent time 300 second.

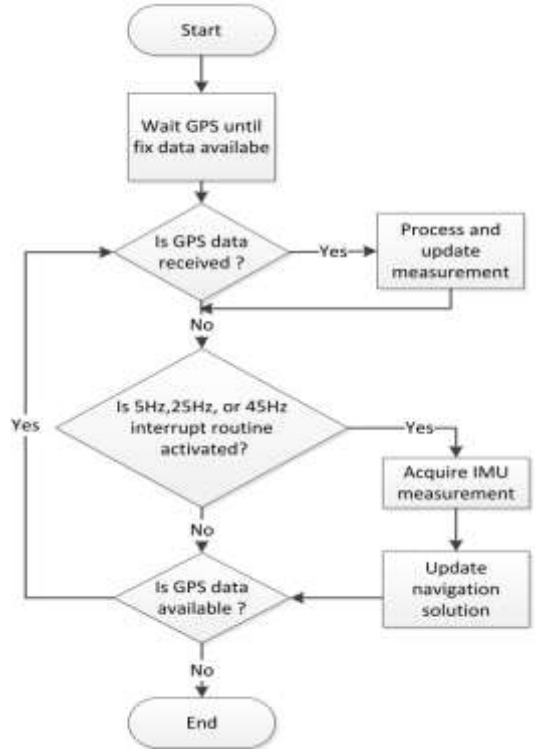



Figure 20. INS/GPS Loosely couple software algorithm

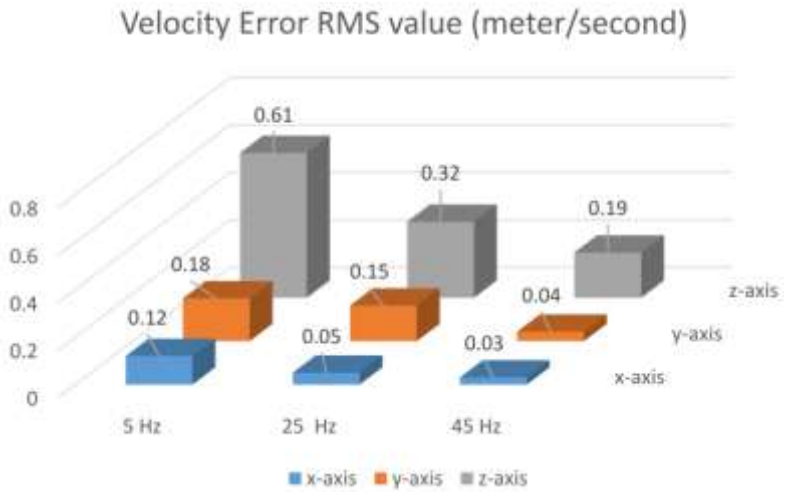
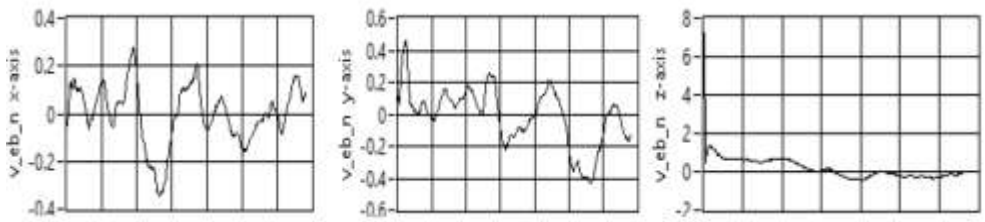
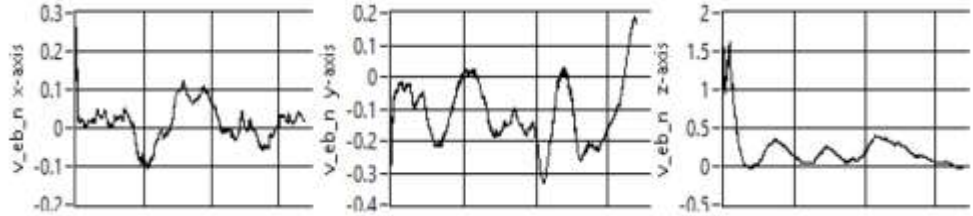


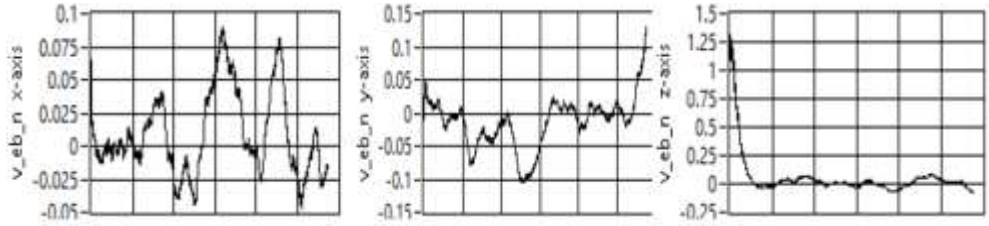

Figure 21. Error RMS value of estimated velocity




a) 5Hz INS equation update rate




b) 25Hz equation update rate




c) 45Hz equation update rate

Figure 22. Estimated velocity of the body in ECEF coordinate frame

## 8. Conclusion

From the simulation result it's shown that, Correction of IMU measurement used in updating the INS navigation equation using GPS and Kalman filter, considerably increases the navigation solution accuracy from range of ($10^4$-$10^6$ meter) to range of (1-10 meter) in position, and from range of ($10^{1.5}$ -$10^3$ meter/second) to range of ($10^{-1.5}$ -10 meter/second) in velocity. Moreover, for navigational position, Tightly-coupled integration gives better performance (RMS error range 0.25 to 2.3 meter) than loosely-couple method (RMS error range 1.9 to 3.4 meter). For navigational velocity, Tightly-coupled integration gives better performance (RMS error range 0.06 to 0.8 meter/second) than loosely-couple method (RMS error range 0.1 to 0.9 meter/second).

From the testing result it's shown that, higher navigation update rate increases the navigation solution accuracy, for loosely couple integration using 5Hz update rate it gives (0.12 to 0.61) meter/second RMS value error in velocity, while for 45Hz update rate it gives (0.03 to 0.19) meter/second RMS value error in velocity.

## 9. Recommendation

For the navigation solution this research analysis only the position and velocity, to complete the navigation solution its required also to discuss the attitude and the inclination of the object.
In this research only one loosely INS/GNSS integration method it tested, to validate the velocity stationary test were applied, method for validating position and attitude is required.

**10. References**

[1]. Paul D. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Artech house, Boston London, 2008.
[2]. S. Grewal, R. Weill, P. Andrews, Global Positioning Systems, Inertial Navigation, And Integration-second edition, John Wiley & Sons, Inc.,2007.
[3]. Wei M., K. P. Schwarz, Testing a decentralized filter for GPS/INS integration, *Proceedings of IEEE PLANS* 1990, 20-23 March, Las Vegas, Nevada, USA, pp. 429– 435, 1990.
[4]. Cannon, M. E., Airborne GPS/INS with an Application to Aerotriangulation. PhD Thesis, UCGE Report No. 20040, The Department of Geomatics Engineering, University of Calgary, Canada, 1991.
[5]. Salychev, O, Inertial Systems in Navigation and Geophysics. Bauman MSTU Press, 1998.
[6]. El-Sheimy N., K. P. Schwarz, Navigating urban areas by VISAT - a mobile mapping system integrating GPS/INS/digital cameras for GIS applications, *Journal of The Institute of Navigation*, Vol. 45, No. 4, pp. 275-286, 1991.
[7]. Jekeli, C., Inertial Navigation Systems with Geodetic Applications. New York, NY: Walter de Gruyter, 2000.
[8]. Scherzinger, B. M., Precise robust positioning with inertial/GPS RTK, *Proceedings of ION GPS* 2000, 19-22 September, Salt Lake City, UT, USA, pp. 155- 162, 2000.
[9]. Petovello, M., O. Mezentsev, G. Lachapelle and M.E. Cannon, High sensitivity GPS velocity updates for personal indoor navigation using inertial navigation systems, *Proceedings of ION GPS/GNSS* 2003, 9-12 September, Portland, OR, USA, pp. 2886-2896, 2003.
[10]. Gelb, A., Applied Optimal Estimation. Cambridge, Massachusetts: The Massachusetts Institute of Technology Press, 1974.

**Tareg Mahmoud** was born in Yanbu, KSA in 1986. He obtained his Bachelor Degree in Aeronautical Engineering (Avionics) in 2008, and his MSc in Electrical Engineering, School of Electrical Engineering & Informatics, ITB in 2016. Currently, he serves as Aeronautical Engineer, Avionics Licensed Aircraft Maintenance Engineer, Electrica.

.

**Bambang Riyanto Trilaksono** was born in Banyuwangi, Indonesia in 1962. He obtained his undergraduate degree from Electrical Engineering in Bandung Institute of Technology in 1986. He obtained his master and PhD degree from Electrical Engineering, Waseda University, Japan, in 1991 and 1994, respectively. Currently, he is a professor in Control and Computer System Research Group in Bandung Institute of Technology. His research interests include control, robotics and artificial intelligence.