ENGO 620 Estimation for Navigation

Fall 2013

Lab2 - Analysis

Instructor : Professor M.G. Petovello
Student : Ranjeeth Kumar Siddakatte
UCID : 10119739

DEPARTMENT OF GEOMATICS ENGINEERING
UNIVERSITY OF CALGARY, ALBERTA, CANADA

1 Task 1 – Kalman Filter System model

1.1 Continuous time system model

1.1.1 Random walk position

States to be estimated are the 3D position in local level frame (3 states), clock bias, and clock drift. Clock drift is included in the system model because the clock bias state is related to clock drift.

In the local level frame, for a given epoch, any error in position is reflected as change in the component of relative position vector between satellite and receiver along North (N), East (E), and Up (U) axes. Hence, states to be estimated will be the delta change in the position along N, E, U axes ($\delta x_n, \delta x_e, \delta x_u$), change in clock bias (δcdt) and clock drift ($\delta c\dot{dt}$). The error states estimated by Kalman filter will be translated to corresponding errors in latitude, longitude, and height, and then full state is corrected before carrying states to the next epoch.

State vector is given by, $X(t) = [\delta x_n \quad \delta x_e \quad \delta x_u \quad \delta cdt \quad \delta c\dot{dt}]^T$

Stochastic model chosen for 3D position states is a random walk model. For clock bias and clock drift, stochastic model is chosen as given in the lab assignment. Since the stochastic model for all states is a random walk, shaping matrix G will be an identity matrix.

Stochastic model vector, $w(t) = [w_{xn} \quad w_{xe} \quad w_{xu} \quad w_b \quad w_d]^T$

Corresponding continuous-time system model is written using state space notation as,

$$\dot{X}(t) = F(t) \cdot X(t) + G(t) \cdot w(t)$$

$$\begin{bmatrix} \delta\dot{x}_n \\ \delta\dot{x}_e \\ \delta\dot{x}_u \\ \delta\dot{cdt} \\ \delta\ddot{cdt} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta x_n \\ \delta x_e \\ \delta x_u \\ \delta cdt \\ \delta\dot{cdt} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} w_{x_n} \\ w_{x_e} \\ w_{x_u} \\ w_b \\ w_d \end{bmatrix}$$

States are assumed to have spectral density values $q_n\ q_e\ q_u\ q_b\ q_d$ respectively for north-east-up position states, clock bias, and clock drift states. The values for q_b and q_d will be chosen as given in the assignment. Spectral density values for 3D position states ($q_n\ q_e\ q_u$) will be set by tuning.

1.1.2 Random walk velocity

In this case the state vector includes delta velocities along N-E-U axes also. Since position and velocity are linearly related, stochastic modeling of velocity states will also contribute to uncertainty in position states. Hence, the stochastic model is a random walk for each velocity state, and random constant for position states. G matrix is again, an identity matrix.

$$\dot{X}(t) = F(t) \cdot X(t) + G(t) \cdot w(t)$$

$$\begin{bmatrix} \delta\dot{x}_n \\ \delta\dot{x}_e \\ \delta\dot{x}_u \\ \delta\dot{c}dt \\ \delta\ddot{x}_n \\ \delta\ddot{x}_e \\ \delta\ddot{x}_u \\ \delta\ddot{c}dt \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta x_n \\ \delta x_e \\ \delta x_u \\ \delta cdt \\ \delta\dot{x}_n \\ \delta\dot{x}_e \\ \delta\dot{x}_u \\ \delta\dot{c}dt \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ w_b \\ w_{v_n} \\ w_{v_e} \\ w_{v_u} \\ w_d \end{bmatrix} \quad (1.2)$$

States are assumed to have spectral density values qv_n , qv_e, qv_u, q_b, q_d respectively for north-east-up velocity states, clock bias, and clock drift states.

1.2 Transition matrix and process noise matrix

1.2.1 Random walk position

1.2.1.1 Transition matrix ($\phi_{k,k+1}$)

$\phi_{k,k+1}$ is approximated as the following ignoring the higher order terms

$$\phi_{k,k+1} = I + F\Delta t + \frac{F^2\Delta t^2}{2}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \Delta t + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \frac{\Delta t^2}{2}$$

$$\phi_{k,k+1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

1.2.1.2 Process noise matrix (Q_k)

$$Q_k = \int_{t_k}^{t_{k+1}} \phi(t_{k+1}, \tau) G(\tau) Q(t) G^T(\tau) \phi^T(t_{k+1}, \tau) d\tau$$

Spectral density matrix Q(t) = $\begin{bmatrix} q_n & 0 & 0 & 0 & 0 \\ 0 & q_e & 0 & 0 & 0 \\ 0 & 0 & q_u & 0 & 0 \\ 0 & 0 & 0 & q_b & 0 \\ 0 & 0 & 0 & 0 & q_d \end{bmatrix}$

$$Q_k = \int_0^{\Delta t} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_n & 0 & 0 & 0 & 0 \\ 0 & q_e & 0 & 0 & 0 \\ 0 & 0 & q_u & 0 & 0 \\ 0 & 0 & 0 & q_b & 0 \\ 0 & 0 & 0 & 0 & q_d \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \tau & 1 \end{bmatrix} d\tau$$

$$= \begin{bmatrix} q_n \Delta t & 0 & 0 & 0 & 0 \\ 0 & q_e \Delta t & 0 & 0 & 0 \\ 0 & 0 & q_u \Delta t & 0 & 0 \\ 0 & 0 & 0 & q_b \Delta t + \frac{\Delta t^3}{3} q_d & \frac{\Delta t^2}{2} q_d \\ 0 & 0 & 0 & \frac{\Delta t^2}{2} q_d & \Delta t q_d \end{bmatrix}$$

1.2.2 Random walk Velocity

1.2.2.1 Transition matrix ($\phi_{k,k+1}$)

$\phi_{k,k+1}$ is approximated as the following ignoring the higher order terms

$$\phi_{k,k+1} = I + F\Delta t + \frac{F^2 \Delta t^2}{2}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \Delta t + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \frac{\Delta t^2}{2}$$

$$\phi_{k,k+1} = \begin{bmatrix} 1 & 0 & 0 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

1.2.2.2 Process noise matrix (Q_k)

$$Q_k = \int_{t_k}^{t_{k+1}} \phi(t_{k+1},\tau) G(\tau) Q(t) G^T(\tau) \phi^T(t_{k+1},\tau) d\tau$$

Spectral density matrix Q(t) = $\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_b & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & q_{vn} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & q_{ve} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & q_{vu} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & q_d \end{bmatrix}$

$$Q_k = \int_0^{\Delta t} \begin{bmatrix} 1 & 0 & 0 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_b & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & q_{vn} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & q_{ve} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & q_{vu} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & q_d \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \Delta t & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \Delta t & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \Delta t & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \Delta t & 0 & 0 & 0 & 1 \end{bmatrix} d\tau$$

$$= \begin{bmatrix} \frac{\Delta t^3}{3} q_{vn} & 0 & 0 & 0 & \frac{\Delta t^2}{2} q_{vn} & 0 & 0 & 0 \\ 0 & \frac{\Delta t^3}{3} q_{ve} & 0 & 0 & 0 & \frac{\Delta t^2}{2} q_{ve} & 0 & 0 \\ 0 & 0 & \frac{\Delta t^3}{3} q_{vu} & 0 & 0 & 0 & \frac{\Delta t^2}{2} q_{vu} & 0 \\ 0 & 0 & 0 & \Delta t q_b + \frac{\Delta t^3}{3} q_d & 0 & 0 & 0 & \frac{\Delta t^2}{2} q_d \\ \frac{\Delta t^2}{2} q_{vn} & 0 & 0 & 0 & \Delta t q_{vn} & 0 & 0 & 0 \\ 0 & \frac{\Delta t^2}{2} q_{ve} & 0 & 0 & 0 & \Delta t q_{ve} & 0 & 0 \\ 0 & 0 & \frac{\Delta t^2}{2} q_{vu} & 0 & 0 & 0 & \Delta t q_{vu} & 0 \\ 0 & 0 & 0 & \frac{\Delta t^2}{2} q_d & 0 & 0 & 0 & \Delta t q_d \end{bmatrix}$$

2 Task 2 – Design matrix for incorporating pseudo range rate measurement into the system.

Since two sets of measurements; pseudo-range and pseudo-range rate measurements are available, we have two parametric model functions, $h_{PR}(x)$ and $h_{PRR}(x)$, for two different measurements. Also, we have eight states (derived in ECEF frame) in our measurement model, 3D positions, clock bias, 3D velocity, and clock drift.

For pseudo-range measurement, $h_{PR}(X) = \sqrt{(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2} + cdt$ (2.1)

Partial derivative of above expression with respect to a given state gives us the element of design matrix corresponding to that element. Also, since the instantaneous position states of the receiver do not reveal anything about instantaneous velocity states, the partial derivative of $h_{PR}(x)$with respect to velocity vx, vy, vz will be zero. Similarly, the partial derivative of $h_{PR}(x)$ with respect to $c\dot{dt}$ will also be zero. Elements of H matrix corresponding to pseudo-range measurement is given in the table below.

Table 1 Elements of H matrix corresponding to pseudo-range measurements (ECEF frame)

$\frac{\partial h_{PR}(x)}{\partial x_r}\Big\|_{x=\hat{x}} = \frac{-(x_s - \hat{x}_r)}{\hat{r}_k}$
$\frac{\partial h_{PR}(x)}{\partial y_r}\Big\|_{y=\hat{y}r} = \frac{-(y_s - \hat{y}_r)}{\hat{r}_k}$
$\frac{\partial h_{PR}(x)}{\partial z_r}\Big\|_{z=\hat{z}r} = \frac{-(z_s - \hat{z}_r)}{\hat{r}_k}$
$\frac{\partial h_{PR}(x)}{\partial cdt}\Big\|_{cdt=\hat{c}dt} = 1$
$\frac{\partial h_{PR}(x)}{\partial vx_r}\Big\|_{vx=v\hat{x}_r} = 0$
$\frac{\partial h_{PR}(x)}{\partial vy_r}\Big\|_{vy=v\hat{y}_r} = 0$
$\frac{\partial h_{PR}(x)}{\partial vz_r}\Big\|_{vz=v\hat{z}_r} = 0$
$\frac{\partial h_{PR}(x)}{\partial \dot{c}dt}\Big\|_{\dot{c}dt=\hat{c}\dot{dt}} = 0$

For pseudo-range rate measurement,

$h_{PRR}(X) = \frac{\Delta\vec{v}_i \bullet \Delta\vec{r}_i}{\rho_i}$: for *i*th satellite.

$$= \frac{(x_{si} - x_r)(\dot{x}_{si} - \dot{x}_r) + (y_{si} - y_r)(\dot{y}_{si} - \dot{y}_r) + (z_{si} - z_r)(\dot{z}_{si} - \dot{z}_r)}{\rho_i} \quad (2.2)$$

Now, $\frac{\partial h_{PRR}(X)}{\partial x_r}\Big|_{x_r=\hat{x}_r} = \frac{-(\dot{x}_{si} - \dot{x}_r)}{\rho_i} + \frac{(x_{si} - x_r)^2(\dot{x}_{si} - \dot{x}_r)}{\rho_i^3}$

Assuming the maximum velocity of satellite in x-axis is about 1000 m/s, for a static user, first term in above expression will be a negative value very close to zero (for a dynamic user, it will be even close to zero). Also, the second term has denominator with ρ^3. Hence, value of above expression is generally a very small negative value close to zero. For a static receiver at (0-latitude and 0-longitude), and when a satellite is exactly in the zenith ($y_s = z_s = 0$, $x_s = x_r + \rho$), above expression is zero (theoretically). Hence, we can assume $\frac{\partial h_{PRR}(X)}{\partial x_r}|_{x_r=\hat{x}_r} \sim 0$

Similarly, it can be shown that following expressions are negative values very close to zero.

$\frac{\partial h_{PRR}(X)}{\partial y_r}|_{y_r=\hat{y}_r} \sim 0$

$\frac{\partial h_{PRR}(X)}{\partial z_r}|_{z_r=\hat{z}_r} \sim 0$

Since the pseudo-range rate does not observe the clock bias state, $\frac{\partial h_{PRR}(X)}{\partial cdt}|_{cdt=c\hat{d}t} = 0$

Partial derivative of $h_{PRR}(X)$ with respect to velocity and clock drift states is below table.

Table 2 Elements of H matrix corresponding to pseudo-range rate measurements (ECEF frame)

$\frac{\partial h_{PRR}(x)}{\partial x_r}\|_{x=\hat{x}} \sim 0$
$\frac{\partial h_{PRR}(x)}{\partial y_r}\|_{y=\hat{y}r} \sim 0$
$\frac{\partial h_{PRR}(x)}{\partial z_r}\|_{z=\hat{z}r} \sim 0$
$\frac{\partial h_{PRR}(x)}{\partial cdt}\|_{cdt=\hat{c}dt} = 0$
$\frac{\partial h_{PRR}(x)}{\partial vx_r}\|_{vx=v\hat{x}_r} = \frac{-(x_s-\hat{x}_r)}{\hat{r}_k}$
$\frac{\partial h_{PRR}(x)}{\partial vy_r}\|_{vy=v\hat{y}_r} = \frac{-(y_s-\hat{y}_r)}{\hat{r}_k}$
$\frac{\partial h_{PRR}(x)}{\partial vz_r}\|_{vz=v\hat{z}_r} = \frac{-(z_s-\hat{z}_r)}{\hat{r}_k}$
$\frac{\partial h_{PRR}(x)}{\partial \dot{c}dt}\|_{c\dot{d}t=\hat{c}\dot{d}t} = 1$

For a system of K satellites, design matrix incorporating both pseud-orange and pseud-orange rate measurements is given below. First K set of rows correspond to K pseudo-range measurements and next K rows correspond to K pseudo-range rate measurements. Column-wise states are given respectively as $[\delta x \quad \delta y \quad \delta z \quad \delta cdt \quad \delta v\dot{x} \quad \delta v\dot{y} \quad \delta v\dot{z} \quad \delta \dot{c}dt]$.

$$\begin{bmatrix} \frac{-(x_{s1}-\hat{x}_r)}{\hat{r}_1} & \frac{-(y_{s2}-\hat{y}_r)}{\hat{r}_1} & \frac{-(z_{s3}-\hat{z}_r)}{\hat{r}_1} & 1 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{-(x_{sK}-\hat{x}_r)}{\hat{r}_K} & \frac{-(y_{sK}-\hat{y}_r)}{\hat{r}_K} & \frac{-(z_{sK}-\hat{z}_r)}{\hat{r}_K} & 1 & 0 & 0 & 0 & 0 \\ \sim 0 & \sim 0 & \sim 0 & 0 & \frac{-(x_{s1}-\hat{x}_r)}{\hat{r}_1} & \frac{-(y_{s2}-\hat{y}_r)}{\hat{r}_1} & \frac{-(z_{s3}-\hat{z}_r)}{\hat{r}_1} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & 1 \\ \sim 0 & \sim 0 & \sim 0 & 0 & \frac{-(x_{sK}-\hat{x}_r)}{\hat{r}_K} & \frac{-(y_{sK}-\hat{y}_r)}{\hat{r}_K} & \frac{-(z_{sK}-\hat{z}_r)}{\hat{r}_K} & 1 \end{bmatrix}$$

2.1 Design matrix in local level frame

Components of magnitude of *relative* position vector (P) between satellite and receiver along North-East-Up axes can be written as,

$$\begin{bmatrix} x_e \\ x_n \\ x_u \end{bmatrix} = \begin{bmatrix} -\cos(el_k)\sin(az_k) \\ -\cos(el_k)\cos(az_k) \\ -\sin(el_k) \end{bmatrix} [|\vec{P}_k|] \qquad (2.1)$$

Considering the clock bias, the pseudo-range measurement equation for *k*th satellite can be constructed as,

$$z_{PR\_k} = \begin{bmatrix} -\cos(el_k)\sin(az_k) & -\cos(el_k)\cos(az_k) & -\sin(el_k) & 1 \end{bmatrix} \begin{bmatrix} x_e \\ x_n \\ x_u \\ b \end{bmatrix} + v_{PR\_k} \qquad (2.2)$$

$$\delta z_k = \begin{bmatrix} -\cos(el_k)\sin(az_k) & -\cos(el_k)\cos(az_k) & -\sin(el_k) & 1 \end{bmatrix} \begin{bmatrix} \delta x_e \\ \delta x_n \\ \delta x_u \\ \delta b \end{bmatrix} + v_{PR\_k}$$

*k*th row H matrix elements for pseudo-range measurement can be written as,

$H_{PR\_k} = \begin{bmatrix} -\cos(el_k)\sin(az_k) & -\cos(el_k)\cos(az_k) & -\sin(el_k) & 1 \end{bmatrix}$

Differentiating equation (2.2) with respect to time we get pseudo-range rate equation and velocity states.

$$z_{PRR\_k} = \begin{bmatrix} -\cos(el_k)\sin(az_k) & -\cos(el_k)\cos(az_k) & -\sin(el_k) & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_e \\ \dot{x}_n \\ \dot{x}_u \\ d \end{bmatrix} + v_{PRR\_k} \qquad (2.3)$$

*k*th row H matrix elements for pseudo-range rate measurement is same as H_{PR_k}, given by,
$H_{PRR\_k} = \begin{bmatrix} -\cos(el_k)\sin(az_k) & -\cos(el_k)\cos(az_k) & -\sin(el_k) & 1 \end{bmatrix}$

For a system of K satellites, design matrix incorporating both pseud-orange and pseud-orange rate measurements in local level frame is given below. First K set of rows correspond to K pseudo-range measurements and next K rows correspond to K pseudo-range rate measurements. Column-wise states are given respectively as $\begin{bmatrix} \delta x_n & \delta x_e & \delta x_u & \delta cdt & \delta \dot{x}_n & \delta \dot{x}_e & \delta \dot{x}_u & \delta \dot{c}dt \end{bmatrix}$.

$$\begin{bmatrix} -\cos el_1 \cos az_1 & -\cos el_1 \sin az_1 & -\sin el_1 & 1 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -\cos el_K \cos az_K & -\cos el_K \sin az_k & -\sin el_K & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\cos el_1 \cos az_1 & -\cos el_1 \sin az_1 & -\sin el_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & -\cos el_K \cos az_K & -\cos el_K \sin az_k & -\sin el_K & 1 \end{bmatrix}$$

3 Task 4 Least-Squares and Kalman Filtering with Static Data

Task details

Dataset considered:	Static data
Measurement used:	Pseudo-range only
Kalman filter:	EKF with random walk position and clock model as discussed in section 1.1
Number of states:	5
States:	Delta change in position along east, north, and vertical axis: δx_e, δx_n, δx_u Delta change in clock bias and clock drift Full state vector includes {Latitude, Longitude, Altitude, Clock bias, and Clock drift}
Zenith pseudo-range error:	0.25 m
R Matrix weighing scheme:	Weighted based on true elevation
A priori information to KF:	From Least square solution in first epoch.
Number of Iteration in LS:	5
Initial guess of states (to LS):	Latitude = 51.0 degrees Longitude = -114.0 degrees Altitude = 1000.0 m Clock bias = Clock drift = 0.0 units

The dataset is processed with single epoch least squares, and KF with four different values of process noise spectral densities for position states, with same value per axis. Values as given in the handout are squared and used to make them consistent with the theory. Process noise spectral density for clock bias and clock drift states are considered as given in the handout.

Figure 1 to Figure 3 show the position error plots. 5 plots are shown on a given figure, representing the position error by a solution from LS, KF with following spectral densities per axis; (i) 1 m/s/√Hz, (ii) 0.1 m/s/√Hz, (iii) 0.01 m/s/√Hz, and (iv) 0.0 m/s/√Hz.

Observations:

- In all the plots LS solution is noisy and less accurate.
- For a high process noise value (q=1 units/axis), KF solution is approximately same as that of LS solution. This is due to the fact with a high process noise associated with the predicted states in the system model; KF gives less weightage to prediction than the information derived from latest observation (measurement). Since the implemented LS (with no *apriori* information) solution receives the information from observation in a same manner, and the states the estimated only from the measurement, we except same solution from both LS and KF.
- Decreasing the process noise (or the uncertainity) associated with the states means that the prediction model is to be trusted more. As it can be seen in the below figures, as the q value decreased, the error in the estimated tends smoother. A little importance is given to the innovation (new information) derived from measurement. If the measurements are affected by systematic errors (probably the middle region in the plots), they did not affect much to the states. This can be clearly seen when the process noise was made zero.
- Zero process noise means that prediction is modeled perfectly. Now, KF highly relies on the a-priori information to update the posteriori states. However, Kalman gain (weightage to measurement innovation) will not be zero as still an uncertainty will be associated with states in the form of co-variance matrix.

- In the north error plot, initially all the curves shows a bias. This could be due to the geometry of satellite. We usually see no satellites towards the northern region of sky over Calgary. Hence North DOP is high. However, even when the measurement suffers from the systematic errors (middle region in time), the north error did not surge up like other cases, rather, KF outputs states relying highly on apriori information.

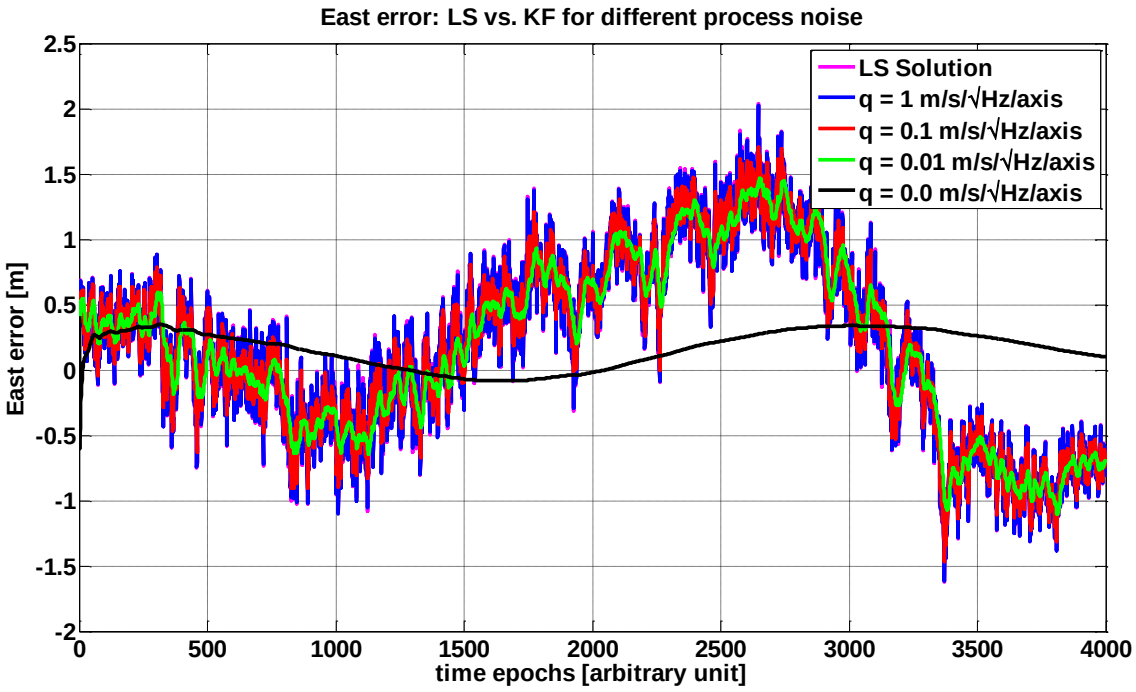


Figure 1 East error: LS vs. KF with different spectral density values

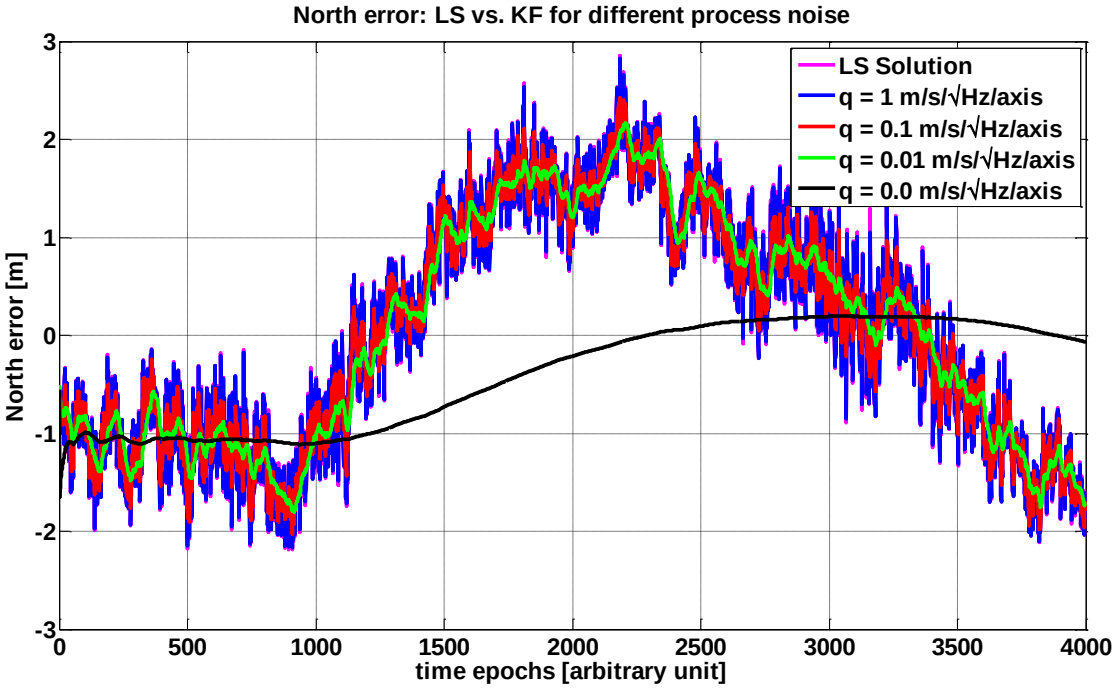


Figure 2 North error: LS vs. KF with different spectral density values

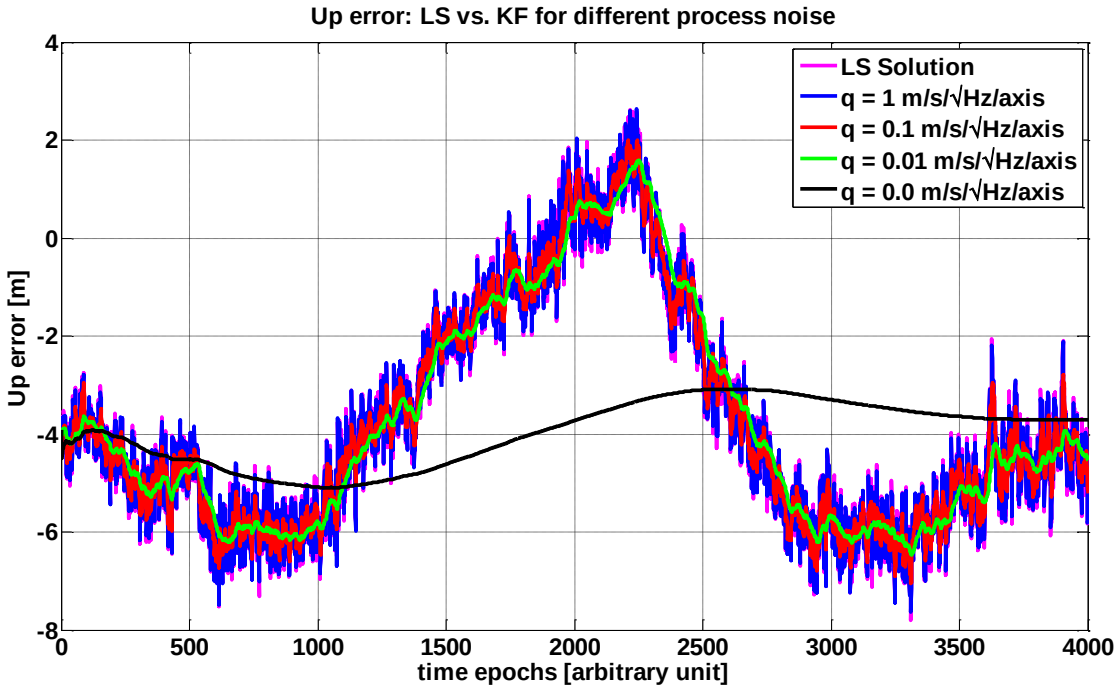
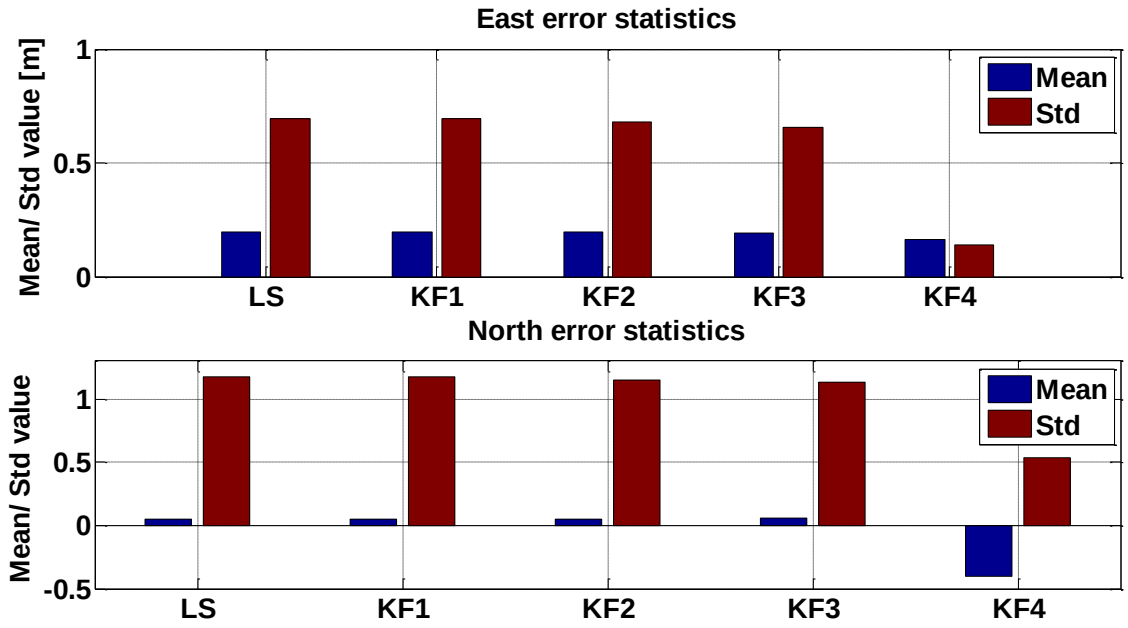


Figure 3 Up error: LS vs. KF with different spectral density values

The below figure summarizes the statistics of 5 methods implemented to process static data.
Notations:
LS – Least-squares
KF1 – Kalman filter with q = 1 units/ Hz
KF2 – Kalman filter with q = 0.01 units/ Hz
KF3 – Kalman filter with q = 0.0001 units/ Hz
KF4 – Kalman filter with q = 0.0 units/ Hz



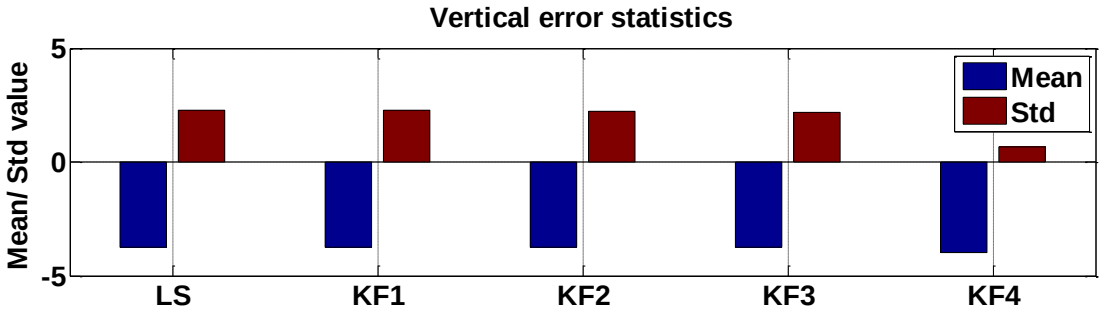


Figure 4 Statistics: Estimators vs. accuracy

Similarities and differences between least square and Kalman filter

Kalman filter always assumes an apriori information of the states is available. Whereas for a LS filter, it is not the case always (Sequential LS is an exception). For a given set of measurements, least squares and KF derive the information from the available measurement in a same manner. LS obtains the *a priori* information from outside source, may be from a point close to the trajectory. Whereas, the KF gets *a priori* information by a prediction from the latest estimates. Prediction is obtained by using the system model which describes how the states evolve (transition in states) over time. System model also includes additional terms in the form of process noise to cater for uncertainty we have in the assumed physical model. If the physical model and its uncertainty are modeled properly, KF will generally output smoother and accurate solutions, unlike the LS.

KF can also estimate the states with fewer observations than states, which is not possible with LS (if no apriori information is available). If some states are not observable by the measurement model, KF can still estimate them unlike a LS filter.

However, if the system model is over constrained, KF is generally more prone to give ‘surges’ in the estimated states. This situation is generally not seen in LS as it operates epoch by epoch. Moreover in KF, blunders in measurements, or erroneous estimated states, if undetected, will be carried forward in time and affect the future estimates as well.

Reference: M. G. Petovello, ‘What are the difference between least-squares and Kalman filtering?’ Inside GNSS March/April 2013

4 Task 5 Kinematic data processing

4.1 Case 1

Task details

Dataset considered:	Kinematic data
Measurement used:	Pseudo-range only
Kalman filter:	EKF with random walk velocity and clock model as discussed in section 1.2.2
Number of states:	8
States:	Delta change in position along east, north, and vertical axis: δx_e, δx_n, δx_u Delta change in clock bias Delta change in velocities along east, north, and vertical axis Delta change in clock drift Full state vector includes {Latitude, Longitude, Altitude, Clock bias, East velocity, North velocity, Up velocity, and Clock drift}
Zenith pseudo-range error:	0.25 m
R Matrix weighing scheme:	Weighted based on true elevation
A priori information to KF:	From Least square solution in first epoch.
Number of Iteration in LS:	5
Initial guess of states (to LS):	Latitude = 51.0 degrees Longitude = -114.0 degrees Altitude = 1000.0 m Clock bias = Clock drift = 0.0 units

As discussed in the previous case, a priori information for the states is obtained from the LS. Since LS cannot estimate velocity states using only the pseudo-range observation, initial a priori values for position states are set manually as follows.

Ve = Vn = Vu = 0 m/s

σVe = σVn = 200 m/s, assumes components along east and north axis with speed of about 80 km/hr in random direction.

σVn = 3 m/s, assumes minimum motion along vertical axis.

Design matrix (H_{NX8}) in measurement model is given as follows.

$$\begin{bmatrix} -\cos el_1 \sin az_1 & -\cos el_1 \cos az_1 & -\sin el_1 & 1 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -\cos el_K \sin az_K & -\cos el_K \cos az_k & -\sin el_K & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

In order to analyze the position and velocity accuracy as a function of time, and to correlate it with actual vehicle dynamics following plots are considered.

- Figure 6, Figure 8, and Figure 11 represent the position error plot in E, N, and U axes respectively.
- Figure 7, Figure 9, and Figure 12 represent the velocity error plot in E, N, and U axes.
- Figure 10 represents the horizontal position plot showing estimated and true trajectory.
- Two cases are considered in each plot, one with q = 1 units/ Hz, and the other with q = 0.0001 units/ Hz (shown on Y2 axes).
- To compare the effect of dynamics on the error plots, following parameters are obtained from true trajectory: true azimuth (vehicle with respect to local north), true east velocity, and true

north velocity. Change in states and their error is highly correlated with these parameters, looking at the system model under consideration. These are shown in Figure 5.

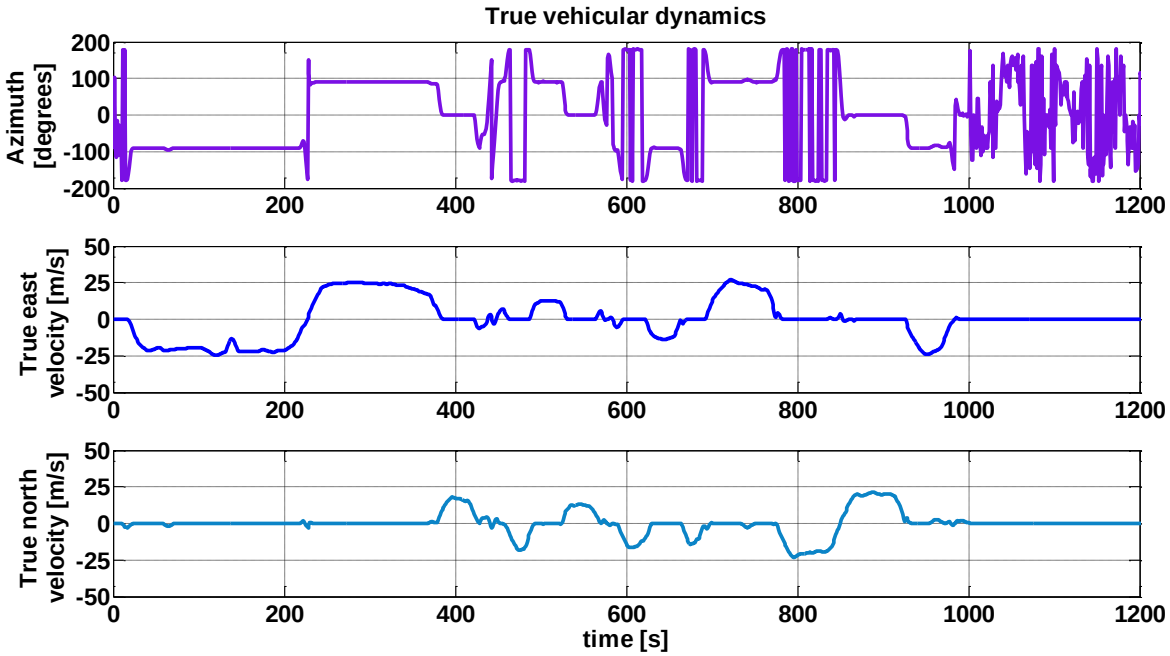


Figure 5 True vehicular dynamics: Azimuth, North velocity, East velocity

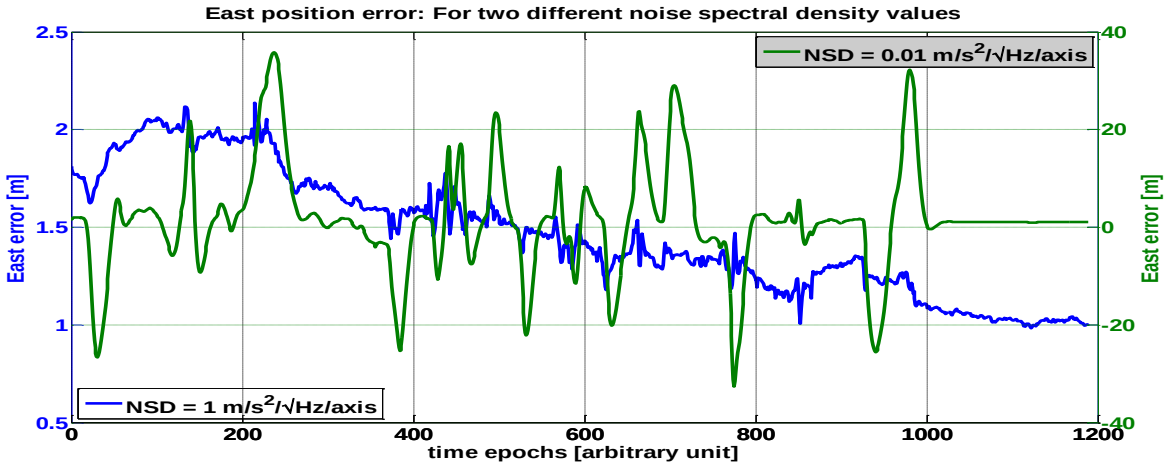


Figure 6 East position error: Kinematic data, Pseudo-range measurement only

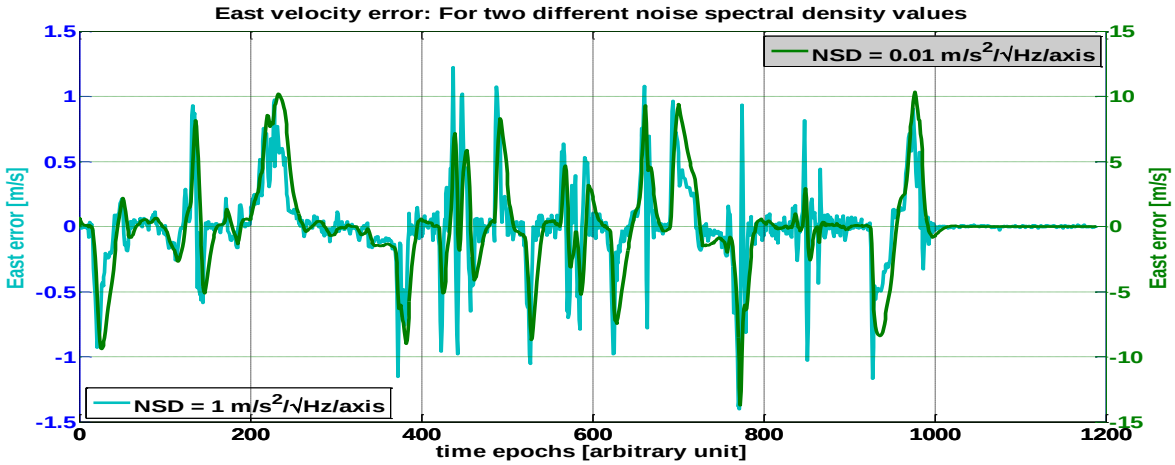


Figure 7 East velocity error: Kinematic data, Pseudo-range measurement only

Observations:

- In all the axes, both position and velocity error is less when noise spectral density (NSD) is high (q = 1 units/Hz). Note that NSD = 0.01 m/s^2/√Hz case shown on Y2-axis has larger scale steps.
- With less q, KF is made to believe forcefully that the system model (1.2.2) is been modeled accurately with less uncertainty. Although the plot looks smoother than the former case, we can see large variations up to 30-40 m in position, and up to 10 m/s in velocity.
- These are particularly observed at the instances when the vehicle is making a turn/ accelerating/ decelerating.
- For example, consider the epoch at around 220th second. At this instant, vehicle makes 180 degree 'U' turn, changing its direction to positive east. We see surge in east error, whereas the north error is not affected much. This effect is shown in Figure 10, first quadrant. We can observe that the trajectory estimated with relatively high q value is in agreement with true trajectory since it overall KF model believes more on the measurement than the predictions at all instances.
- Surge in error can also be due to changing velocity. Consider the epoch at around 150 in Figure 5. Here east velocity decelerates and east position worsens as it is directly related to east velocity.
- Similar effect is observed in north velocity and position plot at about 380th second when vehicle accelerates towards north.

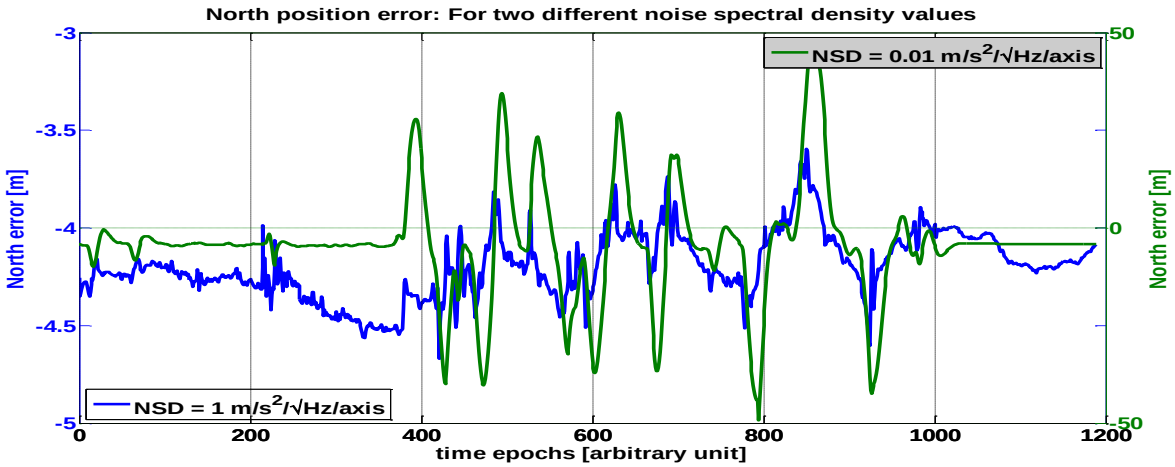


Figure 8 North position error: Kinematic data, Pseudo-range measurement only

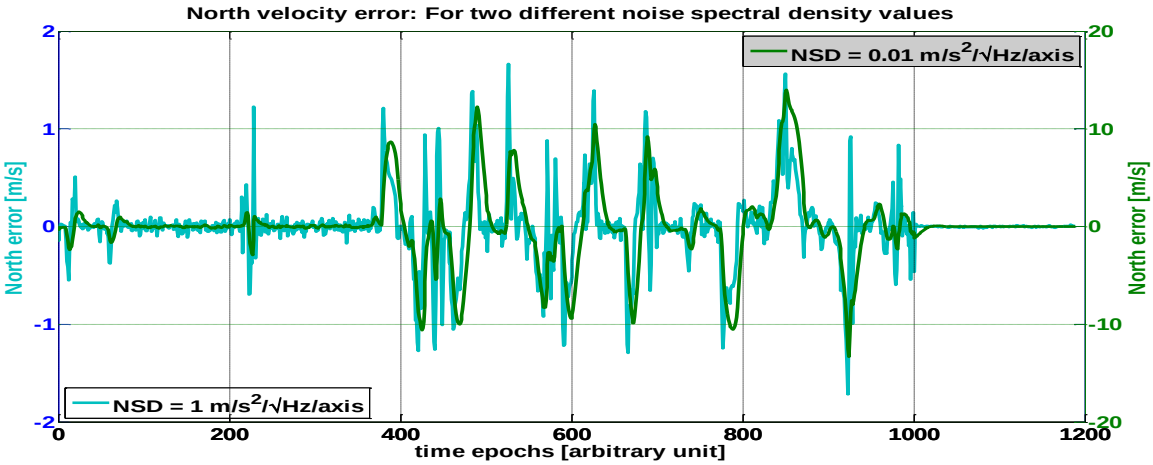


Figure 9 North velocity error: Kinematic data, Pseudo-range measurement only

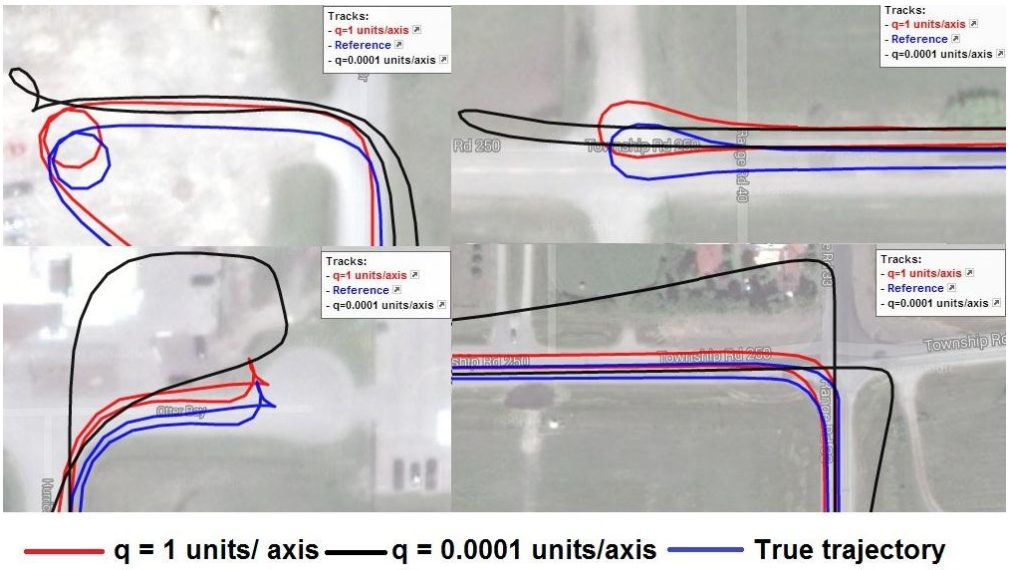


Figure 10 Horizontal position plot: Kinematic data, Pseudo-range measurement only

- Figure 10 shows more such 'sluggish' trajectory estimated by KF initialized with lower q. The filter takes a while to converge the states back to the nominal value. Innovations obtained from measurement accumulated over time (estimated states as a-priory information) cause this effect.
- However effect of reducing 'q' on vertical axis error is not much as compared to other two axes. This is due to the mode in which the data is collected. Since the vehicle was travelling on planar surface, there could be less transverse motion. Hence lesser is the error. However, position error is biased; this could be due to un-modeled systematic errors, which are generally highly correlated with vertical direction.
- In this analysis only pseudo-range measurements are considered. Hence there is less 'innovation' for velocity states from the measurement model. Effect of including additional observations is considered in the next section.
- Over all learning of this experiment is that over-constraining the KF system model (or, any other model in general) has worse effect than under-constraining it.

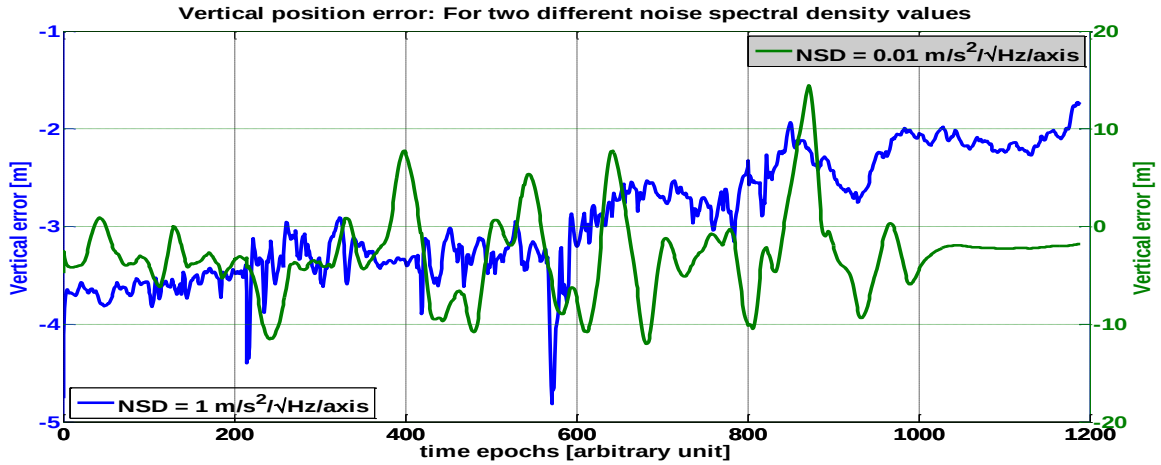


Figure 11 Vertical position error: Kinematic data, Pseudo-range measurement only

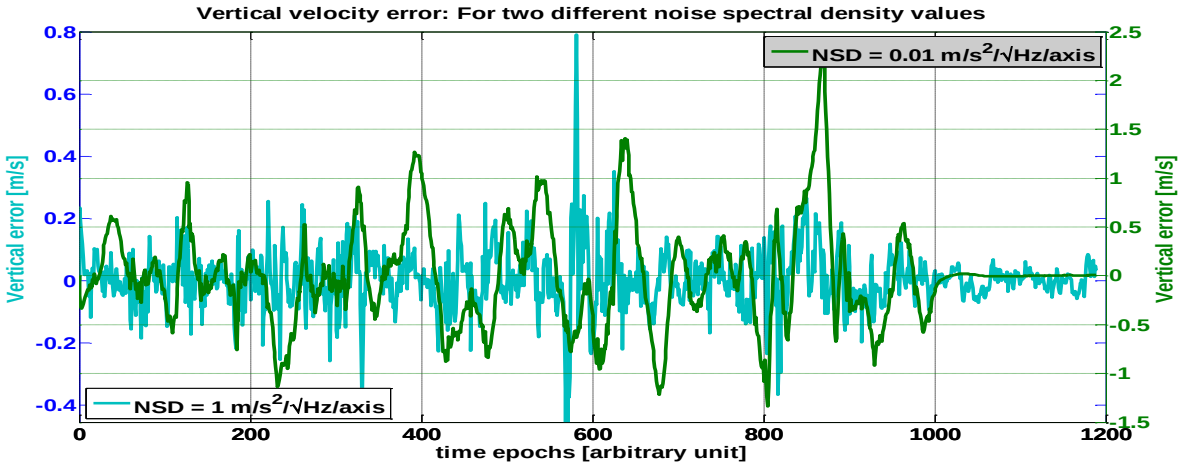


Figure 12 Vertical velocity error: Kinematic data, Pseudo-range measurement only

4.2 Case 2

Dataset considered:	Kinematic data
Measurement used:	Pseudo-range with pseudo-range rate
Kalman filter:	EKF with random walk velocity and clock model as discussed in section 1.2.2
Number of states:	8
States:	Full state vector includes {Latitude, Longitude, Altitude, Clock bias, East velocity, North velocity, Up velocity, and Clock drift}
Zenith pseudo-range error:	0.25 m
Zenith range rate error:	3 cm/s
R Matrix weighing scheme:	Weighted based on true elevation
A priori information to KF:	From Least square solution in first epoch.
Number of Iteration in LS:	5
Design matrix:	As discussed in section 2.1

In order to analyze the position and velocity accuracy as a function of time, and to correlate it with actual vehicle dynamics following plots are considered.

- Figure 13, Figure 15, and Figure 17 represent the position error plot in E, N, and U axes respectively.
- Figure 14, Figure 16, and Figure 18 represent the velocity error plot for E, N, and U axes respectively.
- To compare these figures with their counter-parts in previous section Y axis scale is maintained as before.

Observations:

- Inclusion of new set of measurement has following effects.
    - In all the axes, for both the NSD values, the surge errors in position and velocities have reduced.
    - Especially, the error velocity plot has reduced drastically. This is because; the velocity states are now directly observable by the new measurement model (equation 2.1).
    - Since the velocity estimate has improved, the prediction of positions from improved velocity states is also better now; significantly for east and north position states.
    - Specifically, the effect of dynamics on states is reduced by the added velocity observations. Although the states deviate from actual trajectory at turns, they tend to converge faster to the true trajectory than the earlier case.
    - Effect of over-constraining still persists, although it is reduced. This is because; process noise is independent of the number or the quality of measurement.

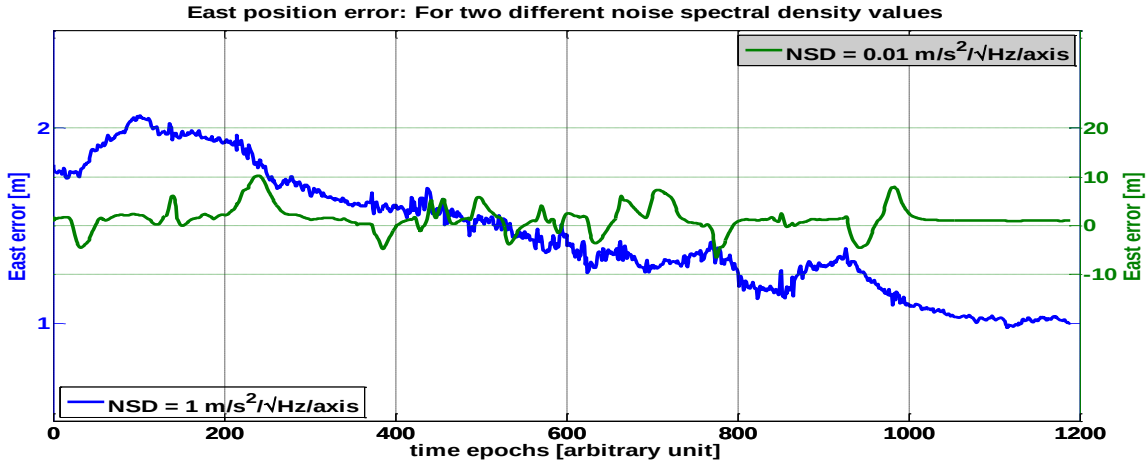


Figure 13 East position error: Kinematic data, Pseudo-range + pseudo-range rate measurement

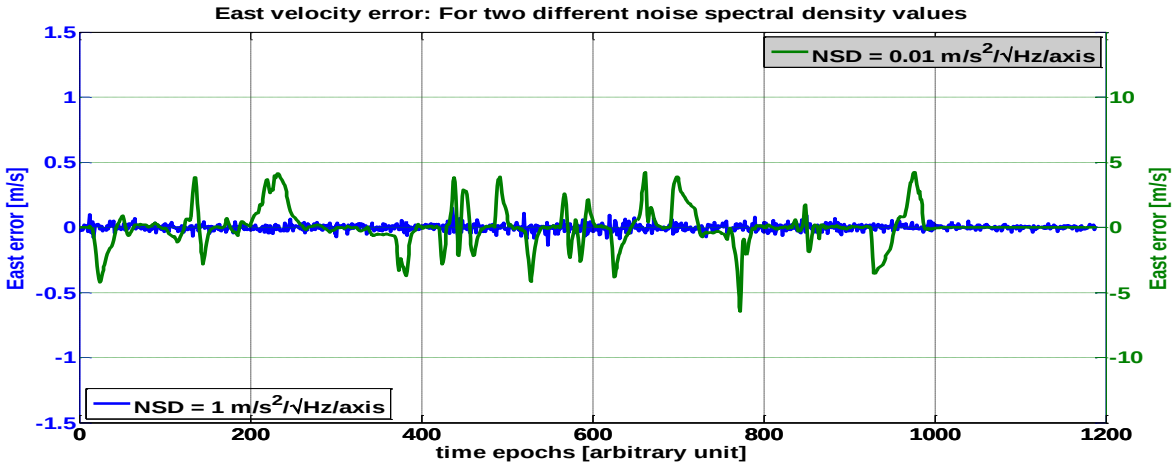


Figure 14 East velocity error: Kinematic data, Pseudo-range + pseudo-range rate measurement

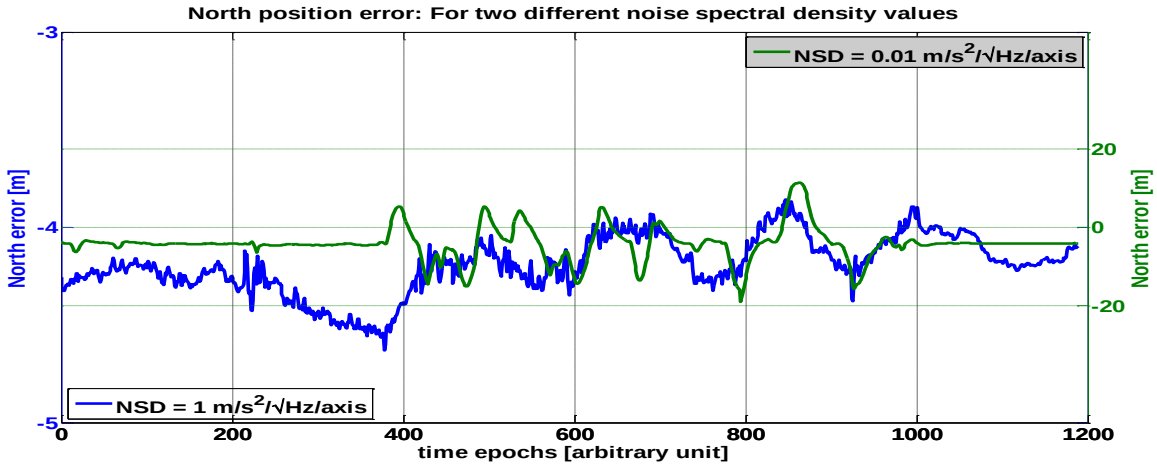


Figure 15 North position error: Kinematic data, Pseudo-range + pseudo-range rate measurement

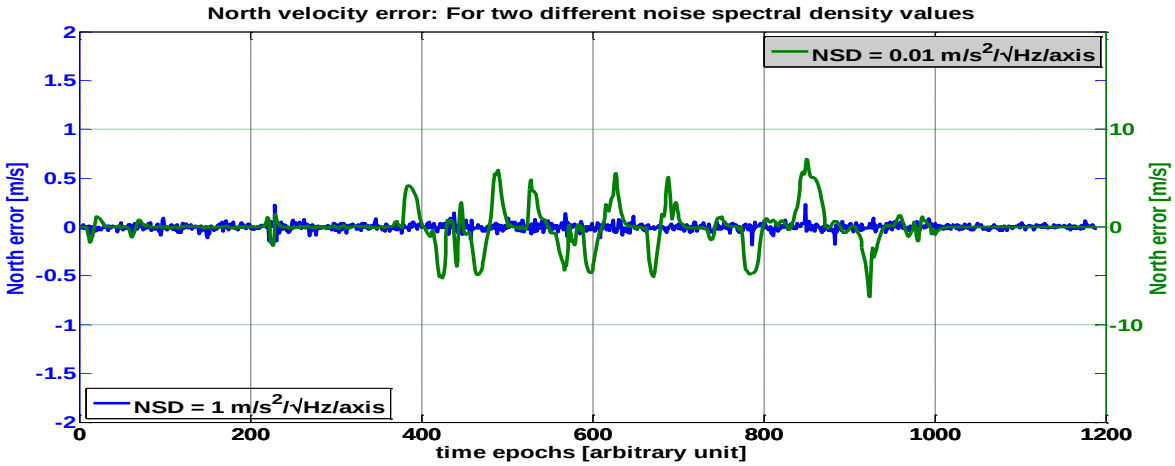


Figure 16 North velocity error: Kinematic data, Pseudo-range + pseudo-range rate measurement

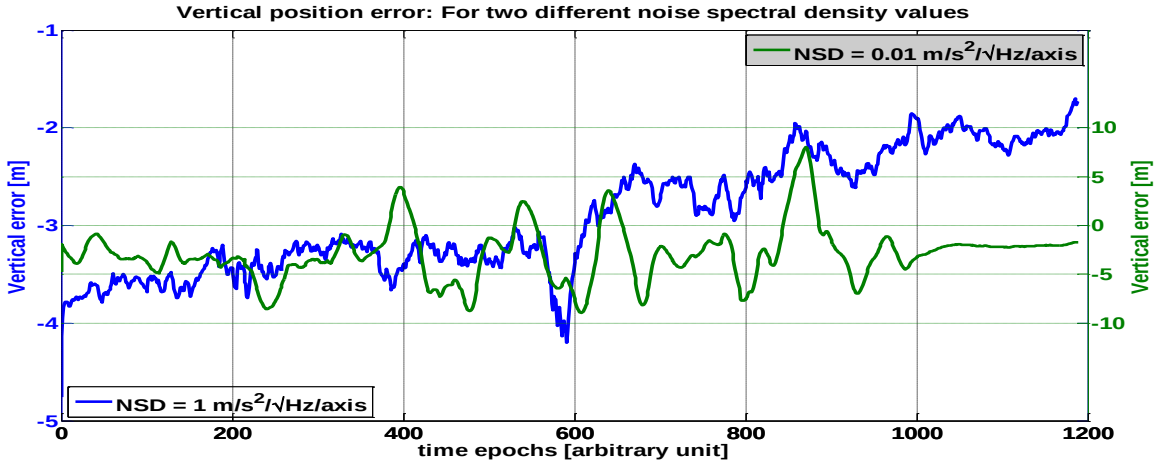


Figure 17 Vertical position error: Kinematic data, Pseudo-range + pseudo-range rate measurement

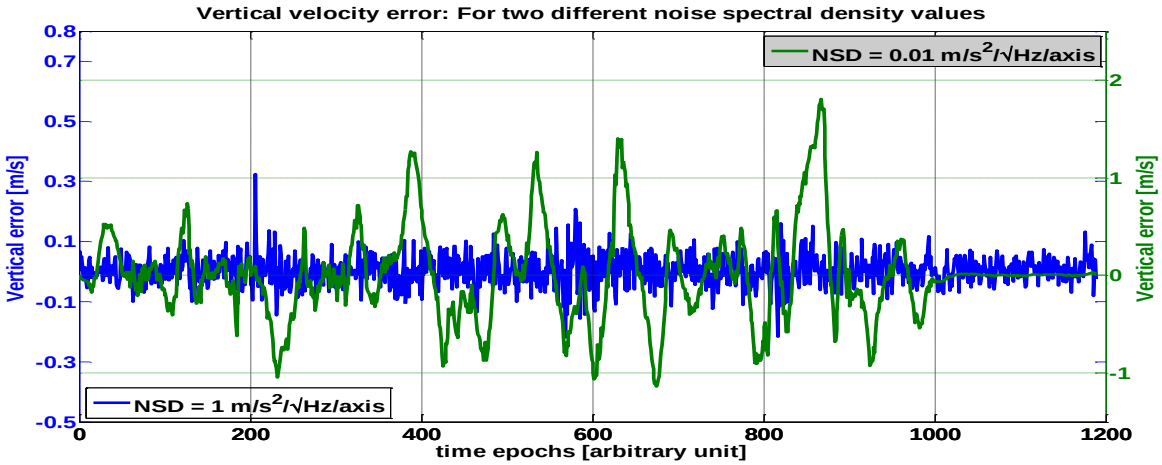


Figure 18 Vertical velocity error: Kinematic data, Pseudo-range + pseudo-range rate measurement

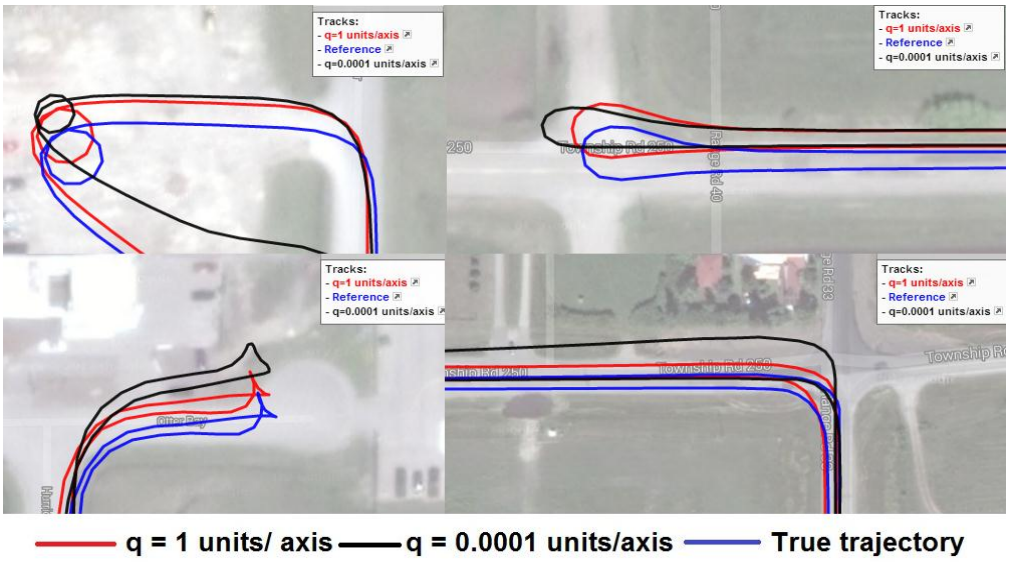


Figure 19 Horizontal position plot: Kinematic data, Pseudo-range + pseudo-range rate measurement

Figure 19 shows two sets of trajectory; q = 1 units/ axis, q = 0.0001 units/ axis, estimated using pseudo-range with pseudo-range rate observations. Comparing this with Figure 10, we can observe that the slug in the trajectory has reduced.

5 Task 6 Circular motion

In this model following assumptions are made for all the options that are derived.

- Receiver is assumed to rotate about a random point in horizontal plane only.
- Altitude is assumed to be constant, true altitude of 1000 m is considered.
- Vertical velocity is assumed to be zero. Hence altitude and vertical velocities are omitted from the state vector

Option 1: Including acceleration states in state vector

State vector $X = [\, xe \quad xn \quad cdt \quad ve \quad vn \quad \dot{cdt} \quad ae \quad an \,]'$

Where *an* and *ae* are the acceleration along north and east axis respectively.
System model can be written as follows, where random constant is assumed for velocity states and random walk for acceleration and clock states.

$$\begin{bmatrix} \delta\dot{x}_e \\ \delta\dot{x}_n \\ \delta\dot{c}dt \\ \delta\ddot{x}_e \\ \delta\ddot{x}_n \\ \delta\ddot{c}dt \\ \delta\dddot{x}_e \\ \delta\dddot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta x_e \\ \delta x_n \\ \delta cdt \\ \delta\dot{x}_e \\ \delta\dot{x}_n \\ \delta\dot{c}dt \\ \delta\ddot{x}_e \\ \delta\ddot{x}_n \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ w_b \\ 0 \\ 0 \\ w_d \\ w_{ae} \\ w_{an} \end{bmatrix}$$

Transition matrix is given as,

$$\phi_{k+1,k} = \begin{bmatrix} 1 & 0 & 0 & \triangle t & 0 & 0 & \triangle t^2/2 & 0 \\ 0 & 1 & 0 & 0 & \triangle t & 0 & 0 & \triangle t^2/2 \\ 0 & 0 & 1 & 0 & 0 & \triangle t & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \triangle t & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \triangle t \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Process noise matrix is given as,

$$Q_k = \begin{bmatrix}
\frac{\Delta t^5}{20}q_{ae} & 0 & 0 & \frac{\Delta t^4}{8}q_{ae} & 0 & 0 & \frac{\Delta t^3}{6}q_{ae} & 0 \\
0 & \frac{\Delta t^5}{20}q_{an} & 0 & 0 & \frac{\Delta t^4}{8}q_{an} & 0 & 0 & \frac{\Delta t^3}{6}q_{ae} \\
0 & 0 & \Delta t q_b + \frac{\Delta t^3}{3}q_d & 0 & 0 & \frac{\Delta t^2}{2}q_d & 0 & 0 \\
\frac{\Delta t^4}{8}q_{ae} & 0 & 0 & \frac{\Delta t^3}{3}q_{ae} & 0 & 0 & \frac{\Delta t^2}{2}q_{ae} & 0 \\
0 & \frac{\Delta t^4}{8}q_{an} & 0 & 0 & \frac{\Delta t^3}{3}q_{an} & 0 & 0 & \frac{\Delta t^2}{2}q_{an} \\
0 & 0 & \frac{\Delta t^2}{2}q_d & 0 & 0 & \Delta t q_d & 0 & 0 \\
\frac{\Delta t^3}{6}q_{ae} & 0 & 0 & \frac{\Delta t^2}{2}q_{ae} & 0 & 0 & \Delta t q_{ae} & 0 \\
0 & \frac{\Delta t^3}{6}q_{an} & 0 & 0 & \frac{\Delta t^2}{2}q_{an} & 0 & 0 & \Delta t q_{an}
\end{bmatrix}$$

Above system model fits the relation between position, velocity, and acceleration perfectly in a *linear motion*. We know that the trajectory is a circle, hence it is better to adapt a model that considers the curvilinear motion. Moreover, the number of states in the model is eight. Hence a second *circular motion* model is considered which involves only 7 states.

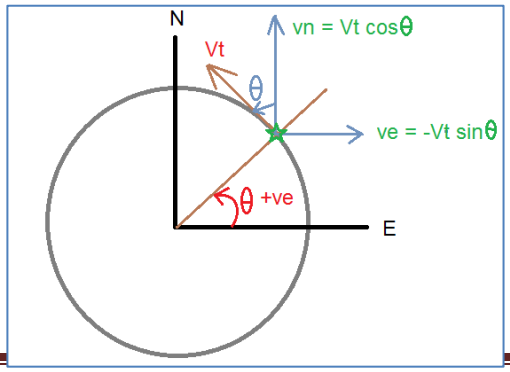
Option 2: Angular rate model

Following are the assumptions.

- Angle θ, is made by moving receiver with respect to east axis, with center of rotation as center.
- Antilock-wise motion makes a positive θ
- It is given that the receiver moves with a constant speed, hence tangential velocity is constant
  $V_t = sqrt\ (Ve^2 + Vn^2)$
  Since tangential speed is constant, angular speed ώ = dθ/dt is also constant.
- Theta is given by $\theta = atan2\ (-Ve/Vn)$

With these assumptions, state vector includes only seven states, given by

State vector $X = [\ \delta xe \quad \delta xn \quad \delta cdt \quad \delta ve \quad \delta ve \quad \delta cd\dot{t} \quad \dot{\theta} \quad]'$



$$ve = -\sqrt{ve^2 + vn^2}\sin\theta$$
$$\dot{ve} = -\sqrt{ve^2 + vn^2}\cos\theta$$
$$\dot{ve} = -V_t\cos\theta.\dot{\theta}$$
$$vn = \sqrt{ve^2 + vn^2}\cos\theta$$
$$\dot{vn} = -V_t\sin\theta.\dot{\theta}$$

Figure 20 Circular motion

Hence, system model can be written as,

$$\begin{bmatrix} \delta\dot{x}_e \\ \delta\dot{x}_n \\ \delta\dot{c}dt \\ \delta\ddot{x}_e \\ \delta\ddot{x}_n \\ \delta\ddot{c}dt \\ \delta\ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -Vt\cos\theta \\ 0 & 0 & 0 & 0 & 0 & 0 & -Vt\sin\theta \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta x_e \\ \delta x_n \\ \delta cdt \\ \delta\dot{x}_e \\ \delta\dot{x}_n \\ \delta\dot{c}dt \\ \delta\dot{\theta} \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ w_b \\ w_{ve} \\ w_{vn} \\ w_d \\ w_{\dot{\theta}} \end{bmatrix}$$

Although dθ/dt is constant, its process noise is set as a random walk to cater for uncertainty in predicting θ from Ve and Vn. Also, yet another constant value, Vt is also derived from Ve and Vn.

Transition matrix is given by,

$$\phi_{k+1,k} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 & -Vt\cos\theta.\Delta t^2/2 \\ 0 & 1 & 0 & 0 & \Delta t & 0 & -Vt\sin\theta.\Delta t^2/2 \\ 0 & 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -Vt\cos\theta.\Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 & -Vt\sin\theta.\Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Process noise matrix is a huge complex matrix. Hence, it is represented in following text format (from MATLAB™ code).

Where, ***th = θ***

K =Vt

dt =Δt

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    QMatrix(1,1) = SDV(4)*((dt).^3)/3 + SDV(7)*K*K*cos(th)*cos(th)*((dt).^5)/20;
    QMatrix(1,2) = SDV(7)*K*K*cos(th)*sin(th)*((dt).^5)/20;
    QMatrix(1,3) = 0;
    QMatrix(1,4) = SDV(4)*((dt).^2)/2 + SDV(7)*K*K*cos(th)*cos(th)*((dt).^4)/8;
    QMatrix(1,5) = SDV(7)*K*K*cos(th)*sin(th)*((dt).^4)/8;
    QMatrix(1,6) = 0;
    QMatrix(1,7) = -SDV(7)*K*cos(th)*((dt).^3)/6;

    QMatrix(2,1) = SDV(7)*K*K*cos(th)*sin(th)*((dt).^5)/20;
    QMatrix(2,2) = SDV(5)*((dt).^3)/3 + SDV(7)*K*K*sin(th)*sin(th)*((dt).^5)/20;
    QMatrix(2,3) = 0;
    QMatrix(2,4) = SDV(7)*K*K*sin(th)*cos(th)*((dt).^3)/6;
    QMatrix(2,5) = SDV(5)*((dt).^2)/2 + SDV(7)*K*K*sin(th)*sin(th)*((dt).^4)/8;
    QMatrix(2,6) = 0;
    QMatrix(2,7) = -SDV(7)*K*sin(th)*((dt).^3)/6;

    QMatrix(3,1) = 0;
    QMatrix(3,2) = 0;
    QMatrix(3,3) = SDV(3)*(dt) + SDV(6)*((dt).^3)/3;
    QMatrix(3,4) = 0;
    QMatrix(3,5) = 0;
    QMatrix(3,6) = SDV(6)*((dt).^2)/2;
    QMatrix(3,7) = 0;

    QMatrix(4,1) = SDV(4)*((dt).^2)/2 + SDV(7)*K*K*cos(th)*cos(th)*((dt).^4)/8;
    QMatrix(4,2) = SDV(7)*K*K*cos(th)*sin(th)*((dt).^4)/8;
    QMatrix(4,3) = 0;
    QMatrix(4,4) = SDV(4)*(dt) + SDV(7)*K*K*cos(th)*cos(th)*((dt).^3)/6;
    QMatrix(4,5) = SDV(7)*K*K*cos(th)*sin(th)*((dt).^3)/6;
    QMatrix(4,6) = 0;
    QMatrix(4,7) = -SDV(7)*K*cos(th)*((dt).^2)/2;



    QMatrix(5,1) = SDV(7)*K*K*cos(th)*sin(th)*((dt).^4)/8;
    QMatrix(5,2) = SDV(5)*((dt).^2)/2 + SDV(7)*K*K*sin(th)*sin(th)*((dt).^4)/8;
    QMatrix(5,3) = 0;
    QMatrix(5,4) = SDV(7)*K*K*sin(th)*cos(th)*((dt).^3)/3;
    QMatrix(5,5) = SDV(5)*(dt) + SDV(7)*K*K*sin(th)*sin(th)*((dt).^3)/3;
    QMatrix(5,6) = 0;
    QMatrix(5,7) = -SDV(7)*K*sin(th)*((dt).^2)/2;

    QMatrix(6,1) = 0;
    QMatrix(6,2) = 0;
    QMatrix(6,3) = SDV(6)*((dt).^2)/2;
    QMatrix(6,4) = 0;
    QMatrix(6,5) = 0;
    QMatrix(6,6) = SDV(6)*(dt);
    QMatrix(6,7) = 0;
```

```
    QMatrix(7,1) = -SDV(7)*K*cos(th)*((dt).^3)/6;
    QMatrix(7,2) = -SDV(7)*K*sin(th)*((dt).^3)/6;
    QMatrix(7,3) = 0;
    QMatrix(7,4) = -SDV(7)*K*cos(th)*((dt).^2)/2;
    QMatrix(7,5) = -SDV(7)*K*sin(th)*((dt).^2)/2;
    QMatrix(7,6) = 0;
    QMatrix(7,7) = SDV(7)*(dt);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Choice of measurement model

Since the trajectory involves continuously changing north and east velocities as indicated in Figure 20, measurement model should involve more observability to velocity states. Hence in this context, both pseudo-range and pseudo-range rate observations are considered. Measurement model is as given in Section 2.

Result and observations:

- Figure 21 shows the estimated trajectory overlapped on true trajectory.
- KF assumed following spectral density values for the states.
    - qve = 1 units/ Hz
    - qvn = 1 units/ Hz
    - qθ = 0.01 units/ Hz
    - For clock states, as given in the hand-out
- Same data is processed with the Velocity model method (Task 5), without changing any KF settings. Spectral density values are initialized as the above case. Estimated trajectory from velocity model is given in Figure 22.
- There is no significant change in the estimated trajectory between the two cases, except that the one estimated from angular rate is a little closer to the true trajectory.
- However, it was observed that the velocity states estimated from velocity model method show spikes at some instances (Figure 25), significantly at the start and ending of the trajectory. Whereas, the velocity error plot from angular rate model is a small sinusoidal error as shown in Figure 23.
- Velocity model method has no prediction model to estimate angular change in the velocity.
- A small bias is observed in both the methods when processed with qve = qvn = 1units/Hz
- When the q value is increased to 20, estimated trajectory overlaps with the true value as shown in Figure 24.

With these observations, following points can be concluded.

- Although angular rate model method predicts angle of rotation rate, it is only able to given good transient response than the velocity model method
- Steady state response of both the models are model or less the same for the 'given data'
- By increasing q in angular rate model, precision has improved with a little loss in accuracy.
- In any given model, there always exists trade-off between what accuracy to expect vs. precision. Like any other filter, KF has a delay associated input to output transition. This has a little contribution in producing a small offset in the estimated position.
- By increasing q, solution behaves as though LS and is able to give better precision, which means, for the given scenario, system model under consideration still has a large uncertainty.
- Since the previous measurements are considered to estimate dθ/dt (and in turn other states), it seems that measurement rate is not sufficient considering the speed and radius of rotation. It may be necessary, either to have a faster measurement rate, or a faster prediction rate. However

unless the time gap between measurements are reduced, predicting θ would not a linear solution as it is assumed to be.

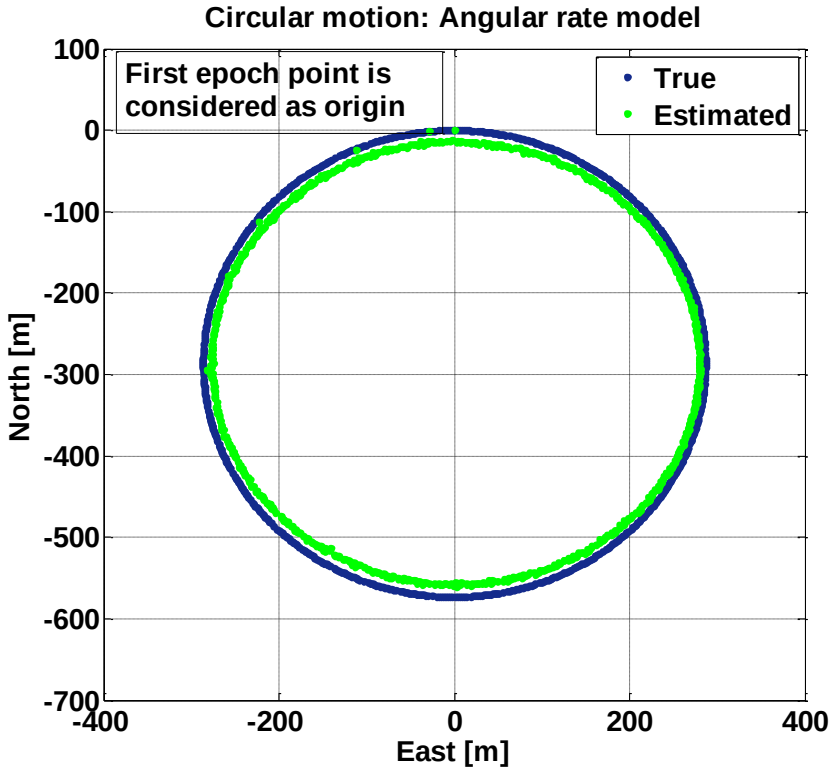


Figure 21 Trajectory with angular rate model qve = qvn = 1 units

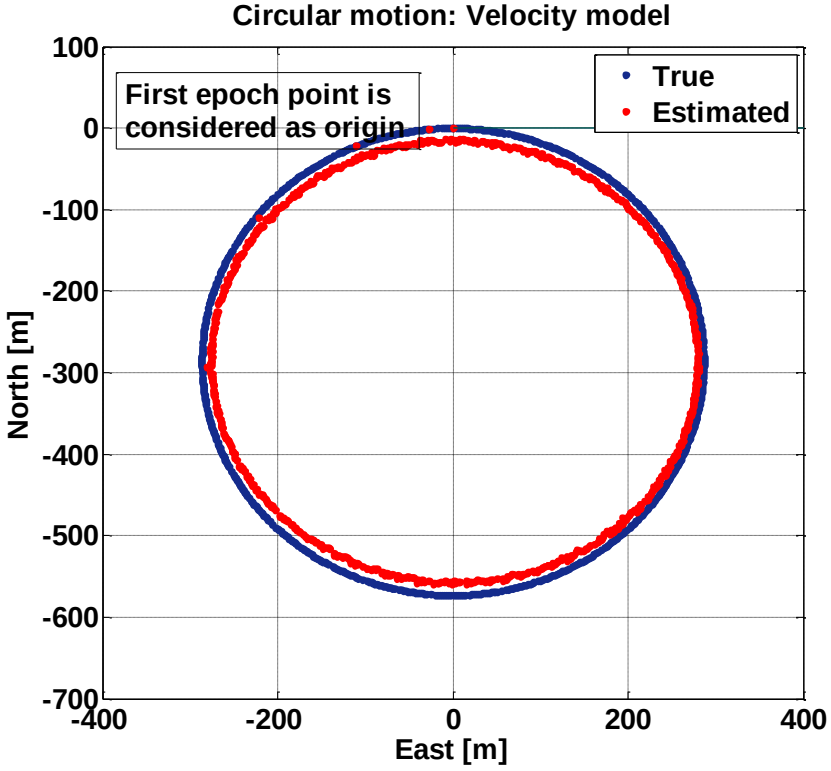


Figure 22 Trajectory with velocity model as discussed in Task 5 with qve = qvn = 1 units

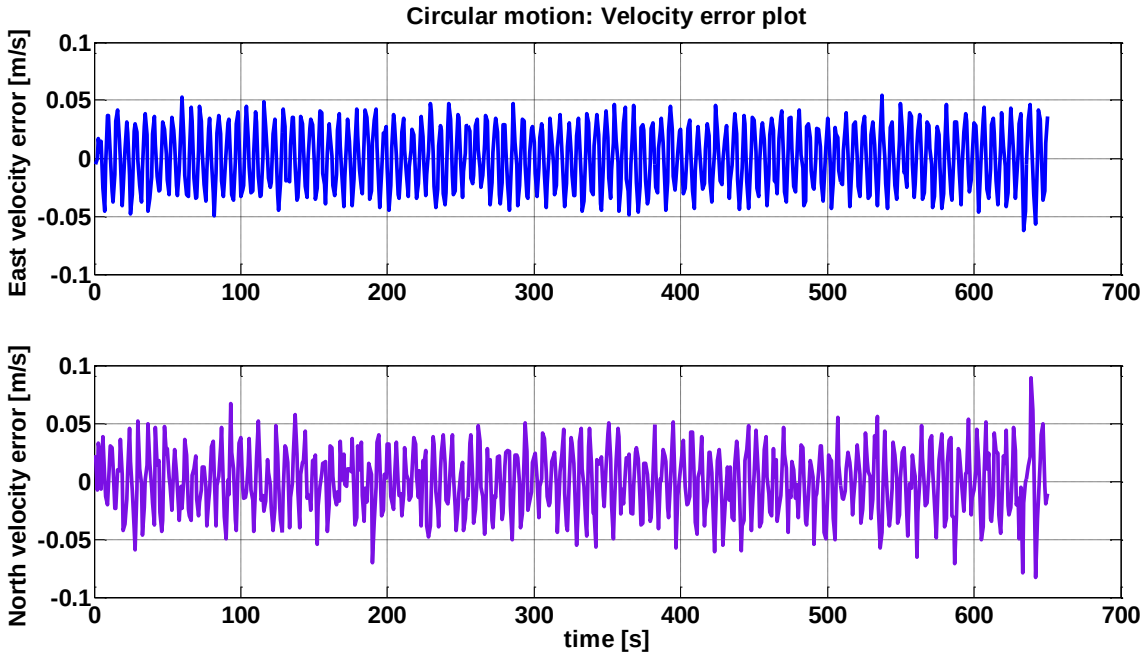


Figure 23 Circular motion with angular rate model north and east velocity errors with qve = qvn = 1 units

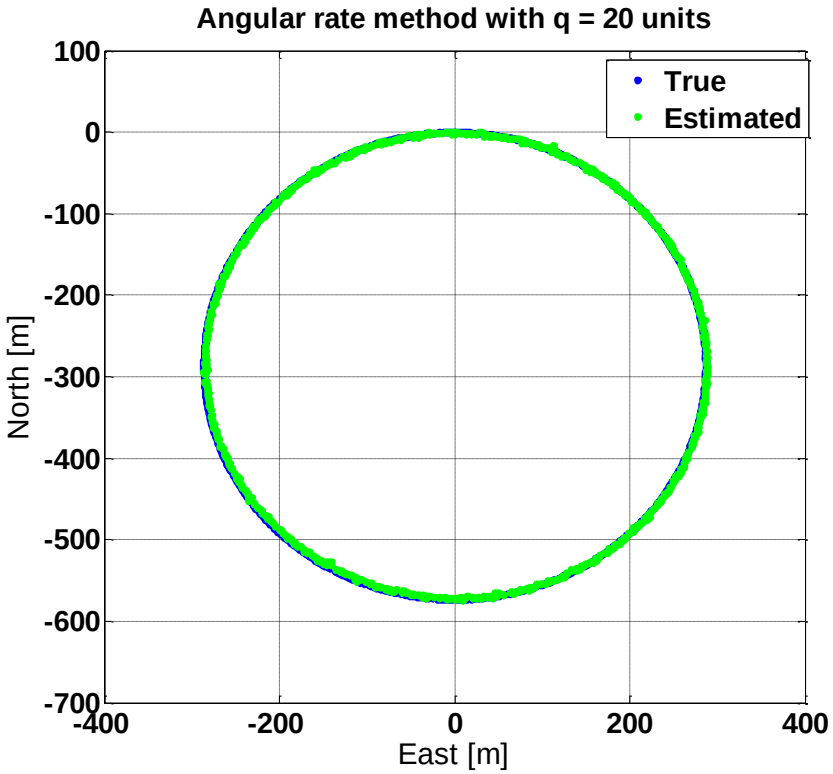


Figure 24 Angular rate model with qve = qvn = 20 units

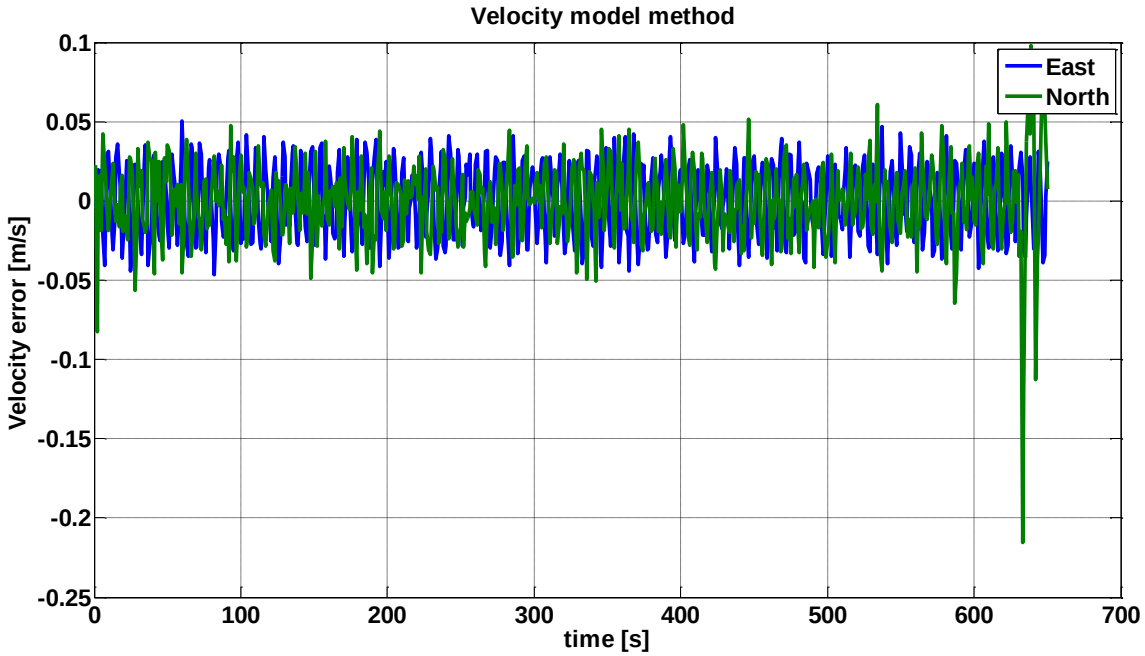


Figure 25 Spikes in velocity error at the start and end of motion: Velocity model method