Navigation and Guidance Systems

Course 086759- Spring 2018

Assignment #9

Daniel Engelsman 300546173

1 Position Error

A body accelerates south at 4 $[\frac{m}{s^2}]$ with initial azimuth error 5 [mrad].

$$\frac{d\Delta V^L}{dt} = A\Delta\psi + C_B^{L_c}\delta f_B \tag{1.1}$$

$$f_{L_c}^C = C_B^L f^B = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} -4 \\ 0 \\ g \end{bmatrix} \tag{1.2}$$

$$A \doteq [f_{L_c}^C]_X = \begin{bmatrix} 4 \\ 0 \\ g \end{bmatrix}_X = \begin{bmatrix} 0 & -g & 0 \\ g & 0 & 4 \\ 0 & -4 & 0 \end{bmatrix} \qquad and \qquad C_B^{L_c} = I_{3X3} \tag{1.3}$$

$$\frac{d\Delta V^L}{dt} = \begin{bmatrix} 0 & -g & 0 \\ g & 0 & 4 \\ 0 & -4 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0.005 \end{bmatrix} + C_B^{L_c}\cancelto{0}{\delta f_B} = \begin{bmatrix} 0 \\ 0.02 \\ 0 \end{bmatrix} [\frac{m}{sec^2}] \tag{1.4}$$

$$\frac{d\Delta P^L}{dt} = \begin{bmatrix} 0 \\ 0.02 \\ 0 \end{bmatrix} t \ [\frac{m}{sec}] \qquad and \qquad \Delta P^L = \begin{bmatrix} 0 \\ 0.01 \\ 0 \end{bmatrix} t^2 \ [m] \tag{1.5}$$

$$\Delta P^L(t = 8[min]) = \begin{bmatrix} 0 \\ 0.01 \\ 0 \end{bmatrix} (8 \cdot 60)^2 = \begin{bmatrix} 0 \\ 2304 \\ 0 \end{bmatrix} [m] \tag{1.6}$$

$$x(t = 8[min]) = x_0 + v_0 t + \frac{1}{2}at^2 = 0 + 0 + \frac{1}{2} \cdot -4(8 \cdot 60)^2 = -460.8[km] \tag{1.7}$$

To sum up the final true position : $P^L = \begin{bmatrix} -460.8 \\ 2.304 \\ 0 \end{bmatrix} [km]$

The difference is rather big although started as minor. But when no correction / estimation is used along motion and navigation is "blind", these error evolve with time quadratically.

2 Position and Velocity Errors

$$V^L = \begin{bmatrix} 0 \\ 200 \\ 0 \end{bmatrix} [\frac{m}{sec}] \quad ; \quad \Delta\psi = \begin{bmatrix} 0.001 \\ 0 \\ 0 \end{bmatrix} \quad ; \quad \delta f_B = \underline{0} \tag{2.1}$$

$$f^C_{L_c} = C^L_B f^B = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{2.2}$$

Compute its velocity and position errors :

$$\frac{d\Delta V^L}{dt} = A\Delta\psi + C^{L_c}_B \cancelto{0}{\delta f_B} = \begin{bmatrix} 0 & -g & 0 \\ g & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.001 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.0098 \\ 0 \end{bmatrix} [\frac{m}{sec^2}] \tag{2.3}$$

For short periods :

$$\Delta V^L = \begin{bmatrix} 0 \\ 0.0098 \\ 0 \end{bmatrix} t \ [\frac{m}{sec}] \qquad \Delta P^L = \begin{bmatrix} 0 \\ 0.0049 \\ 0 \end{bmatrix} t^2 \ [m] \tag{2.4}$$

And for longer period, given a constant velocity :

$$\frac{d\Delta P^L}{dt} = \begin{bmatrix} -\frac{V_E}{R} \\ \frac{V_N}{R} \\ \frac{V_E}{R} tan\lambda \end{bmatrix} \Delta P_L + \Delta V^L = \frac{d\Delta P^L}{dt} = -3.1347 \cdot 10^{-5} \begin{bmatrix} 0 \\ 0 \\ tan\lambda \end{bmatrix}_X \Delta P_L + \Delta V^L \tag{2.5}$$

3 Gyro Drift

(i) If the gyro drift is in the x direction, in which direction is the position error?

Recall that gyro error is expressed as : $\frac{d\Delta\psi}{dt} = -C_B^{L_c}\delta\omega_B$. Hence, a cross product

projecting such that : $\begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ V_x & V_y & V_z \\ d_x & d_y & d_z \end{vmatrix}$, we got $\begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ V_x & V_y & V_z \\ d_x & 0 & 0 \end{vmatrix} \rightarrow \Delta\psi \sim d_x \begin{bmatrix} 0 \\ V_z \\ -V_y \end{bmatrix}$.

One can tell that error in the X-axis is projected onto the 2 other axes.

(ii) Derive a simpler approximation valid for short times :

$$\frac{d\Delta V^L}{dt} = A\Delta\psi + C_B^{L_c}\cancelto{0}{\delta f_B} = \begin{bmatrix} 0 & -d_z & d_y \\ d_z & 0 & -d_x \\ -d_y & d_x & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} d_y \\ -d_x \\ 0 \end{bmatrix} g \cdot t \ [\frac{m}{sec^2}] \tag{3.1}$$

$$\frac{d\Delta P^L}{dt} = \begin{bmatrix} d_y \\ -d_x \\ 0 \end{bmatrix} \frac{g \cdot t^2}{2} \ [\frac{m}{sec}] \tag{3.2}$$

$$\Delta P^L = \begin{bmatrix} d_y \\ -d_x \\ 0 \end{bmatrix} \frac{g \cdot t^3}{6} \ [m] \tag{3.3}$$

How long can your approximation be used?

Short time period criterion is $\omega_s t \ll 1$, here :

$$\frac{d\Delta V^L}{dt} = \frac{g}{R} \cdot \Delta P^L \rightarrow \frac{g}{R} \cdot t^2 \sim 1 \rightarrow t \sim \sqrt{\frac{R}{g}} = 800 \text{ [sec]} = 13.333 \text{ [min]}$$

4 Rocket Firing

Compute the position errors at the end of the 30 seconds. **Case 1 :** $\theta_0 = 90\ [^\circ]$:

$$f^C_{L_c} = C^L_B f^B = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 10g \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -9 \cdot g \end{bmatrix} [\frac{m}{sec^2}] \tag{4.1}$$

$$\frac{d\Delta V^L}{dt} = A\Delta\psi + C^{L_c}_B \cancelto{0}{\delta f_B} = \begin{bmatrix} 0 & 9g & 0 \\ -9g & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0.010 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} [\frac{m}{sec^2}] \tag{4.2}$$

Hence, no error will evolve since all cross product results in "0". **Case 2 :** $\theta_0 = 60\ [^\circ]$:

$$f^C_{L_c} = C^L_B f^B = \begin{bmatrix} 0.5 & 0 & 0.866 \\ 0 & 1 & 0 \\ -0.866 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 10g \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} 49.05 \\ 0 \\ -75.145 \end{bmatrix} [\frac{m}{sec^2}] \tag{4.3}$$

$$\frac{d\Delta V^L}{dt} = A\Delta\psi + C^{L_c}_B \cancelto{0}{\delta f_B} = \begin{bmatrix} 0 & 75.145 & 0 \\ -75.145 & 0 & -49.05 \\ 0 & 49.05 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0.010 \end{bmatrix} = \begin{bmatrix} 0 \\ -0.49 \\ 0 \end{bmatrix} [\frac{m}{sec^2}] \tag{4.4}$$

$\Delta P(t) = \begin{bmatrix} 0 \\ -0.245 \\ 0 \end{bmatrix} t^2 \rightarrow \Delta P_E(t = 30[sec]) - 0.245 \cdot 30^2 = -220.5[m]$. **Case 3 :** $\theta_0 = 45\ [^\circ]$:

$$f^C_{L_c} = C^L_B f^B = \begin{bmatrix} 0.707 & 0 & 0.707 \\ 0 & 1 & 0 \\ -0.707 & 0 & 0.707 \end{bmatrix} \begin{bmatrix} 10g \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} 69.36 \\ 0 \\ -59.55 \end{bmatrix} [\frac{m}{sec^2}] \tag{4.5}$$

$$\frac{d\Delta V^L}{dt} = A\Delta\psi + C^{L_c}_B \cancelto{0}{\delta f_B} = \begin{bmatrix} 0 & 59.55 & 0 \\ -59.55 & 0 & -69.36 \\ 0 & 69.36 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0.010 \end{bmatrix} = \begin{bmatrix} 0 \\ -0.6936 \\ 0 \end{bmatrix} [\frac{m}{sec^2}] \tag{4.6}$$

After Integration : $\Delta P(t) = \begin{bmatrix} 0 \\ -0.347 \\ 0 \end{bmatrix} t^2 \rightarrow \Delta P_E(t = 30[sec]) - 0.347 \cdot 30^2 = -312.1[m]$

5 Ground Vehicle with IMU

Let us first present ideal path of the vehicle emanated from simple kinematics :

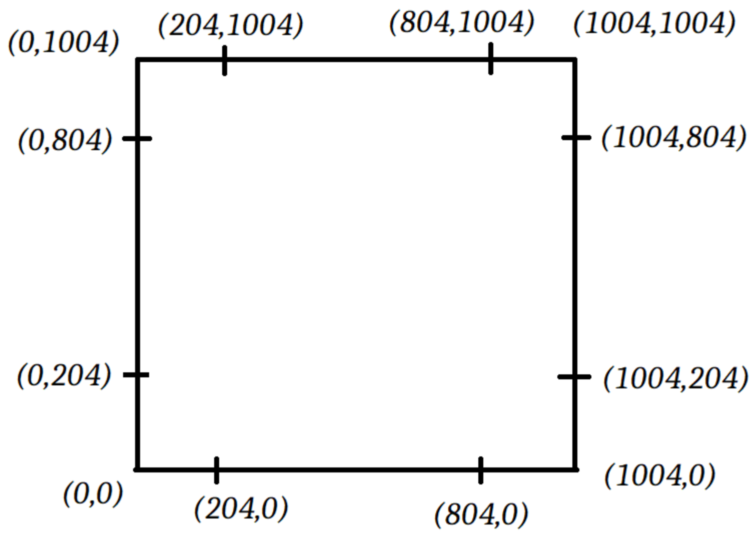

Figure 1: Ideal path of the vehicle

Along the route, the vehicle experiences as follows :

	Seg. [#]	C_B2L(psi)	Acc_ X [m/s2]	V_Xo [m/s]	Time [sec]
(i)	1	0	0.1	0	20.387
(ii)			0	20	30
(iii)			-0.1	0	20.387
(i)	2	90	0.1	0	20.387
(ii)			0	20	30
(iii)			-0.1	0	20.387
(i)	3	180	0.1	0	20.387
(ii)			0	20	30
(iii)			-0.1	0	20.387
(i)	4	270	0.1	0	20.387
(ii)			0	20	30
(iii)			-0.1	0	20.387

Figure 2: Vehicle Segments of motion

Let us divide the path into 4 segments, where the vehicle turns its azimuth $(\phi = \theta = 0)$:

(i) $C_B^L(\psi = 0[°]) = I_{3X3}$ $\qquad$ (ii) $\quad C_B^L(\psi = 90[°]) = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

(iii) $C_B^L(\psi = 180[°]) = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $\qquad$ (iv) $\quad C_B^L(\psi = 270[°]) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Note : For the sake of **clarity**, I will present only 1 segment, since they repeat along path :

(a). Compute the error resulting from a gyro-Z 1,000 [Hz] error.
As shown in lectures, we'll assume bounded typical gyro white noise $w_g = \pm 5[\frac{\circ}{hr}]$

$$\frac{d\Delta\psi}{dt} = -C_B^{L_c}\delta\omega_B = I_{3X3}\begin{bmatrix} 0 \\ 0 \\ -\delta\omega_Z \end{bmatrix} \quad \rightarrow \quad \Delta\psi = (-\delta\omega_Z \cdot t)\hat{k} \tag{5.1}$$

$$\frac{d\Delta V^L}{dt} = [f_{L_c}^C]_X \begin{bmatrix} 0 \\ 0 \\ \psi_i \end{bmatrix} + C_B^{L_c}\cancelto{0}{\delta f_B}[\frac{m}{sec^2}] \quad = \quad \begin{bmatrix} 0 & -g & 0 \\ g & 0 & \pm 0.1g \\ 0 & \mp 0.1g & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ \psi_i \end{bmatrix}[\frac{m}{sec^2}] \tag{5.2}$$

Let us present the errors measured and those evolve respectively :

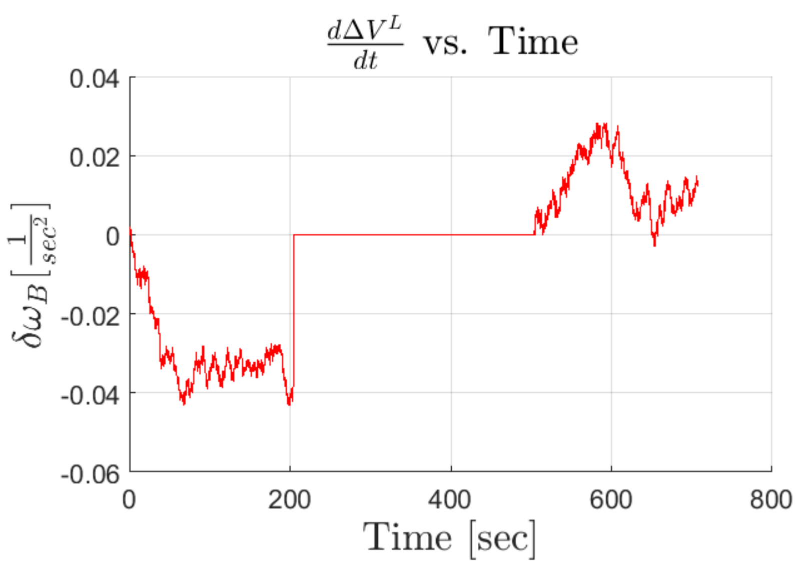

Figure 3: Gyro's Bounded White Noise

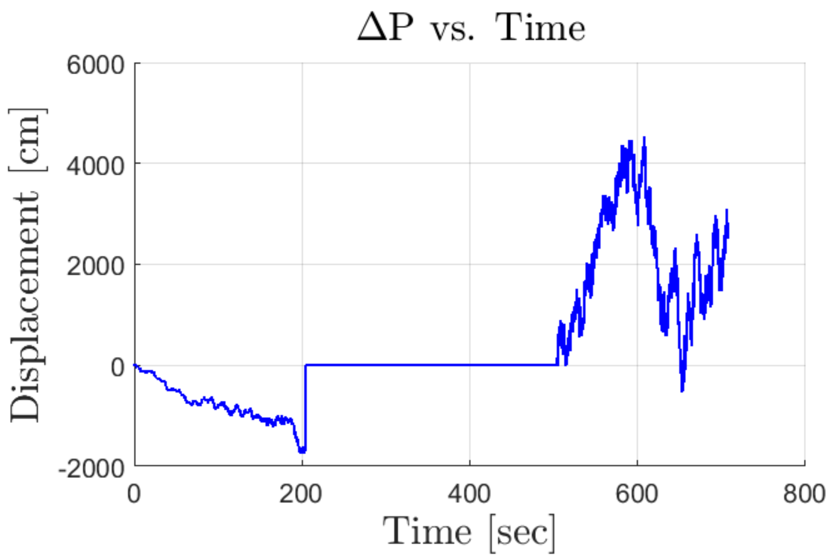

Figure 4: Gyro's Evolving position error

Note : In gyro error, when the vehicle does not accelerate, error is not projected. Let us now present the general path that the vehicle would go through :

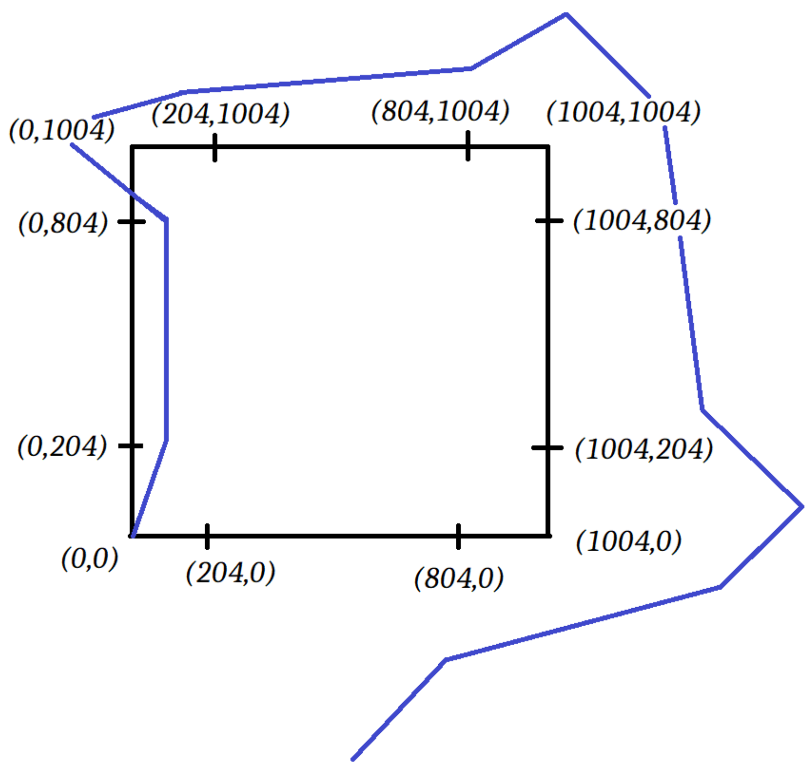

Figure 5: Estimated Path of vehicle under Gyro error

(b). Compute the error resulting from an accelerometer-X 1,000 [Hz] error.
As shown in lectures, we'll assume typical a_{SF}^{B} bounded white noise $w_a = \pm mg[\frac{m}{sec^2}]$

The vehicle travels each segment subjected to the following :

$$f_{L_c}^{C} = C_B^L(\psi) f^B = C_B^L(\psi) \begin{bmatrix} \pm 0.1g \\ 0 \\ g \end{bmatrix} [\frac{m}{sec^2}] \tag{5.3}$$

$$\frac{d\Delta V^L}{dt} = [f_{L_c}^{C}]_X \cancelto{0}{\begin{bmatrix} 0 \\ 0 \\ \psi_{0,i} \end{bmatrix}} + C_B^{L_c} \begin{bmatrix} \delta f_X \\ 0 \\ 0 \end{bmatrix} [\frac{m}{sec^2}] \quad \rightarrow \quad \Delta P(t) = \iint \frac{d\Delta V^L}{dt} dt \;\; [m] \tag{5.4}$$

Let us present the errors measured and those evolve respectively :

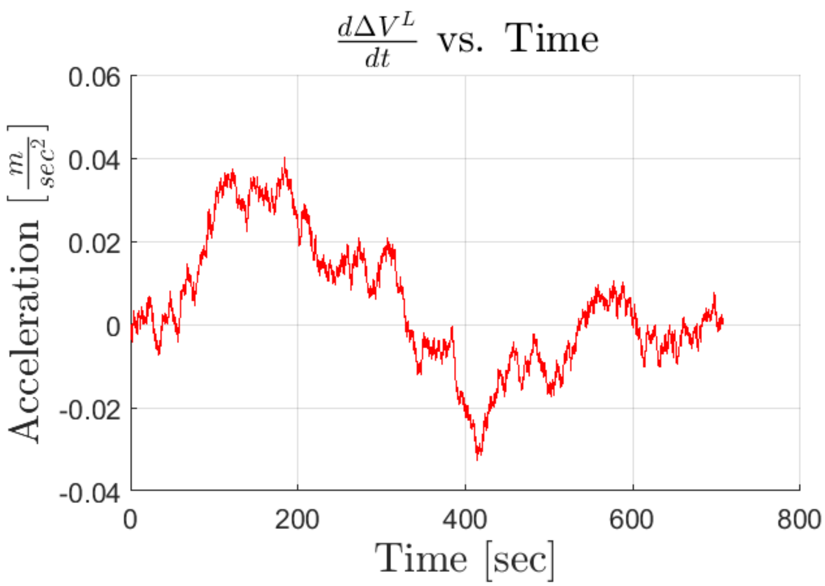

Figure 6: Accelerometer Bounded White Noise

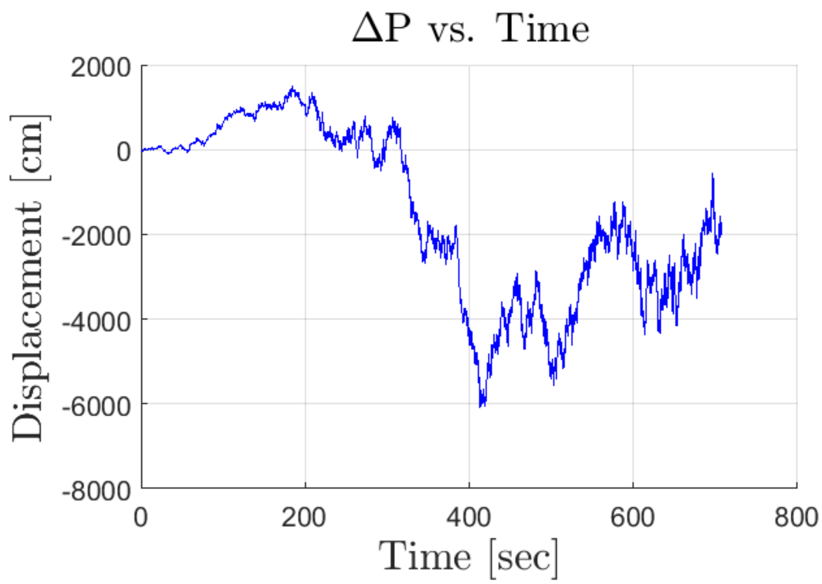

Figure 7: Accelerometer's Evolving position error

Let us now present the general path that the vehicle would go through :

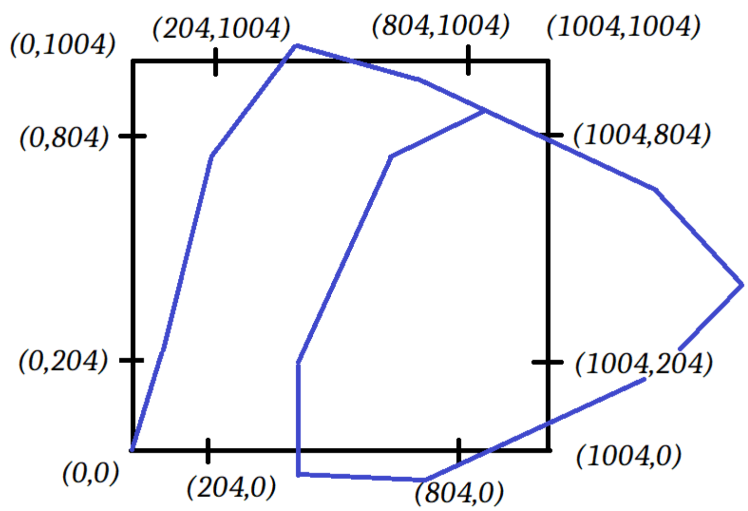

Figure 8: Estimated Path of vehicle under Accelerometer error

5.1 Discussion

1. The gyro error in the z axis under x-axis acceleration is translated into azimuth error which is projected onto the East direction. The obtained drift it clearly seen.

2. When vehicle is not accelerating, no projection occurs and thus why we see that measurements nullified.

3. The accelerometer's error is being projected into the N direction, and getting worse when the azimuth angle changes, since it's being re-projected.

4. Overall, under same magnitude of error, the gyro's error diverges faster and exhibits bigger. However, here since the body stops accelerating, the projection stops and the total position error is smaller.

5. When no estimation or external sources of information are in fusion with the model, the error evolves and diverges. Ideally, the error needs to be truncated such that every once in a while the error is cut down into "real results" (AKA ground truth) :

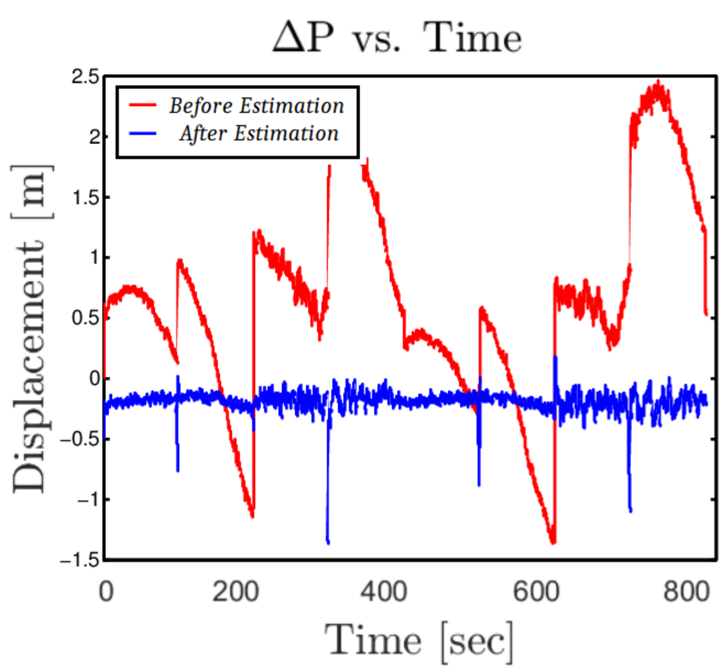

Figure 9: Error truncation when Estimation model is used

6 Appendix - Code

```
%% ————————— Ground Vehicle with IMU ————————— %
clc; clear all; close all; set(0,'defaultfigurecolor',[1 1 1]);

g = 9.81;                              % Gravity Acc. [m/sec2]
dt = 1/10;                             % Step size per [Hz]
t_segm = [20.387 30 20.387];           % Duration of sub segment
   [s]
sub_segm = round( t_segm/dt );         % Total steps of sub
   segment

% ————————— Define Rotation Matrix for each segment ————————— %
for i = 1:4
    C_L2B(:,:,i) = angle2dcm(0, 0, (pi/2)*(i-1));
end

f_B = g*[0.1 0 1; 0 0 1; -0.1 0 1];      % Accelerometer sub-
   segments
d_Psi = zeros(3,1); df_B = zeros(3,1);   % Initialization
V_1_3 = zeros(3,1); P_1_3 = V_1_3;       % Initialization
Tot_t = 0;                               % Initialization
run = 2;                                 % 1 == gyro execurion;
   else == Acc;

% ——————————————— Outer Loop for each segment ——————————————— %
for j = 1:1
    % ———————————— Inner Loop for each sub-segment ———————————— %
    for i = 1:3
        A = skew_symmetric( C_L2B(:,:,j)*f_B(i,:)' );
        % ————————————————— White Noise Loop ————————————————— %
        k = 3*(j-1)+i;                                % Rolling
           Index
        d_del_v(k).t = zeros(3,1);
        tt(k).t(1,:) = dt:dt:sub_segm(i);
```

```
        for t = 1:length(tt(k).t)-1
            if ( run == 1 )
                Wg_z = 0.001*(-1+2*rand);                % gyro White
                   Noise
                d_Psi = [0 0 Wg_z]'*dt;
            else % ( run == acc. )
                Wa_x = 0.001*(-1+2*rand);                % Acc. White
                   Noise
                df_B = [Wa_x 0 0]';
            end
            % Velocity Error vector
            d_del_v(k).t(:,t+1) = d_del_v(k).t(:,t) + A*d_Psi +
               df_B;
        end
        %          D_P(j,:) = D_V(j,:)*t
        Tot_t = horzcat(Tot_t, Tot_t(end) + tt(k).t );
        V_last = V_1_3(1,end);
        V_1_3 = horzcat(V_1_3, V_last + d_del_v(k).t(:,2:end) );
        % ---------------- Double Integration of D_V ---------------- %
        D_P(k).t = d_del_v(k).t.*( (0.5*(Tot_t(end) + tt(k).t))
           .^2);
        P_last = P_1_3(1,end);
        P_1_3 = horzcat(P_1_3, P_last + D_P(k).t(:,2:end) );
    end
end

%%
figure(1); hold on; grid on; %close all;
plot(Tot_t(1:length(V_1_3)), V_1_3(1,:), 'r-', 'LineWidth', 1);
ind(1) = title('$\frac{d \Delta V^L}{dt}$ vs. Time') ;
ind(2) = xlabel('Time [sec]');
ind(3) = ylabel('Acceleration $[\frac{m}{sec^2}]$');
% ind(3) = ylabel('$\delta \omega_B [\frac{1}{sec^2}]$');
% ind(4) = legend('$V_x$', '$V_y$');
set(ind, 'Interpreter', 'latex', 'fontsize', 25 );
```