Methodology for step-by-step Mechanization in the LLF Local Level Frame

Redding the binary data and initializing the algorithm

1- Reading the provided binary data using MATLAB.

2- Initializing the algorithm using the given initial position and setting the initial velocity equal to zero because the given data in the static mode.

IMU Error Compensation

Compensation of the deterministic errors before using the IMU measurements:

3- Compensation of the accelerometers' deterministic errors:

$$a_b = (I + S_a + N_a)^{-1}(\mathrm{f}_b - \mathrm{b}_\mathrm{a}) \tag{1}$$

4- Compensation of the gyros' deterministic errors:

$$\omega_{ib}^{b} = \left(I + S_\mathrm{g} + N_\mathrm{g}\right)^{-1}\left(\widetilde{\omega}_{ib}^{b} - \mathrm{b}_\mathrm{g}\right) \tag{2}$$

Where

S_a and S_g are the scale factors for the accelerometers & gyros

b_g gyro drifts (deg/hr)

b_a accelerometers biases (m/s^2)

N_a and N_g are the non-orthogonality matrix for the accelerometers & gyros

Calculate the Earth Parameters

5- The calculate the normal gravity value for the WGS84 parameters (the semi-major axis, the eccentricity of ellipsoid, Earth Rotation Rate, the normal gravity vector parameters a1 to a6 and the current position in Lat):

$$\mathrm{g} = \mathrm{a}_1 * (1 + \mathrm{a}_2 \sin^2 \emptyset + a_3 \sin^4 \emptyset) + (a_4 + a_5 \sin^2 \emptyset) * h + a_6 * h^2 \tag{3}$$

6- Calculate the radii of curvature - the prime vertical (N) and meridian (M) directions using equations (4 and 5):

$$M = \frac{a(1-e^2)}{(1-e^2\sin^2\emptyset)^{3/2}} \tag{4}$$

$$N = \frac{a}{(1-e^2\sin^2\emptyset)^{1/2}} \tag{5}$$

IMU Alignment

Initial alignment algorithm for inertial navigation is a mathematical procedure for determining the initial attitude information between the body frame of the IMU and the navigation frame.

There exist various algorithms for initial alignment, each is suitable under certain circumstances. These algorithms can be divided classified as:

- Accuracy of Attitude to be determined: Coarse / fine alignment.
- Dynamics of the vehicle: Static / in-motion alignment.

In the course project, the coarse alignment, which requires static inertial data, can be implemented in two steps as:

7-Accelerometer leveling:

The roll and pitch in ENU frame can be calculated as:

$$r = -sign(f_z)\sin^{-1}\left(\frac{\mathrm{f_x}}{\mathrm{g}}\right) \tag{6}$$

$$p = -sign(f_z)\sin^{-1}\left(\frac{\mathrm{f_x}}{\mathrm{g}}\right) \tag{7}$$

8- Gyro Compassing:

Makes use of the fact that a gyro with its sensitive axis in the horizontal plane (i.e. after accelerometer leveling) will sense a component of the Earth rotation at any point P on the surface of the Earth, This component will be maximum ($\omega_e \cos\emptyset$) when the sensitive axis points North and zero when it points East. The azimuth In ENU frame can be calculated as:

(8)

$$A = \tan^{-1}\left(\frac{\omega_x^b}{\omega_y^b}\right)$$

9- Consequently, the initial rotation matrix R_b^l can be obtained after determining the initial attitude angles during the alignment process), which can be written as:

$$R_b^l = \begin{vmatrix} \cos A \cos r + \sin A \sin p \sin r & \sin A \cos p & \cos A \sin r - \sin A \sin p \cos r \\ -\sin A \cos r + \cos A \sin p \sin r & \cos A \cos p & -\sin A \sin r - \cos A \sin p \cos r \\ -\cos p \sin r & \sin p & \cos p \cos r \end{vmatrix} \quad (9)$$

Note: This rotation matrix resulted from using the right-handed system where x is right (sideways), y is forward and z is pointing up.

10- Compute the quaternion parameters from the rotation matrix R_b^l , which can be written as:

$$\begin{vmatrix} q_{1(t_k)} \\ q_{2(t_k)} \\ q_{3(t_k)} \\ q_{4(t_k)} \end{vmatrix} = \begin{vmatrix} 0.25(R_{32} - R_{23})/q_{4(t_k)} \\ 0.25(R_{13} - R_{31})/q_{4(t_k)} \\ 0.25(R_{21} - R_{12})/q_{4(t_k)} \\ 0.5\sqrt{1 + R_{11} + R_{22} + R_{33}} \end{vmatrix} \quad (10)$$

11-Once the quaternion parameters are determined it should satisfy the following condition:

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \quad (11)$$

Due to computational errors, the above equality may not be valid. To compensate for this effect, special normalization procedures are usually implemented by calculating the errors and then normalize the quaternion parameters by this value. The computational error can be calculated as:

$$\Delta = 1 - q_1^2 + q_2^2 + q_3^2 + q_4^2 \quad (12)$$

The normalized quaternion parameters can be written as:

$$\hat{q} = \frac{q}{\sqrt{1 - \Delta}} \quad (13)$$

The rotation matrix R_b^l can be obtained as a function of the quaternion parameters using the following direct relationship:

$$R_b^l = \begin{vmatrix} q_1^2 - q_2^2 - q_3^2 - q_4^2 & 2(q_1 q_2 - q_3 q_4) & 2(q_1 q_3 + q_2 q_4) \\ 2(q1\, q3 + q2\, q4\,) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2 q_3 - q_1 q_4) \\ 2(q_1 q_3 - q_2 q_4) & 2(q2\, q3 + q1\, q4\,) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{vmatrix} \quad (14)$$

Angular Velocity Compensation

12- The gyro sensitive axis along the north and up directions observes part of the earth rotation (the stationary component). Also, because of the vehicle movement the local level frame change orientation and these angular changes monitored along the three axes of the gyro which is called the transportation rate. to estimate the actual angular changes of the moving body we should compensates for the Earth rotation and the local-level frame transportation rate. This correction terms will be used to compensate for the Coriolis Effect and can be calculated as:

$$\omega_{\mathrm{il}}^{\mathrm{b}} = R_l^b \begin{vmatrix} -\dfrac{V^n}{M+h} \\ \dfrac{V^e}{N+h} + \omega_e \cos \emptyset \\ \dfrac{V^e \tan \emptyset}{N+h} + \omega_e \sin \emptyset \end{vmatrix} \tag{15}$$

Consequently, the angular changes corresponding to $\omega_{\mathrm{il}}^{\mathrm{b}}$ can be determined as:

$$\theta_{\mathrm{il}}^{\mathrm{b}} = \omega_{\mathrm{il}}^{\mathrm{b}} * \Delta t \tag{16}$$

Then the measured angular increments θ_{ib}^{b} is compensated for $\theta_{\mathrm{il}}^{\mathrm{b}}$ in order to get the actual angular changes of the vehicle $\theta_{\mathrm{l}b}^{\mathrm{b}}$ as follows:

$$\theta_{\mathrm{l}b}^{\mathrm{b}} = \theta_{\mathrm{i}b}^{\mathrm{b}} - \theta_{\mathrm{il}}^{\mathrm{b}} = |\Delta\theta_x \quad \Delta\theta_y \quad \Delta\theta_z|^T \tag{17}$$

Attitude integration Loop

In this loop the attitude updated throw three main steps
13- Update the quaternion parameters, which can be written as follow:

$$\begin{vmatrix} q_{1(t_{k+1})} \\ q_{2(t_{k+1})} \\ q_{3(t_{k+1})} \\ q_{4(t_{k+1})} \end{vmatrix} = \begin{vmatrix} q_{1(t_k)} \\ q_{2(t_k)} \\ q_{3(t_k)} \\ q_{4(t_k)} \end{vmatrix} + \frac{1}{2} \begin{vmatrix} 0 & \Delta\theta z & -\Delta\theta_y & \Delta\theta x \\ -\Delta\theta z & 0 & \Delta\theta x & \Delta\theta_y \\ \Delta\theta_y & -\Delta\theta x & 0 & \Delta\theta z \\ -\Delta\theta x & -\Delta\theta_y & -\Delta\theta z & 0 \end{vmatrix} \begin{vmatrix} q_{1(t_k)} \\ q_{2(t_k)} \\ q_{3(t_k)} \\ q_{4(t_k)} \end{vmatrix} \tag{18}$$

Where $\Delta\theta_x, \Delta\theta_y$ and $\Delta\theta z$ can be obtained from equation (17) and the quaternion parameters at time t_k represent the values at the previous time.

14- Check if the Updated quaternion parameters satisfy equation (11) or not.in case of not satisfying this assumption the compensation for these errors is necessary and can be determined using equation (13).

15- Compute the rotation matrix from the updated quaternion parameters by using equation (14)

16- Estimate Euler angels (roll, Pitch and Azimuth) from equations (6, 7 and 8) respectively.

Velocity and position integration Loop

17-Determine the body velocity increment as:

$$\Delta V^l = R_b^l * \Delta V^b - \left(2\Omega_{ie}^l + 2\Omega_{el}^l\right) V^l * \Delta t + \mathrm{g}^l * \Delta t \tag{19}$$

Where

- $R_b^l * \Delta V^b$: is the measured velocity increments after transformation into the local-level frame.
- $\left(2\Omega_{ie}^l + 2\Omega_{el}^l\right) V^l * \Delta t$: is the Coriolis correction that compensates for the Earth rotation and the Local level frame change of orientation.
- $\mathrm{g}^l * \Delta \mathrm{t}$: is the gravity correction.
- Ω_{ie}^l and Ω_{el}^l represent the skew symmetric matrix corresponding to the angular velocity vector ω_{ie}^l and ω_{el}^l respectively.
- g^l represent the normal gravity vector in the local level frame = $\begin{vmatrix} 0 \\ 0 \\ -\mathrm{g} \end{vmatrix}$
- $\Delta t = \frac{1}{data\ rate}$

18-Compute the velocity components:

$$V^l(t_{k+1}) = V^l(t_k) + \frac{1}{2}\left(\Delta V^l(t_k) + \Delta V^l(t_{k+1})\right) \tag{20}$$

Note: at the first iteration $\Delta V^l(t_k) = 0$

19- Compute the altitude component:

$$h(t_{k+1}) = h(t_k) + \frac{1}{2}\left(V^u(t_{k+1}) + V^u(t_k)\right) \tag{21}$$

20- Compute the latitude:

$$\varnothing(t_{k+1}) = \varnothing(t_k) + \frac{1}{2}\left(\frac{V^n(t_k) + V^n(t_{k+1})}{M + h}\right)\Delta t \tag{22}$$

21- Compute the Longitude:

$$\lambda(t_{k+1}) = \lambda(t_k) + \frac{1}{2}\left(\frac{V^e(t_k) + V^e(t_{k+1})}{(N + h)\cos\varnothing}\right)\Delta t \tag{23}$$

Repeat – Steps 1, 3-6 and 12-21